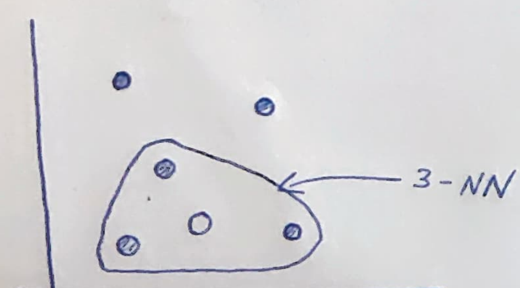
Q1) Graph learning with missing data

In real world scenario, missing node attributes is a common problem that we encounter while learning graph. From the problem statement, two things can be inferred,

1) For a given node, the data is partially missing & not entirely
2) Multiple attributes are missing from a single node.

This is a problem of multivariate missing value imputation. There are several techniques in Machine learning and Data mining literature for this problem. Here we will be making use of famous k-NN imputation to find missing attributes.

The key idea of k-NN is that we make use of similar nodes which are 'nearby' and impute the missing values. Here, Euclidean distance is calculated between nodes to find the nearest neighbour by ignoring the missing values.



There are some Graph Neural Network Architectures which discuss about missing value imputation along with graph learning. {"Wasserstein diffusion on graphs with missing attributes" by Chen, Zhixian and Ma, Tengf & Song, 2021}

But due to time limit, I have decided to go with k-NN imputation.

→ Now coming to graph learning, there are 3 main kind of graph learning

1) Linear combination model
2) Smooth signal model
3) Probabilistic Graphical model

Here, we will be learning graph under the assumption of smoothness. {Paper: "How to learn a graph from smooth Signals" Vassilis Kalofolias"}

→ A graph signal is smooth if the signal values associated with two end vertices of edge with large weights in the graph tend to be similar

→ Define, combinatorial graph laplacian $L = D - W$

$W \Rightarrow$ Weight matrix

$D \Rightarrow$ Diagonal matrix with each element equal to sum of weights connected to that node

$$D = \sum_{j} W_{ij}$$

→ For the smoothness, we ~~need~~ need to minimize,

$$\min \ \text{trace}(X^T L X)$$

$$\rightarrow tr(X^T L X) = tr(X^T D X) - tr(X^T W X)$$

$$= \sum_{i=1}^{m} \sum_{j=1}^{m} x_i^T W_{ij} x_i - \sum_{i=1}^{m} \sum_{j=1}^{m} x_i^T W_{ij} x_j$$

$$= \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} (x_i - x_j)^T W_{ij} (x_i - x_j)$$

$$= \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} W_{ij} \| x_i - x_j \|_2^2$$

$$= \frac{1}{2} \| W \circ Z \|_{1,1} = \frac{1}{2} tr(W, Z)$$

where $Z_{ij} = \| x_i - x_j \|_2^2$ ~~an~~

and $\circ \rightarrow$ Hadmard product

So, the objective func. is

$$\min_{W} \ \tfrac{1}{2} \sum_{j} W_{ij} \| x_i - x_j \|^2 + f(W)$$

Role of $f(W)$: $f(W)$ has 2 ~~£~~ roles to play

~~1) Using l_1 norm~~ .

1) To prevent W from going into trivial solution $W = 0$

2) To impose a prior belief in the graph signal

→ Using l_1 norm as $f(W)$ will be usefull in creating a sparse W, but it will translate into adding a constant to square distance in Z, so we make use of l_2 norm.

$$tr(X^T L X) + \eta \|W\|_{1,1} = \tfrac{1}{2} \| W \circ (2\eta + Z) \|_{1,1}$$

→ The final objective function :

$$\min_{W} \ \| W \circ Z \|_{1,1} - \alpha \, 1^T \log (W \cdot 1) + \beta \| W \|_F^2$$

where $1 = [1, 1 \ldots 1]^T$

$W 1 \rightarrow$ Diagonal elements of L

Taking the log prevents ~~from~~ the weights of a node going to zero, or in other words, it prevents formation of isolated nodes.

→ The addition of the Frobenius norm will penalize larger weights but do not penalize smaller ones.

→ ~~The well~~ α and β are the ~~hyp~~ parameters which can be tuned.

⇒ Furthermore, if we do a change of variable from $\tilde{W} = W/\gamma$, we can show that

$$F(z, \alpha, \beta) = \gamma F(z, \alpha/\gamma, \beta\gamma)$$

$$= \alpha F(z, 1, \alpha\beta)$$

→

This implies, if we want to ~~learn~~ obtain a W with fixed scale, then we can solve by tuning only one parameter β (Take $\alpha = 1$)

→ For a the optimization point of view, we use vector form representation of W

$$\Rightarrow \min_{w \in W} f_1(w) + f_2(Kw) + f_3(w)$$

where ~~$W1 \geq Kw$ to impose a fixed scale on W~~

$f_1(w) = 1$

where, $f_1(w) = \mathbb{1}\{w \geq 0\} + 2\, w^T z$

$$\mathbb{1}\{w \geq 0\} = \begin{cases} 0 & \text{if } w \geq 0 \\ \infty & \text{otherwise} \end{cases}$$

$$f_2(Kw) = f_2(d) = -\alpha\, 1^T \log(d)$$

where $Kw = W1$ to enforce a fixed scale on W

$$f_3(w) = \beta \|w\|^2$$