CSCI 4961/6961: Homework 1

Assigned Monday September 13 2021. Due by 11:59pm Thursday September 23 2021.

Your answers must be LEGIBLE, clearly labeled, and provide well-written, clear, and convincing arguments. Please start early so you have ample time to see me during office hours.

Submit your code on Submitty as a Jupyter notebook and supply answers to the questions in the order that they were posed. Use LaTeX as appropriate to write mathematical expressions.

We have seen SVMs, logistic regression, and naive Bayes classification models. In this homework you will apply them all to the problem of classifying spam texts.

1. [**10** points] Download the data set in csv format from `http://www.cs.rpi.edu/~gittea/teaching/fall2021/files/spam.csv`[1] to your working directory. Use the following Pandas commands to load the data set as a DataFrame

```
 import pandas as pd

df = pd.read_csv('spam.csv', encoding='latin-1')
df = df.iloc[:, :2]
df = df.replace('ham', 0).replace('spam', 1)
df = df.rename(columns = {'v1': 'label', 'v2': 'message'})
df = df.drop_duplicates().reset_index(drop=True)
```

   Explain what this code does in detail.

2. [**20** points] Convert the messages to vectors of features using SkLearn to tokenize the messages, then convert each message into a vector of counts of the tokens. Split the data set into training and testing sets.

```
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.model_selection import train_test_split

messages = df['message']
y = df['label']
x = CountVectorizer().fit_transform(messages)

x_train, x_test, y_train, y_test = train_test_split(x, y,
                                   test_size=0.2, random_state=2021)
```

   Explain what it means to tokenize the data, and how the tokens are defined by CountVectorizer with its default parameters. How many *unique* tokens are identified in the data set? Does the training data set contain at least one instance of each token?

3. [**20** points] Fit a multinomial naive Bayes model and compute its accuracy, precision, and recall on the test set using SkLearn

```
from sklearn.naive_bayes import MultinomialNB
from sklearn.metrics import precision_score, recall_score

mnb_clf = MultinomialNB()
mnb_clf.fit(x_train, y_ytrain)
y_hat = mnb_clf.predict(x_test)
[mnb_clf.score(x_test, y_test), precision_score(y_test, y_hat), recall_score(y_test, y_hat)]
```

   Explain what accuracy, precision, and recall are, precisely — i.e. give equations and explain them—, and why we care about each of them as a measure of performance of the model.

[1] Source: `https://www.dt.fee.unicamp.br/~tiago/smsspamcollection/`

4. Similarly, fit a linear support vector machine using `sklearn.svm.LinearSVC` and compute its accuracy, precision, and recall. Use the default parameters for the SVM.

5. Simlarly, fit a logistic regression model using `sklearn.linear_model.LogisticRegression` with its default parameters, and compute its accuracy, precision, and recall.

6. [**30** points] Compare the performance of the models by showing the three metrics in a table. Does anything about these results surprise you? Which model would you use in practice, and why?

7. [**20** points] Look into the provenance of this data set. What might be some caveats in using this particular data set to construct a spam filtering service for T-Mobile in 2021?

CSCI6961 students only. [**25** points] In class we gave an expression for $p(y \mid \mathbf{x})$ for naive Bayes. Similarly, give an expression for the model $y \mid \mathbf{x}$ for the multinomial naive Bayes model[2] and explain the model in words. Explain why, given a finite amount of data, the multinomial naive Bayes model is expected to work better than the naive Bayes model we introduced in class[3].

[2] See the paper ”Spam filtering: how the dimensionality reduction affects the accuracy of Naive Bayes classifiers”.

[3] Note that the NB model introduced in class differs from the one presented in section 4.1 of the referenced paper.