Homework 1

Erik Pak

2024-05-10

Import Libraries

```
# Load library
library(tidyverse)      # Data manipulation package
library(dplyr)          # rename feature
library(corrplot)       # correlation plot
library(psych)          # describe & pairs.panel plot
library(ggplot2)        # used for ggplot
library(car)            # Variance Inflation Factor (VIF)
library(leaps)          # model selection methods
library(QuantPsyc)      # normalize coefficients
library(pracma)         # calculate dot product between vectors
```

Exploratory Data Analysis

```
# load csv data to a dataframe
data <- read.csv("olympics.csv")

# check dimension of the dataframe & view
glimpse(data)
```

```
## Rows: 20
## Columns: 9
## $ ISO.country.code <chr> "USA", "CHN", "JPN", "DEU", "FRA", "BRA", "GBR", "ITA~
## $ Country.name     <chr> "US", "China", "Japan", "Germany", "France", "Brazil"~
## $ X2011.GDP        <dbl> 1.509400e+13, 7.298100e+12, 5.867150e+12, 3.570560e+1~
## $ X2010.population <int> 309349000, 1338300000, 127451000, 81777000, 64895000,~
## $ Female.count     <int> 271, 208, 162, 176, 148, 128, 269, 122, 227, 23, 25, ~
## $ Male.count       <int> 260, 163, 141, 219, 187, 138, 287, 159, 208, 60, 25, ~
## $ Gold.medals      <int> 46, 38, 7, 11, 11, 3, 29, 8, 24, 0, 4, 1, 4, 0, 0, 1,~
## $ Silver.medals    <int> 29, 27, 14, 19, 11, 5, 17, 9, 26, 2, 4, 3, 0, 1, 2, 0~
## $ Bronze.medals    <int> 29, 23, 17, 14, 12, 9, 19, 11, 32, 4, 4, 3, 2, 2, 3, ~
```

```
# check for missing values
sum(is.na(data))
```

```
## [1] 0
```

```r
# describe data excluding character columns
describe(data)
```

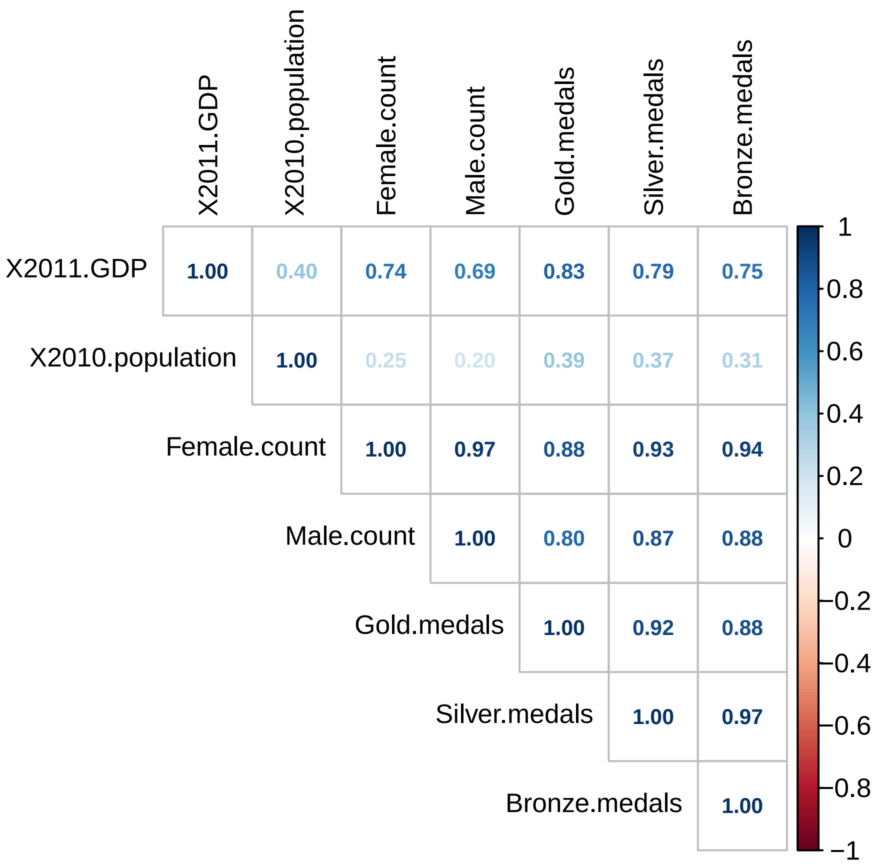
```
##                    vars  n         mean           sd       median      trimmed
## ISO.country.code*     1 20 1.050000e+01 5.920000e+00 1.050000e+01 1.050000e+01
## Country.name*         2 20 1.050000e+01 5.920000e+00 1.050000e+01 1.050000e+01
## X2011.GDP             3 20 2.274939e+12 3.671919e+12 9.315249e+11 1.443832e+12
## X2010.population      4 20 1.827455e+08 3.847068e+08 4.253606e+07 6.822170e+07
## Female.count          5 20 9.330000e+01 9.743000e+01 3.250000e+01 8.244000e+01
## Male.count            6 20 9.960000e+01 9.609000e+01 4.450000e+01 8.919000e+01
## Gold.medals           7 20 9.400000e+00 1.378000e+01 3.500000e+00 6.500000e+00
## Silver.medals         8 20 8.500000e+00 1.002000e+01 3.500000e+00 7.120000e+00
## Bronze.medals         9 20 9.350000e+00 1.000000e+01 4.000000e+00 7.880000e+00
##                            mad       min        max        range skew kurtosis
## ISO.country.code* 7.410000e+00         1 2.0000e+01 1.900000e+01 0.00    -1.38
## Country.name*     7.410000e+00         1 2.0000e+01 1.900000e+01 0.00    -1.38
## X2011.GDP         1.373792e+12 816054092 1.5094e+13 1.509318e+13 2.19     4.72
## X2010.population  5.901795e+07    104000 1.3383e+09 1.338196e+09 2.29     3.79
## Female.count      4.300000e+01         3 2.7100e+02 2.680000e+02 0.56    -1.34
## Male.count        5.263000e+01         6 2.8700e+02 2.810000e+02 0.52    -1.35
## Gold.medals       5.190000e+00         0 4.6000e+01 4.600000e+01 1.45     0.77
## Silver.medals     5.190000e+00         0 2.9000e+01 2.900000e+01 0.87    -0.80
## Bronze.medals     5.930000e+00         0 3.2000e+01 3.200000e+01 0.91    -0.50
##                             se
## ISO.country.code* 1.320000e+00
## Country.name*     1.320000e+00
## X2011.GDP         8.210661e+11
## X2010.population  8.602306e+07
## Female.count      2.179000e+01
## Male.count        2.149000e+01
## Gold.medals       3.080000e+00
## Silver.medals     2.240000e+00
## Bronze.medals     2.240000e+00
```

```r
# summary of data
summary(data)
```

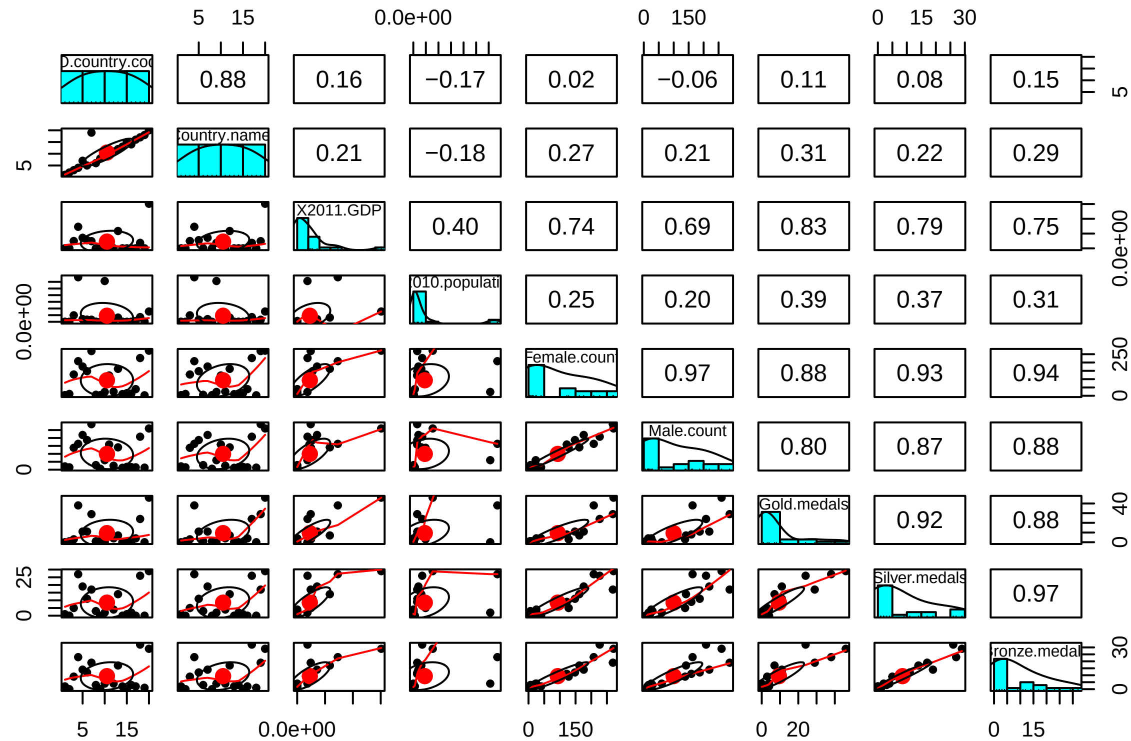
```
##  ISO.country.code   Country.name         X2011.GDP         X2010.population
##  Length:20          Length:20          Min.   :8.161e+08   Min.   :1.040e+05
##  Class :character   Class :character   1st Qu.:8.365e+09   1st Qu.:3.008e+06
##  Mode  :character   Mode  :character   Median :9.315e+11   Median :4.254e+07
##                                        Mean   :2.275e+12   Mean   :1.827e+08
##                                        3rd Qu.:2.551e+12   3rd Qu.:1.310e+08
##                                        Max.   :1.509e+13   Max.   :1.338e+09
##   Female.count      Male.count      Gold.medals   Silver.medals
##  Min.   :  3.00   Min.   :  6.00   Min.   : 0.0   Min.   : 0.00
##  1st Qu.: 10.75   1st Qu.: 15.75   1st Qu.: 0.0   1st Qu.: 0.75
##  Median : 32.50   Median : 44.50   Median : 3.5   Median : 3.50
##  Mean   : 93.30   Mean   : 99.60   Mean   : 9.4   Mean   : 8.50
##  3rd Qu.:165.50   3rd Qu.:169.00   3rd Qu.:11.0   3rd Qu.:14.75
##  Max.   :271.00   Max.   :287.00   Max.   :46.0   Max.   :29.00
##  Bronze.medals
##  Min.   : 0.00
```

```
##  1st Qu.: 2.00
##  Median : 4.00
##  Mean   : 9.35
##  3rd Qu.:14.75
##  Max.   :32.00
```

```r
# correlation
corr <- cor(data[, -1:-2])

# plot correlation
corrplot(corr, method = 'number', addCoef.col = 'green', tl.col = "black",
        type = 'upper', number.cex = 0.7, tl.cex = 0.8,)
```




```r
# pairs panels plot
pairs.panels(data)
```



The initial exploratory data analysis on a dataset with 20 observations and nine independent variables has obvious evidence of a linear relationship between GDP, and all the medals earned. In addition, all observed quantitative values are skewed to the right, confirmed by the pairs panel plot. Interestingly, there was a significant positive coorelation between female and male athletes, including the type of Olympic medals awarded. This suggests that countries that perform well in the Olympics tend to perform well in both male and female events, and tend to win both gold, silver and bronze medals in these events.

Feature Engineering

```
# removing first two columns & copy
data_cleaned <- data[-c(1:2)]

# column names
names(data_cleaned)
```

```
## [1] "X2011.GDP"        "X2010.population" "Female.count"     "Male.count"
## [5] "Gold.medals"      "Silver.medals"    "Bronze.medals"
```

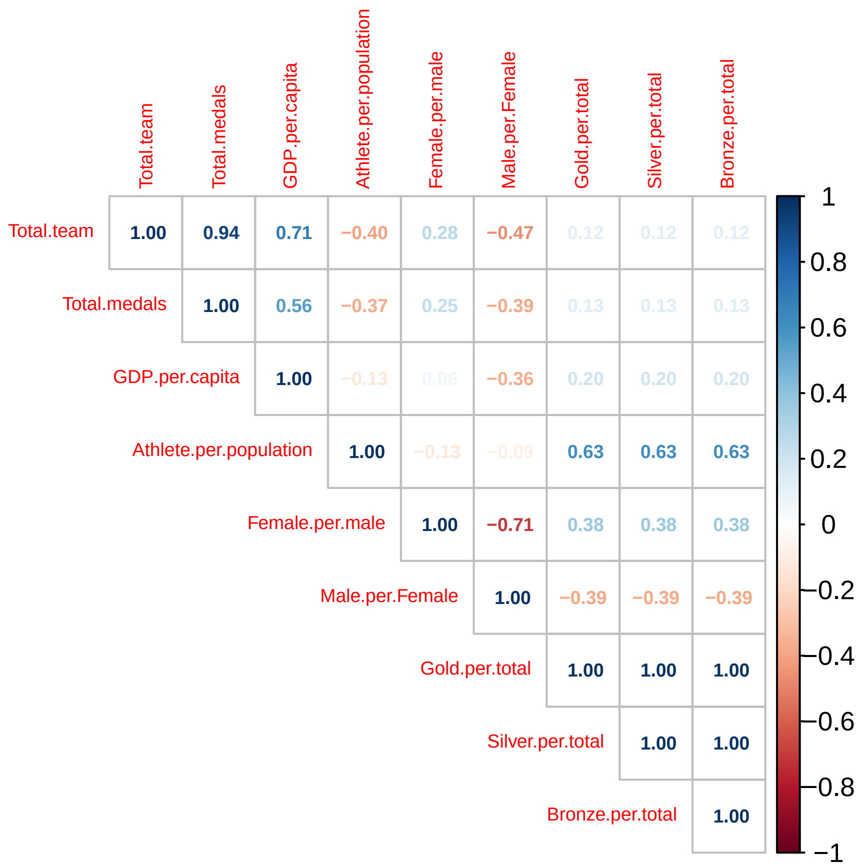
```
# change column using colnames()
colnames(data_cleaned)[1] = "GDP.2011"
colnames(data_cleaned)[2] = "Population.2010"

# New Features created from existing features
# total Team count
data_cleaned$Total.team <- data_cleaned$Female.count + data_cleaned$Female.count
```

```r
# total medals
data_cleaned$Total.medals <- (data_cleaned$Gold.medals +
                                data_cleaned$Silver.medals +
                                data_cleaned$Bronze.medals)

# create GDP per capita
data_cleaned$GDP.per.capita <- data_cleaned$GDP.2011 / data_cleaned$Population.2010

# create athlete per population
data_cleaned$Athlete.per.population <- (data_cleaned$Male.count +
              data_cleaned$Female.count) / data_cleaned$Population.2010

# create female per male
data_cleaned$Female.per.male <- data_cleaned$Female.count / data_cleaned$Male.count

# create female per male
data_cleaned$Male.per.Female <- data_cleaned$Male.count / data_cleaned$Female.count

# create gold per total medal
data_cleaned$Gold.per.total <- data_cleaned$Gold.medals /
  (data_cleaned$Gold.medals +  data_cleaned$Silver.medals +
     data_cleaned$Bronze.medals)

# create silver per total medal
data_cleaned$Silver.per.total <- data_cleaned$Gold.medals /
  (data_cleaned$Gold.medals +  data_cleaned$Silver.medals +
     data_cleaned$Bronze.medals)

# create bronze per total medal
data_cleaned$Bronze.per.total <- data_cleaned$Gold.medals /
  (data_cleaned$Gold.medals +  data_cleaned$Silver.medals +
     data_cleaned$Bronze.medals)

# remove features that were used to create new features
data_cleaned <- data_cleaned[-c(1:7)]

# lists the "structure" of the dataset
str(data_cleaned)
```

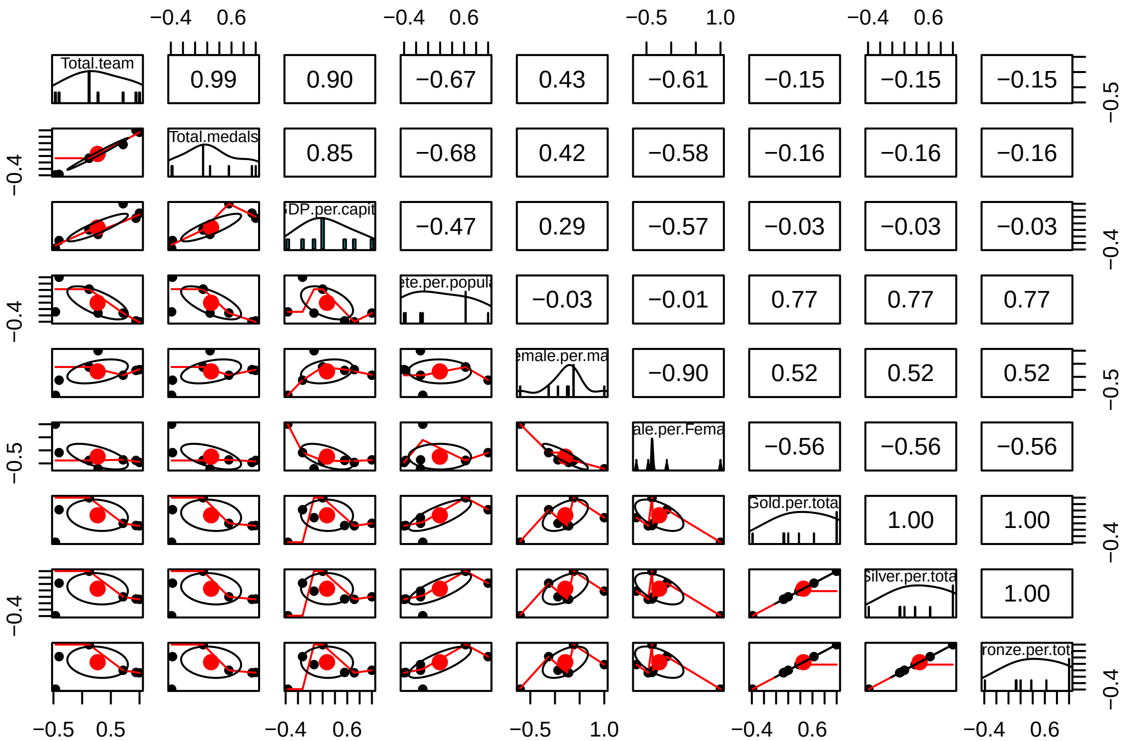
```
## 'data.frame':    20 obs. of  9 variables:
##  $ Total.team            : int  542 416 324 352 296 256 538 244 454 46 ...
##  $ Total.medals          : int  104 88 38 44 34 17 65 28 82 6 ...
##  $ GDP.per.capita        : num  48793 5453 46035 43662 42731 ...
##  $ Athlete.per.population: num  1.72e-06 2.77e-07 2.38e-06 4.83e-06 5.16e-06 ...
##  $ Female.per.male       : num  1.042 1.276 1.149 0.804 0.791 ...
##  $ Male.per.Female       : num  0.959 0.784 0.87 1.244 1.264 ...
##  $ Gold.per.total        : num  0.442 0.432 0.184 0.25 0.324 ...
##  $ Silver.per.total      : num  0.442 0.432 0.184 0.25 0.324 ...
##  $ Bronze.per.total      : num  0.442 0.432 0.184 0.25 0.324 ...
```

```r
# correlation
corr_cleaned <- cor(data_cleaned)
```

```r
# plot correlation
corrplot(corr_cleaned, method = 'number', addCoef.col = 'green',
                       type = 'upper', number.cex = 0.6, tl.cex = 0.6,)
```




```r
# pairs.panels plot
pairs.panels(corr_cleaned, pch = 16)
```



Created new features to gain additional information regarding the data. These new features seem useful and can potentially provide more insights into the data.

Brief overview of what these new features represent:

Total.team: This feature represents the total number of athletes in a team, which is calculated by adding the male and female counts.

Total.medals: This feature represents the total number of medals won by a team, which is calculated by adding the gold, silver, and bronze medals.

GDP.per.capita: This feature represents the Gross Domestic Product (GDP) per capita, which is calculated by dividing the GDP by the population.

Athlete.per.population: This feature represents the number of athletes per capita, which is calculated by dividing the total team count by the population.

Female.per.male: This feature represents the ratio of female athletes to male athletes.

Male.per.Female: This feature represents the ratio of male athletes to female athletes.

Gold.per.total: This feature represents the proportion of gold medals won out of the total medals won.

Silver.per.total: This feature represents the proportion of silver medals won out of the total medals won.

Bronze.per.total: This feature represents the proportion of bronze medals won out of the total medals won.

The correlation between the type of medals awarded variables is +1, which indicates a perfect positive linear relationship between the two variables. In addition, a pairs panel plot observed a clear linear relationship between the type of medal awarded. Therefore, there may be a clear linear relationship between these variables. However, it is essential to remember that correlation does not necessarily imply causation and further analysis may be needed to determine the nature of the relationship between the variables.

Variable Selection

```
# Compute the null model
fitNull = lm(Total.medals ~ 1, data=data_cleaned)

# Compute the full model
fitFull = lm(Total.medals ~ ., data=data_cleaned)

# stepwise variable selection
fitStepwise = step(fitNull, scope = list(lower=fitNull, upper=fitFull),
                   direction="both", trace=F)

# summary of the model
summary(fitStepwise)
```

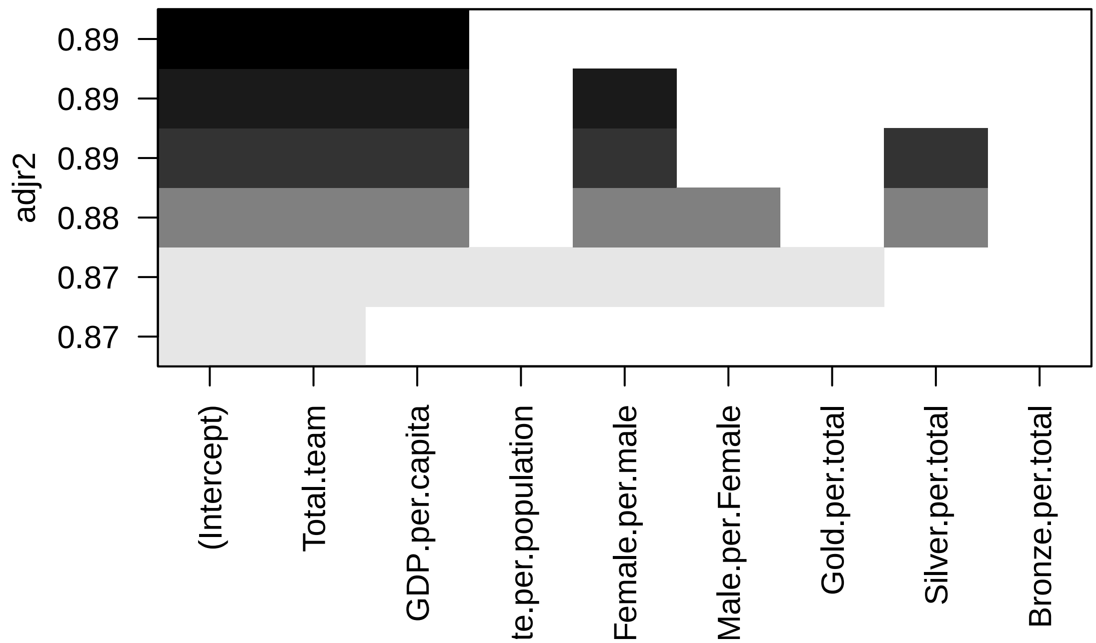
```
##
## Call:
## lm(formula = Total.medals ~ Total.team + GDP.per.capita, data = data_cleaned)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -25.0601  -2.8331  -0.0927   3.7950  23.8469
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)    -0.2154047  3.5057449  -0.061   0.9517
## Total.team      0.1858469  0.0180593  10.291 1.02e-08 ***
## GDP.per.capita -0.0004173  0.0001954  -2.135   0.0476 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10.74 on 17 degrees of freedom
## Multiple R-squared:  0.9046, Adjusted R-squared:  0.8934
## F-statistic: 80.62 on 2 and 17 DF,  p-value: 2.116e-09
```

```
# regsubsets with all the variables
fitAll = regsubsets(Total.medals ~ ., data=data_cleaned, nvmax=8, nbest = 1)
```

```
## Warning in leaps.setup(x, y, wt = wt, nbest = nbest, nvmax = nvmax, force.in =
## force.in, : 2 linear dependencies found
```

```
## Warning in leaps.setup(x, y, wt = wt, nbest = nbest, nvmax = nvmax, force.in =
## force.in, : nvmax reduced to 6
```

```
# plot
plot(fitAll, scale="adjr2")
```



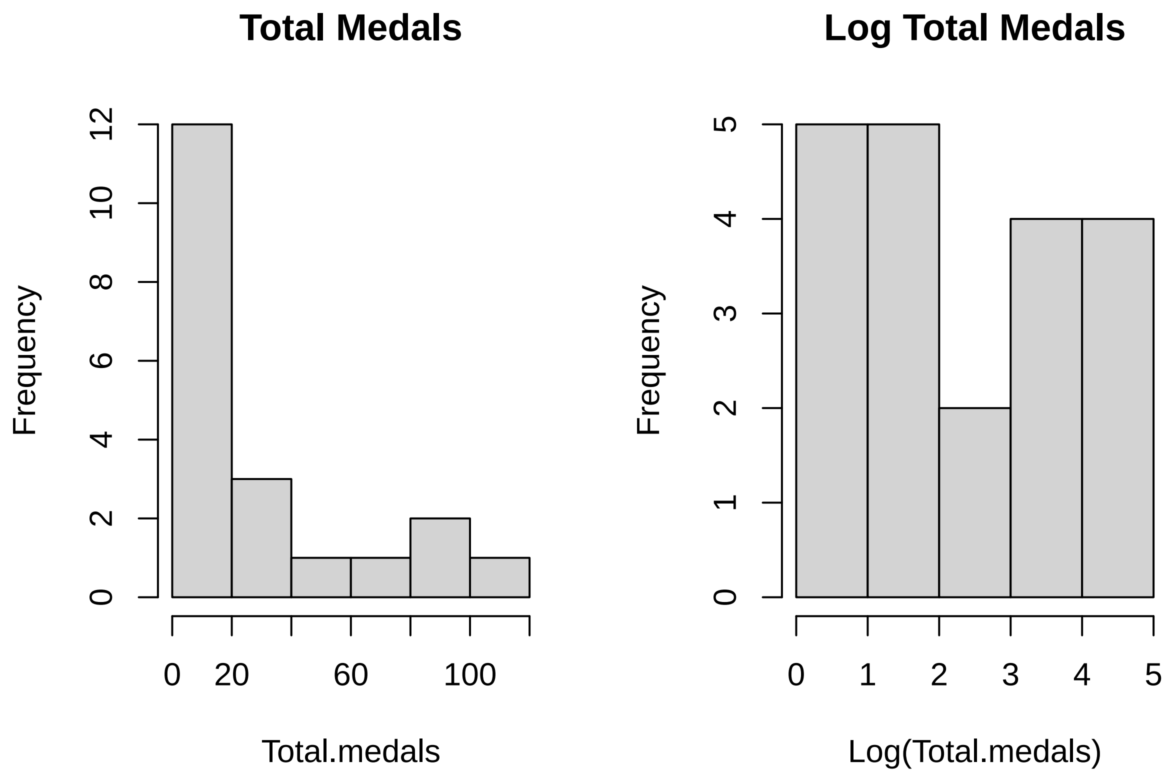
To determine the variables that are most important for predicting the number of metals awarded, stepwise and all-subsets analysis was performed to determine the most important variables for predicting the number of awarded medals. Both analyses indicated that only two variables, Total.team, and GDP.per.capita, are needed for predicting the number of awarded medals.

However, from all-subsets analysis, there is a linear dependency between the type of medal awarded variables. This suggests that the kind of medal awarded is not an independent predictor of the type number of awarded medals, as it is related to the other metal awarded variables linearly.

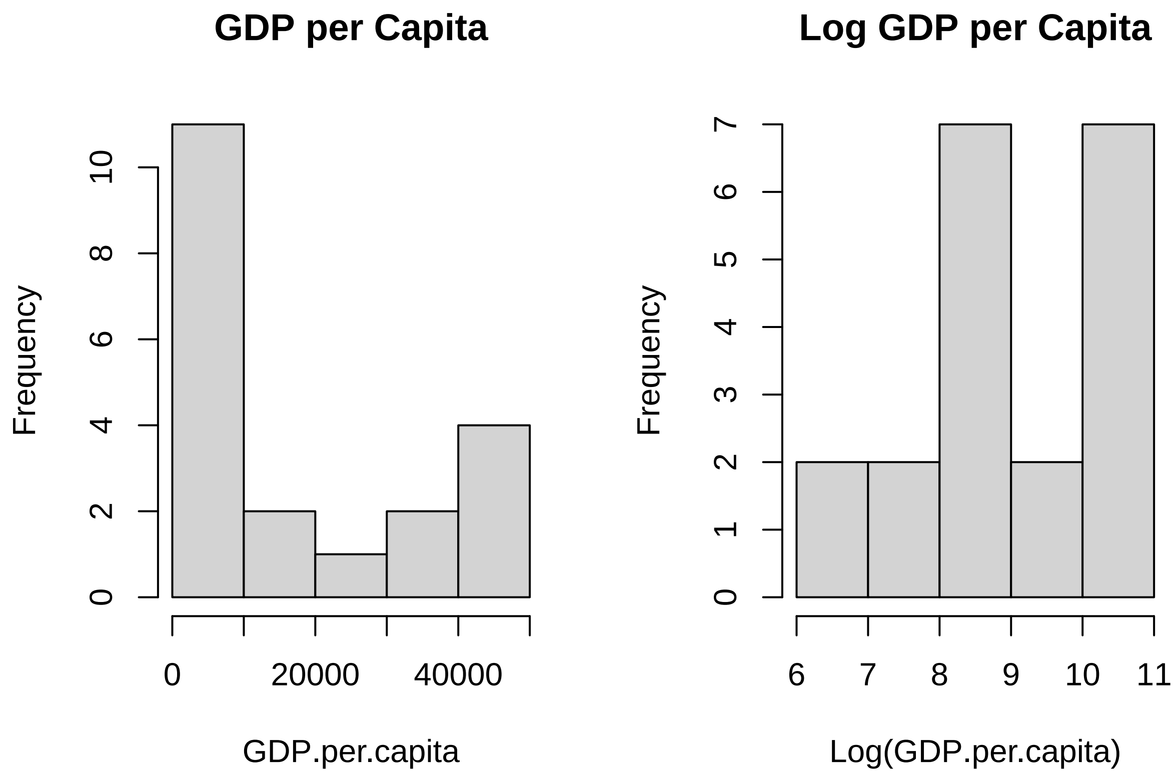
This confirmed the linear relationship by the correlation matrix and pair plot, which likely showed a positive strong correlation between the type of medal awarded. Therefore, it is essential to consider this relationship when building the model to remove redundant variables.

Histogram of two significant & response variables

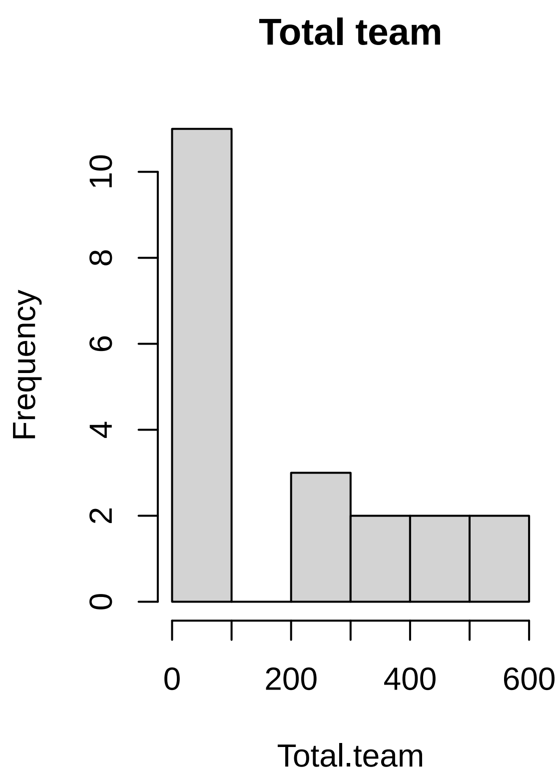
```
par(mfrow=c(1,2))
# plot histograms from variable selection analysis
hist(data_cleaned$Total.medals, main = "Total Medals", xlab = "Total.medals")
hist(log(data_cleaned$Total.medals), main = "Log Total Medals",
     xlab = "Log(Total.medals)")
```

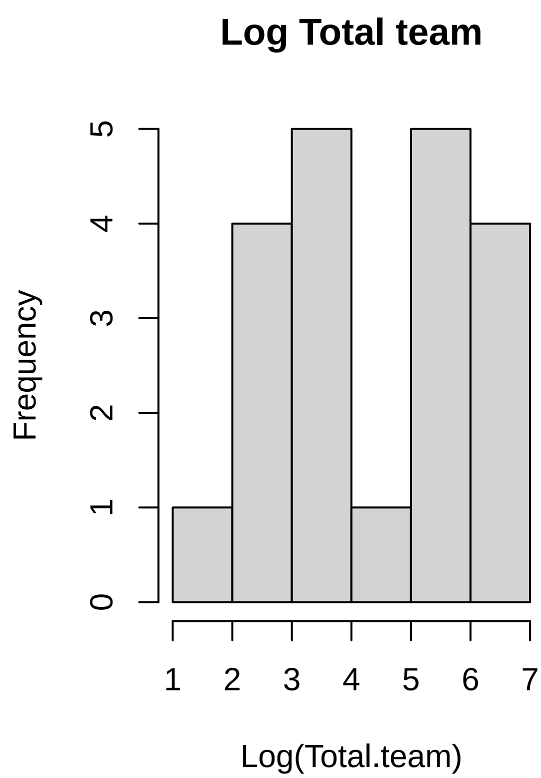



```
par(mfrow=c(1,2))
# histograms GDP.per.capita and log transform
hist(data_cleaned$GDP.per.capita, main = "GDP per Capita",
     xlab = "GDP.per.capita")
hist(log(data_cleaned$GDP.per.capita), main = "Log GDP per Capita",
     xlab = "Log(GDP.per.capita)")
```




```r
par(mfrow=c(1,2))
# histograms Total.team and log transform
hist(data_cleaned$Total.team, main = "Total team",
     xlab = "Total.team")
hist(log(data_cleaned$Total.team), main = "Log Total team",
     xlab = "Log(Total.team)")
```





Initial step was to log transform since all the indenpent and dependent variables are all skew to the right but log transformation did not help with improving the distribution of the variables per the histogram plot. However, it's important to keep in mind that the presence of skewness in the variables can still have an impact on the performance of the model. Therefore, it's important to consider alternative techniques to address skewness and ensure that the assumptions of the model are met.

Model

```
# initial model
initial_fit <- lm(formula = Total.medals ~ Total.team + GDP.per.capita,
                  data = data_cleaned)

# summary of medal_fit
summary(initial_fit)
```

```
##
## Call:
## lm(formula = Total.medals ~ Total.team + GDP.per.capita, data = data_cleaned)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -25.0601  -2.8331  -0.0927   3.7950  23.8469
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)    -0.2154047  3.5057449  -0.061   0.9517
```

```
## Total.team      0.1858469  0.0180593  10.291 1.02e-08 ***
## GDP.per.capita -0.0004173  0.0001954  -2.135   0.0476 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10.74 on 17 degrees of freedom
## Multiple R-squared:  0.9046, Adjusted R-squared:  0.8934
## F-statistic: 80.62 on 2 and 17 DF,  p-value: 2.116e-09
```

```r
# analysis of variance
anova(initial_fit)
```

```
## Analysis of Variance Table
##
## Response: Total.medals
##                Df  Sum Sq Mean Sq  F value    Pr(>F)
## Total.team      1 18085.2 18085.2 156.6721 5.265e-10 ***
## GDP.per.capita  1   526.2   526.2   4.5587    0.0476 *
## Residuals      17  1962.4   115.4
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

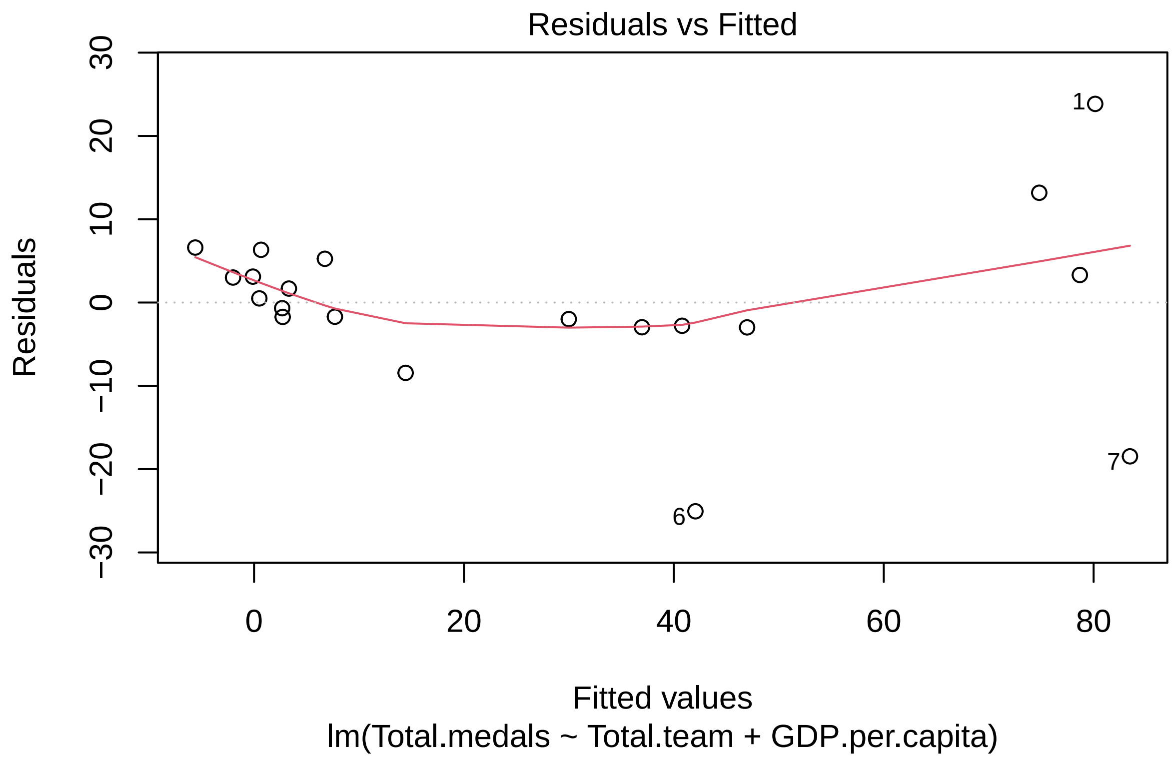
```r
# variance inflation factor (VIF)
vif(initial_fit)
```

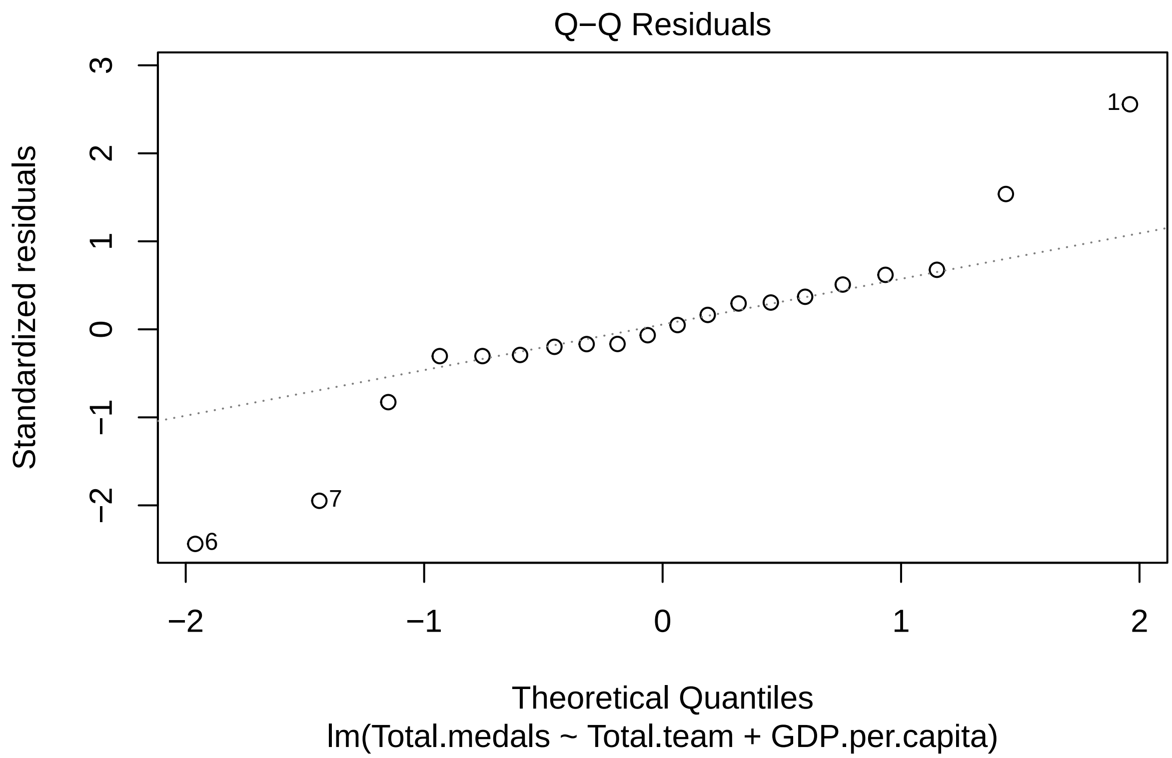
```
##     Total.team GDP.per.capita
##       2.038377       2.038377
```

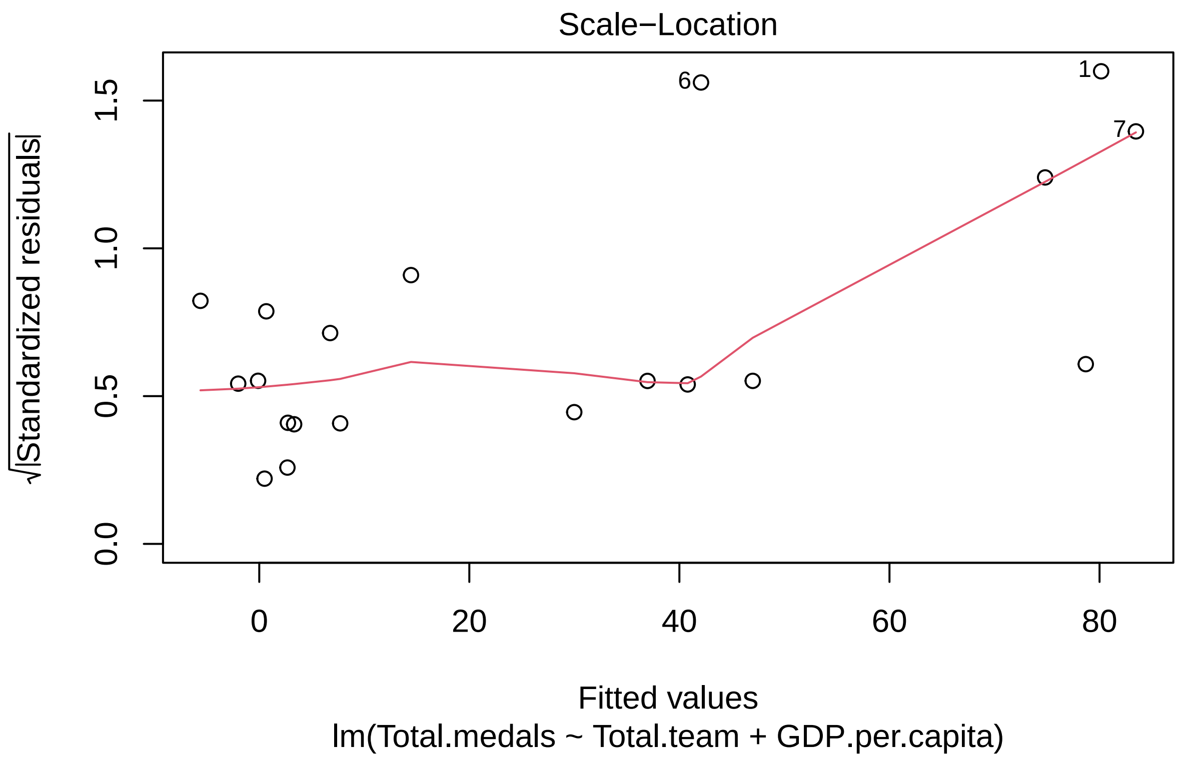
```r
# standardized coefficients
lm.beta(initial_fit)
```

```
##     Total.team GDP.per.capita
##      1.1005411     -0.2283341
```

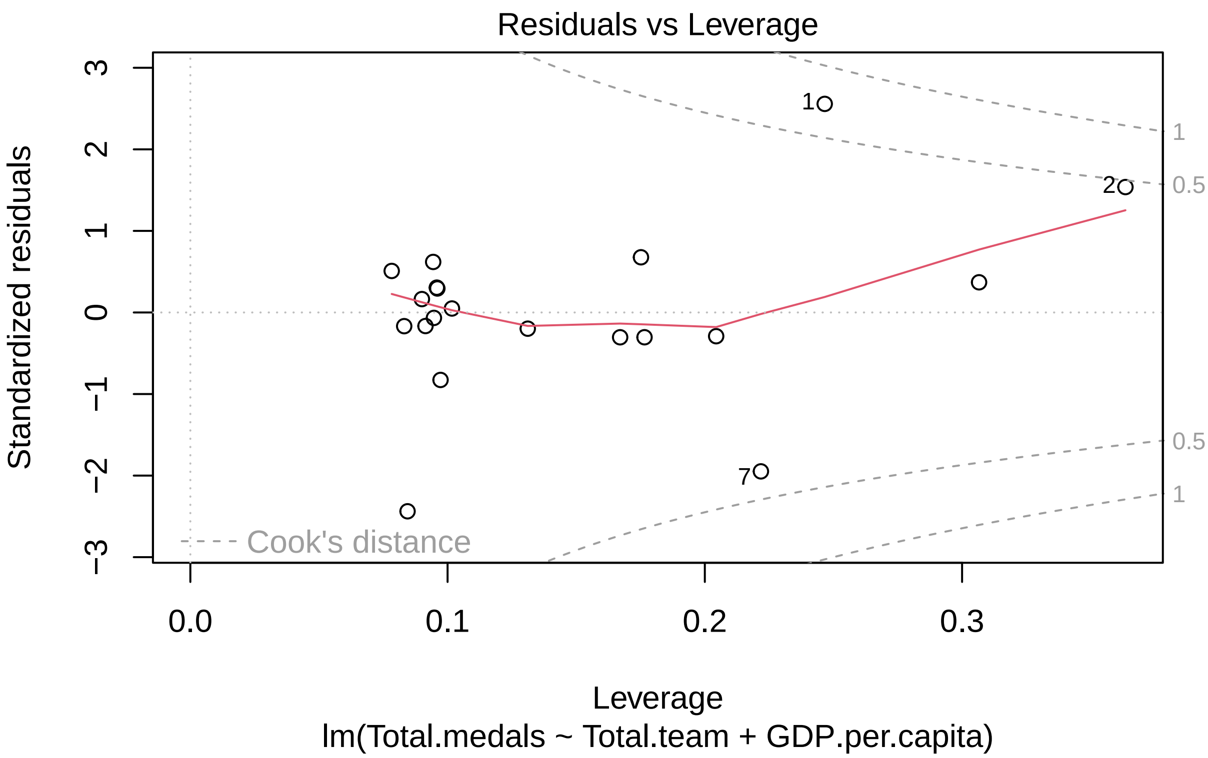
```r
# plot residuals
plot(initial_fit)
```


Residuals vs Fitted
Residuals
30
20
10
0
−10
−20
−30
0
20
40
60
80
1
6
7
Fitted values
lm(Total.medals ~ Total.team + GDP.per.capita)


Q−Q Residuals
Standardized residuals
Theoretical Quantiles
lm(Total.medals ~ Total.team + GDP.per.capita)
1
6
7
3
2
1
0
−1
−2
−2
−1
0
1
2



Scale−Location
√|Standardized residuals|
0.0
0.5
1.0
1.5
0
20
40
60
80
6
1
7
Fitted values
lm(Total.medals ~ Total.team + GDP.per.capita)



The `initial.fit` model produced a Multiple R-squared of 0.9046 and an Adjusted R-squared of 0.8934 which explains 90.46% of the variability of two independent variables (Total.team & GDP.per.capita) from the summary report. Also, the t-statistic for the predictor variables and the p-value corresponding to this test statistic is statistically significant. In this model, all the coefficients (Total.team and GDP.per.capita) have p values less than 0.05, indicating the null hypothesis that the coefficient is zero can be rejected; therefore, two independent variables are statistically significant.

An F-statistic was performed using ANOVA and determined that the two variables are statistically significant in the model in predicting the outcome variable, including variance inflation factor analysis confirmed there is minimum multicollinearity in our independent variables.

The `initial.fit` model residual versus fitted plot assumptions of linearity, homoscedasticity, and independence of errors are met because the points in the plot are randomly scattered around a horizontal line at zero. Also, in a normal probability plot, also known as a Q-Q plot (quantile-quantile plot), the differences between the observed values and the predicted values and the residuals are normally distributed with two outliers at the lower left and one on the top right. Lastly, examining the Standardized residuals vs. Leverage plot, which identify influential or unusual observations that may be impacting the regression model. Three possible points indicate influential observations that are having a disproportionate impact on the regression model, and one point where the standardized residual is close to -3, the four total points may need to be removed or further investigated.

Model Equation: E(Total.medals) = -0.2154047 + 0.1858469(Total.team) - 0.0004173(GDP.per.capita)

Standarized Beta Coefficients:

Total.team: 1.1005411

GDP.per.capita: -0.2283341

The regression analysis shows that higher Total.medals are associated with larger team size, and interestingly

GDP.per.capita penalizes the total medal count. Also, looking at the standardized beta coefficients confirm team size has significant importance rather than GDP per Capita.

Increasing the team size to impact the medal count may not be the most effective strategy. It is essential to consider other factors that may affect the team's performance, such as the level of training, experience, and skill of the athletes. Additionally, increasing the team size may also increase the cost of participation, which may only be feasible for some countries.

Regarding the gender difference in medal count, collecting data on the total events for male and female participants could be a good starting point to investigate this further. However, it is essential to be cautious when making any conclusions based on minor differences in the data, as they may be insignificant. It would be necessary to conduct a proper statistical analysis to determine whether the difference is statistically significant.

In summary, while the initial regression analysis provides some useful insights, it is crucial to keep in mind the data's limitations and consider collecting more data to increase the statistical power and reliability of any conclusions or predictions made. Additionally, it is essential to consider other factors that may impact the team's performance before making any significant strategic changes.

Problem 2 Housing Data

1. CRIM: per capita crime rate by town
2. ZN: proportion of residential land zoned for lots over 25,000 sq.ft.
3. INDUS: proportion of non-retail business acres per town
4. CHAS: Charles River dummy variable (= 1 if tract bounds river; 0 otherwise)
5. NOX: nitric oxides concentration (parts per 10 million)
6. RM: average number of rooms per dwelling
7. AGE: proportion of owner-occupied units built prior to 1940
8. DIS: weighted distances to five Boston employment centers
9. RAD: index of accessibility to radial highways
10. TAX: full-value property-tax rate per $10,000
11. PTRATIO: pupil-teacher ratio by town
12. LSTAT: % lower status of the population
13. MEDV: Median value of owner-occupied homes in $1,000's

```r
# load csv housing data to a dataframe
housing <- read.csv("housing.csv")

# check dimension of the dataframe & view
glimpse(housing)
```

```
## Rows: 506
## Columns: 13
## $ CRIM    <dbl> 0.00632, 0.02731, 0.02729, 0.03237, 0.06905, 0.02985, 0.08829,~
## $ ZN      <dbl> 18.0, 0.0, 0.0, 0.0, 0.0, 0.0, 12.5, 12.5, 12.5, 12.5, 12.5, 1~
## $ INDUS   <dbl> 2.31, 7.07, 7.07, 2.18, 2.18, 2.18, 7.87, 7.87, 7.87, 7.87, 7.~
## $ CHAS    <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ NOX     <dbl> 0.538, 0.469, 0.469, 0.458, 0.458, 0.458, 0.524, 0.524, 0.524,~
## $ RM      <dbl> 6.575, 6.421, 7.185, 6.998, 7.147, 6.430, 6.012, 6.172, 5.631,~
## $ AGE     <dbl> 65.2, 78.9, 61.1, 45.8, 54.2, 58.7, 66.6, 96.1, 100.0, 85.9, 9~
## $ DIS     <dbl> 4.0900, 4.9671, 4.9671, 6.0622, 6.0622, 6.0622, 5.5605, 5.9505~
## $ RAD     <int> 1, 2, 2, 3, 3, 3, 5, 5, 5, 5, 5, 5, 5, 4, 4, 4, 4, 4, 4, 4, 4,~
## $ TAX     <int> 296, 242, 242, 222, 222, 222, 311, 311, 311, 311, 311, 311, 31~
## $ PTRATIO <dbl> 15.3, 17.8, 17.8, 18.7, 18.7, 18.7, 15.2, 15.2, 15.2, 15.2, 15~
```

```
## $ LSTAT   <dbl> 4.98, 9.14, 4.03, 2.94, 5.33, 5.21, 12.43, 19.15, 29.93, 17.10~
## $ MEDV    <dbl> 24.0, 21.6, 34.7, 33.4, 36.2, 28.7, 22.9, 27.1, 16.5, 18.9, 15~
```

```r
# check for missing values
sum(is.na(housing))
```

```
## [1] 0
```

```r
# describe data excluding character columns
describe(housing)
```

```
##         vars   n   mean     sd median trimmed    mad    min    max  range  skew
## CRIM       1 506   3.61   8.60   0.26    1.68   0.33   0.01  88.98  88.97  5.19
## ZN         2 506  11.36  23.32   0.00    5.08   0.00   0.00 100.00 100.00  2.21
## INDUS      3 506  11.14   6.86   9.69   10.93   9.37   0.46  27.74  27.28  0.29
## CHAS       4 506   0.07   0.25   0.00    0.00   0.00   0.00   1.00   1.00  3.39
## NOX        5 506   0.55   0.12   0.54    0.55   0.13   0.38   0.87   0.49  0.72
## RM         6 506   6.28   0.70   6.21    6.25   0.51   3.56   8.78   5.22  0.40
## AGE        7 506  68.57  28.15  77.50   71.20  28.98   2.90 100.00  97.10 -0.60
## DIS        8 506   3.80   2.11   3.21    3.54   1.91   1.13  12.13  11.00  1.01
## RAD        9 506   9.55   8.71   5.00    8.73   2.97   1.00  24.00  23.00  1.00
## TAX       10 506 408.24 168.54 330.00  400.04 108.23 187.00 711.00 524.00  0.67
## PTRATIO   11 506  18.46   2.16  19.05   18.66   1.70  12.60  22.00   9.40 -0.80
## LSTAT     12 506  12.65   7.14  11.36   11.90   7.11   1.73  37.97  36.24  0.90
## MEDV      13 506  22.53   9.20  21.20   21.56   5.93   5.00  50.00  45.00  1.10
##         kurtosis   se
## CRIM       36.60 0.38
## ZN          3.95 1.04
## INDUS      -1.24 0.30
## CHAS        9.48 0.01
## NOX        -0.09 0.01
## RM          1.84 0.03
## AGE        -0.98 1.25
## DIS         0.46 0.09
## RAD        -0.88 0.39
## TAX        -1.15 7.49
## PTRATIO    -0.30 0.10
## LSTAT       0.46 0.32
## MEDV        1.45 0.41
```

```r
# summary of data
summary(housing)
```

```
##       CRIM                ZN             INDUS            CHAS
##  Min.   : 0.00632   Min.   :  0.00   Min.   : 0.46   Min.   :0.00000
##  1st Qu.: 0.08205   1st Qu.:  0.00   1st Qu.: 5.19   1st Qu.:0.00000
##  Median : 0.25651   Median :  0.00   Median : 9.69   Median :0.00000
##  Mean   : 3.61352   Mean   : 11.36   Mean   :11.14   Mean   :0.06917
##  3rd Qu.: 3.67708   3rd Qu.: 12.50   3rd Qu.:18.10   3rd Qu.:0.00000
##  Max.   :88.97620   Max.   :100.00   Max.   :27.74   Max.   :1.00000
##       NOX               RM             AGE              DIS
##  Min.   :0.3850   Min.   :3.561   Min.   :  2.90   Min.   : 1.130
```

```
##  1st Qu.:0.4490   1st Qu.:5.886   1st Qu.: 45.02   1st Qu.: 2.100
##  Median :0.5380   Median :6.208   Median : 77.50   Median : 3.207
##  Mean   :0.5547   Mean   :6.285   Mean   : 68.57   Mean   : 3.795
##  3rd Qu.:0.6240   3rd Qu.:6.623   3rd Qu.: 94.08   3rd Qu.: 5.188
##  Max.   :0.8710   Max.   :8.780   Max.   :100.00   Max.   :12.127
##       RAD              TAX           PTRATIO          LSTAT
##  Min.   : 1.000   Min.   :187.0   Min.   :12.60   Min.   : 1.73
##  1st Qu.: 4.000   1st Qu.:279.0   1st Qu.:17.40   1st Qu.: 6.95
##  Median : 5.000   Median :330.0   Median :19.05   Median :11.36
##  Mean   : 9.549   Mean   :408.2   Mean   :18.46   Mean   :12.65
##  3rd Qu.:24.000   3rd Qu.:666.0   3rd Qu.:20.20   3rd Qu.:16.95
##  Max.   :24.000   Max.   :711.0   Max.   :22.00   Max.   :37.97
##       MEDV
##  Min.   : 5.00
##  1st Qu.:17.02
##  Median :21.20
##  Mean   :22.53
##  3rd Qu.:25.00
##  Max.   :50.00
```

Initial Full Model

```r
# initial model with all the variables
MEDV_initial_model <- lm(MEDV ~ ., data = housing)

# summary of initial_model
summary(MEDV_initial_model)
```

```
##
## Call:
## lm(formula = MEDV ~ ., data = housing)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -15.1304  -2.7673  -0.5814   1.9414  26.2526
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  41.617270   4.936039   8.431 3.79e-16 ***
## CRIM         -0.121389   0.033000  -3.678 0.000261 ***
## ZN            0.046963   0.013879   3.384 0.000772 ***
## INDUS         0.013468   0.062145   0.217 0.828520
## CHAS          2.839993   0.870007   3.264 0.001173 **
## NOX         -18.758022   3.851355  -4.870 1.50e-06 ***
## RM            3.658119   0.420246   8.705  < 2e-16 ***
## AGE           0.003611   0.013329   0.271 0.786595
## DIS          -1.490754   0.201623  -7.394 6.17e-13 ***
## RAD           0.289405   0.066908   4.325 1.84e-05 ***
## TAX          -0.012682   0.003801  -3.337 0.000912 ***
## PTRATIO      -0.937533   0.132206  -7.091 4.63e-12 ***
## LSTAT        -0.552019   0.050659 -10.897  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

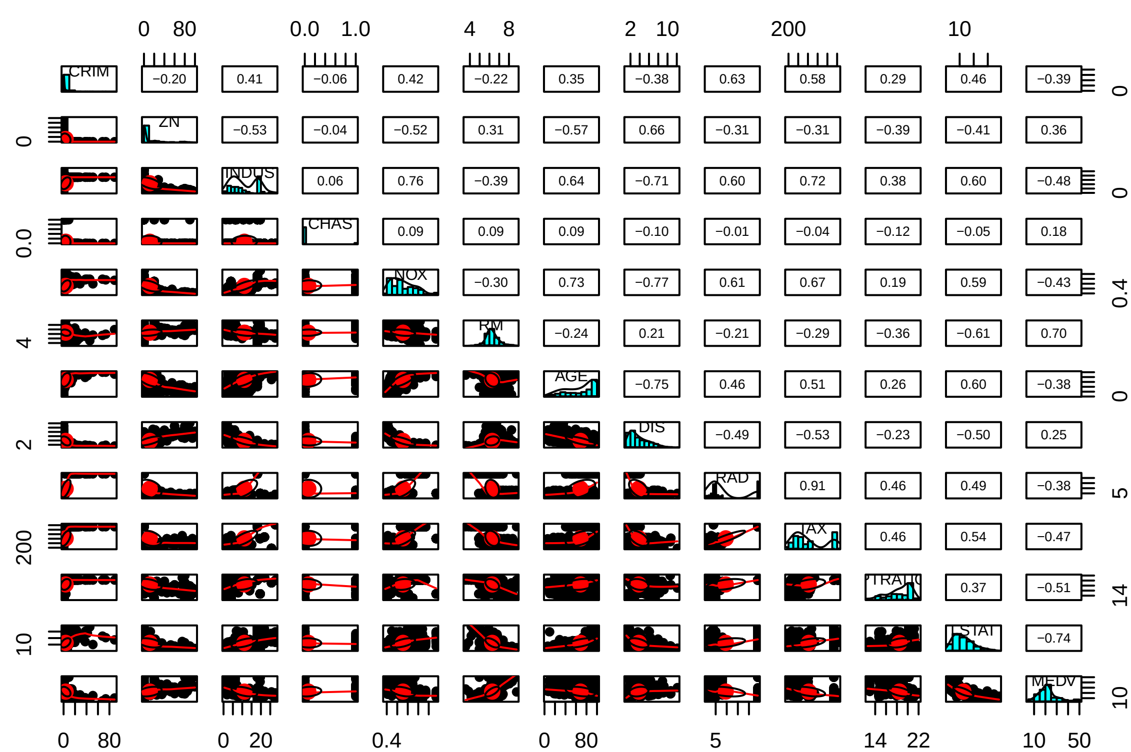
```
##
## Residual standard error: 4.798 on 493 degrees of freedom
## Multiple R-squared:  0.7343, Adjusted R-squared:  0.7278
## F-statistic: 113.5 on 12 and 493 DF,  p-value: < 2.2e-16
```

a. (5 points) Fit an initial linear regression model of MEDV based on all the other variables and report R2, Adjusted R2, the utility of the model (F-Test), the estimated coefficients, their standard errors, and statistical significance. Interpret your results. Treat the RAD ordinal variable as numeric.

The standard errors for each coefficient estimate measure the variability around this estimate for the regression slope. This value is used to calculate the t-statistic for the predictor variable and the p-value corresponding to this test statistic and to determine if the predictor variable was statistically significant. In this model, all but two of the coefficients (INDUS and AGE) have p values greater than 0.05, indicating null hypothesis that the coefficient is zero can not be rejected therefore it's not statistically significant.

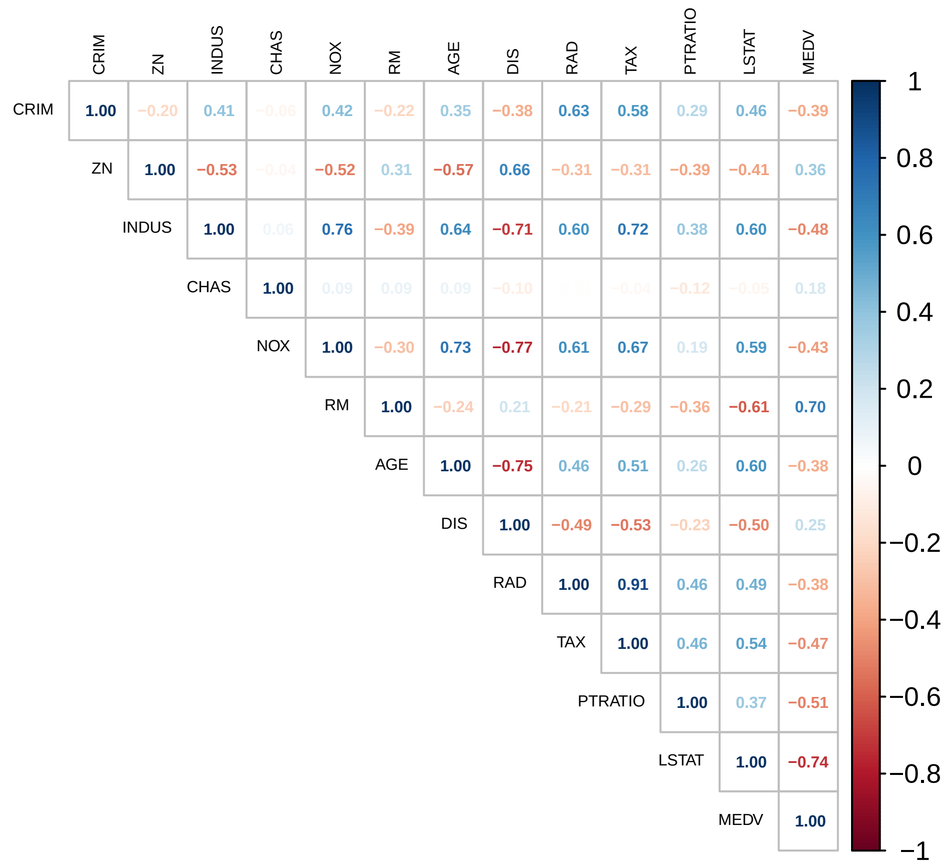
Finally, the multiple R-squared is 0.7343, indicating that the model explains about 73.43% of the variability. The F-statistic and associated p-value indicate the overall significance of the model, where smaller p-values indicate stronger evidence against the null hypothesis that all of the coefficients are zero. For example, in this model, the F-statistic is 113.5 with a p-value less than 2.2e-16, indicating strong evidence against the null hypothesis and overall statistical significance of the model.

```r
# pairs panels plot
pairs.panels(housing, pch=16)
```




```r
# correlation
corr_housing <- cor(housing)
```

```r
# plot correlation
corrplot(corr_housing, method = 'number', addCoef.col = 'green', tl.col = "black",
        type = 'upper', number.cex = 0.5, tl.cex = 0.5,)
```




```r
# second model with transformation
MEDV_transform_model <- lm(MEDV ~ CRIM + ZN + INDUS + CHAS + NOX + RM +
                    AGE + log(DIS) + log(RAD) + TAX + PTRATIO + sqrt(LSTAT),
                    data=housing)

summary(MEDV_transform_model)
```

```
##
## Call:
## lm(formula = MEDV ~ CRIM + ZN + INDUS + CHAS + NOX + RM + AGE +
##     log(DIS) + log(RAD) + TAX + PTRATIO + sqrt(LSTAT), data = housing)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -15.4850  -2.5740  -0.4802   2.0381  22.0802
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  56.203396   4.647158  12.094  < 2e-16 ***
## CRIM         -0.145004   0.029615  -4.896 1.33e-06 ***
```

```
## ZN            0.032752   0.012075   2.712 0.006911 **
## INDUS        -0.036807   0.056579  -0.651 0.515644
## CHAS          2.615187   0.791870   3.303 0.001028 **
## NOX         -23.038009   3.622651  -6.359 4.62e-10 ***
## RM            3.123663   0.389080   8.028 7.30e-15 ***
## AGE           0.003203   0.012555   0.255 0.798722
## log(DIS)     -7.863803   0.808318  -9.729  < 2e-16 ***
## log(RAD)      2.297957   0.458099   5.016 7.36e-07 ***
## TAX          -0.009793   0.002929  -3.343 0.000892 ***
## PTRATIO      -0.778286   0.118566  -6.564 1.33e-10 ***
## sqrt(LSTAT)  -4.959338   0.352636 -14.064  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.365 on 493 degrees of freedom
## Multiple R-squared:  0.7801, Adjusted R-squared:  0.7747
## F-statistic: 145.7 on 12 and 493 DF,  p-value: < 2.2e-16
```

b. (5 points) Plot the dataset in a scatterplot matrix and also the correlation with a corrplot. Interpret the result. Are there variables whose correlation with MEDV are weak? Are their variables whose relationship to MEDV are non-linear, or for which a log transform should be applied (look for a lot of samples on the axis with relatively few at high values)? Look for at least two transformations to apply that can increase the R2 value of the regression. Transform the variables, rerun the regression, and compare the results to the initial regression.

Among the 13 variables, eight variables are negatively correlated with MEDV, which means that if the value of these variables increases, the value of MEDV decreases. These variables are CRIM (per capita crime rate by town), INDUS (proportion of non-retail business acres per town), NOX (nitric oxides concentration (parts per 10 million)), AGE (proportion of owner-occupied units built prior to 1940), RAD (index of accessibility to radial highways), TAX (full-value property-tax rate per $10,000), PTRATIO (pupil-teacher ratio by town), and LSTAT (% lower status of the population).

On the other hand, four variables are positively correlated with MEDV, which means that if the value of these variables increases, the value of MEDV also increases. These variables are ZN (proportion of residential land zoned for lots over 25,000 sq.ft.), CHAS (Charles River dummy variable (1 if tract bounds river; 0 otherwise)), RM (average number of rooms per dwelling), and DIS (weighted distances to five Boston employment centers).

Out of these variables, RM (average number of rooms per dwelling) has a strong positive correlation with MEDV, which means that as the average number of rooms per dwelling increases, the median value of owner-occupied homes also increases. On the other hand, LSTAT (% lower status of the population) has a strong negative correlation with MEDV, which means that as the percentage of lower status of the population increases, the median value of owner-occupied homes decreases.

Although the DIS variable has a weak correlation with the response variable, it may still be an important predictor variable in a predictive model as it may capture some other aspect of the area that is relevant to the response variable. Additionally, the relationship DIS and CHAS may not be linear respect to MEDV.

The CHAS variable is a categorical variable that may not provide any meaningful correlation information with the response variable. However, it could still be an important predictor variable in a predictive model if it is relevant to the response variable.

Overall, it is important to consider all available variables when building a predictive model, even if they have weak correlations with the response variable or are categorical variables that may not provide direct correlation information.

The first full model has an R-squared of 0.7342, while the second transformed model has an R-squared of

0.7801. The increase in R-squared from the first full model to the second transformed model is approximately 5%, which is a significant improvement in explaining the variability of the data.

Both models were tested for statistical significance using an F-statistic, and it seems that both models were found to be significant. This means that the results are unlikely to be due to chance and are likely to represent a real relationship between the independent and dependent variables.

Overall, it appears that the second model, with the higher R-squared and transformed Adjusted R-squared, is a better fit for the data and can explain more of the variability in the dependent variable.

Feature Selection

```r
# new transformed features
housing_transform <- housing

# change column name & transform feature
housing_transform <- housing_transform %>% mutate(DIS = log(DIS)) %>%
  mutate(RAD = log(RAD)) %>% mutate(LSTAT = sqrt(LSTAT)) %>%
  rename(DIS_log = DIS) %>% rename(RAD_log = RAD) %>% rename(LSTAT_sqrt = LSTAT)

# stepwise variable selection
# Compute the null model
fit_null = lm(MEDV ~ 1, data=housing_transform)

# Compute the full model
fit_full = lm(MEDV ~ ., data=housing_transform)

# stepwise variable selection
fit_stepwise = step(fit_null, scope = list(lower=fit_null, upper=fit_full),
                direction="both", trace=F)

# summary of the model
summary(fit_stepwise)
```

```
##
## Call:
## lm(formula = MEDV ~ LSTAT_sqrt + RM + PTRATIO + DIS_log + NOX +
##     CRIM + RAD_log + TAX + CHAS + ZN, data = housing_transform)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -15.5370  -2.5161  -0.4242   2.0604  22.1446
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  56.067526   4.632756  12.102  < 2e-16 ***
## LSTAT_sqrt   -4.942820   0.325743 -15.174  < 2e-16 ***
## RM            3.172389   0.375029   8.459 3.06e-16 ***
## PTRATIO      -0.783605   0.117591  -6.664 7.13e-11 ***
## DIS_log      -7.811720   0.726833 -10.748  < 2e-16 ***
## NOX         -23.362258   3.481706  -6.710 5.34e-11 ***
## CRIM         -0.143502   0.029374  -4.885 1.40e-06 ***
## RAD_log       2.363082   0.444390   5.318 1.59e-07 ***
## TAX          -0.010611   0.002658  -3.993 7.52e-05 ***
## CHAS          2.576331   0.785487   3.280  0.00111 **
```

```
## ZN            0.033540   0.011775   2.848  0.00458 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.358 on 495 degrees of freedom
## Multiple R-squared:  0.7799, Adjusted R-squared:  0.7754
## F-statistic: 175.4 on 10 and 495 DF,  p-value: < 2.2e-16
```

```r
# standardized coefficients
lm.beta(fit_stepwise)
```

```
##   LSTAT_sqrt          RM     PTRATIO     DIS_log         NOX        CRIM
## -0.53053548  0.24235614 -0.18445602 -0.45827320 -0.29434963 -0.13420914
##     RAD_log         TAX        CHAS          ZN
##  0.22477759 -0.19445374  0.07114986  0.08505142
```

c. (5 points) Perform a feature selection on the transformed data by using the stepwise selection method of the regression analysis. Which variables are dropped in the stepwise selection model and how is the adjusted R2 affected? Evaluate the result in comparison to the full model.

The stepwise variable selection model dropped two variables (INDUS and AGE) because these variables were not statistically significant predictors of the response variable and therefore, were removed from the model.

The multiple R-squared value of the full model (with all variables) was 0.7343, while the adjusted R-squared was 0.7278. The multiple R-squared value of the stepwise model was 0.7799, while the adjusted R-squared was 0.7754. The adjusted R-squared value of the stepwise model was greater than the adjusted R-squared value of the full model, indicating that the stepwise variable selection model produced a better model by removing the two non-significant variables.

Overall, the stepwise variable selection model can be a useful tool for identifying the most important variables in a predictive model and improving its performance by removing irrelevant or non-significant variables. However, it is important to note that stepwise variable selection models can sometimes result in overfitting the data, so it is important to evaluate the model's performance on new, unseen data to ensure that it is generalizable.

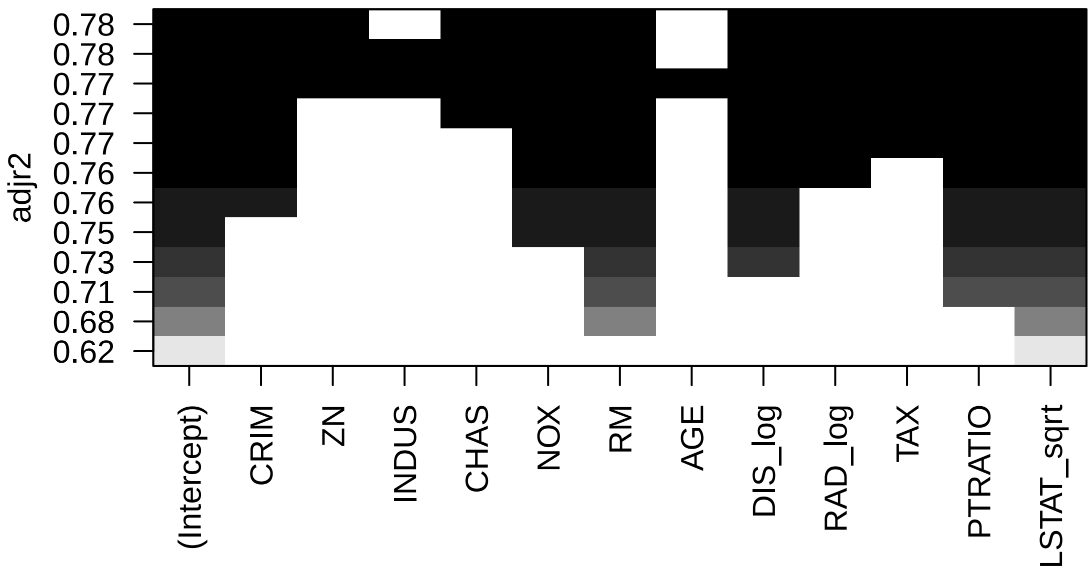
Subset Analysis

```r
# regsubsets with all the variables
fit_all = regsubsets(MEDV ~ ., data=housing_transform, nvmax=12, nbest = 1)

# summary
summary(fit_all)
```

```
## Subset selection object
## Call: regsubsets.formula(MEDV ~ ., data = housing_transform, nvmax = 12,
##     nbest = 1)
## 12 Variables  (and intercept)
##            Forced in Forced out
## CRIM           FALSE      FALSE
## ZN             FALSE      FALSE
## INDUS          FALSE      FALSE
## CHAS           FALSE      FALSE
## NOX            FALSE      FALSE
## RM             FALSE      FALSE
## AGE            FALSE      FALSE
```

```
## DIS_log        FALSE      FALSE
## RAD_log        FALSE      FALSE
## TAX            FALSE      FALSE
## PTRATIO        FALSE      FALSE
## LSTAT_sqrt     FALSE      FALSE
## 1 subsets of each size up to 12
## Selection Algorithm: exhaustive
##           CRIM ZN  INDUS CHAS NOX RM  AGE DIS_log RAD_log TAX PTRATIO
## 1  ( 1 )  " "  " " " "   " "  " " " " " " " "     " "     " " " "
## 2  ( 1 )  " "  " " " "   " "  " " "*" " " " "     " "     " " " "
## 3  ( 1 )  " "  " " " "   " "  " " "*" " " " "     " "     " " "*"
## 4  ( 1 )  " "  " " " "   " "  " " "*" " " "*"     " "     " " "*"
## 5  ( 1 )  " "  " " " "   " "  "*" "*" " " "*"     " "     " " "*"
## 6  ( 1 )  "*"  " " " "   " "  "*" "*" " " "*"     " "     " " "*"
## 7  ( 1 )  "*"  " " " "   " "  "*" "*" " " "*"     "*"     " " "*"
## 8  ( 1 )  "*"  " " " "   " "  "*" "*" " " "*"     "*"     "*" "*"
## 9  ( 1 )  "*"  " " " "   "*"  "*" "*" " " "*"     "*"     "*" "*"
## 10  ( 1 ) "*"  "*" " "   "*"  "*" "*" " " "*"     "*"     "*" "*"
## 11  ( 1 ) "*"  "*" "*"   "*"  "*" "*" " " "*"     "*"     "*" "*"
## 12  ( 1 ) "*"  "*" "*"   "*"  "*" "*" "*" "*"     "*"     "*" "*"
##           LSTAT_sqrt
## 1  ( 1 )  "*"
## 2  ( 1 )  "*"
## 3  ( 1 )  "*"
## 4  ( 1 )  "*"
## 5  ( 1 )  "*"
## 6  ( 1 )  "*"
## 7  ( 1 )  "*"
## 8  ( 1 )  "*"
## 9  ( 1 )  "*"
## 10  ( 1 ) "*"
## 11  ( 1 ) "*"
## 12  ( 1 ) "*"
```

```r
# plot
plot(fit_all, scale="adjr2")
```



d. (5 points) Perform an all-subsets analysis with "regsubsets" (set the "nvmax" parameter high enough that the search will include the regression with all the variables). Write out the model as an equation, plot and interpret the results (using the adjusted R2 value on the vertical axis). What variables are dropped in the "best" model and how does it compare to the stepwise model? Leave the parameter "nbest" at its default of 1 to reduce the complexity of the graph.

Model Equation E(MEDV) = Bo + B1(CRIM) + B2(ZN) + B3(CHAS) + B4(NOX) + B5(RM) + B6(log((DIS)) + B7(log(RAD)) + B8(TAX) + B9(PTRATIO) + B10(sqrt(LSTAT)) + Error

It seems that the all-subsets analysis aimed to find the best subset of variables that could explain the variation in the dependent variable, as measured by Adjusted R-squared which was roughly 0.78. The all-subsets analysis selected various combinations of independent variables covering all twelve variables according to the summary report, and the best model dropped the INDUS and AGE variables. The variable selection process resulted in identical variables for both the all-subsets analysis and the stepwise variable selection.

Find Parsimonious Model

```r
# Parsimonious Mode
par_model <- lm(MEDV ~ CRIM + NOX + RM + DIS_log + PTRATIO +
                  LSTAT_sqrt, data = housing_transform)

# summary of the model
summary(par_model)
```

```
##
## Call:
## lm(formula = MEDV ~ CRIM + NOX + RM + DIS_log + PTRATIO + LSTAT_sqrt,
```

```
##     data = housing_transform)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -13.7317  -2.8192  -0.4949   1.9278  23.8306
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  54.95673    4.64214  11.839  < 2e-16 ***
## CRIM         -0.11381    0.02761  -4.122 4.39e-05 ***
## NOX         -22.04750    3.30816  -6.665 7.04e-11 ***
## RM            3.51613    0.38524   9.127  < 2e-16 ***
## DIS_log      -6.96615    0.69956  -9.958  < 2e-16 ***
## PTRATIO      -0.88275    0.10486  -8.418 4.07e-16 ***
## LSTAT_sqrt   -5.06575    0.33719 -15.023  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.535 on 499 degrees of freedom
## Multiple R-squared:  0.7598, Adjusted R-squared:  0.7569
## F-statistic:   263 on 6 and 499 DF,  p-value: < 2.2e-16
```

```r
# standardized coefficients
lm.beta(par_model)
```

```
##       CRIM        NOX         RM    DIS_log    PTRATIO LSTAT_sqrt
## -0.1064394 -0.2777845  0.2686167 -0.4086679 -0.2077933 -0.5437306
```

e. (5 points, e.c. for DSC 324) Suppose you were trying to find parsimonious model (i.e. as few features as possible) to make the result easier to explain and use practically. Investigate the graph of the reg subsets result and determine if there is a model that reduces the number of variables significantly without significantly reducing predictive power (more than a percent or two). Explain your choice, and discuss which variables are included in the model? Compute that model with lm and compare the model practically with the stepwise model in terms of the effect of each variable on house value.

Examining all-subsets analysis using the adjusted R-squared plot to reduce additional independent variables while minimizing the loss in predictive power. Based on the adjusted R-squared plot, the new model was selected that included the variables CRIM, NOX, RM, DIS_log, PTRATIO, and LSTAT_sqrt while removing CHAS, ZN, TAX, and log(RAD).

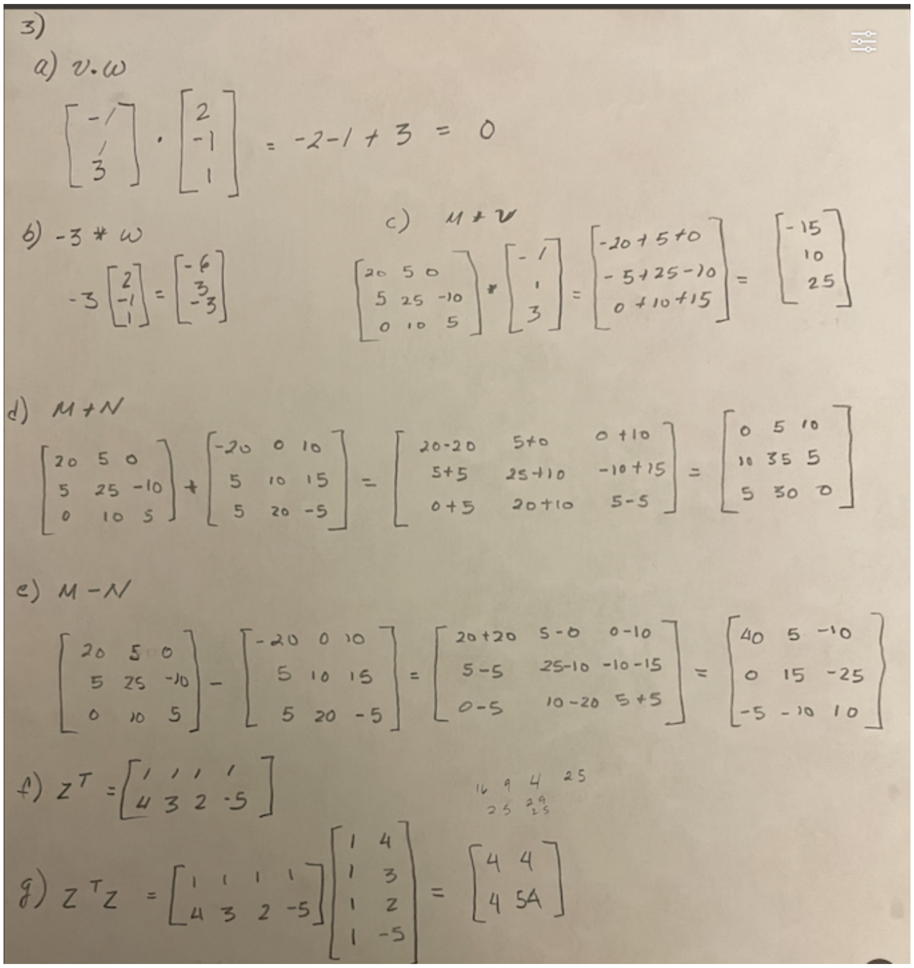
The new model produced a multiple R-squared value of 0.7598 and an adjusted R-squared value of 0.7569, which is slightly lower than the multiple R-squared value and adjusted R-squared value of the stepwise variable selection model. However, the decrease in predictive power is only about 2.01% for the multiple R-squared and 1.85% for the adjusted R-squared, which suggests that the new model is still a good fit for the data.

Additionally, it's worth noting that the standard error of all the coefficients increased slightly in the new model. Removing addtional four variables and retaining the other variables may have reduced the accuracy of the coefficients slightly. However, the increase in standard error is slight, and the overall model fit remains strong.

Overall, using an all-subsets analysis plot can be a helpful way to identify the simpler ones while minimizing the loss in predictive power. However, it's essential to carefully evaluate the model's performance on new, unseen data to ensure that it is generalizable.

3)

```r
# hand solution
knitr::include_graphics("hw.png")
```



3)

a) $v \cdot w$

$$\begin{bmatrix}-1\\1\\3\end{bmatrix} \cdot \begin{bmatrix}2\\-1\\1\end{bmatrix} = -2-1+3 = 0$$

b) $-3 * w$

$$-3\begin{bmatrix}2\\-1\\1\end{bmatrix} = \begin{bmatrix}-6\\3\\-3\end{bmatrix}$$

c) $M * v$

$$\begin{bmatrix}20 & 5 & 0\\5 & 25 & -10\\0 & 10 & 5\end{bmatrix} * \begin{bmatrix}-1\\1\\3\end{bmatrix} = \begin{bmatrix}-20+5+0\\-5+25-10\\0+10+15\end{bmatrix} = \begin{bmatrix}-15\\10\\25\end{bmatrix}$$

d) $M + N$

$$\begin{bmatrix}20 & 5 & 0\\5 & 25 & -10\\0 & 10 & 5\end{bmatrix} + \begin{bmatrix}-20 & 0 & 10\\5 & 10 & 15\\5 & 20 & -5\end{bmatrix} = \begin{bmatrix}20-20 & 5+0 & 0+10\\5+5 & 25+10 & -10+15\\0+5 & 20+10 & 5-5\end{bmatrix} = \begin{bmatrix}0 & 5 & 10\\10 & 35 & 5\\5 & 30 & 0\end{bmatrix}$$

e) $M - N$

$$\begin{bmatrix}20 & 5 & 0\\5 & 25 & -10\\0 & 10 & 5\end{bmatrix} - \begin{bmatrix}-20 & 0 & 10\\5 & 10 & 15\\5 & 20 & -5\end{bmatrix} = \begin{bmatrix}20+20 & 5-0 & 0-10\\5-5 & 25-10 & -10-15\\0-5 & 10-20 & 5+5\end{bmatrix} = \begin{bmatrix}40 & 5 & -10\\0 & 15 & -25\\-5 & -10 & 10\end{bmatrix}$$

f) $z^T = \begin{bmatrix}1 & 1 & 1 & 1\\4 & 3 & 2 & -5\end{bmatrix}$

g) $z^T z = \begin{bmatrix}1 & 1 & 1 & 1\\4 & 3 & 2 & -5\end{bmatrix}\begin{bmatrix}1 & 4\\1 & 3\\1 & 2\\1 & -5\end{bmatrix} = \begin{bmatrix}4 & 4\\4 & 54\end{bmatrix}$

R Solution 4)

```r
# a) matrix v & w
v <- matrix(c(-1,1,3))
w <- matrix(c(2,-1,1))

# v dot w
dot(v,w)
```

```
## [1] 0
```

```r
# b) scalar multiply matrix w
-3 * w
```

```
##      [,1]
## [1,]   -6
## [2,]    3
## [3,]   -3
```

```r
# c) matrix M
M <- matrix(c(20,5,0,5,25,-10,0,10,5), nrow = 3, ncol = 3, byrow = TRUE)

# M multiply v
M %*% v
```

```
##      [,1]
## [1,]  -15
## [2,]  -10
## [3,]   25
```

```r
# d) matrix N
N <- matrix(c(-20,0,10,5,10,15,5,20,-5), nrow = 3, ncol = 3, byrow = TRUE)

# M + N
M + N
```

```
##      [,1] [,2] [,3]
## [1,]    0    5   10
## [2,]   10   35    5
## [3,]    5   30    0
```

```r
# e) matrix M minus matrix N
M - N
```

```
##      [,1] [,2] [,3]
## [1,]   40    5  -10
## [2,]    0   15  -25
## [3,]   -5  -10   10
```

```r
# f) matrix Z
Z <-matrix(c(1,1,1,1,4,3,2,-5), nrow = 4, ncol = 2)

# transpose Z
t(Z)
```

```
##      [,1] [,2] [,3] [,4]
## [1,]    1    1    1    1
## [2,]    4    3    2   -5
```

```r
# g) transpose matrix Z multiply matrix Z
t(Z) %*% Z
```

```
##      [,1] [,2]
## [1,]    4    4
## [2,]    4   54
```