Imam Mohammad Ibn Saud Islamic University
College of Computer and Information Sciences
Computer Science Department
Bachelor of Science in Computer Science

CS 360 Assignment Two

Instruction:

- You are required to use Python 3 to complete all the programming assignments.
- All the assignment must be completed in groups (3-4) students.
- You may use Google's Colaboratory service: colab.research.google.com/.

Dataset:

In this assignment we will use the Smarket data. This data set consists of percentage returns for the S&P 500 stock index over 1,250 days, from the beginning of 2001 until the end of 2005. For each date, we have recorded the percentage returns for each of the five previous trading days, Lag1 through Lag5. We have also recorded Volume (the number of shares traded on the previous day, in billions), Today (the percentage return on the date in question) and Direction (whether the market was Up or Down on this date).

Part I: Logistic Regression Classifier

- Fit a logistic regression model in order to predict Direction using Lag1 through Lag5 and Volume. Print the resulting model (coefficients, standard errors, test-statics, p-values).
- Is there an association between Lag1 and Direction? Justifies your answer.
- Display the predicted probabilities for the first ten observations in your dataset (you may use predict() with your fitted model).
- Convert these predicted probabilities into class labels, Up or Down based on whether the predicted probability of a market increase is greater than or less than 0.5 and display the class label for the first ten observations in your dataset.
- Use confusion matrix to evaluate the performance of a logistic regression model, display the matrix and compute the predictive accuracy, precision, recall, and F1-score.

Part II: Naive Bayes Classifier

- Split the data into training and test sets.
- Fit a Gaussian Naive Bayes model to the data.
- Returns an array containing the unique classes present in the target variable.
- Provides the prior probabilities of each class.
- Contains the mean and variance of each feature for each class.
- Uses the trained model to predict the probabilities of each class for the first five samples in the test set.
- Uses the trained model to predict labels for the test set.
- Generate a confusion matrix comparing the predicted labels with the actual labels.
- Uses the trained model to predict the probabilities of each class for the first ficve samples in the test set.