CS5644 Assignment 3

Q1. Data Analysis on Congressional Voting Records dataset

a. Introduction

This report documented the machine learning analysis of taking the Bike Share daily/hourly dataset and predicting the daily/hourly values for the number of different kinds of riders. 3 models for linear regression and 3 models for KNN regression were used and compared, and the performance results and the best model were reported. The voting Records dataset is from the UCI machine learning repository (https://archive.ics.uci.edu/ml/datasets/Bike+Sharing+Dataset).

b. Methods

1) Data preprocessing

Both hourly data and daily data are structured fairly well for applying a machine learning model. The only change is to change the date attribute to something a machine-learning model can recognize. All dates are changed to ordinal numbers so they can be used in calculation later. However, after comparing the R^2 score of using the date or the components such as year and month, I found out that if I don't use the date, features such as year, month and season can extract of more meaningful information. For this dataset, whether it is the working day is important, and I think the best indicator compared to the holiday and weekday since it has a better score and those features is kind of repeat with it.

Features didn't used (and why):

1. Dteday: repeat with Yr and mnth, also decrease the R^2 score a lot for KNN in daily data, using components like season, year, and month instead of the actual date is often more beneficial when building predictive models
2. Holiday: kind of repeat with workingday
3. Weekday: kind of repeat with workingday
4. casual, registered, and total: defeat the purpose of the learning.
5. Instant: The instant here is just the index number, usually is not relevant to the prediction

Features used (and why):

1. Season: impact weather
2. Yr: maybe there has a more popular year.
3. Mnth: maybe there has a more popular month.
4. Workingday: Whether every working person has time should be important.
5. Weathersit: the weather should be important.
6. Atemp: the feeling of the temp should be important.

7. Temp: the temp should be important.
8. Hum: the humanity should be important
9. Windspeed: the windspeed should be important.

2) Machine learning models

Linear regression and KNN regression were used to predict hourly and daily values for different kinds of riders. For KNN, 5 neighbors with uniform weights were used. Each result includes mean squared error and R^2 score.

c. Results

1) Hour

i) Predict the hourly values for the number of casual riders

Linear Regression

Mean Squared Error	R^2
1288.56	0.47

KNN

Mean Squared Error	R^2
429.77	0.82

ii) Predict the hourly values for the number of registered riders

Linear Regression

Mean Squared Error	R^2
15665.86	0.32

KNN

Mean Squared Error	R^2
2020.82	0.91

iii) Predict the hourly values for the number of total ridership count

Linear Regression

Mean Squared Error	R^2
20046.94	0.38

KNN

Mean Squared Error	R^2
2954.39	0.91

2) Day

i) Predict the daily values for the number of casual riders

Linear Regression

Mean Squared Error	R^2
167088.00	0.67

KNN

Mean Squared Error	R^2
119638.16	0.72

ii) Predict the daily values for the number of registered riders

Linear Regression

Mean Squared Error	R^2
512593.07	0.79

KNN

Mean Squared Error	R^2
413331.45	0.83

iii) Predict the daily values for the number of total ridership count

Linear Regression

Mean Squared Error	R^2
847010.01	0.79

KNN

Mean Squared Error	R^2
623912.31	0.84

d. Discussion and Conclusion

The KNN model generally outperformed the Linear Regression model, especially in an hourly dataset. KNN can capture non-linear relationships instead of assuming a linear relationship like Linear Regression. In the hourly dataset, it has many more data points, a KNN can adapt to the local variability in the data better than Linear Regression. It allows KNN to have a richer set of neighbors since KNN makes assumptions about the form from the dataset. On the other hand, in a daily dataset, the data is "smoother" and has less noise. The trend is much more straightforward for having a linear relationship. The patterns are more general and manageable for a parametric method like Linear Regression.

Q2. Clustering

e. Introduction

This report documented the machine learning analysis of the Seeds data set and try to cluster it into three clusters based on the seven features. A PCA was also conducted to generate a visualized result to compare and visually inspect with the ground truth data.

The voting Records dataset is from the UCI machine learning repository (https://archive.ics.uci.edu/ml/datasets/Congressional+Voting+Records).

f. Methods

3) Data preprocessing

The data was stored in a txt file, but it was not formatted very well. Sometimes it has two "\t" in a row. After preprocessing the data and applying the column names, a typical pandas data frame was built and good to go.

4) Machine learning models

A Kmeans model was used to cluster the dataset. The “class” feature was took out before the training for making the comparison with the result. There are 3 classes in the data, so it was a 3 clusters model. After the training, the results were compared with the ground truth, and a PCA was conducted to get a better understanding of the performance.

g. Results

Cluster (result of k-means)

Group A Count	Group B Count	Group C Count
77	61	72

Ground Truth

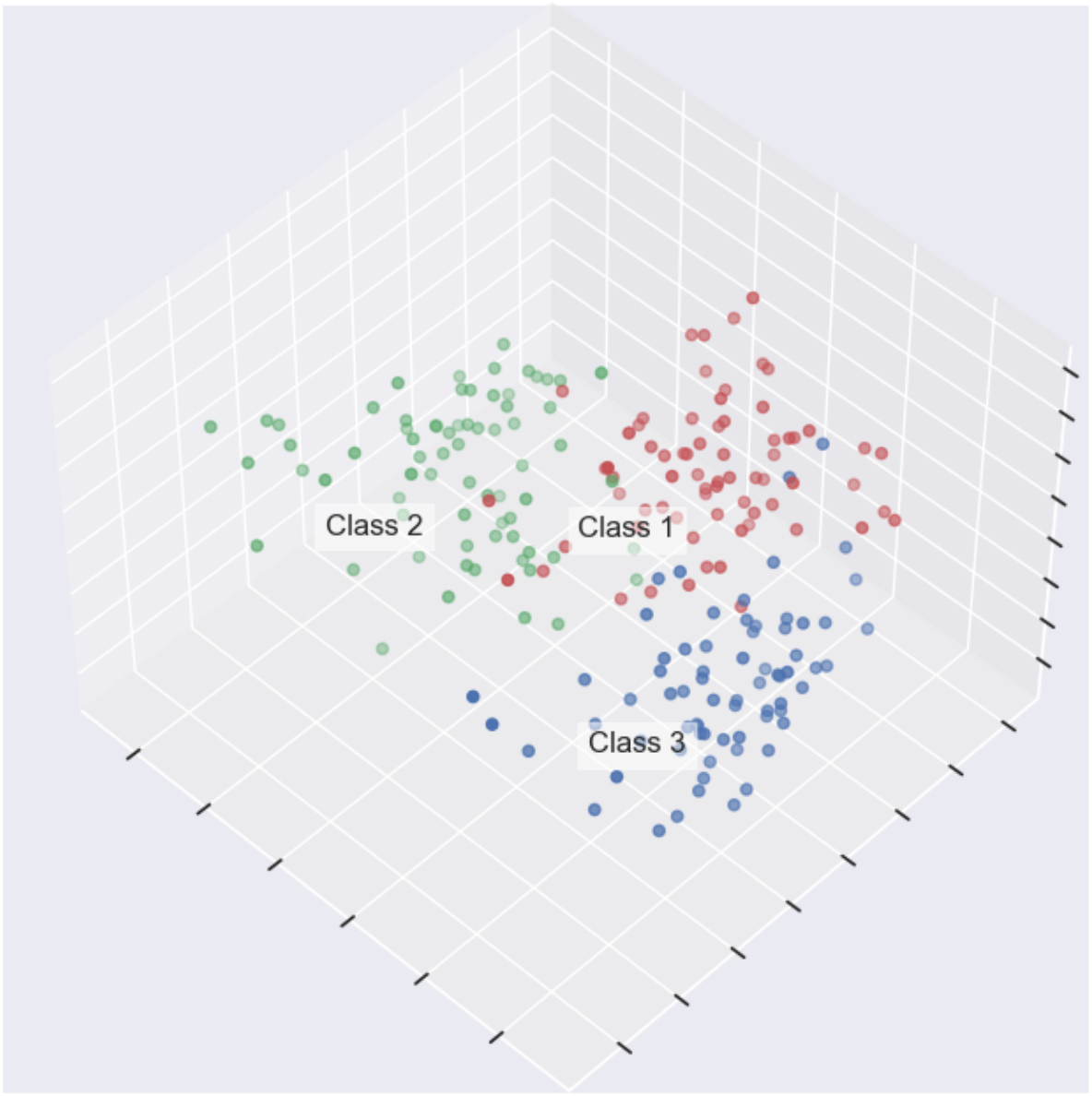
Class 1 Count	Class 2 Count	Class 3 Count
70	70	70

Predict by every 10^{th} element:
[0 0 0 0 0 0 2 1 1 1 0 1 1 1 2 2 2 2 2 2 2]
Truth by taking every 10^{th} element from target (Class -1, since index is starting from 0):
[0 0 0 0 0 0 0 1 1 1 1 1 1 1 2 2 2 2 2 2 2]

This result indicates that we don’t have to reorder the labels to match the cluster results
The “0” in cluster -> “0” in target (Class 1)
The “1” in cluster -> “1” in target (Class 1)
The “2” in cluster -> “2” in target (Class 1)
The distribution is also making sense with the total count.

PCA demonstrated and plotting.

A PCA was performed with 3 components with the same dataset, a 3D plot for demonstrating the PCA transformation was created. The labels were assigned based on the comparation of the visually inspect action with the ground truth data. The order of the label is most likely the same as the target’s based on the comparation before, so I am confident enough to put the Class label on the plot.



h. Discussion and Conclusion

There are mismatches between predicted clusters and the ground truth, but it is generally good to see the relations between the clusters and the actual labels. The total count is also mismatched, with some minor differences. From the PCA visualization result, it indicated the clusters are generally distinct. We can observe the “area” of the cluster easily in this 3-dimensional space. Since KMeans assumes that clusters are of a similar size, in this case, the total count is all the same, so that’s why it works pretty well. Another reason is we already know there are 3 kinds of data from the dataset. Knowing the number of clusters beforehand helps a lot since it guides the algorithm.