Homework 4

For this homework you are no longer given a code skeleton. It is up to you to decide how to structure your code. Please note, you are expected to implement everything in jupyter notebook and submit the notebook as your solution.

Part 1: Naïve Bayes Classification

You are given a training dataset in CSV format (hw4_naive.csv). The training data has 5,600 rows:

- Columns 1 through 6 of the given CSV file represent the features (X)
- The last column ("Label") represents the class label (Y) (0 or 1)

You are required to implement the following models and train/test them using this dataset. **Note that you can use sklearn unless the question asks that you implement your code from scratch.**

1. Divide the data into train / test sets (80% and 20% respectively)

2. (25 points) Implement a Multinomial Naïve Bayes classifier **from scratch**, with smoothing. You can set the default smoothing value to 1. You are free to code this up however you like, however, make sure that there is a function that can be called with a test X vector and returns the predicted Y.
3. (25 points) Implement a Gaussian Naïve Bayes classifier **from scratch** (no need for smoothing here).
4. (10 points) Calculate the accuracy and the F1 score of test data using both of your models implemented above.

Part 2: Clustering

You are given a training dataset in CSV format (hw4_cluster.csv). The files each contain 40 rows with 2 columns. Column 1 & 2 are the features. There are no labels for this dataset. Your goal for this assignment is to implement a clustering algorithm and run it on this dataset. For this assignment you can the Euclidean distance as the distance function.

1) (35 points) Implement a generalized K-means algorithm **from scratch**. You should have a single function that takes in as input the data points, K, and some other hyperparameters, specified below. The function should return K sets of data points. Each set corresponding to one cluster.

The hyperparameters your functions should support and the values they can take are:

- The method for calculating the centroid (i.e. the mean)
- The initialization method: Random Split Initialization or Random Seed Selection Method
- Max_iter: max number of iterations to run the algorithm.
- K: number of clusters

Note that your stopping condition should have two parts:

1. Stop if you reach the max iterations
2. Stop if no change is made to the clusters in the last step.

You will be running this code as part of the next question. For this part you just need to implement the function.

2) (15 points) Silhouette score. In this part of the assignment, you are implementing a function **from scratch** that calculates the Silhouette score for a list of clusters. The function should take in a list of clusters (such as the output of the last function you implemented) and return a single Silhouette score. Report the Silhouette score for {k=5, Initialization method = Random Seed Selection, Max_iter = 50, method for calculating centroid = mean} using your K-Means code from the previous question.

Bonus Points: Correctly answering the following question will improve your lowest assignment grade for this term by up to two points. It's important to note that these additional points can only be applied to a single assignment and cannot be divided among multiple assignments. For instance, if you earn full credit for this bonus problem and your lowest assignment score is 11, we will use the bonus grade to raise that particular assignment score by 1.5 points. However, if you earn full credit for this bonus problem and your lowest assignment score is 10, we will use the bonus grade to raise that particular assignment score by 2 points.

(2pt Bonus Question) Finding best K. Run the code you implemented in question 1 for k=2,3,4,5. Set the other hyperparameters to the following:

- The method for calculating the centroid: Mean
- The initialization method: Random Split Initialization
- Max_iterations: 100

Calculate the Silhouette score for each K using the function in question 2 and use these scores to pick the best K. What is the best value of K?