Pokročilé metódy štatistického modelovania

Tomáš Bacigál

2024-10-09

Obsah

Predslov

Táto kniha slúži ako učebný materiál predmetu Pokročilé metódy štatistického modelovania (I1-PMSM) na 1. stupni inžinierskeho štúdia programu Matematicko-počítačové modelovanie (MPM) na Stavebnej fakulte (SvF) Slovenskej technickej univerzity v Bratislave (STU).

Nadväzuje na nasledujúce predmety bakalárskeho stupňa: Softvér na analýzu údajov (B1-SAD), Štatistické metódy (B1-STM), Teória pravdepodobnosti (B1-TP) a Analýza časových radov (B1-ACR). Pokračovaním PMSM je povinne voliteľný predmet Hĺbková analýza údajov a strojové učenie (I1-HAU).

Štruktúra a obsah prvej polovice knihy (prvá až piata kapitola) je do značnej miery inšpirovaná publikáciou James et al. (2021) a doplnená o poznatky čerpané prevažne v Hastie, Tibshirani, a Friedman (2009) či Shalizi (2021). Ostatné kapitoly sú spracované použitím heterogénnej kolekcie zdrojov vrátane vedeckých článkov, elektronických kníh, blogov a ďalších odborných internetových stránok.

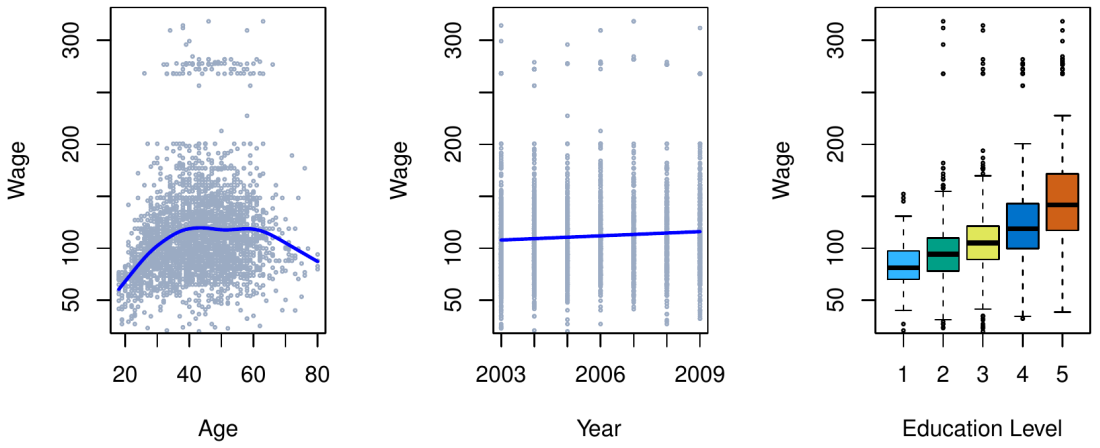
1 Úvod

1.1 Štatistické modelovanie

- Štatistické modelovanie je rozsiahla sada metód a nástrojov určená na porozumenie dátam.
- S rozvojom strojového učenia sa zaužíval aj názov štatistické učenie.
- Sú dva základné spôsoby učenia: *asistované a neasistované* .
- Asistované predstavuje výstavbu štatistického modelu na *odhad výstupu* pomocou jedného či viacerých *vstupov*.
- V neasistovanom učení sú vstupy, no žiaden asistujúci výstup. Stále však možno skúmať vzťahy v dátach a ich štruktúru.

Príklad (mzda)

- Majme súbor údajov o mzde (skupiny mužov v štátoch východného pobrežia USA).
- Chceme *porozumieť súvislosti* medzi výškou mzdy zamestnanca a jeho vekom či dosiahnutým vzdelaním, prípadne jej vývoju v čase.

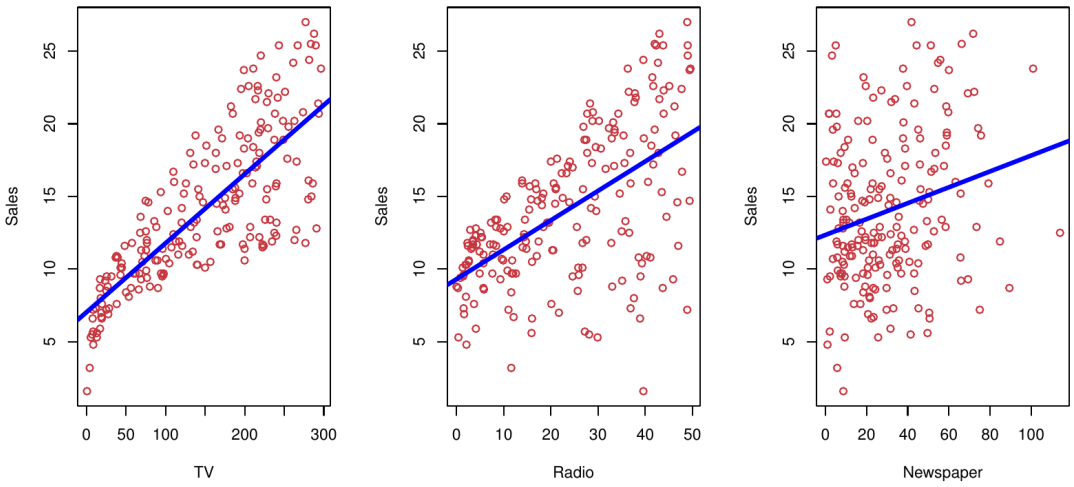


- Každý zo vstupov môže byť (s väčšou či menšou dávkou neistoty) použitý na predikciu mzdy.
- Prirodzene, najpresnejšia predpoveď bude s použitím všetkých troch vstupov.

- Najbežnejšie sú *lineárne* regresné modely, nie vždy však svojou presnosťou postačujú.

Príklad (reklama)

- Klient nás ako štatistických konzultantov najal na vyšetrenie *súvislosti medzi reklamou a odbytom* konkrétneho produktu.
- Súbor obsahuje údaje o odbyte (*sales*, v tisícoch jednotiek) a rozpočet na reklamu (v tisícoch dolárov) v troch rôznych typoch médií: TV, rádio, noviny.
- Ak dokážeme nájsť významnú súvislosť, môžeme klientovi poradiť, v ktorom médiu má posilniť reklamu, aby zvýšil odbyt.
- Cieľom je teda vyvinúť presný model na predikciu odbytu.



1.1.1 Náhodné premenné a ich vzťah

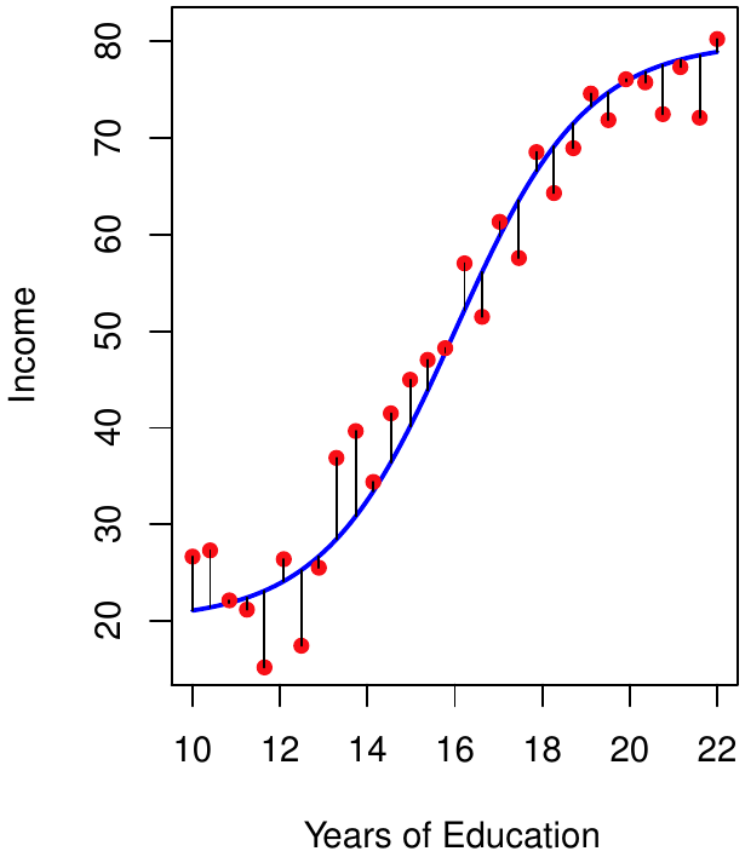
- Odbyt v predošlom príklade je výstupná premenná – inými slovami odozva (response), závislá premenná.
- Rozpočet na reklamu (v jednotlivých zložkách) je vstupná premenná – alebo aj prediktor, nezávislá premenná, črta (feature).
- Vo všeobecnosti, predpokladajme kvantitatívnu odozvu Y a sadu p rôznych prediktorov, $\vec{X} = (X_1, X_2, ...X_p)$. Medzi nimi je nejaký (systematický) vzťah a ten je reprezentovaný (neznámou) funkciou f, takže

$$Y = f(\vec{X}) + \varepsilon$$

kde ε predstavuje náhodnú chybovú zložku, ktorá **ne**závisí od $\vec{X}$ a zvyčajne má nulovú strednú hodnotu.

Príklad (príjem)

- Predpokladajme, že príjem po skončení školy by sa dal predpovedať pomocou počtu rokov venovaných vzdelaniu.
- Modrá krivka predstavuje skutočnú funkciu f. Tú však v praxi nepoznáme, ale za predpokladu určitých vlastností chybovej zložky môžeme f do istej miery odhadnúť.



1.1.2 Ciele modelovania

- Prečo vlastne potrebujeme odhad f?
- Sú dva hlavné dôvody (ciele modelovania):
  - *predpovedanie* (prediction) a
  - *dedukcia* (inference).

1.1.2.1 Predpoveď

- V mnohých situáciách sú vstupné hodnoty ľahko dostupné, no výstupné sa získavajú ťažko. Pretože chybová zložka sa nasčíta na nulu, predpoveď je jednoduchá:

$$\hat{Y} = \hat{f}(\vec{X})$$

- Tu odhad $\hat{f}$ môže byť (a v strojovom učení často býva) ako čierna skrinka.
- Význam predikcie je napr. v určení rizika nepriaznivej reakcie na určitý liek ak prediktormi sú charakteristiky pacientovej krvi (získateľné laboratórne). Je lepšie vopred predpovedať reakciu organizmu a tak sa vyhnúť podaniu lieku s nežiadúcim účinkom pri pacientoch s vysokou hodnotou odozvy.
- Presnosť predpovede závisí na dvoch chybách:
  - *redukovateľnej* (chyba odhadu funkcie) a
  - *neredukovateľnej* (šum, vplyv nemeraných premenných, nemerateľnej premenlivosti - reakcia pacienta na liek podľa nálady, odchýľka v zložení tablety atď.)

$$\begin{aligned} E[(Y - \hat{Y})^2] &= E\left[\left(f(\vec{X}) + \varepsilon - \hat{f}(\vec{X})\right)^2\right] \\ &= \left(f(\vec{X}) - \hat{f}(\vec{X})\right)^2 + var(\varepsilon) \end{aligned}$$

1.1.2.2 Dedukcia

- Často nás zaujíma spôsob, ako je odozva ovplyvňovaná zmenou vstupov. Preto síce chceme odhadnúť funkciu f, ale nie kvôli predpovediam, ale analýze jej tvaru.
- Tu už sa s funkciou nemôže zaobchádzať ako s čiernou skrinkou.
- Chceme odpovede na otázky
  - Ktoré premenné sú asociované s odozvou?
  - Aký je vzťah medzi Y a jednotlivými prediktormi (kladný, záporný)? Ovplyvňuje ho nejaký iný prediktor?
  - Môže byť tento vzťah zhrnutý do lineárnej rovnice?

1.1.2.3 Ilustrácie cieľov

- Príkladom modelovania kvôli predikcii môže byť záujem komerčnej spoločnosti identifikovať jedincov ako potenciálnych zákazníkov na základe demografických ukazovateľov, pričom odozva na marketingovú kampaň (pozitívna alebo negatívna) slúži ako výstup. Firmu teda nezaujímajú detaily vzťahu medzi prediktormi a odozvou ale predpovede.

- V príklade s inzerciou a odbytom naopak môže byť záujem o odpovede na otázky:
  - Ktoré médiá prispievajú k odbytu?
  - Ktoré médiá vytvárajú najväčšie oživenie obchodu?
  - Aké navýšenie predaja možno očakávať s konkrétnou investíciou do reklamy v TV?

1.1.3 Metódy aproximácie skutočnej funkcie

- Od účelu závisí voľba metódy:
  - jednoduchší ale interpretovateľný model, alebo
  - predikčne výkonný no menej zrozumiteľný model, s ktorým je dedukcia ťažšia.
- Preberieme viacero lineárnych i nelineárnych prístupov ku odhadu f. Tieto metódy majú niektoré črty spoločné (a na tie sa zameriame).
- Majme n pozorovaní, ktoré sa súhrnne nazývajú *trénovacia vzorka*, pretože pomocou nich naučíme konkrétnu metódu odhadnuť f.
- Nech x_{ij} je i-te pozorovanie j-teho prediktoru, y_i zodpovedajúce i-te pozorovanie odozvy, pričom $i = 1, ...n$ a $j = 1, ...p$. Potom trénovaciu vzorku tvoria dvojice $\{(x_1, y_1), ..., (x_n, y_n)\}$ kde $x_i = (x_{i1}, x_{i2}, ..., x_{ip})$.
- Cieľom je aplikovať štatistickú metódu učenia na trénovaciu vzorku na odhad neznámej funkcie f.
- Väčšina štatistických metód modelovania je charakterizovaných buď ako *parametrické*, alebo *neparametrické*.

1.1.3.1 Parametrické metódy

- Parametrický prístup sa skladá z dvoch krokov
  1. Vytvorenie *predpokladu o tvare* funkcie (funkčnom predpise), napr. že f je lineárna v premenných $\vec{X}$, teda $f(\vec{X}) = \beta_0 + \beta_1 X_1 + ... + \beta_p X_p$. Tak sa problém odhadu f značne zjednodušil (z odhadu celej p-rozmernej funkcie) na odhad $p+1$ parametrov $\beta_0, ..., \beta_p$.
  2. Voľba a použitie metódy, ktorá náš model natrénuje na dátach (napasuje na pozorovania). V prípade lineárneho modelu je to najčastejšie (bežná) metóda najmenších štvorcov.
- Ukážeme si rôzne funkčné predpisy i metódy odhadu parametrov.
- Výhoda parametrického prístupu je teda v zjednodušení odhadu.

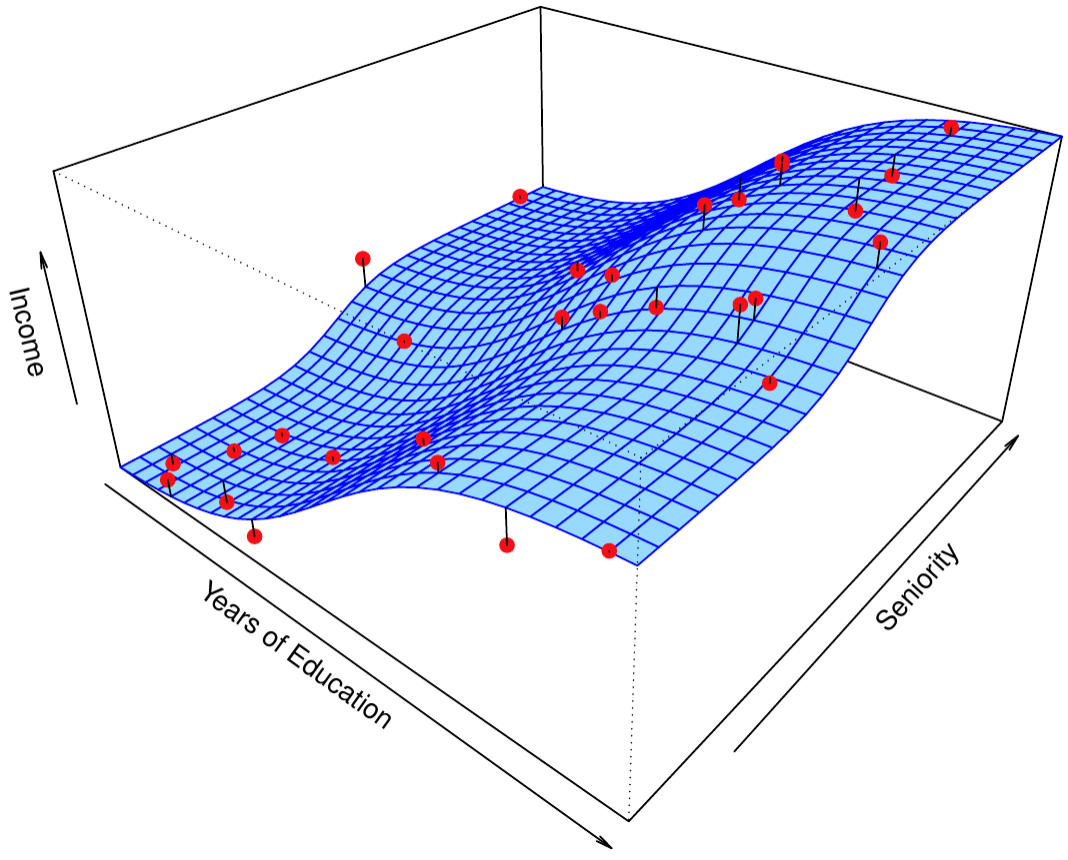
- Naopak, jeho potenciálna nevýhoda je v tom, že zvolený model zvyčajne celkom nevystihne skutočný tvar neznámej funkcie f. To sa síce dá obísť voľbou flexibilnejších modelov, ktoré sú schopné imitovať veľa rôznych tvarov funkcií, no vo všeobecnosti aj vyžadujú odhad väčšieho počtu parametrov.
- A komplexnejšie modely zas môžu viesť k javu, ktorý sa nazýva *"overfitting"*. Čiže flexibilné modely sú náchylné na prehnanú snahu popísať okrem funkcie aj náhodný šum.

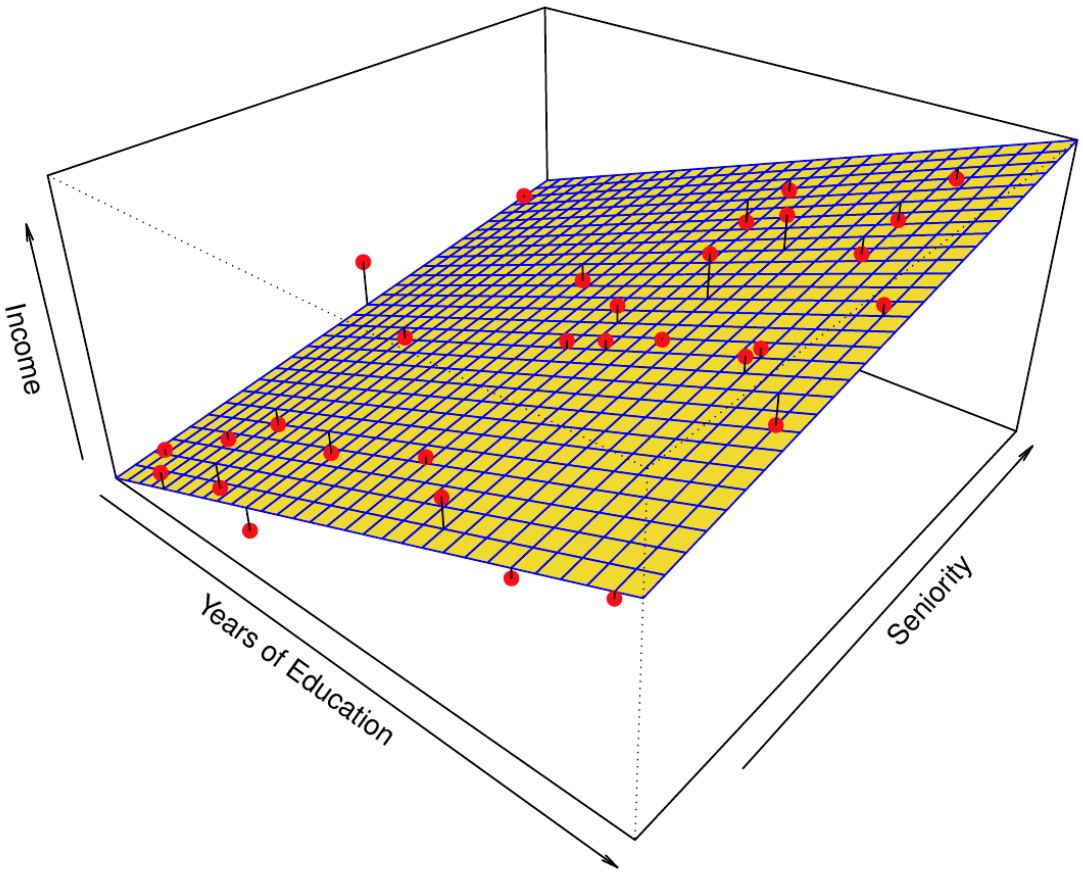
Príklad (príjem)

- Parametrické metódy si ilustrujeme na príklade závislosti príjmu od vzdelania a skúsenosti (prax, trvanie kariéry) popísanou lineárnym modelom

$$prjem \approx \beta_0 + \beta_1 \times vzdelanie + \beta_2 \times sksenos$$

- Na obrázku vľavo je skutočný funkčný vzťah (poznáme ho, pretože dáta sú simulované), vpravo jeho lineárna aproximácia.





1.1.3.2 Neparametrické metódy

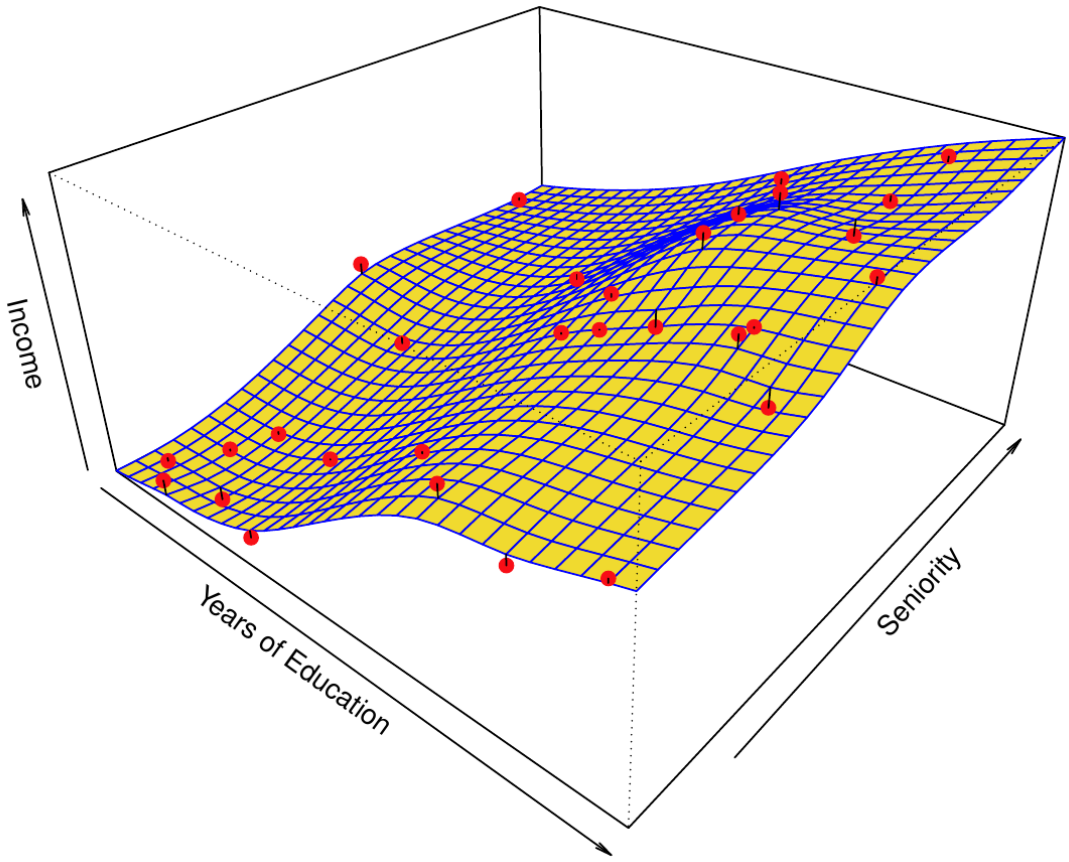
- Tento prístup sa neobmedzuje na predpoklad o tvare funkcie, ale snaží sa k dátam primknúť čo najtesnejšie bez toho, aby aproximácia bola príliš rovná, alebo naopak, príliš pokrútená.
- Pretože však neredukuje problém odhadu f na odhad malého počtu parametrov, jeho hlavnou *nevýhodou je nutnosť veľkého počtu údajov.*

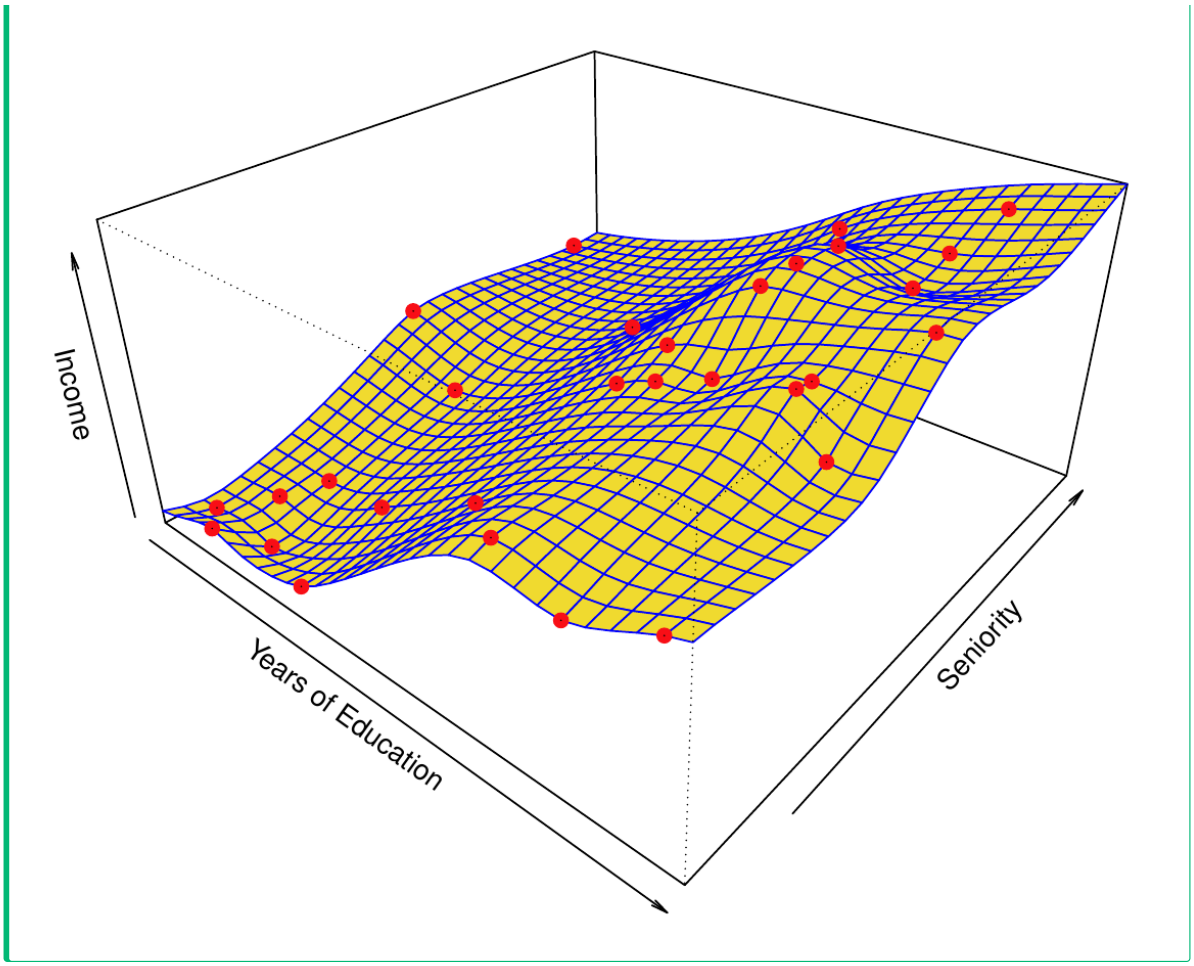
Príklad (príjem)

- Nadviažeme na predošlý príklad s príjmom. Na obrázku vľavo je zobrazený neparametrický model zo skupiny vyhladzujúcich splajnov (2D smoothing spline, konkrétne thin-plate spline) s optimálnym stupňom vyhladenia.
- Na pravom obrázku je splajn s vysokým stupňom aproximácie (nízkym stupňom

vyhladenia), ktorý prechádza cez každý bod trénovacej vzorky (prakticky je to interpolačný splajn) a necháva chybovú zložku úplne nulovú.

- Pravý obrázok je príkladom overfitting-u.





- Prefitovanie je nežiadúce, pretože znižuje presnosť predpovedí (mimo trénovacej vzorky). Preto potrebujeme metódy na voľbu správneho stupňa vyhladenia.

1.1.3.3 Kompromis (presnosť vs. interpretovateľnosť)

- S rozvojom IT je nedostatok dát či výpočtová kapacita čoraz zriedkavejším problémom. Prečo by sme potom nemali zakaždým uprednostniť flexibilnú metódu odhadu f pred tou reštriktívnou?
- Dôvodom je schopnosť interpretácie modelu, ktorá je žiadúca, ak nám primárne nejde o predikcie ale o dedukciu (porozumenie procesu, ktorý generoval údaje).

1.1.4 Prítomnosť odozvy

- Väčšina úloh štatistického modelovania spadá do dvoch kategórií – asistované a neasistované – podľa toho, či niektorá z náhodných premenných vystupuje v role vysvetľovanej premennej alebo nie.

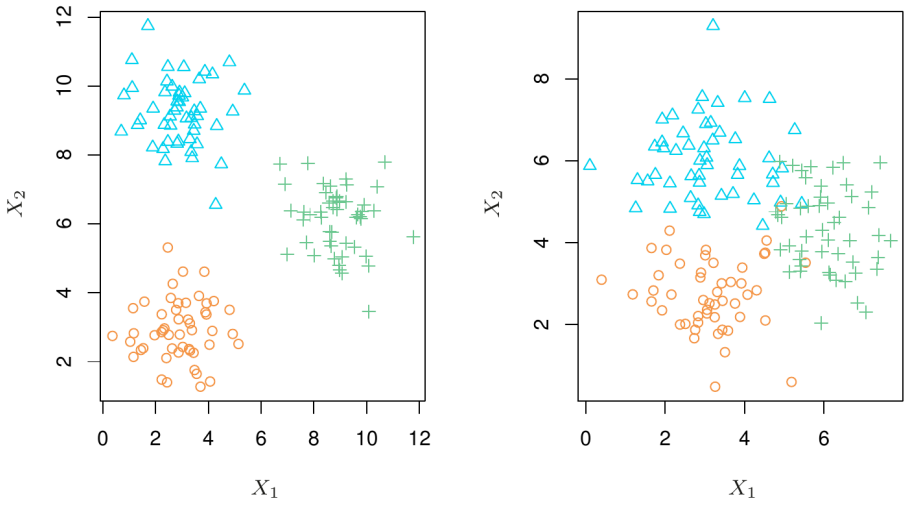
1.1.4.1 Asistované učenie

- Všetky doteraz uvádzané príklady sú z oblasti asistovaného učenia, kde každé pozorovanie hodnôt prediktorov je spojené s pozorovaním odozvy a našim cieľom je modelovať ich vzťah (či už na jeho lepšie pochopenie alebo predpovedanie ďalších hodnôt). Prirodzene sem patria klasické metódy ako lineárna či logistická regresia, ale i množstvo novších (GAM, boosting, SVM, ANN).

1.1.4.2 Neasistované učenie

- Na druhej strane, neasistované učenie je náročnejšie v tom, že žiadne pozorovanie odozvy k dispozícii nie je. Akoby sme modelovali naslepo. Typickým príkladom je zhluková

analýza (clustering), v ktorej sa na základe podobných hodnôt premenných pozorovania zatrieďujú do relatívne odlišných skupín. Napr. pomocou takých charakteristík, ako sú PSČ, príjem v rodine a nákupné návyky, chceme pozorovania (zákazníkov) zatriediť do skupín s veľkými výdavkami na nákupy (tí, čo veľa míňajú) a s nízkymi výdavkami (sporovliví).



- Na ilustračnom obrázku vľavo sú skupiny jasne odlíšené (takže zhlukovanie je jednoduché) zatiaľčo vpravo sú značné presahy medzi skupinami (zhlukovanie je veľmi obtiažne).
- Ak je prediktorov viac, p, jeden bodový graf nestačí, treba ich $p(p-1)/2$, a tak namiesto vizuálneho posúdenia sa musíme spoľahnúť na automatické metódy.
- V jednej úlohe je možné aj kombinovať oba prístupy, napr. keď hodnoty odozvy sú dostupné iba pre časť pozorovaní prediktorov: polo-asistované učenie (semi-supervised).

1.1.5 Typ odozvy (regresia vs. klasifikácia)

- Premenné sa delia na kvantitatívne a kvalitatívne (kategorické), resp. spojité a diskrétne
- Napr. {vek, výška, príjem, cena} oproti {stav, pohlavie, značka, indikácia zadĺženia, druh choroby}
- Modelovanie
  - kvantitatívnej odozvy sa označuje ako *regresná úloha* (napr. lineárna regresia MNŠ)
  - kvalitatívnej odozvy ako *klasifikačná úloha*
- Hranica medzi nimi nie je vždy ostrá, napr. pre binárne premenné sa používa logistická regresia, ktorá napriek svojmu názvu je klasifikačnou metódou, ale keďže priamo modeluje pravdepodobnosť zaradenia do jednotlivých tried, má veľa spoločné s lineárnou regresiou.

- Niektoré metódy ako k-najbližších susedov sa dajú použiť v oboch kontextoch (úlohách).
- Typ premenných v úlohe prediktorov je menej podstatný, pred samotnou analýzou sa však predpokladá ich správne "kódovanie".

1.2 Vyhodnotenie modelu

- Cieľom predmetu je zoznámiť sa so širokou paletou nástrojov štatistického modelovania.
- Prečo?
- Pretože neexistuje niečo ako supermetóda, ktorá funguje najlepšie v každej situácii, pre každý dataset.
- Výber najvhodnejšej metódy je tak dôležitou úlohou štatistického modelovania a prakticky môže byť jednou z jej najnáročnejších častí.

1.2.1 Meranie kvality

- Na výber vhodnej metódy budeme potrebovať nejakú mieru zhody modelu s realitou, predikcie s pozorovaním.
- V regresnej úlohe je najčastejšie používanou mierou stredná kvadratická chyba (mean squared error) MSE

$$MSE = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{f}(x_i)\right)^2$$

  ktorá je malá, ak všetky predpovede sú blízko skutočnej odozvy, a veľká, ak sa niektoré odhady od reality značne odlišujú.
- Ak sa MSE vypočíta z rovnakých údajov, ako na ktorých bol model natrénovaný, mala by sa označiť pojmom *trénovacia MSE*.
- Zvyčajne nám však neveľmi záleží na tom, ako model funguje na známych dátach, ale naopak, chceme poznať presnosť predpovedí, keď (už natrénovaný) model aplikujeme na nové dáta.

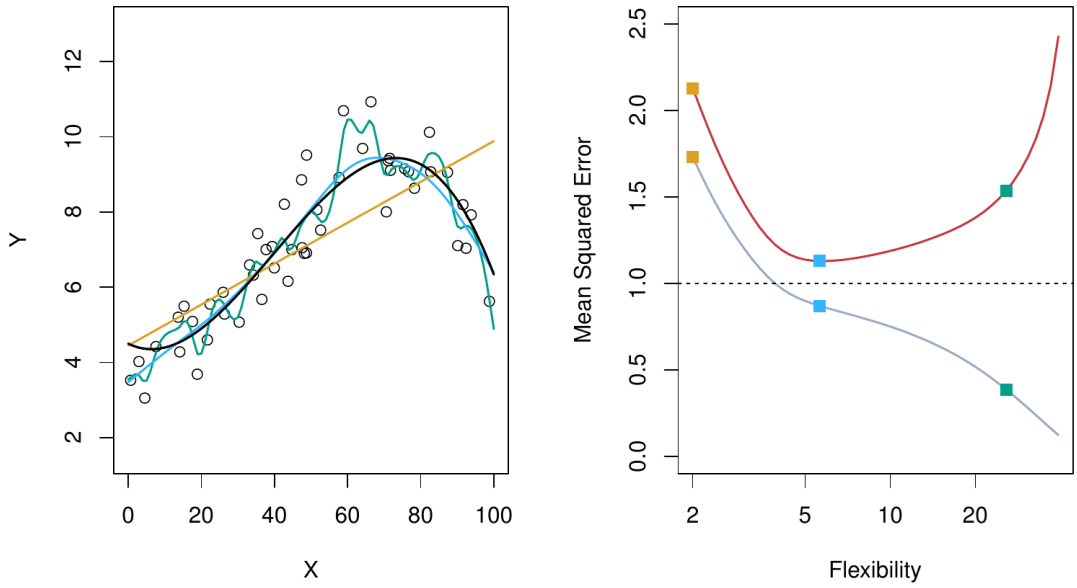
Prečo?

- Predpokladajme, že chceme zbohatnúť na obchodovaní s akciami. Vytvoríme model, ktorý pomocou predošlých výnosov odhadne aktuálnu cenu. Teraz už len spoľahlivo zistiť, či bude cena zajtra rásť (a treba nakupovať), alebo klesať (a akcie rýchlo predať).
- Alebo majme klinické záznamy (napr. hmotnosť, tlak, výška, vek, rodinná anamnéza) určitého počtu pacientov, o ktorých vieme aj to, či majú cukrovku alebo nie. Tieto záznamy použijeme na natrénovanie modelu. Predpovede z neho však nepot-

rebujeme na overenie, či pacient z našej skupiny cukrovku má (pretože to vieme), ale na predikciu rizika pri budúcich pacientov (na základe ich klinických záznamov).

- Formálne, ak trénovacia vzorka je $(x_1, y_1), ..., (x_n, y_n)$, označme nové pozorovanie ako (x_0, y_0). Zaujíma nás, ako vybrať štatistickú metódu, ktorá minimalizuje nie trénovaciu MSE ale $E[(y_0 - \hat{f}(x_0))^2]$, čiže tzv. *testovaciu MSE*.
- Na to je potrebná testovacia vzorka.
- Trénovacia MSE nie je dobrou náhradou testovacej MSE.

Ilustrácia (nelinárna funkcia)

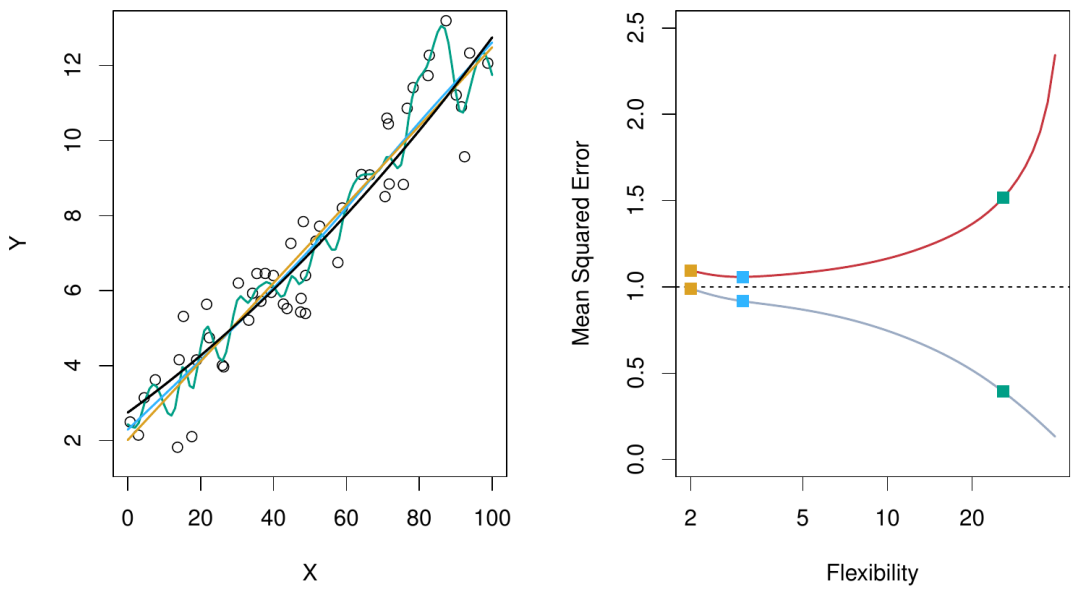


- Na ľavom obrázku je skutočná funkcia závislosti Y od X (čierna), dáta z nej simulované (body), lineárny regresný model (oranžová priamka) a dve splajnové krivky (modrá a zelená).
- Napravo je graf trénovacej MSE (sivá), testovacej MSE (červená) a najmenšej dosiahnuteľnej testovacej MSE (prerušovaná), ktorá sa rovná neredukovateľnej chybe predpovedí, $var(\varepsilon)$. Prvé dve sa menia s prispôsobivosťou modelov, ktorá sa formálne označuje ako *stupne voľnosti* modelu. Najmenej stupňov voľnosti tu má priamka (oranžové štvorce), ktorá - ako vidno aj z testovacej MSE - nedostatočne aproximuje skutočnú funkciu. Modrá splajnová krivka je blízko optimálnej. Zelený model je reprezentovaný bodom na vzostupnej časti grafu testovacej MSE, pretože sa príliš vlní (overfitting).

- To, že trénovacia chyba je (takmer) vždy menšia ako testovacia je dané tým, že väčšina

štatistických metód odhadu (priamo alebo nepriamo) minimalizuje trénovaciu MSE.

Ilustrácia (takmer lineárna funkcia)



- V tomto prípade lineárny model je blízky optimálnemu, preto prevážia jeho interpretačné vlastnosti a býva uprednostnený pred lepším, ale zložitejším modelom.

- V praxi sa testovacia chyba odhaduje z trénovacích dát pomocou *resamplovacích metód*, konkrétne tzv. *krížovou validáciou*.

1.2.2 Kompromis (výchyľka vs. rozptyl)

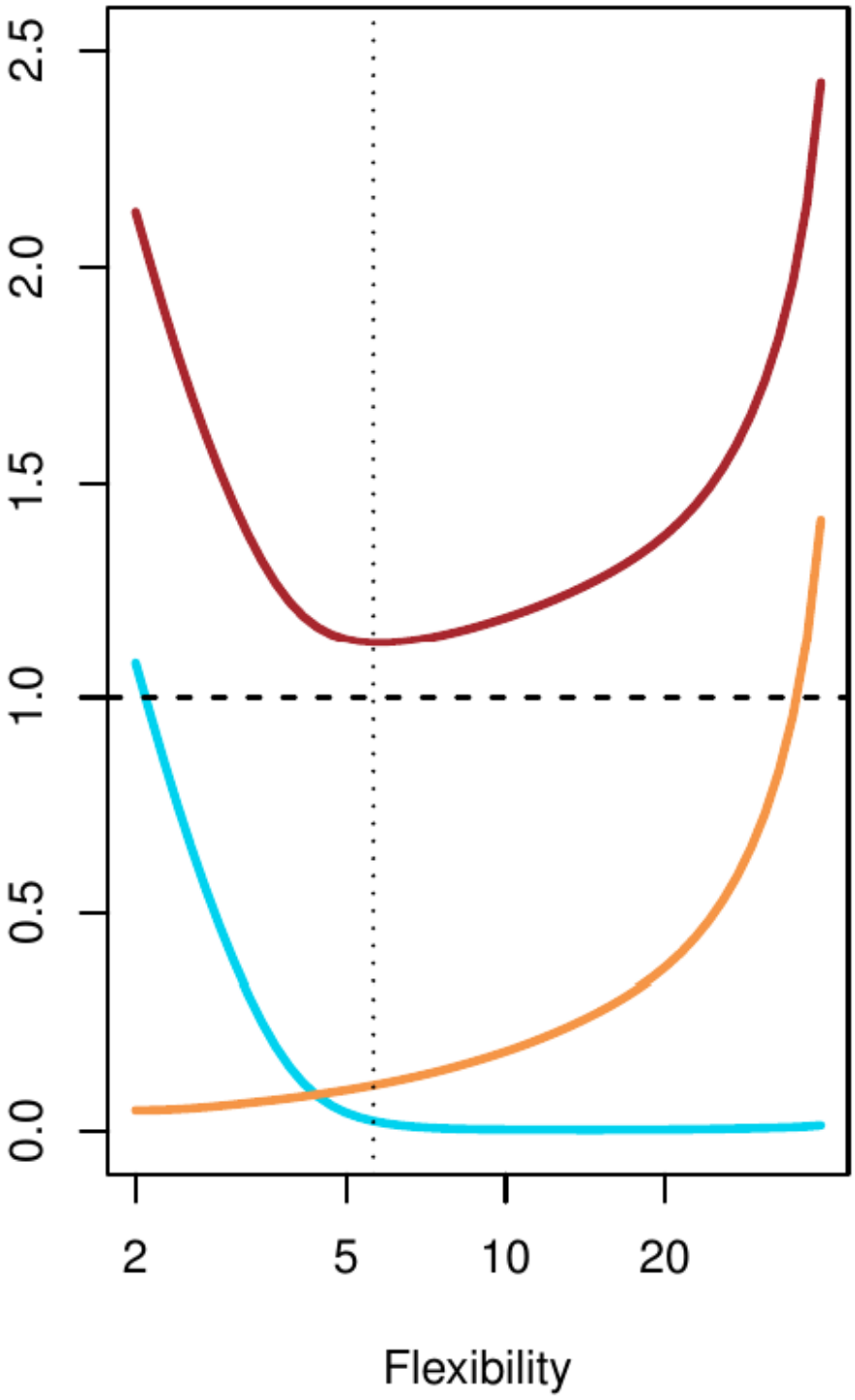
- Tvar písmena U grafu testovacej MSE je dôsledkom kompozície dvoch druhov chýb: *variancie* predpovede $\hat{f}(x_0)$ a vychýlenia (*bias*) predpovede.
- Celkovo teda môžeme testovaciu MSE rozložiť ako vo vzťahu

$$E\left[(y_0 - \hat{f}(x_0))^2\right] = var(\hat{f}(x_0)) + [bias(\hat{f}(x_0))]^2 + var(\varepsilon)$$

- Optimálny model tak získame minimalizáciou rozptylu a súčasne aj výchylky.

- Variancia predstavuje mieru, do akej by sa odhad funkcie zmenil, ak by sme na trénovanie modelu použili iný náhodný výber (z rovnakej populácie). To znamená, že ohybnejšie modely sú citlivejšie na zmeny v dátach.
- Bias predstavuje chybu aproximácie (zložitej) skutočnosti (jednoduchým) modelom.

- Nájsť dobrý kompromis medzi oboma chybami (*bias-variance trade-off*) je jednou z hlavných tém štatistického modelovania.


2.5
2.0
1.5
1.0
0.5
0.0
2
5
10
20
Flexibility

1.2.3 Presnosť v klasifikácii

- Všetky spomenuté princípy platia aj pri klasifikačnej úlohe, iba namiesto strednej kvadratickej chyby použijeme mieru vhodnú pre kvalitatívnu premennú, napr. chybovosť (*error rate*, ER). Trénovacia ER sa vypočíta vzťahom

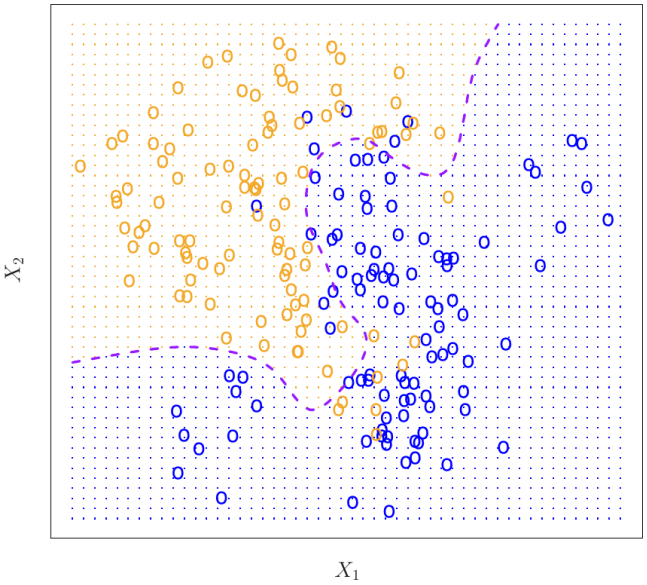
$$ER = \frac{1}{n}\sum_{i=1}^{n} I(y_i \neq \hat{y}_i)$$

  kde $I(x)$ je indikačná funkcia, ktorá sa rovná 1 ak podmienka x je splnená, inak sa rovná 0.
- Testovacia ER je definovaná analogicky ako MSE, čiže ako stredná hodnota ER pre nové pozorovanie, $E\left[I(y_0 \neq \hat{y}_0)\right]$.
- Dobrý klasifikátor je taký model, pre ktorý je testovacia ER najmenšia.
- Dá sa ukázať, že takúto vlastnosť spĺňa jeden jednoduchý klasifikátor, ktorý každé pozorovanie zaradí do najpravdepodobnejšej triedy (podmienenej hodnotami prediktorov), teda do triedy j pre ktorú podmienená pravdepodobnosť

$$Pr(Y = j|\vec{X} = \vec{x}_0)$$

  je najväčšia. Takýto klasifikátor sa nazýva Bayesov.
- V prípade binárnej odozvy je tou hraničnou pravdepodobnosťou hodnota 1/2. V priestore prediktorov tvorí tzv. *Bayesovu rozhodovaciu hranicu*.

Ilustrácia (Bayesova rozhodovacia hranica)



2 Lineárny regresný model

2.1 Úvod

- Lineárna regresia je jednoduchý prístup ku asistovanému učeniu.
- Oproti moderným metódam sa môže zdať až príliš jednoduchý, ale zároveň patrí medzi najužitočnejšie, bežne používané nástroje štatistického modelovania.
- Zároveň je dobrým štartovacím bodom pre pochopenie a využitie novších prístupov, pretože mnohé z nich ho zovšeobecňujú.

Príklad (reklama)

- Oprášme príklad inzerovania v troch médiách. Našou úlohou je (na základe dát) navrhnúť marketingovú stratégiu na zvýšenie odbytu pre ďalšie obdobie. Aké informácie by na poskytnutie takého odorúčania mohli byť užitočné?
- Tu je zopár dôležitých otázok, na ktoré budeme hľadať odpovede:

  1. *Existuje vzťah medzi rozpočtom na reklamu a odbytom?* Prvou úlohou je teda zistiť, či pozorovnia poskytujú dostatočný dôkaz o súvislosti medzi premennými. Ak nie, potom je zbytočné utrácať peniaze za reklamu.

  2. Ak existuje vzťah, *aký je silný?*
  3. *Ktoré médiá* súvisia s odbytom? Chceme rozlíšiť významnosť príspevok jednotlivých médií, ak sme investovali do všetkých troch.
  4. *Aký silný je vzťah medzi odbytom a jednotlivými médiami?* O koľko sa zvýši odbyt s každým dolárom naliatym do reklamy v konkrétnom médiu a ako presne dokážeme tento nárast predpovedať?
  5. Aká je presnosť predpovede budúceho odbytu? Pre akúkoľvek kombináciu objemu reklamy v médách.
  6. *Je vzťah lineárny?* Ak sa dá vzťah reklamy v jednotlivých médiách ku odbytu vyjadriť priamkou, potom lineárna regresia je vhodný nástroj. Ak nie, dá sa linearizovať transformáciou prediktorov alebo odozvy?
  7. *Je medzi médiami synergia?/* Je možné, aby investovanie napr. $50,000 do reklamy v TV a $50,000 v rádiu prineslo vyšší odbyt než $100,000 iba v jednom z nich? Toto sa v štatistike nazýva efekt *interakcie*.

- Lineárna regresia sa dá použiť na zodpovedanie všetkých týchto otázok.

2.2 Jednoduchá regresia (jeden prediktor)

- Lineárny regresný model s jediným prediktorom X sa zvykne označovať ako jednoduchá regresia (*simple* linear regression), formálne zapísaný vzťahom

$$Y = \beta_0 + \beta_1 X + \varepsilon \tag{2.1}$$

kde β_0, β_1 sú neznáme parametre (koeficienty) vyjadrujúce priesečník regresnej priamky s osou Y (intercept) a sklon (slope). Chybový člen ε je náhodná premenná s nulovou strednou hodnotou, štandardnou odchýlkou σ a je nezávislá od X.

2.2.1 Odhad parametrov

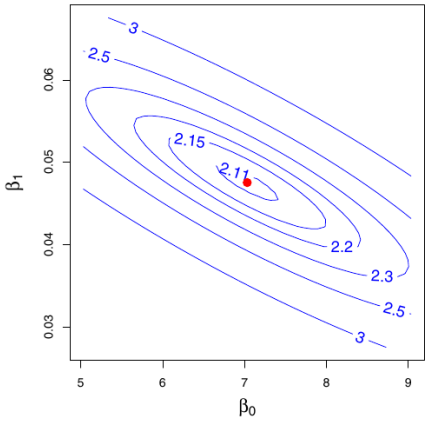
- Parametre musia byť odhadnuté tak, aby priamka bola čo najbližšie ku pozorovaniam X a Y, teda bodom so súradnicami $(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$.
- Najpoužívanejšou mierou blízkosti je súčet štvorcov rezíduí (residual sum of squares), $RSS = \sum_{i=1}^{n} \hat{\varepsilon}_i^2$
- Rezíduá sú odchýlky reality od modelu, $\hat{\varepsilon}_i = y_i - \hat{y}_i = y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i$.
- Minimalizáciou RSS (metóda najmenších štvorcov, least squares) dostaneme odhady parametrov

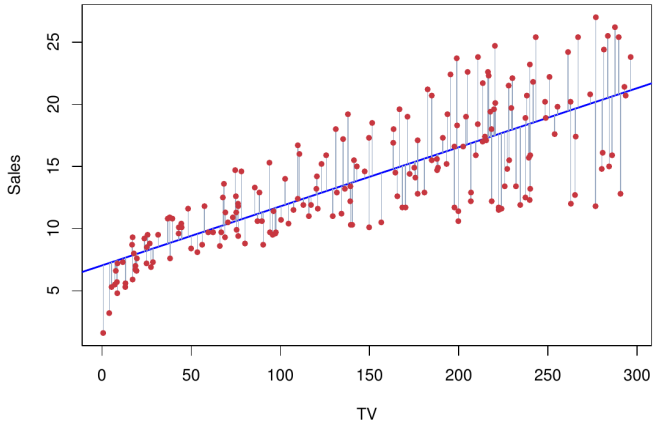
$$\hat{\beta}_1 = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n}(x_i - \bar{x})^2},$$
$$\hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x},$$

kde $\bar{y} \equiv \frac{1}{n}\sum_{i=1}^{n} y_i$ a analogicky aj $\bar{x}$ sú výberové priemery.

Príklad (reklama)

- Odhad metódou najmenších štvrcov v priestore parametrov (vľavo s izočiarami RSS) a v priestore premenných (vpravo s naznačením rezíduí):






```
dat_Advertising <- read.csv("data/Advertising.csv")[-1]
fit <- lm(sales ~ TV, dat_Advertising)
fit |> coefficients() |> round(3)
```

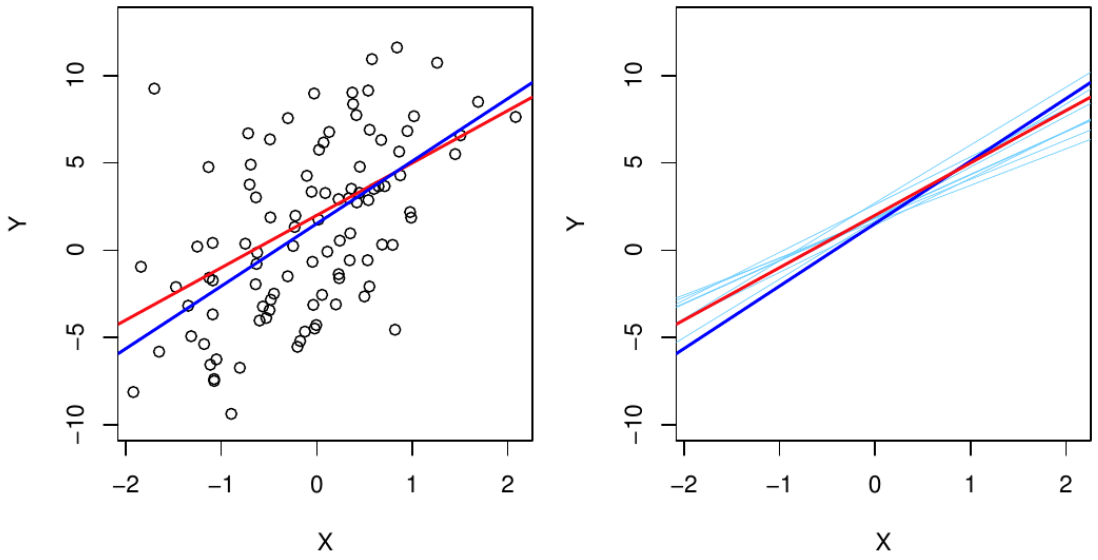
```
(Intercept)          TV
      7.033       0.048
```

- Podľa modelu by bol priemerný objem predaja pri nulovej reklame v TV 7,033 jednotiek a s každým navýšením o 1,000 dolárov (o 1 v mierke datasetu) by vzrástol o 48 jednotiek (0.048 v mierke datasetu).

2.2.2 Presnosť parametrov

Ilustrácia (priamka)

- Nech je skutočný regresný vzťah daný vzťahom $f(X) = 2 + 3X$.
- Odhad na základe jednej realizácie náhodného výberu (simulácie z modelu $Y = 2 + 3X + \varepsilon$) je zobrazený na obrázku vľavo (spolu s dátami). Na obr. vpravo je zobrazený spolu s odhadmi regresnej priamky získaných na základe ďalších 10 simulovaných datasetov.



- Všetky odhadnuté priamky sa nachádzajú niekde okolo skutočnej priamky v dôsledku neistoty v určení parametrov.
- Pretože metóda najmenších štvorcov dáva *nevychýlený* odhad (za podmienky splnenia predpokladov), spriemerovanie veľkého množstva odhadov vedie ku skutočnej hodnote (tu ku priamke).
- V praxi máme iba jeden dataset, preto nás zaujíma, ako blízko sme sa pomocou neho dostali ku skutočnosti.

- Podobne ako presnosť odhadu nepodmienenej strednej hodnoty (čo je vlastne 1D prípad) závisela od štandardnej odchýlky údajov a ich počtu (pripomeňme si vzťah $var(\hat{\mu}) = SE(\hat{\mu})^2 = \sigma^2/n$), tak to bude platiť aj pri presnosti odhadu parametrov,

$$var(\hat{\beta}_0) = \sigma^2 \left(\frac{1}{n} + \frac{\bar{x}^2}{\sum_{i=1}^{n}(x_i - \bar{x})^2} \right), \quad var(\hat{\beta}_1) = \left(\frac{\sigma^2}{\sum_{i=1}^{n}(x_i - \bar{x})^2} \right),$$

na ich presnosť navyše vplýva aj rozptyl hodnôt x_i.

- Rozptyl σ^2 je tiež neznámy parameter modelu. Odhadne sa jednoducho ako

$$\hat{\sigma}^2 = RSS/(n-2).$$

- Štandardné chyby sa dajú použiť na konštrukciu *intervalov spoľahlivosti*,

$$\hat{\beta}_j \pm t_{1-\alpha/2}(n-2) \cdot SE(\hat{\beta}_j).$$

  Interpretácia: ak by sme veľa-krát opakovali náhodný výber a zakaždým určili 95%-ný interval spoľahlivosti, potom 95% z nich by obsahovalo skutočnú hodnotu parametra.
- Rovnako tak sa dajú použiť na *testovanie štatistických hypotéz* o parametroch, napr. H_0: $\beta_1 = 0$ (t.j. neexistuje vzťah medzi X a Y). Treba rozhodnúť, či $\hat{\beta}_1$ je dostatočne blízko (H_0), resp. ďaleko (H_1) od nuly. Ale koľko je "dostatočne"? To závisí od presnosti odhadu. Preto testovacia štatistika t-testu

$$t = \frac{\hat{\beta}_1 - 0}{SE(\hat{\beta}_1)} \quad \sim t(n-2)$$

  vyjadruje počet štandardných chýb o ktoré je $\hat{\beta}_1$ vzdialené od 0.

2.2.3 Vyhodnotenie presnosti modelu

- Akonáhle sme pomocou testu sklonu regresnej priamky vyhodnotili, že vzťah medzi premennými je významný, zaujíma nás, *do akej miery model vystihuje dáta*.
- Jednou populárnou mierou (do akej model *ne*vystihuje dáta) je reziduálny rozptyl (resp. reziduálna štandardná chyba). Jej nevýhodou je závislosť od mierky dát, takže je občas ťažké posúdiť, aké hodnoty ešte predstavujú dobrú zhodu.
- Alternatívnou mierou je koeficient determinácie

$$R^2 = \frac{TSS - RSS}{TSS} = 1 - \frac{RSS}{TSS}$$

  kde $TSS = \sum_{i=1}^{n}(y_i - \bar{y})^2$ je celková suma štvorcov (čiže RSS obyčajného priemeru). Je zjavné, že ako pomer variancie vysvetlenej modelom ku celkovej variancii, je R^2 bezrozmerný a v rozsahu intervalu $[0, 1]$, takže nezávisí od mierky Y.
- Je tiež zaujímavé, že R^2 sa zhoduje s druhou mocninou koeficientu korelácie medzi X a Y.

Príklad (reklama)

- Vyjadrime vzťah reklamy v TV a odbytom pomocou lineárneho regresného modelu.

```
fit |> summary()
```

```
Call:
lm(formula = sales ~ TV, data = dat_Advertising)

Residuals:
    Min      1Q  Median      3Q     Max
-8.3860 -1.9545 -0.1913  2.0671  7.2124

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 7.032594   0.457843   15.36   <2e-16 ***
TV          0.047537   0.002691   17.67   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.259 on 198 degrees of freedom
Multiple R-squared:  0.6119,    Adjusted R-squared:  0.6099
F-statistic: 312.1 on 1 and 198 DF,  p-value: < 2.2e-16
```

- Odhad parametra β_1 je asi 17-násobne väčší než jeho štandardná chyba, preto aj p-hodnota je veľmi malá a H_0 zamietneme, čo značí výrazný vzťah medzi TV a *sales*.
- Skutočný odbyt by v priemere fluktuoval 3.26 tisíc jednotiek okolo predpovede.
- Model pomocou reklamy v TV vystihuje asi 61% variability odbytu.

2.3 Viacnásobná regresia (viac prediktorov)

- V praxi je zvyčajne k dispozícii viac premenných, ktoré môžu byť užitočné na vysvetlenie zmeny v strednej hodnote odozvy.
- Zopakovať jednoduchú regresiu pre každú premennú naráža (minimálne) na problém, ako z viacerých modelov skonštruovať jedinú predpoveď.
- Najjednoduchšie rozšírenie jednoduchej lineárnej regresie predpokladá aditivitu vplyvov jednotlivých prediktorov $X_1, X_2, \ldots, X_p$ a má tvar

$$Y = \beta_0 + \beta_1 X_1 + \ldots + \beta_p X_p + \varepsilon \tag{2.2}$$

- Hodnota β_j sa interpretuje ako priemerný vplyv jednotkového nárastu premennej X_j na odozvu Y *za stavu, v ktorom všetky ostatné prediktory zostávajú nezmenené.*

2.3.1 Odhad parametrov

- Odhad je analogický jednorozmernému prípadu a jednoduchšie sa reprezentuje v maticovej podobe.
- Nech $\mathbf{X}$ je matica plánu (regresná matica), ktorej i-ty riadok tvorí vektor $(1, x_{i1}, \dots, x_{ip})$, pričom $i = 1, \dots, n$, ďalej $\mathbf{y}$ je vektor pozorovaní odozvy Y a nakoniec $\beta = (\beta_0, \beta_1, \dots, \beta_p)'$.

- Odhad parametrov metódou najmenších štvorcov dostaneme zo vzťahu

$$\widehat{\beta} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{y}.$$

  Potom predikcie dostaneme ako podmienenú strednú hodnotu $\hat{\mathbf{y}} \equiv E(Y|\mathbf{X}) = \mathbf{X}\widehat{\beta}$, reziduá budú $\hat{\varepsilon} = \mathbf{y} - \hat{\mathbf{y}}$ a odhad rozptylu chybovej zložky

$$\hat{\sigma}^2_\varepsilon = \frac{\hat{\varepsilon}'\hat{\varepsilon}}{n - (p+1)},$$

- Kovariančná matica odhadu neznámych parametrov sa vypočíta zo vzťahu

$$\widehat{\Sigma}_\beta = \hat{\sigma}^2_\varepsilon(\mathbf{X}'\mathbf{X})^{-1}.$$

  Jej diagonálu tvoria štvorce štandardných chýb odhadov parametrov.

Príklad (reklama)

- Okrem TV uvažujme aj reklamu v rádiu a novinách.
- Jednoduché regresné modely by mali nasledujúce odhady parametrov:

```
for(i in names(dat_Advertising)[1:3]) {
  lm(paste("sales ~ ", i), dat_Advertising) |>
    summary() |> coef() |> round(4) |> print()
  cat("\n")
}
```

```
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   7.0326     0.4578 15.3603        0
TV            0.0475     0.0027 17.6676        0

            Estimate Std. Error t value Pr(>|t|)
(Intercept)   9.3116     0.5629 16.5422        0
radio         0.2025     0.0204  9.9208        0

            Estimate Std. Error t value Pr(>|t|)
(Intercept)  12.3514     0.6214 19.8761   0.0000
newspaper     0.0547     0.0166  3.2996   0.0011
```

- Viacnásobný lineárny regresný model má odhady parametrov pre TV a rádio veľmi podobné, no v prípade novín sa zásadne líši: vplyv reklamy v novinách vyzerá byť nevýznamný:

```
fit <- lm(sales ~ TV + radio + newspaper, dat_Advertising)
fit |> summary() |> coef() |> round(4)
```

```
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   2.9389     0.3119  9.4223   0.0000
TV            0.0458     0.0014 32.8086   0.0000
radio         0.1885     0.0086 21.8935   0.0000
newspaper    -0.0010     0.0059 -0.1767   0.8599
```

- Vysvetlenie treba hľadať v interpretáci koeficientov. Všimnime si korelácie medzi premennými:

```
cor(dat_Advertising) |> round(3)
```

```
             TV radio newspaper sales
TV        1.000 0.055     0.057 0.782
radio     0.055 1.000     0.354 0.576
newspaper 0.057 0.354     1.000 0.228
sales     0.782 0.576     0.228 1.000
```

Objem reklamy v novinách koreluje s ďalším prediktorom – reklamou v rádiu. Obe médiá pozitívne korelujú s odbytom. To znamená, že v jednoduchej regresii noviny iba *sprostredkovali* (nahrádzali) vplyv reklamy v rádiu a ako také nemajú dosah na zvýšenie odbytu.

- Táto zdanlivá kontroverzia z príkladu o reklame je prítomná v mnohých reálnych situáciách.

Ilustrácia (žraloky)

- Uvažujme absurdný príklad, keď sa na základe záznamov z dovolenkových rezortov snažíme vysvetliť počet napadnutí žralokom pomocou odbytu zmrzliny.
- Je pravdepodobné, že nám vyjde pozitívna korelácia a regresný koeficient je štatisticky významný.
- Zdravý rozum nám hovorí, že netreba hneď zakazovať predaj zmrzliny.
- Dôvod je celkom zjavne v tom, že vyššia teplota priláka na pláž viac ľudí a tým stúpne spotreba zmrzliny i prípady kolízie so žralokom.

- Uvedená situácia sa nazýva zdanlivá regresia (spurious regression) a býva rizikom vtedy, ak si *koreláciu mýlime s kauzalitou*.

2.3.2 Využitie modelu

Viacnásobný regresný model nám umožňuje zodpovedať nasledujúce otázky:

1. Je aspoň jeden z prediktorov užitočný pre odhad strednej hodnoty odozvy?
2. Sú všetky prediktory užitočné, alebo len niektoré?
3. Nakoľko model vystihuje pozorované dáta?
4. Za predpokladu danej množiny prediktorov, aká je predikcia hodnoty odozvy a jej presnosť?

2.3.2.1 Test prítomnosti vzťahu s aspoň jedným prediktorom

- V prípade jednoduchej regresie stačí zistiť, či $\beta_1 = 0$.
- Pri viacnásobnej regresii však už treba overiť platnosť zložitejšej hypotézy,

$$H_0 : \beta_1 = \beta_2 = ... = \beta_p = 0$$

  oproti alternatívnej, H_1: *aspoň jedno* $\beta_j \neq 0$.
- Testovacia štatistika má tvar

$$F = \frac{(TSS - RSS)/p}{RSS/(n-p-1)} \quad \sim F(p, n-p-1)$$

  - ak vzťah absentuje, jej hodnota (pri veľkom n) je blízka 1,
  - naopak, ak H_0 neplatí, potom $E[(TSS - RSS)/p)] > \sigma^2$

Príklad (reklama)

- Hoci sú výsledky F-testu dostupné už v súhrne modelu (dostupnom cez funkciu `summary`), ukážeme si detaily porovnaním viacnásobného regresného modelu s konštantnou funkciou (*baseline* model):

```
lm(sales ~ 1, dat_Advertising) |>
  anova(fit)
```

```
Analysis of Variance Table

Model 1: sales ~ 1
Model 2: sales ~ TV + radio + newspaper
  Res.Df    RSS Df Sum of Sq      F    Pr(>F)
```

```
1    199 5417.1
2    196  556.8  3    4860.3 570.27 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

- Z nich vidieť, že $TSS = 5417$ a $RSS = 557$, takže testovacia štatistika $F = 570$ je vysoko nad 1 a plocha pod hustotou F-rozdelenia za hodnotou F je menšia než 5% (dokonca menšia než $2.2 \cdot 10^{-16}$), a to znamená, že H_0 zamietneme: reklama v aspoň jednom médiu má významný vzťah ku odbytu.

- Podobne sa dá overiť hypotéza, že iba časť (napríklad posledných q) parametrov je nulová, $H_0 : \beta_{p-q+1} = ... = \beta_p = 0$. Potom druhý, porovnávací linárny model (baseline) bude obsahovať všetky okrem tých q testovaných parametrov a testovacia štatistika sa zmení na

$$F = \frac{(RSS_0 - RSS)/q}{RSS/(n-p-1)} \quad \sim F(q, n-p-1)$$

  kde RSS_0 zodpovedá baseline modelu (a teda i nulovej hypotéze).
- F-testom tak testujeme parciálny vplyv pridania ďalších parametrov do modelu.

Príklad (reklama)

- Overme, že vplyv reklamy v novinách je nevýznamný.

```
update(fit, . ~ . - newspaper, dat_Advertising) |>
  anova(fit)
```

```
Analysis of Variance Table

Model 1: sales ~ TV + radio
Model 2: sales ~ TV + radio + newspaper
  Res.Df    RSS Df Sum of Sq      F Pr(>F)
1    197 556.91
2    196 556.83  1  0.088717 0.0312 0.8599
```

- V prípade pridania iba jedného parametra je vidieť zhodu s t-testom jeho významnosti ($t^2 = F$, porovnaj s riadkom pre `newspaper` v príklade z predošlej časti).

- Ak však máme p-hodnoty ku jednotlivým parametrom, načo nám je celkový F-test? Veď sa zdá pravdepodobné, že ak aspoň jedna p-hodnota je malá, tak aspoň jeden prediktor má vzťah s odozvou, či nie? Žiaľ, takéto uvažovanie je chybné, obzvlášť pri veľkom počte prediktorov.

- Napr. nech $p = 100$ a hypotézaH_0: $\beta_1 = ... = \beta_p = 0$ je pravdivá. V dôsledku náhody približne 5 p-hodnôt (z individuálnych t-testov) bude i tak menších ako 0.05 (to je dôvod, prečo sa vo viacnásobnom testovaní zavádzajú korekcie p-hodnôt, napr. Bonferroniho).
- F-test touto devalváciou netrpí, pretože zohľadňuje počet prediktorov.
- Ak je prediktorov príliš veľa (relatívne ku počtu pozorovaní), alebo dokonca ak $p > n$, potom F-test nebude (dobre alebo vôbec) fungovať.

2.3.2.2 Dôležité prediktory

- Ak pomocou F-testu zistíme nejaký vzťah ku odozve, je prirodzené pýtať sa: *Tak ktoré prediktory sú za to zodpovedné*, t.j. ktoré sú vlastne užitočné?
- Mohli by sme sa pozrieť na individuálne p-hodnoty, ale pri väčšom počte je to riziko (nekorigované viacnásobné testovanie).
- Problému určenia užitočných premenných (*variable selection*) sa budeme dôkladnejšie venovať v samostatnej prednáške, tu spomeňme iba základné metódy.
- Pri $p = 2$ prediktoroch sú $2^p = 4$ rôzne možnosti, ako zostaviť model (bez prediktorov, iba s jedným alebo druhým a s oboma). Ako však rozhodnúť, ktorý je najlepší? Používajú sa na to rôzne štatistiky, napr. Mallowove C_p , Akaikeho a bayesovské informačné kritériá (AIC, BIC), alebo korigovaný (adjusted) R^2.
- Pri veľkom počte p je takýto postup výpočtovo náročný (už $2^{20} > 10^6$ môže byť dosť), preto sa používajú efektívnejšie postupy na výber premenných.
- Medzi klasiku patrí výber premenných po krokoch (*stepwise selection*):
  - *pridávaním* prediktorov (forward selection) - Začína sa základným modelom bez prediktorov, v každom kroku sa pridá ten prediktor, ktorý prispel najviac (pridaním do modelu z predošlého kroku sa dosiahlo najmenšie RSS) a procedúra sa končí vtedy, keď už žiaden nový príspevok nie je štatisticky významný.
  - *uberaním* (backward selection) - Postup je opačný, z plného modelu sa postupne odstraňujú premenné, ktoré najmenej vysvetľujú variabilitu odozvy.
  - *kombináciou* (mixed selection) - Odstraňuje niektoré nevýhody predošlých postupov.

Príklad (reklama)

```
fit1 <- step(fit, direction = "backward")
```

```
Start:  AIC=212.79
sales ~ TV + radio + newspaper

            Df Sum of Sq    RSS    AIC
- newspaper  1      0.09  556.9 210.82
```

```
<none>                     556.8 212.79
- radio      1   1361.74 1918.6 458.20
- TV         1   3058.01 3614.8 584.90

Step:  AIC=210.82
sales ~ TV + radio

        Df Sum of Sq    RSS    AIC
<none>                556.9 210.82
- radio  1    1545.6 2102.5 474.52
- TV     1    3061.6 3618.5 583.10
```

- Model s reklamou v TV a rádiu má najnižšie AIC.

2.3.2.3 Miera zhody s dátami

- Na vyjadrenie kvality odhadu modelu sa najčastejšie používajú miery ako reziduálny rozptyl a koeficient determinácie.
- V prípade jednoduchej regresie je R^2 zhodné so štvorcom korelácie odozvy a prediktoru, teda $cor^2(X, Y)$.
- Aj vo viacnásobnej regresii súvisí s koreláciou, pretože vo všeobecnosti $R^2 = cor^2(Y, \hat{Y})$.
- Zatiaľ čo R^2 vždy rastie s počtom parametrov p, reziduálny rozptyl klesá s p len, ak príspevok prediktoru nie je príliš malý.
- Okrem číselných mier je dobré zhodu modelu s dátami posúdiť pomocou grafických súhrnov, pretože pomocou nich dokážeme odhaliť slabé miesta modelu. Napr. neuváženie synergie prediktorov a špecifické odchýlky od linearity.

2.3.2.4 Predikcie

- Predpovede z viacnásobného lineárneho regresného modelu sú zaťažené 3 druhmi neistoty:
  1. Nepresnosť odhadu parametrov súvisí s redukovateľnou chybou (z minulej prednášky). *Intervalom spoľahlivosti* (confidence interval) určíme, ako blízko bude $\hat{Y}$ ku $f(X)$.
  2. Voľba lineárnej aproximácie $f(X)$ je ďalší zdroj redukovateľnej chyby (model bias).
  3. Ak by sme aj poznali $f(X)$, i tak sa odozva nedá predpovedať presne - kvôli neredukovateľnej chybe. *Predpovedným intervalom* (prediction interval) určíme, ako blízko Y kolíše okolo $\hat{Y}$. Vždy je širší ako konfidenčný, pretože obsahuje obe chyby - neredukovateľnú chybu aj neistotu určenia $f(X)$.

Príklad (reklama)

- Predpokladajme výdavky 100,000 USD na reklamu v TV a 20,000 USD v rádiu.

```
sapply(c("confidence", "prediction"),
       function(x) predict(fit1,
                           newdata = data.frame(TV = 100, radio = 20),
                           interval = x,
                           level = 0.95),
       simplify = FALSE
) |>
  lapply(round, digits = 2) |>
  do.call(rbind.data.frame, args = _)
```

```
             fit   lwr   upr
confidence 11.26 10.99 11.53
prediction 11.26  7.93 14.58
```

- Odbyt by bol v priemere 11,260 jednotiek.
- Skutočná stredná hodnota $E(Y|TV = 100, Radio = 20)$ sa s 95% pravdepodobnosťou nachádza niekde medzi 10,990 a 11,530. Interval vyjadruje neistotu sprevádzajúcu priemer cez veľké množstvo trhov (miest).
- Odbyt v jednom konkrétnom meste by sa pohyboval *pravdepodobne* niekde v rozmedzí 7,930 až 14,580.

2.4 Kvalitatívny prediktor

- Prediktory nemusia byť iba kvantitatívne.
- Regresný model však ako vstupy vyžaduje iba číselné hodnoty, preto je nutné kvalitatívne premenné vhodne nahradiť.
- Situácia je jednoduchá v prípade binárnej premennej. Jej hodnoty sa iba prekódujú na 0 a 1. (V prostredí strojového učenia sa to nazýva one-hot encoding.)
- Vo všeobecnosti, náhodnú premennú X s m možnými hodnotami $\{a_1, \dots, a_m\}$ treba nahradiť $(m-1)$ pomocnými, *indikačnými premennými* (označovaných aj ako *dummy*), napr. pomocou indikačnej funkcie

$$1_x(X) = \begin{cases} 1 & \text{ak } X = x \\ 0 & \text{inak} \end{cases}$$

Ak hodnotu x_1 nastavíme ako základnú (referenčnú, baseline), regresný vzťah má potom tvar

$$Y = \beta_1 + \beta_2 1_{a_2}(X) + ... + \beta_m 1_{a_m}(X) + \varepsilon$$
$$= \begin{cases} \beta_1 + \varepsilon & \text{pre} \quad X = a_1 \\ \beta_1 + \beta_2 + \varepsilon & \quad X = a_2 \\ \vdots & \\ \beta_1 + \beta_m + \varepsilon & \quad X = a_m \end{cases}$$

- Interpretácia parametrov je tak priamočiara:
  - $\beta_1 = E(Y|X = a_1)$, čiže absolútny člen β_1 predstavuje strednú hodnotu Y pri prvej úrovni X,
  - $\beta_j = E(Y|X = a_j) - E(Y|X = a_1)$, $j \geq 2$, teda rozdiel v strednej hodnote Y oproti prvej úrovni X.
- Možností kódovania je veľa. Nemajú vplyv na odhad modelu ani test významnosti, iba na interpretáciu parametrov. Slúžia na ľahšie vyhodnotenie konkrétnych rozdielov (contrasts), špecifických pre potreby danej praktickej aplikácie.

Príklad (kreditky)

- Majme údaje z datasetu *Credit* o držiteľoch kreditných kariet:
  - odozvu *priemerný dlh* (balance),
  - niekoľko kvantitatívnych prediktorov ako vek, počet kariet (cards), počet rokov vzdelania, príjem, limit, úverové hodnotenie (rating),
  - a kvalitatívnych prediktorov ako vlastníctvo domu (own), či študuje (student), životný stav (status, slobodný atď.) a región (východ,západ,juh).
- *Vzťah výšky dlhu a toho, či klient vlastní dom*, sa z dát neukazuje významný. Test významnosti parametra je totožný s dvojvýberovým t-testom (pri rovnakom rozptyle).

```
lm(Balance ~ Own, ISLR2::Credit) |>
  summary() |>
  coef() |>  round(3)
```

```
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  509.803     33.128  15.389    0.000
OwnYes        19.733     46.051   0.429    0.669
```

```
t.test(Balance ~ Own, ISLR2::Credit)
```

```
    Welch Two Sample t-test

data:  Balance by Own
t = -0.42838, df = 395.58, p-value = 0.6686
alternative hypothesis: true difference in means between group No and group Yes is not equa
95 percent confidence interval:
 -110.29408   70.82784
sample estimates:
 mean in group No mean in group Yes
         509.8031          529.5362
```

- Test vzťahu dlhu a regiónu je totožný s analýzou rozptylu (jednofaktorová ANOVA pri homogénnom rozptyle). Opäť sa vzťah nepotvrdil.

```
lm(Balance ~ Region, ISLR2::Credit) |>
  summary()
```

```

Call:
lm(formula = Balance ~ Region, data = ISLR2::Credit)

Residuals:
    Min      1Q  Median      3Q     Max
-531.00 -457.08  -63.25  339.25 1480.50

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   531.00      46.32  11.464   <2e-16 ***
RegionSouth   -12.50      56.68  -0.221    0.826
RegionWest    -18.69      65.02  -0.287    0.774
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 460.9 on 397 degrees of freedom
Multiple R-squared:  0.0002188, Adjusted R-squared:  -0.004818
F-statistic: 0.04344 on 2 and 397 DF,  p-value: 0.9575
```

```
oneway.test(Balance ~ Region, ISLR2::Credit, var.equal = TRUE)
```

```
    One-way analysis of means
```

```
data:  Balance and Region
F = 0.043443, num df = 2, denom df = 397, p-value = 0.9575
```

- Funkcia *lm* použila štandardné kódovanie kvalitatívnej premennej, ktoré sa dá zistiť pomocou `options("contrasts")` a zobraziť funkciou *contrasts*: vľavo sú pôvodné hodnoty X, hore sú členy použité v regresnej rovnici.

```
contrasts(ISLR2::Credit$Region)
```

```
      South West
East      0    0
South     1    0
West      0    1
```

2.5 Aditivita a linearita

- Štandardná viacnásobná lineárna regresia (2.2) má minimálne dva, veľmi obmedzujúce predpoklady:
  - aditivita – vzťah medzi odozvou a prediktorom X_j nezávisí od hodnôt ostatných prediktorov,
  - linearita – zmena Y spojená s jednotkovou zmenou X_j je konštantná.

2.5.1 Uvoľnenie aditivity

- Jedným zo spôsobov, ako odstrániť predpoklad o nezávislosti vplyvu jedného prediktoru od vplyvu druhého prediktoru, je pridať do rovnice tretí prediktor, tzv. interakčný člen, ktorý je iba násobkom prvých dvoch, takže model má tvar

$$\begin{aligned} Y &= \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_1 X_2 + \varepsilon \\ &= \beta_0 + \underbrace{(\beta_1 + \beta_3 X_2)}_{\tilde{\beta}_1} X_1 + \beta_2 X_2 + \varepsilon \end{aligned}$$

- Parameter $\tilde{\beta}_1$ je zjavne lineárnou funkciou X_2, čiže vzťah medzi X_1 a Y sa mení s premennou X_2 (rovnako aj v obrátenom poradí prediktorov).

Ilustrácia (výrobné linky)

- Skúmajme produktivitu v továrni. Chceme predpovedať objem produkcie výrobkov pomocou počtu výrobných liniek a celkového počtu pracovníkov.

- Prirodzene očakávame, že efekt pridávania liniek závisí od dostupnosti ich obsluhy (žiaden pracovník rovná sa žiadna výroba).
- Ak by model mal tvar

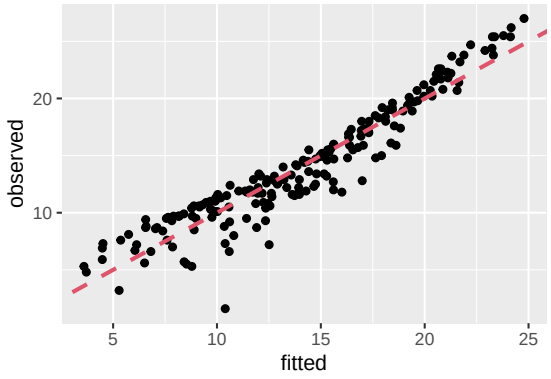
$$Vroba \approx 1.2 + 3.4 \cdot Linky + 0.22 \cdot Pracovnci + 1.4 \cdot (Linky \cdot Pracovnci),$$

potom by pridanie jednej linky zvýšilo produkciu o $3.4 + 1.4\times$ počet pracovníkov.
- Samozrejme takýto model nerieši situáciu, keď si pracovníci pre veľký počet začnú navzájom zavadzať.

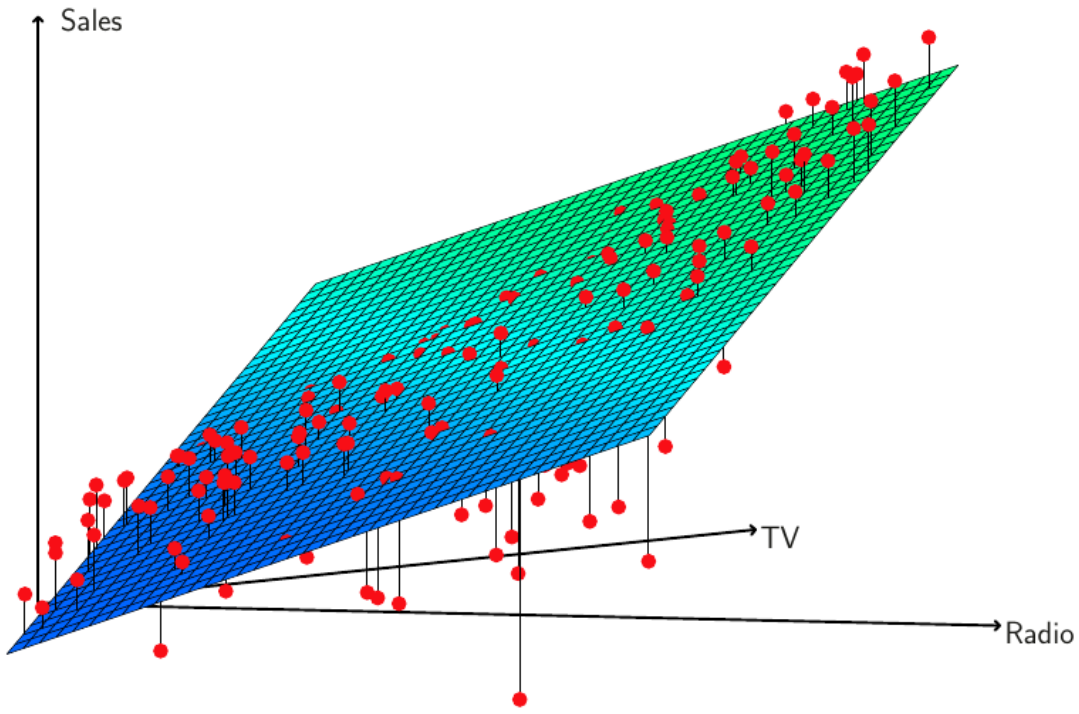
Príklad (reklama)

- Hoci koeficient determinácie pre model s TV a rádiom je pomerne optimistický ($R^2 = 0.897$), pri pohľade na bodový graf vzťahu odhadnutých a pozorovaných hodnôt (obrázok vľavo) cítime, že niečo škrípe.

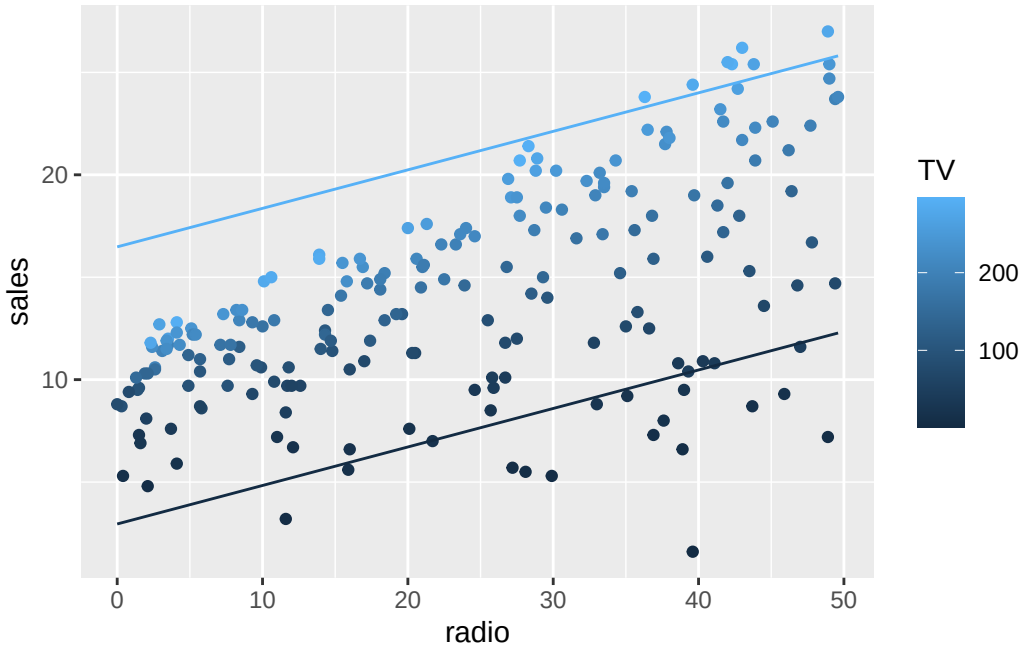
```
dat_Advertising |>
  dplyr::select(-newspaper) |>
  dplyr::mutate(fitted = predict(fit1)) |>
  dplyr::rename(observed = sales) |>
  ggplot() + aes(x = fitted, y = observed) +
  geom_point() +
  geom_abline(slope = 1, color = 2, linetype = 2, linewidth = 1)
```



- Pre lepšiu predstavu treba model zobraziť v priestore všetkých zúčastnených premenných.




```
grid_Advertising <- expand.grid(
  TV = range(dat_Advertising$TV),
  radio = range(dat_Advertising$radio)
) |>
  modelr::add_predictions(fit1, var = "sales")
ggplot() + aes(x = radio, y = sales, color = TV) +
  geom_point(data = dat_Advertising) +
  geom_line(aes(group = as.factor(TV)), data = grid_Advertising)
```



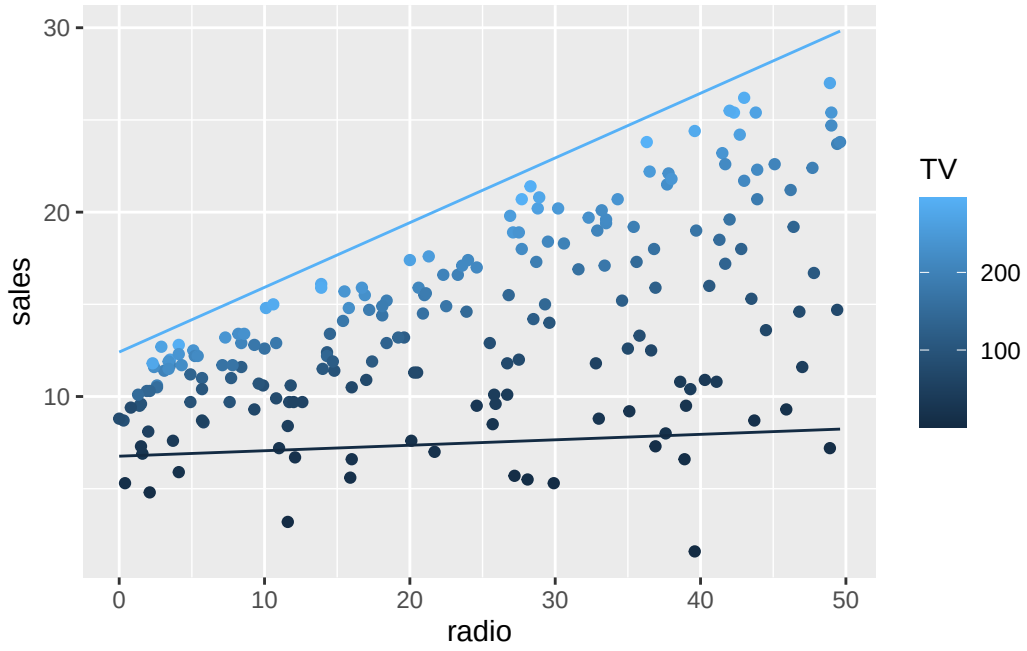
- Aditívny Lineárny model má tendenciu podhodnocovať v oblasti blízko diagonály (v rovine na obr. vľavo), tam kde je rozpočet na reklamu v oboch médiách v rovnováhe. Naopak, odbyt je nadhodnocovaný v prípadoch nesymetrických investícií do reklamy.
- Odhadnime teda model s interakciou medzi TV a rádiom.

```
fit2 <- update(fit1, . ~ . + TV:radio, data = dat_Advertising)
summary(fit2)
```

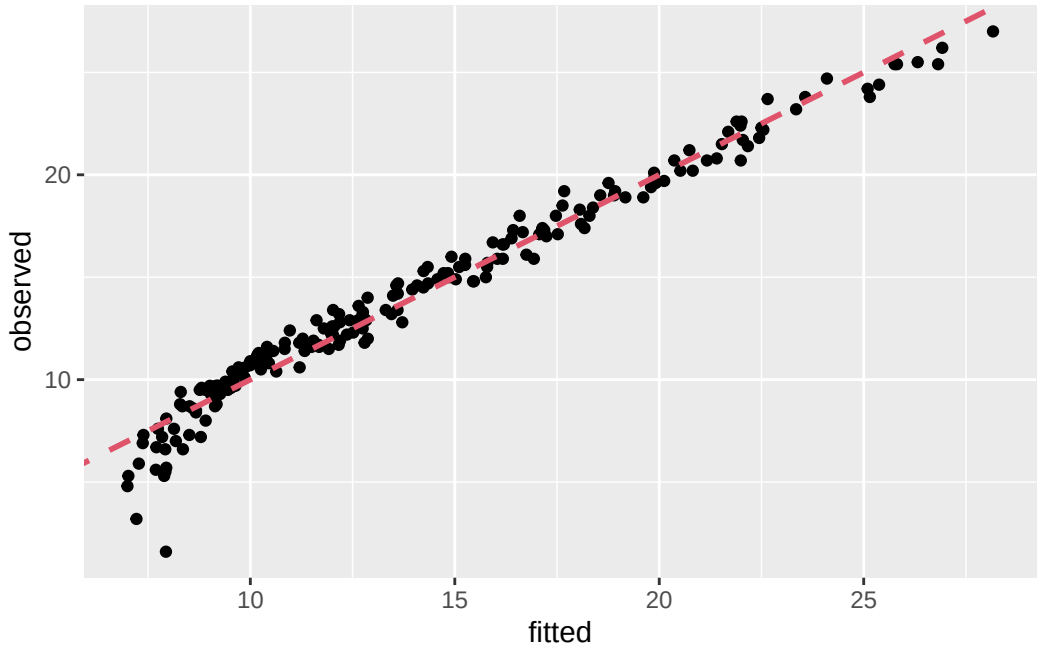
```
Call:
lm(formula = sales ~ TV + radio + TV:radio, data = dat_Advertising)

Residuals:
    Min      1Q  Median      3Q     Max
-6.3366 -0.4028  0.1831  0.5948  1.5246

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 6.750e+00  2.479e-01  27.233   <2e-16 ***
TV          1.910e-02  1.504e-03  12.699   <2e-16 ***
radio       2.886e-02  8.905e-03   3.241   0.0014 **
```

```
TV:radio    1.086e-03  5.242e-05  20.727   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.9435 on 196 degrees of freedom
Multiple R-squared:  0.9678,    Adjusted R-squared:  0.9673
F-statistic:  1963 on 3 and 196 DF,  p-value: < 2.2e-16
```

```r
grid_Advertising <- modelr::add_predictions(
  data = grid_Advertising,
  model = fit2, var = "sales")
ggplot() + aes(x = radio, y = sales, color = TV) +
  geom_point(data = dat_Advertising) +
  geom_line(aes(group = as.factor(TV)), data = grid_Advertising)
```




```r
dat_Advertising |>
  dplyr::select(-newspaper) |>
  dplyr::mutate(fitted = predict(fit2)) |>
  dplyr::rename(observed = sales) |>
  ggplot() + aes(x = fitted, y = observed) +
  geom_point() +
  geom_abline(slope = 1, color = 2, linetype = 2, linewidth = 1)
```



- Nový model je podstatne lepší, interakčný člen pomohol vysvetliť $(96.8 - 89.7)/(100 - 89.7) = 69\%$ zvyšnej variability odbytu. Rezervu má v oblasti nízkeho odbytu, avšak môže ísť aj o prítomnosť extrémne nízkych pozorovaní. Tzv. *outlier*-om sa ešte budeme venovať.
- Ak navýšime rozpočet na reklamu v rádiu o 1,000 USD, odbyt by mal stúpnuť o $29 + 1.1 \cdot TV$ jednotiek. Čiže napr. o 139,000 ak $TV = 100,000$ USD.

- V poslednom príklade vyšiel efekt interakčného člena aj oboch premenných signifikantný. Avšak nie vždy to tak býva a jeden (alebo aj oba) individuálne členy môžu vyjsť nevýznamné. V takom prípade sa – podľa hierarchického princípu – v modeli ponechávajú *aj samostatne* všetky premenné, ktoré vystupujú v interakčnom člene. (Inak by sa interakčný člen ťažko interpretoval.)
- Interakcia (synergia) sa neobmedzuje len na spojité prediktory. Nech X_1 je kvantitatívna a X_2 kvalitatívna premenná, ktorá náhodne nadobúda jednu z m hodnôt $\{a_1, \ldots, a_m\}$.

Potom model viažuci obe premenné aj ich interakciu má tvar

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 1_{a_2}(X_2) + ... + \beta_m 1_{a_m}(X_2) + \\ + \beta_{m+1} X_1 1_{a_2}(X_2) + ... + \beta_{2m-1} X_1 1_{a_m}(X_2) + \varepsilon$$

$$= \begin{cases} \beta_0 + \beta_1 X_1 + \varepsilon & \text{pre} \quad X_2 = a_1 \\ (\beta_0 + \beta_2) + (\beta_1 + \beta_{m+1}) X_1 + \varepsilon & \quad X_2 = a_2 \\ \vdots & \\ (\beta_0 + \beta_m) + (\beta_1 + \beta_{2m-1}) X_1 + \varepsilon & \quad X_2 = a_m \end{cases}$$

- Pre každú hodnotu premennej X_2 tvorí model jednu regresnú priamku s odlišným priesečníkom i sklonom. Tento rozdiel je daný prírastkom $\beta_2, \dots, \beta_m$ resp. $\beta_{m+1}, \dots, \beta_{2m-1}$ oproti parametrom pre prvú, referenčnú úroveň prediktora X_2.

Príklad (kreditky)

- Chceme predpovedať veľkosť dlhu držiteľov kreditných kariet pomocou dvoch prediktorov: mesačného príjmu (income) a indikátoru štúdia.
- Vytvoríme dva modely – jeden *bez* a druhý *s* interakciou – a porovnáme ich.

```
fit1 <- lm(Balance ~ Income + Student, data = ISLR2::Credit)
fit1 |> summary() |> coef() |> round(3)
```

```
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  211.143     32.457   6.505        0
Income         5.984      0.557  10.751        0
StudentYes   382.671     65.311   5.859        0
```

$$Dlh = 211 + 5.98 \cdot Prjem + 383 \cdot 1_{no}(tudent) + \varepsilon$$

$$= \begin{cases} 211 + 5.98 \cdot Prjem & \text{ak} \quad tudent = nie \\ 594 + 5.98 \cdot Prjem & \quad tudent = no \end{cases}$$

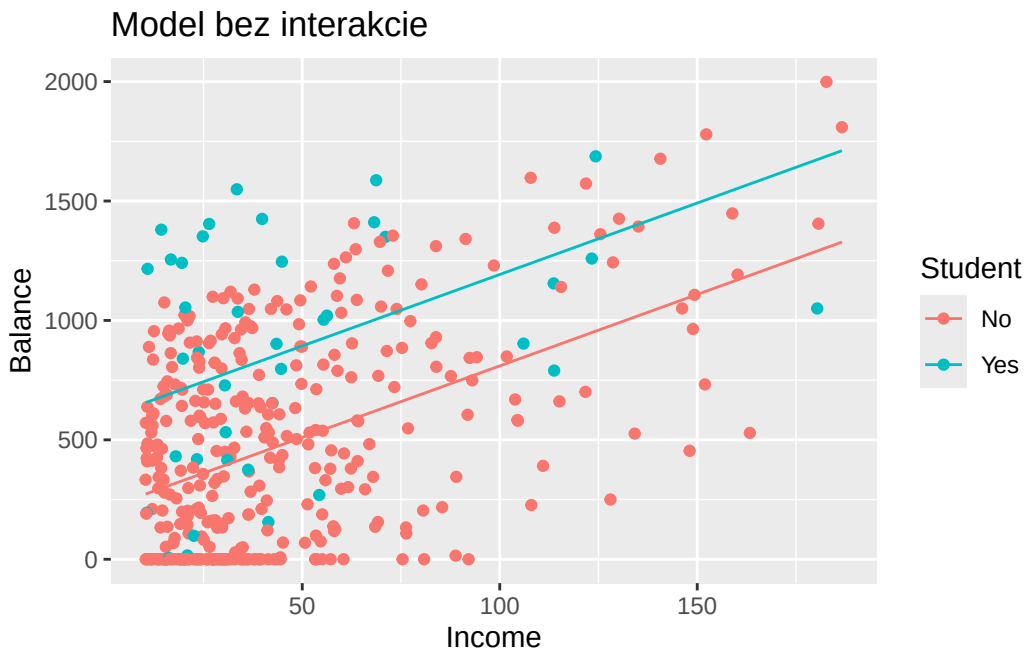
```
fit2 <- lm(Balance ~ Income * Student, data = ISLR2::Credit)
fit2 |> summary() |> coef() |> round(3)
```

```
                 Estimate Std. Error t value Pr(>|t|)
(Intercept)       200.623     33.698   5.953    0.000
Income              6.218      0.592  10.502    0.000
StudentYes        476.676    104.351   4.568    0.000
Income:StudentYes  -1.999      1.731  -1.155    0.249
```

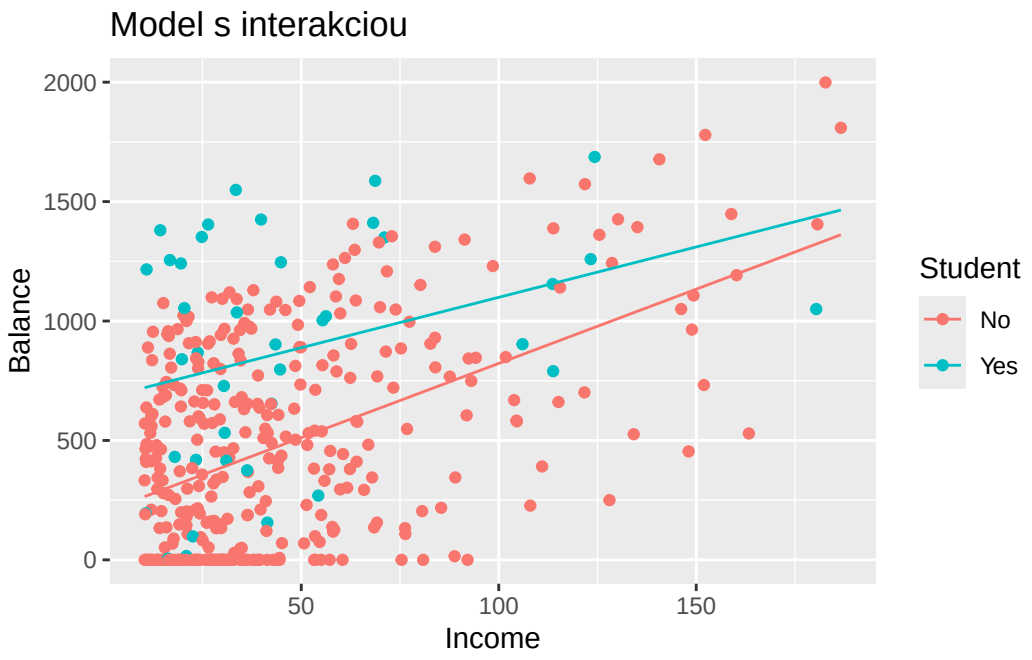
$$Dlh = 201 + 6.21 \cdot Prjem + 477 \cdot 1_{no}(tudent) - 2.00 \cdot Prjem \cdot 1_{no}(tudent) + \varepsilon$$

$$= \begin{cases} 201 + 6.21 \cdot Prjem & \text{ak} \quad tudent = nie \\ 678 + 4.21 \cdot Prjem & tudent = no \end{cases}$$

```
grid_Credit <- expand.grid(
  Income = range(ISLR2::Credit$Income),
  Student = levels(ISLR2::Credit$Student)    # alebo unique()
) |>
  modelr::add_predictions(fit1, var = "Balance")
ggplot() + aes(x = Income, y = Balance, color = Student) +
  geom_point(data = ISLR2::Credit) +
  geom_line(aes(group = as.factor(Student)), data = grid_Credit) +
  ggtitle("Model bez interakcie")
```




```
grid_Credit <- grid_Credit |>
  modelr::add_predictions(fit2, var = "Balance")
ggplot() + aes(x = Income, y = Balance, color = Student) +
  geom_point(data = ISLR2::Credit) +
  geom_line(aes(group = as.factor(Student)), data = grid_Credit) +
  ggtitle("Model s interakciou")
```



- Hoci druhý model neprispieva ku vysvetleniu variability dlhu významnejšie ako prvý model (interakčný člen je štatisticky nevýznamný), príklad dobre ilustruje, že s interakciou dochádza k zmene sklonu regresnej priamky.

2.5.2 Uvoľnenie linearity

- Lineárny regresný model (2.1) predpokladá linearitu medzi odozvou Y a prediktorom.
- To reálne nebýva často splnené.
- Aj v takých prípadoch sa však lineárny model dá použiť, a to *rozšírením pomocou bázových funkcií* b_j,

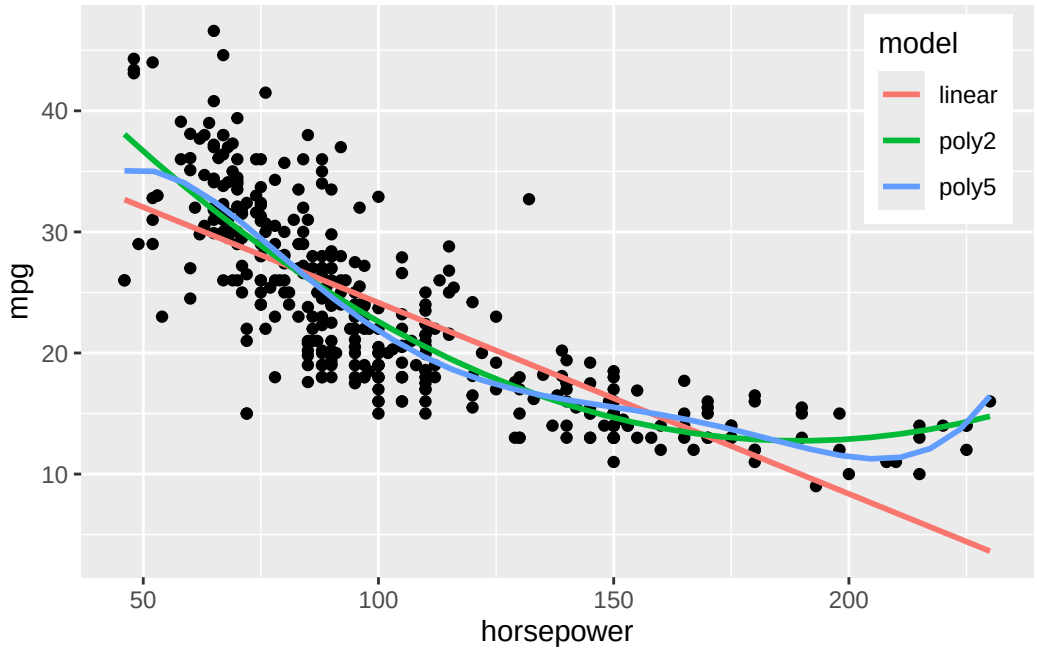
$$Y = \beta_0 + \beta_1 b_1(X) + \beta_2 b_2(X) + ... + \beta_q b_q(X) + \varepsilon$$

- Bázové funkcie sú zvolené vopred, takže model zostáva *lineárny v parametroch* a dá sa rovnako elegantne odhadnúť metódou najmenších štvorcov ako aj použiť nástroje štatistickej dedukcie o kvalite modelu, o jeho parametroch a predikcii. To, čo treba osobitne prispôsobiť rozšíreniu, je *interpretácia*.
- Ako jednoduché bázové funkcie sa často používajú mocninové transformácie $b_j(x) = x^j$ (výsledkom je tzv. polynomická regresia), ďalej odmocninové, logaritmické, po častiach lineárne či goniometrické transformácie.
- Nakoniec, aj indikačná funkcia, ktorou sme prekódovali kvalitatívnu premennú do $\{0, 1\}$, sa dá zaradiť medzi bázové funkcie.

Príklad (autá)

- Majme dataset pozorovaní o 392 automobiloch, ktorý obsahuje také vlastnosti ako dojazd (mpg, miles per gallon), počet valcov, zdvihový objem, výkon motora (horsepower), váha, zrýchlenie, pôvod, rok návrhu a názov modelu.
- Chceme modelovať vzťah medzi dojazdom a výkonom motora.
- Lineárna funkcia už napohľad nepostačuje, preto siahneme po polynomickej regresii s rôznym stupňom.

```
fit1 <- lm(mpg ~ horsepower, ISLR2::Auto)
fit2 <- lm(mpg ~ horsepower + I(horsepower^2), ISLR2::Auto)
fit3 <- lm(mpg ~ poly(horsepower, 5), ISLR2::Auto)

grid_Auto <- expand.grid(horsepower = modelr::seq_range(ISLR2::Auto$horsepower,
                                                        n = 30)
                         ) |>
  modelr::gather_predictions(linear = fit1, poly2 = fit2, poly5 = fit3,
                             .pred = "mpg")
ggplot() + aes(x = horsepower, y = mpg) +
  geom_point(data = ISLR2::Auto) +
  geom_line(aes(color = model), data = grid_Auto, linewidth = 1) +
  theme(legend.position = "inside", legend.position.inside = c(0.9,0.8))
```



- S rastúcim stupňom polynómu klesá vychýlenosť a rastie rozptyl modelu (zložka redukovateľej chyby). Optimálny stupeň by sa určil pomocou testovacej chyby, avšak z interpretačného hladiska nie je ani polynomický model ideálny – hlavne kvôli zmene monotónnosti. Je nelogické, aby s rastúcim výkonom (nad 180 koní) rástol aj dojazd.

- Ako dočasnú náhradu za vyhodnotenie všetkých troch modelov pomocou odhadu testovacej chyby teraz použijeme F-test – sériovo (čiže prvý oproti druhému a druhý voči tretiemu).

```
anova(fit1, fit2, fit3)
```

```
Analysis of Variance Table

Model 1: mpg ~ horsepower
Model 2: mpg ~ horsepower + I(horsepower^2)
Model 3: mpg ~ poly(horsepower, 5)
  Res.Df    RSS Df Sum of Sq        F    Pr(>F)
1    390 9385.9
2    389 7442.0  1   1943.89 103.8767 < 2.2e-16 ***
3    386 7223.4  3    218.66   3.8949  0.009196 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

- Z popisného hľadiska polynomická funkcia nielen druhého stupňa (oproti lineárnej) vystihuje dáta ale aj piateho stupňa (oproti kvadratickej).

- Rozšírenie tohoto prístupu na viacnásobnú regresiu s prediktormi $\vec{X} = (X_1, X_2, \dots, X_p)$ je tiež pomerne priamočiare,

$$Y = \beta_0 + \beta_1 b_1(\vec{X}) + \beta_2 b_2(\vec{X}) + \dots + \beta_q b_q(\vec{X}) + \varepsilon$$

  a zahŕňa aj lineárny model s interakciou, kedy (napr. pre $p = 2$) bude $b_1(\vec{X}) = X_1$, $b_2(\vec{X}) = X_2$ a $b_3(\vec{X}) = X_1 X_2$.

2.6 Nesplnené predpoklady

- Žiaden model nie je univerzálny – každý je spoľahlivý iba vtedy, ak sú splnené určité podmienky.
- Ak nie sú splnené predpoklady použitia lineárnych regresných modelov,

  - nie sú spoľahlivé ani závery dedukcie urobené pomocou (takto) odhadnutého modelu a
  - presnosť predikcií výrazne klesá.

- Hlavné príčiny problémov sú:

  1. nelinearita vzťahu odozvy a prediktorov
  2. korelácia v šumovej zložke
  3. premenlivý rozptyl šumovej zložky
  4. odľahlé pozorovania
  5. pozorovania s veľkým pákovým efektom
  6. silný vzťah medzi prediktormi (tzv. kolinearita)

- Identifikácia a prekonanie týchto problémov niekdy hraničí až s umením.
- Preto je dobré poznať niekoľko užitočných diagnostických nástrojov.

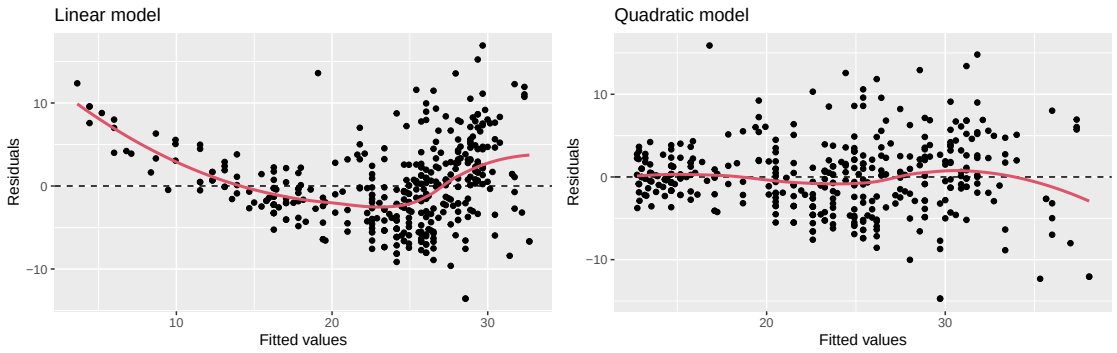
2.6.1 Nelinearita

- Jedným z diagnostických nástrojov je graf rezíduí (voči predikciám) modelu. Pomocou neho sa dá odhaliť aj porušenie predpokladu linearity regresnej funkcie.

Príklad (autá)

- Reziduálne grafy (prvých dvoch modelov z predošlého príkladu) odhaľujú jednak
  - výraznu zmenu v strednej hodnote rezíduí lineárneho modelu, jednak
  - konštantnosť strednej hodnoty rezíduí kvadratického modelu.
- To ukazuje na vychýlenosť (bias) lineárneho modelu.

```r
fit1 |>
  broom::augment() |>
  ggplot() + aes(x = .fitted, y = .resid) +
  geom_point() +
  geom_hline(yintercept = 0, linetype = 2) +
  geom_smooth(se = FALSE, color = 2) +
  labs(title = "Linear model", x = "Fitted values", y = "Residuals")
fit2 |>
  broom::augment() |>
  ggplot() + aes(x = .fitted, y = .resid) +
  geom_point() +
  geom_hline(yintercept = 0, linetype = 2) +
  geom_smooth(se = FALSE, color = 2) +
  labs(title = "Quadratic model", x = "Fitted values", y = "Residuals")
# alternatívely e.g.
# fit1 |> ggfortify:::autoplot.lm(which = 1, ncol = 1)
# or tools of the packages ggResidpanel, gglm ...
```



- V tomto prípade bol problém vyriešený mocninovou transformáciou prediktoru (rozšírením modelu pomocou polynomickej bázovej funkcie).

2.6.2 Korelácia šumovej zložky

- Lineárna regresia predpokladá, že hodnoty chybovej zložky majú rovnaké rozdelenie a navzájom nijak nesúvisia, teda napr. že realizácia ε_i neposkytuje žiadnu informáciu o tom, akú hodnotu nadobudne ε_{i+1}.
- Ak to nie je splnené, konkrétne ak je prítomná pozitívna sériová korelácia, potom odhady štandardných chýb parametrov sú *podhodnotené*, čo vyústi do

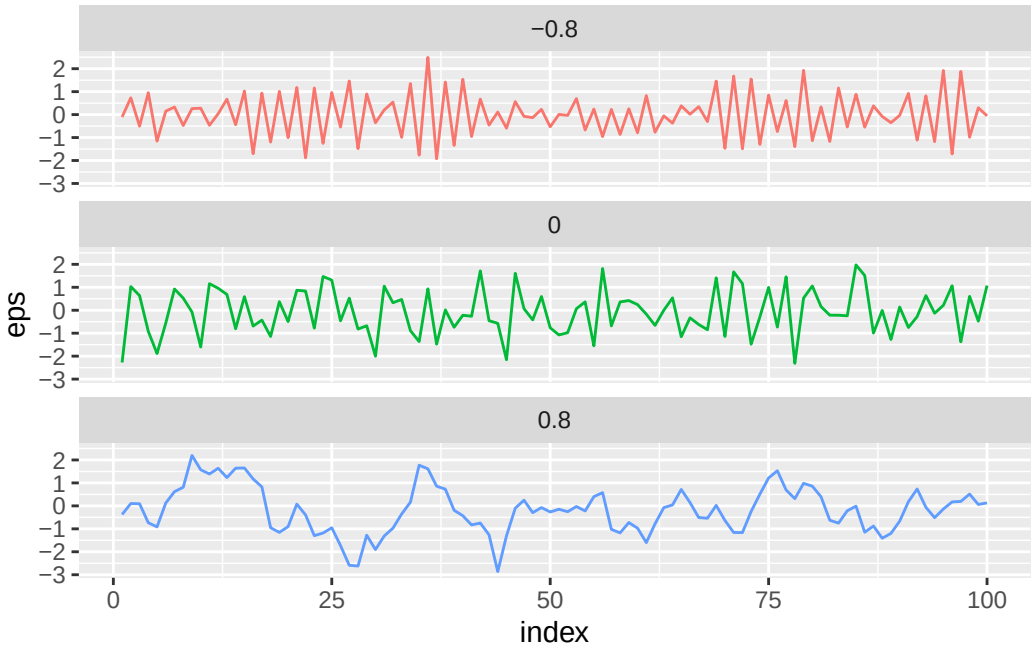
  - užších konfidenčných a predikčných intervalov (napr. 95% interval bude v skutočnosti zodpovedať menšej pravdepodobnosti než 0.95, že obsahuje skutočnú hodnotu parametra),
  - menších p-hodnôt testu významnosti modelu, takže sa môžeme mylne domnievať, že regresný vzťah je významný.

- Ak sú teda chyby kladne autokorelované, vedie nás to k prehnanej dôvere ku modelu.
- Korelovanosť chýb často vzniká v kontexte *časových radov*, teda pozorovaní získaných v časových okamihoch nasledujúcich "dostatočne krátko" po sebe.

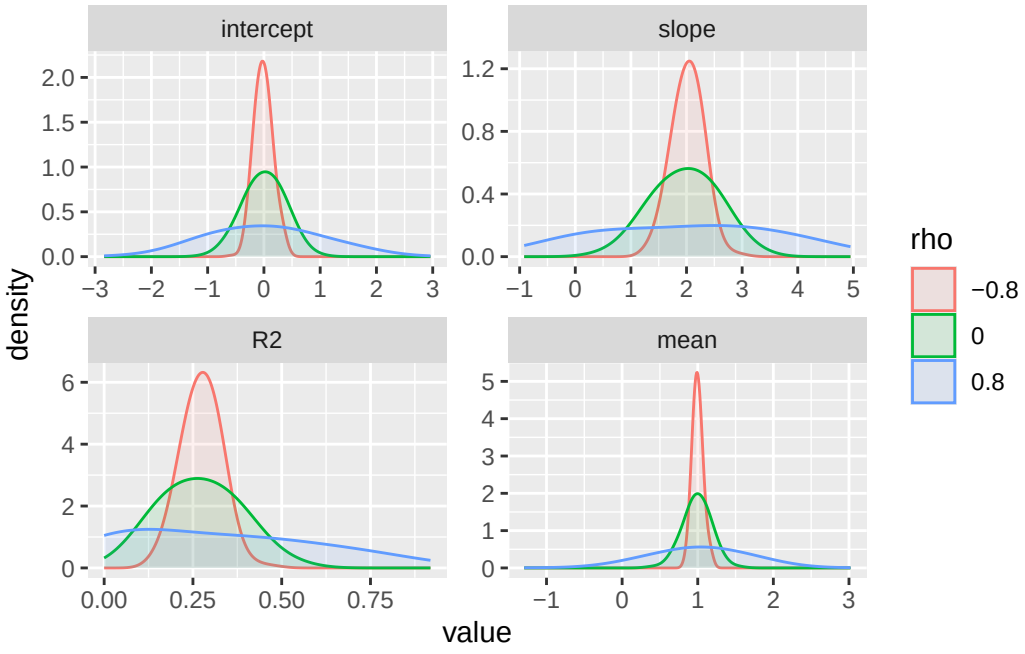
Ilustrácia (autokorelácia)

- Grafy znázorňujú šumovú zložku generovanú AR(1) procesom pre tri rôzne hodnoty parametra a jej vplyv na presnosť určenia niekoľkých štatistík súvisiacich s modelom. Skutočný vzťah je daný rovnicou $Y = 0 + 2X + \varepsilon$, pričom $\sigma_\varepsilon = 1$ a predpoveď (mean) bola počítaná v bode $X = 0.5$.

```
set.seed(4)
lapply(c(-0.8, 0, 0.8), FUN = function(x) {
  n <- 100
  arima.sim(list(ar = x), n = n) |>
    suppressWarnings() |>
    cbind.data.frame(index = 1:n, eps = _, rho = x) |>
    transform(eps = eps/sd(eps))
}) |>
  dplyr::bind_rows() |>
  dplyr::mutate(rho = factor(rho)) |>
  ggplot() + aes(x = index, y = eps, color = rho) +
  geom_line(show.legend = FALSE) +
  facet_wrap(vars(rho), ncol = 1)
```


−0.8
0
0.8
eps
2
1
0
−1
−2
−3
0
25
50
75
100
index

```r
# Function to calculate sampling distribution of several estimates involved in simple linea
# in: xvalues - numeric vector, values of the predictor
#     xpredict - numeric value, point of conditional mean
#     linepars - numeric vector, true model parameters
#     rho - numeric value, parameter of AR(1) model
#     sd_noise - numeric value, standard deviation of noise series
#     N - integer, number of replications
# out: Nx3 data frame
fun_simplereg_sim <- function(xvalues = seq(0, 1, length.out = 30),
                              xpredict = median(xvalues),
                              linepars = c(b0 = 0, b1 = 2),
                              rho = 0, sd_noise = 1,
                              N = 300) {
  n <- length(xvalues)
  noise <- replicate(N, {
    out <- arima.sim(list(ar = rho), n = n) |>
      suppressWarnings()
    out/sd(out)*sd_noise
  })
  signal <- ( cbind(1,xvalues) %*% linepars ) |>
    rep(N) |>
    matrix(ncol = N)
  data <- signal + noise
  apply(data, 2, function(y) {
    fit <- lm(y ~ xvalues)
    pars <- coefficients(fit) |>  unname()
    mean <- predict(fit, newdata = data.frame(xvalues = xpredict)) |> unname()
    c(intercept = pars[1],
      slope = pars[2],
      R2 = cor(fit$fitted.values, y)^2,
      mean = mean
      )
  }) |> t() |> data.frame()
}

lapply(c(-0.8, 0, 0.8),
       FUN = function(x) {
         fun_simplereg_sim(rho = x) |>
           cbind(rho = x)
         }) |>
  dplyr::bind_rows() |>
  tidyr::pivot_longer(-rho, names_to = "variable", values_to = "value") |>
  dplyr::mutate(rho = factor(rho), variable = forcats::as_factor(variable)) |>
  ggplot() + aes(x = value, fill = rho, color = rho) +
  geom_density(adjust = 2, alpha = 0.1) +
  facet_wrap(vars(variable), ncol = 2, scales = "free")
```



- Výsledky pre sklon a R^2 sú veľmi citlivé na rozdelenie prediktoru. V tomto príklade bola pre hodnoty X použitá rovnomerná mriežka 30 bodov v intervale $[0, 1]$.

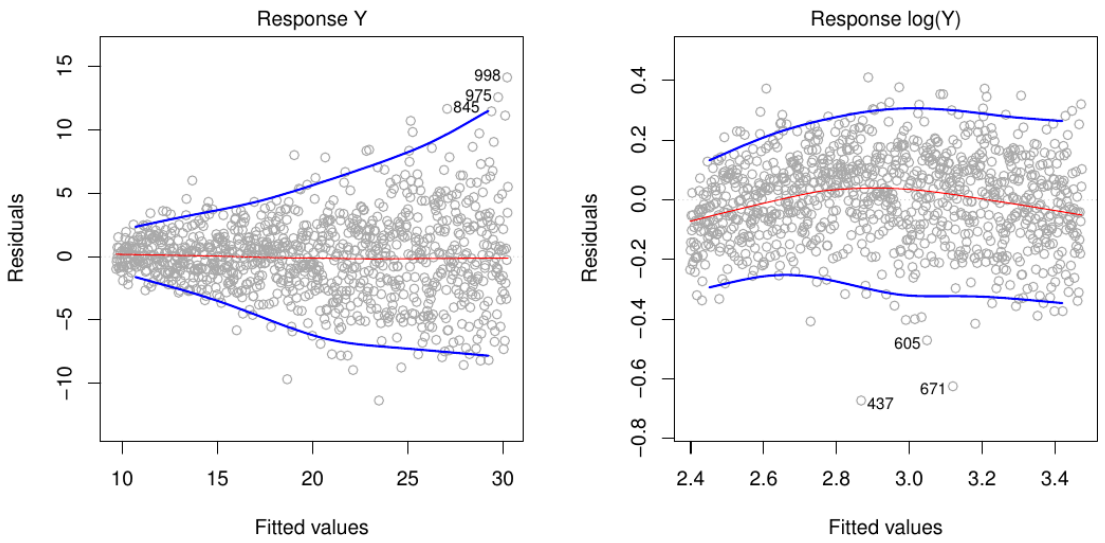
- Na riešenie problému takéhoto druhu (keď model nevyužíva informáciu dôležitého prediktoru – času) slúžia metódy analýzy časových radov.
- Korelácia šumu však môže vzniknúť aj v dôsledku vynechania kvalitatívnej premennej ako významného faktoru vplývajúceho na odozvu. Napríklad ak v analýze výšky jednotlivcov pomocou ich váhy zanedbáme fakt, že niektorí sú členmi jednej rodiny, dodržiavajú tú istú diétu, alebo boli vystavení rovnakým vplyvom prostredia.
- Ak sa takýmto vplyvom dá vyhnúť, zvyčajne sa problém vyrieši dobrým *návrhom experimentu* (design of experiment). Ak nie, a štatistické jednotky prirodzene tvoria skupiny, ktorých vplyv však nie je predmetom záujmu výskumu, vtedy sa na modelovanie používajú tzv. *linear mixed-effects* modely.

2.6.3 Premenlivá variabilita

- Narušenie podmienky konštantnosti rozptylu šumovej zložky, $var(\varepsilon_i) = \sigma^2$, ktoré sa nazýva *heteroskedasticita*, má vplyv na štandardné chyby odhadov a tak aj na testy hypotéz, ktoré s modelom súvisia.
- Jedným z príkladov je rozptyl rastúci so strednou hodnotou, kedy rezíduá majú tvar lievika. Riešením môže byť vhodná konkávna transformácia odozvy, napr. odmocninová,

logaritmická alebo ich zovšeobecnenie – Box-Coxova transformácia.

Ilustrácia (heteroskedasticita)



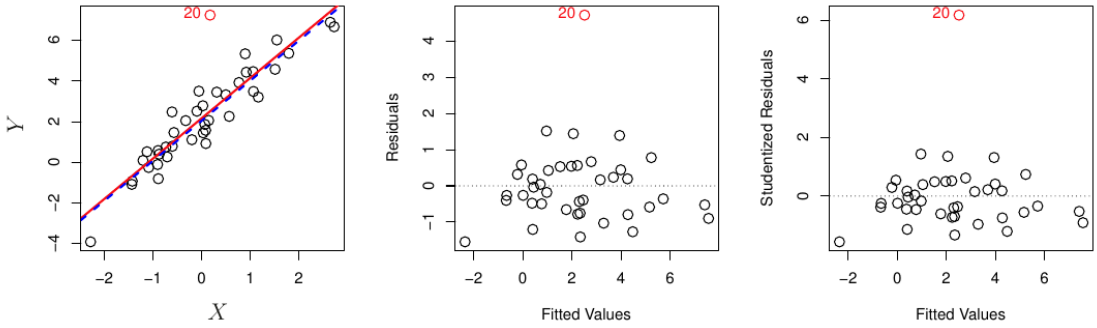
- Iným zdrojom heteroskedasticity býva agregovanie pozorovaní. Napr. nech i-ta hodnota odozvy ($i = 1, \ldots, n$) vznikne priemerom n_i surových nekorelovaných pozorovaní s rozptylom σ^2. Jej rozptyl tak bude $\sigma_i^2 = \sigma^2/n_i$.
- V tomto prípade je pomoc tiež jednoduchá – odhadnúť model metódou vážených najmenších štvorcov (*weighted least squares*) s váhami úmernými prevrátenej hodnote rozptylov, tu konkrétne $w_i = n_i$.

2.6.4 Odľahlé pozorovania

- Odľahlé pozorovanie (outlier) je bod v ktorom je *hodnota odozvy* nezvyčajne ďaleko od predpovede modelu.
- Vznikajú z rôznych dôvodov, napr. omylom pri zázname merania.
- Odľahlé pozorovania nemusia mať výrazný vplyv na odhad parametrov modelu (ak nevytvárajú pákový efekt). Nezvyknú mať ani netypické hodnoty prediktorov. Avšak spôsobujú iné problémy, napr. zvýšením RSS dôjde k rozšíreniu konfidenčných intervalov, zvýšeniu p-hodnoty testov významnosti či zníženiu R^2.
- Identifikovať sa dajú z grafu rezíduí, ale ešte lepšie z rezíduí normovaných vlastnou štandardnou odchýlkou (studentized residuals), pretože potom stačí podozrivú hodnotu porovnať s kvantilom t-rozdelenia zodpovedajúcim nejakej vysokej pravdepodobnosti, najčastejšie sa nahrubo porovná s hodnotou 3.

- Ak outlier považujeme za omyl pri zázname, je vhodné ho jednoducho zo súboru pozorovaní vylúčiť. Avšak pozor, môže indikovať aj slabinu modelu, ako napr. chýbajúci prediktor.

Ilustrácia (outlier)



2.6.5 Pákový efekt

- Oproti pozorovaniam s odľahlou hodnotou odozvy z predošlej časti, pozorovania s veľkým pákovým vplyvom (*leverage* effect) na odhad modelu sú tie, ktoré majú *odľahlé hodnoty prediktorov*.
- V jednoduchej regresii je identifikácia takých bodov jednoduchá (v nasledujúcej ilustrácii na obrázku vľavo).
- Vo viacnásobnej regresii sa však nemusia prejaviť nezvyčajnou hodnotou jednotlivých prediktorov (obr. v strede) a vo vyšších rozmeroch na odhalenie nestačia ani grafické nástroje.
- Preto je užitočné vypočítať *mieru pákového efektu* (leverage statistics, obr. vpravo). Vo všeobecnosti je definovaná ako miera vplyvu odozvy i-teho pozorovania na predikciu,

$$h_i = \frac{\partial \hat{y}_i}{\partial y_i} \tag{2.3}$$

- V kontexte jednoduchej regresie sa vypočíta zo vzťahu

$$h_i = \frac{1}{n} + \frac{(x_i - \bar{x})^2}{\sum_{k=1}^{n}(x_k - \bar{x})^2} \ ,$$

  z korého vidieť, že $h_i \in [\frac{1}{n}, 1]$ a závisí od vzdialenosti hodnoty prediktora od jeho priemeru.
- Pri viacnásobnej regresii pákovú štatistiku vyjadríme ako diagonálny prvok ortoprojekčnej matice $\mathbf{H}$ (*hat matrix*). Ak na lineárny regresný model pozeráme ako na

špeciálny prípad lineárneho vyhladzovania (čiže váženého priemeru, kde váhy závisia od prediktorov),

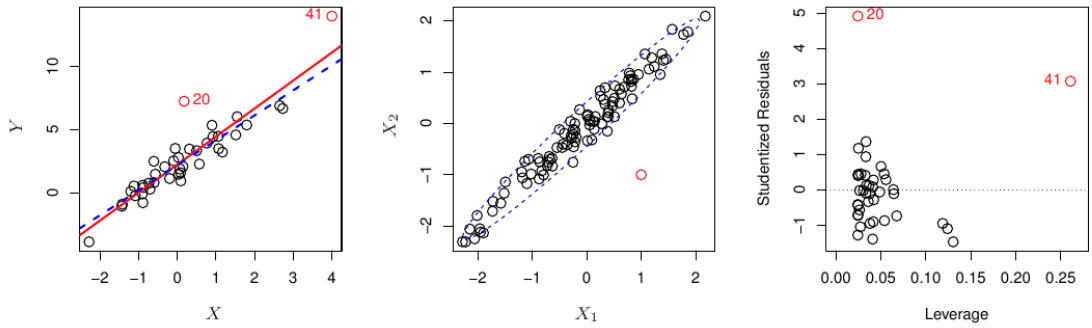
$$\hat{\mathbf{y}} = \mathbf{X}\hat{\beta} = \underbrace{\mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'}_{\mathbf{H}}\ \mathbf{y}$$

potom práve $\mathbf{H}$ je tá matica, ktorá obsahuje váhy. Takže znovu, vezmúc do úvahy všeobecnú definíciu (2.3), potom

$$h_i = \{\mathbf{H}\}_{ii}$$

- Súčet diagonálnych prvkov sa rovná počtu parametrov (*stupňov voľnosti lineárneho modelu*), $df \equiv tr(\mathbf{H}) = p + 1$.
- Podľa jednoduchého zaužívaného pravidla má i-te pozorovanie *výrazný* pákový efekt vtedy, keď $h_i > 2\frac{p+1}{n}$.
- Alternatívne sa na identifikáciu dá použiť tzv. Cookova vzdialenosť $D_i = \frac{\hat{\varepsilon}_i^2}{(p+1)\hat{\sigma}_i^2}\left[\frac{h_i}{(1-h_i)^2}\right]$ ktorá indikuje pákový efekt ak (pre veľké n) $D_i > 1$.

Ilustrácia (leverage)



- Z obrázku vpravo vidno, že bežný outlier (bod 20) nemá výrazný pákový efekt, naopak bod 41 je odľahlý nielen v odozve ale i hodnotou prediktoru, čo je výrazne riziková kombinácia pre vznik pákového efektu.

2.6.6 Kolinearita

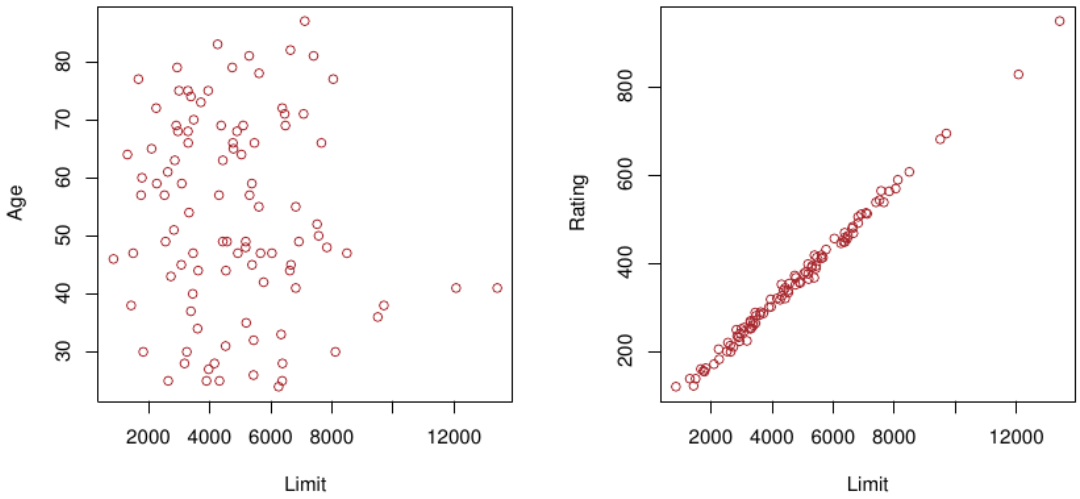
- Kolinearita predstavuje situáciu, keď dva alebo viac prediktorov navzájom spolu úzko súvisia.
- Lineárnu regresiu komplikuje preto, lebo je náročné oddeliť efekt jednotlivých premenných na odozvu.

Príklad (kreditky)

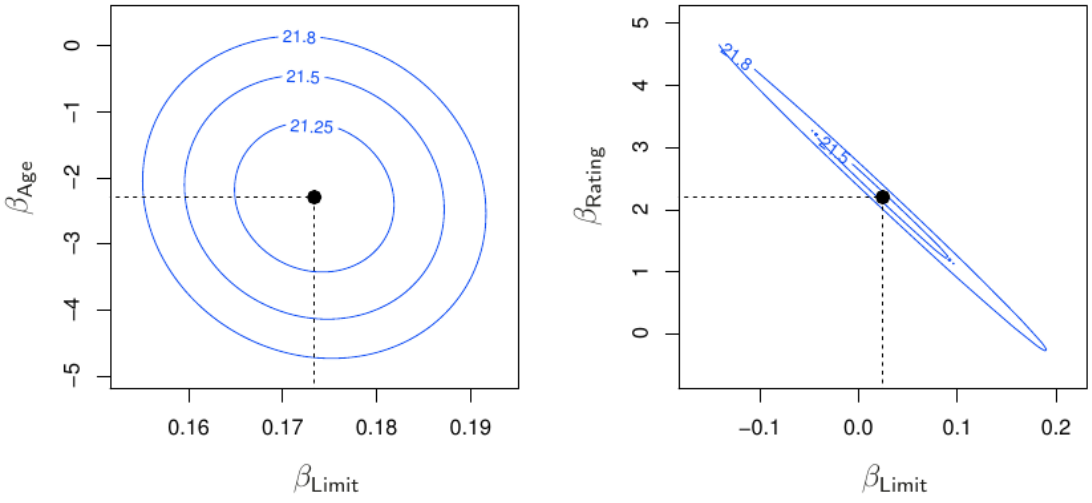
- Modelujme zadĺženie (*balance*) držiteľov kreditnej karty pomocou premenných ako vek (*age*), povolené prečerpanie (*limit*) a odhad schopnosti splácať úver (credit

rating).

- Zatiaľčo prvé dva prediktory spolu takmer nekorelujú, posledné dva sú korelované takmer dokonale.



- Ako sa to prejaví na modelovaní súvislosti dlhu a jednotlivých dvojíc? Na nasledujúcom obrázku je zobrazenie oboch modelov v priestore parametrov (v rozsahu $4 \cdot SE(\beta)$) s izočiarami RSS.



- Vľavo je optimum jednoznačne určené bodom $(0.173, -2.3)$, avšak vpravo vidieť celé údolie bodov s hodnotami RSS blízko minimu. To znamená vyššiu citlivosť na súbor údajov, ked malá zmena v pozorovaných údajoch spôsobí veľkú zmenu v odhadoch

parametrov (pozdĺž toho údolia) a to až do takých hodnôt, napr. $(-0.1, 4)$, ktoré by sme sotva očakávali (a dokázali interpretovať).

```
lm(Balance ~ Age + Limit, ISLR2::Credit) |>
  summary() |> coef() |> round(3) |>
  pander::pander()
lm(Balance ~ Rating + Limit, ISLR2::Credit) |>
  summary() |> coef() |> round(3) |>
  pander::pander()
```

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	-173.4	43.83	-3.957	0
Age	-2.291	0.672	-3.407	0.001
Limit	0.173	0.005	34.5	0

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	-377.5	45.25	-8.343	0
Rating	2.202	0.952	2.312	0.021
Limit	0.025	0.064	0.384	0.701

- Kolinearita teda redukuje presnosť odhadu parametrov, t.j. zvyšuje ich štandardnú chybu a znižuje testovaciu štatistiku v t-teste významnosti, takže menej často (než v 95%) dokážeme zamietnuť nulovú hypotézu $H_0 : \beta_j = 0$ keď v skutočnosti neplatí – tzn. znižuje *silu testu*.
- Dôležitosť významných prediktorov je tak v dôsledku kolinearity zamaskovaná, ťažšie odhaliteľná.
- Jednoduchým spôsobom detekcie kolinearity je pohľad na korelačnú maticu prediktorov. Ten žiaľ nefunguje v prípade kolinearity troch a viac premenných, teda *multikolinearity*.
- Istejšou cestou je výpočet tzv. faktoru inflácie rozptylu (*variance inflation factor*, VIF), ktorý predstavuje pomer rozptylu parametra β_j v plnom modeli ku rozptylu v modeli, kde je β_j jediným prediktorom.
- Najnižšou možnou hodnotou VIF je 1 (žiadna kolinearita). Zaužívaným pravidlom je spozornieť, keď VIF prekročí hodnotu 5 alebo 10.
- Prakticky sa vypočíta pomocou vzťahu

$$VIF(\hat{\beta}_j) = \frac{1}{1 - R^2_{X_j|X_{-j}}}$$

  kde $R^2_{X_j|X_{-j}}$ je koeficient determinácie regresného modelu závislosti X_j od ostatných prediktorov. Ten bude v prítomnosti kolinearity blízko 1, takže VIF veľmi narastie.
- Na kolinearitu sú dve jednoduché riešenia:

– vynechať z regresie jednu z problémových premenných (keďže je z hľadiska informačnej hodnoty nadbytočná),
– skombinovať korelujúce premenné do jedného (no stále ešte interpretovateľného) prediktoru.

Príklad (kreditky)

```
lm(Balance ~ Age + Rating + Limit, ISLR2::Credit) |>
  car::vif() |>
  round(2) |>
  rbind(VIF = _) |>
  pander::pander()
```

	Age	Rating	Limit
VIF	1.01	160.7	160.6

- Vysoká hodnota VIF indikuje, že jedna z premenných Rating a Limit je prebytočná, a stačí ju z regresie vylúčiť.

```
lm(Balance ~ Age + Limit, ISLR2::Credit) |>
  car::vif() |>
  round(2) |>
  rbind(VIF = _) |>
  pander::pander()
```

	Age	Limit
VIF	1.01	1.01

2.7 Záver

Na začiatku sme si ako analytici položili sedem otázok. Teraz, s novými znalosťami ich vieme zodpovedať:

1. *Existuje vzťah medzi rozpočtom na reklamu a odbytom?*
   Viacnásobnou regresiou a výpočtom vysokej hodnoty F-štatistiky (nízkej p-hodnoty) sme získali dôkaz o ich vzťahu.
2. *Aký je silný?*
   Podľa R^2 taký model vysvetľuje takmer 90% variability odbytu.

3. *Ktoré médiá sú za to zodpovedné?*
   Testy významnosti parametrov modelu ukazujú na reklamu v TV a rádiu.
4. *Aký silný je ich vzťah (z hladiska reklamy) ku odbytu?*
   Interval spoľahlivosti ukazuje na zmenu v odbyte medzi 43 až 49 jednotiek v prípade TV a medzi 172 až 206 jednotiek v prípade rádia s každou zmenou rozpočtu na reklamu (v danom médiu) o 1,000 USD.
5. *Ako presne dokážeme predpovedať odbyt?*
   Ak chceme predpovedať odbyt v konkrétnom meste, použijeme predikčný interval. Naopak, na určenie neistoty predpovede strednej hodnoty (priemerný odbyt na všetkých trhoch) je potrebný konfidenčný interval.
6. *Je vôbec ich vzťah lineárny?*
   Pomocou grafu pozorovaných vs. predpovedaných hodnôt sme zistili, že lineárny regresný model (a predpoklad aditivity) je nedostačujúci …
7. *Nie je medzi médiami synergia?*
   … pretože prediktory pôsobia v pozitívnej interakcii. S rastom rozpočtu pre TV rastie efekt rádia na odbyt (aj v opačnom poradí). Po rozšírení modelu súčinovou bázovou funkciou nie je v grafe rezíduí badať už žiaden podozrivý, systematický vzorec správania, a aj R^2 vzrástlo z 90% na takmer 97%.

3 Prevzorkovanie (resampling)

3.1 Úvod

- Prevzorkovacie metódy sú nenahraditeľným nástrojom modernej štatistiky.
- Pomocou jednoduchého triku dokážu získať ďalšie informácie o odhadovanom modeli.
- Trik spočíva v *aproximácii náhodného výberu zo základného súboru pomocou náhodného výberu z trénovacej vzorky* údajov.
- Hoci sú výpočtovo náročnejšie ako tradičné metódy (ktoré predpokladajú asymptotické vlastnosti zložiek modelu), pretože vyžadujú mnohonásobné opakovanie odhadu, tak vďaka pokroku vo výkone výpočtovej techniky nie je táto nevýhoda už veľmi obmedzujúca.
- Preberieme si dve najčastejšie používané metódy resamplingu:
  - *krížovú validáciu*, ktorou sa dá napr. odhadnúť testovacia chyba (vyhodnotenie modelu, angl. *model assessment*) alebo zvoliť vhodný stupeň flexibility (výber modelu, angl. *model selection*), a
  - *bootstrap*, ktorou sa okrem ďalších využití určuje miera presnosti odhadu parametrov modelu.

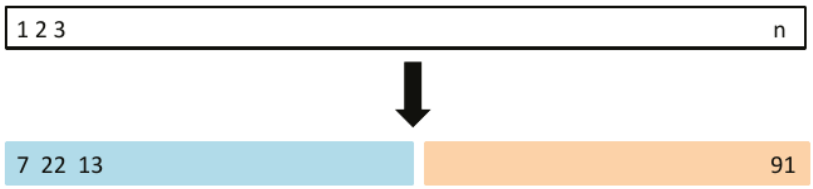
3.2 Krížová validácia

- Vieme už, že testovacia chyba vyjadruje presnosť predpovede odozvy pre nové pozorovania prediktorov (teda pre tie, ktoré neboli použité na trénovanie).
- Za model sa dá zaručiť, len ak má nízku testovaciu chybu.
- Testovacia chyba sa dá odhadnúť pomocou na-to-vyhradenej testovacej vzorky. Tú však často nemáme k dispozícii, pretože všetky dostupné pozorovania sa použili na tréning modelu a nové neboli zozbierané. Zároveň odhad trénovacej chyby je nespoľahlivou náhradou, pretože testovaciu chybu výrazne podhodnocuje.
- Sú však prístupy, ako testovaciu chybu odhadnúť z trénovacej vzorky.
  - Jeden z nich na to ide matematickou *korekciou* trénovacej chyby. Budeme sa mu venovať inde – v rámci metód výberu prediktorov.
  - Druhý pred odhadom modelu *odčlení* zo súboru pozorovaní časť, ktorú následne použije ako testovaciu vzorku. To je aj prípad krížovej validácie (cross-validation, CV) a dvoch jej metód, ktoré preberieme nižšie, po predstavení jednoduchšej metódy.

- Krížová validáciá sa spravidla používa na jeden z nasledujúcich cieľov:
  - odhad *hodnoty* testovacej chyby (na vyhodnotenie kvality modelu),
  - nájdenie *minima* testovacej chyby ako funkcie flexibility modelu (pre výber vhodného kandidáta spomedzi modelov jednej triedy).

3.2.1 Metóda jednej validačnej sady (validation set)

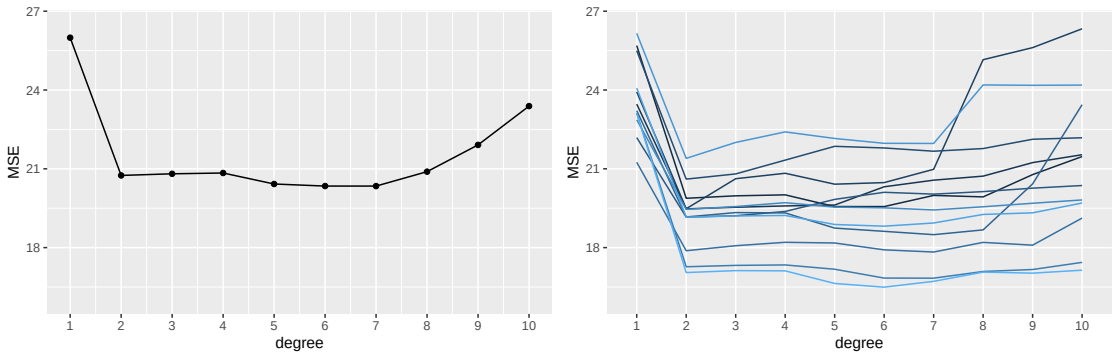
- V najjednoduchšej z metód druhého prístupu sa dataset náhodne rozdelí na trénovaciu a validačnú vzorku (iba raz).
- Predikcie modelu natrénovanom na prvej vzorke sa porovnajú s druhou a vypočíta sa stredná chyba. V prípade regresnej úlohy je to najčastejšie MSE.



Príklad (autá)

- V kapitole o lineárnom regresnom modeli sme použili F-test na posúdenie predikčnej schopnosti polynomickej regresie druhého či dokonca až piateho stupňa. Tento prístup využíval iba trénovaciu vzorku a polynóm 5 stupňa ešte označil za užitočný. Z interpretačného hľadiska však máme problém už aj s kvadratickou funkciou.
- Teraz, keď je k dispozícii validačná vzorka, porovnáme modely z hľadiska odhadu testovacej MSE.
- Na obrázku vľavo je zobrazená MSE pre 10 modelov (s rôznymi stupňami polynomickej funkcie) odhadnutých na rovnakej trénovacej vzorke a vyhodnotených na rovnakej validačnej vzorke. Je vidieť približný tvar písmena U a minimum niekde v bode 6 alebo 7, ale priebeh dáva tušiť veľkú variabilitu v odhade MSE.
- Aby sme získali predstavu o rozsahu neistoty určenia MSE touto metódou, skúsime náhodné rozdelenie súboru údajov do trénovacej a validačnej vzorky zopakovať 12-krát. Výsledok je na obrázku vpravo. Jediné, čo sa z neho dá vytušiť, je to, že lineárny model má systematicky najvyššiu chybu.

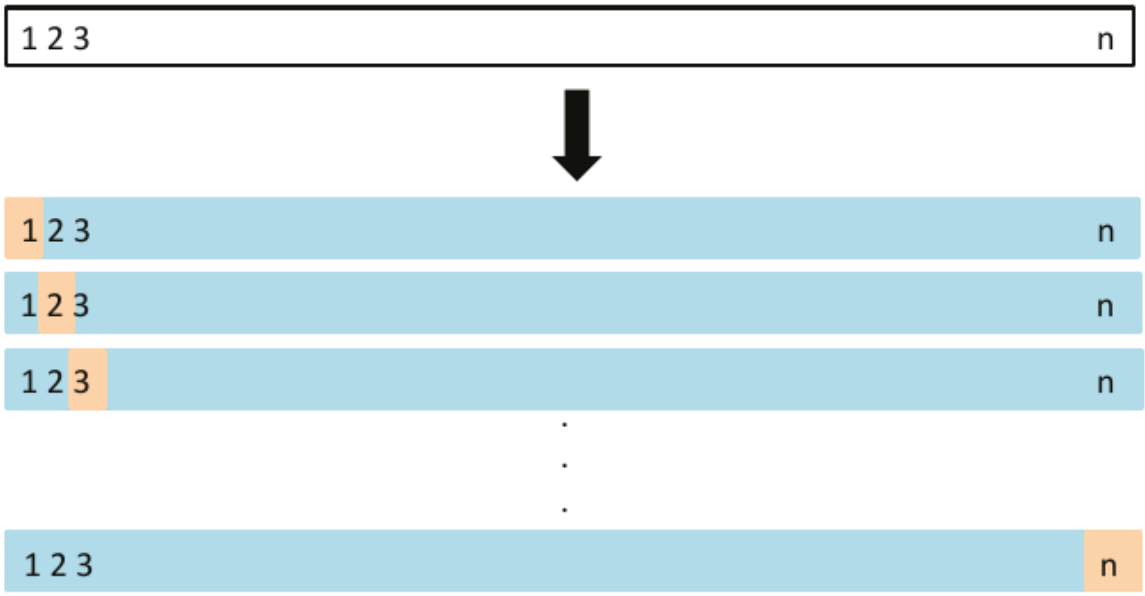
```r
# Function to
#  1. split the data proportionaly into training and evaluation sample,
#  2. fit the linear model given by formula to training set,
#  3. calculate MSE based on evaluation set.
# in: formula - formula defining relationship among variables,
#     data - data frame to split
#     proportion - numeric value in [0,1] interval, proportion of training sample to the wh
# out: numeric value of MSE
mse_validset_lm <- function(formula, data, proportion = 1/2, seed = sample(1000,1)) {
  n <- nrow(data)
  set.seed(seed)
  ind <- 1:n %in% sample(n, size = n*proportion)
  train <- data[ind,]
  valid <- data[!ind,]
  fit <- lm(formula, train)
  yname <- names(fit$model)[[1]]
  RSS <- sum((valid[[yname]] - predict(fit, newdata = valid))^2)
  df <- nrow(valid) - length(fit$coefficients)
   RSS / df
}

# 1 split, 10 polynomials of certain degrees
tibble::tibble(
  degree = 1:10,
  MSE = sapply(degree, function(x) mse_validset_lm(mpg ~ poly(horsepower, x),
                                                   ISLR2::Auto,
                                                   seed = 2))
) |>
  ggplot() + aes(x = degree, y = MSE) +
  geom_line() + geom_point() +
  scale_x_continuous(n.breaks = 10) +
  ylim(16, 26.5)
# 12 splits, 10 models
set.seed(1234)
replicate(n = 12, local({
  seed <- sample(1000, 1)
  sapply(1:10, function(x) mse_validset_lm(mpg ~ poly(horsepower, x),
                                           ISLR2::Auto, seed = seed))
})
) |>
  t() |> as.data.frame() |>
  setNames(1:10) |>
  dplyr::mutate(replication = 1:dplyr::n()) |>
  tidyr::pivot_longer(-replication, names_to = "degree", values_to = "MSE",
                      names_transform = forcats::as_factor) |>
  ggplot() + aes(x = degree, y = MSE, color = replication, group = replication) +
  geom_line(show.legend = FALSE) +
  ylim(16, 26.5)
```



- Metóda jednej validačnej sady je jednoduchá (koncepčne aj implementačne), ale má dve nevýhody:
  - Odhad testovacej chyby má veľký rozptyl, t.j. veľmi závisí od náhody pri rozdelení súboru na dve vzorky.
  - Iba časť údajov sa využije na trénovanie modelu. Ak je však *trénovaný na menšom počte* bodov, validačná chyba má tendenciu *nadhodnocovať testovaciu chybu*, ktorú by mal model trénovaný na celom datasete.
- Tieto nevýhody sa snažia eliminovať metódy krížovej validácie (pomocou aritmetického priemeru).

3.2.2 Vynechanie jedného pozorovania (leave-one-out CV)

- V metóde, pre ktorú sa zaužívala skratka LOOCV, sa vytvorí n rôznych dvojíc vzoriek, pričom v každej tvorí validačnú vzorku jediné pozorovanie a trénovacia vzorka obsahuje zvyšných $n-1$ pozorovaní. (To znamená, že v i-tej trénovacej vzorke je vynechané i-te pozorovanie.)



- Pre každú z n dvojíc vzoriek sa natrénuje model, vypočíta predikcia (v jedinom bode) a z nej MSE,

$$MSE_i = (y_i - \hat{y}_i)^2.$$

- Jednotlivé MSE_i sú síce vďaka vylúčeniu i-teho pozorovania z trénovacej vzorky približne nevychýleným odhadom, no keďže sú vypočítané z jedinej hodnoty, majú veľmi *veľký rozptyl*.
- Variabilitu takýchto MSE *znížime priemerom*. Odhad testovacej chyby bude mať tvar

$$\overline{MSE}_{(n)} = \frac{1}{n}\sum_{i=1}^{n} MSE_i$$

- Oproti metóde jednej validačnej sady má LOOCV okrem nižšieho rozptylu aj ďalšie *výhody:*
  - *menšia vychýlenosť* (na trénovanie sú zakaždým použité takmer všetky pozorovania, takže skutočnú chybu až tak veľmi nenadhodnocuje),
  - *stálosť výsledku* (pretože delenie na trénovaciu a validačnú vzorku už nie je náhodné).
- Naopak *nevýhodou* LOOCV je *výpočtová náročnosť*, pretože model musí byť odhadnutý až n-krát.
- V prípade lineárneho modelu však existuje jednoduchá skratka k výsledku,

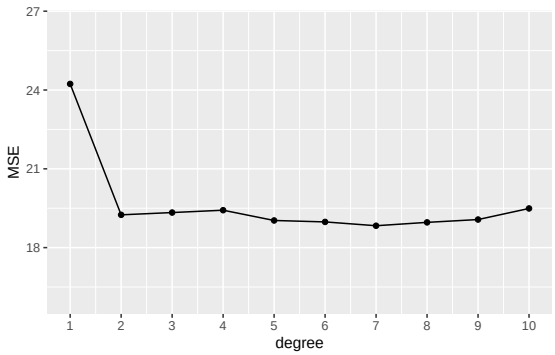
$$\overline{MSE}_{(n)} = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{y_i - \hat{y}_i}{1 - h_i}\right)^2$$

pri ktorej stačí urobiť jediný odhad modelu a to na celej vzorke. Z neho sú vypočítané stredné hodnoty $\hat{y}_i$ a diagonálne prvky h_i matice váhových koeficientov (hat matrix) $\mathbf{H}$ vyjadrujúce relatívnu veľkosť pákového efektu. Rezíduá bodov s vysokým pákovým efektom sú tak dostatočne nafúknuté, aby vzťah mohol platiť.

Príklad (autá)

- Príklad z predošlej časti zopakujeme pre LOOCV metódu.
- Súbor údajov *Auto* obsahuje záznamy o 392 autách, a bez použitia efektívnejšej verzie pre lineárny model by výpočet MSE pre všetkých 10 regresných modelov trval približne 8s (na CPU z roku 2020).

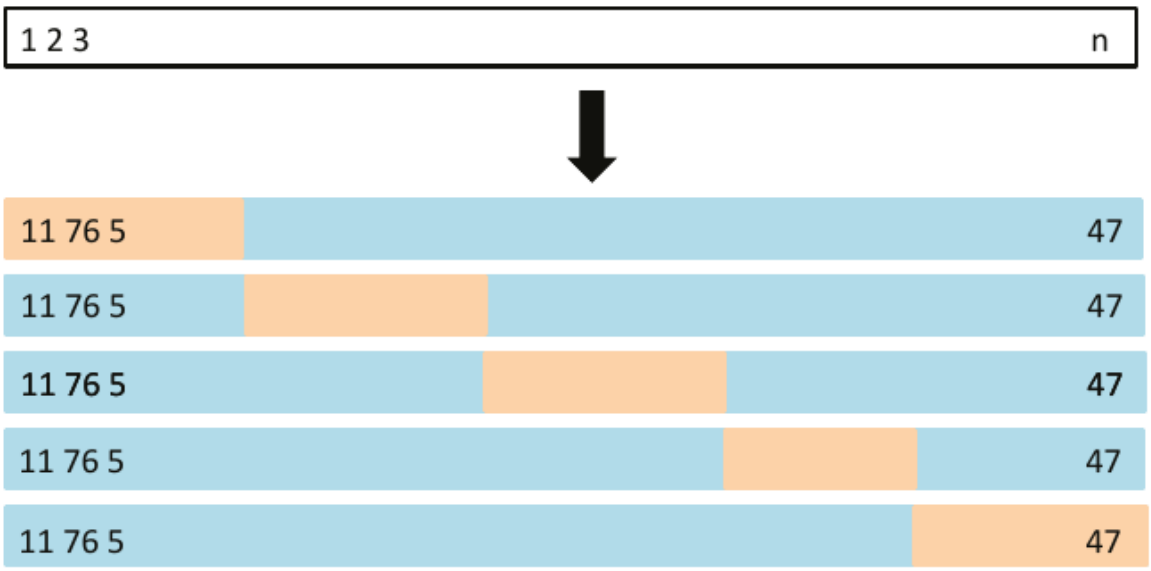
```r
# (General version)
# Function that for every observation from data repeats the following steps
#  1. fit the linear model given by formula to training set,
#     where the one particular observation is ommited,
#  2. calculate MSE based on prediction for the ommited observation,
# and calculates average MSE.
# in: formula - formula defining relationship among variables,
#     data - data frame of observations
# out: numeric value of MSE
mse_loocv_lm <- function(formula, data) {
  n <- nrow(data)
  y <- formula |> _[[2]] |> as.character() |> getElement(data, name = _)
  df <- formula |> terms() |> attr("term.labels") |> length() |> (`+`)(1)
  MSE <- sapply(1:n, function(i) {
    lm(formula, data[-i,]) |>
      predict(newdata = data[i,]) |>
      (`-`)(y[i]) |>
      (`^`)(2)
  })
  mean(MSE)
}
# (Effective version)
# Function, that implements the effective version of LOOCV, valid for linear models.
mse_loocv_lm <- function(formula, data) {
  n <- nrow(data)
  fit <- lm(formula, data)
  eps <- residuals(fit)
  h <- hatvalues(fit)
  mean((eps/(1-h))^2)
}

# repeat for 10 models

# system.time(
  dat_mse <- tibble::tibble(
    degree = 1:10,
    MSE = sapply(degree, function(x) mse_loocv_lm(mpg ~ poly(horsepower, x),
                                                  ISLR2::Auto))
  )
# ) |> _["elapsed"] |> round(1) |> cat("time: ", a = _, "s")

  # display
dat_mse |>
  ggplot() + aes(x = degree, y = MSE) +
  geom_line() + geom_point() +
  scale_x_continuous(n.breaks = 10) + ylim(16, 26.5)
# built-in implementation of the general version:
# glm(mpg ~  poly(horsepower, 1), data = ISLR2::Auto) |>
#   boot::cv.glm(data = ISLR2::Auto) |>
#   _$delta[1]
```



3.2.3 k-násobná (k-fold CV)

- Videli sme, že metóda jednej validačnej sady
  - vedie k nadhodnoteným stredným chybám (*high bias*), pretože model je natrénovaný iba na polovici pozorovaní.
  - Zároveň hodnoty veľmi fluktuujú s náhodným výberom trénovacej vzorky, pretože MSE nie je spresnené žiadnym priemerovaním.
- Ten druhý problém by sčasti riešilo vyhodnotenie (validácia) MSE do kríža, čiže druhá chyba MSE by sa vypočítala zamenením rolí trénovacej a validačnej vzorky, a potom by sa obe MSE spriemerovali.
- Na prvý i druhý problém odpovedá metóda LOOCV, ktorá model zakaždým odhaduje z takmer celej sady pozorovaní, takže odhad MSE je takmer nevychýlený. Avšak v dôsledku vysokého prekrytu trénovacích vzoriek sú jednotlivé hodnoty MSE veľmi korelované, takže rozptyl strednej MSE je vysoký (*high variance*).
- *Kompromisné riešenie* problému vysokého vychýlenia odhadu MSE na jednej strane a vysokého rozptylu na druhej strane (bias-variance trade-off) ponúka metóda *k-násobnej krížovej validácie* (k-fold CV).
- Jej princíp spočíva v náhodnom rozdelení súboru pozorovaní na k vzoriek (skupín, folds) približne rovnakej veľkosti. Následne v j tom kroku cyklu (z celkového počtu k) sa j-ta vzorka označí za validačnú a všetky zvyšné spolu za trénovaciu, odhadne sa model, vypočítajú predikcie a chyba MSE_j. Výsledkom je priemer z k odhadov MSE,

$$\overline{MSE}_{(k)} = \frac{1}{k}\sum_{j=1}^{k} MSE_j$$

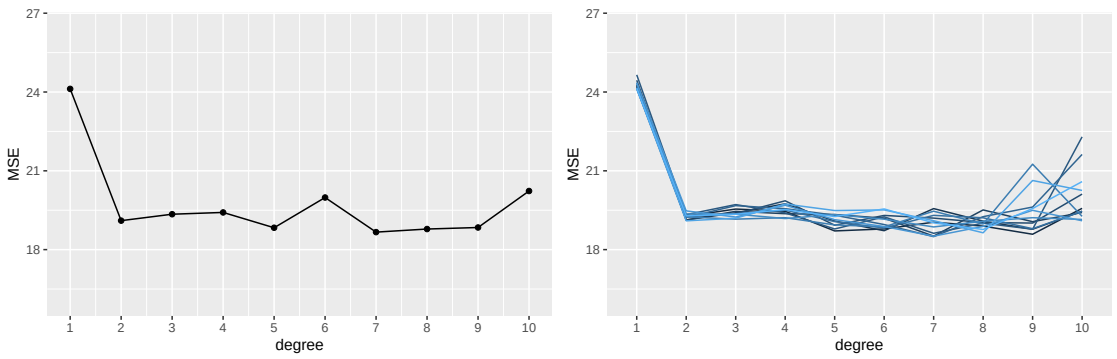


- Je zjavné, že voľba $k = 2$ vedie k priamemu rozšíreniu metódy jednej validačnej sady na validáciu do kríža, zatiaľčo na opačnom konci spektra je špeciálnym prípadom metóda LOOCV, ak $k = n$.
- Presné určenie hodnoty k nie je nijak zásadné, štandardne sa zvykne voliť $k = 5$ alebo $k = 10$. Výhoda oproti LOOCV je nielen v znížení rozptylu, ale aj menšej výpočtovej náročnosti. Navyše pri tomto počte skupín sa výpočet dá efektívne paralelizovať a tým (na viac-vláknových procesoroch) ešte výraznejšie zredukovať výpočtový čas.

Príklad (autá)

- Príklad z predošlých častí zopakujeme pre k-násobnú CV.
- Na obrázku vľavo je jeden odhad MSE pre 10 lineárnych modelov s rôznym stupňom polynomickej funkcie a $k = 5$. Sú veľmi podobné odhadom pomocou LOOCV.
- Na obrázku vpravo je výpočet zopakovaný 12-krát. Výsledky sa líšia v dôsledku náhody vo výbere pozorovaní do jednotlivých skupín, no ich rozptyl je podstatne menší než v prípade metódy jednej validačnej sady.

```r
# Function that uses boot::cv.glm to split observations into K folds and repeats the follow
#  1. fit the linear model given by formula to training set,
#     where one particular fold is ommited,
#  2. calculate MSE based on predictions for the ommited fold,
# and calculates average MSE.
# in: formula - formula defining relationship among variables,
#     data - data frame of observations
#     K - integer, number of folds
# out: numeric value of average MSE
mse_kfold_lm <- function(formula, data, K = 5) {
  glm(formula, data = data) |> boot::cv.glm(data = data, K = K) |> _$delta[1]
}

# single run
set.seed(1)
tibble::tibble(
  degree = 1:10,
  MSE = sapply(degree, function(x)
    mse_kfold_lm(mpg ~  poly(horsepower, x), data = ISLR2::Auto)
    )
  ) |>
  ggplot() + aes(x = degree, y = MSE) +
  geom_line() + geom_point() +
  scale_x_continuous(n.breaks = 10) + ylim(16, 26.5)
# 12-times repeat
replicate(n = 12, sapply(1:10, function(x)
  mse_kfold_lm(mpg ~ poly(horsepower, x), ISLR2::Auto)
  )) |>
  t() |> as.data.frame() |>
  setNames(1:10) |>
  dplyr::mutate(replication = 1:dplyr::n()) |>
  tidyr::pivot_longer(-replication, names_to = "degree", values_to = "MSE",
                      names_transform = forcats::as_factor) |>
  ggplot() + aes(x = degree, y = MSE, color = replication, group = replication) +
  geom_line(show.legend = FALSE) +
  ylim(16, 26.5)
```



3.3 Bootstrap

- Bootstrap metóda je nesmierne užitočný štatistický nástroj, ktorý umožňuje kvantifikovať neistotu štatistického odhadu.
- Napríklad v lineárnej regresii sa dá použiť na odhad štandardných chýb parametrov modelu. (To sa síce nemusí zdať veľmi užitočné, keďže poznáme explicitné matematické vzťahy na ich výpočet, tie však platia iba vtedy, ak sú splnené určité predpoklady modelu.)
- Sila bootsrap spočíva v tom, že sa dá aplikovať na široké spektrum štatistických metód, a to aj takých, ktoré neumožňujú jednoduché vyjadrenie miery variability (a tá potom nie je ani súčasťou výstupu štatistického softvéru).
- Názov pochádza z frázy "pull oneself up by one's bootstraps" použitej v príbehu o dobrodružstvách baróna Prášila (Adventures of Baron Munchausen):

  The Baron had fallen to the bottom of a deep lake. Just when it looked like all was lost, he thought to pick himself up by his own bootstraps.

- Aj princíp metódy bootstrap môže vyvolávať pochybnosti, ale – na rozdiel od barónovho spôsobu záchrany – v skutočnosti aj funguje.

3.3.1 Princíp

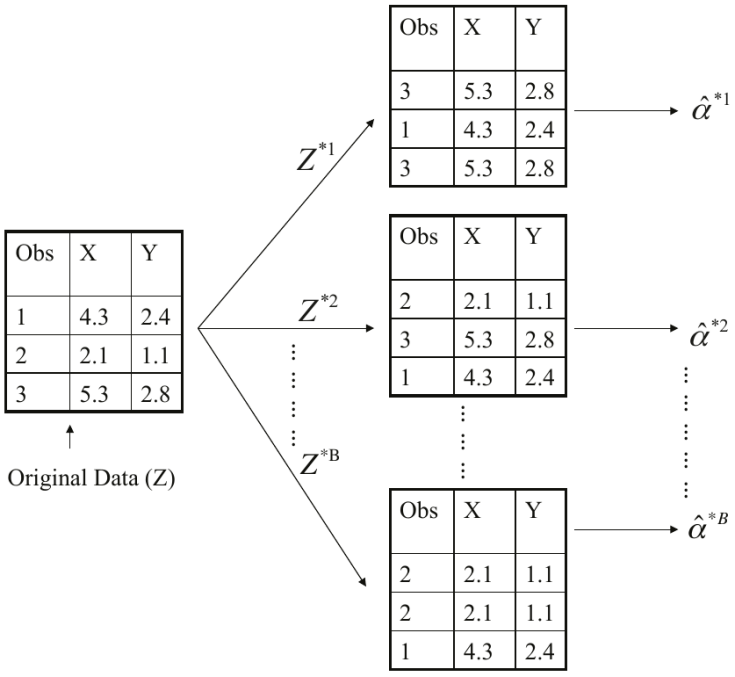
- Označme naše pozorovania (jednu realizáciu náhodného výberu zo základného súboru) ako Z a štatistiku, ktorú potrebujeme určiť (napr. parameter modelu alebo strednú hodnotu odozvy), označme symbolom α.
- Na základe jednej sady údajov dokážeme odhadnúť iba strednú hodnotu $\hat{\alpha}$.
- Na určenie variability odhadu α potrebujeme poznať ďalšie informácie o jeho rozdelení pravdepodobnosti. Tu máme dve možnosti:

  - budeme veriť predpokladu o tomto rozdelení (napr. že je gaussovské) a použijeme vzťah pre výpočet štandardnej chyby platný za podmienky tohoto predpokladu,
  - získame ďalšie súbory pozorovaní $Z^1, Z^2, \ldots, Z^N$, pomocou nich odhadneme $\hat{\alpha}^1, \hat{\alpha}^2, \ldots, \hat{\alpha}^N$ a z nich štandardnú chybu $SE(\hat{\alpha})$.

- Prvá možnosť predstavuje asymptotický prístup, ktorý sa v štatistike aplikuje oddávna, v mnohých prípadoch napr. využíva centrálnu limitnú vetu na zaistenie predpokladu normality pri väčšej vzorke.
- Druhá možnosť je v praxi problematická, pretože získanie nových pozorovaní býva ekonomicky nákladné, niekedy aj nemožné. Preto sa využíva iba v simulačných štúdiách, kedy je známe rozdelenie (alebo základný súbor), z ktorého boli pozorovania získané. Väčšinou je to nejaký model.
- Čo však robiť v reálnej štúdii, kedy nové pozorovania nemáme a o splnení predpokladov pochybujeme alebo ich na odhad presnosti nevieme použiť?
- Riešením je *základný súbor aproximovať existujúcim súborom pozorovaní* Z a náhodným výberom (*s opakovaním*) získať *rovnako veľké* vzorky $Z^{*1}, Z^{*1}, \ldots, Z^{*B}$. Z nich sa vypočítajú odhady $\hat{\alpha}^{*1}, \hat{\alpha}^{*2}, \ldots, \hat{\alpha}^{*B}$, ktoré tvoria podklad vzorkovacieho rozdelenia (*sampling distribution*) odhadu $\hat{\alpha}$.

Ilustrácia (bootstrap)

Nasledujúci obrázok ilustruje bootstrap metódu pre $n = 3$.



- Pomocou vzorkovacieho rozdelenia vieme určiť rôzne vlastnosti rozdelenia odhadu $\hat{\alpha}$, ako

napr.

  - štandardnú odchýlku $SE(\hat{\alpha}) = \sqrt{\frac{1}{B-1}\sum_{i=1}^{B}(\hat{\alpha}^{*i} - \bar{\hat{\alpha}}^{*})^2}$, kde $\bar{\hat{\alpha}}^{*} = \frac{1}{B}\sum_{i=1}^{B}\hat{\alpha}^{*i}$,
  - 95% interval spoľahlivosti $[\hat{\alpha}^{*}_{(2.5\%)}, \hat{\alpha}^{*}_{(97.5\%)}]$, kde $\hat{\alpha}^{*}_{(p\%)}$ predstavuje p-ty percentil (bootstrap percentile).

- V prípade dát s komplexnejšou štruktúrou, ako sú napr. časové rady, je nutné postup prispôsobiť. V prípade stacionárnych časových radov napr. tak, že sa dáta rozdelia do blokov a náhodný výber (s opakovaním) sa vykoná na nich namiesto na jednotlivých pozorovaniach.

3.3.2 Príklady

Ilustrácia (investovanie)

- Predstavme si, že chceme investovať určitú jednotkovú sumu peňazí do dvoch finančných aktív, z ktorých jedno dáva výnos (return) X a druhé Y. Oboje sú náhodné premenné s normálnym rozdelením a mierne spolu korelujú.
- Do aktíva s výnosom X investujeme čiastku α, do Y zas čiastku $1-\alpha$.
- Pretože s výnosmi je spojená premenlivosť, chceme zvoliť α tak, aby sa minimalizovalo celkové riziko (t.j. rozptyl) našej investície.
- Deriváciou celkového rozptylu

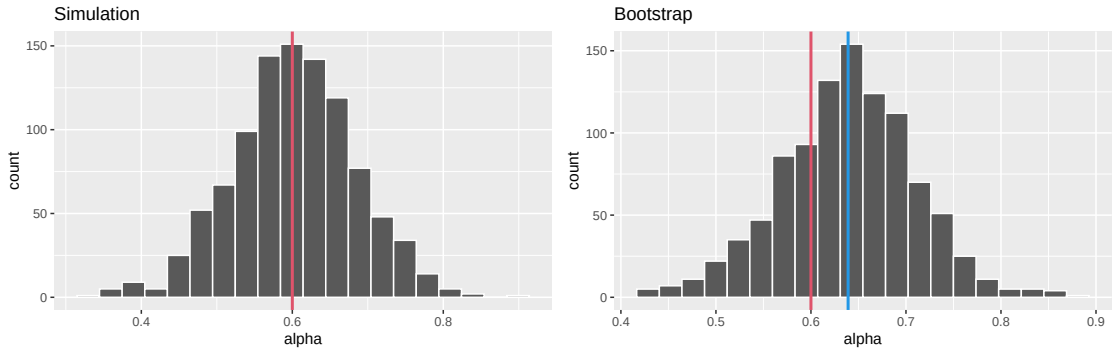
$$var(\alpha X + (1-\alpha)Y) = \alpha^2\sigma_X^2 + (1-\alpha)^2\sigma_Y^2 + \alpha(1-\alpha)\sigma_{XY}$$

  podľa α, kde sme označili $\sigma_X^2 = var(X)$ a $\sigma_{XY} = cov(X,Y)$, a položením derivácie do nuly dostaneme vzťah

$$\alpha = \frac{\sigma_Y^2 - \sigma_{XY}}{\sigma_X^2 + \sigma_Y^2 - 2\sigma_{XY}}$$

- Rozptyly a kovariancia sú v praxi neznáme, dajú sa však odhadnúť z minulých pozorovaní výnosov. Z odhadov $\hat{\sigma}_X^2$, $\hat{\sigma}_Y^2$, $\hat{\sigma}_{XY}$ dostaneme i odhad $\hat{\alpha}$.
- Keďže v našej ilustrácii poznáme (združené) rozdelenie X a Y, tisíckrát vygenerujeme súbor 100 párov hodnôt výnosov a pre každý súbor vypočítame odhad hodnoty podielu α.
- Potom však jeden z týchto súborov považujeme za súbor pozorovaní a z neho pomocou bootstrap metódy s počtom prevzorkovaní $B = 1000$ vypočítame hodnoty $\hat{\alpha}^{*}$.
- Na obrázkoch je zobrazené samplovacie rozdelenie náhodnej premennej α, vľavo pomocou náhodných výberov zo základného súboru, vpravo pomocou náhodných výberov z jedného takého súboru. Vidno, že hoci odhad strednej hodnoty, $\hat{\alpha}$ (modrá), sa nachádza "iba viac-menej blízko" skutočnej strednej hodnoty α (červená), rozptyl bootstrap vzorky je veľmi podobný tomu zo simulácií.

```r
n <- 100
# Distribution of alpha based on samples from true population
# -----------------------------------------------
# number of data points and data sets
N <- 1000
# Function to construct bivariate covariance matrix from standard deviation 'sd' and correl
make_cov_matrix <- function(sd = c(1,sqrt(1.25)), cor = 0.5/prod(sd)) {
  diag(sd) %*% cbind(c(1,cor), c(cor,1)) %*% diag(sd)
}
# Function to calculate ratio alpha from bivariate covariance matrix.
make_alpha <- function(cov) {
  (cov[2,2] - cov[1,2]) / (cov[1,1] + cov[2,2] - 2*cov[1,2])
}
# Function to generaten pairs from 2D normal distribution with mean vector 'mu' and covaria
simulate_data <- function(n, mu = c(0,0), cov = make_cov_matrix()) {
  X1 <- rnorm(n)
  X2 <- rnorm(n)
  cor <- cov[1,2]/sqrt(cov[1,1]*cov[2,2])
  X3 <- cor * X1 + sqrt(1-cor^2) * X2
  data.frame(X = mu[1] + sqrt(cov[1,1]) * X1,
             Y = mu[2] + sqrt(cov[2,2]) * X3)
  # alternatively: mvtnorm::rmvnorm(n, mu = mu, sigma = cov)
}
# simulate values of alpha
sim_alpha <- make_cov_matrix() |> # theoretical covariance
  simulate_data(n = n, cov = _)  |>  # generate random data
  cov() |>  # estimate covariance
  make_alpha() |>  # calculate alpha
  replicate(n = N)  # repeat
# display hsitogram with true value of alpha indicated
sim_alpha |>
  data.frame(alpha = _) |>
  ggplot() + aes(x = alpha) +
  geom_histogram(bins = 20, color = "white") +
  geom_vline(xintercept = make_cov_matrix() |> make_alpha(),
             color = 2, linewidth = 1) +
  ggtitle("Simulation")
# Distribution of alpha based on bootstrap samples
# -----------------------------------------------
B <- 1000
# single data set of observations
set.seed(1)
dat <- simulate_data(n = 100)
# Function to make 'n' random drawings from 'data'.
resample_data <- function(data, n) {
  ind <- sample(nrow(data), size = n, replace = TRUE)
  data[ind,]
}
# make sampling distribution of alpha
boot_alpha <- dat |>
  resample_data(n = 100) |>
  cov() |>
  make_alpha() |>
```




```
rbind(alpha = c(true = make_cov_matrix() |> make_alpha(),
                sim = mean(sim_alpha),
                boot = mean(boot_alpha)),
      SE = c(true = NA,
             sim = sd(sim_alpha),
             boot = sd(boot_alpha))
  ) |> round(3)
```

```
      true   sim  boot
alpha  0.6 0.602 0.639
SE      NA 0.081 0.071
```

Príklad (autá)

- Naposledy sa vrátime k modelovaniu dojazdu auta pomocou výkonu motora a porovnáme odhady štandardných chýb parametrov lineárneho modelu metódou bootstrap s odhadmi odvodenými analyticky na základe predpokladov.
- Tentokrát využijeme vstavanú funkciu *boot* zo základného balíka *boot*, pre ktorú je iba potrebné pripraviť funkciu na výpočet štatistiky (jednej alebo viacerých), ktorá nás zaujíma.

```
# Function to calculate parameters of linear regression based on data subset given by vecto
make_linpar <- function(data, indices) {
  lm(mpg ~ horsepower, data = data, subset = indices) |> coef()
}
n <- nrow(ISLR2::Auto)
make_linpar(ISLR2::Auto, sample(n, n, replace = TRUE))
```

```
(Intercept)  horsepower
 40.8819444  -0.1627521
```

- Funkcia so vstupom súboru pozorovaní *Auto* a náhodnej množiny (aj opakujúcich sa) indexov riadkov vráti mierne odlišné odhady ako sme videli v predošlej kapitole. Pre získanie miery variability takýchto odhadov (vo všeobecnosti pre získanie vzorkovacieho rozdelenia) je potrebné odhad veľakrát zopakovať a z výsledku získať požadovanú mieru.

```
B <- 1000
make_linpar(ISLR2::Auto, sample(n, n, replace = TRUE)) |>
  replicate(n = B) |>
  apply(MARGIN = 1, sd) |>
  rbind(SE = _)
```

```
   (Intercept)  horsepower
SE   0.8463759 0.007384191
```

- Vstavaná funkcia k tomu pridá i strednú hodnotu štatistiky odhadnutú z pôvodného súboru (original) a odchýľku odhadu strednej hodnoty zo vzorkovacieho rozdelenia (bias).

```
boot::boot(ISLR2::Auto, make_linpar, R = B)
```

```
ORDINARY NONPARAMETRIC BOOTSTRAP


Call:
boot::boot(data = ISLR2::Auto, statistic = make_linpar, R = B)


Bootstrap Statistics :
      original       bias    std. error
t1* 39.9358610  0.035484127 0.850724147
t2* -0.1578447 -0.000318217 0.007465228
```

- Porovnajme bootstrap odhad presnosti s odhadom na základe predpokladov modelu.

```
lm(mpg ~ horsepower, ISLR2::Auto) |>
  summary() |>
  coef()
```

```
              Estimate  Std. Error   t value      Pr(>|t|)
(Intercept) 39.9358610 0.717498656  55.65984 1.220362e-187
horsepower  -0.1578447 0.006445501 -24.48914  7.031989e-81
```

- Štandardné chyby vypočítané analyticky z predpokladov modelu sú zjavne nižšie. Znamená to chybu v bootstrap metóde?
- Nie. Asymptotické odhady sú podhodnotené kvôli nesplneným podmienkam lineárneho modelu: jednak počítajú s nenáhodnosťou hodnôt prediktorov (t.j. že všetok rozptyl odozvy pochádza zo šumovej zložky), jednak skutočný vzťah medzi dojazdom a výkonom (podľa bodového grafu) je zjavne nelineárny.

4 Výber prediktorov

4.1 Úvod

- Lineárny regresný model má svoje obmedzenia (aditivita, linearita) ktoré ho znevýhodňujú pri predikcii v prípade zložitejších vzťahov medzi odozvou a prediktormi. Napriek tomu je v praxi veľmi obľúbeným nástrojom – je jednoducho vysvetliteľný (dedukcia) a prekvapivo často konkurencieschopný (z hľadiska predikcie).
- Ako sme si ukázali, niektoré obmedzenia (aditivita, linearita) sa dajú vyriešiť rozšírením modelu pomocou vhodných bázových funkcií. Neskôr sa ešte povenujeme rozšíreniu konštrukcie lineárneho modelu na adaptovanie sa nelineárnym vzťahom (pri zachovaní aditivity).
- Lineárny regresný model sa však dá vylepšiť aj vhodnou *metódou odhadu* – tak z hľadiska presnosti predpovedí ako aj intepretovateľnosti modelu.
- Z hľadiska *presnosti predpovede:*
  - Ak je skutočný vzťah lineárny, potom klasická metóda najmenších štvorcov dáva odhad s nízkou výchylkou (bias) a navyše, ak je k dispozícii dostatok dát ($n \gg p$), tak aj s nízkou varianciou.
  - Avšak, ak n nie je oveľa väčšie ako p, potom odhad môže byť veľmi premenlivý a viesť k prehnanému prispôsobeniu pozorovaniam (overfitting), čo v konečnom dôsledku znamená nekvalitné predpovede.
  - Nakoniec, ak $p > n$, tak obyčajná metóda najmenších štvorcov sa ani nedá použiť, pretože neexistuje jediné riešenie (variancia je nekonečná).
  - Obmedzením (*stlačením*) odhadov parametrov sa dá významne *zredukovať varianciu*, a to na úkor iba *zanedbateľného nárastu výchylky* (bias).
- Z hľadiska *interpretovateľnosti modelu:*
  - Viaceré premenné použité v regresii nemajú žiaden vzťah ku odozve takže iba zbytočne zvyšujú komplexnosť modelu.
  - Ich odstránenie – t.j. stlačenie zodpovedajúcich parametrov až na nulu – zlepší interpretovateľnosť modelu.
  - Obyčajná MNŠ prakticky nikdy nevyústi do nulových odhadov parametrov.
  - To je úlohou automatických metód na *výber podmnožiny relevantných prediktorov* (feature/variable selection).

- Je *veľa alternatív* ako nahradiť obyčajnú metódu najmenších štvorcov *na odhad modelu*. Je dobré vedieť o týchto troch triedach metód:
  - výber podmnožiny relevantných prediktorov (subset selection),
  - redukcia vplyvu irelevantných prediktorov (shrinkage),
  - transformácia priestoru prediktorov do podpriestoru s väčšou informačnou hodnotou (dimension reduction).

4.2 Výber podmnožiny

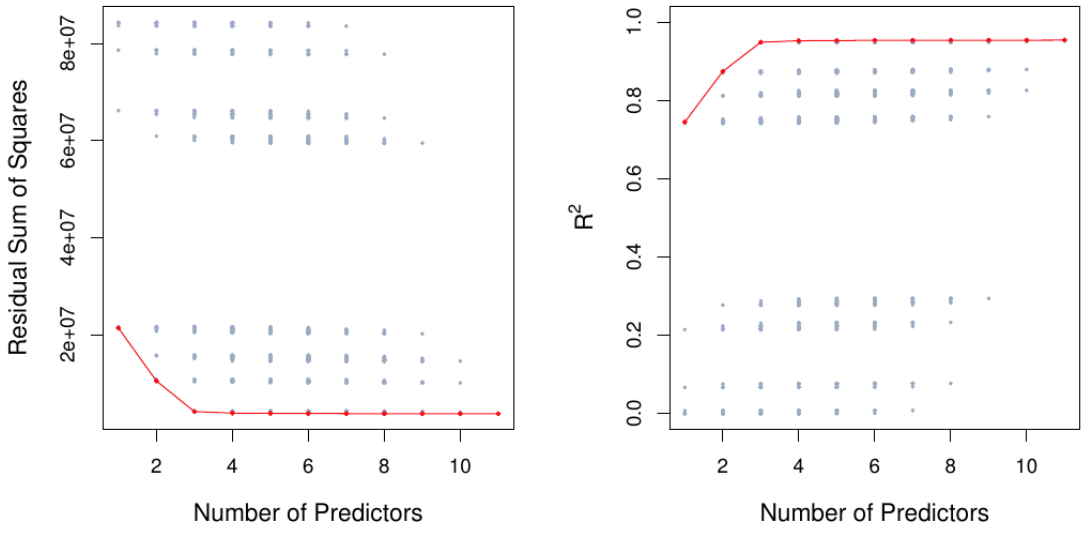
Niektoré metódy z tejto triedy sme si v krátkosti predstavili už v kapitole o regresnom modeli.

4.2.1 Metóda hrubej sily

- Úloha vybrať najlepší z 2^p možných lineárnych regresných modelov už napohľad (kvôli množstvu kombinácií) nie je triviálna.
- Metóda výberu najlepšej podmnožiny prediktorov (*best subset selection*) na to ide hrubou silou.
- Postup je rozdelený do dvoch krokov:
  1. Pre $k = 0, 1, \ldots, p$ zopakovať procedúru:
     a) odhadnúť $\binom{p}{k}$ modelov, ktoré obsahujú presne k prediktorov,
     b) model s najmenším RSS (alebo najväčším R^2) označiť ako $\mathcal{M}_k$.
  2. Spomedzi $\mathcal{M}_0, \ldots, \mathcal{M}_p$ vybrať najlepší model v zmysle jedného z kritérii: krížová validácia, C_p (AIC), BIC, korigovaný R^2.

Príklad (kreditky)

- Dataset *Credits* obsahuje okrem odozvy (*Balance*) aj 10 ďalších premenných, z toho jednu kvalitatívnu s viac ako dvomi úrovňami (*Region*), takže celkový počet prediktorov je 11.
- Na obrázkoch nižšie sú zobrazené hodnoty RSS, resp. R^2 pre všetky možné lineárne regresné modely. Červená krivka spája najnižšie hodnoty pre každý počet parametrov.



- Od modelu s tromi prediktormi je už zjavne iba minimálne zlepšenie. Tie tri sú v poradí (podľa súhrnu nižšie) *Rating*, *Income* a *Student*.

```
# leaps::regsubsets(Balance ~ ., data = ISLR2::Credit, method = "exhaustive") |>
#   plot() # broom::tidy()
tab <- leaps::regsubsets(Balance ~ ., data = ISLR2::Credit,
                  method = "exhaustive", nvmax = 8) |>
  broom::tidy() |>
  dplyr::mutate(across(-(13:16), ~ifelse(.x, "*", "")),
                across(13:15, ~ signif(.x, 4)),
                across(16, ~ signif(.x, 2))
                )
```

```
tab |>
 pander::pander()
```

Tabuľka 4.1: Table continues below

(Intercept)	Income	Limit	Rating	Cards	Age	Education	OwnYes
*			*				
*	*		*				
*	*		*				
*	*	*		*			

*	*	*	*	*			
*	*	*	*	*	*		
*	*	*	*	*	*		*
*	*	*	*	*	*		*

Tabuľka 4.2: Table continues below

StudentYes	MarriedYes	RegionSouth	RegionWest	r.squared	adj.r.squared
				0.7458	0.7452
				0.8751	0.8745
*				0.9499	0.9495
*				0.9536	0.9531
*				0.9542	0.9536
*				0.9547	0.954
*				0.9548	0.954
*			*	0.9549	0.954

BIC	mallows_cp
-535.9	1800
-814.2	690
-1173	41
-1198	11
-1197	8.1
-1196	5.6
-1191	6.5
-1186	7.8

- Metóda hrubej sily má dve hlavné nevýhody:
  - výpočtová náročnosť – napr. pri $p = 30$ prediktoroch už treba odhadnúť miliardu modelov,
  - riziko overfiting-u – čím väčší priestor, tým väčšia šanca nájdenia modelu, ktorý sadne na trénovacie dáta, ale zlyhá na testovacích.

4.2.2 Metóda postupného výberu

- Oproti metóde hrubej sily prejde metóda postupného výberu oveľa menšiu sadu modelov.
- Pri metóde dopredného (forward) výberu je postup nasledujúci:

1. Odhadnúť model bez prediktorov $\mathcal{M}_0$ a pre $k = 0, 1, \ldots, p-1$ zopakovať procedúru:
   a) Odhadnúť $(p-k)$ modelov, ktoré obsahujú presne $(k+1)$ prediktorov.
   b) Model s najmenším RSS (alebo najväčším R^2) označiť ako $\mathcal{M}_{k+1}$.
2. Spomedzi $\mathcal{M}_0, \ldots, \mathcal{M}_p$ vybrať najlepší model v zmysle jedného z kritérii: krížová validácia, C_p (AIC), BIC, korigovaný R^2.

- Dopredný výber žiaľ negarantuje nájdenie najlepšieho modelu. Napr. pre $p = 3$ ak najlepší jedno-prediktorový model obsahuje X_1 a najlepši dvoj-prediktorový model obsahuje X_2, X_3, potom ten v doprednej procedúre nebude vybraný, pretože $\mathcal{M}_2$ už musí obsahovať prediktor X_1.
- Alternatívna metóda ku doprednej je spätná (backward stepwise selection):

1. Odhadnúť model so všetkými prediktormi $\mathcal{M}_p$ a pre $k = p, p-1, \ldots, 1$ zopakovať procedúru:
   a) Odhadnúť k rôznych modelov, ktoré obsahujú presne $k-1$ prediktorov z $\mathcal{M}_k$.
   b) Model s najmenším RSS (alebo najväčším R^2) označiť ako $\mathcal{M}_{k+1}$.
2. Spomedzi $\mathcal{M}_0, \ldots, \mathcal{M}_p$ vybrať najlepší model v zmysle jedného z kritérii: krížová validácia, C_p (AIC), BIC, korigovaný R^2.

- Podobne negarantuje nájdenie najlepšieho modelu. Nedá sa použiť v prípade $n < p$.
- Ďalšou alternatívou sú *hybridné* prístupy, ktoré riešia nevýhodu predošlých dvoch a pri pridávaní nových premenných do modelu môžu niektoré neproduktívne aj uberať.

4.2.3 Kritériá výberu optimálneho modelu

- Všetky spomenuté metódy výberu podmnožiny prediktorov majú jeden krok spoločný: výber modelu, ktorý je optimálny z hľadiska testovacej chyby. Sú dva prístupy k jej odhadu:
  - jeden ju nepriamo nahrádza kritériami, v ktorých je trénovacia chyba korigovaná o stupne voľnosti,
  - druhý priamo odhaduje testovaciu chybu pomocou metódy validačnej sady alebo krížovej validácie.

4.2.3.1 Korekcia trénovacej chyby

- Označme $k \leq p$ ako počet prediktorov použitých v posudzovanom modeli.
- Prvým z nepriamych odhadov testovacej chyby, ktorý preberieme, je *Mallowovo* C_p, dané vzťahom

$$C_p = \frac{1}{n}(RSS + 2k\hat{\sigma}^2)$$

kde $\hat{\sigma}^2$ je odhad rozptylu chybovej zložky (zväčša sa odhadne pomocou plného modelu s p prediktormi). Dá sa ukázať, že ak je $\hat{\sigma}^2$ nevychýleným odhadom σ^2, potom je aj C_p nevychýleným odhadom testovacej MSE.

- *Akaikeho informačné kritérium* je zvyčajne definované pomocou funkcie vierohodnosti, ale ak je splnený prepoklad normality chybovej zložky, potom sú metódy odhadu pomocou maximálnej vierohodnosti a najmenších štvorcov totožné a kritérium je dané rovnakým vzťahom ako C_p (až na násobnú konštantu). V prípade MNŠ sa zvykne používať aj druhý vzťah pre AIC, ktorý explicitne neobsahuje odhad σ^2).
- *Bayesovské informačné kritérium* bolo odvodené z bayesovského pohľadu, ale jeho tvar je nakoniec veľmi podobný AIC (a C_p až na násobnú konštantu),

$$BIC = \frac{1}{n}\left(RSS + \ln(n)k\hat{\sigma}^2\right)$$

pričom je zjavné, že pre $n > 7$ je penalizácia počtu prediktorov silnejšia než pre AIC.

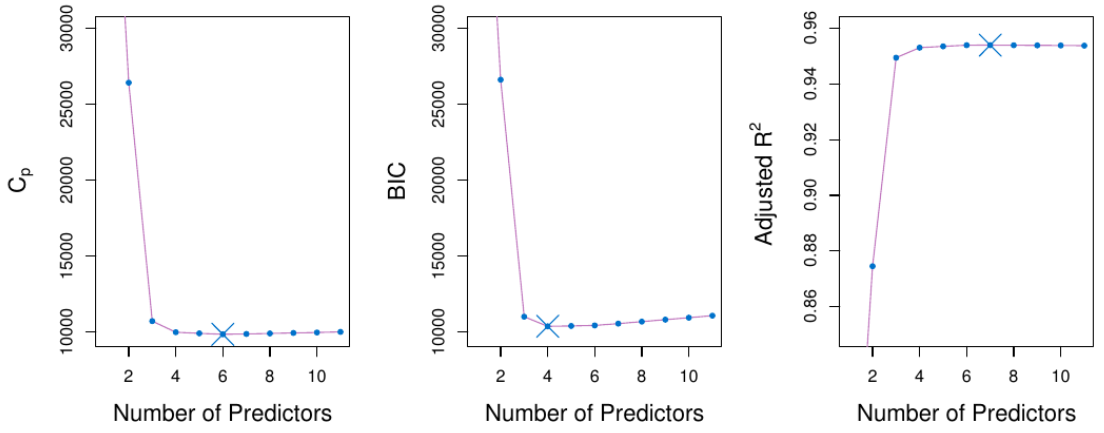
- Ďalšou, softvérmi často reportovanou mierou je korigovaný (adjusted) R^2,

$$adjR^2 = 1 - \frac{RSS/(n-k-1)}{TSS/(n-1)}$$

ktorá – na rozdiel od predošlých – s presnosťou modelu *rastie*. A zatiaľčo RSS s počtom prediktorov vždy klesá, korekcia $RSS/(n-k-1)$ môže aj rásť a tak penalizovať prebytočné prediktory.

- C_p, AIC a BIC majú za sebou starostlivé teoretické zdôvodnenie spoliehajúce na asymptotické (t.j. $n \to \infty$) vlastnosti odhadov.

Príklad (kreditky)



4.2.3.2 Validácia a krížová validácia

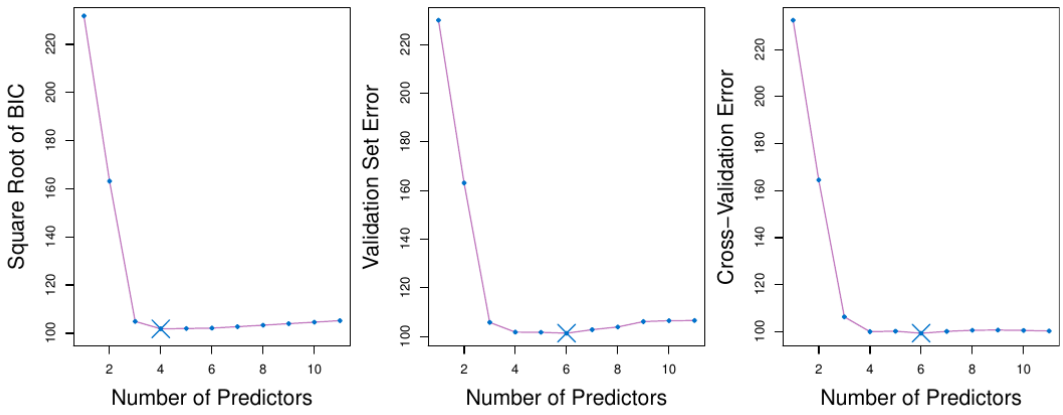
- Oproti predošlému prístupu má tento niekoľko výhod:

  - priamy odhad testovacej chyby,
  - menej predpokladov o skutočnom modeli,
  - využitie v širšom spektre úloh (nevyžaduje určenie stupňov voľnosti ani odhad σ^2)

- Nevýhoda v podobe výpočtovej náročnosti bola z veľkej časti odstránená modernými počítačmi.

Príklad (kreditky)

- Príklad s určením počtu prediktorov pre modelovanie dlhu držiteľov kreditných kariet.



- Priebeh testovacej chyby (priamym či nepriamym odhadom) je od modelu s tromi prediktormi veľmi stabilný, plochý, takže určenie minima kolíše od metódy ku metóde medzi hodnotami 4 až 7, a aj zopakovaním krížovej validácie by sme pravdepodobne dostali inú než aktuálne zobrazenú hodnotu 6.

- Aby v prípade plochého priebehu testovcej chyby (ako v príklade s kreditkami) nebol náhodne zvolený zbytočne zložitý model, môžeme s krížovou validáciou použiť *pravidlo jednej štandardnej chyby*:

  1. pre každý model sa spolu s odhadom MSE vypočíta aj jeho štandardná chyba $SE(\widehat{MSE})$,
  2. vyberie sa taký *najmenší model*, ktorý je ešte v dosahu jednej štandardnej chyby od minima.

- (V príklade s kreditkami by to znamenalo výber modelu s tromi prediktormi.)

4.3 Redukcia parametrov (regularizácia)

- Metódy výberu podmnožiny prediktorov používali MNŠ na odhad jednotlivých modelov s danou sadou premenných.
- Alternatívnym prístupom je odhad plného modelu (obsahujúceho všetkých p prediktorov) so zavedením podmienok na parametre, ktoré by stláčali ich odhady smerom k nule.

4.3.1 Hrebeňová (ridge) regresia

- Ridge regresia rozširuje MNŠ o penalizačný člen so súčtom štvorcov parametrov modelu, takže ich odhadom

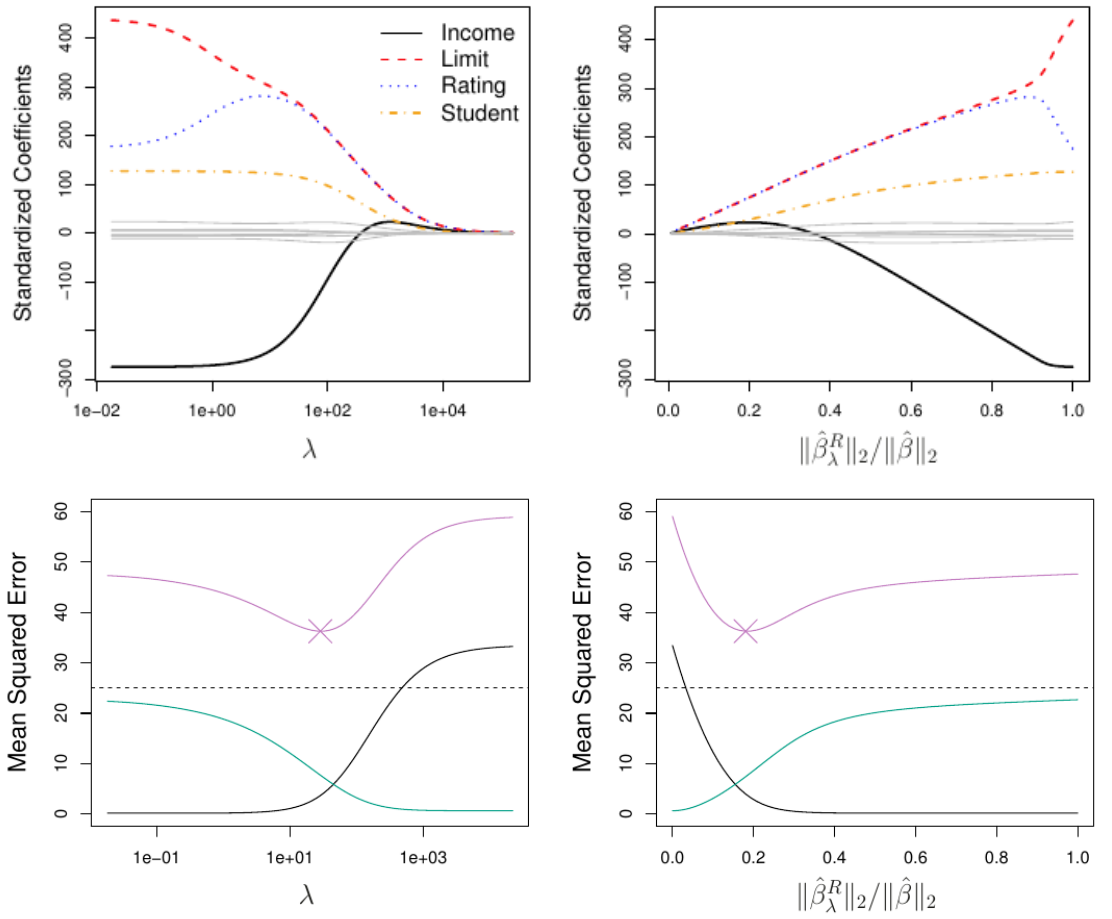
$$\hat{\beta}^R = \arg\min_{\beta} \left\{ \underbrace{\sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2}_{RSS} + \lambda \underbrace{\sum_{j=1}^{p} \beta_j^2}_{\lambda ||\beta||_2^2} \right\}$$

  nedovolí narásť do zbytočne veľkých hodnôt, resp. stlačí vplyv menej dôležitých prediktorov smerom k nule.
- Parameter λ určuje pomer vplyvu tých dvoch členov na odhad regresných koeficientov:
  - keď $\lambda = 0$, potom je metóda totožná s MNŠ,
  - a $\lambda \to \infty$ spôsobí stlačenie odhadu koeficientov až takmer na nulu.
- Správny výber λ je pre úspech metódy kritický a v praxi sa naň používa krížová validácia.
- V maticovom tvare sa odhad dá vyjadriť vzťahom

$$\hat{\beta}^R = (\mathbf{X}'\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}'\mathbf{y}$$

  z ktorého je vidieť, že regularizácia je spôsobená "hrebeňom" (ridge) lambda parametrov na diagonále penalizačnej matice $\lambda\mathbf{I}$.
- Ridge regresia ako regularizačná technika dokáže spresniť odhad MNŠ preto, že prenáša problém bias-variance kompromisu na problém voľby λ. Hodí sa na prípady, kedy je p porovnateľne veľké ako n, takže odhady MNŠ majú veľký rozptyl (t.j. drobné zmeny v dátach vyvolajú veľké zmeny v odhadoch). Ridge regresia dokáže tento rozptyl významne znížiť – za cenu iba malého nárastu výchylky.

Príklad (kreditky)



- Oproti metóde hrubej sily má výpočtovú výhodu: namiesto množstva modelov odhaduje iba jeden (pre dané λ).

4.3.2 LASSO

- Hrebeňová regresia má však aj jednu podstatnú nevýhodu: vplyv irelevantných premenných nikdy nie je stlačený úplne na nulu, takže na rozdiel od metód výberu podmnožiny prediktorov, finálny model z hrebeňovej regresie bude vždy obsahovať všetkých p prediktorov.
- To komplikuje najmä interpretáciu modelu.
- Alternatíva v podobe metódy LASSO (least absolute shrinkage and selection operator)

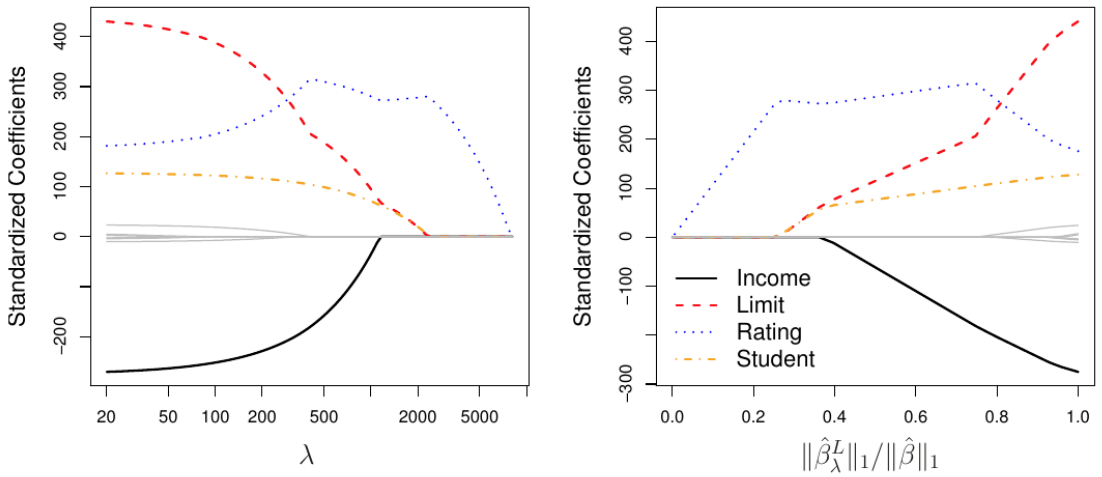
túto nevýhodu odstraňuje a to malou modifikáciou penalizačného člena,

$$\hat{\beta}^L = \arg\min_{\beta} \left\{ \underbrace{\sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2}_{RSS} + \lambda \underbrace{\sum_{j=1}^{p} |\beta_j|}_{\lambda ||\beta||_1} \right\}$$

(čiže zmenou normy ℓ_2 na ℓ_1).

- Dôsledkom je to, že hodnoty parametrov môžu klesnúť úplne na nulu, ak je λ dostatočne veľké.
- LASSO teda môžeme zaradiť aj medzi metódy výberu premenných.

Príklad (kreditky)



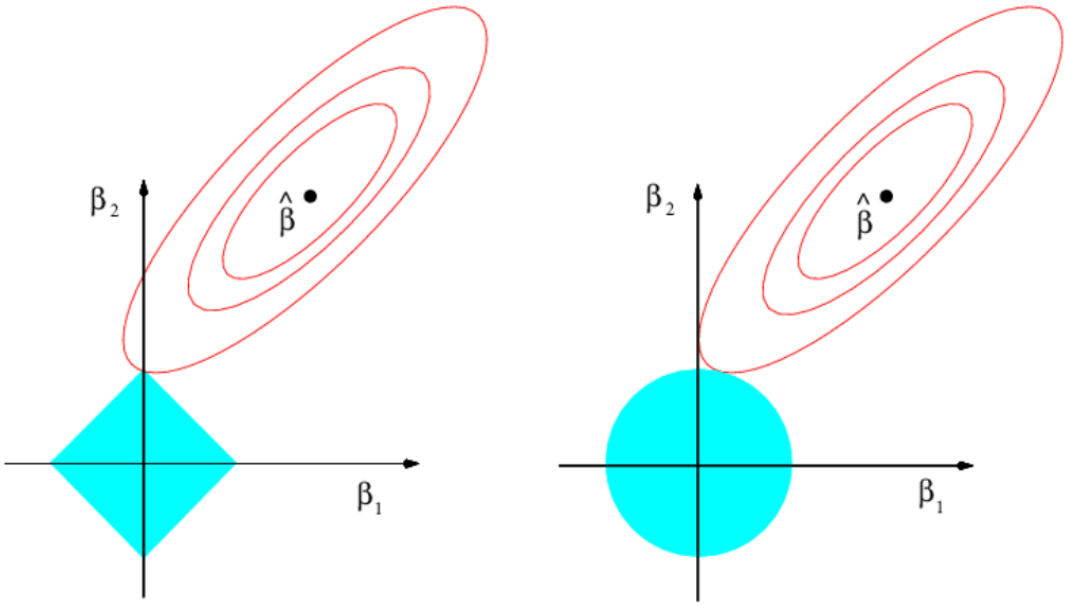
4.3.3 Porovnanie

- Odhad LASSO aj ridge regresie sa dá preformulovať ako problém hľadania minima funkcie $RSS(\beta)$ na oblasti danej hranicou
  - $||\beta||_1 \leq s$ pre LASSO,
  - $||\beta||_2 \leq s$ pre ridge regresiu,

  kde úroveň s môžeme interpretovať ako rozpočet na prerozdelenie medzi parametre.
- Ak je rozpočet dostatočne veľký na to, aby pokryl odhad MNŠ, tak minimum je práve v ňom (to je prípad $\lambda = 0$). V opačnom prípade globálne minimum leží niekde na hranici.

Ilustrácia (LASSO/ridge)

- Z geometrického hladiska hranica má v dvojrozmernom prípade tvar štvorca (LASSO) resp. kruhu (ridge). Na obrázku dole je spolu s izočiarami RSS.



- Vo svetle poslednej formulácie sú obe metódy podobné metóde výberu najlepšej podmnožiny (t.j. hrubej sily), ktorá minimum $RSS(\beta)$ hľadá na oblasti $\sum_{j=1}^{p} I(\beta_j \neq 0) \leq s$, kde I značí indikačnú funkciu. V pôvodnej formulácii zodpovedá penalizačnému členu $\lambda \sum_{j=1}^{p} |\beta_j|^0$ s konvenciou $0^0 = 0$.
- Aby odhady okrem ladiaceho (tuning) parametra λ nezáviseli aj od mierky jednotlivých parametrov, je potrebné *hodnoty prediktorov vopred vydeliť štandardnou odchýlkou* (aby mali jednotkový rozptyl), čiže v regresii použiť pozorovania $\tilde{x}_{ij} = x_{ij}/\hat{\sigma}_{X_j}$.
- Ak okrem škálovania na jednotkový rozptyl hodnoty prediktorov aj centrujeme (aby mali nulovú strednú hodnotu) – spolu sa táto transformácia nazýva *štandardizácia* – potom $\hat{\beta}_0 = \bar{y}$ a centrovaním odozvy možno zmenšiť priestor hľadaných parametrov o β_0.
- Štandardizáciu robí softvér automaticky.
- Ani jedna z regularizačných metód nie je vo všeobecnosti lepšia, než tá druhá. Ridge je vhodnejšia v prípade, že každý prediktor má aspoň minimálny vplyv na odozvu, zatiaľčo LASSO predpokladá absenciu vplyvov niektorých premenných.
- V prípade korelovaných prediktorov ridge stláča parametre tejto skupiny spoločne, zatiaľčo LASSO má tendenciu uprednostniť jeden prediktor pred ostatnými.

- Z výpočtového hľadiska sú vďaka efektívnym algoritmom obe metódy podobne náročné ako MNŠ.
- LASSO aj hrebeňová regresia sú špeciálnymi prípadmi tzv. *elastickej siete* s penalizačným členom

$$\lambda \sum_{j=1}^{p} \left(\frac{1-\alpha}{2} \beta_j^2 + \alpha |\beta_j| \right)$$

ktorá spája výhody oboch [1]. Napr. v prípade skupiny korelovaných prediktorov nastavenie $\alpha = 0.5$ má tendenciu vybrať alebo vynechať celú skupinu.

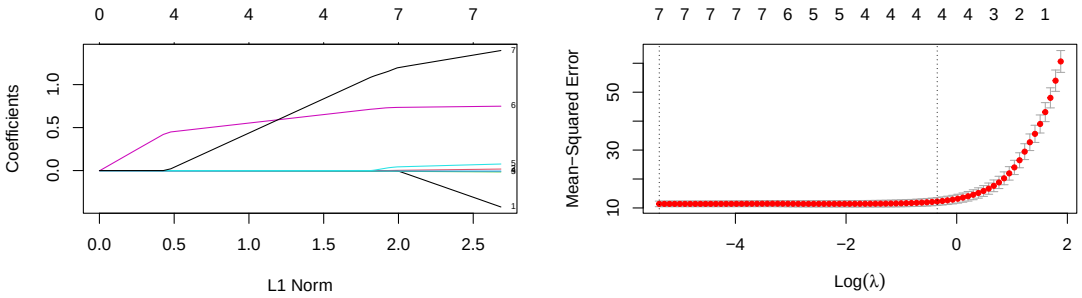
- Parameter α má aj stabilizačnú funkciu: napr. elastická sieť s α blízkou 1 sa správa ako LASSO, ale zároveň odstraňuje anomálie spôsobené extrémnymi koreláciami medzi prediktormi.

Príklad (autá)

- Súbor údajov *ISLR2::Auto* obsahuje niekoľko vlastností 392 automobilov: dojazd (*mpg*), počet valcov (*cylinders*), zdvihový objem (*displacement*), výkon (*horsepower*), hmotnosť (*weight*), zrýchlenie (*acceleration*), rok uvedenia modelu (*year*), pôvod (*origin*) a nakoniec názov modelu (*name*).
- Nech dojazd je odozva, potom regularizačnou metódou elastická sieť (prakticky LASSO) odhadnime lineárny regresný model so všetkými premennými zo súboru okrem názvu modelu.
- Funkcia *glmnet* z balíku *glmnet* odhadne model na predvolenej mriežke hodnôt λ. Na grafe vľavo je namiesto nich na osi x zobrazený počet nenulových parmetrov (hore) a ich súčty v absolútnej hodnote (dole).

```
X <- model.matrix(mpg ~ . - name, data = ISLR2::Auto)[,-1]  # without ones
y <- ISLR2::Auto$mpg
glmnet::glmnet(x = X, y = y, alpha = 0.99) |>
  plot(label = TRUE)
set.seed(12)
cvfit <- glmnet::cv.glmnet(x = X, y = y, alpha = 0.99)
plot(cvfit)
```

[1] Uvedená parametrizácia je často implementovaná v softvéroch, ako napr. v balíku *glmnet* systému R.



- Vhodný počet parametrov (resp. vhodnú hodnotu λ) je objektívnejšie zvoliť minimalizáciou testovacej chyby (odhadu pomocou krížovej validácie). Podľa grafu vpravo by sme mali zvoliť model so siedmimi prediktormi (minimum indikované zvislou čiarou vľavo), ale uplatnením pravidla jednej štandardnej chyby uprednostníme model so 4 prediktormi.

```
cvfit[c("lambda.min","lambda.1se")] |> unlist()   # column left
fit <- glmnet::glmnet(x = X, y = y, alpha = 0.99, lambda = cvfit$lambda.1se)
coef(fit)                                         # column right
```

```
 lambda.min  lambda.1se
0.004622949 0.702647151
```

```
8 x 1 sparse Matrix of class "dgCMatrix"
                      s0
(Intercept)  -6.459341529
cylinders     .
displacement  .
horsepower   -0.006472496
weight       -0.005411354
acceleration  .
year          0.601425571
origin        0.633118032
```

- Parametre sú v pôvodnej mierke prediktorov, takže sa z nich priamo dajú vypočítať predikcie. Napr. pre model s výkonom 100 koní, váhou 3000 libier, rokom uvedenia (19)80 a pôvodom z USA.

```
list(horsepower = 100, weight = 3000, year = 80, origin = 1) |>
  with(c(
    manual = -6.46 - 0.00647*horsepower - 0.00541*weight + 0.601*year + 0.633*origin,
    built_in = fit |>
      predict(newx = cbind(NA, NA, horsepower, weight, NA, year, origin)) |> c(),
    lm = lm(mpg ~ horsepower + weight + year + origin, ISLR2::Auto) |>
      predict(newdata = as.list(environment())) |> unname()
  )) |>
  rbind(prediction = _)   # just a cosmetics
```

```
           manual built_in       lm
prediction 25.376 25.40651 25.65319
```

- Ak by rovnaký model auta bol vyvinutý v Japonsku, podľa modelu by mal stredný dojazd dlhší o $2 \cdot 0.63 = 1.26$ míle na galón paliva. Tu treba poznamenať, že premenná *origin* je nesprávne považovaná za numerickú premennú. Vopred mala byť prevedená na dátový typ *factor*.
- Na záver ešte porovnajme výber premenných s VIF faktorom všetkých pôvodných. Vidno, že (aspoň v tomto prípade) LASSO viedlo k vylúčeniu časti premenných v multikolinearite s ostatnými, pričom vo výslednom modeli sú prediktory oveľa menej prepojené.

```
list(
  full = lm(mpg ~ . - name, ISLR2::Auto) |> car::vif(),
  selected = lm(mpg ~ horsepower + weight + year + origin, ISLR2::Auto) |> car::vif()
) |>
  lapply(round, dig = 1)
```

```
$full
   cylinders displacement   horsepower       weight acceleration         year
        10.7         21.8          9.9         10.8          2.6          1.2
      origin
         1.8

$selected
horsepower     weight       year     origin
       4.5        4.9        1.2        1.5
```

5 Za hranicami linearity

5.1 Úvod

- Lineárny regresný model, na ktorý sme sa dosiaľ zameriavali, má výhodu v jednoduchej interpretácii. Ale ak modelovaný vzťah nie je ani približne lineárny, jeho predpovedná schopnosť je slabá.
- V predošlej kapitole sme videli, ako sa dá presnosť lineárneho modelu zlepšiť redukciou komplexnosti (zásadne znížiť varianciu na úkor malého zvýšenia výchylky) a tým zlepšiť aj jeho interpretovateľnosť.
- V druhej kapitole sme zas predstavili elegantný spôsob ako model pomocou bázových funkcií rozšíriť o nelinearitu v premenných pri zachovaní linearity v parametroch. Tým sa jednak zvýšila presnosť aproximácie nelineárnych vzťahov, jednak interpretovateľnosť ako-tak udržala.
- V nasledujúcej časti si ukážeme jednu užitočnú triedu bázových funkcií, ktorá vedie ku tzv. regresným splajnom.
- Potom sa na splajny pozrieme z opačného pohľadu ako na neparametrickú triedu vyhladzovacích metód.
- Nakoniec oba prístupy využijeme v jednoduchom rozšírení do priestoru viacerých prediktorov pri zachovaní aditivity.

5.2 Regresné splajny

- Najpopulárnejšími bázovými funkciami v regresii sú mocninové funkcie, ktoré vedú k polynomickej regresii

$$Y = \beta_0 + \beta_1 X + \beta_2 X^2 + ... + \beta_q X^q + \varepsilon.$$

  Hoci model umožňuje vystihnúť až extrémne nelineárny priebeh funkcie, v praxi sa nepoužíva stupeň q väčší ako 3 alebo 4, pretože pri väčšom vznikajú čudesné tvary – obzvlášť na hraniciach hodnôt X, kde je variancia predikcií najväčšia.
- Týmto spôsobom však model nelineárnej funkcii vnucuje globálnu štruktúru.
- Namiesto toho môžeme rozsah prediktoru X rozbiť na intervaly (bins) ohraničené K uzlami (knots) a na každom zvlášť odhadnúť nejaký polynóm.
- Taká trieda lineárnych modelov sa nazýva *regresné splajny* a v nasledujúcich častiach sú popísané najčastejšie voľby bázových funkcií.

5.2.1 Konštantné

- Najjednoduchším regresným splajnom je po častiach konštantná funkcia (*step function*)

$$Y = \beta_0 + \beta_1 \mathbf{1}_{[\xi_1,\xi_2)}(X) + \beta_2 \mathbf{1}_{[\xi_2,\xi_3)}(X) + ... + \beta_K \mathbf{1}_{[\xi_K,\infty)}(X) + \varepsilon$$

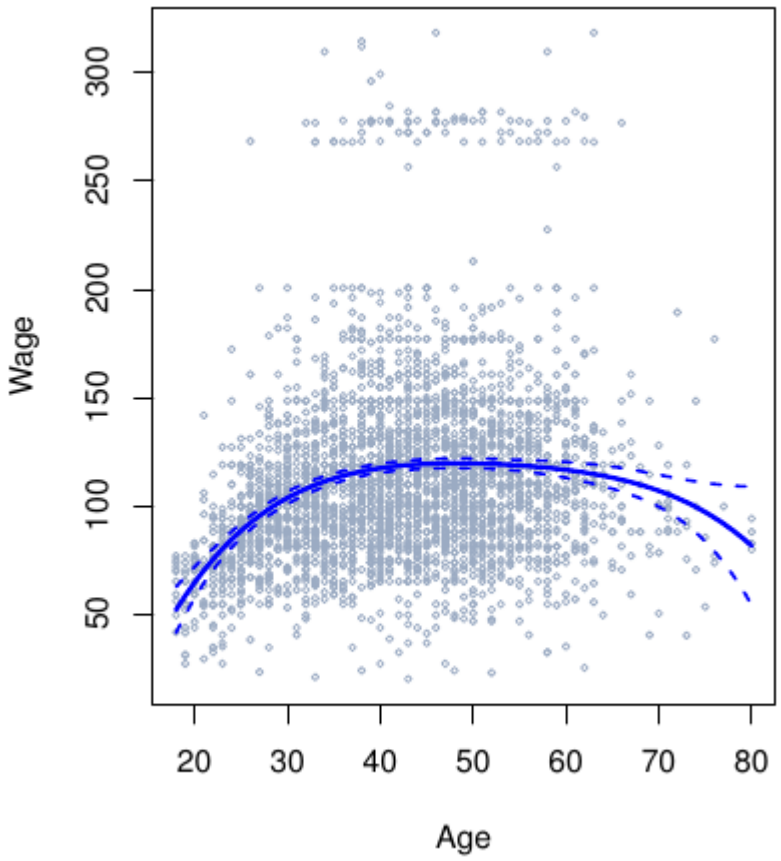
  kde ako bázová funkcia pre jednotlivé intervaly je použitá indikačná funkcia

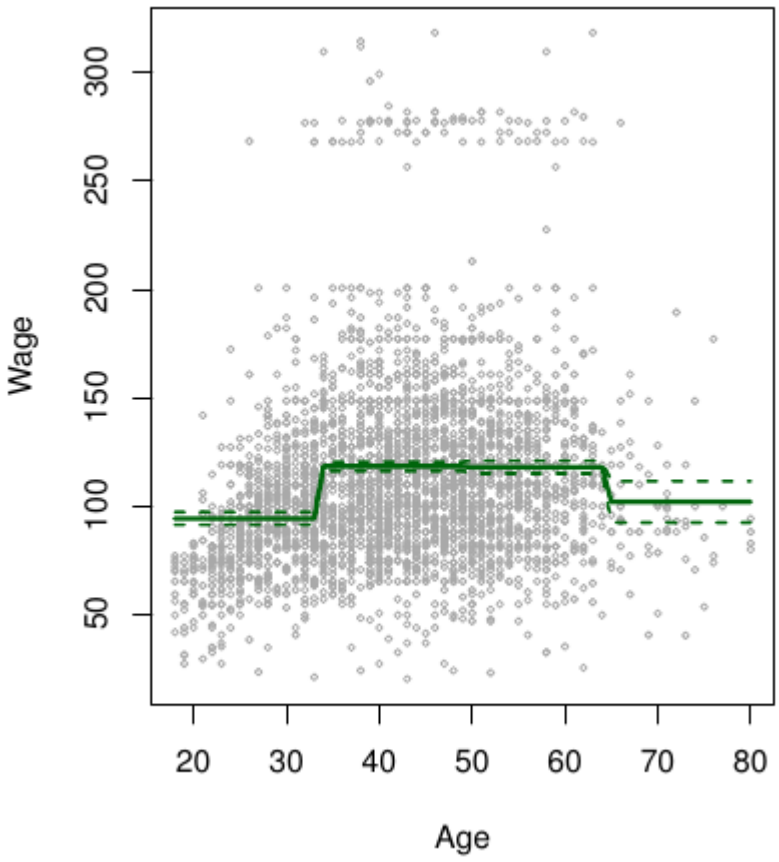
$$\mathbf{1}_{[a,b)}(X) = \begin{cases} 1 & \text{ak } a \leq X < b \\ 0 & \text{inak} \end{cases}$$

- Prvý parameter je interpretovaný ako stredná hodnota pre prvý (referenčný) interval, $\beta_0 = E(Y|X < \xi_1)$, a ďalšie parametre ako priemerný prírastok odozvy ku β_0 pre zodpovedajúci interval.
- Samozrejme, ako referenčný interval môže byť zvolený ktorýkoľvek iný než prvý. No takisto nemusí byť žiaden, ak prvú bázovú funkciu zmeníme z $b_0(X) = 1$ na $b_0(X) = \mathbf{1}_{(-\infty,\xi_1)}(X)$.
- Takýmto konštantným splajnom sa spojitý prediktor X prakticky nahradil ordinálnou kategorickou premennou.
- Voľba uzlov musí byť urobená vopred. Najčastejšie
  - rovnomerným rozmiestnením v rozsahu premennej X alebo
  - prirodzeným umiestnením do bodov, kde dochádza ku zmene priebehu skutočnej regresnej funkcie.

Príklad (mzda)

- Súbor údajov *ISLR::Wage* obsahuje príjem (v tisícoch USD) a demografické údaje mužov stredovýchodu USA.
- Modelujme príjem pomocou veku pomocou globálneho polynómu 4. stupňa (obrázok vľavo) a konštantným regresným splajnom s rovnomerným rozmiestnením uzlov (obr. vpravo).


Wage
50
100
150
200
250
300
20
30
40
50
60
70
80
Age



- Hoci stupňovitá funkcia nemá v tomto prípade takú dobrú aproximačnú schopnosť ako polynomická vyššieho stupňa, interpretácia parametrov je oveľa ľahšia: priemerný ročný plat mužov do 34 rokov je 94,000 dolárov ($\pm 1,500$), pričom napr. muži v seniorskom veku majú iba o $7,600$ USD vyšší (s pomerne vysokým rozptylom).

```
lm(wage ~ cut(age, 4), ISLR2::Wage) |>
  summary() |>
  coef() |>
  signif(2)
```

```
                        Estimate Std. Error t value Pr(>|t|)
(Intercept)                 94.0        1.5    64.0  0.0e+00
cut(age, 4)(33.5,49]        24.0        1.8    13.0  2.0e-38
cut(age, 4)(49,64.5]        24.0        2.1    11.0  1.0e-29
cut(age, 4)(64.5,80.1]       7.6        5.0     1.5  1.3e-01
```

5.2.2 Lineárne

- Ak konštantný priebeh v jednotlivých intervaloch na popis vzťahu nestačí, polynóm nultého stupňa môžme nahradiť lineárnym,

$$\begin{aligned} Y &= \beta \cdot \left\{ (1, X) \otimes \left(1, \mathbf{1}_{[\xi_1,\xi_2)}(X), \mathbf{1}_{[\xi_2,\xi_3)}(X), ..., \mathbf{1}_{[\xi_K,\infty)}(X) \right) \right\} + \varepsilon \\ &= \beta_0 + \beta_1 \mathbf{1}_{[\xi_1,\xi_2)}(X) + ... + \beta_K \mathbf{1}_{[\xi_K,\infty)}(X) + \\ &\quad \beta_{K+1} X + \beta_{K+2} X \mathbf{1}_{[\xi_1,\xi_2)}(X) + ... + \beta_{2K+1} X \mathbf{1}_{[\xi_K,\infty)}(X) + \varepsilon \end{aligned}$$

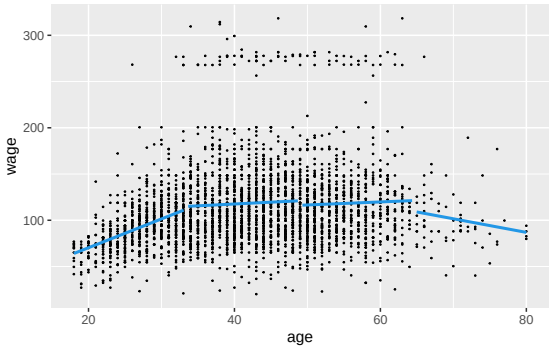
  (kde $\otimes$ je vonkajší súčin vektorov, krútené zátvorky značia rozloženie matice do vektora po riadkoch) a výsledná funkcia bude po častiach lineárna, no vo všeobecnosti stále nespojitá.
- Spojitosť sa dá zaviesť K podmienkami (constraints) na uzloch splajnu, napr. v regresnej rovnici

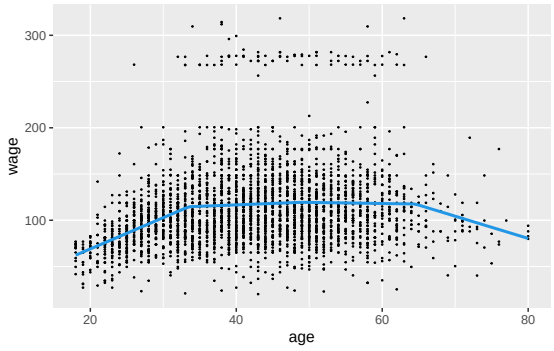
$$\begin{aligned} Y &= \beta_0 + \beta_1 X + \beta_2 (X - \xi_1)_+ + ... + \beta_{K+1} (X - \xi_K)_+ + \varepsilon \\ &= \begin{cases} \beta_0 + \beta_1 X & \text{ak} \quad X < \xi_1 \\ (\beta_0 - \beta_2 \xi_1) + \beta_2 X & \xi_1 \leq X < \xi_2 \\ \vdots & \\ (\beta_0 - \sum_{j=1}^{K} \beta_{j+1} \xi_j) + \beta_{K+1} X & \xi_K \leq X \end{cases} + \varepsilon \end{aligned}$$

  pomocou bázovej funkcie $(X - \xi_j)_+ = \max(0, X - \xi_j)$, ktorá centruje okolo uzlu a zdola orezáva nulou.
- Zavedením K podmienok v lineárnom splajne klesol počet parametrov o K.
- Interpretácia parametrov je stále pomerne jednoduchá, sklon lineárneho segmentu v j-tom intervale sa rovná β_j a (pomyselný) priesečník s vertikálnou osou je daný všetkými predošlými parametrami až po β_j.

Príklad (mzda)

```r
dat <- ISLR2::Wage  # |> dplyr::slice_sample(n = 100)
grid <- modelr::seq_range(dat$age, 100) |>
  data.frame(age = _)

# --- discontinuous ---

# function to discretize continuous variable by breaking into bins
# in: x - numeric vector to be discretized
#     knots - numeric vector of breaks (without extremes)
#           - single integer, number of breaks
# out: character vector indicating intervals
make_ind <- function(x, knots) {
  min <- floor(min(x)); max <- ceiling(max(x))
  if(length(knots) == 1 & is.integer(knots)) {
    cut(x, knots + 1, right = FALSE, include.lowest = TRUE)
  } else {
    cut(x, c(min, knots, max), right = FALSE, include.lowest = TRUE)
  }
}
# example use:
# make_ind(1:10, 3)
# make_ind(1:10, 3L)

fit1 <- dat |>
  dplyr::mutate(ind = make_ind(age, 3L) |> as.factor()) |>
  lm(wage ~ ind*age, data = _)

grid |>
  transform(ind = make_ind(age, 3L)) |>
  modelr::add_predictions(fit1, var = "wage") |>
  ggplot() + aes(x = age, y = wage) +
  geom_point(data = dat, size = 0.2) +
  geom_line(aes(group = ind), linewidth = 1, color = 4)
# --- continuous ---

# function to round a vector as much as possible while keeping elements distinguishable
# in: x - numeric vector
#     add - integer, number of digits by which to increase precision (or decrease if negati
# out: numeric vector
round_but_keep_different <- function(x, add = 0) {
  x |> diff() |> abs() |> min() |> log10() |> (`-`)() |> ceiling() |> (`+`)(add) |>
    round(x, digits = _)
}
# example use:
# c(1.2345, 1.2456, 1.2356) |> round_but_keep_different(add=1)   # (1.23 1.25 1.24)
# c(1.2345, 1.2456, 1.2356) |> round_but_keep_different(add=1)   # (1.234 1.246 1.236)

# function to apply truncated linear basis function to predictor
# in: x - numeric vector, values of predictor
#     knots - numeric vector of breaks (without extremes)
#           - single integer, number of interior breaks
# out: data frame, one column for each knot
```





	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	8	10	0.79	0.43
ind[33.5,49)	94	15	6.3	2.7e-10
ind[49,64.5)	92	20	4.5	7e-06
ind[64.5,80.1]	200	81	2.4	0.015
age	3.1	0.36	8.6	9.3e-18
ind[33.5,49):age	-2.7	0.45	-6.1	1.1e-09
ind[49,64.5):age	-2.8	0.49	-5.7	1e-08
ind[64.5,80.1]:age	-4.6	1.2	-3.8	0.00014

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	1.4	8.3	0.17	0.86
age	3.4	0.28	12	7.9e-33
max(34)	-3.1	0.4	-7.7	1.8e-14
max(49)	-0.43	0.36	-1.2	0.24
max(64)	-2.3	0.91	-2.5	0.012

5.2.3 Kubické

- Lineárna lomená funkcia je výhodná najmä vtedy, keď v jednotlivých intervaloch potrebujeme nadväzujúce a ľahko interpretovateľné modely.
- Hladký, po častiach definovaný polynóm dosiahneme zvýšením jeho stupňa. Kvadratický síce zabezpečuje spojitosť aj v prvej derivácii, ale až kubický spojitý splajn so spojitými druhými deriváciami v uzloch je vizuálne dokonale hladký. (Spojitosť vyššieho stupňa už ľudský zrak nepostrehne.)
- Spojitosť do m-tej derivácie v uzle $X = \xi_j$ sa dá dosiahnuť pridaním bázovej funkcie $(X - \xi_j)_+^m = \max(0, X - \xi_j)^m$, tzv. truncated-power basis do polynomickej regresnej rovnice m-tého stupňa.
- Spojitý kubický splajn s K uzlami má teda tvar

$$Y = \beta_0 + \beta_1 X + \beta_2 X^2 + \beta_3 X^3 + \beta_4 (X - \xi_1)_+^3 + ... + \beta_{K+3} (X - \xi_K)_+^3 + \varepsilon$$

- Kubický splajn má však rovnakú nevýhodu ako globálny polynóm vyššieho stupňa: nepríjemné krútenie (a väčší rozptyl) na oboch koncoch.
- Riešením je tzv. *prirodzený splajn*, ktorý vznikne zavedením ďalších hraničných podmienok, a síce linearity na hraničných intervaloch (čiže pred prvým a za posledným uzlom).
- Definícia prirodzeného kubického splajnu je už komplikovanejšia,

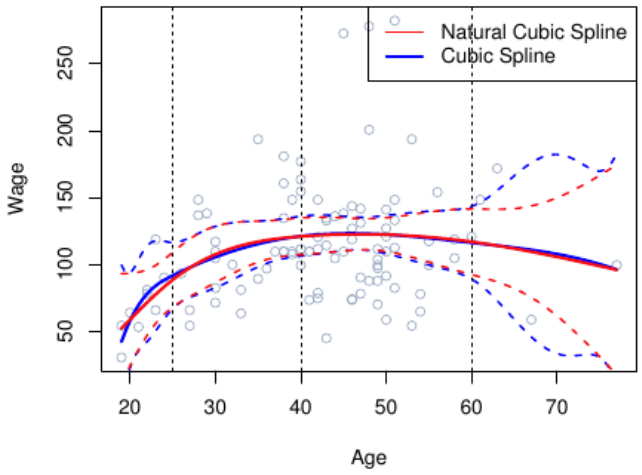
$$Y = \beta_0 + \beta_1 X + \sum_{k=1}^{K} \beta_{j+1}\Big(d_{j-1}(X) - d_K(X)\Big) + \varepsilon,$$

$$\text{kde} \quad d_j(X) = \frac{(X-\xi_j)_+^3 - (X-\xi_{K+1})_+^3}{\xi_{K+1} - \xi_j} \quad \text{pre} \quad j = 0, \dots, K$$

a ξ_0, ξ_{K+1} sú extrémy pozorovaných hodnôt X.

Príklad (mzda)

Aby bol rozdiel v šírke pásu spoľahlivosti medzi kubickým a prirodzeným kubickým splajnom zjavnejší, súbor údajov o mzdách bol náhodne zredukovaný.



- Funkcie truncated-power bázy nemajú nosič (support, nenulovú časť) definovaný lokálne, čo môže viesť ku korelácii medzi bázovými funkciami a tak aj ku numerickej nestabilite pri odhade. Preto sa v softvéroch používajú radšej iné – stabilnejšie, napr. B-spline.

5.2.4 Voľba počtu uzlov

- Regresný splajn je najohybnejší okolo uzlov, preto prirodzenou voľbou je umiestniť viac uzlov tam, kde je vzťah odozvy a prediktoru dynamickejší.
- V praxi sa však často uzly umiestňujú rovnomerne – do kvantilov rovnomerného rozdelenia – a to buď zadaním ich počtu alebo *stupňami voľnosti*.

- Kubický splajn má $K + 3 + 1$ stupňov voľnosti, zatiaľčo prirodzený kubický o 4 menej.
- V softvéroch zvyčajne počet stupňov voľnosti nezahŕňa absolútny člen.
- Samotná voľba počtu (uzlov alebo stupňov) je buď vizuálna alebo pomocou krížovej validácie.

5.3 Vyhladzovanie

5.3.1 Vyhladzovacie splajny

- Videli sme, že regresný splajn sa vytvorí voľbou množiny uzlov, následne vytvorením sekvencie bázových funkcií a nakoniec odhadom parametrov lineárneho modelu.
- Iný prístup vedie k tzv. vyhladzovaciemu splajnu (smoothing spline): hľadáme funkciu g, ktorá vystihuje pozorované dáta v zmysle nízkeho RSS a zároveň je hladká. (Ak by sme g nijak neobmedzili, minimalizácia RSS dôjde až ku nule a povedie ku interpolačnej funkcii.)
- Formálne teda hľadáme také g, ktoré minimalizuje výraz

$$\sum_{i=1}^{n}(y_i - g(x_i))^2 + \lambda \int g''(t)^2 dt$$

  pozostávajúci zo stratovej funkcie (loss function) RSS a členu penalizujúceho *krivosť* funkcie. Parameter λ reguluje kompromis výchylky a rozptylu (bias-variance).
- Dá sa dokázať, že *riešením* tejto optimalizačnej úlohy *je prirodzený kubický splajn s uzlami vo všetkých bodoch súboru pozorovaní*, $x_1, \ldots, x_n$.
- Treba však poznamenať, že nejde presne o regresný splajn, ale o jeho zrazenú verziu, pričom λ určuje úroveň zrazenia (shrinkage).
- S toľkými uzlami by mal prirodzený regresný splajn príliš veľa stupňov voľnosti, no parameter $\lambda \in [0, \infty)$ reguluje drsnosť funkcie a tým aj tzv. *efektívne stupne voľnosti*, $df_\lambda \in [n, 2)$.

5.3.2 Lokálna (vážená) regresia

- Iným prístupom k vystihnutiu nelineárneho vzťahu odozvy a prediktoru je obmedziť trénovaciu vzorku pre zvolený model na pozorovania z blízkeho okolia bodu x_0 (bodu predpovede). Presnejšie povedané, upraviť veľkosť vplyvu pozorovaní na odhad $g(x)$ podľa blízkosti ku x_0.
- Preto sa definuje váhová funkcia $K(x_i, x)$, nazývaná tiež jadrová (kernel) funkcia, ktorá najvzdialenejším pozorovaniam x_i priradí nízku hodnotu (až nulu), a naopak blízkym pozorovaniam vysokú váhu.

- Formálne, lokálne vážená regresná funkcia je definovaná ako odhad

$$\hat{g}(x) = \mathbf{b}(x) \cdot \hat{\beta}(x_0),$$

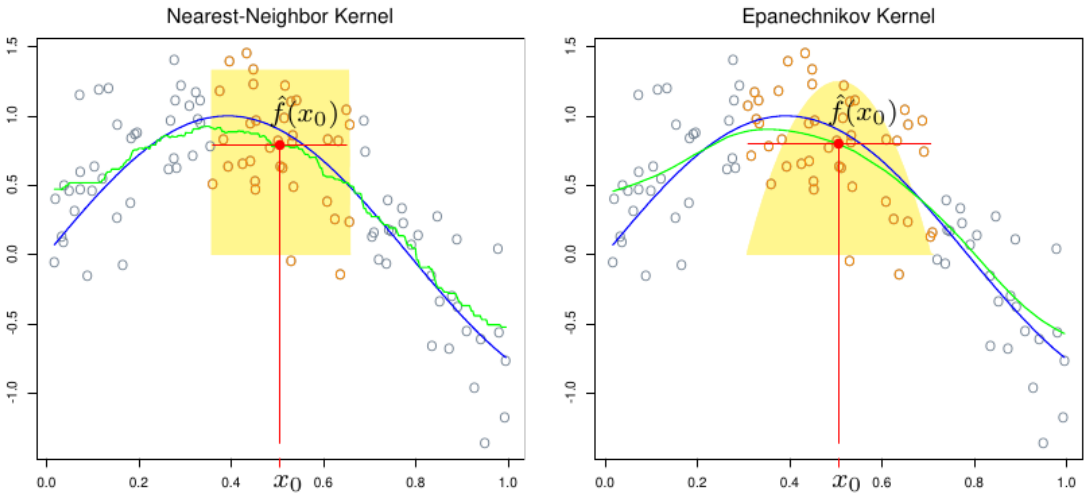
kde $\mathbf{b}(x)$ je vektor bázových funkcií a parametre β viazané na x_0 sú odhadnuté váženou metódou najmenších štvorcov,

$$\hat{\beta}(x_0) = \underset{\beta(x_0)}{\operatorname{argmin}} \sum_{i=1}^{n} \underbrace{\frac{K(x_i, x_0)}{\sum_{j=1}^{n} K(x_j, x_0)}}_{\text{normovaná váha}} \Big(y_i - g(x_i)\Big)^2.$$

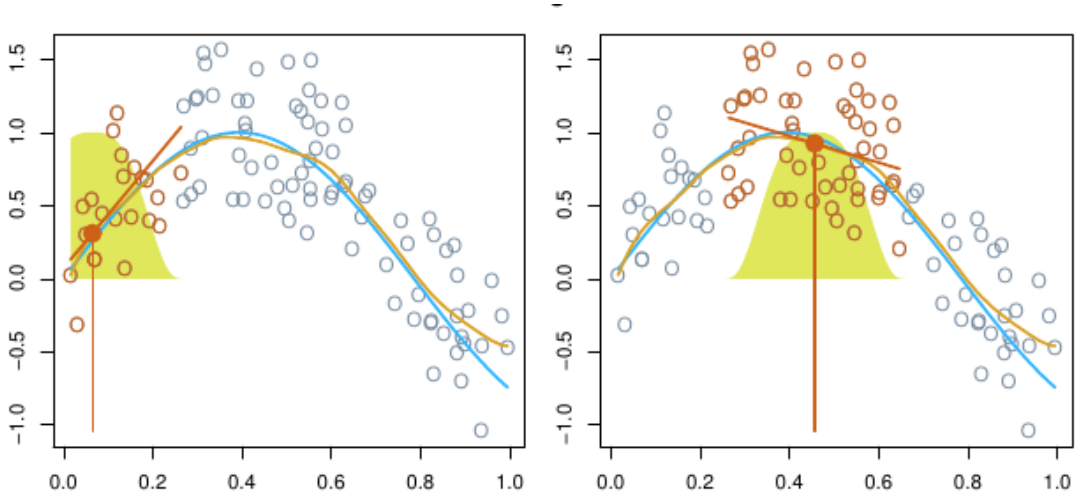
- To znamená, že pre výpočet každej jednej predikcie je nutné odhadnúť model nanovo. Preto lokálna regresia patrí medzi metódy založené na pamäti (*memory-based*).
- Pri lokálnej regresii je potrebné zvoliť:
  - bázové funkcie (zvyčajne mocninové po stupeň 2),
  - typ váhovej funkcie K,
  - miera dosahu váhovej funkcie (šírka pásma, bandwidth) h.
- Jadrová funkcia funkcia sa najčastejšie volí ako
  - hustota rovnomerného rozdelenia $U(-h/2, h/2)$,
  - hustota normálneho rozdelenia $N(0, h)$,
  - indikačná (napr. pre množinu určitého počtu najbližších susedov),
  - trikubická, alebo tzv. Epanechnikova.
- Ak je napr. jadrová funkcia definovaná tak, že priradí nenulovú a rovnakú váhu iba k najbližším pozorovaniam, a g je konštantná funkcia, potom dostávame tzv. regresnú metódu *k-najbližších susedov*.

Ilustrácia (lokálna regresia)

- Prvá dvojica obrázkov ilustruje voľbu *konštantnej* funkcie g, vľavo s indikačnou jadrovou funkciou (regresná metóda k-najbližších susedov), vpravo s jadrovou funkciou, ktorá zvýhodňuje bližšie pozorovania. Zelenou farbou je odhad $\hat{g}$, modrou funkcia, z ktorej boli generované dáta.



- Na ďalších dvoch obrázkoch je znázornená predikcia v dvoch bodoch pomocou *lineárnej* funkcie g a jadrovej funkcie gaussovského typu. Modrou farbou je skutočná regresná funkcia, z ktorej boli generované pozorovania, oranžovou je zobrazený odhad $\hat{g}(x)$.



5.3.3 Trieda lineárnych vyhladzovacích modelov

- Všetky spomenuté metódy (od globálnych regresných funkcií, cez splajny po lokálnu regresiu) patria do triedy lineárnych vyhladzovacích modelov (*linear smoothers*), kde

deterministická časť je daná funkciou

$$\begin{aligned} f(x) &= \sum_{i=1}^{n} w(x_i, x) y_i \\ &= \mathbf{w}(x) \cdot \mathbf{y} \end{aligned}$$

a váhy sú funkciou prediktoru. (Či už ako skalárna funkcia w alebo vektorová $\mathbf{w}$.)

Ilustrácia (lineárna regresia)

- Napr. lineárny regresný model (s odhadom parametrov metódou najmenších štvorcov) má podmienenú strednú hodnotu v tvare

$$\begin{aligned} \hat{f}(x) &= \hat{\beta}_1 + \hat{\beta}_2 x \\ &= \bar{y} + \hat{\beta}_2 (x - \bar{x}) \\ &= \frac{1}{n} \sum_{1=n} y_i + \left(\frac{\frac{1}{n} \sum_i (y_i - \bar{y})(x_i - \bar{x})}{\frac{1}{n} \sum_i (x_i - \bar{x})^2} (x - \bar{x}) \right) \\ &= \frac{1}{n} \sum_{1=n} y_i + \frac{(x - \bar{x})}{n \hat{\sigma}_X^2} \sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y}) \\ &= \sum_{i=1}^{n} \underbrace{\frac{1}{n} \left(1 + \frac{(x - \bar{x})(x_i - \bar{x})}{\sigma_X^2} \right)}_{w(x_i, x)} y_i \\ &= \underbrace{\mathbf{b}(x) \left(\mathbf{X}' \mathbf{X} \right)^{-1} \mathbf{X}'}_{\mathbf{w}(x)} \cdot \mathbf{y} \end{aligned}$$

  kde $\mathbf{b}(x) = (1, x)$ je *vektorová bázová funkcia*, ktorá bola samozrejme použitá aj na zostavenie regresnej matice, čiže pre jej i-ty riadok platí vzťah $\{\mathbf{X}\}_{i\cdot} = \mathbf{b}(x_i)$.

- Ak z vektorovej váhovej funkcie po riadkoch (pre každé pozorovanie) poskladáme *váhovú maticu* $\mathbf{W}$, čiže $\{\mathbf{W}\}_{i\cdot} = \mathbf{w}(x_i)$ – v kontexte lineárnej regresie sme ju nazývali "hat matrix" – potom môžme odhadnuté (alebo vyhladené) hodnoty odozvy v bodoch pozorovaní zapísať vzťahom

$$\hat{\mathbf{y}} = \mathbf{W}\mathbf{y}$$

  a pomocou nej vyjadriť *efektívne stupne voľnosti*

$$df = \operatorname{tr}(\mathbf{W}) = \sum_{i=1}^{n} \{\mathbf{W}\}_{ii}$$

  alebo výpočtovo rýchly odhad testovacej chyby metódou krížovej validácie s jednoprvko-

vými validačnými sadami (LOOCV)

$$MSE = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{y_i - \hat{f}(x_i)}{1 - \{\mathbf{W}\}_{ii}}\right)^2.$$

5.4 Aditívny model

- Videli sme, že lineárne vyhladzovacie modely zovšeobecňujú jednoduchú lineárnu regresiu.
- Rozšírenie o ďalšie prediktory je možné, ale vo všeobecnosti (najmä pri neparametrických metódach) naráža na známy problém nedostatku dát so zväčšujúcim sa počtom premenných (curse of dimensionality).
- Kompromisným riešením je skombinovať aditivitu lineárnych regresných modelov s prispôsobivosťou lineárnych vyhladzovacích modelov v rámci *triedy aditívnych modelov*. Najjednoduchší p-rozmerný má tvar

  $$Y = \beta_0 + \sum_{j=1}^{p} f_j(X_j) + \varepsilon$$

  a umožňuje vybudovať celkový model z príspevkov jednotlivých prediktorov osobitne pomocou nelineárnych funkcií f_j, ktoré dostali názov parciálna odozva/reakcia (*partial response*).
- Nejednoznačnosť viacerých konštantných členov v jednom vzťahu rieši konvencia $E[f_j(X_j)] = 0$ pre všetky j.
- Ak sú všetky funkcie lineárne v parametroch, potom sa dá aj celkový aditívny model odhadnúť pomocou jednoduchej MNŠ (pretože je tiež lineárny v parametroch). Napr. vtedy, ak by boli f_j zvolené z triedy globálnych polynómov alebo regresných splajnov.
- Vo všeobecnosti sa však obyčajná MNŠ nedá použiť.
- Aditívny model sa odhaduje pomocou jednoduchej iteračnej metódy s názvom *backfitting*. Ňou sa v každom kroku cyklu pomocou vyhladzovacej metódy ($\mathcal{S}_k$) aktualizuje funkcia (f_k) vždy iba pre jeden prediktor (X_k, ostatné sú fixované), pričom odozvou sú tzv. *parciálne rezíduá* ($y_i - \sum_{j\neq k} f_j(x_{ij})$, $\forall i$).

Algoritmus (backfitting)

$\hat{\beta}_0 \leftarrow \frac{1}{n}\sum_{i=1}^{n} y_i$
$\hat{f}_j \leftarrow 0$ pre $j = 1, \ldots, p$
`while` $(\exists j : |\hat{f}_j - g_j| > \delta)$ `{`
  `for` $(k = 1, \ldots, p)$ `{`
    $y_i^{(k)} = y_i - \sum_{j\neq k} \hat{f}_j(x_{ij})$ pre $i = 1, \ldots, n$

$g_k \leftarrow \mathcal{S}_k(y^{(k)} \sim x_{\cdot k})$
$g_k(x) \leftarrow g_k(x) - \frac{1}{n}\sum_{i=1}^{n} g_k(x_{ik})$
$\hat{f}_k \leftarrow g_k$
}
}
return $(\hat{\beta}_0, \hat{f}_1, \ldots, \hat{f}_p)$

- Pre veľkú triedu lineárnych vyhladzovacích modelov $\mathcal{S}_k$ je backfitting totožný s Gauss-Seidelovým algoritmom riešenia špeciálnej lineárnej sústavy rovníc.
- Aditívny model má niekoľko výhod:
  - umožňuje automatický odhad nelineárnej funkcie pre každý prediktor zvlášť, bez potreby manuálneho skúšania množstva transformácií (t.j. bázových funkcií) individuálne na každý prediktor,
  - potenciálne presnejšie predpovede oproti lineárnemu modelu (ak je skutočný vzťah nelineárny),
  - vďaka aditivite možnosť skúmať efekt každého prediktoru zvlášť (vrátane vyjadrenia stupňov voľnosti),
- Aditivita môže byť aj nevýhodou, avšak podobne ako pri lineárnom modeli, aj tu je možné interakčný člen pridať ako nový prediktor $X_j X_k$, prípadne sa dá do rovnice zaradiť interakčná funkcia $f_{jk}(X_j, X_k)$ ako dvojrozmerný lineárny vyhladzovací model (napr. thin-plate spline, tensor spline, dvojrozmerné jadrové vyhladzovanie a pod.).
- Aditívny model predstavuje užitočný kompromis medzi lineárnym modelom a plne neparametrickými modelmi (boosting, random forest a pod.).

Príklad (mzda)

- Opäť modelujme výšku mzdy, no tentokrát nielen pomocou veku zamestnanca, ale aj pomocou roku záznamu a stupňa dosiahnutého vzdelania (ordinálna kvalitatívna premenná s úrovňami od "< HS grad" až po "Advanced Degree").
- Jednou možnosťou je odhadnúť aditívny model ako lineárny s použitím bázových funkcií regresných splajnov.

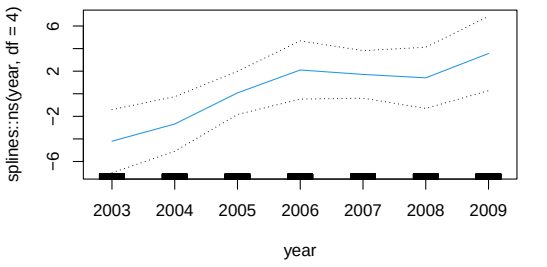
```
fit1 <- lm(wage ~ splines::ns(year, df = 4) + splines::ns(age, df = 5) + education,
           data = ISLR2::Wage)
fit1 |> coef()
```

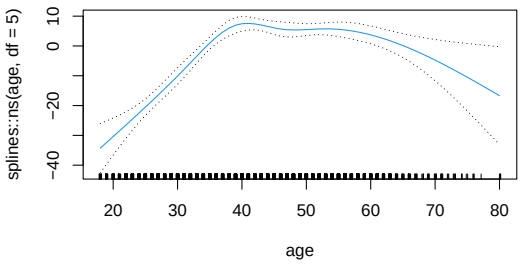
```
                (Intercept)  splines::ns(year, df = 4)1
                  46.949491                    8.624692
splines::ns(year, df = 4)2  splines::ns(year, df = 4)3
                   3.762266                    8.126549
```

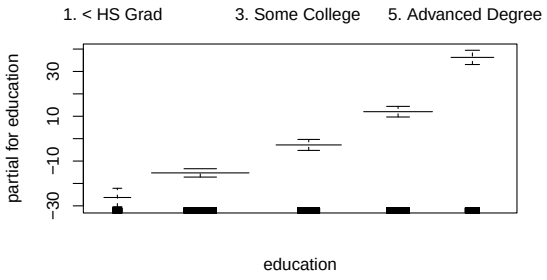
```
splines::ns(year, df = 4)4   splines::ns(age, df = 5)1
                  6.806473                  45.170123
 splines::ns(age, df = 5)2   splines::ns(age, df = 5)3
                 38.449704                  34.239237
 splines::ns(age, df = 5)4   splines::ns(age, df = 5)5
                 48.677634                   6.557265
      education2. HS Grad    education3. Some College
                 10.983419                  23.472889
 education4. College Grad education5. Advanced Degree
                 38.313667                  62.553971
```

- Metóda zobrazenia *plot.Gam* zohľadňuje hierarchiu parametrov a zobrazí parciálnu odozvu na jednotlivé premenné.

```
gam::plot.Gam(fit1, se = TRUE, col = 4)
```







- Podľa grafu by parciálna reakcia na prediktor *year* mohla byť lineárna funkcia. Overíme to testom.

```
lm(wage ~ year + splines::ns(age, 5) + education, data = ISLR2::Wage) |>
  anova(fit1)
```

```
Analysis of Variance Table

Model 1: wage ~ year + splines::ns(age, 5) + education
Model 2: wage ~ splines::ns(year, df = 4) + splines::ns(age, df = 5) +
    education
  Res.Df     RSS Df Sum of Sq      F Pr(>F)
1   2989 3694885
2   2986 3691919  3    2966.4 0.7997 0.4939
```

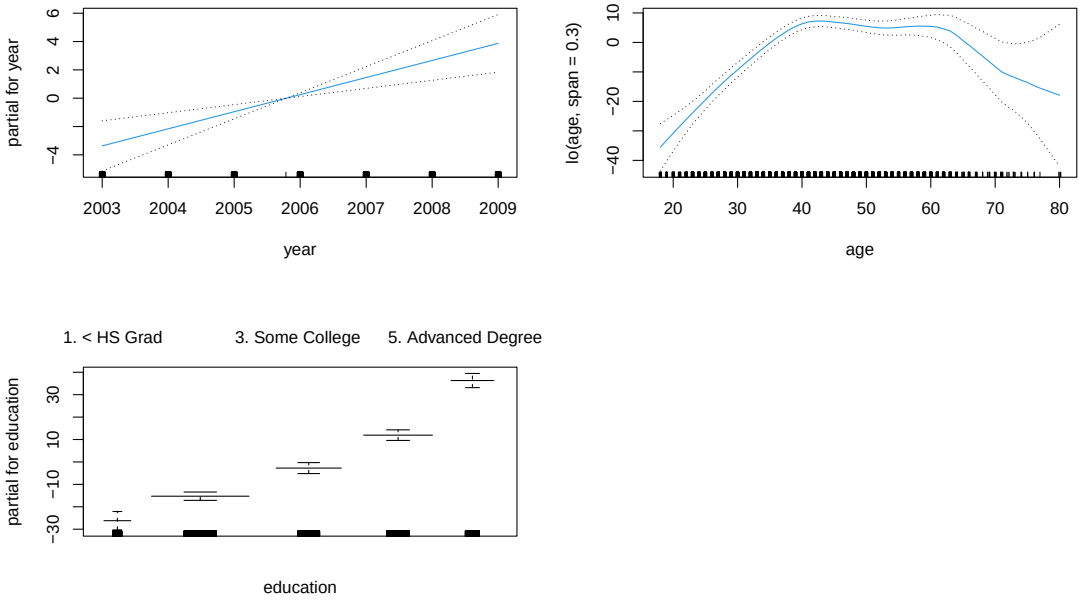
- Okrem regresných splajnov je možné zapojiť aj vyhladzovacie splajny či lokálnu regresiu. To umožňuje až implementácia zovšeobecneného aditívneho modelu (GAM), funkcia *gam::gam*. Metóda *summary.Gam* testuje lineárnu (parametrickú) a neparametrickú zložku zvlášť, takže sa dá jednoducho posúdiť, či sú stupne voľnosti zvolenej vyhladzovacej metódy adekvátne (metóda *anova.Gam* zobrazí iba výsledok testu významnosti neparametrickej zložky).

```
library(gam)  # necessary to load due to inaccountable behaviour of "gam::s" in gam() formu
gam::gam(wage ~ s(year, df = 4) + s(age, df = 5) + education,
                 data = ISLR2::Wage) |>
  anova()   # summary()
```

```
Anova for Nonparametric Effects
                Npar Df Npar F  Pr(F)
(Intercept)
s(year, df = 4)       3  1.086 0.3537
s(age, df = 5)        4 32.380 <2e-16 ***
education
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

- V poslednom GAM modeli je už nastavená lineárna parciálna odozva na prediktor *year* a pre ilustráciu použitá lokálna regresia na vystihnutie efektu *age*. Špecifikáciou vyhladzovacieho parametra `span` (podiel okolia x_0 na celkovom počte pozorovaní) sa interne použije funkcia *stats::loess*.

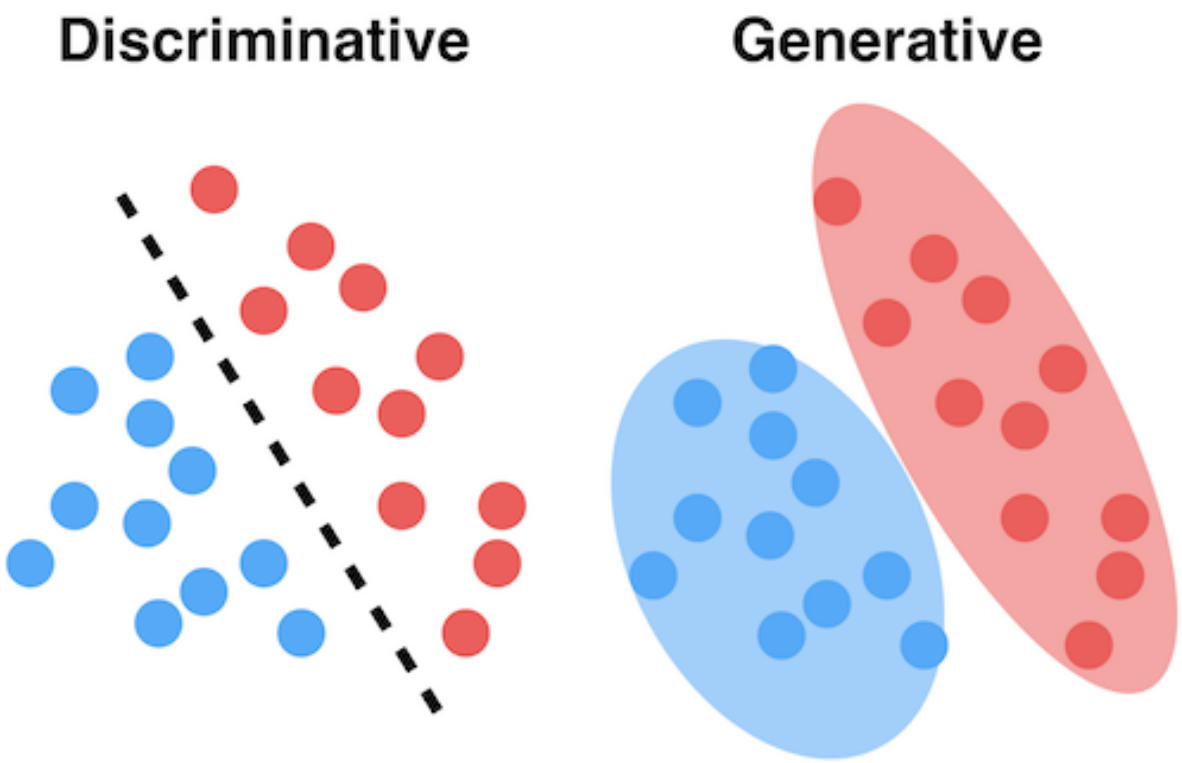
```
gam::gam(wage ~ year + lo(age, span = 0.3) + education,
                 data = ISLR2::Wage) |>
  plot.Gam(se = TRUE, col = 4)
```


partial for year
−4
0
2
4
6
2003
2004
2005
2006
2007
2008
2009
year
lo(age, span = 0.3)
−40
−20
0
10
20
30
40
50
60
70
80
age
1. < HS Grad
3. Some College
5. Advanced Degree
partial for education
−30
−10
10
30
education

6 Diskrimininatívne metódy klasifikácie

6.1 Úvod

- Predpovedanie kvalitatívnej (kategoriálnej) náhodnej premennej sa nazýva klasifikácia.
- Použitie bežnej (napr. lineárnej) regresie na modelovanie kvalitatívnej odozvy má príliš veľa nevýhod, najmä:
  - konverzia viachodnotovej kvalit. na kvantitatívnu *vnúti poradie* jej hodnotám,
  - ordinálnej premennej sa konverziou *vnúti vzdialenosť* medzi hodnotami,
  - binárna premenná sa síce dá zmysluplne reprezentovať hodnotami $\{0, 1\}$, čo sú extrémy *pravdepodobnosti*, no lineárnou regresiou by vznikli aj predpovede *menšie ako 0 a väčšie ako 1*.
- V tomto kurze preberieme metódy klasifikácie založené na modelovaní rozdelenia pravdepodobnosti. Ostatné metódy sú skôr náplňou voliteľného predmetu *Hĺbková analýza údajov a strojové učenie* v 2.ročníku I-MPM.
- Pravdepodobnostné klasifikátory sa delia na dve skupiny modelov:
  - diskriminatívne – pomocou podmienenej pravdepodobnosti (napr. logistická regresia, k-najbližších susedov)
  - generatívne – pomocou združeného rozdelenia a bayesovej vety (napr. lineárna/kvadratická diskriminačná analýza, naivný bayesovský klasifikátor)



6.2 Logistická regresia

6.2.1 Modelovanie pravdepodobnosti

- Logistickou regresiou modelujeme pravdepodobnosť, že binárna odozva Y (nech $\mathcal{H}(Y) = \{c_1, c_2\}$) nadobudne jednu z dvoch hodnôt, teda napríklad $Pr[Y = c_1|X]$.
- Konkrétne, ak Y je typ prevodovky, hodnota c_1 predstavuje triedu manuálnych prevodoviek a X napr. hmotnosť vozidla.
- Štandardne potom predpoveď $p(X) > 0.5$ znamená zatriedenie do skupiny c_1.
- Prakticky sa hodnoty Y prekódujú na 0 a 1 (tak ako pri indikačných premenných), takže modelujeme pravdepodobnosť $p(X) \equiv Pr(Y = 1|X)$.
- Modelovanie pomocou lineárnej regresnej funkcie

  $$p(X) = b_1 + b_2 X$$

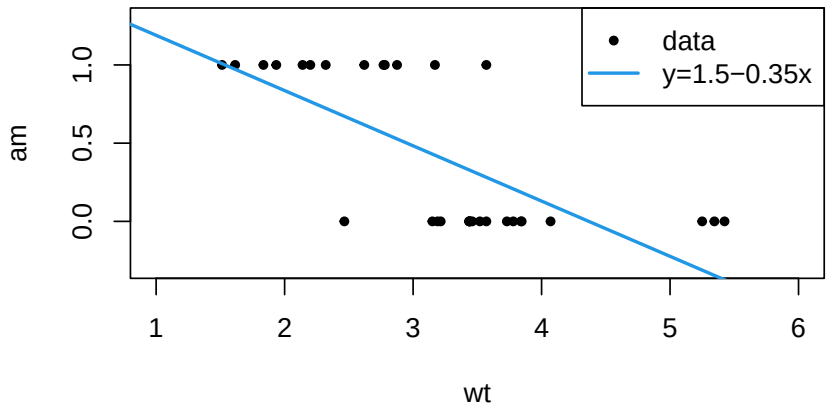
  je možné, ale vedie k neinterpretovateľným hodnotám.

Príklad (autá)

- Modelujme vzťah medzi typom prevodovky (odozva) a hmotnosťou vozidla (pred-

iktor) na základe súboru pozorovaní *mtcars* pomocou lineárnej regresie.

```
plot(am ~ wt, mtcars, pch=20, xlim = c(1,6), ylim = c(-0.3,1.3))
lm(am ~ wt, mtcars) |> abline(reg = _, col = 4, lwd = 2)
legend("topright",
       legend = c("data", "y=1.5-0.35x"),
       pch=c(20,NA), lty=c(0,1), col=c(1,4), lwd = c(NA,2)
  )
```



6.2.2 Transformácia

- Riešením je transformovať priamku (lineárnu funkciu) tak, aby sa zmestila do [0,1]:
  1. Aplikovaním exponenciálnej funkcie získame dolné ohraničenie.
  2. Hyperbolickou funkciou $f(x) = \frac{x}{x+1}$ na intervale $(0, \infty)$ dostaneme žiadané ohraničenie (0,1).
- Pravdepodobnosť je tak vyjadrená tzv. logistickou funkciou

$$p(X) = \frac{e^{b_1+b_2X}}{1+e^{b_1+b_2X}}$$

- Platí $p\left(-\frac{b_1}{b_2}\right) = 0.5$ (prah klasifikácie) a s rastúcim $|b_2|$ je prechod cez prah strmší. Konkrétne, po zderivovaní $p(X)$ a úprave dostaneme

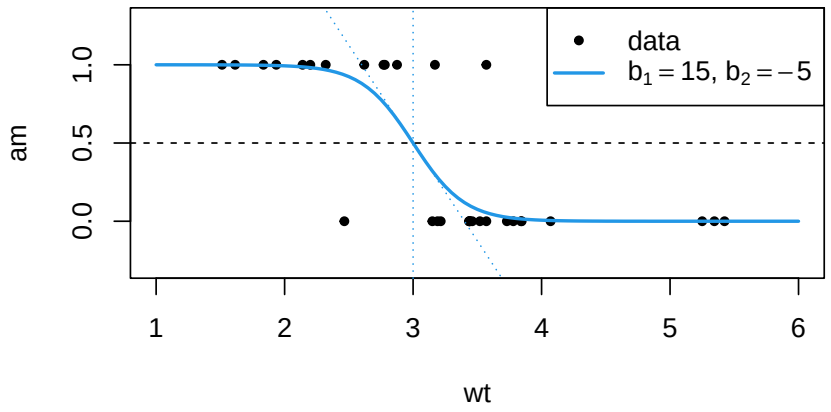
$$p'(X) = p(X)\frac{b_2}{1+e^{b_1+b_2X}}$$

takže v bode polovičnej pravdepodobnosti bude smernica dotyčnice $p'\left(-\frac{b_1}{b_2}\right) = \frac{b_2}{4}$.

Príklad (autá)

- Teraz rovnaký vzťah medzi prevodovkou a hmotnosťou vyjadrime pomocou logistickej funkcie so *zvolenými* hodnotami parametrov.

```
plot(am ~ wt, mtcars, pch=20, xlim = c(1,6), ylim = c(-0.3,1.3))
plot(function(x) exp(15-5*x)/(1+exp(15-5*x)), from = 1, to = 6, add=T, col = 4, lwd=2)
abline(h = 0.5, lty="dashed")
abline(a = 15/4+1/2, b=-5/4, lty="dotted", col = 4)
abline(v = 15/5, lty="dotted", col = 4)
legend("topright",
       legend = c("data", expression(paste(b[1]==15,", ", b[2]==-5))),
       pch=c(20,NA), lty=c(0,1), col=c(1,4), lwd = c(NA,2)
  )
```



- Prah klasifikácie (rozhodovacia hranica) pre polovičnú pravdepodobnosť je bode $wt = -15/(-5) = 3$ a dotyčnica v tom bode má sklon $p'(3) = -5/4$.

6.2.3 Šanca

- Podelením pravdepodobnosti jej doplnkom dostaneme tzv. *šancu* (odds), čo je hodnota v intervale $[0, \infty)$. Je známa najmä zo stávkovania.

- Napr. pravdepodobnosť 0.2 zodpovedá šanci $\frac{0.2}{1-0.2} = 1/4$, čiže iba 1 auto s manuálnou prevodovkou ku 4 autám s automatikou.
- Logaritmus šance (log odds, **logit**) je lineárnou funkciou prediktoru X,

$$\frac{p(x)}{1-p(x)} = e^{b_1+b_2x}$$
$$\ln\left(\frac{p(x)}{1-p(x)}\right) = b_1 + b_2x$$

- To znamená, že
  - zvýšenie x o 1 zmení šancu e^{b_2}-násobne,
  - zmena v pravdepodobnosti závisí už aj od hodnoty x,
  - ak $b_2 > 0$, potom $p(x)$ je rastúca funkcia (a naopak).

6.2.4 Odhad parametrov

- Odhad parametrov b_1, b_2 by sa dal urobiť nelineárnou metódou najmenších štvorcov, ale lepšie štatistické vlastnosti má všeobecnejšia metóda – metóda maximálnej vierohodnosti.
- Vierohodnostná funkcia

$$\begin{aligned} L(b_1, b_2) &= \prod_{i=1}^{n} Pr(Y = y_i | X = x_i) \\ &= \prod_{i=1}^{n} p(x_i)^{y_i}(1-p(x_i))^{1-y_i} \\ &= \prod_{i:y_i=1} p(x_i) \prod_{i:y_i=0} (1-p(x_i)) \end{aligned}$$

  vychádza z hustoty Bernouliho rozdelenia (špeciálny prípad binomického, pre jediný pokus).
- Štandardne sa vierohodnostná funkcia zlogaritmuje, derivácie podľa parametrov položia nule a rieši sústava rovníc. Žiaľ, presné riešenie nedokážeme explicitne vyjadriť, preto sa v takých prípadoch hľadá približné riešenie pomocou Newtonovej metódy: začne štartovacími hodnotami parametrov a podľa veľkosti a smeru gradientu sa pohybuje po povrchu log-vierohodnostnej funkcie.
- V kontexte logistickej regresie je tento postup rovnaký ako iteračné použitie váženej metódy najmenších štvorcov (iteratively re-weighted least sqares):

  1. Definujme logit funkciu $g(p) \equiv \log \frac{p}{1-p}$, potom
     - model je v tvare $g(p) = b_1 + b_2x$,
     - pravdepodobnosť $p(x) = g^{-1}(b_1 + b_2x)$ a
     - rozdelenie odozvy $Y|X \sim Binom\Big(1, p(x)\Big)$ s varianciou $V(p) = p(x)(1-p(x))$.

2. Zdalo by sa, že môžeme priamo urobiť lineárnu regresiu $g(Y)$ podľa X, samozrejme váženú, pretože rozptyl je premenlivý. Problém je, že $g(0) = -\infty$ a $g(1) = \infty$. Obídeme to Taylorovým rozvojom

$$\begin{aligned} g(y) \approx z &= g(p) + (y-p)g'(p) \\ &= b_1 + b_2 x + (y-p)g'(p) \end{aligned}$$

kde Z sa stane našou pseudo-odozvou. Pritom, ak sú parametre správne, platí $E[Y|X=x] = p$ takže $(y-p)$ by mal byť šum s nulovou strednou hodnotou avšak s premenlivým rozptylom $var[Z|X=x] = var[(Y-p)g'(p)|X=x] = (g'(p))^2 V(p)$. Na odhad b_1, b_2 je teda nutné použiť váženú MNŠ.

3. Iteračný cyklus začína použitím štartovacích hodnôt b_1, b_2 na výpočet hodnôt pseudo-odozvy z_i a ich váh. Tie vzápätí použije na výpočet presnejších hodnôt b_1, b_2 atď.

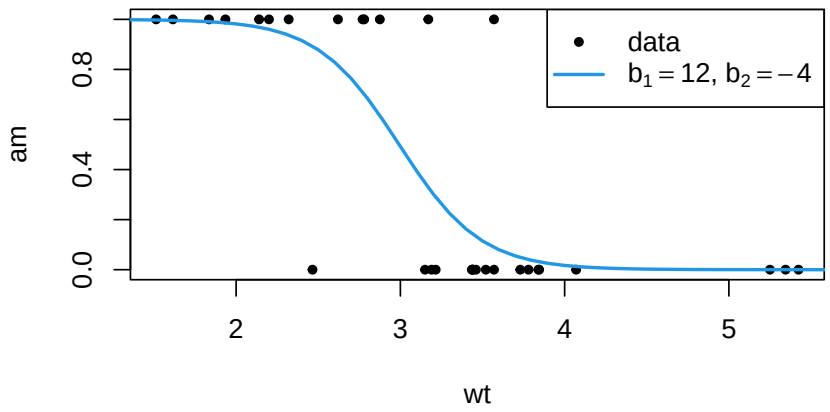
Príklad (autá)

- Rovnaký príklad s hmotnosťou *wt* a typom prevodovky *am*:

```
fit <- glm(am ~ wt, mtcars, family = "binomial")
fit
```

```
Call:  glm(formula = am ~ wt, family = "binomial", data = mtcars)

Coefficients:
(Intercept)           wt
     12.040       -4.024

Degrees of Freedom: 31 Total (i.e. Null);  30 Residual
Null Deviance:      43.23
Residual Deviance: 19.18    AIC: 23.18
```

```
xgrid <- seq(1,6,0.1)
plot(am ~ wt, mtcars, pch=20)
lines(xgrid,
      predict(fit, newdata = data.frame(wt = xgrid), type = "response"),
      col = 4, lwd = 2)
legend("topright",
       legend = c("data", expression(paste(b[1]==12,", ", b[2]==-4))),
       pch=c(20,NA), lty=c(0,1), col=c(1,4), lwd = c(NA,2)
  )
```



- Výstup funkcie *glm* obsahuje tzv. *null deviance* a *residual deviance*.
  - (Reziduálna) deviancia D je rozdiel medzi log-vierohodnostnými funkciami saturovaného modelu (dokonale priliehajúceho k dátam) a odhadnutého modelu. A keďže likelihood funkcia saturovaného modelu sa rovná jednej, jej logaritmus je nulový, potom residual deviance $D = -2 \log L(\hat{b}_1, \hat{b}_2)$ je vždy väčšia ako nula, nanajvýš rovná nule (ak je model dokonalý).
  - Naproti tomu null deviance je referenčnou hodnotou zodpovedajúcou základnému modelu (bez prediktorov) $D_0 = -2 \log L(\hat{b}_1 | b_2 = 0)$.
  - Násobok 2 zabezpečuje deviancii (asymptoticky) χ^2 rozdelenie pravdepodobnosti.
- Akaikeho inf. kritérium AIC je iba miera vychádzajúca z deviancie a penalizujúca zložitejšie modely, $AIC = D + 2k$, kde k je počet parametrov modelu, v poslednom príklade $k = 2$. Preferujeme model s čo najnižším AIC.

6.2.5 Multinomická logistická regresia

- Logistická regresia sa dá použiť aj na klasifikovanie odozvy do viac ako dvoch tried. Jedna z tried pritom môže byť zvolená ako *referenčná*.
- Ak za referenčnú zvolíme poslednú zo všetkých K tried, potom pre $k = 1, \ldots, K-1$ platí

$$Pr[Y = k | X = x] = \frac{e^{b_{k1} + b_{k2} x}}{1 + \sum_{l=1}^{K-1} e^{b_{l1} + b_{l2} x}},$$

$$Pr[Y = K | X = x] = \frac{1}{1 + \sum_{l=1}^{K-1} e^{b_{l1} + b_{l2} x}},$$

čiže

$$\log\left(\frac{Pr[Y=k|X=x]}{Pr[Y=K|X=x]}\right) = b_{k1} + b_{k2}x,$$

log-šanca medzi ktorýmkoľvek párom tried je lineárnou funkciou prediktoru X. Interpretácia je viazaná na voľbu referenčnej triedy.

- Alternatívna formulácia, tzv. *softmax*, nepoužíva referenčnú triedu:

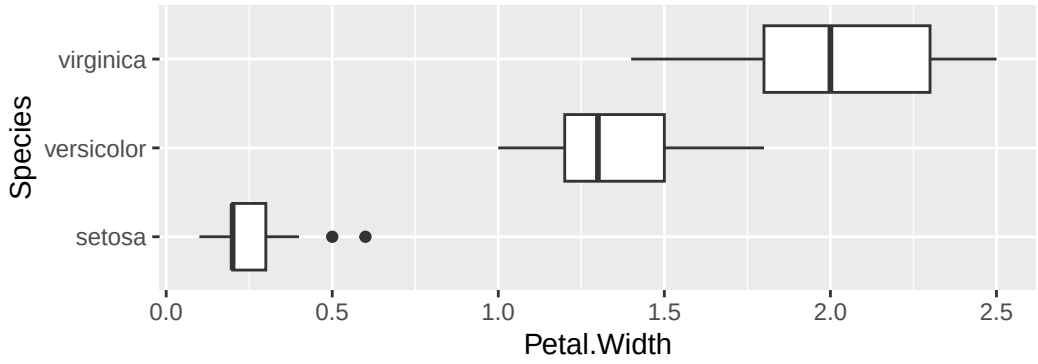
$$Pr[Y=k|X=x] = \frac{e^{b_{k1}+b_{k2}x}}{\sum_{l=1}^{K} e^{b_{l1}+b_{l2}x}} \qquad \text{pre} \quad k \in \{1,\dots,K\},$$

$$\log\left(\frac{Pr[Y=k_1|X=x]}{Pr[Y=k_2|X=x]}\right) = (b_{k_1 1} - b_{k_1 1}) + (b_{k_1 2} - b_{k_2 2})x \qquad \text{pre} \quad k_1, k_2 \in \{1,\dots,K\}.$$

Príklad (kosatce)

- Na základe datasetu *iris* klasifikujme druh (*Species*) kosatca podľa šírky korunného lupienku kvetu (*Petal.Width*).

```
# odhad
fit <- nnet::multinom(Species ~ Petal.Width, data = iris, trace = FALSE)

# rozdelenie údajov
ggplot(iris) + aes(x = Petal.Width, y = Species) +
  geom_boxplot(varwidth = T)
```

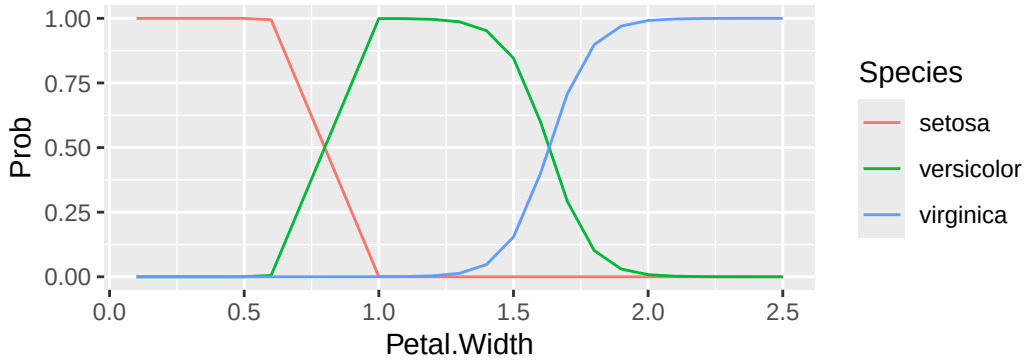



```
# súhrn modelu
fit
```

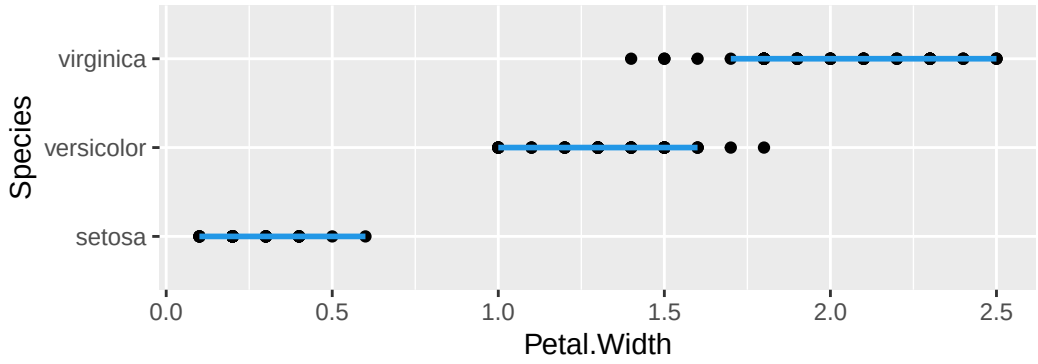
```
Call:
nnet::multinom(formula = Species ~ Petal.Width, data = iris,
    trace = FALSE)
```

```
Coefficients:
           (Intercept) Petal.Width
versicolor   -24.08868    31.44849
virginica    -45.20657    44.39249

Residual Deviance: 33.44131
AIC: 41.44131
```

```
# zobrazenie pravdepodobností
predict(fit, type = "probs") |>
  as.data.frame() |>
  transform(Petal.Width = iris$Petal.Width) |>
  tidyr::pivot_longer(cols = -Petal.Width, names_to = "Species", values_to = "Prob") |>
  ggplot() + aes(x = Petal.Width, y = Prob, color = Species, group = Species) +
  geom_line()
```




```
# klasifikácia
iris |>
  transform(fit = predict(fit, type = "class")) |>
  ggplot() + aes(x = Petal.Width) +
  geom_point(aes(y = Species)) +
  geom_line(aes(y = fit), color = 4, lwd = 1)
```



6.2.6 Ordinálna logistická regresia

- Iným názvom aj *ordered logit* alebo *proportional odds logistic regression* je ďalším rozšírením logistickej regresie (klasifikácie na tri a viac tried) a využíva *zoradenosť* úrovní kvalitatívnej odozvy.
- Oproti multinomickej tak ordinálna LR modeluje šancu, že odozva nadobudne *nanajvýš* určitú hodnotu oproti tomu, že nadobudne *vyššie* hodnoty,

$$\log\left(\frac{Pr[Y \leq k|X=x]}{Pr[Y > k|X=x]}\right) = b_{k1} - b_2 x, \qquad \text{pre} \quad k = 1, \ldots, K-1$$

- Každý z $K-1$ log-podielov má v lineárnom modeli svoj absolútny člen b_{k1}, avšak parameter pri X si zdieľajú (na rozdiel od MLR). Vďaka tomu je model pomerne jednoduchý (v komplikovanejších prípadoch je však možné použiť zovšeobecnený OLR model).
- Parametrizácia so znamienkom mínus má význam pre zachovanie štandardnej interpretácie parametra stojaceho pri prediktore.

Príklad (autá)

- Modelujeme rozdelenie ordinálnej odozvy *cyl* pomocou prediktoru *disp*.

```
fit <- mtcars |>
  dplyr::mutate(cyl = ordered(cyl, c("4","6","8"))) |>
  MASS::polr(cyl ~ disp, data=_)
```

```
Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

```
fit
```

```
Call:
```

```
MASS::polr(formula = cyl ~ disp, data = dplyr::mutate(mtcars,
    cyl = ordered(cyl, c("4", "6", "8"))))

Coefficients:
     disp
0.4443344

Intercepts:
     4|6      6|8
 64.9643 118.0268

Residual Deviance: 3.924663
AIC: 9.924663
Warning: did not converge as iteration limit reached
```

- Kladná hodnota parametra značí, že *cyl* je rastúcou funkciou prediktoru *disp*. Konkrétne, s nárastom zdvihového objemu o 1 jednotku (inch^3) sa šanca nižších tried voči vyšším zmení ($e^{-0.444} = 0.6$)-násobne, čiže klesne.
- Skonštruujme predpoveď počtu valcov pre motor so zdvihovým objemom 150 inch^3:

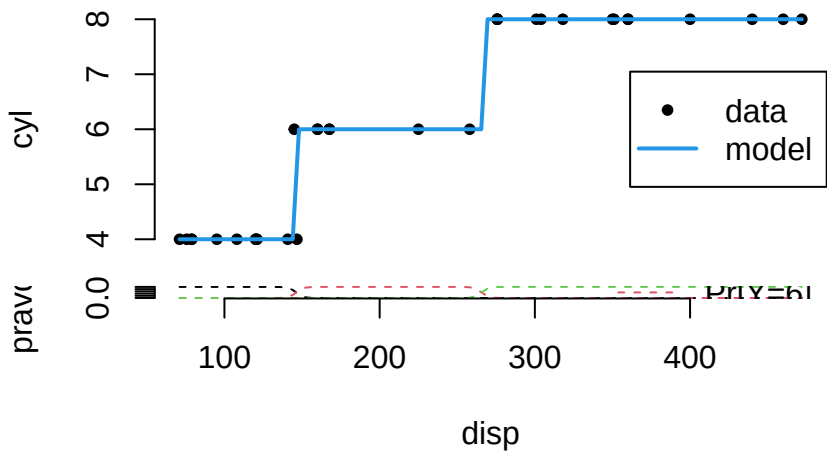
$$logit(Pr[cyl \leq 4|disp = 150]) = 65.0 - 0.444 \cdot 150 = -1.6$$
$$Pr[cyl \leq 4|disp = 150] = \frac{e^{-1.6}}{1 + e^{-1.6}} = 0.168$$
$$Pr[cyl \leq 6|disp = 150] = \frac{e^{118.0 - 0.444 \cdot 150}}{1 + e^{118.0 - 0.444 \cdot 150}} \approx 1$$

  potom pravdepodobnosti jednotlivých tried

  $Pr[cyl = 4|disp = 150] = Pr[cyl \leq 4|disp = 150] = 0.168$
  $Pr[cyl = 6|disp = 150] = Pr[cyl \leq 6|disp = 150] - Pr[cyl \leq 4|disp = 150] = 0.832$
  $Pr[cyl = 8|disp = 150] = 1 - Pr[cyl \leq 6|disp = 150] \approx 0$

  takže $E[cyl|disp = 150] = 6$ valcov.

```
xnew <- seq(min(mtcars$disp), max(mtcars$disp), length.out = 100)
layout(rbind(1,2), heights = c(5,3))
old <- par(oma=c(3, 3, 3, 3), mar=c(1, 4, 0, 0))

plot(cyl ~ disp, mtcars, axes=F, pch=20)
lines(x = xnew,
      y = as.numeric(as.character(predict(fit, newdata = list(disp = xnew)))),
      col = 4, lwd=2)
legend("right", c("data","model"), lty=c(NA,1), lwd=c(NA,2), col=c(1,4), pch = c(20,NA))
axis(2); par(mar=c(4, 4, 0, 0))
matplot(xnew, predict(fit, newdata = list(disp = xnew), type = "p"),
        xlab = "disp", ylab = "pravdep.", type= "l", lty = 2, axes = F)
legend("right", c("Pr[Y=4]","Pr[Y=6]","Pr[Y=8]"), lty=2, col=1:3, bty="n")
axis(1); axis(2)
```




```
layout(rbind(1))
par(old)
```

6.3 k-najbližších susedov

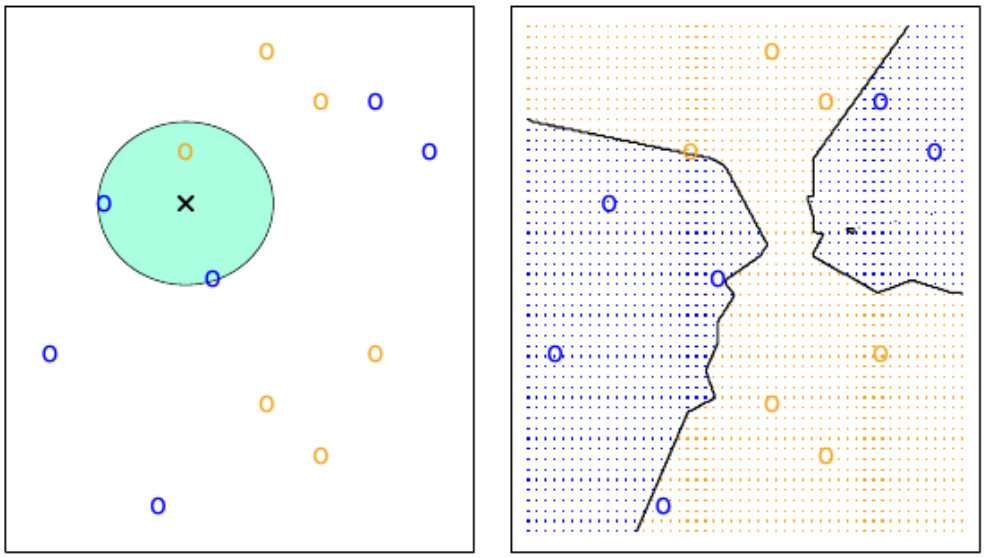
- Neparametrickou alternatívou ku logistickej regresii v diskriminatívnom prístupe ku klasifikačnej úlohe je metóda κ-najbližších susedov (kNN), ktorú poznáme už z regresnej úlohy.
- Nech $\mathcal{N}_\kappa$ je množina indexov κ pozorovaní (trénovacej vzorky), ktoré sa nachádzajú v najbližšom okolí bodu x v zmysle euklidovskej vzdialenosti.
- kNN klasifikátor odhaduje podmienenú pravdepodobnosť triedy j ako podiel počtu bodov v okolí x, ktorých hodnota odozvy je rovná j, čiže

$$Pr[Y = j|X = x] = \frac{1}{\kappa} \sum_{i \in \mathcal{N}_\kappa} I(y_i = j)$$

- Zatriedenie do skupiny sa potom vykoná klasicky, podľa najvyššej pravdepodobnosti.

Ilustrácia (κ-najbližších susedov)

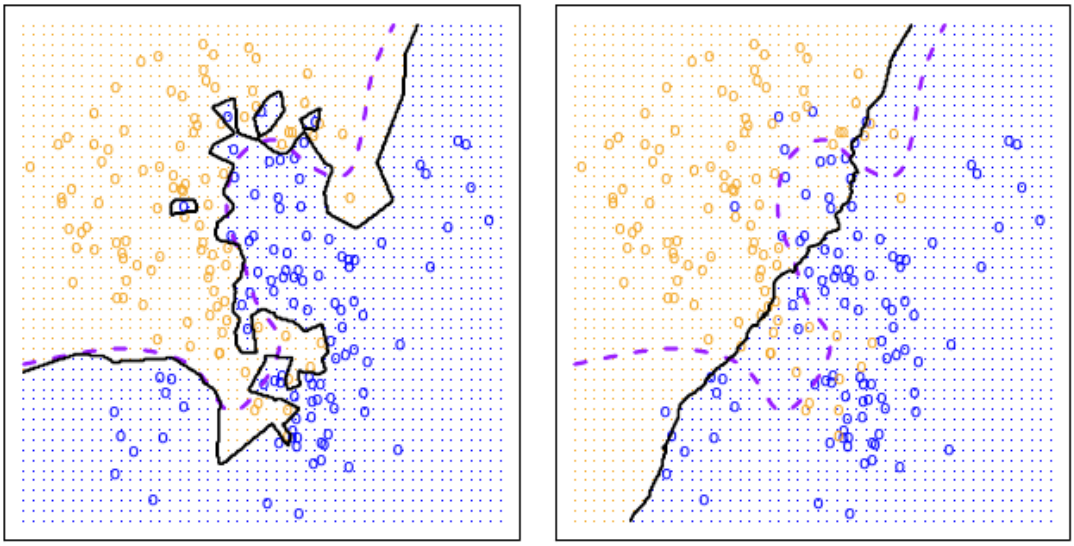
- Príklad s dvomi prediktormi, dvomi triedami a tromi susedmi: zeleným kruhom je znázornené okolie bodu x, čiernou krivkou rozhodovacia hranica medzi modrou a žltou triedou.



- Pretože sa príslušnosť do množiny susedov $\mathcal{N}_\kappa$ meria euklidovskou vzdialenosťou, musia byť pre $p \geq 2$ hodnoty prediktorov *normované* (štandardizované, t.j. centrované s jednotkovým rozptylom).
- Flexibilita modelu klesá s rastom počtu susedov.

Ilustrácia (počet susedov)

- Na nasledujúcom ilustračnom obrázku vľavo je $\kappa = 1$, v pravo $\kappa = 100$, fialovou prerušovanou čiarou je znázornená najlepšia (tzv. Bayesova) rozhodovacia hranica.

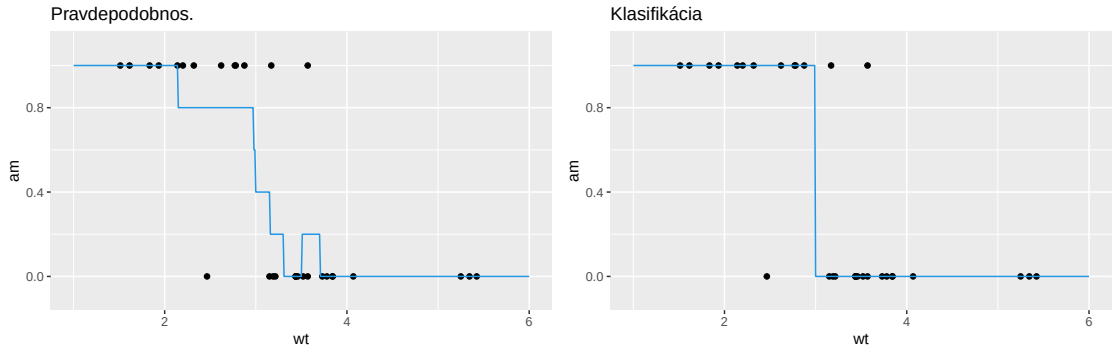


- Často používaným pravidlom pre ad-hoc voľbu počtu susedov je $\kappa \approx \sqrt{n}$. K ešte menšej chybe vedú rôzne metódy resamplingu ako napr. dobre známa krížová validácia.

Príklad (autá)

Jeden prediktor, dve triedy, dataset *mtcars*.

```r
# Funkcia na výpočet podmienených pravdepodobností:
knn.prob <- Vectorize(
  FUN = function(x, vx, vy, k) {
    x |> (`-`)(vx) |> abs() |> sort(index.return=T) |> getElement("ix") |> head(k) |> (`[`)
    },
  vectorize.args = "x"
)
# použitie: knn.prob(6, mtcars$wt, mtcars$am, k=5)

# klasifikácia a výpočet pravdepodobností:
pred <- data.frame(wt = seq(1,6,0.01)) |>
  transform(
    clas = class::knn(train = cbind(mtcars$wt),  # prediktory (ako matica plánu)
                      test = cbind(wt),  # nové hodnoty prediktorov
                      cl = as.factor(mtcars$am),  # odozva (triedy)
                      k = 5   # počet susedov, floor(sqrt(nrow(mtcars)))
                      ) |> as.character() |> as.numeric(),  # factor to number
    prob = knn.prob(wt, vx = mtcars$wt, vy = mtcars$am, k = 5)
  )

# zobrazenie:
ggplot(mtcars) + aes(x = wt) + geom_point(aes(y = am)) +
  geom_line(aes(y = prob), data = pred, color = 4) +
  ggtitle("Pravdepodobnosť") +
  ylim(-0.1, 1.1)
ggplot(mtcars) + aes(x = wt) + geom_point(aes(y = am)) +
  geom_line(aes(y = clas), data = pred, color = 4) +
  ggtitle("Klasifikácia") +
  ylim(-0.1, 1.1)
```



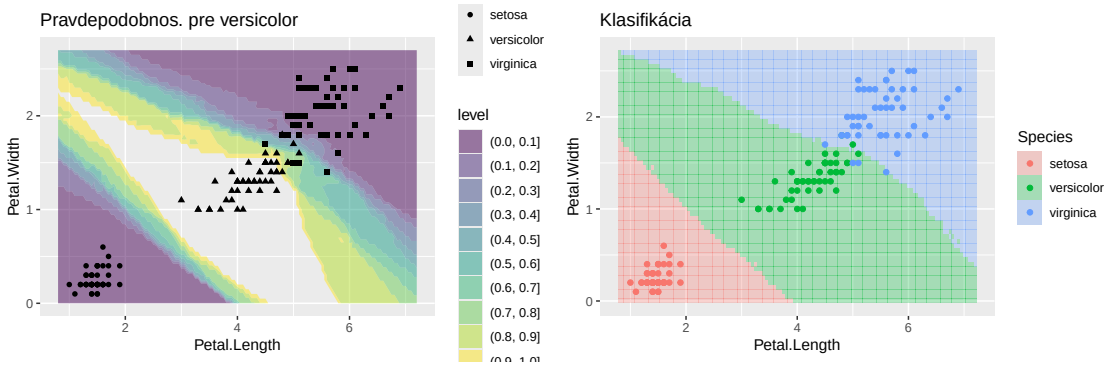
Príklad (kosatce)

- Dva prediktory a tri triedy na datasete *iris*.
- Tentokrát použijeme sofistikovanejšie nástroje – z balíka *caret*.

```
# trénovanie (k=9 zvolené pomocou resamplingu)
fit <- caret::train(Species ~ Petal.Length + Petal.Width, data = iris,
             method = "knn",
             preProcess = c("center", "scale")
             )
# predpovede a zobrazenie
xgrid <- expand.grid(Petal.Length = seq(0.8,7.2,0.05),
                     Petal.Width = seq(0,2.7,0.05))
fit |>
  predict(newdata = xgrid, "prob") |>
  cbind(xgrid) |>
  dplyr::select(-setosa, -virginica) |>
  dplyr::rename(prob_versicolor = versicolor) |>
  ggplot() + aes(x = Petal.Length, y = Petal.Width) +
  geom_contour_filled(aes(z = prob_versicolor, colour = prob_versicolor), alpha = 0.5) +
  geom_point(aes(shape = Species), data = iris) + ggtitle("Pravdepodobnosť pre versicolor")
```

```
Warning: The following aesthetics were dropped during statistical transformation:
colour.
i This can happen when ggplot fails to infer the correct grouping structure in
  the data.
i Did you forget to specify a `group` aesthetic or to convert a numerical
  variable into a factor?
```

```
fit |>
  predict(newdata = xgrid, "raw") |>
  cbind(Species = _, xgrid) |>
  ggplot() + aes(x = Petal.Length, y = Petal.Width) +
  geom_tile(aes(fill = Species), alpha = 0.3) +
  geom_point(aes(color = Species), data = iris) +
  ggtitle("Klasifikácia")
```


Pravdepodobnos. pre versicolor
Petal.Width
Petal.Length
setosa
versicolor
virginica
level
(0.0, 0.1]
(0.1, 0.2]
(0.2, 0.3]
(0.3, 0.4]
(0.4, 0.5]
(0.5, 0.6]
(0.6, 0.7]
(0.7, 0.8]
(0.8, 0.9]
Klasifikácia
Petal.Width
Petal.Length
Species
setosa
versicolor
virginica

7 Zovšeobecnený lineárny model

7.1 Úvod

- Lineárna i logistická regresia sú špeciálnym prípadom zovšeobecneného lineárneho modelu (angl. *generalized linear model*, GLM), ktorý sa skladá z troch častí:
  1. rozdelenie pravdepodobnosti Y,
  2. *lineárny prediktor*
  $$\eta(x) \equiv b_1 + b_2 x,$$
  3. *link* funkcia g, pre ktorú platí
  $$\mu = g^{-1}(\eta) \qquad \text{resp.} \qquad g(\mu) = \eta$$
  kde μ tradične predstavuje strednú hodnotu Y, teda $\mu(x) \equiv E[Y|X = x]$.
- Variancia odozvy je modelovaná dvojzložkovo, $var[Y|X] = \sigma^2 V(\mu)$, kde V je variančná funkcia (definovaná rozdelením) a σ^2 je základný rozptyl.

7.2 Rozdelenie pravdepodobnosti

- Premenná Y môže mať ktorékoľkvek z *exponenciálnej triedy rozdelení*, ktorá je daná hustotou
$$f_Y(y|\theta) = h(y) \exp\Big(\alpha(\theta) \cdot T(y) - A(\theta)\Big)$$
so známymi funkciami h, α, T, A.
- Do exponenciálnej triedy patria napríklad
  - z diskrétnych rozdelení: Bernouliho, binomické (s daným n), Poissonovo,
  - zo spojitých rozdelení: normálne, exponenciálne, gamma, χ^2, beta.
- Niektoré rozdelenia sú členom exponenciálnej triedy iba pre zafixované hodnoty niektorého zo svojich parametrov. Podmienkou je totiž nemenný nosič hustoty pravdepodobnosti.

- Pravdepodobnostná funkcia (prob. mass function) *binomického rozdelenia s fixným parametrom* n sa dá prepísať do všeobecného tvaru:

$$f(y) = \binom{n}{y} p^y (1-p)^{n-y} = \binom{n}{y} \exp\left(y \ln\left(\frac{p}{1-p}\right) + n \ln(1-p)\right)$$

  pre $y \in \{0, 1, 2, \dots, n\}$, kde je vidieť, že

  - $h(y) = \binom{n}{y}$,
  - $\alpha = logit$,
  - $T(y) = y$ a
  - $A(p) = -n \ln(1-p)$.

- Pre *normálne rozdelenie* $f(y; \mu, \sigma) = \frac{1}{2\pi\sigma^2} exp\left(-\frac{(y-\mu)^2}{2\sigma^2}\right)$ zas platí

  - $\alpha(\mu, \sigma) = \left(\frac{\mu}{\sigma^2}, -\frac{1}{2\sigma^2}\right)$,
  - $h(y) = \frac{1}{\sqrt{2\pi}}$,
  - $T(y) = (y, y^2)$,
  - $A(\mu, \sigma) = \frac{\mu^2}{2\sigma^2} + \ln \sigma$.

7.3 Odhad parametrov

Postup metódou iteračne vážených najmenších štvorcov je rovnaký ako pri logistickej regresii:

1. Máme dáta $(x_1, y_1), \dots, (x_n, y_n)$, zvolíme link funkciu g, variančnú funkciu V a štartovacie hodnoty parametrov b_1, b_2.
2. Pokým odhad parametrov neskonverguje, je treba
   1. vypočítať $\eta(x_i) = b_1 + b_2 x_i \quad \text{a} \quad \mu(x_i) = g^{-1}(\eta(x_i))$,
   2. získať zástupnú odozvu $z_i = \eta(x_i) + (y_i - \mu(x_i)) g'(\mu(x_i))$
   3. vypočítať váhy $w_i = \left[g'(\mu(x_i))^2 V(\mu(x_i))\right]^{-1}$,
   4. získať presnejšie parametre b_1, b_2 z regresie z_i podľa x_i s váhami w_i.

   Pre odhad parametrov nie je nutné poznať σ^2, pretože váhy majú byť iba *úmerné* prevrátenej hodnote variancie.

7.4 Príklady

7.4.1 Prehľad

odozva	rozdelenie	nosič	link $g(\mu)$	var.fu. $V(\mu)$
nastatie udalosti, áno/nie	Bernouliho	$\{0, 1\}$	$logit(\mu)$	$\mu(1-\mu)$
počet úspechov z n pokusov, pomer	binomické	$\{0, 1, \dots, n\}$	$logit(\mu)$	$\mu(n-\mu)/n$
počet	Poissonovo	$\mathbb{N}$	$\ln(\mu)$	μ
spojitá, lineárna	normálne	$\mathbb{R}$	μ (identity)	1
rozsah, miera	gamma	$(0, \infty)$	$\frac{1}{\mu}$ (inverse)	μ^2

7.4.2 Lineárna regresia

- S link funkciou $g(x) = x$ a variančnou funkciou $V(\mu) = 1$ je zjavné, že

$$E[Y|X = x] = b_1 + b_2 x \qquad \text{a} \qquad var[Y|X = x] = \sigma^2.$$

- Pretože $g' = 1$, všetky váhy $w_i = 1$, takže iteračná procedúra nie je potrebná.

7.4.3 Binomická regresia

- Odozva y_i je *počet v rozsahu* od 0 po n_i, napr.
  - počet pacientov s určitou diagnózou na nemocničnom oddelení,
  - počet žiakov v triede, ktorí prešli testom,
  - počet vadných výrobkov na výstupe z továrne.
- Logistická regresia je špeciálnym prípadom, keď $n_i = 1$, pre $\forall i$.

Príklad (menarche)

- Pomocou datasetu *MASS::menarche* modelujme vzťah veku a podielu pohlavne dospelých dievčat vo svojej (vekovej) skupine.
- Ukážka datasetu:

```
# data
dat <- MASS::menarche
head(dat, 6)
```

```
    Age Total Menarche
1  9.21   376        0
2 10.21   200        0
3 10.58    93        0
4 10.83   120        2
```

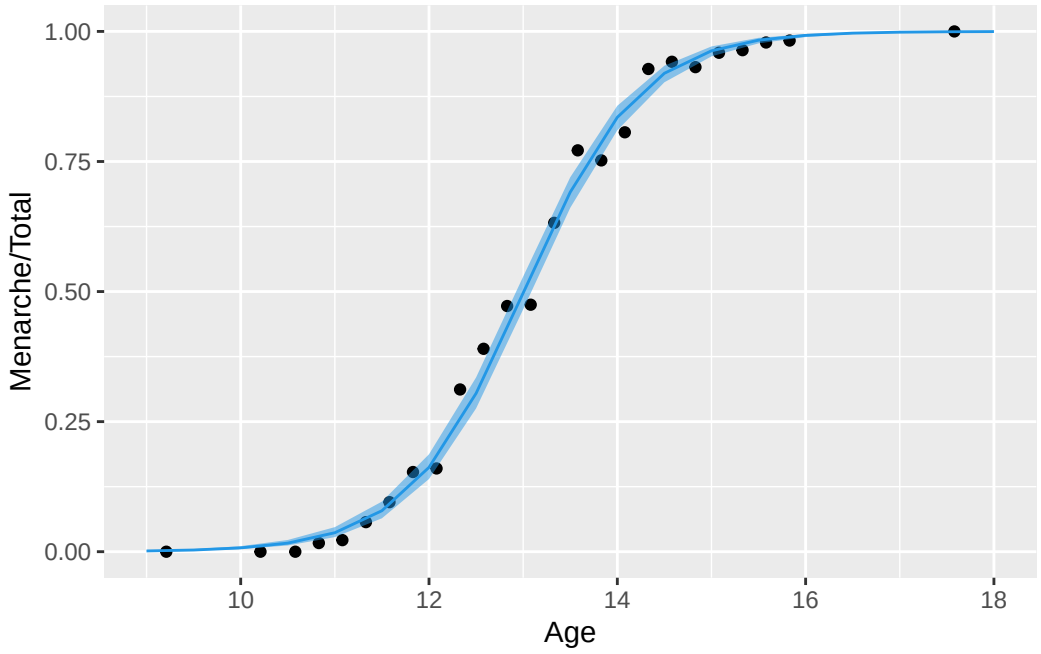
```
5 11.08    90        2
6 11.33    88        5
```

- Odhad parametrov a predpovede

```
# model
fit <- glm(cbind(Menarche, Total - Menarche) ~ Age, dat, family = binomial)
# alternatívne: glm(Menarche/Total ~ Age, dat, family = binomial, weights = dat$Total)
fit
```

```
Call:  glm(formula = cbind(Menarche, Total - Menarche) ~ Age, family = binomial,
    data = dat)

Coefficients:
(Intercept)          Age
    -21.226        1.632

Degrees of Freedom: 24 Total (i.e. Null);  23 Residual
Null Deviance:      3694
Residual Deviance: 26.7     AIC: 114.8
```

```
# zobrazenie
linkinv <- fit$family$linkinv
data.frame(Age = seq(9, 18, by = 0.5)) |>
  (\(x) cbind(x, predict(fit, newdata = x, se.fit = TRUE)) )() |>
  transform(up = fit + 2*se.fit,  #  95% interval spoľahlivosti
            lo = fit - 2*se.fit,
            se.fit = NULL,
            residual.scale = NULL) |>
  dplyr::mutate(across(-Age, .fns = linkinv)) |>  # do pôvodnej mierky
  ggplot() + aes(x = Age) +
  geom_point(aes(y = Menarche/Total), data = dat) +
  geom_ribbon(aes(ymin = lo, ymax = up), fill = 4, alpha = 0.5) +
  geom_line(aes(y = fit), color = 4)
```



- Pre modelovanie spojitej premennej predstavujúcej pomer v *otvorenom intervale* $(0, 1)$ je vhodná tzv. beta-regresia (v R pomocou balíku *betareg*), taký model však už nepatrí do triedy GLM. Pre veľké n_i dáva podobné výsledky ako GLM.

7.4.4 Poissonova regresia

- Poissonovo rozdelenie s pravdepodobnostnou funkciou $p(y) = \frac{e^{-\mu}\mu^y}{y!}$ a momentami $E[Y] = var[Y] = \mu$ vznikne z binomického rozdelenia, keď pravdepodobnosť úspechu v jednom pokuse je stlačená smerom k nule a naopak, počet pokusov rastie k nekonečnu, takže stredná hodnota zostáva rovnaká.
- To robí Poissonovo rozdelenie vhodné na modelovanie premenných predstavujúcich *počet* (angl. count) *bez horného ohraničenia*.
- Pretože stredná hodnota musí byť nezáporná, prirodzenou voľbou link funkcie je $g(\mu) = \ln(\mu)$. Keďže $g'(\mu) = 1/\mu$, potom váhy budú $w = \mu$.
- V bežnej regresii by sme s rastúcim rozptylom zmenšovali váhu pozorovania, no tu je to naopak, pretože väčšie stredné hodnoty počtu poskytujú viac informácií o parametroch.

Príklad (bicykle)

- S pomocou súboru údajov *ISLR2::Bikeshare* modelujeme počet požičaných bicyklov za hodinu pomocou vonkajšej teploty – preškálovanej vzťahom $temp = \frac{T - T_{\min}}{T_{\max} - T_{\min}}$.

- Dáta sú obmedzené na jarné obdobie počas víkendov za slnečného počasia. Ukážka:

```
# data
dat <- ISLR2::Bikeshare |>
  dplyr::filter(season == 2,
                holiday == 0,
                workingday == 0,
                weathersit == "clear",
               #  hr %in% as.character(8:19)
               TRUE)
dat |> dplyr::select(temp, bikers) |> head(3)
```

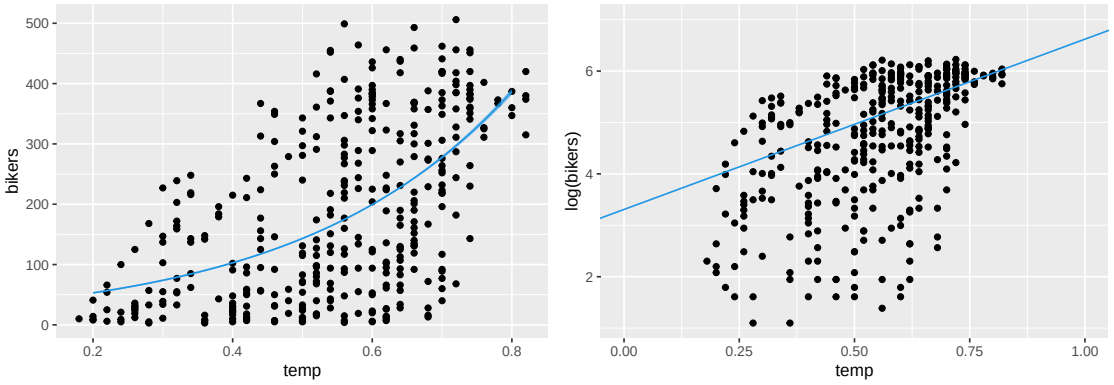
```
  temp bikers
1 0.26     27
2 0.22     25
3 0.20      9
```

- Odhad a predpovede:

```
# model
fit <- glm(bikers ~ temp, dat, family = poisson)
fit |> summary()
```

```

Call:
glm(formula = bikers ~ temp, family = poisson, data = dat)

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  3.30850    0.01873   176.6   <2e-16 ***
temp         3.31208    0.03024   109.5   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 45711  on 374  degrees of freedom
Residual deviance: 32270  on 373  degrees of freedom
AIC: 34727

Number of Fisher Scoring iterations: 5
```

```r
# zobrazenie
linkinv <- fit$family$linkinv
data.frame(temp = seq(0.2, 0.8, by = 0.01)) |>
  (\(x) cbind(x, predict(fit, newdata = x, se.fit = TRUE)) )() |>
  transform(up = fit + 2*se.fit,  #  95% interval spoľahlivosti
            lo = fit - 2*se.fit,
            se.fit = NULL,
            residual.scale = NULL) |>
  dplyr::mutate(across(-temp, .fns = linkinv)) |>  # do pôvodnej mierky
  ggplot() + aes(x = temp) +
  geom_point(aes(y = bikers), data = dat) +
  geom_ribbon(aes(ymin = lo, ymax = up), fill = 4, alpha = 0.5) +
  geom_line(aes(y = fit), color = 4)
dat |>
  ggplot() + aes(x= temp, y = log(bikers)) + geom_point() +
  geom_abline(intercept = fit$coefficients[1], slope = fit$coefficients[2],
              color = 4) +
  expand_limits(x = c(0,1), y = 7)
```



- Počet cyklistov pri minimálnej teplote by mal byť v priemere $e^{3.308} \approx 27$. Pri maximálnej teplote ($temp = 1$) by ich malo byť $e^{3.312} = 27.4$-násobne viac. Zvýšenie teploty o 10% jej bežného rozsahu ($\Delta temp = 0.1$) by podľa modelu malo priniesť $e^{3.312/10} \approx 1.4$-násobný nárast počtu cyklistov (čiže o 40% vyšší).
- Z grafu je vidieť, že rozptyl so strednou hodnotou stúpa, čo zodpovedá predpokladom modelu.
- Všimnime si však aj to, že (reziduálna) deviancia je až cca 100-násobne vyššia oproti počtu stupňov voľnosti. (Ak $D \sim \chi^2(k)$ potom $E[D] = k$ a $var[D] = 2k$.) To naznačuje oveľa vyšší rozptyl, než s akým model na základe predpokladaného rozdelenia Y počíta. Je to badať aj z príliš úzkeho pásu spoľahlivosti okolo regresnej

krivky.

7.5 Modelovanie rozptylu

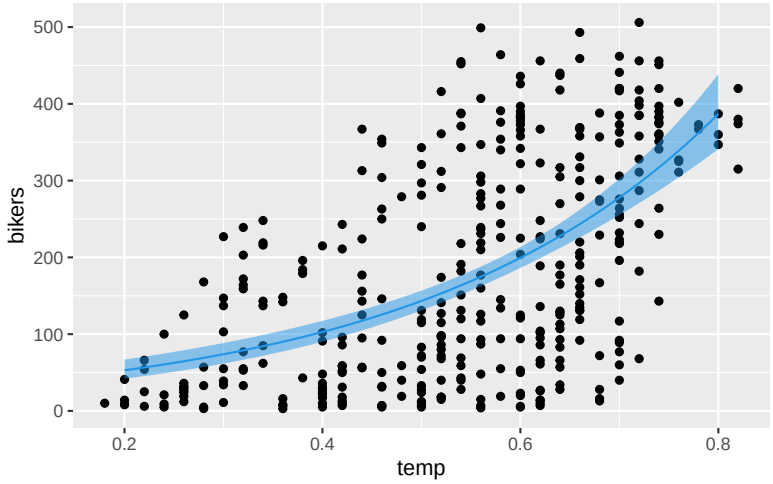
- S voľbou triedy (podmieneného) rozdelenia Y dostaneme aj variančnú funkciu $V(\mu(x))$. Skutočná podmienená variancia $var[Y|X = x]$ však tomu nemusí zodpovedať.
- Ak je skutočný rozptyl väčší, potom proces je označený ako *over-dispersed*. V opačnom prípade *under-dispersed*. Prvý prípad je častejší a tiež závažnejší.
- Vyšší rozptyl je veľa krát spôsobený nejakým zanedbaným faktorom (chýbajúci prediktor alebo autokorelované pozorovania).
- Ak pôvod prebytočnej variancie nezahrnieme do modelu, ešte stále sa dá použiť symptomatická liečba a zvoliť rozdelenie, ktoré dokáže rozptyl škálovať podľa potreby.
- Takými sú rozdelenia z triedy tzv. kvázi-rozdelení.

Príklad (bicykle)

```
# model
fit <- glm(bikers ~ temp, dat, family = quasipoisson)
fit |> summary()
```

```

Call:
glm(formula = bikers ~ temp, family = quasipoisson, data = dat)

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   3.3085     0.1687   19.62   <2e-16 ***
temp          3.3121     0.2722   12.17   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for quasipoisson family taken to be 81.0479)

    Null deviance: 45711  on 374  degrees of freedom
Residual deviance: 32270  on 373  degrees of freedom
AIC: NA

Number of Fisher Scoring iterations: 5
```

```r
# zobrazenie
linkinv <- fit$family$linkinv
data.frame(temp = seq(0.2, 0.8, by = 0.01)) |>
  (\(x) cbind(x, predict(fit, newdata = x, se.fit = TRUE)) )() |>
  transform(up = fit + 2*se.fit,  #  95% interval spoľahlivosti
            lo = fit - 2*se.fit,
            se.fit = NULL,
            residual.scale = NULL) |>
  dplyr::mutate(across(-temp, .fns = linkinv)) |>  # do pôvodnej mierky
  ggplot() + aes(x = temp) +
  geom_point(aes(y = bikers), data = dat) +
  geom_ribbon(aes(ymin = lo, ymax = up), fill = 4, alpha = 0.5) +
  geom_line(aes(y = fit), color = 4)
```



- Všimnime si rádovo vyššie štandardné chyby parametrov, na základe ktorých je teraz test významnosti vplyvu prediktorov na odozvu podstatne spoľahlivejší.

- Neohraničené počty s vyšším rozptylom sa zvyknú modelovať aj pomocou negatívneho binomického rozdelenia (v R napr. pomocou funkcie *MASS::glm.nb*).

8 Generatívne metódy klasifikácie

8.1 Úvod

- Diskriminačné metódy ako napr. logistická regresia modelujú podmienenú pravdepodobnosť $Pr[Y = k|\vec{X} = \vec{x}]$ *priamo*.
- Generatívne metódy idú na problém *nepriamo*:
  1. Najprv sa modeluje rozdelenie vysvetľujúcich premenných $\vec{X} = (X_1, X_2, \ldots, X_p)$ zvlášť pre každú hodnotu odozvy Y.
  2. Pomocou Bayesovej vety sa potom *konvertuje* na podmienené rozdelenie odozvy.
- Ak je rozdelenie $\vec{X}$ normálne, oba prístupy dávajú veľmi podobný výsledok, generatívny je však
  - s menším počtom údajov presnejší,
  - pri dokonalej separácii hodnôt netrpí nestabilitou odhadu parametrov (tak ako logistická regresia).
- Generatívny prístup je jednoducho a prirodzene rozšíriteľný na viac ako 2 triedy.
- Nech
  - odozva Y je kvalitatívna premenná s $K \geq 2$ hodnotami (bez definovaného poradia),
  - π_k je pravdepodobnosť, že náhodne zvolené pozorovanie patrí do triedy k, teda $Pr[Y = k]$. V bayesovskom kontexte sa nazýva *apriórnou*.
  - Funkcia $f_k(X)$ bude označovať
    - (v prípade diskrétnej X) pravdepodobnostnú funkciu, $f_k(\vec{x}) = Pr[\vec{X} = \vec{x}|Y = k]$,
    - (v spojitom prípade) hustotu pravdepodobnosti, čiže $f_k(\vec{x})dx_1dx_2 \ldots dx_p = Pr[\vec{X} = \vec{x}|Y = k]$,

    takže bude nadobúdať veľké hodnoty, ak pre pozorovanie z k-tej triedy bude $\vec{X} \approx \vec{x}$, a naopak, malé, ak hodnoty $\vec{X}$ sú ďaleko od $\vec{x}$.

- Potom podľa Bayesovej vety platí vzťah pre pravdepodobnosť, že pozorovanie $\vec{X} = \vec{x}$ patrí do k-tej triedy,
$$Pr[Y = k|\vec{X} = \vec{x}] = \frac{\pi_k f_k(\vec{x})}{\sum_{l=1}^{K} \pi_l f_l(\vec{x})}.$$
Nazýva sa aj *aposteriórnou* pravdepodobnosťou.
- Takže namiesto priameho odhadu $Pr[Y = k|\vec{X} = \vec{x}]$ (ako pri diskriminatívnom prístupe) nám stačí iba odhadnúť π_k, f_k a dosadiť do vzťahu.
  - Odhad apriórnej pravdepodobnosti je jednoducho iba podiel počtu pozorovaní v konkrétnej skupine ku počtu všetkých, teda $\pi_k = \frac{n_k}{n}$.
  - Odhad hustoty f_k je už *náročnejší*, a aby sa dal rozumne rýchlo vypočítať, budeme potrebovať *zjednodušujúce predpoklady*.
- V nasledujúcom si preberieme tri (generatívne) klasifikátory, ktoré používajú rôzne odhady hustoty f_k.

8.2 Lineárna diskriminačná analýza

8.2.1 Jeden prediktor

- V lineárnej diskriminačnej analýze (LDA) predpokladáme, že f_k je hustota *normálneho* rozdelenia,
$$f_k(x) = \frac{1}{\sqrt{2\pi}\sigma_k} \exp\left(-\frac{(x-\mu_k)^2}{2\sigma_k^2}\right),$$
kde μ_k a σ_k^2 sú stredná hodnota a rozptyl v k-tej triede.
- Ďalej predpokladáme, že $\sigma_1 = ... = \sigma_K = \sigma$.
- Po dosadení takto špecifikovanej hustoty do Bayesovej vety dostávame
$$Pr[Y = k|X = x] = \frac{\pi_k \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(x-\mu_k)^2}{2\sigma^2}\right)}{\sum_{l=1}^{K} \pi_l \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(x-\mu_l)^2}{2\sigma^2}\right)}.$$
- Keďže (bayesovský) klasifikátor zaradí pozorovanie $X = x$ do tej triedy, ktorej zodpovedá najvyššia aposteriórna pravdepodobnosť, stačí pravdepodobnosť nahradiť jednoduchšou funkciou, tzv. *diskriminačnou* funkciou δ_k. Dostaneme ju logaritmovaním vzťahu pre $Pr[Y = k|X = x]$ a vynechaním členov spoločných pre všetky triedy, takže stále platí
$$\delta_k(x) \propto Pr[Y = k|X = x]$$

(čiže *zachová proporcionalitu* medzi skupinami). Nakoniec

$$\delta_k(x) = x\frac{\mu_k}{\sigma^2} - \frac{\mu_k^2}{2\sigma^2} + \ln \pi_k.$$

Diskriminačná funkcia je lineárna v premennej x, čo vysvetľuje názov metódy.

- Parametre skutočného rozdelenia sotvakedy poznáme, najlepšie odhady sú výberový priemer a (vážený) rozptyl

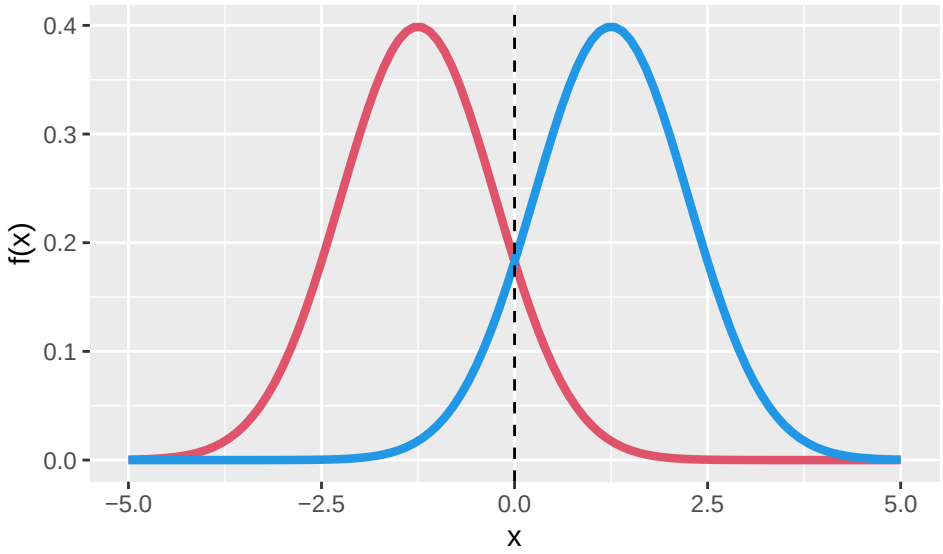
$$\hat{\mu}_k = \frac{1}{n_k} \sum_{i:\ y_i=k} x_i$$

$$\hat{\sigma}^2 = \frac{1}{n-K} \sum_{k=1}^{K} \sum_{i:\ y_i=k} (x_i - \hat{\mu}_k)^2 = \frac{1}{n-K} \sum_{k=1}^{K} (n_k - 1)\sigma_k^2$$

Ilustrácia (LDA)

- Pre $K = 2$ a $\pi_1 = \pi_2$, z rovnice $\delta_1(x) = \delta_2(x)$ dostaneme intuitívnu hranicu (decision boundary) medzi triedami v bode $x = \frac{\mu_1+\mu_2}{2}$.
- Teoretická hodnota rozhodovacej hranice pre rozdelenia $N(-1.25, 1^2)$ a $N(1.25, 1^2)$ je teda nula.

```
pars <- list(list(mean=-1.25, sd=1),
             list(mean=1.25, sd=1))

ggplot() +
  geom_function(fun = dnorm, args = pars[[1]], color = 2, linewidth = 1.5) +
  geom_function(fun = dnorm, args = pars[[2]], color = 4, linewidth = 1.5) +
  geom_vline(xintercept = 0, linetype=2) +
  xlab("x") + ylab("f(x)") +
  xlim(-5, 5)
```

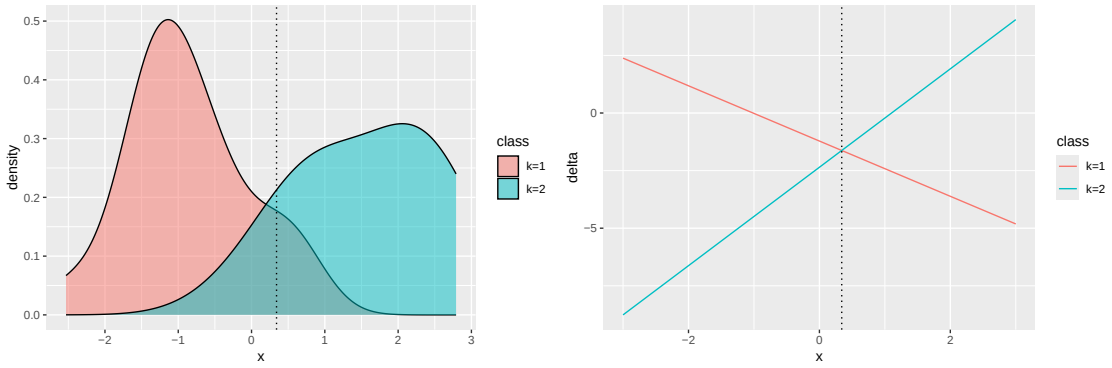


- Odhad hranice bude závisieť od náhodného výberu z rozdelení.

```r
# data
set.seed(4)
dat <- list("k=1" = data.frame(x = rnorm(20, mean = pars[[1]]$mean, sd = pars[[1]]$sd)),
            "k=2" = data.frame(x = rnorm(20, mean = pars[[2]]$mean, sd = pars[[2]]$sd))
           ) |>
  dplyr::bind_rows(.id = "class")
# súhrny
tmp1 <- dat |>
  dplyr::group_by(class) |>
  dplyr::summarise(mu = mean(x),
                   s2 = var(x),
                   n = dplyr::n()
                   ) |>
  dplyr::ungroup()
tmp2 <- tmp1 |>
  dplyr::summarise(mu = weighted.mean(mu, w = n),
                   s2 = weighted.mean(s2, w = n-1), # pooled variance
                   n = sum(n)
                   )
# priesečník
tmp2$boundary <- tmp2$mu - diff(log(tmp1$n/tmp2$n)) * tmp2$s2 / diff(tmp1$mu)

# report
dplyr::full_join(tmp1, rbind(tmp2), dplyr::join_by(mu, s2, n)) |>
  pander::pander()
```

class	mu	s2	n	boundary
k=1	-0.8737	0.6421	20	NA
k=2	1.556	0.8149	20	NA
NA	0.3411	0.7285	40	0.3411

```r
# graf vyhladenej empirickej hustoty
dat |>
  ggplot() + aes(x = x, fill = class) +
  geom_density(alpha = 0.5, adjust = 1.5) +
  geom_vline(xintercept = tmp2$boundary, linetype = 3)
# graf diskriminačných funkcií
LDfun <- Vectorize(
  function(x, mu, s2, pi) x * mu/s2 - mu^2/(2*s2) + log(pi),
  vectorize.args = "x")
xgrid <- c(-3,3)
LDfun(xgrid, mu = tmp1$mu, s2 = tmp2$s2, pi = tmp1$n/tmp2$n) |>
  t() |> as.data.frame() |> setNames(tmp1$class) |>
  cbind(x = xgrid) |>
  tidyr::pivot_longer(-x, names_to = "class", values_to = "delta") |>
  ggplot() + aes(x = x, y = delta, group = class, color = class) + geom_line() +
  geom_vline(xintercept = tmp2$boundary, linetype = 3)
```



8.2.2 Viac prediktorov

- Rozšírenie modelu o ďalšie vysvetľujúce premenné so sebou prináša zahrnutie predpokladov nielen o ich marginálnom ale aj o ich *združenom rozdelení* pravdepodobnosti.
- LDA predpokladá združené normálne rozdelenie, ktoré v sebe implicitne obsahuje aj normálne marginálne rozdelenia.
- Nech $\vec{X} = (X_1, \ldots, X_p)$ je náhodný vektor vysvetľujúcich premenných. Potom jeho rozdelenie pre $Y = k$, je formálne $\vec{X} \sim N(\vec{\mu}_k, \ )$ s hustotou definovanou vzťahom

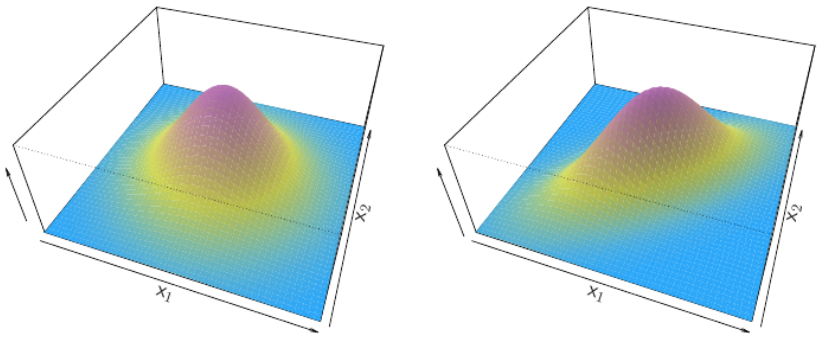
$$f_k(\vec{x}) = \frac{1}{(2\pi)^{p/2}|\ |^{1/2}} \exp\left(-\frac{1}{2}(\vec{x} - \vec{\mu}_k)^T \ ^{-1}(\vec{x} - \vec{\mu}_k)\right)$$

  kde parameter

– $\vec{\mu}_k = E[\vec{X}|Y = k]$ je vektor stredných hodnôt a
– $= cov[\vec{X}|Y = 1] = ... = cov[\vec{X}|Y = K]$ je $p \times p$ kovariančná matica.

Ilustrácia (2D normálne rozdelenie)

- Príklad hustoty dvojrozmerného združeného normálneho rozdelenia s identickými varianciami a
  – diagonálnou kovariančnou maticou (na orázku vľavo),
  – kladnou kovarianciou (vpravo).

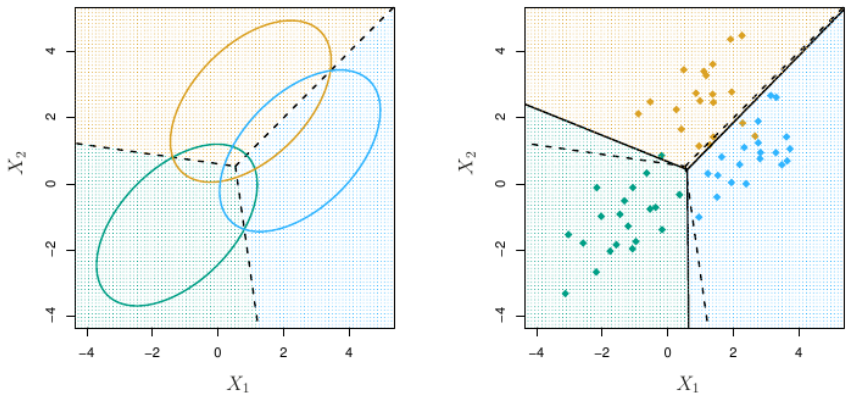


- Diskriminačná funkcia potom nadobudne tvar

$$\delta_k(\vec{x}) = \vec{x}^T \, ^{-1}\vec{\mu}_k - \frac{1}{2}\vec{\mu}_k^T \, ^{-1}\vec{\mu}_k + \ln \pi_k$$

Ilustrácia (LDA)

- Príklad pre 3 triedy a 2 prediktory so strednými hodnotami $\vec{\mu}_1 = (1, 2.5)^T$, $\vec{\mu}_2 = (2.5, 1)^T$, $\vec{\mu}_3 = (-1.25, -1.25)^T$ a kovariančnou maticou $= \begin{pmatrix} 1 & 0.5 \\ 0.5 & 1 \end{pmatrix}$. Z každého rozdelenia pochádza 20 pozorovaní, čiarkované sú teoretické rozhodovacie hranice, plné sú určené na základe pozorovaní.



8.3 Kvadratická diskriminačná analýza

- QDA zovšeobecňuje LDA v tom, že umožňuje každej skupine mať *vlastnú* kovariančnú maticu $_k = cov[\vec{X}|Y = k]$, takže

$$(\vec{X}|Y = k) \sim N(\vec{\mu}_k, \ _k).$$

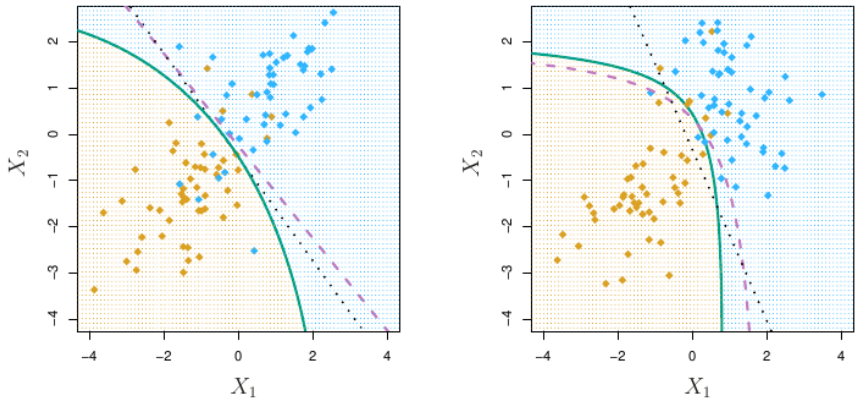
- V dôsledku toho má diskriminačná funkcia tvar

$$\begin{aligned}\delta_k(\vec{x}) &= -\frac{1}{2}(\vec{x} - \vec{\mu}_k)^T \ _k^{-1}(\vec{x} - \vec{\mu}_k) - \frac{1}{2}\ln|\ _k^{-1}| + \ln \pi_k \\ &= -\frac{1}{2}\vec{x}^T \ _k^{-1}\vec{x} + \vec{x}^T \ _k^{-1}\vec{\mu}_k - \frac{1}{2}\vec{\mu}_k^T \ _k^{-1}\vec{\mu}_k - \frac{1}{2}\ln|\ _k^{-1}| + \ln \pi_k\end{aligned}$$

  čiže je *kvadratickou* funkciou argumentu x.
- Pri QDA treba odhadnúť $Kp(p+1)$ parametrov, zatiaľčo pri LDA iba Kp parametrov. Voľba medzi nimi je často otázkou bias-variance kompromisu.
- Zhruba sa však dá povedať, že LDA zvolíme pri menšom počte dát, kedy je dôležité redukovať rozptyl modelu. Naproti tomu QDA sa odporúča pri veľkej trénovacej vzorke (takže rozptyl modelu nie je veľký problém), alebo keď je zjavný rozdiel medzi kovariančnými maticami v jednotlivých triedach.

Ilustrácia (LDA, QDA)

- Príklad s 2 triedami a 2 prediktormi: čiarkovanou fialovou je skutočná hranica medzi triedami, bodkovanou čiernou LDA a plnou zelenou QDA rozhodovacia hranica. Všimnime si rozdiel v tvare/orientácii mraku bodov na pravom obrázku.



8.4 Naivný Bayesov klasifikátor

- Namiesto predpokladu, že združená hustota f_k je z konkrétnej parametrickej triedy (z normálneho rozdelenia v prípade diskriminačnej analýzy), naivný Bayesov klasifikátor predpokladá iba **nezávislosť prediktorov** $X_1, \dots, X_p$, čiže

$$f_k(\vec{x}) = f_{k1}(x_1) \cdot f_{k2}(x_2) \cdot \dots \cdot f_{kp}(x_p)$$

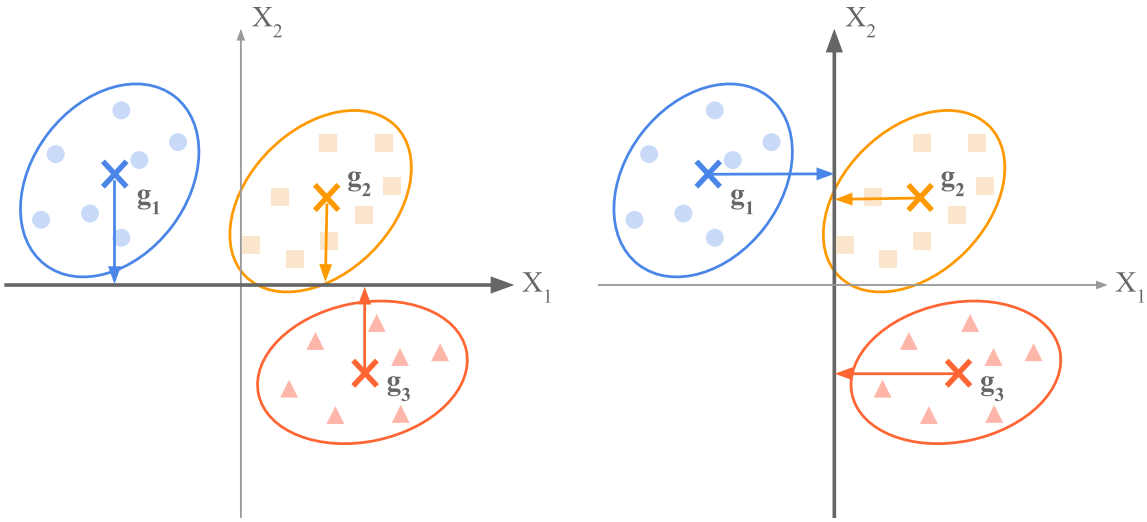
  kde f_{kj} je hustota rozdelenia premennej X_j ak $Y = k$.
- To znamená, že už netreba odhadovať parametre združeného rozdelenia zodpovedné za vzťahy medzi prediktormi (v prípade normálneho rozdelenia to boli kovariancie).
- Nezávislosti prediktorov samozrejme veľmi neveríme, preto názov *naivný*, no tento predpoklad slúži na uľahčenie výpočtu a často vedie k slušným výsledkom. Voľba je opäť otázkou bias-variance kompromisu.
- Odhad f_{kj} závisí od typu premennej, máme niekoľko možností.
  - Ak je X_j kvantitatívna, potom môžeme predpokladať normalitu $(X_j|Y = k) \sim N(\mu_{kj}, \sigma^2_{kj})$. Vtedy je to špeciálny prípad QDA s diagonálnymi kovariančnými maticami. Namiesto normálneho však môžeme predpokladať akékoľvek iné parametrické, spojité rozdelenie.
  - Ak je X_j kvantitatívna, ale nechceme model viazať predpokladom o parametrickej triede rozdelenia, môžeme použiť histogram alebo jadrový odhad hustoty.
  - Ak je X_j kvalitatívna, potom nám stačí rozdelenie odhadnúť relatívnou početnosťou v skupinách.

8.5 Geometrický pohľad na LDA

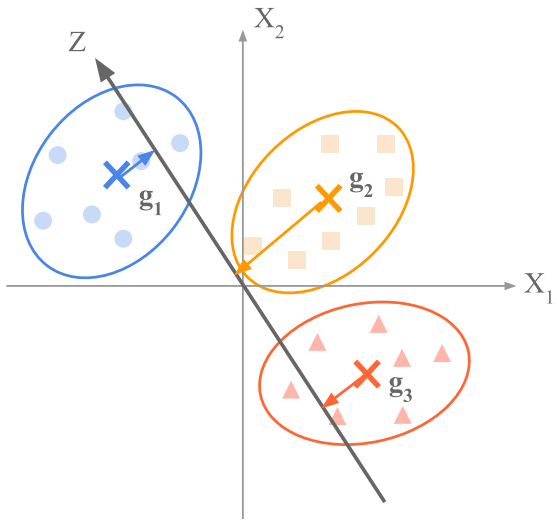
8.5.1 Hľadanie vhodného podpriestoru

8.5.1.1 Problém

- Klasifikačný problém sa dá riešiť aj geometricky. Je základom špeciálneho prípadu LDA, ktorý sa volá *Kanonická diskriminačná analýza* (canonical discrimination analysis, CDA). Vyžaduje rovnaké apriórne pravdepodobnosti aj kovariančné matice naprieč skupinami.
- Pre jednoduchosť majme dva prediktory X_1, X_2 a trojhodnotovú odozvu Y (t.j. 3 skupiny).
- Z prieskumného/popisného hľadiska hľadáme taký podpriestor, v ktorom by boli skupiny dobre rozlíšiteľné.
- Projekcia na jednu či druhú os nie je vo všeobecnosti najlepšia.



- Hľadáme takú os (sprostredkovanú takým vektorom $\mathbf{u}$), že lineárna kombinácia $Z = u_1X_1 + u_2X_2$ pomôže skupiny oddeliť najlepšie



8.5.1.2 Rozptyl *v* skupinách a *medzi* skupinami

- Celková variancia sa dá rozložiť na varianciu *vo vnútri* skupín (angl. *w*ithin) a varianciu *medzi* skupinami (angl. *b*etween):

$$\mathbf{V} = \mathbf{W} + \mathbf{B}$$

kde

- celková variancia je daná vzťahom $\mathbf{V} = \frac{1}{n-1}\mathbf{X}^T\mathbf{X}$,
- variancia vo vnútri skupín sa skladá z jednotlivých variančných matíc,

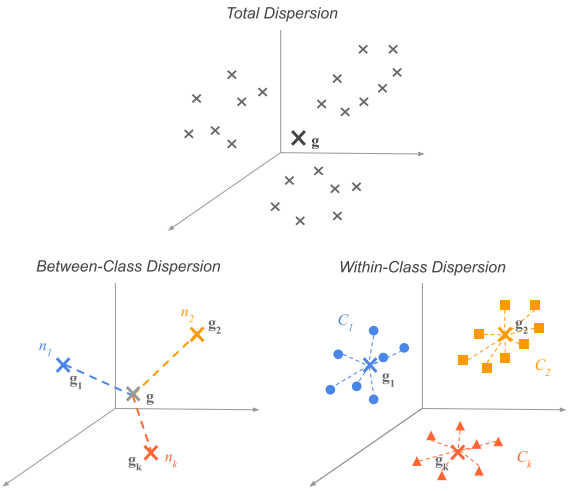
$$\mathbf{W} = \sum_{k=1}^{K} \frac{n_k - 1}{n - 1}\mathbf{W}_k$$

pričom $\mathbf{W}_k = \frac{1}{n_k-1}\mathbf{X}_k^T\mathbf{X}_k$,

- variancia medzi skupinami vyjadruje variabilitu centroidov $\mathbf{g}_k$ jednotlivých skupín okolo globálneho centroidu (ťažiska) $\mathbf{g}$, takže platí

$$\mathbf{B} = \sum_{k=1}^{K} \frac{n_k}{n - 1}(\mathbf{g}_k - \mathbf{g})(\mathbf{g}_k - \mathbf{g})^T$$

kde $\mathbf{g} = \sum_{k=1}^{K} \frac{n_k}{n}\mathbf{g}_k$, a $\mathbf{g}_k = (\bar{x}_{k1}, \ldots, \bar{x}_{kp})^T$, pričom $\bar{x}_{kj}$ je priemer j-tej premennej v k-tej skupine.

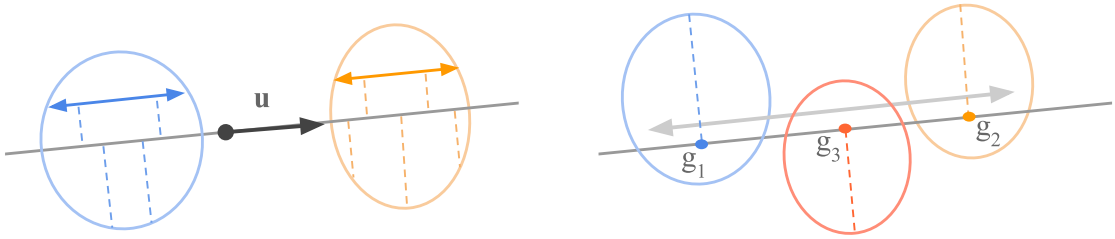


8.5.1.3 Reformulácia problému

- Z rozkladu celkovej variancie dostávame aj rozklad kvadratickej formy

$$\mathbf{u}^T\mathbf{V}\mathbf{u} = \mathbf{u}^T\mathbf{W}\mathbf{u} + \mathbf{u}^T\mathbf{B}\mathbf{u}$$

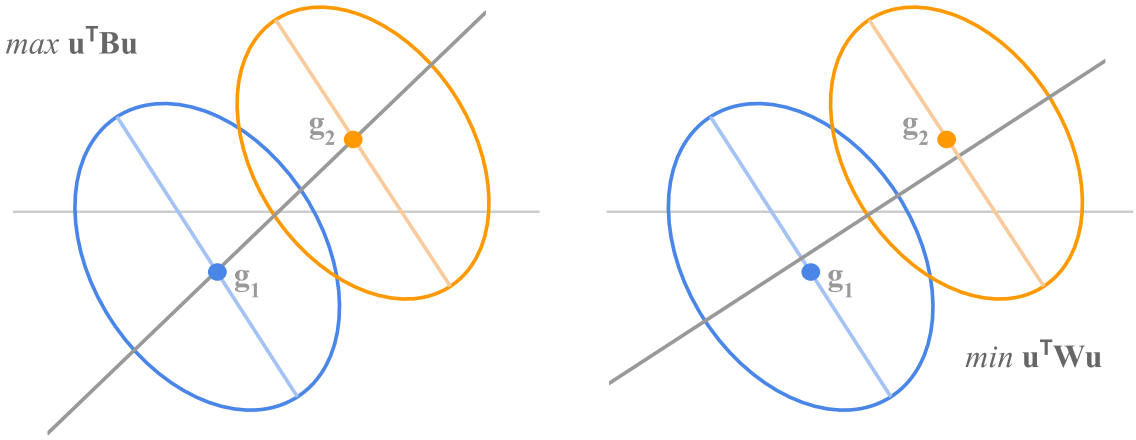
- Algebraicky, hľadáme takú lineárnu kombináciu prediktorov, ktorá (ideálne)
  - minimalizuje rozptyl vo vnútri skupín: $\min\{\mathbf{u}^T\mathbf{W}\mathbf{u}\}$
  - maximalizuje rozptyl medzi skupinami: $\max\{\mathbf{u}^T\mathbf{B}\mathbf{u}\}$



- To je však (vo všeobecnosti) nemožné dosiahnuť súčasne.
- Východiskom z patovej situáce je kompromis.

8.5.1.4 Kompromisné kritérium

- Sú dve možnosti:



  - korelačný pomer $\max\left\{\frac{\mathbf{u}^T\mathbf{Bu}}{\mathbf{u}^T\mathbf{Vu}}\right\}$
  - F-pomer $\max\left\{\frac{\mathbf{u}^T\mathbf{Bu}}{\mathbf{u}^T\mathbf{Wu}}\right\}$
- Oba optimalizačné problémy sa riešia rovnako, ukážeme si prvý z nich.
- Najprv je výhodné zaviesť normalizačné obmedzenie $\mathbf{u}^T\mathbf{Vu} = 1$, takže problém sa zmení na hľadanie extrému funkcie s podmienkou, a to metódou Lagrangeových multiplikátorov,

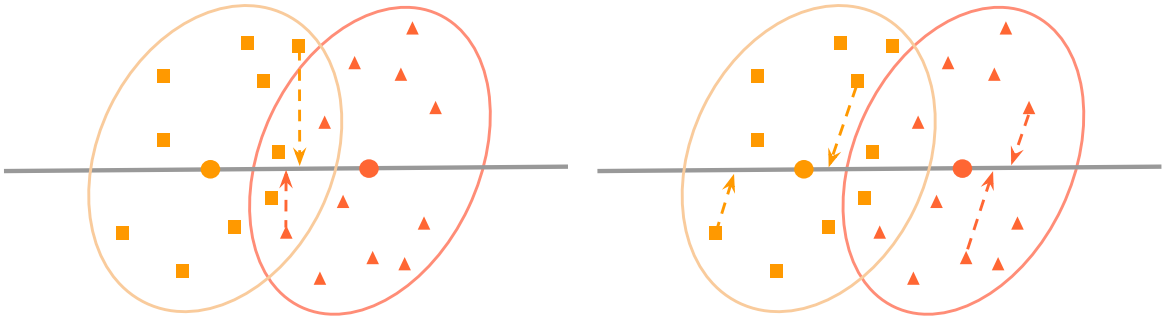
$$
\begin{aligned}
L(\mathbf{u}) &= \mathbf{u}^T\mathbf{Bu} - \lambda(\mathbf{u}^T\mathbf{Vu} - 1) \\
\frac{\partial L(\mathbf{u})}{\partial \mathbf{u}} &= 2\mathbf{Bu} - 2\lambda\mathbf{Vu} \\
2\mathbf{B}\hat{\mathbf{u}} - 2\lambda\mathbf{V}\hat{\mathbf{u}} &= 0 \\
\mathbf{Bu} &= \lambda\mathbf{Vu} \\
\mathbf{V}^{-1}\mathbf{Bu} &= \lambda\mathbf{u}
\end{aligned}
$$

takže náš hľadaný optimálny vektor $\mathbf{u}$ je *vlastný vektor* matice $\mathbf{V}^{-1}\mathbf{B}$.
(Pre detaily riešenia problému zovšeobecnených vlastných čísel pozri Ghojogh et al., 2022.)

8.5.1.5 Šikmá projekcia

- Princíp hľadania vhodného podpriestoru využíva aj tzv. analýza hlavných komponentov (angl. principal component analysis, PCA), čo je metóda učenia bez asistencie. Hľadá takú lineárnu transformáciu, aby osi boli orientované v smere postupne od najväčšieho rozptylu údajov až po najmenší rozptyl.

- Diskriminačná analýza môže v špeciálnom prípade pripomínať PCA vykonanú na centroidoch, avšak na rozdiel od nej, diskriminačná analýza body jednotlivých pozorovaní zobrazuje na osi šikmo, v smere hlavnej poloosi elipsoidu spoľahlivosti, čiže v smere vektora $\mathbf{u}$.



- Po nájdení prvej významnej osi (canonical axis) asociovanej s prvým optimálnym vektorom $\mathbf{u}$ môžeme hľadať druhú, ktorá je s predošlou *nekorelovaná* a je čo najviac diskriminačná (v smere čo najväčšieho rozptylu, aby sa ukázali ďalšie rozdiely medzi triedami odozvy), atď.
- Najväčší počet hľadaných osí sa rovná $\min(K-1, p)$.

8.5.2 Vzdialenosť a klasifikácia

8.5.2.1 Metrika

- Euklidovská vzdialenosť ľubovoľných dvoch vektorov $\mathbf{a}, \mathbf{b} \in \mathbb{R}^p$ sa dá vyjadriť pomocou skalárneho súčinu (a jednoduchej metriky danej jednotkovou maticou $\mathbf{I}_p$)

$$d^2(\mathbf{a}, \mathbf{b}) = (\mathbf{a}-\mathbf{b})^T(\mathbf{a}-\mathbf{b}) = (\mathbf{a}-\mathbf{b})^T\mathbf{I}_p(\mathbf{a}-\mathbf{b})$$

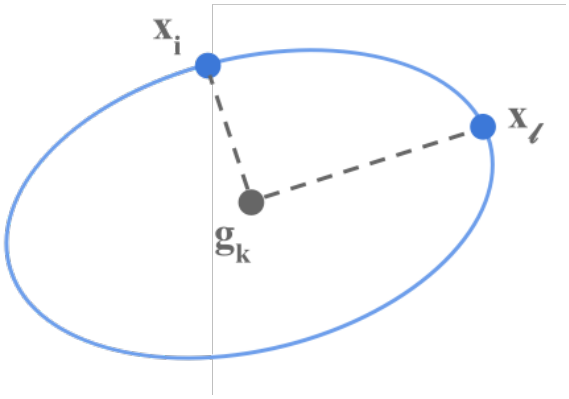
- Vo všeobecnosti môže byť vzdialenosť daná nejakou **metrikou**, ktorú reprezentuje matica $\mathbf{M}$,

$$d^2_{\mathbf{M}}(\mathbf{a}, \mathbf{b}) = (\mathbf{a}-\mathbf{b})^T\mathbf{M}(\mathbf{a}-\mathbf{b})$$

- Keď chceme zaradiť určitý bod $\mathbf{x}_i$ do jednej z tried, musíme porovnať vzdialenosti k ich centroidom $\mathbf{g}_k$ a vybrať tú najmenšiu. Namiesto euklidovskej metriky, ktorá nerešpektuje korelačnú štruktúru v skupinách, sa použije *Mahalanobisova metrika* s maticou $\mathbf{W}^{-1}$, pre ktorú je vzdialnosť daná

$$d^2(\mathbf{x}_i, \mathbf{g}_k) = (\mathbf{x}_i-\mathbf{g}_k)^T\mathbf{W}^{-1}(\mathbf{x}_i-\mathbf{g}_k)$$

- Napríklad, na nasledujúcom obrázku body $\mathbf{x}_i, \mathbf{x}_\ell$ ležia na tom istom elipsoide (t.j. vrstevnici hustoty rozdelenia). Mahalanobisova vzdialenosť je rovnaká, avšak euklidovská nie.

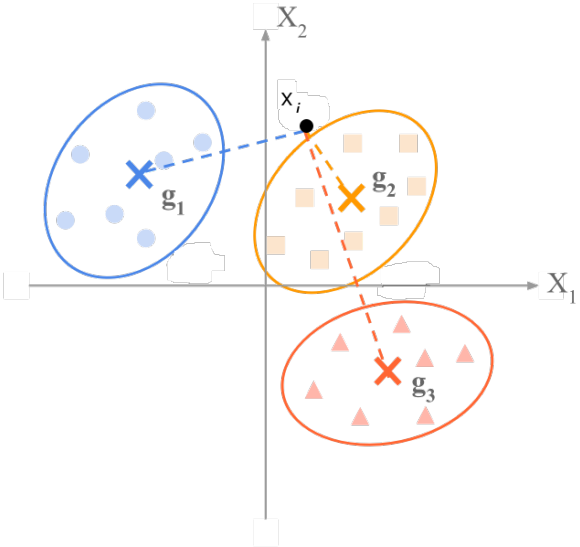


8.5.2.2 Klasifikácia

- Keď rozvinieme vzťah pre Mahalanobisovu vzdialenosť,

$$\begin{aligned} d^2(\mathbf{x}_i, \mathbf{g}_k) &= (\mathbf{x}_i - \mathbf{g}_k)^T \mathbf{W}^{-1} (\mathbf{x}_i - \mathbf{g}_k) \\ &= \underbrace{\mathbf{x}_i^T \mathbf{W}^{-1} \mathbf{x}_i}_{\text{konšt.}} - \underbrace{2\mathbf{x}_i^T \mathbf{W}^{-1} \mathbf{g}_k + \mathbf{g}_k^T \mathbf{W}^{-1} \mathbf{g}_k}_{\text{závisí od } k} \end{aligned}$$

  okamžite vidno podobnosť s diskriminačnou funkciou LDA.



8.6 Príklady

8.6.1 Lineárna diskriminačná analýza

Príklad (kosatce)

- Príklad s rozmermi korunného lupienku kosatcov: 2 prediktory (šírka a dĺžka lupienku), 3 triedy.

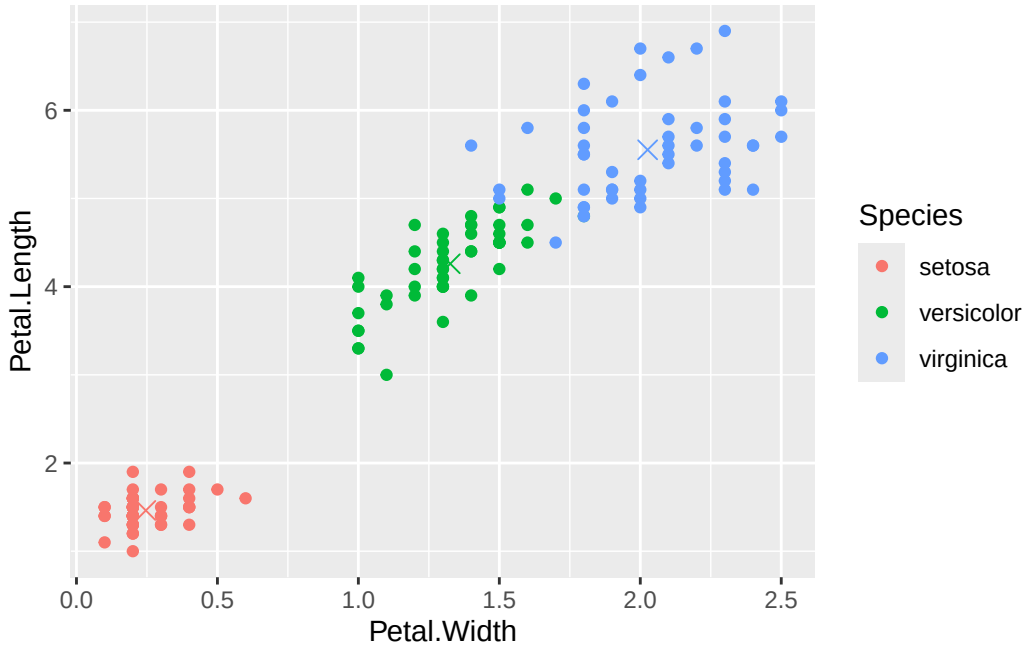
```
# odhad modelu
fit <- MASS::lda(Species ~ Petal.Width + Petal.Length, data = iris)
fit
```

```
Call:
lda(Species ~ Petal.Width + Petal.Length, data = iris)

Prior probabilities of groups:
    setosa versicolor  virginica
 0.3333333  0.3333333  0.3333333

Group means:
           Petal.Width Petal.Length
setosa           0.246        1.462
versicolor       1.326        4.260
virginica        2.026        5.552

Coefficients of linear discriminants:
                  LD1       LD2
Petal.Width  2.402394  5.042599
Petal.Length 1.544371 -2.161222

Proportion of trace:
   LD1    LD2
0.9947 0.0053
```

- Výstup z funkcie *MASS::lda* obsahuje
  - odhady apriórnych pravdepodobností π_k,
  - skupinové priemery μ_{kj} (centroidy $\vec{\mu}_k = (\mu_{k1}, \mu_{k2})$ aka $\mathbf{g}_k$),
  - transformačné vektory $\mathbf{u}_i$ (stĺpce) a nakoniec
  - percento vysvetlenosti medzi-skupinovej variability jednotlivými diskriminantami. Tu by nám na klasifikáciu zjavne stačil iba prvý z nich.

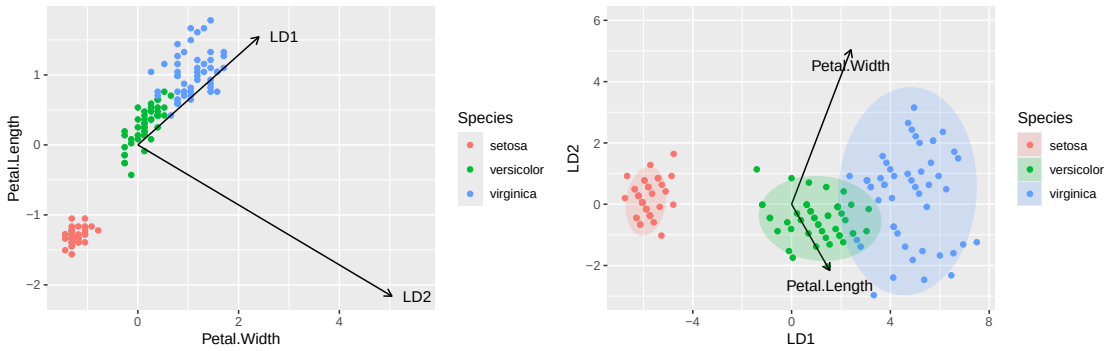
- Zobrazíme východiskovú situáciu s vypočítanými centroidmi.

```
# zobrazenie tried v priestore prediktorov aj s centroidmi
centr.df <- fit |>
  getElement("means") |> # treba ich ešte
  tibble::as_tibble(rownames = "Species")

iris |>
  ggplot() +
  aes(x = Petal.Width, y = Petal.Length, color = Species) +
  geom_point() +
  geom_point(data = centr.df, shape = 4, size = 3, show.legend = FALSE)
```



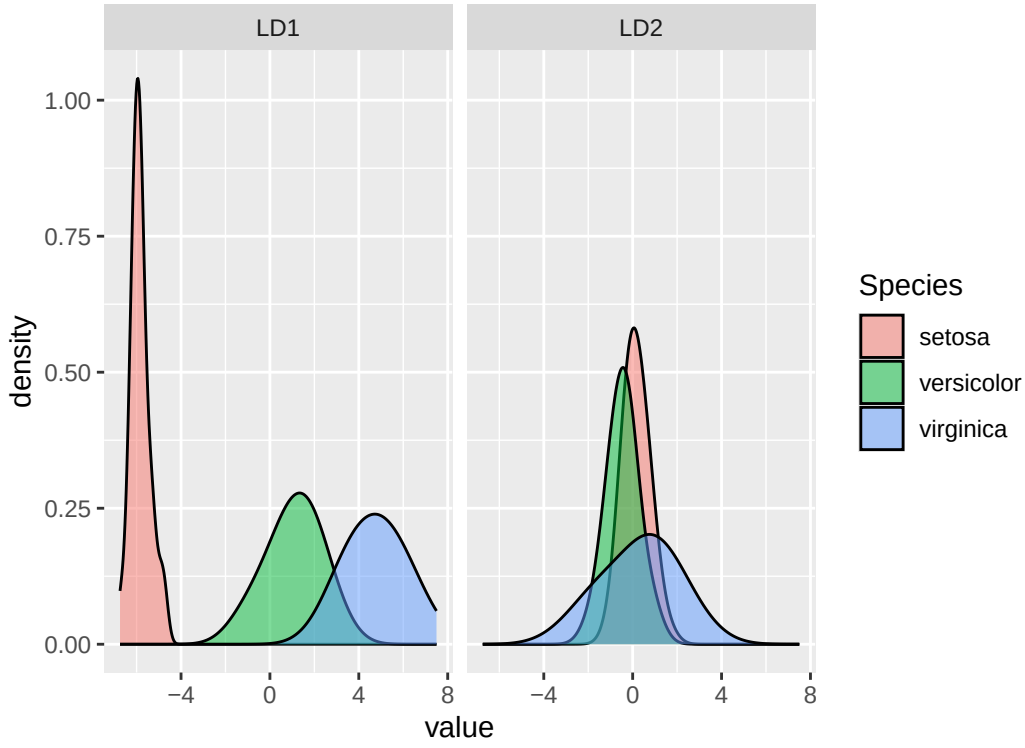
- Vidno, že triedy sú pomerne dobre vymedzené, ale v priestore lineárnych diskriminantov bude separácia ešte presnejšia.
- Pomocou tzv. *biplotu* (pozri detailnú publikáciu Greenacre, 2010) zobrazíme oba priestory premietnuté na sebe.

```r
# prvky transformačných vektorov "u" pre zobrazenie v priestore prediktorov
trans.df <- fit |>
  coef() |>   #  to isté ako getElement("scaling")
#  apply(2, function(x) x/sqrt(sum(x^2))) |>  # konverzia na jednotkové vektory (voliteľné)
  t() |>
  as.data.frame() |>
#  setNames(fit$scaling |> rownames()) |>   # lebo apply zruší názvy riadkov
  cbind(variable = fit$scaling |> colnames())

# biplot
iris |>
  purrr::map_if(is.numeric, scale) |>   # centrovanie a škálovanie
  as.data.frame() |>
  ggplot() + aes(x = Petal.Width, y = Petal.Length) +
  geom_point(aes(color = Species)) +
  geom_segment(aes(x = 0, y = 0, xend = Petal.Width, yend = Petal.Length),
               data = trans.df,
               arrow = arrow(length = unit(0.5, "char"))
               ) +
  geom_text(aes(x = Petal.Width, y = Petal.Length, label = variable),
            data = trans.df,
            nudge_x = 0.5)
# biplot pomocou externého balíku
# library(devtools)
# install_github('fawda123/ggord')
# ggord::ggord(fit, iris$Species, size = 2)

# prvky transformačných vektorov "u" pre zobrazenie v priestore diskriminantov
trans.df <- fit |>
  getElement("scaling") |>
  as.data.frame() |>
  tibble::rownames_to_column(var = "variable")

# výpočet hodnôt diskriminantov (aj pomocou predict(fit)$x)
LD <- iris[,4:3] |>
  as.matrix() |>
  (`%*%`)(fit$scaling) |>   # transformácia dát do priestoru diskriminantov
  apply(2, scale, center = TRUE, scale = FALSE) |> # centrovanie diskriminantov
  cbind.data.frame(iris[5])

# biplot v priestore diskriminantov
LD |>
  ggplot() + aes(LD1, LD2) +
  geom_point(aes(color = Species)) +
  geom_segment(aes(x = 0, y = 0, xend = LD1, yend = LD2),
               data = trans.df,
               arrow = arrow(length = unit(0.5, "char"))
               ) +
  geom_text(aes(x = LD1, y = LD2, label = variable),
            data = trans.df,
            nudge_y = -0.5) +
  stat_ellipse(aes(fill = Species), geom = "polygon", alpha = 0.2) +
  ylim(-3, 6)
```



- Vyhladená hustota hodnôt diskriminantov napovie, ako vyzerá výsledná, aposteriórna pravdepodobnosť jednotlivých tried.

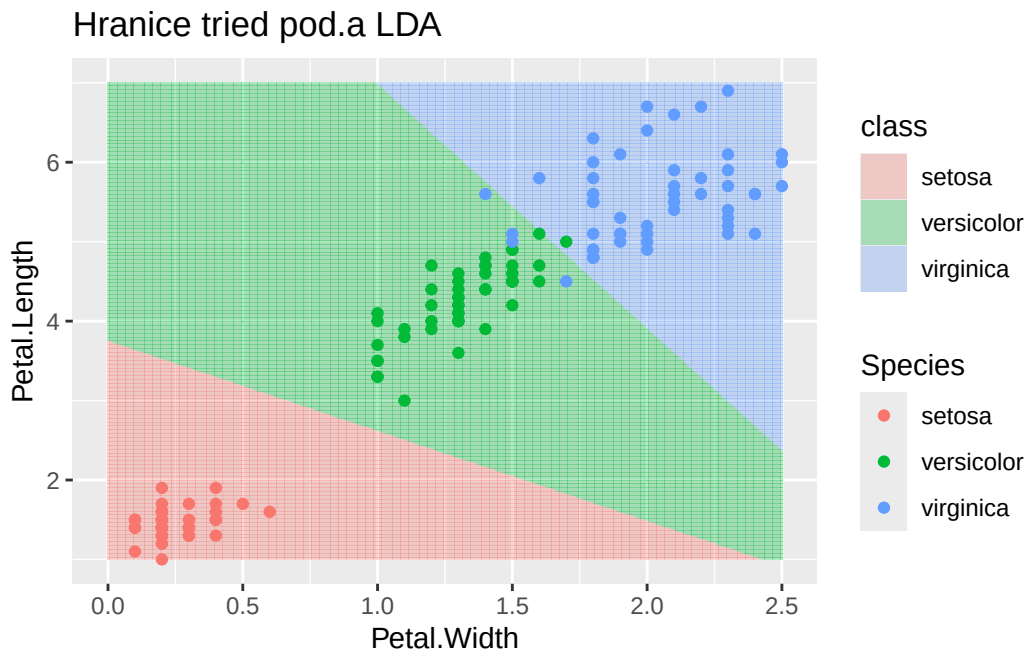
```
# hustota pravdepodobnosti podľa tried v priestore diskriminantov
LD |>
  tidyr::pivot_longer(-Species, names_to = "diskriminant", values_to = "value") |>
  ggplot() + aes(x = value, fill = Species) + geom_density(alpha = 0.5, adjust = 2) + facet
```



```
# empirická hustota v triedach pre prvý diskriminant vstavanou funkciou
# MASS::ldahist(iris[3:4], iris$Species)
```

- Nakoniec v pravidelnej mriežke hodnôt prediktorov (šírky a dĺžky lupienkov) vykonáme predpovede a zobrazíme delenie priestoru, takže je možná aj vizuálna klasifikácia.

```
# predikcia v trénovacej vzorke (obsahuje triedy, ich pravdepodobnosti aj hodnoty diskrimin
# pred <- predict(fit)
# (urpbiť: vykresliť pravdepodobnosti pre jednotlivé triedy a diskrimininanty ako alternati


# nové dáta a predpovede
xgrid <- expand.grid(Petal.Length = seq(1,7,0.01), Petal.Width = seq(0,2.5,0.01))
pred <- cbind.data.frame(
  xgrid,
  class = predict(fit, newdata = xgrid) |> getElement("class")
)
# zobrazenie oblastí klasifikácie do jednotlivých tried
pred |>
  ggplot() + aes(x = Petal.Width, y = Petal.Length) + geom_tile(aes(fill = class), alpha =
  geom_point(aes(color = Species), data = iris) +
  ggtitle("Hranice tried podľa LDA")
```




```
# externým balíkom
# klaR::partimat(Species ~ Petal.Width + Petal.Length, data = iris, method = "lda")
```

8.6.2 Kvadratická diskriminačná analýza

Príklad (kosatce)

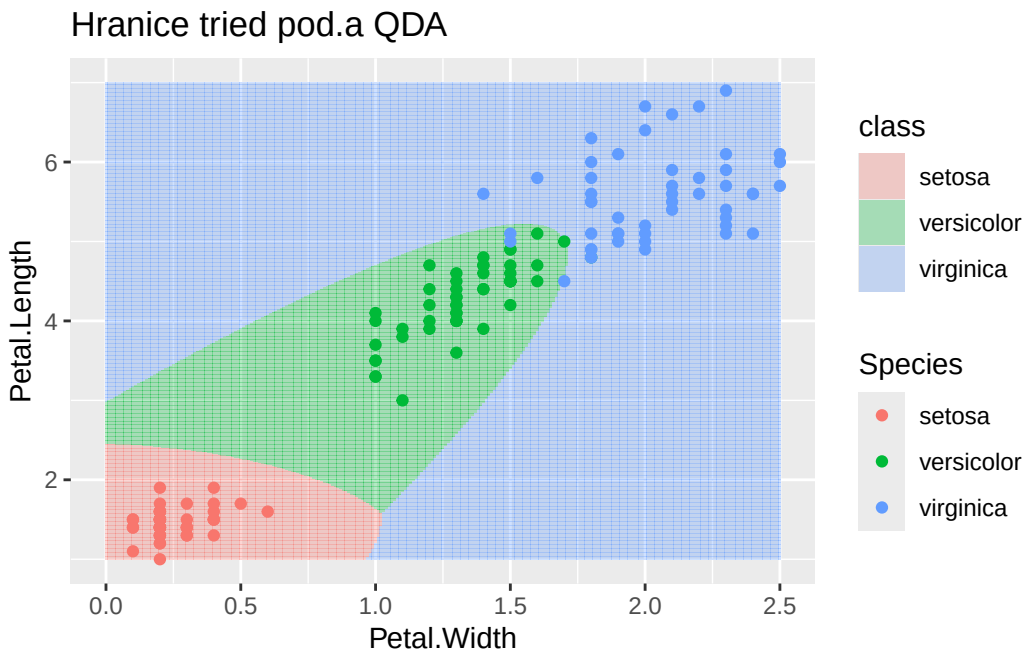
- Rovnaký príklad, ale tentokrát umožníme triedam mať svoju vlastnú kovariančnú maticu. Delenie priestoru sa zásadne zmení, pretože hranice už nie sú lineárne, ale kvadratické.

```
# kvadratická diskriminačná analýza
fit <- MASS::qda(Species ~ Petal.Width + Petal.Length, data = iris)
fit
```

```
Call:
qda(Species ~ Petal.Width + Petal.Length, data = iris)

Prior probabilities of groups:
    setosa versicolor  virginica
```

```
 0.3333333  0.3333333  0.3333333

Group means:
           Petal.Width Petal.Length
setosa           0.246        1.462
versicolor       1.326        4.260
virginica        2.026        5.552
```

```r
# nové dáta a predpovede
xgrid <- expand.grid(Petal.Length = seq(1,7,0.01),
                     Petal.Width = seq(0,2.5,0.01)
                     )
pred <- cbind.data.frame(
  xgrid,
  class = predict(fit, newdata = xgrid) |> getElement("class")
)
# zobrazenie oblastí klasifikácie do jednotlivých tried
pred |>
  ggplot() + aes(x = Petal.Width, y = Petal.Length) + geom_tile(aes(fill = class), alpha =
  geom_point(aes(color = Species), data = iris) +
  ggtitle("Hranice tried podľa QDA")
```



8.6.3 Naivný Bayesov klasifikátor

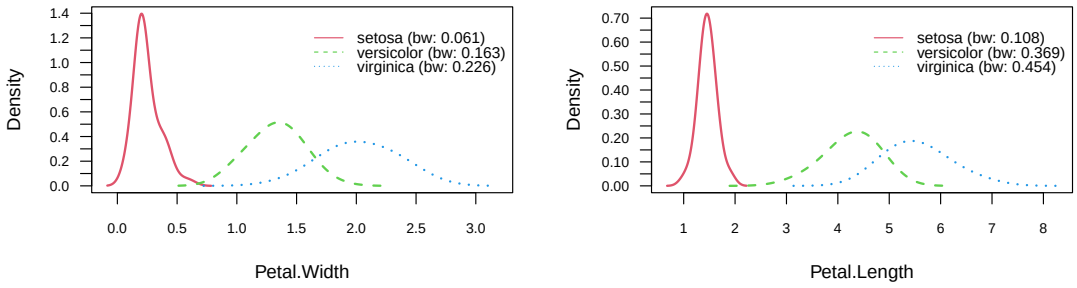
Príklad (kosatce)

- Tentokrát uvoľníme požiadavku normality, a na druhej strane zavedieme obmedzenie – nezávislosť prediktorov.
- Z tých známejších balíkov sú dostupné funkcie *klaR::NaiveBayes* a *e1071::naiveBayes*, avšak z hladiska funkcionality, rýchlosti, užívateľskej príjemnosti aj (minimálnej) externej závislosti je najlepší balík *naivebayes*.

```
# --- balíkom klaR
# fit <- klaR::NaiveBayes(Species ~ Petal.Width + Petal.Length, data = iris, usekernel = T
# plot(fit)

# --- balíkom e1071
# fit <- e1071::naiveBayes(Species ~ Petal.Width + Petal.Length, data = iris)
# výstup tvorí vektor apriórnych pravdepodobností a pod názvom "Conditional probabilities"

# --- balíkom naivebayes
# odhad modelu s predpokladom normálneho (parametrického) a vyhladeného (neparametrického)
fit <- list(
  gauss = naivebayes::naive_bayes(Species ~ Petal.Width + Petal.Length, data = iris),
  kernel = naivebayes::naive_bayes(Species ~ Petal.Width + Petal.Length, data = iris,
                                   usekernel = TRUE, adjust = 2)
)
```

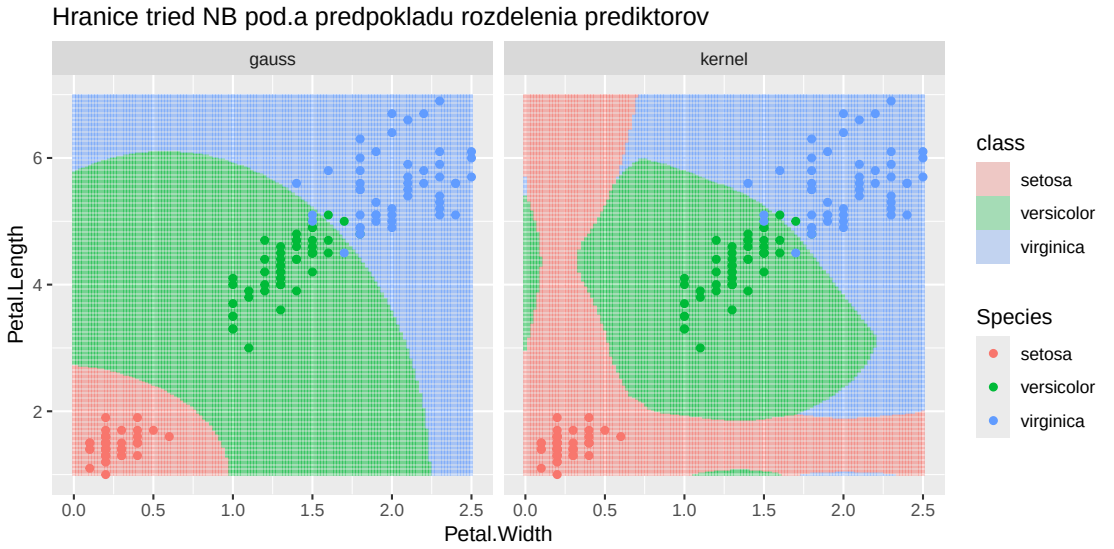
- Najprv zobrazíme (vyhladenú) hustotu v triedach.

```
# zobrazenie vyhladenej hustoty v jednotlivých triedach
fit$kernel |> plot(arg.num = list(lwd = 2))
```



- Následne zobrazíme delenie priestoru podľa jednotlivých modelov (predpokladov).

```
xgrid <- expand.grid(Petal.Length = seq(1,7,0.02),
                     Petal.Width = seq(0,2.5,0.02)
                     )

fit |>
  purrr::map_dfc(predict, newdata = xgrid, type = "class") |>
  cbind(xgrid) |>
  tidyr::pivot_longer(c(gauss, kernel), names_to = "distrib", values_to = "class") |>
   ggplot() + aes(x = Petal.Width, y = Petal.Length) + geom_tile(aes(fill = class), alpha =
  geom_point(aes(color = Species), data = iris) +
  facet_wrap(~distrib) +
  ggtitle("Hranice tried NB podľa predpokladu rozdelenia prediktorov")
```



8.7 Zdroje

- James, Witten, Hastie, Tibshirani (2021): An Introduction to Statistical Learning - with Applications in R. 2nd ed. Springer. https://www.statlearning.com/
- Sanchez, G., Marzban, E. (2020) All Models Are Wrong: Concepts of Statistical Learning. https://allmodelsarewrong.github.io
- Venables, W. N. and Ripley, B. D. (2002) Modern Applied Statistics with S. Fourth edition. Springer.

9 Presnosť klasifikácie

9.1 Celková presnosť klasifikačných modelov

- Presnosť predpovedných modelov sa často meria celkovou chybou predpovedí, teda ako nejaká *miera odchýlky* modelu $\hat{f}$ od skutočnosti f.
- V *regresnej* úlohe sa zvyčajne používa stredná kvadratická chyba, čiže MSE, menej často deviancia, MAE, MAPE, atď.
- V *klasifikačnej* úlohe s K triedami je možností tiež niekoľko:
  - celková klasifikačná chyba (*error rate*)

$$ER = \frac{1}{n}\sum_{i=1}^{n} I(\hat{y}_i \neq y_i)$$

  - vážená klasifikačná chyba (*mean per class error*)

$$MPCE = \frac{1}{K}\sum_{k=1}^{K}\left(\frac{1}{n_i}\sum_{i=1}^{n} I(\hat{y}_i \neq y_i)I(y_i = k)\right)$$

  - stredná kvadratická chyba (pomocou doplnkovej pravdepodobnosti)

$$MSE = \frac{1}{n}\sum_{i=1}^{n}\left(1 - \hat{Pr}(\hat{y}_i = y_i)\right)^2$$

  - deviancia (cross-entropy) obsahuje logaritmus pravdepodobnosti násobený triedou

$$D = -\sum_{i=1}^{n}\sum_{k=1}^{K} I(y_i = k)\log \hat{Pr}(\hat{y}_i = y_i)$$

    čiže obzvlášť trestá za nízky odhad pravdepodobnosti skutočnej triedy.

- Vo všeobecnosti sa delia podľa typu predpovedí, teda či sa počítajú
  - z predpovedí samotných hodnôt odozvy (**hard** predictions),
  - alebo ich pravdepdobnosti (**soft** predictions).

9.2 Klasifikačná matica a celková chyba

- Namiesto jediného súhrnného čísla môže byť užitočnejšie zobraziť maticu počtu prípadov správnej a nesprávnej klasifikácie, tzv. *confussion matrix* **C**

		y		
		1	**2**	**3**
$\hat{y}$	**1**	c_{11}	c_{12}	c_{13}
	2	c_{21}	c_{22}	c_{23}
	3	c_{31}	c_{32}	c_{33}

pričom prvky $c_{jk} = \sum_{i=1}^{n} I(\hat{y}_i = j)I(y_i = k)$ predstavujú početnosti v jednotlivých situáciách. Prirodzene platí $c_{jk} \geq 0$ a tiež $\sum_{j,k} c_{jk} = n$.

- Dokonalý klasifikátor má nenulové prvky iba na hlavnej diagonále.
- Celková klasifikačná chyba sa vypočíta ako podiel súčtu mimo-diagonálnych a všetkých prvkov

$$ER = 1 - \frac{\sum_k c_{kk}}{n} = \frac{\sum_{j \neq k} c_{jk}}{n}$$

- Člen $\frac{\sum_k c_{kk}}{n}$ sa označuje ako celková presnosť klasifikáce (*accuracy*).
- Ak je $ER < 0.5$, model je úspešnejší ako náhodné hádanie.
- Zdalo by sa, že najlepší model je ten, ktorý má najvyššiu celkovú presnosť. To však neplatí, ak
  - sú početnosti tried *rôzne*. Vtedy sa môže stať, že konštantný model (predpovedá iba najpočetnejšiu triedu) je presnejší ako nejaký sofistikovanejší model.
  - triedy nemajú rovnakú *váhu*. Napr. banka viac stratí na jednom neplatičovi než získa na úrokoch od platiaceho klienta.

- Problém s nevyváženým počtom tried rieši *Cohenovo kappa*, $\kappa \in [-1, 1]$, ktoré sa v aplikáciách používa hlavne ako miera zhody dvoch hodnotiteľov. Je definovaná vzťahom

$$\kappa = \frac{p_o - p_e}{1 - p_e}$$

  kde

  - $p_o = \frac{1}{n} \sum_k c_{kk}$ je relatívna početnosť prípadov, kedy *došlo* ku zhode ("o" ako "observed"),
  - $p_e = \frac{1}{n^2} \sum_k c_{k\cdot} c_{\cdot k}$ je relatívna početnosť hypotetických prípadov, kedy *by došlo* ku zhode, ak by bol klasifikátor úplne náhodný ("e" ako "expected"). Člen $c_{k\cdot} = \sum_j c_{kj}$ je riadkový súčet, analogicky je definovaný aj stĺpcový súčet $c_{\cdot k}$.

Hodnota $\kappa = 0$ predstavuje nezávislosť (náhodné tipovanie), zatiaľčo $\kappa = 1$ dokonalú zhodu.
- Pri druhom probléme (rozdielnej dôležitosti tried) treba chyby špecifikovať podľa záujmu/zamerania.

9.3 Špecifické chyby

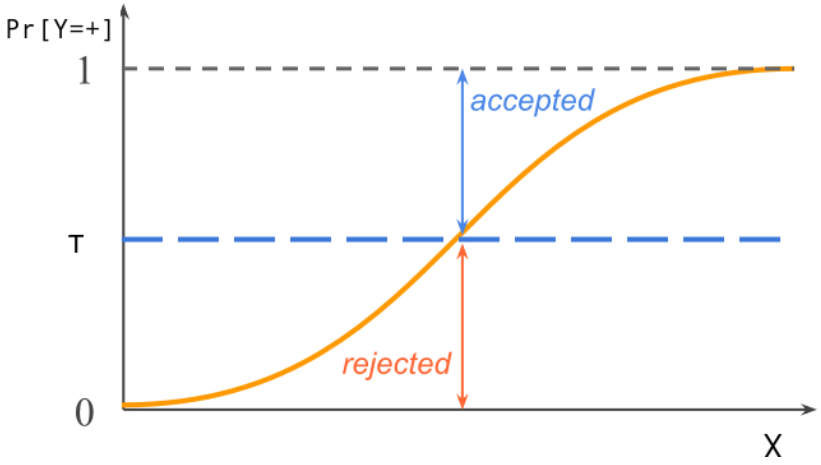
- Z klasifikačnej matice sa dá vyvodiť niekoľko špecifických mier presnosti, jednoduchšie však bude ukázať ich na prípade binárnej odozvy, ktorá indikuje nastatie nejakej udalosti (+/-), napr. splatenie pôžičky klientom banky.

		y	
		(+)	(–)
$\hat{y}$	(+)	**T**rue **P**ositive	**F**alse **P**ositive
	(–)	**F**alse **N**egative	**T**rue **N**egative

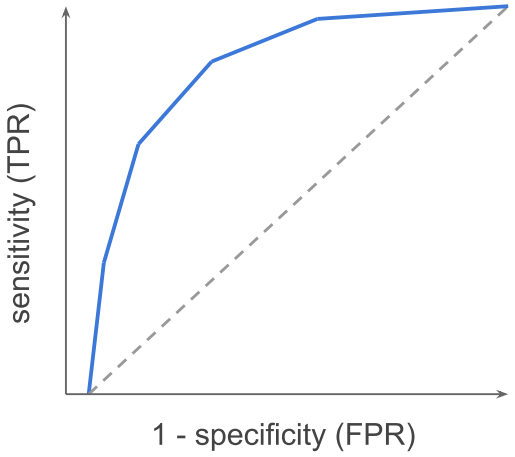
- Samozrejme celková presnosť sa potom vyjadrí vzťahom $accuracy = \frac{TP+TN}{n}$.
- Špecifické miery presnosti klasifikátora:
  - $precision = \frac{TP}{TP+FP}$ predstavuje podmienenú pravdepodobnosť $Pr[Y = \oplus|\hat{Y} = \oplus]$.
  - true positive rate (*sensitivity*, *recall*) $TPR = \frac{TP}{TP+FN}$ predstavuje podmienenú pravdepodobnosť $Pr[\hat{Y} = \oplus|Y = \oplus]$.
  - true negative rate (*specificity*) $TNR = \frac{TN}{TN+FP}$ predstavuje podmienenú pravdepodobnosť $Pr[\hat{Y} = \ominus|Y = \ominus]$.
  - false negative rate $FNR = 1 - TPR = \frac{FN}{TP+FN}$ je doplnkom ku TPR, podobne
  - false positive rate $FPR = 1 - TNR = \frac{FP}{TN+FP}$.
- Ideálne by sme chceli klasifikátor, ktorý zároveň minimalizuje FPR aj FNR, problém je však v tom, že sa to nedá – so zvyšovaním senzitivity sa znižuje špecificita a naopak. Je potrebný kompromis.
- Pozn.: Príkladom binárnej klasifikácie je aj testovanie štatistických hypotéz. Ak platnosť H_0 zodpovedá kladnej hodnote odozvy, potom *chyba I druhu* (α, zamietnutie platnej H_0) zodpovedá *FNR*, a *chyba II druhu* (β, prijatie neplatnej H_0) zodpovedá *FPR*. Aj tu platí nepriama úmernosť medzi oboma chybami. Keďže sa α zvykne zafixovať (najčastejšie na 5%), je výhodné použiť taký štatistický test, ktorý má čo najnižšiu chybu β, t.j. čo najväčšiu silu $(1 - \beta)$. (Napr. parametrické testy majú väčšiu silu, ak sú splnené predpoklady - zväčša o normalite údajov.)

9.4 ROC krivka

- Kompromis závisí na voľbe rozhodovacieho pravidla - konkrétne na voľbe hraničnej pravdepodobnosti τ, po ktorú je bod zaradený do jednej triedy a od ktorej je už zaradený do druhej triedy.

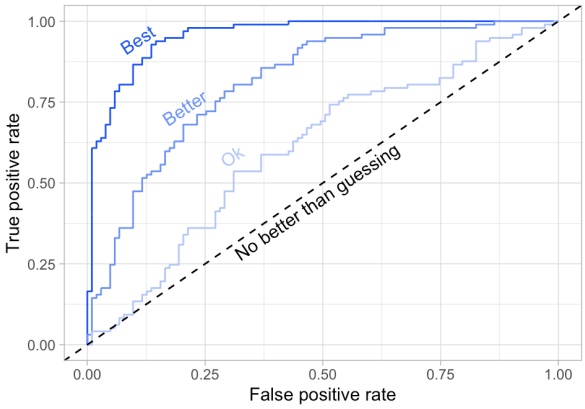


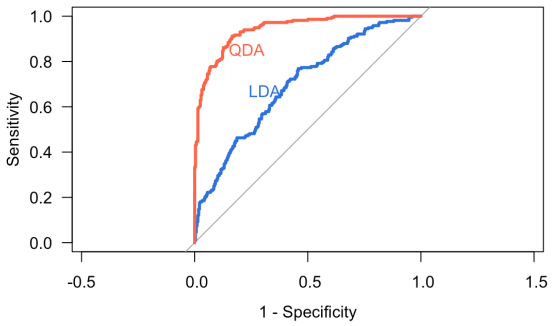
- V príklade s rozhodnutím, či poskytnúť alebo neposkytnúť pôžičku, voľba $\tau = 0.5$ z pohľadu banky nedáva veľký zmysel, pretože poskytnutie pôžičky so sebou prináša vyššie riziko než je zisk z úrokov. Banka teda pre ochranu svojich záujmov radšej zodvihne hranicu, napr. na $\tau = 0.7$.
- Samozrejme v inom scenári je lepšie zvoliť nižšie hodnoty prahovej hodnoty τ.
- S rozhodnutím, aké τ zvoliť, pomáha tzv. *ROC krivka* (Receiver Operating Characteristic, názov pochádza z prvej aplikácie pre obsluhu vojenského radarového prijímača). Znázorňuje kompromis medzi TPR a FPR.



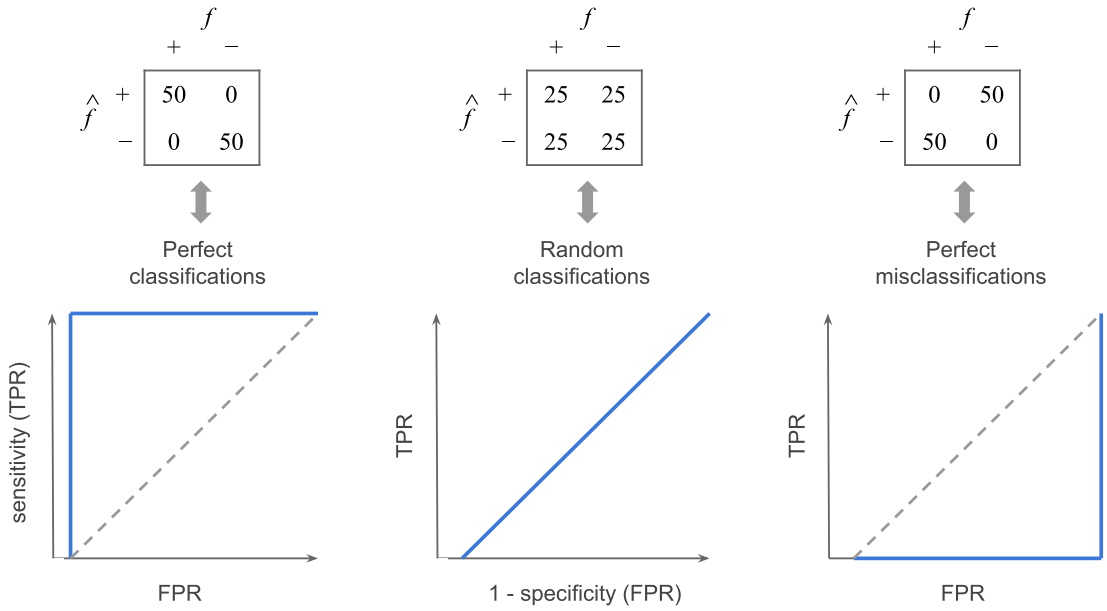
9.5 Plocha pod ROC krivkou

- Čím je model úspešnejší v súčasnej minimalizácii FNR a FPR, tým je ROC krivka vypuklejšia smerom ku rohu [0,1].
- Plocha pod krivkou (area under the curve, AUC) tak slúži ako miera úspešnosti klasifikačného modelu.





- Hodnota plochy
  - 0 zodpovedá dokonale neúspešnému modelu,
  - 1/2 hádaniu (náhodnému tipovaniu, hodu mincou),
  - 1 dokonalej klasifikácii.



- Ak má odozva viac hodnôt, klasifikačný model sa dá vyhodnotiť dvoma spôsobmi, a to výpočtom ROC/AUC pre jednu triedu voči
  - ostatným triedam (One-vs-Rest),
  - inej triede (One-vs-One).

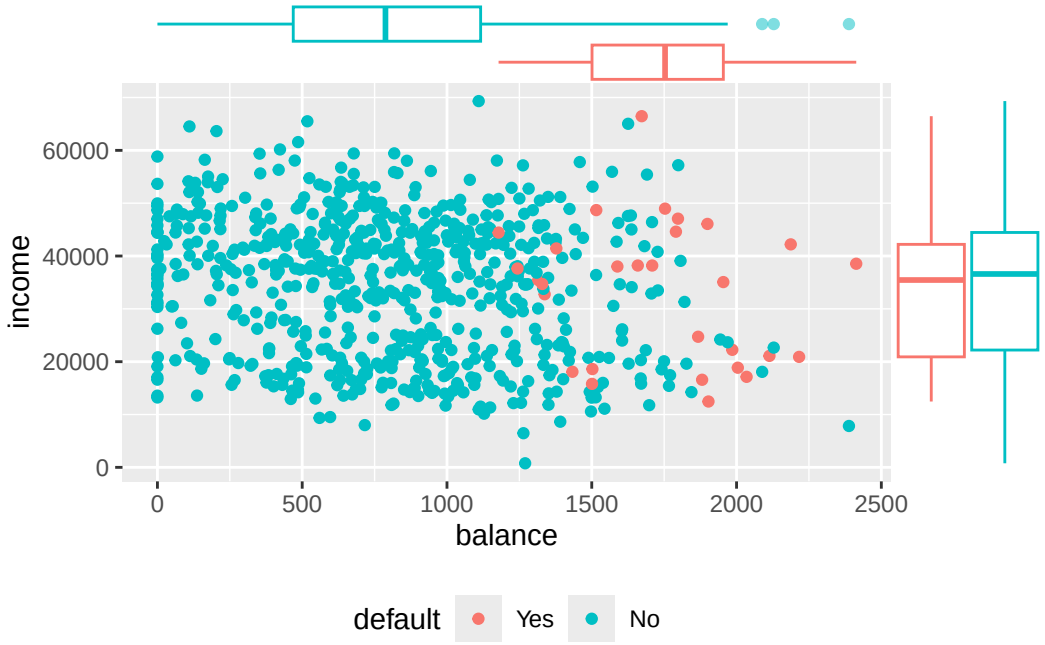
  Súhrnná AUC je potom už len priemerom z jednotlivých kombinácií.

9.6 Príklady

Príklad (kreditky)

- Zo (simulovaného) datasetu *ISLR2::Default* použijeme záznamy o 1000 klientoch banky na predpovedanie, či držiteľ kreditnej karty zlyhá v splácaní svojho dlhu (premešká lehotu splatnosti, angl. *default* on payment).
- Vysvetľujúce premenné sú
  - ročný príjem (*income*),
  - priemerný mesačný dlh na kreditke (*balance*).

```r
set.seed(3)
dat <- ISLR2::Default |>
  dplyr::slice_sample(n = 1000) |>
  split(sample(c("train","eval"), size = 1000, replace = TRUE, prob = c(.70, 0.30)))

p <- dat$train |>
  transform(default = relevel(default, ref = "Yes")) |>  # len kvôli farbám
  ggplot() + aes(x = balance, y = income, color = default) +
  geom_point() +
    theme(legend.position = "bottom")

p |> ggExtra::ggMarginal(type = "boxplot", groupColour = TRUE)
```




```r
fit <- list(
  LogisR = glm(default ~ balance + income, dat$train, family = binomial)
  )
fit$LogisR |> summary()
```

```
Call:
glm(formula = default ~ balance + income, family = binomial,
    data = dat$train)
```

```
Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept) -1.112e+01  1.468e+00  -7.576 3.56e-14 ***
balance      5.300e-03  7.400e-04   7.163 7.91e-13 ***
income       3.110e-05  1.703e-05   1.826   0.0678 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 240.85  on 692  degrees of freedom
Residual deviance: 132.59  on 690  degrees of freedom
AIC: 138.59

Number of Fisher Scoring iterations: 8
```

- Zdá sa, že ani model nenašiel veľmi užitočnú informáciu vo výške príjmu. Okrem toho je badať, že disperzia je modelom mierne nadhodnotená (under-dispersion). Použitie kvázi-binomického rozdelenia by vplyv *income* trochu zvýraznil (klesol by odhad štandardnej chyby), avšak na hladine významnosti 5% by jeho parameter stále nemusel byť odlíšiteľný od nuly.
- Pre prahovú hodnotu pravdepodobnosti $\tau = 0.5$ vypočítajme klasifikačnú maticu, hodnoty celkovej presnosti a bod na ROC krivke.

```r
prob <- data.frame(
  LogisR = fit$LogisR |> predict(newdata = dat$eval, type = "response")
  )
pred = list(
  LogisR = ifelse(prob$LogisR > 0.5, "Yes", "No") |> factor()
  )
# confusion matix
cm <- data.frame(predicted = pred$LogisR,
                 true = dat$eval$default) |>
  purrr::map_df(relevel, ref = "Yes") |>   # iba kvôli zoradeniu
  table()
cm
```

```
         true
predicted Yes  No
      Yes   6   2
      No    4 295
```

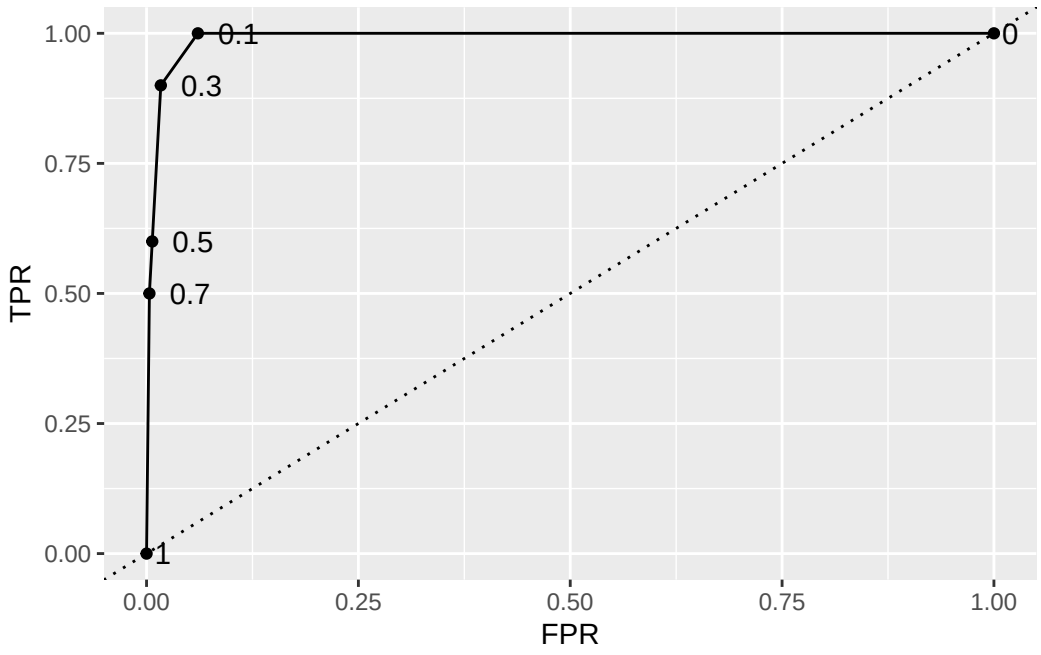
```r
# celková presnosť
accuracy_kappa <- function(cmatrix) {
  n <- sum(cmatrix)
  pO <- sum(diag(cmatrix)) / n
  pE <- sum(rowSums(cmatrix) * colSums(cmatrix)) / (n^2)
  c(
    accuracy = pO,
    kappa = (pO-pE)/(1-pE)
  )
}
# bod na ROC krivke pre
roc <- function(cmatrix) {
  c(
    TPR = cmatrix["Yes","Yes"]/sum(cmatrix[,"Yes"]),
    FPR = cmatrix["Yes","No"]/sum(cmatrix[,"No"])
    )
}
c(accuracy_kappa(cm),
  roc(cm)
  ) |>
  round(4) |> rbind("tau=0.5: " = _)
```

```
          accuracy  kappa TPR    FPR
tau=0.5:    0.9805 0.6567 0.6 0.0067
```

```r
#table(
#  predicted = pred$LogisR |> relevel("Yes"),
#  true = dat$eval$default |> relevel("Yes")
#  )
# caret::confusionMatrix(data = pred$LogisR, ref = dat$eval$default)
```

- Nakoniec zobrazíme ROC krivku pre niekoľko prahových hodnôt τ a vypočítame (približnú) plochu pod ňou.

```r
# ROC
# tau_grid <- exp(seq(log(0.01), log(1), length.out = 7))
tau_grid <- c(0.001, 0.1, 0.3, 0.5, 0.7, 1) |> round(2)
pred$LogisR <- tau_grid |>
  sapply(function(x) ifelse(prob$LogisR > x, "Yes", "No") ) |>
  (`colnames<-`)(tau_grid)

datROC <- apply(pred$LogisR, 2, function(x) table(
  factor(x, c("Yes", "No")),
  dat$eval$default),
  simplify = FALSE
  ) |>
  sapply(roc) |> t() |>
  as.data.frame() |>
  tibble::rownames_to_column("tau")
datROC |>
  ggplot() + aes(x=FPR, y=TPR) +
  geom_text(aes(label = tau), hjust=-0.5) +
  geom_point() +
  geom_line() +
  geom_abline(slope = 1, linetype = 3)
```



```r
with(datROC,
     sum( diff(FPR) * (TPR[-1] + head(TPR,-1)) / 2 )
     ) |>
  abs() |> round(3) |> setNames("AUC")
```

```
  AUC
0.991
```

```r
#ROCR::prediction(prob$LogisR, labels = dat$eval$default) |>
#  ROCR::performance(measure = "tpr", x.measure = "fpr") |>
#  plot("ROC pre model logistickej regresie")
```

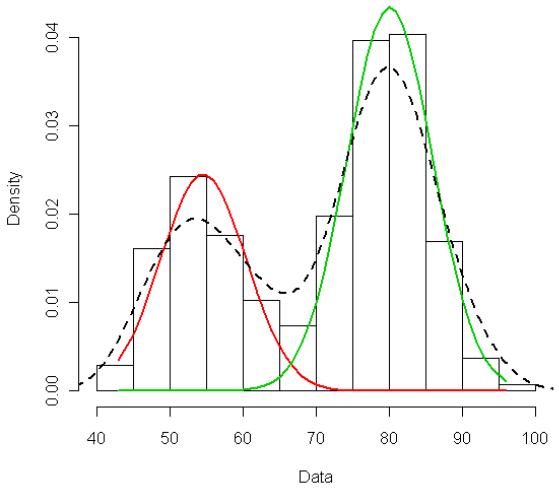
9.7 Zdroje

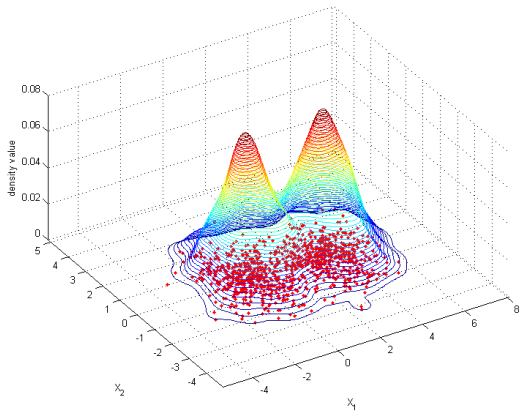
- Boehmke, B., Greenwell, B. (2019). Hands-on machine learning with R. Chapman and Hall/CRC. https://bradleyboehmke.github.io/HOML/process.html#model-eval
- James, Witten, Hastie, Tibshirani (2021): An Introduction to Statistical Learning - with Applications in R. 2nd ed. Springer. https://www.statlearning.com/
- Kuhn, M., Johnson, K. (2019). Feature engineering and selection: A practical approach for predictive models. CRC Press. https://bookdown.org/max/FES/measuring-performance.html#class-metrics
- Sanchez, G., Marzban, E. (2020) All Models Are Wrong: Concepts of Statistical Learning. https://allmodelsarewrong.github.io/classperformance.html#classification-error-measures

10 Združené rozdelenie pravdepodobnosti

10.1 Rekapitulácia

- V *regresnej úlohe* štatistického učenia sme modelovali rozdelenie pravdepodobnosti jedinej náhodnej premennej – odozvy. Rozdelenie vysvetľujúcich premenných – prediktorov – nás zväčša nezaujímalo.
- Ak je odozva *náhodný vektor* (multivariate regression), modelované rozdelenie už nie je jednorozmerné, ale združené (štandardne z normálneho rozdelenia).
- Vďaka zovšeobecnenému lineárnemu modelu môže mať odozva rozdelenie z exponenciálnej triedy rozdelení - spojitých aj diskrétnych.
- Tak môžeme v *klasifikačnej úlohe* modelovať rozdelenie binárnej odozvy (Bernouliho r.) pomocou rozdelenia logaritmu šance (normálne r.) priamo cez link funkciu ($p \to \mu$). To je princíp logistickej regresie.
- Naproti tomu generatívne metódy v klasifikačnej úlohe počítajú s *vysvetľujúcimi premennými ako náhodnými*, teda aj s definovaným združeným rozdelením.
- Neasistovaná verzia generatívnych klasifikačných metód je známa ako zhluková analýza založená na *zmesi* rozdelení.



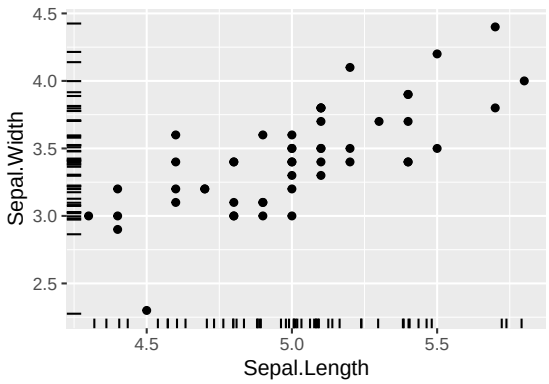


10.2 Cieľ a prostriedky

- *Náhodný vektor* je vektor náhodných premenných, napr. $\vec{X} = (X_1, X_2)$.
- Vo všeobecnosti, ak by sme poznali združené rozdelenie náhodného vektora, dokázali by sme pomocou podmieňovania (Bayesova veta) vyjadriť rozdelenie pravdepodobnosti jednej alebo viacerých jeho premenných.
- Vyjadriť alebo odhadnúť združené rozdelenie je *náročné* a táto náročnosť rastie so zvyšujúcim sa rozmerom náhodného vektora. Preto je často nutné zaviesť *zjednodušujúce predpoklady*, napr. o nezávislosti niektorých (alebo všetkých) premenných.
- Tento všeobecný prístup si ukážeme na regresnej úlohe, v ktorej všetky premenné budeme uvažovať ako spojité.
- Potrebujeme na to sadu teoretických i praktických nástrojov.
- Z tých teoretických je treba pochopiť pojem okrajového (marginálneho) rozdelenia, podmieňovanie, nezávislosť, miery a funkcie popisujúce rozdelenie (momenty, korelačné koeficienty, PDF, CDF), rozklad združeného rozdelenia.
- Tie praktické zahŕňajú aj nástroje popisnej štatistiky ako histogram, kobercový a bodový graf, ktoré odhaľujú charakter jedno- resp. dvojrozmerného rozdelenia (okrajové rozdelenie aj koreláciu).

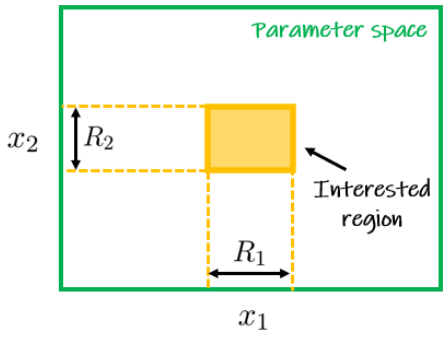
Ilustrácia (bodový a kobercový graf)

```
#mtcars |>
#  subset(select = c(disp, hp)) |>
#  ggplot() + aes(disp, hp) + geom_point()

iris |>
  subset(select = c(Sepal.Length, Sepal.Width), subset = Species == "setosa") |>
  ggplot() + aes(Sepal.Length, Sepal.Width) + geom_point() +
  geom_rug(position = "jitter")
```



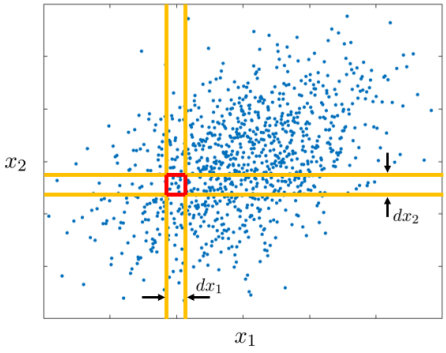
10.3 Hustota združeného rozdelenia

- Rovnako ako v jednorozmernom prípade, aj združené rozdelenie je charakterizované
  - napr. distribučnou funkciou (angl. *joint cumulative distribution function*, CDF) alebo
  - ešte názornejšie – hustotou (angl. *joint probability density function*, PDF).
- Zvyčajne nás zaujíma pravdepodobnosť, že sa náhodné premememenné zrealizujú do (t.j. že spozorujeme ich hodnoty v) určitej oblasti priestoru.
- Môžeme si to predstaviť ako združené javy (A & B), kedy jav A predstavuje zrealizovanie X_1 do oblasti R_1 a analogicky jav B nadobudnutie hodnoty premennou X_2 z oblasti R_2.
- V jednotlivých prípadoch by nám stačili hustoty $f_i(x_i)$ pre každú premennú X_i zvlášť, ale na vyjadrenie pravdepodobnosti *súčasného* nastatia oboch javov potrebujeme združenú hustotu $f(x_1, x_2)$.



- Ak $R_1 = [x_1, x_1 + dx_1]$ a $R_2 = [x_2, x_2 + dx_2]$ pre malé diferencie dx_1, dx_2, potom pravdepodobnosť nastatia združeného javu A & B je približne násobok hustoty a plochy oblasti $R_1 \times R_2$,

$$Pr(x_1 \leq X_1 \leq x_1 + dx_1,\ x_2 \leq X_2 \leq x_2 + dx_2) \approx f(x_1, x_2) dx_1 dx_2.$$



- Presnejšie, pre ľubovoľnú oblasť $[a_1, b_1] \times [a_2, b_2]$ to bude

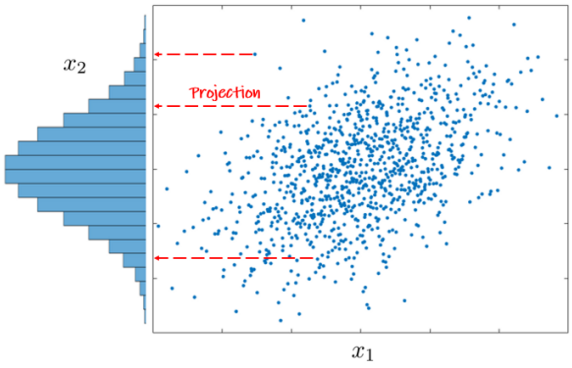
$$Pr(a_1 \leq X_1 \leq b_1, a_2 \leq X_2 \leq b_2) = \int_{a_2}^{b_2} \int_{a_1}^{b_1} f(x_1, x_2) dx_1 dx_2.$$

- Integrál hustoty cez celý priestor sa musí rovnať pravdepodobnosti istej udalosti,

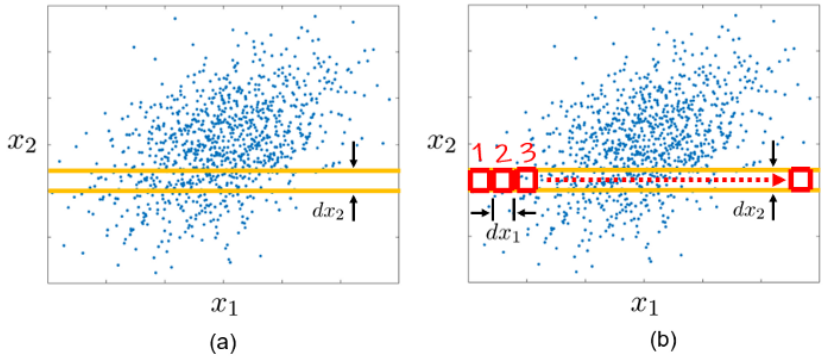
$$\iint_{\mathcal{R}^2} f(x_1, x_2) dx_1 dx_2 = 1.$$

10.4 Marginálne rozdelenie

- Rozdelenie pravdepodobnosti iba určitej podmnožiny náhodného vektora sa nazýva okrajové, marginálne rozdelenie. Dostaneme ho napr. *marginalizáciou* celého združeného rozdelenia.
- Majme bodový graf a chceme vizualizovať hustotu rozdelenia iba jednej z premenných, napr. X_2.
- Vtedy stačí jednoducho zobraziť každý bod na vertikálnu os a vykresliť zodpovedajúci histogram.
- Po jeho normalizácii, teda preškálovaní plochy stĺpcov, aby sa v súčte rovnala jednej, dostávame aproximáciu hustoty $f_2(x_2)$.
- Tým prakticky zanedbáme rozdelenie X_1 a zobrazíme histogram na základe hodnôt datasetu v stĺpci premennej X_2.



- Z druhého uhla pohľadu, marginálne rozdelenie X_2 vyjadruje pravdepodobnosť, že hodnota X_2 padne do nekonečne malého intervalu $[x_2, x_2 + dx_2]$, čo v kontexte bodového grafu je to isté, ako pravdepodobnosť, že bod (x_1, x_2) sa ocitne v oranžovom páse.



- Ak si pás rozdelíme na plôšky $dx_1 dx_2$, potom ich váženým súčtom dostaneme

$$Pr(x_2 \leq X_2 \leq x_2 + dx_2) \approx \sum_i f(x_1^{(i)}, x_2) dx_2 dx_1$$

  a porovnaním so vzťahom pre jednorozmernú hustotu

$$Pr(x_2 \leq X_2 \leq x_2 + dx_2) \approx f_2(x_2) dx_2$$

  vznikne

$$f_2(x_2) \approx \sum_i f(x_1^{(i)}, x_2) dx_1$$

$$f_2(x_2) = \int_{-\infty}^{\infty} f(x_1, x_2) dx_1$$

- Analogicky to platí aj pre X_1,

$$f_1(x_1) = \int_{-\infty}^{\infty} f(x_1, x_2) dx_2$$

- Rozšírenie pre p-rozmerný náhodný vektor je jednoduché,

$$f_1(x_1) = \int_{-\infty}^{\infty} \dots \int_{-\infty}^{\infty} f(x_1, \dots, x_p) dx_2 \dots dx_p$$

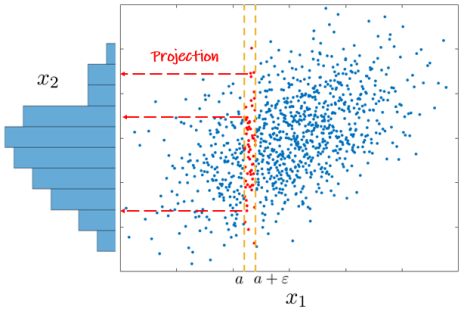
- Vyjadriť sa dá aj *združené marginálne* rozdelenie,

$$f_{12}(x_1, x_2) = \int_{-\infty}^{\infty} \dots \int_{-\infty}^{\infty} f(x_1, \dots, x_p) dx_3 \dots dx_p$$

10.5 Podmienené rozdelenie

- Marginálne rozdelenie nám hovorí o tom, ako sa správa X_2, ale *bez ohľadu* na rozdelenie X_1.
- Niekedy sa v praxi stáva, že odpozorujeme hodnotu X_1 a chceme vedieť, či a ako táto znalosť *ovplyvní* predpokladané rozdelenie X_2. Hľadáme teda rozdelenie X_2 *podmienené* tým, že X_1 nadobudne konkrétnu hodnotu.
- Podmienené rozdelenie tvorí kľúčový pojem v štatistickej dedukcii, kde hlavným cieľom je usúdiť, aký je parameter rozdelenia na základe pozorovaných údajov (t.j. nimi podmienený) .
- V príklade s bodovým grafom sa dá podmienená hustota $f_{2|1}(x_2|X_1 = a)$ odhadnúť nasledujúcim spôsobom:

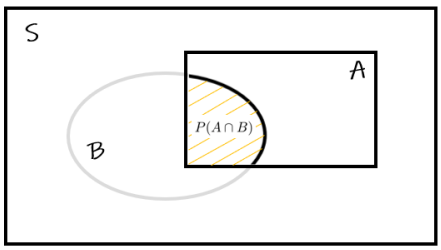
  1. Zameriame sa iba na body v úzkom páse pri hodnote $X_1 = a$.
  2. Tieto body zobrazíme na os X_2 a vypočítame normalizovaný histogram.



- Vynásobením malým prírastkom dostávame podmienenú pravdepodobnosť,

$$f_{2|1}(x_2|X_1 = a)dx_2 = Pr(x_2 \leq X_2 \leq x_2 + dx_2 | a \leq X_1 \leq a + \varepsilon)$$

- Pravdepodobnosť nastatia javu B ak nastal jav A sa vypočíta ako podiel nejakej miery (napr. počet pozorovaní) v priestore javu A, teda $Pr(B|A) = \frac{n(A \cap B)}{n(A)}$



- Normovaním mierou celého priestoru získame podiel pravdepodobností

$$Pr(B|A) = \frac{n(A \cap B)/n(S)}{n(A)/n(S)} = \frac{Pr(A \cap B)}{Pr(A)}$$

Prepojením na predošlý kontext, $B = (x_2 \leq X_2 \leq x_2 + dx_2)$ a $A = (a \leq X_1 \leq a + \varepsilon)$, je možné pravdepodobnosti prepísať v reči hustoty na

$$f_{2|1}(x_2|X_1 = a)dx_2 = \frac{f(a, x_2)dx_2}{f_1(a)}$$

Vo všeobecnosti a po úprave je podmienená hustota premennej X_2 (ak je známa hodnota X_1) daná vzťahom

$$f_{2|1}(x_2|x_1) = \frac{f(x_1, x_2)}{f_1(x_1)}$$

- Pretože podmienená hustota je stále hustota pravdepodobnosti, integrovaním predošlého vzťahu sa dá ukázať, že platí $\int_{\mathcal{R}} f(x_2|x_1)dx_2 = 1$ a rovnako môžeme písať

$$Pr(a \leq X_2 \leq b|X_1 = x_1) = \int_a^b f_{2|1}(x_2|x_1)dx_2$$

- Integrovaním vzťahu pre združenú hustotu podľa x_1

$$f(x_1, x_2) = f_{2|1}(x_2|x_1)f_1(x_1)$$

dostávame alternatívne vyjadrenie marginálnej hustoty

$$f_2(x_2) = \int_{\mathcal{R}} f_{2|1}(x_2|x_1)f_1(x_1)dx_1$$

ktoré je známe ako *veta o úplnej pravdepodobnosti*.

- Analogicky podmienená pravdepodobnosť pre X_1 je daná vzťahom

$$f_{1|2}(x_1|x_2) = \frac{f(x_1, x_2)}{f_2(x_2)}$$

Rozšírenie na p-rozmerný náhodný vektor je priamočiare,

$$f_{1|2\ldots p}(x_1|x_2, \ldots, x_p) = \frac{f(x_1, \ldots, x_p)}{f_{2\ldots p}(x_2, \ldots, x_p)}$$

Funkcia v menovateli je hustota *združeného marginálneho* rozdelenia premenných $X_2, \ldots, X_p$.

- Podobne vieme vyjadriť rozdelenie (X_1, X_2) podmienené nastatím $X_3 = x_3, \ldots, X_p = x_p$,

$$f_{12|3\ldots p}(x_1, x_2|x_3, \ldots, x_p) = \frac{f(x_1, \ldots, x_p)}{f_{3\ldots p}(x_3, \ldots, x_p)}$$

- Podmienená pravdepodobnosť sa využíva aj na *faktorizáciu* (rozloženie na činitele) združenej hustoty, napr.

$$f(x_1, x_2, x_3) = f_{1|23}(x_1|x_2, x_3)f_{2|3}(x_2|x_3)f_3(x_3)$$

ktorú by inak bolo ťažké počítať priamo.

10.6 Bayesova veta

- Bayesova veta je jednou z najdôležitejších v teórii pravdepodobnosti, tvorí základ bayesovskej štatistickej dedukcie.
- Ak vyjdeme z oboch možných faktorizácií dvojrozmernej združenej hustoty (nateraz bez značenia hustoty indexami),

$$f(x_1, x_2) = f(x_2|x_1)f(x_1)$$
$$f(x_1, x_2) = f(x_1|x_2)f(x_2)$$

  ich kombináciou dostaneme Bayesov vzorec

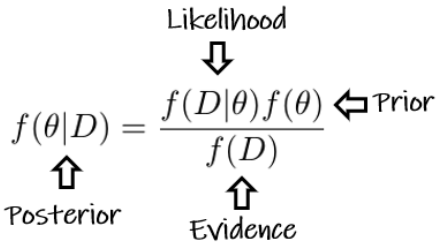
$$f(x_2|x_1) = \frac{f(x_1|x_2)f(x_2)}{f(x_1)}$$

10.7 Bayesovská štatistika

- Príklady otázok, ktoré možno riešiť bayesovskou štatistikou:
  - Ak hodíme mincu 10 krát a pozorujeme 7-krát hlavu a 3-krát znak, ako veľmi sa môžme spoľahnúť, že minca nie je falošná?
  - Ak je ráno zamračené, aká je pravdepodobnosť že popoludní spŕchne?
  - Majúc dataset dvojíc (x_i, y_i), aké parametre sú najlepšie na preloženie priamky?
  - Ako veľmi si môžeme byť istí, že naše odhadnuté parametre sú správne?
- Bayesovská štatistika ponúka jeden spôsob štatistickej dedukcie, čiže ako odhaliť fundamentálny model z pozorovaných dát.
- Bayesovské usudzovanie
  1. zvyčajne začína predpokladmi o fundamentálnom pravdepodobnostnom modeli. Vopred môžeme mať napríklad informáciu, že model má určitú formu a jeho parametre sa nachádzajú v určitých intervaloch. Tieto predpoklady tvoria východzie (*prior*) znalosti.
  2. Následne zbierame dáta, pričom dúfame, že nám tieto pozorované údaje odhalia viac informácií o modeli.
  3. Nakoniec použijeme Bayesovu vetu na získanie týchto informácií z údajov a aktualizovanie východzích predpokladov. Tým dostávame posteriórny odhad (*posterior*).

  Dá sa chápať ako proces *prerozdelovania dôveryhodnosti medzi možnosťami*: spočiatku máme iba predstavu o možných scenároch, každý má pridelenú svoju pravdepodobnosť. No potom tým scenárom, ktoré sú v zhode s pozorovaniami, sa pravdepodobnosť navyšuje, na úkor tých, ktoré dátam nezodpovedajú až tak dobre.

- Bayesovská štatistika je elegantná, pretože tento abstraktný proces aktualizovania vedomostí opisuje merateľným spôsobom. Pozrime sa na jednotlivé komponenty Bayesovho vzorca, ktorý najprv prepíšeme do symbolického tvaru



$$f(\theta|D) = \frac{f(D|\theta)f(\theta)}{f(D)}$$

- Člen θ predstavuje parameter fundamentálneho pravdepodobnostného modelu, ktorý je predmetom odhadu, a D označujú dáta. Potom
  - $f(\theta)$ je apriorné rozdelenie parametra θ, formuluje naše predstavy o skutočnom parametri pred tým, než zrealizujeme pozorovania,
  - $f(D|\theta)$ je funkcia vierohodnosti, poskytuje pravdepodobnosť toho, že by sme pozorovali rovnaké dáta ako D, ak by platil predpoklad o θ,
  - $f(\theta|D)$ je aposteriórne rozdelenie θ, teda aktualizácia naších predstáv urobená na základe reálnych pozorovaní,
  - $f(D)$ predstavuje celkovú pravdepodobnosť pozorovania D vypočítanú ako vážený priemer vierohodnostnej funkcie naprieč všetkými hodnotami θ, čiže $f(D) = \int f(D|\theta)f(\theta)d\theta$. Nazýva sa aj marginálnou vierohodnosťou.
- Menovateľ $f(D)$ nezávisí od θ. Často je to jednoducho normalizačná konštanta, ktorá zabezpečuje, že $f(\theta|D)$ je riadna hustota pravdepodobnosti.
- V praxi sú vierohodnostná funkcia a prior často natoľko komplikované, že sa posterior neoplatí riešiť analyticky. Na uľahčenie výpočtu boli vyvinuté a používajú sa metódy ako Laplaceova aproximácia, Markov Chain Monte Carlo (MCMC) a variational Bayes.
- (tu by sa zišiel príklad)

Literatúra na hlbšie štúdium

- McElreath, R. (2020). Statistical rethinking: A Bayesian course with examples in R and Stan (Second Edition). CRC Press. https://xcelab.net/rm/statistical-rethinking/
- Kurz, S. (2021). Statistical rethinking with brms, ggplot2, and the tidyverse: Second edition https://bookdown.org/content/4857/

10.8 Nezávislosť

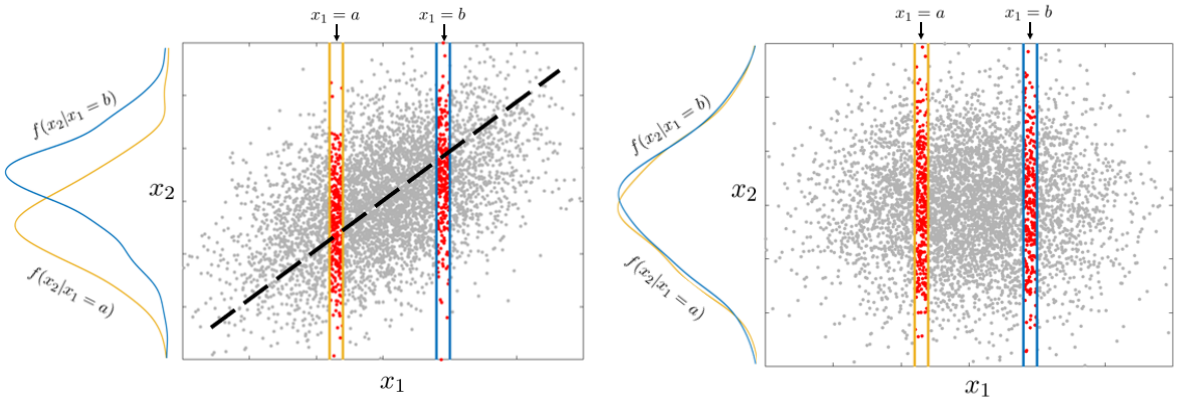
- Ak dve náhodné premenné spolu nesúvisia (nie sú vo vzájomnom vzťahu), hovoríme, že sú *štatisticky nezávislé*. Potom

$$f_{1|2}(x_1|x_2) = f_1(x_1), \qquad f_{2|1}(x_2|x_1) = f_2(x_2)$$

  a dosadením do združenej hustoty

$$f(x_1, x_2) = f_1(x_1) f_2(x_2)$$

- V praxi sa nezávislosť najjednoduchšie posúdi vizuálne – z bodového grafu.



10.9 Podmienená nezávislosť

- Podmienená nezávislosť znamená takú situáciu, keď dve premenné X_1 a X_2 sú nezávislé, ak máme informáciu o tretej premennej X_3 (inak sa môžu javiť ako závislé). Potom pre podmienenú hustotu platí vzťah

$$f_{12|3}(x_1, x_2|x_3) = f_{1|3}(x_1|x_3) f_{2|3}(x_2|x_3)$$

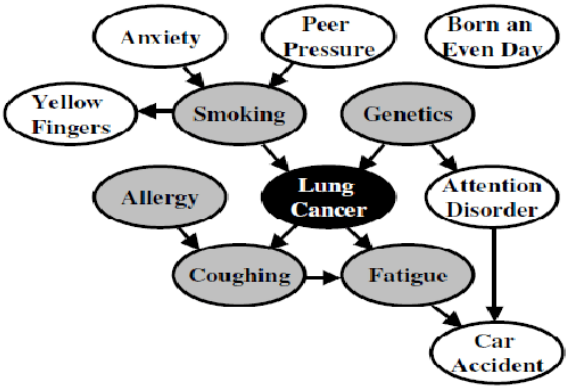
- Názorný príklad: Vo vrecku sú dve mince, jedna obyčajná (s hlavou aj znakom), druhá falošná s dvoma znakmi. Náhodne vyberiem jednu mincu a 2-krát ju hodím.
  - Jav A: Prvý hod ukáže znak.
  - Jav B: Druhý hod ukáže znak.
  - Jav C: Vybraná minca je obyčajná (nie je falošná).

Javy A a B nie sú nezávislé, výskyt A ovplyvní pravdepodobnosť výskytu B, tzn. $Pr(B|A) \neq Pr(B)$, pretože ak pozorujeme jav A je viac pravdepodobné, že bola použitá falošná minca, čo zvyšuje pravdepodobnosť výskytu B.
Ak však *vieme, že nastal jav C*, potom javy A a B sú nezávislé, lebo hádžeme obyčajnou mincou.

- Znalosť podmienenej nezávislosti je užitočná, pretože zjednodušuje faktorizáciu združenej PDF.

$$\begin{aligned} f(x_1, x_2, x_3) &= f_{1|23}(x_1|x_2, x_3) f_{2|3}(x_2|x_3) f_3(x_3) \\ &\overset{X_1 \perp X_2 | X_3}{=} f_{1|3}(x_1|x_3) f_{2|3}(x_2|x_3) f_3(x_3) \end{aligned}$$

Toto zjednodušenie je základom Bayesovských sietí, čo sú pravdepodobnostné modely využívajúce grafickú reprezentáciu na vyjadrenie podmienenej závislosti (ktorá v mnohorozmernom prípade môže byť značne komplikovaná).



- Vo všeobecnosti, náhodné premenné $X_1, \ldots, X_p$ sú štatisticky nezávislé, ak platí

$$f(x_1, \ldots, x_p) = f_1(x_1) \cdot f_2(x_2) \cdot \ldots \cdot f_p(x_p).$$

10.10 Stredná hodnota a rozptyl

- Rovnako ako v jednorozmernom prípade, aj na popis združeného rozdelenia sa používa stredná hodnota a rozptyl.
- Stredná hodnota sa stane vektorom, $E[\vec{X}] = (E[X_1], \ldots, E[X_p])$. Jednotlivo

$$E[X_1] = \int_{\mathcal{X}} x_1 f_1(x_1) dx_1$$

kde $f_1(x_1)$ je marginálna hustota, za ktorú ak dosadíme skoršie vyjadrenie $f_1(x_1) = \int_{\mathcal{R}} f(x_1, f_2)dx_2$, dostaneme

$$E[X_1] = \iint_{\mathcal{R}^2} x_1 f(x_1, x_2)dx_1 dx_2$$

- Podobne rozptyl, ktorý vyjadruje mieru variability okolo strednej hodnoty,

$$\begin{aligned} var[X_1] = E\left[(X_1 - E[X_1])^2\right] &= \int_{\mathcal{R}} (x_1 - E[X_1])^2 f_1(x_1)dx_1 \\ &= \iint_{\mathcal{R}^2} (x_1 - E[X_1])^2 f(x_1, x_2)dx_1 dx_2 \\ = E[X_1^2] - E[X_1]^2, & \end{aligned}$$

kde $E[X_1^2] = \int_{\mathcal{R}} x_1^2 f_1(x_1)dx_1$.

10.11 Kovariancia

- Pri viac než jednej premennej je okrem strednej hodnoty a variancie na popis rozdelenia treba aj mieru závislosti. Jednou takou je kovariancia.
- Kovariancia vyjadruje, ako sa dve premenné menia spolu,

$$\begin{aligned} cov[X_1, X_2] &= E[(X_1 - E[X_1])(X_2 - E[X_2])] \\ &= \iint_{\mathcal{R}^2} (x_1 - E[X_1])(x_2 - E[X_2]) f(x_1, x_2)dx_1 dx_2 \end{aligned}$$

- Jednoduchší je výpočet po úprave

$$cov[X_1, X_2] = E[X_1 X_2] - E[X_1]E[X_2].$$

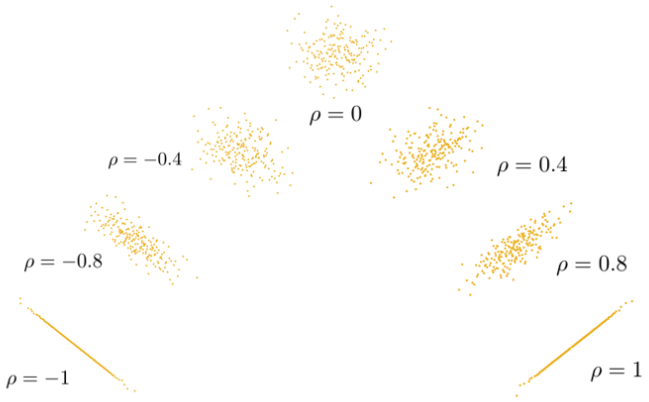
pričom $E[X_1 X_2] = \iint_{\mathcal{R}^2} x_1 x_2 f(x_1, x_2)dx_1 dx_2$.
- Zopakujme si aj vlastnosti kovariancie:
  - $cov[X_1, X_1] = var[X_1]$
  - $cov[X_1, X_2] = cov[X_2, X_1]$
  - $cov[a \cdot X_1, X_2] = a \cdot cov[X_1, X_2]$
  - $cov[X_1 + c, X_2] = cov[X_1, X_2]$
- Kovariancia nezávislých premenných sa rovná nule.

10.12 Korelačný koeficient

- Nevýhodou kovariancie je to, že jej veľkosť nám (sama o sebe) neprezrádza silu vzájomného vzťahu medzi premennými. Pretože *závisí od mierky*.
- Toto je napravené normalizáciou kovariancie v Pearsonovom korelačnom koeficiente

$$\rho(X_1, X_2) = \frac{cov[X_1, X_2]}{\sqrt{var[X_1]var[X_2]}}$$

- Korelačný koeficient (a kovariancia) vyjadruje stupeň iba *lineárnej* závislosti, preto nulová hodnota *neznamená* automaticky *nezávislosť*. Ak sú však dve premenné nezávislé, potom korelácia/kovariancia je vždy nulová.



10.13 Distribučná funkcia

- Okrem hustoty sa rozdelenie pravdepodobnosti dá reprezentovať aj kumulatívnou distribučnou funkciou (CDF), ktorá je vo všeobecnosti definovaná ako pravdepodobnosť *súčasného* neprekročenia hodnôt $\vec{x} = (x_1, \ldots, x_p)$,

$$F(\vec{x}) \equiv Pr(X_1 \leq x_1 \wedge X_2 \leq x_2 \wedge \ldots \wedge X_p \leq x_p)$$

  a voči hustote platia nasledujúce prevodové vzťahy:

$$F(\vec{x}) = \int_{-\infty}^{x_1} \cdots \int_{-\infty}^{x_p} f(y_1, \ldots, y_p) dy_1 \ldots dy_p$$

$$f(\vec{x}) = \left. \frac{\partial F(y_1, \ldots, y_p)}{\partial y_1 \ldots \partial y_p} \right|_{\vec{y}=\vec{x}}$$

- Podmienená distribučná funkcia sa vypočíta analogicky, ako určitý integrál podmienenej hustoty, napr.

$$\begin{aligned} F_{12|3\dots p}(x_1, x_2|x_3, \dots, x_p) &= \int_{-\infty}^{x_1}\int_{-\infty}^{x_2} f_{12|3\dots p}(y_1, y_2|x_3, \dots, x_p)dy_1dy_2 \\ &= \int_{-\infty}^{x_1}\int_{-\infty}^{x_2} \frac{f(y_1, y_2, x_3 \dots, x_p)}{f_{3\dots p}(x_3, \dots, x_p)}dy_1dy_2 \\ &= \frac{\int_{-\infty}^{x_1}\int_{-\infty}^{x_2} f(y_1, y_2, x_3, \dots, x_p)dy_1dy_2}{\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(y_1, y_2, x_3 \dots, x_p)dy_1dy_2} \\ &= \frac{\frac{\partial F(y_1, y_2, x_3 \dots, x_p)}{\partial y_1 \partial y_2}\Big|_{y_1=x_1, y_2=x_2}}{\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(y_1, y_2, x_3 \dots, x_p)dy_1dy_2} \end{aligned}$$

- Podobnou úvahou dostaneme aj marginálu distribučnú funkciu

$$F_{12}(x_1, x_2) = \int_{-\infty}^{x_1}\int_{-\infty}^{x_2}\int_{-\infty}^{\infty}\cdots\int_{-\infty}^{\infty} f(y_1, \dots, y_p)dy_1 \dots dy_p$$

- Inverzná funkcia ku jednorozmernej distribučnej funkcii F_X premennej X sa nazýva *kvantilová funkcia* Q. Čiže ak platí, že $F_X(q) \equiv Pr(X \leq q) = p$, potom platí aj

$$Q(p) \equiv F_X^{-1}(p) = q, \qquad \forall p \in [0, 1]$$

  Hodnota q sa nazýva kvantil premennej X, niekedy sa značí aj s dolným indexom, $q_p = Q(p)$, najznámejšie sú medián $q_{0.5}$, kvartily $q_{0.25}$ (dolný) a $q_{0.75}$ (horný), decily $q_{0.1}, q_{0.2}, \cdots, q_{0.9}$ a percentily (kde $p = 0.01, 0.02, \dots, 0.99$).

10.14 Funkcia prežitia

- Duálna funkcia ku CDF v zmysle doplnkovej pravdepodobnosti sa nazýva tzv. funkcia prežitia S (*survival function*), ktorá predstavuje pravdepodobnosť *(súčasného) prekročenia* určitých kvantilov náhodnými premennými, napr. v prvých troch rozmeroch sa vypočíta podľa vzťahov

$$\begin{aligned} S_1(x_1) &\equiv Pr(X_1 > x_1) = 1 - F_1(x_1) \\ S_{12}(x_1, x_2) &\equiv Pr(X_1 > x_1 \wedge X_1 > x_1) \\ &= 1 - F_1(x_1) - F_2(x_2) + F_{12}(x_1, x_2) \\ S_{123} &= 1 - F_1 - F_2 - F_3 + F_{12} + F_{13} + F_{23} - F_{123} \end{aligned}$$

- Pozor, jednoduchý doplnok distribučnej funkcie do plnej pravdepodobnosti znamená pravdepodobnosť prekročenia *aspoň v jednej* premennej (v jednej, v druhej alebo vo všetkých súčasne), teda napr.

$$1 - F_{12}(x_1, x_2) = Pr(X_1 > x_1 \vee X_2 > x_2)$$

10.15 Literatúra

Rozdelenie pravdepodobnosti

- Guo S. (2020) Practical Probability Theory: All About That Single Random Variable.
- Guo S. (2020) Multivariate Probability Theory: All About Those Random Variables.

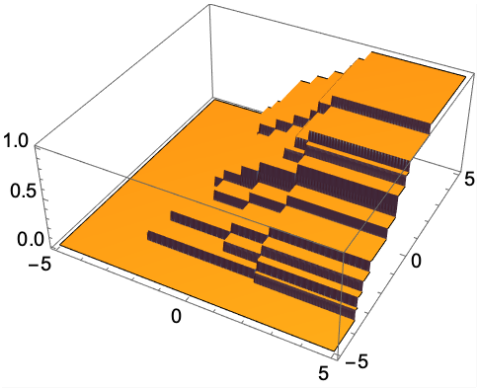
11 Model združeného rozdelenia a aplikácie

11.1 Modelovanie združeného rozdelenia

- *Empirická distribučná funkcia* $\hat{F}_n$ je (nevychýleným) neparametrickým odhadom združenej CDF,

$$\hat{F}_n(\vec{x}) = \frac{1}{n}\sum_{i=1}^{n}\prod_{j=1}^{p} 1(x_{ij} \leq x_j)$$

  kde x_j je prvok argumentu a x_{ij} značí i-te pozorovanie.



- *Jadrovo vyhladená hustota*, s jadrovou funkciou definovanou (pre jednoduchosť) napr. ako súčin jednorozmerných jadrových funkcií K_j so šírkou pásma h_j, je daná vzťahom

$$\hat{f}(\vec{x}) = \frac{1}{n\prod_{j=1}^{p} h_j}\sum_{i=1}^{n}\prod_{j=1}^{p} K_j\left(\frac{x_j - x_{ij}}{h_j}\right)$$

  Delenie šírkou pásma zaručuje, že určitý integrál (plocha pod funkciou) f bude 1.

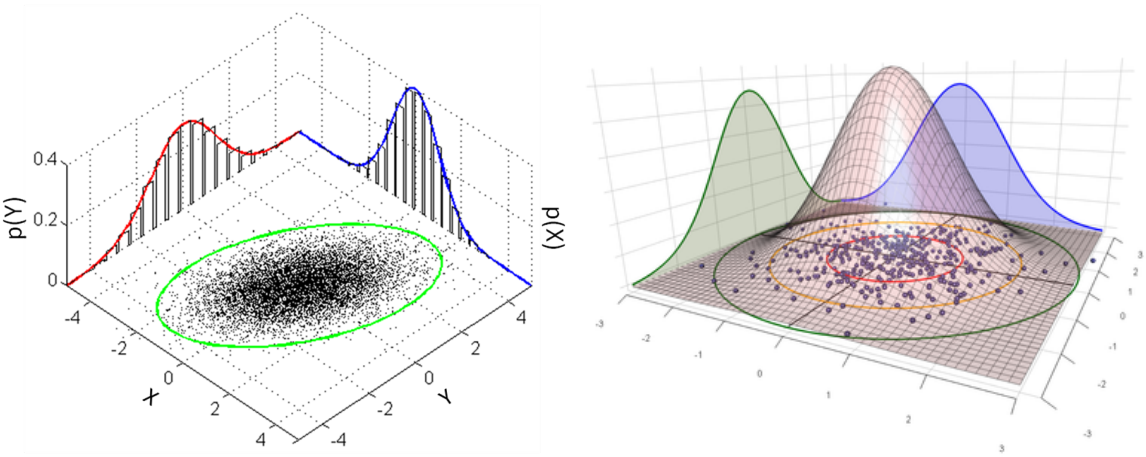
- Výhodou jadrového odhadu rozdelenia je v tom, že máme hladkú funkciu, ktorú možno derivovať. Nevýhodou je výpočtová náročnosť (najmä odhad šírky pásma) so vzrastajúcim p, jav známy ako *kliatba rozmernosti*.
- Kliatbe rozmernosti sa dá vyhnúť špeciálnymi predpokladmi. *Parametrické modely* zavádzajú pomerne silné predpoklady o tom, ako rozdelenie vyzerá, takže stačí už len určiť

parametre. Menej drastické sú predpoklady o nejakom druhu štruktúry prítomnej v rozdelení, ktoré využívajú metódy ako PCA, faktorové modely, zmesi rozdelení či grafické modely (predp. nezávislosť).

- Pre veľa (jednorozmerných) parametrických tried rozdelení existuje rozšírenie na náhodný vektor, napr. hustota združeného normálneho rozdelenia je definovaná

$$f(\vec{x}) = \frac{1}{\sqrt{(2\pi)^p |\ |}} e^{-\frac{1}{2}(\vec{x}-\vec{\mu})^T\ ^{-1}(\vec{x}-\vec{\mu})}$$

kde je kovariančná matica, takže $\Sigma_{ij} = cov[X_i, X_j]$.



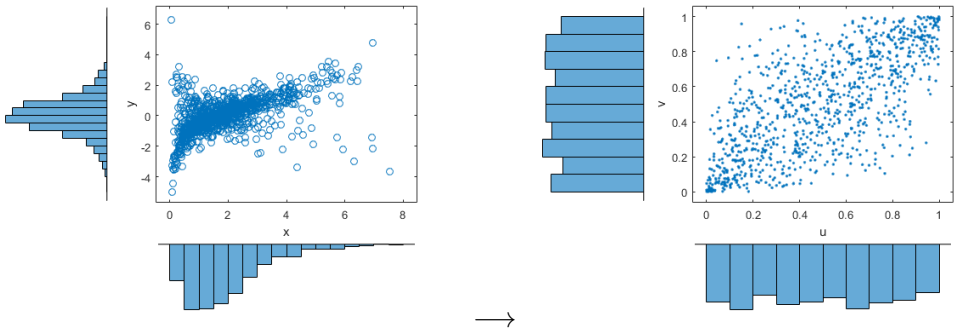
- Kompozitný model umožňuje *rozložiť* analýzu rozdelenia náhodného vektora zvlášť na okrajové rozdelenia a zvlášť na vzťahy medzi premennými.

11.2 Rozklad združeného rozdelenia

- Nevýhoda zovšeobecnení jednorozmerných rozdelení do väčšieho rozmeru spočíva v tom, že marginálne rozdelenia takého združeného rozdelenia sú všetky z *rovnakej* triedy.
- Ak chceme modelovať stochastickú závislosť bez ohľadu na individuálne vlastnosti náhodných premenných, musíme združené rozdelenie reprezentovať ako *marginálie zviazané kopulou*. V reči distribučných funkcií, resp. funkcií hustoty sa rozklad zapíše

$$F(x_1, \ldots, x_p) = C\Big(F_1(x_1), \ldots, F_p(x_p)\Big)$$
$$f(x_1, \ldots, x_p) = c\Big(F_1(x_1), \ldots, F_p(x_p)\Big) \cdot f_1(x_1) \cdot \ldots \cdot f_p(x_p)$$

kde C je kopula náhodného vektora $\vec{X}$, a c je jej hustota.



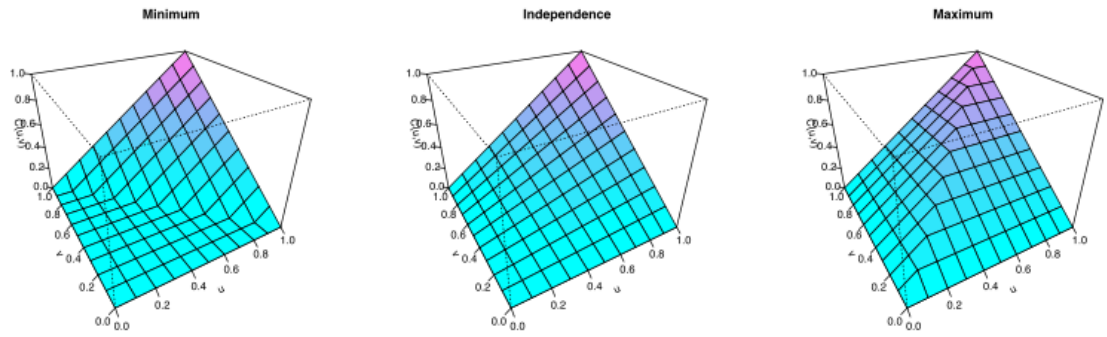
- Transformované náhodné premenné $U_i = F_i(X_i)$ majú rovnomerné rozdelenie na intervale $[0, 1]$.
- Výpočet podmienenej združenej CDF je s kopulami jednoduchší:

$$F_{p|-p}(x_p|\vec{x}_{-p}) = \frac{\int_{-\infty}^{x_p} f(x_1, ..., x_{p-1}, t)dt}{\int_{-\infty}^{\infty} f(x_1, ..., x_{p-1}, t)dt} = \frac{\int_{0}^{F_p(x_p)} c(F_1(x_1), ..., F_{p-1}(x_{p-1}), t)dt}{\int_{0}^{1} c(F_1(x_1), ..., F_{p-1}(x_{p-1}), t)dt}$$

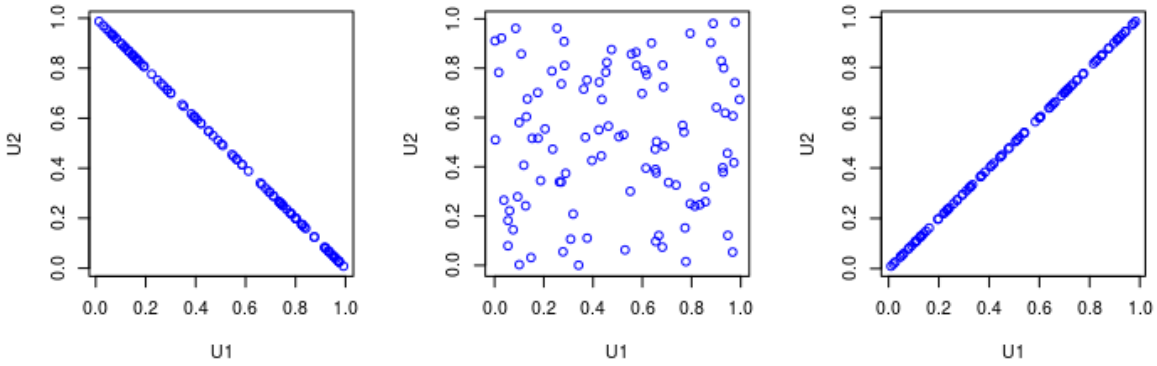
11.3 Kopula

- Vlastnosti:
  1. $C(u_1, \dots, u_2) = 0$ pre ľubovoľné $u_i = 0$,
  2. $C(1, \dots, 1, u_i, 1 \dots, 1) = u_i$,
  3. C je p-rastúca funkcia, čo pre $p = 2$ znamená $C(v_1, v_2) - C(v_1, u_2) - C(u_1, v_2) + C(u_1, u_2) \geq 0$, pričom $u_i \leq v_i$.
- Špeciálnymi prípadmi sú kopule M, Π, W, ktoré predstavujú
  - úplnú negatívnu závislosť, $W(u_1, u_2) = \max(0, u_1 + u_2 - 1)$,
  - nezávislosť, $\Pi(u_1, \dots, u_p) = u_1 \cdot \dots \cdot u_p$
  - úplnú závislosť, $M(u_1, \dots, u_p) = \min(u_1, \dots, u_p)$,

  Funkcia $W(u_1, \dots, u_p) = \max(0, 1 - p + \sum_{i=1}^{p} u_i)$ je dolným ohraničením pre akúkoľvek kopulu C, platí
  $$W < C \leq M$$
  ale iba v dvojrozmernom prípade platí $W \leq C$, pretože pre $p > 2$ už W nie je kopula.
- Graf špeciálnych kopúl W, Π, M:

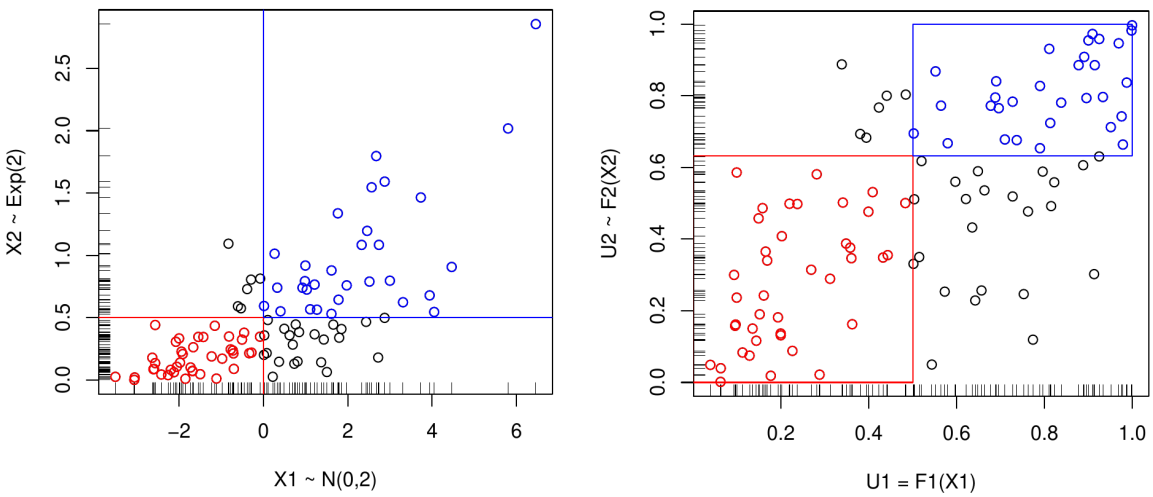


- Bodové grafy náhodných výberov simulovaných z kopúl W, Π, M:



11.4 Využitie modelu závislosti

- Kopula C teda predstavuje tú časť združeného rozdelenia, ktorá popisuje závislosť medzi premennými.
- Pomocou modelu závislosti môžeme
  - skúmať vlastnosti vzťahu ako napr. závislosť v extrémoch (tail dependence) či rôzne (a)symetrie,
  - určovať pravdepodobnosť, že premenné (ne)prekročia kritickú hodnotu,
  - vypočítať kvantily, ktoré (ne)budú prekročené s danou pravdepodobnosťou.
- Konkrétne je možné priamo odhadnúť *pravdepodobnosti*:
  - $Pr(U_1 \leq u_1 \wedge U_2 \leq u_2) = C(u_1, u_2)$, súčasného neprekročenia hodnôt u_1, u_2;

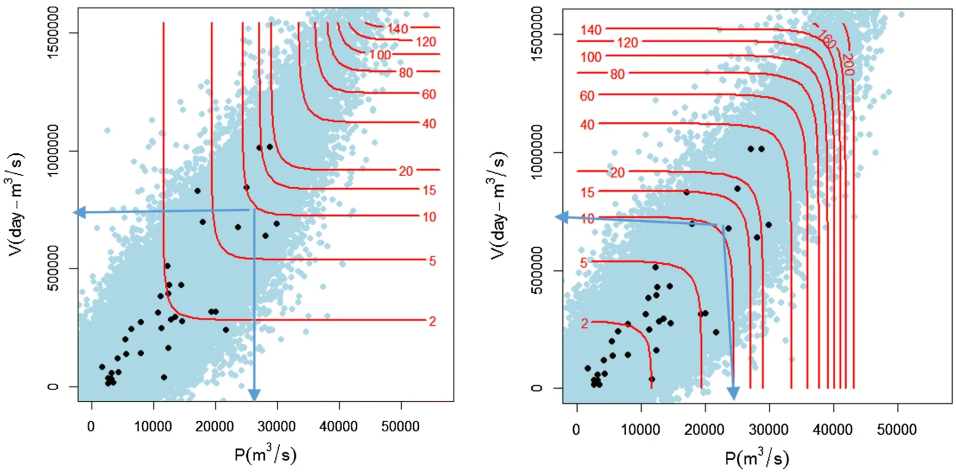


  - $Pr(U_1 > u_1 \vee U_2 > u_2) = 1 - C(u_1, u_2)$, že aspoň jedna premenná prekročí daný prah;
  - $Pr(U_1 > u_1 \wedge U_2 > u_2) = 1 - u_1 - u_2 + C(u_1, u_2)$, súčasného prekročenia (survival function);
  - $Pr(U_1 \leq u_1 | U_2 = u_2) = \frac{\partial C(u_1,u_2)}{\partial u_2}$, neprekročenia pri jednej prem. za podmienky, že druhá dosiahla určitý prah,
  - $Pr(U_1 \leq u_1 | U_2 \leq u_2) = \frac{C(u_1,u_2)}{u_2}$, prípadne. že ho neprekročila
  - $Pr(U_1 > u_1 | U_2 > u_2) = 1 - \frac{u_1 - C(u_1,u_2)}{1-u_2}$, resp. prekročila.

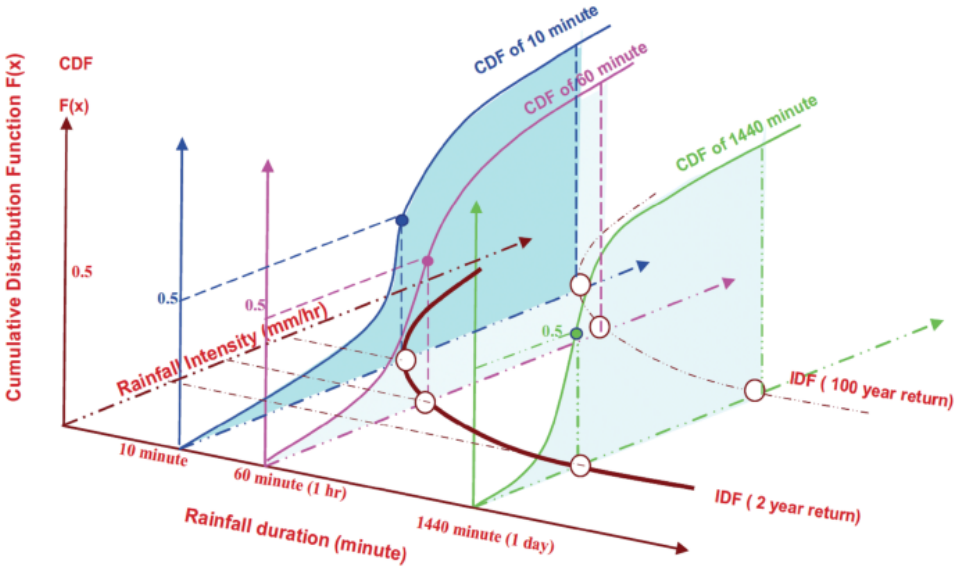
- V hydrologickej praxi sa často konštruuje *vrstevnicový* graf pravdepodobnostného rozdelenia na odčítanie doby návratu

$$T = \frac{1}{Pr(X_1 > x_1 \quad \wedge/\vee \quad X_2 > x_2)} \text{ [č.j.]}$$

  extrémneho javu (povodeň, zrážky, sucho).
- Napr. doba návratu povodňovej vlny charaktrizovanej objemom V (volume) a kulminačným prietokom P (peak) pre pravdepodobnosti súčasného (*vavo*) alebo "jednotlivého" (*vpravo*) prekročenia daných hodnôt veličín V a P. Graf obsahuje pozorované (čierne body) aj simulované povodne.



- Alebo doba návratu zrážkovej udalosti charakterizovanej intenzitou a trvaním (duration) pre pravdepodobnosti súčasného neprekročenia. Vrstevnice distribučnej funkcie CDF sa tu označujú ako čiary IDF (intensity-duration-frequency curves).



11.5 Parametrické triedy

Nasledujúce triedy kopúl sú **užitočné v 2D problémoch**:

- *Eliptické* - napr. normálna (Gaussovská) alebo t-kopula, modelujú iba lineárny vzťah, konštruujú sa štandardizáciou marginálií eliptických rozdelení,

$$C_\Phi(u_1, ..., u_p) = \Phi\left[\Phi_1^{-1}(u_1), ..., \Phi_p^{-1}(u_p)\right]$$

- *Archimedovské* - napr. Gumbelova, Claytonova, Frankova, konštruujú sa jednoducho pomocou *generátora* $f\colon [0,1] \to [0,\infty)$ ale sú permutačne symetrické (exchangeability),

$$C(u_1, \dots, u_p) = f^{(-1)}\Big(f(u_1) + \dots + f(u_p)\Big)$$

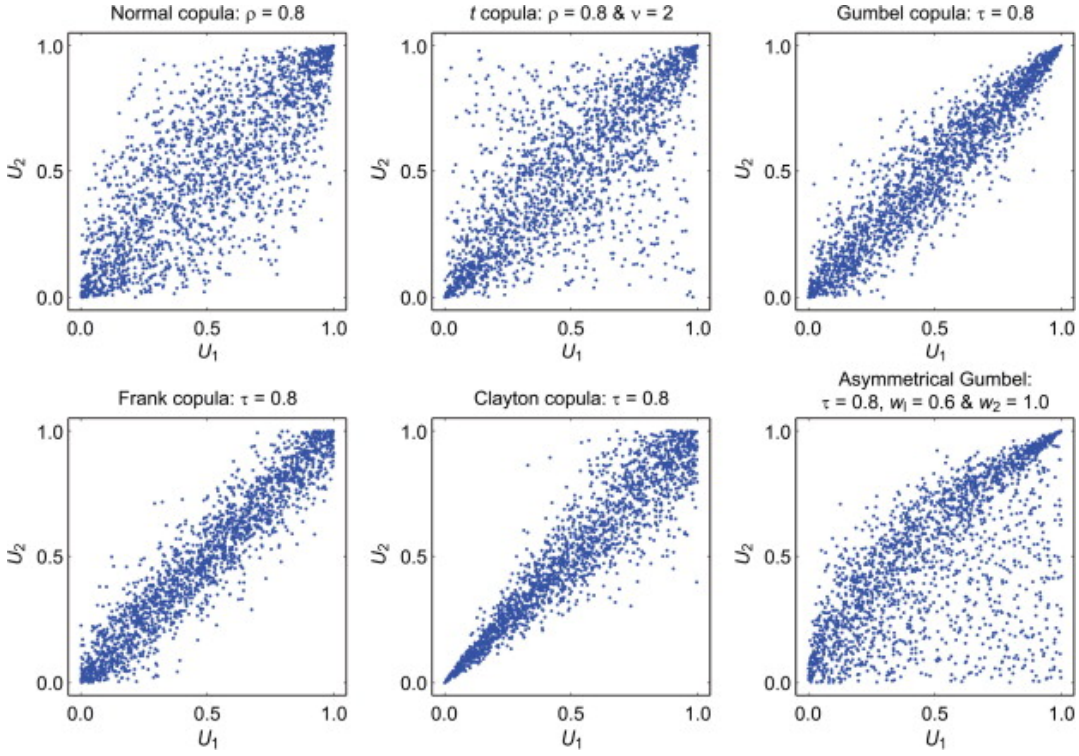
- *Extreme-value* - napr. Gumbelova, Galambosova, Hüsler-Reiss, modelujú maximálne hodnoty, konštruujú sa pomocou *dependence funkcie* $\ell\colon [0,\infty)^p \to [0,\infty)$ alebo Pickandsovej depend.funkcie $D\colon \Delta_{p-1} \to [1/p, 1]$ (kde Δ_{p-1} je jednotkový simplex)

$$\begin{aligned} C(u_1, \dots, u_p) &= \exp(-\ell(-\log u_1, \dots, -\log u_p)) \\ &= e^{(\sum_{i=1}^p \log u_i)\ D\left(\frac{\log u_1}{\sum_{i=1}^p \log u_i}, \dots, \frac{\log u_p}{\sum_{i=1}^p \log u_i}\right)} \end{aligned}$$

pomocou *dependence funkcie* $\ell\colon [0,\infty)^p \to [0,\infty)$ alebo Pickandsovej depend.funkcie $D\colon \Delta_{p-1} \to [1/p, 1]$ (kde Δ_{p-1} je jednotkový simplex)

$$\begin{aligned} C(u_1, \dots, u_p) &= \exp(-\ell(-\log u_1, \dots, -\log u_p)) \\ &= e^{(\sum_{i=1}^p \log u_i)\ D\left(\frac{\log u_1}{\sum_{i=1}^p \log u_i}, \dots, \frac{\log u_p}{\sum_{i=1}^p \log u_i}\right)} \end{aligned}$$

Pre $p > 2$ je pomerne komplikované hľadať dependence funkcie.



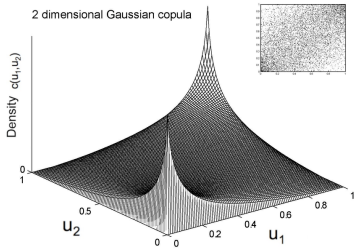
- Na obrázku šestice simulovaných náhodných výberov

– prvé dve sú eliptické,
– ďalšie tri Archimedovské (iba Frankova je radiálne symetrická),
– v poslednom stĺpci sú EV kopuly (líšia sa permutačnou symetriou).

Nasledujúce triedy (a konštrukčné metódy) sa v súčasnosti používajú ako dostatočne flexibilné a zároveň analyticky rozumne dostupné vo **viacrozmerných problémoch**:

- *Eliptické kopuly* s explicitným vyjadrením hustoty, napr. Gaussovská kopula s korelačnou maticou $\mathbf{R}$ a kvantilovou funkciou normovaného norm. rozd. Φ^{-1},

$$c(\vec{u}) = \frac{1}{\sqrt{|\mathbf{R}|}} e^{-\frac{1}{2}\left(\Phi^{-1}(u_1),\ldots,\Phi^{-1}(u_p)\right)(\mathbf{R}^T-\mathbf{I})\left(\Phi^{-1}(u_1),\ldots,\Phi^{-1}(u_p)\right)^T}$$

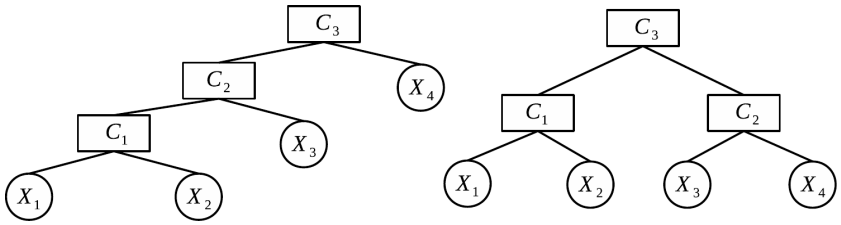


- *Hierarchické Archimedovské kopuly* sú zložené z Archimedovských kopúl, napr.

$$C(u_1,u_2,u_3) =$$

$$C_3\Big(C_2(C_1(u_1,u_2),u_3),u_4\Big) \qquad C_3\Big(C_1(x_1,x_2),C_2(x_3,x_4)\Big)$$

fully nested partially nested



C_1, C_2, C_3 však musia byť z rovnakej parametrickej triedy (parametric family) Archimedovských kopúl a závislosť musí s úrovňou hierarchie klesať.

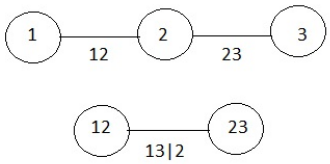
- *Vine kopuly* využívajú podmienenú pravdepodobnosť na faktorizáciu viacrozmerného združeného rozdelenia pomocou dvojrozmerných kopúl (pair-copula decomposition).

– Napr. pre $p = 3$ jeden z možných zápisov (rozkladov) je

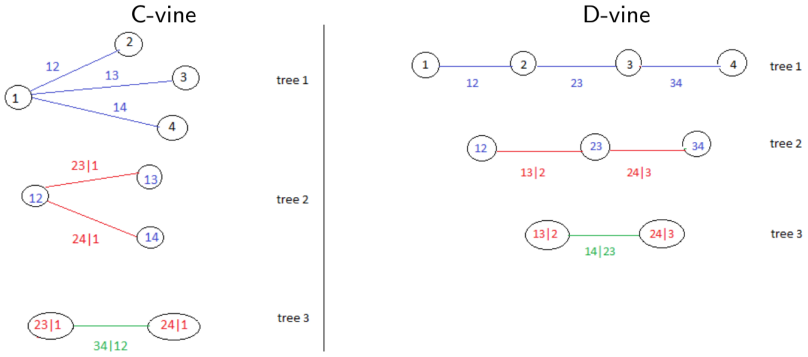
$$\begin{aligned} f(x_1, x_2, x_3) &= f_1(x_1) \cdot f_{2|1}(x_2|x_1) \cdot f_{3|12}(x_3|x_1, x_2) \\ &= f_1(x_1) \cdot \\ &\quad \cdot c_{12}\left[F_1(x_1), F_2(x_2)\right] \cdot f_2(x_2) \cdot \\ &\quad \cdot c_{31|2}\left[F_{3|2}(x_3|x_2), F_{1|2}(x_1|x_2)\right] \cdot \underbrace{c_{23}\left[F_2(x_2), F_3(x_3)\right] \cdot f_3(x_3)}_{f_{3|2}(x_3|x_2)} \end{aligned}$$

kde sme použili úpravu $f_{2|1}(x_2|x_1) = \frac{f(x_1,x_2)}{f_1(x_1)} = \frac{c_{12}\left(F_1(x_1),F_2(x_2)\right) f_1(x_1) f_2(x_2)}{f_1(x_1)}$ (pričom podobne bola vyjadrená aj hustota $f_{3|12}$, kde sme z podmieňujúcich premenných zvolili X_2), a podmienené distribučné funkcie $F_{i|j}(x_i|x_j) = \frac{\partial C_{ij}\left(F_i(x_i),F_j(x_j)\right)}{\partial F_j(x_j)}$.

– S pribúdajúcimi rozmermi rastie aj komplexnosť všetkých možných rozkladov, preto je model graficky reprezentovaný sekvenciou vnorených stromov s neorientovanými hranami (*vine tree*).



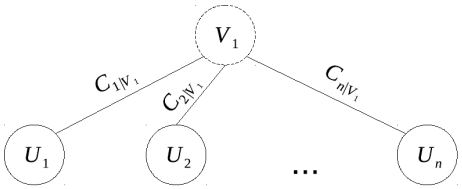
– Pri viac ako troch premenných môžu mať všeobecné vine stromy (regular vines) rôzne tvary, najpopulárnejšie sú C-vine (canonical) a D-vine (drawable).



- *Faktorové kopuly.* V jednofaktorovom modeli je závislosť medzi premennými U_i indukovaná faktorom V_1 cez tzv. linking kopuly $C_{i|V_1}$, na ktoré nie sú kladené žiadne obmedze-

nia,

$$C(u_1, \dots, u_p) = \int_0^1 \prod_{i=1}^{p} C_{i|V_1}(u_i, v_1) dv_1$$



11.6 Odhad parametrov rozdelenia

Na odhad rozdelenia zo zvolenej parametrickej triedy sa používajú hlavne tieto tri metódy:

1. *Metóda momentov* - využíva priamy funkčný vzťah medzi momentami rozdelenia (ktoré sa dajú odhadnúť z náhodného výberu) a parametrami rozdelenia. Výhoda je v jednoduchosti (rýchlosti výpočtu), nevýhoda zas v častej nedostupnosti takých vzťahov, a tiež v tom, že metóda nedosahuje vlastnosti optimálneho odhadu ako ostatné dve.
2. *Metóda maximálnej vierohodnosti* (maximum likelihood, ML) - využíva hustotu rozdelenia na vyjadrenie pravdepodobnosti získania daných pozorovaní podmienenú danými parametrami, teda

   $$\hat{\theta} = \arg\max_{\theta \in \Theta} \frac{1}{n} \sum_{i=1}^{n} \ln f(X_{i1}, \dots, X_{ip}|\theta).$$

   Odhad je konzistentný (s rastúcim n pravdepodobnosťou konverguje k správnej hodnote), nevychýlený (angl. unbiased), výdatný (s najmenším rozptylom, angl. efficient) a normálne rozdelený (asymptoticky). Nevýhoda je v komplikovanosti výpočtu, vychýlenosti pre malé výbery a citlivosti na počiatočné podmienky.
3. *Metóda najmenších štvorcov* - pomerne univerzálna metóda, hoci nemá tak dobré vlastnosti ako ML. Parametre môžu byť odhadnuté minimalizáciou napr. vzdialenosti teoretickej a empirickej distribučnej funkcie,

   $$\hat{\theta} = \arg\min_{\theta \in \Theta} \sum_{i=1}^{n} \left(F(X_{i1}, \dots, X_{ip}|\theta) - \hat{F}_n(X_{i1}, \dots, X_{ip}) \right)^2$$

11.7 Príklad

Príklad (autá)

Výstavba a aplikácia modelu dvojrozmerného združeného rozdelenia (objemu valcov a dojazdu automobilov pomocou datasetu *datasets::mtcars*) od prieskumnej analýzy až po zobrazenie podmienenej funkcie hustoty.

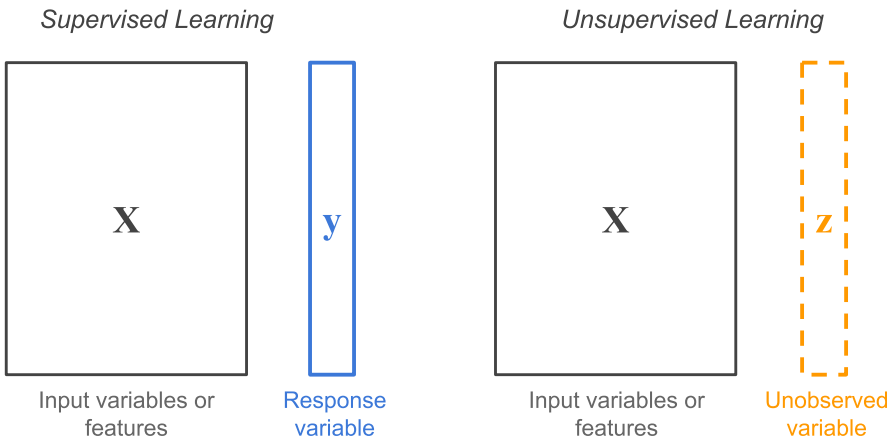
11.8 Literatúra

Kopuly

- Nelsen, R. B. (2007). An introduction to copulas. Springer Science & Business Media.
- Okhrin, O., Ristig, A., & Xu, Y. F. (2017). Copulae in high dimensions: an introduction. Applied quantitative finance, 247-277.
- Joe, H. (2015). Dependence modeling with copulas. no. 134 in monographs on statistics and applied probability.
- Schirmacher, D., Schirmacher, E. (2008). Multivariate dependence modeling using pair-copulas. Technical report.
- Salvadori G., De Michele C. (2010) Multivariate multiparameter extreme value models and return periods: a copula approach. DOI

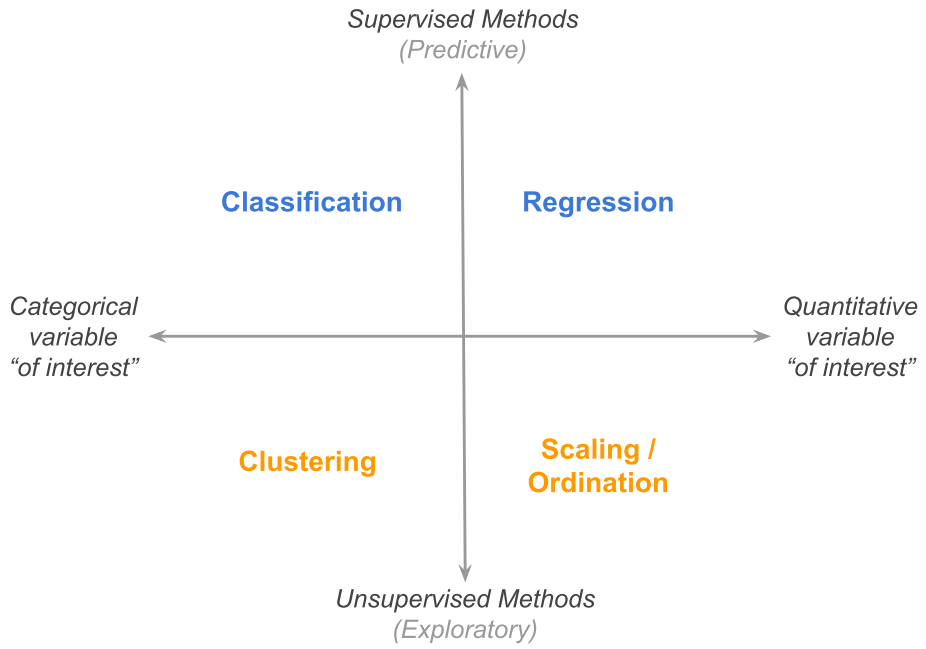
12 Metódy neasistovaného učenia

12.1 Úvod



- V neasistovanom (štatistickom/strojovom) učení odozva Y *nie je definovaná*, všetky vlastnosti pozorovaného objektu sú rovnocenné náhodné premenné $X_1, \ldots, X_p$.
- Cieľom analýzy tak už nie je predpovedať hodnotu (alebo rozdelenie premennej), ale skôr *objavovať zaujímavé* veci, napr. ako vhodne zobraziť údaje, alebo nájdenie skupín medzi pozorovanými objektami (riadky datasetu) či ich vlastnosťami (stĺpce datasetu).
- Preto sa neasistované učenie často využíva v rámci prieskumnej analýzy údajov (exploratory data analysis, EDA).
- Má tendenciu byť viac subjektívna, a pretože nemá jeden cieľ (predpoveď), je ťažšie vyhodnotiť úspešnosť.
- Základným prístupom ku hľadaniu skupín medzi pozorovaniami je zhluková analýza (*clustering*). Naopak, hľadaniu skupín užitočných a menej užitočných vlastností sa venuje oblasť metód redukcie rozmernosti (*dimensionality reduction*, ordination), ktorá je užitočné aj pre optimálne zobrazenie údajov.
- Redukcia rozmernosti v *širšom kontexte* zahŕňa
  - výber vlastností (*feature selection*, častejšie v asistovanom učení – napr. regularizačná technika LASSO) a
  - extrahovanie vlastností (*feature extraction*).

- Základnou metódou používanou na extrakciu užitočných vlastností v oblasti neasistovného učenia je analýza hlavných komponentov (*principal component analysis*, PCA).[1]

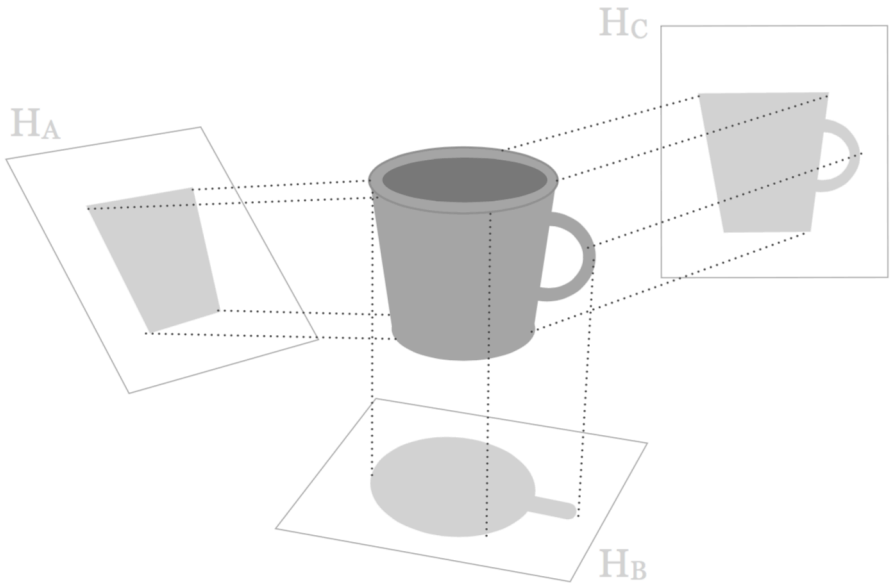


12.2 Analýza hlavných komponentov

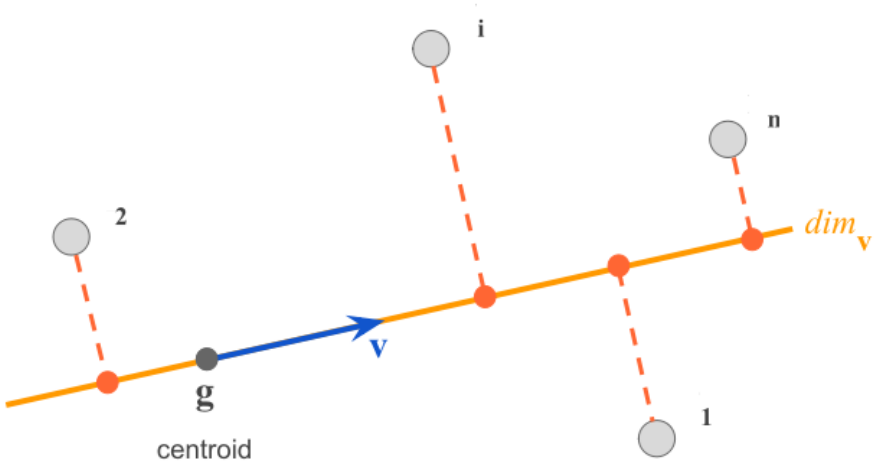
12.2.1 Princíp

- Vlastnosti $X_1, \ldots, X_p$ tvoria **osi** p-rozmerného priestoru a skúmané objekty (štatistické jednotky) sa v ňom prostredníctvom pozorovaných hodnôt ich vlastností x_{ij} (kde $i = 1, \ldots, n$, a $j = 1, \ldots, p$) zobrazia ako **body**.
- Ak by sme aj dokázali vidieť vo viac ako 3-rozmernom priestore tvar mraku bodov (charakter skúmaného javu), potrebujeme nájsť spôsob, ako túto *informáciu* sprostredkovať ďalším ľuďom (či už obrázkom alebo modelom).

[1] Ďalšími ordinačnými metódami sú napr. principal coordinate analysis (PCoA) resp. multidimensional scaling (MDS), uniform manifold approximation and projection for dimension reduction (UMAP), t-distributed stochastic neighbor embedding (t-SNE)…



- Hľadáme podpriestor, do ktorého projekcia "najlepšie" reprezentuje mrak bodov.
- V prípade PCA, os Z "najlepšieho" (1D) podpriestoru vedie v *smere najväčšieho rozptylu* bodov.
- Ekvivalentne, *kolmá vzdialenosť* bodov od tohto podpriestoru je *najmenšia*.



- Os Z sa nazýva hlavný (principal) komponent a je lineárnou kombináciou osí X_j pôvodného priestoru,

$$Z = v_1 X_1 + ... + v_p X_p$$

pričom pre vektor $\mathbf{v}$ tzv. záťaží (*loadings*) platí $||\mathbf{v}|| = 1$. Fixovaním dĺžky vektora sa zafixuje aj rozptyl hlavného komponentu, čo je nevyhnutné pre nájdenie maxima.

12.2.2 Optimalizácia

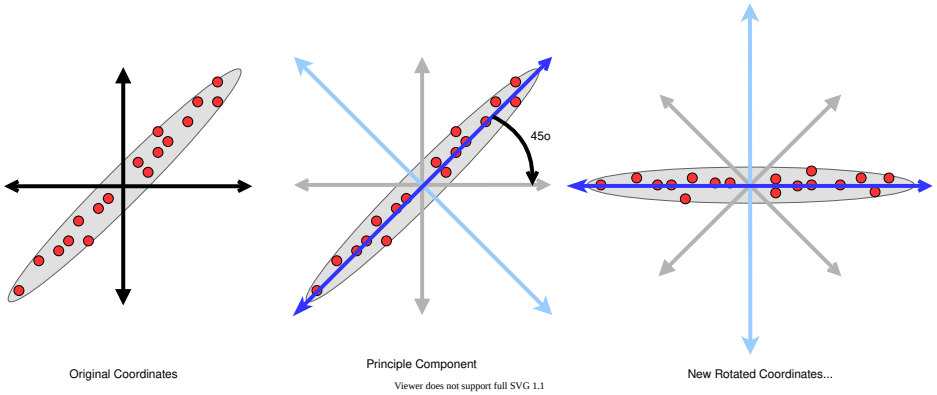
- Nech stĺpce matice $\mathbf{X}$ obsahujú (*centrované*) súradnice bodov v pôvodnom priestore a vektor

$$\mathbf{z} = \mathbf{X}\mathbf{v}$$

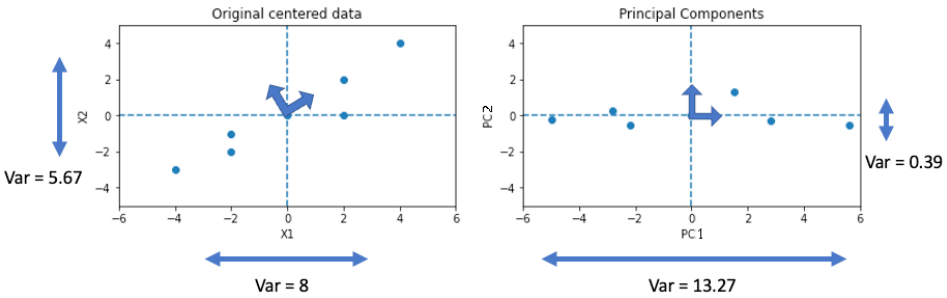
obsahuje polohu bodov na osi Z. V žargóne sa tieto nové súradnice nazývajú *scores* (zásahy?).
- Vyjadrime rozptyl novej premennej Z, ktorý chceme maximalizovať:

$$\begin{aligned} var(Z) &= E[Z^2] \\ &= \frac{1}{n}\mathbf{z}^T\mathbf{z} \\ &= \frac{1}{n}\mathbf{v}^T\mathbf{X}^T\mathbf{X}\mathbf{v} \\ &= \mathbf{v}^T\left(\frac{1}{n}\mathbf{X}^T\mathbf{X}\right)\mathbf{v} \\ &= \mathbf{v}^T\ \mathbf{v} \end{aligned}$$

kde je kovariančná matica náhodného vektora $(X_1, ..., X_p)$.
- Táto nová premenná pritom nemusí byť jediná, vo všeobecnosti sa ich dá nájsť rovnaký počet ako je vlastností. Označíme ich $(Z_1, ..., Z_p)$ alebo $\{PC_j\}_{j=1}^p$ (kde $j = 1, \ldots, p$), každému zodpovedá smerový vektor $\mathbf{v}_j$ a sú na seba kolmé (rovnako osi X_j), pretože z geometrického hľadiska ide o obyčajnú rotáciu súradnicového systému.



- Rozmernosť podpriestoru (počet hlavných komponentov) je subjektívna voľba, hlavné komponenty sú *zoradené* podľa variability, ktorú "vysvetľujú".



- Vektor záťaží $\mathbf{v}_1$ *prvého hlavného komponentu* maximalizuje celkový rozptyl s uplatnením jedinej podmienky,

$$\mathbf{v}_1 = \arg\max_{\mathbf{v}} (\mathbf{v}^T\ \mathbf{v}) \quad \text{pre} \quad \mathbf{v}^T\mathbf{v} = 1$$

  Uplatnením metódy Lagrangeových multiplikátorov

$$L(\mathbf{v}) = \mathbf{v}^T\ \mathbf{v} - \lambda(\mathbf{v}^T\mathbf{v} - 1)$$
$$\frac{\partial L}{\mathbf{v}} = 2\ \mathbf{v} - 2\lambda\mathbf{v}$$
$$\mathbf{v}_1 - \lambda_1\mathbf{v}_1 = 0$$
$$\mathbf{v}_1 = \lambda_1\mathbf{v}_1$$

  dostaneme riešenie, ktorým je (normovaný) vlastný vektor zodpovedajúci najväčšiemu vlastnému číslu (λ_1) matice .
- Vektor záťaží $\mathbf{v}_2$ *druhého* hlavného komponentu musí pri maximalizácii spĺňať okrem jednotkovej dĺžky aj podmienku kolmosti na $\mathbf{v}_1$, čiže

$$\mathbf{v}_2 = \arg\max_{\mathbf{v}} (\mathbf{v}^T\ \mathbf{v}) \quad \text{pre} \quad \mathbf{v}^T\mathbf{v} = 1 \quad \text{a} \quad \mathbf{v}_1^T\mathbf{v} = 0$$

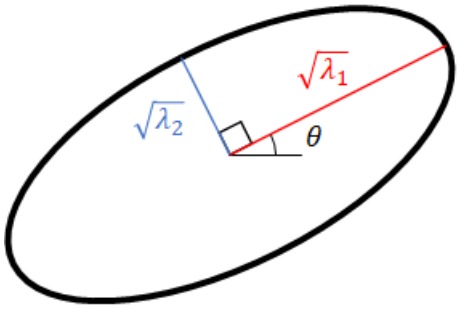
  Riešenie vo forme (normovaného) vlastného vektora matice , ktorý zodpovedá druhému najväčšiemu vlastnému číslu (λ_2), spĺňa aj druhú podmienku, pretože vlastné vektory tvoria ortonormálnu bázu.
- Takto môžeme hlavné komponenty nájsť všetky naraz, a to *diagonalizáciou* kovariančnej matice , čiže rozkladom (*eigenvalue decomposition*)

$$= \mathbf{V}\ \mathbf{V}^T$$

  kde

  - je diagonálna matica s vlastnými číslami $(\lambda_1, \ldots, \lambda_p)$ na diagonále, predstavuje kovariančnú maticu hlavných komponentov,
  - $\mathbf{V} = (\mathbf{v}_1, \ldots, \mathbf{v}_p)$ je matica s vlastnými vektormi v stĺpcoch, pričom tie sú ortonormálne, takže platí $\mathbf{V}^T\mathbf{V} = 1$.

- V dvojrozmernom prípade je $\mathbf{V}$ jednoduchá matica rotácie o uhol θ,



$$\mathbf{V} = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix}$$

- *Symetrické* matice majú p reálnych vlastných čísel (vrátane duplicít), ktorým zodpovedá p navzájom lineárne nezávislých vlastných vektorov. Množina vlastných čísel sa nazýva aj *spektrum* matice.
- Smer vektorov záťaží $\mathbf{v}_1, \ldots, \mathbf{v}_p$ je unikátny, ale ich orientácia nie, dve implementácie metódy teda môžu dať hodnoty s opačnými znamienkami.
- Ak sú vlastnosti $X_1, \ldots, X_p$ pozorované v rôznych jednotkách, je okrem centrovania potrebné aj ich *preškálovanie* a to podelením štandardnou odchýlkou. V opačnom prípade by vlastnosti s najväčším rozptylom zásadne ovplyvnili transformáciu.

12.2.3 Aplikácie

- Transformáciu medzi vlastnosťami (features) v matici $\mathbf{X}$ a zásahmi (scores) v matici $\mathbf{Z} = (\mathbf{z}_1, \ldots, \mathbf{z}_p)$ realizujeme pomocou záťaží (loadings) $\mathbf{V}$ pomocou vzťahu

$$\underset{n\times p}{\mathbf{Z}} = \underset{n\times p}{\mathbf{X}} \cdot \underset{p\times p}{\mathbf{V}}$$
$$\mathbf{X} = \mathbf{Z} \cdot \mathbf{V}^T$$

  takže pri použití všetkých hlavných komponentov dokážeme obnoviť pôvodné pozorovania vlastností.
- Zmysel PCA je však v redukcii rozmernosti, preto sa v aplikáciách nepoužívajú všetky PC, ale iba tie "najvýznamnejšie".
- Ak je treba zobraziť mnohorozmerný priestor, zvyčajne siahneme po prvých dvoch, najvýš prvých troch PC.
- PCA sa dá prirodzene použiť na filtrovanie dát (*odstránenie šumu*),

$$\underset{n\times p}{\mathbf{X}} = \underset{n\times q}{\mathbf{Z}} \cdot \underset{q\times p}{\mathbf{V}^T} + \underset{n\times p}{\mathbf{R}} \qquad \text{pre} \quad q < p$$

  kde matica $\mathbf{R}$ obsahuje rezíduá.

- Hlavné komponenty môžu poslúžiť ako vysvetľujúce premenné v klasickej lineárnej regresii (*principal component regression*, PCR)

$$Y = \beta_0 + \beta_1 Z_1 + ... + \beta_q Z_q + \varepsilon$$

  za *predpokladu*, že smer najväčšej variability vlastností dokáže vysvetliť variabilitu odozvy. Aj keď to nebýva splnené úplne, predpovedná schopnosť takého modelu je často dobrá a zároveň rieši problém multikolinearity. Počet hlavných komponentov zaradených do regresie sa určí napr. metódou krížovej validácie.
- Pozn.: Smery vysvetľujúcich premenných v PCR sú určené neasistovaným prístupom. *Asistovanú alternatívu* poskytuje metóda parciálnych najmenších štvorcov (*partial least squares*, PLS).
- Iná aplikácia sa núka v oblasti *dopočítania chýbajúcich hodnôt* (missing value imputation). Tie však
  - musia byť *náhodne chýbajúce* (napr. vybité baterky v elektrickej váhe),
  - nie *systematicky* (neschopnosť pacienta zúčastniť sa merania v dôsledku extrémnej nadváhy).

  Štandardný postup, v ktorom sa chýbajúce hodnoty nahradia priemerom v danom stĺpci (vlastnosti), zanedbáva korelácie medzi stĺpcami. Môže však poslúžiť ako počiatočný stav $\mathbf{X}^{(0)}$ v nasledujúcom algoritme (s indexom iterácie i):
  1. pomocou $\mathbf{X}^{(i-1)}$ sa vypočítajú matice $\mathbf{Z}$ a $\mathbf{V}$ (pre prvých q komponentov),
  2. z nich pomocná matica $\mathbf{X}$ a
  3. z tej sa doplnia hodnoty na chýbajúcich pozíciách do $\mathbf{X}^{(i-1)}$, takže vznikne aktualizovaná matica $\mathbf{X}^{(i)}$.
- Týmto spôsobom sa dajú "dopočítať" napr. aj preferenčné matice, v ktorých riadky predstavujú zákazníkov, stĺpce zodpovedajú produktom (napr. film) a samotné prvky obsahujú hodnotenie produktu zákazníkom. Také matice sú veľmi riedke (*nutne chýbajúce* hodnoty), ale ak je dostatočný prekryt riadkov (každý film videlo viac zákazníkov), odporúčací algoritmus (recommender system) zákazníkov zaradí do "skupiniek" a filmy do "žánrov". Jedna skupinka obľubuje jeden žáner (spolu je ich toľko ako hlavných komponentov), potom zásahy (scores) reprezentujú silu príslušnosti zákaznikov do skupiniek, a záťaže (loadings) reprezentujú intenzitu, s akou film patrí do žánru.

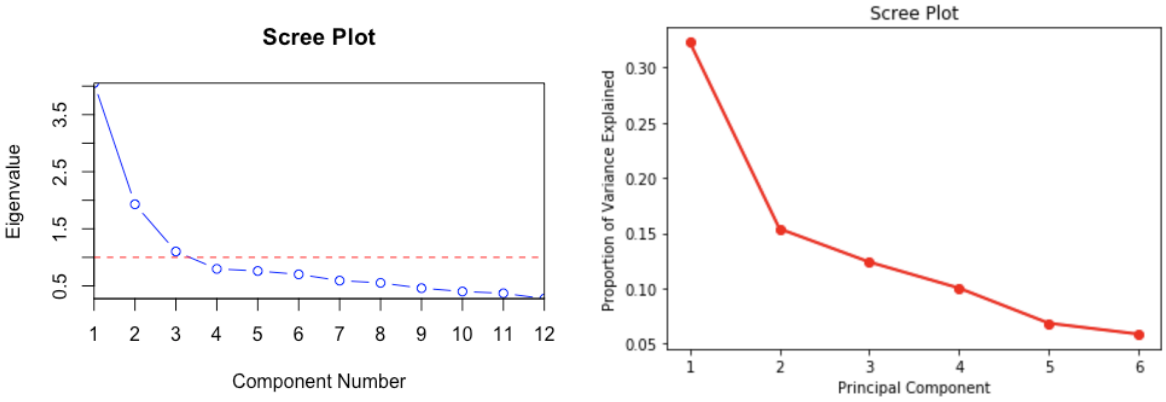
12.2.4 Určenie počtu PC

- Čím menej hlavných komponentov by sme mohli použiť, tým ľahšie by sa dali interpretovať (a my by sme ľahšie porozumeli dátam). Koľko "málo" je optimálne?
- Naneštastie neexistuje jediná alebo jednoduchá odpoveď.
- V rozhodnutí však pomôže tzv. *scree plot* (suťový graf?). Na horizontálnej osi má index (poradie) hlavného komponentu, na vertikálnej osi y môže mať buď priamo varianciu

λ_k vysvetlenú k-tym komponentom, alebo jej pomer ku celkovej variancii (*proportion of variance explained*, PVE)

$$PVE(k) = \frac{\lambda_k}{\sum_{j=1}^{p} \lambda_j}$$

- Za q zvolíme také najväčšie k,
  - pre ktoré $\lambda'_k > 1$ (Kaiser criterion, λ'_k je vl.číslo *korelačnej* matice), alebo
  - od ktorého je pokles PVE výrazne pomalší ("lakeť" klesajúcej funkcie).



Príklad (zločiny)

- Štatistický súbor údajov *datasets::USArrests* o počte zatknutí na 100 tis. obyvateľov v roku 1973 v 50 štátoch USA podľa typu zločinu (vražda, napadnutie, znásilnenie). Dataset navyše obsahuje podiel obyvateľov žijúcich v mestách (*UrbanPop*).
- Zobrazenie datasetu je už aj pre 4 premenné pomerne náročné.

```
c("mean", "var") |>
  sapply(function(x) apply(USArrests, 2, FUN = x)) |>
  t() |> round(1)
```

```
     Murder Assault UrbanPop Rape
mean    7.8   170.8     65.5 21.2
var    19.0  6945.2    209.5 87.7
```

- Rozptyl počtu napadnutí (*Assault*) je oveľa väčší ako u ostatných. Hoci ide o bezrozmerné premenné, škálovanie na rovnaký rozptyl je nutné, inak by bol prvý hlavný komponent takmer výhradne v smere napadnutia a nič zaujímavé by sme sa nedozvedeli.
- V Rku na analýzu PC slúži napr. funkcia *stats::prcomp*. Jej výstup obsahuje

– (*sdev*) štandardnú odchýlku vysvetlenú jednotlivými PC,
– (*rotation*) maticu záťaží **V**, ktorou sa *rotuje* súradnicový systém vlastností do súr.systému hlavných komponentov,
– (*center*) centroid datasetu,
– (*scale*) štandardnú odchýlku jednotlivých vlastností, ktorou sa škálujú na rovnakú mierku a
– (x) maticu zásahov **Z**.

Zobrazíme maticu záťaží a časť matice zásahov:

```
# manuálny výpočet:
# USArrests |> cor() |> eigen() |> getElement("values") |> sqrt()  # std.dev.
# V <- USArrests |> cor() |> eigen() |> getElement("vectors")  # loadings
# Z <- USArrests |> as.matrix() |> apply(2, scale) |> (`%*%`)(V)  # scores

# automatika:
fit <- prcomp(USArrests, scale = TRUE)

fit$rotation |> round(3)
```

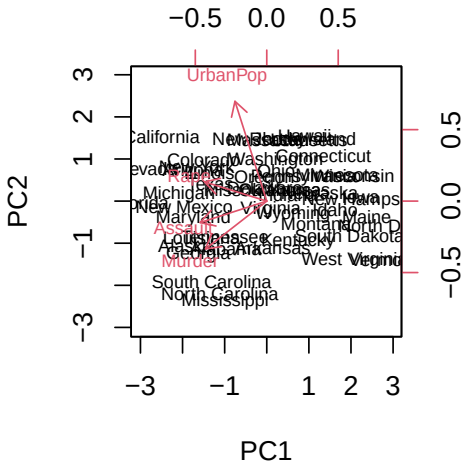
```
            PC1    PC2    PC3    PC4
Murder   -0.536 -0.418  0.341  0.649
Assault  -0.583 -0.188  0.268 -0.743
UrbanPop -0.278  0.873  0.378  0.134
Rape     -0.543  0.167 -0.818  0.089
```

```
fit$x |> head() |> round(2)
```

```
             PC1   PC2   PC3   PC4
Alabama    -0.98 -1.12  0.44  0.15
Alaska     -1.93 -1.06 -2.02 -0.43
Arizona    -1.75  0.74 -0.05 -0.83
Arkansas    0.14 -1.11 -0.11 -0.18
California -2.50  1.53 -0.59 -0.34
Colorado   -1.50  0.98 -1.08  0.00
```

- Dáta s osami oboch systémov znázorníme pomocou *biplotu*.

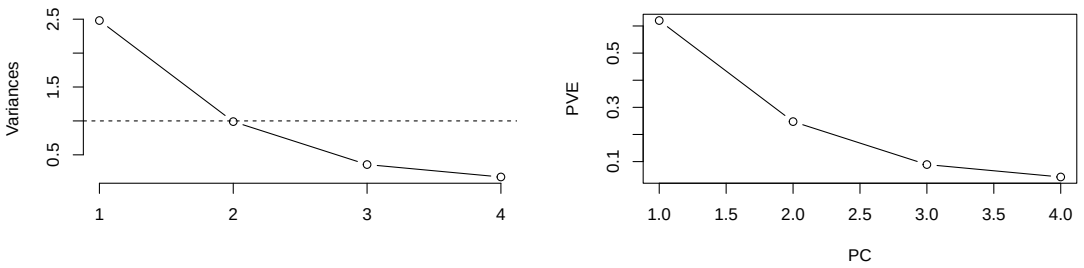
```
biplot(fit, scale = 0, cex = 0.7)  # scale=0 aby šípky zodpovedali záťažiam
```



- Vidíme, že všetky druhy zločinu majú v prvom PC približne rovnaké váhy, zatiaľčo podiel mešťanov iba polovičnú. To znamená, že PC1 približne zodpovedá celkovej miere závažných zločinov.

- Druhý PC naložil väčšinu záťaže na UrbanPop, takže predstavuje stupeň urbanizácie jednotlivých štátov.
- Premenné zodpovedajúce zločinom sú sústredené blízko pri sebe, to znamená, že sú navzájom značne korelované: štáty s vysokou kriminalitou jedného druhu majú problém aj s inými druhmi kriminality. S urbanizovanosťou súvisia menej.
- z grafu zásahov je vidieť, že najvyššiu kriminalitu mala Kalifornia, Nevada a Florida. Kalifornia má tiež vysoký podiel mestského obyvateľstva.
- Podľa suťového grafu by nám stačil prvý, nanajvýš aj s druhým hlavným komponentom.

```
screeplot(fit, type = "lines", main = "")
abline(h = 1, lty = 2)
plot(fit$sdev^2 / sum(fit$sdev^2), type = "b", ylab = "PVE", xlab = "PC")
```



- Prvý PC vysvetľuje asi 62% variability, druhý už iba 25%, spolu teda 87% celkovej variablity údajov.

12.2.5 Použitá literatúra

- James, Witten, Hastie, Tibshirani (2021): An Introduction to Statistical Learning - with Applications in R. 2nd ed. Springer. https://www.statlearning.com/
- Sanchez, G., Marzban, E. (2020) All Models Are Wrong: Concepts of Statistical Learning. https://allmodelsarewrong.github.io/pca.html

12.3 Zhluková analýza

12.3.1 Úvod

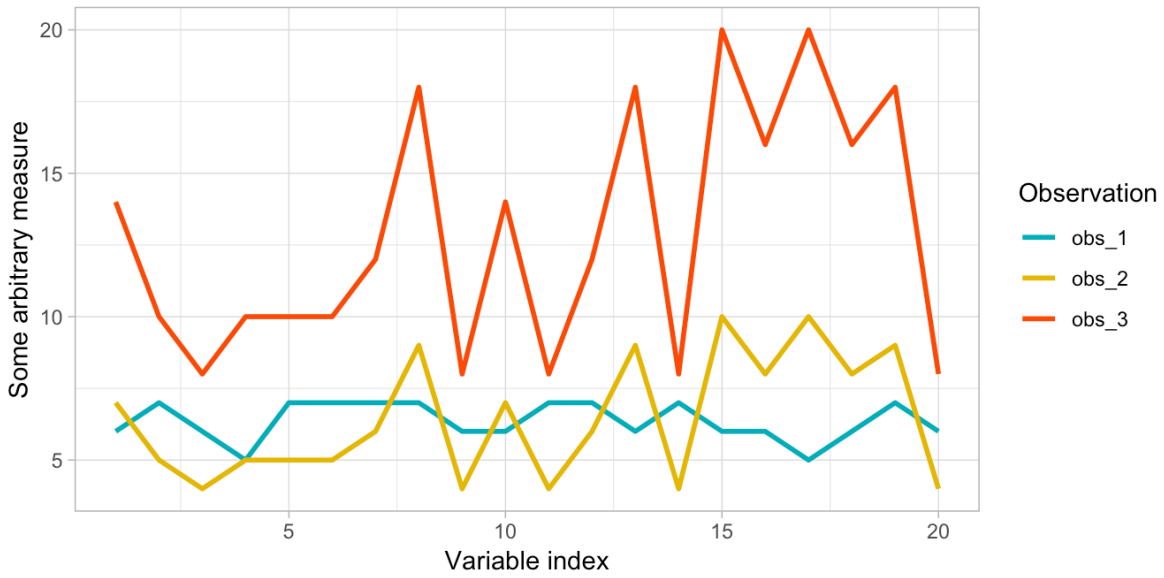
12.3.1.1 Princíp

- Zhlukovanie (angl. *clustering*) označuje širokú množinu metód neasistovaného učenia určených na hľadanie podskupín (zhlukov) v súbore údajov.
- Zvyčajne sa hľadajú *zhluky pozorovaní* (riadky tabuľky údajov), no môže sa použiť aj na hľadanie skupín medzi *vlastnosťami* (stĺpce).
- Objekty (štatistické jednotky - riadky, alebo vlastnosti - stĺpce) v skupine sú si navzájom "celkom podobné" (within-cluster homogeneity) zatiaľčo medzi skupinami sa "dosť líšia" (between-cluster heterogeneity).
- Napríklad
  - v marketingu je cieľom objaviť skupiny zákazníkov na cielenie konkrétnej reklamy,
  - v medicíne sa hľadajú skupiny pacientov vhodných pre špecifické liečebné postupy, resp. zhlukovaním sú predbežne diagnostikované varianty ochorenia.
  - v astronómii je záujem zoskupiť podobné hviezdy a galaxie,

– v genetike zas objaviť skupiny génov s podobným procesom expresie (prenosu informácie na RNA alebo proteíny).

12.3.1.2 Podobnosť a odlišnosť

- Aby sa zhluky dali od seba odlíšiť, je potrebné definovať mieru "podobnosti", resp. "odlišnosti".
- Niektoré miery vzdialenosti už poznáme (euklidovskú, manhattanskú, Mahalanobisovu,...), z nich napr. euklidovská je síce často používaná, ale je aj citlivá na odľahlé hodnoty.
- Pri numerických vlastnostiach daných v rôznej mierke sa hodí vzdialenosť založená na *korelácii*.

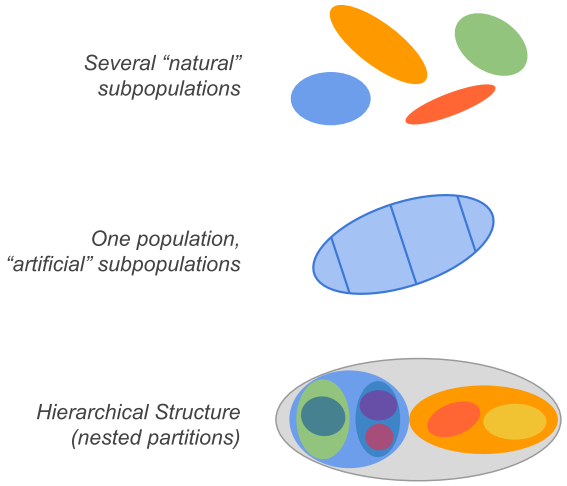


- V text mining-u sa bežne používa *kosínusová* vzdialenosť.
- *Gowerova* miera je vhodná pre zmiešané údaje (mixed data):
  - pre kvantitatívnu (intervalovú) premennú X je definovaná ako manhattanská miera normovaná rozsahom,
$$d_{i,j} = \frac{|x_i - x_j|}{\max_l(x_l) - min_l(x_l)}$$
  - ordinálna je najprv prekódovaná na poradia (s korekciou pre zhody),
  - nominálna premenná je prekódovaná na pomocné binárne premenné a z nich vypočítaný kockový (Sorensen-dice) koeficient podobnosti $S = \frac{2a}{2a+b}$, kde a je počet zhôd na hodnote 1, b je počet nezhôd (a zhody na 0 sa nerátajú). Miera vzdialenosti (nepodobnosti) je potom doplnok do 1. Celková vzdialenosť je priemerom čiastkových

vzdialeností zo všetkých uvažovaných vlastností (kvantitatívných a kvalitatívnych). Môže byť vážený pre korekciu neproporcionality medzi počtom jedničiek kvalitatívnych voči kvantitatívnym.

12.3.1.3 Metódy

- Zhlukovacie metódy sa delia podľa toho, či sú objekty do zhlukov zaradené výlučne (jednoznačná príslušnosť, *hard* methods) alebo s určitou mierou príslušnosti (*soft*, fuzzy methods).
- Bežnejšie sú metódy z prvej skupiny. Tam sa ďalej líšia podľa spôsobu separácie zhlukov na
  - metódy s **priamym** delením, napr. k-means alebo k-medoids clustering,
  - metódy s **hierarchickým** delením.



- *Hard* metódy používajú heuristické algoritmy, ktoré sa musia nejak vysporiadať s komplexnosťou problému zaradenia n objektov do K (výlučných) skupín.
  - Počet možných delení do K tried je tzv. Stirlingovo číslo druhého druhu $\left\{ {n \atop K} \right\} = \frac{1}{K!}\sum_{k=0}^{K}(-1)^k\binom{K}{k}(K-k)^n$,
  - a počet všetkých delení n objektov zas tzv. Bellovo číslo $B_n = \sum_{k=0}^{n}\left\{ {n \atop k} \right\} = \frac{1}{e}\sum_{i}^{\infty}\frac{i^n}{i!}$. Napríklad $B_{10} = 115975$, no už $B_{100} = 4.8 \times 10^{115}$ je viac, než počet atómov v pozorovateľnom vesmíre (cca 10^{80}).
- Zo *soft* metód sú obľúbené tie, ktoré modelujú zmes (normálnych) rozdelení, angl. Gaussian mixture models.

12.3.2 K-means

12.3.2.1 Princíp

- Jednoduchá a elegantná metóda.
- Počet tried K treba definovať vopred.

- Matematicky je problém formulovaný nasledovne: ak $C_1, \dots, C_K$ označujú množiny indexov objektov (pozorovaní) v každom zhluku, pričom

  – $C_1 \cup C_2 \cup \dots \cup C_K = \{1, \dots, n\}$,
  – $C_k \cap C_{k'} = \emptyset$ pre všetky $k \neq k'$,

  čiže každý objekt patrí práve do jedného zhluku, potom cieľom je minimalizovať vnútrozhlukový rozptyl W,

$$\underset{C_1,\dots,C_K}{minimize} \left\{ \sum_{k=1}^{K} W(C_k) \right\}, \qquad \text{kde} \quad W(C_k) = \frac{1}{n_k} \sum_{i \in C_k} \sum_{j=1}^{p} (x_{ij} - g_{kj})^2,$$

  $g_{kj} = \frac{1}{n_k} \sum_{i \in C_k} x_{ij}$ a použitá bola euklidovská vzdialenosť.

- Nájsť presné riešenie je veľmi náročné.
- Našťastie existuje jednoduchý algoritmus, ktorým sa dá nájsť celkom dobré riešenie, aj keď predstavuje iba lokálne optimum.
- S danými počiatočnými hodnotami centroidov $\mathbf{g}_1, \dots, \mathbf{g}_K$ sa opakujú nasledujúce kroky:

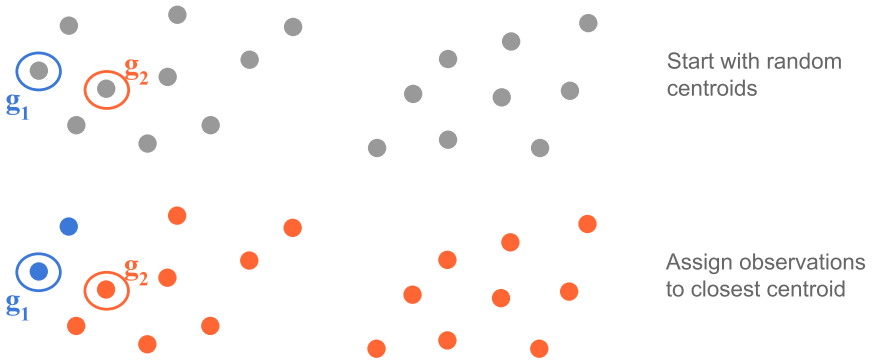
  1. zaradenie objektov do zhluku podľa najmenšej vzdialenosti od centroidu,
  2. prepočítanie centroidov,

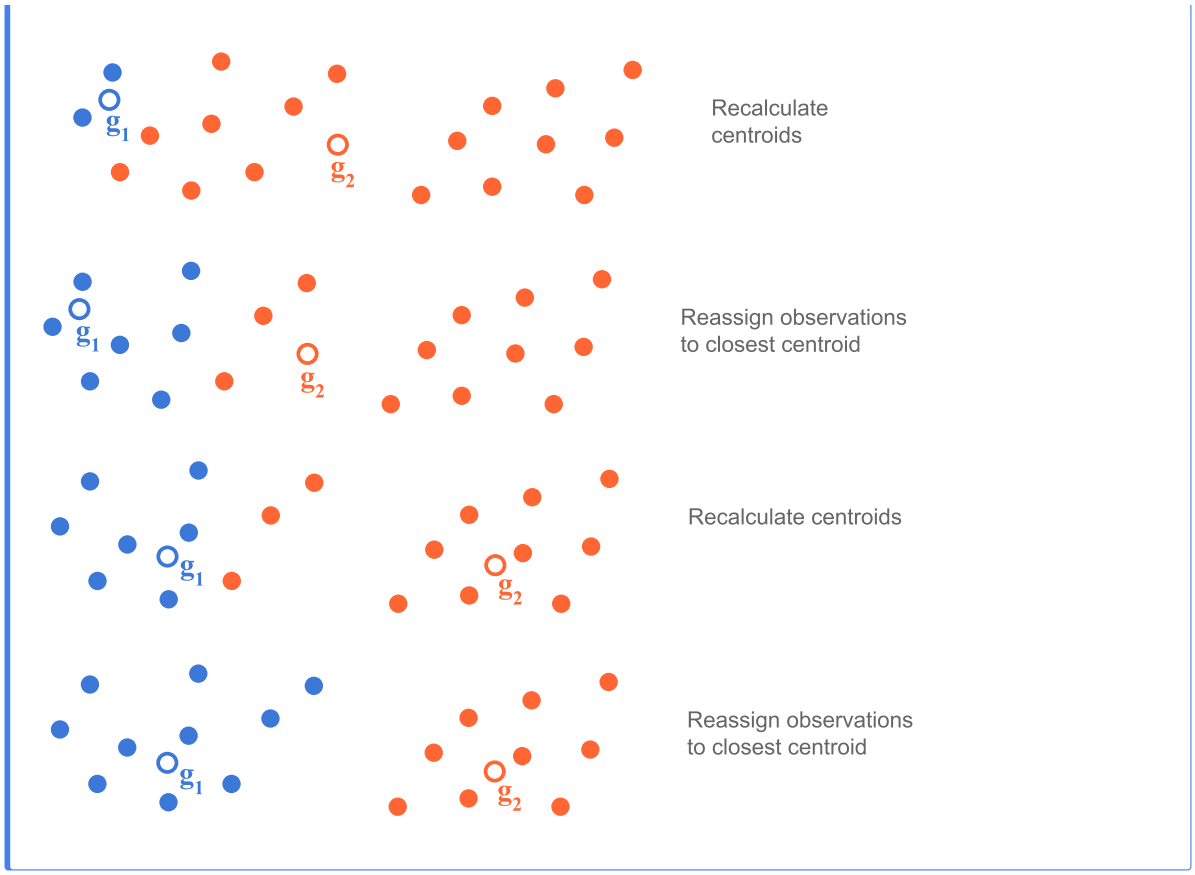
  až kým sa zaradenie v bode 1 nezmení.

- Počiatočné hodnoty sa určia buď

  – (Forgy-ho metódou) náhodným stotožnením centroidov s K objektami, alebo
  – náhodným zaradením objektov do K tried a výpočtom ich centroidov.

Ilustrácia (Forgy)



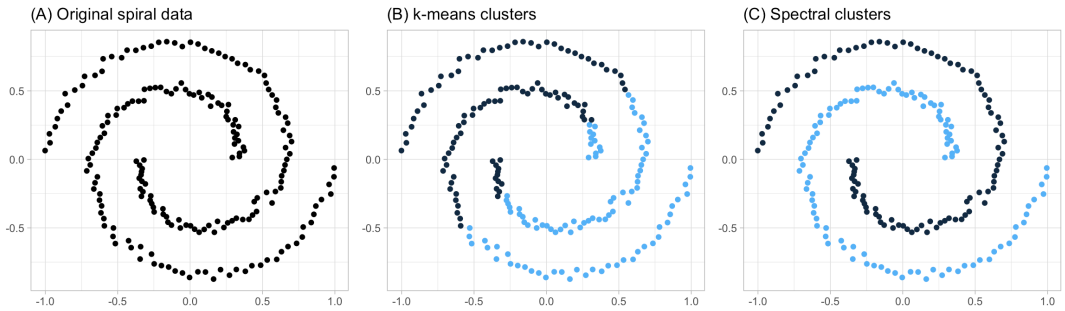


- Algoritmus vo všeobecnosti nájde skôr lokálne, než globálne minimum. Výsledok závisí od počiatočných (náhodných) podmienok.
- Preto je dôležité nechať odhad zbehnúť viackrát, vždy s inými počiatočnými hodnotami centroidov. Nakoniec sa vyberie najlepšie riešenie.

12.3.2.2 Alternatívy

- Ak dáta obsahujú odľahlé hodnoty, alternatívou ku k-priemerom môže byť metóda *k-medoidov* (partitioning around medians, *PAM*).
- Robustnosť je však kompenzovaná vyššou výpočtovou náročnosťou. V Rku sa použije funkcia *cluster::pam*
- Súradnice medoidov sú mediány jednotlivých vlastností, navyše sú vždy totožné s niektorými z pozorovaní.)
- S nárastom počtu objektov sa zhlukovanie spomaľuje. Preto sa pre veľké datasety oplatí použiť algoritmus (clustering large applications, *CLARA*), ktorý rozdelí dataset na viacero menších (rovnako veľkých) *vzoriek*. Na každej aplikuje PAM, získa medoidy ale

zhluky urobí *nad celým datasetom* a získa celkovú mieru vnútrozhlukovej vzdialenosti. Vyhrávajú medoidy (tej vzorky), pre ktoré je táto miera najmenšia.
- Algoritmus je implementovaný vo funkcii *cluster::clara*.
- Klasické zhlukovanie predpokladá konvexné hranice zhlukov. Komplikovanejšie štruktúry vyžadujú komplikovanejšie metódy, napr. *spektrálne* zhlukovanie. V Rku ich implementuje napr. balík *kernlab*.



Príklad (gejzír)

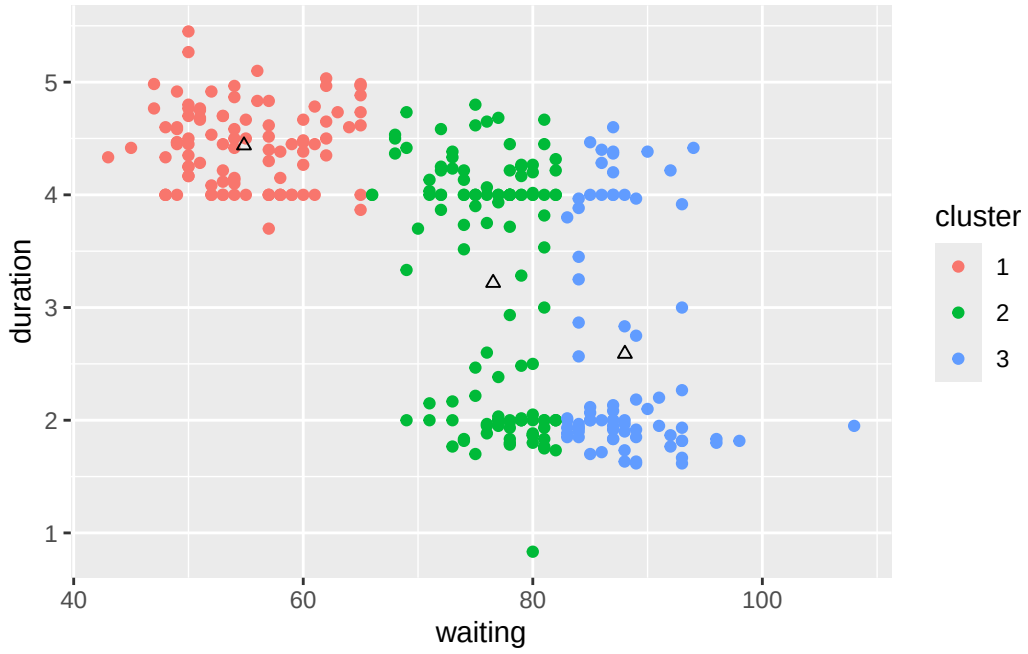
- Dataset *MASS::geyser* s dĺžkou trvania gejzírov (duration) a čakacou dobou na erupciu.

```
# skrátenie výpisu z objektu vráteného funkciou kmeans
print.kmeans <- function (x, ...)
{
    cat("K-means clustering with ", length(x$size), " clusters of sizes ",
        paste(x$size, collapse = ", "), "\n", sep = "")
    cat("\nCluster means:\n")
    print(x$centers, ...)
    cat("\nClustering vector:\n")
    print(head(x$cluster,10), ...)
    cat("...\n")
    cat("\nWithin cluster sum of squares by cluster:\n")
    print(x$withinss, ...)
    ratio <- sprintf(" (between_SS / total_SS = %5.1f %%)\n",
        100 * x$betweenss/x$totss)
    cat(sub(".", getOption("OutDec"), ratio, fixed = TRUE), "Available components:\n",
        sep = "\n")
    print(names(x))
    if (!is.null(x$ifault) && x$ifault == 2L)
        cat("Warning: did *not* converge in specified number of iterations\n")
    invisible(x)
}
```

```r
dat <- MASS::geyser
fit <- kmeans(dat, centers = 3, nstart =  10)
print(fit)  # modifikovaná metóda pre skrátenie niektorých dlhších výpisov
```

```
K-means clustering with 3 clusters of sizes 99, 123, 77

Cluster means:
   waiting duration
1 54.83838 4.438889
2 76.54472 3.218835
3 88.02597 2.589827

Clustering vector:
 1  2  3  4  5  6  7  8  9 10
 2  2  1  2  2  2  1  3  2  1
...

Within cluster sum of squares by cluster:
[1] 2819.925 2020.012 1444.276
 (between_SS / total_SS =  89.1 %)

Available components:

[1] "cluster"      "centers"      "totss"        "withinss"     "tot.withinss"
[6] "betweenss"    "size"         "iter"         "ifault"
```

```r
dat |>
  transform(cluster = as.factor(fit$cluster)) |>
  ggplot() + aes(x = waiting, y = duration) +
  geom_point(aes(color = cluster)) +
  geom_point(data = as.data.frame(fit$center), shape = 2)
```



- Zhluková analýza je zjavne ovplyvnená rozdielnym rozsahom časov, preto ju zopakujeme pre normované dáta.

```r
# inverse to function scale()
# - x is matrix or data frame to be unsclaed,
# - scaled is the output of scale()
unscale <- function(x, scaled = x) {
  par <- attributes(scaled)[c("scaled:center", "scaled:scale")] |>
    setNames(c("mean","sd"))
  n <- nrow(x)
  x * rep(par$sd, each=n) + rep(par$mean, each=n)
}

# clustering on scaled data
sdat <- MASS::geyser |>
  scale()
fit <- kmeans(sdat, centers = 3, nstart =  10)
# plot unscaled data
centroids <- fit$centers |>
  unscale(sdat) |>
  as.data.frame() |>
  tibble::rownames_to_column("cluster")
dat |>
  transform(cluster = as.factor(fit$cluster)) |>
  ggplot() + aes(x = waiting, y = duration) +
  geom_point(aes(color = cluster)) +
  geom_point(data = centroids, shape = 13, size = 3, show.legend = FALSE)
```

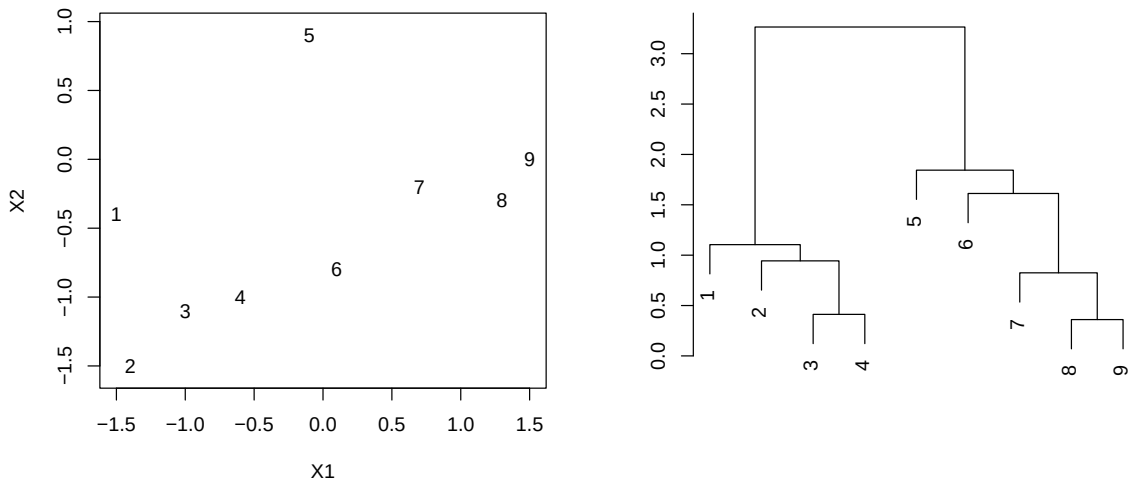
12.3.3 Hierarchické zhlukovanie

12.3.3.1 Dendrogram

- Nevýhodou metódy k-centroidov (k-means) je nutnosť špecifikovať počet zhlukov K vopred.
- Hierarchické zhlukovanie toto nevyžaduje, navyše výstupom je prehľadný stromový diagram, ktorý sa volá *dendrogram*.
- Každý *list predstavuje jeden objekt*, navzájom sú *spájané* vetvami v rôznej výške. Vertikálna súradnica spojov predstavuje mieru vzdialenosti (odlišnosti) skupín objektov.

```
dat1 <- data.frame(
  X1 = c(-1.5, -1.4, -1.0, -0.6, -0.1, 0.1, 0.7, 1.3, 1.5),
  X2 = c(-0.4, -1.5, -1.1, -1.0, 0.9, -0.8, -0.2, -0.3, 0.0)
  )

plot(X2 ~ X1, dat1, asp = 1, type = "n")
text(X2 ~ X1, dat1, labels = row.names(dat1))
fit <- dat1 |>
  dist() |>  # distance matrix calculation
```

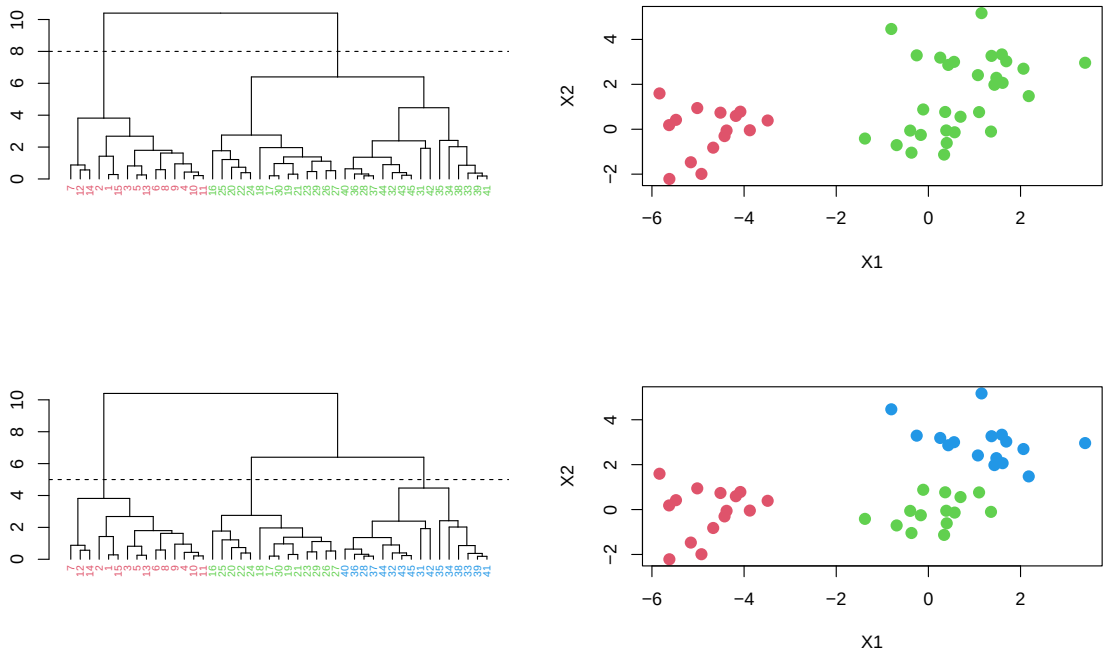
```
  hclust()  # hierarchical clustering
plot(fit, ann = FALSE)
```



- Odlišnosť dvoch objektov tak môžeme určiť podľa *výšky*, v ktorej sa ich vetvy spájajú. Pozor, nemôžeme súdiť na základe horizontálnej vzdialenosti!
- Identifikácia zhlukov v dendrograme prebieha pomocou *rezov* v konkrétnej výške.

```
# generate data
set.seed(1)
dat <- list(A = c(-5,0), B = c(0,0), C = c(1,3)) |>
  lapply(mvtnorm::rmvnorm, n = 15, sigma = diag(2)) |>
  lapply(as.data.frame) |>
  lapply(setNames, nm = paste0("X",1:2)) |>
  dplyr::bind_rows(.id = "class") |>
  dplyr::mutate(class = as.factor(class))

# fit model
fit <- dat |>
  dplyr::select(-class) |>
  dist() |>
  hclust()

# --- visualize result
```

```r
# basic dendrogram
#fit |>
#  plot(hang = -1, cex = 0.6, main="", ylab="")

# cut to get 2 clusters
fit |>
  as.dendrogram() |>
  dendextend::set("labels_col", 2:3, k=2) |>
  dendextend::set("labels_cex", 0.6) |>
  plot()
abline(h = 8, lty = 2)
# plot with dots or labels
plot(X2 ~ X1, dat,
     col = cutree(fit, k = 2) + 1,
     pch = 16, cex=1.5)
#plot(X2 ~ X1, dat, type = "n")
#text(X2 ~ X1, dat,
#     col = cutree(fit, k = 3) + 1,
#     labels = rownames(dat))


# cut to get 3 clusters
fit |>
  as.dendrogram() |>
#  dendextend::set("branches_k_col", 2:4, k=3) |>
  dendextend::set("labels_col", 2:4, k=3) |>
  dendextend::set("labels_cex", 0.6) |>
  plot()
abline(h = 5, lty = 2)
# plot with dots or labels
plot(X2 ~ X1, dat,
     col = cutree(fit, k = 3) + 1,
     pch = 16, cex=1.5)
#plot(X2 ~ X1, dat, type = "n")
#text(X2 ~ X1, dat,
#     col = cutree(fit, k = 3) + 1,
#     labels = rownames(dat))


# use package dendextend to color leaves and possibly branches
# https://talgalili.github.io/dendextend/articles/dendextend.html
# http://www.sthda.com/english/wiki/beautiful-dendrogram-visualizations-in-r-5-must-known-met
```

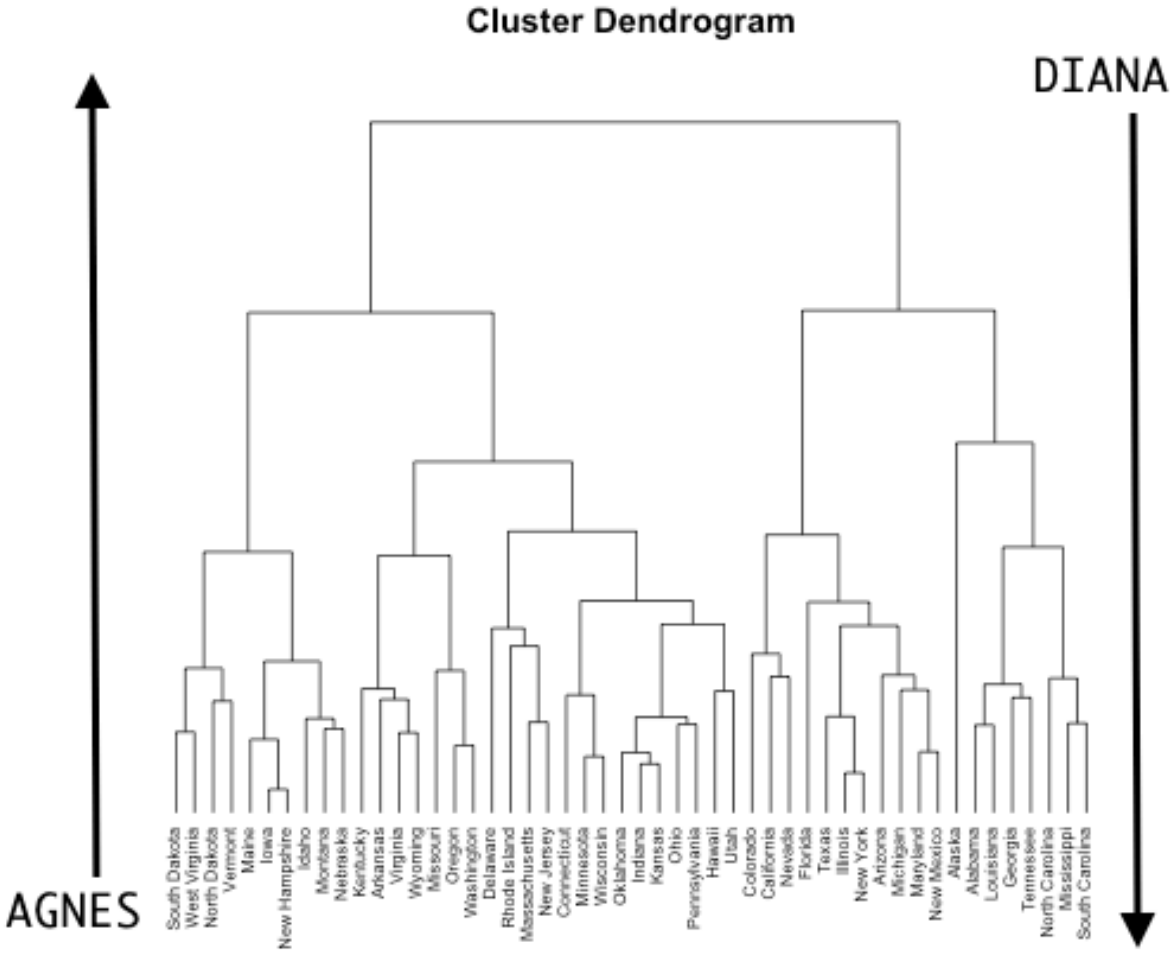


- Výška rezu tak slúži rovnakému účelu ako voľba počtu zhlukov v metóde k-means.
- Hierarchické zhlukovanie nemusí vždy zodpovedať realite. Napr. ak by naše pozorovania tvorili dve skupiny podľa pohlavia a tri skupiny podľa národnosti, potom nedáva zmysel, aby tretia skupina vznikla rozdelením jednej z tých dvoch. Vtedy môže hierarchické zhlukovanie dávať menej presné výsledky ako k-means.

12.3.3.2 Metódy

- Cieľom hierarchického zhlukovania je nájsť vnorené delenia objektov.
- Na to existujú dva prístupy:
  - zoskupujúci (*agglomerative*)
  - rozkladný (*divisive*)
- Aglomeratívna metóda *AGNES* (AGglomerative NESting) začína zdola:
  1. Každý objekt je na začiatku považovaný za samostatný zhluk (list).
  2. V každom kroku iteračného algoritmu sa spoja *dva najpodobnejšie zhluky*.
  3. Táto procedúra skončí, keď sú všetky objekty súčasťou jedného zhluku (koreň).
- Divizívna metóda *DIANA* (DIvisive ANAlysis clustering) začína z vrchu hierarchie (od koreňa). V každom kroku iteračného algoritmu je zhluk rozdelený na také dva menšie,

ktoré sú zo všetkých možných čo najviac heterogénne. Cyklus končí vtedy, keď každý zhluk obsahuje jediný objekt.



- Hierarchické zhlukovanie je implementované v dvoch predinštalovaných balíkoch Rka.
  - Aglomeratívne vo funkciách *stats::hclust* a *cluster::agnes*.
  - Divizívne v *cluster::diana*

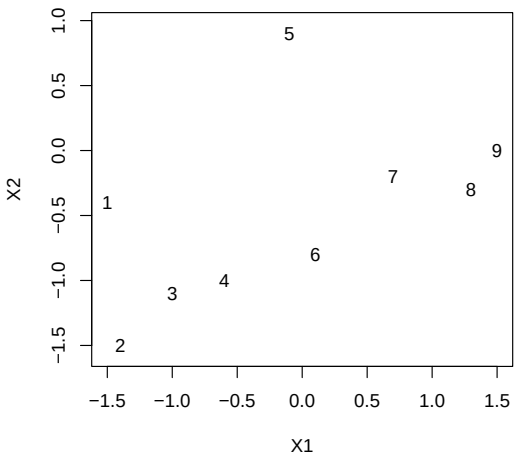
12.3.3.3 Vzdialenosť zhlukov

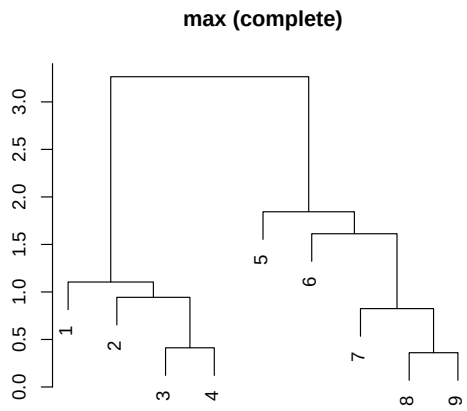
- Odlišnosť jednotlivých objektov už vieme merať, napr. euklidovskou alebo manhattanskou vzdialenosťou.
- Ale ako vyjadriť rozdielnosť dvoch zhlukov?
- Na to vzniklo niekoľko metód (*linkage*):

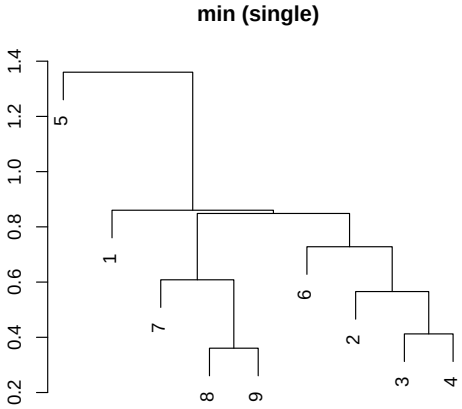
- maximum (*complete* linkage) - Najväčšia vzdialenosť medzi objektami z jedného a z druhého zhluku. Má tendenciu tvoriť zhluky rovnakého priemeru (diameter, najv.vzdialenosť), je citlivá na odľahlé hodnoty. Takže objekty môžu byť oveľa bližšie ku cudzím než ku vlastným.
- minimum (*single* linkage) - Najmenšia vzdialenosť medzi objektami z jedného a z druhého zhluku. Má tendenciu tvoriť reťaze (pridávať prvky po jednom alebo asociovať také zhluky, ktoré sú prepojené reťazou).
- priemer (*average* linkage) - Priemerná vzdialenosť medzi objektami z jedného a z druhého zhluku. Kompromis medzi predošlými. Avšak mení sa s monotónnou transformáciou vzdialenosti objektov, zatiaľčo max a min závisia len od poradia.
- centroid - Vzdialenosť medzi centroidmi. Môže viesť k inverziám, kedy dva zhluky sú spojené v menšej výške, než sú jednotlivo (problém s vizualizáciou a interpretáciou).
- Ward - Minimalizuje vnútro-zhlukový rozptyl a tak produkuje kompaktné zhluky.

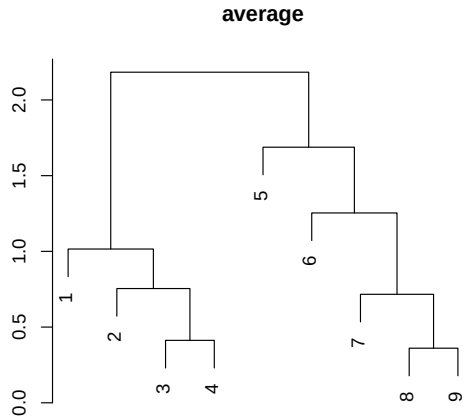
Ilustrácia (linkage)

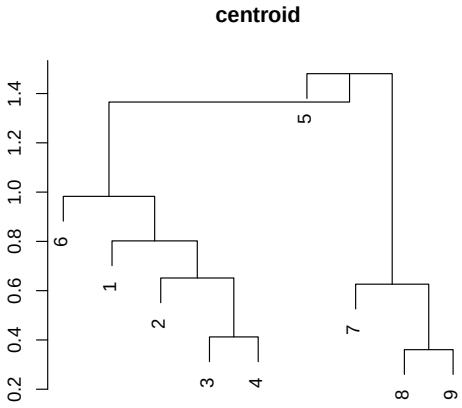
```
plot(X2 ~ X1, dat1, asp = 1, type = "n")
text(X2 ~ X1, dat1, labels = row.names(dat1))
dat1 |>
  dist() |>  # distance matrix calculation
  hclust(method = "complete")  |> # hierarchical clustering
  plot(xlab = "", ylab = "", main = "max (complete)", sub = "")
dat1 |>
  dist() |>  # distance matrix calculation
  hclust(method = "single")  |> # hierarchical clustering
  plot(xlab = "", ylab = "", main = "min (single)", sub = "")
dat1 |>
  dist() |>  # distance matrix calculation
  hclust(method = "average")  |> # hierarchical clustering
  plot(xlab = "", ylab = "", main = "average", sub = "")
dat1 |>
  dist() |>  # distance matrix calculation
  hclust(method = "centroid")  |> # hierarchical clustering
  plot(xlab = "", ylab = "", main = "centroid", sub = "")
dat1 |>
  dist() |>  # distance matrix calculation
  hclust(method = "ward.D")  |> # hierarchical clustering
  plot(xlab = "", ylab = "", main = "Ward", sub = "")
```

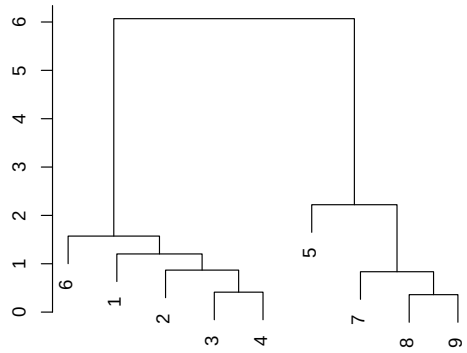

X1
X2
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
1
2
3
4
5
6
7
8
9


max (complete)
0.0
0.5
1.0
1.5
2.0
2.5
3.0
1
2
3
4
5
6
7
8
9


min (single)
0.2
0.4
0.6
0.8
1.0
1.2
1.4
5
1
7
8
9
6
2
3
4


average
0.0
0.5
1.0
1.5
2.0
1
2
3
4
5
6
7
8
9


centroid
0.2
0.4
0.6
0.8
1.0
1.2
1.4
5
6
1
2
3
4
7
8
9


Ward
0
1
2
3
4
5
6
6
1
2
3
4
5
7
8
9

- Funkcia *cluster::agnes* vypočíta aj hodnotu tzv. *aglomeračného koeficientu* (AC), ktorý v intervale [0,1] vyjadruje silu zhlukovej štruktúry. Vyvážené zhluky majú hodnotu blízke 1. Nevýhodou je, že rastie s n, takže sa nedajú porovnať dáta príliš rozdielnej dĺžky.

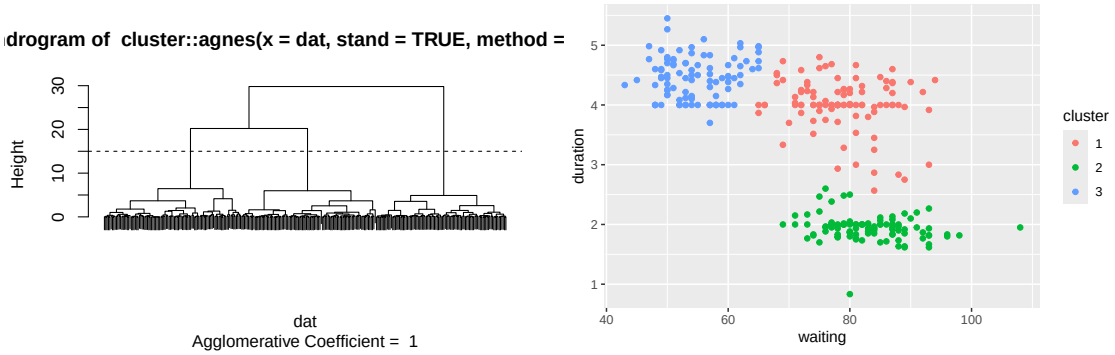
```
c("complete", "single", "average", "ward") |>
  sapply(function(x) cluster::agnes(dat1, method = x)$ac) |>
  round(2) |>
  rbind(AC = _)
```

```
   complete single average ward
AC     0.73   0.54    0.65  0.8
```

Príklad (Agnes)

- Dataset *MASS::geyser* s dĺžkou trvania gejzírov (duration) a čakacou dobou na erupciu.

```
dat <- MASS::geyser
fit <- dat |>
  cluster::agnes(stand = TRUE, method = "ward")
plot(fit, which = 2, labels = FALSE)
abline(h=15, lty=2)
#fit |> as.dendrogram() |> plot()

dat |>
  transform(
    cluster = fit |> cutree(k = 3) |> as.factor()
    ) |>
  ggplot() + aes(x = waiting, y = duration) +
  geom_point(aes(color = cluster))
```



12.3.4 Zmes normálnych rozdelení

12.3.4.1 Princíp

- Tradičné zhlukovacie metódy ako k-means a hierarchické sú heuristické. Zhluky tvoria priamo z dát, teda bez použitia nejakej pravdepodobnostnej miery.
- Metódy založené na modeli pravdepodobnosti poskytujú neostrú hranicu medzi zhlukmi (*soft assignment*), každý objekt má priradenú pravdepodobnosť príslušnosti ku každému zhluku.
- Tento prístup navyše rieši problém identifikácie optimálneho počtu zhlukov automaticky (implicitne).
- Najpopulárnejšia je trieda modelov so zmiešaným gaussovským rozdelením (Gaussian mixture model, GMM).
- GMM predpokladá, že každý objekt (jeho pozorované vlastnosti) pochádza z jedného z K (p-rozmerných) normálnych rozdelení s hustotou

$$f_k(\mathbf{x}) = \frac{1}{(2\pi)^{p/2}|\ _k|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x} -\ _k)^T\ _k^{-1}(\mathbf{x} -\ _k)\right)$$

  kde $_k$ je centroid a $_k$ kovariančná matica k-teho zhluku.
- Potom GMM je hustotou zmesi všetkých K normálnych rozdelení,

$$f(\mathbf{x}) = \sum_{k=1}^{K} \pi_k f_k(\mathbf{x})$$

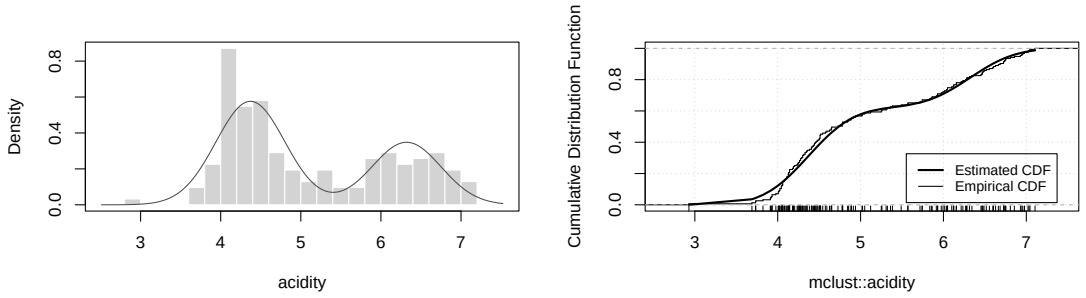
  kde $\{\pi_k\}$ sú váhy.
- Váhy spolu s centroidmi a kovariančnými maticami tvoria parametre GMM.

Príklad (kyslosť)

- Príklad: dataset *mclust::acidity*, rozdelenie pravdepodobnosti indexu kyslosti vody

v 155 jazerách v USA.

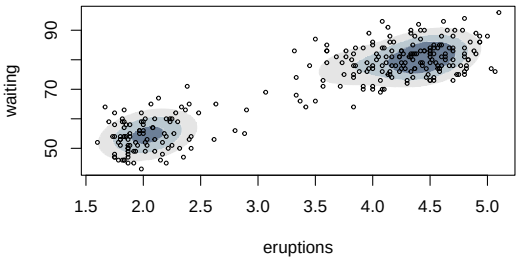
```
# 1D
fit <- mclust::acidity |>
  mclust::densityMclust(plot = FALSE)
fit |>
  plot(what = "density", data = mclust::acidity, breaks = 15, xlab = "acidity")
fit |>
  plot(what = "diagnostic", type = "cdf")
```

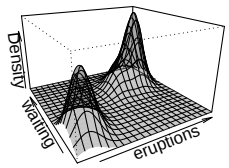


Príklad (gejzír)

- Príklad: *datasets::faithful* s dĺžkou trvania gejzírov (duration) a čakacou dobou na erupciu, gejzír Old Faithful v Yellowstonskom národnom parku.

```
# 2D
fit <- faithful |>
  mclust::densityMclust(plot = FALSE)
fit |>
  plot(what = "density", type = "hdr",
       data = faithful, points.cex = 0.5)
fit |>
  plot(what = "density", type = "persp")
```





- Zdá sa, že aj tieto gejzírové dáta vykazujú trimodalitu, jeden menší zhluk je však veľmi blízko druhého, väčšieho.

12.3.4.2 Typy kovariancií

- Rozkladom kovariančnej matice

$$_k = \lambda_k \mathbf{D}_k \mathbf{A}_k \mathbf{D}_k^\top$$

získame kontrolu nad geometriou zhlukov, konkrétne

  - *objemom* elipsoidu (volume), prostredníctvom skalárnej hodnoty λ_k,
  - *tvarom* (shape), cez diagonálnu maticu $\mathbf{A}_k$, pričom $|\mathbf{A}_k| = 1$,
  - *orientáciou*, pomocou ortogonálnej matice $\mathbf{D}_k$.

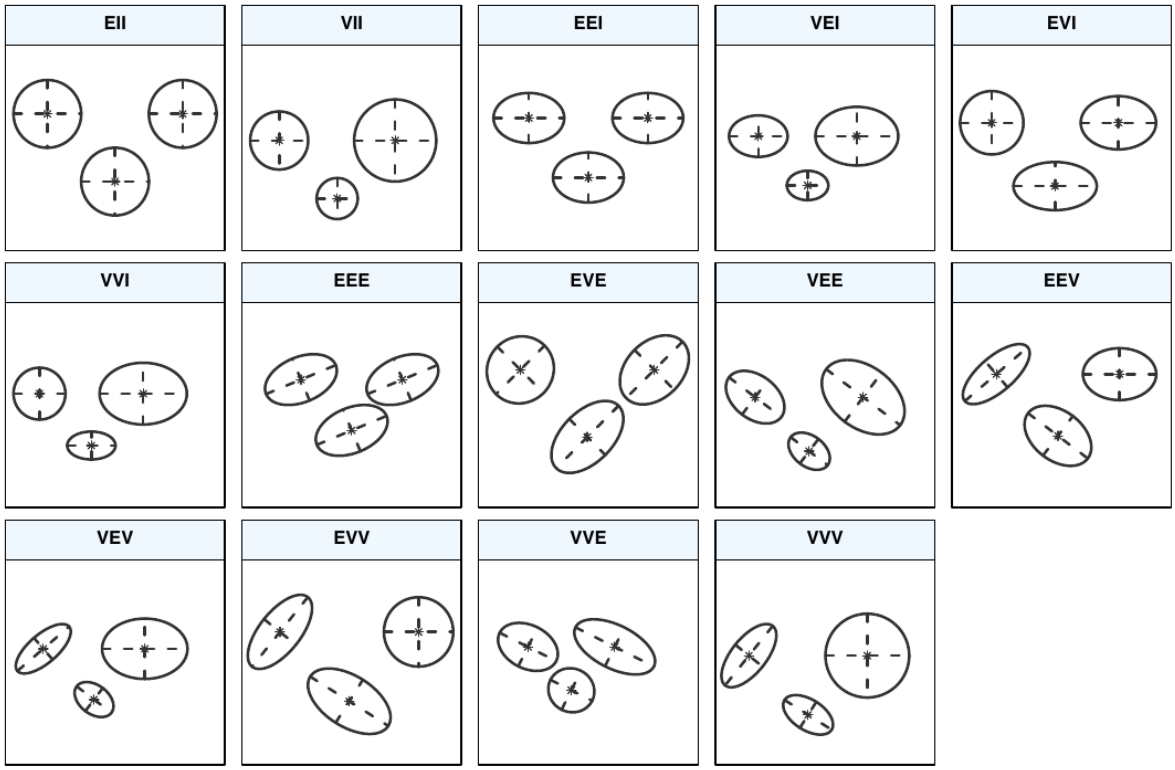
- Taxonómia kóduje parametrizáciu modelu trojpísmenovou skratkou v poradí podľa jednotlivých komponentov kovariančnej matice. Jednotlivé písmená potom značia:

  - E = equal (rovnaké),
  - I = identity matrix (jednotková matica),
  - V = variable (rôzne).

- V jednorozmernom prípade sú iba dva modely: E pre rovnaký rozptyl a V pre variabilný rozptyl.´
- Prehľad taxonómie modelov dáva nasledujúca tabuľka a po nej ilustrácie zodpovedajúcich elíps.

Model	$_k$	Distribution	Volume	Shape	Orientation
EII	$\lambda\mathbf{I}$	Spherical	Equal	Equal	—
VII	$\lambda_k\mathbf{I}$	Spherical	Variable	Equal	—

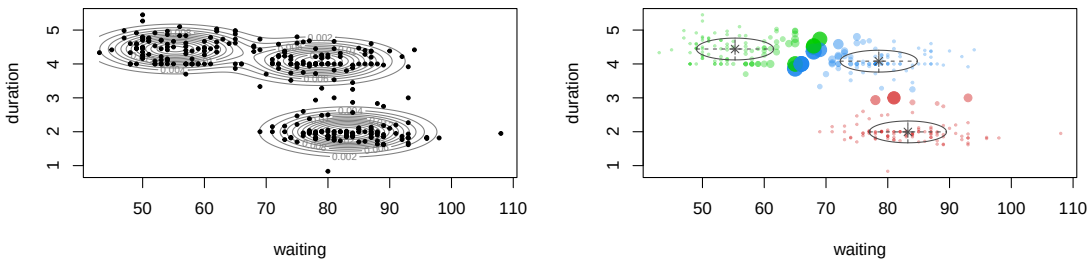
Model	$_k$	Distribution	Volume	Shape	Orientation
EEI	$\lambda\mathbf{A}$	Diagonal	Equal	Equal	Coordinate axes
VEI	$\lambda_k\mathbf{A}$	Diagonal	Variable	Equal	Coordinate axes
EVI	$\lambda\mathbf{A}_k$	Diagonal	Equal	Variable	Coordinate axes
VVI	$\lambda_k\mathbf{A}_k$	Diagonal	Variable	Variable	Coordinate axes
EEE	$\lambda\mathbf{DAD}^\top$	Ellipsoidal	Equal	Equal	Equal
EVE	$\lambda\mathbf{DA}_k\mathbf{D}^\top$	Ellipsoidal	Equal	Variable	Equal
VEE	$\lambda_k\mathbf{DAD}^\top$	Ellipsoidal	Variable	Equal	Equal
VVE	$\lambda_k\mathbf{DA}_k\mathbf{D}^\top$	Ellipsoidal	Variable	Variable	Equal
EEV	$\lambda\mathbf{D}_k\mathbf{AD}_k^\top$	Ellipsoidal	Equal	Equal	Variable
VEV	$\lambda_k\mathbf{D}_k\mathbf{AD}_k^\top$	Ellipsoidal	Variable	Equal	Variable
VVV	$\lambda_k\mathbf{D}_k\mathbf{A}_k\mathbf{D}_k^\top$	Ellipsoidal	Variable	Variable	Variable



Príklad (gejzír)

- Hustota zhlukov v datasete *MASS::geyser* má (v reze) eliptický tvar s približne rovnakým objemom, varianciou aj orientáciou (s osami).
- Model umožňuje zobraziť neistotu v zaradení jednotlivých objektov do zhlukov.

```
library(mclust)
fit <- MASS::geyser |>
  mclust::Mclust(G = 3)

plot(fit, what = "density")
points(MASS::geyser, pch=19, cex=0.5)
plot(fit, what = "uncertainty")
fit |> summary()
```




```
----------------------------------------------------
Gaussian finite mixture model fitted by EM algorithm
----------------------------------------------------

Mclust EEI (diagonal, equal volume and shape) model with 3 components:

 log-likelihood   n df      BIC       ICL
      -1371.823 299 10 -2800.65 -2814.577

Clustering table:
  1   2   3
 91 107 101
```

12.3.5 Použitá literatúra

- Boehmke, B., Greenwell, B. (2019). Hands-on machine learning with R. Chapman and Hall/CRC. https://bradleyboehmke.github.io/HOML/
- Hastie, T., Tibshirani, R., Friedman, J. H. (2009). The elements of statistical learning: data mining, inference, and prediction. Springer. https://hastie.su.domains/Papers/ESLII.pdf
- James, Witten, Hastie, Tibshirani (2021): An Introduction to Statistical Learning - with Applications in R. 2nd ed. Springer. https://www.statlearning.com/
- Sanchez, G., Marzban, E. (2020) All Models Are Wrong: Concepts of Statistical Learning. https://allmodelsarewrong.github.io/clustering.html
- Scrucca, L., Fop, M., Murphy, T. B., Raftery, A. E. (2016). mclust 5: clustering, classification and density estimation using Gaussian finite mixture models. The R journal, 8(1), 289. https://journal.r-project.org/archive/2016/RJ-2016-021/RJ-2016-021.pdf

References

Hastie, Trevor, Robert Tibshirani, a Jerome Friedman. 2009. *The Elements of Statistical Learning*. Springer New York. https://doi.org/10.1007/978-0-387-84858-7.

James, Gareth, Daniela Witten, Trevor Hastie, a Robert Tibshirani. 2021. *An Introduction to Statistical Learning*. Springer US. https://doi.org/10.1007/978-1-0716-1418-1.

Shalizi, Cosma Rohilla. 2021. *Advanced Data Analysis from an Elementary Point of View*. https://www.stat.cmu.edu/~cshalizi/ADAfaEPoV/.