BSc (Hons) in Information Technology
Year 4
Assignment –AI & ML

SE4010– Current Trends in Software Engineering

100 Marks (10% of the Total Grade)

IT20252090 – Athapaththu A.H.M.C.P.

Question 1

a)

I. Based on the given scenario, I would recommend using the Association Rule Mining (ARM) model.

II. The reasoning behind this recommendation is that the goal is to predict which items should be placed together in a supermarket. Association Rule Mining is a technique that is specifically designed for identifying the relationships between different items in a dataset, making it ideal for this scenario.

Association Rule Mining works by identifying patterns or rules that exist within a dataset. These rules are typically in the form of "if X, then Y," where X and Y are both items within the dataset. For example, the model may identify that customers who purchase bread are likely to also purchase milk. This information can be used to suggest that bread and milk should be placed together in the supermarket.

Additionally, the dataset provided contains details of each bill issued to customers over the last 3 years. This means that there is a large amount of transactional data available, making it an ideal dataset for Association Rule Mining.

III. Another approach that could be used to solve the given scenario is the Collaborative Filtering (CF) model.

Collaborative Filtering is a technique commonly used in recommender systems. It works by analyzing the historical data of customers and their purchasing behavior. Based on this data, the model identifies customers who have similar buying behavior and recommends items that these customers have previously purchased. This technique is used by many online retailers to recommend products to their customers.

In the context of this scenario, Collaborative Filtering could be used to identify which items are commonly purchased together by different customers. By analyzing the historical data of customer purchases, the model could identify customers who frequently purchase the same items and recommend that those items be placed together in the supermarket.

However, compared to ARM, CF requires a large dataset of customer behavior, which might be difficult to obtain. Furthermore, CF may not be as useful for identifying items that are not frequently purchased together, which could limit the recommendations for the supermarket layout.

b)

I. Based on the scenario, I would recommend using a Reinforcement Learning (RL) model for training the navigation system of the robot.

II. Reinforcement Learning is a type of machine learning that involves training an agent to interact with an environment and learn from the feedback it receives in the form of rewards. In the case of the robot navigation problem, the environment would be the surface of Mars, and the rewards would be based on the robot's ability to reach the assigned locations, avoid obstacles, and extract samples successfully.

The RL model would enable the robot to learn from its experiences and improve its navigation strategy over time. It would use techniques like Q-learning or policy gradient methods to optimize the robot's actions based on the rewards it receives. The RL model would also be able to adapt to different situations that might occur during navigation, such as unexpected obstacles or changes in terrain.

Moreover, the RL model would also be able to optimize the trade-off between exploration and exploitation of the environment, allowing the robot to discover new locations while still prioritizing the assigned ones.

III. Another approach that could be used to solve the given scenario is a Computer Vision-based model. This model would involve using cameras or other sensors on the robot to capture images of the surroundings and then using image processing and object detection techniques to identify the locations, obstacles, and samples.

The Computer Vision model would be able to identify the terrain and obstacles more accurately and, in more detail, than the RL model. However, it would require extensive preprocessing of the sensor data and would not be able to adapt to unexpected situations as effectively as the RL model.

Additionally, the Computer Vision model might struggle with identifying landmarks or features in the terrain if they are not easily distinguishable, while the RL model would be able to use a variety of feedback signals to navigate to the correct location.

Question 2

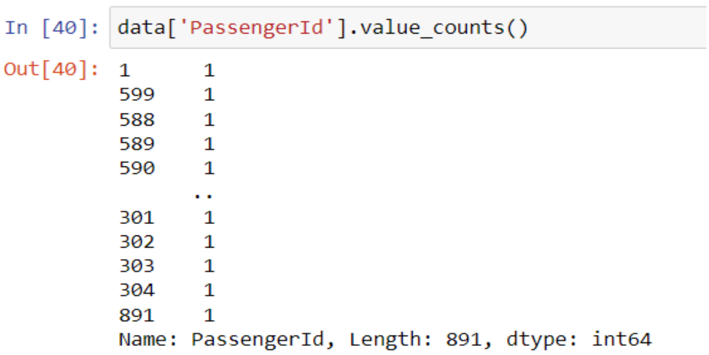
I.)

Column	Selected/Not Selected	Reasoning
PassengerId	Not Selected	PassengerId is just a unique identifier assigned to each passenger and does not provide any meaningful information for predicting the survival rate.
Survived	Selected	Survived is the target variable we are trying to predict and therefore must be selected.
Pclass	Selected	Pclass is likely to be a significant factor in determining a passenger's survival rate as the class of accommodation could be correlated with a passenger's location on the ship. Higher-class passengers might be more likely to survive than lower-class passengers.
Name	Not Selected	Although names don't directly impact the survival rate, they could be used to extract useful information, such as the passenger's title (e.g., Mr, Mrs, Miss, etc.), which could potentially be correlated with their survival rate.
Sex	Selected	The sex of a passenger is likely to be a significant factor in determining a passenger's survival rate, as women and children were given priority in the lifeboats.
Age	Selected	Age is likely to be a significant factor in determining a passenger's survival rate, as children were given priority in the lifeboats, and older passengers may have been less able to escape the sinking ship.
SibSp	Selected	The number of siblings or spouses that a passenger had on board could be a significant factor in their survival rate, as passengers traveling with family members might have had a higher chance of survival if they stuck together and helped each other.
Parch	Selected	The number of parents or children that a passenger had on board could be a significant factor in their survival rate, as passengers traveling with family members might have had a higher chance of survival if they stuck together and helped each other.
Ticket	Not Selected	The ticket number is unlikely to be a significant factor in determining a passenger's survival rate, as it is just a unique identifier assigned to each passenger's ticket.
Fare	Selected	The fare paid by a passenger could be correlated with their class of accommodation, which, as mentioned earlier, could be a significant factor in determining their survival rate.
Cabin	Not Selected	The cabin number is unlikely to be a significant factor in determining a passenger's survival rate, as passengers' cabins were distributed across the ship's different levels and areas.
Embarked	Selected	The port of embarkation could potentially be correlated with a passenger's survival rate, as passengers from different ports may have had different characteristics, such as social status or nationality, that could impact their survival rate.

II.)

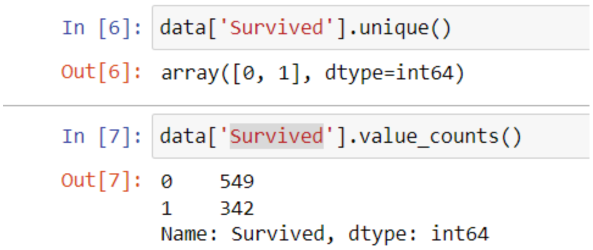
Column	Pre-Processing Technique	Reasoning
PassengerId	Not Applicable	This column contains unique identifiers for each passenger, and it does not provide any relevant information for our prediction task.
Survived	Not Applicable	This is the target variable we want to predict, so no pre-processing is needed.
Pclass	Not Applicable	This column already represents the class of the passenger's ticket, and it is categorical in nature. No further processing is required.
Name	Feature Engineering: Extract Title	The name column includes the passenger's title, which can provide additional information about their socio-economic status and could be used to group passengers into different categories. We can extract the title from the name and create a new feature for it.
Sex	Not Applicable	This is a categorical variable and does not require any pre-processing.
Age	Imputation, Binning	There are missing values in the Age column, which can be imputed with either the mean or median value. Additionally, it can be useful to create age groups by binning the data to make it easier to analyze and model.
SibSp	Not Applicable	This is a numerical variable and does not require any pre-processing.
Parch	Not Applicable	This is a numerical variable and does not require any pre-processing.
Ticket	Not Applicable	The Ticket column contains unique ticket numbers for each passenger, and it does not provide any relevant information for our prediction task.
Fare	Imputation, Binning	Like the Age column, there are missing values in the Fare column, which can be imputed with either the mean or median value. Additionally, it can be useful to create fare groups by binning the data to make it easier to analyze and model.
Cabin	Feature Engineering: Extract Cabin Letter	The Cabin column contains the cabin number for each passenger. While there are many missing values, we can extract the first letter of the cabin number to create a new feature, which could provide useful information about the passenger's location on the ship.
Embarked	Imputation	There are a few missing values in the Embarked column, which can be imputed with the mode value. This column is already categorical and does not require any further pre-processing.

➢ PassengerId



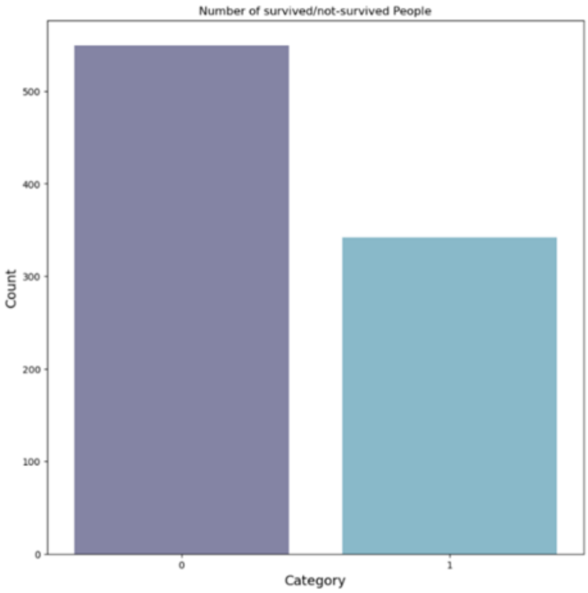
```
In [40]: data['PassengerId'].value_counts()

Out[40]: 1      1
         599    1
         588    1
         589    1
         590    1
               ..
         301    1
         302    1
         303    1
         304    1
         891    1
         Name: PassengerId, Length: 891, dtype: int64
```

➢ Survived

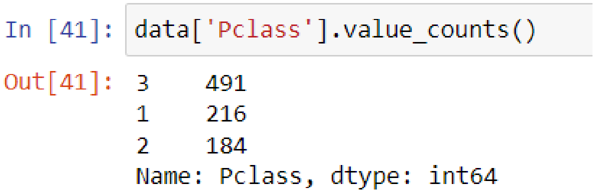


```
In [6]: data['Survived'].unique()

Out[6]: array([0, 1], dtype=int64)
```

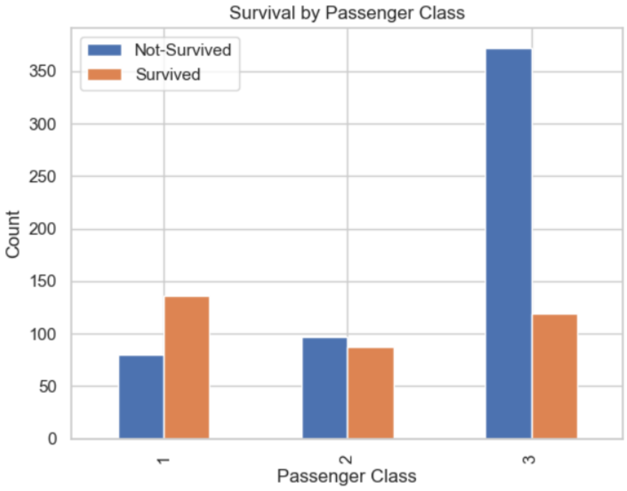
```
In [7]: data['Survived'].value_counts()

Out[7]: 0    549
        1    342
        Name: Survived, dtype: int64
```



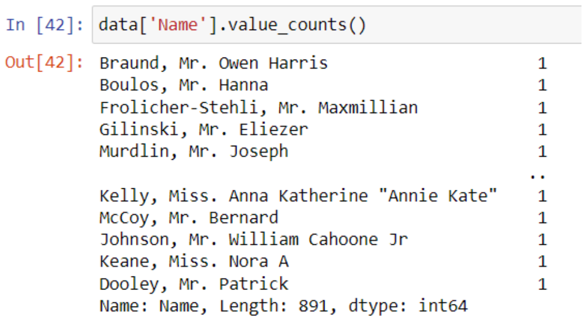
➢ Pclass



```
In [41]: data['Pclass'].value_counts()

Out[41]: 3    491
         1    216
         2    184
         Name: Pclass, dtype: int64
```

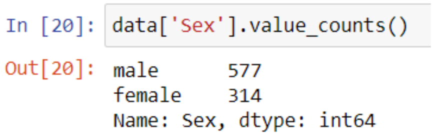


➢ Name

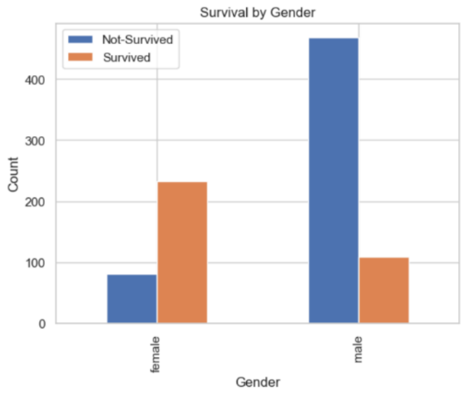


```
In [42]: data['Name'].value_counts()

Out[42]: Braund, Mr. Owen Harris                                1
         Boulos, Mr. Hanna                                      1
         Frolicher-Stehli, Mr. Maxmillian                       1
         Gilinski, Mr. Eliezer                                  1
         Murdlin, Mr. Joseph                                    1
                                                               ..
         Kelly, Miss. Anna Katherine "Annie Kate"               1
         McCoy, Mr. Bernard                                     1
         Johnson, Mr. William Cahoone Jr                        1
         Keane, Miss. Nora A                                    1
         Dooley, Mr. Patrick                                    1
         Name: Name, Length: 891, dtype: int64
```

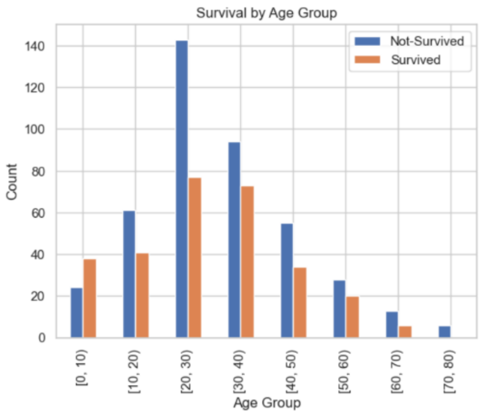
➢ Sex

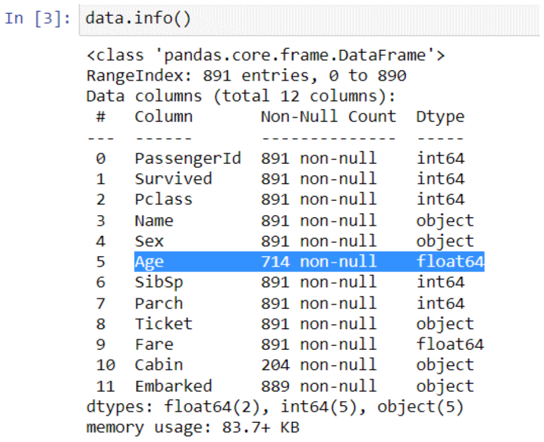


```
In [20]: data['Sex'].value_counts()

Out[20]: male      577
         female    314
         Name: Sex, dtype: int64
```



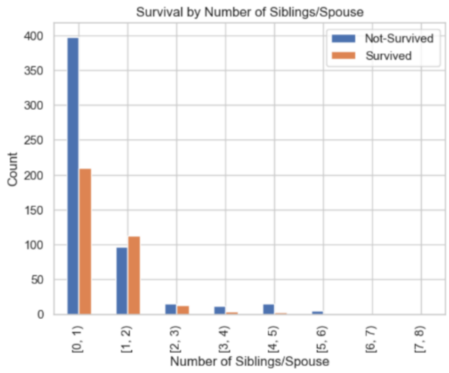
➢ Age



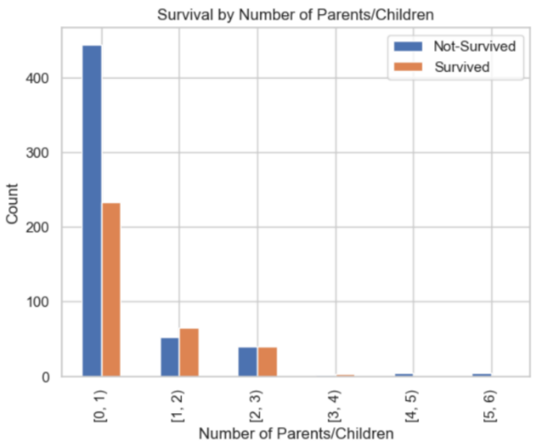


```
In [3]: data.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 891 entries, 0 to 890
Data columns (total 12 columns):
 #   Column       Non-Null Count  Dtype
---  ------       --------------  -----
 0   PassengerId  891 non-null    int64
 1   Survived     891 non-null    int64
 2   Pclass       891 non-null    int64
 3   Name         891 non-null    object
 4   Sex          891 non-null    object
 5   Age          714 non-null    float64
 6   SibSp        891 non-null    int64
 7   Parch        891 non-null    int64
 8   Ticket       891 non-null    object
 9   Fare         891 non-null    float64
 10  Cabin        204 non-null    object
 11  Embarked     889 non-null    object
dtypes: float64(2), int64(5), object(5)
memory usage: 83.7+ KB
```

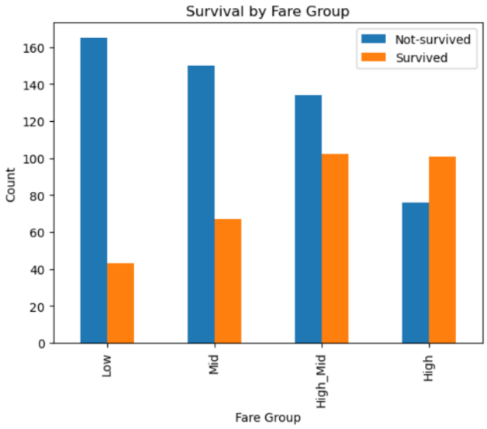
➢ SibSp



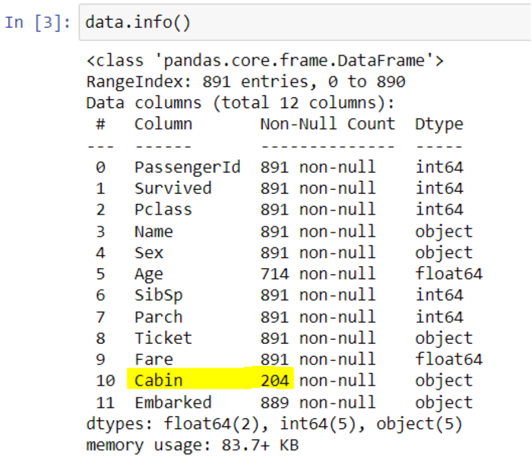
➢ Parach



➢ Fare



➢ Cabin



```
In [3]: data.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 891 entries, 0 to 890
Data columns (total 12 columns):
 #   Column       Non-Null Count  Dtype  
---  ------       --------------  -----  
 0   PassengerId  891 non-null    int64  
 1   Survived     891 non-null    int64  
 2   Pclass       891 non-null    int64  
 3   Name         891 non-null    object 
 4   Sex          891 non-null    object 
 5   Age          714 non-null    float64
 6   SibSp        891 non-null    int64  
 7   Parch        891 non-null    int64  
 8   Ticket       891 non-null    object 
 9   Fare         891 non-null    float64
 10  Cabin        204 non-null    object 
 11  Embarked     889 non-null    object 
dtypes: float64(2), int64(5), object(5)
memory usage: 83.7+ KB
```