UNDERSTANDING THE HOUSING MARKET THROUGH MACHINE LEARNING TECHNIQUES

Registration ID - 2112601

University of Essex

MA335 - Modelling Experimental and Observational Data

UNDERSTANDING THE HOUSING MARKET THROUGH MACHINE LEARNING TECHNIQUES

WORD COUNT: 2098

ABSRACT

The aim of this report is to analyse the provided housing dataset using machine learning techniques to deride insights and patterns within the housing market. The report analyses the dataset through regression, clustering and classification techniques to achieve these aims. Notably, the report finds that our regression model was able to account for 79% of sale price variance (adjusted R-squared of 0.79) and using K-means and Hierarchical clustering, were able to successfully group houses based on their features.

TABLE OF CONTENTS

INTRODUCTION

Asking a home buyer what their dream house would entail, it is unlikely they would begin with the height of the basement ceiling. However this data set has shown that more than simply the number of bedrooms influences the price negotiations. The task of this report was to analyse the housing data set provided by the estate agency to better understand the housing market. To do this, the report uses regression techniques to understand whether certain variables can predict house prices and pattern identification techniques such as clustering and classification to categorise houses into different classes based on their features.

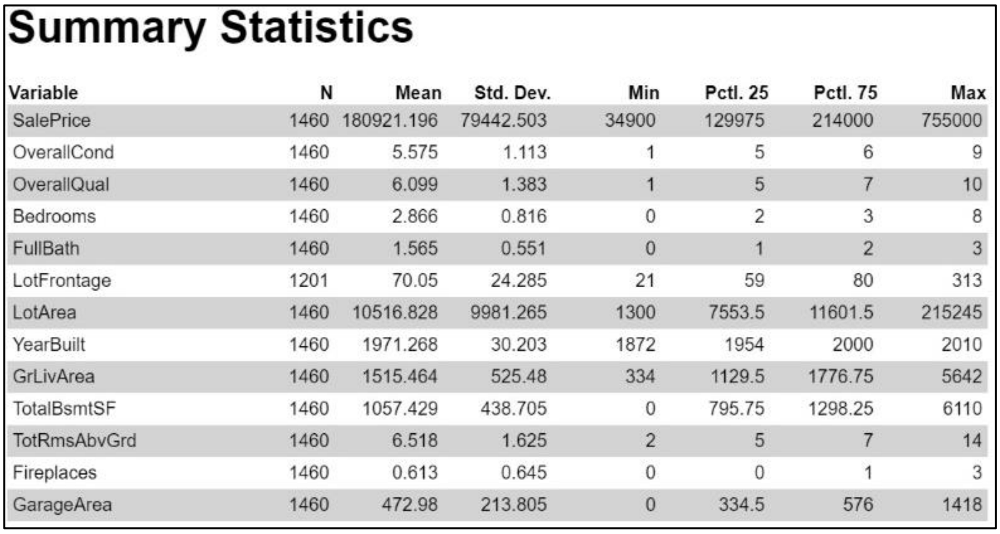
The report is structured as follows. A preliminary analysis is first conducted using numerical and graphical descriptive statistics. The main analysis is then undertaken, this includes regression, clustering and classification algorithms to understand the relationship between housing features and housing sale prices. The conclusion then summaries the findings of the report.

PRELIMINARY ANALYSIS

The preliminary analysis section provides a thorough EDA of the data set and its variables. In this section, the data size and structure is explored, the summary statistics of important numeric variables are provided as well as univariate and bivariate analysis of important variables.

The housing data set comprises 1460 observations from 51 variables, of which the last one is the response variable (SalePrice). The variables consist of integer and character variables, however most of the character variables are actually ordinal factors. Before running the clustering and classification algorithms they will require some cleaning and feature engineering.

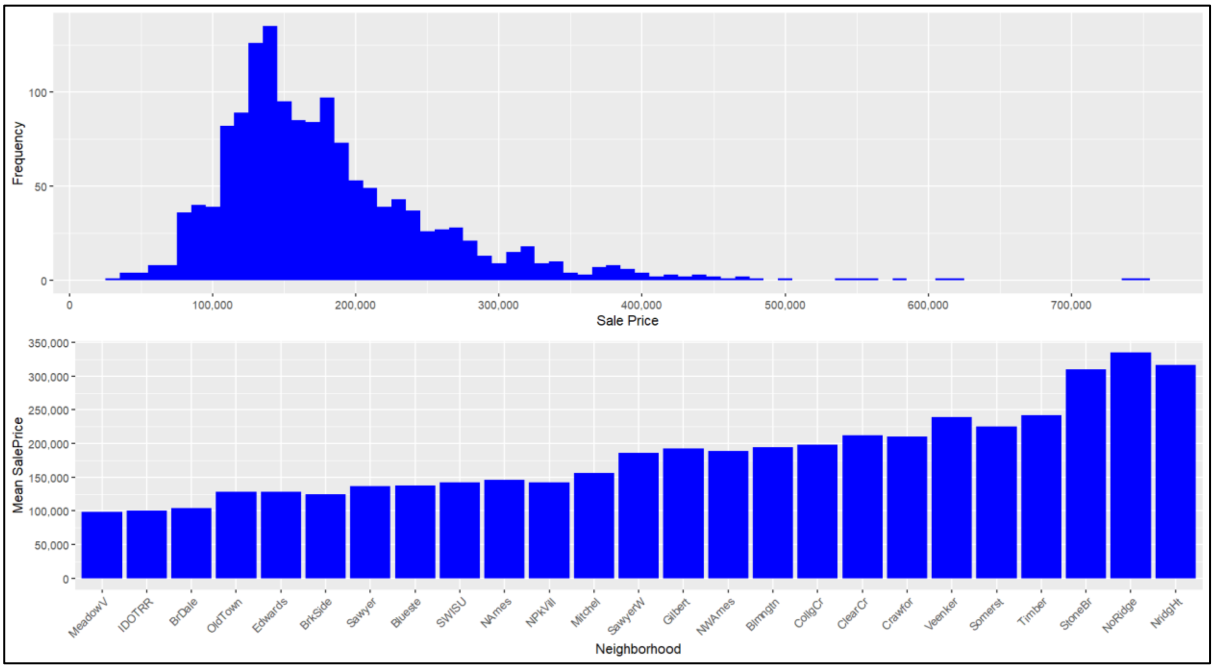
The summary statistic table below contains 13 variables out of the data set (including the dependent variable ‘SalePrice’) that are useful for gauging an idea of the range of houses within this dataset. At the top of table we see the variable SalePrice with a mean of $180,921.20, with a minimum recorded value of $34,900 and maximum of $755,000. OverallCond (the overall condition of the house) on average has a rating of 5, meaning average condition. OverallQual (the overall material and finish of house) on average had a rating of 6 meaning above average quality. The average lot size in square feet (LotArea variable) was 10,516 feet, however looking at the maximum quartile you can see there are some houses with disproportionately larger lot areas (215,245). This seems to be an outlier and possibly an error within the data when considering that the 75 percentile value is 11,601 feet. These outliers will need to be revisited and possibly removed before modelling. Finally in regards to the summary statistics table below, we can observe that on average the houses have between 2 and 3 bedrooms with 1 to 2 bathrooms and a basement (if they have one) of 1057 square feet.



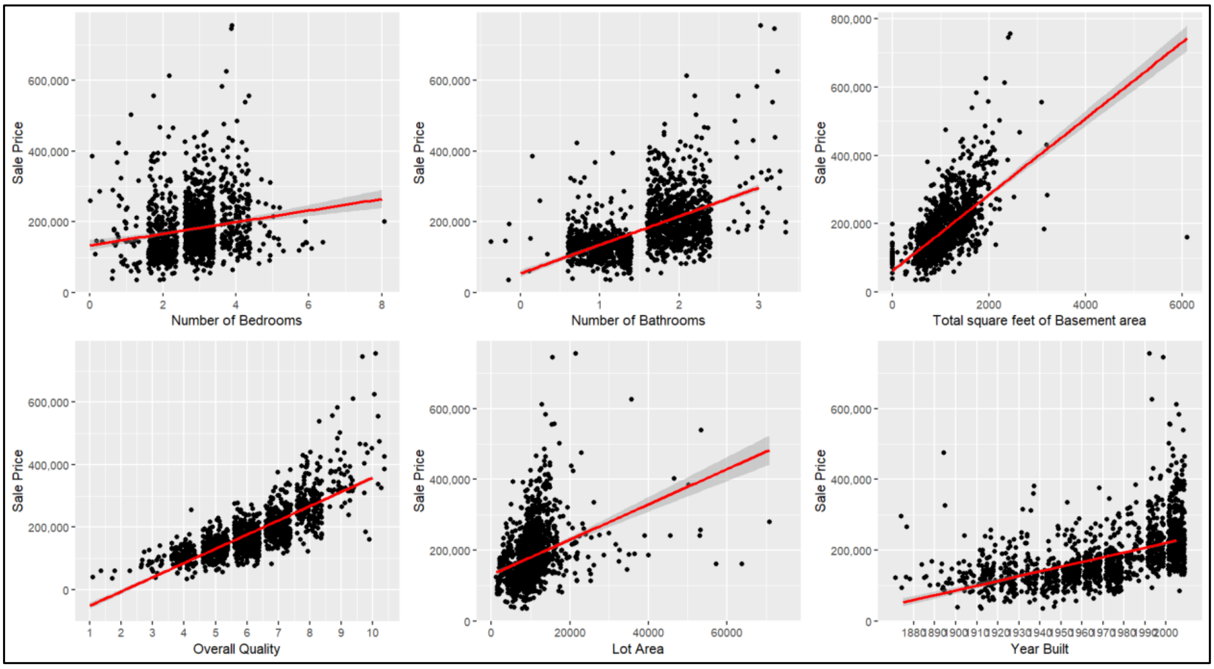
Summary Statistics

Variable	N	Mean	Std. Dev.	Min	Pctl. 25	Pctl. 75	Max
SalePrice	1460	180921.196	79442.503	34900	129975	214000	755000
OverallCond	1460	5.575	1.113	1	5	6	9
OverallQual	1460	6.099	1.383	1	5	7	10
Bedrooms	1460	2.866	0.816	0	2	3	8
FullBath	1460	1.565	0.551	0	1	2	3
LotFrontage	1201	70.05	24.285	21	59	80	313
LotArea	1460	10516.828	9981.265	1300	7553.5	11601.5	215245
YearBuilt	1460	1971.268	30.203	1872	1954	2000	2010
GrLivArea	1460	1515.464	525.48	334	1129.5	1776.75	5642
TotalBsmtSF	1460	1057.429	438.705	0	795.75	1298.25	6110
TotRmsAbvGrd	1460	6.518	1.625	2	5	7	14
Fireplaces	1460	0.613	0.645	0	0	1	3
GarageArea	1460	472.98	213.805	0	334.5	576	1418

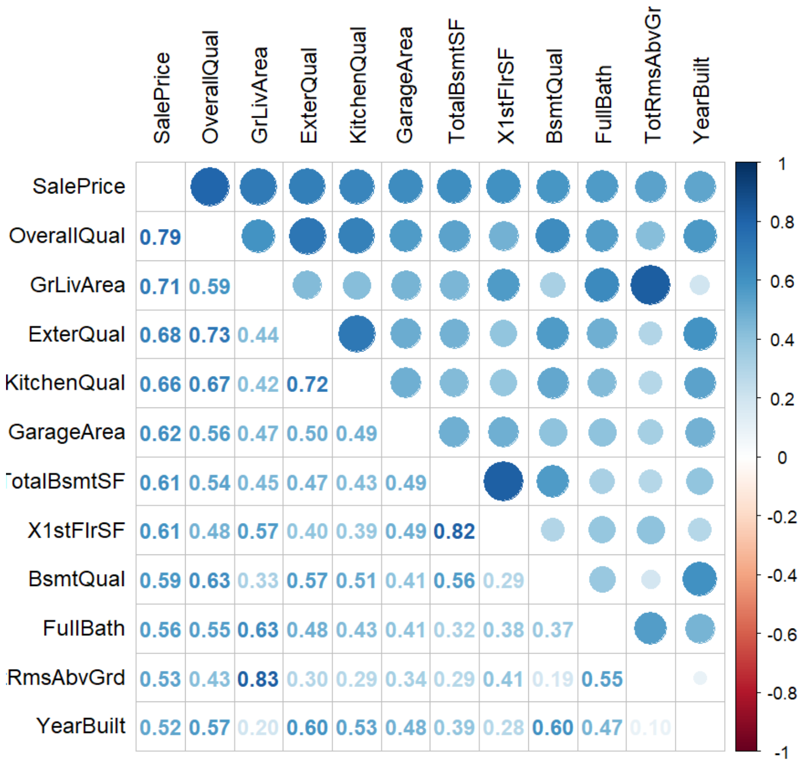
Figure 1 (seen below) combines two graphs which help shed light into the response variable; SalePrice. As seen in the top graph within figure 1 and further reflected from the summary statistics seen earlier, the bulk of house prices sit between $100,000-$200,000. This means that the response variable is right



skewed as the majority of instances occupy the left hand side of distribution. This is perhaps unsurprising since it is not the majority of people who are able to purchase expensive homes. The second graph in figure 1 displays the mean sale price among each neighbourhood. Although not completely conclusive, one can start to speculate on whether a neighbourhood is poor or rich relative to it Mean SalePrice. After all, it is more likely for more expensive houses to be in more expensive neighbourhoods. From the graph we can observe that the three most expensive neighbourhoods are Stone Brook, Northridge and Northridge Heights.



One of the tasks the estate agency has set is to run regression models to predict house prices. As a result it makes sense in the exploratory analysis phase to plot some of the independent variables against Sale Price and to insert a regression line through them. This is what Figure 2 (seen above) does. The plots not only provide a visual representation of the summary statistics seen earlier but demonstrates which variables may be of interest when conducting the linear regression in the next section. From Figure 2 we can see that all variables included (number of bedrooms, number of bathrooms, basement total square feet, overall quality, lot area, year built) have a positive correlation with the predictor variable SalePrice. Overall Quality seems to have the strongest correlation to SalePrice. This make sense - if the overall quality of the house, the materials used and the finish are to a high standard, naturally the price you pay is more. The ‘Year Built’ graph in the bottom right of figure 2 demonstrates the appeal for newer houses. As the age of house increases, the value of it decreases. The amount of bedrooms, bathrooms, basement area and lot area all to have positive correlations, however people seem to be more interested in how many bathrooms a house has as opposed to bedroom. What figure 2 does not show is whether there is any collinearity between variables. While on the face of it many of the variables in figure 2 seem like great predictors for house prices, there's likely collinearity between them. The size of your basement is constrained by the size of your lot, and perhaps so to an extent is the number of bedrooms and bathrooms. To rectify this a correlation matrix of all the numeric variables were plotted in order to see not only the variables which correlate with SalePrice but the predictor variables which have collinearity with each other.



MAIN ANALYSIS

Linear Regression

In order to predict house prices using linear regression, two models were created. Model1 (the full model) used all 30 numeric variables. Analysing the model1 output then lead to the final model which had a total of only 9 variables but had the same predictive accuracy of the full model ($Adjusted\ R^2$: 0.8196). Within model2, all the coefficients and the intercept are statistically significant with a p-value very close to 0. Therefore there is strong evidence that neighbourhood, above ground living area (square feet), overall quality, overall

```
Call: (model1)
lm(formula = SalePrice ~ ., data = numericVars)

Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 34590 on 1430 degrees of
freedom, Multiple R-squared:  0.8142, Adjusted R-
squared:  0.8104    F-statistic: 216.1 on 29 and 1430
DF,  p-value: < 2.2e-16
```

```
Call: (model2)
lm(formula = SalePrice ~ Neighborhood + GrLivArea +
OverallQual + OverallCond + GarageArea + TotalBsmtSF +
BsmtQual + LotArea + Fireplaces)

Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 34010 on 1427 degrees of freedom
Multiple R-squared:  0.8208, Adjusted R-squared:  0.8168
F-statistic: 204.2 on 32 and 1427 DF,  p-value: < 2.2e-16
```

condition, garage space, total basement size, basement height, lot area and the number of fireplaces are important factors in predicting house prices. The F-statistic in model2 is 204.2 which is slightly less than model1 at 216.1 suggesting that both models seem to fit the data similarly. As

mentioned previously the R^2 remained the same at 81%.. Comparing the two models with an

ANOVA table (F test) shows that the null hypothesis can be rejected at the significance level closing near to 0% that the full model fits the data better than the reduced model. This means that the features in the reduced model are more important in predicting house prices. Furthermore the Variance Inflation Factor (VIF) for each variable were checked. 2 out of 25 neighbourhoods indicated a degree of multicollinearity (VIF >10), however the rest of the neighbourhoods and the other model variables did not indicate multicollinearity (VIF <5).

```
Analysis of Variance Table

Model 1: SalePrice ~ Id + LotFrontage + LotArea + Street + OverallQual + OverallCond + YearBuilt
+ MasVnrArea + ExterQual + ExterCond + BsmtQual + BsmtCond + TotalBsmtSF + X1stFlrSF +
X2ndFlrSF + LowQualFinSF + GrLivArea + FullBath + Bedrooms + KitchenAbvGr + KitchenQual +
TotRmsAbvGrd + Functional + Fireplaces + GarageArea + GarageCond + PavedDrive + PoolArea +
PoolQC + MiscVal

Model 2: SalePrice ~ GrLivArea + OverallQual + OverallCond + GarageArea + TotalBsmtSF +
BsmtQual + LotArea + Fireplaces

  Res.Df        RSS   Df    Sum of Sq       F      Pr(>F)

1   1430  1.7109e+12

2   1451  2.0824e+12  -21   -3.7148e+11  14.785   < 2.2e-16 ***

Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

To briefly summarise this section – the task was to employ linear regression in order to predict house prices. The final model demonstrates that the neighbourhood, above ground living area (square feet), overall quality, overall condition, garage space, total basement size, basement height, lot area and the number of fireplaces are important factors in predicting house prices and can do so to a high degree of accuracy.

Multi-class Classification

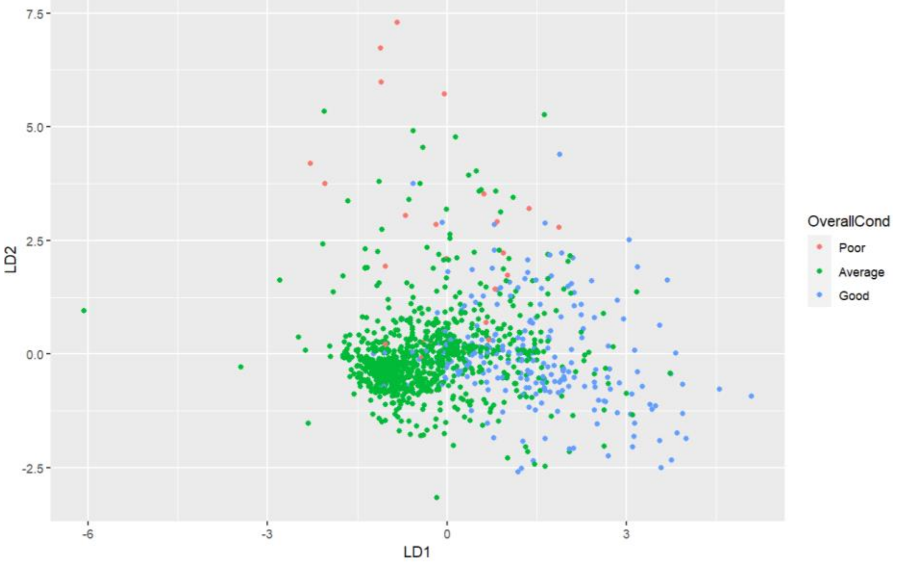
The next task was to transform the Overall Condition variable (OverallCond) into a categorical variable with 3 levels and then implement logistic regression, quadratic discriminant analysis (QDA) and linear discriminant analysis (LDA). Which level to be assigned depends on the original rating the OverCond variable had. ‘Poor’ is assigned if the overall condition is between 1 and 3, ‘Average’ if between 4 and 6 and ‘Good’ if the overall condition is between 7 and 10.

Logistic Regression

The logistic regression to be used in this case is multinomial logistic regression. This is because we are not testing for a binary outcome like regular logistic regression but in fact 3 classes (Poor, Average & Good). The data was split into training and test datasets (80% for training and 20% for testing). The multinomial logistic regression was successful in categorising the houses into the 3 classes and has a successful classification rate of 81%. The residual deviance of the model was 917.032 and the AIC was 1025.032.

Linear Discriminant Analysis

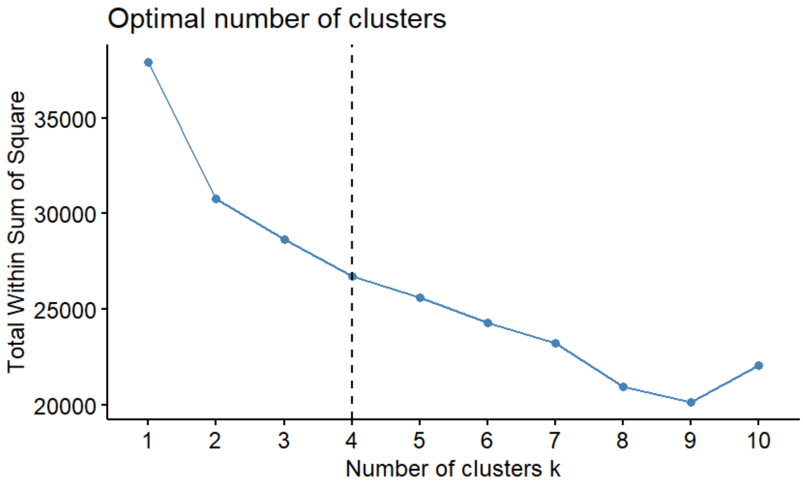
For the LDA model, the same split training and testing data was used from the logistic regression model but was transformed to be centred and scaled. Again the model performed well achieving a successful classification prediction of 79%. Below is a visualisation of the LDA model accurately classifying the houses bases on their overall condition.



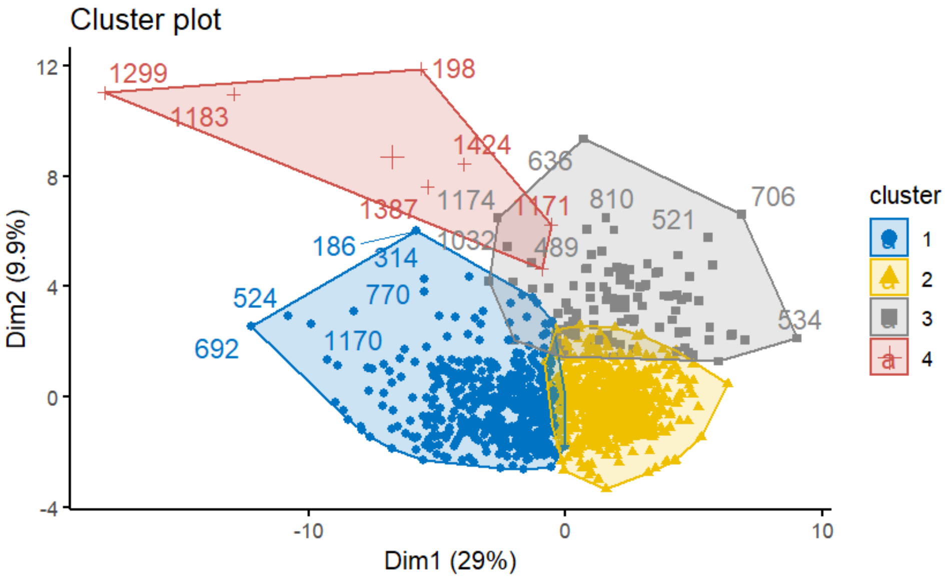
Clustering

K-Means Clustering

The objective of *k*-means clustering within this report was to cluster similar houses together and then comment on the clustering in relation to the bedroom variable. The objective is to minimise intra-cluster variation and maximise inter-cluster variation. In order to determine the correct amount of clusters to use, the elbow method was employed to see how the Total Within Sum of Squares would reduce in relation to the number of k clusters. Although selecting the optimal amount of clusters isn't a science, the graph below helps to visualise where the most optimal cluster number may lie. From the graph it can be seen that 4 *k* clusters were selected as being the optimal amount and therefore selected as the number going forward.



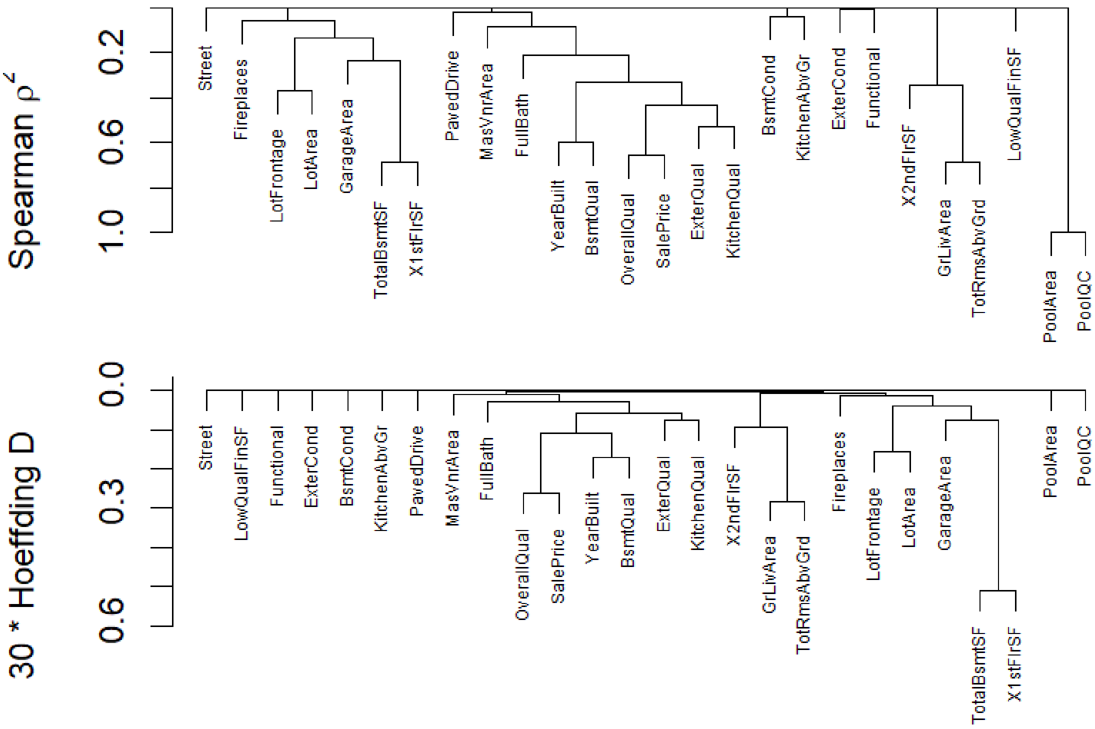
The rationale behind this was as follows. Although the Total WSS still decreases after 4 clusters, it decreases most precipitously for the first 4 before levelling out somewhat. 4 clusters as opposed to 7 was chosen as over-clustering wanted to be avoided



Variable clustering

For this variable clustering, two methods were used (Spearman & Hoeffding) to cluster variables. First looking at the Spearman Tree Plot we can see that PoolArea and PoolQC are most highly correlated. This make sense given that most houses did not have pools and so the value and therefore PoolQC will be none and none. Second highest correlated pair is GrLivArea and TotRmsAbvGrd. Against understandable as on average the more total rooms in a house, the larger the overall living area is going to be.

Looking at the Hoeffding Tree Plot, OverallQuality and SalePrice, which we have established already are highly correlated, as to is TotalBsmtSF and X1stFlSF.



CONCLUSION

The aim of this report has been to extensively analyse the housing dataset provided by the estate agency to better understand the housing market. The report utilised regression methods to identify the most important variables in predicting house prices and used machine learning classification methods to group houses based on the similarities of their features.

APPENDIX

R CODE

```
#Required packages
library(plyr)
library(dplyr)
library(tidyverse)
library(vtable)
library(na.tools)
library(psych)
library(gridExtra)
library(grid)
library(patchwork)
library(GGally)
library(tableone)
library(xtable)
library(ggplot2)
library(scales)
library(knitr)
library(corrplot)
library(randomForest)
library(faraway)
library(caret)
library(nnet)
library(MASS)
library(factoextra)
library(Hmisc)
library(VIM)
library(gridExtra)
```

```
#DATA2R
house_data <- read.csv("C:/Users/Ryan/Documents/University of Essex/MA335 - Modelling Experimental and Observational Data/Assessments/2.0/house_data.csv")
attach(house_data)
detach(house_data)

#IMPUTTING MISSING DATA & LABEL ENCODING
NAcol <- which(colSums(is.na(house_data)) > 0)
sort(colSums(sapply(house_data[NAcol], is.na)), decreasing = TRUE)
Qualities <- c('None' = 0, 'Po' = 1, 'Fa' = 2, 'TA' = 3, 'Gd' = 4, 'Ex' = 5)
#POOL QUALITY (Na's & FACTOR2ORDINAL)
house_data$PoolQC[is.na(house_data$PoolQC)] <- 'None'
house_data$PoolQC<-as.integer(revalue(house_data$PoolQC, Qualities))
#MISCELLANEOUS FEATURES (Na's & CHARACTER2FACTOR))
house_data$MiscFeature[is.na(house_data$MiscFeature)] <- 'None'
house_data$MiscFeature <- as.factor(house_data$MiscFeature)
#ALLEY (Na's & CHARACTER2FACTOR)
house_data$Alley[is.na(house_data$Alley)] <- 'None'
house_data$Alley <- as.factor(house_data$Alley)
#FENCE (Na's & CHARACTER2FACTOR)
house_data$Fence[is.na(house_data$Fence)] <- 'None'
house_data$Fence <- as.factor(house_data$Fence)
#LOT FRONTAGE (Na's)
for (i in 1:nrow(house_data)){
  if(is.na(house_data$LotFrontage[i])){
    house_data$LotFrontage[i] <- as.integer(median(house_data$LotFrontage[house_data$Neighborhood==house_data$Neighborhood[i]], na.rm=TRUE))
  }
}
#GARAGE TYPE (Na's & CHARACTER2FACTOR)
```

```
house_data$GarageType[is.na(house_data$GarageType)] <- 'No Garage'

house_data$GarageType <- as.factor(house_data$GarageType)

#GARAGE CONDITION (Na's & FACTOR2ORDINAL)

house_data$GarageCond[is.na(house_data$GarageCond)] <- 'None'

house_data$GarageCond<-as.integer(revalue(house_data$GarageCond, Qualities))

#BASEMENT QUALITY (Na's & FACTOR2ORDINAL)

house_data$BsmtQual[is.na(house_data$BsmtQual)] <- 'None'

house_data$BsmtQual<-as.integer(revalue(house_data$BsmtQual, Qualities))

#BASEMENT CONDITION (Na's & FACTOR2ORDINAL)

house_data$BsmtCond[is.na(house_data$BsmtCond)] <- 'None'

house_data$BsmtCond<-as.integer(revalue(house_data$BsmtCond, Qualities))

#MASONRY VENEER AREA (Na's)

house_data$MasVnrArea[is.na(house_data$MasVnrArea)] <-0

#KITCHEN QUALITY (FACTOR2ORDINAL)

house_data$KitchenQual<-as.integer(revalue(house_data$KitchenQual, Qualities))

#HOME FUNCTIONALITY (FACTOR2ORDINAL)

house_data$Functional <- as.integer(revalue(house_data$Functional, c('Sal'=0, 'Sev'=1, 'Maj2'=2,
'Maj1'=3, 'Mod'=4, 'Min2'=5, 'Min1'=6, 'Typ'=7)))

#EXTERIOR QUALITY (FACTOR2ORDINAL)

house_data$ExterQual<-as.integer(revalue(house_data$ExterQual, Qualities))

#EXTERIOR CONDITION (FACTOR2ORDINAL)

house_data$ExterCond<-as.integer(revalue(house_data$ExterCond, Qualities))

#SALE TYPE (CHARACTER2FACTOR)

house_data$SaleType <- as.factor(house_data$SaleType)

#SALE CONDITION (CHARACTER2FACTOR)

house_data$SaleCondition <- as.factor(house_data$SaleCondition)

#FOUNDATION (CHARACTER2FACTOR)

house_data$Foundation <- as.factor(house_data$Foundation)

#HEATING (CHARACTER2FACTOR)

house_data$Heating <- as.factor(house_data$Heating)

#ROOF STYLE (CHARACTER2FACTOR)
```

```
house_data$RoofStyle <- as.factor(house_data$RoofStyle)
```

#ROOF MATERIAL (CHARACTER2FACTOR)

```
house_data$RoofMatl <- as.factor(house_data$RoofMatl)
```

#BUILDING TYPE (CHARACTER2FACTOR)

```
house_data$BldgType <- as.factor(house_data$BldgType)
```

#HOUSE STYLE (CHARACTER2FACTOR))

```
house_data$HouseStyle <- as.factor(house_data$HouseStyle)
```

#NEIGHBORHOOD (CHARACTER2FACTOR)

```
house_data$Neighborhood <- as.factor(house_data$Neighborhood)
```

#CONDITION1 (CHARACTER2FACTOR)

```
house_data$Condition1 <- as.factor(house_data$Condition1)
```

#CONDITION2 (CHARACTER2FACTOR)

```
house_data$Condition2 <- as.factor(house_data$Condition2)
```

#STREET (CHARACTER2ORDINAL)

```
house_data$Street <- as.integer(revalue(house_data$Street, c('Grvl' = 0, 'Pave' = 1)))
```

#PAVED DRIVEWAY (CHARACTER2ORDINAL)

```
house_data$PavedDrive <- as.integer(revalue(house_data$PavedDrive, c('N'=0, 'P'=1, 'Y'=2)))
```

#EXTERIOR1ST (CHARACTER2FACTOR)

```
house_data$Exterior1st <- as.factor(house_data$Exterior1st)
```

#LOT CONFIGURATION (CHARACTER2FACTOR)

```
house_data$LotConfig <- as.factor(house_data$LotConfig)
```

#MONTH SOLD (INTEGER2FACTOR)

```
house_data$MoSold <- as.factor(house_data$MoSold)
```

#YEAR SOLD (INTEGER2FACTOR)

```
house_data$YrSold <- as.factor(house_data$YrSold)
```

#REMOVING UTILITIES VARIABLE

```
house_data$Utilities <- NULL
```

#1(EDA)

```
length(select_if(house_data,is.numeric))
```

#Summary Statistics Table

```
house_data %>%
  dplyr::select(SalePrice, OverallCond, OverallQual, Bedrooms, FullBath, LotFrontage, LotArea, YearBuilt, GrLivArea, TotalBsmtSF, TotRmsAbvGrd, Fireplaces, GarageArea) %>%
  sumtable()
sumtable(house_data)

#Bedrooms~SalesPrice
eda1<- ggplot(house_data, aes(Bedrooms,SalePrice)) +
  scale_y_continuous(labels = comma) +
  geom_jitter() +
  geom_smooth(method = "lm", col = "red") +
  xlab("Number of Bedrooms") + ylab("Sale Price")

#Bathrooms~SalesPrice
eda2<-ggplot(house_data, aes(FullBath,SalePrice)) +
  scale_y_continuous(labels = comma) +
  geom_jitter() +
  geom_smooth(method = "lm", col = "red") +
xlab("Number of Bathrooms") + ylab("Sale Price")

#TotalBasementArea~SalePrice
eda3<- ggplot(house_data, aes(TotalBsmtSF,SalePrice)) +
  scale_y_continuous(labels = comma) +
  geom_jitter() +
  geom_smooth(method = "lm", col = "red") +
xlab("Total square feet of Basement area") + ylab("Sale Price")

#Sale Price Frequency
eda8 <- ggplot(house_data, aes(SalePrice)) +
  geom_histogram(fill="blue", binwidth = 10000) +
  scale_x_continuous(breaks= seq(0, 800000, by=100000), labels = comma) +
  xlab("Sale Price") + ylab("Frequency")

#Frequency Overall Quality
ggplot(house_data, aes(OverallQual)) +
  scale_x_continuous(n.breaks = 10) +
  geom_bar() +
  xlab("Overall Quality") + ylab("Frequency")

#Quality~SalePrice
eda4<- ggplot(house_data, aes(OverallQual,SalePrice)) +
  scale_y_continuous(labels = comma) +
  scale_x_continuous(n.breaks = 10) +
  geom_jitter() +
  geom_smooth(method = "lm", col = "red") +
  xlab("Overall Quality") + ylab("Sale Price")

#Condition~SalePrice
ggplot(house_data, aes(OverallCond,SalePrice)) +
  scale_y_continuous(labels = comma) +
  scale_x_continuous(n.breaks = 10) +
  geom_jitter() +
  geom_smooth(method = "lm", col = "red") +
  xlab("Overall Condition") + ylab("Sale Price")

#LotArea~SalePrice
eda5<- ggplot(house_data, aes(LotArea,SalePrice)) +
  scale_y_continuous(labels = comma) +
  xlim(0,75000) +
  geom_jitter() +
  geom_smooth(method = "lm", col = "red") +
  xlab("Lot Area") + ylab("Sale Price")
```

```
#YearBuilt~SalePrice

eda6<- ggplot(house_data, aes(YearBuilt,SalePrice)) +

  scale_y_continuous(labels = comma) +

  scale_x_binned()+

  geom_jitter() +

  geom_smooth(method = "lm", col = "red") +

  xlab("Year Built") + ylab("Sale Price")
```

```
#YearSold~SalePrice

ggplot(house_data, aes(YrSold,SalePrice)) +

    scale_y_continuous(labels = comma) +

    geom_jitter() +

    geom_smooth(method = "lm", col = "red") +

  xlab("Year Sold") + ylab("Sale Price")
```

```
#ExteriorCondition~SalePrice

ggplot(house_data, aes(ExterCond, SalePrice)) +

  scale_y_continuous(labels = comma) +

  geom_jitter() +

  geom_smooth(method = "lm", col = "red") +

  xlab("Exterior Condition") + ylab("Sale Price")
```

```
#NEIGHBOURHOOD MEDIAN HOUSE PRICES

eda7 <- ggplot(house_data, aes(x=reorder(Neighborhood, SalePrice, FUN=median), y=SalePrice)) +

  geom_bar(stat='summary', fun.y = "median", fill='blue') + labs(x='Neighborhood', y='Mean SalePrice') +

  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +

  scale_y_continuous(breaks= seq(0, 800000, by=50000), labels = comma)


grid.arrange(eda1,eda2,eda3,eda4,eda5,eda6, nrow = 2)
grid.arrange(eda8,eda7)

#CORRELATION MATRIX
#selecting numeric variables to use
numericVars <- select_if(house_data, is.numeric)
numericVarNames <- names(numericVars)
#correlations of all numeric variables
cor_numVar <- cor(numericVars, use="pairwise.complete.obs")
#sort on decreasing correlations with SalePrice
cor_sorted <- as.matrix(sort(cor_numVar[,'SalePrice'], decreasing = TRUE))
#select only high correlations
CorHigh <- names(which(apply(cor_sorted, 1, function(x) abs(x)>0.5)))
cor_numVar <- cor_numVar[CorHigh, CorHigh]
#plotting correlation matrix
corrplot.mixed(cor_numVar, tl.col="black", tl.pos = "lt")
```

#2 (LINEAR REGRESSION)

```
#full model
model1 <- lm(SalePrice~.,data=numericVars)
#model output
summary(model1)
#checking for collinearity
vif(model1)
#reduced model
model2 <- lm(SalePrice~Neighborhood+GrLivArea+OverallQual+OverallCond+GarageArea+TotalBsmtSF+BsmtQual+LotArea+Fireplaces)
#reduced model output
summary(model2)
#plotting model output
```

```
plot(model2)

#checking for collinearity

vif(model2)

#comparing full and reduced model

anova(model1,model2)
```

#3 (LR, QDA & LDA)

```
#PREPARING DATA 4 MODELLING

classifyds <- house_data

attach(classifyds)

detach(classifyds)

#Dropping highly correlated variables & redundant variables

dropvars <- c('Id', 'GarageCond', 'MiscVal', 'SaleType')

classifyds <- classifyds[,!(names(classifyds) %in% dropvars)]

#Removing Outliers

classifyds <- classifyds[-c(524,1299)]

#Selecting only numeric variables

classifyds <- classifyds[,sapply(classifyds, is.numeric)]

#adding OverallCond to dataset

classifyds$OverallCond <- house_data$OverallCond

#changing OverallCond to factor with 3 levels

classifyds$OverallCond=cut(classifyds$OverallCond, br=c(0,3,6,10), labels = c("Poor", "Average",
"Good"))
```

#LOGISITC REGRESSION

```
set.seed(123)

#splitting data into training and test set

training.samples <- classifyds$OverallCond %>%

  createDataPartition(p = 0.8, list = FALSE)

train.data <- classifyds[training.samples, ]
```

```
test.data <- classifyds[-training.samples, ]
#fitting the model
LogRegModel <- nnet::multinom(OverallCond ~., data = train.data)
#model output
summary(LogRegModel)
str(LogRegModel)
#predictions
predicted.classes <- LogRegModel %>% predict(test.data)
#model accuracy
mean(predicted.classes == test.data$OverallCond)
#error rate
error <- mean(test.data$OverallCond != predicted.classes)
error

#LINEAR DISCRIMINANT ANALYSIS
#normalising data & estimating preprocessing params
preproc.param <- train.data %>%
  preProcess(method = c("center", "scale"))
#transforming data using estimated params
train.transformed <- preproc.param %>% predict(train.data)
test.transformed <- preproc.param %>% predict(test.data)
#model fitting
ldamodel <- lda(OverallCond~., data = train.transformed)
#predictions
predictions <- ldamodel %>% predict(test.transformed)
#model accuracy
mean(predictions$class==test.transformed$OverallCond)
names(predictions)
head(predictions$class)
head(predictions$posterior)
head(predictions$x)
```

```
#VISUALISING LDA
lda.data <- cbind(train.transformed, predict(ldamodel)$x)
ggplot(lda.data, aes(LD1, LD2)) +
  geom_point(aes(color = OverallCond))
```

#4 (CLUSTERING ALGORITHMS)

```
clusteringdf <- classifyds
clusteringdf <- subset(clusteringdf, select = -c(OverallCond, Bedrooms))
clusteringdf <- scale(clusteringdf)
```

#Heirarchical Kmeans clustering

```
#determining optimal clusters
fviz_nbclust(clusteringdf, kmeans, method = "wss")+
  geom_vline(xintercept = 4, linetype = 2)
#performing HKmeans
res.hk <-hkmeans(clusteringdf, 4, "euclidean")
#dendogram plot
fviz_dend(res.hk, cex = 0.6, palette = "jco",
          rect = TRUE, rect_border = "jco", rect_fill = TRUE)
#cluster plot
fviz_cluster(res.hk, palette = "jco", repel = TRUE,
             ggtheme = theme_classic())
```

#VARIABLE CLUSTERING

#variable clustering using spearman method

```
clus_var <- varclus(as.matrix(clusteringdf), similarity = "spearman", minlev = 0.05)
plot(clus_var, cex = 0.5)
```

#variable clustering using hoeffding method

```
clus_var2 <- varclus(as.matrix(clusteringdf), similarity = "hoeffding", minlev = 0.05)

plot(clus_var2, cex = 0.5)
```