DS3030: Data Analytics Lab
Assignment 6

Date: Sep 08, 2025

Timing: 2:00 to 5:00 PM | Max marks: 19

Instructions

- Submit one .ipynb file containing all answers named as [**student name**]_**assignment[number].ipynb**
- Write the **Part number** in **separate text blocks** before the answers.
- Write **justifications/comments** as required.
- You can use **networkx** library for graph operations and visualization.

---

1 Part I - Actor Collaboration Network

Dataset: "actorfilms.csv"
Dataset Name: "Movie Actor and Film Data"
Dataset Details: Contains information collected up to 5/5/21 about the actors and the various films they have acted in. The key features (columns) are "Actor", "Film", "Year" (release year of the film), "Votes" (the number of user votes), and "Rating" (IMDB rating of the film). More details can be found here.

1. (3 points) **Task: Actor Collaboration Strengths (Node & Edge Weights)**

   Print the number of nodes and edges in a graph built as follows:

   - **Nodes** represent actors who have acted in films with an IMDB rating between 8.5 and 9.
   - **Edges** represent movie collaborations between two actors (added if they have a film in common).
   - **Edge weights** are increased by 1 for every movie collaboration, representing the strength of actor collaborations.

   Workflow :

   1. Filter rows with "Rating" between 8.5 and 9.
   2. Form all pairs of actors belonging to a particular film.

3. Add the pairs as edges to an initially empty graph (created using networkx).
4. Increase the 'weight' property of an edge by 1, every time the corresponding pair is encountered.

NOTE: Adding an edge to a graph implicitly adds the respective nodes as well.

2. (3 points) **Task: Collaboration Network (Graph) Visualization**

   Visualize the collaboration network with:

   - Only those edges and their respective nodes that have an **edge weight of 2 or above** (stronger collaborations only).
   - Nodes should be **labelled by the actor names**.

   Workflow :

   1. Form a new graph by selecting those edges with a weight greater than 1.
   2. Print the number of nodes and edges of the above subgraph
   3. Nodes, Edges, and node labels can be drawn using the library functions from networkx and matplotlib.

2 Part II - Social Trust Network Analysis

Dataset: The Epinions dataset represents a directed social trust network from the consumer review site Epinions.com.

- Nodes: Users on the platform
- Edges: Directed trust relationships between users (A $\rightarrow$ B means user A trusts user B).
- Note: Trust relationships are not necessarily mutual; A trusting B does not imply B trusts A.

1. **Q: Total Users and Trust Relationships**

   Task:

   1. Load the dataset into Python.
   2. Create a graph using the dataset.
   3. Print the total number of users (nodes) and trust relationships (edges) in the entire network.

(2)

2. **Q: Random 1000 Users from total number of users (nodes) – With and Without Isolated Users**

   We want to study a small random sample of 1000 users:

   2(a): Select 1000 random users and create a subgraph.

   - Print the number of users (nodes) and trust relationships (edges).
   - Plot the subgraph.

   2(b): Remove isolated users (users with no trust relationships).

   - Print the number of users (nodes) and trust relationships (edges) after removal.
   - Plot the subgraph again to compare.

(3)

3. **Q: Similarity Between Users (Jaccard Coefficient)** Given two users A and B in a social network:

   Let N(A) be the set of users that A trusts & N(B) be the set of users that B trusts.

   The Jaccard coefficient is calculated as::

$$J(A, B) = \frac{|N(A) \cap N(B)|}{|N(A) \cup N(B)|}$$

   It measures how similar two users are based on the overlap of their friends.

   Task:

   1. Using the random 1000-user subgraph (before isolated nodes are removed), calculate the Jaccard similarity for each pair of connected users.
   2. Print the top 5 pairs of users with the highest similarity scores.
   3. **(Optional)** Plot a histogram of all similarity scores to see the distribution.

(2)

3 Part III - Centrality Analysis

Dataset: Karate Club network (available in networkx).

1. **Q: Centrality Calculation Matrix**

   Calculate the following centrality measures for **ALL nodes**:

   1. **Degree Centrality**
   2. **Betweenness Centrality**
   3. **Closeness Centrality**

   Deliverables:

   - Create a pandas DataFrame with Node_ID and all three centrality measures.
   - All centrality values must be normalized to a 0-1 scale for comparison.
   - Identify the top 5 nodes for each centrality measure.

   (4)

2. **Q: Create a Centrality Correlation Scatter Plot**

   Visualize the relationship between different centrality measures to understand how they correlate in the Karate Club network. Create a scatter plot that shows the relationship between **Degree Centrality and Betweenness Centrality**.

   Requirements:

   - **X-axis:** Degree Centrality values
   - **Y-axis:** Betweenness Centrality values
   - **Professional formatting:** Title, axis labels, grid

   Note: For solving question 2 you should have completed Question 1 and have:

   - A pandas DataFrame (centrality_df) with all centrality measures
   - Top 5 nodes identified for each centrality measure

   (2)