# Pre-processing:

The pre-processing consisted of removing columns which had more than 40% null entries. The rest of the columns were all related to basement and parking. The null entries in these columns could indicate that there is no basement / parking. Therefore I decided to introduce a new class called “No Basement” and “NoParking”. This is a binary class, if this was zero then there was a basement/ parking and if this was 1 there was no basement / parking on the premises.

The only columns left with null values were RoadAccessLength, FaçadeArea and ElectricalSystems. For RoadAccessLength I removed the outliers using the outlier removal given in the tutorial github repo. The null values in RoadAccessLength were considered to be zero. Similar outlier removal was done to the FacadeArea. ElectricalSystem had only one null entry, since most of the hotels had the same electrical system I consider it to not be of much use in our prediction and I replaced the null entry with the most frequent entry.

I applied PCA, but I found it to not be useful for this dataset as the feature reduction was not significant.

# Models:

I have used two models, one is a decision tree with a ridge regression model at each leaf and the other is a meta model which uses the output of two models to determine the final output. The first model gave a relatively large RMSE(Root Mean Square Error) of 80k, this could be due to the fact that I had no control of which features were used in the decision tree and which were used in the linear model. Since the decision tree had divided the data, each ridge regression model had less data to work with.

The hyper parameters were tuned for the decision tree and we realised that the depth parameter had negligible as we increased it, but it gave worse results if we decreased it from the optimum, the alpha for the ridge model also behaved in a similar way.

The initial idea was to capture the non-linear relations with the decision tree and the linear model would capture the linear relation. This gave rise to another model, where both the gradient boost and the linear model would get the same data and the output of both would be the input of another linear model. This gave much better results, the RMSE was 27k, this was a significant reduction from 80k.