Who is Dominating Televised Collegiate Sports?

Jakob De La O; jmd5677

Introduction

The argument over which college or which NCAA Division I-A conference is dominating collegiate athletics is something I often see debated over on social media platforms like Youtube and Twitter. As a sports fan myself, I felt it fitting to try and answer these questions by focusing on two of the major televised sports in College Football and Men's College Basketball. The reason I wanted to focus on these two sports instead of multiple sports, is because these sports have viewership in the tens of millions and are often what most sports fans build their perception off of when comparing college athletic programs and conferences.

In this analysis, I will use data sets from the 2020-2021 athletic seasons for both College Football and Men's College Basketball since they both include post season wins. I will also only be considering the 127 collegiate teams and 11 conferences (in NCAA Division I-A) that participated in both sports for the 2020-2021 athletic season.

For the College Football data, I will be using a dataset obtained from Kaggle (https://www.kaggle.com/jeffgallini/college-football-team-stats-2019?select=cfb20.csv (https://www.kaggle.com/jeffgallini/college-football-team-stats-2019?select=cfb20.csv)) that includes the school names & respective conferences, total games, wins, losses, and another 148 advanced team statistics from the 2020-2021 football season. For the Men's College Basketball data, I will be using a dataset obtained from Kaggle (https://www.kaggle.com/andrewsundberg/college-basketball-dataset?select=cbb20.csv (https://www.kaggle.com/andrewsundberg/college-basketball-dataset?select=cbb20.csv)) that includes the rank, school names, conferences, total games, wins, and another 17 advanced team statistics from the 2020-2021 basketball season.

I expect the Power 5 conferences, like the ACC, Big 12, Big 10, Pac 12, and SEC, to have the higher overall win percentages in both sports compared to the rest of the conferences. This analysis should help in determining which of those has the highest overall and more specifically, which school(s) have the highest overall.

Analysis

First, we will read in the two datasets:

```r
library(tidyverse)
cfb.df <- read.csv("cfb20.csv") #reading in both sets
cbb.df <- read.csv("cbb20.csv")

head(cfb.df[1:10], 5) # a look at both sets
```

```
##                        Team Games Win Loss Off.Rank Off.Plays Off.Yards
## 1 Air Force (Mountain West)     6   3    3       73       373      2336
## 2               Akron (MAC)     6   1    5      121       360      1687
## 3             Alabama (SEC)    11  11    0        5       764      5983
## 4      App State (Sun Belt)    12   9    3       26       845      5424
## 5          Arizona (Pac-12)     5   0    5       86       373      1847
##   Off.Yards.Play Off.TDs Off.Yards.per.Game
## 1           6.26      19              389.3
## 2           4.69      12              281.2
## 3           7.83      68              543.9
## 4           6.42      49              452.0
## 5           4.95      10              369.4
```

```
head(cbb.df[1:10], 5)
```

```
##   RK         TEAM CONF  G  W ADJOE ADJDE BARTHAG EFG_O EFG_D
## 1  1       Kansas  B12 30 28 116.1  87.7  0.9616  53.7  43.7
## 2  2       Baylor  B12 30 26 114.5  88.4  0.9513  49.4  45.2
## 3  3      Gonzaga  WCC 33 31 121.3  94.3  0.9472  57.5  47.6
## 4  4       Dayton  A10 31 29 119.5  93.4  0.9445  59.7  46.6
## 5  5 Michigan St.  B10 31 22 114.8  91.3  0.9326  52.6  43.3
```

Tidying and Cleaning

Notice that in the CFB dataframe, our `Team` variable contains both the school name and conference name. We can use `dplyr` functions to tidy the data:

```
cfb.df <- cfb.df %>%
  mutate(Team = str_replace(Team, "\\(FL\\)", "FL")) %>%
  mutate(Team = str_replace(Team, "\\(OH\\)", "OH")) %>%
  separate(Team, into = c("Team", "Conference"), sep = "[()]") %>% # separates conferenc
es surrounded in ()
  mutate(Team = str_replace(Team, "App State", "Appalachian St.")) %>%
  mutate(Team = str_replace(Team, "Army West Point", "Army")) %>%
  mutate(Team = str_replace(Team, "Fla. Atlantic", "Florida Atlantic")) %>%
  mutate(Team = str_replace(Team, "Ga. Southern", "Georgia Southern")) %>%
  mutate(Team = str_replace(Team, "Middle Tenn.", "Middle Tennessee")) %>%
  mutate(Team = str_replace(Team, "Northern Ill.", "Northern Illinois")) %>%
  mutate(Team = str_replace(Team, "Southern Miss.", "Southern Miss")) %>%
  mutate(Team = str_replace(Team, "South Fla.", "South Florida")) %>%
  mutate(Team = str_replace(Team, "Southern California", "USC")) %>%
  mutate(Team = str_replace(Team, "California", "UCLA")) %>%
  mutate(Team = str_replace(Team, "Western Ky.", "Western Kentucky")) %>%
  mutate(Team = str_replace(Team, "Western Mich.", "Western Michigan")) %>%
  mutate(Team = str_replace(Team, "Central Mich.", "Central Michigan")) %>%
  mutate(Team = str_replace(Team, "Eastern Mich.", "Eastern Michigan")) %>%
  # fixes disparity w cbb dataset for later merging
  mutate(Team = str_remove_all(Team, " ")) # removes spaces
```

Since we are concerned with overall performance, we will want each dataset to have total games played, wins, losses, and some way to measure the margin of the difference in points between each team and their opponents. First, let us focus on the CFB dataset:

```
cfb.df.new <- cfb.df %>%
  select(Team, Conference, Games, Win, Loss,
         Avg.Points.per.Game.Allowed, Points.Per.Game) %>% #selects all relevent columns
  rename(Gcfb = Games, Wcfb = Win, Lcfb = Loss,
         OPPG = Avg.Points.per.Game.Allowed, PPG = Points.Per.Game)
# renames columns to account for cfb stat prior to merge

head(cfb.df.new, 5) # a look at the new dataset
```

```
##             Team    Conference Gcfb Wcfb Lcfb OPPG  PPG
## 1       AirForce Mountain West    6    3    3 15.0 24.3
## 2          Akron           MAC    6    1    5 41.3 17.2
## 3        Alabama           SEC   11   11    0 19.5 49.7
## 4 AppalachianSt.      Sun Belt   12    9    3 20.0 33.8
## 5        Arizona        Pac-12    5    0    5 39.8 17.4
```

Now, since our CBB dataset does not include points per game and/or points per game allowed, we will use the “adjusted offensive efficiency” (`ADJOE`) and “adjusted defensive efficiency” (`ADJDE`) variables to examine the margin of the difference in points scored by each team and their opponents. The adjusted offensive efficiency is a metric that averages the points scored by a team per 100 possessions (times they have the ball), while the adjusted defensive efficiency is a metric that averages the points scored by a team’s opponent per 100 possessions (times the opponent had the ball). Although this is not the exact same as average points per game and average opponents points per game, it’s a similar enough statistic that we can use to compare. Cleaning the CBB dataset for our desired statistics, we have that:

```
cbb.df.new <- cbb.df %>%
  select(TEAM, G, W, ADJOE, ADJDE) %>% # selects relevant variables
  rename(Team = TEAM, Gcbb = G, Wcbb = W) %>% # renames to account for cbb data prior to
merge
  mutate(Lcbb = Gcbb - Wcbb) %>%
  # creates a column for total losses for cbb data
  mutate(Team = str_replace(Team, "Louisiana Lafayette", "Louisiana")) %>%
  mutate(Team = str_replace(Team, "Louisiana Monroe", "ULM")) %>%
  mutate(Team = str_replace(Team, "North Carolina St.", "NC State")) %>%
  mutate(Team = str_replace(Team, "Mississippi", "Ole Miss")) %>%
  mutate(Team = str_remove_all(Team, " ")) %>% # removes spaces
  mutate(Team = str_replace(Team, "OleMissSt.", "MississippiSt."))
  # fixes disparity w cfb dataset for later merging

head(cbb.df.new, 5) # a look at the new dataset
```

```
##          Team Gcbb Wcbb ADJOE ADJDE Lcbb
## 1      Kansas   30   28 116.1  87.7    2
## 2      Baylor   30   26 114.5  88.4    4
## 3     Gonzaga   33   31 121.3  94.3    2
## 4      Dayton   31   29 119.5  93.4    2
## 5 MichiganSt.   31   22 114.8  91.3    9
```

Because we already have conferences in the CFB dataset, I felt it was more efficient to remove the `CONF` column from the CBB dataset.

Merging the Data

Being that our data is now clean and tidy, we can now join the two sets to begin our statistical analysis. We only have one key to merge the data, that being the name of the school. To ensure we have columns from both sets while only including the 127 teams that participated in both sports, we will use the `inner_join` function to merge the data:

```
ath.stats <- cfb.df.new %>%
 inner_join(cbb.df.new, by = "Team") #inner join to get teams that participated in both
 sports

head(ath.stats, 5)
```

```
##             Team    Conference Gcfb Wcfb Lcfb OPPG  PPG Gcbb Wcbb ADJOE ADJDE
## 1       AirForce Mountain West    6    3    3 15.0 24.3   31   12 106.4 109.1
## 2          Akron           MAC    6    1    5 41.3 17.2   29   24 107.9  98.8
## 3        Alabama           SEC   11   11    0 19.5 49.7   31   16 111.4  99.2
## 4 AppalachianSt.      Sun Belt   12    9    3 20.0 33.8   31   18  98.3 101.5
## 5        Arizona        Pac-12    5    0    5 39.8 17.4   32   21 110.9  91.2
##   Lcbb
## 1   19
## 2    5
## 3   15
## 4   13
## 5   11
```

If we consider the number of variables and observations in each dataset prior to the merge, using the `glimpse()` function, we have that:

```
glimpse(cfb.df.new)
```

```
## Rows: 127
## Columns: 7
## $ Team       <chr> "AirForce", "Akron", "Alabama", "AppalachianSt.", "Arizona"…
## $ Conference <chr> "Mountain West", "MAC", "SEC", "Sun Belt", "Pac-12", "Pac-1…
## $ Gcfb       <int> 6, 6, 11, 12, 5, 4, 10, 11, 11, 10, 7, 9, 7, 11, 5, 7, 12, …
## $ Wcfb       <int> 3, 1, 11, 9, 0, 2, 3, 4, 9, 6, 6, 2, 5, 6, 0, 6, 11, 1, 3, …
## $ Lcfb       <int> 3, 5, 0, 3, 5, 2, 7, 7, 2, 4, 1, 7, 2, 5, 5, 1, 1, 3, 3, 4,…
## $ OPPG       <dbl> 15.0, 41.3, 19.5, 20.0, 39.8, 23.2, 34.9, 37.2, 14.0, 23.7,…
## $ PPG        <dbl> 24.3, 17.2, 49.7, 33.8, 17.4, 40.2, 25.7, 32.9, 27.3, 25.7,…
```

```
glimpse(cbb.df.new)
```

```
## Rows: 353
## Columns: 6
## $ Team  <chr> "Kansas", "Baylor", "Gonzaga", "Dayton", "MichiganSt.", "Duke", …
## $ Gcbb  <int> 30, 30, 33, 31, 31, 31, 30, 31, 31, 31, 31, 31, 32, 31, 31, 31, …
## $ Wcbb  <int> 28, 26, 31, 29, 22, 25, 24, 21, 24, 30, 24, 23, 21, 19, 21, 21, …
## $ ADJOE <dbl> 116.1, 114.5, 121.3, 119.5, 114.8, 115.3, 120.6, 114.6, 115.1, 1…
## $ ADJDE <dbl> 87.7, 88.4, 94.3, 93.4, 91.3, 91.9, 96.4, 92.6, 93.9, 92.8, 93.7…
## $ Lcbb  <int> 2, 4, 2, 2, 9, 6, 6, 10, 7, 1, 7, 8, 11, 12, 10, 10, 7, 7, 10, 5…
```

This tells us that the new CFB dataset has 7 variables and 127 observations, and the new CBB dataset has 6 variables with 353 observations. Since there is only one key in each dataset and those keys are common, when we merged the two datasets, it resulted in 12 variables since $7 + (6 - 1) = 7 + 5 = 12$. Since we used an inner join, we expect to have 127 observations. Taking a glimpse of our merged dataset will verify how many variables there are, their type, and the total number of observations:

```
glimpse(ath.stats)
```

```
## Rows: 127
## Columns: 12
## $ Team       <chr> "AirForce", "Akron", "Alabama", "AppalachianSt.", "Arizona"…
## $ Conference <chr> "Mountain West", "MAC", "SEC", "Sun Belt", "Pac-12", "Pac-1…
## $ Gcfb       <int> 6, 6, 11, 12, 5, 4, 10, 11, 11, 10, 7, 9, 7, 11, 5, 7, 12, …
## $ Wcfb       <int> 3, 1, 11, 9, 0, 2, 3, 4, 9, 6, 6, 2, 5, 6, 0, 6, 11, 1, 3, …
## $ Lcfb       <int> 3, 5, 0, 3, 5, 2, 7, 7, 2, 4, 1, 7, 2, 5, 5, 1, 1, 3, 3, 4,…
## $ OPPG       <dbl> 15.0, 41.3, 19.5, 20.0, 39.8, 23.2, 34.9, 37.2, 14.0, 23.7,…
## $ PPG        <dbl> 24.3, 17.2, 49.7, 33.8, 17.4, 40.2, 25.7, 32.9, 27.3, 25.7,…
## $ Gcbb       <int> 31, 29, 31, 31, 32, 31, 32, 30, 29, 31, 30, 30, 31, 32, 29,…
## $ Wcbb       <int> 12, 24, 16, 18, 21, 20, 20, 16, 15, 25, 18, 26, 20, 13, 21,…
## $ ADJOE      <dbl> 106.4, 107.9, 111.4, 98.3, 110.9, 106.1, 108.9, 102.2, 99.0…
## $ ADJDE      <dbl> 109.1, 98.8, 99.2, 101.5, 91.2, 94.4, 96.1, 108.7, 107.5, 9…
## $ Lcbb       <int> 19, 5, 15, 13, 11, 11, 12, 14, 14, 6, 12, 4, 11, 19, 8, 11,…
```

We see that our new dataset `ath.stats` does indeed have 12 variables and 127 observations. Since we have 127 observations in our merged dataset, we know that we dropped 226 rows given that $353 - 127 = 226$. This can be verified by the following code:

```
nrow(cbb.df.new) - nrow(ath.stats) # difference in rows
```

```
## [1] 226
```

There is no issue with the omission of the 226 observations since they pertain to schools that did not participate in NCAA Division 1-A College Football during the 2020-2021 season or they do not participate in NCAA Division 1-A College Football in general.

Wrangling

With our new merged dataset, we can now perform `dplyr` functions on it to begin our analysis. First, we will look at overall win percentage per school and conference. Let us first look at each school:

```
tot.wins.tm <- ath.stats %>% # dataset containing win totals & win pct by team
  group_by(Team) %>%
  summarize(W = Wcfb + Wcbb, L = Lcfb +Lcbb, G = Gcfb + Gcbb)  %>%
  mutate(win.pct = W/G) # new column containing win pctage

tot.wins.conf <- ath.stats %>% # dataset containing win totals & win pct by conf
  group_by(Conference) %>%
  summarize(W = sum(Wcfb + Wcbb), L = sum(Lcfb +Lcbb), G = sum(Gcfb + Gcbb)) %>%
  mutate(win.pct = W/G) # new column containing win pctage
```

With these two new datasets, we can find the most “winningest” teams and conferences respectively (the team(s)/conference(s) with the highest win percentage):

```
tot.wins.tm %>%
  arrange(desc(win.pct)) %>% # arranges win pctages in descending order
  select(Team, win.pct) %>%
  head(5) # top 5 winningest teams
```

```
## # A tibble: 5 × 2
## # Groups:   Team [5]
##   Team        win.pct
##   <chr>         <dbl>
## 1 Liberty       0.929
## 2 SanDiegoSt.   0.872
## 3 BYU           0.814
## 4 Oregon        0.757
## 5 Auburn        0.756
```

```
tot.wins.conf %>%
  select(Conference, win.pct) %>%
  slice_max(order_by = win.pct, n = 5) # essentially same process as before but with a d
ifferent function
```

```
## # A tibble: 5 × 2
##   Conference      win.pct
##   <chr>             <dbl>
## 1 FBS Independent   0.7
## 2 Pac-12            0.600
## 3 Big 12            0.583
## 4 MAC               0.578
## 5 Big Ten           0.571
```

We see that the schools with the highest winning percentages overall for football and men’s basketball during the 2020-2021 athletic season were Liberty, San Diego State, Brigham Young University, Oregon and then Auburn, with Liberty having had the highest at about a 93% win rate.

Then, looking at conferences, we see that the conferences with highest winning percentages overall for football and men’s basketball during the 2020-2021 athletic season were FBS Independent, the Pac 12, the Big 12, the MAC, and then the Big 10 with the Independent schools having had the highest winning percentage at a 70% win rate. It may be worth noting that the conference listed as “FBS Independent” consists of all the schools that are not a part of a conference. These are schools that have full freedom over creating their own schedules and subsequently play schools of their choice.

If we consider teams that had a winning percentage of over 50%, then we consider them “winning programs”. With this in mind, we have that:

```
winsNstat <- tot.wins.tm %>%
  mutate(status = ifelse(win.pct > 0.5, "Winning", "Average or Below")) # defining winni
ng programs
```

Now, we can determine the percentage of teams that consider “winning” programs out of the total 127 colleges that participated in both sports during the 2020-2021 season:

```
winners <- winsNstat %>%
  filter(status == "Winning") %>%
  nrow() # counts number of winning programs

winners/127 # percentage of winning programs
```

```
## [1] 0.6692913
```

From this calculation, we see that about 67% of the 127 college programs that participated in both college football and men’s basketball were considered winning programs during the 2020-2021 athletic season.

Now, let us consider the point differentials for both college football and men’s basketball. We can create a dataset with the point differentials for each sport by team, that way we can create visualizations with it later:

```
pt.diff.df <- ath.stats %>%
  select(Team, Conference, PPG, OPPG, ADJOE, ADJDE) %>% # uses only relevant stats
  mutate(pt.diff.cfb = PPG - OPPG, pt.diff.cbb = ADJOE-ADJDE) # new columns w pt differe
ntials for each sport
```

With these two new datasets, we can find the teams that had the largest point differentials in each respective sport for the 2020-2021 athletic season:

```
pt.diff.df %>%
  select(Team, pt.diff.cfb) %>%
  slice_max(order_by = pt.diff.cfb, n =5)  # top 5 largest pt diffs for footballl
```

```
##         Team pt.diff.cfb
## 1    Alabama        30.2
## 2        BYU        28.2
## 3    Clemson        27.4
## 4 Cincinnati        23.3
## 5    Buffalo        21.5
## 6    OhioSt.        21.5
```

```
pt.diff.df %>%
  select(Team, pt.diff.cbb) %>%
  slice_max(order_by = pt.diff.cbb, n =5) # top 5 largest pt diffs for basketball
```

```
##          Team pt.diff.cbb
## 1      Kansas        28.4
## 2      Baylor        26.1
## 3 MichiganSt.        23.5
## 4        Duke        23.4
## 5     OhioSt.        22.0
```

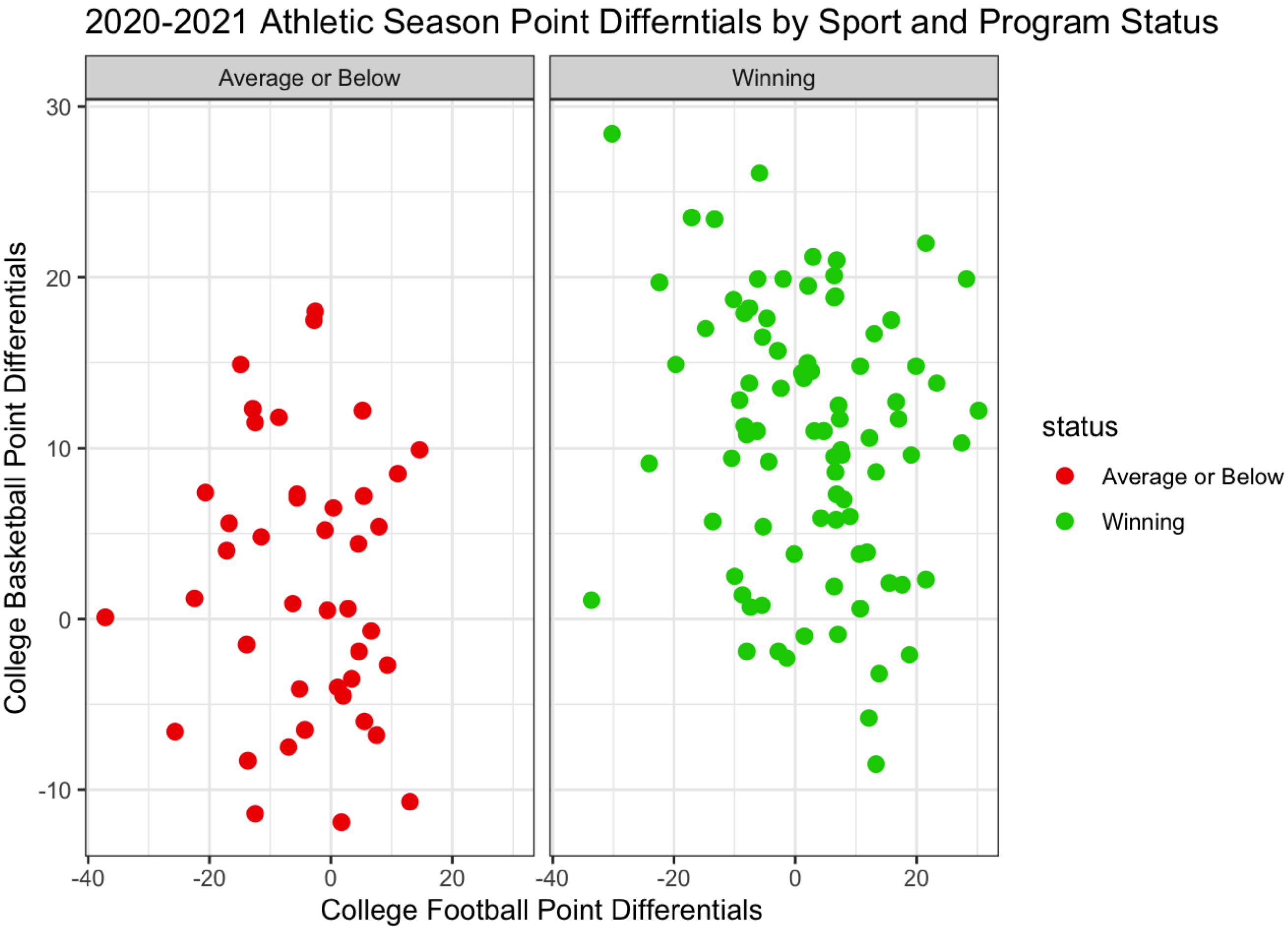
We see that, in the 2020-2021 football season, the teams with the largest point differentials were Alabama, Brigham Young University, Clemson, Cincinnati, the Buffalo and Ohio State. Here, we see that Alabama had the largest point differential in college football with a point differential of 30.2 points.

Then, in the 2020-2021 men's college basketball season, the teams with the largest point differentials were Kansas, Baylor, Michigan State, Duke, and Ohio State, with Kansas having had the largest point differential of 28.4 points.

Visualizations

With our new dataset that includes point differentials, we can merge that with our dataset that includes our "winning program" classification so that we can create a scatter plot:

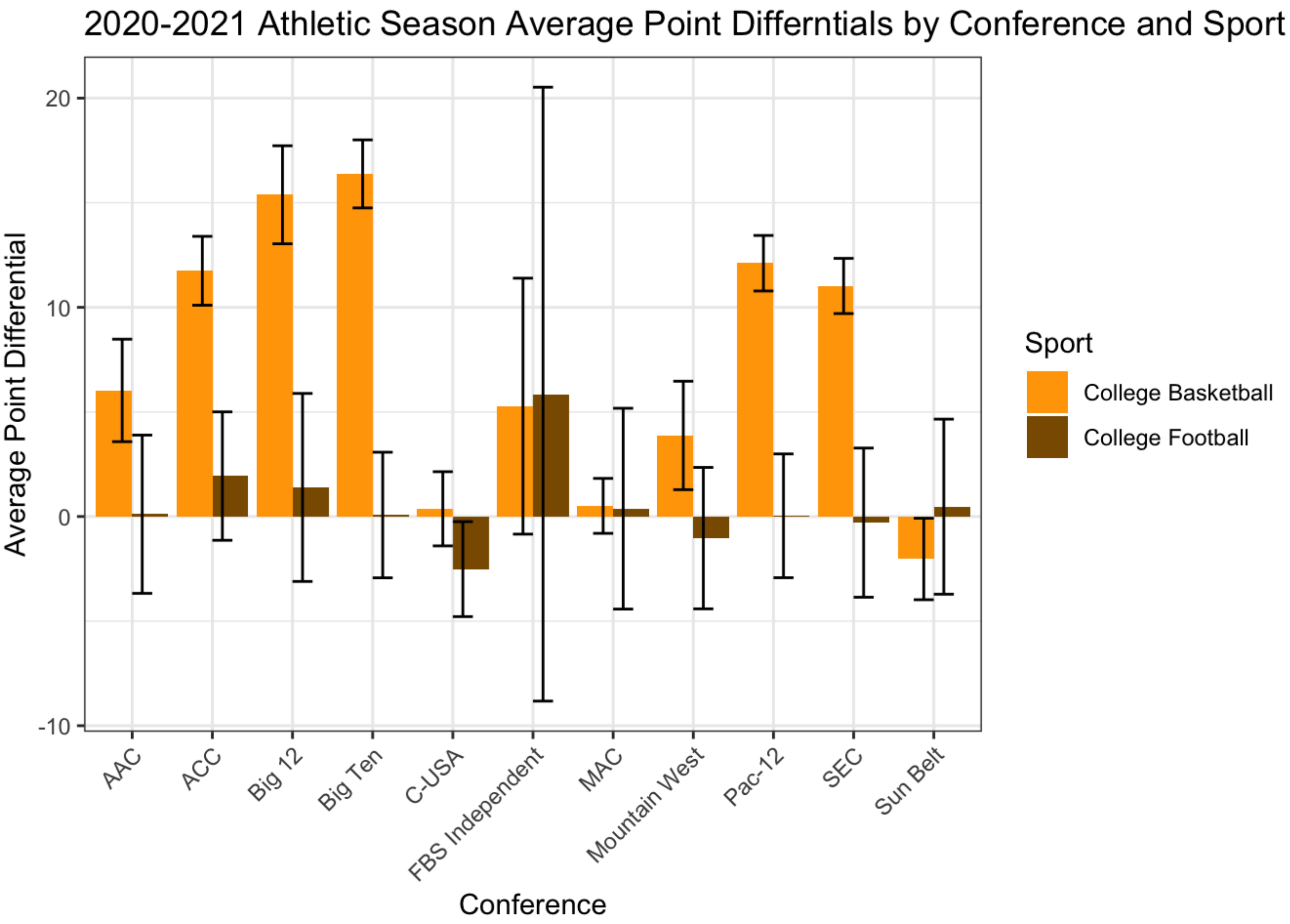
```
pt.diff.df %>%
  inner_join(winsNstat, by = "Team") %>% # so we can include winning teams classificatio
n
  ggplot(aes(x = pt.diff.cfb, y = pt.diff.cbb, color = status)) + # separates by status
  geom_point(size = 2.5) + # scatter plot
  facet_grid(~status) + # separates by status
  labs(x = "College Football Point Differentials",
       y = "College Basketball Point Differentials", title = "2020-2021 Athletic Season
 Point Differntials by Sport and Program Status") + # labels
  scale_color_manual(values = c("red2", "green3")) + # changes color scale
  theme_bw() # changes theme
```



In explaining the plot above, it is important to note that observations that fall within the region where college basketball point differentials are less than or equal to 0 and college football point differentials are less than or equal to 0 ($y \leq 0, x \leq 0$) denotes negative point differentials for each sport. We would expect “winning” teams to have observations that fall outside of this quadrant, with more observations in the region where college basketball point differentials are greater than 0 and college football point differentials are greater than 0 ($y > 0, x > 0$). Observing the plot above, we see that this is indeed what seems to be going on. Most of the observations for teams with average or below average winning percentages seem to lie in (or lie very closely to) the region where college basketball point differentials are less than or equal to 0 and college football point differentials are less than or equal to 0. Notice then that, most of the observations for teams with winning percentages seem to lie outside the same region, with very few observations inside.

Our next visualization will seek to answer the question we posed in the introduction of our project. We will explore the average point differentials by conference:

```
pt.diff.df %>%
  pivot_longer(pt.diff.cfb:pt.diff.cbb, names_to = "Sport", values_to = "diff") %>%
  # separates by sport so we can split the bars for each sport
  mutate(Sport = str_replace(Sport, "pt.diff.cfb", "College Football")) %>%
  mutate(Sport = str_replace(Sport, "pt.diff.cbb", "College Basketball")) %>%
  # renames variables for presentation in legend
  ggplot(aes(x = Conference, y = diff, fill = Sport)) + # normal ggplot func
  geom_bar(stat = "summary", fun = "mean", position = "dodge") + # avg pt diff in each c
onf
  geom_errorbar(stat = "summary", fun.data = "mean_se", width = 0.5,
                position = "dodge") + # error bars
  labs(y = "Average Point Differential", title = "2020-2021 Athletic Season Average Poin
t Differntials by Conference and Sport") + # labels
  theme_bw() + # changes theme
  scale_fill_manual(values = c("orange", "orange4")) + # changes color scale
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) #declutters x-axis
```



Observing the plot above we see that the conference with the largest point differential in men's basketball was the Big Ten, with an average point differential of over 15 points. For college football, we have that the collection of FBS Independent schools had the largest average point differential of over 5 points. Notice here that the error bar for this average is very large. This may be because of the fact that the collection of FBS Independent schools played significantly less college football games compared to the actual conferences. Moving on to the conferences with the worst average point differentials, we have that the Sun Belt conference had the worst point differential in men's basketball during the 2020-2021 season as it had the only negative average point differential (between 0 and -5 points). For college football we see that Conference USA had the worst average point differential during the 2020-2021 season with the Mountain West Conference being the only other conference with a negative average, but not as low as that of Conference USA.

```
##
sysname
##
"Darwin"
##
release
##
"20.6.0"
##
version
## "Darwin Kernel Version 20.6.0: Mon Aug 30 06:12:20 PDT 2021; root:xnu-7195.141.6~3/RE
LEASE_ARM64_T8101"
##
nodename
##
"Jakobs-Air.lan"
##
machine
##
"arm64"
##
login
##
"root"
##
user
##
"jakobdelao"
##
effective_user
##
"jakobdelao"
```