ST5225 Assignment

AY23/24 Semester 1

Instructions:

- The submission dateline is 22 Oct at 2359pm. If you are unable to make the dateline please let me know early and give me a valid reason.
- Upload your solutions to Assignment → Assignment Submission in Canvas. Upload a pdf or word file solution set with a txt file containing the codes that you used. I will be reading your pdf file and checking your txt file only if I have doubts on the codes that you use, so do make sure that the full solutions are in your pdf file. If unsure you can always reproduce your codes in both the pdf and txt file. Under Canvas you can submit multiple files within the same submission. Solutions with equation displays can be handwritten however make sure that I am able to read your handwriting.
- Make sure that you have contributed substantially to the solutions that you have submitted. Penalties will be imposed if the solutions of two students are unreasonably similar.
- This assignment is worth 30 marks. There are 5 questions with a total of 17 parts. Each part is worth 2 marks. You get full credit if you answer at least 15 parts correctly.
- All graphs in the assignment are undirected.

1. (a) Can you express the graph in Figure 1 as a bipartite graph? That is divide the nodes into 2 groups such that all edges are between the two groups.

   (b) Compute the modularity of your bipartite graph.

   (c) Is it true that all trees can be expressed as a bipartite graph? Is the expression unique? That is can there be two distinct ways of dividing the nodes into 2 groups? Please elaborate.

2. Let A be an $n \times n$ symmetric matrix. Let $\lambda_1, \ldots, \lambda_n$ be its eigenvalues and let $\mathbf{x}_i$ be the eigenvector associated with λ_i.

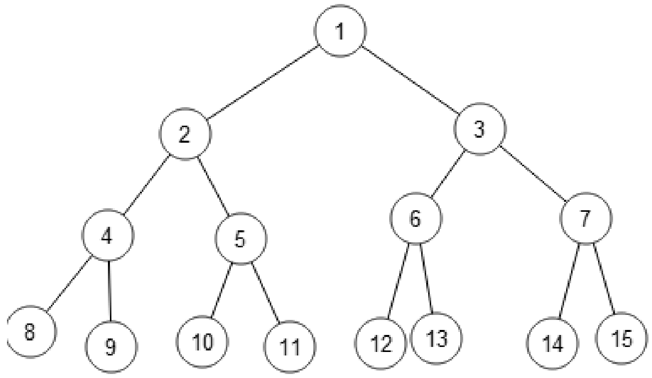

Figure 1:

(a) Let k be a non-negative integer. Show that the eigenvalues of A^k are $\lambda_1^k, \ldots, \lambda_n^k$ and that the eigvenvector associated with λ_i^k is $\mathbf{x}_i$.

(b) The Taylor expansion of the exponential function is

$$e^a = 1 + a + \tfrac{a^2}{2!} + \tfrac{a^3}{3!} + \cdots$$

for any real number a.

Similarly given a matrix A we define the matrix exponential

$$\exp(A) = I + A + \tfrac{A^2}{2!} + \tfrac{A^3}{3!} + \cdots .$$

What are the eigenvalues and eigenvectors of $\exp(A)$? Hint: This is simpler than it looks. The answer is similar to that of Question 2(a).

(c) A graph researcher would like to define a new measure of node centrality using the matrix exponential. Can you help him? It is similar to Katz centrality. Provide some explanation of what your node centrality is measuring.

3. The matrix exponential can be used in link prediction. Consider a graph generated using the codes below. Let A be the adjacency matrix and remove 48 of the edges randomly to form B. We want to predict the missing edges using B.

```
> #model for generating the edges of A
> set.seed(5225)
> y=rnorm(100)
> alist=matrix(nrow=1,ncol=2,data=0)
>
> for(i in 1:99) for(j in (i+1):100){
+ p=exp(-6*abs(y[i]-y[j]))
+ success=(runif(1)<p)
```

```
+ if(success==1) alist=rbind(alist,c(i,j))
+ }
>
> alist=alist[-1,]
> ell=nrow(alist)
> ell

[1] 483

>
> #creating the matrix A
> A=matrix(nrow=100,ncol=100,data=0)
> for(k in 1:ell) A[alist[k,1],alist[k,2]]=1
> A=A+t(A)
>
> #selecting a 90% subset of the edges of A
> bsample=sample(c(1:ell),48,replace=FALSE)
> blist=alist[-bsample,]
>
> #creating the matrix of B
> B=matrix(nrow=100,ncol=100,data=0)
> blength=nrow(blist)
> for(k in 1:blength) B[blist[k,1],blist[k,2]]=1
> B=B+t(B)
>
> #using B^2 for link prediction
> C=B%*%B
> pred=C*(1-B)*(1-diag(100))
> sort(-pred)[1:96]

 [1] -8 -8 -8 -8 -7 -7 -7 -7 -7 -7 -7 -7 -7 -7 -7 -7 -7 -7 -7 -7 -7 -7 -7 -7 -7 -7 -7 -
[35] -6 -6 -6 -6 -6 -6 -6 -6 -6 -6 -6 -6 -6 -6 -6 -6 -6 -6 -6 -6 -6 -6 -6 -6 -6 -6 -6 -6 -
[69] -6 -6 -6 -6 -6 -6 -6 -6 -6 -6 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -5 -

>
> #predicting 39 of the missing edges to be those with the largest entries
> #of B^2 in which b_ij=0 and i not equal to j.
>
> sum((pred>5)*A)/2

[1] 8

>
> #we only got 8 out of the 39 predictions correct.
> #but keep in mind that there are 4515 non-edge node pairs in B and
> #only 48 of them are edges in A
```

(a) Apply the matrix exponential $\exp(\alpha A)$ instead of A^2 for link prediction. Do note that you need to download the package `expm` to do this. Do not type `exp(A)` using the R software directly as you will simply get `exp(a[i,j])` for the entry in the ith row and jth column. Consider two values of α and decide which α is the better one.

(b) Explain how the method above can be made more complete by using cross-validation. That is using cross-validation to choose the best value of α.

(c) Repeat the exercise in Question 3(a) but now using preferential attachment. That is we score a node pair (u, v) by $k_u k_v$, where k_u is the degree of node u in the graph with missing edges.

4. In the codes below we constructed a graph using a block model of three blocks with 30 nodes in each block. In addition we have also constructed a network object `A2.network` containing the given graph and node attributes `"xval"` with values 1–3 depicting which group the nodes belong to.

```
> set.seed(5225)
> A2=matrix(nrow=90,ncol=90,data=0)
> x2=c(rep(1,times=30),rep(2,times=30),rep(3,times=30))
> for(i in 1:89) for(j in (i+1):90){
+ if(x2[i]==x2[j]) A2[i,j]=(runif(1)<0.5)+0
+ if(x2[i]!=x2[j]) A2[i,j]=(runif(1)<0.1)+0
+ }
>
> A2=A2+t(A2)
> library(ergm)
> A2.network=as.network(A2,directed=FALSE)
> A2.network%v%"xval"=x2
> A2.network

 Network attributes:
  vertices = 90
  directed = FALSE
  hyper = FALSE
  loops = FALSE
  multiple = FALSE
  bipartite = FALSE
  total edges= 918
    missing edges= 0
    non-missing edges= 918

 Vertex attribute names:
    vertex.names xval
```

```
No edge attributes
```

(a) Compute the transitivity and density of the graph.

(b) Is the transitivity of this graph significant against a random graph with the same degree sequence? Generate 999 random graphs with the same degree sequence and find a one-sided Monte Carlo p-value.

(c) Assume that the node attributes x_i are known to the user. Apply an appropriate logistic regression model and make the necessary conclusions.

(d) Assume that the node attribute information is not available. Apply an ERGM with GWESP and GWD counts. Conclude which terms are significant in your model.

(e) Next apply ERGM but with the node information available. Are the GWESP and GWD counts significant?

(f) Provide some comments on what you observed in Questions 4(d) and (e).

5. This is a problem to understand whether the graph Laplacian is effective in generating lost node information. Consider the adjacency matrix A generated in Question 3.

   (a) Apply the graph Laplacian on A and generate the values of x_i minimizing the graph Laplacian. Note that there may be isolated nodes or nodes lying in small components. For those nodes set $x_i = 0$.

   (b) Compute the correlation between x_i and y_i. Is the correlation significantly different from zero? Apply the Fisher transformation of Tutorial 1 to compute the two-sided p-value.