assignment\_9\_CurryAdam

Adam Curry

8/2/2019

### How to import and clean my data

There are three combine datasets with several columns. Several of the columns within the three datasets are unnecessary. Some don’t apply to the analysis I will conduct, and some contain incomplete information. The variables I chose to include combine the three datasets (nfl\_stadiums.csv,spreadspoke\_scores.csv, nfl\_teams.csv’), perform some minor aggregation methods, and are specific to the winter months of the football season. I utilized sqldf library to manipulate the data in SQL syntax, as there were several joins, alterations, and aggregation methods that needed to be applied to the data.

# import the libraries used to clean the data  
library(sqldf)  
library(tidyverse)  
  
# set the working directory and import all three files  
setwd("C:\\Users\\adamp\\OneDrive\\Desktop\\a\_StatsR\\Final")  
df\_stadiums <- read.csv('nfl\_stadiums.csv')  
spreadspoke\_scores <- read.csv('spreadspoke\_scores.csv')  
df\_teams <- read.csv('nfl\_teams.csv')  
  
df\_stadiums <- sqldf("SELECT DISTINCT  
 stadium\_name  
 ,stadium\_weather\_type  
 ,ELEVATION  
 ,LATITUDE   
 ,LONGITUDE  
 ,stadium\_open  
 FROM  
 df\_stadiums  
 ")  
# remove white space and special characters from stadium names to join further downstream   
df\_stadiums$stad\_pkey <- str\_replace\_all(df\_stadiums$stadium\_name, "[^[:alnum:]]", " ")  
df\_stadiums$stad\_pkey <- gsub("[[:space:]]", "", df\_stadiums$stad\_pkey)  
  
# designate an SKEY for a unique identifier further downstream  
df\_main <- sqldf("SELECT   
 schedule\_date||team\_home AS SKEY  
 ,schedule\_week  
 ,schedule\_date  
 ,team\_home  
 ,team\_away  
 ,spread\_favorite  
 ,over\_under\_line  
 ,team\_favorite\_id  
 ,score\_home  
 ,score\_away  
 ,over\_under\_line  
 ,weather\_temperature  
 ,weather\_wind\_mph  
 ,weather\_humidity  
 ,stadium  
 FROM  
 spreadspoke\_scores  
 WHERE  
 schedule\_playoff <> 'TRUE'  
 AND schedule\_season > 1980  
 AND lower(schedule\_week) NOT IN ('wildcard','division','conference','superbowl')  
 ")  
  
# remove white space and special characters from stadium names to join further downstream   
df\_main$stad\_fkey <- str\_replace\_all(df\_main$stadium, "[^[:alnum:]]", " ")  
df\_main$stad\_fkey <- gsub("[[:space:]]", "", df\_main$stad\_fkey)  
  
# add a foreign key for the teams table  
df\_main$fav\_fkey <- str\_replace\_all(df\_main$team\_favorite\_id, "[^[:alnum:]]", " ")  
df\_main$fav\_fkey <- gsub("[[:space:]]", "", df\_main$fav\_fkey)  
  
# pull in required fields from the team table  
df\_teams <- sqldf("SELECT  
 team\_id  
 ,team\_name  
 FROM   
 df\_teams  
 WHERE   
 team\_division IS NOT NULL AND team\_division <> ''  
 AND team\_name <> 'San Diego Chargers'  
 ")  
# add primary key to the teams table  
df\_teams$fav\_pkey <- str\_replace\_all(df\_teams$team\_id, "[^[:alnum:]]", " ")  
df\_teams$fav\_pkey <- gsub("[[:space:]]", "", df\_teams$fav\_pkey)  
  
#combine all three tables  
df\_final <- sqldf("SELECT  
 a.\*, b.\*,c.\*  
 FROM   
 df\_main a  
 LEFT JOIN   
 df\_stadiums b ON a.stad\_fkey = b.stad\_pkey  
 LEFT JOIN  
 df\_teams c ON a.fav\_fkey= c.fav\_pkey  
 ")  
# had to remove null divisions further upstream and had to remove San Diego Chargers and keep LA Chargers  
checkDups <- sqldf("SELECT  
 COUNT(SKEY) as CNT  
 ,SKEY  
 FROM  
 df\_final  
 GROUP BY SKEY  
 HAVING COUNT(SKEY) > 1")  
  
winningTeam <- sqldf(' SELECT  
 a.SKEY  
 ,a.schedule\_date  
 ,a.team\_home  
 ,a.team\_away  
 ,a.spread\_favorite  
 ,CASE WHEN winningTeam = favTeam THEN 1 ELSE 0 END AS favWon  
 ,favTeam,winningTeam,winningTeamScore,losingTeamScore  
 ,a.over\_under\_line  
 ,winningTeamScore + losingTeamScore AS actualOverUnder  
 FROM   
 df\_final a  
 INNER JOIN  
 (SELECT  
 SKEY  
 ,team\_home  
 ,schedule\_date  
 ,CASE WHEN score\_home > score\_away THEN team\_home   
 WHEN score\_away > score\_home THEN team\_away  
 ELSE "tie" END AS winningTeam  
 ,CASE WHEN score\_home > score\_away THEN score\_home   
 WHEN score\_away > score\_home THEN score\_away  
 ELSE 0 END AS winningTeamScore  
 ,CASE WHEN score\_home > score\_away THEN score\_away   
 WHEN score\_away > score\_home THEN score\_home  
 ELSE 0 END AS losingTeamScore  
 ,over\_under\_line  
 ,team\_name as favTeam  
 FROM   
 df\_final a  
 ) scores ON a.SKEY = scores.SKEY   
 ')  
   
# create a calculated field to show who won and the final point spread outcome  
spreadCorrect1 <- sqldf('  
 SELECT  
 SKEY  
 ,schedule\_date  
 ,team\_home  
 ,team\_away  
 ,favTeam  
 ,winningTeam  
 ,winningTeamScore  
 ,losingTeamScore  
 ,spread\_favorite  
 ,actualOverUnder  
 ,over\_under\_line  
 ,CASE WHEN favWon = 1 THEN losingTeamScore - winningTeamScore   
 WHEN favWon = 0 THEN winningTeamScore - losingTeamScore  
 ELSE spread\_favorite end as actualSpread  
 ,CASE WHEN favTeam = winningTeam THEN 1 ELSE 0 END AS favTeamWon  
 FROM   
 winningTeam  
 ')  
  
# create a calculated field to show how off was the Over Under  
df\_final <- sqldf('SELECT  
 b.\*  
 ,a.favTeam  
 ,a.winningTeam  
 ,a.winningTeamScore  
 ,a.losingTeamScore  
 ,a.actualSpread  
 ,CASE WHEN a.actualSpread > 0 THEN 1 ELSE 0  
 END AS upsetInd  
 ,CASE WHEN a.favTeamWon = 1 THEN (abs(a.actualSpread) - abs(a.spread\_favorite))/abs(a.spread\_favorite)  
 ELSE (abs(a.actualSpread) - (a.spread\_favorite))/(a.spread\_favorite)  
 END\*100 prcntSprdOff  
 ,a.actualOverUnder  
 ,((abs(a.actualOverUnder) - abs(a.over\_under\_line))/abs(a.over\_under\_line))\*100 as prcntOvrUndrOff  
 FROM   
 spreadCorrect1 a  
 INNER JOIN df\_final b ON a.SKEY = b.SKEY  
 ')  
  
# bring in some weather variables to indicate the severity of the weather  
# cold season is defined as any games played in the last three weeks of the regular season  
# numDaysCold is defined as any game played where the temprature is <32 degrees  
df\_final <- sqldf("SELECT DISTINCT   
 a.\*  
 ,CASE WHEN schedule\_week < 14 THEN 0 ELSE 1 END AS coldSeason   
 ,CASE WHEN weather\_temperature > 20 and weather\_temperature <= 32 THEN 1  
 WHEN weather\_temperature > 10 and weather\_temperature < 20 THEN 2  
 WHEN weather\_temperature <= 10 THEN 3  
 ELSE 0  
 END AS coldDayLvl  
 FROM  
 df\_final a  
 ")  
  
# include the final metrics for analysis  
df\_final <- sqldf("SELECT  
 SKEY  
 ,schedule\_date  
 ,weather\_temperature  
 ,weather\_wind\_mph  
 ,weather\_humidity  
 ,stadium\_weather\_type  
 ,ELEVATION  
 ,LATITUDE  
 ,LONGITUDE  
 ,favTeam  
 ,winningTeam  
 ,actualSpread  
 ,upsetInd  
 ,prcntSprdOff  
 ,actualOverUnder  
 ,prcntOvrUndrOff  
 ,coldSeason  
 ,coldDayLvl  
 FROM df\_final  
 ")  
  
# remove NA values  
df\_cleaned <- na.omit(df\_final)

### What does the final data set look like?

head(df\_cleaned)

## SKEY schedule\_date weather\_temperature  
## 1 09/05/1981Tampa Bay Buccaneers 09/05/1981 81  
## 3 09/06/1981Buffalo Bills 09/06/1981 66  
## 4 09/06/1981Chicago Bears 09/06/1981 63  
## 5 09/06/1981Cincinnati Bengals 09/06/1981 69  
## 6 09/06/1981Denver Broncos 09/06/1981 67  
## 9 09/06/1981New England Patriots 09/06/1981 62  
## weather\_wind\_mph weather\_humidity stadium\_weather\_type ELEVATION  
## 1 8 74 warm 5.8  
## 3 8 94 cold 178.0  
## 4 8 84 cold 177.7  
## 5 8 71 cold 193.9  
## 6 9 58 cold 1611.2  
## 9 10 79 cold 24.4  
## LATITUDE LONGITUDE favTeam winningTeam  
## 1 27.96194 -82.54030 Tampa Bay Buccaneers Tampa Bay Buccaneers  
## 3 42.88900 -78.89010 Buffalo Bills Buffalo Bills  
## 4 41.85580 -87.60940 Chicago Bears Green Bay Packers  
## 5 39.10000 -84.51667 Cincinnati Bengals Cincinnati Bengals  
## 6 39.76330 -104.86940 Oakland Raiders Denver Broncos  
## 9 42.04790 -71.00500 New England Patriots Baltimore Colts  
## actualSpread upsetInd prcntSprdOff actualOverUnder prcntOvrUndrOff  
## 1 -8 0 700.0000 34 -8.108108  
## 3 -31 0 675.0000 31 -24.390244  
## 4 7 1 -275.0000 25 -32.432432  
## 5 -6 0 50.0000 48 17.073171  
## 6 2 1 -166.6667 16 -57.894737  
## 9 1 1 -116.6667 57 26.666667  
## coldSeason coldDayLvl  
## 1 0 0  
## 3 0 0  
## 4 0 0  
## 5 0 0  
## 6 0 0  
## 9 0 0

str(df\_cleaned)

## 'data.frame': 6989 obs. of 18 variables:  
## $ SKEY : chr "09/05/1981Tampa Bay Buccaneers" "09/06/1981Buffalo Bills" "09/06/1981Chicago Bears" "09/06/1981Cincinnati Bengals" ...  
## $ schedule\_date : Factor w/ 2300 levels "01/01/1967","01/01/1978",..: 270 280 280 280 280 280 280 280 280 288 ...  
## $ weather\_temperature : int 81 66 63 69 67 62 70 67 75 66 ...  
## $ weather\_wind\_mph : int 8 8 8 8 9 10 11 4 8 6 ...  
## $ weather\_humidity : Factor w/ 96 levels "","10","100",..: 71 91 81 68 53 76 61 78 77 77 ...  
## $ stadium\_weather\_type: chr "warm" "cold" "cold" "cold" ...  
## $ ELEVATION : num 5.8 178 177.7 193.9 1611.2 ...  
## $ LATITUDE : num 28 42.9 41.9 39.1 39.8 ...  
## $ LONGITUDE : num -82.5 -78.9 -87.6 -84.5 -104.9 ...  
## $ favTeam : chr "Tampa Bay Buccaneers" "Buffalo Bills" "Chicago Bears" "Cincinnati Bengals" ...  
## $ winningTeam : chr "Tampa Bay Buccaneers" "Buffalo Bills" "Green Bay Packers" "Cincinnati Bengals" ...  
## $ actualSpread : int -8 -31 7 -6 2 1 -14 4 -16 30 ...  
## $ upsetInd : int 0 0 1 0 1 1 0 1 0 1 ...  
## $ prcntSprdOff : num 700 675 -275 50 -167 ...  
## $ actualOverUnder : int 34 31 25 48 16 57 34 70 36 58 ...  
## $ prcntOvrUndrOff : num -8.11 -24.39 -32.43 17.07 -57.89 ...  
## $ coldSeason : int 0 0 0 0 0 0 0 0 0 0 ...  
## $ coldDayLvl : int 0 0 0 0 0 0 0 0 0 0 ...  
## - attr(\*, "na.action")= 'omit' Named int 2 7 8 12 21 25 26 30 34 36 ...  
## ..- attr(\*, "names")= chr "2" "7" "8" "12" ...

### What information is not self-evident?

At this point, finding a key that will link the three databases is a challenge. Looking through the datasets, there are several “like” columns. However, the data within them also varies. For example, there are several special characters in stadium name in one table, but not in the other. Also, the spreadspoke\_scores.csv dataset uses an abbreviation of the team name to determine the winner. This makes it a challenge to do any sort of one to one comparison of home team compared to favorite or away team vs favorite.

### What are different ways you could look at this data?

Utilizing the weather patterns, I hope to uncover some metrics that could help tell which teams could be considered “cold teams”, and which teams could be considered “warm teams” in terms of climate.

I also know that simple wins and losses aren’t necessarily enough to tell how well a team performed. For example, if the Miami Dolphins are heavy favorites against the Chicago Bears in December at Soldier Field, a win may not translate to cold weather impact. However, the total win margin vs the predicted spread and the over/under may be more telling.

### How do you plan to slice and dice the data?

I utilized my domain knowledge of the sport of football, along with some data manipulation to portray this data in different ways. I added several variables to the dataset, based on minor calculations from the three datasets. Joining the three datasets was a challenge, as the three data sets were missing a primary key.

### How could you summarize your data to answer key questions?

* Was the over/under correctly when cold weather was a factor
* Were spreads accurate when cold weather was a factor
* Do non-cold teams play worse in the cold (adjust by favorite and spread vs actual)
* Does wind correlate to wins/loses in cold weather?

### What types of plots and tables will help you to illustrate the findings to your questions?

I will start with a distribution of the data once it has been filtered, and once the new variables have been included in the data. This can be visualized with a histogram.

I want to show how well the data correlates when transformed. For example, will we see a linear relationship between data that is manipulated (i.e. wind chill and temperature) compared to under performance by a southern away team? A scatter plot can tell a valuable story, or even a timeseries showing how the weather has impacted the spread over the years. Have warmer climate teams performed better or worse in colder environments?

I also want to visualize the variables correlation coefficients in a correlation matrix.

Finally, I want to visualize the success of the algorithms applied.

### Do you plan on incorporating any machine learning techniques to answer your research questions? Explain.

I am going to apply a linear regression to see if we can predict future impacts based on cold weather scenarios.

I will utilize a logistic regression analysis to see if we can predict weather a team is impacted by cold weather events.