# UFC Modeling

Brandon Trahms

## Intro

I decided to use UFC bout and personal fighter statistics. I am personally a fan of the sport and have done martial arts since I was a small kid which gives me some domain knowledge. I see a lot of great statistics and data here that would be cool to dissect. I am using a data set of historical UFC data taken from Kaggle.com. It provides a lot of data and variables but due to the denseness of this data set it make be hard to properly analyze. The fighter statistics are from when the fight took place but the sport of mma has change alot so the same statistics may not carry the same weight consistently throught out time.

## Data Prep

```
# merge two relevant data sets
process_data <- fight_data[c(1,2,33,38)]
process_data$date <- as.Date(process_data$date, format = "%B %d, %Y")
process_data <- merge(process_data, data, by = c("R_fighter", "B_fighter", "date"))

# tidy win_by variable for a model
process_data <- process_data[c(1, 2, 4, 7, 10:145)] %>%
  na.omit() %>%
  filter(Winner != "Draw")
  # mutate('Could Not Continue' = (win_by == "Could Not Continue")*1,
  #        'KO/TKO' = (win_by == "KO/TKO")*1,
  #        'Decision - Majority' = (win_by == "Decision - Majority")*1,
  #        'Decision - Split' = (win_by == "Decision - Split")*1,
  #        'Decision - Unanimous' = (win_by == "Decision - Unanimous")*1,
  #        'DQ' = (win_by == "DQ")*1,
  #        'Other' = (win_by == "Other")*1,
  #        'Overturned' = (win_by == "Overturned")*1,
  #        'Submission' = (win_by == "Submission")*1,
  #        "TKO - Doctor's Stoppage" = (win_by == "TKO - Doctor's Stoppage")*1)

process_data <- process_data %>%
  mutate(B_win_by_KO.TKO = B_win_by_KO.TKO/B_wins)

process_data <- process_data %>%
  mutate(B_win_by_Submission = B_win_by_Submission/B_wins)

process_data <- process_data %>%
  mutate(B_win_by_Decision = (B_win_by_Decision_Majority + B_win_by_Decision_Split + B_win_by_Decision_U

process_data <- process_data %>%
```

```
  mutate(R_win_by_KO.TKO = R_win_by_KO.TKO/R_wins)

process_data <- process_data %>%
  mutate(R_win_by_Submission = R_win_by_Submission/R_wins)

process_data <- process_data %>%
  mutate(R_win_by_Decision = (R_win_by_Decision_Majority + R_win_by_Decision_Split + R_win_by_Decision_


process_data$B_win_by_KO.TKO[is.nan(process_data$B_win_by_KO.TKO)] <- 0
process_data$B_win_by_Submission[is.nan(process_data$B_win_by_Submission)] <- 0
process_data$B_win_by_Decision[is.nan(process_data$B_win_by_Decision)] <- 0

process_data$R_win_by_KO.TKO[is.nan(process_data$R_win_by_KO.TKO)] <- 0
process_data$R_win_by_Submission[is.nan(process_data$R_win_by_Submission)] <- 0
process_data$R_win_by_Decision[is.nan(process_data$R_win_by_Decision)] <- 0

# reverse and append data
# make Winner variable numeric
org_data <- process_data
names(process_data) <- sub('^B_','r_', names(process_data))
names(process_data) <- sub('^R_','B_', names(process_data))
names(process_data) <- sub('^r_','R_', names(process_data))
org_data[4] <- (org_data$Winner == "Red") * 1 - 1 * (org_data$Winner == "Blue")
process_data[4] <- (process_data$Winner == "Blue") * 1 - 1 * (process_data$Winner == "Red")
process_data <- rbind(org_data, process_data)
process_data <- process_data %>% mutate(B_avg_SIG_STR_landed = B_avg_SIG_STR_landed/15 , R_avg_SIG_STR_

# drop Unused Variables
process_data <- process_data[c(3, 4, 5, 7, 8, 9, 10, 11, 16, 18, 24, 56, 57, 59, 60, 61, 65, 66, 69, 70
```

## 3 Data Science Questions

Can we predict the winner of a fight based on both fighter's full statistics? Success would be predicting fight winners with a R squared above .8 and failure would be the converse.

Can we predict how a fight will end based on both fighter's full statistics? Success would be predicting fight outcome method with a R squared above .8 and failure would be the converse.

Can we predict the winner of a fight based on both fighter's full statistics plus the way they win? Success would be predicting fight winners with a R squared above .9 and failure would be the converse.

## Modeling Winner of fight

Input: all Relevant Statistics variables(up to 138) of both fighters and their records used linearly * These are a lot of variable which means it will most likely find a lot of relevant and interesting relationships * However, with so many variables it also makes it harder to narrow in on if a specific relationship is more complex than linear

Output: 1 for Red as the winner -1 for Blue as the winner * The binomial nature allows for me to represent these factors as numeric without much drop in performance.

If this model works, I would really want to look at predicting win_by as it would require a more complex classifier and most likely multiple models. This data will have impacts on Sports Betting which can come with a lot of financial incentive which can raise some ethical questions of playing a rigged game where I know the outcome.

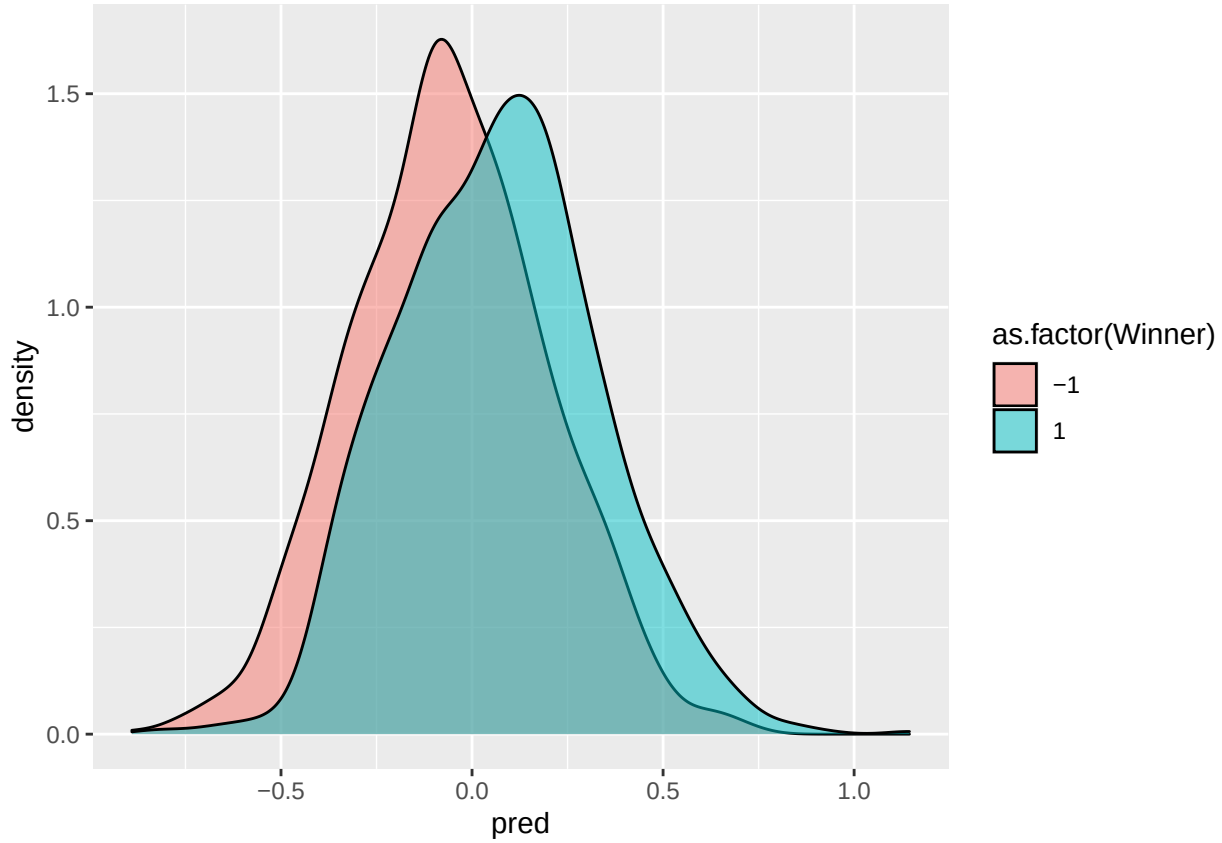
## Model build Winner

```
Winner_data <- process_data[-1]

set.seed(123456789)
Winner_data <- Winner_data[shuffle(nrow(Winner_data)),]
Train_set <- Winner_data[1:5500,]
Test_set <- Winner_data[5501:7696,]

win_model <- lm(Winner~ . - R_draw - B_draw, Train_set)

summary(win_model)
```

```
##
## Call:
## lm(formula = Winner ~ . - R_draw - B_draw, data = Train_set)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.8263 -0.9240 -0.1866  0.9326  1.8907
##
## Coefficients:
##                            Estimate Std. Error t value Pr(>|t|)
## (Intercept)              -1.716e-01  5.234e-01  -0.328 0.743057
## B_avg_KD                 -2.460e-02  4.135e-02  -0.595 0.551926
## B_avg_SIG_STR_pct        -1.730e-01  1.258e-01  -1.375 0.169262
## B_avg_opp_SIG_STR_pct     3.397e-01  1.248e-01   2.723 0.006490 **
## B_avg_TD_pct              5.146e-02  6.590e-02   0.781 0.434916
## B_avg_opp_TD_pct          8.639e-02  5.820e-02   1.484 0.137738
## B_avg_SUB_ATT            -3.219e-02  2.329e-02  -1.382 0.167004
## B_avg_SIG_STR_landed     -3.293e-02  1.419e-02  -2.320 0.020359 *
## B_avg_opp_SIG_STR_landed  3.011e-02  1.462e-02   2.060 0.039477 *
## B_avg_TD_landed          -4.031e-02  1.333e-02  -3.024 0.002505 **
## B_current_win_streak     -6.924e-03  1.032e-02  -0.671 0.502188
## B_current_lose_streak     8.814e-03  2.457e-02   0.359 0.719775
## B_wins                   -3.705e-02  6.781e-03  -5.464 4.85e-08 ***
## B_losses                  2.978e-02  1.079e-02   2.759 0.005820 **
## B_win_by_KO.TKO           1.373e-01  6.850e-02   2.004 0.045066 *
## B_win_by_Submission       1.580e-01  6.949e-02   2.274 0.022977 *
## B_Height_cms              1.123e-02  3.749e-03   2.994 0.002766 **
## B_Reach_cms              -9.151e-03  2.907e-03  -3.148 0.001651 **
## B_Weight_lbs             -4.299e-03  1.369e-03  -3.140 0.001697 **
## R_avg_KD                  1.362e-02  3.999e-02   0.341 0.733400
## R_avg_SIG_STR_pct         6.028e-02  1.246e-01   0.484 0.628626
## R_avg_opp_SIG_STR_pct    -3.668e-01  1.239e-01  -2.960 0.003093 **
## R_avg_TD_pct             -6.426e-02  6.579e-02  -0.977 0.328680
```

```
## R_avg_opp_TD_pct         -7.414e-02  5.815e-02  -1.275 0.202385
## R_avg_SUB_ATT             2.362e-02  2.338e-02   1.010 0.312423
## R_avg_SIG_STR_landed      5.539e-02  1.444e-02   3.835 0.000127 ***
## R_avg_opp_SIG_STR_landed -4.324e-02  1.458e-02  -2.967 0.003023 **
## R_avg_TD_landed           4.886e-02  1.317e-02   3.710 0.000209 ***
## R_current_win_streak      2.095e-02  1.035e-02   2.025 0.042952 *
## R_current_lose_streak     6.148e-05  2.479e-02   0.002 0.998021
## R_wins                    1.997e-02  6.884e-03   2.901 0.003733 **
## R_losses                 -1.388e-02  1.076e-02  -1.289 0.197285
## R_win_by_KO.TKO          -1.127e-01  6.808e-02  -1.656 0.097833 .
## R_win_by_Submission      -1.030e-01  6.876e-02  -1.499 0.134020
## R_Height_cms             -8.124e-03  3.723e-03  -2.182 0.029143 *
## R_Reach_cms               7.799e-03  2.890e-03   2.699 0.006985 **
## R_Weight_lbs              4.035e-03  1.377e-03   2.929 0.003410 **
## B_age                     2.831e-02  3.891e-03   7.275 3.96e-13 ***
## R_age                    -3.045e-02  3.869e-03  -7.870 4.24e-15 ***
## B_win_by_Decision         1.647e-01  6.259e-02   2.631 0.008531 **
## R_win_by_Decision        -1.609e-01  6.202e-02  -2.595 0.009478 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9702 on 5459 degrees of freedom
## Multiple R-squared:  0.06581,    Adjusted R-squared:  0.05897
## F-statistic: 9.615 on 40 and 5459 DF,  p-value: < 2.2e-16
```

```
Test_set <- Test_set %>% add_predictions(win_model)

Test_set %>%
  ggplot(aes(x = pred, fill = as.factor(Winner))) + geom_density(alpha = .5)
```




```
Test_set <- Test_set %>%
  mutate(pred = (pred > 0) * 1 - (pred < 0) * 1)

confusionMatrix(as.factor(Test_set$pred), as.factor(Test_set$Winner))
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction  -1   1
##         -1 659 439
##         1  434 664
##
##                Accuracy : 0.6025
##                  95% CI : (0.5816, 0.623)
##     No Information Rate : 0.5023
##     P-Value [Acc > NIR] : <2e-16
##
##                   Kappa : 0.2049
##
##  Mcnemar's Test P-Value : 0.8923
##
##             Sensitivity : 0.6029
##             Specificity : 0.6020
##          Pos Pred Value : 0.6002
##          Neg Pred Value : 0.6047
```

```
##              Prevalence : 0.4977
##          Detection Rate : 0.3001
##    Detection Prevalence : 0.5000
##       Balanced Accuracy : 0.6025
##
##        'Positive' Class : -1
##
```

```
# # NN
# # Fit the model
# model <- multinom(Winner ~., data = Train_set, maxit = 500)
# # Make predictions
# predicted.classes <- model %>% predict(Test_set)
# Test_set <- cbind(Test_set, predicted.classes)
#
# confusionMatrix(Test_set$predicted.classes, as.factor(Test_set$Winner))
# Test_set <- Test_set[-44]
```

## Model build Win_by

```
Win_data <- process_data[-(2)]
Win_data$win_by <- as.factor(Win_data$win_by)
set.seed(1234)
Win_data <- Win_data[shuffle(nrow(Winner_data)),]
Train_set <- Win_data[1:5500,]
Test_set <- Win_data[5501:7696,]

# Conditional Inference Tree
fit <- ctree(win_by ~ . , data=Train_set)
# plot(fit, main="Conditional Inference Tree for data")
png("fit.png", res=80, height=1600, width=14000)
   plot(fit)
dev.off()
```

```
## pdf
##   2
```

```
summary(fit)
```

```
##     Length      Class       Mode
##          1 BinaryTree         S4
```

```
Test_set <- cbind(Test_set, predict(fit, Test_set))
names(Test_set)[44] <- "pred"

confusionMatrix(Test_set$pred, Test_set$win_by)
```

```
## Confusion Matrix and Statistics
##
```

```
##                          Reference
## Prediction                Decision - Majority Decision - Split
##   Decision - Majority                       0                0
##   Decision - Split                          0                0
##   Decision - Unanimous                      6              183
##   DQ                                        0                0
##   KO/TKO                                    5               55
##   Submission                                0                0
##   TKO - Doctor's Stoppage                   0                0
##                          Reference
## Prediction                Decision - Unanimous  DQ KO/TKO Submission
##   Decision - Majority                        0   0      0          0
##   Decision - Split                           0   0      0          0
##   Decision - Unanimous                     636   2    361        284
##   DQ                                         0   0      0          0
##   KO/TKO                                   220   3    325         93
##   Submission                                 0   0      0          0
##   TKO - Doctor's Stoppage                    0   0      0          0
##                          Reference
## Prediction                TKO - Doctor's Stoppage
##   Decision - Majority                           0
##   Decision - Split                              0
##   Decision - Unanimous                         15
##   DQ                                            0
##   KO/TKO                                        8
##   Submission                                    0
##   TKO - Doctor's Stoppage                       0
##
## Overall Statistics
##
##                Accuracy : 0.4376
##                  95% CI : (0.4167, 0.4587)
##     No Information Rate : 0.3898
##     P-Value [Acc > NIR] : 2.756e-06
##
##                   Kappa : 0.1146
##
##  Mcnemar's Test P-Value : NA
##
## Statistics by Class:
##
##                      Class: Decision - Majority Class: Decision - Split
## Sensitivity                            0.000000                 0.0000
## Specificity                            1.000000                 1.0000
## Pos Pred Value                              NaN                    NaN
## Neg Pred Value                         0.994991                 0.8916
## Prevalence                             0.005009                 0.1084
## Detection Rate                         0.000000                 0.0000
## Detection Prevalence                   0.000000                 0.0000
## Balanced Accuracy                      0.500000                 0.5000
##                      Class: Decision - Unanimous Class: DQ Class: KO/TKO
## Sensitivity                               0.7430  0.000000        0.4738
## Specificity                               0.3649  1.000000        0.7457
## Pos Pred Value                            0.4277       NaN        0.4584
```

```
## Neg Pred Value                            0.6897  0.997723        0.7572
## Prevalence                                0.3898  0.002277        0.3124
## Detection Rate                            0.2896  0.000000        0.1480
## Detection Prevalence                      0.6771  0.000000        0.3229
## Balanced Accuracy                         0.5540  0.500000        0.6097
##                      Class: Submission Class: TKO - Doctor's Stoppage
## Sensitivity                     0.0000                         0.00000
## Specificity                     1.0000                         1.00000
## Pos Pred Value                     NaN                             NaN
## Neg Pred Value                  0.8283                         0.98953
## Prevalence                      0.1717                         0.01047
## Detection Rate                  0.0000                         0.00000
## Detection Prevalence            0.0000                         0.00000
## Balanced Accuracy               0.5000                         0.50000
```

```
# NN
# Fit the model
model <- multinom(win_by ~., data = Train_set, maxit = 500)
```

```
## # weights:  308 (258 variable)
## initial  value 10702.505820
## iter  10 value 8277.194426
## iter  20 value 8153.839858
## iter  30 value 8108.784826
## iter  40 value 8089.950054
## iter  50 value 8043.657780
## iter  60 value 7929.766246
## iter  70 value 7738.127665
## iter  80 value 7503.658085
## iter  90 value 7437.367300
## iter 100 value 7375.073371
## iter 110 value 7308.377588
## iter 120 value 7266.967037
## iter 130 value 7227.474572
## iter 140 value 7208.298286
## iter 150 value 7196.571671
## iter 160 value 7189.630245
## iter 170 value 7187.212193
## iter 180 value 7184.725220
## iter 190 value 7182.443592
## iter 200 value 7181.621050
## iter 210 value 7180.856005
## iter 220 value 7180.487120
## iter 230 value 7180.257535
## iter 240 value 7180.142998
## iter 250 value 7180.101477
## iter 260 value 7180.090386
## final  value 7180.089731
## converged
```

```
# Make predictions
predicted.classes <- model %>% predict(Test_set)
Test_set <- cbind(Test_set, predicted.classes)
```

```
confusionMatrix(Test_set$predicted.classes, Test_set$win_by)
```

```
## Confusion Matrix and Statistics
##
##                          Reference
## Prediction                Decision - Majority Decision - Split
##   Decision - Majority                       0                0
##   Decision - Split                          0                0
##   Decision - Unanimous                      7              168
##   DQ                                        0                0
##   KO/TKO                                    4               58
##   Submission                                0               12
##   TKO - Doctor's Stoppage                   0                0
##                          Reference
## Prediction                Decision - Unanimous  DQ KO/TKO Submission
##   Decision - Majority                        0   0      0          0
##   Decision - Split                           0   0      0          0
##   Decision - Unanimous                     592   4    318        232
##   DQ                                         0   0      0          0
##   KO/TKO                                   231   1    341        101
##   Submission                                33   0     27         44
##   TKO - Doctor's Stoppage                    0   0      0          0
##                          Reference
## Prediction                TKO - Doctor's Stoppage
##   Decision - Majority                           0
##   Decision - Split                              0
##   Decision - Unanimous                         10
##   DQ                                            0
##   KO/TKO                                       10
##   Submission                                    3
##   TKO - Doctor's Stoppage                       0
##
## Overall Statistics
##
##                Accuracy : 0.4449
##                  95% CI : (0.424, 0.466)
##     No Information Rate : 0.3898
##     P-Value [Acc > NIR] : 8.231e-08
##
##                   Kappa : 0.1438
##
##  Mcnemar's Test P-Value : NA
##
## Statistics by Class:
##
##                      Class: Decision - Majority Class: Decision - Split
## Sensitivity                            0.000000                  0.0000
## Specificity                            1.000000                  1.0000
## Pos Pred Value                              NaN                     NaN
## Neg Pred Value                         0.994991                  0.8916
## Prevalence                             0.005009                  0.1084
## Detection Rate                         0.000000                  0.0000
## Detection Prevalence                   0.000000                  0.0000
```

```
## Balanced Accuracy                       0.500000                  0.5000
##                       Class: Decision - Unanimous Class: DQ Class: KO/TKO
## Sensitivity                                0.6916  0.000000        0.4971
## Specificity                                0.4485  1.000000        0.7318
## Pos Pred Value                             0.4448       NaN        0.4571
## Neg Pred Value                             0.6948  0.997723        0.7621
## Prevalence                                 0.3898  0.002277        0.3124
## Detection Rate                             0.2696  0.000000        0.1553
## Detection Prevalence                       0.6061  0.000000        0.3397
## Balanced Accuracy                          0.5700  0.500000        0.6144
##                       Class: Submission Class: TKO - Doctor's Stoppage
## Sensitivity                     0.11671                         0.00000
## Specificity                     0.95877                         1.00000
## Pos Pred Value                  0.36975                             NaN
## Neg Pred Value                  0.83967                         0.98953
## Prevalence                      0.17168                         0.01047
## Detection Rate                  0.02004                         0.00000
## Detection Prevalence            0.05419                         0.00000
## Balanced Accuracy               0.53774                         0.50000
```

## Real Predictions

```
# Real Test of win_by
Real_win_by <-  Train_set %>%
  rbind(RealData[-c(1,2,4)])
Real_win_by <- Real_win_by[5501:5514,]
predicted.classes <- model %>% predict(Real_win_by)
Real_win_by <- cbind(Real_win_by, predicted.classes)
confusionMatrix( Real_win_by$predicted.classes, Real_win_by$win_by)
```

```
## Confusion Matrix and Statistics
##
##                          Reference
## Prediction                Decision - Majority Decision - Split
##   Decision - Majority                       0                0
##   Decision - Split                          0                0
##   Decision - Unanimous                      0                0
##   DQ                                        0                0
##   KO/TKO                                    0                1
##   Submission                                0                0
##   TKO - Doctor's Stoppage                   0                0
##                          Reference
## Prediction                Decision - Unanimous DQ KO/TKO Submission
##   Decision - Majority                        0  0      0          0
##   Decision - Split                           0  0      0          0
##   Decision - Unanimous                       2  0      2          2
##   DQ                                         0  0      0          0
##   KO/TKO                                     2  0      4          1
##   Submission                                 0  0      0          0
##   TKO - Doctor's Stoppage                    0  0      0          0
##                          Reference
```

```
## Prediction                TKO - Doctor's Stoppage
##   Decision - Majority                           0
##   Decision - Split                              0
##   Decision - Unanimous                          0
##   DQ                                            0
##   KO/TKO                                        0
##   Submission                                    0
##   TKO - Doctor's Stoppage                       0
##
## Overall Statistics
##
##                Accuracy : 0.4286
##                  95% CI : (0.1766, 0.7114)
##     No Information Rate : 0.4286
##     P-Value [Acc > NIR] : 0.601
##
##                   Kappa : 0.0968
##
##  Mcnemar's Test P-Value : NA
##
## Statistics by Class:
##
##                      Class: Decision - Majority Class: Decision - Split
## Sensitivity                                  NA                 0.00000
## Specificity                                   1                 1.00000
## Pos Pred Value                               NA                     NaN
## Neg Pred Value                               NA                 0.92857
## Prevalence                                    0                 0.07143
## Detection Rate                                0                 0.00000
## Detection Prevalence                          0                 0.00000
## Balanced Accuracy                            NA                 0.50000
##                      Class: Decision - Unanimous Class: DQ Class: KO/TKO
## Sensitivity                               0.5000        NA        0.6667
## Specificity                               0.6000         1        0.5000
## Pos Pred Value                            0.3333        NA        0.5000
## Neg Pred Value                            0.7500        NA        0.6667
## Prevalence                                0.2857         0        0.4286
## Detection Rate                            0.1429         0        0.2857
## Detection Prevalence                      0.4286         0        0.5714
## Balanced Accuracy                         0.5500        NA        0.5833
##                      Class: Submission Class: TKO - Doctor's Stoppage
## Sensitivity                     0.0000                             NA
## Specificity                     1.0000                              1
## Pos Pred Value                     NaN                             NA
## Neg Pred Value                  0.7857                             NA
## Prevalence                      0.2143                              0
## Detection Rate                  0.0000                              0
## Detection Prevalence            0.0000                              0
## Balanced Accuracy               0.5000                             NA
```

```
# Real Test of Winner
RealWinner <- RealData %>% add_predictions(win_model)
RealWinner <- RealWinner %>%
  mutate(pred = (pred > 0) * 1 - (pred < 0) * 1, Winner = (Winner == "Red") * 1 - (Winner == "Blue") * 1
```

```
confusionMatrix(as.factor(RealWinner$pred), as.factor(RealWinner$Winner))
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction -1 1
##         -1  4 4
##         1   3 3
##
##                Accuracy : 0.5
##                  95% CI : (0.2304, 0.7696)
##     No Information Rate : 0.5
##     P-Value [Acc > NIR] : 0.6047
##
##                   Kappa : 0
##
##  Mcnemar's Test P-Value : 1.0000
##
##             Sensitivity : 0.5714
##             Specificity : 0.4286
##          Pos Pred Value : 0.5000
##          Neg Pred Value : 0.5000
##              Prevalence : 0.5000
##          Detection Rate : 0.2857
##    Detection Prevalence : 0.5714
##       Balanced Accuracy : 0.5000
##
##        'Positive' Class : -1
##
```