САНКТ-ПЕТЕРБУРГСКИЙ ПОЛИТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ им. ПЕТРА ВЕЛИКОГО

Институт прикладной математики и механики

Высшая школа прикладной математики и вычислительной физики

Отчёт

по лабораторной работе №5 «Регрессия»
по дисциплине «Системы искусственного интеллекта»

Студентка гр. 3630201/70101 _____ О. В. Саксина

Преподаватель _____ Л. В. Уткин

Санкт-Петербург,
2021г.

Содержание

1 Задание 1

1.1 Постановка задачи

Загрузите данные из файла reglab1.txt. Используя функцию lm, постройте регрессию (используйте разные модели). Выберите наиболее подходящую модель, объясните свой выбор.

1.2 Реализация

Коэффициенты детерминации (R-squared) для разных моделей, показывающие процент изменчивости, который обуславливается независимой переменной, представлены ниже.

Формула	R-squared
$z \sim x + y$	0.9686
$z \sim (x + y)^2$	0.9997
$z \sim x * y$	0.9997
$z \sim x/y$	0.9441

Таблица 1: Коэффициенты детерминации для моделей регрессии

Лучшими являются модели с формулами $z \sim (x + y)^2$ и $z \sim x * y$.

2 Задание 2

2.1 Постановка задачи

Реализуйте следующий алгоритм для уменьшения количества признаков, используемых для построения регрессии: для каждого $k \in \{0, 1, ..., d\}$ выбрать подмножество признаков мощности k[1], минимизирующее остаточную сумму квадратов RSS. Используя полученный алгоритм, выберите оптимальное подможество признаков для данных из файла reglab2.txt. Объясните свой выбор.

2.2 Реализация

На рис. 2 приведены результаты работы алгоритма. Оптимальным подмножеством является то, в которое входят все признаки, так как в этом случае остаточная сумма квадратов минимальна.

	formula	RSS
1	x1	157.2197758
2	x2	268.2457708
3	x3	393.4904686
4	x4	394.5904975
5	x1+x2	0.5379617
6	x1+x3	156.3540657
7	x1+x4	157.2192683
8	x2+x3	267.7954542
9	x2+x4	267.8061361
10	x3+x4	393.4587281
11	x1+x2+x3	0.3322662
12	x1+x2+x4	0.3619682
13	x1+x3+x4	156.3483397
14	x2+x3+x4	267.4415472
15	x1+x2+x3+x4	0.1928635

Рис. 1: RSS для различных моделей

3 Задание 3

3.1 Постановка задачи

Загрузите данные из файла cygage.txt. Постройте регрессию, выражающую зависимость возраста исследуемых отложений от глубины залегания, используя веса наблюдений. Оцените качество построенной модели.

3.2 Реализация

Построенная регрессия изображена на рис. 1. Коэффициент детерминации построенной модели равняется 0.9561.

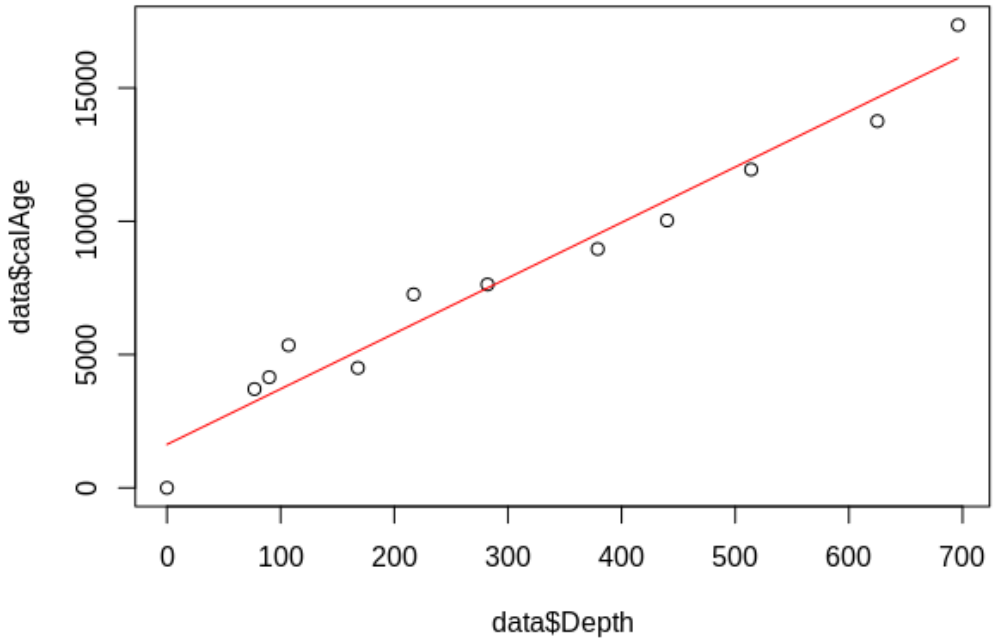

Рис. 2: Регрессия для данных cygage

4 Задание 4

4.1 Постановка задачи

Загрузите данные Longley (макроэкономические данные). Данные состоят из 7 экономических переменных, наблюдаемых с 1947 по 1962 годы (n=16):
GNP.deflator - дефлятор цен,
GNP - валовой национальный продукт,
Unemployed – число безработных
Armed.Forces – число людей в армии
Population – население, возраст которого старше 14 лет
Year - год
Employed – количество занятых
Построить регрессию lm(Employed $\sim$.). Исключите из набора данных longley переменную "Population". Разделите данные на тестовую и обучающую выборки равных размеров случайным образом. Постройте гребневую регрессию для значений $\lambda = 10^{-3+0.2i}, i = 0, ..., 25$, подсчитайте ошибку на тестовой и обучающей выборке для данных значений λ, постройте графики. Объясните полученные результаты.

4.2 Реализация

Коэффициент детерминации регрессии lm(Employed $\sim$.) равен 0.9955, статистически не значимыми оказались параметры GNP.deflator (p-value = 0.863141), GNP (p-value = 0.312681), Population (p-value = 0.826212).

На рис. 3 представлен график зависимости ошибки на тестовой и обучающей выборке от значения λ. Видно, что при увеличении λ ошибка увеличивается.

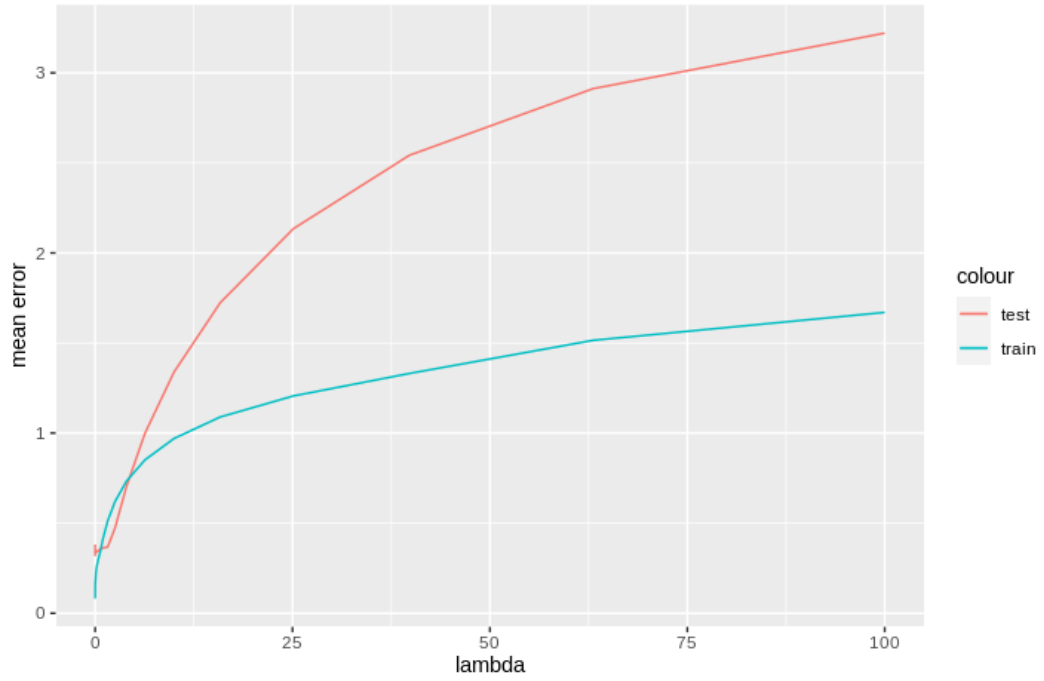

Рис. 3: Зависимость ошибки от значения λ на обучающей и тестовой выборках

5 Задание 5

5.1 Постановка задачи

Загрузите данные EuStockMarkets из пакета «datasets». Данные содержат ежедневные котировки на момент закрытия фондовых бирж: Germany DAX (Ibis), Switzerland SMI, France CAC, и UK FTSE. Постройте на одном графике все кривые изменения котировок во времени. Постройте линейную регрессию для каждой модели в отдельности и для всех моделей вместе. Оцените, какая из бирж имеет наибольшую динамику.

5.2 Реализация

На рис. 4 показаны изменения котировок для каждой биржи. Построив регрессию для каждой модели, можно увидеть (Таблица 2), что коэффициент year принимает наибольшее значение в модели SMI, значит она имеет наибольшую динамику.

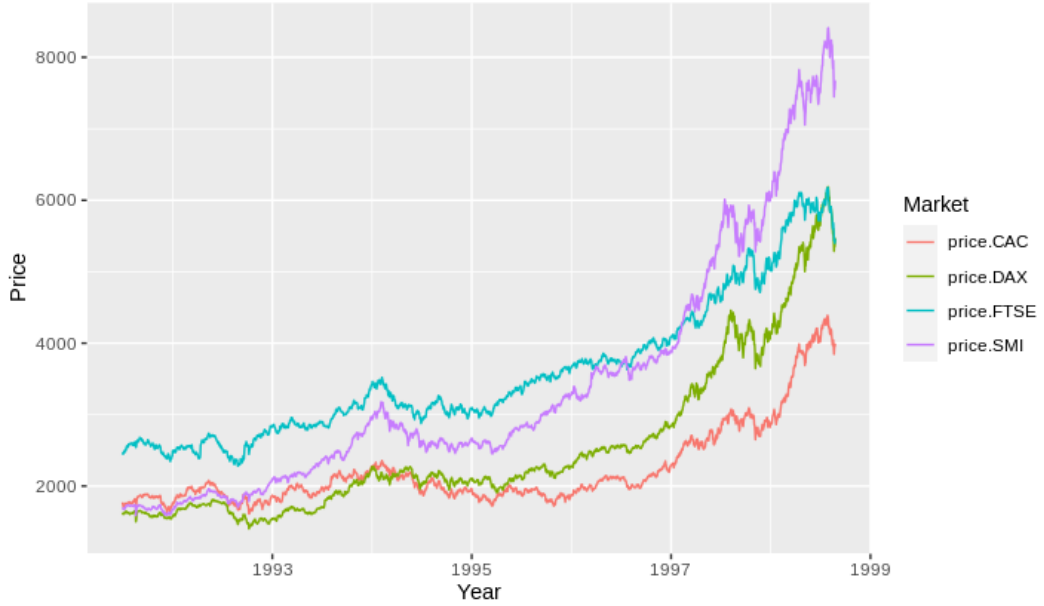

Рис. 4: Кривые изменения котировок во времени

Market	(Intercept)	year
CAC	-405915.2706	204.5757
DAX	-894557.8528	449.6524
FTSE	-865200.4152	435.4562
SMI	-1428160.1622	717.5365

Таблица 2: Коэффициенты построенных моделей регрессии

6 Задание 6

6.1 Постановка задачи

Загрузите данные JohnsonJohnson из пакета «datasets». Данные содержат поквартальную прибыль компании Johnson Johnson с 1960 по 1980 гг. Постройте на одном графике все кривые изменения прибыли во времени. Постройте линейную регрессию для каждого квартала в отдельности и для всех кварталов вместе. Оцените, в каком квартале компания имеет наибольшую и наименьшую динамику доходности. Сделайте прогноз по прибыли в 2016 году во всех кварталах и в среднем по году.

6.2 Реализация

На рис. 5 представлен график изменения прибыли за всё время с линией регрессии.

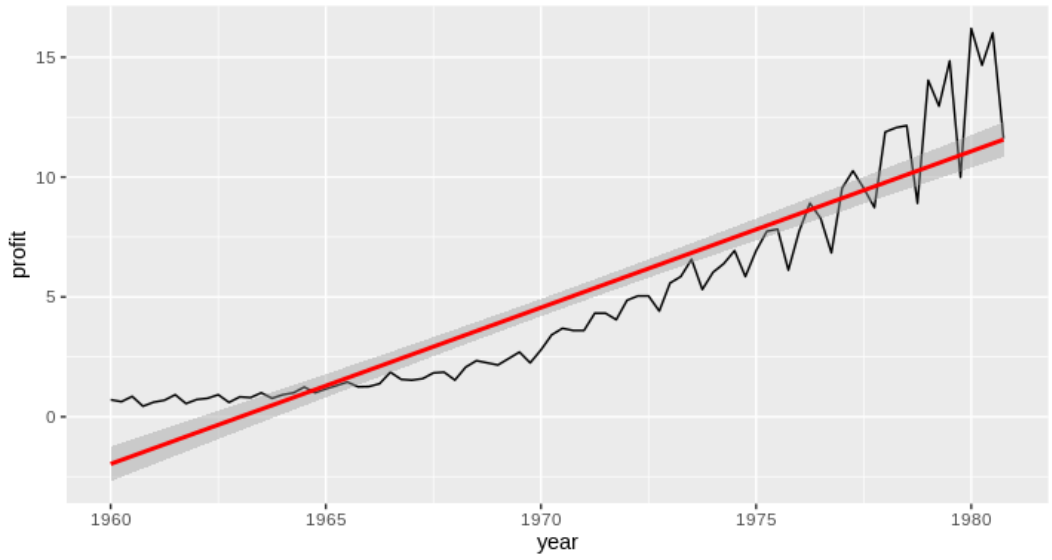

Рис. 5: График изменения прибыли

Наибольшую динамику доходности компания получила в четвёртом квартале 1979 года (коэффициент year = 24.8), наименьшую – в третьем квартеле 1979 года (коэффициент year = -19.4)
Предсказанные значения в 2016 году:

- 1 квартал: 34.55608
- 2 квартал: 34.71912
- 3 квартал: 34.88217
- 4 квартал: 35.04522
- в среднем по году: 34.80065

7 Задание 7

7.1 Постановка задачи

Загрузите данные sunspot.year из пакета «datasets». Данные содержат количество солнечных пятен с 1700 по 1988 гг. Постройте на графике кривую изменения числа солнечных пятен во времени. Постройте линейную регрессию для данных.

7.2 Реализация

Построенный график представлен на рис. 6.

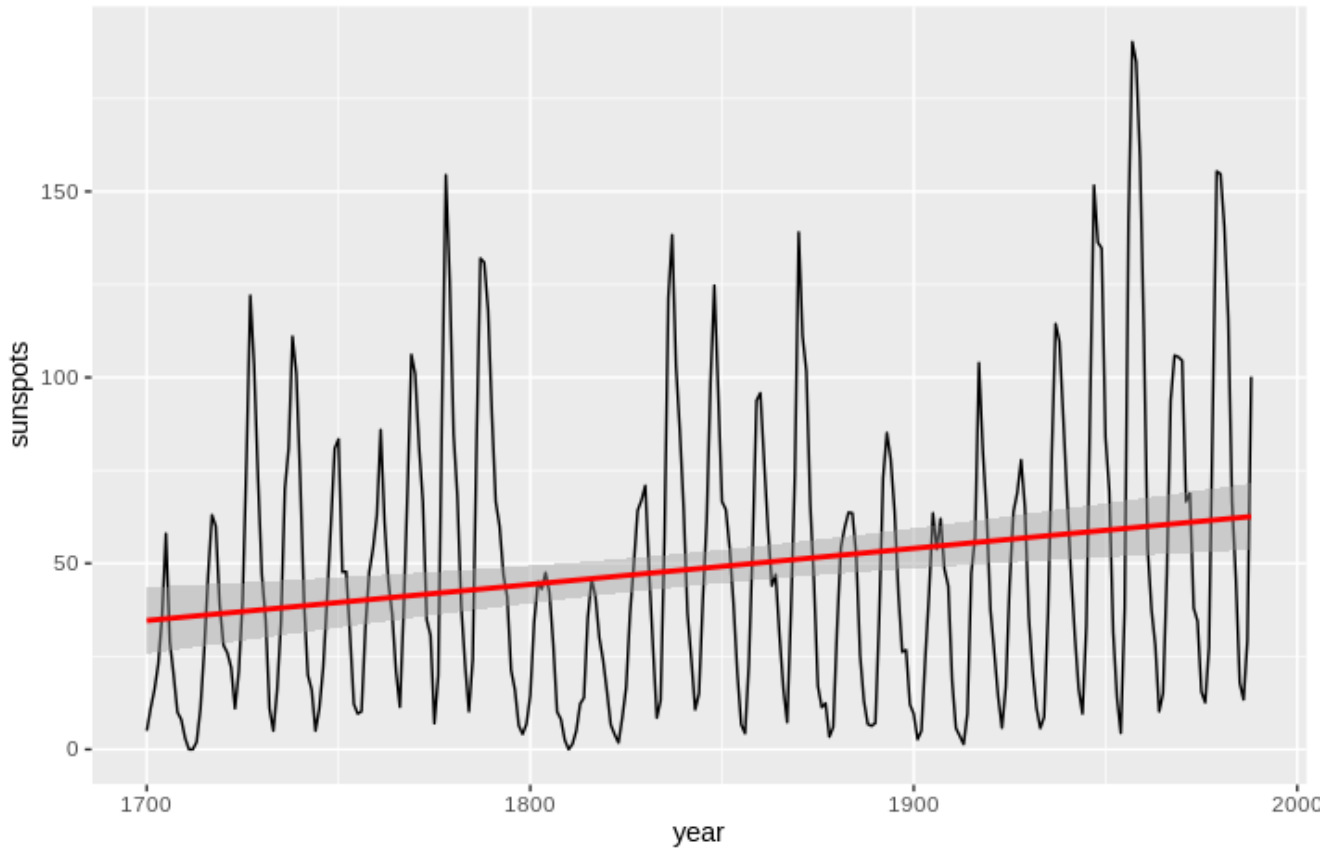

Рис. 6: Кривая изменения числа солнечных пятен во времени с линией регрессии

8 Задание 8

8.1 Постановка задачи

Загрузите данные из файла пакета «UKgas.scv». Данные содержат объемы ежеквартально потребляемого газа в Великобритании с 1960 по 1986 гг. Постройте линейную регрессию для каждого квартала в отдельности и для всех кварталов вместе. Оцените, в каком квартале потребление газа имеет наибольшую и наименьшую динамику доходности. Сделайте прогноз по потреблению газа в 2016 году во всех кварталах и в среднем по году.

8.2 Реализация

Наибольшая динамика получилась в третьем квартале 1985 года (коэффициент time = 2023), наименьшая – в первом квартале 1985 года (коэффициент time = -2209)
Предсказанные значения в 2016 году:

- 1 квартал: 1351.585
- 2 квартал: 1357.532
- 3 квартал: 1363.479
- 4 квартал: 1369.426
- в среднем по году: 1360.506

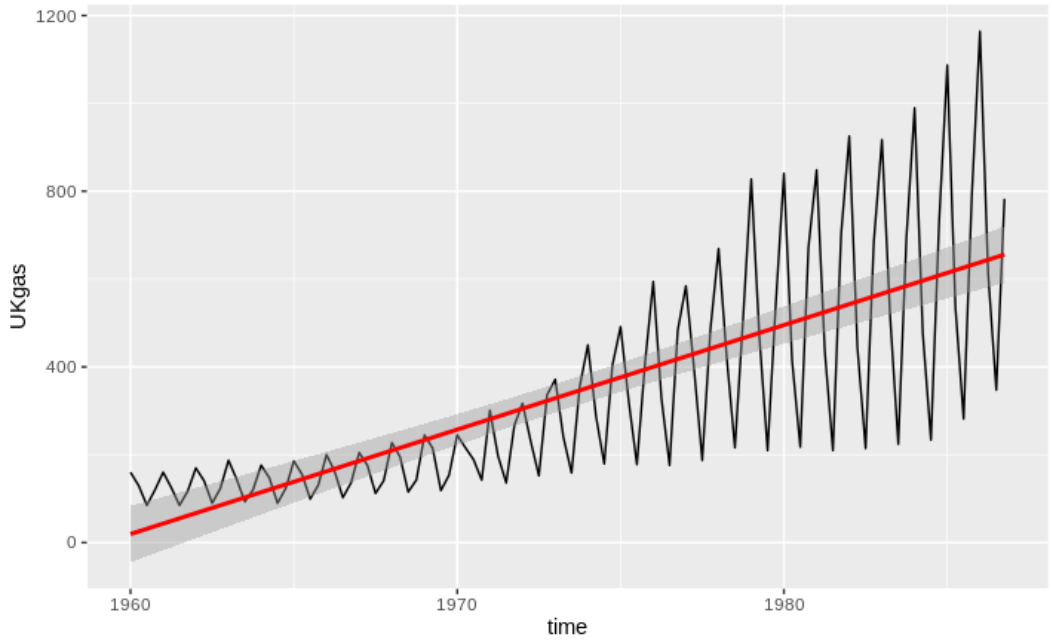

Рис. 7: График изменения потребления газа

9 Задание 9

9.1 Постановка задачи

Загрузите данные cars из пакета «datasets». Данные содержат зависимости тормозного пути автомобиля (футы) от его скорости (мили в час). Данные получены в 1920 г. Постройте регрессионную модель и оцените длину тормозного пути при скорости 40 миль в час.

9.2 Реализация

Предсказанная длина тормозного пути при скорости 40 миль в час – 139.7173

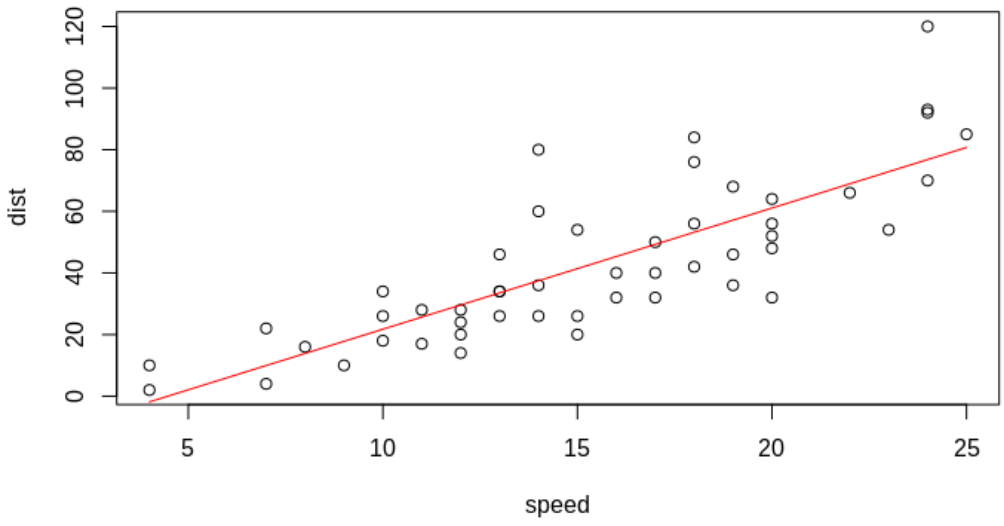

Рис. 8: Зависимость тормозного пути автомобиля от его скорости

Приложение

Задание 1

```
library(scatterplot3d)

setwd("/home/olga/MyProjects/Polikek/ML/Regression/datasets")

data = read.delim("reglab1.txt")
plot(data)
plot3d(x=data$x, y=data$y, z=data$z)
scatterplot3d(x = data$x, y = data$y, z = data$z)
fit = lm(z ~ x+y, data, subset = !is.na(x) & !is.na(y))
summary(fit)
fit = lm(z ~ x/y, data, subset = !is.na(y) & !is.na(y))
summary(fit)
fit = lm(z ~ (x+y)^2, data, subset = !is.na(x) & !is.na(y))
summary(fit)
fit = lm(z ~ x * y, data, subset = !is.na(x) & !is.na(y))
summary(fit)

plot(fit$residuals, ylab = "residuals", log = "")
abline(0, 0, col = "red")
```

Задание 2

```
library(qpcR)

setwd("/home/olga/MyProjects/Polikek/ML/Regression/datasets")

data = read.delim("reglab2.txt")
x = colnames(data)[-1]
min_RSS = 10000
rss = c()
formulas = c()
k = 1
set = c()
for (i in 1:(dim(data)[2]-1)){
  c = combn(x, i)
  d = dim(c)[2]
  for (j in 1:d){
    formula = as.formula(paste("y ~", paste(c[,d], collapse = "+")))
    fit = lm(formula, data)
    rss[k] = RSS(fit)
    k = k + 1
    if (RSS(fit) < min_RSS){
      min_RSS = RSS(fit)
      set = c[,d]
    }
  }
}
```

Задание 3

```
setwd("/home/olga/MyProjects/Polikek/ML/Regression/datasets")

data = read.delim("cygage.txt")
f = lm(calAge ~ Depth, data, weights = data$Depth)
plot(data$Depth, data$calAge)
lines(data$Depth, predict(f), col = 'red')
summary(f)
```

Задание 4

```r
library(MASS)
library(glmnet)
library(lmridge)
library(ggplot2)
etwd("/home/olga/MyProjects/Polikek/ML/Regression/datasets")

plot(longley)
reg = lm(Employed ~ ., longley)
summary(reg)
data = subset(longley, select=-c(Population))
s = sample(seq(dim(data)[1]), dim(data)[1] * 0.5)
train = data[s,]
test = data[-s,]
test_er = c()
train_er = c()
lambda_seq = 10^(-3+0.2*(0:10))
j = 1
for (i in lambda_seq){
  reg = lm.ridge(Employed ~ ., train, lambda=i)
  pred.train = scale(train[1:5],center = TRUE, scale = reg$scales)%*%
    reg$coef + reg$ym
  pred.test = scale(test[1:5],center = TRUE, scale = reg$scales)%*%
    reg$coef + reg$ym
  test_er[j] = mean(sqrt((test$Employed - pred.test)^2))
  train_er[j] = mean(sqrt((train$Employed - pred.train)^2))
  j = j + 1
}

df = data.frame(lambda = lambda_seq, test = test_er, train = train_er)
ggplot(df, aes(x = lambda)) +
  geom_line(aes(y = test, color = "test")) +
  geom_line(aes(y = train, color = "train")) +
  labs(y = "mean error")

plot(data)
```

Задание 5

```r
library(datasets)
library(tidyr)

data = data.frame(year=as.numeric(time(EuStockMarkets)),
                  price=as.matrix(EuStockMarkets))
df = gather(data, key=measure, value=Rate, c("price.DAX", "price.SMI", "price.CAC", "price.FTSE"))

ggplot(df, aes(x=year, y = Rate, group = measure, colour = measure)) +
  geom_line() +
  labs(x = "Year", y = "Price", color = "Market")

all = data.frame(Year=rep(data$year, 4),
                 Price=c(data$price.CAC, data$price.DAX, data$price.FTSE, data$price.SMI))

ggplot(data, aes(x = year)) +
  geom_line(aes(y = price.DAX)) +
  geom_line(aes(y = price.SMI)) +
  stat_smooth(method = "lm", col = "red")

ggplot(all, aes(x=Year, y=Price)) +
  geom_point() +
  stat_smooth(method = "lm", col = "red")

reg.CAC = lm(price.CAC ~ year, data)
reg.DAX = lm(price.DAX ~ year, data)
reg.FTSE = lm(price.FTSE ~ year, data)
```

```
reg.SMI = lm(price.SMI ~ year, data)
reg = lm(price ~ year, all)

summary(reg.CAC)
summary(reg.DAX)
summary(reg.FTSE)
summary(reg.SMI)
summary(reg)
reg.CAC$coefficients
reg.DAX$coefficients
reg.FTSE$coefficients
reg.SMI$coefficients
```

Задание 6

```
library(datasets)

data = data.frame(year=as.numeric(time(JohnsonJohnson)), profit=as.matrix(JohnsonJohnson))
ggplot(data, aes(x=year, y=profit)) + geom_line() +
  stat_smooth(method = "lm", col = "red")
reg = lm(profit ~ year, data)

mean(c(predict.lm(reg, list(year=2016.00)),
predict.lm(reg, list(year=2016.25)),
predict.lm(reg, list(year=2016.50)),
predict.lm(reg, list(year=2016.75))))

#plot(data$year, data$profit)
#lines(data$year, predict(reg), col = 'red')

regs = c()
date_min = 0
date_max = 0
coef_max = 0
coef_min = 100
for (i in 1:(dim(data)[1]-1)){
  df = data.frame(year = c(data$year[i], data$year[i+1]),
                  profit = c(data$profit[i], data$profit[i+1]))
  reg = lm(profit ~ year, df)
  if (reg$coefficients["year"] < coef_min){
    coef_min = reg$coefficients["year"]
    date_min = data$year[i]
  }
  else{
    if (reg$coefficients["year"] > coef_max){
      coef_max = reg$coefficients["year"]
      date_max = data$year[i]
    }
  }
}
```

Задание 7

```
library(datasets)
library(forecast)
library(TSstudio)

data = sunspot.year
df = data.frame(year=time(data), sunspots=as.matrix(data))
reg = lm(sunspots ~., df)
ggplot(df, aes(x=year, y=sunspots)) +
  geom_line() +
  stat_smooth(method = "lm", col = "red")
summary(reg)
```

Задание 8

```r
library(dplyr)
library(ggpubr)
setwd("/home/olga/MyProjects/Polikek/ML/Regression/datasets")

data = read.csv("UKgas.csv")
data = data[ ,-1]
reg = lm(UKgas ~ time, data)
ggplot(data, aes(x=time, y=UKgas)) + geom_line() +
  stat_smooth(method = "lm", col = "red")

predict.lm(reg, list(time=2016.00))
predict.lm(reg, list(time=2016.25))
predict.lm(reg, list(time=2016.50))
predict.lm(reg, list(time=2016.75))

date_min = 0
date_max = 0
coef_max = 0
coef_min = 100
for (i in 1:(dim(data)[1]-1)){
  df = data.frame(time = c(data$time[i], data$time[i+1]),
                  UKgas = c(data$UKgas[i], data$UKgas[i+1]))
  reg = lm(UKgas ~ time, df)
  if (reg$coefficients["time"] < coef_min){
    coef_min = reg$coefficients["time"]
    date_min = data$time[i]
  }
  else{
    if (reg$coefficients["time"] > coef_max){
      coef_max = reg$coefficients["time"]
      date_max = data$time[i]
    }
  }
}
```

Задание 9

```r
data = cars
plot(data)
reg = lm(dist ~ speed, data)
summary(reg)
lines(data$speed, predict(reg), col = 'red')

predict(reg, list(speed=40))
```