Metode za identifikaciju spacijalnih regija ekspresije gena

Prusina, Tomislav

Master's thesis / Diplomski rad

2022

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **Josip Juraj Strossmayer University of Osijek, Department of Mathematics / Sveučilište Josipa Jurja Strossmayera u Osijeku, Odjel za matematiku**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:126:227733

Sveučilište Josipa Jurja Strossmayera u Osijeku

Odjel za matematiku

Sveučilišni diplomski studij matematike
smjer: Matematika i računarstvo

Metode za identifikaciju spacijalnih regija ekspresije gena

Diplomski rad

Mentor:

izv.prof.dr.sc.
Domagoj Matijević

Kandidat:

Tomislav Prusina

Osijek, 2022

Sadržaj

1 | Uvod

Promatranje organa i tkiva javlja se kao potreba u raznim znanostima poput medicine, biologije i agronomije. Od primjena u medicini i agronomiji, raspoznavanje tkiva je važno za detekciju neželjenih tkiva, prepoznavanje anomalija, utvrđivanje svojstava organa ili traženje uzročnika bolesti. Jedan od problema koji se javlja pri prepoznavanju tkiva je nepreciznost ljudskog oka i ogromna količina promatranih informacija. Kao pomoć javljaju se mnogi algoritmi raznih pristupa, korištenih tehnologija i rezultata. U ovom radu bavit ćemo se tim algoritmima specijaliziranim za problem označavanja tkiva istih funkcija sa naglaskom na metode bazirane na neuronskim mrežama.

Kao najjednostavniji primjer problema označavanja tkiva navodimo mozak. Mozak, ljudski ili mišji organ, ima dobro vidljiva različita tkiva i dobro istaknute rubove među njima. Jedan od ekstenzivnijih radova koji se bavio proučavanjem ljudskog mozga je *Transcriptome-scale spatial gene expression in the human dorsolateral prefrontal cortex* [28] čije rezultate mnogi znanstvenici koriste za testiranje svojih ideja. Kratak opis DLPFC podataka nalazi se na slici 1.1. Te podatke, kao i algoritamski generirane podatke, ćemo koristiti za evaluaciju algoritama. Dobivene rezultate zajedno sa slikama i mjerama točnosti ćemo prikazat u odjeljku eksperimenti.

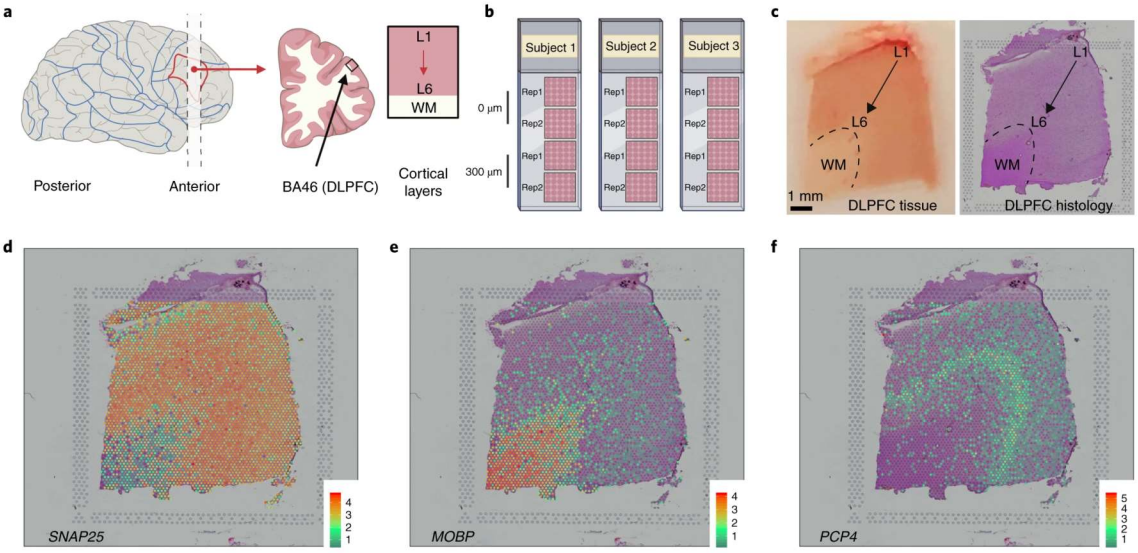
Ovaj problem kojim ćemo se baviti možemo i matematički formalno definirati. Naime, tkivo koje promatramo reprezentirat ćemo dvijema matricama,

$$X \in \mathbb{N}^{n \times m} \text{ i } D \in \mathbb{R}^{n \times 2}.$$

Matricu X ćemo zvati matricom ekspresije gena. Ona predstavlja broj određenog gena u pojedinoj stanici. Stupac matrice X predstavlja jednu vrstu gena a redak predstavlja jednu stanicu. Dakle naše tkivo ima n stanica i m različitih gena. Dodatno, uz matricu ekspresije gena poznata nam je prostorna razmještenost gena. Tu spacijalnu informaciju nam daje matrica D. Svaki redak matrice D odgovara stanici matrice X i njegovi elementi su koordinate na slici tkiva, obično izražene u mikrometrima. Za svaku od metoda ćemo tražiti da za dane podatke X, D i broj $k \in \mathbb{N}$ domena koje želimo, vrati niz labela

$$labels_i \in \{1 \dots k\}, \forall i = 1 \dots n$$

kojem klasteru/domeni pripada [44].



Slika 1.1: a, Tkivni blokovi DLPFC-a dobiveni su u anatomskoj ravnini okomitoj na površinu piala i prošireni do spoja sive i bijele tvari. b, Shema eksperimentalnog dizajna uključujući dva para prostornih replika od tri neovisna neurotipična odrasla donora. c, DLPFC blok tkiva i odgovarajuća histologija iz uzorka 151673. d–f, Spotplots koji prikazuju log (transformiranu normaliziranu ekspresiju) (log(brojevi)) za uzorak 151673 za gene SNAP25 (d), MOBP (e) i PCP4 (f). (Slika preuzeta sa Transcriptome-scale spatial gene expression in the human dorsolateral prefrontal cortex.)

Neke od metoda kao dodatan argument primaju histologiju, tj. sliku tkiva $tissue\_img \in \mathbb{R}^{n\times 3}$ u RGB [45] formatu, no iako imamo tu informaciju za DLPFC podatke, nećemo ih dati metodama. Dodatno, poznate su nam originalne labele za sve podatke što ćemo iskoristiti u evaluaciji metoda. Kao dodatnu informaciju, svakom od algoritama ćemo proslijediti broj originalnih domena kao broj traženih klastera.

2 | Pregled metoda

Različiti su pristupi u rješavanju danog problem od kojih su neki bazirani na vjerojatnosnim modelima, neki na hvatanju topologije stanica, a neki su bazirani na popularnim matematičkim konceptima poznatim kao neuronske mreže. U ovom poglavlju ćemo navesti prednosti i nedostatke nekih od metoda. Budući da se iza svake od metoda krije puno matematike, razmišljanja i utrošenog vremena, mi ćemo se u ovom radu više fokusirati samo na metode bazirane na neuronskim mrežama dok ćemo ostale metode samo kratko opisati.

U našem problemu, kako smo već naveli u uvodnom dijelu, dano je tkivo čije stanične domene želimo prepoznati i označiti. Dano je tkivo koje ima n stanica i m različitih gena koje je zapisano u matricama

$$X \in \mathbb{N}^{n \times m} \text{ i } D \in \mathbb{R}^{n \times 2}.$$

Matrica X je matrica ekspresije gena a matrica D pozicijska matrica gdje $D_i = (x, y)$ označava koordinate stanice i u promatranom tkivu. Svaki od dalje navedenih algoritama prima ove dvije matrice i broj traženih domena $k \in \mathbb{N}$ kao ulaz, a vraća listu oznaka

$$labels_i \in \{1 \dots k\}, \forall i = 1 \dots n$$

kojem klasteru koja stanica pripada.

Prvi pristup koji ćemo predstaviti je matematički zanimljiv pristup baziran na Markovljevim slučajnim poljima (Markov random fields).

Metode bazirane na Markovljevim slučajnim poljima. Prva skupina metoda koju ćemo predstaviti u ovom radu bazirana je na vjerojatnosnom modelu zvanom Markovljeva slučajna polja. Markovljevo slučajno polje [20] je skup slučajnih varijabli čije je Markovljevo svojstvo opisano neusmjerenim grafom. Modeliranje podataka Markovljevim slučajnim poljem i optimizacijom njenih parametara, ove metode simuliraju neprekidnost tkiva i time dobivaju na većoj prostornoj preciznosti. Kao pretpostavku uzimaju da podaci dolaze iz prirode i time imaju "prirodnu" strukturu, što većinom i je istina za dani problem.

Za glavnog predstavnika ovog pristupa predstavljamo metodu zvanu BayesSpace [51]. BayesSpace je jedna od state-of-the-art metoda koje se koriste u bioinformatici i sve novije metode se uspoređuju sa njom. Ova metoda implementira potpuni Bayesov model sa Markov random fieldom prije zbližavanja

točaka istog klastera. Takvi modeli su i prije bili korišteni za analizu slika. Promatrano tkivo je raspoređeno u heksagonalne ili pravokutne rešetke što daje prirodan način definiranja strukture susjedstva. Matematički rečeno, za svaku točku i i njenu nižu d-dimenzionalnu reprezentaciju y_i (dobivenu sa na primjer PCA [10]) modeliramo pripadnost klasteru k sa vjerojatnosti

$$(y_i \mid z_i = k,\ w_i) \sim \mathcal{N}\left(y_i;\ \mu_k, w_i^{-1}\Lambda^{-1}\right)$$

gdje $z_i \in \{1, \ldots, q\}$ predstavlja latentni kluster kojem stanica i pripada, μ_k je očekivani centar klastera k, Λ predstavlja matricu preciznosti a $w_i \in \mathbb{R}$ je neki nepoznatni faktor skaliranja. Navedeni parametri su inicijalizirani uzorkovanjem specijalnih distribucija i dalje optimizirani Gibbsovim uzorkovanjem [12] i Metropolis-Hastings algoritmom [15].

Kao specijalizacija BayesSpace metode javlja se BayesTME[50] metoda koja modelira više različitih svojstava u tumor mikro-okolini.
Jedna od metoda koju je važno spomenut je FICT [40] metoda. FICT za bazu koristi generativni model koji je specijalni slučaj Hidden Markov Random Field modela i ona je specijalizirana za podatke tipa FISH.

Sljedeću vrstu metoda koje ćemo predstaviti u ovom radu bit će metode bazirane na prostornoj razmještenosti gena.

Metode bazirane na identifikaciji manifold struktura. Ove topološke metode pokušavaju uhvatiti strukturu ulaznih podataka za koje se pretpostavlja da su zadani kao unija više različitih manifolda u visoko-dimenzionalnom prostoru. Drugim riječima, pretpostavka koju slijede je da su podaci reprezentirani u visokoj dimenziji (koja ovisi o broju različitih gena) već podijeljeni u manifolde po vrsti tkiva kojem pripadaju. Identifikacija manifolda se radi nelinearnim preslikavanjem u niže dimenzionalni prostor, tzv. prostor značajki (feature space). U niže dimenzionalnom prosturu ima smisla koristiti algoritme klasteriranja i algoritme detekcije manifolda za rješavanje danog problema.

U ovom radu, kao glavnog predstavnika ovog pristupa uzimamo Seurat [13, 36, 39, 14]. Seurat je paket programskog jezik R napravljen za statističku obradu i analizu tkiva. Jedna od funkcija tog paketa je i klasteriranje tkiva. Taj proces možemo sažeti u par koraka: projiciramo podatke u niži prostor pomoću PCA, u tom prostoru konstruirajmo graf pomoću WNN (weighted k-nearest neighbours) [2] te zatim klasteriramo podatke pomoću Louvain algoritma [4]. Ovakav naivan pristup ne daje dobre rezultate iako je korišten u mnogim napredniji metodama za predprocesuiranje podataka.

Naravno, sniženje dimenzije i algoritam za klasteriranje nisu ograničeni samo na PCA i Louvain. U praksi je učestalo korištenje metoda za sniženje dimenzije poput tSNE[41] i UMAP [27] koje se koriste za vizualizaciju podataka projiciranih u dvije ili tri dimenzije. Algoritmi klasteriranja su brojni, Louvain

i Leiden za community detection, k-means i EM za tradicionalno klasteriranje te mclust algoritam stohastičke linearne regresije red(reference). Od navedenih algoritama Louvain metoda je najpopularnija za klasteriranje i scanpy[46] paket programskog jezika python ju samu koristi za detekciju klastera tkiva.

Od topoloških metoda vrijedno je spomenuti metodu STEEL [49] koja matričnim manipulacijama detektira manifolde ugrađene u višu dimenziju i time klasterira tkivo. No na žalost ni ova metoda nema rezultate dobre kao trenutno korištene state-of-the-art metode.

Metode bazirane na neuronskim mrežama. Ove metode postižu bolje rezultate u praksi od dosad opisanih i nadograđuju se na ideje topoloških metoda. Za razliku od topoloških metoda koje pretpostavljaju neku povezanost između podataka, metode bazirane na neuronskim mrežama (NN metode) [17] uče svojstva gena i prostorne informacije koje su specifične za svaku od domena. Budući da su NN metode glavni fokus ovoga rada, mi ćemo posvetiti cijelo sljedeće poglavlje samo njima.

3 | Metode bazirane na neuronskim mrežama

U ovom poglavlju ćemo se baviti metodama zasnovanim na neuronskim mrežama. Svakoj od metoda ćemo posvetiti jedan odjeljak u kojoj ćemo ju detaljnije pojasniti.

NN metode, kao što iz imena možemo naslutiti, su metode bazirane na učenju neuronskih mreža. Neuronska mreža je funkcija uređena po uzoru na ljudski mozak. Ona se sastoji od ulaznog, izlaznog i skrivenih slojeva. Pojedinačni neuroni, kao i slojevi, su opremljeni aktivacijskim funkcijama te su međusobno spojeni vezama kroz koje putuju signali. Neuronske mreže su trenutno dosta popularne i imaju razne primjene u prevođenju jezika, prepoznavanju lica, predviđanju vremenske prognoze, generiranju priča, skladbi i umjetničkih slika, igranju šaha i go-a, pa i u problemu klasteriranja tkiva. Prva od NN metoda koju ćemo predstaviti u ovom radi zove se SpaGCN.

3.1 SpaGCN

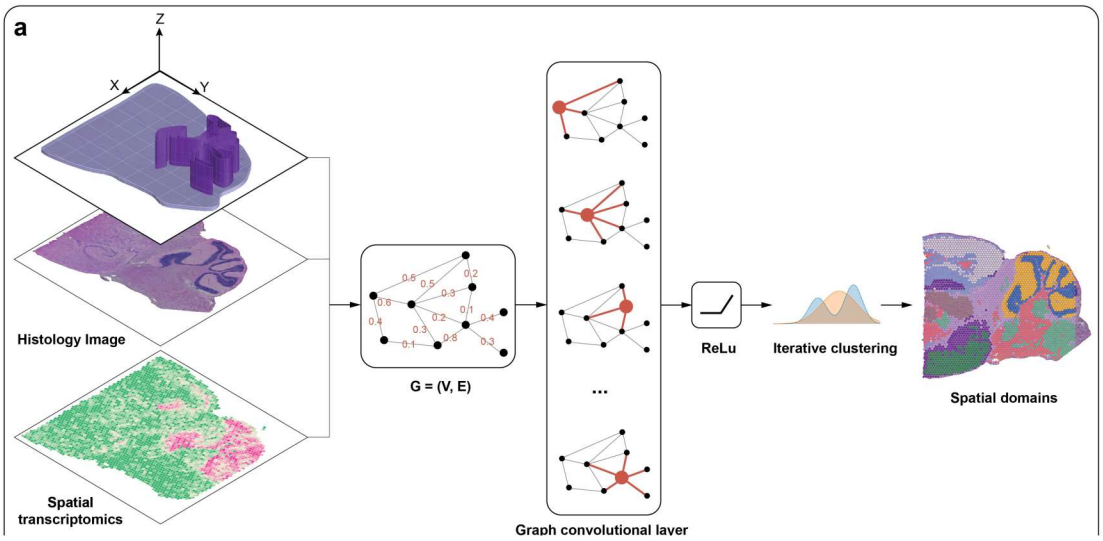
SpaGCN[18] je pristup koji koristi graf konvolucijsku mrežu za integriranje ekspresije gena, prostornu lokaciju i histologiju u analizi tkiva. Kroz konvoluciju grafa, SpaGCN agregira ekspresiju gena svake točke sa susjednima, što omogućuje identifikaciju prostornih domena s koherentnim izrazom i histologijom. Način na koji on to radi prikazan je u slici 3.1.

3.1.1 Konstrukcija grafa

SpaGCN, kao i svaka metoda, svoje podatke tkiva predprocesuira na standardni način: gene koji se pojavljuju u tri ili manje stanica izbacujemo iz matrice X, zatim sve gene u svakoj od stanica podijelimo sa ukupnim brojem gena te stanice, pomnožimo sa brojem 10^4 te prebacimo u log skalu na način

$$X \leftarrow \log(1 + X).$$

Takve podatke $X \in \mathbb{R}^{n \times \hat{m}}$ zajedno sa matricom lokacija $D \in \mathbb{R}^{n \times 2}$ i histologijom $tissue\_img \in \mathbb{R}^{n \times 3}$ pretvara u neusmjereni graf $G(V, E)$. U grafu, svaki vrh $v \in V$



Slika 3.1: Vizualizacija koraka SpaGCN metode. (Slika preuzeta sa SpaGCN github.)

odgovara jednoj točki na tkivu te su svaka dva vrha spojena bridom odgovarajuće težine. Ta težina je definirana kao sličnost povezanih točaka i određuju ju dva faktora:

1. Fizikalna udaljenost točaka tkiva,
2. Odgovarajuće histološke informacije točaka.

Dvije točke smatramo bliskima ukoliko su fizički blizu i imaju slične histološke informacije. Proces računanja težine brida je opisan sljedećim postupkom.

Za svaku točku i na danom tkivu imamo spremljene njene koordinate u pozicijskoj matrici $D_i = (x_i, y_i)$ i, ukoliko je dana, njenu histološku informaciju $tissue\_img_i = (r_i, g_i, b_i)$. Zatim visinu točke i (tj. njezinu z-koordinatu) definiramo kao

$$z_i = \frac{er_i V_r + eg_i V_g + eb_i V_b}{V_r + V_g + V_b}$$

gdje su er_i, eg_i i eb_i srednje vrijednosti r, g i b kanala od pravokutnika veličine 50×50 točaka sa centrom u točki (x_i, y_i) a V_r, V_g i V_b su varijance histologije za sve točke u tkivu. Ovakvom definicijom z_i dobivamo glatkost i osiguravamo da boja nije dominirana jednim pikselom. Zatim, po potrebi, reskaliramo visinu z_i na način

$$z_i^* = s \frac{z_i - \mu_z}{\sigma_z} \max(\sigma_x, \sigma_y)$$

gdje je μ_z očekivanje z; σ_x, σ_y i σ_z standardne devijacije od x, y i z; te $s \in \mathbb{R}$ faktor skaliranja koliko nam je histologija bitna. Ukoliko nam nije dana histologija, z_i^* postavljamo na 0 za sve točke i iz tkiva.

Iz ovakvo definiranih visina prirodno slijedi definicija udaljenosti

$$d(u,v) = \sqrt{(x_u - x_v)^2 + (y_u - y_v)^2 + (z_u^* - z_v^*)^2}$$

koju koristimo za računanje težina bridova. Težine bridova spremamo u matricu susjedstva $A \in \mathbb{R}^{n \times n}$ i računamo ih formulom

$$A_{i,j} = \exp\left(-\frac{d(i,j)^2}{2l^2}\right)$$

gdje je $l \in \mathbb{R}$ odabrani faktor skaliranja. Ovako konstruirana matrica A definira strukturu grafa. Takav graf zajedno sa matricom ekspresije gena X dajemo kao ulazni podatak graf konvolucijskoj mreži na treniranje.

3.1.2 Arhitektura mreže

SpaGCN u svojem procesu raspoznavanja domena tkiva koristi GCN (graf konvolucijsku mrežu) za učenje grafa. Vođeni idejom rada Kipf i Welling [22], ovu mrežu $f(\cdot)$ zapisujemo formulom

$$f\left(\tilde{X}, A\right) = \delta\left(A\tilde{X}B\right),$$

gdje su $\tilde{X} \in \mathbb{R}^{n \times 50}$ matrica dobivena metodom PCA, $A \in \mathbb{R}^{n \times n}$ matrica susjedstva, $B \in \mathbb{R}^{50 \times 50}$ matrica parametara graf konvolucijske mreže $f(\cdot)$, te $\delta(\cdot)$ aktivacijska funkcija ReLU [1].

3.1.3 Treniranje i klasteriranje

SpaGCN koristi nenadzirani iterativni algoritam klasteriranja zvan DEC. DEC algoritam definiran je u radu od Xie et al. [48] a ovdje se samo koristi.
Iz informacija

$$h = f\left(\tilde{X}, A\right) \in \mathbb{R}^{n \times 50}$$

dobivenih od GCN mreže, inicijaliziramo centre klastera μ_j pomoću Louvain metode. Zatim počinje iterativna procedura klasteriranja u kojoj definiramo distribuciju

$$q_{ij} = \frac{\left(1 + \left\|h_i - \mu_j\right\|^2\right)^{-1}}{\sum_{j'=1}^{k}\left(1 + \left\|h_i - \mu_j\right\|^2\right)^{-1}}$$

čiji je kernel Studentova t-distribucija. Ovu distribuciju interpretiramo kao vjerojatnost q_{ij} da točka i pripada domeni j (soft clustering). Zatim kao pomoćnu distribuciju definiramo

$$p_{ij} = \frac{q_{ij}^2 / \sum_{i'=1}^{n} q_{i'j}}{\sum_{j'=1}^{k}\left(q_{ij'}^2 / \sum_{i'=1}^{n} q_{i'j'}\right)}$$

koji povećava težinu mjesta s visokom pouzdanošću i normalizira doprinos svakog centra kako bi spriječili iskrivljavanje prostora od strane velikih klastera. Pomoću tih distribucija definiramo funkciju cilja kao Kullback-Leibler divergenciju[8]

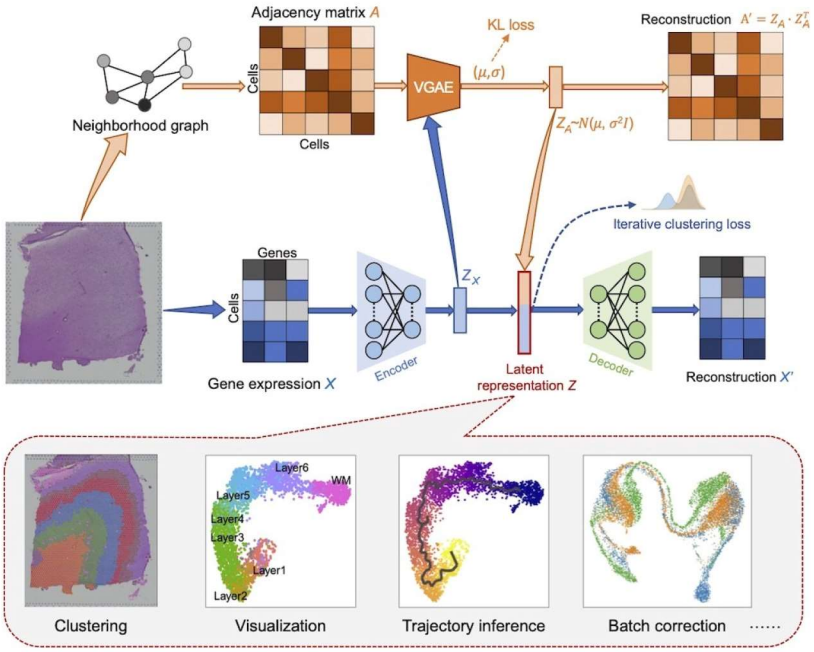
$$\mathrm{KL}\,(P\|Q) = \sum_{i=1}^{n}\sum_{j=1}^{k} p_{ij}\log\frac{p_{ij}}{q_{ij}}.$$

Minimizacijom ove funkcije istovremeno optimiziramo parametre mreže f (matrica B) i centre klastera μ.

Kao dodatni korak, SpaGCN metoda nudi uglađivanje rezultata u kojem sve točke koje su okružene točkama drugog klastera pridruži tom klasteru.

3.2 SEDR

SEDR (Spatially Embedded Deep Representation)[11] je NN metoda koja uči latentnu niže-dimenzionalnu reprezentaciju podataka pomoću auto-enkodera [25]. Za razliku od SpaGCN, SEDR prvo uči nelinearno mapiranje u niže-dimenzionalni prostor, a zatim provodi klasteriranje. Koraci ovog algoritma su slični onima od SpaGCN i ilustrirani su slikom 3.2.



Slika 3.2: Slika prikazuje osnovne korake SEDR metode. (Preuzeto sa SEDR github i promijenjena u skladu s notacijom.)

3.2.1 Konstrukcija grafa

Iz dane matrice pozicija $D \in \mathbb{R}^{n\times 2}$, SEDR izračuna međusobne Euklidske udaljenosti točaka te za svaku odabere 10 najbližih susjeda. Zatim matricu susjedstva

$A \in \mathbb{R}^{n \times n}$ konstruira na način

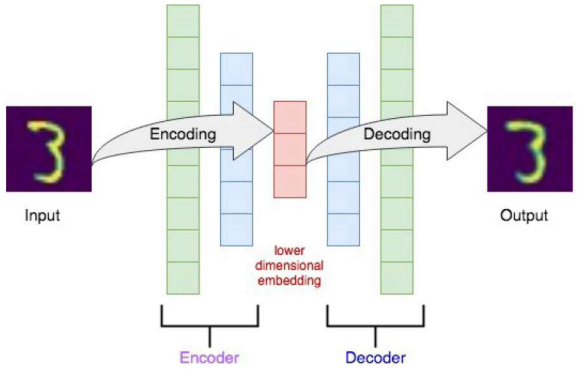
$$A_{ij} = A_{ji} = \begin{cases} 1, & i \text{ je susjed od } j, \\ 0, & \text{inače.} \end{cases} \tag{3.1}$$

3.2.2 Arhitektura mreže i treniranje

Prije nego SEDR započne učiti na podacima, on ih predprocesuira na način da za svaku stanicu iz matrice $X \in \mathbb{N}^{n \times m}$, gene skalira sa brojem ne jako ekspresiranih gena te uzme prvih 200 principalnih komponenti. Zatim takve podatke $\tilde{X} \in \mathbb{R}^{n \times 200}$ zajedno sa matricom susjedstva $A \in \mathbb{R}^{n \times n}$ iz (3.1) proslijedi deep auto-encoder mreži na učenje niže-dimenzionalne reprezentacije podataka.

Deep auto-encoder mreža koju SEDR koristi sastoji se od dva dijela: deep auto-encoder mreže, kojom SEDR uči nižedimenzionalnu reprezentaciju podatak $X \in \mathbb{R}^{n \times m}$, i variational graph auto-encodera (VGAE)[23] kojom mreža uči novu reprezentaciju matrice susjedstva $A \in \mathbb{R}^{n \times n}$ u latentnom niže-dimenzionalnom prostoru. Na slici 3.2, prvi dio mreže označen je plavom bojom a drugi dio mreže narančastom bojom.

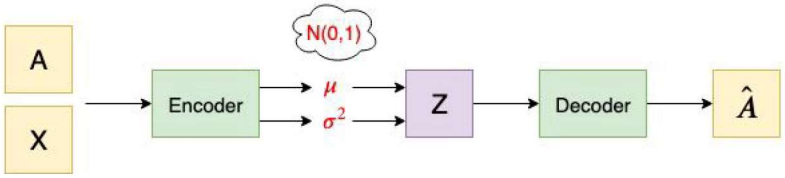
Tradicionalna auto-encoder mreža je skicirana na sljedećoj slici:



Slika 3.3: Oblik tradicionalnog auto-encodera sa MNIST podatkom kao primjer. (Preuzeto sa Tutorial on Variational Graph Auto-Encoders.)

Enkoder dio auto-encoder mreže vraća matricu ugrađivanja $Z_X \in \mathbb{R}^{n \times d_X}$, reprezentiranu crvenom bojom na slici 3.3. Zatim, u ovoj metodi ta reprezentacija podataka se konkatenira sa reprezentacijom $Z_A \in \mathbb{R}^{m \times d_A}$ dobivenom iz drugog dijela mreže (VGAE dio) te ti konkatenirani podaci se nadalje prosljeđuju dekoderu koji pokušava rekonstruirati matricu ekspresije gena $X \in \mathbb{R}^{n \times m}$. Kriterijska funkcija koju ovaj dio mreže minimizira je MSE (mean squared error)[30] funkcija cilja između originalnih podatak X i rekonstruiranih podataka X'.

Drugi dio SEDR-ove mreže je VGAE mreža kojom SEDR pokušava rekonstruirati matricu susjedstva. Općeniti oblik takve mreže skiciran je sljedećom slikom:



Slika 3.4: Slika opisuje općeniti oblik VGAE mreže. (Slika preuzeta sa Tutorial on Variational Graph Auto-Encoders.)

Za danu matricu susjedstva $A \in \mathbb{R}^{n\times n}$ definiranu u (3.1), njenu degree matricu $D \in \mathbb{R}^{n\times n}$

$$D_{ii} = \sum_{j=1}^{n} A_{ij}$$

i matricu $Z_X \in \mathbb{R}^{n\times d_X}$ dobivenu iz prvog dijela, dvoslojni VGAE korišten u ovoj metodi možemo zapisati formulom

$$GCN\left(A, Z_X\right) = \tilde{A}\mathrm{ReLU}\left(\tilde{A}Z_X W_0\right) W_1$$

gdje su W_0 i W_1 parametri mreže, te

$$\tilde{A} = D^{-\frac{1}{2}} A D^{-\frac{1}{2}}.$$

Matricu $\tilde{A}$ možemo shvatit kao normaliziranu matricu A s kojom je numerički stabilnije raditi [9]. Tada rekonstrukciju nižeg ranga matrice A dobivamo formulom

$$A' = Z_A Z_A^T$$

gdje je $Z_A = GCN\left(A, Z_X\right)$. Funkcije cilja koje ova mreža istovremeno optimizira su cross-entropy loss [31] između početne matrice susjedstva A i rekonstruirane matrice $\tilde{A}$; te Kullback-Leibler divergencija između

$$g\left(Z_A \mid A, Z_X\right) = \prod \mathcal{N}\left(z_i \mid \mu_i, \mathrm{diag}\left(\sigma^2\right)\right)$$

i

$$p\left(Z_A\right) = \prod \mathcal{N}\left(z_i \mid 0, I\right),$$

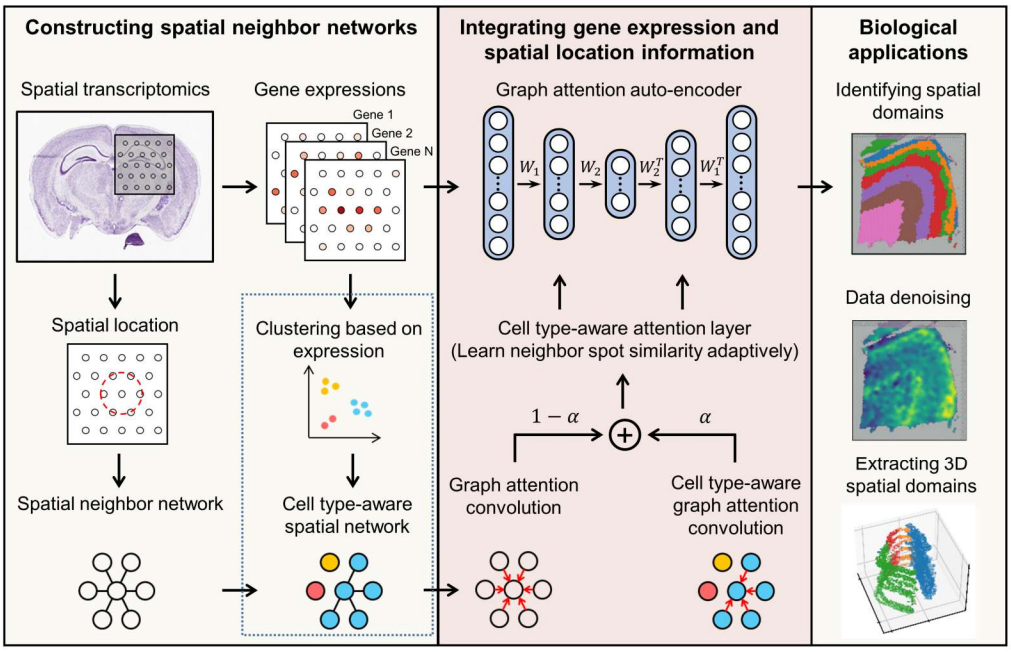
gdje su $\mu = GCN_\mu\left(A, Z_X\right)$ i $\log\sigma = GCN_\sigma\left(A, Z_X\right)$. Ova funkcija cilja je samo preuzeta iz rada *Auto-Encoding Variational Bayes* [21]. Za ovaj poseban slučaj, Kullback-Leibler dio funkcije cilja se može pojednostaviti na

$$\frac{1}{2}\sum_{i=1}^{n}\left(1 + \log\sigma_i^2 - \mu_i^2 - \sigma_i^2\right).$$

SEDR metoda, za razliku od SpaGCN, uči reprezentaciju podataka neovisno o problemu koji rješavamo i time nije vezana za algoritam klasteriranja koji koristimo (iako rezultati mogu znatno varirati). SEDR u svojem procesu koristi Louvain[4] metodu klasteriranje.

3.3 STAGATE

STAGATE (**S**patialy resolved **T**ranscriptomics with an **A**daptive **G**raph **AT**ention auto-**E**ncoder), kao i SEDR, ideja ove metode je naučiti nižedimenzionalnu latentnu reprezentaciju podataka koristeći autoencoder mrežu. Kao nadogradnju na SEDR, STAGATE dodaje attention layer u svrhu adaptivnog učenja težina bridova. Opis mreže dan je slikom 3.5.



Slika 3.5: Slika prikazuje osnovne korake metode STAGATE. (Slika preuzeta sa STAGATE githuba.)

3.3.1 Konstrukcija grafa

U svrhu inkorporiranja informacije sličnosti susjeda, STAGATE pretvara prostornu informaciju u neusmjereni graf u kontekstu unaprijed definiranog geometrijskog radijusa r. Preciznije, matricu susjedstva je definirana na sljedeći način:

$$A_{ij} = \begin{cases} 1, & d(i,j) < r, \\ 0, & \text{inače} \end{cases},$$

gdje je $d(i,j)$ Euklidska udaljenost točke i i točke j.

3.3.2 Arhitektura mreže i treniranje

Prije treniranja STAGATE normalizira dane podatke. Za razliku od ostalih metoda STAGATE za treniranje koristi samo gene sa visokom varijancom. Točnije, iz matrice ekspresije gena X STAGATE izvuče 3000 gena sa najvećom varijancom

i odbaci ostale stupce. Svaki gen svake od stanica podijeli sa zbrojem zadržanih gena te stanice, te pomnoži sa 10000. Dodatno, tu matricu logaritmira na način

$$X \leftarrow \log(1 + X).$$

Graph Attention Auto-Encoder [35] mreža koju STAGATE koristi sastoji se od tri dijela: enkoder, dekoder i attention sloj.

Encoder u arhitekturi uzima normaliziranu matricu ekspresije gena i generira nižedimenzionalnu reprezentaciju agregacijom informacija susjeda. Preciznije, ukoliko podatak x_i zapišemo kao $h_i^{(0)} := x_i$, izlaz na sloju k računa se po formuli

$$h_i^{(k)} = \sigma\left(\sum_{j \in S_i} att_{ij}^{(k)} \left(W_k h_j^{(k-1)}\right)\right)$$

gdje je W_k učiva matrica parametara, σ nelinearna aktivacijska funkcija, S_i susjedstvo točke i i $att_{ij}^{(k)}$ predstavlja attention skalar za dane i, j i k. Tada je nižedimenzionalna reprezentacija od x_i rezultat zadnjeg sloja bez att_{ij} težina, tj.

$$h_i^{(L)} = \sigma\left(W_L h_i^{(L-1)}\right).$$

Dekoder dio mreže je definiran kao komplement enkoder dijelu, tj.

$$\hat{h_i}^{(k-1)} = \sigma\left(\sum_{j \in S_i} \hat{att}_{ij}^{(k)} \left(\hat{W}_k \hat{h_j}^{(k)}\right)\right)$$

i

$$\hat{h_i}^{(0)} = \sigma\left(\hat{W}_1 \hat{h_i}^{(1)}\right).$$

Da bi izbjegao prenaučenost mreže, STAGATE koristi i iste težine u encoder i decoder dijelu mreže, tj. $\hat{W}^{(k)} = W^{(k)}$ i $\hat{att}^{(k)} = att^{(k)}$.

Ideja iza graf attention sloja je implementacija mehanizma koji će omogućiti mreži da na adaptivan način uči sličnost susjeda. Attention je implementiran kao mreža jednog sloja s dijeljenim parametrima među čvorova. Preciznije, u k-tom sloju enkodera, težina brida koji spaja točke i i j se računa na sljedeći način:

$$w_{ij}^{(k)} = \text{Sigmoid}\left({v_s^{(k)}}^T \left(W_k h_i^{(k-1)}\right) + {v_r^{(k)}}^T \left(W_k h_j^{(k-1)}\right)\right)$$

gdje su $v_s^{(k)}$ i $v_r^{(k)}$ učivi parametri mreže. Konačno, da bi izrazili vrijednosti težina za fiksni i i $\forall j \in S_i$ u kontekstu vjerojatnosnog vektora, dodatno ih normaliziramo na sljedeći način:

$$att_{ij}^{(k)} = \frac{\exp\left(w_{ij}^{(k)}\right)}{\sum_{j' \in S_i} \exp\left(w_{ij'}^{(k)}\right)}.$$

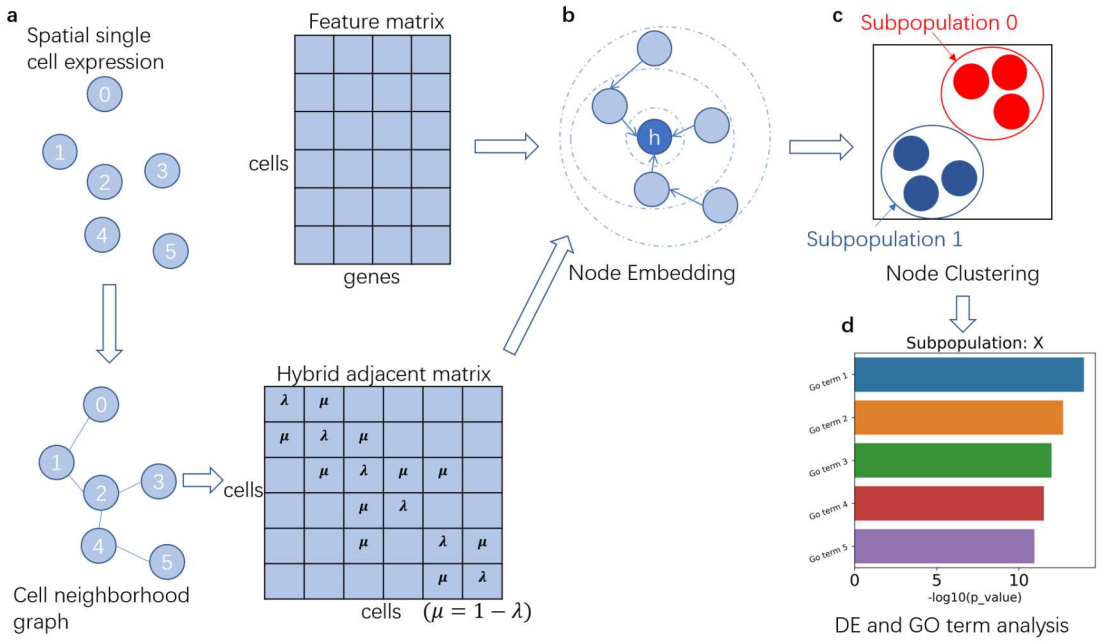
Funkcija cilja koju STAGATE minimizira je

$$\sum_{i=1}^{n} \left\| x_i - \hat{h}_i^{(0)} \right\|^2.$$

Jednako kao i SEDR, STAGATE nije vezan za jednu metodu klasteriranja. Ukoliko je dan broj klastera, STAGATE koristi mclust[37] algoritam dok ukoliko nije nije poznat broj klastera unaprijed, STAGATE primjenjuje Louvain metodu klasteriranja.

3.4 CCST

CCST (Cell clustering for spatial transcriptomics) je GCN pristup ugrađivanja informacija ekspresije gena i globalnih prostornih informacija stanica pomoću minimizacije MI (mutual information)[38] indeksa. Ovaj rad uzima ideju iz prijašnjih radova *Deep Graph Infomax*[16] i *Deep InfoMax*[43] gdje, za razliku od autoenkodera, minimizacijom MI funkcije se uči reprezentacija podataka u nižedimenzionalnom prostoru. Skica koraka koje CCST provodi dana je slikom 3.6.



Slika 3.6: Slika opisuje korake CCST metode redom aKonstrukcija grafa, b Reprezentacija vrhova, c Klasteriranje stanica. (Slika preuzeta sa CCST githuba i pojednostavljena.)

3.4.1 Konstrukcija grafa

Po uzoru na STAGATE, CCST matricu susjedstva konstruira sa

$$A_{0ij} = \begin{cases} 1 & d(i,j) < r \\ 0 & \text{inače} \end{cases}$$

gdje je r unaprijed određen radijus susjedstva. Kako bismo balansirali težinu između prostorne informacije i ekspresije gena, CCST definiran hibridnu matricu udaljensti

$$A = \lambda I + (1 - \lambda) A_0$$

za unaprijed određen λ.

3.4.2 Arhitektura mreže i treniranje

Kao i sve metode dosad, CCST svoje podatke predprocesuira. CCST prvo izbaci sve gene koji se pojavljuju u pet ili manje stanica, normalizira stanice tako da im gene podijeli sa ukupnim zbrojem gena, u koji ne uključi jako ekspresionirane gene, pojedine stanice. Zatim dobivenu matricu normalizira na očekivanje 0 i varijancu 1 te izvuče prvih 200 principalnih komponenata algoritmom PCA koje dalje koristi za učenje svoje mreže.

Mreža korištena u ovom algoritmu je DGI (DeepGraph InfoMax)[43]. DGI mreža se oslanja na maksimiziranje MI između lokalnih reprezentacija i globalnih sadržaja. Takvo grafom izdvojeno lokalno obilježje sadrži informacije o podgrafu koji je centriran u tom čvoru. Da bi bolje istražio high-level značajke, DGI je dizajniran da uči enkoder maksimiziranjem MI po patchevima.

Enkoder te mreže se sastoji od četiri graf konvolucijska sloja koji se mogu zapisati formulom na sljedeći način:

$$GCN^{(l)}\left(H^{(l)}, A\right) = \sigma\left(\tilde{D}^{-1/2}\tilde{A}\tilde{D}^{-1/2}H^{(l)}W^{(l)}\right)$$

gdje su $H^{(l+1)}$ i $H^{(l)}$ ulaz i izlaz l-tog sloja, $W^{(l)}$ učive težine mreže,

$$\tilde{A} = A + I,$$

$$\tilde{D}_{ii} = \sum_j \tilde{A}_{ij},$$

te

$$\sigma(x) = \text{PReLU}(x) = \begin{cases} x, & x \geq 0 \\ ax, & \text{inače} \end{cases}$$

čiji je a učivi parametar. Iz tog mapiranja mreža dalje uči dvije funkcije, funkciju za globalnu reprezentaciju $S : \mathbb{R}^{n \times m} \mapsto \mathbb{R}^m$ i diskriminator funkciju $D : \mathbb{R}^m \times \mathbb{R}^m \mapsto \mathbb{R}$. Diskriminator funkcija D je uvedena kao evaluacijska funkcija kolike informacije grafa se nalazi u lokalnom patchu. Veći broj $D(h_i, s)$ navodi da će

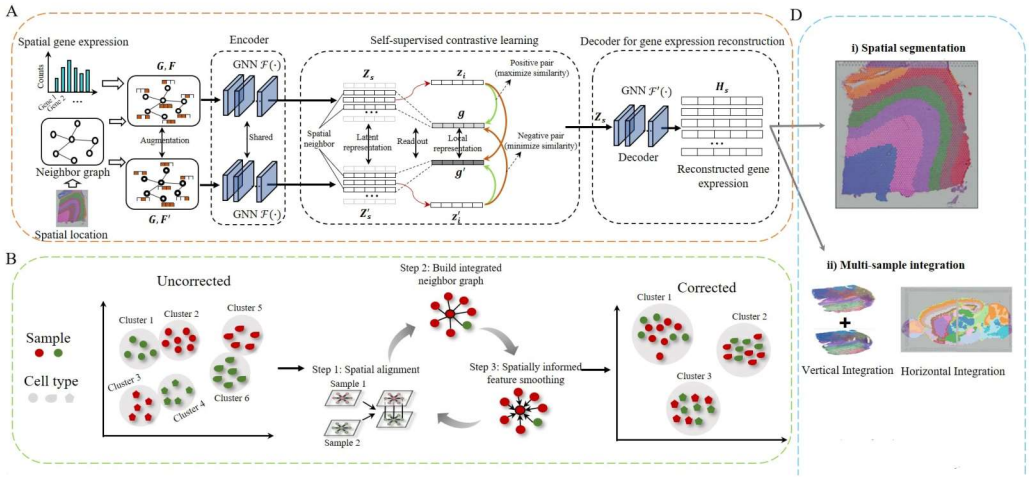
ovi patchevi s većom vjerojatnošću biti sadržani u globalnoj reprezentaciji. Za treniranje diskriminatora generiramo negativne podatke korupcijskom funkcijom $\bar{A} = C(A)$ koja permutira bridove grafa. Time dobivamo reprezentaciju $\bar{h}_i$ za negativan podatak. Tada funkcija cilja koju algoritam maksimizira je

$$\sum_{i=1}^{n} E\left[\log D\left(h_i, s\right)\right] + E\left[\log\left(1 - D\left(\bar{h}_i, s\right)\right)\right].$$

Optimiziranje te funkcije, DGI kao izlaz daje reprezentciju vrhova koja sadrži strukturalne informacije grafa. Time, kao i STAGATE i SEDR, nije vezana za metodu klasteriranja. CCST za klasteriranje koristi k-means++ [3] algoritam na podacima u konačnici dobivenim PCA procesuiranjem podataka prije toga preslikanim gore definiranom neuronskom mrežom.

3.5 DeepST

DeepST, novi graf contrastive samonadzirani okvir učenja, kombinira prijašnje metode u jednu. Contrastive learning pristupom ova metoda pokušava upotpunosti uhvatiti prostorne infromacije zajedno sa ekspresijom gena te dodatno pokušava ukloniti negativne efekte batcheva. Tok rada algoritma je rezimiran u slici
.



Slika 3.7: Slika prikazuje DeepST algoritam. (Slika preuzeta sa DeepST githuba.)

3.5.1 Konstrukcija grafa

DeepST matricu susjedstva konstruira na način

$$A_{ij} = \begin{cases} 1, & i \text{ je jedan od } k \text{ najbližih susjeda od } j \\ 0, & \text{inače.} \end{cases}$$

Time DeepST inkorporira prostornu informaciju u svoj proces grupiranja.

3.5.2 Arhitektura mreže i treniranje

Augumentacija podataka je bitna u kontrastivnom učenju. Ova metoda prvo konstruira negativne podatke na način da za dane podakte $X \in \mathbb{R}^{n \times m}$ i matricu susjedstva $A \in \mathbb{R}^{n \times n}$ stvori negativne podatke $X' \in \mathbb{R}^{n \times m}$ permutacijom ekspresije gena. Tada par (X', A) ne negativan podatak pridružen podatku X. Tako generirane negativne podatke zajedno sa originalnim podacima prosljđeujemo autoenkoder neuronskoj mreži.

Enkoder dio mreže se sastoji od GCN slojeva oblika

$$Z^l = \sigma\left(\tilde{A} Z^{l-1} W_e^{l-1}\right)$$

gdje su, kao i u prošlim metodama, $\tilde{A} = D^{-\frac{1}{2}} A D^{-\frac{1}{2}}$ normalizirana matrica susjedstva, $D_{ii} = \sum_{j=1}^{n} A_{ij}$ diagonalna matrica stupnjeva, W_e^l učivi parametri mreže te $\sigma(\cdot)$ nelinearna aktivacijska funkcija ReLU. Z^0 postavljamo kao ulaz mreže X a ostali Z^l predstavljaju izlaz l-tog sloja. Izlaz zadnje sloja mreže smatramo nižedimenzionalnom reprezentacijom danih podataka.

Dekoder dio mreže poprima simetričnu strukturu enkoder dijelu, to jest oblika je

$$H^t = \sigma\left(\tilde{A} H^{t-1} W_d^{t-1}\right)$$

gdje su sada W_d^t drugi učivi parametri mreže, $H^0 = Z^L$ izlaz enkodera i H^t izlazi odgovarajućih slojeva dekodera. Ovako konstruiranom mrežu učimo da minimizira grešku rekonstrukcije

$$L_{recon} = \sum_{i=1}^{n} \left\| x_i - \hat{h_i} \right\|.$$

Dani izlaz dekoder mreže kasnije koristimo za određivanje domena tkiva.

Za usavršavanje reprezentacije, DeepST koristi kontrastiv samonadzirani mehanizam učenja u kojem uči readout funkciju $R : \mathbb{R}^{n \times d} \mapsto \mathbb{R}^d$ zajedno sa diskriminator funkcijom $D : \mathbb{R}^d \times \mathbb{R}^d \mapsto \mathbb{R}$. Kao i u metodi CCST, uloga readout funkcije R je naučiti globalnu reprezentaciju podatak koja inkorporira važne spacijalne podatke i podatke gena, dok uloga diskriminatora D je naučit razaznati korisnost svakog od podatka. Formalno funkcija cilja koju ova mreža uči minimizirat je dana izrazom

$$L_{CSL} = -\frac{1}{2n}\left(\sum_{i=1}^{n} \mathbb{E}\left[\log D\left(z_i, g\right)\right] + \mathbb{E}\left[\log\left(1 - D\left(z_i', g\right)\right)\right]\right).$$

Ugrađeni podaci z_i i z_i' su dobiveni prolaskom originalnih i njegovih negativnih podatak kroz enkoder mreže dok je $g = R(Z)$ lokalni kontekst podataka. Da

bismo model napravili stabilnijim i balansiranijmi, uvodimo negativan kontekst $g' = R(Z')$ nad kojim definiramo funkciju

$$L_{CLS'} = -\frac{1}{2n}\left(\sum_{i=1}^{n}\mathbb{E}\left[\log D\left(z'_i, g'\right)\right] + \mathbb{E}\left[\log\left(1 - D\left(z_i, g'\right)\right)\right]\right)$$

koju istovremeno sa L_{CLS} pokušavamo minimizirati.

Kompletna funkcija koju DeepST uči minimizirat je

$$L = \lambda_1 L_{recon} + \lambda_2\left(L_{CLS} + L_{CLS'}\right) + \lambda_3\left(\|W_e\|_F^2 + \|W_d\|_F^2\right)$$

gdje su λ_1, λ_2, λ_3 podesivi težinski parametri a treći član je uveden da izbjegnemo prekotrenirajne mreže.

3.5.3 Klasteriranje

Za razliku od dosadašnjih embedding pristupa, DeepST za klasteriranje tkiva koristi izlaz H iz dekoder dijela mreže. Algoritam koji koristi je mclust, nespacijalni algoritham dodjeljivanja. Ukoliko broj klastera nije unaprijed poznat, izabire se podijela s najvećim Silhoutte rezultatom[34]. Vođeni pretpostavkom da domene tkiva su prostorno povezane DeepST, kao opcionalan, korak nudi doradu klasteriranja. Za dani radijus parametar r, promatramo točku i sve njene susjede udaljene najviše za r od nje. Tada ukoliko više od pola susjeda pripada istom klasteru, promatrana točka je ponovno dodijeljena tom dominantnom klasteru.

4 | Eksperimenti

U ovom poglavlju pokazat ćemo eksperimente provedene na NN metodama. Za svaku od metoda uzet je službeni tutorial s githuba i pokrenut je na odabrana dva skupa podataka. Prvi skup podataka DLPFC (human dorsolateral prefrontal cortex) [24] je odabran jer je bio zajednički ovim metodama za pokazivanje rezultata. Drugi skup podataka generiran je algoritmom za simuliranje ekspresije gena. Izgled generiranih podatak je jednostavnijeg oblika ali se kao takav može naći i u prirodi. Prostorna organizacija mozga temeljno je povezana s njegovom funkcijom. Ovaj odnos strukture i funkcije posebno je očit u kontekstu laminarne organizacije ljudskog cerebralnog korteksa, u kojem stanice koje se nalaze unutar različitih kortikalnih slojeva pokazuju različite obrasce ekspresije gena i pokazuju različite obrasce morfologije, fiziologije i povezanosti. [1]

4.1 DLPFC

Prostorna organizacija mozga temeljno je povezana s njegovom funkcijom. Ovaj odnos strukture i funkcije posebno je očit u kontekstu laminarne organizacije ljudskog cerebralnog korteksa, u kojem stanice koje se nalaze unutar različitih kortikalnih slojeva pokazuju različite obrasce ekspresije gena i pokazuju različite obrasce morfologije, fiziologije i povezanosti.

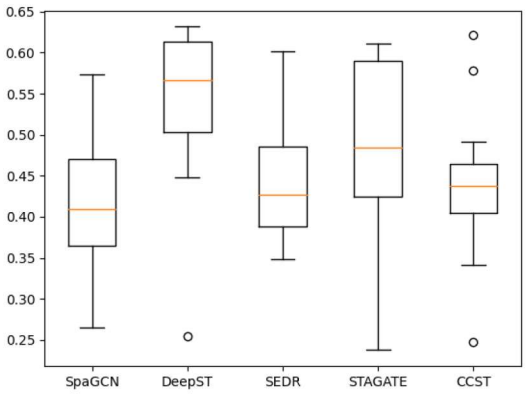
Ovi podaci su profilirani iz ljudskog postmortalnog DLPFC [28] tkiva iz dva para "prostornih replika" od tri neovisna neurotipična odrasla donora. Svaki se par sastojao od dva izravno susjedna serijska odsječka tkiva od 10 μm s drugim parom smještenim 300 μm posteriorno od prvog, što je rezultiralo s ukupno 12 uzoraka izvedenih na platformi Visium. Shema **b** na slici 1.1.

Svaka od gore navedenih NN metoda ima službeni github repozitorij sa kodovima korištenim u radovima. Te python [42] skripte smo preuzeli i pokrenuli na lokalnom računalu čije rezultate smo dalje međusobno uspoređivali. Rezultate zajedno sa traženom podijelom tkiva smo prikazali pomoću python paketa matplotlib [19]. Uzorak sa nazivom 151673 prikazan je na slici 4.2f a metode primjenjene na njemu su prikazane u slikama 4.2. Rezultati dobiveni na svih 12 uzoraka sažeti su u slici 4.1

Kao dodatni eksperiment, za svaku od metoda ugrađivanja pokrenuli smo

[1] Sav kod korišten u eksperimentima nalazi se na githubu.

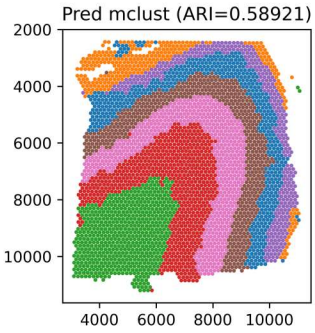
četiri različita algoritma klasteriranja. Za k-means [47] algoritam klasteriranja uzeta je implementacija scikit-learn [29, 5]. Za louvain i leiden algoritme uzeta je implementacija scanpy pythonovog paketa za koje smo tražili rezoluciju da dobijemo točan broj klastera. Mclust algoritam klasteriranja posuđujemo iz jezika R [32] te ga koristimo s parametrima koje STAGATE metoda preporučuje. Rezultati na uzorku 151673 mogu se vidjeti u tablici 4.1.



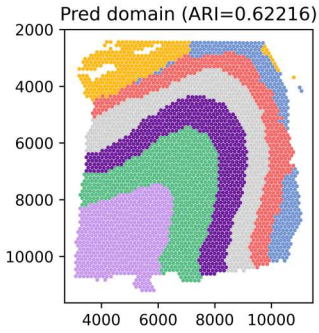
Slika 4.1: Slika prikazuje kutijatisti dijagram [26] sa ARI [33] rezultatima svih metoda na DLPFC podacima.

4.2 Simulirani podaci

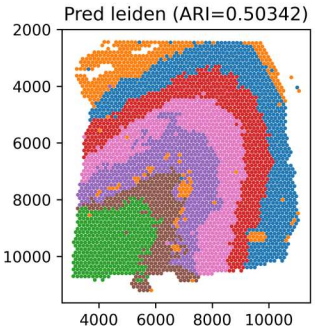
Simulirani podaci generirani su pomoću algoritma cell2location[24]. Za svaki uzorak simuliranog tkiva generirano je 12 okomitih traka koji predstavljaju domene. Dodatno, motivirano SpiceMixom [7], perturbiran je određen broj stanica uz rubove traka. Primjer tako generiranog tkiva dano je na slici 4.3f. Istim postupkom kao i kod DLPFC podataka, obrađeni su simulirani podaci i rezultati su prikazani na slici 4.4 te uspoređeni u tablici 4.2.



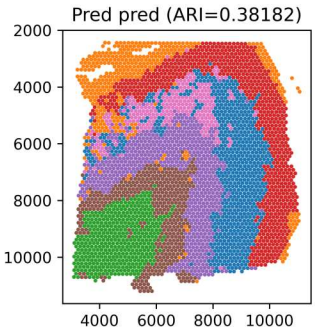
(a) STAGATE na 151673



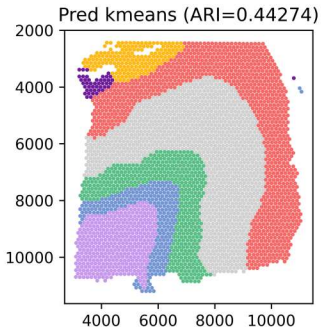
(b) DeepST na 151673



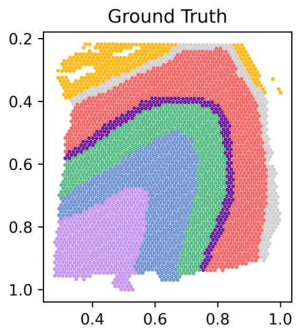
(c) SEDR na 151673



(d) SpaGCN na 151673

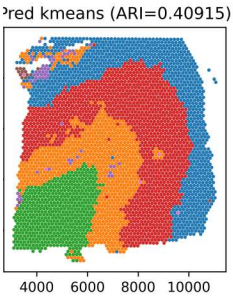
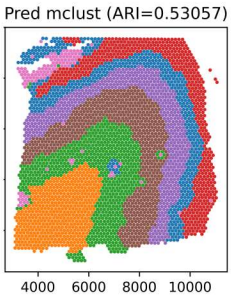
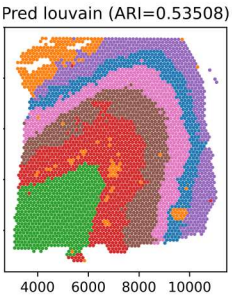
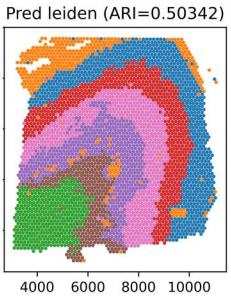
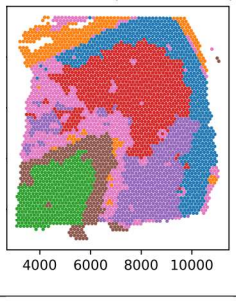
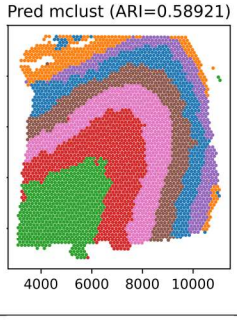
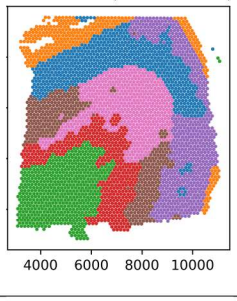
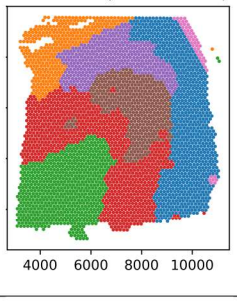
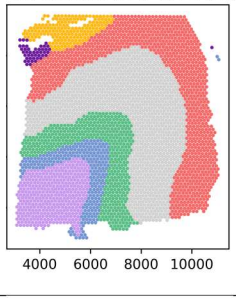
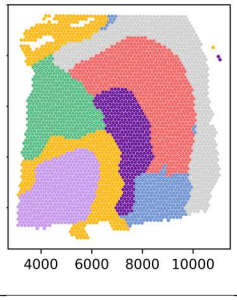
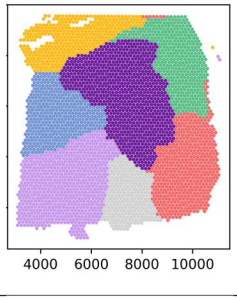
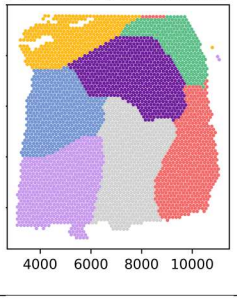
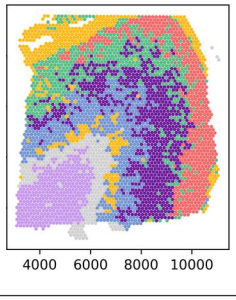
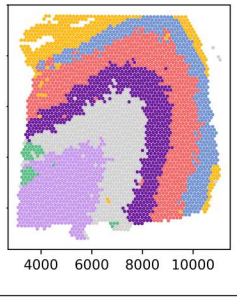
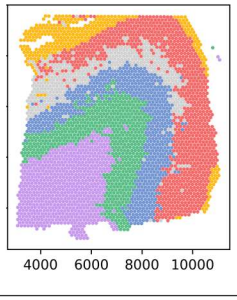
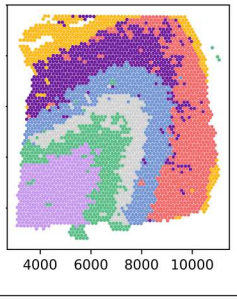


(e) CCST na 151673

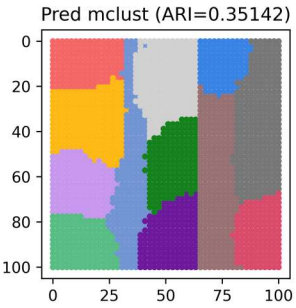


(f) Original 151673

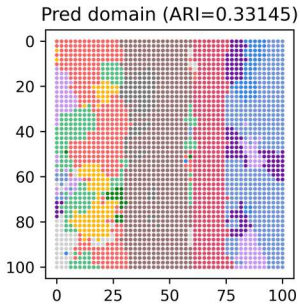
Slika 4.2: Slike su dobivene primjenom svakog od algoritama sa originalnim postavkama na primjeru 151673 iz DLPFC skupa podataka

	k-means	mclust	louvain	leiden
SEDR				
STAGATE				
CCST				
DeepST				

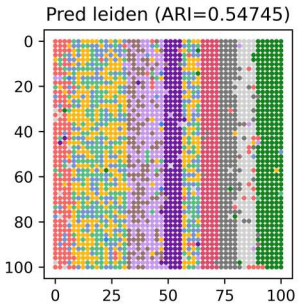
Tablica 4.1: Četiri embedding metode sa različitim algoritmima klasteriranja na uzorku 151673.



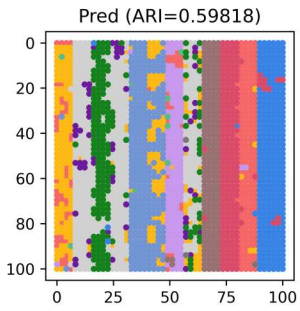
(a) STAGATE



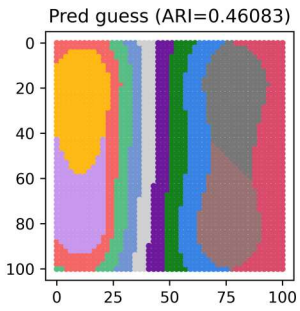
(b) DeepST



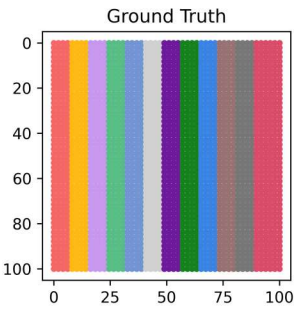
(c) SEDR



(d) SpaGCN

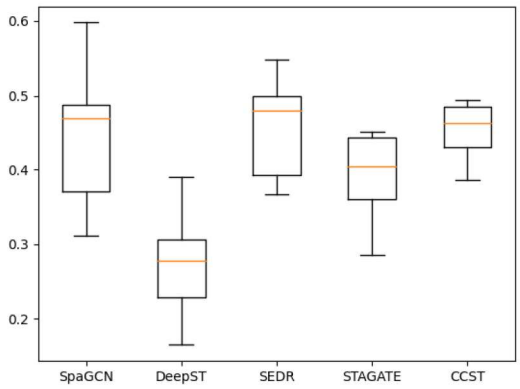


(e) CCST

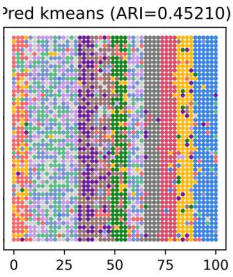
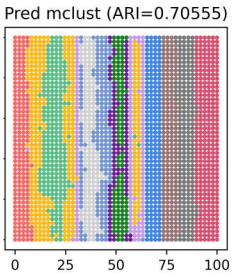
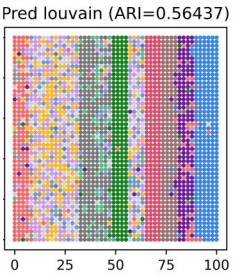
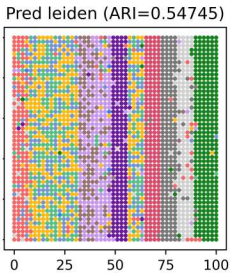
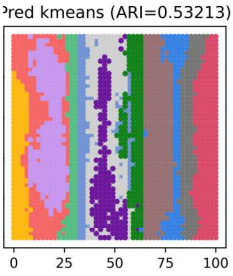
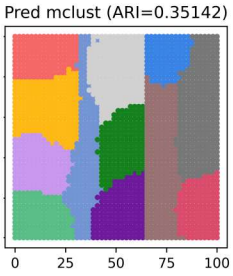
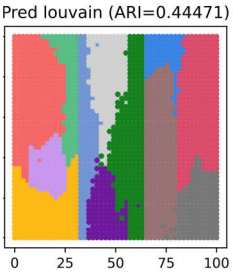
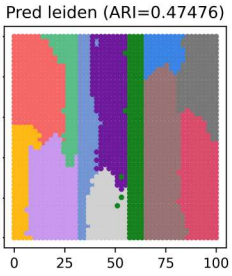
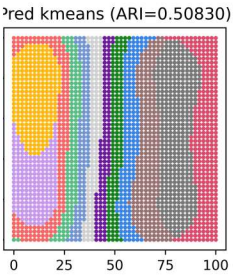
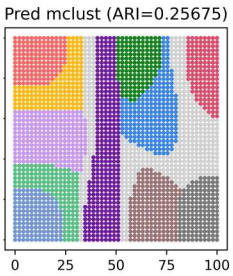
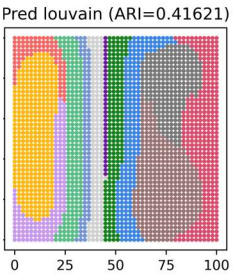
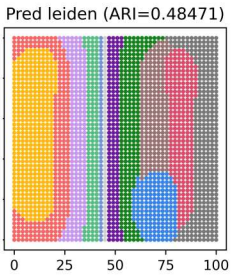
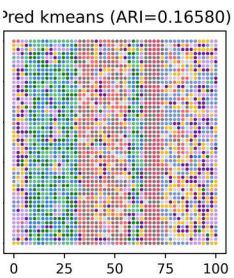
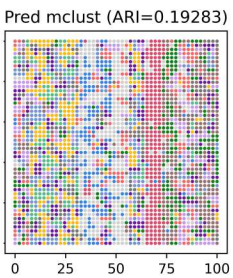
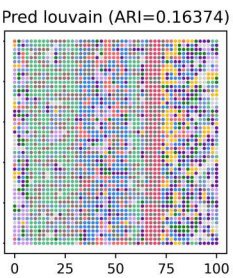
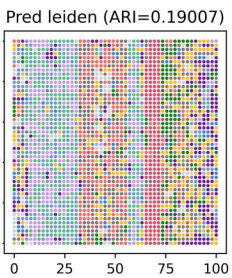


(f) Original

Slika 4.3: Slike su dobivene primjenom svakog od algoritama sa originalnim postavkama na istom simuliranom primjeru.



Slika 4.4: ARI rezultati svih od metoda na simuliranim podacima

	k-means	mclust	louvain	leiden
SEDR				
STAGATE				
CCST				
DeepST				

Tablica 4.2: Četiri embedding metode sa različitim algoritmima klasteriranja na simuliranim podacima.

5 | Diskusija rezultata

Spatial Transcriptomics je moderna tehnologija. Ideja da se pomoću genskih i prostornih informacija klasterira tkivo je nova i stoga se još nisu stigle razviti metode koje će dobro riješiti ovaj navedeni problem. S time na umu, u ovom odlomku ćemo prokomentirati opisane algoritme i rezultate dobivene njima, te "znanstvene" radove koji su ih inspirirali. Tekst koji slijedi dijelom je subjektivan, a dijelom rezultat provedenih eksperimenata.

Nijedna od metoda ne daje prihvatljive rezultate. Iako slike 4.2 izgledaju slično originalu, ARI indeks nam govori suprotno. Svaka od metoda je u svom radu tvrdila da je bolja od ostalih i uspoređivali su se na primjeru DLPFC podataka. Ti rezultati nisu bili konzistentni u drugim radovima i čini se da su često odabrani hiperparametri metode "namješteni" tako da odgovaraju pojedinim ulaznim podacima. Na dijagramu 4.1 vidimo da metoda DeepST, metoda koja objedinjuje sve ideje od prijašnjih, je generalno bolja od ostalih na primjeru DLPFC. Simulirani podaci pokazuju suprotno. Najjednostavnija metoda SpaGCN daje najbolje rezultate na simuliranim podacima dok CCST metoda je najsigurnija (ARI rezultati ne variraju znatno).

Kompleksnost DeepST metode ne garantira dobre rezultate te ideja objedinjavanja prijašnjih ideja ne pokazuje se uvijek najboljim izborom. Riješiti problem na jednom primjeru nije teško, što možemo vidjeti kod metode RESEPT [6].

S druge strane, imamo jednostavne metode poput SpaGCN pristupa. Svojim jednim slojem, reduciranjem dimenzije na 50 (PCA) i fiksiranjem *seed*-a[1] ne daje puno prostora za napredak. Naime, autori originalnog DEC algoritma [48] posvećuju puno vremena "pametnoj" inicijalizaciji NN, dok se kod autora SpaGCN-a "pametna" inicijalizacija NN svodi na odabir "najboljeg" *random seed*-a. Kao što vidimo na slikama 4.1 i 4.4, njeni rezultati dosta variraju. Svojom nasumičnom inicijalizacijom i pogreškom u kodu[2] ulijevaju dodatnu nesigurnost u valjanost metode na različitim ulaznim podacima.

SEDR metoda, kao i DeepST metoda, dodatno komplicira svoju funkciju cilja. Računanje niskodimenzionalne reprezentacije grafa i ekspresije gena te njihova konkatenacija uvodi dodatne parametre koje treba pažljivo odabrati za

[1] Zadavanjem *seed* unutar funkcije globalno postavlja seed. github

[2] Računanje Studentove t-distribucije ne odgovara formuli napisanoj u radu. github

učenje modela. Uočimo na slikama 4.2, 4.1 i 4.2, da SEDR metoda nije odabrala najbolju metodu klasteriranja za nju (što bi bilo mclust). Stoga možda i slika 4.4 nije dobra usporedba za metodu SEDR.

STAGATE metoda, iako se nije previše proslavila u ovim eksperimentima, ima jednostavnost ideje i korištenja koja ulijeva sigurnost u korisnika. Svojim medijanom koji dominira u slučaju DLPFC podataka (4.1) i najvećim medijanom na simuliranim podacima (4.4) empirijski podupire ovu tvrdnju. Učenjem susjedstva dodaje nelinearnost mreži i oslobađa korisnika od naštimavanja hiperparametara. Kao metoda dobra je za biologe za koje je napravljena.

CCST metoda se pokazala dobrom u ovim eksperimentima. Prednost ove metode je što koristi moderne ideje a nedostatak je što zahtjeva vremena i procesorske snage da da rezultate. Jednako tako i naštimavanje parametara nije intuitivno što može dovesti do nepotrebnog gubitka vremena ukoliko korisnik hoće bolje rezultate. Ova metoda ima prostora za poboljšanje i daljnjim usavršavanjem bi mogla biti bolja od ostalih.

Literatura

[1] Agarap, A. F. Deep learning using rectified linear units (relu). *arXiv preprint arXiv:1803.08375* (2018).

[2] Altman, N. S. An introduction to kernel and nearest-neighbor nonparametric regression. *The American Statistician 46*, 3 (1992), 175–185.

[3] Arthur, D., and Vassilvitskii, S. K-means++: The advantages of careful seeding. In *Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms* (USA, 2007), SODA '07, Society for Industrial and Applied Mathematics, p. 1027–1035.

[4] Blondel, V. D., Guillaume, J.-L., Lambiotte, R., and Lefebvre, E. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment 2008*, 10 (oct 2008), P10008.

[5] Buitinck, L., Louppe, G., Blondel, M., Pedregosa, F., Mueller, A., Grisel, O., Niculae, V., Prettenhofer, P., Gramfort, A., Grobler, J., Layton, R., VanderPlas, J., Joly, A., Holt, B., and Varoquaux, G. API design for machine learning software: experiences from the scikit-learn project. In *ECML PKDD Workshop: Languages for Data Mining and Machine Learning* (2013), pp. 108–122.

[6] Chang, Y., He, F., Wang, J., Chen, S., Li, J., Liu, J., Yu, Y., Su, L., Ma, A., Allen, C., Lin, Y., Sun, S., Liu, B., Otero, J., Chung, D., Fu, H., Li, Z., Xu, D., and Ma, Q. Define and visualize pathological architectures of human tissues from spatially resolved transcriptomics using deep learning. *bioRxiv* (2021).

[7] Chidester, B., Zhou, T., Alam, S., and Ma, J. Spicemix: Integrative single-cell spatial modeling of cell identity. *bioRxiv* (2022).

[8] Csiszar, I. *I*-Divergence Geometry of Probability Distributions and Minimization Problems. *The Annals of Probability 3*, 1 (1975), 146 – 158.

[9] Dubois, Y. Interpretation of symmetric normalised graph adjacency matrix? https://math.stackexchange.com/questions/3035968/interpretation-of-symmetric-normalised-graph-adjacency-matrix. Accessed: 2022-09-09.

[10] F.R.S., K. P. Liii. on lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science 2*, 11 (1901), 559–572.

[11] Fu, H., Xu, H., Chong, K., Li, M., Ang, K. S., Lee, H. K., Ling, J., Chen, A., Shao, L., Liu, L., and Chen, J. Unsupervised spatially embedded deep representation of spatial transcriptomics. *bioRxiv* (2021).

[12] George, E. I., Casella, G., and George, E. I. Explaining the gibbs sampler. *The American Statistician* (1992).

[13] Hao, Y., Hao, S., Andersen-Nissen, E., III, W. M. M., Zheng, S., Butler, A., Lee, M. J., Wilk, A. J., Darby, C., Zagar, M., Hoffman, P., Stoeckius, M., Papalexi, E., Mimitou, E. P., Jain, J., Srivastava, A., Stuart, T., Fleming, L. B., Yeung, B., Rogers, A. J., McElrath, J. M., Blish, C. A., Gottardo, R., Smibert, P., and Satija, R. Integrated analysis of multimodal single-cell data. *Cell* (2021).

[14] Hao, Y., Hao, S., Andersen-Nissen, E., Mauck, W. M., Zheng, S., Butler, A., Lee, M. J., Wilk, A. J., Darby, C., Zager, M., Hoffman, P., Stoeckius, M., Papalexi, E., Mimitou, E. P., Jain, J., Srivastava, A., Stuart, T., Fleming, L. M., Yeung, B., Rogers, A. J., McElrath, J. M., Blish, C. A., Gottardo, R., Smibert, P., and Satija, R. Integrated analysis of multimodal single-cell data. *Cell 184*, 13 (2021), 3573–3587.e29.

[15] Hastings, W. K. Monte Carlo sampling methods using Markov chains and their applications. *Biometrika 57*, 1 (04 1970), 97–109.

[16] Hjelm, R. D., Fedorov, A., Lavoie-Marchildon, S., Grewal, K., Bachman, P., Trischler, A., and Bengio, Y. Learning deep representations by mutual information estimation and maximization, 2018.

[17] Hopfield, J. J. Neural Networks and Physical Systems with Emergent Collective Computational Abilities. *Proceedings of the National Academy of Science 79*, 8 (Apr. 1982), 2554–2558.

[18] Hu, J., Li, X., Coleman, K., Schroeder, A., Ma, N., Irwin, D. J., Lee, E. B., Shinohara, R. T., and Li, M. Spagcn: Integrating gene expression, spatial location and histology to identify spatial domains and spatially variable genes by graph convolutional network. *Nature Methods 18*, 11 (Nov 2021), 1342–1351.

[19] Hunter, J. D. Matplotlib: A 2d graphics environment. *Computing in Science & Engineering 9*, 3 (2007), 90–95.

[20] Kindermann, R., and Snell, J. L. *Markov random fields and their applications*, vol. 1 of *Contemporary Mathematics*. American Mathematical Society, Providence, R.I., 1980.

[21] Kingma, D. P., and Welling, M. Auto-encoding variational bayes, 2013.

[22] Kipf, T. N., and Welling, M. Semi-supervised classification with graph convolutional networks. *CoRR abs/1609.02907* (2016).

[23] Kipf, T. N., and Welling, M. Variational graph auto-encoders, 2016.

[24] Kleshchevnikov, V., Shmatko, A., Dann, E., Aivazidis, A., King, H. W., Li, T., Elmentaite, R., Lomakin, A., Kedlian, V., Gayoso, A., Jain, M. S., Park, J. S., Ramona, L., Tuck, E., Arutyunyan, A., Vento-Tormo, R., Gerstung, M., James, L., Stegle, O., and Bayraktar, O. A. Cell2location maps fine-grained cell types in spatial transcriptomics. *Nature Biotechnology 40*, 5 (May 2022), 661–671.

[25] Liou, C.-Y., Cheng, W.-C., Liou, J.-W., and Liou, D.-R. Autoencoder for words. *Neurocomputing 139* (2014), 84–96.

[26] McGill, R., Tukey, J. W., and Larsen, W. A. Variations of box plots. *The American Statistician 32*, 1 (1978), 12–16.

[27] McInnes, L., Healy, J., and Melville, J. Umap: Uniform manifold approximation and projection for dimension reduction, 2018.

[28] Pardo, B., Spangler, A., Weber, L. M., Hicks, S. C., Jaffe, A. E., Martinowich, K., Maynard, K. R., and Collado-Torres, L. spatiallibd: an r/bioconductor package to visualize spatially-resolved transcriptomics data. *BMC Genomics* (2022).

[29] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research 12* (2011), 2825–2830.

[30] Pishro-Nik, H. *Introduction to Probability, Statistics, and Random Processes*. Kappa Research, LLC, 2014.

[31] PyTorch. Pytorch binary cross-entropy with logits. https://pytorch.org/docs/stable/generated/torch.nn.BCEWithLogitsLoss.html. Accessed: 2022-09-09.

[32] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2022.

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

[34] Rousseeuw, P. J. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics 20* (1987), 53–65.

[35] Salehi, A., and Davulcu, H. Graph attention auto-encoders, 2019.

[36] Satija, R., Farrell, J. A., Gennert, D., Schier, A. F., and Regev, A. Spatial reconstruction of single-cell gene expression data. *Nature Biotechnology 33* (2015), 495–502.

[37] Scrucca, L., Fop, M., Murphy, T. B., and Raftery, A. E. mclust 5: clustering, classification and density estimation using Gaussian finite mixture models. *The R Journal 8*, 1 (2016), 289–317.

[38] Shannon, C. E. A mathematical theory of communication. *The Bell System Technical Journal 27*, 3 (1948), 379–423.

[39] Stuart, T., Butler, A., Hoffman, P., Hafemeister, C., Papalexi, E., III, W. M. M., Hao, Y., Stoeckius, M., Smibert, P., and Satija, R. Comprehensive integration of single-cell data. *Cell 177* (2019), 1888–1902.

[40] Teng, H., Yuan, Y., and Bar-Joseph, Z. Clustering spatial transcriptomics data. *Bioinformatics 38*, 4 (10 2021), 997–1004.

[41] van der Maaten, L., and Hinton, G. Visualizing data using t-sne. *Journal of Machine Learning Research 9*, 86 (2008), 2579–2605.

[42] Van Rossum, G., and Drake, F. L. *Python 3 Reference Manual*. CreateSpace, Scotts Valley, CA, 2009.

[43] Veličković, P., Fedus, W., Hamilton, W. L., Liò, P., Bengio, Y., and Hjelm, R. D. Deep graph infomax, 2018.

[44] Wikipedia. Cluster analysis — Wikipedia, the free encyclopedia. http://en.wikipedia.org/w/index.php?title=Cluster%20analysis&oldid=1104837324, 2022. [Online; accessed 08-September-2022].

[45] Wikipedia. RGB color model — Wikipedia, the free encyclopedia. http://en.wikipedia.org/w/index.php?title=RGB%20color%20model&oldid=1110525189, 2022. [Online; accessed 20-September-2022].

[46] Wolf, F. A., Angerer, P., and Theis, F. J. Scanpy: large-scale single-cell gene expression data analysis. *Genome Biology 19*, 1 (Feb 2018), 15.

[47] Wu, J. *Advances in K-means clustering: a data mining thinking*. Springer Science & Business Media, 2012.

[48] Xie, J., Girshick, R. B., and Farhadi, A. Unsupervised deep embedding for clustering analysis. *CoRR abs/1511.06335* (2015).

[49] Yamao Chen, Shengyu Zhou, M. L. e. a. Steel enables high-resolution delineation of spatiotemporal transcriptomic data. *Research Square* (Januray 2022).

[50] Zhang, H., Hunter, M. V., Chou, J., Quinn, J. F., Zhou, M., White, R., and Tansey, W. Bayestme: A unified statistical framework for spatial transcriptomics. *bioRxiv* (2022).

[51] Zhao, E., Stone, M. R., Ren, X., Guenthoer, J., Smythe, K. S., Pulliam, T., Williams, S. R., Uytingco, C. R., Taylor, S. E. B., Nghiem, P., Bielas, J. H., and Gottardo, R. Spatial transcriptomics at subspot resolution with bayesspace. *Nature Biotechnology 39*, 11 (Nov 2021), 1375–1384.

Sažetak

U ovom radu se bavimo metodama za klasteriranje tkiva baziranim na neuronskim mrežama. Uvodimo problem klasteriranja tkiva, navodimo neke od state-of-the-art metoda za rješavanje tog problema te detaljnije objašnjavamo metode bazirane na neuronskim mrežama.

Ključne riječi

Spacial transcriptomics, neuronek mreže, klasteriranje, strojno učenje, biologija.

Neural network methods for spacial transcriptomics clustering

Summary

In this paper, we consider the problem of clustering spatial domains for certain tissue samples. We list some of the state-of-the-art methods that are used in solving this problem and then explain in more detail the ideas behind the neural network methods that we mention. To show the capability of each NN method, here we provide the results of some experiments.

Keywords

Spacial transcriptomics, neural network, clustering algorithms, machine learning, biology.

Životopis

Rođen sam u Osijeku 1998. Godine 2017. završavam srednjoškolsko obrazovanje u III. gimnaziji Osijek te upisujem sveučilišni preddiplomski studij Matematike i Računarstva na Odjelu za matematiku Sveučilišta Josipa Jurja Strossmayera u Osijeku. Preddiplomski studij završavam 2020. godine nakon čega odmah upisujem sveučilišni diplomski studij Matematike, smjer: Matematika i Računarstvo na istom odjelu.