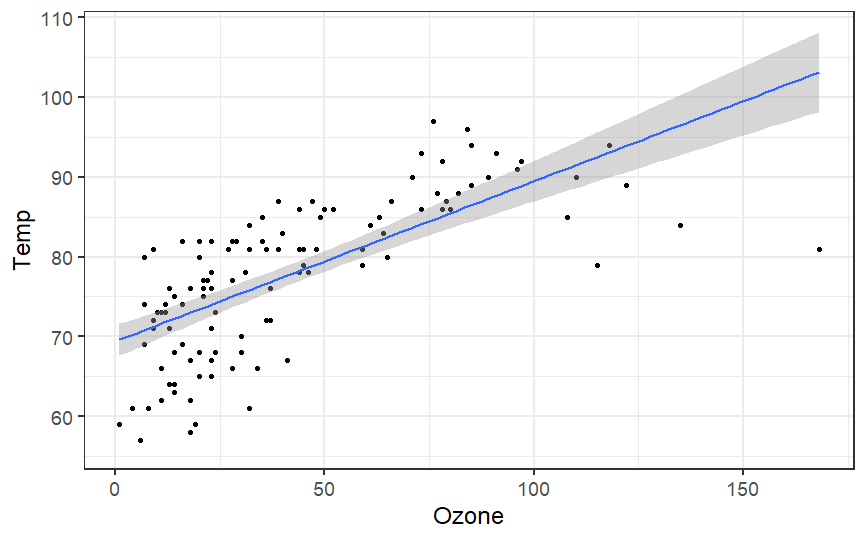
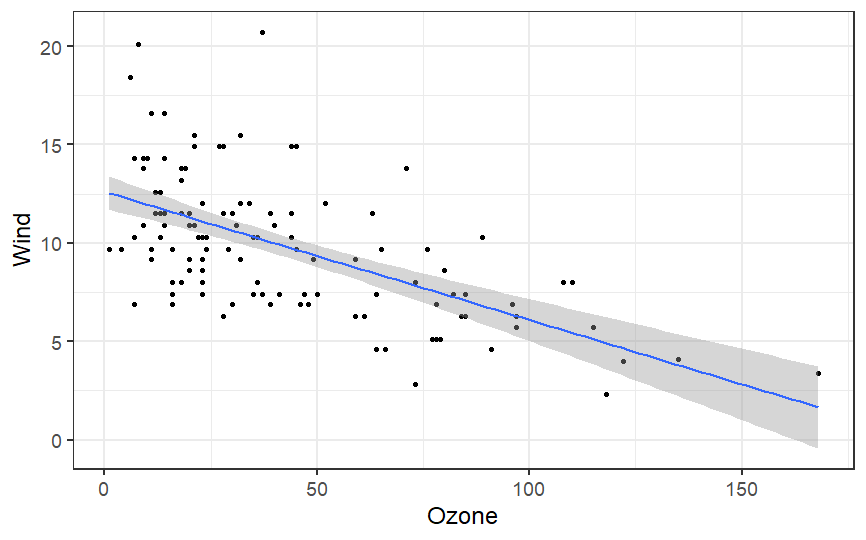
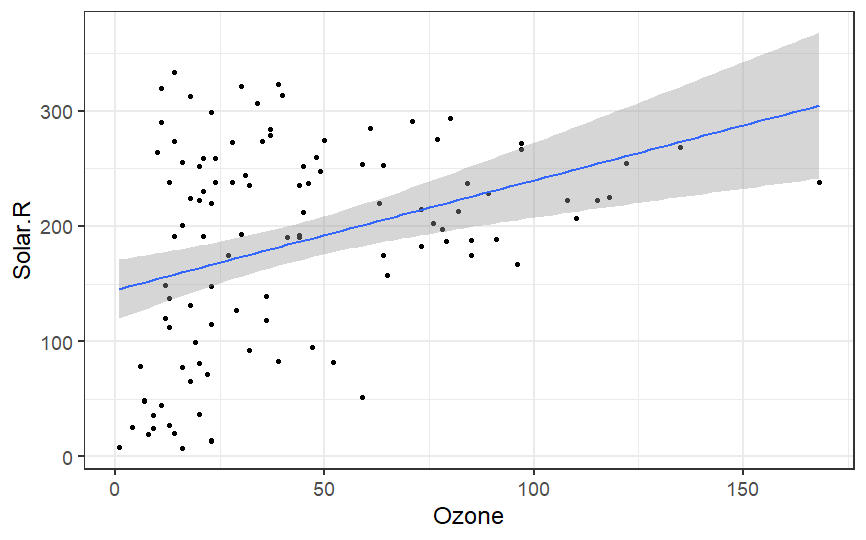
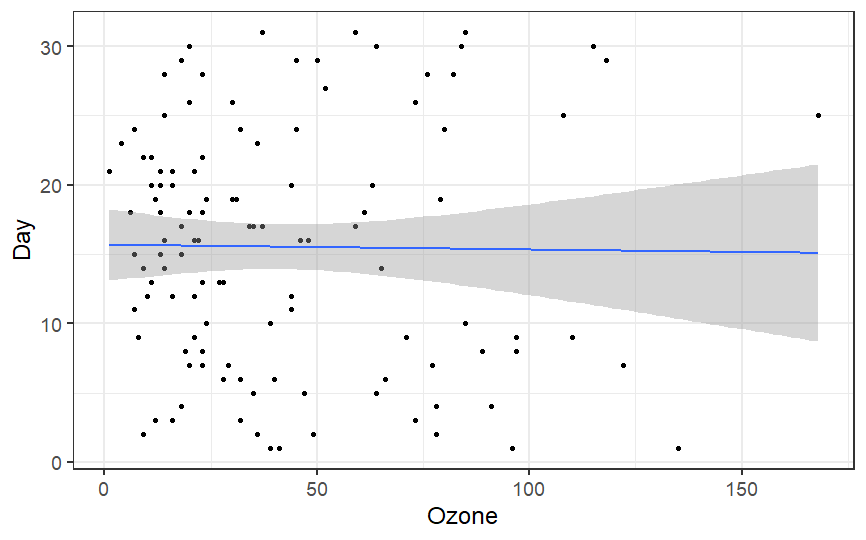
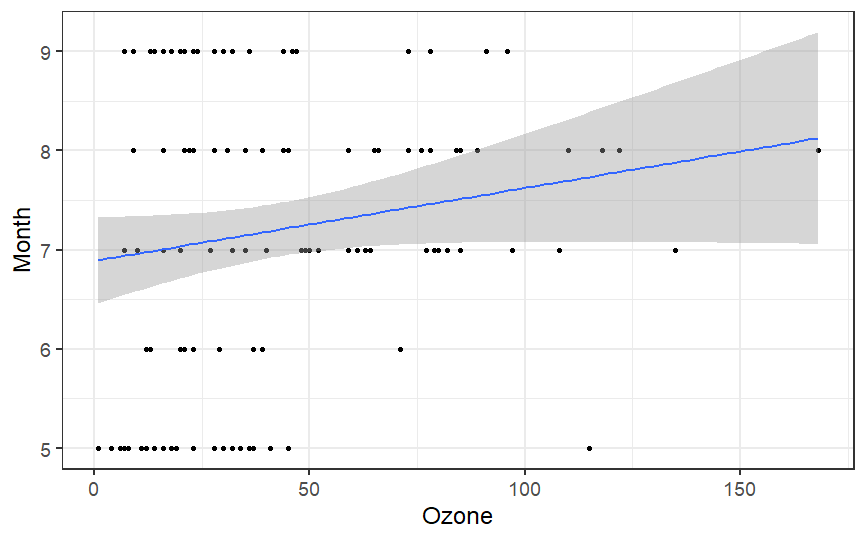
1. Графики корреляционного поля для парной регрессии.



1. Уравнения парной и множественной регрессии.

model$coefficients

(Intercept) Solar.R

18.599 0.127

> model <- lm(data=d, Ozone~Wind)

> model$coefficients

(Intercept) Wind

96.87 -5.55

> model <- lm(data=d, Ozone~Temp)

> model$coefficients

(Intercept) Temp

-147.00 2.43

> model <- lm(data=d, Ozone~Month)

> model$coefficients

(Intercept) Month

15.66 3.68

> model <- lm(data=d, Ozone~Day)

> model$coefficients

(Intercept) Day

* 1. -0.0499

1. Полный набор расчетов по моделям, получаемый с помощью команды summary.

Residuals:

Min 1Q Median 3Q Max

-48.29 -21.36 -8.86 16.37 119.14

Coefficients:

Estimate Std. Error t value Pr(>|t|)

(Intercept) 18.5987 6.7479 2.76 0.00686 \*\*

Solar.R 0.1272 0.0328 3.88 0.00018 \*\*\*

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

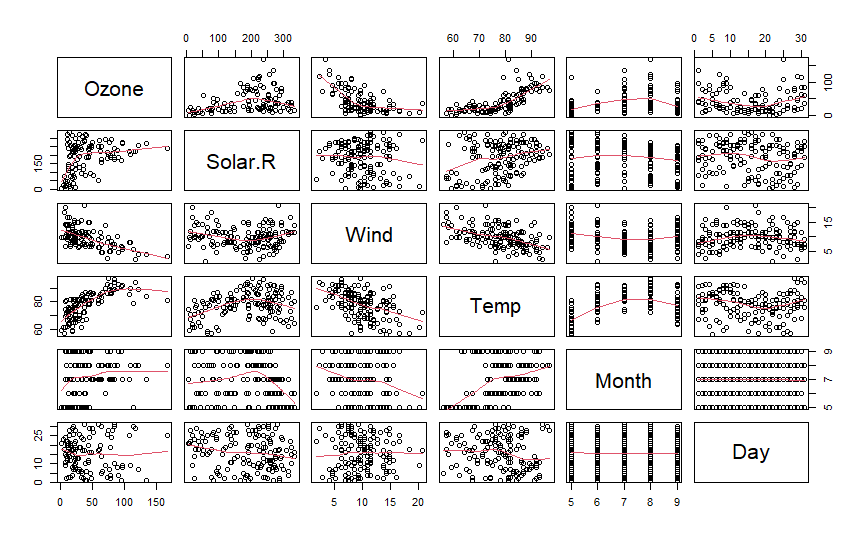
Residual standard error: 31.3 on 109 degrees of freedom

(42 пропущенных наблюдений удалены)

Multiple R-squared: 0.121, Adjusted R-squared: 0.113

F-statistic: 15.1 on 1 and 109 DF, p-value: 0.000179

1. Для модели множественной регрессии – корреляционную матрицу, диаграммы рассеяния и сглаженные распределения, полученные с помощью функции ggpairs.



Выводы по работе:

В ходе выполнения работы, я научился:

1. Загрузить набор данных
2. Строить график корреляционного поля для каждого фактора.
3. Строить уравнение парной линейной регрессии для каждого фактора.
4. Проверить значимость каждого из полученных уравнений регрессии.
5. Строить прогнозы по каждому из уравнений парной регрессии для заданных в варианте значений факторов.
6. Строить уравнение множественной линейной регрессии и получить корреляционную матрицу.
7. Строить прогноз по уравнению множественной регрессии для заданных в варианте значений факторов.

library("psych")# описательные статистики

library("lmtest") # тестирование гипотез в линейных моделях

library("ggplot2")# графики

library("dplyr") # манипуляции с данными

library ("MASS") # подгонка распределений

library(datasets)

data(airquality)

str(airquality)

#Корреляционное поле

d <- airquality

qplot(data=d, Ozone , Solar.R)

qplot(data=d, Ozone , Wind)

qplot(data=d, Ozone , Temp)

qplot(data=d, Ozone , Month)

qplot(data=d, Ozone , Day)

library("psych")# описательные статистики

library("lmtest") # тестирование гипотез в линейных моделях

library("ggplot2")# графики

library("dplyr") # манипуляции с данными

library ("MASS") # подгонка распределений

library(datasets)

data(airquality)

str(airquality)

#уравнение парной линейной регрессии

d <- airquality

model <- lm(data=d, Ozone~Solar.R)

model$coefficients

model <- lm(data=d, Ozone~Wind)

model$coefficients

model <- lm(data=d, Ozone~Temp)

model$coefficients

model <- lm(data=d, Ozone~Month)

model$coefficients

model <- lm(data=d, Ozone~Day)

model$coefficients

library("psych")# описательные статистики

library("lmtest") # тестирование гипотез в линейных моделях

library("ggplot2")# графики

library("dplyr") # манипуляции с данными

library ("MASS") # подгонка распределений

library(datasets)

data(airquality)

str(airquality)

#уравнение парной линейной регрессии

d <- airquality

qplot(data = d, Ozone, Solar.R) + stat\_smooth(method="lm", level = 0.95) + theme\_bw(base\_size = 18)

qplot(data = d, Ozone, Wind) + stat\_smooth(method="lm", level = 0.95) + theme\_bw(base\_size = 18)

qplot(data = d, Ozone, Temp) + stat\_smooth(method="lm", level = 0.95) + theme\_bw(base\_size = 18)

qplot(data = d, Ozone, Month) + stat\_smooth(method="lm", level = 0.95) + theme\_bw(base\_size = 18)

qplot(data = d, Ozone, Day) + stat\_smooth(method="lm", level = 0.95) + theme\_bw(base\_size = 18)

library("psych")# описательные статистики

library("lmtest") # тестирование гипотез в линейных моделях

library("ggplot2")# графики

library("dplyr") # манипуляции с данными

library ("MASS") # подгонка распределений

library(datasets)

data(airquality)

str(airquality)

#уравнение парной линейной регрессии

d <- airquality

model <- lm(data=d, Ozone~Solar.R)

nd <- data.frame(Solar.R=35,Wind = 8,3,Temp = 80)

predict(model,nd)

library("psych")# описательные статистики

library("lmtest") # тестирование гипотез в линейных моделях

library("ggplot2")# графики

library("dplyr") # манипуляции с данными

library ("MASS") # подгонка распределений

library(datasets)

data(airquality)

str(airquality)

#уравнение парной линейной регрессии

d <- airquality

model <- lm(data=d, Ozone~Solar.R)

nd <- data.frame(Solar.R=35,Wind = 8,3,Temp = 80)

predict(model,nd)

pairs(d, panel = panel.smooth)

library("psych")# описательные статистики

library("lmtest") # тестирование гипотез в линейных моделях

library("ggplot2")# графики

library("dplyr") # манипуляции с данными

library ("MASS") # подгонка распределений

library(datasets)

data(airquality)

str(airquality)

#уравнение парной линейной регрессии

d <- airquality

model <- lm(formula = Ozone ~ Solar.R + Wind + Temp ,

data = d)

nd2 <- data.frame(Solar.R=35,Wind = 8,3,Temp = 80)

predict(model,nd2)

library("psych")# описательные статистики

library("lmtest") # тестирование гипотез в линейных моделях

library("ggplot2")# графики

library("dplyr") # манипуляции с данными

library ("MASS") # подгонка распределений

library(datasets)

data(airquality)

str(airquality)

#уравнение парной линейной регрессии

d <- airquality

model <- lm(formula = Ozone ~ Solar.R + Wind + Temp ,

data = d)

nd2 <- data.frame(Solar.R=35,Wind = 8,3,Temp = 80)

predict(model,nd2)