O histogramih

Primož Peterlin

→ **Histogram je grafični pripomoček za prikaz množice meritev zvezne številske spremenljivke. Prispevek predstavi konstrukcijo histograma, kumulativni histogram, razliko med histogramom in stolpičnim diagramom, težave s histogrami, za konec pa se pomudi še ob ocenjevanju porazdelitve z jedri.**

Od podatkov do histograma

Pri fizikalnih meritvah pogosto merimo vrednost ene količine v odvisnosti od druge, npr. napetost na kondenzatorju v odvisnosti od časa. Še preprostejše pa so meritve, pri kateri merimo vrednosti ene same količine, npr. telesno višino učencev v razredu ali število jedrskih razpadov, zaznanih z Geiger-Müllerjevo cevjo. V statistiki takšnim primerom, pri katerih imamo opravka z eno samo spremenljivko (telesna višina, število razpadov v časovni enoti), pravimo univariatni, za razliko od bivariatnih, kadar sta spremenljivki dve, ali v splošnem multivariatnih.

Vsi učenci niso enako visoki in v izbranem časovnem intervalu ne razpade vedno enako število jeder, zato tako zbrani podatki navadno niso vsi enaki. Za zgled smo zbrali rezultate 49-ih učencev dveh paralelk 6. razreda pri skoku v daljavo z mesta; rezultati so podani v centimetrih.

135	168	160	166	130	171	148	152
156	120	176	139	189	115	130	135
140	134	180	180	125	106	141	169
193	129	139	130	165	149	120	148
140	95	150	176	184	159	152	169
185	147	150	190	175	120	149	155
174							

Množica podatkov je nepregledna, zato želimo informacijo strniti in na preprost način predstaviti njene glavne značilnosti. Dva uporabna parametra za opis množice podatkov sta povprečje in standardni odklon. Prvi pove »težišče« podatkov, drugi pa, koliko se podatki med seboj razlikujejo. Standardni odklon je uporabno merilo za opis raznolikosti podatkov, včasih pa želimo vedeti še kaj več o tem, kako so podatki porazdeljeni. Ali obstaja ena takšna vrednost, okrog katere so izmerjene vrednosti posejane pogosteje kot sicer, ali pa morda dve ali celo več takih vrednosti? So vrednosti okoli povprečja posejane simetrično ali niso? Pripomoček, s katerim lahko dobimo približno grafično sliko o porazdelitvi meritev univariatne številske spremenljivke, je *histogram*.

Kako se lotimo priprave histograma? Podatke razvrstimo v *razrede* ali predalčke. V zgornjem zgledu, denimo, zberemo skupaj rezultate med 91 in 100 cm, potem med 101 in 110 cm itd. Zgornjo in spodnjo mejo postavimo tako, da zajamemo vse podatke, širino predalčka pa izberemo tako, da se kar najbolj pokaže oblika porazdelitve. Prevelika širina predalčka bo morda zgladila in skrila kakšno sicer morda zanimivo lastnost porazdelitve, ob premajhni pa bo v posameznem predalčku premalo podatkov in prevladale bodo naključne razlike. Ko bomo končali, bo število v posameznem predalčku govorilo o tem, kako pogosto se določene vrednosti pojavljajo v vzorcu; pogostosti se s tujko pravi tudi *frekvenca* (ki pa s frekvenco pri nihanju in valovanju nima neposredne zveze).

Razvrščanje podatkov v predalčke je na srečo posel, ki ga dobro obvladajo programi za obdelavo podatkov. Za zglede v prispevku bomo uporabili prosti programski paket GNU R (`http://www.r-project.org/`), o katerem smo v Preseku že pisali [2].

```
1  # datoteka z dolžinami skokov
2  x <- scan("skoki.txt")
3
4  hist(x, main=NULL,
5       xlab = "dol\u017Eina (cm)",
6       ylab = "frekvenca")
```

Programček je tako kratek, da o njem skoraj ni kaj povedati. Z ukazom `scan()` v vektor `x` preberemo podatke iz datoteke `skoki.txt`, kjer je v zapisana po ena vrednost v vrstici. Klic funkcije `hist()`, ki ji vektor `x` podamo kot argument, opravi vse ostalo: vrednosti razvrsti v razrede in izriše histogram. Ker imamo podpis pod sliko, se odrečemo naslovu histograma (`main=NULL`), z izbirama `xlab` in `ylab` pa nastavimo oznaki na abscisi in ordinati. V oznakah lahko uporabimo tudi znake izven nabora ASCII; prikličemo jih s kodami ISO10646/Unicode. Rezultat vidimo na sliki 1a. Vidimo, da je `hist()` samodejno izbral povsem razumne meje intervalov. Če z njegovo izbiro ne bi bili zadovoljni, bi lahko z izbiro `breaks=` sami določili meje razredov.

Prikaz frekvence (števila v posameznem podatkovnem razredu) ni edina mogoča predstavitev histograma. Če število v vsakem razredu delimo s številom vseh meritev v vzorcu, dobimo relativne frekvence, npr. $1/49 = 0{,}02$ ipd. Če pa relativno frekvenco delimo še s širino razreda, tak histogram prikazuje *gostoto verjetnosti*. Programček v tem primeru popravimo tako, da funkciji `hist()` dodamo izbiro `probability=TRUE`:

```
# datoteka z dolžinami skokov
x <- scan("skoki.txt")

hist(x, probability=TRUE, main=NULL,
     xlab = "dol\u017Eina (cm)",
     ylab = "gostota verjetnosti")
```

Rezultat je prikazan na sliki 1b. Gostota verjetnosti je definirana tako, da je skupna ploščina vseh stolpcev natanko 1.

Histogram in stolpični diagram

Histogram pogosto zamenjujejo s stolpičnim diagramom. Kljub temu, da so pri obeh podatki predstavljeni s stolpci, pa je razlika med njima precejšnja:

- Stoplpični diagram prikazuje frekvence na diskretni osi *kategorialne* spremenljivke, bodisi nominalne (npr. moški/ženske) bodisi ordinalne (npr. stopnja izobrazbe).
- Histogram je približek porazdelitve po *zvezni* spremenljivki.

Starost	Frekvenca
0–4	28
5–9	46
10–15	58
16	20
17	31
18–19	64
20–24	149
25–59	316
60+	103

TABELA 1.
Udeleženci prometnih nesreč v londonskem okrožju Harrow v letu 1985.

Razlika med obema je morda najbolj očitna v primeru, ko razredi niso enako široki (zgled je izposojen iz učbenika statistike, [2, str. 25]).

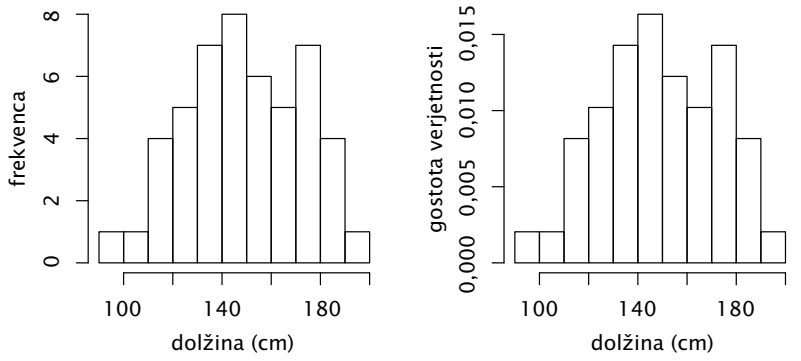
V londonskem okrožju Harrow so za leto 1985 zbrali statistiko udeležencev prometnih nesreč po starosti (tabela 1).

Podatke najprej prikažemo s stolpičnim diagramom.

```
vrednosti <- c(28, 46, 58, 20, 31,
     64, 149, 316, 103)
barplot(vrednosti,
     names.arg = c("0-4", "5-9",
          "10-15", "16", "17",
          "18-19", "20-24", "25-59",
          "60+"),
     xlab = "Starost (leta)",
     ylab = "\u0160tevilo")
```

Programček je spet zelo preprost. Vektor `vrednosti` vsebuje frekvence (desni stolpec v tabeli 2), stolpični diagram pa izrišemo s funkcijo `barplot()`, ki ji podamo ta vektor. Argument `names.arg` je vektor z oznakami za posamezne stolpce, `xlab` in `ylab` pa oznaki osi.

Rezultat programa je stolpični diagram, prikazan na sliki 2. Diagram pravzaprav ne pove več od tabele 1 in ne odraža porazdelitve udeležencev prometnih nesreč po starosti. Vidimo lahko, denimo, da je število odraslih udeležencev prometnih nesreč (torej tistih v starostni skupini 25–59 let) desetkrat



SLIKA 1.
Histogram porazdelitve skokov v dolžino.

tolikšno kot število udeležencev, starih 17 let. Vendar prva skupina zajema dosti večji delež populacije kot druga, zato nam ta podatek sam po sebi ne pove veliko.

Predstavimo zdaj podatke iz tabele 2 kot histogram. Za razliko od primera na sliki 1 so predalčki tu različno široki.

```
meje <- c(0, 5, 10, 16, 17, 18, 20,
      25, 60, 80)
vrednosti <- c(28, 46, 58, 20, 31,
      64, 149, 316, 103)

mojhist <- list(breaks=meje,
      counts=vrednosti,
      density=vrednosti/diff(meje),
      xname=NULL)
class(mojhist) <- "histogram"

plot(mojhist, xlab="Starost (leta)",
ylab="\u0160tevilo na leto starosti",
main=NULL)
```

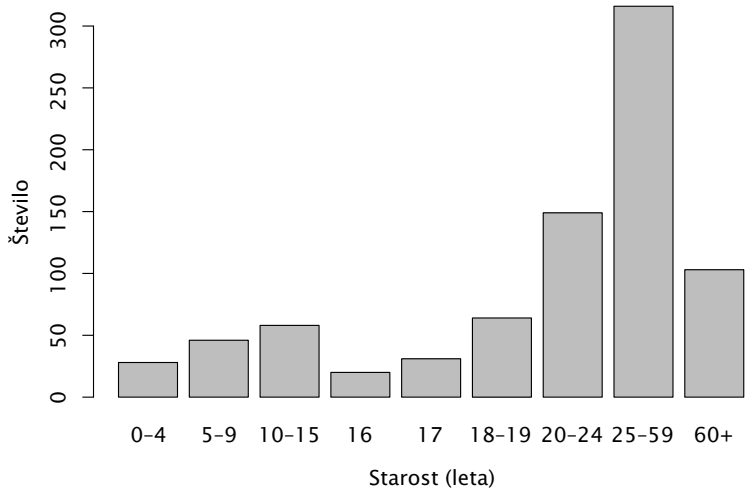
V tem primeru ne potrebujemo klica funkcije `hist()`, ki razvrsti podatke po predalčkih, saj je nekdo to že opravil namesto nas. V vektorju `meje` so shranjene meje razredov (najvišji interval smo omejili na 80 let), v vektorju `vrednosti` pa frekvence posameznih razredov. Zatem sestavimo seznam `mojhist` z elementi `breaks`, kateremu podamo meje; `counts`, kateremu podamo frekvence; `density`, kateremu podamo vrednosti, deljene s širino razreda; (`diff(meje)` vrne vektor razlik med zaporednimi elementi vektorja `meje`, kar so ravno širine razredov); naslova (`xname`) pa ne nastavimo. Potem uporabimo predmetno naravo jezika R in seznamu `mojhist`, ki smo ga ravnokar ustvarili takega, da že ima pravilno strukturo histograma, priredimo razred `histogram`. Končno histogram `mojhist` izrišemo s funkcijo `plot()`. Izbire `main`, `xlab` in `ylab` imajo enak pomen kot prej.

Rezultat je na sliki 3. Diagram se precej razlikuje od tistega na sliki 2. Vidimo lahko, da so mladostniki približno trikrat pogosteje udeleženci prometnih nesreč kot odrasli (ali tudi kot otroci). Slika 3 tudi nazorno pokaže, da je na histogramu frekvenci sorazmerna *ploščina* posameznega stolpca, ne pa njegova višina. Pri razredih enake širine sta ploščina in višina stolpca resda premo sorazmerni, kar lahko zavede.

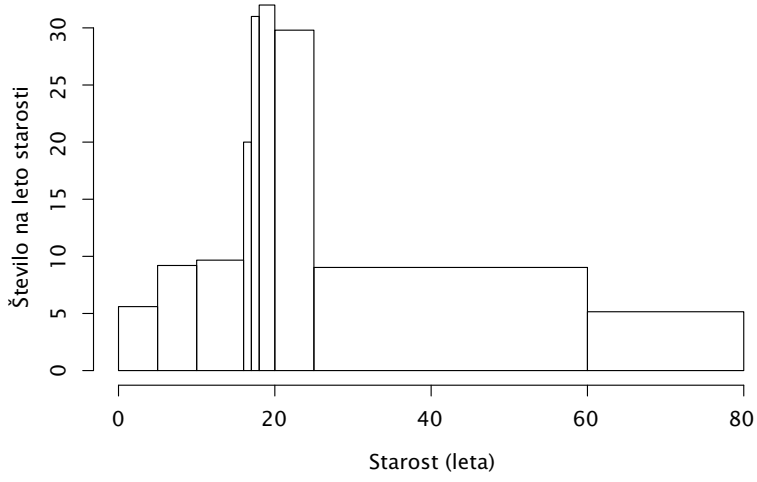
Kumulativni histogram

Iz histograma na sliki 1 lahko neposredno preberemo podatek o frekvenci posameznega razreda, denimo, koliko učencev je skočilo med 140 in 149 cm. Včasih pa nas zanima drugačno vprašanje: denimo, koliko učencev je skočilo manj kot 150 cm. Odgovor lahko seveda izračunamo, tako da seštejemo frekvence v razredih 90–99 cm, 100–109 cm in tako dalje do 140–149 cm. *Kumulativni histogram* (slika 4b) pa omogoča, da tak podatek preberemo neposredno iz diagrama. Poseben primer obrnjenega kumulativnega histograma so tudi krivulje preživetja, ki se uporabljajo v biomedicinskih vedah [4].

```
x <- scan("skoki.txt")
h <- hist(x, plot=FALSE)
h$counts <- cumsum(h$counts)
plot(h, main=NULL,
     xlab = "dol\u017Eina (cm)",
     ylab = "frekvenca")
```

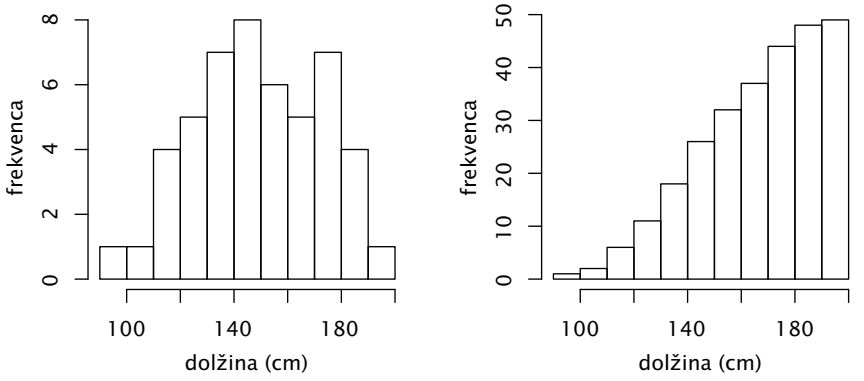


SLIKA 2.
Stolpični diagram števila udeležencev prometnih nesreč po starosti.



SLIKA 3.
Histogram števila udeležencev prometnih nesreč po starosti.

Oglejmo si še, kako v R pripravimo kumulativni histogram. Enako kot prej v vektor `x` preberemo podatke iz datoteke. Dve razliki pa sta pri funkciji `hist()`. Prva je ta, da smo ji podali izbiro `plot=FALSE`, s katero smo zahtevali, da histogram sicer izračuna (določi podatkovne razrede in vanje razvrsti podatke), a ga ne izriše. Druga pa je ta, da smo rezultat izračuna histograma shranili v spremenljivko `h`. Iz zgleda z razredi neenake širine že vemo, da je ta spremenljivka seznam; skladnja `h$counts` nam vrne element seznama `counts`, ki je vektor s frekvencami. V tretji vrstici histogram pretvorimo v kumulativni histogram s klicem funkcije `cumsum()`. Če ji podamo vektor dolžine n, bo vrnila vektor iste dolžine s kumulativnimi vsotami: na prvem mestu bo kar prvi element podanega vektorja na drugem mestu vsota prvih dveh elementov podanega vektorja, in tako dalje do zadnjega elementa, kjer bo vsota vseh elementov podanega vektorja. Vektor s kumulativnimi vsotami shranimo kot element `h$counts` histograma. Tako spremenjen histogram zdaj narišemo z ukazom `plot()`, ki mu podamo izračunani histogram `h`, izbire `main`, `xlab` in `ylab` pa imajo že znani pomen.



SLIKA 4.
Histogram (a) in kumulativni histogram (b) porazdelitve skokov v dolžino.

Težave s histogrami

V prvem zgledu smo nekoliko zlahka odpravili izbiro intervala in določitev števila razredov. Čas je, da priznamo, da sta prav ti dve izbiri srž težav s histogrami. Posebej v primeru, če podatkov ni veliko, je histogram odvisen od izbire teh dveh parametrov.

Oglejmo si najprej zgled, kako na histogram vpliva izhodišče razredov. Levi diagram na sliki 5 že poznamo, pri desnem (slika 5b) pa smo izbrali razrede tako, da zaobjamejo vrednosti 95–104, 105–114 itd. V programčku smo to izvedli z izbiro `breaks=`, pri kateri smo uporabili funkcijo `seq()`. Tej smo podali tri parametre: prvi element, zadnji element in korak; funkcija vrne vektor z zaporedjem, ki sledi podanemu pravilu.

```
1   x <- scan("skoki.txt")
2   hist(x, main=NULL,
3        breaks = seq(95, 205, 10),
4        xlab = "dol\u017Eina (cm)",
5        ylab = "frekvenca")
```

Vnaprej lahko uganemo, da večja širina razreda zgladi histogram, kar lahko vidimo tudi na sliki 6. Levi histogram uporablja privzete meje razredov (`breaks = seq(90, 200, 10)`), desni pa dvakrat tolikšno širino razredov (`breaks = seq(90, 210, 20)`). Problem izbire optimalnega števila razredov so precej preučevali in različni raziskovalci so prišli do različnih formul za optimalno število razredov. Altman kot praktični nasvet navaja [1], da je navadno dovolj 8–15 razredov, razen če je podatkov zelo veliko. Med bolj znanimi so še Sturgesova formula, $k = \lceil \log_2 n + 1 \rceil$, kjer je n število podatkov, k število razredov, $\lceil x \rceil$ pa označuje zaokroževanje navzgor, in Scottova formula $h = 3{,}5\hat{\sigma}/n^{1/3}$, kjer je $\hat{\sigma}$ standardni odklon vzorca, h pa širina razreda.

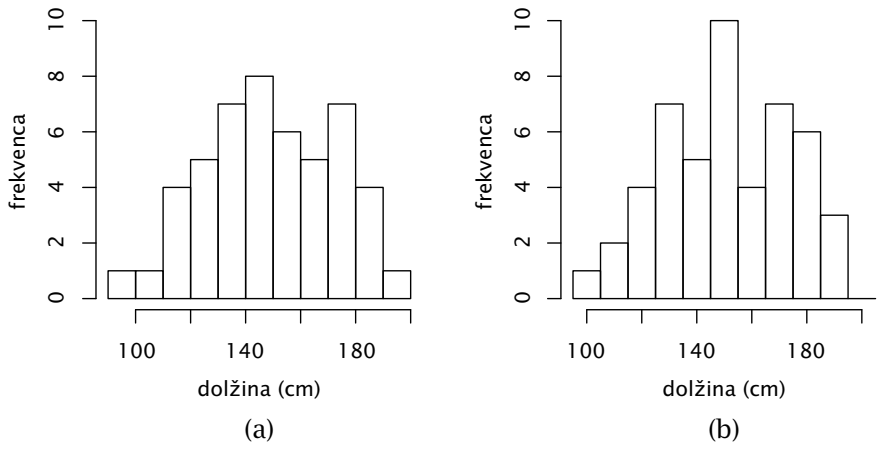
Kateri od histogramov je pravi? Pravega ali najboljšega histograma ni. Ne poznamo postopka, s katerim bi za poljubno porazdelitev vhodnih podatkov izračunali najboljši histogram. Na nas je, da s poskušanjem in spreminjanjem izhodišča in širine razredov izračunamo histogram, ki je sprejemljiv. Na srečo si lahko pomagamo z računalnikom, kar vsakokratno razvrščanje podatkov v razrede napravi skoraj hipno. Zato je histogram kljub naštetim pomanjkljivostim še vedno uporabno orodje za kvalitativno oceno eksperimentalnih porazdelitev.

Ocenjevanje gostote verjetnosti z jedri

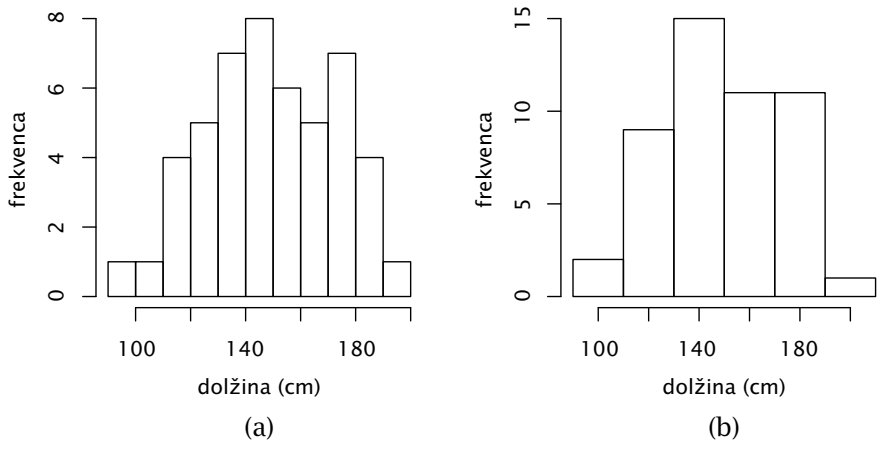
Histogram ima zaradi svoje enostavne konstrukcije in interpretacije zagotovo svoje prednosti, vseeno pa se moramo glede na vse prej omenjene težave s histogrami vprašati, ali ne obstaja kakšen boljši način za oceno porazdelitve v vzorcu, pridobljenem s poskusom. Obstaja. Metoda je poznana kot ocenjevanje gostote z jedri (angl. kernel density estimation). Matematično je znatno bolj zapletena in preobsežna za ta članek. Osnovna zamisel pa je preprosta in jo bomo nakazali.

Za zgled si oglejmo vzorec 12-ih meritev (vrednosti nalašč ni preveč, da je primer preglednejši):

2,064	2,212	2,351	2,409	2,459	2,639
2,656	2,673	3,350	3,373	3,599	3,861



SLIKA 5.
Histogram istih podatkov z enako širokimi razredi in različnim izhodiščem.

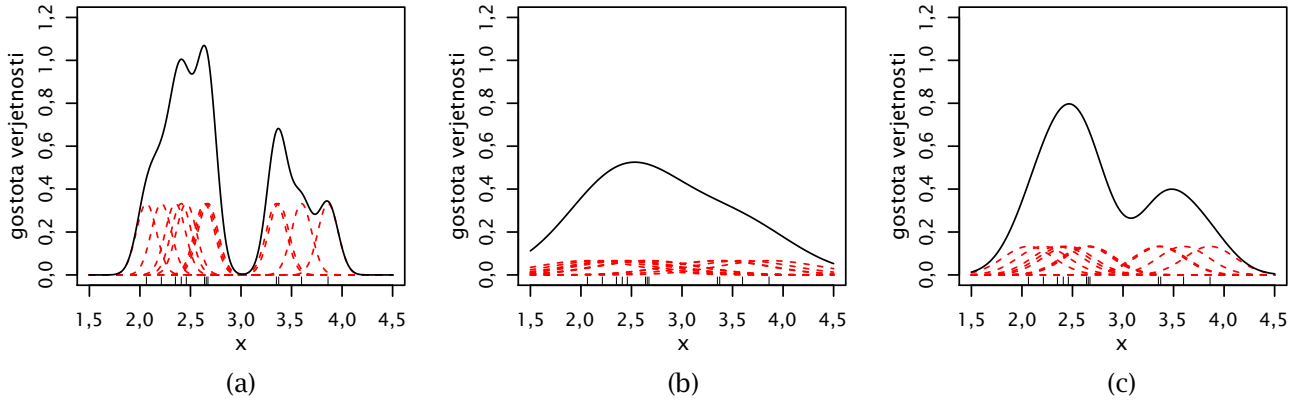


SLIKA 6.
Histogram istih podatkov z različno širokimi razredi.

Ko smo konstruirali histogram, smo najprej izbrali izhodišče in širino razreda – v zgornjem zgledu bi lahko za izhodišče izbrali vrednost 2, za širino razreda pa 0,5. Če rišemo gostoto verjetnosti, vsaki od 12-ih meritev ustreza pravokotnik širine 0,5, kolikor je širina razreda, in višine 1/6, tako da je skupna ploščina vseh pravokotnikov ravno 1. Histogram dobimo tako, da pravokotnike zlagamo v odgovarjajočih razredih drugega vrh drugega kot kocke Lego.

Kot smo videli, je ena od težav s histogrami naša svoboda, da si prosto izbiramo izhodišče histograma. Namesto tega se zdaj lotimo zadeve drugače: pravokotnik, ki pripada posamični meritvi, narišemo tako, da sega za pol širine razreda levo in desno od izmerjene vrednosti. Pri prvi meritvi iz vzorca sta meji pravokotnika tako 1,814 in 2,314. Če se pravokotnika, ki pripadata dvema meritvama, delno prekrivata, oba prispevka seštejemo tako, da narišemo v delu, kjer se prekrivata, pravokotnik dvojne višine. Tako ostaja ploščina dobljenega lika enaka ploščini dveh pravokotnikov.

Ko bi z delom končali, bi dobili histogramu podoben diagram, ki pa bi že lepše prikazal porazdelitev podatkov v vzorcu. Ne bomo ga narisali, ker smo spotoma dobili še boljšo zamisel: namesto s pravokotnikom prispevek vsake meritve v vzorcu opišemo z »gladko« krivuljo, takšno, ki ima vrh pri vrednosti meritve, levo in desno od meritve pa simetrično pada, in to tako, da je ploščina pod krivuljo enaka 1/12. S tem odpravimo še eno težavo histograma, namreč to, da praviloma zvezne porazdelitve prikaže nezvezno stolpičasto. Res pa je, da je seštevanje gladkih krivulj prezamudno, da bi to lahko počeli ročno na milimetrskem papirju. A na srečo si lahko pomagamo z računalnikom.



SLIKA 7.

Ocenjevanje gostote verjetnosti z Gaussovimi jedri; (a) premajhno glajenje, (b) preveliko glajenje, (c) optimalno glajenje. Rdeče črtkane krivulje podajajo prispevke posameznih jeder, črna neprekinjena črta pa na osnovi jeder dobljeno gostoto verjetnosti. Črtice na notranji strani abscisne osi označujejo vrednosti meritev.

V zgledu prispevek vsake posamične meritve opišimo z Gaussovo porazdelitvijo, ki ima vrh pri vrednosti te meritve. Matematični funkciji, s katero opišemo prispevek posamične vrednosti, pravimo *jedro*. Skupno porazdelitev potem dobimo kot seštevek posamičnih prispevkov. Porazdelitev je odvisna tudi od širine Gaussovega jedra, ki jo podaja standardni odklon σ. Na sliki 7 so predstavljene porazdelitve gostote verjetnosti, dobljene z Gaussovimi jedri z različnimi vrednostmi σ: $\sigma = 0{,}1$ (a), $\sigma = 0{,}5$ (b) in $\sigma = 0{,}25$ (c). V R oceno gostote z jedri izračuna fukcija `density()`.

Na sliki 7 vidimo, da širina jedra močno vpliva na oceno gostote verjetnosti. To še ni vse: brez obrazložitve smo za jedro vzeli Gaussovo funkcijo, lahko pa bi tudi kakšno drugo. Smo torej sploh kaj na boljšem kot pri histogramih, ki smo jim očitali preveliko subjektivnost? Malo bolje je vseeno. Načeloma je naloga preprosta: optimalna, z jedri ocenjena porazdelitev je tista, ki se čimbolj ujema s pravo; težava pa je v tem, da prave porazdelitve ne poznamo, ampak bi jo radi šele ugotovili. Kljub vsemu obstajajo postopki, ki v asimptotičnem približku vodijo k optimalni jedrni funkciji in optimalni širini. Ne najnovejši, pa še vedno precej bran učbenik s tega področja je [3], področje pa se še vedno razvija.

Za konec povzamimo dobre in slabe lastnosti obeh pristopov. Histogram je preprosto konstruirati s svinčnikom in milimetrskim papirjem, preprosto ga je interpretirati in kljub težavam z arbitrarnostjo izbire izhodišča in širine razredov večinoma nudi grobo oceno za porazdelitev izmerjenih vrednosti. Ocena gostote z jedri je matematično bolj kompleksna, računsko zahtevnejša, nudi pa nekoliko boljšo oceno porazdelitve. Dostopnost zmogljivih računalnikov ter izvedbe v večini programskih jezikov in paketov pomenita, da je ta metoda, nekoč omejena na raziskovalne laboratorije, dostopna vsakomur. Zato je dobro, da tudi razumemo, kako deluje.

Literatura

[1] D. G. Altman, *Practical statistics for medical research*, London: Chapman & Hall, 1991.

[2] P. Peterlin, *Obdelava meritev in risanje grafov z R*, Presek **37**, 24–30, 2010.

[3] B. W. Silverman, *Density estimation for statistics and data analysis*, London: Chapman & Hall, 1986.

[4] J. Stare, *Krivulje preživetja*, Medicinski razgledi **40**, 173–181, 2001.

×××