UFC RESEARCH QUESTION 1

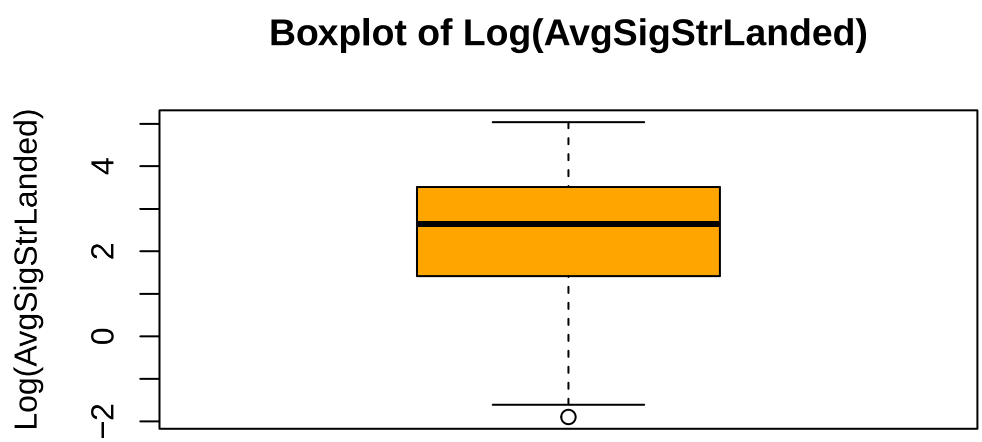
Due 8th December, 2024

```r
ufc <- read.csv("ufc-master.csv")

# View the data
head(ufc)
```

```r
filtered_ufc_blue = ufc[!is.na(ufc$BlueAvgSigStrLanded),
                        c("BlueReachCms","BlueAvgSigStrLanded","WeightClass")]
colnames(filtered_ufc_blue) <- c("ReachCms", "AvgSigStrLanded", "WeightClass")
filtered_ufc_red = ufc[!is.na(ufc$RedAvgSigStrLanded),
                       c("RedReachCms","RedAvgSigStrLanded","WeightClass")]
colnames(filtered_ufc_red) <- c("ReachCms", "AvgSigStrLanded", "WeightClass")

# appending the two data sets
filtered_ufc = rbind(filtered_ufc_blue, filtered_ufc_red)

# exclude outlier(one observation with 0 cm reach)
filtered_ufc = filtered_ufc[filtered_ufc$ReachCms > 0,]
filtered_ufc <- filtered_ufc[filtered_ufc$AvgSigStrLanded > 0, ]
head(filtered_ufc)
```

```
  ReachCms AvgSigStrLanded       WeightClass
1   172.72            2.72          Flyweight
2   165.10            3.71 Women's Flyweight
3   205.74            3.16 Light Heavyweight
4   172.72            3.70 Women's Flyweight
5   190.50            3.47      Middleweight
6   187.96            3.17       Welterweight
```

```r
# Log-transform the variables
filtered_ufc$LogAvgSigStrLanded <- log(filtered_ufc$AvgSigStrLanded)
filtered_ufc$LogReachCms <- log(filtered_ufc$ReachCms)

# Boxplot for log-transformed AvgSigStrLanded
boxplot(filtered_ufc$LogAvgSigStrLanded,
        main = "Boxplot of Log(AvgSigStrLanded)",
        ylab = "Log(AvgSigStrLanded)",
        col = "orange")
```




```r
model_q1 <- lm(LogAvgSigStrLanded ~ LogReachCms * WeightClass, data = filtered_ufc)
summary(model_q1)
```

```
Call:
lm(formula = LogAvgSigStrLanded ~ LogReachCms * WeightClass,
    data = filtered_ufc)

Residuals:
    Min      1Q  Median      3Q     Max
-4.2808 -1.0156  0.1242  0.9675  3.0182
```

```
Coefficients:
                                               Estimate Std. Error t value
(Intercept)                                   17.481380   4.818352   3.628
LogReachCms                                   -2.918097   0.933286  -3.127
WeightClassCatch Weight                        0.139533  11.682777   0.012
WeightClassFeatherweight                      15.564256   6.992545   2.226
WeightClassFlyweight                          56.353430   8.348930   6.750
WeightClassHeavyweight                        -9.181965   7.294519  -1.259
WeightClassLight Heavyweight                 -10.248035   7.651211  -1.339
WeightClassLightweight                         2.977911   6.476798   0.460
WeightClassMiddleweight                       -2.414576   6.975805  -0.346
WeightClassWelterweight                       -0.622770   6.480375  -0.096
WeightClassWomen's Bantamweight               -0.175266  10.935239  -0.016
WeightClassWomen's Featherweight              10.480862  29.301762   0.358
WeightClassWomen's Flyweight                 -23.436916   9.329845  -2.512
WeightClassWomen's Strawweight               -37.875335   8.431193  -4.492
LogReachCms:WeightClassCatch Weight           -0.114204   2.250887  -0.051
LogReachCms:WeightClassFeatherweight          -2.969015   1.350596  -2.198
LogReachCms:WeightClassFlyweight             -10.998515   1.622531  -6.779
LogReachCms:WeightClassHeavyweight             1.801642   1.394803   1.292
LogReachCms:WeightClassLight Heavyweight       2.024423   1.464198   1.383
LogReachCms:WeightClassLightweight            -0.503961   1.250192  -0.403
LogReachCms:WeightClassMiddleweight            0.514160   1.339423   0.384
LogReachCms:WeightClassWelterweight            0.205765   1.247897   0.165
LogReachCms:WeightClassWomen's Bantamweight    0.008631   2.126171   0.004
LogReachCms:WeightClassWomen's Featherweight  -2.149513   5.676000  -0.379
LogReachCms:WeightClassWomen's Flyweight       4.422802   1.816459   2.435
LogReachCms:WeightClassWomen's Strawweight     7.384097   1.649178   4.477
                                             Pr(>|t|)
(Intercept)                                  0.000287 ***
LogReachCms                                  0.001772 **
WeightClassCatch Weight                      0.990471
WeightClassFeatherweight                     0.026045 *
WeightClassFlyweight                         1.55e-11 ***
WeightClassHeavyweight                       0.208147
WeightClassLight Heavyweight                 0.180467
WeightClassLightweight                       0.645682
WeightClassMiddleweight                      0.729247
WeightClassWelterweight                      0.923442
WeightClassWomen's Bantamweight              0.987213
WeightClassWomen's Featherweight             0.720584
WeightClassWomen's Flyweight                 0.012017 *
WeightClassWomen's Strawweight               7.12e-06 ***
```

```
LogReachCms:WeightClassCatch Weight          0.959536
LogReachCms:WeightClassFeatherweight         0.027948 *
LogReachCms:WeightClassFlyweight             1.27e-11 ***
LogReachCms:WeightClassHeavyweight           0.196493
LogReachCms:WeightClassLight Heavyweight     0.166810
LogReachCms:WeightClassLightweight           0.686877
LogReachCms:WeightClassMiddleweight          0.701084
LogReachCms:WeightClassWelterweight          0.869034
LogReachCms:WeightClassWomen's Bantamweight  0.996761
LogReachCms:WeightClassWomen's Featherweight 0.704916
LogReachCms:WeightClassWomen's Flyweight     0.014913 *
LogReachCms:WeightClassWomen's Strawweight   7.63e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.117 on 11431 degrees of freedom
Multiple R-squared:  0.04748,	Adjusted R-squared:  0.0454 
F-statistic: 22.79 on 25 and 11431 DF,  p-value: < 2.2e-16
```

```r
# Load necessary libraries
library(car)          # For VIF
```

```
Loading required package: carData
```

```r
library(ggplot2)      # For residual plots

# 1. Check Variance Inflation Factor (VIF) for collinearity
vif_values <- vif(model_q1)
```

```
there are higher-order terms (interactions) in this model
consider setting type = 'predictor'; see ?vif
```

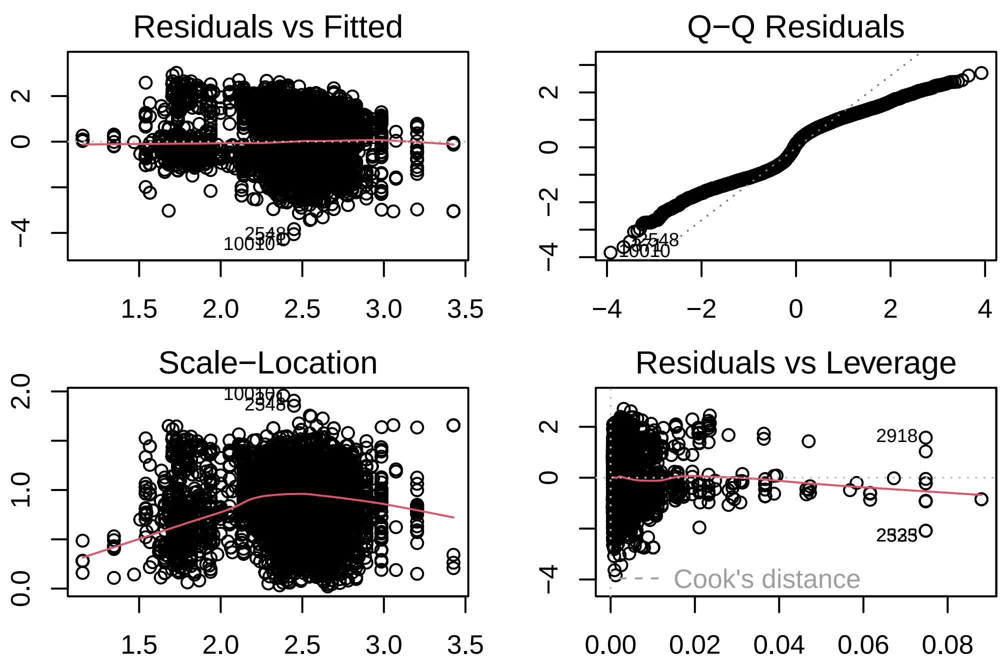
```r
print("Variance Inflation Factor (VIF):")
```

```
[1] "Variance Inflation Factor (VIF):"
```

```r
print(vif_values)
```

```
                              GVIF Df GVIF^(1/(2*Df))
LogReachCms             2.982482e+01  1        5.46121
WeightClass             2.614566e+52 12      152.77712
LogReachCms:WeightClass 2.656355e+52 12      152.87809
```

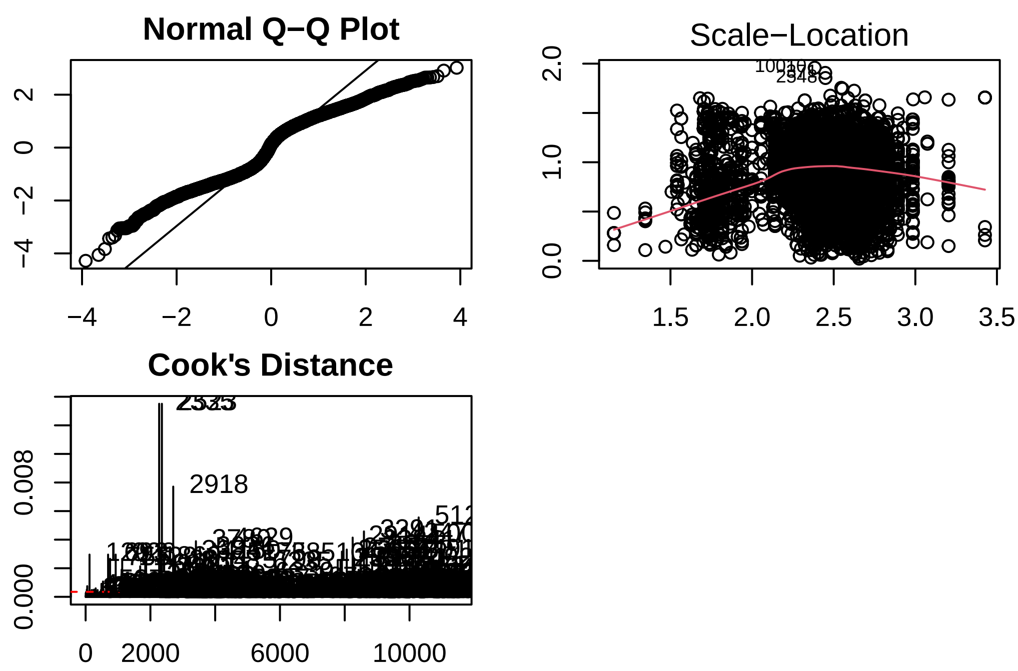
```
# 2. Residuals vs Fitted Plot for Linearity
par(mfrow = c(2, 2), mar = c(2, 2, 2, 2))  # Set plotting layout
plot(model_q1)
```




```
# 3. Normal Q-Q Plot for Normality of Residuals
qqnorm(residuals(model_q1))
qqline(residuals(model_q1))

# 4. Scale-Location Plot for Homoscedasticity
plot(model_q1, which = 3)

# 5. Check for influential points using Cook's Distance
cooksd <- cooks.distance(model_q1)
plot(cooksd, type = "h", main = "Cook's Distance", ylab = "Cook's Distance")

# Highlight observations with Cook's Distance > threshold
threshold <- 4 / nrow(filtered_ufc)
```

```
influential <- which(cooksd > threshold)
abline(h = threshold, col = "red", lty = 2)
text(x = influential, y = cooksd[influential], labels = names(cooksd[influential]), pos = 4)

# 6. R-squared value
r_squared <- summary(model_q1)$r.squared
cat("R-squared:", r_squared, "\n")
```

```
R-squared: 0.04748068
```




```
# Load necessary library
library(knitr)

# Create a summary of the model
model_summary <- summary(model_q1)

# Extract coefficients and format into a data frame
coef_table <- as.data.frame(model_summary$coefficients)
colnames(coef_table) <- c("Estimate", "Std. Error", "t value", "Pr(>|t|)")

# Round to 3 decimal places
```

```r
coef_table <- round(coef_table, 3)

# Create a kable table
kable(coef_table, caption = "Regression Coefficients for model_log", format = "markdown")
```

Table 1: Regression Coefficients for model_log

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	17.481	4.818	3.628	0.000
LogReachCms	-2.918	0.933	-3.127	0.002
WeightClassCatch Weight	0.140	11.683	0.012	0.990
WeightClassFeatherweight	15.564	6.993	2.226	0.026
WeightClassFlyweight	56.353	8.349	6.750	0.000
WeightClassHeavyweight	-9.182	7.295	-1.259	0.208
WeightClassLight Heavyweight	-10.248	7.651	-1.339	0.180
WeightClassLightweight	2.978	6.477	0.460	0.646
WeightClassMiddleweight	-2.415	6.976	-0.346	0.729
WeightClassWelterweight	-0.623	6.480	-0.096	0.923
WeightClassWomen's Bantamweight	-0.175	10.935	-0.016	0.987
WeightClassWomen's Featherweight	10.481	29.302	0.358	0.721
WeightClassWomen's Flyweight	-23.437	9.330	-2.512	0.012
WeightClassWomen's Strawweight	-37.875	8.431	-4.492	0.000
LogReachCms:WeightClassCatch Weight	-0.114	2.251	-0.051	0.960
LogReachCms:WeightClassFeatherweight	-2.969	1.351	-2.198	0.028
LogReachCms:WeightClassFlyweight	-10.999	1.623	-6.779	0.000
LogReachCms:WeightClassHeavyweight	1.802	1.395	1.292	0.196
LogReachCms:WeightClassLight Heavyweight	2.024	1.464	1.383	0.167
LogReachCms:WeightClassLightweight	-0.504	1.250	-0.403	0.687
LogReachCms:WeightClassMiddleweight	0.514	1.339	0.384	0.701
LogReachCms:WeightClassWelterweight	0.206	1.248	0.165	0.869
LogReachCms:WeightClassWomen's Bantamweight	0.009	2.126	0.004	0.997
LogReachCms:WeightClassWomen's Featherweight	-2.150	5.676	-0.379	0.705
LogReachCms:WeightClassWomen's Flyweight	4.423	1.816	2.435	0.015
LogReachCms:WeightClassWomen's Strawweight	7.384	1.649	4.477	0.000

Results

Research Question 1: Fighter reach vs Total Strikes Landed

A multiple linear regression (MLR) model was used, with the log-transformed average significant strikes landed as the response variable and an interaction term between log-transformed reach and Weight Class as predictors. Log transformation of the response variable was applied to address linearity, and diagnostic plots were used to evaluate model assumptions. Alternative approaches, including Weighted Least Squares (WLS) and Generalized Linear Models (GLM) were explored but showed similar performance to the log-transformed MLR. The MLR model was ultimately selected for its simplicity.

The model revealed some significant relationships. Notably, a negative interaction between reach and certain weight classes, such as Flyweight and Featherweight, was observed, indicating that the relationship between reach and strikes landed varied across weight divisions. Significant main effects were also identified for some weight classes, such as Flyweight and Women's Strawweight. However, the overall model performance was poor, with an adjusted R-squared value of 0.045, suggesting limited explanatory power. The limited explanatory power may indicate that factors beyond reach and weight class, such as fighting style, experience, or strategy, play a more substantial role in determining the number of strikes landed.

Diagnostic evaluations highlighted several issues. Scale-location plots indicated non-constant variance, which was not fully resolved even with WLS. Normal Q-Q plots showed some departures from normality, particularly in the tails. Additionally, high Variance Inflation Factor (VIF) values for interaction terms indicated multicollinearity concerns, further complicating the model's interpretability.

In conclusion, while the analysis identified a modest and significant relationship between reach and strikes anded that varies by weight class, the low R-squared value suggests that other unexamined factors are likely more influential. Despite adjustments, violations of key model assumptions limit the reliability of these findings. Future research should explore additional predictors, nonlinear methods, or more advanced modeling approaches to better capture the complexity of factors affecting fight dynamics.

Conclusion (for q1)

This study examined how a fighter's reach relates to the total number of strikes landed, considering weight classes. Using a multiple linear regression (MLR) model with log-transformed variables, we found significant interactions between reach and weight classes like Flyweight and Featherweight, showing that reach impacts different divisions differently. Significant effects were also observed for weight classes such as Flyweight and Women's Strawweight.

Our approach included evaluating model assumptions and exploring alternatives like weighted least squares and generalized linear models. Future work could include additional factors like skill level or fight strategy and explore advanced modeling techniques to improve insights into combat sports performance.