Introduction to Data Science

Homework 5

Student Name: Eric Niblock

Student Netid: ejn9259

Part 1: Naive Bayes (5 Points)

1. From your reading you know that the naive Bayes classifier works by calculating the conditional probabilities of each feature, e_i, occuring with each class c and treating them independently. This results in the probability of a certain class occuring given a set of features, or a piece of evidence, E, as

$$P(c \mid E) = \frac{p(e_1 \mid c) \cdot p(e_2 \mid c) \cdot \cdot \cdot p(e_k \mid c) \cdot p(c)}{p(E)}.$$

The conditional probability of each piece of evidence occuring with a given class is given by

$$P(e_i \mid c) = \frac{\text{count}(e_i, c)}{\text{count}(c)}.$$

In the above equation $\text{count}(e_i, c)$ is the number of documents in a given class that contain feature e_i and $\text{count}(c)$ is the number of documents that belong to class c.

A common variation of the above is to use Laplace (sometimes called +1) smoothing. Recall the use of Laplace smoothing introduced toward the end of Chapter 3 in the section Probability Estimation. This is done in sklearn by setting `alpha=1` in the `BernoulliNB()` function (this is also the default behavior). The result of Laplace smoothing will slightly change the conditional probabilities,

$$P(e_i \mid c) = \frac{\text{count}(e_i, c) + 1}{\text{count}(c) + 2}.$$

In no more than **one paragraph**, describe why this is useful, and use the bias-variance tradeoff to justify its use. Try to think of a case when not using Laplace smoothing would result in "bad" models. Try to give an example. Be precise.

--------ANSWER---------

Laplace smoothing is useful because without it, we would have a tendency to lose information. If we have some feature e_i that never occurs in conjunction with some class c, we then lose information related to all of the other features, that probably have a stronger relationship with c (this is evident from the first equation, where if some $p(e_i \mid c) = 0$, the whole expression results in

zero). Laplace smoothing never allows $P(e_i \mid c) = 0$, and hence we retain information. Smoothing our empirical data results in lower variance, as the influence of the data itself becomes more muddled. This could tentatively lead to higher bias, though this is acceptable, as it often leads to a dramatic reduction in variance. Say we trained a model to classify the sentiment of a sentence as either positive or negative and one of the features was the presence of the word "bad," appearing once in a datum associated with the negative class. Our classifier would then never predict the positive class if a datum contained the word "bad" (assuming no Laplace smoothing) though there is obviously tons of positive sentiments that employ the word "bad" (i.e. "not bad").

--------ANSWER---------

Part 2: Text classification for sentiment analysis (20 Points)

For this part of the assignment, we are going to use a data set of movie ratings from IMDB.com. The data consists of the text of a movie review and a target variable which tells us whether the reviewer had a positive feeling towards the movie (equivalent to rating the movie between 7 and 10) or a negative feeling (rating the movie between 1 and 4). Neutral reactions are not included in the data.

The first column is the review text; the second is the text label 'P' for positive or 'N' for negative.

1 (1 Point) . Load the data into a pandas `DataFrame()` .

In [2]:
```
import pandas as pd
import numpy as np
data = pd.read_csv(r'imdb.csv')
```

2 (1 Point). Code the target variable to be numeric: use the value `1` to represent 'P' and `0` to represent 'N'.

In [3]:
```
data['Class_Binary'] = np.where(data['Class']=='P', 1, 0)
data
```

Out[3]:

	Text	Class	Class_Binary
0	'One of the first of the best musicals Anchors...	P	1
1	'Visually disjointed and full of itself the di...	N	0
2	'These type of movies about young teenagers st...	P	1
3	'I would rather of had my eyes gouged out with...	N	0
4	'The title says it all. Tail Gunner Joe was a ...	N	0
...	...	...	...
8495	'Alright friends a serious movie buff is expec...	N	0
8496	'I found this film embarrassing to watch. I fe...	N	0
8497	'To put it simply I am not fond of westerns. A...	N	0
8498	'Some of these viewer comments are just ridicu...	N	0
8499	'Sometimes a premise starts out good but becau...	N	0

8500 rows × 3 columns

3 (2 Points). Put all of the text into a data frame called `X` and the target variable in a data frame called `Y`. Make a train/test split where you give 75% of the data to training. Feel free to use any function from sklearn.

In [101]:
```
import sklearn.model_selection as sk
X = data['Text']
Y = data['Class_Binary']
X_train, X_test, y_train, y_test = sk.train_test_split(X,Y, test_size=0.25, randc
```

4 (5 Points). Create a binary `CountVectorizer()` and a binary `TfidfVectorizer()`. Use the original single words as well as bigrams (in the same model). Also, use an "english" stop word list. Fit these to the training data to extract a vocabulary and then transform both the train and test data. Hint - look at the API documentation for both vectorizers to see what we mean by "binary."

In [102]:
```
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.feature_extraction.text import TfidfVectorizer

c_vectorizer = CountVectorizer(binary=True, stop_words='english', ngram_range=(1,
X_train_cvect = c_vectorizer.fit_transform(X_train)
X_test_cvect = c_vectorizer.transform(X_test)

t_vectorizer = TfidfVectorizer(binary=True, stop_words='english', ngram_range=(1,
X_train_tvect = t_vectorizer.fit_transform(X_train)
X_test_tvect = t_vectorizer.transform(X_test)
```

5 (6 Points). Create `LogisticRegression()` and `BernoulliNB()` models. For all settings, keep the default values. In a single plot, show the AUC curve for both classifiers and both

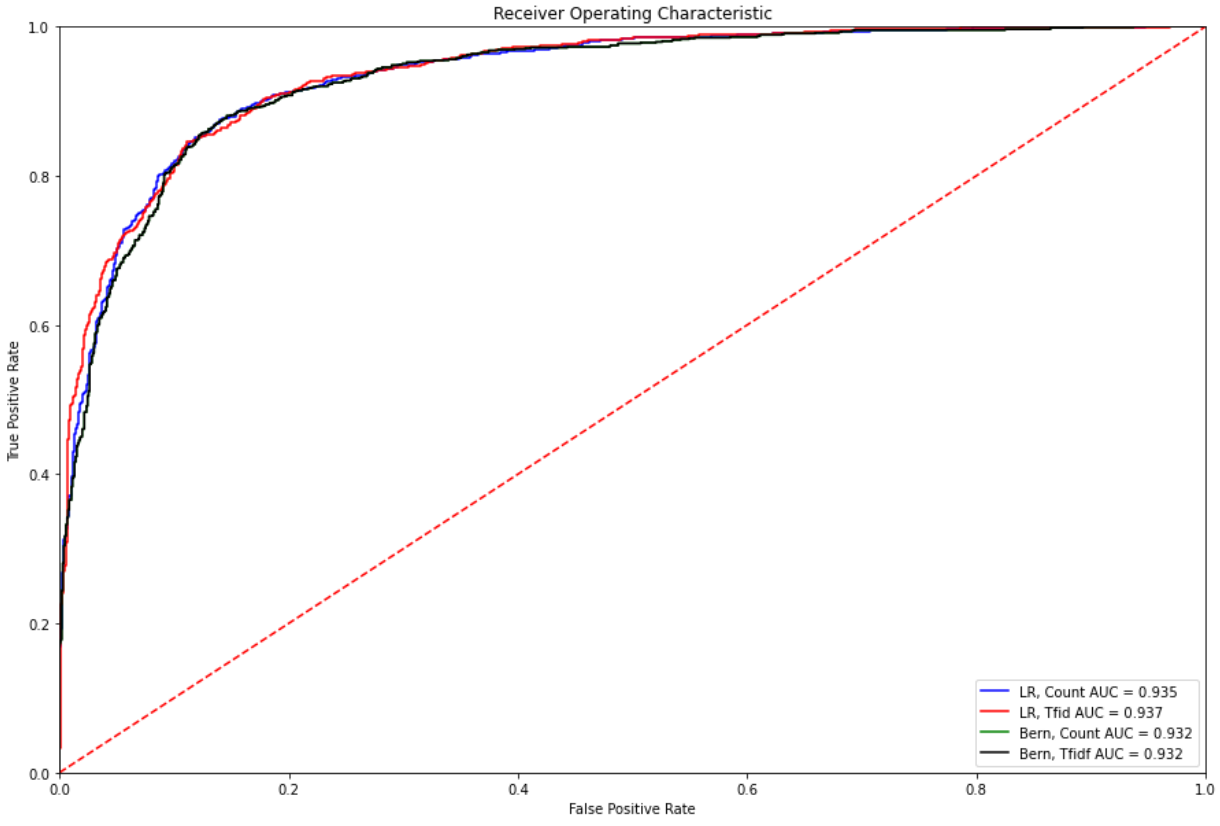
vectorizers defined above. In the legend, include the area under the ROC curve (AUC). Do not forget to label your axes. Your final plot will be a single window with 4 curves.

Which model do you think does a better job? Why? Explain in no more than a paragraph.

Extra credit (2 points): Do any of the options perform identically? If so, can you explain why?

In [103]:
```
# Run this so your plots show properly
import matplotlib.pyplot as plt
%matplotlib inline
plt.rcParams['figure.figsize'] = 12, 12
```

In [104]:
```
from sklearn.linear_model import LogisticRegression
from sklearn.naive_bayes import BernoulliNB
from sklearn import metrics

clf_c = LogisticRegression().fit(X_train_cvect, y_train)
clf_t = LogisticRegression().fit(X_train_tvect, y_train)
clf_cb = BernoulliNB().fit(X_train_cvect, y_train)
clf_tb = BernoulliNB().fit(X_train_tvect, y_train)
clfs = [clf_c, clf_t, clf_cb, clf_tb]

plt.figure(figsize = (15,10))
plt.title('Receiver Operating Characteristic')

for i in range(len(clfs)):

    vectors = [X_test_cvect,X_test_tvect,X_test_cvect,X_test_tvect]
    y_pred_prob = clfs[i].predict_proba(vectors[i])[:,1]
    fpr, tpr, threshold = metrics.roc_curve(y_test, y_pred_prob)
    roc_auc = metrics.auc(fpr, tpr)
    color = ['b','r','g','k']
    labels = ['LR, Count AUC','LR, Tfid AUC','Bern, Count AUC', 'Bern, Tfidf AUC'
    plt.plot(fpr, tpr, color[i], label = labels[i] +' = %0.3f' % roc_auc)

plt.legend(loc = 'lower right')
plt.plot([0, 1], [0, 1],'r--')
plt.xlim([0, 1])
plt.ylim([0, 1])
plt.ylabel('True Positive Rate')
plt.xlabel('False Positive Rate')
plt.show()
```



--------ANSWER---------

The Bernoulli Naive Bayes models performed identically. This is because Bernoulli Naive Bayes gives equal weight to all features, and so regardless of whether the input came from CountVectorizer or TfidfVectorizer, the input was treated as binary (0 or 1) for whether the feature was present or absent.

--------ANSWER---------

5. Use the model from question 4 that you think did the best job and predict the rating of the test data. Find 5 examples were labeled positive, but were incorrectly classified as negative. Print out the reviews below and include an explanation as to why you think it may have been incorrectly classified. You can pick any 5. They do not have to be at random.

In [117]:
```
y_pred = clf_t.predict(X_test_tvect)
results = pd.DataFrame(list(zip(X_test,y_test,y_pred)), columns =['Text', 'Actual
badresults = results[(results['Actual'] == 1) & (results['Predicted'] == 0)]

for i in [432,83,1715,1838,541]:
    print('---------------------------')
    print(badresults.loc[i][0])
    print('---------------------------')
```

```
---------------------------
'I wouldnt go so far as to not recommend this movie since the only problems I h
ave with it are due to an overexposure to the plot devices used in the movie -
the sort of things common to every kids movie ever made it seems. That doesnt m
ake it bad just not something Id go far. It is a little saccharine so I might s
ay that for the most part anyone looking for something with a little more wit c
ould be disappointed in an obviously for-kids movie like this. However all of t
hat goes out the window when that squirrel (the one in all the trailers) comes
on-screen. His time is limited but it seems apparent that the decision makers h
ad the wisdom to tell these guys hey could you stick in a little more squirrel?
every time its getting intolerably dull. That doesnt save the movie but you can
leave saying at least there was one aspect where I couldnt stop laughing. And o
f course visually it wont disappoint but thats almost a given with Pixar flick
s. Of all of their stuff Id put this at the bottom...but that isnt in itself ba
d.'
---------------------------
---------------------------
'This is a bit of a puzzle for a lot of the artsy Lynch crowd. They tend to try
to write this off as some kind of meaningless crude side project of Lynchs. Lik
e this is Lynch passing gas between his real pieces of film art. Well it may be
a fart but its one of those intriguing farts that you catch of a whiff of and a
re embarrassed to admit you enjoy. Dumbland distilled down beyond this is art.
What can you do with aspects of modern life but laugh at it. If you took it ser
iously you would go nuts. You hook into it smell it taste it feel its agonies i
ts unreasoning stupidities and then express it in any medium you choose. Thats
called art and art isnt dumb. But it is Dumbland.'
---------------------------
---------------------------
'I was really excited about seeing this film. I thought finally Australia had m
ade a good film.. but I was wrong. This was the most pathetic attempt at a slas
her film ever. I feel sorry for Molly Ringwald having to come all the way to Au
stralia to make an awful movie. The acting was terrible (especially that Austra
lian guy who was trying to speak in an American accent) and the plot was also p
retty bad. When I first heard about this film coming out I thought that the tit
le was pathetic (because it sounds like the cheesy film Stab in Scream 2) but I
was willing to let it slide if it was a good movie. WARNING!!! MAJOR SPOILER
S!!! Probably the worst thing about the film was the ending. I was expecting a
big surprise about who the killer was.. but the killer wasnt even human.. which
turned this realistic slasher film into an awful horror movie. Dont see this fi
lm.. youll probably be disappointed!'
---------------------------
---------------------------
'My wife is a teacher and she is very familiar with the story having read it to
several of her classes. It never sounded all that interesting to me though and
I bought the DVD figuring this would be a movie that wouldnt really be up my al
ley. The first half of the movie has a lot of set-up and I found myself thinkin
g that I was right. It starts off a bit slow and I have to admit that I was a l
ittle bit bored - but curious enough to stay with it. Boy am I glad I did becau
```

```
se this ended up being a very satisfying and rewarding movie. I would most cert
ainly watch this again! The casting was very good. Since I havent read the book
I cant vouch for accuracy but I have to say that Jon Voight was truly delightfu
l. You liked the characters you were supposed to like hated the ones you were s
upposed to hate and laughed at the ones that were supposed to be funny. I can s
ee how some folks might not like this movie. It is tedious at times especially
in the beginning. All the flashbacks can be distracting (though they are essent
ial to the story). Once the story starts to come together at the end though I t
hink youre paid back in spades for your patience. When all is said and done I t
hink this is a very good movie - 8/10. '
---------------------------
---------------------------
'Heres the skinny it seems that this is much older then I thought it was. But i
ts still cool. The bike mechs are cool and the story works for the most part. T
here are some character issues that I hope work themselves out by part 2 and my
biggest complaint of all that it seems to be a MACROSS knock off. Not just the
animation style but several character designs. For example all the girls in thi
s movie look like LYN MINMAY of MACROSS. The mechs look similar to MACROSS as w
ell as the other characters. This is really not made for little kids it has gra
phic violence nudity and graphic sexual content. So to make a long story short
I give this cool MACROSS knock-off 7 STARS.'
---------------------------
```

--------ANSWER---------

These reviews seem to be negatively classified because of their use of negative language, often times expressed when offering a critique of the movie/show. Though all of the reviews are ultimately positive (with the exception of (3), which was evidently mislabeled), the portions related to a critique often carry lots of negative verbiage that most likely confuses the classifier into thinking that the overall sentiment is negative. Here are some of the negative aspects of each review:

(1) "the only problems I have with it are due to an overexposure"
(2) "Well it may be a fart"
(3) "Dont see this film.. youll probably be disappointed!"
(4) "It starts off a bit slow and I have to admit that I was a little bit bored"
(5) "my biggest complaint of all..."

Ultimately, it was likely lines like these and the use of negative language that probably swayed the classifier to declare these as negative comments. Also, some data is probably mislabeled (like (3)).

--------ANSWER---------