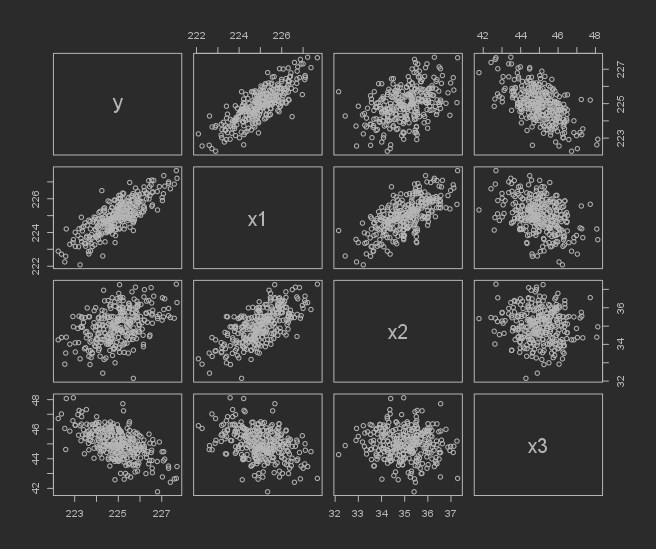
**Отчет**

Данные

Матрица корреляций

Изображение выглядит как текст

Автоматически созданное описание

Еще один способ отображения матрицы корреляций и p-value

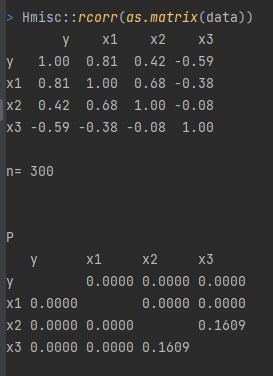
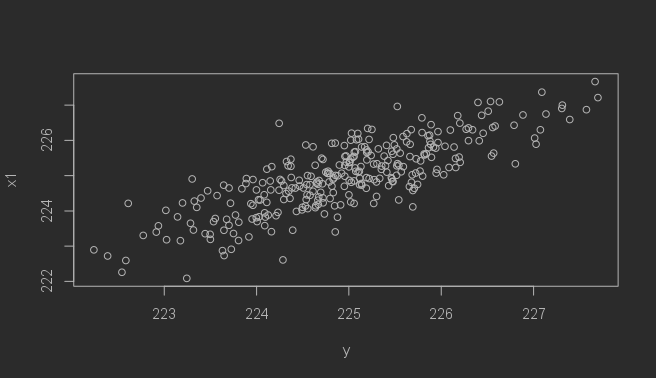


Table 1 другим способом

Как мы видим, взаимосвязь с зависимой переменной **y**

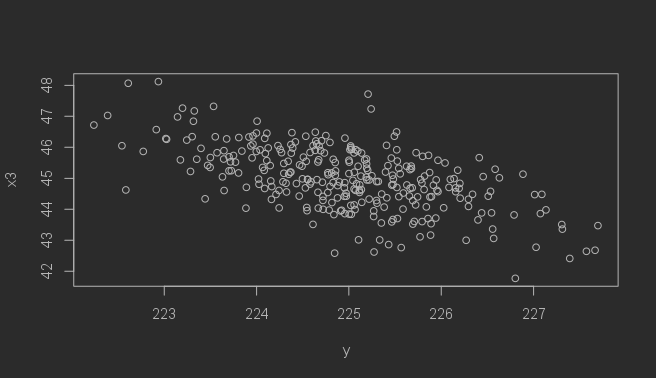
* с фактором **x1** довольно высока, что значит при увеличении **x1**, увеличится и **y**

1 положительная корреляция

* с фактором **x2** очень слабаяИзображение выглядит как текст, монитор, экран

  Автоматически созданное описание

2 Отсутствие корреляции

* а с фактором **x3** средняя отрицательная (при увеличении **x3**, наша **y** будет уменьшаться) 

3 отрицательная

Корреляционный тест пирсонаИзображение выглядит как текст

Автоматически созданное описание

Коэффициент = 0.3326248, что означает очень слабую взаимосвязь между переменными

Следует отобрать факторы x1 и x3

**Тест на длинную и короткую регрессию**

Изображение выглядит как текст

Автоматически созданное описание

Гипотеза H0 отвергается

**ПАРНАЯ РЕГРЕССИЯ**

1. **y ~ x2**

yi = α +βix2i + εi

ŷi = 208.64946 + 0.46576x2i

(2.02749) (0.05787)

1. **y ~ x3**

yi = α +βix3i + εi

ŷi = 251.18225 + -0.58257x3i

(2.09667) (0.04657)

1. **y ~ x1**

yi = α +βix1i + εi

ŷi = 40.14760 + 0.82141x1i

(7.80618) (0.03469)

Sош = 0.5967029; R2 = 0.6528977; F = 560.5; A = Eотн = 0.2106146

**Оценка качества модели:**

R2 = 0.6528977 – модель достаточно удовлетворительного качества (чем ближе к 1 – тем лучше)

Sош = 0.5967029 – очень мала => оценка неизвестного параметра крайне точна. С помощью стандартных ошибок мы можем узнать, насколько близко оценки и находятся от β0 и β1.

A = Eотн = 0.2106146 – очень мала => точность нашей модели высокая

**Оценка модели на значимость в целом:**

Fнабл = 560.5

p-*value* < 2,2e-16

Так как p-value<*alpha*(0,1;0,01;0,05) => отвергаем H0 гипотезу, модель регрессии в целом значима.

**Оценка параметров на значимость:**

tα = 5.143

p-*value* = 4.91e-07

H0 отвергается при *alpha(1%,5%,10%)* - параметр значим

tβ = 23.676

p-*value* < 2e-16

H0 отвергается при *alpha(1%,5%,10%)* - параметр значим

**Доверительные интервалы:**

24.78538 ≤ β0 ≤ 55.50982

0.753132 ≤ β1 ≤ 0.8896855

**Бета-коэффициенты:**

β̃1 = 0.8080208

* Зависимая переменная изменится на 0.8 своего среднеквадратичного отклонения при изменении первого фактора на одно среднеквадратичное отклонение

**Дельта-коэффициенты:**

= 1.016569

**Коэффициенты эластичности:**

= 0.8215362

Зависимая переменная изменится

* на 0,82% при изменении первого фактора на 1%
* Зависимая переменная неэластична, т.к. эластичность меньше 1.

**Проверка на гомоскедастичность и на отсутствие автокорреляции**

**Тест Дарбина-Уотсона:**

DW = 1.8316

p-value = 0.07173

n = 300

k = 1

p1 = 0.0842

dL = 1.65

dU = 1.69

DW > dU

Принимаем гипотезу H1.

Автокорреляция отсутствует

**Тест Бреуша-Годфри:**

BG = 1.5636

p-value = 0.2111

xi = 5.991465

BG < xi – автокорреляции нет

**Тест Голдфельда-Квандта:**

GQ = 1.0008

p-value = 0.4985

Fтабл = 1.18

GQ < Fтабл : гетероскедастичности нет

**Тест Бреуша-Пагана:**

BP = 0.22267

p-value = 0.637

xi = 5.991465

BP < xi: гетероскедастичности нет

Автокорреляция и гомоскедастичности нет.

**МНОЖЕСТВЕННАЯ РЕГРЕССИЯ**

yi = α +βix1i + βix3i + εi

ŷi = 83.20329 + 0.69537x1i + -0.32654x3i

(7.86782) (0.03212) (0.03137) Изображение выглядит как текст, монитор

Автоматически созданное описание

Sош = 0.5116122; R2 = 0.7456901; F = 435.4; A = Eотн = 0.1812383

R2 = 0.7456901 – модель достаточно хорошего качества (чем ближе к 1 – тем лучше)

Sош = 0.5116122 – очень мала => оценка неизвестного параметра крайне точна. С помощью стандартных ошибок мы можем узнать, насколько близко оценки и находятся от β0 и β2.

A = Eотн = 0.1812383– очень мала => точность нашей модели высокая

* **t-критерий Стьюдента для проверки значимости коэффициента корреляции**

Изображение выглядит как текст

Автоматически созданное описание

tнабл x1 = 23.67565

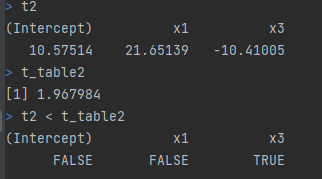
tнабл x3 = 12.50874

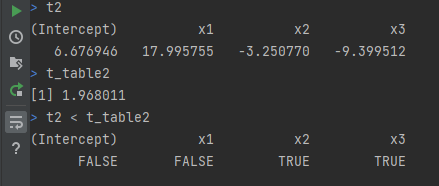
tтабл = 4.302653

Отвергаем H0 гипотезу для всех наблюдений

Т. к. расчетные статистики больше табличной, то корреляции между каждым фактором и зависимой переменной значимо отличаются от 0

* **t-критерий Стьюдента для оценки значимости параметров модели линейной регрессии**





tβ0 = 10.57514

tβ1 = 21.65139

tβ3 = -10.41005

tтабл(α=0,05) = 1.967984

tβ0 и tβ1 > tтабл следовательно эти коэффициенты значимы, отвергаем H0 гипотезу

tβ3 < tтабл следовательно этот коэффициент не значим, принимаем H0 гипотезу

* **Тест Фишера**

Изображение выглядит как текст

Автоматически созданное описание

p-value: < 2.2e-16

alpha(0.05) = 2.64

p-value < alpha: Отвергаем H0 гипотезу, модель в целом значима

Доверительные интервалы Изображение выглядит как текст

Автоматически созданное описание

67.71955 ≤ β0 (α)≤ 98.68704

0.6321619 ≤ β1 ≤ 0.7585713

-0.3882668 ≤ β3 ≤ -0.2648057

Эластичность, бета и дельта

Изображение выглядит как текст

Автоматически созданное описание

**Эластичность**

**X1 | 0.6954746**

**X3 | -0.06532913**

Зависимая переменная изменится

* на 0,69% при изменении первого фактора на 1%,
* на -0,06% - третьего

Зависимая переменная неэластична по всем факторам т.к. каждая эластичность меньше 1.

Наибольшая эластичность по 1 фактору

**Бета**

**X1 | 0.6840331**

**X3 | -0.3288851**

* Зависимая переменная изменится на 0.68 своего среднеквадратичного отклонения при изменении первого фактора на одно среднеквадратичное отклонение
* на -0.32 СКО при изменении третьего фактора

**дельта**

**x1 | 0.7534909**

**x3 | 0.2569418**

Наибольшее влияние оказывает первый фактор

**Проверка на гомоскедастичность и на отсутствие автокорреляции**

**Тест Дарбина-Уотсона:**

DW = 1.794

p-value = 0.03666

n = 300

k = 1

p1 = 0.0842

dL = 1.65

dU = 1.69

DW > dU

Принимаем гипотезу H1.

Автокорреляция отсутствует

**Тест Бреуша-Годфри:**

BG = 2.7239

p-value = 0.09886

ꭓ2 для 2 порядка = 5.991465

BG < xi – автокорреляции нет

**Тест Голдфельда-Квандта:**

GQ = 1.3176

p-value = 0.1222

C = 300/4 = 75

N1, N2 = n-c / 2 = 112.5

Fтабл(110, 110) = 1.26

GQ < Fтабл : гетероскедастичности нет

**Тест Бреуша-Пагана:**

BP = 0.7938

p-value = 0.6724

ꭓ2 для 2 порядка = 5.991465

BP < xi: гетероскедастичности нет

Автокорреляции и гомоскедастичности нет.

**Сравнение моделей**

Сравниваю при помощи теста ANOVA.

Изображение выглядит как текст

Автоматически созданное описание

Согласно этой модели, множественная регрессия лучше, содержит в себе 100% долю от общего объема прогностической мощности

*library(*lmtest*)  
library(*binom*)  
require(*readxl*)*data <- *read.table(***'./9oct/hw/Kurtaev.txt'***)*y <- data$y  
x1 <- data$x1  
x2 <- data$x2  
x3 <- data$x3  
  
# 1  
correl <- *cor(*data*)*correl  
  
rx <- correl*[*1, *]*rx1 <- *cor(*x1, y*)*rx2 <- *cor(*x2, y*)*rx3 <- *cor(*x3, y*)  
  
cor.test(*y, x1+x2+x3, method = **"pearson"***)* # 0.3326248 => очень слабая взаимосвязь между переменными  
  
*cor.test(*y, x1+x2+x3, method = **"spearman"***)*Hmisc::*rcorr(as.matrix(*data*))  
  
plot(*y, x1*)* # Слабая положительная корреляция  
*plot(*y, x2*)* # Отсутствие корреляции  
*plot(*y, x3*)* # Слабая отрицательная корреляция  
  
  
  
#2  
lm1 <- *lm(*y ~ x1*)*slm1 <- *summary(*lm1*)*slm1  
  
  
m <- *lm(*y ~ x1 + x3*)*sm <- *summary(*m*)*sm  
  
# Тест на длинную и короткую регрессию  
r\_long <- *summary(lm(*y ~ x1+x2+x3*))*$r.squared  
r\_short <- sm$r.squared  
F\_see <- *((*r\_long - r\_short*)* / *(*1-r\_long*))* \* *((length(*y*)* - 4*)* / 1*)*F\_see < 3.87  
  
  
#3  
slm1\_r\_squared <- slm1$r.squared  
slm1\_r\_squared  
slm1\_std\_error <- slm1$sigma  
slm1\_std\_error  
slm1\_approx\_error <- *sum(abs(*slm1$residuals/y*))* / *length(*y*)* \* 100  
slm1\_approx\_error  
  
  
smlm\_r\_squared <- sm$r.squared  
smlm\_r\_squared  
smlm\_std\_error <- sm$sigma  
smlm\_std\_error  
smlm\_approx\_error <- *sum(abs(*sm$residuals/y*))* / *length(*y*)* \* 100  
smlm\_approx\_error  
  
  
#4  
# t-критерий Стьюдента для проверки значимости коэффициента корреляции  
t1\_p <- *sqrt((*rx1 ^ 2*)* / *(*1 - rx1 ^ 2*)* \* *(length(*y*)* - slm1$fstatistic*[[***"numdf"***]]))*t\_table1\_p <- *qt(*0.975, df = slm1$fstatistic*[[***"numdf"***]])*t1\_p < t\_table1\_p # корреляции между каждым фактором и зависимой переменной значимо отличаются от 0  
  
t1 <- *c(  
 sqrt((*rx1 ^ 2*)* / *(*1 - rx1 ^ 2*)* \* *(length(*y*)* - sm$fstatistic*[[***"numdf"***]]))*,  
 *sqrt((*rx3 ^ 2*)* / *(*1 - rx3 ^ 2*)* \* *(length(*y*)* - sm$fstatistic*[[***"numdf"***]]))  
)*t\_table1 <- *qt(*0.975, df = sm$fstatistic*[[***"numdf"***]])*t1 < t\_table1 # корреляции между каждым фактором и зависимой переменной значимо отличаются от 0  
  
  
# t-критерий Стьюдента для оценки значимости параметров модели линейной регрессии  
t2\_p <- slm1$coefficients*[*, 1*]* / slm1$coefficients*[*, 2*]*t\_table2\_p <- *qt(*0.975, df = slm1$fstatistic*[[***"dendf"***]])*t2\_p < t\_table2\_p # Все коэфы незначимы  
  
t2 <- sm$coefficients*[*, 1*]* / sm$coefficients*[*, 2*]*t\_table2 <- *qt(*0.975, df = sm$fstatistic*[[***"dendf"***]])*t2 < t\_table2 # Все коэфы незначимы  
  
  
# Тест Фишера  
F\_statistic\_p <- slm1$fstatistic*[***"value"***][[*1*]]*F\_table\_p <- 2.64  
F\_statistic\_p < F\_table\_p # Модель в целом значима  
  
F\_statistic <- sm$fstatistic*[***"value"***][[*1*]]*F\_table <- 2.64  
F\_statistic < F\_table # Модель в целом значима  
  
  
#5  
B\_p <- slm1$coefficients*[*, 1*]*di\_lower\_p <- B\_p - slm1$coefficients*[*, 2*]* \* t\_table2\_p  
di\_upper\_p <- B\_p + slm1$coefficients*[*, 2*]* \* t\_table2\_p  
  
  
B <- sm$coefficients*[*, 1*]*di\_lower <- B - sm$coefficients*[*, 2*]* \* t\_table2  
di\_upper <- B + sm$coefficients*[*, 2*]* \* t\_table2  
  
  
#6  
elastichnost\_p <- *c(* slm1$coefficients*[*2*]* \* *mean(*x1*)* / *mean(*y*)  
)*elastichnost <- *c(* sm$coefficients*[*2*]* \* *mean(*x1*)* / *mean(*y*)*,  
 sm$coefficients*[*3*]* \* *mean(*x3*)* / *mean(*y*)  
)* # зависимая переменная изменится на 58% при изменении первого фактора на 1%, на 27% - второго.  
# Зависимая переменная неэластична по обеим факторам т.к. каждая эластичность меньше 1.  
# Наибольшая эластичность по 1 фактору  
  
# Бета-коэф.  
Sdy <- *sd(*y*)*Sdx1 <- *sd(*x1*)*Sdx2 <- *sd(*x2*)*Sdx3 <- *sd(*x3*)*beta\_p <- slm1$coefficients*[*2*]* \* Sdx1/Sdy  
  
beta <- *c(*sm$coefficients*[*2*]* \* Sdx1/Sdy, sm$coefficients*[*3*]* \* Sdx3/Sdy*)* #  
  
# Дельта-коэф.  
delta\_p <- rx1 \* slm1$coefficients*[*2*]* / slm1$r.squared  
  
delta <- *c(*rx1 \* sm$coefficients*[*2*]* / sm$r.squared,  
 rx3 \* sm$coefficients*[*3*]* / sm$r.squared*)*#7  
xi <- *qchisq(*p = 0.95, df = 2*)*dw\_p <- *dwtest(*lm1*)*bg\_p <- *bgtest(*lm1, order = 1, order.by = NULL, type = *c(***"Chisq"**, **"F"***))*bg\_p$statistic < xi  
  
gq\_p <- *gqtest(*lm1, order.by = x1, fraction = 0.5*)*n <- *length(*y*)*c <- 1  
F\_table\_gq\_p <- 1.18  
gq\_p$statistic < F\_table\_gq\_p  
  
bp\_p <- *bptest(*lm1, studentize = TRUE*)*bp\_p$statistic < xi  
  
  
  
  
dw <- *dwtest(*m*)*bg <- *bgtest(*m, order = 1, order.by = NULL, type = *c(***"Chisq"**, **"F"***))  
qchisq(*p = 0.95, df = 2*)*gq <- *gqtest(*m, order.by = x1, fraction = 0.5*)*bp <- *bptest(*m, studentize = TRUE*)*#8  
  
aov\_model <- *aov(*y ~ x1+x3*)*aov\_model\_p <- *aov(*y ~ x1*)*saov <- *summary(*aov\_model*)*saovp <- *summary(*aov\_model\_p*)  
  
library(*AICcmodavg*)*model.set <- *list(*aov\_model, aov\_model\_p*)*model.names <- *c(***"Множественная"**, **"Парная"***)  
  
aictab(*model.set, modnames = model.names*)*

#%%  
  
import numpy as np  
import pandas as pd  
import statsmodels.api as smmy  
import statsmodels.stats.api as sms  
# import statsmodels.formula.api as sm  
from statsmodels.compat import lzip  
from sklearn.linear\_model import LinearRegression  
import matplotlib.pyplot as plt  
from sklearn import svm  
import sklearn  
import statsmodels.api as sm  
import scipy.stats as t  
import statsmodels.stats as smd  
  
#%%  
  
df = pd.read\_table*(***'Kurtaev.txt'***)*df  
  
#%%  
  
corr\_matrix = df.corr*()*corr\_matrix  
  
#%%  
  
y = df*[***'y'***]*y\_raw = np.array*(*y*)*y = np.array*(*y*)*.reshape*((*-1, 1*))*x1 = np.array*(*df*[***'x1'***])*x2 = np.array*(*df*[***'x2'***])*x3 = np.array*(*df*[***'x3'***])*  
  
  
#%%  
  
fig, ax = plt.subplots*(*figsize=*(*10, 6*))*ax.scatter*(*x = df*[***'x1'***]*, y = df*[***'y'***])*plt.xlabel*(***"x1"***)*plt.ylabel*(***"y"***)*plt.show*()*#%%  
  
  
fig, ax = plt.subplots*(*figsize=*(*10, 6*))*ax.scatter*(*x = df*[***'x2'***]*, y = df*[***'y'***])*plt.xlabel*(***"x2"***)*plt.ylabel*(***"y"***)*plt.show*()*#%%  
  
  
fig, ax = plt.subplots*(*figsize=*(*10, 6*))*ax.scatter*(*x = df*[***'x3'***]*, y = df*[***'y'***])*plt.xlabel*(***"x3"***)*plt.ylabel*(***"y"***)*plt.show*()*#%%  
  
from pprint import pp  
# x1\_x3 = ([[i, j] for i in x1 for j in x3])  
x1\_x3 = np.column\_stack*([*x1, x3*])*x1\_x2\_x3 = np.column\_stack*([*x1, x2, x3*])*x1\_x2\_x3*[*:20*]*#%%  
  
plural\_model = LinearRegression*()*plural\_model.fit*(*x1\_x2\_x3, y\_raw*)*r\_squared\_plural = plural\_model.score*(*x1\_x2\_x3, y\_raw*)*r\_squared\_plural  
  
#%%  
  
twin\_model = LinearRegression*()*.fit*(*y, x1*)*r\_squared\_twin = twin\_model.score*(*y, x1*)*r\_squared\_twin  
  
  
#%%  
  
print*(***f"PLURAL MODEL:** \u0177 **=** *{*plural\_model.intercept\_*}* **+** *{*plural\_model.coef\_*[*0*]}***x**\u2081\u1D62 **+** *{*plural\_model.coef\_*[*2*]}***x**\u2082\u1D62 **+** *{*plural\_model.coef\_*[*1*]}***x**\u2083\u1D62**"***)*print*(***f"TWIN MODEL:** \u0177 **=** *{*twin\_model.intercept\_*}* **+** *{*twin\_model.coef\_*[*0*]}***x**\u2081\u1D62**"***)*#%%  
  
x1\_x2\_x3\_const = sm.add\_constant*(*x1\_x2\_x3*)*x1\_const = sm.add\_constant*(*x1*)*est\_twin = sm.OLS*(*y, x1\_const*)*est\_plural = sm.OLS*(*y, x1\_x2\_x3\_const*)*#%%  
  
print*(*est\_twin.fit*()*.summary*())*#%%  
  
print*(*est\_plural.fit*()*.summary*())*#%%  
  
print*(*t.ttest\_ind*(*df*[***"y"***]*, df*[***"x1"***]))*print*(*t.ttest\_ind*(*df*[***"y"***]*, df*[***"x2"***]))*print*(*t.ttest\_ind*(*df*[***"y"***]*, df*[***"x3"***]))*#%%  
  
model = smmy.OLS*(*y, x1\_x2\_x3\_const*)*results = model.fit*()*results.summary*()*#%%  
  
predictions = results.predict*(*x1\_x2\_x3\_const*)*predictions  
  
#%%  
  
residuals = results.resid  
  
  
#%%  
  
#DWtest  
dw = smd.stattools.durbin\_watson*(*residuals*)*dw  
  
#%%  
  
#BGtest  
smd.diagnostic.acorr\_breusch\_godfrey*(*results*)*#%%  
  
#GQtest  
gq = sms.het\_goldfeldquandt*(*residuals,results.model.exog*)*gq  
  
#%%  
  
#BPtest  
name=*[***'BP'**,**'p-value'***]*bp=sms.het\_breuschpagan*(*residuals,results.model.exog*)*lzip*(*name,bp*)*#%%  
  
  
  
#%%