The Impact of Play Clock Timing on NFL Play Outcomes

Dominic Hoar-Weiler, Nathan Yao, and Justin Fung

January 6, 2025

Contents

1 Introduction

One moment etched in the hearts of Minnesota Vikings fans is the 2018 NFL playoff game against the New Orleans Saints. With just 5 seconds left on the play clock, Case Keenum launches a high, spiraling pass to Stefon Diggs, who streaks down the right sideline. The crowd erupts as Diggs makes the catch and sprints into the end zone, completing one of the most iconic touchdowns in NFL history. This unforgettable play, now famously known as the Minneapolis Miracle, highlights the critical role of the play clock. In football, the play clock isn't just a countdown, it's a pressure cooker that influences decision making, timing, and ultimately, the outcome of a play.

The focus of this research is to examine how the play clock at the snap, along with other variables, impacts play success. By investigating this relationship, we aim to develop a predictive model that forecasts play outcomes based on the timing of the snap.

2 Exploratory Analysis:

2.1 Background and Variables

We analyze multiple datasets, including player statistics, play-by-play data, and game metadata, sourced from the 2024–2025 NFL season as part of the Big Data Bowl competition.

The data that was used for most of analysis came from the "plays.csv" data set, A number of exploratory variables were recorded:
- passResult:Dropback outcome of the play (C: Complete pass, I: Incomplete pass , S: Quarterback sack, IN: Intercepted pass, R: Scramble, text)
- yardsGained:Net yards gained by the offense, including penalty yardage (numeric)
- playClockAtSnap:What the play clock value was at time of snap (numeric)
- yardsToGo:Distance needed for a first down (numeric)
- down: Down (numeric)
- passLength: The distance beyond the LOS that the ball traveled not including

yards into the endzone. If thrown behind LOS, the value is negative. (numeric)
- pff_passCoverage: The pass coverage concept employed by the defense on the play (text)
- dropbackType: The type of drop back after the snap by the QB (Traditional, Designed Rollout, Scramble, Scramble Rollout, Designed Rollout Left, Designed Rollout Right, Scramble Rollout Left, Scramble Rollout Right, Designed Run, QB Draw, Rollout, text)

and many more.

2.2 Some Exploratory Data Analysis with Play Clock at Snap

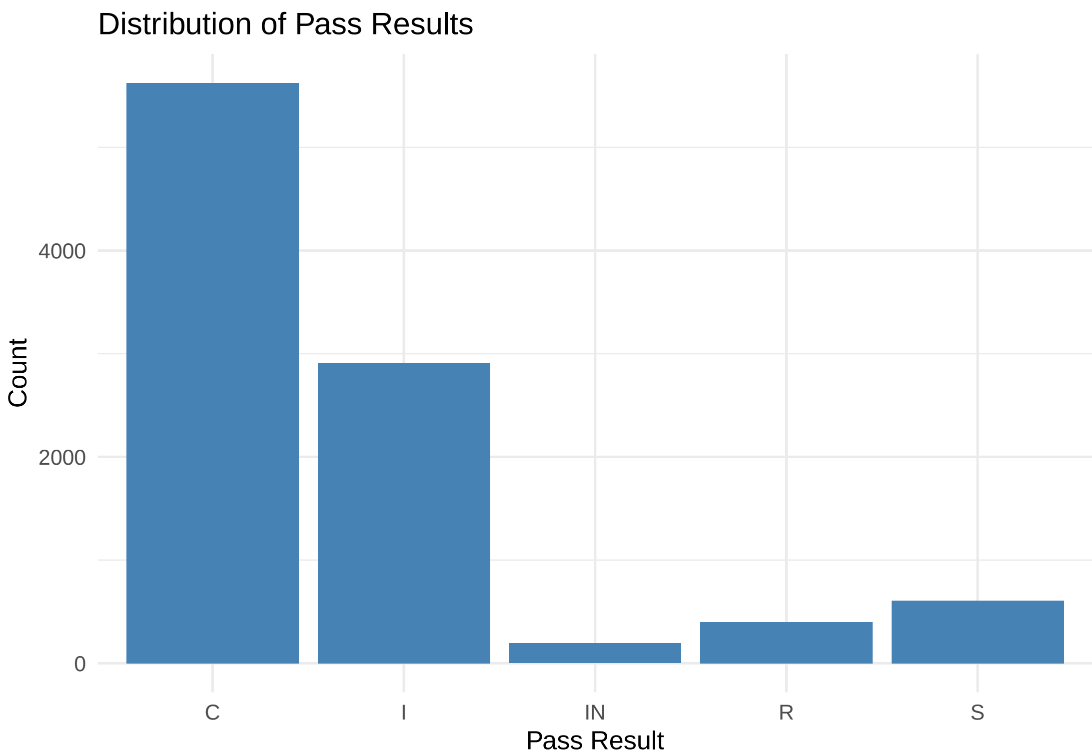
2.2.1 Distributions of Important Variables

There were a variety of possible variables that along with playClockAtSnap that would be important in our analysis. This section explores distributions of important variables and the relationships between them. Below are some of the more important variables and their distributions that we looked at in our analysis:

passResult (Categorical): We have filtered out the NA values which indicate non-pass plays, as we are only trying to explore passing plays. (C: Complete pass, I: Incomplete pass , S: Quarterback sack, IN: Intercepted pass, R: Scramble)

```
plays_filtered <- plays_raw_data %>%
  filter(!is.na(passResult))


ggplot(plays_filtered, aes(x = passResult)) +
```

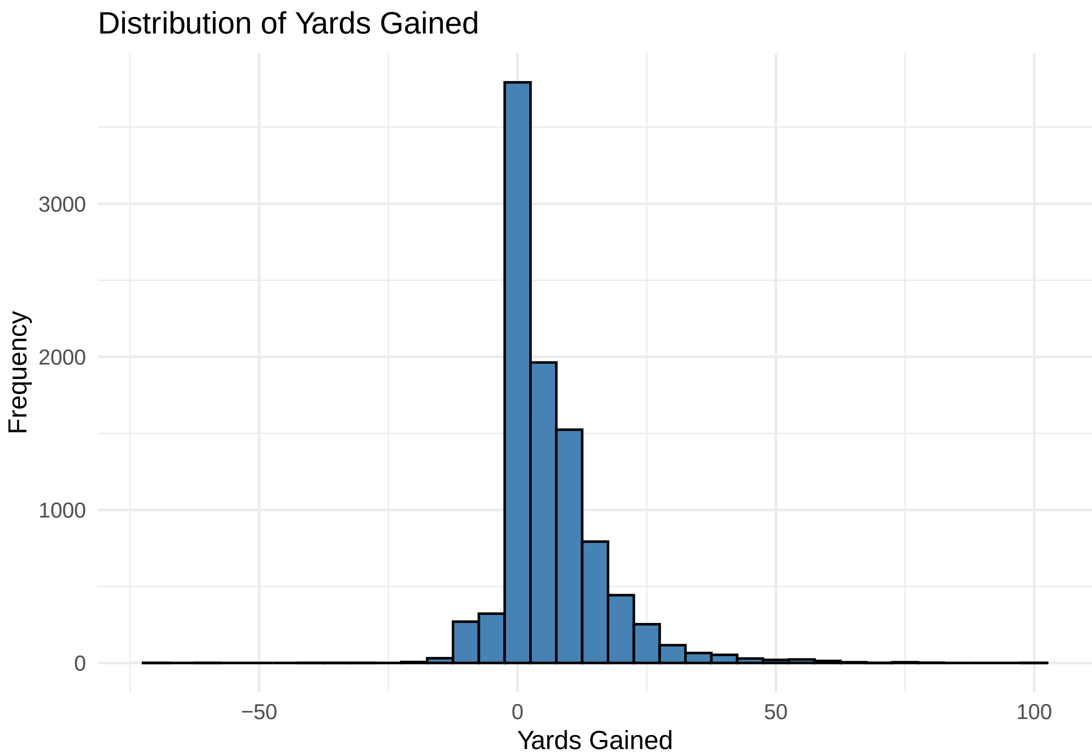
```r
  geom_bar(fill = "steelblue") +
  labs(
    title = "Distribution of Pass Results",
    x = "Pass Result",
    y = "Count"
  ) +
  theme_minimal()
```



The most frequent pass result is C (completed pass). This makes sense as a lot of the time the quarterback will complete his pass to a receiver. The second most frequent category is I (incomplete pass), which is also expected. After that, the next most frequent categories are in order as follows: S (quarterback sack), R (scramble–quarterback improvised run), and IN (intercepted pass).

yardsGained (Quantitative):

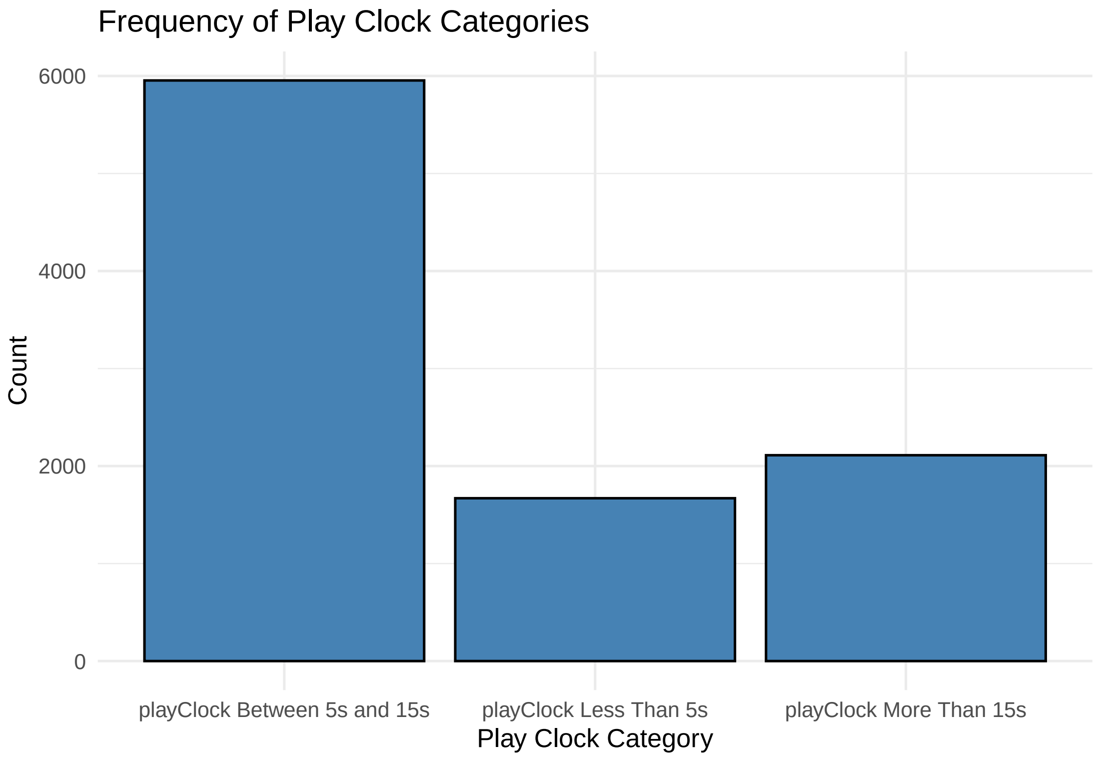
```r
ggplot(plays_filtered, aes(x = yardsGained)) +
  geom_histogram(binwidth = 5, fill = "steelblue", color = "black") +
  labs(
    title = "Distribution of Yards Gained",
    x = "Yards Gained",
    y = "Frequency"
  ) +
  theme_minimal()
```



Yards gained looks like it is slightly normally distributed and is right skewed. The overwhelming majority of plays result in 0 yards gained according to the graphic. This makes sense as all incomplete passes result in 0 yards gained, assuming no penalties.

playClockAtSnap (Quantitative): For the play clock at snap, we split the quantitative variable into 3 buckets: High (>=15 seconds), Medium (5–15 seconds), and Low (<5 seconds). These thresholds were chosen to reflect typical game scenarios: early snaps, moderate decision-making time, and rushed snaps. In this case, we created a new variable called playClockCategory which is a categorical variable that has counts of plays in each of the 3 categories. Below is the distribution:

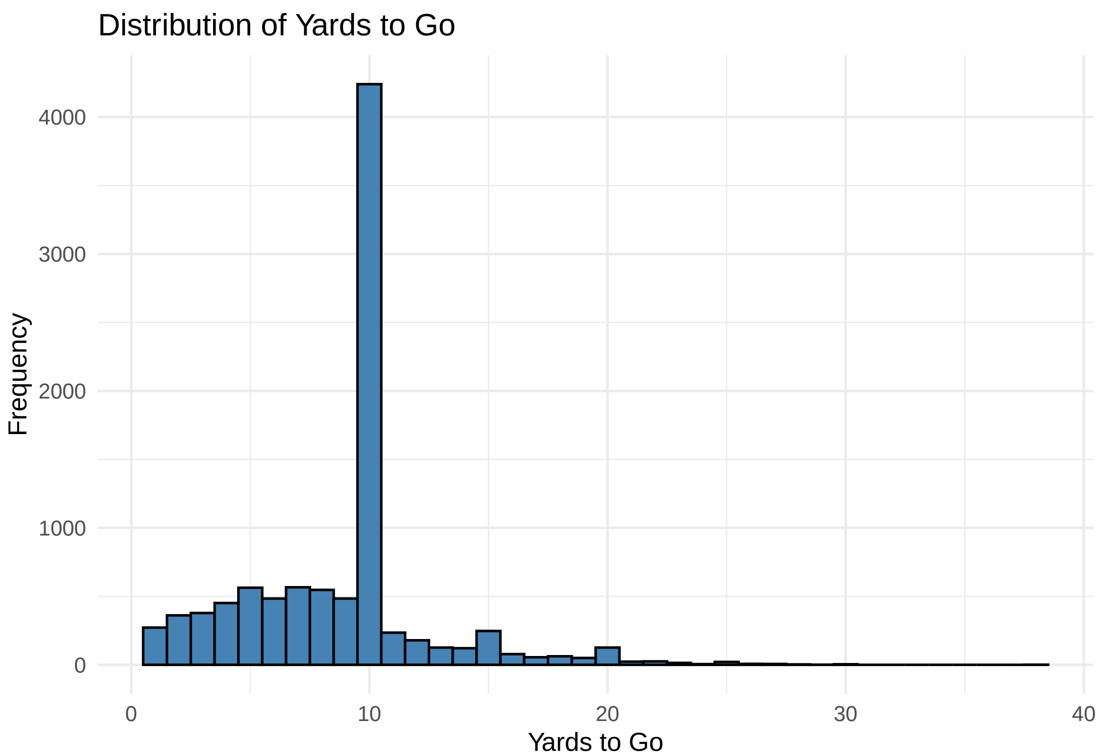
```r
plays_raw_data <- plays_raw_data %>%
  mutate(playClockCategory =
           case_when(playClockAtSnap >= 15 ~ "playClock More Than 15s",
                     playClockAtSnap >= 5 &
                     playClockAtSnap < 15 ~ "playClock Between 5s and 15s",
                     playClockAtSnap < 5 ~ "playClock Less Than 5s",
    )
  )

plays_filtered <- plays_raw_data %>%
  filter(!is.na(playClockCategory), !is.na(passResult))

ggplot(plays_filtered, aes(x = playClockCategory)) +
  geom_bar(fill = "steelblue", color = "black") +
  labs(
    title = "Frequency of Play Clock Categories",
    x = "Play Clock Category",
    y = "Count"
  ) +
  theme_minimal()
```



Based on the above graph, we see that the vast majority of the time the ball is snapped in the moderate play clock zone with between 5 and 15 seconds left on the play clock. This makes sense as most of the time teams want to make sure they have time to huddle up, relay the play to everyone and get set before snapping the ball. The second most frequent category is when the play clock is greater than 15 seconds. This category expresses when teams may be in a hurry-up offense where they might need to get down the field quicker or just want to get the defense on their heels. The least frequent category is when the play clock is less than 5 seconds. This category includes plays where teams may take a little longer to get set. This can be for a variety of reasons; teams may be sending players in motion before the snap or the quarterback may be making audibles or adjustments at the line of scrimmage. The significant parts that are relevant in the analysis below will be the more extreme buckets of less than 5 seconds and more than 15 seconds.

yardsToGo (Quantitative):

```
ggplot(plays_filtered, aes(x = yardsToGo)) +
  geom_histogram(binwidth = 1, fill = "steelblue", color = "black") +
  labs(
    title = "Distribution of Yards to Go",
    x = "Yards to Go",
    y = "Frequency"
  ) +
  theme_minimal()
```



Yards to Go seems to be slightly normally distributed with a significant outlier with the vast majority of plays being at 10 yards to go. This makes sense as every play defaults to 10 yards to go for a first down. The distribution is slightly right skewed with more plays with less than 10 yards to go.

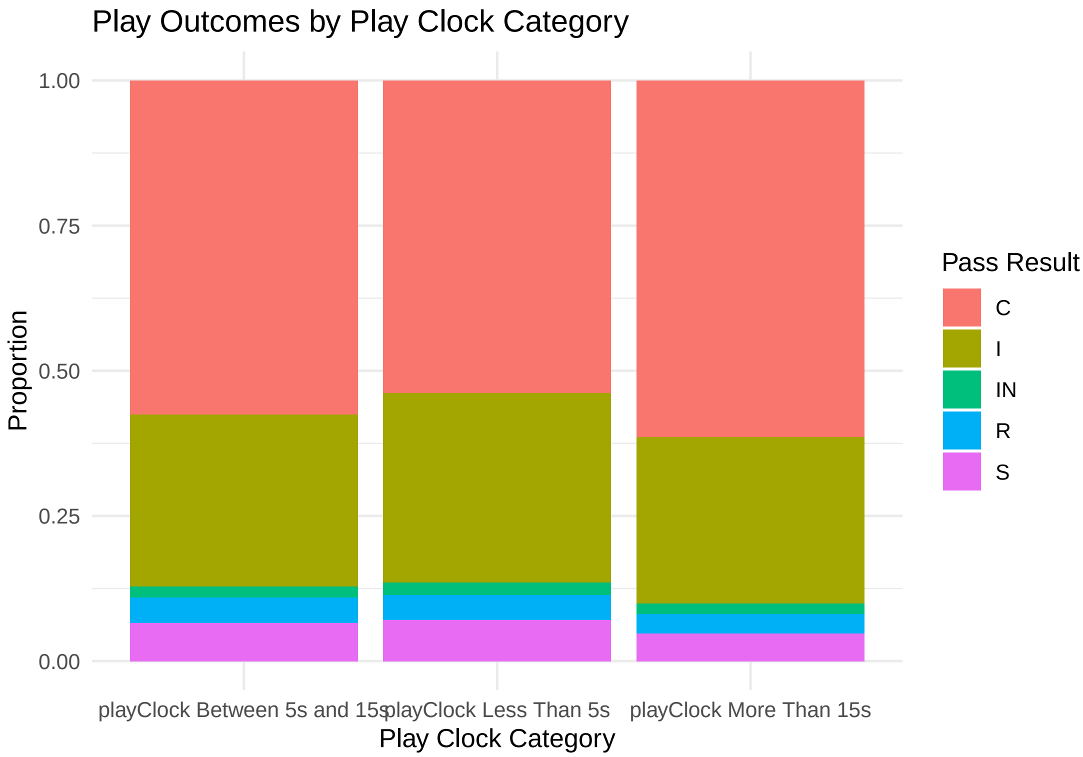
2.2.2 Relationships Between Variables

Before building predictive models, it's important to investigate potential relationships between variables that could impact play outcomes. In this section, we focus on the playClockAtSnap variable, which records the play clock value at the time of the snap, and explore its relationship with pass results and yards gained during a play.

The play clock is a critical element in football, as it dictates when a play must begin. A snap taken with a lower play clock may suggest hurried decision making, potentially leading to rushed plays or errors.

Below are some visuals and descriptions exploring the relationships between variables:

```r
ggplot(plays_filtered, aes(x = playClockCategory, fill = passResult)) +
  geom_bar(position = "fill") +
  labs(
    title = "Play Outcomes by Play Clock Category",
    x = "Play Clock Category",
    y = "Proportion",
    fill = "Pass Result"
  ) +
  theme_minimal()
```



We can see that when the playClock is less than or equal to 5s (lower), there is a more varied distribution of pass results, including incomplete passes, sacks, and interceptions. On the other hand (higher), it looks like high play clock snaps tend to result in more complete passes, potentially indicating that the defense had less time to organize and execute a well prepared play.

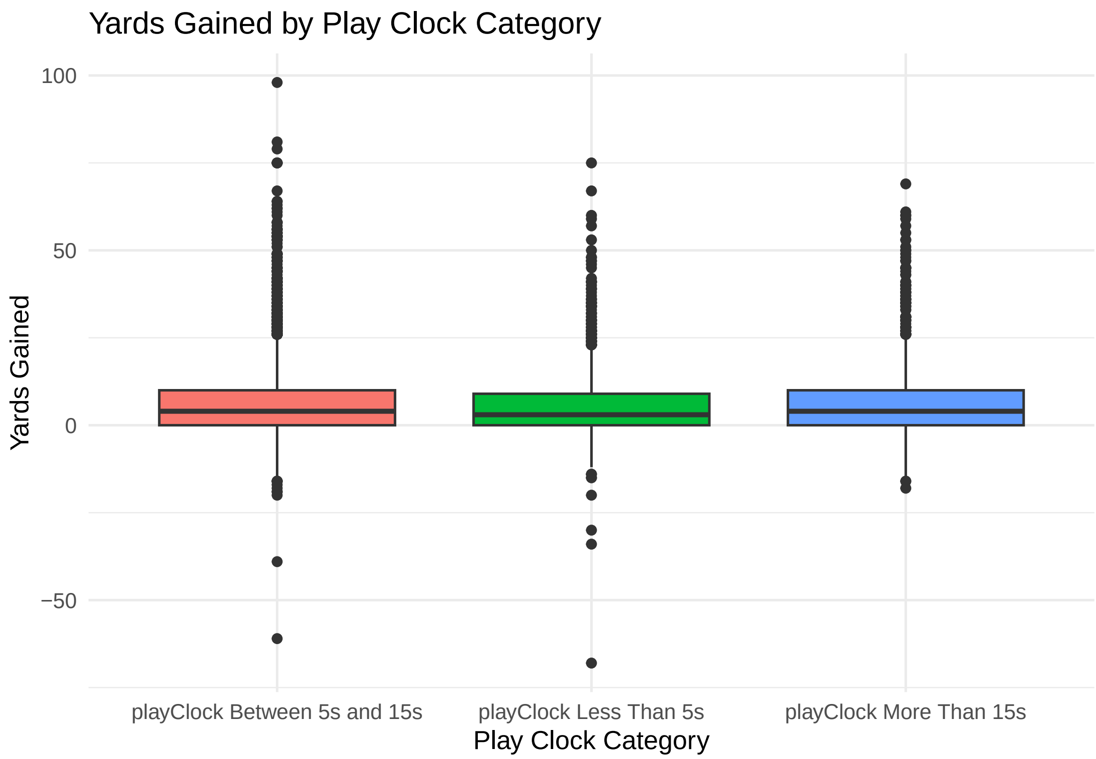
```
contingency_table <- table(plays_filtered$playClockCategory,
                           plays_filtered$passResult)

chi_sq_test <- chisq.test(contingency_table)
print(chi_sq_test)
```

```
##
##  Pearson's Chi-squared test
```

```
##
## data:  contingency_table
## X-squared = 29.539, df = 8, p-value = 0.0002549
```

And we can see here, the playClock is indeed statistically significant with a value of $0.0002549 < 0.05$. The next step would be to find a model to potentially predict results.

```
ggplot(plays_filtered, aes(x = playClockCategory, y = yardsGained,
                           fill = playClockCategory)) +
  geom_boxplot() +
  labs(
    title = "Yards Gained by Play Clock Category",
    x = "Play Clock Category",
    y = "Yards Gained"
  ) +
  theme_minimal() +
  theme(legend.position = "none")
```



We also investigated the relationship that the play block at snap might have with the yards gained that play. However, as we can see the above visualization, there is no clear correlation between the two. That being said, we do notice that there seems to be slightly more variation in yards gained when the play clock is less than 15 seconds compared to when it is more than 15 seconds.

```
# Summarize yardsGained by playClockCategory
plays_summary <- plays_filtered |>
  group_by(playClockCategory) |>
  summarize(
    mean_yards = mean(yardsGained, na.rm = TRUE),
    sd_yards = sd(yardsGained, na.rm = TRUE),   # Standard deviation
    min_yards = min(yardsGained, na.rm = TRUE),
    max_yards = max(yardsGained, na.rm = TRUE),
```

```
    n = n()
  )
print(plays_summary)
```

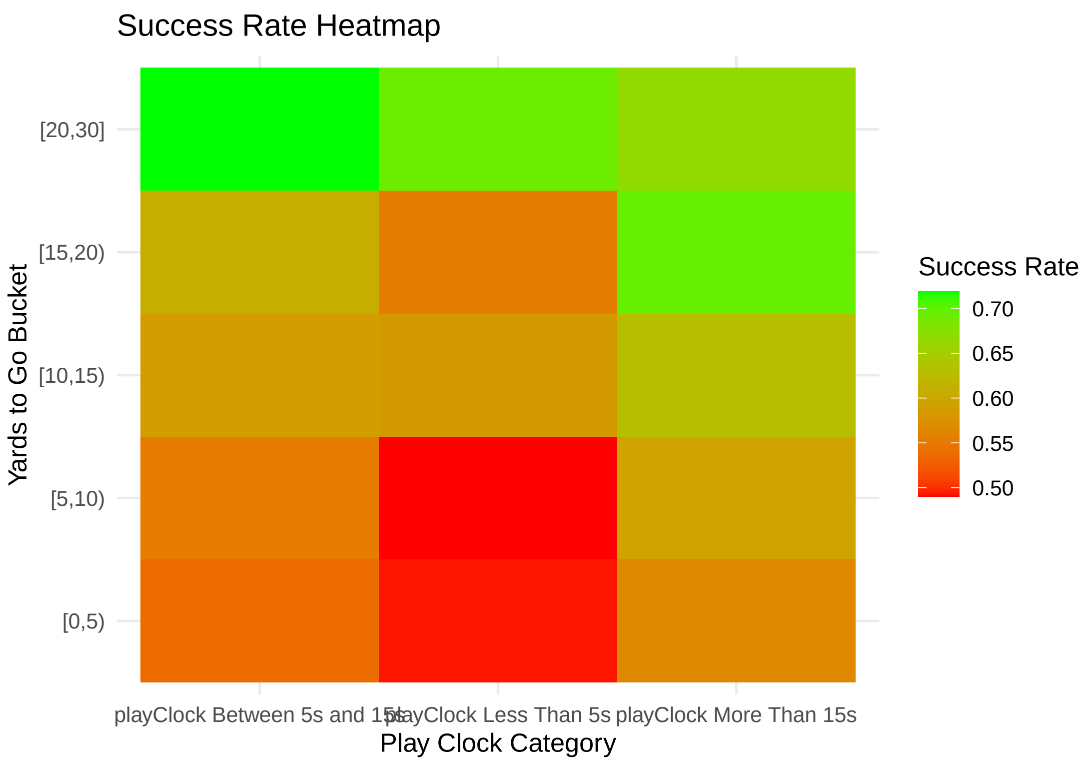
```
## # A tibble: 3 x 6
##   playClockCategory            mean_yards sd_yards min_yards max_yards     n
##   <chr>                             <dbl>    <dbl>     <dbl>     <dbl> <int>
## 1 playClock Between 5s and 15s       6.28     10.1       -61        98  5954
## 2 playClock Less Than 5s             5.63     9.78       -68        75  1670
## 3 playClock More Than 15s            6.43     9.55       -18        69  2111
```

This exploration confirms what we saw above, slightly more variation and range of yards gained when the play clock is lower.

Further exploration of how the play clock affects other variables yields the following heat map. This heat map illustrates the play success rate based on the play clock and yards to go during that play.

```
heatmap_data <- plays_filtered %>%
  filter(!is.na(playClockCategory), !is.na(yardsToGo), yardsToGo >= 0, yardsToGo <
  mutate(
    yardsToGoBucket = cut(
      yardsToGo, breaks = c(0, 5, 10, 15, 20, 30),
      right = FALSE, include.lowest = TRUE
    )
  ) %>%
  group_by(playClockCategory, yardsToGoBucket) %>%
  summarize(success_rate = mean(passResult == "C", na.rm = TRUE), .groups = "drop"
```

```
ggplot(heatmap_data, aes(x = playClockCategory, y = yardsToGoBucket, fill = succes
  geom_tile() +
  labs(
    title = "Success Rate Heatmap",
    x = "Play Clock Category",
    y = "Yards to Go Bucket",
    fill = "Success Rate"
  ) +
  theme_minimal() +
  scale_fill_gradient(low = "red", high = "green")
```



The heatmap above shows the success rate of plays based on whether the pass was complete (passResult == "C"), categorized by play clock and yards-to-go buckets. Notably, the highest success rate occurs when the play clock exceeds 15 seconds, especially in long-yardage situations (20 to 30 yards), where the success rate is over 70%. This suggests that more time on the play clock leads to better play execution, even when facing tougher distances.

Interestingly, short-yardage situations (0 to 10 yards) do not yield higher success rates. In fact, success rates in the [0, 5] and [5, 10] yard buckets are lower than longer-yardage plays, indicating that defensive anticipation may play a role. Additionally, when the play clock is less than 5 seconds, success rates drop across all yardage buckets. Overall, this heatmap shows a positive relationship between higher play clock times and play success, particularly in longer-yardage situations.

3 Modeling

3.1 Multinomial Model

Here for the predictive model, we implemented a multinomial regression model because the prediction outcome is nominal with multiple categories which is perfect for this kind of model. We removed the possibilities of R and S, scrambles and sacks, because the data did not record any results and were unused in our model. We discovered that playClockAtSnap, down, yardsToGo, passLength, pff_passCoverage, dropbackType are all correlated with passResult to some degree. The model predicts the pass result using these variables. To train the model, we have partitioned the data to a train_data and a test_data in which we will train the model on the train_data and test the model's success on the test_data. A common ratio for training and testing is 80 to 20 percent of the whole model which is used below. The code block below splits the plays_filtered data and trains the model named passResult_model. Below are the results:

```r
library(caret)
```

```
## Loading required package: lattice

##
## Attaching package: 'caret'

## The following object is masked from 'package:purrr':
##
##     lift
```

```r
library(nnet)

set.seed(123)

plays_filtered <- plays_filtered %>%
  filter(!passResult %in% c("R", "S"))

plays_filtered$passResult <- factor(plays_filtered$passResult)

train_index <- createDataPartition(plays_filtered$passResult, p = 0.80, list = FAL
train_data <- plays_filtered[train_index, ]
test_data <- plays_filtered[-train_index, ]

train_data$passResult <- droplevels(train_data$passResult)
test_data$passResult <- droplevels(test_data$passResult)

passResult_model <- multinom(
```

```
  passResult ~ playClockAtSnap + down + yardsToGo + passLength +
    pff_passCoverage + dropbackType,
  data = train_data
)
```

```
## # weights:  84 (54 variable)
## initial  value 7656.229040
## iter  10 value 5256.859878
## iter  20 value 5068.654223
## iter  30 value 4798.338396
## iter  40 value 4718.018172
## iter  50 value 4717.366550
## iter  60 value 4717.315123
## iter  70 value 4717.308401
## iter  70 value 4717.308359
## iter  70 value 4717.308359
## final  value 4717.308359
## converged
```

If we were to look at the summary of the mode, it would give a variety of statistics including coefficients and standard errors of all the different variables and relative subcategories. Instead of analyzing these statistics, we analyze the contingency table below that summarizes the success of the model on the test_data:

```
predictions <- predict(passResult_model, test_data)

predictions <- as.factor(predictions)
```

```r
library(caret)

conf_matrix <- confusionMatrix(predictions, as.factor(test_data$passResult))

print(conf_matrix)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction    C    I   IN
##         C  1025  400   27
##         I    99  173   11
##         IN    0    0    0
##
## Overall Statistics
##
##                Accuracy : 0.6905
##                  95% CI : (0.6681, 0.7122)
##     No Information Rate : 0.6478
##     P-Value [Acc > NIR] : 9.62e-05
##
##                   Kappa : 0.2338
##
##  Mcnemar's Test P-Value : < 2.2e-16
##
## Statistics by Class:
##
```

```
##                      Class: C Class: I Class: IN
## Sensitivity            0.9119  0.30192    0.0000
## Specificity            0.3011  0.90534    1.0000
## Pos Pred Value         0.7059  0.61131       NaN
## Neg Pred Value         0.6502  0.72452    0.9781
## Prevalence             0.6478  0.33026    0.0219
## Detection Rate         0.5908  0.09971    0.0000
## Detection Prevalence   0.8369  0.16311    0.0000
## Balanced Accuracy      0.6065  0.60363    0.5000
```

The results of the model are as follows from the confusion matrix: - 1025 plays were correctly predicted as Complete Passes.
- 173 plays were correctly predicted as Incomplete Passes. - 400 plays were misclassified as complete when they were actually incomplete.
- 27 plays were misclassified as complete when they were actually interceptions.
- 99 plays were misclassified as incomplete when they were actually complete.
- The model did failed to predict any results as I

When we look at the overall statistics we can see that the accuracy of the model is almost 70% at approximately 69.05% accuracy. This is an improvement to the No Information rate of 64.78% which would be the accuracy if the model predicted passResult purely off of chance. Furthermore, we can see the confidence interval says with 95% confidence that the true prediction accuracy is between 66.81% and and 71.22%.

3.2 Neural Network Model

Justin put your model under here:

Now we want to attempt to build a model that can help us find the optimal play clock time at which to snap the ball given the yards left to go, the down, the game clock and the offensive formation.

```
#filter out relevant variables for model
plays_filtered2 <- plays_raw_data |>
  filter(!is.na(playClockAtSnap), !is.na(yardsToGo), !is.na(gameClock), !is.na(dow
```

Below, we've constructed a model that predicts the play clock at the time of snap given the yards to go, the game clock time, the down and the offensive formation.

```
playClock_model <- lm(playClockAtSnap ~ yardsToGo + gameClock + down + offenseForm
                      data = plays_filtered2)


summary(playClock_model)
```

```
##
## Call:
## lm(formula = playClockAtSnap ~ yardsToGo + gameClock + down +
##     offenseFormation, data = plays_filtered2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -12.5545  -4.2186  -0.9358   3.1060  30.4877
##
## Coefficients:
```

```
##                                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)                    1.547e+01  2.872e-01  53.846  < 2e-16 ***
## yardsToGo                     -2.086e-01  1.343e-02 -15.535  < 2e-16 ***
## gameClock                     -1.887e-05  2.962e-06  -6.369 1.95e-10 ***
## down                          -1.837e+00  6.567e-02 -27.973  < 2e-16 ***
## offenseFormationI_FORM        -5.317e-01  2.547e-01  -2.088  0.03684 *
## offenseFormationJUMBO          1.706e+00  5.681e-01   3.003  0.00268 **
## offenseFormationPISTOL        -7.238e-01  2.933e-01  -2.468  0.01361 *
## offenseFormationSHOTGUN        9.732e-01  1.775e-01   5.482 4.27e-08 ***
## offenseFormationSINGLEBACK  4.437e-01  1.977e-01   2.245  0.02480 *
## offenseFormationWILDCAT       -1.784e-01  6.683e-01  -0.267  0.78946
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.02 on 15925 degrees of freedom
##   (188 observations deleted due to missingness)
## Multiple R-squared:  0.05768,    Adjusted R-squared:  0.05714
## F-statistic: 108.3 on 9 and 15925 DF,  p-value: < 2.2e-16
```

Given the summary statistics, the model is statistically significant with an overall p-value of $2.2*10^{-16}$. Additionally, every single variable we used to predict the play clock at snap was statistically significant at at least an alpha level of 0.05. It is also important to note that this model has an adjusted R-squared of 0.05714, which tells us that the 5.7% of the variation in the playClockAtSnap can be explained by this model.

```r
# Fit a linear model
win_prob_model <- lm(expectedPointsAdded ~ playClockAtSnap + yardsToGo + down +
                     gameClock + offenseFormation,
                     data = plays_filtered2)

# Summarize the model
summary(win_prob_model)
```

```
##
## Call:
## lm(formula = expectedPointsAdded ~ playClockAtSnap + yardsToGo +
##     down + gameClock + offenseFormation, data = plays_filtered2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -13.1436  -0.6265  -0.1683   0.7095   8.7907
##
## Coefficients:
##                             Estimate Std. Error t value Pr(>|t|)
## (Intercept)                1.466e-01  7.273e-02   2.016 0.043867 *
## playClockAtSnap            3.430e-04  1.846e-03   0.186 0.852570
## yardsToGo                 -1.164e-02  3.151e-03  -3.694 0.000222 ***
## down                      -6.635e-02  1.567e-02  -4.235  2.3e-05 ***
## gameClock                  7.559e-07  6.909e-07   1.094 0.273940
## offenseFormationI_FORM    -1.713e-02  5.933e-02  -0.289 0.772771
## offenseFormationJUMBO      2.534e-01  1.324e-01   1.914 0.055585 .
## offenseFormationPISTOL    -1.276e-02  6.834e-02  -0.187 0.851895
```

```
## offenseFormationSHOTGUN     3.625e-02  4.139e-02   0.876 0.381088
## offenseFormationSINGLEBACK -7.955e-03  4.605e-02  -0.173 0.862853
## offenseFormationWILDCAT     1.935e-01  1.557e-01   1.243 0.213807
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.402 on 15924 degrees of freedom
##   (188 observations deleted due to missingness)
## Multiple R-squared:  0.002117,   Adjusted R-squared:  0.00149
## F-statistic: 3.378 on 10 and 15924 DF,  p-value: 0.0002037
```

4 Discussion