**Instructions Guide**

1. Install all necessary libraries and packages that includes scikit-learn, numpy, pandas, xgboost, pickle, matplotlib, seaborn, os etc
2. Import all necessary libraries
3. Read input Xcel files
4. Displaying first 5 instances in the dataset using .head() function
5. Check titles of columns in train and test set using .columns() function. Also check how many instances are there in a given dataset along with its shape. In this case training data contains 364 instances having 2 classes therefore its shape is (364,2)
6. Draw a plot for number of males and females instance in the given dataset. Note that dataset is imbalanced
7. Now we will label encode output values in order to transform categorical values to numerical using. This will be done by using sklearn LabelEncoder() method. Declare instance of LabelEncoder and fit\_transform output labels to get label-encoded output. Here note that male ‘m’ = 1 and female ‘f’ = 0.
8. Now we will separate features from labels and save them in train and test variable respectively.
9. For feature extraction we are using sklearn CountVectorizer() method that will extract number of unique features from training data. First declare an instance of CountVectorizer() and them fit\_transform into training data and only transform into test data. Finally print the shape of both train and test to get feature vectors.
10. Split training data along with its labels into 80:20 ratio using sklearn train\_test\_split() function. Here in this case 80% of the total data is used for training and 20% for validation of results. Test size is set to 0.2 with Shuffle = True and Random State = 3 so that for each execution of cells instances after split must be same.
11. Now we are ready to train different ML algorithms to get the best out of them. In this problem we have trained several ML algo’s including KNN, SVC, Random Forest, Naïve Bayes, Multinomial NB, Gaussian NB, xgboost, Gradient Boosting, SGD etc. The process to train each algorithms are the same
12. Initialize an instance of each ML Algo. Call .fit() function and pass x\_train and y\_train variables. Model have trained in this phase. Now in order to validate the results of trained model call .predict() function and pass x\_val variable. We’ll get predicted output the model has returned. Now calculate accuracy and print classification report to get detailed insight on the performance of each trained model on test data. This will be done by calling accuracy\_score(), classification\_report() and confusion\_matrix() respectively and pass actual output y\_val and predicted output i.e y\_pred. print the results and draw confusion matrix using matplotlib.pyplot function.
13. Once different models are trained and tested the algorithm with best f1 score was saved using pickle.dump() function
14. Finally, we will load saved model and make prediction on test data that we haven’t used for training. Note that same feature extraction method was carried out for test data with the difference is that CountVectorizer() will only be transformed not fitted on test data.
15. Apply .predict() function on test data to get prediction. Note that predicted value will be in numerical format. So we will transform numerical data into categorical data before writing the predictions on csv file. This was done by using simple if else statement. You can also apply labelencoder.inverse\_transform(pred) to get inverse of label encoded values in categorical form.
16. Lastly, results are saved in csv file using to\_csv function