Assignment 3

Includes methods of Directed Graphical Models, Expectation Maximisation, Dynamic programming, Variational Inference, Hidden Markov Models and Spectral Graph Analysis

DD2434 Advanced Machine Learning
Filip Bergentoft, bergento@kth.se

3.1 Easier EM for Advertisements

Submitted a solution to the original version of 3.1 before it was simplified and received full points plus $\frac{1}{4}$ bonus points for my solution.

3.2 Hard EM for Advertisements

Problem removed from assignment.

3.3 Complicated likelihood for leaky units on a tree

Problem formulation: Consider the following model. A binary tree T has random variables associated with its vertices. A vertex u has an observable variable X_u and a latent class variable Z_u. Each class $c \in [C]$ has a normal distribution $N\left(\mu_k, \sigma^2\right)$. If the three neighbors of u are v_1, v_2, and v_3, then

$$p\left(X_u \mid Z_u = c, Z_{v_1} = c_1, Z_{v_2} = c_2, Z_{v_2} = c_2\right) \sim N\left(X_u \mid (1-\alpha)\mu_c + \sum_{i \in [3]} \frac{1}{3}\alpha\mu_{c_i}, \sigma^2\right)$$

The class variables are iid, each follows the categorical distribution π. Provide a linear time algorithm that computes $P(X \mid T, M, \sigma, \alpha, \pi)$ when given a tree T (with vertices $V(T)$), observable variables for its vertices $X = \{X_v : v \in V(T)\}$, and parameters $M = \{\mu_c : c \in [C]\}, \sigma, \alpha$.

We are interested in finding the likelihood of our observations X by marginalising the following

$$p(X) = \sum_Z p(X, Z) \tag{1}$$

where $\sum_Z$ denotes the sum over all latent variables. We will show how this problem can be continuously split up into smaller and smaller subproblems using the structure of the binary tree until the leaves are reached. This will result in a linear algorithm for computing the requested likelihood $P(X \mid T, M, \sigma, \alpha, \pi)$ which we will from now on denote as $p(X)$ for the sake of brevity.

The instructions were unclear regarding the special cases of the root and leaves, since we have two neighbours in the root case and one neighbour in the leaf case. I have made the following interpretation.

- Root case: $p(X_u \mid Z_u = c, Z_{v_1} = c_1, Z_{v_2} = c_2) \sim N\left(X_u \mid (1-\alpha)\mu_c + \sum_{i \in [2]} \frac{1}{2}\alpha\mu_{c_i}, \sigma^2\right)$
- Leaf case: $p(X_u \mid Z_u = c, Z_{v_1} = c_1) \sim N\left(X_u \mid (1-\alpha)\mu_c + \alpha\mu_{c_1}, \sigma^2\right)$

Starting at the root

We will start by showing how one can split the problem into two subproblems when starting at the root.

Let u denote the root and u_1, u_2 denote its children as in figure 1.

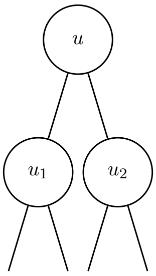

Figure 1: Binary tree starting at root r, branches below u_1, u_2 denotes subtrees

We begin by using that the root u is independent of its children given Z_u, Z_{u_1}, Z_{u_2} which leads to the following factorisation.

$$\begin{aligned}
p(X) = \sum_Z p(X, Z) &= \sum_Z p(X_u, X_{u_1}, X_{u_2}, X_{u_1\downarrow}, X_{u_2\downarrow}, Z) \\
&= \sum_Z p(X_u, X_{u_1}, X_{u_2}, X_{u_1\downarrow}, X_{u_2\downarrow}, Z_{u_1\downarrow}, Z_{u_2\downarrow}|Z_u, Z_{u_1}, Z_{u_2})p(Z_u, Z_{u_1}, Z_{u_2}) \\
&= \sum_Z p(X_u|Z_{u_1}, Z_{u_2}, Z_u)p(Z_{u_1}, Z_{u_2}, Z_u)p(X_{u_1}, X_{u_1\downarrow}, Z_{u_1\downarrow}|Z_{u_1}, Z_u)p(X_{u_2}, X_{u_2\downarrow}, Z_{u_2\downarrow}|Z_{u_2}, Z_u)
\end{aligned} \tag{2}$$

We can now let the sums over $Z_{u_1\downarrow}$ and $Z_{u_2\downarrow}$ move in which yields that

$$\begin{aligned}
p(X) = \sum_{Z_u, Z_{u_1}, Z_{u_2}} \Bigg[& p(X_u|Z_{u_1}, Z_{u_2}, Z_u)p(Z_{u_1}, Z_{u_2}, Z_u) \\
& \Big(\sum_{Z_{u_1\downarrow}} p(X_{u_1}, X_{u_1\downarrow}, Z_{u_1\downarrow}|Z_{u_1}, Z_u) \Big) \Big(\sum_{Z_{u_2\downarrow}} p(X_{u_2}, X_{u_2\downarrow}, Z_{u_2\downarrow}|Z_{u_2}, Z_u) \Big) \Bigg]
\end{aligned} \tag{3}$$

Using that the latent variables Z are independent given π and substituting for the available densities yields

$$p(X) = \sum_{Z_u, Z_{u_1}, Z_{u_2}} \Bigg[\mathcal{N}\Big(X_u|(1-\alpha)\mu_{Z_u} + \frac{\alpha}{2}(\mu_{Z_{u_1}} + \mu_{Z_{u_2}}), \sigma^2 \Big) \pi(Z_u)\pi(Z_{u_1})\pi(Z_{u_2}) \tag{4}$$

$$\underbrace{\Big(\sum_{Z_{u_1\downarrow}} p(X_{u_1}, X_{u_1\downarrow}, Z_{u_1\downarrow}|Z_{u_1}, Z_u) \Big)}_{p_{Z_{u_1\downarrow}}} \underbrace{\Big(\sum_{Z_{u_2\downarrow}} p(X_{u_2}, X_{u_2\downarrow}, Z_{u_2\downarrow}|Z_{u_2}, Z_u) \Big)}_{p_{Z_{u_2\downarrow}}} \Bigg] \tag{5}$$

Where $p_{Z_{u_1\downarrow}}$ and $p_{Z_{u_2\downarrow}}$ denotes two independent subproblems with respect to the sets of latent variables $Z_{u_1\downarrow}$ and $Z_{u_2\downarrow}$. Each can thus be solved separately whilst the other factor is stored an held constant. This is how the linearity is achieved.

Starting at a node within the tree

Now we will show how we can continue to divide each subproblem $p_{Z_{u_1\downarrow}}$ and $p_{Z_{u_2\downarrow}}$ defined above until the leaves are reached. We will now let u_1 and u_2 be the children of node u

and we will treat the problem shown in figure 2

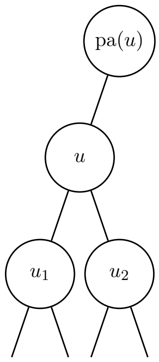

Figure 2: Binary tree starting at parent of u, branches below u_1, u_2 denotes subtrees

Let $p_{Z_{u\downarrow}} = \sum_{Z_{u\downarrow}} p(X_u, X_{u\downarrow}, Z_{u\downarrow} | Z_u, Z_{\text{pa}(u)})$, then

$$\begin{aligned}
p_{Z_{u\downarrow}} &= \sum_{Z_{u\downarrow}} p(X_u, X_{u_1}, X_{u_2}, X_{u_1\downarrow}, X_{u_2\downarrow}, Z_{u_1\downarrow}, Z_{u_2\downarrow} | Z_u, Z_{\text{pa}(u)}, Z_{u_1}, Z_{u_2}) p(Z_{u_1}) p(Z_{u_1\downarrow}) \\
&= \sum_{Z_{u\downarrow}} p(X_u | Z_u, Z_{\text{pa}(u)}, Z_{u_1}, Z_{u_2}) p(Z_{u_1}) p(Z_{u_2}) p(X_{u_1}, X_{u_1\downarrow}, Z_{u_1\downarrow} | Z_{u_1}, Z_u) p(X_{u_2}, X_{u_2\downarrow}, Z_{u_2\downarrow} | Z_{u_2}, Z_u) \\
&= \sum_{Z_{u\downarrow}} \Big[p(X_u | Z_u, Z_{\text{pa}(u)}, Z_{u_1}, Z_{u_2}) p(Z_{u_1}) p(Z_{u_2}) \\
&\qquad p(X_{u_1}, X_{u_1\downarrow}, Z_{u_1\downarrow} | Z_{u_1}, Z_u) p(X_{u_2}, X_{u_2\downarrow}, Z_{u_2\downarrow} | Z_{u_2}, Z_u) \Big] \\
&= \sum_{Z_{u_1}, Z_{u_2}} \Big[p(X_u | Z_u, Z_{\text{pa}(u)}, Z_{u_1}, Z_{u_2}) p(Z_{u_1}) p(Z_{u_2}) \\
&\qquad \Big(\sum_{Z_{u_1\downarrow}} p(X_{u_1}, X_{u_1\downarrow}, Z_{u_1\downarrow} | Z_{u_1}, Z_u) \Big) \Big(\sum_{Z_{u_2\downarrow}} p(X_{u_2}, X_{u_2\downarrow}, Z_{u_2\downarrow} | Z_{u_2}, Z_u) \Big) \Big] \\
&= \sum_{Z_{u_1}, Z_{u_2}} \Big[p(X_u | Z_u, Z_{\text{pa}(u)}, Z_{u_1}, Z_{u_2}) p(Z_{u_1}) p(Z_{u_2}) \Big(p_{Z_{u_1\downarrow}} \Big) \Big(p_{Z_{u_2\downarrow}} \Big) \Big] \\
&= \sum_{Z_{u_1}, Z_{u_2}} \Big[\mathcal{N}\Big(X_u | (1-\alpha)\mu_{Z_u} + \frac{\alpha}{3}(\mu_{Z_{u_1}} + \mu_{Z_{u_2}} + \mu_{Z_{\text{pa}(u)}}), \sigma^2 \Big) \cdot \\
&\qquad \cdot \pi(Z_{u_1}) \pi(Z_{u_2}) \Big(p_{Z_{u_1\downarrow}} \Big) \Big(p_{Z_{u_2\downarrow}} \Big) \Big]
\end{aligned}$$

On a more compact form we have thus shown that

$$p_{Z_{u\downarrow}} = \sum_{Z_{u\downarrow}} p(X_u, X_{u\downarrow}, Z_{u\downarrow} | Z_u, Z_{\text{pa}(u)}) \tag{6}$$

$$= \sum_{Z_{u_1}, Z_{u_2}} \left[\mathcal{N}\Big(X_u | (1-\alpha)\mu_{Z_u} + \frac{\alpha}{3}(\mu_{Z_{u_1}} + \mu_{Z_{u_2}} + \mu_{Z_{\text{pa}(u)}}), \sigma^2 \Big) \cdot \right. \tag{7}$$

$$\left. \cdot\, \pi(Z_{u_1}) \pi(Z_{u_2}) \Big(p_{Z_{u_1\downarrow}} \Big) \Big(p_{Z_{u_2\downarrow}} \Big) \right] \tag{8}$$

Which shows the recursion.

We have thus ended up with two new subproblems $p_{Z_{u_1\downarrow}}$ and $p_{Z_{u_2\downarrow}}$ from $p_{Z_{u\downarrow}}$ which can be divided into subproblems continuously until the leaves are reached.

It is important to note that $p_{Z_{u_1\downarrow}}$ is independent of Z_{u_2} and similarly $p_{Z_{u_2\downarrow}}$ is independent of Z_{u_1}. It is thus only necessary to compute $p_{Z_{u_1\downarrow}}$ when summing over Z_{u_1}. When summing over Z_{u_2} the value for $p_{Z_{u_1\downarrow}}$ can be computed once, stored and then be reused. Applying this for all subproblems results in a linear algorithm for computing $p(X)$.

Starting at a leaf

When equation (7) + (8) has been used recursively until the leaves are reached we need to show that the DP-algorithm can be started there. Let u_1 denote a *leaf node*, it thus has no children which implies that $Z_{u_1\downarrow} = \emptyset$. Using previous definition of $p_{Z_{u\downarrow}}$ yields

$$\begin{aligned} p_{Z_{u_1\downarrow}} &= \sum_{Z_{u_1\downarrow}} p(X_{u_1}, X_{u_1\downarrow}, Z_{u_1\downarrow} | Z_{u_1}, Z_{\text{pa}(u_1)}) \\ &= p(X_{u_1} | Z_{u_1}, Z_{\text{pa}(u_1)}) \\ &= \mathcal{N}\Big(X_{u_1} | (1-\alpha)\mu_{Z_{u_1}} + \alpha\mu_{Z_{\text{pa}(u_1)}}, \sigma^2 \Big) \end{aligned} \tag{9}$$

We have thus shown how the problem can be continuously separated into smaller and smaller independent subproblems that can be solved separately whilst holder the others constant and thus only need to be computed once. We have also shown how the DP-algorithm can be initialised when reaching the leaves of the tree.

3.4 Easier VI for Covid-19

Problem formulation: We have a workplace with K workers, $w_1, \cdots, w_K$, where we monitor Covid-19. Any day d each worker w_k is either non-infected, infected, or has antibodies, i.e., there is a latent variable Z_d^k with a value in $\{n, i, a\}$, with the obvious interpretation. A non-infected individual becomes with probability ι infected the day after the individual has had contact with an infected individual (and though only one such contact may occur with any single infected individual during a day, an uninfected may have contact with several infected during a day). An individual that becomes infected on day d is aware of the infection, and will on day $d+9$ get antibodies with probability α. Otherwise, the individual remains/returns to the non-infected state. An infected individual stays at home with probability σ and is otherwise present at the workplace. We have access to a contact graph G_d and an absence table A_d for each day $d \in [D]$, $A_d^k = 1$ if worker k is home on day d and otherwise 0. Consider $G = G_d$ as given so the joint is

$$p(A, Z, \Omega \mid G)$$

where $\Omega = (\iota, \alpha, \sigma)$. There are beta priors on Bernoulli parameters ι, α, and σ. No other reasons than Covid-19 makes any worker stay at home. On day one w_1 is infected and all other workers are non-infected. Let $Z^k = Z_1^k, ..Z_D^k$ and $Z = Z^1, \ldots Z^K$. Design a VI algorithm for approximating the posterior probability over Z and use the VI distribution

$$q(Z) = \prod_{d,k} q\left(Z_d^k\right)$$

During this problem I had a discussion with Ludvig Doeser regarding the problem formulation of the transition densities which were affected by how many infected workers one had met. This problem formulation was however simplified later on, although I after some time managed (I hope) to solve the harder case.

Working the joint distribution

We start off by simplifying the joint distribution keeping in mind that we are only interested in a variational approximation of the distribution for Z, yielding the complete likelihood.

$$p(A, Z, \Omega|G) = p(A, Z|\Omega, G)p(\Omega) \propto p(A, Z|\Omega, G)$$

The complete likelihood can then be expanded to a product of emission and transmission probabilities.

$$p(A, Z|\Omega, G) = \prod_{d,k} p(A_d^k|Z_d^k, \sigma)p(Z_{d+1}^k|Z_d, G_d, \iota, \alpha) \tag{10}$$

In order to enable us to keep track of when a worker should go from the state of infected to either the state of non-infected or the state of antibodies we expand the latent state to include the number of days a worker has been infected.

$$Z_d^k = \{s, \gamma\},\ s \in \{n, i, a\},\ \gamma \in \{0, 1, ..., 8\} \tag{11}$$

Note: in order to simplify calculations, γ will only be expressed on the conditioning side of the transition probability. We will in addition introduce a counter $n_d^k = g,\ g \in [K]$ that given Z_d^{-k} and G_d tells how many infected workers worker k has met during day d. This is useful when computing the transition probabilities.

The *emission probabilities* can then be expressed as

$$\begin{aligned} p(A_d^k | Z_d^k, \sigma) &= \prod_{s,l} \underbrace{p(A_d^k = l | Z_d^k = s, \sigma)}_{E_{sl}}{}^{I\{A_d^k = l, Z_d^k = s\}} \\ &= \prod_{s,l} E_{sl}^{I\{A_d^k = l, Z_d^k = s\}} \end{aligned} \tag{12}$$

We will later make use of the following **emission matrix** which describes the values for

$$E_{sl} = p(A_d^k = l | Z_d^k = s, \sigma)$$

	$l = 0$	$l = 1$
$s = n$	1	0
$s = i$	$1 - \sigma$	σ
$s = a$	1	0

Table 1: Emission probabilities for E_{sl} in white background

The *transition probabilities* can then be expressed as

$$\begin{aligned} p(Z_{d+1}^k | Z_d, G_d, \iota, \alpha) &= \prod_{s,t,g,\gamma} \underbrace{p(Z_{d+1}^k = t | Z_d^k = \{s, \gamma\}, n_d^k = g, \iota, \alpha)}_{T_{stg\gamma}}{}^{I\{Z_{d+1}^k = t, Z_d^k = \{s,\gamma\}, n_d^k = g\}} \\ &= \prod_{s,t,g,\gamma} T_{stg\gamma}^{I\{Z_{d+1}^k = t, Z_d^k = \{s,\gamma\}, n_d^k = g\}} \end{aligned} \tag{13}$$

Substituting the emission and transition probabilities into equation (10) yields

$$p(A, Z | \Omega, G) = \left(\prod_{d,k} \prod_{s,l} E_{sl}^{I\{A_d^k = l, Z_d^k = s\}} \right) \left(\prod_{d,k} \prod_{s,t,g,\gamma} T_{stg\gamma}^{I\{Z_{d+1}^k = t, Z_d^k = \{s,\gamma\}, n_d^k = g\}} \right) \tag{14}$$

We will later make use of the following **transition matrix** which describes the values for

$$T_{stg\gamma} = p(Z_{d+1}^k = t | Z_d^k = \{s, \gamma\}, n_d^k = g, \iota, \alpha)$$

			$t = n$	$t = i$	$t = a$
$s = a$	$\forall\gamma$	$\forall g$	0	0	1
$s = i$	$\gamma < 8$	$\forall g$	0	1	0
$s = i$	$\gamma = 8$	$\forall g$	$1 - \alpha$	0	α
$s = n$	$\forall\gamma$	$g = g$	$(1 - \iota)^g$	$1 - (1 - \iota)^g$	0

Table 2: Transition probabilities for $T_{stg\gamma}$ in white background

Update equations

Given the *variational distribution* for Z

$$q(Z) = \prod_{d,k} q(Z_d^k) \tag{15}$$

we need to compute

$$\begin{aligned}
\log q^*(Z_x^y) &\propto \underset{\{d,k\}\neq\{x,y\}}{\mathbb{E}} \Big[\log p(A, Z|\Omega, G) \Big] \\
&= \underset{\{d,k\}\neq\{x,y\}}{\mathbb{E}} \Big[\sum_{d,k} \sum_{s,l} I\{A_d^k = l, Z_d^k = s\} \log E_{sl} \Big] && (16) \\
&+ \underset{\{d,k\}\neq\{x,y\}}{\mathbb{E}} \Big[\sum_{d,k} \sum_{s,t,g,\gamma} I\{Z_{d+1}^k = t, Z_d^k = \{s,\gamma\}, n_d^k = g\} \log T_{stg\gamma} \Big] && (17)
\end{aligned}$$

We start by working with the emission term in equation (16). Keeping in mind that we are only interested in Z_x^y and that we are only taking the expectation with respect to all d, k except $\{d,k\} \neq \{x,y\}$. Z_x^y can thus be seen as a constant with respect to the expectation and can thus be moved out of it. This yields that

$$\underset{\{d,k\}\neq\{x,y\}}{\mathbb{E}} \Big[\sum_{d,k} \sum_{s,l} I\{A_d^k = l, Z_d^k = s\} \log E_{sl} \Big] \propto \sum_{s,l} \log E_{sl} I\{A_x^y = l, Z_x^y = s\} \tag{18}$$

Using the same mindset we can simplify the transmission term in equation (17) as follows

$$\begin{aligned}
&\underset{\{d,k\}\neq\{x,y\}}{\mathbb{E}} \Big[\sum_{d,k} \sum_{s,t,g,\gamma} I\{Z_{d+1}^k = t, Z_d^k = \{s,\gamma\}, n_d^k = g\} \log T_{stg\gamma} \Big] \\
&\propto \sum_{s,t,g,\gamma} \log T_{stg\gamma} \underset{\{d,k\}\neq\{x,y\}}{\mathbb{E}} \Big[I\{Z_x^y = t\} I\{Z_{x-1}^y = \{s,\gamma\}\} I\{n_{x-1}^y = g\} + \\
&\qquad + I\{Z_{x+1}^y = t\} I\{Z_x^y = \{s,\gamma\}\} I\{n_x^y = g\} \Big] \\
&= \sum_{s,t,g,\gamma} \log T_{stg\gamma} \Big(I\{Z_x^y = t\} P(Z_{x-1}^y = \{s,\gamma\}) P(n_{x-1}^y = g) + \\
&\qquad + I\{Z_x^y = \{s,\gamma\}\} P(Z_{x+1}^y = t) P(n_x^y = g) \Big) && (19)
\end{aligned}$$

Now, in equation (19) $P(Z_{x+1}^y)$ and $P(Z_{x-1}^y)$ can be computed using the values $q(Z_{x+1}^y)$ and $q(Z_{x-1}^y)$ for the previous VI iteration. However we need to compute the probability $P(n_x^y = g)$ which corresponds to the probability of worker y having met g infected individuals during day x, given the contact graph G_x.

$P(n_x^y = g)$ can be computed by using a function such as the "nchoosek" function in Matlab. Letting v be the column in the contact graph G_x representing the workers that worker y has met during day x, $C = \text{nchoosek}(v, g)$ returns a matrix C containing all possible combinations of the elements of vector v taken g at a time. Each row in C then represents a possible permutation of g infected workers, where c_{ij} represents the index of a worker. The probability is then given by

$$P(n_x^y = g) = \sum_i \prod_j^g q(Z_x^{c_{ij}}) \tag{20}$$

Substituting (18) and (19) into (16) and (17) respectively yields

$$\begin{aligned} \log q^*(Z_x^y) \propto \sum_{s,l} \log E_{sl} I\{A_x^y = l, Z_x^y = s\} + \\ + \sum_{s,t,g,\gamma} \log T_{stg\gamma} \Big(I\{Z_x^y = t\} P(Z_{x-1}^y = \{s,\gamma\}) P(n_{x-1}^y = g) + \\ + I\{Z_x^y = \{s,\gamma\}\} P(Z_{x+1}^y = t) P(n_x^y = g) \Big) \end{aligned} \tag{21}$$

Estimating parameters

Now we have everything we need to start the algorithm except any deterministic values for the parameters in Ω. These have to be estimated in order for us to have any values for $\log(E_{sl})$ and $\log(T_{stg\gamma})$. Given that we have beta priors on all Bernoulli parameters ι, α, σ we can get the values for these by taking the expectation of $\log(E_{sl})$ and $\log(T_{stg\gamma})$ with respect to the beta distribution. These have closed form solutions and can be computed using wolframalpha for example. For example if $\sigma \sim \text{Beta}(a, b)$ then

$$\mathbb{E}[\log(\sigma)] = \psi(a) - \psi(a + b)$$

where ψ is the digamma distribution. The remaining probabilities in the emission and transmission matrices can be computed in the same analog manner.

The algorithm

The algorithm can now be run in the following manner using the following relationship

$$\log q^{i+1}(Z_x^y) \propto \sum_{s,l} \log E_{sl} I\{A_x^y = l, Z_x^y = s\} + \tag{22}$$

$$\begin{aligned} + \sum_{s,t,g,\gamma} \log T_{stg\gamma} \Big(I\{Z_x^y = t\} q^i(Z_{x-1}^y = \{s,\gamma\}) P(n_{x-1}^y = g) + \\ + I\{Z_x^y = \{s,\gamma\}\} q^i(Z_{x+1}^y = t) P(n_x^y = g) \Big) \end{aligned} \tag{23}$$

Algorithm:

1. $i = 0$
2. Initiate $q^i(Z_x^y)$ $\forall x, y$ as a proper density

3. Run until convergence

    - For all x, y, g compute $P(n_x^y = g)$ with probabilities $q^i(Z)$ using equation (20)
    - For all x, y compute $\log q^{i+1}(Z_x^y)$ using equation (21), expectation values for $\log(E_{sl})$ and $\log(T_{stg\gamma})$ computed in the previous section and probabilities $P(n_x^y = g)$ computed on the previous line
    - For all x, y set $q^{i+1}(Z_x^y) = \frac{\exp\{\log q^{i+1}(Z_x^y)\}}{\sum_{x,y} \exp \log q^{i+1}(Z_x^y)}$
    - Set $i = i + 1$

Hard VI for Covid-19

Problem not attempted

3.6 Spectral Graph Analysis

Problem formulation: In this problem, you should solve each of the following three subproblems.

- Let $G = (V, E)$ be an undirected d-regular graph, let A be the adjacency matrix of G, and let $L = I - \frac{1}{d}A$ be the normalized Laplacian of G. Prove that for any vector $\mathbf{x} \in \mathbb{R}^{|V|}$ it is

$$\mathbf{x}^T L \mathbf{x} = \frac{1}{d} \sum_{(u,v)\in E} (x_u - x_v)^2 \tag{24}$$

- Show that the normalised Laplacian is a positive semidefinite matrix.

- Assume that we find a non-trivial vector $\mathbf{x}_*$ that minimises the expression $\mathbf{x}^T L \mathbf{x}$. First explain what non-trivial means. Second explain how $\mathbf{x}_*$ can be used as an embedding of the vertices of the graph into the real line. Use Equation (24) to justify the claim that $\mathbf{x}_*$ provides a meaningful embedding.

We begin by stating some useful properties that will be used throughout the problem. Given that $G = (V, E)$ is an undirected d-regular graph and that A is an adjacency matrix it follows that

- A is symmetric
- $(A)_{uv} = a_{uv} = \begin{cases} 1, & (u, v) \in E \\ 0, & \text{otherwise} \end{cases}$
- Each row/column of A sums up to d, I.E. $d = \sum_i a_{ij} = \sum_j a_{ij}$
- The main diagonal of A is filled with zeros

First subproblem

We can now begin the proof of the first subproblem.

$$x^T Lx = x^T(I - \frac{A}{d})x = \frac{1}{d}x^T(dI - A)x = \frac{1}{d}(\sum_i dx_i^2 - \sum_{i,j} x_i a_{ij} x_j)$$

Can now substitute for $d = \sum_j a_{ij}$ which yields that

$$\begin{aligned} x^T Lx &= \frac{1}{d}(\sum_{i,j} a_{ij}x_i^2 - \sum_{i,j} x_i a_{ij} x_j) \\ &= \frac{1}{2d}(\sum_{i,j} a_{ij}x_i^2 + \sum_{i,j} a_{ij}x_j^2 - 2\sum_{i,j} x_i a_{ij} x_j) \\ &= \frac{1}{2d}\sum_{i,j} a_{ij}(x_i - x_j)^2 \end{aligned}$$

Using that A is symmetric, I.E. that $a_{ij} = a_{ji}$ and that $a_{ii} = 0$, $\forall i$ we get that

$$\begin{aligned} x^T Lx &= \frac{1}{2d}\sum_{i,j} a_{ij}(x_i - x_j)^2 \\ &= \frac{2}{2d}\sum_{i>j} a_{ij}(x_i - x_j)^2 && (25) \\ &= \frac{1}{d}\sum_{(i,j)\in E} (x_i - x_j)^2 && (26) \end{aligned}$$

where we between Equation (25) and Equation (26) used that $a_{uv} = \begin{cases} 1, & (u,v) \in E \\ 0, & \text{otherwise} \end{cases}$. Which was to proven.

Second subproblem

We can now use the result of the first subproblem to show the second subproblem where we want to show that L is a positive semi-definite matrix. I.E. that

$$x^T Lx \geq 0 \; \forall x \in \mathbb{R}^{|V|} \tag{27}$$

In the first subproblem we showed that

$$x^T Lx = \frac{1}{d}\sum_{(i,j)\in E} a_{ij}(x_i - x_j)^2 \tag{28}$$

where d is a positive integer and $x \in \mathbb{R}^{|V|}$. It is thus sufficient to show that equation (28) is non-negative. Using that $f(t) = t^2$ is a non-negative function for all $t \in \mathbb{R}$. $x^T Lx$ is thus a sum of non-negative values multiplied with a positive value $\frac{1}{d}$ which gives that

$$x^T Lx = \frac{1}{d}\sum_{(i,j)\in E} a_{ij}(x_i - x_j)^2 \geq 0 \tag{29}$$

Which in turn proves that L is positive semi-definite.

Third subproblem

In this problem a *trivial* vector would be a constant vector I.E. that all elements in the vector are equal. This is since a constant vector will always minimise equation (24). Thus is, in this setting, a *non-trivial* vector x_* a non-constant vector.

In order to answer the second part of this question we need to understand what a *meaningful embedding* corresponds to in this setting. One of the main uses of spectral graph analysis is to perform spectral clustering, where one aims to cluster points that are similar. A way of measuring similarity within a set of points is by the number of edges within that set, where more edges are better. A meaningful embedding would thus correspond to a way of clustering points such that the number of edges within the clusters are high.

We are given that x_* is a non-trivial vector that minimises equation (24), x_* is thus not a constant vector. So given that a non-constant x_* vector is the solution to

$$x_* = \underset{x}{\operatorname{argmin}} \frac{1}{d} \sum_{(u,v)\in E} (x_u - x_v)^2 \tag{30}$$

where the sum is taken over the vertices $(u, v) \in E$, I.E. the vertices that have an edge between them. Given that a cluster of points is a set of points that have a lot of edges within that set, equation (24) will be minimised if points in the same clusters have similar values for their corresponding element in x_*. Additionally, since we are seeking a non-trivial solution, points from different clusters will have different values for their corresponding element in x_*. Thus can one plot the elements of x_* on the real line in order to receive a meaningful embedding. This will result in clusters of points on the real line representing clusters of the real data.

Note that if several vectors $x_1, x_2, ..., x_m$ minimises (24) (has small corresponding eigenvalues), one can apply k-means to those vectors to get a more accurate representation of the actual clusters that exist in the data.