Transformacije podataka i inženjerstvo značajki

Dubinska analiza podataka

3. predavanje

Pripremio: izv. prof. dr. sc. Alan Jović

Ak. god. 2023./2024.

Sadržaj

- Uvod u transformacije podataka
- Ručni pristup inženjerstvu značajki
  - Transformacija varijabli
  - Izlučivanje značajki
  - Uklanjanje nebitnih i redundantnih značajki
- Poluautomatizirani pristup inženjerstvu značajki
  - Izgradnja značajki
  - Redukcija dimenzionalnosti

Uvod u transformacije podataka

Promjena oblika skupa podataka

- Zašto nam je važno preoblikovati (transformirati) skup podataka?
- **Podaci su predočeni u obliku koji se ne može jednostavno iskoristiti za modeliranje**
  - Nisu normirani/standardizirani što otežava usporedbu
  - Zadani su u obliku vremenskih nizova / signala / teksta koje algoritam za modeliranje ne razumije

VARIJABLE

- **Podataka nema dovoljno / ima previše**
  - Algoritmi neće moći izgraditi dobro generalizirajući model / trajat će predugo

PRIMJERCI

Tema 5. predavanja

Promjena oblika skupa podataka

- U kontekstu promjene oblika skupa podataka, **izvorne kategorije** koje su **mjerene/prikupljene (senzorima, upitnicima, intervjuima)** nazivamo **varijablama**
  - Npr. u tabličnom prikazu podataka: spol osobe (muško/žensko), visina osobe u cm (numerički tip)
  - Npr. u vremenskom nizu: konačna cijena dionice u danu, napon na elektrodi elektrokardiograma u nekom trenutku
  - Npr. u slici: razina boje (ili sivila) piksela

Promjena oblika skupa podataka

- Izvorne varijable moguće je **transformirati (preoblikovati) jednom ili više transformacija**
  - Npr. transformacija u tabličnom prikazu podataka: spol osobe (0/1), visina osobe (visoka/srednja/niska)
  - Npr. transformacija u vremenskom nizu: konačna cijena dionice u danu podijeljena ukupnim volumenom trgovanja, Fourierova transformacija napona na elektrodi elektrokardiograma
  - Npr. transformacija u slici: razina boje (ili sivila) piksela s dodanim šumom

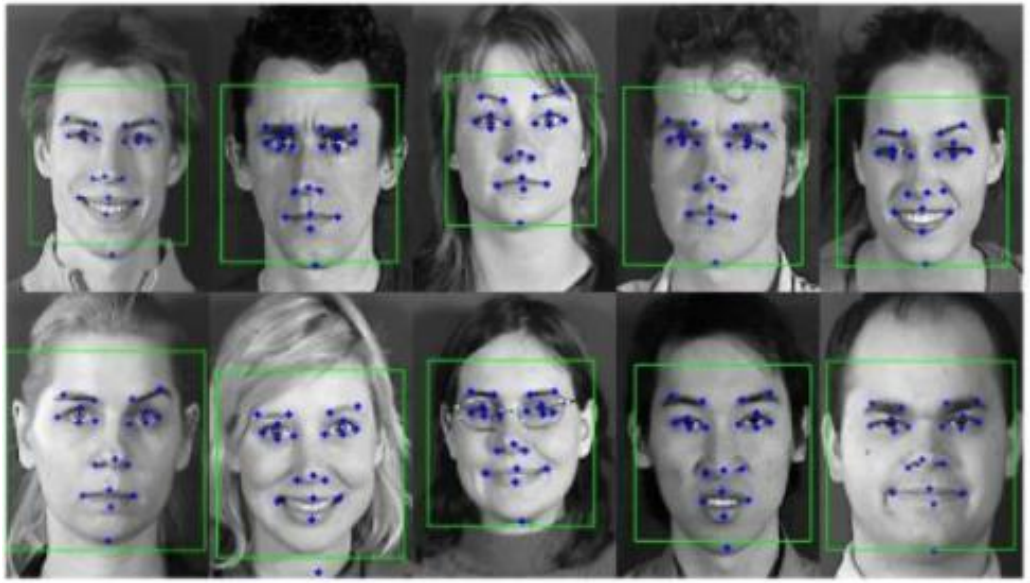
Značajke

- Transformirane varijable koje se koriste **na ulazu** metoda strojnog učenja nazivaju se **značajkama** (engl. **feature**), a cijeli primjerak naziva se **vektor značajki** (engl. ***feature vector***)
- Neki put, izvorne kategorije (varijable) su ujedno i značajke
- Češće, značajke su **rezultat posljednje matematičke operacije** prije nego što podatak uđe u metodu strojnog učenja
  - Npr. prosječna vrijednost unazad 5 dana cijene dionice pri zatvaranju trgovanja podijeljena ukupnim volumenom trgovanja; spektralna gustoća snage u niskofrekventnom pojasu 0.04 – 0.15 Hz vremenskog niza srčanih otkucaja

Značajke – svojstva

- Značajke su često vrlo specifične i **domenski definirane**
  - **Predlažu ih stručnjaci** za pojedinu domenu (područje)
  - **Računaju ih specijalizirani programi** na temelju izvornih varijabli
  - One su **linearne i nelinearne transformacije** jedne ili više izvornih varijabli
  - Često imaju **razumljivu interpretaciju**

https://www.investopedia.com/articles/trading/07/adx-trend-indicator.asp



https://www.slideshare.net/ahmadabdelhafeez5/facial-expression-recognition-based-on-local-binary-patterns-final

Značajke – razlozi korištenja

- Zašto nam trebaju značajke?
  - **Sažetost**: Sirovih podataka (varijabli) ima previše za učinkovito korištenje algoritama strojnog / dubokog učenja
  - **Informativnost**: Sirovi podaci (varijable) nisu toliko informativni, jer se zanimljiva informacija krije u matematičkim odnosima / transformacijama između njih
  - **Interpretabilnost**: Ljudi bolje razumiju značajke od sirovih podataka (varijabli) kada se radi o interpretaciji modela strojnog učenja

Pristupi inženjerstvu značajki

- Područje inženjerstva značajki bavi se svim transformacijama nad varijablama kojima se dobiva konačni vektor značajki
- To je zadatak **izgradnje skupa podataka** po CRISP-DM-u (dolazi nakon čišćenja podataka)
- Pristupi
  - **Ručni pristup inženjerstvu značajki** Tema današnjeg predavanja
  - **Poluautomatizirani pristup inženjerstvu značajki** Tema današnjeg i idućeg predavanja
  - **Automatizirani pristup izlučivanju značajki** Tema 10. predavanja (duboko učenje u DAP-u)

Ručni pristup inženjerstvu značajki

Koraci ručnog pristupa inženjerstvu značajki

1. **Transformacije varijabli**
2. **Izlučivanje značajki**
3. **Uklanjanje nebitnih i redundantnih značajki**

Napomena: prije prvog koraka transformacije varijabli **nužno je provesti tehnike čišćenja podataka**, vidjeti prošlo predavanje

Transformacije varijabli

- Velik broj transformacija varijabli, ovisno o tipu podatka s kojima radimo

- Tablične varijable
  1. **Diskretizacija numeričkih varijabli**
  2. **Pretvorbe kategoričkih varijabli**
  3. **Normalizacija vrijednosti varijabli**

- Vremenski nizovi podataka 9. predavanje
  1. **Razlike n-tog reda**
  2. **Diskretizacija vremenskog niza**
  3. **Dodavanje šuma varijablama**
  4. **Fourierova transformacija**
  5. **Valićna transformacija**

- Slike Ne bavimo se
  1. **Tehnike kontrastiranja / korekcije svjetline**
  2. **Pretvorba u sivu skalu**

- Tekst Ne bavimo se
  1. **Bag of Words (BOW)**
  2. **Term Frequency i Inverse Document Frequency (TF/IDF)**
  3. **Word to Vectors (Word2Vec)**
  4. **Kontekstualizirane vektorske reprezentacije**

Diskretizacija numeričkih varijabli

- **Numeričke vrijednosti varijable --> kategoričke vrijednosti varijable (engl. *binning*)**
- Pretpostavka: broj kategorija << broj numeričkih vrijednosti
- Ponekad se provodi u analizi podataka
  - Neki algoritmi funkcioniraju samo koristeći diskretne, kategoričke vrijednosti (induktivna pristranost)
  - Performance algoritama degradiraju ako varijable nemaju razdiobu gustoće vjerojatnosti blisku uniformnoj
- Primjeri algoritama koji zahtijevaju diskretizaciju:
  - Neki algoritmi stabala odluke (engl. *decision trees*)
  - Neki algoritmi temeljeni na induktivnim pravilima (engl. *induction rules, rule-based system*)
  - Sustavi asocijativnih pravila (engl. *association rules*)
- **Diskretizacijom se uvijek gubi određena informacija!**

Diskretizacija numeričkih varijabli

- Vjerojatno najbolju diskretizaciju neke numeričke varijable mogu predložiti **stručnjaci** iz nekog područja
- U izostanku tog prijedloga, neki češće korišteni postupci diskretizacije su:
  - **Podjela u *K* intervala jednake širine** (engl. *equal width binning*)
  - **Podjela u intervale s jednakim brojem primjeraka** (engl. *equal frequency binning*)
  - **Diskretizacija algoritmom *k*-srednjih vrijednosti** (engl. *k-means discretization*)
  - **Diskretizacija minimizacijom entropije** (engl. *entropy minimization discretization*)
  - ...

Nenadzirani pristup

Nadzirani pristup

- `KBinsDiscretizer` u scikit-learnu

https://machinelearningmastery.com/discretization-transforms-for-machine-learning/

Diskretizacija minimizacijom entropije

- **Fayyad i Irani 1993.** – točke koje dijele skup sortiranih vrijednosti numeričke varijable izabrane su tako da **minimiziraju zajedničku entropiju varijable i ciljne klase**:

$$H(X,Y) = \sum_{x \in X, y \in Y} p(x,y)\, log_2 \frac{p(x,y)}{p(x)p(y)}$$

- Postupak se provodi **iterativno**, tako da se u svakom koraku nad sortiranim vrijednostima varijable traži sljedeća **podjela** (engl. *split*) koja ima najmanju entropiju u odnosu na sve ostale podjele
- S postupkom se staje kada je **informacijski dobitak** (engl. ***information gain, Gain***) od podjele manji (ili jednak) **najmanjoj duljini opisa** (engl. ***minimum description length*, MDL**)
  - *MDL* ovisi o broju primjeraka *N*, broju klasa ciljne varijable *k*, entropiji primjeraka *E*, entropiji primjeraka u svakom podintervalu podjele E_1 i E_2 i broju klasa predstavljenih u svakom podintervalu k_1 i k_2:
  - Ako vrijedi $Gain > \frac{\log_2(N-1)}{N} + \frac{\log_2(3^k - 2) - kE + k_1E_1 + k_2E_2}{N}$ onda dolazi do podjele
  - **Gain = $info$(stanje prije podjele) – $info$(stanje nakon podjele)**

Izvor: Witten IH, Frank E, Hall MA, Pal CJ. Data Mining: Practical Machine Learning Tools and Techniques. 4th ed. Morgan Kaufmann, 2016.

Diskretizacija minimizacijom entropije

$$info[x,y] = entropy\left(\frac{x}{x+y}, \frac{y}{x+y}\right),$$

Za procjenu količine informacije koristi se entropija

$$entropy(p_1, p_2) = -p_1 log_2(p_1) - p_2 log_2(p_2)$$

- **Primjer:**

64	65	68	69	70	71	72	75	80	81	83	85
Yes	No	Yes	Yes	Yes	No	No Yes	Yes Yes	No	Yes	Yes	No

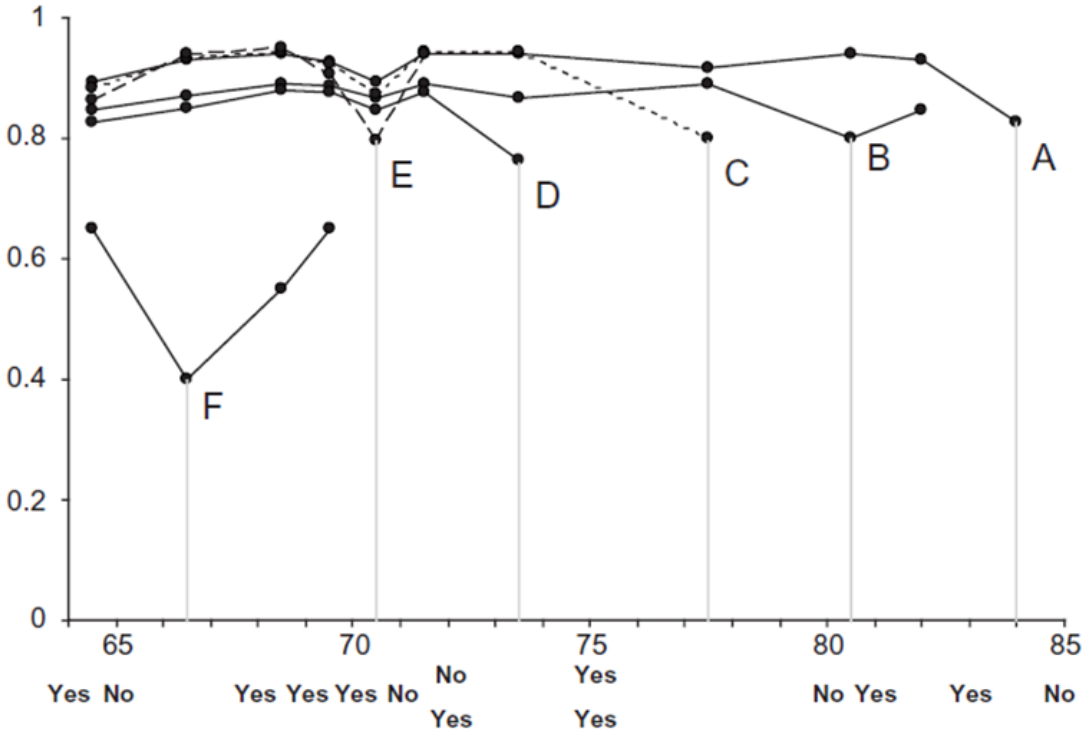
- Stanje prije podjele: $info([9,5])$ = 0,940 bitova

- Stanje nakon podjele na vrijednosti 71,5: $info([4,2],[5,3]) = {}^{6}/_{14}\, info[4,2] + {}^{8}/_{14}\, info[5,3] = 0{,}939$ bitova

Diskusija: Je li ovakva podjela dobra?

Diskretizacija minimizacijom entropije

- Iterativni postupak:



1. iteracija: lom A ima najmanju entropiju (0,827)
2. Iteracija: lom B ima najmanju entropiju (0,8)

...

Može se pokazati da je potrebno razmotriti samo **korisne** lomove – lom je koristan ako je vrijednost varijable Y **različita** s obje strane loma

Konačna diskretizacija:

64	65	68	69	70	71	72	75	80	81	83	85
Yes	No	Yes	Yes	Yes	No	No Yes	Yes Yes	No	Yes	Yes	No
		F			E	D	C	B		A	
		66.5			70.5	73.5	77.5	80.5		84	

Pretvorbe kategoričkih varijabli

- Mnogi algoritmi strojnog učenja ne mogu raditi direktno s kategoričkim vrijednostima, nego zahtijevaju da sve ulazne i ciljne varijable bude numeričke
  - Ograničenje koje je uvela **učinkovita implementacija** algoritama strojnog učenja
- **Dvije vrste pretvorbe:**
  - **Izravna pretvorba** kategoričke varijable u numeričku (engl. ***label encoding***, *integer encoding*)
    - kategorija1 -> 1 ; kategorija2 -> 2 .... kategorija*n* -> *n* **samo u slučaju kada poredak kategorija ima smisla**
  - Pretvorba gdje **svaka kategorija** neke kategoričke varijable **postaje nova binarna varijabla** (engl. ***one-hot encoding***)
    - Od *n* kategorija dobivamo *n* binarnih značajki, koje imaju vrijednost 1 za one primjere za koje bi dotična kategorija vrijedila, a 0 inače

https://machinelearningmastery.com/why-one-hot-encode-data-in-machine-learning/

Normalizacija vrijednosti varijabli

- Potrebno kada su različite značajke u skupu podataka **mjerene na različitim skalama**
  - Značajke mjerene na nižim skalama (npr. između 1 i 10) bile bi manje relevantne modelu od onih na višim skalama (npr. između 1000 i 10000)
  - Modeli strojnog učenja u pravilu lošije rade s nenormiranim varijablama
  - Normalizacija omogućuje uključivanje **stršećih vrijednosti** koje nisu pogreške u modeliranje bez značajnog remećenja rezultata modeliranja
- **Najčešća normalizacija je na raspon vrijednosti između 0 i 1**

Normalizacija vrijednosti varijabli

- Postupci normalizacije vrijednosti varijabli
  - **Decimalno skaliranje** – dijeljenje vrijednosti s **maksimalnom** vrijednosti decimalnog mjesta
    - Npr. sa 100, ako su sve vrijednosti do 100, a veće od 10
    - Varijanta: logaritamsko skaliranje – korisno kada su vrijednosti varijable značajno razvučene (visoka varijanca, npr. kod eksponencijalne razdiobe)
  - Normalizacija **Min-Max**: x’ = (x – min) / (max – min) (ako želimo vrijednosti između 0 i 1)
  - Normalizacija **z-skorom** (statistička normalizacija putem srednje vrijednosti i varijance), poznato i kao **standardizacija:** x’ = (x – mean) / stdev
  - Ponekad se u slučaju malog broja stršećih vrijednosti koje želimo izostaviti može koristiti i **podrezivanje** (engl. *clipping*)
    - if x > max, then x' = max; if x < min, then x' = min

Izlučivanje značajki

- Engl. ***feature extraction, feature elicitation, feature calculation***
- **Matematičko definiranje, implementacija u kodu i računanje značajki**
- Značajke često ovisne o domeni primjene, predlaže ih stručnjak (ekspert) u području primjene
- Potencijalno **beskonačni prostor** značajki
- **Ne računaju se iz tabličnih varijabli**, nego se računaju iz ostalih tipova varijabli (vremenski nizovi, slika, tekst...)

Izlučivanje značajki

- Značajke se obično računaju **nakon prethodne pripreme sirovih podataka**, tj. nakon transformacija izvornih varijabli
- U analizi vremenskih nizova razlikujemo:
  - Značajke vremenske domene (često razne statističke značajke)
  - Značajke frekvencijske domene (značajke dobivene iz spektra signala)
  - Nelinearne značajke (značajke faznog prostora, entropije, ...)
- Različite značajke slike (npr. histogrami boja, karakteristične točke lica)

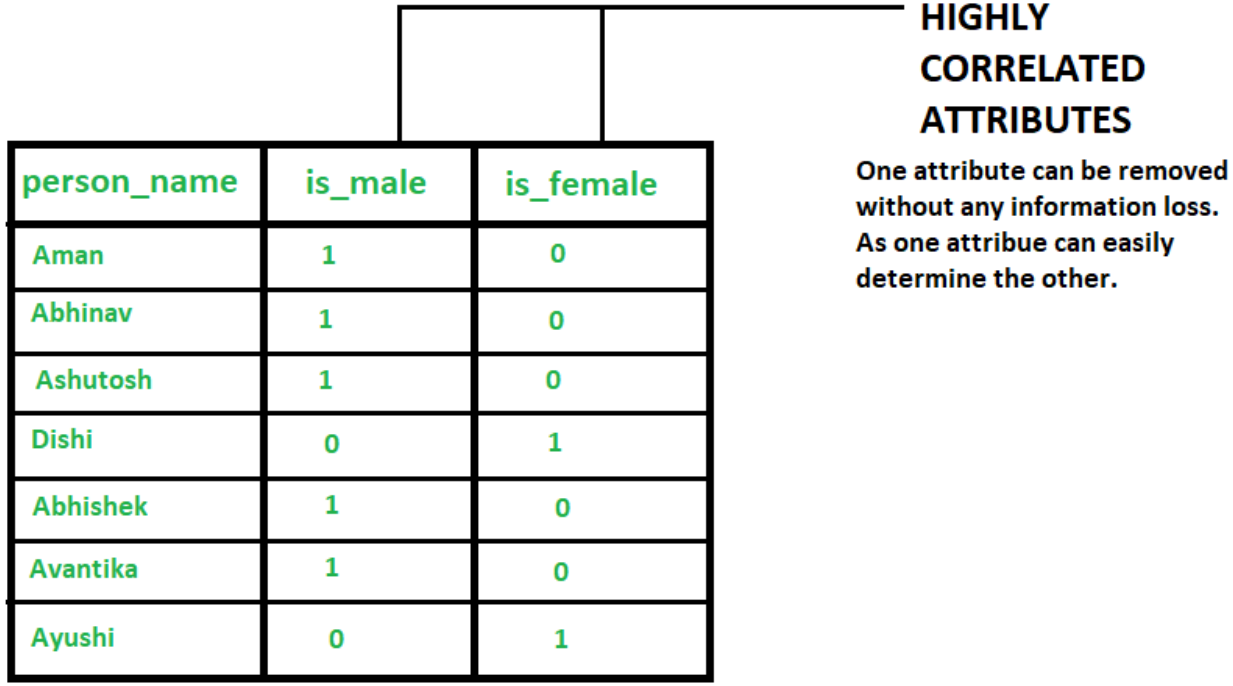
Uklanjanje nebitnih i redundantnih značajki

- Značajke su **nebitne** (engl. *irrelevant*) ako ne poboljšavaju uspješnost modela strojnog učenja (mogu, ali i ne moraju ga pogoršati)
- Vrste nebitnih značajki
  - **Monotone značajke**
  - **Konstantne značajke**
  - **Duplikati značajki**
  - **Nekorelirane značajke**
    - Korelacijski koeficijent između njih i ciljne značajke je jednak (ili vrlo blizak) nuli

Uklanjanje nebitnih i redundantnih značajki

- Značajke su **redundantne** (engl. *redundant*) ili statistički redundantne (engl. *statistically redundant*) ako **uz prisutnost drugih značajki** u modelu strojnog učenja one same ne poboljšavaju uspješnost modela
- Statistički redundantne značajke određuju se:
  - **Korelacijskom analizom**
  - **Markovljevim pokrivačem** (engl. ***Markov blanket***) – tema idućeg predavanja

Uklanjanje redundantnih značajki korelacijskom analizom

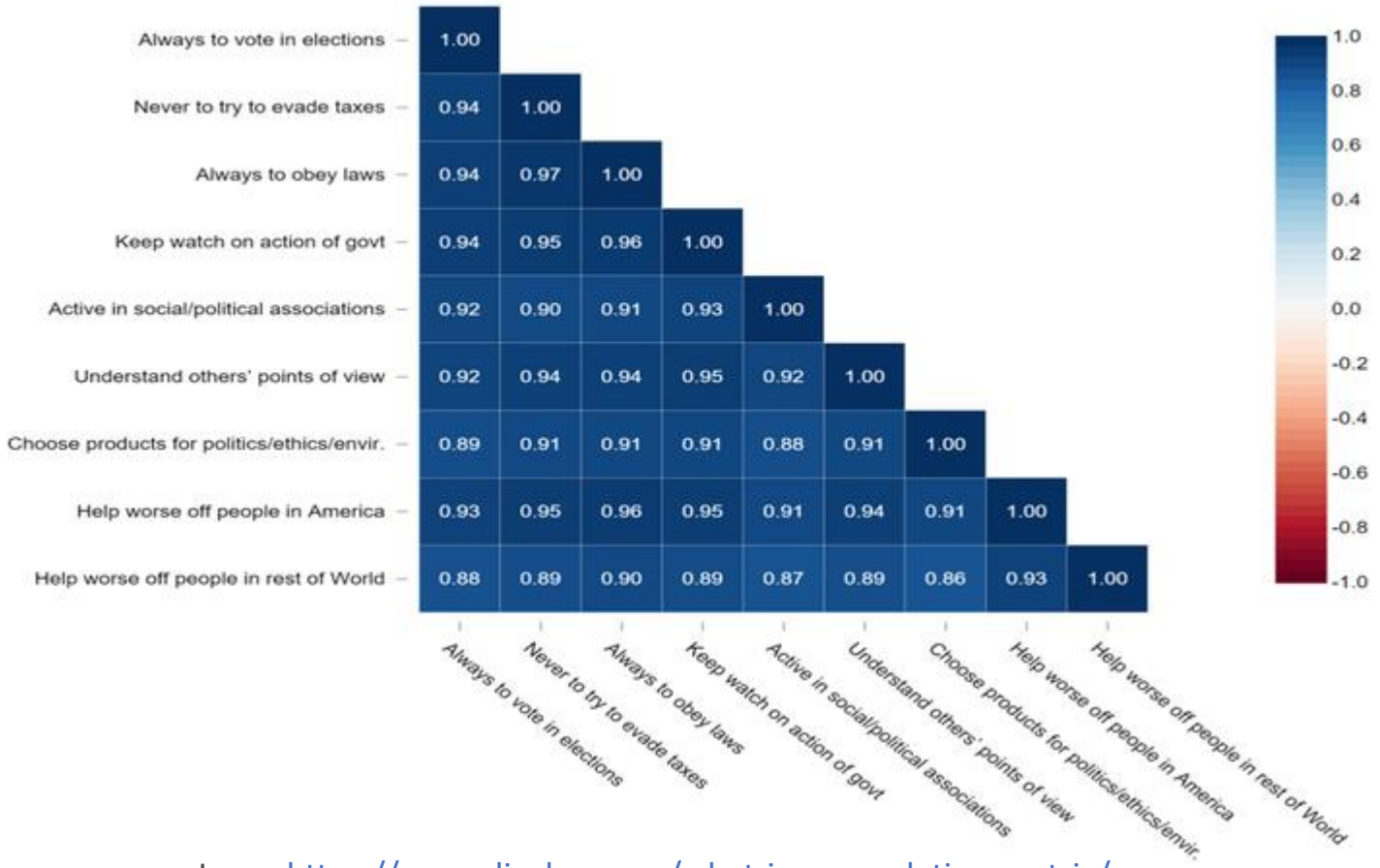


person_name	is_male	is_female
Aman	1	0
Abhinav	1	0
Ashutosh	1	0
Dishi	0	1
Abhishek	1	0
Avantika	1	0
Ayushi	0	1

Source: https://www.geeksforgeeks.org/redundancy-and-correlation-in-data-mining/

Uklanjanje redundantnih značajki korelacijskom analizom

- **Korelacijska matrica** – prikazuje korelaciju između svakih dviju značajki u skupu
- Ako je vrijednost korelacije vrlo visoka, **idealno 1**, odabire se jedna od značajki za uklanjanje
- Prag vrijednosti korelacijskog koeficijenta za odbacivanje neke značajke ovisi o domeni i cilju analize, ali obično je viši od 0.9



Izvor: https://www.displayr.com/what-is-a-correlation-matrix/

Poluautomatizirani pristup inženjerstvu značajki

Poluautomatizirani pristup inženjerstvu značajki

- **Izgradnja značajki** (engl. ***feature construction***)
- **Redukcija dimenzionalnosti** (engl. ***dimensionality reduction***)
- **Odabir značajki** (engl. ***feature selection***) – tema idućeg predavanja

Izgradnja značajki

- Fokus na **poboljšanju performanci**
- **Nema prokušane formule**, treba razumjeti domenu primjene i isprobati različite pristupe
- Iterativna primjena različitih operatora za izgradnju novih značajki
  - **Binarne varijable**: **Konjunkcija, disjunkcija, negacija**
  - **Numeričke varijable**: **Ekvivalencija, nejednakost, zbrajanje, oduzimanje, množenje, dijeljenje**
  - **Kombiniranje varijabli: *M* od *N* - varijabla poprima vrijednost 1 ako je barem *M* od *N* uvjeta (ostale varijable) istinito**
  - **Složenije transformacije podataka** -> vidjeti metode za redukciju dimenzionalnosti

Izgradnja značajki

- Podjela pristupa za izgradnju značajki:
  - **Zasnovani na podacima** – na temelju opaženih podataka (najčešće)
  - **Zasnovani na hipotezi** – na temelju prethodno generirane hipoteze (znanstvena istraživanja)
  - **Zasnovani na znanju** – na temelju domenskog znanja (pomoć eksperta je često nužna)
  - **Hibridni pristupi** – kombiniraju gornja tri pristupa

Izvor: http://www.ar.sanken.osaka-u.ac.jp/~motoda/papers/fdws02.pdf

Izgradnja značajki

- **Vrlo značajno** za postizanje boljih prediktivnih modela
- Može poslužiti i za **redukciju dimenzionalnosti**
- Dobivene značajke po potrebi se dijelom kasnije uklanjaju korištenjem algoritama za odabir značajki – ponekad postupak ide iterativno
- Neki od najboljih automatskih postupaka za izgradnju značajki zasnivaju se na **genetskom programiranju** s višestrukim stablima za predstavljanje značajki
  - Veze između čvorova stabala označavaju operatore
  - https://www.sciencedirect.com/science/article/abs/pii/S0031320319301815?via%3Dihub

Redukcija dimenzionalnosti

- **Problem:** velika dimenzionalnost (broj varijabli) u skupu podataka
- **Prokletstvo dimenzionalnosti**: podaci u velikom broju dimenzija postaju **rijetki**
  - Algoritmi za učenje teško se prilagođavaju rijetkim podacima što dovodi po slabe generalizacije
  - Potrebno je eksponencijalan broj primjeraka u odnosu na broj varijabli da se prostor napuči
- Cilj: smanjiti (reducirati) dimenzionalnost problema uz **zadržavanje inicijalne informacije** u podacima
- Dva moguća pristupa
  - Transformacije značajki – linearne i nelinearne
  - Odabir značajki

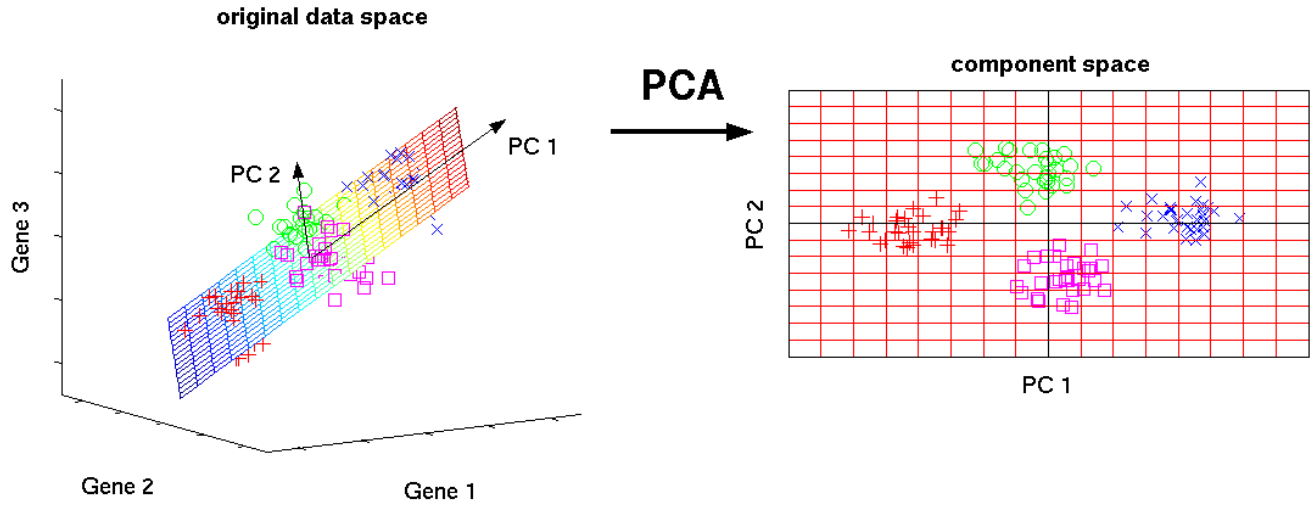
Redukcija dimenzionalnosti

- Ovdje razmatramo samo **neke** često korištene postupke
  - Linearni postupci smanjenja dimenzionalnosti (donekle moguće tumačenje):
    - **Analiza glavnih komponenti** (**engl. *Principal Component Analysis*, PCA**)
  - Nelinearni postupci smanjenja dimenzionalnosti (slabo moguće tumačenje):
    - **Višedimenzijsko skaliranje** (engl. ***Multidimensional Scaling*, MDS**)
    - **Ugradnja pomoću t-distribuiranog stohastičkog susjeda** (engl. ***t-distributed Stochastic Neighbor Embedding***)
    - **Uniformna aproksimacija i projekcija mnogostrukosti** (engl. ***Uniform Manifold Approximation and Projection*, UMAP**)
    - **Autoenkoderi** (engl. ***autoencoders***) – 10. predavanje

Tehnike učenja mnogostrukosti (engl. ***manifold learning***)

Analiza glavnih komponenti

- 1901., Karl Pearson
- Sinonim: Karhunen-Loèveova transformacija (u obradi signala)
- Tradicionalna najčešća metoda transformacije varijabli
- Tehnika **nenadziranog učenja**, ne razmatra ciljnu varijablu
- Korisno za:
  - **Vizualizaciju** visokodimenzionalnih numeričkih podataka u niskodimenzionalnom (2D ili 3D) prostoru
  - Otkrivanje **grupa** podataka
  - Smanjenje **šuma** u podacima – prvih nekoliko komponenti je otporno na šum



Analiza glavnih komponenti

- Analitički iskaz transformacije: **T = X W**
- **X** je matrica izvornih podataka dimenzija *n* x *m***,** gdje je *n* broj primjeraka, a *m* broj varijabli u skupu
- **W** je matrica vlastitih vektora težina dimenzije *m* x *m* koja se koristi za preslikavanje izvornih podataka u novi prostor značajki
- Matrica **W** dobiva se tako da se:
  - izvorni podaci najprije standardiziraju;
  - formira se matrica kovarijance ulaznih podataka;
  - računaju se vlastiti vektori (*eigenvectors*) matrice kovarijanci;
  - ortogonalni vlastiti vektori se normiraju u jedinične vektore

Analiza glavnih komponenti

- Glavne komponente su vlastiti vektori sortirani po apsolutnoj vrijednosti vlastitih vrijednosti
- Glavne komponente su izražene kao linearna kombinacija standardiziranih izvornih varijabli i to takvi da redom pokrivaju **najveću varijabilnost** u podacima, definiraju smjer novog prostora značajki

- Primjer glavnih komponenti za skup podataka **Iris**:
- PC1: -0.581petallength-0.566petalwidth-0.522sepallength+0.263sepalwidth
- PC2: 0.926sepalwidth+0.372sepallength+0.065petalwidth+0.021petallength

} Pokrivaju 95% varijance u podacima

- PC3: -0.721sepallength+0.634petalwidth+0.242sepalwidth+0.141petallength
- PC4: -0.801petallength+0.524petalwidth+0.262sepallength-0.124sepalwidth

Više o izračunu: https://askdatascience.com/652/how-calculate-covariance-matrix-and-principal-components

Višedimenzijsko skaliranje

- MDS je skup metoda koje predstavljaju različite **mjere sličnosti** između parova objekata u visokodimenzionalnom prostoru kao metričku **udaljenost** između točaka u niskodimenzionalnom prostoru
- Metrički MDS (engl. *metric multidimensional scaling*), definira mjeru **stresa**:

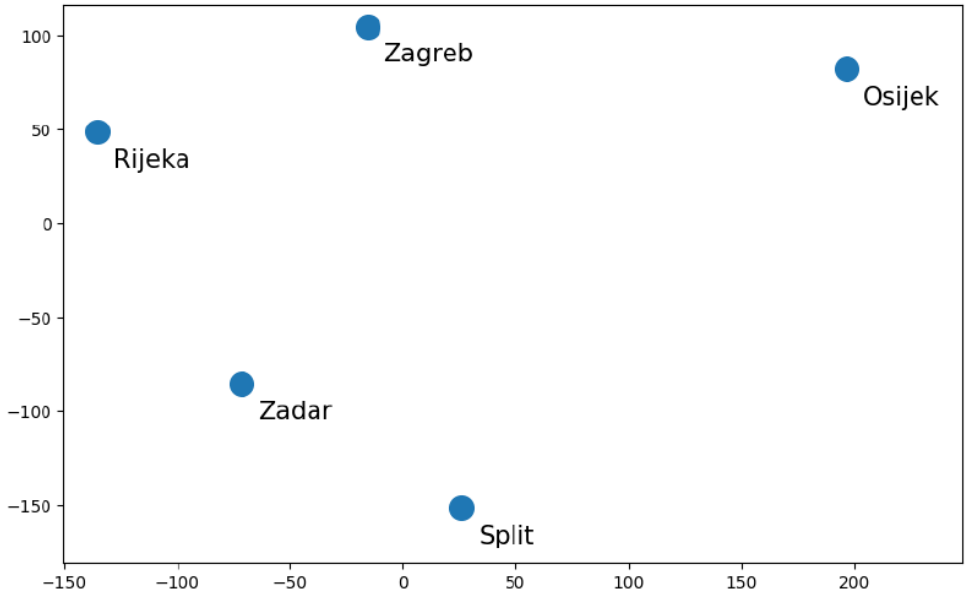
$$Stress_D(x_1, x_2, \ldots, x_N) = \sum_{i \neq j = 1..N} \left(d_{i,j} - \left\|x_i - x_j\right\|\right)^2,$$

D je matrica udaljenosti (nesličnosti) (engl. *dissimilarity matrix*) čiji su elementi $d_{i,j}$ udaljenosti objekata u *M* dimenzija **visokodimenzionalnog** prostora, a *N* je **ciljna dimenzija** problema

- Cilj je pronaći *M* vektora $x_1, x_2, \ldots, x_M \in \mathbb{R}^N$ takvih da je **stres minimalan**, tj. da $d_{i,j} \approx \left\|x_i - x_j\right\|$
- Obično se formulira kao **optimizacijski** problem čije se rješenje nalazi korištenjem **numeričkih metoda**
- Ako je *N* malen (2 ili 3), tada možemo vizualizirati ovo preslikavanje

Višedimenzijsko skaliranje

Grad	Zagreb	Split	Rijeka	Osijek	Zadar
Zagreb	0	259	132	213	198
Split	259	0	257	289	118
Rijeka	132	257	0	333	148
Osijek	213	289	333	0	316
Zadar	198	118	148	316	0



- Primjer: Udaljenost između gradova u RH zadana u matrici *D*, dimenzije *M* = 5, ciljna dimenzija potrebna za vizualizaciju: *N* = 2
- Pronalazimo *M* = 5 vektora u 2D prostoru takvih da je $d_{i,j} \approx \|x_i - x_j\|$

t-SNE

- Maaten i Hinton 2008.
- Traži niskodimenzionalnu strukturu takvu da svojstva grupiranja u višoj dimenziji ostanu sačuvana
- Sličnost u visokodimenzionalnom prostoru predstavljena je **Gaussovim zajedničkim vjerojatnostima** euklidskih udaljenosti dvaju objekata p_{ij}
  - Sličnost točke x_j prema točki x_i je uvjetna vjerojatnost $p_{j|i}$ da bi x_i izabrao kao svojeg susjeda x_j ako se susjedi biraju proporcionalno (lokalnoj) gustoći vjerojatnosti Gaussove funkcije centrirane u x_i ; gdje je $p_{ij} = (p_{j|i} + p_{i|j})/2N$, N je broj primjeraka, $\sum p_{i,j} = 1$
- U ugrađenom prostoru (najčešće 2D ili 3D) sličnost između točaka mjeri se **zajedničkim vjerojatnostima Studentovim t-razdiobama** euklidskih udaljenosti q_{ij} (koje su plosnatije pa su točke razvučenije)
- **Kullback-Leiblerova divergencija** između zajedničkih vjerojatnosti u izvornom prostoru i ugrađenom prostoru:

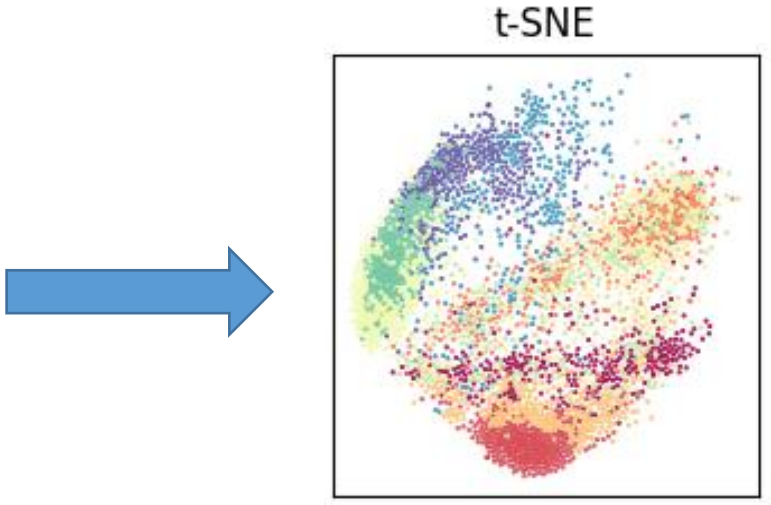
$$KL(P||Q) = \sum_{i \neq j} p_{i,j} \log \frac{p_{i,j}}{q_{i,j}}$$

minimizira se gradijentnim spustom kako bi se dobilo najbolje preslikavanje

t-SNE

- Metoda je računski **vrlo zahtjevna** ali daje izvrsne rezultate
- Metoda nije primijenjiva na testni skup, može se koristiti samo za vizualizaciju bliskosti na čitavom skupu podataka bez pamćenja modela
- Prednost metode je da **čuva lokalnu bliskost točaka** pri preslikavanju iz više dimenzije u nižu i može modelirati nelinearne odnose između varijabli (za razliku od npr. PCA)

Fashion-MNIST: 28x28, 60000 primjera, 10 klasa



Izvor: Eugen Vušak, „Tehnike učenja višestrukosti za povećanje učinkovitosti analize koja koristi sporedna svojstva kriptografskih uređaja" diplomski rad, FER, 2020.

UMAP

- McInnes i Healy, 2018.
- Matematički bolje formulirana varijanta t-SNE-a
- **Mnogo brža metoda od t-SNE-a** i bolje čuva globalnu povezanost podataka, korisna za visokodimenzionalne podatke i može se primijeniti na testni skup podataka
- Pretpostavke:
  - primjerci su uniformno distribuirani po mnogostrukosti (aproksimativno)
  - mnogostrukost je lokalno povezana (ne postoji npr. točka koja je sama udaljena od svih)
- UMAP gradi **graf susjedstva** pomoću brzog algoritma **spusta najbližih susjeda** (engl. *Nearest Neighbor Descent*, NN-descent)
  - omogućuje se brz pronalazak najbližih susjeda i izgradnju grafa susjedstva, pri čemu parametar broja susjeda *k* određuje koliko se globalne strukture mnogostrukosti uzima u obzir pri izgradnji grafa

UMAP

- Umjesto Studentove t-distribucije za modeliranje udaljenosti u niskodimenzionalnom prostoru, UMAP koristi sličnu obitelj krivulja definiranu s:

$$q_{ij} = \left(1 + a\left(y_i - y_j\right)^{2b}\right)^{-1}$$

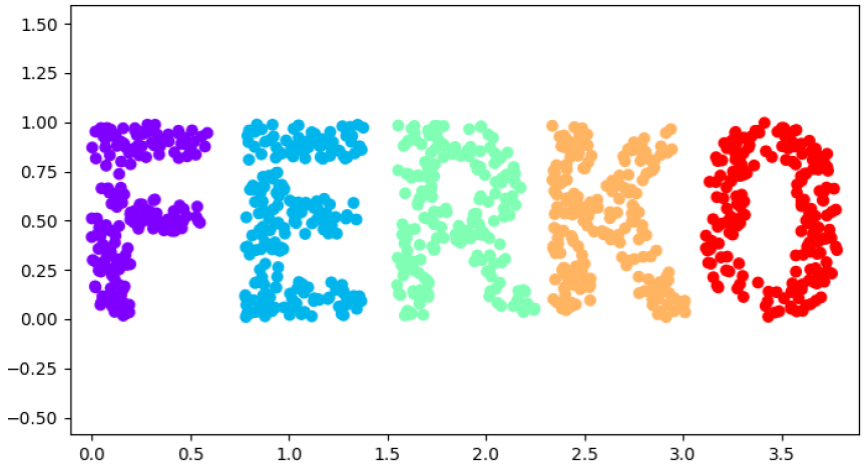
  - Parametre *a* i *b* pronalazi pomoću nelinearne regresije metodom najmanjih kvadrata na dijelovima funkcije, teži se smanjenju gustog lokalnog grupiranja (postoji parametar minimalne udaljenosti točaka)

- Koristi binarnu unakrsnu entropiju (*CE*) kao funkciju gubitka umjesto KL divergencije te je minimizira stohastičkim gradijentnim spustom (umjesto običnog, radi brzine):

$$CE(X,Y) = \sum_i \sum_j \left[p_{ij}(X) \log\left(\frac{p_{ij}(X)}{q_i j(Y)}\right) + (1 - p_{ij}(X)) \log\left(\frac{1 - p_{ij}(X)}{1 - q_{ij}(Y)}\right)\right]$$

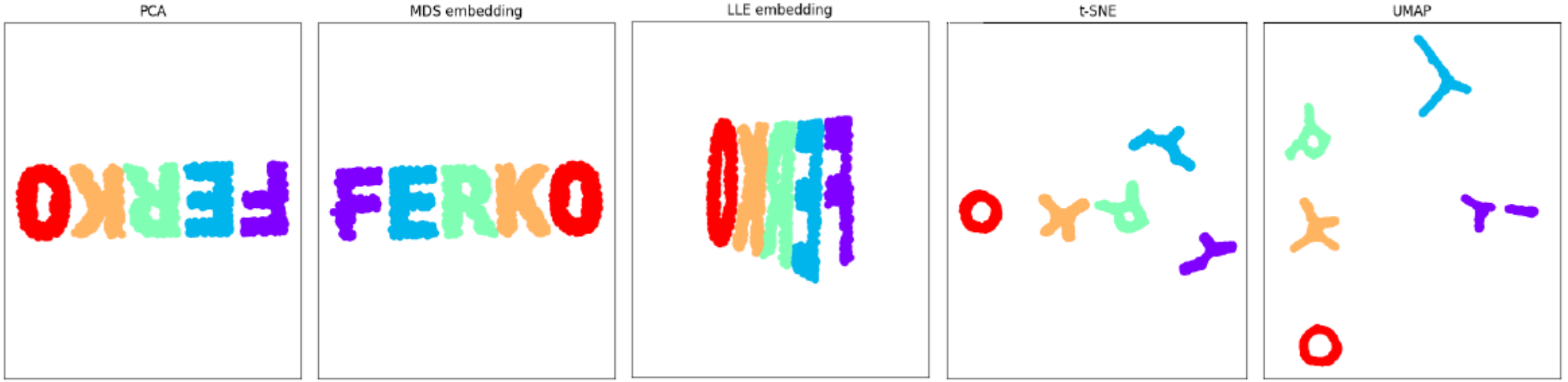
- Prvi član unutar sume omogućuje preslikavanje grupa ispravno, dok drugi član omogućuje preslikavanje razmaka ispravno

Vizualizacija



t-SNE i UMAP, još vizualizacija:
https://www.nature.com/articles/nbt.4314

UMAP vs PaCMAP (2021.), vizualizacije:
https://jlmelville.github.io/smallvis/pacmap-umap.html



Izvor: Eugen Vušak, „Tehnike učenja višestrukosti za povećanje učinkovitosti analize koja koristi sporedna svojstva kriptografskih uređaja" diplomski rad, FER, 2020.

Zaključak

- Značajke omogućuju lakše razumijevanje izvornih podataka
- Značajke se mogu izračunati (izlučiti) na jednostavne ili složene načine, ovisno o domeni primjene
- Za potrebe analize podataka često je nužno transformirati podatke, što se radi ovisno o vrsti podatka
- Odabir značajki i smanjenje dimenzionalnosti načini su kako se inicijalni skup podataka može pojednostaviti za postupak modeliranja
- Tehnike smanjenja dimenzionalnosti teže što boljoj reprezentaciji visokodimenzionalnog skupa podataka u niskodimenzionalnom prostoru, a mogu pridonijeti i smanjenju šuma u podacima