Johann Ruiz

CS5644

Homework #3

Q1:

Introduction:

For homework 3, we are looking at PM2.5 data in Beijing, China. PM2.5 are microscopic particles in the air that have a diameter of 2.5 micrometers or less. Particle matter at this size can have an adverse effect on human health; breathing in these fine particles can cause health problems like heart disease, asthma, and low birth weight. PM2.5 is a good indicator for pollution in an area. Within the PM2.5 dataset, there is meteorological data, like temperature, windspeed, humidity, etc., and when the data was recorded. Our goal is to use regression models to predict PM2.5 levels using meteorological data.

Experiment:

Data Processing:

To process the data, I had to unarchive the rar file provided by archive.ics.uci.edu. From here, the dataset had csv files for 5 Chinese cities. I loaded the Beijing dataset using pandas read_csv function. Within the dataset, it has PM2.5 values for 4 different sites in Beijing: Dongsi, Dongsihuan, Nongzhanguan, and the US Post. From here, I did feature engineering to remove the time-dependent columns (year, month, day, hour). I didn't want the data to add unnecessary noise to the model when the data was captured. For example, I didn't want a correlation to potentially occur if pollution levels were higher on the 7^{th} of a month. However, there can be a correlation that pollution levels rise or lower in a certain time of year, therefore I left the season column in the dataset. Then, I dropped all rows that

N/A values. Next, I separated each of the site's PM2.5 values into 4 DataFrame since we don't want the PM2.5 levels at one site as training data for predicting another site. This wouldn't be useful because the goal is to use other technology to predict meteorological in the future and use these predicted values to then predict the PM2.5 values. Lastly, one of the columns was wind direction, which was presented by categorical values of 'cv', 'SE', 'NE', 'NW'. We want to One Hot Encode these values, which converts one column into four columns, and each of those values represents each categorial value. Now, one of the four columns is going to have a '1' value and the others have '0', which is how to encode a categorial value to a number.

Models:

To predict the PM2.5 levels, we are going to use two different regressions models. The first model is linear regression, which tries to find a linear relationship between the features and the target value. The essential goal of a linear regression is to create a linear equation that is best fit all the data points in the dataset. The second model is k nearest neighbor, which makes predictions by relating it to other data points that are closest to it. The k determines how many values relate to it.

Results:

Below are the results from a 5-Fold cross validation using both the Linear Regression and kNN regression models. Below, I used k=117.

Model	RMSE (Mean)	R2 (Mean)
LR-- Dongsi	72.39	0.203
LR-- Dongsihuan	74.88	0.226
LR-- Nongzhanguan	71.84	0.208
LR-- US Post	73.70	0.213
kNN - Dongsi	71.92	0.215
kNN-- Dongsihuan	71.74	0.289
kNN-- Nongzhanguan	69.77	0.254
kNN -- US Post	71.93	0.251

For selecting a k for the kNN regression model, I looked up what ways are used to select k. In the sklearn library, there GridSearchCV class that will conduct a cross validation with a range of k values and reports with k value had the best RMSE value. Therefore, I used this class to determine my k. When I was first trying to select a k, I used 5, and noticed the R-squared value was extremely low, values being around 0.01. Therefore, I did a range between 0-19 and ran the GridSearchCV and it returned to a k of 19. I noticed that this k value was the highest possible k, therefore I increased my range from 0-99, which returned 99. From here, I increased it to 0-199, and that's when I got a value of 117, which made me confident I was selecting the best possible k.

The two metrics used to determine the performance of the regression models are Root Mean Square Error (RMSE) and R-squared. RMSE takes the difference between the predicted and actual and takes the root mean squared to normalize the value. Therefore, a lower RMSE value is an indicator that the model is accurately predicting the target value. The R-squared is a value used to determine how much weight the values we are using the train the data, like temperature, humidity, wind direction, play into determining the PM2.5 value. For example, a dataset that had temperature and the target value was pressure, the R-squared value would be 1 because they are mathematically directly proportional.

The kNN regression model showed better performance with lower RMSE and higher R-squared values compared to the linear regression model. This can be an indicator that the dataset does not have linear relationship of PM2.5 values to temperature, humidity, wind speed, etc. More likely, there is a more complicated relationship between the values than a best fit line can show. However, it's important to note that both models performed relatively poorly, whereas the R-squared values only increased around 0.05. This can be reflective that more feature engineering needs to take place. Likely, some columns left in the dataset don't have a relationship to PM2.5 level and can be removed before applying to the model. In addition, adding more features to the dataset could also help with data that has non-linear relationships. Therefore, it's important

when doing machine learning to have domain knowledge in the dataset you are training on. In addition

Q2: View Zipped File