# Introduction

### Here are the main parts of this project:

### In the first part, we use data visualisation method to analyse the relationship between price and other variables.

### In the second part, we use classification method to predict the house quality level, given a related continuous variable.

### In the third part, we build a linear regression model of price.

### First we find the general relationship among continious variables. Then we want to find out the natural clusters in the data, and find out how does it relate to the categorical variables.

##   
## Attaching package: 'dplyr'

## The following object is masked from 'package:car':  
##   
## recode

## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

##   
## Attaching package: 'GGally'

## The following object is masked from 'package:dplyr':  
##   
## nasa

## Loading required package: grid

## -------------------------------------------------------------------------

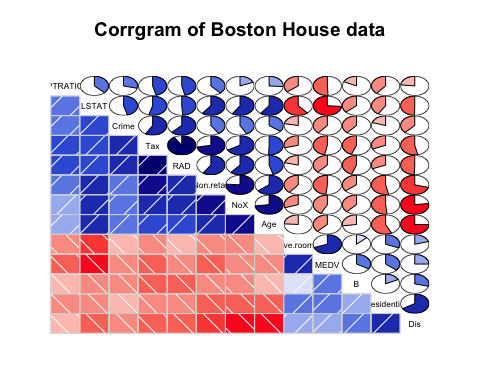
## You have loaded plyr after dplyr - this is likely to cause problems.  
## If you need functions from both plyr and dplyr, please load plyr first, then dplyr:  
## library(plyr); library(dplyr)

## -------------------------------------------------------------------------

##   
## Attaching package: 'plyr'

## The following objects are masked from 'package:dplyr':  
##   
## arrange, count, desc, failwith, id, mutate, rename, summarise,  
## summarize

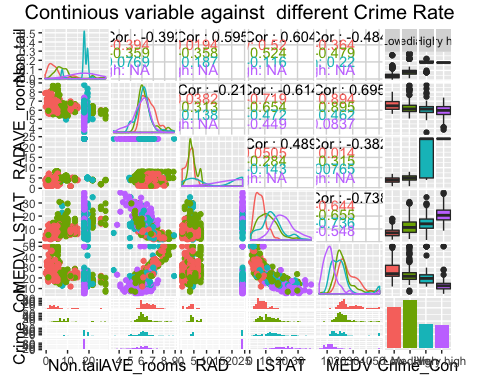
## The following object is masked from 'package:corrgram':  
##   
## baseball



## [1] 506

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.

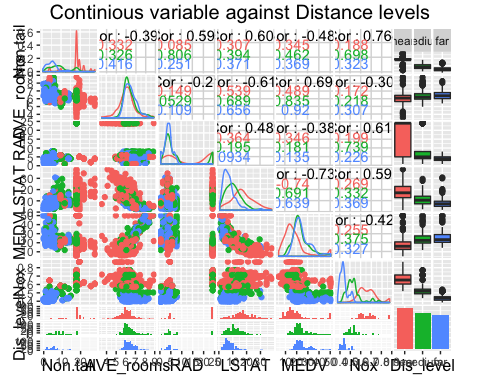
## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.  
## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.  
## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.  
## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



<1> From the graph, the housing price is the lowest where the crime rate is highest; and the housing price is highest where the crime rate is lowest.<2> In "very high" crime rate area, the percentage of low status is the highest; in low crime rate location, the percentage of low status is the lowest.<3> In the "very high" crime rate area, the index of accessibility of radial highwahs is the highest, and in "low" crime rate area, the index is lowest. <4> In the "very high" crime area, the non-retail businese propotion is highest, and in "low" crime rate area, the propotion is the lowest.

## Split distance to the business centers to three different levels "near", "medium", or "far".  
Distance\_level = cut(House$Dis, c(1,2.5, 4.5, 13), labels = c("near", "medium", "far"))  
Selected\_2 = data.frame(House$Non.retail, House$Ave.rooms, House$RAD, House$LSTAT,House$MEDV,House$NoX,Distance\_level)  
names(Selected\_2) = c('Non.tail','AVE\_rooms','RAD', 'LSTAT', 'MEDV','Nox',"Dis\_level")  
## Use ggpairs to find the natural cluster.  
ggpairs(Selected\_2, mapping = ggplot2::aes(color = Dis\_level),title = "Continious variable against Distance levels")

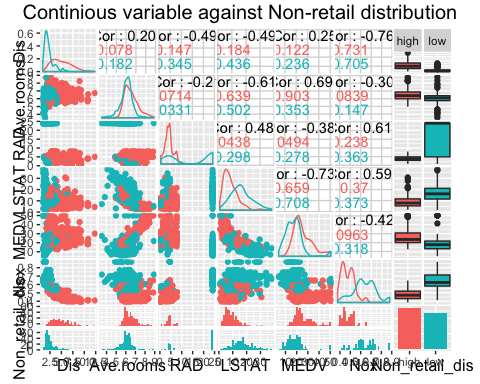
## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.  
## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.  
## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.  
## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.  
## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.  
## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



<1> From the graph, the nearer the distance to five Boston employment centres, the lower of the housing price is. The nearer the distance to the centres, the higher percentage of lower status propotion is. The non-retail businesses are mostly located at the area that closed to the center of business. The nearer the distance to five Boston employment centres, the higher concentration of NoX.

## Split proportion of area of non-retail businesses to three different levels "near", "medium", or "far".  
Non\_retail\_dis = cut(House$Non.retail, c(0,10,30), labels = c("high", "low"))  
Selected\_3 = data.frame(House$Dis, House$Ave.rooms, House$RAD, House$LSTAT,House$MEDV,House$NoX,Non\_retail\_dis)  
names(Selected\_3) = c('Dis','Ave.rooms','RAD', 'LSTAT', 'MEDV','Nox',"Non\_retail\_dis")  
  
## Use ggpairs to find the natural cluster  
  
ggpairs(Selected\_3, mapping = ggplot2::aes(color = Non\_retail\_dis),title = "Continious variable against Non-retail distribution")

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.  
## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.  
## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.  
## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.  
## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.  
## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



<1> From the graph, we could see that the lower concentration of NoX, the higher non-retail rate.

Since both the level of crime rate and levels of distance to business center affect the Boston housing price, we want to see that given different specific levels of one categorical variable, how another categorical variable affect affect housing price. Then we can find out which categorical variable affects the housing price more.

 In three different distance level, the low crime rate will cause high housing price.However, the effect of crime rate on housing price is most in near distance level than others.

### Since we found out that crime rate is highly related to the housing price of Boston, we are interested in construct a classification model to predict whether the house is "Mediocre" or "Supreme". (We split the housing price into 2 levels "Mediocre" and "Supreme", we also assume that price determines the quality of house.)

##   
## Attaching package: 'MASS'

## The following object is masked from 'package:dplyr':  
##   
## select

## model  
## true Mediocre Supreme  
## Mediocre 45 0  
## Supreme 30 15

Since the misclassification rate for the KNN prediction model is 1/3, is relatively high. Then we try alternative method to make classification.

## Call:  
## lda(Price\_level ~ Crime, data = training\_data)  
##   
## Prior probabilities of groups:  
## Mediocre Supreme   
## 0.5 0.5   
##   
## Group means:  
## Crime  
## Mediocre 6.1047136  
## Supreme 0.2894595  
##   
## Coefficients of linear discriminants:  
## LD1  
## Crime 0.1129541

##   
## Mediocre Supreme  
## Mediocre 44 21  
## Supreme 1 24

The misclassification rate for LDA model is 0.24, which is lower than that of KNN method. Therefore for the classificaiton, LDA method is more accurate.

# Model selection

### After we analyze the relation of the data using data visualization method, now, we are interested in finding out how housing price relate to other factors in a numercal way. Thus, we need to build a statistical model about the relationship between housing price and other variables. Here we choose linear regression as a main method.

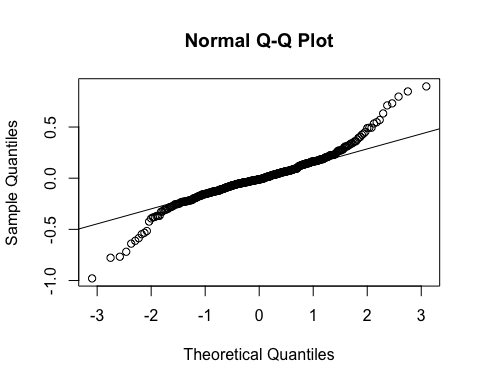
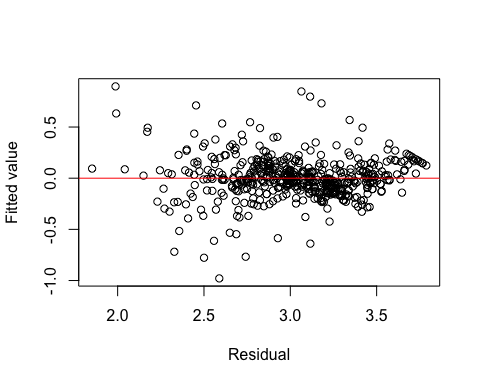
First, we detect multicollinearity, and discard the variables that have high collinearity.

## Residential Non.retail NoX Ave.rooms Age Dis   
## 2.281047 3.942099 4.372226 1.930358 3.092830 3.901573   
## RAD Tax PTRATIO B LSTAT   
## 6.884474 8.875931 1.782825 1.325511 2.859635

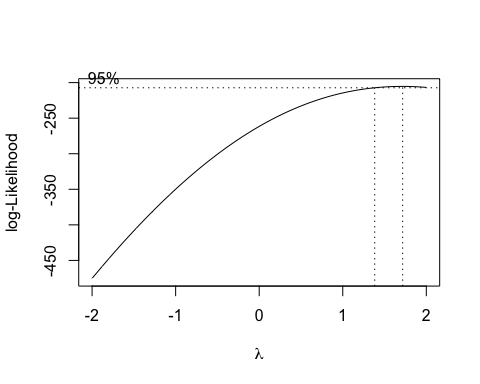
## Residential Non.retail NoX Ave.rooms Age Dis   
## 2.105199 3.168180 3.867240 1.865004 3.064799 3.887726   
## PTRATIO B LSTAT   
## 1.429913 1.258265 2.822066

## Start: AIC=1620.78  
## MEDV ~ Residential + Non.retail + NoX + Ave.rooms + Age + Dis +   
## PTRATIO + B + LSTAT  
##   
## Df Sum of Sq RSS AIC  
## - Age 1 0.13 11970 1618.8  
## - Non.retail 1 8.84 11979 1619.2  
## <none> 11970 1620.8  
## - Residential 1 174.39 12144 1626.1  
## - B 1 328.50 12298 1632.5  
## - NoX 1 387.45 12357 1634.9  
## - Dis 1 1202.66 13172 1667.2  
## - PTRATIO 1 1354.91 13325 1673.0  
## - Ave.rooms 1 2311.95 14282 1708.1  
## - LSTAT 1 2735.97 14706 1722.9  
##   
## Step: AIC=1618.79  
## MEDV ~ Residential + Non.retail + NoX + Ave.rooms + Dis + PTRATIO +   
## B + LSTAT  
##   
## Df Sum of Sq RSS AIC  
## - Non.retail 1 8.86 11979 1617.2  
## <none> 11970 1618.8  
## - Residential 1 179.01 12149 1624.3  
## - B 1 329.97 12300 1630.5  
## - NoX 1 418.33 12388 1634.2  
## - Dis 1 1305.83 13276 1669.2  
## - PTRATIO 1 1358.17 13328 1671.2  
## - Ave.rooms 1 2397.04 14367 1709.2  
## - LSTAT 1 3099.57 15069 1733.3  
##   
## Step: AIC=1617.16  
## MEDV ~ Residential + NoX + Ave.rooms + Dis + PTRATIO + B + LSTAT  
##   
## Df Sum of Sq RSS AIC  
## <none> 11979 1617.2  
## - Residential 1 178.79 12158 1622.7  
## - B 1 338.91 12318 1629.3  
## - NoX 1 568.35 12547 1638.6  
## - Dis 1 1333.09 13312 1668.5  
## - PTRATIO 1 1512.69 13491 1675.3  
## - Ave.rooms 1 2467.91 14447 1710.0  
## - LSTAT 1 3154.27 15133 1733.4

##   
## Call:  
## lm(formula = MEDV ~ Residential + NoX + Ave.rooms + Dis + PTRATIO +   
## B + LSTAT, data = House.2)  
##   
## Coefficients:  
## (Intercept) Residential NoX Ave.rooms Dis   
## 30.42679 0.03667 -15.83674 4.18301 -1.41173   
## PTRATIO B LSTAT   
## -0.92370 0.01000 -0.55145



##   
## Call:  
## lm(formula = log(MEDV) ~ Residential + NoX + Ave.rooms + Dis +   
## PTRATIO + B + LSTAT, data = House.2)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.97921 -0.10547 -0.01107 0.09218 0.89706   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 3.9049782 0.2067680 18.886 < 2e-16 \*\*\*  
## Residential 0.0002524 0.0005644 0.447 0.655   
## NoX -0.7311178 0.1366959 -5.348 1.35e-07 \*\*\*  
## Ave.rooms 0.1043951 0.0173270 6.025 3.29e-09 \*\*\*  
## Dis -0.0433503 0.0079565 -5.448 8.00e-08 \*\*\*  
## PTRATIO -0.0409799 0.0048871 -8.385 5.23e-16 \*\*\*  
## B 0.0005502 0.0001118 4.921 1.17e-06 \*\*\*  
## LSTAT -0.0315562 0.0020205 -15.618 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.2058 on 498 degrees of freedom  
## Multiple R-squared: 0.7501, Adjusted R-squared: 0.7466   
## F-statistic: 213.5 on 7 and 498 DF, p-value: < 2.2e-16



##   
## Call:  
## lm(formula = I(MEDV^0.22) ~ Residential + NoX + Ave.rooms + Dis +   
## PTRATIO + B + LSTAT, data = House.2)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.34605 -0.04761 -0.00817 0.04184 0.39698   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 2.283e+00 8.744e-02 26.111 < 2e-16 \*\*\*  
## Residential 2.519e-04 2.387e-04 1.056 0.292   
## NoX -3.152e-01 5.781e-02 -5.453 7.81e-08 \*\*\*  
## Ave.rooms 5.283e-02 7.327e-03 7.211 2.08e-12 \*\*\*  
## Dis -2.061e-02 3.365e-03 -6.126 1.84e-09 \*\*\*  
## PTRATIO -1.767e-02 2.067e-03 -8.548 < 2e-16 \*\*\*  
## B 2.269e-04 4.728e-05 4.800 2.11e-06 \*\*\*  
## LSTAT -1.284e-02 8.544e-04 -15.031 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.08702 on 498 degrees of freedom  
## Multiple R-squared: 0.7535, Adjusted R-squared: 0.75   
## F-statistic: 217.5 on 7 and 498 DF, p-value: < 2.2e-16

## [1] 253

## [1] 253

##   
## Call:  
## lm(formula = I(MEDV^0.22) ~ NoX + Ave.rooms + Dis + PTRATIO +   
## B + LSTAT, data = data.training)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.271138 -0.045348 -0.001072 0.043783 0.292583   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 2.212e+00 1.144e-01 19.339 < 2e-16 \*\*\*  
## NoX -2.602e-01 7.387e-02 -3.522 0.00051 \*\*\*  
## Ave.rooms 5.563e-02 9.519e-03 5.844 1.61e-08 \*\*\*  
## Dis -2.017e-02 3.862e-03 -5.223 3.74e-07 \*\*\*  
## PTRATIO -1.702e-02 2.483e-03 -6.855 5.71e-11 \*\*\*  
## B 2.903e-04 6.485e-05 4.476 1.16e-05 \*\*\*  
## LSTAT -1.392e-02 1.105e-03 -12.599 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.079 on 246 degrees of freedom  
## Multiple R-squared: 0.8077, Adjusted R-squared: 0.803   
## F-statistic: 172.2 on 6 and 246 DF, p-value: < 2.2e-16

##   
## Call:  
## lm(formula = I(MEDV^0.22) ~ NoX + Ave.rooms + Dis + PTRATIO +   
## B + LSTAT, data = data.testing)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.34641 -0.04879 -0.01304 0.03736 0.41370   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 2.352e+00 1.335e-01 17.623 < 2e-16 \*\*\*  
## NoX -3.736e-01 8.933e-02 -4.182 4.02e-05 \*\*\*  
## Ave.rooms 5.242e-02 1.097e-02 4.778 3.04e-06 \*\*\*  
## Dis -1.799e-02 4.455e-03 -4.039 7.19e-05 \*\*\*  
## PTRATIO -1.967e-02 3.158e-03 -6.229 2.02e-09 \*\*\*  
## B 1.728e-04 6.895e-05 2.506 0.0129 \*   
## LSTAT -1.145e-02 1.324e-03 -8.644 7.06e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.09447 on 246 degrees of freedom  
## Multiple R-squared: 0.6993, Adjusted R-squared: 0.692   
## F-statistic: 95.35 on 6 and 246 DF, p-value: < 2.2e-16

##   
## Call:  
## lm(formula = log(MEDV) ~ NoX + Ave.rooms + Dis + PTRATIO + B +   
## LSTAT, data = data.training)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.76029 -0.09703 0.00734 0.09863 0.69875   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 3.7224431 0.2727226 13.649 < 2e-16 \*\*\*  
## NoX -0.5785637 0.1760927 -3.286 0.00117 \*\*   
## Ave.rooms 0.1088667 0.0226918 4.798 2.79e-06 \*\*\*  
## Dis -0.0444689 0.0092054 -4.831 2.39e-06 \*\*\*  
## PTRATIO -0.0380513 0.0059184 -6.429 6.61e-10 \*\*\*  
## B 0.0007109 0.0001546 4.599 6.81e-06 \*\*\*  
## LSTAT -0.0348564 0.0026332 -13.237 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.1883 on 246 degrees of freedom  
## Multiple R-squared: 0.8049, Adjusted R-squared: 0.8002   
## F-statistic: 169.2 on 6 and 246 DF, p-value: < 2.2e-16

##   
## Call:  
## lm(formula = log(MEDV) ~ NoX + Ave.rooms + Dis + PTRATIO + B +   
## LSTAT, data = data.testing)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.98885 -0.10701 -0.02301 0.09960 0.88842   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 4.0856901 0.3120858 13.092 < 2e-16 \*\*\*  
## NoX -0.8937600 0.2089078 -4.278 2.70e-05 \*\*\*  
## Ave.rooms 0.1021503 0.0256529 3.982 9.00e-05 \*\*\*  
## Dis -0.0399113 0.0104185 -3.831 0.000162 \*\*\*  
## PTRATIO -0.0456934 0.0073853 -6.187 2.54e-09 \*\*\*  
## B 0.0004219 0.0001613 2.616 0.009439 \*\*   
## LSTAT -0.0274616 0.0030967 -8.868 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.2209 on 246 degrees of freedom  
## Multiple R-squared: 0.6965, Adjusted R-squared: 0.6891   
## F-statistic: 94.1 on 6 and 246 DF, p-value: < 2.2e-16