NFL Fantasy Football Sleeper Analysis: Finding Common Attributes in Sleepers in the NFL

Tony Nguyen[1], Andrew Poole[2], Tucker Wheaton[3], and Trey Winters[4]

***Abstract*— Every year millions of individuals compete in Fantasy Football leagues, some do it casually but most are very serious when it comes to winning. Most leagues require dues and even offer prize money to the first place winner which increases the competitive nature of fantasy football. A key to winning a fantasy football league is finding and drafting sleepers. Sleepers are individuals who end up playing drastically better than they were expected to play before the NFL season and are usually drafted in the later rounds. We did some research using the ESPN Fantasy Football API and the Sportsreference API in order to see if their are common attributes in sleepers that could help us predict future sleepers. From the research, we concluded that the depth of the data from the APIs is too limited for us to make a more in-depth conclusion as to what makes a sleeper. We did however find that players who are more versatile and can play across positions will produce more fantasy points.**

I. INTRODUCTION

A. Objective

Our objective for this project is to take data from successful sleepers, players that are playing in the NFL and are projected to bring little to no points at the beginning of the season but by the end are one of the highest ranked players in the NFL league. We want to analyze sleepers and find common data between them and see if that data will help us project future sleepers of the league.

B. Motivation

We got the motivation behind this project from all of us participating in an NFL Fantasy Football league this year. In a fantasy league you and your friends are playing as the general managers of a virtual NFL football team. The league kicks off by having a mock draft where each general manager acquires players to be on their team, in a set order. As the NFL season carries on each general manager sets their starting lineup to accumulate points from touchdowns, field goals, interceptions, pass completions, and a whole lot more. The person with the highest amount of points at the end of the season wins the league and generally the leagues are very competitive and are competing for money. As the season progresses you can play the waiver wire which is where you can auction on players that have not been picked up by any other teams. In the waiver wire you will find players that were projected to make little to no points but by the end of the season they are one of the top ranked players in the NFL league, these are called sleepers. There are multiple sleepers every NFL season and are the key to winning the league.

II. DATASETS USED

The Data that will be obtained will be from a Python API by ESPN, This API allows access to their Fantasy football database. ESPN's API provides us with lots of data to work from such as scoreboard, player information, teams, etc. The one that we are most interested in is boxscore, which holds the values of the weekly points earned by the individual players. We will also look at the other data points to help provide context and insight when analyzing the "sleeper" player's score to determine any trends found on why they ended up doing so well.

Our first dataset will be to find players with low score projections and have performed higher than their projected scores. Once that is determined, looking at the overall score, scoreboard, will be the next dataset used to see if the overall team's performance help influenced the performance of the sleeper player. From this point on, other relative player statistics such as height, weight, and position will be gathered that can positively or negatively affect the player's projection.

All of the datasets gathered will then be graphed and analyzed to spot any trends among the sleeper players that did well. Graphs such as player's physical attributes will be graphed each other, where the X and Y axis will be player name and the individual attributes(height, weight) respectively.

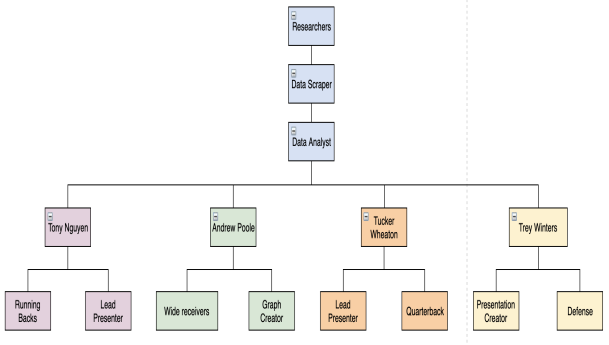
III. Responsibilities

Tucker Wheaton: Data Analyst, Data scraper for quarterbacks, Lead presenter

Andrew Poole: Data analyst, Data scraper for wide receivers, Graph/Table creator

Tony Nguyen: Data Analyst, Data scraper for running backs, Lead presenter

Trey Winters: Data Analyst, Data scraper for defense, Presentation creator



IV. Timeline of Deliverable(s)

September 30th: Group meeting to setup Python, Pymongo, and ESPN Fantasy Football API. This day will be used to familiarize ourselves with the API and accessing data. Each member is tasked to find some NFL statistics from a sleeper pick in the NFL draft and find similarities in their position, weight, height, or any data that reflects direct correlations with previous sleeper picks playing successfully in the NFL.

October 21st: Team members meet again to display some data they found. We will use this day to finish up some in class assignments and implement some code that utilizes each of each other's findings while using the ESPN Fantasy Football API. By then we should have a better understanding of the goal of our implementation.

November 9th: Team members will meet again to discuss better methods of data scraping. By this time we would like to have the majority of our data gathered for presentation. This meeting will be used to finalize that we all have the data ready for analysis.

November 20th: By this time we would like to have our graphs and tables completed using the data we found from the ESPN Fantasy Football API. Our group will meet and compile this data into graphs of a similar format.

December 3rd: Our group plans to be completed with our study by this time. We hope to have compiled graphs and tables showing data that correlates with NFL sleeper picks. We hope to use this data to predict if 5th or 6th round picks will be successful in the NFL.

December 9th: We will have completed the research paper and will have turned it in as finalized. There may be small changes while proofreading.

V. Expected Outcome

By the end of this project we would like to have a set of data that would help us project and find future sleepers of the NFL. We would like to be able to predict what makes sleepers stand out from other NFL players and what are their common similarities. After finding similarities and differences we would like to put our data to the test and project a future sleeper.

As a group, we hope to obtain better knowledge of data analysis and data archaeology using Python. With the help of PyMongo's documention, we plan to learn more about data storage and analysis.

As for what we plan to complete for the semester, we will follow our timeline to the best of our ability. But much of our progress requires us to immediately get started using Python and many other libraries and API's. This can be a very large learning curve but we still believe that we will be able to complete this study before December 3rd. We are motivated to learn more about Fantasy Football and at the same time learn more about data analysis and algorithms using Python.

VI. methodology

A. Gather Data

To kick start the data collection we gathered the top sleepers, who are expertly picked by NFL analyst, from ESPN's Sleeper List of the year for the past three years. Once we had a list of these sleepers we had to manually grab each player's 'player tag' from the the URL of the 'Pro Football Reference' website. A 'player tag', is an unique ID that is assigned to a player. For example, Christian McCaffrey's link to his Pro Football Reference page is "https://www.pro-football-reference.com/players/M/McCaCh01.htm" and his 'player tag' is McCaCh01. The unique 'player tag' is what is used to gather data about each player using the Sportsreference API.

B. Sportsreference API

After we had a compiled list of all the sleeper's and their unique player tags we were able to utilize the

Sportsreference API to gather player attributes like: age, weight, forty-yard dash times, and much more. We gathered this info for every player and grouped the player and their data into groups by position and the year. Once we finished grouping the players we were able to make graphical representations of the player's and their data similar to the one seen in Figure 1. From here we were able to visually compare players and their attributes and see if we could find any common similarities in sleeper attributes.

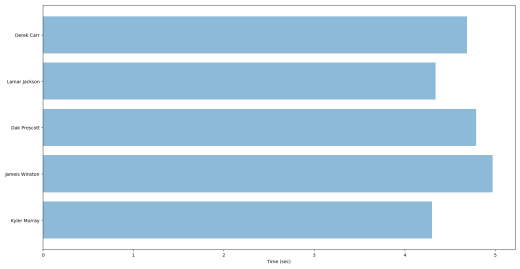


Fig. 1. 40 Yard Dash Time for Quarterback Sleeper Picks

C. ESPN Fantasy Football API

We used ESPN's Fantasy Football API (v3) to gather player's projected and actual points for each week of the NFL season. The API returned a JSON file which consisted of a large bulk of data about each player which we then had to parse and clean using standard Python3 libraries. Once the data was cleaned we were able to gather player's projected and actual points for every week by the position and year. This helped us ultimately see how player's projections shifted over the course of the season and also if their performance increased from start to finish of the NFL season. We graphically represented these projections for every position of every year. The graphs were similar to what is show in figure 2.

D. Comparing Data

After we had both API's (Sportsreference and ESPN Fantasy Football) setup we were able to utilize the data they returned to compare attributes of sleeper's over the course of the past three years. We made graphs which compared player attributes, projected points, and actual points across a particular position. We visually compared all of these graphs in order to see if there were any common patterns or similarities found between the players.

VII. RESULTS

Due to the limited amount of data provided to us by the Sportsreference and ESPN Fantasy Football API, we were not able ot deduce an in depth metric for how to determine whether a player is a sleeper or not, but we were able to find some common trends that we could expound upon in the future. For one, we found that players with higher combine metrics (40-yard dash time, bench press, etc.) were more likely to have breakout potential than those that had lower physical capabilities. We also found that players who were more versatile within their positions had more scoring potential when the season began. While versatility wasn't something we could quantify with concrete metrics, this consisted of things such as running backs who were more capable of catching the ball out of the backfield like Christian McCaffery or quarterbacks who were able to run out of the pocket and consistently create positive plays with their feet like Lamar Jackson. These players who can contribute to their team outside of the standard definitions of their position are an are we would have liked to explore in more detail if we had more time. Given how data driven our analysis was, we would love to have seen how much more in depth we could have made our prediction metrics if we had more access to fantasy footabl data, but even with those limitations we were able to provide valuable information that could be used going forward.

VIII. LIMITATIONS

In the course of developing this project, some limitations came up that weren't accounted for. The biggest issue was the use of ESPN's Fantasy Football API. The API was publicly undocumented and didn't have official documentation on how to use the API to fetch data. This made it difficult to access certain statistics needed such as player points, projected points, rushing yards, etc. Although there was prior documentation from users online that has tinkered with the API and written an informal documentation on how to use the API, it still proposed a challenge on researching for these answers. As of 2019, the API upgraded from it's previous version of V2 and transitioned into V3. This meant that all existing knowledge of how to access the API was invalid and most of the documentation from other users were invalid, as it broke a lot of the ways currently known how to access the API.

After figuring out the bare minimum needed to fetch the required data from the newer V3 API, the dataset available was very limited. The initial plan was to

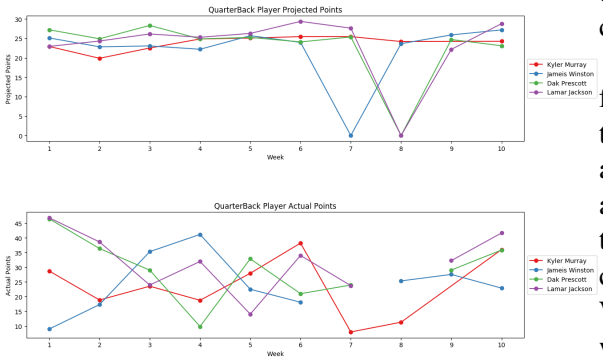


Fig. 2. Quarterback Sleeper Picks Projected vs. Actual Fantasy points for 2018 Season

retrieve statistical data from all the sleeper players from years 2015-2019, giving us a four year range on the players progression and to compare trends between other sleeper players. However, the API only kept stats for the current fantasy season and the previous season. This lowered our dataset by more than 50 percent and greatly reduced our chance to see a recurring trend among the sleeper players.

IX. Future Work

Through this project, we learned a lot about data scraping and how to manipulate APIs. With this increased knowledge, hopefully in the future that we can expand on the data we gather for our analysis. One of the main upgrades we could make in our analysis is actually getting a larger date range of statistics. With the update to the NFL API limiting the available years we weren't able to draw the necessary data to make full conclusions. With more time and resources we should be able to track the trends over decades of NFL Football and have more insights into the outliers. Another upgrade we could make on this project is using more data outside of fantasy point scoring. We started to delve into players measurements and performances at the NFL combine, and believe continuing down that path would help diversify our data. This should lead to a better understanding of what outside factors may contribute to the breakout performances over the years. Outside of fantasy football, a major step for this project moving forward is applying this plan to fantasy sports in general. With basketball season getting into full swing, there are already multiple players that have started out-preforming their fantasy projections. Using the same processes used for this project, there is a great chance of trying to predict breakout basketball players.

To allow our group to make stronger conclusions from the data, our group would have liked to be able to extract data from ESPN's Fantasy Football API and Sports Reference API into a database for further analysis. While most of us were familiar with Python, the steep learning curve of each API we were using cost our team much more time than we had expected. With this database of previous NFL sleeper picks, we would be able to conduct machine learning techniques to further our analysis on specific attributes, statistics, and features of each NFL player. Libraries such as Scikit Learn or Tensorflow, were some methods we had in mind. With these tools, our research would have allowed us to more specifically find the statistics and attributes of an NFL player that optimize for fantasy points in any position.