UFC Statistics

Team #5

7/24/2020

Link to the data set: https://www.kaggle.com/mdabbert/ultimate-ufc-dataset?select=ufc-master.csv

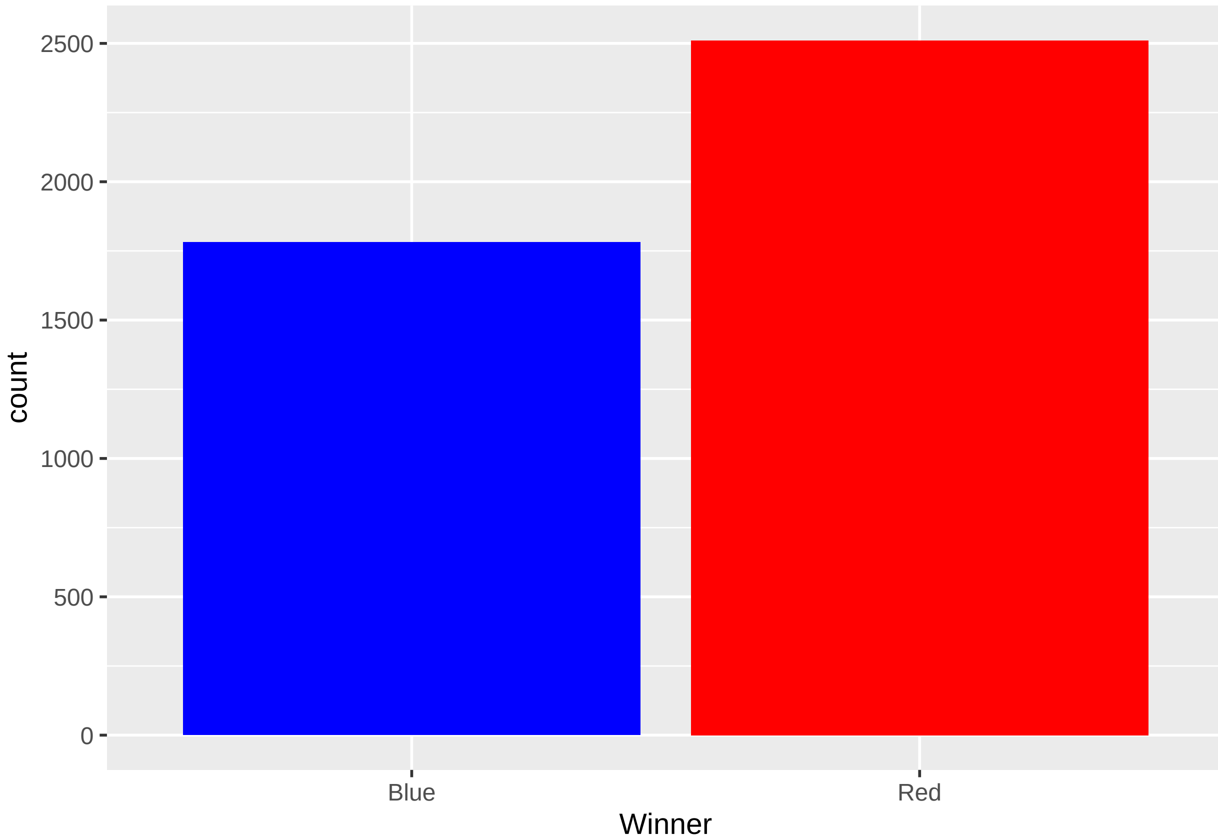
```
df <- read.csv('ufc-master.csv')

# head(df)
# names(df)
# summary(df)
```

Questions:

- Inquiry 1: What is the appropriate analysis to confirm that the color advantage is signficant?
- Inquiry 2: Check the understanding: The effect of reach is statistically significant, but it is a small effect. Also, what is the interpretation of the coefficient for AbsDiff?
- Inquiry 3: Check the understanding: Height appears to have no real advantage at all.
- Inquiry 4: Check the understanding: There is no effect to combining height and reach together.
- Inquiry 5: The understanding of the model starts to fall apart after running the saturated model. Removing the three-way interaction term has a p value of 1, which means that this term can be removed? And then the same for the two-way interactions
- Inquiry 6: Questions similar to Inquiry 2. It looks like age difference is statistically significant, but it's a small effect?

Inquiry 1: Does one color have an advantage over the other? Does gender make a difference?

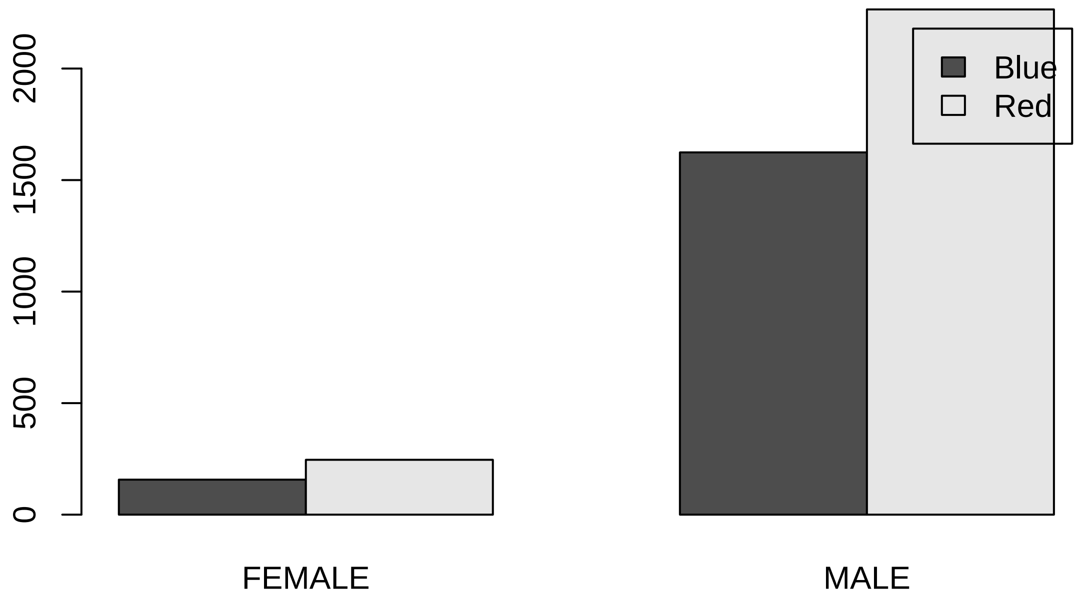
```
# Get the data
color <- df[,c('Winner', 'gender')]

# Plot the graph without gender
colorOnly.bar <- ggplot(color, aes(x = Winner))
colorOnly.bar +
  geom_bar(fill = c('Blue', 'Red'))
```



Based on the graph, it appears that red has a large advantage over blue. (Question: How are colors chosen? At random? Based on rank?)

We can break this down and look at the gender splits.

```
# Plot the graph with gender
colorGender.table <- with(color, table(Winner, gender))
colorGender.bar <- barplot(colorGender.table, beside = TRUE, legend = TRUE)
```




```r
# Calculate win percentages
print(paste('Overall Red Wins: ',
            nrow(subset(color, Winner == 'Red'))))
```

```
## [1] "Overall Red Wins:  2511"
```

```r
print(paste('Overall Blue Wins: ',
            nrow(subset(color, Winner == 'Blue'))))
```

```
## [1] "Overall Blue Wins:  1781"
```

```r
print(paste('Overall Red Win Pct: ',
            round( 100 * nrow(subset(color, Winner == 'Red')) / nrow(color), 2 )))
```

```
## [1] "Overall Red Win Pct:  58.5"
```

```r
print(paste('Overall Blue Win Pct: ',
            round( 100 * nrow(subset(color, Winner == 'Blue')) / nrow(color), 2 )))
```

```
## [1] "Overall Blue Win Pct:  41.5"
```

```r
print(paste('Overall Male Red Wins: ',
            nrow(subset(color, Winner == 'Red' & gender == 'MALE'))))
```

```
## [1] "Overall Male Red Wins:  2265"
```

```r
print(paste('Overall Male Blue Wins: ',
            nrow(subset(color, Winner == 'Blue' & gender == 'MALE'))))
```

```
## [1] "Overall Male Blue Wins:  1624"
```

```r
print(paste('Overall Male Red Win Pct: ',
            round( 100 * nrow(subset(color, Winner == 'Red' & gender == 'FEMALE')) /
                     nrow(subset(color, gender == 'FEMALE')), 2)))
```

```
## [1] "Overall Male Red Win Pct:  61.04"
```

```r
print(paste('Overall Male Blue Win Pct: ',
            round( 100 * nrow(subset(color, Winner == 'Blue' & gender == 'FEMALE')) /
                     nrow(subset(color, gender == 'FEMALE')), 2 )))
```

```
## [1] "Overall Male Blue Win Pct:  38.96"
```

```r
print(paste('Overall Female Red Wins: ',
            nrow(subset(color, Winner == 'Red' & gender == 'FEMALE'))))
```

```
## [1] "Overall Female Red Wins:  246"
```

```r
print(paste('Overall Female Blue Wins: ',
            nrow(subset(color, Winner == 'Blue' & gender == 'FEMALE'))))
```

```
## [1] "Overall Female Blue Wins:  157"
```

```r
print(paste('Overall Female Red Win Pct: ',
            round( 100 * nrow(subset(color, Winner == 'Red' & gender == 'FEMALE')) /
                     nrow(subset(color, gender == 'FEMALE')), 2 )))
```

```
## [1] "Overall Female Red Win Pct:  61.04"
```

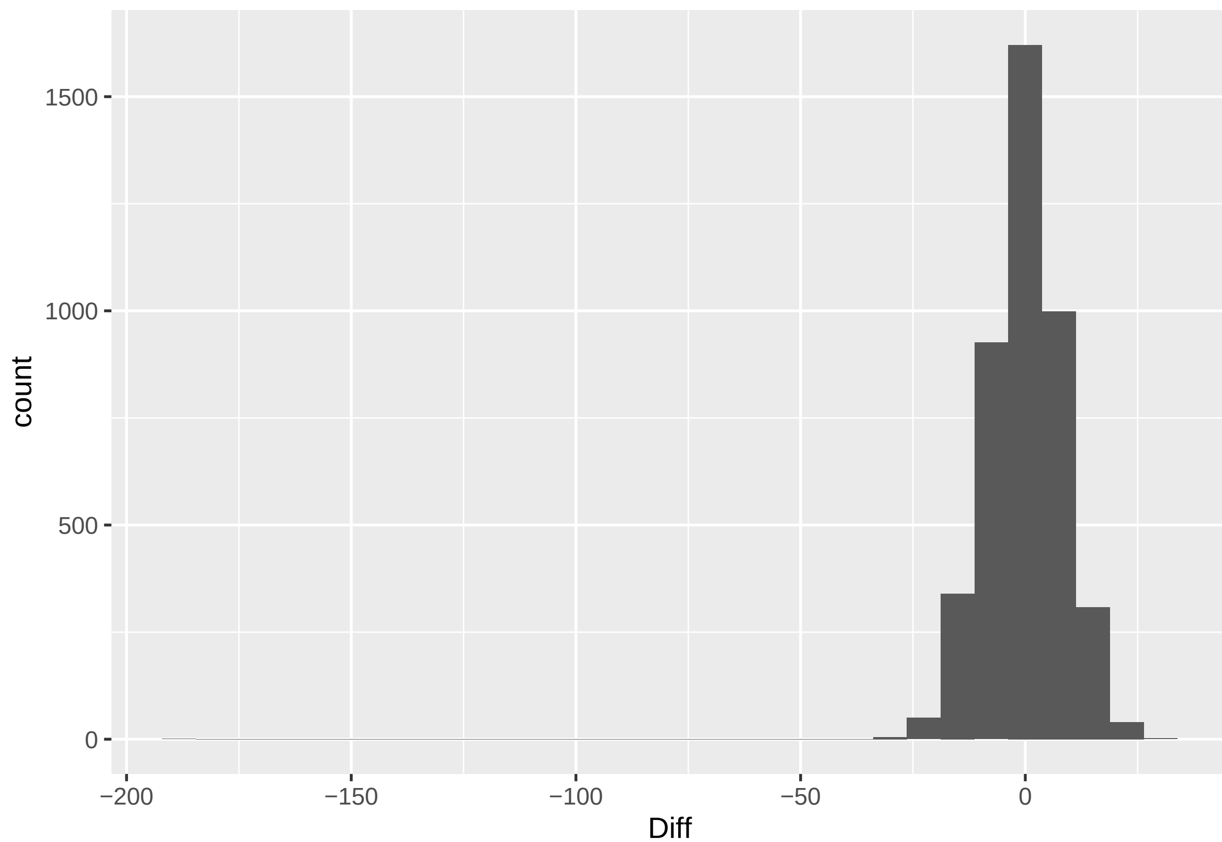
```r
print(paste('Overall Female Blue Win Pct: ',
            round( 100 * nrow(subset(color, Winner == 'Blue' & gender == 'FEMALE')) /
                     nrow(subset(color, gender == 'FEMALE')), 2 )))
```

```
## [1] "Overall Female Blue Win Pct:  38.96"
```

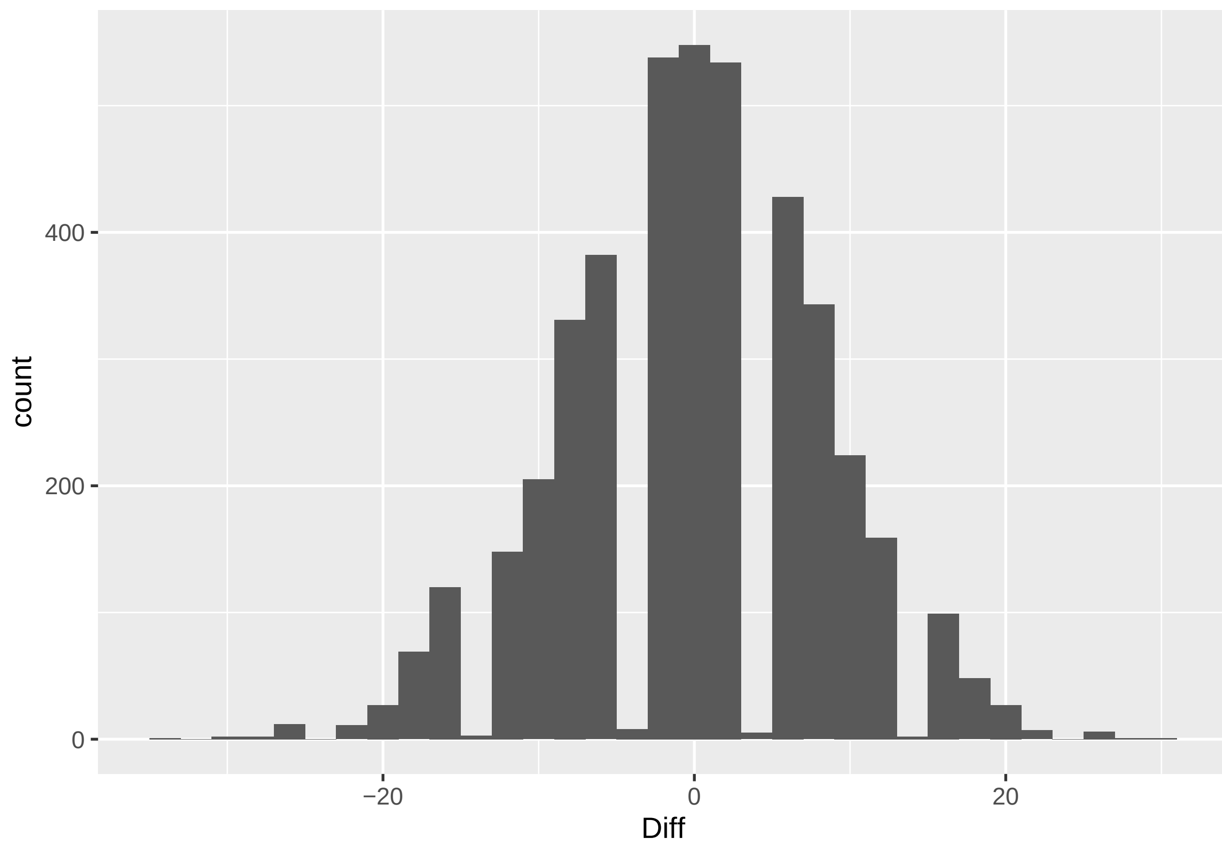
Inquiry 2: Does a reach advantage lead to more wins?

```r
# Create a smaller data frame
reach <- df[,c('Winner', 'gender', 'B_Reach_cms', 'R_Reach_cms')]

# Calculate the difference in reach (positive = blue advantage)
reach$Diff <- reach$B_Reach_cms - reach$R_Reach_cms

# Plot the reach differences
reach.plot <- ggplot(reach, aes(x = Diff))
reach.plot +
  geom_histogram()
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

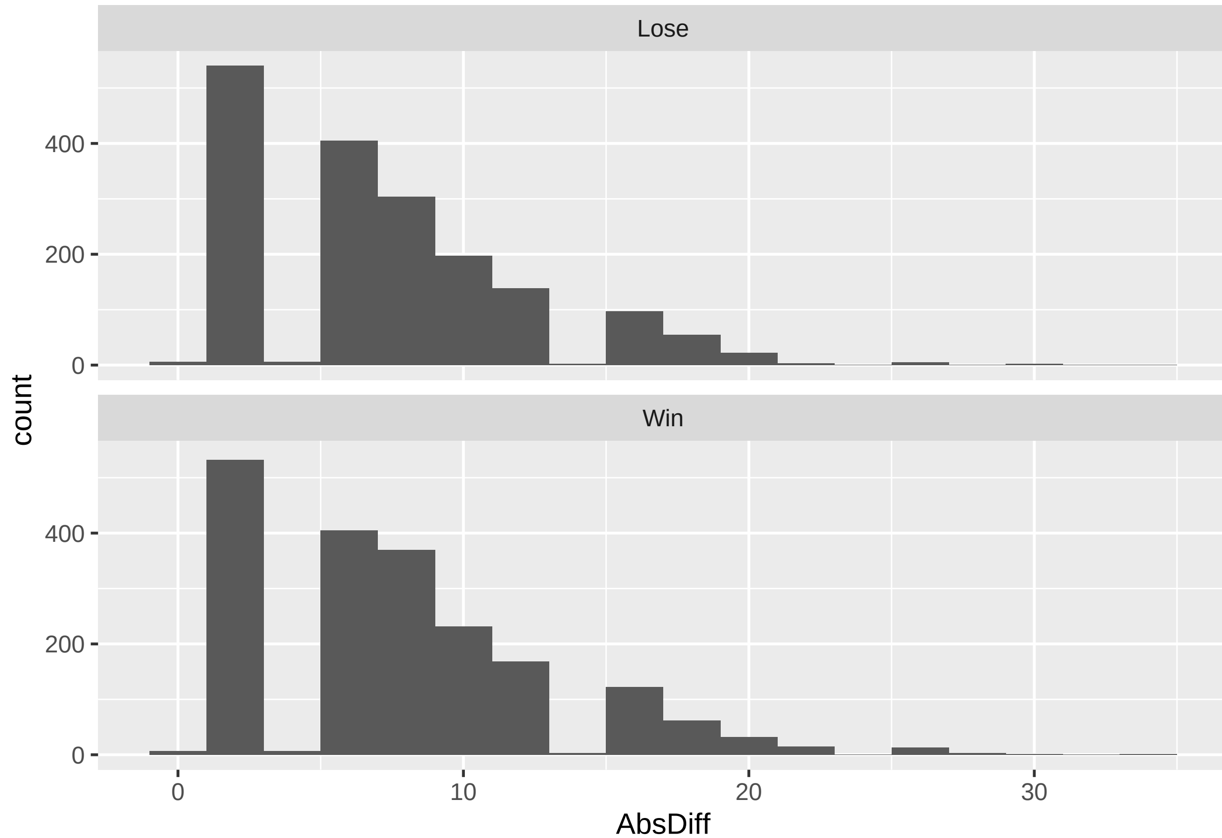



```
# There is an outlier that makes no sense, so remove it and replot
reach <- reach[reach$Diff > -50,]
reach.plot <- ggplot(reach, aes(x = Diff))
reach.plot +
  geom_histogram(binwidth = 2)
```




```r
# Remove all cases where the players had equal reach
reach <- subset(reach, !(Diff == 0))

# Identify the fighter with the longer reach
reach$Advantage <-
  case_when(
    reach$Diff > 0 ~ 'Blue',
    reach$Diff < 0 ~ 'Red'
  )

# Identify if the advantaged fighter won
reach$AdWin <-
  case_when(
    reach$Advantage == reach$Winner ~ 'Win',
    reach$Advantage != reach$Winner ~ 'Lose'
  )

reach$AdWin <- as.factor(reach$AdWin)

# Take the absolute value of the difference
reach$AbsDiff <- abs(reach$Diff)

# Plot the data
reach.hist <- ggplot(reach, aes(x = AbsDiff))
reach.hist +
  geom_histogram(binwidth = 2) +
```

```r
  facet_wrap(~ AdWin, ncol = 1)
```




```r
# Create the logistic regression
reach.model <- glm(AdWin ~ AbsDiff, data = reach, family = binomial())

# Display summary
summary(reach.model)
```

```
##
## Call:
## glm(formula = AdWin ~ AbsDiff, family = binomial(), data = reach)
##
## Deviance Residuals:
##    Min      1Q  Median      3Q     Max
## -1.463  -1.197   1.034   1.158   1.209
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.075187   0.059314  -1.268 0.204937
## AbsDiff      0.023771   0.006688   3.554 0.000379 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 5197.3  on 3755  degrees of freedom
```

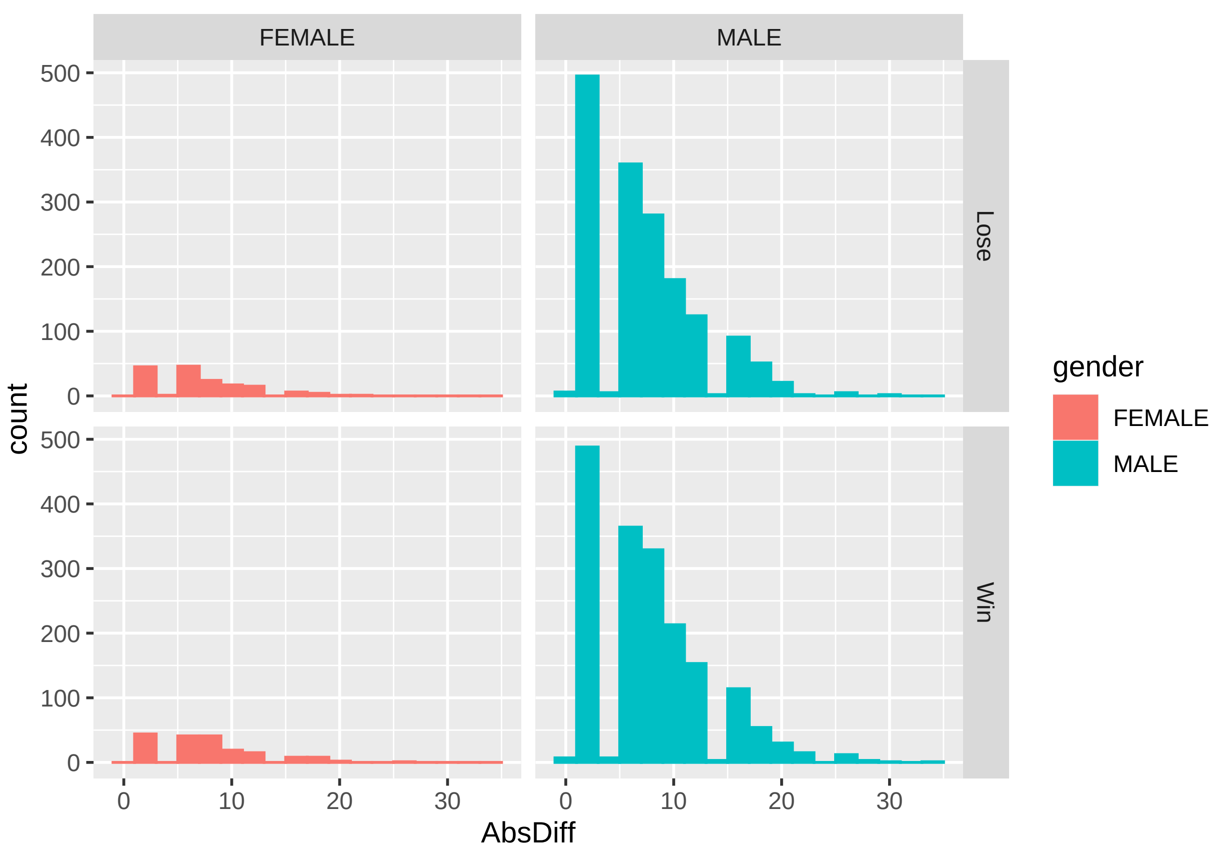
```
## Residual deviance: 5184.6  on 3754  degrees of freedom
## AIC: 5188.6
##
## Number of Fisher Scoring iterations: 3
```

```
# Calculate R^2

logisticPseudoR2s(reach.model)
```

```
## Pseudo R^2 for logistic regression
## Hosmer and Lemeshow R^2   0.002
## Cox and Snell R^2         0.003
## Nagelkerke R^2            0.005
```

Check to see how gender affects this

```
# Plot the data
reach_gender.hist <- ggplot(reach, aes(x = AbsDiff, color = gender))
reach_gender.hist +
  geom_histogram(binwidth = 2, aes(fill = gender)) +
  facet_grid(AdWin ~ gender)
```




```
# Create the logistic regression
reach_diff.model <- glm(AdWin ~ AbsDiff, data = reach, family = binomial())
reach_gender.model <- update(reach_diff.model, .~. + gender)

# Display summary
```

```r
summary(reach_diff.model)
```

```
##
## Call:
## glm(formula = AdWin ~ AbsDiff, family = binomial(), data = reach)
##
## Deviance Residuals:
##    Min      1Q  Median      3Q     Max
## -1.463  -1.197   1.034   1.158   1.209
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.075187   0.059314  -1.268 0.204937
## AbsDiff      0.023771   0.006688   3.554 0.000379 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 5197.3  on 3755  degrees of freedom
## Residual deviance: 5184.6  on 3754  degrees of freedom
## AIC: 5188.6
##
## Number of Fisher Scoring iterations: 3
```

```r
summary(reach_gender.model)
```

```
##
## Call:
## glm(formula = AdWin ~ AbsDiff + gender, family = binomial(),
##     data = reach)
##
## Deviance Residuals:
##    Min      1Q  Median      3Q     Max
## -1.462  -1.196   1.034   1.151   1.210
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.059510   0.119113  -0.500 0.617352
## AbsDiff      0.023784   0.006689   3.556 0.000377 ***
## genderMALE  -0.017338   0.114236  -0.152 0.879368
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 5197.3  on 3755  degrees of freedom
## Residual deviance: 5184.5  on 3753  degrees of freedom
## AIC: 5190.5
##
## Number of Fisher Scoring iterations: 3
```

```r
anova(reach_diff.model, reach_gender.model)
```

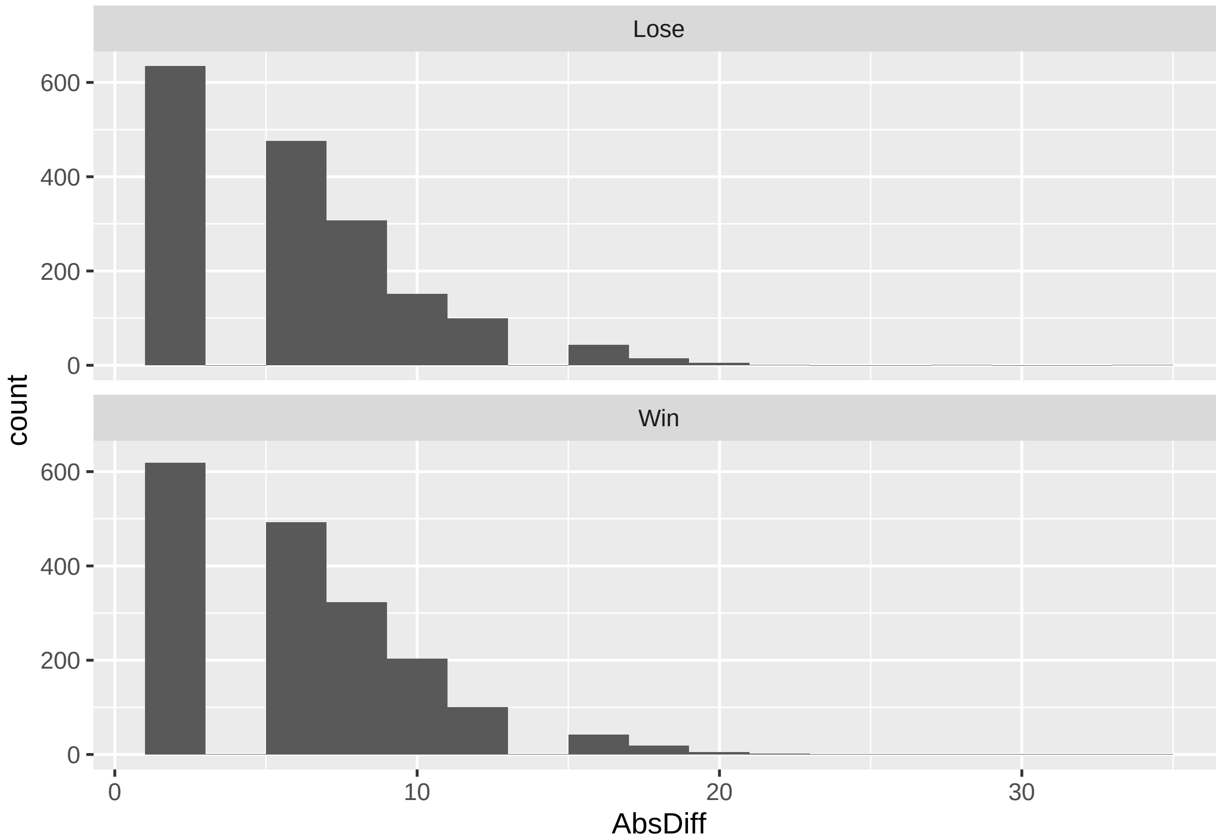
```
## Analysis of Deviance Table
```

```
##
## Model 1: AdWin ~ AbsDiff
## Model 2: AdWin ~ AbsDiff + gender
##   Resid. Df Resid. Dev Df Deviance
## 1      3754     5184.6
## 2      3753     5184.5  1  0.02304
```

It appears that adding gender does not add much to the model.

Inquiry 3: Does height have an advantage?

```
# Create a smaller data frame
height <- df[,c('Winner', 'B_Height_cms', 'R_Height_cms')]

# Calculate the difference in reach (positive = blue advantage)
height$Diff <- height$B_Height_cms - height$R_Height_cms

# Remove all cases where the players had equal reach
height <- subset(height, !(Diff == 0))

# Identify the fighter with the longer reach
height$Advantage <-
  case_when(
    height$Diff > 0 ~ 'Blue',
    height$Diff < 0 ~ 'Red'
  )

# Identify if the advantaged fighter won
height$AdWin <-
  case_when(
    height$Advantage == height$Winner ~ 'Win',
    height$Advantage != height$Winner ~ 'Lose'
  )

height$AdWin <- as.factor(height$AdWin)

# Take the absolute value of the difference
height$AbsDiff <- abs(height$Diff)

# Plot the data
height.hist <- ggplot(height, aes(x = AbsDiff))
height.hist +
  geom_histogram(binwidth = 2) +
  facet_wrap(~ AdWin, ncol = 1)
```




```r
# Create the logistic regression
height.model <- glm(AdWin ~ AbsDiff, data = height, family = binomial())

# Display summary
summary(height.model)
```

```
##
## Call:
## glm(formula = AdWin ~ AbsDiff, family = binomial(), data = height)
##
## Deviance Residuals:
##    Min      1Q  Median      3Q     Max
## -1.369  -1.190   1.102   1.165   1.181
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.045048   0.063632  -0.708    0.479
## AbsDiff      0.014712   0.008979   1.638    0.101
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 4904.4  on 3538  degrees of freedom
## Residual deviance: 4901.7  on 3537  degrees of freedom
## AIC: 4905.7
##
## Number of Fisher Scoring iterations: 3
```

```r
# Calculate R^2
logisticPseudoR2s(height.model)
```

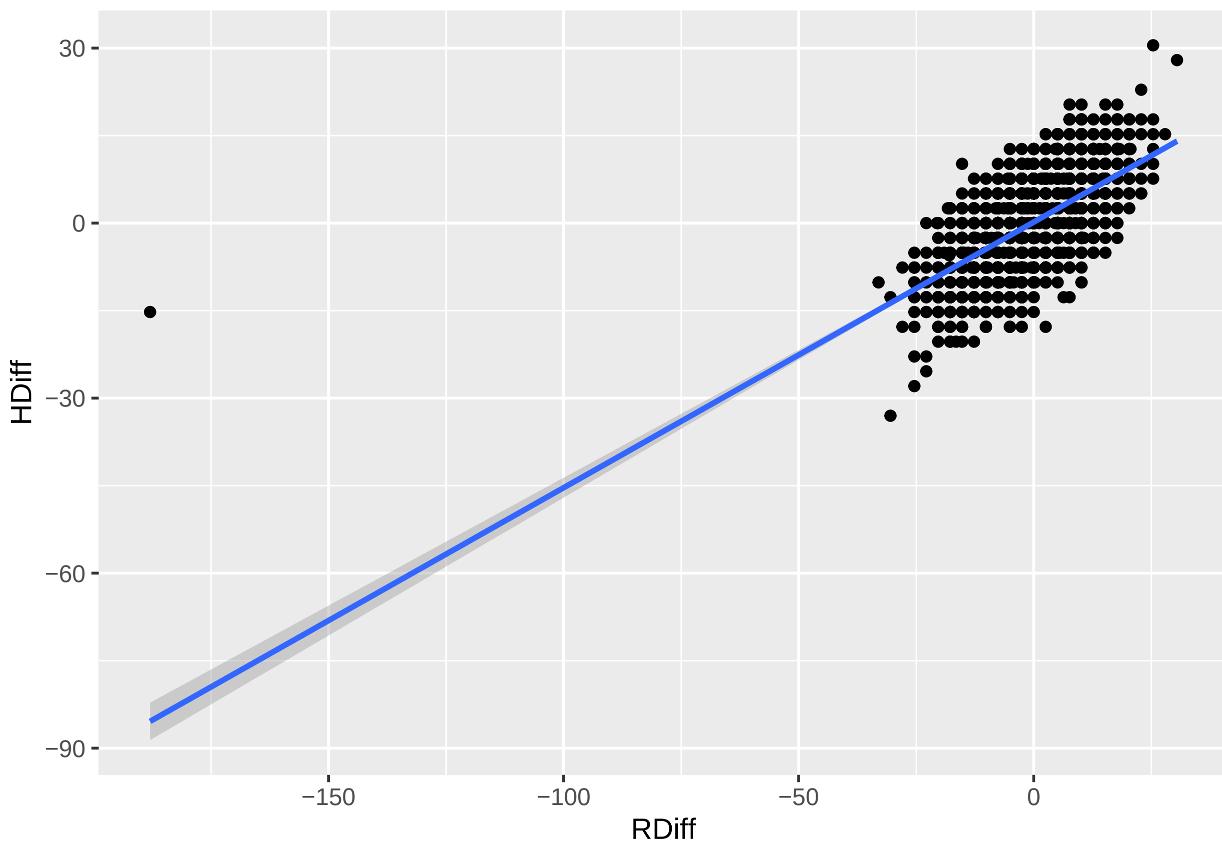
```
## Pseudo R^2 for logistic regression
## Hosmer and Lemeshow R^2   0.001
## Cox and Snell R^2         0.001
## Nagelkerke R^2            0.001
```

There is no significant advantage.

Inquiry 4: What about combining height and reach?

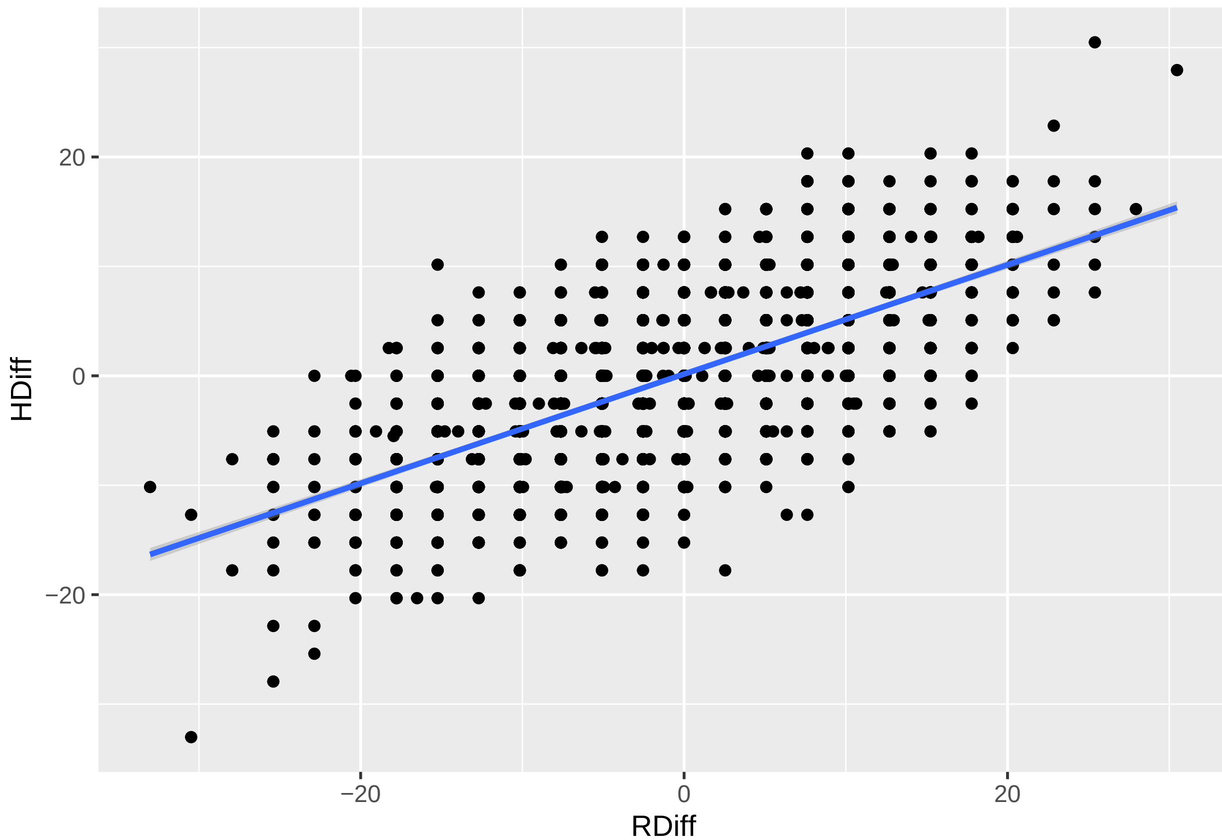
```r
# Create a smaller data frame
height_reach <- df[,c('Winner', 'B_Reach_cms', 'R_Reach_cms', 'B_Height_cms', 'R_Height_cms')]

# Calculate the difference in height and reach relative to Blue
height_reach$HDiff <- height_reach$B_Height_cms - height_reach$R_Height_cms
height_reach$RDiff <- height_reach$B_Reach_cms - height_reach$R_Reach_cms

# Draw a scatterplot of the differences
height_reach.scatter <- ggplot(height_reach, aes(x = RDiff, y = HDiff))
height_reach.scatter +
  geom_point() +
  geom_smooth(method = 'lm')
```

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



```r
# Remove bad data point and replot
height_reach <- height_reach[abs(height_reach$RDiff) < 50,]
height_reach.scatter <- ggplot(height_reach, aes(x = RDiff, y = HDiff))
height_reach.scatter +
  geom_point() +
  geom_smooth(method = 'lm')
```

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



This scatterplot only shows that the height and reach advantages have a correlation with each other, but it does not say anything about wins and losses.

```r
# Calculate the mean advantages for the winner
stat.desc(cbind(height_reach$HDiff, height_reach$RDiff))
```

```
##                         V1           V2
## nbr.val      4291.0000000 4291.00000000
## nbr.null      753.0000000  535.00000000
## nbr.na          0.0000000    0.00000000
## min           -33.0200000  -33.02000000
## max            30.4800000   30.48000000
## range          63.5000000   63.50000000
## sum           484.7200000 -399.16000000
## median          0.0000000    0.00000000
## mean            0.1129620   -0.09302261
## SE.mean         0.0983535    0.12757212
## CI.mean.0.95    0.1928237    0.25010732
```

```
## var            41.5086044   69.83450422
## std.dev         6.4427172    8.35670415
## coef.var       57.0343690  -89.83519766
```

Blue does not appear to have any average advantage/disadvantage with respect to height and reach.

```r
# Create a column that indicates whether blue wins/loses
height_reach$BlueWin <-
  case_when(
    height_reach$Winner == 'Blue' ~ 'Yes',
    height_reach$Winner == 'Red' ~ 'No'
  )
height_reach$BlueWin <- as.factor(height_reach$BlueWin)

# Create a logistic regression
height_reach.model <- glm(BlueWin ~ HDiff * RDiff, data = height_reach, family = binomial())

summary(height_reach.model)
```

```
##
## Call:
## glm(formula = BlueWin ~ HDiff * RDiff, family = binomial(), data = height_reach)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.2685  -1.0422  -0.9697   1.3072   1.5112
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.3474431  0.0347860  -9.988  < 2e-16 ***
## HDiff       -0.0062342  0.0063319  -0.985    0.325
## RDiff        0.0199174  0.0049033   4.062 4.86e-05 ***
## HDiff:RDiff  0.0001482  0.0004509   0.329    0.742
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 5824.1  on 4290  degrees of freedom
## Residual deviance: 5802.8  on 4287  degrees of freedom
## AIC: 5810.8
##
## Number of Fisher Scoring iterations: 4
```

As before, height does not appear to have an impact, but reach does.

Inquiry 5: Compare stances

```r
# Create a smaller data frame
stance <- df[,c('Winner', 'B_Stance', 'R_Stance')]

# Only keep matches where the stances are different
stance <- subset(stance, !(B_Stance == R_Stance))

# Eliminate matches with open stances since there aren't enough of them
```

```r
stance <- subset(stance, !(B_Stance == 'Open Stance'))
stance <- subset(stance, !(R_Stance == 'Open Stance'))

# Treat as factors
stance$B_Stance <- as.factor(stance$B_Stance)
stance$R_Stance <- as.factor(stance$R_Stance)
stance$Winner <- as.factor(stance$Winner)

### Loglinear Analysis

# Create multiple data frames relative to blue's stance
B_Ortho <- subset(stance, B_Stance == 'Orthodox')
B_South <- subset(stance, B_Stance == 'Southpaw')
B_Switch <- subset(stance, B_Stance == 'Switch')

# Show cross tables
CrossTable(B_Ortho$R_Stance, B_Ortho$Winner, sresid = TRUE, prop.t = FALSE, prop.c = FALSE, prop.chisq =
```

```
##
##    Cell Contents
## |-------------------------|
## |                   Count |
## |             Row Percent |
## |            Std Residual |
## |-------------------------|
##
## Total Observations in Table:  740
##
##                  | B_Ortho$Winner
## B_Ortho$R_Stance |      Blue |       Red | Row Total |
## -----------------|-----------|-----------|-----------|
##         Southpaw |       252 |       383 |       635 |
##                  |    39.685% |   60.315% |   85.811% |
##                  |     0.200 |    -0.160 |           |
## -----------------|-----------|-----------|-----------|
##           Switch |        38 |        67 |       105 |
##                  |    36.190% |   63.810% |   14.189% |
##                  |    -0.491 |     0.394 |           |
## -----------------|-----------|-----------|-----------|
##     Column Total |       290 |       450 |       740 |
## -----------------|-----------|-----------|-----------|
##
##
```

```r
CrossTable(B_South$R_Stance, B_South$Winner, sresid = TRUE, prop.t = FALSE, prop.c = FALSE, prop.chisq =
```

```
##
##    Cell Contents
## |-------------------------|
## |                   Count |
## |             Row Percent |
## |            Std Residual |
## |-------------------------|
##
```

```
## Total Observations in Table:  674
##
##                  | B_South$Winner
## B_South$R_Stance |      Blue |       Red | Row Total |
## -----------------|-----------|-----------|-----------|
##         Orthodox |       279 |       363 |       642 |
##                  |   43.458% |   56.542% |   95.252% |
##                  |     0.109 |    -0.095 |           |
## -----------------|-----------|-----------|-----------|
##           Switch |        12 |        20 |        32 |
##                  |   37.500% |   62.500% |    4.748% |
##                  |    -0.489 |     0.426 |           |
## -----------------|-----------|-----------|-----------|
##     Column Total |       291 |       383 |       674 |
## -----------------|-----------|-----------|-----------|
##
##
```

```
CrossTable(B_Switch$R_Stance, B_Switch$Winner, sresid = TRUE, prop.t = FALSE, prop.c = FALSE, prop.chis
```

```
##
##    Cell Contents
## |-------------------------|
## |                   Count |
## |             Row Percent |
## |            Std Residual |
## |-------------------------|
##
## Total Observations in Table:  176
##
##                   | B_Switch$Winner
## B_Switch$R_Stance |      Blue |       Red | Row Total |
## ------------------|-----------|-----------|-----------|
##          Orthodox |        63 |        75 |       138 |
##                   |   45.652% |   54.348% |   78.409% |
##                   |     0.134 |    -0.121 |           |
## ------------------|-----------|-----------|-----------|
##          Southpaw |        16 |        22 |        38 |
##                   |   42.105% |   57.895% |   21.591% |
##                   |    -0.256 |     0.231 |           |
## ------------------|-----------|-----------|-----------|
##      Column Total |        79 |        97 |       176 |
## ------------------|-----------|-----------|-----------|
##
##
```

```
# Create the contingency table
FullContingencyTable <- xtabs(~ R_Stance + B_Stance + Winner, data = stance)

# Create the saturated model
saturated.model <- loglm(~ R_Stance * B_Stance * Winner, data = FullContingencyTable)

print('Saturated Model')
```

```
## [1] "Saturated Model"
```

```r
summary(saturated.model)
```

```
## Formula:
## ~R_Stance * B_Stance * Winner
## attr(,"variables")
## list(R_Stance, B_Stance, Winner)
## attr(,"factors")
##          R_Stance B_Stance Winner R_Stance:B_Stance R_Stance:Winner
## R_Stance        1        0      0                 1               1
## B_Stance        0        1      0                 1               0
## Winner          0        0      1                 0               1
##          B_Stance:Winner R_Stance:B_Stance:Winner
## R_Stance               0                        1
## B_Stance               1                        1
## Winner                 1                        1
## attr(,"term.labels")
## [1] "R_Stance"                 "B_Stance"
## [3] "Winner"                   "R_Stance:B_Stance"
## [5] "R_Stance:Winner"          "B_Stance:Winner"
## [7] "R_Stance:B_Stance:Winner"
## attr(,"order")
## [1] 1 1 1 2 2 2 3
## attr(,"intercept")
## [1] 1
## attr(,"response")
## [1] 0
## attr(,".Environment")
## <environment: R_GlobalEnv>
##
## Statistics:
##                   X^2 df P(> X^2)
## Likelihood Ratio    0  0        1
## Pearson           NaN  0        1
```

```r
# Remove 3-way interaction

print('Remove 3-way term')
```

```
## [1] "Remove 3-way term"
```

```r
threeWay <- update(saturated.model, .~. - R_Stance:B_Stance:Winner)
summary(threeWay)
```

```
## Formula:
## . ~ R_Stance + B_Stance + Winner + R_Stance:B_Stance + R_Stance:Winner +
##     B_Stance:Winner
## attr(,"variables")
## list(., R_Stance, B_Stance, Winner)
## attr(,"factors")
##          R_Stance B_Stance Winner R_Stance:B_Stance R_Stance:Winner
## .               0        0      0                 0               0
## R_Stance        1        0      0                 1               1
## B_Stance        0        1      0                 1               0
## Winner          0        0      1                 0               1
##          B_Stance:Winner
```

```
## .                        0
## R_Stance                 0
## B_Stance                 1
## Winner                   1
## attr(,"term.labels")
## [1] "R_Stance"          "B_Stance"          "Winner"
## [4] "R_Stance:B_Stance" "R_Stance:Winner"   "B_Stance:Winner"
## attr(,"order")
## [1] 1 1 1 2 2 2
## attr(,"intercept")
## [1] 1
## attr(,"response")
## [1] 1
## attr(,".Environment")
## <environment: R_GlobalEnv>
##
## Statistics:
##                          X^2 df P(> X^2)
## Likelihood Ratio 0.00675545  6        1
## Pearson                 NaN  6      NaN
```

```r
anova(saturated.model, threeWay)
```

```
## LR tests for hierarchical log-linear models
##
## Model 1:
##  . ~ R_Stance + B_Stance + Winner + R_Stance:B_Stance + R_Stance:Winner + B_Stance:Winner
## Model 2:
##  ~R_Stance * B_Stance * Winner
##
##             Deviance df Delta(Dev) Delta(df) P(> Delta(Dev)
## Model 1   0.00675545  6
## Model 2   0.00000000  0 0.00675545         6              1
## Saturated 0.00000000  0 0.00000000         0              1
```

```r
# The model without the three way interactions seems to be a good fit still
# Start removing 2-way interaction

R_B.model <- update(threeWay, .~. - R_Stance:B_Stance)
R_Win.model <- update(threeWay, .~. - R_Stance:Winner)
B_Win.model <- update(threeWay, .~. - B_Stance:Winner)

print('Remove Red/Blue Stance interaction')
```

```
## [1] "Remove Red/Blue Stance interaction"
```

```r
anova(threeWay, R_B.model)
```

```
## LR tests for hierarchical log-linear models
##
## Model 1:
##  . ~ R_Stance + B_Stance + Winner + R_Stance:Winner + B_Stance:Winner
## Model 2:
##  . ~ R_Stance + B_Stance + Winner + R_Stance:B_Stance + R_Stance:Winner + B_Stance:Winner
##
##               Deviance df   Delta(Dev) Delta(df) P(> Delta(Dev)
```

```
## Model 1   1.896649e+03 12
## Model 2   6.755450e-03  6 1.896643e+03         6              0
## Saturated 0.000000e+00  0 6.755450e-03         6              1
```

```r
print('Remove Red Stance/Winner interaction')
```

```
## [1] "Remove Red Stance/Winner interaction"
```

```r
anova(threeWay, R_Win.model)
```

```
## LR tests for hierarchical log-linear models
##
## Model 1:
##   . ~ R_Stance + B_Stance + Winner + R_Stance:B_Stance + B_Stance:Winner
## Model 2:
##   . ~ R_Stance + B_Stance + Winner + R_Stance:B_Stance + R_Stance:Winner + B_Stance:Winner
##
##              Deviance df Delta(Dev) Delta(df) P(> Delta(Dev)
## Model 1   1.06397375  8
## Model 2   0.00675545  6 1.05721830         2        0.58942
## Saturated 0.00000000  0 0.00675545         6        1.00000
```

```r
print('Remove Blue Stance/Winner interaction')
```

```
## [1] "Remove Blue Stance/Winner interaction"
```

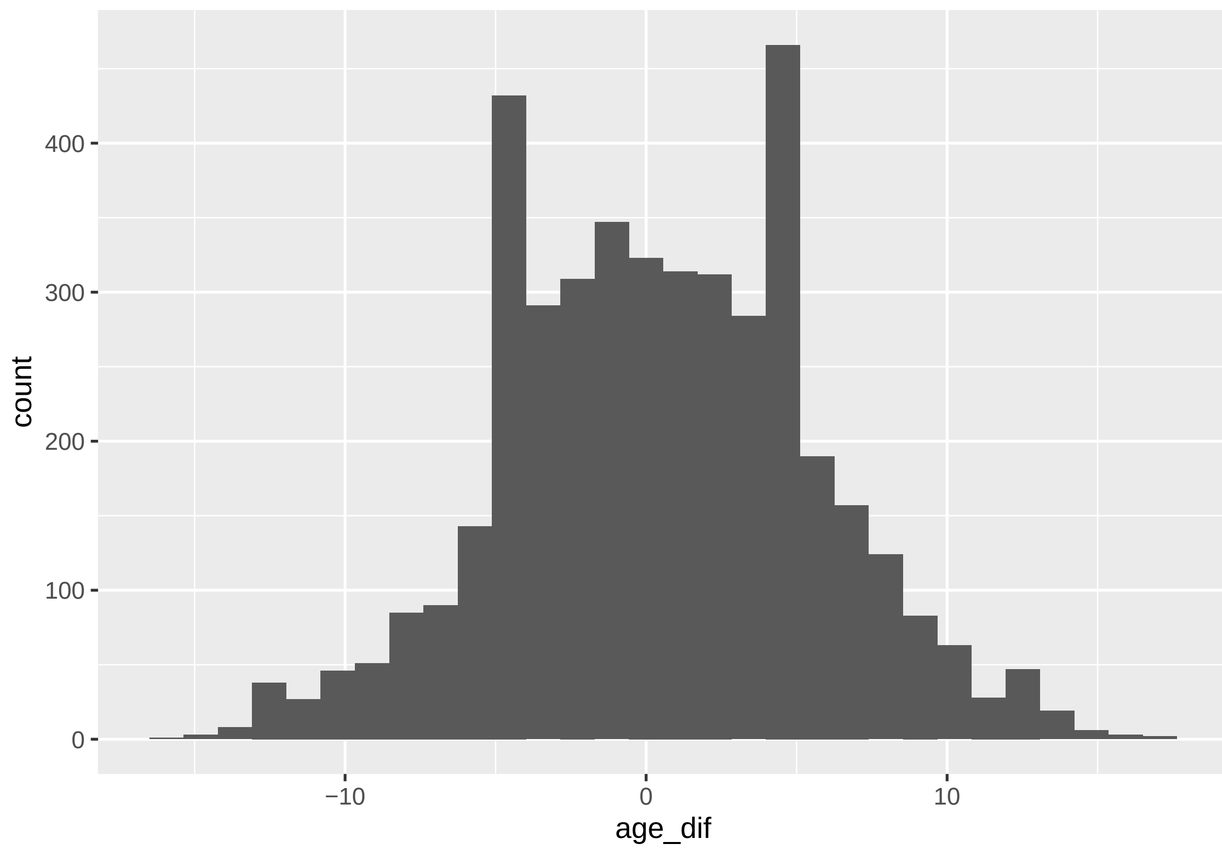
```r
anova(threeWay, B_Win.model)
```

```
## LR tests for hierarchical log-linear models
##
## Model 1:
##   . ~ R_Stance + B_Stance + Winner + R_Stance:B_Stance + R_Stance:Winner
## Model 2:
##   . ~ R_Stance + B_Stance + Winner + R_Stance:B_Stance + R_Stance:Winner + B_Stance:Winner
##
##              Deviance df Delta(Dev) Delta(df) P(> Delta(Dev)
## Model 1   2.33019873  9
## Model 2   0.00675545  6 2.32344328         3        0.50804
## Saturated 0.00000000  0 0.00675545         6        1.00000
```

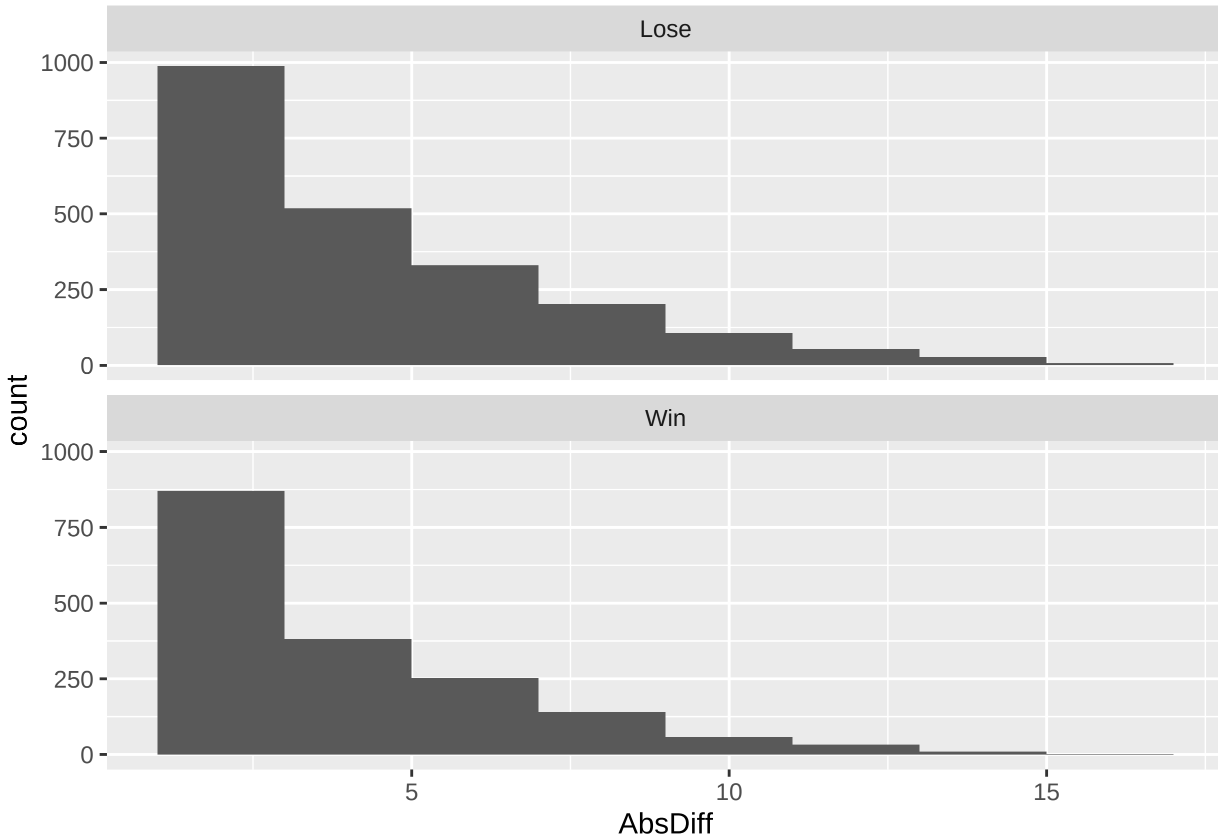
Inquiry 6: Does a younger age lead to more wins?

```r
# Create a smaller data frame
age <- df[,c('Winner', 'B_age', 'R_age', 'age_dif')]
# Plot the age differences
age.plot <- ggplot(age, aes(x = age_dif))
age.plot +
  geom_histogram()
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```r
# Remove all cases where the players had equal age
age <- subset(age, !(age_dif == 0))
# Identify the fighter with the younger age (positive = red advantage)
age$Advantage <-
  case_when(
    age$age_dif > 0 ~ 'Red',
    age$age_dif < 0 ~ 'Blue'
  )
# Identify if the advantaged fighter won
age$AdWin <-
  case_when(
    age$Advantage == age$Winner ~ 'Win',
    age$Advantage != age$Winner ~ 'Lose'
  )
age$AdWin <- as.factor(age$AdWin)
# Take the absolute value of the difference
age$AbsDiff <- abs(age$age_dif)
# Plot the data
age.hist <- ggplot(age, aes(x = AbsDiff))
age.hist +
  geom_histogram(binwidth = 2) +
  facet_wrap(~ AdWin, ncol = 1)
```




```r
# Create the logistic regression
age.model <- glm(AdWin ~ AbsDiff, data = age, family = binomial())
# Display summary
summary(age.model)
```

```
##
## Call:
## glm(formula = AdWin ~ AbsDiff, family = binomial(), data = age)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.1467  -1.0826  -0.9798   1.2529   1.5507
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.02119    0.05703  -0.372     0.71
## AbsDiff     -0.05148    0.01081  -4.761 1.93e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 5441.8  on 3968  degrees of freedom
## Residual deviance: 5418.7  on 3967  degrees of freedom
## AIC: 5422.7
##
```

```
## Number of Fisher Scoring iterations: 4
```

```r
# Calculate R^2
logisticPseudoR2s(reach.model)
```

```
## Pseudo R^2 for logistic regression
## Hosmer and Lemeshow R^2   0.002
## Cox and Snell R^2         0.003
## Nagelkerke R^2            0.005
```

There is a statistically significant correlation between the winner and having an age advantage (younger age).