

# Homework 3

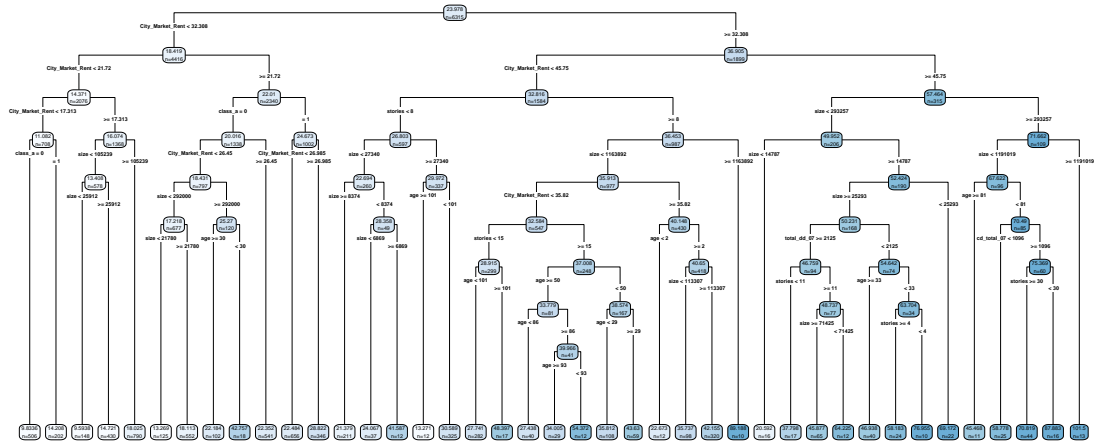
Patrick Massey

4/4/2022

Before developing any models we first begin by performing some feature engineering. The first feature we engineer is the outcome variable of interest revenue which represents the revenue per square foot per calendar year. In order to create this feature we first scale down `leasing_rate` to a percentage by dividing by 100, and then multiplying that by the rent. We also create a new feature called `utility_cost` which is the sum of gas and electricity costs for rents that are quoted on a net contract basis. The purpose of this new feature is to capture the costs associated with a rental offered on a net contract basis. We then create a training set and a testing set with a split of 80/20. This gives us 6315 observations in our training set and 1579 observations in our testing set.

To begin developing our model we start with a linear model using all features of the data set excluding, `CS_PropertyID`, `cluster`, `leasing_rate`, `Rent`, `LEED`, and `Energystar`. We remove `CS_PropertyID` as it is just a unique building ID, and for similar reasons we remove `cluster`. We remove `leasing_rate` and `Rent` since these variables directly calculate our outcome variable. Lastly we remove `LEED` and `Energystar` because we are only concerned if a building is green certified or not, and not what kind of green certification a building may have. We capture this with the `green_rating` feature.

After getting a baseline model we then moved onto predicting using a tree model. The initial tree model generated was extremely complex and not readable. This indicated that there might be some overfitting happening. We then pruned our tree which generated the much simpler decision tree shown below.



The visualization of the tree really highlighted the lack of interactions that were not included in our baseline linear model. Naturally after seeing the performance of the tree as compared to the linear model we wanted to see if it could be improved upon using a random forest.

### gb\_forest

