NFL 2015: A Statistical Analysis

Dave Dyer, Jostein Barry-Straume, Bryan Cikatz and Robert Flamenbaum

November 28, 2017

Abstract

This study investigates how average and variance of yards gained impacts a given team's ability to get touchdowns in the NFL. The theory is that teams with high average yards gained, but with low variance, would get more touchdowns than those with higher average yards gained, but higher variance. In other words, consistency is more important than big plays.

Related Work

There are two related works that also use NFL play-by-play data to ascertain trends in yardage and touchdowns. The first, "Underrated NFL Stats" [@Iyer] looks into the big plays allowed by the defense and the teams with the best / worst big plays allowed stats. The second, "How to Quantify the NFL" [@Kelly] looks at big play differentials, but doesn't actually do any statistical analysis. Our analysis focuses on the variance of the yardage per play numbers and measures the efficacy of teams by measuring the touchdowns from each game & team.

Introduction

The data are the 2015 play-by-play records available at Kaggle (https://www.kaggle.com/maxhorowitz/nflplaybyplay2015). These data are comprised of 46,129 rows describing every NFL play run over the 2015 season, by 32 teams. There are 65 columns that include multiple values, including text description of the play / penalty, names of players , touchdown boolean flags, down markers, timestamps, game ids, and yardages. For the purposes of this study, we focused on the average and variance of yards per play, by team, and measured the 'success' of the team based on the count of touchdowns per game.

The Tools

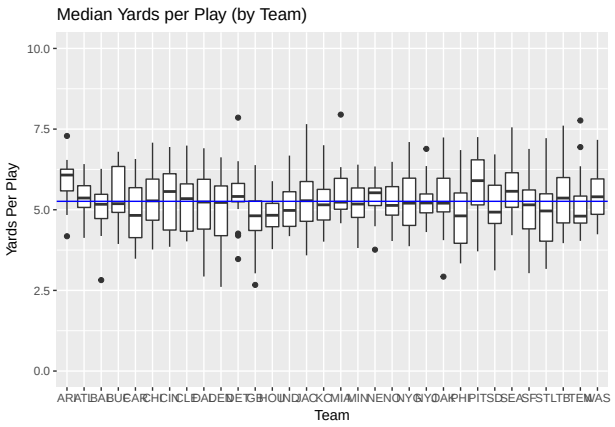
For this analysis, we use MongoDB for data mining and basic statistics, and R for data visualization and Markdown. We originally had planned on using MySQL on bluemix, but abandoned it in favor of Mongo after some serious issues whist loading the data.

We used Mongodb hosted on IBM Bluemix and Rstudio with Rmarkdown for the analysis, visualization and writeup. We used many r packages to support our analysis and data mining. For data mining, we used the mongolite package (https://jeroen.github.io/mongolite/) to connect to the bluemix data set and, where possible, we used mongo to do the statistical analysis in favor of R. Below are the specifications for the Mongo Database:

Parameter	Specification
Bluemix Storage	1GB
Data Size	14.3MB
Database Server	Compose for MongoDB-jj
Database Version	3.2.11
Database Location	US South
Cloud Hosting Service	IBM Bluemix
Processors	1 x 2.0 GHz Cores
Memory	1GB RAM

Analysis

When the spread of per-team yards-per-play is viewed in a team-specific box & whisker plot, it is easy to see that there is not a great amount of variance between the medians; most teams are between 5 and 6 median yards per play, with handful of teams above and below this range.

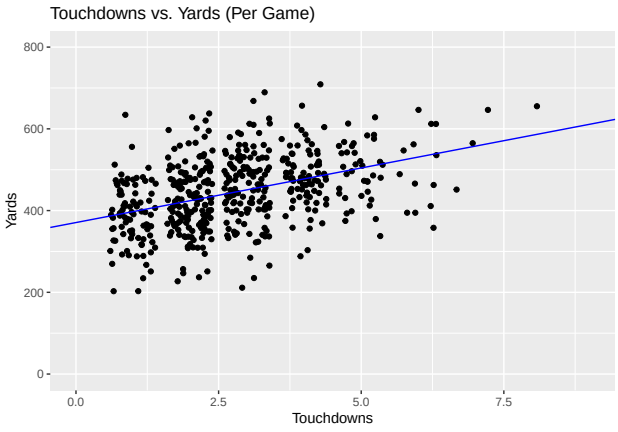


This is about what we'd expect, since the NFL is incredibly competitive, and self-normalizing due to constant trading, strategy, talent balance, and strategic positional matchups that, on balance, make the league pretty competitive.

If one team were overl dominant in median yards per play, we would expect that team to dominate the entire season, year after year, which is not the case. However, there *is* quite a bit of difference in the outer quartiles and in the number of outliers. These data support our investigation into whether or not a large variance changes team performance.

Yards v. Touchdowns

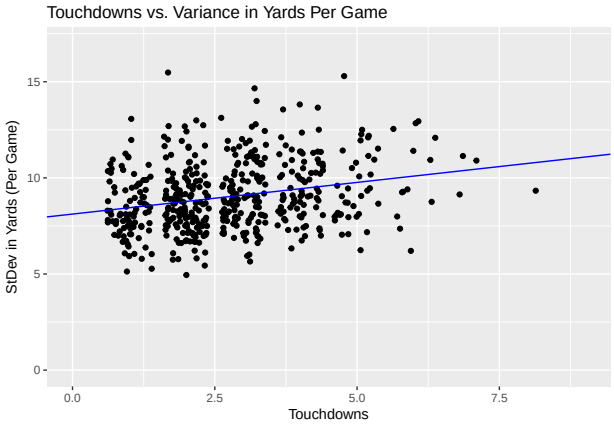
To start, we decided to plot number of touchdowns against yards gained (per game). We expect to see a linear trend upwards that illustrates the basic football concept of more yards = more points. As you can see, there appears to be an upward trend – that is, as yards increase, so does the number of touchdowns.



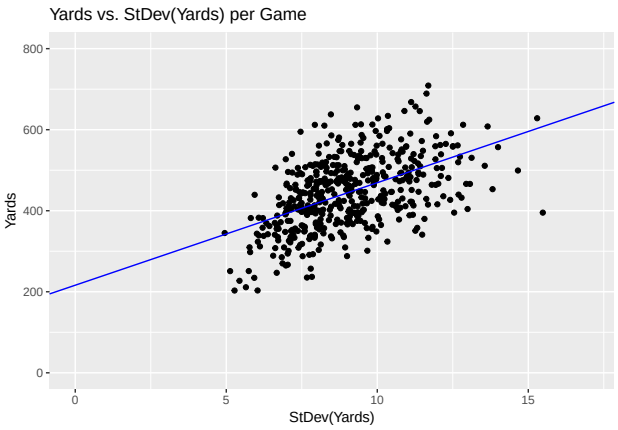
However, the theory of this paper is that, if yards per play is normalized, the lower variance teams will get more touchdowns per game. So it is important to know how the non-normalized standard deviation of yards per play relates to touchdowns per game on average.

Std Dev v. Touchdowns

The below plot shows that it is relatively static, with perhaps a slight rise as standard deviations go up.



However, the data are not normalized, and in order to understand the impact a normalization routine would have on the data, we can examine the relationship between the standard deviation of yards per play and the total yards per game. We expect that, as yards per game goes up, so does total yards, but that is not what's interesting about this plot. Note that it's not uniform linear growth as the standard deviation and the yards per game increase. The spread increases as yards and standard deviation increase, indicating that you get more yards with more variance, but you also run the risk of not getting as many yards (presumably because you're taking more chances, and you get higher rewards for higher risk... sometimes.)



Conclusion

The positive slope of the fit line in the StDev vs. Yards per Game plot is indicative that the surprising result in figure 3 – that is, that higher standard deviation equals more touchdowns – could be explained by this strong increasing trend. In order to find out for sure, we need a way to remove the impact of Yards / Game on Touchdowns / Game. Unfortunately, such an exercise is outside of the scope of this class.

Appendices

Below are the R code chunks used for this markdown.

```
# Returns all plays that scored a touchdown w
# grouping by team and count the plays per te
ydsGainTD <- m$aggregate('[
        { "$match": { "Yards.Gained": { "$gt"
    { "$group": { "_id": "$posteam" , "count"
    { "$sort": { "_id": 1} }
  ]')
ydsGainTD.tibble <- ydsGainTD

# Returns average yards gained and average nu
avgYdsTds <- m$aggregate('[
  { "$group": {
      "_id": "$posteam",
      "ydsGainAvg": { "$avg": "$Yards.Gained"
      "ydsGainSd": { "$stdDevPop": "$Yards.Ga
      "ydsGainSdSample": { "$stdDevSamp": "$Y
      "tdsAvg": { "$avg": "$Touchdown" },
      "tdsSum": { "$sum": "$Touchdown" }
      }
  }]')
avgYdsTds.tibble <- avgYdsTds$`_id`

# Returns sum of touchdowns and sum of penalt
sumTdsPenYds <- m$aggregate('[
  { "$group": {
      "_id": { "posteam": "$posteam"},
      "Touchdown": { "$sum": "$Touchdown" },
      "Penalty_Yards": { "$sum": "$Penalty.Ya
    }
  }]')
sumTdsPenYds.tibble <- sumTdsPenYds$`_id`

# Returns sum of touchdowns by quarter per te
sumTdsPerQtr <- m$aggregate('[
  { "$group": {
    "_id": {"Team": "$posteam", "Quarter": "$
    "Touchdown": { "$sum": "$Touchdown" }
    }
  }]')
sumTdsPerQtr.tibble <- sumTdsPerQtr$`_id`

ydsGainVTd <- m$aggregate('[
  { "$group": {
      "_id": { "GameID": "$GameID", "PosTeam"
      "SumTDs": { "$sum": "$Touchdown"},
      "SumYdsGain": { "$sum": "$Yards.Gained"},
      "AvgYdsGain": { "$avg": "Yards.Gained"},
      "SDYdsGain": { "$stdDevPop": "$Yards.Gained"}
  }}]')
ydsGainVTd <- ydsGainVTd[ydsGainVTd$SumTDs!=0,]
```

```
library(ggplot2)
bygame <- alldata %>%
  group_by_at(vars(posteam, GameID)) %>%
  summarize(TDgameSum = sum(Touchdown),
            avgYdsPerGamePerTeam = mean(Yards$Gained),
            sdYdsPerGamePerTeam = sd(Yards$Gained))
bp1 <- ggplot(data = bygame,
              aes(posteam,
                  x = posteam,
                  y = avgYdsPerGamePerTeam))
bp1 + geom_boxplot() + ylab("Yards Per Play") + xlab("Team
ggtitle("Median Yards per Play (by Team)") +
geom_hline(yintercept = median(bygame$avgYdsPerGamePerTeam
# Added horizontal line for the median of avgYdsPerGamePer

# Calculate intercept and slope for AB Line
coefs2 <- coef(lm(SumYdsGain ~ SumTDs, data = ydsGainVTd))
p2 <- ggplot(ydsGainVTd, aes(SumTDs, SumYdsGain))
p2 + geom_jitter() +
  xlab("Touchdowns") +
  ylab("Yards") +
  ggtitle("Touchdowns vs. Yards (Per Game)") +
  geom_abline(intercept = coefs2[1], slope = coefs2[2], co
  scale_x_continuous(limits=c(0,9)) +
  scale_y_continuous(limits=c(0, 800))

# Calculate intercept and slope for AB Line
coefs3 <- coef(lm(SDYdsGain ~ SumTDs, data = ydsGainVTd))
ggplot(ydsGainVTd, aes(SumTDs, SDYdsGain, group = "_id.Gam
  geom_jitter() +
  xlab("Touchdowns") +
  ylab("StDev in Yards (Per Game)") +
  ggtitle("Touchdowns vs. Variance in Yards Per Game") +
  geom_abline(intercept = coefs3[1], slope = coefs3[2], co
  scale_x_continuous(limits=c(0,9)) +
  scale_y_continuous(limits=c(0, 17))

# Calculate intercept and slope for AB Line
coefs4 <- coef(lm(SumYdsGain ~ SDYdsGain, data = ydsGainVT
ggplot(ydsGainVTd, aes(SDYdsGain, SumYdsGain, group = "_id
  geom_jitter() +
  ylab("Yards") +
  xlab("StDev(Yards)") +
  ggtitle("Yards vs. StDev(Yards) per Game") +
  geom_abline(intercept = coefs4[1], slope = coefs4[2], co
  scale_x_continuous(limits=c(0,17)) +
  scale_y_continuous(limits=c(0, 800))
```

References

Insert references here.

```
#r_refs(file = "r-references.bib")
```