R version 4.1.0 (2021-05-18) -- "Camp Pontanezen"

Copyright (C) 2021 The R Foundation for Statistical Computing

Platform: x86\_64-w64-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.

You are welcome to redistribute it under certain conditions.

Type 'license()' or 'licence()' for distribution details.

R is a collaborative project with many contributors.

Type 'contributors()' for more information and

'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or

'help.start()' for an HTML browser interface to help.

Type 'q()' to quit R.

> library(readxl)

> naeem\_data <- read\_excel("naeem\_data.xlsx")

> View(naeem\_data)

> #

> # Template for Multiple Regression Project

> #

> # Set up with the dependent variable as the first column

> # and the three independent variables in columns 2, 3, & 4

> #

> # The code will use column 1 as y, column 2 as x1, column 3 as x2

> # and column 4 as x3

> #

> # load library regclass for VIF function

> library(regclass)

Loading required package: bestglm

Loading required package: leaps

Loading required package: VGAM

Loading required package: stats4

Loading required package: splines

Loading required package: rpart

Loading required package: randomForest

randomForest 4.6-14

Type rfNews() to see new features/changes/bug fixes.

Important regclass change from 1.3:

All functions that had a . in the name now have an \_

all.correlations -> all\_correlations, cor.demo -> cor\_demo, etc.

Warning messages:

1: package ‘regclass’ was built under R version 4.1.1

2: package ‘bestglm’ was built under R version 4.1.1

3: package ‘leaps’ was built under R version 4.1.1

4: package ‘VGAM’ was built under R version 4.1.1

5: package ‘randomForest’ was built under R version 4.1.1

> # load library lmtest for Breusch-Pagan test

> library(lmtest)

Loading required package: zoo

Attaching package: ‘zoo’

The following objects are masked from ‘package:base’:

as.Date, as.Date.numeric

Attaching package: ‘lmtest’

The following object is masked from ‘package:VGAM’:

lrtest

Warning messages:

1: package ‘lmtest’ was built under R version 4.1.1

2: package ‘zoo’ was built under R version 4.1.1

> # In RStudio, use File, Import Dataset, From Excel...

> # to get Excel data file

> ##########

> #

> # NOTE: Replace RegProj with the name of your imported Excel file

> #

> str(naeem\_data)

tibble [16 x 4] (S3: tbl\_df/tbl/data.frame)

$ soybean10: num [1:16] 7.31 8.05 6.94 7.6 7.62 ...

$ OM : num [1:16] 3.1 2.4 3 2.4 3.2 3.5 3 2.7 3.3 3.6 ...

$ P\_ppm : num [1:16] 367 247 216 170 296 247 179 227 204 470 ...

$ K\_ppm : num [1:16] 207 111 201 164 273 229 254 181 212 281 ...

> # To have most of our R code reuseable for future

> # analyses, we will use a data object called dataobj

> ##########

> # NOTE: Replace RegProj with the name of your imported Excel file

> #

> dataobj <- as.data.frame(naeem\_data)

> str(dataobj)

'data.frame': 16 obs. of 4 variables:

$ soybean10: num 7.31 8.05 6.94 7.6 7.62 ...

$ OM : num 3.1 2.4 3 2.4 3.2 3.5 3 2.7 3.3 3.6 ...

$ P\_ppm : num 367 247 216 170 296 247 179 227 204 470 ...

$ K\_ppm : num 207 111 201 164 273 229 254 181 212 281 ...

> par(mfrow = c(1, 1))

> plot(dataobj[,2:4],main="Crop Yield")

> plot(dataobj[,1:4],main="Crop Yield")

> library(ggplot2)

Keep up to date with changes at https://www.tidyverse.org/blog/

Attaching package: ‘ggplot2’

The following object is masked from ‘package:randomForest’:

margin

> library(Hmisc)

Loading required package: lattice

Attaching package: ‘lattice’

The following object is masked from ‘package:regclass’:

qq

Loading required package: survival

Loading required package: Formula

Attaching package: ‘Hmisc’

The following objects are masked from ‘package:base’:

format.pval, units

Warning messages:

1: package ‘Hmisc’ was built under R version 4.1.1

2: package ‘Formula’ was built under R version 4.1.1

> mat <- as.matrix(dataobj[,2:4])

> rcorr(mat, type="pearson")

OM P\_ppm K\_ppm

OM 1.00 0.44 0.08

P\_ppm 0.44 1.00 -0.11

K\_ppm 0.08 -0.11 1.00

n= 16

P

OM P\_ppm K\_ppm

OM 0.0841 0.7732

P\_ppm 0.0841 0.6951

K\_ppm 0.7732 0.6951

> model <- lm(soybean10 ~ OM + P\_ppm + K\_ppm, data=dataobj)

> summary(model)

Call:

lm(formula = soybean10 ~ OM + P\_ppm + K\_ppm, data = dataobj)

Residuals:

Min 1Q Median 3Q Max

-0.9138 -0.3804 0.1116 0.2901 0.9905

Coefficients:

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

(Intercept) 9.1256435 1.1054702 8.255 2.72e-06 \*\*\*

OM -0.5081346 0.3953361 -1.285 0.2229

P\_ppm 0.0004345 0.0017029 0.255 0.8029

K\_ppm -0.0036419 0.0013409 -2.716 0.0187 \*

---

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

Residual standard error: 0.5946 on 12 degrees of freedom

Multiple R-squared: 0.46, Adjusted R-squared: 0.325

F-statistic: 3.407 on 3 and 12 DF, p-value: 0.05322

> # confidence intervals on model parameters

> confint(model,level=0.95)

2.5 % 97.5 %

(Intercept) 6.717030819 11.5342562395

OM -1.369498032 0.3532288329

P\_ppm -0.003275748 0.0041447603

K\_ppm -0.006563479 -0.0007202754

> # confidence interval on E(y) for specified values of all

> # independent variables

> forecastdata = data.frame(OM=3.1,

+ P\_ppm=367,

+ K\_ppm=207)

> predict(model,newdata=forecastdata,interval="confidence")

fit lwr upr

1 6.956021 6.529714 7.382328

> # prediction interval on y for specified values of all

> # independent variables

> predict(model,newdata=forecastdata,interval="prediction")

fit lwr upr

1 6.956021 5.592245 8.319797

> par(mfrow = c(2, 2))

> plot(model, main="Crop Yield Prediction")

> # Shapiro-Wilk test for normality of errors and use alpha=0.01

> shapiro.test(resid(model))

Shapiro-Wilk normality test

data: resid(model)

W = 0.95215, p-value = 0.5245

> # Breusch-Pagan Test for a common error variance and use alpha=0.01

> bptest(model)

studentized Breusch-Pagan test

data: model

BP = 0.87592, df = 3, p-value = 0.8312

> par(mfrow = c(1, 1))

> # histogram of residuals

> hist(resid(model), main="Crop Yield Histogram of Residuals",xlab="Residuals")

> # boxplot of residuals

> boxplot(resid(model), main="Crop Yield")

> # Influential observations

> influence.measures(model)

Influence measures of

lm(formula = soybean10 ~ OM + P\_ppm + K\_ppm, data = dataobj) :

dfb.1\_ dfb.OM dfb.P\_pp dfb.K\_pp dffit cov.r cook.d hat inf

1 0.015823 -4.60e-02 0.132999 -1.03e-02 0.211377 1.392 1.18e-02 0.1083

2 0.455691 -3.47e-01 0.041682 -2.20e-01 0.515992 1.506 6.82e-02 0.2700

3 -0.003826 -3.01e-03 0.009642 4.33e-03 -0.015930 1.583 6.92e-05 0.1061

4 0.204989 -1.26e-01 -0.075594 -6.46e-02 0.238257 1.785 1.52e-02 0.2532

5 -0.126830 1.34e-01 -0.006315 1.82e-01 0.557016 0.484 6.34e-02 0.0777

6 0.333913 -5.31e-01 0.386355 8.98e-02 -0.676195 0.920 1.06e-01 0.1921

7 -0.105960 -1.83e-01 0.559944 -2.81e-02 -0.758508 0.581 1.21e-01 0.1482

8 0.128960 -7.31e-02 -0.040842 -5.03e-02 0.175400 1.491 8.23e-03 0.1227

9 -0.123187 3.46e-01 -0.404023 -1.23e-01 0.546537 1.110 7.28e-02 0.1860

10 -0.156479 5.58e-02 0.205045 8.21e-02 0.304809 1.916 2.49e-02 0.3131

11 -0.146930 6.06e-03 0.006161 5.81e-01 0.614577 6.444 1.02e-01 0.7882 \*

12 0.302118 -3.53e-01 -0.116374 3.27e-01 -0.661263 1.144 1.06e-01 0.2364

13 -0.084434 1.22e-01 -0.036056 -8.41e-02 0.161353 1.943 7.06e-03 0.2879

14 -0.112142 3.53e-01 -0.717715 2.80e-02 -0.780340 2.492 1.58e-01 0.5307 \*

15 -0.864016 9.16e-01 -0.249927 -3.39e-01 -1.081008 0.855 2.57e-01 0.3048

16 0.000211 4.31e-05 -0.000331 -1.04e-05 0.000881 1.530 2.12e-07 0.0746

> # Check for multicollinearity

> # The VIF function is in R package regclass

> VIF(model)

OM P\_ppm K\_ppm

1.272208 1.278869 1.031972

>