Assignment 4

April 29, 2022

---

*You are currently looking at **version 1.2** of this notebook. To download notebooks and datafiles, as well as get help on Jupyter notebooks in the Coursera platform, visit the Jupyter Notebook FAQ course resource.*

---

1 Assignment 4

```
In [ ]: import networkx as nx
        import pandas as pd
        import numpy as np
        import pickle
```

1.1 Part 1 - Random Graph Identification

For the first part of this assignment you will analyze randomly generated graphs and determine which algorithm created them.

```
In [ ]: P1_Graphs = pickle.load(open('A4_graphs','rb'))
        P1_Graphs
```

`P1_Graphs` is a list containing 5 networkx graphs. Each of these graphs were generated by one of three possible algorithms: * Preferential Attachment (`'PA'`) * Small World with low probability of rewiring (`'SW_L'`) * Small World with high probability of rewiring (`'SW_H'`)

Anaylze each of the 5 graphs and determine which of the three algorithms generated the graph.

The `graph_identification` function should return a list of length 5 where each element in the list is either `'PA'`, `'SW_L'`, or `'SW_H'`.

```
In [ ]: def graph_identification():

            # Your Code Here

            return # Your Answer Here
```

1.2 Part 2 - Company Emails

For the second part of this assignment you will be workking with a company's email network where each node corresponds to a person at the company, and each edge indicates that at least one email has been sent between two people.

The network also contains the node attributes `Department` and `ManagementSalary`.

`Department` indicates the department in the company which the person belongs to, and `ManagementSalary` indicates whether that person is receiving a management position salary.

```
In [ ]: G = nx.read_gpickle('email_prediction.txt')

        print(nx.info(G))
```

1.2.1 Part 2A - Salary Prediction

Using network `G`, identify the people in the network with missing values for the node attribute `ManagementSalary` and predict whether or not these individuals are receiving a management position salary.

To accomplish this, you will need to create a matrix of node features using networkx, train a sklearn classifier on nodes that have `ManagementSalary` data, and predict a probability of the node receiving a management salary for nodes where `ManagementSalary` is missing.

Your predictions will need to be given as the probability that the corresponding employee is receiving a management position salary.

The evaluation metric for this assignment is the Area Under the ROC Curve (AUC).

Your grade will be based on the AUC score computed for your classifier. A model which with an AUC of 0.88 or higher will receive full points, and with an AUC of 0.82 or higher will pass (get 80% of the full points).

Using your trained classifier, return a series of length 252 with the data being the probability of receiving management salary, and the index being the node id.

```
Example:

    1       1.0
    2       0.0
    5       0.8
    8       1.0
        ...
    996     0.7
    1000    0.5
    1001    0.0
    Length: 252, dtype: float64
```

```
In [ ]: def salary_predictions():

            # Your Code Here

            return # Your Answer Here
```

1.2.2 Part 2B - New Connections Prediction

For the last part of this assignment, you will predict future connections between employees of the network. The future connections information has been loaded into the variable `future_connections`. The index is a tuple indicating a pair of nodes that currently do not have a connection, and the `Future Connection` column indicates if an edge between those two nodes will exist in the future, where a value of 1.0 indicates a future connection.

```
In [ ]: future_connections = pd.read_csv('Future_Connections.csv', index_col=0, converters={0: e
        future_connections.head(10)
```

Using network `G` and `future_connections`, identify the edges in `future_connections` with missing values and predict whether or not these edges will have a future connection.

To accomplish this, you will need to create a matrix of features for the edges found in `future_connections` using networkx, train a sklearn classifier on those edges in `future_connections` that have `Future Connection` data, and predict a probability of the edge being a future connection for those edges in `future_connections` where `Future Connection` is missing.

Your predictions will need to be given as the probability of the corresponding edge being a future connection.

The evaluation metric for this assignment is the Area Under the ROC Curve (AUC).

Your grade will be based on the AUC score computed for your classifier. A model which with an AUC of 0.88 or higher will receive full points, and with an AUC of 0.82 or higher will pass (get 80% of the full points).

Using your trained classifier, return a series of length 122112 with the data being the probability of the edge being a future connection, and the index being the edge as represented by a tuple of nodes.

Example:

```
(107, 348)    0.35
(542, 751)    0.40
(20, 426)     0.55
(50, 989)     0.35
              ...
(939, 940)    0.15
(555, 905)    0.35
(75, 101)     0.65
Length: 122112, dtype: float64
```

```
In [ ]: def new_connections_predictions():

            # Your Code Here

            return # Your Answer Here
```