KING COUNTY REAL ESTATE ANALYSIS AND MODELLING

TEAM ALPHA MEMEBERS:

- FLORENCE NGUUNI
- JOSHUA RWANDA
- LEAH KALUMBA
- EDNA WANJIKU
- STEPHANIE MBITHE
- KINOTI MWENDA

1) Business Understanding

1.1.1) Overview

In our analysis, we explored the data provided by Alpha Tennent Stakeholders and build a multiple linear regression model with some of the features stipulated in the dataset. Henceforth, the analysis done and the results came to a solution and on the following factors that have a significant impact on the price of the King County Dataset:

- Have a house by the waterfront
- Increase the number of bathrooms as the number of bedrooms increases
- Improve the overall grade of the home
- Strive to maintain the house to ensure that it's in good condition
- Increase the number of floors and the size of the basement

1.1.2) Introduction

The real-estate business has for a long time been of great interest to investors. Any party interested in investing in the real-estate business will undoubtedly benefit from prior analysis of already existing data on the state of the market in order to minimize risk and maximize ROI.

We got data from various sources such as the `kc_house_data.csv` file from Kaggle that we are going to perform analysis and modelling on.

1.1.3) Problem Statement

We will be reviewing building grade, square-footage of living space, and location-related factors such as proximity to schools, coffee shops, parks, and scientology churches to determine which factors are highly correlated with home sale prices.

1.1.4) Objectives

We focus on the main and specific objective

1.1.5) Main Objective

The main objective is to come up with a predictive /accurate model that is an improvement of the baseline model for better house price prediction in King County.

1.1.6) Specific Objective

- To find out how renovation status affects sale price?
- To determine whether how the number of bedrooms is related to the pricing of the house?
- To determine if the floor number affects the pricing of the house?
- To relate the year-built affects/ is related to the house pricing?
- To find whether the condition of the house is related to the house pricing?

1.1.7) Experimental Design Taken.

Implement changes as we go on with the project

This phase is broken down into four tasks together with its projected outcome or output in detail:

- Collect Initial Data
- Describe Data
- Explore Data
- Verify Data Quality

NB: There was no need to collect any data for this project as it was already provided by the stakeholder. The data consists of house data from King County and is in .csv format

2)Data Wrangling and Understanding

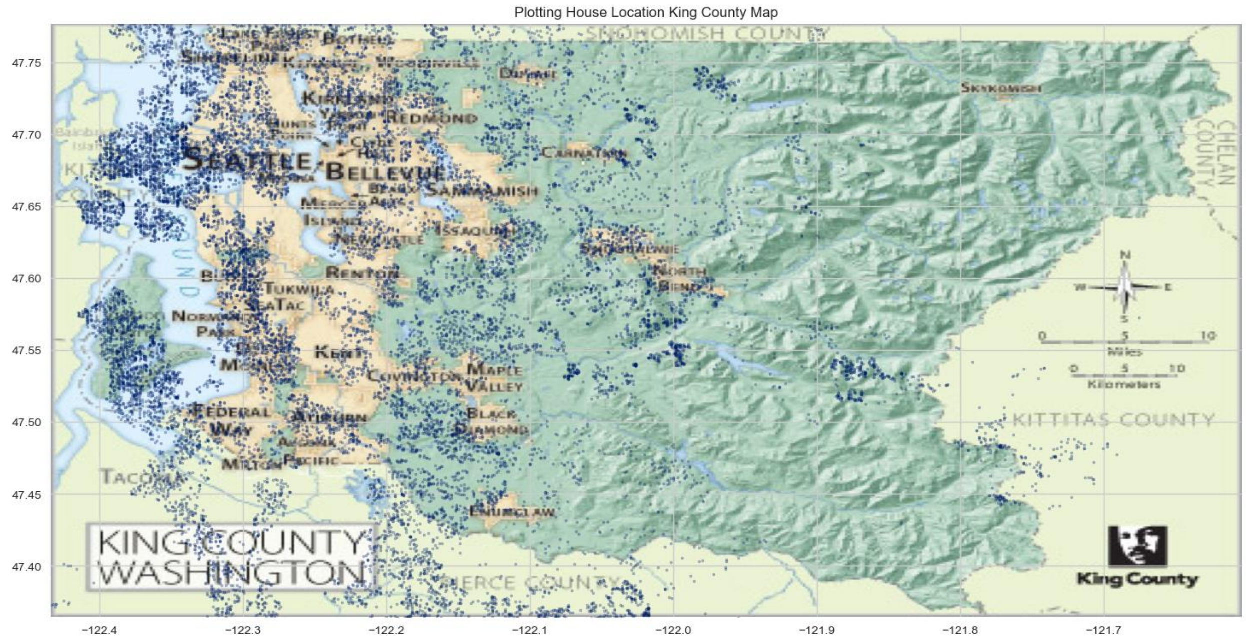
2.1.0) Columns Descriptions

The main dataset we are using comes from the King County housing [dataset](https://www.kaggle.com/datasets/harlfoxem/housesalesprediction) that contains information on house sales between May 2014 and May 2015 consist of the following variables:

- date: Date of house sale
- price: The price which the house sold for
- bedrooms: How many bedrooms the house has
- bathrooms: How many bathrooms the house has
- sqft_living: How much square footage the house has
- sqft_lot: How much square footage the lot has
- floors: How many floors the house has
- waterfront: Whether the house is on the - - - - waterfront. Originally contained 'YES' or 'NO', converted to 0 or 1 for comparative purposes
- view: Whether the house has a view and whether it's fair, average, good, or excellent. Converted to numberical (0-4) for comparative purposes
- condition: overall condition of the house: Poor, Fair, Average, Good, Very Good
- grade: Numerical grading for house
- sqft_above: How much of the houses square footage is above ground
- sqft_basement: How much of the square footage is in the basement
- yr_built: Year the house was built
- yr_renovated: Year the house was renovated, if applicable
- zipcode: House zipcode
- lat: House's latitude coordinate
- long: House's longitude coordinate
- sqft_living15: Average size of living space for the closest 15 houses
- sqft_lot15: Average size of lot for the - - closest 15 houses

2.1.1) Data Mapping

A greater grasp of an area's physical features can be obtained by mapping. In this instance, we depict houses in terms of prices, from the highest prices to the lowest pricing, using the mapping technique. Additionally, reference mapping will be done in the future to help us decide how to build the housing area.



2.1.2) Data Munging/ Cleaning

Data wrangling or data cleaning, is the process of transforming and manipulating raw data into a format that is more suitable for analysis. These are some of the following methods we have implemented for data munging:

- Data Cleaning where we identify the unique values.
- Identified the duplicates and dropped them.
- Also, we have determined the missing values as some we have dropped them in columns such as ‘date’ and ‘id’ columns.
- For the missing values as well, we have changed the view column missing values to NONE then to numerical ordered values
- Column such as sqft_basement column to a numeric data type
- Selecting categorical and numerical variables in our data frame and create a list of the categorical and numerical variables

2.1.2. EDA (Handling Outliers and Providing solutions)

- There were outliers in the price column hence dropping would be unwise.

- Also, outliers were identified in bedroom column and see the outlier in the bedroom's column and as for that we change to 3 bedrooms. Likely due to a typographic error, its encouraged that we change to 3 since it's the average number of bedrooms.

3)Data Analysis

3.1) Patterns

3.1.1) Univariate Analysis

In this section, we'll explore each column in the dataset to see the distributions of features and obtain some useful insights. The main two parts in this section are:

- Categorical Columns (Categorical df)
- Numerical Columns (numerical df)
- Categorical columns

We defined a function below that will take in the categorical data frame created above that contains the categorical columns, that is, ``Condition`` and ``Grade``. 2.1.1.4 Condition

The condition column identifies the condition of the house and the following were the outputs:

• Average	13900
• Good	5643
• Very Good	1687
• Fair	162
• Poor	28

Name: condition, dtype: int64

3.1.1.1)Univariate of Grade

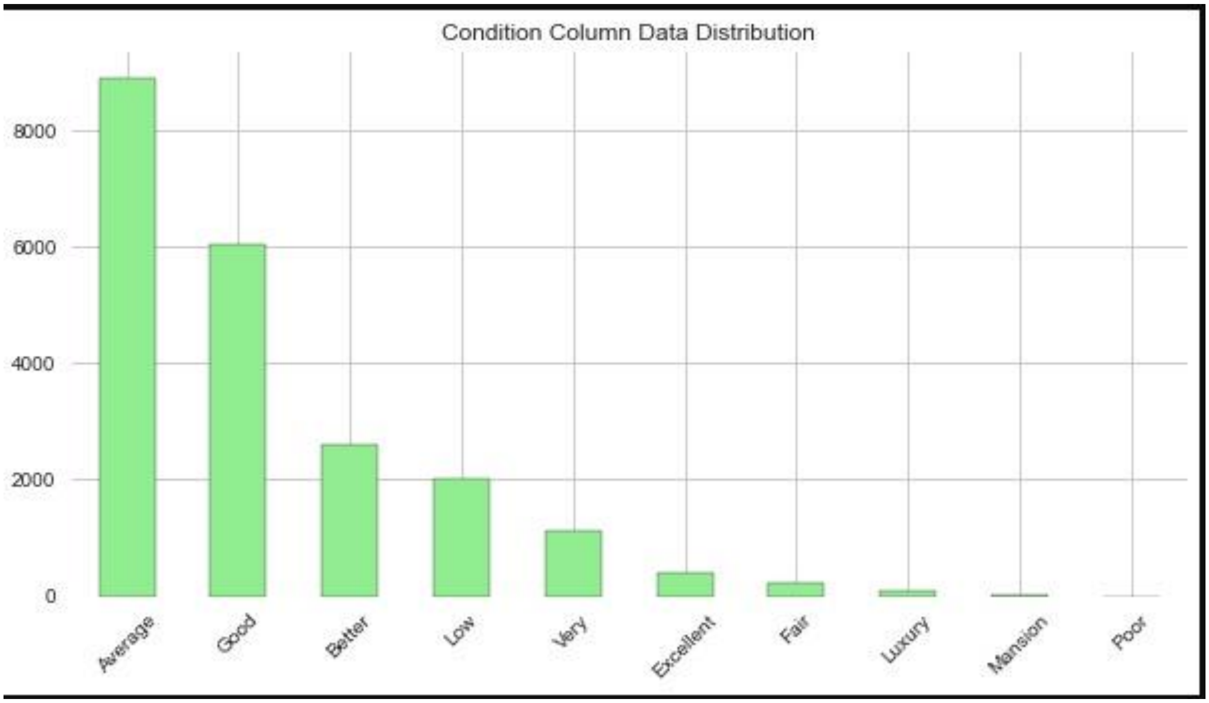
> The grade column describes the home's building and design excellence. The grade corresponds to the caliber of the upgrades' construction. There are 13 grade levels.

Average	8889
Good	6041
Better	2606
Low	2022
Very	1130

Excellent	396
Fair	234
Luxury	88
Mansion	13
Poor	1

Majority of the houses in this dataset range in grade level ``Average`` with ``8889`` houses and the least range in the grade level ``poor`` with ``1`` house.

3.1.1.2) Univariate of the Grade (Plot)

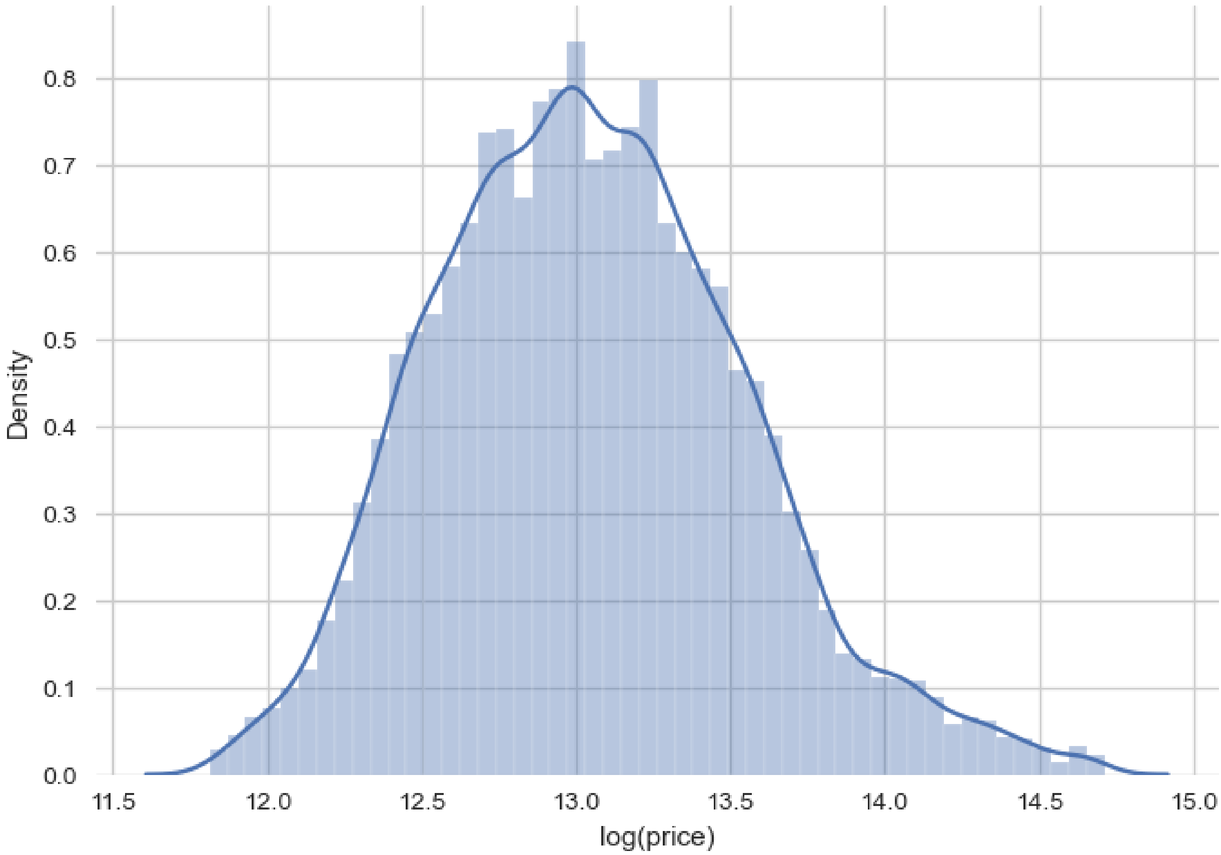


Observation

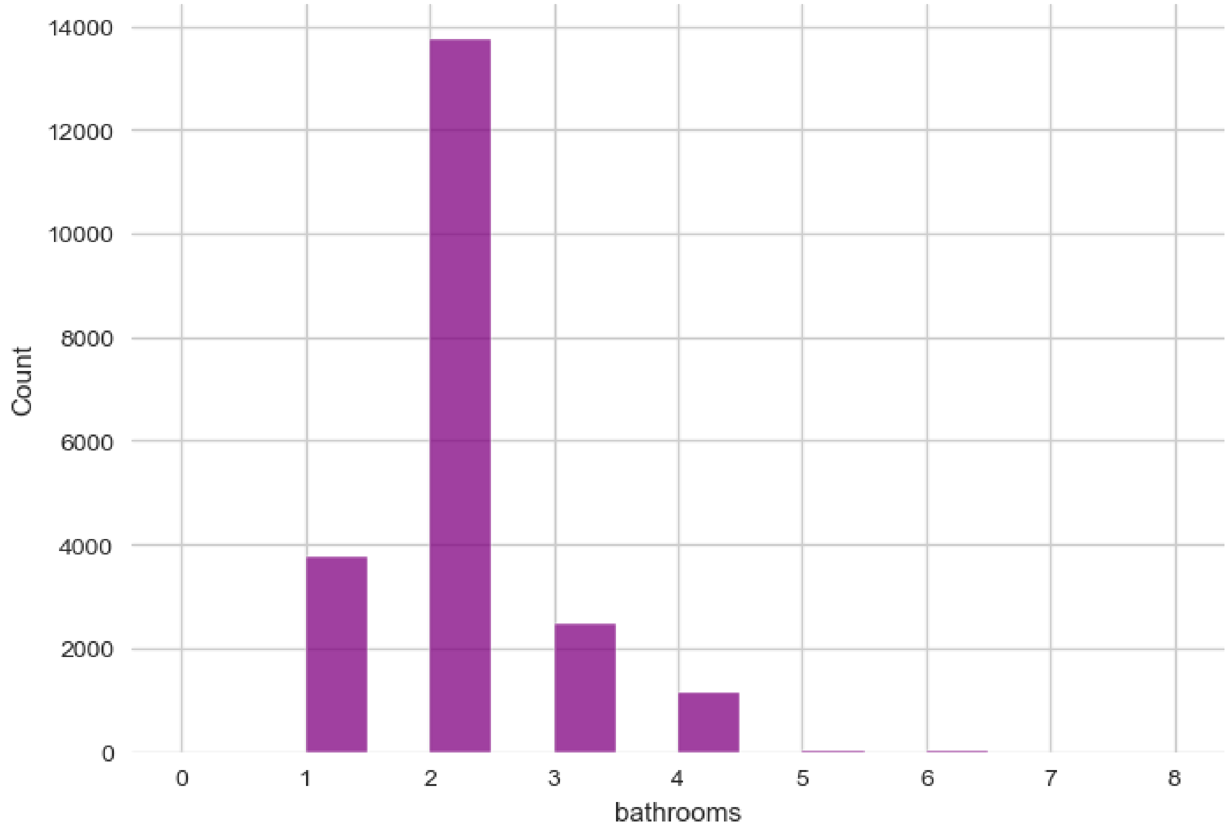
Price is normally distributed although skewed to the right. There may be outliers causing the skew. In the context of real-estate these outliers may be valid and may not warrant dropping.

We are going to improve on the skewness using ``Log Transformation`` with an aim to increase correlation

3.1.1.3) Skewerness and Kurtosis by use of Log Transformation



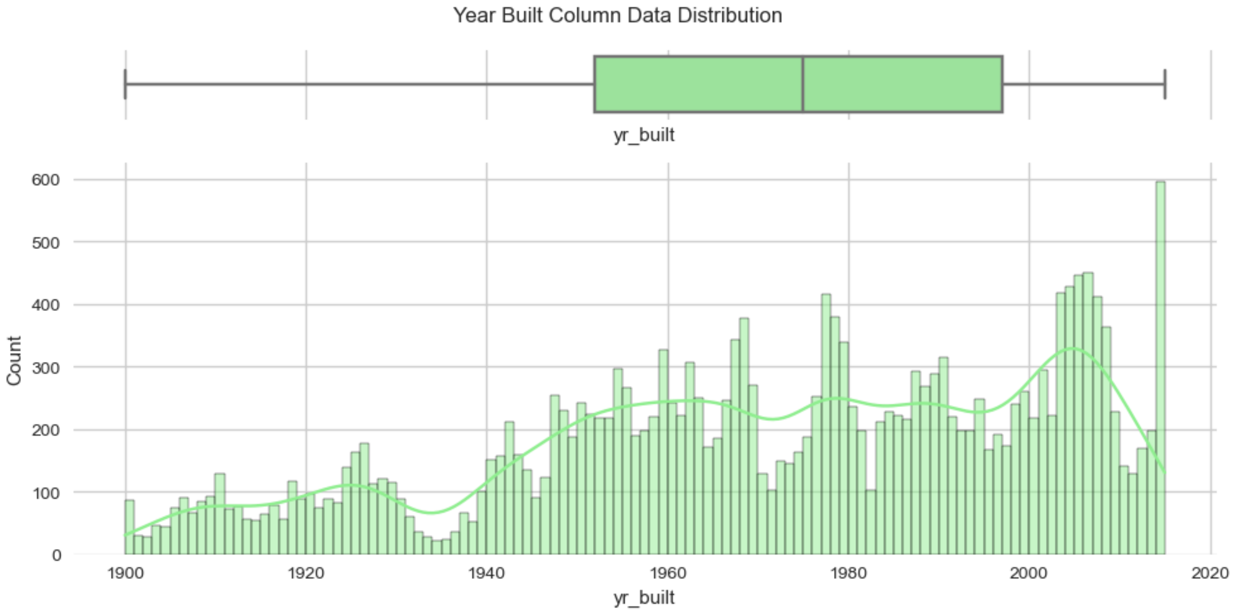
Univariate Analysis in Bathrooms



Observation

Most properties seem to have 2 bathrooms on average. The kde function seems to look a bit strange owing to `bathrooms` being a categorical feature.

3.1.1.4) Univariate in Year Built



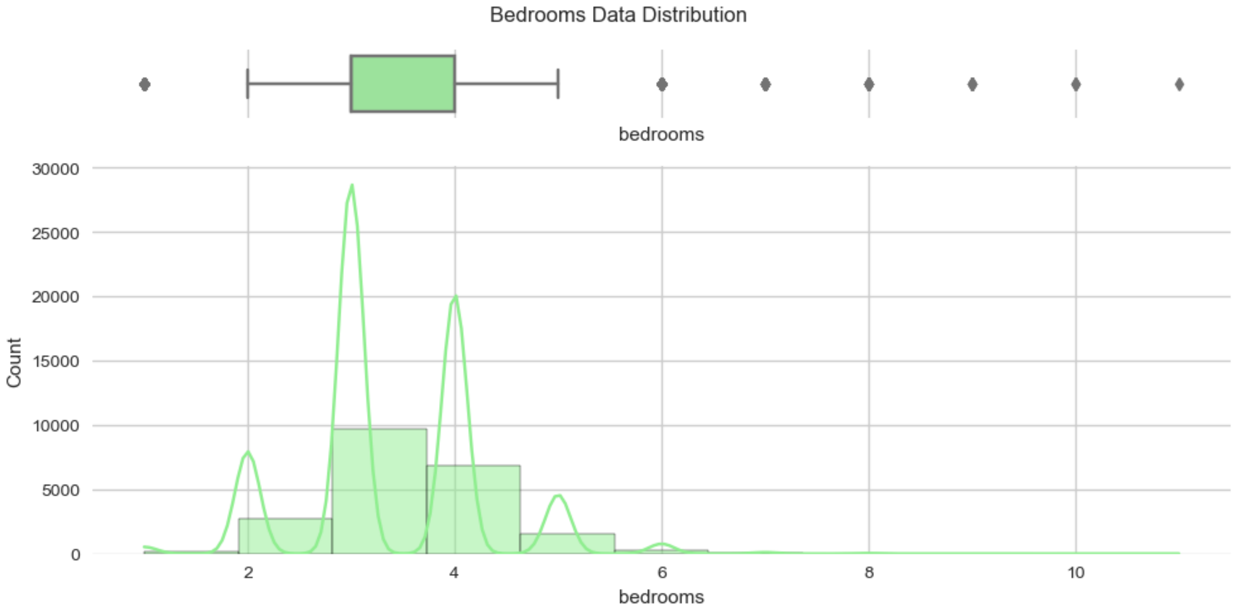
Observation

* From the distributions above we can see that the data is slightly skewed to the left.

* The oldest house in the dataset was built in 1900, and the newest house in the dataset was built in 2015.

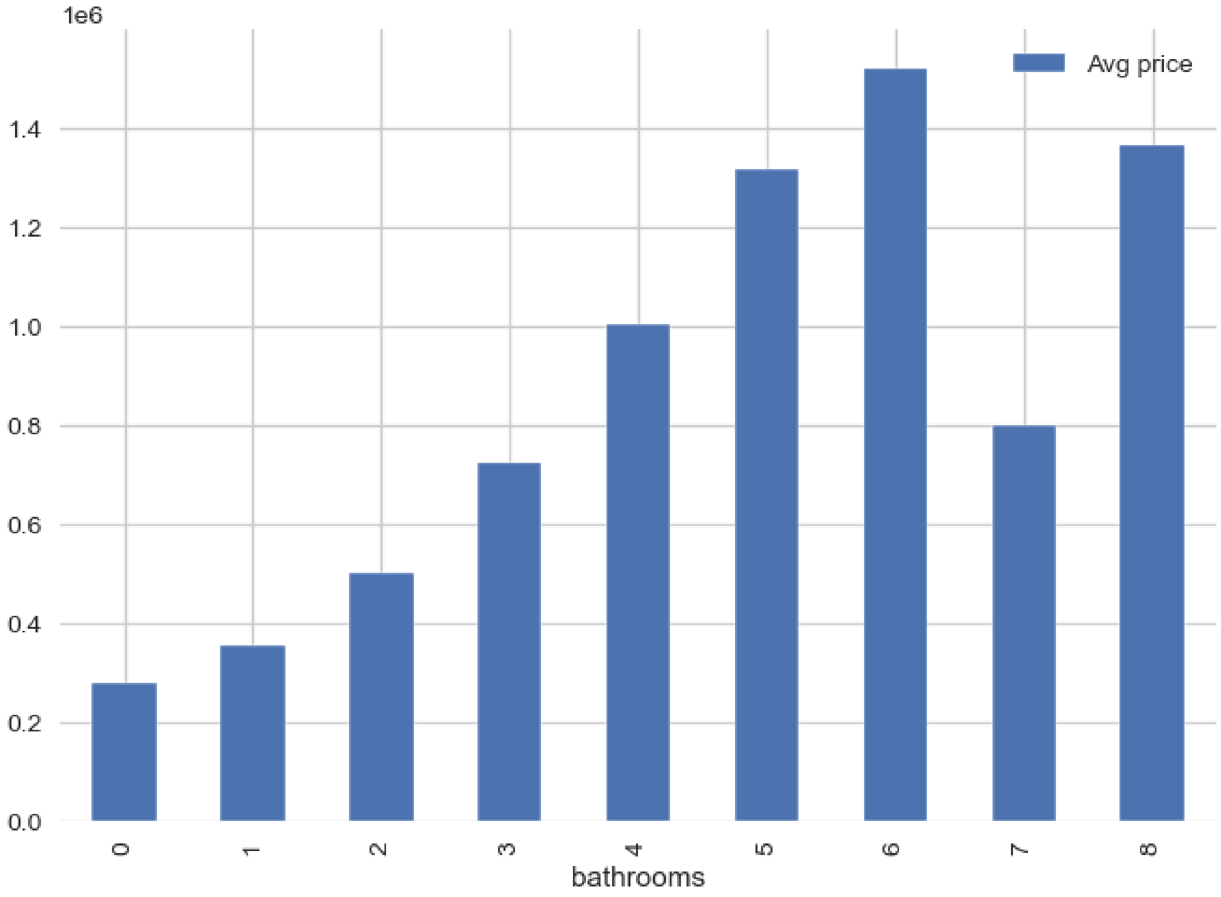
* The mean year the houses in the dataset were built is 1971, and the median year the houses in the dataset were built is 1975.

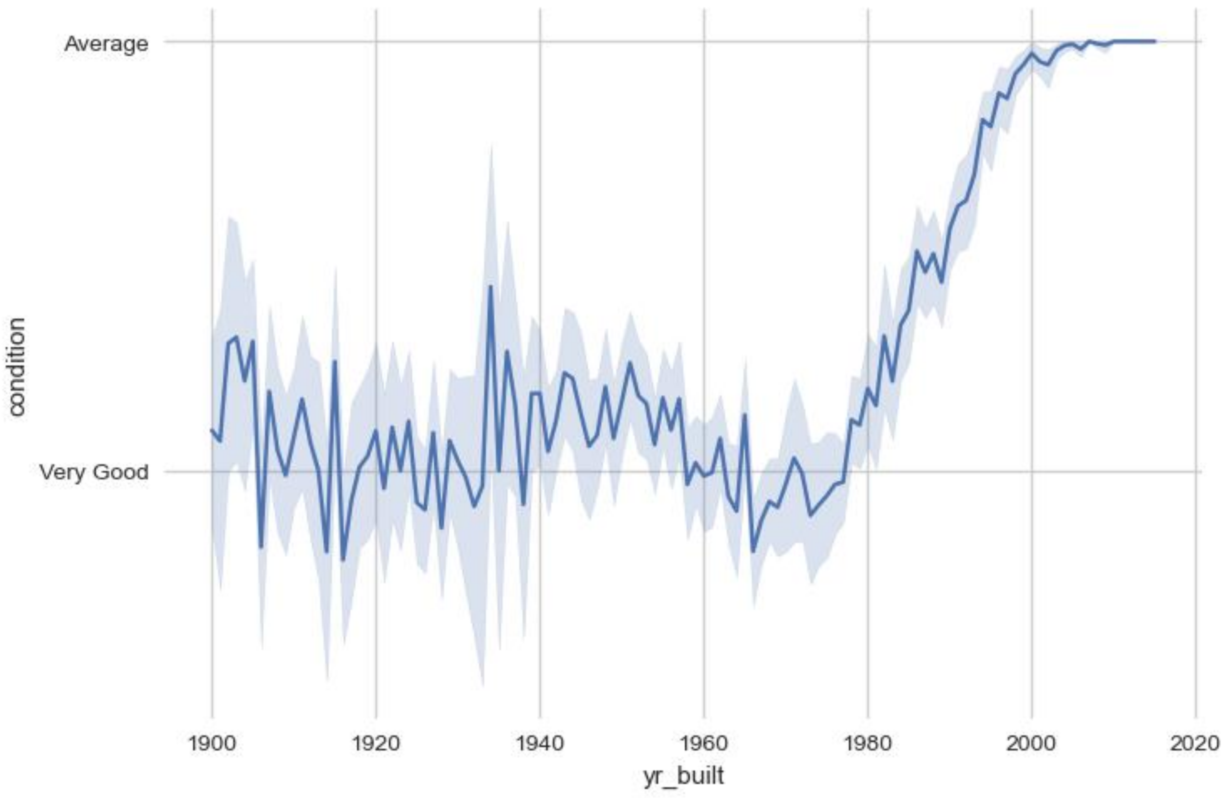
* The standard deviation of the year-built column is 29.37



- The maximum number of bedrooms in the dataset in 11
- The minimum number of bedrooms in the dataset is 1
- The mean number of mean number of bedrooms is 3.37 and the median number of bedrooms is 3
- The standard deviation of the bedroom column is 0.93

Bivariate (Bathroom Against Price)





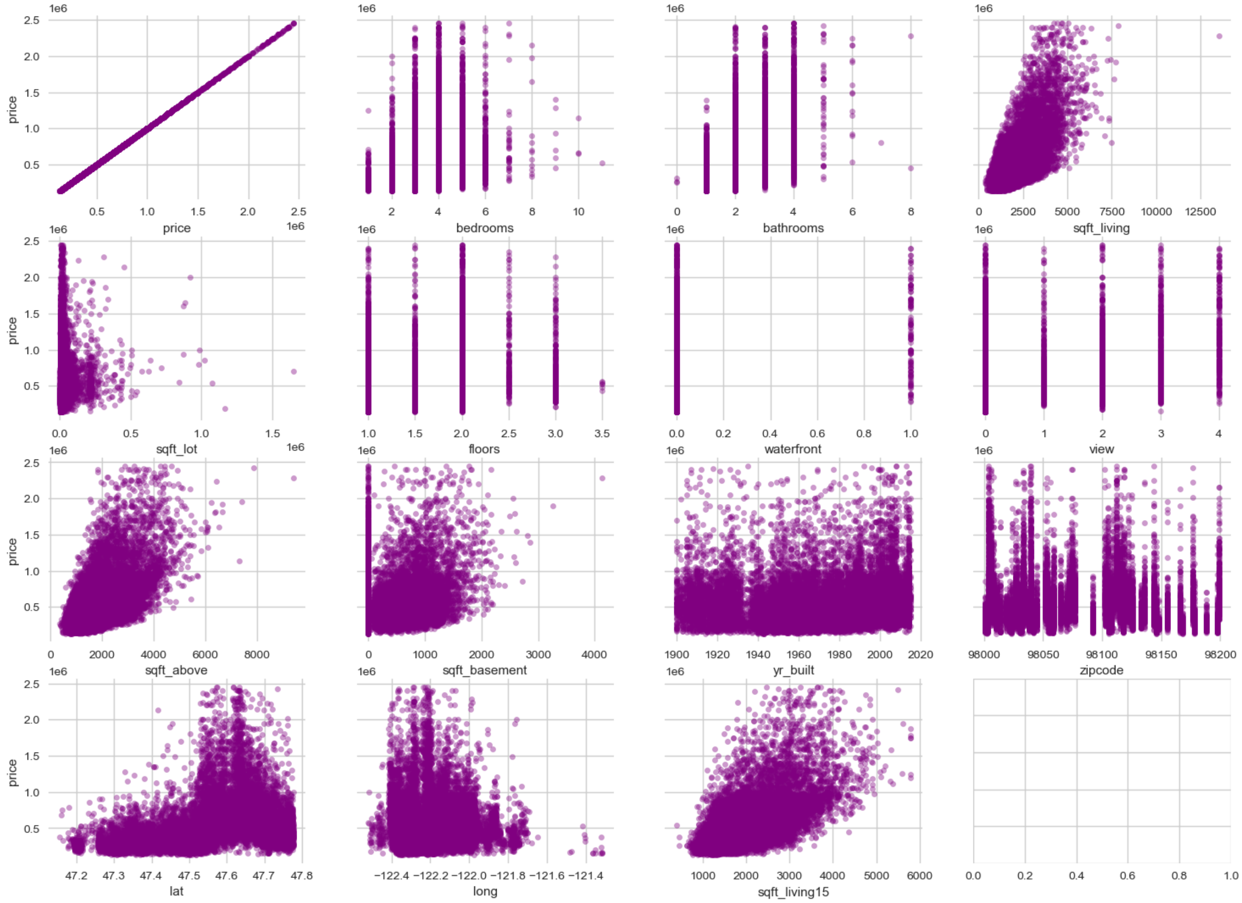
Observation``

* For most houses built in the $20th$ century *i.e.,* 1900-early 1990s houses were in Very good condition.

* For most houses built from the beginning of the $21st$ century, the condition is Average. There seems to be a decline in condition of houses built over time

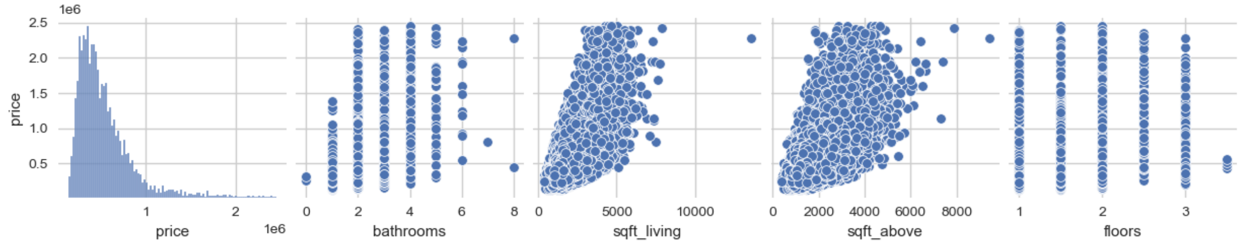
Multivariate Analysis

``Multivariate Analysis`` refers to the analysis of more than two variable or data set, typically using statistical methods.



Observations``

From the plots above, we can observe that some of the relationships within the variable are linear while other are non-linear.



Observations: ``

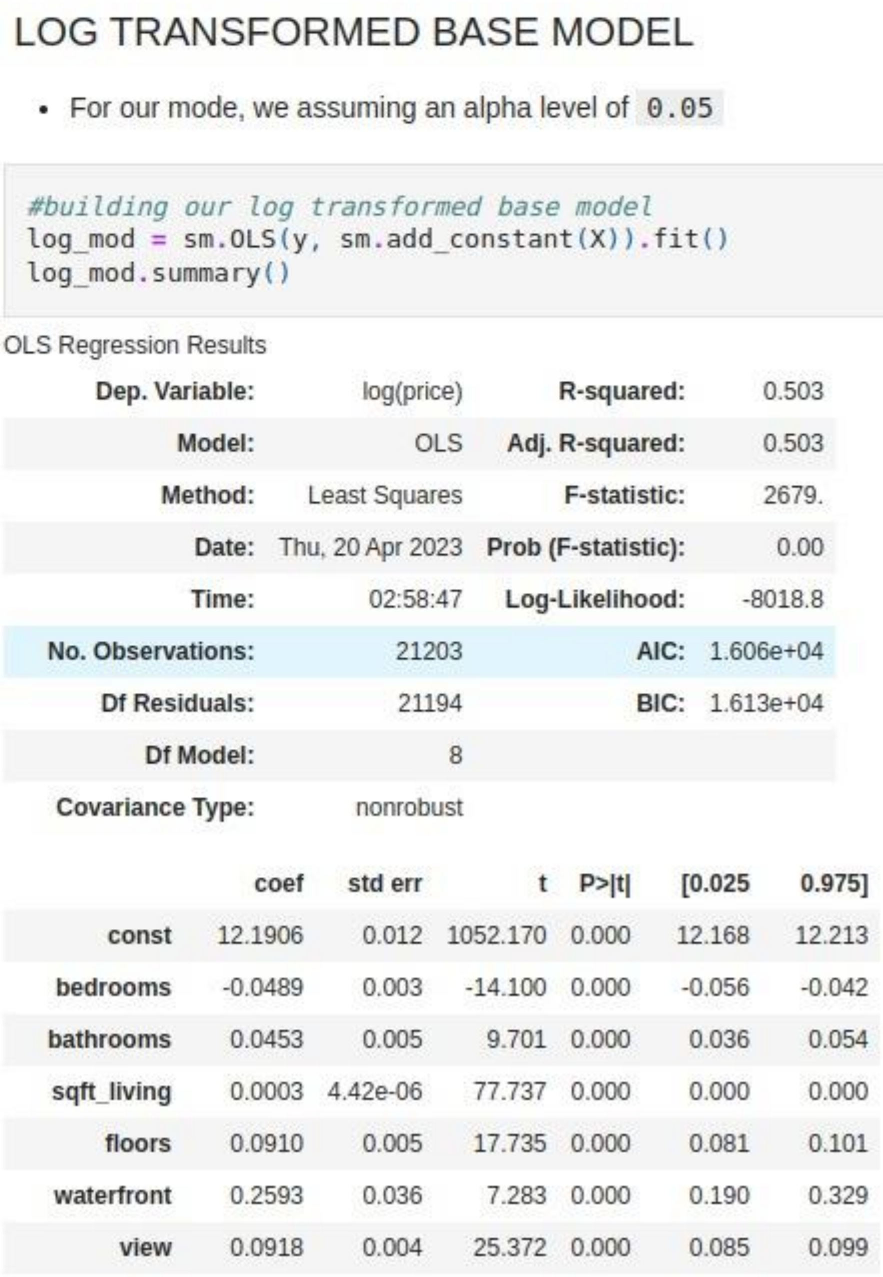
`sqft-living` and `sqft_above` seem to have a linear relationship with `price`

4) Modelling

An intercept-only model can be used to establish a baseline for comparison with more complex models that include predictors such as square footage or number of bedrooms. By fitting an intercept-only model, we can estimate the average value of the response variable, which in this case is the house price, without taking into account any other factors. This provides a point of reference for evaluating the predictive power of other models and determining whether they offer a significant improvement over the intercept-only model.

Additionally, an intercept-only model can also be used to test the overall significance of the model, which can help to determine whether the data provides sufficient evidence to conclude that there is a relationship between the response variable and the predictors.

Log Base Modelling



LOG TRANSFORMED BASE MODEL

- For our mode, we assuming an alpha level of `0.05`

```python
#building our log transformed base model
log_mod = sm.OLS(y, sm.add_constant(X)).fit()
log_mod.summary()
```

OLS Regression Results

Dep. Variable:	log(price)	**R-squared:**	0.503
Model:	OLS	**Adj. R-squared:**	0.503
Method:	Least Squares	**F-statistic:**	2679.
Date:	Thu, 20 Apr 2023	**Prob (F-statistic):**	0.00
Time:	02:58:47	**Log-Likelihood:**	-8018.8
No. Observations:	21203	**AIC:**	1.606e+04
Df Residuals:	21194	**BIC:**	1.613e+04
Df Model:	8		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	12.1906	0.012	1052.170	0.000	12.168	12.213
bedrooms	-0.0489	0.003	-14.100	0.000	-0.056	-0.042
bathrooms	0.0453	0.005	9.701	0.000	0.036	0.054
sqft_living	0.0003	4.42e-06	77.737	0.000	0.000	0.000
floors	0.0910	0.005	17.735	0.000	0.081	0.101
waterfront	0.2593	0.036	7.283	0.000	0.190	0.329
view	0.0918	0.004	25.372	0.000	0.085	0.099

view	0.0918	0.004	25.372	0.000	0.085	0.099
renovated	4.828e-17	5.93e-18	8.138	0.000	3.67e-17	5.99e-17
grade_idx	0.0107	0.001	10.467	0.000	0.009	0.013
condition_idx	0.0384	0.002	19.078	0.000	0.034	0.042

Omnibus:	3.880	**Durbin-Watson:**	1.989
Prob(Omnibus):	0.144	**Jarque-Bera (JB):**	3.853
Skew:	0.023	**Prob(JB):**	0.146
Kurtosis:	2.953	**Cond. No.**	1.71e+21

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The smallest eigenvalue is 3.69e-32. This might indicate that there are
strong multicollinearity problems or that the design matrix is singular.

`Observations from our base model:`

- An R-squared value of 0.503 means that approximately 50.3% of the variation in the dependent variable can be explained by the variation in the independent variable(s) in the regression model.

Assumption for Linear Regression in our model

1. *Linearity*

The dependent variable and the independent variable(s) should have a linear relationship.

In [451…

```python
def linearity(model):
    # Calculate the fitted values and residuals of the model
    fitted_y = model.fittedvalues
    residuals = model.resid

    # Create a scatter plot of the predicted values (fitted values) against the residuals
    fig, ax = plt.subplots(figsize=(6, 2.5))
    _ = ax.scatter(fitted_y, residuals)

    # Add a horizontal line at y=0 to help visualize the deviations of the residuals from the line
    ax.hlines(y=0, xmin=fitted_y.min(), xmax=fitted_y.max(), colors='r', linestyles='--')

    # Set the x-axis label to 'Predicted values', the y-axis label to 'Residuals', and the plot title to 'Linearity Check: Predicte
    ax.set_xlabel('Predicted values')
    ax.set_ylabel('Residuals')
    ax.set_title('Linearity Check: Predicted vs. Residuals')

    # Display the plot
    plt.show()

# Call the linearity function with a linear regression model as an argument to check for linearity
linearity(log_mod)
```

5) Challenges

There were challenges in the following areas:

- Skewerness:

✓ Price: The distribution of house prices is typically skewed to the right, with a long tail of high-priced houses.
✓ Sqft_living: The distribution of square footage of living space is also often skewed to the right, with a long tail of larger houses.
✓ Sqft_lot: The distribution of lot sizes is frequently skewed to the right, with a long tail of larger lots.
✓ Bedrooms: The distribution of the number of bedrooms is typically skewed to the left, with a majority of houses having 3 or fewer bedrooms.
This can result in biased estimates and incorrect inferences

- Log Transformation

✓ Extreme values: Log transformation can amplify the effect of extreme values, particularly if the variable has a long tail.

- Getting the model best fit

✓ Complexity of the data: The King County dataset contains many variables, some of which are highly correlated with each other. This can make it difficult to determine which variables are the most important predictors and which can be omitted from the model.
✓ Nonlinear modeling: Using nonparametric methods such as decision trees, random forests, or neural networks to capture nonlinear relationships between the predictor and outcome variables.

6) Recommendations

To address skewness problems in the King County dataset, it may be necessary to transform the data using techniques such as log-transformation or power transformation amongst other techniques.

Removing the extreme outliers and utilize a different transformation which is less sensitive to the outliers

7)Proposed Solution

- Utilize feature selection methods including correlation analysis, mutual information, and recursive feature elimination to pinpoint the most crucial features.
- Features that are unnecessary or redundant should be removed.
- Choosing a model and training it:
- Train a variety of regression models, including gradient boosting, decision trees, random forests, and linear regression.
- To assess each model's performance, use cross-validation.
- Using the cross-validation results as a guide, choose the model that performs the best.
- tweaking for hyperparameters
- Use methods like grid search or random search to fine-tune the model's hyperparameters.