Midterm Report

More data integration, visualizations, and next steps.

The hardest part of our project thus far has certainly been data extraction. We've run into issues with the availability of data and the amount of time that it would take to scrape data from profootballreference. Salary data has been particularly difficult to find. Spotrac has more complete salary data for recent years but no players have complete salary data for the timespan that we have selected for AV data. Other difficulties arose when trying to scrape player positions and URLs (to serve as id's) from profootballreference. Our initial approach would have taken about 20hrs to complete given the way that data is organized on the website. An alternative approach that we pursued only took 20 minutes but required deeper exploration of the site and collective effort on the part of the team to make things feasible.

Challenges: Some of the greatest challenges that we'll be facing moving forward will be how we utilize transaction and salary data to gain the most insight from relatively sparse salary data. Fortunately we've been able to find near complete salary data for the past 7 years. However, this may affect the type of insights that we want to make using this data.

1. Spotrac.com salary data scraping
2. Assigning player IDs to players
3. Creating a standard for position names

Concrete results/initial insights: Our challenges with data cleaning and integration were plenty, resulting in us diverting our resources to answering how we would approach answering some of the initial questions we proposed in our methodology. In addition to determining the fair value of a player (according to the methodology described in the pre-proposal), one area of interest we turned our attention to is valuing draft picks.

Motivation:

Drafting is the edge every NFL team has. In theory, the amount of value every team gets out of its free agency spending is the same, in that there is a salary cap and we can assume each contract is 'fair.' Drafting, however, is where teams can get cheap labor - i.e. salary is far less than fair salary. Cheap labor only lasts 4-5 years - the length of the rookie contract. We can assume that afterwards, the price the team pays for the player is equal to the fair salary of the player.

This is why draft picks are worth so much - they increase the talent per dollar spent for a team. In other words, a great draft, in theory, should give the team a boost for the next 4-5 years only. Example: The Patriots were docked their first round pick in 2016. They will be feeling this loss until 2021, when that player in theory would have begun their second contract.

- The Seahawks drafted extremely well between 2010 - 2012. This draft gave them a big boost in spending efficiency - i.e. talent per dollar paid until they had to give second contracts to those players. So now that all those players are on their second contracts, the edge they have is now the 2013 or 2014 drafts - which the Seahawks haven't been doing so well on, and thus their performance has dropped correspondingly.
- In theory, we can predict how good a team will be in a given season based on its past 4-5 drafts alone. This is actually something we can look into as a predictor. This, however, operates under the assumption that no team has an edge in free agency (not necessarily true), continuity does not matter (i.e. for culture purposes it might), and coaching is equal across teams (almost definitely not true).

Case Study - Chandler Jones

Chandler Jones was traded for a second round pick (plus Jonathan Cooper, but let's ignore that for simplicity), even though that second round pick would almost certainly not be as good as Chandler Jones (i.e. comparing expected Career AV for that pick with Chandler Jones' expected Career AV). If we consider this trade from the Patriots' perspective, Chandler Jones had just one year left on his rookie deal - i.e. one more year of cheap talent. After the one year, he would have to be compensated fairly, and a second round pick would have four years left on his rookie deal - four years of cheap talent.

Chandler Jones was paid 7.79 million in 2016. Suppose his fair contract value was 13.79 million. Then Chandler Jones offered an additional 6 million dollars above fair value. Over the four years from that second round pick, would the dollars above fair value sum to over 6 million dollars? We can parlay this into a value system for draft picks.

Assumptions:

- A team's goal is to maximize talent per dollar spent.
- Talent per dollar spent is equal for all teams in free agency.
- The value of a draft pick is its dollars above fair value summed over all years of its contract.

For example, suppose using the above, not including draft picks, the Brown's total assets are worth 170 million for 2017. Total assets includes fair value for all the players, plus free salary cap space. Suppose on average, the first overall pick outperforms his rookie contract by 4 million dollars in Year 1, 5 million dollars in Year 2, 6 million dollars in Year 3, 8 million dollars in Year 4, and 4 million dollars in Year 5. Then by using the first overall pick, we expect the Brown's total assets to increase by 4 million to 174 million in 2017, and increase by 5, 6, 8, and 4 million dollars respectively in 2018, 2019, 2020, and 2021. We can then value the first overall pick at 4 + 5 + 6 + 8 + 4 million dollars - or $27 million dollars.

Methodology:

Contract values are relatively fixed for draft picks in terms of both length and value (i.e. a player drafted Round 2, pick 35 will sign a 4 year deal worth roughly 6.38 million), due to constraints in the CBA (Collective Bargaining Agreement) signed in 2011.

For a given unused draft pick: We know the approximate contract values over each year of the rookie contract for that draft pick, and can approximate the fair salaries for each year of the rookie contract based on data for that pick in previous years. Since the salary cap affects fair contract values, and the salary cap is increasing for the NFL, we may need to adjust our fair salaries for projected salary cap increases. We sum the difference between each year's fair salary and actual salary to determine the draft pick's worth.

Completed Steps - Midterm Report

Edan

I worked on two main tasks between the submission of our first blog post and now. The first was scraping salary data from the web, and the second was creating a script that would load all of the data that we collected from various sources and have thus far stored in CSV files into a SQL database that followed the schema that we discussed. These two tasks were relatively open-ended and thus required me to navigate various challenges as I tried to complete them. The steps I took to complete these tasks and the challenges I encountered are outlined below.

1. **Scraping Salary Data**

   I decided to scrape salary data from spotrac.com using BeautifulSoup. Spotrac.com, which despite

only officially displaying salary data for NFL teams since 2011, has some data from prior years that can be accessed by changing the year in the URL. Still, the farther away the years got from 2011, the less players that showed up on each team's payroll - meaning that in the 1990s we only have salary data for a few players from each team. Unfortunately, we weren't able to find any more comprehensive data from those years, so the data that we got from spotrac.com will have to suffice for earlier years as well. Another challenge that I encountered was duplicate removal. This entailed removng players that showed up more than once on the same team as well as players that showed up twice when scraping the data because teams that have had different names in the past could have two different pages on spotrac representing the same franchise - for example, if I put los-angeles-chargers in the URL and later san-diego-chargers in the URL, spotrac would return the same page for certain years, giving me duplicate data for the Chargers in that year. This also meant that I had to make sure that when I got salary data from teams using their old names, that this data was associated with the present-day name of the team. Finally, various players had hyphens in various sections - representing 0s - and I had to be careful when scraping this data with BeautifulSoup so as to not return nulls when I encountered these values.

2. **Loading Data into a Database**

In order to load data into SQL databases, I first had to create tables that mimic the schema that we decided on as a group. Unfortunately, as we continued to scrape data and figure out what information was available to us, this schema, and hence these tables, had to be continuously updated. I also had to make sure the primary keys and foreign keys that we had decided on in our schema fit the data that we had, and I had to declare these when creating each table and update them as the data available to us became clearer. In terms of primary keys, the main issue was coming up with consistent ID's for players, positions, and teams that we could store in our tables. Since we got our data from various sources, this step was crucial for unifying our data. For example, team names, and thus their abbreviations, have changed multiple times between 1993 and 2017. In fact, the team that is currently known as the Los Angeles Rams, with an abbreviation of LAR on pro-football-reference.com, was known as the Los Angeles Rams until 1994 with an abbreviation of RAM on pro-football-reference.com and as the St. Louis Rams between 1994 and 2016 with an abbreviation of STL on pro-football-reference.com. Despite these different names, they still represent the same franchise, and thus I had to ensure that my database loading script changed all older team abbreviations to the abbreviation currently associated with that franchise. I then used the current team abbreviations as the team ID in my team table (which is referenced as a foreign key in various other tables).

We also had to figure out a feasible way to assign player ID's to players. This was especially difficult considering various players in the NFL between 1993 and now have shared the same name - and might have even played on the same team at the same time. Thus, we decided to look at when a player was drafted in order to differentiate between players with the same name. This necessitated the assumption that no two players with the same name were drafted in the same round with the same pick. Since we were able to get the round and pick a player was drafted in when scraping our AV data, this allowed us to easily create unique player ID's for all players drafted between 1993 and now and to make sure that these same ID's were assigned to the same players in the AV table and the draft table. For drafted players, we decided to simply make their player IDs the year they were drafted combined with the pick they were drafted with. Still, this meant that we had various undrafted players/players that were drafted before 1993 in our AV dataset that didn't have player IDs we could assign to them. We handled this by grabbing a unique URL for each player when scraping our AV Data. This allowed us to do the following: for each player in the AV dataset that didn't have a player ID from the draft table, we checked if the unique URL associated with him had a player ID associated with it. If it did, it meant that we had already invented a player ID for that player and we simply assigned it to him. Otherwise, we created a player ID for that player in the same format as the player IDs used for the drafted players, using 0000 as the draft year and

incrementing a counter to assign a unique pick to each undrafted player ID. Finally, we had to make sure that these player IDs were assigned to the right players in our salary data. Unfortunately, we weren't able to find draft information associated with players in our salary data - so in order to assign the correct player IDs to these players, we had to assume that no two players with the same name played for the same team in the same year with the same position. We were then able to use a dictionary that mapped a tuple consisting of player name, team, year, and position to a player ID in order to assign that player ID to the player in the salary data.

This brings us to the final ID we had to figure out - position IDs. There are various positions in football, and these can be broken up into even more specific positions. For instance, the Offensive Line position, which is abbreviate OL, can be broken down into Guard (G) - which can be further broken down into Right Guard (RG) and Left Guard (LG), Tackle (T) - which can be further broken down into Right Tackle (RT) and Left Tackle (LT), and Center (C). Thus, when extracting data from various sources, we had to decide on a consistent way to identify all of these different positions. The final touches to this are still being applied, and we will discuss the decisions we came to in the next blog post. Furthermore, since certain players also played various positions throughout their careers, for the sake of simplicity, we decided to assign the position that the player started out with in the NFL as their position for their entire career.

Once all of this was figured out, the rest of the database loading was fairly simple - all I had to do was follow the schema, look at the order of the columns in the CSV that we were reading the data in from, and assign the values from the CSVs to the correct columns in each table.

Steven

My contributions between the initial blog post and this midterm report can be split into two sections: methodology hashout, and improved AV data scraping.

1. **Methodology Hashout**

   Motivated by the novel, groundbreaking trade of Brock Osweiler to the Browns (who are considered top dogs in the NFL Analytics space), I was compelled to determine a fair valuation of draft picks - extending off our initial idea of valuing players. For reference, the Texans essentially gave away a second round pick for the Browns to take Osweiler's contract off their hands. In other words, they were trading Osweiler for money! I was in part miffed by the wild variance in draft pick valuation by fans (often completely ignoring the impact of salary).

   In particular, when building an NFL team, we must fundamentally understand the constraints that teams have to work with. The two key restraints are salary cap and roster size. Teams, then, want to maximize the collective talent of the 53 players, given their budget restraint.

   Teams can either pay market value for the player - i.e. through free agency - or they can pay far less than market value. Thus, the value of the draft pick can, in theory, be equal to the added value above what the draft pick is actually paid. See more precise calculations in the actual writeup (see above in "Concrete results/initial insights").

2. **Improved AV Data Scraping**

   There were three problems with our previous AV Data:

   1. The draft position (i.e. Round 1 Pick 5) were shown as dates, as opposed to strings (i.e. 1-5 would translate to January 5, 2017).
   2. We had no unique ID for players. What if two players had the same name? They may have

been drafted
3. We didn't have the players' position. This was key for our planned analysis - checking AV by position, for instance. It was also useful for linking players between datasets - if in both datasets, there was a player of the same name playing for the same team at the same position at the same time, we can be pretty sure they were the same player.

Thusly, we had to get a unique ID for each player - write the draft position to the CSV file as a string (easy) - and get the position for each player.

For the unique ID, we noticed that every player on Pro-Football-Reference has a unique player page. So we took the URL of their player page when scraping.

Getting the position was much more difficult:

1. The player's position is not shown on the table in the Pro-Football-Reference Link. Thus, while scraping, we had to open the href link to every player's profile to get their position. On each player's profile page, there was a listed position - but we also had to consider that some players switch positions over their career, so we couldn't use the listed position. We had to find their year-log (career stats for each year) and draw the position from there.
2. However, this was impractical. Before, we had to open 427 URLs - one for each page we were scraping from. Now, for every URL from before, we had to open a player's URL for all 100 players on the page - thus making our scraping process take approximately 100 times as long (20 minutes to ~2000 minutes or 30+ hours).
3. In practice, this also crashed my computer.
4. So we had to opt for a different strategy. On the Pro-Football-Reference AV rankings page, we could filter by position.
5. So we filtered each position individually, and scraped from every single one of those.
6. This necessarily created duplicates in our dataset - for example, a cornerback might show up as both a cornerback and a defensive back.

Kevin was a major assistance in creating functional BeautifulSoup code for this section.

Visualization

Kevin and Isaiah

For our visualization we decided to look at the distribution of AV scores across quarterbacks. The script was abstracted out such that it is straightforward to replicate for a different position, and we chose to examine a histogram of quarterback AV counts. This visualization is particularly interesting because it allows us to examine the numerical basis of the notion that the quarterback is the most important player for the success of a team. One noteworthy component of this visualization is that it shows us that the distribution for QB AV's actually tends to follow a somewhat trimodal distribution with peaks at 0, 9 and 12. When we talk about quarterbacks colloquially, we tend to classify them into two classes 'elite' and 'not elite', focusing on a player's ability to meet a certain threshold to transition between the two. In reality however, considering a quarterback's ability in the context of 3 or 4 different classes may be of more utility to us and may be something that we can do with relative ease by fitting different normal curves to each peak and assessing the probability that a player's performance is derived from one of those distributions. We could conduct a similar analysis for other positions to assess if how they are stratified in terms of performance. Additionally, we'd like to look at how these distributions change over time to assess the relative importance of each position through the 23 year period that we consider for our data.

View visualization on project blog

We feel that we're making good progress on our project but need to devote some time to formalizing what type of machine learning insights we'd like to make. Specifically, selecting appropriate features and classifiers for supervised learning tasks and deciding whether we would like to pursue and unsupervised learning tasks.