Regression_HW3_Warren_Geither

Warren Geither

10/4/2020

Problem 1

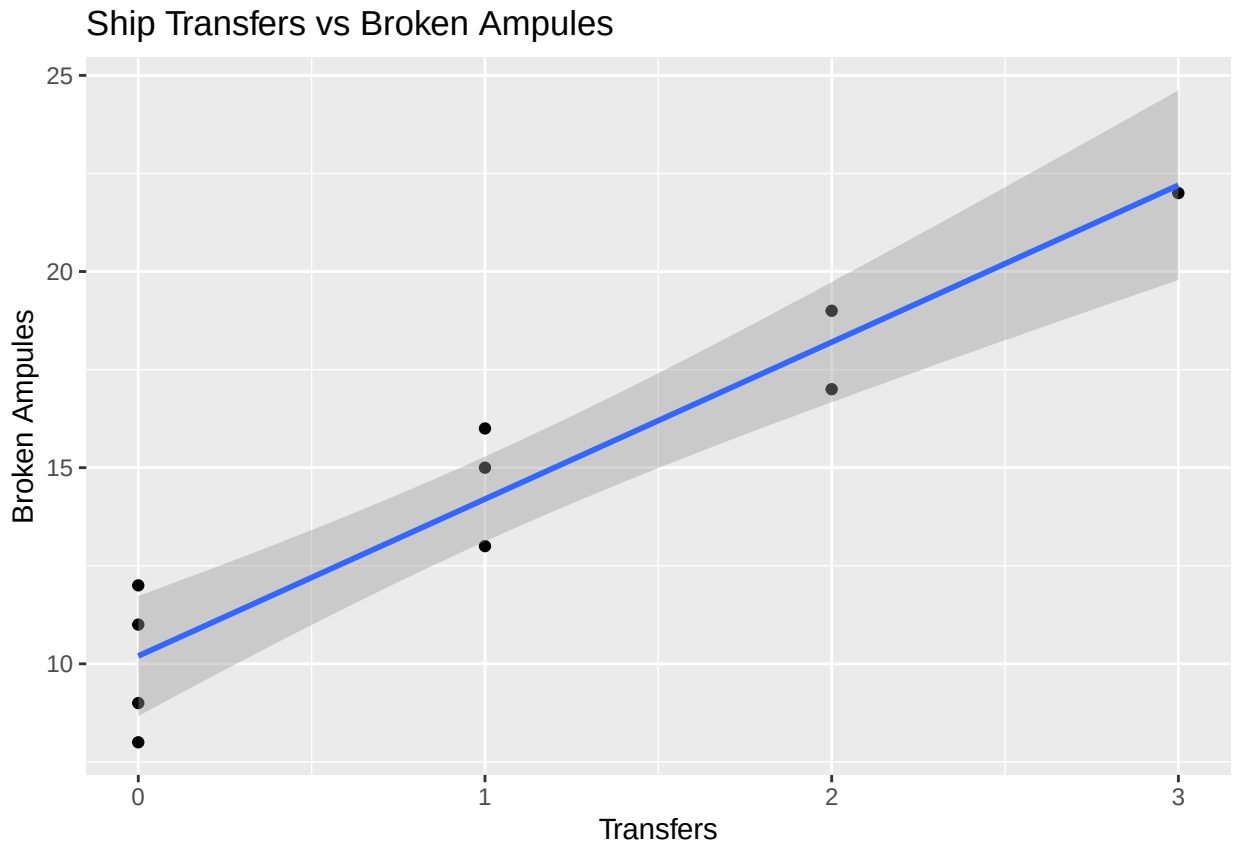
1.a)

```
# create dataframe
freight_data_df <- data.frame(ship_route = c(1,0,2,0,3,1,0,1,2,0)
                              , ampules = c(16,9,17,12,22,13,8,15,19,11))

# fit model
lmfit <- lm(ampules~ship_route, freight_data_df)

# plot scatterplot and estimated regression line
ggplot(freight_data_df, aes(x=ship_route, y=ampules)) +
  geom_point()+
  geom_smooth(method=lm) +
  ggtitle("Ship Transfers vs Broken Ampules") +
  xlab("Transfers") +
  ylab("Broken Ampules")
```

```
## `geom_smooth()` using formula 'y ~ x'
```



The linear regression function does appear to have a good fit

1.b)

```
# print summary to get coeffienient estimates
model_summary <- summary(lmfit)
model_summary
```

```
##
## Call:
## lm(formula = ampules ~ ship_route, data = freight_data_df)
##
## Residuals:
##    Min     1Q Median     3Q    Max
##   -2.2   -1.2    0.3    0.8    1.8
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  10.2000     0.6633  15.377 3.18e-07 ***
## ship_route    4.0000     0.4690   8.528 2.75e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.483 on 8 degrees of freedom
## Multiple R-squared:  0.9009, Adjusted R-squared:  0.8885
```

```
## F-statistic: 72.73 on 1 and 8 DF,  p-value: 2.749e-05
```

```r
# plug in values for beta_hats
b_0 <- 10.2
b_1 <- 4

# estimate value at x=1
y1 <- b_0 + b_1*1

# print results
print(paste0("Point estimate for X=1: ", y1 ))
```

```
## [1] "Point estimate for X=1: 14.2"
```

1.c)

```r
# get point estimate for X=2
y2 <- b_0 + b_1*2

# get increase
delta <- y2 - y1

print(paste0("Increase of broken Ampules from X=1 to X=2: ", delta))
```

```
## [1] "Increase of broken Ampules from X=1 to X=2: 4"
```

1.d)

```r
# find x_bar
x_bar <- mean(freight_data_df$ship_route)

print(paste0("x_bar: ", x_bar))
```

```
## [1] "x_bar: 1"
```

```r
# find y_bar
y_bar <- mean(freight_data_df$ampules)

print(paste0("y_bar: ", y_bar))
```

```
## [1] "y_bar: 14.2"
```

As shown in the plot in part a and the calculation in part b. The line does run through this point.

Problem 2

2.a)

```r
# 95% confidence interval for beta_1
confint(lmfit, "ship_route", level=0.95)
```

```
##                2.5 %   97.5 %
## ship_route 2.918388 5.081612
```

Interpretation of interval: We are 95% confident that true beta_1 lies within this interval. Meaning if we drew 100 samples from the same experiment, 95% of the samples would generate intervals that contain the true beta_1

2.b)

Referring back to the summary in problem 1 part a since it includes a 2 sided t-test.

Ho: beta_1 = 0 Ha: beta_1 != 0 alpha = .05 p_val = 0.0000275

The p value is the probability of viewing results at least as extreme as what we observed under the assumption that the null is true. i.e. if there was no linear relationship, there would be a 0.00275% chance of seeing these results or more extreme results.

Since the p value is less than our alpha of 0.05, there is statistically significant evidence to reject the null hypothesis and say there is a linear relationship between ship route changes and number of ampules broken.

2.c)

```r
# one sided t-test, gets p-values for coefficients
pt(coef(model_summary)[, 3], lmfit$df, lower = FALSE)
```

```
##  (Intercept)   ship_route
## 1.589137e-07 1.374335e-05
```

```r
# stack exchange reference
# https://stats.stackexchange.com/questions/325354/if-and-how-to-use-one-tailed-testing-in-multiple-reg
```

Ho: beta_1 = 0 Ha: beta_1 > 0 alpha = .05 p_val = 0.0000137

The p value is the probability of viewing results at least as extreme as what we observed under the assumption that the null is true. i.e. if there was no linear relationship, there would be a 0.00137% chance of seeing these results or more extreme results.

Since the p value is less than our alpha of 0.05, there is statistically significant evidence to reject the null hypothesis and say there does exist a postive linear relationship between ship route changes and number of ampules broken.

2.d)

```r
# 95% confidence interval for beta_0
confint(lmfit, "(Intercept)", level=0.95)
```

```
##                2.5 %   97.5 %
## (Intercept) 8.67037 11.72963
```

Interpretation of interval: We are 95% confident that true mean of broken ampules lies within this interval (8.6, 11.7). Meaning if we drew 100 samples from the same experiment, 95% of the samples would generate intervals that contain the true beta_0.

2.e)

```
# extract standard error
b_0_sigma <- model_summary$coefficients[1,2]

# sample size
n <- 10

# calc Sxx
sxx <- sum((freight_data_df$ship_route - x_bar)^2)

# test stat formula
test_stat <- (b_0 - 9)/sqrt(b_0_sigma^2*((1/n)+(x_bar^2)/sxx))

# get p-value
p_val <- pt(test_stat, df = 8, lower = FALSE)

print(paste0("The p_value is: ", p_val))
```

```
## [1] "The p_value is: 0.00185442801747058"
```

Ho: b_0 = 9.0 Ha: b_0 > 9.0

Since the p value is less than our alpha of 0.025, there is statistically significant evidence to reject the null hypothesis and say the mean number of broken ampules is above 9.

Problem 3

3.a)

```
# get data ready for x=2
new_data1 <- data.frame(ship_route=2)

# 99% condifence interval for data
predict(lmfit, new_data1, interval = "confidence", level = 0.99)
```

```
##    fit      lwr      upr
## 1 18.2 15.97429 20.42571
```

```
# settoing x = 4
new_data2 <- data.frame(ship_route=4)

# 99% c.i. for x = 2
predict(lmfit, new_data2, interval = "confidence", level = 0.99)
```

```
##    fit      lwr      upr
## 1 26.2 21.22316 31.17684
```

For a future shipment where X=2, we are 99% confident that the number of broken ampules would fall in the range of 15.9 and 20.42. Meaning if we took 100 observations, 99/100 would produce intervals that contain the true number of breakages.

For a future shipment where X=4, we are 99% confident that the number of broken ampules would fall in the range of 21.22 and 31.17.

3.b)

```
# set data for preditcion
predict_data <- data.frame(ship_route=2)

# make prediction
predict(lmfit, predict_data, interval = "predict", level = 0.99)
```

```
##    fit      lwr      upr
## 1 18.2 12.74814 23.65186
```

Given that there will be 2 transfers, we are 99% confident that the number of ampules broken will be between 12.7 to 23.65.

3.c)

```
# set data for preditcion
predict_data <- data.frame(ship_route=c(2,2,2))

# make prediction
predict(lmfit, predict_data, interval = "predict", level = 0.99, df=8)
```

```
##    fit      lwr      upr
## 1 18.2 12.74814 23.65186
## 2 18.2 12.74814 23.65186
## 3 18.2 12.74814 23.65186
```

Since each shipment is independent, we can produce 3 prediction intervals and then take the average of all of them. This would result in a prediction interval of (12.7,23.65)

Problem 4

4.a)

```
anova_table <- anova(lmfit)
anova_table
```

```
## Analysis of Variance Table
##
## Response: ampules
##            Df Sum Sq Mean Sq F value    Pr(>F)
## ship_route  1  160.0   160.0  72.727 2.749e-05 ***
## Residuals   8   17.6     2.2
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The sum of squares column is additive since it can be broken down into its indivridual elements and summed together, and they also sum to SST. Also degrees of freedom column sums up to the degrees of freedom for SST

4.b)

Ho: No linear association Ha: linear association

alpha = 0.05 p-val = 0.00002749

Since the p-val is less than our alpha of 0.05 there is statistically significant evidence to reject the null and say that there is a linear association between ship transfers and broken ampules.

4.c)

```
f_val <- summary(lmfit)$"fstatistic"[1]

t_val <- sqrt(summary(lmfit)$"fstatistic"[1])

print(paste0("F* is: ", f_val))
```

```
## [1] "F* is: 72.7272727272727"
```

```
print(paste0("sqrt(F*): ", t_val))
```

```
## [1] "sqrt(F*): 8.52802865422442"
```

t* in problem1 part b is the square root of F*

4.d)

```
# grab r squared value
r_squared <- summary(lmfit)$r.squared

# take square root to get r
r <- sqrt(r_squared)

# print results
print(paste0("r squared is: ", r_squared))
```

```
## [1] "r squared is: 0.900900900900901"
```

```
print(paste0("r is: ", r))
```

```
## [1] "r is: 0.949157995752499"
```

90% of the proportion of variation in the number of amupules broken is captured by ship transfers.