Predicting the Survival of Titanic Passengers using Machine Learning and Graphical User Interface

Group Members: Zain Siddiqui, Krupal Patel, Haneef Pervez

This project consists of a clean and polished Graphical User Interface (GUI) that interacts with various Machine Learning models and data visualization tools through the use of different Python libraries. In class, we learned about Python being a great general purpose language, which allows for great versatility for developers of all specialties. Therefore, we decided to take advantage of Python's strong support for GUI development as well as its Data Science and Machine Learning capabilities. Using the complex RMS Titanic data set, which includes information about each passengers fate (survived/deceased) according to their economic status, fair, cabin, social class, relatives, gender, port of embarkment and age, we created 8 different Machine Learning models (Logistic Regression, Stochastic Gradient Descent, K Nearest Neighbor, Random Forest, Naive Bayes, Perceptron, Linear Support Vector Machine, Decision Tree) that learn from the data set and then perform accurate predictions of survival on the testing data provided by the user. In addition, we created an extensive GUI that allows the user to learn and interact with the training and testing data by displaying many different data plots and graphs as well as descriptions about the specifics (advantages and disadvantages) of each Machine Learning model. The user can interact with the GUI through selecting which model to run the testing data on, which then takes them to a screen displaying the prediction results of the testing data as well as the general model accuracy. The screen also includes various buttons that, when selected, display complex and attractive data visualizations on the testing data. The goal for this project was to get a good understanding of Python's Data Science and Machine Learning support and to learn about GUI development and integration in Python. Upon completing the project, we had an increased appreciation for the power of Machine Learning and its potential as well as the customizability and complexities of GUI development. By partaking in GUI development, Data Manipulation/Visualization creation and Machine Learning development, this project is a clear representation of the power of the Python programming language and its overall integrability.

We had several main goals for this project which we successfully met, and there were a few challenges. We created a fully functional GUI and connected our machine learning models, however, a few features such as an import data button that would allow the user to test data that they would type in directly to the GUI could not be incorporated. We also wanted to make the GUI show that each time you ran an algorithm it was actually computing something instead of printing out a new string after computation. We tried using Tkinter to add a loading bar, but as we are still learning about GUIs, we found it a bit difficult to add some extra features. As a group, we did exactly what we intended which was to create a platform that uses specific data and gives information on it through modeling. The toughest feat was the GUI which forced us to use Object Oriented Principles, learn interface construction, and connect all the pieces of the backend to create a final product. We look to further this project by creating dynamic data visualizations rather than static and creating a more modern GUI design. This was a great

learning experience to help us understand the complexities of Python development. We intend to continue developing the GUI as well as incorporate new Machine Learning Models and visualization.