Homework 3

The code is submitted at github

https://github.com/mas-dse/w9yan/blob/master/DSE220/homeworks/homework_3/logisticregression_svm.ipynb

Answers:

1. Selected Parameters {'C': 500, 'penalty': 'l1'}
   Test Accuracy = 0.861111111111

   Data loading:

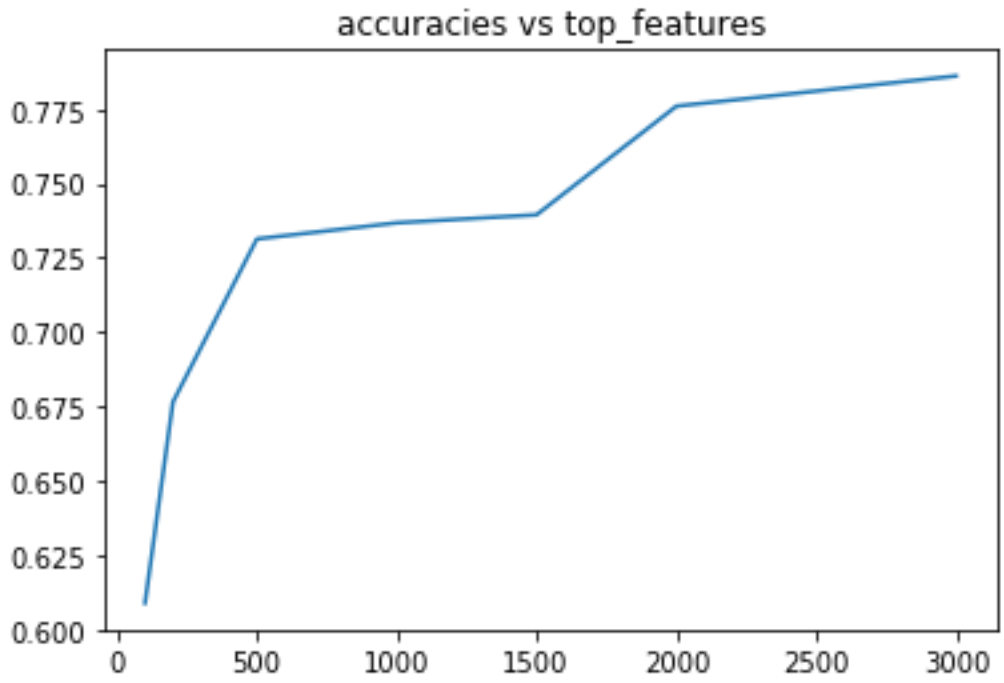
```
from sklearn.datasets import fetch_20newsgroups
cats = ['alt.atheism', 'comp.graphics', 'sci.space', 'talk.politics.mideast']
newsgroups_train = fetch_20newsgroups(subset='train', categories=cats, remove=('headers', 'footers', 'quotes'))
newsgroups_test = fetch_20newsgroups(subset='test', categories=cats, remove=('headers', 'footers', 'quotes'))
y_train = newsgroups_train.target
y_test = newsgroups_test.target

from sklearn.feature_extraction.text import TfidfVectorizer
vectorizer =
TfidfVectorizer(analyzer=u'word',lowercase=True,stop_words='english',smooth_idf=True,max_features=2000)
# fit and transform on train data
X_train = vectorizer.fit_transform(newsgroups_train.data)
# just transform test data with previous fitting
X_test = vectorizer.transform(newsgroups_test.data)
```

2. Perceptron(no penalty) test accuracy = 0.776048714479

3. Varying top feature count for Perceptron model.

```
Perceptron test accuracy with top 100 features: 0.6089
Perceptron test accuracy with top 200 features: 0.6766
Perceptron test accuracy with top 500 features: 0.7314
Perceptron test accuracy with top 1000 features: 0.7368
Perceptron test accuracy with top 1500 features: 0.7395
Perceptron test accuracy with top 2000 features: 0.7760
Perceptron test accuracy with top 3000 features: 0.7862
```

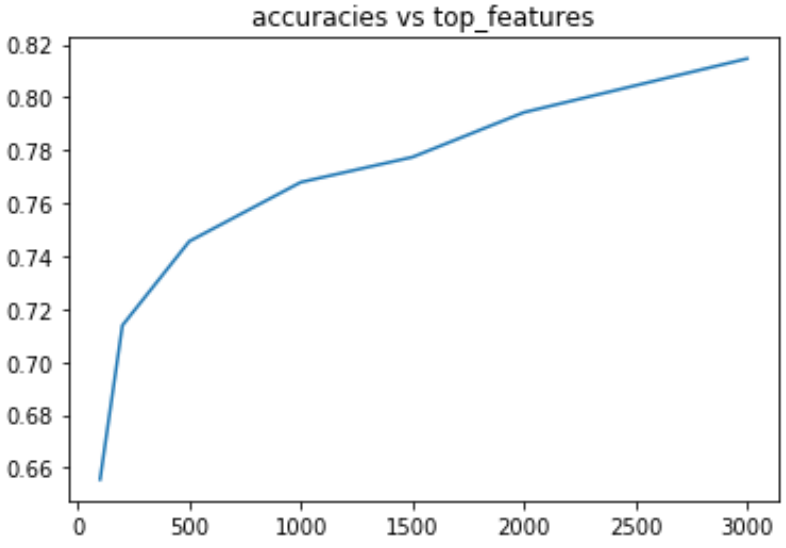


4. Train SVM model with training data and report test accuracy:

```
SVM test accuracy = 0.794316644114
```

5. Varying top feature count for Perceptron model.

```
SVM test accuracy with top 100 features: 0.6556
SVM test accuracy with top 200 features: 0.7138
SVM test accuracy with top 500 features: 0.7456
SVM test accuracy with top 1000 features: 0.7679
SVM test accuracy with top 1500 features: 0.7774
SVM test accuracy with top 2000 features: 0.7943
SVM test accuracy with top 3000 features: 0.8146
```



6. SVM: tune the cost parameter 'C' for values 0.01,0.1,1,10,100.

```
SVM with C:0.01 validation accuracy: 0.2449438202247191
SVM with C:0.1 validation accuracy: 0.7752808988764045
SVM with C:1 validation accuracy: 0.8359550561797753
SVM with C:10 validation accuracy: 0.8247191011235955
SVM with C:100 validation accuracy: 0.8089887640449438
SVM with best C:1 test accuracy: 0.7943166441136671
```

7. SVM(with C=10000): tune kernel values - 'poly' with degree 1, 2, 3, 'rbf ' and 'sigmoid'.

```
SVM with kernel:poly degree:1 validation accuracy: 0.8292134831460675
SVM with kernel:poly degree:2 validation accuracy: 0.2449438202247191
SVM with kernel:poly degree:3 validation accuracy: 0.2449438202247191
SVM with kernel:rbf validation accuracy: 0.8247191011235955
SVM with kernel:sigmoid validation accuracy: 0.8292134831460675
SVM with best kernel:poly test accuracy: 0.2665764546684709
```

8. Custom kernel with cosine similarity and Laplacian

```
SVM test accuracy with kernel cosine_similarity is
0.7943166441136671
SVM test accuracy with kernel laplacian_kernel is 0.2665764546684709
```

9. Combination of kernels

The combination will be a valid kernel. Since K1(x,y) and K2(x,y) are valid kernels, according to Mercer's condition, for any finite subset, the similarity matrix with K1 and K2 are PSD, and the combination K(x,y)=a*K1(x,y) + (1-a)*K2(x,y) will also produce a PSD similarity matrix for any finite subset, because a and 1-a are greater or equal to 0.

This won't hold true for other values of a, because that coulds result in a negative coefficient for kernel K1 or K2.

```
SVM with coefficient a:0.0 validation accuracy 0.2449438202247191
SVM with coefficient a:0.1 validation accuracy 0.7820224719101123
SVM with coefficient a:0.2 validation accuracy 0.8179775280898877
SVM with coefficient a:0.3 validation accuracy 0.8382022471910112
SVM with coefficient a:0.4 validation accuracy 0.8404494382022472
SVM with coefficient a:0.5 validation accuracy 0.8359550561797753
SVM with coefficient a:0.6 validation accuracy 0.8426966292134831
SVM with coefficient a:0.7 validation accuracy 0.8471910112359551
SVM with coefficient a:0.8 validation accuracy 0.8471910112359551
SVM with coefficient a:0.9 validation accuracy 0.8404494382022472
SVM with coefficient a:1.0 validation accuracy 0.8359550561797753
SVM with best combination a:0.7 test accuracy: 0.8010825439783491
```