**CO3093 Big Data and Predictive Analytics**

**Coursework 1 – Model building and Evaluation**

**Initial Steps**

Firstly, I imported the dataset, then I proceeded to do the following:

* Rename incorrectly formatted column names (e.g. SALE\nPRICE) in the CSV file
* Create a list of numerical and categorical columns
* Removed all the commas, $ and spaces in the numerical columns and converted them to numeric.
* Changed suspicious zeros (Land Square Feet, Gross Square Feet, Sale price, year built and Tax class at time of Sale) to NaNs.
* Removed all NaNs.
* Removed duplicates
* Dropped borough, ease-ment and apartment number because they all the houses are from the same borough and easement and apartment number have over 75% of null values.

This then gave me a much smaller dataset (2234,18), from which I was able to study the correlations using a heatmap.

A screenshot of a cell phone

Description automatically generated

**Figure 1.0**

This dataset allowed me to gain a deeper insight into the relationships between variables and allowed me to build a linear model using the five most correlating columns.

A screenshot of a cell phone

Description automatically generated

The R squared score for this model was low, so I decided to go back and explore the data further in order to improve the R Squared score.

**Exploring the data**

I started my exploratory analysis of the data by doing some visualisations. I started off by visualising neighbourhood against sale price.

A close up of a logo

Description automatically generated

Figure 1.2

From figure 1.2, we can see that some neighbourhoods such as ‘Little Italy’ and ‘SOHO’ have exceptionally higher prices than others. This is something I will need to consider when building my model.

I then went on to visualise price by the year they were built possibly to spot potential outliers and get a better understanding of interesting trends in my data set.

A screenshot of a cell phone

Description automatically generated

According to the visualisation, the highest recorded sale between 1900 and 1950 was $1200000000. This piqued my interest because it was a very clear outlier, so I decided to explore it further.

(data.sort\_values('SALE\_PRICE').tail(1)) - using the aforementioned code, I found out that this property is a commercial vacant Land which explains the huge sale price.

**Summary statics**

A screenshot of a cell phone

Description automatically generated

Figure 1.3

To better explore the mean, maximum and minimum prices, I returned the summary statistics on the sale price column and applied a lambda function to show the statistics in simple numbers as opposed to scientific notation. I found that the min of house sales was priced at $1. This therefore led me to assume that some of the data were either recorded incorrectly or cases where the property was transferred from one party to another through inheritance. Following this assumption, I concluded that these entries where not relevant to my analysis and proceeded to filter them by changing all the zeros in the price column to null.

I also changed the zeros of the following columns to null, as they were the most highly correlation columns and there shouldn’t be any zeros in them:

* Land Square Feet
* Gross Square Feet
* Total units

In addition, I replaced empty strings by nan to try and filter out areas where data was missing. Instead of having two separate columns for the date of sale and year in which the house was built, I created a column that would show the age of the house at the time of sale. I believe this column contains the information that the year of sale and year-built column collectively show.

**Cleaning up NA values**

After changing the zeros in the aforementioned columns to null, I then went ahead to visualise all the null data in the data set in order to see which columns contains the most nulls.

A close up of a logo

Description automatically generated

Figure 1.4

As you can see from figure 1.4 Land and Gross Square feet had over 75% of null values, which is outstanding compare to the other fields. Because of this, I decided to drop all the null values in these columns as trying to replace them with the mean seemed to skew my data as shown in figure 1.5.

A screenshot of a cell phone

Description automatically generated

Figure 1.5

Furthermore, I dropped the null values in total unit, then I set a rule that only kept the rows where total unit equals residential units + commercial units. The new column created by me (AGE\_OF\_HOUSE\_AT\_SALE) also had some null values, I simply filled these values up with the mode of my column. For the other non-numerical columns that had null values, I simply did a backward fill of these columns.

**Removing Outliers**

For my analysis I decided to remove outlier sales. Since I want to predict the price of houses using regression models I believed that it would be harder to get a model that performs well for both normal and outlier pattern sales, the latter of which may include multiple commercial properties (for example the commercial vacant Land sold for $1200000000). I understand that doing this renders my models incapable of generalising to outlier house prices and may 'artificially' improve the performance of my regression models. In order to remove these outliers, I used a box plot to visualise caps. Then I removed the observations that fall out side those caps.

A screenshot of a social media post

Description automatically generated

Figure 1.7

After removing outliers from sale price, I then plot sale price using seaborn and I found that It was skewed to the left, so therefore I applied log of prices and removed the outliers from that to make it more symmetrical, thus giving better results. I also visualised and removed outliers from the following columns:

* Age of building at sale
* Total units
* Land Square Feet

I did this because when I visualised the dataset, there were some clear outliers which could have potentially skewed the predictions of my model.

**One hot encoding and Normalisation**

The next step I took was to then one hot encodes the categorical variables that I wanted to use in my feature columns, this significantly increased the number of columns in my data. I did this to cater for the categorical variables in the dataset, as the linear regression model only accepts numerical data. After this I then normalised the dataset, the goal of this is to reduce redundancy, inaccuracy and to organise the data.

**Final Model**

To build this model, I picked the 30 most correlating columns, to ensure that all the categories and numerical values were represented equally.

**A screenshot of a social media post

Description automatically generated**

**Figure 1.8-R Squared Scores**

As you can see from figure 1.8, my r squared score for my second model is much higher compared to that of the first model that I initially built, this therefore suggests that my model is able to explain a good amount of the variation in the response variable around its mean. However, R-squared does not indicate if a regression model provides an adequate fit to your data. A good model can have a low R2 value. On the other hand, a biased model can have a high R2 value. Also, my final model has a mean squared error of 0.01097136600488941 which is quite small, this therefore suggesting that my model is good at prediction.

**Mathematical equation of fitted model**

Log of Sale\_price= Y-axis intercept 0.8607+

RESIDENTIAL\_UNITS: \*0.2528 +

COMMERCIAL\_UNITS: \*-0.1993 +

LAND\_SQUARE\_FEET: \*0.2399 +

GROSS\_SQUARE\_FEET: \*-0.2723 +

AGE\_OF\_HOUSE\_AT\_SALE: \*-0.0724 +

NEIGHBORHOOD\_HARLEM-CENTRAL \*-0.1990 +

NEIGHBORHOOD\_HARLEM-EAST \*-0.1843 +

NEIGHBORHOOD\_HARLEM-UPPER\*-0.1818 +

NEIGHBORHOOD\_HARLEM-WEST \*-0.1859 +

NEIGHBORHOOD\_INWOOD \*-0.2222 +

NEIGHBORHOOD\_JAVITS CENTER \*0.1137 +

NEIGHBORHOOD\_LITTLE ITALY \*-0.0830 +

NEIGHBORHOOD\_MANHATTAN VALLEY \*-0.1493 +

NEIGHBORHOOD\_SOHO \*0.0676 +

NEIGHBORHOOD\_TRIBECA\*0.0526 +

NEIGHBORHOOD\_UPPER WEST SIDE (96-116) \*-0.1179 +

NEIGHBORHOOD\_WASHINGTON HEIGHTS LOWER \*-0.1519 +

NEIGHBORHOOD\_WASHINGTON HEIGHTS UPPER \*-0.1717 +

BUILDING CLASS CATEGORY\_08 RENTALS - ELEVATOR APARTMENTS \*0.0090 +

BUILDING CLASS CATEGORY\_10 COOPS - ELEVATOR APARTMENTS \*-0.3499 +

BUILDING CLASS CATEGORY\_14 RENTALS - 4-10 UNIT \*0.0430 +

BUILDING CLASS CATEGORY\_21 OFFICE BUILDINGS \*.0253 +

BUILDING CLASS CATEGORY\_23 LOFT BUILDINGS \* 0.0342 +

BUILDING CLASS CATEGORY\_25 LUXURY HOTELS \*: -0.4842 +

BUILDING CLASS CATEGORY\_27 FACTORIES \* 0.0793 +

BUILDING CLASS CATEGORY\_29 COMMERCIAL GARAGES \* 0.0627 +

BUILDING CLASS CATEGORY\_30 WAREHOUSES \*: -0.0374+

BUILDING CLASS CATEGORY\_33 EDUCATIONAL FACILITIES \* 0.0111+

BUILDING CLASS CATEGORY\_35 INDOOR PUBLIC AND CULTURAL FACILITIES \* -0.4172 + 0.011192978812302462.