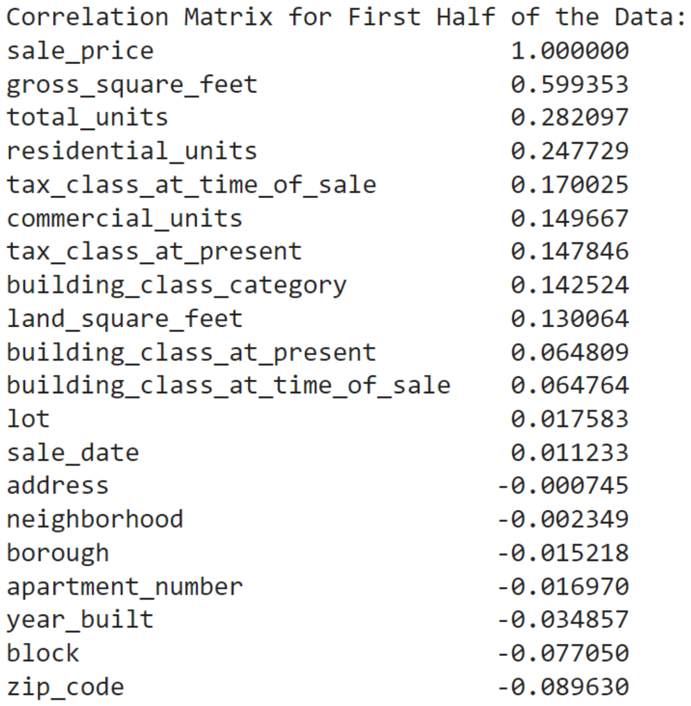
NYC Property Sales Prediction Report

Summary:

Feature Selection:

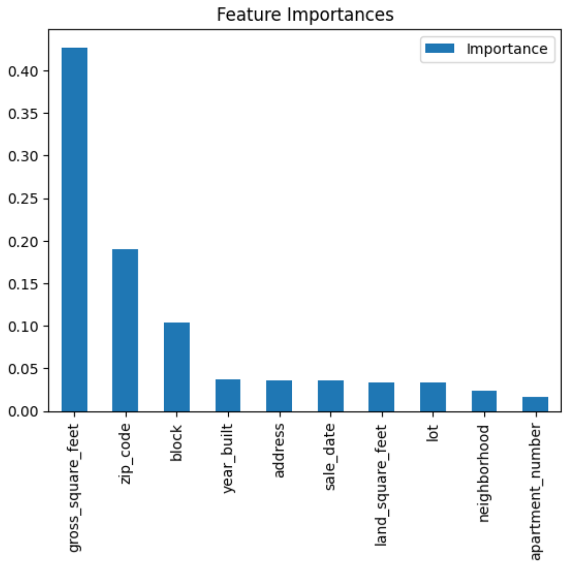
1- Transformation of the data of type string into integers using the LabelEncoder.

2- Use of the correlation analysis to determine the most important features to predict the sale price data. To do so we watch the correlation between every data feature and the target data in our case it is the sale_price data.

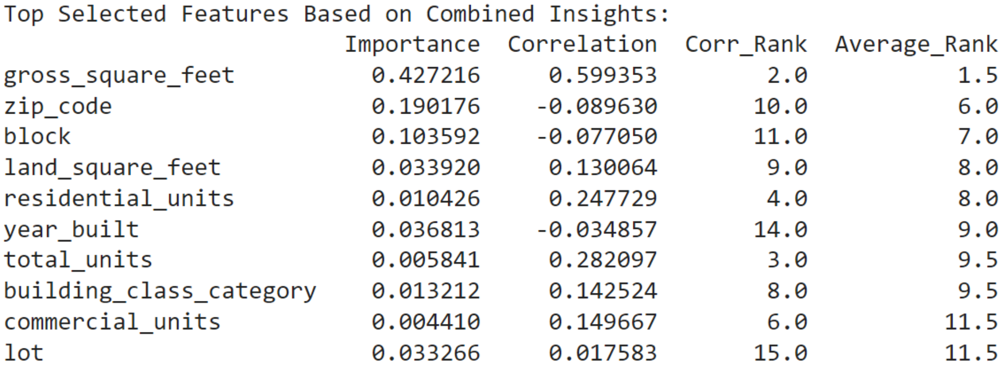


```
Correlation Matrix for First Half of the Data:
sale_price                      1.000000
gross_square_feet               0.599353
total_units                     0.282097
residential_units               0.247729
tax_class_at_time_of_sale       0.170025
commercial_units                0.149667
tax_class_at_present            0.147846
building_class_category         0.142524
land_square_feet                0.130064
building_class_at_present       0.064809
building_class_at_time_of_sale  0.064764
lot                             0.017583
sale_date                       0.011233
address                        -0.000745
neighborhood                   -0.002349
borough                        -0.015218
apartment_number               -0.016970
year_built                     -0.034857
block                          -0.077050
zip_code                       -0.089630
```

3- We use a random Forest regressor model to determine which data fields are the most impactful on the prediction of the sale_price data.



4- We merge both results to have a list of features that will mostly impact positively on the prediction of the sale_price data.



```
Top Selected Features Based on Combined Insights:
                         Importance  Correlation  Corr_Rank  Average_Rank
gross_square_feet          0.427216     0.599353        2.0           1.5
zip_code                   0.190176    -0.089630       10.0           6.0
block                      0.103592    -0.077050       11.0           7.0
land_square_feet           0.033920     0.130064        9.0           8.0
residential_units          0.010426     0.247729        4.0           8.0
year_built                 0.036813    -0.034857       14.0           9.0
total_units                0.005841     0.282097        3.0           9.5
building_class_category    0.013212     0.142524        8.0           9.5
commercial_units           0.004410     0.149667        6.0          11.5
lot                        0.033266     0.017583       15.0          11.5
```

Data preprocessing and eda

1- Data preprocessing:

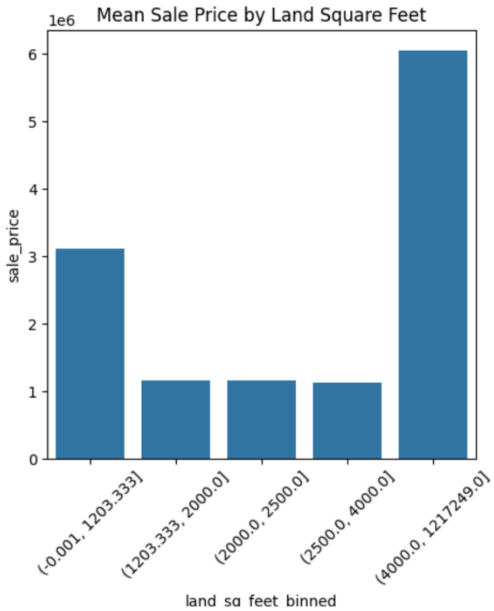
- Selected the data features that are the most impactful for the prediction of the sale price data.
- Removed all the NA values from the data set.

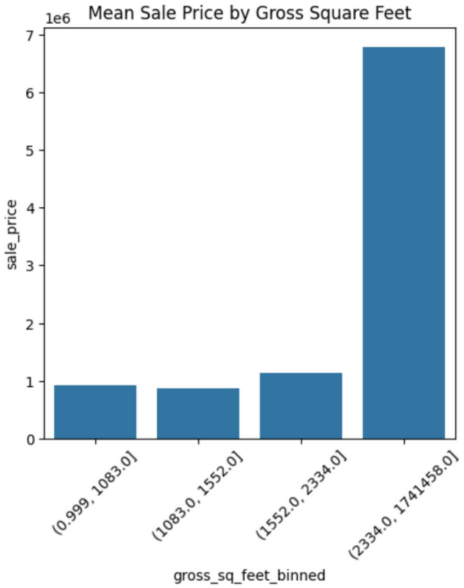
2- EDA:

- We look at some values related to the sale_price:

  Like the minimum value equal to $10 040 and the maximum value equal to $2 155 000 000. With a mean of $2 431 342 and a standard deviation of $18 630 260.

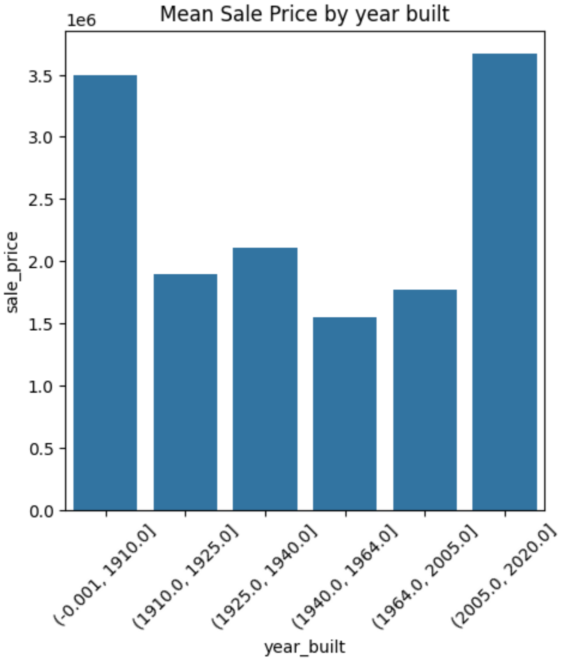
- We look at the table that compare the sale price and the Land square feet/Gross square feet:





We can see that we have approximately a positive proportionality between the size of the land/gross and the price of the property.

- We look at the table that compare the sale price and the year of construction:



We see that the newest and the oldest building are the most expensive, the property build in the 60's – 80's are the cheapest ones.

Model Development:

- Data Preparation
  1- We separate the data in two parts, X and Y, the target, the value that we want to predict.

  2- We convert all the string data into numerical data using LabelEncoder.

  3- We split the totality of the data into 3 parts, the first part for training (X_train/Y_train), the second part, for testing (X_test/Y_test) and the last part for validation (X_valid /Y_valid).

- Hyperparameter tuning:
  1- Create a function that build a Random Forest regressor but where the hyperparameter are variables for example the estimators, the samples split, the max features and the max depth. Then the function returns the average performance of the model.

  2- Define the parameters that should be determined by the tuning.

  3- Start the tuning using the Bayesian optimisation, we loop the optimisation 30 times to get the best result for the prediction algorithm.

- Model Set up:
  1- Create a random forest regressor model using the parameters determined by the hyperparameter tuning algorithm.

  2- Then we train the model using the X_train and the Y_train datasets.

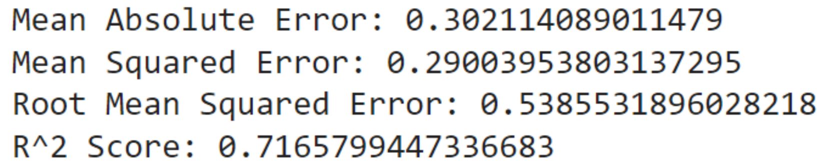
  3- Set a prediction variable using our trained model and the X_test dataset.

  4- Compare the prediction variable with the actual values taken from the Y_test dataset and compute the mean absolute error to see whether the trained model has a good prediction power.

Model evaluation:

1- We use the validation dataset (X_valid /Y_valid) to evaluate the model's efficiency, as well to see if the model is consistent with any datasets.

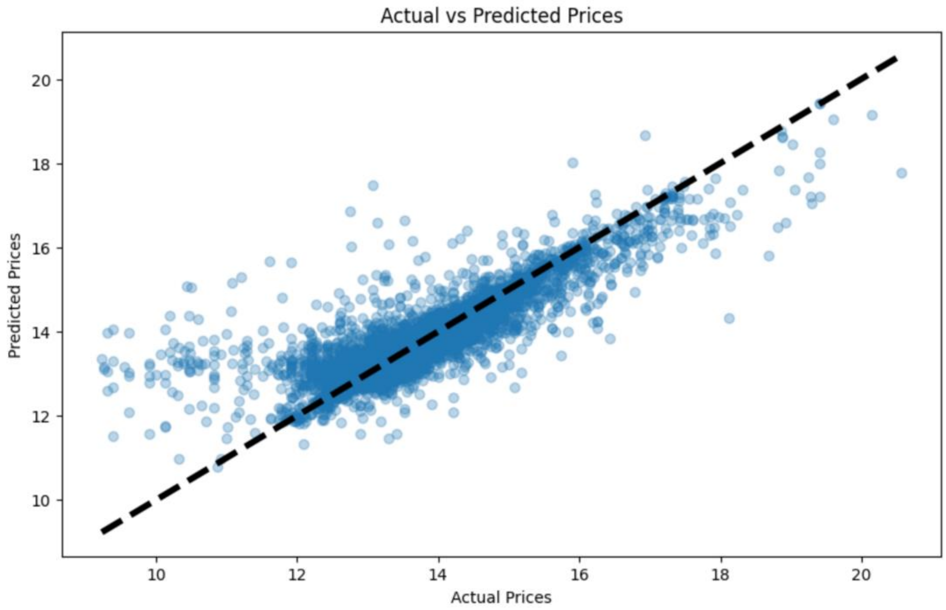
2- We compute the mean squared error, the mean absolute error, the root mean squared error, and the R^2 score and check if they are approximate to the scores of the test dataset (X_test/Y_test).



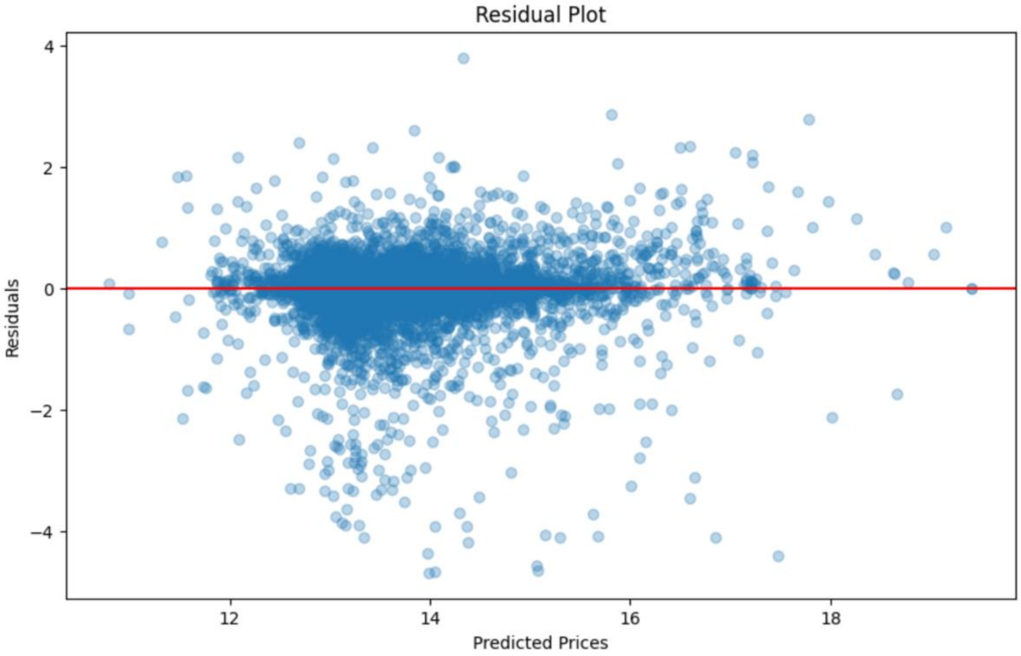
```
Mean Absolute Error: 0.302114089011479
Mean Squared Error: 0.29003953803137295
Root Mean Squared Error: 0.5385531896028218
R^2 Score: 0.7165799447336683
```

3- We plot two graphs to evaluate the performance of our model:

- **Predicted vs. Actual Values Plot:**
  This plot provides a direct visual comparison between the values predicted by the model and the actual values from the validation dataset. If the model's predictions are accurate, the points in this scatter plot will align closely with the diagonal line that stretches from the bottom left to the top right corner of the plot.



- **Residual Plot:**
  This plot focuses on the residuals, which represent the differences between the predicted values and the actual values. The residuals are plotted on the y-axis with the predicted values on the x-axis. For a well-fitted model, these residuals should scatter randomly around the horizontal line at zero.


Residual Plot
Residuals
Predicted Prices