1 Clustering

Clustering (zhlukovanie) je základná metóda analýzy dát, ktorá slúži na vytváranie podmnožín v dátach tzv. zhlukov (angl. clusters) podľa ich vzájomnej podobnosti. Cieľom clusteringu je usporiadať množinu objektov tak, aby objekty v rámci jedného zhluku boli si navzájom čo najviac podobné, zatiaľ čo objekty z rôznych zhlukov boli od seba výrazne odlišné. Podobnosť sa spravidla vyjadruje prostredníctvom vzdialenosti alebo inej metriky podobnosti, ktorá kvantifikuje, do akej miery sú si dva objekty „blízke“ alebo „vzdialené“.

Zhlukovanie patrí medzi metódy neštruktúrovaného (neusmerneného) učenia (unsupervised learning) v rámci strojového učenia a štatistiky. To znamená, že pri zhlukovaní nemáme vopred určené kategórie alebo triedy, ako je to napríklad pri klasifikácii. Úlohou algoritmu je na základe dát objaviť ich vnútornú štruktúru – teda prirodzené skupiny (podmnožiny), ktoré sa v dátach nachádzajú. Clustering nám tak umožňuje objavovať skryté vzory, vzťahy a štruktúry v údajoch, ktoré by inak zostali neviditeľné.

V praktickom kontexte možno clustering chápať ako nástroj objavovania znalostí. V sociálnych vedách či vzdelávacom výskume sa môže použiť napríklad na identifikáciu skupín žiakov s podobnými vzorcami správania, stratégiami učenia alebo profilmi výkonnosti. V oblasti spracovania prirodzeného jazyka (NLP) sa zhlukovanie využíva na zoskupovanie podobných textov alebo pojmov podľa významu. V technických aplikáciách, ako je spracovanie obrazu, clustering pomáha rozpoznať oblasti s podobnými vizuálnymi vlastnosťami.

V teoretickej rovine predstavuje clustering aj zaujímavý problém optimalizácie – algoritmus sa snaží nájsť také rozdelenie dát, ktoré minimalizuje vnútrozhlukovú vzdialenosť (t. j. rozdiely medzi objektmi v rámci jedného zhluku) a zároveň maximalizuje medzizhlukovú vzdialenosť (rozdiely medzi jednotlivými zhlukmi). Tento princíp sa dá matematicky vyjadriť a rôzne algoritmy ho implementujú odlišnými spôsobmi.

Medzi najznámejšie prístupy patria:

- **K-means algoritmus**, ktorý rozdeľuje dáta na vopred určený počet k zhlukov na základe ich vzdialenosti od tzv. centroidov.
- **Hierarchický clustering**, ktorý vytvára postupnú hierarchiu zhlukov znázornenú pomocou dendrogramu.
- **Hustotné metódy** (napr. DBSCAN), ktoré vyhľadávajú oblasti s vysokou hustotou bodov a oddeľujú ich od riedkych oblastí.
- **Pravdepodobnostné modely** (napr. Gaussian Mixture Models), ktoré predpokladajú, že dáta pochádzajú z kombinácie viacerých pravdepodobnostných rozdelení.

Clustering tak tvorí dôležitý most medzi teóriou štatistického modelovania a praktickým objavovaním vzorov a súvislostí v dátach.

1.1 Využitie clusteringu pri odhaľovaní phishingu

Clustering, ako metóda neštruktúrovaného učenia, nachádza uplatnenie aj v oblasti kybernetickej bezpečnosti, najmä pri odhaľovaní phishingových útokov. Phishing je typ sociálneho inžinierstva, pri ktorom útočník používa falošné e-maily, webové stránky alebo iné komunikačné kanály s cieľom získať citlivé informácie (napr. prihlasovacie údaje, finančné dáta).

Pri monitorovaní systémov sa generujú rozsiahle logy – záznamy o aktivitách používateľov, prístupoch do systému, emailovej komunikácii či sieťovom prevádzkovom správaní. Tieto dáta sú často veľmi heterogénne a nemajú explicitne označené záznamy phishingových útokov. Práve tu sa uplatňuje clustering:

- **Objavovanie abnormalít**
  Clustering umožňuje zoskupovať podobné logy, pričom záznamy, ktoré sa výrazne odlišujú od väčšiny, môžu indikovať podozrivé alebo škodlivé správanie. Napríklad e-maily s podobnou štruktúrou, obsahom alebo odosielateľom môžu tvoriť prirodzený zhluk, zatiaľ čo phishingové správy sa často odlišujú svojou neobvyklou syntaxou, doménou odosielateľa alebo časovou aktivitou.

- **Redukcia priestoru pre analýzu**
  Logy často obsahujú milióny záznamov. Clustering umožňuje zoskupiť veľké množiny dát do menšieho počtu reprezentatívnych skupín, čím analytikom alebo automatizovaným systémom uľahčuje identifikáciu potenciálne škodlivých aktivít bez potreby prehľadávať každý záznam jednotlivo.

- **Identifikácia vzorov útokov**
  Použitím algoritmov ako DBSCAN alebo K-means je možné identifikovať vzory podobného správania, ktoré sú typické pre phishing. Napríklad viaceré pokusy o prihlásenie z neznámych IP adries v krátkom čase alebo e-maily s rovnakou šablónou môžu byť zhlukované do jedného „podozrivého“ zhluku.

- **Integrácia s ďalšími metodikami**
  Clustering sa často používa ako predspracovanie alebo detekčný filter pred klasifikátormi založenými na dohľadanom učení (*supervised learning*). Zhlukovanie môže napríklad označiť potenciálne phishingové záznamy, ktoré následne prechádzajú detailnou analýzou alebo modelom klasifikácie, čo zvyšuje presnosť a efektivitu celého systému.

Celkovo clustering umožňuje efektívne pracovať s veľkými, heterogénnymi logmi a objavovať skryté anomálie, čo je kľúčové pri odhaľovaní phishingových útokov, ktoré často menia svoje vzory, aby sa vyhli tradičným pravidlovým alebo podpisovým detekčným systémom.

Medzi najefektívnejšie metódy clusteringu pre detekciu phishingových útokov na základe logov a sieťovej aktivity patria metódy založené na hustote (Density-based clustering), ako sú DBSCAN (Density-Based Spatial Clustering of Applications with Noise) alebo jeho vylepšená verzia HDBSCAN. Tento prístup je vhodný z niekoľkých dôvodov:

- **Detekcia anomálií a odľahlých bodov:** Phishingové útoky sa zvyčajne vyskytujú len sporadicky a v malých množstvách v porovnaní s bežnou prevádzkou používateľov. Hustotné metódy prirodzene označujú body, ktoré nespadnú do žiadneho hustého zhluku, ako „šum“ alebo anomáliu. Takto sa phishingové aktivity, ktoré sú atypické, automaticky zvýraznia bez potreby predchádzajúcej klasifikácie.

- **Nezávislosť od predpokladu tvaru zhluku:** Algoritmy ako K-means predpokladajú, že zhluky majú guľovitý alebo eliptický tvar, čo nemusí zodpovedať skutočným anomáliám. DBSCAN a HDBSCAN však dokážu odhaliť zhluky ľubovoľného tvaru, čo je dôležité pri phishingu, kde sa podozrivé správanie môže prejavovať v rôznych kombináciách atribútov (napr. IP adresa, čas odoslania, obsah správy).

- **Nevyžaduje vopred určený počet zhlukov:** Pri detekcii phishingu nevieme dopredu určiť, koľko typov útokov alebo podozrivých aktivít sa v logoch nachádza. Hustotné algoritmy identifikujú zhluky automaticky na základe lokálnej hustoty bodov, čím odstraňujú potrebu špecifikovať počet zhlukov k, čo je veľká výhoda oproti K-means alebo GMM.

- **Odolnosť voči šumu a variabilite dát:** Logy systémov a e-mailové dáta sú často neúplné, heterogénne a šumové. Hustotné metódy dokážu tieto vlastnosti zvládnuť, pretože sa zameriavajú na koncentrácie bodov, pričom riedke alebo nezvyčajné správanie sa prirodzene označí ako odchýlka.
- **HDBSCAN – hierarchická robustnosť:** HDBSCAN rozširuje DBSCAN o hierarchickú analýzu hustoty, čím umožňuje identifikovať zhluky s rôznou hustotou. To je obzvlášť užitočné pri phishingu, kde niektoré podozrivé správanie sa vyskytuje často (napr. masové spamové kampane), zatiaľ čo iné len sporadicky (cielené útoky).

Z uvedeného vyplýva, že klastrovanie založené na hustote je pre odhaľovanie phishingu ideálne, pretože dokáže efektívne rozlíšiť bežné správanie od anomálií, nevyžaduje presne definovaný počet zhlukov, dokáže pracovať s nepravidelnými tvarmi zhlukov a odoláva šumu a variabilite dát, čo sú všetko kľúčové vlastnosti logov obsahujúcich phishingové aktivity.

2 Metóda DBSCAN – zhlukovanie založené na hustote

2.1 Úvod

Metóda **DBSCAN** (*Density-Based Spatial Clustering of Applications with Noise*), ktorú navrhli Ester, Kriegel, Sander a Xu (1996), patrí medzi najvplyvnejšie algoritmy zhlukovania založené na hustote. Jej hlavnou prednosťou je schopnosť identifikovať zhluky (klastre) ľubovoľného tvaru a zároveň rozpoznať šum (*noise*) – objekty, ktoré nepatria do žiadneho zhluku. Na rozdiel od metód ako *k-means* nevyžaduje predbežné určenie počtu zhlukov.

2.2 Základná idea

Myšlienka *zhlukovania založeného na hustote* (angl. *density-based clustering*) spočíva v tom, že pre každý objekt patriaci do zhluku musí v jeho susedstve existovať aspoň minimálny počet ďalších objektov. Početnosť susedstva teda musí dosiahnuť určitú prahovú hodnotu, označovanú ako **MinPts**.

Týmto spôsobom môžu objekty nachádzajúce sa v hustejších oblastiach viacrozmerného priestoru vytvárať zhluky, ktoré sú oddelené oblasťami s menšou hustotou (tzv. šumom). Hustota je teda určovaná pomocou dvoch parametrov:

- ε – **polomer susedstva (radius)**,ktorý určuje vzdialenosť, v rámci ktorej sa objekty považujú za susedné,
- **MinPts** – **minimálny počet bodov** v rámci zhluku (ε-susedstva), aby bola oblasť považovaná za dostatočne hustú.

2.3 Formálne definície

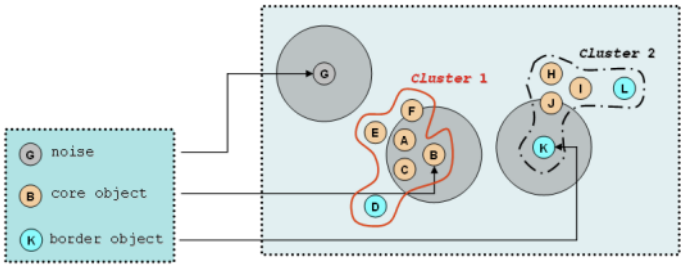
Nech $O = \{o_1, o_2, \ldots, o_n\} \subseteq \mathbb{R}^d$ je množina objektov v d-rozmernom priestore a $dist(o_i, o_j)$ je metrika vzdialenosti (najčastejšie Euklidovská vzdialenosť).

Definícia 2.1: ε-susedstvo

Pre daný bod $o \in O$ a parameter $\varepsilon > 0$ definujeme ε-**susedstvo** bodu o ako množinu:

$$N_\varepsilon(o) = \{o' \in O \mid dist(o, o') \leq \varepsilon\}.$$

To znamená, že ε-okolie objektu $o \in O$ zahŕňa všetky objekty $o' \in O$, ktoré sa nachádzajú vo vnútri polomeru ε okolo bodu o; vzdialenosť $dist(o, o')$ medzi bodmi o a o' nesmie byť väčšia ako ε.



Obr. 1: Density-Based clustering

Obrázok 1 ukazuje, že na konci procesu vzniknú tri triedy objektov:

- **trieda šumu (noise)** – objekty, ktoré nie je možné priradiť k žiadnemu zhluku,
- **trieda jadrových objektov (core objects)**,
- **trieda okrajových objektov (border objects)** – patria do zhluku, ale samy nie sú jadrovými objektmi.

Definícia 2.2: Jadrový bod (*core point*)

Bod $o \in O$ je **jadrový bod**, ak počet bodov v jeho ε-susedstve spĺňa podmienku:

$$|N_\varepsilon(o)| \geq \text{MinPts}.$$

V algoritme DBSCAN zhluk nevzniká z jediného jadrového objektu, ale z viacerých jadrových objektov, ktoré sú navzájom „hustotne dosiahnuteľné“ (*density-reachable*). Jeden jadrový bod síce často „spustí“ proces vytvárania zhluku (pri prechádzaní dát), ale výsledný zhluk sa skladá z viacerých jadrových objektov a prípadne aj z okrajových objektov, ktoré ležia v ich ε-okruhu.

Definícia 2.3: Okrajový objekt

Objekt $o \in O$ sa nazýva *okrajový objekt*, ak platí:

1. $|N_\varepsilon(o)| < \text{MinPts}$ (podmienka jadrového objektu nie je splnená),
2. $\exists\, o' \in \{\text{jadrové objekty}\}$, pre ktoré platí $o \in N_\varepsilon(o')$.

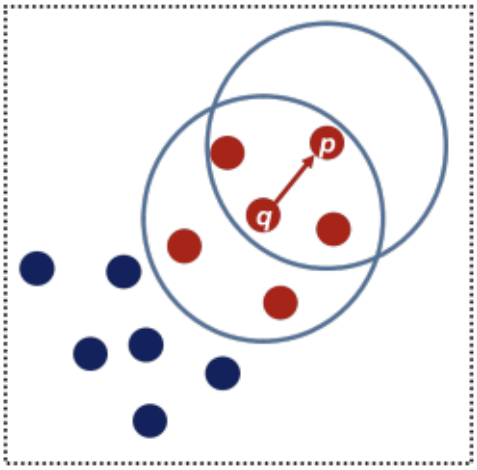
Objekt je teda *okrajovým objektom* vtedy, ak sa v jeho ε-okruhu (susedstve) nachádza menej ako MinPts objektov. Okrajový objekt patrí do zhluku, pretože sa nachádza v ε-okruhu (susedstve) jadrového objektu, sám však nemôže vytvoriť zhluk (porovnaj Obrázok 1).

Všetky objekty v oblasti zhluku musia byť vzhľadom na parametre MinPts a ε priamo hustotne dosiahnuteľné.

Definícia 2.4: Priama hustotná dosiahnuteľnosť (*directly density-reachable*)

Bod p je **priamo hustotne dosiahnuteľný** z bodu q, ak platí:

$$p \in N_\varepsilon(q) \quad \text{a zároveň} \quad |N_\varepsilon(q)| \geq \text{MinPts}.$$



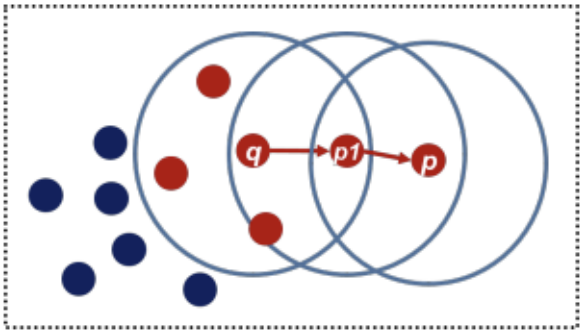
Obr. 2: Priama hustotná dosiahnuteľnosť

Definícia 2.5: Hustotná dosiahnuteľnosť (*density-reachable*)

Bod p je **hustotne dosiahnuteľný** z bodu q, ak existuje konečná postupnosť bodov

$$q = o_1, o_2, \ldots, o_k = p,$$

taká, že každý bod o_{i+1} je *priamo hustotne dosiahnuteľný* z bodu o_i.



Obr. 3: Hustotná dosiahnuteľnosť

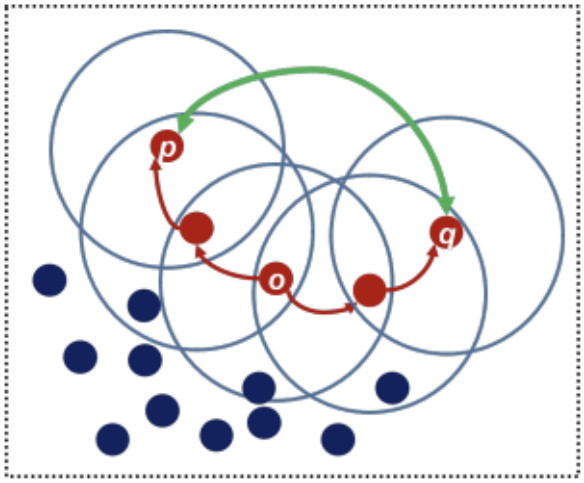
Hustotná dosiahnuteľnosť je rozšírením priamej hustotnej dosiahnuteľnosti. Táto relácia je tranzitívna, ale nie je symetrická (porovnaj Obrázok 3). Symetria spravidla neplatí pre všetky objekty v rámci zhluku, môže však platiť pre určitý objekt o, pre ktorý platí $|N_\varepsilon(o)| \geq \text{MinPts}$.

Dva okrajové objekty jedného zhluku nemusia byť navzájom hustotne dosiahnuteľné, pretože v ich ε-oblasti sa nenachádza dostatočný počet objektov. Je preto potrebný iný jadrový objekt, z ktorého sú oba okrajové body hustotne dosiahnuteľné. Týmto spôsobom sa zavádza pojem *hustotného prepojenia*.

Definícia 2.6: Hustotné prepojenie (*density-connected*)

Dva body $p, q \in O$ sú **hustotne prepojené**, ak existuje bod $o \in O$, z ktorého sú oba body hustotne dosiahnuteľné:

$$\exists o \in O : \ p \text{ je hustotne dosiahnuteľný z } o \ \wedge \ q \text{ je hustotne dosiahnuteľný z } o.$$



Obr. 4: Hustotné prepojenie

Hustotné prepojenie je symetrická relácia. Objekty p a q sú teda cez objekt o navzájom hustotne prepojené. Objekt o pritom môže byť aj jedným z objektov p alebo q (porovnaj Obrázok 4).

Zhluky založené na hustote je možné všeobecne chápať ako *hustotne prepojené množiny*. Hustotne prepojené množiny sa definujú rovnakým spôsobom ako zhluky založené na hustote.

Definícia 2.7: Zhluk založený na hustote (*density-based cluster*)

Nech O je množina objektov. Podmnožina $C \subseteq O$ je **zhluk** vzhľadom na parametre ε a MinPts, ak C nie je prázdna množina a platia nasledujúce podmienky:

- **Maximalita:** $\forall p, q \in O$: ak $p \in C$ a q je hustotne dosiahnuteľný z p vzhľadom na ε a MinPts, potom $q \in C$.
- **Spojitosť:** $\forall p, q \in C$: body p a q sú hustotne prepojené v O vzhľadom na ε a MinPts.

Definícia 2.8: Šum (*noise*)

Body, ktoré nepatria do žiadneho zhluku, sú označené ako **šum**:

$$\text{Noise} = \{o \in O \mid \forall C_i : o \notin C_i\}.$$

2.4 Interpretácia hustoty

Zhlukovanie podľa hustoty možno interpretovať aj cez **lokálnu hustotu bodov**. Pre bod o definujeme lokálnu hustotu ako:

$$\rho(o) = \frac{|N_\varepsilon(o)|}{V_\varepsilon},$$

kde V_ε je objem ε-gule v d-rozmernom priestore:

$$V_\varepsilon = \frac{\pi^{d/2}}{\Gamma\left(\frac{d}{2}+1\right)}\varepsilon^d.$$

DBSCAN však túto hustotu nepočíta explicitne, ale ju odhaduje implicitne porovnávaním počtu bodov s prahom *MinPts*.

2.5 Algoritmus DBSCAN

Vstup:

- množina bodov $O \subseteq \mathbb{R}^d$,
- parametre ε, MinPts.

Výstup:

- množina zhlukov $C_1, C_2, \ldots, C_k$,
- množina bodov šumu N.

Postup algoritmu:

1. Pre každý bod $o \in O$, ktorý ešte nebol navštívený:
   (a) Označ o ako navštívený a vypočítaj $N_\varepsilon(o)$.
   (b) Ak $|N_\varepsilon(o)| <$ MinPts, označ bod ako šum.
   (c) Inak začni nový zhluk a rozširuj ho o všetky body, ktoré sú dosiahnuteľné hustotou z o.
2. Proces opakuj, kým všetky body nie sú označené ako:
   - súčasť zhluku, alebo
   - šum.

2.6 Vlastnosti algoritmu

- Identifikuje zhluky ľubovoľného tvaru.
- Dokáže odlíšiť šum a okrajové body.
- Nevyžaduje predbežné určenie počtu zhlukov.
- Je citlivý na voľbu parametrov ε a MinPts.

2.7 Voľba parametrov

- **MinPts:** odporúča sa MinPts $\geq d + 1$, kde d je dimenzia priestoru.
- ε**:** určuje sa na základe *k-distance grafu*, v ktorom sa hľadá bod prudkého zlomu („koleno“).

2.8 Záver

Metóda DBSCAN poskytuje robustný a teoreticky dobre zdôvodnený prístup k identifikácii zhlukov na základe lokálnej hustoty bodov. Jej matematické jadro spočíva v konceptoch ε-susedstva a hustotnej konektivity, ktoré umožňujú modelovať štruktúru dát bez predpokladu tvaru alebo počtu zhlukov.

2.9 HDBSCAN (Hierarchical Density-Based Spatial Clustering of Applications with Noise)

Algoritmus **HDBSCAN** predstavuje rozšírenie metódy DBSCAN, ktoré odstraňuje jej hlavné obmedzenia, najmä potrebu vopred určiť parameter ε. HDBSCAN umožňuje odhaliť **zhluky s rôznou hustotou** a vytvoriť ich **hierarchickú štruktúru**, pričom automaticky extrahuje stabilné a významné zhluky bez nutnosti manuálneho výberu prahu.

2.9.1 Základná idea

Zatiaľ čo DBSCAN definuje zhluky ako oblasti s hustotou bodov vyššou než daný prah ε, HDBSCAN modeluje hustotu kontinuálne a vytvára *hierarchiu hustotných úrovní*, z ktorej následne extrahuje stabilné zhluky. Tým sa vyhýba problému voľby jediného vhodného ε pre dáta s nehomogénnou hustotou.

2.9.2 Matematické základy

Nech $X = \{x_1, x_2, \ldots, x_n\} \subset \mathbb{R}^d$ je množina dátových bodov a $d(x_i, x_j)$ metrika vzdialenosti.

1. **Core-distance:** Pre bod x_i definujeme jeho *jadrovú vzdialenosť* ako vzdialenosť k k-temu najbližšiemu susedovi:

   $$\text{core_dist}(x_i) = d_k(x_i),$$

   kde $k = \text{MinPts}$ je minimálny počet bodov v susedstve (parameter algoritmu).

2. **Mutual reachability distance:** Pre dvojicu bodov x_i, x_j definujeme *vzájomnú dosiahnuteľnosť* ako:

   $$d_{\text{mreach}}(x_i, x_j) = \max\{\text{core_dist}(x_i),\ \text{core_dist}(x_j),\ d(x_i, x_j)\}.$$

   Táto metrika vyhladzuje rozdiely v lokálnej hustote a zabezpečuje, že body z riedkych oblastí nebudú umelo spájané s hustými regiónmi.

3. **Transformácia do hustotnej škály:** Často sa používa parameter

   $$\lambda = \frac{1}{d}$$

   (pre klasterovacie hranice), teda malé vzdialenosti zodpovedajú veľkým hodnotám λ (vysoká „hustota“). Pri hierarchickom zápise sa pracuje s poriadkom podľa rastúcej vzdialenosti d (resp. klesajúcej λ).

2.9.3 Algoritmus HDBSCAN

1. **Spočítame core-distance:** Pre každý bod vypočítame *core-distance* (k-NN vzdialenosť pri parametri `MinPts`).
2. **Definujeme mutual reachability distance:** Pre všetky páry bodov definujeme *mutual reachability distance*:
$$d_{\text{mreach}}(x_i, x_j) = \max\{\text{core_dist}(x_i), \text{core_dist}(x_j), d(x_i, x_j)\}.$$
3. **Postavíme vážený graf:** Vytvoríme graf s vrcholmi X a hranami váženými d_{mreach}. (V praxi sa často používa k-NN graf alebo efektívne aproximácie, aby sa nebudoval úplný graf.)
4. **Vypočítame minimálne kostrové MST:** Z grafu vypočítame *minimálne kostrové (MST) strom* podľa váh d_{mreach}. MST zachytáva spojenia podľa rastúcej „dostupnosti“ medzi bodmi pri zvyšujúcich sa vzdialenostiach.
5. **Aplikujeme single-linkage hierarchické zhlukovanie:** Nad MST aplikujeme *single-linkage hierarchical clustering* podľa váh d_{mreach}. Výsledkom je hierarchia zhlukov (dendrogram) v mierke hustoty.
6. **Kondenzujeme hierarchiu:** Kondenzujeme strom a zameriame sa na komponenty, ktoré majú veľkosť $\geq$ `min_cluster_size`. To eliminuje veľmi malé, nevýznamné vetvy.
7. **Vyhodnotíme stabilitu zhlukov:** Pre každý kandidátny zhluk v kondenzovanej hierarchii vypočítame jeho *stabilitu* a extrahujeme tie zhluky, ktoré sú najstabilnejšie. Tieto zhluky tvoria konečné, „ploché“ rozdelenie.
8. **Označíme šum:** Body, ktoré nepatria do žiadneho stabilného zhluku, sú označené ako *noise* (šum).

2.9.4 Miera stability zhluku

Každý zhluk C sa v hierarchii *narodí* pri určitej hustote λ_{birth} a *zanikne* pri hustote λ_{death}. Stabilita zhluku je definovaná ako integrál alebo súčet príspevkov všetkých bodov, ktoré zhluk opustia:
$$\text{stability}(C) = \sum_{p \in C} (\lambda_p - \lambda_{\text{birth}}),$$
kde λ_p je hustota (resp. $1/d$), pri ktorej bod p prestane patriť do zhluku. Zhluky s vysokou stabilitou pretrvávajú cez široké rozmedzie hustôt a sú preto považované za *robustné a významné*.

Výhody HDBSCAN oproti DBSCAN a iným metódam

- **Prispôsobivosť lokálnej hustote:** dokáže nájsť zhluky rôznej hustoty v jednom behu (DBSCAN vyžaduje rôzne ε pre rôzne hustoty).
- **Hierarchický výstup:** poskytuje informáciu o vzťahoch medzi zhlukmi na rôznych úrovniach hustoty, čo uľahčuje analýzu a vizualizáciu.
- **Robustná extrakcia zhlukov cez mieru stability:** menej závisí od jediného vysoko citlivého prahu.
- **Lepšie zaobchádzanie so šumom:** body, ktoré nie sú súčasťou žiadneho stabilného zhluku, sú označené ako *outliers* (šum).

Parametre HDBSCAN a ich interpretácia

- **min_cluster_size:** minimálna veľkosť zhluku, ktorý považujeme za relevantný. Určuje granulositu kondenzovaného stromu (typicky volené podľa doménového významu, napr. očakávaný minimálny počet bodov, ktoré považujeme za „skutočný“ zhluk).
- **min_samples** (alebo `MinPts`): používa sa pri výpočte *core-distance*. Vyššia hodnota znamená prísnejšie posudzovanie jadrových bodov, čo robí algoritmus robustnejším voči šumu, ale môže zlúčiť alebo odstrániť malé zhluky.
- **Odporúčania:** často sa volí `min_samples = min_cluster_size` alebo `min_samples = min_cluster_size / 2`, prípadne `min_samples = dimenzionalita datasetu`.
- **Citlivosť:** HDBSCAN je menej citlivý na presné nastavenie `min_cluster_size` než DBSCAN na parameter ε, ale parametre stále ovplyvňujú výsledok zhlukovania.

2.9.5 Použitie a aplikácie

HDBSCAN je vhodný najmä pre úlohy, kde sa v dátach vyskytujú oblasti s rôznou hustotou a tvarom zhlukov – napríklad pri analýze logov, detekcii anomálií či identifikácii phishingových útokov. Jeho schopnosť prispôsobiť sa lokálnej štruktúre dát a automaticky oddeľovať šum z neho robí robustný nástroj na odhaľovanie komplexných vzorcov správania v reálnych dátach.

Praktické odporúčania (najmä pri analýze logov / detekcii phishingu)

- **Škálovanie a výber atribútov:** pred výpočtom metriky normalizujeme alebo škálujeme atribúty (napr. log-transform pre časové intervaly, one-hot pre kategórie alebo vektorové embeddingy pre text e-mailu). Dbáme na správne škálovanie, aby rôzne dimenzie mali primeranú dôležitosť a nezdeformovali core-distance.
- **Výber metriky:** pri textových alebo kombinovaných dátach zvolíme nedej-Euklidovské metriky (cosine pre TF-IDF embeddingy, edit distance pre domény, atď.) alebo skonštruujeme zloženú vzdialenosť.
- **Nastavenie min_cluster_size:** typicky medzi 5–50; pri detekcii phishingu volíme menšiu hodnotu, ak očakávame malé cielené kampane, ale nie príliš malú, aby sa nevybral veľký počet „šumu“ ako zhluky.
- **Nastavenie min_samples:** zvýšime hodnotu pre konzervatívnejšiu extrakciu zhlukov; pri veľkom množstve šumu mierne zvýšime hodnotu.
- **Kombinácia s ďalšími metódami:** po HDBSCAN môžeme aplikovať supervised klasifikátor alebo vykonať ručnú analýzu kandidátnych zhlukov (napr. extrahovanie typických domén alebo šablón správ v zhluku).
- **Interpretovateľnosť:** HDBSCAN prirodzene vytvára hierarchiu, ktorú vizualizujeme (condensed tree, cluster persistence diagram) — veľmi užitočné pri forenznej analýze phishingových útokov.