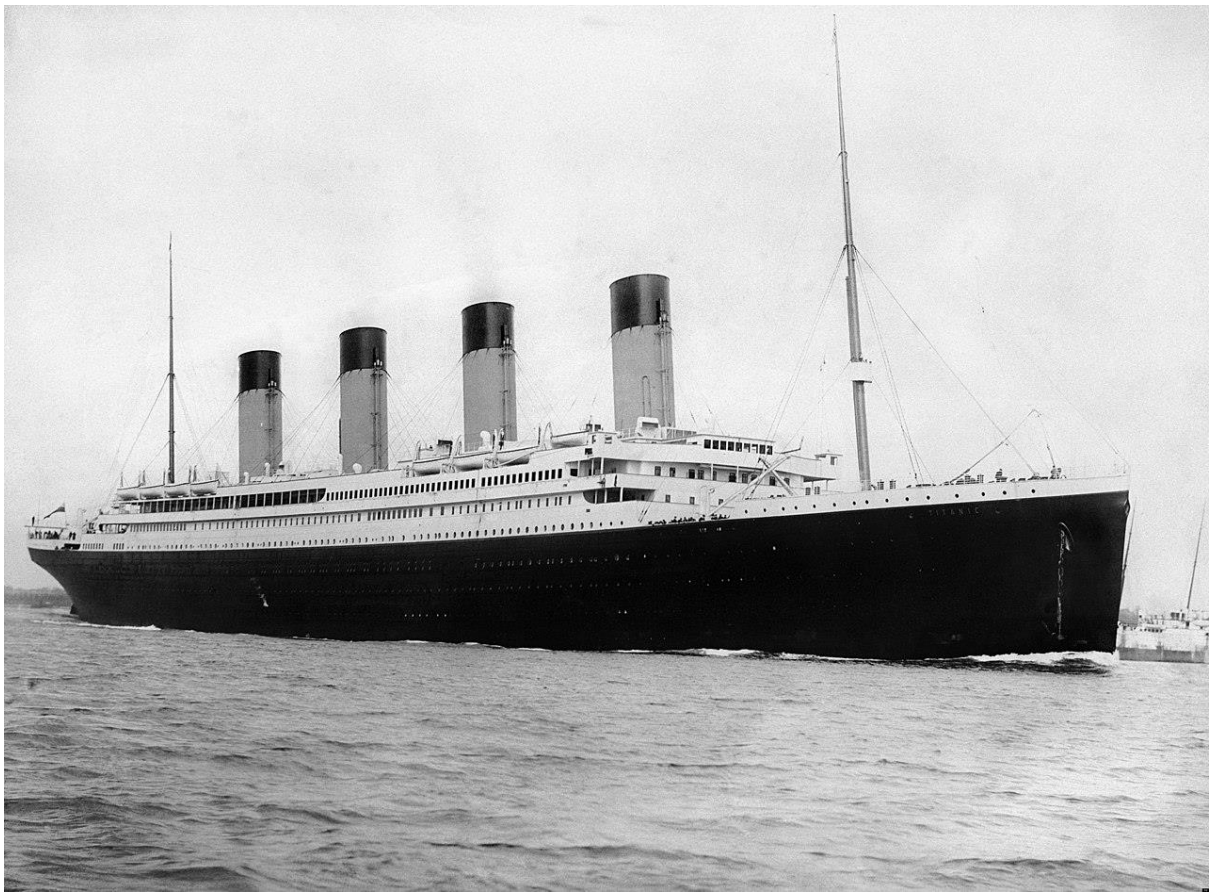

BUSA3020 Advanced Analytics Techniques

Assignment 2: Predictive Analytics

Dataset: Titanic Data

Chosen Software: Python vs Orange

Student Number: 45197083

Introduction (Data Cleansing and Feature Engineering)

The Titanic dataset (1309 Rows x 12 Columns) is an infamous classification problem which allows us to predict whether the passenger will survive in the Titanic disaster. This is an opportunity for us to explore different types of machine learning techniques and experiment it in various statistical programs.

Once we received the dataset, data cleansing is one of the most important steps as it allows us to deal with the missing data and values that may not useful. In the following table, it documented the data cleansing process.

Table 1: The steps of data cleansing. (**First** stage of data cleansing)

1. Remove the missing values within the '**Age'** column.
2. The column '**Life Boat**' is dropped as if the passenger got on a lifeboat, he/she would likely to survive.
3. There are too many missing values within the '**Cabin'** column therefore it is advised to drop the entire column.
4. Drop the missing values of '**Port of Embarkation'** and '**Passenger Fare'** column.

After completing the first stage of data cleansing, feature engineering is needed (such as changing the data type, changing the value of the data) before we conduct any analysis. The steps will be shown in the table below.

Table 2: Feature Engineering

1. Convert the target variable '**Survived'** into a dummy variable: Yes = 1, No = 0
2. Covert the variable '**Gender'** into a dummy variable: Male = 0, Female = 1 as it can be useful for our logistic regression analysis. This also changes the variable type from categorical to numerical.
3. Replace the values in the '**Passenger Class**' column where values such as 'First Class', 'Second Class' and 'Third Class' are transformed into numerical features: '1', '2', '3'.
4. Create a new column called '**Age Category'** and assign the passengers' age into 4 different groups. For instance: '**Toddler/Child**' for passengers who aged 0-2; '**Child'** for passengers who aged 3-17; '**Adult'** for passengers who aged 18-65 and '**Elderly'** for passengers who aged 66 or above.
5. Create a new column called '**Title'** which the titles of passengers into **6** major groups. The groups are: '**Mr'**, '**Miss'**, '**Mrs'**, '**Master**, '**Officer'**, and '**Aristocrat'**. *Note: These titles have replaced by numbers 1-6 for the ease of the data analysis.

By applying appropriate adjustments to the dataset, we can apply different machine learning algorithms and measure the prediction accuracy, nonetheless, a final examination of the data-frame is needed to avoid any duplications or data redundancy.

Table 3: The steps of data cleansing. (**Second** stage of data cleansing)

1. Remove the column '**Age'** as we already have a column called **'Age Group**' so we do not need two columns related to Age.
2. There are duplicate entries (i.e. a family bought the same ticket for all of their family members and therefore its ticket number is the same) within the '**Ticket Number**' therefore it is removed as it will affect the prediction accuracy of our model. *Note: '**Ticket Number**' is acting as an identifier, therefore, we cannot predict our model based on such feature.
3. There is a correlation between '**Ticket Number**' and '**Passenger Fare**' as a result, such a feature is also removed.
4. I have made an assumption which the port of embarkation will **NOT** affect the survival of the passengers. Therefore, the column **'Port of Embarkation'** has taken out.
5. Finally, the column '**Name'** is also taken out due to the fact it is a categorical variable and we already created the column '**Title'** and we can identify the socioeconomic status of the passenger.

	Survived	Passenger Class	Gender	Age Group	No of Siblings or Spouses on Board	No of Parents or Children on Board	Title
1	1	1	1	3	0	0	2
2	1	1	0	1	1	2	4
3	0	1	1	1	1	2	2
4	0	1	0	3	1	2	1
5	0	1	1	3	1	2	3

Diagram 1: The first 5 rows of the dataframe that we have created (after all the data cleansing and feature engineering).

Task One: Experiment with Alternative Algorithms & Programs

After experimenting with different analytics software, **Orange** and **Python** were chosen to carry out our analysis. They are chosen as we can compare the functions and capabilities between a GUI platform and a programming language.

The motives and the results of those chosen machine learning algorithms will be shown in the table below.

Accuracy is used to measure the overall result (both **'Survival'** and **'Deaths'**)

Table 4 The motives and results of chosen machine learning algorithms

Machine Learning Algorithm	**Motivation**	**Orange Result (Accuracy)**	**Python Result (Accuracy)**
Logistic Regression (LR)	Logistic regression is used because our target variable '**Survived'** is a binary variable and our features are numerical variables.	0.796	0.800
Decision Tree (DT)	A decision tree is a supervising learning algorithm which allows us to classify the target variable '**Survived'** based on the features available.	0.777	0.789
Random Forest (RF)	Random Forecast is known for providing high accuracy results – it is an extension of the decision tree algorithm. Therefore it will be interesting to compare the accuracy for these two algorithms	0.768	0.800
Naïve Bayes (NB)	Naïve Bayes is appropriate for this study as there are multiple attributes (such as age group, genders and others) that we can use to predict our target variable '**Survival'**. This problem is also called multi-class classification.	0.739	0.804

*Note: All the algorithms used are all examples of supervised learning – in which the algorithms are learning from the training set of **labelled** examples to conclude to the set of inputs.

Task Two: Advise a client on a preferred program

After carrying out our analysis in both **Orange** and **Python**, a recommendation needs to make to a client on which program to adopt for future predictive analytics tasks. The purpose of the evaluation was to access the key features of these software. These features include the ease of use, flexibility, cost, and others. A recommendation will be made based on a set of criteria listed in the table below.

Table 5 Evaluation Criteria for both Python and Orange

Criteria	**Orange**	**Python**
Ease of Use	- Orange is a using a drag and drop and widget-based hence users will find it easy to use and navigate. Score: 5/5	- Prior programming knowledge is a pre-requisite of using Python. - Although it is easy to learn, the learning curve of Python is fairly steep. Score: 3/5
Cost	- FREE – It is a free, open-sourced data visualisation and analysis tool that is available through both the Anaconda Navigator and its official website. Score: 5/5	- FREE – It's an open-sourced programming language that is available for everyone to use. Score: 5/5
Robustness (In regards to large datasets)	- Orange is not very robust if we are dealing with large datasets and it will eventually crash. Score: 1/5	- Python can deal with large datasets. Score: 5/5
Level of support.	- There is an insufficient amount of support provided by the developers. - Orange is less well known than Python, therefore, there are fewer online tutorials available for the users to follow along. Score: 2/5	- There is documentation for every single Python library (such as NumPy, Matplotlib, Seaborn, and others) therefore it can be concluded users will receive sufficient support while using such a program. - There are numerous amount of YouTube tutorials, study guides, books to help a beginner to get used to the programming language itself. Score: 5/5

Integration	- Orange cannot integrate with other third-party software or plugins. As a result, users will only able to use the functions within such software. Score: 1/5	- Python allows users to install different types of modules to suit their needs. For example, there are modules for GIS (Geographic Information System), Image Manipulation, databases and others. Score: 5/5
Total Score	**14/25**	**23/25**

Conclusion

Based on the selection criteria, I would recommend my client to use **Python** to carry out his/her analysis. Such a decision can be explained by the selection criteria and the scoring metric. Although **Python** has a steeper learning curve than **Orange,** the amount of support through online forums and videos will help the user to familiarise with the program itself.