Student(s) Name: Cavit Cakir

CS412 Machine Learning
HW 3 – Text Classification: Logistic Regression and Naive Bayesian
100pts

- **Please TYPE your answer.**
- **Use this document to type in your answers** (rather than writing on a separate sheet of paper), to keep questions, answers and and grades together so as to facilitate grading.
- **SHOW all your work for partial/full credit.**

Goal:

1. By using gaussian distributed artificial dataset with two cluster, makes the decision boundary and conditional independence assumption clearer.
2. The dataset contains around 200k news headlines from the year 2012 to 2018 obtained from HuffPost, make a classification of 5 hot topics by Naive Bayesian and Logistic Regression.

Grading: The algorithmic parts needs to be supported by discussions. In both parts of the homework, it is very important to discuss Naive Bayesian and Logistic Regression differences. The aim here is to make sure that you can follow a good ML experimental methodology (as taught in HW1); know the weaknesses/strengths and requirements of each classifier for a given problem and that you are able to assess and report your results clearly and concisely.

Data:

1. It is expected to generate two artificial datasets. In each of the data points, they are drawn from Gaussian distributions with different standard deviations.
2. This dataset contains around 200k news headlines from the year 2012 to 2018 obtained from HuffPost. Politics, Wellness, Entertainment and Travel topics are selected for processing. Split in two subsets: one for training (or development) and the other one for testing (or for performance evaluation). The split between the train and test set is based upon a messages posted before and after a specific date.

Software: You may find the necessary function references here:

https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.TfidfTransformer.html

https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html

https://scikit-learn.org/stable/modules/generated/sklearn.naive_bayes.GaussianNB.html

Submission: Fill and submit this document with a link to your Colab notebook (make sure to include the link obtained from the **share link on top right**)

Please follow the instructions of the notebook:

https://colab.research.google.com/drive/1tkKUs1MmR0sMW3OXnfD-3B3upMZ61zJD

Question 1) 25pts – Use a artificial dataset to clarify decision boundary and conditional independence assumption.

a) 10pts - What is the test set performance for Naive Bayesian and Logistic Regression with different standard deviation? Print the confusion matrix, classification report.

Both algorithms are giving much the same results. Higher standard deviation gives worse accuracy, due to the randomness of data points.

```
[18]  1 # Predict
      2 print("Classification Report for Naive Bayesian:")
      3 print(classification_report(val_y1,GNB.predict(val_x1)))
      4 print("Confusion matrix for Naive Bayesian:")
      5 print(confusion_matrix(val_y1, GNB.predict(val_x1)))

Classification Report for Naive Bayesian:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00        45
           1       1.00      1.00      1.00        55

    accuracy                           1.00       100
   macro avg       1.00      1.00      1.00       100
weighted avg       1.00      1.00      1.00       100

Confusion matrix for Naive Bayesian:
[[45  0]
 [ 0 55]]
```

```
[19]  1 # Predict
      2 print("Classification Report for Logistic Regression:")
      3 print(classification_report(val_y1,clf.predict(val_x1)))
      4 print("Confusion matrix for Logistic Regression:")
      5 print(confusion_matrix(val_y1, clf.predict(val_x1)))

Classification Report for Logistic Regression:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00        45
           1       1.00      1.00      1.00        55

    accuracy                           1.00       100
   macro avg       1.00      1.00      1.00       100
weighted avg       1.00      1.00      1.00       100

Confusion matrix for Logistic Regression:
[[45  0]
 [ 0 55]]
```

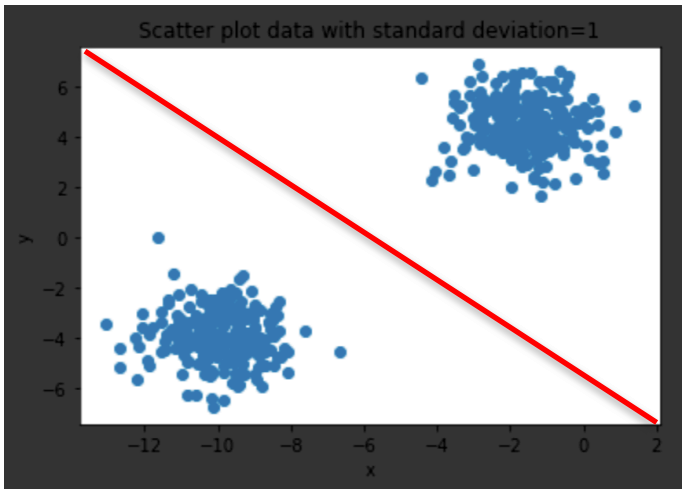
```
[22]  1 # Predict
      2 print("Classification Report for Naive Bayesian:")
      3 print(classification_report(val_y2,GNB.predict(val_x2)))
      4 print("Confusion matrix for Naive Bayesian:")
      5 print(confusion_matrix(val_y2, GNB.predict(val_x2)))

Classification Report for Naive Bayesian:
              precision    recall  f1-score   support

           0       0.84      0.82      0.83        50
           1       0.82      0.84      0.83        50

    accuracy                           0.83       100
   macro avg       0.83      0.83      0.83       100
weighted avg       0.83      0.83      0.83       100

Confusion matrix for Naive Bayesian:
[[41  9]
 [ 8 42]]
```

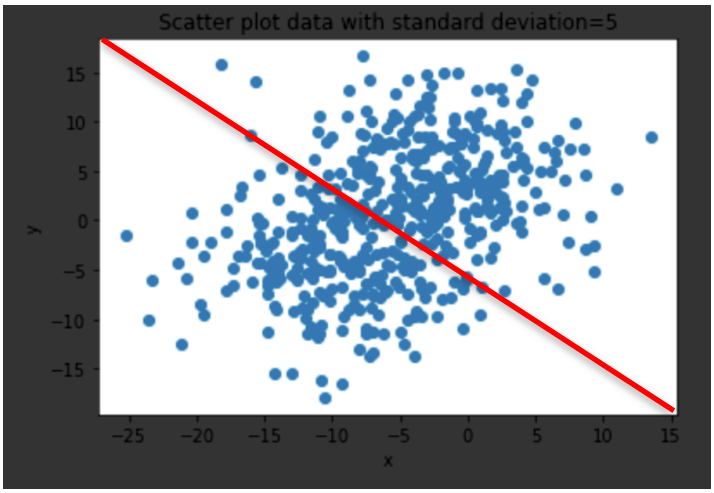
```
[24]  1 # Predict
      2 print("Classification Report for Logistic Regression:")
      3 print(classification_report(val_y2,clf.predict(val_x2)))
      4 print("Confusion matrix for Logistic Regression:")
      5 print(confusion_matrix(val_y1, clf.predict(val_x1)))

Classification Report for Logistic Regression:
              precision    recall  f1-score   support

           0       0.83      0.78      0.80        50
           1       0.79      0.84      0.82        50

    accuracy                           0.81       100
   macro avg       0.81      0.81      0.81       100
weighted avg       0.81      0.81      0.81       100

Confusion matrix for Logistic Regression:
[[45  0]
 [ 0 55]]
```

b) 10pts - Discuss the reason behind why Gaussian Naive Bayesian works better for artificial dataset with the concept of conditional independence.

Gaussian Naive Bayesian algorithm assumes the dataset has a conditional independence and makes choices according to this assumption. Our artificial dataset gives completely conditional independent dataset which makes Gaussian Naive Bayesian works better. In real world, it's really hard to find conditional independent datasets which makes Gaussian Naive Bayesian works not as intended.

c) 5pts - Draw the perfect decision boundary for the dataset on the scatter plots.





Question 2) 20pts – Use a Gaussian Naive Bayesian

Import Kaggle dataset and filter 4 principle topics, Politics, Wellness, Entertainment and Travel. Sample 50000 rows from the data. The occurances of the topics,

POLITICS	8246
WELLNESS	4352
ENTERTAINMENT	3951
TRAVEL	2426

Merge the `short_description and headline` cells of the corresponding row to use as text to process.

a) 15pts - What is the best test set performance you obtained by Gaussian Naive Bayesian?

0.71

b) 5pts – Print the confusion matrix, classification report.

```
[50]  1 from sklearn.naive_bayes import GaussianNB
      2 GNB = GaussianNB()
      3 GNB.fit(train_x.toarray(), train_y)
      4
      5 print("Classification Report for Naive Bayesian:")
      6 print(classification_report(val_y,GNB.predict(val_x.toarray())))
      7
      8 print("Confusion matrix for Naive Bayesian:")
      9 print(confusion_matrix(val_y, GNB.predict(val_x.toarray())))
```

```
Classification Report for Naive Bayesian:
              precision    recall  f1-score   support

           0       0.78      0.77      0.77      1649
           1       0.72      0.69      0.71       871
           2       0.63      0.68      0.65       789
           3       0.58      0.58      0.58       493

    accuracy                           0.71      3802
   macro avg       0.68      0.68      0.68      3802
weighted avg       0.71      0.71      0.71      3802

Confusion matrix for Naive Bayesian:
[[1265   93  214   77]
 [ 135  600   62   74]
 [ 132   60  537   60]
 [  92   75   39  287]]
```

Question 2) 20pts – Use a Logistic Regression

Import Kaggle dataset and filter 4 principle topics, Politics, Wellness, Entertainment and Travel. Sample 50000 rows from the data. The occurances of the topics,

POLITICS	8246
WELLNESS	4352
ENTERTAINMENT	3951
TRAVEL	2426

Merge the `short_description and headline` cells of the corresponding row to use as text to process.

a) 15pts - What is the best test set performance you obtained by Logistic Regression?

9

b) 5pts – Print the confusion matrix, classification report.

```
# Predict
print("Classification Report for Logistic Regression:")
print(classification_report(val_y, clf.predict(val_x.toarray())))

print("Confusion matrix for Logistic Regression:")
print(confusion_matrix(val_y, clf.predict(val_x.toarray())))
```

```
Classification Report for Logistic Regression:
              precision    recall  f1-score   support

           0       0.91      0.96      0.93      1649
           1       0.89      0.91      0.90       871
           2       0.86      0.86      0.86       789
           3       0.96      0.76      0.85       493

    accuracy                           0.90      3802
   macro avg       0.91      0.87      0.89      3802
weighted avg       0.90      0.90      0.90      3802

Confusion matrix for Logistic Regression:
[[1581   35   31    2]
 [  44  789   29    9]
 [  77   27  681    4]
 [  39   32   48  374]]
```

Question 4) 35pts – Report

Write a 3-4 lines summary of your work at the end of your notebook; this should be like an abstract of a paper (you aim for clarity and passing on information, not going to details about know facts such as what logistic regression are or what dataset is, assuming they are known to people in your research area).

> "We evaluated the performance of Logistic Regression and Bayes classifiers (Gaussian Naïve Bayes and Gaussian Bayes with general and shared covariance matrices) on the 4 topics of news dataset.
>
> We have obtained the best results with the ….. classifier , giving an accuracy of …% on test data….
>
> You can also comment on the second best algorithm, or which algorithm was fast/slow in a summary fashion; or talk about errors or confusion matrix for your best approach.

Don't forget to discuss, Naive Bayesian and Logistic Regression with the concept of conditional independence and decision boundary.

Note: You will get full points from here as long as you have a good (enough) summary of your work, regardless of your best performance or what you have decided to talk about in the last few lines.

Link to your Colab notebook (obtained via the share link in Colab): https://drive.google.com/file/d/14x15Ts7FRoa3dTKUuBh5Qdez6RykdEsm/view?usp=sharing