Classifying Spam Messages

This project uses CSV data from a dataset on Kaggle titled *Spam Text Message Classification* (https://www.kaggle.com/datasets/team-ai/spam-text-message-classification). The data are made up of spam and ham (non-spam) text messages along with their classifications. This project aimed to find the best classification model for predicting spam messages. It is organized into three main sections: visualizations, data transformations, and modeling/interpretation. It uses three different classifier models to see which is the best at identifying spam text messages. The end of the project shows a write-up on the interpretation of the model results.

Project Features

This project includes visualizations to find trends in the data, data manipulation and transformation to prepare and clean the data as best as possible for classification modeling, models (including logistic regressions, random forest classifiers, and support vector classifiers, and model interpretations in the form of a write-up.

Installations and Requirements

This project will require the following Python libraries to be imported in order to manipulate and analyze data, transform the data, create visualizations, and run and evaluate models:

- pandas
- matplotlib.pyplot
- wordcloud: WordCloud
- contractions
- string
- nltk: nltk.corpus, nltk.tokenize, nltk.stem.porter
- textblob: TextBlob
- sklearn.feature_extraction.text: CountVectorizer, TfidfVectorizer
- sklearn.preprocessing: StandardScaler
- sklearn.ensemble: RandomForestClassifier
- sklearn.linear_model: LogisticRegression
- sklearn.svm: SVC
- sklearn.model_selection: train_test_split, cross_val_score
- sklearn.metrics: accuracy_score, classification_report

Using the Project

You can use this project in either Jupyter Notebook or any other Python IDE, such as PyCharm. This system could also be run in a Python terminal. However, it is recommended to be used in an IDE, as that is where the script was created and run before. If you wish to use it in Jupyter Notebook, download the .ipynb file for use in your own Jupyter Notebook or copy each cell into your Jupyter Notebook. You may also copy and paste the code into another Python IDE if you prefer a different IDE besides Jupyter Notebook.

The script will generate visualizations to find trends in the data, transform and manipulate the data to prepare it for modeling, model the data using different classification models to determine the best model, and show a write-up discussing the model interpretation at the very end.

Contact

For any questions or concerns, please feel free to contact me, Ahria Dominguez, at ahriadominguez@outlook.com.