Simple Linear Regression - Lab Activity

Katlyn H. Degamo

2023-06-23

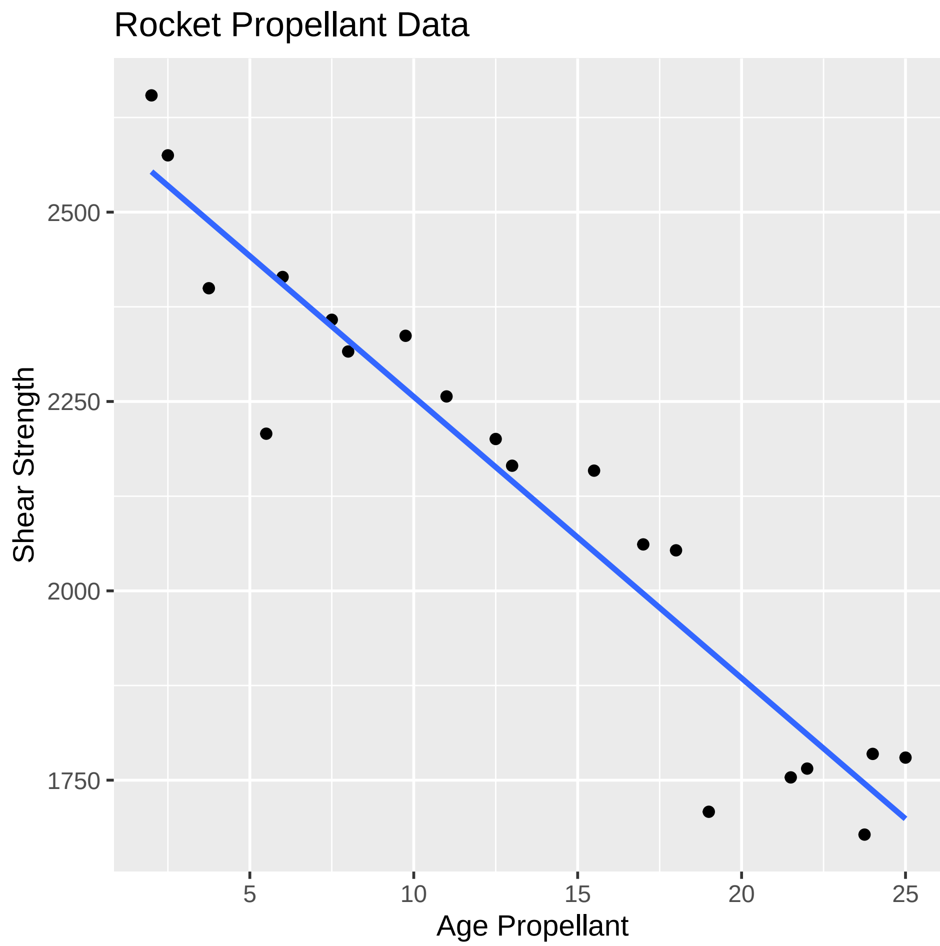
1. The Rocket Propellant Data

A rocket motor is manufactured by bonding an igniter propellant and a sustainer propellant together inside a metal housing. The shear strength of the bond between the two types of propellant is an important quality characteristic. It is suspected that shear strength is related to the age in weeks of the batch of sustainer propellant. Twenty observations on shear strength and the age of the corresponding batch of propellant have been collected.

a. Create a scatter diagram for the data.

```r
# We used the Rocket Propellant Data
# Let  x = Propellant Age, y = Shear Strength

Propellant_age <- c(15.50, 23.75, 8.00, 17.00, 5.50, 19.00, 24.00, 2.50, 7.50,
                    11.00, 13.00, 3.75, 25.00, 9.75, 22.00, 18.00, 6.00, 12.50,
                    2.00, 21.50)
Shear_strength <- c(2158.70, 1678.15, 2316.00, 2061.30, 2207.50, 1708.30,
                    1784.70, 2575.00, 2357.90, 2256.70, 2165.20, 2399.55,
                    1779.80, 2336.75, 1765.30, 2053.50, 2414.40, 2200.50,
                    2654.20, 1753.70)

library(ggplot2)

# Display Scatter Diagram

scatter_diagram <- ggplot(data = NULL,
                          aes(x = Propellant_age, y= Shear_strength)) +
                          geom_point() + geom_smooth(method = "lm",
                          se = FALSE) +
                          labs(title = "Rocket Propellant Data",
                          x = " Age Propellant", y = "Shear Strength")

print(scatter_diagram)
```

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



2. Least-Squares Estimation of the Parameters

```
# Create the dataframe
rocket_propellant_data <- data.frame(Propellant_age = Propellant_age,
                                     Shear_strength = Shear_strength)
# Calculate the model
# Fit a linear regression model
model <- lm(data = rocket_propellant_data,
         formula = Shear_strength ~ Propellant_age)

# view the names of the objects in the model
names(model)
```

```
##  [1] "coefficients"  "residuals"     "effects"       "rank"
##  [5] "fitted.values" "assign"        "qr"            "df.residual"
##  [9] "xlevels"       "call"          "terms"         "model"
```

a. The least-squares fit is

```
# Access the coefficients
model$coefficients
```

```
##    (Intercept) Propellant_age
```

```
##     2627.82236      -37.15359
```

```r
# Obtain a summary of the model
summary(model)
```

```
##
## Call:
## lm(formula = Shear_strength ~ Propellant_age, data = rocket_propellant_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -215.98  -50.68   28.74   66.61  106.76
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)    2627.822     44.184   59.48  < 2e-16 ***
## Propellant_age  -37.154      2.889  -12.86 1.64e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 96.11 on 18 degrees of freedom
## Multiple R-squared:  0.9018, Adjusted R-squared:  0.8964
## F-statistic: 165.4 on 1 and 18 DF,  p-value: 1.643e-10
```

```r
# Stored the summary of the model
model_summary <- summary(model)
```

b. The estimate of sigma^2

```r
# Retrieves the residual standard error from the model summary stored
model_summary$sigma
```

```
## [1] 96.10609
```

```r
#Calculates the squared residual standard error, also known as the residual variance.
sigma_squared <- (model_summary$sigma)^2

print(sigma_squared)
```

```
## [1] 9236.381
```

```r
# Access the residual standard error
residuals <- residuals(model)

# Calculate the residual standard error
sigma_squared <- sum(residuals^2)/ (length(residuals)- length(coefficients(model)))

print(sigma_squared)
```

```
## [1] 9236.381
```

3. Hypothesis Testing on the Slope and Intercept

a. Test for significance of regression in the rocket propellant regression model.

```r
# Extract the coefficients from the model summary corresponding to the
# "Propellant_age" predictor variable.

model_summary$coefficients[Propellant_age]
```

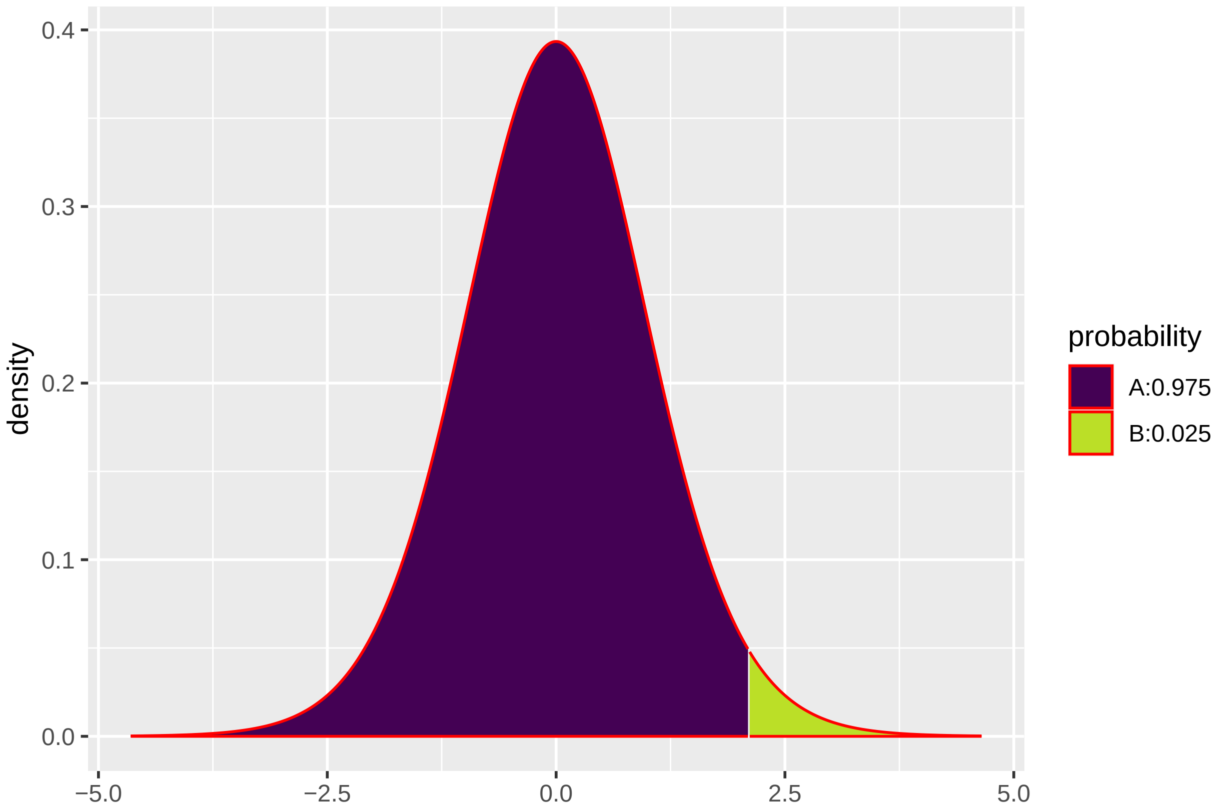
```
##  [1]            NA            NA  1.643344e-10            NA  5.947464e+01
##  [6]            NA            NA -3.715359e+01  4.063559e-22            NA
## [11]            NA  4.418391e+01            NA            NA            NA
## [16]            NA -1.285989e+01            NA -3.715359e+01            NA
```

```r
# Stored the summary of the model
summary_result <- summary(model)

# Extract the p-value for the first coefficient
p_values <- summary_result$coefficients[1, "Pr(>|t|)"]

print(p_values)
```

```
## [1] 4.063559e-22
```

```r
# Calculate the quantile for a 95% confidence level (0.975)
# with 18 degrees of freedom
mosaic::xqt(0.975, 18, col = "red")
```

```
## Registered S3 method overwritten by 'mosaic':
##   method                           from
##   fortify.SpatialPolygonsDataFrame ggplot2
```




```
## [1] 2.100922
```

b. Use an analysis-of-variance approach to test significance of regression.

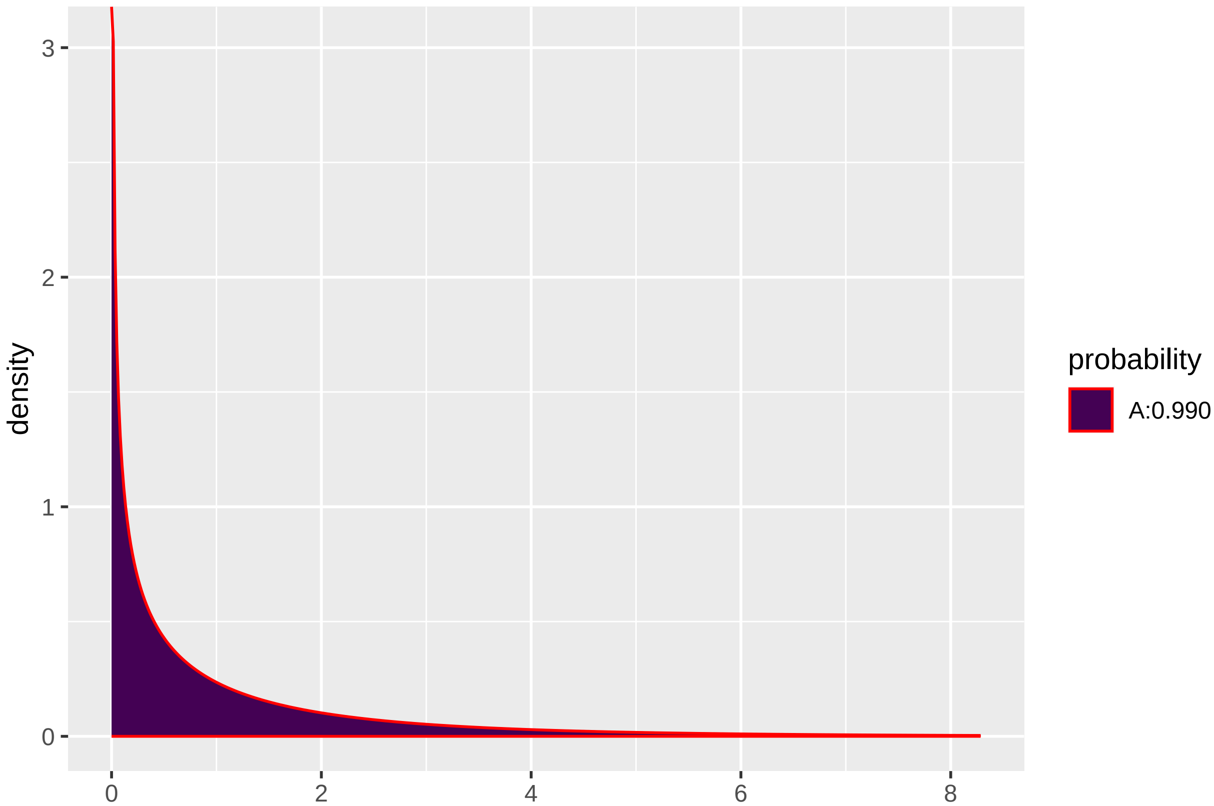
```r
# Perform ANOVA on the model
anova_result <- anova(model)

# Extract a specific p-value from the ANOVA table
p_value <- anova_result$'Pr(>F)'[1]

print(p_value)
```

```
## [1] 1.643344e-10
```

```r
# To access the F-statistic and its related values from the model summary
model_summary$fstatistic
```

```
##    value    numdf    dendf
## 165.3768   1.0000  18.0000
```

```r
# Calculate the quantile for a 99% confidence level (0.99), with numerator
# df = 1 and denominator df = 18
mosaic::xqf(0.99,1,18, col = "red")
```




```
## [1] 8.28542
```