**Logo, company name

Description automatically generated**

Log Book 2

Table of Contents

[**1.0 Introduction** 3](#_Toc90105820)

[**2.0 Measures of Association** 4](#_Toc90105821)

[**2.1 Spearman rank-order correlation coefficient:** 4](#_Toc90105822)

[**2.2 Chi-Square test:** 4](#_Toc90105823)

[**2.3 Principal Component Analysis:** 5](#_Toc90105824)

[**2.5 Correlation and Chi-square test between variables:** 7](#_Toc90105825)

[**2.5.1 Mas.Vnr.Area** 7](#_Toc90105826)

[**2.5.2 Gr.Liv.Area** 7](#_Toc90105827)

[**3.0 Regression Models and Insights:** 8](#_Toc90105828)

[**3.1 Linear Regression model 1:** 8](#_Toc90105829)

[**3.2 Multiple Linear Regression Model 2:** 10](#_Toc90105830)

[**3.3 Multiple Linear Regression Model 3:** 13](#_Toc90105831)

[**4.0 Reflective Summary** 16](#_Toc90105832)

[**Appendix** 17](#_Toc90105833)

# **1.0 Introduction**

We have a database of ames homes. To begin, we'll fix any issues with the variable's data quality, such as removing any outliers, and then add some additional variables, such as the total number of bathrooms, which is the sum of the variables "Full bath, half bath, basement full bath, and basement half bath." Besides this one, we included four others: "Garage, Bsmt, Pool, total\_sf." After that, we removed the corresponding columns from the table.

The focus of this logbook is a regression model, which will be used to estimate the sales price utilizing numerous variables from the ames data set. To begin, though, we'll conduct a principal component analysis to choose which variables to utilize. Then we'll conduct correlation and hypothesis tests to determine how well the variables that we'll acquire from the principal component analysis correlate with one another. Regarding our models we are going to start with a linear model and then move towards more complex multi linear regression models with different variables. We are also going to do a weighted multi linear regression model at the end. **Regarding the split of the model, we have done a stratified random split in the ration of 80:20 keeping sales price column as our target variable so that the training and the test data have all kinds of values and one data does not end up with higher or lower values.**

# **2.0 Measures of Association**

An association measure is a factor or coefficient that can express the strength of a correlation between two or more variables in statistics.

For each variable, the procedure for determining the strength of a relationship is different. A correlation or regression analysis, for example, can be used to calculate an association measure. Correlation and association are sometimes used interchangeably; however, correlation refers to linear correlation, whereas association refers to any link between variables. It is possible to measure data using one of the following three methods: an interval/ratio, ordinal, or nominal/categorical scale. These three qualities can be categorized as continuous, integer, and qualitative. For logbook we are going to use two different methods of analysis mentioned below

## **2.1 Spearman rank-order correlation coefficient:**

Spearman's rank-order correlation coefficient (Spearman rho) measures the strength of a link between two variables on an ordinal or ranked scale. Likert-scale data, for example, is subject to Spearman correlation analysis because it is not interval in nature. Any interval data can be translated into ranks and evaluated using the Spearman rho; however, this results in a significant loss of information. This strategy can be utilized if one variable of interest is assessed on an interval scale while the other is measured on an ordinal scale. Like Pearson's correlation coefficient, Spearman's rho can be evaluated for significance like Pearson's rho.

## **2.2 Chi-Square test:**

Chi-square tests begin with a handful of assumptions and two hypotheses before comparing an obtained statistic with a critical statistic, all while using the same sample distribution and alpha level. Assumptions for the chi-square test are simple: the data must be random, and the variables in issue must be nominal or ordinal. All chi-square tests have the same general null and research hypotheses when it comes to testing the hypotheses. According to the null hypothesis, both variables are unrelated, whereas the research hypothesis is related. If our test statistic exceeds the critical statistic at our specified alpha level, we conclude that our findings are statistically significant.

## **2.3 Principal Component Analysis:**

Large datasets are becoming more common across a wide range of fields. When dealing with large datasets, strategies are needed to reduce their dimensionality in an interpretable manner so that the information is not lost. Principal component analysis (PCA) is one of the earliest and most extensively used methods for this purpose, even though several others have been created since then. Its goal is to keep as much 'variability,' or statistical information, as feasible while reducing the dimensionality of a dataset.

Following is the output from the pca test of our ames housing data set.

Chart, diagram

Description automatically generated

**Fig 1: PCA ames**

Following the pca clusters just to clarify the results a little more I went ahead and ran a correlation test on all the variables by creating the dummy variables for the factor variables and using the numeric variables as it is. Then based on the results from the correlation test and pca clusters the variables for the regression model were decided. Once the variables were decided I plotted some correlation matrices on a graph in which when you hover the cursor over the graph it gives you the correlation between variables. Two matrices were plotted one with all factor and numeric variables which had dummy variables for factors and one only for numeric variables.

A picture containing chart

Description automatically generated

**Chart

Description automatically generatedFig 2: Correlation between all the variables**

**Fig 3: Correlation between numeric variables**

## **2.5 Correlation and Chi-square test between variables:**

The variables were chosen according to their correlation and hypothesis with respect to the sale price variable and an addition chi-square was also performed to confirm the decision. Following are the results of correlation and chi-square test of some variables as it is not feasible to show all the results in the report.

Variables with p value below 0.05 or 5% were chosen with a sample estimate value of more than 0.5.

### **Text, letter Description automatically generated2.5.1 Mas.Vnr.Area**

**Fig 4: Chi-Square test**

Text

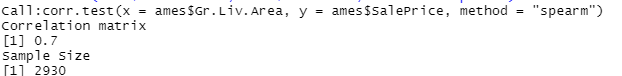
Description automatically generated

**Fig 5: Spearman’s Correlation Test**

### **2.5.2 Gr.Liv.Area**

Text

Description automatically generated

**Fig 6: Chi-Square Test**

**Fig 7: Spearman’s Correlation Test**

# **3.0 Regression Models and Insights:**

In R programming, regression analysis is a collection of statistical techniques used to discover the relationship between dataset variables. For the most part, regression analysis is used to examine the relationship between the dependent and independent variables in a dataset. If the value of one independent variable changes while the other independent variables remain constant, regression analysis can see how the dependent variables change. As a result, a regression model can be constructed, and the values associated with a shift in one of the independent variables predicted.

Following are some models that we built to predict ames housing the data we have picked one random house from the data set to calculate the value manually and see if the models are accurate or not.

## **3.1 Linear Regression model 1:**

We started with a base line model with only one variable that garage area and looking at the coefficient it tells us for every sq. ft we increase the price rises by 300.439$. Our r squared value is pretty low but we will still try to manually predict the value of a house, so we randomly picked a house from the data set which has 700sq. ft of garage are so if we go according to that the value of the house will be -

700\*300.439 – rmse value (76243) = 134,064$

But the actual value of the house was 142600$, so our model despite having allow r square is pretty accurate let’s have a look at the result and the residual graphs.

Text

Description automatically generated

**Fig 8: model 1 Results**

Timeline, scatter chart

Description automatically generated Chart, line chart

Description automatically generated Chart, scatter chart

Description automatically generated Chart, scatter chart

Description automatically generated

The graphs also don’t show any patterns that can be followed meaning that there is still room for improvement. So we go ahead and make another model using multiple variables.

## **3.2 Multiple Linear Regression Model 2:**

For this model we added a whole bunch of variables which were co-related to our sales price column. The columns that we added were total bathrooms, graded living area, masonry veneer area, total basement sq. ft and garage cars. So, looking at the coefficients of this model we can see that every variable is making a whole lot of effect on the sales price for e.g.: if the total bathrooms are increased to 2 the value increase by 15549$ or if we increase masonry veneer area by 1 sq. ft the value increases by 55.247$ and so on. We can see that our r square here jumped from 0.36 in the previous model to 0.7 making our model a little bit more accurate and our rmse value was decreased which as well is a good sign. Let’s have a look at the exact stats and residual maps for assumption checks.

Text

Description automatically generated**Fig 9: model 2 results**

Chart, scatter chart

Description automatically generated

Chart, line chart

Description automatically generated Chart, scatter chart

Description automatically generated Chart, line chart

Description automatically generated

We can clearly see there has been a slight change in the graphs but still no patterns are visible so let’s add another few variables and try to make the graph more accurate.

## **3.3 Multiple Linear Regression Model 3:**

So, the next step for us is to add another few variables to see if we can make the model more accurate. The variables that we added this time are lot are, lot frontage, year remodelled, fireplaces, year built, wood deck sq. ft, open porn sq. ft, neighbourhood, overall quality, pool, land slope and year sold. Through the coefficients of this variable, we can see that some variables are even decreasing the price of the property due to their nature or location or other factors. The noticeable feature about this model is our r square value reached till 0.8 and anything above 0.7 is to be considered a good model. But just to test the accuracy of the model we will pick a house and try and calculate the value manually from the coefficients of the model and then compare and have a look about the accuracy of the model. If we calculate the sale price by coefficients manually the sales price for the house, we chose comes 142,500$ where as the original price in the dataset for the house is 145000$ which is pretty close to the actual output. Let’s look at the details and the graphs for this model.

Chart, scatter chart

Description automatically generated Chart

Description automatically generated

Chart, scatter chart

Description automatically generated Chart, line chart

Description automatically generated

We cann see the graphs forming clusters and forming patterns and the model line seems to be pretty accurate. With the rmse value and the r square being 0.80 I think the model is pretty accurate below are the stats for the model.

Table

Description automatically generated

**Fig 10: model 3 results**

# **4.0 Reflective Summary**

It wasn't that I didn't know much about statistics; I didn't know how to link it to R. While learning "R" for the first time, I was apprehensive since my background is non-technical and I had no prior experience with coding or programming. A book by "Peter Dalgaard" called "Introductory Statistics with R" sparked my interest in statistics. In other words, I started from the beginning. My investigation continued after that, with new themes on YouTube and the internet, along with more exploration of the R console. As a result of this lesson, I believe my logbook will improve my understanding of statistics and R. My initial apprehension was replaced by excitement as I continued through this module. The content became second nature to me rather than complex. After looking through it, we can do a lot with it, as I discovered. As previously stated, I found R to be a fascinating and effective tool for developing regression models and doing statistical studies. For me, the ultimate objective is to become a business analyst in business intelligence, which necessitates a combination of technical and analytical expertise, and I've realized that R is the statistical language with the best analytics libraries. To be a practical business analyst, you must locate, retrieve, alter, and provide insights from data. As a business analyst, I need to have a working knowledge of database systems and machine learning, which is why I chose R.

# **Appendix**

getwd()

setwd("C:/Users/Mohd Hamad/OneDrive/Desktop/Induction")

library(readxl)

library(ggplot2)

library(psych)

library(tidyverse)

library(dplyr)

library(skimr)

library(summarytools)

library(VIM)

library(Hmisc)

library(imputeTS)

library(dplyr)

library(outliers)

library(ggthemes)

library(ggsci)

library(ggpubr)

library(hexbin)

library(ggridges)

library(dummies)

library(gganimate)

library(gifski)

library(knitr)

#get rid of scientific notation throughout

#by setting an immensely high integer penalty

options(scipen = 999999)

### Before heading towards the data creating a mode function as we might need it

X <- c(1, 2, 2, 3, 5, 2, 6, 2)

mode <- function(x) {

ux <- unique(x)

ux[which.max(tabulate(match(x, ux)))]

}

##Read in the Data

ames\_org <- read\_xlsx("ames\_train.xlsx")

## converting ames\_train to ames so as to create duplicate data and working on a dataframe is easy

ames = data.frame(ames\_original)

## descriptive stats

summary(ames)

glimpse(ames)

skim(ames)

describe(ames)

## Describe table print to attach in appendix

my\_summary <- dfSummary(ames)

my\_describe <- descr(ames)

print(my\_summary, file = "Describe\_ames.txt", style = "grid")

print(my\_describe, file = "Describe\_ames.txt", append = TRUE)

#A histogram of SalePrice just to see how the data is distributed

ames %>%

ggplot(aes(SalePrice)) +

geom\_histogram(binwidth = 100000, fill = "#97B3C6") +

theme\_bw()

#correlation

ames %>%

select(where(is.numeric)) %>%

cor()

#another correlation test

ames %>%

select(where(is.numeric)) %>%

corr.test()

corr.test(ames\_2, ames$SalePrice)

## changing a column name

colnames(ames)[colnames(ames) == "Functio.l"] <- "Functional"

##Split the data into types of variables

###converting categorical columns to factors

ames\_categorical = ames %>%

select(MS.SubClass, MS.Zoning, Street, Alley, Lot.Shape, Land.Contour, Utilities, Lot.Config,

Land.Slope, Neighborhood, Condition.1, Condition.2, Bldg.Type, House.Style,

Overall.Qual, Overall.Cond, Roof.Style, Roof.Matl, Exterior.1st, Exterior.2nd, Mas.Vnr.Type,

Exter.Qual, Exter.Cond, Foundation, Bsmt.Qual, Bsmt.Cond, Bsmt.Exposure, BsmtFin.Type.1,

BsmtFin.Type.2, Heating, Heating.QC, Central.Air, Electrical, Kitchen.Qual, Functional,

Fireplace.Qu, Garage.Type, Garage.Finish, Garage.Qual, Garage.Cond, Paved.Drive,

Pool.QC, Fence, Misc.Feature, Sale.Type, Sale.Condition) %>%

mutate(across(!MS.Zoning, as.factor))

###converting continuous and discrete data columns to numerical data

ames\_numerical = ames %>%

select(Lot.Frontage, Lot.Area, Mas.Vnr.Area, BsmtFin.SF.1, BsmtFin.SF.2, Bsmt.Unf.SF,

Total.Bsmt.SF, X1st.Flr.SF, X2nd.Flr.SF, Low.Qual.Fin.SF, Gr.Liv.Area, Garage.Area,

Wood.Deck.SF, Open.Porch.SF, Enclosed.Porch, X3Ssn.Porch, Screen.Porch, Pool.Area,

Misc.Val, Year.Built, Year.Remod.Add, Bsmt.Full.Bath, Bsmt.Half.Bath, Full.Bath,

Half.Bath, Bedroom.AbvGr, Kitchen.AbvGr, TotRms.AbvGrd, Fireplaces, Garage.Yr.Blt,

Garage.Cars, Mo.Sold, Yr.Sold, SalePrice) %>%

mutate(across(!Lot.Frontage, as.numeric))

### creating third variable "SalePrice" as it is the primary variable

SalePrice = ames$SalePrice

###Remove outcome from ames\_numerical

ames\_numerical = ames\_numerical %>%

dplyr::select(., -SalePrice)

###Check Missing Data in Categorical

aggr(ames\_categorical, plot = F)

###convert ames\_categorical to character to replace na values easily

ames\_categorical = ames\_categorical %>%

select(MS.SubClass, MS.Zoning, Street, Alley, Lot.Shape, Land.Contour, Utilities, Lot.Config,

Land.Slope, Neighborhood, Condition.1, Condition.2, Bldg.Type, House.Style,

Overall.Qual, Overall.Cond, Roof.Style, Roof.Matl, Exterior.1st, Exterior.2nd, Mas.Vnr.Type,

Exter.Qual, Exter.Cond, Foundation, Bsmt.Qual, Bsmt.Cond, Bsmt.Exposure, BsmtFin.Type.1,

BsmtFin.Type.2, Heating, Heating.QC, Central.Air, Electrical, Kitchen.Qual, Functional,

Fireplace.Qu, Garage.Type, Garage.Finish, Garage.Qual, Garage.Cond, Paved.Drive,

Pool.QC, Fence, Misc.Feature, Sale.Type, Sale.Condition) %>%

mutate(across(!MS.Zoning, as.character))

###Replace NA where appropriate

ames\_categorical <- ames\_categorical %>%

mutate(Alley = replace\_na(Alley,"No Alley Access")

,Mas.Vnr.Type = replace\_na(Mas.Vnr.Type, "None")

,Bsmt.Qual = replace\_na(Bsmt.Qual,"No Bsmt")

,Bsmt.Cond = replace\_na(Bsmt.Cond,"No Bsmt")

,Bsmt.Exposure = replace\_na(Bsmt.Exposure,"No Bsmt")

,BsmtFin.Type.1 = replace\_na(BsmtFin.Type.1,"No Bsmt")

,BsmtFin.Type.2 = replace\_na(BsmtFin.Type.2,"No Bsmt")

,Electrical = replace\_na(Electrical, "No Electrical")

,Fireplace.Qu = replace\_na(Fireplace.Qu,"No Fireplace")

,Garage.Type = replace\_na(Garage.Type,"No Garage")

,Garage.Finish = replace\_na(Garage.Finish,"No Garage")

,Garage.Qual = replace\_na(Garage.Qual,"No Garage")

,Garage.Cond = replace\_na(Garage.Cond,"No Garage")

,Pool.QC = replace\_na(Pool.QC,"No Pool")

,Fence = replace\_na(Fence,"No Fence")

,Misc.Feature = replace\_na(Misc.Feature,"None"))

### Convert categorical values back to factor

ames\_categorical = ames\_categorical %>%

select(MS.SubClass, MS.Zoning, Street, Alley, Lot.Shape, Land.Contour, Utilities, Lot.Config,

Land.Slope, Neighborhood, Condition.1, Condition.2, Bldg.Type, House.Style,

Overall.Qual, Overall.Cond, Roof.Style, Roof.Matl, Exterior.1st, Exterior.2nd, Mas.Vnr.Type,

Exter.Qual, Exter.Cond, Foundation, Bsmt.Qual, Bsmt.Cond, Bsmt.Exposure, BsmtFin.Type.1,

BsmtFin.Type.2, Heating, Heating.QC, Central.Air, Electrical, Kitchen.Qual, Functional,

Fireplace.Qu, Garage.Type, Garage.Finish, Garage.Qual, Garage.Cond, Paved.Drive,

Pool.QC, Fence, Misc.Feature, Sale.Type, Sale.Condition) %>%

mutate(across(!Street, as.factor))

### again Check Missing Data in Categorical

aggr(ames\_categorical, plot = F)

### running a summary stat again just to have a brief view on categorical data

summary(ames\_categorical)

# Overall Quality

levels(ames$Overall.Qual)

#changing overall quality to character to change one inappropriate factor level

#and then back to factor

ames$Overall.Qual = as.character(ames$Overall.Qual)

ames$Overall.Qual[ames$Overall.Qual == 11] <- NA

ames$Overall.Qual[is.na(ames$Overall.Qual)] <- mode(ames$Overall.Qual)

aggr(ames$Overall.Qual, plot = F)

ames$Overall.Qual = as.factor(ames$Overall.Qual)

### Summary and plots for each column individually just to see if there are any

### outliers and treat them

## Lot Frontage treatment

summary(ames\_numerical$Lot.Frontage)

plot.default(ames\_numerical$Lot.Frontage)

boxplot.default(ames\_numerical$Lot.Frontage)

#removing anything above 150 sq.ft

ames\_numerical$Lot.Frontage[ames\_numerical$Lot.Frontage > 150] <- NA

#replacing them with mean

ames\_numerical$Lot.Frontage[is.na(ames\_numerical$Lot.Frontage)] <- mean.default(ames\_numerical$Lot.Frontage, na.rm = TRUE)

plot(density(ames\_numerical$Lot.Frontage), xlim=c(0, 300)) #a view at skewness and kurtosis of the data

##Lot Area Treatment

summary(ames\_numerical$Lot.Area)

plot.default(ames\_numerical$Lot.Area)

plot(density(ames\_numerical$Lot.Area), xlim=c(0, 2500))

#removing anything above 30000 sq.ft

ames\_numerical$Lot.Area[ames\_numerical$Lot.Area > 30000] <- NA

#removing anything below 10

ames\_numerical$Lot.Area[ames\_numerical$Lot.Area < 1000] <- NA

aggr(ames\_numerical$Lot.Area, plot = F)

#replacing the values with the mean as the data

ames\_numerical$Lot.Area[is.na(ames\_numerical$Lot.Area)] <- mean(ames\_numerical$Lot.Area, na.rm = TRUE)

### Mas.Vnr.Area treatment

summary(ames\_numerical$Mas.Vnr.Area)

plot.default(ames\_numerical$Mas.Vnr.Area)

#removing anything above 800sq ft

ames\_numerical$Mas.Vnr.Area[ames\_numerical$Mas.Vnr.Area > 800] <- NA

#replacing NA with 0 as na means no masonry veneer area

ames\_numerical$Mas.Vnr.Area[is.na(ames\_numerical$Mas.Vnr.Area)] <- mean(ames\_numerical$Mas.Vnr.Area, na.rm = TRUE)

aggr(ames\_numerical$Mas.Vnr.Area, plot = F)

## treating the remaining na columns which should be 0 in place of na

ames\_numerical = ames\_numerical %>%

mutate(.,BsmtFin.SF.1 = replace\_na(BsmtFin.SF.1,0),

BsmtFin.SF.2 = replace\_na(BsmtFin.SF.2,0),

Bsmt.Unf.SF = replace\_na(Bsmt.Unf.SF,0),

Total.Bsmt.SF = replace\_na(Total.Bsmt.SF,0),

Bsmt.Full.Bath = replace\_na(Bsmt.Full.Bath,0),

Bsmt.Half.Bath = replace\_na(Bsmt.Half.Bath,0),

Garage.Cars = replace\_na(Garage.Cars,0),

Garage.Area = replace\_na(Garage.Area,0))

##If no garage, year build should be put into garage year

ames\_numerical = ames\_numerical %>%

mutate(.,Garage.Yr.Blt = replace\_na(Year.Built))

#summary of numerical columns to see if any column contains outliers

summary(ames\_numerical)

###checking outliers in X1st.Flr.SF

summary(ames\_numerical$X1st.Flr.SF)

plot.default(ames\_numerical$X1st.Flr.SF)

#removing anything above 2500 and replacing with mean

ames\_numerical$X1st.Flr.SF[ames\_numerical$X1st.Flr.SF > 2500] <- NA

ames\_numerical$X1st.Flr.SF[is.na(ames\_numerical$X1st.Flr.SF)] <- mean(ames\_numerical$X1st.Flr.SF, na.rm = TRUE)

aggr(ames\_numerical$X1st.Flr.SF, plot = F)

###checking outliers in X2nd.Flr.SF

summary(ames\_numerical$X2nd.Flr.SF)

plot.default(ames\_numerical$X2nd.Flr.SF)

#removing anything above 1500 and replacing with mean

ames\_numerical$X2nd.Flr.SF[ames\_numerical$X2nd.Flr.SF > 1500] <- NA

ames\_numerical$X2nd.Flr.SF[is.na(ames\_numerical$X2nd.Flr.SF)] <- mean(ames\_numerical$X1st.Flr.SF, na.rm = TRUE)

###checking outliers in Low.Qual.Fin.SF

summary(ames\_numerical$Low.Qual.Fin.SF)

plot.default(ames\_numerical$Low.Qual.Fin.SF)

#removing anything above 600 and replacing with mean

ames\_numerical$Low.Qual.Fin.SF[ames\_numerical$Low.Qual.Fin.SF > 600] <- NA

ames\_numerical$Low.Qual.Fin.SF[is.na(ames\_numerical$Low.Qual.Fin.SF)] <- mean(ames\_numerical$Low.Qual.Fin.SF, na.rm = TRUE)

###checking outliers in Gr.Liv.Area

summary(ames\_numerical$Gr.Liv.Area)

plot.default(ames\_numerical$Gr.Liv.Area)

#removing anything above 3000 and replacing with mean

ames\_numerical$Gr.Liv.Area[ames\_numerical$Gr.Liv.Area > 3000] <- NA

ames\_numerical$Gr.Liv.Area[is.na(ames\_numerical$Gr.Liv.Area)] <- mean(ames\_numerical$Gr.Liv.Area, na.rm = TRUE)

aggr(ames\_numerical$Gr.Liv.Area, plot = F)

###checking outliers in Garage.Area

summary(ames\_numerical$Garage.Area)

plot.default(ames\_numerical$Garage.Area)

#removing anything above 1000 and replacing with mean

ames\_numerical$Garage.Area[ames\_numerical$Garage.Area > 1000] <- NA

ames\_numerical$Garage.Area[is.na(ames\_numerical$Garage.Area)] <- mean(ames\_numerical$Garage.Area, na.rm = TRUE)

###checking outliers in Wood.Deck.SF

summary(ames\_numerical$Wood.Deck.SF)

plot.default(ames\_numerical$Wood.Deck.SF)

#removing anything above 550 and replacing with mean

ames\_numerical$Wood.Deck.SF[ames\_numerical$Wood.Deck.SF > 550] <- NA

ames\_numerical$Wood.Deck.SF[is.na(ames\_numerical$Wood.Deck.SF)] <- mean(ames\_numerical$Wood.Deck.SF, na.rm = TRUE)

aggr(ames\_numerical$Wood.Deck.SF, plot = F)

###checking outliers in Open.Porch.SF

summary(ames\_numerical$Open.Porch.SF)

plot.default(ames\_numerical$Open.Porch.SF)

#removing anything above 300 and replacing with mean

ames\_numerical$Open.Porch.SF[ames\_numerical$Open.Porch.SF > 300] <- NA

ames\_numerical$Open.Porch.SF[is.na(ames\_numerical$Open.Porch.SF)] <- mean(ames\_numerical$Open.Porch.SF, na.rm = TRUE)

aggr(ames\_numerical$Open.Porch.SF, plot = F)

###checking outliers in Enclosed.Porch

summary(ames\_numerical$Enclosed.Porch)

plot.default(ames\_numerical$Enclosed.Porch)

#removing anything above 300 and replacing with mean

ames\_numerical$Enclosed.Porch[ames\_numerical$Enclosed.Porch > 300] <- NA

ames\_numerical$Enclosed.Porch[is.na(ames\_numerical$Enclosed.Porch)] <- mean(ames\_numerical$Enclosed.Porch, na.rm = TRUE)

aggr(ames\_numerical$Enclosed.Porch, plot = F)

###checking outliers in X3Ssn.Porch

summary(ames\_numerical$X3Ssn.Porch)

plot.default(ames\_numerical$X3Ssn.Porch)

#removing anything above 400 and replacing with mean

ames\_numerical$X3Ssn.Porch[ames\_numerical$X3Ssn.Porch > 400] <- NA

ames\_numerical$X3Ssn.Porch[is.na(ames\_numerical$X3Ssn.Porch)] <- mean(ames\_numerical$X3Ssn.Porch, na.rm = TRUE)

aggr(ames\_numerical$X3Ssn.Porch, plot = F)

###checking outliers in Screen.Porch

summary(ames\_numerical$Screen.Porch)

plot.default(ames\_numerical$Screen.Porch)

#removing anything above 300 and replacing with mean

ames\_numerical$Screen.Porch[ames\_numerical$Screen.Porch > 300] <- NA

ames\_numerical$Screen.Porch[is.na(ames\_numerical$Screen.Porch)] <- mean(ames\_numerical$Screen.Porch, na.rm = TRUE)

aggr(ames\_numerical$Screen.Porch, plot = F)

###checking outliers in Misc.Val

summary(ames\_numerical$Misc.Val)

plot.default(ames\_numerical$Misc.Val)

#removing anything above 5000 and replacing with mean

ames\_numerical$Misc.Val[ames\_numerical$Misc.Val > 5000] <- NA

ames\_numerical$Misc.Val[is.na(ames\_numerical$Misc.Val)] <- mean(ames\_numerical$Misc.Val, na.rm = TRUE)

###checking outliers in Year.Remod.Add

summary(ames\_numerical$Year.Remod.Add)

plot.default(ames\_numerical$Year.Remod.Add)

#removing anything above 2021 and replacing with mode

ames\_numerical$Year.Remod.Add[ames\_numerical$Year.Remod.Add > 2021] <- NA

ames\_numerical$Year.Remod.Add[is.na(ames\_numerical$Year.Remod.Add)] <- mode(ames\_numerical$Year.Remod.Add)

###checking outliers in Bedroom.AbvGr

summary(ames\_numerical$Bedroom.AbvGr)

plot.default(ames\_numerical$Bedroom.AbvGr)

#removing anything above 6 bedrooms and replacing with mode

ames\_numerical$Bedroom.AbvGr[ames\_numerical$Bedroom.AbvGr > 6] <- NA

ames\_numerical$Bedroom.AbvGr[is.na(ames\_numerical$Bedroom.AbvGr)] <- mode(ames\_numerical$Bedroom.AbvGr)

aggr(ames\_numerical$Bedroom.AbvGr, plot = F)

###checking outliers in Kitchen.AbvGr

summary(ames\_numerical$Kitchen.AbvGr)

plot.default(ames\_numerical$Kitchen.AbvGr)

#removing anything above 2 kitchen and replacing with mode

ames\_numerical$Kitchen.AbvGr[ames\_numerical$Kitchen.AbvGr > 2] <- NA

ames\_numerical$Kitchen.AbvGr[is.na(ames\_numerical$Kitchen.AbvGr)] <- mode(ames\_numerical$Kitchen.AbvGr)

###checking outliers in TotRms.AbvGrd

summary(ames\_numerical$TotRms.AbvGrd)

plot.default(ames\_numerical$TotRms.AbvGrd)

#removing anything above 12 rooms and replacing with mode

ames\_numerical$TotRms.AbvGrd[ames\_numerical$TotRms.AbvGrd > 12] <- NA

ames\_numerical$TotRms.AbvGrd[is.na(ames\_numerical$TotRms.AbvGrd)] <- mode(ames\_numerical$TotRms.AbvGrd)

###checking outliers in Fireplaces

summary(ames\_numerical$Fireplaces)

plot.default(ames\_numerical$Fireplaces)

#removing anything above 3 fireplaces and replacing with mode

ames\_numerical$Fireplaces[ames\_numerical$Fireplaces > 2.5] <- NA

ames\_numerical$Fireplaces[is.na(ames\_numerical$Fireplaces)] <- mode(ames\_numerical$Fireplaces)

aggr(ames\_numerical$Fireplaces, plot = F)

###checking outliers in Garage.Yr.Blt

summary(ames\_numerical$Garage.Yr.Blt)

plot.default(ames\_numerical$Garage.Yr.Blt)

#removing anything below 1800 year built and replacing with mode

ames\_numerical$Garage.Yr.Blt[ames\_numerical$Garage.Yr.Blt < 1800] <- NA

ames\_numerical$Garage.Yr.Blt[is.na(ames\_numerical$Garage.Yr.Blt)] <- mode(ames\_numerical$Garage.Yr.Blt)

###checking outliers in Garage.Cars

summary(ames\_numerical$Garage.Cars)

plot.default(ames\_numerical$Garage.Cars)

#removing anything above 4 car space and replacing with mode

ames\_numerical$Garage.Cars[ames\_numerical$Garage.Cars > 4] <- NA

ames\_numerical$Garage.Cars[is.na(ames\_numerical$Garage.Cars)] <- mode(ames\_numerical$Garage.Cars)

###Outcome treatment that is our salesprice

summary(SalePrice)

plot.default(SalePrice) #scipen = 10

#removing anything above 3000000 and replacing with mean

SalePrice[SalePrice > 600000] <- NA

SalePrice[SalePrice < 50000] <- NA

SalePrice[is.na(SalePrice)] <- mean(SalePrice, na.rm = TRUE)

aggr(SalePrice, plot = F)

##Slap the three back together

ames = cbind(ames\_categorical, ames\_numerical, SalePrice)

## running the pca test

ames\_2 = dummy.data.frame(ames)

dim(ames\_2)

ames\_3 = scale(ames\_2)

prc\_ames = prcomp(ames\_3)

#pca plot

biplot(prc\_ames, scale = 0)

biplot(prc\_ames, scale = 0, expand = 30)

summary(prc\_ames)

### Changins factor levels which will be required for visualizations

# MS.Zoning

class(ames$MS.Zoning)

levels(ames$MS.Zoning)

levels(ames$MS.Zoning)[levels(ames$MS.Zoning) == "A (agr)"] <- "Agriculture"

levels(ames$MS.Zoning)[levels(ames$MS.Zoning) == "C (all)"] <- "Commercial"

levels(ames$MS.Zoning)[levels(ames$MS.Zoning) == "FV"] <- "Floating Village Residenial"

levels(ames$MS.Zoning)[levels(ames$MS.Zoning) == "I (all)"] <- "Industrial"

levels(ames$MS.Zoning)[levels(ames$MS.Zoning) == "RH"] <- "Residential (High Density)"

levels(ames$MS.Zoning)[levels(ames$MS.Zoning) == "RL"] <- "Residential (Low Density)"

levels(ames$MS.Zoning)[levels(ames$MS.Zoning) == "RP"] <- "Residential (Low Density park)"

levels(ames$MS.Zoning)[levels(ames$MS.Zoning) == "RM"] <- "Residential (Medium Density)"

# Street

levels(ames$Street)

levels(ames$Street)[levels(ames$Street) == "Grvl"] <- "Gravel"

levels(ames$Street)[levels(ames$Street) == "Pave"] <- "Paved"

# Alley

levels(ames$Alley)

levels(ames$Alley)[levels(ames$Alley) == "Grvl"] <- "Gravel"

levels(ames$Alley)[levels(ames$Alley) == "Pave"] <- "Paved"

ames$Alley <- factor(ames$Alley,levels = c("Gravel", "Paved", "No Alley Access"))

# Lot Shape

levels(ames$Lot.Shape)

levels(ames$Lot.Shape) <- c("Slightly Irregular", "Moderately Irregular", "Irregular", "Regular")

ames$Lot.Shape <- factor(ames$Lot.Shape, levels = c("Regular", "Slightly Irregular", "Moderately Irregular", "Irregular"))

# Land Contour

levels(ames$Land.Contour)

levels(ames$Land.Contour) <- c("Banked", "Hillside", "Depression", "Near Flat")

ames$Land.Contour <- factor(ames$Land.Contour, levels = c("Near Flat", "Banked", "Hillside", "Depression"))

# Lot Config

levels(ames$Lot.Config)

levels(ames$Lot.Config) <- c("Corner Lot", "Cul-De-Sac", "Frontage on 2 sides"

, "Frontage on 3 sides", "Inside Lot")

ames$Lot.Config <- factor(ames$Lot.Config, levels = c("Inside Lot",

"Corner Lot", "Cul-De-Sac",

"Frontage on 2 sides", "Frontage on 3 sides"))

# Overall Quality

levels(ames$Overall.Qual)

ames$Overall.Qual <- factor(ames$Overall.Qual, levels = c("1", "2", "3", "4", "5", "6", "7",

"8", "9", "10"))

levels(ames$Overall.Qual) <- c("Very Poor", "Poor", "Fair", "Below Average", "Average",

"Above Average", "Good", "Very Good", "Excellent", "Very Excellent")

# overall condition

levels(ames$Overall.Cond)

levels(ames$Overall.Cond) <- c("Very Poor", "Poor", "Fair", "Below Average", "Average",

"Above Average", "Good", "Very Good", "Excellent")

# Garage Quality

levels(ames$Garage.Qual)

ames$Garage.Qual <- factor(ames$Garage.Qual, levels = c("Ex", "Gd", "TA", "Fa", "Po", "No Garage"))

levels(ames$Garage.Qual) <- c("Excellent", "Good", "Typical/Average", "Fair", "Poor", "No Garage")

# Garage Condition

levels(ames$Garage.Cond)

ames$Garage.Cond <- factor(ames$Garage.Cond, levels = c("Ex", "Gd", "TA", "Fa", "Po", "No Garage"))

levels(ames$Garage.Cond) <- c("Excellent", "Good", "Typical/Average", "Fair", "Poor", "No Garage")

# Sale type

levels(ames$Sale.Condition)

ames$Sale.Condition <- factor(ames$Sale.Condition, levels = c("Normal", "Abnorml",

"AdjLand", "Alloca", "Family", "Partial"))

levels(ames$Sale.Condition) <- c("Normal Sale", "Abnormal Sale", "Adjoining Purchase",

"Allocation", "Sale B/W Family", "Incomplete Home")

####

# Visualizations

####

#A histogram of SalePrice just to see how the data is distributed

ames %>%

ggplot(aes(SalePrice)) +

geom\_histogram(binwidth = 100000, fill = "#97B3C6") +

theme\_bw()

# Visualization - 1

ames %>%

ggplot(aes(Lot.Area, SalePrice))+

geom\_point(aes(color = Lot.Shape, shape = Lot.Shape, size = 2, alpha = 0.5))+

geom\_smooth(method = "lm")+

labs(title = "Sales Price in respect with Area and Shape",

x = "Lot Area (sqft)",

y = "Sale Price($)",

caption="Ames Housing Data")+

coord\_flip()+

theme\_stata()+scale\_fill\_aaas()

#doing a facet wrap to understand it in a better way

ames %>%

ggplot(aes(Lot.Area, SalePrice))+

geom\_point(aes(color = Lot.Shape, shape = Lot.Shape, size = 2, alpha = 0.5))+

geom\_smooth(method = "lm", se = FALSE)+

facet\_wrap(Lot.Shape~.)+

labs(title = "Sales Price in respect with Area and Shape",

x = "Lot Area (sqft)",

y = "Sale Price($)",

caption="Ames Housing Data")+

coord\_flip()

theme\_stata()

#visualization 2

ames %>%

ggplot(aes(x = Overall.Qual, SalePrice, fill = Overall.Cond))+

geom\_bar(stat = "identity",position = "dodge")+

labs(title="Sales Price with respect to overall Condition & Quality",

x="Overall Quality",y="Sales Price ($)",

caption="Ames Housing Data",fill="Overall Condition")+

theme\_classic()+scale\_fill\_jco()

# visualization 3

ames %>%

ggplot(aes(SalePrice, Lot.Config))+

geom\_density\_ridges(aes(fill = Land.Contour), alpha = 0.3)+

labs(title="Sales Price with respect to Lot Configuration & Land Contour",

x="Sales Price ($)",y="Lot Configuration",

caption="Ames Housing Data",fill = "Land Contour")+

theme\_bw()+

scale\_fill\_excel\_new()

#doing a facet wrap to understand it in a better way

ames %>%

ggplot(aes(SalePrice, Lot.Config))+

geom\_density\_ridges(aes(fill = Land.Contour), alpha = 0.3)+

facet\_wrap(Land.Contour~.)+

labs(title="Sales Price with respect to Lot Configuration & Land Contour",

x="Sales Price ($)",y="Lot Configuration",

caption="Ames Housing Data",fill = "Land Contour")+

theme\_bw()+

scale\_fill\_excel\_new()

#visualization 4

ames %>%

ggplot(aes(x = Bedroom.AbvGr, y = SalePrice))+

stat\_density\_2d(aes(fill = ..level..), geom = "polygon", color="white")+

scale\_fill\_distiller(palette = "Spectral", direction=-1)+

labs(title="Sales Price with respect to No of Bedrooms",

x="No of Bedrooms",y="Sales Price '$'",

caption="Ames Housing Data")+

theme\_bw()

#visualization 5

ames %>%

ggplot(aes(Gr.Liv.Area, SalePrice))+

geom\_smooth(aes(color = Garage.Qual), method = lm, se = FALSE, fullrange = TRUE) +

geom\_point(aes(color = Garage.Qual), alpha = 0.5)+

labs(title = "Effect of Graded Living area with respect to Garage Quality on Sales Price",

x = " Graded living Area", y = "Sales Price", fill = " Garage Quality",

caption = "Ames Housing Data") +

theme\_fivethirtyeight() +

scale\_fill\_pander()

#facet\_grid of the above plot

ames %>%

ggplot(aes(Gr.Liv.Area, SalePrice))+

geom\_smooth(aes(color = Garage.Qual), method = lm, se = FALSE, fullrange = TRUE) +

geom\_point(aes(color = Garage.Qual), alpha = 0.5)+

labs(title = "Effect of Graded Living area with respect to Garage Quality on Sales Price",

x = " Graded living Area", y = "Sales Price", fill = " Garage Quality",

caption = "Ames Housing Data") +

facet\_grid(Garage.Qual~.) +

theme\_fivethirtyeight() +

scale\_fill\_pander()

### Visualisation 6 (animation)

#plotting the ggplot for animation

graph = ames %>%

ggplot(aes(Mas.Vnr.Area, SalePrice, color = Mas.Vnr.Type)) +

geom\_point(alpha = 0.7) +

theme\_fivethirtyeight() +

scale\_size(range=c(2,12), guide="none") +

scale\_x\_log10() +

labs(title = "Sales price change over the years with respect to masonry veneer area and type",

x = "masonry veneer area ( sqft. )",

y = "Sales Price",

caption = "Ames Housing Data",

color = "masonry veneer type") +

theme(axis.title = element\_text(),

text = element\_text(family = "Rubik"),

legend.text=element\_text(size=10))

# Setting up the animation on which it would run

graph1.animation = graph +

transition\_time(as.integer(ames$Yr.Sold)) +

labs(subtitle = "Yr.Sold: {frame\_time}") +

shadow\_wake(wake\_length = 0.1)

# running and renerring the animation

animate(plot = graph1.animation, height = 500, width = 800, fps = 45, duration = 20,

end\_pause = 70, res = 100, renderer = gifski\_renderer())