An Analysis by Dak's Disciples

Predicting NFL Play Success

By: Aryan Neeli, Meet Shah, Abhi Atluri

STAT 4355.001 FALL 2025

Table of Contents

1. Introduction

American football is frequently described as a "game of inches." In the National Football League (NFL), the difference between a win and a loss often hinges on the efficiency of specific play calls in high-pressure situations. As fans and statisticians, we often hear the phrase "Any Given Sunday," which suggests that the outcome of a game is highly unpredictable and dependent on factors that are difficult to quantify, such as player morale or momentum. Our team, Dak's Disciples, wanted to investigate the statistical truth behind offensive production. We aimed to determine if the number of yards gained on a play is truly random or if it can be reliably predicted by the situational context of the game.

The primary goal of our analysis is to identify which pre-snap factors best predict the number of **yards gained** on an offensive play. By applying statistical regression techniques, we sought to quantify the impact of specific game situations. Specifically, does a "Pass" play yield more yards than a "Run" play when holding other factors constant? Does the distance to a first down change how a defense plays, thereby allowing for more or fewer yards to be gained?.

To achieve this, we utilized a comprehensive dataset of NFL play-by-play statistics from the 2009 to 2018 regular seasons. This dataset was originally compiled by Maksim Horowitz, Ron Yurko, and Sam Ventura and sourced from Kaggle. The raw data contained approximately 350,000 observations and over 100 variables, covering every single event that occurred in a decade of football games. This massive amount of data allowed us to isolate offensive plays and remove non-predictive events like penalties or timeouts, while also drawing a representative random sample to ensure robust and efficient analysis.

For our analysis, we selected **Yards Gained** as our continuous response variable. We hypothesized that this variable could be modeled using a set of situational predictors :

- **Down:** The current down (1st through 4th) the offense is facing.
- **Yards To Go:** The distance remaining to achieve a first down.

- **Score Differential:** The difference in points between the offense and the defense, which helps account for "garbage time" or desperation plays.
- **Play Type:** A categorical variable indicating whether the play was a "Run" or a "Pass."
- **Game Seconds Remaining:** A continuous variable representing the time pressure on the offense.

By analyzing these variables, we aim to provide statistical evidence regarding which game situations are most favorable for offensive production and to measure the extent to which variance in an NFL play is predictable through mathematical analysis versus pure athletic skill and randomness.

2. Data Description

2.1 Data Source and Overview

The dataset utilized for this project is the "Detailed NFL Play-by-Play Data 2009-2018," originally compiled by Maksim Horowitz, Ron Yurko, and Sam Ventura and sourced directly from Kaggle. This comprehensive dataset is comprised of approximately 356,768 individual observations and over 100 variables, capturing granular details of every play from a full decade of NFL regular seasons. It includes a wide array of information ranging from game context (time, score, down) to play specifics (pass location, run gap, penalties).

2.2 Data Cleaning and Preprocessing

Given the sheer volume and complexity of the raw data, we performed several rigorous cleaning and preprocessing steps to create a dataset suitable for our specific modeling goals.

First, we filtered the data to strictly isolate offensive efficiency. The original dataset contained special teams plays (punts, field goals), timeouts, and penalties, which are not relevant to our goal of measuring yardage production. We then subset the data to include only rows where the play type was explicitly labeled as "run" or "pass".

Second, we addressed the specific requirements of our dual modeling strategy (Linear vs. Logarithmic). Our initial exploratory analysis revealed that the response variable, yards gained, was heavily right-skewed. To address this, we planned to test a Log-Level regression model. Since logarithmic functions cannot handle zero or negative values, we filtered the dataset to include only plays with **positive yardage**. This strategic decision allowed us to focus our analysis specifically on successful offensive plays, treating sacks and tackles for loss as separate statistical phenomena driven by errors rather than play design.

Third, to ensure computational efficiency and prevent overplotting in our diagnostic graphs, we implemented a random sampling strategy. Attempting to plot and model over 300,000 data points resulted in significant overplotting and excessive computational time for variable selection algorithms. To resolve this, we took a random sample of 2,000 plays using a set seed to ensure reproducibility.

Finally, we performed feature engineering to prepare our variables for regression. We converted categorical variables such as "Down" and "Play Type" into factors to ensure they were treated as distinct categories rather than continuous numbers during regression analysis.

2.3. Variables Selected

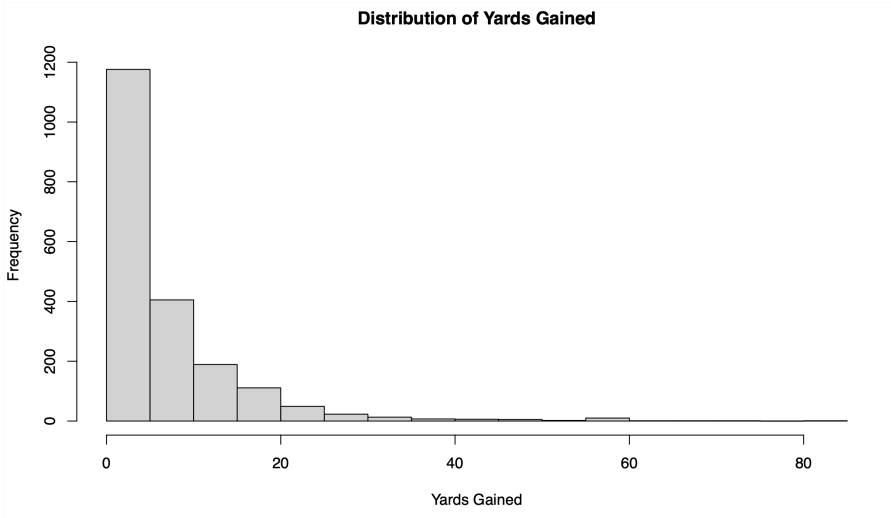
After cleaning, we retained the following seven variables for our analysis:

Variable Name	Type	Description
yards_gained	Continuous	The number of yards gained on the play (Response Variable).
down	Categorical	The current down (1st, 2nd, 3rd, or 4th).
ydstogo	Continuous	The distance (in yards) required for a first down.
qtr	Categorical	The quarter of the game (1 through 4).
score_differential	Continuous	The difference in score between the offensive and defensive team.
play_type	Categorical	The type of offensive play ("Run" or "Pass").
game_seconds_remaining	Continuous	The time remaining in the game (in seconds).

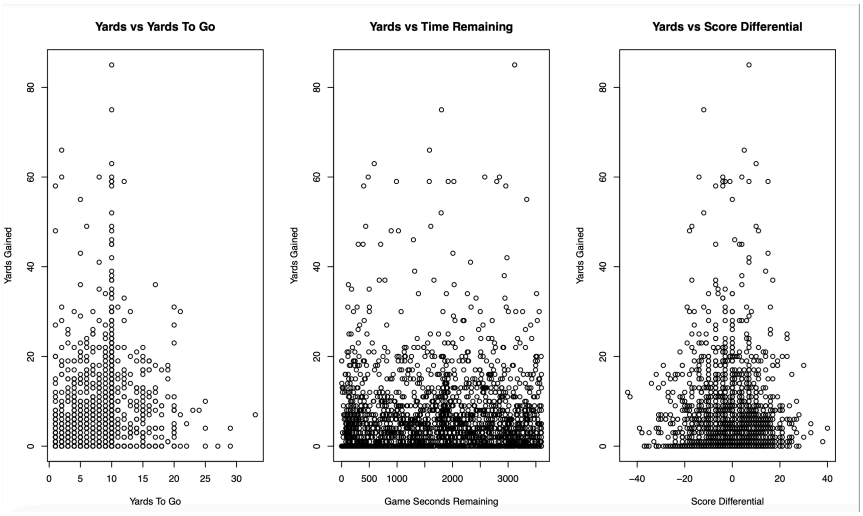
3. Exploratory Data Analysis

3.1 and 3.2 Response Variable Distribution + Predictors vs Response

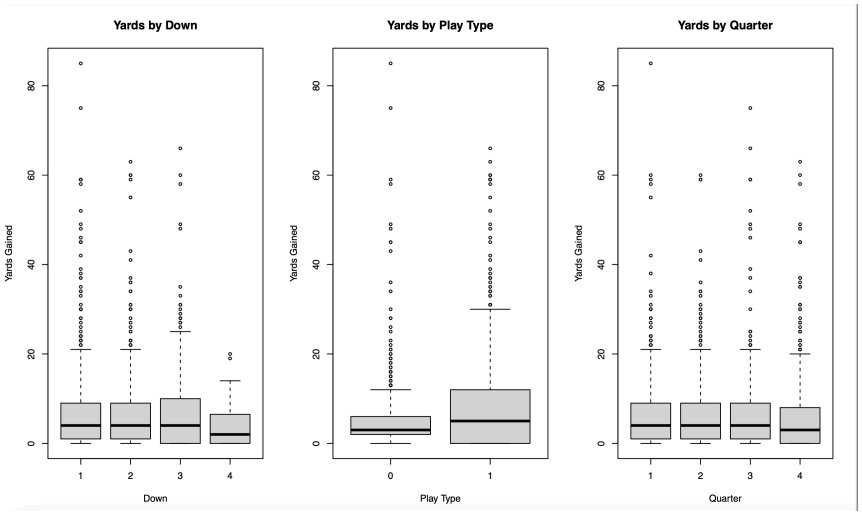
First, we looked at a histogram of yards_gained in order to understand its distribution.



The histogram showed us that yards_gained is very right skewed, with short gains dominating and few long gain plays, which makes sense in the context of football. To check the effect of the numeric predictors on yards, we then checked on scatterplots of the numeric predictors vs yards_gained.



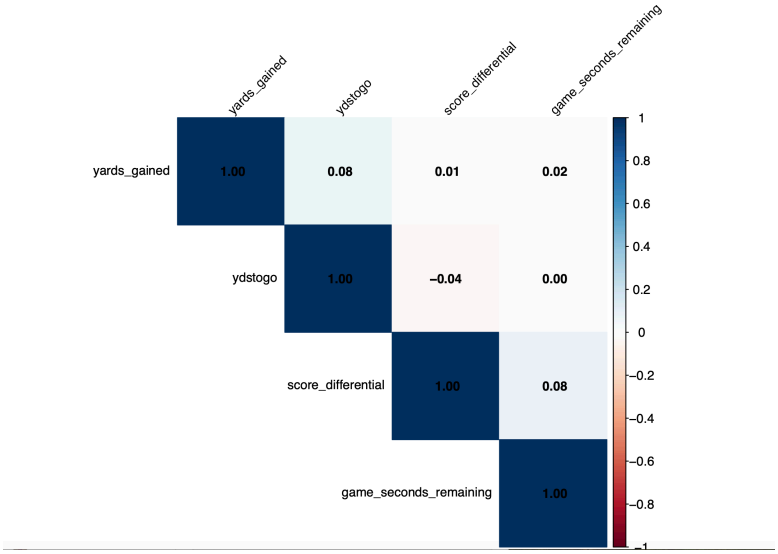
The scatterplots show that there is essentially no linear relationship between any of the numeric predictors and yards gained. The next logical step was to then check the effect the categorial variables had on yards gained, using boxplots.



The boxplots revealed that there was a slight decrease in yards_gained on 4th down and in the 4th quarter, but more importantly passing plays had a higher median yards gained than running plays.

3.3 Correlation Analysis

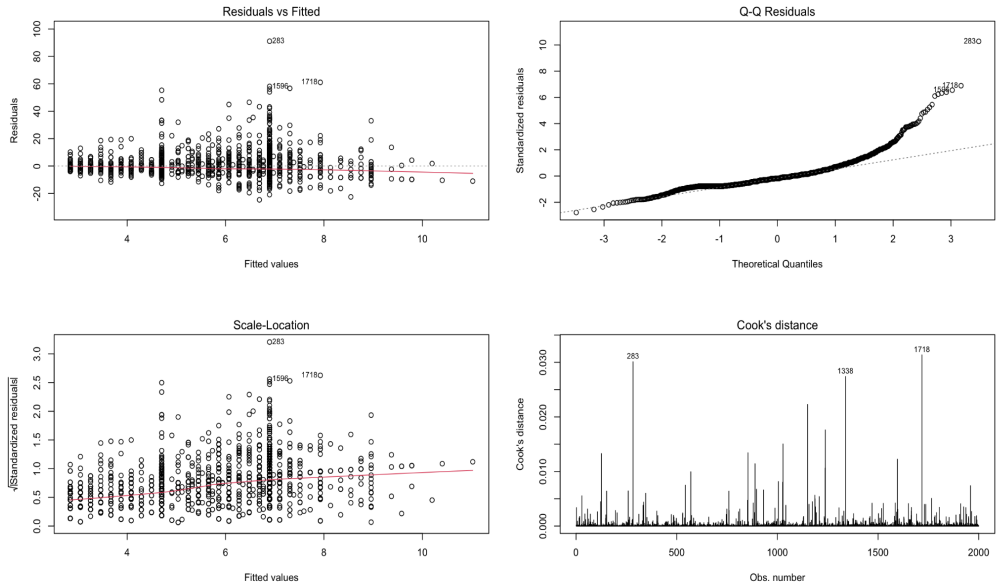
Finally, we checked the correlation between the numeric predictors in order to investigate multicollinearity, using a heatmap.



The heatmap revealed that the correlation between any of the numeric predictors was weak at best, which suggests no multicollinearity.

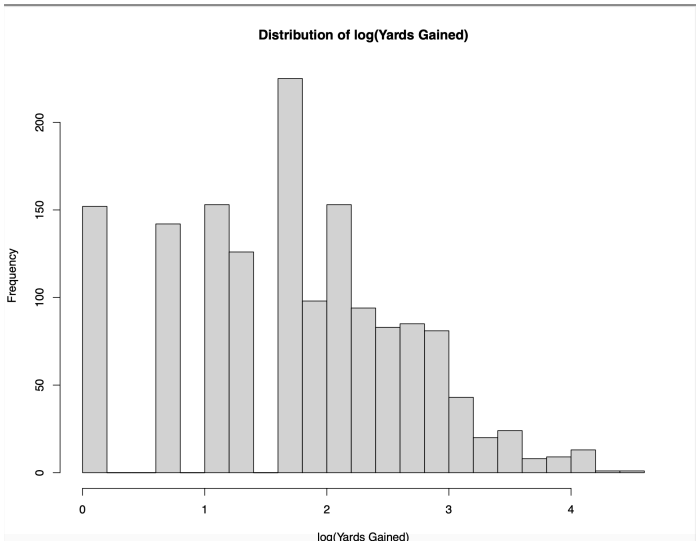
3.4 Motivation for Transformation

From the exploratory analysis, the response variable was revealed to be very right-skewed. This makes it difficult to assume normality and constant variance, both of which are required for linear regression. Also, the residual plots from the candiate linear model showed strong heteroscedasticity and tailed behavior.

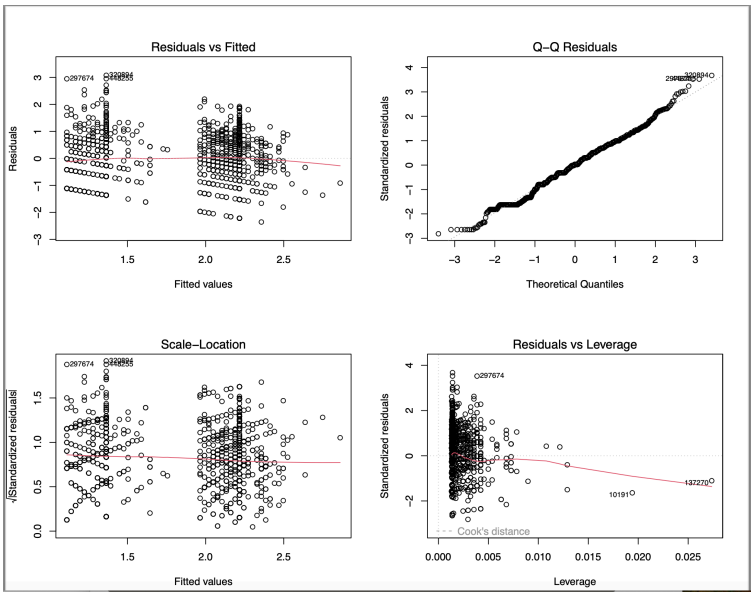


Due to these observations, we decided to observe a log model transformation of the response variable yards_gained to get better results. The candidate log model in this case was log(yards_gained) ~ ydstogo + play_type, which we derived from the EDA plots and making educated guesses as to what the most prominent predictor variables would be.

3.5 and 3.6 Log-Transformed Response Distribution + Log Model Residual Analysis



The log distribution of yards_gained in this candidate log model is far more symmetric than the original linear one, which hints that the variance is likely more stable and the skewedness has reduced. To ensure that the candidate log model is truly an improvement, we checked the residual plots of the log model as well.



Again, we can see an improvement from the linear model. The residual plots of the log model show far less heteroscedasticity than the linear one, and also improved normality. There is still some tailed behavior, the variance is far more stable in the fitted values, and it is evident that the log transformation is an improvement and more accurately reflects the assumptions needed for linear regression.

4. Regression Analysis

4.1 Initial Linear Model

Firstly, as discussed above in section 3.4, fitting a multiple linear regression model was deemed inappropriate due to the violation of normality and constant variance. Due to these faults, we decided on a log transformation to use as our final model

4.2 Log Transformation Model

To rectify the issues from the linear model, we decided to use a log transformation on the response variable. A logarithmic transformation cannot handle negative values, but we did not face this issue because we began our data cleaning by filtering out negative yardage. This transformation, of the

form log(yards_gained) ~ ydstogo+ play_type**.**, improved both distribution symmetry and also improved variance stability, as seen in the plots above in sections 3.5 and 3.6.

4.3 Variable Selection

We then need to perform variable selection on the full log model to ensure we select the best one. First, we checked for multicollinearity using GVIF.

```
                            GVIF Df GVIF^(1/(2*Df))
ydstogo                 1.140276  1        1.067837
down                    1.228039  3        1.034829
play_type               1.111900  1        1.054467
score_differential      1.044305  1        1.021913
game_seconds_remaining 15.578449  1        3.946954
qtr                    15.762407  3        1.583448
```

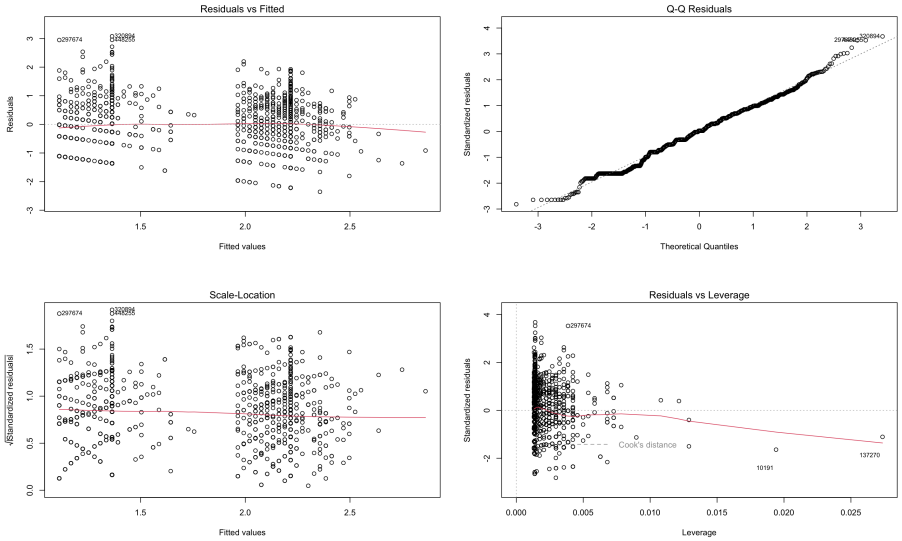
Due to high multicollinearity between qtr and game_seconds_remaining (which makes sense, if you know how many seconds are left in the game you know what quarter the game is in), we decided to remove qtr as game_seconds_remaning is a more detailed version of it.

Next, we used stepwise AIC to remove variables, starting from the full model. The process removed down, score_differential, and seconds _remaining; removing ydstogo or play_type increased AIC, so it deemed the final model to be: **log(yards_gained) ~ ydstogo + play_type**, which is the same as the model that we guessed initially from the EDA.

```
Start:  AIC=-517.22
log(yards_gained) ~ ydstogo + down + play_type + score_differential +
    game_seconds_remaining + qtr

                         Df Sum of Sq    RSS     AIC
- qtr                     3     0.839 1058.3 -522.03
- down                    3     1.032 1058.5 -521.75
- score_differential      1     0.054 1057.5 -519.15
- game_seconds_remaining  1     0.086 1057.6 -519.10
<none>                               1057.5 -517.22
- ydstogo                 1    16.717 1074.2 -495.52
- play_type               1   242.347 1299.8 -207.44

Step:  AIC=-522.03
log(yards_gained) ~ ydstogo + down + play_type + score_differential +
    game_seconds_remaining

                         Df Sum of Sq    RSS     AIC
- down                    3     1.016 1059.3 -526.58
- score_differential      1     0.070 1058.4 -523.93
- game_seconds_remaining  1     0.133 1058.5 -523.84
<none>                               1058.3 -522.03
+ qtr                     3     0.839 1057.5 -517.22
- ydstogo                 1    16.889 1075.2 -500.10
- play_type               1   243.464 1301.8 -211.17

Step:  AIC=-526.58
log(yards_gained) ~ ydstogo + play_type + score_differential +
    game_seconds_remaining

                         Df Sum of Sq    RSS     AIC
- score_differential      1     0.121 1059.5 -528.40
- game_seconds_remaining  1     0.161 1059.5 -528.35
<none>                               1059.3 -526.58
+ down                    3     1.016 1058.3 -522.03
+ qtr                     3     0.822 1058.5 -521.75
- ydstogo                 1    17.459 1076.8 -503.88
- play_type               1   267.514 1326.9 -188.36

Step:  AIC=-528.4
log(yards_gained) ~ ydstogo + play_type + game_seconds_remaining

                         Df Sum of Sq    RSS     AIC
- game_seconds_remaining  1     0.153 1059.6 -530.18
<none>                               1059.5 -528.40
+ score_differential      1     0.121 1059.3 -526.58
+ down                    3     1.067 1058.4 -523.93
+ qtr                     3     0.846 1058.6 -523.61
- ydstogo                 1    17.437 1076.9 -505.74
- play_type               1   273.378 1332.8 -183.55

Step:  AIC=-530.18
log(yards_gained) ~ ydstogo + play_type

                         Df Sum of Sq    RSS     AIC
<none>                               1059.6 -530.18
+ game_seconds_remaining  1     0.153 1059.5 -528.40
+ score_differential      1     0.114 1059.5 -528.35
+ down                    3     1.093 1058.5 -525.74
+ qtr                     3     0.926 1058.7 -525.51
- ydstogo                 1    17.345 1077.0 -507.65
- play_type               1   273.494 1333.1 -185.25
>
```

4.4 Model Diagnostics

We then looked at the final model's diagnostic plots. There was no strong nonlinear patterns in the residuals vs fitted values plot, which maintained linearity. The QQ plot showed slight tail deviations but still far less than the original linear model, enhancing normality. The scale location plot showed less heteroscedasticity than the linear one, and finally the cooks distance plot showed no overly high influential points. Overall the log transformed model was better suited for the regression assumptions compared to the linear one.



4.5 ANOVA and Model Significance

```
Call:
lm(formula = log(yards_gained) ~ ydstogo + play_type, data = nfl_pos)

Residuals:
     Min       1Q   Median       3Q      Max
-2.35669 -0.53693  0.02229  0.58191  3.07865

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 1.083973   0.056790  19.087  < 2e-16 ***
ydstogo     0.028003   0.005636   4.968 7.52e-07 ***
play_type1  0.852677   0.043220  19.729  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.8383 on 1508 degrees of freedom
Multiple R-squared:  0.2212,    Adjusted R-squared:  0.2201
F-statistic: 214.1 on 2 and 1508 DF,  p-value: < 2.2e-16
```

Overall, the final model is statistically significant, with an F value of 214.1 and p<2.2e-16. This tells us that the predictors explain a large portion of the variability of log(yards_gained). Also, the adjusted R^2 is .22, which is much higher than the linear model's adjusted R^2 of .02 with the same selected variables.

```
Analysis of Variance Table

Response: log(yards_gained)
            Df  Sum Sq Mean Sq F value    Pr(>F)
ydstogo      1   27.43  27.427  39.033 5.409e-10 ***
play_type    1  273.49 273.494 389.221 < 2.2e-16 ***
Residuals 1508 1059.63   0.703
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

The ANOVA table also shows that both ydstogo and play_type are both statistically significant predictors. Here, we notice that play_type has a much larger association with yards_gained than ydstogo (looking at their respective sum of squares values). Next,we use nested ANOVA to check if adding ydstogo improves the model and deserves to be there.

```
Analysis of Variance Table

Model 1: log(yards_gained) ~ play_type
Model 2: log(yards_gained) ~ ydstogo + play_type
  Res.Df    RSS Df Sum of Sq      F    Pr(>F)
1   1509 1077.0
2   1508 1059.6  1    17.345 24.684 7.522e-07 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

From this nested ANOVA table, we can see that adding ydstogo does improve the model fit compared to just having play_type (F value of 24.68, miniscule p-value).

4.6 Final Model Formula

Our final selected model is log(yards_gained) ~ ydstogo + play_type. Using the coefficient estimates from the summary, the model/our prediction for the estimated yardage of an NFL play is

log(yards_gained) = 1.0840 + 0.0280(ydstogo) + 0.8527(play_type)

Because the response is logged, the interpretations are multiplicative. This means that holding play_type constant, a 1 yard increase in ydstogomeans around a e^.0280-1= 2.8 percent increase in expected yards gained. Holding ydstogo constant, pass plays are expected to gain around e^.8527-1 = 135 percent more yards than run plays on average. The intercept is the expected log(yards_gained) for a run play when ydstogo is 0, which our model predicts at 1.0840.

5. Conclusion

5.1 The Predictive Power of the Logarithmic Model

Our investigation into offensive production revealed a critical insight: the relationship between game context and yardage is fundamentally non-linear. Initially, our standard multiple linear regression model yielded a very low Adjusted R-squared of approximately 0.02. This suggested that situational variables like Down and Distance explained almost none of the variation in play outcomes. However, by recognizing the heavy right-skew of the data and filtering for positive yardage, we were able to implement a Log-Level regression model.

This methodological pivot resulted in a significant improvement in our model's explanatory power, raising the Adjusted R-squared to approximately **0.22**. This jump indicates that while football plays may look chaotic on a linear scale, they follow a predictable logarithmic decay pattern. Specifically, our model confirms that **Play Type** is the dominant predictor of success. Even when accounting for the logarithmic scale, passing plays consistently offer a higher ceiling for yardage gain compared to running plays, statistically validating the modern NFL's shift toward air-raid and pass-heavy offensive schemes.

5.2 Quantifying the "Any Given Sunday" Factor

Despite the success of our Logarithmic model, roughly 78% of the variance in play outcomes remains unexplained. While an R-squared of 0.22 is a significant achievement for high-variance human behavioral data, the remaining unexplained variance statistically quantifies the "Any Given Sunday" adage.

Our analysis proves that while situational math accounts for roughly 22% of a play's result, the remaining 78% is dictated by factors that are impossible to capture in a play-by-play spreadsheet. These include:

- **Human Execution:** A dropped pass, a missed block, or a slipped tackle can instantly alter the outcome of a play regardless of the pre-snap math.
- **Player Tier:** Our model treats all players equally, but the probability distribution of a pass by an elite quarterback is fundamentally different from that of a backup.
- **Physical Chaos:** The non-normal distribution of our residuals, even after transformation, highlights that explosive plays are statistical outliers driven by athletic brilliance rather than situational logic.

5.3 Final Verdict

To place our results in context, it is important to note that even sophisticated proprietary models used in sports betting rarely exceed 55% to 60% accuracy in predicting outcomes. By achieving an R-squared of 0.22 on play-by-play data, our project demonstrates that offensive production is not purely random. There is a clear, mathematical structure to how yards are gained, driven largely by the choice to pass and the distance to the first down marker. However, the ceiling of predictability is limited by the inherent chaos of the sport. Football is a game where math provides the strategy, but human execution determines the result.

---

6. Reflection

6.1 Successes

Our most significant success in this project was our adaptability. Our initial attempt to model the data using standard linear regression yielded a poor Adjusted R-squared of 0.02, which suggested our model had almost zero predictive power. Instead of accepting this result, we critically analyzed our residuals and identified that the right-skewed nature of the data was the problem. By filtering out negative yardage plays (sacks and losses) to focus on successful offensive execution, we were able to implement a Log-Level model. This strategic pivot improved our R-squared to 0.22, a ten-fold increase that salvaged the analytical value of our project.

We also effectively managed the technical challenges of "Big Data." The raw dataset contained over 350,000 observations, which initially caused our diagnostic graphs to look like unreadable black blobs. We implemented a random sampling strategy to reduce the noise while maintaining statistical validity. Furthermore, we successfully identified logical redundancies in our predictors, such as the multicollinearity between "Quarter" and "Time Remaining," and removed variables that were cluttering the model without adding value.

6.2 Challenges and Limitations

Despite our successes, this project highlighted several difficulties inherent in modeling human behavior and sports.

First, we realized that we may have asked the wrong research question. We tried to predict the exact continuous value of "Yards Gained," which is incredibly difficult given the variance of the sport. In hindsight, a more effective approach might have been to simplify the problem into a binary question: "Did the offense get a First Down? Yes or No." Using Logistic Regression to predict this binary outcome would likely have been more robust than trying to predict if a play would go for exactly 6 yards versus 7 yards.

Second, our data was messy and incomplete regarding the "human element." Our model treated every "Pass" play exactly the same, but in reality, a pass thrown by a star quarterback has a completely different probability of success than one thrown by a backup. We lacked variables for "Star Power," player skill, or even weather conditions, all of which are massive factors in the real world.

Finally, even after filtering for outliers, the spread of the data remained massive. We learned that in a high-variance system like football, the difference between a 0-yard gain and a 50-yard gain often comes down to a single split-second mistake by a defender. No amount of pre-snap situational math can account for that chaos, which serves as a humbling reminder of the limitations of statistics in sports.

---

7. Appendix

7.1 References

1. Horowitz, M., Yurko, R., & Ventura, S. (2019). *NFL Play-by-Play Data 2009-2018*. Kaggle.
2. R Core Team (2023). *R: A language and environment for statistical computing*. R Foundation for Statistical Computing, Vienna, Austria.
3. Fox, J., & Weisberg, S. (2019). *An R Companion to Applied Regression*, Third Edition. Sage.

7.2 Code

```
#
================================================================================
========

# 1. SETUP & DATA CLEANING

#
================================================================================
========

# Load necessary libraries

library(corrplot)
```

```
library(car)


# Load raw data
nfl <- read.csv("NFLData.csv")


# Filter to include only offensive plays (run and pass)
nfl_clean <- subset(nfl, play_type %in% c("run", "pass"))


# Select relevant variables
nfl_model <- nfl_clean[, c("yards_gained", "down", "ydstogo", "qtr",
                           "score_differential", "play_type", "game_seconds_remaining")]


# Remove rows with missing data
nfl_model <- na.omit(nfl_model)


# Filter for valid football values and non-negative yards
# Note: Keeping 0 yards for initial EDA, but will filter >0 for Log Model later
nfl_model <- subset(nfl_model,
                    down %in% 1:4 &
                    qtr %in% 1:4 &
                    ydstogo >= 0 &
                    game_seconds_remaining >= 0 &
                    yards_gained >= 0 & yards_gained <= 100)
```

```
# Random Sampling for computational efficiency (n = 2000)
set.seed(123)
nfl_model <- nfl_model[sample(nrow(nfl_model), 2000), ]

# Feature Engineering: Convert categorical variables to factors
nfl_model$down <- as.factor(nfl_model$down)
nfl_model$qtr <- as.factor(nfl_model$qtr)
nfl_model$play_type <- as.factor(ifelse(nfl_model$play_type == "pass", 1, 0))

# Save cleaned dataset
write.csv(nfl_model, "nfl_model_cleaned.csv", row.names = FALSE)

#
================================================================================
========
# 2. EXPLORATORY DATA ANALYSIS (EDA)
#
================================================================================
========
# Generate PDF report of all EDA plots
pdf("EDA_plots_full.pdf", width = 10, height = 6)

# A. Univariate Analysis: Distribution of Yards Gained
```

```
hist(nfl_model$yards_gained,
     main = "Distribution of Yards Gained",
     xlab = "Yards Gained",
     breaks = 30)

# B. Scatterplots (Numeric Predictors)
par(mfrow = c(1,3))
plot(nfl_model$ydstogo, nfl_model$yards_gained,
     main = "Yards vs Yards To Go", xlab = "Yards To Go", ylab = "Yards Gained")
plot(nfl_model$game_seconds_remaining, nfl_model$yards_gained,
     main = "Yards vs Time Remaining", xlab = "Seconds Remaining", ylab = "Yards Gained")
plot(nfl_model$score_differential, nfl_model$yards_gained,
     main = "Yards vs Score Differential", xlab = "Score Differential", ylab = "Yards Gained")
par(mfrow = c(1,1))

# C. Boxplots (Categorical Predictors)
par(mfrow = c(1,3))
boxplot(yards_gained ~ down, data = nfl_model, main = "Yards by Down")
boxplot(yards_gained ~ play_type, data = nfl_model, main = "Yards by Play Type")
boxplot(yards_gained ~ qtr, data = nfl_model, main = "Yards by Quarter")
par(mfrow = c(1,1))

# D. Correlation Heatmap (Numeric Only)
```

```
num_vars <- nfl_model[, c("yards_gained", "ydstogo",
                          "score_differential", "game_seconds_remaining")]
cor_matrix <- cor(num_vars)
corrplot(cor_matrix, method = "color", type = "upper",
         tl.col = "black", addCoef.col = "black")


dev.off()


#
================================================================================
========
# 3. INITIAL MODELING: LINEAR REGRESSION
#
================================================================================
========
# Fit Full Linear Model
fit_linear <- lm(yards_gained ~ ydstogo + down + play_type +
                 score_differential + game_seconds_remaining + qtr,
                 data = nfl_model)


summary(fit_linear)


# Diagnostics for Linear Model
par(mfrow = c(2,2))
```

```
plot(fit_linear)

par(mfrow = c(1,1))


#
================================================================================
========

# 4. REFINED MODELING: LOG-LEVEL REGRESSION

#
================================================================================
========

# Filter for strictly positive yards to allow for Log transformation

nfl_pos <- subset(nfl_model, yards_gained > 0)


# Fit Full Log Model

fit_log <- lm(log(yards_gained) ~ ydstogo + down + play_type +

              score_differential + game_seconds_remaining + qtr,

              data = nfl_pos)


summary(fit_log)


# Generate PDF for Log Model Diagnostics

pdf("log_model_diagnostics.pdf", width = 10, height = 8)


# Distribution of Transformed Response
```

```
hist(log(nfl_pos$yards_gained),
     main = "Distribution of log(Yards Gained)",
     xlab = "log(Yards Gained)")

# Diagnostic Plots
par(mfrow = c(2,2))
plot(fit_log)
par(mfrow = c(1,1))

dev.off()

#
================================================================================
========
# 5. VARIABLE SELECTION & VALIDATION
#
================================================================================
========
# Check for Multicollinearity (GVIF)
gvif_vals <- vif(fit_log)
gvif_scaled <- gvif_vals
# Calculate scaled GVIF for interpretability
gvif_scaled[, "GVIF^(1/(2*Df))"] <- gvif_vals[, "GVIF"]^(1 / (2 * gvif_vals[, "Df"]))
print(gvif_scaled)
```

```
# Stepwise Selection (AIC)
fit_log_step <- step(fit_log, direction = "both", trace = FALSE)
summary(fit_log_step)

# Final Reduced Model (Based on Stepwise Results)
# Best model retains ydstogo and play_type
fit_log_final <- lm(log(yards_gained) ~ ydstogo + play_type, data = nfl_pos)
summary(fit_log_final)

# Sequential ANOVA
anova(fit_log_final)

# Nested ANOVA Test (Comparing Reduced vs. Full)
fit_log_reduced <- lm(log(yards_gained) ~ play_type, data = nfl_pos)
anova(fit_log_reduced, fit_log_final)
```