Weekly Summary Template

Advait Ashtikar

Table of contents

Tuesday, Feb 14

❗ TIL

Include a *very brief* summary of what you learnt in this class here.
Today, I learnt the following concepts in class:

1. Multicollinearity
2. Variable Selection
3. Shrinkage Estimators

Loading Libraries

```
library(tidyverse)
```

```
-- Attaching packages --------------------------------------- tidyverse 1.3.2 --
v ggplot2 3.4.1     v purrr   1.0.1
v tibble  3.1.8     v dplyr   1.1.0
v tidyr   1.3.0     v stringr 1.5.0
v readr   2.1.4     v forcats 1.0.0
-- Conflicts ------------------------------------------ tidyverse_conflicts() --
x dplyr::filter() masks stats::filter()
x dplyr::lag()    masks stats::lag()
```

```
library(ISLR2)
library(dplyr)
library(readr)
library(purrr)
library(glmnet)
```

```
Loading required package: Matrix

Attaching package: 'Matrix'

The following objects are masked from 'package:tidyr':

    expand, pack, unpack

Loaded glmnet 4.1-6
```

```
library(caret)
```

```
Loading required package: lattice

Attaching package: 'caret'

The following object is masked from 'package:purrr':

    lift
```

```r
library(car)
```

```
Loading required package: carData

Attaching package: 'car'

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

    recode

The following object is masked from 'package:purrr':

    some
```

```r
library(corrplot)
```

```
corrplot 0.92 loaded
```

In this class, we learnt about variable selection. For this, we will use **Boston housing dataset** which is described here:

```r
library(ISLR2)
attach(Boston)

df <- Boston
head(df)
```

```
     crim zn indus chas   nox    rm  age    dis rad tax ptratio lstat medv
1 0.00632 18  2.31    0 0.538 6.575 65.2 4.0900   1 296    15.3  4.98 24.0
2 0.02731  0  7.07    0 0.469 6.421 78.9 4.9671   2 242    17.8  9.14 21.6
3 0.02729  0  7.07    0 0.469 7.185 61.1 4.9671   2 242    17.8  4.03 34.7
4 0.03237  0  2.18    0 0.458 6.998 45.8 6.0622   3 222    18.7  2.94 33.4
5 0.06905  0  2.18    0 0.458 7.147 54.2 6.0622   3 222    18.7  5.33 36.2
6 0.02985  0  2.18    0 0.458 6.430 58.7 6.0622   3 222    18.7  5.21 28.7
```

Explanation of the Variables

The original data are 506 observations on 14 variables, `medv` being the target variable:

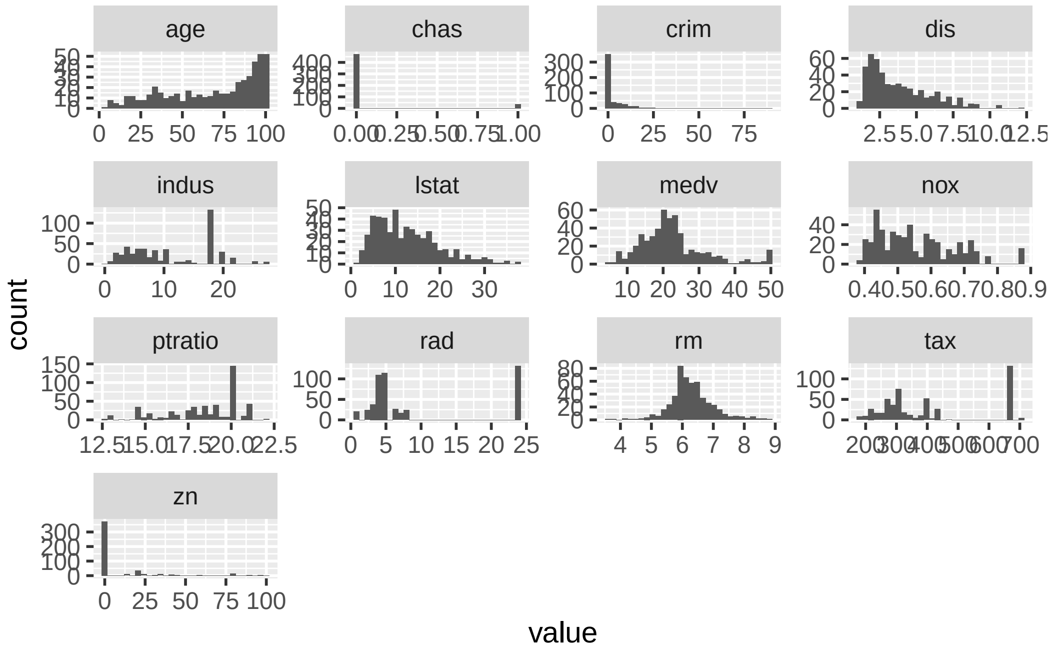
- `crim` per capita crime rate by town
- `zn` proportion of residential land zoned for lots over 25,000 sq.ft
- `indus` proportion of non-retail business acres per town
- `chas` Charles River dummy variable (= 1 if tract bounds river; 0 otherwise)
- `nox` nitric oxides concentration (parts per 10 million)
- `rm` average number of rooms per dwelling
- `age` proportion of owner-occupied units built prior to 1940
- `dis` weighted distances to five Boston employment centres
- `rad` index of accessibility to radial highways
- `tax` full - value property - tax rate per USD 10,000
- `ptratio` pupil - teacher ratio by town
- `lstat` percentage of lower status of the population
- `medv` median value of owner - occupied homes is USD 1000's

Exploratory Data Analysis:

Histogram:

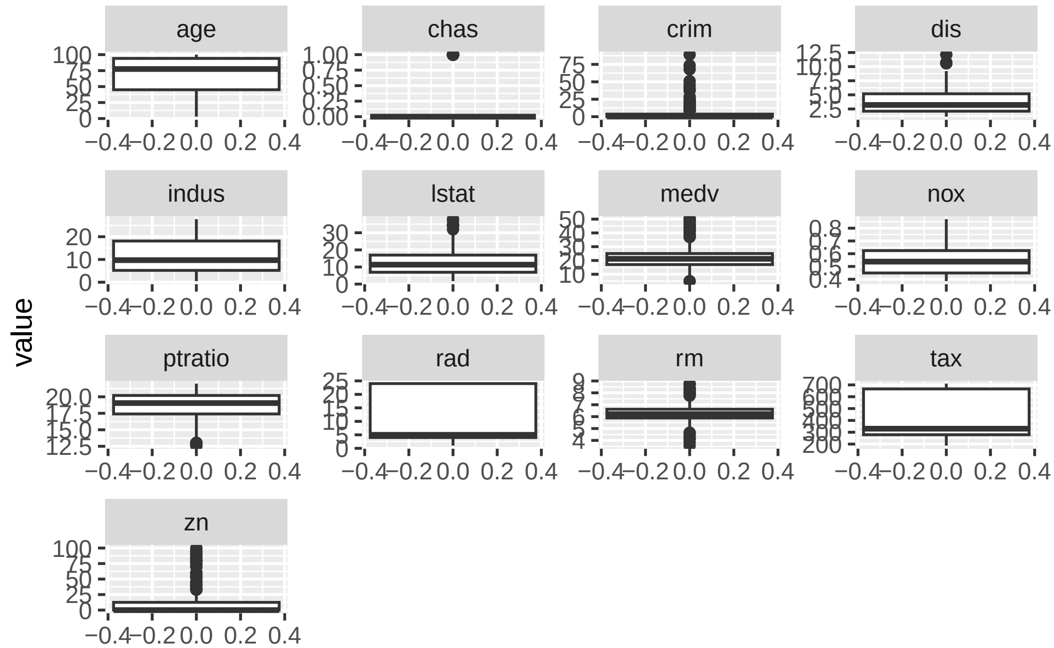
```
df %>%
  keep(is.numeric) %>%
  gather() %>%
  ggplot(aes(value)) +
  geom_histogram() +
  facet_wrap(~ key, scales = "free")
```

```
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



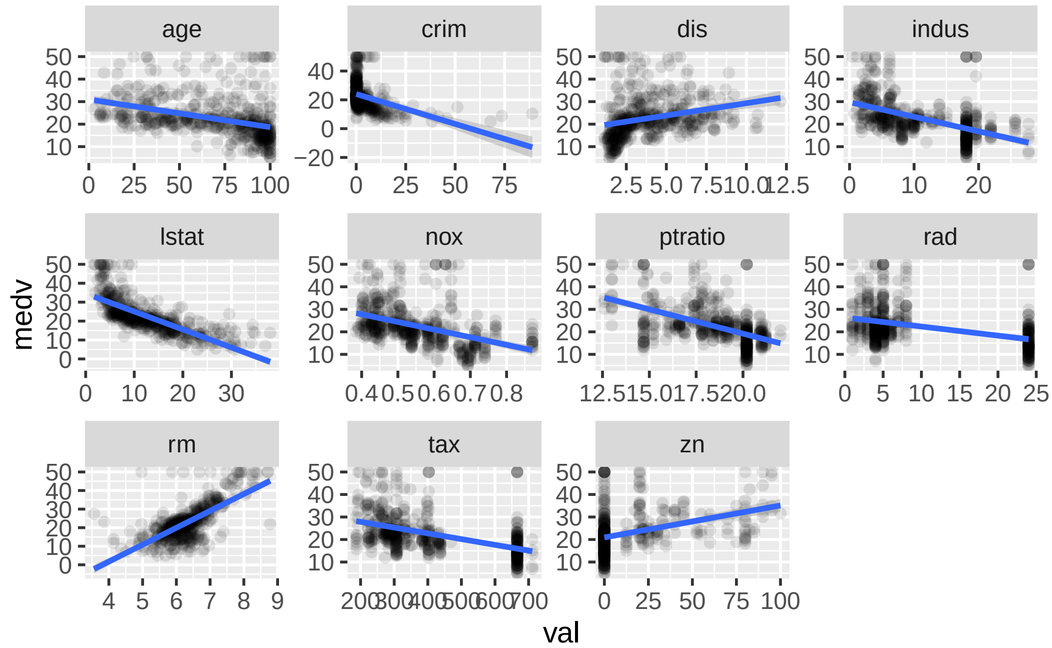
Boxplot:

```
df %>%
  keep(is.numeric) %>%
  gather() %>%
  ggplot(aes(y = value)) +
  geom_boxplot() +
  facet_wrap(~ key, scales = "free")
```



Scatterplot: Used to get a better understanding of the data

```
df %>%
  select(-chas) %>%
  gather(key, val, -medv) %>%
  ggplot(aes(x = val, y = medv)) +
  geom_point(alpha = 0.1) +
  stat_smooth(formula = y ~ x, method = "lm") +
  facet_wrap(~ key, scales = "free")
```



Regression Model

We begin by creating a regression model to predict `medv`

```
full_model <- lm(medv ~ ., df)
summary(full_model)
```

```
Call:
lm(formula = medv ~ ., data = df)

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
```

```r
broom::tidy(full_model)
```

```
# A tibble: 13 x 5
   term         estimate std.error statistic  p.value
   <chr>           <dbl>     <dbl>     <dbl>    <dbl>
 1 (Intercept)  41.6       4.94         8.43  3.79e-16
 2 crim         -0.121     0.0330      -3.68  2.61e- 4
 3 zn            0.0470    0.0139       3.38  7.72e- 4
 4 indus         0.0135    0.0621       0.217 8.29e- 1
 5 chas          2.84      0.870        3.26  1.17e- 3
 6 nox         -18.8       3.85        -4.87  1.50e- 6
 7 rm            3.66      0.420        8.70  4.81e-17
 8 age           0.00361   0.0133       0.271 7.87e- 1
 9 dis          -1.49      0.202       -7.39  6.17e-13
10 rad           0.289     0.0669       4.33  1.84e- 5
11 tax          -0.0127    0.00380     -3.34  9.12e- 4
12 ptratio      -0.938     0.132       -7.09  4.63e-12
13 lstat        -0.552     0.0507     -10.9   6.39e-25
```

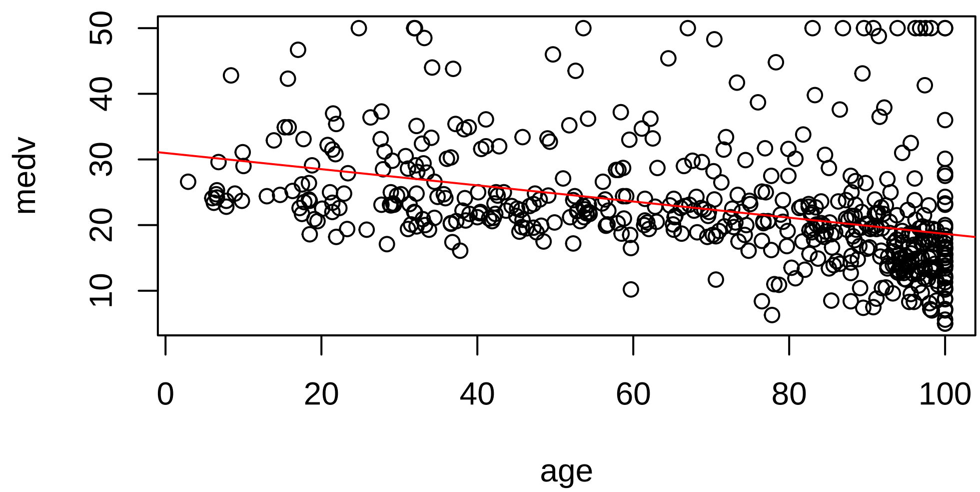
We can see that most of the variables are significant. However, notably

> `age` and `indus` are not significant predictors of `medv`

Is this true?

Plot and Regression Modelfor `age`

```r
plot(medv ~ age, df)
abline(lm(medv ~ age), col = "red")
```



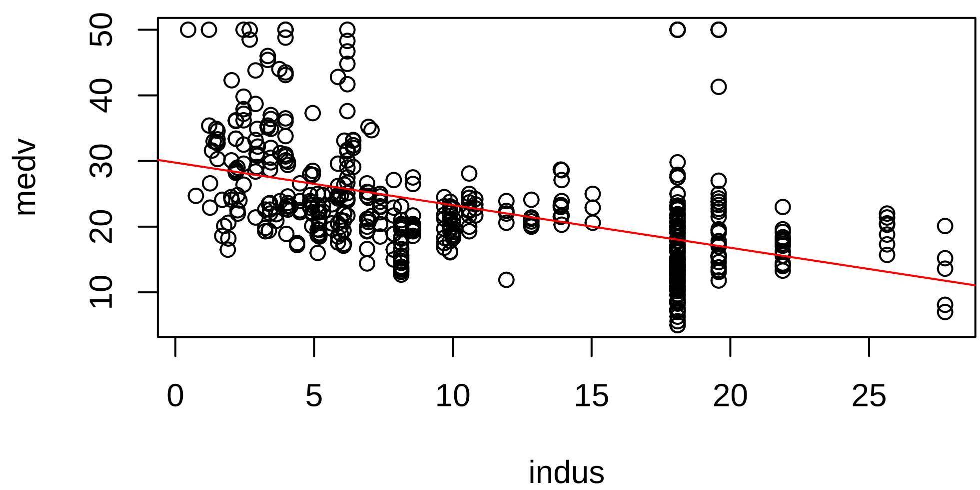

```r
model_age <- lm(medv ~ age, df)
summary(model_age)
```

```
Call:
lm(formula = medv ~ age, data = df)

Residuals:
    Min      1Q  Median      3Q     Max
-15.097  -5.138  -1.958   2.397  31.338

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 30.97868    0.99911  31.006   <2e-16 ***
age         -0.12316    0.01348  -9.137   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 8.527 on 504 degrees of freedom
```

```
Multiple R-squared:  0.1421,    Adjusted R-squared:  0.1404
F-statistic: 83.48 on 1 and 504 DF,  p-value: < 2.2e-16
```

Plot and Regression Model for `indus`

```r
plot(medv ~ indus, df)
abline(lm(medv ~ indus), col = "red")
```




```r
model_indus <- lm(medv ~ indus, df)
summary(model_indus)
```

```
Call:
lm(formula = medv ~ indus, data = df)

Residuals:
    Min      1Q  Median      3Q     Max 
-13.017  -4.917  -1.457   3.180  32.943 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
```

```
(Intercept) 29.75490    0.68345   43.54   <2e-16 ***
indus       -0.64849    0.05226  -12.41   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 8.057 on 504 degrees of freedom
Multiple R-squared:  0.234, Adjusted R-squared:  0.2325
F-statistic:   154 on 1 and 504 DF,  p-value: < 2.2e-16
```

Correlation Table

```
R <- df %>%
  keep(is.numeric) %>%
  cor()
R
```

```
               crim          zn       indus         chas         nox
crim     1.00000000 -0.20046922  0.40658341 -0.055891582  0.42097171
zn      -0.20046922  1.00000000 -0.53382819 -0.042696719 -0.51660371
indus    0.40658341 -0.53382819  1.00000000  0.062938027  0.76365145
chas    -0.05589158 -0.04269672  0.06293803  1.000000000  0.09120281
nox      0.42097171 -0.51660371  0.76365145  0.091202807  1.00000000
rm      -0.21924670  0.31199059 -0.39167585  0.091251225 -0.30218819
age      0.35273425 -0.56953734  0.64477851  0.086517774  0.73147010
dis     -0.37967009  0.66440822 -0.70802699 -0.099175780 -0.76923011
rad      0.62550515 -0.31194783  0.59512927 -0.007368241  0.61144056
tax      0.58276431 -0.31456332  0.72076018 -0.035586518  0.66802320
ptratio  0.28994558 -0.39167855  0.38324756 -0.121515174  0.18893268
lstat    0.45562148 -0.41299457  0.60379972 -0.053929298  0.59087892
medv    -0.38830461  0.36044534 -0.48372516  0.175260177 -0.42732077
                 rm         age         dis          rad         tax    ptratio
crim    -0.21924670  0.35273425 -0.37967009  0.625505145  0.58276431  0.2899456
zn       0.31199059 -0.56953734  0.66440822 -0.311947826 -0.31456332 -0.3916785
indus   -0.39167585  0.64477851 -0.70802699  0.595129275  0.72076018  0.3832476
chas     0.09125123  0.08651777 -0.09917578 -0.007368241 -0.03558652 -0.1215152
nox     -0.30218819  0.73147010 -0.76923011  0.611440563  0.66802320  0.1889327
rm       1.00000000 -0.24026493  0.20524621 -0.209846668 -0.29204783 -0.3555015
age     -0.24026493  1.00000000 -0.74788054  0.456022452  0.50645559  0.2615150
dis      0.20524621 -0.74788054  1.00000000 -0.494587930 -0.53443158 -0.2324705
rad     -0.20984667  0.45602245 -0.49458793  1.000000000  0.91022819  0.4647412
tax     -0.29204783  0.50645559 -0.53443158  0.910228189  1.00000000  0.4608530
```

```
ptratio -0.35550149  0.26151501 -0.23247054  0.464741179  0.46085304  1.0000000
lstat   -0.61380827  0.60233853 -0.49699583  0.488676335  0.54399341  0.3740443
medv     0.69535995 -0.37695457  0.24992873 -0.381626231 -0.46853593 -0.5077867
             lstat       medv
crim     0.4556215 -0.3883046
zn      -0.4129946  0.3604453
indus    0.6037997 -0.4837252
chas    -0.0539293  0.1752602
nox      0.5908789 -0.4273208
rm      -0.6138083  0.6953599
age      0.6023385 -0.3769546
dis     -0.4969958  0.2499287
rad      0.4886763 -0.3816262
tax      0.5439934 -0.4685359
ptratio  0.3740443 -0.5077867
lstat    1.0000000 -0.7376627
medv    -0.7376627  1.0000000
```

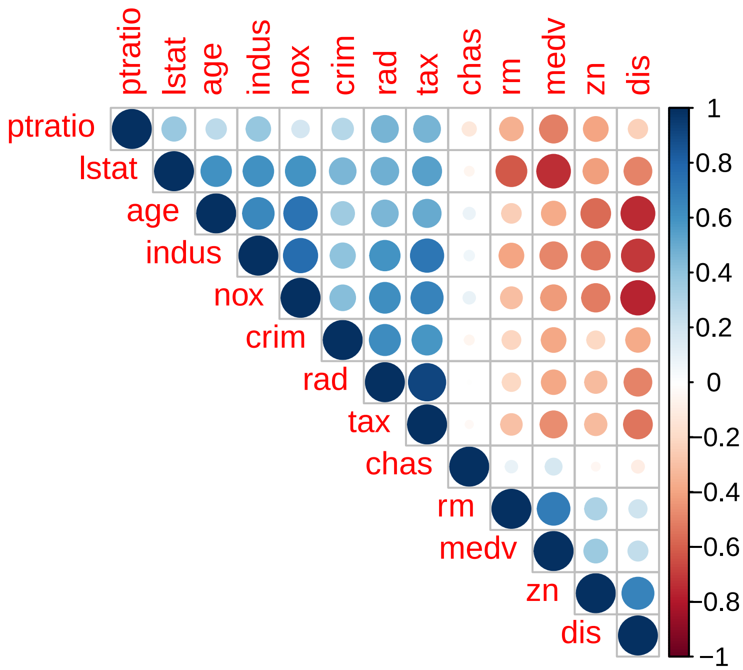
In a correlation table, we are selecting all the numeric values, where every single value is telling what the correlation with every other variable in data frame.

Q. What is an admissible correlation value?

> An admissible correlation value lies between **-1** and **1**.

A good way to visualize correlation is using `corrplot()`

```
library(corrplot)
corrplot(R, type = "upper", order = "hclust")
```



- From the plot we can see that, variables `indus` and `age` are fairly negatively correlated to the `medv` variable
- We can also see that, except `chas` variable, every other variable has some correlation with the other variables

```
new_cols <- colnames(df)[-c(5, 13)]
model <- lm(medv ~., df %>%
              select(-c(indus, nox, dis)))
summary(model)
```

```
Call:
lm(formula = medv ~ ., data = df %>% select(-c(indus, nox, dis)))

Residuals:
     Min       1Q   Median       3Q      Max 
-16.9388  -3.0974  -0.7082   1.8472  28.3443 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept) 21.655695   4.215323   5.137 4.01e-07 ***
crim        -0.091908   0.034722  -2.647 0.008380 ** 
zn           0.008794   0.012670   0.694 0.487957    
```

```
chas          2.952830   0.913519   3.232 0.001309 **
rm            4.100202   0.439135   9.337  < 2e-16 ***
age           0.020892   0.012195   1.713 0.087315 .
rad           0.251852   0.067890   3.710 0.000231 ***
tax          -0.012434   0.003469  -3.584 0.000371 ***
ptratio      -0.886594   0.129206  -6.862 2.03e-11 ***
lstat        -0.573951   0.053177 -10.793  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 5.084 on 496 degrees of freedom
Multiple R-squared:  0.6999,    Adjusted R-squared:  0.6944
F-statistic: 128.5 on 9 and 496 DF,  p-value: < 2.2e-16
```

Variance Inflation Factors

The **variance inflation factor (VIF)** is the ratio of the variance of estimating some parameter in a model that includes multiple other terms (parameters) by the variance of a model constructed using only one term.

If the standard error increases, then the significance of the variable decreases.

```r
library(car)
vif_model <- lm(medv ~ ., df)
vif(vif_model) %>%
  knitr::kable()
```

	x
crim	1.767486
zn	2.298459
indus	3.987181
chas	1.071168
nox	4.369093
rm	1.912532
age	3.088232
dis	3.954037
rad	7.445301
tax	9.002158
ptratio	1.797060
lstat	2.870776

A high inflation factor is any factor that is greater than **2**.

Stepwise Regression

The process of selecting variables that are relatively more important than the other variables is known as **stepwise regression**.

```
null_model <- lm(medv ~ 1, df)
full_model <- lm(medv ~ ., df)
```

The `null_model` does not contain any variable in a data frame.

The `full_model` contains all the variables in a data frame.

```
library(caret)
forward_model <- step(null_model,
                      direction = "forward",
                      scope = formula(full_model))
```

```
Start:  AIC=2246.51
medv ~ 1

          Df Sum of Sq   RSS    AIC
+ lstat    1   23243.9 19472 1851.0
+ rm       1   20654.4 22062 1914.2
+ ptratio  1   11014.3 31702 2097.6
+ indus    1    9995.2 32721 2113.6
+ tax      1    9377.3 33339 2123.1
+ nox      1    7800.1 34916 2146.5
+ crim     1    6440.8 36276 2165.8
+ rad      1    6221.1 36495 2168.9
+ age      1    6069.8 36647 2171.0
+ zn       1    5549.7 37167 2178.1
+ dis      1    2668.2 40048 2215.9
+ chas     1    1312.1 41404 2232.7
<none>                 42716 2246.5

Step:  AIC=1851.01
medv ~ lstat

          Df Sum of Sq   RSS    AIC
+ rm       1    4033.1 15439 1735.6
```

```
+ ptratio  1    2670.1 16802 1778.4
+ chas     1     786.3 18686 1832.2
+ dis      1     772.4 18700 1832.5
+ age      1     304.3 19168 1845.0
+ tax      1     274.4 19198 1845.8
+ zn       1     160.3 19312 1848.8
+ crim     1     146.9 19325 1849.2
+ indus    1      98.7 19374 1850.4
<none>                 19472 1851.0
+ rad      1      25.1 19447 1852.4
+ nox      1       4.8 19468 1852.9

Step:  AIC=1735.58
medv ~ lstat + rm

          Df Sum of Sq   RSS    AIC
+ ptratio  1   1711.32 13728 1678.1
+ chas     1    548.53 14891 1719.3
+ tax      1    425.16 15014 1723.5
+ dis      1    351.15 15088 1725.9
+ crim     1    311.42 15128 1727.3
+ rad      1    180.45 15259 1731.6
+ indus    1     61.09 15378 1735.6
<none>                 15439 1735.6
+ zn       1     56.56 15383 1735.7
+ age      1     20.18 15419 1736.9
+ nox      1     14.90 15424 1737.1

Step:  AIC=1678.13
medv ~ lstat + rm + ptratio

        Df Sum of Sq   RSS    AIC
+ dis    1    499.08 13229 1661.4
+ chas   1    377.96 13350 1666.0
+ crim   1    122.52 13606 1675.6
+ age    1     66.24 13662 1677.7
<none>               13728 1678.1
+ tax    1     44.36 13684 1678.5
+ nox    1     24.81 13703 1679.2
+ zn     1     14.96 13713 1679.6
+ rad    1      6.07 13722 1679.9
+ indus  1      0.83 13727 1680.1
```

```
Step:  AIC=1661.39
medv ~ lstat + rm + ptratio + dis

        Df Sum of Sq   RSS    AIC
+ nox    1    759.56 12469 1633.5
+ chas   1    267.43 12962 1653.1
+ indus  1    242.65 12986 1654.0
+ tax    1    240.34 12989 1654.1
+ crim   1    233.54 12995 1654.4
+ zn     1    144.81 13084 1657.8
+ age    1     61.36 13168 1661.0
<none>                13229 1661.4
+ rad    1     22.40 13206 1662.5

Step:  AIC=1633.47
medv ~ lstat + rm + ptratio + dis + nox

        Df Sum of Sq   RSS    AIC
+ chas   1    328.27 12141 1622.0
+ zn     1    151.71 12318 1629.3
+ crim   1    141.43 12328 1629.7
+ rad    1     53.48 12416 1633.3
<none>                12469 1633.5
+ indus  1     17.10 12452 1634.8
+ tax    1     10.50 12459 1635.0
+ age    1      0.25 12469 1635.5

Step:  AIC=1621.97
medv ~ lstat + rm + ptratio + dis + nox + chas

        Df Sum of Sq   RSS    AIC
+ zn     1   164.406 11977 1617.1
+ crim   1   116.330 12025 1619.1
+ rad    1    58.556 12082 1621.5
<none>                12141 1622.0
+ indus  1    26.274 12115 1622.9
+ tax    1     4.187 12137 1623.8
+ age    1     2.331 12139 1623.9

Step:  AIC=1617.07
medv ~ lstat + rm + ptratio + dis + nox + chas + zn

        Df Sum of Sq   RSS    AIC
```

```
+ crim   1   170.902 11806 1611.8
<none>               11977 1617.1
+ tax    1    31.773 11945 1617.7
+ rad    1    28.311 11948 1617.9
+ indus  1    27.377 11949 1617.9
+ age    1     0.071 11977 1619.1

Step:  AIC=1611.8
medv ~ lstat + rm + ptratio + dis + nox + chas + zn + crim

        Df Sum of Sq   RSS    AIC
+ rad    1   155.006 11651 1607.1
<none>               11806 1611.8
+ indus  1    24.957 11781 1612.7
+ tax    1     1.418 11804 1613.7
+ age    1     0.178 11806 1613.8

Step:  AIC=1607.11
medv ~ lstat + rm + ptratio + dis + nox + chas + zn + crim +
    rad

        Df Sum of Sq   RSS    AIC
+ tax    1   298.573 11352 1596.0
<none>               11651 1607.1
+ indus  1    44.346 11606 1607.2
+ age    1     0.581 11650 1609.1

Step:  AIC=1595.98
medv ~ lstat + rm + ptratio + dis + nox + chas + zn + crim +
    rad + tax

        Df Sum of Sq   RSS    AIC
<none>               11352 1596.0
+ age    1    1.6865 11350 1597.9
+ indus  1    1.0784 11351 1597.9
```

```r
summary(forward_model)
```

```
Call:
lm(formula = medv ~ lstat + rm + ptratio + dis + nox + chas +
    zn + crim + rad + tax, data = df)
```

```
Residuals:
     Min       1Q   Median       3Q      Max 
-15.1814  -2.7625  -0.6243   1.8448  26.3920 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept)  41.451747   4.903283   8.454 3.18e-16 ***
lstat        -0.546509   0.047442 -11.519  < 2e-16 ***
rm            3.672957   0.409127   8.978  < 2e-16 ***
ptratio      -0.930961   0.130423  -7.138 3.39e-12 ***
dis          -1.515951   0.187675  -8.078 5.08e-15 ***
nox         -18.262427   3.565247  -5.122 4.33e-07 ***
chas          2.871873   0.862591   3.329 0.000935 ***
zn            0.046191   0.013673   3.378 0.000787 ***
crim         -0.121665   0.032919  -3.696 0.000244 ***
rad           0.283932   0.063945   4.440 1.11e-05 ***
tax          -0.012292   0.003407  -3.608 0.000340 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 4.789 on 495 degrees of freedom
Multiple R-squared:  0.7342,    Adjusted R-squared:  0.7289 
F-statistic: 136.8 on 10 and 495 DF,  p-value: < 2.2e-16
```

> AIC is like a replacement for R^2
>
> Unlike R^2, where a higher value is better, we prefer to have a low AIC value
>
> Based on `lstat` is the best model, as it has the lowest value

Forward Selection: In this form of stepwise regression we keep building our model from 0 (or a low value), and add variables, until we reach a stage where our AIC ends to be high.

```
backward_model <- step(full_model,
                       direction = "backward",
                       scope = formula(full_model))
```

```
Start:  AIC=1599.85
medv ~ crim + zn + indus + chas + nox + rm + age + dis + rad + 
    tax + ptratio + lstat

          Df Sum of Sq   RSS    AIC
```

```
- indus    1      1.08 11350 1597.9
- age      1      1.69 11351 1597.9
<none>                 11349 1599.8
- chas     1    245.31 11595 1608.7
- tax      1    256.28 11606 1609.2
- zn       1    263.59 11613 1609.5
- crim     1    311.49 11661 1611.6
- rad      1    430.71 11780 1616.7
- nox      1    546.10 11896 1621.6
- ptratio  1   1157.70 12507 1647.0
- dis      1   1258.52 12608 1651.1
- rm       1   1744.36 13094 1670.2
- lstat    1   2733.54 14083 1707.0

Step:  AIC=1597.9
medv ~ crim + zn + chas + nox + rm + age + dis + rad + tax + 
    ptratio + lstat

          Df Sum of Sq   RSS    AIC
- age      1      1.69 11352 1596.0
<none>                 11350 1597.9
- chas     1    251.21 11602 1607.0
- zn       1    262.99 11614 1607.5
- tax      1    299.68 11650 1609.1
- crim     1    313.07 11664 1609.7
- rad      1    453.61 11804 1615.7
- nox      1    574.23 11925 1620.9
- ptratio  1   1168.01 12518 1645.5
- dis      1   1333.19 12684 1652.1
- rm       1   1750.50 13101 1668.5
- lstat    1   2743.21 14094 1705.4

Step:  AIC=1595.98
medv ~ crim + zn + chas + nox + rm + dis + rad + tax + ptratio + 
    lstat

          Df Sum of Sq   RSS    AIC
<none>                 11352 1596.0
- chas     1    254.21 11606 1605.2
- zn       1    261.75 11614 1605.5
- tax      1    298.57 11651 1607.1
- crim     1    313.27 11666 1607.8
- rad      1    452.16 11804 1613.7
```

```
- nox      1    601.74 11954 1620.1
- ptratio  1   1168.51 12521 1643.5
- dis      1   1496.35 12848 1656.6
- rm       1   1848.38 13201 1670.3
- lstat    1   3043.23 14395 1714.2
```

```r
summary(backward_model)
```

```

Call:
lm(formula = medv ~ crim + zn + chas + nox + rm + dis + rad +
    tax + ptratio + lstat, data = df)

Residuals:
     Min       1Q   Median       3Q      Max
-15.1814  -2.7625  -0.6243   1.8448  26.3920

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  41.451747   4.903283   8.454 3.18e-16 ***
crim         -0.121665   0.032919  -3.696 0.000244 ***
zn            0.046191   0.013673   3.378 0.000787 ***
chas          2.871873   0.862591   3.329 0.000935 ***
nox         -18.262427   3.565247  -5.122 4.33e-07 ***
rm            3.672957   0.409127   8.978  < 2e-16 ***
dis          -1.515951   0.187675  -8.078 5.08e-15 ***
rad           0.283932   0.063945   4.440 1.11e-05 ***
tax          -0.012292   0.003407  -3.608 0.000340 ***
ptratio      -0.930961   0.130423  -7.138 3.39e-12 ***
lstat        -0.546509   0.047442 -11.519  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 4.789 on 495 degrees of freedom
Multiple R-squared:  0.7342,    Adjusted R-squared:  0.7289
F-statistic: 136.8 on 10 and 495 DF,  p-value: < 2.2e-16
```

Another way to do the same, is using **Backward Selection**. In this we start with the `full_model`, and start removing variable, until we see a decrease in the *AIC* value. At this point, if we remove any more variables, the *AIC* value would increase.

> In this case, both **forward** and **backward** models have given the same result. This may not always be the case.

- Another option for the `direction` in the `step()` function is `both` this is a hybrid of both **forward** and **backward** selection.

Thursday, Feb 16

> **!** TIL
>
> Include a *very brief* summary of what you learnt in this class here.
> Today, I learnt the following concepts in class:
>
> 1. Item 1
> 2. Item 2
> 3. Item 3