UNIVERZITA PALACKÉHO V OLOMOUCI
PŘÍRODOVĚDECKÁ FAKULTA

BAKALÁŘSKÁ PRÁCE

Pokročilé grafické metody v R

Katedra matematické analýzy a aplikací matematiky
Vedoucí bakalářské práce: **Mgr. Kamila Fačevicová, Ph.D.**
Vypracoval(a): **Aneta Dvořáková**
Studijní program: B1103 Aplikovaná matematika
Studijní obor Matematika–ekonomie se zaměřením na bankovnictví/pojišťovnictví
Forma studia: prezenční
Rok odevzdání: 2022

BIBLIOGRAFICKÁ IDENTIFIKACE

Autor: Aneta Dvořáková

Název práce: Pokročilé grafické metody v R

Typ práce: Bakalářská práce

Pracoviště: Katedra matematické analýzy a aplikací matematiky

Vedoucí práce: Mgr. Kamila Fačevicová, Ph.D.

Rok obhajoby práce: 2022

Abstrakt: Software R umožňuje využít pro tvorbu grafů, kromě běžné knihovny R, i knihovny pro pokročilejší uživatele - ggplot2 a plotly. Hlavním úkolem této bakalářské práce je poskytnout návod pro tvorbu grafů v jednotlivých knihovnách, interpretovat data použitá v práci na základě jednotlivých druhů grafů a zhodnotit, která knihovna je pro používání nejvhodnější.

Klíčová slova: software R, ggplot, plotly, histogram, jádrový odhad hustoty, krabicový graf, houslový graf, bodový graf, heatmapa, interpretace grafu

Počet stran: 130

Počet příloh: 1

Jazyk: český

BIBLIOGRAPHICAL IDENTIFICATION

Author: Aneta Dvořáková

Title: Advanced graphical methods in R

Type of thesis: Bachelor's

Department: Department of Mathematical Analysis and Applications of Mathematics

Supervisor: Mgr. Kamila Fačevicová, Ph.D.

The year of presentation: 2022

Abstract: Beside the basic tools, the R software provides also more sophisticated graphical libraries as ggplot2 and plotly. The main aim of this bachelor thesis is to provide a handbook on creating basic analytical plots using these libraries, use the plots for a graphical inspection of an illustrative dataset and, finally, to compare the introduced graphical tools.

Key words: software R, ggplot, plotly, histogram, kernel density estimation, box-plot, violin-plot, scatter-plot, heatmap, interpretation of the graph

Number of pages: 130

Number of appendices: 1

Language: Czech

Prohlášení

Prohlašuji, že jsem bakalářskou práci zpracovala samostatně pod vedením paní Mgr. Kamila Fačevicová, Ph.D. a všechny použité zdroje jsem uvedla v seznamu literatury.

V Olomouci dne ....

podpis

Obsah

Poděkování

Ráda bych poděkovala vedoucí své bakalářské práce Mgr. Kamile Façevicové, Ph.D. za odborné vedení práce, cenné rady, vstřícnost při konzultacích a trpělivost, kterou mi v průběhu zpracování bakalářské práce věnovala.

Úvod

Tématem bakalářské práce je poskytnout návod na vykreslení různých typů grafů a jejich následné prezentace s využitím často používaných knihoven softwaru R (se zhodnocením, která z knihoven je pro použití nejvíce optimální), kterými jsou běžně dostupná knihovna softwaru R, ggplot2 a plotly.

Běžná knihovna R používá oproti ostatním knihovnám jednoduché kódy pro tvorbu grafů. Příkazy jsou psány ve formě `typ.grafu(...)` (u histogramů by to bylo například `hist(...)`), přičemž do příkazu tohoto typu zadáváme jak vstupní parametry (data z kterých čerpáme, název proměnné, kterou vykreslujeme), tak parametry týkající se vzhledu grafu. Typy grafů, které je možné vykreslit a parametry, kterými tyto grafy definujeme, a které byly zmíněné i v této práci, nalezneme na stránce [16].

Knihovna `ggplot2` používá složitější zápis. Základní kód se skládá ze dvou částí. První příkaz `ggplot(...)` definuje prostředí, jehož součástí je datová sada a parametr `aes(...)` sloužící k estetickému "mapování" (skládá se ze vstupních proměnných a jejich estetických úprav). Následně je k této části kódu pomocí znaménka + připojen "geom"ve formě `geom_typ.grafu()`, který z vložených dat tvoří geometrické útvary, neboli zvolené typy grafů. Dále můžeme k těmto částem přidávat další, které se týkají úprav os, legend, manuálních dodatečných úprav barevných škál apod. Veškeré informace k dostupným typům grafů, jejich tvorbě a možným parametrům (které byly použity i v této práci) lze nalézt na stránce [4], nebo v knize [18].

Knihovna `plotly` je oproti předchozím dvěma zaměřena na tvorbu interaktivních grafů. Příkazy je hodně podobná knihovně ggplot s tím rozdílem, že části

kódu nespojuje pomocí znaménka +, ale pomocí % > %. Výchozím příkazem pro tvorbu grafů v plotly je `plot_ly(...)`, do kterého zadáváme parametry týkající se datové sady, vykreslovaných veličin, vzhledu parametrů, ale také typ grafu, který vykreslujeme. Veškeré návody, informace o typech grafů, které knihovna dokáže vykreslit, a o používaných parametrech (i těch, které byly použity v této práci) lze nalézt na stránce [12].

Celá práce je rozdělena do čtyř kapitol. První kapitola je zaměřena na způsoby grafické reprezentace jedné náhodné veličiny, a popis jednotlivých typů grafů využívaných k této reprezentaci z teoretického, případně z konstrukčního hlediska, pro lepší pochopení důvodu jejich použití.

Druhá kapitola následně doplňuje první kapitolu o možnost zkoumat náhodné veličiny hromadně, prostřednictvím často využívaných grafů, kterými jsou scatterplot a heatmapa. Oběma typům grafů, jako v případě jedné náhodné veličiny, předchází jejich popis z teoretického, případně konstrukčního hlediska.

Třetí kapitola se týká přímo konstrukce jednotlivých typů grafů v takovém pořadí, v jakém by se při analýze dat mělo postupovat. Data, která byla pro tuto kapitolu použita lze nalézt na stránce [6]. Jedná se o data z výzkumu lidí se srdečními nemocemi. Tabulka s daty se skládá z 14 sloupců (veličin) a 270 pozorování (pacientů). V práci byly použity jen některé veličiny, a to kvantitativní veličiny "age" (věk pacientů), "bp" (krevní tlak pacientů v mm Hg), "max_hr" (maximální naměřené hodnoty tepu pacientů), "cholesterol" (cholesterol pacientů v mg/dl) a kategoriální veličiny "thallium" (reakce na thallium v těle s možnostmi "normal", "reversable defect" a "fixed defect"), "sex" (pohlaví pacientů) a "heart_disease" (přítomnost srdeční nemoci s kategoriemi "absence" nebo "presence").

Poslední (čtvrtá) kapitola je následně zaměřena na použití všech v práci zmíněných typů grafů na reálná data, kdy hlavním úkolem je předvést, co vše je možné vyčíst z grafu na základě obrázku. Jelikož se s prací pojí i zhodnocení knihoven, byla v této části na vykreslení grafů použita ta nejvhodnější knihovna - plotly.

Kapitola 1

Analýza jedné náhodné veličiny

Kapitola se zaměřuje na základní grafické reprezentace využívané při statistické analýze dat v situaci, kdy se zabýváme pouze jednou náhodnou veličinou. V této práci máme k dispozici náhodný výběr o rozsahu 270 z rozdělení vybrané zkoumané náhodné veličiny 1.1, na kterou jsou postupně aplikovány jednotlivé grafické metody.

Definice 1.1 *n-tice nezávislých náhodných veličin $X_1, \ldots, X_n$, které mají stejné rozdělení jako zkoumaná náhodná veličina X, se nazývá náhodný výběr rozsahu n z rozdělení náhodné veličiny X. [7]*

1.1. Histogram

Histogram, jakožto neparametrická grafická reprezentace rozdělení pravděpodobnosti, má podobu sloupcového grafu, který nám dává informaci o absolutních, případně relativních, četnostech realizací zkoumané spojité náhodné veličiny na intervalech.

Konstrukce histogramu spočívá v rozdělení intervalu $\langle x_{min}, \ldots, x_{max} \rangle$ na ose x na stejně dlouhé, nepřekrývající se intervaly, které můžeme nazvat třídy. Jednotlivá pozorování zkoumané náhodné veličiny jsou potom rozdělena do tříd. Na základě těchto pozorovaných hodnot vzniknou sloupce s výškou odpovídající jejich četnostem v daném intervalu se základnou, kterou tvoří dané třídy (intervaly). Tyto absolutní četnosti jsou zobrazeny na ose y. Pokud bychom hodnoty na

ose y normovali celkovým rozsahem, dostali bychom na této ose relativní četnosti. V případě normování hodnot na ose y určitou konstantou dostaneme odhad hustoty a výsledný součet ploch sloupců bude roven jedné.

Pro konstrukci normovaného histogramu je potřeba znát dva parametry. Jsou jimi bod t_0, ve kterém začíná první interval a šířka sloupců h, který plní funkci vyhlazovacího parametru. Za pomocí těchto parametrů jsou vytvořeny jednotlivé intervaly, do nichž spadají naše pozorované hodnoty. Pro k-tý sloupec půjde o interval $[t_k, t_k + 1)$, ve zkratce značený B_k, přičemž rozdíl $t_k - t_{k+1}$ je roven šířce sloupce h. Četnosti pozorovaných hodnot v daném intervalu potom normuje pomocí vztahu $\frac{1}{nh}$, aby se výsledná plocha celého histogramu nasčítala na jedničku. Budeme-li uvažovat, že v_k je počet hodnot v k-tém intervalu, neboli v B_k, histogram, jakožto neparametrický odhad hustoty v daném bodě, bude definován touto funkcí [15]:

$$\hat{f}(x) = \frac{v_k}{nh} = \frac{1}{nh} \sum_{i=1}^{n} I_{[t_k, t_{k+1})}(x_i)$$

pro $x \in R$. $I_{[t_k, t_{k+1})}$ zde vystupuje jako indikátorová funkce nabývající hodnot 0 nebo 1 v závislosti na tom, zda se daná pozorovaná hodnota x_i v uvažovaném intervalu nachází nebo ne. Zároveň platí, že v_k má binomické rozdělení $Bi(n, p_k)$, kde

$$p_k = \int_{B_k} f(t)\,\mathrm{d}t$$

přičemž $f(t)$ je teoretická hustota sledované náhodné veličiny. Střední hodnota v_k je potom np_k a rozptyl $np_k(1 - np_k)$ [15]. Tyto skutečnosti bychom následně využili, pokud bychom chtěli minimalizovat střední kvadratickou chybu (MSE) odhadu $\hat{f}(x)$, která je nejvíce ovlivňována volbou šířky sloupce h.

1.1.1. Volba počtu sloupců histogramu

Obecně platí, že jednotlivé sloupce histogramu by měly mít stejnou šířku. Nejčastěji používanou metodou, která nám určí ideální počet sloupců tak, aby byly jejich šířky stejné, je Sturgesovo pravidlo. Toto pravidlo by se však mělo

používat jen v případě, kdy máme k dispozici normálně rozdělená data.

Odvození Sturgesova pravidla je založené na vlastnosti, že binomické rozdělení $B_i(k-1,0{,}5)$ lze při dostatečně velkém k aproximovat normálním rozdělením $N(\frac{k-1}{2}, \frac{k-1}{4})$ [8]. Vzorec pro Sturgesovo pravidlo vznikl na základě myšlenky Sturgese, ve které uvažoval, že má k dispozici histogram o k sloupcích s šířkou rovnou jedné pro každý sloupec. Daný sloupec i potom obsahuje $\binom{k-1}{i}$ hodnot pro $i = 0, 1, \ldots, k-1$. Celkový počet četností ve všech sloupcích nám dává celkový rozsah výběru, z kterého je potom možné určit optimální počet sloupců [15]:

$$n = \sum_{i=0}^{k-1} \binom{k-1}{i} = (1+1)^{k-1} = 2^{k-1}$$

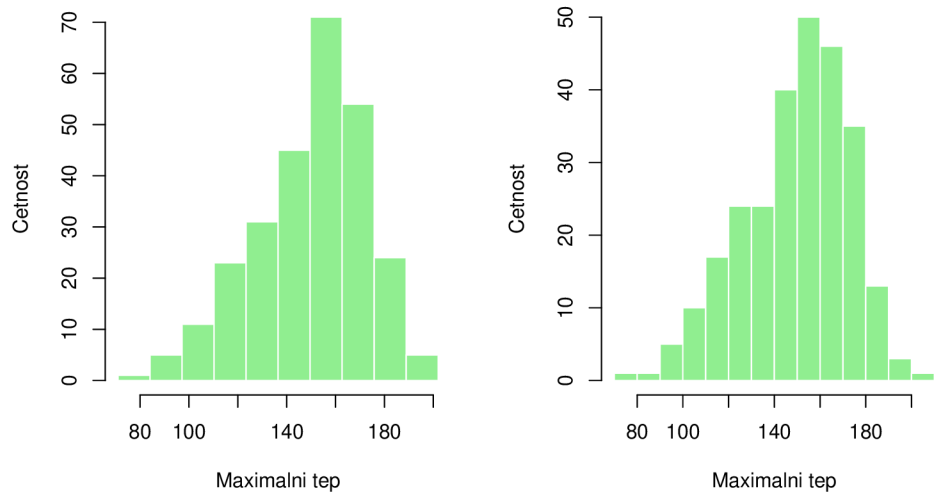
kde součet $\sum_{i=0}^{k-1} \binom{k-1}{i}$ je spočítaný pomocí binomické věty, která má obecně tento tvar [1]:

$$(x+y)^n = \sum_{k=0}^{n} \binom{n}{k} x^{n-k} y^k$$

Pro výpočet optimálního počtu sloupců k, v případě dat z normálního rozdělení, pro daný rozsah výběru n po vyjádření z rovnice dostaneme známý vztah, který ovšem není vhodný pro velké rozsahy výběru (od 200 výš) [15]:

$$k = 1 + \log_2 n$$

Histogram vykreslený Sturgesovým pravidlem v R o 270 pozorování, čemuž odpovídá 10 intervalů, bude následně obsahovat 14 sloupců (důvodem je to, že R upraví počet intervalů tak, aby odhadnutý graf vypadal co nejlépe):



Obrázek 1.1: Histogram naměřené maximální hodnoty tepu s použitím Sturgesova pravidla definovaného výše (vlevo) a tentýž histogram se Sturgesovým pravidlem, který používá software R (vpravo)

1.1.2. Význam použití histogramu

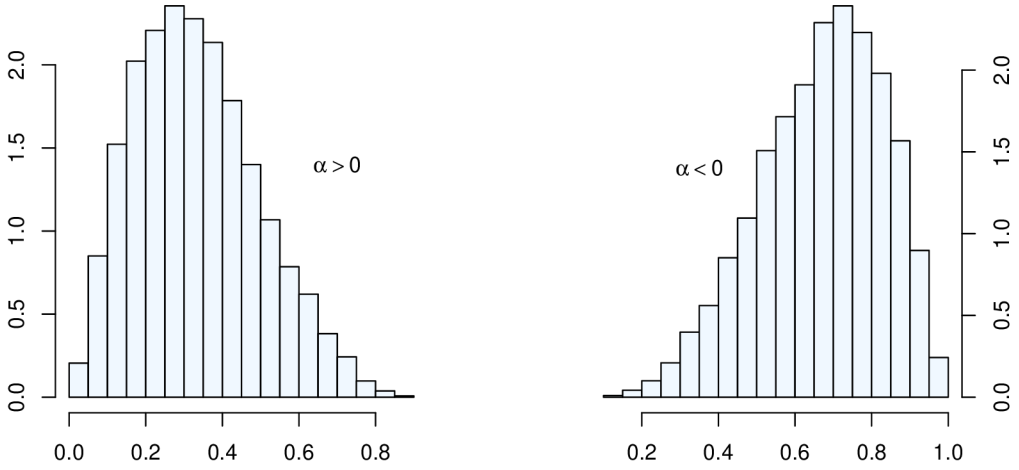
Uplatnění histogram nalézá především při prvotním náhledu na zvolenou sledovanou veličinu. Tvar histogramu totiž vypovídá o rozdělení této veličiny. Pokud by byla veličina např. normálně rozdělená, histogram by byl symetrický kolem svého středu. O symetričnosti a případném sešikmení histogramu lze kromě grafické reprezentace rozhodnout na základě koeficientu šikmosti α hodnot $x_1, \ldots, x_n$, [7]:

$$\alpha = \frac{\frac{1}{n}\sum_{i=1}^{n}(x_i - \overline{x})^3}{s_x^3}$$

kde s_x^3 je:

$$\left(\sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(x_i-\overline{x})^2}\right)^3$$

V případě nulového koeficientu se jedná o symetrické rozdělení. Pokud by byl koeficient záporný, histogram bude sešikmený vpravo, v opačném případě vlevo. Jak mohou jednotlivá zešikmení vypadat v praxi vidíte na obrázku 1.2 [1]



Obrázek 1.2: Ukázka histogramu sešikmeného vlevo (na obrázku vlevo) a histogramu sešikmeného vpravo (na obrázku vpravo)

Shrnutí

Histogramy jsou sice vhodné pro zisk prvotní představy o tvaru hustoty pravděpodobnosti zkoumané proměnné, ale kvůli jeho citlivosti na počet tříd, a skutečnosti, že jde o po částech spojitou funkci (a funkce hustoty je funkcí spojitou), je pro odhad hustoty výhodnější využít jádrových odhadů hustoty.

[1] Pro konstrukci histogramů byly použity náhodně vygenerované hodnoty z Beta rozdělení, přičemž kód v R pro generování z tohoto rozdělení jsem našla na stránce [11].

1.2. Jádrové odhady hustoty

Jádrový odhad hustoty je neparametrický odhad (kromě spojitosti nemáme žádné předpoklady pro rozdělení pravděpodobnosti zkoumané náhodné veličiny), který využíváme v případě, že máme k dispozici náhodný výběr příslušný spojité proměnné, a požadujeme odhad její hustoty. Základní jádrový odhad je dán vzorcem [15]:

$$\hat{f}(x) = \frac{1}{nh}\sum_{i=1}^{n} K\Big(\frac{x - x_i}{h}\Big) = \frac{1}{n}\sum_{i=1}^{n} K_h(x - x_i),$$

kde $K(x)$ (příp. $K_h(x) = K(x/h)/h$) je jádrová funkce (jádro), s podmínkou $K(x) > 0$, h je kladný vyhlazovací parametr (nazýván také šířka vyhlazovacího okna) a n je rozsah výběru. Jádrový odhad hustoty získáme tedy tak, že v každém bodě x_i sestrojíme jádro, a po zprůměrování všech n hodnot jader v bodě x získáme odhad hustoty v tomto bodě.

1.2.1. Kvalita odhadu

To, jak je odhad kvalitní, je dáno velikostí chyby, které jsme se dopustili při nahrazení skutečné hustoty náhodné veličiny hustotou odhadnutou. Kvalitu posuzujeme z hlediska lokálního a globálního.

Z lokálního hlediska je kvalita odhadu funkce $f(x)$ v daném bodě x určená pomocí střední kvadratické chyby odhadu (MSE), dané vztahem [15]:

$$MSE\{\hat{f}(x)\} = E[\hat{f}(x) - f(x)]^2 = Var\{\hat{f}(x)\} + Bias^2\{\hat{f}(x)\}$$

kde

$$Bias^2\{\hat{f}(x)\} = E[\hat{f}(x)] - f(x)$$

MSE jsme tedy schopni vyjádřit pomocí vychýlení odhadu $Bias^2\{\hat{f}(x)\}$ a pomocí rozptylu odhadu $Var\{\hat{f}(x)\}$.

Z hlediska globálního se kvalita vyjadřuje integrální čtvercovou chybou (ISE), danou vztahem [15]:

$$ISE\{\hat{f}(x)\} = \int [\hat{f}(x) - f(x)]^2 \, \mathrm{d}x$$

kde rozdíl

$$[\hat{f}(x) - f(x)]^2 = SE$$

je značen jako SE, což je čtvercová chyba, která vyjadřuje míru přesnosti odhadu.

Ze vztahu pro ISE poté můžeme určit i střední integrální kvadratickou chybu MISE, která je ovlivňována hlavně volbou vyhlazovacího parametru h. MISE se používá hlavně proto, protože hodnota ISE závisí na skutečné neznámé funkci hustoty, konkrétním odhadu a realizacích náhodné veličiny. Z tohoto důvodu je výhodnější používat střední hodnotu jednotlivých realizací ISE, tedy MISE, definovaný vztahem [15]:

$$MISE\{\hat{f}(x)\} = E[ISE\{\hat{f}(x)\}] = E[\int [\hat{f}(x) - f(x)]^2 \,\mathrm{d}x] =$$

$$= \int E[\hat{f}(x) - f(x)]^2 \,\mathrm{d}x = \int MSE\{\hat{f}(x)\} \,\mathrm{d}x$$

Nutno dodat, že MISE má dvě ekvivalentní interpretace, první říká, že MISE je měřítkem průměrné globální chyby, a druhá, že jde o akumulovanou bodovou chybu [15].

Pokud bychom chtěli například zjistit optimální hodnotu vyhlazovacího parametru, musíme zavést asymptotickou střední integrální kvadratickou chybu (AMISE), protože z MISE není možné tuto hodnotu zjistit přímo. AMISE je dána následujícím vztahem [15]:

$$AMISE\{\hat{f}(x)\} = \frac{R(K)}{nh} + \frac{1}{4}\sigma_K^4 h^4 R(f^2) \tag{1.1}$$

kde

$$R(K) = \int K(x)^2 \,\mathrm{d}x < \infty$$

$$\sigma_K^n = \int x^n K(x) \,\mathrm{d}x$$

$$R(f^2) = \int [f^2(x)]^2 \,\mathrm{d}x < \infty$$

Přičemž f^2 je absolutně spojitá funkce, funkcionál $R(f^2) < \infty$ a $R(K) < \infty$.

1.2.2. Jádrová funkce

Na jádrové funkce bývají obvykle kladeny následující podmínky [17]:

$$\int K(x)\,\mathrm{d}x = 1 \tag{1.2}$$

$$\int x^i K(x)\,\mathrm{d}x = 0,\ \ i = 1, \ldots, j-1 \tag{1.3}$$

$$\int x^j K(x)\,\mathrm{d}x \neq 0 \tag{1.4}$$

$$K(x) = K(-x) \tag{1.5}$$

kde j značí řád derivace, kterým je myšlen stupeň prvního nenulového momentu (tzn. v případě, že máme k dispozici jádrovou funkci druhého řádu, její první moment je nulový, druhý nenulový - proto se jí říká funkce druhého řádu).

Podmínka 1.2 požaduje, aby funkce $K(x)$ byla funkcí hustoty. Tím, že jako jádro bývá obvykle volena funkce, která je nezáporná, spojitá, symetrická kolem nuly a zároveň jednorozměrná, je podmínka splněna. Podmínky 1.3, 1.4 požadují, aby funkce byla centrovaná a od určitého řádu derivace měla nenulové momenty. Podmínka 1.5 potom zahrnuje výše zmíněnou sudou, symetrickou funkci. S takto zavedenými podmínkami se potom jádrové funkci K říká jádro řádu j, kdy řádem je myšlen řád derivace zmíněný o pár řádků výše (zároveň platí, že když vybíráme jádrovou funkci, její řád by měl odpovídat našim předpokladům o počtu derivací hustoty, kterou odhadujeme).

Pokud bychom si tedy zvolili jádrovou funkci druhého řádu, půjde vyloženě o funkci hustoty [8]. Při řádech, které jsou větší než dva by se mohlo stát, že výsledný odhad hustoty bude záporný. To by mohlo například způsobit, že nám bude výsledný odhad připadat hrubší pro mírné hodnoty (ty, které nejsou ani moc malé, ani moc velké) vyhlazovacího parametru h (a ne jen pro velmi malé hodnoty h jako u pozitivních jádrových funkcí). Přesto bývají používány i jádrové funkce s vyššími řády, protože zlepšují MISE (střední integrální kvadratická chyba).

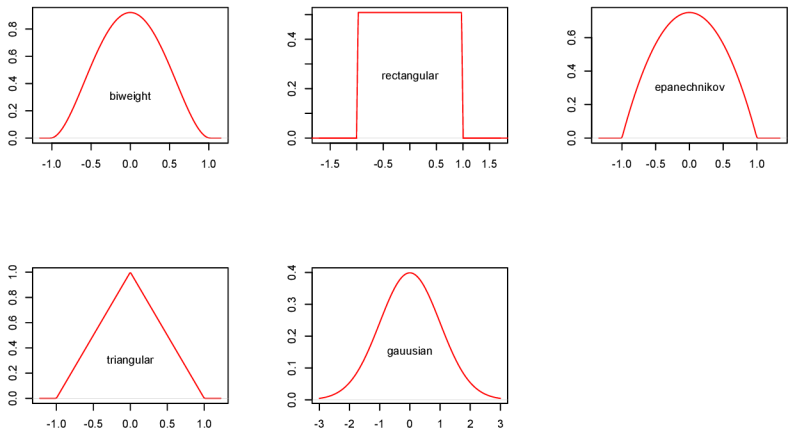
Volba jádrové funkce

Odhad hustoty "dědí" veškeré vlastnosti jádrové funkce (např. střední hodnotu, rozptyl, nebo i řád derivace). I přes to má ale jádrová funkce menší vliv na kvalitu odhadu než vyhlazovací parametr. Jádra, která jsou nejčastěji používána jsou právě jádra s řádem derivace dva (kvůli jistě platným podmínkám určující jádrovou funkci). Mezi taková jádra, která máme v nabídce i v softwaru R, patří [17]:

Jádro	**K(x)**
Obdélníkové	$\frac{1}{2}I(\|x\| \leq 1)$
Trojúhelníkové	$(1 - \|x\|)I(\|x\| \leq 1)$
Epanechnikovo	$\frac{3}{4}(1 - x^2)I(\|x\| \leq 1)$
Kvartické	$\frac{15}{16}(1 - x^2)^2 I(\|x\| \leq 1)$
Gaussovo	$\frac{1}{\sqrt{2\pi}} \exp(\frac{-1}{2}x^2)$

Tabulka 1.1: Typy jader

Kde $I(|x| \leq 1)$ je indikátorová funkce, která přiřazuje hodnotu 1, pokud x leží v intervalu $< -1, 1 >$ a 0 v opačném případě. Jádrové funkce z tabulky můžeme vidět na následujícím obrázku 1.3.



Obrázek 1.3: Jádrové funkce

Obecným doporučením pro volbu jádra je zvolit nějaké hladké, unimodální (jednovrcholové) jádro, které je symetrické podle počátku. Jádro, které je považované za optimální je Epanechnikovo jádro, které ovšem není vhodné na použití pro odhad derivace hustoty, neboť není spojitě diferencovatelné v bodech -1 a 1 [15].

1.2.3. Vyhlazovací parametr

Vyhlazovací parametr h bychom měli volit tak, aby nedocházelo k "přehlazení" ani k "podhlazení". Existuje mnoho metod pro volbu vyhlazovacího parametru. Vzhledem k tomu, že v softwaru R je výchozí metodou metoda referenční hustoty, dalo by se říct, že jde o nejužívanější metodu. Dalšími metodami, které v R lze použít, je metoda křížového ověřování, nebo plug-in metoda.

Metoda referenční hustoty

Metoda odhadu optimálního vyhlazovacího parametru h je založena na použití asymptotické střední integrální čtvercové chyby AMISE, která lze vyjádřit jako funkce vyhlazovacího parametru h, vztahem zadefinovaným v kapitole "Kvalita odhadu" 1.1. Z tohoto vztahu poté vyjádříme optimální hodnotu h tímto způsobem[15]:

$$h = \left[\frac{R(K)}{\sigma_K^4 R(f^2)}\right]^{\frac{1}{5}} n^{-\frac{1}{5}}$$

Princip odhadu vyhlazovacího parametru spočívá v nahrazení odhadované hustoty f nějakou referenční hustotou (tzn. nějakou známou hustotou ze třídy parametrických funkcí). Hodnota vyhlazovacího parametru potom bude záviset na této funkci, za kterou se nejčastěji volí Gaussova funkce (tzn. hustota normálního rozdělení). Optimální hodnota h při zvolení Gaussovy funkce má tuto hodnotu [15]:

$$h = (4/3)^{1/5}\sigma n^{-\frac{1}{5}} \approx 1{,}06\hat{\sigma} n^{-\frac{1}{5}}$$

kde odhad parametru σ je dán směrodatnou odchylkou σ_{SD}:

$$\hat{\sigma}_{SD} = \left(\frac{1}{n-1}\sum_{i=1}^{n}(\bar{x} - x_i)^2\right)^{\frac{1}{2}}, \bar{x} = \frac{1}{2}\sum_{1}^{n} x_i$$

Metoda nevychýleného křížového ověřování UCV

Principem této metody nalezení vhodného vyhlazovacího parametru je minimalizace ISE, kterou je možno podle [15] psát takto:

$$ISE(h) = \int [\hat{f}(x) - f(x)]^2 \, \mathrm{d}x = R(\hat{f}(x)) - 2 \int [\hat{f}(x) f(x) \, \mathrm{d}x] + R(f(x))$$

Přičemž $R(\hat{f}(x)) = \int [\hat{f}(x)^2 \, \mathrm{d}x]$ a hodnota $R(f(x)) = \int [f(x)^2 \, \mathrm{d}x]$ je nezávislá na odhadovaném vyhlazovacím parametru h, a můžeme ji při minimalizaci ISE zanedbat. Dále je možné aproximovat integrál na konci střední hodnotou odhadované hustoty [15]:

$$\int [\hat{f}(x) f(x) \, \mathrm{d}x] = E[\hat{f}(x)]$$

Pro následné určení $E[\hat{f}(x)]$ se postupuje tak, že máme-li k dispozici rozsah výběru n hodnot náhodné veličiny X, jednu hodnotu odebereme a zbylých $n - 1$ hodnot (kvůli tomu, abychom zmírnili závislost funkce na hodnotách x_i) použijeme k odhadu $E[\hat{f}(x)]$ (odhad spočívá v tom, že místo funkce $\hat{f}(x_i)$ bereme funkci $\hat{f}_{-i}(x_i)$). Toto opakujeme n-krát, pokaždé s vynecháním jedné hodnoty (každou z těch n hodnot vynecháme právě jednou). Výsledek následně zprůměrujeme. Poté dostaneme vztah metody nevychýleného křížového ověřování [15]:

$$UCV(h) = R(\hat{f}(x)) - \frac{2}{n} \sum_{i=1}^{n} \hat{f}_{-i}(x_i)$$

kde $\hat{f}_{-i}(x_i)$ značí odhad hustoty při vynechání bodu x_i. Odhadem vyhlazovacího parametru je taková hodnota, která minimalizuje hodnotu funkce UCV(h). Neboli [15]:

$$h = \arg \min_{h>0} UCV(h)$$

Metoda vychýleného křížového ověřování BCV

Oproti předchozí metodě je zde k výběru vhodného vyhlazovacího parametru h využita AMISE. Ve vzorci pro AMISE [15]:

$$AMISE(h) = \frac{R(K)}{nh} + \frac{1}{4}\sigma_K^4 h^4 R(f^2)$$

nahradíme $R(f^2)$ pomocí odhadu [10]:

$$\hat{R}(f^2) = R(\hat{f}^2) - \frac{R(K^2)}{nh^5}$$

kde f^2, $\hat{f}^2$ a K^2 jsou derivace druhého řádu jednotlivých funkcí a dosazením tohoto odhadu do AMISE dostaneme vztah [10]:

$$BCV(h) = \frac{R(K)}{nh} + \frac{1}{4}\sigma_K^4 h^4 \hat{R}(f^2)$$

Následnou optimální hodnotou vyhlazovacího parametru h bude opět, jako v předchozím případě, ta hodnota, která minimalizuje hodnotu funkce BCV(h), tentokrát z lokálního hlediska. Podrobnější odvození a vysvětlení metody je možné nalézt na stránce [10].

Plug-in metoda

Poslední metodou, kterou je možné pro odhad vyhlazovacího parametru h použít v softwaru R, je plug-in metoda. V této metodě je opět využita AMISE s tím rozdílem, že místo odhadu $\hat{R}(f^2)$ pracujeme s pomocnými vyhlazovacími parametry λ_1 a λ_2. Vyhlazovací parametr za použití této metody určíme následovně [15]:

$$h = \left[\frac{\hat{J}_1}{n}\right]^{\frac{1}{5}} + \left[\frac{\hat{J}_1}{n}\right]^{\frac{3}{5}} * \hat{J}_2$$

kde

$$\hat{J}_1 = \frac{R(K)}{(\mu_2)^2 \hat{R}_{\lambda_1}(f^2)}$$

$$\hat{J}_2 = \frac{\mu_4 \hat{R}_{\lambda_2}(f^3)}{(\mu_2)\hat{R}_{\lambda_1}(f^2)}$$

a

$$\hat{R}(f^3) = \int f^3(x)^2 \, \mathrm{d}x$$

$$\hat{R}(f^2) = \int f^2(x)^2 \, \mathrm{d}x$$

přičemž μ_2 je druhý moment, μ_4 je čtvrtý moment a λ_1 s λ_2 se doporučují volit pomocí normálního referenčního pravidla, s jehož použitím by měly tyto hodnoty [15]:

$$\hat{\lambda}_1 = 4{,}29 \, IQR \, n^{-1/11}$$

$$\hat{\lambda}_2 = 0{,}91 \, IQR \, n^{-1/9}$$

kde IQR je mezikvartilové rozpětí, které je v tomto případě podle [15] $IQR = 1{,}348 \, \sigma$, kde σ je směrodatná odchylka normálního rozdělení. Více k této metodě je uvedeno na stranách 184-185 v knize [15].

Tato metoda je na délku doby výpočtu optimálního vyhlazovacího parametru pro velký rozsah výběru nejrychlejší, avšak například metody křížového ověřování jsou lepší v tom, že u nich nedochází k tak velkému "přehlazení"při velkém rozsahu hodnot jako právě u plug-in metody.

1.3. Krabicový graf

Krabicový graf, neboli boxplot, je grafickou metodou pro zobrazení určitých charakteristik polohy. Byl vynalezen především z důvodu odhalení odlehlých hodnot, neboli "outlierů", a je konstruován za pomocí výběrových kvantilů.

Výběrové kvantily jsou hodnoty, které rozdělují soubor hodnot, seřazených od nejmenší po největší ve stanoveném poměru (z hlediska obsazenosti jednotlivých částí danými realizacemi). Kvantil je značen jako $\tilde{x}_p$ pro $p \in (0, 1)$, kde p značí procentní kvantil.

P-kvantil je reálné číslo takové, že $100p\%$ realizací v (uspořádeném) souboru hodnot je menších nebo rovno číslu $\tilde{x}_p$ a $100(1-p)\%$ realizací v (uspořádaném) souboru hodnot je větších nebo rovno číslu $\tilde{x}_p$. Mezi známé kvantily patří [7]:

- percentily - $\tilde{x}_{0,01} \ldots \tilde{x}_{0,99}$
- decily - $\tilde{x}_{0,2} \ldots \tilde{x}_{0,9}$
- dolní kvartil - $\tilde{x}_{0,25}$
- medián - $\tilde{x}_{0,50}$
- horní kvartil - $\tilde{x}_{0,75}$

Pro konstrukci boxplotu je nejdříve potřeba vypočítat kvartily $x_{0,25}$, $x_{0,50}$, $x_{0,75}$, IQR - mezikvartilové rozpětí, konce vousů (maximum a minimum boxplotu) a případně odlehlé hodnoty. Před samotným výpočtem hodnot je potřeba seřadit soubor hodnot od nejmenšího po největší. Tím dostaneme uspořádaný soubor hodnot $x_{(1)}, \ldots, x_{(n)}$. Pro p-tý kvantil spočítáme součin, který vyjadřuje pozici hodnoty p-tého kvantilu v uspořádaném souboru hodnot:

$$np$$

kde n vyjadřuje rozsah souboru. Pokud je np celé číslo, hodnota p-tého kvantilu bude počítána takto [7]:

$$\tilde{x}_p = \frac{x_{(np)} + x_{(np+1)}}{2}$$

Pokud je np necelé číslo, zaokrouhlíme toto číslo na nejbližší číslo dolů, a následně přičteme jedničku, což je matematicky zapsáno následovně [7]:

$$\tilde{x}_p = x_{([np]+1)}$$

kde symbol $[\cdot]$ vyjadřuje funkci s názvem "celá část".

Způsobů pro výpočet kvartilů (kvantilů) ovšem existuje celá řada, proto tento popsaný způsob není jediný možný.

Po výpočtu kvartilů můžeme přejít na výpočet mezikvartilového rozpětí IQR, který je dán jako rodíl horního a dorního kvartilu, neboli:

$$IQR = \tilde{x}_{0,75} - \tilde{x}_{0,25}$$

Konce vousů spočítáme následovně [5]:

- horní vous $= min\{max(x), \tilde{x}_{0,75} + 1{,}5 * IQR\}$
- dolní vous $= max\{min(x), \tilde{x}_{0,25} - 1{,}5 * IQR\}$

přičemž $max(x)$ je největší pozorovaná hodnota a $min(x)$ je nejmenší pozorovaná hodnota.

Nakonec spočítáme odlehlé hodnoty ("outliery") [7]:

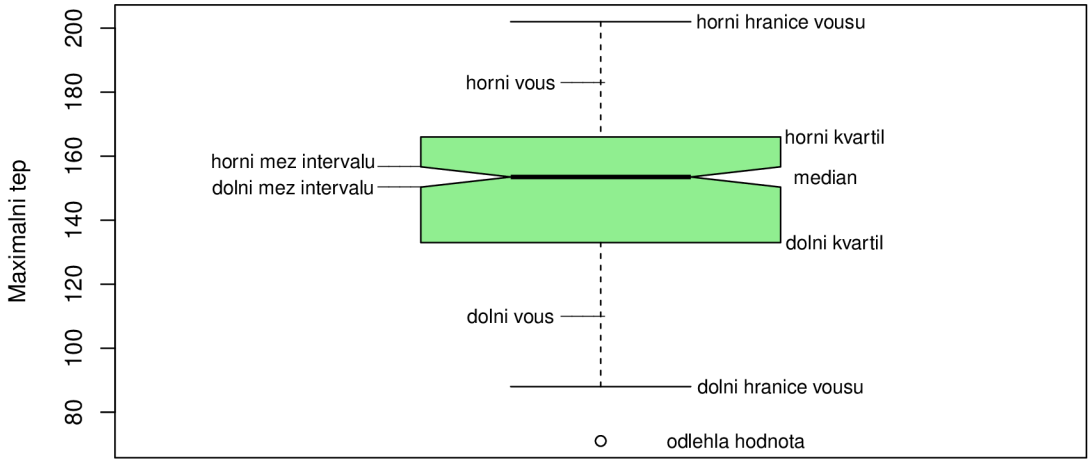
- x je extrémně velká hodnota, pokud platí:

$$x > \tilde{x}_{0,75} + 1{,}5(\tilde{x}_{0,75} - \tilde{x}_{0,25})$$

- x je extrémně malá hodnota, pokud platí:

$$x < \tilde{x}_{0,25} - 1{,}5(\tilde{x}_{0,75} - \tilde{x}_{0,25})$$

Po výpočtu těchto hodnot můžeme následně sestrojit box-plot. Na obrázku 1.4 na následující straně jsou v box-plotu znázorněny jednotlivé pojmy, které tu byly postupně zadefinovány.



Obrázek 1.4: Box-plot spojité proměnné maximální naměřené hodnoty tepu s vysvětlujícími popisky

V boxplotech se často konstruují i výřezy kolem mediánu, které jsou také znázorněny na obrázku 1.4. Tyto výřezy představují intervalový odhad mediánu, který vyjadřuje nejistotu odhadu mediánu. Intervalový odhad je zde volen tak, aby odhadovanou neznámou hodnotu (v tomto případě medián) pokrýval s pravděpodobností 0,95. Jedná se tedy o 95 % interval spolehlivosti.

Pro následný výpočet intervalu spolehlivosti v boxplotu použijeme následující vztah [2]:

$$\tilde{x}_{0,5} \pm \frac{1{,}58 * IQR}{\sqrt{n}},$$

kde $\tilde{x}_{0,5}$ je medián, IQR je mezikvartilové rozpětí a n je rozsah souboru. To, jak tento vzorce vznikl je popsáno na stránce [9].

Výřezy se používají hlavně při porovnávání více box-plotů mezi sebou, kdy sledujeme statisticky (ne)významný rozdíl mezi jejich mediány. V případě, že se výřezy jednotlivých box-plotů překrývají, neočekáváme statisticky významný

rozdíl mezi mediány, v opačném případě statisticky významný rozdíl očekáváme.

1.3.1. Význam použití boxplotu

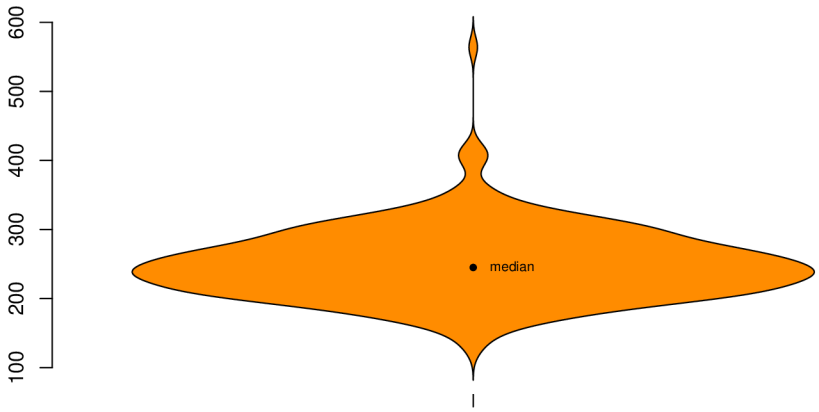
Box-plot se prioritně používá k odhalení odlehlých hodnot, protože tyto hodnoty potom mohou zkreslit například výsledné hodnoty průměrů.

Kromě odlehlých hodnot nám box-plot poskytuje i informaci o symetrii/asymetrii nebo variabilitě rozdělení zkoumané náhodné veličiny. Pokud jsou vousy box-plotu stejně dlouhé a medián je přibližně uprostřed, jedná se o symetrické rozdělení. Kladnou šikmost bychom poznali podle toho, že horní vous by byl delší než dolní a medián by byl blíže hodnotě 1. kvartilu (neboli k hodnotě $\tilde{x}_{0,25}$). Při záporné šikmosti by byl naopak dolní vous delší než horní a medián by byl blíže hodnotě 3. kvartilu ($\tilde{x}_{0,75}$). Variabilitu následně posuzujeme na základě délky vousů a šířky/výšky krabice, kdy platí, že čím větší je variabilita rozdělení, tím delší jsou vousy.

1.4. Houslový graf

Houslový graf je grafickou kombinací krabicového grafu a jádrového odhadu hustoty. Poskytuje nám tedy stejné informace jako krabicový graf (týkající se např. kvartilů a mezikvartilového rozpětí) a k tomu je doplněný z pravé i levé strany odhadem hustoty, která nám něco napoví o rozložení hodnot zkoumané kvantitativní veličiny (můžeme například vidět, zda jde o unimodální, bimodální nebo multimodální rozdělení pravděpodobností náhodné veličiny). Proto je v tomto ohledu jeho využití vhodnější.

Bývá zakreslován buď samostatně, nebo v kombinaci s krabicovým grafem, kvůli jasnějšímu orientování se v charakteristikách polohy. Stejně jako krabicový graf je potom jeho využití výhodné při porovnávání statisticky (ne)významných rozdílů kvantitativní náhodné veličiny při různých kategoriích kvalitativní náhodné veličiny. Samostatný houslový graf můžeme vidět na obrázku 1.5. Vykreslení dalších možností, tedy houslového grafu spolu s krabicovým grafem, nebo při různých kategorií je možné vidět v kapitole 3.4.



Obrázek 1.5: Houslový graf představující celkové množství cholesterolu v krvi člověka v jednotkách mg/dl se zvýrazněným mediánem

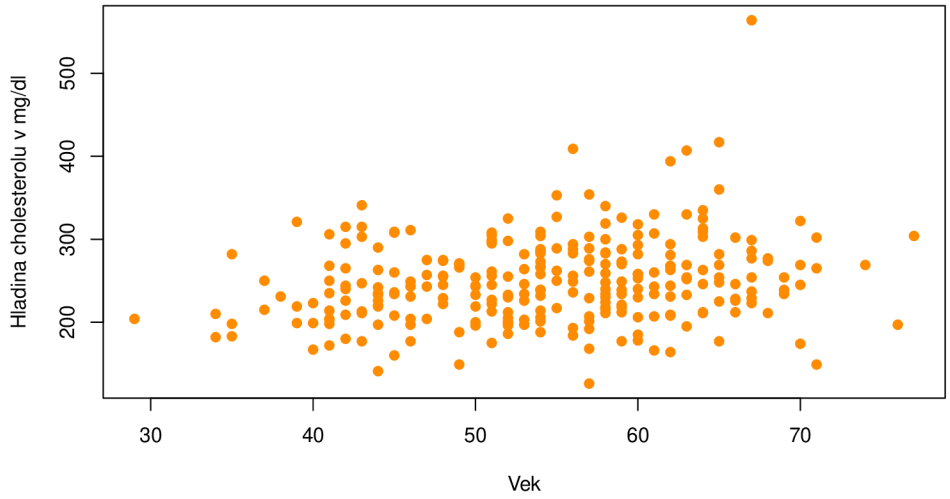
Kapitola 2

Analýza dvou a více náhodných veličin

Tato kapitola je zaměřená na grafické znázornění vztahu dvou nebo více náhodných veličin za použití bodového grafu a korelační heatmapy. Pro ilustraci máme opět k dispozici náhodný výběr o rozsahu 270 z rozdělení některých vybraných náhodných veličin.

2.1. Bodový graf

Bodový graf využíváme pro grafické znázornění vzájemného vztahu (korelace) dvou kvantitativních náhodných veličin (můžeme je označit X a Y). Graf je tvořen body (tzv. uspořádanými dvojicemi $(x_1, y_1), \ldots, (x_n, y_n)$ [7]), kde na vodorovnou osu nanášíme hodnoty jedné náhodné veličiny X a na svislou osu hodnoty druhé náhodné veličiny Y. Jsou-li body v grafu koncentrovány kolem nějaké křivky (např. v případě sledování lineární závislosti kolem přímky), svědčí to pro závislost náhodných veličin X a Y. V případě, že jsou body od této křivky viditelně vzdáleny, svědčí to naopak pro nezávislost veličin. To, jak vypadá základní bodový graf, je možné vidět na obrázku 2.1 (slovem "základní"je zde myšleno to, že jde pouze o vykreslení vztahu dvou kvantitativních veličin, bez zahrnutí informace o dodatečných proměnných, např. prostřednictvím nějakého barevného odlišení bodů dle kategorií kvalitativní veličiny, kterému bude dán prostor v kapitole o vizualizaci v R).



Obrázek 2.1: Bodový graf vyjadřující vztah hladiny cholesterolu (v mg/dl) a věku

Pro vyjádření síly vztahu dvou veličin používáme korelační koeficienty (Pearsonův a Spearmanův). Vidíme-li z bodového grafu, že by mezi dvěma veličinami mohla být lineární závislost, použijeme pro popis jejich vztahu Pearsonův korelační koeficient. Pokud lineární vztah neočekáváme, ale dá se předpokládat alespoň monotónní závislost, použijeme Spearmanův korelační koeficient. Zároveň platí, že Pearsonův korelační koeficient je vhodné použít zejména v případě, kdy pracujeme s výběrem z normálního rozdělení [7].

2.1.1. Pearsonův korelační koeficient

Máme-li k dispozici náhodný výběr z dvourozměrného rozdělení $(X_1, Y_1)', \ldots (X_n, Y_n)'$, potom je Pearsonův (výběrový) korelační koeficient, za předpokladu, že $S_X^2 > 0$ a $S_Y^2 > 0$, definován v knize [7] následovně:

$$R_{X,Y} = \frac{S_{X,Y}}{\sqrt{S_X^2 \, S_Y^2}} = \frac{S_{X,Y}}{S_X \, S_Y}$$

kde jednotlivé položky vypočítáme takto [7]

$$S_{X,Y} = \frac{1}{n-1}\sum_{i=1}^{n}(X_i - \overline{X})(Y_i - \overline{Y})$$

$$S_X^2 = \frac{1}{n-1}\sum_{i=1}^{n}(X_i - \overline{X})^2$$

$$S_Y^2 = \frac{1}{n-1}\sum_{i=1}^{n}(Y_i - \overline{Y})^2$$

$$\overline{X} = \frac{1}{n}\sum_{i=1}^{n}X_i$$

$$\overline{Y} = \frac{1}{n}\sum_{i=1}^{n}Y_i$$

přičemž $S_{X,Y}$ značí výběrovou kovarianci, S_X^2, S_Y^2 výběrové rozptyly a $\overline{X}$, $\overline{Y}$ výběrové průměry. Zároveň platí [7]:

$$P(-1 \leq R_{X,Y} \leq 1) = 1$$

Pokud nám tedy vyjde hodnota korelačního koeficientu 1, půjde o přímou lineární závislost mezi veličinami v tom smyslu, že vysoké hodnoty jedné veličiny se v souboru vyskytují spolu s vysokými hodnotami druhé veličiny a naopak. Pokud bude jeho hodnota rovna -1, půjde o nepřímou lineární závislost v tom smyslu, že vysoké hodnoty jedné veličiny se v souboru vyskytují spolu s nízkými hodnotami druhé veličiny a naopak. Naopak čím více se blíží hodnota korelačního koeficientu k nule, tím slabší je lineární závislost mezi veličinami.

2.1.2. Spearmanův korelační koeficient

Předpokladem pro použití Spearmanova korelačního koeficientu je pouze monotónní závislost. Na rozdíl od Pearsonova korelačního koeficientu zde nepracujeme přímo s náhodným výběrem, ale s pořadím hodnot tohoto výběru. Z tohoto

důvodu patří do kategorie neparametrických metod a je nazýván "Spearmanův korelační koeficient pořadové korelace"[7].

Mějme tedy pořadí $R_1, \ldots, R_n$ veličin $X_1, \ldots, X_n$ a pořadí $Q_1, \ldots, Q_n$ veličin $Y_1, \ldots, Y_n$. Spearmanův korelační koeficient je následně definovaný jako výběrový Pearsonův korelační koeficient, který se počítá z dvojic $(R_1, Q_1)', \ldots, (R_n, Q_n)'$ vztahem z knihy [7]:

$$R_S = \frac{\sum_{i=1}^{n} R_i Q_i - n\overline{R}\,\overline{Q}}{\sqrt{(\sum_{i=1}^{n} R_i^2 - n\overline{R}^2)(\sum_{i=1}^{n} Q_i^2 - n\overline{Q}^2)}}$$

Pro jednotlivé složky v R_S potom platí stejné vztahy jako zmíněné výše. Taktéž platí, že tento koeficient nabývá hodnot $< -1, 1 >$, přičemž při shodných pořadích je roven 1, při opačných pořadích je roven -1 [7]. Následná interpretace (ne)závislostí je obdobná (vypadává zde linearita) jako u Pearsonova korelačního koeficientu. Více informací je zmíněno v knize [7].

2.1.3. Korelace více náhodných veličin

Pro vyjádření vztahu více než dvou náhodných veličin slouží korelační matice. Předpokládáme tedy, že máme k dispozici výběr z rozdělení p-rozměrného náhodného vektoru $\mathbf{X} = (X_1, \ldots, X_p)'$ se střední hodnotou $\boldsymbol{\mu}$ a varianční maticí $\boldsymbol{\Sigma}$, přičemž rozsah výběru je větší než počet složek vektoru ($n > p$). Dále máme výběrový průměr $\overline{\mathbf{X}}$ a výběrovou varianční matici $\mathbf{S}$, které jsou definovány takto [7]:

$$\overline{\mathbf{X}} = \frac{1}{n}\sum_{i=1}^{n} \mathbf{X}_i$$

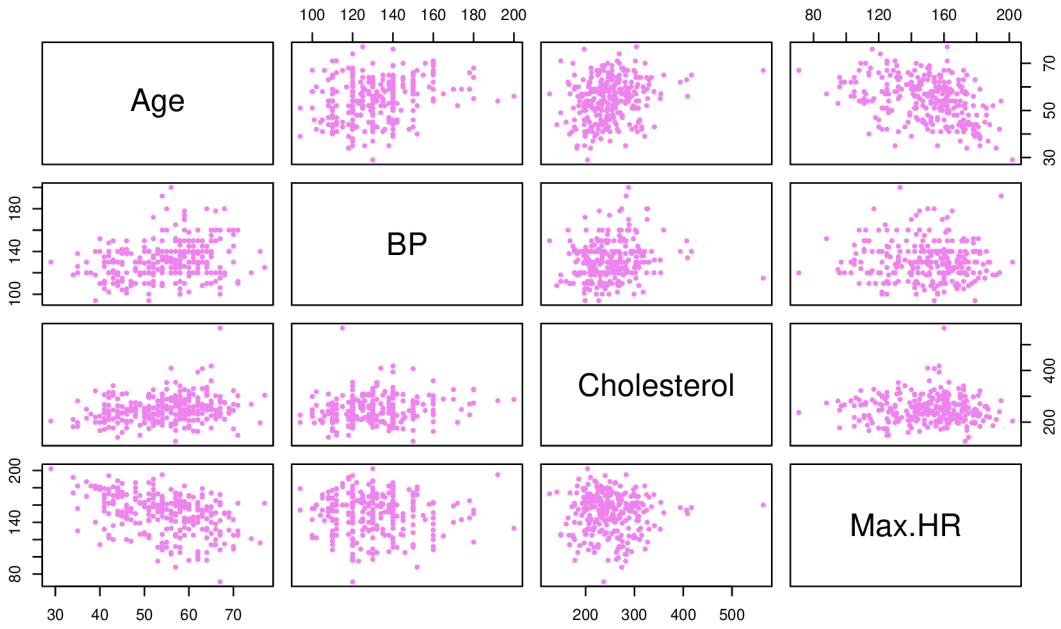
$$\mathbf{S} = \frac{1}{n-1}\sum_{i=1}^{n} (\mathbf{X}_i - \overline{\mathbf{X}})(\mathbf{X}_i - \overline{\mathbf{X}})'$$

Pokud jsou všechny prvky matice $\mathbf{S}$ na diagonále kladné s pravděpodobností 1, lze definovat výběrovou korelační matici tímto vztahem [7]:

$$\mathbf{R_X} = (R_{ij})_{i,j=1}^{p} = \left(\frac{S_{ij}}{\sqrt{S_{ii}S_{jj}}}\right)_{i,j=1}^{p}$$

Matice $\mathbf{R_X}$ má na diagonále jedničky a mimo diagonálu korelační koeficienty odpovídajících složek. Pokud je náš p-rozměrný náhodný vektor $\mathbf{X} = (X_1, \ldots, X_p)'$ normálně rozdělený a jeho složky jsou navzájem nezávislé, korelační matice $\mathbf{R_X}$ bude jednotková [7]. Platí tu tedy stejný princip usuzování o vztahu veličin jako u korelace dvou náhodných veličin, jen jde o zobecněnou formu pro případ, kdy máme veličin více.

Graficky je možné vztah více náhodných veličin zobrazit pomocí matice složené z bodových grafů jednotlivých kombinací veličin, jejíž příklad je možné vidět na obrázku 2.2.

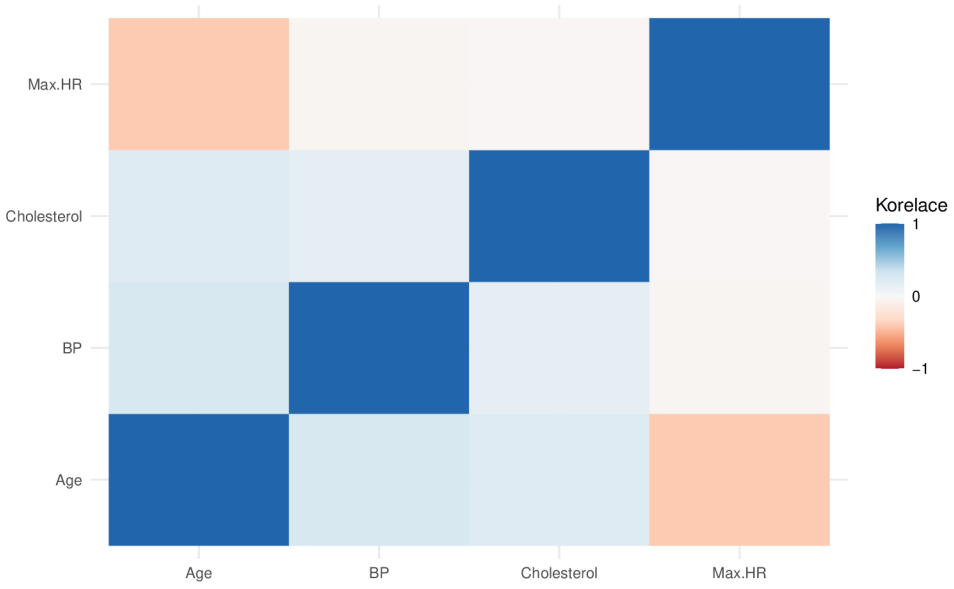


Obrázek 2.2: Matice bodových grafů jednotlivých kombinací těchto náhodných veličin: Věk pacientů (Age), Naměřené hodnoty krevního tlaku (BP), Naměřené hodnoty cholesterolu v mg/dl (Cholesterol) a Maximální naměřená hodnota tepu (Max.HR)

2.2. Heatmapa

Heatmapa je grafická reprezentace dat pomocí škály barev, kde každá barva odpovídá nějaké hodnotě. Škálu barev lze použít buď tak, že budeme mít jednu barvu od nejsvětlejšího odstínu po nejtmavší, (nejtmavší by mohl určovat nejvyšší hodnotu a nejsvětlejší nejnižší, nebo naopak) případně obráceně, a nebo bychom si zvolili libovolnou kombinace barev podle sebe.

Sama o sobě nám heatmapa dává informaci o velikosti hodnot jednotlivých veličin a jejich vztahu k pozorováním, který je znázorněn pomocí barev. Proto se v praxi často používá korelační heatmapa, která za pomocí škály barev vyjadřuje, jak moc jsou jednotlivé dvojice kvantitativních veličin korelované. Pro ukázku přidávám korelační heatmapu 2.3 přímo z knihovny `ggplot`, protože jak uvidíme v praktické části, heatmapa za použití výchozí R knihovny není přehledná.



Obrázek 2.3: Heatmapa vyjadřující korelaci mezi jednotlivými kombinacemi těchto veličin: Maximální naměřená hodnota tepu (Max.HR), Naměřené hodnoty cholesterolu v mg/dl (Cholesterol), Naměřené hodnoty krevního tlaku (BP), a Věk pacientů (Age)

Kapitola 3

Vizualizace v R

3.1. Histogram

3.1.1. Histogram v R

Pro vykreslení histogramu se v běžné knihovně softwaru R používá příkaz:

`hist(x,...)`, kde

`x` - vektor hodnot, které chceme vykreslit.

Příkaz obsahuje několik dalších parametrů, z nichž jsou velmi často využívané zejména parametry:

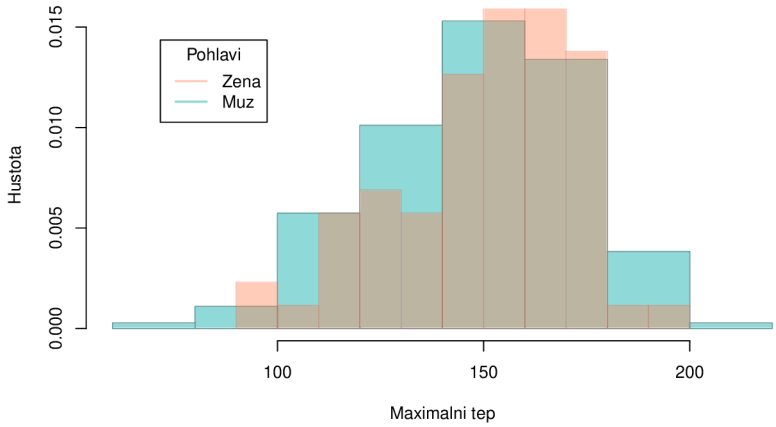
`breaks` - parametr pro konstrukci sloupců histogramu s výchozím nastavením pro Sturgesovo pravidlo (použití pravidla v klasické knihovně R je možné vidět na obrázku 1.1), psáno: `breaks = "Sturges"`

- do parametru lze zadat vektor dělících bodů intervalu, funkci, která nám takovýto vektor vygeneruje, číslo udávající počet sloupců histogramu, funkci pro výpočet počtu sloupců, nebo název známého algoritmu, mezi nimiž je na výběr možnost "Sturges", "Scott"nebo "FD".

`freq` - parametr k manipulaci s osou y s možnostmi `TRUE` nebo `FALSE`, přičemž pro obě varianty můžeme použít dva způsoby zápisu:

- `freq = TRUE` - výchozí nastavení (ale pouze pro konstantní šířku dělících intervalů), které vykreslí histogram s absolutními četnostmi na vertikální ose
- `freq = FALSE` - takto zadaný parametr umožní histogram znormovat (jeho celková plocha bude rovna jedné), a je vhodnější např. pro porovnávání rozdělení spojité proměnné při různých kategoriích diskrétní proměnné, nebo pro porovnávání empirického rozdělení s teoretickou hustotou.

Použití parametru `freq` můžeme vidět na obrázku 3.1, spolu s následnou konstrukcí celého grafu.



Obrázek 3.1: Normovaný histogram naměřené maximální hodnoty tepu vykreslený zvlášť pro muže a ženy

Při konstrukci histogramů na obrázku 3.1 s daty, které máme k dispozici, postupujeme následovně:

1. Vytvoříme histogram pro jednu kategorii (zde kategorie Muži), kde pod `Max.HR` je uloženo `data$Max.HR`

```
hist(Max.HR[data$Sex == "Muz"],
 main = "", xlab = "Maximalni tep", ylab = "Hustota", freq = F,
```

```
border = rgb(0.37, 0.62, 0.62,1), col = rgb(0.56, 0.85, 0.85,1))
```

- `xlab` - název osy x
- `ylab` - název osy y
- `main = ""` - název grafu nastaven tak, aby se žádný nevypsal
- `border = rgb(...)` - parametr umožňující barevné ohraničení grafu nastavený tak, aby byla barva průhlednější (pomocí `rgb`), kde průhlednost je definována číslem na 4. pozici, které se pohybuje v intervalu $< 0, 1 >$, přičemž čím blíž je číslo nule, tím průhlednější graf je
- `col = rgb(...)` - parametr pro vybarvení vnitřní části grafu s taktéž nastavenou průhledností

2. Vytvoříme histogram pro druhou kategorii (zde kategorie Ženy) a sloučíme pomocí parametru `add = T`, který slouží pro přidání vykreslovaného grafu k předchozímu

```
hist(Max.HR[data$Sex == "Zena"],
 freq = F, add = T, border = rgb(0.93, 0.42, 0.31,0.4),
 col = rgb(1, 0.50, 0.31,0.4))
```

3. Pro lepší přehlednost v tom, která barva vyjadřuje kterou kategorii, přidáme legendu

```
legend(locator(1), legend = c("Zena", "Muz"),
 title = "Pohlavi", lty = 1, lwd = 2,
 col = c(rgb(1, 0.50, 0.31,0.4), rgb(0.56, 0.85, 0.85,1)))
```

- `locator(1)` - parametr umožňující umístit legendu na námi zvolené místo
- `legend` - parametr pro název kategorií
- `col` - parametr pro barevné odlišení bodů nebo linek/čar jednotlivých kategorií

- `title` - název legendy
- `lty` - parametr pro volbu typu linky/čáry v legendě
- `lwd` - parametr pro volbu šířky linky/čáry

Po spuštění jednotlivých kódů následně dostaneme histogram maximální naměřené hodnoty tepu rozdělený podle kategorií proměnné na muže a ženy, jež vidíme na obrázku 3.1.

3.1.2. Histogram v ggplotu

Forma zápisu histogramu s využitím knihovny `ggplot2` se oproti klasické R knihovně celkem liší. Základní zápis je dán spojením:

`ggplot(...) + geom_histogram(...)`, kde

`ggplot(...)` - první část kódu, která nastavuje základní parametry grafu, kterými jsou:

- `data` - parametr, za který dosazujeme ta data (datovou sadu), ze kterých hodláme čerpat při konstrukci histogramu nějaké spojité proměnné
- `aes(x,...)` - parametr sloužící k výběru proměnných k jejich vykreslení a nastavení parametrů těchto sledovaných proměnných z estetického hlediska:
  - `x` - vektor hodnot spojité proměnné, kterou chceme vykreslit
  - `fill` - parametr umožňující vybarvit vnitřek histogramu (výhodné zejména pro vybavení histogramu podle kategorií diskrétní proměnné)
  - `color` - parametr sloužící k vybarvení hran histogramu

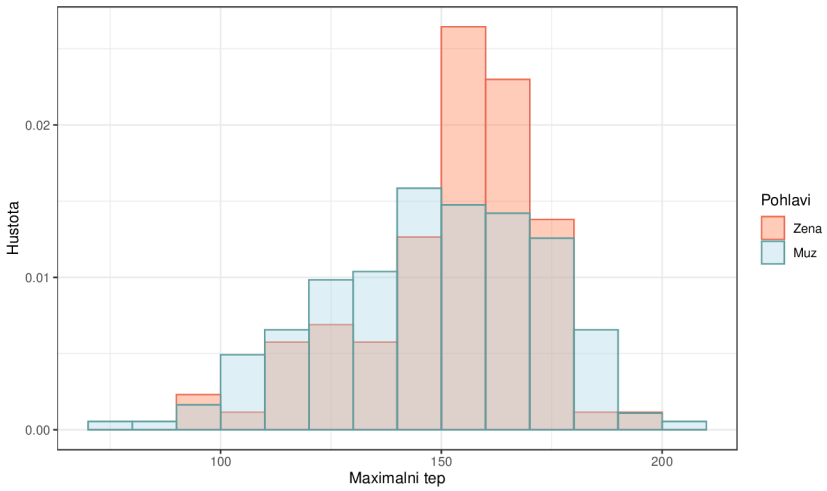
`geom_histogram(...)` - druhá část kódu, která se už týká přímo toho typu grafu, který vykreslujeme (zde histogram), přičemž využívanými parametry v této části jsou například:

- `aes(y =...)`, kam za `y` zadáváme:

  - `..density..` parametr zadávaný pro znormování histogramu (jeho použití je možné vidět na obrázku 3.2)
  - `..count..` - parametr, s jehož zadáním bude osa y obsahovat četnosti histogramu (takto je to přednastaveno v rámci výchozího nastavení)

- `breaks` - parametr, kterým volíme, jakým způsobem budou konstruovány sloupce histogramu, lze do něj zadat vektor dělících bodů intervalu nebo funkci pro výpočet počtu sloupců histogramu (pokud bychom chtěli zadat například přesný počet sloupců, museli bychom použít parametr `bins`)
- `alpha` - parametr používaný pro zprůhlednění barvy histogramu

Pro větší estetičnost je možné přidávat do kódu další části, kterými si můžeme manuálně nastavit, jaké barvy pro vykreslení bude histogram používat v případě, že vybarvujeme podle různých kategorií diskrétní proměnné (jejich použití je uvedeno v rámci konstrukce histogramu v kódu obrázku 3.2).



Obrázek 3.2: Normovaný histogram naměřené maximální hodnoty tepu u mužů a žen

Konstrukce normovaného histogramu spojité proměnné při různých kategorií diskrétní proměnné na obrázku 3.2 v `library(ggplot2)` je provedena následovně:

1. Načteme knihovnu ggplot příkazem

   ```
   library(ggplot2)
   ```

2. Vytvoříme tabulku pomocí `data.frame()` pro přejmenování sloupců, které budu v této sekci o histogramech v ggplotu používat (toto však není povinné, je to tvořeno pro přehlednost)

   ```
   data <- data.frame(pohlavi = data$Sex, Max.HR = data$Max.HR)
   ```

3. Sestavíme kód pro vykreslení histogramů

   ```
   ggplot(data, aes(x = Max.HR, fill = pohlavi)) +
   geom_histogram(breaks = stur, aes(color = pohlavi, y =..density..),
    position = "identity", alpha = 0.4) +
   ggtitle("") + theme_bw() + xlab("Maximalni tep") +
   ylab("Hustota") +
   scale_color_manual(values=c("coral2","cadetblue"),name ="Pohlavi")
   +
   scale_fill_manual(values=c("coral","lightblue"),name ="Pohlavi")
   ```

   - `ylab(...)` - název osy y
   - `xlab(...)` - název osy x
   - `ggtitle(...)` - název grafu
   - `theme_bw(...)` - parametr měnící výchozí vzhled pozadí za histogramem (ve výchozím nastavení bychom měli světle modré pozadí), přehled dalších typů vzhledů najdeme na stránce [3]
   - `breaks` - parametr měnící způsob konstrukce sloupců histogramu, přičemž v tomto kódu byla vytvořena taková funkce (zmíněná v 3.1.2) , pomocí které budou sloupce konstruovány Sturgesovým pravidlem
   - `position` - parametr, který umožní měnit pozice histogramů, zde je zvolena pozice "identity", díky které jsou histogramy srovnatelné,

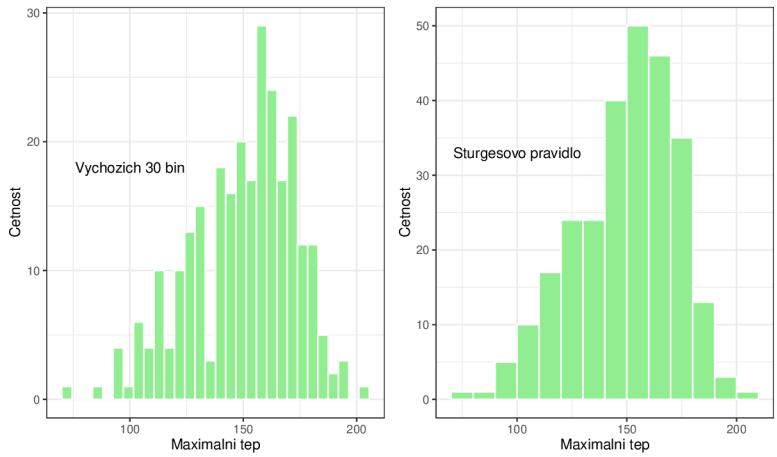
další možností je "fill" (histogramy vykreslí tak, aby zaplnily celou plochu okna), dále výchozí možnost "stack", (vykreslí histogramy přes sebe tak, že do prvního vykresleného histogramu se vykreslí ten další, což pro jejich vzájemné srovnání není vhodné), nebo možnost "dodge" (vykreslí sloupce histogramů tak, že vzhledově budou připomínat barploty - pro každý interval hodnot na ose x je vykreslen sloupec podle všech kategoriích)

- `scale_color_manual` - umožňuje manuálně nastavit barvy hran histogramu pro jednotlivé kategorie, zadáno ve formě `values=c(...)`, a zároveň tyto barvy promítne do legendy (vybarví hrany čtverečku)
- `scale_fill_manual` - umožňuje manuálně nastavit barvy vnitřku histogramu pro jednotlivé kategorie, zadáno ve stejné formě jako `scale_color_manual`, přičemž taktéž promítá barvy do legendy (vybarví vnitřek čtverečku)

4. Po spuštění kódu následně dostaneme histogram maximální naměřené hodnoty tepu zvlášť pro muže a pro ženy, který vidíme na obrázku 3.2

Sturgesovo pravidlo v ggplotu

Oproti předchozí knihovně v ggplotu není pro histogram Sturgesovo pravidlo ve výchozím nastavení. Místo něj je nastavený na `breaks = 30 bins`, což znamená, že se histogram vždy vykreslí s 30 sloupci. Se 30 sloupci se vykreslí i v případě, že máme malý počet pozorování s tím, že neobsazené sloupce zůstanou "neviditelné", a proto se v těchto případech doporučuje parametr `breaks` změnit. Rozdíl ve vykreslení histogramu s výchozím nastavením `breaks` a histogramu s použitím Sturgesova pravidla vidíme na obrázku 3.3.



Obrázek 3.3: Porovnání automaticky vykresleného histogramu maximální naměřené hodnoty tepu v ggplotu (vlevo) s histogramem vykresleným Sturgesovou metodou (vpravo)

Konstrukce histogramů, které vidíme na obrázku 3.3 tímto způsobem:

1. Vytvoříme jednoduchý histogram maximální naměřené hodnoty tepu s výchozím nastavením parametru `breaks`, který vidíme na obrázku 3.3 vlevo

```
ggplot(data, aes(x = Max.HR)) +
geom_histogram(col = "white", fill = "lightgreen",
  show.legend = F) +
theme_bw() + ggtitle("") + xlab("Maximalni tep") +
ylab("Cetnost") +
annotate("text", label = "Vychozich 30 bin", x=100, y=18, size=4)
```

   - `show.legend = FALSE` - parametr sloužící k odstranění legendy
   - `annotate()` - umožňuje vkládat text libovolně do grafu

2. Vytvoříme funkci, která je ekvivalentní nastavení `breaks = "Sturges"` v běžné knihovně R, ve tvaru [14]:

```
stur ← pretty(range(x), n = nclass.Sturges(x), min.n = 1)
```

- `range()` - vypočítá minimum a maximum hodnot
- `n` - značí použití Sturgesova pravidla na hodnoty
- `pretty()` - rozdělí třídy histogramu tak, aby měly stejnou šířku

3. Dosadíme funkci za parametr `breaks`

```
ggplot(data, aes(x = Max.HR)) +
geom_histogram(col = "white", fill = "lightgreen",
 breaks = stur, show.legend = F) +
ggtitle("") + xlab("Maximalni tep") + ylab("Cetnost") +
theme_bw() + annotate("text", label = "Sturgesovo pravidlo",
  x = 100, y = 45, size = 4)
```

4. Po spuštění následně dostaneme histogam s použitím Sturgesova pravidla pro konstrukci sloupců, který vidíme na obrázku 3.3 vpravo, a pokud bychom chtěli vykreslit oba histogramy vedle sebe zároveň, použili bychom knihovnu `library(gridExtra)` s příkazem

```
grid.arrange(p1,p2, nrow = 1)
```

   - `p1` - název, pod který uložíme kód histogramu s výchozím nastavením pro breaks
   - `p2` - název, pod který uložíme kód histogramu s použitím Sturgesova pravidla
   - `nrow = 1` - parametr, který s tímto zapsáním zajistí, že se dva grafy vykreslí na jeden řádek

5. Celkový výsledek je potom vidět na obrázku 3.3

3.1.3. Histogram v plotly

Plotly je knihovna pro tvorbu interaktivních grafů dostupná z příkazu `plotly`, která má svým způsobem podobnou strukturu zápisu jako knihovna ggplot. Nejvíce se od knihoven ggplot a klasické knihovny R, z hlediska konstrukce histogramu, liší poměrně složitějším algoritmem, za pomoci kterého vytváří sloupce histogramu. Pro jednoduché vykreslení histogramu využijeme základní vztah:

`plot_ly(data, type = "histogram", x =...,...)`, kde

`data` - parametr, za který dosazujeme data, ze kterých čerpáme

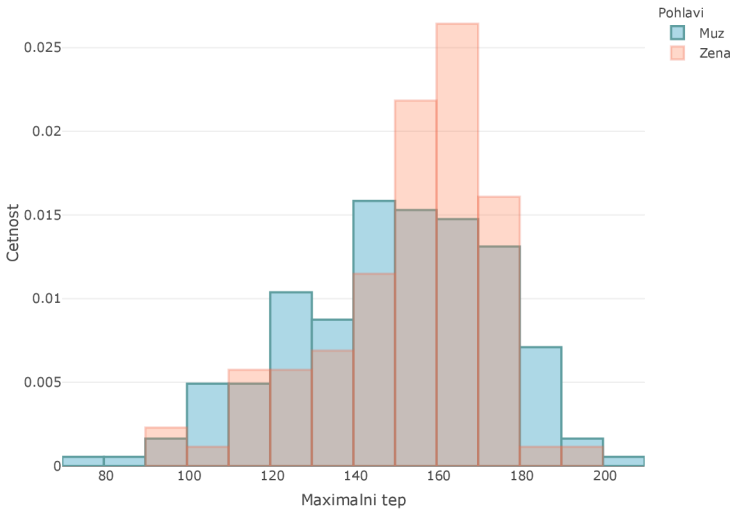
`type` - parametr, do kterého vždy zapisujeme typ grafu, jehož vykreslení požadujeme

`x` - parametr, za který dosazujeme opět název spojité proměnné, kterou chceme vizualizovat

Dalšími důležitými, a často používanými parametry při vykreslování histogramů jsou `xbins` a `histnorm`:

`xbins` - parametr, který umožňuje změnit výchozí nastavení konstrukce sloupců

`histnorm` - parametr, který upravuje hodnoty na ose y tak, abychom mohli pracovat například s normovaným histogramem místo četnostního (použití normování je možné vidět na obrázku 3.4), k dispozici jsou možnosti "percent", "probability"(stejně jako procenta vyjadřuje část z celku, kdy v tomto případě součet všech četností ve sloupcích je roven 1), "density"(součet všech ploch sloupců odpovídá celkovému počtu hodnot podělených délkou intervalu) a "probability density (součet všech ploch sloupců je roven jedničce)



Obrázek 3.4: Normovaný histogram maximálního naměřeného tepu u mužů a žen v plotly

Konstrukce histogramu na obrázku 3.4 probíhá následovně:

1. Načteme knihovnu `plotly`

   ```
   library(plotly)
   ```

2. Vytvoříme histogram pro muže

   ```
   norm_hist <- plot_ly(data) %>%
   add_histogram(histnorm = "probability density",
    x = ~Max.HR[which(data$Sex == "Muz")], name = "Muz",
    marker = list(color = "rgba(173,216,230,1)",
     line = list(color = "rgba(95,158,160,1)", width = 2)))
   ```

   - `x = ~Max.HR(...)` - hodnoty proměnné, kterou vykreslujeme zde musíme zadávat s ~ na začátku
   - `add_histogram` - část kódu, kterou přidáváme histogram (používáno proto, abychom při vykreslování více histogramů podle různých kategorií diskrétní proměnné mohli manipulovat s jednotlivými histogramy zvlášť)

- `name` - parametr, do kterého zadáváme název kategorie, která se následně objeví v legendě
- `marker = list()` - parametr sloužící k vybarvení vnitřku histogramu
- `line = list()` - parametr sloužící k vybarvení hran histogramu, psán uvnitř `marker`
- `width` - parametr, kterým volíme tloušťku hran histogramu (jde o součást parametru `line`)
- `histnorm` - parametr nastavený na normování osy y, dalšími možnostmi je buď výchozí nastavení pro absolutní četnosti na ose y, `"percent"`, `"probability"` a nebo `"density"`

3. Vytvoříme histogram pro ženy

```
norm_hist <- norm_hist %>%
add_histogram(histnorm = "probability density",
 x = ~Max.HR[which(data$Sex=="Zena")],
 name = "Zena", marker = list(color = "rgba(255,127,80,0.3)",
  line = list(color = "rgba(238,106,80,0.3)", width = 2)))
```

4. Přidáme další část kódu, která slouží k doladění informací o histogramu (název os, legenda..)

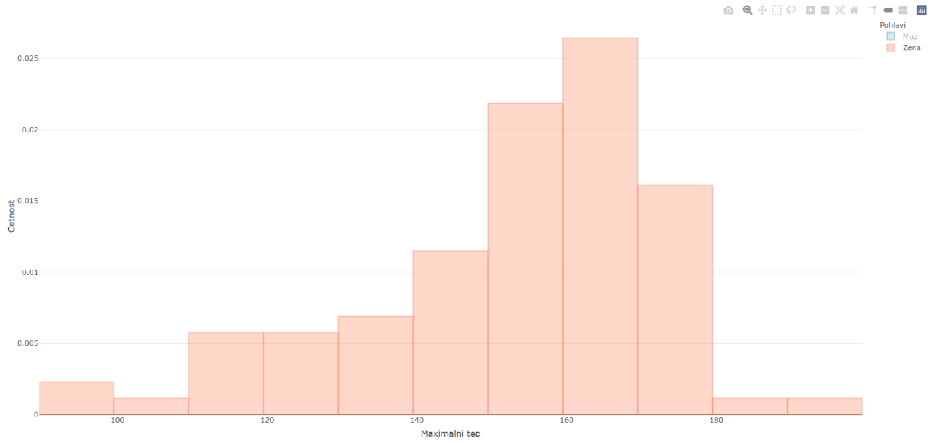
```
norm_hist <- norm_hist %>% layout(barmode = "overlay",
 title= "", xaxis = list(title = "Maximalni tep"),
 yaxis = list(title = "Cetnost"),
 legend = list(title = list(text = "Pohlavi")))
```

- `layout` - část kódu, která slouží k přidání popisů os, názvu legend, názvu grafu a podobných "obecností"
- `xaxis = list()` - název osy x

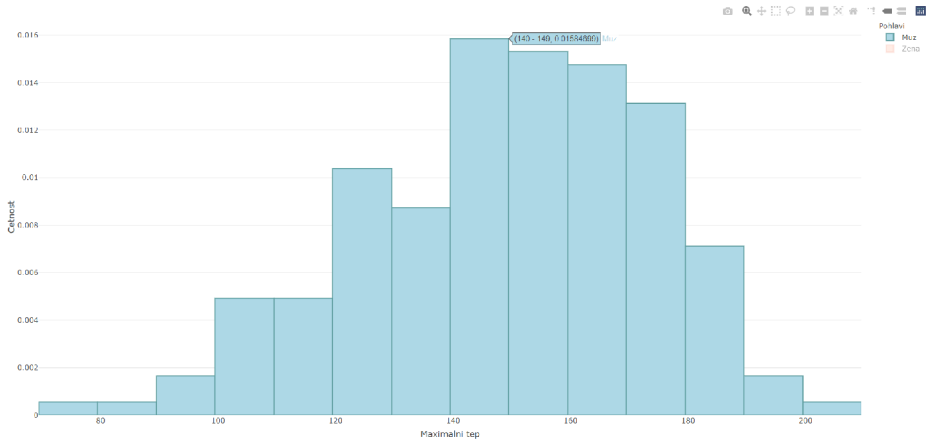
- `yaxis = list()` - název osy y
- `title` - název grafu
- `barmode` - parametr, kterým se nastavuje pozice histogramů (zde nastaven tak, aby byly histogramy srovnatelné - tzn. nastavení jejich tříd je srovnatelné)
- `legend = list()` - parametr sloužící k manipulaci s legendou, kde při změně názvu legendy použijeme `title = list(text = ...)`

5. Po spuštění `norm_hist` dostaneme výsledný histogram maximální naměřené hodnoty tepu zvlášť pro muže a ženy, který vidíme na obrázku 3.4

Výhodou využití plotly knihovny, kromě čistě interaktivního zobrazení, je možnost zobrazení pouze jedné konkrétní kategorie v histogramu, aniž bychom museli konstruovat zvlášť histogram pro tuto danou kategorii. Stačí nám pouze "odkliknout"v legendě čtvereček té kategorie, jejíž vykreslení nepožadujeme. Na následujících obrázcích 3.5, 3.6 uvádím printscreeny grafů při odkliknutí čtverečků, jejichž interaktivní podoba je uložena na přiloženém CD pod názvy "hist_plotly_zena"a "hist_plotly_muz".



Obrázek 3.5: Printscreen html verze histogramu maximálního naměřeného tepu u žen vykresleném v plotly po odkliknutí čtverečku nepožadované kategorie

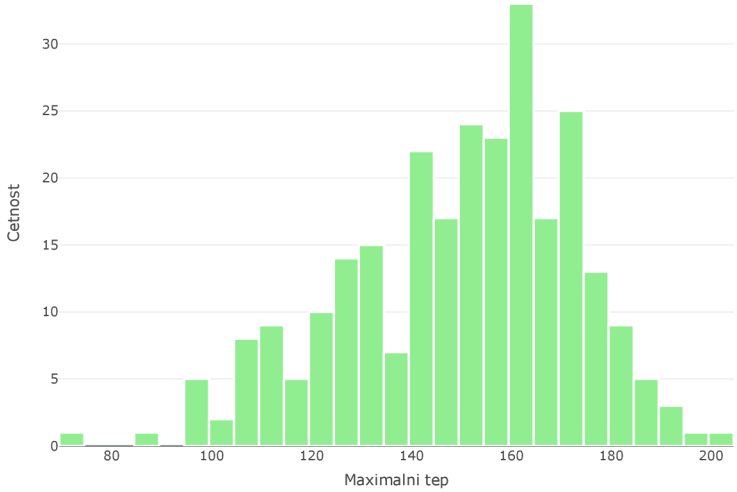


Obrázek 3.6: Printscreen html verze histogramu maximálního naměřeného tepu u mužů vykresleném v plotly po odkliknutí čtverečku nepožadované kategorie

Sturgesovo pravidlo v plotly

Jak již bylo zmíněno, výchozí nastavení v plotly knihovně pro konstrukci sloupců je složitější než v předchozích knihovnách. Počet sloupců určují parametry `start`, `size` a `end` v rámci parametru `xbins`. Výchozí nastavení parametru `start` je dáno tak, že bere jako počátek histogramu nejmenší hodnotu v datech, kterou případně upraví zaokrouhlením dolů na nějakou hezkou "kulatou" hodnotu. Výchozí nastavení parametru `size` říká, že pokud je parametr `nbinsx` roven nule nebo není uveden, vybere se opět nějaké hezké kulaté číslo určující velikost intervalu tak, aby došlo k celkové minimalizaci rozdílu mezi empirickým histogramem a hustotou vizualizované proměnné. Pokud je parametr `nbinsx` nastavený na konkrétní číslo, vykreslí se takový počet sloupců, který není větší než to dané číslo. Parametr `end` nastavuje koncovou hodnotu histogramu na ose x, kdy hrana sloupce nemusí končit přímo na této hodnotě, ale může končit za ní, což je způsobeno tím, že od `start` posunujeme o velikost danou parametrem `size` do té doby, než skončíme na maximální hodnotě dat (to je ta výchozí hodnota parametru `end`), nebo za ní [13].

Histogram konstruovaný tímto způsobem lze vidět na obrázku 3.7. Jedná se o histogram vycházející z 270 pozorování spojité proměnné naměřené maximální hodnoty tepu. Celkem jsme dostali 27 sloupců, což je pro nás z hlediska tohoto algoritmu optimum.



Obrázek 3.7: Histogram maximálního naměřeného tepu se sloupci vykreslenými pomocí výchozího nastavení v plotly

Stejně jako v knihovně ggplot je možné výchozí nastavení vykreslení sloupců změnit na Sturgesovo pravidlo (nebo jakékoliv jiné podle preferencí). Pro konstrukci histogramu s použitím Sturgesova pravdila budeme postupovat následovně:

1. Sestavíme kód pro tvorbu jednoduchého histogramu

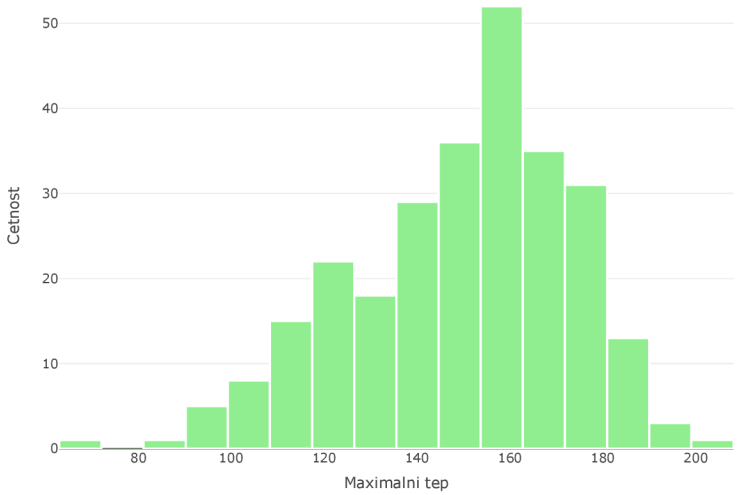
```
hist_stur <- plot_ly(data, x = ~Max.HR, type = "histogram",
 marker = list(color = "lightgreen",
  line = list(color = "white", width = 2))) %>%
layout(title = "", xaxis = list(title = "Maximalni tep"),
 yaxis = list(title = "Cetnost"))
```

2. Přidáme parametr `xbins` k nastavení způsobu konstrukce sloupců histogramu, za který dosadíme sturgesovo pravidlo $1 + log_2 n$

```
hist_stur<- plot_ly(data, x = ~Max.HR, type = "histogram",
 xbins=list(size = 1 + log2(length(data$Max.HR))),
 marker = list(color = "lightgreen",
```

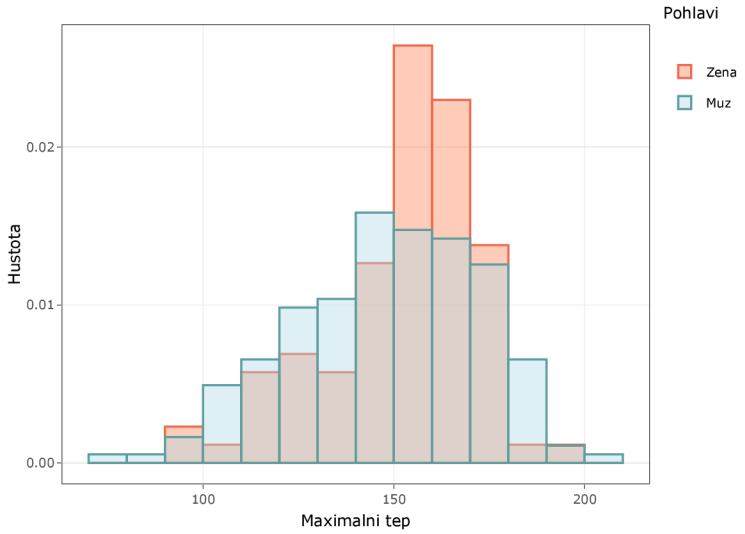
```
    line = list(color = "white", width = 2))) %>%
  layout(title = "", xaxis = list(title = "Maximalni tep"),
    yaxis = list(title = "Cetnost"))
```

3. Spustíme `hist_stur` a dostaneme histogram na obrázku 3.8



Obrázek 3.8: Histogram maximálního naměřeného tepu s použitím Sturgesova pravidla v plotly

Mimo tento postup existuje v prostředí plotly menší vychytávka, která umožní přeměnit náš již vykreslený histogram v ggplotu (třeba právě s již použitou Sturgesovou metodou) na interaktivní. Stačí použít příkaz `ggplotly(...)`, do kterého dosadíme název, pod kterým máme uložený histogram v ggplotu. Pro ukázku, pro převedení obrázku 3.2 z ggplotu do plotly dosadíme do příkazu název, pod kterým si uložíme graf `ggplotly("norm_hist")` a po spuštění dostaneme graf na následujícím obrázku 3.9.



Obrázek 3.9: Histogram maximálního naměřeného tepu vykreslený pro ženy a muže převedený z ggplotu do plotly

3.2. Jádrový odhad hustoty

3.2.1. Jádrový odhad hustoty v R

Základní způsob zadání kódu pro vykreslení jádrového odhadu hustoty v běžné knihovně softwaru R je dán tímto příkazem:

```
plot(density(x,...),...)
```

pokud bychom chtěli vybarvenou plochu pod křivkou hustoty, přidali bychom příkaz:

```
polygon(density(x,...),...)
```

`plot(...)` - slouží jednak pro vykreslení jádrového odhadu hustoty, druhak pro nastavení typických parametrů grafů, jako jsou názvy os, název grafu, k jejichž zadávání využíváme parametry zmíněné v části o histogramech v R 3.1.1, nebo vybarvení křivky hustoty pomocí parametru `col` a případně určení tloušťky křivky hustoty parametrem `lwd`

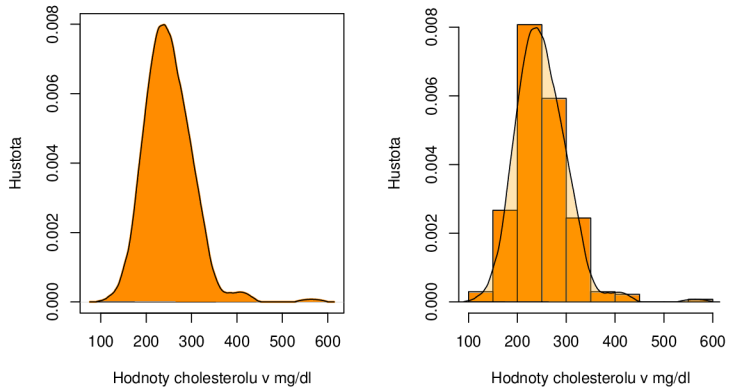
`density(x,...)` - hlavní část kódu, která definuje, jakým způsobem chceme hustotu veličiny odhadovat, kromě vektoru hodnot `x` proměnné, které chceme vykreslit obsahuje 2 důležité parametry:

- `bw` - vyhlazovací parametr, za který můžeme dosazovat buď přímo číselnou hodnotu vyhlazovacího parametru, nebo známé metody pro určení vyhlazovacího parametru, přičemž máme k dispozici metody (zadávají se přímo s uvozovkami jak jsou uvedeny):
  - `"nrd0"` - referenční metoda výchozího nastavení, která místo $h = 1{,}06\hat{\sigma}n^{-\frac{1}{5}}$ používá modifikaci $h = 0{,}9 * min\{\hat{\sigma}, (\frac{IQR}{1{,}34})\}n^{-\frac{1}{5}}$
  - `"nrd"` - referenční metoda, která používá $h = 1{,}06\hat{\sigma}n^{-\frac{1}{5}}$
  - `"SJ"` - plug-in metoda
  - `"ucv"` - nevychýlené křížové ověřování

  - `"bcv"` - vychýlené křížové ověřování

- `kernel` - parametr pro volbu jádrové funkce, kterou chceme použít pro odhad hustoty veličiny, na výběr máme mezi těmito jádry (psány s uvozovkami):
  - `"gaussian"` - používá se ve výchozím nastavení
  - `"epanechnikov"`
  - `"biweight"`
  - `"rectangular"`
  - `"triangular"`
  - `"cosine"`
  - `"optocosine"`

Na následujícím obrázku 3.10 jsem vyzkoušela vykreslit jádrový odhad hustoty proměnné množství cholesterolu v mg/dl s využitím Epanechnikova jádra a vyhlazovacím parametrem získaným pomocí plug-in metody, následně jsem pro ukázku tuto hustotu porovnala v histogramem téže proměnné.



Obrázek 3.10: Jádrový odhad hustoty proměnné množství cholesterolu v mg/dl (vlevo) s Epanechnikovým jádrem a "SJ"metodou vyhlazovacího parametru a histogram vykreslený s touto hustotou pro porovnání (vpravo)

Postup pro vykreslení grafů na obrázku 3.10 je dán následujícími body:

1. Vykreslíme křivku hustoty proměnné množství cholesterolu v mg/dl, kde pod `Cholesterol` je uloženo `data$Cholesterol`

   ```
   plot(density(Cholesterol, bw = "SJ", kernel = "epanechnikov"),
    main = "", xlab = "Hodnoty cholesterolu v mg/dl",
    ylab = "Hustota", lwd = 2, col = "darkorange",)
   ```

2. Následně vybarvíme plochu pod křivkou

   ```
   polygon(density(Cholesterol, bw = "SJ",
    kernel = "epanechnikov"), col = "darkorange")
   ```

3. Pro vytvoření histogramu s jádrovým odhadem hustoty proměnné množství cholesterolu v mg/dl, nejprvne vytvoříme histogram

   ```
   hist(Cholesterol, freq = FALSE, main = "",
    xlab = "Hodnoty cholesterolu v mg/dl", ylab ="Hustota",
    col = "darkorange")
   ```

4. Následně přidáme jádrový odhad hustoty s vybarvenou plochou pod křivkou této hustoty pomocí:

   ```
   polygon(density(Cholesterol, bw="SJ",
    kernel = "epanechnikov"), col = rgb(1,0.647,0,0.2))
   ```

5. Pro přidání výsledných grafů do jednoho obrázku 3.10 použijeme příkaz `par(mfrow=c(1,2))`, kde takto zadaný parametr `mfrow` říká, že požadujeme do řádku vykreslit grafy tak, aby tento řádek byl rozdělený na dva sloupce

Pokud bychom chtěli porovnávat jádrový odhad hustoty kvantitativní proměnné (zde opět proměnná množství cholesterolu v mg/dl) z hlediska jednotlivých kategorií kvalitativní proměnné (zde pro muže a ženy), využijeme následující postup, u kterého jsem nechala jádro i vyhlazovací parametr ve výchozím nastavení:

1. Vytvoříme jádrový odhad hustoty pro kategorii mužů, kde pod `Cholesterol` je opět uloženo `data$Cholesterol`

   ```
   plot(density(Cholesterol[which(data$Sex=="Muz")]),
    main = "",xlab = "Cholesterol v mg/dl",
    ylab= "Hustota", col = rgb(0.56, 0.85, 0.85,1), lwd=2)
   ```

2. Vybarvíme plochu pod křivkou hustoty u mužů

   ```
   polygon(density(Cholesterol[which(data$Sex=="Muz")]),
    col=rgb(0.56, 0.85, 0.85,1))
   ```

3. Připojíme k jádrovému odhadu hustoty pro kategorii mužů jádrový odhad hustoty pro kategorii žen

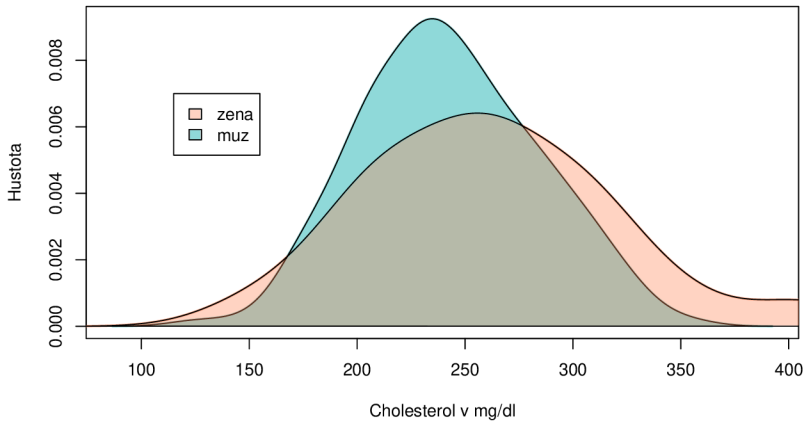
   ```
   polygon(density(Cholesterol[which(data$Sex == "Zena")]),
    col = rgb(1, 0.50, 0.31,0.35))
   ```

4. Nakonec přidáme legendu

   ```
   legend(115, 0.007, legend = c("zena","muz"),
    fill = c(rgb(1, 0.50, 0.31, 0.35), rgb(0.56, 0.85, 0.85, 1)))
   ```

   - `legend(115, 0.007)` - legenda je zde nastavena tak, aby se vykreslila v místě daném osou x = 115 a osou y = 0.007

5. Výsledný graf získaný těmito příkazy vidíme na obrázku na následující straně 3.11



Obrázek 3.11: Jádrový odhad hustoty proměnné množství cholesterolu v mg/dl vykreslená zvlášť pro kategorii mužů a kategorii žen

3.2.2. Jádrový odhad hustoty v ggplotu

Základní vazba pro vykreslení jádrových odhadů hustot v knihovně `ggplot2` je následující:

`ggplot(...) + geom_density(...)`, kde

`ggplot(...)` - 1. část kódu, za kterou dosazujeme stejné parametry, jaké byly zmíněny v kapitole u histogramů v ggplotu 3.1.2

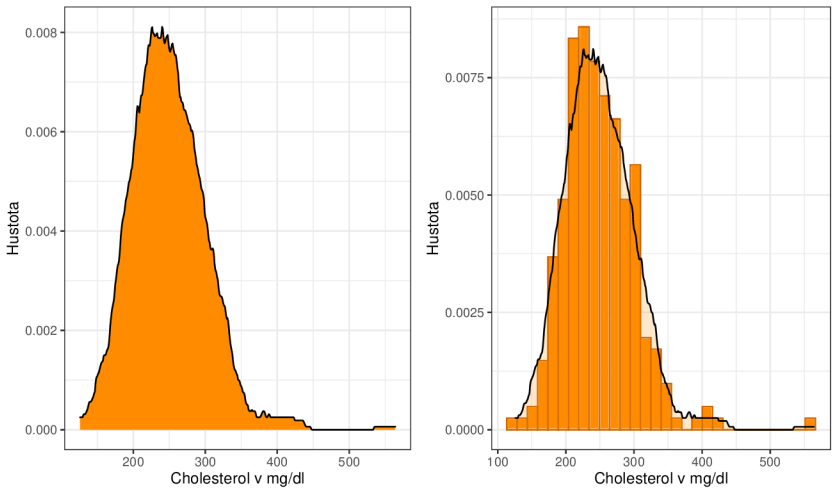
`geom_density(...)` - 2. část kódu, kam kromě estetických parametrů jako jsou `fill`, `col` nebo `lwd` dosazujeme tyto důležité parametry:

- `bw` - vyhlazovací parametr, za který je možné dosazovat buď přímo zvolenou hodnotu, nebo stejné metody výpočtu vyhlazovacího parametru jaké máme k dispozici v běžné knihovně softwaru R (`"nrd0"` - výchozí nastavení, `"nrd", "SJ", "ucv", "bcv"`)

- `adjust` - parametr sloužící k úpravě vyhlazovacího parametru tak, že tento parametr "multiplikuje"(tzn. vezme hodnotu našeho vyhlazovacího parametru, který následně pronásobí libovolně zvolenou konstantou)

- `kernel` - jádrová funkce, za kterou lze dosazovat stejná jádra jako ta, která máme k dispozici v běžné knihovně R (`"gaussian"` - výchozí nastavení, `"epanechnikov"`, `"biweight"`, `"rectangular"`, `"cosine"`, `"triangular"`, `"optocosine"`)

Na ukázku uvádím na následujícím obrázku 3.12 jádrový odhad hustoty proměnné množství cholesterolu v mg/dl vykreslenou v knihovně `ggplot2`, pro změnu s použitím Obdélníkového jádra s vyhlazovacím parametrem získaným metodou vychýleného křížového ověřování, a histogram s touto hustotou pro jejich porovnání



Obrázek 3.12: Jádrový odhad hustoty proměnné množství cholesterolu v mg/dl v ggplotu s Obdélníkovým jádrem a "bcv"metodou vyhlazovacího parametru (vlevo) a histogram pro porovnání s touto hustotou (vpravo)

Konstrukce grafů na obrázku 3.12 se skládá z následujících příkazů:

1. Vytvoříme jednoduchou hustotu proměnné množství cholesterolu v mg/dl

```
h1 <- ggplot(data, aes(x = Cholesterol)) +
geom_density(kernel = "rectangular", bw= "bcv",
  fill = "darkorange", col = "black") +
ggtitle("") + xlab("Cholesterol v mg/dl") +
```

```
ylab("Hustota") + theme_bw()
```

2. Vytvoříme histogram proměnné množství cholesterolu v mg/dl

```
h2 <- ggplot(data, aes(x = Cholesterol)) +
geom_histogram(aes(y= ..density..), col = "darkorange3",
 fill = "darkorange", show.legend = FALSE) +
xlab("Cholesterol v mg/dl") + ggtitle("") +
ylab("Hustota") + theme_bw()
```

3. Přidáme k němu hustotu

```
h2 <- h2 + geom_density(alpha = 0.2, bw = "bcv",
 kernel = "rectangular", fill = "darkorange", col = "black")
```

4. Nakonec spojíme oba grafy, s využitím knihovny `library(gridExtra)`, dohromady a dostaneme dvojici grafů z obrázku 3.12

```
grid.arrange(h1, h2, nrow = 1)
```

I v ggplotu existuje možnost porovnávat jádrový odhad hustoty kvantitativní proměnné při různých kategorií kvalitativní proměnné. Na obrázku 3.13 je tento jádrový odhad hustoty k nahlédnutí, přičemž jde opět o hustotu proměnné množství cholesterolu v mg/dl vykreslená zvlášť pro kategorii mužů a kategorii žen (vyhlazovací parametr i jádro zde nechávám ve výchozím nastavení).

Konstrukce jádrových odhadů hustot na obrázku 3.13 je popsána v následujících bodech:

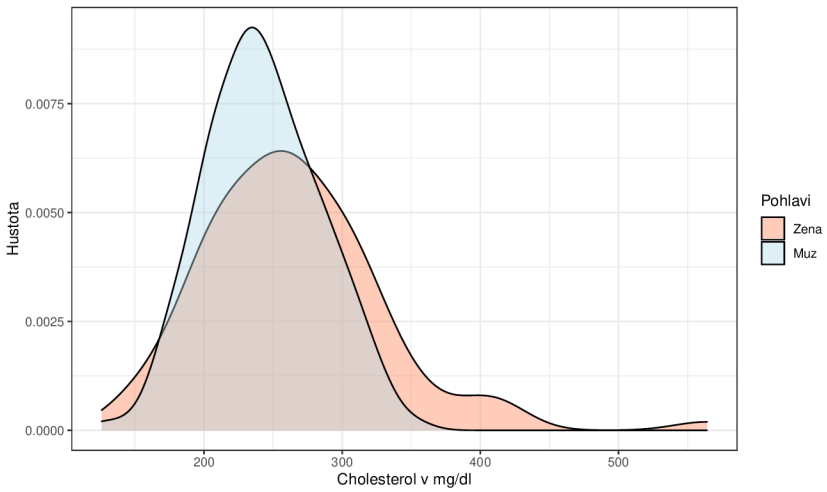
1. V první řadě jsem si opět vytvořila, pro přehlednost a konzistentnost popisu, tabulku prostřednictvím příkazu `data.frame()`

```
data <- data.frame(pohlavi = data$Sex,
 Cholesterol = data$Cholesterol)
```

2. Pro následnou konstrukci jádrových odhadů hustot dvou kategorií (mužů a žen) kvantitativní proměnné (množství cholesterolu v mg/dl) nám stačí jeden příkaz

```
ggplot(data, aes(x = Cholesterol, fill = pohlavi)) +
geom_density(alpha=0.4) + ggtitle("") +
xlab("Cholesterol v mg/dl") + ylab("Hustota") + theme_bw() +
scale_fill_manual(values = c("coral","lightblue"),
 name = "Pohlavi", labels = c("Zena","Muz")) +
scale_colour_manual(values = c("coral2","cadetblue"),
 name = "Pohlavi", labels = c("Zena","Muz"))
```

3. Po spuštění příkazu dostaneme následující obrázek 3.13



Obrázek 3.13: Jádrový odhad hustoty proměnné množství cholesterolu v mg/dl vykreslená v ggplotu zvlášť pro kategorii mužů a kategorii žen

3.2.3. Jádrový odhad hustoty v plotly

Pro vykreslení jádrového odhadu hustoty v knihovně `plotly` musí základní příkaz vypadat takto:

```
plot_ly(data, x, y, type = "scatter", mode = "lines",...)
```

`x` a `y` zde tvoří souřadnice bodu, kdy za `x` jsou dosazovány naměřené hodnoty veličiny a za `y` jsou dosazovány hodnoty hustoty veličiny

Plotly nemá možnost jádrový odhad hustoty zadávat přímo za parametr `type`. Proto se zde využívá kombinace těchto parametrů:

- `type = "scatter"` - takto nastavený parametr nám vykreslí body `[x,y]`
- `mode = "lines"` - parametr, který umožní propojení jednotlivých bodů křivkou `[x,y]`

Vykreslení jádrového odhadu hustoty je zde tedy trochu komplikovanější. Pro její samostatné vykreslení uvádím následující postup, kde jsem použila Trojúhelníkové jádro s vyhlazovacím parametrem získaným metodou nevychýleného křížového ověřování:

1. Zjistíme si hodnoty jádrového odhadu hustoty proměnné (zde opět množství cholesterolu v mg/dl), a uložíme

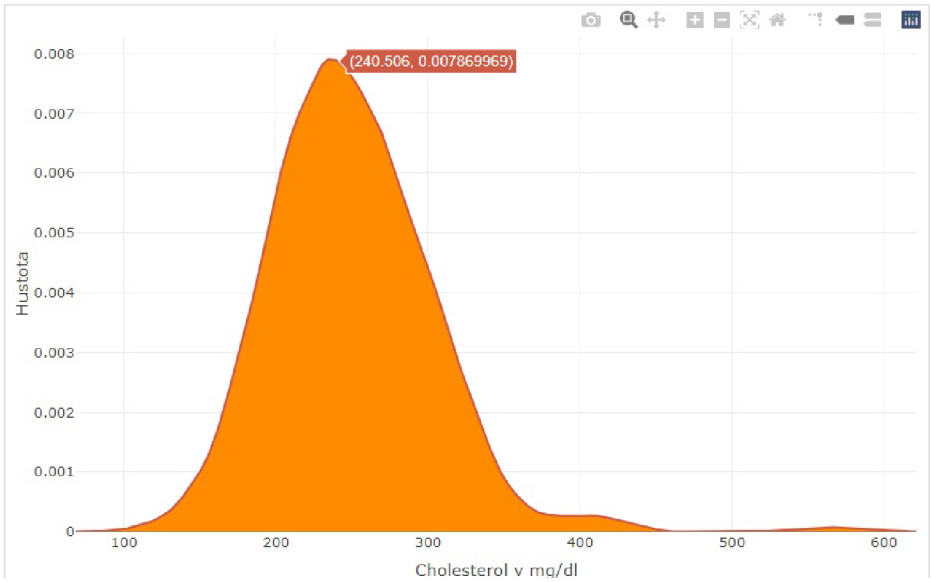
   ```
   hustota <- density(data$Cholesterol, kernel = "triangular",
    bw = "ucv")
   ```

2. Následně můžeme vytvořit jádrový odhad hustoty tímto způsobem

   ```
   plot_ly(data, x = ~hustota$x, y = ~hustota$y, type = "scatter",
    line = list(color = "rgba(205,91,69,1)"), mode = "lines",
    fill = "tozeroy", fillcolor = "darkorange") %>%
   layout(xaxis = list(title = "Cholesterol v mg/dl"),
    yaxis = list(title = "Hustota"))
   ```

- `fill = "tozeroy"` - parametr, který umožňuje vybarvit plochu pod křivkou jádrového odhadu hustoty

3. Výsledný graf vidíme na následujícím printscreenu 3.14, kde jsem pro ukázku interaktivity nechala zobrazit najetím myši na křivku jádrového odhadu hustoty její hodnotu pro $x = 240{,}506$



Obrázek 3.14: Printscreen jádrového odhadu hustoty proměnné množství cholesterolu v mg/dl vykreslené v plotly s ukázkou interaktivity

Vykreslení histogramu spolu s jádrovým odhadem hustoty je v plotly opět trochu složitější. Pro ukázku porovnání histogramu s jádrovým odhadem hustoty jsem použila opět proměnnou množství cholesterolu v mg/dl s Trojúhelníkovým jádrem a vyhlazovacím parametrem získaným metodou nevychýleného křížového ověřování. Postup je dán těmito body:

1. Uložíme si, pro jednoduchost zadávání do kódu, jádrový odhad hustoty pod jeden název

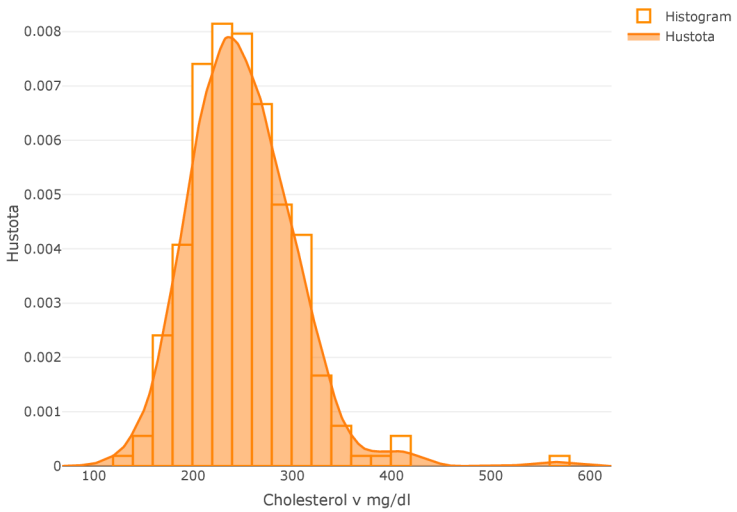
```
hustota <- density(data$Cholesterol, kernel = "triangular",
 bw = "ucv")
```

2. Vytvoříme příkaz pro vykreslení histogramu spolu s jádrovým odhadem hustoty

```
plot_ly(data, x = ~Cholesterol) %>%
add_trace (type = "histogram", name = "Histogram",
 histnorm = "probability density",
 marker = list(color = "white",
  line = list(color = "rgba(255,140,0,1)", width = 2))) %>%
add_trace(x = ~hustota$x, y = ~hustota$y, type = "scatter",
 mode = "lines", fill = "tozeroy", name = "Hustota") %>%
layout(xaxis = list(title = "Cholesterol v mg/dl"),
 yaxis = list(title = "Cetnost"))
```

   - `add_trace` - parametr, kterým lze přidávat libovolné typy grafů dané proměnné do jednoho obrázku
   - `add_trace(name =...)` - za parametr `name` zde uvádíme názvy, které chceme aby se zobrazily v legendě

3. Výsledný histogram spolu s jádrovým odhadem hustoty vidíme na následujícím obrázku 3.15



Obrázek 3.15: Histogram s jádrovým odhadem hustoty proměnné množství cholesterolu v mg/dl vykreslený v plotly

V případě, že požadujeme vykreslit jádrový odhad hustoty kvantitativní proměnné pro různé kategorie kvalitativní proměnné, můžeme použít následující příkazy:

1. Nejprve, pro zjednodušení, uložíme jádrové odhady hustot (ve výchozím nastavení jádra i vyhlazovacího parametru) pro muže a ženy do jednoho názvu pro každou kategorii

```
hustota1 <- density(data$Cholesterol[which(data$Sex=="Muz")])
hustota2 <- density(data$Cholesterol[which(data$Sex=="Zena")])
```

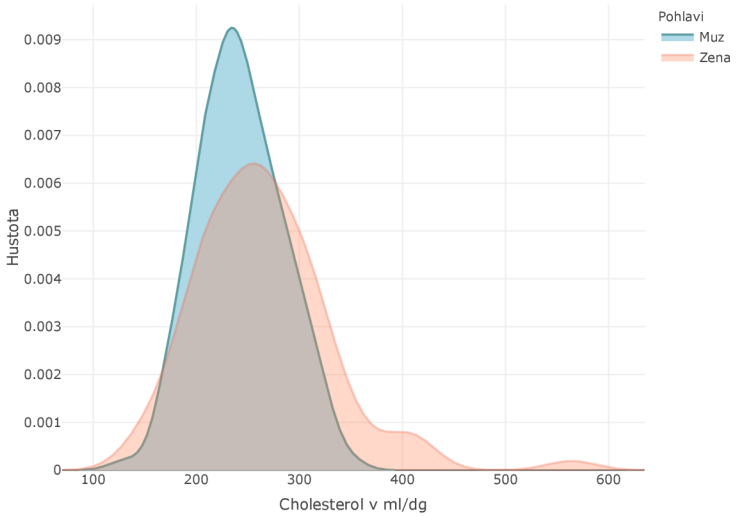
2. Následně sestrojíme jádrový odhad hustoty proměnné množství cholesterolu v mg/dl zvlášť kategorie mužů a žen

```
plot_ly(data, x = ~hustota1$x, y = ~hustota1$y,
 type = "scatter", mode = "lines", name = "Muz",
 fill = "tozeroy",
 line = list(color = "rgba(95,158,160,1)", width = 2),
  fillcolor= "rgba(173,216,230,1)") %>%
```

```
add_trace(x = ~hustota2$x, y = ~hustota2$y, name = "Zena",
  line = list(color = "rgba(238,106,80,0.3)", width = 2),
  fillcolor = "rgba(255,127,80,0.3)", fill = "tozeroy") %>%
layout(xaxis = list(title = "Cholesterol v ml/dg"),
  yaxis = list(title = "Hustota"),
  legend = list(title = list(text = "Pohlavi")))
```

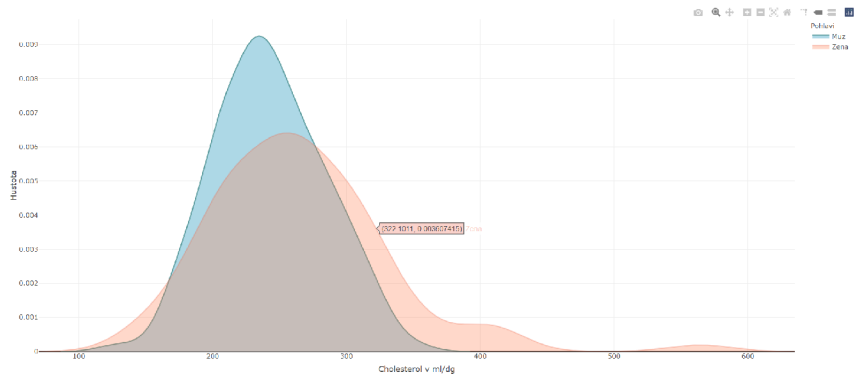
- `add_trace` - parametr, který slouží pro přidání dalšího grafu libovolného typu, kdy pokud přidáváme stejný typ grafu, jako byl ten předešlý, nemusíme znovu psát, o který typ grafu se jedná (proto při přidávání hustoty pro kategorii žen nebylo potřeba znovu zadávat parametry `mode` a `type`)

3. Výsledek po spuštění kódů vidíme na obrázku 3.16



Obrázek 3.16: Jádrový odhad hustoty proměnné množství cholesterolu v mg/dl vykreslená v plotly zvlášť pro kategorii mužů a kategorii žen

Interaktivní verzi vidíme na následujícím obrázku 3.17, která je opět k dispozici na CD pod názvem "kat_hust".



Obrázek 3.17: Jádrový odhad hustoty proměnné množství cholesterolu v mg/dl vykreslená v plotly zvlášť pro kategorii mužů a kategorii žen v html verzi

3.3. Box-plot

3.3.1. Box-plot v R

Základní vazba pro jednoduchý boxplot je v běžné knihovně softwaru R následující:

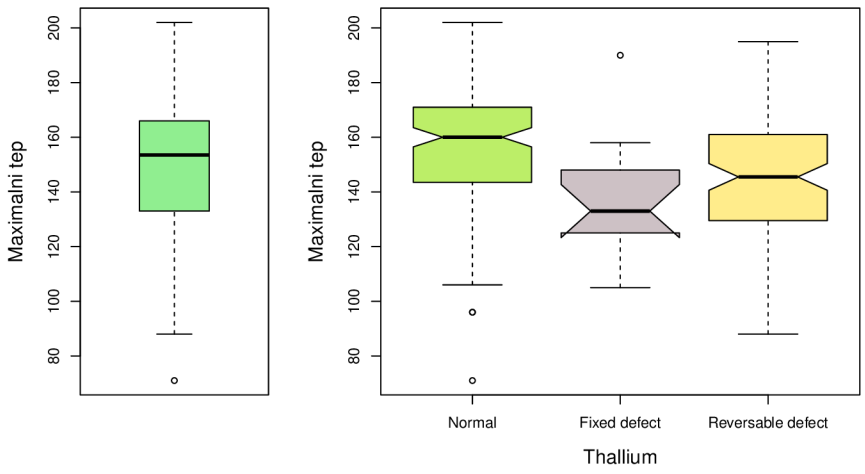
`boxplot(x,...)`, kde

`x` - vektor hodnot, které chceme vykreslit

V boxplotu je nejpoužívanější (a dalo by se říct, že i nejdůležitější při porovnávání více boxplotů mezi sebou) tento parametr:

`notch = T` - parametr sloužící ke znázornění intervalového odhadu mediánu (ve výchozím nastavení je `= FALSE`)

Na následujícím obrázku 3.18 můžeme vidět boxplot pro jednu kvantitativní proměnnou (zde proměnná maximální naměřené hodnoty tepu) a vedle něj boxplot této proměnné při různých kategoriích kvalitativní veličiny (zde množství thallia v těle).



Obrázek 3.18: Boxplot proměnné maximální naměřené hodnoty tepu (vlevo) a boxplot této proměnné vykreslený zvlášť pro kategorie proměnné Thallium (vpravo)

U boxplotu kategorie "Fixed defect" vidíme, že intervalový odhad je zvláštně zobrazený. Důvodem je to, že tento intervalový odhad je větší, než krabice této kategorie.

Pro konstrukci boxplotů na obrázku 3.18 využijeme následující postup:

1. V první řadě, pro finální uspořádání boxplotů do jednoho obrázku využijeme příkaz `layout(matrix(c(...),...))`

   ```
   layout(matrix(c(1,2,1,2), 2, 2, byrow = TRUE), widths = lcm(7))
   ```

   - `matrix(...)` slouží pro maticové zadání obsazení plochy jednotlivými grafy, přičemž zde je matice nastavena tak, aby se první graf vykreslil na první pozici 1. a 2. řádku a druhý graf na druhou pozici 1. a 2. řádku, dvě dvojky za `c(...)` znamenají, že budeme obsazovat 2 sloupce a 2 řádky a parametr `byrow = TRUE` říká, že matice je vyplňována po řádcích (v případě výchozího `FALSE` by byla vyplňována po sloupcích)

2. Vytvoříme jednoduchý boxplot pro jednu kvantitativní proměnnou (maximální naměřené hodnoty tepu), kde pod `Max.HR` je uloženo `data$Max.HR`

   ```
   boxplot(Max.HR, main = "", ylab = "Maximalni tep",
     col = "lightgreen")
   ```

3. Vytvoříme boxplot téže proměnné při různých kategoriích kvalitativní proměnné Thallium, kde pod `Thallium` je uloženo `data$Thallium`

   ```
   boxplot(Max.HR~Thallium, notch = T, main = "", xlab = "Thallium",
     ylab = "Maximalni tep", par(cex.axis = 0.9, cex.lab = 1.3),
     col = c("darkolivegreen2","lavenderblush3","lightgoldenrod1"))
   ```

   - `Max.HR~Thallium` - tento zápis znamená, že vytváříme boxplot kvantitativní proměnné Max.HR pro jednotlivé kategorie proměnné Thallium

- `par(...)` - parametr pro grafickou úpravu
  - `cex.axis` - nastavuje velikost názvů kategorií pod jednotlivými boxploty
  - `cex.lab` - nastavuje velikost názvů os

4. Po spuštění příkazů získáme boxploty ve stejném uspořádání jako vidíme na obrázku 3.18

3.3.2. Box-plot v ggplotu

Pro tvorbu jednoduchého boxplotu nám v knihovně `ggplot2` použijeme příkaz:

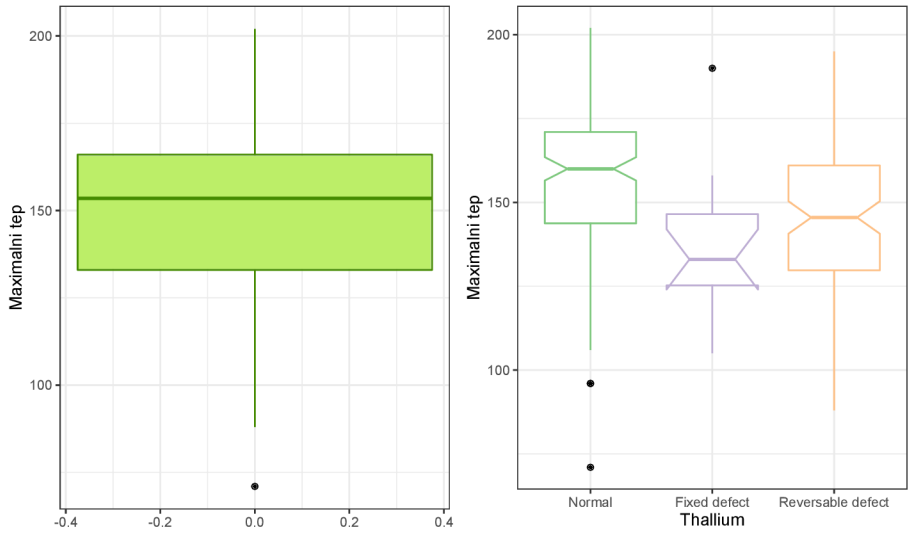
`ggplot(...) + geom_boxplot(...)`, kde

`ggplot(...)` - 1. část kódu, za kterou dosazujeme stejné parametry jako v kapitole o histogramech 3.1.2 s jednou výjimkou, a to je ta, že místo `aes(x =...)` budeme raději používat `aes(y =...)` (lze ale použít obě varianty, pokud bychom zadali proměnnou za `x`, byl by boxplot naležato, přehlednější je však, dle mého názoru, zadávání proměnné za `y`)

`geom_boxplot(...)` - 2. část kódu, kterou říkáme, že vykreslujeme box-plot, a do které zadáváme parametry týkající se přímo boxplotu, jako jsou estetické parametry pro barvení boxplotu, které už byly zmíněné u histogramů 3.1.2, nebo další parametry, ze kterých zmíním tyto často používané:

- `notch = TRUE` - parametr pro znázornění intervalového odhadu mediánu (ve výchozím nastavení je `= FALSE`)
- `outlier.colour` - parametr, kterým můžeme vybarvit extrémní hodnoty (”outliery”)
- `outlier.shape` - parametr měnící tvar extrémní hodnoty
- `outlier.size` - parametr měnící velikost velikost extrémní hodnoty

Na ukázku opět uvádím stejnou dvojici grafů 3.19, jakou jsme viděli na obrázku 3.18, tentokrát s použitím knihovny `ggplot2`.



Obrázek 3.19: Boxplot proměnné maximální naměřené hodnoty tepu (vlevo) a boxplot této proměnné vykreslený zvlášť pro kategorie proměnné Thallium (vpravo) v ggplotu

Tvorba boxplotů z obrázku 3.19 je dána následujícími body:

1. Vytvoříme boxplot pro kvantitativní proměnnou maximální naměřené hodnoty tepu

```
box <- ggplot(data, aes(y = Max.HR))
b1 = box + geom_boxplot(outlier.colour = "black",
  color = "chartreuse4", fill = "darkolivegreen2") +
ggtitle("") + theme_bw() + ylab("Maximalni tep")
```

2. Vytvoříme boxplot téže proměnné při různých kategorií proměnné Thallium

```
box2 <- ggplot(data, aes(y = Max.HR, x = Thallium,
  col = Thallium))
b2 = box2 + geom_boxplot(notch = T, show.legend = F,
  outlier.colour = "black") +
ggtitle("") + xlab("Thallium") + ylab("Maximalni tep") +
```

`theme_bw() + scale_color_brewer(palette = "Accent")`

- `x` - zde dosazujeme kategoriální proměnnou, pro jejíž kategorie budou vytvořeny boxploty
- `col = Thallium` - boxplot bude mít vybarvené hrany podle proměnné Thallium
- `scale_color_brewer(palette = "Accent")` - parametr umožňující použít škálu barev, přičemž zde je typu "Qualitative", což znamená, že každá kategorie má svou barvu (jiné možnosti škál jsou "Diverging"nebo "Sequential")

3. Nakonec spojíme předchozí příkazy s využitím knihovny `library(gridExtra)`

   `grid.arrange(b1, b2, nrow = 1)`

4. Po spuštění dostaneme grafy z obrázku 3.19

3.3.3. Box-plot v plotly

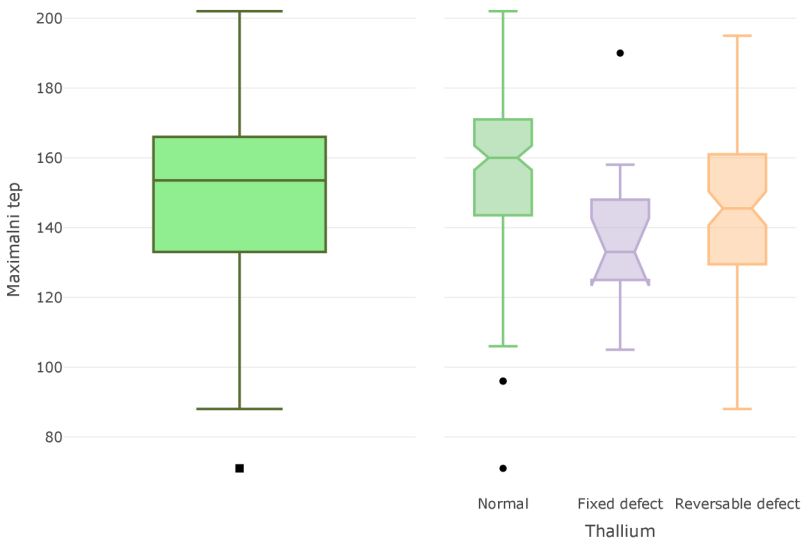
Pro tvorbu jednoduchého boxplotu v knihovně `plotly` používáme spojení:

`plot_ly(data, y =..., type = "box",...)`, kde

`data` - parametr, za který dosazujeme data, ze kterých čerpáme při konstrukci

`y` - parametr, za který dosazujeme název spojité proměnné, jejíž boxplot chceme vykreslit

Na následujícím obrázku 3.20 můžeme vidět boxplot proměnné maximální naměřené hodnoty tepu a boxplot této proměnné pro různé kategorie kvalitativní proměnné Thallium.



Obrázek 3.20: Boxplot proměnné maximální naměřené hodnoty tepu (vlevo) a boxplot této proměnné vykreslený zvlášť pro kategorie proměnné Thallium (vpravo) v plotly

Konstrukce grafů, které vidíme na obrázku 3.20 je popsána následujícími body:

1. Vytvoříme jednoduchý boxplot pro kvantitativní proměnnou maximální naměřené hodnoty tepu

```
b1 <- plot_ly(data, type = "box", y = ~Max.HR, x = "",
 notched = T, fillcolor = "lightgreen",
 line = list(color = c("darkolivegreen")),
 marker = list(symbol = "square-dot", outliercolor = "black",
  color = "black")) %>%
layout(title = "", xaxis = list(title = ""),
 yaxis = list(title = "Maximalni tep"), showlegend = FALSE)
```

   - `notched = TRUE` - parametr pro znázornění intervalového odhadu mediánu (ve výchozím nastavení `= FALSE`)

- `x = ""` - `x` je nastaveno tak, aby se nezobrazovalo v grafu
- `symbol = ""` - parametr, který slouží k nastavení tvaru odlehlých hodnot (zde chceme čtvereček), můžeme dosazovat přímo názvy symbolů, nebo číselné hodnoty jako u parametr `pch`, přičemž co se druhů tvaru týče, máme k dispozici hodnoty 0 až 18 (pod každou hodnotou je definovaný nějaký jeden tvar)
- `outlier.color` - parametr sloužící k vybarvení odlehlých hodnot

2. Vykreslíme boxplot téže proměnné pro kategorie kvalitativní proměnné Thallium

```
b2 <- plot_ly(data, type = "box", x = ~Thallium, y = ~Max.HR,
 color = ~Thallium, colors = "Accent",
 marker = list(color = "black"))%>%
layout(title = "", showlegend = FALSE,
 xaxis = list(title = "Thallium"))
```

   - `colors = "..."` - parametr, který slouží pro přidání vybrané palety barev (zde Accent, další možnosti palet jsou stejné jako v knihovně ggplot)
   - `showlegend = FALSE` - parametr zakazující zobrazení legendy (ve výchozím nastavení se legenda zobrazuje automaticky, tedy je nastaveno `showlegend = TRUE`)

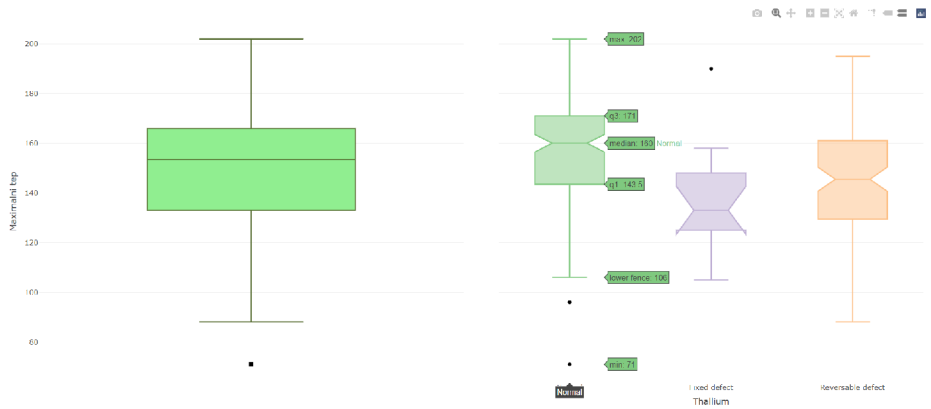
3. Oba grafy spojíme do jednoho obrázku pomocí příkazu `subplot()`

```
subplot(b1, b2, shareX = T, shareY = T)
```

   - `shareX = T` - parametr, který umožní zobrazit popisek osy x v suplotu
   - `shareY = T` - parametr, který umožní zobrazit popisek osy y v suplotu

4. Po spuštění subplotu dostaneme kombinaci grafů, jaké vidíme na obrázku 3.20

Printscreen obrázku 3.20 vidíme na následujícím obrázku 3.21, přičemž jeho interaktivní podoba je dostupná na přiloženém CD pod názvem "sub_box".



Obrázek 3.21: Boxplot proměnné maximální naměřené hodnoty tepu (vlevo) a boxplot této proměnné vykreslený zvlášť pro kategorie proměnné množství Thallia v těle (vpravo) v plotly v html verzi

3.4. Violin-plot

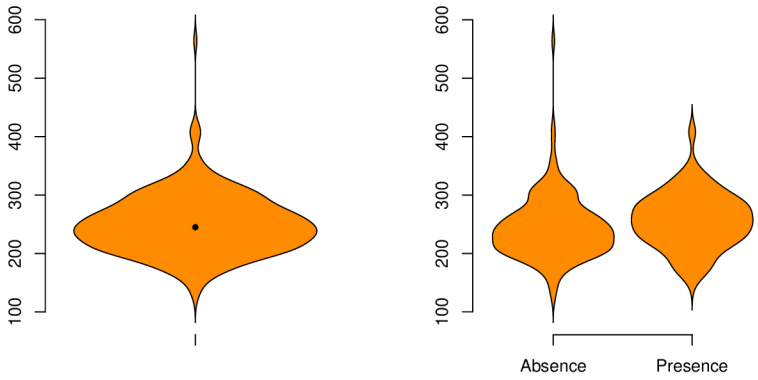
3.4.1. Violin-plot v R

Pro vykreslení violin-plotu v běžné knihovně softwaru R musíme použít knihovnu `UsingR`. Příkaz pro vykreslení violin-plotu bude potom vypadat takto:

`simple.violinplot(x, ...)`, kde

`x` - vektor hodnot proměnné, kterou chceme vykreslit

Nevýhoda používání violin-plotu v běžné knihovně R je ta, že violin-plot není možné vykreslit dohromady s boxplotem, proto v následujících krocích uvedu jen návod na vykreslení samostatného violin-plotu, a violin-plotu kvantitativní proměnné při různých kategoriích kvalitativní proměnné. Druhou nevýhodou je nemožnost rozsáhlejších grafických úprav (není možně vybarvit každou kategorii kvalitativní proměnné jinou barvou, nebo přidávat popisky os, příp. názvu grafu). Grafy, jejichž konstrukce bude popsána vidíme na obrázku 3.22



Obrázek 3.22: Violin-plot proměnné množství cholesterolu v mg/dl (vlevo) a violin-plot této proměnné vykreslený zvlášť pro kategorie proměnné výskytu srdečního onemocnění (vpravo)

Při konstrukci grafů na obrázku 3.22 budeme postupovat následovně:

1. Načteme knihovnu usingR příkazem

   `library(UsingR)`

2. Vytvoříme příkaz na uspořádání violin-plotů do jednoho obrázku

   `par(mfrow=c(1,2))`

3. Vytvoříme jednoduchý violin-plot proměnné množství cholesterolu v mg/dl, kde pod `Cholesterol` je uloženo `data$Cholesterol`

   `simple.violinplot(Cholesterol, col = "darkorange")`

   `points(median(Cholesterol), col = "black", pch = 20)`

   - `points()` - funkce, kterou můžeme ve violin-plotu dokreslit požadované charakteristiky (zde medián)
   - `pch` - parametr, kterým určujeme, jak bude vypadat vykreslený bod (typy, které jsou k dispozici jsou k nahlédnutí v softwaru R při zadání názvu funkce `points` do "Help")

4. Vytvoříme violin-plot téže proměnné pro kategorie kvalitativní proměnné výskytu srdečního onemocnění, kde pod `Heart.Disease` je uloženo `data$Heart.Disease`, přičemž výsledné spojení vidíme na obrázku 3.22

   `simple.violinplot(Cholesterol~Heart.Disease, col = "darkorange")`

3.4.2. Violin-plot v ggplotu

Příkaz pro vykreslení violin-plotu s využitím knihovny `ggplot2` je dán spojením:

`ggplot(...) + geom_violin(...)`, kde

`ggplot(...)` - obsahuje stejné parametry, jako ve všech předcházejících případech, s tím rozdílem, že teď budeme dosazovat za `x` i za `y`:

- `x` - parametr, za který dosadíme `x = 0`, abychom dostali graf symetrický kolem této přímky (můžeme zvolit ale jakoukoliv jinou hodnotu, podmínkou je, abychom za něj něco dosadili, jinak by se graf nevykreslil)
- `y` - za tento parametr dosazujeme název spojité proměnné, kterou chceme vykreslit

`geom_violin(...)` - 2.část kódu, kterou říkáme, že vykreslujeme violin-plot, a do které lze dosazovat parametry týkající se přímo violin-plotu

Pro vykreslení jednoduchého violin-plotu proměnné množství cholesterolu v mg/dl a violin-plotu téže proměnné při různých kategoriích kvalitativní proměnné výskytu srdečních onemocnění využijeme následující postup:

1. Vykreslíme jednoduchý violin-plot proměnné množství cholesterolu v mg/dl

```
ggplot(data, aes(y = Cholesterol, x = 0)) +
geom_violin(col = c("black"), fill = c("darkorange")) +
ggtitle("") + xlab("") + ylab("Cholesterol v mg/dl") +
stat_summary(fun = median, geom = "point", col = 'black',
  size = 3) +
theme_bw()
```

   - `stat_summary(...)` - funkce, která slouží k vykreslení doplňujících charakteristik (pomocí parametru `fun`) a jejich estetické úpravě
     - `geom` - parametr sloužící k přidání vykreslované charakteristiky (nejčastěji přidáváme median/mean), přičemž můžeme tyto charakteristiky vykreslit bodově pomocí `"point"` nebo čarou `"crossbar"`

2. Vykreslíme violin-plot předchozí proměnné při různých kategoriích kvalitativní proměnné výskytu srdečního onemocnění

```
ggplot(data, aes(y = Cholesterol, x = Heart.Disease,
 fill = Heart.Disease)) +
geom_violin() + ggtitle("") + xlab("Srdecni nemoc") +
 ylab("Cholesterol v mg/dl") +
scale_fill_manual(values = c("darkorange","tomato")) +
theme_bw() + theme(legend.position = "none")
```

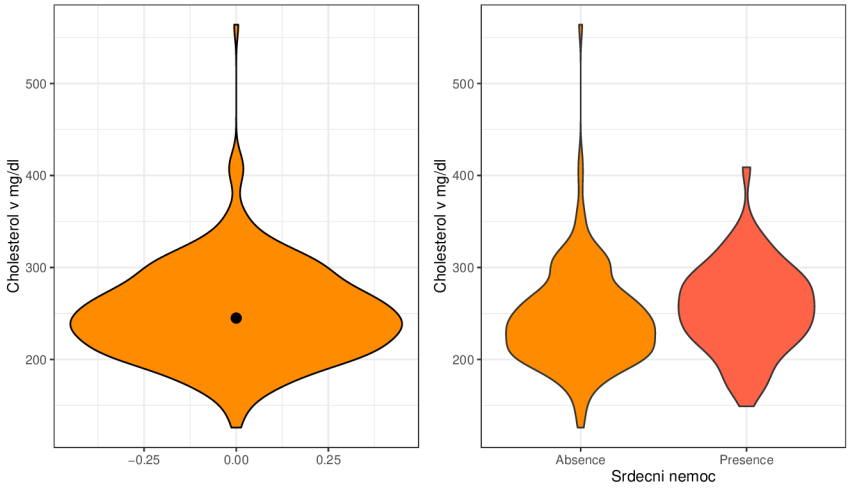
- `theme(legend.position = "none")` - část kódu, kterou odebereme legendu

3. Oba grafy spojíme s využitím knihovny `library(gridExtra)`

```
grid.arrange(v1,v2, ncol = 2)
```

- `v1` - název, pod který uložíme první graf
- `v2` - název, pod který uložíme druhý graf

4. po spuštění příkazů dostaneme dvojici grafů, kterou vidíme na následujícím obrázku 3.23



Obrázek 3.23: Violin-plot proměnné množství cholesterolu v mg/dl (vlevo) a violin-plot této proměnné vykreslený zvlášť pro kategorie proměnné výskytu srdečního onemocnění (vpravo) v ggplotu

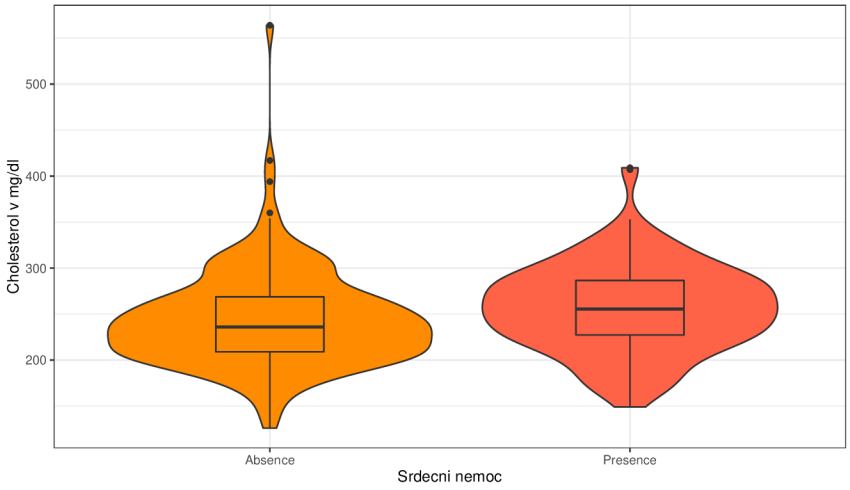
Pokud požedujeme vykreslit violin-plot s boxplotem dohromady, využijeme následující postup:

1. Vykreslení violin-plotu proměnné množství cholesterolu v mg/dl při různých kategoriích kvalitativní proměnné výskytu srdečního onemocnění

```
ggplot(data, aes(y = Cholesterol, x = Heart.Disease,
 fill = Heart.Disease)) +
geom_violin() + ggtitle("") + xlab("Srdecni nemoc") +
 ylab("Cholesterol v mg/dl") +
stat_summary(fun = "median", geom = "point",
 col = "black", size = 3) +
scale_fill_manual(values = c("darkorange","tomato")) +
theme_bw() + theme(legend.position = "none") +
geom_boxplot(width = 0.3)
```

- `width` - parametr upravující šířku boxplotu (bez zvolení by boxplot přesahoval violin-plot - nevypadá to esteticky dobře)

2. Po spuštění dostaneme graf na obrázku 3.24



Obrázek 3.24: Violin-plot proměnné množství cholesterolu v mg/dl při různých kategoriích kvalitativní proměnné výskytu srdečního onemocnění vykreslený spolu s boxploty v ggplotu

3.4.3. Violin-plot v plotly

Pro vykreslení violin-plotu v knihovně `plotly` použijeme tento příkaz:

`plot_ly(data, y =..., type = "violin",...)`, kde

`data` - parametr, za který dosazujeme ta data, ze kterých čerpáme

`y` - parametr, za který opět dosazujeme název spojité proměnné, jejíž violin-plot chceme vykreslit

V následujících krocích je popsán návod na vytvoření violin-plotu v plotly. Jelikož v plotly nelze vykreslit medián u violin-plotu, uvádím vedle obyčejného violin-plotu přímo kombinaci boxplotu a violin-plotu.

1. Vykreslíme obyčejný violin-plot proměnné množství cholesterolu v mg/dl

```
vp1 <- plot_ly(data, y = ~Cholesterol, type = "violin",
  x = "", fillcolor = "rgba(255,140,0,1)",
 line = list(color = "rgba(205,102,0,1)"),
 marker = list(color = "rgba(205,102,0,1)")) %>%
layout(yaxis = list(title = "Cholesterol v mg/dl"))
```

2. Vykreslíme violin-plot téže proměnné při různých kategorií proměnné výskytu srdečního onemocnění

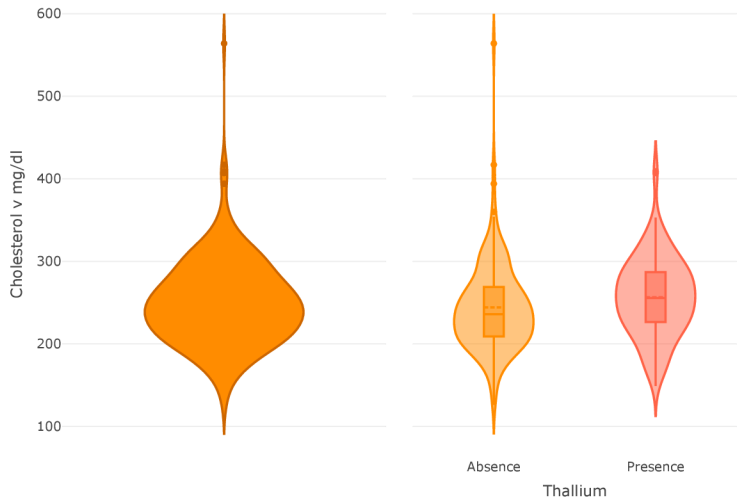
```
vp2 <- plot_ly(data, y = ~Cholesterol, x = ~Heart.Disease,
 type = "violin", box = list(visible = T),
 color = ~Heart.Disease,
 colors = c("darkorange","tomato")) %>%
layout(showlegend = F, xaxis = list(title="Thallium"),
 yaxis = list(title = "Cholesterol v mg/dl"))
```

   - `box = list(visible = T)` - parametr, který do příkazu dosazujeme tehdy, když chceme spolu s violin-plotem vykreslit i boxplot

3. Nakonec spojíme grafy do jednoho příkazem

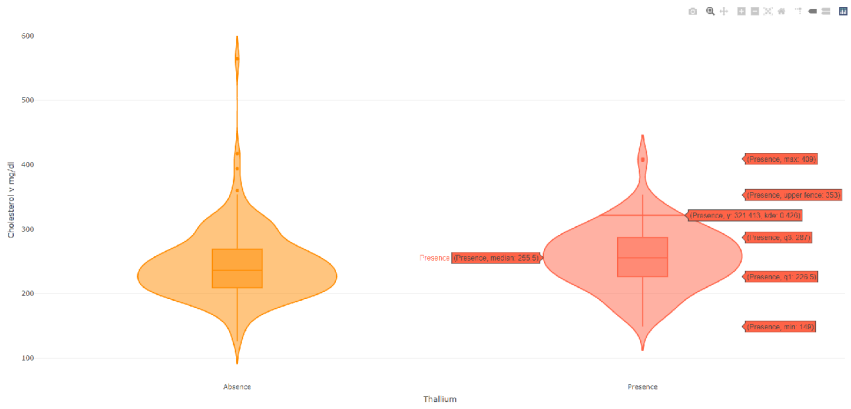
```
subplot(vp1, vp2, shareX = T, shareY = T)
```

4. Výsledek po spuštění příkazů vidíme na obrázku 3.25



Obrázek 3.25: Violin-plot proměnné množství cholesterolu v mg/dl (vlevo) a violin-plot této proměnné vykreslený zvlášť pro kategorie proměnné výskytu srdečního onemocnění (vpravo) v plotly

Printscreen verze obrázku 3.25 vidíme na obrázku 3.26, jehož interaktivní verze je dostupná na přiloženém CD pod názvem "int_viol2".



Obrázek 3.26: Html verze obrázku 3.25

3.5. Scatter-plot

3.5.1. Scatter-plot v R

Základní příkaz pro tvorbu scatter-plotů v běžné knihovně softwaru R vypadá následovně:

`plot(x, y,...)`, kde

`x` - parametr, za který dosazujeme název jedné kvantitativní proměnné (tzn. vektor jejích hodnot)

`y` - parametr, za který dosazujeme název druhé kvantitativní proměnné

Pro vykreslení základního scatter-plotu kvantitativních proměnných (zde věk a množství cholesterolu v mg/dl) při různých kategoriích kvalitativní proměnné (zde pohlaví) využijeme následující postup:

1. Vytvoříme scatter-plot, kde pod `Cholesterol` je, pro zejdnodušení zápisu (tak to bude i u dalších kódů v této sekci), uloženo `data$Cholesterol` a pod `Age` je uloženo `data$Age`

   ```
   plot(Age, Cholesterol, main = "", xlab = "Vek",
     ylab = "Hladina cholesterolu v mg/dl",
     col = c("coral2","lightblue")[factor(data$Sex)],
     pch = c(19,17)[data$Sex])
   ```

   - `c("coral2","lightblue")[factor(data$Sex)]` - takto zadané barvy značí, že vybarvujeme každou kategorii jinou barvou
   - `pch = c(19,17)[data$Sex])` - parametr, kterým navolíme tvar bodů, který bude různý pro různé kategorie

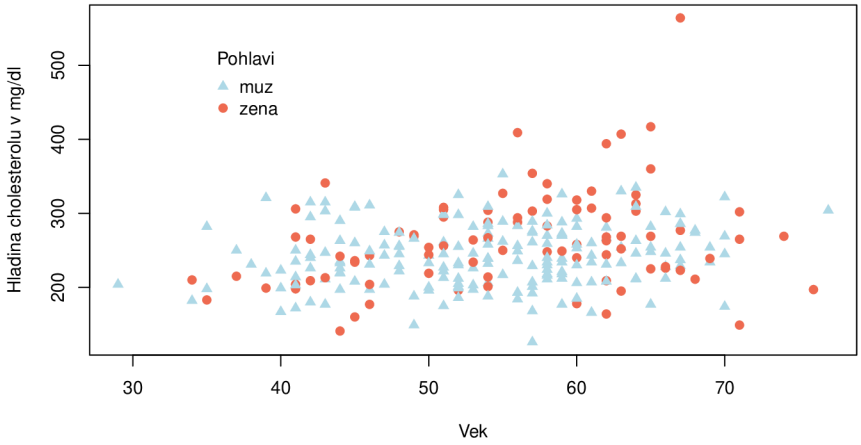
2. Vytvoříme legendu

   ```
   legend(x = 35, y = 530, title = "Pohlavi", c("muz","zena"),
   ```

```
bty= "n", pch = c(19,17)[data$Sex],
col = c("coral2","lightblue")[factor(data$Sex)])
```

- `bty = "n"` - parametr, kterým říkáme, že nechceme ohraničenou legendu

3. Po spuštění dostaneme scatterplot, který vidíme na obrázku 3.27



Obrázek 3.27: Scatter-plot kvantitativních proměnných věk a množství cholesterolu v mg/dl při různých kategoriích kvalitativní veličiny pohlaví

Obecně sice lze v běžné knihovně R vykreslit scatter-plot s vybarvením bodů dle třetí kvantitativní proměnné, ale barevnou škálu tato knihovna už sama nevykreslí. Tato nevýhoda se však dá obejít následujícím postupem:

1. Vytvoříme data.frame, pomocí kterého potom budeme tvořit barevnou škálu, přičemž první sloupce je tvořen samými jedničkami a druhý sloupec je vektor hodnot, který začíná na minimální hodnotě naší proměnné, ze které se po jedničce dostaneme do poslední hodnoty vektoru, a tou je maximální hodnota této proměnné

   ```
   a <-data.frame(each = 1, c(min(data$Max.HR):max(data$Max.HR)))
   ```

2. S využitím knihovny `library(RColorBrewer)` vytvoříme škálu barev, kde si navolíme libovolný počet barev, ze kterých se nám vytvoří škála, a tyto barvy rozdělíme pro 270 hodnot

```
barvy <- colorRampPalette(c("yellow", "orange",
 "pink", "red", "purple", "blue"))
 (length(seq(from = min(data$Max.HR), to = max(data$Max.HR),
  length = 270)))
```

3. Vytvoříme matici "místa"do kterého vložíme následně vytvořený graf s barevnou škálou

```
layout(matrix(c(1,2,0,0), 1, 2, byrow = TRUE), c(2.5,1), c(1,2.5))
```

4. Vytvoříme scatter-plot kvantitativních proměnných věk a množství cholesterolu v mg/dl s body vybarvenými podle hodnot kvantitativní proměnné maximální naměřené hodnoty tepu

```
plot(Age, Cholesterol, main = "", xlab = "Vek",
 ylab = "Hladina cholesterolu v mg/dl",
 cex.lab = 0.8, col = barvy[Max.HR], pch = 20)
```

   - `cex.lab` - parametr k nastavení velikosti písma na osách

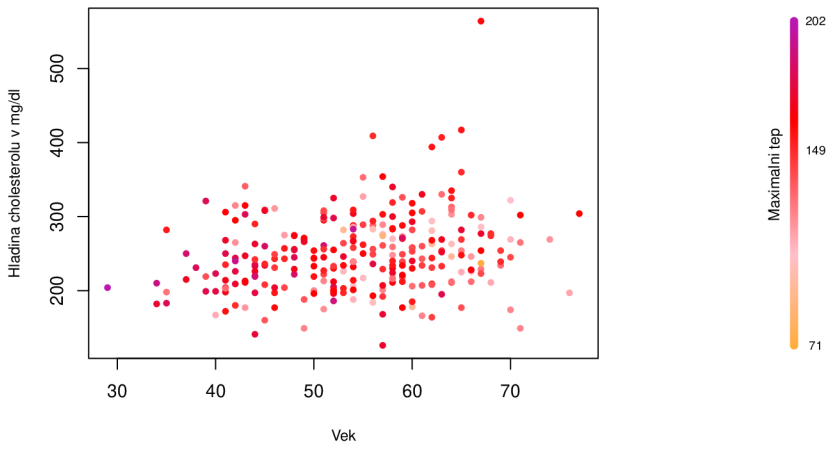
5. Vytvoříme barevnou škálu s popisky

```
plot(a, ylab = "", xlab = "", axes = F,
 yaxt = "n", xaxt = "n", pch = 16,
 col = barvy[c(min(data$Max.HR):max(data$Max.HR),
  length = 270)])
text(1.15,71,"71", cex = 0.7)
text(1.15,149.6778,"149", cex = 0.7)
text(1.15,202,"202", cex = 0.7)
```

```
text(0.87,140, cex = 0.8, srt = 90, "Maximalni tep")
```

- `yaxt = "n"` - parametr sloužící k odstranění osy y
- `xaxt = "n"` - parametr sloužící k odstranění názvu x
- `axes = F` - parametr, sloužící k odstranění "okénka"grafu
- `srt` - parametr, který říká, o kolik stupňů chceme otočit text

6. Výsledný graf vidíme na obrázku 3.28



Obrázek 3.28: Scatter-plot kvantitativních proměnných věk a množství cholesterolu v mg/dl s body vybarvenými podle kvantitativní proměnné maximální naměřené hodnoty tepu

Poslední konstrukcí, kterou u této knihovny zmíním, je možnost konstrukce matic scatter-plotů.

1. Pro konstrukci matic scatter-plotů nám v běžné knihovně R stačí tento jednoduchý příkaz, daný funkcí `pairs`

```
pairs(data[,c(1,4,5,8)], pch = 19, cex = 0.5, col = c("violet"))
```

- `data[,c(1,4,5,8)]` - výběr sloupců, ve kterých se vyskytují kvantitativní proměnné

2. Po spuštění dostaneme matici z obrázku 2.2

3.5.2. Scatter-plot v ggplotu

Pro tvorbu scatter-plotu v knihovně `library(ggplot2)` se používá příkaz:

`ggplot(...) + geom_point(...)`, kde

`ggplot(...)` - 1. část kódu obsahující tyto parametry:

- `data` - parametr, za který opět dosazujeme data, ze kterých čerpáme při konstrukci scatter-plotu
- `aes(...)`
  - `x` - parametr, za který dosazujeme název kvantitativní nezávislé proměnné
  - `y` - parametr, za který dosazujeme název kvantitativní závislé proměnné
  - `shape` - parametr, za který dosazujeme kategoriální proměnnou, aby měly body scatter-plotu pro každou kategorii jiný tvar
  - `color` - parametr, za který dosazujeme kvantitativní proměnnou, podle jejíchž hodnot se vybarví body scatter-plotu
  - `size` - parametr, za který dosazujeme proměnnou (nejvhodnější je kvantitativní), podle jejíchž hodnot budou body ve scatter-plotu nabývat různých velikostí

`geom_point(...)` - 2. část kódu, kterou říkáme, že vykreslujeme scatter-plot, a do které dosazujeme parametry týkající se doplňujících úprav scatter-plotu, například:

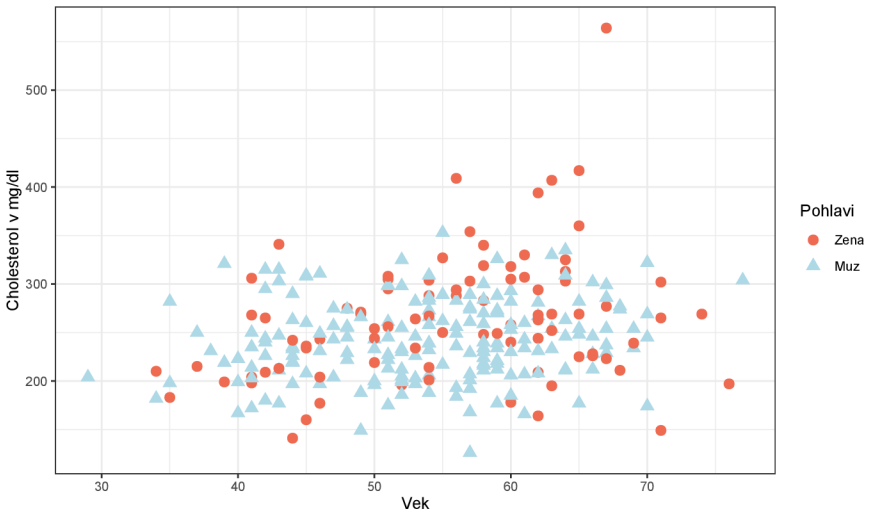
- `size` - parametr, kterým nastavujeme velikost bodů

Pro tvorbu scatter-plot dvou kvantitativních proměnných (zde proměnné věk a množství cholesterolu v mg/dl) při různých kategoriích kvalitativní proměnné (zde pohlaví) použijeme tento postup:

1. Spuštěním následujícího příkazu vytvoříme scatter-plot, který vidíme na obrázku 3.29

```
ggplot(data,aes(x = Age,y = Cholesterol,
  colour = Sex, shape = Sex)) +
geom_point(na.rm=T,size = 3) + xlab("Vek") +
ylab("Cholesterol v mg/dl") + ggtitle("") +
scale_color_manual(name = "Pohlavi",
  labels = c("Zena","Muz"),
  values = c("coral2","lightblue")) +
scale_fill_discrete(name = "Pohlavi",
  labels = c("Zena","Muz")) +
scale_shape_discrete(name = "Pohlavi",
  labels = c("Zena", "Muz")) +
theme_bw()
```

   - `scale_fill_discrete` - část kódu, která upraví legendu tak, aby vnitřní vybarvení bodů v legendě odpovídalo tomu v grafu
   - `scale_shape_discrete` - část kódu, která upraví legendu tak, aby tvar bodů v legendě odpovídal těm v grafu



Obrázek 3.29: Scatter-plot kvantitativních proměnných věk a množství cholesterolu v mg/dl při různých kategoriích kvalitativní veličiny pohlaví v ggplotu

Pokud bychom chtěli vybarvit body scatter-plotu podle kvantitativní proměnné (zde proměnná maximální naměřené hodnoty tepu), budeme postupovat takto:

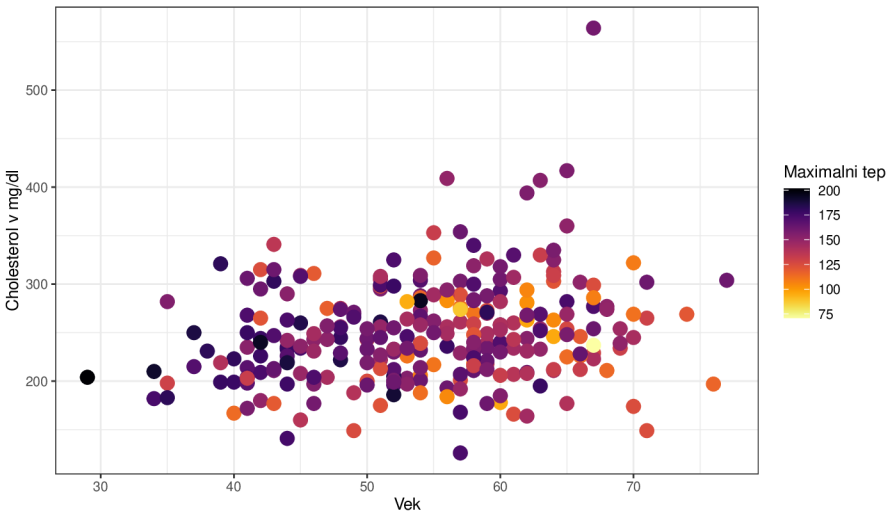
1. Po spuštění následujícího příkazu dostaneme kód na obrázku 3.30, přičemž využijeme knihovnu škál barev `library(viridis)` (není nutností ji použít, tato knihovna má pouze pěkné škály barev, proto ji zde používám)

```
ggplot(data, aes(x = Age, y = Cholesterol,
  colour = Max.HR)) +
geom_point(na.rm = T, size = 4) + ggtitle("") +
xlab("Vek") + ylab("Cholesterol v mg/dl") +
scale_color_viridis(option = "B", direction = -1) +
theme_bw()
```

   - `scale_color_viridis()` - část kódu, kterou upravujeme barevnou škálu

– `option` - zvolíme barevnou škálu, přičemž máme k dispozici možnosti ”A”, ”B”, ”C”, ”D”(výchozí nastavení), nebo ”E”, kdy každá z možností využívá jiné druhy/odstíny barev

– `direction` - parametr, kterým lze změnit ”směr”tónování barev (tzn. můžeme si například zvolit, že nízkým hodnotám bude odpovídat světlá barva a vysokým tmavá, nebo naopak)



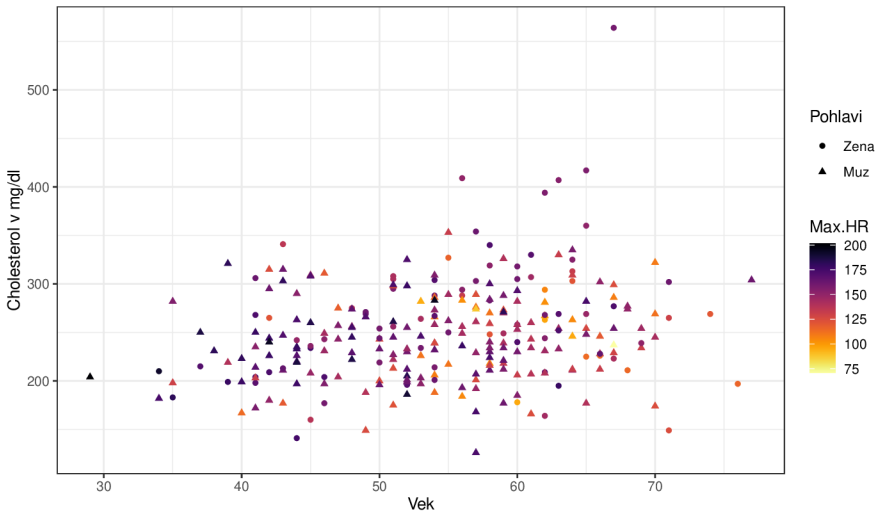
Obrázek 3.30: Scatter-plot kvantitativních proměnných věk a množství cholesterolu v mg/dl s body vybarvenými podle kvantitativní proměnné maximální naměřené hodnoty tepu v ggplotu

Oba předchozí scatter-ploty (3.29, 3.30) můžeme spojit dohromady tímto kódem:

1. Po spuštění dostaneme graf na obrázku 3.31, přičemž byla opět využita knihovna škál barev `viridis`

```
ggplot(data, aes(x = Age, y = Cholesterol,
  color = Max.HR, shape = Sex)) +
geom_point() + ggtitle("") + xlab("Vek") +
ylab("Cholesterol v mg/dl") + theme_bw() +
```

```
scale_color_viridis(option = "B", direction = -1) +
scale_shape_discrete(name = "Pohlavi", labels = c("Zena", "Muz"))
```



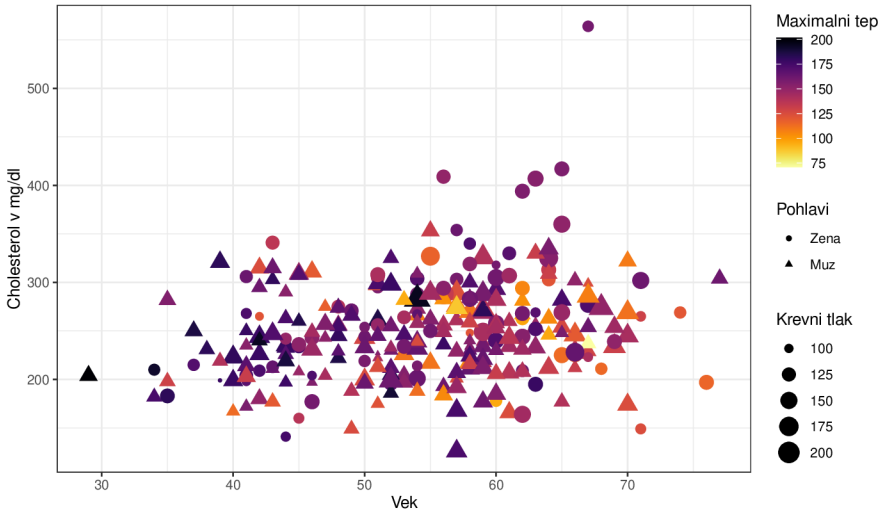
Obrázek 3.31: Scatter-plot kvantitativních proměnných věk a množství cholesterolu v mg/dl s body vybarvenými podle proměnné maximální naměřené hodnoty tepu při různých kategoriích proměnné pohlaví v ggplotu

Poslední možností je připojení další kvantitativní proměnné (zde krevní tlak), kterou velikostně odlišíme body v grafu. Využijeme k tomu následující kód:

1. Opět využijeme knihovnu škál barev `viridis`, přičemž po spuštění následujícího kódu dostaneme scatter-plot na obrázku 3.32

```
ggplot(data, aes(x = Age, y = Cholesterol,
  color = Max.HR, size = BP, shape = Sex)) +
geom_point() + ggtitle("") + xlab("Vek") +
ylab("Cholesterol v mg/dl")+ theme_bw() +
scale_color_viridis(name = "Maximalni tep",
  option = "B", direction = -1) +
scale_size_continuous(name = "Krevni tlak") +
scale_shape_discrete(name = "Pohlavi")
```

- `scale_size_continuous()` - část kódu sloužící k úpravám velikostní legendy



Obrázek 3.32: Scatter-plot kvantitativních proměnných věk a množství cholesterolu v mg/dl s body vybarvenými podle proměnné maximální naměřené hodnoty tepu při různých kategoriích proměnné pohlaví, s body velikostně odlišenými podle krevního tlaku v ggplotu

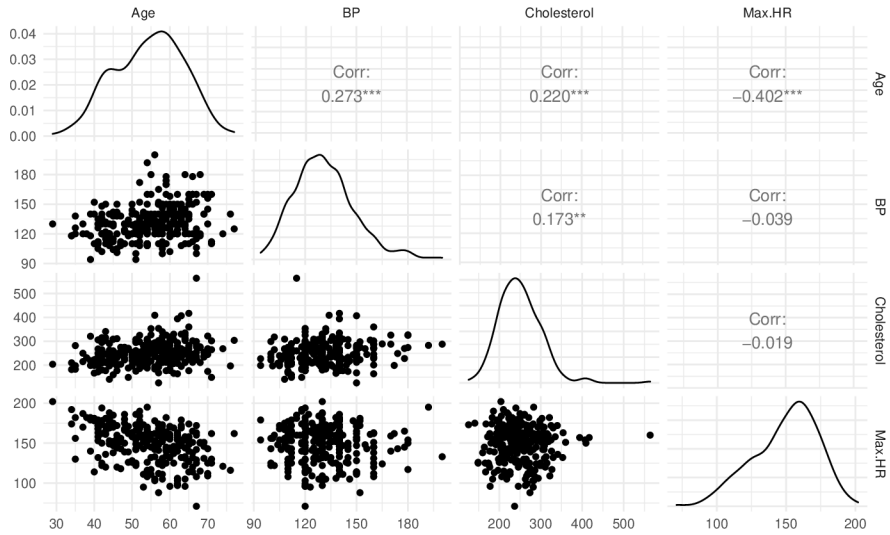
Knihovna `ggplot2` sice nemá přímo zabudovaný příkaz k vykreslení matice scatter-plotů, tak jako běžná knihovna softwaru R, ale pokud použijeme "podpůrnou"knihovnu `GGally`, můžeme matici scatter-plotů vykreslit, a dokonce i s "bonusy".

1. Pro vykreslení matic scatter-plotů s využitím `GGally` spustíme tento příkaz

   ```
   ggpairs(data[,c(1,4,5,8)]) + theme_minimal()
   ```

   - `theme_minimal()` - část kódu, kterou zvolíme typ pozadí grafu

2. Výsledný graf, který kromě scatter-plotů zobrazuje i korelační koeficienty jednotlivých dvojic proměnných a jádrové odhady hustot každé proměnné, vidíme na obrázku 3.33



Obrázek 3.33: Matice scatter-plotů jednotlivých kombinací proměnných věk (Age), naměřené hodnoty krevního tlaku (BP), naměřené hodnoty cholesterolu v mg/dl (Cholesterol) a maximílní naměřené hodnoty tepu (Max.HR) v ggplotu

Vysvětlení hvězdiček u korelačních koeficientů, které vidíme i na obrázku 3.33 je následující:

- * - korelační koeficient je roven hodnotě, kterou vidíme, s p-value $< 0{,}05$
- ** - korelační koeficient je roven hodnotě, kterou vidíme, s p-value $< 0{,}01$
- *** - korelační koeficient je roven hodnotě, kterou vidíme, s p-value $< 0{,}001$

3.5.3. Scatter-plot v plotly

Základní příkaz pro tvorbu scatter-plotů v plotly je dán následovně:

```
plot_ly(data, x =..., y =..., type = "scatter",
        mode = "markers",...), kde
```

`x` - parametr, za který dosazujeme název kvantitativní nezávislé proměnné

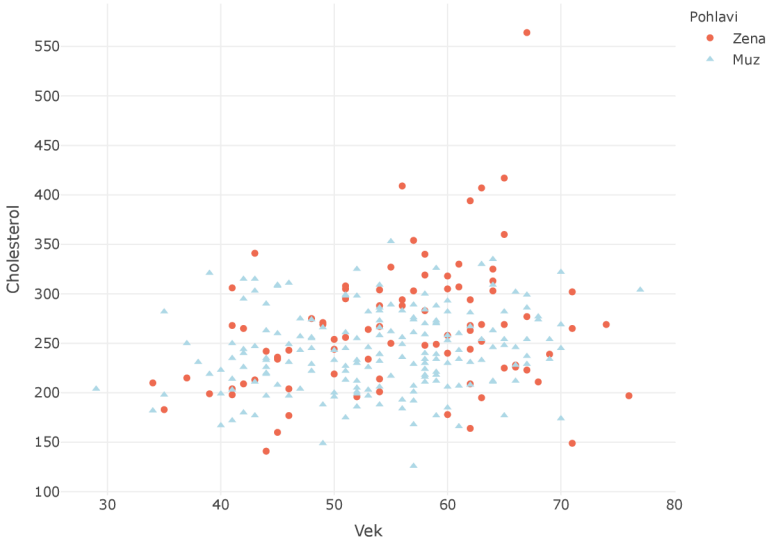
`y` - parametr, za který dosazujeme název kvantitativní závislé proměnné

Pro vykreslení scatter-plotu proměnných věk a cholesterolu při různých kategoriích proměnné pohlaví použijeme následující postup:

1. Vytvoříme příkaz pro scatter-plot

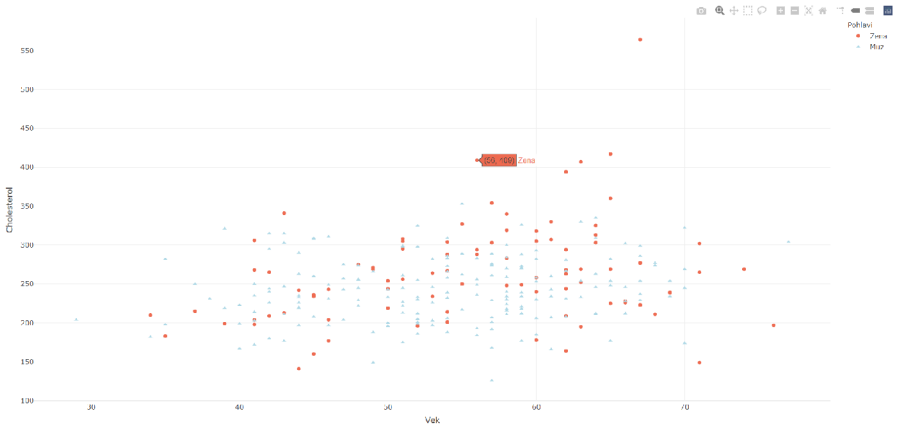
```
plot_ly(data, x = ~Age, y = ~Cholesterol, type = "scatter",
  mode = "markers", symbol = ~Sex,
  symbols = c("circle","triangle-up"),
  color = ~Sex, colors = c("coral2","lightblue")) %>%
layout(title = "", xaxis = list(title = "Vek"),
  yaxis = list(title = "Cholesterol")) %>%
layout(legend = list(title = list(text = "Pohlavi")))
```

2. Po spuštění získáme graf, který vidíme na obrázku 3.34



Obrázek 3.34: Scatter-plot kvantitativních proměnných věk a množství cholesterolu v mg/dl při různých kategoriích kvalitativní veličiny pohlaví v plotly

Interaktivní podoba scatter-plotu z obrázku 3.34 je dostupná na přiloženém CD pod názvem "scat_pl". Printscreen této verze vidíme na následujícím obrázku 3.35.



Obrázek 3.35: Printscreen scatter-plotu kvantitativních proměnných věk a množství cholesterolu v mg/dl při různých kategoriích kvalitativní veličiny pohlaví v plotly

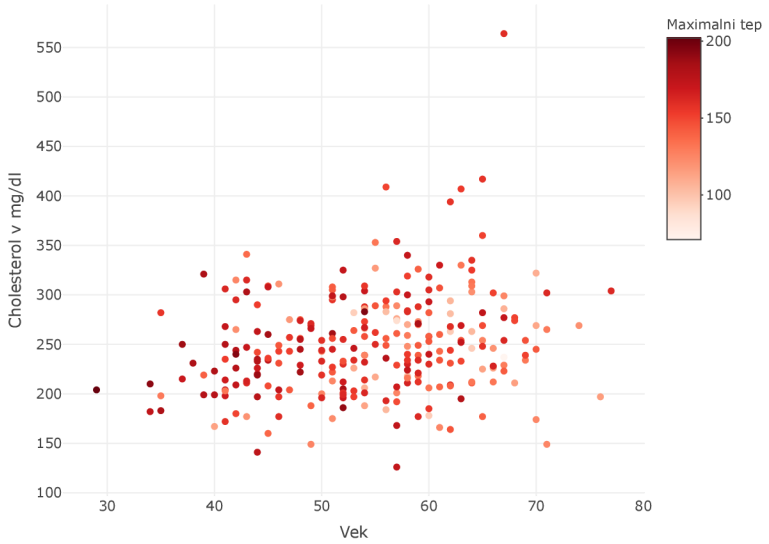
Pro konstrukci scatter-plotu předchozích kvantitativních proměnných s body vybarvenými podle kvantitativní proměnné maximální naměřené hodnoty tepu budeme postupovat následovně:

1. Vytvoříme příkaz pro scatter-plot

```
plot_ly(data, x = ~Age, y = ~Cholesterol, type = "scatter",
 mode = "markers", color = ~Max.HR, colors = "Reds")
layout(title = "", xaxis = list(title = "Vek"),
 yaxis = list(title = "Cholesterol")) %>%
colorbar(title = "Maximalni tep")
```

   - `colors` - parametr, za který volíme barvy, které chceme použít na škálu barev
   - `colorbar(title = ...)` - část kódu, kterou je možné přejmenovat název škály barev

2. Po spuštění získáme graf z obrázku 3.36

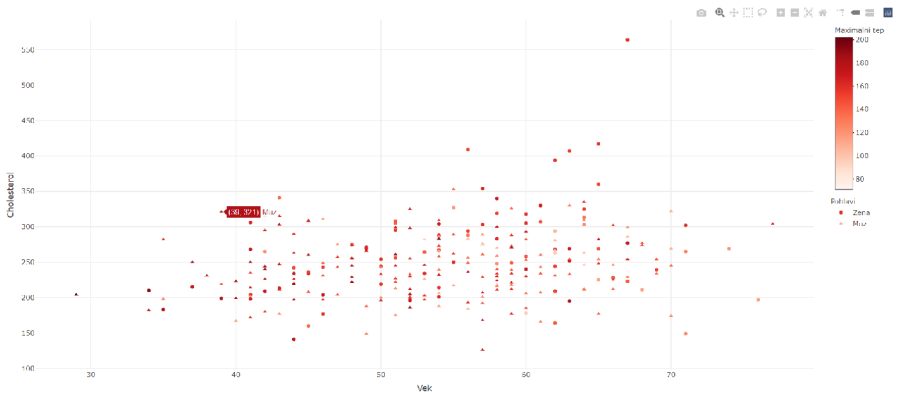


Obrázek 3.36: Scatter-plot kvantitativních proměnných věk a množství cholesterolu v mg/dl s body vybarvenými podle kvantitativní proměnné maximální naměřené hodnoty tepu v plotly

Předchozí scatter-ploty (3.34, 3.36) můžeme opět vykreslit dohromady. Pro tento účel využijeme následující kód:

1. Po spuštění kódu dostaneme graf z printscreenu 3.37, jehož interaktivní podoba je dostupná na přiloženém CD pod názvem "int_komb_2"

```
plot_ly(data, x = ~Age, y = ~Cholesterol,
  color = ~Max.HR, colors = "Reds", symbol = ~Sex,
  symbols = c("circle","triangle-up"),
  type = "scatter", mode = "markers") %>%
layout(title = "", xaxis = list(title = "Vek"),
  yaxis = list(title = "Cholesterol")) %>%
layout(legend = list(title= list(text = "Pohlavi"))) %>%
colorbar(title="Maximalni tep")
```

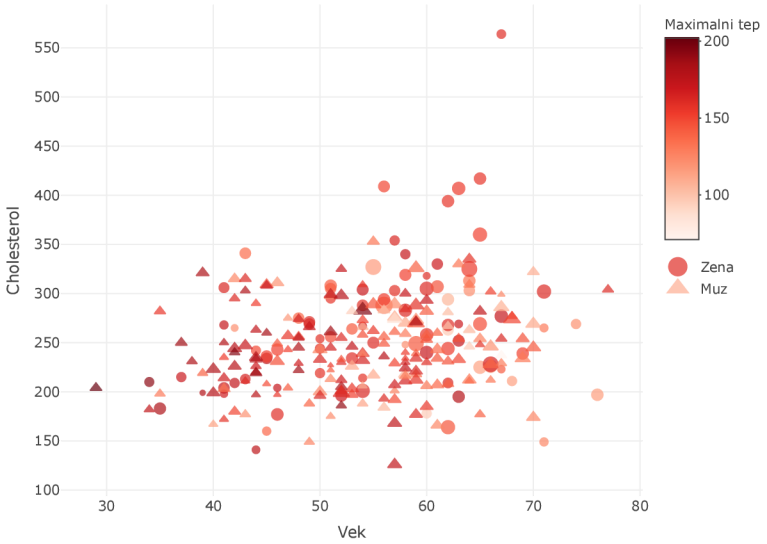


Obrázek 3.37: Printscreen Scatter-plotu kvantitativních proměnných věk a množství cholesterolu v mg/dl s body vybarvenými podle kvantitativní proměnné maximální naměřené hodnoty tepu při různých kategoriích proměnné pohlaví v plotly

Poslední možností je i přidání další kvantitativní proměnné (zde krevní tlak) pro velikostní odlišení bodů. Oproti ggplotu je tu ale ta nevýhoda, že plotly nevytváří pro tuto proměnnou velikostní legendu, a to ani při převodu z ggplotu pomocí funkce `ggplotly()`. Na ukázku ale opět uvádím kód pro tvorbu scatte-plotu se všemi využitými možnostmi (interaktivní podoba je dostupná na CD pod názvem "vse_plotly").

1. Po spuštění následujícího kódu dostaneme graf na obrázku 3.38

```
plot_ly(data, x = ~Age, y = ~Cholesterol, size = ~BP,
  symbol = ~Sex, symbols = c("circle","triangle-up"),
  color = ~Max.HR, colors = "Reds",
  type = "scatter", mode = "markers")) %>%
layout(title = "", xaxis = list(title = "Vek"),
  yaxis = list(title = "Cholesterol")) %>%
colorbar(title = "Maximalni tep")
```



Obrázek 3.38: Scatter-plot kvantitativních proměnných věk a množství cholesterolu v mg/dl s body vybarvenými podle proměnné maximální naměřené hodnoty tepu při různých kategoriích kvalitativní veličiny pohlaví, s body velikostně odlišenými podle krevního tlaku v plotly

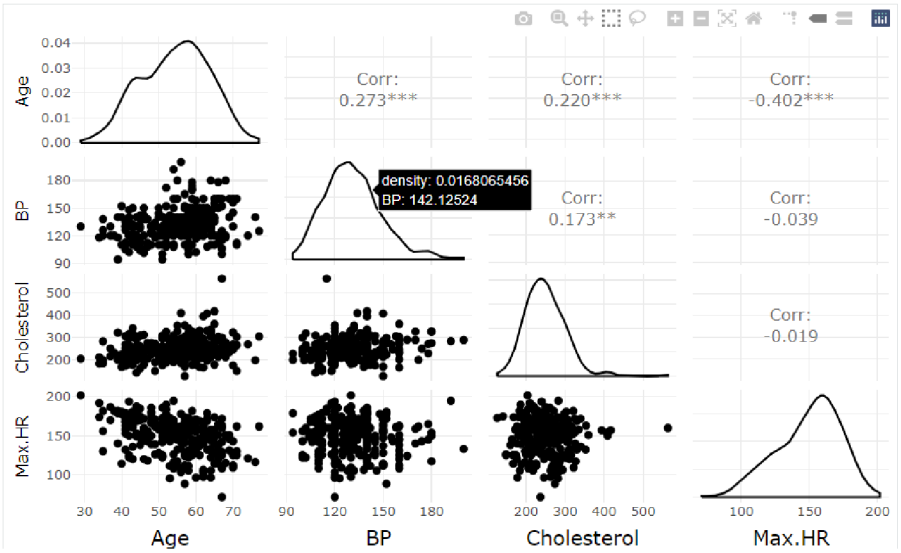
Knihovna plotly neumí tvořit matice scatter-plotů. Dá se to ale opět obejít převedením grafu z ggplotu do plotly pomocí příkazu `ggplotly(...)`.

1. Převedení matice scatter-plotů z ggplotu do plotly provedeme tak, že si uložíme tuto matici a název, pod kterým jsme ji uložili dosadíme do příkazu `ggplotly(...)`, tedy:

   ```
   mat <- ggpairs(data[,c(1,4,5,8)]) + theme_minimal()

   ggplotly(mat)
   ```

2. Výsledkem je interaktivní matice scatter-plotů, jejíž printscreen vidíme na obrázku 3.39 spolu s ukázkou interaktivity při najetí myší na hustotu proměnné naměřeného krevního tlaku (BP)



Obrázek 3.39: Matice scatter-plotů jednotlivých kombinací proměnných věk (Age), naměřené hodnoty krevního tlaku (BP), naměřené hodnoty cholesterolu v mg/dl (Cholesterol) a maximílní naměřené hodnoty tepu (Max.HR) v plotly

Graf z obrázku 3.39 je dostupný na přiloženém CD pod názvem "print_mat".

3.6. Heatmapy

3.6.1. Heatmapa v R

V běžné knihovně softwaru R konstruujeme heatmapu pomocí tohoto příkazu:

`heatmap(x,...)`, kde

`x` - matice číselných hodnot, z kterých chceme vytvořit heatmapu

Konstrukce heatmapy je dána následujícími body:

1. Uložíme si sloupce z našich dat, pro které budeme chtít vykreslit heatmapu

   ```
   kor <- data[,c(1,4,5,8)]
   ```

2. Vytvoříme korelační matici

   ```
   kormat <- round(cor(kor), 2)
   ```

   - `round(...)` - funkce, kterou zaokrouhlujeme čísla na požadovaný počet desetinných míst (zde na 2, přičemž využití to bude mít až u ggplotu, kde můžeme vykreslit do heatmapy hodnoty korelačních koeficientů jednotlivých dvojic)

3. S pomocí knihovny `library(RColorBrewer)` si vytvoříme, a uložíme, barevnou škálu, která bude na heatmapu použita (není to ale nutnost)

   ```
   col <- colorRampPalette(brewer.pal(8, "RdBu"))(270)
   ```

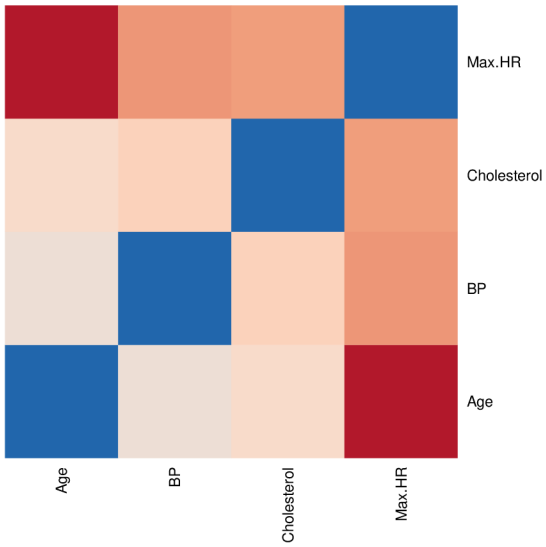
   - `brewer.pal(...)` - funkce, kterou vybíráme, jakou paletu barev použijeme (zde "RdBu", k dispozici je velké množství dalších palet, které je možné nalézt v "Help"softwaru R) a současně vybíráme počet barev z dané palety, které chceme použít (minimem jsou 3, maximum závisí na dané paletě barev), závorka (270) následně značí, že tvoříme škálu barev pro 270 hodnot

4. Vytvoříme heatmapu (nevýhodou je zde však to, že nemáme k vidění barevnou škálu, ani korelační koeficienty)

   ```
   heatmap(kormat, symm = TRUE, cexRow = 1, cexCol = 1, Colv = NA,
   Rowv = NA ,col = col)
   ```

   - `symm` - parametr, kterým říkáma, že chceme symetrickou heatmapu
   - `cexRow` - parametr, kterým měníme velikost popisů v řádcích
   - `cexCol` - parametr, kterým měníme velikost popisů v sloupcích
   - `Colv/Rowv` - parametr, kterým se nastavuje viditelnost sloupcového/řádkového dendogramu (při `NA` nepožadujeme)

5. Výslednou heatmapu vidíme na obrázku 3.40, přičemž modrá barva vyjadřuje korelační koeficient = 1, mimo modrou barvu platí, že čím tmavší je barva, tím nižší je hodnota korelačního koeficientu



Obrázek 3.40: Heatmapa jednotlivých kombinací kvantitativních proměnných věk (Age), naměřené hodnoty krevního tlaku (BP), naměřené hodnoty cholesterolu v mg/dl (Cholesterol) a maximální naměřené hodnoty tepu (Max.HR)

3.6.2. Heatmapa v ggplotu

Pro konstrukci heatmapy v knihovně `ggplot2` se používá tento příkaz:

`ggplot(...) + geom_tile(...)`, kde

`ggplot(...)` - 1. část kódu, která bude obsahovat parametry:

- `data` - parametr, do kterého tentokrát zadáváme tabulku hodnot, ze kterých chceme vytvořit heatmapu (měla by být tvořena třemi sloupci, kde v prvních dvou jsou všechny možné kombinace dvojic proměnných a ve třetím sloupci jsou korelační koeficienty těchto dvojic)
- `aes(...)`
  - `x` - parametr, za který dosazujeme název jednoho sloupce proměnných z tabulky
  - `y` - parametr, za který dosazujeme název druhého sloupce proměnných z tabulky
  - `fill` - parametr, za který dosazujeme název sloupce s hodnotami korelačních koeficientů, podle kterých se heatmapa vybarví

`geom_tile(...)` - 2. část kódu, kterou říkáme, že vykreslujeme heatmapu, a do které lze popřípadě dosazovat parametry týkající se doplňujících úprav heatmapy

Pro tvorbu heatmapy spolu s vypsáním korelačních koeficientů využijeme následující postup:

1. Pomocí knihovny `reshape` vytvoříme z předchozí korelační matice "kormat"tabulku jednotlivých kombinací dvojic a jejich korelačních koeficientů, kterou použijeme pro tvorbu heatmapy

   ```
   tab <- melt(kormat)
   ```

2. Pro přehlednost přejmenujeme sloupce tabulky

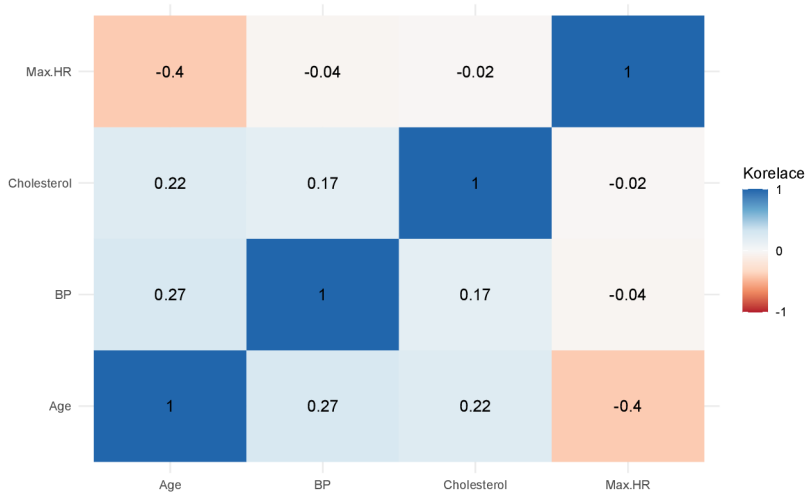
   ```
   colnames(tab) <- c("sloupec1", "sloupec2", "Korelace")
   ```

3. Vytvoříme heatmapu

```
ggplot(tab, aes(x = sloupec1, y = sloupec2,
 fill = Korelace)) +
geom_tile() + xlab("") + ylab("") +
scale_fill_distiller(palette = "RdBu", direction = 1,
 limits = c(-1, 1), breaks = c(-1,0,1)) +
theme_minimal() + geom_text(aes(sloupec1, sloupec2,
 label = Korelace), color = "black", size = 4)
```

   - `scale_fill_distiller` - část kódu, která slouží k nastavení škály barev, přičemž palety barev pochází, jako v předchozím případě, z `ColorBrewer`
     - `limits = c(...)` - parametr, kterým nastavujeme interval hodnot, pro které se barevná škála rozdělí
     - `breaks = c(...)` - parametr, kterým nastavujeme, jakým způsobem chceme rozdělit interval hodnot barevné škály (ty hodnoty které zadáme, se vypíší na škále barev)
   - `geom_text(...)` - část kódu, kterou lze přidat text do grafu (zde požadujeme vypsání korelačních koeficientů parametrem `label`)

4. Po spuštění příkazů, dostaneme heatmapu z obrázku 3.41



Obrázek 3.41: Heatmapa jednotlivých kombinací kvantitativních proměnných věk (Age), naměřené hodnoty krevního tlaku (BP), naměřené hodnoty cholesterolu v mg/dl (Cholesterol) a maximílní naměřené hodnoty tepu (Max.HR) vykreslená v ggplotu

3.6.3. Heatmapa v plotly

Heatmapu v knihovně `plotly` zkonstruujeme pomocí tohoto příkazu:

`plot_ly(x =..., y =..., z =..., type = "heatmap")`, kde

`x` - parametr, za který dosazujeme vektor názvů proměnných, které se zobrazí na ose x

`y` - parametr, za který dosazujeme vektor názvů proměnných, které se zobrazí na ose y

`z` - parametr, za který dosazujeme matici číselných hodnot, z kterých chceme vytvořit heatmapu

Heatmapu v knihovně `plotly` zkonstruujeme pomocí následujích bodů:

1. Při tvorbě heatmapy si opět uložíme sloupce z dat, pro které budeme vykreslovat heatmapu, a názvy těchto sloupců si uložíme do vektoru

```
kor <- data[,c(1,4,5,8)]

jmena <- names(kor)
```

2. Opět vytvoříme stejnou korelační matici, kterou taktéž převedeme do tabulky s využitím knihovny `reshape`

```
kormat <- round(cor(kor), 2)

tab <- melt(kormat)
```

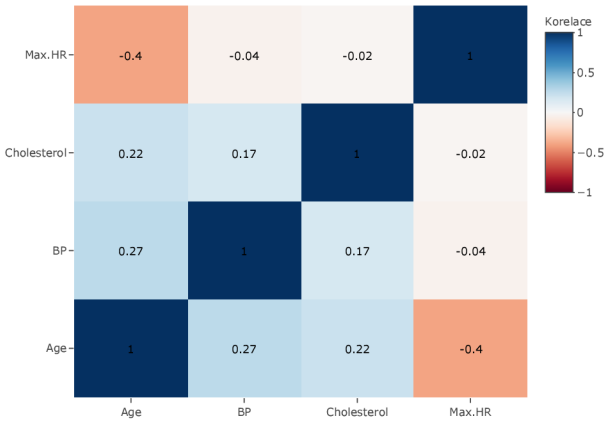
3. Nakonec vytvoříme heatmapu

```
plot_ly(x = ~jmena, y = ~jmena, z = ~kormat,
 type = "heatmap", colors = "RdBu") %>%
colorbar(limits = c(-1,1), title = "Korelace") %>%
layout(xaxis = list(title = ""),
 yaxis = list(title = "")) %>%
add_annotations(x = tab$sloupec1, y = tab$sloupec2,
 text = tab$Korelace, showarrow = FALSE,
 font = list(color = "black"))
```

   - `add_annotations` - část kódu, která umožňuje přidávat text do heatmapy
     - `x` a `y` zde plní roli "objektu", ze kterého se bude při vypisování do heatmapy čerpat
     - `text` - parametr, za který zadáváme, co chceme vypsat do heatmapy (zde je odkázáno na sloupec "Korelace"z naší tabulky - tím dojde k vypsání korelačních koeficientů na příslušná místa, určená parametry `x` a `y`)
     - `showarrow` - parametr, který ve výchozím nastavení (`= TRUE`) vykresluje šipky v heatmapě, které směřují z daného textu do místa, kam text patří

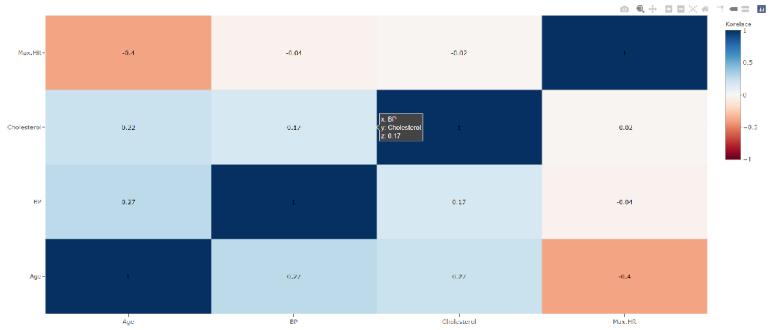
– `font` - parametr sloužící k úpravě textu

4. Výslednou heatmapu vidíme na obrázku 3.42



Obrázek 3.42: Heatmapa jednotlivých kombinací kvantitativních proměnných věk (Age), naměřené hodnoty krevního tlaku (BP), naměřené hodnoty cholesterolu v mg/dl (Cholesterol) a maximální naměřené hodnoty tepu (Max.HR) vykreslená v plotly

Na následujícím obrázku 3.43 vidíme printscreen heatmapy z obrázku 3.42, jehož interaktivní podoba je k dispozici na přiloženém CD pod názvem "int_heat_pl".



Obrázek 3.43: Printscreen heatmapy jednotlivých kombinací kvantitativních proměnných věk (Age), naměřené hodnoty krevního tlaku (BP), naměřené hodnoty cholesterolu v mg/dl (Cholesterol) a maximální naměřené hodnoty tepu (Max.HR) vykreslená v plotly

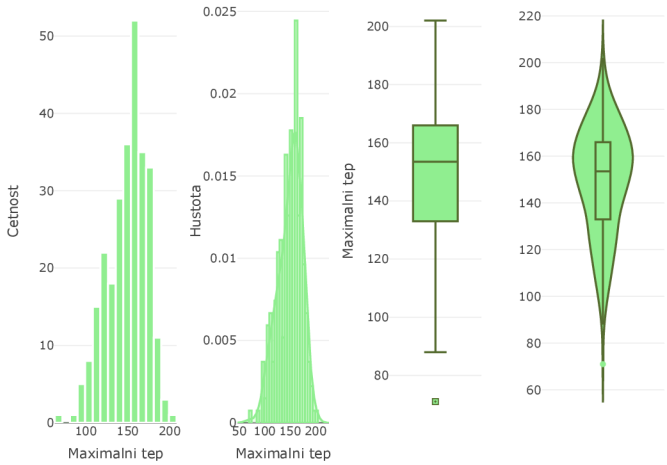
Kapitola 4

Interpretace dosažených výsledků

V rámci této kapitoly se podíváme na to, jak interpretovat vzhled jednotlivých typů grafů (s případným použitím jednoduchých ověřovacích testů), které byly v práci uvedeny.

4.1. Jednotlivé proměnné

4.1.1. Maximální hodnota tepu



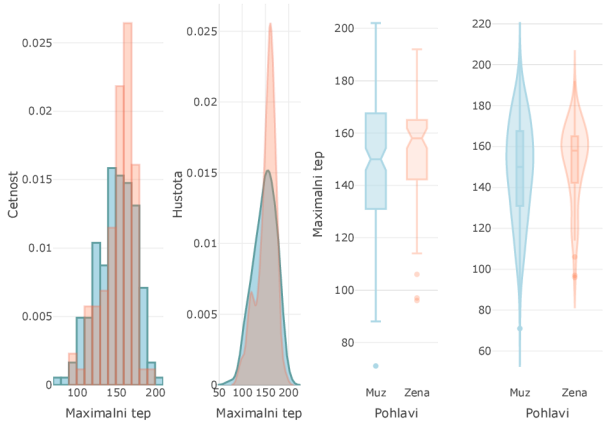
Obrázek 4.1: Grafy kvantitativní proměnné maximální naměřené hodnoty tepu

Nejprve se podívejme na proměnnou "Max.HR" udávající maximální naměřenou hodnotu tepu. Z histogramu 4.1 na první pohled vidíme, že tato charakteristika

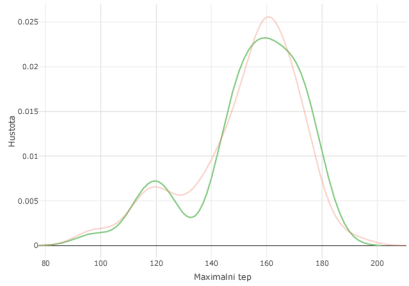
není normálně rozdělena. Histogram, i hustota, jsou sešikmeny vpravo, což vypovídá o tom, že se v datech vyskytuje mnoho lidí, s poměrně vysokou hodnotou tepu. Nejčastěji dosahované hodnoty tepu se pochybují v intervalu <150-165>, přičemž rozpětí dosahovaných hodnot je celkem široké. V rozmezí zhruba <135-145> ale také vidíme mírný pokles četností.

V boxplotu pozorujeme jednu odlehlou hodnotu, která by, vzhledem k nejčastěji dosahovaným hodnotám, mohla být způsobena například chybou v měření. Z dat se jedná o muže s tepem 71 za minutu, kterému je 67 let a má vysoký cholesterol (237 mg/dl). Horní vous boxplotu je o něco kratší než dolní, a medián dosahovaných hodnot, který má hodnotu zhruba 152 mg/dl, je blíže k hodnotě horního kvartilu (cca tep 165 za minutu), než k hodnotě dolního kvartilu (cca 134 za minutu). Poměrně dlouhé délky vousů vypovídají o velké variabilitě dosahovaných hodnot. Z violin-plotu plyne taktéž sešikmení vpravo, vysoká variabilita, ale i to, že nejčastěji dosahovanou hodnotou tepu je hodnota 160 za minutu. Pro detailnější analýzu je vhodné proměnnou prozkoumat z hlediska kategoriálních proměnných (zde z hlediska proměnných pohlaví a množství thallia v těle).

Proměnná maximální naměřené hodnoty tepu dle kategorie pohlaví



Obrázek 4.2: Grafy proměnné maximální naměřené hodnoty tepu rozdělené dle kategorií proměnné pohlaví



Obrázek 4.3: Hustoty proměnné maximální naměřené hodnoty tepu u žen (růžová) a při úrovni thallia Normal u žen (zelená)

Z histogramu na obrázku 4.2 u kategorie žen vidíme, že jim byl nejčastěji naměřen tep v rozmezí zhruba 150-160, přičemž u hustoty této kategorie pozorujeme bimodalitu. Jako vysvětlení lokálního maxima vlevo se nabízí, že by mohlo jít o skupinu žen, které měli po provedení testu thallia v těle normální výsledek - výsledek, který byl lékaři očekáván (důkaz vidíme na obrázku 4.3 - hustoty se v tomto lokálním maximu překrývají).

U mužů jsou naopak dosahované hodnoty tepu vyrovnanější (tzn. nedá se říct, že by existoval nějaký typický interval maximálně naměřených hodnot tepu, kterým bychom je mohli charakterizovat).

Z boxplotů, a jejich intervalových odhadů mediánů, které se nepřekrývají plyne, že mezi kategoriemi lze očekávat statisticky významný rozdíl. Kromě toho z boxplotu mužů vidíme, že dosahované hodnoty tepu jsou poměrně symetricky rozděleny kolem mediánu. Tento boxplot obsahuje i jednu odlehlou hodnotu, která je stejná jako na obrázku 4.1 a jedná se tedy o zmíněného 67 letého muže s vysokým cholesterolem (237 mg/dl). Délka vousů boxplotu vypovídá o velké variabilitě dosahovaných hodnot tepu. Zároveň je dolní vous mírně delší než horní, což odpovídá mírnému sešikmení vpravo, které vypovídá o tom, že se v datech vyskytuje menší množství mužů, kteří dosahovali vyšších hodnot tepu.

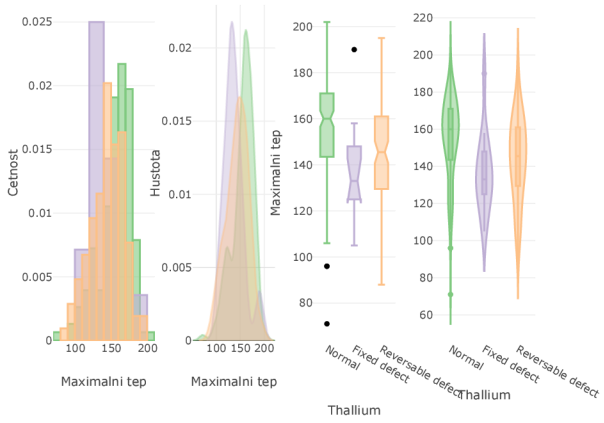
U žen obsahuje boxplot více odlehlých hodnot (tep 96, 97 a 106 za minutu), přičemž jde o ženy v letech 57-74, které mají poměrně vysoký cholesterol (nad 260 mg/dl). Medián hodnot je blíže hodnotě horního kvartilu a vousy vypadají symetricky, což také svědčí pro sešikmení vpravo, které říká, že větší množství

žen dosahovalo vyšších hodnot tepu.

Violin-ploty následně vypovídají o velké variabilitě a vcelku vyrovnaných hodnotách tepu u mužů, a u žen o menší variabilitě a nejčastěji dosahovaných hodnotách tepu kolem 160.

Dále se podíváme na proměnnou maximální naměřené hodnoty tepu dle kategorií proměnné množství thallia v těle 4.4.

Proměnná maximální naměřené hodnoty tepu dle kategorie thallium



Obrázek 4.4: Grafy proměnné maximální naměřené hodnoty tepu rozdělené dle kategorií proměnné množství thallia v těle

Histogramy kategorií "Normal" (zelený) a "Reversable defect" (oranžový), jsou sešikmeny vpravo, což vypovídá o tom, že větší množství pacientů s těmito varianty thallia dosahovalo spíše vyšších hodnot tepu.

Histogram kategorie "Fixed defect" (fialový) naopak naznačuje, že větší množství pacientů s touto variantou dosahovalo nižších hodnot tepu. Varianta "Fixed defect" je zde zastoupena nejméně (vypovídají o tom širší intervaly histogramu), což znamená, že lidí, u kterých jejich srdce nezachytilo po prodělaném infarktu thallium v těle je malé množství (tzn. jen málo z nich má nevratně poškozeno srdce). Zároveň u této varianty pozorujeme dvě lokální maxima. Z proměnných, které byly použity nebyl zjištěn žádný vztah mezi lokálním maximem varianty "Fixed defect" vpravo a těmito proměnnými.

U varianty "Normal" ale vidíme stejné lokální maximum jako u kategorie žen, je tedy mezi touto hodnotou a pohlavím souvislost. Varianta "Reversable defect" je oproti předchozím rovnoměrněji rozložená, co se dosahovaných hodnot maximálního naměřené tepu týče.

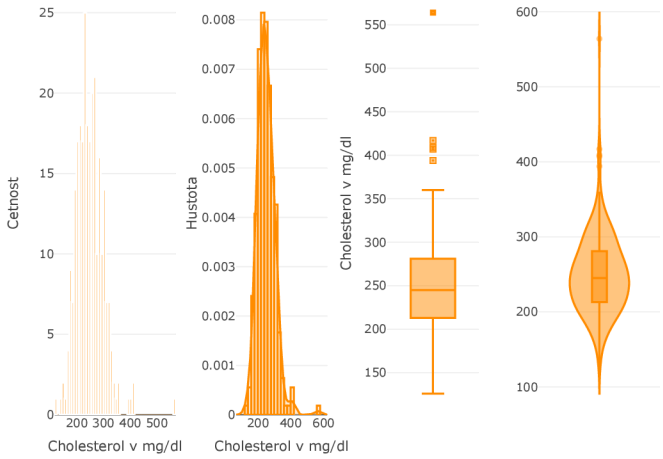
Mezi boxploty variant "Normal" a "Fixed defect" se dá očekávat statisticky významný rozdíl, neboť jejich intervalové odhady mediánů se nepřekrývají, stejně tak je statisticky významný rozdíl mezi boxploty variant "Normal" a "Reversable defect".

Z boxplotu i violin-plotu varianty "Normal" vidíme velkou variabilitu, která je dána především díky odlehlým hodnotám. Jednou z odlehlých hodnot (tep 71 za minutu) je muž ve věku 67 let s vysokým cholesterolem (237 mg/dl) a druhou (tep 96 za minutu) 60 letá žena, s cholesterolem 178 mg/dl. Nízké hodnoty tepu by se tedy mohly pojit se staršími lidmi s vysokým cholesterolem spadající do kategorie thallia "normal".

Boxplot varianty "Fixed defect" má intervalový odhad mediánu větší, než je délka krabice, což může být způsobeno tím, že četnost této varianty je nízká. Odlehlou hodnotou v tomto boxplotu je muž ve věku 52 let, který má, v porovnání s výše zmíněnými pacienty, nižší cholesterol (186 mg/dl).

Boxplot i violin-plot varianty "Reversable defect" následně vypovídají o již zmíněném, poměrně symetrickém, rozložení hodnot kolem mediánu.

4.1.2. Cholesterol v mg/dl

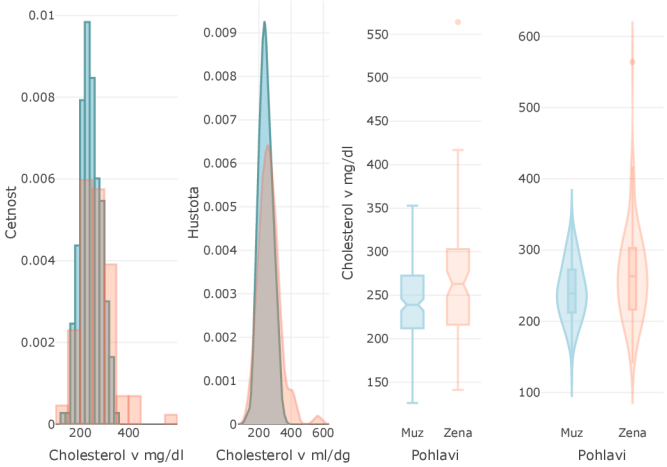


Obrázek 4.5: Grafy proměnné množství cholesterolu v mg/dl

Další proměnnou, kterou se zde budeme zabývat, je proměnná "Cholesterol", která vyjadřuje množství cholesterolu v mg/dl pacientů. Z histogramu můžeme vidět velkou variabilitu v naměřených hodnotách
množství cholesterolu v mg/dl. Histogram je výrazněji sešikmený vlevo, což značí, že se v datech vyskytlo nemalé množství pacientů s nižší hodnotou cholesterolu. Dosahované hodnoty se nejčastěji pohybují kolem 200-250 mg/dl. Vidíme také výraznější lokální maximum u hustoty u hodnot pohybujících se kolem 400 mg/dl.

Boxplot vypovídá taktéž o nižších dosahovaných hodnotách cholesterolu,
přičemž obsahuje více odlehlých hodnot, které by se mohly pojit k lidem s určitou reakcí na množství thallia v těle. Hodnota mediánu (cca 245 mg/dl) je mírně přiblížená k hodnotě dolního kvartilu. Spolu s odlehlými hodnotami vypovídá o sešikmení hodnot k těm "nižším". Z violinplotu vidíme, že velká variabilita je dána především díky výskytu odlehlých hodnot (tou největší odlehlou hodnotou je žena ve věku 67 let s cholesterolem 564 mg/dl, s tlakem pod 120 mm Hg a tepem 160). Mohlo by však jít spíše chybu v měření, vzhledem k tomu, že takové množství cholesterolu je nepřirozeně velké. Pro podrobnější prozkoumání se na tuto veličinu podíváme z hlediska kategorií pohlaví a thallium.

Proměnná množství cholesterolu v mg/dl dle kategorie pohlaví



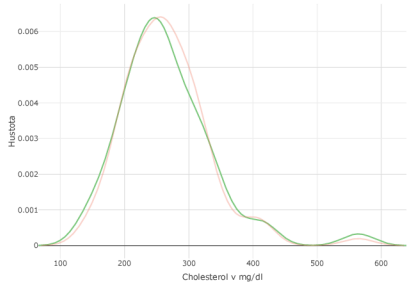
Obrázek 4.6: Grafy proměnné množství cholesterolu v mg/dl dle kategorií proměnné pohlaví

Z histogramu mužů i žen lze vidět, že jsou pro obě pohlaví nejvíce typické dosahované hodnoty cholesterolu v mg/dl v rozmezí zhruba 200-300.

U žen jsou naměřené hodnoty cholesterolu mírně vyrovnané, avšak jak z histogramu, tak z hustoty lze pozorovat dvě lokální maxima v rozmezí hodnot 400-600. Na obrázku 4.7 vidíme, že jako vysvětlení by se nabízelo, že spolu s vyšším cholesterolem se pojí varianta "Normal" u množství Thallia v těle.

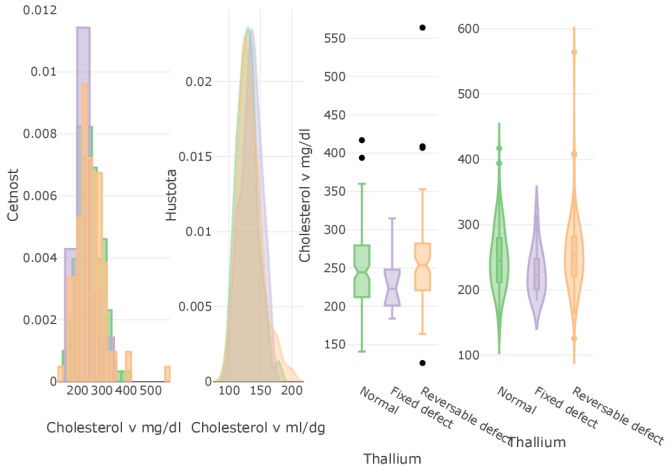
U boxplotů lze vzhledem k nepřekrývajícím se intervalům spolehlivosti očekávat statisticky významný rozdíl mezi pohlavími. Variabilita dosahovaných hodnot je u žen vyšší (muži dosáhli nejvýše hodnoty 352 mg/dl, ženy cca 580 mg/dl), tím pádem je i medián u žen vyšší (cca o 100 mg/dl).

Odlehlou hodnotou v boxplotu žen je již výše zmíněná žena ve věku 67 let s nízkým tlakem (pod 120 mm Hg), která spadá do kategorie "Reversable defect"u proměnné thallium.



Obrázek 4.7: Hustoty proměnné množství cholesterolu v mg/dl u žen (růžová) a při úrovni thallia Normal u žen(zelená)

Proměnná množství cholesterolu v mg/dl dle kategorie thallium



Obrázek 4.8: Grafy proměnné množství cholesterolu v mg/dl dle kategorií proměnné thallium

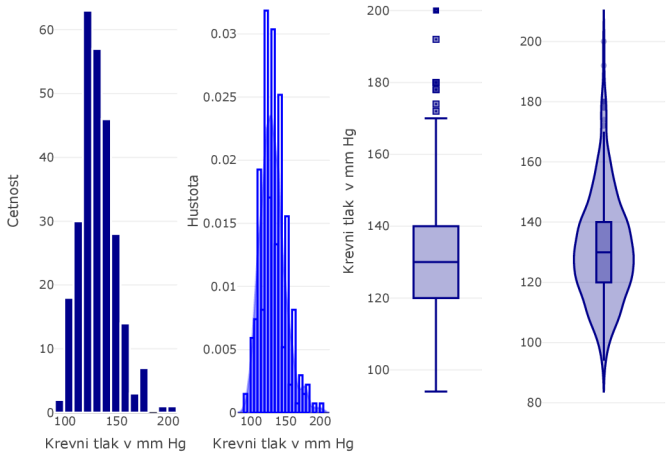
Z histogramů jednotlivých kategorií na první pohled vidíme, že se výrazně překrývají (s výjimkou lokálních maxim oranžové kategorie "Reversable defect") a všechny jsou mírně sešikmené vlevo (histogram kategorie "Fixed defect" (fialový) výrazně, ostatní méně). Sešikmení nám říká, že s výjimkou odlehlých hodnot zde máme velké množství osob, jejichž cholesterol se pohybuje v rozmezí 200-300. Vyskytuje se zde však i malé množství osob s netypicky vysokou hodnotou cholesterolu. Z histogramů i hustot také vidíme příčinu lokálních maxim proměnné množství cholesterolu v mg/dl.

Z boxplotů lze usuzovat, že mezi kategoriemi není statisticky významný rozdíl. Kategorie "Fixed defect" má ale oproti ostatním menší variabilitu, což je nejspíše dáno tím, že do této kategorie spadá výrazně méně lidí než do ostatních dvou. Velké hodnoty cholesterolu se vztahují k lidem, kteří spadají buď do kategorie "Normal" nebo "Reversable defect".

Pokud jde o vysoké odlehlé hodnoty kategorie "Reversable defect", jedná se ve všech případech o ženy ve věku nad 55 let s nižším krevní tlakem (cca 120-151 mm Hg), taktéž nižším tepem (kolem 140). Nízkou odlehlou hodnotou je muž ve věku 57 let s nižším krevním tlakem (kolem 150 mm Hg) a vyšším tepem (nad 170).

Odlehlými hodnotami kategorie "Normal" jsou pouze ženy nad 55 let s různými hodnotami krevního tlaku (120-170 mm Hg) a s mírně vyšším tepem (150-170).

4.1.3. Krevní tlak v mm Hg



Obrázek 4.9: Grafy proměnné krevního tlaku v mm Hg

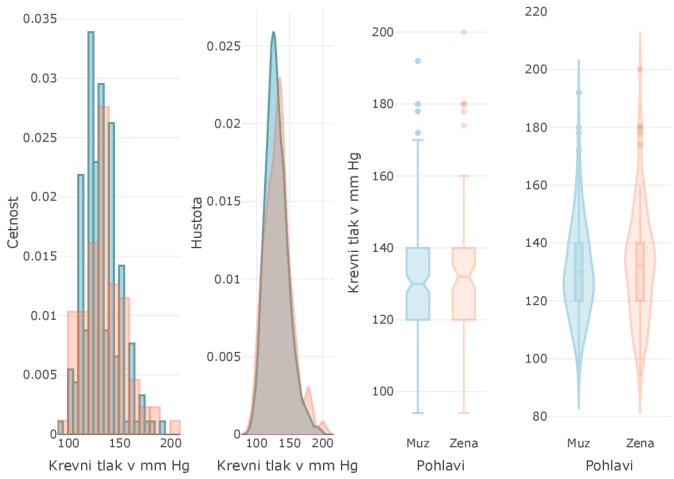
Poslední zkoumanou proměnnou je proměnná "BP", která vyjadřuje krevní tlak v mm Hg. Histogram proměnné vypovídá o velkém množství osob ze souboru, kterým byly naměřeny spíše menší hodnoty krevního tlaku. Vyskytuje se zde však i malé množství pacientů, kterým byla naměřena nestandardně vysoká hodnota tlaku. Z histogramu i z hustoty vidíme menší lokální maximum kolem hodnoty 170 mm Hg. Nejčastěji dosahované hodnoty se pohybují kolem 120-140 mm Hg. Pravděpodobný důvod tohoto lokálního maxima bude popsán při rozdělení této proměnné podle kategorií proměnné pohlaví 4.10.

Boxplot s violin-plotem vypovídají o poměrně větší variabilitě, kterou zapříčiňují odlehlé hodnoty. Mezi odlehlé hodnoty spadají 4 muži ve věku 51-68 let (všichni spadající pod kategorii thallia "Reversable defect") a 5 žen ve věku 54-66 let (2 z kategorie thallia "Reversable defect" a 3 z kategorie "Normal"). Ženám v těchto odlehlých hodnotách byly naměřeny vyšší hodnoty cholesterolu (v rozmezí 220-327 mm Hg) než u mužů (190-282 mm Hg).

Z boxplotu se také zdá, že naměřené hodnoty krevního tlaku jsou rovnoměrně rozloženy kolem mediánu (130 mm Hg), avšak dolní vous, který je kratší než horní spolu s odlehlými hodnotami vypovídají právě o sešikmení směrem k malým

hodnotám.

Proměnná krevní tlak v mm Hg dle kategorie pohlaví



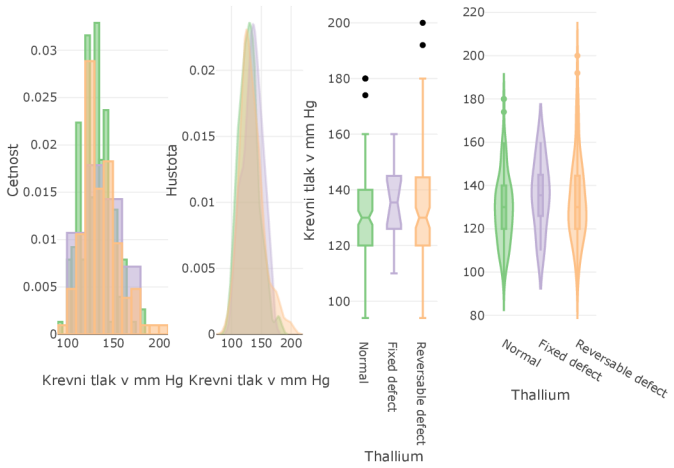
Obrázek 4.10: Grafy proměnné krevního tlaku v mm Hg dle kategorií proměnné pohlaví

Histogramy mužů a žen jsou vcelku porovnatelné. Muži dosahují nejčastěji hodnot krevního tlaku jen o něco menších než ženy (muži kolem 120-125 mm Hg a žen 125-135).

U žen opět pozorujeme větší lokální maximum u hodnot v rozmezí 150-190, který se, na základě dat, pravděpodobně pojí s vyšším věkem pacientů.

Z boxplotů vidíme, že mezi kategoriemi u této proměnné nelze očekávat statisticky významný rozdíl (překrývají se jejich intervalové odhady mediánu). Oba boxploty obsahují spoustu odlehlých hodnot, které byly popsány už výše 4.1.3. Violin-ploty spolu s boxploty naznačují sešikmení směrem k menším hodnotám (s výjimkou malého počtu osob s velmi vysokými hodnotami tlaku máme v datech velké množství pacientů s malou hodnotou tlaku) a vysokou variabilitu naměřených hodnot krevního tlaku jak u mužů, tak u žen.

Proměnná krevní tlak v mm Hg dle kategorie thallium



Obrázek 4.11: Grafy proměnné krevního tlaku v mm Hg dle kategorií proměnné pohlaví

Z histogramů vidíme nepoměr mezi nejčastěji dosahovanými hodnotami kategorie "Fixed defect" (fialová) a ostatních kategorií, za což opět může nepoměr mezi počty pacientů.

Histogramy kategorií "Normal" a "Reversable defect" jsou sešikmeny směrem k malým hodnotám, a proto jsou osoby, které spadají do těchto kategorií, charakterizovány spíše nižšími hodnotami tlaku (opět s výjimkou již zmíněných odlehlých hodnot). Kategorii "Fixed defect" bychom mohli charakterizovat normálním rozdělením (což je pozorovatelné i na boxplotu s violin-plotem).

Nejčastěji kategorie "Normal" (zelená) a "Reversable defect" (oranžová) dosahovaly hodnot v rozmezí 120-150 mm Hg, u kategorie "Fixed defect" (fialová) jsou dosahované hodnoty v rozmezí 110-160. U kategorie "Reversable defect" se ale objevují i vyšší dosahované hodnoty (do 200 mm Hg), stejně tak u kategorie "Normal".

Intervalové odhady mediánů boxplotů vypovídají o absenci statisticky významného rozdílu mezi kategoriemi (u kategorie "Fixed defect" je to dáno nejspíše díky velké rezervě v podobě širokého intervalového odhadu mediánu). První dva boxploty mají hodnoty rovnoměrně rozděleny kolem mediánů, v případě kategorie

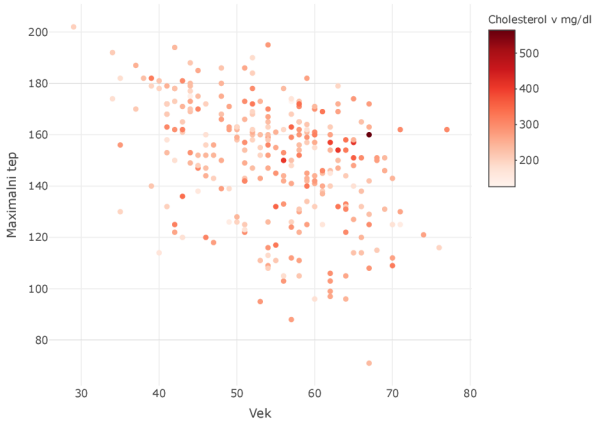
"Reversable defect" se medián blíží hodnotě dolního kvartilu. U kategorie "Normal" a "Reversable defect" vypadají hodnoty jejich hodnoty mediánu podobně.

Odlehlé hodnoty kategorie "Normal" patří dvěma ženám v letech 55 a 64. Obě ženy lze dále charakterizovat vysokým cholesterolem (nad 320 mg/dl). Jednou z odlehlých hodnot kategorie "Reversable defect"je 54 letý muž s vyšším cholesterolem (283 mg/dl) a druhou odlehlou hodnotou je žena ve věku 55 let s podobně vysokým cholesterolem (288 mg/dl).

Violin-ploty i boxploty vypovídají o tom, že největší variabilitu má kategorie "Reversable defect", a hned po ní kategorie "Normal".

4.2. Vztahy mezi proměnnými

4.2.1. Vztah proměnných věk, množství cholesterolu v mg/dl a maximálně naměřené hodnoty tepu

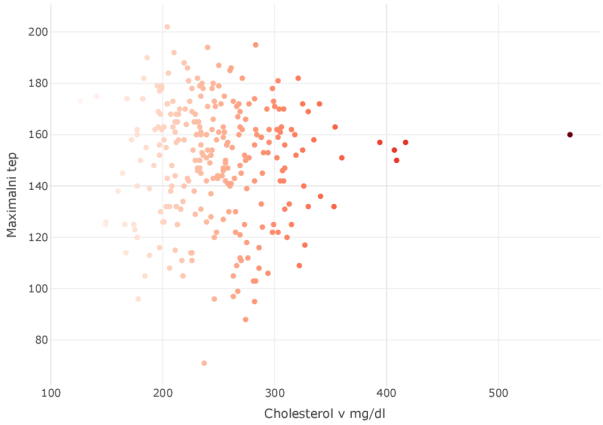


Obrázek 4.12: Scatter-plot proměnných věk a maximální naměřené hodnoty tepu s body barevně odlišenými dle kvantitativní proměnné množství cholesterolu v mg/dl

Body scatter-plotu, jejichž x-ové souřadnice jsou tvořeny hodnotami proměnné věk a y-ové souřadnice hodnotami proměnné maximální naměřené hodnoty tepu, jsou mírně koncentrovány kolem pomyslné přímky vyjadřující nepřímou lineární závislost. Uspořádání bodů tedy (s poměrně velkou rezervou) vypovídá o nepřímém lineárním vztahu těchto veličin. Vyšší tep bychom tedy očekávali spíše u mladších jedinců. Následně se podíváme na vztah mezi proměnnou maximální naměřené hodnoty tepu a proměnnou množství cholesterolu v mg/dl, jejíž hodnoty jsou pro dané uspořádané dvojice bodů $(x_1, y_1), \ldots, (x_{270}, y_{270})$ vybarveny příslušnou barvou z barevné škály (tzn. dané kombinaci věk-tep odpovídá určitá hodnota cholesterolu). Vypadá to, že vysoké hodnoty tepu se objevují spíše u vyšších hodnot cholesterolu, a naopak, s nižšími hodnotami tepu se pojí spíše nižší hodnoty cholesterolu (lépe můžeme jejich vztah vidět na obrázku 4.13). Z pohledu dvojice věk-cholesterol, vzhledem k velkému množství světlých bodů vyjadřujících nízkou hodnotu cholesterolu v různých letech pacientů, nelze říct, že bychom mezi nimi

mohli očekávat vztah. (Oba předpoklady je ale takto "od oka" nutno brát s rezervou, vzhledem k tomu, že body jsou po celé ploše hodně barevně rozmanité na jednoznačné potvrzení vztahu).

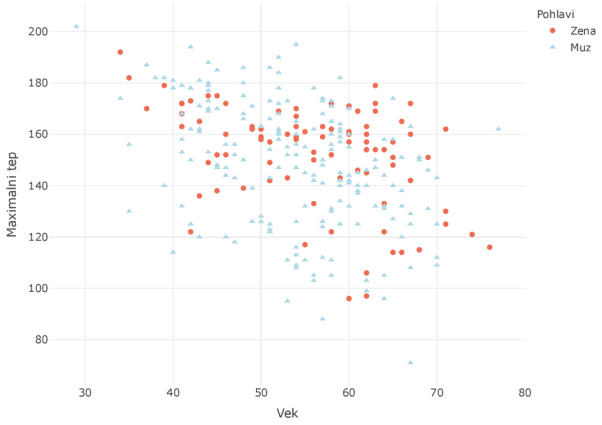
Detailní porovnání proměnných cholesterol v mg/dl a maximální naměřené hodnoty tepu



Obrázek 4.13: Scatter-plot proměnných množství cholesterolu v mg/dl a kvantitativní proměnné maximální naměřené hodnoty tepu

Na obrázku 4.13 (kde jsou body vybarveny podle proměnné množství cholesterolu v mg/dl - jen kvůli estetice) vidíme přímo vztah proměnných množství cholesterolu v mg/dl a maximální naměřené hodnoty tepu, které jsou zde vykreslené pro lepší rozpoznání jejich vztahu, o kterém bylo psáno výše u obrázku 3.36. Z tohoto zobrazení proměnných už se zdá spíše to, že mezi nimi není žádný vztah.

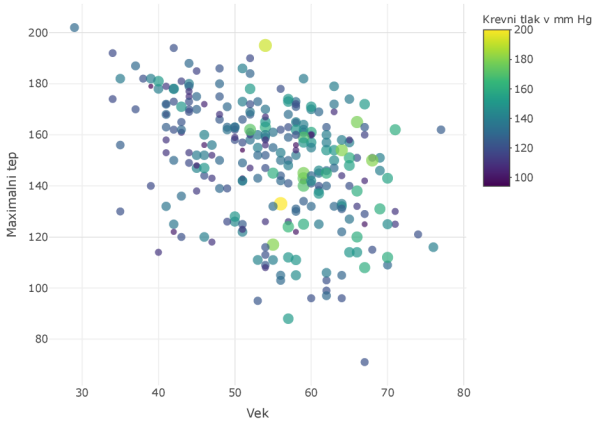
4.2.2. Vztah proměnných věk, maximální naměřené hodnoty tepu a pohlaví



Obrázek 4.14: Scatter-plot proměnných věk a maximální naměřené hodnoty tepu s body tvarově odlišenými dle kategoriální proměnné pohlaví

Pro zjišťování vztahu mezi proměnnými bývá užitečné scatter-plot barevně odlišit podle jednotlivých kategorií (zde podle pohlaví). Z obrázku 4.14 vidíme, že se výzkumu zúčastnilo více mužů než žen. U mužské kategorie jsou dosahované hodnoty tepu o něco vyšší než u žen (viz. obrázek 4.2). Zároveň se zdá, že se zvyšujícím se věkem u mužů tep pomalu klesá. U žen lze pozorovat strmější pokles (než u mužů) hodnot tepu se zvyšujícím se věkem, s výjimkou období zhruba mezi 48-60 lety. Mezi věkem mužů i žen a jejich tepem by tedy v určité míře mohl existovat vztah.

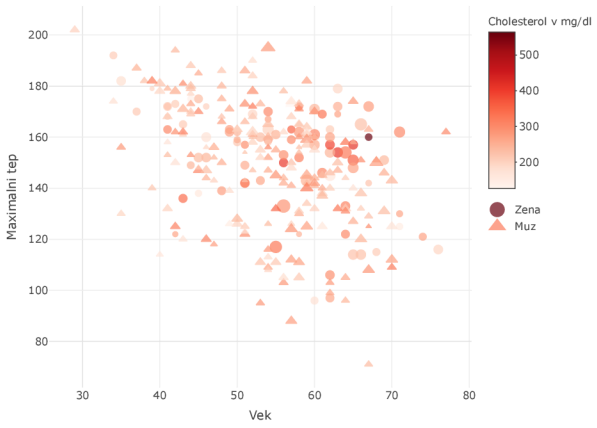
4.2.3. Vztah proměnných věk, maximální naměřené hodnoty tepu a krevní tlak v mm Hg



Obrázek 4.15: Scatter-plot proměnných věk a množství cholesterolu v mg/dl s body velikostně odlišenými dle kvantitativní proměnné krevní tlak v mm Hg

Další možností, kterou je možno využít při konstrukci scatter-plotu je velikostní odlišení bodů podle kvantitativní proměnné (zde podle hodnot krevního tlaku v mm Hg). Oproti ggplotu tu plotly tvoří legendu při velikostním odlišením jinak - nevytvoří pár velikostně odlišených bodů s hodnotami pro orientaci, ale vytvoří barevnou škálu, což je někdy nevýhodné (vysvětleno pod obrázkem 3.38). Co se týče vztahu hodnot krevního tlaku s ostatními proměnnými, lze říct, že velmi vysokých hodnot krevního tlaku je dosahováno pouze ve zhruba 54 - 68 letech. Těmto kombinacím let a hodnot tlaku odpovídají velmi různé hodnoty tepu, proto mezi nimi nelze očekávat žádný vztah.

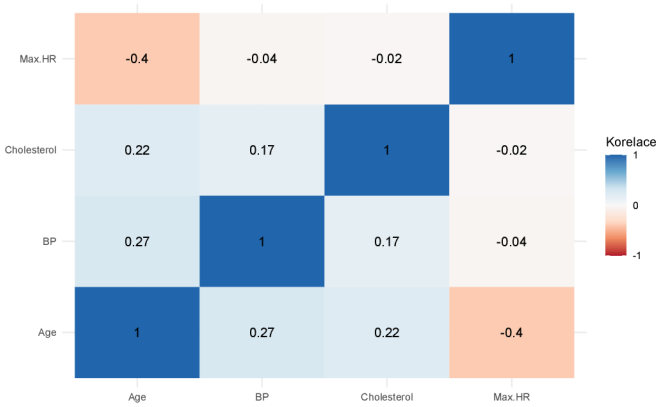
4.2.4. Znázornění vztahu mezi proměnnými věk, pohlaví, množství cholesterolu v mg/dl, krevní tlak v mm Hg a maximální naměřené hodnoty tepu



Obrázek 4.16: Scatter-plot proměnných věk a maximální naměřená hodnota tepu s body velikostně odlišenými dle kvantitativní proměnné krevní tlak v mm Hg, tvarově dle pohlaví a barevně dle množství cholesterolu v mg/dl

Na obrázku 4.16 vidíme maximální množství proměnných, které můžeme najednou ve scatter-plotu porovnávat. Menší nevýhoda u knihovny plotly je právě výše zmiňovaná tvorba velikostního odlišení bodů v legendě. Při tvorbě musí být zadefinována barva pro odlišení, abychom k této variantě získali legendu. Tím, že barvu využijeme na legendu jiné proměnné, už tato možnost odpadá. Proto zde máme velikostní odlišení podle proměnné krevní tlak (malá velikost = malé hodnoty a naopak), ale už nevidíme tuto proměnnou v legendě. Ze scatter-plotu však můžeme například zjistit, že muži dosahují vyšších hodnot tepu spíše v nižším věku, a nižších hodnot tepu ve věku vyšším. Vyššího krevního tlaku muži dosahují spíše v kombinaci buď s nízkým cholesterolem, nebo s nízkým věkem. Ženy ve vyšším věku dosahují taktéž spíše nižších hodnot tepu, a velmi vysokých hodnot tepu ve věku nižším. Vyššího tlaku dosahují ženy v kombinaci s vyšším cholesterolem, nejvíce v letech cca 56-72.

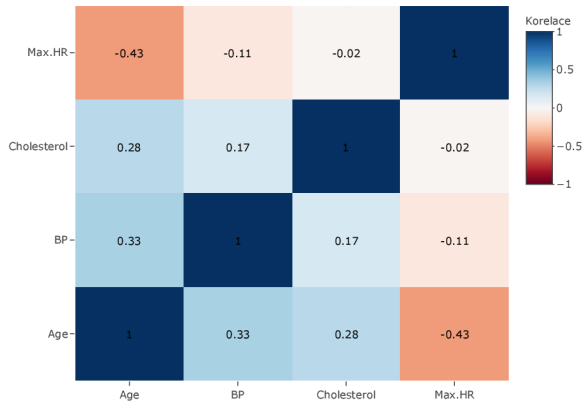
4.2.5. Heatmapa kvantitativních proměnných věk, množství cholesterolu v mg/dl, krevní tlak v mm Hg a maximální naměřené hodnoty tepu



Obrázek 4.17: Heatmapa jednotlivých kombinací kvantitativních proměnných věk (Age), naměřené hodnoty krevního tlaku (BP), naměřené hodnoty cholesterolu v mg/dl (Cholesterol) a maximální naměřené hodnoty tepu (Max.HR)

V heatmapě vidíme vykresleny korelační koeficienty vyjadřující korelaci mezi jednotlivými dvojicemi proměnných. Tyto korelační koeficienty odpovídají barevné škále, kde s rostoucí hodnotou korelačního koeficientu se čtvereček v heatmapě dané dvojice proměnných bude zbarvovat do tmavě modra, a v opačném případě do ruda. Vidíme, že veličiny jsou mezi sebou velmi málo korelovány. Největší hodnotu korelačního koeficientu můžeme pozorovat jen u kombinace proměnných maximální naměřené hodnoty tepu a věku. Tento korelační koeficient je záporný, mělo by tedy s vyšším věkem docházet k poklesu tepu, což vzhledem k ne příliš velké korelaci nelze brát jako pravidlo.

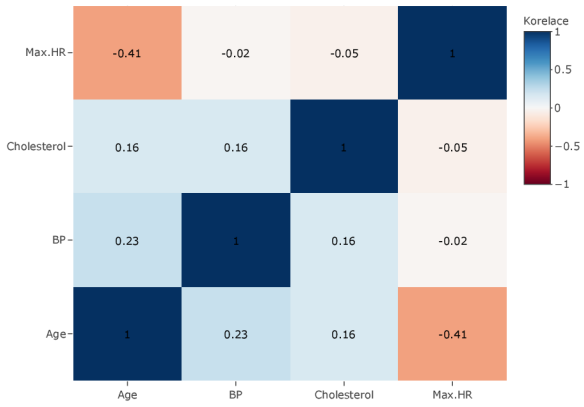
Heatmapa pro kategorii žen



Obrázek 4.18: Heatmapa jednotlivých kombinací kvantitativních proměnných věk (Age), naměřené hodnoty krevního tlaku (BP), naměřené hodnoty cholesterolu v mg/dl (Cholesterol) a maximální naměřené hodnoty tepu (Max.HR) u kategorie žen

Z heatmapy žen na obrázku 4.18 vidíme, že se mírně zvýšila korelace mezi kombinacemi tlak-věk, oproti heatmapě z obrázku 4.17, která byla tvořena pro muže i ženy dohromady. Korelace u kombinace tep-tlak se oproti předchozí mírně přiblížila k hodnotě -1 (tzn. se zvyšující se hodnotou tepu klesají hodnoty tlaku a naopak). Z této heatmapy lze tedy, s rezervou, tvrdit, že kromě nižších hodnot tepu ve vyšším věku bychom ve vyšším věku mohli pozorovat i vyšší hodnoty tlaku, a při vyšších hodnotách tlaku bychom mohli pozorovat nižší hodnoty tepu a naopak.

Heatmapa pro kategorii mužů



Obrázek 4.19: Heatmapa jednotlivých kombinací kvantitativních proměnných věk (Age), naměřené hodnoty krevního tlaku (BP), naměřené hodnoty cholesterolu v mg/dl (Cholesterol) a maximální naměřené hodnoty tepu (Max.HR) u kategorie mužů

V heatmapě mužské kategorie na obrázku 4.19 vidíme, oproti kategorii žen, jedinou významnější korelaci. Tou je korelace mezi proměnnými maximální naměřené hodnoty tepu a věkem. Heatmapa obou kategorií na obrázku 4.17 tedy vykazuje velmi nízké korelace pro kombinace proměnných tep-tlak a tep-věk především díky mužské kategorii.

Závěr

Výběr knihovny pro tvorbu grafu se odvíjí od toho, k čemu má být graf použit. Pro rychlé zjištění informací o proměnných a jejich vztahu mezi nimi postačí běžná knihovna softwaru R. V případě, že požadujeme sofistikovanější výstup, u kterého máme zvýšené nároky na vzhled grafu, z důvodu například následné prezentace výsledku, je vhodnější použít knihovnu ggplot.

Z mého pohledu je však nejzajímavější knihovna plotly, především díky interaktivní povaze grafu, kterou je knihovna schopna vytvořit. Velkou výhodou u této knihovny je také možnost již vytvořený graf v ggplotu převést do interaktivní podoby. Proto pokud člověk dobře ovládá knihovnu ggplot, má do jisté míry vyhráno i s touto knihovnou.

V práci nebyly samozřejmě využity veškeré výhody knihovny plotly. Pro náročnější uživatele softwaru existuje i možnost vykreslení interaktivního 3D grafu, nebo lze využít možnosti přidat do grafu tlačítka, kterými lze překlikávat například mezi jednotlivými kategoriemi proměnné.

Osobně pro mě práce na toto téma představovala příležitost zdokonalit si své znalosti v oblasti tvorby a prezentace grafů v běžné knihovně R a ggplotu, a doplnit tyto znalosti o tvorbu grafů v interaktivní podobě tvořených prostřednictvím knihovny plotly.

Literatura

[1] *Binomická věta*, [online], [cit. 2022-03.05], dostupné z:
https://cs.wikipedia.org/wiki /Binomická_věta

[2] DataNovia, *GGplot Boxplot*, [online], [cit. 2022-02.25] dostupné z:
https://www.datanovia.com/en /lessons/ggplot-boxplot/

[3] ggplot2 3.3.5: *Complete themes*, [online], [cit. 2022-03.30], dostupné z:
https://ggplot2.tidyverse.org/reference/ggtheme.html

[4] ggplot2: *Overview*, [online], [cit. 2022-04.04], dostupné z:
https://www.rdocumentation.org/packages/ggplot2/versions/3.3.5

[5] Holčík, J., Komenda, M. (eds.) a kol.: *Matematická biologie: e-learningová učebnice 1. vydání*, [online], dostupné z:
https://portal.matematickabiologie.cz/index.php?pg=analyza-a-hodnoceni-biologickych-dat--statisticke-modelovani--pruzkumova-analyza-jednorozmernych-dat--diagnosticke-grafy--krabicovy-diagram-box-plot.
Masarykova univerzita, Brno, 2015

[6] Hoyt, R.: *Heart_Disease_Prediction*, [online], [cit.2022-04.04], dostupné z:
https://data.world/informatics-edu/heart-disease-prediction/workspace/file?filename=+Heart_Disease_Prediction.csv

[7] Hron, K., Kunderová, P., Vencálek, O.: *Základy počtu pravděpodobnosti a metod matematické statistiky (3. přepracované vydání)*. Univerzita Palackého v Olomouci, Olomouc, 2018. Skripta.

[8] Hyndman, J., R.: *The problem with Sturges' rule for constructing histograms*, [online], [cit. 2022-01.15], dostupné z:
https://www.researchgate.net/publication/222105804_The_problem_with_Sturges'_rule_for_constructing_histograms

[9] McGill, Robert, John W. Tukey, and Wayne A. Larsen. *"Variations of Box Plots."* The American Statistician 32, no. 1 (1978): 12–16. [online], [cit. 2022-02.26], dostupné z:

https://doi.org/10.2307/2683468

[10] Notes for Nonparametric Statistics, *Bandwidth selection*, [online], [cit. 2022-02.19] dostupné z:

https://bookdown.org/egarpor/NP-UC3M/kde-i-bwd.html

[11] Pipis, G.:*Skewed distribution*, [online], dostupné z:

https://stackoverflow.com/questions/28099590/create-sample-vector-data-in-r-with-a-skewed-distribution-with-limited-range

[12] Plotly, *Plotly R Open Source Graphing Library*, [online], [cit. 2022-04.04], dostupné z:

https://plotly.com/r/

[13] Plotly, *R Figure Reference: histogram Traces*, [online], [cit. 2022-03.06], dostupné z:

https://plotly.com/r/reference /histogram/

[14] R CODER: *Histogram in ggplot2 with Sturges method*, [online], dostupné z: https://r-charts.com/distribution/histogram-sturges-ggplot2/

[15] Scott, W., D.: *Multivariate density estimation: Theory,Practice, and Visualization (second edition)*. Willey, New York, 1992

[16] The R Graphics Package: *Documentation for package 'graphics' version 4.3.0*, [online], [cit. 2022-04.04], dostupné z:

https://stat.ethz.ch/R-manual/R-devel/library/graphics/html/00Index.html

[17] Turlach, A., B.,: *Bandwidth Selection in Kernel Density Estimation: A Review*, [online], [cit. 2022-01.15], dostupné z:

https://www.researchgate.net/publication/2316108_Bandwidth_Selection_in_Kernel _Density_Estimation_A_Review

[18] Wickham, H., Grolemund, G.: *R for Data Science: import, tidy, transform, visualize, and model data*. O'Reilly, Beijing, 2017

Přílohy

Seznam příloh:

- CD přiložené k bakalářské práci obsahující veškeré kódy grafů