Buğra Sipahioğlu

December 10, 2019

Mehmet Gönen, ENGR 421

Homework 06: Modeling Cash Withdrawals from ATMs

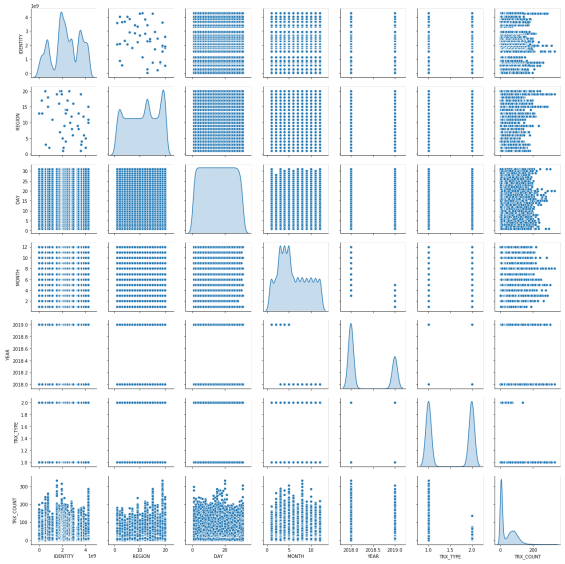
I've chosen to use Random Forest Regression Tree for this assignment. I am going to explain my implementation part by part:

Import Libraries & Data

I used sklearn's regression library to handle random forest. In order to plot the data, I used seaborn library since it allows users to see joint distributions of different columns in the dataset. In order to import data, I used pandas library since it is highly useful.

Analyze Data

First of all, I looked at the first rows of the data to get an understanding. Then, I've investigated the data to look for any unknown values. If such values were existed, I would have either notify the instructor or remove those rows. After these, I looked at the joint distributions of the data:



When I've noticed the correlations between features are low, I decided to use a decision tree. Since the raw implementation gave poor results, I decided to use random forest tree since it gives higher accuracy and probability of overfitting is decreasing with the number of trees used. In addition, the dataset is large enough to use overfitting since training set has over fourty-thousand rows. Last of all, I've analyzed the statistics of data. If the data intervals of columns were too diverse, I would have normalized the data. This weren't the case :

	count	mean	std	min	25%	50%	75%	max
IDENTITY	42958.0	2.352173e+09	1.193318e+09	17419935.0	1.719547e+09	2.290604e+09	3.429897e+09	4.264240e+09
REGION	42958.0	1.119149e+01	6.044721e+00	1.0	6.000000e+00	1.200000e+01	1.700000e+01	2.000000e+01
DAY	42958.0	1.574398e+01	8.808176e+00	1.0	8.000000e+00	1.600000e+01	2.300000e+01	3.100000e+01
MONTH	42958.0	6.017505e+00	3.264414e+00	1.0	3.000000e+00	5.000000e+00	9.000000e+00	1.200000e+01
YEAR	42958.0	2.018330e+03	4.703683e-01	2018.0	2.018000e+03	2.018000e+03	2.019000e+03	2.019000e+03
TRX_TYPE	42958.0	1.500000e+00	5.000058e-01	1.0	1.000000e+00	1.500000e+00	2.000000e+00	2.000000e+00

Random Forest Regression & Hyper Parameter Tuning

I used the libraries I've mentioned to fit and predict with random forest regressor. After that, I've calculated root mean square error and mean absolute error with sklearn functions.

I conducted a grid search. Since it takes too much time, in the first part (random forest regression) I used the parameters that I've extracted from the tuning process. In the tuning process, I conducted a grid search. Here are my best RMSE and MAE values, which are better than the quick tree implementation of instructor:

```
Root Mean Squared Error:  6.769485461617503
Mean Absolute Error:  3.930029971134597
```

And lastly, here are my parameters for this result:

n_estimators=400, max_depth = 50, min_samples_split=2, min_samples_leaf=1, random_state=0