Data warehousing and mining

23.1.23

ABHIJIT MISHRA

2020BCS0094

ICS- 321

Goal

1. Visualizing data

1. Read the pupae data. Convert ’CO2_treatment’ to a factor. Inspect the levels of this factor variable.

2. Make a scatter plot of Frass vs. PupalWeight, with blue solid circles for a CO2 concentration of 280ppm and red for 400ppm. Also add a legend.

3. The problem with the above figure is that data for both temperature treatments is combined. Make two plots (either in a PDF, or two plots side by side), one with the ’ambient’ temperature treatment, one with ’elevated’.

4. In the above plot, make sure that the X and Y axis ranges are the same for both plots. Hint: use xlim and ylim

2. Statistics

1. When tossing a fair coin 10 times, find the probability of seeing no heads (Hint: this is a binomial distribution.)

2. Find the probability of seeing exactly 5 heads.

3. Simulate a sample of 100 random data points from a normal distribution with mean 100 and standard deviation 5, and store the result in a vector.

a) Plot a histogram and a boxplot of the vector you just created

b) Calculate the sample mean and standard deviation

c) Calculate the median and interquartile range.

d) Using the data above, test the hypothesis that the mean equals 100 (using t.test).

e) Test the hypothesis that mean equals 90

f) Repeat the above two tests using a Wilcoxon signed rank test. Compare the p-values with those from the t-tests you just did.

3. Simple linear regression

1. For this question, use the pupae data. Perform a simple linear regression of Frass on PupalWeight. Produce and inspect the following:

a) Plots of the data.

b) Summary of the model.

c) Diagnostic plots.

Note

I have used built in software to upload this code on RPubs, code with output is printed in a neat and clean way.

https://rpubs.com/Panda_250/994522

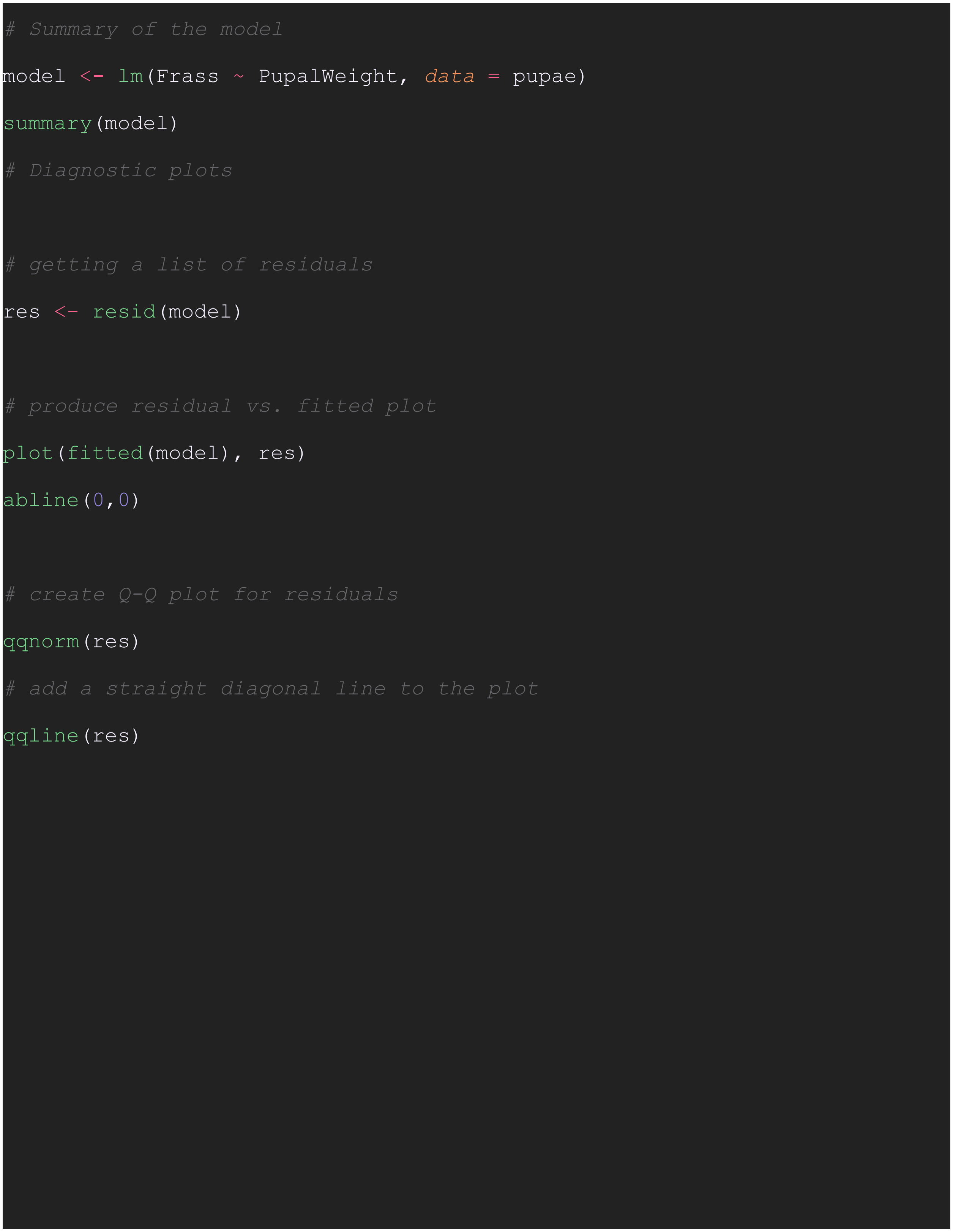
Code

```
# Find all the files on
# https://github.com/Abhijit25Mishra/R-Lab--ICS-321-
# https://rpubs.com/Panda_250
print("Abhijit Mishra")

# setting the working directory to use the csv file
print(getwd())
setwd("C:/Users/ASUS/OneDrive/Desktop/Study-Material/IIIT-Kottayam/SEM-6/D
ata warehousing and mining ICS 321/Lab/Lab-3")

# reading the csv file
pupae <- read.csv("pupae.csv")
```

```
# Visualizing data

# Convert ’CO2_treatment’ to a factor. Inspect the.
# levels of this factor variable.
pupae$CO2_treatment <- as.factor(pupae$CO2_treatment)
levels(pupae$CO2_treatment)

# Make a scatter plot of Frass vs. PupalWeight, with blue solid circles
# for a CO2 concentration of 280ppm and red for 400ppm. Also add a legend.

palette(c("blue", "red"))
plot(Frass ~ PupalWeight, col = CO2_treatment, data = pupae, pch = 19)
legend("topleft", levels(pupae$CO2_treatment), col = palette(), pch = 19)

# The problem with the above figure is that data for both temperature
# treatments is combined. Make two plots (either in a PDF, or two plots
# side by side), one with the ’ambient’ temperature treatment, one
with’elevated’.

# Solution 1: separate windows windows()
plot(Frass ~ PupalWeight, col = CO2_treatment, data = subset(pupae,
CO2_treatment =="280"), pch = 19)
plot(Frass ~ PupalWeight, col = CO2_treatment, data = subset(pupae,
CO2_treatment =="400"), pch = 19)

# solution 2: side by side
par(mfrow = c(1, 2))
```

```
plot(Frass ~ PupalWeight, col = CO2_treatment, data = subset(pupae,
T_treatment =="ambient"), pch = 19)
plot(Frass ~ PupalWeight, col = CO2_treatment, data = subset(pupae,
T_treatment =="elevated"), pch = 19)

# in the above plot, make sure that the X and Y axis ranges are the same
# for both plots. Hint: use xlim and ylim

par(mfrow = c(1, 2))
plot(Frass ~ PupalWeight, col = CO2_treatment, data = subset(pupae,
T_treatment =="ambient"), xlim = c(0, 0.5), ylim = c(0, 3.5), pch = 19)
plot(Frass ~ PupalWeight, col = CO2_treatment, data = subset(pupae,
T_treatment =="elevated"), xlim = c(0, 0.5), ylim = c(0, 3.5), pch = 19)

# Statistics

# dbinom finds the probability of 'x' occurrences (0 in this case) when we
# repeat N ('size') events (here, 10), each with probability 'prob' (here,
0.5).

# When tossing a fair coin 10 times, find the probability of seeing no
heads
dbinom(x = 0, size = 10, prob = 0.5)
# Find the probability of seeing exactly 5 heads.
dbinom(x = 5, size = 10, prob = 0.5)

# Simulate a sample of 100 random data points from a normal
# distribution with mean 100 and standard deviation 5, and store the
# result in a vector.
```

```
r <- rnorm(100, mean = 100, sd = 5)


# Plot a histogram and a boxplot of the vector
par(mfrow=c(1,2))
hist(r)
boxplot(r)


# Calculate the sample mean, standard deviation, median, Interquartile range
mean(r)
sd(r)
median(r)
IQR(r)


# test the hypothesis that the mean = 100 and mean = 90 using
# t-test and wilcox-test
t.test(r,mu=100)
t.test(r,mu=90)
wilcox.test(r,mu=100)
wilcox.test(r,mu=90)


# Simple linear regression
# Perform a simple linear regression of Frass on PupalWeight


# Plots of the data
par(mfrow = c(1,1))
plot(Frass ~ PupalWeight,data = pupae)
```



```r
# Summary of the model

model <- lm(Frass ~ PupalWeight, data = pupae)

summary(model)

# Diagnostic plots


# getting a list of residuals

res <- resid(model)


# produce residual vs. fitted plot

plot(fitted(model), res)

abline(0,0)


# create Q-Q plot for residuals

qqnorm(res)

# add a straight diagonal line to the plot

qqline(res)
```