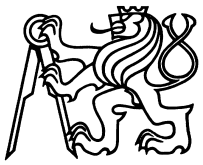
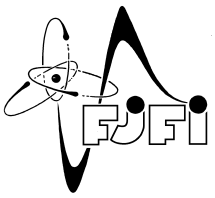


České vysoké učení technické v Praze
Fakulta jaderná a fyzikálně inženýrská



Detekce překrývajících se komunit v bipartitních grafech

Overlapping Community Detection in Bipartite Graphs

Diplomová práce

Autor: **Bc. Tomáš Zikmund**

Vedoucí práce: **Ing. Radek Mařík, CSc.**

Akademický rok: 2019/2020

Katedra: matematiky Akademický rok: 2018/2019

ZADÁNÍ DIPLOMOVÉ PRÁCE

Student:	Bc. Tomáš Zikmund
Studijní program:	Aplikace přírodních věd
Obor:	Matematická informatika
Název práce (česky):	Detekce překrývajících se komunit v bipartitních grafech
Název práce (anglicky):	Overlapping Community Detection in Bipartite Graphs

Pokyny pro vypracování:

1. Vytvořte přehled modelů a příslušných metod pro detekci komunit. Zaměřte se na modely schopné popsat nepřekrývající a překrývající se komunity a komunity v bipartitních grafech.
2. Vyberte model komunit v grafu, popisující překrývající se komunity v bipartitním grafu. Navrhněte, odvoďte podmínky použití a vlastnosti detekční metody.
3. Implementujte algoritmus hledající parametry modelu pro komunity v bipartitním grafu.
4. Ověřte výkonnost a efektivitu detekční metody. Zvolte vhodná syntetická i reálná data.
5. Proveďte diskusi získaných výsledků.

Doporučená literatura:

1. M. E. J. Newman, Networks. Oxford university press, 2018.
2. B. Karrer, M. E. J. Newman, Stochastic blockmodels and community structure in networks. Physical review E 83, 2011, 016107.
3. B. Ball, B. Karrer, M. E. J. Newman, An efficient and principled method for detecting communities in networks. Physical Review E 84, 2011, 036103.
4. J. Yang, J. Leskovec, Overlapping community detection at scale: a nonnegative matrix factorization approach. In 'WSDM '13 Proceedings of the sixth ACM international conference on Web search and data mining', ACM New York, 2013, 587-596.
5. D. B. Larremore, A. Clauset, A. Z. Jacobs, Efficiently inferring community structure in bipartite networks. Physical review E 90, 2014, 012805.
6. R. Marik, T. Zikmund, Overlapping Communities in Bipartite Graphs. Accepted in 'Complex Networks & Their Applications VII - Volume 1 Proceedings The 7th International Conference on Complex Networks and Their Applications COMPLEX NETWORKS 2018', Springer Nature Switzerland AG, 2018.

Jméno a pracoviště vedoucí diplomové práce:

Ing. Radek Mařík, CSc.
Katedra telekomunikační techniky, FEL, ČVUT v Praze, Technická 2, 160 00 Praha

Jméno a pracoviště konzultanta:

Datum zadání diplomové práce: 31.10.2018

Datum odevzdání diplomové práce: 6.5.2019

Doba platnosti zadání je dva roky od data zadání.

V Praze dne 24. října 2019

.....
garant oboru

.....
vedoucí katedry

.....
děkan

Poděkování:
Chtěl bych zde poděkovat především svému školiteli Radkovi Maříkovi za pečlivost, ochotu, vstřícnost a odborné i lidské zázemí při vedení mojí diplomové práce.

Čestné prohlášení:
Prohlašuji, že jsem tuto práci vypracoval samostatně a uvedl jsem všechnu použitou literaturu.

V Praze dne 3. ledna 2019 Bc. Tomáš Zikmund

Název práce:

Detekce překrývajících se komunit v bipartitních grafech

Autor: Bc. Tomáš Zikmund

Obor: Matematická informatika

Druh práce: Diplomová práce

Vedoucí práce: Ing. Radek Mařík, CSc.Katedra telekomunikační techniky, Fakulta elektrotechnická, ČVUT v Praze,

Abstrakt: Práce se zabývá modelováním komunit v grafech pomocí stochastických blokových modelů s Poissonovým rozdělením a následně detekcí komunit optimalizací těchto modelů příslušnými metodami. Speciálně se zaměřuje na detekci překrývajících se komunit v bipartitním grafu. Dále popisuje implementaci a vlastnosti metody pro detekci překrývajících se komunit v bipartitních grafech. V závěru práce je také obsaženo srovnání rozebíraných metod, jak z hlediska optimalizace účelové funkce, tak z hlediska úspěšnosti odhalení připravených komunit v náhodně generovaných grafech.

Klíčová slova: detekce komunit, stochastické blokové modely, překrývající se komunity, komunity v bipartitních grafech, unipartitní grafy, bipartitní grafy,

Obsah

Značení

Značka	Popis
$\boldsymbol{A}$	matice
$\boldsymbol{0}$	nulová matice
$\boldsymbol{I}$	matice identity
$\mathbb{R}^{m,n}$	prostor reálných matic rozměru $m \times n$
$\boldsymbol{A}_{ij}$	ij-tý prvek matice
$\boldsymbol{A}_{i\bullet}$	i-tý řádek matice
$\boldsymbol{A}_{\bullet j}$	j-tý sloupec matice
$\vec{x}_i$	i-tý prvek vektoru
$\sigma(\boldsymbol{A})$	spektrum matice
$\rho(\boldsymbol{A})$	spektrální poloměr matice
$\vec{x}$	vektor
$\vec{0}$	nulový vektor
$\vec{x}^T, \boldsymbol{A}^T$	transpozice vektoru, matice
$\boldsymbol{A}^{-1}$	inverzní matice
$\|\vec{x}\|$	norma vektoru
$\hat{n}$	$\{m \in \mathbb{N} \mid m \leq n\}$
$\hat{n}^0$	$\hat{n} \cup \{0\}$
$[a, b]$	uzavřený interval $a, b \in \mathbb{R}$
$\mathcal{P}(X)$	potenční množina množiny X
$\mathcal{G}(V, E)$	matematický graf $\mathcal{G}$ s vrcholy v množině V a hranami E
$\mathcal{G}(\boldsymbol{A})$	matematický graf $\mathcal{G}$ s maticí sousednosti $\boldsymbol{A}$
n	počet vrcholů grafu
m	počet hran grafu
d_i	stupeň i-tého vrcholu
k	počet komunit

Úvod

V práci se zabývám návrhem a implementací detekční metody hledající překrývající se komunity v grafech. Základem detekce komunit je model schopný tyto komunity popsat. Musí být dostatečně obsáhlý na to, aby dokázal popsat i překrývající se komunity v bipartitních grafech.

Součástí práce je srovnání detekčních metod založených na stochastických blokových modelech s Poissonovým rozdělením a testování v testovacím prostředí obsahujícím generátor náhodných grafů se strukturou překrývajících se komunit. Implementován je v testovacím prostředí také nástroj, který porovnává nalezený model s modelem pro generování náhodného grafu.

Diplomovou prací navazuji na výzkumný úkol a na svoji bakalářskou práci. V bakalářské práci jsem se zabýval možnostmi a metodami pro detekci komunit v grafech. Metody popsané v bakalářské práci se snažily vysvětlit změny v hustotě hran mezi vrcholy tak, že rozdělily vrcholy do skupin, které jsou vzájemně propojeny relativně hustěji vzhledem ke zbytku grafu. Po tomto rozdělení vrcholů může být daleko snazší vysvětlit a popsat význam různých částí matematického grafu. Ve výzkumném úkolu jsem se zaměřil na modely popisující komunity, vytvořil jsem postup a nástroj pro jejich porovnání a generátor náhodných grafů obsahujících strukturu popsanou různými modely. Také jsem zkombinováním vlastností dostupných modelů odvodil požadavky, abych dokázal popsat překrývající se komunity v bipartitních grafech.

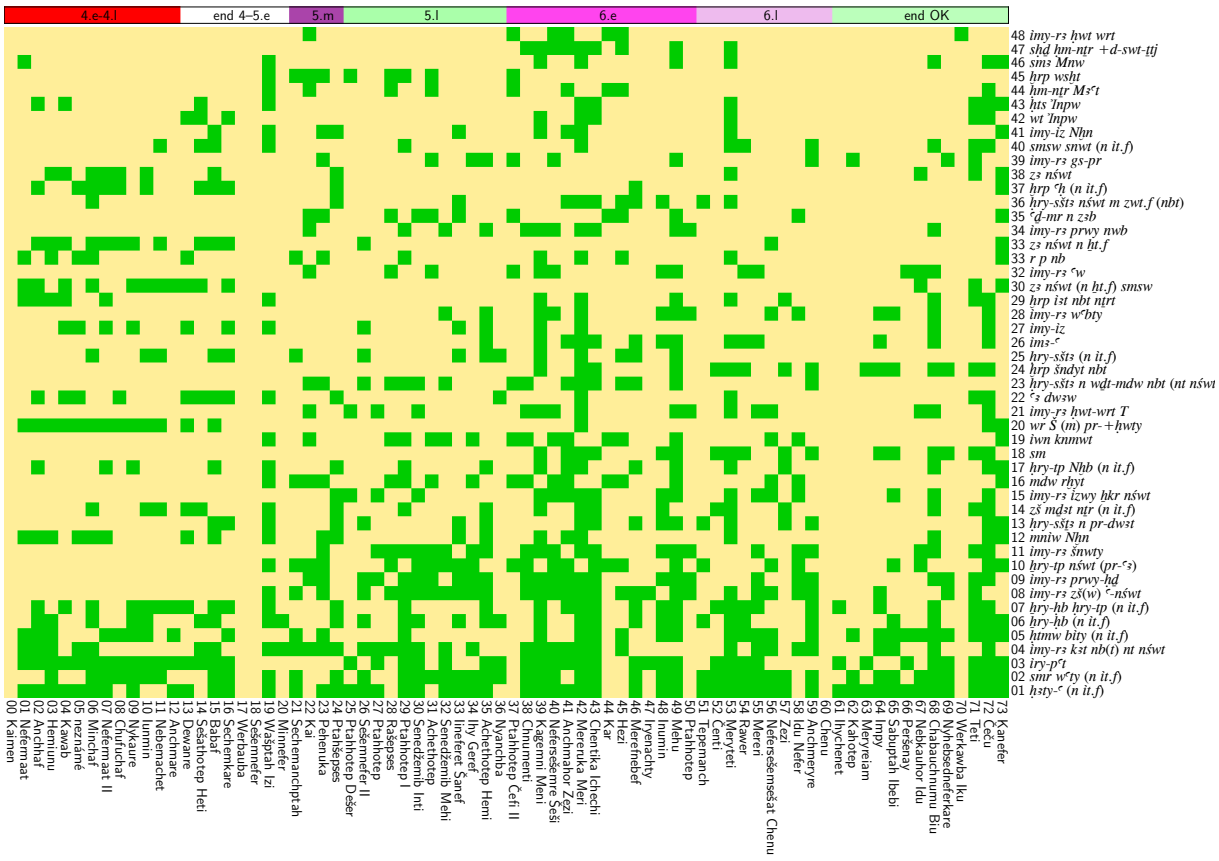
V rešeršní části jsem využil zdrojů použitých pro bakalářskou práci a výzkumný úkol, ale zaměřil jsem se na jiné teoretické aspekty. Především tedy na algoritmy detekce komunit prostřednictvím stochastických blokových modelů, jejich účelové funkce a jejich společné vlastnosti.

Testovací prostředí bylo navrženo v rámci výzkumného úkolu se záměrem umožnit vyvinout detekční metodu schopnou nalézt překrývající se komunity v bipartitním grafu. V bipartitním grafu nelze komunity chápat jako shluky hustěji propojených komunit, ale jako stejný způsob propojení hranami. Vývojem takové metody se zabývá tato práce.

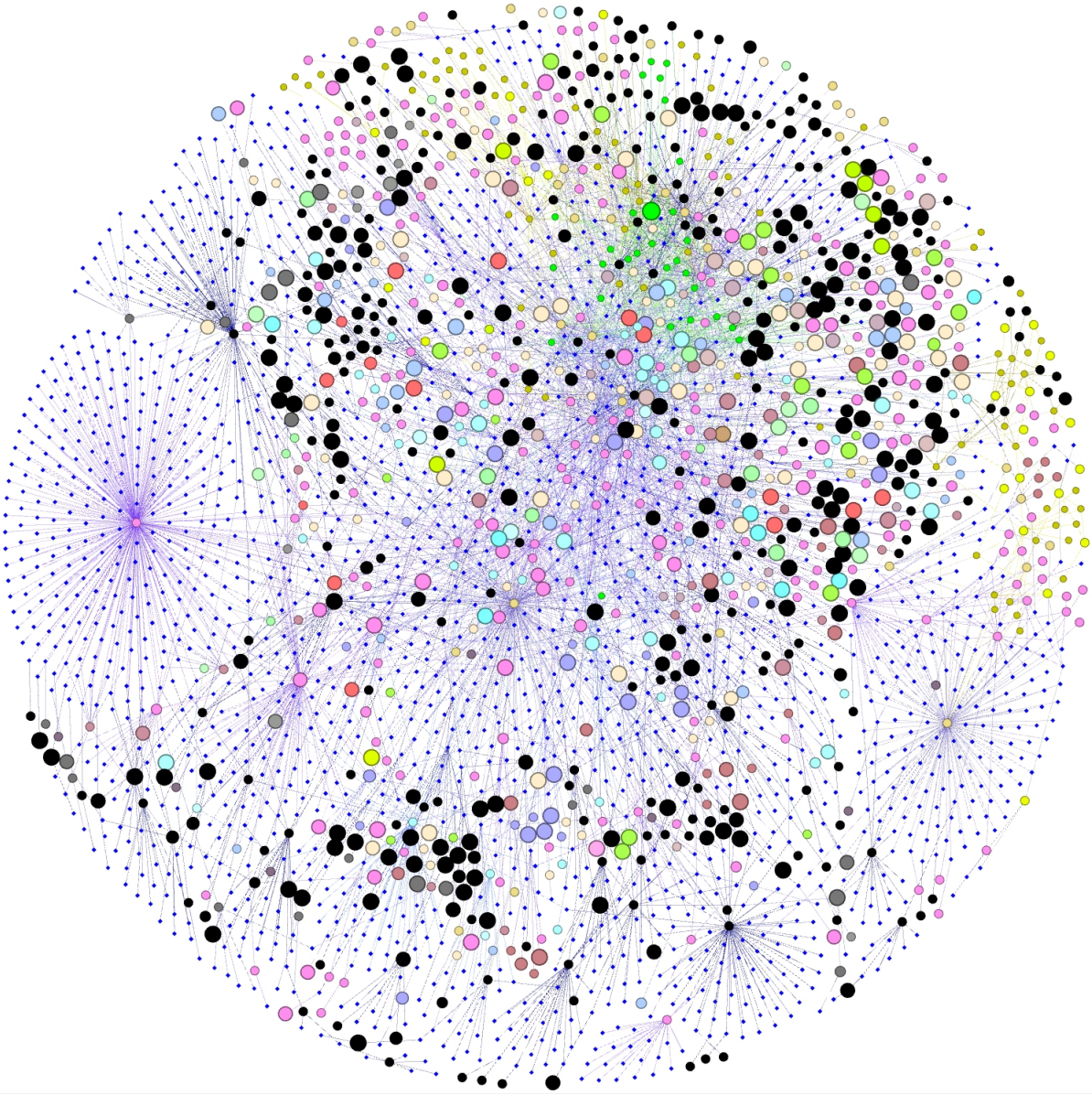
Motivací vývoje takové metody jsou data z informaticko-sociálního projektu pro GAČR číslo 16-072105: *Complex network methods applied to ancient Egypt data in the Old Kingdom (2700–2180 BC)*. Výzkumný projekt se zabýval vývojem společnosti starověkého Egypta z hlediska nepotismu (protekce a předávání strategických úřednických a společenských funkcí příbuzným) a dalších společenských procesů přetvářející království na stát [12].

Zdrojem dat jsou symboly umístěné na objevených hrobkách výše postavených lidí ve starověké egyptské společnosti. Ukázka dat je zobrazena na obrázku 1. Z těchto symbolů by mělo být možné získat pro každou osobu její indentifikaci, příbuzenské vztahy a administrativní posty (např. správa pokladnic, soudnictví, správa sýpek, péče o panovníkovo tělo, apod.).

Na obrázku 2 je vyobrazena síť titulů a jejich nositelů. Tato síť je bipartitní graf lidi–tituly. Vyobrazení vzniklo Fruchtermanovým-Reingoldovým zobrazovacím algoritmem [16]. Tento algoritmus může sloužit jako pomůcka pro detekci komunit, ale jeho výstup není dostatečný pro další zpracování. Výsledkem je takové zobrazení jako na obrázku 2, kde opticky vznikají shluky lidí kolem titulů, ale chybí jejich strojově čitelné rozřazení do komunit.



Obrázek 1: Vstupní data ze starověkého Egypta, získaná z hrobek vezírů. Obrázek zobrazuje incidentní matici bipartitního grafu. Sloupce reprezentují vezíry a řádky patří titulům. Zelené políčko ■ znamená, že vezír (sloupec) byl nositelem příslušného titulu (řádek). Poskytnuto vedoucím práce.



Obrázek 2: Vyobrazení sítě titulů a jejich nositelů. Velké barevné body jsou tituly. Malé modré kosočtverce ♦ jsou lidé (nositelé titulů) a malé tmavě-žluté kroužky ○ jsou nositelé z královské rodiny. Poskytnuto vedoucím práce.

Nejprve připravím v kapitole 1 definice a věty pro text práce. Dále se pokusím zařadit detekci komunit mezi ostatní přístupy zpracování dat, strojového učení a redukce dimenze v kapitole 2. Poté popíšu stochastické blokové modely (kap. 3), k nim příslušné řešící metody a jejich společné vlastnosti. Na základě stochastických blokových modelů jsem již ve výzkumném úkolu odvodil požadavky na model, kterým lze popsat překrývající se komunity i v bipartitních grafech. Konkrétní podoba, odvození a vlastnosti modelu použitého k detekci budou popsány v kapitole 4. V kapitolách 5 a 6 popíšu, jak navrženým modelem detekovat komunity v grafech. V závěru budou v kapitole 7 výsledky a diskuse srovnání schopností a vlastností metod detekujících komunity v grafech. Jak z hlediska optimalizace účelové funkce, tak z hlediska úspěšnosti detekce komunit.

Kapitola 1

Matematický úvod

1.1 Matematický graf

Matematickým grafem uvažujeme množinu vrcholů propojených hranami [11]. Počet vrcholů budeme značit n a počet hran jako m. Hrany, které spojují vrchol se sebou samým, budeme nazývat *smyčky* (angl. self-edge). Graf, který neobsahuje smyčky a ani mnohočetné hrany, budeme nazývat *jednoduchým grafem*.

Existuje mnoho způsobů, jak graf matematicky reprezentovat. Jedním z nich je pomocí zavedení množin V pro vrcholy a $E \subset \{\{i,j\} : i,j \in V\}$ pro hrany. Graf je také možno reprezentovat pomocí matice sousednosti (angl. adjacency matrix).

Definice 1.1.1. (matice sousednosti) [11] Pro graf $\mathcal{G}(V,E)$ definujeme matici sousednosti $\boldsymbol{A}$ jako

$$\boldsymbol{A}_{ij} = \left\{ \begin{array}{ll} 1 & \text{pokud } \{i,j\} \in E, \\ 0 & \text{jinak.} \end{array} \right\}. \tag{1.1}$$

Definice 1.1.2. (graf zadaný maticí sousednosti) [11] Buď $\boldsymbol{A}$ maticí sousednosti nějakého grafu $\mathcal{G}(V,E)$. Grafem zadaným maticí sousednosti $\mathcal{G}(\boldsymbol{A})$ rozumíme tento $\mathcal{G}(V,E)$, jež má za matici sousednosti $\boldsymbol{A}$.

Poznámka 1.1.3.

- Pro ohodnocený graf ($\exists \phi : E \to \mathbb{R} \setminus \{0\}$) je možné do matice sousednosti namísto hodnoty 1 vložit $\phi(\{i,j\})$.
- Je-li graf $\mathcal{G}$ jednoduchý a neorientovaný, potom je $\boldsymbol{A}$ symetrická.
- Neobsahuje-li $\mathcal{G}$ smyčky, potom je diagonála $\boldsymbol{A}$ nulová. Pro smyčku z vrcholu i do vrcholu i, je hodnota $\boldsymbol{A}_{ii} = 2$. (Hrana má dva konce a oba jsou v i [33].)
- Ze vztahu (1.1) plyne $\sum_{i,j \in V} \boldsymbol{A}_{ij} = 2m$, kde $m = |E|$ je počet hran.

Definice 1.1.4. (multigraf) [11] *Multigrafem* rozumíme ohodnocený graf $\mathcal{G}(V,E,\phi)$, kde $\phi : E \to \mathbb{N} \setminus \{0\}$. Hodnota $\phi(\{i,j\})$ má význam „počtu hran“ mezi vrcholy i a j.

1.1.1 Hypergraf

Uvažujme nyní hranu, která je schopna propojit více než jen dva vrcholy. Takovou hranu budeme nazývat *hyperhrana* (angl. hyper-edge) a graf, který takové hrany obsahuje nazveme *hypergrafem* [11]. Hyperhranu si můžeme jednoduše představit jako skupinu vrcholů. Hypergraf je možné reprezentovat bipartitním grafem [33].

1.1.2 Bipartitní graf

V *bipartitním grafu* jsou dva typy vrcholů, označme jejich množiny V a W. Hrany bipartitního grafu spojují vždy jen vrcholy opačného typu ($E \subset \{(v, w) : v \in V, w \in W\}$). Bipartitní graf můžeme označit symbolem $\mathcal{B}(V, W, E)$. Celý jej můžeme reprezentovat pomocí *incidentní matice* (angl. incidence matrix).

Definice 1.1.5. (incidentní matice)[1] [33] Pro graf $\mathcal{B}(V, W, E)$ definujeme incidentní matici $\boldsymbol{B}$ jako

$$\boldsymbol{B}_{ij} = \left\{ \begin{array}{ll} 1 & \text{vrchol } i \in V \text{ je spojen hranou s } j \in W, \\ 0 & \text{jinak.} \end{array} \right\}. \tag{1.2}$$

Poznámka 1.1.6. V souvislosti s hypergrafem můžeme říct, že $\boldsymbol{B}_{ij} = 1$, když vrchol $i \in V$ patří do skupiny $j \in W$.

Poznámka 1.1.7. Bipartitní graf obsahuje vrcholy dvojího typu a nemůže v něm existovat hrana, která by spojila dva vrcholy stejného typu.

1.1.3 Stupeň a sousedství

Definice 1.1.8. (Sousedství a stupeň vrcholu) [33] Buďte $G(V, E)$ graf a vrchol $i \in V$. Potom množinu

$$\mathcal{N}(i) = \{j \in V : \exists (i, j) \in E\} \tag{1.3}$$

nazveme *sousedstvím vrcholu* v a číslo $d_i = |\mathcal{N}(v)|$ nazveme *stupněm vrcholu*.

Pro matici sousednosti $\boldsymbol{A}$ a stupeň vrcholu platí

$$d_i = \sum_{j \in V} \boldsymbol{A}_{ij}. \tag{1.4}$$

Z toho plyne

$$\sum_{i \in V} d_i = \sum_{i,j \in V} \boldsymbol{A}_{ij} = 2m, \tag{1.5}$$

kde $m = |E|$ je počet hran.

1.1.4 Podgraf a cesta

Definice 1.1.9. (podgraf) [5] Jestliže pro grafy $\tilde{\mathcal{G}}(\tilde{V}, \tilde{E})$ a $\mathcal{G}(V, E)$ platí

$$(\tilde{V} \subset V) \wedge (\tilde{E} \subset E), \tag{1.6}$$

nazveme graf $\tilde{\mathcal{G}}(\tilde{V}, \tilde{E})$ *podgrafem grafu* $\mathcal{G}(V, E)$.

[1] Podle [4] se nazývá *bi-adjacency matrix* a podle [5] zase *bipartite adjacency matrix*.

Definice 1.1.10. (sled) [11] V grafu $\mathcal{G}(V, E)$ *sledem* délky k nazveme posloupnost $(v_0, v_1, \ldots, v_k)$ vrcholů z V takovou, že $\forall i \in \hat{k} \quad \{v_{i-1}, v_i\} \in E$. Jednoduchý graf, jehož vrcholy lze uspořádat do lineární sekvence:

$$V = \{x_1, x_2, \ldots, x_n\}, \quad E = \{\{x_1, x_2\}, \{x_2, x_3\}, \ldots, \{x_{n-1}, x_n\}\}, \tag{1.7}$$

Definice 1.1.11. (cesta) [5], [11] Jednoduchý graf, jehož vrcholy lze uspořádat do lineární sekvence:

$$V = \{x_1, x_2, \ldots, x_n\}, \quad E = \{\{x_1, x_2\}, \{x_2, x_3\}, \ldots, \{x_{n-1}, x_n\}\}, \tag{1.8}$$

nazveme *cestou*. Vrcholy se nesmí opakovat ($i \neq j \Rightarrow x_i \neq x_j$). Vrcholy x_1 a x_n jsou *krajní* a vrcholy $x_2, x_3, \ldots, x_{n-1}$ jsou *vnitřní*.

Poznámka 1.1.12. Cesta v grafu je sled, jehož vrcholy se neopakují.

Věta 1.1.13. [11] Pro graf $\mathcal{G}(\boldsymbol{A})$ zadaný maticí $\boldsymbol{A}$ platí, že prvek matice

$$\left(\boldsymbol{A}^k\right)_{ij} = \text{počet sledů délky } k \text{ mezi vrcholy } i \text{ a } j. \tag{1.9}$$

1.2 Poissonovo rozdělení

Především u generujících modelů využijeme Poissonovo rozdělení pravděpodobnosti. Poissonovo rozdělení je vhodné pro modelování nezávislého výskytu událostí. Často se označuje jako rozdělení bez paměti, při modelování výskytu událostí v čase. To, že událost v čase nastala, nijak neovlivňuje její budoucí výskyty. V našem případě při použití na výskyt hrany to analogicky znamená, že existence jedné hrany mezi dvěma vrcholy nijak neovlivní existenci jiné hrany.

Poissonovo rozdělení má jeden parametr, často se značí λ. Při použití rozdělení na události v čase, má význam jako průměrný výskyt událostí na jednotku času. V naší aplikaci na výskyt hrany, záleží na použitém modelu, ale obecně můžeme tento parametr chápat jako sílu vazby mezi vrcholy, nebo jejich skupinami [43].

Pravděpodobnost, že náhodná veličina X nabývá hodnoty x s Poissonovým rozdělením s parametrem λ je

$$P[\ X = x\ ] = \frac{\lambda^x}{x!} \exp(-\lambda). \tag{1.10}$$

Vícerozměrnou variantu Poissonova rozdělení získáme jako produkt pravděpodobností, že složky náhodného vektoru nabývají odpovídajících hodnot [33]:

$$P[\ \vec{X} = (x_1, x_2, \ldots, x_n)\ ] = \prod_{i=1}^{n} \frac{\lambda_i^{x_i}}{x_i!} \exp(-\lambda_i) \tag{1.11}$$

Při generování hran do náhodného grafu s pomocí Poissonova rozdělení, pravděpodobnost, že při generování Y grafů na množině vrcholů V vznikne mezi vrcholy i a j x-krát hrana je [19]

$$P[\ x \text{ hran v } Y \text{ grafech }] = \frac{(\lambda Y)^x}{x!} \exp(-\lambda Y). \tag{1.12}$$

Můžeme se ptát, po kolika generování vznikne další hrana [19].

$$P[\ \text{další hrana } \{i, j\} \text{ vznikne po } Y \text{ generování }] = \exp(-\lambda Y). \tag{1.13}$$

$$P[\text{ hrana } \{i,j\} \text{ v } Y \text{ grafech už vznikla }] = 1 - \exp(-\lambda Y). \tag{1.14}$$

Pro naše generování nás zajímá, jestli hrana vznikla již v prvním generování grafu, tedy

$$p(i,j) = P[\text{ hrana } \{i,j\} \text{ v } Y = 1 \text{ grafech už vznikla }]. \tag{1.15}$$

1.3 Podobnostní míry a metriky

1.3.1 Kullbackova-Leiblerova divergence

Kullbackova-Leiblerova divergence je známá také jako *relativní entropie*, která poměřuje rozdíly mezi dvěmi pravděpodobnostními distribucemi [21]. Kullbackova-Leiblerova divergence není metrika, protože není symetrická [25].

Definice 1.3.1. (Kullbackova-Leiblerova divergence) [22] Buďte p, q pravděpodobnostní distribuce. Potom

$$\mathrm{KL}(p||q) = \sum_i p(i) \log \frac{p(i)}{q(i)} \tag{1.16}$$

nazveme *Kullbackovou-Leiblerovou divergencí*.

1.3.2 Frobeniova norma

Frobeniova norma je maticovou normou, která je blízká euklidovskému prostoru.

Definice 1.3.2. (Frobeniova norma) [39] Buď $\boldsymbol{A} \in \mathbb{R}^{n,m}$ matice. Potom

$$||\boldsymbol{A}||_F = \sqrt{\sum_{i=1}^{n}\sum_{j=1}^{m} |\boldsymbol{A}_{ij}|^2} \tag{1.17}$$

nazveme *Frobeniovou normou*.

1.3.3 Jaccardův index

Jaccardův index je podobnostní míra, známá také jako *relativní překryv*. Nabývá hodnot mezi 0 a 1 a je dán poměrem počtu stejných prvků dvou množin vůči celkovému počtu prvků v obou množinách [14].

Definice 1.3.3. (Jaccardův index) [14] Buďte A, B spočetné množiny a nechť $A \cup B \neq \emptyset$. Potom

$$\mathrm{J}(A,B) = \frac{|A \cap B|}{|A \cup B|}. \tag{1.18}$$

nazveme *Jaccardovým indexem*.

1.3.4 Randův index

Jestliže Jaccardův index lze chápat jako míru relativního překryvu, pak Randův index je *míra absolutního překryvu*. Nabývá hodnot mezi 0 a 1 a je dán poměrem počtu stejných prvků dvou podmnožin nějakého celku vůči celkovému počtu prvků v tomto celku [37].

Definice 1.3.4. (Randův index) [37] Buď $X \neq \emptyset$ spočetná množina a $A, B \subset X$ podmnožiny. Potom

$$\mathrm{R}(A,B) = \frac{|A \cap B|}{|X|}. \tag{1.19}$$

nazveme *Randovým indexem*.

1.4 Jensenova nerovnost

Věta 1.4.1. Nechť φ je reálná konvexní funkce na uzavřeném intervalu $[a,b]$, $\forall z : y_z \in [a,b]$. Potom

$$\varphi\left(\sum_z \lambda_z y_z\right) \leq \sum_z \lambda_z \varphi\left(y_z\right), \tag{1.20}$$

kde $\forall z : \lambda_z \in [0,1] \wedge \sum_z \lambda_z = 1$ [9].

Jensenovu nerovnost využijeme ve tvaru pro $\varphi = -\log$ a $y_z = \frac{x_z}{\lambda_z}$:

$$\log\left(\sum_z x_z\right) \geq \sum_z \lambda_z \log\left(\frac{x_z}{\lambda_z}\right). \tag{1.21}$$

Rovnost nastává právě tehdy, když

$$\forall i,j : \quad \frac{x_i}{\lambda_i} = \frac{x_j}{\lambda_j}. \tag{1.22}$$

To plyne z toho, že z hlediska λ jako pravděpodobnosti při splnění podmínky (1.22) je v pravé i levé straně Jensenovy nerovnosti střední hodnota konstanty.

Kapitola 2

Analýza struktury dat

Analýza struktury dat je proces rozdělování množiny datových objektů do podmnožin. Cílem je získat z heterogenního celku několik více homogenních částí.

V mnohé literatuře, zvláště v té zabývající se analýzou multidimenzionálních dat [2, 20, 23, 33, 38, 41], se mezi tyto metody řadí

- shluková analýza (angl. clustering),
- dvojité shlukování (angl. biclustering),
- nebo faktorizace matic (angl. matrix factorization).

Výsledkem může být také aproximace celku sestavená kombinací faktorů s výrazně nižší dimenzí (NMF, SVD, PCA) [41].

2.1 Shluková analýza

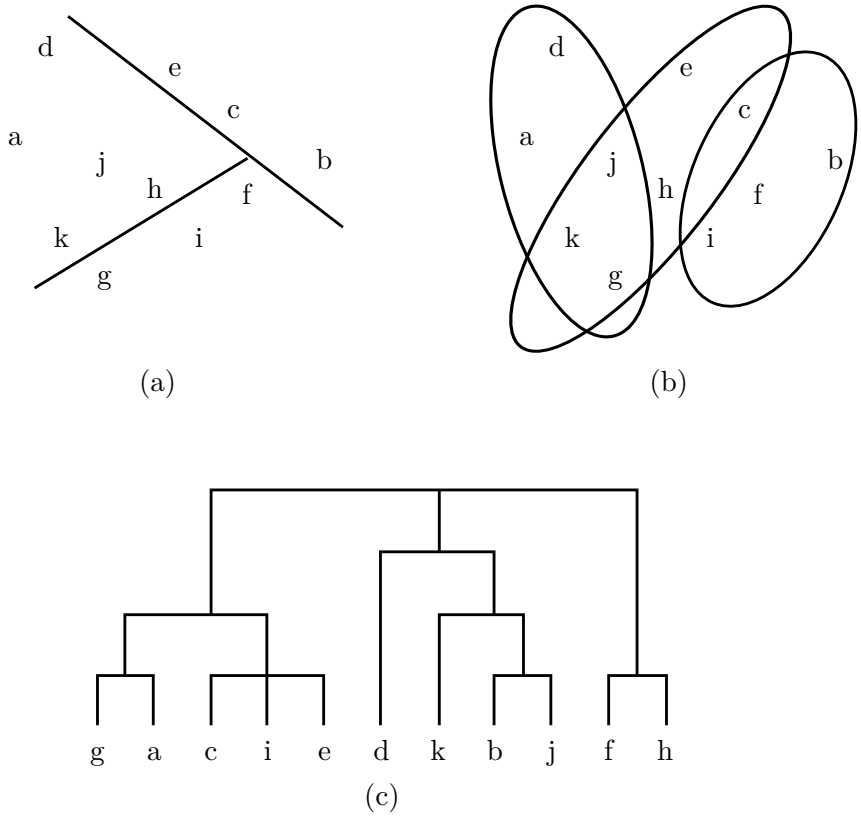
Shluková analýza (angl. cluster analysis) je proces rozdělování množiny datových objektů do podmnožin takových, že objekty uvnitř jsou si více podobné. Z tohoto důvodu potřebujeme funkci, která mezi objekty posuzuje podobnost (viz kapitola 1.3). Výše popsané podmnožiny jako více homogenní část nazveme *shluky* (angl. clusters) a platí, že datové objekty uvnitř stejného shluku jsou si podobnější, než objekty z různých shluků. Z hlediska strojového učení je shlukování učením bez učitele [18, 41].

Datové objekty uvažujme v podobě nějakých množin vlastností. Můžou to být třeba uspořádané n-tice čísel, řádky v tabulce nebo vrcholy matematického grafu apod. Podle použití mohou mít shluky různé podoby: disjunktní množiny, hyperhrany, pravděpodobnost členství nebo vnořené množiny. Shluky tak mohou být výlučné, překrývající se, pravděpodobnostní, hierarchické a další [41]. Příklady různých podob shluků jsou na obrázku 2.1.

2.1.1 Dvojité shlukování

V případě, že data vykazují multivariabilitu (tj. více typů objektů, např. lidi – tituly), je potřeba pracovat s datovými objekty a jejich vlastnostmi symetricky. Takový přístup bude demonstrován na dvojshlukování.

V některých aplikacích nestačí datové objekty shlukovat v jedné dimenzi. Dvojité shlukování (angl. biclustering) pracuje s datovými objekty a jejich vlastnostmi najednou. Hledá matice dat pro submatice šablon chápaných jako shluky. Výsledné shluky jsou známé jako dvojshluky (angl.



Obrázek 2.1: Různé typy shluků. (a) Nepřekrývající se shluky. (b) Překrývající se shluky (hyperhrany). (c) Hierarchická struktura shluků. Převzato z [41] a upraveno.

biclusters). Dvojshluky obsahují zpravidla pouze část objektů a část jejich vlastností. Algoritmy můžeme dělit na optimalizační a výčtové [18].

2.2 Faktorizace matic

Faktorizaci matic lze pochopit skrze následující myšlenku: Uvažme matici $\boldsymbol{M}$ tvořenou nezápornými prvky. Předpokládejme, že sloupce matice $\boldsymbol{M}$ vznikly jako kompozice vzorových vektorů (faktorů), které slouží jako šablony pro sloupce. Ideálně by každý sloupec matice $\boldsymbol{M}$ byl lineární kombinací těchto faktorů. Avšak skutečná data se zpravidla od těchto lineárních kombinací můžou lišit.

Úkolem algoritmů pro faktorizaci matic [1, 21, 22, 27, 44] je odhalit v matici $\boldsymbol{M}$ tyto faktory a sloupce matice popsat jejich kombinací. Výsledkem jsou dvě matice $\boldsymbol{F}$ a $\boldsymbol{G}$. Matice $\boldsymbol{F}$ je po sloupcích tvořena jednotlivými faktory. Matice $\boldsymbol{G}$ de facto popisuje matici $\boldsymbol{M}$ vyjádřenou pomocí faktorů z matice $\boldsymbol{F}$.

Poznámka 2.2.1. Například, když budeme zkoumat faktorizací matic *dokumenty a slova* [1], tak

- sloupce matice $\boldsymbol{M}$ reprezentují slova a řádky představují dokumenty,
- hodnota prvků matice $\boldsymbol{M}$ znamená počet výskytů slov v dokumentu,
- výsledné faktory (sloupce v matici $\boldsymbol{F}$) souvisí s tématy v dokumentech.

Datové objekty budeme reprezentovat jako nezáporné číselné vektory vlastností objektů. Tyto vektory, jakožto sloupce, poskládáme vedle sebe do matice $\boldsymbol{M}$. Pro tuto matici nalezneme faktory a kombinace velmi podobné sloupcům [27]. Součin

$$\boldsymbol{F} \cdot \boldsymbol{G} = \widehat{\boldsymbol{M}}. \tag{2.1}$$

je aproximací původní matice $\boldsymbol{M}$.

Algoritmy často generují kromě faktorů současně ke každému vektoru koeficienty, jak jej pomocí faktorů nakombinovat. Tyto algoritmy bývají založeny na minimalizování vzdálenosti původní matice $\boldsymbol{M}$ a její aproximace $\widehat{\boldsymbol{M}}$. K tomu se využije nějaká metrika nebo norma, např. euklidovská, frobeniova, apod [27].

První zásadní vlastností faktorizace matic je vyjádření společných vlastností vektorů (příp. datových objektů) pomocí faktorů. Druhou zásadní vlastností je redukce dimenze. Snahou je nalezení co nejmenšího množství faktorů, který již dostatečně vystihuje obsah matice $\boldsymbol{M}$. Pro n datových objektů a m jejich vlastností máme matici $\boldsymbol{M}$ rozměru $m \times n$. Počet faktorů k by měl být $k \leq n$.

Lze očekávat, že sloupce matice $\boldsymbol{F}$ budou lineárně nezávislé. Pro tvorbu matice $\boldsymbol{G}$ jsou lineárně závislé faktory nadbytečné, proto je nutné lineární kombinace z matice $\boldsymbol{F}$ vyřadit. Sníží se tím hodnota k. Sloupce matice $\boldsymbol{F}$ potom tvoří bázi podprostoru s dimenzí k menší než n. Vidíme ze vztahu 2.1, že matice $\boldsymbol{G}$ je aproximace $\widehat{\boldsymbol{M}}$ matice $\boldsymbol{M}$ vyjádřena v tomto podprostoru. Matice $\boldsymbol{F}$ je maticí přechodu mezi bázemi [27].

Třetí zásadní vlastností je nezápornost prvků matic $\boldsymbol{M}, \boldsymbol{F}$ a $\boldsymbol{G}$. Tato vlastnost umožnuje algoritmům fungovat. Řešení faktorizace není jednoznačné, ale díky nezápornosti prvků může být nalezeno extremální řešení. Existují příbuzné metody, které rozkládají i matice se zápornými prvky. Podmínku nezápornosti prvků může vynahradit to, že nalezené faktory budou kolmé. Příkladem těchto metod jsou SVD (singular value decomposition) nebo PCA (principal component analysis).

2.3 Detekce komunit

Předchozí kapitoly popisují metody pracující v multidimenzionálním prostoru, ve kterém lze zavést podobnost, typicky například na základě metrik. Ne všechna data mohou být popsána v takovém prostoru. Z tohoto důvodu se zavádí detekce komunit, využívající topologických vlastností řídkých datových struktur. Topologické vlastnosti se modelují pomocí matematického grafu. Mezi datovými objekty není určena vzdálenost, ale pouze existence, nebo neexistence hrany. Když nelze mezi datovými objekty stanovit nějakou míru, vzdálenost nebo podobnost a tyto „vzdálenosti“ mezi objekty nelze určovat a zbývají pouze topologické vlastnosti, modelované pomocí grafů, začínáme hovořit o detekci komunit.

Komunita je poměrně vágní pojem ohýbaný podle použití. Základní popis definuje komunitu jako množinu vrcholů v matematickém grafu, které jsou vzájemně propojeny více mezi sebou, než vůči zbytku grafu [15,33]. Naopak pro účely bipartitního a vícepartitního grafu, kdy vrcholy stejné parity nesmí být propojeny vůbec, musí být komunita chápána, jako společná šablona pro hrany. Tedy, že komunitu tvoří vrcholy propojené stejným způsobem do stejného sousedství [26].

2.3.1 Detekce komunit bez překryvu

Základním přístupem [6, 8, 13, 17, 24, 32] je rozdělit vrcholy grafu do disjunktních množin, které jsou propojeny hranami hustěji, než zbytek grafu. Naivní přístup by vyhledával v grafu úzká hrdla (angl. bottle-neck), což jsou hrany, přes které vede nejvíce nejkratších cest mezi všemi vrcholy. Odebíráním úzkých hrdel se graf bude rozpadat na komponenty souvislosti odpovídajících komunitám.

Komunity bez překryvu, ačkoliv jsou disjunktní, mohou být do sebe vnořené. Naivní přístup neřeší kolik komunit (komponent souvislosti) je ideální počet. Tento problém řeší [30] zavedením tzv. modularity. To je objektivní hodnotící funkce rozdělení vrcholů, která nabývá maxima při vhodném rozdělení vrcholů do komunit.

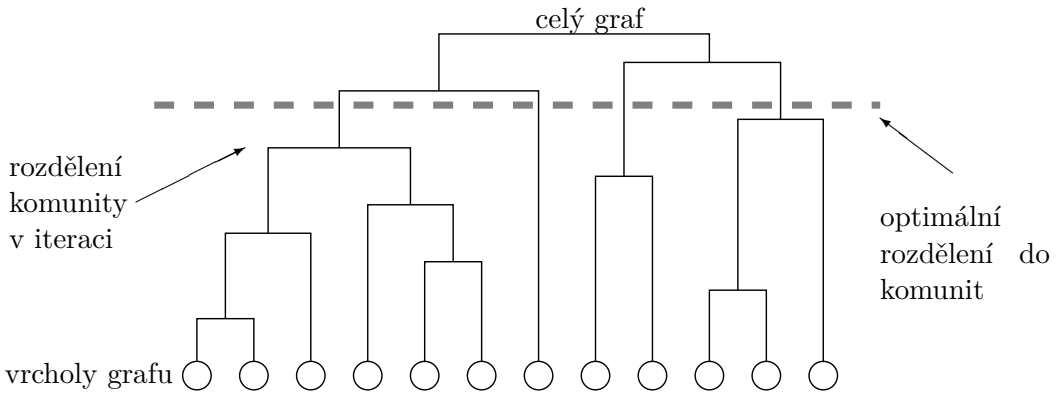
V základu je dendrogram kořenový strom D reprezentující sekvenci dělení vrcholů (viz obrázek 2.2). Jeho kořen zastupuje celý graf. Vrcholy v jednotlivých patrech stromu představují podmnožiny (komunity) vrcholů a listy tohoto stromu ztělesňují vrcholy původního grafu $\mathcal{G}$ jako *solo-komunity* [32].

Do dendrogramu je možné současně promítnout jednotlivé iterace algoritmu, který komunity buď rozděluje, nebo je slučuje. Potom pro vykreslení dendrogramu použijeme svislé hrany, které se rozdvojí (příp. sloučí) v iteraci a vedou od kořene až po listy (příp. od listů ke kořeni). Příklad takového dendrogramu je na obrázku 2.2 [32]. Kromě těchto slučovacích a štěpících přístupů existují také optimalizační. Zástupcem optimalizačního přístupu se zabývám v kapitole 3.

2.3.2 Překrývající se komunity

V některých případech není dostačující, aby vrchol patřil do právě jedné komunity. Potom přistupujeme k tomu, aby vrchol patřil do více komunit současně.

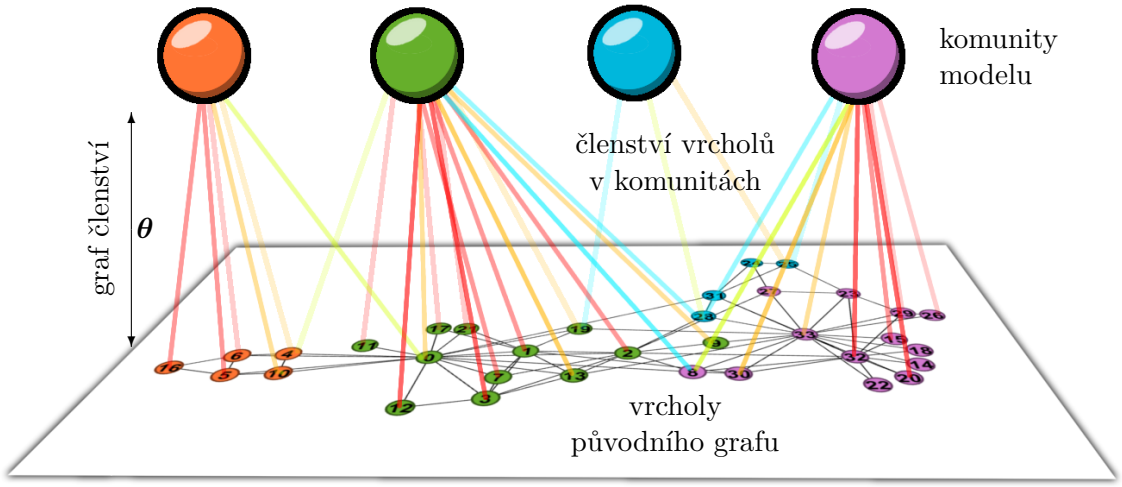
Koncept překrývajících se komunit je založen na tom, že hrany v grafu existují na základě komunit. Komunita je příčinou existence hrany mezi vrcholy. Jsou-li dva vrcholy společně v jedné komunitě, je pravděpodobnost existence hrany spojující tyto vrcholy vyšší, než určitý téměř nulový práh ϵ. Vrchol, který patří do více komunit současně, může mít vyšší stupeň. Mezi vrcholy, které sdílejí společně stejné komunty, bude s velikou pravděpodobností existovat



Obrázek 2.2: Dendrogram. Strom zobrazující postup hierarchického shlukování. Nahoře je celý graf jako jedna komunita. Dole jsou vrcholy grafu v samostatných komunitách. Svislé přímky reprezentují (pod)komunity. Vodorovné přímky zobrazují krok algoritmu: sloučení komunit, respektive rozštěpení komunity. Algoritmus může postupovat shora-dolů, respektive zdola-nahoru. Převzato z [32] a upraveno.

hrana [3, 15, 29, 32, 42, 43]. Mezi vrcholy, které nemají společnou komunitu, by hrana existovat neměla, maximálně s pravděpodobností prahu ϵ. Popsaný princip je zobrazen na obrázku 2.3.

Metody detekce překrývajících se komunit [3, 29, 43] do jisté míry připomínají faktorizaci matic. Konkrétně faktorizaci matice sousednosti (angl. adjacency matrix). Zasádní rozdíl oproti běžné faktorizaci matic dělají prvky matice sousednosti, které tvoří pouze jedničky a nuly. Další rozdíl je podobnostní míra, podle které se faktorizace provádí. V detekci komunit se narozdíl od „tvrdých“ metrik (např. frobeniova norma (1.17)) používají pravděpodobnostní míry, jako Kullback-Leiblerovu divergenci (1.16). Hledání řešení se provádí maximalizováním nějaké věrohodnostní funkce.



Obrázek 2.3: Ilustrace modelu překrývajících se komunit. Obrázek ukazuje dva grafy. Model lze chápat jako bipartitní graf s ohodnocenými hranami. Vrcholy modelu jsou komunity (nahoře) a vrcholy původního grafu (dole). V dolní části je celý původní graf: vrcholy a hrany. Model chápe hrany v původním grafu jako projekci.

Kapitola 3

Stochastické blokové modely

Princip *stochastického blokového modelu* (angl. stochastic block-model) [24, 35] spočívá v rozdělení vrcholů grafu do skupin a následného popisu grafu pomocí grafu tvořeného skupinami (komunitami). Takto popisuje třídy ekvivalence grafů. SBM model lze tak využít ke generování náhodných grafů z dané třídy nebo s vhodným algoritmem k nalezení komunit.

Stochastický blokový model je velmi obecný a lze jím popsat grafy různého typu. Pro detekci komunit stačí nalézt parametry blokového modelu tak, aby s největší pravděpodobností blokový model generoval takové grafy jako ten, který je podroben ke zkoumání.

V této práci se zabývám stochastickými blokovými modely s Poissonovým rozdělením, kterými popisuji strukturu jak unipartitních grafů, tak bipartitních grafů. Předpokládáme, že existence hrany $\{i, j\}$ neovlivňuje existenci hrany $\{u, v\}$. Tedy, že se existence dvou různých hran jsou nezávislé jevy. Pro model z této třídy můžu z jeho parametrů vyjádřit matici středních hodnot Poissonova rozdělení $\boldsymbol{\Lambda}$. Matice $\boldsymbol{\Lambda}$ má stejný rozměr jako matice sousednosti grafu a je symetrická.

Definice 3.0.1. *Stochastický blokový model s Poissonovým rozdělením* je stochastický blokový model, z jehož parametrů lze vyjádřit matice $\boldsymbol{\Lambda}$ o rozměru $n \times n$, kde n je počet vrcholů, s vlastností

$$\sum_{ij=1}^{n} \Lambda_{ij} = 2\mathrm{E}m, \tag{3.1}$$

kde m je počet hran. A pravděpodobnost výskytu hrany je

$$p(\{i, j\}) = 1 - \exp(-\boldsymbol{\Lambda}_{ij}). \tag{3.2}$$

Máme-li daný graf $\mathcal{G}(\boldsymbol{A})$ zadaný maticí $\boldsymbol{A}$, pak detekcí komunit rozumíme hledání parametrů modelu, které maximalizují pravděpodobnost, že model odpovídá grafu $\mathcal{G}(\boldsymbol{A})$:

$$P(\mathcal{G}|\boldsymbol{\Lambda}) = \prod_{i<j} \frac{(\boldsymbol{\Lambda}_{ij})^{\boldsymbol{A}_{ij}}}{\boldsymbol{A}_{ij}!} \exp(-\boldsymbol{\Lambda}_{ij}) \cdot \prod_{i} \frac{(\frac{1}{2}\boldsymbol{\Lambda}_{ii})^{\boldsymbol{A}_{ii}/2}}{(\boldsymbol{A}_{ii}/2)!} \exp(-\frac{1}{2}\boldsymbol{\Lambda}_{ii}). \tag{3.3}$$

Obecně také můžeme říci, že hledání parametrů modelů spočívá v minimalizaci KL-divergence mezi grafem a jeho aproximací

$$\mathrm{KL}(\boldsymbol{A}||\boldsymbol{\Lambda}) = -\frac{1}{\sum_{ij} \boldsymbol{A}_{ij}} \sum_{ij} \boldsymbol{A}_{ij} \log \boldsymbol{\Lambda}_{ij} - \log \frac{\sum_{ij} \boldsymbol{\Lambda}_{ij}}{\sum_{ij} \boldsymbol{A}_{ij}}. \tag{3.4}$$

Věta 3.0.2. Nechť $\mathcal{G}(\boldsymbol{A})$ je jednoduchý neorientovaný graf bez smyček a $\sum_{ij} \boldsymbol{A}_{ij} = \sum_{ij} \boldsymbol{\Lambda}_{ij}$. Pak

$$\log P(\mathcal{G}(\boldsymbol{A})|\boldsymbol{\Lambda}) = -m \,\mathrm{KL}(\boldsymbol{A}||\boldsymbol{\Lambda}) - m, \tag{3.5}$$

kde $m = \frac{1}{2}\sum_{ij} \boldsymbol{A}_{ij}$ je počet hran.

Důkaz. Matice $\boldsymbol{A}$ a $\boldsymbol{\Lambda}$ jsou symetrické a matice $\boldsymbol{A}$ má nulovou diagonálu. Proto logaritmus (3.3) je

$$\log P(\mathcal{G}(\boldsymbol{A})|\boldsymbol{\Lambda}) = \frac{1}{2}\sum_{ij} \boldsymbol{A}_{ij} \log \boldsymbol{\Lambda}_{ij} - \frac{1}{2}\sum_{ij} \boldsymbol{\Lambda}_{ij}. \tag{3.6}$$

Druhý člen je podle předpokladu počet hran. První člen upravíme tak, aby prvky matic vystupovaly jako pravděpodobnosti:

$$\sum_{ij} \boldsymbol{A}_{ij} \log \boldsymbol{\Lambda}_{ij} = \sum_{ij} \boldsymbol{A}_{ij} \left(\log \frac{\boldsymbol{\Lambda}_{ij}}{\sum_{ij} \boldsymbol{\Lambda}_{ij}} + \log \sum_{ij} \boldsymbol{\Lambda}_{ij} \right) = \sum_{ij} \boldsymbol{A}_{ij} \log \frac{\boldsymbol{\Lambda}_{ij}}{\sum_{ij} \boldsymbol{\Lambda}_{ij}} + \sum_{ij} \boldsymbol{A}_{ij} \log \sum_{ij} \boldsymbol{\Lambda}_{ij} \tag{3.7}$$

$$\sum_{ij} \boldsymbol{A}_{ij} \log \frac{\boldsymbol{\Lambda}_{ij}}{\sum_{ij} \boldsymbol{\Lambda}_{ij}} = \underbrace{\left(\sum_{ij} \boldsymbol{A}_{ij} \right)}_{2m} \sum_{ij} \frac{\boldsymbol{A}_{ij}}{\sum_{ij} \boldsymbol{A}_{ij}} \log \frac{\boldsymbol{\Lambda}_{ij}}{\sum_{ij} \boldsymbol{\Lambda}_{ij}} \tag{3.8}$$

Nyní vyjádříme Kullbackovu-Leiblerovu divergenci:

$$\sum_{ij} \frac{\boldsymbol{A}_{ij}}{\sum_{ij} \boldsymbol{A}_{ij}} \log \frac{\boldsymbol{\Lambda}_{ij}}{\sum_{ij} \boldsymbol{\Lambda}_{ij}} = \underbrace{\sum_{ij} \frac{\boldsymbol{A}_{ij}}{\sum_{ij} \boldsymbol{A}_{ij}} \log \frac{\frac{\boldsymbol{\Lambda}_{ij}}{\sum_{ij} \boldsymbol{\Lambda}_{ij}}}{\frac{\boldsymbol{A}_{ij}}{\sum_{ij} \boldsymbol{A}_{ij}}}}_{-\mathrm{KL}(\boldsymbol{A}||\boldsymbol{\Lambda})} + \sum_{ij} \frac{\boldsymbol{A}_{ij}}{\sum_{ij} \boldsymbol{A}_{ij}} \log \frac{\boldsymbol{A}_{ij}}{\sum_{ij} \boldsymbol{A}_{ij}} \tag{3.9}$$

$$\sum_{ij} \boldsymbol{A}_{ij} \log \boldsymbol{\Lambda}_{ij} = -2m \,\mathrm{KL}(\boldsymbol{A}||\boldsymbol{\Lambda}) + \sum_{ij} \boldsymbol{A}_{ij} \log \frac{\boldsymbol{A}_{ij}}{\sum_{ij} \boldsymbol{A}_{ij}} + \sum_{ij} \boldsymbol{A}_{ij} \log \sum_{ij} \boldsymbol{\Lambda}_{ij} \tag{3.10}$$

Graf je jednoduchý a proto prvky matice $\boldsymbol{A}$ nabývají hodnot 0 nebo 1.

$$\sum_{ij} \boldsymbol{A}_{ij} \log \boldsymbol{\Lambda}_{ij} = -2m \,\mathrm{KL}(\boldsymbol{A}||\boldsymbol{\Lambda}) + \sum_{ij} \underbrace{\boldsymbol{A}_{ij} \log \boldsymbol{A}_{ij}}_{0} + 2m \log \frac{\sum_{ij} \boldsymbol{\Lambda}_{ij}}{\sum_{ij} \boldsymbol{A}_{ij}} \tag{3.11}$$

Podle předpokladu jsou si sumy rovny a poslední člen se vynuluje.

$$\log \frac{\sum_{ij} \boldsymbol{\Lambda}_{ij}}{\sum_{ij} \boldsymbol{A}_{ij}} = \log \frac{2m}{2m} = \log 1 = 0 \tag{3.12}$$

Zíkali jsme vztah pro KL-divergenci

$$\sum_{ij} \boldsymbol{A}_{ij} \log \boldsymbol{\Lambda}_{ij} = -2m \,\mathrm{KL}(\boldsymbol{A}||\boldsymbol{\Lambda}), \tag{3.13}$$

který po dosazení do (3.6) dokazuje tvrzení. □

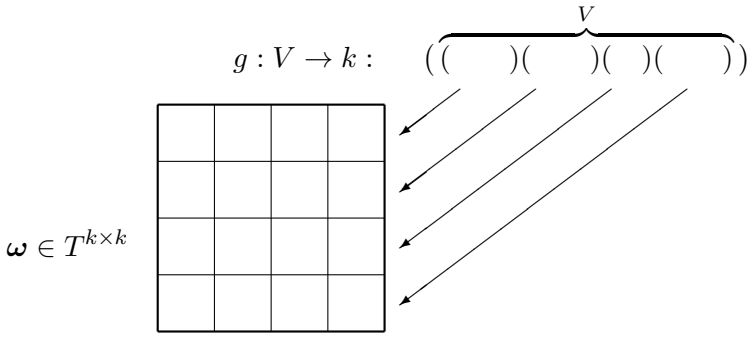
3.1 Standardní stochastický blokový model

Základní blokový model uvažuje rozdělení vrcholů do k skupin. Jeho popisující parametry jsou matice $\boldsymbol{\omega}$ rozměru $k \times k$ a vektor/zobrazení $g : V \to k$. Vrcholy jsou roztříděny do skupin vektorem/zobrazením $g : V \to k$. Matice $\boldsymbol{\omega}$ popisuje graf mezi skupinami, sílu a způsob propojení skupin. Blokový model generuje hranu mezi vrcholy i a j s pravděpodobností

$$p(i,j) = 1 - \exp\left(-\boldsymbol{\omega}_{g_i g_j}\right). \tag{3.14}$$

Vektorem $\vec{g}$ se pouze vybere, která střední hodnota z $\boldsymbol{\omega}$ se v pravděpodobnosti uplatní viz obrázek 3.1. Matice středních hodnot má tvar

$$\boldsymbol{\Lambda}_{ij} = \boldsymbol{\omega}_{g_i g_j}. \tag{3.15}$$



Obrázek 3.1: Schematické znázornění stochastického blokového modelu. Zobrazení $\vec{g}$ mapuje disjunktní podmnožiny V na komunity, jimž odpovídají sloupce a řádky matice $\boldsymbol{\omega}$, kde V je množina vrcholů grafu.

Pro detekci komunit v grafu a odhadu jeho parametrů v tomto případě dosadíme matici středních hodnot (3.15) do (3.3). Blokový model s parametry $\boldsymbol{\omega}$ a $\vec{g}$ vygeneruje zkoumaný graf $\mathcal{G}(\boldsymbol{A})$ s pravděpodobností [24]:

$$P(\mathcal{G}|\boldsymbol{\omega}, \vec{g}) = \prod_{i<j} \frac{(\boldsymbol{\omega}_{g_i g_j})^{\boldsymbol{A}_{ij}}}{\boldsymbol{A}_{ij}!} \exp(-\boldsymbol{\omega}_{g_i g_j}) \cdot \prod_i \frac{(\frac{1}{2}\boldsymbol{\omega}_{g_i g_i})^{\boldsymbol{A}_{ii}/2}}{(\boldsymbol{A}_{ii}/2)!} \exp(-\frac{1}{2}\boldsymbol{\omega}_{g_i g_i}). \tag{3.16}$$

Pro jednoduchý neorientovaný graf odvodíme účelovou funkci zlogaritmováním (3.16) do tvaru

$$\log P(\mathcal{G}|\boldsymbol{\omega}, \vec{g}) = \frac{1}{2}\sum_{ij} \boldsymbol{A}_{ij} \log \boldsymbol{\omega}_{g_i g_j} - \frac{1}{2}\sum_{ij} \boldsymbol{\omega}_{g_i g_j} \tag{3.17}$$

a dosazením nejlepší odhadu matice parametrů $\boldsymbol{\omega}$

$$\hat{\boldsymbol{\omega}}_{rs} = \frac{\boldsymbol{m}_{rs}}{n_r n_s}, \tag{3.18}$$

kde m_{rs} je počet hran mezi vrcholy přiřazených do skupin r a s a n_r je počet vrcholů přiřazených do skupiny r [24]. Odhad matice $\boldsymbol{\omega}$ získáme diferencováním (3.17). Účelová funkce je po dosazení závislá pouze na rozdělení vrcholů do skupin vektorem $\vec{g}$:

$$F(\vec{g}) = \log P(\mathcal{G}|\hat{\boldsymbol{\omega}}(\vec{g}), \vec{g}) = \frac{1}{2}\sum_{ij} \boldsymbol{A}_{ij} \log \frac{\boldsymbol{m}_{g_i g_j}}{n_{g_i} n_{g_j}} - \frac{1}{2}\sum_{ij} \frac{\boldsymbol{m}_{g_i g_j}}{n_{g_i} n_{g_j}}. \tag{3.19}$$

Pro nalezení vektoru $\vec{g}$ se používá v [24] modifikovaný Kernighanův-Linův algoritmus, který je podrobněji popsán v sekci 3.1.3.

Věta 3.1.1. Pro jednoduchý neorientovaný graf $\mathcal{G}(\boldsymbol{A})$ zadaný maticí $\boldsymbol{A}$ je účelová funkce

$$F(\vec{g}) = \frac{1}{2}\sum_{rs} \boldsymbol{m}_{rs} \log \boldsymbol{m}_{rs} - \sum_{r} \kappa_r \log n_r - m, \tag{3.20}$$

kde m je počet hran.

Důkaz. Pomocí $\boldsymbol{A}$ a $\vec{g}$ vyjádříme některé paramtery:

$$d_i = \sum_j \boldsymbol{A}_{ij}, \quad \kappa_r = \sum_i d_i \delta_{rg_i}, \quad n_r = \sum_i \delta_{rg_i}, \quad \boldsymbol{m}_{rs} = \sum_{ij} \boldsymbol{A}_{ij}\delta_{rg_i}\delta_{sg_j}, \tag{3.21}$$

kde d_i je stupeň vrcholu i, κ_r je stupeň komunity r, n_r je počet vrcholů v komunitě r (mohutnost komunity), $\boldsymbol{m}_{rs}$ je počet hran mezi komunitami r a s.

Ukažme, že druhý člen vztahu (3.19) je počet hran:

$$\frac{1}{2}\sum_{ij} \frac{\boldsymbol{m}_{g_ig_j}}{n_{g_i}n_{g_j}} = \frac{1}{2}\sum_{ij}\sum_{rs} \frac{\boldsymbol{m}_{rs}}{n_r n_s}\delta_{rg_i}\delta_{sg_j} = \frac{1}{2}\sum_{rs} \frac{\boldsymbol{m}_{rs}}{n_r n_s} \underbrace{\sum_i \delta_{rg_i}}_{n_r} \underbrace{\sum_j \delta_{sg_j}}_{n_s} = \frac{1}{2}\sum_{rs} \boldsymbol{m}_{rs} \tag{3.22}$$

a součet prvků matice $\boldsymbol{m}$ je roven součtu prvků $\boldsymbol{A}$:

$$\frac{1}{2}\sum_{rs} \boldsymbol{m}_{rs} = \frac{1}{2}\sum_{rs}\sum_{ij} \boldsymbol{A}_{ij}\delta_{rg_i}\delta_{sg_j} = \frac{1}{2}\sum_{ij} \boldsymbol{A}_{ij} \underbrace{\sum_r \delta_{rg_i} \sum_s \delta_{sg_j}}_{1} = \frac{1}{2}\sum_{ij} \boldsymbol{A}_{ij} = m. \tag{3.23}$$

První člen vztahu (3.19) můžeme díky logaritmu rozložit na další trojici:

$$\frac{1}{2}\sum_{ij} \boldsymbol{A}_{ij} \log \frac{\boldsymbol{m}_{g_ig_j}}{n_{g_i}n_{g_j}} = \frac{1}{2}\sum_{ij} \boldsymbol{A}_{ij} \log \boldsymbol{m}_{g_ig_j} - \frac{1}{2}\sum_{ij} \boldsymbol{A}_{ij} \log n_{g_i} - \frac{1}{2}\sum_{ij} \boldsymbol{A}_{ij} \log n_{g_j}. \tag{3.24}$$

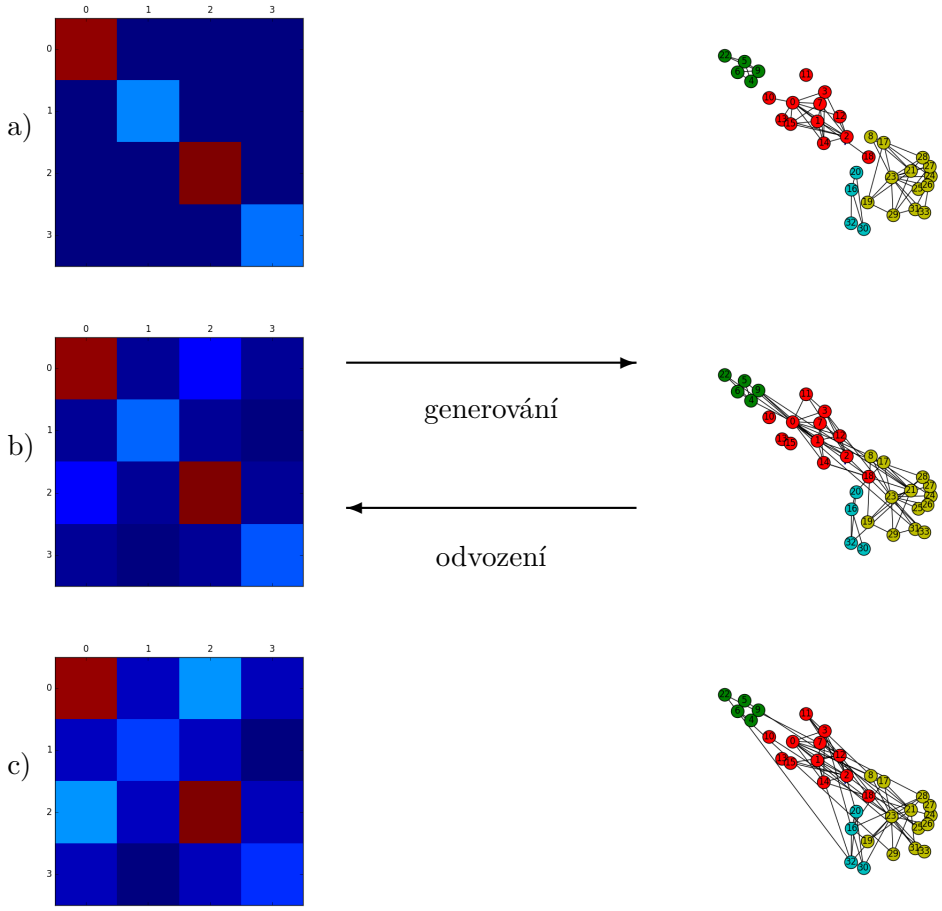
Dosazením z (3.21) upravme

$$\sum_{ij} \boldsymbol{A}_{ij} \log \boldsymbol{m}_{g_ig_j} = \sum_{rs} \underbrace{\sum_{ij} \boldsymbol{A}_{ij}\delta_{rg_i}\delta_{sg_j}}_{\boldsymbol{m}_{rs}} \log \boldsymbol{m}_{rs} = \sum_{rs} \boldsymbol{m}_{rs} \log \boldsymbol{m}_{rs} \tag{3.25}$$

a poslední dva členy z trojice analogicky:

$$\sum_{ij} \boldsymbol{A}_{ij} \log n_{g_i} = \sum_{r}\sum_{ij} \boldsymbol{A}_{ij}\delta_{rg_i} \underbrace{\sum_s \delta_{sg_j}}_{1} \log n_r = \sum_r \underbrace{\sum_{ij} \boldsymbol{A}_{ij}\delta_{rg_i}}_{\kappa_r} \log n_r = \sum_r \kappa_r \log n_r. \tag{3.26}$$

Dosazením zpět do (3.19) získáme tvrzení. □



Obrázek 3.2: Ukázka významu matice $\boldsymbol{\omega}$ v SBM. Červená znamená vyšší hodnotu. (a) Čím vyšší je hodnota na diagonále, tím hustěji je propojená komunita. (b, c) Čím vyšší je hodnota mimo diagonálu, tím více jsou propojeny komunity mezi sebou. SBM může grafy generovat nebo pomocí odhadu prvků matice $\boldsymbol{\omega}$ lze komunity detekovat.

3.1.1 Stochastický blokový model s kontrolou stupňů

Standardní stochastický blokový model nezohledňuje stupně vrcholů. To podle poznámky 3.1.2 vede na rovnoměrné rozdělení stupňů vrcholů. U grafů vytvořených na základě reálných dat se projevují rozmanité distribuce stupňů vrcholů [34, 40]. Z tohoto důvodu je v [24] představen modifikovaný stochasticky blokový model s kontrolou velikosti stupňů vrcholů. K předchozím parametrům $\boldsymbol{\omega}$ a $\vec{g}$ se přidá vektor $\vec{\theta}$ velikosti n. Hodnoty θ_i splňují normalizační podmínku vůči skupině

$$\forall r \in k, \quad \sum \vec{\theta}_i \delta_{r g_i} = 1. \tag{3.27}$$

Pak lze $\vec{\theta}_i$ interpretovat tak, že hrana vedoucí do skupiny g_i povede s pravděpodobností $\vec{\theta}_i$ právě do vrcholu i. Matice středních hodnot modelu má tvar

$$\boldsymbol{\Lambda}_{ij} = \theta_i \theta_j \boldsymbol{\omega}_{g_i g_j}. \tag{3.28}$$

Pokud by se model použil ke generování grafů, pak střední hodnoty stupňů jeho vrcholů by platilo [24]

$$\bar{d_i} = \kappa_{g_i}\theta_i. \tag{3.29}$$

Poznámka 3.1.2. (Degradace SBM s kontrolou stupňů na standardní SBM)
Parametr $\vec{\theta_i}$ funguje jako pravděpodonost, že hrana vedoucí do komunity g_i, skončí ve vrcholu i. Kdyby tato pravděpodobnost byla rovnoměrná

$$\theta_i = \frac{1}{n_{g_i}}, \tag{3.30}$$

pak bude platit

$$\theta_i\theta_j\hat{\boldsymbol{\omega}}_{g_ig_j} = \frac{m_{g_ig_j}}{n_{g_i}n_{g_j}}, \tag{3.31}$$

což znamená, že výskyt na levé straně můžeme nahradit odhadem $\hat{\boldsymbol{\omega}}$ z (3.18). Tím *standardní blokový model* a *blokový model s kontrolou stupňů* splynou.

Pro detekci komunit v grafu $\mathcal{G}(\boldsymbol{A})$ zadaného maticí $\boldsymbol{A}$ upraveným blokovým modelem použijeme pravděpodobnost, že SBM vygeneruje zkoumaný graf $\mathcal{G}(\boldsymbol{A})$ [24]:

$$P(\mathcal{G}|\vec{\theta},\boldsymbol{\omega},\vec{g}) = \prod_{i<j}\frac{(\theta_i\theta_j\boldsymbol{\omega}_{g_ig_j})^{\boldsymbol{A}_{ij}}}{\boldsymbol{A}_{ij}!}\exp(-\theta_i\theta_j\boldsymbol{\omega}_{g_ig_j})\cdot\prod_i\frac{(\frac{1}{2}\theta_i^2\boldsymbol{\omega}_{g_ig_i})^{\boldsymbol{A}_{ii}/2}}{(\boldsymbol{A}_{ii}/2)!}\exp(-\frac{1}{2}\theta_i^2\boldsymbol{\omega}_{g_ig_i}). \tag{3.32}$$

Pro jednoduchý neorientovaný graf odvodíme věrohodnost zlogaritmováním (3.32):

$$\log P(\mathcal{G}|\vec{\theta},\boldsymbol{\omega},\vec{g}) = \frac{1}{2}\sum_{ij}\boldsymbol{A}_{ij}\log\left(\theta_i\theta_j\boldsymbol{\omega}_{g_ig_j}\right) - \frac{1}{2}\sum_{ij}\theta_i\theta_j\boldsymbol{\omega}_{g_ig_j}. \tag{3.33}$$

Věta 3.1.3. Pro jednoduchý neorientovaný graf $\mathcal{G}(\boldsymbol{A})$ zadaný maticí $\boldsymbol{A}$ a SBM s kontrolou stupňů jsou nejlepší odhady $\boldsymbol{\omega}$ a $\vec{\theta}$

$$\hat{\theta}_i = \frac{d_i}{\kappa_{g_i}}, \qquad \hat{\boldsymbol{\omega}}_{rs} = \boldsymbol{m}_{rs}, \tag{3.34}$$

kde $\boldsymbol{m}_{rs}$ je počet hran mezi vrcholy přiřazených do skupin r a s, n_r je počet vrcholů přiřazených do skupiny r a κ_r je stupeň komunity [24].

Důkaz. Budeme diferencovat (3.33) a najdeme stacionární body:

$$\frac{\partial}{\partial\boldsymbol{\omega}_{rs}}\left[\frac{1}{2}\sum_{ij}\boldsymbol{A}_{ij}\log\left(\theta_i\theta_j\boldsymbol{\omega}_{g_ig_j}\right) - \frac{1}{2}\sum_{ij}\theta_i\theta_j\boldsymbol{\omega}_{g_ig_j}\right] = 0 \tag{3.35}$$

$$\frac{\partial}{\partial\boldsymbol{\omega}_{rs}}\left[\sum_{rs}\sum_{ij}\boldsymbol{A}_{ij}\delta_{rg_i}\delta_{sg_j}\log\left(\theta_i\theta_j\boldsymbol{\omega}_{rs}\right) - \sum_{rs}\sum_{ij}\delta_{rg_i}\delta_{sg_j}\theta_i\theta_j\boldsymbol{\omega}_{g_ig_j}\right] = 0 \tag{3.36}$$

$$\sum_{ij}\boldsymbol{A}_{ij}\delta_{rg_i}\delta_{sg_j}\frac{1}{\boldsymbol{\omega}_{rs}} - \sum_{ij}\theta_i\delta_{rg_i}\theta_j\delta_{sg_j} = 0 \tag{3.37}$$

Díky normalizační podmínce (3.27) dostáváme

$$\boldsymbol{\omega}_{rs} = \sum_{ij}\boldsymbol{A}_{ij}\delta_{rg_i}\delta_{sg_j} = \boldsymbol{m}_{rs}. \tag{3.38}$$

$$\frac{\partial}{\partial\theta_i}\left[\frac{1}{2}\sum_{ij}\boldsymbol{A}_{ij}\log\left(\theta_i\theta_j\boldsymbol{\omega}_{g_ig_j}\right)-\frac{1}{2}\sum_{ij}\theta_i\theta_j\boldsymbol{\omega}_{g_ig_j}\right]=0 \tag{3.39}$$

$$\sum_j \boldsymbol{A}_{ij}\frac{1}{\theta_i}-\sum_j\theta_j\boldsymbol{\omega}_{g_ig_j}=0 \tag{3.40}$$

Druhý člen upravíme:

$$\sum_j\theta_j\boldsymbol{\omega}_{g_ig_j}=\sum_{rs}\sum_j\delta_{rg_i}\delta_{sg_j}\theta_j\boldsymbol{\omega}_{rs}=\sum_{rs}\delta_{rg_i}\boldsymbol{\omega}_{rs}\underbrace{\sum_j\theta_j\delta_{sg_j}}_{1}=\sum_r\delta_{rg_i}\underbrace{\sum_s\boldsymbol{\omega}_{rs}}_{\kappa_r}=\kappa_{g_i}. \tag{3.41}$$

Získáváme stacionární bod

$$\theta_i=\frac{\sum_j\boldsymbol{A}_{ij}}{\kappa_{g_i}}=\frac{d_i}{\kappa_{g_i}}. \tag{3.42}$$

□

Dosazením nejlepších odhadů zpět do (3.33) získáme jako v předchozí kapitole účelovou funkci závislou pouze na rozdělení vrcholů do skupin danou vektorem $\vec{g}$. Vektor $\vec{g}$ nalezneme modifikovaným Kernighaným-Liným algoritmem, který je podrobněji popsán v sekci 3.1.3.

Věta 3.1.4. Pro jednoduchý neorientovaný graf $\mathcal{G}(\boldsymbol{A})$ zadaný maticí $\boldsymbol{A}$ je účelová funkce

$$F(\vec{g})=\frac{1}{2}\sum_{rs}\boldsymbol{m}_{rs}\log\boldsymbol{m}_{rs}+\sum_i d_i\log d_i-\sum_r\kappa_r\log\kappa_r-m, \tag{3.43}$$

kde m je počet hran.

Důkaz. Budeme postupovat analogicky jako v důkazu věty 3.1.1. Dosadíme nejlepší odhady (3.34) do (3.33):

$$F(\vec{g})=\frac{1}{2}\sum_{ij}\boldsymbol{A}_{ij}\log\frac{d_id_j\boldsymbol{m}_{g_ig_j}}{\kappa_{g_i}\kappa_{g_j}}-\frac{1}{2}\sum_{ij}\frac{d_id_j\boldsymbol{m}_{g_ig_j}}{\kappa_{g_i}\kappa_{g_j}}. \tag{3.44}$$

První člen rozdělíme díky logaritmu na trojici:

$$\sum_{ij}\boldsymbol{A}_{ij}\log\frac{d_id_j\boldsymbol{m}_{g_ig_j}}{\kappa_{g_i}\kappa_{g_j}}=\sum_{ij}\boldsymbol{A}_{ij}\log\boldsymbol{m}_{g_ig_j}+2\sum_{ij}\boldsymbol{A}_{ij}\log d_i-2\sum_{ij}\boldsymbol{A}_{ij}\log\kappa_{g_i}. \tag{3.45}$$

Poslední člen účelové funkce je roven počtu hran m:

$$\frac{1}{2}\sum_{ij}\frac{d_id_j\boldsymbol{m}_{g_ig_j}}{\kappa_{g_i}\kappa_{g_j}}=\frac{1}{2}\sum_{rs}\underbrace{\sum_i d_i\delta_{rg_i}}_{\kappa_r}\underbrace{\sum_j d_j\delta_{sg_j}}_{\kappa_s}\frac{\boldsymbol{m}_{rs}}{\kappa_r\kappa_s}=\frac{1}{2}\sum_{rs}\boldsymbol{m}_{rs}\overset{3.23}{=}m. \tag{3.46}$$

První člen jsme vyřešili v důkazu věty 3.1.1:

$$\sum_{ij}\boldsymbol{A}_{ij}\log\boldsymbol{m}_{g_ig_j}\overset{3.25}{=}\sum_{rs}\boldsymbol{m}_{rs}\log\boldsymbol{m}_{rs}. \tag{3.47}$$

Druhý člen se upraví triviálně:

$$\sum_{ij} \boldsymbol{A}_{ij} \log d_i = \sum_i \underbrace{\sum_j \boldsymbol{A}_{ij}}_{d_i} \log d_i = \sum_i d_i \log d_i. \tag{3.48}$$

Úprava prostředního členu je analogií (3.26):

$$\sum_{ij} \boldsymbol{A}_{ij} \log \kappa_{g_i} = \sum_r \sum_i \underbrace{\sum_j \boldsymbol{A}_{ij}}_{d_i} \delta_{rg_i} \log \kappa_r = \sum_r \underbrace{\sum_i d_i \delta_{rg_i}}_{\kappa_r} \log \kappa_r = \sum_r \kappa_r \log \kappa_r. \tag{3.49}$$

Dosazením upravených členů do (3.44) získáme tvrzení. □

3.1.2 Bipartitní SBM

Bipartitní graf musí splňovat podmínku, že neexistuje hrana mezi vrcholy stejné parity [11]. Vrcholy v bipartitním grafu jsou rozděleny do dvou disjunktních množin V a W. Vrcholům určíme jeden z typů a nebo b zobrazením [26]

$$t : V \cup W \to \{a, b\} : t_i \mapsto \begin{cases} a & \text{pro } i \in V, \\ b & \text{pro } i \in W. \end{cases} \tag{3.50}$$

Pro modelování bipartitních grafů přidáme do SBM podmínku [26]

$$t_i = t_j \Longrightarrow p(i, j) = 0. \tag{3.51}$$

Potom blokový model bude generovat hranu mezi vrcholy i a j s pravděpodobností [26]

$$p(i, j) = \begin{cases} 1 - \exp\left(-\theta_i \theta_j \omega_{g_i g_j}\right) & \text{pro } t_i \neq t_j, \\ 0 & \text{pro } t_i = t_j. \end{cases} \tag{3.52}$$

Poznámka 3.1.5. Podmínku (3.51) lze přeformulovat tak, že

- máme také skupiny jsou dvou parit,
- vrcholy jsou pomocí $\vec{g}$ přiřazeny do skupiny se shodnou paritou a
- graf nad skupinami $\mathcal{G}(\boldsymbol{\omega})$ je také bipartitní.

Fakticky se jedná o stejný model popsaný v předchozí kapitole 3.1.1, pouze KL-algoritmus popsaný v kapitole 3.1.3 nezkouší přiřadit vrchol do skupiny s jinou paritou. Jedná se tedy o optimalizaci při hledání komunit. Vrchol musí patřit do komunity se stejnou paritou.

3.1.3 Detekce komunit standardním SBM modelem

Detekce komunit se provádí fitováním parametrů SBM modelu k danému grafu. Na počátku jsou vrcholy náhodně rozděleny do skupin. V bipartitním případě jsou vrcholy rozděleny do skupin se shodnou paritou. Následně je rozdělení vrcholů do skupin upraveno Kernighan-Linovým algoritmem.

Věta 3.1.6. Pokud vrchol i přesuneme z komunity r do komunity s, změní se účelová funkce (3.43) o [24]:

$$\begin{aligned}\Delta F_{i,r\to s} = \sum_{t\neq r,s} [a(\boldsymbol{m}_{rt} - \boldsymbol{K}_{it}) - a(\boldsymbol{m}_{rt})a(\boldsymbol{m}_{st} + \boldsymbol{K}_{it}) - a(\boldsymbol{m}_{st})] \\ + a(\boldsymbol{m}_{rs} - \boldsymbol{K}_{is} + \boldsymbol{K}_{ir}) - a(\boldsymbol{m}_{rs}) \\ + \frac{1}{2}a(\boldsymbol{m}_{rr} - 2\boldsymbol{K}_{ir}) - \frac{1}{2}a(\boldsymbol{m}_{rr}) + \frac{1}{2}a(\boldsymbol{m}_{ss} + 2\boldsymbol{K}_{is}) - \frac{1}{2}a(\boldsymbol{m}_{ss}) \\ - a(\kappa_r - d_i) + a(\kappa_r) - a(\kappa_s + d_i) + a(\kappa_s),\end{aligned} \tag{3.53}$$

kde $a(x) = x\log(x)$ a $a(0) = 0$, $\boldsymbol{K}_{ir}$ je počet hran vedoucí z vrcholu i do vrcholů v komunitě r, κ_r je stupeň komunity r, d_i je stupeň vrcholu i.

Důkaz. Buď a funkce definovaná $a(x) = x\log(x)$ a $a(0) = 0$, pak účelová funkce (3.43) lze zapsat jako

$$F(\vec{g}) = \frac{1}{2}\sum_{rs} a(\boldsymbol{m}_{rs}) + \sum_i a(d_i) - \sum_r a(\kappa_r) - m. \tag{3.54}$$

Díky symetričnosti $\boldsymbol{A}_{ij}$ je symetrická také $\boldsymbol{m}_{rs}$:

$$\boldsymbol{m}_{rs} = \sum_{ij} A_{ij}\delta_{rg_i}\delta_{sg_j} = \sum_{ij} A_{ji}\delta_{rg_j}\delta_{sg_i} = \boldsymbol{m}_{sr}. \tag{3.55}$$

Díky symetričnosti $\boldsymbol{m}_{rs}$ můžeme rozepsat sumu

$$\sum_{tl} a(\boldsymbol{m}_{tl}) = \sum_{t,l\neq r,s} a(\boldsymbol{m}_{tl}) + 2\sum_{t\neq r,s} [a(\boldsymbol{m}_{tr}) + a(\boldsymbol{m}_{ts})] + 2a(\boldsymbol{m}_{rs}) + a(\boldsymbol{m}_{rr}) + a(\boldsymbol{m}_{ss}). \tag{3.56}$$

Analogicky můžeme rozepsat také

$$\boldsymbol{m}_{rs} = \sum_{ij} \boldsymbol{A}_{ij}\delta_{rg_i}\delta_{sg_j} = \sum_{i,j\neq u} \boldsymbol{A}_{ij}\delta_{rg_i}\delta_{sg_j} + \sum_{i\neq u} \boldsymbol{A}_{iu}(\delta_{rg_i}\delta_{sg_u} + \delta_{sg_i}\delta_{rg_u}) + \boldsymbol{A}_{uu}\delta_{rg_u}\delta_{sg_u}. \tag{3.57}$$

Stupeň komunity rozepíšeme jako

$$\kappa_r = \sum_{ij} \boldsymbol{A}_{ij}\delta_{rg_i} = \sum_{i\neq u}\sum_j \boldsymbol{A}_{ij}\delta_{rg_i} + \sum_j \boldsymbol{A}_{uj}\delta_{rg_u}. \tag{3.58}$$

Definujme

$$\boldsymbol{K}_{ir} = \sum_j \boldsymbol{A}_{ij}\delta_{rg_j} \tag{3.59}$$

pro počet hran vedoucí z vrcholu i do vrcholů v komunitě r.

Nechť $t \neq r, s$. Při přesunu vrcholu i z komunity r do komunity s, zůstanou zachovány hodnoty d_i, ale hodnoty matice $\boldsymbol{m}$ se změní následovně:

$$\begin{aligned}\boldsymbol{m}'_{rt} &= \boldsymbol{m}_{rt} - \sum_j \boldsymbol{A}_{ij}\delta_{tg_j} = \boldsymbol{m}_{rt} - \boldsymbol{K}_{it}, \\ \boldsymbol{m}'_{st} &= \boldsymbol{m}_{st} + \sum_j \boldsymbol{A}_{ij}\delta_{tg_j} = \boldsymbol{m}_{st} + \boldsymbol{K}_{it}, \\ \boldsymbol{m}'_{rr} &= \boldsymbol{m}_{rr} - 2\sum_j \boldsymbol{A}_{ij}\delta_{rg_j} = \boldsymbol{m}_{rr} - 2\boldsymbol{K}_{ir}, \\ \boldsymbol{m}'_{ss} &= \boldsymbol{m}_{ss} + 2\sum_j \boldsymbol{A}_{ij}\delta_{sg_j} = \boldsymbol{m}_{ss} + 2\boldsymbol{K}_{is}, \\ \boldsymbol{m}'_{rs} &= \boldsymbol{m}_{rs} + \sum_j \boldsymbol{A}_{ij}\delta_{rg_j} - \sum_j \boldsymbol{A}_{ij}\delta_{sg_j} = \boldsymbol{m}_{rs} + \boldsymbol{K}_{ir} - \boldsymbol{K}_{is}\end{aligned} \tag{3.60}$$

a stupně komunit $\vec{\kappa}$ se změní:

$$\begin{aligned}\kappa'_r &= \kappa_r - \sum_j \boldsymbol{A}_{ij} = \kappa_r - d_i,\\ \kappa'_s &= \kappa_s + \sum_j \boldsymbol{A}_{ij} = \kappa_s + d_i.\end{aligned} \tag{3.61}$$

Ve výrazu (3.54) se přesunem vrcholu mezi komunitami mění první a předposlední člen. S využitím rozkladů sum a rozdílem s čárkovanými hodnotami získáme tvrzení. □

Poznámka 3.1.7. Pro SBM bez kontroly stupňů je diference při přesunu vrcholu do jiné komunty

$$\begin{aligned}\Delta F_{i,r\to s} = \sum_{t\neq r,s} &[a(\boldsymbol{m}_{rt} - \boldsymbol{K}_{it}) - a(\boldsymbol{m}_{rt})a(\boldsymbol{m}_{st} + \boldsymbol{K}_{it}) - a(\boldsymbol{m}_{st})]\\ &+a(\boldsymbol{m}_{rs} - \boldsymbol{K}_{is} + \boldsymbol{K}_{ir}) - a(\boldsymbol{m}_{rs})\\ &+\frac{1}{2}a(\boldsymbol{m}_{rr} - 2\boldsymbol{K}_{ir}) - \frac{1}{2}a(\boldsymbol{m}_{rr}) + \frac{1}{2}a(\boldsymbol{m}_{ss} + 2\boldsymbol{K}_{is}) - \frac{1}{2}a(\boldsymbol{m}_{ss})\\ &-(\kappa_r - d_i)\log(n_r - 1) + \kappa_r \log n_r - (\kappa_s + d_i)\log(n_s + 1) + \kappa_s \log n_s.\end{aligned} \tag{3.62}$$

Kernighanův-Linův algoritmus zváží každému vrcholu změnu do jiné komunity. V bipartitním případě zkouší pouze komunity se stejnou paritou jako má vrchol. Provede se změna komunity vrcholu, která má maximální hodnotu a vrchol se vyřadí z množiny zkoumaných vrcholů. Každému vrcholu změní skupinu právě jednou. Algoritmus postupně podle maximální změny účelové funkce přesune všechny vrcholy do jiné komunity až bude množina zkoumaných vrcholů prázdná. Maximální změna účelové funkce může být i záporná. Když je množina zkoumaných vrcholů prázdná, algoritmus se vrátí do výchozího stavu a zopakuje tolik změn, při kterých bylo dosaženo maximální hodnoty účelové funkce. Příklad průběhu hodnoty účelové funkce je na obrázku 7.1. Volání KL-algoritmu se provádí, dokud změna mezi opakováními není pod nastavenám prahem nebo do nastaveného maximálního počtu opakování.

3.2 SBM a překrývající se komunity

Model *BKN-SBM* (*Ball-Karrer-Newman's stochastic block model*) [3] umožňuje modelovat grafy pomocí tzv. *překrývajících se komunit*. To znamená, že vrcholy nejsou pouze rozděleny do několika skupin, ale vrcholy mohou nabývat členství s různou silou v několika komunitách současně.

Celé vlastnosti modelu jsou popsány maticí $\boldsymbol{\theta}$ rozměru $n \times k$, kde k je počet komunit a n je počet vrcholů. Prvek $\boldsymbol{\theta}_{ir}$ představuje sílu členství vrcholu i v komunitě r. Každý vrchol tak může mít nenulové, různě významné členství hned v několika komunitách (skupinách). Pravděpodobnost hrany $\{i, j\}$ složená ze členství v komunitách dle parametrů v matici $\boldsymbol{\theta}$ je [3]:

$$p(i,j) = 1 - exp\left(-\sum_z \boldsymbol{\theta}_{iz}\boldsymbol{\theta}_{jz}\right). \tag{3.63}$$

Matice středních hodnot má tvar

$$\boldsymbol{\Lambda}_{ij} = (\boldsymbol{\theta}\boldsymbol{\theta}^T)_{ij}. \tag{3.64}$$

3.2.1 Podobnost se standardním SBM

Způsob jak můžeme model pro překrývající se komunity přiblížit ke *stochastickým blokovým modelům*, je spojit vektory $\vec{\theta}$ a $\vec{g}$ do matice $\boldsymbol{\theta}$ vztahem (4.7) uvedeným v kapitole 4.2. Po této úpravě SBM může platit

$$(\boldsymbol{\theta}\boldsymbol{\theta}^T)_{ij} = (\boldsymbol{\theta}\boldsymbol{I}\boldsymbol{\theta}^T)_{ij} = (\boldsymbol{\theta}\boldsymbol{\omega}\boldsymbol{\theta}^T)_{ij} = \theta_i\theta_j\boldsymbol{\omega}_{g_ig_j}, \tag{3.65}$$

kde $\boldsymbol{\omega} = \boldsymbol{I}$ (viz kap. 4.2). Matice $\boldsymbol{\omega}$ jako jednotková matice, má význam, že vrcholy uvnitř komunity jsou propojeny hustěji.

3.2.2 Detekce překrývajcích se komunit

Metoda detekce komunit v grafu $\mathcal{G}(\boldsymbol{A})$ popsaná v [3] provádí faktorizaci matice $\boldsymbol{A}$ na součin $\boldsymbol{\theta}\cdot\boldsymbol{\theta}^T$ vzhledem ke KL-divergenci $\mathrm{KL}(\boldsymbol{A}||\boldsymbol{\Lambda})$. Účelová funkce je odvozena z pravděpodobnosti, že zkoumaný graf $\mathcal{G}(\boldsymbol{A})$ by vzniknul náhodným generováním hran z tohoto modelu s parametry $\boldsymbol{\theta}$ [3]:

$$P(\mathcal{G}|\boldsymbol{\theta}) = \prod_{i<j}\frac{(\boldsymbol{\theta}\boldsymbol{\theta}^T)_{ij}^{\boldsymbol{A}_{ij}}}{\boldsymbol{A}_{ij}!}\exp(-(\boldsymbol{\theta}\boldsymbol{\theta}^T)_{ij})\cdot\prod_i\frac{(\frac{1}{2}(\boldsymbol{\theta}\boldsymbol{\theta}^T)_{ii})^{\boldsymbol{A}_{ii}/2}}{(\boldsymbol{A}_{ii}/2)!}\exp(-\frac{1}{2}(\boldsymbol{\theta}\boldsymbol{\theta}^T)_{ii}). \tag{3.66}$$

Aplikací Jensenovy nerovnosti (1.21) ve tvaru

$$\log\left(\sum_z x_z\right) \geq \sum_z q_z\log\frac{x_z}{q_z}, \tag{3.67}$$

je do modelu přidán vektor parametrů $\vec{q}_{ij}(\cdot)$ splňující normalizační podmínku

$$\sum_z q_{ij}(z) = 1. \tag{3.68}$$

Tento vektor lze interpretovat jako pravděpodobnost, že hrana $\{i,j\}$ ve zkoumaném grafu vznikla v důsledku členství vrcholů i a j ve společné komunitě z.

Pro jednoduchý neorientovaný graf $\mathcal{G}(\boldsymbol{A})$ zadaný maticí $\boldsymbol{A}$ odvodíme účelovou funkci z logaritmu (3.66)

$$\log P(\mathcal{G}|\boldsymbol{\theta}) = \frac{1}{2}\sum_{ij}\boldsymbol{A}_{ij}\log\left(\sum_z\theta_{iz}\theta_{jz}\right) - \frac{1}{2}\sum_{ijz}\theta_{iz}\theta_{jz}. \tag{3.69}$$

Suma v argumentu logaritmu znemožňuje snadno analyticky nalézt stacionární body. Proto dále využijeme nerovnost (3.67) pro každou hranu

$$\sum_{i<j}\boldsymbol{A}_{ij}\log\left(\sum_z\theta_{iz}\theta_{jz}\right) \geq \sum_{i<j}\boldsymbol{A}_{ij}\sum_z q_{ij}(z)\log\frac{\theta_{iz}\theta_{jz}}{q_{ij}(z)}. \tag{3.70}$$

Pro nalezení maxima účelové funkce hledáme maximum pravé strany nerovnosti a předpokládáme, že v tomto bodě bude mít maximum i levá strana. Pravá strana bude mít maximum v případě, že nastane rovnost, která podle (1.22) nastane při splnění

$$\forall\{i,j\}\in E: \quad q_{ij}(z) = \frac{\theta_{iz}\theta_{jz}}{\sum_z\theta_{iz}\theta_{jz}}. \tag{3.71}$$

S předpokladem (3.71) nalezneme stacionární body pravé strany analyticky diferencováním.

Věta 3.2.1. Pro jednoduchý neorientovaný graf $\mathcal{G}(\boldsymbol{A})$ zadaný maticí $\boldsymbol{A}$ má účelová funkce

$$F_{\boldsymbol{A}}(\boldsymbol{\theta}) = \frac{1}{2}\sum_{ij}\boldsymbol{A}_{ij}\sum_{z}q_{ij}(z)\log\frac{\theta_{iz}\theta_{jz}}{q_{ij}(z)} - \frac{1}{2}\sum_{ijz}\theta_{iz}\theta_{jz}. \tag{3.72}$$

vzhledem $\boldsymbol{\theta}$ maximum v bodě [3]

$$\theta_{iz} = \frac{\sum_j A_{ij}q_{ij}(z)}{\sqrt{\sum_{ij}A_{ij}q_{ij}(z)}} = \frac{\sum_j A_{ij}q_{ij}(z)}{\sqrt{\kappa_z}}. \tag{3.73}$$

Důkaz. Hledejme stacionární bod jako řešení

$$\frac{\partial}{\partial\theta_{iz}}\left[\frac{1}{2}\sum_{ij}\boldsymbol{A}_{ij}\sum_{z}q_{ij}(z)\log\frac{\theta_{iz}\theta_{jz}}{q_{ij}(z)} - \frac{1}{2}\sum_{ijz}\theta_{iz}\theta_{jz}\right] = 0 \tag{3.74}$$

$$\sum_{j}\boldsymbol{A}_{ij}\sum_{z}q_{ij}(z)\frac{1}{\theta_{iz}} - \sum_{j}\theta_{jz} = 0 \tag{3.75}$$

$$\theta_{iz} = \frac{\sum_j \boldsymbol{A}_{ij}\sum_z q_{ij}(z)}{\sum_j \theta_{jz}} = \frac{\sum_j \boldsymbol{A}_{ij}\sum_z q_{ij}(z)}{\sum_i \theta_{iz}} \tag{3.76}$$

Jmenovatele vyřešíme součtem θ_{iz} přes i

$$\sum_{i}\theta_{iz} = \frac{1}{\sum_i\theta_{iz}}\sum_{ij}\boldsymbol{A}_{ij}\sum_{z}q_{ij}(z), \tag{3.77}$$

$$\left(\sum_{i}\theta_{iz}\right)^2 = \sum_{ij}\boldsymbol{A}_{ij}\sum_{z}q_{ij}(z) = \kappa_z. \tag{3.78}$$

□

Algoritmus v každé iteraci nasčítává hodnoty matici $\boldsymbol{\theta}$ pro každou hranu. Inicializace algoritmu pro každou hranu $\{i,j\}$ vygeneruje náhodné pravděpodobnostní rozdělení $q_{ij}(z)$. Následně se iteruje přes hrany grafu $\mathcal{G}(\boldsymbol{A})$. Z těchto rozdělení vysčítá stupně komunit κ_z a členství vrcholů v komunitách matici $\boldsymbol{\theta}$:

$$\theta_{iz} = \frac{\sum_j A_{ij}q_{ij}(z)}{\sqrt{\kappa_z}}. \tag{3.79}$$

V dalších iterací se hodnoty $q_{ij}(z)$ neurčují náhodně, ale vypočtou se vztahem

$$q_{ij}(z) = \frac{\theta_{iz}\theta_{jz}}{\sum_z\theta_{iz}\theta_{jz}}. \tag{3.80}$$

Tento postup včetně latentní pravděpodobnosti je variací EM algoritmu (angl. expectation maximalization) [3]. Podrobněji je algoritmus vysvětlen v kapitole 5.2.

Kapitola 4

Popisující model

Motivační problém této práce je klasifikovat tituly egyptských vezírů a jejich nositele (viz obrázky 1 a 2). Tato data lze popsat prostřednictvím bipartitního grafu. Jeden typ vrcholů grafu představují tituly a druhý typ vrcholů jsou jejich nositelé. Zároveň předpokládáme, že podle existujících hran lze roztřídit vrcholy jak na straně titulů, tak lidí na straně druhé do skupin. Navíc tituly resp. lidi mohou patřit do více skupin titulů resp. lidí současně.

V rešeršní části jsou popsány komunity vrcholů uvnitř unipartitních grafů. Běžné definice komunity popisují jako skupiny vrcholů, které mají mezi sebou více hran, než se zbytkem grafu [31]. Klasifikaci vrcholů tímto způsobem lze zjemnit pomocí tzv. překrývajících se komunit. Připouštíme, že vrchol může být propojen do více skupin vrcholů.

Klasická definice komunit jako hustěji propojených skupin vrcholů představuje problém pro detekci překrývajících se komunit v bipartitních grafech [26, 36]. V bipartitním grafu neexistují hrany mezi vrcholy stejného typu a podle poznámky 3.1.5 nemůžou být propojeny ani komunity stejného typu. Pro bipartitní stochastický blokový model je potřeba komunity chápat jako skupinu vrcholů s podobnou šablonou propojení k ostatním skupinám [26].

V rámci výzkumného úkolu mým cílem bylo upravit bipartitní stochastický model popsaný v *bipartitní SBM* [26] po vzoru modelů *BigClam* [43] a *BKN-SBM* [3] tak, aby umožňoval sdílet komunity mezi vrcholy. Navrhnul jsem model schopný popsat překrývající se komunity na bipartitních grafech. V obecné podobě zároveň dokáže popsat i komunity na unipartitních grafech a i bez překryvu. Díky tomu lze tento model využít ke srovnání detekčních metod založených na SBM s Poissonovým rozdělením. Řešení je také přímo uvedeno v příloze článku [3] avšak v jiném kontextu. V této práci dále popisuji detekční metodu založenou na tomto modelu.

4.1 Model grafu

Model nad grafem $\mathcal{G}(V, E)$ tvoří dva ohodnocené grafy $\mathcal{B}(\boldsymbol{\theta})$ resp. $\mathcal{G}(\boldsymbol{\omega})$, popsané maticemi $\boldsymbol{\theta}$ resp. $\boldsymbol{\omega}$. První graf $\mathcal{B}(\boldsymbol{\theta})$ je ohodnocený bipartitní a spojuje vrcholy z původního grafu $\mathcal{G}(V, E)$ s komunitami v grafu $\mathcal{G}(\boldsymbol{\omega})$. Graf $\mathcal{G}(\boldsymbol{\omega})$ je také ohodnocený a jeho vrcholy jsou komunity. Řádky matice $\boldsymbol{\theta}$ reprezentují vrcholy a sloupce zase komunity. Nenulové prvky matice $\boldsymbol{\theta}$ vyjádřují členství vrcholů v komunitách. Hodnota prvku $\boldsymbol{\theta}_{ir}$ je úměrná stupni vrcholu a stupni komunity. Druhý graf $\mathcal{G}(\boldsymbol{\omega})$ je ohodnocený unipartitní a popisuje vzájemné propojení komunit. Střední hodnota pravděpodobnosti hrany $\{i, j\}$ je dána součtem ohodnocených cest z vrcholu i grafem $\mathcal{B}(\boldsymbol{\theta})$, dále skrz graf $\mathcal{G}(\boldsymbol{\omega})$ a grafem $\mathcal{B}(\boldsymbol{\theta})$ do vrcholu j. Podle věty 1.1.13 vzorcem lze tento součet

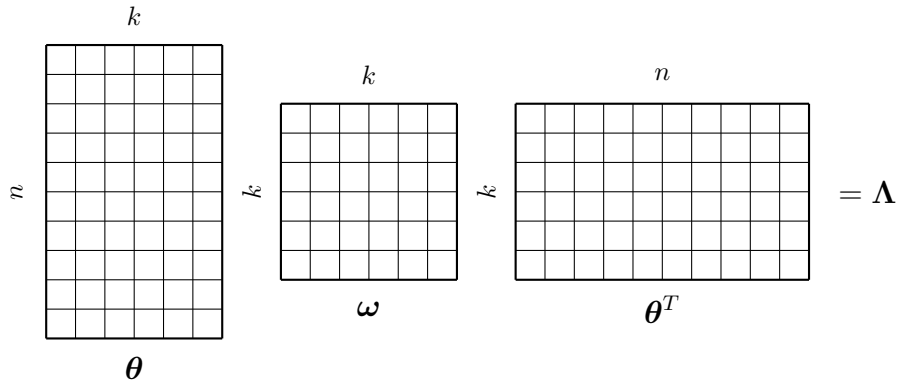
cest vyjádřit:

$$\Lambda_{ij} = \left(\boldsymbol{\theta\omega\theta}^T\right)_{ij}. \tag{4.1}$$

Pravděpodobnost existence hrany $\{i, j\}$ s Poissonovo rozdělením je v důsledku:

$$p(i,j) = 1 - \exp\left(-\left(\boldsymbol{\theta\omega\theta}^T\right)_{ij}\right). \tag{4.2}$$

Princip grafu $\mathcal{G}$ jako důsledku modelu sestaveného ze dvou grafů je zobrazen na obrázku 4.2.



Obrázek 4.1: Schéma popisujícího modelu. Počet vrcholů je n a počet komunit je k. Model je popsán pomocí dvou matic. Matice $\boldsymbol{\theta}$ rozměru $k \times n$ přiřazuje vrcholům členství v komunitách. Řádky reprezentují vrcholy a sloupce zase komunity. Symetrická matice $\boldsymbol{\omega}$ rozměru $k \times k$ určuje, jak se komunity propojují mezi sebou. Součin $\boldsymbol{\theta\omega\theta}^T$ dává matici středních hodnot počtu hran $\boldsymbol{\Lambda}$.

Aby vzniknul bipartitní graf, musí být vrcholy dvou parit a nesmí existovat hrana mezi vrcholy se stejnou paritou. Z tohoto důvodu jsou komunity také dvou parit (viz poznámka 3.1.5). Žádný vrchol nesmí nabývat ani částečného členství v komunitě s druhou paritou a pravděpodobnost hrany mezi komunitami stejné parity je nulová. Potom lze permutacemi řádků a sloupců přepsat matice $\boldsymbol{\theta}$ a $\boldsymbol{\omega}$ do blokového tvaru

$$\boldsymbol{\omega} = \begin{pmatrix} \mathbf{0} & \boldsymbol{C} \\ \boldsymbol{C}^T & \mathbf{0} \end{pmatrix}, \quad \boldsymbol{\theta} = \begin{pmatrix} \boldsymbol{A} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{B} \end{pmatrix}, \tag{4.3}$$

kde matice $\boldsymbol{A}$ a $\boldsymbol{B}$ obsahují členství uzlů jednoho a druhého typu, zatímco matice $\boldsymbol{C}$ popisuje bibartitní vztahy mezi komunitami.

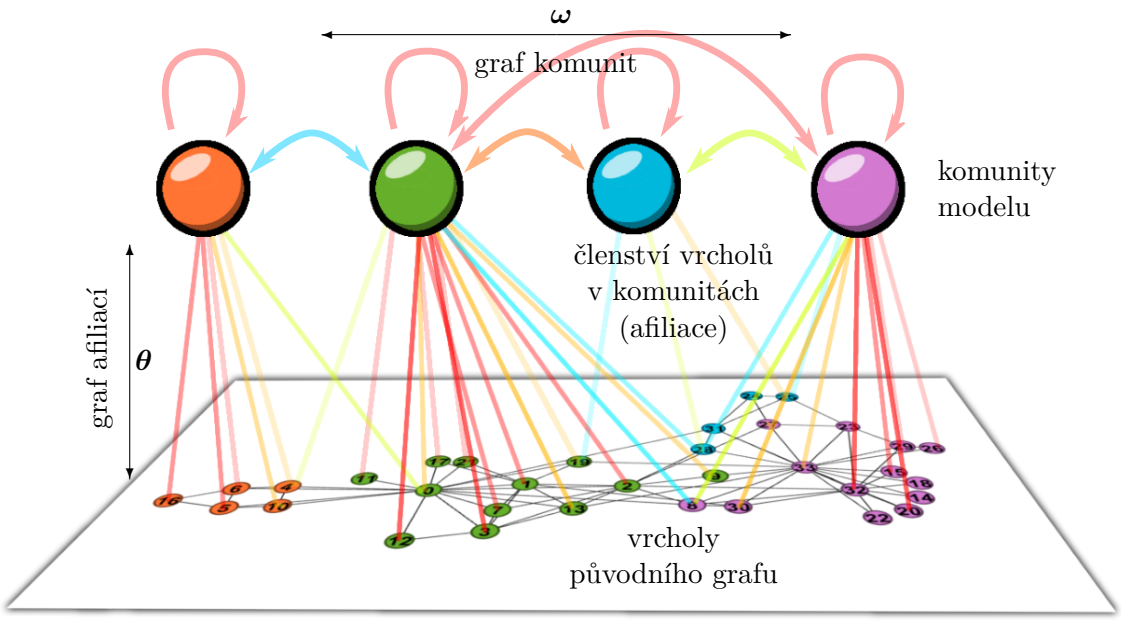
V případě modelu bipartitního grafu, pro vyjádření pravděpodobnosti hrany mezi vrcholy i a j různých parit stačí součin trojice matic $\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}$:

$$p(i,j) = 1 - \exp\left(\left(\boldsymbol{ACB}^T\right)_{ij}\right), \tag{4.4}$$

neboť

$$\boldsymbol{\theta\omega\theta}^T = \begin{pmatrix} \boldsymbol{A} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{B} \end{pmatrix} \begin{pmatrix} \mathbf{0} & \boldsymbol{C} \\ \boldsymbol{C}^T & \mathbf{0} \end{pmatrix} \begin{pmatrix} \boldsymbol{A}^T & \mathbf{0} \\ \mathbf{0} & \boldsymbol{B}^T \end{pmatrix} = \begin{pmatrix} \mathbf{0} & \boldsymbol{ACB}^T \\ \boldsymbol{A}^T\boldsymbol{C}^T\boldsymbol{B} & \mathbf{0} \end{pmatrix}. \tag{4.5}$$

Takto navržený model umožňuje popsat strukturu grafu, ve kterém jsou vrcholy propojeny podle členství v komunitách. Navíc lze určit, jak jsou komunity propojeny mezi sebou a díky tomu lze vytvořit bipartitní graf. Zároveň mohou vrcholy mohou patřit do více komunit najednou a také lze vynutit rozložení stupňů vrcholů.



Obrázek 4.2: Obrázek ilustruje výstupní graf jako důsledek modelu, který je tvořen dvěma grafy. V horní části je znázorněn graf komunit s maticí sousednosti $\boldsymbol{\omega}$. V prostřední části je naznačen graf afiliací s incidenční maticí $\boldsymbol{\theta}$. Ve spodní části se nachází původní graf, jehož struktura je popsána modelem z $\boldsymbol{\omega}$ a $\boldsymbol{\theta}$. V případě běžného modelu [3, 43] překrývajících se komunit v unipartitním grafu, graf komunit obsahuje pouze smyčky, neboť $\boldsymbol{\omega} = \boldsymbol{I}$.

4.2 Srovnání s jinými modely

Tento model lze zařadit do skupiny stochastických generujících modelů (SBM). Standardní model SBM je popsán maticí $\boldsymbol{\omega}$ a zobrazením/vektorem g přiřazujícím vrcholům jejich komunitu. Prvky matice $\boldsymbol{\omega}_{rs}$ popisují pravděpodobnost hrany mezi vrcholy z komunit r a s. Při volbě

$$\boldsymbol{\theta}_{ir} = \delta(r, g_i), \tag{4.6}$$

se navrhovaný model zredukuje na *standardní SBM* (kap. 3.1).

V grafech popisujících vazby ze skutečného světa jsou stupně vrcholů různorodé. Z tohoto důvodu je potřeba modelovat i stupně vrcholů a na výsledky klasifikace mají podstatný vliv [24]. M. E. J. Newman v [24] zavádí *SBM s kontrolou stupňů*, který tuto schopnost má. Ten má navíc vektor θ, který udává vrcholům „sílu“ a tím jejich stupeň. Vektor $\vec{\theta}$ splňuje normalizační podmínku (3.27). Tedy součet prvků θ_i pro všechny vrcholy v každé komunitě r je normovaný na 1. Při volbě

$$\boldsymbol{\theta}_{ir} = \vec{\theta}_i \delta(r, g_i), \tag{4.7}$$

se předložený model zredukuje na *SBM s kontrolou stupňů* (kap. 3.1.1). Matice $\boldsymbol{\theta}$ vznikne z vektoru $\vec{\theta}$ rozdělením jeho hodnot do sloupců podle skupin definovaných vektorem $\vec{g}$.

Pro detekci překrývající se komunity jsem v rešeršní části uvedl *BKN-SBM* [3]. Pro popis parametrů modelu stačí matice členství $\boldsymbol{\theta}$ (viz obrázek 2.3). V tomto odstavci pro odlišení budu značit matici členství BKN-SBM jako $\boldsymbol{F}$. BKN-SBM není úplně běžný SBM model. Implicitně

vnímá komunity jako hustěji propojené skupiny vrcholů, což odpovídá tomu, že jejich graf komunit je tvořen pouze smyčkami, viz kapitola 3.2. Pravděpodobnost, že graf popsaný tímto modelem bude mít hranu $\{i, j\}$ je dána

$$p(i,j) = 1 - exp\left(-(\boldsymbol{F}\boldsymbol{F}^T)_{ij}\right). \tag{4.8}$$

Tímto je velice podobný navrhovanému modelu vztahem

$$\boldsymbol{F}\boldsymbol{F}^T = \boldsymbol{F}\boldsymbol{I}\boldsymbol{F}^T = \boldsymbol{\theta}\boldsymbol{\omega}\boldsymbol{\theta}^T, \tag{4.9}$$

kde $\boldsymbol{\theta} = \boldsymbol{F}$ a $\boldsymbol{\omega} = \boldsymbol{I}$. Rovnost $\boldsymbol{\omega} = \boldsymbol{I}$ přesně vystihuje klasické chápání komunit, jako skupin vrcholů propojených hustěji mezi sebou, než se zbytkem grafu, viz kapitola 3.2.

4.3 Možnosti modelu

Na základě srovnání (kap. 4.2) s jinými modely je zřejmé, že navržený model umožňuje popsat

- unipartitní graf s komunitami bez překryvu,
- unipartitní graf s překrývajícími se komunitami,
- bipartitní graf s komunitami bez překryvu a
- bipartitní graf s překrývajícími se komunitami.

Zároveň je možné ovlivňovat velikost stupňů vrcholů. Podobně jako v [3, 15, 43] je hrana důsledekem síly členství vrcholu v komunitě. Tedy i stupně vrcholů jsou důsledky síly členství.

4.4 Princip polohran

V kapitole 2.3.2 je vysvětleno, že členství vrcholu ve více komunitách je dáno stupněm a sousedními vrcholy. Ve skutečnosti do komunit nepatří vrcholy, ale hrany. Každé hraně je přiřazena komunita. Komunita v modelech znamená skrytý princip vzniku hrany mezi danými vrcholy. Mnohonásobné členství vrcholu ve více komunitách je důsledkem komunit hran vycházejících z tohoto vrcholu.

Mnohonásobné členství vrcholů ve více komunitách současně lze vysvětlit principem polohran. Vrcholy jsou pouze „místo“ setkání konců hran (polohran). Každý konec hrany přísluší nějaké komunitě. O vrcholu tvrdíme, že má členství v komunitách polohran, které jsou definovány vrcholem. Sdílené členství vrcholu v komunitách je důsledkem komunit konců hran setkávajících se ve vrcholu. Síla členství vrcholu v jednotlivých komunitách pak odpovídá poměru komunit polohran.

Z principu polohran vychází, že parametry $q_{ij}(r, s)$ v modelu jsou pravděpodobnost. Jsou svázány normovací podínkou

$$\sum_{rs} q_{ij}(r,s) = 1. \tag{4.10}$$

Ve výstupu je polohraně určena právě nejpravděpodobnější komunita konce hrany. Pravděpodobnost $q_{ij}(r, s)$ slouží jako aproximace pro účel výpočtu. Z toho přirozeně plyne prahování matice $\boldsymbol{\theta}$ a fakt, že vrchol může sdílet členství v tolik komunitách, jaký má stupeň.

Kapitola 5

Detekce

5.1 Účelová funkce

Detekce se provádí maximalizací účelové funkce

$$F_{\mathcal{G}(\boldsymbol{A})}\left(q,\boldsymbol{\theta},\boldsymbol{\omega}\right)=\frac{1}{2}\sum_{ijrs}\left(\boldsymbol{A}_{ij}q_{ij}(r,s)\log\frac{\boldsymbol{\theta}_{ir}\boldsymbol{\omega}_{rs}\boldsymbol{\theta}_{jz}}{q_{ij}(r,s)}-\boldsymbol{\theta}_{ir}\boldsymbol{\omega}_{rs}\boldsymbol{\theta}_{js}\right). \tag{5.1}$$

Účelová funkce je za použití Jensenovy nerovnosti (1.21) odvozena z věrohodnosti

$$\mathcal{L}_{\mathcal{G}(\boldsymbol{A})}\left(\boldsymbol{\theta},\boldsymbol{\omega}\right)=\frac{1}{2}\sum_{ij}\boldsymbol{A}_{ij}\log\left(\sum_{rs}\boldsymbol{\theta}_{ir}\boldsymbol{\omega}_{rs}\boldsymbol{\theta}_{js}\right)-\frac{1}{2}\sum_{ijrs}\boldsymbol{\theta}_{ir}\boldsymbol{\omega}_{rs}\boldsymbol{\theta}_{js}\geq F_{\mathcal{G}(\boldsymbol{A})}\left(q,\boldsymbol{\theta},\boldsymbol{\omega}\right). \tag{5.2}$$

Věrohodnostní funkce vznikne zanedbáním členů, které nemají vliv na polohu maxima, z logaritmu pravděpodobnosti

$$P\left(\mathcal{G}(\boldsymbol{A})|\boldsymbol{\theta},\boldsymbol{\omega}\right)=\prod_{i<j}\frac{\left(\sum_{rs}\boldsymbol{\theta}_{ir}\boldsymbol{\omega}_{rs}\boldsymbol{\theta}_{js}\right)^{A_{ij}}}{A_{ij}!}\exp\left(-\sum_{rs}\boldsymbol{\theta}_{ir}\boldsymbol{\omega}_{rs}\boldsymbol{\theta}_{js}\right), \tag{5.3}$$

která popisuje pravděpodobnost, že graf $\mathcal{G}$ vygenerovaný modelem s parametry $\boldsymbol{\theta}$ a $\boldsymbol{\omega}$ bude odpovídat grafu $\mathcal{G}(\boldsymbol{A})$. Jinými slovy se jedná o pravděpodobnost, že parametry $\boldsymbol{\theta}$ a $\boldsymbol{\omega}$ popisují graf $\mathcal{G}(\boldsymbol{A})$.

Poznámka 5.1.1.

- Pro jednoduchost a přehlednost nepřipouštíme v grafu $\mathcal{G}(\boldsymbol{A})$ smyčky.
- Člen $\sum_{ij}\boldsymbol{A}_{ij}$ v účelové funkci (5.1) a věrohodnosti (5.2) plní funkci selektoru hran grafu $\mathcal{G}(\boldsymbol{A})$.
- Účelová funkce (5.1) a věrohodnost (5.2) jsou svázány nerovností. Můžeme předpokládat, že když nalezneme lokální maximum účelové funkce, pak v tomto bodě bude mít lokální maximum i věrohodnost.

Poznámka 5.1.2.

$$\begin{aligned}\sum_{ijrs} \boldsymbol{\theta}_{ir}\boldsymbol{\omega}_{rs}\boldsymbol{\theta}_{js} &= \sum_{r} \underbrace{\sum_{i} \boldsymbol{\theta}_{ir}}_{=1} \sum_{s} \boldsymbol{\omega}_{rs} \underbrace{\sum_{j} \theta_{js}}_{=1} = \sum_{rs} \boldsymbol{\omega}_{rs} = \\ &= \sum_{rs}\sum_{ij} \boldsymbol{A}_{ij} q_{ij}(r,s) = \sum_{ij} \boldsymbol{A}_{ij} \underbrace{\sum_{rs} q_{ij}(r,s)}_{=1} = 2|E|\end{aligned} \tag{5.4}$$

Poznámka 5.1.3. Kullbackova-Leilblerova divergence:

$$\mathrm{KL}(P||Q) = -\sum_{x\in\mathcal{X}} P(x)\log\frac{Q(x)}{P(x)}. \tag{5.5}$$

$$F(q,\boldsymbol{\theta},\boldsymbol{\omega}) = -\,\mathrm{KL}(q||P) + |E|\Big(\log(2|E|) - 2\Big), \tag{5.6}$$

kde $P_{ij}(r,s) = \boldsymbol{\theta}_{ir}\boldsymbol{\omega}_{rs}\boldsymbol{\theta}_{js} / \left(\sum_{ijrs} \boldsymbol{\theta}_{ir}\boldsymbol{\omega}_{rs}\boldsymbol{\theta}_{js}\right)$.

$$\begin{aligned}2F(q,\boldsymbol{\theta},\boldsymbol{\omega}) &= \sum_{ers} q_e(r,s)\log\frac{\frac{\boldsymbol{\theta}_{ir}\boldsymbol{\omega}_{rs}\boldsymbol{\theta}_{js}}{\sum_{ijrs}\boldsymbol{\theta}_{ir}\boldsymbol{\omega}_{rs}\boldsymbol{\theta}_{js}}}{q_e(r,s)} + |E|\Big(\log(2|E|) - 2\Big) = \\ &= \sum_{ers} q_e(r,s)\left(\log\frac{\boldsymbol{\theta}_{ir}\boldsymbol{\omega}_{rs}\boldsymbol{\theta}_{js}}{q_e(r,s)} - \log\sum_{ijrs}\boldsymbol{\theta}_{ir}\boldsymbol{\omega}_{rs}\boldsymbol{\theta}_{js}\right) + |E|\Big(\log(2|E|) - 2\Big) = \\ &= \sum_{ers} q_e(r,s)\left(\log\frac{\boldsymbol{\theta}_{ir}\boldsymbol{\omega}_{rs}\boldsymbol{\theta}_{js}}{q_e(r,s)}\right) - \sum_{e}\underbrace{\sum_{rs} q_e(r,s)}_{=1}\log(2|E|) + |E|\Big(\log(2|E|) - 2\Big) = \\ &= \sum_{ers} q_e(r,s)\left(\log\frac{\boldsymbol{\theta}_{ir}\boldsymbol{\omega}_{rs}\boldsymbol{\theta}_{js}}{q_e(r,s)}\right) - 2|E|\end{aligned} \tag{5.7}$$

5.2 Hledání maxima

Argument maxima účelové funkce (5.1) hledáme pomocí algoritmu expectation-maximization algoritmu (dále EM) [3, 10]. Postupujeme úplně analogickým způsobem jako v článku [3], avšak s ohledem na přidanou matici $\boldsymbol{\omega}$ a dvojici komunit r, s. Použitím Jensenovy nerovnosti v (5.2) nám do modelu přibude latentní pravděpodobnost $q_{ij}(r,s)$.

EM algoritmus řeší situace, kdy je obtížné maximalizovat funkci. Namísto hledání maxima funkce se hledá maximum dolního odhadu. Dolní odhad hledané věrohodnosti (5.2) je účelová funkce (5.1). Správnou volbu $\boldsymbol{q}_{ij}$ nastává rovnost pro konkrétní hodnoty $\boldsymbol{\theta}_{ir}$ a $\boldsymbol{\omega}_{rs}$.

V tzv. E-kroku nalzeneme $\boldsymbol{q}_{ij}$, aby platila rovnost v (5.2). Pro dané hodnoty $\boldsymbol{\theta}_{ir}$ a $\boldsymbol{\omega}_{rs}$ nastává rovnost s volbou

$$q_{ij}(r,s) = \frac{\boldsymbol{\theta}_{ir}\boldsymbol{\omega}_{rs}\boldsymbol{\theta}_{js}}{\sum_{rs}\boldsymbol{\theta}_{ir}\boldsymbol{\omega}_{rs}\boldsymbol{\theta}_{js}}. \tag{5.8}$$

S touto volbou můžeme $q_{ij}(r,s)$ interpretovat jako pravděpodobnost, že hrana $\{i,j\}$ má konce v komunitách r a s.

V tzv. M-kroku maximalizujeme dolní odhad (5.1) pro konkrétní $\boldsymbol{q}_{ij}$. Hledání extrému (5.1) s $\boldsymbol{q}_{ij}$ jako parametry se nám zjednoduší na analytické řešení stacionárních bodů pomocí derivování. Pro daná optimální $q_{ij}(r,s)$ můžu derivováním (5.1) získat nejlepší odhady parametrů $\boldsymbol{\theta}_{ir}$ a $\boldsymbol{\omega}_{rs}$.

$$\frac{\partial}{\partial \boldsymbol{\theta}_{ir}} \left[\sum_{ijrs} \left(\boldsymbol{A}_{ij} q_{ij}(r,s) \log \frac{\boldsymbol{\theta}_{ir}\boldsymbol{\omega}_{rs}\boldsymbol{\theta}_{jz}}{q_{ij}(r,s)} \right) - \sum_{ijrs} \boldsymbol{\theta}_{ir}\boldsymbol{\omega}_{rs}\boldsymbol{\theta}_{js} \right] = 0 \tag{5.9}$$

$$\sum_{js} \left(\boldsymbol{A}_{ij} q_{ij}(r,s) \frac{1}{\boldsymbol{\theta}_{ir}} \right) - \sum_{js} \boldsymbol{\omega}_{rs}\boldsymbol{\theta}_{js} = 0 \tag{5.10}$$

$$\sum_{js} \left(\boldsymbol{A}_{ij} q_{ij}(r,s) \frac{1}{\boldsymbol{\theta}_{ir}} \right) = \sum_{js} \boldsymbol{\omega}_{rs}\boldsymbol{\theta}_{js} \tag{5.11}$$

$$\sum_{js} \boldsymbol{A}_{ij} q_{ij}(r,s) = \boldsymbol{\theta}_{ir} \sum_{js} \boldsymbol{\omega}_{rs}\boldsymbol{\theta}_{js} \tag{5.12}$$

$$\boldsymbol{\theta}_{ir} = \frac{\sum_{js} \boldsymbol{A}_{ij} q_{ij}(r,s)}{\sum_{js} \boldsymbol{\omega}_{rs}\boldsymbol{\theta}_{js}} \tag{5.13}$$

$$\frac{\partial}{\partial \boldsymbol{\omega}_{rs}} \left[\sum_{ijrs} \left(\boldsymbol{A}_{ij} q_{ij}(r,s) \log \frac{\boldsymbol{\theta}_{ir}\boldsymbol{\omega}_{rs}\boldsymbol{\theta}_{jz}}{q_{ij}(r,s)} \right) - \sum_{ijrs} \boldsymbol{\theta}_{ir}\boldsymbol{\omega}_{rs}\boldsymbol{\theta}_{js} \right] = 0 \tag{5.14}$$

$$\sum_{ij} \left(\boldsymbol{A}_{ij} q_{ij}(r,s) \frac{1}{\boldsymbol{\omega}_{rs}} \right) - \sum_{ij} \boldsymbol{\theta}_{ir}\boldsymbol{\theta}_{js} = 0 \tag{5.15}$$

$$\boldsymbol{\omega}_{rs} = \frac{\sum_{ij} \boldsymbol{A}_{ij} q_{ij}(r,s)}{\sum_{ij} \boldsymbol{\theta}_{ir}\boldsymbol{\theta}_{js}} \tag{5.16}$$

Ze vztahů (5.10) a (5.15) je zřejmé, že druhé derivace budou nekladné.

Řešením soustavy rovnic jsou stacionární body popsané rovnicemi

$$\frac{\boldsymbol{\theta}_{ir}}{\sum_i \boldsymbol{\theta}_{ir}} = \frac{\sum_{js} \boldsymbol{A}_{ij} q_{ij}(r,s)}{\sum_{ijs} \boldsymbol{A}_{ij} q_{ij}(r,s)}, \quad \boldsymbol{\omega}_{rs} = \frac{\sum_{ij} \boldsymbol{A}_{ij} q_{ij}(r,s)}{\sum_i \boldsymbol{\theta}_{ir} \sum_j \boldsymbol{\theta}_{js}}, \tag{5.17}$$

ve kterých pro daná $q_{ij}(r,s)$ účelová funkce může nabývat lokálního maxima. Přidáním podmínky $\sum_i \boldsymbol{\theta}_{ir} = 1$ dostaneme jedinné řešení

$$\boldsymbol{\theta}_{ir} = \frac{\sum_{js} \boldsymbol{A}_{ij} q_{ij}(r,s)}{\sum_{ijs} \boldsymbol{A}_{ij} q_{ij}(r,s)}, \quad \boldsymbol{\omega}_{rs} = \sum_{ij} \boldsymbol{A}_{ij} q_{ij}(r,s). \tag{5.18}$$

Dodatečná podmínka dává parametrům $\boldsymbol{\theta}_{ir}$ význam pravděpodobnosti, že polohrana komunity r náleží vrcholu i.

Pro odvození nejlepších odhadů parametrů $\boldsymbol{\theta}_{ir}$ a $\boldsymbol{\omega}_{rs}$ jsme předpokládali znalost $q_{ij}(r,s)$. Pro výpočet $q_{ij}(r,s)$ vyžadujeme znalost $\boldsymbol{\theta}_{ir}$ a $\boldsymbol{\omega}_{rs}$. Iterujeme přes rovnice

$$q_{ij}(r,s) = \frac{\boldsymbol{\theta}_{ir}\boldsymbol{\omega}_{rs}\boldsymbol{\theta}_{js}}{\sum_{rs} \boldsymbol{\theta}_{ir}\boldsymbol{\omega}_{rs}\boldsymbol{\theta}_{js}}, \quad \boldsymbol{\theta}_{ir} = \frac{\sum_{js} \boldsymbol{A}_{ij} q_{ij}(r,s)}{\sum_{ijs} \boldsymbol{A}_{ij} q_{ij}(r,s)}, \quad \boldsymbol{\omega}_{rs} = \sum_{ij} \boldsymbol{A}_{ij} q_{ij}(r,s). \tag{5.19}$$

Úplně počáteční krok vyřešíme náhodnou inicializací $q_{ij}(r,s)$. Konvergence je zaručena vlastností EM algoritmu, že monotóně vylepšuje hodnotu (5.1).

Věta 5.2.1. Pro EM algoritmus popsaný (5.19) platí

$$\mathcal{L}_{\mathcal{G}(\boldsymbol{A})}\left(\boldsymbol{\theta}^{(k+1)}, \boldsymbol{\omega}^{(k+1)}\right) \geq \mathcal{L}_{\mathcal{G}(\boldsymbol{A})}\left(\boldsymbol{\theta}^{(k)}, \boldsymbol{\omega}^{(k)}\right). \tag{5.20}$$

Důkaz. [10]

$$\begin{aligned} \mathcal{L}_{\mathcal{G}(\boldsymbol{A})}\left(\boldsymbol{\theta}^{(k+1)}, \boldsymbol{\omega}^{(k+1)}\right) &\overset{5.2}{\geq} F_{\mathcal{G}(\boldsymbol{A})}\left(q^{(k)}, \boldsymbol{\theta}^{(k+1)}, \boldsymbol{\omega}^{(k+1)}\right) \\ &\geq F_{\mathcal{G}(\boldsymbol{A})}\left(q^{(k)}, \boldsymbol{\theta}^{(k)}, \boldsymbol{\omega}^{(k)}\right) = \mathcal{L}_{\mathcal{G}(\boldsymbol{A})}\left(\boldsymbol{\theta}^{(k)}, \boldsymbol{\omega}^{(k)}\right). \end{aligned} \tag{5.21}$$

První nerovnost je dána Jensenovou nerovností. Druhá nerovnost je dána M-krokem a nalezením maxima dolního odhadu. Poslední rovnost je dána E-krokem. □

5.3 Detekce gradientní metodou

Z kapitol 3.2.2 a 5.2 je zřejmé, že z věrohodnosti (5.2) nelze snadno analyticky odvodit stacionární body jako v kapitole 3.1. Avšak můžeme postupovat analogicky s článkem [43] a určit gradient pro konkrétní stav $\boldsymbol{\omega}$ a $\boldsymbol{\theta}$:

$$\frac{\partial}{\partial\theta}\mathcal{L}_{\mathcal{G}}(\boldsymbol{\theta}, \boldsymbol{\omega}) = \boldsymbol{B}\boldsymbol{\theta}\boldsymbol{\omega}, \tag{5.22}$$

$$\frac{\partial}{\partial\omega}\mathcal{L}_{\mathcal{G}}(\boldsymbol{\theta}, \boldsymbol{\omega}) = \boldsymbol{\theta}^T\boldsymbol{B}\boldsymbol{\theta}. \tag{5.23}$$

Gradientní matice $\boldsymbol{B}$ závisí na konkrétní volbě věrohodnosti. V případě (5.3) má gradientní matice $\boldsymbol{B}$ tvar

$$\boldsymbol{B}_{ij} = \frac{\boldsymbol{A}_{ij}}{(\boldsymbol{\theta}\boldsymbol{\omega}\boldsymbol{\theta})_{ij}} - 1. \tag{5.24}$$

V případě věrohodnosti

$$\mathcal{L}_{\mathcal{G}(\boldsymbol{A})} = \sum_{ij} A_{ij} \log\left(1 - \exp\left(-(\boldsymbol{\theta}\boldsymbol{\omega}\boldsymbol{\theta})_{ij}\right)\right) + \sum_{ij}(A_{ij} - 1)(\boldsymbol{\theta}\boldsymbol{\omega}\boldsymbol{\theta})_{ij} \tag{5.25}$$

inspirované z [43] je gradientní matice

$$\boldsymbol{B}_{ij} = \frac{A_{ij}}{1 - \exp(-(\boldsymbol{\theta}\boldsymbol{\omega}\boldsymbol{\theta})_{ij})} - 1. \tag{5.26}$$

Po určení gradientu metoda [43] využívá k nalezení maxima *backtracking line search* [7]. Dále je potřeba omezit se na nezáporné hodnoty

$$\boldsymbol{\theta}_{ir} \leftarrow \max(0, \boldsymbol{\theta}_{ir}), \quad \boldsymbol{\omega}_{rs} \leftarrow \max(0, \boldsymbol{\omega}_{rs}). \tag{5.27}$$

Při srovnání s rozkladem nezáporných matic (NMF) [21] je tento postup aditivní, zatímco postup v kapitole 5.2 je multiplikativní.

Poznámka 5.3.1. Pokud dosadíme do vzorců M kroku EM algoritmu (5.18) vzorec E kroku (5.8), získáme iterační vztahy ve tvaru

$$\boldsymbol{\theta}_{ir} \leftarrow \frac{(\boldsymbol{C}\boldsymbol{\theta}\boldsymbol{\omega})_{ir}\boldsymbol{\theta}_{ir}}{\sum_i (\boldsymbol{C}\boldsymbol{\theta}\boldsymbol{\omega})_{ir}\boldsymbol{\theta}_{ir}}, \quad \boldsymbol{\omega}_{rs} \leftarrow (\boldsymbol{\theta}^T\boldsymbol{C}\boldsymbol{\theta})_{rs}\boldsymbol{\omega}_{rs}, \tag{5.28}$$

kde matice

$$\boldsymbol{C}_{ij} = \frac{\boldsymbol{A}_{ij}}{(\boldsymbol{\theta}\boldsymbol{\omega}\boldsymbol{\theta})_{ij}} = \boldsymbol{B}_{ij} + 1. \tag{5.29}$$

Tyto vztahy připomínají krok Newtonovy gradientní metody.

Kapitola 6

Implementace

6.1 Proměnné a optimalizace

Narozdíl od popisu modelu implementace využívá optimalizační zkratky, převzaté z implementace podle článku [3]. Zásadní rozdíly oproti uvedeným vztahům v (5.19) je využívání počtu hran oproti poměrům.

Proměnná `expected_vertex_comm_degree`:

$$\boldsymbol{k}_{ir} = \sum_{js} A_{ij} q_{ij}(r,s) \tag{6.1}$$

popisuje počet konců hran komunity r ve vrcholu i. Je uložena jako matice rozměru $n \times k$. Proměnná `expected_comm_comm_degree`:

$$\boldsymbol{\omega}_{rs} = \sum_{ij} A_{ij} q_{ij}(r,s) \tag{6.2}$$

je matice rozměru $k \times k$ a má význam počtu hran s jedním koncem v komunitě r a druhým v s. Proměnná `expected_comm_degree`:

$$\kappa_r = \sum_s \boldsymbol{\omega}_{rs} = \sum_s \sum_{ij} A_{ij} q_{ij}(r,s) \tag{6.3}$$

je vektorem velikosti k a ukládá stupně komunity. Proměnná `reduced_comm_comm_degree`:

$$\tilde{\boldsymbol{\omega}}_{rs} = \frac{\boldsymbol{\omega}_{rs}}{\kappa_r \kappa_s} \tag{6.4}$$

slouží k výpočtu pravděpodobnostního rozložení $q_{ij}(r,s)$ komunit konců hrany $\{i,j\}$ vztahem

$$q_{ij}(r,s) = \frac{\boldsymbol{\theta}_{ir}\boldsymbol{\omega}_{rs}\boldsymbol{\theta}_{js}}{(\boldsymbol{\theta}\boldsymbol{\omega}\boldsymbol{\theta}^T)_{ij}} = \frac{\boldsymbol{k}_{ir}\tilde{\boldsymbol{\omega}}_{rs}\boldsymbol{k}_{js}}{(\boldsymbol{k}\tilde{\boldsymbol{\omega}}\boldsymbol{k}^T)_{ij}}. \tag{6.5}$$

Pravděpodobnostního rozložení $q_{ij}(r,s)$ komunit konců hrany $\{i,j\}$, proměnná `edge_probability` rozměru $k \times k$, se napočítává teprve v iteraci pro každou hranu. Tím se optimalizují paměťové nároky.

6.2 Inicializace

Inicializace spočívá ve vygenerování náhodné matice $\boldsymbol{q}_{ij}$ v proměnné `edge_probability` ke každé hraně. Tato pravděpodobnost komunity konců hrany se připočte k agregovaným stupňům $\boldsymbol{k}_{ir}$ (`expected_vertex_comm_degree`) a $\boldsymbol{\omega}_{rs}$ (`expected_comm_comm_degree`):

$$\begin{aligned} \boldsymbol{k}_{i\cdot} &\leftarrow \boldsymbol{k}_{i\cdot} + \sum_s \boldsymbol{q}_{ij}(\cdot, s)^T, \\ \boldsymbol{k}_{j\cdot} &\leftarrow \boldsymbol{k}_{j\cdot} + \sum_r \boldsymbol{q}_{ij}(r, \cdot), \\ \boldsymbol{\omega} &\leftarrow \boldsymbol{\omega} + \boldsymbol{q}_{ij}(\cdot, \cdot)^T + \boldsymbol{q}_{ij}(\cdot, \cdot). \end{aligned} \tag{6.6}$$

Inicializaci lze provést jemnou nebo hrubou. Jemná inicializace spočívá v generování $\boldsymbol{q}_{ij}(r, s)$ jako matice $k \times k$ náhodných čísel, která je normována $\sum_{rs} \boldsymbol{q}_{ij}(r, s) = 1$. Při hrubé inicializaci je každému konci hrany náhodně zvolena komunita. Například hrana $\{i, j\}$ bude mít jeden konec v r a druhý v s a náhodná matice se volí jako $\boldsymbol{q}_{ij}(t, u) = \delta_{rt}\delta_{su}$. Dopady inicializace na průběh detekce komunit jsou v sekci 7.2.

V případě detekce komunit v bipartitním grafu je požadavek na zachovávání stejné parity komunity a jejích členů. Tento požadavek je uspokojen v případě hrubé inicializace omezením výběru komunit polohran. V případě jemné inicializace se před normováním nulují řádky nebo sloupce komunit opačné parity:

```
def __get_random_matrix_soft__(self, u, v, node_types) -> np.ndarray:
    matrix = np.random.rand(self.communities_number, self.communities_number)
    if self.is_bipartite:
        if node_types[u] == node_types[0]:
            matrix[self.communities_number_first:, :] = 0
        else:
            matrix[:self.communities_number_first, :] = 0
        if node_types[v] == node_types[0]:
            matrix[:, self.communities_number_first:] = 0
        else:
            matrix[:, :self.communities_number_first] = 0
    matrix /= np.sum(matrix)
    return matrix
```

Ke konci inicializace se z $\boldsymbol{\omega}_{rs}$ připraví stupně komunit κ_r (`expected_comm_degree`) a redukovaná matice grafu komunit $\tilde{\boldsymbol{\omega}}_{rs}$ (`reduced_comm_comm_degree`), které jsou potřeba pro výpočty v iteraci:

$$\begin{aligned} \kappa_r &\leftarrow \sum_s \boldsymbol{\omega}_{\cdot s}, \\ \tilde{\boldsymbol{\omega}}_{rs} &\leftarrow \frac{\boldsymbol{\omega}_{rs}}{\kappa_r \kappa_s}. \end{aligned} \tag{6.7}$$

6.3 Iterace

Iterace připraví nulové $\boldsymbol{k}_{ir}$ (`expected_vertex_comm_degree`) a $\boldsymbol{\omega}_{rs}$ (`expected_comm_comm_degree`) a projde všechny hrany. Pro každou hranu $\{i, j\}$ se z předem spočítané redukované matice grafu komunit $\tilde{\boldsymbol{\omega}}_{rs}$ (`reduced_comm_comm_degree`) a matice stupňů vrcholů na komunitu $\boldsymbol{k}_{ir}$

(`expected_vertex_comm_degree`) z předchozí iterace vypočítá rozdělení $q_{ij}(r,s)$ (`edge_probability`), jak popisuje (6.5). Z rozdělení $q_{ij}(r,s)$ se vztahem (6.6) aktualizují matice $\boldsymbol{k}_{ir}$ a $\boldsymbol{\omega}_{rs}$.

Na konci inicializace se z $\boldsymbol{\omega}_{rs}$ opět pro další iteraci připraví stupně komunit κ_r (`expected_comm_degree`) a redukovaná matice grafu komunit $\tilde{\boldsymbol{\omega}}_{rs}$ (`reduced_comm_comm_degree`), viz (6.7). Kritérium pro zastavení je převzato z [3] jako maximum absolutní hodnoty rozdílu matic $\boldsymbol{k}_{ir}$ a $\boldsymbol{\omega}_{rs}$:

$$\max\left\{\max_{i,r}|\boldsymbol{k}_{ir}^{-}-\boldsymbol{k}_{ir}|,\quad \max_{r,s}|\boldsymbol{\omega}_{rs}^{-}-\boldsymbol{\omega}_{rs}|\right\}\leq T, \tag{6.8}$$

kde symbol x^{-} označuje předchozí iteraci a T je nastavený práh. Vzhledem k významu matic $\boldsymbol{k}_{ir}$ a $\boldsymbol{\omega}_{rs}$ lze říci, že kritérium je maximální počet polohran, které najednou změní komunitu. Toto kritérium nezávisí na velikosti grafu. Další případ zastavení je dosažení nastaveného maximálního počtu iterací.

Kapitola 7

Výsledky

7.1 SBM pro komunity bez překryvu

Společným znakem SBM pro komunity bez překryvu je hledání kombinatorického řešení prostřednictvím Kernighanova-Linova algoritmu popsaného v kapitole 3.1.3. KL-algoritmus sestavuje SBM model pro nepřekrývající se komunity. Z implementace přiložených k článkům [24, 26] jsem implementoval třídu v jazyce Python, která sjednocuje všechny varianty

- základní SBM bez kontroly stupňů,
- SBM s kontrolou stupňů,
- bipartitní SBM bez kontroly stupňů a
- bipartitní SBM s kontrolou stupňů.

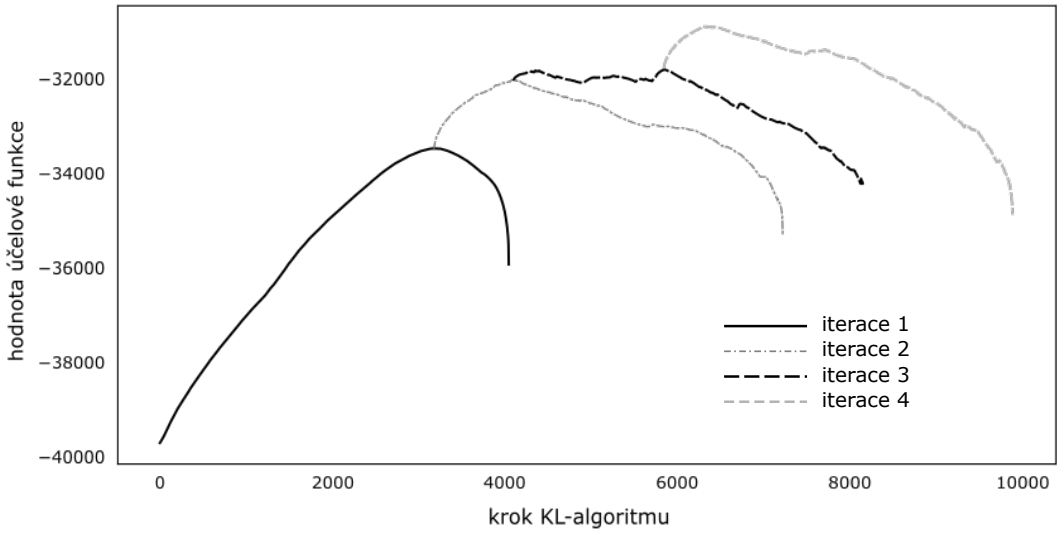
Na obrázku 7.1 je vyobrazen průběh skore KL-algoritmu pro několik iterací. Přesně podle popisu algoritmu je znázorněno, jak každá iterace má tolik kroků, kolik má graf vrcholů a jak každá další iterace vychází z maxima předcházející iterace.

7.2 SBM pro překrývající se komunity a inicalizace

Na obrázku 7.3 je zobrazen průběh hodnoty účelové funkce pro měkkou a tvrdou inicializaci a inicializace pro bipartitní grafy. Algoritmus, jehož počáteční řešení je inicalizováno měkce v prvních iteracích nevykazuje výraznou změnu hodnotící funkce, což vede ke kolizi s kritériem pro zastavení. S tvrdou inicializací vykazuje algoritmus změny od začátku, ale má tendenci uvíznout v lokálním extrému. Implementace tvrdé a měkké inicializace je popsána v kapitole 6.2. Pro bipartitní grafy lze model inicializovat tak, aby byla nulová pravděpodobnost, že konec hrany bude patřit komunitě s opačnou paritou. Metoda pak dosáhne rychleji a vyššího skore.

EM algoritmus v kombinaci s modelem *BKN-SBM* (kap. 3.2) při měkké inicializaci díky diagonální matici $\boldsymbol{\omega}$ a kvadratickým vztahům netrpí pomalou konvergencí v prvních iteracích. Proto u něj ani nedochází ke kolizi s kritériem pro zastavení. Navzdory této výhodě je podle dalších výsledků model *BKN-SBM* absolutně nepoužitelný pro detekci komunit v bipartitních grafech.

Poznámka 7.2.1. Ukázalo se, že pro SBM algoritmus popsaný v této práci je řešení SBM bez překryvu lokálním extrémem a nelze tak použít KL-algoritmus pro inicializaci EM-algoritmu.



Obrázek 7.1: Graf znázorňuje průběh skore účelové funkce Kernighanova-Linova algoritmu pro 4 iterace. Iterace jsou rozlišeny stylem čáry. KL-algoritmus přesune všechny vrcholy do jiné komunity. Každá křivka má délku 4037 kroků, kolik bylo vrcholů v grafu. Následující iterace vychází z kombinace, při které bylo dosaženo maxima účelové funkce jednoho běhu KL-algoritmu.

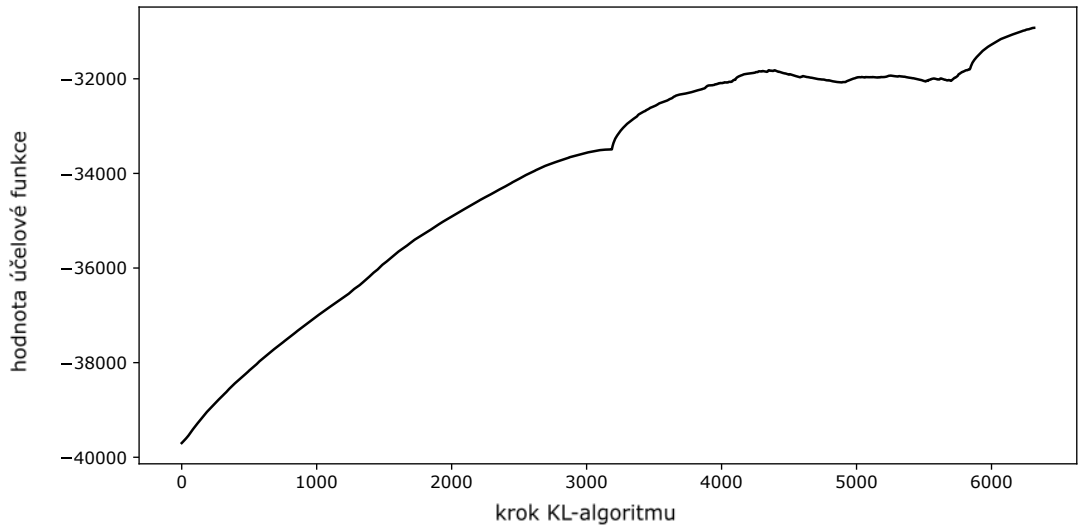
7.3 Srovnání metod SBM vzhledem k překryvu komunit

Na obrázcích 7.2 resp. 7.3 je průběh skóre metod detekujících nepřekrývající se, resp. překrývající se komunity na motivačním grafu (obrázek 2) o přibližně 4000 vrcholech. Na obrázcích 7.4 resp. 7.5 je průběh skóre metod detekujících nepřekrývající se, resp. překrývající se komunity na bipartitním grafu o 100 vrcholech. Z průběhů na obrázcích jsem usoudil několik závěrů. SBM s překryvem komunit (EM-algoritmus) s tvrdou inicializací zkonvergují rychleji, než KL-algoritmus pro komunity bez překryvu, ale nemusí dosáhnout tak vysokého skore. EM-algoritmus s měkkou inicializací dosahuje nejvyššího skore. KL-algoritmus pro velké grafy dokonverguje po řádově větším počtu kroků. Krok KL-algoritmu je podobně náročný jako iterace EM-algoritmu. Porovnáním průběhů na velkém (motivačním) bipartním grafu (obrázky 7.2 a 7.3) a malém grafu o 100 vrcholech (obrázky 7.4 a 7.5) vidíme, že KL-algoritmus na menším grafu nalezne řešení rychleji (co do počtu kroků), než EM-algoritmus s měkkou inicializací.

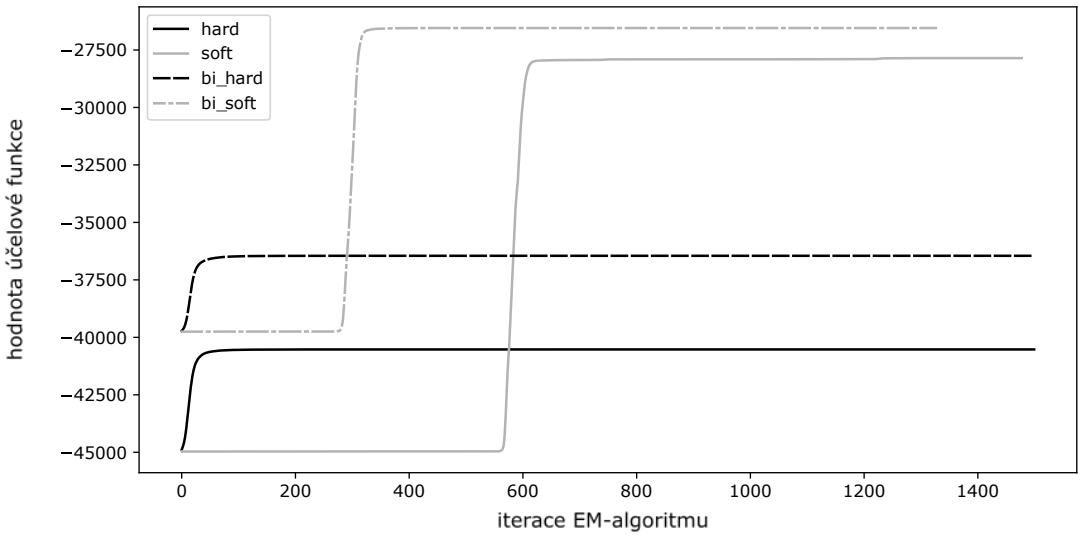
7.4 Detekce na motivačním grafu

Na obrázku 2 je vyobrazen bipartitní graf titulů a jejich nositelů. Zdrojem dat jsou symboly umístěné na objevených hrobkách výše postavených lidí ve starověké egyptské společnosti. Detekce komunit v tomto grafu je motivačním problémem pro hledání algoritmu detekujícího překrývající se komunity v bipartitních grafech. Na obrázcích 7.2 resp. 7.3 je průběh skóre metod detekujících nepřekrývající se, resp. překrývající se komunity.

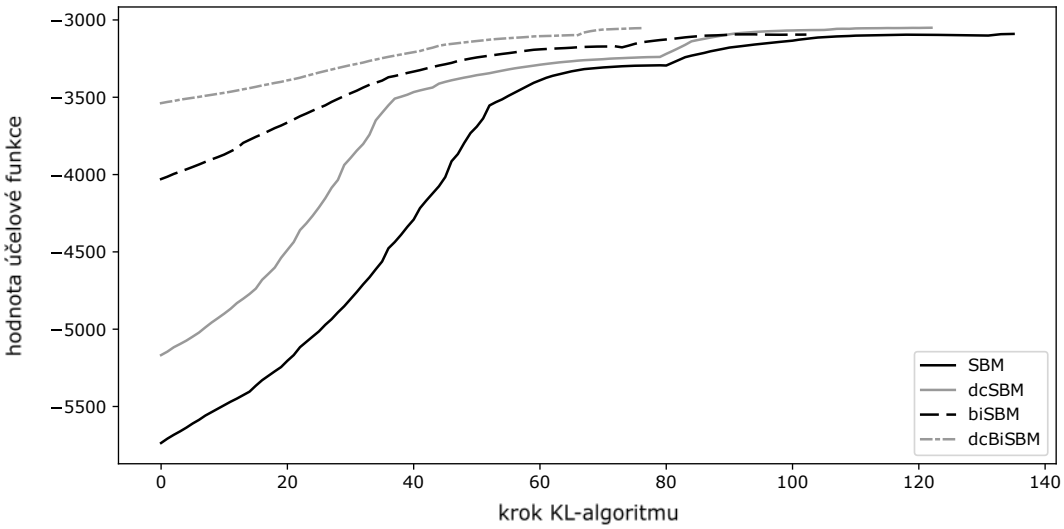
Výsledek detekce překrývajících se komunit je pro jednu komunitu zobrazen na obrázku 7.6. Jedná se o komunitu vezírských titulů, která byla odhalena jinou metodou a jejíž správnost je



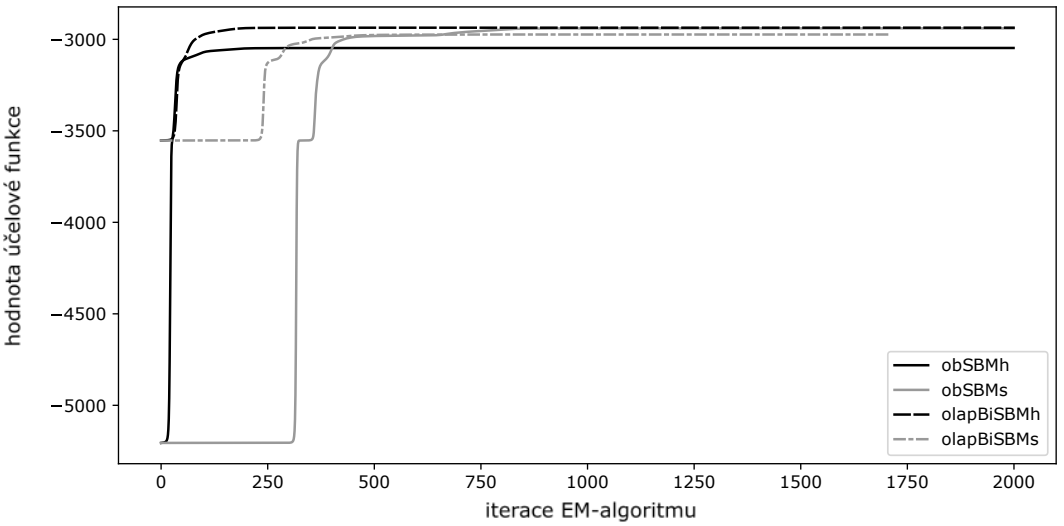
Obrázek 7.2: Průběh účelové funkce KL-algoritmu na motivačním bipartitním grafu.



Obrázek 7.3: Průběh účelové funkce EM-algoritmu na motivačním bipartitním grafu pro různé konfigurace: s měkkou inicializací (soft), s tvrdou inicializací (hard) a s bipartitními komunitami (bi).



Obrázek 7.4: Průběh účelové funkce KL-algoritmu na malém bipartitním grafu o 100 vrcholech pro různé konfigurace: bez kontroly stupňů, s kontrolou stupňů (dc) a bipartitními komunitami (bi).



Obrázek 7.5: Průběh účelové funkce EM-algoritmu na malém bipartitním grafu o 100 vrcholech pro různé konfigurace: s měkkou inicializací (soft), s tvrdou inicializací (hard) a s bipartitními komunitami (bi).

potvrzena [28]. Její diagram je k obrázku přiložen. Komunita titulů je v obrázku vyznačena barevně. Barevná intenzita titulů odpovídá prvkům matice $\boldsymbol{\theta}$ modelu z kapitoly 4. Samozřejmě je vybrán jeden sloupec matice odpovídající zobrazované komunitě. Elegance modelu spočívá v jeho symetrii, která nám umožňuje pouhým součinem matic $\boldsymbol{\theta\omega}$ odpovědět na otázku, kdo jsou nositelé vezírských titulů. Ti jsou v grafu na obrázku vyznačeni odlišnou barvou. Detekční metoda byla nastavena pro hledání 15 komunit titulů a 30 komunit lidí. Inicalizace byla provedena měkce.

7.5 Srovnání detekčních metod

V rámci výzkumného úkolu jsem navrhnul testovací prostředí pro metody detekující komunity v grafech. Testovací prostředí dokáže pro různé varianty SBM modelů s Poissonovým rozdělením vygenerovat náhodné grafy na základě předem daných komunit a jejich vazeb. Vygenerovaný graf je podroben sadě metod pro detekci komunit. Detekované komunity jsou porovnány s komunitami generujícího modelu pomocí Jaccardova a Randova indexu (1.18, 1.19). Postup výpočtu evaluace je uveden v kapitole 7.5.2.

Při evaluaci modelů překrývajících se komunit fitovaných na grafy generované pomocí komunit bez překryvu vyvstal problém s touto penalizací. Využitím překryvu v komunitách lze hrany grafu modelovat přesněji. Nalezený model může být přesnější než model, na jehož základě byl testovací graf vygenerován. Tato penalizace je však zvolena záměrně. Má demonstrovat, že ačkoliv hodnotu účelové funkce lze zvýšit, tak nalezený model nemusí reflektovat principy zakodované do hran v grafu. Vzhledem k tomu, že porovnáváme detekční metody založené na SBM s Poissonovým rozdělením, můžeme kvalitu modelu vzhledem ke grafům poměřovat hodnotou účelové funkce (3.5), dále skore.

7.5.1 Testovací grafy

Pro účely testování vlastností metod a jejich srovnání byli generovány náhodné grafy Poissonovým rozdělením podle pravděpodobnosti hrany (3.2) ze čtyř druhů modelů. Generující blokové modely byli zvoleny:

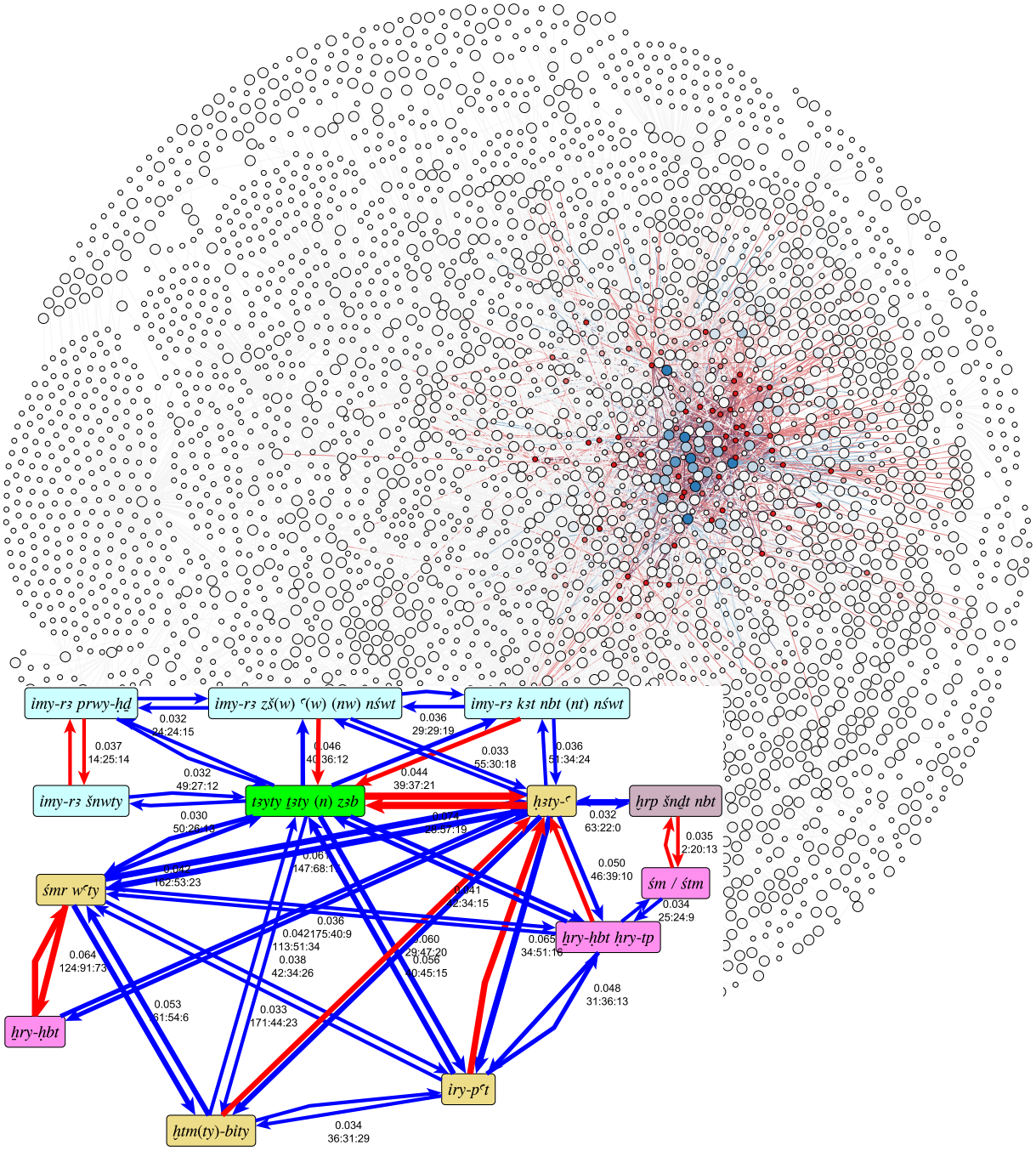
- unipartitní graf s nepřekrývajícími se komunitami,
- unipartitní graf s překrývajícími se komunitami,
- bipartitní graf s nepřekrývajícími se komunitami a
- bipartitní graf s překrývajícími se komunitami.

Ukázka testovacích grafů je na obrázku 7.7. Modely byli parametrizovány následujícím způsoby.

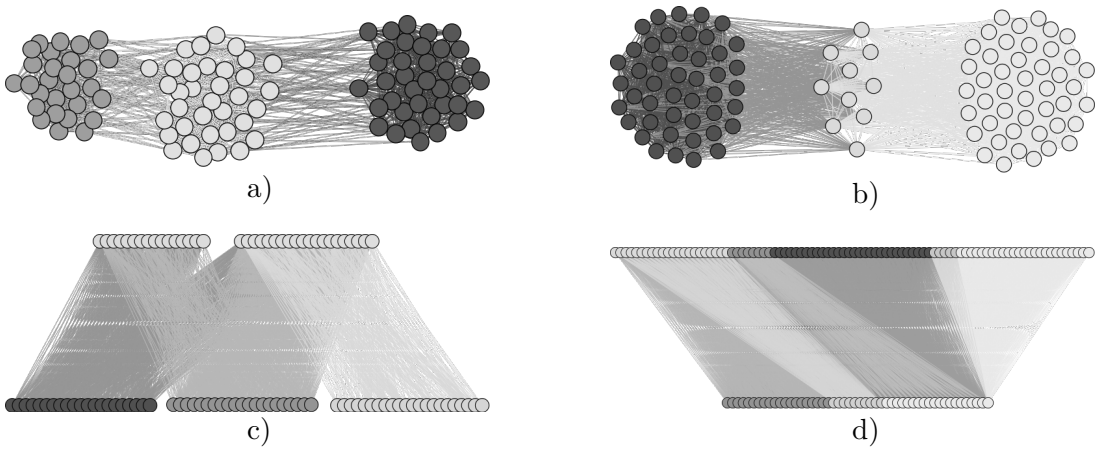
Model s nepřekrývajícími se komunitami pro unipartitní graf byl tvořen 3 komunitami. Matice komunit $\boldsymbol{\omega}$ měla tvar

$$\boldsymbol{\omega} = \begin{pmatrix} 1-\mu & \mu & 0 \\ \mu & 1-2\mu & \mu \\ 0 & \mu & 1-\mu \end{pmatrix}, \tag{7.1}$$

kde μ je mixovací parametr udávající vzájemné propojení komunit. V testech nabývá hodnot $\mu \in \{0; 0{,}01; 0{,}1; 0{,}2\}$ pro grafy s $N = 100$ vrcholy a $\mu = 0{,}15$ pro grafy s $N \in \{200, 400, 800\}$ vrcholy.



Obrázek 7.6: Na obrázku je detail motivačního grafu z obrázku 2. Velké vrcholy představují tituly a malé vrcholy představují lidi. Dále je k obrázku zobrazen diagram komunity vezírských titulů, která byla identifikována jinou metodou. Diagram je převzat z [28]. Byla aplikována detekce překrývajích se komunit pomocí modelu popsaného v kapitole 4. Barevně jsou modrou barvou • vyznačeny tituly patřících do komunity vezírských titulů. Červenou barvou • jsou vyznačeny vrcholy patřící do komunit, které jsou propojeny s komunitou vezírských titulů.



Obrázek 7.7: Ukázka testovacích grafů o 100 vrcholech: a) unipartitní graf generovaný modelem 3 nepřekrývajících se komunit s mixovacím parametrem $\mu = 0{,}1$, b) unipartitní graf generovaný modelem s 2 překrývajícími se komunitami a 10 společnými vrcholy, c) bipartitní graf popsaný 5 nepřekrývajícmi se komunitami s mixovacím parametrem $\mu = 0{,}1$, d) bipartitní graf 5 překrývajících se komunit s 10 a 16 společnými vrcholy.

Model s překrývajícími se komunitami pro unipartitní graf byl tvořen 2 komunitami a parametry byl počet vrcholů a počet společných vrcholů.

Model s nepřekrývajícími se komunitami pro bipartitní graf byl tvořen 3 komunitami parity A a 2 komunitami parity B. Matice komunit měla tvar

$$\boldsymbol{C} = \begin{pmatrix} 1-\mu & \mu & 0 \\ \mu & 1-2\mu & \mu \end{pmatrix}. \tag{7.2}$$

Mixovací parametr nabýval hodnot $\mu \in \{0; 0{,}01; 0{,}1; 0{,}2\}$ pro grafy s $N = 100$ vrcholy a $\mu = 0{,}15$ pro grafy s $N \in \{200, 400\}$ vrcholy.

Model s překrývajícími se komunitami pro bipartitní graf byl tvořen 2 komunitami parity A a 3 komunitami parity B. Parametrizován byl počtem vcholů a počtem společných vrcholů.

7.5.2 Hodnotící funkce

Hodnotící funkci jsem převzal ze svého výzkumného úkolu. Úkolem hodnotící funkce je spočítat jedno číslo vyjadřuje, jak moc se od sebe liší generující a detekovaný model. Zobrazit výsledky detekce, obzvláště překrývajících se komunit, je obtížné a podobně obtížné je kvantifikovat shodu nebo odlišnost nalezené struktury grafu a zadané struktury grafu. Celý postup výpočtu hodnocení detekční metody je schematicky znázorněn na obrázku 7.8.

Z rozdělení vrcholů do komunit, případně z $\boldsymbol{\theta}$, se ke každé komunitě vytvoří seznam vrcholů, které v ní mají členství. U překrývajících se komunit se tak zanedbá informace o síle členství vrcholu v komunitě. Matice $\boldsymbol{\omega}$ nemá na hodnotu hodnotící funkce vliv. V detekovaném modelu je více komunit, než v generujícím modelu. Navíc komunity jsou v jiném pořadí. Detekované

Zkratka	kontrola stupňů	bipartitní model	překrývající se komunity	algoritmus	Poznámka
original	-	-	-	-	generující model
SBM	NE	NE	NE	KL	
dcSBM	ANO	NE	NE	KL	
olapSBM	ANO	NE	ANO	EM	BKN-SBM
obSBMh	ANO	NE	ANO	EM	tvrdá inicializace
obSBMs	ANO	NE	ANO	EM	měkká inicializace
biSBM	NE	ANO	NE	KL	
dcBiSBM	ANO	ANO	NE	KL	
olapBiSBMh	ANO	ANO	ANO	EM	tvrdá inicializace
olapBiSBMs	ANO	ANO	ANO	EM	měkká inicializace

Tabulka 7.1: Významy zkratek detekčních metod a jejich konfigurace.

komunity jsou proto namapovány na generující komunity. Jako mapovací funkci používám Jaccardův index (1.18) pro určení shodných členů komunit. Sloučené detekované komunity podle nejvyšší hodnoty Jaccardova indexu s generující komunitou označuji jako *agregovaná komunita*. Metoda je penalizována hodnotící funkcí, když agregované komunity obsahují další (cizí) vrcholy v porovnání s generujícími komunitami.

Cílem předchozích kroků je získat čtvercovou matici, jejíž prvky tvoří podíly počtu společných vrcholů vůči počtu vrcholů v grafu podle Randova indexu (1.19). Na diagonále hodnota odpovídá velikosti komunity a mimo diagonálu velikosti překryvu komunit. V posledním kroku jsou vypočteny čtvercové matice Randova indexu (1.19)

$$\boldsymbol{R}^A_{rs} = R(M^A_r, M^A_s), \quad \boldsymbol{R}^G_{rs} = R(M^G_r, M^G_s) \tag{7.3}$$

pro agregované komunity a pro generující indexy, kde M^A_r a M^G_r jsou množiny členských vrcholů agregovaných a generujících komunit pro $r \in \hat{k}$ a k je počet generujících komunit. Matice jsou čtvercové a jejich řádky a sloupce si vzájemně odpovídají. Celkové hodnocení detekční metody je dáno určením „vzdálenosti“ těchto dvou matic pomocí Frobeniovy normy (1.17) jako

$$\text{EVAL} = ||R^A - R^G||_F. \tag{7.4}$$

7.5.3 Výsledky detekce

V testovacím prostředí, které jsem navrhnul včetně hodnotící funkce ve výzkumném úkolu jsem otestoval vlastnosti detekčních metod. Grafická reprezentace výsledků zobrazují obrázky 7.9, 7.10, 7.11, 7.12 a 7.13. Podrobnější hodnoty výsledků uvádějí tabulky výsledků 7.3, 7.4, 7.5 a 7.6. V diagramech a tabulkách jsou hodnoty hodnotící funkce a výsledné hodnoty účelové funkce. Jedna vyjadřuje rozdíl generujícího a detekovaného modelu (eval). Druhá vyjadřuje shodu vygenerovaného grafu a k němu detekovanému modelu (skore). Dále jsou v diagramech a tabulkách uváděny zkratky detekčních metod a jejich konfigurace. Význam zkratek je uveden v tabulce 7.1.

Z diagramů na obrázcích je patrné, že generující model (original) má vždy nulovou hodnotu hodnotící funkce testovacího prostředí (eval). Jedná se totiž o vzdálenost dvou identických matic. Oproti tomu má generující model nižší (horší) hodnotu účelové funkce (score), která vyjadřuje

Model	vrcholů typu A	vrcholů typu B	společných vrcholů A	společných vrcholů B
A	50	90	0	0
B	50	90	0	10
C	50	90	0	16
D	50	90	10	0
E	50	90	10	10
F	50	90	10	16
G	50	90	20	0
H	50	90	20	10
I	50	90	20	16

Tabulka 7.2: Parametry modelů bipartitního grafu překrývajících se komunit pro hodnocení detekčních metod zobrazených v obrázcích 7.12 a 7.13.

shodu generujícího modelu s vygenerovaným grafem. Generující model je vždy obecnější, než konkrétně vygenerovaný graf.

Z hodnot na obrázcích 7.9 a 7.11 vidíme, že u grafu vygenerovaného podle modelu s komunitami bez překryvu, jsou metody detekující překrývající se komunity penalizovány hodnotící funkci, ale současně dosahují lepšího skore. Jak bylo v úvodu této kapitoly uvedeno, důvodem je, že model s překrývajícími komunitami dokáže konkrétní graf popsat lépe pomocí vrcholů ve více komunitách současně. Tedy detekované komunity jsou rozdílné oproti generujícím, což způsobuje vyšší penalizaci, ale detekovaný model vygenerovanému grafu odpovídá dobře, o čemž vypovídá účelová funkce.

U bipartitních grafů, jejichž výsledky jsou na obrázcích 7.11, 7.12 a 7.13, je nejhůře hodnocena metoda *olapSBM* z článku [3]. Důvodem je, že model překrývajících se komunit popsaný v [3] a stejně tak obdobný model [43], předpokládá a vynucuje, aby komunita byla množina vrcholů, které uvnitř komunity mají poměrně více hran v porovnání se zbytkem grafu. Matice $\boldsymbol{\omega}$ tohoto SBM modelu je striktně diagonální. Takový požadavek u bipartitního grafu, který naopak zakazuje hrany mezi vrcholy stejné parity, je neuspokojitelný.

Konkrétně na obrázku 7.11 jsou diagramy výsledků testování metod. Metody byly testovány na bipartitních grafech generovaných podle nepřekrývajících se komunit. Výsledky jsou rozděleny podle parametrů generujících modelů (A, B, C, ...) a barevně podle testovaných metod. V horní části je penalizace hodnotící funkcí (eval). Generující model (original) má nulovou a nejlepší penalizaci. Dále metody nepřekrývajících se komunit mají nízkou penalizaci. Metody překrývajících komunit mají penalizaci vysokou. V dolní části je výsledná hodnota účelové funkce (skore). Zde má nejhorší a nejzápornější hodnotu generující model (original), protože je obecnější, než konkrétní realizace náhodného grafu. Druhé nejhorší skore má model BKN-SBM (olapSBM), která má problém popsat bipartitní graf. Ostatní metody aproximují náhodné grafy srovnatelně a podstatně lépe. Černými úsečkami jsou vyznačeny směrodatné odchylky.

Model	Metoda	Skóre	Hodnocení
A	original	-1561,834± 58,517	0,000±0,000
A	SBM	-1515,659± 59,712	0,000±0,000
A	dcSBM	-1495,707± 59,921	0,000±0,000
A	olapSBM	-1475,997± 56,466	0,000±0,000
A	obSBMs	-1468,206± 55,450	0,000±0,000
A	obSBMh	-1483,512± 77,564	0,009±0,040
B	original	-1673,266± 36,557	0,000±0,000
B	SBM	-1597,903± 35,483	0,000±0,000
B	dcSBM	-1577,431± 33,933	0,000±0,000
B	olapSBM	-1497,286±246,466	0,225±0,051
B	obSBMs	-1535,003± 31,311	0,225±0,047
B	obSBMh	-1548,884± 39,718	0,217±0,046
C	original	-2195,545± 58,583	0,000±0,000
C	SBM	-2091,512± 55,927	0,000±0,000
C	dcSBM	-2063,625± 56,243	0,000±0,000
C	olapSBM	-1949,606± 49,665	0,866±0,054
C	obSBMs	-1959,526± 50,778	0,777±0,072
C	obSBMh	-1972,336± 55,792	0,791±0,083
D	original	-2532,631± 67,845	0,000±0,000
D	SBM	-2428,863± 66,173	0,001±0,003
D	dcSBM	-2397,088± 66,021	0,002±0,005
D	olapSBM	-2244,865± 58,515	1,107±0,068
D	obSBMs	-2261,069± 62,129	0,984±0,111
D	obSBMh	-2271,874± 61,160	0,989±0,104
E	original	-2400,241± 81,858	0,000±0,000
E	SBM	-2290,290± 76,167	0,000±0,000
E	dcSBM	-2265,228± 77,178	0,000±0,000
E	olapSBM	-2130,563± 65,673	1,006±0,073
E	obSBMs	-2144,145± 67,943	0,925±0,080
E	obSBMh	-2154,802± 69,557	0,936±0,120
F	original	-2406,259± 62,126	0,000±0,000
F	SBM	-2297,180± 57,371	0,000±0,000
F	dcSBM	-2270,583± 58,561	0,000±0,000
F	olapSBM	-2129,950± 51,847	1,028±0,063
F	obSBMs	-2146,928± 52,561	0,943±0,069
F	obSBMh	-2158,510± 50,294	0,940±0,108
G	original	-2423,234± 83,434	0,000±0,000
G	SBM	-2314,394± 77,748	0,000±0,002
G	dcSBM	-2289,622± 77,084	0,000±0,000
G	olapSBM	-2148,972± 70,407	1,011±0,077
G	obSBMs	-2161,115± 74,719	0,929±0,072
G	obSBMh	-2172,437± 73,713	0,964±0,103

Parametry generujících modelů počet vrcholů N a mixovací parmatr μ jsou

Model	N	μ
A	100	0,00
B	100	0,01
C	100	0,10
D	100	0,20
E	200	0,15
F	400	0,15
G	800	0,15

Tabulka 7.3: Výsledky testovacího prostředí pro unipartitní grafy generované nepřekrývajícími se komunitami.

Model	Metoda	Skóre		Hodnocení
A	original	-5488,182±	2213,513	0,000±0,000
A	SBM	-2461,850±	13,296	0,115±0,000
A	dcSBM	-2449,215±	13,548	0,115±0,000
A	olapSBM	-2445,766±	25,015	0,000±0,000
A	obSBMs	-2427,335±	21,853	0,003±0,007
A	obSBMh	-2422,167±	20,725	0,001±0,003
B	original	-6074,371±	2659,648	0,000±0,000
B	SBM	-2620,502±	16,642	0,192±0,000
B	dcSBM	-2603,486±	17,358	0,192±0,000
B	olapSBM	-2594,147±	26,602	0,000±0,000
B	obSBMs	-2579,979±	23,780	0,002±0,007
B	obSBMh	-2574,012±	22,034	0,001±0,004
C	original	-6802,076±	2369,530	0,000±0,000
C	SBM	-2733,236±	15,170	0,104±0,000
C	dcSBM	-2719,407±	15,474	0,104±0,000
C	olapSBM	-2709,464±	16,164	0,000±0,000
C	obSBMs	-2699,207±	14,836	0,003±0,009
C	obSBMh	-2695,917±	17,228	0,002±0,006
D	original	-6611,874±	1923,580	0,000±0,000
D	SBM	-2909,238±	16,627	0,173±0,000
D	dcSBM	-2888,052±	15,435	0,173±0,000
D	olapSBM	-2875,676±	18,952	0,000±0,000
D	obSBMs	-2862,106±	20,417	0,003±0,009
D	obSBMh	-2860,269±	20,395	0,009±0,013
E	original	-8925,328±	3954,739	0,000±0,000
E	SBM	-3312,554±	17,652	0,346±0,000
E	dcSBM	-3290,964±	17,094	0,346±0,000
E	olapSBM	-3279,757±	21,460	0,000±0,000
E	obSBMs	-3270,877±	135,480	0,020±0,093
E	obSBMh	-3252,738±	22,061	0,006±0,014
F	original	-31212,001±	12758,973	0,000±0,000
F	SBM	-12527,155±	51,828	0,260±0,000
F	dcSBM	-12490,585±	40,060	0,260±0,000
F	olapSBM	-12454,700±	44,935	0,000±0,000
F	obSBMs	-12821,444±	1296,916	0,058±0,196
F	obSBMh	-12403,055±	48,733	0,003±0,007
G	original	-135853,085±	54332,748	0,000±0,000
G	SBM	-50234,740±	138,074	0,260±0,000
G	dcSBM	-50143,857±	106,225	0,260±0,000
G	olapSBM	-50179,104±	111,822	0,000±0,000
G	obSBMs	-69277,110±	188,828	0,746±0,000
G	obSBMh	-49960,412±	130,394	0,001±0,002

Parametry generujících modelů o dvou překrývajících se komunitách velikost první komunity $|C_1|$, velikost druhé komunity $|C_2|$ a počet společných vrcholů $|C_1 \cap C_2|$ jsou

Model	$\lvert C_1\rvert$	$\lvert C_2\rvert$	$\lvert C_1 \cap C_2\rvert$
A	30	60	6
B	30	60	10
C	50	50	6
D	50	50	10
E	50	50	20
F	100	100	30
G	200	200	60

Tabulka 7.4: Výsledky testovacího prostředí pro unipartitní grafy generované překrývajícími se komunitami.

Model	Metoda	Skóre		Hodnocení
A	original	-1703,062±	684,060	0,000±0,000
A	SBM	-729,264±	80,599	0,207±0,041
A	dcSBM	-758,566±	82,678	0,207±0,041
A	olapSBM	-1241,885±	134,329	0,567±0,137
A	obSBMs	-769,546±	100,554	0,217±0,047
A	obSBMh	-773,863±	87,960	0,209±0,041
A	biSBM	-767,356±	90,368	0,210±0,044
A	dcBiSBM	-755,730±	83,096	0,208±0,042
A	olapBiSBMs	-755,118±	81,948	0,207±0,041
A	olapBiSBMh	-753,389±	82,975	0,208±0,042
B	original	-2331,134±	844,187	0,000±0,000
B	SBM	-1102,803±	89,288	0,050±0,028
B	dcSBM	-1075,643±	90,908	0,099±0,054
B	olapSBM	-1553,124±	236,405	0,845±0,117
B	obSBMs	-1064,818±	114,419	0,395±0,074
B	obSBMh	-1095,337±	83,063	0,371±0,078
B	biSBM	-1099,103±	90,300	0,054±0,030
B	dcBiSBM	-1074,545±	88,480	0,088±0,048
B	olapBiSBMs	-1034,067±	84,995	0,380±0,068
B	olapBiSBMh	-1048,301±	82,732	0,357±0,059
C	original	-3551,521±	1138,928	0,000±0,000
C	SBM	-1700,015±	121,091	0,039±0,024
C	dcSBM	-1668,273±	120,415	0,036±0,024
C	olapSBM	-2429,917±	195,577	1,373±0,166
C	obSBMs	-1620,472±	120,327	0,692±0,091
C	obSBMh	-1644,827±	110,049	0,631±0,075
C	biSBM	-1695,381±	119,121	0,051±0,028
C	dcBiSBM	-1666,849±	120,316	0,033±0,021
C	olapBiSBMs	-1613,911±	118,107	0,685±0,103
C	olapBiSBMh	-1615,413±	117,723	0,677±0,075
D	original	-3464,652±	825,238	0,000±0,000
D	SBM	-1806,651±	131,140	0,065±0,031
D	dcSBM	-1777,027±	131,209	0,065±0,038
D	olapSBM	-2639,484±	210,229	1,608±0,246
D	obSBMs	-1737,378±	129,897	0,760±0,130
D	obSBMh	-1751,985±	126,532	0,699±0,105
D	biSBM	-1802,518±	137,131	0,081±0,036
D	dcBiSBM	-1774,259±	137,677	0,069±0,034
D	olapBiSBMs	-1730,239±	129,397	0,754±0,134
D	olapBiSBMh	-1730,973±	133,971	0,745±0,116
E	original	-17060,514±	3397,047	0,000±0,000
E	SBM	-7260,493±	270,102	0,034±0,013
E	dcSBM	-7154,939±	266,525	0,032±0,013
E	olapSBM	-10634,996±	395,004	1,745±0,212
E	obSBMs	-7037,006±	252,713	0,789±0,074
E	obSBMh	-7039,828±	264,161	0,766±0,092
E	biSBM	-7239,827±	274,067	0,048±0,020
E	dcBiSBM	-7150,190±	267,501	0,035±0,015
E	olapBiSBMs	-7009,631±	260,324	0,797±0,125
E	olapBiSBMh	-7004,211±	260,558	0,835±0,102
F	original	-67403,012±	17566,916	0,000±0,000
F	SBM	-28855,619±	900,396	0,045±0,011
F	dcSBM	-28526,203±	909,461	0,045±0,016
F	olapSBM	-43371,536±	1382,291	2,091±0,233
F	obSBMs	-30429,784±	6398,980	0,926±0,159
F	obSBMh	-28092,570±	883,021	1,019±0,121
F	biSBM	-28842,382±	926,239	0,052±0,014
F	dcBiSBM	-28566,538±	933,771	0,043±0,012
F	olapBiSBMs	-28990,386±	1981,431	0,794±0,196
F	olapBiSBMh	-28204,721±	903,110	0,960±0,143

Parametry generujících modelů počet vrcholů N, počet komunit prvního typu K_A, počet komunit druhého typu K_B, a mixovací parmatr μ jsou

Model	N	K_A	K_B	μ
A	100	3	2	0,00
B	100	3	2	0,01
C	100	3	2	0,10
D	100	3	2	0,20
E	200	3	2	0,15
F	400	3	2	0,15

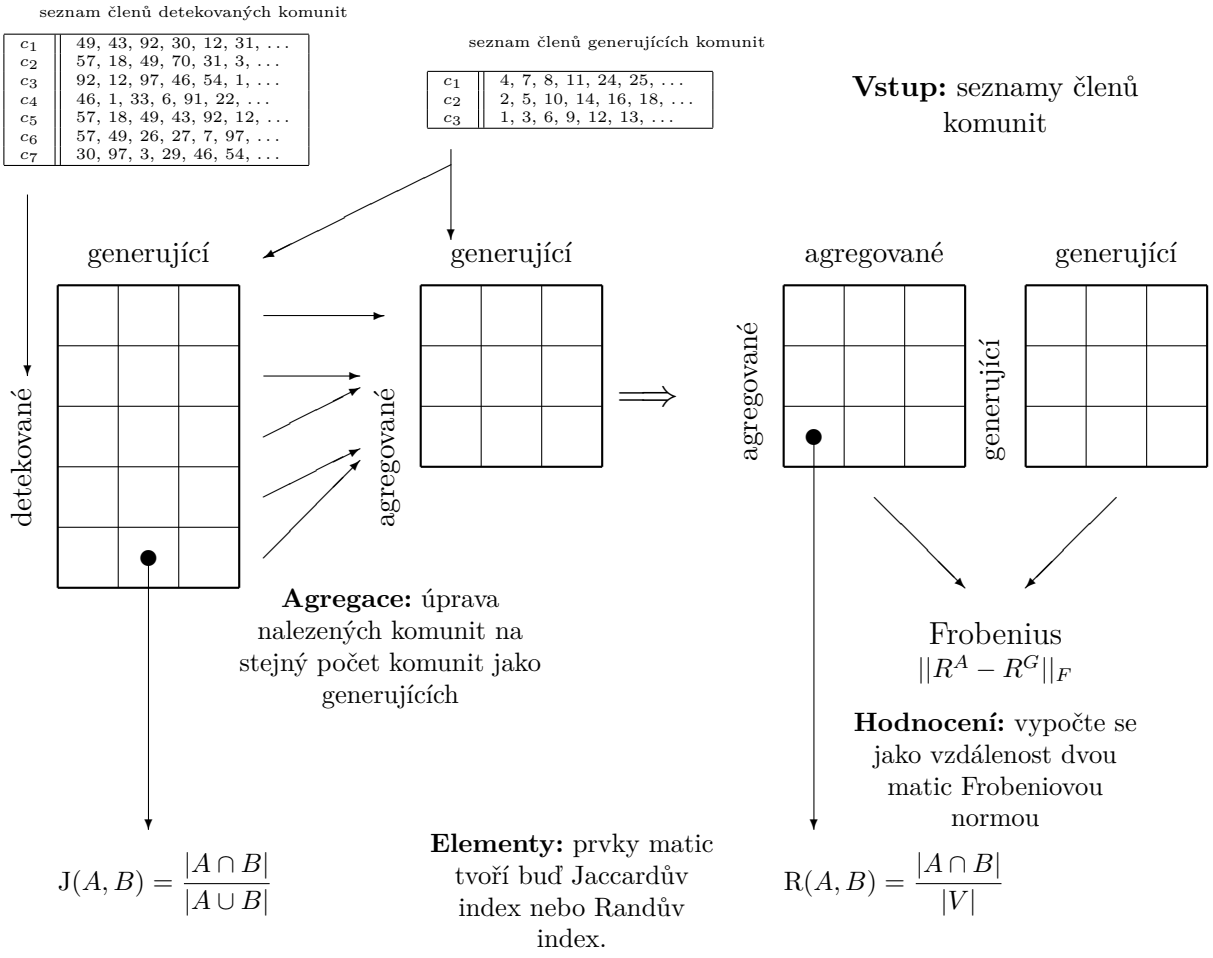
Tabulka 7.5: Výsledky testovacího prostředí pro bipartitní grafy generované nepřekrývajícími se komunitami.

Model	Metoda	Skóre	Hodnocení
A	original	-6631,496±1547,650	0,000±0,000
A	SBM	-2981,527± 10,318	0,000±0,000
A	dcSBM	-2969,428± 6,844	0,000±0,000
A	olapSBM	-4860,109± 36,840	1,547±0,199
A	obSBMs	-2957,571± 9,500	0,644±0,099
A	obSBMh	-2964,326± 15,310	0,608±0,118
A	biSBM	-2979,060± 13,471	0,000±0,000
A	dcBiSBM	-2969,938± 6,305	0,000±0,000
A	olapBiSBMs	-2953,699± 8,738	0,678±0,126
A	olapBiSBMh	-2953,868± 8,481	0,648±0,129
B	original	-6880,498±1549,844	0,000±0,000
B	SBM	-3104,711± 6,363	0,061±0,002
B	dcSBM	-3090,875± 4,839	0,061±0,002
B	olapSBM	-5056,037± 30,443	1,511±0,251
B	obSBMs	-3077,626± 9,622	0,601±0,117
B	obSBMh	-3083,369± 9,236	0,530±0,088
B	biSBM	-3101,350± 13,095	0,061±0,001
B	dcBiSBM	-3090,295± 5,813	0,061±0,002
B	olapBiSBMs	-3074,738± 8,476	0,571±0,119
B	olapBiSBMh	-3074,129± 7,768	0,608±0,105
C	original	-7200,624±1909,657	0,000±0,000
C	SBM	-3194,566± 115,329	0,100±0,025
C	dcSBM	-3162,041± 7,552	0,096±0,003
C	olapSBM	-5165,447± 38,218	1,426±0,217
C	obSBMs	-3146,627± 10,015	0,560±0,105
C	obSBMh	-3153,522± 12,649	0,528±0,072
C	biSBM	-3173,768± 10,432	0,098±0,003
C	dcBiSBM	-3161,197± 7,647	0,097±0,003
C	olapBiSBMs	-3143,988± 10,358	0,556±0,102
C	olapBiSBMh	-3145,009± 10,165	0,572±0,108
D	original	-8011,490±2572,925	0,000±0,000
D	SBM	-3420,862± 175,902	0,146±0,033
D	dcSBM	-3351,684± 11,144	0,138±0,001
D	olapSBM	-5443,233± 47,271	1,608±0,249
D	obSBMs	-3357,987± 127,976	0,626±0,121
D	obSBMh	-3345,386± 13,870	0,593±0,119
D	biSBM	-3369,962± 12,696	0,138±0,001
D	dcBiSBM	-3352,147± 9,022	0,138±0,001
D	olapBiSBMs	-3335,868± 13,634	0,694±0,137
D	olapBiSBMh	-3336,091± 13,773	0,643±0,136
E	original	-8545,626±2532,142	0,000±0,000
E	SBM	-3516,086± 167,461	0,159±0,033
E	dcSBM	-3451,721± 11,346	0,151±0,001
E	olapSBM	-5603,323± 44,767	1,374±0,257
E	obSBMs	-3438,086± 13,741	0,565±0,098
E	obSBMh	-3444,105± 12,854	0,529±0,095
E	biSBM	-3464,881± 19,670	0,151±0,001
E	dcBiSBM	-3451,140± 10,577	0,151±0,001
E	olapBiSBMs	-3433,985± 12,549	0,529±0,083
E	olapBiSBMh	-3432,557± 13,415	0,536±0,091
F	original	-8065,613±1850,392	0,000±0,000
F	SBM	-3601,804± 195,549	0,181±0,038
F	dcSBM	-3510,014± 12,836	0,169±0,002
F	olapSBM	-5685,251± 58,177	1,224±0,237
F	obSBMs	-3493,851± 17,558	0,531±0,084
F	obSBMh	-3499,860± 22,832	0,480±0,067
F	biSBM	-3524,293± 17,212	0,169±0,002
F	dcBiSBM	-3507,995± 12,128	0,169±0,001
F	olapBiSBMs	-3489,904± 15,724	0,502±0,073
F	olapBiSBMh	-3489,613± 15,929	0,514±0,076
G	original	-9349,068±3131,500	0,000±0,000
G	SBM	-3787,307± 170,804	0,288±0,036
G	dcSBM	-3694,575± 26,885	0,276±0,001
G	olapSBM	-5955,150± 48,604	1,478±0,223
G	obSBMs	-3673,751± 14,061	0,579±0,123
G	obSBMh	-3673,043± 13,032	0,576±0,118
G	biSBM	-3704,939± 21,196	0,276±0,001
G	dcBiSBM	-3687,916± 11,757	0,276±0,001
G	olapBiSBMs	-3668,629± 13,082	0,574±0,121
G	olapBiSBMh	-3665,048± 14,981	0,585±0,120
H	original	-8937,879±3359,451	0,000±0,000
H	SBM	-3888,304± 190,932	0,306±0,047
H	dcSBM	-3760,815± 10,914	0,282±0,002
H	olapSBM	-6058,258± 50,210	1,241±0,173
H	obSBMs	-3741,631± 16,586	0,514±0,079
H	obSBMh	-3744,564± 15,193	0,521±0,094
H	biSBM	-3772,789± 21,942	0,283±0,001
H	dcBiSBM	-3758,331± 10,451	0,283±0,001
H	olapBiSBMs	-3737,413± 15,648	0,482±0,078
H	olapBiSBMh	-3735,248± 16,123	0,519±0,088
I	original	-8796,342±3199,834	0,000±0,000
I	SBM	-3937,120± 169,272	0,310±0,039
I	dcSBM	-3808,686± 10,522	0,292±0,002
I	olapSBM	-6139,393± 44,770	1,179±0,195
I	obSBMs	-3788,404± 13,285	0,494±0,078
I	obSBMh	-3791,149± 14,472	0,445±0,051
I	biSBM	-3827,000± 14,397	0,293±0,002
I	dcBiSBM	-3805,535± 9,869	0,293±0,001
I	olapBiSBMs	-3783,836± 13,168	0,446±0,001
I	olapBiSBMh	-3781,400± 13,160	0,475±0,059

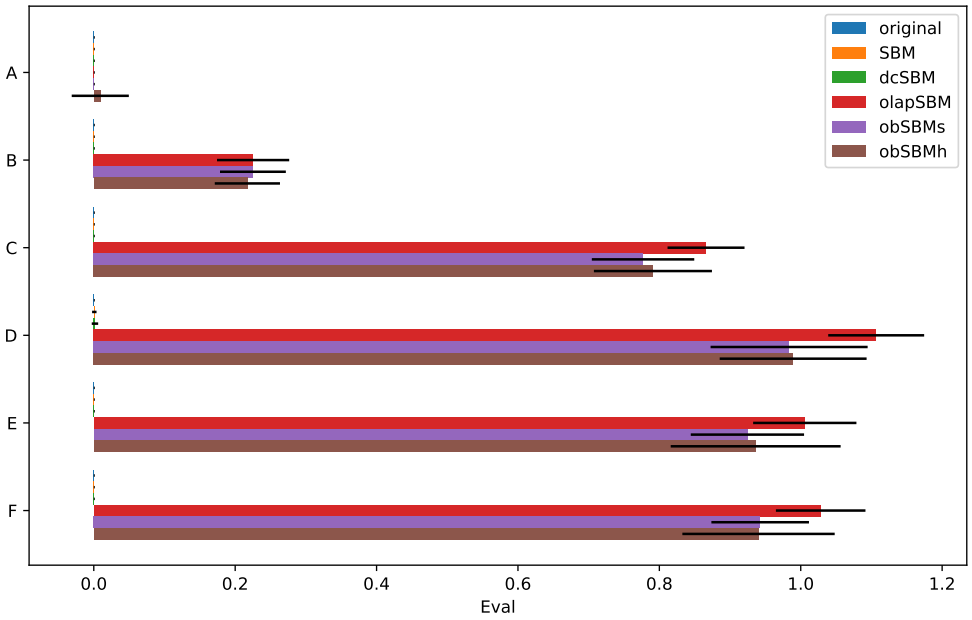
Parametry generujících modelů počet vrcholů prvního typu N_A, počet vrcholů druhého typu N_B, počet společných vrcholů $|C_{Ar} \cap C_{As}|$ v komunitách prvního typu, a počet společných vrcholů $|C_{Br} \cap C_{Bs}|$ v komunitách prvního typu jsou

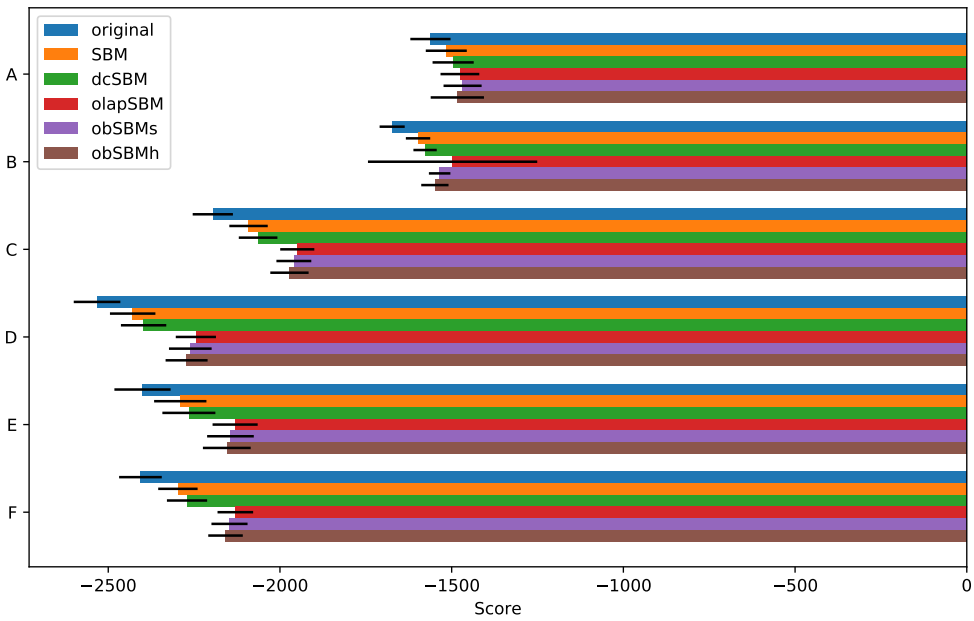
Model	$\|N_A\|$	$\|N_B\|$	$\|C_{Ar} \cap C_{As}\|$	$\|C_{Br} \cap C_{Bs}\|$
A	50	90	0	0
B	50	90	0	10
C	50	90	0	16
D	50	90	10	0
E	50	90	10	10
F	50	90	10	16
G	50	90	20	0
H	50	90	20	10
I	50	90	20	16

Tabulka 7.6: Výsledky testovacího prostředí pro bipartitní grafy generované překrývajícími se komunitami.

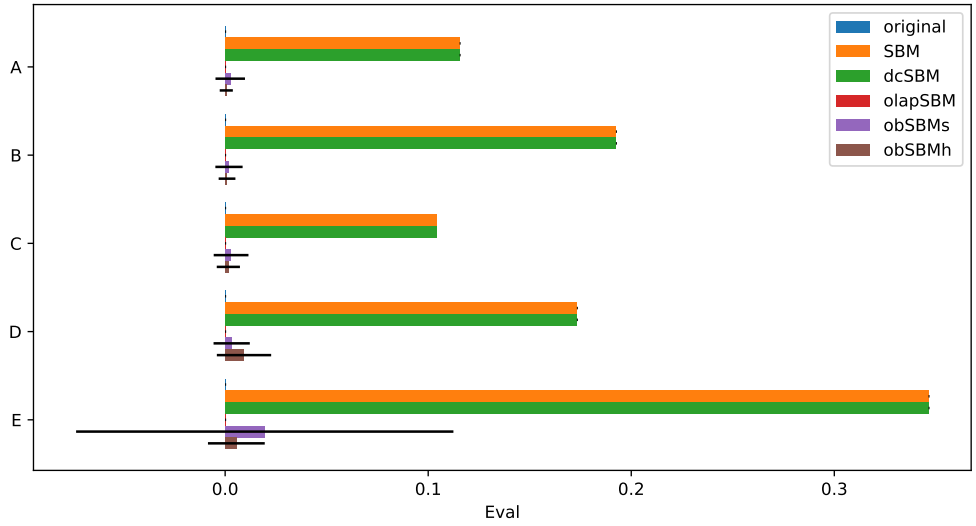


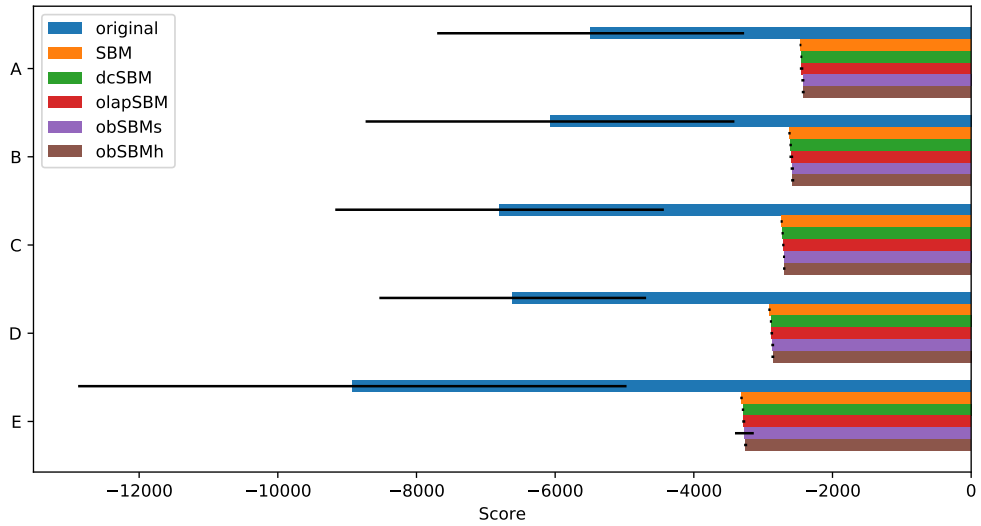
Obrázek 7.8: Schéma výpočtu hodnotící funkce. Vstupem hodnotící funkce jsou dva seznamy členů komunit. Jeden pro detekované komunity a jeden pro zadané (generující) komunity. Prvním krokem je agregace detekovaných komunit. Ke každé detekované komunitě se použitím Jaccardova indexu (1.18) nalezne nejbližší generující komunita. Podle generujících komunit se detekované komunity sloučí. V druhém kroku se vypočtou matice Randova indexu (1.19), jak pro již sloučené (agregované) detekované komunity, tak pro generující komunity. Tyto matice jsou čtvercové a stejného rozměru. Hodnoty jejich prvků odpovídají velikosti komunit a jejich překryvů. Navíc sloupce a řádky si v obou maticích odpovídají. Posledním krokem je poměření rozdílů v těchto maticích pomocí Frobeniovy normy.



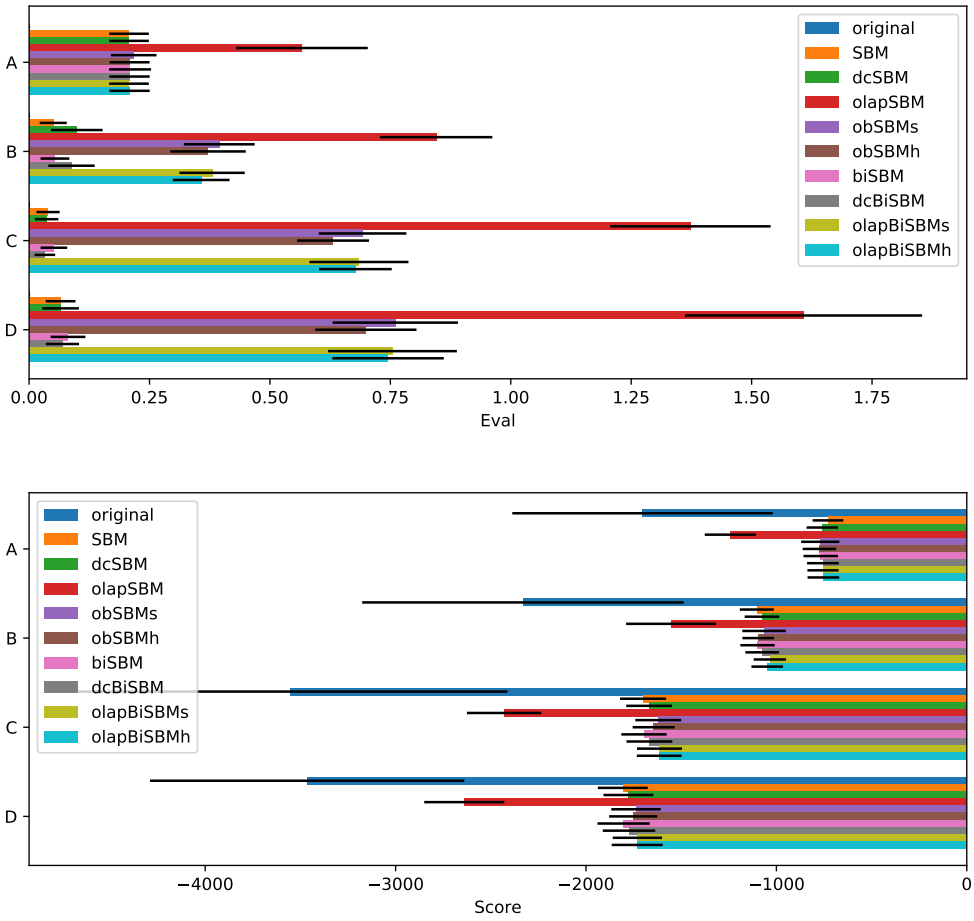


Obrázek 7.9: Evaluace (eval) a hodnota účelové funkce (score) detekčních metod na unipartitním grafu generovaného modelem nepřekrývajících se komunit s mixovacím parametrem (A) $\mu = 0$ (B) $\mu = 0{,}01$ (C) $\mu = 0{,}1$ (D) $\mu = 0{,}2$ a s $\mu = 0{,}15$ pro počet vrcholů 200 (E) a 400 (F).

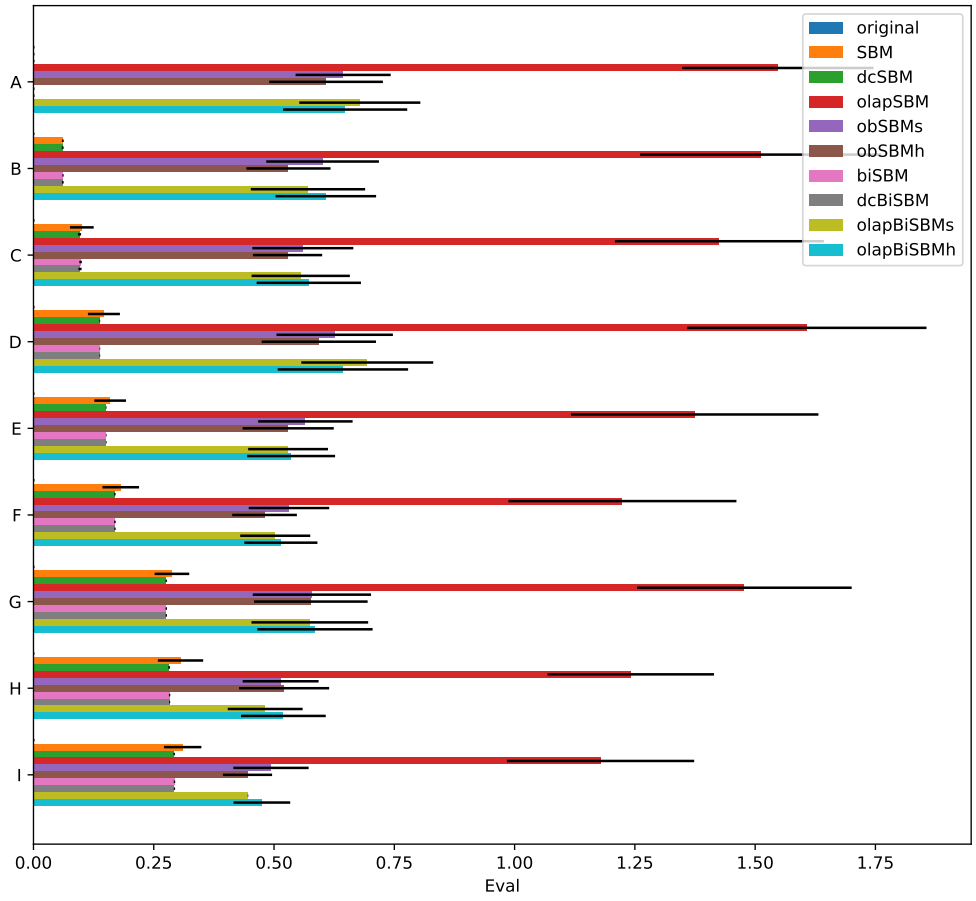




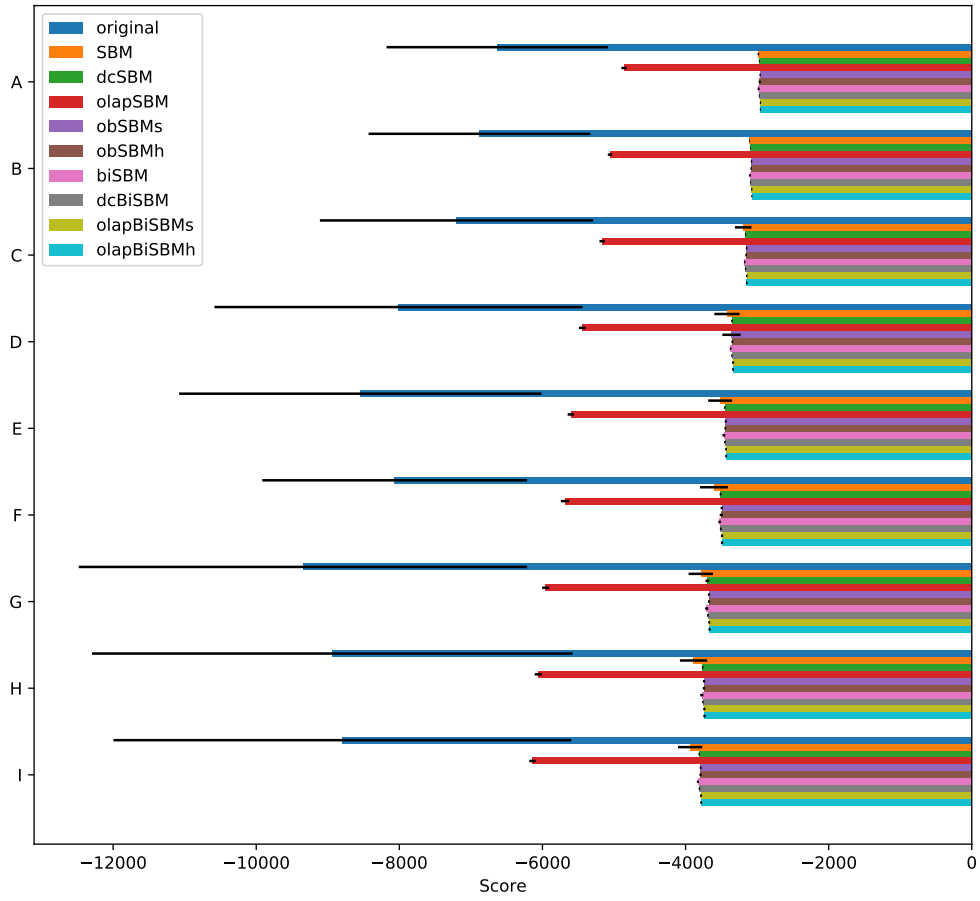
Obrázek 7.10: Evaluace (eval) a hodnota účelové funkce (score) detekčních metod na unipartitním grafu generovaného modelem překrývajících se 2 komunit o velikosti 30 a 60 s 6 společnými vrcholy (A), 10 společnými vrcholy (B), o velikosti 50 a 50 vrcholů s 6 (C), 10 (D) a 20 (E) společnými vrcholy.



Obrázek 7.11: Evaluace (eval) a hodnota účelové funkce (score) detekčních metod na bipartitním grafu generovaného modelem nepřekrývajících se komunit s mixovacím parametrem (A) $\mu = 0$ (B) $\mu = 0{,}01$ (C) $\mu = 0{,}1$ (D) $\mu = 0{,}2$.



Obrázek 7.12: Evaluace detekčních metod na bipartitním grafu generovaného modelem překrývajících se komunit. Parametry jsou uvedeny v tabulce 7.2.



Obrázek 7.13: Výsledná hodnota účelové funkce detekovaných modelů na bipartitním grafu generovaného modelem překrývajících se komunit. Parametry jsou uvedeny v tabulce 7.2.

Závěr

Práci jsem rozdělil na 7 kapitol bez úvodu. V úvodu jsem uvedl motivační problém vedoucí k požadavku detekovat překrývající komunity v bipartitních grafech. V matematickém úvodu (kap. 1) jsem vytvořil stručný přehled definicí a vět potřebných k pochopení principů rozebíraných v práci. V další kapitole 2 jsem se pokusil zařadit detekci komunit v grafech po boku obdobných problémů zpracování dat, shlukování, strojového učení nebo redukce dimenze. Po obecném úvodu jsem se zaměřil v kapitole 3 už konkrétněji na stochastické blokové modely s Poissonovým rozdělením a k nim optimalizační metody, jejich prostřednictvím se detekují komunity v grafech. Sjednotil jsem značení a odvodil jsem společné jednotné vlastnosti pro tuto třídu modelů.

Základem práce je dokončení popisujícího modelu pro překrývající se komunity v k-partitních grafech v kapitole 4. Zde jsem se opíral o část výsledků mého výzkumného úkolu a článku [3]. Druhým pilířem je dokončení omezujících podmínek modelu a dokončení implementace navržené v [3]. V kapitole 5 jsou uvedeny detaily odvození. Podrobný popis implementace je v kapitole 6. V poslední kapitole 7 byli srovnány vlasnosti a dovedností metod ze třídy stochastických blokových modelů s Poissonovým rozdělením, kam také patří metoda z této práce. Experimentálně byly ověřeny některé předpoklady.

V experimentální části je komentována konvergence metod pro různá nastavení. Dále byly metody testovány na kombinacích unipartitních a bipartitních grafů s nepřekrývajícími se a překrývajícími se komunitami. Testovací prostředí vyvinuté v rámci výzkumného úkolu generovalo podle zadaného modelu náhodné grafy, které byly podrobeny detekčním metodám za účelem odhalení zadaného modelu. U detekovaných modelů byla hodnocena shoda s generovacím modelem pomocí hodnotící funkce navržené ve výzkumném úkolu, popsané v kapitole 7.5.2. Dále jsem využil sjednocenou účelovou funkci (3.5) pro hodnocení, jak dobře detekované modely aproximují konkrétní realizaci náhodného grafu.

Z průběhů skore jako hodnotící funkce pro KL a EM algoritmy na obrázcích 7.2, 7.3, 7.4 a 7.5 jsem usoudil, že KL-algoritmus řešící SBM modely pro nepřekrývající se komunity nalézá řešení rychleji, co do počtu kroků výpočtů, než EM algoritmus hledající překrývající se komunity. U velkých grafů je situace opačná.

Dále je komentován vývoj hodnotící funkce EM algoritmu se zde navrženým modelem. Během vývoje implementace se dlouhou dobu nedařilo překonat problém měkké inicializace modelu s pomalou konvergencí v prvních iteracích a kolizí s kritériem pro zastavení. Zdálo se, že metoda nebude dostatečně účinná pro jakoukoliv detekci komunit.

Metody detekce komunit jsem aplikoval také na motivační graf lidí a titulů z obrázku 2. Především jsem se zajímal o detekci překrývajících se komunit. Jedná se o graf založen na skutečných datech. Jeho struktura není kompletně známa. Nicméně se nám s vedoucím práce podařilo rozpoznat jednu z klíčových komunit titulů. Tato klíčová komunita byla odhalena jinou metodou mimo tuto práci a validována Egyptology. Tento výsledek je také na obrázku 7.6.

Ze srovnání metod lze hodnotit z hlediska aproximace modelů k testovanému grafu všechny

srovnatelně úspěšné, o čemž vypovídá hodnota účelové funkce (skore). Výjimku tvoří *BKN-SBM* při použití na bipartitních grafech. Z hlediska nalezení generujících komunit lze konstatovat, že vysokou penalizaci (eval) dostaly „nevhodně“ použité metody, tedy detekce nepřekrývajících komunit na grafech generovaných pomocí překrývajících komunit a detekce překrývajících v grafech generovaných podle nepřekrývajících se komunít. Navržená metoda v této práci si nejlépe vedla v odhalování generujících komunit na unipartitních a pak na bipartitních grafech generovaných pomocí překrývajících se komunit.

Pro předem daný počet komunit jsou sestaveny modely a náhodně inicializovány. Detekční metody dále optimalizují své modely tak, aby odpovídaly struktuře grafu. Žádná z rozebíraných metod optimalizující modely však neřeší optimální počet komunit, které má model obsahovat, nebo jejich hierarchii. Tento nedostatek je příležitostí k dalšímu rozvoji metod pro detekci komunit. Jednou z cest by mohla být nějaká analogie k hledání „vlastních vektorů“ grafu nebo PCA.

Literatura

[1] Charu C. Aggarwal. *Data Mining: The Textbook*. Springer, 2015.

[2] Phipps Arabie and Lawrence J. Hubert. Combinatorial data analysis. *Annual Review of Psychology*, 43:169–203, 1992.

[3] Brian Ball, Brian Karrer, and M. E. J. Newman. An efficient and principled method for detecting communities in networks. *CoRR*, abs/1104.3590, 2011.

[4] Suman Banerjee, Mamata Jenamani, and Dilip Kumar Pratihar. Properties of a projected network of a bipartite network. *CoRR*, abs/1707.00912, 2017.

[5] J.A. Bondy and U.S.R. Murty. *Graph Theory with Applications: By J.A. Bondy and U.S.R. Murty*. Macmillan, 1976.

[6] Federico Botta and Charo I del Genio. Finding network communities using modularity density. *Journal of Statistical Mechanics: Theory and Experiment*, 2016(12):123402, 2016.

[7] Stephen Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

[8] Aaron Clauset, M. E. J. Newman, and Cristopher Moore. Finding community structure in very large networks. *Physical Review E*, 70(6):066111+, August 2004.

[9] T. M. Cover. *Elements of information theory*. Wiley series in telecommunications. Wiley, New York, 1991.

[10] Arthur P Dempster, Nan M Laird, and Donald B Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society: Series B (Methodological)*, 39(1):1–22, 1977.

[11] Reinhard Diestel. *Graph Theory, 4th Edition*, volume 173 of *Graduate texts in mathematics*. Springer, 2012.

[12] Veronika Dulíková, Radek Mařík, Miroslav Barta, and Matej Cibuľa. HMM model vývoje a trendů správy země v období Staré říše. In *16. ročník konference Počítačová podpora v archeologii, Písek CZ, 29. - 31. května 2017*. Katedra archeologie Západočeské univerzity v Plzni, CZ, 2017.

[13] Dario Fasino and Francesco Tudisco. A modularity based spectral method for simultaneous community and anti-community detection. *Linear Algebra and its Applications*, September 2017.

[14] Santo Fortunato. Community detection in graphs. *Physics Reports*, 486(3-5):75–174, January 2010.

[15] Santo Fortunato and Claudio Castellano. Community structure in graphs. 2007.

[16] Thomas M. J. Fruchterman and Edward M. Reingold. Graph drawing by force-directed placement. *Software: Practice and Experience*, 21(11):1129–1164, 1991.

[17] Mohadeseh Ganji. *Semi-supervised community detection and clustering.* PhD thesis, 2018/03/19 2017.

[18] Jiawei Han, Micheline Kamber, and Jian Pei. Data mining concepts and techniques, third edition, 2012.

[19] David Heeger. Poisson model of spike generation, 2000.

[20] Arnon Hershkovitz, Ronit Azran, Sharon Hardof-Jaffe, and Rafi Nachmias. Types of online hierarchical repository structures. *The Internet and Higher Education*, 14:107–112, 2011.

[21] Ngoc-Diep Ho. *Non-negative matrix factorization: Algorithms and applications.* PhD thesis, U.C. Louvain, June 2008.

[22] Cho-Jui Hsieh and Inderjit S. Dhillon. Fast coordinate descent methods with variable selection for non-negative matrix factorization. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '11, pages 1064–1072, New York, NY, USA, 2011. ACM.

[23] Anil K. Jain and Richard C. Dubes. *Algorithms for Clustering Data.* Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1988.

[24] Brian Karrer and M. E. J. Newman. Stochastic blockmodels and community structure in networks. August 2010.

[25] Solomon Kullback. *Information theory and statistics.* Peter Smith, 1978.

[26] Daniel B. Larremore, Aaron Clauset, and Abigail Z. Jacobs. Efficiently inferring community structure in bipartite networks. *CoRR*, abs/1403.2933, 2014.

[27] Tao Li and Chris H. Q. Ding. Nonnegative matrix factorizations for clustering: A survey. In *Data Clustering: Algorithms and Applications*, pages 149–176. 2013.

[28] Radek Marik. Feature space decomposition using information theory. In *Proceedings of the 2018 International Conference on Algorithms, Computing and Artificial Intelligence*, ACAI 2018, New York, NY, USA, 2018. Association for Computing Machinery.

[29] Zide Meng, Fabien Gandon, Catherine Faron-Zucker, and Ge Song. Detecting topics and overlapping communities in question and answer sites. *Social Network Analysis and Mining*, 5(1), jun 2015.

[30] M. E. J. Newman. Fast algorithm for detecting community structure in networks. *Physical Review E*, 69(6):066133+, September 2003.

[31] M. E. J. Newman. Modularity and community structure in networks. *Proceedings of the National Academy of Sciences*, 103(23):8577–8582, February 2006.

[32] M. E. J. Newman and M. Girvan. Finding and evaluating community structure in networks. *Physical Review E*, 69(2):026113+, August 2003.

[33] Mark E. J. Newman. *Networks: An Introduction*. Oxford University Press, 2010.

[34] MEJ Newman. Power laws, pareto distributions and zipf's law. *Contemporary Physics*, 46(5):323–351, 2005.

[35] Krzysztof Nowicki and Tom A.B. Snijders. Estimation and prediction for stochastic blockstructures. *Journal of the American Statistical Association*, 96(455):1077–1087, 2001.

[36] Paola Pesantez-Cabrera and Ananth Kalyanaraman. Detecting communities in biological bipartite networks. In *Proceedings of the 7th ACM International Conference on Bioinformatics, Computational Biology, and Health Informatics*, BCB '16, pages 98–107, New York, NY, USA, 2016. ACM.

[37] William M. Rand. Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical Association*, 66(336):846–850, 1971.

[38] Douglas Steinley and Michael J. Brusco. K-means clustering and mixture model clustering: Reply to mclachlan (2011) and vermunt (2011). *Psychological Methods*, 16:89–92, 2011.

[39] Gilbert Strang. *Linear Algebra and Its Applications, 4th Edition*. Cengage Learning, 2006.

[40] Matt Visser. Zipf's law, power laws and maximum entropy. *New Journal of Physics*, 15(4):043021, 2013.

[41] Ian H. Witten, Eibe Frank, and Mark A. Hall. *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 3rd edition, 2011.

[42] Jaewon Yang and Jure Leskovec. Structure and overlaps of communities in networks, September 2012.

[43] Jaewon Yang and Jure Leskovec. Overlapping community detection at scale: a nonnegative matrix factorization approach. In *Proceedings of the sixth ACM international conference on Web search and data mining*, pages 587–596. ACM, 2013.

[44] Zhenhua. *Computational intelligence and intelligent systems : 6th International Symposium, ISICA 2012, Wuhan, China, October 27-28, 2012. Proceedings*. Springer, Berlin New York, 2012.