Notebook

April 23, 2020

0.0.1 Question 6c

Provide brief explanations of the results from 6a and 6b. Explain why the number of false positives, number of false negatives, accuracy, and recall all turned out the way they did.

Since the zero predictor is predicting 0 for every email, this means that there are no positive (1) predictions. Thus, there will be neither false positive, nor true positives as there are no positive predictions at all. Instead, the number of true negatives will be equal to the number of actual ham emails, and the number of false negatives will be equal to the number of actual spam emails. Thus, accuracy will be the number of correct values over the total number of values in general which is just the number of true negatives which we know is just the number of actual ham emails all over the number of total emails. Then for recall, we know that the number of true positives is 0 so our work is done right there!

0.0.2 Question 6e

Are there more false positives or false negatives when using the logistic regression classifier from Part A?

There are more false positives when using logistic regression classifier in comparison to the zero predictor classifier and less false negatives since the logistic regression classifier will actually identify some of the spam emails in comparison to the zero predictor classifier which will miss all the spam emails.

0.0.3 Question 6f

1. Our logistic regression classifier got 75.8% prediction accuracy (number of correct predictions / total). How does this compare with predicting 0 for every email?
2. Given the word features we gave you above, name one reason this classifier is performing poorly. Hint: Think about how prevalent these words are in the email set.
3. Which of these two classifiers would you prefer for a spam filter and why? Describe your reasoning and relate it to at least one of the evaluation metrics you have computed so far.

1. An accuracy of ab75% means that we're only doing slightly better than guessing ham for every email.
2. The words we used in our classifier: ['drug', 'bank', 'prescription', 'memo', 'private'], may not be good indicators of spam/ham emails as a result of not being very prevalent in either of the spam/ham email sets, thus not being good variables to predict over.
3. I would prefer the logistic regression classifier since it will actually catch some of the spam emails versus the zero predictor classifier which will let all of the spam emails through. We can see a direct numerical result of this in the recall value of the two sets, where the zero predictor classifier has a recall of 0 meaning it didn't catch any of the spam emails versus the logistic regression classifier which managed to catch about 11% of the actual spam emails.

0.0.4 Question 7: Feature/Model Selection Process

In the following cell, describe the process of improving your model. You should use at least 2-3 sentences each to address the follow questions:

1. How did you find better features for your model?
2. What did you try that worked / didn't work?
3. What was surprising in your search for good features?

1. I found better features for my model by observing some patterns that I saw in the data and then running these features through 50-fold cross-validation.

2. First, I noticed that many spam emails have lots of capital letters, and so I created a method which returns an array of the number of capital letters in each email. I also took into account the total body length of the email itself and whether or not the email is a reply and contains the string 'Re:' in the subject line. Finally, my last set of features were garnered by researching the most common words in english called stop-words and running the words_in_texts method on them to generate a set of features.

3. I was surprised by the 50-fold cross validation results because the model was not too overfitt, despite having quite a few features, and will yield results of 88% or above accuracy.

Generate your visualization in the cell below and provide your description in a comment.

In [78]:
```
# Write your description (2-3 sentences) as a comment here:
# I graphed the accuracy of the predictions based on the number of features included in the mo
# Thus we can see that as the number of features increases, the accuracy is also increasing
# I also plotted a chart of the variances and bias to see how the relationship between these a
# features increased

# Write the code to generate your visualization here:
def bias(x):
    return 1 / len(x) * sum((Y_train - x)**2)

accuracy_scores = []
variances = []
biases = []

x = np.arange(1,len(indicator_words)+1,40)

for i in x:
    new_indicator_words = indicator_words[:i]
    word_array = words_in_texts(new_indicator_words, text)
    processed_Xtrain = pd.DataFrame(word_array).assign(capitals = num_caps, length = email_leng
    accuracy_scores.append(compute_CV_error(model, processed_Xtrain, Y_train))

    y_hat = model.predict(processed_Xtrain)
    variance = np.var(y_hat)
    bias2 = bias(y_hat)

    variances.append(variance)
    biases.append(bias2)
    print('bias for',i,':', bias2)
    print('variance for',i, ':', variance)

plt.plot(x ,accuracy_scores)
plt.plot(x, variances)
plt.plot(x, biases)
plt.legend(['accuracy', 'variances', 'biases'])
plt.xlabel('Number of features')
plt.show()
```

```
error: 8.422317490505153
accuracy: 0.7561545253863136
bias for 1 : 0.24384400372687343
variance for 1 : 0.01520151352545334


/srv/conda/envs/data100/lib/python3.7/site-packages/sklearn/linear_model/_logistic.py:940: ConvergenceW
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

… Omitting 122 lines …
    https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression
  extra_warning_msg=_LOGISTIC_SOLVER_CONVERGENCE_MSG)
/srv/conda/envs/data100/lib/python3.7/site-packages/sklearn/linear_model/_logistic.py:940: ConvergenceW
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.
```

```
… Omitting 122 lines …
    https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression
  extra_warning_msg=_LOGISTIC_SOLVER_CONVERGENCE_MSG)
/srv/conda/envs/data100/lib/python3.7/site-packages/sklearn/linear_model/_logistic.py:940: ConvergenceWa
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

… Omitting 26 lines …
    https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression
  extra_warning_msg=_LOGISTIC_SOLVER_CONVERGENCE_MSG)


error: 5.709859658354571
accuracy: 0.8346852097130244
bias for 41 : 0.1641155330760016
variance for 41 : 0.15939202874917427


/srv/conda/envs/data100/lib/python3.7/site-packages/sklearn/linear_model/_logistic.py:940: ConvergenceWa
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

… Omitting 122 lines …
    https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression
  extra_warning_msg=_LOGISTIC_SOLVER_CONVERGENCE_MSG)
/srv/conda/envs/data100/lib/python3.7/site-packages/sklearn/linear_model/_logistic.py:940: ConvergenceWa
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

… Omitting 122 lines …
    https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression
  extra_warning_msg=_LOGISTIC_SOLVER_CONVERGENCE_MSG)
/srv/conda/envs/data100/lib/python3.7/site-packages/sklearn/linear_model/_logistic.py:940: ConvergenceWa
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

… Omitting 122 lines …
    https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression
  extra_warning_msg=_LOGISTIC_SOLVER_CONVERGENCE_MSG)
/srv/conda/envs/data100/lib/python3.7/site-packages/sklearn/linear_model/_logistic.py:940: ConvergenceWa
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

… Omitting 10 lines …
    https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression
  extra_warning_msg=_LOGISTIC_SOLVER_CONVERGENCE_MSG)


error: 4.891006344206212
accuracy: 0.8583929359823399
bias for 81 : 0.14148808731532012
variance for 81 : 0.16966061347074718


/srv/conda/envs/data100/lib/python3.7/site-packages/sklearn/linear_model/_logistic.py:940: ConvergenceWa
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

… Omitting 122 lines …
    https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression
  extra_warning_msg=_LOGISTIC_SOLVER_CONVERGENCE_MSG)
```

```
/srv/conda/envs/data100/lib/python3.7/site-packages/sklearn/linear_model/_logistic.py:940: ConvergenceWa
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

… Omitting 122 lines …
    https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression
  extra_warning_msg=_LOGISTIC_SOLVER_CONVERGENCE_MSG)
/srv/conda/envs/data100/lib/python3.7/site-packages/sklearn/linear_model/_logistic.py:940: ConvergenceWa
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

… Omitting 122 lines …
    https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression
  extra_warning_msg=_LOGISTIC_SOLVER_CONVERGENCE_MSG)
/srv/conda/envs/data100/lib/python3.7/site-packages/sklearn/linear_model/_logistic.py:940: ConvergenceWa
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

… Omitting 10 lines …
    https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression
  extra_warning_msg=_LOGISTIC_SOLVER_CONVERGENCE_MSG)


error: 4.211138286222081
accuracy: 0.8780768211920528
bias for 121 : 0.11087448422733928
variance for 121 : 0.17041337939833254
```