KING FAHD UNIVERSITY OF PETROLEUM &

MINERALS SYSTEM ENGINEERING DEPARTMENT

Math405: Learning from Data

KNN Regression Forecasting

Mohammed Zaki Al Saleem 201766890

Introduction:

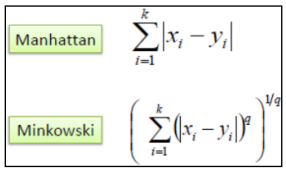
The problem is to forecast the travel times between two places (points) which from point (0) Burrard st. to point (1) Highburry st. in the year 2020. The measurements time of the 2017, 2018, 2019 years are given.
The data we have are day order, the season in the year (1 to 3), the day of the week (1 to 6), the period of the day (1 to 4), and the travel time, but there are some measurement errors we need to deal with.

Literature review on KNN Regression:

KNN regression is a method that depends on the neighbor observations with independent variables to forecast future values. KNN regression considers as an intuitive manner, not a parametric method.
The number of neighbors which will be calculated represented by 'k', calculating the distance between neighbors can be done in many ways, the famous one is Minkowski distance. When the power parameter (p) equal to 2 it called "Euclidean Distance", and when it equal to 1 it called "Manhattan Distance". The distance has an inverse relationship with the weight of neighbors.

Mathematical Methodology for KNN Regression:

The first step is calculating the distance between our point (the day and the period with all features) and nearest k neighbors, in this project Manhattan distance is used (Minkowski distance with p = 1) which is the distance between the real vectors of features calculated by the sum of their absolute difference.



After calculating the distances with the nearest k neighbors, we calculate the weighted average of their corresponding times and assign it as the time of our point.

x: is the features
f: is the time
w: weight
d: distance

$$\hat{f}(x_q) \leftarrow \frac{\Sigma_{i=1}^{k} w_i f(x_i)}{\Sigma_{i=1}^{k} w_i}$$

$$\text{where} \quad w_i \equiv \frac{1}{d(x_q, x_i)^2}$$

Work Methodology:

- First step was arranging the Data in excel. There was no information about how the effect strength of each year in the forecast, so assume all three years have the same effect, then we can put the three years together as one array.
- The features of the year 2020 are created by excel, (some simple functions and micro are used to create it fast) then the rows with DayOfWeek equal seven (Sunday) are deleted, then January 1st and December 25th also deleted. The file is named “2020x.csv”.
- Source data and 2020 features are imported to Python as ‘df’ and ‘x20’ respectively. To deal with measurement errors, in source data the rows with a time equal to zero are deleted; we have enough measurements so delete errors is the best and easier way.
- Source data separated to ‘x’ data frame with the features, and ‘t’ data frame with the time.
- The forecasting done by K Neighbors Regression (KNN Regression), KNeighborsRegressor fitted to ‘x’ and ‘t’ to get model for forecasting, and the parameters (n_neighbors = 2, p = 1, algorithm = 'brute', weights = 'distance') sited by try and error and logically until getting the best forecasts.
- The model with the year 2020 features as input is used to predict the times in the year 2020. The features merged with the times and exported as “year2020_Forecast.csv”.

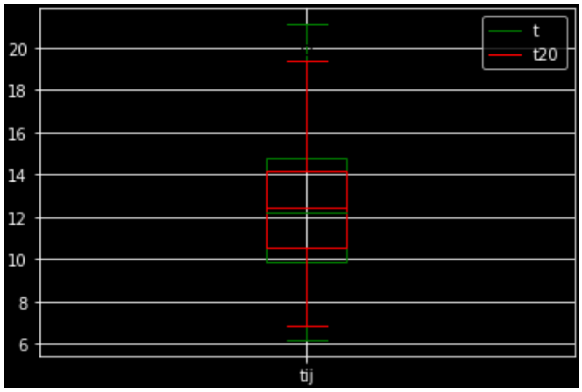
Parameters calibration table:

Parameters				Results (measured stats - forecasted stats)				Notes
n_neighbors	p	algorithm	weights	Mean	Max	Min	STD	
2	1	brute	distance	-0.0867	1.2243	-0.6947	0.688	The best results.
3	1	brute	distance	-0.1099	2.0087	-1.5278	0.9907	
4	1	brute	distance	-0.1596	2.6436	-1.7216	1.1907	
1	1	brute	distance	-0.1198	0.1408	-0.2968	-0.0111	The change in results is too smooth.
1	2	brute	distance	-0.1812	0.1408	-0.767	-0.0256	The change in results is too smooth.
2	2	brute	distance	-0.0986	1.2243	-1.0452	0.7697	
2	3	brute	distance	-0.0894	1.2243	-1.0452	0.7659	The change in results is too smooth.
2	1	brute	uniform	-0.0461	1.2243	-0.7444	0.7544	The uniform weights let the change in results too smooth.
2	3	brute	uniform	-0.0486	1.2243	-1.1404	0.824	
2	1	auto	uniform	-0.1646	1.2243	-1.1404	0.7678	
2	1	ball_tree	distance	-0.1926	1.2243	-1.0452	0.6994	
2	1	kd_tree	distance	-0.1844	1.2243	-1.0452	0.7031	
2	1	auto	distance	-0.1844	1.2243	-1.0452	0.7031	Auto should choose the best algorithm, but here it chooses kd_tree, it is good but not the best.

Discussion and Graphics:

By comparing some stats of given time (2017,2018,2019) and forecasted time (2020), we can see the differences are small, which indicates the forecast is good.

	Years 2017 to 2019	Year 2020	Difference
Mean	12.378	12.465	-0.087
Max	21.119	19.894	1.224
Min	6.176	6.870	-0.695
STD	3.149	2.461	0.688



The box plot compares some stats (median, Q1, Q3, min, max), the two boxes are similar, but the forecast has a smaller range.

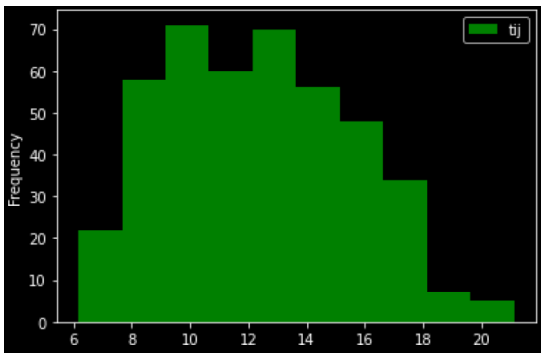
Furthermore, by looking at the results of the forecast we can see it is reasonable and like the previous years.

Part of the results:

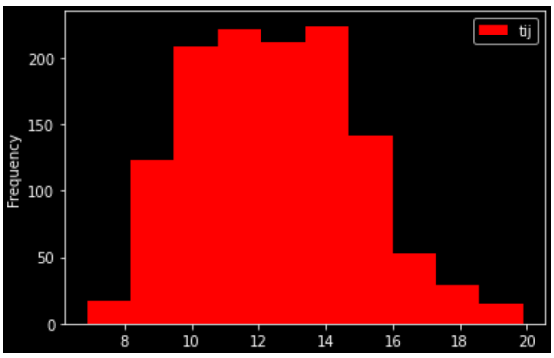
time for year 2020:

	DayOrder	Season	DayOfWeek	Period	tij
0	2	1	4	1	15.753333
1	2	1	4	2	18.375527
2	2	1	4	3	16.856300
3	2	1	4	4	14.901335
4	3	1	5	1	15.753333
5	3	1	5	2	18.375527
6	3	1	5	3	16.856300
7	3	1	5	4	14.901335
8	4	1	6	1	15.753333
9	4	1	6	2	18.375527
10	4	1	6	3	16.856300
11	4	1	6	4	14.901335
12	6	1	1	1	14.299341
13	6	1	1	2	14.386296
14	6	1	1	3	13.429794

The two histograms show the frequency of time, and they have a similar pattern.



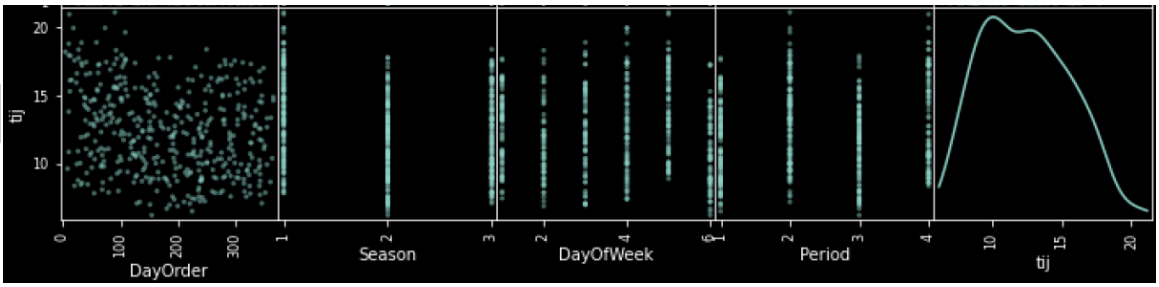
The measured time.



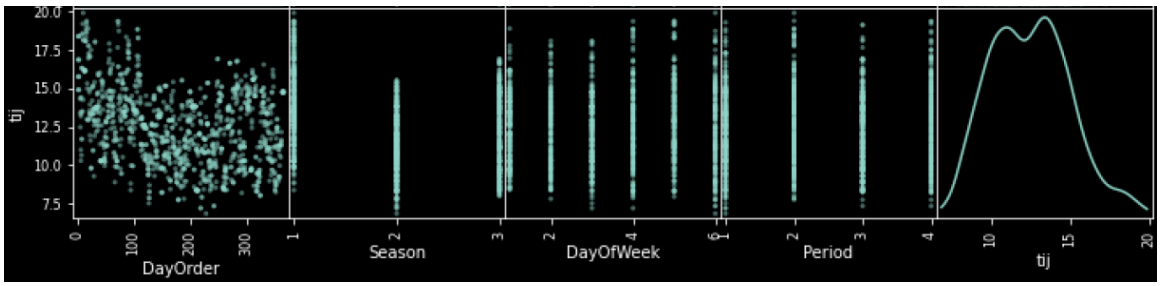
The forecasted time.

The scatter matrix shows the relationship between the data in the one matrix, here are the last rows of the scatter matrix for measured data and for forecasted data:

The measured data.



The forecasted data.



For example, we can see the time at the beginning of the year is a little higher in both measured and forecasted times. Also, on Friday (Day of week = 5) The time is clearly higher in the measured time, in the forecast it is also higher but not clear.

Conclusion:

In KNN Regression we must set the parameters and the algorithm to do a good forecast. In this project, it was concluded that the two nearest neighbors with Manhattan distance give the best results. But even the good forecast will have some differences with the real measurements, like having a smaller range. Finally, the scatter matrix shows how the forecast is very similar to measurements with small differences which is acceptable.

research references:

Coursera course: Applied machine learning in Python by university of Michigan.
Coursera course: Applied Plotting, Charting & Data Representation in Python by university of Michigan.
https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.KNeighborsRegressor.html
https://stackoverflow.com/
https://medium.com/analytics-vidhya/knn-k-nearest-neighbors-1add1b5d6eb2