

```

library(openxlsx)
library(ggplot2)

excel_file <- "~/Data-Science-G7/DATA.xlsx"

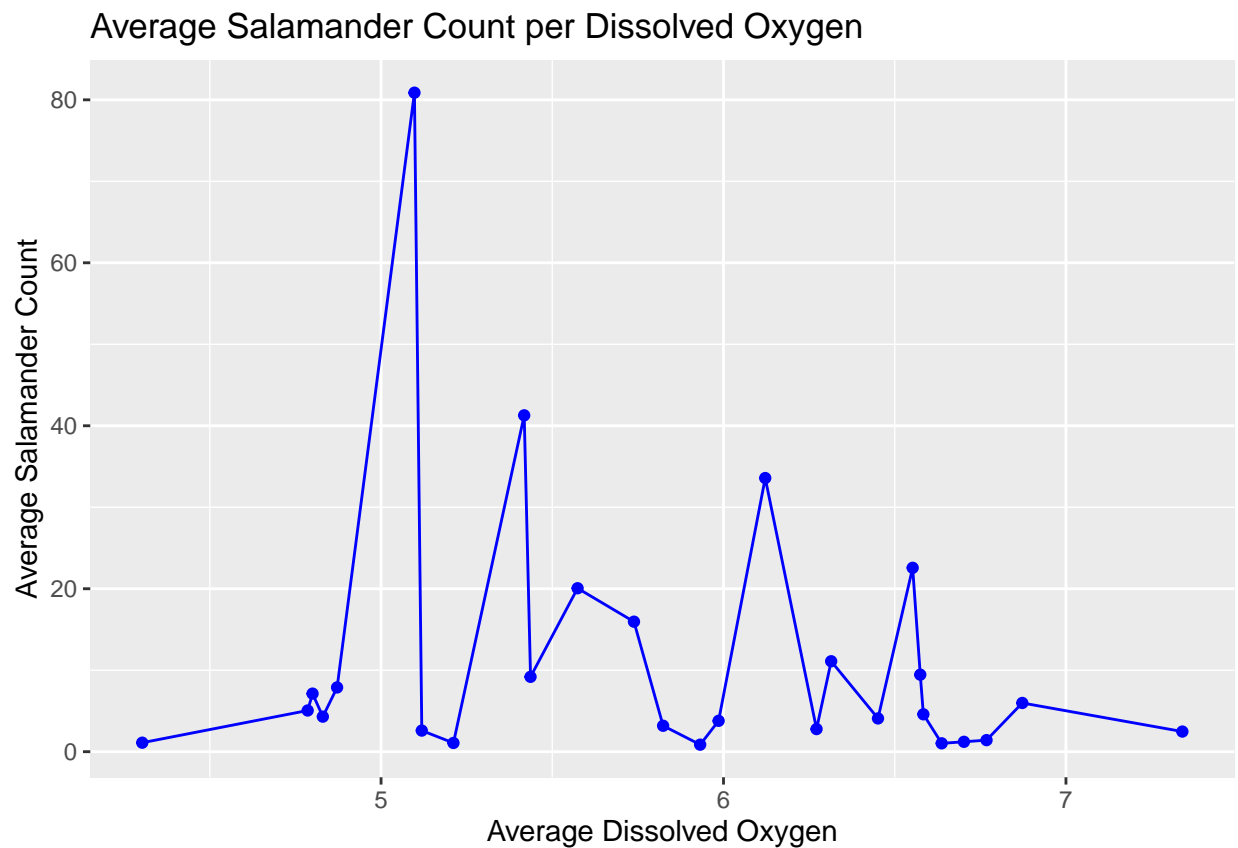
#install.packages("readxl")
library(readxl)

sheet_names = excel_sheets(excel_file)
#sheet_names

dissolvedOxyvsSal <- read_excel(excel_file, sheet = sheet_names[2])

plot <- ggplot(dissolvedOxyvsSal, aes(x = average_dissolved_oxygen, y = average_salamander)) +
  geom_line(color = "blue") +
  geom_point(color = "blue") +
  labs(x = "Average Dissolved Oxygen", y = "Average Salamander Count",
       title = "Average Salamander Count per Dissolved Oxygen")
plot

```



```

ggsave(filename = "~/Data-Science-G7/AvgSalPOxy.png", plot = plot)

```

```
## Saving 6.5 x 4.5 in image
```

```
#put units
```

```

# Fit a linear regression model
linear_model <- lm(average_salamander ~ average_dissolved_oxygen, data = dissolvedOxyvsSal)

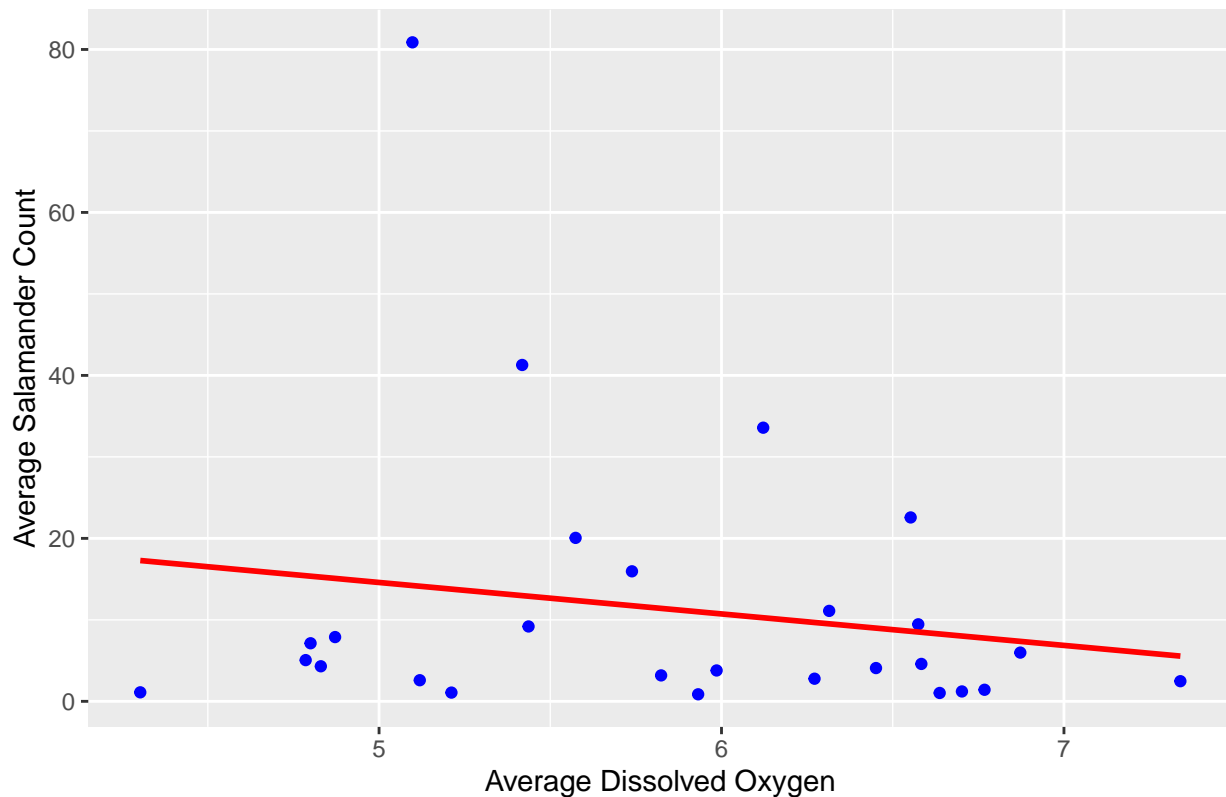
# Display the summary of the linear model
summary(linear_model)

##
## Call:
## lm(formula = average_salamander ~ average_dissolved_oxygen, data = dissolvedOxyvsSal)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -16.176  -8.228  -6.339   1.267  66.663
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)      33.896     25.127   1.349   0.189
## average_dissolved_oxygen  -3.862      4.253  -0.908   0.373
##
## Residual standard error: 17.25 on 25 degrees of freedom
## Multiple R-squared:  0.03192,    Adjusted R-squared:  -0.006803
## F-statistic: 0.8243 on 1 and 25 DF,  p-value: 0.3726

# Visualize the linear regression line along with the data points
ggplot(dissolvedOxyvsSal, aes(x = average_dissolved_oxygen, y = average_salamander)) +
  geom_point(color = "blue") +
  geom_smooth(method = "lm", se = FALSE, color = "red", formula = y ~ x) +
  labs(x = "Average Dissolved Oxygen", y = "Average Salamander Count",
       title = "Average Salamander Count per Dissolved Oxygen")

```

Average Salamander Count per Dissolved Oxygen



```
# Fit a quadratic polynomial regression model
quadratic_model <- lm(average_salamander ~ poly(average_dissolved_oxygen, 2), data = dissolvedOxyvsSal)

# Display the summary of the quadratic model
summary(quadratic_model)
```

```
##
## Call:
## lm(formula = average_salamander ~ poly(average_dissolved_oxygen,
##    2), data = dissolvedOxyvsSal)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -14.499  -8.107  -5.438   1.565  65.863
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)      11.282      3.286   3.434  0.00217 **
## poly(average_dissolved_oxygen, 2)1  -15.660      17.073  -0.917  0.36814
## poly(average_dissolved_oxygen, 2)2  -21.021      17.073  -1.231  0.23014
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 17.07 on 24 degrees of freedom
## Multiple R-squared:  0.08944,    Adjusted R-squared:  0.01356
## F-statistic: 1.179 on 2 and 24 DF,  p-value: 0.3249
```

```

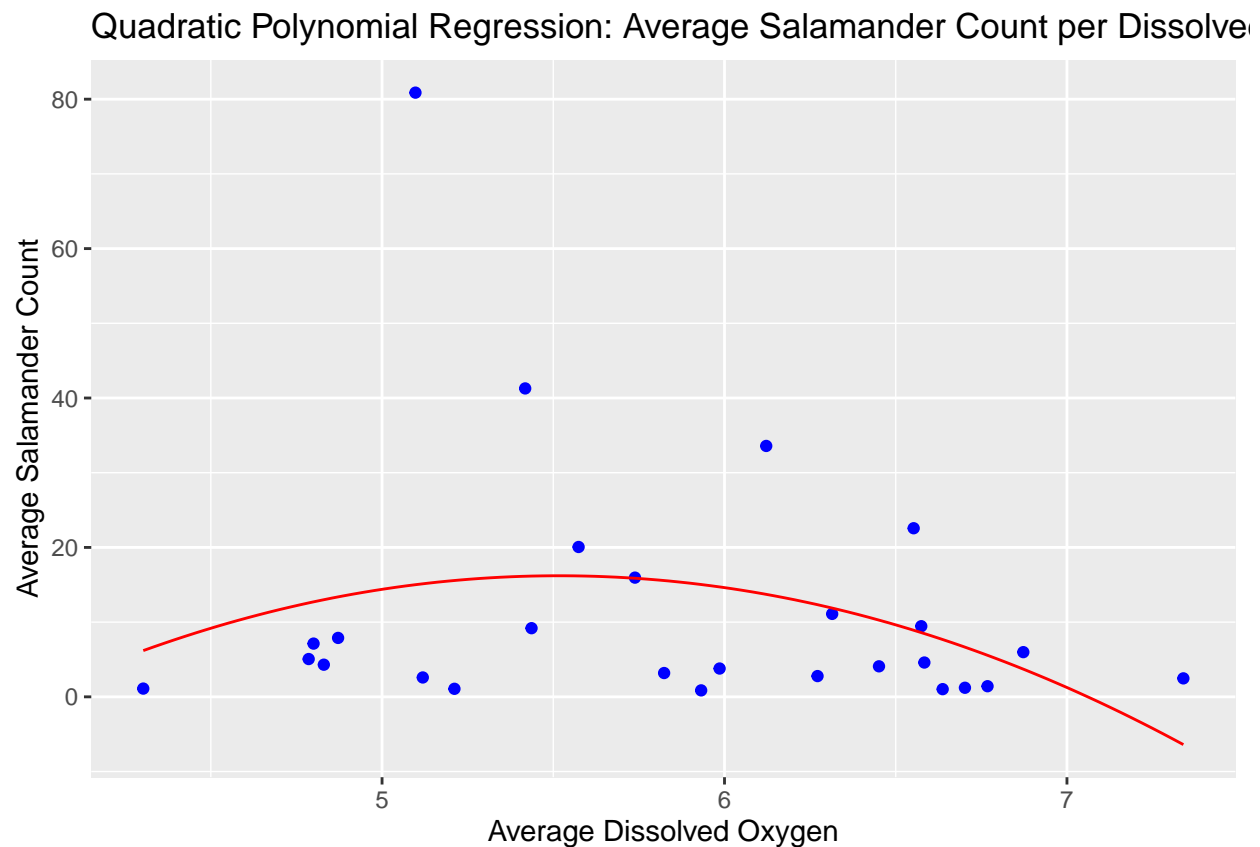
# Generate a sequence of values for plotting the regression curve
year_seq <- seq(min(dissolvedOxyvsSal$average_dissolved_oxygen), max(dissolvedOxyvsSal$average_dissolved_oxygen))

# Predict the average dissolved oxygen using the quadratic model
predicted_values <- predict(quadratic_model, newdata = data.frame(average_dissolved_oxygen = year_seq))

# Create a data frame for plotting
plot_data <- data.frame(average_dissolved_oxygen = year_seq, predicted_values = predicted_values)

# Visualize the quadratic regression curve along with the data points
ggplot() +
  geom_point(data = dissolvedOxyvsSal, aes(x = average_dissolved_oxygen, y = average_salamander), color = "blue") +
  geom_line(data = plot_data, aes(x = average_dissolved_oxygen, y = predicted_values), color = "red") +
  labs(x = "Average Dissolved Oxygen", y = "Average Salamander Count",
       title = "Quadratic Polynomial Regression: Average Salamander Count per Dissolved Oxygen")

```



```

# exponential regression model
exp_model <- lm(log(average_salamander) ~ log(average_dissolved_oxygen), data = dissolvedOxyvsSal)

# summary of the exponential model
summary(exp_model)

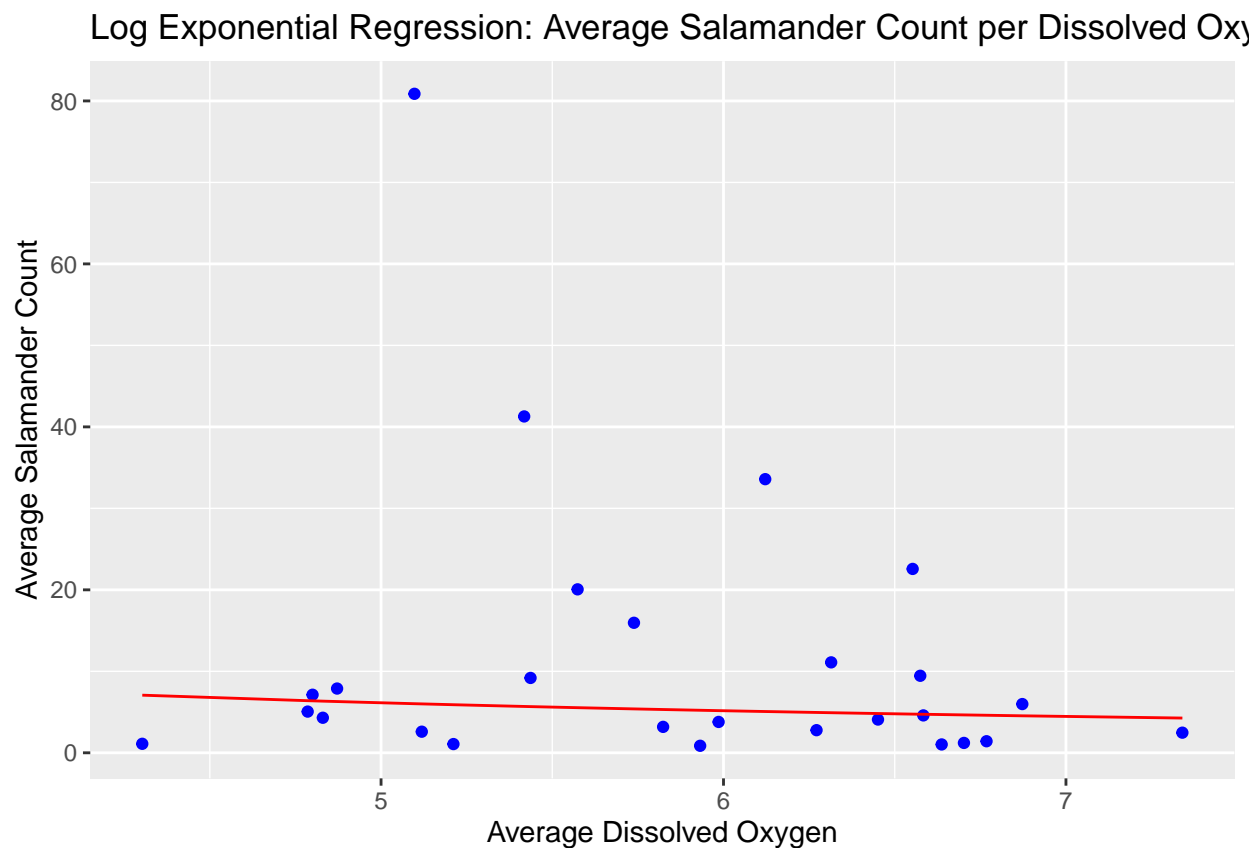
##
## Call:
## lm(formula = log(average_salamander) ~ log(average_dissolved_oxygen),
##     data = dissolvedOxyvsSal)

```

```
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.8573 -0.7086 -0.1658  0.7524  2.5968
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)      3.3381     3.0673   1.088   0.287
## log(average_dissolved_oxygen) -0.9467     1.7392  -0.544   0.591
##
## Residual standard error: 1.233 on 25 degrees of freedom
## Multiple R-squared:  0.01171,    Adjusted R-squared:  -0.02782
## F-statistic: 0.2963 on 1 and 25 DF,  p-value: 0.591

# Generate predicted values using the exponential model
predicted_values <- exp(predict(exp_model, newdata = dissolvedOxyvsSal))

# exponential regression curve along with the data points
ggplot(dissolvedOxyvsSal, aes(x = average_dissolved_oxygen, y = average_salamander)) +
  geom_point(color = "blue") +
  geom_line(aes(x = average_dissolved_oxygen, y = predicted_values), color = "red") +
  labs(x = "Average Dissolved Oxygen", y = "Average Salamander Count",
       title = "Log Exponential Regression: Average Salamander Count per Dissolved Oxygen")
```

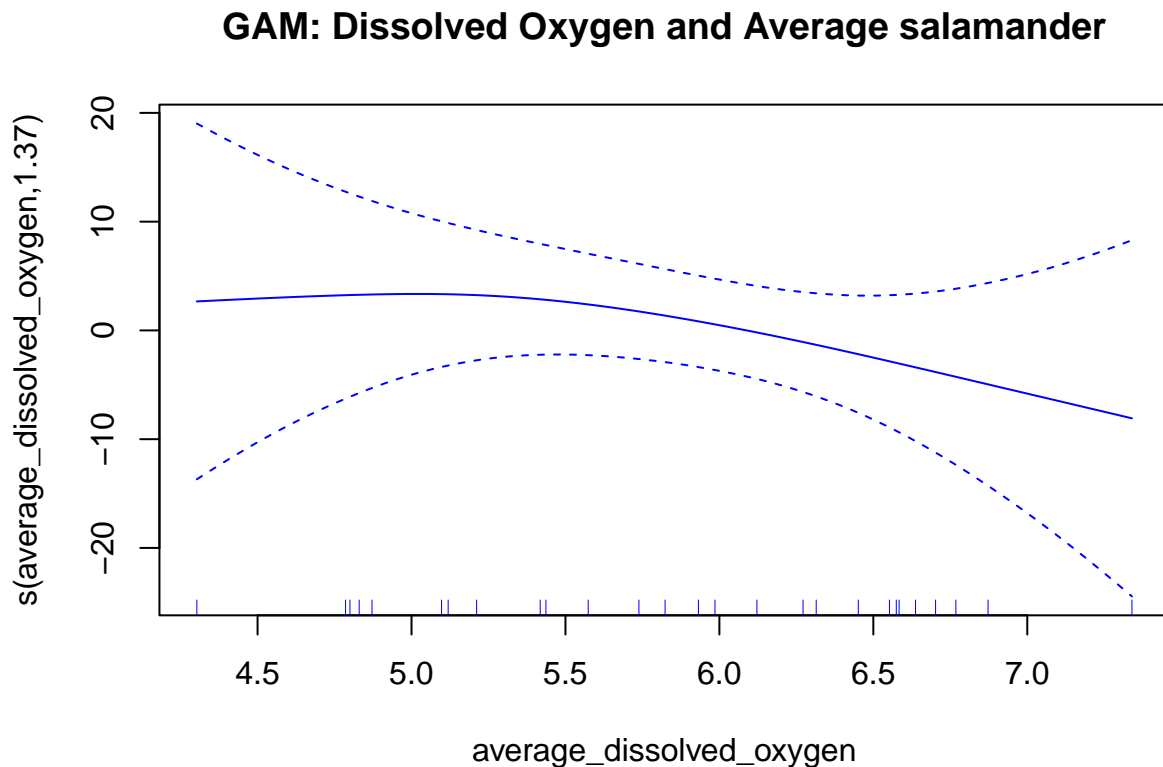


```
library(mgcv)
```

```
## Loading required package: nlme
```

```
## This is mgcv 1.9-0. For overview type 'help("mgcv-package")'.
gam_model <- gam(average_salamander ~ s(average_dissolved_oxygen), data = dissolvedOxyvsSal)
summary(gam_model)

##
## Family: gaussian
## Link function: identity
##
## Formula:
## average_salamander ~ s(average_dissolved_oxygen)
##
## Parametric coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  11.282    3.287    3.432 0.00212 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##             edf Ref.df    F p-value
## s(average_dissolved_oxygen) 1.37  1.661 0.899  0.515
##
## R-sq.(adj) =  0.0127   Deviance explained = 6.48%
## GCV = 319.79   Scale est. = 291.72    n = 27
plot(gam_model, se = TRUE, col = "blue", main = "GAM: Dissolved Oxygen and Average salamander")
```



```
library(ggplot2)

# data frame with predicted values from the GAM model
predicted_data <- data.frame(average_dissolved_oxygen = dissolvedOxyvsSal$average_dissolved_oxygen,
                             predicted_values = predict(gam_model, newdata = dissolvedOxyvsSal))
```