Laboratory Exercise Week 14

Brian Tipton

11/29/23

Directions:

- Write your R code inside the code chunks after each question.
- Write your answer comments after the # sign.
- To generate the word document output, click the button `Knit` and wait for the word document to appear.
- RStudio will prompt you (only once) to install the `knitr` package.
- Submit your completed laboratory exercise using Blackboard's Turnitin feature. Your Turnitin upload link is found on your Blackboard Course shell under the Laboratory folder.

For this exercise, you will need to use the packages `mosaic` and `dplyr` to find numerical and graphical summaries.

```r
# install packages if necessary
if (!require(mosaic)) install.packages(`mosaic`)
if (!require(dplyr)) install.packages(`dplyr`)
# load the package in R
library(mosaic) # load the package mosaic to use its functions
library(dplyr) # load the package dplyr to use data management functions
```

1. Researchers found that the speed of a prey (twips/s) and the length of prey (twips x 100) are good predictors of the time (s) required to catch a prey. (A twip is a measure of distance used by programmers). Data were collected in an experiment in which subjects were asked to "catch" an animal prey moving across his or her computer screen by clicking on it with the mouse. The investigators varied the length of the prey and the speed with which prey moved across the screen.

```r
prey <- read.csv("https://www.siue.edu/~jpailde/prey.csv")
prey
```

```
##    Prey.Length Prey.Speed Catch.Time
## 1            7         20        1.1
## 2            6         20        1.2
## 3            5         20        1.2
## 4            4         20        1.4
## 5            3         20        1.5
## 6            3         40        1.4
## 7            4         40        1.4
## 8            6         40        1.3
## 9            7         40        1.3
## 10           7         80        1.4
## 11           6         60        1.4
## 12           5         80        1.4
## 13           7        100        1.4
```

```
## 14              6          100        1.4
## 15              7          120        1.7
## 16              5           80        1.5
## 17              3           80        1.4
## 18              6          100        1.5
## 19              3          120        1.9
```

i) Fit a multiple regression model for predicting catch time using prey length and speed as predictors.

ii) Construct 95% confidence interval for the regression slopes of each predictor. Interpret your result. Will the interpretation change if you change the confidence level to 90% and 99%?

iii) Predict the catch time for two animals whose lengths are 4 and 6; and whose speeds are 30 and 60, respectively. State your result in paragraph form including the associated prediction intervals.

iv) Is the multiple regression model useful for predicting catch time? Use `R2`, `adj-R2`, and test the relevant hypothesis using `alpha = 0.05`. State your conclusion.

v) The primary researchers suggest that a simple regression model with the single predictor `x = length/speed` might be a better model for predicting catch time. Calculate and add the `x` values to the data using the function `mutate` in the package `dplyr`. Fit a simple linear regression model using the new variable/column `x`.

vi) Which of the two models considered (the multiple regression model in part (i) or the simple regression model in part (v)) would you recommend for predicting catch time? Justify your choice.

Code chunk

```r
# start your code
#i)
model1 <- lm(Catch.Time ~ Prey.Length + Prey.Speed, data = prey)
print(model1)
```

```
##
## Call:
## lm(formula = Catch.Time ~ Prey.Length + Prey.Speed, data = prey)
##
## Coefficients:
## (Intercept)  Prey.Length   Prey.Speed
##    1.451651    -0.053543     0.003875
```

```r
#ii)
confint(model1, level = 0.95)
```

```
##                      2.5 %       97.5 %
## (Intercept)    1.266083854   1.637217568
## Prey.Length   -0.086057599  -0.021028381
## Prey.Speed     0.002494635   0.005256103
```

```r
confint(model1, level = 0.90)
```

```
##                      5 %         95 %
## (Intercept)    1.298824004   1.604477418
## Prey.Length   -0.080320944  -0.026765037
## Prey.Speed     0.002738242   0.005012496
```

```r
confint(model1, level = 0.99)
```

```
##                   0.5 %        99.5 %
## (Intercept)    1.19597879   1.707322635
## Prey.Length  -0.09834125  -0.008744728
## Prey.Speed    0.00197301   0.005777729
```

```r
# iii)
predict(
  model1,
  interval = "prediction",
  level = 0.95,
  newdata = data.frame(Prey.Length = c(4, 6),
                       Prey.Speed = c(30, 60))
)
```

```
##        fit      lwr      upr
## 1 1.353740 1.133718 1.573762
## 2 1.362915 1.148635 1.577195
```

```r
# The above predicted method shows that
# a) A prey of length 4, speed of 30, the catch time will be
# 1.353740 with a lower bound of 1.133718 and upper bound of 1.573762
# b) A prey of length 6, speed of 60, the catch time will be
# 1.362915 with a lower bound of 1.148635 and upper bound of 1.577195


# iv)
msummary(model1)
```

```
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)   1.4516507  0.0875354  16.584 1.68e-11 ***
## Prey.Length  -0.0535430  0.0153378  -3.491  0.00302 **
## Prey.Speed    0.0038754  0.0006513   5.950 2.04e-05 ***
##
## Residual standard error: 0.09787 on 16 degrees of freedom
## Multiple R-squared:  0.7253, Adjusted R-squared:  0.691
## F-statistic: 21.12 on 2 and 16 DF,  p-value: 3.243e-05
```

```r
# v)
prey <- prey %>%
  mutate(x = Prey.Length / Prey.Speed)
model2 <- lm(Catch.Time ~ x, data = prey)
print(model2)
```

```
##
## Call:
## lm(formula = Catch.Time ~ x, data = prey)
##
## Coefficients:
## (Intercept)            x
##       1.586       -1.409
```

```r
#vi)
msummary(model2)
```

```
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   1.58648    0.04815  32.948  < 2e-16 ***
```

```
## x            -1.40864    0.31314  -4.498 0.000317 ***
##
## Residual standard error: 0.1224 on 17 degrees of freedom
## Multiple R-squared:  0.5434, Adjusted R-squared:  0.5166
## F-statistic: 20.24 on 1 and 17 DF,  p-value: 0.0003168
# Model1 (Multiple Regression Model) would be the better choice.

# last R code line
```