Assignment 1

Due on October 23, 2024 (23:00:00)
Programming Language: Python 3

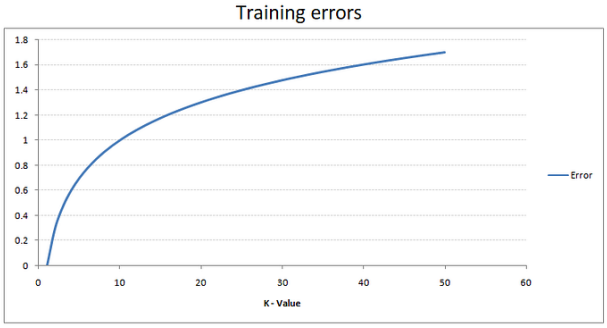
Instructions There are two parts on this assignment. The first part involves a series of theory questions and the second part involves coding.

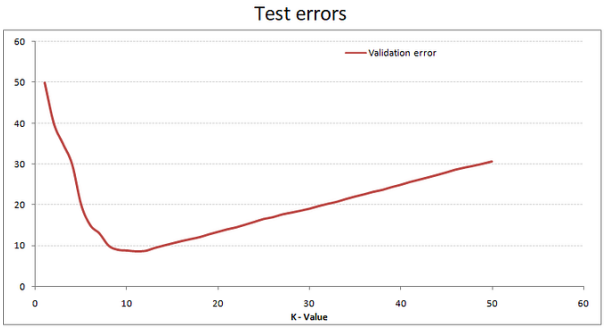
PART I: Theory Questions

k-Nearest Neighbor Classification

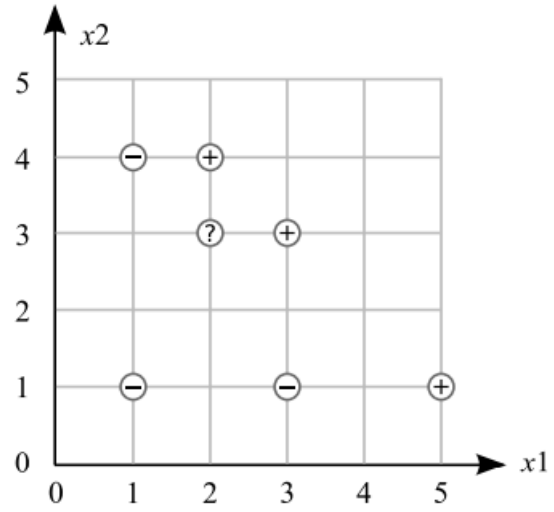
1. Assume that you have a large training dataset. Specify a disadvantage of the k-Nearest Neighbor method when using it during testing. State also your reason about your answer.

2. Considering the image below, state an optimal k-value depending on that the algorithm you are using is k-Nearest Neighbor. State also your reason behind the optimal value you preferred.





3. One of the problems with k-nearest neighbor learning is how to select a value for k. Say you are given the following data set. This is a binary classification task in which the instances are described by two real-valued attributes (+ and - denote positive and negative classes, respectively).



Data points are: Negative: (1, 1) (3, 1) (1, 4) Positive: (2, 4) (3, 3) (5, 1). Data points are classified as either +1 or -1. An unknown point is located at (2, 3)

- Draw the 1-NN decision boundaries on the graph above.
- How would 1-NN classify the unknown point (2, 3).
- What is the minimum value of k for the unknown value be negative?
- Explain the effects of the smaller and larger value of k?

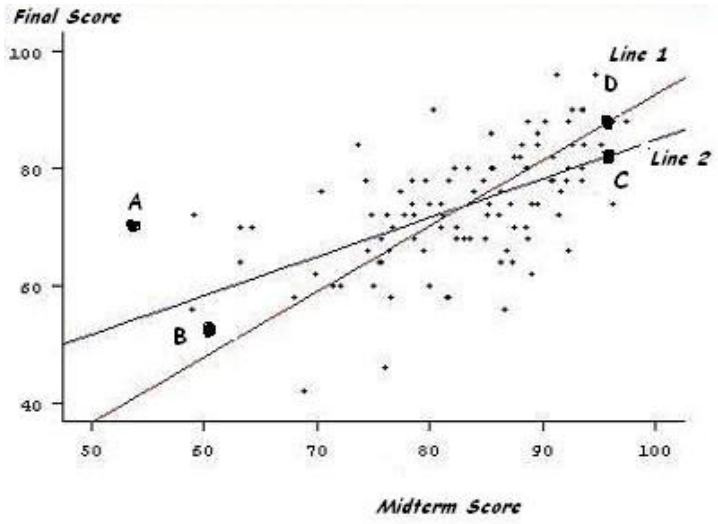
4. Fill the blanks with T (True) or F (False) for the statements below:

- Computational complexity of the training is higher than testing in kNN. (_)
- Scaling the dataset increases the performance of kNN extraordinarily. (_)
- kNN has no idea of the functional form of the problem that it solves. (_)
- There is no difference between Euclidian Distance and Manhattan Distance for kNN while calculating the distance. (_)

Linear Regression

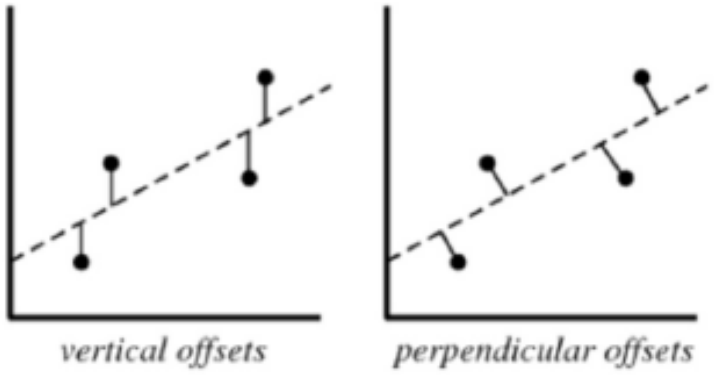
1. Suppose we have trained a linear regression model y = ax+b where a = 1.5 and b = 1.0, on a set of training data points D=(1.0, 1.5),(1.5, 3.25),(3.0, 4.0). Please calculate the mean squared errors of this model on D.

2. Answer the questions according the figure below: These questions pertains to the scatter plot above which shows the midterm and final exam scores of 107 students.



- Which is the regression line?
- Look at students A, B, C and D on the graph. How did their actual scores on the final compare to their predicted scores?
- Without any information about a particular student's midterm score, what would you expect him to score on the final exam? Explain your reasoning clearly.

3. Considering the figure below, which of the offsets used in linear regressions least square line fit? Assume that horizontal axis represents independent variable and vertical axis represents dependent variable. State your answer with your proper explanation.



4. Considering the table below, consisting of four training examples:

x	y
3	9
1.5	6
2	7
-2	-1

   Assume that you are trying to fit the data above to the linear regression model $f_\theta(x) = \theta_0 + \theta_1 x_1$. Find the θ_0 and θ_1 values by using closed form solution ($\theta = \left(X^T X\right)^{-1} X^T y$). Also state dimension values of X, y, and θ matrices. Finally, show your calculations step by step.

5. Explain the importance of the feature scaling in logistic regression.

PART II: Telecommunication Customer Classification System

Let's say a telecommunication company has divided its customers into four groups based on how they most typically utilize its services. If group participation can be predicted using demographic data, the company can customize offers to certain potential clients. It is a matter concerning classification. In other words, it must be created a model to be used in predicting the class of a new or unknown case given the dataset and established labels. In this part, you are supposed to implement a nearest-neighbor algorithm to determine classes of the new clients with help of the existing clients. You are also supposed to extend your implementation to a weighted nearest-neighbor algorithm.

Dataset

The Customers of Telecommunication Company is provided for you as a playground. The dataset contains 1,000 customers and four categories. The categories are "Fundamental Service", "E-Service", "Advanced Service", "Complete Service" which are given in service field. Dataset also contains 11 data fields about the customers they are: "district", "customer_since", "age", "is_married", "address", "salary", "ed", "employment_status", "is_retired", "gender", and "reside".

Steps to follow

1. Import and visualize the data in any aspects that you think it is beneficial for the reader's better understanding of the data.

2. Split data into train and test set randomly (you can use 80% of the data for training and 20% of it for the test purposes).

3. For the test set that you separated at the previous step try to determine classes for the customers.

4. Finally compute performance of your model to measure the success of your KNN Classification method for each setting you have used:

$$\textbf{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN}$$
$$\textbf{Precision} = \frac{TP}{TP+FP}$$
$$\textbf{Recall} = \frac{TP}{TP+FN}$$

   You will report Accuracy, Precision, and Recall measures.

5. The most important part of this project is doing as much experiment as you can to show strengths and weaknesses of the kNN algorithm. In short, you are supposed to experiment with different scenarios and comment about them, note that commenting is as much important as the experimenting, so, please explain your reasoning and inference for every experiment that you did. Some examples that you may try are (Note that the following ones are only examples, you can add anything that you think it is beneficial to try for better understanding about kNN.):

   - You can use different k values and try to determine optimal k value for this dataset according to your implementation. You may benefit from validation error for this purpose.
   - You can compare the performance of your model with raw data and eliminated data.
   - You can compare the performance of your model with raw data and scaled data.
   - You can compare the performance of the kNN and weighted kNN algorithm. You can use three weighted kNN approaches:
     - Weights inversely proportional to the distances.
     - Weights that are inversely proportional to the occurence of the class during the train set.
     - Combination of the both of the approach.

What to Hand In

You are required to submit all your code in a Jupyter notebook, along with a report in ipynb format, which should also be prepared using Jupyter notebook. The code you submit should be thoroughly commented. Your report should be self-contained and include a concise overview of the problem and the details of your implemented solution. Note that your report also has to contain necessary libraries to be installed (!pip install commands are preffered). Feel free to include pseudocode or figures to highlight or clarify specific aspects of your solution.

Submission hierarchy must be as follows:

```
- b<StudentID>.zip
    - assignment1.ipynb
```

Do not send the dataset.
P.S.: Please divide your Jupyter Notebook into two main parts for the two parts of this project, so, in short, you are supposed to give your answers to the theoretical questions in the first part at your Jupyter Notebook too.
Note that submission format is crucial and submit system is set to give you score as one if you follow the submission hierarchy, which is really easy (there might be some issues for the MacOS users but it can be overcomed via the mini guide that is shared along with this assignment itself at the Ed Platform). If you do not score one from the submit system **you will penalized by 20% even if your submission hierarchy is correct**.

Grading

- Part I : 20 Points
- Part II: 80 Points

P.S.: You can use libraries for visualization and explanation, but you must implement kNN and things related to its mechanics from scratch, which means any library usage is forbidden except for the visualization and explanation purposes. Note that you can use train_test_split from scikit-learn, it is an exception.
Note: Preparing a good report is important as well as the correctness of your solutions! You should explain your choices and their effects on the results. You can create a table (or any content you believe that it is beneficial to show your all work) to report your results.

Late Policy

You may use up to five extension days (in total) over the course of the semester for the three problem sets you will take. Any additional unapproved late submissions will not be evaluated.

Academic Integrity

All work on assignments must be done individually unless stated otherwise. You are encouraged to discuss with your classmates about the given assignments, but these discussions should be carried out in an abstract way. That is, discussions related to a particular solution to a specific problem (either in actual code or in the pseudocode) will not be tolerated. In short, turning in someone else's work, in whole or in part, as your own will be considered as a violation of academic integrity. Please note that the former condition also holds for the material found on the web as everything on the web has been written by someone else.