Exam Solutions

Student Name: Aditya Sharma

Matched Answers

Q1. Short answer/objective question.

(a) Consider the following incidence matrix of a simple undirected graph. Convert this into an adjacency matrix representation.

Answer:

```
Given
100
111
010
001

Each column represents an edge
Edge 1 = B/w node 1 and 2
Edge 2 = B/w node 2 and 3
Edge 3 = B/w node 2 and 4

So edges are (1,2) (2,3) (2,4)
 Adjacency Matrix =>
0 1 0 0
1 0 1 1
0 1 0 0
0 1 0 0
```

(b) Which network model assumes that edges are formed between pairs of nodes with a uniform probability, independent of other edges?

Answer:

```
Ans B is correct (Erdős{Rényi (Random Network) Model)
because in this model, edges between each pair of nodes
are created independently with a fixed probability.
```

(c) In game theory, a situation where no player can improve their outcome by unilaterally changing their strategy, given the strategies of other players, is known as:

Answer:

```
Ans C is correct => Nash Equilibrium
In Nash Equilibrium, no player can benefit by changing
strategies unilaterally.
```

(d) The tendency for individuals in a social network to associate and bond with similar others is defined as:

Answer:

```
Ans B is correct => Assortative Mixing
This is the tendency to connect with similar others in network.
```

(e) Why might betweenness centrality be a more relevant measure than degree centrality for identifying critical nodes in a network transmitting information that must follow specific paths?

Answer:

```
Ans D is correct => Because it quantifies how often a node
lies on the shortest paths between other nodes.
```

Q2. A novel influenza strain...

Answer:

```
Our Aim is to Vaccinate 5% of the population to minimize spread.
Strategy we can follow:
Using Betweenness centrality and degree centrality together
- Degree Centrality: Nodes with many direct connections spread the disea
- Betweenness Centrality: Nodes that act as bridges between different gr

Steps:
(i) Compute degree centrality for all nodes and select nodes with highes
```

```
(ii) Compute Betweenness centrality and select the bridge node.
(iii) Rank nodes using a weighted combination of both scores.
(iv) Select top 5% of the nodes based on the ranking.
```

Q3. You are tasked with improving the 'suggested collaborators' feature...

Answer:

```
Suggested Collaborators: Using Link Prediction + node embedding

Approach:
(i) Graph Construction: Build a graph using co-authorship and citation r
(ii) Node Embedding:
    a) Train Node2Vec on the graph to learn vector representation.
    b) Capture both homophily (similar fields) and structural roles.
(iii) Link Prediction
    a) Use similarity (cosine/dot-product) between embeddings to predict
    b) Combine with traditional link prediction scores like Jaccard, Ada

Homophily: Researchers from similar field have similar neighbors and thu
This leads to more accurate collaboration prediction within fields.

Promote Cross-Disciplinary Collaboration
Diversify top suggestion by adding an "explore" component
-> Penalize overly similar field
-> Encourage connections with structurally similar but topically differe
-> Use clustering or topic models to find diverse candidate.
```

Q4. Describe the core idea behind the Girvan-Newman algorithm...

Answer:

```
Three core idea of Girvan-Newman Algorithm
(i) Repeatedly remove edges with highest edge betweenness to break the g
(ii) Betweenness identifies bridges between communities.
(iii) As bridges are removed, the graph decomposes into densely connecte

Uses of Edge Betweenness
(i) After each edge removal, recompute edge betweenness.
```

(ii) This iterative recomputation reflects the evolving structure of the

Major Computational Limitation
(i) High time complexity: Recomputing edges betweenness is O(nm) per ste
(ii) Not scalable to large networks.

Louvain Method
(i) Greedy modularity optimization
    Two phases:
    a) Assign nodes to communities to locally maximize the modularity.
    b) Collapse communities into super nodes and repeat.
(ii) Very fast, scalable and handles larger graph well.

Q5. Explain the intuition behind the PageRank algorithm...

Answer:

The PageRank algorithm originally developed by Google is used to measure

Imagine a random web surfer clicking on links indefinitely. The more oft

Let PR(i) be the PageRank of node i, and L(j) be the number of outbound
PR(i) = (PR(j) / L(j)) for j In(i)

Damping factor d represents the probability (usually d = 0.85) that the

Equation with Damping:
PR(i) = (1-d)/N + d (PR(j) / L(j)) for j In(i)

Where N: Total number of nodes
(1-d)/N: Represents random jumps.

Nodes with no outgoing links are Dangling nodes in a web context; dead e

If a node has no outbound link, it can "absorb" rank. It breaks the assu

Cause the PageRank vector to lose mass and not remain stochastic.

To handle this in PageRank:
(i) Pretend that dangling nodes link to all nodes (including themselves)
(ii) Distribute their rank uniformly across the network.

```
If node(j) is dangling:
PR(i) is added to all nodes / N

Which modifies PageRank Equation:
PR(i) = (1-d)/N + d  (PR(j) / L(j)) + d  (PR(j) / N) for j  D

Where D: set of Dangling nodes.
```

Q6. Two players on a network edge play a game...

Answer:

```
A pure strategy Nash Equilibrium is a strategy profile (choice for each
no player can increase their payoff by unilaterally changing their choi

Let's analyze all 4 outcomes:
(i) (U,A)
    where player 1's payoff: 3 (better than 2)
    player 2's payoff: 2 (better than 0)
    No one has incentive to deviate => Nash Equilibrium
(ii) (U,B)
    player 1 gets 0 => could switch to L => gets 2 would deviate
     Not a Nash Equilibrium
(iii) (L,A)
    Player 1 gets 2 switching to U gives 3 would deviate -
     Not a Nash Equilibrium
(iv) (L,B)
    Player 1 gets 2 => switch to U gives 0 {No deviate}
    Player 2 gets 3 => switch to A gives 0
     Nash Equilibrium

Two pure strategy Nash Equilibria
(i) (U,A)
(ii) (L,B)

Expected payoff for player 2
Let p be the probability that player 1 plays U and (1-p) is for they pla

Now if player 2 plays A
E2(A) = p x 2 (U,A) + (1-p) x 0 (L,A) = 2p
```

```
If player 2 plays B
E2(B) = p x 1 (U,B) + (1-p) x 3 (L,B)
      = p + 3(1-p) = p + 3 - 3p
      => 3 - 2p

Hence
E2(A) = 2p
E2(B) = 3 - 2p

Expected outcome when p = 0.7
For A: E2(A) = 2p => 2 x 0.7 => 1.4
For B: E2(B) = 3 - 2p = 3 - 2 x 0.7 = 3 - 1.4 = 1.6
Player 2 should choose strategy B as 1.6 > 1.4
```

Q7. Consider the following simple directed graph...

Answer:

```
Given: A directed graph with edges:
A -> B, C -> B, D -> B

So neighbors of B = {A, C, D}

-> Initial feature vectors.
h_A^(0) = [1 1], h_C^(0) = [0 3], h_D^(0) = [2 2]

-> Weight Matrix W:
W = [0.5 0]
    [0.1 0.2]

-> Activation function ReLU applied element-wise as ReLU(x) = max(0,x)

Step 1: Aggregate Neighbor Features.
We compute the average of the feature from B's neighbor

h_N^(0) = 1/3  h_u^(0) for u  N(B)

Here N(B) = {A, C, D}
h_A^(0) + h_C^(0) + h_D^(0) = [1 1] + [0 3] + [2 2] = [3 6]
```

```
=> h_N^(0) = 1/3 [3 6] = [1 2]

Step 2: Linear Transformation
Apply the weighted matrix W to the aggregated vector

Wh_N^(0) = W . [1 2]

= [0.5 0] . [1]
  [0.1 0.2] [2]

= [(0.5)(1) + (0)(2)]
  [(0.1)(1) + (0.2)(2)]

= [0.5]
  [0.1 + 0.4]

= [0.5]
  [0.5]

Step 3: Activation (ReLU)
ReLU([0.5]
     [0.5]) = [max(0,0.5)]
              [max(0,0.5)]

= [0.5]
  [0.5]

Final Ans => h_B^(1) = [0.5]
                       [0.5]

This is the updated feature vector for node B.
```