Rookie Contract Analysis

Aidan McGrath, Ethan Martin, and Mitchell Darling

2025-05-07

Overview

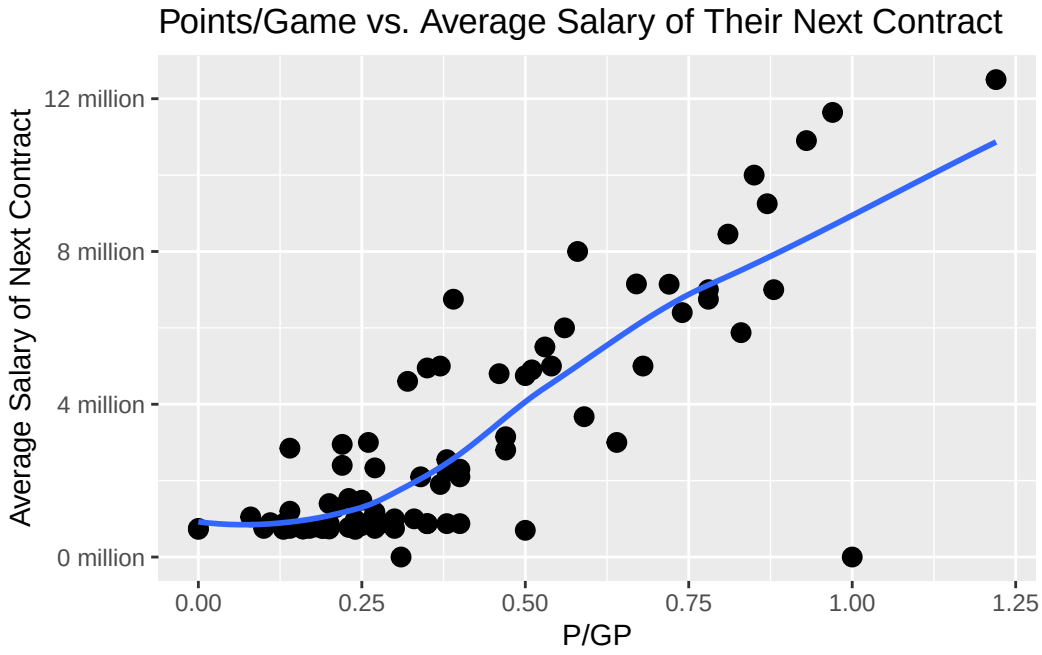
The goal of our project is to see if there is a relation between an athlete's performance during their rookie contract and how much their next contract is worth. We are looking at the NHL, NBA, and NFL to see if there are major differences between the leagues.

NHL Data

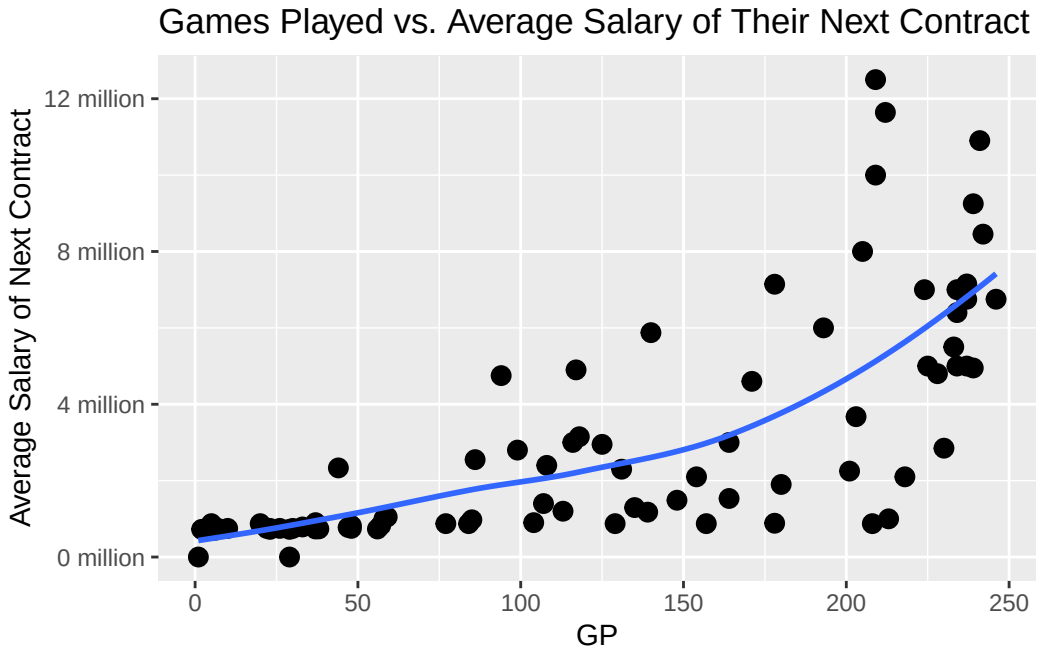
The NHL's draft is different from other leagues because drafted players often do not join their teams right away. This leads rookie contracts for players drafted in the same year to not end at the same time. This created some challenges because I could not simply merge a table of players in a draft class with their statistics over the next three seasons. I had to get data for different three season periods because of this. This allowed me to merge players with the correct statistics for their rookie contract.

The NHL player statistics were pulled from nhl.com using their export feature. Their website allows users to filter statistics for players drafted in certain drafts across specific seasons. They provide an export button to export the filtered statistics into an excel file. The contract data was taken from spotrac.com. There was no way to scrape the data directly using R because there was not a page that had all of the information I needed. I had to manually enter player's contract information into an excel file that had the first 50 players drafted in 2015 and 2016. I then had to wrangle the data to remove cases that had empty values. I was then able to merge the files together so every player had their correct statistics for their rookie contract. This resulted in each case being individual player.

This data adheres to CARE principles because all of the data I used is publicly available and I gave credit to them. It also adheres to FAIR principles because this data is easily found at the websites I mentioned. I also used tidyverse to create the following data visualizations.



This data visualization shows the relationship between a player's points per game played and the average salary of of their next contract. In the NHL, when a player scores a goal or assists on a goal they are awarded a point. There is a clear relationship here that shows that the more points per game played a player gets, the higher their next contract is worth. There is an outlier in this data set where a player had 1 point per game but did not receive a contract after their rookie contract ended. This case is Egor Korshkov who decided to play in the Russian league after his rookie contract was over.

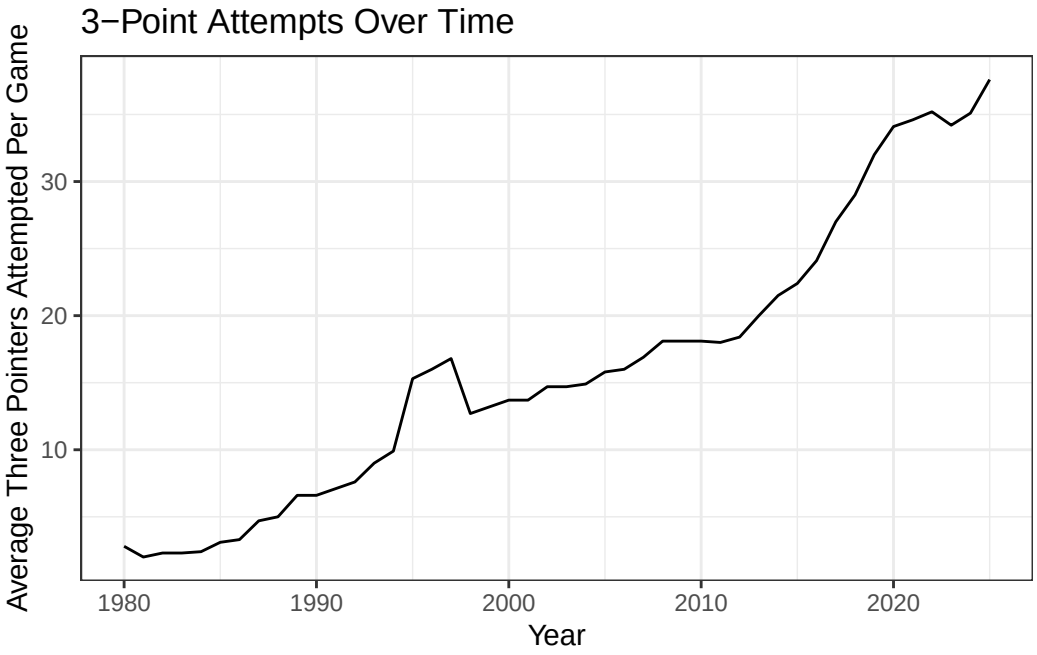


This data visualization shows the relationship between games played during a player's rookie contract and the average salary of their next contract. There is still a relationship in this graph but it is not as strong as the first one. This means that the strongest predictor for average salary in the NHL is points per game played. This is an example of exploratory data analysis because the goal of these visualizations is to see if there are any relationships, not confirming a hypothesis.

NBA Data

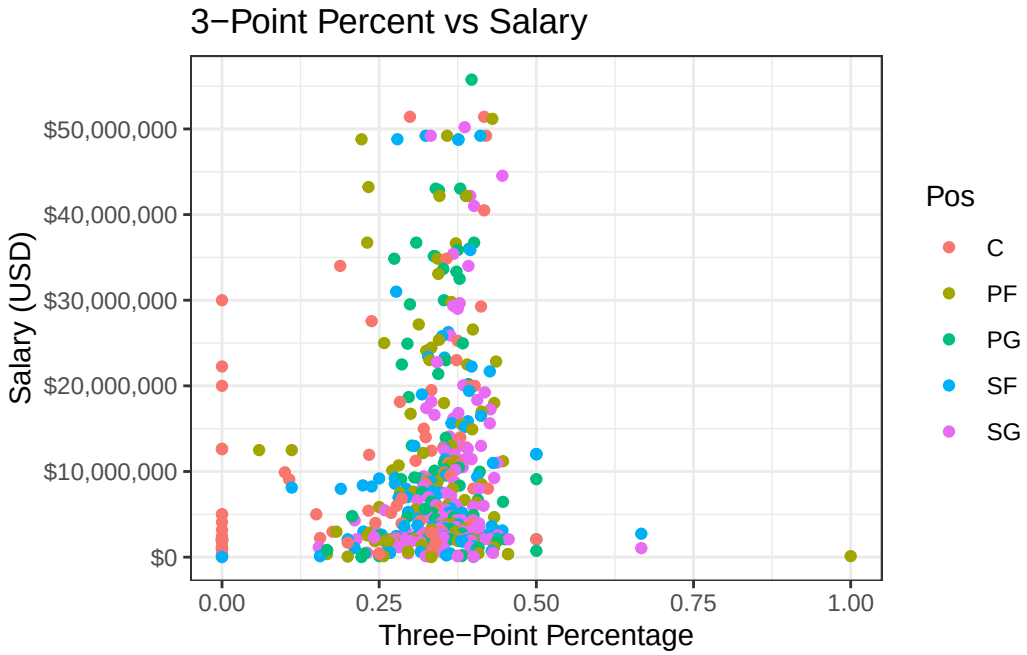
Three-Point Shooting Over Time

Since the implementation of the 1979-1980 Season, NBA teams have been taking more threes every year. The graph below is the league average 3-pointers attempted per game for a team.



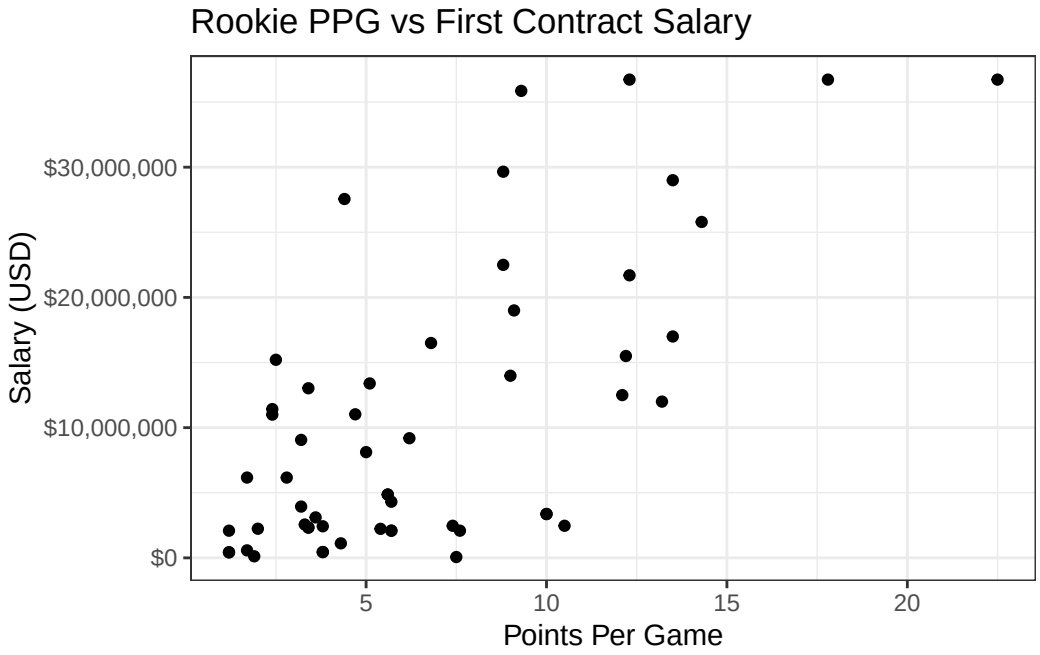
Three-Point Percentage vs Salary

With the exception of the center position, every player in the NBA is expected to shoot threes efficiently. You will find very few NBA players that are paid more than 20 million dollars per year that shoot less than 25% from three. The few that do are Power Forwards (PF) and Centers (C) who can traditionally create value from their size with rebounds and blocks.



Rookie Production vs First Salary

The graph below shows the relationship between the 2019-2020 rookie production (measured in PPG) and Salary in 2024-2025, which is the first year that most signed their first contract after their rookie contracts. Usually rookie contracts act as a sort of trial period that pay better every year for four years. Starting in their fifth year, rookies can sign a full contract that pays much more. This is a performance incentive for rookies.



NFL Data

This section focuses on data from the NFL. This ranges from the draft data, to contract/salary information, to player statistics throughout the duration of the season. I aim to perform an EDA based approach on these datasets in order to identify what factors play into the value of a player's contract, whether that be draft pick positioning, college career stats, production in the league, or any other factor that becomes relevant when discussing a player's paycheck. Specifically, I will be focusing on data from the 2020 NFL Draft class. This gives me a 5-year period (seasons 2020-2024) to study and evaluate the performance of new players. In an attempt to isolate variables and factors that will play into how a player is payed, I am focusing my analysis to offensive players. Specifically what are called "Skill Position" players. These are offensive players that score touchdowns: Quarterback (QB), running back (RB), wide receiver (WR), and tight end (TE). However, I will be omitting the tight end position from the majority of analyses due to variations in data, as the tight end position's purpose isn't just to score points. They are often used as a power back, a blocker, and then also a receiver to catch passes, so due to the varied nature of this position, I won't be using it when evaluating scoring statistics.

The NFL uses position abbreviations in order to simplify names and reduce space taken up, and so have I. So for those unaware of football argon, here are what each position abbreviation means:

C – Center, CB – Corner Back (Often grouped with safeties as DB), ED – Edge, FB – Full Back, IDL – Interior Defensive Lineman, K – Kicker, LB – Linebacker, LG – Left Guard, LS – Long Snapper, LT – Left Tackle, P – Punter, QB – Quarterback, RB – Running Back, RG – Right Guard, RT – Right Tackle, S – Safety, TE – Tight End, WR – Wide Receiver.

The data is sourced from various sites, but the majority is from websites like Pro Football Reference and Spotrac, two sites that prioritize accurate, free, and public records of anything NFL related. Draft stats, team stats, stat leaders, and any other statistic of the sort that pertains to NFL football can be found on these sites and are commonly cross referenced and used by professional sport analysts nationwide. I used Pro Football Reference for the 2020 NFL Draft order and the college career stats of the players selected in said draft year. Spotrac was used for contract information, mainly for salary extensions throughout the specified time period. I also used an r package called "nflreadr", which had more useful information about contract periods and player salary.

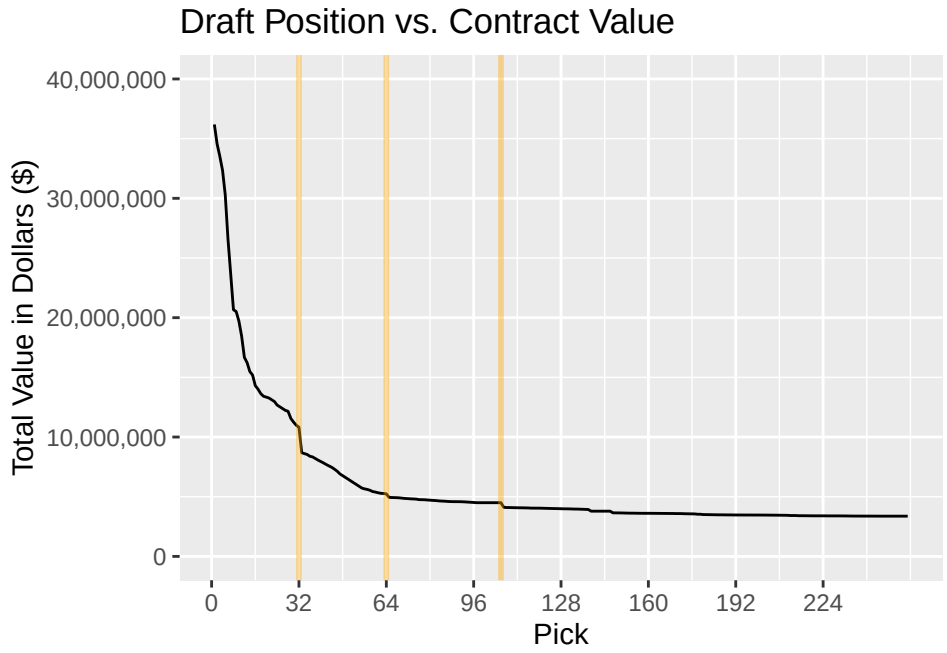


Figure 1: Scale of Contract Value vs. Draft Pick

The graph above is a typical line graph of the rookie contract value of all players drafted in the 2020 NFL draft. The draft is split into 7, 32 pick rounds, but this number isnt always firm due to forfeiture and compensatory picks. But the first and second round all experience a significant drop in salary value upon their conclusion, while the drop decreases steadily as the draft progresses. This graph has a few vertical lines to isolate the first, second, and third rounds. We can pick out from this graph that the NFL (and their teams) all prioritize and

dump as much money as they can into the first round of the draft, as this is when the best players end up getting drafted. This is one of the few solid factors that are directly related to a player's contract value.

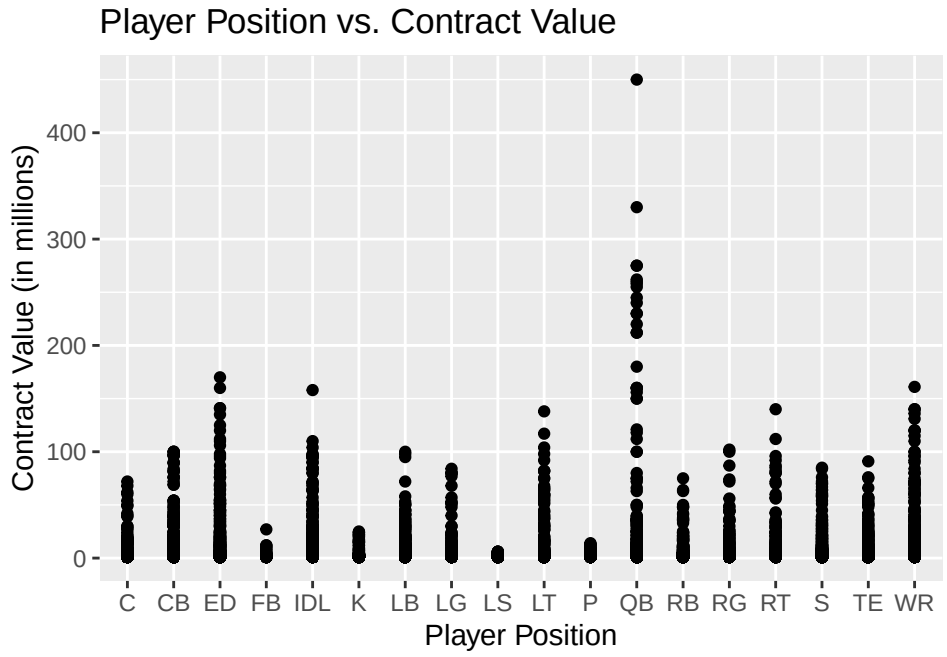


Figure 2: Position vs. Contract Value by Position

This is a grouped scatter plot, grouped by player position, where each case is an individual player and their contract. If the position names at the bottom are still a little confusing, please refer to the key stated in the NFL section's introduction. Not much can be gained from this graph, or a visualization of any sort based upon this dataset due to its scattered and varied nature, however, we do see that the Quarterback position, by far, has the highest value associated with it. Which makes sense when we consider that the quarterback is the only player (aside from the center, who starts the play) who handles the football every single play. They also are the primary point scorers, as they throw the ball to the receivers (which counts as a score for both players, statistic-wise), and can also run the ball to score. Coming up next behind the QB position is the edge rushers, followed by wide receivers. Edge rushers value comes from their ability to pressure the quarterback and halt run plays, making them incredibly valuable on defense. As for the receivers, they bring the bag home in regards to how necessary they are for a high powered offense to score. They create threats down the field and open the play up, again, making them a highly sought after asset.

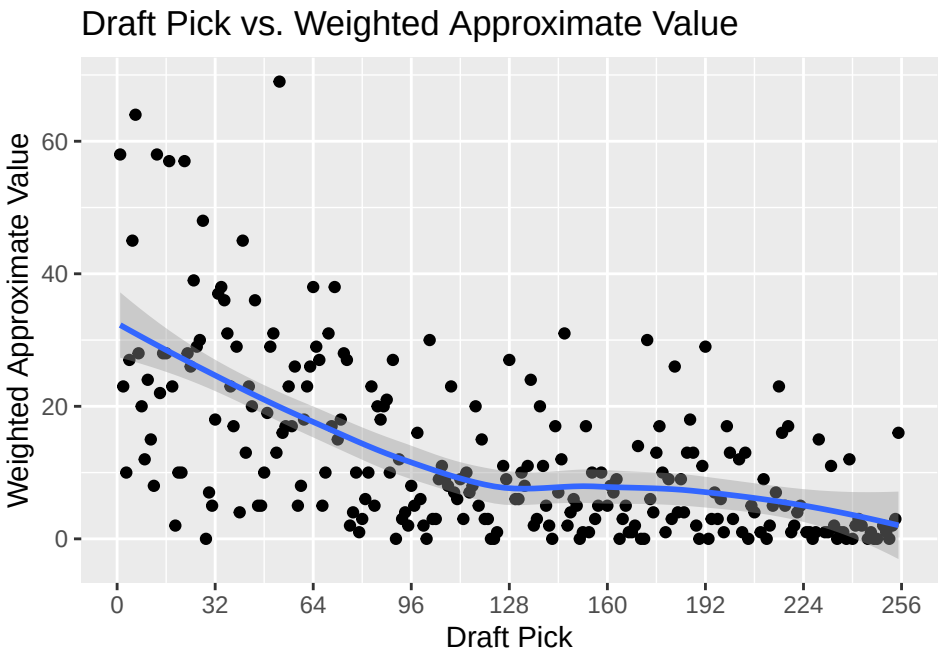


Figure 3: Draft Pick vs. Weighted Approximate Value

This visualization explores the affect of WAV (wighted approximate value). WAV is a statistic used to evaluate a college football player's overall "value" or contribution to the team. This statistic varies across different positions and rarely is a determining factor in where and when a player gets drafted. It is solely a number that a draftee or college player is given that corresponds to their "seasonal value". Here we can see that the higher WAV players do tend to get drafted earlier, but that observation only holds true in the earlier rounds of the draft. After the third round (usually around pick 96) we can see that higher WAV does not relate to the position the draftee is taken. I have added a trend line to help visualize variations in this graph but it proves the point that WAV does not play a meaningful factor in the outcome of a player's draft spot.

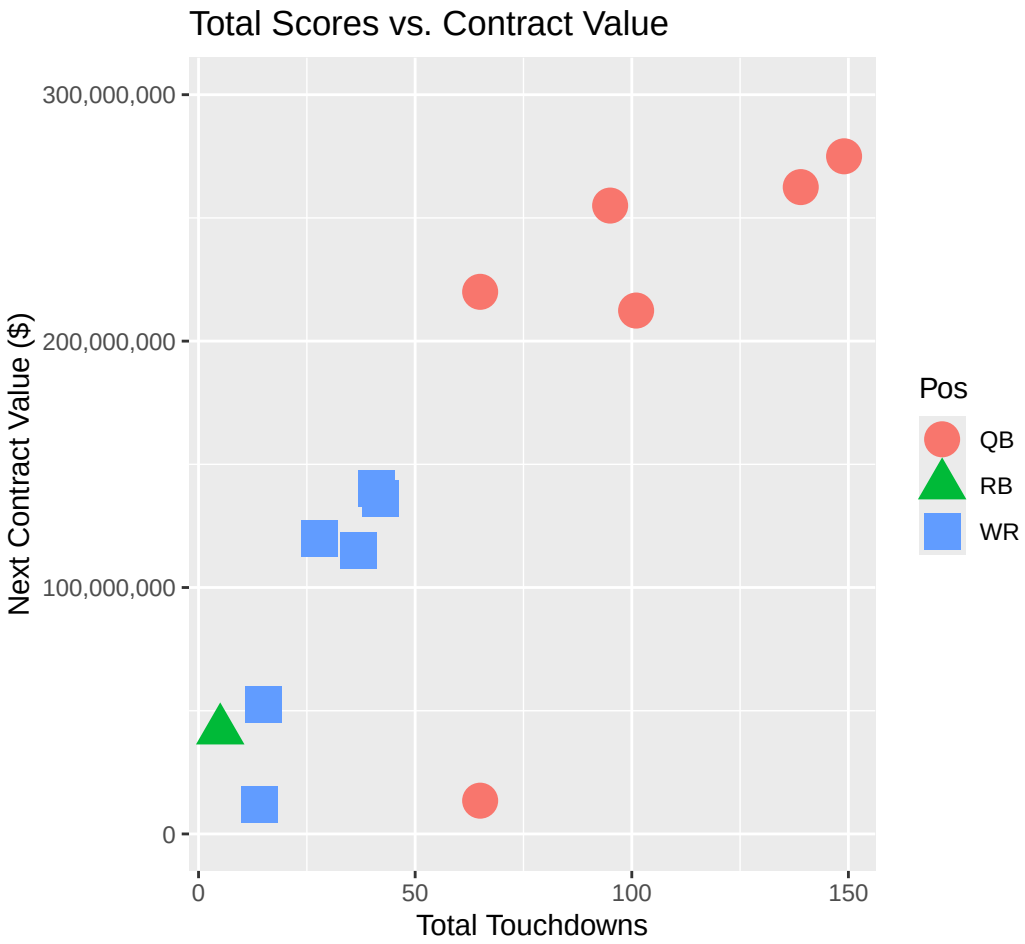


Figure 4: Number of Touchdowns vs. Salary by Position

Context for this graph: this is data that has been extensively filtered to match specific cases I wanted to isolate. Unlike other visualizations that utilized the entire set of data, I filtered these only to match players who have signed a contract extension sometime after their initial draft in 2020. In other words, these are players that make it through the jump to the professional level of the sport. So, due to these conditions (skill positions minus TE, drafted in 2020, and re-signed sometime before 2025) the sample size for this visualization is small, but provides solid cases that should help in further analysis.

This is a visualization that attempts to put the skill positions on a common, level playing field, that is the total amount of scores (touchdowns) they achieve. However, this "even playing field" is about as even and level as the Rocky Mountains, as not every skill position scores as often as the others do. But, there is still important information we can extract from this visualization. We can see that the successful QBs in this plot both score and are paid more

than their offensive counterparts, both WRs and RBs.

Table 1: Dollar Per Score By Skill Position

Pos	Touchdowns	Total Salary	Dollars per Touchdown
QB	614 (77.14%)	1,238,400,000 (66.73%)	2,016,938 (14.76%)
RB	5 (0.63%)	42,000,000 (2.26%)	8,400,000 (61.46%)
WR	177 (22.24%)	575,390,000 (31.01%)	3,250,791 (23.78%)
Total	796 (100.00%)	1,855,790,000 (100.00%)	13,667,729 (100.00%)

This is a summary table based upon the visualization above (Figure 4). Using the same sample size, we can look at the makeup of their salaries, total touchdowns, and the relation between them. QBs are scoring the vast majority of touchdowns, making up more than three quarters of the sample data. However, this table breeds a very important relation that is applicable across all point scorers, which is Dollars per Touchdown. QBs, on average, will make about 2 million dollars per touchdown they score, and WRs about 3 and a quarter million per score. But RBs get a little interesting. The running back position in the NFL has been a very controversial position in regards to payment. They are the most injury-prone, and have some of the lowest pay in the league for their troubles. Consulting the Position vs. Salary graph (Figure 2), we can see that they have some of the lowest median and mean pay for their position group. However, when evaluated along the same criteria I have been using, we can see that they will be paid a whopping 8.4 million per touchdown that they score. So we see here that RBs might not score the most often or get paid frequently, but when they score, it is apparent that it means a lot more in terms of monetary value.

Comparing Sports

All three leagues we looked at have at least one player statistic that has a strong relationship with the average salary of their next contract. This shows that it is possible to predict the value of a player's contract only with one or two statistics. This is important to recognize for players so they can ensure they are being paid adequately.

For the NFL, the main factors that play into a rookie player's salary and contract value are draft positioning, and if they are a point scorer or not. Draft positioning (Figure 1) proves the first point among what factors affect a player's contract value. The second, according to player position (Figure 2), that we see is that point scorers, especially quarterbacks, get the bag at an incredibly disproportionate level.

For the NHL, the main factor that impacted a player's salary was points per game. This makes sense because if a player is scoring or assisting a lot, they are valuable and teams will want to keep them. This makes the team willing to pay them more. Another factor that impacted a player's salary was games played. This also makes sense because if a coach is choosing to play someone, it is likely because they are good which means they get paid more.

For the NBA, the main predictor of salary after a player's rookie contract is points per game (PPG). It is not required for players to average a lot of points for a rookie, which is usually around 15+ PPG, but every player who did meet that mark signed a near max contract (30+ million) this year. The data also suggests that scoring above 10 PPG as a rookie is a good way to make at least 10 million on your next deal.

After comparison between all three sports, we can identify that the one commonality is how they score. Whether its points per game or touchdowns, the main idea is if you score you get paid. As much as draft "stock" and positioning play into a player's initial rookie contract, their prolonged success (and consequently payment and contract value) is determined by how efficient they are at scoring.

Citations

"Statistics." NHL.Com, www.nhl.com/stats/. Accessed 7 May 2025.

NHL | Spotrac.Com, www.spotrac.com/nhl. Accessed 8 May 2025.

"NFL Contract Extensions." Spotrac.Com, www.spotrac.com/nfl/contracts/extensions. Accessed 8 May 2025.

"2020 NFL Draft Listing." Pro Football Reference, www.pro-football-reference.com/years/2020/draft.html. Accessed 7 May 2025.

"NFLREADR • Download NFLVERSE Data." NFLreadr, nflreadr.nflverse.com/. Accessed 7 May 2025.

Code Apendix

```
#loading the necessary packages
library(tidyverse)
library(rvest)
library(dplyr)
library(ggplot2)
library(readxl)

#importing data
```

```
nhl_draft_2015 <- read_excel(
  "/Users/ethanmartin/Downloads/nhl_draft_2015.xlsx"
  )
nhl_draft_2016 <- read_excel(
  "/Users/ethanmartin/Downloads/nhl_draft_2016.xlsx"
  )
rookies_2015_to_2017 <- read_excel(
  "/Users/ethanmartin/Downloads/rookies_2015_to_2017.xlsx"
  )
rookies_2015_to_2018 <- read_excel(
  "/Users/ethanmartin/Downloads/rookies_2015_to_2018.xlsx"
  )
rookies_2015_to_2019 <- read_excel(
  "/Users/ethanmartin/Downloads/rookies_2015_to_2019.xlsx"
  )
rookies_2015_to_2020 <- read_excel(
  "/Users/ethanmartin/Downloads/rookies_2015_to_2020.xlsx"
  )
rookies_2016_to_2018 <- read_excel(
  "/Users/ethanmartin/Downloads/rookies_2016_to_2018.xlsx"
  )
rookies_2016_to_2019 <- read_excel(
  "/Users/ethanmartin/Downloads/rookies_2016_to_2019.xlsx"
  )
rookies_2016_to_2020 <- read_excel(
  "/Users/ethanmartin/Downloads/rookies_2016_to_2020.xlsx"
  )
rookies_2016_to_2021 <- read_excel(
  "/Users/ethanmartin/Downloads/rookies_2016_to_2021.xlsx"
  )

#filtering the data
rookies2015to2017 <- nhl_draft_2015 %>%
  filter(`End of Rookie Contract` == 2017)

rookies2015to2018 <- nhl_draft_2015 %>%
  filter(`End of Rookie Contract` == 2018)

rookies2015to2019 <- nhl_draft_2015 %>%
  filter(`End of Rookie Contract` == 2019)

rookies2015to2020 <- nhl_draft_2015 %>%
```

```
  filter(`End of Rookie Contract` == 2020)

rookies2016to2018 <- nhl_draft_2016 %>%
  filter(`End of Rookie Contract` == 2018)

rookies2016to2019 <- nhl_draft_2016 %>%
  filter(`End of Rookie Contract` == 2019)

rookies2016to2020 <- nhl_draft_2016 %>%
  filter(`End of Rookie Contract` == 2020)

rookies2016to2021 <- nhl_draft_2016 %>%
  filter(`End of Rookie Contract` == 2021)

#merges the filtered data with player stats
rookieStats2015to2017 <- left_join(
  x = rookies2015to2017,
  y = rookies_2015_to_2017,
  by = join_by(Player == Player)
) %>%
  drop_na() %>%
  mutate(S = as.numeric(S))

rookieStats2015to2018 <- left_join(
  x = rookies2015to2018,
  y = rookies_2015_to_2018,
  by = join_by(Player == Player)
) %>%
  drop_na() %>%
  mutate(S = as.numeric(S))

rookieStats2015to2019 <- left_join(
  x = rookies2015to2019,
  y = rookies_2015_to_2019,
  by = join_by(Player == Player)
) %>%
  drop_na() %>%
  mutate(S = as.numeric(S))

rookieStats2015to2020 <- left_join(
  x = rookies2015to2020,
  y = rookies_2015_to_2020,
```

```
  by = join_by(Player == Player)
) %>%
  drop_na() %>%
  mutate(S = as.numeric(S))

rookieStats2016to2018 <- left_join(
  x = rookies2016to2018,
  y = rookies_2016_to_2018,
  by = join_by(Player == Player)
) %>%
  drop_na() %>%
  mutate(S = as.numeric(S))

rookieStats2016to2019 <- left_join(
  x = rookies2016to2019,
  y = rookies_2016_to_2019,
  by = join_by(Player == Player)
) %>%
  drop_na() %>%
  mutate(S = as.numeric(S))

rookieStats2016to2020 <- left_join(
  x = rookies2016to2020,
  y = rookies_2016_to_2020,
  by = join_by(Player == Player)
) %>%
  drop_na() %>%
  mutate(S = as.numeric(S))

rookieStats2016to2021 <- left_join(
  x = rookies2016to2021,
  y = rookies_2016_to_2021,
  by = join_by(Player == Player)
) %>%
  drop_na() %>%
  mutate(S = as.numeric(S))

#combines all data sets and generalizes the positions
all_rookie_stats <- bind_rows(
  rookieStats2015to2017,
  rookieStats2015to2018,
  rookieStats2015to2019,
```

```
  rookieStats2015to2020,
  rookieStats2016to2018,
  rookieStats2016to2019,
  rookieStats2016to2020,
  rookieStats2016to2021
) %>%
  select(-Pos) %>%
  mutate(Position = ifelse(Position == "C", "Forward", Position)) %>%
  mutate(Position = ifelse(Position == "LW", "Forward", Position)) %>%
  mutate(Position = ifelse(Position == "RW", "Forward", Position)) %>%
  mutate(Position = ifelse(Position == "D", "Defenseman", Position))

#creates points per game graph
ppg_graph <- ggplot(
  all_rookie_stats, aes(x = `P/GP`, y = `Average Salary of Next Contract`)
  ) +
  geom_point(size = 3) +
  geom_smooth(method = "loess", se = FALSE) +
  scale_y_continuous(labels = function(x) paste0(x / 1e6, " million")) +
  labs(
    title = "Points/Game vs. Average Salary of Their Next Contract"
  )

#creates games played graph
gp_graph <- ggplot(all_rookie_stats, aes(
  x = `GP`, y = `Average Salary of Next Contract`)
  ) +
  geom_point(size = 3) +
  geom_smooth(method = "loess", se = FALSE) +
  scale_y_continuous(labels = function(x) paste0(x / 1e6, " million")) +
  labs(
    title = "Games Played vs. Average Salary of Their Next Contract"
  )
ppg_graph
gp_graph
library(rvest)
library(tidyverse)
library(dplyr)

# Scrapes 2024-2025 regular season player stats from basketball reference
LeagueStatsWebPage <- read_html(
  "https://www.basketball-reference.com/leagues/NBA_stats_per_game.html#stats-Regular-Season
```

```
) %>% html_elements(css = "table") %>%
  html_table()

LeagueStatsRaw <- LeagueStatsWebPage[[1]]

# Converts names of columns
colnames(LeagueStatsRaw) <- c("Rank", "Season", "Lg", "Age", "Height", "Weight", "Games", "M

# Function to convert Season to Year format
convert_year <- function(year) {
  yearEnd <- substr(year, 6, 7)
  yearStart <- substr(year, 1, 2)
  if (year == "1999-00") {
    newYear <- paste("20", sep = "", yearEnd)
  } else {
    newYear <- paste(yearStart, sep = "", yearEnd)
  }
  return(newYear)
}

# Removes all but Year, 3PA, and 3P% and removes leftover column name rows. Also filters out
LeagueStatsCleaned <- select(LeagueStatsRaw, "Season", "ThreePA", "ThreePercent") %>%
  filter(Season != "Season") %>%
  filter(ThreePercent != "Shooting") %>%
  rowwise() %>%
  mutate(
    Year = convert_year(Season)
  ) %>%
  filter(Season > 1979)

# Makes char type data numeric
LeagueStatsCleaned <- mutate(LeagueStatsCleaned, ThreePA = as.numeric(ThreePA))
LeagueStatsCleaned <- mutate(LeagueStatsCleaned, Year = as.numeric(Year))

# Graphs league average three point attempts per game
ggplot(
  data = LeagueStatsCleaned,
  mapping = aes(
  x = Year,
  y = ThreePA
  )
) + geom_line() +
```

```
  theme_bw() +
  labs(
    title = "3-Point Attempts Over Time",
    x = 'Year',
    y = 'Average Three Pointers Attempted Per Game'
  )

library(rvest)
library(tidyverse)
library(dplyr)
library(scales)


#Takes the 2024-2025 players' salary information from basketball reference
SalaryWebPage <- read_html(
  "https://www.basketball-reference.com/contracts/players.html#player-contracts"
) %>% html_elements(css = "table") %>%
  html_table()

SalaryRaw <- SalaryWebPage[[1]]

# Renames the columns
colnames(SalaryRaw) <- c(
  "Rank", "Player", "Team", "Salary", "Salary1", "Salary2", "Salary3", "Salary4", "Salary5",

# Removes undesired tables
SalaryCleaned <- select(SalaryRaw, -"Salary1", -"Salary2", -"Salary3", -"Salary4", -"Salary5"
  filter(Rank != "Rk") %>%
  filter(Salary != "Salary")

SalaryCleaned$Salary = as.numeric(gsub("[\\$,]", "", SalaryCleaned$Salary))

#------------------------------------------------------------------------

# Takes the 2024-2025 players regular season per game stats from basketball reference
StatsWebPage <- read_html(
  "https://www.basketball-reference.com/leagues/NBA_2025_per_game.html#per_game_stats"
) %>% html_elements(css = "table") %>%
  html_table()

StatsRaw <- StatsWebPage[[1]]
```

```
# Drops awards table
StatsCleaned <- select(StatsRaw, -"Awards")

#--------------------------------------------------------------------------

# Joins Stats and Salary tables; drops "2TM" and old teams for traded players
Players <- inner_join(SalaryCleaned, StatsCleaned)

# Graphs
ggplot(
  data = Players,
  mapping = aes(
  x = `3P%`,
  y = Salary,
  color = `Pos`
  )
) + geom_point() +
  scale_y_continuous(
    labels = label_currency(),
    n.breaks = 6
  ) +
  theme_bw() +
  labs(
    title = '3-Point Percent vs Salary',
    x = 'Three-Point Percentage',
    y = 'Salary (USD)'
  )

library(rvest)
library(tidyverse)
library(dplyr)

# Reads 2019-2020 rookie data from basketball reference
RookiesWebPage <- read_html(
  "https://www.basketball-reference.com/leagues/NBA_2020_rookies-season-stats.html#rookies"
) %>% html_elements(css = "table") %>%
  html_table()

RookiesRaw <- RookiesWebPage[[1]]

# Renames columns
colnames(RookiesRaw) <- c(
```

```
  "Rank", "Player", "Debut", "Age", "EXP", "GP", "MP", "FG", "FGA", "ThreePM", "ThreePA", "T
)

# Removes all columns except Player, PPG, and ThreePercent
RookiesCleaned <- select(RookiesRaw, "Player", "PPG", "ThreePercent") %>%
  filter(PPG != "Per Game") %>%
  filter(Player != "Player")

#-------------------------------------------------------------------------------

#Takes the 2024-2025 players' salary information from basketball reference
SalaryWebPage <- read_html(
  "https://www.basketball-reference.com/contracts/players.html#player-contracts"
) %>% html_elements(css = "table") %>%
  html_table()

SalaryRaw <- SalaryWebPage[[1]]

#Renames the columns
colnames(SalaryRaw) <- c(
  "Rank", "Player", "Team", "Salary", "Salary1", "Salary2", "Salary3", "Salary4", "Salary5",

SalaryCleaned2 <- select(SalaryRaw, "Player", "Salary") %>%
  filter(Player != "Player") %>%
  filter(Salary != "Salary")

# Formats char type salary info as numeric without dollar sign or commas
SalaryCleaned2$Salary = as.numeric(gsub("[\\$,]", "", SalaryCleaned2$Salary))

#-------------------------------------------------------------------------------

# Joins SalaryCleaned2 and RookiesCleaned dataframes
RookiesSalaryCleaned <- inner_join(SalaryCleaned2, RookiesCleaned)

# Changes PPG from char type to numeric
RookiesSalaryCleaned <- mutate(RookiesSalaryCleaned, PPG = as.numeric(PPG))

# Graphs PPG of rookie year vs 2024-2025 salary
ggplot(
  data = RookiesSalaryCleaned,
  mapping = aes(
  x = PPG,
```

```
  y = Salary
  )
) + geom_point() +
  scale_y_continuous(
    labels = label_currency(),
    n.breaks = 6
  ) +
  theme_bw() +
  labs(
    title = "Rookie PPG vs First Contract Salary",
    x = 'Points Per Game',
    y = 'Salary (USD)'
  )
# Packages ----
# install.packages("rvest")
# install.packages("tidyverse")
# install.packages("googlesheets4")
# install.packages("nflreadr")
# install.packages("ggplot2")
# install.packages("dplyr")
# install.packages("hrbrthemes")
# install.packages("viridis")
# install.packages("janitor")
# install.packages("knitr")
# install.packages("kableExtra")
# install.packages("ggpmisc")
# install.packages("tinytex")
library(nflreadr)
library(rvest)
library(tidyverse)
library(googlesheets4)
library(ggplot2)
library(dplyr)
library(hrbrthemes)
library(viridis)
library(data.table)
library(janitor)
library(knitr)
library(kableExtra)
library(ggpmisc)
library(tinytex)
library(chromote)
```

```r
# Scraping ----
# NFL signing and salary table
nfl_salary_raw <- read_html_live("https://www.spotrac.com/nfl/contracts") %>%
  html_elements(css = "table") %>%
  html_table()
nfl_salary <- nfl_salary_raw[[2]]

# NFL contract extensions since 2021
extensions_2021_raw <- read_html_live("https://www.spotrac.com/nfl/contracts/extensions/_/yea
  html_elements(css = "table") %>%
  html_table()
extensions_2021 <- extensions_2021_raw[[2]]

extensions_2022_raw <- read_html_live("https://www.spotrac.com/nfl/contracts/extensions/_/yea
  html_elements(css = "table") %>%
  html_table()
extensions_2022 <- extensions_2022_raw[[2]]

extensions_2023_raw <- read_html_live("https://www.spotrac.com/nfl/contracts/extensions/_/yea
  html_elements(css = "table") %>%
  html_table()
extensions_2023 <- extensions_2023_raw[[2]]

extensions_2024_raw <- read_html_live("https://www.spotrac.com/nfl/contracts/extensions/_/yea
  html_elements(css = "table") %>%
  html_table()
extensions_2024 <- extensions_2024_raw[[2]]

extensions_2025_raw <- read_html_live("https://www.spotrac.com/nfl/contracts/extensions/_/yea
  html_elements(css = "table") %>%
  html_table()
extensions_2025 <- extensions_2025_raw[[2]]
names(extensions_2021)[2] <- "Player"
names(extensions_2022)[2] <- "Player"
names(extensions_2023)[2] <- "Player"
names(extensions_2024)[2] <- "Player"
names(extensions_2025)[2] <- "Player"
extensions_all <- bind_rows(
  list(extensions_2021, extensions_2022, extensions_2023, extensions_2024, extensions_2025)
)
# NFL rookie contract "signing scale" for the 2020 draft year by pick
rookie_scale_raw <- read_html_live("https://www.spotrac.com/nfl/cba/rookie-scale/_/year/2020
```

```
  html_elements(css = "table") %>%
  html_table()
rookie_scale <- rookie_scale_raw[[1]]
rookie_scale$`Total Value` = as.numeric(gsub("[\\$,]","", rookie_scale$`Total Value`))
rookie_scale$`Year 1` = as.numeric(gsub("[\\$,]","", rookie_scale$`Year 1`))
rookie_scale$`Year 2` = as.numeric(gsub("[\\$,]","", rookie_scale$`Year 2`))
rookie_scale$`Year 3` = as.numeric(gsub("[\\$,]","", rookie_scale$`Year 3`))
rookie_scale$`Year 4` = as.numeric(gsub("[\\$,]","", rookie_scale$`Year 4`))
# 2020 NFL draftee college stats
college_rookie_stats_raw <- read_html_live("https://www.pro-football-reference.com/years/2020
  html_elements(css = "table") %>%
  html_table()
college_rookie_stats <- college_rookie_stats_raw[[1]]
header_names <- as.character(college_rookie_stats[1, ])
header_names[29] <- "dropME" # MAKE SURE TO DROP LAST COL
for (i in 1:29){
  if (i == 26) {
    header_names[i] <- paste0("defense_", header_names[i])
  } else if (between(i, 14, 18)) {
    header_names[i] <- paste0("passing_", header_names[i])
  } else if (between(i, 19, 21)) {
    header_names[i] <- paste0("rushing_", header_names[i])
  } else if (between(i, 22, 24)) {
    header_names[i] <- paste0("receiving_", header_names[i])
  } else {
    next
  }
}
colnames(college_rookie_stats) <- header_names
college_rookie_stats <- college_rookie_stats %>%
  filter(Rnd != "Rnd")
college_rookie_stats[[11]] <- as.integer(college_rookie_stats[[11]])
college_rookie_stats[[2]] <- as.integer(college_rookie_stats[[2]])

# 2020-2024 NFL draftee stats per year

player_stats_2020 <- load_player_stats(2020)
colnames(player_stats_2020)[3] <- "Player"
player_stats_2021 <- load_player_stats(2021)
colnames(player_stats_2021)[3] <- "Player"
player_stats_2022 <- load_player_stats(2022)
colnames(player_stats_2022)[3] <- "Player"
```

```r
player_stats_2023 <- load_player_stats(2023)
colnames(player_stats_2023)[3] <- "Player"
player_stats_2024 <- load_player_stats(2024)
colnames(player_stats_2024)[3] <- "Player"

# 2020 NFL draft table with contract
rookie_contracts_raw <- read_html_live("https://www.spotrac.com/nfl/draft/_/year/2020/sort/pi
  html_elements(css = "table") %>%
  html_table()
rookie_contracts <- rookie_contracts_raw[[1]]

# Tidying ----
# For rookies only, filter player stats 2020-2024
rookie_stats_2020 <- semi_join(player_stats_2020, college_rookie_stats, by = "Player")
rookie_stats_2021 <- semi_join(player_stats_2021, college_rookie_stats, by = "Player")
rookie_stats_2022 <- semi_join(player_stats_2022, college_rookie_stats, by = "Player")
rookie_stats_2023 <- semi_join(player_stats_2023, college_rookie_stats, by = "Player")
rookie_stats_2024 <- semi_join(player_stats_2024, college_rookie_stats, by = "Player")

# Skill Position wrangling ----
## QB
rookie_stats_2020_qb <- rookie_stats_2020 %>%
  filter(position == "QB") %>%
  group_by(Player) %>%
  summarise(total_yds = sum(passing_yards),
            comp = sum(completions),
            att = sum(attempts),
            yds = sum(passing_air_yards),
            passing_td = sum(passing_tds),
            ints = sum(interceptions),
            rushes = sum(carries),
            rush_yds = sum(rushing_yards),
            )
rookie_stats_2021_qb <- rookie_stats_2021 %>%
  filter(position == "QB") %>%
  group_by(Player) %>%
  summarise(total_yds = sum(passing_yards),
            comp = sum(completions),
            att = sum(attempts),
            yds = sum(passing_air_yards),
            passing_td = sum(passing_tds),
            ints = sum(interceptions),
```

```
            rushes = sum(carries),
            rush_yds = sum(rushing_yards),
  )
rookie_stats_2022_qb <- rookie_stats_2022 %>%
  filter(position == "QB") %>%
  group_by(Player) %>%
  summarise(total_yds = sum(passing_yards),
            comp = sum(completions),
            att = sum(attempts),
            yds = sum(passing_air_yards),
            passing_td = sum(passing_tds),
            ints = sum(interceptions),
            rushes = sum(carries),
            rush_yds = sum(rushing_yards),
  )
rookie_stats_2023_qb <- rookie_stats_2023 %>%
  filter(position == "QB") %>%
  group_by(Player) %>%
  summarise(total_yds = sum(passing_yards),
            comp = sum(completions),
            att = sum(attempts),
            yds = sum(passing_air_yards),
            passing_td = sum(passing_tds),
            ints = sum(interceptions),
            rushes = sum(carries),
            rush_yds = sum(rushing_yards),
  )
rookie_stats_2024_qb <- rookie_stats_2024 %>%
  filter(position == "QB") %>%
  group_by(Player) %>%
  summarise(total_yds = sum(passing_yards),
            comp = sum(completions),
            att = sum(attempts),
            yds = sum(passing_air_yards),
            passing_td = sum(passing_tds),
            ints = sum(interceptions),
            rushes = sum(carries),
            rush_yds = sum(rushing_yards),
  )
total_rookie_stats_qb <- bind_rows(
  list(rookie_stats_2020_qb, rookie_stats_2021_qb, rookie_stats_2022_qb, rookie_stats_2023_q
  group_by(Player) %>%
```

```
  summarise(total_yds = sum(total_yds),
            comp = sum(comp),
            att = sum(att),
            yds = sum(yds),
            tds = sum(passing_td),
            ints = sum(ints),
            rushes = sum(rushes),
            rush_yds = sum(rush_yds),
            qbr = (((((sum(comp) / sum(att)) - 0.3) * 5) +
                    (((sum(yds)/sum(att)) - 3) * 0.25) +
                    ((sum(passing_td)/sum(att) * 20)) +
                    (2.375 - (sum(ints)/sum(att)) * 25)
                    ) / 6) * 100
  )

# 2020 Draft Class extensions
rookie_extensions <- semi_join(extensions_all, college_rookie_stats, by = "Player")
extensions_qb <- rookie_extensions %>% filter(Pos == "QB")
extensions_wr <- rookie_extensions %>% filter(Pos == "WR")
extensions_rb <- rookie_extensions %>% filter(Pos == "RB")
extensions_te <- rookie_extensions %>% filter(Pos == "TE")
extensions_db <- rookie_extensions %>% filter(Pos == "CB" | Pos == "S")
extensions_ol <- rookie_extensions %>% filter(Pos == "LT" | Pos == "RT" | Pos == "G" | Pos ==
extensions_dl <- rookie_extensions %>% filter(Pos == "DT")
extensions_lb <- rookie_extensions %>% filter(Pos == "ILB" | Pos == "OLB")
extensions_st <- rookie_extensions %>% filter(Pos == "LS" | Pos == "K")

total_rookie_stats_qb_salary <- inner_join(
  x = total_rookie_stats_qb,
  y = extensions_qb,
  by = join_by(Player == Player)
)
total_rookie_stats_qb_salary$`Value` = as.numeric(gsub("[\\$,]","", total_rookie_stats_qb_sal
## RB
rookie_stats_2020_rb <- rookie_stats_2020 %>%
  filter(position == "RB") %>%
  group_by(Player) %>%
  summarise(rushes = sum(carries),
            rush_yds = sum(rushing_yards),
            rush_tds = sum(rushing_tds),
            fumbles = sum(rushing_fumbles_lost, receiving_fumbles_lost),
            receptions = sum(receptions),
```

```
            rec_yds = sum(receiving_yards),
            rec_tds = sum(receiving_tds),
            total_yds = sum(rushing_yards, receiving_yards)
  )
rookie_stats_2021_rb <- rookie_stats_2021 %>%
  filter(position == "RB") %>%
  group_by(Player) %>%
  summarise(rushes = sum(carries),
            rush_yds = sum(rushing_yards),
            rush_tds = sum(rushing_tds),
            fumbles = sum(rushing_fumbles_lost, receiving_fumbles_lost),
            receptions = sum(receptions),
            rec_yds = sum(receiving_yards),
            rec_tds = sum(receiving_tds),
            total_yds = sum(rushing_yards, receiving_yards)
  )
rookie_stats_2022_rb <- rookie_stats_2022 %>%
  filter(position == "RB") %>%
  group_by(Player) %>%
  summarise(rushes = sum(carries),
            rush_yds = sum(rushing_yards),
            rush_tds = sum(rushing_tds),
            fumbles = sum(rushing_fumbles_lost, receiving_fumbles_lost),
            receptions = sum(receptions),
            rec_yds = sum(receiving_yards),
            rec_tds = sum(receiving_tds),
            total_yds = sum(rushing_yards, receiving_yards)
  )
rookie_stats_2023_rb <- rookie_stats_2023 %>%
  filter(position == "RB") %>%
  group_by(Player) %>%
  summarise(rushes = sum(carries),
            rush_yds = sum(rushing_yards),
            rush_tds = sum(rushing_tds),
            fumbles = sum(rushing_fumbles_lost, receiving_fumbles_lost),
            receptions = sum(receptions),
            rec_yds = sum(receiving_yards),
            rec_tds = sum(receiving_tds),
            total_yds = sum(rushing_yards, receiving_yards)
  )
rookie_stats_2024_rb <- rookie_stats_2024 %>%
  filter(position == "RB") %>%
```

```
  group_by(Player) %>%
  summarise(rushes = sum(carries),
            rush_yds = sum(rushing_yards),
            rush_tds = sum(rushing_tds),
            fumbles = sum(rushing_fumbles_lost, receiving_fumbles_lost),
            receptions = sum(receptions),
            rec_yds = sum(receiving_yards),
            rec_tds = sum(receiving_tds),
            total_yds = sum(rushing_yards, receiving_yards)
  )
total_rookie_stats_rb <-bind_rows(
  list(rookie_stats_2020_rb, rookie_stats_2021_rb, rookie_stats_2022_rb, rookie_stats_2023_rb
)  %>%
  group_by(Player) %>%
  summarise(rushes = sum(rushes),
            rush_yds = sum(rush_yds),
            rush_tds = sum(rush_tds),
            fumbles = sum(fumbles),
            receptions = sum(receptions),
            rec_yds = sum(rec_yds),
            tds = sum(rec_tds),
            total_yds = sum(total_yds)
  )
total_rookie_stats_rb_salary <- inner_join(
  x = total_rookie_stats_rb,
  y = extensions_rb,
  by = join_by(Player == Player)
)
total_rookie_stats_rb_salary$`Value` = as.numeric(gsub("[\\$,]","", total_rookie_stats_rb_sal
## WR
rookie_stats_2020_wr <- rookie_stats_2020 %>%
  filter(position == "WR") %>%
  group_by(Player) %>%
  summarise(catches = sum(receptions),
            rec_yds = sum(receiving_yards),
            rec_td = sum(receiving_tds),
            fumbles = sum(receiving_fumbles_lost, rushing_fumbles_lost),
            yac = sum(receiving_yards_after_catch),
            rush_yds = sum(rushing_yards),
            rush_tds = sum(rushing_tds),
            total_yds = sum(receiving_yards, rushing_yards),
            total_tds = sum(receiving_tds, rushing_tds),
```

```
  )
rookie_stats_2021_wr <- rookie_stats_2021 %>%
  filter(position == "WR") %>%
  group_by(Player) %>%
  summarise(catches = sum(receptions),
            rec_yds = sum(receiving_yards),
            rec_td = sum(receiving_tds),
            fumbles = sum(receiving_fumbles_lost, rushing_fumbles_lost),
            yac = sum(receiving_yards_after_catch),
            rush_yds = sum(rushing_yards),
            rush_tds = sum(rushing_tds),
            total_yds = sum(receiving_yards, rushing_yards),
            total_tds = sum(receiving_tds, rushing_tds),
  )
rookie_stats_2022_wr <- rookie_stats_2022 %>%
  filter(position == "WR") %>%
  group_by(Player) %>%
  summarise(catches = sum(receptions),
            rec_yds = sum(receiving_yards),
            rec_td = sum(receiving_tds),
            fumbles = sum(receiving_fumbles_lost, rushing_fumbles_lost),
            yac = sum(receiving_yards_after_catch),
            rush_yds = sum(rushing_yards),
            rush_tds = sum(rushing_tds),
            total_yds = sum(receiving_yards, rushing_yards),
            total_tds = sum(receiving_tds, rushing_tds),
  )
rookie_stats_2023_wr <- rookie_stats_2023 %>%
  filter(position == "WR") %>%
  group_by(Player) %>%
  summarise(catches = sum(receptions),
            rec_yds = sum(receiving_yards),
            rec_td = sum(receiving_tds),
            fumbles = sum(receiving_fumbles_lost, rushing_fumbles_lost),
            yac = sum(receiving_yards_after_catch),
            rush_yds = sum(rushing_yards),
            rush_tds = sum(rushing_tds),
            total_yds = sum(receiving_yards, rushing_yards),
            total_tds = sum(receiving_tds, rushing_tds),
  )
rookie_stats_2024_wr <- rookie_stats_2024 %>%
  filter(position == "WR") %>%
```

```
  group_by(Player) %>%
  summarise(catches = sum(receptions),
            rec_yds = sum(receiving_yards),
            rec_td = sum(receiving_tds),
            fumbles = sum(receiving_fumbles_lost, rushing_fumbles_lost),
            yac = sum(receiving_yards_after_catch),
            rush_yds = sum(rushing_yards),
            rush_tds = sum(rushing_tds),
            total_yds = sum(receiving_yards, rushing_yards),
            total_tds = sum(receiving_tds, rushing_tds)
  )
total_rookie_stats_wr <-bind_rows(
  list(rookie_stats_2020_wr, rookie_stats_2021_wr, rookie_stats_2022_wr, rookie_stats_2023_wr
) %>%
  group_by(Player) %>%
  summarise(catches = sum(catches),
            rec_yds = sum(rec_yds),
            rec_td = sum(rec_td),
            fumbles = sum(fumbles),
            yac = sum(yac),
            rush_yds = sum(rush_yds),
            rush_tds = sum(rush_tds),
            total_yds = sum(total_yds),
            tds = sum(total_tds)
  )
total_rookie_stats_wr_salary <- inner_join(
  x = total_rookie_stats_wr,
  y = extensions_wr,
  by = join_by(Player == Player)
)
total_rookie_stats_wr_salary$`Value` = as.numeric(gsub("[\\$,]","", total_rookie_stats_wr_sal

# Visualizations ----
# Contract signing scale
marks_no_sci <- function(x) format(x, big.mark = ",", decimal.mark = ",", scientific = FALSE)
draft_contract_scale <- ggplot(rookie_scale, aes(x = Pick, y = `Total Value` )) +
  geom_line() +
  scale_y_continuous(labels = marks_no_sci) +
  coord_cartesian(ylim = c(0, 40000000)) +
  ylab("Total Value in Dollars ($)") +
  scale_x_continuous(breaks = seq(0, 255, by = 32)) +
  geom_vline(xintercept = 32, color = "orange", linewidth = 1, alpha = 0.4) +
```

```
  geom_vline(xintercept = 64, color = "orange", linewidth = 1, alpha = 0.4) +
  geom_vline(xintercept = 106, color = "orange", linewidth = 1, alpha = 0.4)

## Position vs. average contract value
player_contracts <- load_contracts()
player_contracts[[7]] <- as.integer(player_contracts[[7]])
player_contracts <- player_contracts %>%
  filter(year_signed >= 2020) %>%
  filter(value > 0)
pos_vs_contract <- ggplot(player_contracts, aes(x = position, y = value, fill = position)) +
  geom_point() +
  xlab("Player Position") +
  ylab("Contract Value (in millions)") +
  guides(fill = FALSE)

# Draft pick vs. weighted approximate value
pick_vs_wav <- ggplot(data = college_rookie_stats, aes(x = Pick, y = wAV)) +
  geom_point() +
  geom_smooth() +
  labs(x = "Draft Pick", y = "Weighted Approximate Value") +
  scale_x_continuous(breaks = seq(0, 257, by = 32))

# Skill position scores vs. salary
stats_salary <- bind_rows(
  list(total_rookie_stats_qb_salary, total_rookie_stats_rb_salary, total_rookie_stats_wr_sala
)
stats_salary_plot <- ggplot(data = stats_salary,
       aes(x = tds, y = Value, shape = Pos, color = Pos)) +
  geom_point(size = 6) +
  labs(x = "Total Touchdowns", y = "Next Contract Value ($)") +
  coord_cartesian(ylim = c(0, 300000000)) +
  scale_y_continuous(labels = marks_no_sci)

dollar_per_td_by_pos <- stats_salary %>%
  group_by(Pos) %>%
  summarise(Touchdowns = sum(tds),
            `Total Salary` = sum(Value),
            `td/dollar` = sum(Value)/sum(tds)
  ) %>%
  adorn_totals(where = c("row")) %>%
  adorn_percentages(denominator = "col") %>%
  adorn_pct_formatting(digits = 2)
```

```r
formatNs <- attr(dollar_per_td_by_pos, "core") %>%
  adorn_totals(where = c("row")) %>%
  mutate(
    across(where(is.numeric), .fns = ~format(.x, big.mark = ","))
  )
dollar_per_td_by_pos <- dollar_per_td_by_pos %>%
  adorn_ns(position = "front", ns = formatNs)
dollar_per_td_by_pos %>% kable(
  format = "latex",
  booktabs = TRUE
) %>%
  kable_paper("hover", full_width = T)

marks_no_sci <- function(x) format(x,
                                   big.mark = ",",
                                   decimal.mark = ",",
                                   scientific = FALSE)
draft_contract_scale <- ggplot(rookie_scale,
                               aes(x = Pick, y = `Total Value` )) +
  geom_line() +
  scale_y_continuous(labels = marks_no_sci) +
  coord_cartesian(ylim = c(0, 40000000)) +
  ylab("Total Value in Dollars ($)") +
  scale_x_continuous(breaks = seq(0, 255, by = 32)) +
  geom_vline(xintercept = 32, color = "orange", linewidth = 1, alpha = 0.4) +
  geom_vline(xintercept = 64, color = "orange", linewidth = 1, alpha = 0.4) +
  geom_vline(xintercept = 106, color = "orange", linewidth = 1, alpha = 0.4) +
  labs(title = "Draft Position vs. Contract Value")
draft_contract_scale
player_contracts <- load_contracts()
player_contracts[[7]] <- as.integer(player_contracts[[7]])
player_contracts <- player_contracts %>%
  filter(year_signed >= 2020) %>%
  filter(value > 0)
pos_vs_contract <- ggplot(player_contracts,
                          aes(x = position, y = value, fill = position)) +
  geom_point() +
  xlab("Player Position") +
  ylab("Contract Value (in millions)") +
  guides(fill = FALSE) +
  labs(title = "Player Position vs. Contract Value")
pos_vs_contract
```

```r
pick_vs_wav <- ggplot(data = college_rookie_stats, aes(x = Pick, y = wAV)) +
  geom_point() +
  geom_smooth() +
  labs(x = "Draft Pick", y = "Weighted Approximate Value") +
  scale_x_continuous(breaks = seq(0, 257, by = 32)) +
  labs(title = "Draft Pick vs. Weighted Approximate Value")
pick_vs_wav
stats_salary <- bind_rows(
  list(total_rookie_stats_qb_salary, total_rookie_stats_rb_salary, total_rookie_stats_wr_sala
)
stats_salary_plot <- ggplot(data = stats_salary,
       aes(x = tds, y = Value, shape = Pos, color = Pos)) +
  geom_point(size = 6) +
  labs(x = "Total Touchdowns", y = "Next Contract Value ($)") +
  coord_cartesian(ylim = c(0, 300000000)) +
  scale_y_continuous(labels = marks_no_sci) +
  labs(title = "Total Scores vs. Contract Value")
stats_salary_plot
dollar_per_td_by_pos <- stats_salary %>%
  group_by(Pos) %>%
  summarise(Touchdowns = sum(tds),
            `Total Salary` = sum(Value),
            `Dollars per Touchdown` = sum(Value)/sum(tds)
  ) %>%
  adorn_totals(where = c("row")) %>%
  adorn_percentages(denominator = "col") %>%
  adorn_pct_formatting(digits = 2)
formatNs <- attr(dollar_per_td_by_pos, "core") %>%
  adorn_totals(where = c("row")) %>%
  mutate(
    across(where(is.numeric), .fns = ~format(.x, big.mark = ","))
  )
dollar_per_td_by_pos <- dollar_per_td_by_pos %>%
  adorn_ns(position = "front", ns = formatNs)
dollar_per_td_by_pos %>% kable(
  format = "latex",
  booktabs = TRUE
) %>%
  kable_paper("hover", full_width = T)
```