

Vilniaus universitetas

Matematikos ir informatikos fakultetas

Įvadas į statistiką su R

Remigijus Lapinskas

2003 m. rugsėjis

Turinys

Įvadinės pastabos

1. Aprašomoji statistika
 - 1.1. Keli statistikos uždavinių pavyzdžiai
 - 1.2. Kintamųjų tipai
 - 1.3. Ėminiai ir jų grafinės charakteristikos
 - 1.4. Ėminiai ir jų skaitinės charakteristikos
2. R – bendrieji faktai
 - 2.1. R instaliacija
 - 2.2. R ekranas
 - 2.3. R bibliotekos ir duomenų rinkiniai
 - 2.4. R literatūra, konferencija, archyvai
3. Duomenų įrašymas ir programavimo pavyzdžiai
 - 3.1. Duomenų įrašymas rankomis
 - 3.1.1. Skaitiniai vektoriai ir matricos
 - 3.1.2. Kompleksiniai ir loginiai vektoriai
 - 3.1.3. Simboliniai vektoriai ir matricos
 - 3.1.4. Duomenų sistemos
 - 3.1.5. Vardiniai kintamieji (faktorai)
 - 3.1.6. Ranginiai kintamieji
 - 3.1.7. Sąrašai
 - 3.2. Duomenų importas ir eksportas
 - 3.3. (Pseudo)atsitiktinių skaičių generavimas
 - 3.4. Apie R funkcijas ir `source` komandą
 - 3.5. Programavimo pavyzdžiai
4. Vienmačiai duomenys: aprašomoji statistika ir duomenų priešanalizė
 - 4.1. Vardiniai kintamieji
 - 4.2. Skaitiniai kintamieji
 - 4.2.1. Histogramos
 - 4.2.2. Kvantilių grafikai
 - 4.2.3. Stačiakampės diagramos
 - 4.2.4. Skaitinės charakteristikos
 - 4.2.5. Funkcija `eda . shape`
5. Dvimačiai duomenys: aprašomoji statistika ir duomenų priešanalizė
 - 5.1. Vardiniai kintamieji
 - 5.2. Mišrus atvejis: vardiniai ir skaitiniai kintamieji
 - 5.3. Skaitiniai atvejis
6. Daugiamatiai duomenys: aprašomoji statistika ir duomenų priešanalizė
 - 6.1. Duomenų pertvarkos
 - 6.2. Grafinė analizė
 - 6.3. Skaitinės charakteristikos
7. Centrinė ribinė teorema ir didžiųjų skaičių dėsnis
 - 7.1. Centrinė ribinė teorema (vienmatis atvejis)
 - 7.2. Centrinė ribinė teorema (daugiamatis atvejis)
 - 7.3. Didžiųjų skaičių dėsnis

8. Sprendžiamoji statistika: parametrų įverčiai
 - 8.1. Taškiniai įverčiai
 - 8.2. Intervaliniai įverčiai
9. Sprendžiamoji statistika: hipotezių tikrinimas (vienas ėminys)
 - 9.1. Hipotezės apie proporciją
 - 9.2. Hipotezės apie vidurkį
 - 9.3. Pareto skirstinys
 - 9.4. Hipotezės apie medianą
 - 9.5. Suderinamumo kriterijai
10. Sprendžiamoji statistika: hipotezių tikrinimas (du ėminiai)
 - 10.1. Hipotezės apie proporcijas
 - 10.2. Hipotezės apie požymių nepriklausomumą (dažnių lentelės)
 - 10.3. Hipotezės apie vidurkius
 - 10.4. Hipotezės apie "centrų" lygybę
 - 10.5. Hipotezės apie skirstinių lygybę

Įvadinės pastabos

Šiame konspekte yra aprašomas kompiuterinis matematinės statistikos paketas R ir jo taikymai, skirti pradinėms matematinės statistikos savokoms iliustruoti.

Statistiko darbas visuomet buvo susijęs su (dažnai labai dideliais) skaičiavimais, todėl kompiuterių atsiradimas turėjo milžiniškos įtakos kaip teorinės taip ir praktinės statistikos vystymuisi. Šiuo metu egzistuoja labai daug statistikai skirtų kompiuterinių paketų, o tarp jų kūrėjų vyksta ganėtinai aštri konkurencija. Tenka pripažinti, kad dabar daugumos populiarių komercinių paketų galimybės yra daugmaž vienodos, o vieno ar kito produkto pasirinkimą dažnai nulemia gamintojo reklaminės veiklos intensyvumas ar aptarnavimo kokybė, kaina, profesiniai interesai ar tiesiog pripratimas.

Programinę įrangą galima klasifikuoti, remiantis įvairiais kriterijais. Jei kalbėtume apie kainą, tai vienai grupei priklauso komerciniai produktai (juos reikia pirkti, “piratinių” kopijų naudojimas yra ir amoralus, ir baudžiamas). Jai priklauso tokie statistiniai paketai kaip SAS, SPSS, Statistica, Statgraphics, S-Plus, Stata, Gauss, Ox, TSP, Minitab, EViews (=Econometric Views) ir t.t. Studentams skirtos šių produktų versijos, kaip taisyklė, kainuoja žymiai pigiau, pvz., ekonometrijai skirtas Eviews 4.1 paketo individuali licencija universiteto darbuotojui kainuoja 360 eurų, o jo Student Version 3.1 studentui kainuoja 40 eurų. Kitai grupei priklauso nemokami (free) produktai (dėl suprantamų priežasčių, jie kartais yra menkesnės kokybės). Į šią grupę kartais pakliūna komercinių paketų senesnės versijos (pvz., NCSS 6.0 Junior versija), nekomercinių organizacijų produktai (pvz., Europos Sąjungos statistikos departamento (Eurostat’o) laiko eilučių analizei skirta DEMETRA) arba įvairių entuziastų (arba jų grupių) kūryba (pvz., Herman’o J. Bierens’o ekonometriškai skirtas EasyReg 2000 arba Luke Tierney sukurtas produktas XLISP-STAT). Skyrium stovi pasaulinės (programuojančių) statistikų bendruomenės GNU programos pagrindu¹ kuriamas produktas R) – tai sparčiai vystomas tarptautinis projektas, kuris jau dabar leidžia spręsti praktiškai visus statistikos uždavinius.

Kitas paketų klasifikavimo kriterijus galėtų būti statistinės analizės komandų vykdymo būdas. Dauguma aukščiau išvardintų paketų turi meniu tipo komandų sistemą – norint apskaičiuoti, tarkime, (ėminio) vidurkį, užtenka pateiktajame komandų sąrašė spragtelėti ant **Mean** langelio ir ekrane bus pateiktas skaičiavimo rezultatas. Kito tipo paketuose (programuojamuose paketuose) komandą `mean(x)` komandiniame lange reiktų surinkti pačiam ir tik po to pamatytume rezultatą. Abu būdai turi privalumų ir trūkumų. Pirmasis būdas paprastesnis (nereikia mokytis gana greitai užmirštamų komandų), tačiau antrasis statistikos profesionalui teikia daugiau galimybių. Kartais sakoma: jei jums reikia lentelių ir grafikų (variantas: jeigu mėgstate kiną) – naudokitės pirmojo tipo produktais, o jei statistinės analizės (variantas: jeigu mėgstate skaityti knygas) – antrojo. Pažymėsime, kad kai kurie antrojo tipo produktai (pvz., S-Plus, SAS) dabar turi abi galimybes. Antra vertus, dauguma pirmo tipo produktų dabar irgi turi didesnes ar mažesnes programavimo galimybes (pvz., toks yra SPSS ar Eviews). R iš esmės yra programavimo kalba su specializuota (statistikos reikmėms skirta)

¹ “... The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software – to make sure the software is free for all its users...”

aplinka. Kai kas iš R vystymo branduolio (The R Development Core Team) mano, kad meniu variantas apskritai nereikalingas, tačiau progresas šia kryptimi yra akivaizdus (plg. `library(Rcmdr)`). Šiame konspekte R paketo meniu galimybėmis nesinaudosime².

R yra nemokamas (ir labai geros kokybės) produktas. R yra “jaunesnysis” komercinio S-Plus paketo “brolis”. Šių dviejų kalbų sintaksė praktiškai ta pati, nors programiniai interpretavimo principai skiriasi. Dauguma S kalba parašytų programų veikia ir R aplinkoje.

R projekto internetinis adresas yra <http://www.r-project.org/>, tačiau teisingiau būtų kreiptis į vieną iš CRAN (=Central R Archive Network) veidrodinių kopijų (mes paprastai naudosimės <http://cran.hu.r-project.org/> svetaine). Ten rasite paskutinę R versiją (šiuo metu R 1.7.1), įvairiems statistikos skyriams skirtas bibliotekas ir kitą informaciją (pvz., įvairių autorių parašytus R vadovus). Daug su R susijusios informacijos rasite autoriaus sudarytame kompaktiniame diske R1 (jį galima nemokamai gauti iš remigijus.lapinskas@maf.vu.lt).

Trumpai aprašysime bendrąją konspekto struktūrą. 1 skyriuje yra pateiktos pagrindinės aprašomosios statistikos sąvokos. 2 ir 3 skyriuose skaitytojas ras trumpą R paketo aprašymą ir įvadą į programavimą R kalba. Paprastai, prieš pradėdant taikyti statistikos kriterijus, atliekama duomenų priešanalizė – tai aptarta 4, 5 ir 6 skyriuose. 7 skyrius skirtas ribinėms tikimybių teorijos teorems, tiksliau kalbant, centrinei ribinei teoremai ir didžiųjų skaičių dėsniai. 8 skyriuje aptartos populiacijos parametrų vertinimo procedūros. 9 ir 10 skyriai skirti statistinių hipotezių tikrinimui.

² Tiksliau sakant, naudosimės tik tomis minimaliomis R grafinių sąsajų (=GUI=Grafical User's Interface) galimybėmis, kurias matome 2.1 pav. antroje eilutėje.

1. Aprašomoji statistika

1.1. Keli statistikos uždavinių pavyzdžiai

Panagrinėkime kelis būdingus statistikos uždavinius.

1.1 pvz. Norint patikrinti teiginį, kad Lietuvos gyventojų ūgis kasmet didėja, 2000 metais buvo išmatuotas tūkstančio atsitiktinai paimtų vyrų (jų amžius buvo tarp 20 ir 25 metų) ūgis. (Vieno tūkstančio didumo) ėminiu¹ vadiname skaičių rinkinį $(x_1, x_2, \dots, x_{1000})$ (čia x_i yra i -jo vyro ūgis), o panašus (bet tik žymiai didesnis) skaičių rinkinys, kurį gautume išmatavę visus nurodyto amžiaus Lietuvos vyrus, vadinamas populiacija². Panašus tyrimas, bet su 1500 vyrais, buvo atliktas ir 1995 bei 1990 metais. Aišku, kad skaičių turime labai daug (žr. 1.1 lentelę žemiau), todėl norėdami patikrinti mūsų hipotezę, juos turėtume pateikti suprantamu, sutrauktu arba kondensuotu pavidalu. Tuo užsiima aprašomoji (descriptive) statistika. Antra vertus, sakykime, 2000-ųjų metų vyrų ūgio vidurkis yra didesnis už 1995-ųjų. Bet ar tai iš tikrųjų reiškia, kad vyrų ūgis padidėjo? Juk gal tik šitam tūkstančiui vyrų vidurkis didesnis, kitam tūkstančiui jis gal būtų mažesnis? Būdas, kurie leidžia ėminio rezultatus praplėsti visai populiacijai, nagrinėja sprendžiamoji (inferential) statistika.

1.1 lentelė
2000 metų ūgio matavimo rezultatai (duomenų rinkinys u2000)

[1]	180	179	174	183	179	179	184	181	179	183	174	184	177	182	181	180	176	183	182
[20]	181	176	175	176	179	181	178	177	176	180	175	184	181	179	177	184	176	178	183
[39]	177	181	177	178	181	189	181	176	179	186	183	183	191	175	177	184	174	176	184
[58]	181	182	179	173	168	178	173	182	174	186	185	182	179	179	173	184	189	176	178
[77]	183	178	175	180	180	184	176	188	174	178	178	178	180	175	179	179	178	185	180
[96]	181	177	182	180	179	181	181	179	174	183	179	184	181	177	178	181	178	185	183
[115]	182	188	181	171	176	176	180	180	186	178	183	177	186	185	182	183	189	178	181
[134]	173	181	178	177	181	174	177	192	182	173	177	179	191	180	187	172	186	179	177
[153]	182	180	180	178	185	191	179	181	184	178	185	185	179	178	181	184	178	186	178
[172]	172	186	186	179	186	182	187	184	186	184	188	180	178	189	178	182	183	177	179
[191]	172	185	188	169	181	184	174	174	185	167	181	182	189	185	167	183	187	179	174
[210]	187	178	183	185	183	175	187	167	171	178	183	186	182	180	180	179	183	176	186
[229]	176	178	188	177	185	178	188	181	182	178	178	185	187	182	180	183	180	190	181
[248]	172	183	175	182	189	182	186	178	183	180	178	187	179	181	178	173	177	188	173
[267]	174	179	183	181	188	170	190	183	180	171	169	179	177	184	177	178	175	176	175
[286]	179	183	183	183	177	187	178	181	172	187	180	185	181	178	170	178	177	180	181
[305]	182	186	185	182	176	186	182	175	173	180	181	180	175	175	179	173	174	181	182
[324]	175	181	166	182	179	176	180	176	175	186	189	179	174	181	183	178	187	177	175
[343]	182	178	178	180	180	180	182	186	175	179	188	181	174	176	183	180	179	179	184
[362]	183	183	178	179	180	192	180	183	181	177	184	183	179	184	178	183	178	171	178
[381]	176	185	184	183	181	183	174	181	176	180	176	176	179	182	177	178	175	181	188
[400]	185	170	177	171	175	179	178	182	178	176	179	171	175	176	183	179	176	174	181
[419]	179	182	184	182	182	162	179	172	183	185	186	178	182	192	181	181	175	177	177
[438]	181	188	178	182	178	191	178	177	174	184	179	176	183	175	167	173	181	173	181
[457]	186	182	175	181	179	177	181	176	180	177	170	186	198	184	185	180	182	174	175
[476]	182	176	179	178	176	184	170	177	172	176	185	179	179	175	181	176	187	172	180
[495]	183	176	175	174	187	173	186	180	183	182	181	185	181	182	170	177	183	179	184
[514]	172	168	183	180	172	182	177	188	178	178	182	184	181	177	180	175	177	175	177
[533]	191	182	191	186	185	181	174	184	183	176	180	188	183	173	187	184	182	173	177
[552]	184	177	178	183	179	177	177	171	178	183	181	186	178	177	176	182	179	182	180
[571]	170	176	188	179	179	179	178	177	177	179	188	185	189	189	177	171	189	183	187
[590]	179	176	177	186	185	184	174	179	182	186	187	183	179	181	184	186	179	183	178
[609]	180	182	176	179	170	182	185	181	182	181	182	176	177	189	174	176	179	177	184
[628]	180	179	175	184	180	178	186	179	179	178	172	171	177	178	179	184	180	178	177
[647]	177	186	187	175	175	179	177	178	185	179	176	178	178	185	174	161	177	177	179

¹ Žodžio “ėminys” sinonimas yra “imtis”.

² Atkreipsime dėmesį – populiacija vadinsime ne Lietuvos vyrų rinkinį, bet visų jų ūgio matavimo rezultatų rinkinį

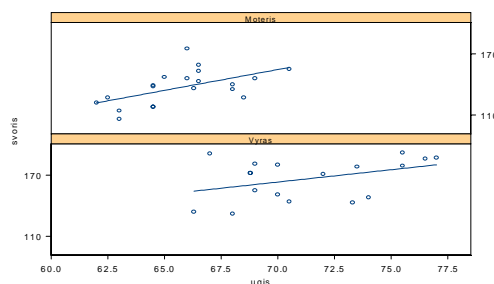
[666] 173 178 177 180 181 181 190 183 179 174 177 174 182 181 184 190 179 175 186
 [685] 192 181 183 181 181 190 183 174 184 185 187 184 182 185 178 180 183 175 188
 [704] 183 174 185 181 179 175 183 182 181 182 182 173 181 177 176 175 175 179 173
 [723] 182 186 182 181 169 184 185 182 184 185 178 186 185 180 182 173 174 182 178
 [742] 171 184 179 182 177 179 174 182 178 183 177 183 181 178 176 173 177 179 184
 [761] 173 181 181 187 179 192 180 180 181 186 171 172 179 175 176 184 181 185 183
 [780] 176 184 175 178 179 178 185 172 169 174 187 187 190 171 180 175 180 173 182
 [799] 188 178 177 176 174 181 176 178 181 185 181 189 180 174 177 184 176 182 182
 [818] 187 176 181 183 180 179 180 178 178 183 186 180 175 178 184 182 182 181 176
 [837] 183 179 178 184 172 173 180 180 177 181 177 175 188 179 180 177 188 177 185
 [856] 177 178 177 176 180 181 183 169 185 176 177 182 186 180 185 179 177 168 185
 [875] 174 190 172 176 188 181 186 177 177 182 177 178 191 182 177 174 175 175 183
 [894] 181 173 178 175 191 176 174 180 170 178 176 176 174 180 181 193 180 177 177
 [913] 186 177 180 180 180 183 181 180 184 185 176 174 181 176 186 176 179 176 184
 [932] 183 176 185 178 178 180 188 181 188 175 187 181 184 183 183 177 174 186 172
 [951] 173 180 181 185 176 184 178 169 173 180 181 187 180 185 173 184 174 181 189
 [970] 175 174 177 172 175 183 178 179 175 180 179 183 182 175 180 186 179 186 176
 [989] 176 174 177 182 180 180 175 191 176 175 177 179

1.2 pvz. Žemiau yra pateiktas duomenų rinkinio IQ dalis (jį radome internete, žr. <http://lib.stat.cmu.edu/DASL/Stories/BrainSizeandIntelligence.html>). Čia lytis (kin-tamasis x_{1i}) įgyja reikšmes M (=Moteris) arba V (=Vyras), ūgis (x_{2i}) užrašytas coliais, svoris (x_{3i}) – svarais, o NA (= Not Available) reiškia, kad dėl kažkokių priežasčių matavimo rezultatas nėra žinomas. Šį kartą ėminiui vadinsime trejetų rinkinį $((x_{11}, x_{21}, x_{31}), \dots, (x_{40,1}, x_{40,2}, x_{40,3}))$, o pats rinkinys vadinamas trimačiu. Atkreipsime dėmesį į tai, kad pirmoji komponentė nėra skaičius.

1.2 lentelė
Dalis duomenų rinkinio IQ

lytis	ūgis	svoris					
[1]	M	64.5	118	[14]	M	70.5	155
[2]	V	72.5	NA	[15]	M	66.0	146
[3]	V	73.3	143	[16]	M	68.0	135
[4]	V	68.8	172	[17]	M	68.5	127
[5]	M	65.0	147	[18]	V	73.5	178
[6]	M	69.0	146	[19]	M	66.3	136
[7]	M	64.5	138	[20]	V	70.0	180
[8]	M	66.0	175	[21]	V	NA	NA
[9]	V	66.3	134	[22]	V	76.5	186
[10]	V	68.8	172	[23]	M	62.0	122
[11]	M	64.5	118	[24]	V	68.0	132
[12]	V	70.0	151	[25]	M	63.0	114
[13]	V	69.0	155	[26]	V	72.0	171
				[27]	M	68.0	140
				[28]	V	77.0	187
				[29]	M	63.0	106
				[30]	M	66.5	159
				[31]	M	62.5	127
				[32]	V	67.0	191
				[33]	V	75.5	192
				[34]	V	69.0	181
				[35]	M	66.5	143
				[36]	M	66.5	153
				[37]	V	70.5	144
				[38]	M	64.5	139
				[39]	V	74.0	148
				[40]	V	75.5	179

Aišku, kad žmogaus svoris priklauso ir nuo ūgio ir nuo lyties. Norėdami susidaryti įspūdį apie šią priklausomybę, šiuos duomenis pateiksime grafiškai (žr. 1.1 pav. žemiau). Paveiksle pateiktos dvi vadinamosios sklaidos diagramos, vyrams ir moterims atskirai, kiekvienoje taip pat išbrėžtos bendrą svorio priklausomybės nuo ūgio tendenciją atspindinčios tiesės.



1.1 pav. Moterų (viršuje) ir vyrų (apačioje) svorio priklausomybės nuo ūgio grafikai

1.3 pvz. Marketingo padalinys nori išsiaiškinti, kuris iš dviejų naujų gaminių bus labiau perkamas. Kadangi skonis yra pakankamai subjektyvus kriterijus, potencialūs pirkėjai buvo prašomi įvertinti abu gaminius skaičiais nuo 0 iki 4 (jų prasmė paaiškinta lentelėje):

0	1	2	3	4
Labai nepatiko	Nepatiko	Neturiu nuomonės	Patiko	Labai patiko

Kiekvieną gaminį vertino po 15 pirkėjų, štai jų atsakymai:

pirmojo gaminio įverčiai: 3 2 3 0 1 0 2 1 0 0 4 3 2 0 3

antrojo gaminio įverčiai: 4 2 2 0 3 3 1 2 2 4 3 2 3 2 2

Atrodo, kad antrasis gaminys vertinamas geriau, tačiau kaip tai pagrįsti? Vienas iš būdų yra toks - apjunkime abu ėminius į vieną ir išdėstykite įverčius didėjimo tvarka: 0 0 0 0 0 1 1 1 2 2 2 2 2 2 2 2 2 2 3 3 3 3 3 3 3 3 4 4 4. Jei pirmasis gaminys būtų tiek pat geras kaip ir antrasis, tai jo įverčiai (jie pabraukti) būtų daugmaž tolygiai išsidėstę apjungtame ėminyje. Matome, kad daugiau pabrauktų įverčių yra kairėje, todėl matyt geresnis yra antrasis produktas. Deja, tikslaus atsakymo į šį klausimą teks palaukti iki 10 skyrelio (taikysime Vilkoksono rangų sumų testą, žr. 10.2 pvz.).

1.4 pvz. 112 kritiškos būsenos ligonių lentelėje šokas suklasifikuoti pagal du požymius: jei pacientas į ligoninę pakliuvo šoko būsenoje, tai (žr. 1.3 lentelę) Šokas=1, priešingu atveju – 0; jei pacientas ligoninėje mirė – 1, priešingu atveju – 0. Šiame pavyzdyje “skaičiais” 0 ir 1 iš tikrųjų yra užšifruoti žodžiai ar netgi (būseną aprašantys) sakiniai – aišku, kad, tarkime, skirtumas 1-0 čia jokios prasmės neturi.

1.3 lentelė
Duomenų rinkinys šokas

Šokas Ar mirė?								
1	1	1	39	0	0	78	0	0
2	1	1	40	0	0	79	1	1
3	1	0	41	1	1	80	0	0
4	1	1	42	1	1	81	1	1
5	0	0	43	0	0	82	1	0
6	1	1	44	0	0	83	1	0
7	1	0	45	1	1	84	1	1
8	0	0	46	1	1	85	1	0
9	1	1	47	0	0	86	1	1
10	1	1	48	0	0	87	1	1
11	1	1	49	0	0	88	1	1
12	0	0	50	0	0	89	1	0
13	1	0	51	0	0	90	0	0
14	1	1	52	1	0	91	1	1
15	1	1	53	1	1	92	1	1
16	1	1	54	0	0	93	1	1
17	0	0	55	1	0	94	0	0
18	1	0	56	1	1	95	1	1
19	0	0	57	0	0	96	0	0
20	1	0	58	0	0	97	1	1
21	1	0	59	1	1	98	1	1
22	1	1	60	1	0	99	0	0

23	0	0	61	0	0	100	1	1
24	1	1	62	1	0	101	1	1
25	1	1	63	1	0	102	1	0
26	0	0	64	1	1	103	0	0
27	1	1	65	0	0	104	1	0
28	1	0	66	0	0	105	1	0
29	1	0	67	0	0	106	1	1
30	1	0	68	1	1	107	1	1
31	0	1	69	0	0	108	1	0
32	0	0	70	1	0	109	1	1
33	0	0	71	1	1	110	1	1
34	1	0	72	1	1	111	0	0
35	1	0	73	1	0	112	1	1
36	0	1	74	1	1			
37	1	0	75	1	1			
38	1	0	76	0	0			

Štai tipiškas klausimas: ar priklauso paciento likimas nuo to, su šoku ar be jo, jis buvo atgabentas į ligoninę? Paprastai tokie, kokybiniai, duomenys sumuojami požymių sąveikos (kitaip - dažnių) lentelėje. Atrodo, kad šoko buvimas turi didelę įtaką ligo-

Požymių sąveikos lentelė

		Ar mirė?		Eilučių suma
		Ne	Taip	
Šokas	Nėra	35	2	37
	Yra	29	46	75
Stulpelių suma		64	48	112

nio likimui, tačiau tai pagrįsti galėsime tik vėliau (žr. 10.1 užduotį).

1.2. Kintamųjų tipai

Pereisime prie kiek nuoseklesnio dėstymo. Vieno populiacijos objekto stebėjimo (matavimo) rezultatas vadinamas įrašu (įrašų kiekis vadinamas ėminio dydžiu). Tuo atveju, kai matuojame tik vieną parametą (pvz., ūgį), įrašas turės vienintelę komponentę, o pats stebimasis dydis (ir pats ėminys) vadinamas vienmačiu. Jei mums rūpi kelios stebimojo objekto charakteristikos (pvz., lytis, ūgis ir svoris), įrašą sudarys p komponentių (paminėtuojų atveju $p=3$), o pats stebimasis dydis (ir ėminys) vadinamas p -mačiu.

Komponentes sudaro įvairių tipų kintamieji. Dažniausiai sutinkami **skaitiniai** (kitaip: kardinalieji ar kiekybiniai) kintamieji, jie dar skirstomi į tolydžiuosius (temperatūra, tūris, laikas,...) ir diskrečiuosius (vaikų šeimoje skaičius, avarių ar klientų per dieną skaičius,...). Bet kuris iš šių tipų skirstomas dar į dvi grupes – santykinis (svoris, ūgis ir pan.; prasmę turi ne tik svorių skirtumas, bet ir santykis) ir skirtuminius (temperatūra, IQ, data; skirtumai turi prasmę, tačiau santykiai - ne). Dydžio priskyrimas vienai ar kitai grupei iš esmės priklauso nuo to, galime ar ne įvesti prasmingą nulį. Santykiniams kintamiesiems teiginys pavidalo: Jonas (60 kg) yra du kartus sunkesnis už mažąjį Augustą (30 kg), yra teisingas (nes, kokius svorio vienetus beįvestume, sąvoka “svoris lygus nuliui” reiškia tą patį), tačiau teigti, kad šiandien (+20°C) yra dvigubai šilčiau negu vakar (+10°C) yra neteisinga, nes, pvz., Farenheito skalėje šios tempera-

tūros užrašomos kaip $+68^{\circ}\text{F}$ ir $+50^{\circ}\text{F}$. Panašiai yra su intelektualumo koeficientu IQ (nes psichologai nesutaria, ką reiškia $\text{IQ}=0$) ar su data (paprastai atskaitos pradžia (pvz., Kristaus gimimas) yra susitarimo reikalas). Antra vertus, teiginys, kad futbolo rungtynių kėlinys trunka du su puse karto ilgiau negu krepšinio, yra teisingas (laiko momentų (ar temperatūrų) skirtumas yra santykinis kintamasis, nes sąvoka “įvykis truko 0 laiko” yra vienareikšmiškai suprantamas).

Kitą dydžių klasę sudaro **ranginiai** (kitaip: tvarkos arba ordinalieji (lot. *ordinatio* – sutvarkymas)) kintamieji (socialinė grupė, išsimokslinimas, ...). Tarkime, kad keturių komandų turnyre komandos U, V ir Z po pirmojo rato surinko atitinkamai 45, 16 ir 18 taškų. Kadangi 0 yra natūrali vertinimo skalės pradžia, taškų skaičius yra skaitinis santykinis kintamasis. Antra vertus, sporto esmė yra kuo aukštesnė vieta, todėl komandas galima išdėstyti pagal užimtą vietą. Kitais žodžiais, U yra 1-ji, V – 3-ji, o Z – 2-ji komanda, tačiau dabar skaičiai 1, 2, ir 3 yra komandos vieta arba rangas. Tiesą sakant, tai netgi ne skaičiai, o simboliai, komandas mes galime pavadinti auksine, sidabrine arba bronzine (A, S ir B). Jei tartume, kad pirmenybėse buvo ir antrasis ratas, kuriame komandos surinko atitinkamai 28, 30 ir 12 taškų, tai aišku, kad jų taškus galima (ir reikia) sudėti, tačiau simbolių (ranginių kintamųjų) A, S ir B suma prasmės neturi.

	1-as ratas	2-as ratas	Galutinis rezultatas
U	45	28	73
	A	S	A
V	16	30	46
	B	A	S
Z	18	12	30
	S	B	B

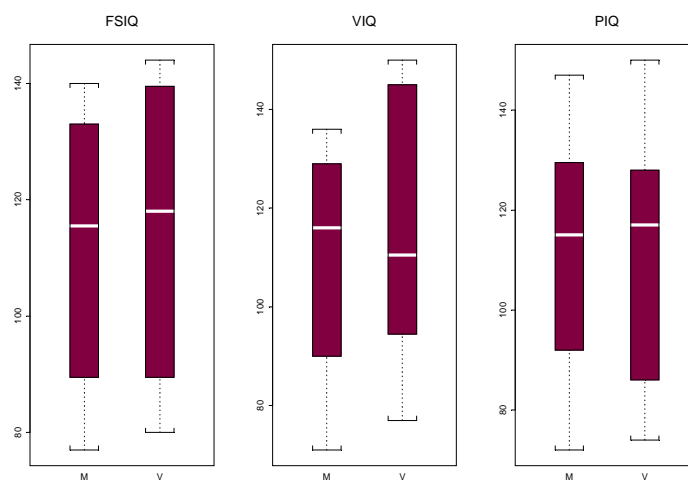
Trečią didelę dydžių klasę sudaro **vardiniai**³ (kitaip: kategoriniai, kokybiniai arba nominalieji (nominus - lot. vardas)) kintamieji (akių spalva, socialinė grupė, automobilio gamintojo vardas,...). Šiuo atveju jokio natūralaus išdėstymo “didėjimo tvarka” nėra. Štai būdingas uždavinys: ar vyrų ir moterų intelektualumo koeficientas IQ toks pat? Žemiau yra pateikta aukščiau minėto duomenų rinkinio IQ dalis:

lytis	FSIQ	VIQ	PIQ				
M	133	132	124	V	83	83	86
V	140	150	124	V	97	107	84
V	139	123	150	M	135	129	134
V	133	129	128	V	139	145	128
M	137	132	134	M	91	86	102
M	99	90	110	V	141	145	131
M	138	136	131	M	85	90	84
M	92	90	98	V	103	96	110
V	89	93	84	M	77	83	72
V	133	114	147	M	130	126	124
M	132	129	124	M	133	126	132
V	141	150	128	V	144	145	137
V	135	129	124	V	103	96	110
M	140	120	147				

³ Ranginiai ir vardiniai kintamieji visuomet diskretūs.

M	96	100	90	V	90	96	86
M	83	71	96	M	83	90	81
M	132	132	120	M	133	129	128
V	100	96	102	V	140	150	124
M	101	112	84	M	88	86	94
V	80	77	86	V	81	90	74
				V	89	91	89

Čia Lytis yra vardinis, o FSIQ (Full Scale IQ), VIQ (Verbal IQ) ir PIQ (Performance IQ) yra skaitiniai kintamieji. Duomenis reiktų sugrupuoti (atskirai vyrams ir moterims), o po to juos palyginti. Pasirodo, kad apskritai (žr. 1.2 pav. žemiau; stačiakampių diagramų paaiškinimus žr. 4.2.3 skyrelį) vyrų IQ yra didesnis, tačiau pagal VIQ, t.y. verbalinį (žodinį) IQ testą, paremtą keturiais Wechslerio (1981) potesčiais, geriau atrodo moterys.



1.2 pav. Moteris ir vyrus lyginame pagal tris intelektualumo koeficiento (IQ) testus; juodo stačiakampio viduje esanti balta linija žymi ėminio “centro” padėtį

Mes susipažinome su įvairiomis matavimo skalėmis. Vaizdžiai kalbant, blogiausia yra vardinė, toliau “gerėjimo” tvarka eina ranginė, skirtuminė ir santykinė skalės.

Vardinių kintamųjų statistinė analizė paprastai apsiriboja narių skaičiaus grupėse (counts) registravimu (kiek yra vyrų ir kiek moterų su IQ didesniu nei 100?) arba jų palyginimu (ar teisybė, kad tokių moterų yra daugiau?).

Nagrinėdami ranginius kintamuosius, “centrą” paprastai nusakome mediana (žr. žemiau). Šiuos kintamuosius galime analizuoti, taikydami visus statistinius metodus, pagrįstus rangais.

Kadangi intervaliniams kintamiesiems yra prasmingos aritmetinės operacijos, tai kartu su mediana, “centrą” galime charakterizuoti ir vidurkiu (žr. žemiau). Galime taip pat skaičiuoti ėminio standartą, Pirsono koreliacijos koeficientą, taikyti Stjudento kriterijų ir daug kitų statistinių metodų (santykiniams ir intervaliniams kintamiesiems taikomos tos pačios statistinės procedūros).

1.1 UŽDUOTIS. 1.1 skyrelyje pateikti keturi duomenų rinkinių pavyzdžiai. Nustatykite visų juose pateiktų kintamųjų tipus.

1.2 UŽDUOTIS. Klasifikuokite šiuos kintamuosius:

- a) Lytis
- b) Pajamos
- c) Ūgis
- d) Rasė
- e) Amžius
- f) Šeimyninė padėtis
- g) Vaikų skaičius šeimoje
- h) Mokinio klasė (mokykloje)
- i) Religija
- j) Svoris
- k) Užimta vieta lenktynėse
- l) Atestato vidurkis
- m) Darbuotojų skaičius įstaigoje
- n) Karinis laipsnis
- o) Kalendoriaus data

1.3. Ėminiai ir jų grafinės charakteristikos

Intuityvus atsitiktinio dydžio (a.d.) apibrėžimas galėtų būti toks: tai skaitinis dydis, kurio reikšmių nei paaiškinti nei prognozuoti (pagal kokio kito dydžio ar jo paties ankstesnes reikšmes) negalime. Pavyzdžiui, jei nagrinėjame 20-25-mečių Lietuvos vyrų ūgį X , tai k -ojo už durų stovinčio vyro ūgis yra a.d. X_k (laikome, kad atsitiktiniai dydžiai X_k yra tarpusavyje nepriklausomi, o visų jų skirstinys yra toks pat kaip ir X ; rinkinys (X_1, \dots, X_n) vadinamas atsitiktiniu ėminiu). Antra vertus, jei šis vyras įėjo vidun, tai jo ūgio matavimo rezultatas yra konkreti šio a.d. realizacija, x_k (n vyrų matavimo rezultatai (x_1, \dots, x_n) vadiname (konkrečiuoju) ėminiu). Norėdami rasti visas a.d. X charakteristikas⁴, turėtume išmatuoti visus Lietuvos vyrus, tačiau mes žinome tik konkretų baigtinį ėminį. Aišku, kad jis tik apytiksliai nusako a.d. X . Jei ėminys nėra didelis, jo reikšmes galėtume tiesiog išvardinti, tačiau jei ėminio dydis yra dešimtys ar šimtai (plg. 1.1 pvz.), šis sąrašas mažai ką sako. Reikia ieškoti kitokių stebimojo atsitiktinio dydžio charakteristikų.

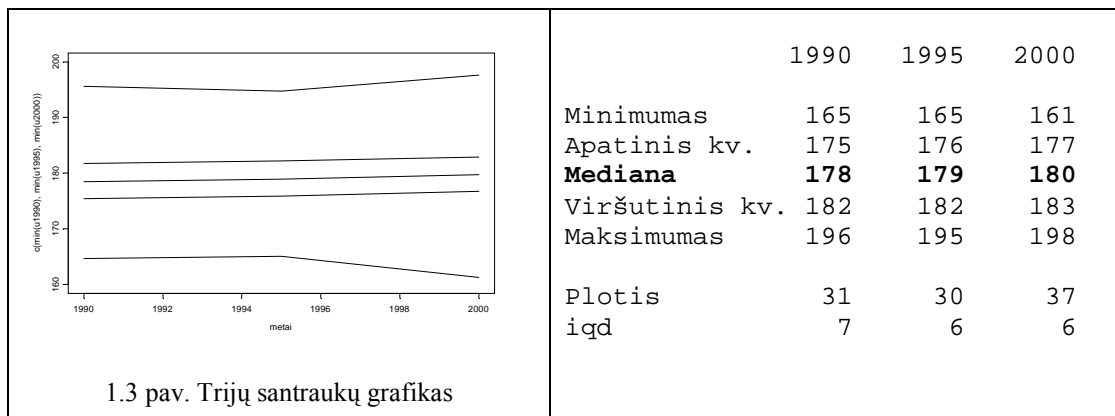
Nagrinėkime vienmatį skaitinį ėminį (x_1, x_2, \dots, x_n) . Perrašę jį didėjimo tvarka, gautume naują objektą, vadinamąją variacinę eilutę: $(x_1^*, x_2^*, \dots, x_n^*)$, $x_1^* \leq x_2^* \leq \dots \leq x_n^*$. Skirtumas tarp didžiausio ėminio nario x_n^* ir mažiausio x_1^* vadinamas ėminio pločiu. Vidurinis variacinės eilutės narys vadinamas mediana (jei ėminys turi tris elementus, tai mediana bus x_2^* , tačiau jei keturis – tai $(x_2^* + x_3^*)/2$). Apatiniu (pirmuoju) kvartiliu Q_1 vadiname visų variacinės eilutės elementų, nedidesnių už medianą, medianą, antruoju (žymėsime Q_2) – pačią medianą, o viršutiniu (trečiuoju) kvartiliu Q_3 – elementų, nemažesnių už medianą, medianą; skaičius $iqd = Q_3 - Q_1$ vadinamas tarp-

⁴ Pavyzdžiui, kokią procentą Lietuvos vyrų sudaro vyrai aukštesni nei 200 cm? Arba – koks vidutinis Lietuvos vyro ūgis?

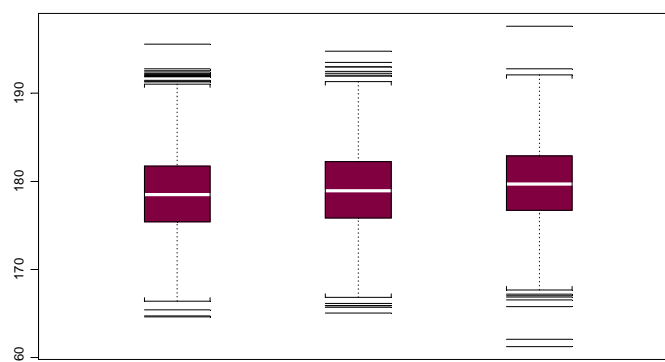
kvartiliniu pločiu (*iqd* - nuo interquartile distance). Štai 1.1 pvz. duomenų variacinės eilutės dalis:

```
[1] 161 162 166 167 167 167 167 168 168 168 169 169 169 169 169 169 170 170 170
[20] 170 170 170 170 170 170 171 171 171 171 171 171 171 171 171 171 171 171 172
.....
[248] 177 177 177 177 177 177 177 177 177 177 177 177 177 177 177 177 177 177 177
[267] 177 177 177 177 177 177 177 177 177 177 177 177 177 177 177 177 177 177 177
.....
[495] 180 180 180 180 180 180 180 180 180 180 180 180 180 180 180 180 180 180 180
[514] 180 180 180 180 180 180 180 180 180 180 180 180 180 180 180 180 180 180 180
.....
[704] 182 182 182 182 182 182 182 182 182 182 182 182 182 182 183 183 183 183 183
[723] 183 183 183 183 183 183 183 183 183 183 183 183 183 183 183 183 183 183 183
.....
[970] 189 189 189 189 189 189 189 189 190 190 190 190 190 190 190 191 191 191 191
[989] 191 191 191 191 191 192 192 192 192 192 193 198
```

Dažnai ėminys charakterizuojamas “penkių skaičių santrauka” – be trijų kvartilų dar pateikiamas ėminio minimumas ir maksimumas. Štai visų trijų tyrimų rezultatai:



Aišku, kad mediana yra tam tikra prasme centrinė, vidutinė ar vidurinė ėminio reikšmė. Ėminio reikšmių išsibarstymą nusako jo plotis arba *iqd* (pastaroji charakteristika – stabilesnė, t.y., vienodesnė skirtingose ėminiuose). Penkių skaičių santrauka gana vaizdžiai charakterizuoja turimą ėminį, ja remiantis jau galima būtų daryti kai kurias išvadas apie 1990-ųjų, 1995-ųjų ir 2000-ųjų metų vyrų populiacijas, tačiau turimus duomenis patogiau vaizduoti stačiakampe diagrama (arba stačiakampiu su žandenoimis (ar ūsais), boxplot, box and whisker plot).



1.4 pav. 1990, 1995 ir 2000 m. vyrų ūgio ėminių stačiakampės diagramos

Tamsaus stačiakampio viduryje esanti linija žymi medianą, tamsaus stačiakampio apačia – apatinį kvartilį Q_1 , o viršus – viršutinį kvartilį Q_3 , prie stačiakampio taškine linija (žandenomis) prijungtos užlenktos atkarpos arba sutampa su ekstremaliomis reikšmėmis arba lygios atitinkamai $Q_1 - 1,5iqd$ ir $Q_3 + 1,5iqd$ (žiūrint kuri arčiau medianos); dar toliau esančios atkarpos žymi reikšmes vadinamas išskirtimis. Jei duomenys turi Gauso skirstinį, maždaug 99,3% duomenų turi būti tarp žandenu.

Jeigu 2000 m. duomenis suskaidytume į keturias grupes, tai jie demonstruotų nemažesnę kintamumą negu aukščiau buvusieji:

Min	166	162	161	168
1-asis kv.	178	176	177	176
Mediana	180	179	180	180
3-asis kv.	183	182	183	183
Max	192	198	192	193

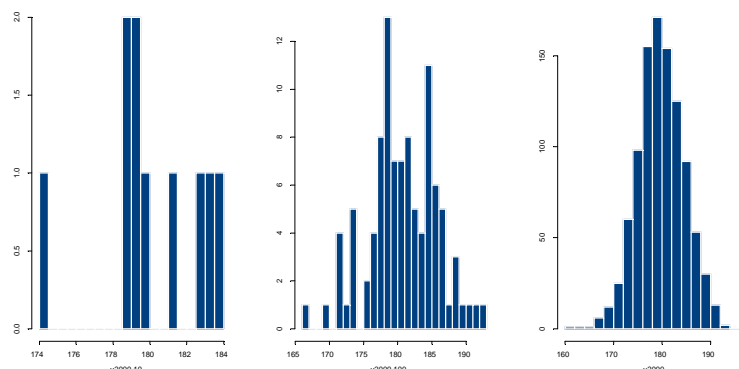
Kadangi šį kartą mes tikimės, kad šie skirtumai neesminiai, tai kuo remdamiesi teigiame, kad ankstesni duomenys įrodo ūgio didėjimo tendenciją? Į šiuos klausimus atsakysime vėliau, tai tipiškas sprendžiamosios statistikos uždavinys.

Svarbi ėminio charakteristika yra įvairių reikšmių dažniai. Jei reikšmių yra daug, geriau iš pradžių duomenis sugrupuoti. Žemiau yra pateikti visų 2000 m. vyrų ūgio dažnių lentelė (duomenys sugrupuoti į 2 cm pločio intervalus, apatinėje eilutėje nurodyta, kelių vyrų iš duomenų rinkinio $n=2000$ ūgis priklauso atitinkamai grupei):

1.4 lentelė.
2000 m. vyrų ūgio dažnio lentelė

160-162	162-164	164-166	166-168	168-170	170-172	172-174	174-176	176-178	178-180	180-182	182-184	184-186	186-188	188-190	190-192	192-194	194-196	196-198
1	1	1	6	12	25	60	98	155	171	154	125	92	53	30	13	2	0	1

Ši lentelė gana didelė, ją lengviau suvokti, pavaizdavus grafiškai – tai vadinamoji histograma. k -jo histogramos stulpelio aukštis yra lygus grupės narių skaičiui n_k (tai vadinamieji *dažniai*, jų suma lygi ėminio dydžiui n) arba santykiui n_k/n (tai vadinamieji *santykiniai dažniai*, jų suma visada lygi vienetui).

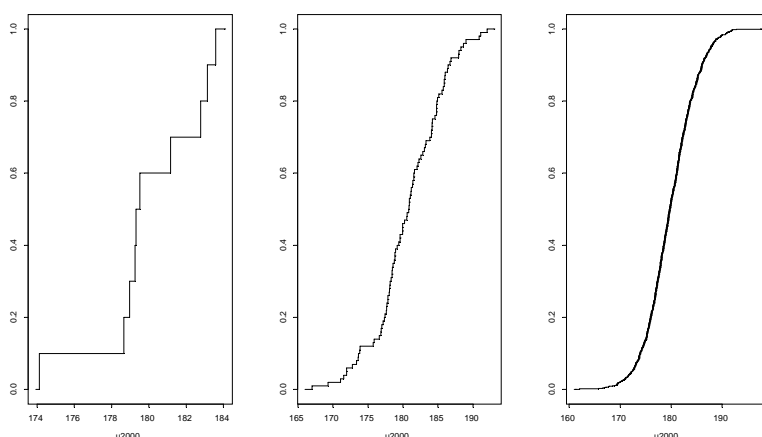


1.5 pav. 2000 m. ūgio reikšmių histogramos:
kairėje – pirmųjų dešimties, viduryje – pirmojo šimto, dešinėje – visų vyrų
(didėjant ėminio dydžiui histograma darosi vis reguliarsnė)

Iš histogramos matyti, kokios ėminio reikšmės yra dažniausiai sutinkamos⁵ (arba - labiausiai tikėtinos), reikšmių ribos, simetriškumas ir pan.

Tikslesnė (nes ėminio reikšmės negrupuojamos), bet ne tokia vaizdi, yra sukaupųjų santykinų dažnių kreivė (kitaip: empirinė (reikšmių) skirstinio funkcija F_n): kiekvienam x ši funkcija lygi mažesnių už x ėminio reikšmių skaičiui, padalintam iš n . Ši funkcija yra pastovi tarp gretimų variacinės eilutės reikšmių, o taške x_k apibrėžiama taip: jei reikšmė x_k ėminyje sutinkama n_k kartų, tai F_n šiame taške daro dydžio n_k/n šuolį aukštyn.

1.3 UŽDUOTIS. 1.6 pav. dešiniame grafike yra išbrėžta visų 2000 m. vyrų ūgio empirinė skirstinio funkcija. Remdamiesi šių duomenų variacine eilute, raskite F_{1000} reikšmes taškuose 170, 171, 198 ir 200.



1.6 pav. 2000 m. vyrų ūgio sukaupųjų santykinų dažnių kreivė: kairėje – pirmųjų dešimties, viduryje – pirmojo šimto, dešinėje – visų vyrų (didėjant ėminio dydžiui funkcija F_n darosi vis reguliaresnė)

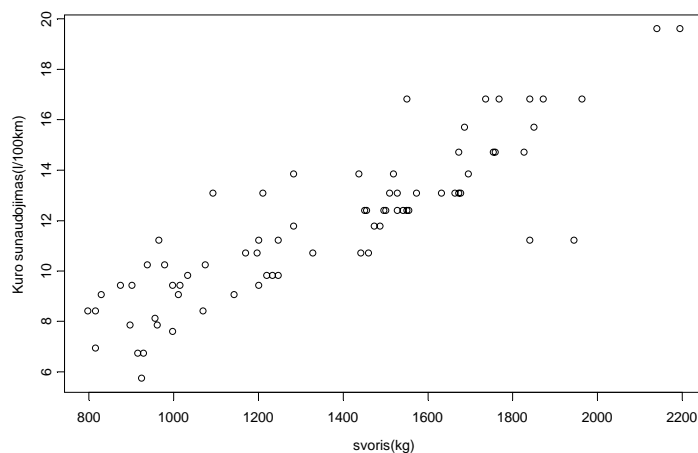
Nauji uždaviniai atsiranda, nagrinėjant **dvimačius** ėminius. Dabar labai dažnai mums rūpi ne tik kiekvienos komponentės individualios reikšmės, bet ir jų ryšiai. Žemiau pateikti 1979 m. surinktų duomenų masyvas `auto`, kuriame pateikti duomenys apie 74 automobilių modelių tris charakteristikas – kuro sunaudojimą (`sun`, l/100km), svorį (`svo`, kg) ir cilindrų tūrį (`tur`, cm³). Vėl susiduriame su būdinga padėtimi – duomenų daug, todėl susidaryti išpūdį apie jų tarpusavio priklausomybę yra sunku.

	sun	svo	tur			
				Linc. Cont. Mark V	19	2140 6554
				Linc. Versailles	16	1737 4948
				Mazda Glc	7	898 1409
Amc Concord	10	1329	1982	Merc Bobcat	10	1170 2294
Amc Pacer	13	1519	4227	Merc Cougar	16	1841 4948
Amc Spirit	10	1197	1982	Merc Cougar XR-7	16	1873 4948
Audi 5000	13	1283	2146	Merc Marquis	15	1687 4948
Audi Fox	10	938	1589	Merc Monarch	13	1528 4096
BMW 320i	9	1202	1982	Merc Zephyr	11	1283 2294
Buick Century	11	1474	3211	Olds 98	11	1841 5735
Buick Electra	15	1850	5735	Olds Cutlass	12	1496 3785
Buick Le Sabre	13	1664	3785	Olds Cutl. Supr.	12	1501 3785
Buick Opel	9	1011	1818	Olds Delta 88	13	1673 3785
Buick Regal	11	1487	3211	Olds Omega	12	1528 3785
Buick Riviera	14	1759	3785			

⁵ Aukščiausio stulpelio vidurys vadinamas ėminio moda – 5 pav. kairiajame brėžinyje moda lygi $(178,5+179,5)/2=179$, centriniame – $(178+179)/2=178,5$, dešiniajame – $(178+180)/2=179$.

Buick Skylark	12	1542	3785	Olds Starfire	9	1233	2474
Cad. Deville	16	1964	6964	Olds Toronado	14	1827	5735
Cad. Eldorado	16	1769	5735	Peugot 604 Sl	16	1551	2671
Cad. Seville	11	1945	5735	Plym Arrow	8	1070	2556
Chev Chevette	8	957	1605	Plym Champ	6	816	1409
Chev Impala	14	1673	4096	Plym Horizon	9	997	1720
Chev Malibu	10	1442	3277	Plym Sapporo	9	1143	1950
Chev Monte Carlo	10	1460	3277	Plym Volare	13	1510	3687
Chev Monza	9	1247	2474	Pont Catalina	13	1678	3785
Chev Nova	12	1555	4096	Pont Firebird	13	1573	3785
Datsun 200-SX	10	1075	1950	Pont Grand Prix	12	1456	3785
Datsun 210	6	916	1392	Pont Le Mans	12	1451	3785
Datsun 510	9	1034	1950	Pont Phoenix	12	1551	3785
Datsun 810	11	1247	2392	Pont Sunbird	9	1220	2474
Dodge Colt	7	961	1605	Renault Le Car	9	830	1294
Dodge Diplomat	13	1632	5211	Subaru	6	929	1589
Dodge Magnum XE	14	1755	5211	Toyota Celica	13	1093	2195
Dodge St. Regis	13	1696	3687	Toyota Corolla	7	997	1589
Fiat Strada	11	966	1720	Toyota Corona	13	1211	2195
Ford Fiesta	8	816	1605	Volk Rabbit	9	875	1458
Ford Mustang	11	1202	2294	Volk Rabbit(d)	5	925	1474
Honda Accord	9	1016	1753	Volk Scirocco	9	902	1589
Honda Civic	8	798	1491	Volks Dasher	10	979	1589
Linc. Continental	19	2195	6554	Volvo 260	13	1437	2671

Dvimačių ėminių tyrimas paprastai pradedamas nuo jų sklaidos diagramos brėžimo. Duomenų rinkinio (*svo* , *sun*) atveju kiekvienas taškas atitinka automobilio modelį, taško abscisė rodo jo svorį, o ordinatė – kuro sunaudojimą. Sklaidos diagrama aiškiai demonstruoja tiesinį ryšį tarp dydžių *svo* ir *sun*.



1.7 pav. Kintamųjų *svo* ir *sun* sklaidos diagrama

Pabrėšime, kad mes netvirtiname, jog žinant svorį galima vienareikšmiškai prognozuoti kuro sunaudojimą (*sun* yra atsitiktinis dydis, gerai matyti, kad (maždaug) tai pačiai *svo* reikšmei *sun* reikšmės gana pastebimai skiriasi). Antra vertus, tendencija yra aiški: didėjant *svo* reikšmėms, atsitiktinio dydžio *sun* vidutinė reikšmė didėja. Būtent ją ir prognozuoja vadinamieji regresiniai modeliai.

1.4. Ėminiai ir jų skaitinės charakteristikos

Skaitinio ėminio grafinės charakteristikos yra labai naudingos, tačiau pasirodo, kad gana dažnai ėminių galima sėkmingai charakterizuoti dar paprasčiau, keliais skaičiais. Kelias skaitines charakteristikas jau žinome – tai penkių skaičių santrauka, dažnių lentelė. Šiame skyrelyje pakalbėsime apie momentus.

Vieną ėminio “centro” charakteristiką jau žinome – tai mediana. Kita, dar populiare-
nė ėminio (x_1, x_2, \dots, x_n) charakteristika, yra jo (empirinis) vidurkis \bar{x} : tai ėminio
reikšmių aritmetinis vidurkis

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i .$$

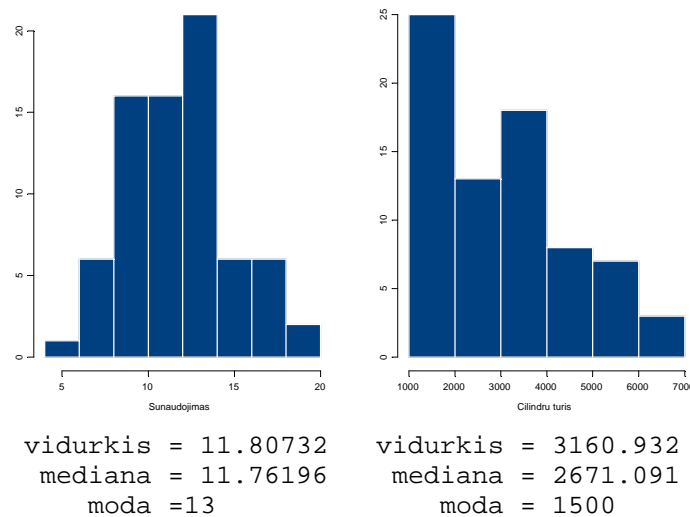
Jei ėminys yra pateiktas dažnių lentelė

x_1	x_2	...	x_N
n_1	n_2	...	n_N

(t.y., reikšmė x_1 ėminyje kartojasi n_1 kartą, reikšmė x_2 – n_2 kartus ir t.t.), vidurkį gali-
ma perrašyti kitokiu pavidalu:

$$\bar{x} = \frac{1}{n} \sum_{k=1}^N n_k x_k ;$$

čia N yra skirtingų ėminio reikšmių skaičius. Jei ėminio histograma yra maždaug si-
metrinė, tai mediana ir vidurkis yra beveik vienodi, tačiau tačiau priešingu atveju jie
gali pastebimai skirtis.



1.8 pav. auto duomenų histogramos: sun (kairėje) ir tur (dešinėje)

Pažymėsime, kad viena (vienintelė!) nenormaliai didelė ėminio reikšmė (išskirtis) gali
pastebimai pakeisti vidurkį, tuo tarpu mediana⁶ išskirtims mažiau jautri.

Nagrinėkime dvi variacines eilutes: (99.99, 99.99, 100.01, 100.01) ir (0, 0, 200, 200).
Aišku, kad vidurkių prasme jos abi vienodos (abiejų vidurkiai lygūs 100), tačiau jų
struktūra visai skirtinga: pirmuoju atveju matuojame praktiškai nekintantį dydį, o ant-
rosios eilutės reikšmių išsibarstymas vidurkio atžvilgiu yra didžiulis. Ėminio reikšmių
išsibarstymo matu galėtų būti ėminio reikšmių vidutinis nuotolis nuo vidurkio, t.y.

⁶ Vardiniams kintamiesiems vienintelė prasminga “centro” reikšmė yra moda, ranginiams – moda ir
mediana (nes ranginiams kintamiesiems apibrėžta sąvoka “daugiau”), o skaitiniams – moda, mediana ir
vidurkis.

$$\frac{1}{n} \sum_{i=1}^n |x_i - \bar{x}|,$$

tačiau populiarnesnis yra ėminio vidutinis kvadratinis nuokrypis

$$s_1^2 = \frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})^2.$$

Pažymėsime, kad dėl tam tikrų priežasčių paprastai vartojamas kiek “pataisytas” vidutinis kvadratinis nuokrypis: skaičius

$$s^2 = \frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})^2$$

vadinamas ėminio (empirine) dispersija⁷. Beje, jei x_i matuojami, pvz., centimetrais (cm), tai s^2 dimensija bus cm^2 . Dydisio $s = \sqrt{s^2}$ (jis vadinamas ėminio standartiniu nuokrypiu arba tiesiog standartu) dimensija jau bus cm , t.y., lygiai tokia pati kaip ir x_i , todėl būtent jis, standartas, ir yra populiariausia reikšmių išsibarstymo charakteristika.

Nesunku apskaičiuoti, kad pirmos iš anksčiau minėtų variacinių eilučių vidutinis kvadratinis nuokrypis lygūs 0.01^2 (nes $\frac{1}{4}((99.99 - 100)^2 + \dots) = \frac{1}{4}(0.01^2 + \dots) = 0.01^2$), o antros - 100^2 !

Vidurkio ir vidutinio kvadratinio nuokrypio savokas galima apibendrinti: skaičius

$$a_k = \frac{1}{n} \sum_{i=1}^n x_i^k, k \in N,$$

vadinamas k -ju (pradiniu⁸) ėminio momentu, o skaičius

$$m_k = \frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})^k, k \in N, -$$

k – ju centriniu⁹ momentu. Kitaip sakant, vidurkis yra pirmasis momentas, o vidutinis kvadratinis nuokrypis – antrasis centrinis momentas. Iš principo, kiekvienam ėminiui galime apskaičiuoti be galo daug momentų, tačiau minėti du yra svarbiausi: vidurkis yra ėminio “centro”, o standartas – ėminio reikšmių išsibarstymo charakteristikos.

⁷ Būkite atsargūs – kartais (empirine) dispersija vadinamas dydis s_1^2 !

⁸ Nes “pradžios”, t.y. nulio atžvilgiu.

⁹ Nes “centro”, t.y. vidurkio atžvilgiu.

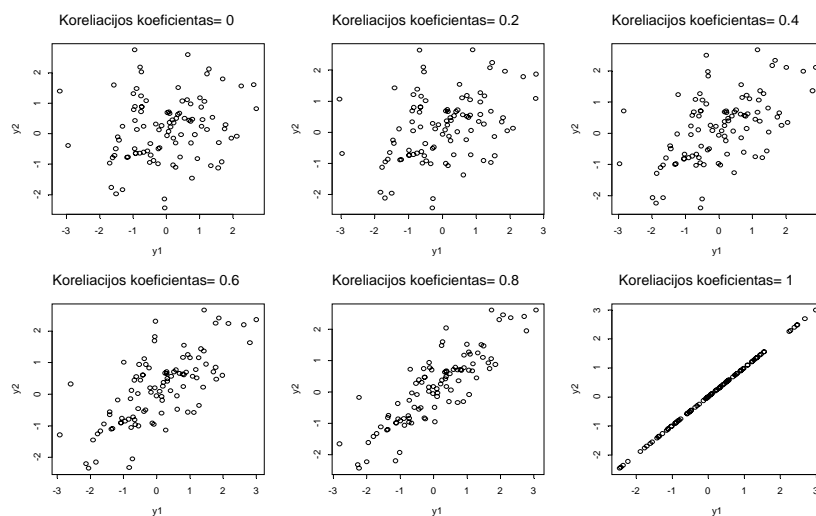
Jei iš kiekvieno ėminio (x_1, x_2, \dots, x_n) nario atimsime vidurkį, tai naujasis ėminys $(x_1 - \bar{x}, x_2 - \bar{x}, \dots, x_n - \bar{x})$ vadinamas centruotu. Jei centruoto ėminio kiekvieną narį dar padalinsime iš standarto, tai naujasis ėminys $((x_1 - \bar{x})/s, (x_2 - \bar{x})/s, \dots, (x_n - \bar{x})/s)$ bus vadinamas (centruotu ir) normuotu. Histogramos forma nuo šių transformacijų nesikeičia, tačiau normuoto ėminio vidurkis visuomet 0, o standartas – 1.

1.4 UŽDUOTIS. Patikrinkite pastarąjį teiginį.

Jei tiriamasis ėminys $((x_1, y_1), \dots, (x_n, y_n))$ yra dvimatis skaitinis, tai kiekvieną komponentę vėl galima charakterizuoti jos vidurkiu ir standartu. Antra vertus, dabar yra dar viena, komponentių ryšį nusakanti, skaitinė charakteristika – tai vadinamasis normuotų komponentių mišrusis momentas arba Pirsono (Pearson) empirinis koreliacijos koeficientas r :

$$r = \frac{1}{n} \sum_{i=1}^n \left(\frac{x_i - \bar{x}}{s_x} \right) \left(\frac{y_i - \bar{y}}{s_y} \right).$$

Galima įrodyti, kad visuomet $-1 \leq r \leq 1$. Jei r neigiamas, tai x 'sui didėjant, y , apskri-tai kalbant, mažėja, o jei teigiamas, tai x 'sui didėjant y irgi didėja. Gruboka taisyklė tokia: jei $|r| > 0.7$, tai x ir y ryšys pakankamai stiprus.

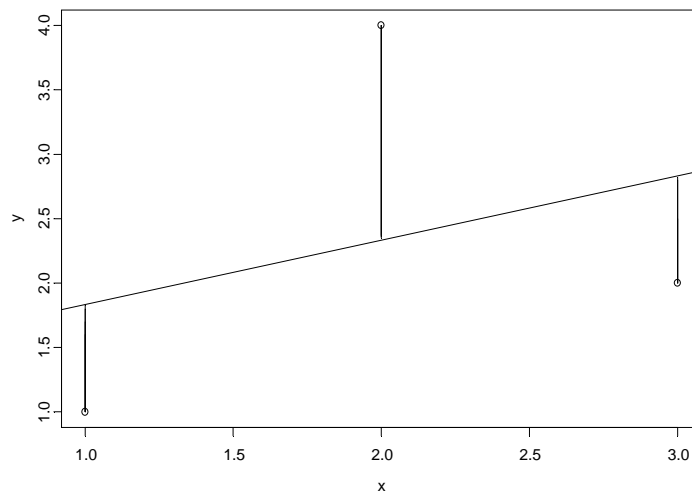


1.9 pav. Koreliacijos koeficiento r įtaka sklaidos diagramos formai

Kita svarbi dvimačio skaitinio duomenų masyvo charakteristika yra (y 'ko) regresijos (x 'so atžvilgiu) tiesė: tai “arčiausiai visų sklaidos diagramos taškų esanti” tiesė $y = b_0 + b_1 x$. Tiksliau kalbant, iš visų galimų tiesių $y = \beta_0 + \beta_1 x$ pasirinksime tokią, kuriai jos atitinkamų taškų atstumų nuo sklaidos diagramos taškų kvadratų suma yra mažiausia, t.y. ieškosime funkcijos

$$RSS = RSS(\beta_0, \beta_1) = \sum_{i=1}^n (y_i - (\beta_0 + \beta_1 x_i))^2 = \sum_{i=1}^n e_i^2$$

(RSS – nuo Residual Sum of Squares (angl. liekanų kvadratų suma)) minimumo pagal



1.10 pav. Trys stebėjimų taškai ir regresijos tiesė

β_0 ir β_1 . Prilygindami RSS dalines išvestines pagal β_0 ir β_1 nuliui, gauname dviejų lygčių sistemą; jos sprendiniai (patikrinkite) yra

$$b_1 = \hat{\beta}_1 = \frac{rs_y}{s_x}, \quad b_0 = \hat{\beta}_0 = \bar{y} - b_1\bar{x}$$

Nesunku apskaičiuoti, kad duomenų sistemos (t_{ur}, s_{un}) atveju $r=0,8179$, $s_{tur}=1509,03$, $s_{sun}=3,01$, $b_0=6,6377$, o $b_1=0,0016$ (šiais laikais tokie skaičiavimai paprastai atliekami su kompiuteriu), kitais žodžiais regresijos tiesės lygtis atrodo taip:

$$sun=6,6377+0,0016tur.$$

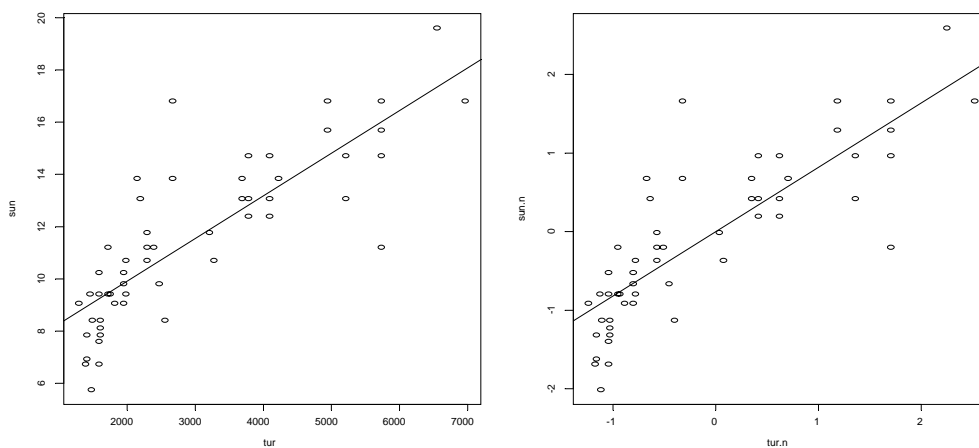
Pastebėsime, kad nors koeficientas 0,0016 labai mažas, tai dar nereiškia, kad kuro sunaudojimas sun beveik nepriklauso nuo cilindrų tūrio tur . Atvirkščiai, koreliacijos koeficiento reikšmė signalizuoja, kad ryšys pakankamai stiprus. Iš tikrųjų koeficientų b_0 ir b_1 reikšmės labai priklauso nuo matavimo vienetų: jei cilindrų tūrį matuotume litrais, tai regresijos lygtis būtų

$$sun=11,8073+2,4680 \text{ tur} . l ,$$

o jei pereitume prie normuotų sun ir tur reikšmių, tai (kodėl?)

$$sun . n=0+r \text{ tur} . n=0,8179 \text{ tur} . n .$$

Pažymėsime, kad sklaidos diagramos ir regresijos tiesių grafikai visais trim atvejais atrodytų “vienodai”, nes kompiuteris braižydamas grafikus paprastai pasirenką “teisingus” ašių mastelius (plg. žemiau pateiktą 1.11 pav.).



1.11 pav. Sklaidos diagramos ir regresijos tiesės:
 kairėje - (tur, sun), o dešinėje – (tur.n, sun.n) sistema
 (atkreipkite dėmesį į skaičius prie ašių)

Mes glaustai išdėstėme aprašomosios statistikos pagrindus. Norėdami pereiti prie sprendžiamosios statistikos, pirmiausiai turėtume daugiau sužinoti apie tikimybių teoriją. Tuo užsiims kiti kursai. O dabar pereisime prie nuoseklesnio mūsų kurso – įvado į statistiką – dėstymo.

2. R – bendrieji faktai

2.1. R instaliacija

Svetainėje <http://cran.hu.r-project.org/> nuvairuokite į Precompiled Binary Distributions\Windows (95 and later)\base ir atsisiųskite (download) iš ten failą `rw1071.exe` (maždaug 20 megabaitų). Patalpinkite jį bet kur, o po to paleiskite šią programą – instaliaciją ši programa atliks (beveik) automatiškai. Pažymėsime, kad šis failas taip pat yra kompaktiniame diske R1, instaliaciją galima atlikti ir spragtelėjus ant jo vardo.

R standartinė instaliacija sukuria darbinę (working) direktoriją `C:\Program Files\R\rw10711`, kurios podirektorijoje `...\.bin` yra failas `Rgui.exe`. R paleisti galima, spragtelėjus ant šio failo, arba, “ištraukus” jį į desktopą (darbalaukį) ir sukūrus R ikoną (pavadinkime ją R 1.7.1). Jei ateityje šiuo produktu naudosis vienas vartotojas, kuris dirbs tik su vienu projektu, tai tuo instaliaciją galima ir baigti. Jei vartotojų ar projektų bus keli, tai kiekvienam galima sukurti savąją ikoną: sukurkite naują direktoriją, pvz., `C:\aR2`; joje sukurkite podirektorijas `Mano1`, `Mano2` ir t.t. Dešiniuojų pelės klavišu spragtelėkite ant R 1.7.1 ikonos ir pasirinkite Copy; po to bet kur desktape spragtelėkite dešiniuojų klavišu ir pasirinkite Paste. Spragtelėję ant pasirodžiusios ikonos kopijos dešiniuojų klavišu, pasirinkite Properties\Shortcut; Target eilutėje įrašykite `"C:\Program Files\R\rw1071\bin\Rgui.exe"`, o Start in eilutėje - `"C:\aR\Mano1"`, OK, o po to šią ikoną pervardinkite į, pvz., `R_Mano1`; aprašytą procedūrą pakartokite su `Mano2` ir t.t. Ateityje savąjį projektą bus galima paleisti, spragtelėjus ant reikalingos ikonos.

2.2. R ekranas

Spragtelėkite ant reikalingos R ikonos (2.1 pav. yra vaizdas, kurį matote ekrane). Šis (beveik) tuščias ekranas ateityje bus užpildytas komandomis, kurias rašysime į dešinę nuo kreipinio `>`, o taip pat grafikais, kurie atsiras grafiniame lange. Štai paprasčiausia komanda³:

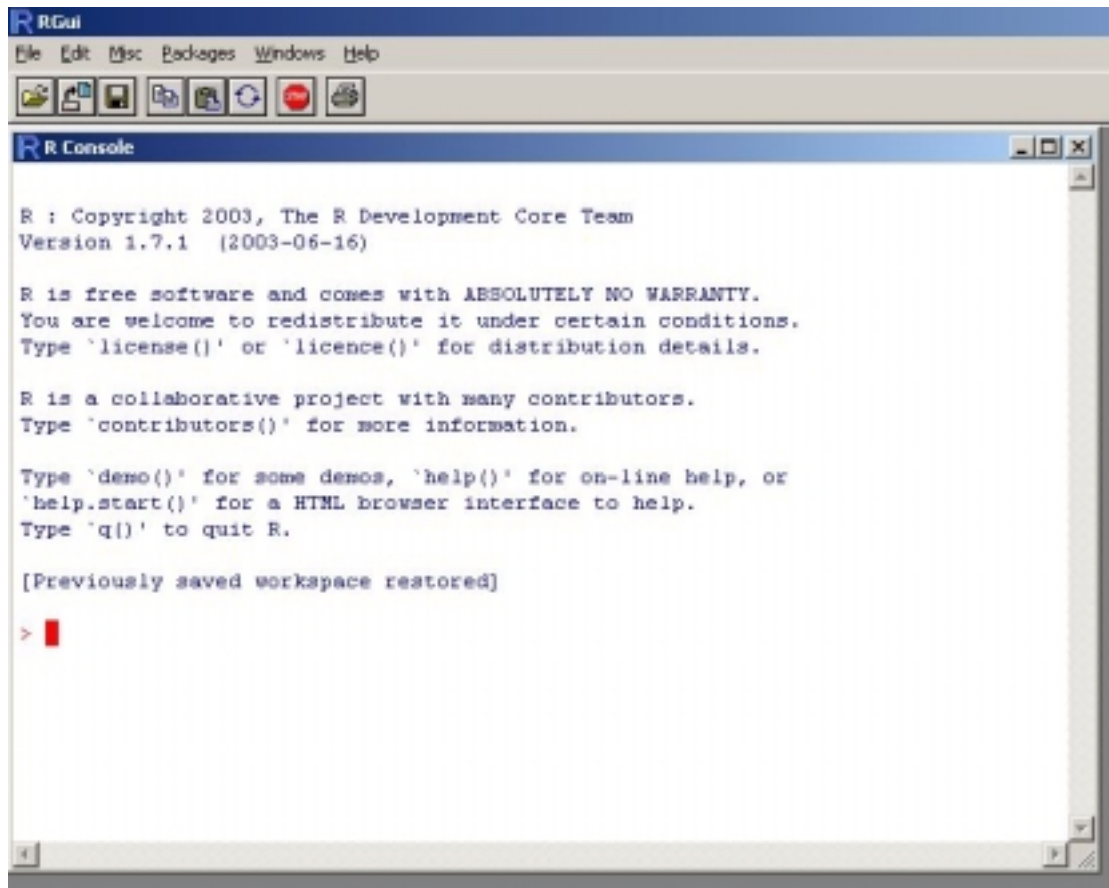
```
> y <- 1:5
```

(simbolį `<-` gauname paspaudę klavišą `<` (mažiau) ir po to minuso (`-`) klavišą; jei `a` ir `b` yra sveiki skaičiai, tai binarios operacijos `a:b` reikšmė yra seka `a, a+1, ..., b`), kuri sukurs skaitinį vektorių `y` lygų 1, 2, 3, 4, 5. Ši komanda yra ekvivalenti komandoms

¹ Atsiradus naujesnei R versijai, atsisiųskite ją iš CRAN'o ir instaliuokite (tai sukurs naują direktoriją, tarkime `...\.rw1080`); jei norite išsaugoti senus duomenis ir anksčiau sukurtas funkcijas – perkeltite senuosius failus `.Rdata` ir `.Rhistory` (ir, gal būt, dar kitus reikalingus failus, pvz., duomenų failus `*.txt`) į šią direktoriją, o senąją direktoriją (tarkime, `...\.rw1071`) tiesiog ištrinkite (su `unins000.exe` iš `C:\Program Files\R\rw1071`).

² Komputerinėse klasėse kiekvienas studentas turi savo asmeninę direktoriją diske U. Sukurkite joje direktoriją `aR`, o joje dvi podirektorijas – `IntroStat` ir `Ataskaitos`. Pirmoji podirektorija gali tarnauti kaip darbinė (t.y., kaip `Mano1` ekvivalentas), o antroje galima kaupti užduočių ataskaitas.

³ Toliau šiame konspekte kreipinį `>` dažnai praleisime.



2.1 pav. R komandinio lango (konsolės) vaizdas

```
> y_1:5
```

(simbolis `_` yra⁴ simbolio `<-` sinonimas; ilgose programose geriau rašyti `<-`, nes tuomet tekstas aiškesnis), arba komandai

```
y <- seq(1,5)
```

(funkcija `seq(a,b)` yra `a:b` sinonimas; komanda `seq(1,50,7)` generuos vektorių 1, 8, 15,...,50), arba komandai

```
> y <- c(1,2,3,4,5)
```

(funkcija `c` (nuo *combine* (angl.) = apjungti) apjungia skliaustuose nurodytus skaičius į vektorių). Pasižiūrėkime į sukurtą objektą `y`:

```
y <- 1:5
y
[1] 1 2 3 4 5 # Komanda > (y <- 1:5) apjungia dvi komandas:
# > y <- 1:5 ir > y
```

Dvi žaliai nuspalvintas komandas (eilutes) R komandiniame lange galite surinkti rankomis, arba, jei skaitote kompiuterinę šio teksto versiją, perkelti į šį langą su Copy ir Paste komandomis. Paspaudę Enter klavišą, ekrane pamatysite

⁴ Greičiau “buvo” – simbolių “_” (nuo R 1.8.0 versijos) žadama uždrausti.

```
[1] 1 2 3 4 5
```

Pažymėsimė, kad R vartoja vektorinę aritmetiką, t.y. operacijos

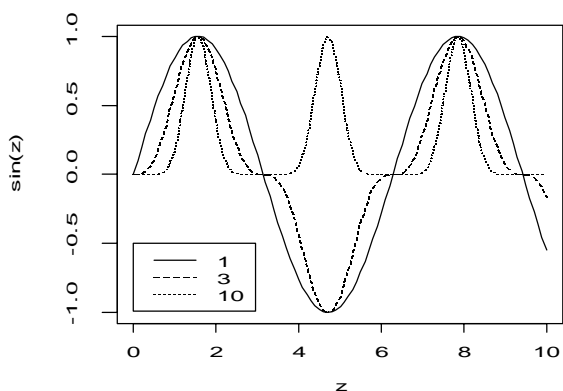
```
> (1:5)^3
[1] 1 8 27 64 125
```

rezultatas yra vektorius su koordinatėmis $1^3, 2^3, 3^3, 4^3, 5^3$. Štai dar vienas pavyzdys:

```
> z<-(0:100)/10
> z
 [1] 0.0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1.0 1.1
[13] 1.2 1.3 1.4 1.5 1.6 1.7 1.8 1.9 2.0 2.1 2.2 2.3
.....
[85] 8.4 8.5 8.6 8.7 8.8 8.9 9.0 9.1 9.2 9.3 9.4 9.5
[97] 9.6 9.7 9.8 9.9 10.0
> round(sin(z),4)
 [1] 0.0000 0.0998 0.1987 0.2955 0.3894 0.4794 0.5646 0.6442
 [9] 0.7174 0.7833 0.8415 0.8912 0.9320 0.9636 0.9854 0.9975
.....
[89] 0.5849 0.5010 0.4121 0.3191 0.2229 0.1245 0.0248 -0.0752
[97] -0.1743 -0.2718 -0.3665 -0.4575 -0.5440
```

O dabar išbrėškime pirmąjį grafiką:

```
plot(z,sin(z),type="l") # z yra x koordinatė, o sin(z)- y koordinatė;
# "l" nurodo, kad taškus reikia sujungti
# linijomis
lines(z,(sin(z))^3,lty=2) # Funkcija "lines" papildo pirmą grafiką
# linijomis; lty=2 nurodo linijos tipą
lines(z,(sin(z))^10,lty=3)
legend(0,-0.5,c("1","3","10"),lty=c(1,2,3)) # Legendos viršutinis
# kairysis kampas bus
# taške (0;0,5)
```



2.2 pav. Funkcijų $y = \sin z$, $y = \sin^3 z$ ir $y = \sin^{10} z$ grafikai

Jei tekstą iš ekrano su Copy+Paste nesunku perkelti į Word'o dokumentą, tai su grafika yra sudėtingiau: suaktyvinę grafikos langą, spragtelėkite dešiniu klavišu ir pasirinkite Copy as metafile; grįžkite į Word'o dokumentą ir pasirinkite Edit|Paste Special...|Picture (Enhanced Metafile)|OK.

Grįžkime prie R ekrano nagrinėjimo. Meniu eilutėje spragtelėkite ant Help skyriaus.

1) Pasirinkę FAQ on R, matome DPK (=dažniausiai pateikiamus klausimus) apie R ir atsakymus į juos. Štai šio dokumento pradžia:

R FAQ

R FAQ

Frequently Asked Questions on R

Version 1.7-18, 2003-06-13

ISBN 3-901167-51-X

Kurt Hornik

- [Introduction:](#)
- [R Basics:](#)
- [R and S:](#)
- [R Web Interfaces:](#)
- [R Add-On Packages:](#)
- [R and Emacs:](#)
- [R Miscellanea:](#)
- [R Programming:](#)
- [R Bugs:](#)
- [Acknowledgments:](#)

Štai vienas šio dokumento naudojimo pavyzdžių: įjungus R, pakraunamos tik pagrindinės bibliotekos (package). Dauguma specializuotų funkcijų yra kitose R bibliotekose. Spragtelėję ant R Add-On Packages, pamatysime visų (šios R versijos) bibliotekų sąrašą:

5 R Add-On Packages

- [Which add-on packages exist for R?:](#)
- [How can add-on packages be installed?:](#)
- [How can add-on packages be used?:](#)
- [How can add-on packages be removed?:](#)
- [How can I create an R package?:](#)
- [How can I contribute to R?:](#)

5.1 Which add-on packages exist for R?

- [Add-on packages in R:](#)
- [Add-on packages from CRAN:](#)
- [Add-on packages from Omegahat:](#)
- [Add-on packages from BioConductor:](#)
- [Other add-on packages:](#)

5.1.1 Add-on packages in R

The R distribution comes with the following extra packages:

ctest

A collection of Classical TESTs, including the Ansari-Bradley, Bartlett, chi-squared, Fisher, Kruskal-Wallis, Kolmogorov-Smirnov, t, and Wilcoxon tests.

eda

Exploratory Data Analysis. Currently only contains functions for robust line fitting, and median polish and smoothing.

lqs

Resistant regression and covariance estimation.

methods

Formally defined methods and classes for R objects, plus other programming tools, as described in the Green Book.

modreg

MODern REGression: smoothing and local methods.

mva

MultiVariate Analysis. Currently contains code for principal components, canonical correlations, metric multidimensional scaling, factor analysis, and hierarchical and k-means clustering.

.....

2) Pasirinkę FAQ on R for Windows, pamatysime html tipo dokumentą, kuriame rasime (Windows aplinkoje dirbančiam) R vartotojui svarbią informaciją. Štai jos pradžia:

R for Windows FAQ

Version for rw1071

B. D. Ripley

Table of Contents

- [1 Introduction](#)
- [2 Installation and Usage](#)
 - [2.1 Where can I find the latest version?](#)
 - [2.2 How do I install R for Windows?](#)

- [2.3 Can I customize the installation?](#)
- [2.4 How do I run it?](#)
- [2.5 How do I UNinstall R?](#)
- [2.6 What's the best way to upgrade?](#)
- [2.7 There seems to be a limit on the memory it uses!](#)
- [2.8 How can I keep workspaces for different projects in different directories?](#)
- [2.9 How do I print from R?](#)
- [2.10 Can I use R BATCH?](#)
- [2.11 Can I use rw1071 with ESS and \(X\)emacs?](#)
- [2.12 What are HOME and working directories?](#)
- [2.13 How do I set environment variables?](#)
- [2.14 R can't find my file, but I know it is there!](#)
- [2.15 Does R use the Registry?](#)
- [2.16 Does R support automation \(OLE, COM\)?](#)
- [2.17 The internet download functions fail.](#)
- [2.18 Entering certain characters crashes Rgui.](#)
- [2.19 Other strange crashes.](#)
- [3 Packages](#)
 - [3.1 Can I install packages \(libraries\) in this version?](#)
 - [3.2 I don't have permission to write to the rw1071\library directory.](#)
 - [3.3 The packages I installed do not appear in the HTML help system.](#)
 - [3.4 My functions are not found by the HTML help search system.](#)
 - [3.5 Loading a package fails.](#)
 - [3.6 Package TclTk does not work.](#)
 - [3.7 Hyperlinks in Compiled HTML sometimes do not work.](#)
 - [3.8 update.packages\(\) fails](#)
- [4 Windows Features](#)
 - [4.1 What should I expect to behave differently from the Unix version of R?](#)
 - [4.2 I hear about some nifty features: please tell me about them!](#)
 - [4.3 Circles appear as ovals on screen](#)
 - [4.4 How do I move focus to a graphics window or the console?](#)
- [5 Workspaces](#)
 - [5.1 My workspace gets saved in a strange place: how do I stop this?](#)
 - [5.2 How do I store my workspace in a different place?](#)
 - [5.3 Can I load workspaces saved under Unix/GNU-Linux or MacOS?](#)
- [6 The R Console and Fonts](#)
 - [6.1 I would like to be able to use Japanese fonts](#)
 - [6.2 I don't see characters with accents at the R console, for example in ?text.](#)
 - [6.3 When using Rgui the output to the console seems to be delayed.](#)
 - [6.4 Long lines in the console or pager are truncated.](#)
- [7 Building from Source](#)
 - [7.1 How can I compile R from source?](#)
 - [7.2 Can I use a fast BLAS?](#)

- o [7.3 How do I include compiled C code?](#)
- o [7.4 How do I debug code that I have compiled and dyn.load-ed?](#)
- o [7.5 How do I include C++ code?](#)
- o [7.6 The output from my C code disappears. Why?](#)
- o [7.7 The output from my Fortran code disappears. Why?](#)
- o [7.8 The console freezes when my compiled code is running.](#)

3) Pasirinkę R functions (text)... ir langelyje Help on surinkę, pvz., mean|OK, pamatysime anglišką funkcijos mean aprašymą⁵:

```

mean                                package:base                        R Documentation

Arithmetic Mean

Description:

    Generic function for the (trimmed) arithmetic mean.

Usage:

    mean(x, ...)
    mean.default(x, trim = 0, na.rm = FALSE)

Arguments:

    x: a numeric vector containing the values whose mean is to be
        computed. A complex vector is allowed for `trim=0`, only.

    trim: the fraction (0 to 0.5) of observations to be trimmed from
        each end of `x' before the mean is computed.

    na.rm: a logical value indicating whether `NA' values should be
        stripped before the computation proceeds.

Value:

    If `trim' is zero (the default), the arithmetic mean of the val-
    ues
    in `x' is computed.

    If `trim' is non-zero, a symmetrically trimmed mean is computed
    with a fraction of `trim' observations deleted from each end
    before the mean is computed.

See Also:

    `weighted.mean'

Examples:

    x <- c(0:10, 50)
    xm <- mean(x)
    c(xm, mean(x, trim = .10))
    all.equal(mean(x, trim = 0.5), median(x))

```

Jei norite pamatyti funkcijos mean tekstą, surinkite

```
> mean
```

⁵ Tokią patį rezultatą gausite, jei ekrano komandiniame lange (konsolėje) surinksite help("mean") arba ?mean.

```
function (x, ...)
UseMethod("mean")
```

R yra objektiškai orientuota kalba, kas reiškia, kad, pvz., funkcija `mean` pirmiausiai patikrina savo argumento (objekto) klasę, o jau paskui taiko jam tinkamą metodą.

```
> methods(mean)
[1] "mean.data.frame" "mean.default" "mean.POSIXct" "mean.POSIXlt"
```

Matome, kad yra keturi skirtingi funkcijos `mean` taikymo variantai. Tai reiškia, kad jei tiriamasis objektas nėra duomenų sistema (`data.frame`) arba jis nepriklauso POSIX klasei (tai datų ir laiko objektai), tai funkcija `mean` kreipsis į standartinę `mean.default` funkciją. Štai jos tekstas:

```
> mean.default
function (x, trim = 0, na.rm = FALSE)
{
  if (na.rm)
    x <- x[!is.na(x)]
  trim <- trim[1]
  n <- length(c(x, recursive = TRUE))
  if (trim > 0 && n > 0) {
    if (mode(x) == "complex")
      stop("trimmed means are not defined for complex data")
    if (trim >= 0.5)
      return(median(x, na.rm = FALSE))
    lo <- floor(n * trim) + 1
    hi <- n + 1 - lo
    x <- sort(x, partial = unique(c(lo, hi)))[lo:hi]
    n <- hi - lo + 1
  }
  sum(x)/n
}
```

Funkcijos aprašymas, o taip pat ir jos tekstas atrodo gana komplikuoti, kadangi į juos įtraukta `trim`⁶ opcija, diagnostiniai žingsniai ir instrukcijos ką daryti, kai vektorius `x` turi praleistų reikšmių (tai dažnai pasitaiko realiuose uždaviniuose). Jei `trim=0` (tai standartinė (= default (angl.)) `trim` reikšmė, jos nurodyti nereikia), tai surinkę ekrane tekstą `mean(y)` (arba `mean(1:5)`), rastume pirmų penkių natūraliųjų skaičių (aritmetinį) vidurkį:

```
> mean(y)
[1] 3
```

Tą patį rezultatą gautume ir su tokia paprasta funkcija:

```
> mean.mano <- function(x){sum(x)/length(x)}
> mean.mano(y)
[1] 3
```

Vidurkis dažnai naudojamas, charakterizuojant ėminio centrinę ar vidutinę ar vidurinę reikšmę. Deja, statistikoje dažnai susiduriame su įrašų klaidomis, o tuomet vidurkis gali smarkiai nutolti nuo “tikrosios” reikšmės. Pvz.,

⁶ `trim` (angl.) = pakirpimas, palyginimas

```
mean(c(1,2,3,4,50))
[1] 12
```

Funkcija `mean` bus atsparesnė klaidoms, jei atmesime, pvz., po 20% didžiausių ir mažiausių reikšmių:

```
mean(c(1,2,3,4,50),trim=0.2)
[1] 3
```

Grįžkime prie meniu eilutės `Help` skyriaus.

4) Pasirinkę `Html help`, pakliūtume į puslapį su daugeliu sąsajų (link'ų). Sąsajos gali mus nukreipti į `An Introduction to R`, `The R language definition`, `Writing R extensions` ar kitus skyrius. Šiuos tekstus (tiksliau, hipertekstus) lengva skaityti, kadangi juose galima keliauti iš vienos vietos į kitą, naudojantis vidinėmis sąsajomis. Pvz., spragtelėję ant `Search Engine & Keywords`, pakliūname į kitą R pagalbos aplinką (`Search Engine R`): paieškos langelyje surinkę `mean`, vėl pakliūtume į anksčiau pateiktą funkcijos `mean` aprašymo puslapį, tačiau jo apačioje dabar yra sąsaja su gimininga funkcija `weighted.mean` ir `mean.POSIXct`.

R meniu eilutėje yra ir daugiau skyrių (`File`, `Edit`, `Misc`, `Packages`, `Windows`). Juos aptarsime tinkamu laiku.

2.3. R bibliotekos ir duomenų rinkiniai

R funkcijos yra apjungtos į bibliotekas (= `packages` (angl.)), kurios gali būti prijungtos prie darbinės srities arba, kai nebereikalingos, atjungtos. Įjungiant R, automatiškai instaliuojamos septynios. Tuo galima įsitikinti, apžiūrėjus paieškos kelią:

```
> search()
[1] ".GlobalEnv" "package:methods" "package:ctest"
[4] "package:mva" "package:modreg" "package:nls"
[7] "package:ts" "Autoloader" "package:base"
```

Dauguma šiam kursui reikalingų funkcijų priklauso šioms bibliotekoms. Jei mūsų tyrimui reikėtų bibliotekų, nesančių minėtoje direktorijoje, jas galima atsisiųsti iš <http://cran.hu.r-project.org/> (žr. `Software|Package Sources`) ir išzipuoti į `...\\rw1071\\library` direktoriją. Instaliuokime, pavyzdžiui, `car` biblioteką – ji mums pravertė ateityje, į ją įeina procedūros, naudojamos J.Fox'o knygoje "Companion to Applied Regression". Be jau minėto metodo, ją galima instaliuoti dar bent dviem būdais. Jei jūsų kompiuteris prijungtas prie interneto ir įjungta kuri nors interneto naršyklė, surinkite⁷

```
install.packages("car") –
```

tolimesnė instaliacija bus atlikta automatiškai. Biblioteka `car` taip pat yra autoriaus sudarytame kompaktiniame diske `R1`, instaliuoti iš jo galima taip: R meniu eilutėje renkamės `Packages|Install package from local zip file`, pasirodžius lentelėi `Select zip file to`

⁷ Vienu metu galima instaliuoti ir kelias bibliotekas, pavyzdžiui: `install.packages(c("car", "gregmisc"))`.

install, nurodome kelią ...R-packages\car.zip ir spragtelime ant Open. Bibliotekų direktorija bus papildyta nauja eilute:

```
car                Companion to Applied Regression
```

visų jų sąrašą galima gauti su komanda

```
> library().
```

O štai pats (autorius kompiuteryje esančių bibliotekų) sąrašas:

```
Packages in library 'C:/PROGRA~1/R/rw1071/library':
```

base	The R base package
boot	Bootstrap R (S-Plus) Functions (Canty)
car	Companion to Applied Regression
class	Functions for classification
cluster	Functions for clustering (by Rousseeuw et al.)
cctest	Classical Tests
eda	Exploratory Data Analysis
foreign	Read data stored by Minitab, S, SAS, SPSS, Stata, ...
grid	The Grid Graphics Package
KernSmooth	Functions for kernel smoothing for Wand & Jones (1995)
lattice	Lattice Graphics
lqs	Resistant Regression and Covariance Estimation
MASS	Main Library of Venables and Ripley's MASS
methods	Formal Methods and Classes
mgcv	Multiple smoothing parameter estimation and GAMS by GCV
modreg	Modern Regression: Smoothing and Local Methods
multcomp	Multiple Tests and Simultaneous Confidence Intervals
mva	Classical Multivariate Analysis
nlme	Linear and nonlinear mixed effects models
nls	Nonlinear regression
nnet	Feed-forward neural networks and multinomial log-linear models
Rcmdr	R Commander
rpart	Recursive partitioning
spatial	functions for kriging and point pattern analysis
splines	Regression Spline Functions and Classes
stepfun	Step Functions, including Empirical Distributions
strucchange	Testing for Structural Change
survival	Survival analysis, including penalised likelihood.
tcltk	Tcl/Tk Interface
tools	Tools for package development
ts	Time series functions
tseries	Time series analysis and computational finance

Jei esate R aplinkoje ir norite prijungti car biblioteką, surinkite komandą

```
> library(car) # Atkreipkite dėmesį: ne package(car)!
```

Dabar

```
> search()
[1] ".GlobalEnv"      "package:car"      "package:methods"
```

```
[4] "package:ctest"      "package:mva"        "package:modreg"
[7] "package:nls"        "package:ts"         "AutoLoads"
[10] "package:base"
```

Jei car biblioteka bus dažnai naudojama, ją galima pakrauti įjungimo metu. Atidarykite bet kokio teksto redaktoriaus (pvz., Notepad'o) naują langą ir surinkite eilutę library(car), po to pasirinkite File | Save as...| Save as type: | All Files | File name: ".Rprofile" ir išsaugokite ją su Save in savo projekto darbinėje direktorijoje. Kitą kartą car biblioteka bus pakrauta įjungimo metu.

Šiuo metu R turi labai daug specializuotų bibliotekų. Jei jūsų kompiuteris prijungtas prie interneto, jų sąrašą galima gauti su

```
> CRAN.packages()[,1]
```

o bibliotekų šiuo metu (2003 m. rugpjūtis) yra 232:

```
> length(CRAN.packages()[,1])
[1] 232
```

Jums reikalingų bibliotekų pakrovimą galima automatizuoti. Pvz., norint instaliuoti gregmisc ir xgobi bibliotekas, reikia surinkti

```
> install.packages(c("gregmisc", "xgobi"))
```

Jei norite instaliuoti absoliučiai visas šiuo metu CRAN'e esančias bibliotekas (tai 20+Mb), surinkite

```
> install.packages(CRAN.packages()[,1])
```

(po to kartą į kelias savaites neužmirškite atnaujinti jas su update.packages()).

Tolimesniame darbe dažnai naudosisimės į R įmontuotais duomenų rinkiniais. Norėdami gauti sąrašą rinkinių, prijungtų prie dabartinio paieškos kelio, surinkime

```
> data()
Data sets in package `car':
```

Adler	Experimenter Expectations
Angell	Moral Integration of American Cities
Anscombe	U. S. State Public-School Expenditures
Baumann	Methods of Teaching Reading Comprehension
Bfox	Canadian Women's Labour-Force Participation
Burt	Fraudulent Data on IQs of Twins Raised Apart
Can.pop	Canadian Population Data
Chile	Voting Intentions in the 1988 Chilean Plebiscite
Chirot	The 1907 Romanian Peasant Rebellion
Davis	Self-Reports of Height and Weight
Duncan	Duncan's Occupational Prestige Data
Erickson	The 1980 U.S. Census Undercount
Florida	Florida County Voting
Freedman	Crowding and Crime in U. S. Metropolitan Areas
Friendly	Format Effects on Recall
Ginzberg	Data on Depression
Greene	Refugee Appeals
Guyer	Anonymity and Cooperation
Hartnagel	Canadian Crime-Rates Time Series

Leinhardt	Data on Infant-Mortality
Mandel	Contrived Collinear Data
Migration	Canadian Interprovincial Migration Data
Moore	Status, Authoritarianism, and Conformity
Mroz	U.S. Women's Labor-Force Participation
Ornstein	Interlocking Directorates Among Major Canadian Firms
Prestige	Prestige of Canadian Occupations
Quartet	Four Regression Datasets
Robey	Fertility and Contraception
SLID	Survey of Labour and Income Dynamics
Sahlins	Agricultural Production in Mazulu Village
States	Education and Related Statistics for the U.S. States
UN	GDP and Infant Mortality
US.pop	Population of the United States
Vocab	Vocabulary and Education
Womenlf	Canadian Women's Labour-Force Participation

Data sets in package `base':

Formaldehyde	Determination of Formaldehyde concentration
HairEyeColor	Hair and eye color of statistics students
InsectSprays	Effectiveness of insect sprays
LifeCycleSavings	Intercountry life-cycle savings data
OrchardSprays	Potency of orchard sprays
PlantGrowth	Results from an experiment on plant growth
Titanic	Survival of passengers on the Titanic
ToothGrowth	The effect of vitamin C on tooth growth in guinea pigs
UCBAdmissions	Student admissions at UC Berkeley
USArrests	Violent crime statistics for the USA
USJudgeRatings	Lawyers' ratings of state judges in the US Superior Court
USPersonalExpenditure	Personal expenditure data
VADeaths	Death rates in Virginia (1940)
airmiles	Passenger miles on US airlines 1937-1960
airquality	New York Air Quality Measurements
anscombe	Anscombe's quartet of regression data
attenu	Joiner-Boore Attenuation Data
attitude	Chatterjee-Price Attitude Data
cars	Speed and Stopping Distances for Cars
chickwts	The Effect of Dietary Supplements on Chick Weights
co2	Moana Loa Atmospheric CO2 Concentrations
discoveries	Yearly Numbers of `Important' Discoveries
esoph	(O)esophageal Cancer Case-control study
euro	Conversion rates of Euro currencies
eurodist	Distances between European Cities
faithful	Old Faithful Geyser Data
freeny	Freeny's Revenue Data
infert	Secondary infertility matched case-control study
iris	Edgar Anderson's Iris Data as data.frame
iris3	Edgar Anderson's Iris Data as 3-d array
islands	World Landmass Areas
longley	Longley's Economic Regression Data
morley	Michaelson-Morley Speed of Light Data
mtcars	Motor Trend Car Data
nhtemp	Yearly Average Temperatures in New Haven CT
phones	The Numbers of Telephones
precip	Average Precipitation amounts for US Cities
presidents	Quarterly Approval Ratings for US Presidents
pressure	Vapour Pressure of Mercury as a Function of Temperature
quakes	Earthquake Locations and Magnitudes in the Tonga Trench
randu	Random Numbers produced by RANDU
rivers	Lengths of Major Rivers in North America

```

sleep           Student's Sleep Data
stackloss      Brownlee's Stack Loss Plant Data
state          US State Facts and Figures
sunspots       Monthly Mean Relative Sunspot Numbers
swiss          Swiss Demographic Data
trees          Girth, Height and Volume for Black Cherry Trees
uspop          Populations Recorded by the US Census
volcano        Topographic Information on Auckland's Maunga Whau Volcano
warpbreaks     Breaks in Yarn during Weaving
women          Heights and Weights of Women

```

Jei norite pamatyti visus instaliuoto paketo MASS duomenų rinkinius, surinkite

```
> data(package="MASS"),
```

o jei visų instaliuotų paketų duomenų rinkinius -

```
> data(package = .packages(all.available = TRUE))
```

Kiekvieną R biblioteką sudaro dviejų rūšių objektai: duomeniniai (data sets) ir funkciniai (functions). Visus car objektus galime pamatyti su

```
> library(help=car)
```

(duomeninių objektų sąrašą matėme aukščiau, dabar jis bus papildytas bibliotekai car priklausančiomis funkcijomis).

Kaip galima apžiūrėti duomeninius objektus? car bibliotekoje yra rinkinys Davis, kuriame pateikti 200 reguliariai užsiiminėjančių sportu asmenų (vyrų=M ir moterų=F) duomenys apie jų svorį (tikrąjį weight ir praneštąjį repwt) bei ūgį (tikrąjį height ir praneštąjį repht). Surinkime

```

> data(Davis) # Duomenys išarchivuojami
> Davis
  sex weight height repwt repht
1   M    77    182    77    180
2   F    58    161    51    159
3   F    53    161    54    158
*****
198  M    81    175    NA     NA
199  M    90    181    91    178
200  M    79    177    81    178

```

Bendrają informaciją apie šį duomenų rinkinį galima gauti su

```

> summary(Davis)
sex      weight      height      repwt      repht
F:112   Min.   : 39.0   Min.   : 57.0   Min.   : 41.00  Min.   :148.0
M: 88   1st Qu.: 55.0   1st Qu.:164.0   1st Qu.: 55.00  1st Qu.:160.5
        Median : 63.0   Median :169.5   Median : 63.00  Median :168.0
        Mean   : 65.8   Mean   :170.0   Mean   : 65.62  Mean   :168.5
        3rd Qu.: 74.0   3rd Qu.:177.3   3rd Qu.: 73.50  3rd Qu.:175.0
        Max.   :166.0   Max.   :197.0   Max.   :124.00  Max.   :200.0
        NA's   : 17.00  NA's   : 17.00

```

arba su⁸

⁸ Jei jūsų objektų sąrašas nėra ilgas, galite surizikuoti ir surinkti ls.str()- pamatysite visų savo objektų aprašymus.

```
> str(Davis)
`data.frame': 200 obs. of 5 variables:
 $ sex : Factor w/ 2 levels "F","M": 2 1 1 2 1 2 2 2 2 2 ...
 $ weight: int 77 58 53 68 59 76 76 69 71 65 ...
 $ height: int 182 161 161 177 157 170 167 186 178 171 ...
 $ repwt : int 77 51 54 70 59 76 77 73 71 64 ...
 $ repht : int 180 159 158 175 155 165 165 180 175 170 ...
```

Aišku, kad (vidutiniškai) vyrai yra aukštesni už moteris, tačiau kiek? Žinome, kad vidurkį skaičiuoja funkcija `mean`, deja komanda `mean(height)` neveikia:

```
> mean(height)
Error in mean(height) : Object "height" not found
```

Reikalas tas, kad `height` yra tik stulpelio vardas, o ne R duomeninis objektas. Norint, kad `Davis` stulpeliai taptų pasiekiami (su sąlyga, kad paieškos kelyje nėra kitų kintamųjų su šių stulpelių vardais), surinkime

```
> attach(Davis) # attach (angl.) = prijungti
> mean(height)
[1] 170.02
```

Deja, mums reikia ne visų tiriamųjų ūgio vidurkio, bet atskirai vyrų ir moterų. Tai galima atlikti tiesiogiai, t.y., skaičiuojant tik, pvz., vyrų ūgio vidurkį. Kiek detaliau paaiškinsime kaip tai galima atlikti.

```
> height # Visų tiriamųjų ūgiai

 [1] 182 161 161 177 157 170 167 186 178 171 175 57 161 168 163
 [16] 166 187 168 197 175 180 170 175 173 171 166 169 166 157 183
 [31] 166 178 173 164 169 176 166 174 178 187 164 178 163 183 179
 [46] 160 180 161 174 162 182 165 169 185 177 176 170 183 172 173
 [61] 165 177 180 173 189 162 165 164 158 178 175 173 165 163 166
 [76] 171 160 160 182 183 165 168 169 167 170 182 178 165 163 162
 [91] 173 161 184 180 189 165 185 169 159 155 164 178 163 163 175
 [106] 164 152 167 166 166 183 179 174 179 167 168 184 184 169 178
 [121] 178 167 178 165 179 169 153 157 171 157 166 185 160 148 177
 [136] 162 172 167 188 191 175 163 165 176 171 160 165 157 173 184
 [151] 168 162 150 162 163 169 172 170 169 167 163 161 162 172 163
 [166] 159 170 166 191 158 169 163 170 176 168 178 174 170 178 174
 [181] 176 154 181 165 173 162 172 169 183 158 185 173 164 156 164
 [196] 175 180 175 181 177
```

```
> sex # Visų tiriamųjų lytis (kadangi duomenų rinkinys Davis yra
# prijungtas, kintamasis sex yra pasiekiamas)
```

```
 [1] M F F M F M M M M M F F F F F M F M F M F M M F F F F F M
 [31] F M M F F M F M M M F M F M M F M F F F M F M M M M F M M M
 [61] M M M F M F F F F M F M F F F F F M M F M F F F F F F F
 [91] M F M M M F M F F F F F F F F F F F F M M F M M M M M M
 [121] M M F F M F F F F F M F F F F F M M M F F F F F F F M
 [151] F F F F F M M F F F F F F M F F F M F M F M M M M F M M M
 [181] M F M F M F F F M F M M F F F M M M M M
Levels: F M
```

```
> sex=="M" # Kurie tiriamieji yra Male?

 [1] TRUE FALSE FALSE TRUE FALSE TRUE TRUE TRUE TRUE TRUE
 [11] TRUE FALSE FALSE FALSE FALSE FALSE TRUE FALSE TRUE FALSE
```

```

[21] TRUE FALSE TRUE TRUE FALSE FALSE FALSE FALSE FALSE TRUE
[31] FALSE TRUE TRUE FALSE FALSE TRUE FALSE TRUE TRUE TRUE
[41] FALSE TRUE FALSE TRUE TRUE FALSE TRUE FALSE FALSE FALSE
[51] TRUE FALSE TRUE TRUE TRUE TRUE FALSE TRUE TRUE TRUE
[61] TRUE TRUE TRUE FALSE TRUE TRUE FALSE FALSE FALSE TRUE
[71] FALSE TRUE FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE
[81] FALSE TRUE FALSE FALSE FALSE TRUE TRUE FALSE FALSE FALSE
[91] TRUE FALSE TRUE TRUE TRUE FALSE TRUE FALSE FALSE FALSE
[101] FALSE TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[111] TRUE TRUE FALSE TRUE FALSE FALSE TRUE TRUE TRUE TRUE
[121] TRUE TRUE FALSE FALSE TRUE FALSE FALSE FALSE FALSE FALSE
[131] FALSE TRUE FALSE FALSE TRUE FALSE FALSE FALSE TRUE TRUE
[141] TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE TRUE
[151] FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE FALSE FALSE
[161] FALSE FALSE FALSE FALSE TRUE FALSE FALSE FALSE TRUE FALSE
[171] TRUE FALSE TRUE TRUE TRUE TRUE FALSE TRUE TRUE TRUE
[181] TRUE FALSE TRUE FALSE TRUE FALSE FALSE FALSE TRUE FALSE
[191] TRUE TRUE FALSE FALSE FALSE TRUE TRUE TRUE TRUE TRUE

```

```

> height[sex=="M"] # Tik vyrų (Male) ūgiai (simbolis "[...]" žymi
# poaibio operacija)

```

```

[1] 182 177 170 167 186 178 171 175 187 197 180 175 173 183 178
[16] 173 176 174 178 187 178 183 179 180 182 169 185 177 176 183
[31] 172 173 165 177 180 189 178 173 182 183 168 182 178 173 184
[46] 180 189 185 178 183 179 179 184 184 169 178 178 167 179 185
[61] 177 188 191 175 184 169 172 163 191 169 170 176 168 178 170
[76] 178 174 176 181 173 183 185 173 175 180 175 181 177

```

```

> mean(height[sex=="M"]) # Vyrų ūgio vidurkis

```

```

[1] 178.0114

```

Kadangi skaičiavimo procedūra dabar aiški, moterų ūgio vidurkį apskaičiuosime iš karto:

```

> mean(height[sex=="F"])

```

```

[1] 163.7411

```

Pasirodo, kad visus `height` įrašus suskirstyti į dvi grupes galima ir kitaip (su funkcija `tapply`⁹):

```

> tapply(height,sex,mean)

```

```

      F      M
163.7411 178.0114

```

arba naudojant funkciją `aov`:

```

> aov(height~sex)$coeff

```

```

(Intercept)      sexM
 163.74107      14.27029 # Tai vyrų (male) ūgio priedas

```

arba funkciją `summary`:

```

> summary(height[sex=="F"])

```

```

  Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  57.0   161.0   165.0   163.7   169.0   178.0

```

```

> summary(height[sex=="M"])

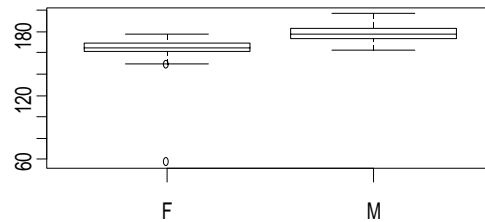
```

⁹ Daugiau apie šią funkciją žr. 3-23 psl.

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
163	173	178	178	183	197

Atrodo, kad duomenų rinkinyje Davis yra klaidų: moterų ūgio minimumas tikrai keistas. Prie klaidų paieškos grįšime vėliau, o dabar tą patį faktą pademonstruosime kitaip, išbrėždami stačiakampes moterų ir vyrų ūgio diagramas:

```
> boxplot(height~sex)
```



2.3 pav. Moterų (F) ir vyrų (M) ūgio stačiakampės diagramos

(taigi F (moterų) grupėje iš tikro yra dvi išskirtys; jos pažymėtos rutuliukais, o viena iš jų lygi maždaug 60).

Darbą su duomenų rinkiniu baigus, jį tikslinga atjungti:

```
> detach(Davis) # Komanda detach() atjungs visus duomenų rinkinius
```

Kai bibliotekos nebereikia, ją taip pat vertėtų atjungti:

```
> detach(package:car)
> search()
[1] ".GlobalEnv"      "package:ctest"  "Autoloads"     "package:base"
```

Mes jau turėjome kelis R grafikos pavyzdžius. Dar kelis pavyzdžius pasižiūrėkime su funkcija demo. Surinkę

```
> demo(),
```

pamatysime sąrašą temų iš base bibliotekos:

Demos in package 'base':

glm.vr	Some glm() examples from V&R with several predictors
graphics	A show of some of R's graphics capabilities
Hershey	Tables of the characters in the Hershey vector fonts
image	The image-like graphics builtins of R
is.things	Explore some properties of R objects and is.FOO() functions. Not for newbies!
Japanese	Tables of the Japanese characters in

```

lm.glm          the Hershey vector fonts
                Some linear and generalized linear
                modelling examples from 'An
                Introduction to Statistical Modelling'
                by Annette Dobson
nlm             Nonlinear least-squares using nlm()
persp          Extended persp() examples
plotmath       Examples of the use of mathematics
                annotation
recursion      Using recursion for adaptive
                integration
scoping        An illustration of lexical scoping.

```

Use `'demo(package = .packages(all.available = TRUE))'`
to list the demos in all `*available*` packages.

Surinkite dabar kurią nors iš komandų

```

demo(graphics)
demo(image)

```

ir pamatysite daug gražių paveikslėlių!

R turi daug funkcijų¹⁰ ir jas visa nėra lengva įsiminti, todėl ryžtingai naudokitės visomis įmanomis pagalbos priemonėmis. Jau žinome funkciją `help(image)` ir jos sinonimą `?image`. Pagalbos failo pabaigoje paprastai yra `Examples` skyrelis, kuriame yra funkcijos taikymo pavyzdžių. Juos apžvelgti galima ir tiesiogiai, pvz., su `example(image)`¹¹. Labai naudinga paieškos funkcija `help.search("image")` – i pateiks visų `library` direktorijoje esančių paketų funkcijas, kurių varduose yra `image`.

Žemiau pateikiame J. Baron'o sudarytą pagrindinių R funkcijų sąrašą:

Parentheses are for functions, brackets are for indicating the position of items in a vector or matrix. (Here, items with numbers like `x1` are user-supplied variables.)

Miscellaneous

```

q(): quit
<-: assign
INSTALL package1: install package1
m1[,2]: column 2 of matrix m1
m1[,2:5] or m1[,c(2,3,4,5)]: columns 2–5
m1$a1: variable a1 in data frame m1

```

```

10 > search()
[1] ".GlobalEnv"      "Davis"           "package:car"
[4] "package:methods" "package:ctest"   "package:mva"
[7] "package:modreg"   "package:nls"     "package:ts"
[10] "Autoloads"       "package:base"
> length(ls(pos=11))
[1] 1658

```

taigi `base` paketas turi 1658 funkcijas!

¹¹ Terminas `image` sakinyje `demo(image)` reiškia temų ratą, o sakinyje `example(image)` – funkcijos vardą. Funkcija `graphics` neegzistuoja. Norint, kad `example(image)` paveikslai nepralėktų žaibu ekrane, prieš šią komandą surinkite `par(ask=TRUE)`, o pasibaigus demonstracijai – `par(ask=FALSE)`.

NA: missing data
is.na: true if data missing
library(mva): load (e.g.) the mva package

Help

help(command1): get help with command1 (NOTE: USE THIS FOR MORE DETAIL THAN THIS CARD CAN PROVIDE.)
help.start(): start browser help
help(package=mva): help with (e.g.) package mva
apropos("topic1") and help.search("topic 1"): commands relevant to topic1
example(command1): examples of command1

Input and output

source("file1"): run the commands in file1.
read.table("file1"): read in data from file1
data.entry(): spreadsheet
scan(x1): read a vector x1
download.file("url1"): from internet
url.show("url1"), read.table.url("url1"): remote input
sink("file1"): output to file1, until sink()
write(object1, "file1"): writes object1 to file1
write.table(dataframe1, "file1"): writes a table

Managing variables and objects

attach(x1) detach(x1): put (remove) x1 in search path
ls(): lists all the active objects.
str(object1): print useful information about object1
rm(object1): remove object1
dim(matrix1): dimensions of matrix1
dimnames(x1): names of dimensions of x1
length(vector1): length of vector1
1:3: the vector 1,2,3
c(1,2,3): creates the same vector
rep(x1,n1): repeats the vector x1 n1 times
cbind(a1,b1,c1), rbind(a1,b1,c1): binds columns or rows into a matrix
merge(df1,df2): merge data frames
matrix(vector1,r1,c1): make vector1 into a matrix with r1 rows and c1 columns
data.frame(v1,v2): make a data frame from vectors v1 and v2
as.factor(), as.matrix(), as.vector(): conversion
is.factor(), is.matrix(), is.vector(): what is it?
t(): switch rows and columns
which(x1==a1): returns indices of x1 where x1==a1

Control flow

for (i1 in vector1): repeat what follows
if (condition1) ...else ...: conditional

Arithmetic

%*%: matrix multiplication
%/%, ^, %%, sqrt(): integer division, power, modulus, square root

Statistics

`max()`, `min()`, `mean()`, `median()`, `sum()`, `var()`: as named
`summary(data.frame)`: prints statistics
`rank()`, `sort()`: rank and sort
`ave(x1,y1)`: averages of x1 grouped by factor y1
`by()`: apply function to data frame by factor
`apply(x1,n1,function1)`: apply function1 (e.g. mean) to x by rows (n1=1) or columns (n2=2)
`tapply(x1,list1,function1)`: apply function to x1 by list1
`table()`: make a table
`tabulate()`: tabulate a vector

Basic statistical analysis

`aov()`, `anova()`, `lm()`, `glm()`: (generalized) linear models, anova
`t.test()`: ttest
`prop.test()`, `binom.test()`: tests on probability
`chisq.test(x1)`: chi-square test on matrix x1
`fisher.test()`: Fisher exact test
`cor(a)`: show correlations
`cor.test(a,b)`: test correlation
`friedman.test()`: Friedman test

Graphics

`plot()`, `barplot()`, `boxplot()`, `stem()`, `hist()`: basic plots
`matplot()`: matrix plot
`pairs(matrix)`: scatterplots
`coplot()`: conditional plot
`stripplot()`: strip plot
`qqplot()`: quantile-quantile plot
`qqnorm()`, `qqline()`: fit normal distribution

O čia dar viena, E. Paradis sudaryta, atmintinė:

1.

Operators					
Arithmetic		Comparison		Logical	
+	addition	<	lesser than	! x	logical NOT
-	subtraction	>	greater than	x & y	logical AND
*	multiplication	<=	lesser than or equal to	x && y	id.
/	division	>=	greater than or equal to	x y	logical OR
^	power	==	equal	x y	id.
%%	modulo	!=	different	xor(x, y)	exclusive OR
%/%	integer division				

2.

<code>sum(x)</code>	sum of the elements of <code>x</code>
<code>prod(x)</code>	product of the elements of <code>x</code>
<code>max(x)</code>	maximum of the elements of <code>x</code>
<code>min(x)</code>	minimum of the elements of <code>x</code>
<code>which.max(x)</code>	returns the index of the greatest element of <code>x</code>
<code>which.min(x)</code>	returns the index of the smallest element of <code>x</code>
<code>range(x)</code>	id. than <code>c(min(x), max(x))</code>
<code>length(x)</code>	number of elements in <code>x</code>
<code>mean(x)</code>	mean of the elements of <code>x</code>
<code>median(x)</code>	median of the elements of <code>x</code>
<code>var(x)</code> or <code>cov(x)</code>	variance of the elements of <code>x</code> (calculated on $n-1$); if <code>x</code> is a matrix or a data frame, the variance-covariance matrix is calculated
<code>cor(x)</code>	correlation matrix of <code>x</code> if it is a matrix or a data frame (1 if <code>x</code> is a vector)
<code>var(x, y)</code> or <code>cov(x, y)</code>	covariance between <code>x</code> and <code>y</code> , or between the columns of <code>x</code> and those of <code>y</code> if they are matrices or data frames
<code>cor(x, y)</code>	linear correlation between <code>x</code> and <code>y</code> , or correlation matrix if they are matrices or data frames

Šių funkcijų¹² reikšmė yra skaičius (t.y., ilgio 1 vektorius). Žemiau pateiktų funkcijų reikšmė gali būti gana komplikuota.

3.

<code>round(x, n)</code>	rounds the elements of <code>x</code> to <code>n</code> decimals
<code>rev(x)</code>	reverses the elements of <code>x</code>
<code>sort(x)</code>	sorts the elements of <code>x</code> in increasing order; to sort in decreasing order: <code>rev(sort(x))</code>
<code>rank(x)</code>	ranks of the elements of <code>x</code>
<code>log(x, base)</code>	computes the logarithm of <code>x</code> with base "base"
<code>scale(x)</code>	if <code>x</code> is a matrix, centers and reduces the data; to center only use the option <code>center=FALSE</code> , to reduce only <code>scale=FALSE</code> (by default <code>center=TRUE</code> , <code>scale=TRUE</code>)
<code>pmin(x, y, ...)</code>	a vector which <code>i</code> th element is the minimum of <code>x[i]</code> , <code>y[i]</code> , ...
<code>pmax(x, y, ...)</code>	id. for the maximum
<code>cumsum(x)</code>	a vector which <code>i</code> th element is the sum from <code>x[1]</code> to <code>x[i]</code>
<code>cumprod(x)</code>	id. for the product
<code>cummin(x)</code>	id. for the minimum
<code>cummax(x)</code>	id. for the maximum
<code>match(x, y)</code>	returns a vector of the same length than <code>x</code> with the elements of <code>x</code> which are in <code>y</code> (NA otherwise)
<code>which(x == a)</code>	returns a vector of the indices of <code>x</code> if the comparison operation is true (TRUE), in this example the values of <code>i</code> for which <code>x[i] == a</code> (the argument of this function must be a variable of mode logical)
<code>choose(n, k)</code>	computes the combinations of <code>k</code> events among <code>n</code> repetitions = $n! / [(n-k)!k!]$
<code>na.omit(x)</code>	suppresses the observations with missing data (NA) (suppresses the corresponding line if <code>x</code> is a matrix or a data frame)
<code>na.fail(x)</code>	returns an error message if <code>x</code> contains at least one NA
<code>unique(x)</code>	if <code>x</code> is a vector or a data frame, returns a similar object but with the duplicate elements suppressed
<code>table(x)</code>	returns a table with the numbers of the different values of <code>x</code> (typically for integers or factors)
<code>subset(x, ...)</code>	returns a selection of <code>x</code> with respect to criteria (... , typically comparisons: <code>x\$V1 < 10</code>); if <code>x</code> is a data frame, the option <code>select</code> gives the variables to be kept (or dropped using a minus sign -)
<code>sample(x, size)</code>	resample randomly and without replacement <code>size</code> elements in the vector <code>x</code> , the option <code>replace = TRUE</code> allows to resample with replacement

¹² Išskyrus `range()`, kuri gražina du skaičius ir `var()`, `cov()` bei `cor()`, kurios gali gražinti matričią.

Pažymėsime, kad visų (pvz., base bibliotekos) funkcijų sąrašą galima rasti iš R meniu eilutės nuvairavus į Help|Html help|Packages|base.

R darbinėje direktorijoje yra naudingas failas .Rhistory , kuriame fiksuojamos visos šios ir ankstesnių sesijų metu įvykdytos (ir išsaugotos! – žr. keliomis eilutėmis žemiau) komandos. Apžiūrėti šį failą galime su `history()` , o pasižymėję jame reikalingas komandas, spragtelėję dešiniuoju klavišu ir pasirinkę Paste to console, galėsite pakartoti ankstesnę analizę.

Norėdami baigti sesiją, surinkite `q()` . Jei į klausimą Save workspace image? atsakysite Yes, tai visa šios sesijos istorija (t.y., vykdytos komandos (bet ne grafikai)) bus išsaugota, o jei No – ne.

2.4. R literatūra, konferencija, archyvai

Yra nemažai literatūros, skirtos darbui su R paketu. Daug jos patalpinta internete, beveik visos šios knygos yra kompaktiniame diske R1. Pradedantiesiems ypač siūlyčiau šias knygas:

- 1) John Verzani, **Simple R** <http://www.math.csi.cuny.edu/Statistics/R/simpleR>
- 2) John Maindonald , **Using R for Data Analysis and Graphics**, žr. <http://cran.hu.r-project.org/Documentation/Contributed>
- 3) Emmanuel Paradis, **R for Beginners**, žr. <http://cran.hu.r-project.org/Documentation/Contributed>

Tiems, kurie ruošiasi dirbti su R paketu ir ateityje, autorius rekomenduoja dalyvauti nuolat veikiančioje konferencijoje (list): adresu r-help-request@lists.r-project.org išsiųskite e laišką, kurio kūne (ne subject'e!) turi būti vienintelis žodis subscribe. Po to kasdien gausite kokį 10-100 laiškų, kurių dažnas bus jums naudingas. Paiešką šios konferencijos archive galima atlikti keliais būdais; jie aprašyti svetainės <http://cran.hu.r-project.org/> CRAN|Search skyriuje.

3. Duomenų įrašymas ir programavimo pavyzdžiai

Duomenų rinkimas ir įrašymas dažnai užima daugiau laiko negu jų statistinė analizė. Deja, tai būtinas etapas.

Kiekvienas R duomeninis objektas visuomet turi du vidinius požymius (tipą (mode) ir ilgį (length)) ir dar gali turėti vieną ar kelis papildomus požymius (attributes) (pvz., klasę (class) ar matavimų skaičių (dimension)). Žemiau esančioje lentelėje pateikta šių faktų santrauka.

Objektas	Galimi tipai	Ar galima naudoti skirtingus tipus viename objekte?	Klasė
Vektorius (vector)	Skaitinis, simbolinis, kompleksinis ar loginis	ne	NULL
Vardinis kintamasis = faktorius (factor)	Skaitinis ar simbolinis	ne	factor
Ranginis kintamasis (ordered factor)	Skaitinis ar simbolinis	ne	factor ordered
Masyvas (array)	Skaitinis, simbolinis, kompleksinis ar loginis	ne	NULL
Matrica (matrix)	Skaitinis, simbolinis, kompleksinis ar loginis	ne	matrix
Duomenų sistema (data frame)	Skaitinis, simbolinis, kompleksinis ar loginis	taip	data.frame
Laiko eilutė (ts= time series)	Skaitinis, simbolinis, kompleksinis ar loginis	taip	ts
Sąrašas (list)	Skaitinis, simbolinis, kompleksinis, loginis, funkcija, reiškinys ar formulė	taip	list

R yra objektiškai orientuota kalba – pvz., funkcijų `plot` ar `summary` elgesys (reikšmė) priklauso nuo objekto klasės.

3.1. Duomenų įrašymas rankomis

Pradėsime nuo to (reto) atvejo, kai duomenys į kompiuterio atmintį įrašomi, naudojantis pačiu paketu R.

3.1.1. Skaitiniai vektoriai ir matricos

Tarkime, kad skaitinio vektoriaus skve komponentės yra 1,2,3,3,3,3,3¹. Šių vektorių galima įrašyti keliais būdais.

1a)

```
skvela <- c(1,2,3,3,3,3,3)
skvel
[1] 1 2 3 3 3 3 3
> mode(skvela)
[1] "numeric"
> length(skvela)
[1] 7
> class(skvela)
NULL
```

1b)

```
skvelb <- c(1,2,rep(3,5)) # Surinkite > ?rep (paeksperimentuoki-
# te: rep(3,1:3), rep(1:3,3), rep(1:3,
# 1:3) ir t.t.)
```

2)

```
skve2 <- scan()
1: 1 2 # Paspauskite Enter
3: 3 3 3 # Norint baigti duomenų įvedimą, Enter
7: # klavišą reikia paspausti du kartus
Read 6 items
```

3) Kai duomenų daug, geriau naudoti funkciją `edit`, kuri leis ne tik patogiai įvesti duomenis, bet ir nesunkiai juos redaguoti. Norint, kad funkcija `edit` pateiktų ekrane elektroninės lentelės pavidalo lapą, `skve3` reikia traktuoti kaip (vieno stulpelio) matricą.

```
skve3 <- matrix(1) # Įvedėme skaičių 1 - tai bus matricos
# elementas su indeksais (1,1)
skve3 <- edit(skve3) # Atsidarys R Data Editor langas;
# kai lentelę užpildysite, langą uždarykite

> mode(skve3)
[1] "numeric"
> dim(skve3)
[1] 6 1 # skve3 yra 6x1 matrica
> class(skve3)
NULL
> skve3
      col1
[1,] 1
[2,] 2
[3,] 3
[4,] 3
[5,] 3
[6,] 3
[7,] 3
```

Reikalui esant, matricai `skve3` galima sugrąžinti vektorinę struktūrą:

¹ Visi vektoriaus elementai turi būti vieno (šiuo atveju - skaitinio) tipo.

```

skve3 <- as.vector(skve3)
dim(skve3)
NULL
> skve3
[1] 1 2 3 3 3 3 3

```

Visi naujai sukurtieji vektoriai dabar yra darbinėje atmintyje (= memory, workspace, curenly active enviroment (angl.)):

```

> ls()      # ls=list=išvardink
 [1] "a"          "b"          "bwages"
 [4] "cars"       "d"          "Davis"
 [7] "doR"        "draw.plotmath.cell" "draw.title.cell"
[10] "ewr"        "f"          "faithful"
[13] "fill"       "g"          "get.c"
[16] "get.r"      "h.f"        "i"
[19] "i.out"      "iris"       "islands"
[22] "kernels"    "l"          "last.warning"
[25] "make.table" "n"          "nc"
[28] "nhtemp"     "nr"         "oldpar"
[31] "op"         "opar"       "pie.sales"
[34] "pin"        "pp"         "precip"
[37] "quakes"     "r"          "RKs"
[40] "scale"      "skvela"     "skvelb"
[43] "skve2"     "skve3"     "tt"
[46] "usr"        "volcano"    "x"
[49] "x.at"       "xadd"       "xdelta"
[52] "xm"         "xscale"     "xx"
[55] "y"          "y.at"       "yadd"
[58] "ydelta"     "yscale"     "yy"
[61] "z"          "zmin"

```

Autoriaus mašinos darbinėje atmintyje yra susikaukę daug (duomeninių ir funkcinių) objektų, norint ją visiškai išvalyti (tai rizikinga operacija!), reikia surinkti

```
>rm(list=ls())      # rm=remove=pašalinti
```

arba ekvivalenčią komandą

```

>rm(list=ls(pat="^[a-z]")) # pat=pattern=pavyzdys;
# pvz., [b-f]-angliškos raidės nuo b iki f;
# mūsų eilutė pašalins visus atmintyje
# esančius objektus, kurių
# vardai prasideda raidėmis nuo a iki z

```

o jei tik naujuosius vektorius –

```
>rm(skvela, skvelb, skve2, skve3)
```

arba

```

>rm(list=ls(pat="^skve")) # Pašalins objektus, kurių vardai
# prasideda simboliais "skve"

```

Jei norite ištrinti visus objektus, išskyrus tuos, kurie prasideda raide “s”, surinkite

```
> rm(list=ls(pat="^[a-r,t-z]"))
```

```
> ls()
[1] "scale" "skvela" "skvelb" "skve2" "skve3"
```

Dar trys variantai: eilutė `ls(pattern="^sk")` išrenks visus objektus, prasidedančius (o eilutė `ls(pattern="sk$")` – pasibaigiančius) simbolių seka `sk`; komanda `ls(pattern="sk")` išrinks visus objektus, kurių varde yra seka `sk`.

Grįžtame prie vektorių įrašymo. Skaitmeninę seką 3,4,5,6 galima sukurti taip:

```
> skse1 <- 3:7
```

o seką 0,3;0,4;0,5;0,6 su

```
> skse2 <- seq(0.3,0.6,0.1)
```

arba

```
> skse2 <- (3:7)/10
```

Pereikime prie matricos įrašymo. Žemiau yra matrica `Pastas`, kurioje pateikti duomenys apie dvidešimties siuntinių svorį, atstumą, kuriuo juos reikėjo pristatyti, ir realią pristatymo kainą.

kaina	svoris	atstumas
2.0	0.3	160
1.9	4.5	53
1.5	0.7	80
4.4	0.8	280
1.7	1.1	90
5.0	2.4	209
9.2	6.6	160
3.9	3.2	145
8.0	3.5	250
3.3	4.1	95
8.0	4.4	202
1.0	0.6	100
11.0	5.1	240
2.6	5.9	47
6.0	6.2	115
14.5	6.5	240
1.1	2.7	160
15.5	7.0	1260
14.0	7.5	190
12.1	8.1	160

Šią matricą galima įrašyti kaip ilgą vektorių, o po to suteikti jam matricos struktūrą ir dar, gal būt, stulpelių vardus.

```
Pastas <- c(2, 1.9, 1.5, 4.4, 1.7, 5, 9.2, 3.9, 8, 3.3, 8, 1, 11,
2.6, 6, 14.5, 1.1, 15.5, 14, 12.1, 0.3, 4.5, 0.7, 0.8, 1.1, 2.4, 6.6,
3.2, 3.5, 4.1, 4.4, 0.6, 5.1, 5.9, 6.2, 6.5, 2.7, 7, 7.5, 8.1, 160,
53, 80, 280, 90, 209, 160, 145, 250, 95, 202, 100, 240, 47, 115, 240,
160, 1260, 190, 160)
```

R kodais šis objektas užrašomas taip:

```
c(2, 1.9, 1.5, 4.4, 1.7, 5, 9.2, 3.9, 8, 3.3, 8, 1, 11, 2.6, 6, 14.5,
1.1, 15.5, 14, 12.1, 0.3, 4.5, 0.7, 0.8, 1.1, 2.4, 6.6, 3.2, 3.5,
4.1, 4.4, 0.6, 5.1, 5.9, 6.2, 6.5, 2.7, 7, 7.5, 8.1, 160, 53, 80,
280, 90, 209, 160, 145, 250, 95, 202, 100, 240, 47, 115, 240, 160,
1260, 190, 160)
```

Dabar šį vektorių paversime matrica.

```
Pastas <- matrix(Pastas,ncol=3)
```

Štai jo išraiška R kodais²:

```
structure(.Data = c(2, 1.9, 1.5, 4.4, 1.7, 5, 9.2, 3.9, 8, 3.3, 8, 1, 11,
2.6, 6, 14.5, 1.1, 15.5, 14, 12.1, 0.3, 4.5, 0.7, 0.8, 1.1, 2.4, 6.6, 3.2,
3.5, 4.1, 4.4, 0.6, 5.1, 5.9, 6.2, 6.5, 2.7, 7, 7.5, 8.1, 160, 53, 80, 280,
90, 209, 160, 145, 250, 95, 202, 100, 240, 47, 115, 240, 160, 1260, 190,
160), .Dim = c(20, 3))
```

Stulpeliams suteikime vardus:

```
colnames(Pastas) <- c("kaina","svoris","atstumas")3
```

arba

```
Pastas <- matrix(c(2.0,1.9,...,12.1,0.3,4.5,...,8.1,160,53,...,160),ncol=3,
dimnames=list(NULL,c("kaina","svoris","atstumas")))4
```

R kodais ši matrica dabar atrodo taip:

```
structure(c(2, 1.9, 1.5, 4.4, 1.7, 5, 9.2, 3.9, 8, 3.3, 8, 1, 11, 2.6, 6,
14.5, 1.1, 15.5, 14, 12.1, 0.3, 4.5, 0.7, 0.8, 1.1, 2.4, 6.6, 3.2, 3.5, 4.1,
4.4, 0.6, 5.1, 5.9, 6.2, 6.5, 2.7, 7, 7.5, 8.1, 160, 53, 80, 280, 90, 209,
160, 145, 250, 95, 202, 100, 240, 47, 115, 240, 160, 1260, 190, 160), .Dim =
c(20, 3), .Dimnames = list(NULL, c("kaina", "svoris", "atstumas")))
```

Matricai `Pastas` galime suteikti duomenų sistemos struktūrą.

```
Pastas.df <- as.data.frame(Pastas)
```

Štai šios duomenų sistemos išraiška R kodais:

```
structure(list(kaina = c(2, 1.9, 1.5, 4.4, 1.7, 5, 9.2, 3.9, 8, 3.3, 8, 1,
11, 2.6, 6, 14.5, 1.1, 15.5, 14, 12.1), svoris = c(0.3, 4.5, 0.7, 0.8, 1.1,
2.4, 6.6, 3.2, 3.5, 4.1, 4.4, 0.6, 5.1, 5.9, 6.2, 6.5, 2.7, 7, 7.5, 8.1),
atstumas = c(160, 53, 80, 280, 90, 209, 160, 145, 250, 95, 202, 100, 240,
47, 115, 240, 160, 1260, 190, 160)), .Names = c("kaina", "svoris", "atstu-
mas"), row.names = c("1", "2", "3", "4", "5", "6", "7", "8", "9", "10",
"11", "12", "13", "14", "15", "16", "17", "18", "19", "20"), class = "da-
ta.frame")
```

Dar kartą pažymėsime, kad ir matricą lengviau įrašyti, naudojantis `edit` komanda (atlikite tai patys).

² Pabandykite `dump("Pastas","Pastas.matrica")` – R objektas `Pastas` bus paverstas tekstinio failu `Pastas.matrica` (jį rasite savo darbinėje direktorijoje). Šį failą galima nusikopijuoti ir po to nusiskaityti kitoje mašinoje su `source("../Pastas.matrica",echo=T)`.

³ Mes kiek aplenkėme įvykius: simbolinio vektoriaus komponentės rašomos kabutėse.

⁴ Dar kartą aplenkiamė įvykius: `dimnames` yra sąrašas (`list`) su dviem komponentėmis – pirmoji komponentė (eilučių vardai) yra tuščia (`NULL`), o antroji nusako stulpelių vardus.

Matricos Pastas 18-oje eilutėje yra klaida – atstumas turi būti 260, o ne 1260. Ją ištaisome:

```
Pastas[18,3] <- 260 # [...] yra poaibio ženklas; [18,3] yra matricos  
# pastas elementas, esantis 18-os eilutės 3-me  
# stulpelyje
```

arba

```
Pastas[18,"atstumas"] <- 260
```

Dar keli pavyzdžiai:

```
Pastas23 <- Pastas[,2:3] # Palikome tik 2-ą ir 3-ią matricos Pastas  
# stulpelius  
Pastas23 <- Pastas[,-1] # Tas pat
```

R naudoja vektorinę (tiksliau, masyvinę) aritmetiką, t.y. operacijos atliekamos pa-komponenčiui. Pailiustruosime tai matricų pavyzdžiu.

```
m1 <- diag(rep(1,3))  
m1  
  [,1] [,2] [,3]  
[1,]  1   0   0  
[2,]  0   1   0  
[3,]  0   0   1  
m2 <- matrix(1:9,3,3)  
m2  
  [,1] [,2] [,3]  
[1,]  1   4   7  
[2,]  2   5   8  
[3,]  3   6   9
```

Šias dvi matricas galima (paelemenčiui) sudėti, sudauginti bei atlikti daug kitokių operacijų:

```
m1+m2  
  [,1] [,2] [,3]  
[1,]  2   4   7  
[2,]  2   6   8  
[3,]  3   6  10
```

$m1 * m2$ – tai Adamaro (J. Hadamard) tiesioginė (paelementė) sandauga:

```
  [,1] [,2] [,3]  
[1,]  1   0   0  
[2,]  0   5   0  
[3,]  0   0   9
```

Įprastinė matricų sandauga apskaičiuojama taip:

```
m1%*%m2  
  [,1] [,2] [,3]  
[1,]  1   4   7  
[2,]  2   5   8  
[3,]  3   6   9
```

O štai transponuota matrica m2:


```
t(m2)
      [,1] [,2] [,3]
[1,]    1    2    3
[2,]    4    5    6
[3,]    7    8    9
```

Matrica $m2$ yra singuliari (t.y. jos determinantas lygus 0 (kodėl?)) ir todėl atvirkštinės ji neturi. Antra vertus, $m2^2$ yra jau “gera” matrica, o jos atvirkštinę randame taip:

```
solve(m2^2)
      [,1] [,2] [,3]
[1,]  1.291667 -2.166667  0.9305556
[2,] -1.166667  1.666667 -0.6111111
[3,]  0.375000 -0.500000  0.1805556
```

Tai tikrai atvirkštinė matrica, kadangi

```
m2^2%%solve(m2^2)
      [,1] [,2] [,3]
[1,]  1.000000e+000  1.776357e-014 -7.105427e-015
[2,]  7.105427e-015  1.000000e+000  1.776357e-015
[3,] -1.065814e-014  1.421085e-014  1.000000e+000
```

Vaizdesnę išraišką gausime, jeigu atsakymą suapvalinsime iki, tarkime, vieno ženklo po kablelio:

```
round(m2^2%%solve(m2^2), 1)
      [,1] [,2] [,3]
[1,]    1    0    0
[2,]    0    1    0
[3,]    0    0    1
```

3.1.2. Kompleksiniai ir loginiai vektoriai

Kompleksinius vektorius rankomis įrašyti tenka retai (žr. `?complex` arba `example(complex)`). Dažniausiai tai būna kai kurių funkcijų reikšmės.

```
> roots <- polyroot(c(1,2,3))
> mode(roots)
[1] "complex"
> roots
[1] -0.3333333+0.4714045i -0.3333333-0.4714045i
```

Čia vektorius $c(1, 2, 3)$ yra polinomo $1+2*x+3*x^2$ koeficientų vektorius; funkcijos `polyroot` reikšmė yra šio polinomo (dviejų kompleksinių jungtinių) šaknų vektorius. Nesunku patikrinti, kad tai iš tikrųjų šio polinomo šaknys:

```
> 1+2*roots[1]+3*roots[1]^2
[1] 0+0i
> 1+2*roots[2]+3*roots[2]^2
[1] 1.110223e-16-4.440892e-16i # Iš tikrųjų čia 0+0i
```

Kadangi R naudoja vektorinę aritmetiką, abi šaknis galime patikrinti iš karto.

```
> round(1+2*roots+3*roots^2,2) #Suapvaliname iki 2 ženklų po kablelio
[1] 0+0i 0+0i
```

Loginiai vektoriai irgi paprastai atsiranda kaip kai kurių (palyginimo) operacijų rezultatai.

```
> x <- 1:5
> x>1
[1] FALSE TRUE TRUE TRUE TRUE
> xT <- x>1
> xT
[1] FALSE TRUE TRUE TRUE TRUE
> !xT # Ženklas ! žymi loginį NE
[1] TRUE FALSE FALSE FALSE FALSE
> x<=4
[1] TRUE TRUE TRUE TRUE FALSE
> x>1|x<=4 # Ženklas | žymi loginį ARBA
[1] TRUE TRUE TRUE TRUE TRUE
> x>1&x<=4 # Ženklas & žymi loginį IR
[1] FALSE TRUE TRUE TRUE FALSE
> x[x>1&x<=4]
[1] 2 3 4
```

3.1.3. Simboliniai vektoriai ir matricos

Jau turėjome simbolinio vektoriaus įrašymo pavyzdį:

```
simbvel <- c("kaina","svoris","atstumas")
mode(simbvel)
[1] "character"
```

Jei įrašomas ilgas vektorius – naudokitės edit komanda:

```
simbvel <- "kaina"
simbvel <- matrix(simbvel)
simbvel <- edit(simbvel) # Užpildykite 1-ąjį stulpelį
simbvel
      coll
[1,] "kaina"
[2,] "svoris"
[3,] "atstumas"
mode(simbvel)
[1] "character"
dim(simbvel)
[1] 3 1
class(simbvel)
NULL
names(simbvel)
NULL
```

Patys panagrinėkite vektorių

```
simbve2 <- c("1","2","3")
```

Aišku, kad jis skiriasi nuo vektoriaus `ve3 <- c(1,2,3)`:

```
simbve2 <- c("1","2","3")
```

```
mean(simbve2)
Error in sum(..., na.rm = na.rm) : invalid "mode" of argument
mean(ve3)
[1] 2
```

Simbolinės matricos (jų visi elementai turi būti simboliniai!) įrašinėjamos retai. Jei to prireiktų – naudokite `edit` funkciją.

3.1.4. Duomenų sistemos

Matricoje `Pastas` iš tikrųjų yra ir ketvirtas stulpelis, būtent didumas: pašto skyriuje siuntiniai dar skirstomi į didelius ir mažus. Štai tas stulpelis:

```
didumas <- c(rep("mazas",10),rep("didelis",10))
```

Skaitinę matricą `Pastas` ir simbolinę matricą(-stulpelį) `didumas` galima apjungti į naują matricą `pastas` taip:

```
pastas <- cbind(Pastas,didumas) # Funkcija cbind matricą didumas
# prirašo šalia matricos Pastas
# (funkcija rbind - po matrica)
```

Deja, visi matricos elementai turi būti vieno tipo, todėl `cbind` automatiškai paverčia visus matricos `pastas` elementus (aukštesnės hierarchijos) simboliniais kintamaisiais:

```
pastas
      kaina   svoris   atstumas   didumas
[1,]  "2.0"    "0.3"    "160"    "mazas"
.....
```

Norint, kad duomenų tipas nebūtų iškraipomas, `pastas` reikia apiforminti kaip duomenų sistemą⁵ (jame stulpeliai gali būti skirtingos prigimties):

```
pastas <- data.frame(Pastas,didumas)
pastas
      kaina   svoris   atstumas   didumas
1      2.0    0.3      160      mazas
.....
dim(pastas)
[1] 20  4
mode(pastas)
[1] "list"
class(pastas)
[1] "data.frame"
names(pastas)
[1] "kaina"   "svoris"   "atstumas" "didumas"
```

⁵ Angliškai tai vadinama `data frame`.

3.1.5. Vardiniai kintamieji (faktoriai)

didumas nėra paprastas vardų rinkinys, jį sudaro vardinio kintamojo didumas reikšmės mazas ir didelis. Norėdami tai pabrėžti, jam suteiksime specialią vadinamojo faktoriaus struktūrą:

```
> didumasf <- factor(didumas)
> didumasf
 [1] mazas   mazas   mazas   mazas   mazas   mazas   mazas   mazas
 [9] mazas   mazas   didelis didelis didelis didelis didelis didelis
[17] didelis didelis didelis didelis
Levels:  didelis mazas
```

Išoriškai skirtumas nėra didelis – didumasf reikšmės dabar rašomos be kabučių, tačiau šikart atsirado dar vienas požymis, būtent Levels:

```
> attributes(didumasf)
$levels
[1] "didelis" "mazas"

$class
[1] "factor"
```

Antra vertus, R viduje (kad būtų taupiau) faktoriai talpinami kaip skaičiai

```
> mode(didumasf)
[1] "numeric"
```

Tuo taip pat galima įsitikinti, surinkus

```
> edit(didumasf) –
```

Notepad'o lange matome

```
structure(c(2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1), .Label = c("didelis", "mazas"), class = "factor")
```

Aišku, kad įrašų lentelėse daug kartų rašyti žodžius mazas ir didelis nepatogu, juos galima užkoduoti, pvz., simboliais 0 ir 1:

```
> DIDUMAS <- c(rep(0,10),rep(1,10))
> attributes(DIDUMAS)
NULL
```

Aišku, kas DIDUMAS išoriškai skiriasi nuo didumas, tačiau R viduje faktoriai didumasf ir

```
DIDUMASF <- factor(DIDUMAS)
> attributes(DIDUMASF)
$levels
[1] "0" "1"

$class
[1] "factor"
```

skiriasi tik Label vardais:

```
> edit(DIDUMASF)
structure(c(1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 2, 2, 2, 2, 2, 2,
2, 2, 2), .Label = c("0", "1"), class = "factor")
```

Nesunku pasiekti, kad `didumasf` ir `DIDUMASF` R vidinių reprezentacijų prasme būtų vienodi. Mat Levels yra priskiriami abėcėlės arba, jeigu tai skaičiai, jų didėjimo tvarka. Pasirinkę lygius Levels taip, kad mažas atitiktų 0, o didelis – 1, pasieksime, kad abu faktoriai reikštų tą patį:

```
> didumasf <- factor(didumas,levels=c("mazas","didelis"))
> edit(didumasf)
structure(c(1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 2, 2, 2, 2, 2, 2,
2, 2, 2), .Label = c("mazas", "didelis"), class = "factor")
```

ir

```
> DIDUMASF <- factor(DIDUMAS,levels=c(0,1))
> edit(DIDUMASF)
structure(c(1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 2, 2, 2, 2, 2, 2,
2, 2, 2), .Label = c("0", "1"), class = "factor")
```

Pažymėsime, kad prijungdama prie skaitinio objekto (mūsų atveju, skaitinės matricos `Pastas`) simbolinių vektorių `didumas`, funkcija `data.frame` automatiškai paverčia jį faktoriumi:

```
pastas <- data.frame(Pastas,didumas)
attach(pastas)
class(didumas)
[1] "factor"
```

Kai kurių funkcijų argumentai turi būti faktoriai – tai joms signalizuoja, kad šis argumentas yra vardinis kintamasis, o ne etikečių rinkinys. Pavyzdžiui,

```
pastas <- data.frame(Pastas,didumas)
attach(pastas)
summary(didumas)
didelis   mazas
      10      10
```

arba

```
tapply(kaina,didumas,mean)
didelis   mazas
      8.58      4.09
```

Funkcija `tapply` skaičiuoja kintamojo kaina vidurkį visus duomenų sistemos `pastas` įrašus suskaidydama į grupes pagal faktoriaus `didumas` reikšmes. Taigi, didelių siuntinių vidutinė pristatymo kaina yra 8,58, o mažų – 4,09.

3.1.6. Ranginiai kintamieji

Kintamieji `didumasf` arba `DIDUMASF` yra vardiniai kintamieji – aišku, kad šiuo atveju skirtumas 1 - 0 (tai ekvivalentu skirtumui `didelis` - `mazas`) prasmės neturi. Antra vertus, `didelis` vistiek “didesnis” už `mazas` – kitais žodžiais, kintamąjį `didumasf` galime interpretuoti kaip ranginį. R kalboje tai galima užrašyti taip (levels reikia išvardinti didėjimo tvarka):

```
> attach(pastas)
> didumaso <- ordered(didumas, levels=c("mazas", "didelis"))
> didumaso
 [1] mazas   mazas   mazas   mazas   mazas   mazas   mazas   mazas
 [9] mazas   mazas   didelis didelis didelis didelis didelis didelis
[17] didelis didelis didelis didelis
Levels: mazas < didelis
> summary(didumaso)
  mazas didelis
     10     10
> attributes(didumaso)
$levels
[1] "mazas" "didelis"

$class
[1] "ordered" "factor"
```

3.1.7. Sąrašai

Sąrašas (list) yra pagrindinis R objektas, jis naudojamas, kai reikia apjungti skirtingos prigimties⁶ objektus į vieną naują objektą. Pažymėsime, kad daugumos R funkcijų reikšmė yra būtent sąrašas. Štai būdingas pavyzdys. Aišku, kad `kaina` (žr. matricą `Pastas`) priklauso nuo kintamojo `atstumas`. Kadangi, didėjant atstumui, `kaina` turėtų didėti, galima tikėtis tokios (regresinės) priklausomybės:

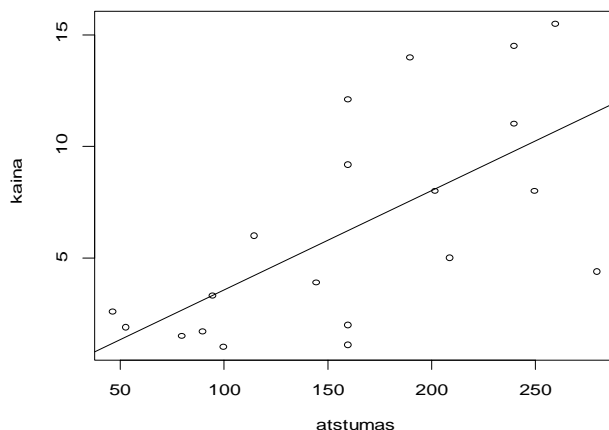
$$kaina = a + b \cdot atstumas + paklaida .$$

Nežinomus (regresijos) koeficientus a ir b , remdamasi matricos `Pastas` duomenimis, skaičiuoja R funkcija `lm`:

```
> kaina.lin <- lm(kaina~atstumas)
> mode(kaina.lin)
[1] "list" # Funkcijos lm reikšmė, t.y., sukurtas objektas kaina.lin,
        # yra sąrašas
> names(kaina.lin) # Šis sąrašas turi daug komponentų:
 [1] "coefficients" "residuals" "effects" "rank"
 [5] "fitted.values" "assign" "qr" "df.residual"
 [9] "xlevels" "call" "terms" "model"
> kaina.lin$coeff # Koeficientų komponentė
(Intercept) atstumas
-0.87284907 0.04454789
```

⁶ Viena komponentė gali būti skaitinis, o kita – simbolinis vektorius, trečia – matrica, ketvirta – kitas sąrašas ir t.t.

Taigi, koeficientas $a=-0,873$, o $b=0,045$.



3.1 pav. Kintamųjų atstumas ir kaina sklaidos diagrama bei regresijos tiesė $kaina=-0,873+0,045atstumas$

Sąrašai natūraliai atsiranda, sudarant kai kurias anketas:

```
anketa <- list(pavarde="Jonaitis Jonas", issilavinimas="magistras",
vaikai=2, vaiku.amzius=c(7,5))
```

Matome, kad sąrašo komponentės gali būti ir skirtingos prigimties ir skirtingo ilgio. Norėdami į sąrašą įtraukti kelias pavardes, elgsimės taip:

```
pavarde <- c("Jonaitis Jonas","Petraitis Antanas") # Simb. vektorius
issilavinimas <- c("magistras","vidurinis") # Simb. vektorius
vaikai <- c(2,3) # Skait. vektorius
vaiku.amzius <- list(vaiku.amzius1=c(7,5),vaiku.amzius2=c(14,12,5))
#Sąrašas
anketa <- list(pavarde=pavarde, issilavinimas=issilavinimas, vaikai
=vaikai, vaiku.amzius=vaiku.amzius) #Sąrašas
> anketa
$pavarde
[1] "Jonaitis Jonas" "Petraitis Antanas"

$issilavinimas
[1] "magistras" "vidurinis"

$vaikai
[1] 2 3

$vaiku.amzius
$vaiku.amzius$vaiku.amzius1
[1] 7 5

$vaiku.amzius$vaiku.amzius2
[1] 14 12 5
```

Sąrašo pirmoji komponentė pasiekiamos su

```
> anketa[[1]]
[1] "Jonaitis Jonas" "Petraitis Antanas"
```

(plg. anketa[1] – tai sąrašas su viena komponente) arba nurodant jos vardą (rezultatą pateiksime kaip simbolinę matricą)

```
> cbind(anketa$pavarde)
      [,1]
[1,] "Jonaitis Jonas"
[2,] "Petraitis Antanas"
```

Funkcija `unlist` paverčia sąrašą vektoriumi:

```
> unlist(anketa)
      pavarde1          pavarde2          issilavinimas1
"Jonaitis Jonas"    "Petraitis Antanas"    "magistras"
      issilavinimas2          vaikai1          vaikai2
      "vidurinis"          "2"          "3"
vaik.amzius.vaik.amzius1 vaik.amzius.vaik.amzius2 vaik.amzius.vaik.amzius1
      "7"          "5"          "14"
vaik.amzius.vaik.amzius2 vaik.amzius.vaik.amzius3
      "12"          "5"
```

Sąrašo komponentių vardus galima praleisti:

```
> unlist(anketa,use.names=F)
[1] "Jonaitis Jonas"    "Petraitis Antanas" "magistras"          "vidurinis"
[5] "2"          "3"          "7"          "5"
[9] "14"          "12"          "5"
```

Kelis sąrašus galima apjungti į vieną su paprasta apjungimo funkcija `c`:

```
list.AB <- c(list.A,list.B)
```

Daugiau apie sąrašus galima pasiskaityti [V&R, p.18], [Intro, p.27] arba [Ma, p.23].

3.2. Duomenų importas ir eksportas

Dažnai tenka apdoroti duomenis, kurie pateikti ne R formatu (pvz., Excel, SAS, SPSS ar dar kitokiu formatu). R turi paketą `foreign`, kuris gali daugumą šių duomenų importuoti (perskaityti). Lengviausia importuoti tekstinius failus⁷, kuriuos galima nuskaityti su `base` paketo funkcijomis `scan` arba `read.table`. Štai keli pavyzdžiai.

Tarkime, kad tekstiniame faile `import1.txt` yra įrašytas vektorius `1 2 3 4`. Perkelkite šį failą į R darbinę direktoriją (priminsime: ją galima sužinoti su `getwd()`) ir komandiniame lange surinkite

```
> x <- scan(file="import1.txt")
Read 4 items
> x
[1] 1 2 3 4
```

Skaitinius vektorius galima importuoti ir taip: surinkite

```
> x <- scan()
```

⁷ Dauguma programinių produktų gali eksportuoti savo duomeninius objektus į ASCII (kitaip sakant, tekstinį) formatą.

ir, spragtelėję Enter, atidarykite `import.txt`, pasižymėkite jį visą ir su Copy + Paste perkeltkite į R konsolę. Du kartus spragtelėję Enter, turėsite R vektorių `x`.

Jei šio vektoriaus koordinatės būtų viena nuo kitos atskirtos kableliu, tai rinktumė

```
x <- scan(file="import1.txt", sep=",")
```

Tarkime, failas `import2.txt` yra matricos pavidalo (pirmoje eilutėje yra kintamųjų vardai):

	wage	lnwage	educ	exper	lnexper	lneduc	male
1	313.8528	5.748924	1	23	3.178054	0	1
2	194.378	5.269804	1	15	2.772589	0	0
3	426.1364	6.05476	1	31	3.465736	0	1
4	284.0909	5.649294	1	32	3.496508	0	1

Jį importuoti galime taip⁸:

```
> x <- read.table(file="import2.txt", header=T)
> x
  wage  lnwage educ exper  lnexper lneduc male
1 313.8528 5.748924  1   23  3.178054    0    1
2 194.3780 5.269804  1   15  2.772589    0    0
3 426.1364 6.054760  1   31  3.465736    0    1
4 284.0909 5.649294  1   32  3.496508    0    1
> mode(x)
[1] "list"
> class(x)
[1] "data.frame"
```

Iš tikrųjų, failas `import2.txt` yra kompaktiniame diske R1 esančio failo `Data\Verbeek\bwages.dat` pirmos keturios eilutės. Norėdami importuoti visą šį failą (jame stulpeliai vardų neturi, todėl juos sukursime), elsimės taip⁹:

```
bwages <- read.table(file="E:/Data/Verbeek/bwages.dat", col.names=
c("wage", "lnwage", "educ", "exper", "lnexper", "lneduc", "male"))
> bwages[1:4,]
  wage  lnwage educ exper  lnexper lneduc male
1 313.8528 5.748924  1   23  3.178054    0    1
2 194.3780 5.269804  1   15  2.772589    0    0
3 426.1364 6.054760  1   31  3.465736    0    1
4 284.0909 5.649294  1   32  3.496508    0    1
> dim(bwages)
[1] 1472    7
```

Failas `bwages.dat` yra patalpintas internete, žr. <http://www.econ.kuleuven.ac.be/GME>. Jame yra 1472 Belgijos šeimų stebėjimų rezultatai. Kintamieji čia tokie:

wage – neapmokestintos šeimos nario valandinės pajamos (Belgijos frankais)

lnwage = log(wage)

educ – išsilavinimo lygis (1 – žemas, ..., 5 - aukštas)

⁸ Jei tikslaus reikalingo failo adreso neprisimenate, patogu naudoti komandą `x <- read.table(file.choose(), header=TRUE)`

⁹ Nurodant tikslų failo adresą, reikia naudoti arba Linux'o/Unix'o stiliaus kelio nuoroda su separatoriumi "/" arba Windows'iniu separatoriumi "\" (nepamirškite tikslaus adreso apsupti kabutėmis).

exper – profesinis patyrimas (metais)
lexper = log(1+exper)
leduc = log(educ)
male – 1 (jei vyras) ir 0 (jei moteris)

Žinant rinkinio internetinį adresą, failą bwages.dat galima nuskaityti ir taip:

```
read.table(file=url("http://www.econ.kuleuven.ac.be/GME/bwages.dat"))
```

R duomeninius failus galima eksportuoti į daugumą populiarių formatų. Pvz., pirmąsias dešimt R duomenų rinkinio bwages eilutes galima eksportuoti į darbinę direktoriją ASCII formatu taip (sukurtasis failas vadinsis bwages.txt):

```
write.table(bwages[1:10,],file="bwages.txt",row.names=F,col.names=F)
```

arba tiesiog

```
write.table(bwages[1:10,],file="bwages.txt")
```

(pastaruoju atveju eilutės bus sunumeruotos, o stulpeliai turės vardus).

Taigi duomenų sistemas eksportuojame su write.table, o importuojame su read.table. Norint eksportuoti vektorių, matricą arba masyvą, galima pasirinkti funkciją write. Štai pavyzdys:

```
> mm <- matrix(1:20,nrow=2)
> mm
      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10]
[1,]    1    3    5    7    9   11   13   15   17   19
[2,]    2    4    6    8   10   12   14   16   18   20
> write(t(mm),"mm.txt",ncol=10) # Matricą mm reikia transponuoti
```

Nuvairavę į darbinę direktoriją, joje rasime ASCII failą mm.txt:

```
1 3 5 7 9 11 13 15 17 19
2 4 6 8 10 12 14 16 18 20
```

Beje, įsitikinti tuo, kad šis failas buvo sėkmingai sukurtas, galime ir su

```
> file.exists("mm.txt") # Šios funkcijos reikšmė – loginis TRUE
[1] TRUE
```

Jau žinome, kad importuoti jį galima su, pvz.,

```
> matrix(scan("mm.txt"),byrow=T,ncol=10)
```

Jei šio failo nebereiks, jį galima ištrinti:

```
> file.remove("mm.txt")
[1] TRUE
> file.exists("mm.txt")
[1] FALSE
```

R objektus galime išsaugoti ir binariniu pavidalu (to gali prireikti, jei juos norėtume perkelti į kitą mašiną). Tai atliekame su funkcija `save`:

```
> save(mm, file="mm.Rdata") # Eksportuojame
> rm(mm)
> mm
Error: Object "mm" not found
> load("mm.Rdata") # Importuojame
> mm
      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10]
[1,]    1    3    5    7    9   11   13   15   17   19
[2,]    2    4    6    8   10   12   14   16   18   20
```

Beje, šitaip galima transportuoti ir kitus R objektus, pvz., funkcijas.

```
> save(mm, mean.mano, file="new.Rdata")
```

Daugiau apie importą ir eksportą galima rasti [Pa, sk. 3.2, 3.3].

3.3. (Pseudo)atsitiktinių skaičių generavimas

Realūs stebėjimų rezultatai retai elgiasi “taip kaip reikia” (pvz., dauguma statistikos modelių reikalauja, kad stebėjimai turėtų Gauso skirstinį, o tuo tarpu matavimo rezultatų histograma nelabai panaši į varpo pavidalo kreivę). Norint geriau suprasti statistikos metodus, dažnai tikslinga nagrinėti “dirbtinius” duomeninius objektus. R moka generuoti (beveik (=pseudo)) atsitiktinius skaičius, kurių histogramos, p reikšmės, modelių paklaidos ir t.t. jau elgiasi “teisingai”.

Atsitiktinius skaičius generuojančios funkcijos yra pavidalo $r + \{\text{beta}, \text{binom}, \text{norm}, \text{unif}, \dots\}$. Štai jų sąrašas (žr. [Pa, 15 p.]):

law	function
Gaussian (normal)	<code>rnorm(n, mean=0, sd=1)</code>
exponential	<code>rexp(n, rate=1)</code>
gamma	<code>rgamma(n, shape, scale=1)</code>
Poisson	<code>rpois(n, lambda)</code>
Weibull	<code>rweibull(n, shape, scale=1)</code>
Cauchy	<code>rcauchy(n, location=0, scale=1)</code>
beta	<code>rbeta(n, shape1, shape2)</code>
'Student' (t)	<code>rt(n, df)</code>
Fisher–Snedecor (F)	<code>rf(n, df1, df2)</code>
Pearson (χ^2)	<code>rchisq(n, df)</code>
binomial	<code>rbinom(n, size, prob)</code>
geometric	<code>rgeom(n, prob)</code>
hypergeometric	<code>rhyper(nn, m, n, k)</code>
logistic	<code>rlogis(n, location=0, scale=1)</code>
lognormal	<code>rlnorm(n, meanlog=0, sdlog=1)</code>
negative binomial	<code>rnbinom(n, size, prob)</code>
uniform	<code>runif(n, min=0, max=1)</code>
Wilcoxon's statistics	<code>rwilcox(nn, m, n), rsignrank(nn, n)</code>

Pradėkime nuo atsitiktinių skaičių, turinčių Puasono skirstinį, generavimo.

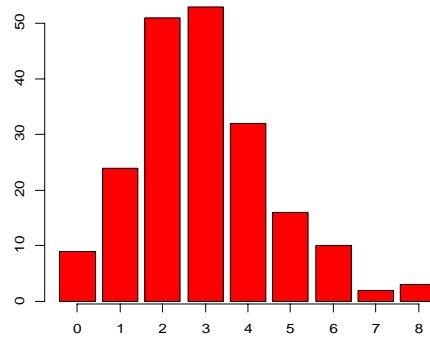
```
?rpois
rpois(20,3)
[1] 2 4 2 3 2 2 0 1 5 2 0 4 3 3 2 3 2 2 4 2 - generuojame seką dvidešimties atsitiktinių skaičių, turinčių Puasono skirstinį su vidurkiu 3;
```

```
rpois(20,3)
[1] 4 3 5 1 5 3 3 3 0 3 5 1 3 6 5 3 1 0 4 2 - šie atsitiktiniai skaičiai yra pirmosios sekos tęsinys, todėl jie kitokie;
```

```
sort(rpois(20,3))
[1] 0 0 1 2 2 2 2 2 2 2 2 2 2 3 3 4 4 5 6 6 - trečia puasoninių dydžių porcija, surūšiuota didėjimo tvarka.
```

Panagrinėkima dar vieną pavyzdį. Tarkime, kad 10 kartų šauname į taikinį, o pataikymo tikimybė yra 0,3. Aišku, kad sėkmingų šūvių skaičius yra atsitiktinis dydis, turintis binominį skirstinį su parametrais 10 ir 0,3 (jo vidurkis lygus $n \cdot p = 10 \cdot 0,3 = 3$). Imituoti tokius eksperimentus galime su funkcija `rbinom` (jos žemiau esantis variantas pateikia sėkmių skaičių kiekviename iš 200 išvaizduojamų eksperimentų):

```
> rb <- rbinom(200,10,0.3)
> rb
 [1] 0 3 1 3 0 0 2 4 4 4 5 3 1 2 4 1
[17] 5 2 3 2 5 1 0 2 2 1 2 4 3 4 8 4
[33] 3 2 2 3 0 6 4 2 3 6 3 2 3 6 0 8
[49] 1 4 1 4 4 3 5 4 2 2 3 1 5 1 3 2
[65] 6 3 3 3 1 1 5 4 1 2 2 6 2 3 2 3
[81] 2 4 3 3 6 4 2 3 2 2 4 0 4 2 2 4
[97] 3 5 2 3 6 0 2 3 1 1 2 3 1 4 2 3
[113] 4 2 2 4 2 3 2 4 5 2 3 3 2 3 4 1
[129] 3 1 1 4 3 0 3 2 5 5 4 3 3 2 2 3
[145] 4 3 3 2 6 5 7 3 3 6 3 2 5 5 2 2
[161] 2 2 3 2 3 2 3 3 4 3 5 4 1 3 7 8
[177] 5 2 3 2 2 1 3 1 4 2 2 2 4 4 3 3
[193] 6 1 3 1 4 1 5 3
> mean(rb)
[1] 2.955
> table(rb)
rb
 0  1  2  3  4  5  6  7  8
 9 24 51 53 32 16 10  2  3
> barplot(table(rb))
```



3.2 pav. Grafike matyti, keli eksperimentai baigėsi 0, 1, ... ar 10-čia sėkmių

Panašų grafiką galime gauti (pabandykite) ir su

```
plot(table(rb), type = "h", col = "red", lwd=10,
main="rbinom(200,10,0.3)")
```

Nors teoriškai sėkmių gali būti nuo 0 iki 10, tačiau matome, kad atlikus 200 eksperimentų nei karto nepataikėme 9 kartų. To priežastis yra aiški – 9 kartus pataikyti “turėjome” 0,0276 kartus, todėl nieko nuostabaus, kad tokio įvykio nestebėjome:

```
> 200*dbinom(9,10,0.3)
[1] 0.0275562
```

Beje,

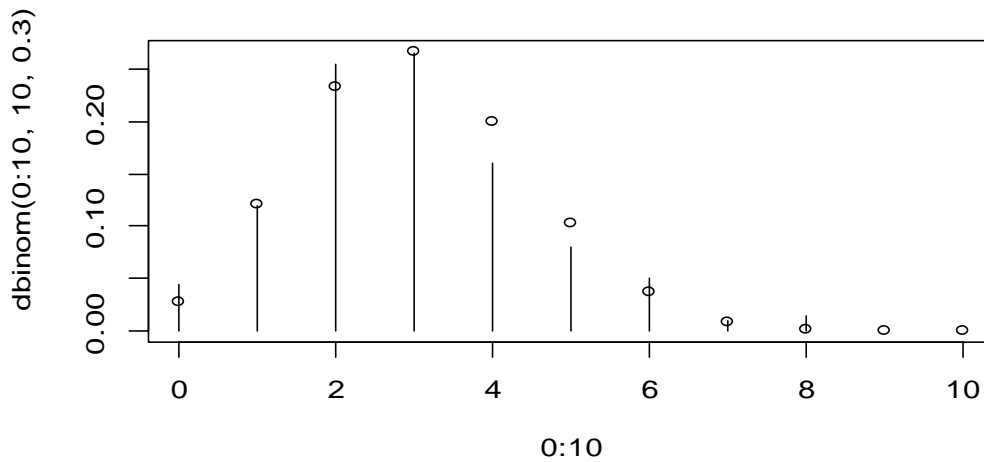
```
> 200*dbinom(3,10,0.3)
[1] 53.36559
```

kas labai arti stebėtųjų 53 kartų. Apskritai, tikimybių ir santykinių dažnių artumą galime pavaizduoti lentele

```
> round(rbind(table(rb)/200,dbinom(0:8,10,0.3)),4)
      0      1      2      3      4      5      6      7      8
[1,] 0.0450 0.1200 0.2550 0.2650 0.1600 0.0800 0.0500 0.010 0.0150
[2,] 0.0282 0.1211 0.2335 0.2668 0.2001 0.1029 0.0368 0.009 0.0014
```

arba grafiškai

```
plot(0:10,dbinom(0:10,10,0.3))
lines(as.integer(names(table(rb))),table(rb)/200,type="h")
```

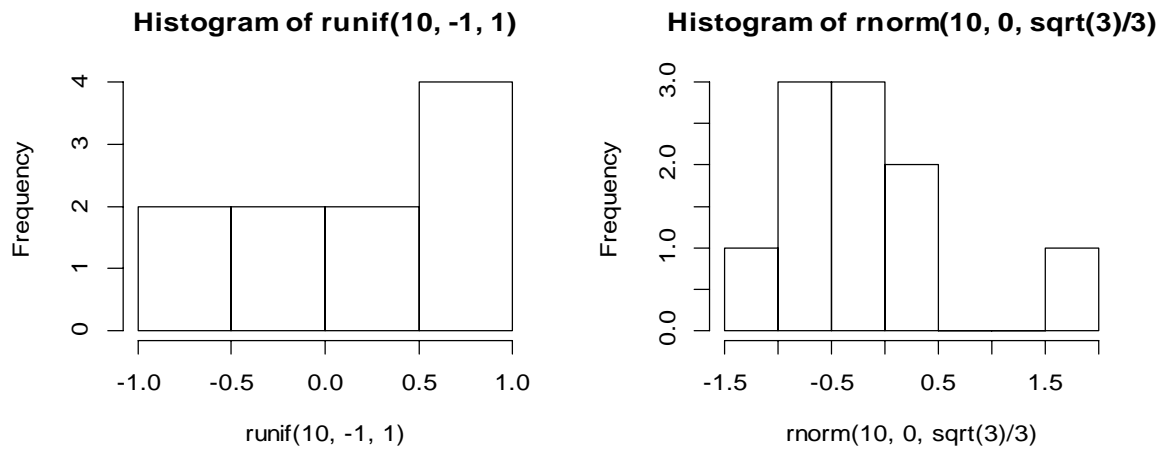


3.3 pav. Tikimybių (rutuliukai) ir santykinių dažnių (stačios atkarpos) grafikas

Jei diskrečiųjų atsitiktinių dydžių dažnius brėžiame su `barplot` komanda (arba `plot` su opcija "h"), tai tolydžiuoju atveju tam naudojame empirinį tankio atitikmenį – histogramą. Palyginkime du ėminius, kurių vienas yra tolygusis su parametrais -1 ir 1, o kitas - normalusis (Gauso) su parametrais 0 ir $\sqrt{3}/3$ (abiejų ėminių vidurkiai ir dispersijos sutampa, ar ne?):

```
par(mfrow=c(1,2)) # Grafiniame lange bus du vienoje eilutėje
                    # esantys polangiai
hist(runif(10,-1,1)) # Generuojame 10 tolygiųjų atsitiktinių
                    # skaičių ir brėžiame jų histogramą
hist(rnorm(10,0,sqrt(3)/3))
```

Matome, kad tuomet, kai imtys nedidelės, netgi skirtingų a.d. histogramos gali būti labai panašios (dvi paskutines eilutes pakartokite kelis kartus)



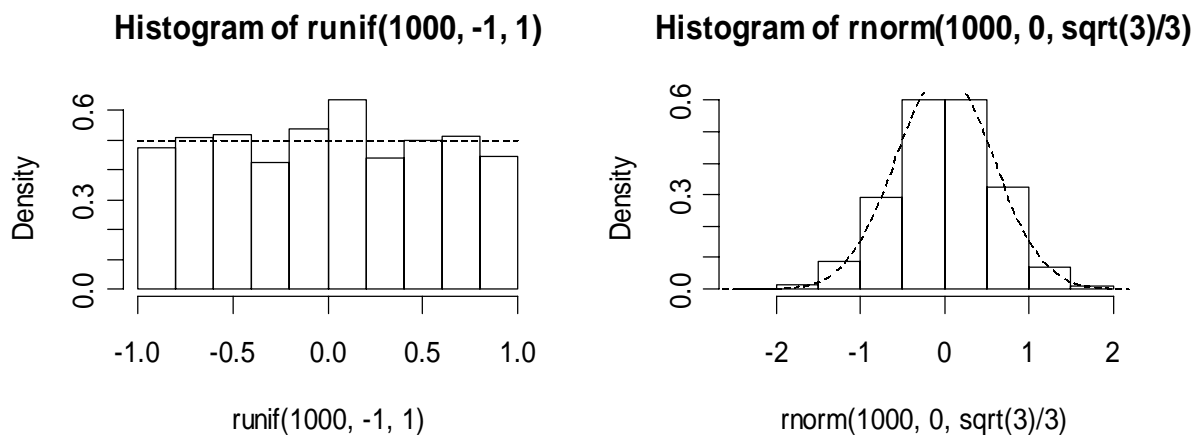
3.4 pav. 10 -ties tolygiųjų (kairėje) ir 10 - ties normaliųjų (dešinėje) atsitiktinių skaičių histogramos

Kai ėminio dydis augs, histograma artėja prie tankio: kelis kartus pabandykite

```
hist(runif(100,-1,1)) # Didėjant imties dydžiui, skirtumai
hist(rnorm(100,0,sqrt(3)/3)) # tarp histogramų ryškėja
```

o dabar pabandykite

```
hist(runif(1000,-1,1),freq=F)
x <- seq(-1,1,length=100)
lines(x,dunif(x,-1,1),lty=2)
hist(rnorm(1000,0,sqrt(3)/3),freq=F)
xx <- seq(-3,3,length=100)
lines(xx,dnorm(xx,0,sqrt(3)/3),lty=2)
```

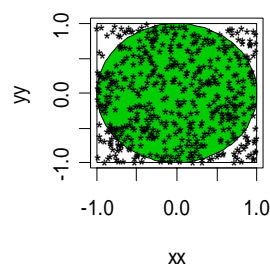


3.5 pav. 1000 tolygiųjų (kairėje) ir 1000 normaliųjų (dešinėje) skaičių histogramos ir atitinkami tankiai

3.1 pvz. Aprašysime kaip, taikant vadinamąjį Monte-Carlo metodą, galima apskaičiuoti skaičių π . Generuokime seką dvimačių atsitiktinių vektorių $\alpha_1, \alpha_2, \dots, \alpha_n$, turinčių tolygų skirstinį kvadratu K su viršūnėmis taškuose $(-1, -1)$, $(1, -1)$, $(1, 1)$ ir $(-1, 1)$ (vieną tokį vektorių generuosime su $runif(2, -1, 1)$). Žinome, kad tolygiojo skirstinio atveju tikimybė, kad taškas pakliūs į kvadrato “gerą” poaibį A yra lygi $P(A) = l(A)/l(K) = l(A)/4$ (čia $l(A)$ yra aibės A Lebego matas, t.y. tiesiog plo-

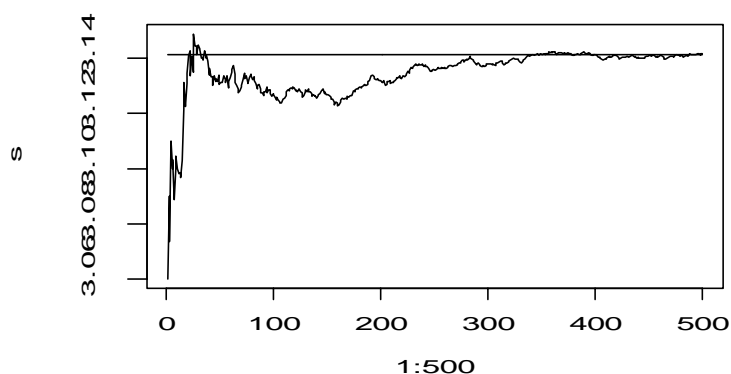
tas). Kitais žodžiais, tikimybė $P(S)$ pakliūti į vienetinį skritulį S lygi $\pi/4$. Monte Carlo metodas tvirtina, kad tuomet, kai bandymų skaičius n didelis, pakliuvusių į A taškų skaičiaus santykinis dažnis maždaug lygus $P(S)$, t.y., $\pi/4$. Štai programa, kuri leidžia apytiksliai apskaičiuoti π :

```
xx <- c(-1,1,1,-1,-1)
yy <- c(-1,-1,1,1,-1)
plot(xx,yy,type="l") # Brėžiame kvadrata K
x <- cos(seq(0,2*pi,length=100))
y <- sin(seq(0,2*pi,length=100)) # Apskritimo lygtis polinėse koord.
polygon(x,y,col=3) # Nuspalvinsime skritulį
points(runif(500,-1,1),
runif(500,-1,1),pch="*")
```



3.6 pav. Į kvadrata K "metėme" 500 taškų

```
xxx <- runif(100000,-1,1)
yyy <- runif(100000,-1,1) # Į kvadrata K "metame" 100000 taškų
s <- numeric(500)
print(s) # Vektorius iš 500 nulių
for(i in 1:500) {s[i] <-
4*sum(iffelse(xxx[1:(200*i)]^2+yyy[1:(200*i)]^2<=1,1,0))/(200*i)}
# Ciklas: kam lygus santykinis dažnis, jei jį
# skaičiuotume pagal pirmuosius 200*i taškus
print(s) # Santykinių dažnių (padaugintų iš 4) vektorius
plot(1:500,s,type="l")
lines(1:500,rep(pi,500))
s[500] # π įvertis pagal 100000 eksperimentų
[1] 3.14088
```



3.7 pav. Santykinio dažnio s elgesys ($n=200*i$, $i=1,500$)

Skaičiavimas trunka gana ilgai, o tikslumas nelabai didelis, todėl geriau π skaičiuoti kitokiais metodais¹⁰ (žinoma, jei skaičiuotume tik vieną s reikšmę, atitinkančią 100000 eksperimentų, tai skaičiavimai truktų žymiai trumpiau). Antra vertus, tai mūsų pirma tokia ilga programa. Tokias programas geriau rašyti ne komandiniame lange. Elgsimės taip. Komandiniame lange pradėkime rašyti (beargumentę) funkciją `py`:

```
py <- function(){} 
```

Toliau ją rašysime kokio nors redaktoriaus lange (standartinis Windows'inio R redaktorius yra Notepad'as): surinkite

```
py <- edit(py)
```

Notepad'o lange perrašysime (kiek pakeitę) anksčiau parašytas eilutes. Galutinis funkcijos tekstas yra toks:

```
function(){
# funkcija py (Monte Carlo metodas)
opar <- par(mfrow=c(1,2)) # Bus du polangiai
on.exit(par(opar)) # Programai baigus darba, polangiu
# bus tiek, kiek anksčiau

xx <- c(-1,1,1,-1,-1)
yy <- c(-1,-1,1,1,-1)
plot(xx,yy,type="l") # Brėžiame kvadrata K
x <- cos(seq(0,2*pi,length=100))
y <- sin(seq(0,2*pi,length=100)) # Apskritimo lygtis polinėse koordinatėse
polygon(x,y,col=3) # Nuspalvinsime skritulį
points(runif(500,-1,1),runif(500,-1,1),pch="*")
xxx <- runif(100000,-1,1)
yyy <- runif(100000,-1,1) # Į kvadrata K "metame" 100000 taškų
s <- numeric(50)
for(i in 1:50) {s[i] <-
4*sum(iffelse(xxx[1:(2000*i)]^2+yyy[1:(2000*i)]^2<=1,1,0))/(2000*i)
cat("ciklo zingsnis=",i,"\n")} # Ciklas: kam lygus santykinis dažnis, jei
# jį skaičiuotume pagal pirmuosius 2000*i
# taškus,i=1:50

plot(1:50,s,type="l")
lines(1:50,rep(pi,50))
s[50] # Įvertis pagal 100000 eksperimentų
} # rezultatus
```

Uždarykite Notepad'o langą (į klausimą Do you want to save the changes? atsakykite Yes) ir surinkę

```
py()
```

po kiek laiko pamatysite komputavimo rezultatą.

¹⁰ Visus skaičiavimus R atlieka dvigubu tikslumu:

```
> pi
[1] 3.141592653589793
```

arba

```
> options(digits=16) # Standartinė reikšmė yra digits=7
> 4*atan(1) # atan (angl.) = arktangentas
[1] 3.141592653589793
```


3.4. Apie R funkcijas ir source komandą

Jau žinome, kad R programas galima rašyti kelias būdais. Pavyzdžiui, tokią dviejų eilučių programą galima rašyti iš karto konsolėje

```
> x <- 1:10
> x
[1] 1 2 3 4 5 6 7 8 9 10
```

Šią procedūrą galima apiforminti kaip funkciją:

```
> seka <- function(){x <- 1:10;x}
> seka()
[1] 1 2 3 4 5 6 7 8 9 10
```

arba, surinkus

```
seka <- function(){}
seka <- edit(seka)
```

baigti redaguoti atsidariusiame Notepad'o lange

```
function(){
x <- 1:10
x
}
> seka()
[1] 1 2 3 4 5 6 7 8 9 10
```

Yra dar viena galimybė, kuri sudėtingesnėms funkcijoms dažnai būna patogi. Darbiniame kataloge atidarykime¹¹ naują tekstinį failą seka.txt (arba, dar geriau, seka.R). Jame parašykite dvi eilutes

```
x <- 1:10
print(x) # Ne x , bet būtent print(x)!
```

ir jo neuždare (bet išsaugoję su File|Save) R konsolės meniu eilutėje pasirinkite File|Source R code... ir, nuvairavę į darbinį katalogą, spragtelėkite ant seka.R:

```
> source("C:/Program Files/R/LabDarbai/seka.R")
[1] 1 2 3 4 5 6 7 8 9 10
```

Jei source eilutę pakeistumėte į

```
> source("C:/Program Files/R/LabDarbai/seka.R",echo=T)
```

tai matytumėte dar ir savo programos tekstą:

```
> x <- 1:10
> print(x)
[1] 1 2 3 4 5 6 7 8 9 10
```

¹¹ Tai gali būti Notepad'o arba, pvz., Word'o failas. Labai patogu redaguoti su WinEdt'u (surinkite > library(RWinEdt)).

Tekstą seka.R faile dabar galima papildyti ar redaguoti, o pakeitimus išsaugojus su File|Save vėl galima kreiptis į source procedūrą. Pažymėsime, kad šitaip patogu atlikti visus laboratorinius darbus, tekstinius failus *.R patogu parsinešti namo ir atgal.

3.5. Programavimo pavyzdžiai

3.2 pvz. R “nemėgsta” ciklą, tiksliau kalbant, kai duomenų rinkiniai dideli, vektorinės aritmetikos funkcijos pagreitina R programų darbą. Štai kelios apply grupės funkcijos:

```
apply(X,MARGIN, FUN,...) # X yra masyvas (arba matrica)
```

Jei MARGIN=1, tai funkcijos FUN argumentas bus kiekviena masyvo (pvz., matricos) X eilutė, jei MARGIN=2 – stulpelis ir t.t.

```
lapply(X,FUN,...) # X yra sąrašas (arba vektorius)
sapply(X,FUN,...,simplify=TRUE,USE.NAMES=TRUE) # Beveik tas pat
```

Taikant funkciją lapply (=list apply) arba jos labiau “user-friendly” variantą sapply, funkcijos FUN argumentas yra kiekvienas sąrašo X elementas (jei sąrašas X yra duomenų sistema, tai šio sąrašo elementai bus stulpeliai, o jei X yra vektorius, tai tiesiog kiekvienas jo elementas).

```
tapply(X,INDEX,FUN=NULL,...,simplify=TRUE) # X paprastai vektorius,
# o INDEX – faktorius
```

(jei, pvz., X yra žmogaus svoris, INDEX – jo lytis, o FUN=mean, tai tapply apskaičiuos atskirai vyrų ir moterų svorių vidurkius). Smulkiau apie kiekvieną funkciją galima sužinoti su, pvz., ?apply ir pan. Čia pateiksime kelis jų taikymo pavyzdžius.

```
set.seed(1)
a1 <- rpois(10,11)
a2 <- rpois(10,12)
a3 <- rpois(10,13)
a4 <- rpois(10,14)
am <- cbind(a1,a2,a3,a4)
# am yra dvimatis masyvas (t.y., 10x4 matrica)
am
      a1 a2 a3 a4
[1,]  7 16 12 16
[2,]  9 15 10 23
[3,]  5 11 22 13
[4,] 14 16 11 13
[5,]  9 12 13 11
[6,] 10 11 12 14
[7,] 11 11  9 17
[8,]  9 14 12 12
[9,] 12 17 11 15
[10,] 17  7 13 11

> apply(am,1,mean) # Skaičiuosime matricos eilučių vidurkius
[1] 12.75 14.25 12.75 13.50 11.25 11.75 12.00 11.75 13.75 12.00
```

Beje, jei funkcijoje `apply` funkcija `FUN` yra `sum` arba `mean`, o masyvas `X` yra labai didelis (megabaitų ar net gigabaitų eilės), tuomet tikslinga naudotis funkcijomis `rowSums`, `colSums`, `rowMeans` ar `colMeans` (esminė šių funkcijų kodo dalis parašyta C kalba, kuri skaičiuoja žymiai greičiau, nei R):

```
> rowMeans(am)
[1] 12.75 14.25 12.75 13.50 11.25 11.75 12.00 11.75 13.75 12.00
```

Matricos `am` stulpelių vidurkius galima suskaičiuoti taip:

```
> apply(am,2,mean) # Skaičiuosime matricos stulpelių vidurkius
  a1  a2  a3  a4
10.3 13.0 12.5 14.5
> lapply(am,mean) # Funkcija lapply čia netinka - ji matricą am
[[1]] # interpretuoja kaip masyvą ir skaičiuoja
[1] 7 # kiekvienos jos komponentės (susidedančios iš
# vienintelio elemento) vidurki
[[2]]
[1] 9
[[3]]
[1] 5
.....
```

Funkcija `lapply` čia netinka, kadangi ji taikoma ne matricoms, bet sąrašams (pvz., duomenų sistemoms):

```
> adf <- data.frame(a1,a2,a3,a4)
> lapply(adf,mean) # Skaičiuoja kiekvieno duomenų sistemos elemento
$a1 # (t.y., stulpelio) vidurki
[1] 10.3
$a2
[1] 13
$a3
[1] 12.5
$a4
[1] 14.5
```

Tokį patį rezultatą gauname ir su¹²

```
> sapply(adf,mean)
  a1  a2  a3  a4
10.3 13.0 12.5 14.5
```

Su funkcija `tapply` galime apskaičiuoti, tarkime, pirmojo stulpelio vidurki kiekvienoje grupėje, kurią nusako antrasis stulpelis:

```
> rbind(a1,a2)
  [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10]
a1   7   9   5  14   9  10  11   9  12  17
a2  16  15  11  16  12  11  11  14  17   7
> tapply(a1,a2,mean)
      7      11      12      14      15      16      17
17.000000  8.666667  9.000000  9.000000  9.000000 10.500000 12.000000
```

¹² Pirmų dešimties skaitmenų kvadratų vektorių galima apskaičiuoti paprastai: `sq <- (0:9)^2`, bet galima ir “sudėtingai”: `sq <- sapply(0:9,function(x)x^2)` arba net `sq <- sapply(0:9,"^",y=2)`

(grupėje su numeriu 11 yra trys įrašai: 5, 10 ir 11; jų vidurkis 8,666667).

Pateiksime dar vieną funkcijų `apply` ir `sapply` taikymo pavyzdį – iš matricos `am` pašalinkite stulpelius, kurių vidurkis mažesnis už 13. Šitai, aišku, galime atlikti rankomis

```
> am[,-c(1,3)]
      a2 a4
[1,] 16 16
[2,] 15 23
[3,] 11 13
[4,] 16 13
[5,] 12 11
[6,] 11 14
[7,] 11 17
[8,] 14 12
[9,] 17 15
[10,]  7 11
```

tačiau, jei stulpelių daug, tokį darbą geriau automatizuoti:

```
> m.am <- apply(am,2,mean)
> m.am<13
      a1      a2      a3      a4
TRUE FALSE TRUE FALSE
> am[,m.am<13]
      a1 a3
[1,]  7 12
[2,]  9 10
[3,]  5 22
[4,] 14 11
[5,]  9 13
[6,] 10 12
[7,] 11  9
[8,]  9 12
[9,] 12 11
[10,] 17 13
```

Antra vertus, jei duomenys apiforminti kaip duomenų sistema, naudosime funkciją `lapply`:

```
> m.adf <- lapply(adf, mean)
> adf[m.adf<13]
      a1 a3
1      7 12
2      9 10
3      5 22
4     14 11
5      9 13
6     10 12
7     11  9
8      9 12
9     12 11
10    17 13
```

Dar vienas pavyzdys. Sudarysime sąrašą `aList`, kurio 1-sis elementas `aList[[1]]` nurodys matricos `am` 1-osios eilutės narių lygių 11-kai numerius, 2-asis elementas `aList[[2]]` nurodys matricos `am` 2-osios eilutės narių lygių 11-kai

numerius ir t.t. (aišku, kad aList turi būti sąrašas, kadangi 11-tukų skaičius kiekvienoje eilutėje gali skirtis).

```
aList <- apply(am, 1, function(x) which(x == 11))
> aList[1:3]
[[1]]
numeric(0) # Pirmoje eilutėje 11-tukų nėra

[[2]]
numeric(0) # Antroje eilutėje 11-tukų nėra

[[3]]
a2      # Trečioje eilutėje 11 yra antroje pozicijoje
 2
```

Jei reikia, aList galima apiforminti kaip 10 atskirų skaitinių vektorių A1,..., A10:

```
for(i in 1:10) assign(paste("A", i, sep=""), aList[[i]])
> A3
a2
 2
```

Dar vienas pavyzdys. Duomenų sistemoje

```
dd <- data.frame(a=c(1,2,NA,4),b=c(NA,2,3,4))
> dd
  a b
1  1 NA
2  2  2
3 NA  3
4  4  4
```

yra trūkstamų reikšmių NA. Norint pakeisti jas 0, galima elgtis keliais būdais.

```
> dd2 <- apply(dd,2,function(x) replace(x, is.na(x), 0))
> dd2
  a b
1 1 0
2 2 2
3 0 3
4 4 4

> class(dd2)
NULL # dd2 yra matrica
> dd3 <- data.frame(apply(dd,2,function(x) replace(x, is.na(x), 0)))
> class(dd3)
[1] "data.frame" # Jei reikėtų duomenų sistemos
```

Funkcijos apply reikšmė yra matrica. Mūsų užduotį galima atlikti ir su lapply funkcija, tačiau jos reikšmė bus jau sąrašas:

```
lapply(dd,function(x) replace(x, is.na(x), 0))
$a
[1] 1 2 0 4

$b
[1] 0 2 3 4
```

Jei reikėtų, su

```
data.frame(lapply(dd,function(x) replace(x, is.na(x), 0)))
```

jį nesunku paversti duomenų sistema. Ne tokį akivaizdų, bet trumpą sprendimą galima gauti ir su

```
dd2 <- dd
dd2[] <- lapply(dd2,function(x) replace(x, is.na(x), 0))
dd2
```

3.1 UŽDUOTIS. Iš matricos `am` (duomenų sistemos `adf`) pašalinkite tuos stulpelius (variantas: eilutes), kurių maksimalus narys didesnis už 18.

3.2 UŽDUOTIS. Iš matricos `am` pašalinkite tą stulpelį (eilutę), kuriame yra didžiausias visos matricos elementas¹³.

3.3 pvz. Kintamasis `gr` (nuo grupė) nurodo grupės numerį (1 arba 2), o `a` yra skaitinis vektorius, pvz.,

```
> cbind(gr,a)
```

```
      gr  a
[1,]  1  2
[2,]  1  5
[3,]  1  9
[4,]  1  3
[5,]  2 15
[6,]  2  9
[7,]  2  7
[8,]  2 10
[9,]  2 11
```

Apskaičiuosime kiekvienos grupės dviejų didžiausių `a` elementų vidurkį (1-je grupėje tai $(5+9)/2=7$, o 2-je – $(15+11)/2=13$) (atkreipkite dėmesį, kad nei viena žemiau pasiūlyta funkcija nenaudoja ciklo).

#1

```
tapply(a, gr, function(x) mean(rev(sort(x))[1:2]))
 1  2
 7 13
```

#2

```
sapply(split(a, gr), function(x)mean(rev(sort(x))[1:2]))
```

Jei kurioje nors grupėje yra mažiau kaip du įrašai, `1:2` reikia pakeisti į `seq(min(2,length(x)))` (funkcija tuomet skaičiuos visų šios grupės `a` elementų vidurkį).

#3

```
listofdata <- split(a,gr)
lapply(listofdata, FUN=function(x)return(mean(rev(sort(x))[1:2])))
```

¹³ `which(am==max(am),arr.ind=TRUE)`

arba iš karto

```
lapply(split(a,gr),FUN=function(x)return(mean(rev(sort(x))[1:2])))
```

#4

```
foo <- function(x, n=2){  
  x.sorted <- rev(sort(x))  
  mean(x.sorted[1:n])}
```

```
tapply(a, gr, FUN=foo)
```

#5

```
gr <- rep(1:2,c(4,5))  
a <- c(2,5,9,3,15,9,7,10,11)  
asc <- tapply(a,gr,sort)  
n <- tapply(a,gr,length)  
N <- unique(gr)  
for(i in 1:length(N)){ print(mean(asc[[i]][n[i]:(n[i]-1)])) }
```

#6

```
t1 <- data.frame(gr=rep(1:2,c(4,5)),a=c(2,5,9,3,15,9,7,10,11))  
t1 <- t1[order(t1$gr,t1$a),]  
t2 <- table(t1$gr)  
t3 <- t1[sort(c(outer(cumsum(t2),0:1,"-"))),]  
tapply(t3$a,t3$gr,mean)
```

3.3 UŽDUOTIS. Sukurkite ilgesnį *gr* variantą su trimis reikšmėmis ir atitinkamą *a* vektorių. Kiekvienoje grupėje apskaičiuokite keturių mažiausių *a* elementų dispersiją (ją skaičiuoja funkcija *var*).

3.4 pvz. Kartais vektoriaus elementas arba matricos eilutė netyčia įrašomi kelis kartus. Pašalinti pasikartojančius įrašus galima su *duplicated* arba *unique* funkcijomis.

```
> x <- c(1:3,2:4)  
> x  
[1] 1 2 3 2 3 4  
> duplicated(x) # Argumentas turi būti vektorius arba duomenų sistema  
[1] FALSE FALSE FALSE TRUE TRUE FALSE  
> which(duplicated(x)==T)  
[1] 4 5 # Galima sužinoti pasikartojančių elementų numerius  
  
> xu <- x[!duplicated(x)]  
> xu # Nesikartojantys elementai  
[1] 1 2 3 4
```

Tą patį rezultatą gautume su

```
> unique(x)  
[1] 1 2 3 4
```

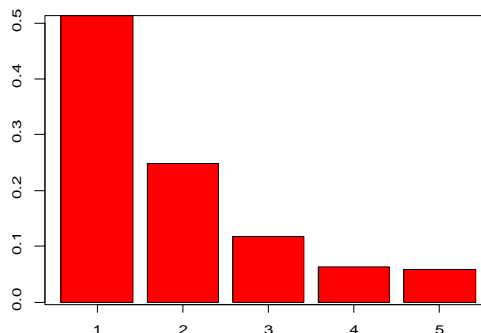
3.4 UŽDUOTIS. Sukurkite duomenų sistemą (iš determinuotų ar atsitiktinių skaičių), kuri turėtų tris stulpelius po 100000 elementų, įgyjančių reikšmes 1, 2, 3 ar 4¹⁴ (taigi

¹⁴ Tam tinka funkcija *sample*.

iš viso bus tik $4^3=64$ skirtingos eilutės (ar galite kelias pirmąsias išvardinti?). Išrinkite nesikartojančias eilutes.

3.5 pvz. Jaunavedžiai nusprendė, kad jų šeima “plėsis” tol, kol gims pirmas sūnus. Antra vertus, jie sutarė, kad bet kokių atveju maksimalus vaikų skaičius šeimoje neviršys skaičiaus MAX. Funkcija bernMC1 Monte Carlo metodu metodu įvertina tikimybę, kad šeimoje bus 1, 2, ..., MAX vaikų (berniuko gimimo tikimybė lygi 0,515).

```
bernMC1 <- function(MAX){
#bernMC
s.nr <- numeric(10000)
for(i in 1:10000){
s <- 1
while(runif(1)> 0.515)s <- s+1
s.nr[i] <- min(s,MAX)}
tab.s.nr <- table(s.nr)
print(tab.s.nr/10000)
barplot(tab.s.nr/10000)
mean(s.nr)
}
```



3.8 pav. Tikimybių įverčiai

```
> bernMC1(5)
s.nr
      1      2      3      4      5
0.5134 0.2486 0.1171 0.0624 0.0585
[1] 1.904 # Vidutinis vaikų skaičius šeimoje
```

Funkcija while yra sąlyginio ciklo funkcija (žr. ?"while"). Pažymėsime, kad tikimybę, kad tokia “politika” besivadovaujančioje šeimoje bus k vaikų, nesunku apskaičiuoti ir “teoriškai”: jei $MAX=5$, tai šios tikimybės atitinkamai lygios 0.5150, 0.2498, 0.1211, 0.0588 ir 0.0553 (įsitikinkite!).

3.5 UŽDUOTIS. Parašykite funkciją bernMC2, kuri pateiktų panašų atsakymą tuomet, kai šeima nutarusi turėti du berniukus. Koks šiuo atveju vidutinis berniukų ir mergaičių skaičius šeimoje? *Nuoroda.* Jei $MAX>2$, p yra berniuko gimimo tikimybė, o q – mergaitės, tai $P(\text{šeimoje yra } k \text{ vaikų}) = (k-1)p^2q^{k-2}$, $k=2, \dots, MAX-1$. Kam lygi tikimybė, kad šeima turės MAX vaikų? Kam lygi tikimybė, kad šeimoje bus MAX-2 mergaitės?

3.6 UŽDUOTIS. Matricoje m1 kai kurios eilutės yra sudarytos iš vienodų skaičių. a) Sugalvokite, kaip galima generuoti tokią matricą, b) Sudarykite matricą m2, kurioje nebūtų tokių eilučių. *Nuoroda.* Išmeskite eilutes, kurių $sd==0$.

3.7 UŽDUOTIS. Tarkime, vektoriaus x komponentės yra 1, 3, 5, 7 ir 9, o vektoriaus y komponentės yra 11, 13, 5, 6 ir 2. Atspėkite, koks bus šių operacijų rezultatas? 1) $x-1$; 2) y^2 ; 3) $\text{length}(x)$; 4) $\text{length}(x+y)$; 5) $\text{sum}(x \geq 4)$; 6) $\text{sum}(x[x \geq 4])$; 7) $\text{sum}(x > 5 | x < 2)$ (žr. ?" |"); 8) $y[4]$; 9) $y[-4]$; 10) $y[x]$.

3.8 UŽDUOTIS. Štai autoriaus bute esančio dujų skaitiklio mėnesiniai parodymai (nuo 1997.12.20 iki 1999.01.20): 18000, 18350, 18650, 19000, 19100, 19222, 19350, 194000, 19550, 19810, 20100, 20500, 20850 (m³). 1) Įrašykite šiuos skaičius į R duomenų rinkinį `gas`; 2) Apskaičiuokite kiekvieno mėnesių dujų sunaudojimą (galite pasinaudoti funkcija `diff`; šį vektorių pavadinkite `dujos`); 3) raskite maksimalų ir minimalų sunaudotų dujų per mėnesį kiekį (funkcijos `max` ir `min` arba `range`); 4) kokiais mėnesiais tai buvo (funkcijos `which.max` ir `which.min`)? ; 5) sukurkite duomenų sistemą `Dujos`, kurios pirmas stulpelis būtų mėnesio vardas, o antras – vektoriūs `dujos` (arba suteikite vardus vektoriaus `dujos` komponentėms); 6) kuriais mėnesiais dujų sunaudojimas viršijo 150 m³?; 7) kiek buvo tokių mėnesių? 8) išbrėžkite mėnesinio dujų sunaudojimo grafiką (funkcija `plot(..., type="l")`); 9) metų pradžioje dujų kaina buvo 0,45 Lt/m³, o nuo rugpjūčio mėn. – 0.59 Lt/m³; sukurkite duomenų sistemą `Dujos.mok`, kurios pirmi du stulpeliai būtų `Dujos`, o trečias – mėnesinis mokestis už dujas; 10) išbrėžkite mokamos sumos grafiką.

3.9 UŽDUOTIS. Atlikite tokią pat savo buto mokesčių analizę (jei reikia, dujas galite pakeiti elektra).

3.10 UŽDUOTIS. Sukurkite aritmetinę ir geometrinę progresijas (aritmetinę (geometrinę) progresiją pateikite kaip funkcijos `ar.pr(n, a1, d)` (atitinkamai, `geo.pr(n, b1, q)`) reikšmę). Viename grafike išbrėžkite dalinių progresijos sumų ir pačios progresijos kreives. Progresijas sukurkite trim būdais: apibrėždami jas rekurentiškai, naudodami bendrojo nario formulę ir naudodami R vektorinę aritmetiką (pvz., progresiją 1, 2, 4, 8 galima užrašyti kaip $2^{(0:3)}$).

3.11 UŽDUOTIS. Sukurkite vektorių iš vieno 1, dviejų 2, ..., devynių 9 (naudokite funkciją `rep`). Šiuos duomenis apiforminkite kaip 5×9 matricą. Apskaičiuokite kiekvienos eilutės vidurkį.

3.12 UŽDUOTIS. Parašykite funkciją, kuri sukurtų n narių Fibonačio seką (priminsime: tai seka, tenkinanti sąlygą $a_1 = 0, a_2 = 1, a_{k+2} = a_{k+1} + a_k$). Išbrėžkite a) jos grafiką, b) paskutiniojo skaitmens reikšmių histogramą ir c) pirmojo skaitmens reikšmių histogramą (o gal geriau taikyti `barplot` funkciją?). *Nuoroda.* Paskutinią natūraliojo skaičiaus x skaitmenį galima apskaičiuoti bent dviem būdais:

```
x-floor(x/10)*10 arba x%%10 # Dalyba moduliu 10
```

(kaip galima rasti paskutinius du skaitmenis?), o pirmąjį realiojo skaičiaus $x \geq 1$ skaitmenį, pvz., taip:

```
floor((x/10^(floor(log(x,base=10))+1))*10)
```

3.13 UŽDUOTIS. Apskaičiuokite kiekvieno duomenų sistemos `USJudgeRatings` (iš `base` paketo) stulpelio minimumą, maksimumą, plotį, vidurkį ir medianą. Nustatykite ir atspausdinkite eilutes, kuriose yra ketvirto stulpelio maksimumas (minimumas). Išbrėžkite visų stulpelių grafikus (išsiaiškinkite tokias komandas:

```
par(mfrow=c(3,4));apply(USJudgeRatings,2,plot,type="l")
```

3.14 UŽDUOTIS. Pabandykite atspėti, koks bus šių komandų rezultatas:

- a) `x <- c(4,8,1,7,15,4,8)`
`for (i in 2:length(x)) x[i] <- max(x[i],x[i-1])`
`x[length(x)]`
- b) `x <- c(4,8,1,7,15,4,8)`
`for (i in 2:length(x)) x[i] <- sum(x[i],x[i-1])`
`x[length(x)]`

3.15 UŽDUOTIS. Štai duomenų sistemos `cabbages` pradžia:

```
> library(MASS)
> data(cabbages)
> cabbages[1:3,]
  Cult Date HeadWt VitC
1  c39  d16    2.5   51
2  c39  d16    2.2   55
3  c39  d16    3.1   45
```

Nustatykite visų stulpelių klases. Faktoriams atspausdinkite jų lygius. Nustatykite ir atspausdinkite eilutę, kurioje `VitC` kiekis didžiausias. Apskaičiuokite `VitC` vidurkį kiekvienoje `Cult` grupėje.

3.16 UŽDUOTIS. Pateikite išsamų duomenų sistemos `anorexia` iš `MASS` paketo aprašymo. Apiforminkite kaip `Word`'o dokumentą. Gal būt grafikai padėtų nustatyti, kuris gydymo metodas efektyvesnis?

3.17 UŽDUOTIS. Iš paketo `MASS` duomenų rinkinio `Cars93` išrinkite tik tuos duomenis, kurie susiję su `small` ir `sporty` automobiliais. Kiekvienoje iš šių grupių apskaičiuokite parametro `MPG.highway` vidurkį bei standartinę nuokrypį ir išbrėžkite histogramas.

3.18 UŽDUOTIS. `MASS` paketo duomenų rinkinyje `Cars93` pašalinkite `small` ir `sporty` automobilių įrašus. Gautojoje duomenų sistemoje pašalinkite automobilius, kurių svoris `weight` didesnis už 3000 (svarų) ir cilindų `Cylinders` skaičius didesnis už 5.

3.19 UŽDUOTIS. `MASS` paketo duomenų rinkinyje `airquality` yra daug trūkstamų duomenų (jie žymimi `NA=Not Available`). Juos visus pakeisti nuliais nėra sunku:

```
x <- as.matrix(airquality)
x[is.na(x)] <- 0
```

Štai dvi šios operacijos variacijos: 1) bet kuriame `airquality` stulpelyje esančius `NA` pakeiskite atitinkamo stulpelio vidurkiu (jums, gal būt, prireiks opcijos `mean(x,na.rm=TRUE)`); 2) pašalinkite visas `airquality` eilutes turinčias `NA`.

3.20 UŽDUOTIS. Iš `R` konsolės, spragtelėkite ant `File|Source R Code...` ir nuvairuokite į disko `R1` direktoriją `Data|Maindonald`. Pasirinkite `orings.R` failą ir spragtelėkite ant `Open`. Jei dabar `R` komandiniame lange surinksite `orings`, tai pamatysite

duomenis apie JAV kosminės šaudyklės O žiedų defektus (priminsime, kad būtent dėl šių žiedų defekto 1986 m. erdvėlaivis kildamas sprogo).

```
> orings
  Temperature Erosion Blowby Total
1           53         3      2      5
2           57         1      0      1
3           58         1      0      1
4           63         1      0      1
*****
```

Koks yra sukurtojo duomeninio objekto `orings` tipas? klasė? Išbrėžkite bendro defektų skaičiaus `Total` priklausomybės nuo `Temperature` grafiką. Kai kurie taškai “sulimpa”, todėl pabandykite funkcijas `jitter` ir `sunflowerplot` (plg. 5.2 pav.). Ar tikrai `Total` lygus `Erosion` ir `Blowby` sumai?

Atsakant į paskutinį klausimą, naudingas galėtų būti toks pavyzdys:

```
> X <- data.frame(a=1:5, b=2:6, ce=c(3,5,7,6,11))
> X
  a b ce
1 1 2  3
2 2 3  5
3 3 4  7
4 4 5  6
5 5 6 11
> X[X$a + X$b != X$ce, ] # Nustatome, kuriose eilutėse ce#a+b
  a b ce
4 4 5  6
```

Variantas:

```
> subset(X, a+b!=ce)
  a b ce
4 4 5  6
```

Beje, tik ką aprašytos skaičių palyginimo procedūros tinka dirbant su sveikais skaičiais, bet nėra tinkamos realiems skaičiams (tai susiję su vidinėmis kompiuterių problemomis).

```
> 1 + 2
[1] 3
> 1 + 2 == 3
[1] TRUE
> 1.1 + 2.2
[1] 3.3
> 1.1 + 2.2 == 3.3
[1] FALSE
```

Todėl, jei reiktų palyginti realius skaičius, elkitės taip:

```
tolerance <- .Machine$double.eps ^ 0.5 # Žr. Machine() ir ?Machine
X[abs(X$a + X$b - X$ce) > tolerance,]
  a b ce
4 4 5  6
```

3.21 UŽDUOTIS. Priminsime, kad kiekvienas R objektas gali turėti vieną ar kelis (išorinius) požymius. Štai pavyzdys.

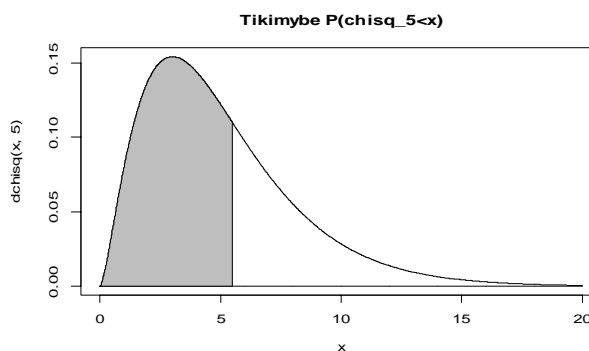
```
> x <- 1:9
> x
[1] 1 2 3 4 5 6 7 8 9
> attributes(x)
NULL
> attributes(x) <- list(Komentaras="Sudaryta 2002.03.24", names =
letters[1:9])
> x
a b c d e f g h i
1 2 3 4 5 6 7 8 9
attr(,"Komentaras")
[1] "Sudaryta 2002.03.24"
> attributes(x)
$Komentaras
[1] "Sudaryta 2002.03.24"
$names
[1] "a" "b" "c" "d" "e" "f" "g" "h" "i"
```

Kokius požymius turi duomenų rinkinys `orings` (žr. 3.20 užduotį)? Papildykite juos naujais. Pakeiskite `orings` klasę. Ar pasikeitė požymiai?

3.22 UŽDUOTIS. Statistikos vadovėliuose ar uždavinyuose paprastai būna kai kurių tikimybių skirstinių lentelės. Parašykite funkciją, kuri “gražiu pavidalu” pateiktą pageidaujamo skirstinio lentelę. Pvz., chi kvadrato su 5 laisvės laipsniais skirstinio atveju, tikimybių $P(\chi_5^2 < x)$ (jas skaičiuoja funkcija `pchisq(x, 5)`) lentelė turėtų atrodyti maždaug taip:

	0.000	0.001	0.002	0.003	...	0.009
0.00	0.00e+00	1.68e-09	9.51e-09	2.62e-08	...	4.07e-07
0.01	5.30e-07	6.72e-07	8.35e-07	1.02e-06	...	2.63e-06
0.02	2.99e-06	3.37e-06	3.79e-06	4.23e-06	...	7.54e-06
.....						
1.02	0.039063	0.039146	0.039228	0.039311	...	0.039807
.....						

Puslapį su šia lentele papildykite atitinkamo skirstinio tankio grafiku – aptartuoju atveju jis galėtų atrodyti šitaip:



3.9 pav.
Chi kvadrato su 5 l.l. tankio grafikas

3.23 UŽDUOTIS. Kaip žinia, Vandermondo (A.T. Vandermonde) matricos

$$\begin{pmatrix} 1 & 1 & 1 & \dots & 1 \\ s_1 & s_2 & s_3 & \dots & s_n \\ s_1^2 & s_2^2 & s_3^2 & \dots & s_n^2 \\ \dots & \dots & \dots & \dots & \dots \\ s_1^{n-1} & s_2^{n-1} & s_3^{n-1} & \dots & s_n^{n-1} \end{pmatrix}$$

determinantas lygus $\prod_{1 \leq i < j \leq n} (s_i - s_j)$. Apskaičiuokite šios matricos determinantą dviem būdais: naudodamiesi a) R paketo funkcija `det` ir b) savo parašyta funkcija `vander`, kuri apskaičiuotų nurodytą sandaugą bet kokiam tarpusavyje nelygių skaičių (s_1, s_2, \dots, s_n) rinkiniui.

3.24 UŽDUOTIS. Įvykdykite šias komandas:

```
nn <- rnorm(100000)
j <- 0
for(i in 1:100000) {if (nn[i]>1) j <- j+1}
```

Jei jums atsibodo laukti programos pabaigos, sustabdykite programos vykdymą su Esc klavišu ir pakeiskite antrą ir trečią eilutę tokia (tai vektorinės aritmetikos pavyzdys):

```
j <- sum(iffelse(nn>1,1,0))
```

Kokia žemiau pateiktų dviejų skaičių tikimybinė prasmė?

```
cbind(j/100000,1-pnorm(1))
```

13.25 UŽDUOTIS. Sukurkite vektorių `vec2`, įterpdami po kiekvienos vektoriaus `vec1` komponentės vienetuką. Pvz., jei `vec1 <- c(2, 3, 4)`, tai `vec2` turėtų būti `2,1,3,1,4,1`.

Pateiksime kelis sprendimo variantus.

- ```
1) vec2 <- rep(1, 2*length(vec1))
 vec2[seq(1, length(result), 2)] <- vec1
 vec2
```
- ```
2) vec1 <- c(2,3,4)
   vec2 <- rep(1,length(vec1)*2)
   vec2[seq(1,length(vec1)*2-1,by=2)] <- vec1
   vec2
```
- ```
3) vec2 <- c(t(cbind(vec1,rep(1,length(vec1))))))
 vec2
```
- ```
4) AddValue <- function(vec,addvalue=1)
   {
     tmp <- cbind(vec,addvalue)
     tmp <- as.vector(t(tmp))
   }
```

```

return(tmp[1:(length(tmp))])
}
vec2 <- AddValue(vec1)
vec2

```

5) `vec2 <- c(eval(parse(text=paste("c(",paste(vec1,collapse=" ",1,"),")"))),1)`
`vec2`

ar net fantastišką

6) `vec2 <- c(rbind(vec1,1)) # Matricos nuskaitomos stulpeliais`
`vec2`

Išsiaiškinkite visus sprendimus ir parašykite bent du variantus, kuriuose `vec3` būtų `vec2` be paskutinio vieneto.

3.26 UŽDUOTIS. Atsitiktiniai dydžiai X_1, X_2, \dots yra nepriklausomi ir turi $[0,1]$ tolygų skirstinį. Tarkime, Y yra a.d., lygus tam k , kai suma $S_k = X_1 + \dots + X_k$ pirmą kartą viršija 1. Patikrinte žinomą faktą, kad $EY = e$. *Nuoroda.* Štai funkcija, kuri generuoja Y realizacijas, kai X 'sai turi $[0,1/3]$ tolygų skirstinį:

```

> my.foo <- function(){
s <- 0
ss <- 0
while(ss<1){
s <- s+1
a <- runif(1,0,1/3)
ss <- ss+a
print(c(s,ss))
}
s
}

```

3.27 UŽDUOTIS. Interneto svetainėje <http://lark.cc.ukans.edu/~pauljohn/R/> arba kompaktiniame diske R1 (...\\Knygos_apie_R&S\PaulJohnson) yra failas StatsRUs.htm, kuriame surinkti trumpi atsakymai į įvairius klausimus (1.1-1.14, 2.1-2.21, 3.1-3.21, 4.1-4.8, 5.1-5.39, 6.1-6.4, 7.1-7.26, 8.1-8.24, 9.1-9.8, 10.1-10.7, 11.1-11.8, 12.1-12.4, 13.1-13.3). Išsiaiškinkite kelis nurodytus klausimus ir atsakymus. Pateikite savus išnagrinėtos problemos taikymo pavyzdžius.

3.28 UŽDUOTIS. Sugeneruokime 2000 Cauchy atsitiktinių skaičių (internete paieškite informacijos apie Cauchy skirstinį) ir paimkime jų sveikąsias dalis:

```

RC <- rcauchy(2000)
rc <- floor(RC)

```

Ar yra šiame vektoriuje reikšmė 347? 13? O praleistosios reikšmės simbolis NA?

Šį uždavinį galima spręsti įvairiai.

- 1) `sort(rc)`
- 2) `table(rc)`
- 3) `barplot(table(rc))`

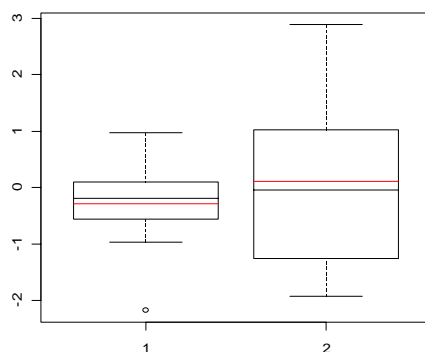
- 4) `which(rc==347)`
- 5) `length(which(rc==347))`
- 6) `347 %in% rc`
- 7) `rc[rc==347]`
- 8) `length(rc[rc==347])` # NA atveju galima elgtis taip:
`length(rc[is.na(rc)])`

Išsiaiškinkite kiekvieną iš šių būdų. Kuris jums priimtinausias? O štai šio uždavinio modifikacija: kelios duomenų rinkinio RC reikšmės priklauso intervalui [0,50]?

3.29 UŽDUOTIS. Programą

```
set.seed(1)
a <- rnorm(10)
b <- rnorm(10)
boxplot(a, b)
```

papildykite viena eilute taip, kad ji brėžtų žemiau pateiktą grafiką (čia raudona linija žymi vidurkius; gali būti naudinga funkcija `segments`):



3.10 pav. Stačiakampės diagramos su vidurkio linija

3.30 UŽDUOTIS. Šio konspekto autorius kiekvienos grupės studentams visų keturių laboratorinių darbų užduotis paskirstė atsitiktinai. Štai pagalbinės programos kompiuterinio rezultatas (vietoje pavardžių pav kol kas įrašytos angliškos raidės):

	pav	lab11	lab12	lab21	lab22	lab3	lab4
1	a	3.22	4.1	5.11	6.14	8.5	11.3
2	b	3.20	4.15	5.3	6.23	7.5	9.1
3	c	3.3	4.1	5.21	6.14	7.7	10.9
4	d	3.18	4.19	5.9	6.25	7.2	9.2
5	e	3.22	4.23	5.4	6.19	7.6	10.16

Parašykite šią programą. *Nuoroda.* Jums gal būt pravers `paste` ir `sample(..., replace=TRUE15)` funkcijos.

¹⁵ Natūralu būtų uždavinius parinkti su `replace=FALSE` (ką tai reiškia?), tačiau kai kuriuose skyriuose uždavinių yra mažiau negu studentų.

Kompakto R1 faile DataMisc\Eko21.xls yra 2002/2003 m.m. ekonometrijos programos 2 kurso 1 gr studentų sąrašas. Štai jo ištrauka:

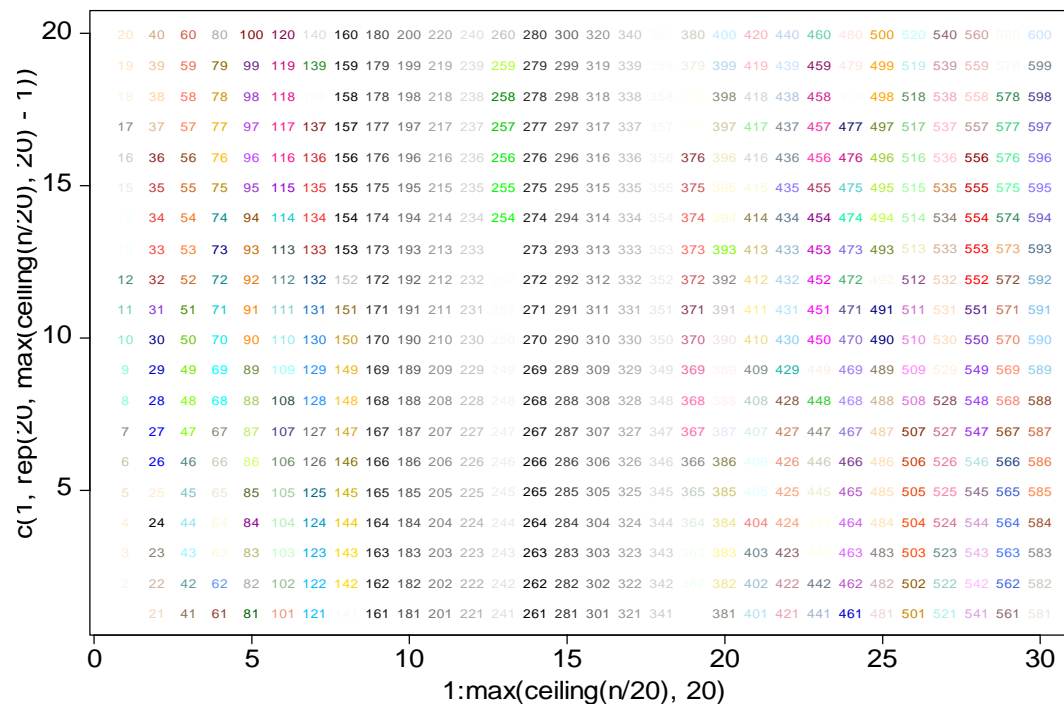
Baltramaitytė Inga
 Bendžiūnaite Laima

 Vaičiulionis Justas
 Vasilevskij Edvard

Nusiskaitykite šį sąrašą ir patobulinkite aukščiau parašytą programą taip, kad jos rezultatas būtų toks:

	Pavarde	Vardas	lab11	lab12	lab21	lab22	lab3	lab4
1	Baltramaityte	Inga	3.14	4.18	5.18	6.8	8.2	9.6
2	Bendziunaite	Laima	3.13	4.7	5.13	6.24	8.2	10.6
.....								
22	Vaiciulionis	Justas	3.25	4.23	5.22	6.24	7.7	11.9
23	Vasilevskij	Edvard	3.5	4.6	5.1	6.14	7.2	11.3

3.31 UŽDUOTIS. R spalvų paletę sudaro net 657 spalvos (žr. `colors()`). Štai pirmųjų 600 spalvų lentelė (popierinėje šio konspekto versijoje bus matomi tik pilkos spalvos atspalviai):



3.11 pav. R spalvų paletė

Parašykite programą, kuri išbrėžtų šią lentelę (tam pravers funkcija `text`; taip pat atidžiai išnagrinėkite tekstus prie 3.11 pav. ašiu).

3.32 UŽDUOTIS. Kaip žinia,

$$\int_0^{\pi} \sin x dx = 2$$

Tą patį rezultatą gautume su integrate funkcija:

```
> integrate(sin,0,pi)
2 with absolute error < 1.1e-14
```

O štai funkcija, kuri apytiksliai apskaičiuos šį integralą pagal trapecijų taisyklę:

```
> trap.rule <- function(x,y) sum(diff(x)*(y[-1]+y[-length(y)]))/2
> x <- seq(0,pi,length=100); y <- sin(x)
> trap.rule(x,y)
[1] 1.999832
```

Parašykite funkciją, kuri skaičiuotų integralą pagal Simpsono formulę:

$$\int_a^b f(x)dx \approx \frac{h}{3}(f(a) + f(b) + 2(f(x_2) + f(x_4) + \dots + f(x_{n-2})) + 4(f(x_1) + f(x_3) + \dots + f(x_{n-1})));$$

čia $x_j = a + jh, j = 0, 1, \dots, n$.

3.33 UŽDUOTIS. Žemiau yra pateiktos keturios programos, kurios sukuria skaitinį vektorių 6, 7, ..., 10, 16, 17, ..., 20, ..., 9996, 9997, ..., 10000:

```
> nums <- 1:10000
> xx <- nums[is.element(nums - 10*floor(nums/10), c(0,6:9))]
> xx[1:50]
 [1]  6  7  8  9 10 16 17 18 19 20 26 27 28 29 30
[16] 36 37 38 39 40 46 47 48 49 50 56 57 58 59 60
[31] 66 67 68 69 70 76 77 78 79 80 86 87 88 89 90
[46] 96 97 98 99 100

> xx <- unlist(matrix(1:10000,byrow=T, ncol=10)[,6:10])
> xx <- outer(-4:0,seq(10,10000,10), "+")
> xx <- outer(10 * 0:999, 6:10, "+")
```

Išsiaiškinkite šias programas ir parašykite keturias funkcijas (paremtas aukščiau pateiktais xx generavimo principais), kurios generuotų 10000 normalių atsitiktinių dydžių ir penkiuose šimtuose atsitiktinai parinktose vietose iš xx pakeistu šias reikšmes simboliais NA.

Štai du tokių (teisybė, kitokiais xx generavimo principais pagrįstų) funkcijų pavyzdžiai:

```
replac <- function(){
rn <- rnorm(10000);ind <- 6:10;Ind <- 6:10
for(i in 1:999) Ind <- c(Ind,ind+10*i)
rn[sample(Ind,500,repl=FALSE)]<- NA
rn }
```

ir

```
x <- matrix(rnorm(10000), 10);x[6:10, ][sample(1:5000, 500)] <- NA
x <- as.vector(x)
```

3.34 UŽDUOTIS. Jei dirbtume “su pieštuku”, matricos A kėlimas¹⁶ aukštu laipsniu yra gana daug laiko (o, jei naudotumėmes R paketu, ir komplikotų žymėjimų) reikalaujanti operacija:

```
> A <- cbind(c(0,0.5,0.5),c(0.5,0,0.5),c(0.5,0.5,0))
> A
      [,1] [,2] [,3]
[1,]  0.0  0.5  0.5
[2,]  0.5  0.0  0.5
[3,]  0.5  0.5  0.0
> A%%A # Tai matricų (matricinė) daugyba
      [,1] [,2] [,3]
[1,] 0.50 0.25 0.25
[2,] 0.25 0.50 0.25
[3,] 0.25 0.25 0.50
> A%%A%%A
      [,1] [,2] [,3]
[1,] 0.250 0.375 0.375
[2,] 0.375 0.250 0.375
[3,] 0.375 0.375 0.250
> A%%A%%A%%A
      [,1] [,2] [,3]
[1,] 0.3750 0.3125 0.3125
[2,] 0.3125 0.3750 0.3125
[3,] 0.3125 0.3125 0.3750
> A%%A%%A%%A%%A
      [,1] [,2] [,3]
[1,] 0.31250 0.34375 0.34375
[2,] 0.34375 0.31250 0.34375
[3,] 0.34375 0.34375 0.31250
```

Štai dvi R funkcijos, kurios palengvina šį darbą:

1) Jei kvadratinė matrica A turi skirtingas tikrines reikšmes, tai ją galima užrašyti taip: $A = V \%*\% diag(\lambda) \%*\% V^{-1}$ - tai vadinamasis matricos A spektrinis dėstiny; čia V yra matricos A tikrinių vektorių matrica, $\%*\%$ - matricų daugybos ženklas, $diag(\lambda)$ - įstrižaininė matrica su matricos A tikrinėmis reikšmėmis ant įstrižainės, o V^{-1} - matricos V atvirkštinė matrica. Tuomet $A^n = V \%*\% diag(\lambda^n) \%*\% V^{-1}$. Pvz.,

```
> A <- cbind(c(0,0.5,0.5),c(0.5,0,0.5),c(0.5,0.5,0))
> A
      [,1] [,2] [,3]
[1,]  0.0  0.5  0.5
[2,]  0.5  0.0  0.5
[3,]  0.5  0.5  0.0
> eA <- eigen(A) # eigen (angl.,vok.) = tikrinis
> eA
$values
[1]  1.0 -0.5 -0.5
```

¹⁶ Turime galvoje įprastinį, matricinį kėlimą laipsniu, o ne paelementį, kurį R atlieka labai lengvai.

```

$ectors
      [,1]      [,2]      [,3]
[1,] 0.5773503 0.5957165 -0.5583803
[2,] 0.5773503 -0.7814298 -0.2367155
[3,] 0.5773503 0.1857133 0.7950957

> round(eA$ectors**diag(eA$values)**solve(eA$ectors),5) # =A
      [,1] [,2] [,3]
[1,] 0.0 0.5 0.5
[2,] 0.5 0.0 0.5
[3,] 0.5 0.5 0.0

> A**A**A**A**A # =A^5
      [,1] [,2] [,3]
[1,] 0.31250 0.34375 0.34375
[2,] 0.34375 0.31250 0.34375
[3,] 0.34375 0.34375 0.31250
> round(eA$ectors**diag(eA$values^5)**t(eA$ectors),5)
      [,1] [,2] [,3]
[1,] 0.31250 0.34375 0.34375
[2,] 0.34375 0.31250 0.34375
[3,] 0.34375 0.34375 0.31250

```

2) Antroji funkcija puikiai demonstruoja R rekurentinių procedūrų galimybes:

```

mp <- function (X, p) if (p == 1) return(X) else X ** Recall(X, p-1)
mp(A,5)
      [,1] [,2] [,3]
[1,] 0.31250 0.34375 0.34375
[2,] 0.34375 0.31250 0.34375
[3,] 0.34375 0.34375 0.31250

```

Tikimybių teorijoje vadinamosios Markovo grandinės yra nusakomos pradinių tikimybių skirstiniu ir perėjimo tikimybių matrica $A=(a_{ij})$ (tai matrica, kurios visi elementai neneigiami, o kiekvienos eilutės elementų suma lygi 1). Grandinė vadinama ergodine, jei egzistuoja riba $\lim_{s \rightarrow \infty} A^s = F$ ir matricos F stulpeliuose visi elementai lygūs. Pvz., matrica

```
A <- cbind(c(0.5,1),c(0.5,0))
```

yra ergodinė, nes jau kai s lygus 20

```

> round(mp(A,20),5)
      [,1] [,2]
[1,] 0.66667 0.33333
[2,] 0.66667 0.33333

```

O dabar pati užduotis – nustatykite kurios iš šių matricų yra ergodinės:

$$\text{i) } \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \text{ ii) } \begin{pmatrix} 1 & 0 \\ 1 & 0 \end{pmatrix} \text{ iii) } \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \text{ iv) } \begin{pmatrix} 1/2 & 1/2 \\ 0 & 1 \end{pmatrix} \text{ v) } \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 1/2 & 1/2 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

$$\text{vi)} \begin{pmatrix} 1/2 & 0 & 1/2 & 0 \\ 0 & 0 & 0 & 1 \\ 1/4 & 1/2 & 1/4 & 0 \\ 0 & 1/2 & 1/2 & 0 \end{pmatrix} \text{vii)} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1/2 & 1/2 & 0 & 0 \\ 1/4 & 1/4 & 1/4 & 1/4 \end{pmatrix}$$

Teorija teigia, kad ergodinių grandinių stulpelių elementai $\{f_k\}$ yra lygčių sistemos

$$\begin{cases} f_k = \sum_j f_j a_{jk} \\ \sum_k f_k = 1 \end{cases}$$

sprendiniai. Patikrinkite. *Nuoroda.* Ieškant sistemos sprendinių, pravers funkcija solve.

3.35 UŽDUOTIS. Žemiau pateikta didelės duomenų sistemos dalis abc:

ID	Numeris	Segmentas
S.13	S.13.1	S.13.1.2
S.13	S.13.1	S.13.1.3
S.13	S.13.2	S.13.2.1
S.13	S.13.2	S.13.2.2
S.13	S.13.2	S.13.2.3
S.13	S.13.3	S.13.3.6
S.13	S.13.3	S.13.3.7
S.13	S.13.3	S.13.3.8
S.13	S.13.3	S.13.3.9

Mus domina tik nagrinėtų objektų numeriai, kitaip sakant, šią sistemą transformuokite į tokią:

```
Numeris
S.13.1
S.13.2
S.13.3
```

Nuoroda. unique.

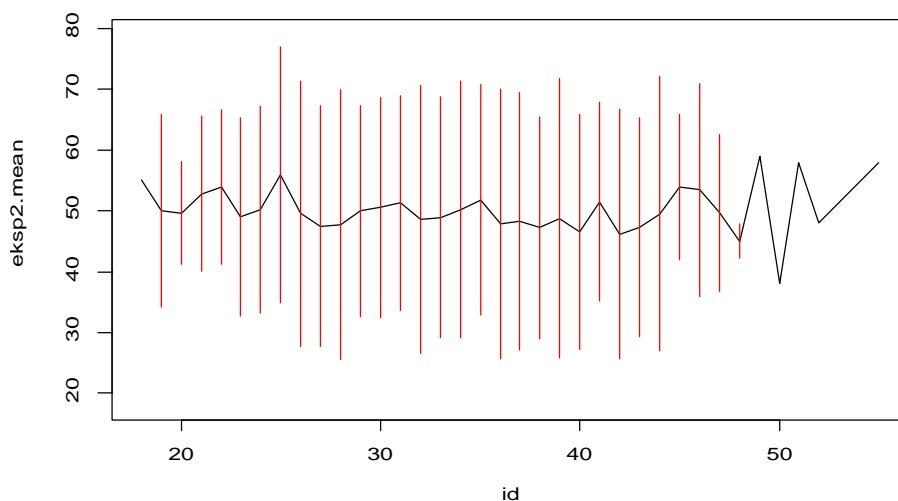
3.36 UŽDUOTIS. i) Visus vektoriaus `rp <- rpois(1000,3)` elementus mažesnius už 3 pakeiskite nulių, o elementus lygius 3 – devynėtu; ii) tą patį atlikite su matrica `rpm <- matrix(rpois(999,3),nrow=333)`; iii) iš šios matricos pašalinkite eilutes, kuriose yra bent vienas 3; iv) iš šios matricos pašalinkite eilutes, sudarytas vien iš 3; v) eilutėse, kurių pirmas elementas lygus 3, kitus du pakeiskite jų kvadratais.

3.37 UŽDUOTIS. Ilgo vektoriaus `x <- rnorm(1000)` duomenis sugrupuokite po 10, kiekvienoje grupėje apskaičiuokite vidurkį, o po to – šio naujo vidurkių vektoriaus standartą. *Nuoroda.* Vektorių `x` galite paversti matrica ir taikyti funkciją `apply` arba įvesti grupės kintamąjį ir taikyti funkciją `tapply` arba remtis funkcija `aggregate`.

3.38 UŽDUOTIS. Dirbtinių duomenų rinkinį `eksp1` generuojame su funkcija

```
eksp1 <- data.frame(ID=rpois(1000,33),Result=
round(rnorm(1000,50,10),0))
```

Suskaičiuokite kelis kartus pasikartoja kiekvienas tiriamasis (ID) šiame sąrašė, kiekvienam tiriamajam apskaičiuokite jo vidurkį ir standartą, išbrėžkite vidurkių grafiką ir kiekvieną vidurkį apsupkite statmenu segmentu $[\text{mean}[i]-2*\text{sd}[i], \text{mean}[i]+2*\text{sd}[i]]$ (galimas atsakymas yra pateiktas 3.12 pav.; jūsų paveikslas ko gero skiriasi nuo šio – kodėl?). *Nuoroda.* Jums gali pagelbėti funkcija `tapply`.



3.12 pav. Vidurkiai ± 2 standartai.
Kai kur raudona linija neišbrėžta – kodėl?

3.39 UŽDUOTIS. Matricioje

```
A <- matrix(rnorm(100,mean=-1),nrow=50)
```

kai kuriose eilutėse abu nariai yra neigiami. Štai funkcija, kuri paliks tik tokias eilutes:

```
A[apply(A, 1, function(x) all(x<=0)),]
```

Parašykite funkciją, kuri pašalintų eilutes, kuriose i) abu nariai neigiami, ii) bent vienas narys neigiamas, iii) paliktų eilutes, kuriose minimalus atstumas nuo nulio yra didesnis už 1, iv) paliktų stulpelius, kurių vidurkis didesnis už -1 .

3.40 UŽDUOTIS. Kartais kai kuriuose įrašuose (eilutėse) trūksta duomenų (R pakete trūkstami duomenys žymimi NA (=Not Available) simboliu). Štai dirbtinis pavyzdys:

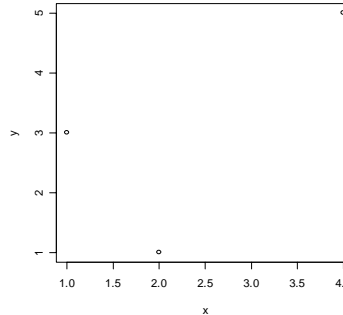
```
dat <- data.frame(x=c(1,NA,2,2,NA,4),y=c(3,2,NA,1,3,5))
dat
  x y
1 1 3
2 NA 2
3 2 NA
```

```
4 2 1
5 NA 3
6 4 5
```

[vairios funkcijos skirtingai reaguoja į NA. Pvz., funkcija `plot` automatiškai išmeta tokius įrašus:

```
plot(x,y)
```

tačiau tiesinės regresijos funkcija `lm` gali elgtis bent dviem būdais: arba tokius įrašus išmesti ir analizę atlikti su likusiais



```
> xy.lm1 <- lm(y~x,na.action=na.omit)
> summary(xy.lm1)
```

```
Call:
lm(formula = y ~ x, na.action = na.omit)
```

```
Residuals:
    1      4      6
1.1429 -1.7143  0.5714
```

```
Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  1.0000      2.6186   0.382   0.768
x              0.8571      0.9897   0.866   0.546
```

```
Residual standard error: 2.138 on 1 degrees of freedom
Multiple R-Squared:  0.4286,    Adjusted R-squared:  -0.1429
F-statistic:  0.75 on 1 and 1 DF,  p-value:  0.5456
```

arba paprašyti, kad tokiu atveju tiesinės regresijos modelis iš vis nebūtų kuriamas:

```
> xy.lm2 <- lm(y~x,na.action=na.fail)
Error in na.fail.default(structure(list(y = c(3, 2, NA, 1, 3, 5), x =
c(1, : missing values in object
```

Duomenų sistemą `dat` galime “pagerinti” keliais būdais: išmesti eilutes, kuriose NA yra tik pirmame (tik antrame stulpelyje), bent viename stulpelyje arba abiejuose stulpeliuose.

```
> is.na(dat$x) # Loginis vektorius
[1] FALSE TRUE FALSE FALSE TRUE FALSE
> dat$x[is.na(dat$x)] # Palieka tik trūkstamus x elementus
[1] NA NA
```

Pažymėsime, kad “standartinė” poaibio sudarymo procedūra `dat$x[dat$x==NA]` neveikia su NA. NA reiškia, kad jūs nežinote, todėl NA nelygus niekam, įskaitant ir NA (aš nežinau nei Bušo nei Putino gimtadienių (taigi rašau NA), tačiau tai nereiškia, kad šios dvi datos sutampa).

```
> dat$x[!is.na(dat$x)] # Palieka tik turimus x elementus
[1] 1 2 2 4
> dat[!is.na(dat$x),] # Palieka tik dat eilutes su turimais x
# elementais (simbolis ! žymi loginį neiginį)
```

```

      x y
1 1 3
3 2 NA
4 2 1
6 4 5

```

Kiek sudėtingiau pašalinti eilutes kuriose yra bent vienas NA:

```

> dat[apply(dat, 1, function(x) !any(is.na(x))),]
      x y
1 1 3
4 2 1
6 4 5

```

O dabar pati užduotis: duomenų rinkinyje `airquality` (kas tai per duomenys?)

```

> library(MASS)
> data(airquality)
> airquality
      Ozone Solar.R Wind Temp Month Day
1      41     190  7.4   67     5   1
2      36     118  8.0   72     5   2
3      12     149 12.6   74     5   3
4      18     313 11.5   62     5   4
5      NA      NA 14.3   56     5   5
6      28      NA 14.9   66     5   6
7      23     299  8.6   65     5   7

```

yra nemažai įrašų su trūkstamais duomenimis. Pertvarkykite šį rinkinį taip, kad jame liktų tik įrašai, i) kurių `Solar.R` nėra NA, ii) kurie neturi NA. *Nuoroda*. `?all`

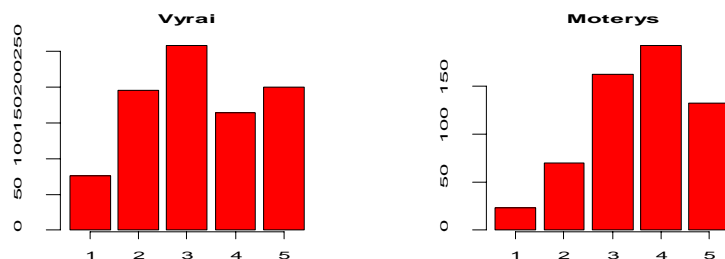
3.41 UŽDUOTIS. Inventorizuojant sandėlį, buvo aprašytas kiekvienos dėžės turinys:

```

g <- sample(letters,100,replace=TRUE) # g=gaminio i-ojoje dėžėje var
                                     # das
k <- rpois(100,20) # k=gaminių kiekis i-ojoje dėžėje
gk <- data.frame(g,k)

```

Apskaičiuokite: a) kelios dėžės su gaminiais a, b ir t.t. yra sandėlyje, ir b) keli a, b ir t.t. gaminiai yra sandėlyje.



4.2 pav. Vyrų (kairėje) ir moterų (dešinėje) išsilavinimo educ stulpelinės diagramos

Atrodo, kad moterų išsilavinimas aukštesnis, tačiau “tikslų” (skaitinį) atsakymą pateiksime tik sprendžiamosios statistikos skyriuje. Antra vertus, dalinį atsakymą galime gauti, palyginę abiejų ėminių medianas:

```
> median(educ[male==1])
[1] 3
> median(educ[male==0])
[1] 4
```

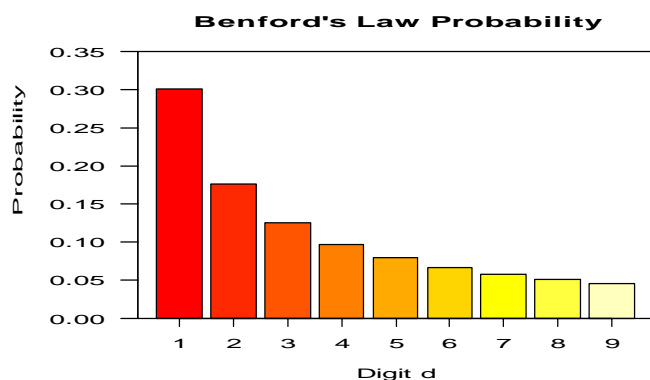
4.1 UŽDUOTIS. Labiausiai tikėtina diskretaus a.d. arba faktoriaus reikšmė vadinama jo moda. Žemiau yra pateiktos kelios R eilutės, kurios pateikia modos vardą.

```
> set.seed(3)
> rp <- rpois(20,2)
> barplot(table(rp))
> names(ta <- table(rp))[ta==max(ta)] # Dvi operacijos vienu metu:
[1] "3" # ta <- table(rp) ir names(ta)
```

Raskite vyrų ir moterų išsilavinimo modas.

4.2 UŽDUOTIS. Vadinamųjų Benfordo tikimybių grafiką galima išbrėžti taip:

```
x <- 1:9
p <- log10(1 + 1/x)
barplot(p, xlab = "Digit d", ylab = "Probability", ylim = c(0, 0.35),
axes = FALSE, names.arg=c(1:9),main = "Benford's Law Probability")
axis(2, seq(0, 0.35, by = 0.05), las = 1)
box()
```



4.3 pav. Benfordo tikimybių grafikas

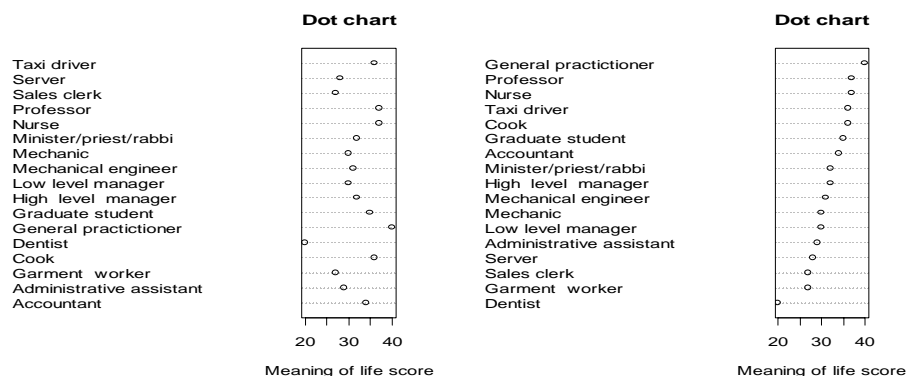
- a) Ar skaičiai p nusako tikimybes?
 b) Pasiskaitykite apie Benfordo tikimybes internete, pvz., <http://mathworld.wolfram.com/BenfordsLaw.html>. Papasakokite, ką sužinojote.

Vietoje stulpelinių diagramų (barplot) kartais tikslinga brėžti taškines diagramas (su dotchart). Štai psichologų surinkti duomenys apie tai, kaip įvairių profesijų žmonės vertina savo gyvenimą:

```
prof <- c('Accountant', 'Administrative assistant', 'Garment worker',
'Cook', 'Dentist', 'General practitioner', 'Graduate student',
'High level manager', 'Low level manager', 'Mechanical engineer',
'Mechanic', 'Minister/priest/rabbi', 'Nurse', 'Professor',
'Sales clerk', 'Server', 'Taxi driver')
mol <- c(34, 29, 27, 36, 20, 40, 35, 32, 30, 31, 30, 32, 37, 37, 27,
28, 36) # mol = Meaning Of Life = Gyvenimo prasmė
```

Funkcijos dotchart ir barplot kintamuosius brėžia ta tvarka, kuria jie pateikiami, todėl pirmasis grafikas (žr. 4.4 pav., kairėje) nėra labai informatyvus – kintamojo mol reikšmes tikslingiau išdėstyti didėjimo tvarka (žr. 4.4 pav., dešinėje).

```
par(mfrow=c(1,2))
# Brėžiame kairinį grafiką:
dotchart(mol, labels=prof, main='Dot chart', xlab='Meaning of life score')
# Brėžiame dešinį grafiką:
names(mol) <- prof
dotchart(sort(mol), main="Dot chart", xlab="Meaning of life score")
```



4.4 pav. Gyvenimo kokybės įverčiai

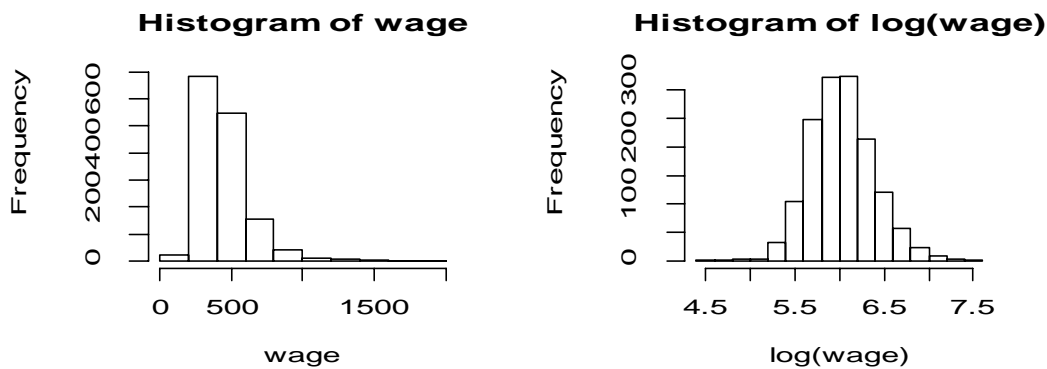
Taigi mažiausiai gyvenimu patenkinti yra (daug uždirbantys!) dantistai, o (beveik) labiausiai – universitetų dėstytojai.

4.2. Skaitiniai kintamieji

4.2.1. Histogramos

Skaitinių kintamųjų atveju vietoje funkcijos barplot naudosime funkciją hist. Išstirkime kintamojo wage elgesį.

```
hist(wage)
hist(log(wage))
```



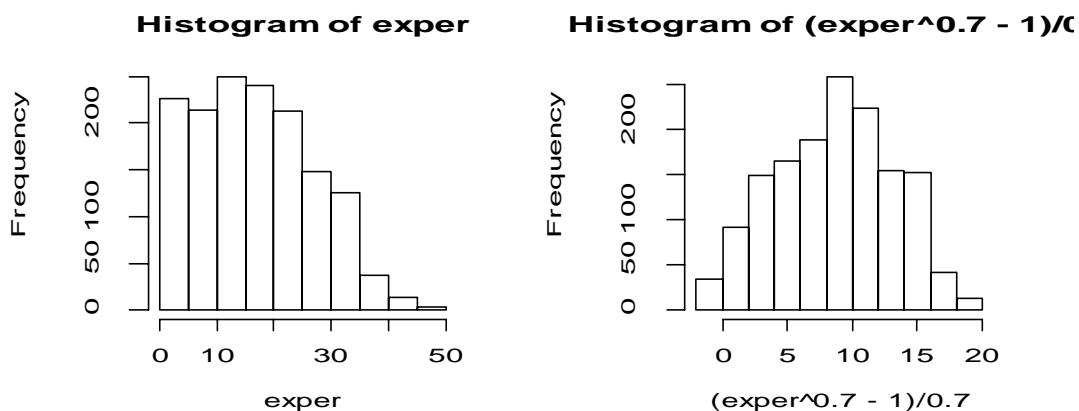
4.5 pav. Kintamojo wage (kairėje) ir jo logaritmo (dešinėje) histogramos

Dauguma klasikinių statistikos kriterijų reikalauja, kad tiriamasis skaitinis kintamasis būtų normalus (arba “beveik normalus”). Matome, kad wage logaritmo tankis yra labiau simetriškas, taigi kintamasis $\log(\text{wage})$ yra “labiau normalus” (ar jis dabar iš tikro normalus – kitas klausimas (čia gali pagelbėti funkcija $\varphi\varphi\text{norm}$, žr. žemiau)). Apskritai, jei kintamasis y yra teigiamas, o skirstinys nėra simetriškas, tai dažnai Box – Cox’o transformacija “pagerina normalumą”. Šis metodas siūlo vietoje y nagrinėti naują kintamąjį $t_\lambda(y)$:

$$t_\lambda(y) = \begin{cases} \frac{y^\lambda - 1}{\lambda}, & \text{jei } \lambda \neq 0, \\ \log y, & \text{jei } \lambda = 0. \end{cases}$$

Parametrą λ siūloma rinktis iš intervalo (-2,2) (jo parinkimą regresiniuose uždaviniuose galima automatizuoti, žr. [Fa, 89 p.]).

Štai dar vienas transformacijos pavyzdys (ši kartą $\lambda = 0,7$ - kitais žodžiais, vietoje exper reikia nagrinėti $1:0,7=1,4$ eilės šaknį iš jo):



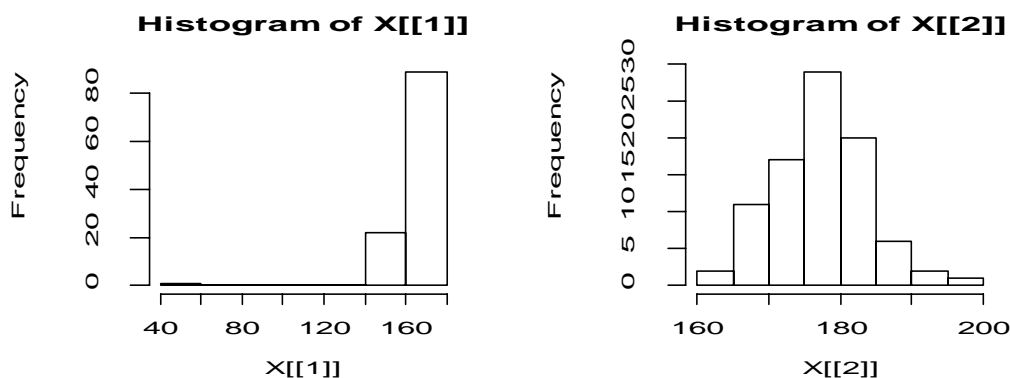
4.6 pav. Kintamojo exper (kairėje) ir jo Box-Cox’o transformacijos (dešinėje) histogramos

Prisiminkime domenų rinkinį Davis :

```
> library(car)
> data(Davis)
> Davis[1:4,]
  sex weight height repwt repht
1  M    77   182    77   180
2  F    58   161    51   159
3  F    53   161    54   158
4  M    68   177    70   175
> attach(Davis) # Į Davis komponentes dabar galima kreiptis vardais
```

Parašysime funkciją, kuri išbrėš dvi – moterų ir vyrų ūgio - histogramas

```
dvi.hist <- function(){
opar<-par(mfrow=c(1,2))
on.exit(par(opar))
tapply(height,sex,hist) # Brėšime dvi histogramas - moterų ir vyrų
invisible()}           # Kas pasikeistų, jei 4-osios funkcijos
dvi.hist()             # eilutės nebūtų?
```



4.7 pav. Moterų (kairėje) ir vyrų (dešinėje) ūgio histogramos

Ūgis paprastai yra “pavyzdingai” normalus, todėl moterų histograma kelia nusistebėjimą – ten arba yra įrašymo klaida arba tyrimuose dalyvavo patologiškai mažo ūgio moteris.

```
> sort(height)[1:8]
[1] 57 148 150 152 153 154 155 156 # Yra viena išskirtis

> which.min(height)
[1] 12 # Kurioje eilutėje?
# 12-ojoje Davis eilutėje
> Davis[(12-4):(12+4),] # Pasidairykime šios eilutės
  sex weight height repwt repht # aplinkoje
8  M    69   186    73   180
9  M    71   178    71   175
10 M    65   171    64   170
11 M    70   175    75   174
12 F   166    57    56   163 # Priežastis aiški: ūgis ir
13 F    51   161    52   158 # svoris sukeisti vietomis
14 F    64   168    64   165
15 F    52   163    57   160
16 F    65   166    66   165

> davis <- Davis # Sukuriame Davis kopiją
> davis[12,2] <- 57 # Joje ištaisome klaidas
> davis[12,3] <- 166
```

```

> davis[(12-4):(12+4),]
  sex weight height repwt repht
8   M     69   186    73   180
9   M     71   178    71   175
10  M     65   171    64   170
11  M     70   175    75   174
12  F     57   166    56   163
13  F     51   161    52   158
14  F     64   168    64   165
15  F     52   163    57   160
16  F     65   166    66   165

> attach(davis)
> sex=="F"
 [1] FALSE TRUE TRUE FALSE TRUE
 [6] FALSE FALSE FALSE FALSE FALSE
[11] FALSE TRUE TRUE TRUE TRUE

.....
hF <- height[sex=="F"]
hist(hF,probability=T)
lines(density(hF))
xx <- seq(min(hF),max(hF),length=100)
lines(xx,dnorm(xx,mean(hF),sd(hF)),lty=2)

```

Dabar gerai

Prijunkime davis

Tai loginis vektorius: jei

įrašytas vyras - bus FALSE,

jei moteris - TRUE

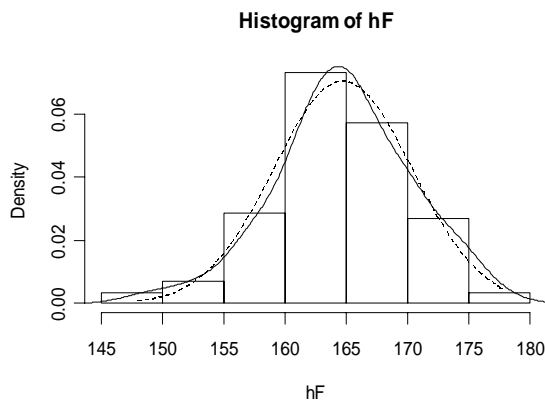
Moterų ūgio vektorius

Moterų ūgio histograma

"Suglodinta" histograma²

Brėžiame atitinkama

normaląjį tankis



4.8 pav. $\text{density}(hF)$ (ištinė linija) mažai skiriasi nuo atitinkamo normaliojo tankio (brūkšniuota linija)

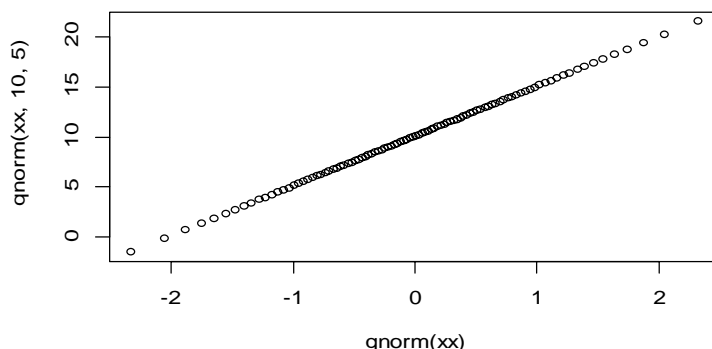
Histogramos ir empirinio tankio grafikai leidžia susidaryti teisingą skirstinio formos vaizdą. Žemiau aptariami kvantilių grafikai taip pat tam tinka, tačiau jie dar vaizdžiai pateikia ir išskirtis.

4.2.2. Kvantilių grafikai

Lyginant tiriamą skirstinį su normaliuoju, dažnai tikslinga remtis vadinamuoju kvantilių grafiku. Teoriniame šio grafiko variante x ašyje atidedami standartinio normaliojo skirstinio kvantiliai, o y ašyje – tiriamojo skirstinio kvantiliai. Jei tiriamasis skirstinys yra beveik normalus, tai gauti taškai bus praktiškai ant tiesės:

² Apie empirinio tankio funkciją density galima daugiau pasiskaityti su $?density$ arba [V&R, 5.6 skyrelyje].

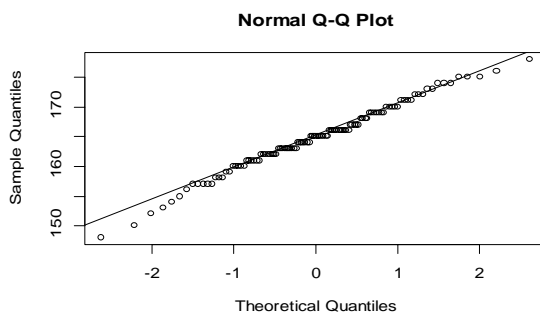
```
xx <- seq(0,1,length=100)
plot(qnorm(xx),qnorm(xx,10,5)) # Standartinį normalųjį a.d. N(0,1)
# lyginame su N(10,5)
```



4.9 pav. Normaliojo skirstinio teorinių kvantilių grafikas

Šio grafiko empirinį variantą brėžiame taip: jei tiriamų duomenų yra n , tai x ašyje atidedame standartinio normaliojo skirstinio $1/n, 2/n, \dots, n/n$ kvantilius, o y ašyje – tiriamojo dydžio variacinės eilutės pirmąją, antrąją, ..., n -ją reikšmes (t.y., jo empirinius kvantilius). Jei tiriamasis dydis beveik normalus, tai gauti taškai bus beveik ant tiesės. hF duomenims kvantilių grafiką galime išbrėžti taip:

```
> qqnorm(hF)
> qqline(hF) # Brėžiame tiesę per 1-jį ir 3-jį kvartilius
```



4.10 pav. Matome, kad taškai yra beveik ant tiesės, todėl hF skirstinys beveik normalus.

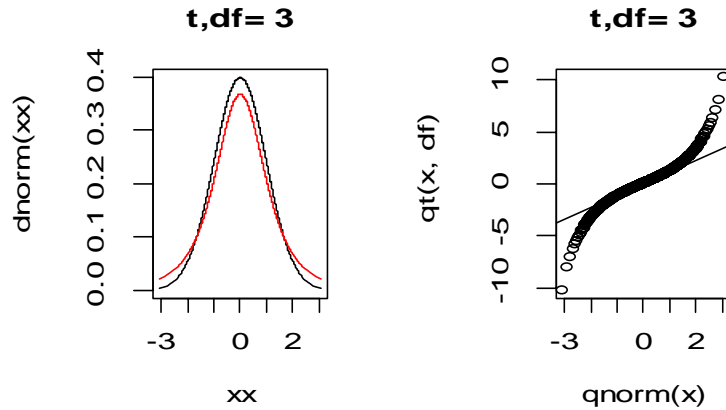
Kaip atrodo nenormalių dydžių kvantilių grafikai? Štai funkcija `forma.t`, kuri lygina normalųjį skirstinį su Stjudento skirstiniu:

```
forma.t <- function(df){
# Funkcija forma.t
# df=degrees of freedom=laisvės laipsnių skaičius=1.1.
opar <- par(mfrow=c(1,2))
on.exit(par(opar))
x <- seq(0,1,length=1000)
xx <- qnorm(x)
plot(xx,dnorm(xx),type="l",main=paste("t,df=",df))
lines(xx,dt(xx,df),col=2)
# dt(...,df) skaičiuoja Stjudento skirstinio su df 1.1. tankį
plot(qnorm(x),qt(x,df),main=paste("t,df=",df))
```

```

Y <- qt(c(0.25, 0.75),df) # Brėšime tiesę per 1-jį ir 3-jį kvartilius
X <- qnorm(c(0.25, 0.75))
slope <- diff(Y)/diff(X) # slope=krypties koeficients
int <- Y[1] - slope * X[1] # int=intercept=laisvasis narys
abline(int, slope) # Brėžiame tiesę
}

```



4.11 pav. Stjudento a.d. su 3 l.l.: tankių grafikas (kairėje) ir kvantilių grafikas (dešinėje)

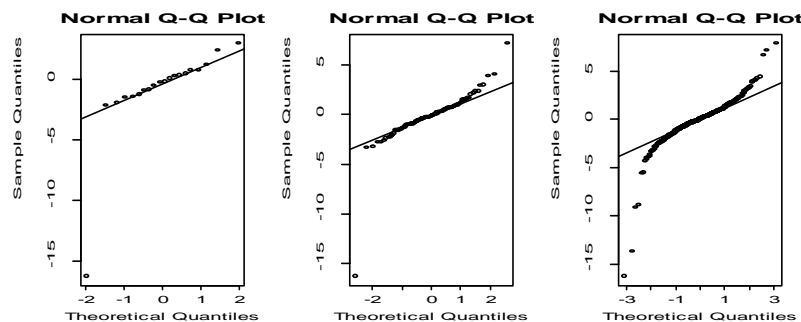
Stjudento tankio (raudona spalva) “uodegos” yra “sunkesnės”³ už normaliąsias (t.y., pvz., $P(T_3 > 2) > P(N(0,1) > 2)$; panašiai yra ir kairėje). Kvantilių grafike tai atitinka tą faktą, kad taškai dešinėje yra aukščiau kvantilių tiesės, o kairėje – žemiau.

Pateiksime empirinį šio reiškinių analogą:

```

par(mfrow=c(1,3))
t3 <- rt(500,3) # Generuojame 500 a.skaičių (Stjudento su 3 l.l.)
qqnorm(t3[1:20]) # Pirmieji 20
qqline(t3[1:20])
qqnorm(t3[1:100]) # Pirmieji 100
qqline(t3[1:100])
qqnorm(t3[1:500]) # Visi 500
qqline(t3[1:500])
rm(t3) # Baigę darba, pašalinkime sukurtą objektą t3

```



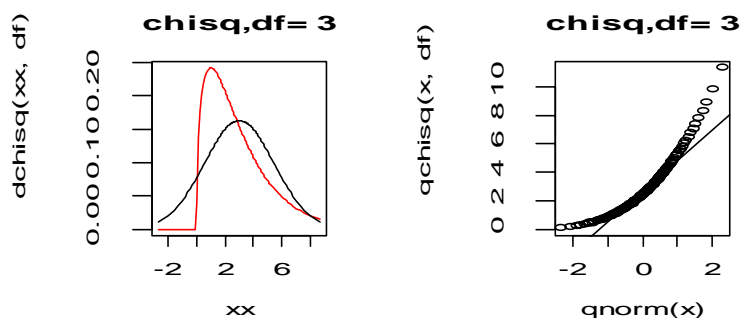
4.12 pav. Stjudento skirstinio kvantilių grafikai: ėminio dydis lygus 20 (kairėje), 100 (viduryje) ir 500 (dešinėje)

³ Sakome, kad a.d. X turi “sunkias” “uodegas”, jei jo tankis begalybėje ėsta lėčiau nei normalusis.

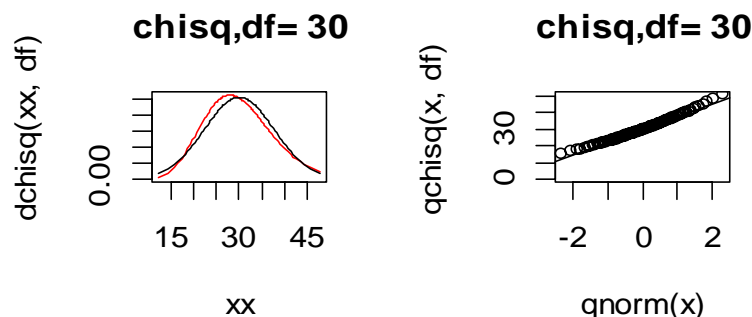
Stebime tipiską reiškinių: ėminio dydžiui augant, darosi vis aiškiau, kad stebimasis dydis nėra normalusis (uodegos sunkesnės nei priklauso).

Funkcijos forma . t modifikacija chi kvadrato skirstiniui galėtų atrodyti taip:

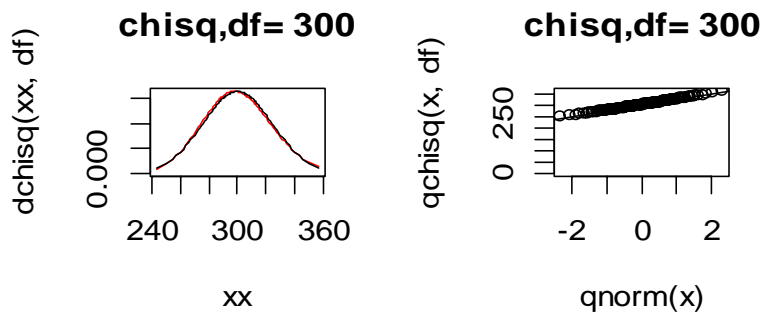
```
forma.chi <- function(df){
# Funkcija forma.chi
# df=degrees of freedom=laisves laipsniu skaicius=l.l.
opar <- par(mfrow=c(1,2))
on.exit(par(opar))
x <- seq(0,1,length=100)
xx <- qnorm(x,df,sqrt(2*df)) # Normaliojo a.d. parametrai sutampa
# su chi kvadrato vidurkiu ir standartu
plot(xx,dchisq(xx,df),type="l",main=paste("chisq,df=",df),col=2)
# dchisq(...,df) skaičiuoja chi kvadrato su df l.laipsniais tankį
lines(xx,dnorm(xx,df,sqrt(2*df)))
plot(qnorm(x),qchisq(x,df),main=paste("chisq,df=",df))
Y <- qchisq(c(0.25, 0.75),df) # Brėšime tiesę per 1-jį ir 3-jį
# kvartilius
X <- qnorm(c(0.25, 0.75))
slope <- diff(Y)/diff(X) # slope=krypties koeficientas
int <- Y[1] - slope * X[1] # int=intercept=laisvasis narys
abline(int, slope) # Brėžiame tiesę y=slope*x+int
}
```



4.13 pav. $df=3$: dešinė uodega sunkesnė, o kairė - lengvesnė



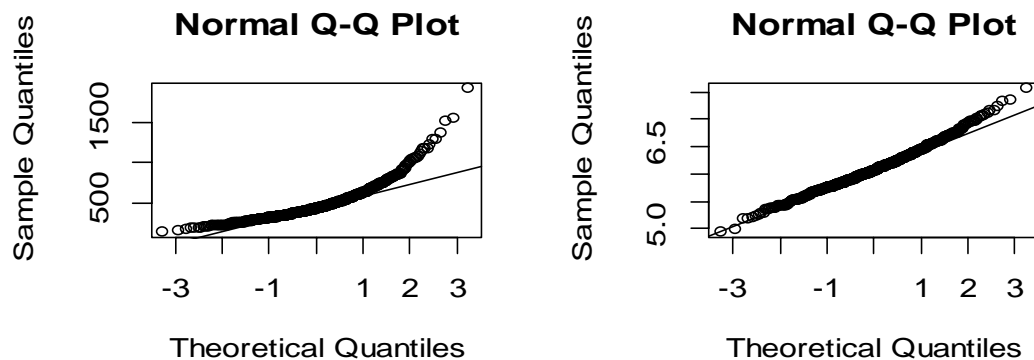
4.14 pav. Kai $df=30$, skirtumas tarp normaliojo skirstinio ir chi kvadrato nėra didelis



4.15 pav. “Įrodėme” centrinę ribinę teoremą: kai df didelis, chi kvadrato a.d. (kuris pats yra a.d. suma) praktiškai sutampa su normaliuoju a.d.

Panaudosime sukaupias žinias ir patikrinsime vyrų atlyginimo normalumą.

```
attach(bwages) # Galėsime naudotis stulpelių vardais
par(mfrow=c(1,2))
w1 <- wage[male==1]
qqnorm(w1)
qqline(w1)
qqnorm(log(w1))
qqline(log(w1))
detach(bwages) # Kai duomenų nebereikia - atjunkime
```



4.16 pav. Kintamųjų w_1 ir $\log(w_1)$ kvantilių grafikai

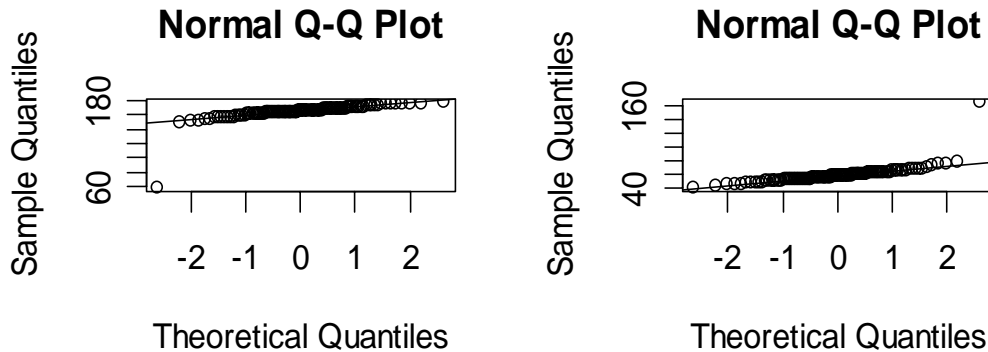
Vyrų atlyginimo dešinioji uodega yra sunkesnė už normaliąją (t.y., nemažai vyrų gauna labai didelį atlyginimą). Antra vertus, $\log(w_1)$ yra praktiškai normalus a.d. (nors dešinė uodega vis dar per sunki; antra vertus, nereguliarus ekstremalių reikšmių elgesys yra tipiškas reiškinys).

Kvantilių grafikai ir žemiau aptariamoms stačiakampės diagramos labai efektyviai išryškina išskirtis. Iš tikrųjų:

```
library(car)
data(Davis)
attach(Davis)
par(mfrow=c(1,2))
qqnorm(height[sex=="F"])
qqline(height[sex=="F"])
qqnorm(weight[sex=="F"])
qqline(weight[sex=="F"])
```

```
qqline(weight[sex=="F"])
detach(Davis)
detach("package:car")
par(mfrow=c(1,1))
```

Pastarosios trys
eilutės pateikia tvarkingos
darbo pabaigos pavyzdį



4.17 pav. Moterų ūgio (kairėje) ir svorio (dešinėje) kvantilių grafikai; abiejuose grafikuose aiškiai matome po vieną išskirtį

4.2.3. Stačiakampės diagramos

Stačiakampės diagramos yra naudingos tiriant skirstinio simetriškumą ir ieškant išskirčių. Jų brėžimas remiasi kintamojo kvartiliais (žr. 0 sk.). Panagrinėkime `wage` ir `log(wage)` atvejus. Štai trys praktiškai ekvivalenčios funkcijos, skaičiuojančios kvartilius (t.y., kvantilius, atitinkančius 1/4, 2/4, 3/4; atkreipkite dėmesį į skirtumus):

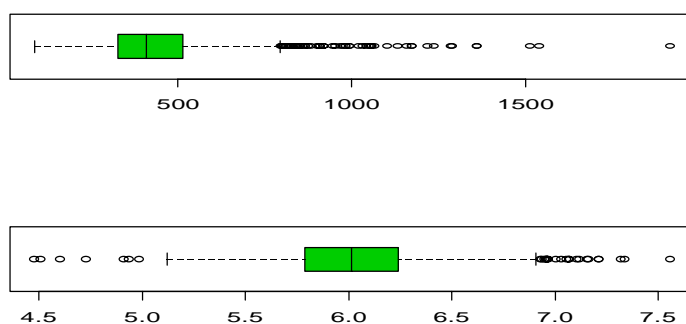
```
> attach(bwages)
> quantile(wage)
      0%      25%      50%      75%     100%
 88.38383 327.27270 408.50815 514.55345 1919.19200
> fivenum(wage)
[1] 88.38383 327.27270 408.50815 514.75280 1919.19200
> summary(wage)
      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
 88.38   327.30   408.50   445.80   514.60  1919.00
```

daugiausia informacijos

Išbrėšime stačiakampes diagramas:

```
> par(mfrow=c(2,1))
> boxplot(wage,horizontal=T,col=3)
> boxplot(log(wage),horizontal=T,col=3)
```

Pažymėsime, kad `wage` diagrama (žr. žemiau) yra labai nesimetriška (yra daug reikšmių dešiniau 3-iojo kvartilio arba, kitais žodžiais, yra daug žmonių, turinčių dideles pajamas). Antra vertus, nemažai didelių nuokrypių nuo medianos matome ir `log(wage)` atveju, tačiau nereiktų užmiršti, kad netgi normaliuoju atveju maždaug vienas stebėjimas iš 100 (arba 14 iš 1472) gali būti už žandenu.

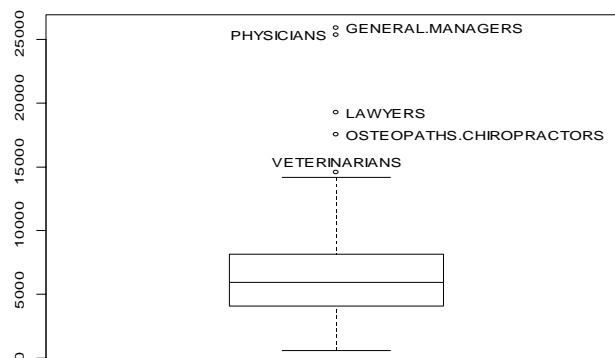


4.18 pav. Kintamųjų $wage$ (viršuje) ir $\log(wage)$ (apačioje) stačiakampės diagramos; $wage$ diagrama yra labai nesimetriška ir turi daug didelių reikšmių

R turi funkciją `identify`, kuri leidžia identifikuoti taškus sklaidos diagramose (žr. žemiau, 6-7 psl.). Ši funkcija leidžia identifikuoti išskirtis ir stačiakampėse diagramose, tačiau šiuo atveju `identify` argumentus reikia nurodyti specialiu būdu:

```
library(car)
data(Prestige)
?Prestige
attach(Prestige)
boxplot(income)
identify(rep(1,length(income)),income,labels=rownames(Prestige))
[1] 2 17 24 25 26
```

Matome, kad penkių kategorijų žmonių uždarbis yra “nenormaliai” didelis. Norėdami nustatyti jų specialybes, naudosisime funkciją `identify`. Kai ši komanda bus įvykdyta, pereikime į grafikos langą, pasirodžiusį kryželį nuvartykite kairiau, dešiniau ar virš norimos išskirties ir spragtelėkite kairiuoju klavišu. Norėdami darbą baigti, spragtelėkite dešiniuoju klavišu.

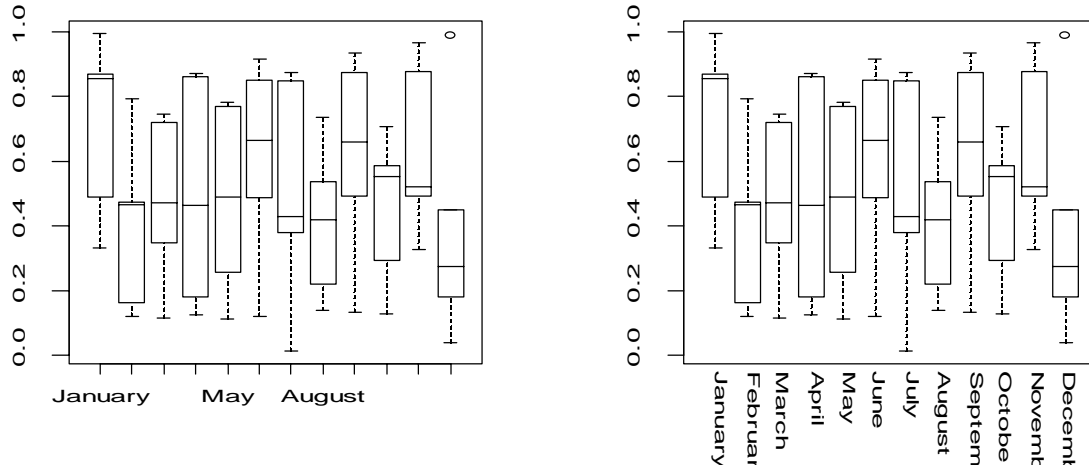


4.19 pav. Uždarbio stačiakampė diagrama ir identifikuotos išskirtys

4.3 UŽDUOTIS. Kartais x ašyje yra per mažai vietos vardamas įrašyti. Tokiu atveju reiktų pasinaudoti funkcijos `par` teikiamomis galimybėmis (pvz., vardus rašyti vertikaliai).

```
par(mfrow=c(1,2))
d <- as.data.frame(matrix(runif(60),5))
names(d) <- month.name
boxplot(d) # Vardams mažai vietos
boxplot(d,xaxt='n')
# Kai xaxt='n', x ašis nebus išbrėžta
ypos <- par()$usr[3]-(par()$usr[4]-par()$usr[3])/50
```

```
# > ypos
#[1] -0.0446786
text(1:ncol(d),ypos, names(d), srt=270, adj=0, xpd=T)
# srt pasuka tekstą 270°, adj=0 tekstą išlygina pagal kairę pusę,
# xpd leidžia spausdinti už grafiko ribų,
```



4.20 pav. Stačiakampės diagramos su vardais

O dabar pati užduotis: pakeiskite antrąjį grafiką taip, kad y ašies skaičiai ir mėnesių vardai po x ašimi “žiūrėtų” į tą pačią pusę.

4.4 **UŽDUOTIS.** Duomenų rinkinyje `emissions` kintamasis `CO2` turi išskirčių (kaip jas galima pamatyti?) Nustatykite jas su `identify`, pašalinkite šiuos įrašus, o pasikui iš naujo išbrėžkite reikalingus grafikus.

4.5 **UŽDUOTIS.** Duomenų pakete `Simple` yra duomenų rinkiniai `south`, `crime` ir `aid` (esant reikalui, juos importuokite su `source`, žr. 4.16 užduotį). Kuris iš šių ėminių yra simetriškas? turi sunkias uodegas? išskirtis?

4.6 **UŽDUOTIS.** Susiraskite kelis ekonominę prasmę turinčius grafikus laikraštyje ar internete ir perbraižykite juos su R.

4.7 **UŽDUOTIS.** Norėdami generuoti diskrečiuosius tolygiai pasiskirsčiusius atsitiktinius skaičius 0, 1, ..., 9, galime elgtis taip:

```
x<-floor(runif(1000)*10)
par(mfrow=c(1,3))
hist(x)
barplot(x)
barplot(table(x))
```

Pakartokite šią procedūrą kelis kartus. Kuri iš šių procedūrų “įrodo” gautų skaičių skirstinio tolygumą? Kodėl `hist(x)` brėžia “neteisingą” histogramą? Atspausdinkite kiekvieno skaitmens pasirodymo santykinę dažnį.

4.2.4. Skaitinės charakteristikos

Iki šiol nagrinėjome grafines ėminio charakteristikas. Labai lakoniškos yra skaitinės charakteristikos. Jau žinome, kad skaitinio ėminio “centrinę” reikšmę nusakome mediana arba vidurkiu \bar{x} . Deja, net viena didelė išskirtis gali smarkiai iškreipti vidurkį, kitais žodžiais, tokio ėminio vidurkis smarkiai nukryps nuo populiacijos vidurkio. Įvėrčio atsparumą išskirtims galima padidinti “nupjaunant” didžiausias ir mažiausias reikšmes:

```
> mean(wage)
[1] 445.7807
> mean(wage,trim=0.1) # Atmetėme 10% didžiausių ir 10% mažiausių
[1] 423.1198 # reikšmių
> mean(wage,trim=0.2) # Atmetėme 20%
[1] 414.6184
> mean(wage,trim=0.5)
[1] 408.5082
> median(wage) # Atmetėme 50% - tas pat kas mediana
[1] 408.5082
```

Simetrinio skirstinio atveju vidurkis ir mediana skiriasi nedaug:

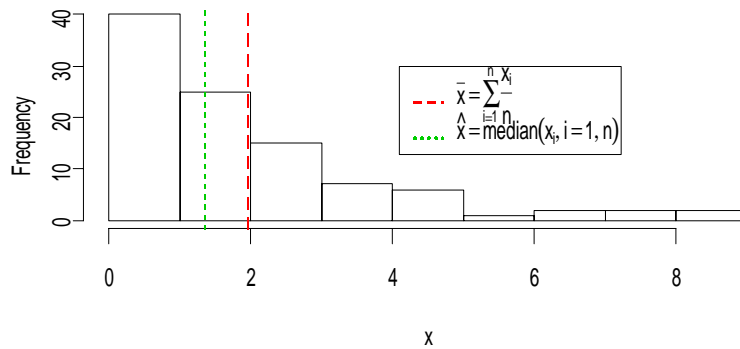
```
> mean(log(wage),trim=0.1)
[1] 6.021636
> median(log(wage))
[1] 6.012511
```

Štai dar vienas, truputį pakeistas pavyzdys iš ?legend: asimetrinio skirstinio atveju vidurkis ir mediana nesutampa. Paskutinėje eilutėje esanti funkcija locator leidžia pasirinkti legendos rėmelio viršutinį kairinį tašką – spragtelėkite ten su kairiuoju pelės klavišu.

```
x <- rexp(100,rate = .5) # Skewed=asimetrinis
hist(x,main = "Mean and Median of a Skewed Distribution")
abline(v=mean(x),col=2,lty=2,lwd=2) # "v" reiškia "vertical"
abline(v=median(x),col=3,lty=3,lwd=2)
ex12 <- expression(bar(x)==sum(over(x[i],n),i==1,n),
                    hat(x)==median(x[i],i==1,n)) # Formulų grafikuose
# rašymo pavyzdys4
#legend(4.1, 30, ex12, col = 2:3, lty=2:3, lwd=2) # Reikia pasirinkti
legend(locator(1),ex12, col = 2:3, lty=2:3, lwd=2) # vieną iš šių
# dviejų eilučių
```

⁴ Dar vienas pavyzdys: plot(1:10, xlab=expression(m²), ylab=expression(mu*g))

Mean and Median of a Skewed Distribution



4.21 pav. Asimetrinio skirstinio atveju vidurkis ir mediana nesutampa

Aišku, kad vienu, centrą charakterizuojančiu skaičiumi, reiškinį aprašyti neįmanoma. Nesunku įsivaizduoti ėminį, kurio visos reikšmės koncentruojasi ties vidurkiu ir kitą ėminį, kurio kai kurios reikšmės yra smarkiai nutolusios nuo vidurkio. Šių dviejų ėminių vidurkiai gali sutapti, tačiau jie skirsis savo reikšmių išsibarstymo didumu. Pastarąjį galima charakterizuoti keliais būdais, dažniausiai tam naudojama dispersija.

```
> var(hF) # "Teisingo" moterų ūgio dispersija
[1] 32.02574
```

Ūgis matuojamas cm , o jo dispersija - cm^2 , todėl tikslingiau reikšmių išsibarstymą matuoti šaknimi iš dispersijos, kitaip sakant, ėminio standartu:

```
> sqrt(var(hF))
[1] 5.659129
> sd(hF) # "Teisingo" moterų ūgio standartas
[1] 5.659129
```

Prisiminkime, kad duomenų rinkinyje `Davis` moterų ūgis buvo pateiktas klaidingai:

```
> library(car)
> data(Davis)
> attach(Davis)
> sd(height[sex=="F"])
[1] 11.64393 # 11.64≠5.66!
```

Matėme, kad mediana yra atsparesnė klaidoms nei vidurkis. Ja pagrįstas reikšmių išsibarstymo matas yra vadinamas MAD (median average deviation) ir apibrėžiamas formule

$$mediana | x_i - mediana(x) | \cdot 1,4826$$

Kitais žodžiais, pirmiausia reikia apskaičiuoti duomenų rinkinio x medianą, iš kiekvienos rinkinio reikšmės atimti medianą, apskaičiuoti naujojo rinkinio modulių medianą ir dar padauginti iš 1,4826 (dauginame tam, kad normaliojo skirstinio atveju MAD sutaptų su standartu).

```
> mad(height[sex=="F"])
[1] 5.9304 # 5,93 žymiai arčiau skaičiaus 5,66 negu 11.64
```

Dar viena išsibarstymo charakteristika yra pagrįsta skirtumu tarp 3-jo ir 1-jo kvartilų:

```
> IQR(hF)
[1] 7.25
> IQR(height[sex=="F"])
[1] 8
```

Matome, kad šie du skaičiai tarpusavy nelabai skiriasi, tačiau, norint kad Gauso skirstinio atveju jie būtų artimi standartui, reikia įvesti korekcinį daugiklį $1/(qnorm(0.75) - qnorm(0.25)) (=0,7413)$:

```
> IQR(hF)*0.7413
[1] 5.374425
> IQR(height[sex=="F"])*0.7413
[1] 5.9304
```

Ėminio normalumui tikrinti galima taip pat naudoti jo trečiąjį ir ketvirtąjį momentus, tiksliau kalbant, ėminio asimetrijos koeficientą

$$ask = \frac{1}{n} \sum_{i=1}^n \left(\frac{x_i - \bar{x}}{sd(s)} \right)^3$$

ir ėminio ekscesą

$$eks = \frac{1}{n} \sum_{i=1}^n \left(\frac{x_i - \bar{x}}{sd(x)} \right)^4.$$

Skirstinių su simetrišku (vidurkio atžvilgiu) tankiu asimetrijos koeficientas lygus 0, o normaliojo dėsnio ekscesas lygus 3. Taigi nuokrypiai nuo šių reikšmių signalizuoja apie galimą ėminio nenormalumą. R neturi funkcijų ask ir eks, todėl parašykime jas patys.

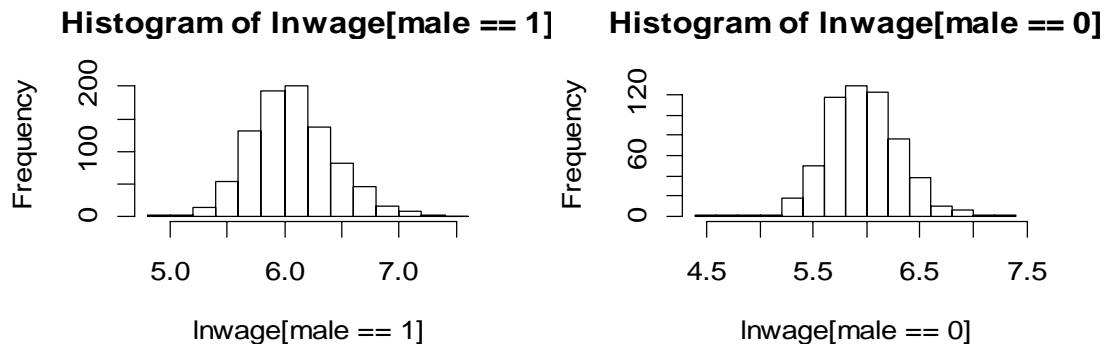
```
ask <- function(x){sum((x-mean(x))^3)/(length(x)*(sd(x))^3)}
eks <- function(x){sum((x-mean(x))^4)/(length(x)*(sd(x))^4)}
> attach(bwages)
> ask(wage)
[1] 1.951409
> eks(wage) # Abu skaičiai signalizuoja apie didelį
[1] 10.30401 # kintamojo wage nenormalumą
> ask(lnwage)
[1] 0.2245817
> eks(lnwage)
[1] 3.95701 # log(wage) gana panašus į normalųjį
```

Galimas daiktas, kad moterų ir vyrų atlyginimų struktūra skiriasi, todėl juos tikslinga panagrinėti atskirai

```
> ask(lnwage[male==1])
[1] 0.3963979 # Vyrų log(wage) asimetriškumas didelis
> eks(lnwage[male==1])
[1] 3.512542 # Panašu į normalumą
> ask(lnwage[male==0])
[1] -0.0854381 # Moterų asimetriškumas mažesnis nei vyrų
> eks(lnwage[male==0])
[1] 4.443928 # Nelabai panašu į normalumą
```



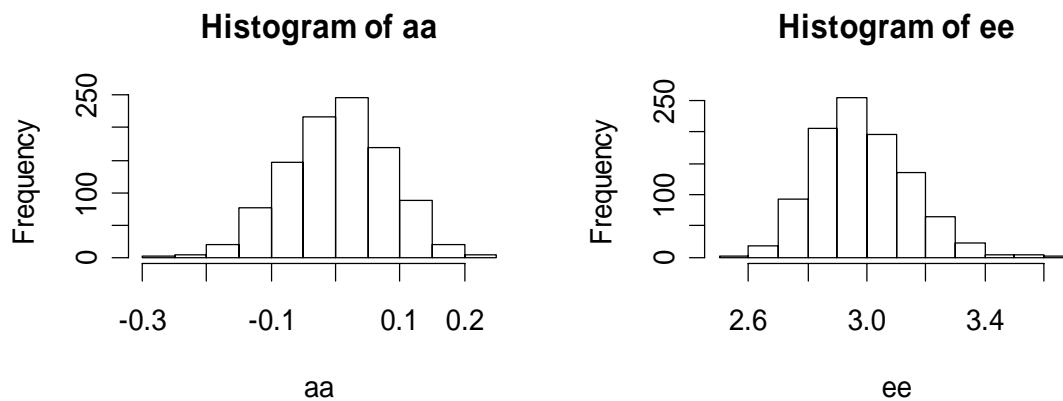
```
> par(mfrow=c(1,2)) # Dar išbrėškime histogramas - jos
> hist(lnwage[male==1]) # gana simetriškos
> hist(lnwage[male==0])
```



4.22 pav. Vyrų (kairėje) ir moterų (dešinėje) atlyginimų logaritmų lnwage histogramos

Norėdami nustatyti, ar vyrų asimetrijos koeficientas 0,396 reiškia didelį nuokrypį nuo simetriškumo, sumodeliuokime 1000 normaliųjų ėminių po `length(lnwage[male == 1]) (=893)` narius.

```
aa <- numeric(1000)
for(i in 1:1000)aa[i] <- ask(rnorm(893))
ee <- numeric(1000)
for(i in 1:1000)ee[i] <- eks(rnorm(893))
par(mfrow=c(1,2))
hist(aa)
hist(ee)
```



4.23 pav. Asimetrijos (kairėje) ir eksceso (dešinėje) modeliuotų reikšmių histogramos

Jei lnwage turėtų normalųjį skirstinį, tai asimetrijos koeficiento reikšmė 0,396 būtų neįtikėtina didelė (kaip beje ir eksceso reikšmė 3,51). Vėliau grįšime prie kiekybinių normalumo nustatymo metodų (tai atlieka suderinamumo kriterijai, žr. 10 sk.), o kol kas darome išvadą, kad, nežiūrint išorinio lnwage histogramos panašumo į normalųjį tankį, vyrų atlyginimų logaritmas greičiausiai nėra normalusis a.d.

4.1 pvz. Žemiau esančioje lentelėje pateikti avarių per metus visose 56 JAV atominėse elektrinėse skaičiai:

```

1 0 3 1 4 2 10
6 5 2 0 3 1 5
4 2 7 12 0 3 8
2 0 9 3 3 4 7
2 4 5 3 2 7 13
4 2 3 3 7 0 9
4 3 5 2 7 8 5
2 4 3 4 0 1 7

```

Su Copy+Paste perkeltime šią lentelę į Notepad'ą: ten matysime

```

1 0 3 1 4 2 10
6 5 2 0 3 1 5
4 2 7 12 0 3 8
2 0 9 3 3 4 7
2 4 5 3 2 7 13
4 2 3 3 7 0 9
4 3 5 2 7 8 5
2 4 3 4 0 1 7

```

Pavadinkime šį failą nuclear.txt ir patalpinkime jį į R darbinę direktoriją. Jis turi matricinę struktūrą, todėl skaitydami jį su `read.table`, gautume duomenų sistemą (pabandykite). Kadangi mes norime gauti vektorių, taikysime `scan` funkciją:

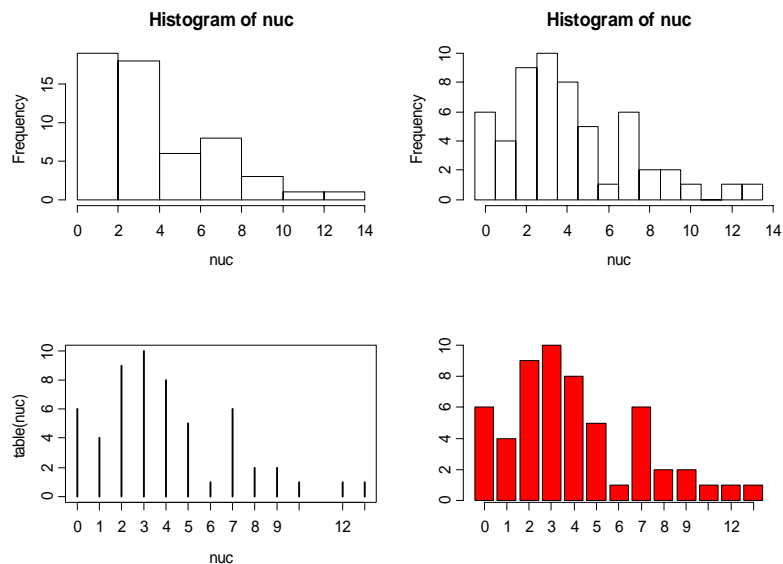
```
nuc<-scan("nuclear.txt")
```

Štai keli grafiniai šios skaitinės informacijos pateikimo variantai.

```

par(mfrow=c(2,2))
hist(nuc)
hist(nuc,breaks=seq(-0.5,13.5))
plot(table(nuc))
barplot(table(nuc))

```



4.24 pav. Keturi avarių skaičiaus “histogramos” variantai

Viršuje kairėje esanti histograma nėra labai “teisinga” skirstinio formos charakteristika, kadangi į pirmąjį stulpelį ji sumeta net tris⁵ reikšmes: 0, 1 ir 2. Iš tikrųjų,

```
> hist(nuc)$breaks
[1] -0.0000014  2.0000014  4.0000014  6.0000014
[5]  8.0000014 10.0000014 12.0000014 14.0000014
```

Avarijų skaičiaus skirstinio formą “teisingai” pateikia viršuje dešinėje esanti histograma, tačiau jos sintaksė gana komplikauta. Panašų paveikslą gauname ir su `barplot` komanda (žr. grafiką apačioje dešinėje), tačiau iš jo sunku suprasti, kad 11 avarijų nebuvo nei vienoje elektrinėje (`barplot` yra pirmiausiai skirta vardiniams kintamiesiems – tokiu atveju 10, 11 ir 12 yra tiesiog vardai). Ko gero paprasčiausiai reikalingą grafiką gauname su `plot(table(nuc))` komanda (grafikas apačioje kairėje).

Kadangi avarijų skaičiaus skirstinys turi sunkią dešiniąją uodegą, vidurkis bus didesnis už medianą.

```
> mean(nuc)
[1] 4.035714
> median(nuc)
[1] 3
```

Kuris iš šių skaičių yra “teisingesnė” centro charakteristika? Kadangi tai priklauso nuo požiūrio, geriausiai pateikti juos abu (matome, kad jie pastebimai skiriasi, taigi skirstinys asimetriškas). Visuomenės nuomone kiekviena avarija pavojinga, todėl didelis vidurkis yra blogai (nes `mean(nuc)*56` reiškia bendrą avarijų skaičių). Antra vertus, vyriausybė žino, kad 12 ir 13 avarijų⁶ buvo dviejose naujose elektrinėse, tiek avarijų jose kitais metais nebus ir todėl geriau į jas nekreipti didelio dėmesio (mediana (beje, ir “nupjautas” (=trimmed) vidurkis) kaip tik ir nekreipia!). Pažymėsime, kad JAV vyriausybė statistika pateikia kasmetinių gyventojų pajamų medianą, nes šimto turtingiausių visuomenės narių pajamos aiškiai iškreipia bendrą vaizdą (t.y., vidurki)⁷. Apskritai, jei teigiamas a.d. turi sunkią (dešinę) uodegą, tai mediana yra yra stabilesnis įvertis (turi mažesnę dispersiją) ir todėl labiau priimtinas.

4.8 UŽDUOTIS. Parašykite dvi funkcijas `apr.st` ir `Apr.st`, kurios pateiktų skaitinio kintamojo x aprašomąsias statistikas taip, kaip pavaizduota žemiau:

```
> apr.st(rnorm(100))
Obs= 100 , Mean= 0.041 , Std.Dev.= 0.951 , Min= -2.605 , Max= 2.224

> Apr.st(rnorm(100))
      Obs      Mean Std.Dev.      Min      Max
100.000   0.031   1.010   -2.770   2.156
```

⁵ Čia problema ta, kad kairysis histogramos taškas 0 (kuris bus priskirtas pirmajam intervalui) kartojasi net šešis kartus. Histograma pirmiausiai yra skirta tolydiems atsitiktiniams dydžiams – tuomet šansai, kad viena kokia reikšmė pasikartos, yra nedidelė (teoriškai to iš vis neturėtų būti, tačiau dėl matavimo rezultatų apvalinimo taip gali įvykti).

⁶ Jas galima interpretuoti kaip išskirtis – patikrinkite su `boxplot`.

⁷ Profesionaliose futbolo komandose dažnai būna keli labai gerai apmokami žaidėjai. Komandos savininkams labiau rūpi vidutinė alga (nes ji nusako jų išlaidas - kodėl?), tačiau eiliniams žaidėjams labiau rūpi atlyginimų mediana (kodėl?).

4.9 UŽDUOTIS. Bibliotekoje Simple (ją galima atsisiųsti iš http://www.math.csi.cuny.edu/Statistics/R/simpleR/Simple_0.4.zip) yra duomenų rinkinys `bumpers`. Kas tai per duomenys? Kaip galima sužinoti, kad tai (duomenų) sąrašas? Transformuokite jį į skaitinį vektorių su vardais ir išbrėžkite histogramą. Pabandykite atspėti vidurkį, modą ir standartą. Savo hipotezes patikrinkite su tinkamomis R funkcijomis.

4.10 UŽDUOTIS. Aukčiau pateiktą užduotį atlikite su kitais Simple duomenų rinkiniais `firstchi` ir `math`.

4.11 UŽDUOTIS. Pakete `car` yra duomenų rinkinys `Davis`. Apskaičiuokite jo komponentės `repwt` vidurkį ir dispersiją. Kadangi ten kai kurių duomenų trūksta, galbūt teks pasinaudoti opcija `mean(x, na.rm=TRUE)` arba `mean(x[!is.na(x)])`. Ar trukdo trūkstami duomenys išbrėžti histogramą? Norėdami atsakyti į pastarąjį klausimą, panagrinėkite funkcijos `hist` programą su

```
> hist; methods(hist); hist.default
```

4.2.5. Funkcija `eda.shape`

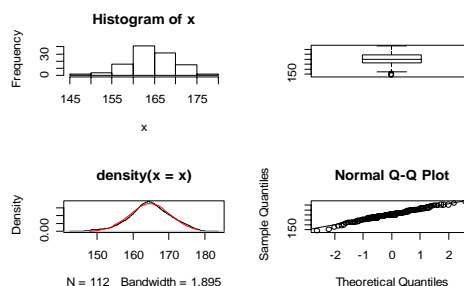
Aukščiau aptarėme įvairias grafines ir skaitines ėminio charakteristikas. Visas šias procedūras tikslinga pateikti viena funkcija, kurią pavadinsime `eda.shape`:

```
eda.shape <- function(x){
  opar <- par(mfrow=c(2,2)); on.exit(par(opar))
  hist(x); boxplot(x)
  plot(density(x)); x <- sort(x)
  lines(x, dnorm(x, mean(x), sd(x)), col=2)
  qqnorm(x); qqline(x)
  cat("Vidurkis=", mean(x), ", Mediana=", median(x),
      "(simetriniu atveju turi beveik sutapti)", # Tekstą papildome komentaru
      "\nStandartas=", sd(x), ", MAD=", mad(x), # Simbolis "\n" nurodo: toliau
      "(kai nera isskirciu, turi beveik sutapti)", # esanti tekstą spausdinti iš
      "\nAsimetrijos koeficientas=", ask(x), # naujos eilutės
      "(simetriniu atveju turi buti 0)",
      "\nEkscesas=", eks(x), "(normaliuoju atveju turi buti 3)\n")
}
```

Štai jos taikymo pavyzdys:

```
> eda.shape(hF)
Vidurkis= 164.7143 , Mediana= 165 (simetriniu atveju turi beveik sutapti)
Standartas= 5.659129 , MAD= 5.9304 (kai nera isskirciu, turi beveik sutapti)
Asimetrijos koeficientas= -0.2311667 (simetriniu atveju turi buti 0)
Ekscesas= 3.152079 (normaliuoju atveju turi buti 3)
```

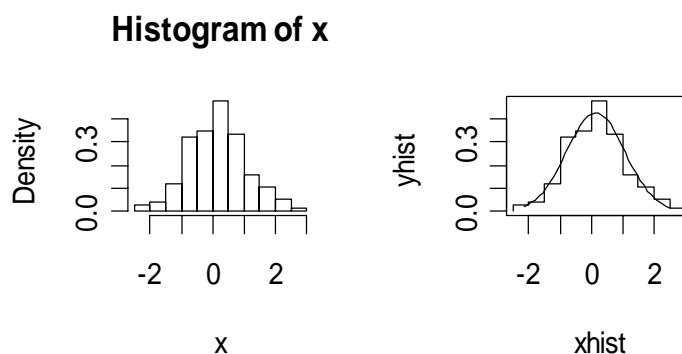
Be šio teksto dar matysime keturis grafikus:



4.25 pav. Funkcijos `eda.shape` pateiktas vaizdas

4.12 UŽDUOTIS. Štai, du žemiau esančius grafikus brėžiančios, programos pradžia:

```
set.seed(15)
x<-rnorm(150)
par(mfrow=c(1,2))
h<-hist(x,breaks=15,freq=F)
```



4.26 pav. Žr. 4.12 užduotį.

Pabaikite šią programą (dešinėje yra ta pati, bet kitaip pateikta histograma, bei normaliojo tankio, nusakomo šio ėminio empiriniais parametrais, grafikas). *Nuroda*. Pasinaudokite funkcija `plot(..., type="s")`.

4.13 UŽDUOTIS. Pateiksime procedūrą, kuri generuoja vadinamąjį dvimodalinį (t.y. toki, kurio tankis turi dvi modas (lokaliuosius maksimumus)) skirstinį.

```
two.mod <- function(n){x <- numeric(n)
for(i in 1:n) if (runif(1)<0.3) x[i]<-rnorm(1) else x[i]<-rnorm(1,
mean=5)
x}
```

Generuokime 1000 šių atsitiktinių skaičių⁸ ir išbrėžkime jų histogramą.

```
par(mfrow=c(1,2))
y<-two.mod(1000)
hist(y,prob=T)
lines(density(y))
> median(y)
[1] 4.445219
```

Kaip toli yra šis medianos įvertis nuo tikrosios? Aišku, kad aprašytąjį modeliavimo procedūrą galime pakartoti daug kartų ir taip iširti, tarkime, skirstinio medianos kintamumą (pvz., rasti jos pasikliauties intervalą arba bent įvertinti jos tarpkvartilinį atstumą IQD⁹). Deja, praktikos uždaviniuose paprastai turime tik vieną ėminį ir teoriškai įvertinti mūsų (aiškiai nenormalaus) skirstinio medianos IQD nėra lengva. Tokiu atveju galima taikyti vadinamąjį butstrepo (= bootstrap (angl.)) metodą, kuris siūlo imti naujus ėminius ne iš visos populiacijos, bet tik iš turimo ėminio. Galima įrodyti, kad tam tikromis sąlygomis abu metodai duoda panašų rezultatą.

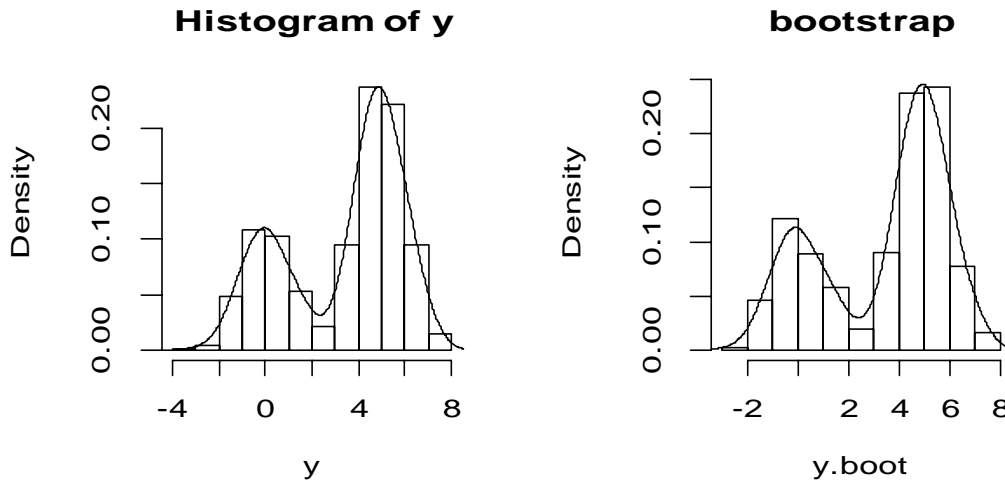
⁸ Tai vadinamasis dviejų normalių tankių mišinys

⁹ Priminsime – tai skirtumas tarp 3-iojo ir 1-ojo kvartilų.

```

> y.boot <- sample(y,1000,replace=T)
> hist(y.boot,main="bootstrap",prob=T)
> lines(density(y.boot))
> median(y.boot)
[1] 4.338387

```



4.27 pav. Originaliojo ėminio y (kairėje) ir vienos butstrepinės kopijos $y.boot$ (dešinėje) histogramos

O dabar UŽDUOTIS. i) Generuokite 500 ėminių po 600 skaičių su `two.mod` funkcija. Apskaičiuokite kiekvieno ėminio medianą ir raskite jos IQD. ii) Paimkite dar vieną ėminį, sudarykite 500 jo butstrepinių kopijų ir apskaičiuokite medianos IQD. Palyginkite abu įverčius.

4.14 UŽDUOTIS. $n=15$ kartų pakartokime Bernulio eksperimentus su sėkmės tikimybe $p=0,3$. Tokio eksperimentų sėkmių skaičių X galima modeliuoti arba su funkcija `rbinom(1,15,0.3)` arba su `sum(sample(c(0,1),15,replace=TRUE,prob=c(0.7,0.3)))` (kuo skiriasi šios dvi procedūros?). Teorinę X skirstinio lentelę palyginkite su empirine (skaitmeniškai ir grafiškai).

4.15 UŽDUOTIS. Sugeneruokite du Košy atsitiktinių skaičių ėminius. Ar bus jų histogramos vienodos?

```

set.seed(2)
x1<-rcauchy(100)
x2<-rcauchy(100)
par(mfrow=c(1,2))
hist(x1)
hist(x2)

```

Papildykite `hist` funkcijas tokiais parametrais, kad abi histogramos turėtų tuos pačius dalinimo taškus.

4.16 UŽDUOTIS. Pakete `Simple` (plg. 4.5 užd.) yra daug duomenų rinkinių. Be tradicinio būdo atidaryti duomenų rinkinį `babies` (su `data(babies)`) galime pasiūlyti dar vieną: komandiniame lange, spragtelėkite ant kairėje viršuje esančio `File|Source R code...`, atsidariusiame `Select file to source` lange nuvairuokite į `...\\rw1071\\library\\Simple\\data\\babies.R` ir spragtelėkite ant `Open`.

```

> source("C:/Program Files/R/rwl071/library/Simple/data/babies.R")
> babies[1:5,]
  bwt gestation parity age height weight smoke
1 120      284     0  27   62   100     0
2 113      282     0  33   64   135     0
3 128      279     0  28   64   115     1
4 123       NA     0  36   69   190     0
5 108      282     0  23   67   125     1
> par(mfrow=c(1,2))
> hist(babies$weight)
# O štai antras variantas:
> attach(babies)
> hist(weight)
# Dar vienas variantas:
> hist(weight)
> barplot(weight)

```

Ar simetriškas `weight` skirstinys? Ar turi jis išskirčių? Kuri uodega sunki? Nubrėškite suglodingą histogramą. Atlikite tas pačias užduotis su kitais kintamaisiais iš `babies`.

4.17 UŽDUOTIS. Naudodamiesi funkcijomis `floor` ir `runif`, generuokite 1000 atsitiktinių skaičių, turinčių tolygų skirstinį skaitmenų aibėje 0, 1, ..., 9. Tą patį atlikite su funkcija `sample`. Palyginkite skaitmenų dažnius abiem atvejais. *Nuoroda.* Funkcija `runif` generuoja tolygiai intervale [0,1] pasiskirsčiusius skaičius x_i , t.y., pvz., $n(\text{runif}(1000) \in [0,3;0,4])/1000 \approx 1/10$ arba, kas ekvivalentu, $n(\text{floor}(10 * \text{runif}(1000)) = 3)/1000 \approx 1/10$.

4.18 UŽDUOTIS. Duomenų rinkinys `huron` yra `R1\Data\Maindonald` direktorijoje.

```

> huron
  year mean.height
1  1860      581.56
2  1861      581.55
3  1862      581.34
*****

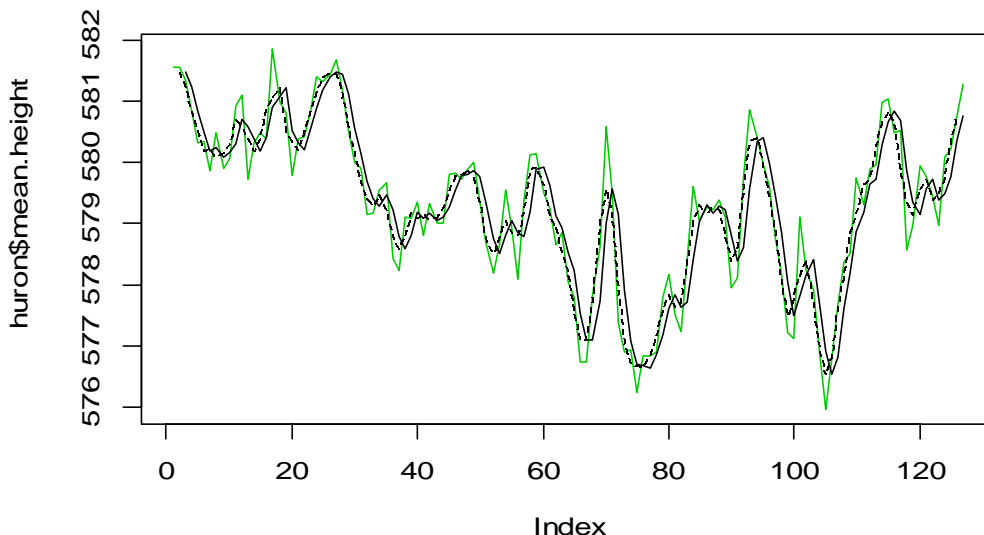
```

Šiame rinkinyje pateikti daugiamečių Hurono ežero lygio stebėjimų rezultatai. Ežero lygis kinta gana chaotiškai (žr. 4.28 pav., žalia linija), todėl norint geriau suvokti tendencijas, duomenis reiktų suglodingi. Tai galima atlikti įvairiai. Vadinamasis slenkamojo vidurkio metodas siūlo lygį $h(t)$ pakeisti gretimų reikšmių vidurkiu, pavyzdžiui, $(h(t-2) + h(t-1) + h(t))/3$ (juoda linija) arba $(h(t-1) + h(t) + h(t+1))/3$ (juoda trūki linija). 4.28 pav. išbrėžtas naudojant tokias komandas:

```

> plot(huron$mean.height, type="l", col=3)
> library(ts)
> lines(filter(huron$mean.height, rep(1, 3), sides = 1)/3)
> lines(filter(huron$mean.height, rep(1, 3))/3, lty=2)

```



4.28 pav. Hurono ežero lygio kitimo grafikas (žalia linija) ir keli sugludinto ežero lygio grafikai (juoda linija)

(Beje, kaip atrodytų paveikslas, jei glodintume ne pagal 3, o pagal 15 taškų?). Parašykite dvi savo funkcijas (viena su `for` ciklu, kita – be jo), kurios atliktu tokias glodinimo procedūras su vektoriumi x . *Nuoroda*. Glodinant vektorių x be ciklo pagal 3 taškus, gali praversti tokia eilutė:

```
cbind(c(NA,NA,x),c(NA,x,NA),c(x,NA,NA))
```

4.19 UŽDUOTIS. Testą sudaro 20 klausimų, į kuriuos reikia atsakyti Taip arba Ne. Parašykite R funkciją, kuri imituotų atsitiktinai spėliojantį studentą. Pakeiskite šią funkciją taip, kad klausimų skaičius būtų bet koks. Pakartokite šį testą 1000 kartų, išbrėžkite teisingų atsakymų skaičiaus histogramą (o gal geriau ne histogramą?), apskaičiuokite empirinį vidurkį ir dispersiją. Palyginkite su teorinėmis šių parametru reikšmėmis.

4.20 UŽDUOTIS. Atlikite 4.19 užduotį, jei vieną teisingą testo atsakymą reikia pasirinkti iš 5 pateiktų variantų.

4.21 UŽDUOTIS. Gaisrų skaičius Vilniaus rajone turi Puasono skirstinį su vidurkiu 1,2 darbo dienomis ir su vidurkiu 1,5 nedarbo dienomis. Imituokite 52 savaitių gaisrų suvestinę. Apskaičiuokite kiekvienos savaitės dienos empirinį gaisrų vidurkį. Papuoškite savo ataskaitą keliais grafikai.

4.22 UŽDUOTIS. Nagrinėkime du duomenų rinkinius.

```
set.seed(1); ru<-runif(40)
```

ir

```
set.seed(1); re<-rexp(30,1)
```

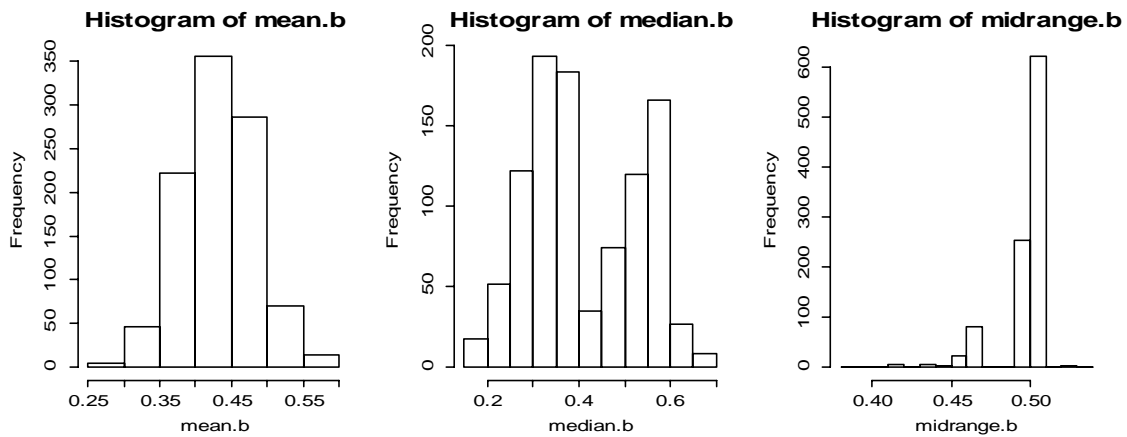
Rinkinio “centro” padėtį galima nustatyti, remiantis įvairiais principais, pvz., tai galėtų būti


```

mean(ru)
median(ru)
(max(ru)+min(ru))/2 # Angliškai ši statistika vadinama midrange
(Q1+2Q2+Q3)/4     # [ČM1, 39 psl.] tai vadinama triskaičiu vidurkiu

```

Vienas svarbiausių įverčiui keliamų reikalavimų, yra jo tikslumas. Jei ėminio skirstinys būtų žinomas, kiekvieno iš aukščiau pateiktų įverčių tikslumą (pvz., dispersiją) galima būtų įvertinti teoriškai. Tardami, kad ru ir re kilmė nėra žinoma, generuokite po 999 kiekvieno įverčio dydžio 40 butstrepo ėminių (tai atlikite su a) `sample` ir b) `boot` funkcija iš `boot` paketo) ir palyginkite histogramas ir IQD. Kuris įvertis tikslesnis?



4.29 pav. ru ėminio atveju tiksliausia trečioji statistika

4.23 UŽDUOTIS. Alisa ir Bilas tarnauja kariniuose daliniuose, dislokuotuose vienoje iš penkių valstijų: Kanzase, Kolorade, Misisipėje, Nebraskoje ir Oklahomoje. Šiuo metu Alisa tarnauja Nebraskoje, o Bilas – Oklahomoje. Kiekvieną mėnesį jie abu yra pervedami tarnauti į atsitiktinai parinktą vieną iš gretimų valstijų. Pavyzdžiui, Alisa po mėnesio su tikimybe $1/3$ tarneaus Kolorade, Kanzase arba Misisipėje.



4.30 pav. Valstijų išsidėstymo schema; Alisa šiuo metu yra Nebraskoje, o Bilas - Oklahomoje

Apskaičiuokite mėnesių, kurie praeis iki to momento, kai Alisa ir Bilas susitiks vienoje valstijoje, vidurkį ir dispersiją. Pažymėsime, kad tikslus sprendimas yra gana komplikotas (žr. P. Грэхем, Д. Кнут, О. Паташник, Конкретная математика, стр. 626, 8.32); mėnesių skaičiaus vidurkis lygus $75/16=4,69$, o dispersija - $105/4=26,25$. Dėl šios priežasties atsakymą rasime Monte Carlo metodu.

```

Randez <- function()
{
# Funkcija Randez
s.nr<-numeric(1000) # 1000 repliku
for(i in 1:1000)
{
A<-4 # Alisa šiuo metu 4-ojoje valstijoje
B<-5 # Bilas šiuo metu 5-ojoje valstijoje
s<-0
while(A!=B)
{
if(A==1) A<-sample(2:5,1) else if(A==2|A==3) A<-sample(c(1,4,5),1) else
A<-sample(1:3,1) # A yra valstijos, į kurią bus perkelta Alisa, numer.
if(B==1) B<-sample(2:5,1) else if(B==2|B==3) B<-sample(c(1,4,5),1) else
B<-sample(1:3,1)
print(c(A,B))
s<-s+1
}
s.nr[i]<-s # i-osios replikos mėnesių iki susitikimo skaičius
cat(paste("s.nr(",i,")="),s.nr[i],"\n")
}
hist(s.nr,prob=T)
print(list(vidurkis=mean(s.nr),dispersija=var(s.nr)))
#s.nr
}

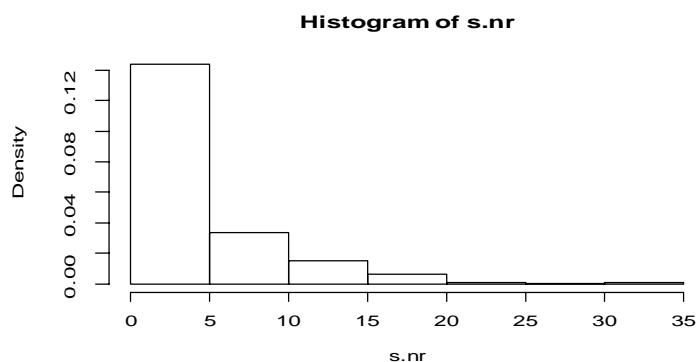
```

```
> Randez()
```

```

.....
[1] 3 3
s.nr( 996 )= 1 # Alisa ir Bilas susitiko po 1 mėnesio 3-joje valst.
[1] 2 3
[1] 1 4
[1] 2 2
s.nr( 997 )= 3 # Ši kartą jie susitiko po 3 mėn. 2-joje valst.
[1] 1 2
[1] 5 4
[1] 1 2
[1] 5 5
s.nr( 998 )= 4
[1] 2 1
[1] 5 3
[1] 3 5
[1] 1 2
[1] 4 5
[1] 2 3
[1] 1 4
[1] 5 2
[1] 1 4
[1] 2 2
s.nr( 999 )= 10
[1] 2 1
[1] 4 3
[1] 3 4
[1] 4 1
[1] 3 5
[1] 5 3
[1] 2 1
[1] 4 5
[1] 2 2
s.nr( 1000 )= 9

```



4.31 pav. Mėnesių iki susitikimo skaičiaus histograma

```

$vidurkis
[1] 4.486
$dispersija
[1] 29.76157

```

1) Išsiaiškinkite Randez funkciją. 2) Mėnesių iki susitikimo skaičiaus s .nr skirstinys yra labai nesimetriškas, todėl jo vidurkio 95% pasikliauties intervalą raskite butstrepo metodu (žr. 4.13 ir 4.22 užduotis). Ar tikroji vidurkio reikšmė priklauso šiam intervalui?

4.24 UŽDUOTIS. Vadinamoji 3σ taisyklė tvirtina, kad (iš tikrųjų neappręžto) normaliojo a.d. X reikšmės praktiškai visuomet priklauso apręžtam intervalui $(EX - 3\sqrt{DX}, EX + 3\sqrt{DX})$. 1) apskaičiuokite šią tikimybę, 2) generuokite 500 normalių a.d. ir raskite koks procentas šio ėminio narių yra intervale $(\bar{x} - 3s, \bar{x} + 3s)$, 3) tą patį atlikite su eksponentiniu skirstiniu ir su Puasono skirstiniu (imkite $\lambda = 0.09$).

4.25 UŽDUOTIS. Jei kai kurios kintamojo reikšmės kartojasi kelis kartus, rezultatai kartais pateikiami grupuotu pavidalu (žr. žemiau `nor` arba `nnor`). Štai kelios eilutės, kurios sukurs dirbtinius tokio pavidalo duomenis.

```

set.seed(1)
nor_table(round(rnorm(30,100,1),0))
nor
# 98 99 100 101 102 103
# 3 6 11 7 2 1
nor1_as.numeric(names(nor))
nor1
#[1] 98 99 100 101 102 103
nor2_nor
dimnames(nor2)_NULL
nor2
#[1] 3 6 11 7 2 1
nnor_cbind(nor1,nor2)
> nnor
      nor1 nor2
[1,]   98    3
[2,]   99    6
[3,]  100   11
[4,]  101    7
[5,]  102    2
[6,]  103    1

```

- Apskaičiuokite `nnor` vidurkį ir standartą (jie lygūs, atitinkamai, $vid=100,0667$ ir $sta=1,201532$);
- Naudodamiesi funkcija `integrate`, apskaičiuokite integralą

$$\int_{90}^{110} x \exp(-(x - vid)^2 / 2sta^2) dx .$$

Išbrėžkite pointegrinės funkcijos grafiką.

4.26 UŽDUOTIS. Kompakto R1 failuose `Data\Misc\d-olive-test.txt` ir `Data\Misc\d-olive-train.txt` yra pateikti duomenys apie Italijos alyvų aliejus. Kiekvienas aliejus buvo

charakterizuojamas gamybos rajonu (Region - tai South=1, Sardinia=2 ir North=3), šio rajono parajonių (Area) ir aštuonių riebiųjų rūgščių (palmitic, palmitoleic ir t.t.) kiekiu (%×100) šiame aliejuje. Tyrimo¹⁰ tikslas buvo išsiaiškinti ar galima pagal rūgščių kombinaciją nustatyti aliejaus kilmės vietą. Mes apsiribosime aprašomąja šių duomenų analize. Apjunkite šiuos du failus į vieną (ko gero tai bus duomenų rinkinys Data\Misc\olive-dat.txt - patikrinkite) ir nusiskaitykite. Skaitines charakteristikas (visiems rajonams kartu ir kiekvienam rajonui atskirai) pateikite maždaug tokiu pavidalu:

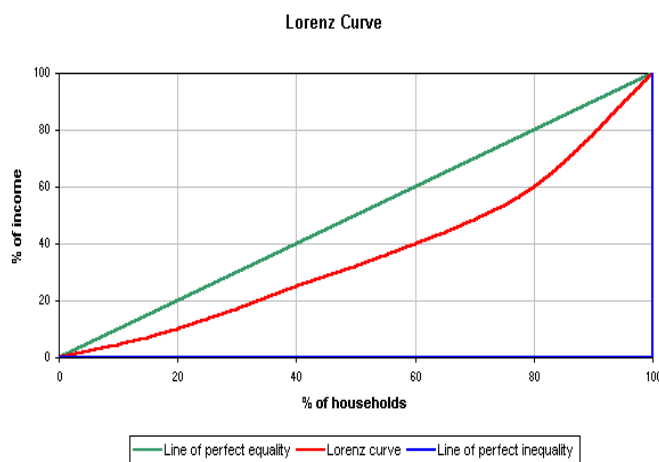
Visiems rajonams:

	palmitic	palmitoleic	eicosenoic
min	610	15.00		1.00
median	1201	110.00		17.00
mean	1232	126.10		16.28
max	1753	280.00		58.00
std.dev	168.6	52.5		14.1

Dabar tą patį apskaičiuokite atskirai pietiniams rajonams (Region=1), Sardinijai (Region=2) ir šiaurės rajonams (Region=3). Kiekvienam rajonui išbrėžkite jo alyvų įvairių rūgščių vidurkius (plg. 6.3 pav.).

4.27 UŽDUOTIS. Kompakto R1 faile Data\Misc\Minrain.csv yra pateikti daugelio metų kritulių kiekiai. Nusiskaitykite šiuos duomenis, išbrėžkite jų histogramą ir kvantilių grafiką. Ar duomenys panašūs į normaliuosius? Gal Box – Cox'o transformacija (žr. 4-4 psl.) galėtų pagerinti duomenų normalumą? Paeksperimentuokite.

4.28 UŽDUOTIS. Tarkime, kad visų šalies namų ūkių (= household (angl.)) pajamos X yra vienodos ir lygios c (variantas: visi akcininkai turi vienodą akcijų skaičių). Jei šiuo "visuotinos lygybės" atveju surinktume duomenis apie minėtas pajamas, tai visi ėminio elementai x_i būtų lygūs c . Antra vertus, "visiškos nelygybės" atveju visos šalies pajamos (variantas: visos AB akcijos) priklausytų vienam namų ūkiui (kitų pajamos būtų lygios nuliui). Aišku, kad tarpinių variantų yra daug, o vienas iš būdų pademonstruoti visuomenės "nelygumą" yra Lorenz'o kreivė ir Gini'o koeficientas.



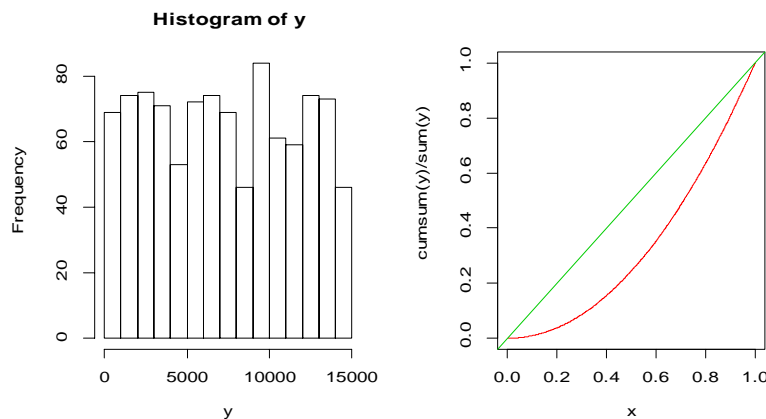
4.32 pav. Trys Lorenz'o kreivės

¹⁰ Žr. <http://www.public.iastate.edu/~dicook/stat503.html>, failą cs_olive.pdf.

(Raudonąją (žr. 4.32 pav.)) Lorenz'o kreivę reiktų interpretuoti taip. Tarkime, x_1, x_2, \dots, x_n yra visų šalies namų ūkių pajamos (šių skaičių suma lygi visam šalies produktui). Jei susumuotume 20% skurdžiausių namų ūkių pajamas, tai ši suma sudarytų tik maždaug 12% viso produkto, o jei 60% - tai 40% viso produkto ir t.t.

Štai kelios eilutės, kurios brėžia kreivę panašią į raudonąją.

```
y <- sort(runif(1000,0,15000))
hist(y)
x <- (1:length(y))/length(y)
plot(x,cumsum(y)/sum(y),type="l",col=2)
abline(0,1,col=3)
```



4.33 pav. 1000-čio namų ūkių pajamos yra tolygiai pasiskirsčiusios intervale $[0, 15000]$ (kairėje); tai nėra (kodėl?) “lygybės visuomenė” (žr. Lorenz'o kreivę dešinėje)

Išbrėžkite dar dvi Lorenz'o kreives, kurios atitiktų atvejus, kai ūkių pajamos turi a) normalųjį skirstinį su vidurkiu 0 ir standartu 5000 (ėminių sudarykite tik iš teigiamų šio objekto elementų) ir b) Pareto skirstinį su parametru $c = 2$.

Pajamų “nelygybę” galima charakterizuoti vadinamuoju Gini'o koeficientu G – skaitmeniškai jis lygus plotui tarp žalios ir raudonos linijos (jei $G = 0$ – “visi lygūs”, jei $G = 1$ – visiška nelygybė). Apskaičiuokite a) ir b) atvejų Gini'o koeficientus. Kuri visuomenė “nelygesnė”? Kaip tai susiję su a) ir b) skirstinių savybėmis?

4.29 UŽDUOTIS. Lorenz'o kreivę galima apibrėžti ne tik ėminiams, bet ir populiacijoms. Tiksliau kalbant, jei namų ūkio pajamos nusakomas skirstiniu F_X , tai šios populiacijos Lorenz'o funkcija apibrėžiama taip:

$$L_X(p) = \frac{1}{EX} \int_0^p F_X^{-1}(u) du, \quad p \in (0,1)$$

(čia F_X^{-1} yra funkcija atvirkštinė skirstinio funkcijai (kitai sakant, tai kvantilių funkcija - pakete R ją skaičiuoja funkcijos `qunif`, `qpareto` ir panašiai)). Įrodykite, kad Lorenz'o funkcija turi savybes:

a) $L_X(0) = 0,$

- b) $L_X(1) = 1$,
- c) $L_X'(p) \geq 0$,
- d) $L_X''(p) \geq 0$ (t.y., Lorenz'o kreivė visuomet yra žemiau tiesės $y = x$ (kodėl?)).
- e) Įrodykite, kad Gini'o koeficientas $G = 1 - 2 \int_0^1 L_X(p) dp$.
- f) Kaip Gini'o koeficientas Pareto skirstiniams priklauso nuo c ?

Išbrėžkite Lorenz'o kreivę, kai X turi tolygų skirstinį intervale $[0,1]$; raskite šio skirstinio Gini'o koeficientą.

4.30 UŽDUOTIS. Duomenų rinkinio attenu:

```
> data(attenu)
> attenu
  event mag station  dist accel
1     1  7.0    117  12.0 0.359
2     2  7.4   1083 148.0 0.014
*****
11    2  7.4    117 370.0 0.004
12    3  5.3   1117   8.0 0.127
13    4  6.1   1438  16.1 0.411
14    4  6.1   1083  63.6 0.018
15    4  6.1   1013   6.6 0.509
*****
```

pirmajame stulpelyje event yra pateiktas žemės drebėjimo Kalifornijoje numeris (nuo 1 iki 23). Kai kuriuos iš jų užregistravo tik viena stotis (pvz., 1-ąjį), tačiau kitus – kelios stotys (pvz., 2-ąjį užregistravo dešimt stočių). Komanda

```
> table(attenu$event)
 1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23
1 10  1  9 11  1  1  5 22  1  3  1  2  4  4  3  3 11 38 16  7 10 18
```

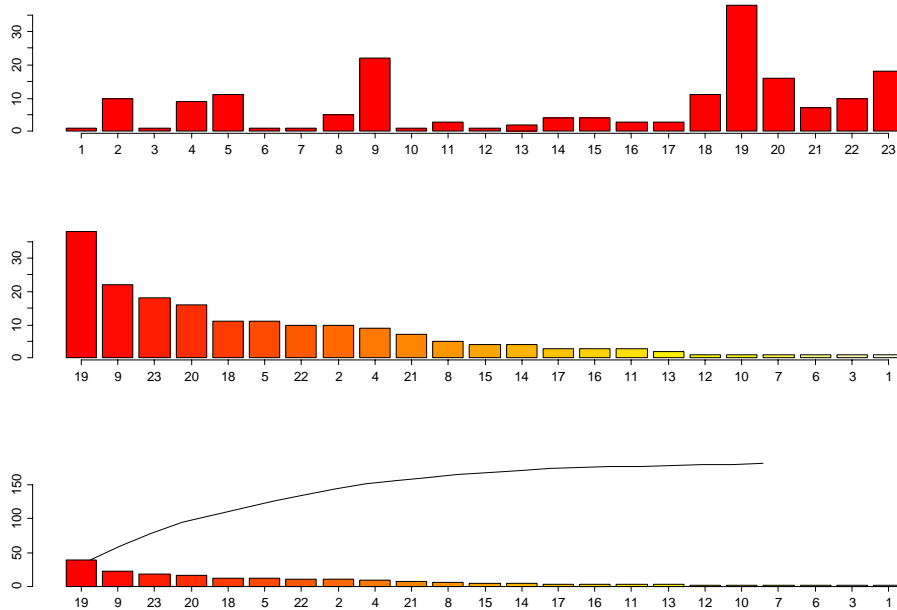
informuoja apie kiekvieno drebėjimo registracijų skaičių, tačiau gerai būtų šią informaciją pateikti vaizdžiau. Komanda

```
> barplot(table(attenu$event)) # Plg. 4.34 pav. viršutinį grafiką
```

pateikia aukščiau užrašytą informaciją grafiškai, o komanda

```
> barplot(rev(sort(table(attenu$event)))) # 4.34 pav., viduryje
```

brėžia tų pačių duomenų Pareto grafiką (žr. [ČM, 55 psl.]). Iš šio grafiko jau aiškiai matome, kad daugiausiai kartų buvo užregistruotas 19-asis žemės drebėjimas, toliau eina 9-asis ir t.t.



4.34 pav. attenuševent stulpelinė (viršuje) ir dvi Pareto diagramos

Parašykite kelias komandas, kurios brėžtų apatinį 4.34 pav. grafiką (čia kreivė viršuje yra sukaupieji stebėjimų dažniai).

4.31 UŽDUOTIS. Išskirtys (outliers) dažnai signalizuoja apie tai, kad ėminyje yra “nereguliarių” stebėjimų arba “klaidų”. Nėra standartinių rekomendacijų, ką daryti su išskirtimis (dažnai išskirtys tiesiog atmetamos, tačiau jei keli taškai prieštarauja mūsų modeliui, tai gal tiesiog modelis blogas¹¹). Kaip ten bebūtų, pabandykime aptarti, ką vadinti išskirtimis ir kaip jų efektą susilpninti. Dažnai teikiama rekomendacija atmesti duomenis, kurie netelpa į intervalą $mean \pm 3sd$ (trijų sigmų taisyklė); deja ji nėra visuomet gera, nes tuomet, kai turime išskirčių kaip vidurkis taip ir standartas gali būti toli nuo “tikrųjų”. Tokiu atveju tikslingiau naudotis “atspariomis” statistikomis, pvz., mediana ar MAD (žr. 4-15 psl.).

Nagrinėkime duomenų vektorių x

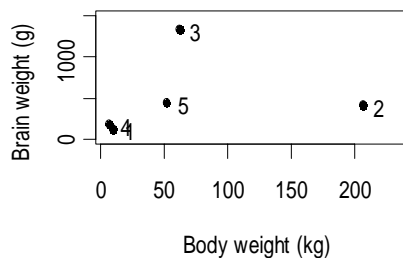
0.378	-22.831	1.123	-10.789	0.191	0.168	0.552	1.192	1.767	0.170
-1.108	-1.714	6.045	-1.424	-1.226	-26.948	1.875	-2.059	-0.491	0.010
-9.526	0.216	-0.479	-27.457	1.148	-3.852	-0.859	-1.338	-0.037	0.912
0.831	0.132	-0.887	0.144	0.042	3.145	-0.546	321.202	0.612	0.067
1.910	0.952	-0.738	-4.508	23.327	0.564	3.658	0.885	-0.116	5.232
0.205	-3.820	-0.236	-0.386	0.727	-3.817	-2.584	-1.514	20.013	-0.160
-0.374	-0.183	-0.368	1.156	-2.995	-1.222	0.041	-0.589	0.993	0.223
0.043	-2.175	0.576	0.540	-0.179	-4.750	0.585	-1.999	-0.830	-0.653
4.030	1.472	-0.302	0.509	3.809	1.811	-19.947	-1.860	4.351	0.058
1.021	1.986	1.389	-0.136	0.459	0.766	-17.920	1.940	1.646	1.211

Keturiuose polanguose išbrėžkite keturias stačiakampes diagramas: 1) pačių duomenų x , 2) rinkinyje x palikite tik duomenis telpančius į intervalą $[mean(x) - 3 * sd(x), mean(x) + 3 * sd(x)]$, 3) rinkinyje x palikite tik duomenis tarp 0,025-ojo ir 0,975-ojo kvantilių, 4) teorija rekomenduoja rinkinyje x palikti tik duomenis iš intervalo $[median(x) - 3,5 * mad(x), median(x) + 3,5 * mad(x)]$.

¹¹ Teigiama, kad ozono “skylė” virš Antarktidos buvo taip ilgai nepastebėta dėl to, kad stebėjimų “išskirtys” buvo automatiškai atmetamos.

4.32 UŽDUOTIS. Su File|Source R code... nuskaitykite duomenų rinkinį primates.R iš Data/Maindonald.

```
> primates
  Species Bodywt Brainwt
1 Potar monkey 10.0 115
2 Gorilla 207.0 406
3 Human 62.0 1320
4 Rhesus monkey 6.8 179
5 Chimp 52.2 440
```



Šių primatų kūno ir smegenų svorio sąryšį pavaizduosime grafiškai.

```
attach(primates)
plot(x=Bodywt, y=Brainwt, pch=16, xlab="Body weight (kg)",
     ylab="Brain weight (g)", xlim=c(5,240), ylim=c(0,1500))
chw <- par()$cxy[1]
text(x=Bodywt+chw, y=Brainwt, labels=row.names(primates), adj=0)
```

Pakeiskite šį grafiką taip, kad taškų etiketės būtų ne eilės numeris, bet primato vardas. Galimas daiktas, kad horizontaliai užrašyti vardai netilps, todėl pasukite juos 45 laipsnių kampu (parametras `str`). Panašiai įveskite dolphins.R duomenis ir išbrėžkite grafiką

```
attach(dolphins)
plot(logweight, logheart)
```

Kaip nors “protingai” pažymėkite `styx` ir `delph` rūšis.

5. Dvimačiai duomenys: aprašomoji statistika ir

duomenų priešanalizė

Kaip ir vieno kintamojo atveju, nagrinėjant du kintamuosius, paprastai kyla klausimai apie kiekvieno jų skirstinio formą ar parametrus. Antra vertus, dabar atsiranda nauja tema apie jų tarpusavio ryšius. Jos sprendimas priklauso nuo kintamųjų tipo. Ar išsilavinimo lygis priklauso nuo lyties? (abu kintamieji vardiniai). Ar priklauso atlyginimas nuo lyties? (pirmasis kintamasis skaitinis, antrasis vardinis). Ar priklauso atlyginimas nuo patyrimo (metais)? (abu kintamieji skaitiniai). Šiems bei panašioms klausimas ir skirtas 5 skyrius.

5.1. Vardiniai kintamieji

bwages duomenų rinkinyje turime informacijos apie tiriamųjų asmenų lytį male ir išsilavinimą educ. Ar galima teigti, kad moterys (o gal vyrai?) yra labiau išsilavinę?

```
> attach(bwages)
> (me <- table(male,educ))
  educ
male 1   2   3   4   5
     0  23  70 162 192 132 # Tai vadinamoji (požymių) sąveikos
     1  76 195 258 164 200 # lentelė (contingency table)
> (em <- table(educ,male))
  male # Kitaip pateikta sąveikos lentelė
educ 0   1
     1  23  76
     2  70 195
     3 162 258
     4 192 164
     5 132 200
```

Štai funkcija marginals, sumuojanti gautąsias lenteles pagal eilutes ir stulpelius (pabandykite ją išsiaiškinti!):

```
marginals <- function (x) {
  ## x is a matrix
  row.sums <- apply(x,1,sum)
  row.names <- c(row.names(x),"Total")
  col.names <- c(col.names(x),"Total")
  x <- cbind(x,row.sums)
  col.sums <- apply(x,2,sum)
  x <- rbind(x,col.sums)
  row.names(x) <- row.names
  col.names(x) <- col.names
  x
}
```

Pritaikykime ją mūsų duomenims:

```
> smem <- simple.marginals(em)
```

```

> smem
      0   1 Total
1     23  76   99
2     70 195  265
3    162 258  420
4    192 164  356
5    132 200  332
Total 579 893 1472
> smme <- simple.marginals(me)
> smme
      1   2   3   4   5   Total
0     23  70 162 192 132   579
1     76 195 258 164 200   893
Total 99 265 420 356 332 1472

```

Kokias nors išvadas apie kintamųjų educ ir male priklausomybę sunku daryti, kadangi vyrų yra žymiai daugiau ir tiesiog lyginti skaičius atitinkamuose langeliuose neverta. Pavyzdžiui, smme lentelėje 5-joje educ grupėje moterų yra 132 (iš 579), o vyrų 200 (iš 893; aišku, kad teisingiau būtų lyginti $132/579=0,228$ su $200/893=0.224$).

```

> (smme/smme[, "Total"])[-3, -6] # Matricos smme stulpeliai cikliškai
# (t.y., teisingai) dalinami iš "Total"
# stulpelio; iš gautosios matricos
# pašaliname 3 eilutę ir 6 stulpelį
      1           2           3           4           5
0 0.03972366 0.1208981 0.2797927 0.3316062 0.2279793
1 0.08510638 0.2183651 0.2889138 0.1836506 0.2239642

```

Štai dar du šios procedūros variantai, naudojantys tik base paketo funkcijas:

```

t(apply(em, 2, function(x){x/sum(x)})) # t=transponuoti(matrica)
t(apply(me, 1, function(x){x/sum(x)}))

```

Tiesą sakant, visus šiuos rezultatus galima gauti ir su base paketo prop.table funkcija: išbandykite

```

prop.table(me, 1) # Procentai eilutėse
prop.table(me, 2) # Procentai stulpeliuose
prop.table(me, NULL) # Procentai lentelėje

```

Panašioms reikalams skirta ir funkcija tab.distrib, kurią galima taikyti matricoms arba masyvams (galimas daiktas, gautiems kaip funkcijų table arba xtabs reikšmės):

```

tab.distrib <- function(x){
  dims <- dim(x)
  jtd <- x/sum(x) # N/Total
  col <- matrix(x, nrow=dims[1])
  cold <- sweep(col, 2, apply(col, 2, sum), "/") # N/ColTotal
  cold <- array(cold, dim=dims)
  row <- matrix(t(col), nrow=dims[2])
  rowd <- sweep(row, 2, apply(row, 2, sum), "/") # N/RowTotal
  rowd <- t(matrix(rowd, nrow=nrow(col), ncol=ncol(col)))
  rowd <- array(rowd, dim=dims)
  attributes(cold) <- attributes(x)
  attributes(rowd) <- attributes(x)
  list(joint.distrib=jtd, col.distrib=cold, row.distrib=rowd)
}

```

Du pavyzdžiai:

5.1 pvz.

```
> tab.distrib(me)$row
educ
male      1      2      3      4      5
0 0.03972366 0.1208981 0.2797927 0.3316062 0.2279793
1 0.08510638 0.2183651 0.2889138 0.1836506 0.2239642
```

5.2 pvz.

```
> data(warpbreaks)
> tab <- xtabs(breaks ~ wool + tension, data = warpbreaks)
> tab
tension
wool   L   M   H
A 401 216 221
B 254 259 169
> tab.distrib(tab)
$joint.distrib
tension
wool   L           M           H
A 0.2638158 0.1421053 0.1453947
B 0.1671053 0.1703947 0.1111842

$col.distrib
tension
wool   L           M           H
A 0.6122137 0.4547368 0.5666667
B 0.3877863 0.5452632 0.4333333

$row.distrib
tension
wool   L           M           H
A 0.4785203 0.2577566 0.2637232
B 0.3724340 0.3797654 0.2478006
```

5.1 UŽDUOTIS. Išsiaiškinkite funkcijos `tab.distrib` tekstą ir 5.2 pavyzdį.

5.2 UŽDUOTIS.

Eilutės

```
sex <- c("Male", "Female")
age <- letters[1:6]
education <- c("low", "med", "high")
data <- expand.grid(sex=sex, age=age, education=education)
counts <- rpois(36, 100)
data1 <- cbind(data, counts)
t1 <- xtabs(counts ~ sex + age + education, data=data1)
```

generuoja “labai tvarkingą” duomenų sistemą `data`. Pakeiskite generavimo procedūrą taip, kad `data` būtų labiau panašus į realų (“netvarkingą”) 36 eilučių duomenų rinkinį. Papildykite jį “vardų” stulpeliu `ID` (“vardai” turėtų būti pavidalo KL245, AV311 ir pan.). *Nuoroda.* Jums gali praversti tokios komandos:

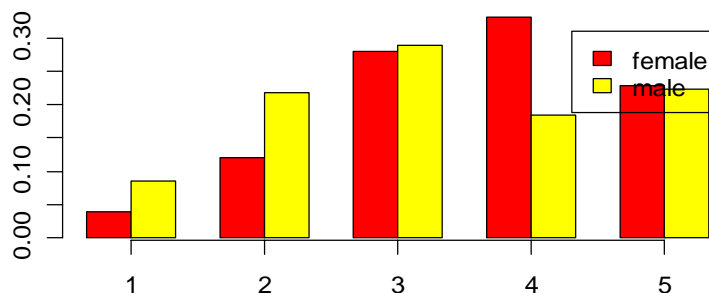
```
i1 <- sample(LETTERS, 36, replace=T)
i2 <- sample(LETTERS, 36, replace=T)
i3 <- sample(100:999, 36)
ID <- paste(i1, i2, i3, sep=" ")
```

Grįžkime prie išsilavinimo tyrimo. Matėme, kad aukščiausiose išsilavinimo grupėse (ypač 4-joje) dominuoja moterys, o žemiausiose – vyrai. Taigi atrodo, kad lytis turi įta-

kos išsilavinimui, tačiau tai patvirtinti statistikos testų skaitmeniniais rezultatais galėsime tik 10 skyriuje (žr. 10.1 užd.).

Tik ką nustatytas faktas bus ypač vaizdus, jei jį pavaizduosime grafiškai.

```
> barplot((smme/smme[, "Total"])[-3, -6], beside=T, legend=c("female",
  "male"))
```



5.1 pav. Moterų ir vyrų išsilavinimo stulpelinės diagramos

Nors kintamieji `educ` ir `male` yra skirti vardiniams kintamiesiems (faktoriams) koduoti, tačiau jie apiforminti kaip skaitiniai kintamieji.

```
> class(educ)
NULL
```

Kartais kintamojo klasė nėra svarbi, tačiau kai kurios funkcijos reikalauja, kad kintamasis būtų, tarkime, faktorius. Pakeisti klasę nėra sunku:

```
> r <- rpois(10,3)
> r
[1] 3 3 4 4 1 2 2 2 2 4
> class(r)
[1] NULL
> rf <- factor(r) # Keičiame klasės požymį
> rf
[1] 3 3 4 4 1 2 2 2 2 4
Levels: 1 2 3 4
> class(rf)
[1] "factor"
> rr <- as.numeric(rf) # Grįžtame atgal
> rr
[1] 3 3 4 4 1 2 2 2 2 4
> class(rr)
NULL
```

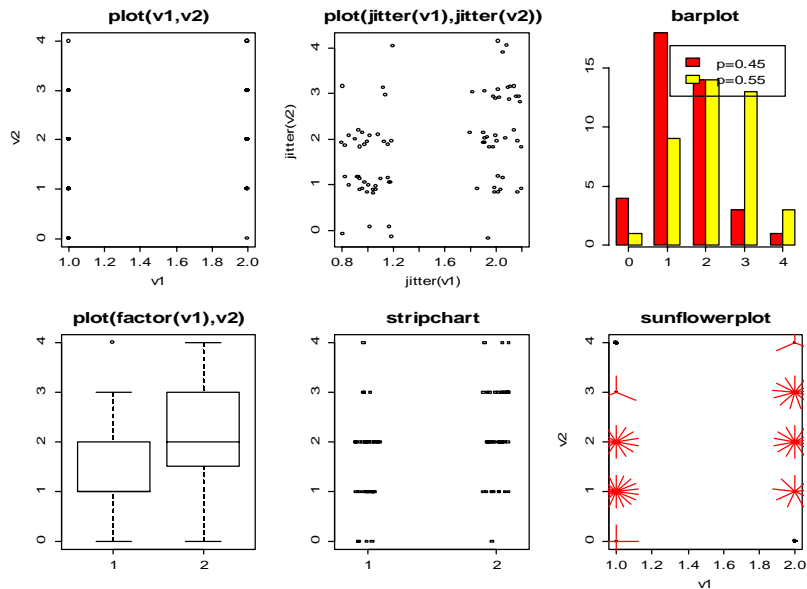
Žemiau pateikiame funkciją `jitt`, kuri, kai skirtingų reikšmių “mažai”, išbrėžia kelis grafikų variantus. Ėminį `r1` sudaro n atsitiktinių skaičių, kurių kiekvienas žymi sėkmių skaičių, atlikus keturis bandymus (sėkmės tikimybė lygi 0,45), o ėminį `r2` sudaro n panašių skaičių (bet sėkmės tikimybė dabar kiek didesnė – 0,55). Grafikas `plot(v1, v2)` šį kartą mažai informatyvus, nes daug taškų (ir neaišku kiek) “sulimpa” į vieną. Tokiu atveju naudinga funkcija `jitter` (jitter (angl.) \approx triukšmas) – ji kiekvieną tašką truputį pastumia į šoną (bet kiekvieną skirtingai), ir dabar aiškiau matyti, keli

taškai buvo sulipę į vieną. Atkreipkite dėmesį į tai, kad priklausomai nuo kintamojo klasės funkcija plot brėžia vis kitokią grafiką – R yra objektiškai orientuota programavimo kalba!

```
jitt <- function(n){
# funkcija jitt
v1 <- rep(1:2,c(n,n))
#n <- 4
#v1 <- rep(1:2,c(n,n))      v1 rodo ėminio numerį
#v1
#[1] 1 1 1 1 2 2 2 2
r1 <- rbinom(n,4,0.45)
r2 <- rbinom(n,4,0.55)
v2 <- c(r1,r2)
opar <- par(mfcol=c(2,3))
on.exit(par(opar))
plot(v1,v2,main="plot(v1,v2)")      # Grafikas visai neinformatyvus
plot(factor(v1),v2,                # Žymiai geriau,sutampa su boxplot
main="plot(factor(v1),v2)")
plot(jitter(v1),jitter(v2),main="plot(jitter(v1),jitter(v2))")
# Visi taškai "padrebinti"
stripchart(v2~v1,method="jitter",vertical=T,main="stripchart")
# Panašus grafikas
barplot(table(v1,v2),beside=T,legend=c("p=0.45","p=0.55"),
main="barplot")                  # Jei p=0,55 - daugiau didelių
# reikšmių
# Spindulių skaičius lygus taško
# kartotinumui
sunflowerplot(v1,v2,
main="sunflowerplot")
cat("Mediana 1=",median(r1),", Mediana 2=",median(r2),"\n")
table(v1,v2)
}
```

```
> jitt(40)
Mediana 1= 1 , Mediana 2= 2
```

```
      v2
v1  0  1  2  3  4
  1  4 18 14  3  1
  2  1  9 14 13  3
```



5.2 pav. Šeši būdai sveikaskaičiams matavimams vaizduoti

5.3 UŽDUOTIS. Išsiaiškinkite žemiau pateiktą tekstą:

```
data(airquality); attach(airquality)
```

```
table(cut(Temp, quantile(Temp)), Month)
#simple two-way contingency table
plot(table(cut(Temp, quantile(Temp)), Month))
```

Pastarasis grafikas iliustruoja tai, kad karščiausia būna liepos mėnesį. Kaip dar galėtumėte pademonstruoti šį faktą?

5.4 UŽDUOTIS. Išsiaiškinkite šį tekstą:

```
data(Titanic)
ftable(Titanic, row.vars = 1:3)
ftable(Titanic, row.vars = 1:2, col.vars = "Survived")
ftable(Titanic, row.vars = 2:1, col.vars = "Survived")
```

5.5 UŽDUOTIS. Išsiaiškinkite šį tekstą:

```
data(UCBAdmissions)
x <- aperm(UCBAdmissions, c(2, 1, 3))
dimnames(x)[[2]] <- c("Yes", "No")
names(dimnames(x)) <- c("Sex", "Admit?", "Department")
ftable(x)

## Fourfold display of data aggregated over departments, with
## frequencies standardized to equate the margins for admission
## and sex.
fourfoldplot(margin.table(x, c(1, 2)))
```

5.2. Mišrus atvejis: vardiniai ir skaitiniai kintamieji

Ar diskriminuojamos Belgijoje moterys? Tiksliau kalbant, ar priklauso atlyginimo wage (tai skaitinis kintamasis) dydis nuo lyties male (tai vardinis kintamasis)? Iš principo, galėtume pasinaudoti tuo, ką jau žinome – kintamąjį wage galėtume skaidyti į grupes, o paskui tirti šių dviejų vardinių kintamųjų sąveikos lentelę. Prisiminkime kaip tai daroma.

```
> attach(bwages)
> summary(wage)
  Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
 88.38 327.30  408.50  445.80 514.60 1919.00
```

Suskaidykime atlyginimą į keturias grupes pagal kiek padailintas kvartilijų reikšmes (tai atlieka funkcija cut):

```
> wage
 [1] 313.85280 194.37800 426.13640 284.09090 318.18180 330.78950
 [7] 331.36360 418.66030 441.91920 290.90910 281.10050 299.87370
.....
> cut(wage, breaks=c(80, 320, 410, 520, 1920))
 [1] (80,320]      (80,320]      (410,520]      (80,320]      (80,320]
 [6] (320,410]     (320,410]     (410,520]     (410,520]     (80,320]
.....
```

Matome, kad pirmasis atlyginimas 313,85 priklauso intervalui (80,320] ir t.t.

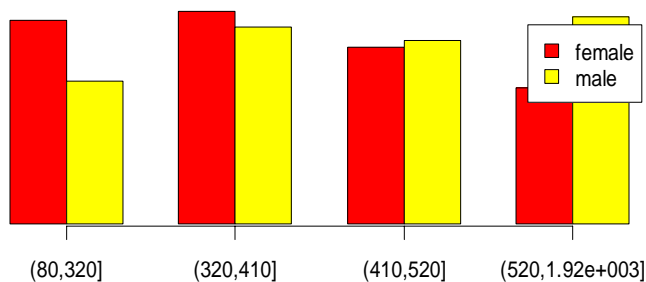
```
> table(cut(wage, breaks=c(80, 320, 410, 520, 1920)))
 (80,320]      (320,410]      (410,520] (520,1.92e+003]
      337              410              364              361
```

Kaip ir reikėjo tikėtis (kodėl?), kiekvienoje grupėje yra maždaug vienodas įrašų skaičius. Dabar patyrinėkime lyties ir atlyginimo sąveikos lentelę.

```
> mw <- table(male,cut(wage,breaks=c(80,320,410,520,1920)))
> mw
male (80,320] (320,410] (410,520] (520,1.92e+003]
  0      162      169      140      108
  1      175      241      224      253
> smmw <- simple.marginals(mw)
> smmw
      (80,320] (320,410] (410,520] (520,1.92e+003] Total
0      162      169      140      108      579
1      175      241      224      253      893
Total   337      410      364      361     1472
```

Iš tikrųjų, mums reikia kiek kitokios lentelės:

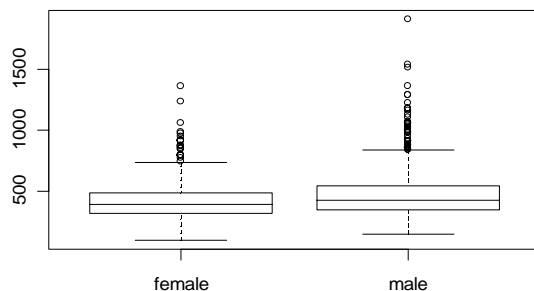
```
> (smmw/smmw[,"Total"])[-3,-5]
      (80,320] (320,410] (410,520] (520,1.92e+003]
0 0.2797927 0.2918826 0.2417962      0.1865285
1 0.1959686 0.2698768 0.2508399      0.2833147
> barplot((smmw/smmw[,"Total"])[-3,-5],beside=T,legend=c("female",
"male"))
```



5.3 pav. Grupuo atlyginimo stulpelinės diagramos (atskirai moterims ir vyrams)

Matome, kad mažų atlyginimų grupėse moterų daugiau, o didelių – mažiau (diskriminacija!). Antra vertus, grupuodami duomenis, kiek sugrabinome tiriamą paveikslą. Nesunku panašų tyrimą atlikti ir su negrupuotais duomenimis.

```
> boxplot(wage[male==0],wage[male==1],names=c("female","male"))
```



5.4 pav. Atlyginimo stačiakampės diagramos (atskirai moterims ir vyrams)

Štai skaitinis šio grafiko ekvivalentas:

```
> tapply(wage,male,summary) # Funkcijos tapply reikšmė yra sąrašas
# su dviem komponentėm, kurių vardai
# "0" ir "1"

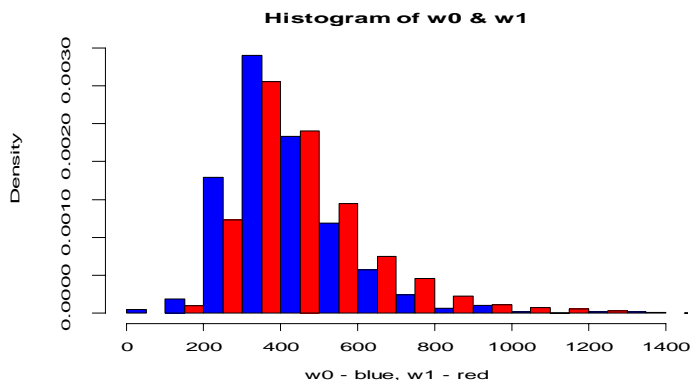
$ "0" # Moterys, t.y. male==0
  Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
 88.38 314.30  383.50  413.90  483.00 1364.00

$ "1" # Vyrų, t.y. male==1
  Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
 139.9  340.9   422.8   466.4   539.8  1919.0
```

Grafike matome, kad abiejose grupėse atlyginimų skirstiniai nesimetriški ir turi daug išskirčių. Taip pat matome, kad vyrų atlyginimų ir mediana ir visi kvantiliai didesni nei moterų (norint išsiaiškinti, ar šis skirtumas nėra dėl ėminio atsitiktinumo, reikia paken-tėti iki 10.2 užduoties).

R yra *open source* produktas, t.y., kiekvienas gali parašyti savo funkcijas ir papildyti jomis esamas bibliotekas. H.Bengtsson'as yra parašęs funkciją `plot.histogram`, leidžiančią išbrėžti dvi ar daugiau histogramas viename brėžinyje. Atsisiųskite šią funkciją iš <http://www.maths.lth.se/matstat/staff/hb/mypackages/R/plot.histogram.R> ir patalpinkite ją bet kur, pvz. C:\spec direktorijoje. RGui aplinkoje surinkite File| Source R code; po lentelėje Select file to source nuvairuokite į C:\spec\plot.histogram.R ir spragtelė-kite ant šio failo: dabar base paketo funkcija `plot.histogram` yra pakeista Bengtsson'o funkcija `plot.histogram` (ji skiriasi nuo senosios tuo, kad yra įvesti du papildomi argumentai – stulpelio plotis `width` ir postūmis `offset`). Brėždama histogramą, funkcija `hist` kreipsis į naująją funkciją `plot.histogram`:

```
attach(bwages)
w0 <- wage[male==0]
w1 <- wage[male==1]
hist(w0, width=0.5, offset=0.0, col="blue", breaks=15,
main="Histogram of w0 & w1", xlab="w0 - blue, w1 - red",prob=T)
hist(w1, width=0.5, offset=0.5, col="red", breaks=15, add=TRUE,prob=T)
```

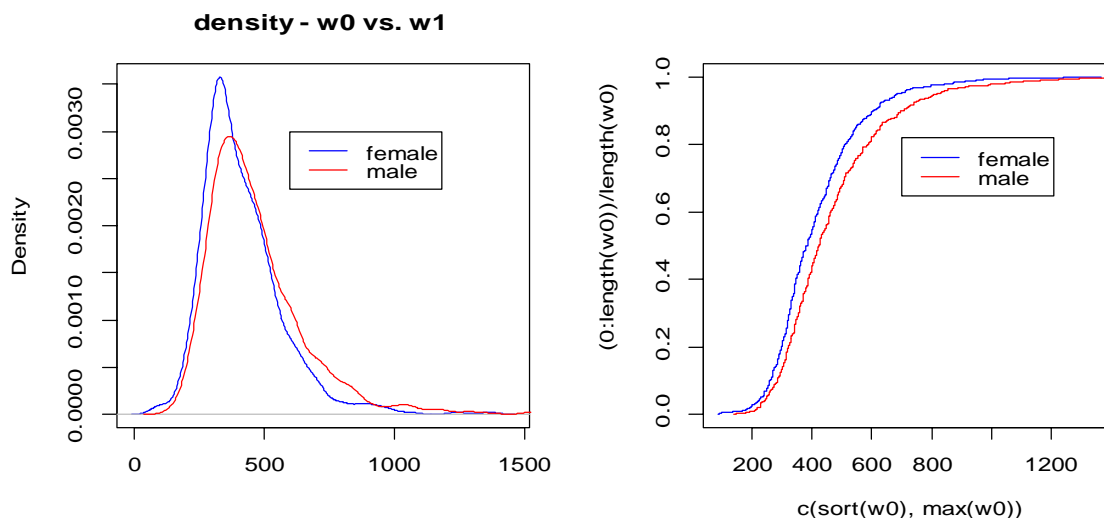


5.5 pav. Moterų ir vyrų atlyginimo histogramos (papildykite šį grafiką legenda)

Vėl matome, kad tarp daug uždirbančių daugumą sudaro vyrai. O štai dar du vyrų ir moterų atlyginimų palyginimo variantai: a) pagrįstas tankio funkcija `density` (žr. 5.6 pav., kairė) ir b) pagrįstas skirtumais tarp empirinių skirstinio funkcijų (žr. 5.6 pav., dešinė)

```
par(mfrow=c(1,2))
plot(density(w0),main="density - w0 vs. w1 ",xlab="",col=4)
lines(density(w1),col=2)
legend(600,.003,c("female","male"),lty=1,col = c(4,2))
plot(c(sort(w0),max(w0)),(0:length(w0))/length(w0),type="S",col=4)
lines(c(sort(w1),max(w1)),(0:length(w1))/length(w1),type="S",col=2)
legend(700,.82,c("female","male"),lty=1,col = c(4,2))
```

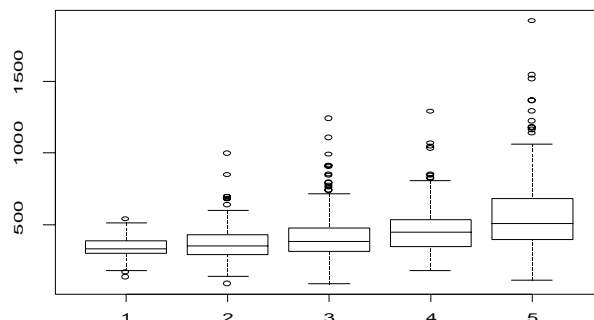
Nagrindėdami empirines skirstinio funkcijas (žr. dešinią 5.6 pav. grafiką) matome, kad raudonoji kreivė yra visur žemiau, kas dar kartą įrodo, kad vyrų atlyginimai yra didesni (kodėl?).



5.6 pav. Vyrų ir moterų atlyginimų empiriniai tankiai (kairėje) ir empirinės skirstinio funkcijos (dešinėje; papildykite šį grafiką vardu ir pakeiskite ašių vardus informatyvesniais)

Dar ištirkime atlyginimo `wage` priklausomybę nuo išsilavinimo `educ`.

```
> boxplot(wage[educ==1], wage[educ==2], wage[educ==3], wage[educ==4],
wage[educ ==5], names=c("1", "2", "3", "4", "5"))
```



5.7 pav. Išsilavinimui didėjant, atlyginimas taip pat didėja

Tekstas `boxplot` komandos skliaustuose yra gana ilgas, tačiau jį galima sutrumpinti:

```
1) boxplot(split(wage,educ),names=c("1","2","3","4","5"))
2) boxplot(wage~educ,names=c("1","2","3","4","5"))
3) with(bwages,boxplot(wage~educ)) # ?with
```

arba tiesiog

```
plot(educ,wage) # plot yra bendrinė funkcija; jei educ yra
# faktorius, plot elgiasi kaip boxplot
```

Užrašas `wage~educ` yra pirmasis formulės pavyzdys (jas dažnai naudosime vėliau regresiniuose modeliuose), o funkciją `split` panagrinėkite patys. Trumpam dar susitokime prie skaitinio paskutiniosios procedūros varianto ir prisiminkime, ką žinome apie sąrašus.

```
> wes <- tapply(wage,educ,summary)1
> wes
$ "1"
  Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
 136.4  300.5   331.4   340.0  388.8   539.8

$ "2"
  Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  88.38 293.30  353.00  371.70  429.60  997.80

$ "3"
  Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  90.91 314.40  384.60  411.60  478.80 1240.00

$ "4"
  Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
 181.8  351.1   448.6   461.1   533.7  1288.0

$ "5"
  Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
 113.6  397.3   507.7   563.2   679.9  1919.0
```

`wes` yra sąrašas:

```
> mode(wes)
[1] "list"
```

Jo pirmąją komponentę pasiekti galime su komanda

```
> wes[["1"]] # Pagal komponentės vardą
  Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
 136.4  300.5   331.4   340.0  388.8   539.8
```

¹ Jei dar norėtume apskaičiuoti panašias charakteristikas sutrumpintam rinkiniui `Bwages` (žr. 6-3 psl.), tai reiktų elgtis taip:

```
detach(bwages)
attach(Bwages)
tapply(wage,educ,summary)
```

Beje, tai galima pastebimai sutrumpinti:

```
with(Bwages, {tapply(wage, educ, summary)})
```

arba

```
> wes$"1" # Pagal komponentės vardą ($ yra [...] sinonimas)
  Min. 1st Qu. Median Mean 3rd Qu. Max.
 136.4 300.5 331.4 340.0 388.8 539.8
```

arba

```
> wes[[1]] # Pagal komponentės numerį
  Min. 1st Qu. Median Mean 3rd Qu. Max.
 136.4 300.5 331.4 340.0 388.8 539.8
```

Komponentė yra skaitinis vektorius su vardais:

```
> mode(wes$"1")
[1] "numeric"
> names(wes$"1")
[1] "Min." "1st Qu." "Median" "Mean" "3rd Qu." "Max."
```

Antra vertus,

```
> wes[1]
$"1"
  Min. 1st Qu. Median Mean 3rd Qu. Max.
 136.4 300.5 331.4 340.0 388.8 539.8
```

yra sąrašas:

```
> mode(wes[1])
[1] "list"
> names(wes[1])
[1] "1"
```

Neužmirškime – R yra programavimo kalba!

5.6 UŽDUOTIS. Ištyrinkite ir palyginkite `davis` duomenų rinkinyje moterų ir vyrų (tikrąjį) svorį `weight` (po to – ūgį `height`). Koks ryšys tarp tikrojo ir praneštojo (reported) ūgio (po to – svorio). Autoriaus hipotezė: ūgį praneša didesnę negu tikrasis, o svorį – mažesnę.

5.3. Skaitiniai kintamieji

Vienas dažnai sutinkamų uždavinių yra dviejų skaitinių kintamųjų, prediktoriaus (arba regresoriaus, arba nepriklausomo kintamojo) x ir (modelio) atsako (arba priklausomo kintamojo) y , sąveikos tyrimas. Tokie modeliai vadinami regresiniais, o dažniausiai nagrinėjami keli klausimai: 1) kaip, žinant x reikšmę, prognozuoti y reikšmę? 2) kaip palyginti du (ar kelis) modelius (Jei turime du Belgiją aprašančius modelius, kuris iš jų “teisingesnis”? Jei Belgijos ir Lietuvos modeliai nedaug skiriasi, tai gal tik dėl ėminių atsitiktinumų?) ir t.t.

Tirkime du skaitinius kintamuosius `weight` ir `height` iš duomenų rinkinio `davis`.

```
attach(davis)
```

```

hF <- height[sex=="F"]
hM <- height[sex=="M"]
wF <- weight[sex=="F"]
wM <- weight[sex=="M"]
par(mfrow=c(1,2))
plot(hF,wF)
plot(hM,wM)

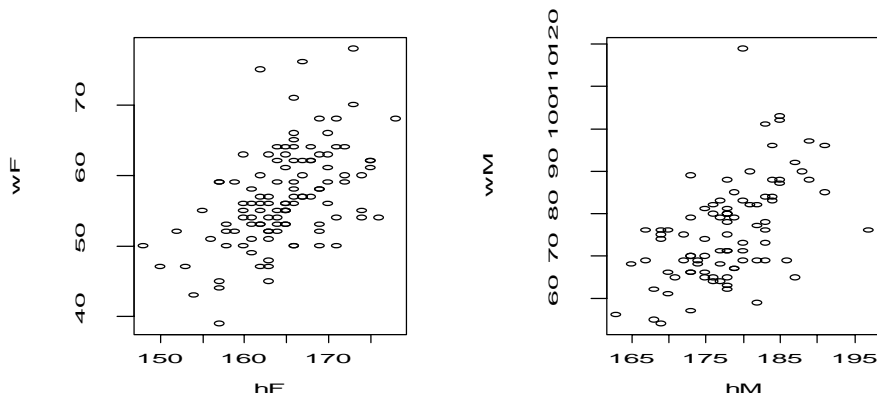
```

Variantas:

```

par(mfrow=c(1,2))
plot(weight~height, data = davis, subset=sex=="F",main="Females")
plot(weight~height, data = davis, subset=sex=="M",main="Males")

```



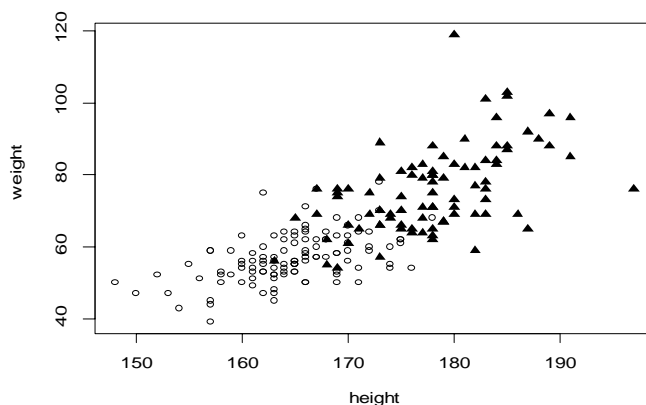
5.8 pav. Šiuose paveiksluose x yra ūgis (prediktorius),
o y – svoris (atsakas) (kairėje - moterys, dešinėje - vyrai)

Atkreipsime dėmesį, kad x ir y ašys abiejuose grafikuose skiriasi – vyrai apskritai yra aukštesni ir sunkesni. Tai bus lengviau pastebėti, jei abu grafikus išbrėšime viename paveiksle.

```

plot(hF,wF,xlim=range(hF,hM),ylim=range(wF,wM), # ?range
xlab="height", ylab="weight")
points(hM, wM, pch=17) # plg. 6.9 ir 6.10 pav.

```



5.9 pav. Moterų (apskritimai) ir vyrų (trikampiai) svorio ir ūgio sklaidos diagrama (papildykite šį grafiką legenda; vyrus ir moteris pažymėkite skirtingomis spalvomis)

Matome, kad didėjant ūgiui svoris apskritai didėja. Kaip šią tendeniją, kurią akis lengvai pagauna, išreikšti matematiškai? Kadangi svoris nuo ūgio priklauso tiesiškai, būtų

protinga per abiejų 5.8 pav. grafikų taškų debesėlių “vidurį” išbrėžti tiesę, kuri viena ar kita prasme būtų “arčiausiai” visų taškų. Vienas iš galimų tiesės parinkimo principų yra mažiausių kvadratų metodas. Tarkime, kad kiekvienas atsako matavimo rezultatas y_i nuo prediktoriaus x_i priklauso taip (tai vadinamasis tiesinės regresijos modelis):

$$y_i = a + bx_i + e_i, \quad i = 1, 2, \dots, n;$$

kitais žodžiais, y ir x priklausomybė tiesinė, bet ją kiek gadina atsitiktinės paklaidos e_i . Tiesės koeficientus a ir b tikslinga parinkti taip, kad paklaidų kvadratų suma (dažnai žymima $RSS(a, b)$ (nuo Residual Sum of Squares)) būtų minimali (tai vadinamasis mažiausių kvadratų metodas):

$$RSS = \min_{a,b} RSS(a, b) = \min_{a,b} \sum_{i=1}^n e_i^2 = \min_{a,b} \sum_{i=1}^n (y_i - (a + bx_i))^2$$

Norint rasti šį minimumą, dalines $RSS(a, b)$ išvestines pagal a ir b reiktų prilyginti 0 ir išspręsti gautąją dviejų tiesinių lygčių sistemą. R pakete šią procedūrą atlieka funkcija `lm` (`lm`=Linear Model (angl. tiesinis modelis)), joje modelį nusakanti formulė $y_i = a + bx_i + e_i$ užrašoma taip: `y~x` (galima rašyti ir `y~1+x`, bet nėra reikalo: laisvasis narys įtraukiamas automatiškai). Funkcija `lm` randa a ir b įverčius \hat{a} ir \hat{b} , ji taip pat apskaičiuoja prognozuojamas y reikšmes $\hat{y}_i = \hat{a} + \hat{b}x_i, i = 1, \dots, n$ ir dar daug ką.

Patikslinkime savo ankstesnę procedūrą:

```
> par(mfrow=c(1,2))
> Fhw1 <- lm(wF~hF) # Sudarome tiesinį modelį moterims
> Fhw1

Call:
lm(formula = wF ~ hF)

Coefficients:
(Intercept)          hF
   -45.6730         0.6227 # Tiesės koeficientų įverčiai  $\hat{a}$  ir  $\hat{b}$ 
# Kitaip sakant, moterų svoris nuo ūgio
# priklauso taip:  $weight = -45,67 + 0,623 \cdot height$ 

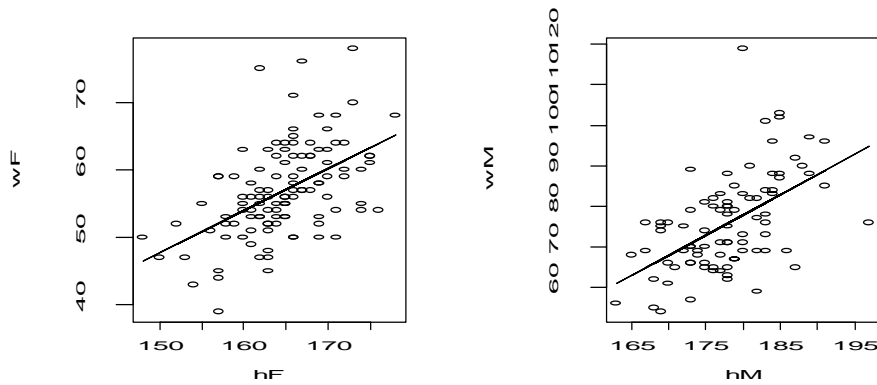
> plot(hF, wF)
> lines(hF, Fhw1$fit) # Fhw1$fit yra vektorius  $(\hat{y}_1, \dots, \hat{y}_n)$ 

> Mhw1 <- lm(wM~hM) # Sudarome tiesinį modelį vyrams
> Mhw1

Call:
lm(formula = wM ~ hM)

Coefficients:
(Intercept)          hM
  -101.3301         0.9956 # Vyrų regresijos tiesės lygtis yra
#  $weight = -101,33 + 0,996 \cdot height$ 

> plot(hM, wM)
> lines(hM, Mhw1$fit) # Sinonimas: abline(Mhw1) - bendrinė
# funkcija abline pati pasiima reikalingus
# koeficientus iš sąrašo Mhw1
```



5.10 pav. Sklaidos diagramos kartu su regresijos tiesėmis

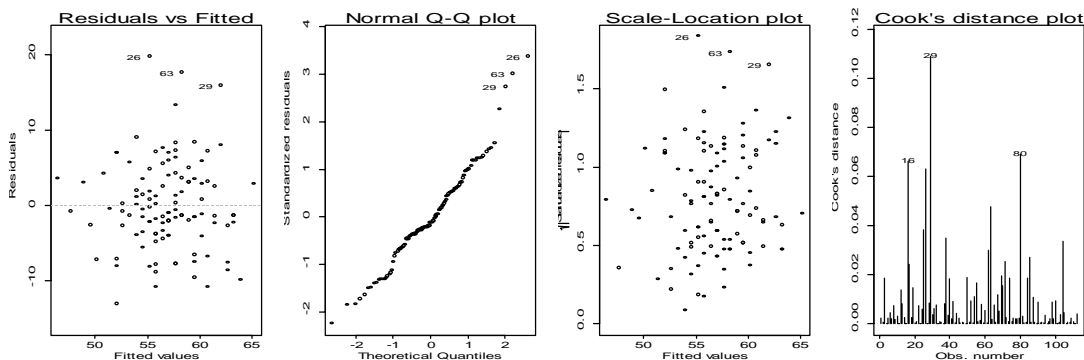
Remdamiesi šiuo tyrimu, galėtume pasiūlyti formules “optimaliam” svoriui apskaičiuoti: tarkime, 170 cm ūgio moteris turėtų sverti $170 \cdot 0.623 - 45.67 = 60.24$ kg. Vyrų svorio formulę trumpai galėtume suformuluoti taip: svoris lygus ūgiui – 100, kitaip sakant 170 cm ūgio vyras turėtų sverti 70 kg.

Mažiausių kvadratų metodas turi daug gerų savybių, tačiau norint, kad jos galiotų, reikia tam tikrų sąlygų. Jos formuluojamos paklaidų terminais:

- paklaidų dispersijos turi nepriklausyti nuo i : $\sigma_i^2 \equiv \sigma^2$
- paklaidos privalo turėti normalųjį skirstinį

Norėdami patikrinti šiuos du teiginius, surinkime

```
par(mfrow=c(1,4))
plot(Fhw1)
```

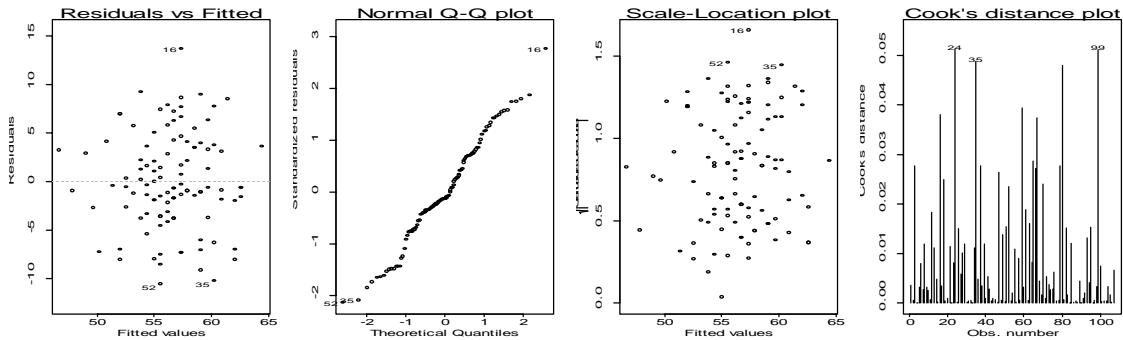


5.11 pav. Tiesinio regresinio modelio Fhw1 diagnostiniai grafikai

Kiekviename grafike nurodyta po tris taškus, kurie, vadovaujantis vienu ar kitu principu (žr. [Fo, 268-277 p.], [ČM1, 47 p.], [Fa, 65-73 p.] arba Grubbs'o kriterijų <http://www.itl.nist.gov/div898/handbook/> (žr. taip pat 9.4 užduotį)) turėtų būti pripažinti iš-skirtimis. Tai įrašai 16, 26, 29, 63 ir 80. Pašalinkime juos iš hF ir wF:

```
hFn <- hF[-c(16,26,29,63,80)]
wFn <- wF[-c(16,26,29,63,80)]
hFn <- hF[-c(16,26,29,63,80)]
```

```
wFn <- wF[-c(16,26,29,63,80)]
Fhwln <- lm(wFn~hFn)
plot(Fhwln)
```



5.12 pav. Tiesinio regresinio modelio Fhwln diagnostiniai grafikai

Dabar turime tik vieną ryškesnę išskirtį – tai įrašas 16 (pagal naują numeraciją) – tačiau tai priimtina (pabandykite ją pašalinti savarankiškai). Pirmas iš kairės grafikas rodo, kad visos modelio paklaidos e_i daugmaž homogeniškai telpa juostoje nuo -10 iki $+10$, taigi visos dispersijos σ_i beveik lygios. Antras iš kairės grafikas rodo, kad paklaidos beveik normalios (galite dar patikrinti su su `eda.shape(Fhwln$res)`). Kadangi

```
> Fhwln$coef
(Intercept)      hFn
-40.3916370    0.5888778
```

tai pagal naują formulę 170 cm ūgio moteris turėtų sverti truputį mažiau: $170 \cdot 0,589 - 40,39 = 59,74$ kg.

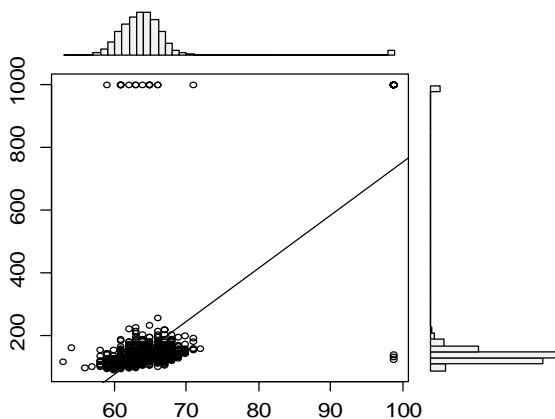
Išpūdingas tiesinės regresijos pavyzdys yra pateiktas [Ve, p.28]. Ten taip pat aptartos atspariosios regresijos funkcijos `r1m` ir `lqs` (jos mažiau jautrios išskirtims).

5.7 UŽDUOTIS. Pati užduotis – kiek vėliau, o dabar pakartosime ūgio ir svorio analizę su kitais duomenimis. Kompaktiniame diske R1 (Žr. Data\StatLab\Data) yra gana didelis (1236 įrašai) duomenų rinkinys `babies_data.htm`, kuriame be kitų kintamųjų yra pateikti duomenys apie moterų ūgį ir svorį. Nukopijuokime šiuos duomenis į R darbinę direktoriją ir nusiskaitykime su

```
> StatLabs <- read.table("babies_data.txt",header=T)
> StatLabs
  bwt gestation parity age height weight smoke
1  120         284     0  27    62   100     0
2  113         282     0  33    64   135     0
3  128         279     0  28    64   115     1
4  123         999     0  36    69   190     0
5  108         282     0  23    67   125     1
6  136         286     0  25    62    93     0
7  138         244     0  33    62   178     0
8  132         245     0  23    65   140     0
9  120         289     0  25    62   125     0
10 143         299     0  30    66   136     1
```

Pakete Simple yra funkcija `simple.scatterplot` – prisijunkime ją su `library(Simple)`

```
attach(StatLabs)
simple.scatterplot(height,weight) # Brėžia sklaidos diagramą ir abiejų
# kintamųjų histogramas
```



5.13 pav. x ašis – height, y ašis - weight

Aiškiai matome abiejų dydžių išskirtis. Pabandykime apžiūrėti StatLabs didžiausių reikšmių srityje. Bet pirmiau keli žodžiai

Apie rūšiavimą. Ilgų skaitinių vektorių perrašymas didėjimo tvarka yra gana daug laiko reikalaujanti procedūra (žr. `?sort`). Štai keli paprastesni pavyzdžiai.

```
> set.seed(5)
> x1 <- rbinom(10,20,.4)
> x1
[1] 7 9 6 10 8 5 7 11 4 13
```

Perrašyti šį vektorių didėjimo tvarka galima su `sort`:

```
> sort(x1)
[1] 4 5 6 7 7 8 9 10 11 13
```

arba su `order` ir `[...]` kombinacija:

```
> order(x1)
[1] 9 6 3 1 7 5 2 4 8 10
```

(mažiausias `x1` elementas yra 9-asis, po eina 6-asis ir t.t.)

```
> x1[order(x1)]
[1] 4 5 6 7 7 8 9 10 11 13
```

Kartais matricą ar duomenų sistemą reikia surūšiuoti, tarkime, pirmojo stulpelio didėjimo tvarka.

```
> x2 <- rbinom(10,20,.4)
> x3 <- rbinom(10,20,.4)
> xx <- data.frame(x1,x2,x3)
> xx[order(x1),]
```



```

      x1 x2 x3
9     4 14  6
6     5 11  9
3     6  6  7
1     7 11  5
7     7  8  8
5     8  7  3
2     9  6  8
4    10  7 12
8    11 10  8
10   13  6  9

```

Šios matricos pirmajame stulpelyje yra du 7-tukai, tačiau kodėl pirmiau eina eilutė 7 11 5, o ne 7 8 8 – sunku pasakyti. Reikalui esant, išrišti lygius elementus galima, pavyzdžiui, pagal antrojo stulpelio reikšmes:

```

> xx[order(x1,x2),]
      x1 x2 x3
9     4 14  6
6     5 11  9
3     6  6  7
7     7  8  8
1     7 11  5
5     8  7  3
2     9  6  8
4    10  7 12
8    11 10  8
10   13  6  9

```

Grįžkime prie StatLabs duomenų rinkinio.

```

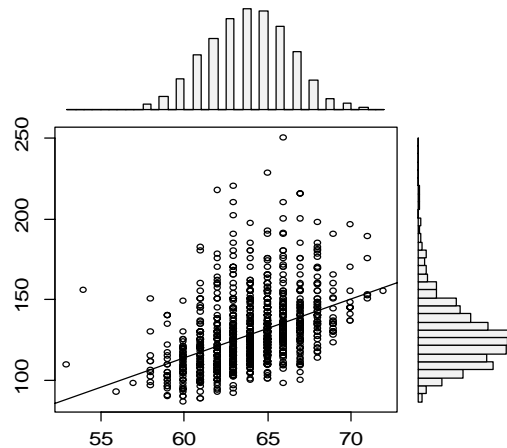
> oo <- order(height,weight) # Jei a yra vektorius, tai a[order(a)] yra
                              # a, perrašytas didėjimo tvarka
> SL <- StatLabs[oo,]         # SL yra StatLabs, perrašytas height didė-
                              # jimo tvarka; kai height vienodi, įrašus
                              # išdėsto weight didėjimo tvarka
> SL[1210:1236,]
      bwt gestation parity age height weight smoke
718  144         287     1  33    71    153     1
148  160         300     0  29    71    175     1
288  148         279     0  27    71    189     0
86   129         274     0  29    71    999     1
176  122         278     0  31    72    155     1
1216 152         267     0  28    99    119     1
848  134         278     0  28    99    126     1
361  107         278     1  27    99    135     0
43   138         302     0  26    99    999     1
153  127         336     0  29    99    999     0
155  129         999     0  23    99    999     1
186  131         286     0  34    99    999     1
231  111         284     0  22    99    999     1
338  131         283     0  31    99    999     0

```

Atrodo, kad skaičiai 99 ir 999 skirti kažkokiam kodavimui. Iš tikrųjų, www.stat.berkeley.edu/users/statlabs/labs.html skyrelyje Birth weight II randame, kad jie žymi tą faktą, kad šios moters ūgis ar, atitinkamai, svoris nežinomas. Pašalinkime šios įrašus. Tam reiktų prisiminti, kad `height==99` (skaitome: `height` lygus 99) žymi loginį vektorių, kurio ilgis 1236 ir kurio beveik visos komponentės lygios FALSE (išskyrus atitinkančias `height==99` – šios lygios TRUE). `height!=99` (skaitome: `height` nelygus 99) yra loginis vektorius, kuriame, lyginant su ankstesniu, TRUE ir

FALSE sukeista vietomis. Loginiame vektoriuje `height!=99&weight!=999` reikšmė TRUE yra ten, kur `height` nelygus 99 ir kartu `weight` nelygus 999 (atkreipiame dėmesį: loginis “arba” žymimas simboliu `|`).

```
> SL <- StatLabs[height!=99&weight!=999,] # Paliekame tik eilutes,
# atitinkančias TRUE
> simple.scatterplot(SL$height,SL$weight)
```



5.14 pav. Ūgio ir svorio sklaidos diagrama su abiejų kintamųjų histogramomis

Savarankiškai įsitikinkite (pvz., su `eda.shape`), kad x koordinatė (t.y., `height`) turi beveik normalųjį skirstinį, tuo tarpu `weight` – tikrai ne. Pastarąjį faktą galima paaiškinti bent dviem priežastimis: 1) nesveiku gyvenimo būdu ir 2) svoris proporcingas tūriui, o pastarasis – ūgio kubui (normaliojo a.d. kubas nėra normalusis).

Dabar pati UŽDUOTIS. Atlikite panašią analizę su vyrų ūgiu ir svoriu. Duomenis ir jų aprašymą rasite R1 disko failuose `babies23_data.htm` ir `All_BABIES.doc`.

5.8 UŽDUOTIS. Kodėl šie du grafikai skiriasi?

```
par(mfrow=c(1,2))
x <- runif(50,-15,20)
y <- -x^2+rnorm(50,sd=2.5)
plot(x,y,t="l")
oo <- order(x)
plot(x[oo],y[oo],type="l")
```

5.9 UŽDUOTIS. `df` yra duomenų sistema, turinti 4 stulpelius: `df <- data.frame(x1,x2,x3,x4)`. Perrašykime šią sistemą `x2` didėjimo tvarka. Štai du variantai.

- 1) `df[sort.list(df$x2),]`
- 2) `df.di <- df[order(df$x2),]` # Rūšiuojame didėjimo tvarka
`df.ma <- df[rev(order(df$x2)),]` # Rūšiuojame mažėjimo tvarka

Susiraskite ir išrūšiuokite realų pavyzdį iš `data()`.

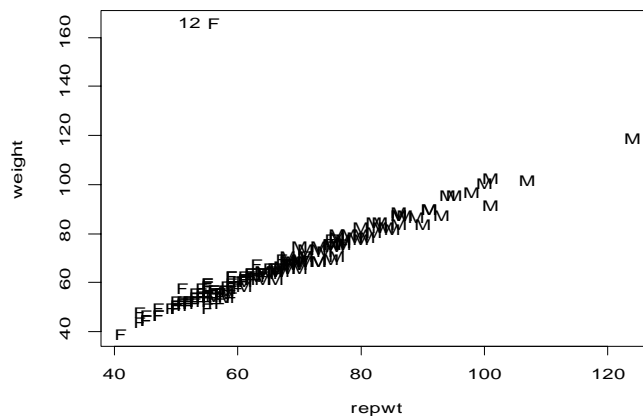
5.10 UŽDUOTIS. Kaip matėme 5.1 paveiksle, tinkamai parinkti legendos vietą ir formą ne visuomet paprasta. Žemiau pateiktas pavyzdys pateikia vieną šios problemos sprendimų (žr. `?par` – ši funkcija turi daugybę opcijų).

```
## increase size of the bottom-border:
par(mar= c(6, 4, 4, 2) + .1)
## Set xpd=TRUE, so all plotting is clipped to the figure
## (not the plot) region:
par(xpd=TRUE)
## Your barplot:
bp <- barplot(1:3)
## Text for the legend:
legend.text <- c("cats", "dogs", "cows")
## And plot the legend below the existing plot -- centering it
legend(mean(range(bp)), -0.3, legend.text, xjust = 0.5,
       fill=heat.colors(length(legend.text)), horiz = TRUE)
```

Pagal šį pavyzdį, pertvarkykite 5.1 paveikslą.

5.11 UŽDUOTIS. Pateikiame dar vieną išskirčių paieškos variantą. Su Copy ir Paste perkeltkite į R žemiau pateiktas komandas (iki `identify` eilutės imtinai), spragtelėkite kairinį pelės klavišą greta išsiskiriančio F taško (pasirodys jo įrašo numeris 12; taškų identifikavimo procedūrą galima tęsti, bet jei ją nusprendėte baigti, paspauskite pelės dešinį klavišą).

```
library(car)
data(Davis)
attach(Davis)
names(Davis)
#[1] "sex"      "weight"   "height"   "repwt"    "repht"
rm(sex,weight,height,repwt,repht) # Dėl viso pikto
plot(repwt,weight,pch=as.character(sex))
identify(repwt,weight)
#warning: no point with 0.25 inches
#[1] 12
```



5.15 pav. Praneštojo ir tikrojo svorio sklaidos diagrama

```
> Davis[11:13,] # Pasidairykime Davis 12 eilutės aplinkoje
  sex weight height repwt repht
11  M     70    175     75    174
12  F    166     57     56    163
13  F     51    161     52    158
```

Atlikite panašią analizę su car paketo Duncan duomenų rinkiniu. Panagrinėkite income ir prestige sklaidos diagramą. Identifikuokite kelis išsiskiriančius taškus su

```
> identify(income,prestige,label=rownames(Duncan))
```

5.12 UŽDUOTIS. Pratęsimе požymių sąveikos lentelės nagrinėjimą. Išsiaiškinkite žemiau pateiktą pavyzdį.

```
data(warpbreaks)
tab <- xtabs(breaks ~ wool + tension, data = warpbreaks)

> tab
      tension
wool  L   M   H
  A 401 216 221
  B 254 259 169
> tab/sum(tab)
      tension
wool  L           M           H
  A 0.2638158 0.1421053 0.1453947
  B 0.1671053 0.1703947 0.1111842
# N/RowTotal
> tab/apply(tab, 1, sum)
      tension
wool  L           M           H
  A 0.4785203 0.2577566 0.2637232
  B 0.3724340 0.3797654 0.2478006
# both N/ColTotal
> sweep(tab, 2, apply(tab, 2, sum), "/")
> tab/(rep(1, 2) %o% apply(tab, 2, sum))
      tension
wool  L           M           H
  A 0.6122137 0.4547368 0.5666667
  B 0.3877863 0.5452632 0.4333333
```

5.13 UŽDUOTIS. Štai warpbreaks duomenų rinkinio analizės tęsinys.

```
data(warpbreaks)
opar <- par(mfrow = c(1,2), oma = c(0, 0, 1.1, 0))
plot(breaks ~ tension, data = warpbreaks, col = "lightgray",
      varwidth = TRUE, subset = wool == "A", main = "Wool A")
plot(breaks ~ tension, data = warpbreaks, col = "lightgray",
      varwidth = TRUE, subset = wool == "B", main = "Wool B")
mtext("warpbreaks data", side = 3, outer = TRUE)
par(opar)
```

Išsiaiškinkite kiekvienos eilutės prasmę. Kokias išvadas galite padaryti iš gautų grafikų?

Štai kelios užduotys iš [Ve].

5.14 UŽDUOTIS. A student evaluation of a teacher is on a 1-5 Leichert scale. Suppose the answers to the first 3 questions are given in this table:

Student	Ques. 1	Ques. 2	Ques. 3
1	3	5	1
2	3	2	3
3	3	5	1
4	4	5	1
5	3	2	1
6	4	2	3

7	3	5	1
8	4	5	1
9	3	4	1
10	4	2	1

Enter in the data for question 1 and 2 using `c()`, `scan()`, `read.table` or `data.entry()`

1. Make a table of the results of question 1 and question 2 separately.
2. Make a contingency table of questions 1 and 2.
3. Make a stacked barplot of questions 2 and 3.
4. Make a side-by-side barplot of all 3 questions.

5.15 UŽDUOTIS. In the library MASS is a dataset `UScereal` which contains information about popular breakfast cereals. Attach the data set as follows:

```
library(MASS)
data(UScereal)
attach(UScereal)
names(UScereal) # to see the names
```

Now, investigate the following relationships, and make comments on what you see. You can use tables, barplots, scatterplots etc. to do your investigation.

1. The relationship between manufacturer and shelf
2. The relationship between fat and vitamins
3. the relationship between fat and shelf
4. the relationship between carbohydrates and sugars
5. the relationship between fibre and manufacturer
6. the relationship between sodium and sugars

Are there other relationships you can predict and investigate?

5.16 UŽDUOTIS. The built-in data set `mammals` contains data on body weight versus brain weight. Use the `cor` to find the Pearson and Spearman correlation coefficients. Are they similar? Plot the data using the `plot` command and see if you expect them to be similar. You should be unsatisfied with this plot. Next, plot the logarithm (`log`) of each variable and see if that makes a difference.

5.17 UŽDUOTIS. For the data set on housing prices, `homedata`, investigate the relationship between old assessed value and new. Use old as the predictor variable. Does the data suggest a linear relationship? Are there any outliers? What may have caused these outliers? What is the predicted new assessed value for a \$75,000 house in 1970.

5.18 UŽDUOTIS. For the `orida` dataset of Bush vs. Buchanan, there is another obvious outlier that indicated Buchanan received fewer votes than expected. If you remove both the outliers, what is the predicted value for the number of votes Buchanan would get in Miami-Dade county based on the number of Bush votes?

5.19 UŽDUOTIS. For the data set `emissions` plot the perCapita GDP (gross domestic product) as a predictor for the response variable CO2 emissions. Identify the outlier and find the regression lines with this point, and without this point.

5.20 UŽDUOTIS. Attach the data set `babies`:

```
library(simple)
data(babies)
attach(babies)
```

This data set contains much information about babies and their mothers for 1236 observations. Find the correlation coefficient (both Pearson and Spearman) between `age` and `weight`. Repeat for the relationship between `height` and `weight`. Make scatter plots of each pair and see if your answer makes sense.

5.21 UŽDUOTIS. Find a dataset that is a candidate for linear regression (you need two numeric variables, one a predictor and one a response.) Make a scatterplot with regression line using R.

5.22 UŽDUOTIS. The built-in data set `mtcars` contains information about cars from a 1974 Motor Trend issue. Load the data set (`data(mtcars)`) and try to answer the following:

1. What are the variable names? (Try `names()`.)
2. what is the maximum `mpg`?
3. Which car has this?
4. What are the first 5 cars listed?
5. What horsepower (`hp`) does the “Valiant” have?
6. What are all the values for the Mercedes 450slc (`Merc 450SLC`)?
7. Make a scatterplot of `cylinders (cyl)` vs. `miles per gallon (mpg)`. Fit a regression line. Is this a good candidate for linear regression?

5.23 UŽDUOTIS. Žemiau yra pateiktas vadinamasis Anskombės (Anscombe) kvartetasis:

x1	y1	x2	y2	x3	y3	x4	y4
10	8.04	10	9.14	10	7.46	8	6.58
8	6.95	8	8.14	8	6.77	8	5.76
13	7.58	13	8.74	13	12.74	8	7.71
9	8.81	9	8.77	9	7.11	8	8.84
11	8.33	11	9.26	11	7.81	8	8.47
14	9.96	14	8.10	14	8.84	8	7.04
6	7.24	6	6.13	6	6.08	8	5.25
4	4.26	4	3.10	4	5.39	8	5.56
12	10.84	12	9.13	12	8.15	8	7.91
7	4.82	7	7.26	7	6.42	8	6.89
5	5.68	5	4.74	5	5.73	19	12.50

Nežiūrint to, kad šios keturios poros pastebimai skiriasi (lengviausia tai pamatyti iš sklaidos diagramų; jas išdėstykite kaip 2×2 matricą), jų vienmatės ir dvimatės skaitinės charakteristikos sutampa. Negana to, sutampa jų regresijos tiesės ir visos tiesinių regresinių modelių `lin.i <- lm(yi~xi)` charakteristikos (pastarąjį faktą galima patikrinti su `summary(lin.i)`). Patikrinkite šiuos teiginius.

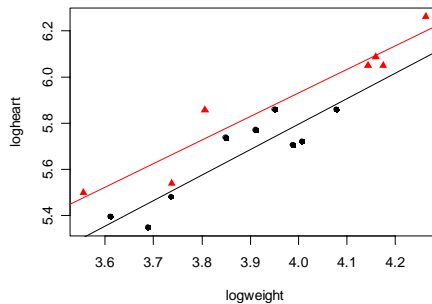
5.24 UŽDUOTIS. R konsolėje spragtelėkite `File|Source R code...`, nuvairuokite į kompaktinio disko R1 failą `Data\Maindonald\dolphins.R` ir spragtelėkite ant `Open`.

```

> source("E:/Data/Maindonald/dolphins.R")
> dolphins
  wt heart logweight logheart species
1  35   245  3.555348  5.501258   styx
2  42   255  3.737670  5.541264   styx
*****
8  54   300  3.988984  5.703782   delph
9  59   350  4.077537  5.857933   delph
*****

```

Pertvarkę (jei reikia) failą dolphins, išbrėžkite kintamųjų logweight (=x) ir logheart (=y) sklaidos diagramą. Skirtingas species grupes pavaizduokite skirtingais simboliais. Kiekvienai grupei išbrėžkite tiesinės regresijos tieses. Papildykite žemiau pateiktą paveikslą legenda ir antrašte.



5.16 pav. 5.23 užduoties iliustracija

Nuoroda. Duomenų sistemą dolphins papildykite stulpeliu spec (naują sistemą pavadinkite dolphins1), sudarytu iš 1 (kai styx) ir 0 (kai delph). Gal būt, jums pravers tokia eilutė:

```

> abline(lm(logheart~logweight,subset(dolphins1,spec==1)),col=2)

```

5.25 UŽDUOTIS. Du skaitinius vektorius galima apjungti kaip į matricą, taip ir į duomenų sistemą:

```

x <- c(NA,1:5)
y <- c(6:10,NA)
xym <- cbind(x,y) # Matricoje eilutės ir stulpeliai lygiateisiai
xyd <- data.frame(x,y) # Duomenų sistema turi "natūralią" stulpelinę
> xym # struktūrą
  x y
[1,] NA 6
[2,] 1 7
[3,] 2 8
[4,] 3 9
[5,] 4 10
[6,] 5 NA
> xyd
  x y
1 NA 6
2 1 7
3 2 8
4 3 9
5 4 10
6 5 NA

```

Norint atrinkti šių objektų poaibius, nusakomus, tarkime, stulpelio y reikšmėmis, lengviau dirbti su duomenų sistemomis ir funkcija subset.

```
> subset(xyd, y>6)
  x y
2 1 7
3 2 8
4 3 9
5 4 10
```

Dažnai reikia pašalinti eilutes, kuriose yra trūstamų duomenų.

```
> subset(xyd, y!="NA")
  x y
1 NA 6
2 1 7
3 2 8
4 3 9
5 4 10
```

Pastarąjį rezultatą gautume ir su `na.omit` funkcija (ji patogesnė, nes pašalintų ir eilutes su blogais x'ais):

```
> na.omit(xyd)
  x y
2 1 7
3 2 8
4 3 9
5 4 10
```

O dabar pati užduotis. MASS bibliotekoje yra duomenų rinkinys `Uscereal`. Aprašykite jį. Palikite jame tik eilutes, kuriose `vitamins` lygūs 100%. Atspausdinkite eilučių vardus tokiu pavidalu.

```
[,1]
[1,] "Just Right Fruit & Nut"
[2,] "Product 19"
[3,] "Total Corn Flakes"
[4,] "Total Raisin Bran"
[5,] "Total Whole Grain"
```

5.26 UŽDUOTIS. Kompaktinio disko R1 direktorijoje `Data\Misc` yra failai `ztemp.txt` ir `zdate.txt`. Faile `ztemp.txt` yra pateikta Žuvinto ežero rajono pirmųjų 152 kiekvienerių metų (nuo 1966 iki 1977) dienų vidutinė paros oro temperatūra. Faile `zdate.txt` yra nurodyta pirmosios kuoduotosios anties pasirodymo Žuvinto ežere tais pačiais metais data (metų dienos numeris; kai kuriais metais duomenų trūksta).

- Išbrėžkite visų metų temperatūrų grafikus
- Ištirkite `zdate` duomenis.
- Apskaičiuokite vidutinę 7 dienų prieš atskrendant šiam paukščiui temperatūrą (gali būti naudinga funkcija `is.na`)
- Išbrėžkite šios temperatūros ir atskridimo datos sklaidos diagramą. Sudarykite atitinkamą tiesinį regresinį modelį. Ar turi temperatūra įtakos atskridimo datai?

5.27 UŽDUOTIS. Bibliotekoje MASS yra duomenų rinkinys `mammals`. Atlikite duomenų priešanalizę, identifikuokite išskirtis. Sudarykite du regresinius modelius 1) `brain~body` ir 2) `log(brain)~log(body)`. Kuris iš jų tinkamesnis?

6. Daugiamačiai duomenys: aprašomoji statistika ir duomenų priešanalizė

Dažnai tiriamas reiškinys yra aprašomas modeliu, kuriame yra daugiau kaip du kintamieji. Paprastai čia susiduriama su keliomis problemomis, kaip antai, patogus duomenų pateikimas, duomenų vizualizacija, tinkamas modelio parinkimas. Kai kurios iš problemų sprendžiamos kiek apibendrinant dvimatį atvejį, tačiau yra ir specifinių daugiamačių aspektų. Pradėkime nuo duomenų pateikimo.

6.1. Duomenų pertvarkos

Daugiamačius duomenis R pakete galima patalpinti įvairiais būdais. Tarkime, kad, analizuojant UAB Geronda darbą, buvo surinkti duomenys apie kiekvieną padalinį (iš viso jų trys). Buvo fiksuojamas kiekvieno darbuotojo amžius (skaitinis kintamasis) ir lytis (v ir m – tai faktorius). Štai dalis surinktų duomenų (patys sukurkite `pa1`¹):

```
> pa1          # pa1 yra duomenų sistema
  am ly
1 25  v
2 35  m
3 29  v
4 46  v
5 33  m

> pa2
  am ly
1 44  v
2 51  v
3 29  m

> pa3
  am ly
1 34  v
2 22  m
3 29  m
4 40  v
```

Apjungti šiuos duomenis į vieną rinkinį `geronda` galima įvairiais būdais.

```
> geronda <- data.frame(pa1,pa2,pa3) # Blogai
Error in data.frame(pa1, pa2, pa3) : arguments imply differing number
of rows: 5, 3, 4
> geronda <- list(pa1,pa2,pa3)      # Dabar gerai, bet dažniausiai
> geronda                          # mums reikia ne tokio pavidalo
```

¹ Pradėkite su

```
> am<-25
> pa1<-data.frame(am)
> pa1<-edit(pa1)
```

```
[[1]]
  am ly
1  1  v
2  4  m
3  2  v
4  5  v
5  3  m
```

```
[[2]]
  am ly
1 44  v
2 51  v
3 29  m
```

```
[[3]]
  am ly
1 34  v
2 22  m
3 29  m
4 40  v
```

Būtų patogiau, jei geronda atrodytų taip:

```
> geronda <- rbind(pa1,pa2,pa3) # Apjungia tris duomenų sistemas
> geronda # (viena po kita)
  am ly
1  25  v
2  35  m
3  29  v
4  46  v
5  33  m
6  44  v
7  51  v
8  29  m
9  34  v
10 22  m
11 29  m
12 40  v
```

Dar geriau būtų, jei geronda turėtų dar vieną stulpelį, kuriame būtų nurodytas padalinio numeris (tai faktorius su reikšmėmis 1, 2 ir 3).

```
> pa <- factor(c(1,1,1,1,1,1,2,2,2,3,3,3,3)) # Surinkome rankomis
```

arba

```
> pa <- factor(rep(1:3,c(dim(pa1)[1], # Jei duomenų daug, geriau
dim(pa2)[1],dim(pa3)[1]))) # šitaip
> pa
[1] 1 1 1 1 1 2 2 2 3 3 3 3
Levels: 1 2 3
```

```
> geronda.p <- data.frame(rbind(pa1,pa2,pa3),pa)
> geronda.p
  am ly pa
1  25  v  1
2  35  m  1
3  29  v  1
4  46  v  1
```

```

5 33 m 1
6 44 v 2
7 51 v 2
8 29 m 2
9 34 v 3
10 22 m 3
11 29 m 3
12 40 v 3

```

Dažnai duomenų sistema pateikiama dar kitokiu pavidalu, įvedant vadinamuosius fiktyvius (= dummy (angl.)) kintamuosius. Pvz., skaitinis kintamasis p2 nurodo, kad šis įrašas priklauso 2-ajam padaliniui, jis apibrėžiamas taip:

```

> p1 <- ifelse(geronda.p[,3]==1,1,0)
> p2 <- ifelse(geronda.p[,3]==2,1,0) # Sukuriame kintamąjį p2 - vek-
> p3 <- ifelse(geronda.p[,3]==3,1,0) # torinės logikos pavyzdys
> geronda.d <- data.frame(geronda.p[, -3], p1, p2, p3)
> geronda.d
  am ly p1 p2 p3
1 25 v 1 0 0
2 35 m 1 0 0
3 29 v 1 0 0
4 46 v 1 0 0
5 33 m 1 0 0
6 44 v 0 1 0
7 51 v 0 1 0
8 29 m 0 1 0
9 34 v 0 0 1
10 22 m 0 0 1
11 29 m 0 0 1
12 40 v 0 0 1

```

Daugiau duomenų pertvarkymo pavyzdžių galima rasti [Ba, Skyriuje Reading and transforming data] arba [My, Exercise 2]). Pateiksime dar vieną pavyzdį. Failas bwages yra didelis ir su juo ne visuomet patogiu dirbti. Imdami tik kas dešimtą bwages eilutę ir išmesdami visus logaritmų stulpelius, sukurkime mažesnę jo variantą Bwages:

```

> Bwages <- data.frame(bwages[seq(1,1472,10),c(1,3,4,7)], row.names =
  as.character(1:148))
> Bwages
      wage educ exper male
1  313.8528   1   23    1
2  281.1005   1   39    0
3  354.6364   1   20    1
.....
146 555.5555   5   33    0
147 396.1039   5    5    0
148 631.3131   5   15    0

```

6.1 UŽDUOTIS. “Ilgėje” duomenų sistemoje df.orig

```

A111 1000
A111 1100
A111 1200
B123 2000
B123 2100
B123 2200

```

kiekvienas “asmuo” (A111, B123 ir t.t.) kartojasi tris kartus. Šią sistemą reikia transformuoti į “plačią” duomenų sistemą `df.fin`

```
A111    1000    1100    1200
B123    2000    2100    2200
```

Štai vienas tokios operacijos variantas:

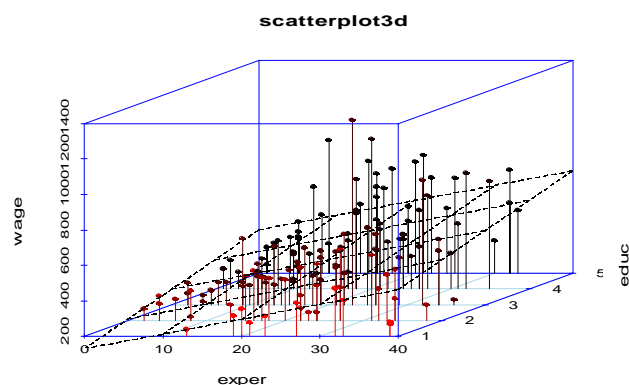
```
df.orig <- read.table("df_orig.txt")
N <- nrow(df.orig)
id <- df.orig[seq(1,N,by=3),1]
dat <- matrix(df.orig[,2], ncol=3, byrow=T)
df.fin <- data.frame(id, dat)
df.fin
  id  X1  X2  X3
1 A111 1000 1100 1200
2 B123 2000 2100 2200
```

Sukurkite savąjį variantą (plg. 6.22 užduotį).

6.2. Grafinė analizė

Akis labai geras matavimo instrumentas, tačiau jau trimatėse sklaidos diagramose ji gana sunkiai įžiūri tendencijas. Iš tikrųjų, panagrinėkime funkciją `scatterplot3d` iš tokio pat pavadinimo paketo (jis yra kompaktiniame diske R1; panašios funkcijos yra Duncan'o Murdoch'o `rgl` pakete, žr. R1 diską arba <http://www.stats.uwo.ca/faculty/murdoch/software/>).

```
attach(Bwages)
Bw3d <- scatterplot3d(exper, educ, wage, # Brėžiame 3-matę sklaidos
highlight.3d=TRUE, col.axis="blue",   # diagramą
col.grid="lightblue",
main="scatterplot3d", pch=20, type="h")
mano.lm <- lm(wage~exper+educ) # Apskaičiuojame regresijos plokštumą
Bw3d$plane3d(mano.lm) # Papildome sklaidos diagramą šia
# regresijos plokštuma
```



6.1 pav. Trimatė sklaidos diagrama ir wage regresijos (`exper` ir `educ` atžvilgiu) plokštuma

Matome, kad jau trimačiu atveju sklaidos diagrama mažai naudinga (o kai matavimų skaičius didesnis, jos apskritai neįmanoma nubrėžti). Tai paaiškina, kodėl daugiamačiu atveju dažniausiai apsiribojama įvairiais pavidalais pateikiama dvimačių ryšių

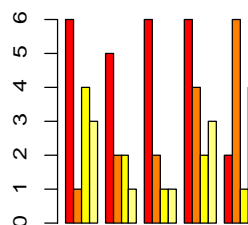
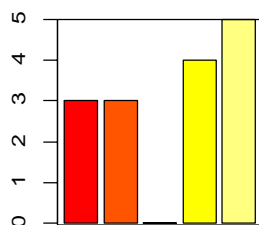
analize. Pradėkime nuo bendro pobūdžio pastabos. R funkcijos `plot`, `boxplot`, `barplot`, `matplot` (ir dar daug kitų) yra bendrinės, t.y., jų reikšmės priklauso nuo argumento tipo. Štai lentelė, kurioje pateikta mums reikalingų faktų santrauka.

6.1 lentelė.

	vektorius	matrica	duomenų sistema
<code>plot</code>	x - koordinatės numeris, y – koordinatės reikšmė	x - pirmasis stulpelis, y - antrasis stulpelis	brėžia daug sklaidos diagramų: x - vienas stulpelis, y - kitas
<code>boxplot</code>	vienas stačiakampis visam vektoriui	vienas stačiakampis visai matricai	vienas stačiakampis kiekvienam stulpeliui (kintamajam)
<code>barplot</code>	po stulpelį kiekvienai koordinatei, jo aukštis lygus koordinatės reikšmei	po stulpelį kiekvienam matricos stulpeliui, eilutės skiriamos spalvomis	neapibrėžta
<code>matplot</code>	x - koordinatės numeris, y - koordinatės reikšmė su žyma	x - eilutės numeris, y - stulpelių reikšmės (kiekvienam stulpeliui sava žyma)	x - eilutės numeris, y - stulpelių reikšmės (kiekvienam stulpeliui sava žyma)

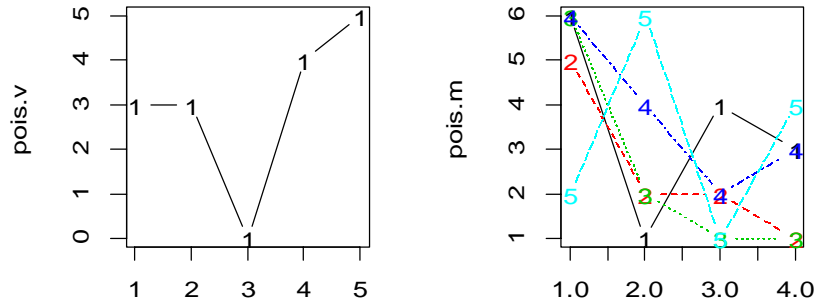
Štai kelios šios lentelės iliustracijos.

```
> par(mfrow=c(1,2))
> pois.v <- rpois(5,3)
> pois.v
[1] 3 3 0 4 5
> barplot(pois.v)
> box() # Diagrama patalpinsime į dėžutę
> pois.m <- matrix(rpois(20,3),ncol=5)
> pois.m
      [,1] [,2] [,3] [,4] [,5]
[1,]    6    5    6    6    2
[2,]    1    2    2    4    6
[3,]    4    2    1    2    1
[4,]    3    1    1    3    4
> barplot(pois.m,beside=T)
```



6.2 pav. Vektoriaus (kairėje) ir matricos (dešinėje) stulpelinės diagramos

```
> matplot(pois.v,type="b") # matplot=matrixplot; yra daug jos
> matplot(pois.m,type="b") # variantų
```



6.3 pav. Vektoriaus grafikas (kairėje) ir matricos grafikas (dešinėje)

Priminsime, kad ir daugelio kitų R funkcijų elgesys priklauso nuo argumento klasės. Žemiau esančioje lentelėje pateikta trumpa šių faktų santrauka.

Funkcija	matrica	duomenų sistema
sum	vienas skaičius	vienas skaičius
max	vienas skaičius	vienas skaičius
median	vienas skaičius	neapibrėžta
mean	vienas skaičius	kiekvienam stulpeliui
sd	kiekvienam stulpeliui	kiekvienam stulpeliui
var	varcov matrica	varcov matrica

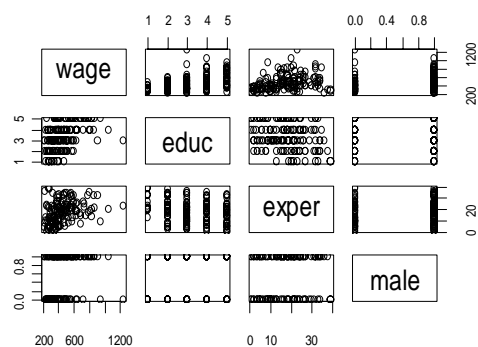
Tenka pripažinti, kad tokia įvairovė yra nepatogi, bet tai kaina, kurią tenka mokėti už programavimo suderinamumą. Priminsime, kad matrica R viduje pateikiama kaip ilgas vektorius su `dim` požymiu, ir, pvz., funkcija `sum` susumuoja visus šio vektoriaus elementus. Norint rasti stulpelių sumas, reiktų naudoti (žr. 3 sk.) vieną iš šių variantų:

```
apply(matrica, 2, sum)
sapply(duomenų sistema, sum)
```

Grįžkime prie duomenų sistemos `Bwages`. Funkcija `plot` turėtų išbrėžti daug sklaidos diagramų, rodančių kiekvieno stulpelio priklausomybę nuo kitų.

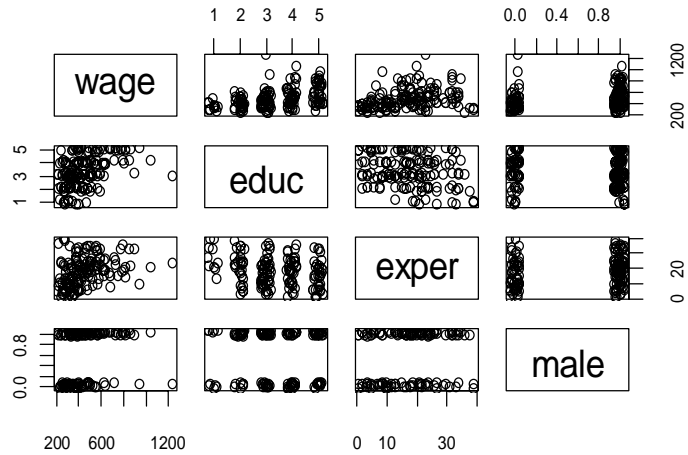
```
plot(Bwages)
```

Deja, šie grafikai mažai informatyvūs, nes daug taškų “sulimpa”. Funkcija `jitter` duomenų sistemoms neapibrėžta, todėl taškus “padrebinkime” patys.



6.4 pav. Visų `Bwages` komponentių porinės sklaidos diagramos

```
Bw.jitt <- Bwages+data.frame(rep(0,148),runif(148,-0.2,0.2),
runif(148,-0.01,0.01),runif(148,-0.05,0.05))
plot(Bw.jitt)
```



6.5 pav. Visų “padrebtų” Bwages komponentų porinės sklaidos diagramos

Mums labiausiai rūpi pirmoji eilutė – matome, kad didėjant išsilavinimui atlyginimas didėja, nuo patyrimo jis priklauso paraboliskai (kaip manote, kodėl?), o vyrų atlyginimas, apskritai, didesnis nei moterų. Beje, didžiausią atlyginimą gauna vidutinio amžiaus ir ne pačio aukščiausio išsilavinimo moteris. Ją nėra sunku rasti.

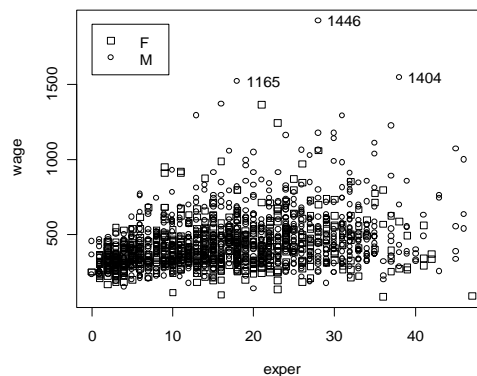
```
> max(wage)
[1] 1239.669
> which(wage==max(wage))
[1] 67
> Bwages[which(wage==max(wage)),]
      wage educ exper male
67 1239.669   3   23   0
```

Kitas variantas pagrįstas identify funkcija.

```
> plot(exper, wage)
> identify(exper, wage, n=1)
[1] 67
```

Surinkę antros eilutės tekstą ir perėję į grafikos langą, pamatysime kryželį (jis žymi kursoriaus padėtį). Nuveskime kryželį ant norimo taško ir spragtelėkime kairiuoju klavišu – pamatysime įrašo eilutės numerį.

6.2 UŽDUOTIS. Originaliajame rinkinyje bwages trys daugiausiai uždirbantys asmenys yra vyrai – patikrinkite (funkcijoje identify imdami n=3, išbrėžkite 6.6 paveikslą). Kokie jų atlyginimai?



6.6 pav. Rinkinyje Bwages daugiausiai uždirbantis asmuo buvo 67-asis, o rinkinyje bwages – 1446-asis, 1165-asis ir 1404-asis

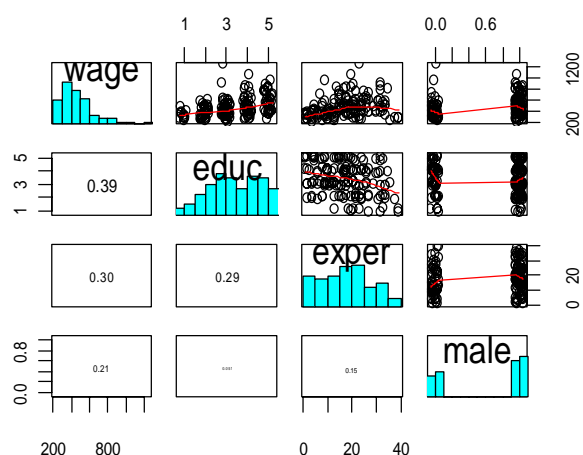
Panagrinėkime funkciją `pairs` – jos išbrėžtas grafikas panašus į funkcijos `plot`, tačiau ji pateikia žymiai daugiau variantų.

```
> pairs(Bw.jitt, upper.panel=panel.smooth, diag.panel=panel.hist,
        lower.panel=panel.cor)
```

Čia `upper.panel` nurodo, kas bus virš įstrižainės (base paketo funkcija `panel.smooth` išbrėž kreivę, einančią per taškų “debesėlio” vidurį – jau turėjome tokios kreivės pavyzdį (regresijos tiesę), bet dabar remiamasi kitokiais principais (kreivę apskaičiuos neparametrinio glodinimo funkcija `lowess`)). Opcija

`diag.panel` nurodo kas dar, be kintamojo vardo, bus ant įstrižainės (funkciją `panel.hist` galima surasti su `?pairs` (žr. pavyzdžių skyrių); prieš vykdant aukščiau užrašytą komandą, funkciją `panel.hist` reikia Pastedu console). Opcija

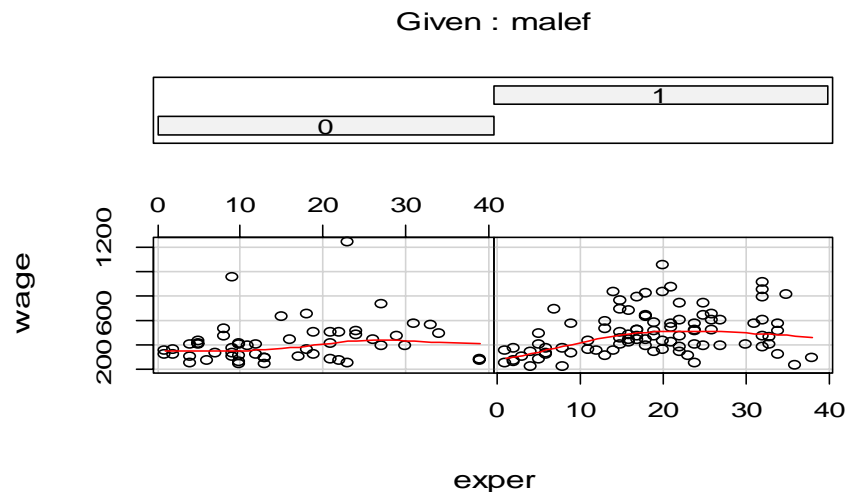
`lower.panel` nurodo, kas bus po įstrižaine (funkciją `panel.cor` irgi rasite funkcijos `pairs` `help`'o pavyzdžių skyriuje, ji apskaičiuoja koreliacijos koeficiento tarp atitinkamų kintamųjų modulį; koeficiento skaitmenų didumas proporcingas koeficiento reikšmei – tai, kad koreliacijos koeficientas tarp `educ` ir `male` praktiškai neįžiūrimas, reiškia, kad jis beveik 0).



6.7 pav. Funkcija `pairs`: duomenų sistemos `Bw.jitt` sklaidos diagramos, glodinančius regresijos kreivės, koreliacijos koeficientai ir histogramos

Kita funkcija, labai naudinga tiriant daugiamatius duomenis, yra `coplot(y~x|a)` arba `coplot(y~x|a*b)`. Ji brėžia keletą y sklaidos diagramų x atžvilgiu (visoms a reikšmėms arba, atitinkamai, visoms porų (a,b) reikšmėms). Norint, kad duomenų sistemos `Bwages` atveju `educ` ir `male` būtų “teisingai” traktuojami, reikėtų pabrėžti, kad jie faktoriai.

```
rm(educ,male) # Dėl viso pikto (kad nebūtų panaudoti panaudoti "pasi-
              # klydę" educ ir male (vietoje educ ir male iš Bwages))
detach(bwages)
attach(Bwages)
educf <- as.factor(educ) # Dabar educ tikrai iš Bwages (o ne gal būt
                        # kažkada atsiradę iš bwages)
malef <- as.factor(male)
Bwf <- data.frame(wage,exper,educf,malef)
rm(educf,malef) # Žiūrėkime į ateitį - "šiukšles" pašalinkime
detach(Bwages)
coplot(wage~exper|malef,
       panel=panel.smooth,data=Bwf) # data=Bwf yra ekvivalentu komandai
                                     # attach(Bwf)viena eilute anksčiau
```

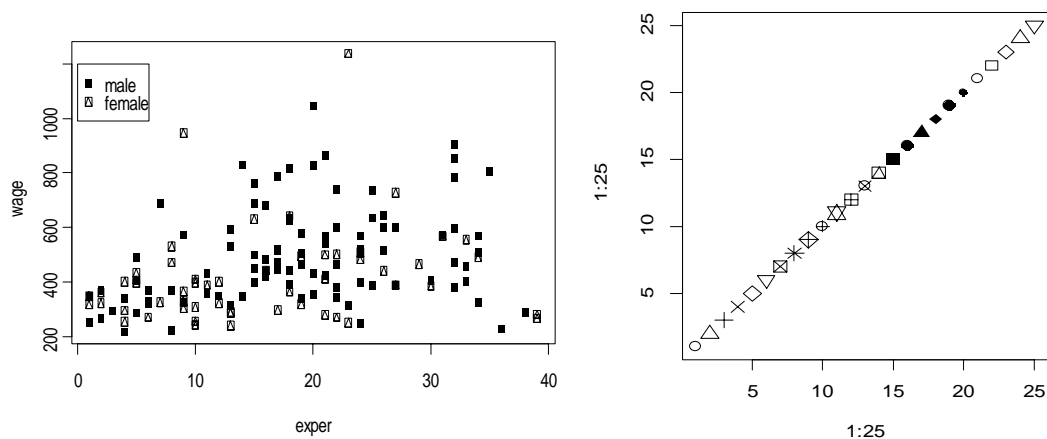
6.8 pav. Atlyginimas priklauso nuo patyrimo, bet vyrams ir moterims skirtingai

Šiuos du grafikus galima pateikti ir vienu paveikslu, bet tuomet vyrus ir moteris reikėtų žymėti skirtingais simboliais (žr. 6.9 pav., kairėje):

```
plot(wage~exper,pch=as.numeric(malef)+13)
legend(0,1200,c("male","female"),pch=as.numeric(malef)+13)
```

Beje, visų 25 R grafinių simbolių lentelę galima atspausdinti su tokiu kodu:

```
plot(1:25,1:25,pch=1:25,cex=1.5) # pch=printing character
# cex=simbolio dydis
```



6.9 pav. Moterų ir vyrų atlyginimo ir patyrimo sklaidos diagramos (kairėje) ir grafinių simbolių lentelė (dešinėje)

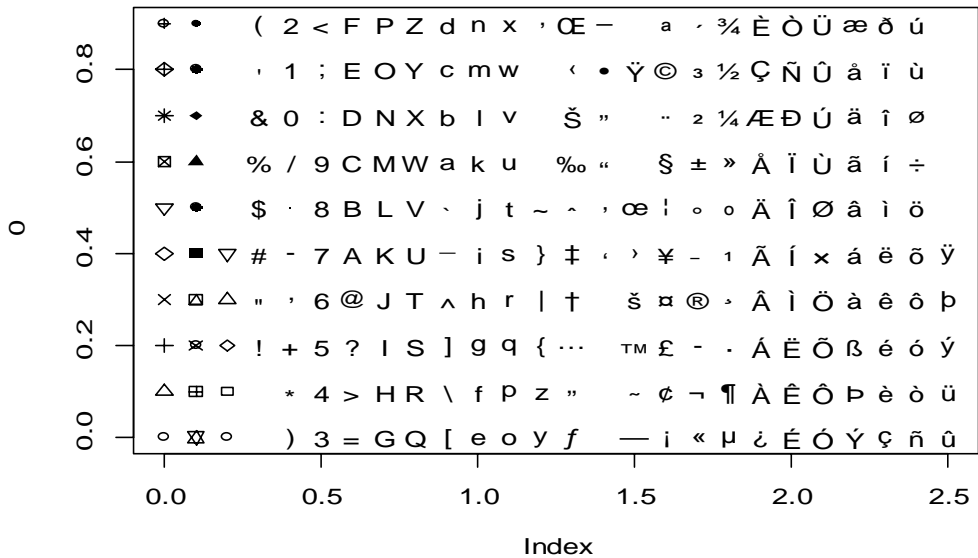
Taigi grafinius R simbolius galima pasirinkti iš 6.9 pav. dešiniajame grafike pateiktų dvidešimt penkių, o apskritai visus `pch` variantus galima išbrėžti su šiuo kodu:

```
plot(x=0,type="n",xlim=c(0,2.5),ylim=c(0,0.9)) # Ką daro ši eilutė?
```

```

for(i in 1:260) {
  j <- i - 1
  x <- j%%10/10 # Surinkite ?"/%"
  y <- j%%10/10
  cat("(",x,",",y,",",i,")\n")
  points(x,y,pch=i)
}

```



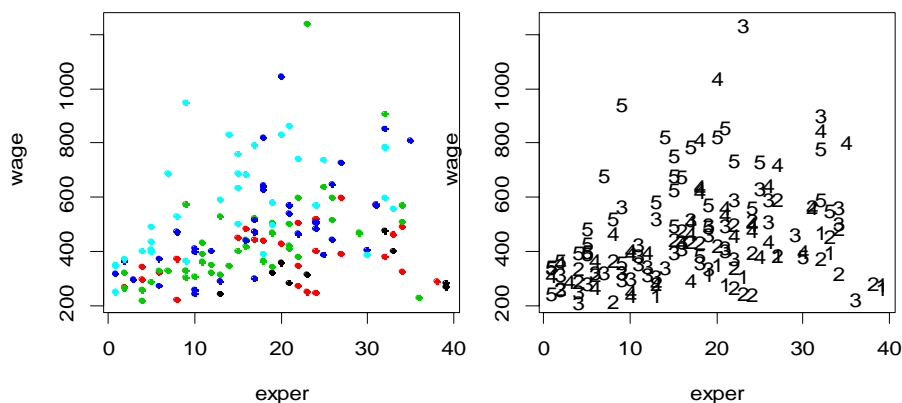
6.10 pav. Visų pch simbolių lentelė

Pažymėsime, kad taškų žymėjimo variantų yra dar daugiau – taškus galima nuspalvinti arba vietoje grafinių simbolių vartoti įvairius simbolinius kintamuosius:

```

plot(wage~exper,pch=20,col=as.numeric(educf))
plot(wage~exper,pch=as.character(educf))

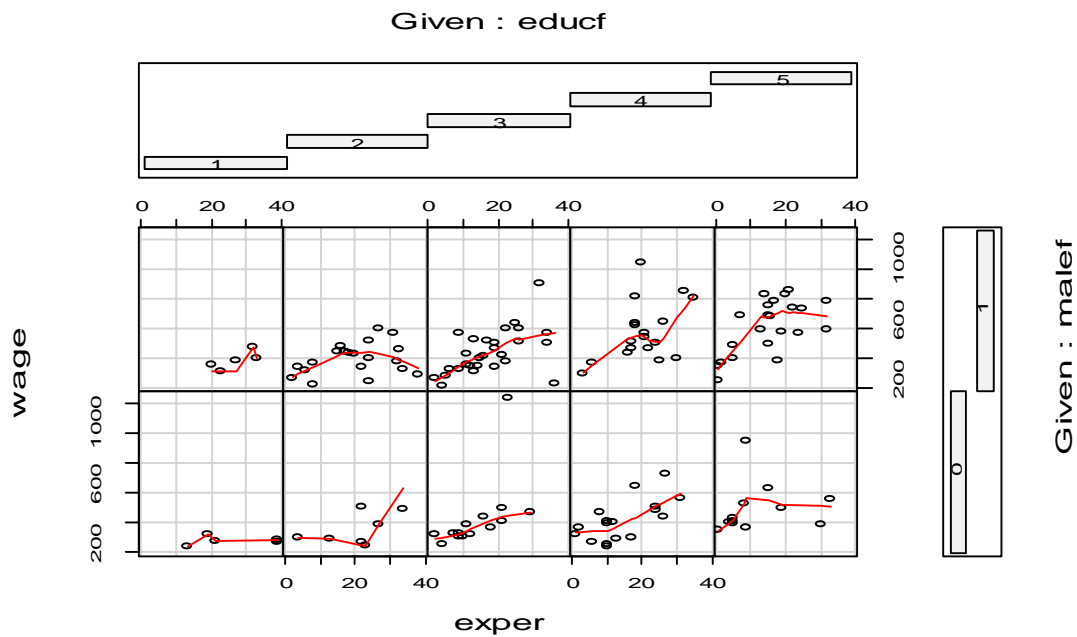
```



6.11 pav. Atlyginimo ir patyrimo sklaidos diagramos: skirtingomis spalvomis (kairėje) ar skirtingais skaitmenimis (dešinėje) žymima išsilavinimo grupė

Paskutiniuose grafikuose spalvų ir skirtingų simbolių aiškiai per daug, geriau vis tik būtų išbrėžti penkis atskirus grafikus. `coplot` leidžia pasirinkti du sąlyginius kintamuosius, mes pasirinksim `educf` ir `malef`.

```
> coplot(wage~exper|educf*malef,panel=panel.smooth,data=Bwf)
```



6.12 pav. Atlyginimo ir patyrimo kografikas (su fiksuotomis lyties ir išsilavinimo reikšmėmis)

Štai dar vienas `coplot` funkcijos vartojimo pavyzdys (čia įdomiausias yra duomenų rinkinio `VADeaths`² transformacijos – duomenų formatą gana dažnai reikia keisti).

```
> data(VADeaths)
> VADeaths
```

	Rural Male	Rural Female	Urban Male	Urban Female
50-54	11.7	8.7	15.4	8.4
55-59	18.1	11.7	24.3	13.6
60-64	26.9	20.3	37.0	19.3
65-69	41.0	30.9	54.6	35.1
70-74	66.0	54.3	71.1	50.0

Šią matricą perrašysime kaip stulpelį, tačiau dar sudarysime antrą, trečią ir ketvirtą stulpelius, kuriuose nurodysime amžių, lytį ir gyvenimo vietą.

```
> dr <- c(VADeaths) # Matricą VADeaths paversime vektoriumi
> dr
 [1] 11.7 18.1 26.9 41.0 66.0  8.7 11.7 20.3 30.9 54.3 15.4 24.3 37.0
[14] 54.6 71.1  8.4 13.6 19.3 35.1 50.0
> n <- length(dr)
> rep(ordered(rownames(VADeaths)),length=n)
 [1] 50-54 55-59 60-64 65-69 70-74 50-54 55-59 60-64 65-69 70-74
[11] 50-54 55-59 60-64 65-69 70-74 50-54 55-59 60-64 65-69 70-74
Levels: 50-54 < 55-59 < 60-64 < 65-69 < 70-74
> gl(2,5,n, labels= c("M", "F")) # Funkcija gl gamina faktorius
 [1] M M M M M F F F F F M M M M M F F F F F
Levels: M F
> gl(2,10, labels= c("rural", "urban"))
 [1] rural rural rural rural rural rural rural rural rural rural
```

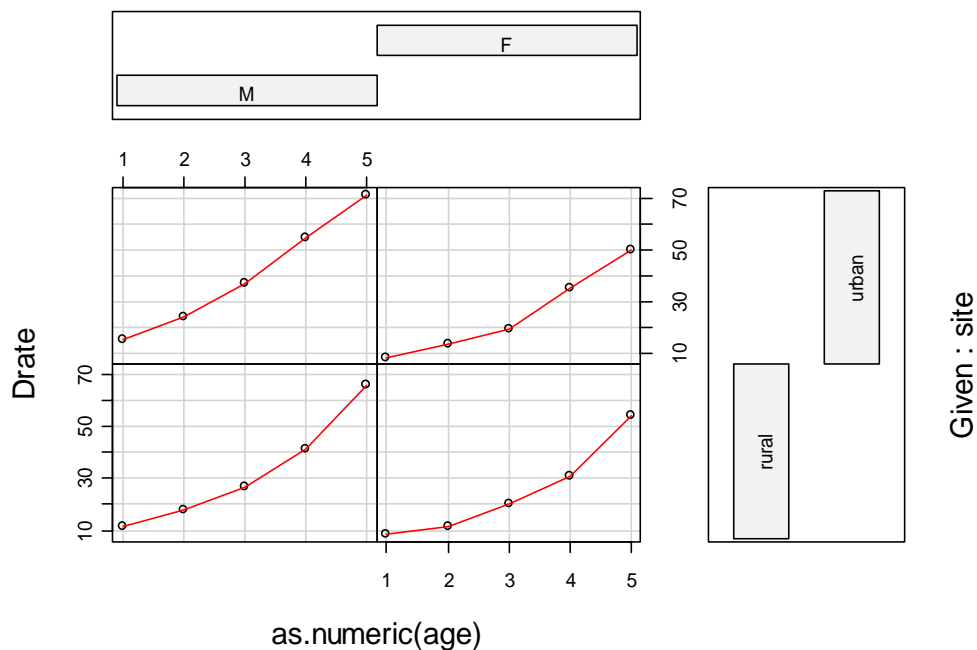
² Tai mirusiųjų skaičius 100-ai gyventojų įvairiose amžiaus grupėse. Duomenys sugrupuoti pagal gyvenamo vietą (kaimas ar miestas) ir lytį (vyras ar moteris).

```

[11] urban urban urban urban urban urban urban urban urban urban urban
Levels: rural urban
> d.VAD <- data.frame (Drate=dr, age=rep(ordered(rownames(VADeaths)),
length=n),gender=gl(2,5,n,labels= c("M", "F")), site=gl(2,10, labels
=c("rural", "urban")))
> d.VAD
  Drate  age gender  site
1  11.7 50-54     M rural
2  18.1 55-59     M rural
3  26.9 60-64     M rural
4  41.0 65-69     M rural
5  66.0 70-74     M rural
6   8.7 50-54     F rural
7  11.7 55-59     F rural
8  20.3 60-64     F rural
9  30.9 65-69     F rural
10 54.3 70-74     F rural
11 15.4 50-54     M urban
12 24.3 55-59     M urban
13 37.0 60-64     M urban
14 54.6 65-69     M urban
15 71.1 70-74     M urban
16  8.4 50-54     F urban
17 13.6 55-59     F urban
18 19.3 60-64     F urban
19 35.1 65-69     F urban
20 50.0 70-74     F urban
> mode(d.VAD)
[1] "list"
> class(d.VAD)
[1] "data.frame"
> coplot(Drate ~ as.numeric(age) | gender * site, data = d.VAD,
panel = panel.smooth)

```

Given : gender



6.13 pav. Drate ir amžiaus kografikas (su fiksuotomis lyties ir gyvenamosios vietos reikšmėms)

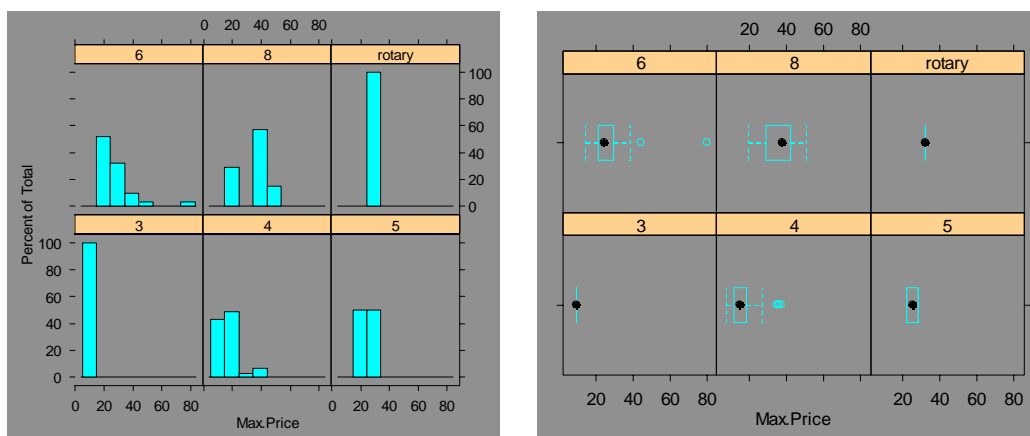
Kografikų metodus yra smarkiai patobulintas lattice bibliotekoje. Panagrinėkime Cars93 duomenis iš MASS bibliotekos.

```
library(lattice); library(MASS)
data(Cars93) # ?Cars93
> Cars93[1:5,]
  Manufacturer Model Type Min.Price Price Max.Price MPG.city MPG.highway
1 Acura Integra Small 12.9 15.9 18.8 25 31
2 Acura Legend Midsize 29.2 33.9 38.7 18 25
3 Audi 90 Compact 25.9 29.1 32.3 20 26
4 Audi 100 Midsize 30.8 37.7 44.6 19 26
5 BMW 535i Midsize 23.7 30.0 36.2 22 30
  AirBags DriveTrain Cylinders EngineSize Horsepower RPM
1 None Front 4 1.8 140 6300
2 Driver & Passenger Front 6 3.2 200 5500
3 Driver only Front 6 2.8 172 5500
4 Driver & Passenger Front 6 2.8 172 5500
5 Driver only Rear 4 3.5 208 5700
  Rev.per.mile Man.trans.avail Fuel.tank.capacity Passengers Length Wheelbase
1 2890 Yes 13.2 5 177 102
2 2335 Yes 18.0 5 195 115
3 2280 Yes 16.9 5 180 102
4 2535 Yes 21.1 6 193 106
5 2545 Yes 21.1 4 186 109
  Width Turn.circle Rear.seat.room Luggage.room Weight Origin Make
1 68 37 26.5 11 2705 non-USA Acura Integra
2 71 38 30.0 15 3560 non-USA Acura Legend
3 67 37 28.0 14 3375 non-USA Audi 90
4 70 37 31.0 17 3405 non-USA Audi 100
5 69 39 27.0 13 3640 non-USA BMW 535i
```

(mes pakrovėme Cars93 duomenis, tačiau, norėdami pademonstruoti “data=” opciją (žr. komandas žemiau), neprijungsimė jų su attach). Pagrindinė lattice idėja yra grafinį langą suskaidyti į keletą polangių (paprastai jie nusakomi koku nors sąlygos kintamuoju). Funkcijos (jų vardai natūralūs, tačiau skiriasi nuo įprastų – pvz., rašome histogram ir bwplot vietoje, atitinkamai, hist ir boxplot) naudoja formulių sintaksę (plg. 5.2 pvz., 5.2 užd., 5-10 psl.). Vienmačių grafikų atveju (o dvi aukščiau užrašytos funkcijos brėžia tokius) kairiąją formulės ženklą ~ puse paliekame tuščią. Pvz., komandos

```
> histogram( ~ Max.Price | Cylinders , data = Cars93)
> bwplot( ~ Max.Price | Cylinders , data = Cars93)
```

pateikia tą pačią informaciją dviem skirtingais pavidalais:



6.14 pav. Kiekvienam cilindų skaičiaus variantui, čia išbrėžtos maksimalios kainos histograma ir stačiakampė diagrama

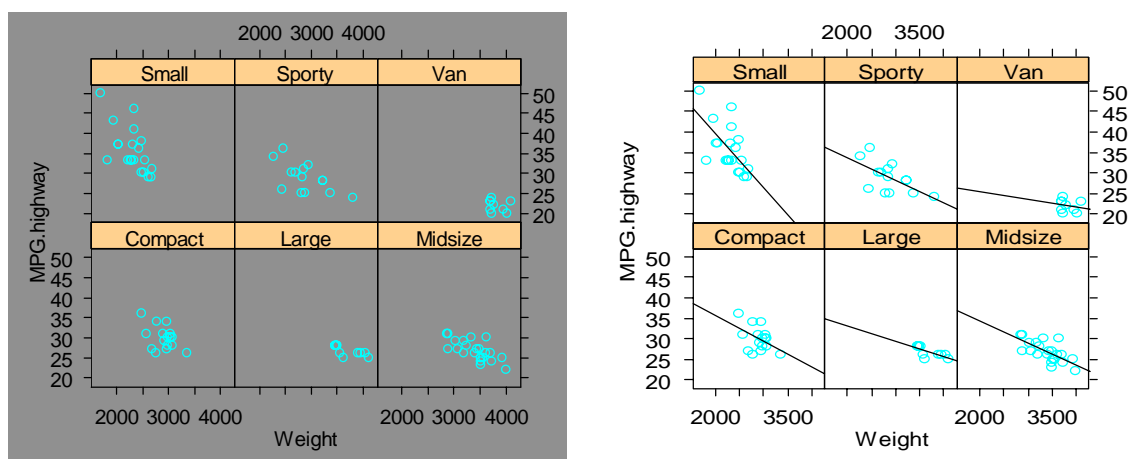
Su lattice taip pat galima brėžti (sąlygines) sklaidos diagramas (tik vietoje plot reikės rašyti xyplot ir, be to, naudoti formulių sintaksę)).

```
attach(Cars93)
xyplot(MPG.highway ~ Weight | Type) # Žr. 6.15 pav., kairėje
```

Tendencijos yra aiškios: kuo didesnis automobilio svoris, tuo mažiau mylių su vienu kuro galonu galima nuvažiuoti. Antra vertus, šias tendencijas geriausiai pavaizduoti regresijos tiesėmis. Funkciją, brėžiančią šias tieses lattice atveju, teks parašyti patiems.

```
plot.regression=function(x,y) # ženklas "=" yra ženklo "<-" sinonimas
{
  panel.xyplot(x,y)
  panel.abline(lm(y~x))
}
```

```
trellis.device(bg="white") # fonas bus baltas
xyplot(MPG.highway ~ Weight | Type, panel = plot.regression)
```



6.15 pav. Sunaudojamo kuro kiekio priklausomybės nuo automobilio svorio grafikai įvairiose automobilių grupėse (dešinėje – kartu su regresijos tiesėmis)

Matome, kad į svorį jautriausiai reaguoja mažieji automobiliai. Atkreipkite dėmesį į tai, kad `trellis.device` komanda standartinį (kiek tamsoką) foną pakeičia baltu.

6.3 UŽDUOTIS. Su lattice pakartokite keliais puslapiais aukščiau atliktą analizę su `Drate` ir `age`.

Daugiamačių duomenų vaizdavimas yra komplikuoatas uždavinys. Be jau aptartų galimybių, dar paminėsime `cloud` ir `wireframe` funkcijas iš lattice bibliotekos, bibliotekas `djmrpl` ir `rgl`, o taip pat išorinę programinę įrangą `xgobi` ir `ggobi`.

6.3. Skaitinės charakteristikos

Jau žinome, kad dažnai naudingos ėminių skaitinės charakteristikos. Daugiamačiu atveju be vidurkio (ar medianos) ir standarto (ar IQD) dar naudojama ėminio koreliacijos matrica.

Pradėkime nuo vienmačių skaitinių charakteristikų.

```
> round(apply(bwages,2,summary),2) # Funkcijoje apply(...,2,...) skaičius
# 2 reiškia "stulpeliams"
      wage lnwage educ exper lnexper lneduc male
Min.   88.38  4.48 1.00  0.00    0.00  0.00 0.00
1st Qu. 327.30 5.79 3.00  9.00    2.30  1.10 0.00
Median  408.50 6.01 3.00 16.50    2.86  1.10 1.00
Mean    445.80 6.03 3.38 17.22    2.69  1.14 0.61
3rd Qu. 514.60 6.24 4.00 24.00    3.22  1.39 1.00
Max.   1919.00 7.56 5.00 47.00    3.87  1.61 1.00
```

Matome, kad, pvz., vyrų ėminyje yra 61% (kodėl?), o vidutinis išsilavinimas lygus 3,38 (neužmirškime, kad educ yra ranginis kintamasis, todėl šis skaičius nieko ypatingo nereiškia).

```
> round(apply(bwages,2,sd),2) # Stulpelių (=kintamųjų) standartai
      wage lnwage educ exper lnexper lneduc male
179.53  0.36  1.20 10.17  0.73  0.43  0.49
```

Dabar apskaičiuosime koreliacijos matricą.

```
> round(cor(bwages),2) # Koreliacijos matrica
      wage lnwage educ exper lnexper lneduc male
wage   1.00  0.95  0.39  0.31  0.33  0.36  0.14
lnwage 0.95  1.00  0.40  0.31  0.34  0.37  0.15
educ   0.39  0.40  1.00 -0.29 -0.27  0.97 -0.14
exper  0.31  0.31 -0.29  1.00  0.93 -0.31  0.16
lnexper 0.33  0.34 -0.27  0.93  1.00 -0.28  0.15
lneduc 0.36  0.37  0.97 -0.31 -0.28  1.00 -0.15
male   0.14  0.15 -0.14  0.16  0.15 -0.15  1.00
```

Matome, kad, pvz., paprastasis (Pearson'o) koreliacijos koeficientas tarp educ ir male lygus $-0,14$. Kadangi tai ranginiai kintamieji, vietoje paprastojo reikia skaičiuoti ranginį (Spearman'o) koreliacijos koeficientą. Jis pagrįstas ne kintamojo reikšmėmis, bet jų rangais (vieta).

```
> a <- c(3,8,6,6,9,100)
> rank(a)
[1] 1.0 4.0 2.5 2.5 5.0 6.0 # 3 yra mažiausias, todėl jo rangas 1;
# 6 užima 2-ą ir 3-ią vietas, todėl jo
# vidutinis rangas 2,5

> a <- c(3,8,6,6,9,10)
> rank(a)
[1] 1.0 4.0 2.5 2.5 5.0 6.0 # Nesvarbu kas, 10 ar 100 yra
# didžiausias - rangas nuo to nesikeičia
```

Jei x ir y surišti tikslia tiesine priklausomybe – jų Pirsono koreliacijos koeficientas lygus $+1$ arba -1 , tačiau jei tikslia paraboline ar eksponentine – koeficientas nebūtinai artimas vienetui. Tuo tarpu, jei Spirmeno koeficiento reikšmės moduliui artimos 1, šitai signalizuoja apie monotoniško (bet nebūtinai tiesinio) trendo egzistenciją:

```
> x <- 1:50
> cor(x,x)
[1] 1 # Priklausomybė tiesinė, todėl =1
```

```

> cor(x,exp(x))
[1] 0.3525162 # Pirsono kor.koef. toli nuo 1
> cor(rank(x),rank(x)) # Spirmeno koreliacijos koeficientas
# tarp x ir y apibrėžiamas kaip
# cor(rank(x),rank(y))
# Spirmeno k.k. lygus 1
[1] 1
> cor(rank(x),rank(exp(x)))
[1] 1 # Spirmeno k.k. vėl lygus 1 (kadangi
# exp(x) monotoniškai priklauso nuo x)

```

Grįžkime prie koreliacijos tarp educ ir male.

```

> attach(bwages)
> cbind(educ,male,rank(educ),rank(male)) # Palyginkime tikrąsias
# ir ranžuotas reikšmes

educ male
[1,] 1 1 50 1026 # educ reikšmių (suvidurkinti)
[2,] 1 0 50 290 # rangai lygūs 50; 232; 574.53 ir t.t.
[3,] 1 1 50 1026
[4,] 1 1 50 1026 # male: (vidutinis) moters rangas
[5,] 1 1 50 1026 # lygus 290, o vyro - 1026
[6,] 1 0 50 290
[7,] 1 1 50 1026
[8,] 1 1 50 1026
[9,] 1 1 50 1026
[10,] 1 1 50 1026
[11,] 1 0 50 290
[12,] 1 0 50 290
[13,] 1 1 50 1026
[14,] 1 0 50 290
[15,] 1 1 50 1026
[16,] 1 0 50 290
.....

> cor(educ,male) # Pearson'o koeficientas
[1] -0.1396446
> cor(rank(educ),rank(male)) # Spearman'o koeficientas
[1] -0.1389874

```

Pažymėsime, kad abu koeficientai praktiškai sutampa. Beje, minuso ženklas rodo neigiamo trendo buvimą – didėjant educ reikšmei, male reikšmė mažėja (nuo 1 link 0), kitais žodžiais, moterų išsilavinimas aukštesnis nei vyrų. Tai galima patikrinti dar ir taip:

```

> tapply(educ,male,mean)
 0 1
3.587219 3.243001

```

6.4 UŽDUOTIS. Išnagrinėkite trimatei grafikai skirtą D. Murdoch'o paketą djmrgl (žr. kompaktinį diską R1 arba <http://www.stats.uwo.ca/faculty/murdoch/software/>) ir pateikite jo apžvalgą.

```

3 > table(rank(educ))
 50 232 574.5 962.5 1306.5
 99 265 420 356 332

```


6.5 UŽDUOTIS. Išsiaiškinkite `plot.table` ir `mosaicplot` funkcijas. Išnagrinėkite duomenų rinkinį `UCBAdmissions`.

6.6 UŽDUOTIS. Išsinagrinėkite žemiau pateiktą pavyzdį (jis priklauso Ross'ui Ihaka'ui, vienam iš R kūrėjų).

```
# First create some fake data. The data will be in the form
# of a matrix, with each row summing to 100.

# Start by generating a matrix of random uniforms and adding 4.
xxx <- function(){
x <- matrix(runif(200), nc=4) + 4

# Compute the row sums and normalise each row by its sum.
rowsums <- apply(x, 1, "sum")
y <- 100 * apply(x, 2, "/", rowsums)

# Now replace each row by its cumulative sums
# The t() is needed to get the right shape
z <- t(apply(y, 1, "cumsum"))

# Get the "x" values to plot against (YMMV).
xvals <- 1:nrow(z)

# Ok, we're ready to plot.
# Start a new plot and set up the plot window.
# The xaxs= and yaxs= remove the 6% padding at the plot edges.
plot.new()
plot.window(xlim = c(0, nrow(z)), ylim=c(0, 100),
            xaxs="i", yaxs = "i")

# Now draw the bars for each time interval.
for(i in 4:1)
  rect(xvals - 1, 0, xvals, z[,i], col = i + 1, border = NA)

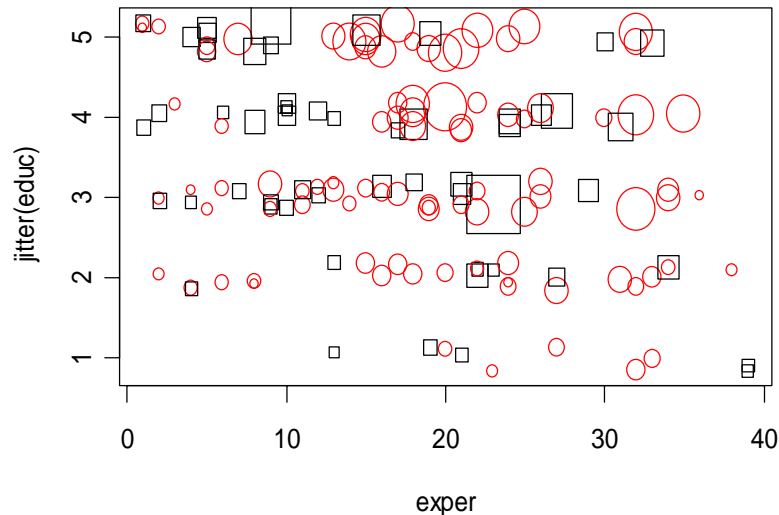
# Finally add the cross-hatching grid and axes.
# I've rotated the labels on the y axis so they are horizontal.
abline(h = seq(5, 95, by = 5))
abline(v = seq(5, 45, by = 5))
axis(1)
axis(2, las=1)
box()}
```

Metai	Mieste	Kaime	1986	2341	1219
1970	1558	1561	1987	2390	1207
1971	1602	1559	1988	2440	1195
1972	1661	1536	1989	2487	1188
1973	1721	1509	1990	2527	1181
1974	1773	1486	1991	2557	1179
1975	1828	1460	1992	2568	1179
1976	1884	1431	1993	2549	1187
1977	1934	1409	1994	2533	1191
1978	1981	1386	1995	2526	1191
1979	2035	1357	1996	2518	1193
1980	2074	1330	1997	2534	1173
1981	2121	1302	1998	2525	1179
1982	2167	1276	1999	2523	1178
1983	2211	1259	2000	2522	1176
1984	2256	1244	2001	2516	1177
1985	2298	1230			

Šios programos idėjas panaudokite, vaizduodami Lietuvos miesto ir kaimo gyventojų santykio kitimą (žr. lentelę kairėje; joje gyventojų skaičius nurodytas tūkstančiais). Horizontalioje ašyje pažymėkite metus. Atspausdinkite gautųjų santykių vektorių. Išbrėžkite panašų grafiką su `plot` funkcija

6.7 UŽDUOTIS. Pateiksime “keturmačio” grafiko brėzimo pavyzdį:

```
attach(Bwages)
plot(exper, jitter(educ), cex=wage/250, col=male+1, pch=male)
```



6.16 pav. “Keturmatė” išsilavinimo ir patyrimo sklaidos diagrama

Paaikškinkite šį paveikslą. Patobulinkite jį, papildydami informatyvia legenda (išsifruokite simbolių ir simbolių didumo reikšmes). Išbrėžkite panašų grafiką su Cars93 duomenimis iš MASS bibliotekos (imkite $x=EngineSize$, $y=Price$, o cex ir col parametrus pasirinkite patys).

6.8 UŽDUOTIS. Realiuose duomenų rinkiniuose dažnai būna praleistų duomenų (žemiau pateiktoje matricoje xx jie žymimi simboliu NA). Matricą xx pasigaminome taip: į “urną” supylėme visus 250 aibės x elementų, po to iš urnos “atsitiktinai” paėmėme vieną elementą, jį užregistravę gražinome atgal ir vėl procedūrą pakartojome (iš viso 30000 kartų – tai padarė funkcija `sample`).

```
x <- c(1:200, rep(NA, 50))
set.seed(45) # Duomenų sistema xx dabar bus reprodukuojama
xx <- data.frame(matrix(sample(x, 30000, replace=T), ncol=3))
xxx <- xx[9900:9920, ]
xxx
[,1] [,2] [,3]
[1,] 16 NA 141
[2,] 39 128 NA
[3,] 194 185 144
[4,] NA 10 94
[5,] 142 92 NA
[6,] 52 42 41
[7,] NA NA 188
[8,] 140 NA 179
[9,] NA NA 127
[10,] 54 173 123
[11,] 31 24 NA
[12,] NA 117 52
```

```
[13,] NA 49 178
[14,] NA NA NA
[15,] 139 182 31
[16,] 15 8 20
[17,] 21 180 75
[18,] 27 64 NA
[19,] 11 196 NA
[20,] 73 134 152
[21,] 82 NA NA
```

Kadangi matrica `xx` turi praleistų reikšmių, kai kurios funkcijos (pvz., `mean` ar `sum`) neveiks:

```
> apply(xx,2,sum) # Skaičiuojame stulpelių sumas
[1] NA NA NA
```

Aprašysime dvi procedūras, kurios pašalins iš duomenų sistemos su trim stulpeliais visas eilutes su bent vienu NA.

1.

```
d1 <- function(x){
x1 <- subset(x,!is.na(x[,1])) # subset yra base paketo funkcija
x2 <- subset(x1,!is.na(x1[,2]))
x3 <- subset(x2,!is.na(x2[,3]))
x3}
```

```
> d1(xxx)
      X1 X2 X3
9900 166 107 2
9901  36  25 63
9902  61  41 51
9904  47  43 72
9905 151  12 116
9909 174  16 66
9913 120  67 170
9915   8 172 103
9919  45 174 59
9920 185   6 152
```

Norint išsiaiškinti kaip dirba ši (ar kokia kita) funkcija, ją galima kiek modifikuoti:

```
d1.1 <- function(x){
x1 <- subset(x,!is.na(x[,1]))
print(is.na(x[,1])) # Funkcijos is.na reikšmė yra loginis vektorius
print(!is.na(x[,1])) # Simbolis ! reiškia loginį neigimą
print(x1) # Pašalintos eilutės, kurių 1-jame stulpelyje buvo NA
x2 <- subset(x1,!is.na(x1[,2]))
print(x2)
x3 <- subset(x2,!is.na(x2[,3]))
x3}
```

2.

```
d2 <- function(x){
x1 <- subset(x,!is.na(x[,1])&!is.na(x[,2])&!is.na(x[,3]))
x1}
```

Žemiau yra funkcijos, kurioms stulpelių skaičius nesvarbus.

3.

```
subset(xxx, complete.cases(xxx)) # complete.cases yra base paketo  
# funkcija
```

4.

```
xxx[complete.cases(xxx), ]
```

5.

```
xxx[-c(unique(which(apply(is.na(xxx), FUN=any, MARGIN=1)==TRUE))), ]
```

6. Jei norite pašalinti eilutes, sudarytas tik iš NA, FUN=any pakeiskite į FUN=all.

7.

```
d7 <- function(x){  
x1 <- x[!is.na(as.matrix(x) %*% rep(1, ncol(x))), ]  
x1}
```

8. Štai paprasčiausias variantas:

```
na.omit(xxx)
```

Kartais pageidautina, kad duomenų sistemoje nebūtų pasikartojančių eilučių. Tai galima pasiekti su funkcija `unique.data.frame`, o štai vienas jos taikymo variantų:

```
dim(unique.data.frame(na.omit(xx)))  
[1] 5164 3
```

Taigi iš pradinių 10000 eilučių beliko 5164.

O dabar pati

6.9 UŽDUOTIS. Nukopijavę eilutes

```
pradzia <- proc.time()  
invisible(d1(xx))  
(proc.time() - pradzia)[3]
```

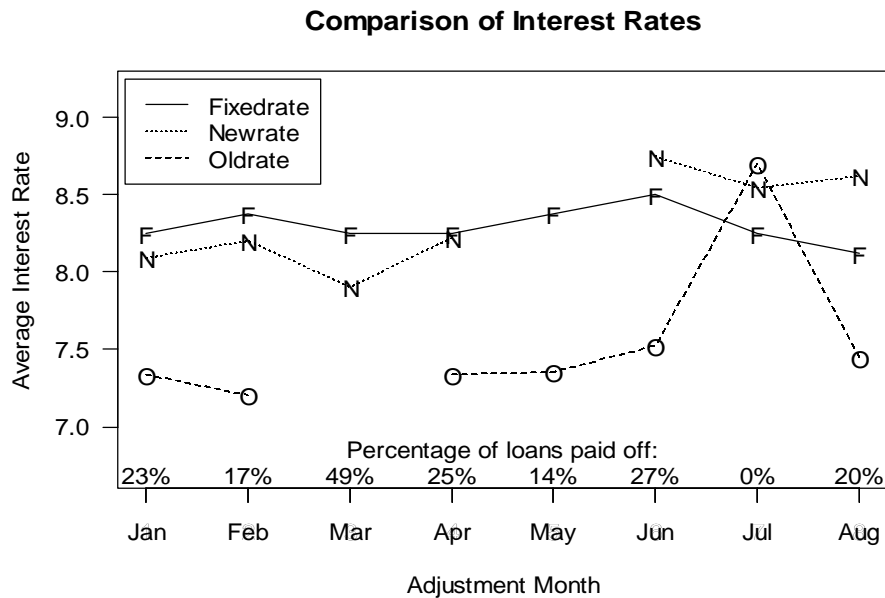
į R konsolę, pamatysite kiek laiko truko `d1` procedūra⁴. Kuri iš pateiktų aštuonių funkcijų skaičiuoja greičiausiai? Kaip priklauso šis laikas nuo `xx` didumo? Parašykite funkcijų `d1` ir `d2` variantus, kurie išmestų tik eilutes, sudarytas vien iš NA.

6.10 UŽDUOTIS. Štai duomenų rinkinys `loan.tab`:

```
num adjmon newrate oldrate fixedrate paid  
1 Jan-00 8.09091 7.33541 8.250 0.23  
2 Feb-00 8.20139 7.20833 8.375 0.17  
3 Mar-00 7.90457 8.03989 8.250 0.49  
4 Apr-00 8.22500 7.34167 8.250 0.20  
5 May-00 8.41346 7.35577 8.375 0.08  
6 Jun-00 8.75000 7.52273 8.500 0.27  
7 Jul-00 8.54348 8.70109 8.250 0.00  
8 Aug-00 8.61923 7.44231 8.125 0.08
```

⁴ Pakartoję šią procedūrą kelis kartus, galite gauti skirtingus laikus.

Remdamiesi jo duomenimis, išbrėžkite žemiau pateiktą grafiką. Paaiškinkite jį.



6.17 pav. Paskolų palūkanų normų grafikas

6.11 UŽDUOTIS. Tiriant per savaitę alkoholiniams gėrimams išleidžiamų pinigų sumą (kintamasis amount – svarais sterlingų per savaitę), buvo gauti tokie duomenys (duomenų rinkinys drink):

```
sex age empl amount
0 20 0 8.83
1 33 0 4.90
1 50 1 0.71
0 48 0 5.70
0 47 0 6.20
0 19 0 7.40
1 21 1 3.58
0 64 0 4.80
1 32 0 4.50
1 57 1 2.80
0 49 0 4.60
0 18 0 5.30
1 39 1 3.42
0 28 0 10.15
0 51 0 6.20
1 43 0 4.80
1 40 0 3.82
0 22 1 7.70
1 30 0 6.20
0 60 0 4.45
```

(sex: vyras 1, empl: dirba 0). Ištikrinkite kiekvieną kintamąjį ir jų tarpusavio ryšius.

6.12 UŽDUOTIS. Štai dvi pagalbinės funkcijos

```
panel.tab <- function (x, y)
{
  par(new = TRUE)
```

```
fourfoldplot(table(x, y))
}
```

ir

```
panel.chisq <- function(x, y)
{
  par(new = TRUE)
  usr <- par("usr")
  on.exit(par(usr))
  par(usr = c(0, 1, 0, 1))
  x2 <- formatC(chisq.test(table(x, y))$p.value,
                digits = 4, format = "f")
  text(0.5, 0.5, x2, cex = 0.75)
  print(x2)
}
```

dirbtiniai duomenys (tai dvi reikšmes įgyjančių vardinių kintamųjų pavyzdys)

```
x <- matrix(rbinom(300,1,0.5),100,3)
```

ir `pairs` funkcijos variantas:

```
pairs(x, upper.panel = panel.tab, lower.panel = panel.chisq)
```

Išsiaiškinkite visų pateiktų funkcijų darbą ir pateikite savą skaitinį pavyzdį.

6.13 UŽDUOTIS. Paketo `Simple` duomenų rinkinys `chips` yra sąrašas (kaip tai sužinoti?), turintis aštuonias komponentes. Išbrėžkite kiekvienos komponentės stačiakampę diagramą (grafiniame lange turi būti aštuonios diagramos, po kiekviena užrašykite jos vardą (plg. 4.3 užduotį). Kaip jums atrodo: ar "centrų" padėtys vienodos? O išsibarstymai?

6.14 UŽDUOTIS. Paketo `Sample` duomenų rinkinyje `chicken` pateikti duomenys apie viščiukų svorį trijose grupėse. Viename brėžinyje išbrėžkite visų trijų grupių stačiakampes diagramas. Kaip manote, ar vienodi svorių vidurkiai šiose grupėse? O reikšmių išsibarstymas?

6.15 UŽDUOTIS. Pakete `base` yra duomenų rinkinys `airquality`. Taikydami funkciją

```
> cor(airquality,use="pairwise.complete.obs"),
```

apskaičiuokite komponentių koreliacijos koeficientų matricą. Parašykite funkciją, kuri atspausdintų tik tuos (ne ant įstrižainės esančius) koeficientus, kurių modulis didesnis už 0,5 (kartu su atitinkamų eilučių ir stulpelių vardais).

6.16 UŽDUOTIS. Su komandomis

```
library(MASS)
data(Cars93)
Cars93
```

galima apžiūrėti didžiulią duomenų sistemą `Cars93`.

```
> names(Cars93)
[1] "Manufacturer" "Model" "Type" "Min.Price"
[5] "Price" "Max.Price" "MPG.city" "MPG.highway"
[9] "AirBags" "DriveTrain" "Cylinders" "EngineSize"
[13] "Horsepower" "RPM" "Rev.per.mile" "Man.trans.avail"
[17] "Fuel.tank.capacity" "Passengers" "Length" "Wheelbase"
[21] "Width" "Turn.circle" "Rear.seat.room" "Luggage.room"
[25] "Weight" "Origin" "Make"
```

Kad tolimesnė analizė būtų paprastesnė, apsiribokime tik dalimi šių parametų.

```
> cars <- Cars93[,c(1,3,5,8,25,26)]
> names(cars)
[1] "Manufacturer" "Type" "Price" "MPG.highway" "Weight" "Origin"
```

Bendrą supratimą apie cars galima susidaryti su summary (atkreipkite dėmesį į tai, kad ši funkcija atsižvelgia į kintamojo klasę).

```
> summary(cars)
  Manufacturer      Type      Price      MPG.highway      Weight      Origin
Ford      : 8 Compact:16  Min.      : 7.40  Min.      :20.00  Min.      :1695  USA      :48
Chevrolet : 8 Large  :11  1st Qu.  :12.20  1st Qu.  :26.00  1st Qu.  :2620  non-USA:45
Dodge     : 6 Midsize:22  Median   :17.70  Median   :28.00  Median   :3040
Pontiac   : 5 Small  :21  Mean     :19.51  Mean     :29.09  Mean     :3073
Mazda     : 5 Sporty :14  3rd Qu. :23.30  3rd Qu. :31.00  3rd Qu. :3525
Volkswagen: 4 Van    : 9  Max.     :61.90  Max.     :50.00  Max.     :4105
(Other)   :57
```

Pažvelkime į Weight kintamąjį – skirtumas tarp minimumo ir maksimumo yra didelis, tačiau tai gali būti paaiškinta tuo, kad automobilio svoris priklauso nuo jo Type. Suskaidykime duomenų sistemą cars į grupes pagal Type reikšmes (plg. 6-15 psl.).

```
> names(split(cars,Type))
[1] "Compact" "Large" "Midsize" "Small" "Sporty" "Van"
> summary(split(cars,Type)$Compact) # Sinonimas: summary(split(cars,Type)[[1]])
  Manufacturer      Type      Price      MPG.highway      Weight      Origin
Chevrolet :2  Compact:16  Min.      :11.10  Min.      :26.00  Min.      :2490  USA      :7
Volvo     :1  Large  : 0  1st Qu.  :13.38  1st Qu.  :27.75  1st Qu.  :2783  non-USA:9
Volkswagen:1  Midsize: 0  Median   :16.15  Median   :30.00  Median   :2970
Subaru    :1  Small  : 0  Mean     :18.21  Mean     :29.88  Mean     :2918
Saab      :1  Sporty : 0  3rd Qu. :20.68  3rd Qu. :31.00  3rd Qu. :3043
Pontiac   :1  Van    : 0  Max.     :31.90  Max.     :36.00  Max.     :3375
(Other)   :9
```

Apskaičiuokite automobilių skaičių kiekvienoje Type ir Origin grupėje. Atspausdinkite Price skaitines charakteristikas ir išbrėžkite histogramas kiekvienai Type grupei. Nustatykite, keli kiekvieno gamintojo (Manufacturer) automobiliai yra rinkinyje cars; štai tie gamintojai:

```
> names(split(cars,Manufacturer)) # Sinonimas: levels(cars$Manufacturer)
[1] "Acura" "Audi" "BMW" "Buick" "Cadillac"
[6] "Chevrolet" "Chrysler" "Chrysler" "Dodge" "Eagle"
[11] "Ford" "Geo" "Honda" "Hyundai" "Infiniti"
[16] "Lexus" "Lincoln" "Mazda" "Mercedes-Benz" "Mercury"
[21] "Mitsubishi" "Nissan" "Oldsmobile" "Plymouth" "Pontiac"
[26] "Saab" "Saturn" "Subaru" "Suzuki" "Toyota"
[31] "Volkswagen" "Volvo"
```

Išbrėžkite benzino sunaudojimo MPG.highway ir svorio Weight sklaidos diagramą. Išbrėžkite joje regresijos tiesę.

6.17 UŽDUOTIS. Su

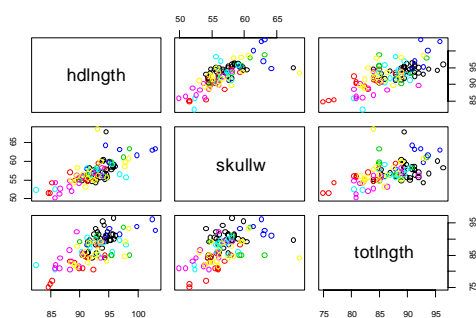
```
library(MASS)
data(Insurance)
Insurance
> Insurance
  District  Group  Age Holders Claims
1         1    <11  <25    197     38
2         1    <11  25-29    264     35
*****
```

galima apžiūrėti duomenų rinkinį Insurance. a) Nustatykite kiekvieno kintamojo tipą ir klasę. b) Kiek įrašų yra kiekvienoje Age grupėje? c) Atspausdinkite lentelę, kurios eilutėse būtų Group, stulpeliuose - Age, o langeliuose – suminė Claims reikšmė. d) Išbrėžkite Claims stačiakampės diagramas kiekvienai District reikšmei. Identifikuokite išskirtis.

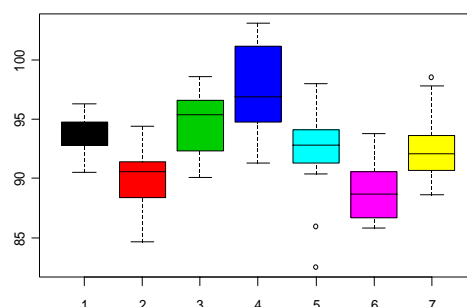
6.18 UŽDUOTIS. Su source nusikopijuokite duomenų rinkinį possum.R iš R1 direktorijos Data\Maindonald. Jame pateikti oposumų (sterbinių žiurkių) morfometrinių matavimų rezultatai (hdlngth = galvos ilgis, skullw = kaulolės svoris, totlngth = bendras ilgis).

```
> possum[1:5,c(2,4,6:8)]
  site sex hdlngth skullw totlngth
C3    1  m   94.1   60.4    89.0
C5    1  f   92.5   57.6    91.5
C10   1  f   94.0   60.0    95.5
C15   1  f   93.2   57.1    92.0
C23   1  f   91.5   56.3    85.5
```

```
pairs(possum[,6:8],col=palette()[as.integer(possum$sex)])
pairs(possum[,6:8],col=palette()[as.integer(possum$site)])
attach(possum); boxplot(hdlngth~site,col=1:7) # žr. 6.19 pav.
```



6.18 pav. Oposumų morfometrinių matavimų porinės sklaidos diagramos



6.19 pav. hdlngth stačiakampės diagramos su visomis site reikšmėmis

Patobulinkite 6.18 pav. (ant įstrižainės atspausdinkite atitinkamo kintamojo stačiakampės diagramas (plg. 6.7 pav.)).

Pašalinkite iš `possum` visas eilutes su trūkstamais duomenimis. Apskaičiuokite ir “gražiai” atspausdinkite visų trijų kintamųjų vidurkius ir standartus kiekvienoje `sex` ir `site` grupėse.

6.19 UŽDUOTIS. Duomenų rinkinio `Bwages` kintamasis `educ` įgyja penkias reikšmes.

```
> attach(Bwages)
> levels(factor(educ))
[1] "1" "2" "3" "4" "5"
```

Norint išskirti įrašus tik su `educ` lygiu 2, reikia elgtis taip:

```
Bwages[educ==2, ]
```

arba

```
Bwages[educ>1&educ<3, ]
```

O kaip išskirtumėte įrašus su nelyginėmis `educ` reikšmėmis?

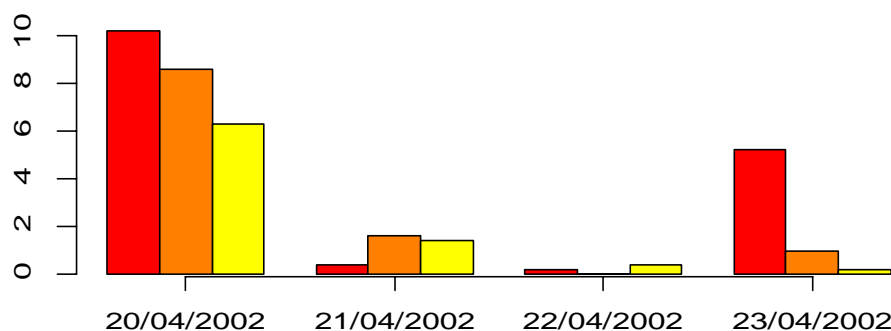
Beje, štai vienas variantas:

```
Bwages[is.element(educ, c(1,3,5)), ]
```

6.20 UŽDUOTIS. Kiekvieną dieną kritulių kiekis yra registruojamas trijuose punktuose:

```
      date    p1    p2    p3
20/04/2002 10.2  8.6  6.3
21/04/2002  0.4  1.6  1.4
22/04/2002  0.2  0.0  0.4
23/04/2002  5.2  1.0  0.2
```

Išbrėžkite tokį grafiką:

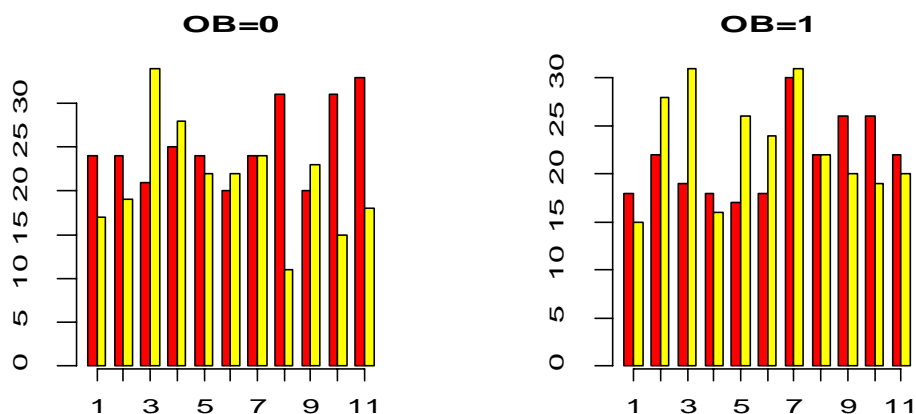


6.20 pav. Kritulių kiekio trijuose stebėjimo punktuose stulpelinės diagramos

6.21 UŽDUOTIS. Dirbtiniame duomenų rinkinyje `mydata` (tai 1000x3 duomenų sistema) yra trys stulpeliai: `OB` - 0 reiškia, kad asmuo neturi viršsvorio, 1 – kad turi,

GEN – 1 žymi moterį, 2 – vyrą ir AGEGR – tai kintamasis, įgyjantis reikšmes nuo 1 iki 11 (jis žymi amžiaus grupę). a) Generuokite matricą `mydata` su tokiais sąryšiais: jei `GEN==1`, `OB` dažniau įgyja reikšmę 1 (t.y., moterys dažniau turi viršsvorį); asmenų su viršsvoriu procentas didėja vyresnėse amžiaus grupėse; b) Išbrėžkite stulpelines diagramas, rodančias vyrų ir moterų su viršsvoriu skaičių įvairiose amžiaus grupėse. *Nuoroda.* Plg. skyrelį 10.2. Be to, atliekant b) užduotį, gali praversti `xtabs` funkcija:

```
mytable <- xtabs(~ GEN + AGEGR, data = subset(mydata,OB == 0))
```



6.21 pav. Atsakymas galėtų atrodyti maždaug taip

6.22 UŽDUOTIS. Keičiant “ilgą” sistemą į “placią” (plg. 6.1 užd.) ir atvirkščiai, gali būti naudinga `reshape` funkcija. Remdamiesi šia funkcija (arba kaip nors kitaip), tranformuokite “placią” sistemą

```
> xx
  Year Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec
1 1993  -9 -10 -12 -24  -9 -18 -11 -18  -9 -15  -1  0
2 1994  -2  -1 -14 -26 -13 -12 -18 -20 -19 -16  -9 -15
3 1995  -4  -5  2 -19  -9  -3  4  -1  3  -2  0  -8
```

į “ilgą”

```
  Year Month  xx
1 1993  Jan  -9
2 1993  Feb -10
3 1993  Mar -12
4 1993  Apr -24
...
36 1995  Dec  -8
```

(gal būt pravers `Month <- names(xx)[-1]` eilutė).

6.23 UŽDUOTIS. Matricoje

```
vv <- matrix(rnorm(100),ncol=4)
```

yra pateikti keturmačio vektoriaus stebėjimų rezultatai. Šio vektoriaus kovariacinę (koreliacinę) matricą galima suskaičiuoti su `Var <- var(vv)` (atitinkamai su `Cor <- cor(vv)`). Dabar tarkime, kad matricos `vv` nežinome, bet žinome tik jos kovariacinę 4×4 matricą `Var`. Apskaičiuokite koreliacinę matricą `Cor`.

7. Centrinė ribinė teorema ir didžiųjų skaičių dėsnis

Kartais sakoma, kad tikimybininkai turi vieną dievą – normalųjį arba Gauso skirstinį. Greitai pamatysime, kad šis teiginys nėra be pagrindo.

7.1. Centrinė ribinė teorema (vienmatis atvejis)

Tarkime, kad κ_n yra sėkmių skaičius po n Bernulio eksperimentų su sėkmės tikimybe p . Gerai žinoma, kad tikimybė $P(\kappa_n = k) = C_n^k p^k q^{n-k}$, $k = 0, 1, \dots, n$, o tikimybė

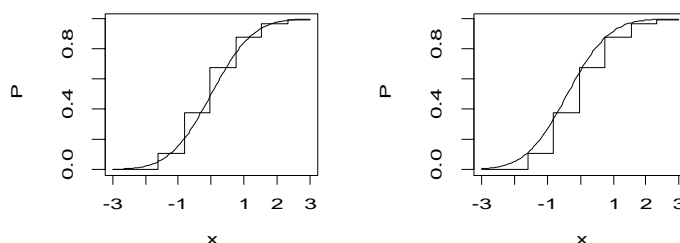
$$P(\kappa_n \leq x) = \begin{cases} 0, & , x < 0, \\ \sum_{k=0}^{[x]} C_n^k p^k q^{n-k} & , x \in [0, n], \\ 1, & , x \geq n \end{cases}$$

(čia $[x]$ žymi sveikąją skaičiaus x dalį). Kai n nėra didelis, šias tikimybes nėra sunku apskaičiuoti tiesiogiai, tačiau augant n skaičiavimai darosi vis sudėtingesni¹. A. de Muavras 1718 m. ir nepriklausomai nuo jo P. Laplasas 1812 m. įrodė, kad, kai n didelis, abi šias tikimybes galima gana sėkmingai apskaičiuoti, naudojant normalųjį skirstinį. Pavyzdžiui,

$$P(\kappa_n \leq x^*) = P\left(\frac{\kappa_n - np}{\sqrt{npq}} \leq \frac{x^* - np}{\sqrt{npq}} = x\right) = P(\kappa_n \leq x\sqrt{npq} + np) \approx \Phi(x), \quad x \in R.$$

(čia Φ yra standartinio normaliojo skirstinio funkcija). Vietoje matematinio šio fakto įrodymo, pateiksime grafines iliustracijas.

```
n <- 10
p <- 0.2
x <- seq(-3, 3, length=100)
P <- pbinom(x*sqrt(n*p*(1-p))+n*p, n, p) # pbinom(x, n, p) skaičiuoja
# tikimybę P(κn ≤ x)
plot(x, P, type="S") # Brėžia laiptuotą ("S"=Step) grafiką
lines(x, pnorm(x)) # pnorm(x)=Φ(x)
```



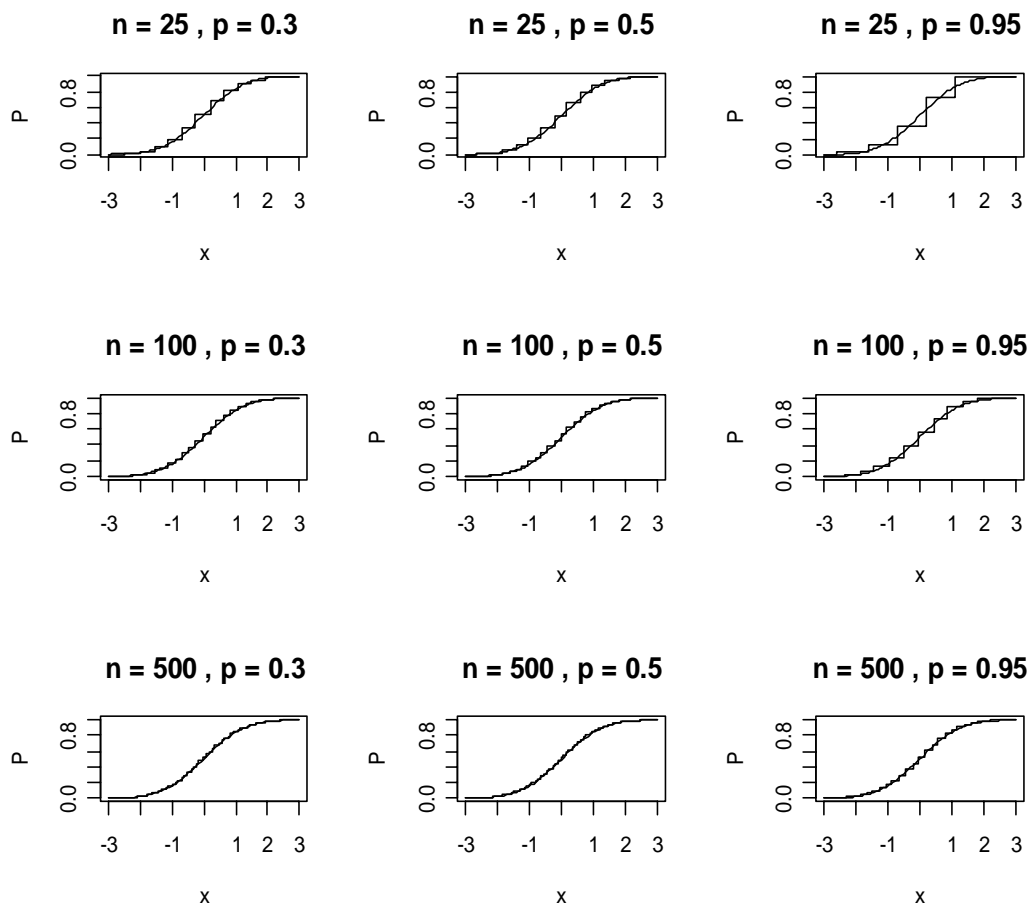
7.1 pav. Binominio dėsnio normalioji aproksimacija (be Yates'o pataisos kairėje ir su Yates'o pataisa dešinėje)

¹ Kompiuterių laikais tokie skaičiavimai nėra didelė problema. Svarbu tai, kad aproksimavomą normaliuju skirstiniu galima taikyti ne tik κ_n skirstiniui.

Matome, kad Φ grafikas yra gana arti (normuoto) binominio. Tais atvejais, kai x^* yra sveikas skaičius (ir n nėra didelis), tikslinga įvesti Jeitso diskretumo pataisą – vietoje $\Phi(x)$ imti $\Phi(x+1/2\sqrt{npq})$ (7.1 pav., grafikas dešinėje).

Normaliosios aproksimacijos tikslumas gerėja didėjant n (tikslumas taip pat geresnis, kai p arti 0,5). Norėdami tuo įsitikinti, parašykime funkciją `b.sim` (žemiau yra tvarkingo funkcijos rašymo pavyzdys; deja, tai ne Notepad'o darbas...):

```
b.sim <- function()
{
#function b.sim
  opar <- par(mfcol = c(3, 3))
  on.exit(par(opar))
  x <- seq(-3, 3, length = 100)
  for(p in c(0.3,0.5, 0.95)) {
    for(n in c(25, 100, 500)) {
      P <- pbinom(x * sqrt(n * p * (1 - p)) + n * p, n, p)
      plot(x, P, type = "S", main = paste("n =", n, ", p =", p))
      lines(x, pnorm(x))
    }
  }
}
```



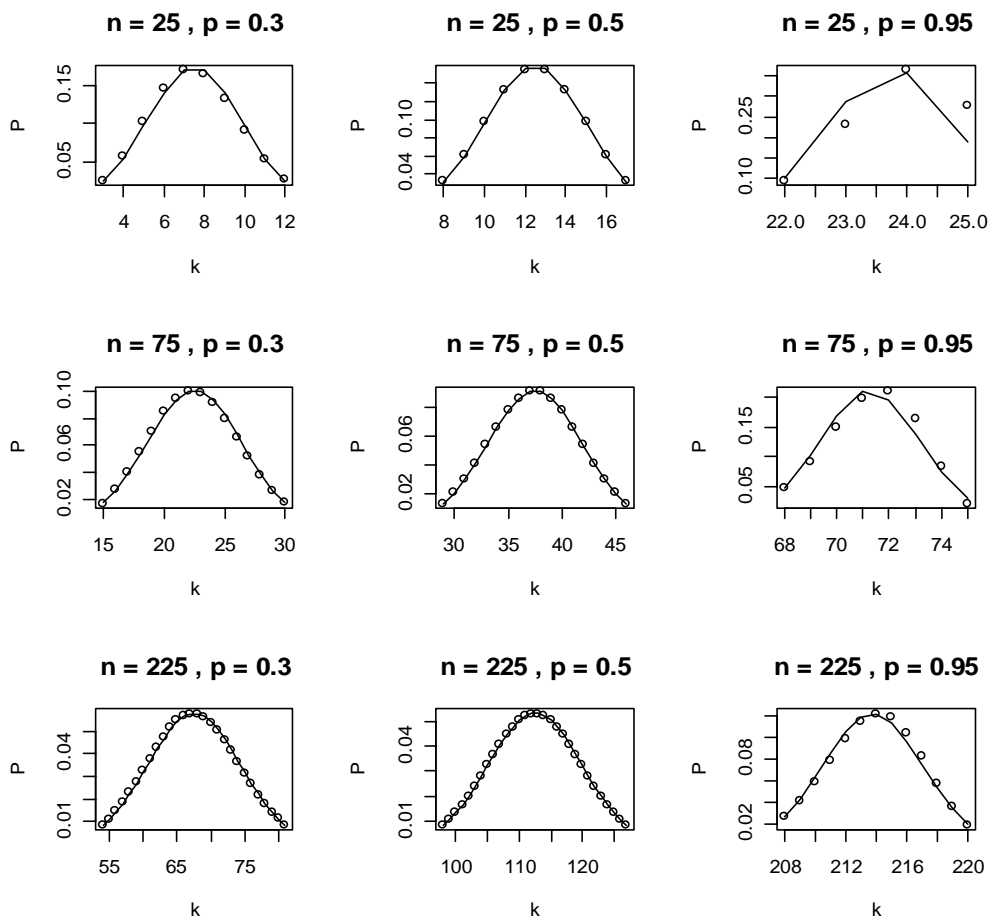
7.2 pav. Binominių skirstinių funkcijų normaliosios aproksimacijos

Tiesą sakant, aproksimacijų tikslumo skirtumai tarp $p=0,3$, $0,5$ ir $0,95$ nelabai pastebimi – akis blogai skiria skirstinio funkcijas (integralines charakteristikas), ji geriau pastebi

diferencialinius skirtumus (pvz., skirtumą tarp tikimybės $P(\kappa_n = k) = C_n^k p^k q^{n-k}$ ir jos normaliojo artinio $\varphi((k - np)/\sqrt{npq})/\sqrt{npq}$). Parašysime atitinkamą funkciją bb.sim.

```
bb.sim <- function(){
#function bb.sim
  opar <- par(mfcol = c(3, 3))
  on.exit(par(opar))
  for(p in c(0.3, 0.5, 0.95)) {
    for(n in c(25, 75, 225)) {
      k <- ceiling(n * p - 2 * sqrt(n * p * (1 - p))):floor(n
* p + 2 * sqrt(n * p * (1 - p)))
      P <- dbinom(k, n, p) # dnorm(k,n,p)=P(kn=k)
      plot(k, P, main = paste("n =", n, ", p =", p))
      lines(k, dnorm((k - n * p)/sqrt(n * p * (1 -
p)))/sqrt(n * p * (1 - p))) # dnorm(x)=φ(x)
    }
  }
}
```

Priminsime, kad standartinis normalus skirstinys yra praktiškai sukonzentruotas intervale (-2,+2) (iš tikrųjų $> \text{pnorm}(2) - \text{pnorm}(-2)$ [1] 0.9544997), todėl k vidiniame cikle kinta tarp $np - 2\sqrt{npq}$ ir $np + 2\sqrt{npq}$.



7.3 pav. Binominių tikimybių normalioji aproksimacija

7.3 pav. matome, kad, kai $p=0.5$, normalioji aproksimacija yra tiksli jau kai $n=25$; jei $p=0,3$ – n turi būti lygus bent 75, o kai $p=0,95$ net $n=225$ negarantuoja didelio tikslumo (kai p arti 0 ar 1, geresnė yra Puasono aproksimacija).

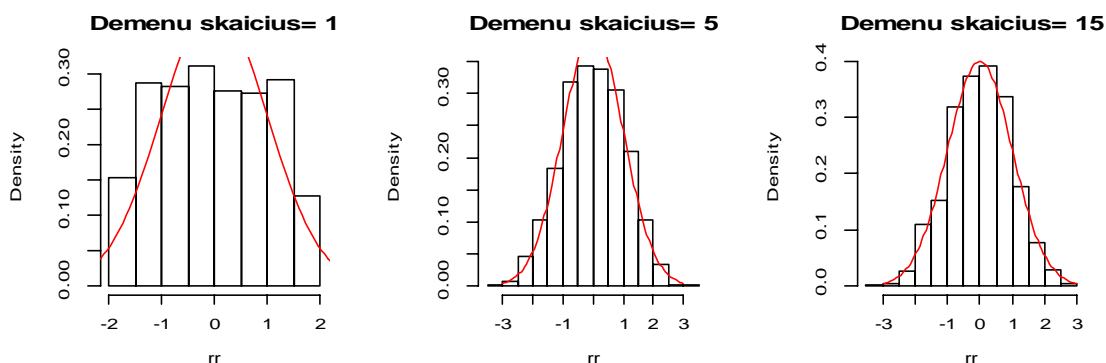
Muavro ir Laplaso teorema yra atskiras vadinamosios centrinės ribinės teoremos (CRT) atvejis. Tiksliau kalbant, tarkime, kad

$$X_i = \begin{cases} 1, & \text{su tikimybe } p, \\ 0, & \text{su tikimybe } q = 1 - p, \end{cases}$$

yra nepriklausomų Bernulio a.d. seka; tuomet sėkmių skaičius Bernulio modelyje $\kappa_n = S_n = X_1 + \dots + X_n$. Pasirodo, kad vietoje Bernulio a.d. galime paimti beveik bet kokius² nepriklausomus vienodai pasiskirsčiusius a.d. – kai n didelis, jų normuota suma $Z_n = (S_n - ES_n) / \sqrt{DS_n}$ turės (beveik) standartinį normalųjį skirstinį (tai ir vadinama CRT). Šitai paaiškina, kodėl normalusis dėsnis taip dažnai sutinkamas gyvenime – stebimą reiškinį nusako daug maždaug vienos svarbos priežasčių.

“Įrodykime” CRT nepriklausomų a.d., turinčių tolygų skirstinį intervale $[-1,+1]$, sekai. Deja, jau dviejų tokių dydžių sumos tankio išraiška gana sudėtinga, todėl sumos tankį pakeisime histograma.

```
uni.sim <- function(){
#function uni.sim
opar <- par(mfcol = c(1, 3))
on.exit(par(opar))
for(i in c(1,5,15)){ # Ciklas: imsime 1, 5 ar 15 dėmenų
rr <- numeric(1000) # Ilgio 1000 nulinis vektorius
for(n in 1:1000) {rr[n] <- sum(runif(i,-1,1))}
rr <- rr*sqrt(3/i) # Normuota suma
# Žemiau pateiktos eilutės skirtos normalumui įrodyti; vietoje hist
# galėtume remtis ir eda.shape funkcija
hist(rr,prob=T,main=paste("Dėmenu skaičius=",i))
k <- seq(-3,3,length=100)
lines(k, dnorm(k),col=2)
}
}
```



7.4 pav. Jau kai dėmenų penki, normuota suma Z_5 yra beveik normali

² Vienintelis reikalavimas: baigtinė a.d. X_i dispersija. Beje, kai kurie dydžiai ekonometrijoje turi tiek sunkias uodegas, kad ši sąlyga nėra patenkinta.

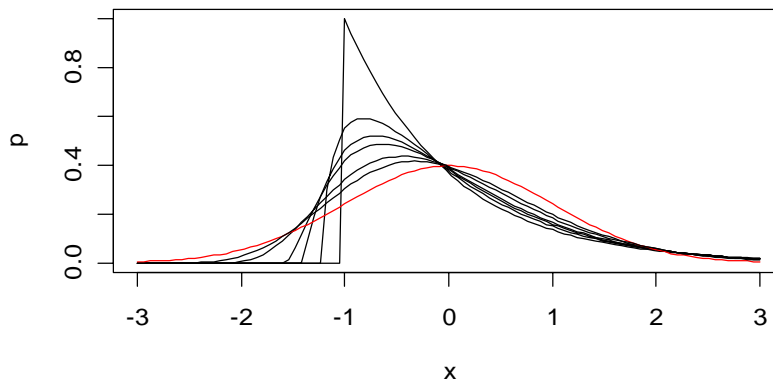
Panagrinėkime dar vieną pavyzdį. Tarkime, n.v.p.a.d. $X_i, i=1, \dots, n$, turi tokį pat skirstinį, kaip ir standartinio normaliojo a.d. kvadratas N^2 . Galėtume, kaip ir praeitame pavyzdyje, nagrinėti jų sumų histogramas, tačiau šį kartą pasiremsime tuo, kad S_n skirstinys yra žinomas – tai vadinamasis χ_n^2 (chi kvadrato su n laisvės laipsniais (l.l.)) skirstinys. Kadangi χ_n^2 yra sumos skirstinys, todėl, kai n didelis, jis turėtų beveik sutapti su normaliuoju. Tiksliau³,

$$\lim_{n \rightarrow \infty} P\left(\frac{\chi_n^2 - E\chi_n^2}{\sqrt{D\chi_n^2}} < x\right) = \lim_{n \rightarrow \infty} P(\chi_n^2 < \sqrt{2n}x + n) = \Phi(x),$$

arba, diferencijuojant šį reiškinį ir pereinant prie tankių⁴,

$$\lim_{n \rightarrow \infty} p_{\chi_n^2}(\sqrt{2n}x + n) \cdot \sqrt{2n} = \varphi(x).$$

Išbrėšime kelių tankių grafikus ir patikrinsime šį teiginį.



7.5 pav. Normuoto χ_n^2 tankis, kai $n=2,3,4,5,10$ ir 20 (juodos linijos) ir standartinis normalusis tankis (raudona linija) (X_1 tankis labai nesimetriškas, todėl konvergavimas lėtas)

7.1 UŽDUOTIS. Nagrinėkime normuotą sumą $Z_n = (S_n - nEX_1) / \sqrt{nDX_1}$; čia S_n yra n.v.p.a.d. X_1, \dots, X_n suma. Palyginkite $P(Z_n \leq x)$ ir $\Phi(x)$, $x = -3; -2,9; -2,8, \dots, 2,9; 3$ kai X_1 yra 1) Bernulio a.d. su $p=0,6$; $n=50000$ (remkitės tuo, kad S_{50000} turi binominį skirstinį su parametrais $n=50000, p=0,6$), 2) χ_1^2 , $n=1000$ (remkitės tuo, kad S_{1000} turi χ^2 skirstinį su 1000 l.l.), 3) eksponentinis a.d. su tankiu $\alpha e^{-\alpha x} 1_{(0, \infty)}(x)$, rate parametru ($\alpha = 2$) (žr. r exp) ir $n=1000$ (remkitės tuo, kad S_{1000} turi gama skirstinį su

³ $E\chi_n^2 = n, D\chi_n^2 = 2n$.

⁴ Beje: jei CRT galioja skirstinio funkcijoms (integralinė teorema), tai iš čia ne visuomet išplaukia ribinė teorema tankiams (lokalinė teorema)!

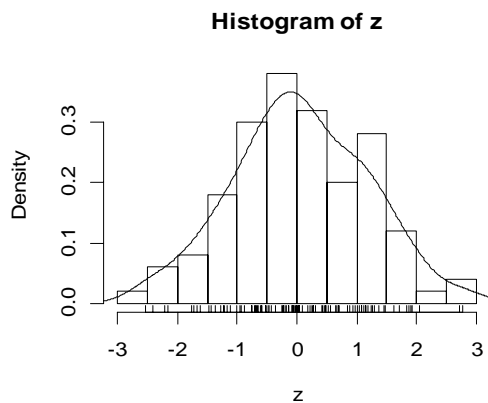
parametrais (žr. ?rgamma) shape=1000 ir scale=1/α). Atsakymą pateikite penkių stulpelių matricos pavidalu (pirmajame stulpelyje patalpinkite x reikšmes).

7.2 UŽDUOTIS. Nagrinėkime Pareto atsitiktinius dydžius (žr. 9 skyrių) su rodikliais 1) c=3, 2) c=2 ir 3) c=1. Remdamiesi histogramomis ir kitais jums prieinamais būdais, patikrinkite ar galioja šioms a.d. CRT.

7.3 UŽDUOTIS. Pateiksime kelis būdus, kaip rasti visų matricos stulpelių sumas:

```
> suma <- function(n)
  {# Stulpelių bus n
  # Eilučių bus 100
  m <- matrix(runif(100 * n, -1, 1), ncol = n)
  laikas <- numeric(5)
  #1 būdas (ciklas)
  dabar <- proc.time()[3]
  s <- NULL # Stulpelių sumų vektorius
  for(i in 1:n) {s[i] <- sum(m[, i])}
  laikas[1] <- proc.time()[3] - dabar
  # Dabar išbrėšime (normuotų) sumų histogramą
  z <- s/sqrt(100/3) # Normuojame sumas (CRT!)
  hist(z,probability=T)
  rug(z) # Pažymi s reikšmes ant x ašies
  lines(density(z))
  #2 būdas (apply funkcija)
  dabar <- proc.time()[3]
  s <- numeric(n) # Kitaip užrašytas stulpelių sumų vektorius
  s <- apply(m, 2, sum)
  laikas[2] <- proc.time()[3] - dabar
  #3 būdas
  dabar <- proc.time()[3]
  s <- NULL
  s <- drop(rep(1, nrow(m)) %*% m)
  laikas[3] <- proc.time()[3] - dabar
  #4 būdas (colSums kreipiasi į greitą vidinę funkciją)
  dabar <- proc.time()[3]
  s <- NULL
  s <- colSums(m)
  laikas[4] <- proc.time()[3] - dabar
  #5 būdas (lapply funkcija)
  dabar <- proc.time()[3]
  s <- NULL
  s <- unlist(lapply(split(m, col(m)), sum))
  laikas[5] <- proc.time()[3] - dabar
  cat(" Komputavimo trukme:\n", round(laikas, 4), "\n")
}

> suma(100) # Bus 100 stulpelių
Komputavimo trukme:
0.09 0.11 0 0.02 0.08
```



7.6 pav. Suglodintas empirinis z tankis panašus į normalųjį (kodėl?)

Pažymėsime, kad jei šią funkciją pakartotume kelis kartus, gautume skirtingus komputavimo laikus. Beje, ši kartą ciklas nėra pats lėčiausias būdas!

O dabar pati UŽDUOTIS. Pakeistite šią funkciją taip, kad ji skaičiuotų eilučių sumas. Išbrėžkite keturias histogramas viename lange, kai $n=1, 2, 4$ ir 10 . Matome, kad kai n didėja, z histograma darosi vis panašesnė į normaląją. Kodėl? (Užuomina: CRT!)

7.2. Centrinė ribinė teorema (daugiamatis atvejis)

CRT galioja ir daugiamatį atvejį. Tiksliau, jei stebime (pvz., dvimačių) atsitiktinių vektorių seką (X_{1i}, X_{2i}) , $i=1, \dots, n$, tai “gerais” atvejais jų normuotos sumos Z_n konverguoja į dvimatį normalųjį skirstinį (o “dar geresniais” atvejais - Z_n tankis konverguoja į dvimatį normalųjį tankį $\varphi_{\bar{a}, \Sigma}(\bar{x})$). Čia

$$\begin{aligned} \varphi_{\bar{a}, \Sigma}(\bar{x}) &= \frac{1}{2\pi \sqrt{\det \Sigma}} \exp\left(-\frac{1}{2} \bar{x} \Sigma^{-1} \bar{x}^T\right) = \\ &= \frac{1}{2\pi \sigma_1 \sigma_2 \sqrt{1-\rho^2}} \exp\left(-\frac{1}{2(1-\rho^2)} \left(\frac{(x_1 - a_1)^2}{\sigma_1^2} - 2\rho \frac{(x_1 - a_1)(x_2 - a_2)}{\sigma_1 \sigma_2} + \frac{(x_2 - a_2)^2}{\sigma_2^2} \right)\right), \end{aligned}$$

\bar{a} yra vidurkių vektorius, o $\Sigma = \begin{pmatrix} \sigma_1^2 & \sigma_1 \sigma_2 \rho \\ \sigma_1 \sigma_2 \rho & \sigma_2^2 \end{pmatrix}$ - kovariacijos matrica. Žemiau yra

pateikta funkcija mvdnorm, kuri skaičiuoja šį tankį. Atkreipiame dėmesį į dar vieną galimybę: visiems funkcijos argumentams nurodytos standartinės reikšmės (pvz., pirmasis komponentės vidurkis $a_1=0$), jei norime, kad koreliacijos koeficientas ρ būtų lygus, tarkime, $0,75$, turime rašyti `mvdnorm(..., ro=0.75)`.

```
mvdnorm <- function(x1,x2,a1=0,a2=0,s1=1,s2=1,ro=0){
# funkcija mvdnorm
# x1,x2 - bet kokie (skaiciai)
# a1,a2 - bet kokie
# s1,s2 - teigiami
# ro kinta nuo -1 iki +1
F1 <- 1/(2*pi*s1*s2*sqrt(1-ro^2))
F2 <- -1/(2*(1-ro^2))
```

```

S1 <- (x1-a1)^2/s1^2
S2 <- -2*ro*(x1-a1)*(x2-a2)/(s1*s2)
S3 <- (x2-a2)^2/s2^2
z <- F1*exp(F2*(S1+S2+S3))
z
}

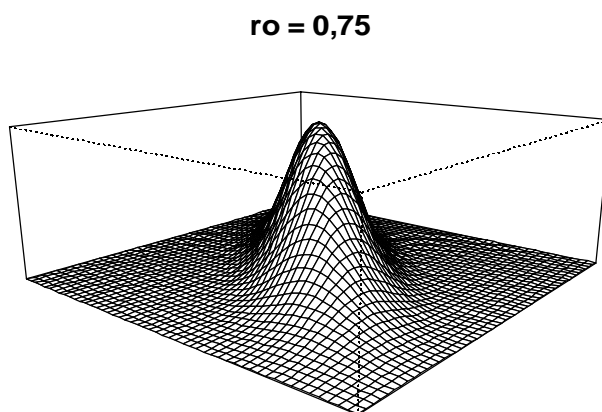
```

Deja, jau žinome, kad daugiamatė atveju grafinės galimybės gana ribotos. Štai vienas iš variantų (funkcija `persp`, ji turi labai daug opcijų), leidžiantis vizualizuoti dvimatę tankio funkciją. Prieš kreipdamiesi į funkciją `persp`, iš vektorių `x1` ir `x2` reikšmių turime sukurti stačiakampę gardelę ir kiekviename jos taške apskaičiuoti funkciją `mvdnorm` (tai atlieka funkciją `outer`, žr. žemiau; daugiklis 10 išryškina tankio kalvos pavidalo formą).

```

x1 <- seq(-3,3,len=50)
x2 <- x1
z <- 10*outer(x1,x2,function(x1,x2) mvdnorm(x1,x2,ro=0.75))
persp(x1,x2,z,theta=130,phi=15,scale=FALSE,axes=FALSE,main="ro=0,75")

```



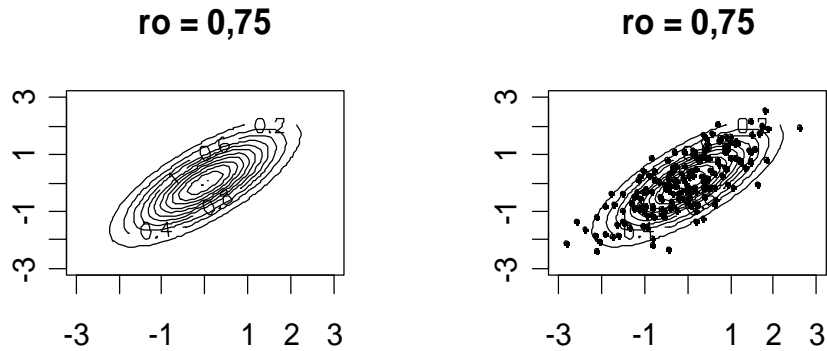
7.7 pav. Dvimačio normaliojo tankio grafikas

Geresnį supratimą apie tankio paviršiaus formą susidarysime, išbrėžę jo lygio linijas. Tai galime atlikti su funkcija `contour` (žr. 7.8 pav žemiau: dvimačio normaliojo tankio lygio linijos yra elipsės; atkreipsime dėmesį: kuo ρ modulis arčiau 1, tuo elipsės labiau “ištemptos”). Beje, MASS pakete yra funkcija `mvrnorm`, kuri generuoja daugiamatius normaliuosius a.d.. 7.8 pav. (dešinėje) lygio linijų grafikas yra papildytas sugeneruotų taškų vaizdais. Matome, kad tai būdingas sklaidos diagramos paveikslas, kitais žodžiais, dažnai stebime būtent dvimatį normalųjį skirstinį.

```

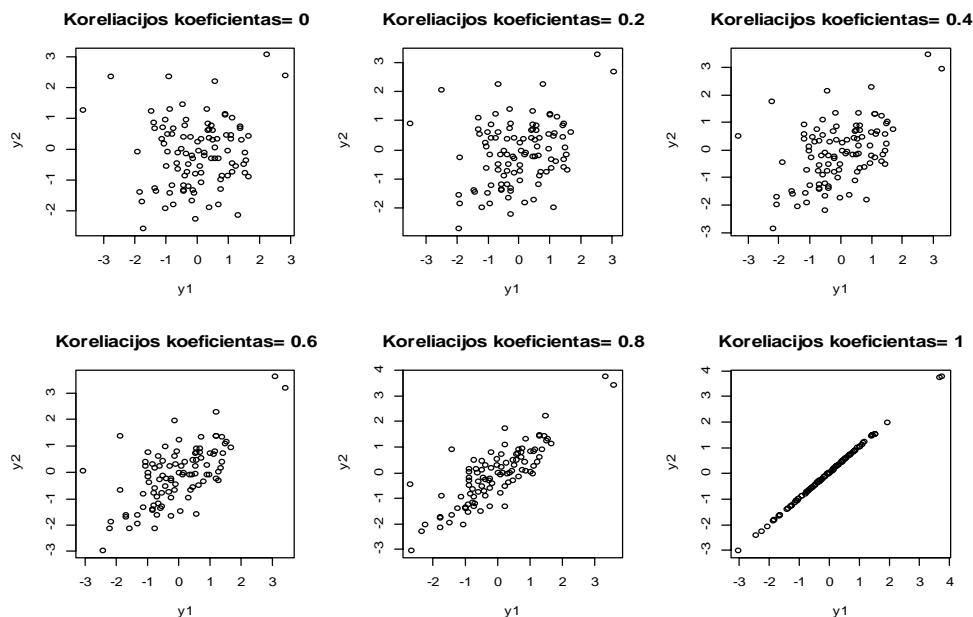
par(mfrow=c(1,2))
contour(x1,x2,z,main="ro = 0,75")
library(MASS)
Sigma <- matrix(c(1,0.75,0.75,1),2,2)
mvn <- mvrnorm(100,c(0,0),Sigma)
contour(x1, x2, z,main="ro = 0,75")
points(mvn[,1],mvn[,2],pch=16,cex=0.5)

```



7.8 pav. Dvimačio normaliojo tankio grafiko lygio linijos

Koreliacijos koeficientas ρ rodo vektoriaus koordinčių ryšio tamprumą (jei stebime dvimatį normalųjį a.d., sąlyga $\rho = 0$ yra ekvivalenti koordinčių nepriklausomumui). Nesunku įsitikinti, kad tik tuomet, kai $|\rho|$ didesnis už maždaug 0,6, elipsinė struktūra darosi ryškesnė (jei ρ lygus, pvz., 0,8, tai, žinant stebimojo atsitiktinio vektoriaus pirmąją koordinatę y_1 , galima gana tiksliai prognozuoti antrosios koordinatės y_2 reikšmę).



7.9 pav. Dvimačio normaliojo a.d. sklaidos diagramos su įvairiomis ρ reikšmėmis

7.4 UŽDUOTIS. Išbrėžkite 7.9 paveikslą.

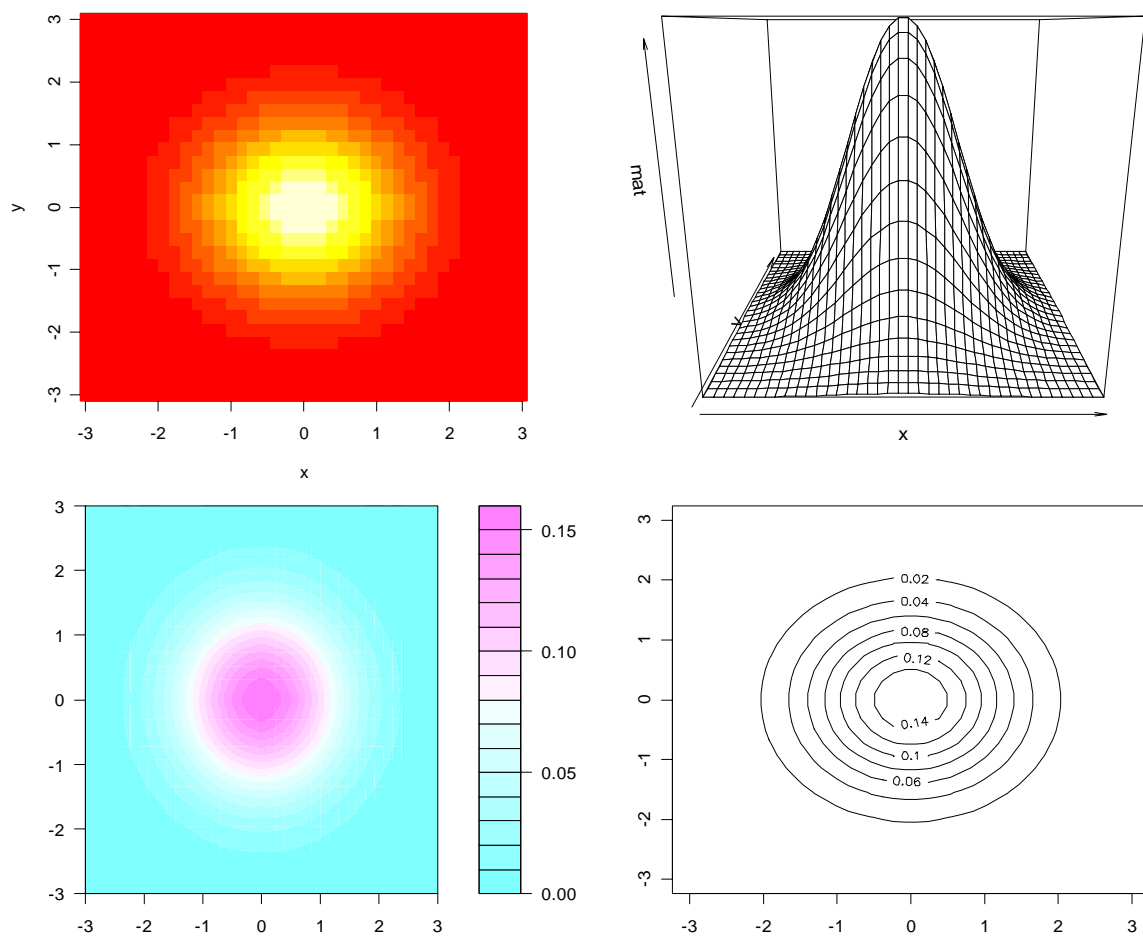
Dar keli žodžiai apie dviejų kintamųjų funkcijų (jų grafikai – paviršiai) vizualizaciją. Jau minėjome du variantus: funkcijas `persp` ir `contour`. Čia aptarsime dar dvi funkcijas: `image` ir `filled.contour`.

```
par(mfrow=c(1,1))
dens <- function(x, y) { dnorm(x) * dnorm(y) }
```

```

# Kuo ypatingas šis dvimatis normalusis tankis? Užuomina:
# nepriklausomumas.
x <- seq(-3, 3, len = 40)
y <- seq(-3, 3, len = 30)
g <- expand.grid(x, y)
mat <- matrix(dens(g[,1], g[,2]), nrow = 40, ncol = 30)
# Dvi aukščiau esančios eilutės yra funkcijos outer ekvivalentas
image(x,y,mat)
win.graph() # Atidaro naują grafinį langą
persp(x,y,mat)
win.graph()
filled.contour(x,y,mat)
win.graph()
contour(x,y,mat)

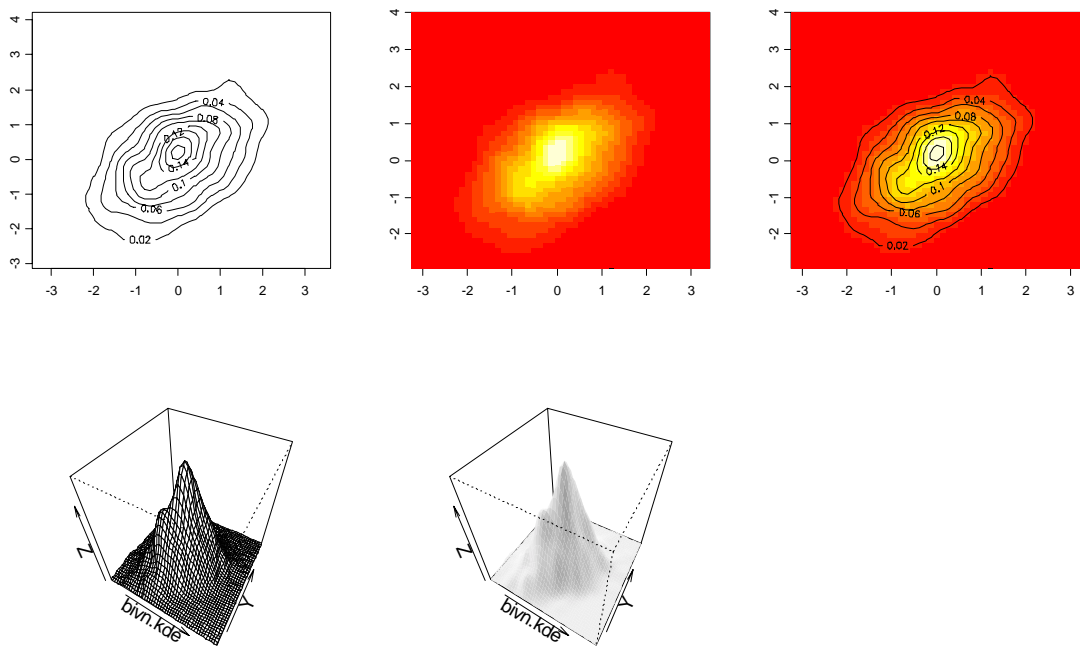
```



7.10 pav. Keturi dvimačio normaliojo tankio grafikai: image (kairėje viršuje), persp (dešinėje viršuje), filled.contour (kairėje apačioje) ir contour (dešinėje apačioje)

Vėliau visus papildomai atidarytus grafinius langus galima uždaryti su `dev.off()`.

7.5 UŽDUOTIS. Aukščiau esančiame paveiksle išbrėžti teorinio dvimačio normaliojo tankio grafikai. Parašykite programą, kuri išbrėžtų žemiau pateiktus sugludinto “empirinio tankio” grafikus.



7.11 pav. Dvimačio empirinio normaliojo tankio branduolinio įverčio įvairūs vizualizacijos tipai

Nuoroda. MASS bibliotekoje yra funkcija `mvrnorm`, kuri generuoja dvimačius normaliuosius a.d. Ten taip pat yra funkcija `kde2d`, kuri pateikia branduolinį dvimačio empirinio tankio įvertį (maždaug taip: `bivn.kde <- kde2d(bivn[,1], bivn[,2], n = 50)`). Brėždami paskutinį grafiką, pasinaudokite opcija `shade = 0.1`.

7.3. Didžiųjų skaičių dėsnis

Be CRT tikimybių teorijoje svarbų vaidmenį vaidina dar viena didelė ribinių teoremų klasė – tai didžiųjų skaičių dėsniai (DSD). Jei X_1, X_2, \dots yra n.v.p.a.d. seka su vidurkiu $EX_i \equiv a$, tai galima įrodyti, kad $S_n/n \rightarrow a$ (pagal tikimybę ir netgi, kas dar stipriau, su tikimybe 1), kai $n \rightarrow \infty$. Šitai ekvivalentu tam, kad

$$F_{(S_n/n)-a}(x) \rightarrow \begin{cases} 0, & \text{kai } x \leq 0, \\ 1, & \text{kai } x > 1. \end{cases}$$

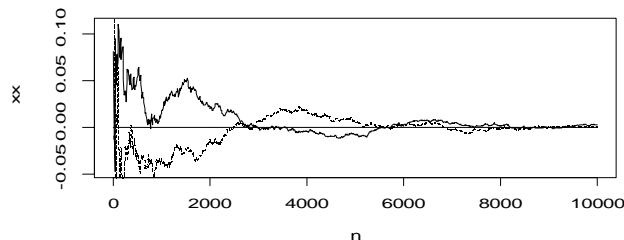
Iš DSD išplaukia, kad ėminio vidurkis \bar{x} (arba dispersija s^2) konverguoja į populiacijos vidurkį a (atitinkamai, dispersiją σ^2), santykinis dažnis konverguoja į tikimybę, empirinė skirstinio funkcija – į teorinę skirstinio funkciją. Didžiųjų skaičių dėsniu yra pagrįstas Monte Carlo metodas (žr. 3.3 skyrelį).

DSD pailustruosime dviem pavyzdžiais. Pirmiausiai, parodysime, kad didinant dėmenų skaičių, ėminio vidurkis artėja į populiacijos vidurkį.

```

DSD <- function(){
# funkcija DSD
n4 <- rnorm(10^4) # Generuojame 10000 standartinių normaliųjų ats.
# skaičių
xx <- numeric(1000)
for(i in 1:1000) xx[i] <- mean(n4[1:(10*i)])
# Ciklas: kaskart ėminių padidiname dešimčia (jei
# ciklo turinys telpa į vieną eilutę, figūrinių
# skliaustų nereikia)
n <- seq(10,10000,10)
plot(n,xx,type="l")
n4 <- rnorm(10^4) # Pakartojame visą procedūrą su kita seka
xx <- numeric(1000)
for(i in 1:1000) xx[i] <- mean(n4[1:(10*i)])
n <- seq(10,10000,10)
lines(n,xx,lty=2)
lines(n,rep(0,1000))
}

```



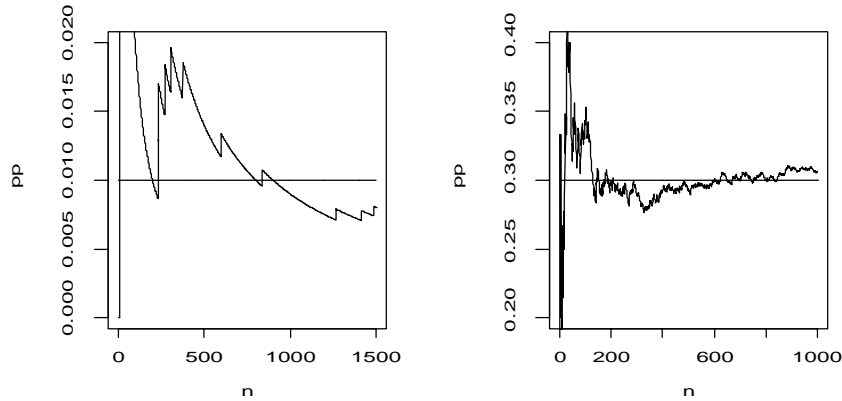
7.12 pav. Emino vidurkis (nemonotoniškai!) artėja į teorinį vidurkį (lygų nuliui)

7.6 UŽDUOTIS. Štai dinaminis (“kaip kine”) aukščiau pateiktos programos variantas, perrašytas Bernulio atsitiktiniams dydžiams (atitinkama DSD teorema vadinama Bernulio teorema, žr. [Ku, 145 p.]).

```

Bernoulli <- function(p,N=1000,lo=max(0,p-0.1),up=min(1,p+0.1))
{
# p - sekmes tikimybe, N - bandymu skaicius,
# lo - y ašies minimumas, o up - maksimumas
set.seed(4)
n <- 1:N
pp <- rep(p,N)
cum.b <- cumsum(rbinom(N,1,p))
plot(n,pp,xlim=c(1,N),ylim=c(lo,up),type="l")
for(i in 1:(N-1)) segments(i,cum.b[i]/i,i+1,cum.b[i+1]/(i+1))
cat("Sekmes tikimybes p ivertis \n(santykinis daznis)=", cum.b[N]/N,
"\n")
}
par(mfrow=c(1,2))
Bernoulli(0.01,N=1500,lo=0,up=0.02)
# Sekmes tikimybes p ivertis
# (santykinis daznis)= 0.008
Bernoulli(0.3)
# Sekmes tikimybes p ivertis
# (santykinis daznis)= 0.306

```



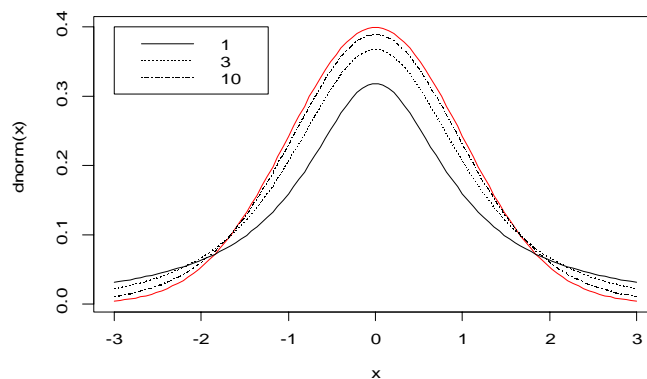
7.13 pav. Tipiškos santykių dažnių kitimo trajektorijos (kairėje $p=0,01$, dešinėje $p=0,3$)

O dabar pati UŽDUOTIS: išsiaiškinkite visus programos žingsnius ir pakomentuokite grafikus. Grafikams suteikite antraštes. Parašykite šios programos variantą Košy atsitiktiniams dydžiams. Pakomentuokite.

Antrajame pavyzdyje aptarsime Student'o su n l.l. skirstinį. Pagal apibrėžimą, šio a.d. tankis sutampa su a.d.

$$T_n = \frac{N}{\sqrt{(N_1^2 + N_2^2 + \dots + N_n^2)/n}} = \frac{N}{\sqrt{\chi_n^2/n}}$$

tankiu; čia N, N_1, N_2, \dots, N_n yra nepriklausomi standartiniai normalieji a.d. Pagal DSD vardiklis χ_n^2/n turi artėti į $EN^2=1$, t.y. T_n tankis tikriausiai artės prie N tankio. Pasirodo, kad taip ir yra.



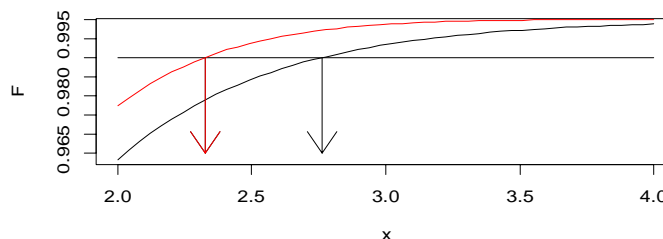
7.14 pav. Student'o tankio su 1, 3 ir 10 l.l. grafikai; kai l.l. skaičius neaprėžtai didėja, Student'o tankis artėja prie standartinio normaliojo (raudonas grafikas)

7.7 UŽDUOTIS. Nubrėžkite 7.14 paveikslą.

T_n tankio uodegos sunkesnės už normaliojo, todėl jo didelių α reikšmių kvantiliai (priminsime: skaičius x_α vadinamas a.d. X α eilės kvantiliu, jei jis tenkina lygybę $P(X < x_\alpha) = \alpha$, $\alpha \in (0,1)$) bus didesni už atitinkamus normaliuosius. Šitai galima

patikrinti bent trim būdais: 1) brėžiant kvantilių grafiką (žr. 4 skyrių); 2) lyginant skirstinio funkcijas (žr. 7.14 pav.); štai atitinkamos komandos:

```
x <- seq(2,4,length=50)
F <- pt(x,10) # pt(x,10) skaičiuoja Stjudento su 10 l.l.
# skirstinio f-ją "taške" x5
plot(x,F,col=1,type="l")
lines(x,pnorm(x),col=2)
arrows(qnorm(0.99),0.99,
qnorm(0.99),0.965,col=2)
# Žr. ?arrows
arrows(qt(0.99,10),0.99,
qt(0.99,10),0.965)
```



7.15 pav. Standartinio normaliojo (raudonas) ir Student'o su 10 l.l. (juodas) skirstinio funkcijų grafikai (x tarp 2 ir 4)

matome, kad standartinio normaliojo dėsnio 0,99 kvantilis (pažymėtas raudona strėle – jis lygus maždaug 2,35) yra mažesnis už Stjudento su 10 l.l. atitinkamą kvantilį (juoda strėlė – tai maždaug 2,75). Žinoma, galime ir 3) tiesiogiai apskaičiuoti kvantilius (rezultatams suteiksime lentelės pavidalą)

```
alpha <- seq(0.8,0.99,0.01)
t10 <- round(qt(alpha,10),2)
n <- round(qnorm(alpha),2)
matrix(c(t10,n),nrow=2,byrow=T,dimnames=list(c("t10","norm"),alpha))
```

	0.8	0.81	0.82	0.83	0.84	0.85	0.86	0.87	0.88	0.89
t10	0.88	0.92	0.96	1.00	1.05	1.09	1.14	1.19	1.25	1.31
norm	0.84	0.88	0.92	0.95	0.99	1.04	1.08	1.13	1.17	1.23

	0.9	0.91	0.92	0.93	0.94	0.95	0.96	0.97	0.98	0.99
t10	1.37	1.44	1.52	1.60	1.70	1.81	1.95	2.12	2.36	2.76
norm	1.28	1.34	1.41	1.48	1.55	1.64	1.75	1.88	2.05	2.33

7.8 UŽDUOTIS. Papildykite šią lentelę naujomis eilutėmis (kai l.l. skaičius kinta nuo 2 iki “daug”) ir išbrėžkite kvantilių elgesį iliustruojantį grafiką (x ašyje atidėkite α , o y ašyje $t_2, t_3, \dots, t_{10}, \dots, \text{norm}$).

7.9 UŽDUOTIS. Tarkime, kad nepriklausomi a.d. $X_i, i=1, \dots, 6$, įgyja tris reikšmes: 0 su tikimybe 0,3, 1 su 0,1 ir 2 su 0,6. Tarkime, $S_n = X_1 + \dots + X_n, n \geq 1$. Naudodamiesi Monte-Karlo metodu (jis – DSD išvada!), apskaičiuokite apytiksles S_1, \dots, S_6 reikšmių tikimybių reikšmes ir išbrėžkite jų grafikus.

7.10 UŽDUOTIS. Išsiaiškinkite ir pakomentuokite šias eilutes:

```
foo <- matrix(0, 49, 100)
for (i in 1:49)
  foo[i, ] <- density(rnorm(5000,mean = sqrt(i),sd = i^0.25),
  from = sqrt(1)-3*1, to = sqrt(49)+3*49^0.25, n = 100)$y
```

⁵ Priminsime, kad R pagrįstas vektorine aritmetika (čia x iš tikrųjų yra vektorius).

```
persp(foo, theta = 80, phi = 30, scale = FALSE, ltheta = -120, shade =  
0.75, border = NA)
```

Papildykite šią programą keliomis eilutėmis, kurios išbrėžtų gauto paviršiaus kontūrinės linijas.

8. Sprendžiamoji statistika: parametru įverčiai

Nors matematinės statistikos tikslas yra įvertinti visos populiacijos parametrus, tačiau ji gali remtis tik jos dalimi – ėminiu. Jau žinome, kad iš DSD išplaukia, kad populiacijos vidurkis maždaug lygus ėminio vidurkiui, tačiau jei norėtume apskaičiuoti ne populiacijos vidurkį, bet kitus ją aprašančio skirstinio parametrus, reiktų kitokių samprotavimų (žemiau aptarsime du metodus – momentų ir didžiausio tikėtimumo). Bet kuriuo atveju, dėl ėminio atsitiktinės prigimties, gautasis įvertis (jis vadinamas taškiniu) bus tik apytiksliai lygus populiacijos vidurkiui ar parametru reikšmėms. Norint nusakyti įverčio paklaidą, galima remtis pasikliauties intervalais – tai toks atsitiktinis intervalas, kuris su didele tikimybe uždengia atitinkamų populiacijos parametru reikšmes. Aišku, matematika turėtų pasiūlyti metodus, kaip surasti patį “geriausią” (pvz., trumpiausią) tokį intervalą.

8.1. Taškiniai įverčiai

Tarkime, kad turimi duomenys neprieštarauja hipotezei, jog stebimasis a.d. turi tam tikrą skirstinį, priklausantį nuo vieno ar kelių nežinomų parametru. Parametru įverčiams rasti naudojami keli metodai - paprastas, bet nelabai tikslus momentų metodas, ir sudėtingesnis, bet tikslesnis (nes įverčių dispersijos mažesnės) didžiausio tikėtimumo (DT) metodas (= maximum likelihood estimate (MLE) (angl.)). Dažnai (pvz., binominio, Puasono ar normaliojo skirstinio atvejais) abu įverčiai sutampa, tačiau kartais išspręsti DT lygtis nėra lengva.

Priminsime apibrėžimus. Tarkime, kad stebimojo a.d. X skirstinio funkcija $F_{\Theta}(x)$ priklauso nuo nežinomo s -mačio parametro $\Theta = (\theta_1, \dots, \theta_s)$. Iš DSD žinome, kad l -tasis ėminio momentas $A_l = (1/n) \sum_{i=1}^n x_i^l$ apytiksliai lygus l -tajam populiacijos momentui $\alpha_l = \alpha_l(\theta_1, \dots, \theta_s)$. Jei a.d. X turi s momentų, tai galima sudaryti s lygčių su s nežinomaisiais sistema: $A_l = \alpha_l(\theta_1, \dots, \theta_s)$, $l = 1, \dots, s$. Jei ją galima išspręsti $\theta_1, \dots, \theta_s$ atžvilgiu, tai sprendinys $\theta_l^* = \theta_l^*(A_1, \dots, A_s) = \theta_l^*(x_1, \dots, x_n)$, $l = 1, \dots, s$, vadinamas parametro $(\theta_1, \dots, \theta_s)$ momentų metodo įverčiu.

Šis įvertis yra palyginti lengvai randamas. Sunkiau yra su geresniuotu DT įverčiu. Tarkime, kad a.d. X turi tankį $p(x; \Theta)$. Funkcija

$$l(\Theta) = l_{x_1, \dots, x_n}(\Theta) = \prod_{i=1}^n p(x_i; \Theta)$$

kaip ir jos logaritmas

$$L(\Theta) = \ln l(\Theta)$$

vadinamos DT funkcijomis¹, o visos DT procedūros tikslas yra rasti tokias $\theta_1^*, \dots, \theta_s^*$ reikšmes, su kuriomis $l(\Theta)$ arba, kas tas pat (kodėl?), $L(\Theta)$ įgyja didžiausią reikšmę. Tai atliekama standartiniu būdu: randame visas dalines išvestines, prisilyginame jas nuliui ir išsprendžiame gautąją sistemą (deja, ši sistema kartais būna gana sudėtinga).

¹ Diskrečiuoju atveju tankį keičiame tikimybėmis.

Panagrinėkime pavyzdį. Tarkime, $X \sim N(a, \sigma)$ yra normalusis a.d. su dviem nežinomais parametrais a ir σ . Žinome, kad $EX = a$, o $\sqrt{DX} = \sigma$. Momentų metodas siūlo populiacijos momentus sulyginti su ėminio momentais: $a = \bar{x}$, $\sigma = s_1$ (čia s_1^2 yra nepaslinktas dispersijos įvertis: $s_1^2 = \sum_{i=1}^n (x_i - \bar{x})^2 / (n-1)$). Šios sistemos sprendinys akivaizdus: $a^* = \bar{x}$, o $\sigma^* = s_1$. Priminsime (žr. [Ku, 287 p.]), kad DT metodas siūlo praktiškai tuos pačius įverčius: $a^* = \bar{x}$, o $\sigma^* = s$ (čia s yra kiek paslinktas standarto įvertis: $s^2 = \sum_{i=1}^n (x_i - \bar{x})^2 / n$).

Momentų ir DT metodų įverčiai sutampa ne visuomet. Tarkime, X turi tolygų skirstinį intervale $[0, \theta]$; čia θ yra nežinomas dešinysis intervalo galas. Momentų metodas siūlo tokį nežinomo parametro įvertį (žr. [ČM, 1.6 sk.]): $\theta_1^* = 2\bar{x}$, o didžiausio tikėtino – kitokį: $\theta_2^* = \max_i x_i$. Įvertis θ_1^* yra nepaslinktas, t.y. $E\theta_1^* = \theta$, tačiau θ_2^* – ne: $E\theta_2^* = \theta(n/(n+1))$. Pakeiskime įvertį θ_2^* į $\theta_2^{**} = ((n+1)/n)\theta_2^*$ – dabar vidurkių prasme jie abu bus vienodai geri. Antra vertus, θ_2^{**} yra vis tik “geresnis”: jo reikšmės bus arčiau tikrosios θ reikšmės negu θ_1^* , nes $D\theta_1^* = (1/3n)\theta^2$, o $D\theta_2^{**} = (1/n(n+2))\theta^2$. Pailiustruosime šį teiginį.

```

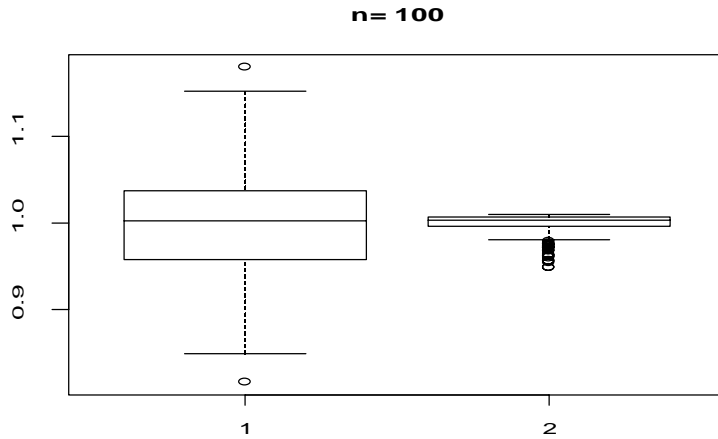
sim.unif <- function(n){                                     # n yra ėminio dydis
#funkcija sim.unif
mm <- numeric(500)
dtm <- numeric(500)
for (i in 1:500) {ru <- runif(n,0,1) # Tikroji θ reikšmė lygi 1, tačiau
# "apsimeskime", kad to nežinome

mm[i] <- 2*mean(ru)
dtm[i] <- ((n+1)/n)*max(ru)}
boxplot(mm,dtm)
title(main=paste("n=",n))
cat("E(theta1)=" ,mean(mm) , "E(theta2)=" ,mean(dtm) , "\n")
cat("D(theta1)=" ,var(mm) , "D(theta2)=" ,var(dtm) , "\n")
}

> sim.unif(10)
E(theta1)= 1.015739, E(theta2)= 0.99365 # Kadangi abu įverčiai
# nepaslinkti, E(...) turėtų
# būti maždaug 1
D(theta1)= 0.036887, D(theta2)= 0.00840 # Apskaičiuokite šias
# dispersijas pagal aukš-
# čiau pateiktas formules

> sim.unif(100)
E(theta1)= 1.000885, E(theta2)= 0.9998426
D(theta1)= 0.003496, D(theta2)= 0.0001093

```



8.1 pav. Matome, kad θ_2^{**} reikšmės žymiai arčiau 1, negu θ_1^* reikšmės; antra vertus, θ_2^{**} skirstinys aiškiai nesimetriškas (kuri uodega “sunkesnė”?)

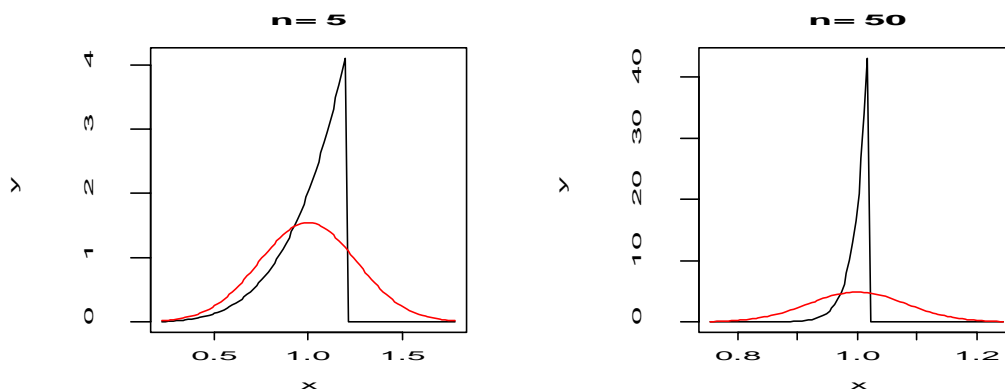
Pakomentuosime 8.1 paveikslą. Kadangi $\theta_1^* = (2/n) \sum_{i=1}^n x_i$ yra a.d. suma, tai pagal CRT šis a.d. yra maždaug normalus: $\theta_1^* \sim N(\theta, \theta/\sqrt{3n})$. Galima įrodyti, kad θ_2^{**} turi nesimetrišką tankį:

$$p_{\theta_2^{**}}(x) = \frac{n^2}{(n+1)\theta} \left(\frac{nx}{(n+1)\theta} \right)^{n-1} 1_{\left(0, \frac{n+1}{n}\theta\right)}(x).$$

Abu tankiai, didėjant n , kaupiasi apie tikrąją θ reikšmę (todėl galima vis tiksliau “atkurti” θ), tačiau antrasis tankis yra labiau sukonzentruotas apie θ ir todėl jis tai daro “geriau”.

```
> theta <- function(n){
#funkcija theta
x <- seq(1-3/sqrt(3*n), 1+3/sqrt(3*n), length=100)
y <- ifelse(x<0|x>(n+1)/n,
0, n*n*((n*x/(n+1))^(n-1))/(n+1))
plot(x,y,type="l")
lines(x,dnorm(x,1,1/sqrt(3*n)),col=2)
title(main=paste("n=",n))}
> par(mfrow=c(1,2))
> theta(5)
> theta(50)
```

Tarkime, $\theta=1$
Trijų sigmų taisyklė
Skaičiuojame antrojo
tankio reikšmes



8.2 pav. Juoda linija - θ_2^{**} tankis, raudona - θ_1^* tankis (atkreipkite dėmesį į x ir y ašis)

R paketo privalumai išaiškėja sudėtingais atvejais. Panagrinėkime gama skirstinį. Šio a.d. tankis priklauso nuo dviejų nežinomų parametrų – formos parametro a ir mastelio parametro s :

$$p(x; a, s) = \frac{1}{s^a \Gamma(a)} x^{a-1} e^{-x/s} 1_{(0, \infty)}(x), \quad a, s > 0,$$

jo vidurkis $EX = as$, o dispersija $DX = as^2$. Išsprendę šias dvi lygtis, gauname $a = (EX)^2 / DX$, o $s = DX / EX$. Kitais žodžiais, momentų metodas siūlo tokius lengvai apskaičiuojamus nežinomų parametrų įverčius: $a^* = \bar{x}^2 / s_1^2$ ir $s^* = s_1^2 / \bar{x}$. Deja, šie įverčiai nelabai tikslūs (jų dispersijos didelės). Pabandykime rasti šių parametrų DT įverčius. Tikėtinumo funkcija dabar atrodo taip

$$l(a, s) = l_{X_1, \dots, X_n}(a, s) = \prod_{i=1}^n p(X_i; a, s) = \left(\prod_{i=1}^n X_i \right)^{a-1} e^{-\sum_{i=1}^n X_i / s} s^{-na} \Gamma^{-n}(a),$$

o štai jos logaritminis variantas:

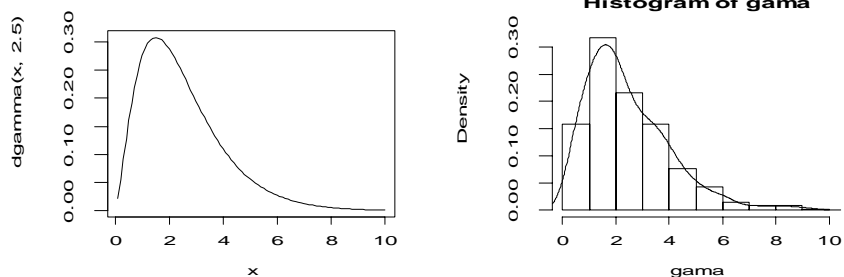
$$L(a, s) = \ln l(a, s) = (a-1) \sum_{i=1}^n \ln X_i - s^{-1} \sum_{i=1}^n X_i - na \ln s - n \ln \Gamma(a).$$

Suradę $L(a, s)$ dalines išvestines ir prisilyginę jas nuliui, gauname reikalingą sistemą

$$\begin{cases} \frac{\partial L}{\partial a} = \sum_{i=1}^n \ln X_i - n \ln s - n \Gamma'(a) / \Gamma(a) = 0, \\ \frac{\partial L}{\partial s} = s^{-2} \sum_{i=1}^n X_i - na / s = 0 \end{cases}$$

(šios sistemos šaknys ir bus ieškomi parametrų a ir s DT įverčiai). Tai komplikuota netiesinė sistema, o spręsti ją reikia iteraciniais metodais – pasirodo, kad R kalba tai galima atlikti keliomis eilutėmis:

```
par(mfrow=c(1,2))
x <- (1:100)/10
plot(x, dgamma(x, 2.5), type="l")
gama <- rgamma(500, 2.5) # Generuojame 500 gama a.d. su parametrais
# a=2,5 ir s=1
hist(gama, prob=TRUE)
lines(density(gama))
```



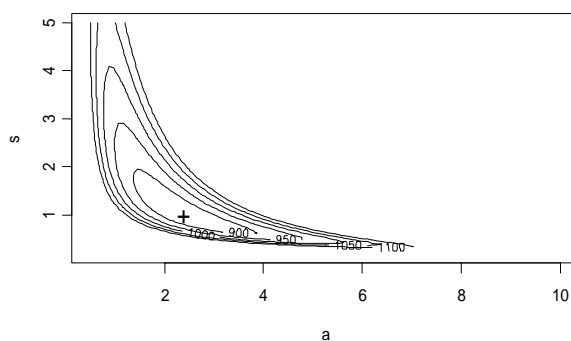
8.3 pav. Gama skirstinio tankis (kairėje) ir gama ėminio histograma ir tankio branduolinis įvertis

Dabar pagal ėminį bandysime atstatyti (atrodo, kad) gama skirstinio parametrų reikšmes.

```

mean(gama)^2/var(gama)
#[1] 2.405004
#Kadangi ėminys didelis, tai MM duoda gana tikslų parametro a įvertį
var(gama)/mean(gama)
#[1] 0.997878
#Formos parametro s įvertis irgi gana tikslus
mlgamma <- function( x)
-sum(dgamma(gama,shape= x[1],scale= x[2],log = TRUE))
# Funkcija mlgamma skaičiuoja gama skirstinio DT funkciją L(a,s)
mle <- nlm(mlgamma, c(shape = 2.4, scale = 1), hessian = TRUE)
# Minimizuojame DT funkciją, nulinė iteracija yra MM įvertis
mle$estimate
#[1] 2.519790 1.007186 a ir s įverčiai
solve( mle$hessian)
# [,1] [,2] Informacijos (atvirkštinė Heso) matrica
# [1,] 0.022550172 -0.009016688 0,022 yra a įverčio dispersija
# [2,] -0.009016688 0.004410811 0,004 yra s įverčio dispersija
avals <- seq( 0.5, 10, len = 101)
svals <- seq( 0.2, 5, len = 101)
grid <- matrix( 0.0, nrow = 101, ncol = 101)
for (i in seq( along = avals)){
  for (j in seq( along = svals)){
    grid[ i, j] <- mlgamma( c( avals[ i], svals[ j]))}
# grid yra 101x101 DT funkcijos reikšmių minimumo aplinkoje matrica
min(grid)
# 871.1726 Minimali mlgamma reikšmė
par(mfrow=c(1,1))
contour(avals, svals, grid, levels = seq(900,1100, 50))
# Išbrėžėme DT funkcijos kontūrini grafiką minimumo aplinkoje
points( mle$estimate[1], mle$estimate[2], pch = "+", cex = 1.5)
title( xlab = "a", ylab = "s")

```



8.4 pav. Gama skirstinio tikėtinumo funkcijos lygio linijos

Matome, kad tikėtinumo funkcijos lygio linijos yra mažai panašios į elipses, todėl informacijos matricos naudojimas parametrų dvimatei pasiklivimo sričiai skaičiuoti būtų nepagrįstas. Vis tik įrodysime, kad DT įverčių dispersijos yra mažesnės už MM įverčių dispersijas. Tam parašysime funkciją `disp`:

```

> disp <- function(N){
  aMM <- sMM <- aDT <- sDT <- numeric(N)
  set.seed(3)
  for(i in 1:N){
    gama <- rgamma(500,2.5)

```

```

v <- var(gama)
m <- mean(gama)
aMM[i] <- m^2/v
sMM[i] <- v/m
mlgamma <- function( x)
{-sum( dgamma( gama, shape= x[1], scale= x[2], log = TRUE))}
mle <- nlm(mlgamma, c(shape = aMM[i], scale = sMM[i]))
aDT[i] <- mle$estimate[1]
sDT[i] <- mle$estimate[2]}
cat("dispersija aMM=",var(aMM)," , dispersija sMM=",var(sMM),"\\n")
cat("dispersija aDT=",var(aDT)," , dispersija sDT=",var(sDT),"\\n")
}

```

```

> disp(1000)
dispersija aMM= 0.03353019 , dispersija sMM= 0.00625129
dispersija aDT= 0.02135913 , dispersija sDT= 0.004266431

```

DT metodo dispersija iš tikro mažesnė, tačiau tik maždaug 1,5 karto.

8.1 UŽDUOTIS. Šio skyrelio pradžioje yra pateikti normaliojo skirstinio parametrų a ir σ DT metodo įverčių formulės. Pakartokite tik ką aprašytą procedūrą ir šiuos įverčius raskite skaitmeniškai. Ar jie sutampa su aukščiau minėtais?

8.2 UŽDUOTIS. Gaminių patikimumas dažnai aprašomas Veibulo (Weibull) skirstiniu, kurio tankis atrodo taip:

$$p(x; a, s) = \frac{a}{s} \left(\frac{x}{s}\right)^{a-1} \exp\left(-\left(\frac{x}{s}\right)^a\right) \mathbb{1}_{(0, \infty)}(x), \quad a, s > 0.$$

Nesunku įsitikinti, kad $EX = s\Gamma\left(1 + \frac{1}{a}\right)$, o $DX = s^2\left(\Gamma\left(1 + \frac{2}{a}\right) - \Gamma\left(1 + \frac{1}{a}\right)^2\right)$.

- i) Išbrėžkite kelis Veibulo tankio grafikus,
- ii) Pakartokite aukščiau aprašytą analizę su Veibulo skirstiniu.

8.3 UŽDUOTIS. Pakete MASS yra funkcija `fitdistr`, kuri skaičiuoja daugelio (įskaitant gama ir Veibulo) skirstinių parametrų DT įverčius. Generuokite 200 gama atsitiktinių skaičių, įvertinkite skirstinio parametrus aukščiau aprašytu metodu ir naudodami `fitdistr` funkciją. Palyginkite rezultatus.

8.4 UŽDUOTIS. Mes jau nagrinėjome duomenų rinkinį `Davis` ir jo “pataisytą” variantą `davis` (tai duomenys apie reguliariai sportu užsiiminėjančių asmenų svorį ir ūgį):

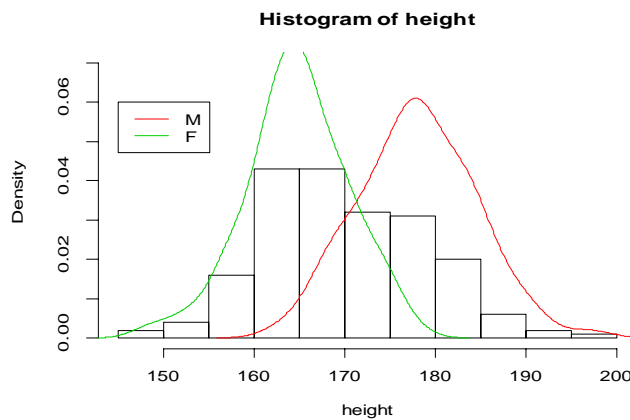
```

> summary(davis)
sex      weight      height      repwt      repht
F:112   Min.    : 39.00   Min.    :148.0   Min.    : 41.00   Min.    :148.0
M: 88   1st Qu.: 55.00   1st Qu.:164.0   1st Qu.: 55.00   1st Qu.:160.5
        Median : 63.00   Median :169.5   Median : 63.00   Median :168.0
        Mean   : 65.25   Mean   :170.6   Mean   : 65.62   Mean   :168.5
        3rd Qu.: 73.25   3rd Qu.:177.3   3rd Qu.: 73.50   3rd Qu.:175.0
        Max.   :119.00   Max.   :197.0   Max.   :124.00   Max.   :200.0
        NA's   : 17.00   NA's   : 17.00

```

Aišku, kad kintamasis `height` neturėtų būti normalus (nes šis ėminys yra dviejų (tikriausiai) normaliųjų a.d. (būtent, vyrų ir moterų ūgių) mišinys, žr. 8.5 pav.; pakartokite šį

paveikslą – jame raudona ir žalia kreivės išbrėžtos su density funkcija). Beje, tikrinant normalumo hipotezę su shapiro.test funkcija (plg. 9-9 psl.), normalumo hipotezė nėra atmetama (nes p reikšmė didesnė už 0,05):



8.4 pav. Kintamojo davis\$height histograma ir vyrų bei moterų ūgio tankių branduolinių įverčių grafikai

```
> shapiro.test(height)

Shapiro-Wilk normality test
```

```
data: height
W = 0.9898, p-value = 0.1697
```

Taigi, formaliai žiūrint, kintamasis height yra normalusis, nors taip tikriausiai nėra. Dabar užmirškime, ką mes žinome apie šį kintamąjį ir pabandykime atstatyti jo struktūrą, remdamiesi prielaida, kad height yra dviejų ėminių (su nežinoma proporcija p) mišinys, o kiekviena mišinio komponentė yra normali su nežinomais parametrais, t.y., height tankis yra pavidalo

```
mixt <- function(x,p,mean1,sd1,mean2,sd2)
{p*dnorm(x,mean1,sd1)+(1-p)*dnorm(x,mean2,sd2)}
```

(p turėtų būti lygus moterų daliai ėminyje, t.y., $112/(112+88)=0,56$, mean1 maždaug lygus $\text{mean}(\text{height}[\text{sex}=="F"])=164.7$ ir t.t.). Šį uždavinį išspręskite remdamiesi didžiausio tikėtimumo principu ir taikydami funkciją fitdistr. Gal būt jums pagelbės atsakymas:

	p	mean1	sd1	mean2	sd2
	0.5987635	165.2547700	5.9410993	178.4856961	6.3547038
	(0.1899365)	(2.0607014)	(0.9848752)	(3.3615904)	(1.5020201)

8.5 UŽDUOTIS. Stebimasis a.d. X turi normalųjį skirstinį su nežinomais a ir σ^2 . Generuokite “daug” normaliųjų a.d. ir “įrodykite”, kad 1) s_1 yra paslinktas standarto σ įvertis; 2) statistikos

$$\sigma_1^* = \left(\frac{n-1}{2}\right)^{1/2} \frac{\Gamma((n-1)/2)}{\Gamma(n/2)} s_1 \quad \text{ir} \quad \sigma_2^* = \frac{\sqrt{\pi}}{\sqrt{2n(n-1)}} \sum_{k=1}^n |x_k - \bar{x}|$$

yra nepaslinkti standarto įverčiai, bet σ_1^* geresnis (nes $D\sigma_1^* < D\sigma_2^*$). Nuoroda. Lengviau dirbti, kai n nėra didelis (pvz., $n=4$). Monte Carlo replikų skaičių N imkite lygiu 10000.

8.6 UŽDUOTIS. Tarkime, kad $x_1^*, x_2^*, \dots, x_n^*$ ($x_1^* \leq x_2^* \leq \dots \leq x_n^*$) yra variacinė seka, gauta iš imties x_1, x_2, \dots, x_n , o $\theta_1^* = \bar{x}$, $\theta_2^* = (x_1^* + x_n^*)/2$. Ištirkite $E\theta_k^*$ ir $D\theta_k^*$ ($k=1,2$) savybes tuomet, kai a) X turi $(0,2\theta)$ tolygų skirstinį ir b) X yra eksponentinis a.d. su tankiu $p(x) = (1/\theta)\exp(-x/\theta)1_{(0,\infty)(x)}$ (abiems atvejais parametras θ nežinomas).

8.7 UŽDUOTIS. Nežinomo parametro θ įvertis $\theta^*(n)$ vadinamas suderintu, jei $\theta^*(n)$, kai $n \rightarrow \infty$, konverguoja pagal tikimybę į θ (t.y., jei $P_\theta\{|\theta^*(n) - \theta| \geq \varepsilon\} \rightarrow 0$ koks bebūtų $\varepsilon > 0$). Tarkime, kad a.d. X turi normalųjį skirstinį su nežinomu vidurkiu θ . “Įrodykite”, kad empirinis vidurkis $\theta^*(n) = \bar{x}$ yra suderintas θ įvertis. Nuoroda. Tarkime, $\varepsilon = 0.25$. Generuokite $N=10000$ normaliojo a.d. $N(2,1)$ dydžio $n=10$ ėminių ir apskaičiuokite kiekvieno jų vidurkį. Apskaičiuokite santykinį dažnį $n(|\bar{x} - 2| > 0.25)/n$. Pakartokite šią procedūrą, kai $n=100$ ir kai $n=1000$. Įsitikinkite, kad gautieji santykiniai dažniai artėja į nulį.

8.8 UŽDUOTIS. Nežinomo parametro θ įvertis $\theta^*(n)$ vadinamas suderintu, jei $\theta^*(n)$, kai $n \rightarrow \infty$, konverguoja pagal tikimybę į θ (t.y., jei $P_\theta\{|\theta^*(n) - \theta| \geq \varepsilon\} \rightarrow 0$ koks bebūtų $\varepsilon > 0$). Tarkime, kad a.d. X turi apibendrintą Koši skirstinį su tankiu $p_\theta(x) = 1/(\pi(1+(x-\theta)^2))$ (taškas θ yra tankio simetrijos centras arba “vidutinė” reikšmė; priminsime, kad a.d. X vidurkis neegzistuoja). “Įrodykite”, kad a) empirinis vidurkis $\theta^*(n) = \bar{x}$ yra nesuderintas θ įvertis, tačiau b) empirinė mediana yra suderintas (populiacijos medianos) θ įvertis. Nuoroda. Perskaitykite 8.7 užduoties nuorodą.

8.2. Intervaliniai įverčiai

Praėjusiame skyrelyje buvo paaiškinta, kaip galima rasti nežinomo parametro θ įverčius θ^* . Būtų gerai, jei sugebėtume nurodyti, kiek gautasis įvertis skiriasi nuo tikrosios parametro reikšmės, t.y. jeigu sugebėtume rasti $\varepsilon (> 0)$ tokį, kad $|\theta^* - \theta| \leq \varepsilon$. Deja, kadangi θ^* yra a.d., tai galima kalbėti tik apie įvykio $\{|\theta^* - \theta| \leq \varepsilon\}$ tikimybę. Tarkime, kad egzistuoja dvi statistikos², θ_1^* ir θ_2^* , $\theta_1^* < \theta_2^*$, tokios, kad $P_\theta(\theta_1^* \leq \theta \leq \theta_2^*) = \alpha$. Jei α mažai skiriasi nuo 1, tai galima laikyti, kad praktiškai visuomet $\theta_1^* \leq \theta \leq \theta_2^*$. Atsitiktinis intervalas $[\theta_1^*, \theta_2^*]$ yra vadinamas parametro θ pasikliauties intervalu, atitinkančiu pasikliauties tikimybę (ar lygmenį) α . Paprastai imama $\alpha = 0,9; 0,95; 0,99$ ir pan., o statis-

² Apibrėžimas. Bet kokia ėminio funkcija θ vadinama statistika. Štai keli statistikų pavyzdžiai: $\theta_1 = \bar{x}$, $\theta_2 = \max_k x_k$.

tikos θ_1^* ir θ_2^* parenkamos taip, kad duotai tikimybei α intervalas $[\theta_1^*, \theta_2^*]$ būtų kuo trumpesnis.

Pradėkime vienu pavyzdžiu. Iš DSD žinome, kad, didėjant ėminio dydžiui, jo vidurkis konverguoja į populiacijos vidurkį. Antra vertus, nedideliems n šie du skaičiai gali pastebimai skirtis. Pamodeliuokime šį reiškinį – kelis kartus generuokime po 10 $N(0,1)$ atsitiktinių skaičių ir apskaičiuokime jų empirinius vidurkius ir standartus:

```
> set.seed(1) # Dabar rezultatai bus reprodukuojami
> rn <- rnorm(10)
> rn
[1] -0.9746257 -0.5607552 -2.1839227 0.0895115 0.9695195
[6] -0.4447884 -0.0228788 0.0996069 -0.3551539 0.5429091
> mean(rn)
[1] -0.2840578 # Ėminio vidurkis (jis turėtų būti maždaug 0)
> sd(rn)
[1] 0.8694902 # Ėminio standartas (jis turėtų būti maždaug 1)
> rn <- rnorm(10) # Dar kartą generuokime 10 normalių skaičių
> rn
[1] -1.9304081 -0.9490871 -1.5298516 0.8682471 2.0663557
[6] 0.5861246 -1.2559018 -0.6659785 2.8956619 1.0220281
> mean(rn)
[1] 0.1107190 # Antrojo ėminio vidurkis arčiau 0 nei pirmojo
> sd(rn)
[1] 1.621472 # Antrojo ėminio standartas toliau nuo 1
```

Matome, kad ėminio rn vidurkis (kaip ir standartas) gana pastebimai kinta, kitaip sakant, nelogiška tvirtinti, kad “nežinomas” populiacijos vidurkis $a = -0,284$ ir, kartu, $a = 0,111$. Teisinga būtų teigti, kad $a \approx -0,284$ ir $a \approx 0,111$, tačiau gerai būtų, jei kartu galėtume nurodyti ir daromą paklaidą. Elgsimės taip: tarkime, kad populiacijos dispersija σ^2 yra žinoma; tuomet iš CRT gauname, kad a.d. $Z = \sqrt{n}(a - \bar{x})/\sigma$ turi maždaug standartinę normalųjį³ skirstinį. Jei simboliu $z(\alpha)$ pažymėtume standartinio normaliojo skirstinio α -ąjį kvantilį, $z_1 = z(\alpha_1)$, $z_2 = z(\alpha + \alpha_1)$, $0 \leq \alpha_1 \leq 1 - \alpha$, tai tuomet

$$P(z_1 \leq \sqrt{n}(a - \bar{x})/\sigma \leq z_2) \approx P(z_1 \leq N \leq z_2) = \alpha (\approx 0,95).$$

Kitaip sakant,

$$P\left(\bar{x} + \frac{z_1\sigma}{\sqrt{n}} \leq a \leq \bar{x} + \frac{z_2\sigma}{\sqrt{n}}\right) \approx \alpha,$$

t.y. $[\bar{x} + z_1\sigma/\sqrt{n}, \bar{x} + z_2\sigma/\sqrt{n}]$ yra nežinomo vidurkio a lygmens α pasikliauties intervalas. Galima įrodyti, kad šis intervalas trumpiausias, kai $\alpha_1 = (1 - \alpha)/2$, $\alpha_2 = (1 + \alpha)/2$, t.y. kai abiejų “uodegų” tikimybės lygios. Jei pasirinktume populiariausią reikšmę $\alpha = 0,95$, tai tuomet $z_1 = -1,96$, $z_2 = 1,96$, o (nežinomo) vidurkio a pasikliauties intervalas lygus $[\bar{x} - 1,96\sigma/\sqrt{n}, \bar{x} + 1,96\sigma/\sqrt{n}]$. Didinant n , intervalas siaurėja (vidurkis a bus nustatomas vis tiksliau), bet lėtai – kaip $const/\sqrt{n}$ (norint intervalą susiaurinti dvigubai, ėminio dydį reikia padidinti keturis kartus!).

³ Mūsų pavyzdyje ėmėme iš normaliosios populiacijos, todėl ši statistika turi tiksliai normalųjį skirstinį.

Aptartasis pasikliaudies intervalas vadinamas centriniu. Kartais naudingi kitokie intervalai: pasirinkę $z_1 = z(0) = -\infty$ ir $z_2 = z(\alpha)$, gautume vadinamąjį apatinį⁴ (arba kairinį), o pasirinkę $z_1 = z(1-\alpha)$ ir $z_2 = z(1) = +\infty$ - viršutinį⁵ (arba dešinį) pasikliaudies intervalus. Nors visų jų pasikliaudies lygmuo lygus α , trumpiausias yra centrinis intervalas.

Grįžkime prie mūsų pavyzdžio. Kadangi $\sigma = 1$, $n = 10$, tai abiem atvejais “tikrasis” vidurkis 0 priklauso intervalams $[-0,284 - 1,96 \cdot 1/\sqrt{10}; -0,284 + 1,96 \cdot 1/\sqrt{10}] = [-0,904; 0,336]$ ir $[0,111 - 1,96 \cdot 1/\sqrt{10}; 0,111 + 1,96 \cdot 1/\sqrt{10}] = [-0,509; 0,731]$.

Tiriant realius duomenys, populiacijos dispersija σ^2 žinoma retai. Tokiais atvejais, logiška vietoje nežinomo σ^2 imti jo nepaslinktą įvertį s_1^2 . Galima įrodyti, kad dabar a.d. $T_{n-1} = \sqrt{n}(a - \bar{x})/s_1$ turi jau ne (beveik) normalų, bet (beveik⁶) Stjudento skirstinį su $n-1$ l.l. Trumpiausias lygmens α pasikliaudies intervalas yra pavidalo $[\bar{x} + t_{n-1}((1-\alpha)/2) \cdot s_1/\sqrt{n}, \bar{x} + t_{n-1}((1+\alpha)/2) \cdot s_1/\sqrt{n}]$ (čia $t_n(\alpha)$ yra Stjudento su $n-1$ l.l. α eilės kvantilis). Priminsime, kad $t_{n-1}(\alpha) \rightarrow z(\alpha)$, $n \rightarrow \infty$, bet nedideliems n Stjudento kvantilių moduliai yra didesni už normaliųjų kvantilių modulius (taigi dabar pasikliaudies intervalai kiek platesni).

Grįžkime prie savo pavyzdžio. Tare, kad dabar mes nežinome ne tik vidurkio, bet ir dispersijos, pirmuoju atveju randame

```
> -0.284+qt((1-0.95)/2,9)*0.869/sqrt(10)
[1] -0.9056451
> -0.284+qt((1+0.95)/2,9)*0.869/sqrt(10)
[1] 0.3376451
```

(t.y., “nežinomo vidurkio” pasikliaudies intervalas dabar kiek platesnis: pirmuoju atveju jis lygus $[-0,906; 0,338]$), o antruoju – $[-1,049; 1,271]$).

R neturi funkcijos vidurkio pasikliaudies intervalui skaičiuoti. Šią funkciją galima parašyti pačiam (tai kelių eilučių funkcija, parašykite ją!), arba pasinaudoti funkcija `t.test`:

```
> t.test(rn)$conf.int
[1] -1.049212 1.270650
attr(,"conf.level")
[1] 0.95 # Skaičiuojame 95% pasikliaudies intervalą
```

(rezultatas nežymiai skiriasi nuo mūsų skaičiavimo, nes mes apvalinome vidurkio ir standarto reikšmes).

Padarysime dar vieną pastabą. Mes netvirtiname, kad nežinomas vidurkis visuomet yra pasikliaudies intervalo viduje – jei pasikliaudies (pasitikėjimo) tikimybė $\alpha = 0,95$, tai

⁴ Jei $\alpha=0,95$, tai $z_2=1,64$.

⁵ Jei $\alpha=0,95$, tai, kadangi standartinio normaliojo tankis yra lyginė funkcija, $z_1=-1,64$.

⁶ Jei stebime normalųjį a.d. X , tai statistika T_{n-1} turi tiksliai Stjudento skirstinį. Pažymėsime, kad net tuomet, kai stebimasis a.d. “nelabai normalus”, T_{n-1} skirstinys mažai nutolsta nuo Stjudento.

mes tvirtiname tik tiek, kad pasikliauties intervalas (maždaug) 95 procentams ėminių uždengs vidurkį (taigi kartais galime suklysti, bet tai bus retai). Patikrinsime savo teiginį.

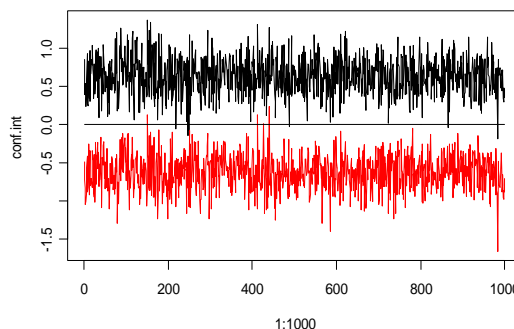
```
> conf <- function(){
# funkcija conf
# eminio dydis=20
# alpha=0.99 # Imame "nestandartinį" reikšmingumo lygm.
set.seed(5)
t.lo <- qt(0.005,19)
t.up <- qt(0.995,19)
coef.lo <- t.lo/sqrt(20)
coef.up <- t.up/sqrt(20)
up <- numeric(1000)
lo <- numeric(1000)
sk <- 0 # "Teisingų" atvejų (kai pasikl. intervalas
# uždengia 0) skaitiklis

for(i in 1:1000){
  rn <- rnorm(20)
  m <- mean(rn)
  st <- sd(rn)
  lo[i] <- m+coef.lo*st # Pasikliauties intervalo apatinis rėžis
  up[i] <- m+coef.up*st # Pasikliauties intervalo viršutinis rėžis
  sk <- sk+ifelse(lo[i]>0|up[i]<0,0,1)} # Jei pasikl. intervalas
# "teisingas"-pridedame 1

plot(1:1000,up,type="l", # Viršutinis rėžis (juoda spalva)
ylim=c(min(lo),max(up)),ylab="conf.int")
lines(1:1000,lo,col=2) # Apatinis rėžis (raudona spalva)
lines(1:1000,rep(0,1000))
cat("Daznis=",sk/1000,"\n")
}

> conf()
Daznis= 0.99
```

Stebime idealų atitikimą – teoriškai turėtu būti 0,99, gavome 0,99 (plg. 8.6 pav.).



8.6 pav. Tik dešimt kartų iš tūkstančio pasiklovimo intervalas neuždengia 0 (ar galite nurodyti tuos atvejus?)

Aukščiau pateiktus samprotavimus galima pakartoti dispersijoms ir išvesti, kad tuomet, kai stebime normalųjį⁷ a.d. X su nežinomais vidurkiu ir dispersija, dispersijos σ^2 lygmens α pasikliauties intervalas atrodo taip:

$$[(n-1)s_1^2 / \chi_{n-1}^2((1+\alpha)/2), (n-1)s_1^2 / \chi_{n-1}^2((1-\alpha)/2)]$$

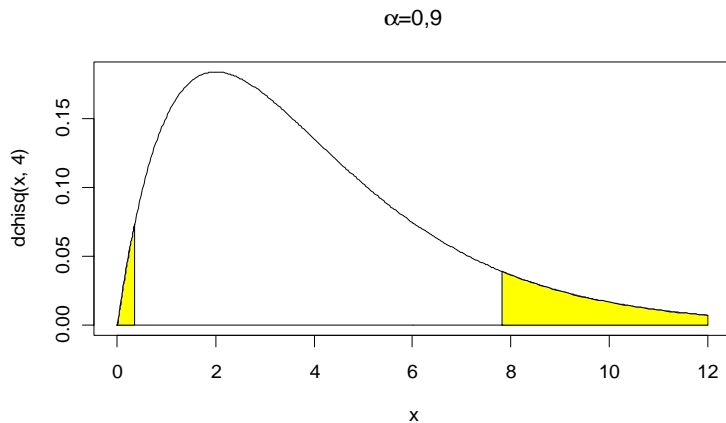
(čia $\chi_n^2(\alpha)$ yra chi kvadrato su n l.l. α eilės kvantilis).

⁷ Stjudento statistika nelabai jautri nuokrypiams nuo normalumo, tačiau su χ^2 procedūra reikia elgtis atsargiau. Pažymėsime, kad nenormaliuoju atveju dispersija apskritai nėra geras reikšmių sklaidos matas (pvz., gali negalioji 3σ taisyklė), todėl tokiu atveju sklaidai matuoti reikėtų vartoti kitas charakteristikas (pvz., tarpkvartilinį atstumą *iqd*).

Štai funkcija, kuri pateikia geometrinę dispersijos pasikliauties intervalo interpretaciją:

```
> chi <- function(){
# funkcija chi
x <- seq(0,12,length=400)
plot(x,dchisq(x,4),type="l",
main=expression(paste(alpha,"=0,9")))
lines(x,rep(0,400))
xx.lo <- seq(0,qchisq(0.05,3),length=50) # Reikšmingumo lygmuo  $\alpha=0,9$ 
XX.lo <- c(xx.lo,-sort(-xx.lo),0)
YY.lo <- c(rep(0,50),dchisq(-sort(-xx.lo),4),0)
polygon(XX.lo,YY.lo,col="yellow") # Nuspalvinsime kairiąją uodegą
# (iki 0,05-jo kvantilio))

xx.up <- seq(qchisq(0.95,3),12,length=50)
XX.up <- c(xx.up,-sort(-xx.up),xx.up[1])
YY.up <- c(rep(0,50),dchisq(-sort(-xx.up),4),0)
polygon(XX.up,YY.up,col="yellow") # Nuspalvinsime dešinę uodegą;
# jos plotas irgi lygus  $(1-0,9)/2$ 
}
```



8.7 pav. Galima įrodyti, kad $(n-1)s_1^2/\sigma^2$ turi χ_{n-1}^2 skirstinį; todėl

$$P(\chi_{n-1}^2((1-\alpha)/2) < (n-1)s_1^2/\sigma^2 < \chi_{n-1}^2((1+\alpha)/2)) = \alpha$$

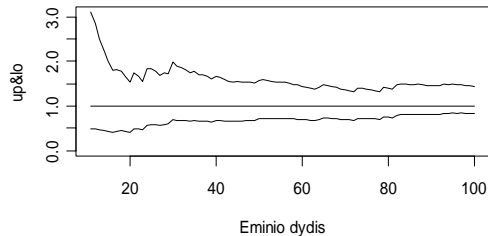
R neturi funkcijos dispersijos pasikliauties intervalui skaičiuoti, ją parašysime patys.

```
> conf.var <- function(x,conf.level=0.95){ # Standartine  $\alpha$  reikšme
# imame 0,95
# funkcija conf.var
n <- length(x)
s1 <- var(x)
s1*(n-1)/c( qchisq((1+conf.level)/2,n-1),qchisq((1-conf.level)/2,n-1))
}

> conf.var(rn)
[1] 0.4012191 1.4799255
```

Matome, kad “tikroji” dispersijos reikšmė (priminsime: $\sigma^2 = 1$) priklauso pasikliauties intervalui (jei tokį eksperimentą pakartosime daug kartų, tai tik maždaug penkis kartus iš šimto suklysimė, teigdami, kad gautasis intervalas uždengia populiacijos dispersiją).

8.9 UŽDUOTIS. Dispersijos pasikliauties intervalas siaurėja, kai n didėja, tačiau iš jo išraiškos to nesimato. Parašykite funkciją, kuri išbrėžtų tai iliustruojantį grafiką (maždaug tokį kaip 8.8 pav.).



8.8 pav. Didėjant ėminio dydžiui, dispersijos pasikliauties intervalas (nemonotoniškai) siaurėja

Dabar aptarkime binominę atvejį. Paskutiniuose rinkimuose už Laimės žiburio partiją balsavo 13,2% rinkėjų. Po metų buvo atlikta sociologinė apklausa, kurioje iš 1000 apklaustųjų 170 pareiškė, kad, jei rinkimai būtų rytoj, jie balsuotų už šią partiją. Ar suderinami šie duomenys su LŽP oponentų teigimu, kad “šitai dar nieko nereiškia”? Oponentų teiginį galima iššifruoti taip: šios partijos gerbėjų procentas nepasikeitė, o padidėjimą galima paaiškinti ėminio atsitiktinumu. Formalizuokime jų teiginį. Tarkime, kad p yra tikimybė, kad jog rinkėjas rytoj balsuotų už LŽP, t.y., $p = (\text{kiek bus "už"}) / (\text{turi balso teisę}) = 0,13^8$. Simboliu p^* pažymėkime šios tikimybės empirinį ekvivalentą: $p^* = (\text{kiek "už" buvo tarp apklaustųjų}) / 1000 = 0,17$. Tikslūs (dabartinio) p pasikliauties intervalo rėžiai užrašomi gana komplikuoti (žr. [Kr, 193 p., (6.94)]), tačiau, naudojant R paketą, to nesijaučia:

```
> binom.test(170,1000,p=0.132)

Exact binomial test

data: 170 and 1000
number of successes = 170, number of trials = 1000, p-value = 0.000633
alternative hypothesis: true probability of success is not equal to
0.132
95 percent confidence interval:
0.1472170 0.1947441 # Proporcijos p centrinis pasikliauties intervalas
sample estimates:
probability of success
0.17
```

Taigi, centrinis nežinomos proporcijos p pasikliauties intervalas $[0,147; 0,195]$ neužden-
gia $0,13$, todėl apklausos duomenys paneigia oponentų teiginį – LŽP gerbėjų dalis per
metus padidėjo ir su tikimybe $0,95$ yra nurodytame intervale. Beje, kadangi LŽP gerbėjų
skaičius padidėjo, tai jie mielai pasirinktų kitą, viršutinį⁹, pasikliauties intervalą, kuris
pateiktų jiems dar palankesnę prognozės rezultatą:

```
> binom.test(170,1000,p=0.132,alt="g")$conf.int # "g"=greater
```

⁸ Kitais žodžiais, p yra LŽP gerbėjų dalis visoje populiacijoje.

⁹ Objektas `binom.test(170,1000,p=0.132,alt="g")` yra sąrašas. Norint sužinoti visas jo kom-
ponentes, reikia surinkti `names(binom.test(170,1000,p=0.132,alt="g"))` (mums reikia
komponentės "conf.int").

```
[1] 0.1507 1.0000 # Viršutinis 95% pasikliauties intervalas
attr(,"conf.level")
[1] 0.95
```

Antra vertus, LŽP oponentams labiau patiktų mažos p reikšmės, todėl jie mielai paskelbtų apatinį intervalą:

```
> binom.test(170,1000,p=0.132,alt="l")$conf.int # "l"=less
[1] 0.0000000 0.1907543 # Apatinis 95% pasikliauties intervalas
attr(,"conf.level")
[1] 0.95
```

Matome, kad dabar apklausos rezultatai neprieštarauja oponentų teiginiui, kad p liko toks koks buvo (ar net pasidarė visai mažas; teisybė, jis neprieštarauja ir tam, kad dabar tikimybė p lygi, pvz., 0,19). Taigi pasikliauties intervalas nenurodo tikslios p reikšmės¹⁰, jis tik duoda jos reikšmes, kurios (su 95% tikimybe) yra suderinamos su turimais duomenimis (pasikliauties intervalas duoda intervalinį p įvertį, tikslią p reikšmę galėtų pateikti tik rytoj įvyksiantys rinkimai).

Iki šiol nagrinėjome tikslų nežinomos tikimybės p pasikliauties intervalą. Apytikslį intervalą nesunku gauti iš CRT. Sakysime, kad $X_i = 1$, jei i -asis apklausos dalyvis atsakė, kad jis balsuotų už LŽP (to tikimybė p) ir $X_i = 0$ - priešingu atveju (tikimybė $q = 1 - p$). Reiškiny $p^* = S_n/n$ turi maždaug normalų skirstinį $N(p, \sqrt{pq/n})$, todėl su tikimybe 0,95

$$p - z\sqrt{pq/n} \leq p^* \leq p + z\sqrt{pq/n}$$

(čia $z = 1,96$). Išsprendę šią nelygybę p atžvilgiu, gauname nežinomos tikimybės p centrinį 95% pasiklovimo intervalą

$$\frac{n}{n+z^2} \left(p^* - \frac{z^2}{2n} - z\sqrt{\frac{p^*(1-p^*)}{n} - \frac{z^2}{4n^2}} \right) \leq p \leq \frac{n}{n+z^2} \left(p^* - \frac{z^2}{2n} + z\sqrt{\frac{p^*(1-p^*)}{n} - \frac{z^2}{4n^2}} \right).$$

Šia formule remiasi funkcija `prop.test`:

```
> prop.test(170,1000)$conf.int
[1] 0.1475206 0.1950591
attr(,"conf.level")
[1] 0.95
```

arba

```
> prop.test(170,1000,alt="less")$conf.int
[1] 0.0000000 0.1909442
attr(,"conf.level")
[1] 0.95
```

Kadangi ėminys didelis, šie intervalai tik nežymiai skiriasi nuo tikslų.

¹⁰ Negana to, jis dar priklauso nuo "užsakovo". Sekančiame skyrelyje kalbėsime apie hipotezių tikrinimą ir ten aptarsime intervalo pasirinkimo principus.

8.10 UŽDUOTIS. Išsinagrinėkite funkciją `prop.test` ir jos tekste raskite nurodytą formulę.

8.11 UŽDUOTIS. Pakete MASS yra funkcija `fitdistr`, kuri pateikia populiacijos parametrų DT įverčius. Pavyzdžiui,

```
> library(mass)
> x <- rnorm(30, 4, 1.5)
> fitdistr(x, "normal", list(mean=0, sd=1))
      mean      sd
4.1058149  1.7049985
(0.3112887) (0.2201328)
```

(skliausteliuose yra parametrų įverčių standartinių paklaidų įverčiai). O dabar UŽDUOTIS. Raskite ėminio `ga` iš

```
> set.seed(1)
> ga <- rgamma(500, shape=2.5, rate=0.1)
```

parametrų DT įverčius su a) `fitdistr` funkcija ir b) kartodami procedūrą iš 8-5 psl.

8.12 UŽDUOTIS. Su

```
library(Simple)
data(exec.pay)
?exec.pay
```

pakraukite duomenų rinkinį `exec.pay` (tai JAV vadovaujančių darbuotojų (CEO = Chief Executive Officer (angl.)) atlyginimai). Raskite 90%, 95% ir 99% vidurkio ir medianos pasikliauties intervalus. “Gražiai” juos atspausdinkite. Paieškokite “informatyvių” šio kintamojo grafikų. *Nuoroda.* Kadangi skirstinys nesimetriškas, šią užduotį geriausiai atlikti butstrepe metodu (plg. 4.13 ir 4.22 užduotis).

8.13 UŽDUOTIS. Tais atvejais, kai populiacijos skirstinys priklauso nuo kelių parametrų, didžiausio tikėtimumo įverčių radimas kartais būna gana kompliktuotas (plg. 8-4 – 8-6 psl.). Vienmačiu atveju dažnai viskas būna paprasčiau. Pvz., Puasono skirstinio su nežinomu parametru $\lambda > 0$ tikėtimumo funkcija atrodo taip (kodėl?):

$$l(\lambda) = e^{-n\lambda} \frac{\lambda^{x_1 + \dots + x_n}}{x_1! \dots x_n!}, \lambda > 0.$$

Jos išvestinę prilyginę 0, matome, kad maksimumą ši funkcija pasiekia taške $\hat{\lambda} = (x_1 + \dots + x_n) / n = \bar{x}$ - tai ir bus nežinomo parametro λ didžiausio tikėtimumo įvertis (jis, beje, sutampa su momentų metodu gautuoju (kodėl?)).

Žemiau pateiktas funkcijos `mle.pois`, brėžiančios Puasono skirstinio tikėtimumo funkcijos grafiką, kodas. Pažymėsime, kad nedidelių papildomų komplikacijų atsiranda dėl to, kad viename grafike norime išbrėžti dvi kreives (tikėtimumo funkcijos ir jos logaritmo) su dviem y ašimis, grafiko kairėje ir dešinėje (šios komandos pažymėtos mėlsva spalva).

```

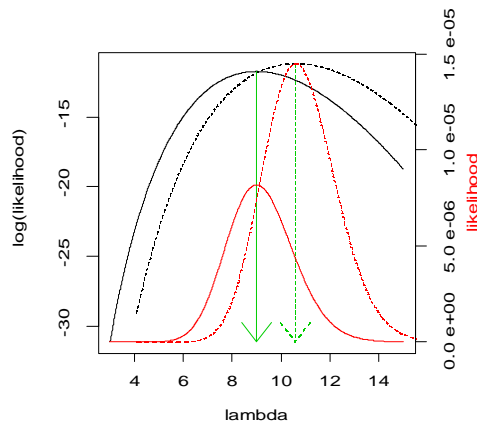
> mle.pois <- function(){
#Puasono skirstinio tikėtinumo funkcija
x_rpois(5,10) # Generuojame penkis Puasono su  $\lambda=10$  ats. skaičius
lhat_mean(x)
lambda_seq(lhat-2*sqrt(lhat),lhat+2*sqrt(lhat),by=0.01)#Teig.parametro
lambda_lambda[lambda>0] #  $\lambda$  kitimo režius nustatome pagal  $2\sigma$  taisyklę
likelihood_exp(-lambda*length(x))*lambda^sum(x)/prod(gamma(x+1))
x2_rpois(5,10) # Generuojame dar penkis Puasono su  $\lambda=10$  ats. skaičius
lhat2_mean(x2)
lambda2_seq(lhat2-2*sqrt(lhat2),lhat2+2*sqrt(lhat2),by=0.01)
lambda2_lambda2[lambda2>0]
likelihood2_exp(-lambda2*length(x2))*lambda2^sum(x2)/prod(gamma(x2+1))
par(mar=c(5,4,4,4)) #Paruošiamė vietos dviem papildomoms linijoms deš.
yMIN.log <- min(min(log(likelihood),min(log(likelihood2))))
yMAX.log <- max(max(log(likelihood),max(log(likelihood2))))
plot(lambda,log(likelihood),type="l",ylim=c(yMIN.log,yMAX.log))
lines(lambda2,log(likelihood2),lty=2)
MAX <- which.max(likelihood)
MAX2 <- which.max(likelihood2)
arrows(lambda[MAX],log(likelihood)[MAX],lambda[MAX],yMIN,col=3)
arrows(lambda2[MAX2],log(likelihood2)[MAX2],lambda2[MAX2],yMIN,col=3,
lty=2) # Strėlės (arrows) žymės tikėt. funkcijų maksimumus
par(new=T) # Kita plot komanda neištrins ankstesnio grafiko
yMIN <- min(min(likelihood,min(likelihood2)))
yMAX <- max(max(likelihood),max(likelihood2))
plot(lambda,likelihood,type="l",ylim=c(yMIN,yMAX),col=2,axes=F,
xlab="",ylab="")
axis(side=4,col=2)
lines(lambda2,likelihood2,lty=2,col=2)
mtext("likelihood",side=4,line=2,col=2)
print(list(mean=lhat,mean2=lhat2))} # Spausdiname ėminių vidurkius

> mle.pois()
$mean
[1] 9.4

$mean2
[1] 11.8

```

8.9 pav. (žr. žemiau) matome, kad ir tikėtinumo funkcija (raudona spalva), ir jos logaritmas (juoda spalva) maksimumą pasiekia tame pačiame taške (jis lygus ėminio vidurkiui ir kartu su ėminiu gali kisti).



8.9 pav. Dvi Puasono tikėtinumo funkcijos (jos atitinka du ėminusius su vidurkais 9,4 ir 11,8)

O dabar **UŽDUOTIS**. Išbrėžkite grafiką su viena kreive (stebimo a.d. tikėtinumo funkcija) ir su viena y ašimi, kai stebime a.d. X su eksponentiniu skirstiniu, t.y., kai

$$p_x(x) = \begin{cases} 0, & \text{kai } x < 0, \\ \theta \exp(-\theta x), & \text{kai } x \geq 0. \end{cases}$$

(parametras $\theta > 0$ nežinomas).

9. Sprendžiamoji statistika: hipotezių tikrinimas (vienas ėminys)

Pasikliauties intervalų skaičiavimas yra glaudžiai susijęs su hipotezių tikrinimu. Pirmuoju atveju randame intervalą, kuriame turėtų būti nežinoma parametro reikšmė. Antroju atveju tariame, kad parametras turi konkrečią reikšmę ir klausiamo, ar ši prielaida (hipotezė) yra suderinama su turimais duomenimis. Vienas iš šios problemos sprendimo variantų yra toks: jei tinkamai parinktas pasikliauties intervalas uždengia hipotetinę reikšmę, tai (kiek diplomatiškai) sakome, kad turimi duomenys neprieštaruoja mūsų hipotezei. Dažniausiai ta pati R funkcija skaičiuoja ir pasikliauties intervalus ir tikrina hipotezes.

Dauguma šiame skyriuje aptariamų testų priklauso base ir/arba ctest (=classical test) paketams. Štai visų R 1.7.1 paketo ctest funkcijų sąrašas:

```
> library(help=ctest)
```

```
Information on Package 'ctest'
```

Description:

```
Package: ctest
Version: 1.7.1
Priority: base
Title: Classical Tests
Author: Kurt Hornik <Kurt.Hornik@ci.tuwien.ac.at>, with major
       contributions by Peter Dalgaard <p.dalgaard@kubism.ku.dk> and
       Torsten Hothorn <Torsten.Hothorn@rzmail.uni-erlangen.de>.
Maintainer: R Core Team <R-core@r-project.org>
Description: A collection of classical tests, including the
             Ansari-Bradley, Bartlett, chi-squared, Fisher, Kruskal-Wallis,
             Kolmogorov-Smirnov, t, and Wilcoxon tests.
License: GPL
Built: R 1.7.1; i386-pc-mingw32; 2003-06-16 08:49:22
```

Index:

ansari.test	Ansari-Bradley Test
bartlett.test	Bartlett Test for Homogeneity of Variances
binom.test	Exact Binomial Test
chisq.test	Pearson's Chi-squared Test for Count Data
cor.test	Test for Association/Correlation Between Paired Samples
fisher.test	Fisher's Exact Test for Count Data
fligner.test	Fligner-Killeen Test for Homogeneity of Variances
friedman.test	Friedman Rank Sum Test
kruskal.test	Kruskal-Wallis Rank Sum Test
ks.test	Kolmogorov-Smirnov Tests
mantelhaen.test	Cochran-Mantel-Haenszel Chi-Squared Test for Count Data
mcnemar.test	McNemar's Chi-squared Test for Count Data
mood.test	Mood Two-Sample Test of Scale
oneway.test	Test for Equal Means in a One-Way Layout

pairwise.prop.test	Pairwise comparisons of proportions
pairwise.t.test	Pairwise t tests
pairwise.table	Tabulate p values for pairwise comparisons
pairwise.wilcox.test	Pairwise Wilcoxon rank sum tests
power.anova.test	Power calculations for balanced one-way analysis of variance tests
power.prop.test	Power calculations two sample test for of proportions
power.t.test	Power calculations for one and two sample t tests
print.power.htest	Print method for power calculation object
prop.test	Test for Equal or Given Proportions
prop.trend.test	Test for trend in proportions
quade.test	Quade Test
shapiro.test	Shapiro-Wilk Normality Test
t.test	Student's t-Test
var.test	F Test to Compare Two Variances
wilcox.test	Wilcoxon Rank Sum and Signed Rank Tests

Nemažai šių (ir kitų) testų taikymo pavyzdžių galima rasti, pvz., svetainėse <http://www.sjsu.edu/faculty/gerstman/EpiInfo/> arba <http://www.math.yorku.ca/SCS/friendly.html>

9.1. Hipotezės apie proporciją

Pradėkime hipotezėmis apie nežinomas tikimybes (proporcijas populiacijoje). Tą uždavinį, kurį anksčiau formulavome intervalų terminais, dabar spėsime kitaip. LŽP opo-
nentai teigia, kad, nežiūrint apklausos rezultatų, šiai partijai prijaučiančių dalis liko ta pati. Šis teiginys vadinamas pagrindine¹ (arba nuline) hipoteze ir žymimas $H_0: p = p_0 (= 0,132)$. Kadangi pagal DSD santykinis dažnis $p^* = \kappa_n / n$ turi būti maždaug lygus “tikrajai” tikimybei p , tai “nedidelės” skirtumo $p^* - p_0$ reikšmės turėtų liudyti hipotezės H_0 naudai, o “didelės” – prieš ją. Taigi esminis klausimas yra toks: ar skirtumas $p^* - p_0 = 0,17 - 0,132 = 0,038$ didelis? Aišku, kad šis klausimas nėra prasmingas, nes, jį reformulavę procentų terminais, galėtume paklausti: ar skirtumas $17 - 13,2 = 3,8$ (procento) didelis? Šį skirtumą reikėtų kažkokia prasme normuoti ir padaryti jį nepriklausomą nuo parametro “dimensijos”. Elgsimės taip: jei teisinga H_0 , tai (pagal CRT) nuokrypio (kitaip - kriterijaus) statistika $z^* = (p^* - p_0) / \sqrt{p_0(1 - p_0) / n}$ turi (maždaug) standartinį normalųjį skirstinį² N ir todėl labiausiai tikėtina, kad z^* reikšmės bus artimos 0. Tolesni samprotavimai priklauso nuo pasirinktos alternatyvos, t.y., nuo sprendimo, kurį darysime, kai z^* bus “toli” nuo 0. Jei mūsų tikslas būtų įrodyti, kad LŽP rėmėjų dalis visoje populiacijoje pasikeitė (nesvarbu, padidėjo ar sumažėjo), tai alternatyvioji hipotezė būtų formuluojama taip: $H_1: p \neq p_0$. Tarkime, kad gautas nuokry-

¹ Nulinė hipotezė paprastai pateikia konservatyvų situacijos vertinimą: dabar yra taip kaip buvo anksčiau. Pats terminas “nulinė hipotezė” atsirado kartu su matematine statistika XX a. pradžioje ir reiškė, kad, pvz., papildomas trešimas duoda nulinių derlingumo priedą.

² Tuo pačiu simboliu z^* žymime du skirtingus dalykus: skirtinguose ėminiuose z^* įgyja skirtingas reikšmes (taigi tai a.d.; jis turi standartinį normalųjį skirstinį N); mūsų ėminyje jis įgijo konkrečią skaitinę reikšmę (taigi dabar tai skaičius). Kai kuriuose matematinės statistikos tekstuose atsitiktinius dydžius žymi didžiosiomis raidėmis, o konkrečias (skaitines) jų reikšmes – mažosiomis. Mes abiem atvejais vartosime tą patį simbolį.

pis $|z^*|$ yra “labai” didelis, tiksliau, įvykio $|N| \geq z^*$ tikimybė tiek maža, kad šį įvykį galima pavadinti (beveik) negalimu. Kadangi neįtikėtini įvykiai paprastai nepasirodo, tai jo pasirodymas matyt reiškia, kad mūsų hipotezė H_0 neteisinga. Tikimybė $P(|N| \geq z^*)$ vadinama kriterijaus p reikšme (p-value), jei ji mažesnė už kriterijaus reikšmingumo lygmenį α (jo standartinė reikšmė 0,05) - H_0 atmetame ir priimame H_1 .

```
> prop.test(170,1000,p=0.132,alt="two.sided")$p.value
[1] 0.0004594326 # H0 neabejotinai atmetame
```

Tuo atveju, kai pagrindinė hipotezė yra ta pati $H_0: p = p_0 (= 0,132)$, bet alternatyva $H_1: p > p_0$, H_0 teks atmesti, jei nuokrypio statistika z^* įgis dideles teigiamas reikšmes, tiksliau, kai tikimybė $P(N \geq z^*)$ bus mažesnė už α .

```
> prop.test(170,1000,p=0.132,alt="greater")$p.value
[1] 0.0002297163 # H0 neabejotinai atmetame
```

Alternatyva $H_1: p < p_0$ nėra labai prasminga (kadangi $p^* = 0,17$, tai aišku, kad iš dviejų hipotezių, $H_0: p = 0,132$ ir $H_1: p < 0,132$, rinksimės pirmąją), tačiau, formaliai taikydami savo kriterijų, gauname

```
> prop.test(170,1000,p=0.132,alt="less")$p.value
[1] 0.9997703 # Nėra mažiau už 0,05 - priimame H0
```

taigi nėra jokio pagrindo atmesti H_0 .

Pažymėsime, kad šioms hipotezėms tikrinti galime taikyti ir tikslų kriterijų, tačiau, kadangi ėminys didelis, išvados bus tos pačios.

```
> binom.test(170,1000,p=0.132,alt="g")$p.value
[1] 0.0003509407 # H0 neabejotinai atmetame
```

9.1 pvz. Išspręsimė vieną uždavinį (žr. ČM1, 171 psl., 7 uždavinys). Naujo medikamento reklamoje teigiama, kad jis sukelia pašalines reakcijas ne daugiau kaip 1% pacientų. Ištyrus 1000 vaistą vartojusių ligonių nustatyta, kad pašalinį poveikį pajuto 32 ligoniai. Ar duomenys neprieštarauja reklaminiam teiginiui? ($\alpha=0,05$)

Nulinė hipotezė suformuluota uždavinyje - $H_0: p \leq p_0 = 0,01$. Jei ši hipotezė būtų teisinga, nežinomos tikimybės p įvertis \hat{p} (t.y., procentas ligonių, pajutusių pašalinį poveikį) turėtų būti maždaug lygus (arba mažesnis už) p_0 . Kadangi $\hat{p} = 32/1000$ yra maždaug tris kartus didesnis už p_0 , kyla įtarimas, kad ko gero teisinga ne hipotezė H_0 , o $H_1: p > p_0$. Taigi tikrinsime hipotezę $H_0: p \leq p_0 = 0,01$ su alternatyva $H_1: p > p_0$ (pažymėsime, kad sprendimo procedūra (tiksliau, kritinė sritis ar p reikšmė) liks tokia pati, jei nulinę hipotezę pakeisime į $H_0: p = p_0$). Aišku, kad mūsų eksperimentai aprašomi Bernulio modeliu ($n=1000$), kiekvieną ligonį, pajutusį pašalinį poveikį, sąlyginai pavadinkime sėkme (iš viso sėkmių yra $\kappa_{1000} = 32$, o sėkmės tikimybė, kai teisinga H_0 , lygi p_0). Kadangi H_0 nenaudai turėtų liudyti didelis “sėkmių” skaičius, apskaičiuokime

kriterijaus p reikšmę, t.y., (stebėto arba dar didesnio nuokrypio į dešinę) tikimybę $P(\kappa_{1000} \geq 32; p = p_0)$ (plg. formules [ČM1, 164 psl.]):

```
> 1-pbinom(31,1000,0.01)
[1] 1.938156e-08
```

Kadangi ši tikimybė aiškiai mažesnė už (pageidaujamą reikšmingumo lygmenį, t.y., $\alpha =$) 0,05, todėl H_0 neabejotinai reikia atmesti. Pažymėsime, kad lygiai tą patį rezultatą gausime ir su

```
> binom.test(32,1000,p=0.01,alt="greater")

Exact binomial test

data: 32 and 1000
number of successes = 32, number of trials = 1000, p-value = 1.938e-08
alternative hypothesis: true probability of success is greater than
0.01
95 percent confidence interval:
 0.02338818 1.00000000 # Kadangi šis intervalas neuždengia 0,01, todėl
sample estimates:      #  $H_0$  (su 95 % reikšm. lygmeniu) reikia atmesti
probability of success
                        0.032
```

“Klasikiniai” vadovėliai siūlo tikimybę $P(\kappa_{1000} \geq 32; p = p_0)$ apskaičiuoti apytiksliai, remiantis CRT (arba, kitais žodžiais, Muavro ir Laplaso integraline teorema):

$$P(\kappa_{1000} \geq 32; p_0) = P\left(\frac{\kappa_{1000} - np_0}{\sqrt{np_0q_0}} \geq \frac{32 - 1000 \cdot 0,01}{\sqrt{1000 \cdot 0,01 \cdot 0,99}}\right)$$

Jei šią tikimybę aproksimuotume dydžiu $P(N \geq 6,992)$ (t.y., nenaudotume Yates'o tolydumo pataisos), tai

```
> 1-pnorm(6.992)
[1] 1.355027e-12
```

Skaitmeniškai rezultatai skiriasi (kadangi visos skirstinio funkcijos aproksimacijos, kai argumentas (šiuo atveju, 6.992) yra “toli nuo vidurkio” (t.y. šiuo atveju, “toli” nuo 0) yra netikslios), tačiau atsakymas toks pat - H_0 reikia neabejotinai atmesti. Pažymėsime, kad lygiai tokį patį atsakymą gautume ir su apytiksliai

```
> prop.test(32,1000,p=0.01,alt="greater",correct=FALSE)

1-sample proportions test without continuity correction

data: 32 out of 1000, null probability 0.01
X-squared = 48.8889, df = 1, p-value = 1.354e-12
alternative hypothesis: true p is greater than 0.01
95 percent confidence interval:
 0.02403373 1.00000000
sample estimates:
      p
0.032
```

Iš esmės tokią pačią išvadą darome ir su kiek tikslesniu prop. test variantu:

```
> prop.test(32,1000,p=0.01,alt="greater")
```

```
1-sample proportions test with continuity correction
```

```
data: 32 out of 1000, null probability 0.01
X-squared = 46.6919, df = 1, p-value = 4.154e-12 # H0 reikia atmesti
alternative hypothesis: true p is greater than 0.01
95 percent confidence interval:
 0.02360359 1.00000000
sample estimates:
      p
0.032
```

Panagrinėkime dar du, kiek nestandartinius, hipotezių tikrinimo pavyzdžius (ši kartą hipotezės $H_0: p = p_0$ nenaudai liudys ne didelės skirtumo $|\kappa_n/n - p_0|$ reikšmės, bet (įtartinai) mažos!) 3-20 psl. buvo aprašyta Monte Carlo procedūra skaičiaus π apytikslei reikšmei rasti. Priminsime: metame “taškus” į kvadratą $[-1,1] \times [-1,1]$; jei taškų metame “daug”, tai “sėkmių” (t.y., taškų, pakliuvusių į vienetinį skritulį arba, kitaip sakant, tenkinančių sąlygą $x_i^2 + y_i^2 < 1$) santykinis dažnis κ_n/n turėtų būti maždaug lygus $\pi/4$.

9.2 pvz. Žinomas statistikas D.B. (vardas ir pavardė autoriui žinomi) tašką metė 10000 kartų ir gavo, kad $\kappa_{10000}/10000 = 0,7854$, taigi³ sutampa su $p = \pi/4$

```
> options(digits=8)
> pi
[1] 3.1415927
> pi/4
[1] 0.78539816
> eps <- 0.7854-pi/4
> eps
[1] 0.0000018
```

penkių ženklų po kablelio tikslumu. Ar negalima įtarti, kad tikslumas ϵ yra “per daug geras”? Apskaičiuokime gauto (arba dar geresnio) tikslumo tikimybę:

$$P\left(\left|\frac{\kappa_n}{n} - p\right| \leq \epsilon\right) = P\left(\left|\frac{\kappa_n - n \cdot p}{\sqrt{n \cdot p \cdot q}}\right| \leq \frac{10000 \cdot 0,0000018}{\sqrt{10000 \cdot \pi/4 \cdot (1 - \pi/4)}}\right) \approx \\ \approx P(|N| \leq 0,00045)$$

```
> stat <- (10000*eps)/sqrt(10000*(pi/4)*(1-pi/4))
> pnorm(stat)-pnorm(-stat)
[1] 0.00035694
```

³ Tiksliau sakant,

```
rdat <- matrix(runif(10000 * 2, min = -1, max = 1), nrow = 2)
sum(colSums(rdat * rdat) < 1) # Ar aišku, kas čia vyksta?
[1] 7854
```


Taigi tokį arba dar geresnį tikslumą galime gauti tik 1 kartą iš 2778 – rezultatas neįtikėtinai⁴ tikslus. (Tai dar kartą įrodo, kad D.B. yra reikšmingas!)

9.3 pvz. Tarkime, kad plokštuma suliniuota lygiagrečiomis linijomis, tarp kurių atstumas $2a$. Jei ant šios plokštumos mestume ilgio $2l$, $l < a$, adatą, tai galima įrodyti (tai vadinamasis Buffon'o uždavinys), kad tikimybė p adatai kirsti vieną iš linijų yra $2l/a\pi$. Šį faktą vėl galėtume panaudoti skaičiui π apytiksliai apskaičiuoti. Pasiremsime žinomu Lazzarini'o rezultatu: 1901 metais jis metė adatą 3408 kartus ir nustatė, kad $\pi \approx 3,1415929$. Įsitikinsime, kad toks "tikslumas" mažai tikėtinas.

Lazzarini'o eksperimente π apskaičiuotas su šešiais tiksliais skaitmenimis po kablelio. Antra vertus, jei kirtimų skaičius (tarkime $\kappa_{3408} = m$) pasikeistų vienetu, šitai pakeistų bent ketvirtą dešimtainį π įverčio skaitmenį. Iš tikro, jei n mažesnis už 5000,

$$\frac{a(m+1)}{2nl} - \frac{am}{2nl} = \frac{a}{2nl} \geq \frac{1}{2n} > 0,0001.$$

Vadinasi, yra tik viena m reikšmė, su kuria Lazzarini's galėjo gauti savo įvertį. Pagal lokaliąją Muavro ir Laplaso teoremą,

$$P(\kappa_n = m) \approx \frac{1}{\sqrt{2\pi np(1-p)}} e^{-\frac{(m-np)^2}{2np(1-p)}} \leq \frac{1}{\sqrt{2\pi np(1-p)}} \Big|_{n=3408}.$$

Tarkime, kad $a = 2l$; tuomet $p = 1/\pi \approx 1/3$, o $P(\kappa_{3408} = m) \leq 0,015$. Taigi, jei Lazzarini's savo eksperimentą pakartotų 100 kartų, tokį tikslumą jis gautų tik maždaug vieną kartą - mažai tikėtina, kad tai buvo pats pirmasis kartas. Kitaip sakant, jis savo rezultatus, matyt, kiek "pakoregavo"...

9.2. Hipotezės apie vidurkį

Norėdami patikrinti, ar kintamojo vidurkis įgyja konkrečią reikšmę, remsimės Student'o kriterijumi t. test. Šį kriterijų naudojame, kai populiacija yra (maždaug) normali (jei taip nėra, tuomet neparametriniai testai (pvz., Wilcoxon'o ranginis kriterijus) labiau tinka populiacijos centro nustatymui).

4 skyriuje nagrinėjome duomenų rinkinį davis (jame buvo pateikti duomenys apie tikrąjį ir praneštąjį apklaustųjų asmenų ūgį ir svorį). Patikrinkime hipotezę, kad moterys teisingai pranešė savo ūgį (tiksliau kalbant, kad moterų tikrojo ūgio vidurkis lygus praneštojo ūgio vidurkiui).

```
> attach(davis)
> davisF_davis[sex=="F",] # Atrenkame tik moteris
> detach(davis)
> attach(davisF)
> dim(davisF)
[1] 112 5
```

⁴ Standartinė "neįtikėtinai" įvykio reikšmė yra 0,05 (taigi 1 kartas iš 20).

```

> mean(height)
[1] 164.7143
> var(height)
[1] 32.02574
> mean(reph)
[1] NA
# Kadangi dalies duomenų trūksta...
> mean(reph,na.rm=T)
# Trūkstamus duomenis išmesime
[1] 162.1980

```

Anksčiau matėme, kad moterų ūgis h_F turi beveik normalųjį skirstinį, todėl hipotezę formuluosime vidurkių terminais: $H_0: a = 162,1980$, $H_1: a \neq 162,1980$ (čia a yra tikrojo ūgio vidurkis). Jei H_0 teisinga, tai pagal DSD skirtumas $\bar{x} - 162,1980$ neturėtų būti didelis. Tiksliau kalbant, neturėtų būti didelė normuotos statistikos $t_{n-1} = (\bar{x} - 162,1980) / \sqrt{s_1^2/n}$ modulio reikšmė. Kadangi tikimybė

$$P\left(|T_{112-1}| > \left| \frac{164,7143 - 162,1980}{\sqrt{32,0257/112}} \right| \right) = 2P(T_{111} < -4,7057) = 7,345 \cdot 10^{-6}$$

mažesnė už 0,05, tai nulinę hipotezę tenka atmesti – moterys klaidingai pranešė savo ūgį. Pažymėsime, kad lygiai tokį patį rezultatą gautume su funkcija `t.test`:

```

> t.test(height,mu=mean(reph,na.rm=T))

One Sample t-test

data: height
t = 4.7056, df = 111, p-value = 7.347e-06
alternative hypothesis: true mean is not equal to 162.1980
95 percent confidence interval:
 163.6547 165.7739
sample estimates:
mean of x
 164.7143

```

9.1 UŽDUOTIS. Surinkę

```

library(MASS)
data(michelson)
michelson
  Speed Run Expt
1   850   1    1
2   740   2    1
3   900   3    1
*****

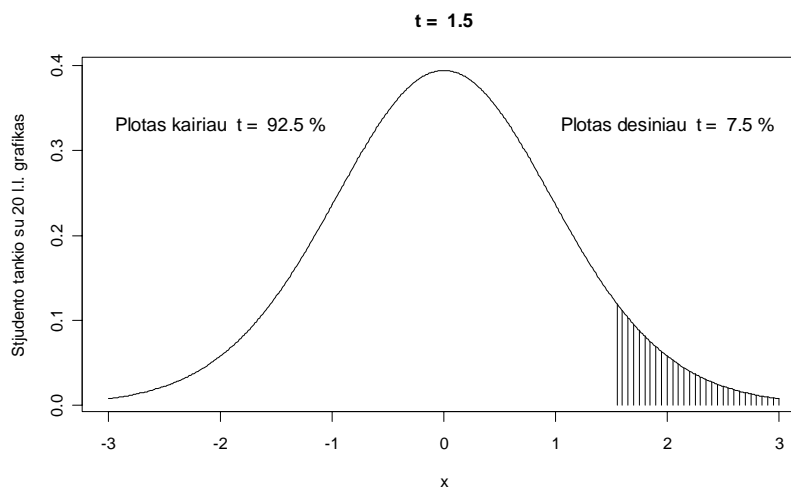
```

pamatysite garsiuosius Michelson'o šviesos greičio matavimų (atliktų 1879 m.) rezultatus (norint gauti tikrąjį greitį, prie šių skaičių reikia pridėti 299000 km/s). a) Ką galite pasakyti apie `Speed` skirstinį? b) Ar matavimo rezultatai suderinami su anksčiau žinomu šviesos greičiu lygiu 0 (prancūzas Cornu, 1876)? c) Ar skiriasi matavimų rezultatai skirtingose `Expt` grupėse? d) Kam "lygus šviesos greitis" po Michelsono bandymų?

9.2 UŽDUOTIS. Hipotezių apie vidurkį tikrinimas yra pagrįstas Student'o kriterijaus p reikšmių skaičiavimu. Štai funkcija `p.value`, kuri pateikia grafinę šio skaičiavimo interpretaciją.

```
p.value<-function(t, df)
{
  z1 <- seq(-3, 3, 0.01)
  tankis <- dt(z1, df)
  PlotasKairiau <- round(pt(t,df), 3)
  PlotasDesiniau <- round(1 - PlotasKairiau, 3)
  plot(z1, tankis, type = "l",xlab = "x", ylab =
    paste("Student'o tankio su",as.character(df),
      "1.1. grafikas"),main = paste("t = ",
    t))
  text(-2, 0.33, paste("Plotas kairiau t = ",
    100 * PlotasKairiau,"%"), cex = 1.2)
  text(2, 0.33, paste("Plotas desiniau t = ",
    100 * PlotasDesiniau,"%"), cex = 1.2)
  z2 <- seq(-3, 3, 0.05)
  height2 <- dt(z2, df)
  len <- length(z2[z2 > t])
  segments(z2[z2 > t], rep(0,len), z2[z2 > t],
    height2[z2 > t])
  cat("p.value=",
    PlotasDesiniau, "\n")
}
```

```
> p.value(1.5,20)
p.value= 0.075
```



9.1 pav. Dešininė p reikšmė

Modifikuokite šią funkciją: 1) vietoje vieno grafiko išbrėžkite tris greta vieną kito (su kairiaja, dvipuse ir dešiniąja p reikšmėmis), 2) atitinkamą uodegą ne užbrūkšniuokite, bet nuspalvinkite (žr. `?polygon`).

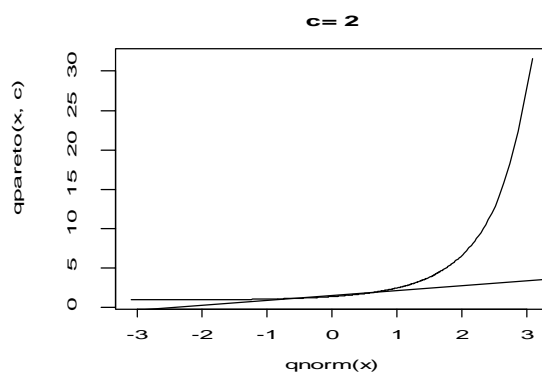
9.3. Pareto skirstinys

Populiariausios populiacijos centro charakteristikos yra jos vidurkis ir mediana. Mes jau aptarėme Student'o kriterijų hipotezėms apie vidurkį tikrinti. Deja, jis nėra tikslus, jei nagrinėjame nedidelį ėminių iš akivaizdžiai nenormalios populiacijos. Be to, šis testas apskritai nepritaikomas, jei nagrinėjamas a.d. neturi baigtinio vidurkio (tai atitinka gana populiarių ekonometrijoje "sunkių uodegų" atveji). Ėminių prasme sunkios uodegos reiškia, kad juose bus (gal ir nedaug, bet) "didelių" reikšmių (Lietuvoje yra žmonių, turinčių labai dideles pajamas (teisybė, jų nedaug); internetu kartas nuo karto perduodami labai dideli failai ir pan.). Abiem šiais atvejais remsimės Wilcoxon'o kriterijumi medianoms. Tačiau prieš tai pakalbėkime apie Pareto skirstinį.

Sakome, kad a. d. X turi Pareto skirstinį, jei jo tankis yra pavidalo $p(x) = cx^{-(c+1)}1_{(1,\infty)}(x)$, $c > 0$. Šis a.d. turi baigtinį vidurkį tik tuomet, kai $c > 1$:

$$m_1 = EX = \int_1^{\infty} x \cdot cx^{-(c+1)} dx = c/(c-1), \quad c > 1;$$

antra vertus, koks bebūtų c , a.d. X turi visus momentus iki c : $m_v = EX^v = c/(c-v)$, $0 < v < c$. Taigi, jei $c \leq 1$, a.d. X vidurkio neturi, nors jo mediana egzistuoja visuomet: $mX = \sqrt[3]{2}$, $c > 0$. Vidurkio egzistavimas yra glaudžiai susijęs su "uodegos sunkumu": kuo sunkesnė uodega (t.y., kuo lėčiau artėja į nulį tikimybė $P(X > x)$, kai $x \rightarrow \infty$), tuo mažiau momentų turi a.d. X .



9.2 pav. Pareto skirstinio kvantilių grafikas (plg. 4 skyrių); dešinioji uodega "labai sunki"

Parašysime keturias R funkcijas, kurios skaičiuos Pareto tankio funkciją `dpareto`, skirstinio funkciją `ppareto`, kvantilius `qpareto` ir generuos Pareto atsitiktinius skaičius `rpareto`.

```
dpareto <- function(x,c){
  if(c<=0)stop("c turi buti > 0") # Diagnostinis žingsnis
  ifelse(x<1,0,c/x^(c+1))}

ppareto <- function(q,c){
  if(c<=0)stop("c turi buti > 0")
  ifelse(q<1,0,1-1/q^c)}
```

```

qpareto <- function(p,c){
  if(c<=0) stop("c turi buti > 0")
  if(any(p<0)|any(p>1)) # Simbolis | žymi loginį AR
  stop("p turi buti tarp 0 ir 1")
  q_(1-p)^(-1/c)
  q}

```

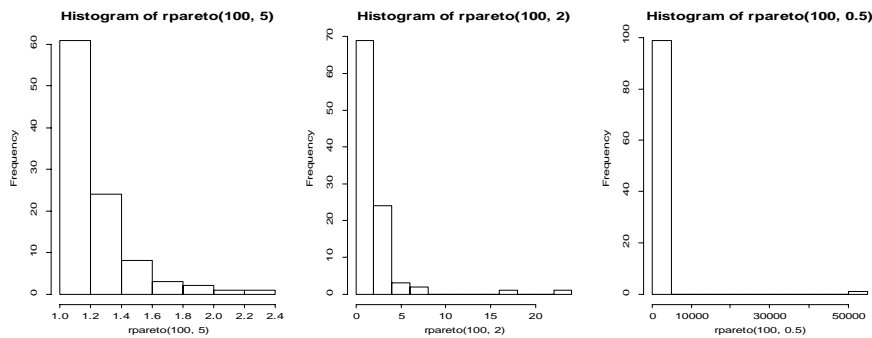
Tarkime, kad α yra $[0,1]$ tolygusis a.d. Tuomet a.d. $X = F^{-1}(\alpha)$ skirstinio funkcija yra F (iš tikrųjų, $P(X < x) = P(F^{-1}(\alpha) < x) = P(\alpha < F(x)) = F(x)$; tai standartinis, nors ne visuomet pats efektyviausias atsitiktinių skaičių generavimo būdas). Kadangi Pareto atveju $F^{-1}(\alpha) = \sqrt[c]{\frac{1}{1-\alpha}}$, o a.d. $1-\alpha$ skirstinys sutampa su α skirstiniu, tai

```

rpareto <- function(n,c){
  if(c<=0) stop("c turi buti >0")
  rp_runif(n)^(-1/c)
  rp}

```

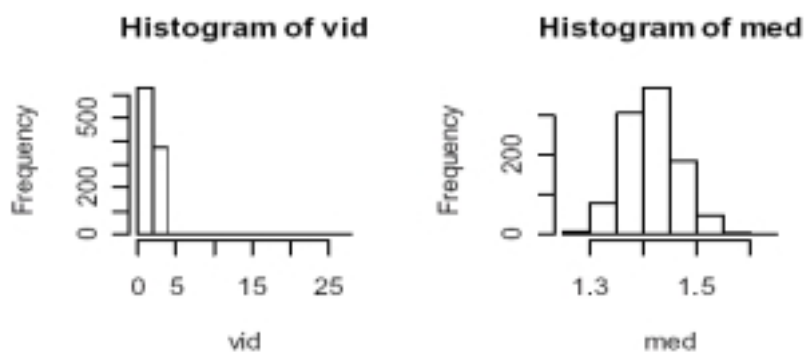
Išbrėšime tris Pareto ėminių histogramas.



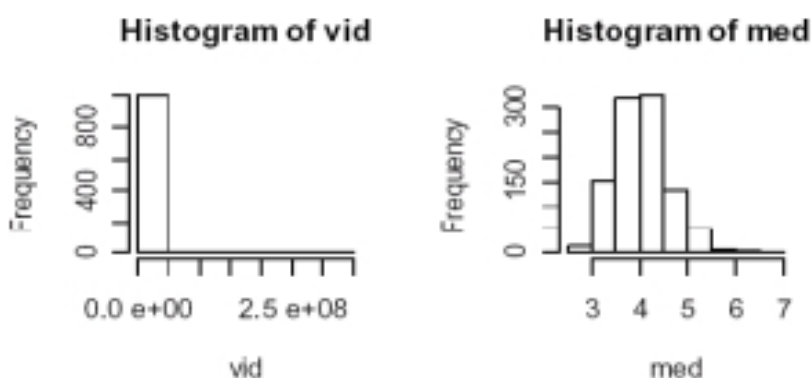
9.3 pav. Kuo c mažesnis, tuo didesnės reikšmės atsiranda ėminyje (kai $c = 2$ - dispersija begalinė, kai $c = 0.5$ - begalinis ir vidurkis)

Pademonstruosime, kad tuomet, kai c mažas, geresnė “centro” charakteristika yra mediana, o ne vidurkis. Generuosime 1000 ėminių po 200 Pareto atsitiktinių skaičių, apskaičiuosime kiekvieno ėminio vidurkį ir medianą ir pasižiūrėsime jų histogramas (kuo c mažesnis, tuo didesni medianos pranašumai).

9.4 pav. $c=5$ – medianų histograma (dešinėje) kiek siauresnė už vidurkių histogramą



9.5 pav. $c=2$ – medianų histograma (dešinėje) žymiai siauresnė už vidurkių histogramą



9.6 pav. $c=0.5$ – medianų histograma (dešinėje) yra žymiai siauresnė už vidurkių histogramą

9.3 UŽDUOTIS⁵. Pareto skirstiniui, kaip ir daugeliui kitų, be formos parametro c galima įvesti ir dar vieną, mastelio, parametą σ :

$$p(x) = c\sigma^c x^{-(c+1)} 1_{(\sigma, \infty)}(x).$$

- 1) Perrašykite d pareto, p pareto, q pareto ir r pareto funkcijas dviejų parametų atvejui.
- 2) Tarkime, kad kažkokios populiacijos pajamos yra aprašomos Pareto skirstiniu su formos parametru 3 ir mastelio parametru 1000. Kokios populiacijos dalies pajamos yra tarp 2000 ir 4000?
- 3) Apskaičiuokite pajamų medianą ir 90% kvantilį.
- 4) Apskaičiuokite pirmąjį ir trečiąjį kvartilius ir IQD.
- 5) Apskaičiuokite pajamų vidurkį.
- 6) Apskaičiuokite pajamų standartą.

⁵ Tai tikimybių teorijos, o ne matematinės statistikos uždavinys.

9.4. Hipotezės apie medianą

Priminsime, kad duomenų rinkinio `bwages` poaibį `w0` (`w1`) sudaro duomenys apie Belgijos moterų (atitinkamai, vyrų) atlyginimus. Ar galime teigti, kad moterų atlyginimas yra toks pat kaip ir vyrų? Abi procedūros,

```
> attach(bwages)
> tapply(wage,male,mean)
      0      1
413.9497 466.4193
```

ir

```
> tapply(wage,male,median)
      0      1
383.5227 422.8220
```

tvirtina, kad vyrų atlyginimas didesnis. Tačiau ar negalima šio skirtumo paaiškinti vien ėminio atsitiktinumu? Šį klausimą galima suformuluoti kaip hipotezių tikrinimo uždavinį: $H_0 : a_m = 466,4193 (= a_v)$ (diskriminacijos nėra) su alternatyva $H_1 : a_m < 466,4193$ (gal vis tik “skaičiai nemeluoja”?). Deja, taip formuluoti uždavinį negerai, nes

1) moterų atlyginimo ėminys yra aiškiai nenormalus.

R turi kelias normalumo tikrinimo procedūras⁶. Kolmogorovo ir Smirnovo testas `ks.test` mums netinka, kadangi jis reikalauja, kad normaliojo a.d. parametrai būtų žinomi iš anksto (t.y., jie negali būti pakeičiami empiriniais ėminio momentais). Tokiu atveju galima taikyti chi kvadrato kriterijų, tačiau R funkcija `chisq.test` nėra tam tiesiogiai pritaikyta. Normalumą patikrinsime su Shapiro ir Wilk'o kriterijumi [Li, 493 p.]:

```
> shapiro.test(w0)

Shapiro-Wilk normality test

data:  w0
W = 0.9011, p-value = < 2.2e-16
```

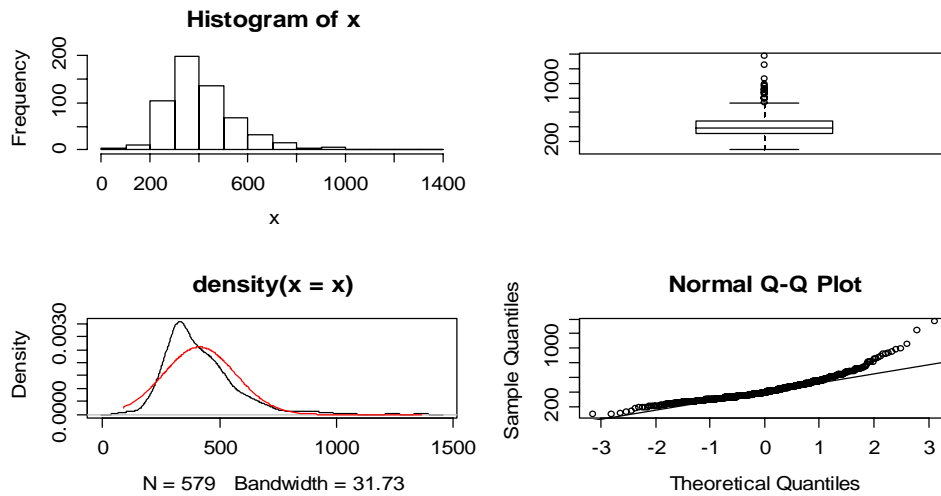
Kadangi p reikšmė yra labai maža, nulinę hipotezę apie `w0` normalumą tenka neabejotinai atmesti.

2) Tai, kad `w0` skirstinys nėra normalus – pusė bėdos. Blogiau yra tai, kad, atrodo, jis turi sunkias uodegas ir galimas daiktas neturi baigtinio (teorinio) vidurkio:

```
> eda.shape(w0)
Vidurkis= 413.9497 , Mediana= 383.5227 (simetriniu atveju turi beveik sutapti)
Standartas= 153.6380 , MAD= 122.1459 (kai nera isskirciu, turi beveik sutapti)
```

⁶ Procedūros, tikrinančios hipotezes apie stebimojo a.d. skirstinio funkciją, vadinamos suderinamumo (=goodness of fit arba tiesiog `gof` (angl.)) kriterijais. Kolmogorovo ir Smirnovo kriterijus tinka tik tolydiems skirstiniams, o chi kvadrato – tolydiems ir diskretiems.

Asimetrijos koeficientas= 1.576122 (simetriniu atveju turi buti 0)
 Ekscesas= 7.823961 (normaliuoju atveju turi buti 3)



9.7 pav. Kintamojo w0 skirstinio priešanalizės grafikai

Tokiu įtartinu atveju geriau hipotezes formuluoti medianoms:
 $H_0 : med(w0) = 422,8220 (= med(w1))$ su alternatyva $H_1 : med(w0) < 422,8220$. Tai-
 kysime Wilcoxon'o kriterijų:

```
> wilcox.test(w0,mu=median(w1),alt="l")
```

Wilcoxon signed rank test with continuity correction

```
data: w0
V = 67201, p-value = 1.590e-05
alternative hypothesis: true mu is less than 422.822
```

Kadangi p reikšmė žymiai mažesnė už 0,05, nulinę hipotezę apie atlyginimų (medianų) lygybę atmetame. Pažymėsime, kad tam tikra prasme jungtinė hipotezė⁷
 $H_0 : med(w1) = 383,5227 (= med(w0))$ su alternatyva $H_1 : med(w1) > 383,5227$ yra at-
 metama dar ryžtingiau:

```
> wilcox.test(w1,mu=median(w0),alt="g")
```

Wilcoxon signed rank test with continuity correction

```
data: w1
V = 284298, p-value = < 2.2e-16
alternative hypothesis: true mu is greater than 383.5227
```

Taigi, moterų atlyginimas neabejotinai mažesnis.

⁷ Vyru atlyginimo skirstinys dar labiau panašus į Pareto (pasižiūrėkite su `eda.shape(w1)`).

9.5. Suderinamumo kriterijai

Daugelis statistinių procedūrų galioja tik tuomet, kai populiacija turi konkretų (pvz., normalųjį) skirstinį. Aptarsime du kriterijus, kurie leidžia patikrinti hipotezes apie stebimojo a.d. skirstinį. Pirmasis jų yra χ^2 kriterijus, kuris taikomas kaip diskretiems taip ir tolydiems a.d. Jis pagrįstas tuo, kad (tuomet kai teisinga nulinė hipotezė apie skirstinį) empiriniai dažniai negali labai skirtis nuo teorinių. Taikant χ^2 kriterijų, tolydžius duomenis reikia grupuoti, dėl ko prarandama dalis informacijos. Todėl tolydžių stebėjimų atveju geriau taikyti kitą, būtent Kolmogorovo kriterijų (žr. `ks.test`), kuris remiasi tuo, kad (tuomet kai teisinga nulinė hipotezė apie skirstinį) empirinė skirstinio funkcija negali labai skirtis nuo teorinės. Kolmogorovo kriterijaus pagrindinis trūkumas yra tas, kad hipotetinės skirstinio funkcijos parametrai turi būti žinomi iš anksto, jų negalime vertinti iš ėminio⁸.

9.3 pavyzdys. Žemiau pateikti klasikiniai Bortkiewicz'iaus duomenys apie skaičių žmonių, užmuštų arklio kanopos smūgiu 10-tyje prūsų armijos korpų per 20 metų (1875-1894).

v_i – mirčių skaičius viename korpuse per metus	0	1	2	3	4	
Skaičius atvejų, kai įvyko i mirčių	109	65	22	3	1	$n=200$

Kadangi kareivių korpuse daug, o tokios mirties tikimybė maža, tikėtina, kad mirčių skaičius turi (diskretųjį) Puasono skirstinį. Šią hipotezę tikrinsime su χ^2 kriterijumi. Priminsime, kad mūsų atveju nuokrypio statistika⁹

$$\left(\sum_{i=1}^k \frac{(O_i - E_i)^2}{E_i} \right) = \sum_{i=1}^k \frac{(v_i - np_i(\hat{\lambda}))^2}{np_i(\hat{\lambda})}$$

(čia $k (=5)$ yra grupių (reikšmių) skaičius, $\hat{\lambda}$ yra Puasono skirstinio parametro λ χ^2 minimumo įvertis [Ku, 324 psl.] (paprastai $\hat{\lambda} \approx \bar{x}$), o $p_i(\hat{\lambda}) = e^{-\hat{\lambda}} \hat{\lambda}^i / i!$ yra Puasono tikimybės) turi (beveik) χ_{k-2}^2 skirstinį¹⁰ tik tuomet, kai ėminio dydis n pakankamai didelis. Praktinės rekomendacijos įvairuoja: kartais reikalaujama, kad visi np_i būtų¹¹ ≥ 10 [ČM1, 198 psl.], kartais ≥ 5 [Ku, 324 psl.], o kartais tiesiog ≥ 1 [LL1, 365 psl.]. Ši kartą remsimės paskutiniąja rekomendacija (R paketo funkcija `chisq.test` remiasi viduriniąja: jei bent vienas $np_i < 5$, R atspausdina įspėjimą (`warning`) apie galimai neteisingą rezultatą). Iš pradžių šio testo reikalingas statistikas apskaičiuosime “rankomis”.

```
> B_as.matrix(read.table("bort.txt"))
> B
```

⁸ Yra dvi malonios išimtys: tai normalusis ir eksponentinis skirstiniai. Vistik ir tuomet reikia naudotis ne Kolmogorovo statistika, o jos modifikacijomis (plg. 9.12 užduotį).

⁹ Čia O_i yra registruotų (*Observed*), o E_i – prognozuojamų (*Expected*) stebėjimų skaičius i -joje grupėje.

¹⁰ Laisvės laipsnių skaičius yra ne $k-1$, bet (kadangi (vieną) parametru λ reikia įvertinti iš ėminio) $k-2$.

¹¹ Atkreipiame dėmesį: ne v_i , bet $np_i!$

```

      V1 V2 V3 V4 V5
1     0  1  2  3  4
2    109 65 22  3  1

> wm <- weighted.mean(B[1,],B[2,]/200)
> wm
[1] 0.61 # Tai  $\bar{x}$ .

> cbind(B[2,], 200*dpois(0:4,wm), (B[2,]-200*dpois(0:4,wm))^2/
(200*dpois(0:4,wm)))
      [,1]      [,2]      [,3]
V1    109 108.6701738 0.001001060
V2     65  66.2888060 0.025057337
V3     22  20.2180858 0.157048402
V4      3   4.1110108 0.300253401
V5      1   0.6269291 0.222005731 # Šios grupės teorinis dažnis <1!

```

Kadangi paskutinės grupės teorinis dažnis $0,6269291 < 1$, paskutines dvi grupes apjungsiame į vieną.

```

> B2_c(B[2,1:3],B[2,4]+B[2,5]) # Sudedame dažnius
> pB2_c(dpois(0:2,wm), dpois(3,wm)+dpois(4,wm)) # Sudedame tikimybes
> sB2 <- sum((B2-200*pB2)^2/(200*pB2))
> sB2 <- sum((B2-200*pB2)^2/(200*pB2))
> sB2
[1] 0.2980418
> 1-pchisq(sB2,4-2)
[1] 0.8615511

```

Taigi,

$$P(\chi_2^2 \geq 0,298) = 0,86$$

yra žymiai didesnė už $0,05$ ir todėl nėra nė menkiausio pagrindo atmesti hipotezę apie tai, kad mirčių skaičius turi Puasono skirstinį.

Visus skaičiavimus gali pagreitinti funkcija `chisq.test`. Vienintelė problema čia ta, kad ši funkcija taria, kad l.l. skaičius lygus $k-1$, o ne mums reikalingas $k-2$.

```

> chiB2_chisq.test(B2,p=pB2)
Warning message:
Chi-squared approximation may be incorrect in: chisq.test(B2, p = pB2)
> chiB2

```

Chi-squared test for given probabilities

```

data: B2
X-squared = 0.298, df = 3, p-value = 0.9604
# Neteisingas l.l. skaičius ir neteisinga p reikšmė
> 1-pchisq(chiB2$statistic,2) # Rankomis įrašome teisingą l.l. skaičių
X-squared
0.8615511 # Teisingas atsakymas

```

9.4 UŽDUOTIS. R1 disko direktorijoje ...Data\Misc yra failas FlowMeter.txt. Jame pateikti šilumos srauto matavimo prietaiso kalibravimo rezultatai (tai “gerų” duomenų pavyzdys). Atlikite šių duomenų priešanalizę. Be kitų procedūrų, patikrinkite ar ėminys turi išskirčių. Tam taikykite Grubbs’o kriterijų: jei

$$G > \frac{n-1}{\sqrt{n}} \sqrt{\frac{t_{\alpha/2n}^2(n-2)}{n-2+t_{\alpha/2n}^2(n-2)}}$$

(čia $G = \frac{\max |x_i - \bar{x}|}{s}$, o $t_{\alpha}(n)$ yra Student'o skirstinio su n l.l. lygmens α viršutinė kritinė reikšmė, t.y., $P(T_n > t_{\alpha}(n)) = \alpha$, $\alpha \approx 0.05$), tai hipotezė H_0 : išskirčių ėminyje nėra yra atmetama su reikšmingumo lygmeniu α . Tokiu atveju, toliausiai nuo \bar{x} nutolęs narys pašalinamas iš ėminio, likę nariai (n pakeitus į $n-1$) tikrinami iš naujo ir t.t. Taip pat apskaičiuokite ėminio skaitines charakteristikas ir jų pasikliauties intervalus. Patikrinkite hipotezė $H_0 : a = 5$ su visomis trijomis ($\neq, >$ ir $<$) alternatyvomis.

9.5 UŽDUOTIS. Su File|Source R code...|rw1051|library|Simple|data nusiskaitykite cancer.R duomenų rinkinį (po to pabandykite `> cancer`). Kokias išvadas galima padaryti iš `cancer$colon`¹² histogramos? Ką reiškia, kad mediana lygi 372, o vidurkis 457?

```
> median(cancer$colon)
[1] 372
> mean(cancer$colon)
[1] 457.4118
```

Kodėl vidurkis yra žymiai didesnis už medianą? Ką aktualiau šiuo atveju žinoti ligoniui (ir gydytojui) – medianą ar vidurkį? Yra žinoma, kad ligonių, nevartojančių vitamino C, gyvenimo trukmės mediana yra 350. Patikrinkite hipotezė, kad mūsų atveju mediana lygi 350 (t.y., vitamino C teikiamas gyvenimo trukmės priedas yra nereikšmingas) su alternatyva, kad iš tikrųjų ji didesnė (taikykite a) ženklų kriterijų `simple.median.test` iš paketo `simple` (žr. [Ku], 4 sk. 15 skyrelis) ir b) Wilcoxon'o ranginį kriterijų `wilcox.test`).

9.6 UŽDUOTIS. JAV gimstančių kūdikių svoris turi maždaug normalų skirstinį su vidurkiu 115,2 uncijos (= 3,2659 kg). Pediatras surinko duomenis apie 20-ties smarkiai rūkančių moterų pagimdytų vaikų svorį – jo vidurkis buvo 114,0, o $s=4,3$. Patikrinkite hipotezė, kad rūkančių moterų pagimdytų vaikų svoris mažesnis negu visoje populiacijoje.

9.7 UŽDUOTIS. Kompakto R1 direktorijoje Data\StatLab yra duomenų rinkinys `babies1-data`. Štai jo pradžia:

```
bwt smoke
120 0
113 0
128 1
123 0
.....
```

Čia `bwt` yra gimusio kūdikio svoris, o `smoke = 0`, jei motina niekuomet nerūkė ir `=1`, jei nėštumo metu rūkė (plg. 9.6 užduotį). a) Patikrinkite hipotezė, kad rūkančių moterų pagimdytų vaikų svoris mažesnis negu visoje populiacijoje, b) Patikrinkite hipotezė, kad

¹² `cancer$colon` komponentėje yra duomenys apie ligonių, naudojančių didelius kiekius vitamino C, gyvenimo trukmę.

visų vaikų svorio vidurkis lygus 115,2 ir c) Patikrinkite hipotezę, kad rūkančių motinų vaikai sveria mažiau negu 115,2¹³.

9.8 UŽDUOTIS. Uždaviniai, kuriuose reikia patikrinti konservatyvią hipotezę, kad “niekas nepasikeitė”, dažnai atsiranda medicinoje – tai vadinamieji “prieš ir po” uždaviniai (pvz., to paties ligoonio kraujospūdį matuojame prieš ir po gydymo). Formalizuoti šią hipotezę galime įvairiai, pvz., kaip hipotezę apie skirstinio funkcijų lygybę $H_0: F_{prieš} = F_{po}$ (žr. Kolmogorovo-Smirnovio kriterijų 10 sk.) arba kaip¹⁴ $H_0: \text{stebėjimų skirtumo vidurkis} = 0$ arba kaip $H_0: \text{stebėjimų skirtumo mediana} = 0$. Tarkime, kad x yra ligoonio depresijos skalės reikšmė pirmojo vizito pas gydytoją metu, o y – praėjus tam tikram laikui nuo gydymo pradžios (žr. `wilcox.test`):

```
x <- c(1.83, 0.50, 1.62, 2.48, 1.68, 1.88, 1.55, 3.06, 1.30)
y <- c(0.878, 0.647, 0.598, 2.05, 1.06, 1.29, 1.06, 3.14, 1.29)
```

Patikrinkite hipotezes, kad skirtumo a) vidurkis, b) mediana lygūs nuliui. Išbrėžkite sklaidos diagramą bei patikrinkite hipotezę, kad x ir y koreliacijos koeficientas lygus nuliui.

9.9 UŽDUOTIS. Rutherford’as, Chadwick’as ir Ellis’as atliko $N=2608$ bandymus¹⁵, kurių metu stebėjo radioaktyvią medžiagą. Kiekvienas bandymas truko 7,5 s, jo metu buvo registruojamas dalelių, pasiekusių Geigerio skaitiklį, skaičius i :

i	0	1	2	3	4	5	6	7	8	9	10	
v_i	57	203	383	525	532	408	273	139	45	27	16	2608

Patikrinkite hipotezę, kad skaitiklį pasiekusių dalelių skaičius turi Puasono skirstinį. Palyginkite santykinų dažnių ir Puasono tikimybių grafikus.

9.10 UŽDUOTIS. Žemiau pateikti Weldon’o¹⁶ eksperimento duomenys: 4096 kartus buvo metami 12 žaidimų kauliukų ir kiekvieną kartą registruojamas iškritusių šešiu-kių skaičius i :

i	0	1	2	3	4	5	6	≥ 7	
v_i	447	1145	1181	796	380	115	24	8	4096

Patikrinkite hipotezę, kad kauliukai taisyklingi.

9.11 UŽDUOTIS. Žemiau esančioje lentelėje pateikti duomenys apie kritulių kiekį X tam tikroje vietovėje (x_i žymi kritulių kiekį cm)

¹³ Daugiau informacijos apie šį eksperimentą galima rasti kompacto R1 faile `Knygos_apie_R&S\Stat-Lab\sample.pdf`.

¹⁴ Čia nagrinėjame vadinamuosius porinius (paired, matched) ėminius – aišku, kad įrašų kiekis abiejose ėminiuose yra vienodas, o reikšmės x_i ir y_i yra priklausomos.

¹⁵ Tai klasikiniai fizikos bandymai. Ką žinote apie juos?

¹⁶ Plg. <http://psychclassics.yorku.ca/Fisher/Methods/chap3.htm>

9.1 lentelė

x_i	v_i	x_i	v_i	x_i	v_i
16	1	24	2	32	7
17	0	25	12	33	4
18	0	26	4	34	4
19	3	27	7	35	4
20	2	28	4	36	3
21	3	29	8	37	3
22	0	30	9	38	0
23	3	31	6	39	1
					$n=90$

Mes norėtume patikrinti hipotezę apie kritulių kiekio normalumą. Tai galima atlikti bent penkiais būdais.

1-asis būdas. Palyginkite X asimetriją ir ekscesą su normaliojo a.d. tais pačiais parametrais. Priminsime (plg. 4.16+ psl.), kad empiriniu asimetrijos koeficientu vadiname

$$ask = m_3 / m_2^{3/2}$$

(čia $m_r = \sum_i v_i (x_i - \bar{x})^r / n$), o empiriniu ekscesu

$$eks = m_4 / m_2^2 - 3.$$

Fisher'is siūlo vietoje ask nagrinėti "patobulintą" asimetriją $g_1 = k_3 / k_2^{3/2}$, kurioje $k_3 = m_3 / \{(1-1/n)(1-2/n)\}$, o $k_2 = m_2 / (1-1/n)$. Jei X turi normalųjį skirstinį, tai tokį pat skirstinį turi ir g_1 (su vidurkiu 0 ir dispersija $6n(n-1) / \{(n+3)(n+1)(n-2)\}$). Panašiai, vietoje dydžio eks tikslinga nagrinėti reiškinį $g_2 = k_4 / k_2^2$, kuriame $k_4 = m_4 / \{(1-2/(n+1))(1-2/n)(1-3/n)\} - 3m_2^2 / \{(1-2/n)(1-3/n)\}$. Dydis g_2 turi dabar normalųjį skirstinį su vidurkiu 0 ir dispersija $24n(n-1)^2 / \{(n+5)(n+3)(n-2)(n-3)\}$. Patikrinkite, kad $g_1 = -0,231$ su standartine paklaida $\pm 0,254$, o $g_2 = -0,302$ su standartine paklaida $\pm 0,503$. Kitaip sakant, turimi duomenys neprieštarauja hipotezei apie kritulių kiekio normalumą.

2-asis būdas. Taikysime χ^2 kriterijų. Pirmiausiai įvertinsime nežinomus skirstinio parametrus a ir σ^2 . Remsimės vidurkio ir dispersijos formulėmis grupuotiems duomenims (grupavimo intervalo plotis (kitai sakant, apvalinimo dydis) $h=1$) su Šepardo pataisomis (žr., pvz., [LL1, 351 psl.]):

$$\hat{a} = \frac{1}{n} \sum_{j=1}^k v_j x_j,$$

$$\hat{\sigma}^2 = \frac{1}{n} \sum_{i=1}^k v_i (x_i - \hat{a})^2 - \frac{h^2}{12}.$$

Apskaičiavę gauname, kad $\hat{a} = 28,62$, $\hat{\sigma}^2 = 23,013 - 1/12 = (4,788)^2$. Norėdami taikyti χ^2 kriterijų, (jau kiek grupuotus) duomenis reikia sugrupuoti į stambesnius intervalus. Pradėti reiktų nuo “uodegų”. Remiantis Kokreno rekomendacija tolydiesiems skirstiniams, grupes “uodegose” reiktų apjungti taip, kad teorinis dažnis grupėje būtų nemažesnis už 1, o kitur – bent 5 (grupavimo intervalų pločiai nebūtinai vienodi). 9.1 lentelėje gretimos grupės (iš viso jų 12) nuspalvintos skirtingomis spalvomis¹⁷.

Grupavimo intervalas	Elementų skaičius v_i	Tikimybė np_i
$(-\infty; 16,5)$	1	$90 \cdot \Phi\left(\frac{16,5 - 28,62}{4,788}\right)$
$(16,5; 20,5)$	5	$90 \cdot \left(\Phi\left(\frac{20,5 - 28,62}{4,788}\right) - \Phi\left(\frac{16,5 - 28,62}{4,788}\right)\right)$
...
$(38,5; \infty)$	1	$90 \cdot \left(1 - \Phi\left(\frac{38,5 - 28,62}{4,788}\right)\right)$

Apskaičiuokite χ^2 statistiką (ji turės 12-1-2=9 l.l.) ir patikrinkite hipotezę apie kritulių kiekio normalumą.

9.12 UŽDUOTIS. Išnagrinėkite 9.11 užduoties duomenis ir dar dviem būdais patikrinkite hipotezę apie duomenų normalumą.

3-asis būdas. Taikant χ^2 kriterijų, duomenis grupuoti galima įvairiai ir todėl atsakymas nėra vienareikšmis. Antra vertus, grupuodami duomenis neišvengiamai prarandame dalį informacijos, todėl dar ir dėl šios priežasties verta paieškoti kitokių suderinamumo hipotezių tikrinimo kriterijų. Vienas iš jų – Kolmogorovo kriterijus. Deja, Kolmogorovo kriterijus taikomas tik duomet, kai hipotetinės (tolydaus) skirstinio funkcijos visi parametrai žinomi. Kadangi mūsų atveju vidurkį ir dispersiją reikia skaičiuoti pagal ėminio duomenis, taikysime Kolmogorovo kriterijaus Lilliefors'o modifikaciją. Šiuo atveju reikės apskaičiuoti dydį

$$D_n = \sup_{x \in R} |F_n(x) - \Phi((x - \bar{x})/s_1)|.$$

Viena iš problemų, atsirandančių skaičiuojant šį dydį, yra ta, kad tarp mūsų duomenų $x = \{16,19,19,19,20,20,21,\dots\}$ yra pasikartojančių (Kolmogorovo kriterijus skirtas tolydiems duomenims, todėl to neturėtų būti). Kadangi tai atsirado dėl apvalinimo, duomenis “atapvalinkime”:

```
x <- x + runif(90,-0.5,0.5)
```

Dabar komanda

¹⁷ Įvairiai grupuojant duomenis, galima gauti skirtingus atsakymus (tačiau labai skirtingi jie neturėtų).

`ks.test(x, "pnorm", mean=mean(x), sd=sd(x))$statistic`

apskaičiuotų dydį D_n . Norint nustatyti, ar šis skirtumas reikšmingas, reikėtų pasinaudoti Lilliefors'o lentelėmis. Jose pateikti statistikos D_n $1-\alpha$ eilės kvantiliai $D_n(1-\alpha)$: jei $D_n > D_n(1-\alpha)$, tai hipotezė H_0 : dydis X turi normalųjį skirstinį atmetama su reikšmingumo lygmeniu α .

Lilliefors'o (testo kritinių reikšmių) lentelė

n	α			n	α		
	0.1	0.05	0.01		0.1	0.05	0.01
4	0.352	0.381	0.417	15	0.201	0.220	0.257
5	0.315	0.337	0.405	16	0.195	0.213	0.250
6	0.294	0.319	0.364	17	0.189	0.206	0.245
7	0.276	0.300	0.348	18	0.184	0.200	0.239
8	0.261	0.285	0.331	19	0.179	0.195	0.235
9	0.249	0.271	0.311	20	0.174	0.190	0.231
10	0.239	0.258	0.294	25	0.158	0.173	0.200
11	0.230	0.249	0.284	30	0.144	0.161	0.187
12	0.223	0.242	0.275	>30	$0.805/\sqrt{n}$	$0.886/\sqrt{n}$	$1.031/\sqrt{n}$
13	0.214	0.234	0.268				
14	0.207	0.227	0.261				

- i) Remdamiesi Lilliefors'o kriterijumi, patikrinkite hipotezę apie mūsų duomenų normalumą.
- ii) Taikydami Monte-Carlo metodą, apytiksliai apskaičiuokite eilutės su $n=8$ nurodytus kvantilius. Tam generuokite 8 standartinius normaliuosius a.d. ir apskaičiuokite Lilliefors'o D_8 ; šią procedūrą pakartokite 100000 kartų ir raskite gauto rinkinio 0.9, 0.95 ir 0.99 kvantilius.

4-asis būdas. Vienas efektyvių būdų normalumui tikrinti yra `shapiro.test`. Patikrinkite mūsų duomenų normalumą šiuo būdu.

9.13 UŽDUOTIS. Jei teisinga hipotezė apie duomenų normalumą, vadinamoji Jarque-Bera statistika

$$JB = \frac{n}{6} \left(ask^2 + \frac{1}{4} (eks - 3)^2 \right)$$

turi χ^2 skirstinį su 2 laisvės laipsniais. Patikrinkite 9.11 užduoties duomenų normalumą.

9.14 UŽDUOTIS. Nuvairuokite į <http://psychclassics.yorku.ca/Fisher/Methods/chap3.htm>. Pakartokite TABLE 10 pateiktą analizę. Ar taisyklingas aprašytasis žaidimų kauliukas?

9.15 UŽDUOTIS. Genų jungimasi ir paveldimumą valdo tikimybiniai dėsniai. Snedecor'as yra pateikęs tokius kukurūzo chlorofilo paveldimumo duomenis: tarp 1103

savidulkių heterozigotinių žalių sėjinukų, 854 buvo žali, o 249 – geltoni. Teorija tvirtina, kad šis santykis turėtų būti lygus 3:1. Atspausdinkite turimus ir prognozuojamus dažnius. Patikrinkite ar šie duomenys neprieštaruja teorijai.

10. Sprendžiamoji statistika: hipotezių tikrinimas (du ėminiai)

R pakete yra daug statistinių kriterijų skirtų dviejų populiacijų parametrų lyginimui, kitais žodžiais, hipotezėms $H_0 : \theta_1 = \theta_2$ (čia θ_1 ir θ_2 yra tie patys abiejų populiacijų parametrai) tikrinti. Štai trumpa šių kriterijų apžvalga.

- Dviejų populiacijų proporcijų lygybės testas `prop.test`.
- Dviejų požymių nepriklausomumo testai `fisher.test` ar `chisq.test` (tai vadinamieji dažnių lentelių uždaviniai).
- Dviejų populiacijų Stjudento kriterijus `t.test` – jis skirtas hipotezei $H_0 : a_1 = a_2$ tikrinti (čia a_1 ir a_2 yra populiacijų vidurkiai). Jei duomenų rinkiniai nedideli, jie turėtų būti maždaug normalūs. Šis testas turi keletą variantų: jį galima taikyti suporuotiems įrašams, o taip pat tam atveju, kai populiacijų dispersijos nėra lygios.
- Dviejų populiacijų Vilkoksono testas `wilcox.test`. Šiuo atveju nulinė hipotezė tvirtina, kad abiejų populiacijų skirstiniai sutampa, o alternatyva – kad skirstiniai skiriasi tik postūmiu (normaliuoju atveju tai reikštų, kad nesutampa vidurkiai).
- Kolmogorovo ir Smirnovovo dviejų populiacijų kriterijus `ks.test`. Nulinė hipotezė teigia, kad abi populiacijos turi tą patį (tolydų) skirstinį.

10.1. Hipotezės apie proporcijas

Diske R1 (žr. `Data\Verbeek\Verbeek_Data\DataSets`) yra failas `benefits` (žr. www.econ.kuleuven.ac.be/GME), kuriame pateikti 4877 įrašai apie netekusius darbo darbininkus JAV 1982-1991 metais. Be kitų duomenų, jame pateikti faktai apie darbininkų odos spalvą (fiktyvus kintamasis `nwhite` lygus 1, jei ne baltasis, ir 0 – jei baltasis) ir ar gavo bedarbystės pašalpą (fiktyvus kintamasis `y` lygus 1, jei gavo). Kadangi (skaitinių) kintamųjų faile `benefits` yra net 22, o vienas įrašas užima keturias eilutes, todėl importuoti šį failą į R šį kartą reikia ne su `read.table`:

```
> benefits <- scan("benefits.dat", what=list(rep(0,22)), multi.line=T)
> matrix(unlist(benefits), byrow=T, ncol=22)[1:5, ]
      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10] [,11] [,12] [,13]
[1,]  4.5  167   42  49 240.1  21   0   0   0   0   0   1   1
[2,] 10.5  251   55  26  67.6   2   1   0   0   0   0   1   1
[3,]  7.2  260   21  40 160.0  19   0   0   0   0   1   0   1
[4,]  5.8  245   56  51 260.1  17   1   0   0   1   0   0   1
[5,]  6.5  125   58  33 108.9   1   1   0   0   0   1   1   1
      [,14] [,15] [,16] [,17] [,18]      [,19]      [,20] [,21] [,22]
[1,]      1      0      0      0      7 0.2906310 0.0844664      1      1
[2,]      1      0      1      1     10 0.5202020 0.2706102      1      0
[3,]      1      1      0      0     10 0.4324895 0.1870471      1      1
[4,]      1      1      0      0     10 0.5000000 0.2500000      0      1
[5,]      1      1      1      1      4 0.3906250 0.1525879      1      0
```

Kintamasis `nwhite` yra įrašytas 10-ajame šios matricos stulpelyje, o `y` – 22-ajame:

```
> nwhite <- matrix(unlist(benefits),byrow=T,ncol=22)[,10]
> y <- matrix(unlist(benefits),byrow=T,ncol=22)[,22]
> benef <- data.frame(nwhite,y)
> rm(nwhite,y,benefits)
> attach(benef)
> tapply(y,nwhite,mean) # Atskirai skaičiuoja y vidurki baltųjų
                        0      1      # (nwhite==0) ir nebalėtųjų (nwhite==1)
0.6821351 0.6935933      # grupėse
```

Kintamasis `y` įgyja tik dvi reikšmes, todėl jo teorinis vidurkis yra ne kas kita kaip vienetukų (t.y., gavusiųjų pašalpą) dalis visoje populiacijoje. Matome, kad šių vidurkių įverčiai abiejose ėminiuose beveik lygūs: $(0,682) = \hat{p}_0 \approx \hat{p}_1 (= 0,694)$, kitais žodžiais labai panašu, kad visoje populiacijoje pašalpą gavusių baltųjų dalis p_0 yra lygi pašalpą gavusių nebalėtųjų daliai p_1 . Hipotezę $H_0: p_0 = p_1$ galima patikrinti keliais būdais.

1. Taikysime proporcijų lygybę tikrinančią testą `prop.test`. Į jį kreiptis galima įvairiai, mes pradėsime nuo požymių sąveikos lentelės skaičiavimo.

```
> tableb <- table(benef)
> tableb
      y
nwhite 0      1
      0 1322 2837 # Langelyje (0,0) yra negavusių pašalpos baltųjų
      1   220  498 # skaičius 1322, langelyje (0,1) – pašalpą gavusių
                  # baltųjų skaičius 2837 ir t.t.

> prop.test(cbind(tableb[,2],tableb[,1])) # Alternatyva  $H_1: p_0 \neq p_1$ 

      2-sample test for equality of proportions with continuity
correction

data:  cbind(tableb[, 2], tableb[, 1])
X-squared = 0.3207, df = 1, p-value = 0.5712
alternative hypothesis: two.sided
95 percent confidence interval:
 -0.04884403  0.02592766
sample estimates:
  prop 1    prop 2
0.6821351 0.6935933
```

Šio kriterijaus p reikšmė lygi 0,5712, todėl nėra jokio pagrindo atmesti nulinę hipotezę. Tą pačią išvadą galime padaryti ir kitaip: proporcijų skirtumo $p_0 - p_1$ 95% pasiklovimo intervalas $(-0,0488; 0,0259)$ uždengia nulį, todėl su 5% (ar bet kuriuo reikšmingesniu¹) lygmeniu hipotezę H_0 priimame.

2. Nulinę hipotezę galima interpretuoti kaip hipotezę, kad abiejose populiacijose pašalpą gavusiųjų dalys yra lygios. Šios hipotezės (apie požymio homogeniškumą) tikrinimas pagristas χ^2 statistika [ČM, (3.5.9)]

¹ Jei pasikliauties lygmuo yra 95%, tai atitinkamos hipotezės reikšmingumo lygmuo lygus $(100-95)\%=5\%$; kuo šis skaičius mažesnis, tuo reikšmingumo lygmuo “aukštesnis”.

$$\chi^2 = \sum_{i=1}^2 \sum_{j=1}^2 \frac{(o_{ij} - n_i \cdot n_j / n)^2}{n_i \cdot n_j / n},$$

kuri, kai hipotezė H_0 teisinga, neturėtų būti “didelė”. Kadangi ši statistika, kai H_0 teisinga ir n didelis, turi $\chi^2_{(2-1)(2-1)} = \chi^2_1$ skirstinį, reikia apskaičiuoti tikimybę $P(\chi^2_1 > \chi^2)$ - jei ji mažesnė už 0,05, H_0 atmetame. χ^2 reikšmę pirmiausiai apskaičiuosime “rankomis”:

```
> simple.marginals(tableb)
      0      1 Total
0     1322 2837 4159
1       220  498  718
Total 1542 3335 4877

> (1322-(a11<-4159*1542/4877))^2/a11+(2837-(a12<-4159*3335/4877))^2/
a12+(220-(a21<-718*1542/4877))^2/a21+(498-(a22<-718*3335/4877))^2/a22
[1] 0.3718071
```

(atkreipkite dėmesį į galimybę vienu metu atlikti veiksmus ir kartu sukurti naują objektą: `a11<-4159*1542/4877`). Gautoji reikšmė nesutampa su aukščiau pateiktu skaičiumi 0,3207, tačiau priežastis čia ta, kad `prop.test` naudojo tolydumo korekciją. Jei jos nenaudotume, gautume tiksliai tą pačią reikšmę:

```
> prop.test(cbind(tableb[,2],tableb[,1]),correct=F)

      2-sample test for equality of proportions without continuity
      correction

data:  cbind(tableb[, 2], tableb[, 1])
X-squared = 0.3718, df = 1, p-value = 0.542
alternative hypothesis: two.sided
95 percent confidence interval:
 -0.04802743  0.02511106
sample estimates:
  prop 1    prop 2 
0.6821351 0.6935933
```

Beje, tą patį rezultatą galime gauti ir su

```
> summary(tableb)
Number of cases in table: 4877
Number of factors: 2
Test for independence of all factors:
      Chisq = 0.3718, df = 1, p-value = 0.542
```

Priminsime, kad funkcija `summary` yra bendrinė, todėl priklausomai nuo argumento klasės² jos reikšmė keičiasi. Sąveikos lentelėms ji tikrina požymių nepriklausomumo hipotezę ir skaičiuoja atitinkamą p reikšmę (komentarą apie ryšį tarp nepriklausomumo ir homogeniškumo galite rasti [ČM, 207 p.]). Čia pažymėsime, kad “išoriškai” abi procedūros nesiskiria, skiriasi tik jų interpretacija.

² > class(tableb)
[1] "table"

10.1 UŽDUOTIS. Remdamiesi duomenų rinkinio `bwages` duomenimis (žr. 5.1 skyrelį), patikrinkite ar vyrai ir moterys yra vienodai išsilavinę.

10.2 UŽDUOTIS. 5.2 skyrelyje buvo nagrinėjami kintamųjų `wage` ir `male` tarpusavio santykiai. Ar vyrų ir moterų atlyginimai `wage` (tiksliau - `cut(wage, breaks=c(80, 320, 410, 520, 1920))`) yra vienodi?

10.2. Hipotezės apie požymių nepriklausomumą (dažnių lentelės)

Aptarkime kelis uždavinius, kurie dažnai vadinami dažnių lentelių uždaviniais. Kaip žinia, bendrojo χ^2 kriterijaus (žr. [ČM1, 197+ psl.])

$$\chi^2 = \sum_{k=1}^k \frac{(O_k - E_k)^2}{E_k}$$

(čia O_i yra stebėtieji (=Observed) dažniai, o E_i yra teoriniai arba tikėtini (=Expected) dažniai) taikymų yra labai daug. Tikrinkime hipotezę, kad diskretūs a.d. X ir Y yra nepriklausomi, t.y. $p_{ij} = P(X = x_i, Y = y_j) = P(X = x_i)P(Y = y_j) = p_i q_j$ kokie bebūtų i ir j . Jei stebėjimų duomenys yra pateikti lentelė

	y_1	y_2	...	y_c	Σ
x_1	o_{11}	o_{12}	...	o_{1c}	$n_{1\cdot}$
x_2	o_{21}	o_{22}	...	o_{2c}	$n_{2\cdot}$
...
x_r	o_{r1}	o_{r2}	...	o_{rc}	$n_{r\cdot}$
Σ	$n_{\cdot 1}$	$n_{\cdot 2}$...	$n_{\cdot c}$	n

tai χ^2 statistika tuomet atrodo taip:

$$\chi^2 = \sum_{i=1}^r \sum_{j=1}^c \frac{(O_{ij} - n_i \cdot n_{\cdot j} / n)^2}{n_i \cdot n_{\cdot j} / n}$$

Jei skirtumai tarp stebėtųjų reikšmių O_{ij} ir teorinių reikšmių $n_i \cdot n_{\cdot j} / n$ yra dideli, tiksliau, jei apskaičiuotoji χ^2 statistikos reikšmė yra didesnė už a.d. $\chi_{(r-1)(c-1)}^2 (1-\alpha)$ eilės kvantilį (čia α yra kriterijaus reikšmingumo lygmuo), tai hipotezę apie a.d. (požymių) X ir Y nepriklausomumą atmetame.

10.1 pvz. Žemiau esančioje lentelėje yra pateikti garsiųjų Salk'o poliomielioto vakcinacijos tyrimų rezultatai.

	Nesusirgo poli- mielitu	Neparaližuojantis poli- mielitas	Paraližuojantis poli- mielitas	Iš viso
Vakcinuoti	200688	24	33	200745
Placebo ³	201087	27	115	201229
Iš viso	401775	51	148	401974

Aukščiau pateikto pavidalo lentelės yra vadinamos sąveikos lentelėmis (šį kartą joje pateikti dviejų vardinių kintamųjų, vakcinacijos metodo ir ligos eigos, sąveikos rezultatai). Paprastai domimasi ar šie kintamieji yra susieti kokia nors (statistine) priklausomybe, o nulinė hipotezė teigia, kad jie yra nepriklausomi. Šia hipotezę galima tikrinti arba su asimptotiniu chi kvadrato testu `chisq.test` arba su tiksliu Fisher'io testu `fisher.test`. Tais atvejais, kai yra papildomų (stratifikuojančių) kintamųjų, nepriklausomumas tikrinamas su Mantelhaen'o (arba Cochran'o, Mantel'io ir Haenszel'io) testu⁴ `mantelhaen.test` iš `ctest` bibliotekos. Jei sąveikos lentelėse yra poruotų duomenų, reikia taikyti `mcnemar.test`⁵.

Grįžkime prie Salk'o duomenų. Mūsų duomenis reikia pateikti matricos arba dviejų vektorių pavidalu.

```
> salk.matr <- rbind(c(200688,24,33),c(201087,27,115))
> salk.matr
      [,1] [,2] [,3]
[1,] 200688  24  33
[2,] 201087  27 115
> chisq.test(salk.matr)
```

Pearson's Chi-squared test

```
data: salk.matr
X-squared = 45.4224, df = 2, p-value = 1.370e-10
```

Kadangi p reikšmė yra ženkliai mažesnė už 0,05, vakcinacija neabejotinai turi įtakos⁶ ligos eigai (pažymėsime, kad tikslus Fisher'io kriterijus dėl didelio įrašų skaičiaus šiuo atveju negali būti taikomas).

10.2 pvz. Viena teorijų teigia, kad “kai kurie žmonės jau gimsta nusikaltėliais”, kitaip sakant, polinkį nusikalsti nusako genai. Žemiau pateiktoje (žr. [LL1, 241 psl.] ir [LL1, 376 psl.]) lentelėje yra duomenys apie brolių dvynių nusikaltimus. Nulinė hipotezė teigia, kad du požymiai, giminytės ryšys ir brolio teistumas, yra nepriklausomi.

	Brolis yra teistas	Brolis nėra teistas	
Vieno kiaušinėlio dvyniai ⁷	10	3	13

³ Placebo = “netikra” vakcina, neutrali medžiaga.

⁴ Žr. pavyzdį [Splus6_1, 192 psl.]

⁵ Žr. pavyzdį [Splus6_1, 195 psl.]

⁶ Maža p reikšmė nebūtinai reiškia didelį vakcinacijos efektą, dažnai ji būna maža tiesiog dėl didelio įrašų skaičiaus.

⁷ Tokių brolių genai identiški.

Dviejų kiaušinėlių dvyniai	2	15	17
	12	18	30

```
dvy.matr <- rbind(c(10,3),c(2,15))
```

Kadangi alternatyva šį kartą yra ne bet koks nuokrypis nuo prognozuojamų reikšmių, bet tik ryšį tarp požymių įrodantis, Fisher'io kriterijuje alternatyva pasirinkime "greater":

```
> fisher.test(dvy.matr,alternative="greater")
```

```
Fisher's Exact Test for Count Data
```

```
data: dvy.matr
p-value = 0.0004652
alternative hypothesis: true odds ratio is greater than 1
95 percent confidence interval:
 3.509263      Inf
sample estimates:
odds ratio
 21.30533
```

Taikydami apytikslį chi kvadrato kriterijų, gauname panašų atsakymą (kadangi renk- amės vienpusę alternatyvą, p reikšmę daliname iš 2):

```
> chisq.test(dvy.matr)$p.value/2
[1] 0.0006105493
```

Taigi, kokį kriterijų betaikytume, nulinę hipotezę reikia neabejotinai atmesti.

10.3 pvz. Ar didėja žmogaus svoris su amžiumi? Į šį klausimą atsakyti nėra lengva – tam reikėtų daugiamečių stebėjimų. Norėdami pailustruoti metodus, modeliukime du stulpelius: ob (=obesity=nutukimas: 1-nutukęs, 0-ne) ir agegr (=age group=amžiaus grupė: 1-jauniausi, ..., 11-vyriausi). Tarkime, kad nutukimo tikimybė didėja su amžiumi. Štai vienas iš galimų modeliavimo (ir **tikimybių pasirinkimo**) vari- antų:

```
agegr <- sample(1:11,200,repl=T)
ob <- numeric(200)
for(i in 1:200) ob[i] <- sample(1:0,1,prob=c(pr <- agegr[i]/12,
1-pr)) # Didėjant amžiaus grupei, 1-tuko tikimybė didėja
mytable1 <- table(ob,agegr)
mytable1
```

```
      agegr
ob    1  2  3  4  5  6  7  8  9 10 11
 0   13 12 20 13 10 11 6  6  5  1  7
 1    0  3  5  6  9  5  9 13 11 15 20
```

Nulinę hipotezę "svoris nepriklauso nuo amžiaus" galima patikrinti ir su summary funkcija (kai jos argumentas – dvimatė lentelė, atsakymas praktiškai sutampa su chisq.test rezultatu (patikrinkite)).

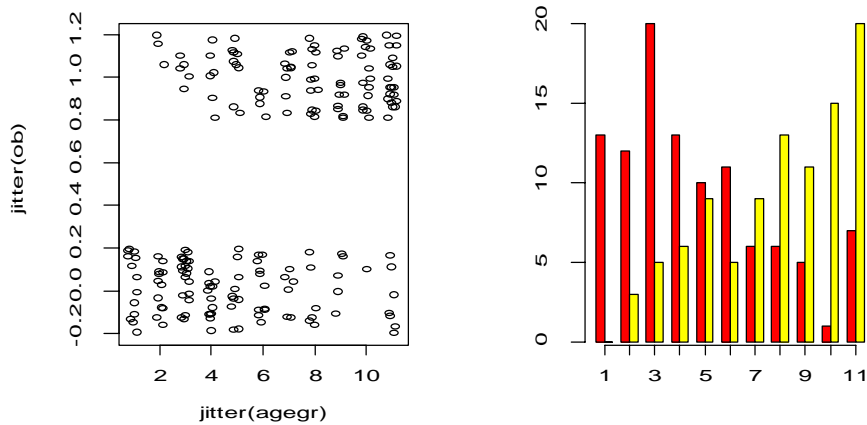
```
summary(mytable1)
Number of cases in table: 200
Number of factors: 2
Test for independence of all factors:
    Chisq = 55.99, df = 10, p-value = 2.062e-08
```

Dar vienas variantas:

```
> mytable2 <- xtabs(~ob+agegr)
> mytable2
  agegr
ob  1  2  3  4  5  6  7  8  9 10 11
  0 13 12 20 13 10 11 6  6  5  7
  1  0  3  5  6  9  5  9 13 11 15 20
> summary(mytable2) # Bus kaip mytable1
Call: xtabs(formula = ~ob + agegr)
Number of cases in table: 200
Number of factors: 2
Test for independence of all factors:
    Chisq = 55.99, df = 10, p-value = 2.062e-08
```

Kaip visuomet, grafikai irgi naudingi:

```
par(mfrow=1:2)
plot(jitter(agegr), jitter(ob))
barplot(mytable2, beside=TRUE)
```



10.1 pav. Tarp senesnių žmonių nutukusių žmonių skaičius didesnis

10.3 UŽDUOTIS. Atsakykite į klausimą, pateiktą 1 skyriaus 1.4 pavyzdyje.

10.3. Hipotezės apie vidurkius

Štai klasikiniai Student'o duomenys apie papildomas miego valandas (buvo tiriamos dvi grupės po 10 žmonių, grupės vartojo du skirtingus migdomuosius vaistus).

```
> data(sleep)
> sleep
```

```

      extra group
1      0.7      1
2     -1.6      1
3     -0.2      1
4     -1.2      1
5     -0.1      1
6      3.4      1
7      3.7      1
8      0.8      1
9      0.0      1
10     2.0      1
11     1.9      2
12     0.8      2
13     1.1      2
14     0.1      2
15    -0.1      2
16     4.4      2
17     5.5      2
18     1.6      2
19     4.6      2
20     3.4      2

```

Kadangi kiekvienoje grupėje duomenys beveik normalūs:

```

> attach(sleep)
> shapiro.test(extra[group==1])
      Shapiro-Wilk normality test
data:  extra[group == 1]
W = 0.9258, p-value = 0.4079

> shapiro.test(extra[group==2])
      Shapiro-Wilk normality test
data:  extra[group == 2]
W = 0.9193, p-value = 0.3511,

```

o jų dispersijos lygios⁸

```

>var.test(extra[group == 1], extra[group == 2]) #  $H_0: \sigma_1^2 = \sigma_2^2$ 
F test to compare two variances
data:  extra[group == 1] and extra[group == 2]
F = 0.7983, num df = 9, denom df = 9,
p-value = 0.7427 # p reikšmė >0,05 - nulinės hipotezės neatmetame
alternative hypothesis: true ratio of variances is not equal to 1
95 percent confidence interval:
 0.198297 3.214123 # Pasikliauties intervalas uždengia 1
sample estimates:
ratio of variances
 0.7983426

```

remsimės Student'o dviejų imčių su lygiomis dispersijomis testu:

```

> t.test(extra[group == 1], extra[group == 2],var.equal=T)

      Two Sample t-test

data:  extra[group == 1] and extra[group == 2]

```

⁸ Tikslai formuluotė būtų tokia: stebėjimo duomenys neprieštarauja hipotezei apie populiacijų dispersijų lygybę.


```

t = -1.8608, df = 18, p-value = 0.07919
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -3.3638740  0.2038740
sample estimates:
mean of x mean of y
  0.75      2.33

```

Taigi nors grupių grupių vidurkiai pastebimai skiriasi (atitinkamai, 0,75 ir 2,33 papildomo miego valandos), t statistikos reikšmė (žr. [ČM, (3.4.5)]) $-1,8608$ nėra tiek didelė, kad nulinę hipotezę $H_0: a_1 = a_2$ atmestume⁹ su standartiniu 5% reikšmingumo lygmeniu¹⁰. Antra vertus, “slidi” 7,9% p reikšmė reiškia, kad bandymus geriausia būtų pakartoti su didesniu žmonių skaičiumi. Iš tikrųjų, jei teisinga alternatyva $H_1: a_1 - a_2 = a \neq 0$, tai, pvz., 95% vidurkių skirtumo pasikliauties intervalo formulė (žr. [ČM, 174 p.]

$$(\bar{X} - \bar{Y}) \pm t_{0,025}(2n - 2) \sqrt{2S_p^2/n} \approx a \pm c / \sqrt{n}$$

teigia, kad, kai n pakankamai didelis, 0 šiam intervalui nepriklausys.

Paskutiniam teiginiui pailiustruoti parašysime funkciją `t.testas`, kuri įrodys, kad kai n didelis, galime atskirti net labai artimas hipotezes.

```

t.testas <- function()
{
# Funkcija t.testas
# Vienas ėminys iš N(0,1), o kitas - iš N(0.2,1)
# Ar galima teigti, kad jų vidurkiai skiriasi?
opar <- par(mfrow = c(1, 2))
on.exit(par(opar))
p.reiksme <- numeric(100)
vid.skirtumas <- numeric(100)
for(i in 1:100) {
  imtis1 <- rnorm(i * 10, 0, 1)
  imtis2 <- rnorm(i * 10, 0.2, 1)
  vid.skirtumas[i] <- mean(imtis1) - mean(imtis2)
# Funkcijos t.test(x1,x2) reikšmė - sąrašas; imame jo
# p.value komponentę
  p.reiksme[i] <- t.test(imtis1, imtis2)$p.value
}
  imties.dydis <- (1:100) * 10
  plot(imties.dydis, vid.skirtumas, type = "l")
  lines(imties.dydis, rep(-0.2, 100), lty = 2)
  plot(imties.dydis, p.reiksme, type = "l")
  lines(imties.dydis, rep(0.05, 100), lty = 2)
}

```

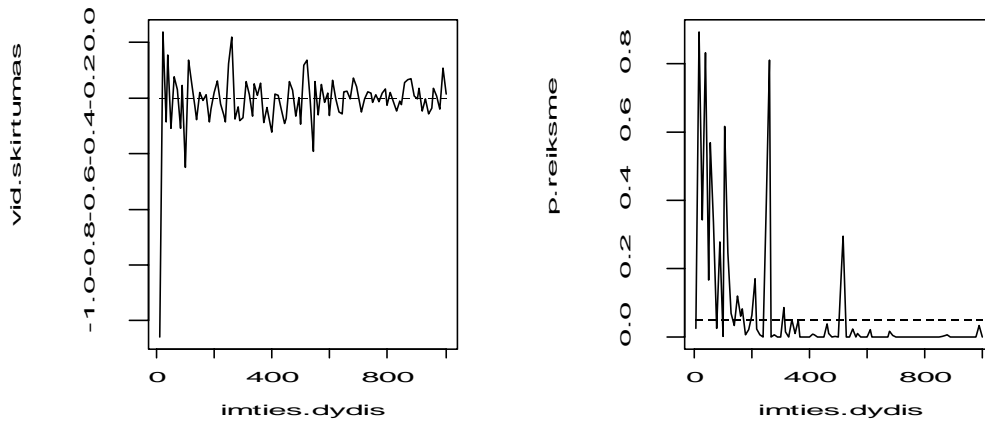
Kelis kartus išbandome savo funkciją (žr. 10.2 pav.):

```
> t.testas()
```

⁹ Abiejose grupėse duomenų nėra daug, todėl šią analizę gerai būtų pakartoti su tinkamu neparametriniu Stjudento testo analogu (žr. 10.6 užduotį).

¹⁰ Atkreipkite dėmesį į 95% vidurkių skirtumo $a_1 - a_2$ pasikliauties intervalą – jis uždengia nulį, kas dar kartą patvirtina, kad atmesti H_0 5% reikšmingumo lygmeniu nėra pagrindo.

```
> t.testas()
```



10.2 pav. Ėminių vidurkių skirtumas artėja į -0.2, hipotezė apie vidurkių lygybę $H_0: a_1 - a_2 = 0$ galų gale bus atmesta, o hipotezė $H_1: a_1 < a_2$ priimta

10.4 UŽDUOTIS. Nagrinėkite du ėminius – vienas iš $N(0;1)$, o kitas iš $N(0;1,2)$. Parašykite funkciją `var.testas` ir įrodykite, kad, kai n pakankamai didelis, hipotezė $H_0: \sigma_1^2 = \sigma_2^2$ bus atmesta.

Iki šiol nagrinėjome atvejį, kai stebėjome dvi nesusijusias populiacijas. Prisiminkime, kad rinkinyje `davis` turėjome duomenis apie tikrąjį ir praneštąjį svorį ir ūgį (tai žmonių, reguliariai užsiimančių fizinėmis pratybomis, svoris ir ūgis; pamatysime, kad jie gana gerai žino savo svorį, bet prasčiau ūgį).

```
> davis[1:5,]
  sex weight height repwt repht
1  M    77    182    77    180
2  F    58    161    51    159
3  F    53    161    54    158
4  M    68    177    70    175
5  F    59    157    59    155
```

Ar teisingai moterys praneša savo svorį? Aišku, kad stulpeliai `weight` ir `repwt` yra priklausomi ir todėl kriterijaus Student'o statistika turės kitokį laisvės laipsnių skaičių. Tokiems uždaviniams spręsti naudojame Student'o **porinį** kriterijų.

```
> attach(davis)
> t.test(weight[sex=="M"], repwt[sex=="M"], paired=T)
```

Paired t-test

```
data: weight[sex == "M"] and repwt[sex == "M"]
t = -2.1294, df = 81, p-value = 0.03625
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -1.13231233 -0.03841937
sample estimates:
mean of the differences
 -0.5853659
```

Nors skirtumas tarp tikrojo ir praneštojo svorio yra nedidelis (tik $-0,585$ kg; praneštasis svoris didesnis), p reikšmė (lygi $0,036$ ($<0,05$) rodo, kad hipotezę apie tai, kad svoris pranešamas visai tiksliai tenka atmesti. Tai nėra keista – kai n didelis, t test fiksuoja net mažus skirtumus. Taigi ateityje, norėdami sužinoti tikslų šių moterų svorį, iš praneštojo svorio reikia (vidutiniškai!) atimti $0,585$ kg (jei nulinės hipotezės nebūtume atmetę, atimti nieko nereikėtų).

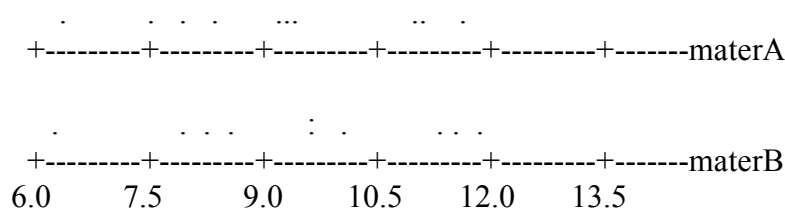
10.5 UŽDUOTIS. Štai ištrauka iš <http://www.anu.edu.au/nceph/surfstat/surfstat-home/> (žr. Statistical inference|Comparing means of two continuous variables):

A shoe company wanted to compare two materials (A and B) for use on the soles of boys' shoes. We could design an experiment to compare the two materials in two ways. One design might be to recruit ten boys (or more if our budget allowed) and give five of the boys shoes with material A and give five boys shoes with material B. Then after a suitable length of time, say three months, we could measure the wear on each boy's shoes. This would lead to independent samples. Now, you would expect a certain variability among ten boys - some boys wear out shoes much faster than others. A problem arises if this variability is large. It might completely hide an important difference between the two materials.

An alternative design, a paired design, attempts to remove some of this variability from the analysis so we can see more clearly any differences between the materials we are studying. Again, suppose we started with the same ten boys, but this time had each boy test both materials. There are several ways we could do this. Each boy could wear material A for three months, then material B for a second three months. Or we could give each boy a special pair of shoes with the sole on one shoe made from material A and the other from material B. This latter procedure produced the data in the table below:

Boy	Material A	Material B
1	13.2	14.0
2	8.2	8.8
3	10.9	11.2
4	14.3	14.2
5	10.7	11.8
6	6.6	6.4
7	9.5	9.8
8	10.8	11.3
9	8.8	9.3
10	13.3	13.6

MINITAB (toks statistikos paketas) pateikia tokius grafinį ir skaitinį abiejų ėminių aprašymą:



	N	MEAN	MEDIAN	TRMEAN	STDEV	SEMEAN
materA	10	0.630	0.750	10.675	2.451	0.775
materB	10	1.040	11.250	11.225	2.518	0.796

	MIN	MAX	Q1	Q3
materA	6.600	14.300	8.650	13.225
materB	6.400	14.200	9.175	13.700

The first design results in two independent samples, while the second design contains dependent samples. Different methods of analysis are appropriate in each case.

Išbrėžkite panašų grafiką ir pateikite panašiai suformatuotas skaitines charakteristikas. Patikrinkite hipotezę apie tai, kad abi medžiagos yra vienodai atsparios dilimui.

10.4. Hipotezės apie “centrų” lygybę

9 skyriuje nagrinėjome `bwages` duomenų rinkinį ir, remdamiesi Wilcoxon'o (rangų ženklų) kriterijumi, tikrinome hipotezę $H_0 : med(w0) = 422,8220 (= med(w1))$ su alternatyva $H_1 : med(w0) < 422,8220$. Panašias (bet ne tapačias) hipotezes galima suformuluoti taip: $H_0 : med(w0) = med(w1)$ su alternatyva $H_1 : med(w0) < med(w1)$. Taikydami Wilcoxon'o (rangų sumų) kriterijų dviems ėminiams, gauname

```
> wilcox.test(w0,w1)

Wilcoxon rank sum test with continuity correction

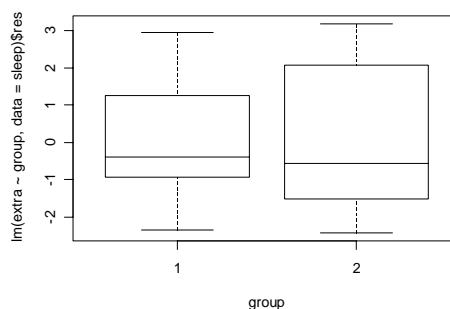
data: w0 and w1
W = 214892, p-value = 4.33e-08
alternative hypothesis: true mu is not equal to 0,
```

taigi nulinę hipotezę vėl neabejotinai atmetame.

10.6 UŽDUOTIS. Duomenų rinkinyje `sleep` kiekvienoje grupėje buvo tik po 10 įrašų. Antra vertus, 10.3 paveiksle matome, kad tiesinio modelio likučiai kažin ar tenkina klasikinį tiesinio modelio reikalavimus.

```
> plot(sleep.lin1$res~group)
```

Visa tai reiškia, kad hipotezę apie papildomo miego trukmės lygybę abiejose grupėse tikslinga pakartoti, naudojant neparametrinį Wilcoxon'o kriterijų. Šią analizę atlikite su `wilcox.test` ir su `wilcox.exact` iš `exactRankTests` paketo.



10.3 pav. Modelio paklaidos grupėse nesimetriškos, dispersijos nevienodos

10.4 pvz. 1 skyriaus 1.3 pavyzdyje buvome suformulavę tokį uždavinį. Marketingo padalinys nori išsiaiškinti, kuris iš dviejų naujų gaminių bus labiau perkamas. Kadangi skonis yra pakankamai subjektyvus kriterijus, potencialūs pirkėjai buvo prašomi įvertinti abu gaminius skaičiais nuo 0 iki 4 (jų prasmė paaiškinta lentelėje):

0	1	2	3	4
Labai nepatiko	Nepatiko	Neturiu nuo- monės	Patiko	Labai patiko

Pirmąjį gaminį vertino $n = 15$, o antrąjį gaminį – $m = 15$ pirkėjų, štai jų atsakymai:
 pirmojo gaminio įverčiai: $x \leftarrow c(3, 2, 3, 0, 1, 0, 2, 1, 0, 0, 4, 3, 2, 0, 3)$
 antrojo gaminio įverčiai: $y \leftarrow c(4, 2, 2, 0, 3, 3, 1, 2, 2, 4, 3, 2, 3, 2, 2)$

Atrodo, kad antrasis gaminy vertinamas geriau, tačiau kaip šitai (jei tai tiesa) pagrįsti? Vienas iš būdų yra toks - apjunkime abu ėminius į vieną ir išdėstykite įverčius didėjimo tvarka: 0 0 0 0 0 0 1 1 1 2 2 2 2 2 2 2 2 2 2 2 3 3 3 3 3 3 3 3 4 4 4. Jei pirmasis gaminy būtų tiek pat geras kaip ir antrasis, tai pirmojo gaminio įverčiai (jie pabraukti) būtų daugmaž tolygiai išsidėstę apjungtame ėminyje. Matome, kad daugiau pabrauktų įverčių yra kairėje, todėl matyt geresnis yra antrasis produktas. Šį teiginį galima patikrinti tokiais samprotavimais. Kiekvienam įverčiui priskirkime jo rangą, t.y., vietą bendroje sekoje. Kadangi 0 kartojasi šešis kartus, reikšmei 0 suteikime vidutinį rangą $(1+2+3+4+5+6)/6=3,5$. Reikšmės 1 vidutinis rangas yra 8; 2 – 14,5; 3 – 23,5 ir 4 – 29. Vadinasi, pirmojo gaminio rangų suma yra $S=5*3,5+2*8+3*14,5+4*23,5+1*29=200$. Formaliai žiūrint (žr. [Ku, 346 p.]), Wilcoxon'o rangų sumų (kitaip – Mann'o ir Whitney'o) testas yra skirtas tolydiesiems a.d. ir tikrina hipotezę, kad "a.d. X ir Y skirstiniai yra vienodi" su alternatyva "skirstiniai skiriasi". Iš tikrųjų, šio testo taikymas kiek siauresnis – alternatyva yra "skirstiniai skiriasi postūmiu"¹¹. Jei teisinga nulinė hipotezė, galima įrodyti, jog nuokrypio statistikos

$$W = S - \frac{n(n+1)}{2}$$

skirstinys nepriklauso nuo (bendro a. dydžių X ir Y) skirstinio, o priklauso tik nuo "laisvės laipsnių" skaičių n ir m. Deja, mūsų atveju, X ir Y nėra tolydūs, be to yra pasikartojančių reikšmių, todėl remsimės tuo faktu, kad a.d.

$$\frac{S - (m/2)(m+n+1)}{\sqrt{(mn/12)(m+n+1)}}$$

turi maždaug standartinį normalųjį skirstinį¹². Kaip ten bebūtų, R pakete kreipiamės į funkciją `wilcox.test`:

```
> wilcox.test(x,y)
Wilcoxon rank sum test with continuity correction
W = 80, p-value = 0.1699 # Daugiau už 0,05!
alternative hypothesis: true mu is not equal to 0
```

¹¹ Kitaip sakant, "centrai skiriasi".

¹² Mūsų atveju kai kurie įverčiai kartojasi, todėl reikšinių po šaknies ženklų reikia pakoreguoti. Vieną korekciją galima rasti [LL2, 130 psl.], kitą – funkcijos `wilcox.test.default` tekste.

Komputavimo rezultatas kiek netikėtas: atmesti hipotezę apie skirstinių¹³ lygybę pagrindo nėra!

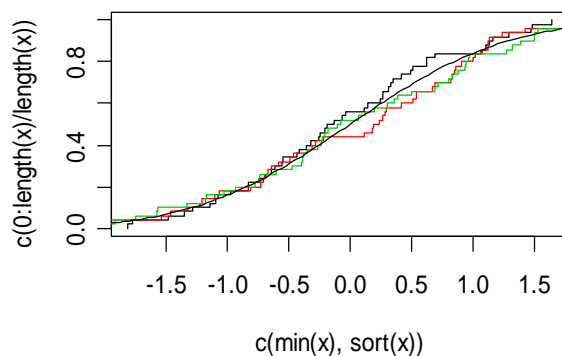
10.5. Hipotezės apie skirstinių lygybę

Tarkime, kad, stebėdami a.d. X , gavome ėminių x_1, x_2, \dots, x_n , o stebėdami a.d. Y – ėminių y_1, y_2, \dots, y_m . Remdamiesi šiais ėminiais, norime nustatyti, ar X ir Y skirstiniai (atitinkamai, F ir G) sutampa (tuomet, pvz., galėtume apjungti abu ėminius į vieną). Hipotezę $H_0 : F \equiv G$ galima tikrinti keliais būdais, o būdo pasirinkimas priklauso nuo alternatyvos. Jei alternatyvioji hipotezė yra $H_1 : F(x) \neq G(x)$ bent vienam x , taikysime Kolmogorovo-Smirnovo dviejų ėminių testą (jis taikomas, kai abi skirstinio funkcijos tolydžios). Jei $H_1 : F(x) = G(x - \delta)$, $\delta \neq 0$, taigi skirstinio funkcijos skiriasi tik postūmiu, taikysime Wilcoxon'o rangų sumos testą `wilcox.test` iš `ctest` arba, kai ėminių dydžiai nėra dideli, `wilcox.exact` iš `exacRankTests` (stebimieji dydžiai dabar gali būti ir ranginiai). Tuo atveju, kai tikrinama "sukeičiamumo" (exchangeability) galimybė, taikysime keitinių testą `perm.test` iš `exacRankTests` paketo (stebimieji dydžiai turi įgyti sveikas reikšmes).

Štai dvi paprastos funkcijos, kurios brėžia empirinės skirstinio funkcijos grafikus ("gražesnis" variantas yra `stepfun` pakete):

```
plot.sdf <- function(x) plot(c(min(x),sort(x)),c(0:length(x)/length(x)),type="s")
lines.sdf <- function(x,col=2) lines(c(min(x),sort(x)),c(0:length(x)/length(x)),type="s",col=col)

plot.sdf(rnorm(50)) # Pirmoji empirinė skirstinio funkcija
lines.sdf(rnorm(50)) # Antroji empirinė skirstinio funkcija
lines.sdf(rnorm(50),col=3) # Trečioji empirinė skirstinio funkcija
curve(pnorm,-2,2,add=T) # Dar viena funkcija grafikams brėžti
```

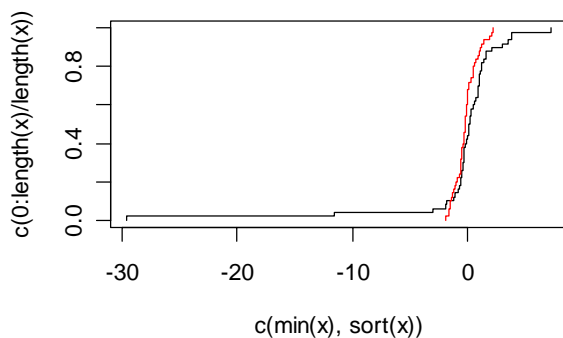


10.4 pav. Trijų ėminių iš standartinės normaliosios populiacijos empirinės skirstinio funkcijos (laiptuotos kreivės) ir standartinė normalioji skirstinio funkcija (glodi kreivė)

¹³ Tiksliau sakant, atsakymą formuluojame taip: abiejų produktų vertinimai vidutiniškai (tiksliau, "medianiškai") vienodi.

Išbandykime KS testą su dviem nedidelėmis imtimis.

```
set.seed(1)
rst <- rt(50,2) # Generuojame Stjudento su 2 l.l. a. skaičius
rn <- rnorm(50)
plot.sdf(rst)
lines.sdf(rn)
```



10.5 pav. Standartinio normaliojo ir Stjudento ėminių empirinės skirstinio funkcijos (abiejų ėminių dydis 50)

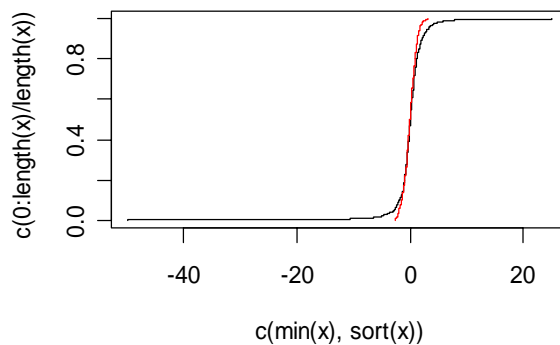
```
> ks.test(rst,rn)

Two-sample Kolmogorov-Smirnov test

data:  rst and rn
D = 0.14, p-value = 0.7166
alternative hypothesis: two.sided
```

Taigi, KS testas nemato pagrindo atmesti hipotezę apie skirstinių lygybę. Antra vertus, kai ėminiai didesni, skirtumai išryškėja.

```
rst <- rt(500,2)
rn <- rnorm(500)
plot.sdf(rst)
lines.sdf(rn)
```



10.6 pav. Standartinio normaliojo ir Stjudento ėminių empirinės skirstinio funkcijos (abiejų ėminių dydis 500)

```
ks.test(rst,rn)
```

Two-sample Kolmogorov-Smirnov test

```
data: rst and rn
D = 0.08, p-value = 0.08152
alternative hypothesis: two.sided
```

```
> wilcox.test(rst,rn)
```

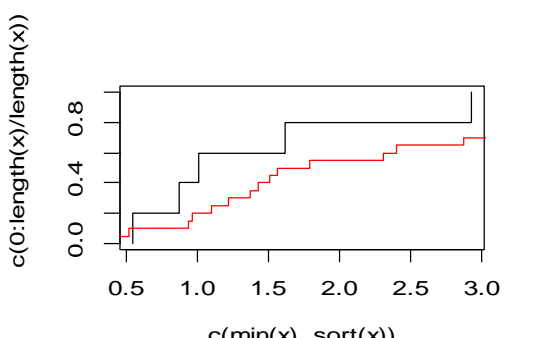
Wilcoxon rank sum test with continuity correction

```
data: rst and rn
W = 119867, p-value = 0.2611
alternative hypothesis: true mu is not equal to 0
```

Matome, kad KS testas beveik atmata hipotezė apie skirstinių lygybę, tuo tarpu kai Vilkoksono testas to nesiūlo. Antra vertus, jei du ėminiai skiriasi tik postūmiu, rezultatai atrodo taip:

Be postūmio

```
set.seed(1)
rg <- rgamma(5,shape=2)
rg1 <- rgamma(20,shape=2)
plot.sdf(rg)
lines.sdf(rg1)
```



```
> ks.test(rg,rg1)

Two-sample Kolmogorov-Smirnov test

data: rg and rg1
D = 0.4, p-value = 0.4906
alternative hypothesis: two.sided

> wilcox.test(rg,rg1)

Wilcoxon rank sum test

data: rg and rg1
W = 32, p-value = 0.2431
alternative hypothesis: true mu is not equal to 0

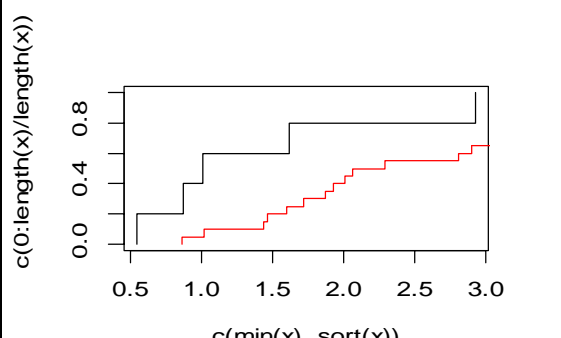
> wilcox.exact(rg,rg1)

Exact Wilcoxon rank sum test

data: rg and rg1
W = 32, p-value = 0.2431
alternative hypothesis: true mu is not equal to 0
```

Su postūmiu 0,5

```
set.seed(1)
rg <- rgamma(5,shape=2)
rg1 <- rgamma(20,shape=2)+0.5
plot.sdf(rg)
lines.sdf(rg1)
```



```
> ks.test(rg,rg1)

Two-sample Kolmogorov-Smirnov test

data: rg and rg1
D = 0.55, p-value = 0.1441
alternative hypothesis: two.sided

> wilcox.test(rg,rg1)

Wilcoxon rank sum test

data: rg and rg1
W = 20, p-value = 0.04235
alternative hypothesis: true mu is not equal to 0

> wilcox.exact(rg,rg1)

Exact Wilcoxon rank sum test

data: rg and rg1
W = 20, p-value = 0.04235
alternative hypothesis: true mu is not equal to 0
```


10.7 UŽDUOTIS. Lentelėje pateikti duomenys apie mieste ir kaime paimtų mėginių užterštumą polichloruotu bifenilu (PCB) (dešimttūkstantosiomis gramu dalimis vienam kilogramui dirvos).

Kaime			Mieste				
3.5	1.0	1.6	12.0	24.0	11.0	107.0	18.0
8.1	5.3	23.0	8.2	29.0	49.0	94.0	12.0
1.8	9.8	1.5		16.0	22.0	141.0	18.0
9.0	15.0	9.7		21.0	13.0	11.0	

a) Miesto ir kaimo duomenis palyginkite grafiškai; b) įrodykite, kad “miesto ir kaimo užterštumas reikšmingai skiriasi”.

10.8 UŽDUOTIS. 6 skyriaus gale, užduočių skyrelyje, yra duomenų rinkinys `drink`. Patikrinkite hipotezes apie kintamojo amount vidurkių lygybę įvairiose sex ir empl grupėse.

10.9 UŽDUOTIS. Funkcija `twoWay` yra dar vienas funkcijos, tiriančios sąveikos lenteles, variantas (plg. 5 skyriaus pradžią).

```
twoWay <- function (x = NA, y = NA, userDefined = NA)
{if (is.na(userDefined)){result <- chisq.test(table(x, y))}
else {result <- chisq.test(userDefined)}
print(result)
observed <- result$observed
expected <- result$expected
chi.table <- ((observed - expected)^2)/expected
row.sum <- apply(observed, 1, sum)
col.sum <- apply(observed, 2, sum)
N <- sum(observed)
fullArray <- cbind(observed, row.sum)
fullArray <- rbind(fullArray, c(col.sum, N))
rownames(fullArray)<- c(rownames(observed), "Total")
colnames(fullArray)<-c(colnames(observed),"Total")
proportion <- fullArray/N
row.proportion <- fullArray/c(row.sum, N)
col.proportion <- t(t(fullArray)/c(col.sum, N))
return(list(fA = fullArray,e = expected,ct = chi.table,
p = proportion, rp = row.proportion, cp = col.proportion))
}
```

```
> twoWay(male,educ)$rp
```

Pearson's Chi-squared test

```
data: table(x, y)
```

```
X-squared = 61.2132, df = 4, p-value = 1.612e-12
```

	1	2	3	4	5	Total
0	0.03972366	0.1208981	0.2797927	0.3316062	0.2279793	1
1	0.08510638	0.2183651	0.2889138	0.1836506	0.2239642	1
Total	0.06725543	0.1800272	0.2853261	0.2418478	0.2255435	1

Išsiaiškinkite šios funkcijos tekstą. Kaip reikia interpretuoti gautąją p reikšmę?

10.10 UŽDUOTIS. Žemiau pateikta ištrauka iš garsiojo Fisher'io (kai kas tvirtina, kad Andersono) duomenų rinkinio apie trijų vilkdalgio rūšių (setosa, versicolor ir virginica) taurėlapio (=sepal) ir vainiklapio (=petal) ilgį ir plotį (cm) (žr. ?iris ir iris).

	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
1	5.1	3.5	1.4	0.2	setosa

50	5.0	3.3	1.4	0.2	setosa
51	7.0	3.2	4.7	1.4	versicolor

100	5.7	2.8	4.1	1.3	versicolor
101	6.3	3.3	6.0	2.5	virginica

150	5.9	3.0	5.1	1.8	virginica

Pateikite grafines (žr. taip pat ?matplotlib) ir skaitines kiekvieno parametro ir jų porų charakteristikas, patikrinkite tinkamas hipotezes.

10.11 UŽDUOTIS. Ledas turi vadinamąją latentinę tirpimo šilumą. Štai dviem metodais gauti rezultatai:

```
Metodas A: 79.98 80.04 80.02 80.04 80.03 80.03 80.04 79.97
80.05 80.03 80.02 80.00 80.02
Metodas B: 80.02 79.94 79.98 79.97 79.97 80.03 79.95 79.97
```

- įveskite šiuos skaičius su `scan`
- ar abu metodai duoda tą patį rezultatą? (pasinaudokite `boxplot`, `t.test`; `shapiro.test`, `var.test` ir `wilcox.test` iš paketo `ctest`).

10.12 UŽDUOTIS. Pakete MASS yra duomenų rinkinys `fgl`, kurį apžiūrėti galima su

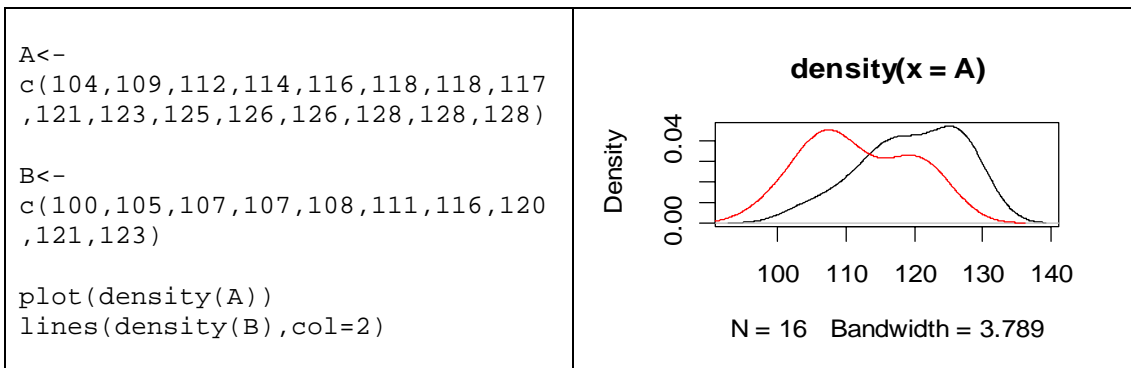
```
library(MASS)
?fgl
```

Išbrėžkite RI, Na ir Mg stačiakampes diagramas visoms `type` grupėms. Patikrinkite hipotezes apie RI "centrų" lygybę visose grupių porose. Pritaikykite `pairs` funkciją grupėje `WinF` ir išbrėžkite grafiką panašų į 6.7 pav.

10.13 UŽDUOTIS. Pakete MASS yra duomenų rinkinys `Aids2`. Ar vienoda šiame sąrašė esančių vyrų ir moterų amžiaus struktūra?

10.14 UŽDUOTIS. Kolmogorovo kriterijus taikomas tuomet, kai stebimieji a.d. yra tolydūs. Antra vertus, matavimai paprastai apvalinimi, todėl netgi tuomet, kai dydžiai "iš tikrųjų" yra tolydūs, gali atsirasti susietų (vienodų) reikšmių (angl. ties). Viena išėitis yra padrebinti (=jitter (angl.)) šiuos dydžius ir tuomet taikyti KS testą.

Žemiau pateikiame duomenis iš Sokal'o ir Rohlf'o knygos "Biometry" (tikslaus duomenų apibrėžimo nėra, bet šio konspekto autoriaus mano, kad tai dviejų rūšių (A ir B) gyvūnų svorio matavimai; kadangi grupėje būna kelių kartų atstovai, tokie skirstiniai paprastai turi kelias modas).



Padreinkite duomenis (pridėdami tolygų atsitiktinių skaičių iš intervalo $[-0,5;+0,5]$) ir patikrinkite hipotezę apie skirstinių lygybę su KS testu. Tą patį atlikite su Wilcoxon'o ir perstatinių testais, taikydami juos pradiniais duomenimis.

10.15 UŽDUOTIS. Kompaktinio disko R1 direktorijoje Data\Misc yra failas iqdata.txt. Jame pateikti du skaitiniai stulpeliai (tai penkiamečių vaikų iq (intelektualumo koeficientas) ir bp (=behavioral problem score=elgesio koeficientas)) ir simbolinis stulpelis ms (=mental state=psichinė būseną: ND=non-depressed mother, D=depressed mother), kuris vaikus suskirsto į dvi grupes. Prašom atsakyti į šiuos klausimus:

- ◆ ar šiose grupėse iq (ir/arba bp) skiriasi?
- ◆ ar šie du kintamieji susiję?

Priminsime, kad Student'o testas taikuomas tuomet, kai stebėjimai

- ◆ skirtingose grupėse yra nepriklausomi;
- ◆ yra gauti stebint normaliuosius a.d.;
- ◆ yra iš populiacijų su vienodomis dispersijomis.

Neparametrinė šio testo alternatyva yra Wilcoxon'o rangų sumų testas.

Taigi, išbrėžkite iq (bp) stačiakampes diagramas, histogramas ir kvantilių-kvantilių grafikus. Ar yra išskirčių? Jei taip, gal geriau jas pašalinti? Ar galima duomenims taikyti Student'o kriterijų? Ar Wilcoxon'o kriterijus duoda tą patį atsakymą? Ką galima pasakyti apie koreliaciją (regresiją) tarp iq ir bp? Ar vaiko elgesio problemos (pvz., $bp > 8$ ar $bp \leq 8$) susijusios su motinos psichine būseną¹⁴?

10.16 UŽDUOTIS. R1 disko Data\Misc direktorijoje yra failas ceramic.txt:

Run	Lab	Batch	Y	X1	X2	X3
1	1	1	608.781	-1	-1	-1
2	1	2	569.670	-1	-1	-1
3	1	1	689.556	-1	-1	-1
4	1	2	747.541	-1	-1	-1
.....						
957	8	1	611.999	1	-1	-1
958	8	2	748.130	1	-1	-1
959	8	1	530.680	1	-1	-1
960	8	2	689.942	1	-1	-1

¹⁴ Taikykite `chisq.test` arba `fisher.test`.

Jame Run yra mėginio numeris, Lab – laboratorijos numeris (nuo 1 iki 8), Batch – partijos numeris (1 arba 2), Y – keraminio mėginio stiprumas. Išstirkite ar skiriasi mėginio stiprumo matavimo rezultatai Y abiejose partijose.

10.17 UŽDUOTIS. Požymių sąveikos lentelės

behaviour =	no	yes
emo=1	51	3
emo=2	69	11
emo=3	28	22
emo=4	7	13

eilutėse yra nurodytas vaiko emocionalumo lygmuo (1-mažas, ..., 4-didelis), o stulpeliuose – ar stebimas nenorminis elgesys (ne/taip). Ar priklausomi šie du požymiai?

10.18 UŽDUOTIS. Žemiau užrašytos matricos taxrevenue.txt eilutėse pateikti valstybės surinkti mokesčiai už nurodytus produktus:

	1990	1991	1992
tobacco	5035.3	5636.0	6289.5
spirits	1513.5	1703.0	1742.1
beer	2074.2	2290.0	2324.9
wine	791.2	855.3	924.5
cider	58.8	68.6	73.8
betting	976.1	1006.4	1052.8

Nusiskaitykite šią matricą su, pvz.,

```
tax <- scan("taxrevenue.txt", list(names="", x=0, y=0, z=0), skip=1)
```

Apibrėžkite metinę prieaugio normą

```
a <- (x-y)/x; b <- (z-y)/y
```

ir patikrinkite hipotezes, kad a “centras” lygus 0,1 (taikykite parametrinį Student’o ir neparametrinį Wilcoxon’o kriterijus). Ar tiesa, kad a ir b “centrai” lygūs? (kokį kriterijų reikia taikyti – poruotiems ar neporuotiems ėminiams?) Ar koreliuoti vektoriai a ir b ? Koks koreliacijos koeficientas labiau tinka – Pearson’o ar Spearman’o? O gal čia geriau taikyti `friedman.test`?

10.19 UŽDUOTIS.

a) Požymių sąveikos lentelės atrodo taip:

i)	<table border="1"><tr><td>25</td><td>24</td></tr><tr><td>26</td><td>25</td></tr></table>	25	24	26	25	ii)	<table border="1"><tr><td>25</td><td>17</td></tr><tr><td>12</td><td>25</td></tr></table>	25	17	12	25	iii)	<table border="1"><tr><td>25</td><td>1</td></tr><tr><td>2</td><td>26</td></tr></table>	25	1	2	26	iv)	<table border="1"><tr><td>25</td><td>26</td></tr><tr><td>2</td><td>1</td></tr></table>	25	26	2	1	v)	<table border="1"><tr><td>25</td><td>3</td></tr><tr><td>26</td><td>2</td></tr></table>	25	3	26	2
25	24																												
26	25																												
25	17																												
12	25																												
25	1																												
2	26																												
25	26																												
2	1																												
25	3																												
26	2																												

Pabandykite atspėti požymių priklausomybės laipsnį visais trimis atvejais.

b) Lentelėje pateikti atsitiktinės apklausos rezultatai (eilutėse – miesto rajonas, kuriame gyvena apklaustoji šeima, stulpeliuose – šeimos pajamos per metus):

	< 12000	tarp 12000 ir 50000	> 50000
A	20	22	3
B	10	9	15

Ar priklauso gyvenamasis rajonas nuo šeimos pajamų?

10.20 UŽDUOTIS. Pasikraukite iš Simple paketo duomenų rinkinį blood (kiekvienas kraujo mėginys vertinamas dviem metodais, Machine ir Expert). Ar skiriasi šie metodai?

10.21 UŽDUOTIS. Žemiau pateiktoje lentelėje yra duomenys iš 1984 General Social Survey of the National Data Program in the United States (Norušis, 1988):

Pajamos (US\$)	Pasitenkinimas darbu			
	Labai nepatenkintas	Nelabai patenkintas	Visai neblogas	Labai patenkintas
<6000	20	24	80	82
6000-15000	22	38	104	125
15000-25000	13	28	81	113
>25000	7	18	54	92

Patikrinkite hipotezę H_0 : *pasitenkinimas darbu nepriklauso nuo pajamų*. Atspausdinkite panašią lentelę, kurioje šalia esamų skaičių dar būtų ir prognozuojamų dažnių reikšmės.

Literatūra

- [Intro] R|Rgui|Help|Manuals|An Introduction to R
- [Ba] Baron J., Li Y. Notes on the use of R for psychology experiments and questionnaires <http://www.psych.upenn.edu/~baron/rpsych.htm>
- [ČM1] Čekanavičius V., Murauskas G. Statistika ir jos taikymai.I. 2001, TEV
- [ČM2] Čekanavičius V., Murauskas G. Statistika ir jos taikymai.II. 2002, TEV
- [Fa] Faraway J.J. Practical Regression and Anova using R, <http://www.stat.lsa.umich.edu/~faraway/book/>
- [Fo] Fox J. Applied regression analysis, linear models , and related methods, 1997 Sage Publications.
- [Kr] Kruopis J. Matematinė statistika. 2-asis leidimas, 1993, Mokslo ir enciklopedijų leidykla, Vilnius.
- [Ku] Kubilius J. Tikimybių teorija ir matematinė statistika. Antrasis leidimas, 1996, Vilniaus universiteto leidykla.
- [Ma] Maindonald J.H. Data Analysis and Graphics Using R – An Introduction <http://www.maths.anu.edu.au/~johnm/>
- [My] Myatt M. Open source solutions – R <http://www.myatt.demon.co.uk>
- [Li] Lindgren B. Statistical theory, Third edition, 1976 Macmillan Publishing Co., Inc. NY, Collier Macmillan Publishers, London.
- [LL1] Справочник по прикладной статистике, т.1, под ред. Э. Ллойда , У. Ледермана, Москва, Финансы и статистика, 1989
- [LL2] Справочник по прикладной статистике, т.2, под ред. Э. Ллойда , У. Ледермана, Москва, Финансы и статистика, 1990
- [Pa] Paradis E. R for beginners <http://cran.r-project.org/>
- [Splus6_1] <http://www.insightful.com/DocumentsLive/23/11/statman1.pdf>
- [Splus6_2] <http://www.insightful.com/DocumentsLive/23/11/statman2.pdf>
- [Ve] Verzani J. simpleR – Using R for Introductory Statistics. <http://www.math.csi.cuny.edu/Statistics/R/simpleR>
- [V&R] Venables W.N., Ripley B.D. Modern Applied Statistics with S-PLUS. Third edition, 1999 Springer-Verlag New York, Inc.