PES University, Bangalore

Established under Karnataka Act No. 16 of 2013

UE20CS312 - Data Analytics - Worksheet 2a - Simple Linear Regression

Designed by Vibha Masti, Dept. of CSE - vibha@pesu.pes.edu

Simple Linear Regression

Simple linear regression is a statistical technique for finding the existence of an association relationship between a dependent variable and an independent variable. Simple linear regression implies that there is only one independent variable in the model. Regression is one of the most important techniques in predictive analytics since many prediction problems are modeled using regression.

Action Potentials in Dragons

Brain cells, called neurons (diagram shown below), send information throughout the brain and body. The information is sent via electro-chemical signals known as action potentials that travel down the length of the neuron. These neurons are then triggered to release chemical messengers at synapses, called neurotransmitters, which help trigger action potentials in nearby cells, and so help spread the signal all over. An action potential travels down a neuron's axon in an ion cascade. (Source: Khan Academy).

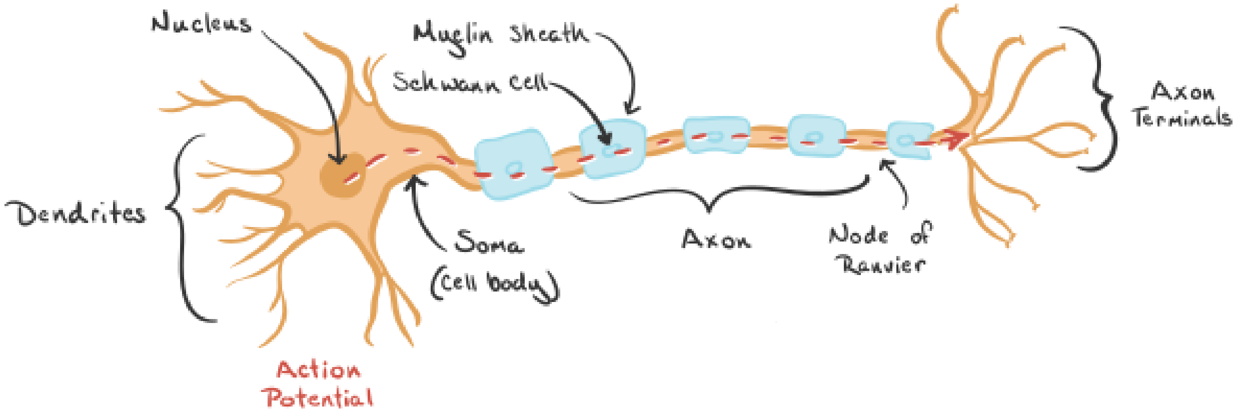


Figure 1: Diagram of a neuron - Source: Khan Academy

In the imaginary land of Westeros, the once extinct dragons were spotted again. The maesters of the capital, King's Landing, were summoned to study the nervous systems of these dragons. They were curious about how such large beings were able to move around so quickly. They studied 67 nerve bundles of two dragons and measured the **maximal conduction velocity** across fibers and the **axon diameter** of the largest fiber (Similar to the study conducted by Hursh in 1939). What they observed is stored on the GitHub repository.

Data Dictionary

```
axon_diameter: diameter of the axon in micrometers
conduction_velocity: conduction velocity of action potentials in meters per second
```

Points

The problems in this worksheet are for a total of 10 points with each problem having a different weightage.

- *Problem 1*: 1 point
- *Problem 2*: 3 points
- *Problem 3*: 3 points
- *Problem 4*: 1 point
- *Problem 5*: 2 points

Data reading

```
dragon_neurons <- read.csv('dragon_neurons.csv')
head(dragon_neurons)
```

```
##   X axon_diameter conduction_velocity
## 1 0            72            4.541130
## 2 1            66            4.275300
## 3 2            74            4.912093
## 4 3             9            2.872806
## 5 4             9            2.395194
## 6 5            65            5.120160
```

Problem 1 (1 point)

Find if a linear model is appropriate for representing the relationship between the conduction velocity (response variable) and axon diameter (explanatory variable) by finding the OLS solution. Print out the slope and the coefficient. Plot the OLS best-fit line of the model (Hint: use the `ggplot` library).

```
ols.lm <- lm(formula = conduction_velocity ~ axon_diameter, data = dragon_neurons)
ols.lm
```

```
##
## Call:
## lm(formula = conduction_velocity ~ axon_diameter, data = dragon_neurons)
##
## Coefficients:
##   (Intercept)  axon_diameter
##       2.98761        0.02475
```

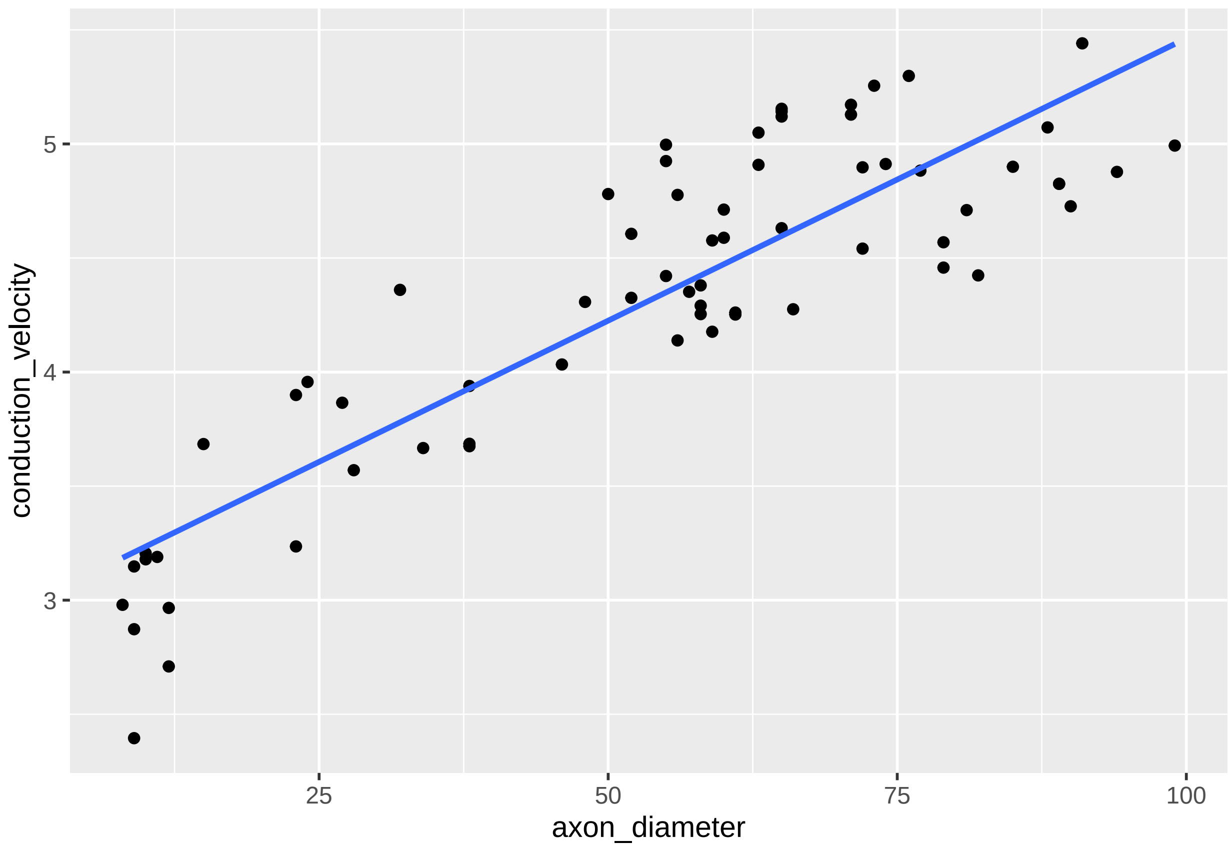
Plotting

```
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 4.1.2
```

```
ggplot(dragon_neurons, aes(x=axon_diameter, y=conduction_velocity)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE)
```

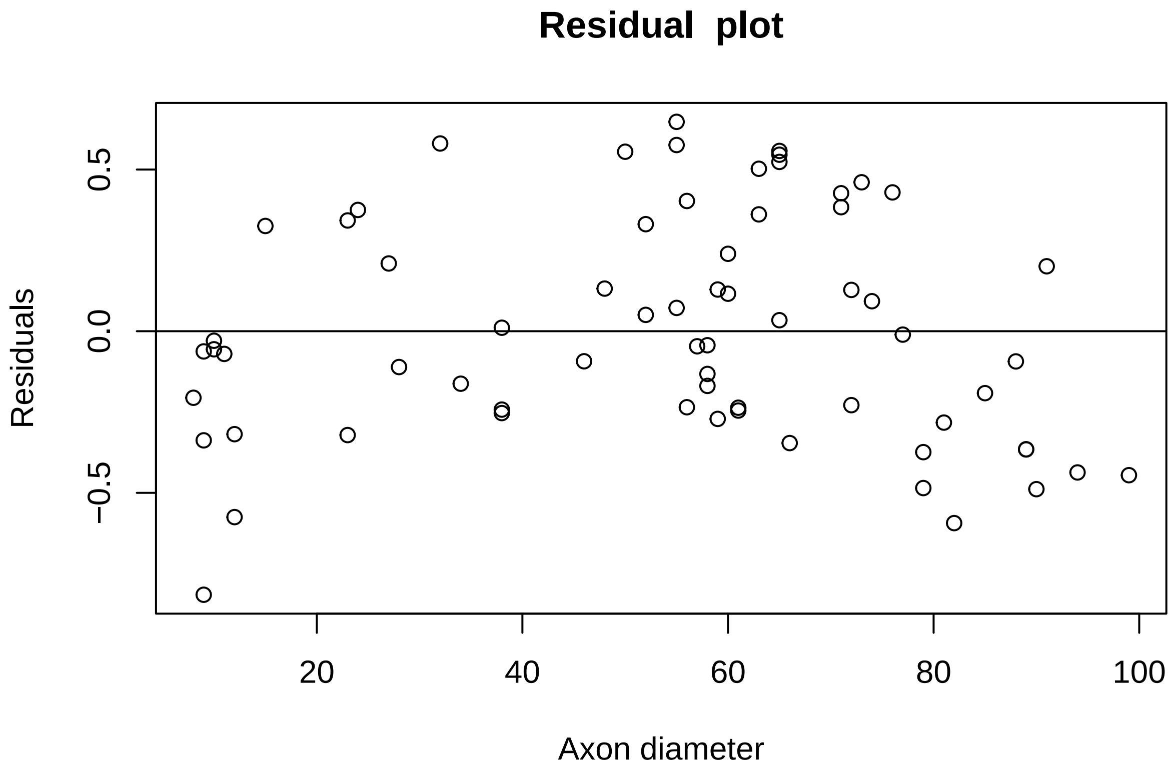
```
## `geom_smooth()` using formula 'y ~ x'
```



Problem 2 (3 points)

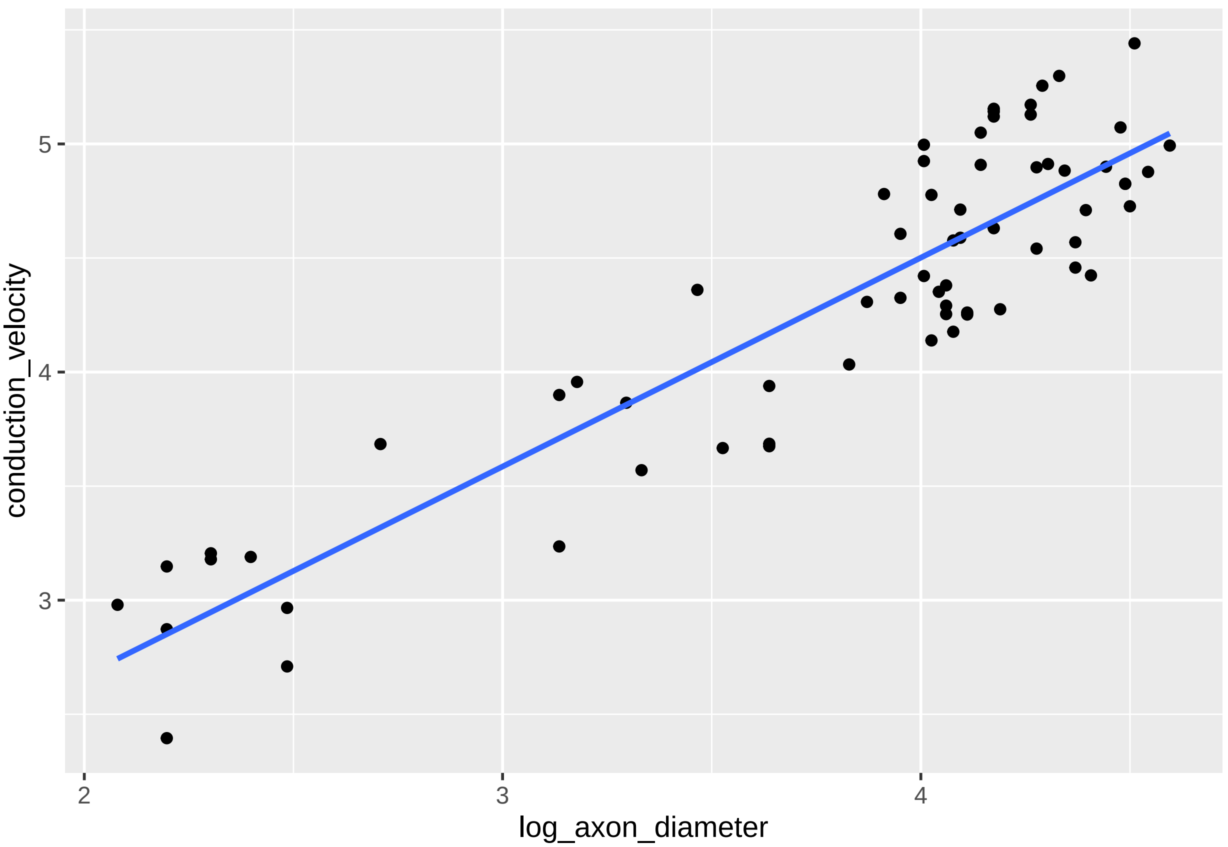
Plot the residuals of the model. Do the residuals look like white noise? If they do not, try to find a suitable functional form (hint: try transforming either x or y using natural-log or squares).

```
ols.res <- resid(ols.lm)

plot(dragon_neurons$axon_diameter, ols.res, ylab='Residuals', xlab='Axon diameter',
     main='Residual  plot')
abline(0, 0)
```

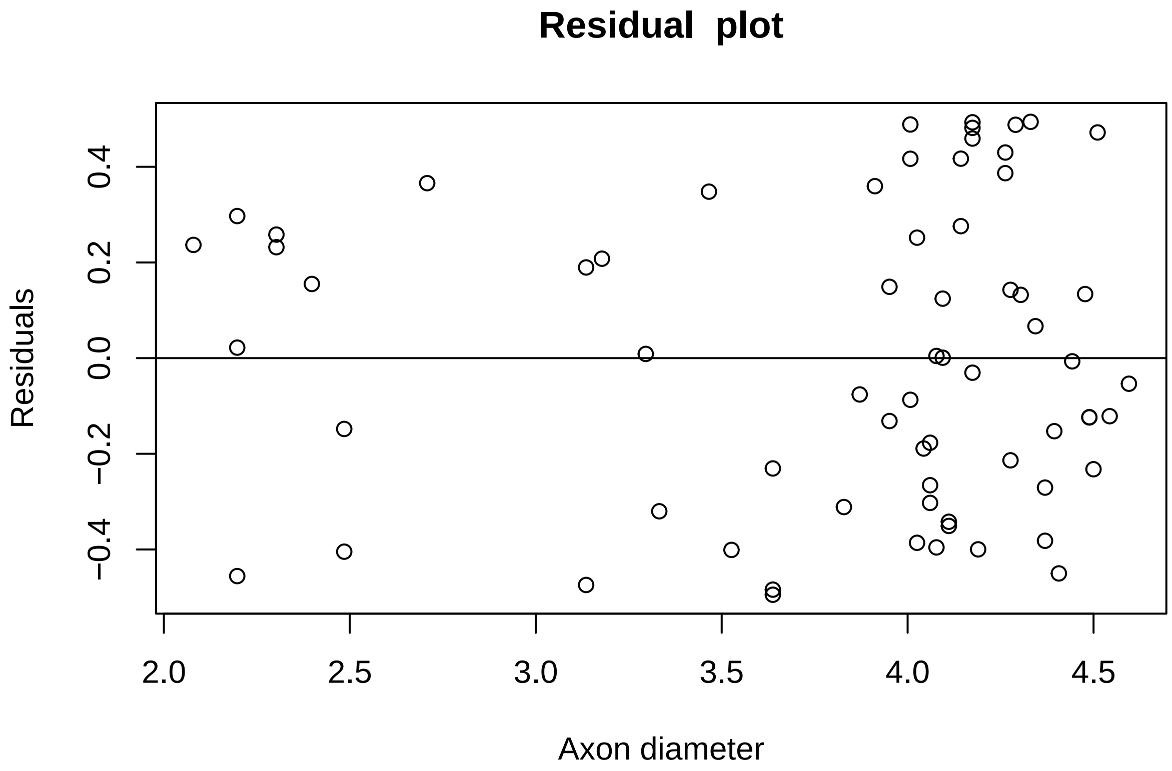


```
dragon_neurons$log_axon_diameter <- log(dragon_neurons$axon_diameter)
log_ols.lm <- lm(formula = conduction_velocity ~ log_axon_diameter, data = dragon_neurons)

ggplot(dragon_neurons, aes(x=log_axon_diameter, y=conduction_velocity)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE)
```

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



```
log_ols.res <- resid(log_ols.lm)
plot(dragon_neurons$log_axon_diameter, log_ols.res, ylab='Residuals', xlab='Axon diameter',
     main='Residual  plot')
abline(0, 0)
```



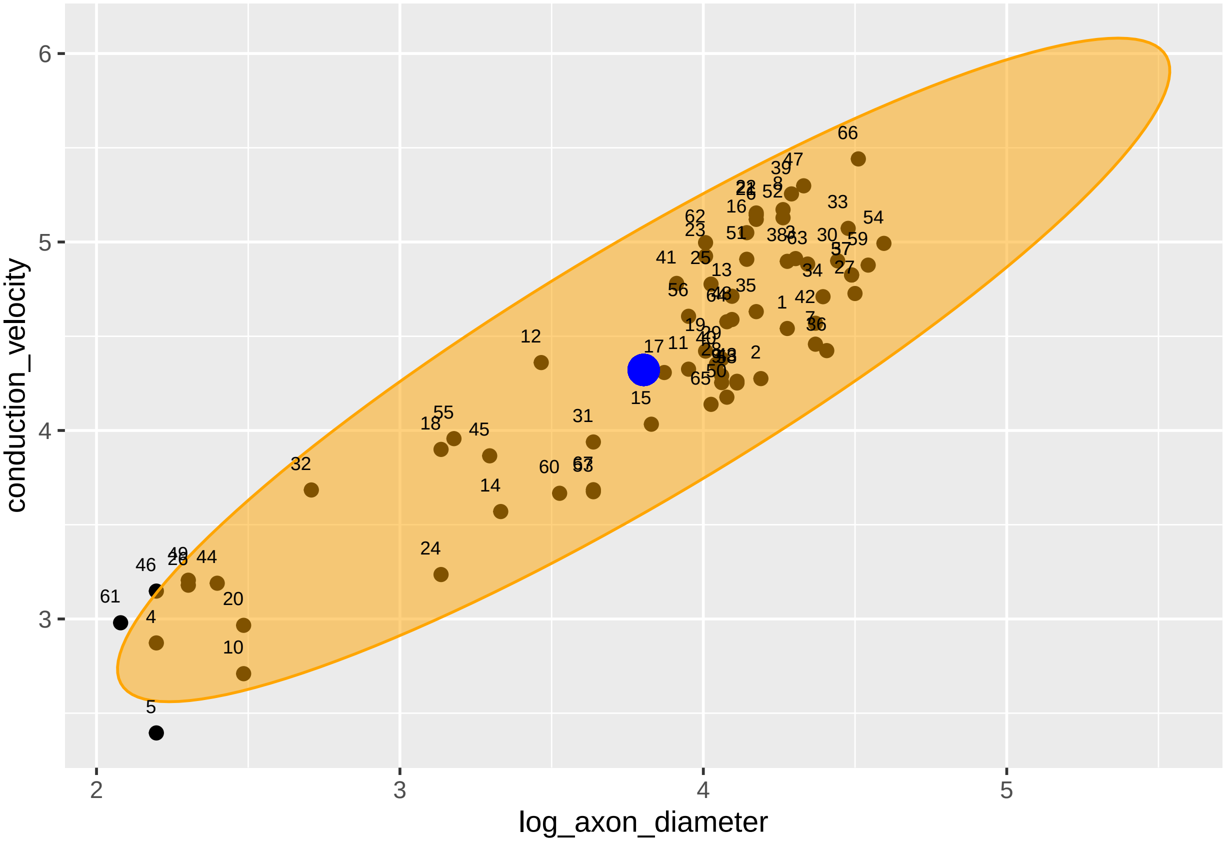
Problem 3 (3 points)

Using Mahalanobis distance as a metric, are there any potential outliers you notice? What are their Mahalanobis distances? Use the model that you decided on in the previous problem (Problem 2) as your regression model. Ensure that you plot the ellipse with a radius equal to the square root of the Chi-square value with 2 degrees of freedom and 0.95 probability.

```
log_model <- dragon_neurons[c('log_axon_diameter' ,'conduction_velocity')]

# Find the center and covariance
log_model.center <- colMeans(log_model)
log_model.cov <- cov(log_model)

# Find the radius of the ellipse
log_model.rad <- sqrt(qchisq(p=0.95, df=ncol(log_model)))

# Find the ellipse coordinates
ellipse <- car::ellipse(center=log_model.center, shape=log_model.cov, radius=log_model.rad, segments=15
```

Plot the ellipse

```
ellipse <- as.data.frame(ellipse)
colnames(ellipse) <- colnames(log_model)

# Create scatter Plot
ggplot(log_model , aes(x=log_axon_diameter, y=conduction_velocity)) +
       geom_point(size = 2) +
       geom_polygon(data=ellipse , fill="orange", color="orange", alpha=0.5) +
       geom_point(aes(log_model.center[1] , log_model.center[2]) , size=5 , color="blue") +
       geom_text(aes(label=row.names(log_model)), hjust=1, vjust=-1.5, size=2.5)
```



```
# Finding distances
distances <- mahalanobis(x=log_model, center = log_model.center, cov=log_model.cov)

# Cutoff value for ditances from Chi-Sqaure Dist.
# with p = 0.95 df = 2 which in ncol(air)
cutoff <- qchisq(p=0.95 , df=ncol(log_model))

## Display observation whose distance greater than cutoff value
log_model$distances <- distances
log_model[distances > cutoff,]
```

```
##    log_axon_diameter conduction_velocity distances
## 5           2.197225            2.395194  7.289960
## 46          2.197225            3.147883  6.052955
## 61          2.079442            2.979719  6.500782
```

Points 5, 46, and 61 are potential outliers with Mahalanobis distances 7.29, 6.05 and 6.50 respectively.

Problem 4 (1 point)

What are the R-squared values of the initial linear model and the functional form chosen in Problem 2? What do you infer from this? (hint: use the `summary` function on the created linear models)

```
summary(ols.lm)
```

```
##
## Call:
## lm(formula = conduction_velocity ~ axon_diameter, data = dragon_neurons)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.81519 -0.24935 -0.04665  0.32827  0.64757
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)   2.987611   0.101069   29.56   <2e-16 ***
## axon_diameter 0.024753   0.001699   14.57   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3509 on 65 degrees of freedom
## Multiple R-squared:  0.7656, Adjusted R-squared:  0.762
## F-statistic: 212.3 on 1 and 65 DF,  p-value: < 2.2e-16
```

```
summary(log_ols.lm)
```

```
##
## Call:
## lm(formula = conduction_velocity ~ log_axon_diameter, data = dragon_neurons)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.49467 -0.26822 -0.00671  0.25506  0.49396
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)        0.83911    0.21037   3.989 0.000171 ***
## log_axon_diameter  0.91559    0.05439  16.833  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3131 on 65 degrees of freedom
## Multiple R-squared:  0.8134, Adjusted R-squared:  0.8105
## F-statistic: 283.3 on 1 and 65 DF,  p-value: < 2.2e-16
```

The R-squared value of the log-linear model is higher, indicating that the model is a better fit.

Problem 5 (2 points)

Using the same `summary` function as Problem 4, determine if there is a statistically significant linear relationship at a significance value of 0.05 of the **overall model** chosen in Problem 2. What do you understand about the relationship between dragons' axon diameters and conduction velocity? (Hint: understand the values displayed in `summary` and search for the right data).

```
summary(log_ols.lm)
```

```
##
## Call:
## lm(formula = conduction_velocity ~ log_axon_diameter, data = dragon_neurons)
```

```
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.49467 -0.26822 -0.00671  0.25506  0.49396
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)        0.83911    0.21037   3.989 0.000171 ***
## log_axon_diameter  0.91559    0.05439  16.833  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3131 on 65 degrees of freedom
## Multiple R-squared:  0.8134, Adjusted R-squared:  0.8105
## F-statistic: 283.3 on 1 and 65 DF,  p-value: < 2.2e-16
```

Since the p-value of the F-statistic is almost zero, the linear model is statistically significant.