Лабораторная работа 4

Могильников Дмитрий

2022-11-30

Задание 1

Используйте датасет из Лабораторной работы 3.

Загрузим датасет и выведем его через head()

```
options(width = 100)
library("MASS")
boston_df <- Boston

head(boston_df)
```

```
##      crim zn indus chas   nox    rm  age    dis rad tax ptratio  black lstat medv
## 1 0.00632 18  2.31    0 0.538 6.575 65.2 4.0900   1 296    15.3 396.90  4.98 24.0
## 2 0.02731  0  7.07    0 0.469 6.421 78.9 4.9671   2 242    17.8 396.90  9.14 21.6
## 3 0.02729  0  7.07    0 0.469 7.185 61.1 4.9671   2 242    17.8 392.83  4.03 34.7
## 4 0.03237  0  2.18    0 0.458 6.998 45.8 6.0622   3 222    18.7 394.63  2.94 33.4
## 5 0.06905  0  2.18    0 0.458 7.147 54.2 6.0622   3 222    18.7 396.90  5.33 36.2
## 6 0.02985  0  2.18    0 0.458 6.430 58.7 6.0622   3 222    18.7 394.12  5.21 28.7
```

Таблица с описанием для каждой переменной в проедставленном датасете:

Feature Variable	Description
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	nitrogen 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 mean of distances to five Boston employment centres.
RAD	index of accessibility to radial highways.
TAX	full-value property-tax rate per $10,000.
PTRATIO	pupil-teacher ratio by town.
BLACK	1000(Bk - 0.63)^2 where Bk is the proportion of blacks by town.
LSTAT	lower status of the population (percent).
MEDV	median value of owner-occupied homes in $1000s.

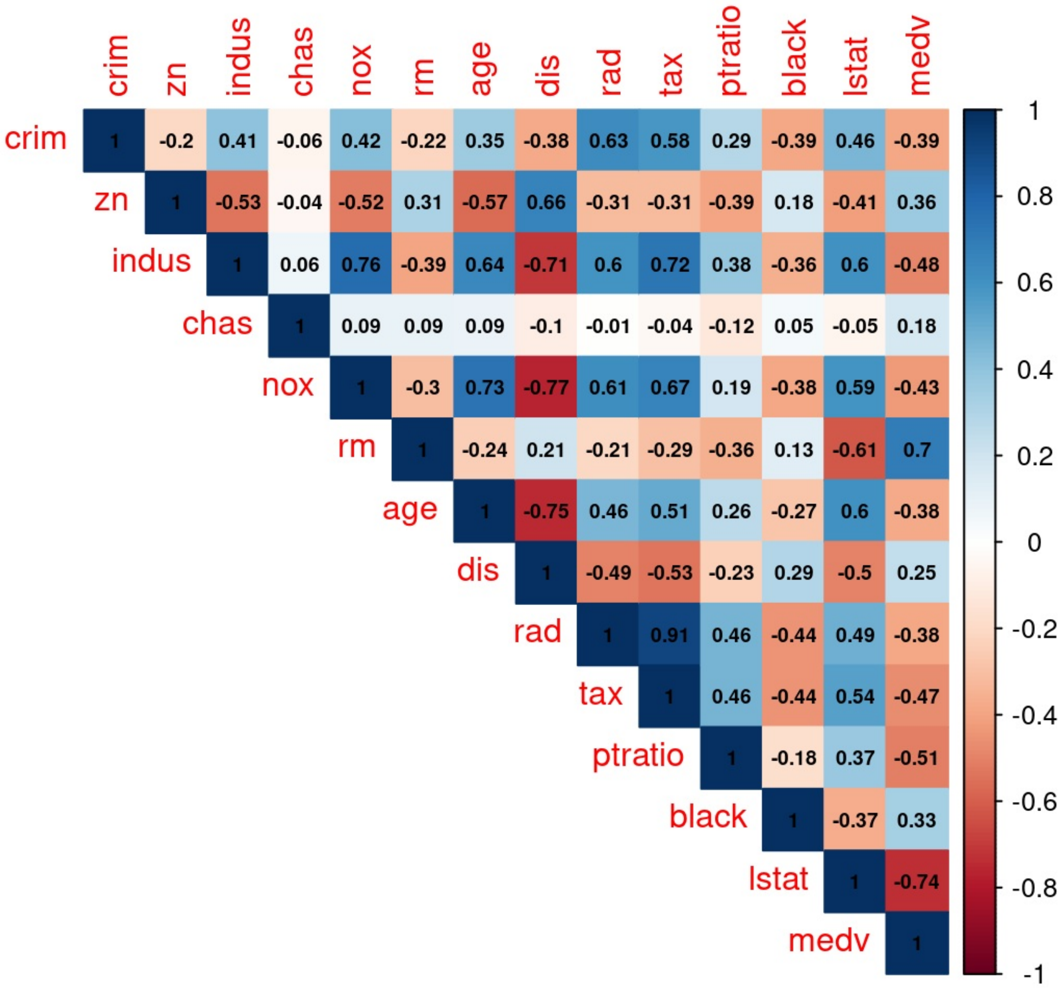
Задание 2

Повторно выведите корреляционную матрицу по всем переменным.

```
attach(boston_df)
library(ggplot2)
library(corrplot)
```

```
## corrplot 0.92 loaded
```

```
corrplot(cor(boston_df),
         method = "color",
         addCoef.col = 1,
         type="upper",
         number.cex = 0.6)
```



Задание 3

Выберите зависимую и независимую переменные (y и x), постройте парную линейную регрессию, выведите график наблюдаемых значений и полученной прямой.

Коэффициент корреляции варьируется от -1 до 1. Если значение близко к 1, это означает, что между двумя переменными существует сильная положительная корреляция. Когда он близок к -1, переменные имеют сильную отрицательную корреляцию.

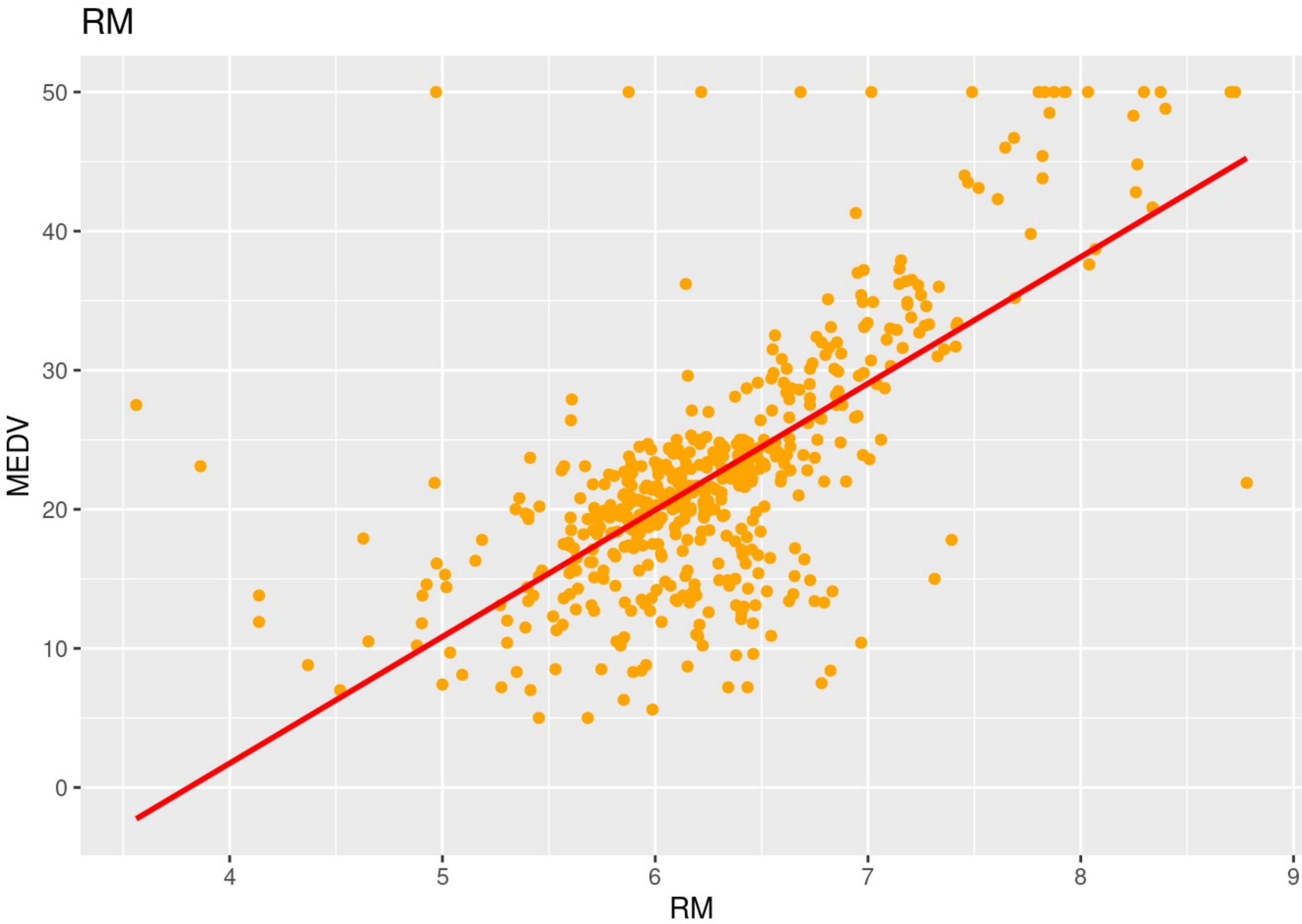
В качестве зависимой переменной выберем MEDV(медианная стоимость домов), чтобы получить соответствие в модели линейной регрессии, мы выбираем те функции, которые имеют высокую корреляцию с нашей целевой переменной MEDV. Основываясь на приведенных выше наблюдениях, в качестве независимой переменной выберем переменные RM и LSTAT, которые имеют сильную положительную и отрицательную корелляции соответственно.

```
fit_rm <-lm(medv~rm)
summary(fit_rm)
```

```
##
## Call:
## lm(formula = medv ~ rm)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -23.346  -2.547   0.090   2.986  39.433
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -34.671      2.650  -13.08   <2e-16 ***
## rm             9.102      0.419   21.72   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.616 on 504 degrees of freedom
## Multiple R-squared:  0.4835, Adjusted R-squared:  0.4825
## F-statistic: 471.8 on 1 and 504 DF,  p-value: < 2.2e-16
```

```
ggplot(data = boston_df, aes(x = rm, y = medv)) +
  geom_point(color = "orange") +
  labs(title = "RM",
       x = "RM",
       y = "MEDV") +
  geom_smooth(method = lm, se = FALSE, color = "red")
```

```
## `geom_smooth()` using formula = 'y ~ x'
```

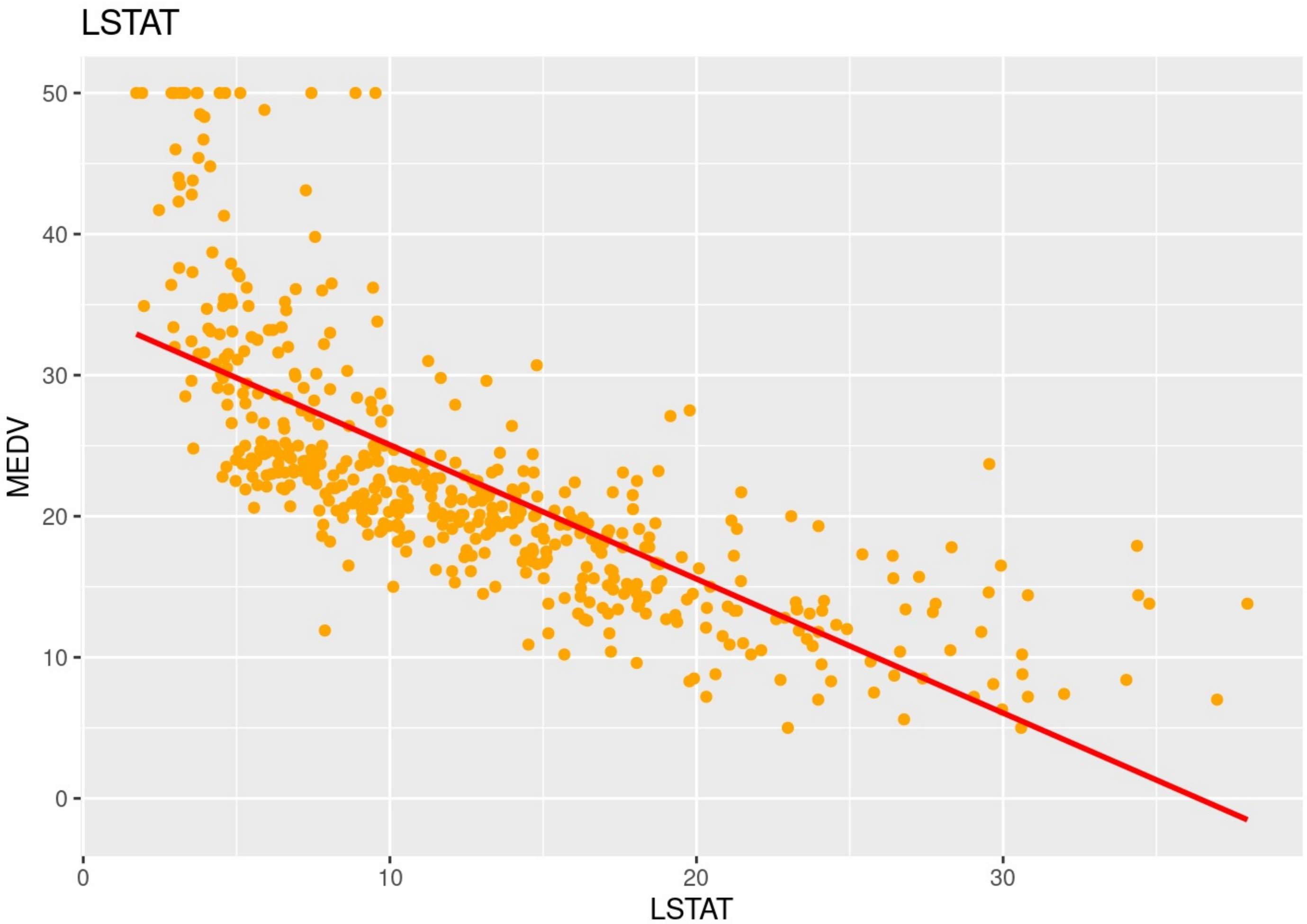



```
fit_lstat <-lm(medv~lstat)
summary(fit_lstat)
```

```
##
## Call:
## lm(formula = medv ~ lstat)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -15.168  -3.990  -1.318   2.034  24.500
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 34.55384    0.56263   61.41   <2e-16 ***
## lstat       -0.95005    0.03873  -24.53   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.216 on 504 degrees of freedom
## Multiple R-squared:  0.5441, Adjusted R-squared:  0.5432
## F-statistic: 601.6 on 1 and 504 DF,  p-value: < 2.2e-16
```

```
ggplot(data = boston_df, aes(x = lstat, y = medv)) +
  geom_point(color = "orange") +
  labs(title = "LSTAT",
       x = "LSTAT",
       y = "MEDV") +
  geom_smooth(method = lm, se = FALSE, color = "red")
```

```
## `geom_smooth()` using formula = 'y ~ x'
```



Задание 4

Сделайте выводы по полученным оценкам.

Полученные линейные зависимости описываются следующими уравнениями:

1. MEDV = -34.671 + 9.102 * RM
2. MEDV = 34.55384 - -0.95005 * LSTAT

Для обеих зависимостей мы видим что p-value лежит в области доверительного интервала (< 0,05), значит можем считать, что модель в целом является статистически значимой. Multiple R-squared: 0.4835 для переменной RM, Multiple R-squared: 0.5441 для переменной LSTAT, возможно такие значения обусловлены тем, что точки распределены в достаточно большом диапазоне по целевой переменной.

Цены растут по мере увеличения среднего количества комнат линейно.

Цены, как правило, снижаются с увеличением LSTAT (более низкий статус населения (в процентах)). Хотя, похоже, что он не следует точно линейной зависимости в начале.

Задание 5

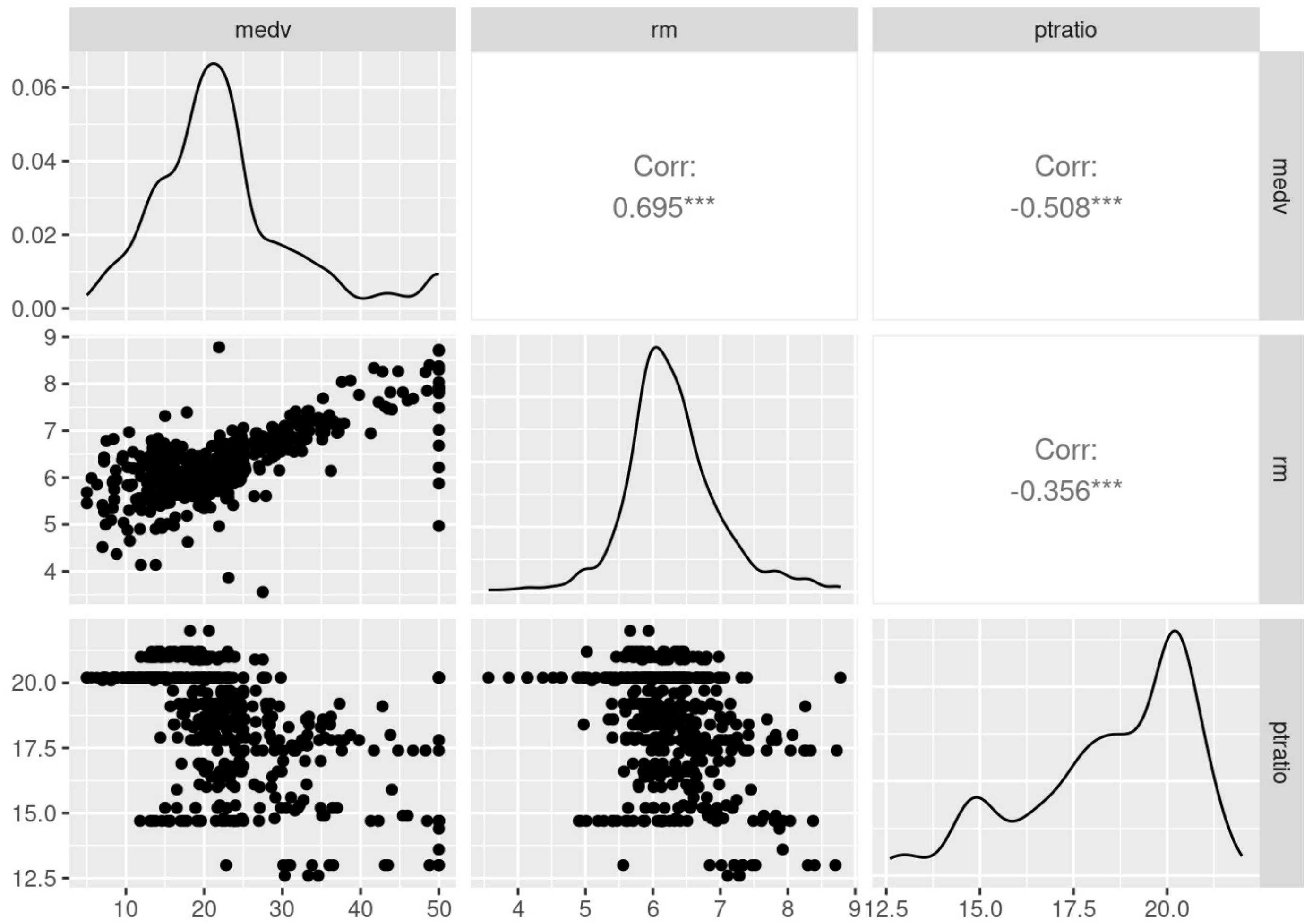
Выберите зависимую и несколько независимых переменных (минимум две), постройте множественную линейную регрессию.

Переменные RM и LSTAT имеют коэффициент корелляции -0,61 и могут быть зависимыми, поэтому для построения множественной линейной регрессии будем использовать переменные RM и PTRATIO(соотношение учеников и учителей по городам.) Коэффициент корреляции между этими двумя переменными -0,36

```r
library(GGally)
```

```
## Registered S3 method overwritten by 'GGally':
##   method from
##   +.gg   ggplot2
```

```
ggpairs(boston_df[, c("medv", "rm", "ptratio")])
```




```
fit_rm_ptratio <-lm(formula = medv ~ rm + ptratio)
summary(fit_rm_ptratio)
```

```
##
## Call:
## lm(formula = medv ~ rm + ptratio)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -17.672  -2.821   0.102   2.770  39.819
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -2.5612     4.1889  -0.611    0.541
## rm            7.7141     0.4136  18.650   <2e-16 ***
## ptratio      -1.2672     0.1342  -9.440   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.104 on 503 degrees of freedom
## Multiple R-squared:  0.5613, Adjusted R-squared:  0.5595
## F-statistic: 321.7 on 2 and 503 DF,  p-value: < 2.2e-16
```

Построим множественную регрессию для переменных RM и LSTAT, а также проверим значения R-квадрат, построив Ridge-регрессию(используется когда в данных может присутствовать мультиколлинеарность)

```
# Классическая множественная линейная регрессия
fit_rm_lstat <-lm(formula = medv ~ rm + lstat)
summary(fit_rm_lstat)
```

```
##
## Call:
## lm(formula = medv ~ rm + lstat)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -18.076  -3.516  -1.010   1.909  28.131
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.35827    3.17283  -0.428    0.669
## rm           5.09479    0.44447  11.463   <2e-16 ***
## lstat       -0.64236    0.04373 -14.689   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.54 on 503 degrees of freedom
## Multiple R-squared:  0.6386, Adjusted R-squared:  0.6371
## F-statistic: 444.3 on 2 and 503 DF,  p-value: < 2.2e-16
```

```r
#Построение Ridge-регрессии
library(glmnet)
```

```
## Загрузка требуемого пакета: Matrix
```

```
## Loaded glmnet 4.1-6
```

```r
y = medv
x = data.matrix(boston_df[, c("rm", "lstat")])
model <- glmnet(x, y, alpha = 0)

#Проведем кросс-валидацию и получим лучший параметр лямбда, дающий наименьшую среднеквадратичную ошибку
cv_model <- cv.glmnet(x, y, alpha = 0)
best_lambda <- cv_model$lambda.min
best_lambda
```

```
## [1] 0.6777654
```

```r
#Рассчитаем R-квадрат полученной модели
y_predicted <- predict (model, s = best_lambda, newx = x)
sst <- sum ((y - mean (y))^2)
sse <- sum ((y_predicted - y)^2)
rsq <- 1 - sse/sst
rsq
```

```
## [1] 0.6372866
```

Задание 6

Сделайте выводы по полученным оценкам.

Для зависимой переменной MEDV и независимых переменных RM и PTRATIO множественная линейная регрессия является статистически значимой, p-value < 0,05

Для зависимой переменной MEDV и независимых переменных RM и LSTAT множественная линейная регрессия также является статистически значимой, p-value < 0,05. R-квадрат полученный в ходе построения множественной линейной регрессии совпадает со значением полученным из ridge-регрессии, это говорит о том, что переменные RM и LSTAT независимы.