Klasteriranje podataka

Matej Mihelčić

Prirodoslovno-matematički fakultet, Sveučilište u Zagrebu

matmih@math.hr

04. svibnja, 2026.

Klasteriranje (grupiranje) podataka

- Klasteriranje je **nenadzirani** zadatak. Uglavnom nemamo zadanu ciljnu varijablu koja definira kojoj grupi pripadaju entiteti iz skupa podataka.
- Osim nepostojanja varijable cilja, tablični podaci koji opisuju podatke koje koristimo za klasteriranje se ne razlikuju od nadziranih podataka (binarni, kategorijski, numerički atributi).

Definicija

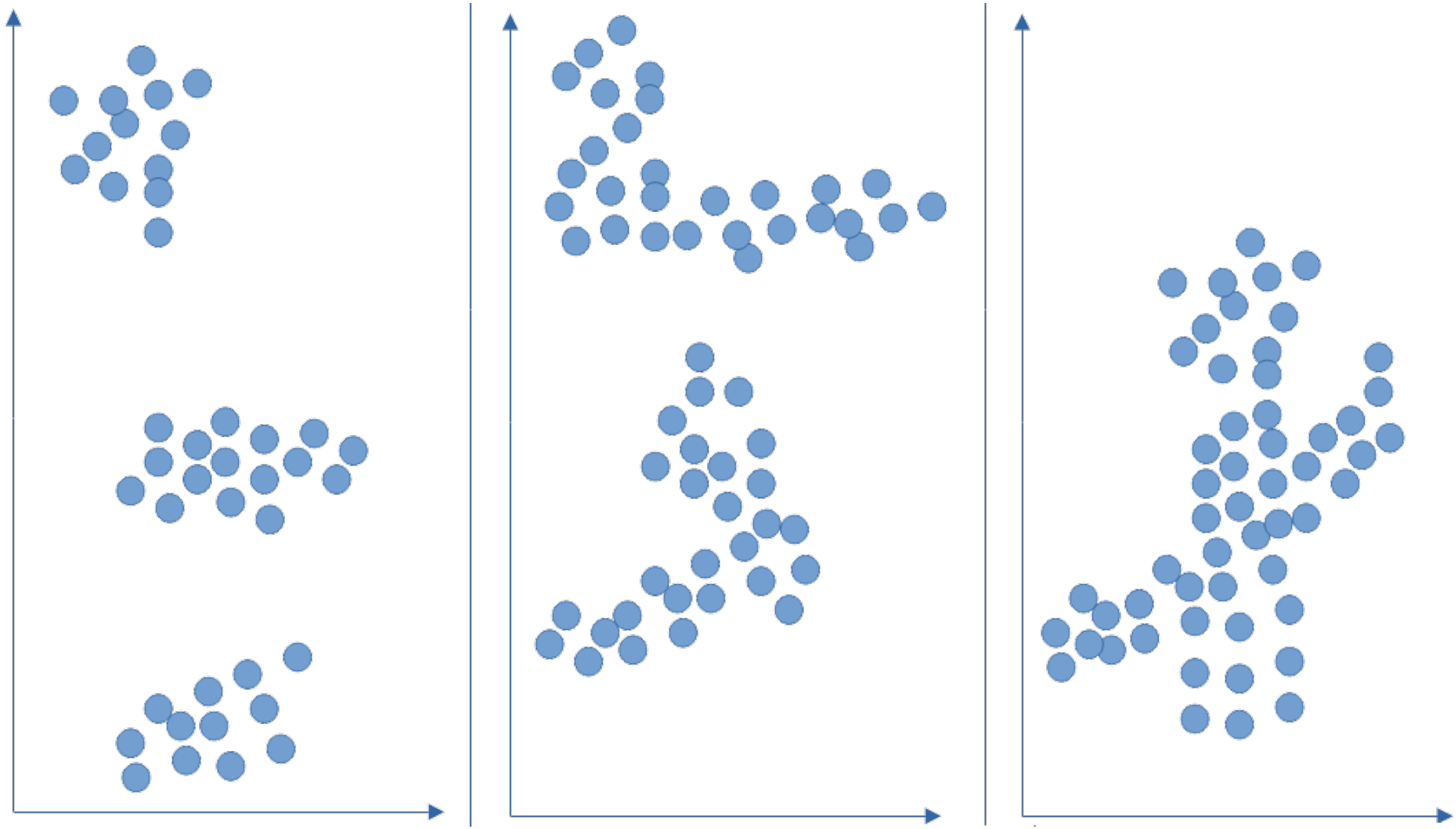
Za zadani skup točaka sadržanih u skupu podataka D, zadatak klasteriranja je particionirati ih u grupe, gdje svaka grupa sadrži što je moguće sličniji podskup točaka iz D.

- Gornja (klasična) definicija zadatka klasteriranja ne obuhvaća sve moguće varijante zadatka kao npr. **preklapajuće klasteriranje**, **neizrazito klasteriranje** itd.

Klasteriranje (grupiranje) podataka

- Sam zadatak nije precizno matematički definiran, stoga je izrazito teško odrediti što je **dobro/odgovarajuće** klasteriranje.
- Unatoč tome, grupiranje je jedan od ključnih zadataka nenadziranog učenja i dubinske analize podataka s preko 100 različitih razvijenih algoritama.
- Iz aspekta dubinske analize podataka, klasteriranje nam daje uvid u to koji entiteti su međusobno slični s obzirom na svoja svojstva.
  - Možemo dobiti uvid u veličine grupa, potencijalno njihov oblik.
  - Daljnjom statističkom analizom možemo odrediti neka svojstva koja definiraju članove svake pojedine grupe.
- Grupiranje možemo promatrati kao **sažeti model podataka**.

Klasteriranje (grupiranje) podataka



Klasteriranje (grupiranje) podataka

Domene u kojima nailazimo na problem klasteriranja:

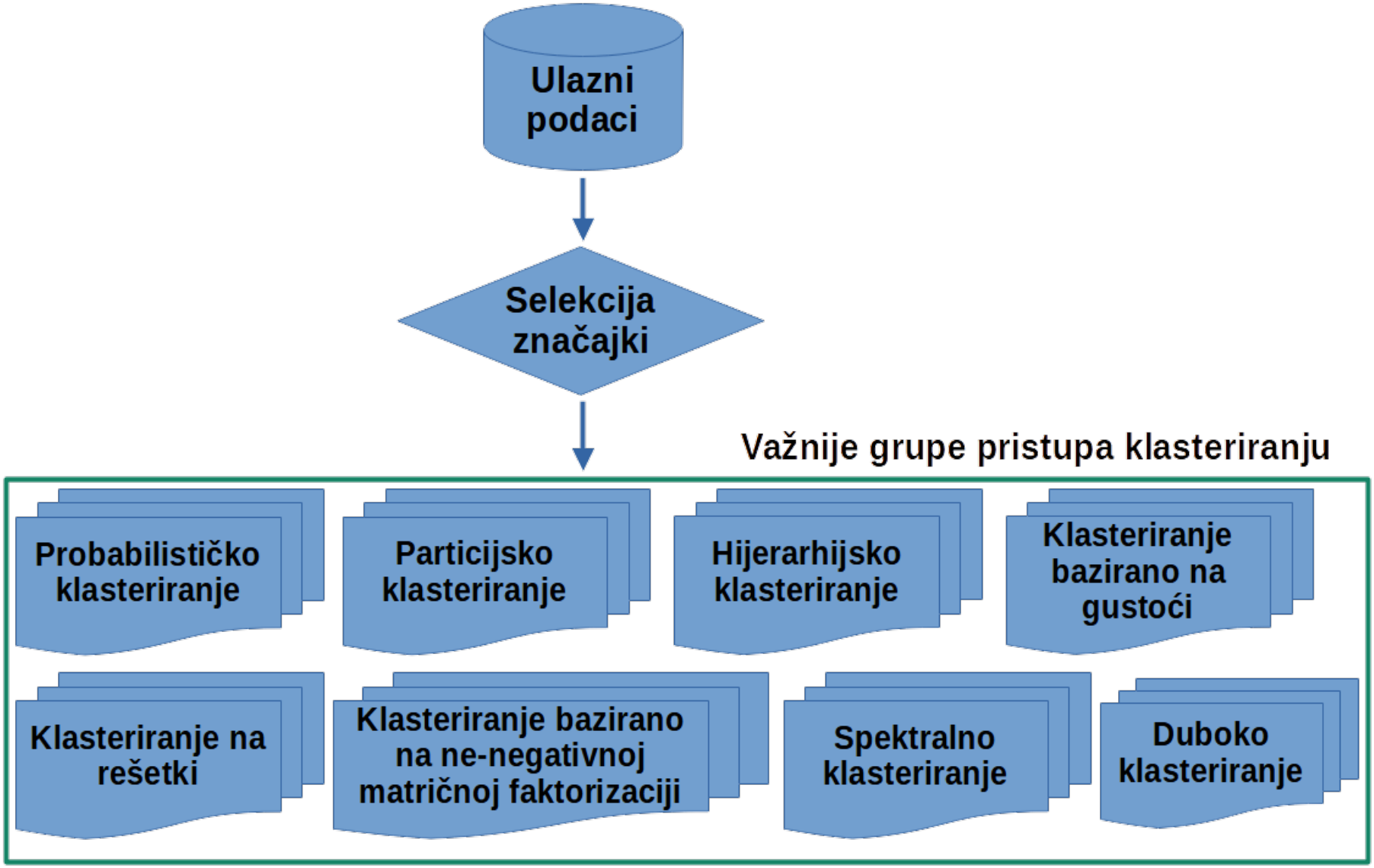
- Klasteriranje se često provodi kao korak algoritama strojnog učenja ili korak nekih drugih zadataka dubinske analize podataka. Npr. često se javlja i kod analize stršećih vrijednosti (outliera).
- Klasteriranje se koristi i u algoritmima za preporučivanje. Primarno pronalazimo grupe korisnika koje dijele preferencije prema sadržaju određene vrste.
- Kod zadataka segmentacije kupaca, grupiramo slične kupce u grupe.
- Često određene vrste klasteriranja koristimo i za sumarizaciju podataka, odnosno stvaranje kompaktnih reprezentacija.
- Klasteriranje se koristi kao važan korak u određivanju izmjene trenda kod zadataka koji uključuju tokove podataka. Dosta često takve situacije susrećemo u primjenama na društvene interakcije preko Interneta.

Klasteriranje (grupiranje) podataka

Domene u kojima nailazimo na problem klasteriranja (nastavak):

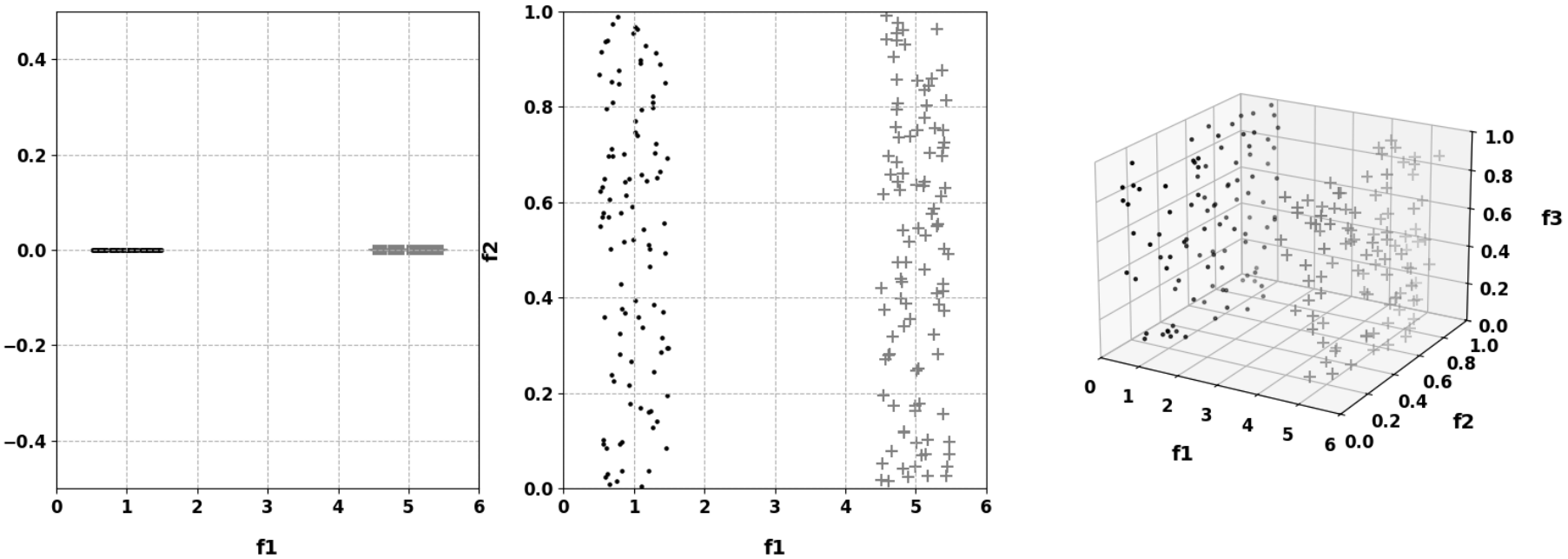
- Klasteriranje susrećemo i kod analize multimedijskih podataka. Npr. određivanje sličnih segmenata glazbe, sličnih fotografija.
- Klasteriranje se često upotrebljava za analizu bioloških podataka. Često se radi o klasteriranju sekvenci ili traženju povezanih čvorova u mreži.
- Kod analize društvenih mreža, struktura društvene mreže se koristi za otkrivanje važnih zajednica. Otkrivanje zajednica nam pruža uvid i bolje razumijevanje društvenih struktura. Klasteriranje možemo koristiti i za sumarizaciju kod društvenih mreža.

Klasteriranje (grupiranje) podataka



Klasteriranje - selekcija značajki

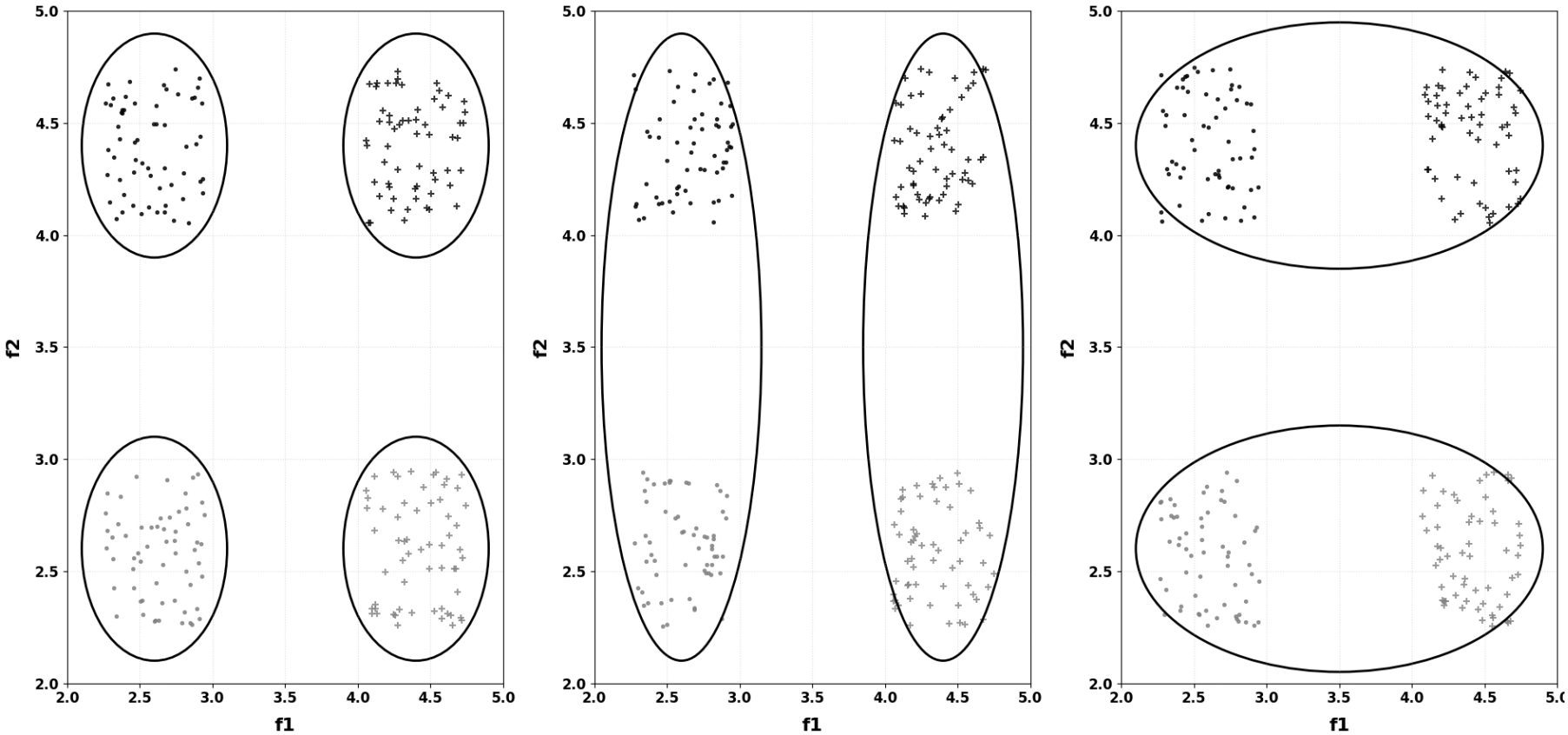
- Eliminacija irelevantnih značajki neće negativno utjecati na točnost klasteriranja, no znatno će reducirati memorijsku potrošnju i vrijeme računanja.



- Na gornjoj slici vidimo relevantnu značajku f_1 i dvije irelevantne značajke f_2 i f_3.

Klasteriranje - selekcija značajki

- Različite relevantne značajke mogu proizvesti različita klasteriranja.



- Klasteriranje korištenjem značajki f_1 i f_2 (lijevo), korištenjem f_1 (sredina) i korištenjem f_2 (desno).

Klasteriranje - selekcija značajki

- Različiti podskupovi relevantnih značajki rezultiraju različitim klasteriranjem. To pomaže otkrivanju različitih skrivenih uzoraka u podacima.
- Zbog toga različite tehnike klasteriranja koriste selekciju značajki koja eliminira irelevantne i redundantne značajke uz zadržavanje relevantnih značajki. To poboljšava efikasnost klasteriranja i kvalitetu.
- Metode za selekciju značajki se dijele na **filter metode**, **metode omotača** i **hibridne metode**.
- Filter metode evaluiraju značajke neovisno o algoritmu klasteriranja. Nemaju pristranost prema algoritmu klasteriranja i obično su brze.
- Metode omotača biraju razne podskupove atributa, računaju klastere nekim algoritmom za klasteriranje, te procjenjuju kvalitetu podskupova značajki na temelju kvalitete dobivenih klastera. Za fiksni algoritam klasteriranja omogućavaju dobivanje boljih klastera.

Klasteriranje - selekcija značajki

- Kod hibridnih modela koristimo kriterije filter metoda za selekciju podskupova značajki (kandidata), te na njima provodimo metode omotača. Time smanjujemo računalni trošak računanja metoda omotača.

Spektralan odabir značajki

- SPEC spada u filter metode odabira značajki.
- Određuje značajnost značajke određujući njezinu konzistentnost sa spektrom matrice dobivene iz matrice sličnosti S.
- SPEC koristi radijalne bazne funkcije (eng. radial-base function, RBF) kao funkciju sličnosti između dva entiteta x_i i x_j.
- Matrica S se računa kao: $S_{i,j} = e^{-\frac{\|x_i - x_j\|^2}{2\sigma^2}}$ (Gausova jezgra).
- Graf G konstruiramo iz S tako da bridovi u G imaju težine koje odgovaraju sličnostima iz S.
- Iz G konstruiramo matricu susjedstva W, te dijagonalnu matricu stupnjeva $\bar{D}$, gdje $\bar{D}_{i,i} = \sum_{j=1}^{n} W_{i,j}$.
- Korištenjem W i $\bar{D}$ konstruiramo Laplace-ovu matricu $L = \bar{D} - W$ i normaliziranu Laplace-ovu matricu $\mathcal{L} = \bar{D}^{-\frac{1}{2}} L \bar{D}^{-\frac{1}{2}}$.

Spektralan odabir značajki

Algoritam Spektralan odabir značajki (SPEC)

Ulaz:

D: skup podataka

$\Psi \in \{\Psi_1, \Psi_2, \Psi_3\}$: funkcije za rangiranje značajki

n: broj entiteta

Izlaz:

F: rang lista značajki

Konstruiraj matricu sličnosti S iz D

Konstruiraj graf G iz S

Konstruiraj W iz S

Konstruiraj $\bar{D}$ iz W

Definiraj L i $\mathcal{L}$

for svaki vektor značajke f_i **do**

$$\hat{f}_i \leftarrow \frac{\bar{D}^{-\frac{1}{2}} f_i}{\|\bar{D}^{-\frac{1}{2}} f_i\|}$$

$F_i \leftarrow \psi(\hat{f}_i)$

end for

Rangiraj F prema ψ

Spektralan odabir značajki

- Za vektor značajke f_i, definiramo $\tilde{f}_i = D^{\frac{1}{2}} f_i$, te $\hat{f}_i = \frac{\tilde{f}_i}{\|\tilde{f}_i\|}$.
- $\psi_1(F_i) = \hat{f}_i^{\,t} \mathcal{L} \hat{f}_i$ - mjeri koliko f varira lokalno, koliko je gladak nad G. Manja vrijednost mjere označava da je značajka bolja za klasteriranje.
- $\psi_2(F_i) = \frac{\hat{f}_i^{\,t} \mathcal{L} \hat{f}_i}{1 - \hat{f}_i^{\,t} \xi_0}$ - koristimo u slučaju kada je $\hat{f}_i$ blizak svojstvenoj vrijednosti ξ_0 (najmanja, trivijalna svojstvena vrijednost). U tom slučaju manja vrijednost od $\psi_1(F_i)$ ne označava bolju separabilnost od F_i. Stoga normaliziramo sa sumom težina udjela ostalih svojstvenih vrijednosti u ocjeni dobrote značajke. Sada manja vrijednost označava korespodenciju značajke F_i s netrivijalnim svojstvenim vektorima koji odgovaraju malim svojstvenim vrijednostima.
- $\psi_3(F_i) = \sum_{j=1}^{k-1} (2 - \lambda_j) \alpha_j^2$ - koristimo uz zadani broj grupa k, veća vrijednost od ψ_3 označava bolju separabilnost značajke.

Klasteriranje - particijsko klasteriranje

- Traže grupe optimiziranjem **specifične funkcije cilja**, te **iterativnim poboljšavanjem kvalitete particija**.
- Obično sadrže korisničke parametre koje određuju način odabira točaka prototipa koje reprezentiraju klaster.
- Zbog toga se zovu i metodama klasteriranja **baziranim na prototipovima**.
- K-means klastering je najpoznatija i najkorištenija metoda particijskog klasteriranja.
- Minimizacija funkcije dobitka K-means algoritma je $\mathcal{NP}$-težak problem.

Klasteriranje K-Means algoritmom

Algoritam K-Means algoritam

Izaberi K točaka kao inicijalne centroide.
repeat
  Formiraj K klastera dodijeljujući svaku točku njegovom najbližem centroidu.
  Preračunaj centroid svakog klastera.
until uvjet konvergencije nije zadovoljen

- K-means algoritam može koristiti raspon mjera udaljenosti za računanje centroida.
- Izbor mjere utječe na dodijeljivanje točaka centroidima i kvalitetu konačnog rješenja.
- Možemo koristiti Manhattan udaljenost (L_1 normu), Euklidsku udaljenost (L_2 normu), kosinusovu sličnost, itd. Euklidska udaljenost je najčešći izbor.

Klasteriranje K-Means algoritmom

- Različite vrijednosti parametra K i izbor mjera udaljenosti će dati različita klasteriranja.
- Funkcija cilja koju koristi K-means algoritam je **suma kvadratne pogreške** ili **rezidualna suma kvadrata**.
- Za zadani skup podataka $D = (x_1, x_2, \ldots, x_n)$ označimo klasteriranje dobiveno K-means algoritmom s $C = \{C_1, C_2, \ldots, C_k, \ldots, C_K\}$.
- Suma kvadratne pogreške klasteriranja se definira kao

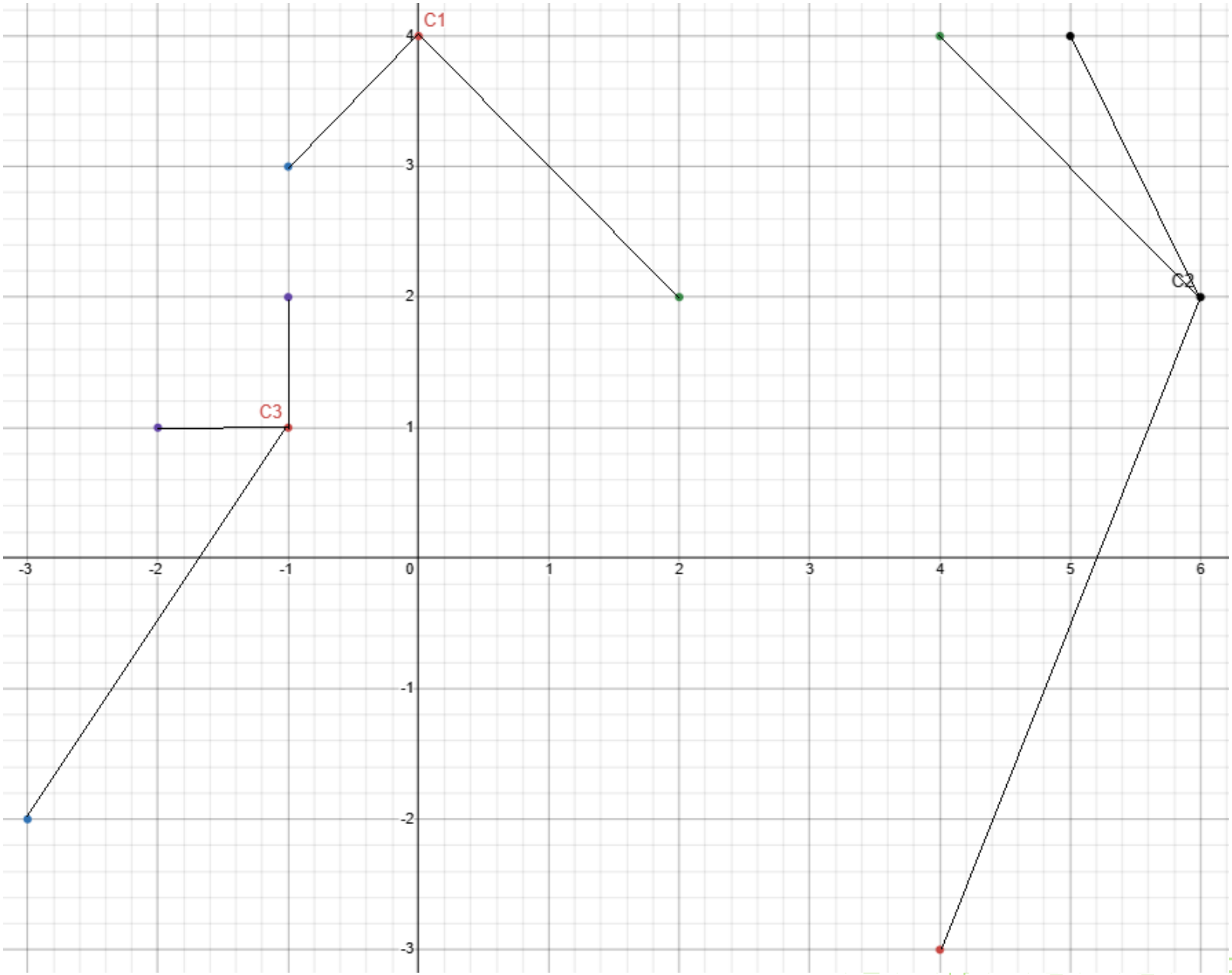
$$SKP(C, D) = \sum_{k=1}^{K} \sum_{x_i \in C_k} \|x_i - c_k\|^2.$$

- $c_k = \dfrac{\sum_{x_i \in C_k} x_i}{|C_k|}$.

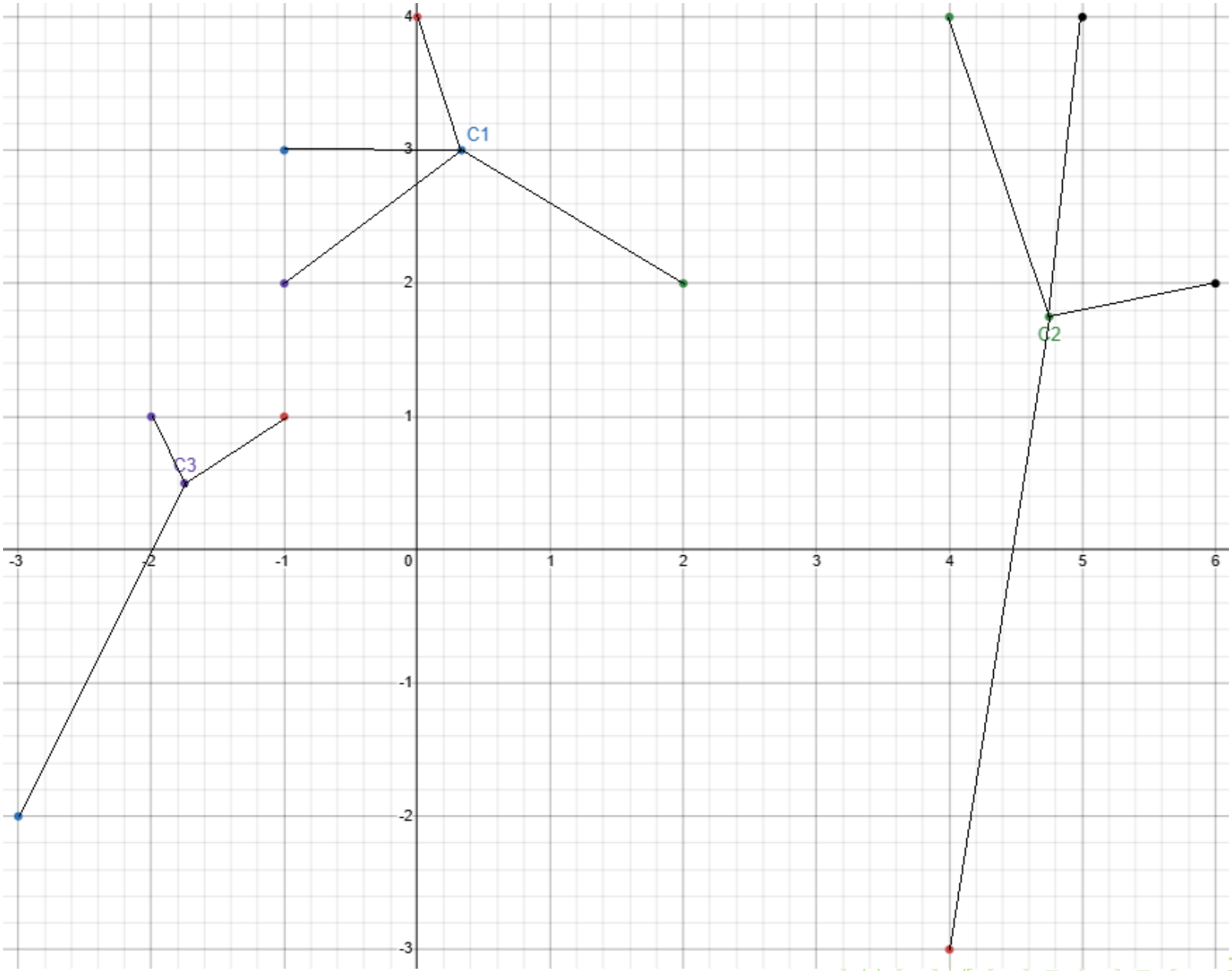
Klasteriranje K-Means algoritmom - primjer

- Pretpostavimo da želimo grupirati 11 točaka u dvodimenzionalnom prostoru s ko-ordinatama: $P_1 = (0, 4), P_2 = (-1, 3), P_3 = (2, 2), P_4 = (-2, 1), P_5 = (6, 2), P_6 = (4, -3), P_7 = (-3, -2), P_8 = (4, 4), P_9 = (-1, 2), P_{10} = (5, 4), P_{11} = (-1, 1)$.
- Pretpostavimo da je $K = 3$ i $C_1 = P_1$, $C_2 = P_5$ i $C_3 = P_{11}$.
- Pretpostavmo da je mjera udaljenosti standardna Euklidska udaljenost točaka.
- Dodijelimo točke najbližem centroidu: $C_1 : \{P_1, P_2, P_3\}$, $C_2 = \{P_5, P_6, P_8, P_{10}\}$, $C_3 = \{P_4, P_7, P_9, P_{11}\}$
- Izračunamo ponovo centroide: $C_1 = (0.33\dot{3}, 3)$, $C_2 = (4.75, 1.75)$, $C_3 = (-1.75, 0.5)$.
- Dodijelimo točke najbližem centroidu: $C_1 : \{P_1, P_2, P_3, P_9\}$, $C_2 = \{P_5, P_6, P_8, P_{10}\}$, $C_3 = \{P_4, P_7, P_{11}\}$
- Izračunamo ponovo centroide: $C_1 = (0, 2.75)$, $C_2 = (4.75, 1.75)$, $C_3 = (-2, 0)$.

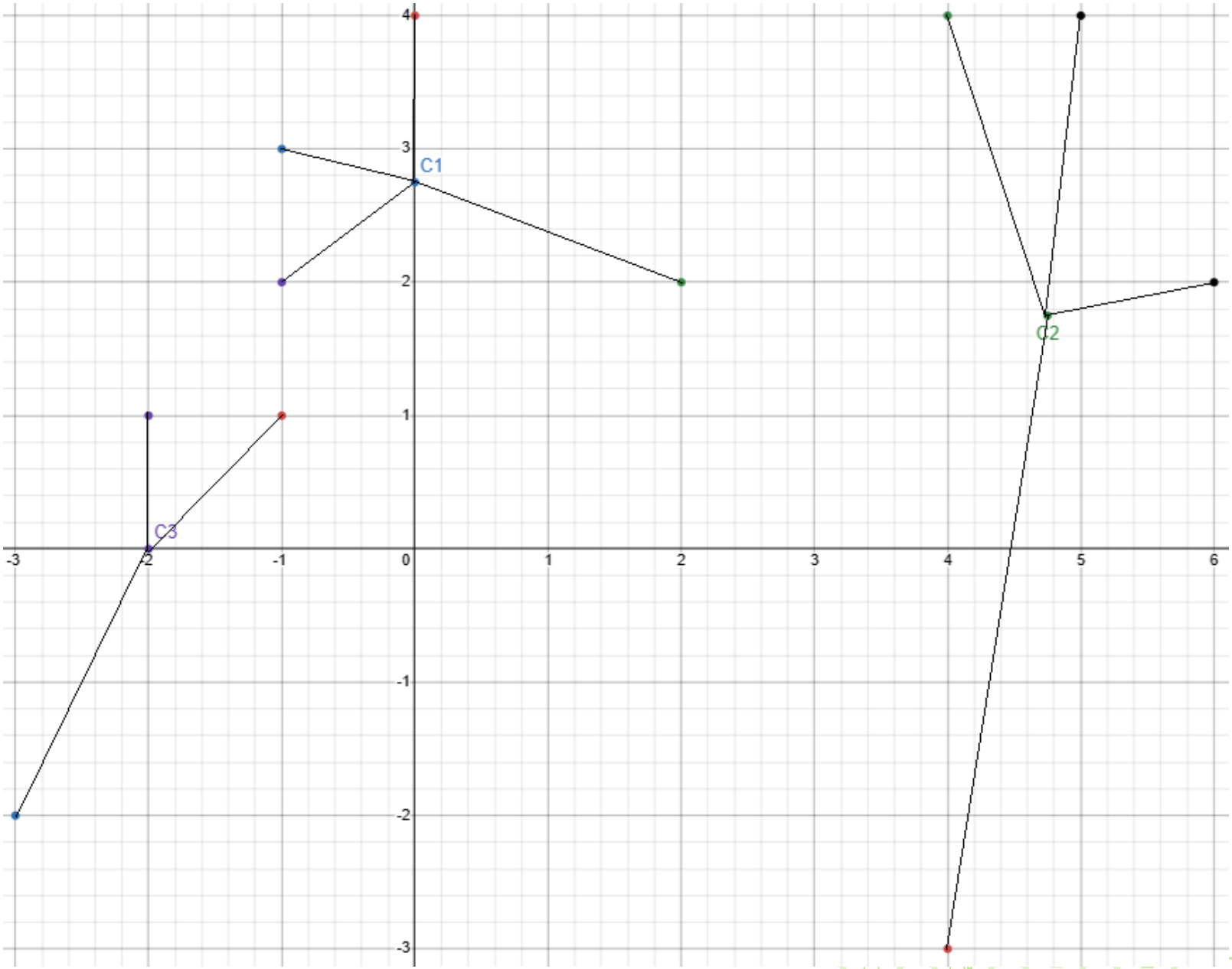
Klasteriranje K-Means algoritmom - primjer



Klasteriranje K-Means algoritmom - primjer



Klasteriranje K-Means algoritmom - primjer



O funkciji cilja K-Means algoritma

- K-means algoritam minimizira sumu kvadratne pogreške.
- Pokazat ćemo da izabirom srednje vrijednosti točaka za centroid u svakom klasteru zapravo minimiziramo sumu kvadratne pogreške.
- Označimo s C_k k-ti klaster, x_i je točka iz C_k, a c_k je srednja vrijednost k-tog klastera.
- Točku ekstrema određujemo deriviranjem izraza srednje kvadratne pogreške po varijabli c_k i izjednačavanjem s nulom. Zanima nas za kakav c_k se postiže vrijednost ekstrema.
- $SKP(C, D) = \sum_{k=1}^{K} \sum_{x_1 \in C_k} (c_k - x_i)^2.$
- $\frac{\partial}{\partial c_j} SKP(C, D) = \frac{\partial}{\partial c_j} \sum_{k=1}^{K} \sum_{x_i \in C_k} (c_k - x_i)^2 = \sum_{k=1}^{K} \sum_{x_i \in C_j} \frac{\partial}{\partial c_j} (c_j - x_i)^2$

O funkciji cilja K-Means algoritma

- $\frac{\partial}{\partial c_j} SKP(C, D) = \sum_{x_i \in C_j} 2 * (c_j - x_i) = 0$
- $\sum_{x_i \in C_j} 2 * (c_j - x_i) = 0 \Rightarrow |C_j| \cdot c_j = \sum_{x_i \in C_j} x_i \Rightarrow c_j = \frac{\sum_{x_i \in C_j} x_i}{|C_j|}$
- Iz toga slijedi da je najbolji izabir centroida za minimiziranje srednje kvadratne pogreške upravo srednja vrijednost.
- Srednja kvadratna pogreška monotono pada kroz iteracije algoritma. To je posljedica konvergencije algoritma u lokalni minimum.

K-Means algoritam

- Faktori koji utječu na performanse K-means algoritma su:
  - Izabir incijalnih centroida.
  - Odabir broja klastera K.
- Postoje brojni postupci odabira inicijalnih centroida:
  - na slučajan način
  - računajući gustoću susjeda i separabilnost među inicijalnim kandidatima
  - korištenjem sume kvadratne pogreške za evaluaciju udaljenosti klastera
  - uzorkovanjem i primjenom K-means algoritama na uzorcima, te određivanjem inicijalnih centroida iz skupa kandidata dobivenih na uzorcima
  - iterativnim izabirom, gdje se prvi centroid bira na slučajan način, drugi tako da je maksimalno udaljen od prvog itd.

Procjena broja klastera

Postoje razni kriteriji za procjenu broja klastera K, npr:

- Calinski–Harabasz indeks - biramo k koji maksimizira ovaj indeks.
- Gap statistika - stvaramo B skupova podataka s istim rasponom podataka kao u originalnom skupu.
  $Gap(K) = \frac{1}{B} \times \sum_{b \in \mathcal{B}} \log(DUK(b,k)) - \log(DUK(D,k))$. Tražimo najmanju vrijednost K za koju vrijedi $Gap(K) \geq Gap(K+1) - s_{k+1}$, gdje je s_{k+1} procjena standardne devijacije $\log(DUK(b,k+1))$.
- Akaike informacijski kriterij (eng. Akaike Information Criterion AIC). Za skup podataka s M entiteta, Akaike kriterijem određujemo K računanjem: $K = argmin_K(SKP(K) + 2MK)$.
- Bayesov informacijski kriterij (eng. Bayesian Information Criterion BIC). Definira se kao: $BIC = \frac{-2 * \ln(L)}{N} + \frac{K * \ln(N)}{N} = \frac{1}{N} \cdot \ln\left(\frac{N^K}{L^2}\right)$

Procjena broja klastera

- Bayesov informacijski kriterij (nastavak). N označava broj točaka u skupu, K broj klastera, L vjerodostojnost (eng. likelihood). Biramo K koji minimizira BIC.
- Koeficijent siluete - za danu točku x_i računamo prosječnu udaljenost od svih točaka unutar klastera (označimo s a_i), te prosječnu udaljenost do svih točaka izvan klastera (b_i). Koristeći te vrijednosti računamo: $S = \frac{\sum_{i=1}^{N} \frac{b_i - a_i}{\max(a_i, b_i)}}{N}$. Viša vrijednost koeficijenta siluete označava bolje klasteriranje.

Varijante K-means algoritma

- K-medoid - umjesto srednje vrijednosti, centar definira kao medoid (točku klastera koja ima najmanju ukupnu udaljenost od preostalih točaka u tom klasteru). Ova modifikacija povećava otpornost na stršeće vrijednosti u skupu podataka.
- K-median - umjesto srednje vrijednosti, definira centar kao median vrijednosti koponenti točaka klastera. Uobičajeno koristimo L_1 normu umjesto L_2 norme koju koristimo kod K-means algoritma.
- K-mod - koristimo ga za klasteriranje podataka koji ne sadrže numeričke atribute. Centroid određujemo kao mod komponenti točaka klastera.
- Fuzzy (Neizraziti) K-means - omogućava da točke istovremeno pripadaju nekolicini klastera. Algoritam dodjeljuje numeričku vrijednost (iz [0,1]) pripadanosti točke klasteru.
- X-means - varijanta algoritma koja može odrediti najbolji K. To radi određivanjem skupova centroida koji se mogu dijeliti. Za to koristi Akaike ili Bayesov informacijski kriterij.

Varijante K-means algoritma

- Inteligentan k-means - bazira se na kriteriju koji daje povećani interes točkama koje se nalaze daleko od centroida. Poronalazi klastere koji se zovu klasteri anomalijskih uzoraka.
- Bisekcijski K-means - hijerarhijska varijanta K-means algoritma koja dijeli klaster roditelj da bi stvorila dva klastera dijece.
- Kernel K-means - stvara klastere nakon projeciranja u višedimenzionalan prostor.
- Težinski K-means - uvodi težine značajnosti atributa u postupak klasteriranja.
- Genetski K-means - koristi se genetski algoritam u kombinaciji s korakom standardnog K-means algoritma da se izbjegne konvergencija u lokalni minimum.