

Elektronická podpora pre časť: Štatistické výpočty v MATLABe

(Doplnenia a komentár textu skript, riešenia niektorých príkladov a úloh)

Kap.1 Opisná štatistika jednorozmerného štatistického súboru

- 1.1 Štatistický súbor, náhodný výber, jednorozmerný štatistický súbor
- 1.2 Tabuľky, diagramy, histogramy
- 1.3 Charakteristiky polohy
- 1.4 Charakteristiky variability
- 1.5 Empirická distribučná funkcia, výberové kvantily
- 1.6 Doplnenia a úlohy

Úloha (str. 114) Nech základným súborom je množina všetkých študentov prvého ročníka našej fakulty. Navrhните taký postup výberu, ktorý by sme mohli označiť ako náhodný výber rozsahu 40.

Objasnenie: Zámer úlohy je podnietiť uvažovanie o realizovaní náhodného výberu. Nikto neočakáva, že podáte skutočné riešenie tejto úlohy t.j. také riešenie, ktoré by vyhovovalo nielen z teoretických, ale aj z praktických aspektov. Prvý návrh by mohol vyzeráť takto:

Každý študent prvého ročníka napíše na lístok svoje meno a k nemu pre istotu aj rodné číslo :-). Do škafule dáme všetky lístky, zatrasieme ňou a vyberieme jeden po druhom 40 lístkov (pre dôkladnosť, po každom výbere škafulej dobre zatrasieme). Realizácia výberu končí vybratím 40-teho lístka. Na stole máme 40 rôznych lístkov, pretože vybrané lístky sme – pochopiteľne – späť do škafule nedávali.

Intuitívne je jasné, že sme naplnili obsah požadovaného, pretože

1. Každý prvoročník má rovnakú šancu dostať sa do výberu.
2. Vybratie i -teho študenta neovplyvňuje šancu j -teho študenta dostať sa do výberu.

Prirodzená námietka: praktická realizácia navrhnutého postupu má slabiny – zdĺhavosť a ťažkopádnosť. Dobré vieme, že dnes sa takéto úlohy riešia pomocou počítačov. Ako realizovať náhodný výber na počítači však nie je jednoduché a vecná diskusia o ňom je na mieste až potom, keď sa vysvetlia základy teórie pravdepodobnosti.

To, čo teraz uvedieme, je "nepovinné" a cieľom nasledujúcich riadkov je podporiť uvažovanie a budovať intuíciu. V MATLABe môžeme realizovať náhodný výber rozsahu k z množiny $\{1, 2, \dots, N\}$ využitím funkcie "randperm", ktorá dáva náhodnú permutáciu $[i_1, i_2, \dots, i_N]$ základného poradia $[1, 2, \dots, N]$. Zo získaného vektora vezmeme prvých k zložiek. Napr. nech $N = 9, k = 5$.

N = 9; k = 5; nahod = randperm(N), vyber = nahod(1:k)

```
nahod =
     7     5     4     6     1     8     3     2     9
vyber =
     7     5     4     6     1
```

•

Zmysel nasledujúceho príkladu je ukázať na jednoduchej situácii, čo je *štandardný* tvar dát. Zatiaľ v tomto článku ide o viacrozmerné dáta, čo znamená, že na jednotke výberu nás zaujímajú dva, resp. viac znakov. Pre objasnenie:

Príklad (str. 114), "študent-pohlavie-krv.skupina-vek-výška-hmotnosť" sa týka situácie, v ktorej na jednotke výberu sme merali 5 znakov: pohlavie, krv.skupina, vek, výška, hmotnosť. Nasledujúci príklad ("učiteľov záznam o krúžku") sa týka situácie, v ktorej ide o 6 znakov (z nich je jeden ordinálny). Text príkladu nás vyzýva urobiť potrebné úpravy, teda vyrobiť v MATLABe maticu rozmeru $[16, 7]$, doplniť hodnoty znakov a pritom sa zamyslieť nad tým, že medzi jednotlivými stĺpcami očakávame akúsi väzbu (ten, kto bol úspešný v oboch testoch, sotva môže byť veľmi neúspešný na skúške). Vecnejšie o väzbe medzi znakmi bude reč v nasledujúcej kapitole.

Príklad. Nasledujúca (skrátaná) tabuľka predstavuje učiteľov záznam o práci 16-tich študentov istého krúžku. Urobte potrebné úpravy, tabuľku doplňte a prezentujte dáta v MATLABe v štandardnej forme maticou rozmeru $[16, 7]$. Aktivita je ordinálny znak (malá-0, dobrá-4, veľmi dobrá-8, výnimočná-10) a je

kódovaný tak, aby jeho hodnoty korešpondovali s bodmi získanými v testoch, resp. bodoch zo zadania, resp. s bodmi získanými na skúške. Rozsahy pre bodové hodnotenie testov: 0 až 15, zadania: 0 až 10, resp. body zo skúšky: 0 až 50.

študent	aktivita	test 1	test 2	zadania	skúška	body celkom
1	8	13	14	7	39	81
2	4	9	11	6	42	72
..	..	..	..	..	..	..
16	10	14	13	8	43	88

Riešenie. Je zrejmé, že je veľká voľnosť v tom, ako doplníme hodnoty znakov. Jedna z mnohých možností je daná maticou, ktorej identifikátorom nech je "kruzok114" (mnemo: data sú z úlohy na strane 114).

kruzok114

kruzok114 =

1	8	13	14	7	39	81
2	4	9	11	6	42	72
3	4	10	10	4	30	58
4	10	12	13	8	47	90
5	4	11	12	5	38	70
6	8	14	12	10	45	89
7	10	15	14	9	48	96
8	8	12	13	7	44	84
9	8	11	8	6	43	76
10	8	13	13	8	47	89
11	4	9	11	5	35	64
12	8	10	12	5	31	66
13	0	5	8	3	27	43
14	8	12	14	9	46	89
15	8	11	12	6	37	74
16	10	14	13	8	43	88

••

Príklad. Nasledujúca tabuľka uvádza výsledok experimentu, v ktorom sa meral vplyv dvoch faktorov na výkon [W] (konkrétneho, skúmaného) motora. Prvým faktorom je typ benzínovej zmesi a druhým je hodnota predzápalu. Pri každej dvojici faktorov sa robili dve merania. Typ zmesi aj hodnota predzápalu sú kódované číslami 1, 2, 3 (napr. pre zmes-1 a s predzápalom-1 boli namerané výkony 1268 W a 1096 W). Uveďte dáta v štandardnom tvare, identifikujte prvky výberu a štatistické znaky, resp. hodnoty znakov na jednotkách výberu.

	predzápal-1	predzápal-2	predzápal-3
zmes-1	1268, 1096	1292, 1480	1230, 1125
zmes-2	1248, 1203	1902, 1764	1432, 1360
zmes-3	1596, 1508	1849, 1693	1598, 1470

Riešenie: Nech maticou dát je matica "motor115". Matica predstavuje dáta v štandardnom tvare.

motor115

motor115 =

1	1	1	1268
2	1	1	1096
3	1	2	1292
4	1	2	1480
5	1	3	1230
6	1	3	1125
7	2	1	1248
8	2	1	1203
9	2	2	1902
10	2	2	1764

11	2	3	1432
12	2	3	1360
13	3	1	1596
14	3	1	1508
15	3	2	1849
16	3	2	1693
17	3	3	1598
18	3	3	1470

• • •

Strany skript 117, 118 majú svoju "vylepšenú", novú verziu, ktorá bola ponúknutá na stránke predmetu už v minulosti v súbore "str117-118nova.pdf". Pre pohodlie ju na tomto mieste uvádzame a nadväzujeme na riadky str.116₁₂:

Realizujme tieto kroky na údajoch, ktoré získame realizáciou M-funkcie "rand" (pôjde o umelý výber generovaný počítačom). Prvý príkaz zabezpečí, že aj keď sú dáta náhodnej povahy, budú to vždy tie isté dáta (len preto, aby bolo možné porovnávať uvádzané medzivýsledky). Nech rozsah výberu $n = 100$.

```
rand('state',0), x = -500*log(1 - rand(100,1)); xmin = min(x), xmax = max(x)
xmin =          xmax =
    4.9551          2225.6
```

Rozhodnime do koľkých tried chceme údaje zatriediť. Interval $[0, 2240]$ obsahuje všetky dáta a je na nás, pre aký počet tried sa rozhodneme (Sturgesovo pravidlo dáva orientačnú pomôcku: $k \approx 1 + 3.3 \log_{10}(n)$), avšak predovšetkým volíme triedy tak, aby ich hranice, resp. reprezentanti tried neboli "strapaté" čísla). Pre jednoduchosť (a názornosť) voľme 10 tried šírky 224. Pre MATLAB hranice tried tvoria vektor, ktorý označme "ht". Pretože príkaz "hist" (histogram) pracuje s reprezentantami tried (pre nás je to opäť vektor, ktorý označme "rt"), vytvoríme vektor "rt" napr. takto:

```
ht = 0: 224: 2240; rt = ( ht(1:10) + ht(2:11) )/2
rt =
    112    336    560    784   1008   1232   1456   1680   1904   2128
```

Absolútne triedne početnosti získame volaním M-funkcie "hist" s jedným výstupným argumentom:

```
nj = hist(x, rt)
nj =
    32    22    18     9     8     6     2     1     1     1
```

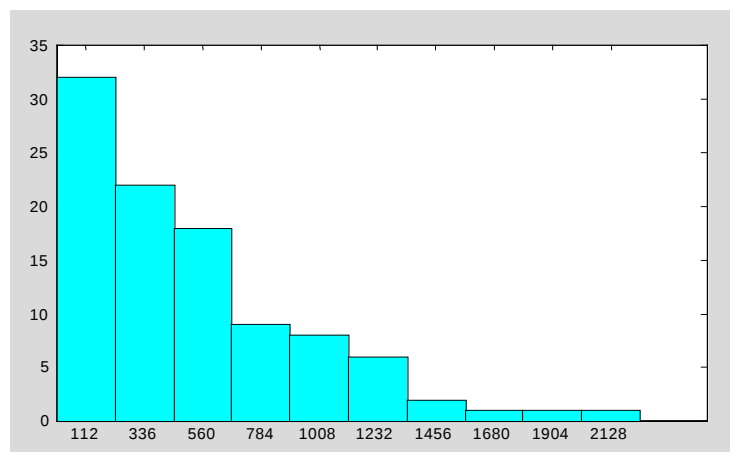
Prehľadne sa absolútne početnosti tried uvádzajú v tabuľke, ktorú doplníme o stĺpec kumulatívnych početností, resp. o relatívne početnosti tried (prípadne aj o kumulatívne relatívne početnosti).

V našom prípade sa jedná o jednoduché čísla, je samozrejme možné využiť M-funkciu "cumsum", resp. vektor relatívnych početností získať ako nj/n (v našom prípade $n = 100$).

číslo triedy	hranice triedy		reprezentant triedy	absolútna početnosť	kumulatívna abs. početnosť	relatívna početnosť	kumulatívna relat. početnosť
1	0	224	112	32	32	0.32	0.32
2	224	448	336	22	54	0.22	0.54
3	448	672	560	18	72	0.18	0.72
4	672	896	784	9	81	0.09	0.81
5	896	1120	1008	8	89	0.08	0.89
6	1120	1344	1232	6	95	0.06	0.95
7	1344	1568	1456	2	97	0.02	0.97
8	1568	1792	1680	1	98	0.01	0.98
9	1792	2016	1904	1	99	0.01	0.99
10	2016	2240	2128	1	100	0.01	1.00

Vizualizáciu rozdelenia početností tried dáva histogram. Získame ho funkciou "hist" volaním bez výstupného argumentu:

```
hist(x, rt), colormap(cool)
```



Pri niektorých štatistických spracovaniach je treba zvoliť triedy tak, aby početnosť každej z nich bola aspoň 5. V našom prípade to dosiahneme napríklad tak, že zlúčime posledné štyri triedy do jednej. Namiesto s reprezentantami je prirodzenejšie pracovať s hranicami tried, a preto si pripomeňme vektor hraníc "ht" s ktorým sme začínali:

```
ht = 0:224:2240
```

```
ht =
```

```
0    224    448    672    896    1120    1344    1568    1792    2016    2240
```

Zlúčenie posledných štyroch tried dosiahneme vytvorením nového vektora hraníc príkazom:

```
ht2 = [ ht(1:7) 2240 ]
```

```
ht2 =
```

```
0    224    448    672    896    1120    1344    2240
```

Na získanie nových triednych početností použijeme M-funkciu "histc" (c od *count*, lebo funkcia počíta, koľko dát sa dostane do jednotlivých tried [ht(i), ht(i+1))). Vstupnými argumentami funkcie "histc" sú: vektor dát a vektor hraníc tried. Výstupným argumentom je stĺpcový vektor absolútnych početností tried.

```
pt2 = histc( x, ht2 ); pt2'
```

```
ans =
```

```
32    22    18     9     8     6     5     0
```

Posledná zložka vektora "pt2" (v našom prípade je to 0) znamená počet dát, ktoré sa zhodujú s hodnotou poslednej zložky vektora hraníc. O tom, že zatriedenie dát sa deje do zľava uzavretých intervalov sa môžeme presvedčiť napríklad takto (ilustrujeme na preverení početnosti poslednej triedy):

```
sum(x < 2240) - sum(x < 1344)
```

```
ans =
```

```
5
```

Kontrolu toho, že skutočne všetky dáta sú zatriedené, dosiahneme pomocou príkazu:

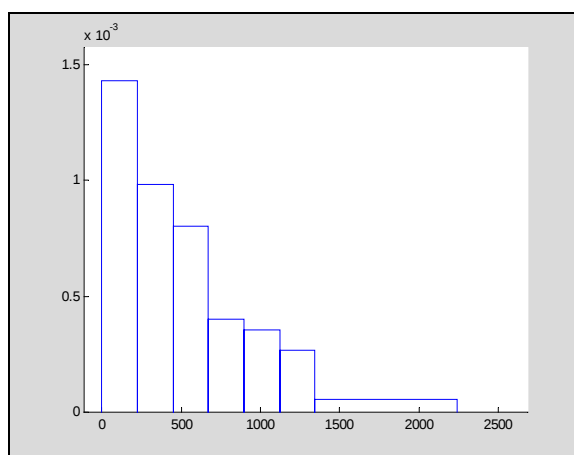
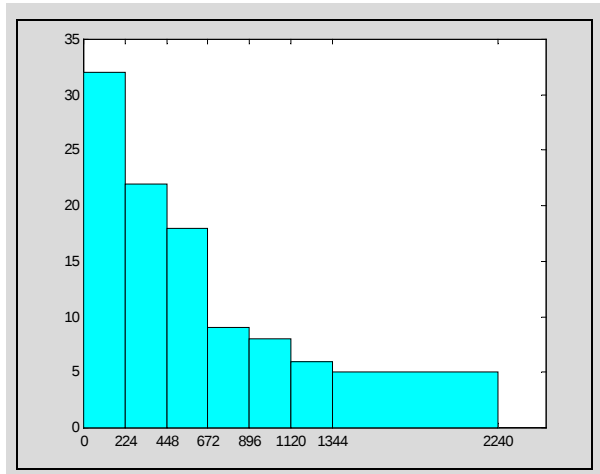
```
sum(pt2) == length(x)
```

```
ans =
```

```
1
```

Stĺpcový diagram zobrazujúci početnosti tried môžeme dostať príkazom "bar", volaním:

```
bar(ht2, pt2, 'histc'), colormap(cool)
```



Všimnime si, že tento stĺpcový diagram (obr. vľavo), vizualizuje početnosti jednotlivých tried *výškou* obdĺžnikov (porovnajme napr. výšky posledných dvoch, ktoré odpovedajú početnostiam 6, resp. 5). Takýto diagram však radšej nenazývame histogram.

Histogramom sme nazvali odozvu príkazu "hist" a to v prípade, keď triedy mali rovnakú šírku. V prípade, keď triedy z nejakého dôvodu nemajú rovnakú šírku, je histogram správne zostrojený vtedy, ak *plochy* obdĺžnikov sú úmerné početnostiam tried. Len tak zostrojený histogram možno označiť za *výberový* protipól funkcie hustoty (o pojme *funkcia hustoty* budeme hovoriť v kap. 4).

• • • • •

Úloha. Napíšte M-funkciu "hist2", ktorej argumentami budú vektory: "ht" (hranice tried), "pt" (absolútne početnosti tried) a odozvou bude histogram, v ktorom plochy obdĺžnikov budú úmerné početnostiam tried. Pomôcka: keďže šírky základní obdĺžnikov sú dané, výšky určíme tak, aby sa plocha obdĺžnika rovnala relatívnej početnosti triedy. Správne zostrojený histogram je na obrázku hore vpravo. Všimnite si, že teraz plochy obdĺžnikov vizualizujú početnosti tried (porovnajme plochy posledných dvoch obdĺžnikov).

Riešenie: Podľa autora tohoto textu, neočakávame od bežného poslucháča 2.ročníka, že dokáže písať skutočne dobré M-funkcie. Pre tých, ktorých taká činnosť baví, je výzvou napísať funkciu, ktorá by poskytovala taký obrázok, ako je hore vpravo. To, čo ukážeme tu a teraz, je síce menej, ale postačuje na to, aby bolo zdôraznený obsah slov:

"nie výšky obdĺžnikov histogramu, ale plochy obdĺžnikov majú reprezentovať početnosti tried".

Využijeme funkciu "bar" s tretím argumentom "histc". Najprv však je treba predeliť početnosti tried šírkou tried. Upravené početnosti označíme ako "upt2". Všetko ďalšie bude jasné z postupu, ktorý uplatníme na naše hore uvedené dáta. Pripomeňme si "ht2":

```
ht2 =
      0      224      448      672      896      1120      1344      2240
```

Vektor "sirky" vytvoríme takto:

```
sirky = ht2(2:end) - ht2(1:end-1)
sirky =
      224      224      224      224      224      224      896
```

pôvodné početnosti máme vo vektore "pt2":

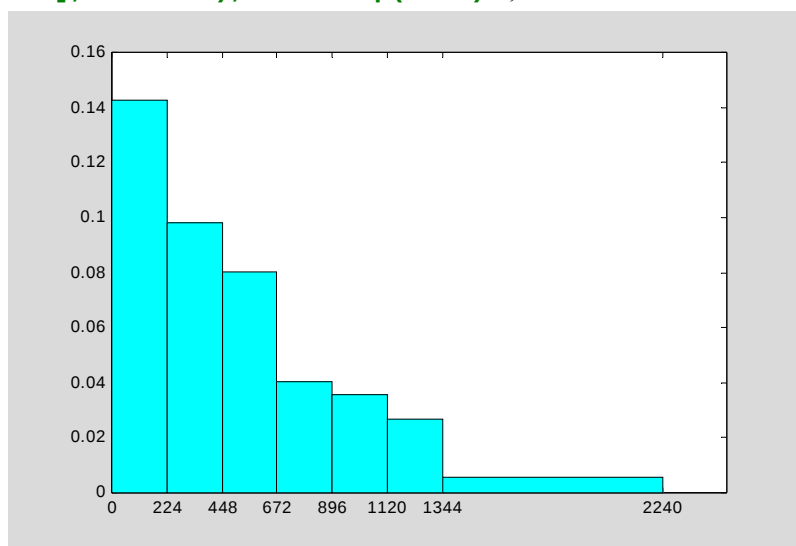
```
pt2 =
      32      22      18       9       8       6       5
```

upravené početnosti získame takto:

```
upt2 = pt2./sirky
upt2 =
      0.1429      0.0982      0.0804      0.0402      0.0357      0.0268      0.0056
```

Žiadaný histogram:

```
bar( ht2, [upt2 0], 'histc'), colormap(cool) ,
```



Všimnime si, že bolo potrebné pridať nulu k vektoru "upt2" (čo súvisí s prácou príkazu "histc").

1.4 Charakteristiky variability, t.j. miery rozptýlenosti

Zľahka komentujeme len záver tohoto článku a súčasne "prevzdušníme" záver str. 120 (v skriptách išlo o boj s priestorom).

Medzikvartilové rozpätie definujeme ako vzdialenosť medzi dolným a horným kvartilom, teda

$$mkr = q_3 - q_1$$

Keďže, voľne povedané, 50% hodnôt súboru leží medzi dolným a horným kvartilom, veľké *mkr* znamená veľkú variabilitu súboru (a naopak, malá hodnota *mkr* znamená malú variabilitu).

Zatiaľ sme však nepovedali, ako sa dolný a horný kvartil vypočítajú. Na tomto mieste to vezmeme prirodzene:

hodnotu dolného kvartilu určíme ako medián "dolnej polovice" súboru,
hodnotu horného kvartilu určíme ako medián "hornej polovice" súboru.

Ako presne určiť medián už vieme – to je vysvetlené v článku 1.3. Teraz poznamenávame, že v nasledujúcom článku bude reč o kvantiloch a stanovenie kvartilov cez kvantily je tiež možné. Začiatok môže znechutiť, že určenie kvartilu cez medián nedáva vždy tú istú hodnotu, ako určenie kvartilu cez kvantil. Toto treba zobrať ako fakt, ktorý sa v štatistike sem-tam objavuje. Takéto skutočnosti prestanú človeka rozladovať až vtedy, keď sa oboznámi so štatistikou dôkladnejšie a získa potrebný nadhľad.

V pružnom prostredí MATLABu, uvedené miery variability nájdeme celkom ľahko. Aby bolo jednoduché výpočty preveriť, použijeme súbor len štyroch čísel:

```
x = [1 3 4 6]; n = length(x); m = mean(x);
```

```
rozptyl = sum((x - m).^2)/n,
```

```
rozptyl =
```

```
1.8028
```

```
oprozptyl = sum((x - m).^2)/(n-1)
```

```
oprozptyl =
```

```
4.3333
```

```
smod = sqrt(rozptyl)
```

```
smod =
```

```
1.8028
```

Pretože rozptyl v angličtine označuje slovo *variance*, v MATLABe rozptyl získame M-funkciou "var". Overte, že

hodnotu opraveného rozptylu dáva funkcia "var(x)" a hodnotu rozptylu dáva odozva funkcie "var(x, 1)".

Hodnoty priemernej odchýlky, výberového rozpätia, či medzikvartilového rozpätia dostaneme takto:

```
priemod = sum(abs(x - median(x)))/n, rozp = max(x) - min(x), mkr = iqr(x)
```

```
priemod =
```

```
1.5000
```

```
rozp =
```

```
5
```

```
mkr =
```

```
3
```

Riešenie niektorých úloh článku 1.6

1.6.1 Zobrazme dáta 'defekty v paneloch' krabicovým diagramom!

Riešenie: matica štandardného tvaru má 30 riadkov a 3 stĺpce:

```
panely115
```

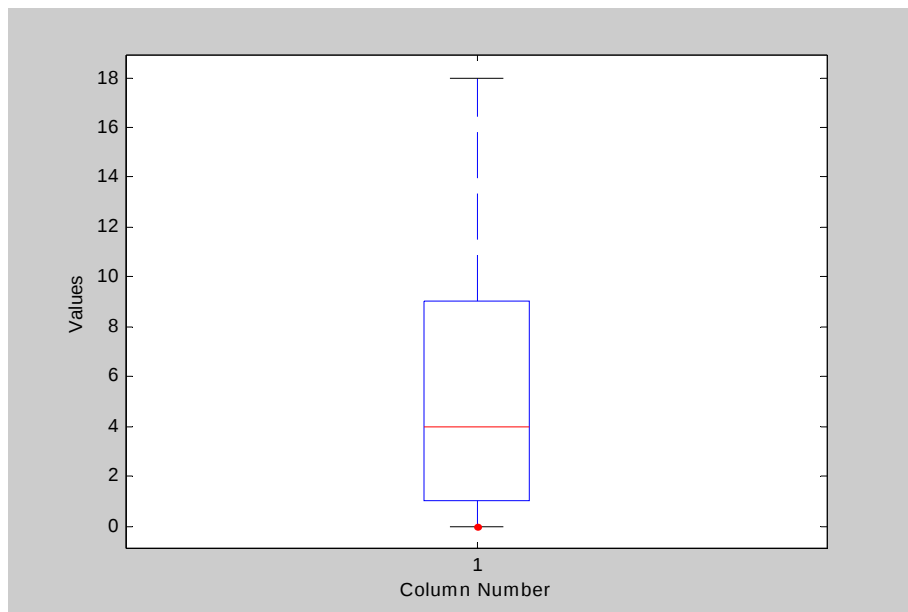
```
panely115 =
```

1	1	0
2	1	2
3	1	1
4	1	1
5	1	0
6	1	1
7	1	0
8	1	2
9	1	3
10	1	0
11	2	4
12	2	1

13	2	7
14	2	1
15	2	4
16	2	0
17	2	2
18	2	5
19	2	6
20	2	9
21	3	6
22	3	7
23	3	18
24	3	4
25	3	14
26	3	10
27	3	10
28	3	15
29	3	11
30	3	9

Funkcia "boxplot" je súčasť Štatistického toolboxu MATLABu. Urobiť "boxplot" má zmysel len pre data tretieho stĺpca – počty defektov:

boxplot(mat(:, 3))



•

1.6.3 Merala sa dĺžka bezporuchovej práce elektronického zariadenia (v hod.). Tabuľka udáva výsledky 50-tich meraní:

repr. tried	200	260	320	380	440	500	560	620	680
početnosti	2	6	12	11	8	5	3	2	1

Zlúčte niektoré triedy tak, aby každá trieda mala početnosť aspoň 5. Potom zostrojte histogram tak, aby plochy jednotlivých obdĺžnikov boli úmerné (relatívnym) početnostiam nových tried.

Riešenie: Za hranice tried považujeme čísla: 170, 230, 290, 350, 410, 470, 530, 590, 650, 710. Zlúčme prvé dve triedy a posledné tri. Za nové hranice a ich početnosti berme vektory "ht", "pt".

ht = [170 290 350 410 470 530 710]

ht =
170 290 350 410 470 530 710

pt = [8 12 11 8 5 6]

pt =
8 12 11 8 5 6

Nájdime šírky nových tried:

```
sirky = ht(2:7) - ht(1:6)
```

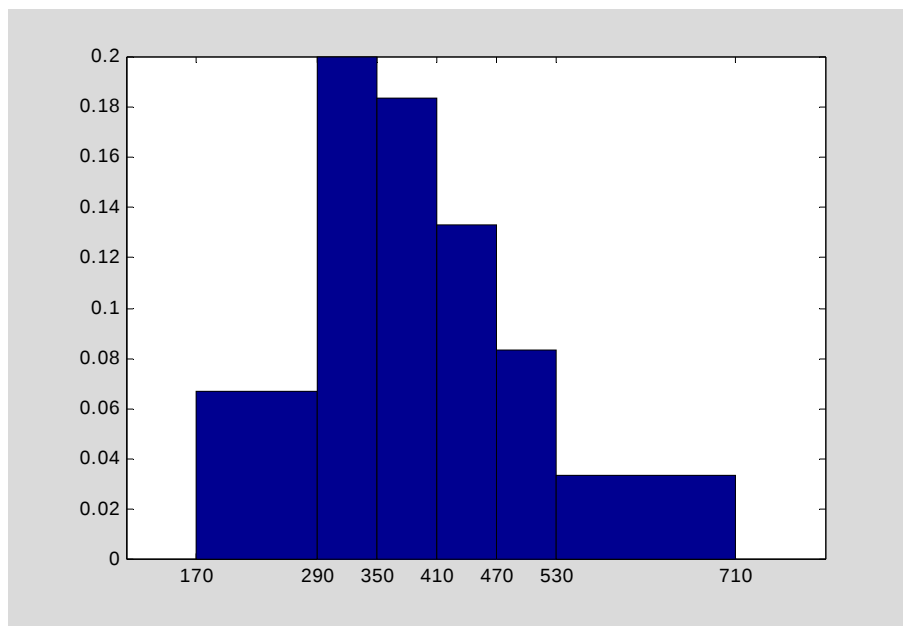
```
sirky =  
      120      60      60      60      60     180
```

Je to v poriadku, zlúčili sme prvé dve triedy a posledné štyri. Upravme početnosti tried tak, ako sme to urobili (v tomto texte) v riešení úlohy zo strany 118 skript.

```
upt = pt./sirky
```

```
upt =  
      0.0667      0.2000      0.1833      0.1333      0.0833      0.0333
```

```
bar(ht, [upt 0], 'histc')
```



• •

Úlohy 1.6.4 až 1.6.7 sú jednoduché a predpokladám, že nikomu nebudú robiť problémy. Ak nemáte k dispozícii štatistický toolbox, tak nemôžete znázorniť rozloženie veličín x a y v úlohe 1.6.8 krabicovým diagramom. Nevadí, urobte to histogramom.

1.6.9 Ukážte, že funkcia S definovaná vzťahom $S(a) = \frac{1}{n} \sum_{i=1}^n (x_i - a)^2$ nadobúda minimum pre $a = \bar{x}$.

Pomôcka: $\sum (x_i - a)^2 = \sum (x_i - \bar{x} + \bar{x} - a)^2 = \sum [(x_i - \bar{x})^2 + (\bar{x} - a)^2] \geq \sum (x_i - \bar{x})^2$

1.6.10 Ukážte, že funkcia $S(a) = \frac{1}{n} \sum_{i=1}^n |x_i - a|$ nadobúda minimum pre a rovné mediánu súboru (x_i)

(napriek tomuto faktu je priemerná odchýlka v MATLABe definovaná ako $S(\bar{x})$, vid' "help mad").

Úloha 1.6.10 Vás nemusí trápiť. Stačí si uvedomiť jej obsah.