FAKULTA INFORMAČNÍCH TECHNOLOGIÍ ČVUT V PRAZE

Zadání diplomové práce

Název:	Studijní podpora pro vizualizaci dat
Student:	Bc. Alžbeta Gogoláková
Vedoucí:	Ing. Magda Friedjungová, Ph.D.
Studijní program:	Informatika
Obor / specializace:	Znalostní inženýrství
Katedra:	Katedra aplikované matematiky
Platnost zadání:	do konce letního semestru 2022/2023

Pokyny pro vypracování

V bakalářském studiu specializace Umělá inteligence je nový předmět Vizualizace dat (BI-VIZ), pro který je třeba připravit studijní podporu s ohledem na předpokládané znalosti zapsaných studentů a na využití v oblasti strojového učení.

1. Proveďte rešerši výuky vizualizací dat se zaměřením na strojové učení na jiných univerzitách.
2. Analyzujte dostupné metody se zaměřením na různé typy dat (diskrétní, časové řady, geografické apod.). Analýzu diskutujte s vedoucím práce.
3. Na základě analýzy zvolte vhodné metody s ohledem na předpokládané znalosti studentů.
4. Navrhněte a implementujte několik vzorových studijních podpor (pravděpodobně Jupyter notebooky) demonstrujících zvolené metody na vhodných datech.
5. Navrhněte několik samostatných prací, ve kterých si studenti budou moci získané znalosti prakticky ověřit.

Elektronicky schválil/a Ing. Karel Klouda, Ph.D. dne 14. listopadu 2021 v Praze.

FAKULTA
INFORMAČNÍCH
TECHNOLOGIÍ
ČVUT V PRAZE

Diplomová práca

Študijná podpora pre vizualizáciu dát

Bc. Alžbeta Gogoláková

Katedra aplikovanej matematiky
Vedúci práce: Ing. Magda Friedjungová, Ph.D.

2. mája 2022

Poďakovanie

Predovšetkým by som chcela poďakovať Ing. Magde Friedjungovej, Ph.D. za vedenie práce, priateľský prístup a ochotu pomôcť s každým problémom, ktorý počas tvorby práce nastal. Taktiež ďakujem všetkým ľuďom, ktorí ma počas písania práce podporovali.

Prehlásenie

Prehlasujem, že som predloženú prácu vypracoval(a) samostatne a že som uviedol(uviedla) všetky informačné zdroje v súlade s Metodickým pokynom o etickej príprave vysokoškolských záverečných prác.

Beriem na vedomie, že sa na moju prácu vzťahujú práva a povinnosti vyplývajúce zo zákona č. 121/2000 Sb., autorského zákona, v znení neskorších predpisov, a skutočnosť, že České vysoké učení technické v Praze má právo na uzavrenie licenčnej zmluvy o použití tejto práce ako školského diela podľa § 60 odst. 1 autorského zákona.

V Prahe 2. mája 2022 . . . . . .

Odkaz na túto prácu

Gogoláková, Alžbeta. *Študijná podpora pre vizualizáciu dát.* Diplomová práca. Praha: České vysoké učení technické v Praze, Fakulta informačních technologií, 2022.

Abstrakt

Na Fakulte informačných technológií ČVUT v Prahe sa bude vyučovať nový predmet s názvom Vizualizácia dát. Táto práca popisuje proces tvorby samostatných prác a študijných materiálov spomínaného predmetu na základe rešerše výuky podobných predmetov na iných univerzitách a analýzy metód vizualizácie dát.

Kľúčové slová vizualizácia dát, strojové učenie, výuka, študijné materiály

Abstract

A new course called Data visualization will be taught at the Faculty of Information Technologies of the Czech Technical University in Prague. This thesis describes the process of creating study materials of the mentioned course, based on research of similar courses at other universities and analysis of data visualization methods.

Keywords data visualization, machine learning, teaching, study materials

Obsah

Zoznam obrázkov

Úvod

Vizualizácia dát je proces tvorby grafickej reprezentácie informácií, ktoré sú v daných dátach obsiahnuté. Grafy, siete, mapy či iné formy vizualizácie nám pomáhajú nájsť trendy a iné vzory, ktoré by sme bez nich neboli schopní detekovať. Čím viac dát máme k dispozícii, tým ťažšie je sa v nich vyznať. Vzhľadom k tomu, že objem produkovaných dát stále rastie, vizualizačné nástroje a technológie sú čoraz dôležitejšie. Vizualizácie sú veľmi nápomocné aj v oblasti strojového učenia. Je vďaka nim možné lepšie pochopiť dáta, s ktorými pracujeme, ale aj výsledné modely, ktoré sú na nich natrénované. Na Fakulte informačných technológií ČVUT v Prahe sa preto bude vyučovať nový predmet Vizualizácia dát (BI-VIZ) [1] zameraný na aplikáciu v strojovom učení. V tejto práci sa budem zaoberať procesom tvorby študijných materiálov a samostatných prác na spomínaný predmet.

Práca je rozdelená do šiestich kapitol. V prvej kapitole priblížim cieľ tejto práce a v druhej kapitole sa budem venovať rešerši výuky vizualizácie dát na iných českých aj zahraničných univerzitách. Výstupom rešerše je zoznam tematických celkov, ktoré sa budú preberať na predmete BI-VIZ. Tretia kapitola je hlavnou časťou tejto práce a obsahuje analýzu metód vizualizácie dát pre každý vybraný tematický celok. Venujem sa v nej všeobecným témam ako sú typy dát, vizuálne premenné či proces vizualizácie dát ale aj vhodným vizualizáciám pre konkrétne typy dát ako sú texty, obrázky, časové rady a grafy. V závere kapitoly sa zaoberám vizualizáciami, ktoré sa používajú v procese strojového učenia. Na túto analýzu naviažem v štvrtej kapitole, kde vyberiem vhodné metódy, ktoré sa budú preberať v rámci tematických celkov. V piatej kapitole sa venujem rôznym metódam vyučovania a predstavím nástroj na podporu aktivity študentov. Posledná kapitola zameraná na tvorbu študijných materiálov obsahuje analýzu nástrojov, ktoré sa budú v predmete využívať. Taktiež v nej popisujem vytvorené študijné materiály, samostatné práce a návody, ktoré som v rámci tejto práce vytvorila.

[1] https://bilakniha.cvut.cz/cs/predmet6614006.html

Kapitola 1

Cieľ práce

V bakalárskom štúdiu špecializácie Umelá inteligencia na FIT ČVUT v Prahe sa bude vyučovať nový predmet Vizualizácia dát (BI-VIZ). Cieľom tejto práce je vytvoriť niekoľko vzorových študijných materiálov a samostatných prác s ohľadom na predpokladané znalosti študentov.

Vo svojej práci sa budem zo začiatku venovať rešerši výuky vizualizácie dát na iných univerzitách. Na základe spomínanej rešerše a konzultácie s vedúcou práce určím tematické celky, ktoré sa budú v predmete BI-VIZ preberať. Následne spravím rešerš každého vybraného tematického celku a určím, ktoré metódy by sa mali zahrnúť do výuky predmetu BI-VIZ s ohľadom na predpokladané znalosti študentov. Na záver vytvorím niekoľko študijných materiálov demonštrujúcich zvolené metódy a taktiež navrhnem dve samostatné práce, na ktorých si študenti budú môcť prakticky overiť získané znalosti.

Kapitola 2

Rešerš obsahu výuky v predmetoch zameraných na vizualizáciu dát

Mojou úlohou bolo spraviť rešerš výuky vizualizácií dát so zameraním na strojové učenie na iných univerzitách. Bohužiaľ sa mi nepodarilo nájsť predmety iných univerzít, ktoré by sa priamo zameriavali na strojové učenie. Tento fakt bol konzultovaný s vedúcou práce a dohodli sme sa na vypracovaní rešerše nasledujúcich štyroch predmetov:

- Zpracování a vizualizace dat v prostředí Python, FIT VUT
- Vizualizace, FEL ČVUT
- Vizualizace, MUNI
- Visualization, Stanford University

Zpracování a vizualizace dat v prostředí Python na FIT VUT je predmet s praktickým zameraním. Prvé dve prednášky sa zaoberajú úvodom do jazyka Python. Predmet taktiež detailne preberá niektoré grafické balíčky (matplotlib a seaborn) a balíčky na prácu s dátami (pandas a numpy). Ďalšie prednášky sa venujú základom vizualizácie dát, procesu získavania dát, analýze dát a taktiež spôsobom vizualizácie obrázkov a časových radov. Cvičenia sú taktiež zamerané na praktické znalosti. Venujú sa jazyku Python, balíčkom numpy, pandas, matplotlib či seaborn a vizualizácii geografických dát a časových radov.

Predmet s názvom *Vizualizace* vyučovaný na FEL ČVUT je primárne zameraný na vizualizácie konkrétnych typov dát. Semester začína prednáškami na úvod do vizualizácie dát a pokračuje prednáškami o vizualizácii skalárnych, objemových, vektorových, n-dimenzionálnych a relačných dát nasledovaných

prednáškami o vizualizácii textu a časovo premenných dát. V závere semestra sa preberajú trendy v oblasti vizualizácie dát, užívateľské rozhranie a interakcie vo vizualizácii a jedna prednáška je venovaná aj strojovému učeniu.

Predmet vyučovaný na MUNI začína podobne ako ten vyučovaný na FEL ČVUT - prednáškami poskytujúcimi úvod do vizualizácie dát. Následne sa preberajú techniky vizualizácie priestorových, geografických a n-dimenzionálnych dát a pokračuje sa vizualizáciou stromov a sietí a textových dokumentov. Posledné štyri prednášky sa venujú interakciám, návrhom efektívnych vizualizácií, vizualizačným nástrojom a systémom a špecifickým aplikáciám vizualizácií (napr. vizualizácie v medicíne).

Predmet *Visualization* vyučovaný na Stanforde sa zameriava hlavne na teóriu vizualizácie dát. Detailne sa preberajú dôvody vizualizácie, typy dát, vizuálne premenné, vnímanie vizualizácií, vnímanie farieb a animácií, vizualizačný proces a efektívne využitie miesta. Ďalšie prednášky sa zaoberajú vizualizáciou a analýzou sietí, prirodzeným spracovaním jazyka a možnosťami vizualizácie textových dát a vizualizáciami v strojovom učení. V priebehu semestra mali študenti vypracovať tri samostatné práce. V prvej z nich mali pre zadaný dataset vytvoriť statickú vizualizáciu (jeden obrázok), ktorá odpovedá na nimi zvolenú otázku. Druhá úloha bola zameraná na exploračnú analýzu dát a v tretej úlohe mali študenti vytvoriť interaktívny vizualizačný systém vhodný na vizualizáciu veľkého množstva dát vytvorených pri inšpekcii reštaurácií.

2.1 Záver

Na základe zistených informácií som spolu s vedúcou práce určila, že sa v analýze metód vizualizácie dát budem zaoberať tematickými celkami, ktoré sa venujú úvodu do vizualizácie dát (vizualizačný proces, typy dát a vizuálne premenné, exploračná analýza dát) a vizualizácii špecifických typov dát (obrázky, texty, časové rady, grafy a siete). Vzhľadom k tomu, že je predmet určený pre špecializáciu Umelá inteligencia spravím aj analýzu vizualizácií používaných v strojovom učení a predstavím nástroj TensorBoard.

Kapitola 3

Analýza metód vizualizácie dát

3.1 Proces vizualizácie dát

Ktoré z tisícok súborov na našom počítači zaberajú najviac miesta? Ako sa 3,1 milióna nukleových báz v našej DNA líši od báz šimpanzov či myší? Vďaka aplikácii metód z oblastí počítačovej vedy, štatistiky, vyťažovania dát, grafického dizajnu a vizualizácie informácie môžeme na tieto otázky odpovedať pomocou vhodnej vizualizácie. Vďaka tomu dokážeme odpoveď v krátkom čase sprostredkovať aj ľuďom, ktorí sa v daných oblastiach nevyznajú. Benjamin Fry, expert na vizualizáciu dát, vo svojej knihe *Visualizing Data* [22] proces vizualizácie rozdeľuje do siedmich krokov a vysvetľuje ďalšie dôležité veci, na ktoré je potrebné pri vizualizácii dát myslieť. V tejto sekcii popíšem poznatky zo spomínanej knihy, ktoré vnímam ako dôležité.

3.1.1 Na akú otázku vizualizáciou odpovedáme?

V súčasnej dobe je veľmi jednoduché nazbierať a uložiť enormné množstvo dát. Často potom prichádzame do situácie, že do procesu vizualizácie dát vstupujeme s otázkou: „Ako môžeme pochopiť takýto objem dát?“ Takáto otázka je málo špecifická.

Pre pochopenie problému sa zamyslime nad mapami metra. Tieto mapy abstrahujú od komplexného tvaru mesta a sústredia sa na cieľ pasažiera - dostať sa z jedného miesta do druhého. Ukážka je na obrázku 3.1. Otázkou, na ktorej je postavená vizualizácia teda je: „Ako sa pasažier dostane z miesta A do miesta B?“ Správna vizualizácia sa preto nezameriava na nedôležité informácie ako sú zákruty či presné geografické lokácie ale zobrazuje len informácie, ktoré pasažierovi pomôžu zodpovedať jeho otázku.

Pokiaľ máme málo špecifickú otázku, nevieme na aké dáta sa sústrediť, aký graf zvoliť, ako vyhodnotiť kvalitu vizualizácie a podobne. Najdôležitejším krokom pri porozumení dát je teda identifikovať otázky, na ktoré sa pomocou nich snažíme odpovedať. Čím špecifickejšia otázka, tým jasnejšia bude konečná

Obr. 3.1: Orezaná mapa pražského metra [1]

vizualizácia. Vhodná otázka je taká, ktorá vie zaujať cieľového používateľa vizualizácie (tzn. človeka, ktorý si bude naše vizualizácie pozerať) a nie je príliš orientovaná na matematiku. Vizualizácia dát je ako každá iná komunikácia. Jej úspech je založený na tom, či cieľoví používatelia pochopia, čo sme zistili a budú nadchnutí našimi výsledkami.

Pokiaľ máme určenú otázku, vieme povedať aj čo je dobrá vizualizácia. Jedná sa o vizualizáciu, ktorá je naratívna a poskytuje jasnú odpoveď na určenú otázku bez toho, aby zobrazovala nedôležité detaily. Nedôležité detaily sú tie, ktoré nie sú podstatné na zodpovedanie otázky.

3.1.2 Postup pri vytváraní vizualizácie

Vytvorenie zmysluplnej vizualizácie si z dôvodu komplexnosti dát často žiada znalosti z viacerých odvetví. Proces pochopenia dát začína s množinou čísel či textov a otázkou, ktorú chceme zodpovedať. K odpovedi sa môžeme dostať pomocou siedmich krokov. Samozrejme nie je vhodné tieto kroky slepo nasledovať. V niektorých prípadoch budeme potrebovať všetkých sedem, v iných iba tri. Kroky sú nasledovné:

1. získanie dát (angl. acquire),
2. spracovanie dát (angl. parse),
3. filtrácia dát (angl. filter),
4. analýza dát (angl. mine),
5. výber vhodného grafu (angl. represent),
6. vylepšenie grafu (angl. refine),
7. pridanie interaktívnych prvkov (angl. interact).

Aj keď sú kroky usporiadané, neznamená to, že sa nemôžeme vracať do predchádzajúcich krokov. Môže sa stať, že po vytvorení vizualizácie uvidíme, že je presýtená informáciami. V taktom prípade sa môžme vrátiť do štvrtého kroku a aplikovať inú metódu filtrácie.

Prvý krok (*získanie dát*) v sebe zahŕňa všetky metódy, vďaka ktorým nadobudneme dáta. To môže byť jednoduché, pokiaľ máme dáta v nejakom súbore, ale aj zložité, pokiaľ sú rozložené medzi viacerými stránkami na internete. V tom druhom prípade hovoríme o hromadnom získaní dát webu a využívame metódy ako web crawling a web scraping, ktoré dokážu vďaka počiatočnej URL a špecifikácii dát, ktoré nás zaujímajú prechádzať prepojené URL adresy a sťahovať relevantné dáta.

V druhom a treťom kroku (*spracovanie dát, filtrácia dát*) sa pozeráme na dáta a snažíme sa v nich vytvoriť nejakú štruktúru, pochopiť ich význam, rozdeliť ich do kategórií a následne odstrániť tie, ktoré nie sú potrebné.

Počas štvrtého kroku (*analýza dát*) sa snažíme získať čo najviac informácií o dátach. Aplikujeme v ňom metódy štatistiky a vyťažovania znalostí z dát. Cieľom je nájsť v dátach zaujímavé vzory a dostať ich do matematického kontextu. Typickými úlohami v tomto kroku sú napríklad tvorba deskriptívnych štatistík, redukcia dimenzionality a podobne.

V piatom a šiestom kroku (*výber vhodného grafu, vylepšenie grafu*) začína vznikať výsledná vizualizácia. Začneme výberom grafu (napr. stĺpcový graf, koláčový graf, ...) a postupne ho vylepšujeme tak, aby bol čo najpochopiteľnejší a vizuálne atraktívny.

Posledný krok (*pridanie interaktívnych prvkov*) má zmysel aplikovať len v prípade, že cieľoví používatelia budú môcť s vizualizáciou interagovať. Jedná sa o pridanie interaktívnych elementov, vďaka ktorým je možné manipulovať s dátami a kontrolovať, ktoré informácie sú viditeľné.

3.1.3 Princípy

Benjamin Fry počas svojej kariéry pracoval na mnohých projektoch a dostal sa k trom princípom, ktoré mu pomáhali pri každej vizualizácii.

Prvým z nich je, že každý projekt je jedinečný. Pri niektorých projektoch budú stačiť grafy, ktoré vygeneruje nejaký software. Iné projekty môžu byť špecifickejšie a vtedy sa netreba báť vymyslieť úplne unikátnu vizualizáciu. Takéto vizualizácie je ale vhodné otestovať pomocou používateľského testovania. To je proces, počas ktorého vizualizáciu ukážeme nezainteresovaným ľuďom a oni popíšu, čo z nej pochopili.

Druhým princípom je niečo, čo som už v tejto sekcii spomínala. Vždy treba vizualizovať len to, čo je naozaj potrebné. To, že zobrazíme aj detaily, ktoré nie sú dôležité, môže zapríčiniť, že si používateľ neodnesie to, čo je naozaj podstatné. V horšom prípade používateľ vizualizáciu nepochopí vôbec, pretože bude príliš komplexná.

Kto vlastne je ten koncový používateľ? Aký je jeho cieľ? Čo sa snaží z vizualizácie zistiť? Akým spôsobom bude s vizualizáciou interagovať? Mať odpoveď na všetky tieto otázky je základom tretieho princípu - poznaj koncového používateľa. Tento poznatok by mal výrazne ovplyvniť výsledok našej práce. Aplikácia na zobrazenie máp na telefóne by mala fungovať úplne inak ako desktopová aplikácia, pretože každú z nich budú ľudia používať iným spôsobom. Vizualizácia pre deti na základnej škole by mala vyzerať inak ako vizualizácia pre vysokoškolákov.

3.2 Základné typy dát a ich vizualizácia

V tejto sekcii priblížim rôzne typy dát a vysvetlím aké sú medzi nimi rozdiely. Následne zhrniem, čo sú to značky a vizuálne premenné a ako je pomocou nich možné zobraziť zavedené typy dát. Ďalej uvediem dve kritériá, na základe ktorých je v niektorých prípadoch možné rozhodnúť, ktorá z dvoch vizualizácií je lepšia. V závere tejto sekcie popíšem rôzne praktiky manipulácie s vizualizáciou, ktorých aplikácia môže ovplyvniť mienku koncového používateľa. Takýmto praktikám je dobré sa vyhýbať.

3.2.1 Typy dát

Existuje viacero typov dát, medzi ktorými mnoho z nás intuitívne rozlišuje. Výška a šírka objektu majú spoločné vlastnosti ale napríklad typ oblečenia funguje na iných princípoch. V tejto sekcii dáta rozdelím na nominálne, ordinálne, intervalové a pomerové podľa delenia, ktoré zaviedol S. S. Stevens v článku *On the Theory of Scales of Measurement* [23]. V praxi sa často používajú pojmy kategorické a kvantitatívne dáta. Pojem kategorické dáta združuje ordinálne a nominálne dáta a pod pojmom kvantitatívne dáta sú združené intervalové a pomerové dáta.

Nominálne dáta

Nominálne dáta slúžia na priradenie kategórie. Môže sa jednať napríklad o čísla, znaky či reťazce, pričom jednotlivé hodnoty niekedy nazývame triedy. Dva príklady zdanlivo rôznych typov nominálnych dát, s ktorými sa môžeme stretnúť je a) číslo futbalistu a b) národnosť človeka. Číslo futbalistu v tíme je unikátne, zatiaľ čo môže existovať veľa ľudí s rovnakou národnosťou. Oba príklady sú ale založené na rovnakej podstate - nominálne dáta rozdeľuje entity do kategórií. Typ A je špeciálny prípad, v ktorom každá kategória obsahuje len jednu entitu.

Jedinou definovanou operáciou nad nominálnymi dátami je rovnosť (=). To znamená, že pre dve pozorovania vieme povedať, či sa ich hodnota nominálneho príznaku rovná alebo nie.

Ak by sme si povedali, že SVK bude znamenať české občianstvo a CZK slovenské občianstvo a zamenili tieto dve hodnoty v celom datasete, bude to síce mätúce, ale nič sa z pohľadu významu nezmení. Na nominálne dáta teda môžme aplikovať vzájomnú substitúciu dvoch hodnôt a dokonca aj permutáciu všetkých hodnôt a stále si budú správne plniť svoju funkciu. Môžme teda o nich povedať, že sú z matematického pohľadu permutačnou grupou.

Jediná štatistika relevantná pre nominálne dáta typu A je počet kategórií. Akonáhle niektoré kategórie obsahujú viac ako jednu entitu (typ B), je možné určiť najčastejšie sa vyskytujúcu hodnotu (módus) a skonštruovať kontingenčnú tabuľku. Tieto štatistiky (a všetky ďalšie štatistiky, ktoré budú uvedené v tejto sekcii) detailnejšie rozoberám v sekcii 3.3 o exploračnej analýze dát.

Ordinálne dáta

Ordinálne dáta slúžia podobne ako nominálne dáta na priradenie kategórie. Rozdiel je v tom, že ordinálne kategórie sú usporiadané. Ako príklad uvediem ohodnotenie vedomostí študenta po skúške z predmetu. Často sa hodnotí pomocou šiestich kategórií - A (výborne), B (veľmi dobre), C (dobre), D (uspokojivo), E (dostatočne) a F (nedostatočne). Ďalším príkladom môže byť energetická náročnosť spotrebičov či najvyššie dosiahnuté vzdelanie človeka.

Z usporiadanosti kategórií vyplýva, že okrem operácie rovnosti ($=$) je na ordinálnych dátach definovaná aj operácia porovnania ($<, \leq, \geq, >$).

Ordinálne kategórie sa vždy dajú namapovať na numerické dáta. Tento proces sa často používa pri metódach strojového učenia, keďže niektoré algoritmy vedia pracovať iba s numerickými dátami. Ak máme ordinálne dáta v numerickej podobe, môžeme ich transformovať pomocou ľubovoľnej rastúcej funkcie a ich význam zostane rovnaký. Vzhľadom k tomu, že sa pojem rastúca funkcia používa v dvoch rôznych kontextoch, pridávam aj presnú definíciu:

Definícia 1. Funkcia f je *rastúca* práve vtedy, keď pre každé x_1, x_2 z definičného oboru funkcie platí:

$$x_1 < x_2 \implies f(x_1) < f(x_2).$$

Vzhľadom k tomu, že je zadefinované len poradie hodnôt a nie vzdialenosti medzi nimi, nemá zmysel počítať štatistiky ako sú stredná hodnota a smerodatná odchylka. Okrem štatistík, ktoré je možné aplikovať na nominálne dáta ale môžeme navyše vypočítať medián a kvantily.

Intervalové dáta

Intervalové dáta už radíme medzi kvantitatívne. Neslúžia na rozdelenie do kategórií ale na vyjadrenie miery nejakej vlastnosti entity. Význam intervalových dát zostane zachovaný aj po transformácii pomocou funkcie

$$x' = ax + b.$$

Na intervalové dáta môžeme aplikovať všetky zvyčajné štatistiky, ktoré nevyžadujú znalosť skutočnej nuly (napríklad stredná hodnota, smerodatná odchýlka, rozptyl, ...). Pozícia nuly v intervalových dátach nie je pevne daná. Niekedy môže byť daná konvenciou, ale to nevyvracia predchádzajúce tvrdenie. Dobrým príkladom sú často používané teplotné stupnice Fahrenheit a Celsius. Obe stupnice sa používajú na vyjadrenie tej istej veličiny - teploty. Každá stupnica má konvenciou danú pozíciu nuly. Medzi stupnicami môžme prevádzať hodnoty pomocou transformácie v tvare $x' = ax + b$.

Intervalové dáta majú definovaný rozdiel (-) medzi dvoma hodnotami. Taktiež je možné vypočítať pomer dvoch rozdielov (intervalov, preto sa nazývajú intervalové dáta). Samozrejme sú definované aj všetky operácie, ktoré sú definované na ordinálnych dátach.

Dátum môže slúžiť ako ďalší príklad intervalových dát. Pozícia nuly je z konvencie daná časom, kedy sa začal letopočet. Nevieme priamo vydeliť dva dátumy, alebo povedať, že jeden je dvojnásobkom druhého. Vieme ale vytvoriť časový interval pomocou rozdielu dvoch dátumov. Dva časové intervaly už je možné vydeliť.

Pomerové dáta

Pomerové dáta majú zadefinované všetky operácie ostatných skupín dát - rovnosť, porovnanie, rozdiel a pomer rozdielov. Okrem toho je definovaný aj pomer dvoch hodnôt. Pozícia nuly je pevne daná a medzi dvoma stupnicami je možné prevádzať len pomocou násobenia:

$$x' = ax.$$

Medzi pomerové dáta patrí napríklad dĺžka. Môžme ju merať napríklad v centimetroch či palcoch a platí:

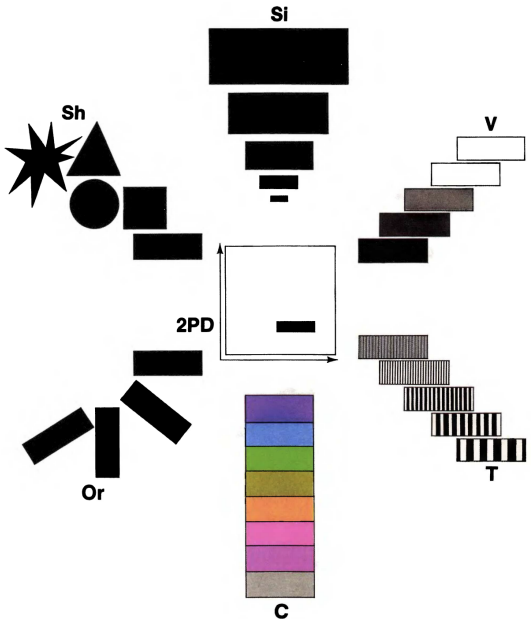
- 0 cm = 0 in (pozícia nuly je pevne daná)
- 1 cm $\doteq$ 0,3937 in (prevádzame len pomocou násobenia)

3.2.2 Vizuálne premenné

Rôzne typy dát si vyžadujú rôzne štýly vizualizácie. Touto problematikou sa zaoberal Jacques Bertin v jeho knihe *Semiology of Graphics* [2], z ktorej budem v tejto sekcii čerpať. Bertin nerozlišuje medzi intervalovými a pomerovými dátami a nazýva ich dokopy pojmom kvantitatívne dáta.

Pri voľbe správnej vizualizácie nejakého typu dát musíme myslieť na jeho vlastnosti. Pokiaľ chceme vizualizovať ordinálne dáta, potrebujeme zvoliť vizualizáciu, z ktorej je jasne vidieť poradie. Vizualizácia je tvorená značkami a vizuálnymi premennými.

Značky sú geometrické primitíva. Pri 2D vizualizácii (tzn. vizualizácii v jednej rovine) sa jedná o body, čiary a oblasti. Bod slúži na reprezentáciu



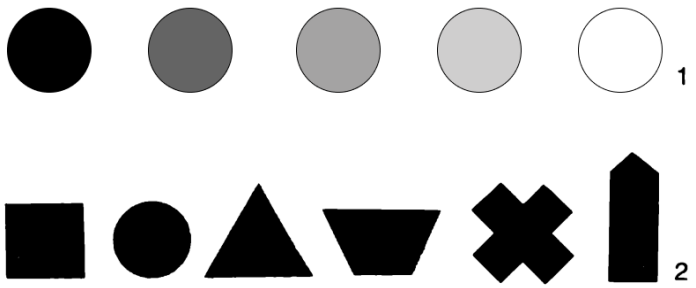
Obr. 3.2: Vizuálne premenné: Si - veľkosť, V - hodnota, T - textúra, C - farba, Or - orientácia, Sh - tvar, 2PD - súradnice na vyjadrenie polohy [2]

pozície v rovine, nenesie v sebe informáciu o dĺžke, či obsahu. Čiara slúži na reprezentáciu merateľnej dĺžky ale nenesie v sebe informáciu o obsahu. Nie je teda dôležité, akú hrúbku čiary zvolíme, ale záleží len na jej dĺžke. Oblasť sa používa na vizualizáciu merateľnej veľkosti a rozloženia.

Značky môžme v rovine umiestniť pomocou dvoch súradníc. Ďalšie vlastnosti, ktoré môžeme meniť sú:

- veľkosť,
- hodnota,
- textúra,
- farba,
- orientácia,
- tvar.

Pri tvorbe vizualizácie teda máme 8 premenných, na základe ktorých vieme zobraziť rôzne vlastnosti dát. Tieto premenné sa nazývajú vizuálne premenné a ich ukážka je na obrázku 3.2. Nie vždy môžme na nejakú značku aplikovať všetky vizuálne premenné. Napríklad nie je vhodné, aby sme menili pozíciu bodu, pretože pozícia je pre bod vždy daná pevne. Taktiež nie je možné zmeniť



Obr. 3.3: Porovnanie hodnoty (pôsobí usporiadane) a tvaru (nepôsobí usporiadane) [2]

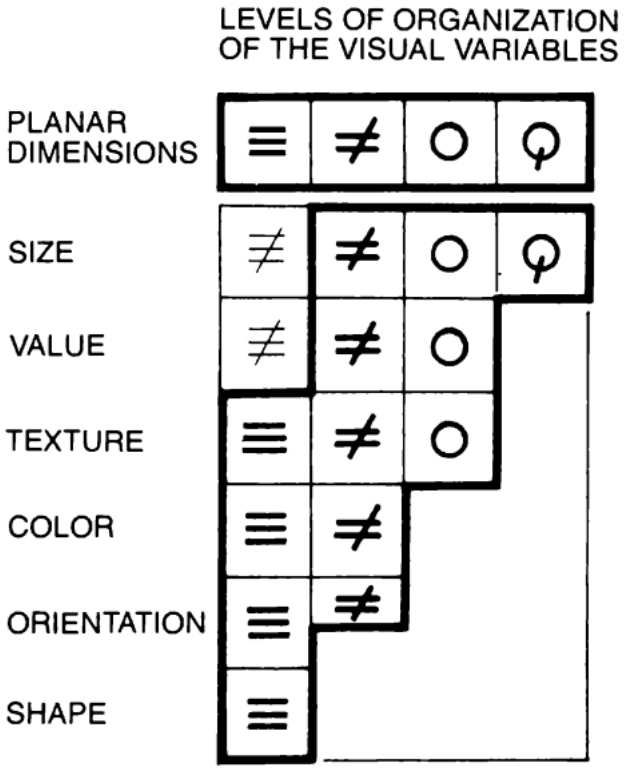
veľkosť, tvar a orientáciu nejakej oblasti bez toho, aby sa zmenil jej vnímaný zmysel.

Keď chceme použiť vizuálne premenné na vizualizáciu nejakého typu dát, je potrebné, aby vedeli zobraziť všetky vlastnosti daných dát. Zamyslime sa preto nad tým, aké vlastnosti majú rôzne typy dát.

Nominálne dáta nie sú usporiadané. To značí, že pri vizualizácii môžeme ľubovoľne meniť ich poradie. Z toho vyplýva, že na vizualizáciu takýchto dát nie je vhodné použiť vizuálnu premennú, ktorá pôsobí usporiadane (napr. hodnota). Druhou vlastnosťou je, že všetky hodnoty majú rovnakú dôležitosť. Nemali by sme ich zobrazovať pomocou premenných, ktoré nejakú kategóriu zobrazujú viac viditeľne ako inú (napr. veľkosť, hodnota). Podľa otázky, na ktorú sa snažíme vizualizáciou odpovedať môžeme k nominálnym dátam pristúpiť dvoma spôsobmi - selektívnym a asociatívnym. Selektívny prístup je založený na vnímaní rozdielu medzi rôznymi hodnotami kategórie a asociatívny prístup umožňuje vnímať rôzne hodnoty dokopy ako celok. V týchto prípadoch sa treba zamyslieť a zvoliť vhodnú vizuálnu premennú, ktorá implikuje selektivitu či asociativitu.

Ordinálne dáta sú naopak usporiadané, a preto by vždy mali byť vizualizované v správnom poradí. Taktiež je nutné ich vizualizovať pomocou vizuálnych premenných, ktoré pôsobia usporiadane. Na obrázku 3.3 je vidieť, že hodnta pôsobí usporiadane (je možné ju zoradiť buď sprava doľava alebo zľava doprava), zatiaľ čo tvar usporiadaný nie je. Tvar teda nie je vhodný na zobrazenie ordinálnych dát. Podobne ako nominálne dáta, aj ordinálne kategórie majú rovnakú dôležitosť. Z toho vyplýva, že pre ne platia rovnaké pravidlá, aké som spomínala vyššie.

Kvantitatívne dáta je vhodné zobrazovať pomocou vizuálnych premenných, na ktorých je vidieť vzdialenosť a pomer medzi dvoma hodnotami. Často sú vizualizované vzhľadom k enumeračným jednotkám. Enumeračná jednotka je taká, pomocou ktorej sa združujú dáta. Môže to byť napríklad časový interval



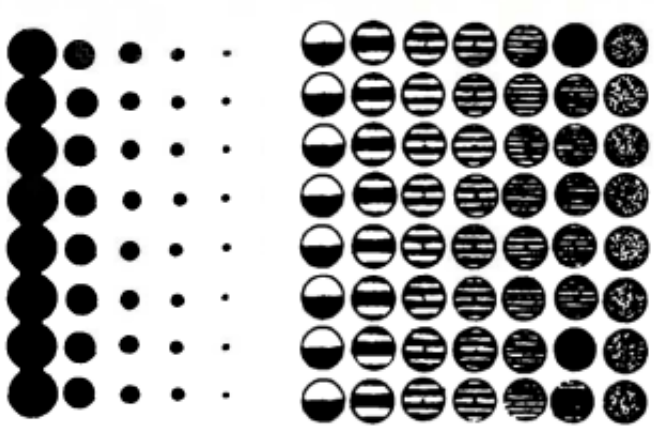
Obr. 3.4: Vlastnosti vizuálnych premenných [2]

(počet obetí moru za rôzne obdobia) alebo geografická plocha (HDP krajín). Často sa stáva, že enumeračné jednotky nie sú ekvivalentné (krajiny majú rôznu rozlohu, časové intervaly sú rôzne dlhé). Z tohto dôvodu je potrebné pri vizualizácii rozlišovať medzi kvantitami, ktoré sú závislé na enumeračnej jednotke (počet obyvateľov mesta), a ktoré sú nezávislé na enumeračnej jednotke (úmrtnosť v meste). Pri závislých kvantitách je vhodné pred vizualizáciou vykonať úpravu dát, ktorá ich spraví nezávislými. To sa dá dosiahnuť vydelením enumeračnou jednotkou. Pokiaľ by sme chceli vizualizovať počet obyvateľov mesta, je vhodné počet obyvateľov vydeliť rozlohou mesta a vizualizovať hustotu obyvateľstva, ktorá už je nezávislá na enumeračnej jednotke.

Každý typ dát má iné vlastnosti a je vhodné ho zobraziť pomocou iných vizuálnych premenných. Vizuálna premenná, ktorá sa použije na vizualizáciu dát musí mať dostačujúce vlastnosti na zobrazenie danej informácie. Zhrnutie všetkých vlastností vizuálnych premenných je na obrázku 3.4.

Definícia 2. Vizuálna premenná je *selektívna* ($\neq$), pokiaľ umožňuje okamžitú identifikáciu všetkých výskytov objektov s rovnakou kategóriou.

Príkladom selektívnej premennej môže byť farba. Povedzme, že má tri



Obr. 3.5: Porovnanie veľkosti (disociatívna premenná) a textúry (asociatívna premenná) [2]

kategórie - červená, žltá a zelená. Z vizualizácie vieme okamžite identifikovať všetky výskyty červenej farby a náš mozog ich vníma ako jednu skupinu, odlišnú od skupín vizualizovaných žltou či zelenou farbou.

Definícia 3. Vizuálna premenná je *asociatívna* ($\equiv$), pokiaľ umožňuje okamžité združenie objektov s rôznymi hodnotami danej premennej pomocou využitia inej vizuálnej premennej. Premenná, ktorá nie je asociatívna sa nazýva *disociatívna* ($\not\equiv$).

Ako príklad je možné uviesť tvar. Napriek tomu, že majú objekty rozličný tvar, je stále možné ich vnímať ako jeden celok, pokiaľ majú rovnakú veľkosť a farbu. Veľkosť je disociatívna, pretože kruhy troch rôznych veľkostí nepôsobia ako jeden celok, ale ako tri samostatné kategórie. Ukážka rozdielu medzi asociatívnou a disociatívnou premennou je na obrázku 3.5.

Definícia 4. Vizuálna premenná je *usporiadaná* (O), pokiaľ jej kategórie evokujú univerzálne poradie.

Šedá je univerzálne vnímaná ako farba medzi bielou a čiernou. Stredná veľkosť je medzi malou a veľkou. Hodnota a veľkosť sú preto usporiadané. Štvorce, kruhy a trojuholníky neevokujú žiadne poradie. Z toho vyplýva, že tvar nie je usporiadaný.

Definícia 5. Vizuálna premenná je *kvantitatívna* (Q), pokiaľ je pomocou nej možné vnímať vzdialenosť a pomer.

Typickým príkladom kvantitatívnej premennej je veľkosť. Pri porovnaní dvoch čiar rôznej dĺžky vieme od oka povedať, že jedna z nich je dvakrát dlhšia.

3.2.3 Evaluácia vizualizácie

Na základe informácií z predchádzajúcich sekcií a niekoľkých ďalších pravidiel (napr. vždy je potrebné pomenovať osi, vizualizácia má mať nadpis, pokiaľ to pomôže, treba pridať legendu, ...) už je možné vytvoriť dobrú vizualizáciu. Dobrá vizualizácia je z estetického aj matematického hľadiska korektná vizualizácia, ktorá nepotrebuje ďalší komentár k jej pochopeniu.

Na zodpovedanie každej otázky existuje niekoľko dobrých vizualizácií, ktoré spĺňajú všetky pravidlá. Ako vieme vyhodnotiť, ktorá z nich je najlepšia? Úplne jednoznačne sa to niekedy povedať nedá, ale môžu k tomu pomôcť dve kritériá, ktoré vo svojej dizertačnej práci [24] navrhol Jock Mackinlay. Jedná sa o efektivitu a expresivitu.

Expresivita

Expresivita je kritérium, ktoré skúma ako dobre vizualizácia odpovedá na položenú otázku. Dajú sa z nej získať všetky potrebné informácie? Nie je v nej naopak nejaká nepotrebná informácia, ktorá berie pozornosť?

Definícia 6. Súbor faktov je *vyjadriteľný* (angl. expressible) vizuálnym jazykom, pokiaľ daný jazyk obsahuje vety (tj. vizualizácie), ktoré:

1. zobrazujú všetky fakty súboru,

2. zobrazujú len fakty obsiahnuté v súbore.

Veta, ktorá spĺňa uvedené podmienky je expresívna vzhľadom k danému súboru faktov.

Keď chceme porovnať dve vizualizácie z hľadiska expresivity, musíme začať pri otázke, na ktorú odpovedajú. Najskôr overíme, či obe obsahujú všetky potrebné informácie a potom zistíme, či jedna z nich nemá naviac informácie, ktoré nie sú dôležité. Pokiaľ sú obe vizualizácie expresívne, môže nám pomôcť ďalšie kritérium.

Efektivita

Na rozdiel od expresivity, závislej len na schopnosti vizualizácie správne zobraziť potrebné informácie je efektivita závislá aj na jej koncovom používateľovi.

Definícia 7. Vizualizácia je *efektívnejšia* ako iná vizualizácia, pokiaľ sú ňou prenášané informácie vnímané jednoduchšie ako informácie prenášané druhou vizualizáciou.

3.2.4 Chybné a zavádzajúce vizualizácie

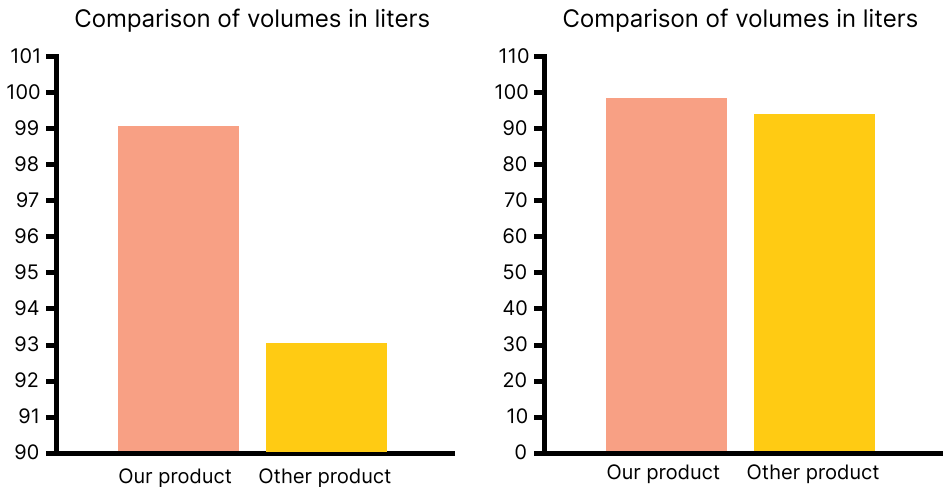
V súčasnej dobe sa môžeme čoraz častejšie stretnúť s dátovým žurnalizmom. Jedná sa o typ žurnalizmu, ktorý je založený na analýze a filtrovaní dát za účelom vytvorenia spravodajského príbehu. Problémom v tejto oblasti je, že niektorí žurnalisti sú nedostatočne informovaní o problematike vizualizácií, alebo cielene zavádzajú. V oboch prípadoch sú výsledkom takéhoto žurnalizmu zavádzajúce, alebo ťažko pochopiteľné vizualizácie. Čitateľ, ktorý nie je pozorný, alebo problematike nerozumie, si potom môže odniesť nepravdivé informácie. Vzhľadom k tomu, že sú momentálne médiá plné dezinformácií, ktoré polarizujú spoločnosť a predstavujú hrozbu pre fungovanie demokratických krajín, je dobré vedieť o častých chybách vo vizualizáciách a dávať si na ne pozor. V tejto sekcii popíšem niektoré z nich.

Základné pravidlo, ktoré sa dá aplikovať takmer na všetky vizualizácie sa vzťahuje k osám. Každú os treba popísať. Pokiaľ to nie je z kontextu jasné, je potrebné pridať aj jednotky, v ktorých bola daná veličina meraná. Príklad, ktorý porušuje toto pravidlo je v prílohe B.1. Ide o graf, ktorý použila spoločnosť Apple pri predstavení M1 čipu. Graf je síce správne popísaný, ale na osách nie sú merítka a nevieme z neho určiť v akých jednotkách sú dáta merané.

Veľmi často je možné vidieť stĺpcové grafy, ktoré porovnávajú dve alebo viacero hodnôt pričom os zobrazujúca hodnotu nezačína od nuly. Potom sa môže mylne zdať, že nejaká hodnota je napríklad dva krát taká veľká ako iná hodnota aj keď reálny rozdiel nie je taký veľký. Ukážka takéhoto grafu je na obrázku 3.6. Niekedy sú ale rozdiely v hodnotách také malé, že bez priblíženia by sme videli dva rovnako vysoké stĺpce. V takýchto prípadoch je samozrejme možné nezačínať od nuly, ale je nutné o tom koncového používateľa informovať. Rovnaký princíp platí pri transformácii osí (napr. logaritmovanie, umocňovanie).

Pri vytváraní vizualizácie, na ktorej os x zobrazuje čas je dobré vyskúšať viacero časových intervalov. V praxi často nastáva prípad, že sa vizualizácia zameria na neprimerane malé obdobie, z ktorého nie je vidieť celkový trend dát. Ako príklad uvediem vizualizáciu, ktorá by sa snažila ukázať, že globálne otepľovanie je skutočné pomocou priemernej dennej teploty nameranej v Prahe od 1. 1. 2022 do 1. 6. 2022. Dáta by určite mali rastúci trend, ale ten by nebol spôsobený globálnym otepľovaním ale jednoduchou zmenou ročných období. Globálne otepľovanie je samozrejme skutočný problém a keby sme ho chceli vizualizovať, bolo by vhodné zobraziť dlhodobý vývoj teploty (napr. od začiatku priemyselnej revolúcie).

Niekedy sa môžeme v médiách stretnúť s grafmi, ktoré sú matematicky nekorektné. Môže sa jednať o rôzne chyby pri prepise dát. Príkladom môže byť koláčový graf, ktorého časti dokopy nedávajú 100%. Aj matematicky korektné grafy môžu byť zavádzajúce. Niekedy stačí, keď je nesprávne zvolený nadpis. V prílohe B.2 je graf s veľkým nadpisom *časté zranenia detí* a pod ním malý



Obr. 3.6: Stĺpcový graf s priblíženou osou y (vľavo), ktorý je mätúci a jeho oddialená verzia (vpravo)

popis *zranenia, kvôli ktorým sú deti hospitalizované*. Človek, ktorý len rýchlo preletí novinami, v ktorých je táto vizualizácia sa môže mylne domnievať, že viac ako 5% detí si zraní chrbticu. Vizualizáca ale hovorí, že viac ako 5% detí zo všetkých hospitalizovaných detí, je hospitalizovaných kvôli zraneniu chrbtice.

Ďalšou častou chybou je vytváranie 3D vizualizácií, keď to vôbec nie je potrebné. 3D pohľad totiž môže vizualizáciu skresliť. Oblasti, ktoré sú vpredu pôsobia väčšie ako by reálne mali byť a naopak oblasti vzadu pôsobia menšie. Príklad zavádzajúceho 3D koláčového grafu a jeho presnejšia 2D verzia, je v prílohe B.3.

3.3 Exploračná analýza dát

Analýza dát je proces systematického aplikovania štatistických či logických techník za cieľom popísania, ilustrácie a vyhodnotenia dát. Existujú rôzne metódy, vďaka ktorým je možné rozlíšiť v dátach signál od šumu (štatistických fluktácií), či objaviť zaujímavé vlastnosti dát [25]. Dva prístupy k analýze dát sú:

- exploračná analýza dát
  - angl. exploratory data analysis, skrátene EDA
- konfirmačná analýza dát
  - angl. confirmatory data analysis, skrátene CDA

EDA je neformálna metóda, ktorá slúži na zoznámenie sa s datasetom. Môžeme vďaka nej zistiť aká je všeobecná štruktúra dát, získať deskriptívne štatistiky a nápady na vlastnosti dát, ktoré by bolo vhodné overiť sofistikovanejšou analýzou. CDA je naopak metóda, pomocou ktorej vieme napr. produkovať bodové a intervalové odhady, či overovať pravdivosť hypotéz na základe štatistických testov (napr. t-test).

Pokiaľ pomocou EDA neodhalíme nejakú zaujímavú vlastnosť dát, pravdepodobne nebudeme mať dôvod aplikovať CDA. Nie je teda vhodné slepo aplikovať CDA bez exploračnej fázy. Rovnako nevhodné je usudzovať závery len na základe EDA. Veľa zistení, ktoré vyplynú z EDA, môže byť zavádzajúcich.

V tejto sekcii sa budem zaoberať exploračnou analýzou dát. Zameriam sa na metódy, ktoré zaviedol John W. Tukey v jeho knihe *Exploratory Data Analysis* [26]. Táto kniha je mojim primárnym zdrojom, informácie z iných zdrojov budem citovať explicitne.

3.3.1 Čistenie a kategorizácia dát

Proces EDA začína pochopením, validáciou a čistením dát. Pre každý príznak datasetu určíme, či je kategorický alebo spojitý a pozrieme sa na jeho hodnoty vzhľadom k jeho rozsahu, či ostatným príznakom za účelom identifikácie chybných dát. Príklad prekročenia rozsahu je príznak *mesiac* s hodnotou väčšou ako 12, či menšou ako 1. Predstavme si, že máme dataset udalostí s príznakmi *dátum začiatku* a *dátum konca*. Pokiaľ je *dátum začiatku* neskorší ako *dátum konca*, jedná sa o chybu v dátach vzhľadom k ostatným príznakom. Potom prichádza fáza čistenia dát. Čistenie dát v sebe zahŕňa identifikáciu chýbajúcich hodnôt a zvolenie stratégie na ich doplnenie, konverziu dát (napr. všetky dátumy by mali byť v rovnakom formáte), alebo úpravu chýb, ktoré sme našli pri validácii.

3.3.2 Univariačná deskriptívna štatistika

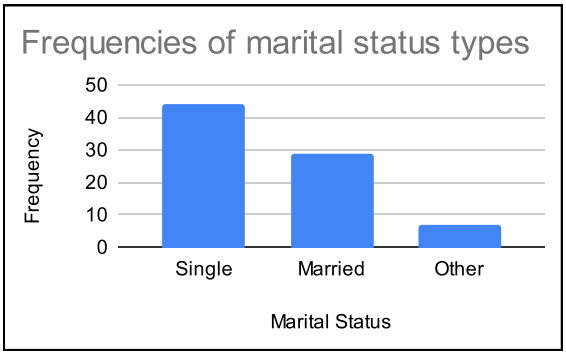
Jedná sa o štatistický popis jedného príznaku. Vlastnosti, ktorými príznak popíšeme a typy grafov, ktoré použijeme závisia na type príznaku.

Kategorický príznak

Štatistiky, ktoré slúžia na popísanie jedného kategorického príznaku môžu byť napríklad frekvencia, relatívna frekvencia či percento výskytu každej hodnoty. Všetky tri údaje poskytujú tú istú informáciu - aké časté sú rôzne hodnoty príznaku.

Tieto štatistiky je možné vizualizovať pomocou frekvenčnej tabuľky alebo stĺpcového či koláčového grafu (odporúčané len v prípade binárneho príznaku). Frekvenčná tabuľka a stĺpcový graf sú ukázané na obrázku 3.7.

Marial Status		
	Frequency	Percent
Single	44	55
Married	29	36.25
Other	7	8.75
Total	80	100



Obr. 3.7: Frekvenčná tabuľka (vľavo) a stĺpcový graf (vpravo)

Spojitý príznak

Spojitý príznak sa väčšinou popisuje podľa dvoch typov štatistík. Sú to:

- miery centrálnej tendencie,
- miery disperzie.

Miery centrálnej tendencie

Miery centrálnej tendencie popisujú príznak pomocou priemernej, centrálnej či typickej hodnoty. Najčastejšie používaná miera centrálnej tendencie je stredná hodnota. Tá sa vypočíta pomocou aritmetického priemeru všetkých hodnôt príznaku (pozorovaní):

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i = \frac{x_1 + x_2 + \cdots + x_n}{n}.$$

Stredná hodnota je vhodná na popísanie veličín, ktoré sa blížia k normálnemu rozdeleniu a neobsahujú outlierov. Pokiaľ dáta obsahujú outlierov, je možné použiť vážený aritmentický priemer

$$\bar{x} = \sum_{i=1}^{n} x_i w_i,$$

kde w_i je váha pre i-te pozorovanie [27]. Váhy sú určené na základe dvoch podmienok:

- $\sum_{i=1}^{n} w_i = 1$,
- hodnoty na okraji rozdelenia majú nízku váhu, zatiaľ čo hodnoty v strede rozdelenia majú vysokú váhu.

Ďalšia možnosť spočíva v odstránení odľahlých hodnôt. Orezaná stredná hodnota (angl. trimmed mean) sa vypočíta tak, že sa odstráni niekoľko percent najväčších aj najmenších hodnôt a stredná hodnota sa vypočíta zo zvyšku. Niekedy sa namiesto orezanej strednej hodnoty používa winsorizovaná stredná hodnota (angl. winsorized mean). Pri jej výpočte sa zoberie niekoľko percent najmenších hodnôt a nahradia sa najväčšou z nich. Následne sa vyberie rovnaký počet najväčších hodnôt a nahradia sa najmenšou z nich. Winsorizovaná stredná hodnota sa vypočíta z takto upravených hodnôt. Z toho vyplýva, že ide o špeciálny prípad váženého aritmetického priemeru.

Medián je ďalšou mierou centrálnej tendencie. Určí sa tak, že sa hodnoty zoradia od najmenšej po najväčšiu a vyberie sa hodnota presne v strede. Pre nepárny počet hodnôt existuje práve jedna, ktorá je v strede. Pre párny počet sú v strede dve hodnoty a medián dostaneme pomocou ich aritmetického priemeru.

Poslednou mierou centrality, ktorú spomeniem je módus. Módus reprezentuje najčastejšie sa vyskytujúcu hodnotu a je možné ho aplikovať aj na kategorické dáta. Pokiaľ existuje viac módusov, nerobíme ich aritmetický priemer ale uvedieme všetky.

Miery disperzie

Miery disperzie popisujú ako veľmi homogénny či heterogénny je daný príznak. Najjednoduchším príkladom je rozsah, ktorý získame rozdielom najväčšej a najmenšej hodnoty:

$$r_x = x_{max} - x_{min}.$$

Ďalšou mierou je kvantil, ktorý rozdeľuje zoradené pozorovania zo vzorky do skupín s rovnakou pravdepodobnosťou.

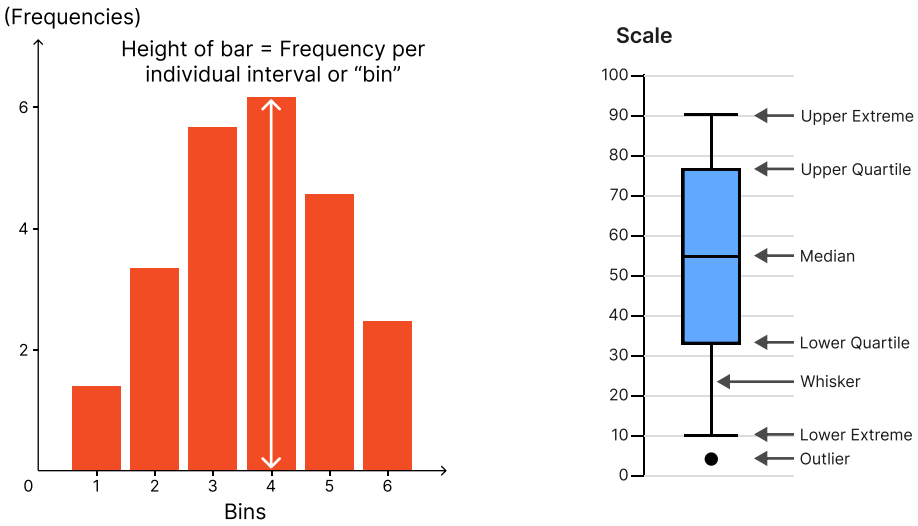
Definícia 8. Nech X je náhodná veličina s distribučnou funkciou F_X a $\alpha \in (0, 1)$. Q_α je *α-kvantil náhodnej veličiny X* práve vtedy, keď

$$q_\alpha = inf\{x|F_X(x) \geq \alpha\}.[28]$$

Pokiaľ chceme pozorovania rozdeliť na 100 skupín, použijeme špeciálny kvantil, ktorý sa nazýva percentil. Ak povieme, že študent na teste dosiahol percentil 96 ($Q_{0,96}$), znamená to, že mal lepšie výsledky ako 96% študentov. Na rovnakom princípe fungujú kvantily, ktoré delia vzorku dát na štvrtiny. Takéto kvantily sa nazývajú kvartily. Dolný kvartil je označovaný $Q_{0,25}$, medián $Q_{0,5}$ a horný kvartil $Q_{0,75}$. Niekedy sa udáva aj rozdiel dolného a horného kvartilu označovaný ako medzikvartilový rozsah (angl. interquartile range, skrátene IQR).

Miery disperzie, ktoré sa často udávajú spolu so strednou hodnotou sú smerodatná odchýlka a rozpyl. Vzorec pre výpočet rozptylu pre vzorku X o veľkosti n je:

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



Obr. 3.8: Histogram (vľavo) a box plot (vpravo) [3]

Pre smerodatnú odchýlku potom platí:

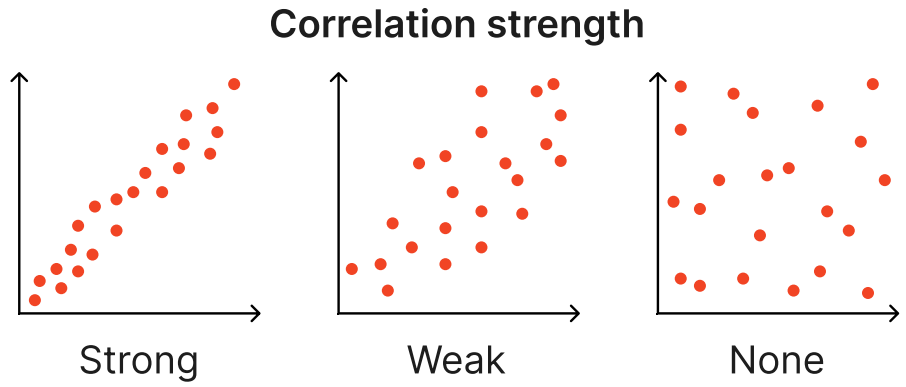
$$\sigma = \sqrt{var(X)}.$$

Vizualizácie štatistík spojitého príznaku

Typické vizualizácie rozdelenia sú histogram a box plot. Ukážka oboch grafov je na obrázku 3.8. Histogram funguje na princípe združovania pozorovaní do približne rovnako veľkých intervalov (angl. bins) a vykreslenia počtu pozorovaní, ktoré spadli do každého intervalu. Nástroje, ktoré vykresľujú histogramy často umožňujú zvoliť počet intervalov, alebo šírku intervalu. Box plot využíva na vykreslenie kvartily. Obsahuje obdĺžnik (angl. box, preto box plot), ktorého horná hrana znázorňuje horný kvartil, dolná hrana zas dolný kvartil. Obdĺžnik je predelený na dve časti mediánom. Graf obsahuje ešte dve horizontálne čiary znázorňujúce minimálnu a maximálnu hodnotu. Body prekračujúce tieto hodnoty sa vykreslia bodkou. Existuje viac možností ako určiť minimálnu a maximálnu hodnotu. Častým riešením je ako maximum zvoliť $Q_{0,75}$ + 1,5IQR a ako minimum zas $Q_{0,25}$ - 1,5IQR.

3.3.3 Bivariačná deskriptívna štatistika

Vzájomný vzťah dvoch príznakov je možné vyjadriť pomocou bivariačných deskriptívnych štatistík. Pokiaľ príznaky medzi sebou nemajú žiadny vzťah, sú nezávislé. Znamená to, že sa príznaky navzájom neovplyvňujú a nemôžeme jeden predikovať na základe druhého. Presnejšia definícia je nasledovná:



Obr. 3.9: Rôzne veľké korelácie (zľava silná, slabá a žiadna) ukázané na bodovom grafe [3]

Definícia 9. Nech X a Y sú náhodné veličiny a $F_{X,Y}$, F_X a F_Y kumulatívne distribučné funkcie. X a Y sú *nezávislé*, ak pre všetky $x \in X$ a $y \in Y$ platí:

$$F_{X,Y}(x, y) = F_X(x)F_Y(y).[29]$$

Mieru závislosti dvoch príznakov môžeme vyjadriť pomocou korelačných koeficientov. V praxi sa často používa Pearsonov korelačný koeficient, ktorý predpokladá lineárny vzťah medzi príznakmi a je vhodný len pre spojité príznaky. Vypočíta sa nasledovne (E predstavuje strednú hodnotu):

$$\rho_{X,Y} = \frac{cov(X, Y)}{\sigma_X \sigma_Y} = \frac{E[(X - E[X])(Y - E[Y])]}{\sigma_X \sigma_Y}.[29]$$

Kompaktný spôsob ako zobraziť korelácie medzi všetkými dvojicami príznakov je korelačná matica. Nech $X_1, X_2, \cdots, X_n$ sú príznaky. Korelačná matica je $n \times n$ matica, ktorej i, j bunka zobrazuje koreláciu medzi X_i a X_j (metriku si môžme zvoliť).

Pri voľbe správnej vizualizácie záleží na type príznakov, ktoré chceme porovnať. Pre dva kategorické príznaky je vhodné vytvoriť kontingenčnú tabuľku. Označme X všetky kategórie prvého príznaku a Y všetky kategórie druhého príznaku. Kontingenčná tabuľka má pre každú dvojicu (x, y) kde $x \in X, y \in Y$ bunku, ktorej hodnota sa rovná počtu pozorovaní, kde mal prvý príznak hodnotu x a druhý y.

Pre dva spojité príznaky je vhodné využiť bodový graf. Graf je tvorený dvoma osami (každá pre hodnoty jedného príznaku) a pozorovania (dvojice hodnôt) sa vykreslia ako body. Ukážka je na obrázku 3.9.

Vzťah medzi kategorickým a spojitým príznakom môžme skúmať napríklad pomocou box plotov či histogramov pre každú kategóriu zvlášť.

3.4 Textové dáta

Pod pojmom textové dáta rozumieme dáta, ktoré obsahujú prirodzený jazyk. Môže sa jednať o množinu hodnotení zákazníkov eshopu, zoznam vedeckých článkov, databázu vyjadrení politikov a podobne. Spracovanie prirodzeného jazyka (angl. natural language processing, skrátene NLP) je odvetvie strojového učenia, ktoré definuje metódy a postupy, na základe ktorých môžu počítače vyťažovať informácie z textu. Pokiaľ chceme vytvoriť zaujímavé grafické reprezentácie textových dát, je nutné aplikovať niektoré NLP metódy. V tejto sekcii vysvetlím, v akom formáte sa často ukladajú textové dáta, pomocou akých metód je možné ich spracovať a ktoré vizualizácie sa v tomto odvetví často používajú. Budem vychádzať zo zdroja [30].

3.4.1 Reprezentácia

Dokumenty môžu byť v počítači uložené v rôznych súboroch, alebo napríklad v jednom csv súbore, pričom texty oddelíme nejakým vopred určeným znakom. Druhá možnosť, ktorá je vhodná pri určitých typoch NLP úloh, je z textov vytvoriť tabuľku. Takáto tabuľka sa nazýva *bag of words*, pretože sa nesústredí na poradie slov v dokumente, ale len na to či a koľko krát sa v texte vyskytujú.

Tabuľka *bag of words* obsahuje riadok pre každý dokument a stĺpec pre každé slovo, ktoré sa aspoň raz vyskytlo v aspoň jednom dokumente. Takýchto tabuliek existuje viac typov, pričom sa líšia v tom, akým spôsobom sa vypočíta hodnota v ich bunkách. Popíšem tri z nich - booleovu, frekvenčnú a tf-idf tabuľku. Tvorba každého z týchto typov začína vypočítaním frekvencie každého slova v každom dokumente.

Definícia 10. Nech D je množina všetkých dokumentov a W je množina všetkých slov, ktoré sa v dokumentoch vyskytujú. Pre každé slovo $w \in W$ a dokument $d \in D$ definujeme *frekvenciu* slova v danom dokumente n_{wd} ako počet výskytov slova w v dokumente d. *Relatívna frekvencia* slova tf_{wd} (angl. term frequency) sa potom vypočíta nasledovne:

$$tf_{wd} = \frac{n_{wd}}{\sum_{w' \in d} n_{w'd}}.$$

Definícia 11. Nech D je množina všetkých dokumentov a W je množina všetkých slov, ktoré sa v dokumentoch vyskytujú. *Inverzná dokumentová frekvencia* (angl. inverse document frequency) reprezentuje zlogaritmovaný inverzný pomer dokumentov, ktoré obsahujú dané slovo:

$$idf_w = log\left(\frac{|D|}{|\{d \in D : w \in d\}|}\right).$$

Hodnotu bunky h_{wd} pre slovo w a dokument d v rôznych typoch tabuliek potom vypočítame podľa nasledovnej tabuľky:

typ	h_{wd}
booleova tabuľka	$\begin{cases} 0 & \text{ak } n_{wd} = 0 \\ 1 & \text{inak} \end{cases}$
frekvenčná tabuľka	n_{wd}
tf-idf tabuľka	$\frac{tf_{wd}}{idf_w}$

3.4.2 Predspracovanie

Aby sme mohli reprezentovať text pomocou *bag of words* tabuľky, je nutné z neho extrahovať jednotlivé slová. Tento proces sa nazýva tokenizácia. Slová sú v texte obvykle oddelené medzerami či interpunkčnými znamienkami. Tokenizácia teda spočíva v tom, že si určíme oddeľovače (biele znaky, interpunkčné znamienka, ...), pomocou ktorých text rozdelíme na slová. Podobnou úlohou je rozdeľovanie textu na vety (angl. sentence splitting). Aj keď by sa mohlo zdať, že sa jedná o triviálnu záležitosť, nie je to tak. Bodky a iné interpunkčné znamienka v texte nevystupujú vždy ako rozdeľovače viet. Bodku môžeme v texte vidieť napríklad za skratkami či titulmi.

Po tokenizácii je vhodné vykonať lematizáciu, ktorá zníži dimenzionalitu. Lematizácia je proces, ktorý k danému slovu nájde jeho základný tvar (tzv. lemma). Dá sa aplikovať na všetky jazyky, ale má obzvlášť veľký prínos pri jazykoch, ktoré obsahujú skloňovanie. Bez lematizácie by sa slová vizualizácia, vizualizáciou a vizualizáciami považovali za tri rozdielne slová. Po lematizácii by boli správne identifikované ako jedno slovo.

Medzi ďalšie časté úpravy patrí odstránenie slov s chybami a odstránenie často používaných slov (angl. stop words). Často používané slová sú napríklad predložky a spojky. Jedná sa o slová, ktoré sú tak časté, že majú nízku informačnú hodnotu a nepomôžu nám pri riešení úlohy (napr. identifikovanie témy dokumentu, analýza podobnosti dokumentov, ...). Obvykle je k dispozícii slovník často používaných slov, ktorý je možné použiť na ich identifikáciu. Pokiaľ slovník nemáme, dajú sa odstrániť slová, ktoré majú vysokú frekvenciu naprieč všetkými dokumentami. Na podobnom princípe funguje aj identifikácia chybných slov - jedná sa o slová, ktoré majú veľmi nízku frekvenciu (napr. jeden výskyt v celej databáze dokumentov).

Obr. 3.10: Word cloud graf [4]

3.4.3 Vizualizácie textových dát

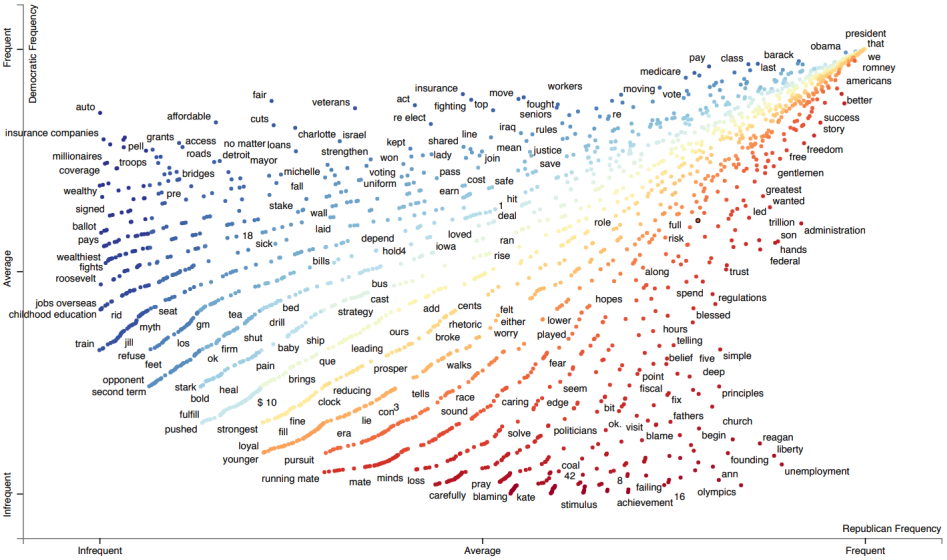
Textové dáta sa samozrejme dajú vizualizovať veľmi jednoducho - použijeme ekvivalent metódy `print()` v programovacom jazyku, v ktorom pracujeme. Takáto vizualizácia sa ale s narastajúcim počtom dokumentov stáva veľmi neprehľadnou. V tejto sekcii ukážem niekoľko vizualizácií, ktorých cieľom je sumarizovať dokument, alebo množinu dokumentov a ukázať vzťahy medzi rôznymi dokumentami či slovami v dokumente.

Word cloud

Word cloud je veľmi často používaná vizualizácia jedného či viacerých dokumentov. Ako môžeme vidieť na obrázku 3.10, zobrazuje slová, ktoré boli použité v dokumente, v rôznej farbe a veľkosti podľa ich frekvencie. Pomocou tejto vizualizácie môžeme jedným pohľadom zistiť, ktoré slová sa v dokumente vyskytovali najčastejšie. Pokiaľ pred vizualizovaním aplikujeme lematizáciu a odstránime stop words, najčastejšie slová by mali úzko súvisieť s témou dokumentu. Niekedy sa namiesto jednotlivých slov vizualizujú N-gramy slov.

Definícia 12. Pod pojmom *n-gram* rozumieme sekvenciu n písmen či slov.

N-gramy sa používajú napríklad pri automatickom doplňovaní slov a celých fráz vo vyhľadávačoch. Doplňovanie funguje na základe toho, že niektoré sek-



Obr. 3.11: Bodový graf porovnávajúci frekvenciu slov v prejavoch demokratických a republikánskych politikov z roku 2012 [5]

vencie znakov a slov sú pravdepodobnejšie ako iné. Na podobnom princípe funguje aj automatická oprava písaného textu (angl. autocorrect).

Treba podotknúť, že na word cloud sa často referuje ako na koláčový graf v NLP. Koláčový graf nie je vhodný pre vizualizáciu viac ako troch kategórií. V prípade, že majú niektoré kategórie podobné percentá, je pre koncového používateľa ťažké detekovať, ktorá kategória je percentuálne zastúpená viac. Podobný problém nastáva aj pri word cloude. Skúste sa pozrieť na obrázok 3.10 a odpovedať na otázku: „Ktoré slovo je piate najčastejšie používané?“ Nie je to také jednoduché. Pre porovnanie prehľadnosti word cloudu a stĺpcového grafu sa môžete pozrieť do prílohy C.1. Stĺpcový graf zobrazuje rovnakú informáciu a dá sa pomocou neho odpovedať na predchádzajúcu otázku oveľa jednoduchšie. Môžeme teda povedať, že stĺpcový graf je v tomto prípade efektívnejší.

Bodový graf

Bodový graf vieme v NLP využiť na porovnanie frekvencie slov v dvoch sadách dokumentov [5]. Na obrázku 3.11 je vidieť bodový graf, ktorý bol postavený na vyjadreniach amerických politikov z roku 2012. V Amerike je väčšina politikov členmi jednej z dvoch strán - demokratickej a republikánskej. Bodový graf preto porovnáva frekvenciu slov vo vyjadreniach republikánskych politikov (os x) a demokratických politikov (os y). V pravom dolnom rohu nájdeme slová, ktoré používali prevažne republikáni, v ľavom hornom rohu zas slová, ktoré používali hlavne demokrati.

Vizualizácie založené na pokročilých metódach

Vizualizácie textu, ktoré boli spomenuté doteraz záviseli len na frekvencii slov. Po aplikácii niektorých pokročilejších metód NLP môžu vzniknúť ďalšie zaujímavé vizualizácie. Jednou z týchto metód je analýza sentimentu. Cieľom analýzy sentimentu je určiť, či je daný text pozitívny, neutrálny alebo negatívny. Obvykle sú výstupom takejto analýzy tri čísla reprezentujúce pravdepodobnosť, že sa jedná o pozitívny, neutrálny či negatívny text. Je viacero spôsobov ako k analýze sentimentu pristúpiť. Jedným z nich je využiť lexikón, v ktorom sú uvedené slová s pozitívnym a negatívnym významom a celkový sentiment dokumentu určiť podľa početnosti takýchto slov. Druhý spôsob je natrénovať ML model na dátach, ktoré už majú sentiment určený. Po vykonaní analýzy sentimentu môžeme zobraziť histogram pre každú z troch kategórií sentimentu, alebo napríklad zobraziť časový vývoj sentimentu vzhľadom k nejakému človeku/produktu.

Ďalšou zaujímavou metódou NLP je výpočet podobnosti dvoch dokumentov. Relatívne jednoduchá metrika na výpočet podobnosti je Jaccardova podobnosť. Jej nevýhodou je, že neberie do úvahy početnosť výskytov jednotlivých slov.

Definícia 13. Nech D_1 a D_2 sú množiny všetkých slov dokumentov d_1 a d_2. *Jaccardovu podobnosť* vypočítame nasledovne:

$$J(D_1, D_2) = \frac{|D_1 \cap D_2|}{|D_1 \cup D_2|}.$$

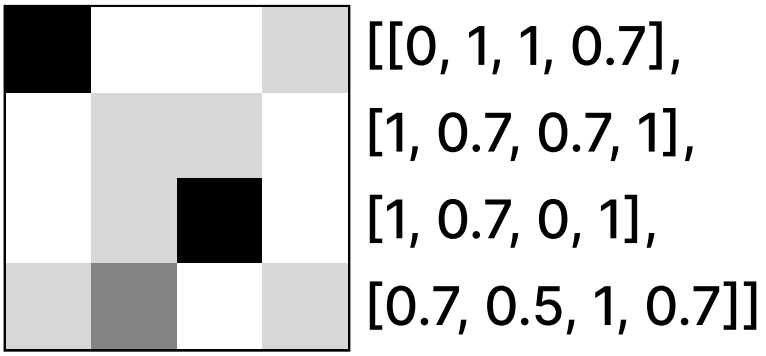
Pokiaľ máme vypočítanú podobnosť každej dvojice dokumentov, môžeme textové dáta transformovať na graf. Medzi každé dva dokumenty umiestnime hranu, pokiaľ je ich podobnosť väčšia ako zvolená medza. Potom je možné aplikovať všetky vizualizácie popísané v sekcii 3.7.

3.5 Obrazové dáta

Vďaka pokroku, ktorý nastal v oblasti strojového videnia je možné využiť umelú inteligenciu na riešenie rôznych problémov založených na spracovaní obrazu. Patrí medzi ne napríklad klasifikácia obrázkov, rozpoznávanie tváre, detekcia a sledovanie objektov, optické rozpoznávanie znakov a podobne. V tejto sekcii na základe informácií zo zdroja [31] vysvetlím ako sú reprezentované digitálne obrázky, popíšem niektoré farebné modely a ukážem ako sa dajú obrázky upravovať pomocou algebraických operácií.

3.5.1 Pixel

Každý digitálny obrázok sa dá deliť na menšie a menšie časti. Najmenšia jednotka, ktorá sa ďalej už nedá deliť sa nazýva základná jednotka. V termi-



Obr. 3.12: Zjednodušený obrázok (vľavo) a jeho reprezentácia pomocou 2D poľa pixelov (vpravo)

nológii 2D obrázkov sa základná jednotka nazýva pixel, pri 3D objektoch sa nazýva voxel.

Pixel je teda základnou jednotkou 2D obrázkov. Každý pixel má dve súradnice vyjadrujúce jeho pozíciu vrámci obrázka. Keď obrázok rozdelíme na pixely, vznikne nám 2D pole. Súradnice teda obvykle vyjadrujú riadok a stĺpec, v ktorom sa daný pixel nachádza. Okrem súradníc má každý pixel jednu alebo viacero numerických hodnôt, ktoré reprezentujú jeho vlastnosti. Čiernobiele obrázky je možné definovať pomocou pixelov s jednou hodnotou (obvykle v rozsahu $[0, 1]$) reprezentujúcou jas pixelu:

- 0 - najnižší jas, čierna farba
- $(0, 1)$ - odtiene šedej
- 1 - najvyšší jas, biela farba

Farebné obrázky sa obvykle reprezentujú pomocou pixelov s troma hodnotami. Jednotlivé hodnoty popisujú množstvo červenej, zelenej a modrej farby a často sú v rozsahu $[0, 255]$ alebo $[0, 1]$. Ako je možné z troch farieb dostať všetky ostatné farby je vysvetlené v sekcii 3.5.3.

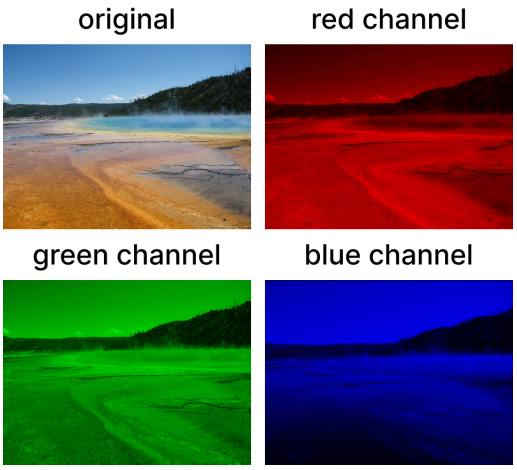
3.5.2 Reprezentácia obrázku

Definícia 14. Nech pre pixel so súradnicami (x, y) a n-dimenzionálnou hodnotou v platí $x \in X$, $y \in Y$ a $v \in V^n$. *Digitálny obrázok* potom môžme definovať ako funkciu

$$f : X \times Y \to V^n,$$

ktorá každému pixelu priradí na základe jeho súradníc (x, y) hodnotu

$$v = f(x, y).$$



Obr. 3.13: Dekompozícia obrázka do troch farebných kanálov [6]

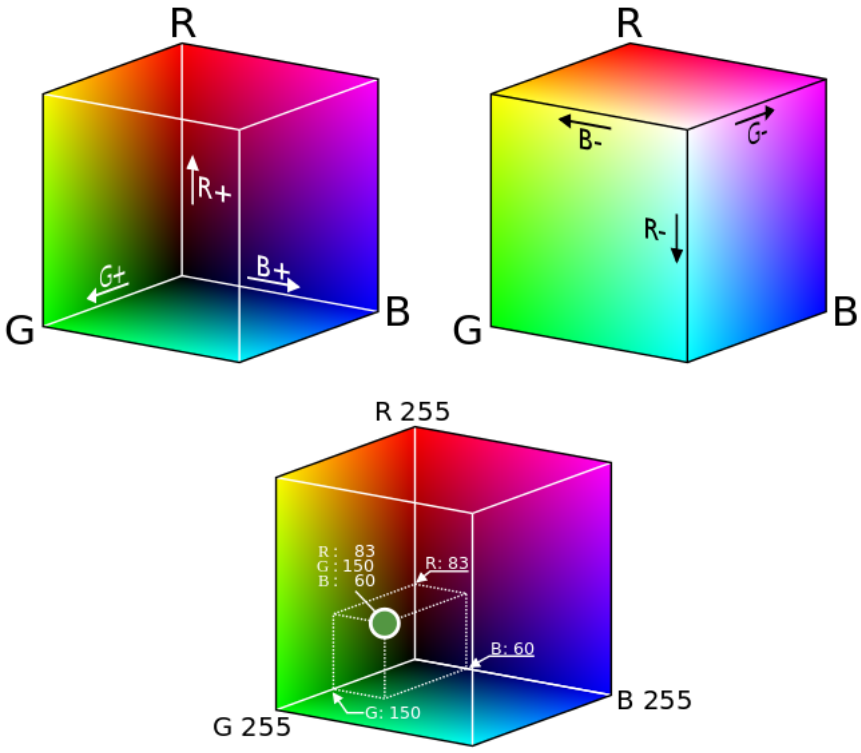
V praxi sa stretneme s obrázkami reprezentovanými pomocou 2D poľa hodnôt (hodnota môže byť skalár, alebo vektor), kde sú súradnice pixelov dané implicitne pomocou pozície v poli (tzn. sú to celé čísla). Ukážka je na obrázku 3.12.

3.5.3 Farby

Ako som už spomínala, farebné digitálne obrázky sa obvykle reprezentujú pomocou pixelov s troma hodnotami. To znamená, že ich vieme rozložiť do troch kanálov. Kanál v tomto kontexte znamená obrázok rovnakej veľkosti ako pôvodný obrázok, ktorého pixely majú len jednu hodnotu. Ak mal teda pôvodný obrázok tri hodnoty odpovedajúce intenzitám červenej, zelenej a modrej farby, dá sa rozložiť do troch kanálov (jeden pre každú farbu). Túto dekompozíciu je možné vidieť na obrázku 3.13. Každý kanál je čiernobiely obrázok, pričom biela farba predstavuje maximálnu intenzitu farby kanálu.

Farebné modely

Digitálne obrázky je možné kódovať aj inými spôsobmi ako intenzitou červenej, zelenej a modrej farby. Pod pojmom farebný model rozumieme súradnicový systém na reprezentáciu farieb. Každá farba odpovedá nejakému bodu v tomto systéme. Existuje veľa farebných modelov, vytvorených na rôzne účely. Patria medzi ne napríklad RGB, CMY, CMYK, HSI, HSV, HSB a podobne. V tejto sekcii popíšem ako fungujú niektoré z nich.



Obr. 3.14: Súradnicová sústava RGB modelu [7]

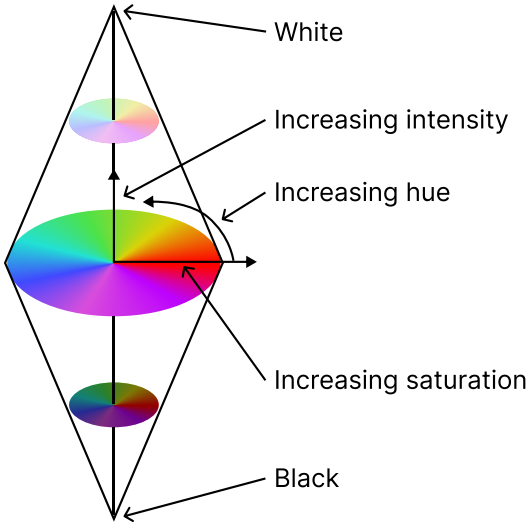
RGB model

RGB model je často používaný model na digitálnu reprezentáciu farieb, ktorý je inšpirovaný ľudským zrakovým systémom. Ľudské oko má tri druhy receptorov (tzv. čapíky), ktoré sú citlivé v červenej, zelenej a modrej oblasti viditeľného spektra. Svetlá rôznej vlnovej dĺžky môžu byť zachytené jedným receptorom alebo nejakou ich kombináciou a vďaka tomu je možné vidieť mnoho farieb.

RGB model funguje podobne. Má definované tri základné farby - červenú (R), zelenú (G) a modrú (B). Jedná sa o 3D súradnicový systém, kde každá os reprezentuje jednu zo základných farieb a ich kombináciou je možné získať ostatné farby. Ukážka je na obrázku 3.14.

HSI model

HSI je skratka pre odtieň (angl. hue), sýtosť (angl. saturation) a intenzitu (angl. intensity). HSI model si môžeme predstaviť ako dva kužele spojené podstavami ako je ukázané na obrázku 3.15. Pomocou intenzity sa v tomto objekte pohybujeme vertikálne. Najnižšia intenzita reprezentuje čiernu farbu (na obrázku úplne dolu) a najvyššia bielu (na obrázku úplne hore). Ak inten-



Obr. 3.15: Vizualizácia HSI modelu

zitu zafixujeme na nejakej hodnote a spravíme v nej prierez celého objektu, dostaneme kruh. Pre ľubovoľný bod na tomto kruhu vieme skonštruovať vektor vedúci zo stredu kruhu do daného bodu. Odtieň (H) potom dostaneme ako uhol, ktorý tento vektor zviera s osou R a sýtosť (S) je priamoúmerná dĺžke tohto vektora. Je vhodné poznamenať, že pokiaľ sa bod nachádza priamo na osi intenzity, hodnota sýtosti je 0 a odtieň nie je definovaný.

CMY a CMYK model

CMY je model, ktorý sa používa na fyzickú reprezentáciu farieb, napríklad v tlačiarňach. C, M a Y reprezentujú tri farby - tyrkysovú (angl. cyan), purpurovú (angl. magenta) a žltú (angl. yellow). Zmiešaním všetkých troch farieb by teoreticky mala vzniknúť čierna. Týmto sa CMY model líši od RGB modelu, v ktorom zmiešanie všetkých troch farieb vytvorí bielu. Jednoduchá rovnica na približnú konverziu z CMY do RGB je:

$$R = 1 - C$$
$$G = 1 - M$$
$$B = 1 - Y.$$

Rozdiel medzi CMY a CMYK je v tom, že CMYK model má okrem troch základných farieb k dispozícii aj čiernu. Tento model vznikol kvôli tomu, že výsledok zmiešania tyrkysovej, purpurovej a žltej v niektorých prípadoch (farby nie sú presne dané) nevytvoril čiernu. Čierny atrament do tlačiarní je často lacnejší ako tie farebné, takže použitie CMYK modelu je ekonomickejšie.

3.5.4 Úpravy a vizualizácia obrázkov

Na základe znalosti digitálnej reprezentácie obrázka ho môžeme jednoducho upravovať. Operácia orezania funguje na princípe výberu niektorých pixelov (tzn. určíme, aký interval stĺpcov a riadkov nás zaujíma). Rotácia obrázka o uhol θ proti smeru hodinových ručičiek sa dá dosiahnuť násobením rotačnou maticou:

$$R = \begin{pmatrix} cos(\theta) & -sin(\theta) \\ sin(\theta) & cos(\theta) \end{pmatrix}$$

Nové súradnice x' a y' potom dostaneme z pôvodných súradníc x a y nasledovne:

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = R \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} xcos(\theta) - ysin(\theta) \\ xsin(\theta) + ycos(\theta) \end{pmatrix}$$

Roztiahnutie obrázku môžeme dosiahnuť vynásobením maticou

$$\begin{pmatrix} j & 0 \\ 0 & k \end{pmatrix}, \text{ pre } j > 0 \text{ a } k > 0,$$

kde j určuje faktor roztiahnutia pre os x a k pre os y. Obrázok sa v danej osi zväčší, pokiaľ je faktor väčší ako 1. Podobné matice existujú aj na iné transformácie ako je prevrátenie obrázku po jednej z osí či jeho skosenie. Tieto operácie sú problematické kvôli tomu, že po rotácii musia byť nové pozície zaokrúhlené na celé čísla. Z toho dôvodu sa môže stať, že sa niekoľko pixelov premietne na tú istú pozíciu a na niektoré pozície sa naopak nepremietne žiadny pixel.

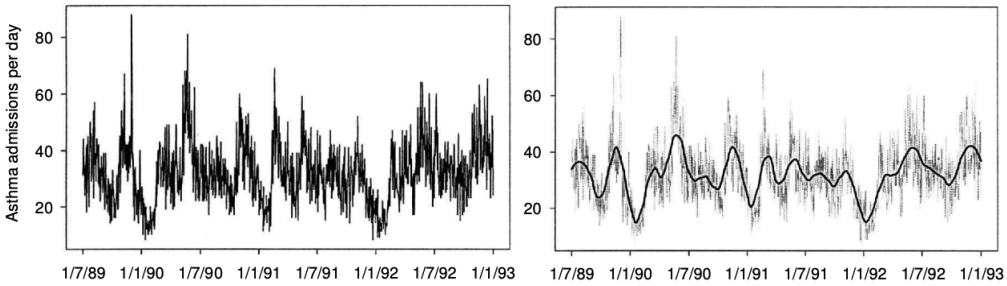
Pomocou vynásobenia obrázkovej matice kladným číslom väčším resp. menším ako 1 je možné zvýšiť resp. znížiť jas. Odčítaním dvoch obrázkov vieme detekovať, či medzi nimi nastala zmena. Na čiernobiely obrázok môžeme aplikovať prahovanie. Ide o jednoduchú metódu, ktorá sa často používa pri segmentácii obrázkov. Pomocou prahovania vytvoríme nový obrázok, ktorého pixely majú binárnu hodnotu. Nová hodnota v' vznikne porovnaním pôvodnej hodnoty v s medzou t, ktorú zadáme:

$$v' = \begin{cases} 1, & \text{ak } v \geq t \\ 0, & \text{inak.} \end{cases}$$

Okrem vizualizácie samotného obrázka alebo nejakej z jeho transformácií, o ktorých som písala vyššie, je niekedy vhodné zobraziť aj jeho vlastnosti. To môžeme dosiahnuť pomocou histogramu jasu či histogramov jednotlivých základných farieb.

3.6 Časové rady

Na časové rady človek narazí snáď v každom odvetví, kde má zmysel porovnávať vývoj nejakej veličiny naprieč časom. Medzi dáta, ktoré sa často



Obr. 3.16: Pôvodný (vľavo) a vyhladený (vpravo) graf časového radu [8]

prezentujú vo forme časového radu patria napríklad menové kurzy, ceny komodít, meteorologické, demografické, epidemiologické dáta a mnoho ďalších. Časový rad sa často definuje ako náhodný proces takto.

Definícia 15. Buď (Ω, F, P) pravdepodobnostný priestor a T množina indexov interpretovaných ako čas. *Časový rad* je potom množina $\{X_t, t \in T\}$, kde X_t sú náhodné veličiny z (Ω, F, P) [32].

V prípade, že $t \in \mathbb{Z}$ alebo $t \in \mathbb{N} + \{0\}$ hovoríme o náhodnom procese s diskrétnym časom. Ak naopak $t \in \mathbb{R}$, ide o náhodný proces so spojitým časom. V nasledujúcom texte ale postačí zjednodušená definícia, ktorá sa na časový rad pozerá ako na súbor nejakých pozorovaní (označovaných napríklad y_t) získaných v konkrétnych časových okamihoch t.

Časové rady sa obvykle vizualizujú pomocou čiarového grafu. X-ová os reprezentuje čas t a y-ová os hodnoty pozorovaní x_t. Graf sa buď vykreslí zo surových dát, alebo dôjde k ich vyhladeniu pomocou nejakej z vyhladzovacích metód. Graf časovej rady pred a po vyhladení je možné vidieť na obrázku 3.16.

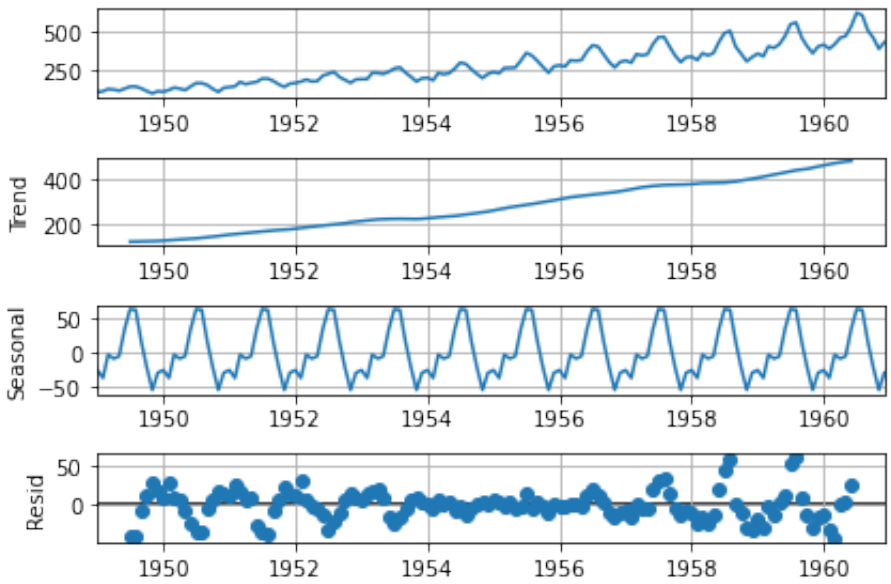
Pri analýze časového radu sledujeme jeho charakteristické vlastnosti ako sú napríklad periodicita, trendy, sezónnosť a podobne. Analýza je často zameraná na jeden z dvoch cieľov:

- popis pozorovaného javu,
- predikcia budúceho vývoja.

V oboch prípadoch sú často výstupom analýzy rôzne vizualizácie. V nasledujúcich sekciách popíšem niektoré z nich.

3.6.1 Dekompozícia časového radu

Pokiaľ explicitne neuvediem iný zdroj, v tejto sekcii vychádzam zo zdroja [33]. Pri analýze časových radov sa vychádza z predpokladu, že ich je možné rozložiť na štyri komponenty. Takéto delenie sa nazýva dekompozícia časového radu. Medzi spomínané komponenty paria:



Obr. 3.17: Ukážka aditívnej dekompozície vývoja pasažierov aerolinky. Zhora vidíme originálny časový rad, trend, sezónnu zložku a reziduály.

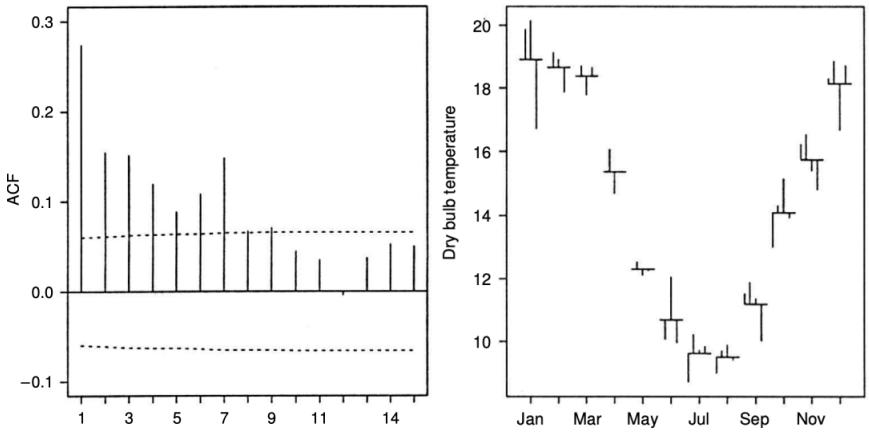
- trend T_t,
- sezónnosť S_t,
- cyklické zmeny C_t,
- ďalšie nepravidelné fluktácie.

Cyklické zmeny a nepravidelné fluktácie sa často zlučujú do jednej zložky označovanej E_t. Vizualizácia dekompozície časového radu do týchto troch zložiek je ukázaná na obrázku 3.17. Zo získaných zložiek môžeme dekompozíciu realizovať pomocou aditívnej alebo multiplikatívnej formy:

- aditívna dekompozícia: $Y_t = T_t + S_t + E_t$,
- multiplikatívna dekompozícia: $Y_t = T_t \cdot S_t \cdot E_t$.

Popis komponentov

Trend vyjadruje všeobecnú tendenciu vývoja skúmaného javu za dlhé obdobie. Môže byť rastúci, klesajúci a existujú aj časové rady bez trendu. Časový rad obsahuje trend, pokiaľ je v ňom pozorovateľná dlhodobá zmena v strednej hodnote [34].



Obr. 3.18: Ukážka autokorelačného (vľavo) a sezónneho grafu (vpravo) [8]

Sezónna zložka je pravidelne sa opakujúca odchýlka od trendu. O časovom rade povieme, že je na ňom pozorovateľná sezónnosť, pokiaľ je ovplyvnený sezónnymi faktormi ako sú napr. deň v týždni alebo mesiac v roku. Sezónnosť sa teda dá definovať ako vzor, ktorý sa v časovom rade opakuje vo fixných časových intervaloch. Typický príklad sezónnosti v časových radoch si môžme predstaviť na vývoji teploty v priebehu roku, počte zákazníkov v horských centrách a podobne. Sezónny graf (angl. seasonal subseries plot) slúži na vizualizáciu zmien vrámci sezón. Pre každú sezónu sa vykreslí stredná hodnota v podobe horizontálnej čiary a následne sa ako body alebo vertikálne čiary nanesú jednotlivé hodnoty radené podľa času. Sezónny graf je ukázaný na obrázku 3.18.

Cyklické zmeny udávajú kolísanie okolo trendu v dôsledku dlhodobého cyklického vývoja, kedy dochádza k striedaniu fáz rastu a klesania. Jednotlivé cykly sa vytvárajú za dlhé obdobie a môžu mať rôznu amplitúdu.

Nepravidelné fluktácie sú náhodné a iné nesystematické výkyvy (napríklad chyby merania). Mali by byť tvorené normálnym bielym šumom. Pod pojmom biely šum rozumieme náhodný proces $\{X_t\}$, pre ktorý platí:

$$\begin{aligned} \mathbb{E}[X_t] &= 0, \\ var(X_t) &= \sigma^2 < \infty, \\ cov(X_t, X_{t+\pi}) &= 0, \pi > 0. \end{aligned}$$

Ide teda o proces, ktorého rozdelenie má konštantný konečný rozptyl, nulovú strednú hodnotu a jednotlivé náhodné veličiny sú vzájomne nezávislé. Normálny (gaussovský) biely šum je špeciálny biely šum, pre ktorý navyše platí

$$X_t \sim \mathcal{N}(0, \sigma^2).$$

Realizácia dekompozície

Existuje viacero metód na dekompozíciu časových radov. Tá klasická vznikla už okolo roku 1920 a dodnes slúži ako základ pre sofistikovanejšie metódy. Prvý krok tejto metódy je použiť metódu kĺzavého priemeru na odhad trendu.

Odhad trendu pomocou kĺzavého priemeru

Kĺzavý priemer stupňa m (označuje sa m-MA z anglického moving average) sa dá zapísať ako

$$\hat{T}_t = \frac{1}{m} \sum_{j=-k}^{k} y_{t+j},$$

kde $m = 2k + 1$. Teda odhad trendu v čase t dostaneme tak, že spriemerujeme hodnoty od y_{t-k} do y_{t+k}. Myšlienka za priemerovaním spočíva v tom, že hodnoty, ktoré idú v časovom rade bezprostredne za sebou budú pravdepodobne podobné. Priemerovanie potom eliminuje šum a zostane odhad trendu. Obvykle je stupeň m nepárne číslo, aby bol kĺzavý priemer symetrický (tzn. vypočítal sa z rovnakého počtu pozorovaní na pravej a ľavej strane od y_t).

Je možné vytvoriť kĺzavý priemer z kĺzavého priemeru. Značenie vyzerá veľmi podobne. 2x4-MA znamená kĺzavý priemer stupňa 2 aplikovaný na kĺzavý priemer stupňa 4. Jeden z dôvodov dvojitej aplikácie je spraviť kĺzavý priemer nepárneho stupňa symetrickým. To je viditeľné na tomto rozpísanom výraze pre 2x4-MA:

$$\begin{aligned}\hat{T}_t &= \frac{1}{2}\left[\frac{1}{4}(y_{t-2} + y_{t-1} + y_t + y_{t+1}) + \frac{1}{4}(y_{t-1} + y_t + y_{t+1} + y_{t+2})\right] \\ &= \frac{1}{8}y_{t-2} + \frac{1}{4}y_{t-1} + \frac{1}{4}y_t + \frac{1}{4}y_{t+1} + \frac{1}{8}y_{t+2}\end{aligned}$$

Klasická dekompozícia

Označme sezónnu periódu m (napr. $m = 7$ pre denné dáta, $m = 12$ pre mesačné dáta) a predpokladajme, že sa v priebehu časového radu nemení. Aditívnu a multiplikatívnu dekompozíciu môžme realizovať veľmi podobným princípom. Popíšem teda princíp pre aditívnu dekompozíciu a rozdiely s tou multiplikatívnou uvediem v hranatých zátvorkách.

1. Ak je m párne, odhadneme trend $\hat{T}_t$ pomocou 2xm-MA. Ak je m nepárne, odhadneme trend $\hat{T}_t$ pomocou m-MA.

2. Vypočítame detrendovaný časový rad ako $y_t - \hat{T}_t$ [$y_t/\hat{T}_t$].

3. Na odhad sezónnej komponenty potom stačí spraviť priemer detrendovaných hodnôt pre každú sezónu a pomocou replikácie skonštruovať odpovedajúci časový rad $\hat{S}_t$.

4. Reziduály vypočítame pomocou odčítania [delenia] odhadnutých komponentov trendu a sezónnosti: $\hat{R}_t = y_t - \hat{T}_t - \hat{S}_t$ [$\hat{R}_t = y_t/(\hat{T}_t\hat{S}_t)$].

Iné metódy dekompozície

Klasická dekompozícia má určité nevýhody. Odhad trendu nie je dostupný pre niekoľko prvých a posledných pozorovaní a je nepresný pri rapídnych zmenách v dátach. Druhou nevýhodou je, že predpoklad na nemennosť sezónnej periódy nie je vždy splnený a v takom prípade táto metóda nedosiahne dobré výsledky.

Ďalšou populárnou metódou na dekompozíciu kvartálnych a mesačných dát je metóda X11. Je založená na klasickej dekompozícii, ale obsahuje veľa pridaných procedúr, aby vyriešila jej nedostatky. Odhady trendu sú dostupné pre všetky pozorovania a sezónna komponenta sa môže v priebehu času jemne meniť. Táto metóda je tiež omnoho odolnejšia voči outlierom [35]. Ďalšie metódy dekompozície sú napríklad:

- SEATS - funguje len na kvartálne a mesačné dáta [35]
- STL - veľmi robustná metóda, ktorá podporuje ľubovoľný typ sezónnosti (nemusí sa jednať ani o konštantnú sezónnosť) [36]

3.6.2 Autokorelácia

Charakteristická vlastnosť časových radov je to, že pozorovania sú zoradené v čase. Autokorelačná funkcia $r(k)$ udáva mieru závislosti dvoch zložiek časového radu, ktoré boli namerané v časoch posunutých o k (posun, angl. lag) časových jednotiek. Na autokoreláciu sa teda môžme pozerať ako na koreláciu časového radu s jeho oneskorenou verziou. Hodnotu autokorelačnej funkcie vypočítame ako:

$$r_k = \frac{\sum_{t=k+1}^{n}(x_t - \tilde{x})(x_{t-k} - \tilde{x})}{\sum_{t=1}^{n}(x_t - \tilde{x})^2}.[8]$$

Ako môžeme vidieť na obrázku 3.18, autokorelačný graf zobrazuje na x-ovej osi hodnotu posunu k a na y-ovej osi hodnotu autokorelačnej funkcie. Voliteľne môže obsahovať aj intervaly spoľahlivosti.

3.6.3 Vzťah medzi viacerými časovými radmi

Vzťah medzi dvoma časovými radmi je možné zobraziť pomocou bodového grafu. Jeden bod predstavuje hodnotu jedného (x-ová súradnica) aj druhého (y-ová súradnica) radu v tom istom čase. Na skúmanie vzťahu viacerých časových radov sa dá využiť matica bodových grafov, ktorá obsahuje bodový graf pre každú dvojicu.

3.7 Grafy a siete

Aký je rozdiel medzi pojmami graf a sieť? To bola jedna z prvých otázok, ktorú som mala, keď som začínala s rešeršou tejto témy. Rovnakú otázku mal aj vedec Albert-László Barabási a odpovedal na ňu v jeho knihe *Network Science* [37]. Píše, že vo vedeckej literatúre sa pojmy graf a sieť a im prislúchajúce pojmy používajú zameniteľne a záleží hlavne na kontexte. Ukážka pojmov a zaradenie do kontextu je v nasledujúcej tabuľke.

teória sietí (network science)	teória grafov (graph theory)
sieť (network)	graf (graph)
uzol (node)	vrchol (vertex)
hrana (link)	hrana (edge)

Ďalej podotýka, že medzi oboma terminológiami je jemný rozdiel. Pojmy ako sieť, uzol a hrana sa používajú na označenie reálnych systémov. WWW je sieť webových dokumentov prepojených pomocou URL linkov, spoločnosť je sieť individuálov prepojených pomocou rodinných, kamarátskych či profesionálnych vzťahov a podobne. Na druhej strane pojmy ako graf, vrchol a hrana sa používajú hlavne v matematickej reprezentácii sietí. WWW sa teda dá z matematického hľadiska reprezentovať webovým grafom a spoločnosť zas sociálnym grafom. V tejto sekcii nie je potrebné medzi pojmami rozlišovať, a tak budem používať výlučne pojmy graf, vrchol a hrana. Začnem základnými definíciami, pri ktorých čerpám zo zdroja [38].

Definícia 16. *Neorientovaný graf* je usporiadaná dvojica (V, E), kde V je neprázdna konečná množina vrcholov a E je množina hrán. Hrana je neusporiadaná dvojica vrcholov. Nech $\binom{V}{2}$ je množina všetkých dvojprvkových podmnožín množiny V. Potom platí $E \subseteq \binom{V}{2}$.

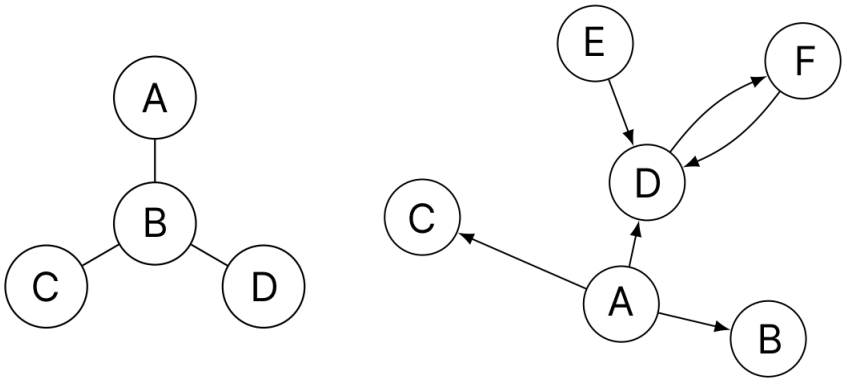
Definícia 17. *Orientovaný graf* je usporiadaná dvojica (V, E), kde V je neprázdna konečná množina vrcholov a E je množina orientovaných hrán. Orientovaná hrana je usporiadaná dvojica vrcholov. Platí $E \subseteq V \times V$.

Pri neorientovanom grafe sú dva vrcholy susedné, pokiaľ medzi nimi existuje hrana. V orientovanom grafe pribúdajú kvôli orientácii hrán nové pojmy. V orientovanej hrane (u, v) sa vrchol u označuje ako predchodca a vrchol v ako následník.

Grafy sa najčastejšie vizualizujú pomocou vykreslenia vrcholov a hrán. Orientované grafy majú pri hranách vyznačený aj smer. Ukážka takýchto vizualizácií je na obrázku 3.19.

3.7.1 Miery centrality

Centralita je metrika, ktorá vrcholu grafu priradí nejaké číslo vyjadrujúce mieru jeho dôležitosti v grafe. Môže pomôcť pri identifikácii vplyvných osôb



Obr. 3.19: Neorientovaný (vľavo) a orientovaný graf (vpravo)

na sociálnych sieťach, dôležitých dopravných uzlov a podobne. Obvykle sa vizualizuje rôznym zafarbením alebo veľkosťou vrcholu. Ukážka je na obrázku 3.20. Dobrým nástrojom na analýzu grafu sú aj vizualizácie distribúcie miery centrality.

Centralitu môžeme merať podľa rôznych kritérií. Medzi najpoužívanejšie patria centralita stupňa (angl. degree centrality), centralita blízkosti (angl. closeness centrality) a centralita medziľahlosti (angl. betweenness centrality). Definície rôznych mier centrality, ktoré uvádzam v nasledujúcich podsekciách sú zo zdroja [39].

Centralita stupňa

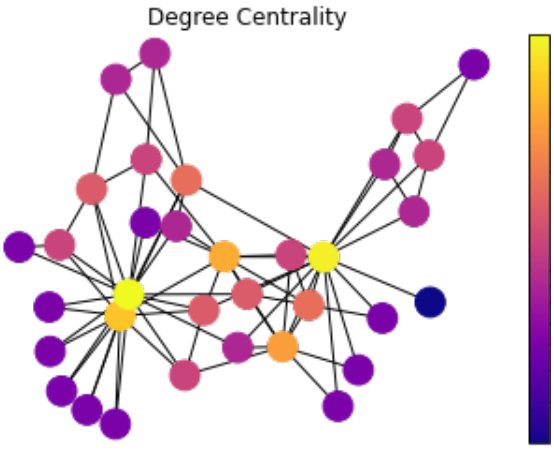
Začnem definíciou stupňa vrcholu pre neorientovaný a orientovaný graf a následne zadefinujem centralitu stupňa pre jeden vrchol a pre celý graf.

Definícia 18. Nech $G = (V, E)$ je neorientovaný graf a $v \in V$ nejaký jeho vrchol. *Stupeň* vrcholu v $deg_G(v)$ je počet hrán grafu G obsahujúcich vrchol v.

Definícia 19. Nech $G = (V, E)$ je orientovaný graf a $v \in V$ nejaký jeho vrchol. *Vstupný stupeň* vrcholu v $deg^+_G(v)$ je počet hrán grafu G končiacich vo vrchole v. *Výstupný stupeň* vrcholu v $deg^-_G(v)$ je počet hrán grafu G vychádzajúcich z vrcholu v. *Stupeň* vrcholu v dostaneme ako súčet vstupného a výstupného stupňa:

$$deg_G(v) = deg^+_G(v) + deg^-_G(v).$$

Pokiaľ je z kontextu jasné o aký graf sa jedná, môžme z označenia stupňa vrcholu graf vynechať (z $deg_G(v)$ sa stane $deg(v)$).



Obr. 3.20: Centralita stupňa vrcholu zobrazená pomocou farby (žltá reprezentuje vysokú a modrá nízku centralitu) [9]

Definícia 20. Nech $G = (V, E)$ je graf a $v \in V$ nejaký jeho vrchol. *Centralita stupňa* pre vrchol v sa značí $C_D(v)$ a je rovná jeho stupňu. Teda:

$$C_D(v) = deg(v).$$

Označme v' vrchol s najväčšou centralitou. *Centralitu stupňa pre celý graf* $C_D(G)$ je potom možné vypočítať nasledovne:

$$C_D(G) = \sum_{i=1}^{|V|} \frac{C_D(v') - C_D(v_i)}{(|V| - 1)(|V| - 2)}.$$

Niekedy je pri orientovaných grafoch vhodné uvažovať aj o vstupnej a výstupnej centralite stupňa. Definície by boli analogické, až na to, že sa v nich namiesto stupňa vrcholu použije vstupný či výstupný stupeň.

Centralita blízkosti

Na definíciu centrality blízkosti je potrebné poznať pojmy súvisiace s vzdialenosťou dvoch vrcholov v grafe.

Definícia 21. Graf V, E je *cesta* P_m dĺžky m (teda cesta s m hranami) práve vtedy, keď platí $V = \{0, ..., m\}$ a $E = \{\{i, i + 1\} | i \in \{0, ..., m - 1\}\}$.

Definícia 22. Podgraf grafu G izomorfný s nejakou cestou P sa nazýva *cesta v grafe* G. Každá cesta má počiatočný vrchol u a koncový vrchol v, a preto sa niekedy nazýva u-v-cesta, alebo cesta z u do v.

Definícia 23. *Vzdialenosť* d_{uv} dvoch vrcholov u a v v grafe G je rovná dĺžke najkratšej u-v-cesty v G. Pokiaľ neexistuje žiadna u-v-cesta, $d_{uv} = \infty$.

Definícia 24. Graf G je *súvislý*, ak v ňom pre každé dva vrcholy u a v existuje u-v-cesta.

Definícia 25. Pre súvislý graf $G = (V, E)$ a jeho vrchol v je možné *centralitu blízkosti* C_C definovať nasledovne:

$$C_C(v) = \frac{1}{\sum_{u \in V} d_{vu}}.$$

Normalizovaná centralita blízkosti C_{Cn} umožňuje porovnávať vrcholy v rôzne veľkých súvislých grafoch a vypočítame ju ako

$$C_{Cn}(v) = \frac{|V| - 1}{\sum_{u \in V} d_{vu}}.$$

Centralita blízkosti teda funguje len pre súvislé grafy. V nesúvislých grafoch by bola pre každý vrchol nulová. Existuje aj harmonická centralita, ktorá tento problém rieši. Analogicky ako pri centralite blízkosti existuje aj jej normalizovaná verzia.

Definícia 26. *Harmonická centralita* C_H pre vrchol v grafu $G = (V, E)$ sa definuje nasledovne:

$$C_H(v) = \sum_{u \in V \setminus \{v\}} \frac{1}{d_{vu}}.$$

Centralita medziľahlosti

O centralite medziľahlosti má zmysel uvažovať len pri súvislých grafoch. Vrcholy s veľkou centralitou sa často vyskytujú na najkratších cestách medzi inými vrcholmi.

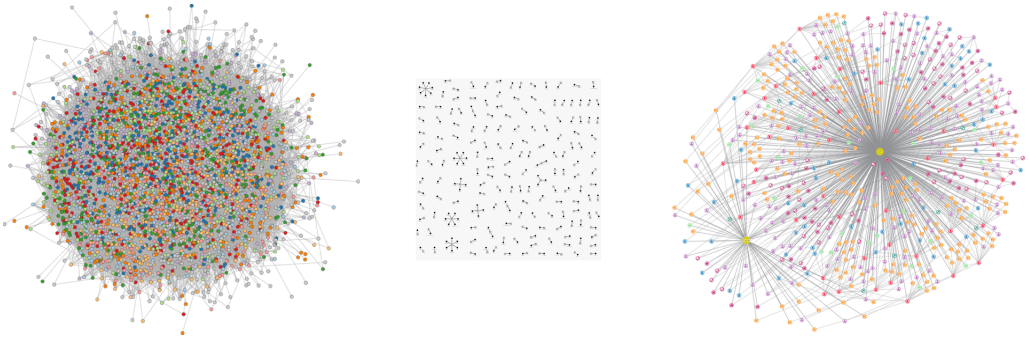
Definícia 27. *Centralita medziľahlosti* C_B pre vrchol V grafu $G = (V, E)$ sa definuje ako

$$C_B(v) = \sum_{s \neq v \neq t \in V} \frac{g_{st}(v)}{g_{st}},$$

kde g_{st} je počet najkratších s-t-ciest a $g_{st}(v)$ je počet najkratších s-t-ciest prechádzajúcich vrcholom v.

3.7.2 Problematické rozloženia grafov a ich úpravy

Existuje niekoľko rozložení grafov, ktoré nemajú príliš veľkú výpovednú hodnotu z pohľadu analýzy dát, ktoré zobrazujú. Medzi tri najčastejšie rozloženia s touto charakteristikou patria príliš hustý graf (angl. hairball), nesúvislý graf (angl. snowstorm) a hviezda (angl. starbursts). Ukážka všetkých troch rozložení je na obrázku 3.21. Postupy, ktoré sa tieto rozloženia snažia mitigovať sú často založené na tom, že si vyberú len podmnožinu hrán, či vrcholov,



Obr. 3.21: Ukážka problematických rozložení grafov. Zľava vidíme príliš hustý graf [10], nesúvislý graf [11] a hviezdu [11]

ktoré zobrazia. Pokiaľ koncový používateľ vizualizácie o filtrovaní nevie, môže dôjsť k nesprávnym záverom. Z hľadiska používateľskej prívetivosti je teda nutné o tom používateľa informovať.

Príliš hustý graf

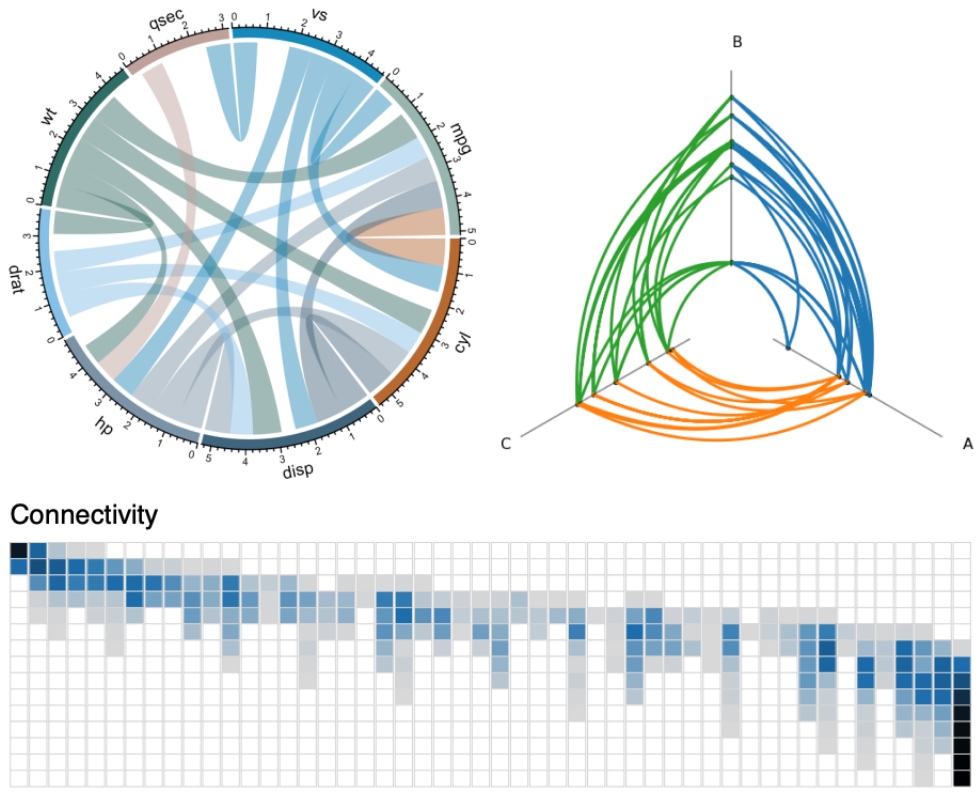
Problémom príliš hustého grafu je, že jeho vizualizácia je veľmi neprehľadná. Obrovský počet hrán zhoršuje viditeľnosť komunít a iných informácií, ktoré od vizualizácie očakávame. Problémom sú dáta, ktoré sa snažíme vizualizovať. Zjednodušene povedané zobrazujeme príliš veľa informácií. Riešenia tohto problému sú preto často založené na filtrovaní nedôležitých hrán.

Pred tým, než začneme príliš hustý graf filtrovať na základe metrík, je vhodné sa zamyslieť nad tým, čo má vizualizovať. V niektorých prípadoch totiž problém vznikol na základe toho, že sa snažíme vykresliť všetky dáta, ktoré máme v datasete a nie nutne len tie, ktoré sú potrebné. Druhá možnosť je združiť niektoré vrcholy podľa sémantiky. Táto možnosť je vhodná hlavne pri interaktívnych vizualizáciách, kde je možné na každý vrchol kliknúť a zobraziť vrcholy, ktoré združuje.

Pokiaľ vyššie uvedené prístupy nepomôžu, či nie sú aplikovateľné, ďalšou možnosťou je filtrovať hrany podľa nejakej metriky. Vhodnými metrikami sú napríklad medziľahlosť, frekvencia či vzájomná informácia hrán [40]. Detailný popis spomenutých techník je nad rámec tejto rešerše. V prípade záujmu je možné zistiť viac z uvedenej referencie.

Nesúvislý graf

Jedná sa o graf, v ktorom existujú len malé neprepojené komunity. Pokiaľ takáto vizualizácia nemá príliš veľkú výpovednú hodnotu, môže byť vhodné dataset doplniť o dáta z iných zdrojov.



Obr. 3.22: Chord diagram (vľavo hore), hive plot [12] (vpravo hore), GraphPrism [13] (dole)

Hviezda

Hviezda je graf, ktorý má jeden vrchol s veľkým stupňom a ostatné vrcholy sú nízkeho stupňa. To, či je vhodné hviezdy z vizualizácie odstrániť alebo nie, záleží od typu otázky, na ktorú sa snažíme odpovedať. Hviezdy je možné odstrániť pomocou filtrácie vrcholov alebo hrán.

Základný spôsob, ktorý často stačí, je odstránenie všetkých hrán spojených s hlavným aktérom (vrcholom v strede hviezdy). Informáciu, ktorú pred tým zobrazovala hrana môžeme zobraziť zafarbením alebo zväčšením incidentného vrcholu. Iná stratégia je postavená na filtrovaní najmenej dôležitých vrcholov. Dôležitosť vrcholu je možné odhadnúť podľa niektorej z centralít, ktoré som definovala v tejto kapitole.

3.7.3 Alternatívne vizualizácie grafov

Aj keď vizualizácia grafu pomocou vykreslenia jeho hrán a vrcholov ľubovoľne v priestore je najčastejšia, existujú aj alternatívne spôsoby. Príčinou vzniku týchto spôsobov je často nevhodnosť klasického prístupu pri zobrazení určitého

typu dát. Niektoré sa zameriavajú na lepšiu klasifikáciu vrcholov do kategórií, iné sa snažia vyriešiť vizualizáciu príliš hustého grafu pomocou vizualizácie metrík jeho vrcholov. Alternatívne spôsoby vizualizácie sú ukázané na obrázku 3.22 a patrí medzi ne napríklad:

- chord diagram - vrcholy sú usporiadané v kruhu a hrany medzi nimi tvoria určitý vzor,
- hive plot [12] - vrcholy sú usporiadané na lineárnych osách, ktoré sú radiálne orientované,
- GraphPrism [13] - technika, ktorá zobrazuje určitú metriku vrcholov v 2D matici.

3.8 Vizualizácie v strojovom učení

Vizualizácie hrajú pri strojovom učení veľkú rolu. Môžeme vďaka nim lepšie pochopiť dáta, s ktorými pracujeme ale aj výsledné modely, ktoré sú na nich natrénované. Doposiaľ som sa v tejto kapitole zaoberala vizualizáciami dát. Či už šlo o exploračnú analýzu kvalitatívnych, kvantitatívnych dát a vzťahov medzi nimi alebo o komplexnejšie dátové štruktúry ako sú napríklad obrázky, texty, časové rady či grafy. V poslednej sekcii tejto kapitoly sa budem venovať vizualizáciám, ktoré súvisia s modelmi. Ukážem ako sa dá vizualizovať samotný model, jeho úspešnosť a vplyv, ktorý na ňu majú rôzne kombinácie hyperparametrov. Informácie čerpám zo zdroja [41].

3.8.1 Výsledný model

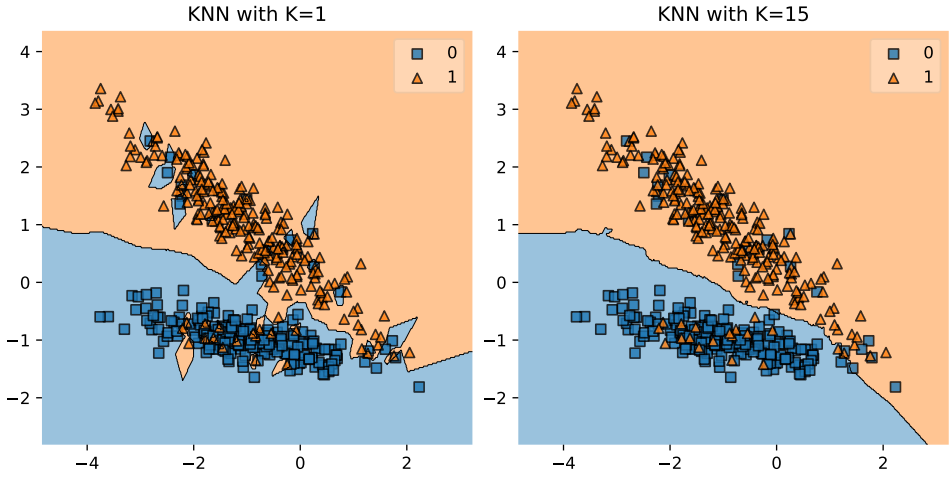
Vďaka vizualizácii výsledného modelu vieme lepšie rozumieť tomu, podľa čoho sa model rozhoduje a niekedy dokážeme detekovať jeho preučenie. Niektoré typy vizualizácie architektúry modelu sa dajú aplikovať na viaceré modely, iné sú špecifické pre daný model. Začnem ukážkou vizualizácie, ktorá sa dá aplikovať na ľubovoľný klasifikátor a potom ukážem vizualizácie špecifické pre rozhodovací strom a neurónovú sieť.

Všeobecný klasifikátor

Ľubovoľný klasifikátor, ktorý predikuje na základe maximálne troch príznakov vieme vizualizovať pomocou jeho rozhodovacích oblastí (angl. decision regions). Aby bolo možné zadefinovať rozhodovaciu oblasť, je najskôr potrebné vysvetliť pojem priestor vstupov (angl. input space).

Definícia 28. Majme model natrénovaný na n príznakoch $p_1, p_2, ..., p_n$. Nech $D_1, D_2, ..., D_n$ sú domény týchto príznakov. Pod pojmom *priestor vstupov* rozumieme množinu všetkých valídnych vstupov daného modelu:

$$D = D_1 \times D_2 \times \cdots \times D_n.$$



Obr. 3.23: Rozhodovacie oblasti kNN pre k=1 (vľavo) a k=15 (vpravo)

Definícia 29. Pre klasifikátor predikujúci na základe n príznakov, je *rozhodovacia oblasť* n-dimenzionálny objekt v priestore vstupov, pre ktorý klasifikátor predikuje jednu triedu.

Ukážka takejto vizualizácie pre kNN klasifikátor predikujúci na základe dvoch príznakov je na obrázku 3.23. Na obrázku sú vizualizácie dvoch kNN klasifikátorov. Klasifikátor vľavo sa rozhoduje na základe jedného najbližšieho suseda. Jeho rozhodovacia hranica (tzn. hranica medzi jeho rozhodovacími regiónmi) je príliš komplikovaná. Klasifikátor vpravo sa rozhoduje na základe väčšieho počtu susedov a je vidieť, že sa vďaka tomu rozhodovacia hranica vyhladila. Z vizualizácií teda môžeme povedať, že model predikujúci na základe jedného suseda je preučený.

Pokiaľ by sme tento typ vizualizácie chceli použiť na klasifikátor, ktorý predikuje na základe troch príznakov, je nutné využiť animáciu, v ktorej sa postupne mení hodnota tretieho príznaku alebo interaktívne prvky.

Rozhodovací strom

Vizualizácie rozhodovacieho stromu sú založené na reprezentácii pravidiel, na základe ktorých sa rozhoduje. Okrem pravidiel sa často zobrazuje, koľko pozorovaní spadá pod každý vrchol a v prípade listov aj výsledná predikcia. Presná podoba vizualizácie sa líši od konkrétneho balíčku, ktorý sa na vizualizáciu použije. V prílohách D.1 a D.2 sú ukážky vizualizácie klasifikačného rozhodovacieho stromu pomocou scikit-learn [20] a dtreeviz [21]. Zatiaľ čo scikit-learn zobrazuje vyššie spomínané informácie v textovej podobe, dtreeviz ponúka detailnejšiu vizualizáciu. Deliace vrcholy sú zobrazené pomocou histogramu príznaku, na základe ktorého sa pozorovania vo vrchole budú deliť. Histogram je rozdelený farebne do tried a je na ňom vyznačený deliaci bod. Listy

sú zobrazené pomocou koláčového grafu tried pozorovaní, ktoré sú v danom liste obsiahnuté. Oba balíčky ponúkajú aj variant vizualizácie pre regresné rozhodovacie stromy.

Neurónové siete

Pri vizualizácii neurónovej siete je vhodné zobraziť neuróny v jednotlivých vrstvách a smer prúdenia dát. S narastajúcou zložitosťou siete narastá aj zložitosť vizualizácie, a preto je v niektorých prípadoch vhodné zobraziť len konceptuálny model, ktorý má jeden vrchol na každú vrstvu. Vizualizácia neurónových sietí je vo všeobecnosti komplexná záležitosť a pre tento účel vzniklo viacero nástrojov (napr. Tensorboard [42], Weights & Biases [43], ANN Visualizer [44]).

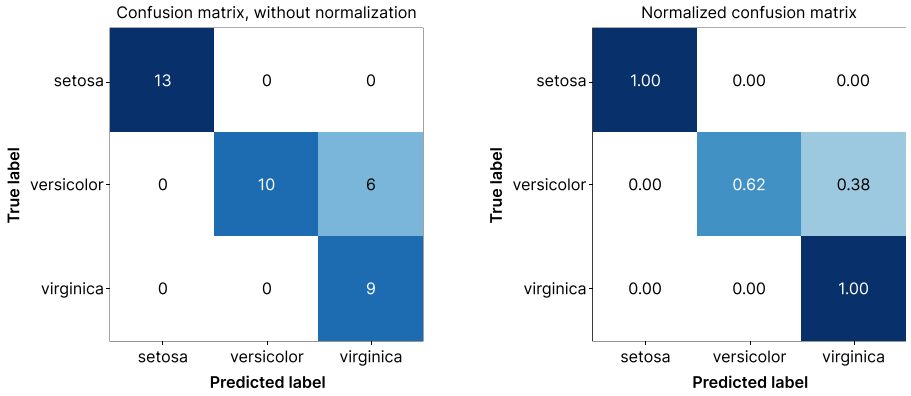
3.8.2 Úspešnosť modelu

Existuje niekoľko metrík úspešnosti modelu, ktorých použitie závisí na type úlohy (tzn. či sa jedná o klasifikáciu alebo regresiu) a ďalších atribútoch ako je napríklad vyváženosť tried pri klasifikácii či početnosť outlierov pri regresii. V tejto sekcii predstavím základné metriky úspešnosti pre klasifikáciu a regresiu a ich časté vizualizácie. Pri klasifikácii vychádzam zo zdroja [45] a pri regresii zo zdroja [46].

Klasifikácia

Pri binárnej klasifikácii je úlohou klasifikátoru predikovať, či pozorovanie p spadá do pozitívnej triedy (hodnota vysvetľovanej premennej je 1), alebo negatívnej triedy (hodnota vysvetľovanej premennej je 0). Pri klasifikovaní pozorovania p môžu vzhľadom k vysvetľovanej premennej nastať štyri prípady:

- skutočná pozitivita (angl. true positive)
  - skutočná hodnota je 1, predikovaná hodnota je 1
- falošná pozitivita (angl. false positive)
  - skutočná hodnota je 0, predikovaná hodnota je 1
- skutočná negativita (angl. true negative)
  - skutočná hodnota je 0, predikovaná hodnota je 0
- falošná negativita (angl. false negative)
  - skutočná hodnota je 1, predikovaná hodnota je 0



Obr. 3.24: Klasická (vľavo) a normalizovaná (vpravo) matica zámen

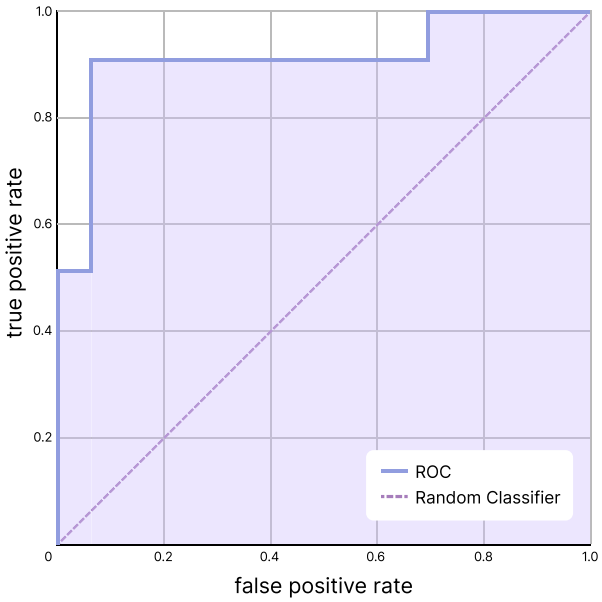
Matica zámen patrí k základným nástrojom na evaluáciu úspešnosti klasifikátora. Predpokladajme, že máme problém binárnej klasifikácie a klasifikátor vykonal predikciu vysvetľovanej premennej pre niekoľko pozorovaní. Označme v poradí TP, FP, TN a FN počet pozorovaní, ktoré boli skutočne pozitívne, falošne pozitívne, skutočne negatívne a falošne negatívne. Matica zámen je potom matica v nasledujúcom formáte:

$$\begin{pmatrix} TP & FP \\ FN & TN \end{pmatrix}.$$

Niekedy sa matica zámen zobrazuje aj v normalizovanom tvare (tzn. namiesto počtu sa zobrazuje percentuálny pomer). Je možné ju tiež zovšeobecniť a použiť pri klasifikácii do viacerých tried. Pre triedy $1, 2, ..., n$ sa bude jednať o maticu s rozmermi $n \times n$. V i-tom riadku a j-tom stĺpci bude takáto matica obsahovať počet (resp. percento) pozorovaní s predikciou i a skutočnou hodnotou j. Ukážka klasickej a normalizovanej verzie matice zámen pre klasifikáciu do viacerých tried je na obrázku 3.24. Ak si zadefinujeme N ako počet všetkých pozorovaní, z matice zámen sa dajú odvodiť ďalšie používané metriky ako je napríklad:

- presnosť (angl. accuracy) - $\frac{TP+TN}{N}$
- TPR (skratka z angl. true postive rate) - $\frac{TP}{TP+FN}$
- FPR (skratka z angl. false postive rate) - $\frac{FP}{FP+TN}$

TPR a FPR sa používajú pri druhej známej vizualizácii úspešnosti klasifikátoru, ktorá sa nazýva ROC krivka (angl. ROC curve). Zamerajme sa na binárnu klasifikáciu, kde model predikuje pravdepodobnosť, že pozorovanie x patrí do pozítivnej triedy (tzn. vysvetľovaná premenná je 1). Označme túto



Obr. 3.25: ROC krivka

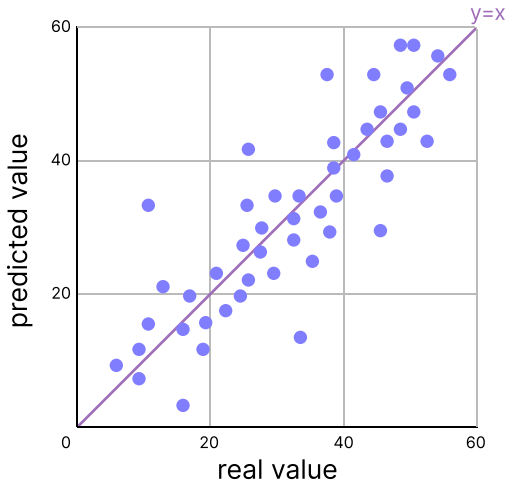
pravdepodobnosť p_x. Prirodzená predikcia vysvetľovanej premennej $\hat{Y}$ by potom vyzerala takto:

$$\hat{Y} = \begin{cases} 1, & \text{ak } p_x \geq 0,5 \\ 0, & \text{inak.} \end{cases}$$

To znamená, že pokiaľ je p_x väčšie ako 0,5, model bude predikovať pozitívnu triedu. Namiesto konštanty 0,5 ale môžeme zvoliť ľubovoľné číslo $\tau \in [0, 1]$. Pre každú hodnotu τ je možné vypočítať TPR a FPR. ROC krivka je graf, ktorý porovnáva FPR (os x) a TPR (os y) ako implicitnú funkciu τ. Jej ukážka je na obrázku 3.25. V ideálnom prípade krivka vystúpa veľmi rýchlo do ľavého horného rohu. Pokiaľ je krivka blízko priamke $y = x$, úspešnosť klasifikátoru je podobná náhodnej predikcii.

Regresia

Regresor predikuje hodnotu spojitej vysvetľovanej premennej. Každé pozorovanie má skutočnú hodnotu Y a predikovanú hodnotu $\hat{Y}$. Keď chceme vizualizovať úspešnosť regresoru, je možné skonštruovať graf porovnávajúci skutočné hodnoty (os x) a predikované hodnoty (os y) zobrazený na obrázku 3.26. Priamka $y = x$ reprezentuje dokonalý regresor, ktorý vždy predikuje správnu hodnotu. Čím ďalej je bod od tejto priamky, tým väčšiu chybu spravil regresor pri predikcii daného pozorovania.



Obr. 3.26: Graf skutočných a predikovaných hodnôt

Medzi známe metriky na evaluáciu regresoru patria napríklad MSE, RMSE, RMSLE či MAE (skratky sú vysvetlené v zozname skratiek). Vzorce na ich výpočet sú nasledovné (N vyjadruje počet pozorovaní):

$$\text{MSE} = \frac{1}{N}\sum_{i=1}^{N}(Y_i - \hat{Y}_i)^2$$

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(Y_i - \hat{Y}_i)^2}$$

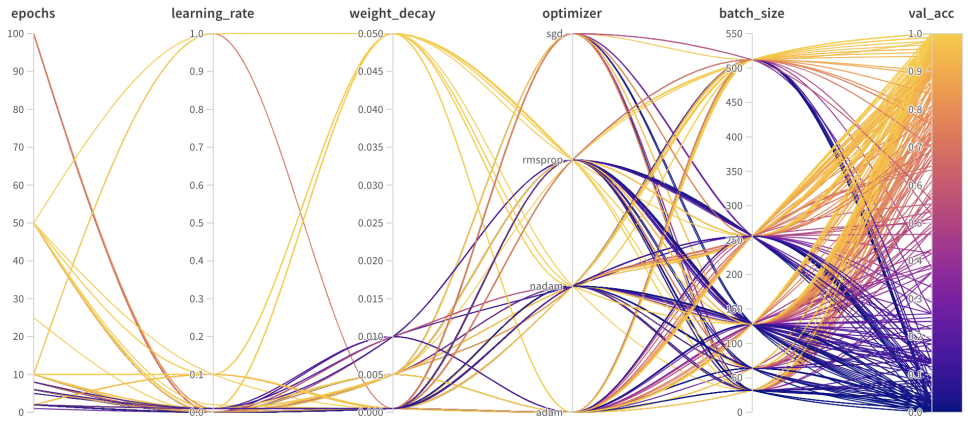
$$\text{RMSLE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(logY_i - log\hat{Y}_i)^2}$$

$$\text{MAE} = \frac{1}{N}\sum_{i=1}^{N}|Y_i - \hat{Y}_i|$$

Vhodnou vizualizáciou úspešnosti regresoru je vizualizácia distribúcie chyby. Postupujeme tak, že zvolíme metriku a vykreslíme histogram, box plot alebo iný graf na vizualizáciu distribúcie. Obvykle sa porovnáva distribúcia chyby vzhľadom k trénovacím, validačným a testovacím dátam.

3.8.3 Ladenie hyperparametrov

Modely obvykle majú určité hyperparametre, ktoré je nutné zvoliť pred ich trénovaním. Voľbu konkrétnej kombinácie hyperparametrov zakladáme na validačnej úspešnosti modelu, ktorý bol s nimi vytvorený. Vizualizácia hodnoty



Obr. 3.27: Graf paralelných súradníc zobrazujúci úspešnosť viacerých kombinácií hyperparametrov [14]

metriky na trénovacej a validačnej množine vzhľadom k voľbe hyperparametrov nám môže pomôcť rýchlo identifikovať, ktorá kombinácia je najlepšia a v niektorých prípadoch vďaka nej vieme určiť vplyv konkrétnych hyperparametrov na výslednú úspešnosť. V prílohe D.3 je ukážka vývoja presnosti klasifikačného rozhodovacieho stromu vzhľadom k rastúcej hĺbke. Pri väčšom počte hyperparametrov je možné vytvoriť pole všetkých kombinácií a použiť rovnaký graf. Na osi x budeme zobrazovať index v poli hyperparametrov. Takáto vizualizácia dáva zmysel len pre rozumný počet kombinácií hyperparametrov. V prípade, že je ich viac, je možné využiť graf paralelných súradníc, ktorý je zobrazený na obrázku 3.27.

Druhá možnosť, ktorá môže v niektorých prípadoch poskytnúť viac informácií je vizualizácia výsledného modelu, o ktorej som písala v sekcii 3.8.1. Základný postup je vybrať niektoré kombinácie hyperparametrov a vykresliť graf pre každú z nich. Tento postup je zobrazený na obrázku 3.23, ktorý porovnáva výsledný kNN model pre dve voľby hyperparametru k. Vzhľadom k tomu, že máme väčšinou k dispozícii veľké množstvo kombinácií hyperparametrov, je často vhodné využiť interaktívne prvky.

Kapitola 4

Výber metód s ohľadom na predpokladané znalosti študentov

V tejto kapitole vytvorím odhad predpokladaných znalostí študentov predmetu BI-VIZ. Následne budem hodnotiť analyzované metódy z kapitoly 3 pre každý tematický celok. Pri hodnotení ma bude zaujímať, či sa metódy dajú vysvetliť bez predchádzajúcich znalostí, alebo je nutné mať znalosti v nejakej špecifickej oblasti. Pokiaľ sú nutné predchádzajúce znalosti, je potrebné určiť, či ich študenti budú mať pred začiatkom semestra alebo ich nadobudnú až počas semestra. V druhom prípade je dôležité správne načasovanie prednášok.

4.1 Predpokladané znalosti študentov

Predmet BI-VIZ bude povinný predmet bakalárskej špecializácie Umelá inteligencia a je doporučené si ho zapísať v treťom semestri štúdia. Predpokladané znalosti študentov som preto odhadla podľa doporučeného priechodu štúdiom pre špecializáciu Umelá inteligencia.

Na začiatku tretieho semestra by študenti, okrem iného, mali mať za sebou predmety so zameraním na programovanie a algoritmizáciu, unixové operačné systémy, verzovacie systémy, matematickú analýzu a lineárnu algebru. Z toho sa dá predpokladať, že budú vedieť programovať, ovládať terminál, pracovať s fakultným GitLabom a čítať matematické notácie.

V treťom semestri by mali študenti spolu s predmetom BI-VIZ študovať ďalšie relevantné predmety ako sú Pravdepodobnosť a štatistika, Algoritmy a grafy 1 a Strojové učenie 1. Z toho sa dá predpokladať, že sa študenti v priebehu semestra zoznámia s teóriou pravdepodobnosti a štatistiky, teóriou grafov a základmi strojového učenia.

4.2 Zhodnotenie analyzovaných metód

Prvé dve sekcie kapitoly 3 tvoria úvod do vizualizácie dát a nepotrebujú k pochopeniu žiadne predchádzajúce znalosti. Sekcie sa zaoberajú vizualizačným procesom, princípmi, ktoré je treba dodržiavať pri tvorbe vizualizácie, typmi dát a vizuálnych premenných, metrikami, na základe ktorých vieme povedať, ktorá z dvoch vizualizácií je lepšia a častými chybami, ktoré spôsobia, že vzniknutá vizualizácia je zavádzajúca. Všetky spomínané informácie sú dôležité pre tvorbu korektnej a zaujímavej vizualizácie, takže by mali byť súčasťou predmetu.

Tretia sekcia sa venuje exploračnej analýze dát. Sústredila som sa na základné metriky a nezachádzala som do pokročilých tém. Preto si myslím, že by bolo vhodné vyučovať všetky analyzované metódy. Zložitejšie na pochopenie môžu byť napr. pojmy kvantil a korelácia. Oba pojmy budú exaktne vysvetlené v predmete Pravdepodobnosť a štatistika, a preto si môžeme dovoliť ich vysvetliť zjednodušene pre potreby vizualizácií.

Tematický celok na vizualizáciu textových dát vysvetľuje základné metódy NLP, ktoré je možné vysvetliť bez predchádzajúcich znalostí. Jedinou pokročilou témou, ktorá bola spomenutá je analýza sentimentu. Tá sa buď nemusí preberať, alebo môže byť vysvetlená len rámcovo.

Ďalšou témou bola vizualizácia obrázkov. Analyzovala som reprezentáciu digitálnych obrázkov, farebné modely a operácie s obrázkami. Vzhľadom k tomu, že operácie sa vykonávajú pomocou algebraických operácií s maticami, je vyžadovaná znalosť lineárnej algebry. Tieto znalosti by študenti mali mať pred začiatkom semestra, takže do vyučovania je možné zahrnúť všetky metódy.

Časové rady resp. grafy a siete sú tematické celky, ktoré vyžadujú znalosti z pravdepodobnosti a štatistiky resp. teórie grafov. Budú preto vyučované v druhej polovici semestra, aby študenti už mali potrebné znalosti.

Posledným tematickým celkom boli vizualizácie v strojovom učení. Analyzovala som metódy na vizualizáciu výsledných modelov (klasifikátorov aj regresorov), evaluácie modelov a procesu ladenia hyperparametrov. Z osnovy predmetu Strojové učenie 1 vyplýva, že evaluácia modelov a ladenie hyperparametrov sa začnú preberať až v deviatom týždni semestra. Bolo by preto vhodné sa týmito témami zaoberať paralelne s tým ako sa budú preberať v predmete Strojové učenie 1.

Kapitola 5

Metódy vyučovania

Vyučovací proces pozostáva z komunikácie medzi vyučujúcim a študentom. Táto pedagogická komunikácia sa skladá z dvoch procesov, a to z vyučovania (činnosť pedagóga) a učenia sa (činnosť študenta). Vyučovanie by teda malo byť naplánované tak, aby čo najviac podporilo proces učenia sa. K tomuto účelu môžu slúžiť rôzne vyučovacie metódy, organizačné formy a materiálne pomôcky [47].

Každý z nás si v živote prešiel nejakou formou vzdelávania, takže s rôznymi typmi vyučovania a učenia sa máme vlastnú skúsenosť. Nakoľko je každý človek jedinečný svojim zmýšľaním a tým pádom aj spôsobom ako sa učí, neexistuje jedna optimálna stratégia na najefektívnejší spôsob vyučovania. Niektorí ľudia sa radšej učia sami, iní preferujú učenie sa v skupine. Niekomu stačia skriptá, niekto zas potrebuje audiovizuálne materiály.

V tejto kapitole sa budem venovať dvom formám vyučovania. Najskôr popíšem tradičné vyučovanie, ktoré je dodnes najčastejšou formou vyučovania. Jedná sa o vyučovanie výkladovou formou, kde študenti pasívne prijímajú nové vedomosti. V ďalšej sekcii potom vysvetlím, čo je to aktívne vyučovanie a v čom sa líši od toho tradičného. Zjednodušene môžme povedať, že aktívne učenie sa je všetko, čo robí študent na hodine (alebo mimo vyučovania) okrem pasívneho počúvania a čítania. Niekoľko štúdií [48][49][50] sa zhoduje na tom, že aktívne vyučovanie je pre väčšinu študentov efektívnejšie ako to tradičné. Je ale dobré poznamenať, že existujú aj štúdie [51], ktoré sa dostali k opačnému záveru. Toto tvrdenie teda nie je všeobecne prijímané.

Zo záverov už spomínaných štúdií, rešerší foriem vyučovania a vlastnej skúsenosti pre mňa vyplynulo, že aktívne vyučovanie môže pozitívne ovplyvniť proces učenia sa. Preto som sa rozhodla niektoré metódy aktívneho vyučovania zahrnúť aj do výuky predmetu BI-VIZ. V poslednej sekcii tejto kapitoly sa budem venovať popisu metód výuky, ktoré sa budem snažiť aplikovať v predmete BI-VIZ.

Obr. 5.1: Maľba zobrazujúca stredovekú prednášku na Università di Bologna [15]

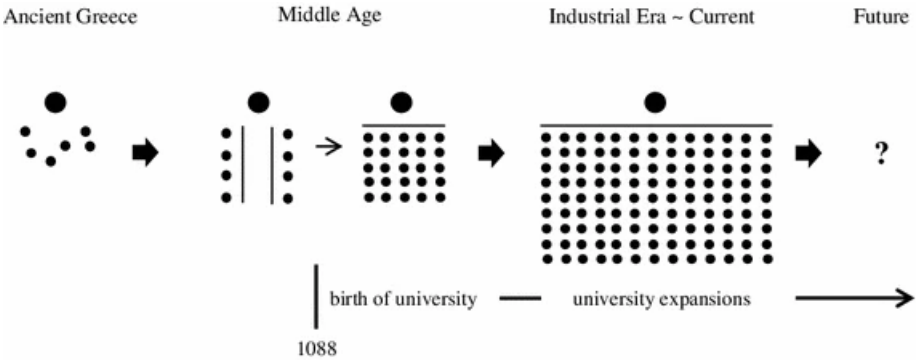
5.1 Tradičné vyučovanie

Podľa zdroja [16] je tradičné vyučovanie založené na princípoch, ktoré sa začali používať v stredovekých univerzitách. To zahŕňa vyučovacie metódy, ale aj dizajn učební. Od stredovekých čias sa toho ale dosť zmenilo a kritici tohto formátu výuky argumentujú práve tým, že sa na zmeny dostatočne neadaptoval. Adaptácia spôsobu výuky pritom nie je nič nezvyčajné a v dejinách pedagogiky nastala už niekoľko krát.

5.1.1 História

Táto sekcia je voľnou parafrázou zdroja [16].

Dizajn učební a metódy vyučovania sa v priebehu času menili. V starovekom Grécku napríklad vôbec neexistoval formálny edukačný systém. Výuka prebiehala formou dialógu medzi vyučujúcim a študentami. Čo sa týka priestorového usporiadania, taktiež neexistovali žiadne pravidlá. Vyučujúci a študenti boli usadení v nepravidelnom vzore tak, ako to dovoľoval dostupný priestor.



Obr. 5.2: Vývoj dizajnu učební [16]

Ak sa časovo posunieme do stredoveku, už sa učilo v usporiadanejšom priestore. Stoly boli radené do dvoch rovnobežných čiar a študenti sedeli tak, aby každý na každého videl. Takýto spôsob sa v tej dobe používal aj počas bohoslužieb v kláštoroch.

Neskôr v tej dobe vznikli prvé univerzity a zvýšil sa záujem o vzdelávanie sa, čo spôsobilo vyšší počet študentov na hodine a bolo potrebné znovu zmeniť priestorové usporiadanie. Nové usporiadanie a vlastne aj štýl výuky zobrazil vo svojej maľbe taliansky maliar Laurentius de Voltolina. Jej ukážka je na obrázku 5.1. Môžeme na nej vidieť niekoľko radov stolov usporiadaných rovnobežne za sebou a vyučujúceho, ktorý sedí pred týmito radmi a číta knihu. Keďže knihy a papier boli v tejto dobe vzácne, hlavnou úlohou prednášok bolo predať informácie z nejakej rešpektovanej knihy formou čítania.

V priemyselnej dobe začalo byť vzdelávanie dostupné aj širokej verejnosti a prednáškové miestnosti sa niekoľkonásobne zväčšili, aby pokryli vyšší dopyt. Historický vývoj dizajnu učební si môžete pozrieť na obrázku 5.2. Dizajn učební a výuka formou prednášania vedomostí z nejakých zdrojov zostala viac menej nezmenená dodnes.

5.1.2 Nevýhody tradičného vyučovania

Už som spomínala, že tradičné vyučovanie je vyučovanie výkladovou formou, kde študenti pasívne prijímajú nové vedomosti a väčšinou sú usadení do niekoľkých rovnobežných radov za sebou. Ide teda o úplne klasický formát výuky, na ktorý sme zvyknutí zo stredných škôl, či prednášok na univerzitách. Nejde ale iba o formu ale aj o obsah výuky. Prednášajúci vysvetlí tému a jej všeobecné princípy. Následne predvedie model, ukáže jeho aplikáciu na príklade, dá študentom úlohy s podobnými príkladmi aplikácie a nakoniec testom overuje schopnosť študentov urobiť to isté [52].

Z toho vyplýva hneď niekoľko nevýhod. Prvou z nich je, že študenti majú problém s pozornosťou. V súčasnej dobe sme zahltení množstvom informácií

a pre študentov je veľmi jednoduché prestať dávať pozor, pokiaľ majú len pasívne počúvať niekoho výklad [53]. Taktiež už je jednoduché si na internete vyhľadať informácie k ľubovoľnej téme, a preto forma čítania/prednášania pravdivých informácií už nie je tak potrebná ako kedysi. Druhý problém spočíva v tom, že niektorí študenti sú hanbliví a v prednáškovej miestnosti, kde môže byť viac ako sto ľudí majú problém vyjadriť svoj názor, alebo sa spýtať pokiaľ niečomu nerozumejú. Niektoré zdroje [16] tvrdia, že je problém aj v dizajne učební. Môžu totiž existovať dobré a zlé zóny, ktoré diskriminujú schopnosť študenta sa niečo naučiť na základe toho, kde je v miestnosti usadený. Posledný problém, ktorý spomeniem sa týka toho, čo sa často v tradičnom štýle výuky hodnotí. Úlohy a testy, ktorých cieľom je zopakovať naučený postup na podobnom probléme vedú na memorovanie postupu [52] a nepripravia študentov na praktické problémy, ktoré sa im stanú v profesionálnej kariére.

5.2 Aktívne vyučovanie

Táto sekcia je voľnou parafrázou zdrojov [52] a [54].

Na začiatku tejto kapitoly som aktívne učenie zjednodušene popísala ako všetko, čo robí študent na hodine (alebo mimo vyučovania) okrem pasívneho počúvania a čítania. Táto definícia je dobrá na vytvorenie základnej predstavy, ale nie vždy je postačujúca. Presnejšie aktívne učenie pozostáva z krátkych individuálnych alebo skupinových aktivít, na ktorých participujú všetci študenti a ktoré sa týkajú preberanej látky. Po takejto aktivite spravidla prichádza fáza, kde študenti majú možnosť prezentovať výsledky a vyučujúci ich zhodnotí a prípadne môže doplniť ďalšie informácie.

Aké sú vhodné aktivity? Záleží len na kreativite vyučujúceho. Študenti môžu odpovedať na otázku, vysvetľovať nejaký zložitejší koncept, vypočítať príklad, nakresliť diagram, vymyslieť otázku k práve preberanej látke. Pri návrhu aktivity je potrebné myslieť na jej predpokladanú dĺžku trvania a vyvarovať sa nasledujúcim chybám:

- triviálne skupinové aktivity,
- príliš dlhé skupinové aktivity,
- vyvolávanie dobrovoľníkov po každej aktivite.

Prečo sa jedná o chyby? Pokiaľ sa vyučujúci spýta otázku, na ktorú väčšina študentov vie odpoveď hneď, vytváranie skupín zaberá zbytočný čas. Skupinová aktivita by mala byť dostatočne náročná na to, aby študenti zo skupinovej práce benefitovali.

Pokiaľ je naopak aktivita príliš náročná (napr. odhadovaný čas je desať minút), zvýraznia sa rozdiely medzi jednotlivými skupinami. Najrýchlejšia skupina môže skončiť po minúte a študenti sa začnú nudiť a tá najpomalšia

skupina to nestihne ani za desať minút a študenti z toho budú frustrovaní. Ideálna dĺžka aktivity je do troch minút.

Poslednou chybou bolo vyvolávanie dobrovoľníkov. Pokiaľ po každej aktivite vyučujúci vyvolá dobrovoľníkov, študenti rýchlo zistia, že v aktivite nemusia participovať. Pokiaľ naopak vedia, že po každej aktivite to môžu byť práve oni, ktorí budú musieť prezentovať odpoveď, angažovanosť v úlohe bude vyššia.

5.3 Výuka predmetu BI-VIZ

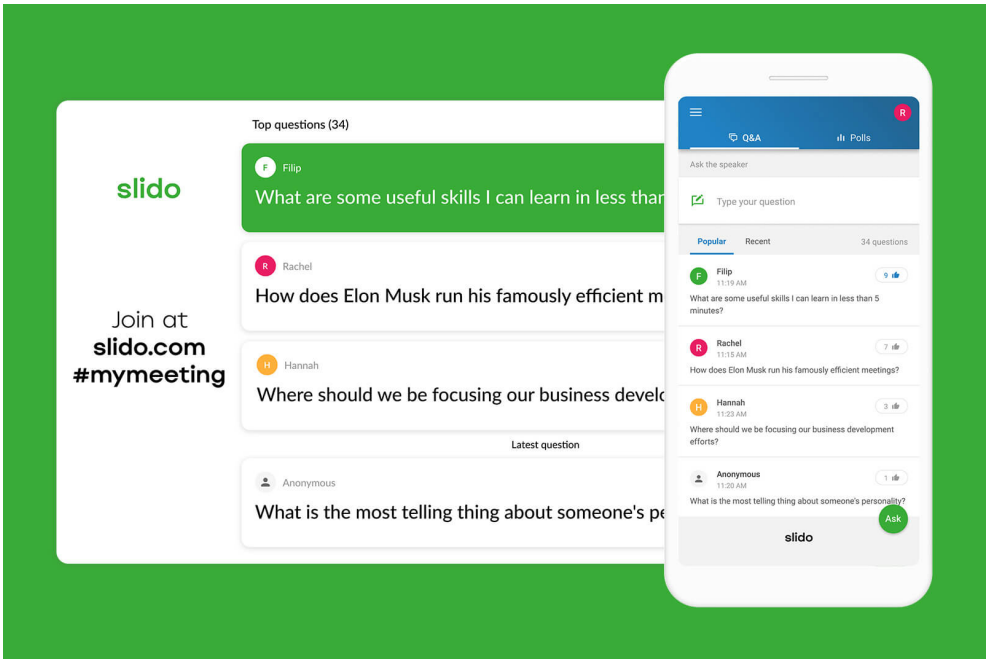
Predmet BI-VIZ bude vyučovaný vo veľkej prednáškovej miestnosti v trojhodinových blokoch. Jeho kapacita sa odhaduje na 100 ľudí. Uplatňujú sa teda niektoré nevýhody, ktoré som v tejto kapitole už spomínala. Študenti sa môžu hanbiť vyjadrovať svoj názor pred tak veľkou skupinou ľudí a budú existovať odľahlé časti miestnosti, ktoré môžu negatívne vplývať na pozornosť študentov a teda ich schopnosť sa efektívne naučiť látku. Tieto nevýhody je možné aspoň čiastočne mitigovať použitím vhodného nástroja. Ja som sa rozhodla využiť nástroj Slido [17], ktorý zároveň pomôže s aplikáciou metód aktívnej výuky.

5.3.1 Slido ako pomôcka pri výuke

Slido [17] je nástroj na zefektívnenie komunikácie medzi prednášajúcim a poslucháčmi. Väčšinou sa používa na konferenciách či firemných meetingoch ale dá sa využiť aj pri vzdelávaní. Ako je možné vidieť na obrázku 5.3 nástroj sa skladá z webovej aplikácie, ktorú používa prednášajúci a mobilnej aplikácie, ktorú používajú hlavne poslucháči.

Medzi hlavné funkcionality patria otázky a odpovede (angl. Questions and Answers skrátene Q&A), kvízy a ankety. Vďaka Q&A môžu poslucháči počas celej doby trvania udalosti písať otázky pre prednášajúceho. Taktiež je možné lajkovať otázky od ostatných. Výhodou je, že prednášajúci nie je otázkami rušený a sám si môže vybrať čas, kedy na ne zodpovie. Táto funkcionalita môže pomôcť hanblivým študentom, pretože otázku stačí napísať a je možné sa pýtať aj anonymne. Taktiež môže pomôcť aj študentom v zadných radách, ktorí nebudú musieť otázku niekoľko krát opakovať, pretože ich nie je dostatočne počuť.

Kvízy a ankety si prednášajúci pripraví vopred. Počas udalosti ich môže aktivovať a zas deaktivovať. Odpovedať sa dá len v čase, keď je kvíz či anketa aktívna. Výhodou Slida je, že sa odpovede zaznamenávajú okamžite a vizualizácia odpovedí sa neustále aktualizuje. Tieto funkcionality nám umožnia aplikovať metódy aktívneho vyučovania aj v tak veľkom počte študentov. Môžme vďaka nim rozdeliť tri hodiny prednášania do menších blokov oddelených aktívnou prácou študentov, po ktorej bude nasledovať diskusia nad



Obr. 5.3: Q&A sekcia aplikácie Slido [17]

výsledkami. Ankety môžu slúžiť aj na zbieranie spätnej väzby, ktorá bude hlavne v prvých behoch predmetu veľmi dôležitá.

5.3.2 Hodnotenie predmetu

Predmet bude pravdepodobne ukončený klasifikovaným zápočtom, čo znamená, že študent bude hodnotený podľa svojej práce počas semestra. Body sa budú dať získať v troch domácich úlohách, z ktorých jedna bude skupinová. Vďaka skupinovej domácej úlohe budú študenti nútení sa o preberanej látke rozprávať a navzájom si vysvetľovať problematické časti. Domáce úlohy budú koncipované tak, aby mali študenti čo najväčšiu voľnosť a tým pádom museli pri vypracovaní premýšľať. Takéto úlohy sú síce náročnejšie, ale študenti sa vďaka nim naučia viac ako keby mali len opakovať naučený postup.

Kapitola 6

Tvorba študijných materiálov

Navrhnuté študijné materiály predmetu BI-VIZ pozostávajú z prezentácií, praktických ukážok a zadaní samostatných prác. Táto kapitola sa venuje analýze doposiaľ použitých nástrojov a predstaveniu navrhnutých materiálov.

6.1 Použité nástroje

Pred tým ako som začala vytvárať študijné materiály, bolo potrebné určiť, aké nástroje sa budú v predmete využívať. Najdôležitejšia bola voľba vývojového prostredia, ktoré sa bude používať v praktických ukážkach a samostatných prácach a voľba grafického balíčka, pomocou ktorého sa bude tvoriť väčšina vizualizácií. Všetky rozhodnutia bolo treba spraviť s ohľadom na to, že sa v predmete bude používať programovací jazyk Python.

6.1.1 Jupyter Notebook

Jupyter Notebook [2] je nástroj na vytváranie zošitov, v ktorých je možné kombinovať mnoho elementov vrátane kódu, matematických notácií, naratívneho textu, obrázkov či vizualizácií. Vďaka tejto vlastnosti sú zošity vhodné na praktické ukážky, kde je možné slovne a pomocou obrázkov vysvetliť preberanú látku a pridať ukážku v podobe kódu. Vzhľadom k tomu, že kurz vysvetľuje problematiku vizualizácie dát je vhodné podotknúť, že zošity sú taktiež vhodné na vytvorenie naratívnej vizualizácie (kombinácia textu a vizualizácií s cieľom odpovedať na určitú otázku). Nástroj si nájde uplatnenie aj pri samostatných prácach, kde bude od študentov vyžadované, aby svoj kód komentovali pomocou textových buniek. Vďaka tomu sa úlohy stanú čitateľnejšími, bude možné lepšie overiť, či študent správne pochopil preberanú látku.

Nevýhodou tohto nástroja je, že môže byť obtiažne jednotlivé zošity spojazdniť. Je potrebné si nainštalovať samotný Jupyter Notebook a stiahnuť

[2] https://jupyter.org

si všetky balíčky, ktoré sa v danom zošite používajú. Z tohto dôvodu som pre študentov vypracovala návod ako Jupyter Notebook používať s pomocou nástroja Anakonda [3]. Viac o návode píšem v nasledujúcej sekcii. Alternatívnym riešením je použiť cloudové nástroje ako sú Colab a Deepnote, do ktorých je možné zošity importovať. O tejto možnosti budú študenti informovaní. Deepnote sa bude pravdepodobne využívať pri domácich úlohách, pretože je v ňom možné komentovať konkrétne bunky, čo uľahčí komunikáciu medzi študentom a pedagógom.

6.1.2 Grafické balíčky

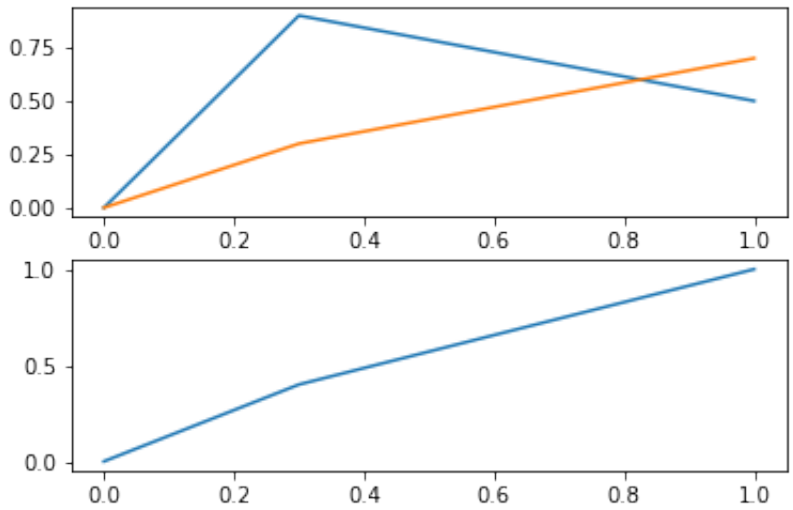
V predmete sa budú používať dva grafické balíčky - matplotlib [4] a plotly [5]. Matplotlib vznikol už v roku 2003 a doteraz je najpoužívanejším nástrojom na tvorbu vizualizácií v Python komunite. Je optimalizovaný a dokáže vytvoriť grafy vedeckej kvality. Z pohľadu architektúry sa matplotlib dá rozdeliť do troch vrstiev. Backend vrstva je najkomplexnejšou vrstvou a má na starosti vykresľovanie grafického výstupu. Jej základnými prvkami sú `FigureCanvas` (oblasť, na ktorú sa vykresľuje), `Renderer` (kreslí na `FigureCanvas`) a `Event` (spracováva používateľské vstupy). Artist vrstva spravuje všetky prvky, ktoré sa majú vykresliť na grafické plátno. Jej najdôležitejšou časťou je abstraktná trieda `Artist`. Takmer všetky viditeľné elementy grafov (legenda, osi, nadpisy, ...) sú podtriedami triedy `Artist`. Poslednou z troch vrstiev je skriptovacia vrstva. Je to vrstva, s ktorou najčastejšie prichádza do kontaktu koncový používateľ. Jej úlohou je zjednodušiť prácu s balíčkom v prípade častých používateľských scenárov ako je napríklad vytvorenie typických grafov. Skriptovacia vrstva ponúka dva typy rozhrania:

- procedurálne stavové rozhranie (pyplot API a pylab API),
- objektovo orientované rozhranie (preferované).

Veľmi často je v článkoch na internete vidieť kód, ktorý tieto dve rozhrania používa súčasne. Takýto postup vedie na chyby, ktoré sa prejavia len v niektorých prípadoch a tým pádom je ťažké ich odhaliť. V nasledujúcej ukážke je ekvivalentný kód v oboch rozhraniach, ktorý produkuje grafy z obrázku 6.1.

```
x = [0,0.3,1]
y1 = [0,0.9,0.5]
y2 = [0,0.3,0.7]
y3 = [0,0.4,1]
```

[3] https://www.anaconda.com
[4] https://matplotlib.org
[5] https://plotly.com/python

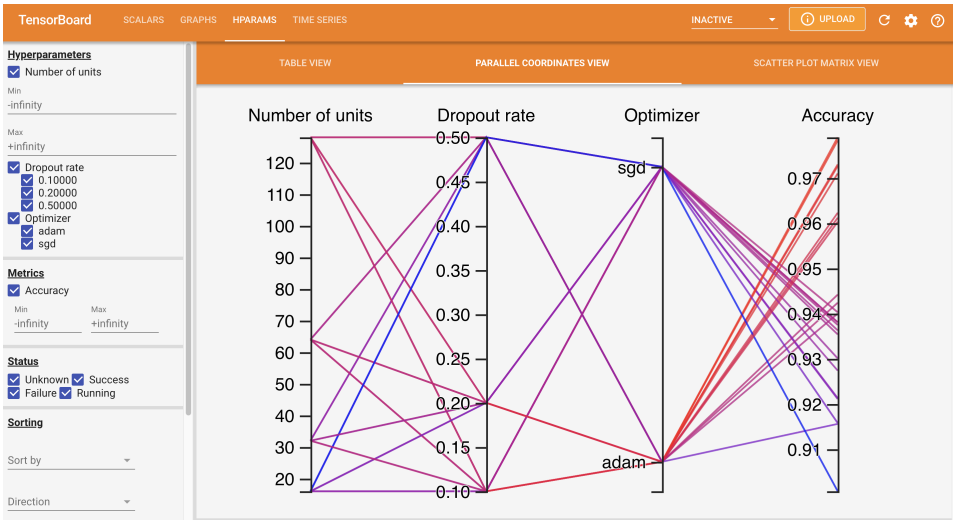


Obr. 6.1: Graf vyrobený pomocou balíčka matplotlib

```
# state-based interface
plt.figure(figsize=(6,4))
plt.subplot(2,1,1)
plt.plot(x, y1)
plt.plot(x, y2)
plt.subplot(2,1,2)
plt.plot(x, y3)
# object-oriented interface
fig = plt.figure(figsize=(6,4))
ax = fig.add_subplot(2,1,1)
ax.plot(x, y1)
ax.plot(x, y2)
ax = fig.add_subplot(2,1,2)
ax.plot(x, y3)
```

Plotly je novší balíček, ktorý vznikol v roku 2013. Jeho výhodou je, že tvorba každého grafu je postavená na jednotnej syntaxi. Výsledné grafy sú taktiež estetické a interaktívne bez toho, aby programátor musel tieto vlastnosti špecifikovať.

V predmete BI-VIZ budeme používať hlavne matplotlib (presnejšie jeho objektovo orientované rozhranie), pretože je stále používanejší ako plotly. Jeho predstaveniu bude venovaný samostatný Jupyter Notebook zošit. V niektorých prípadoch bude využité aj plotly. Jedná sa o prípady, kedy plotly implementuje vizualizácie, ktoré matplotlib neponúka (napr. graf paralelných súradníc).



Obr. 6.2: Dashboard HParams v nástroji TensorBoard

6.1.3 Pandas

Pandas [6] je Python balíček používaný na analýzu tabuľkových dát a ich následné spracovanie. Tento balíček sa používa aj v iných predmetoch zameraných na strojové učenie vyučovaných na FIT ČVUT a je často používaný komunitou datových analytikov pracujúcich v Pythone.

6.1.4 Tensorboard

TensorFlow [7] je end-to-end open source platforma pre strojové učenie. Zameriava sa predovšetkým na neurónové siete. Pomáha riešiť problémy ako sú klasifikácia obrázkov, preklad alebo generovanie textov, zvýšenie rozlíšenia videa a podobne. TensorFlow poskytuje nástroje na trénovanie modelov v jazykoch Python a JavaScript, ale aj na ich nasadenie do webových a mobilných aplikácií, na cloud a podobne.

TensorBoard [8] je vizualizačný nástroj pre TensorFlow, ktorý produkuje vizualizácie súvisiace s architektúrou a evaluáciu modelu či ladením jeho hyperparametrov. Jeho ukážka je na obrázku 6.2. Vzniknuté vizualizácie je možné publikovať. TensorBoard ponúka aj pokročilé nástroje ako sú *Embedding projector* (graficky zobrazuje nízko-dimenzionálne vnorenia), *What-If tool* (zjednodušuje interpretáciu modelu) či *Profiling tool* (pomáha s optimalizáciou).

Nástroj funguje na jednoduchom princípe - počas trénovania modelu sa zapisujú logy do vopred určených súborov a TensorBoard ich následne načíta

[6] https://pandas.pydata.org
[7] https://www.tensorflow.org
[8] https://www.tensorflow.org/tensorboard

a zobrazí v niekoľkých dashboardoch. Existuje viacero dashboardov, pričom každý sa zameriava na vizualizáciu iných logov. Dashboard *Graphs* vizualizuje architektúru modelu, dashboard *HParams* zas vývoj metriky v závislosti na zvolených hyperparametroch a podobne.

6.2 Navrhnuté materiály

Počas tvorby tejto práce som vytvorila štyri prednáškové prezentácie. Cieľom prvej prednášky na tému *typy dát a nástroje na ich spracovanie* je zoznámiť študentov s rôznymi typmi a formátmi dát ako aj metódami na ich získanie, spracovanie a uloženie. Má nasledujúcu osnovu:

- Typy dát
  - Rozdelenie dát podľa štruktúry
  - Rozdelenie príznakov podľa charakteru
- Získavanie dát
  - Otvorené dáta
  - Web crawling, Data Scraping
- Spracovanie dát
  - Formáty na výmenu dát
  - Balíček Pandas
- Ukladanie dát

Druhá prednáška s témou *základné prístupy k vizualizácii dát* vysvetľuje, čo sú to vizuálne premenné a aké majú vlastnosti. Tiež predstavuje rôzne kritériá, podľa ktorých je možné hodnotiť vizualizácie. Táto znalosť študentom pomôže vytvoriť estetické vizualizácie, ktoré vie ľudský mozog intuitívne spracovať. Ďalej sú v prednáške prezentované často používané grafy a grafické balíčky. Osnova prednášky je nasledovná:

- Ako zvoliť správnu vizualizáciu?
  - Značky a vizuálne premenné
  - Typy vizuálnych premenných
- Ako vyhodnotiť kvalitu vizualizácie?
  - Estetika, matematická korektnosť
  - Expresivita a efektivita
- Často používané grafy

- Vizualizácia dát v jazyku Python
  - Matplotlib
  - Seaborn
  - Plotly

Tretia prednáška sa venuje téme *vizualizácie v strojovom učení*. Jej cieľom je zoznámiť študentov s vizualizáciami, ktoré môžu využiť v procese trénovania modelov. Prednáška ukazuje vizualizácie rôznych klasifikačných aj regresných modelov, ale aj vizualizácie používané pri zhodnotení úspešnosti modelov a ladení ich hyperparametrov. Jej osnova je nasledovná:

- Motivácia
- Vizualizácia modelov
  - Rozhodovací strom
  - Algoritmus kNN
  - Lineárna regresia
- Evaluácia modelu
  - Matica zámen
  - ROC krivka
  - Graf predikcií a skutočných hodnôt
  - Tabuľka chýb
  - Grafy distribúcie chyby
- Ladenie hyperparametrov

Posledná prednáška zoznamuje študentov s nástrojom TensorBoard. Vzhľadom k tomu, že TensorBoard sa primárne používa na vizualizáciu neurónových sietí, prednáška začína ľahkým úvodom do tejto problematiky. Ďalej sú vysvetlené nástroje TensorFlow a Keras, ktoré sa budú používať v Jupyter Notebook zošitoch. Druhá polovica prednášky sa zameriava na vysvetlenie všetkých dôležitých dasboardov nástroja TensorBaord. Osnova prednášky je nasledovná:

- Neurónové siete
  - Umelý neurón
  - Viacvrstvová neurónová sieť
- TensorFlow

- Keras
- TensorBoard
  - Scalars
  - Graphs
  - Histograms & Distributions
  - Texts & Images
  - HParams
  - Ostatné dashboardy

Ďalej som vytvorila jedenásť Jupyter Notebook zošitov, ktoré obsahujú praktické ukážky k témam Jupyter Notebook, pandas, matplotlib, vizualizácie v strojovom učení a TensorBoard. Na tému matplotlib som vytvorila dva zošity. Prvý sa venuje jeho základom a ukazuje ako meniť dizajn grafu (šírka, farba a štruktúra čiar) a ako manipulovať s objektami ako sú osi, merítka, mriežka, legenda, popisky osí a podobne. Druhý zošit ponúka ukážku korektných a estetických grafov vytvorených pomocou balíčka matplotlib. Vizualizácie v strojovom učení sú komplexnou témou, a preto som látku rozdelila do štyroch zošitov. Zošity sa venujú vizualizáciám

- natrénovaných modelov,
- evaluácie modelov,
- ladenia hyperparametrov.

Posledný zošit obsahuje praktickú ukážku látky, ktorá bola vysvetlená vo všetkých predchádzajúcich zošitoch. Za pomoci vhodných vizualizácií v ňom trénujem rozhodovací strom na predikciu cien domov. Na tému TensorBoard som vytvorila tri zošity. Prvý zošit preberá základy fungovania tohto nástroja a vysvetľuje hlavné dashboardy - *Scalars* a *Graphs*. Druhý zošit ukazuje ako do nástroja TensorBoard logovať texty a obrázky na praktickom príklade logovania matice zámen po každej epoche trénovania neurónovej siete. Posledný zošit vysvetľuje, ako sa dá TensorBoard využiť na vizualizácie počas procesu ladenia hyperparametrov.

Keďže spojazdnenie zošitov môže byť pre niektorých študentov komplikované, vytvorila som návod ako používať Jupyter Notebook s pomocou nástroja Anakonda. Pri inštalovaní nástroja Anaconda sa totiž nainštaluje aj Python, Jupyter Notebook a ďalšie bežne používané balíčky v strojovom učení. Po nainštalovaní spomínaného nástroja by si ale študenti aj tak museli manuálne stiahnuť všetky balíčky, ktoré sa v zošitoch používajú. Na tento účel som vytvorila konfiguračný súbor `environment.yml`, vďaka ktorému je možné v termináli pomocou jedného príkazu vytvoriť prostredie so všetkými potrebnými

závislosťami. Je nutné poznamenať, že fungovanie súboru bolo zatiaľ overené len na operačnom systéme macOS Monterey a na iných operačných systémoch sa jeho správanie môže líšiť. Je možné, že pre niektoré operačné systémy bude potrebné vytvoriť iné konfiguračné súbory.

V tejto práci som sa venovala aj tvorbe dvoch samostatných prác. Prvá z nich je zameraná na exploračnú analýzu dát z útulku v Austine a druhá na sieťovú analýzu cestovných poriadkov pražskej integrovanej dopravy.

Všetky spomínané materiály sú nahraté na priloženej microSD karte a preto ich v texte práce nerozoberám detailnejšie. Vedúca práce spolu s oponentom práce majú k materiálom prístup aj prostredníctvom repozitára na fakultnom GitLabe.

Záver

Cieľom tejto diplomovej práce bolo vytvoriť niekoľko vzorových študijných materiálov a samostatných prác pre predmet BI-VIZ s ohľadom na predpokladané znalosti študentov. Okrem vypracovanej analýzy tematických celkov, ktoré sa v predmete budú preberať sú výsledkom práce štyri prednáškové prezentácie, jedenásť Jupyter Notebook zošitov s praktickými ukážkami, dve zadania samostatných prác a jeden návod na počiatočnú konfiguráciu vývojového prostredia.

V úvode práce som sa venovala rešerši výuky vizualizácie dát na iných univerzitách. Následne som spolu s vedúcou práce určila tematické celky, ktoré sa budú v predmete BI-VIZ preberať. Pre každý tematický celok som vykonala rešerš z pohľadu vizualizácie dát a vybrala metódy, ktoré by bolo vhodné v rámci predmetu vyučovať. Metódy som vybrala s ohľadom na predpokladané znalosti študentov. Odhad znalostí som založila na doporučenom priechode štúdiom pre bakalársku špecializáciu Umelá inteligencia. Následne som pre niektoré tematické celky vytvorila študijné materiály a samostatné práce na overenie získaných znalostí.

Literatúra

[1] dopravní podnik hl. m. prahy: Bezbariérové cestování - metro. Dostupné z: https://www.dpp.cz/cestovani/bezbarierove-cestovani/metro

[2] Bertin, J.: *Semiology of Graphics: Diagrams, Networks, Maps*. Esri Press, 2010, iSBN: 978-1-589-48261-6.

[3] Ribecca, S.: The Data Visualisation Catalogue. [cit. 2022-04-08]. Dostupné z: https://datavizcatalogue.com/

[4] Davies, J.; Ramjohn, I.: File:summer seminar motivations wordcloud.svg. Dostupné z: https://commons.wikimedia.org/wiki/File:Summer_Seminar_motivations_wordcloud.svg

[5] Kessler, J. S.: Scattertext: a browser-based tool for visualizing how corpora differ. *arXiv preprint arXiv:1703.00565*, 2017.

[6] COSTE, A.: Image channel decomposition. Sep 2012. Dostupné z: http://www.sci.utah.edu/~acoste/uou/Image/project1/Arthur_COSTE_Project_1_report.html

[7] Frank, H.: A RGB color cube explained with three diagrams. [prístup 2022-04-12]. Dostupné z: https://commons.wikimedia.org/wiki/File:RGB_color_cube.svg

[8] Erbas, B.; Hyndman, R.: Data visualisation for time series in environmental epidemiology. *Journal of Epidemiology and Biostatistics*, ročník 6, č. 6, 2001: s. 433–443.

[9] Aksakalli, C.: Network centrality measures and their visualization. 2017.

[10] Nocaj, A.: *Untangling Networks: Focus on Less to See More*. Dizertační práce, University of Konstanz, 2015.

[11] Williams, D.: Graph visualization: Fixing data hairballs. Jan 2022. Dostupné z: https://cambridge-intelligence.com/how-to-fix-hairballs/

[12] Krzywinski, M.; Birol, I.; Jones, S. J.; aj.: Hive plots—rational approach to visualizing networks. *Briefings in bioinformatics*, ročník 13, č. 5, 2012: s. 627–644.

[13] Kairam, S.; MacLean, D.; Savva, M.; aj.: GraphPrism: compact visualization of network structure. In *Proceedings of the international working conference on advanced visual interfaces*, 2012, s. 498–505.

[14] Shukla, L.: Better models faster with weights & biases. Jan 2020. Dostupné z: https://wandb.ai/site/articles/better-models-faster-with-weights-biases

[15] de Voltolina, L.: Medieval university. [prístup 2022-04-02]. Dostupné z: https://commons.wikimedia.org/wiki/File:Laurentius_de_Voltolina_001.jpg

[16] Park, E. L.; Choi, B. K.: Transformation of classroom spaces: Traditional versus active learning classroom in colleges. *Higher Education*, ročník 68, č. 5, 2014: s. 749–771.

[17] sli.do s.r.o: Slido [software]. 2012, [prístup 2022-04-02]. Dostupné z: https://www.sli.do/

[18] Apple unleashes M1. Apr 2022. Dostupné z: https://www.apple.com/newsroom/2020/11/apple-unleashes-m1/

[19] Temple, S.: Word clouds are lame. Oct 2019. Dostupné z: https://towardsdatascience.com/word-clouds-are-lame-263d9cbc49b7

[20] Pedregosa, F.; Varoquaux, G.; Gramfort, A.; aj.: Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, ročník 12, 2011: s. 2825–2830.

[21] Terence Parr, P. G., Tudor Lapusan: dtreeviz (Software). Dostupné z: https://github.com/parrt/dtreeviz

[22] Fry, B.: *Visualizing data: Exploring and explaining data with the processing environment*. Ö'Reilly Media, Inc.", 2008.

[23] Stevens, S. S.: On the theory of scales of measurement. *Science*, ročník 103, č. 2684, 1946: s. 677–680.

[24] Mackinlay, J. D.: Automatic design of graphical presentations. Technická zpráva, Stanford Univ., CA (USA), 1987.

[25] Shamoo, A. E.; Resnik, D. B.: *Responsible conduct of research*. Oxford University Press, 2009.

[26] Tukey, J. W.; aj.: *Exploratory data analysis*, ročník 2. Reading, MA, 1977.

[27] Dempír, J.; Dohnal, L.: Některé robustní postupy určení střední hodnoty a rozptỳlení souboru vỳsledků a jejich použití. *Klin. Biochem. Metab*, ročník 13, č. 34, 2005: str. 3.

[28] Hyndman, R. J.; Fan, Y.: Sample quantiles in statistical packages. *The American Statistician*, ročník 50, č. 4, 1996: s. 361–365.

[29] Yang, J.; Kim, M.: Independence test of a continuous random variable and a discrete random variable. *Communications for Statistical Applications and Methods*, ročník 27, č. 3, 2020: s. 285–299.

[30] Eisenstein, J.: *Introduction to natural language processing*. MIT press, 2019.

[31] Zhang, Y.-J.: Image engineering. In *Handbook of Image Engineering*, Springer, 2021, s. 55–83.

[32] Fuller, W. A.: *Introduction to statistical time series*. John Wiley & Sons, 2009.

[33] Hyndman, R. J.; Athanasopoulos, G.: *Forecasting: principles and practice*. OTexts, 2018.

[34] Makridakis, S.; Wheelwright, S. C.; Hyndman, R. J.: *Forecasting methods and applications*. John wiley & sons, 2008.

[35] Dagum, E. B.; Bianconcini, S.: *Seasonal adjustment methods and real time trend-cycle estimation*. Springer, 2016.

[36] Cleveland, R. B.; Cleveland, W. S.; McRae, J. E.; aj.: STL: A seasonal-trend decomposition. *J. Off. Stat*, ročník 6, č. 1, 1990: s. 3–73.

[37] Barabási, A.-L.: Network science. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, ročník 371, č. 1987, 2013: str. 20120375.

[38] HLINĚNÝ, P.: Základy teorie grafů pro (nejen) informatiky. 2006.

[39] Powell, J.: *A librarian's guide to graphs, data and the semantic web*. Elsevier, 2015.

[40] Edge, D.; Larson, J.; Mobius, M.; aj.: Trimming the hairball: Edge cutting strategies for making dense graphs usable. In *2018 IEEE International Conference on Big Data (Big Data)*, IEEE, 2018, s. 3951–3958.

[41] Alpaydin, E.: *Introduction to machine learning*. MIT press, 2020.

[42] Abadi, M.; Agarwal, A.; Barham, P.; aj.: TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems. 2015, software available from tensorflow.org. Dostupné z: `https://www.tensorflow.org/`

[43] inc., W. . B.: Weights & Biases (Software). Dostupné z: `https://wandb.ai/site`

[44] Gheorghiu, T.: ANN Visualizer (Software). Dostupné z: `https://github.com/RedaOps/ann-visualizer`

[45] Hossin, M.; Sulaiman, M. N.: A review on evaluation metrics for data classification evaluations. *International journal of data mining & knowledge management process*, ročník 5, č. 2, 2015: str. 1.

[46] Botchkarev, A.: Performance metrics (error measures) in machine learning regression, forecasting and prognostics: Properties and typology. *arXiv preprint arXiv:1809.03006*, 2018.

[47] Droščák, M.: Aktívne učenie, aktívne vyučovanie - ich význam, podstata a spôsoby realizácie v školskej praxi. *Paedagogica*, ročník 23, 2011: s. 65–79, [cit. 2022-04-02]. Dostupné z: `https://fphil.uniba.sk/fileadmin/fif/katedry_pracoviska/kped/projekty/Archiv_Paedagogica/23_-_4.pdf`

[48] McCarthy, J. P.; Anderson, L.: Active learning techniques versus traditional teaching styles: Two experiments from history and political science. *Innovative higher education*, ročník 24, č. 4, 2000: s. 279–294.

[49] Melo Prado, H.; Hannois Falbo, G.; Rodrigues Falbo, A.; aj.: Active learning on the ward: outcomes from a comparative trial with traditional methods. *Medical education*, ročník 45, č. 3, 2011: s. 273–279.

[50] Michael, J.: Where's the evidence that active learning works? *Advances in physiology education*, 2006.

[51] Schwerdt, G.; Wuppermann, A. C.: Is traditional teaching really all that bad? A within-student between-subject approach. *Economics of Education Review*, ročník 30, č. 2, 2011: s. 365–379.

[52] TOMENGOVÁ, A.: Aktívne učenie sa žiakov–stratégie a metódy. *Bratislava: MPC*, 2012.

[53] Bradbury, N. A.: Attention span during lectures: 8 seconds, 10 minutes, or more? 2016.

[54] Felder, R. M.; Brent, R.: Active learning: An introduction. *ASQ higher education brief*, ročník 2, č. 4, 2009: s. 1–5.

Dodatok **A**

Zoznam použitých skratiek

FIT VUT Fakulta informačních technologií Vysokého učení technického v Brně

FEL ČVUT Fakulta elektrotechnická Českého vysokého učení technického v Praze

MUNI Masarykova univerzita

NLP Natural Language Processing

MSE Mean Squared Error

RMSE Root Mean Squared Error

RMSLE Root Mean Squared Logaritmic Error

MAE Mean Absolute Error

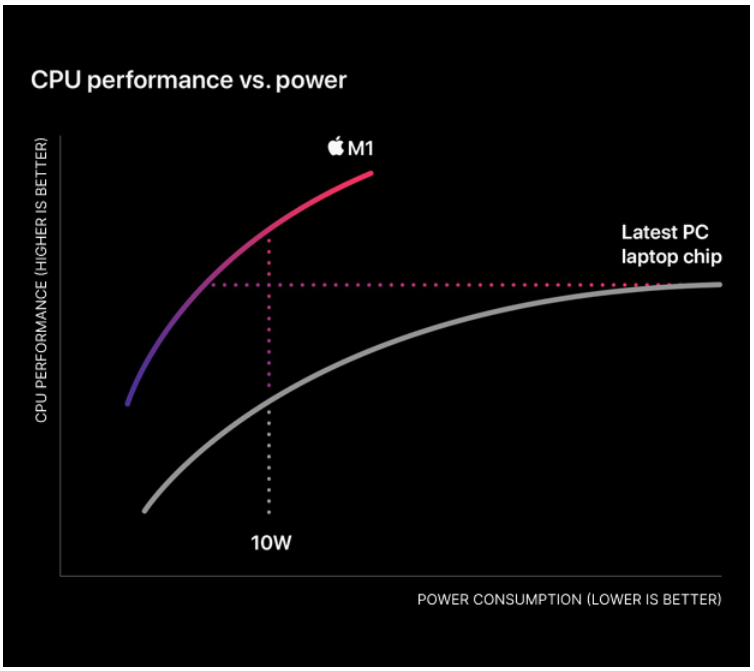
Q&A Questions and Answers (sekcia aplikácie Slido)

GUI Graphical User Interface

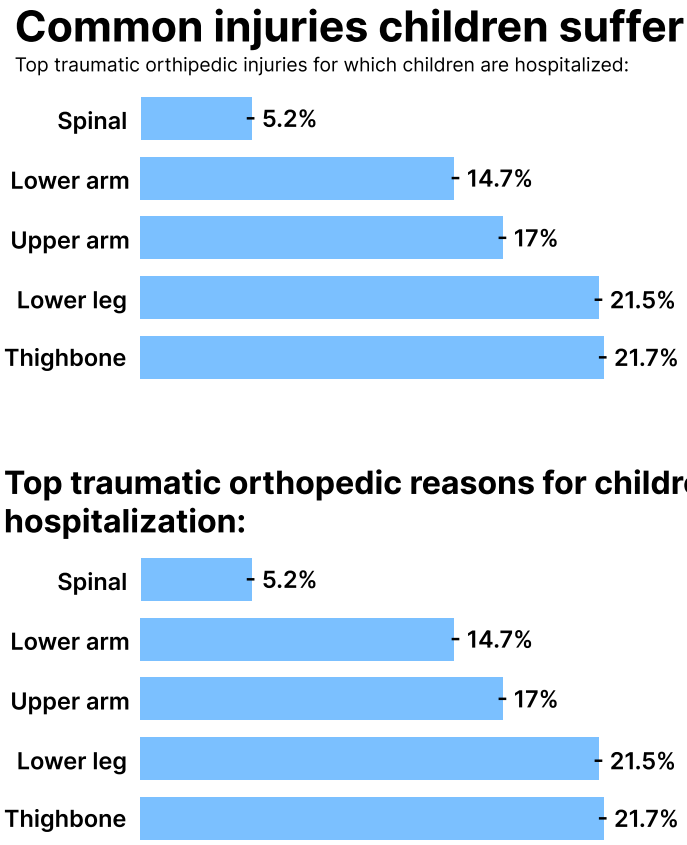
API Application Programming Interface

Dodatok B

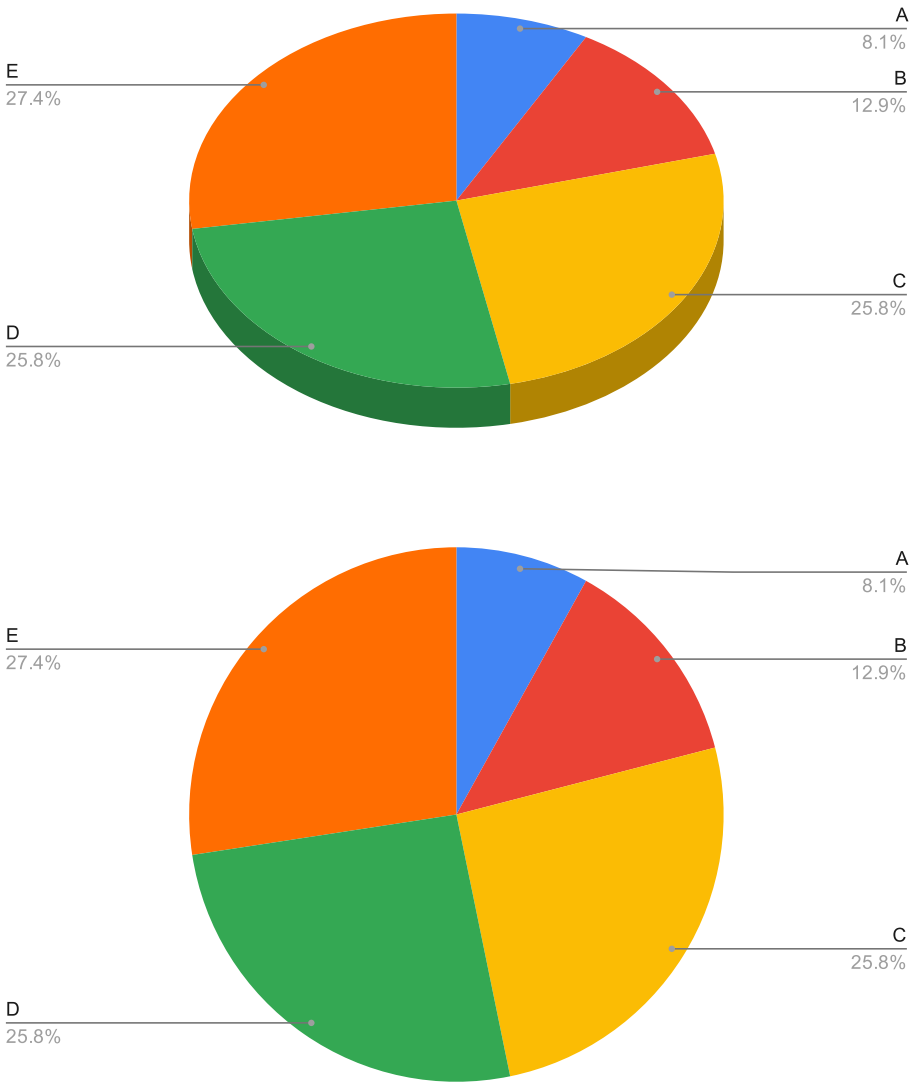
Mätúce grafy



Obr. B.1: Graf použitý pri predstavení M1 chipu. Osi nemajú merítka a nie je zrejmé v akých jednotkách sa dáta merali [18].



Obr. B.2: Ukážka zavádzajúceho nadpisu (hore) a lepšieho nadpisu (dole)



Obr. B.3: 3D koláčový graf v porovnaní s 2D koláčovým grafom. 2D graf zobrazuje proprcie lepšie.

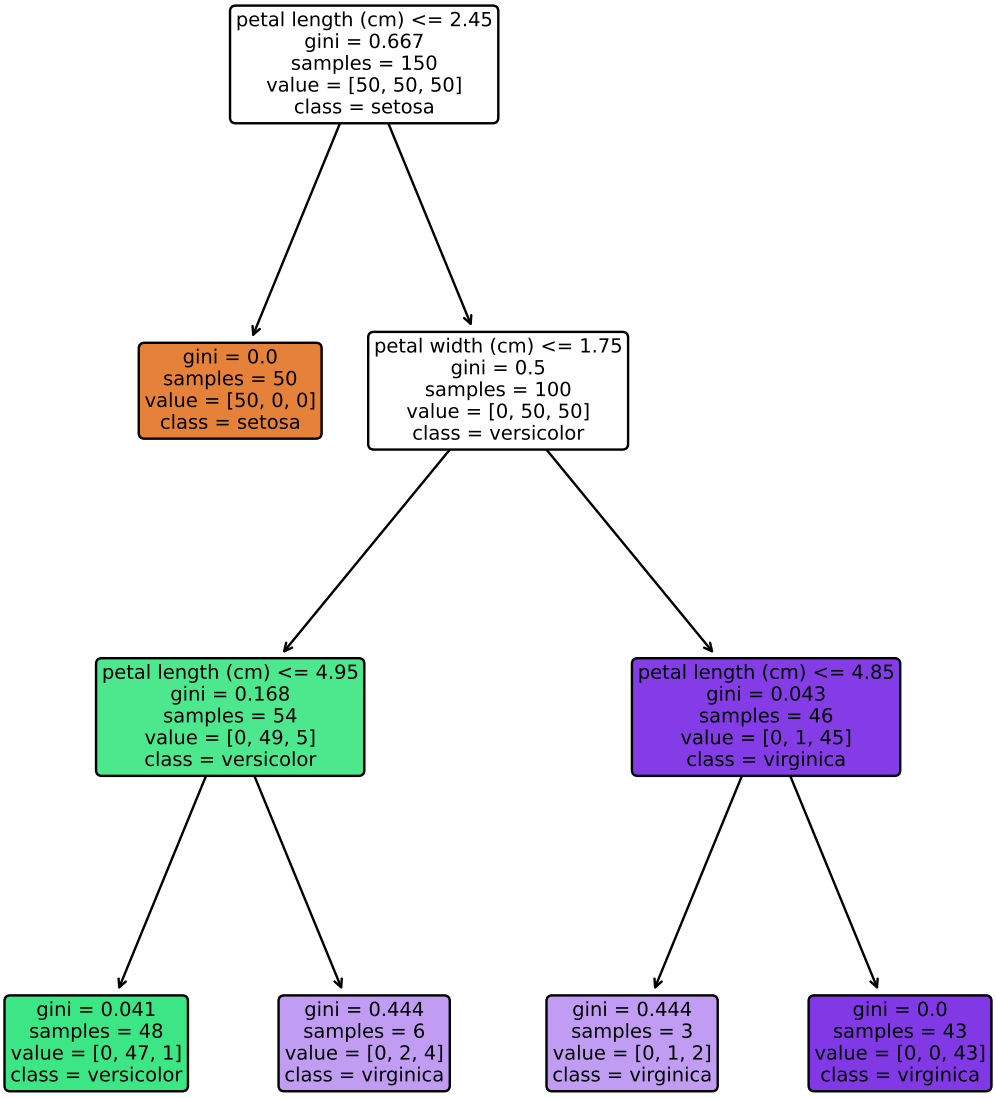
Dodatok C

Vizualizácie textu

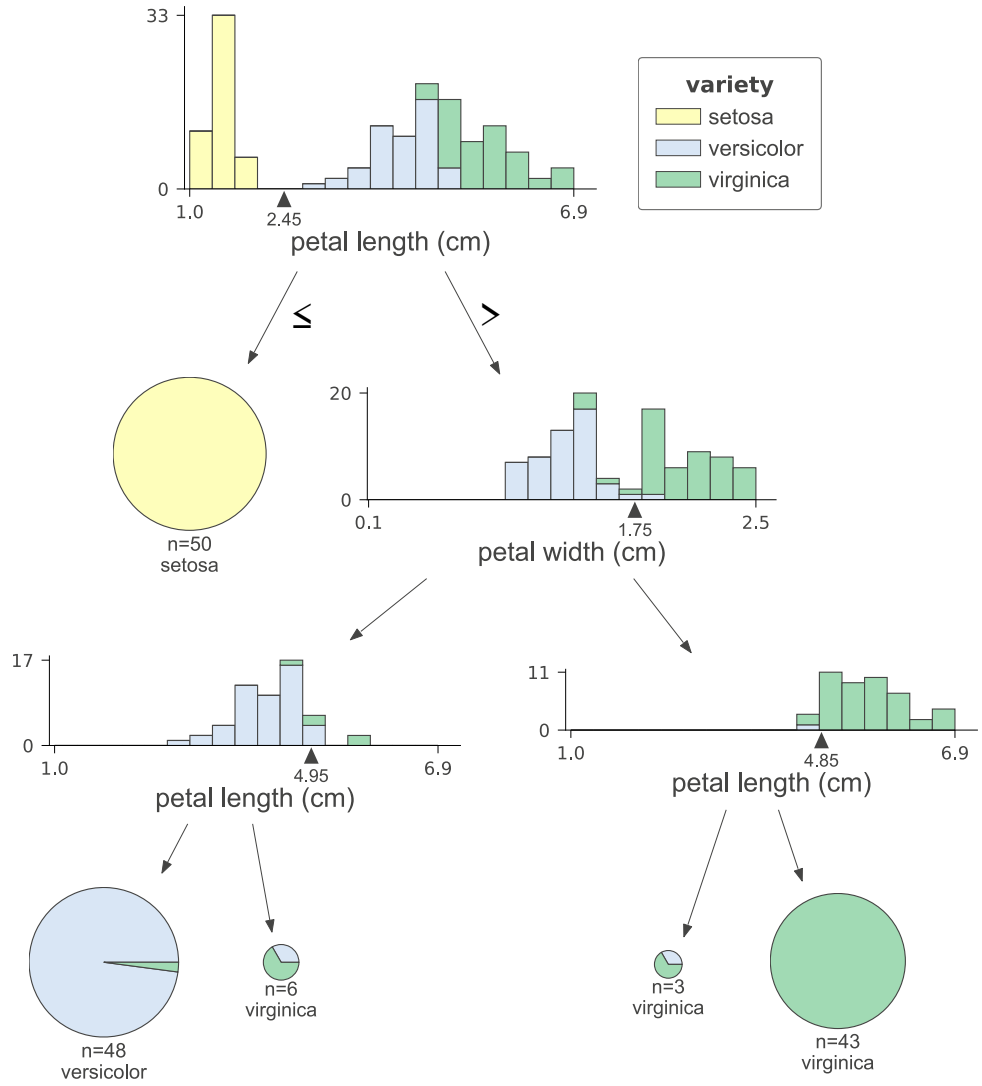
Obr. C.1: Porovnanie word cloudu (hore) a stĺpcového grafu (dole). Zo stĺpcového grafu sa odpovedá ľahšie na otázky ohľadom frekvencie slov [19].

Dodatok D

Vizualizácie používané v strojovom učení



Obr. D.1: Klasifikačný rozhodovací strom vykreslený pomocou scikit-learn [20]



Obr. D.2: Klasifikačný rozhodovací strom vykreslený pomocou dtreeviz [21]

Obr. D.3: Vývoj validačnej a trénovacej presnosti pre rôzne maximálne hĺbky rozhodovacieho stromu

Dodatok **E**

Obsah priloženej microSD karty

- `readme.txt` ..... stručný popis obsahu microSD karty
- `src`
  - `materials` ..... navrhnuté študijné materiály
    - `lectures` ..... prednášky vo formáte PDF
    - `tutorials` ..... zošity s praktickými ukážkami
    - `homeworks` ..... zadania samostatných prác
    - `guides` ..... návody
  - `thesis` ..... zdrojová forma práce vo formáte LaTeX
- `text` ..... text práce
  - `thesis.pdf` ..... text práce vo formáte PDF