Predicting NFL Success Based on Combine Performance

Ethan Goldstein

I. Motivation

I was motivated to pursue this project because it allowed me to apply my analytical skills to a topic I am very passionate about – football – and draw valuable insights into how teams can find success. With the NFL draft coming up in just a few short weeks, I felt a worthwhile project would be to look into how NFL combine performances can serve as predictors for success in the NFL, and how teams can potentially leverage players' combine outputs to determine who to draft when the time comes. In particular, I wanted to investigate how some of the main combine events, such as the 40-yard dash, would vary as a predictor between each position (i.e. whether teams should take it into account more for cornerbacks or running backs). In theory, I hypothesized that certain combine events generally offer more predictive value than others, but that each combine event would be able to predict success better for certain positions as opposed to others (i.e. hand size might be more important for quarterbacks and wide receivers than it is for defensive linemen.)

II. Data Sources

The first data source that I used is a csv file containing all the combine data from 1987-2018, titled 'NFL Combine Data.csv'. I was able to access this file for free online through this link: https://data.world/sportsvizsunday/nfl-combine-data/workspace/file?filename=NFL+Combine+Data.xlsx. This file was initially a .xlsx file ('NFL Combine Data.xlsx), but after downloading it I converted it to a .csv file so I could perform the read_csv function on it in pandas. When I opened this file using pd.read_csv, I found that it contains 10029 rows and 19 columns. The data shows each player's name, their position, their physical measurements (i.e. height and weight), and their performance in combine events such as the 40-yard dash, broad jump, and even their wonderlic score (a test of a player's mental aptitude).

The second data source I used is a data table from stathead football on their Player Season Finder page, which contains information about player success in terms of a score for value added, accolades, and games played for all players who entered the NFL between 1987 up until 2018, which includes the players who are in the NFL combine dataset. To access this data from stathead football, I did need to pay for a subscription account, which is free for the first month, and following that it is $8/month. Once I made the stathead account, I went to the website's Player Season Finder page, where I was able to select specific criteria that only looked at players from 1987-2018. Then, I sorted the players by an approximate value coefficient created by the website while also displaying each players' accumulated accolades – this includes number of years played, total games played/started, number of pro bowls made, and number of first team all-pro teams made.

This is the link to this data: https://stathead.com/football/psl_finder.cgi?request=1&seasons_comp==&draft_slot_min=1&undrafted=E&order_by=av&draft_year_max=2021&draft_pick_in_round=pick_overall&season_start=1&order_by_asc=0&pro_bowls_comp==&weight_max=500&bmi_min=0&seasons_val=-1&conference=any&all_pros_first_team_val=-1&pro_bowls_val=-1&year_min=1987&all_pros_first_team_comp==&height_max=99&draft_slot_max=500&match=combined&year_max=2018&height_min=0&season_end=-1&draft_year_min=1936&draft_type=B&age_min=0&age_max=99&bmi_max=100&offset=0. To look at this data in jupyter lab, I performed a pd.read_html function through pandas with the link to the stathead html page with all of the data, and then while it initially shows a list result when you do the read_html, indexing through the list to get the 0th item will provide the desired table of information with the information mentioned above.

III. Data Manipulation

In order to manipulate the data, the first step that I took was making sure that I imported all of the python packages necessary to complete my desired analyses and visualizations – these packages were simply pandas, numpy, and matplotlib. After importing these packages, I loaded in my two data sources using the read_csv and read_html functions as specified in the data sources section and named the respective DataFrames combine and players – when using the stathead data, the read_html function would only import and certain amount of players each time, so I had to apply the read_html function to the stathead data 10 times to get a desired amount of players from the stathead page to use as data points.

The next step was to figure out how to merge the two datasets into one main data set that put together each player's combine statistics and career accomplishments. This process initially offered me a few different challenges that I had to work through – when I loaded in the stathead data to my jupyter lab file, it was loaded in as a MultiIndex data frame where many of the columns would have a title of 'Unnamed: 0_level_0' with the desired column name (i.e. 'Name') below it. To fix this issue, I ended up learning that I had to use the function players.columns.droplevel() so that the top level of column names would be dropped and I would be left with my desired column names.

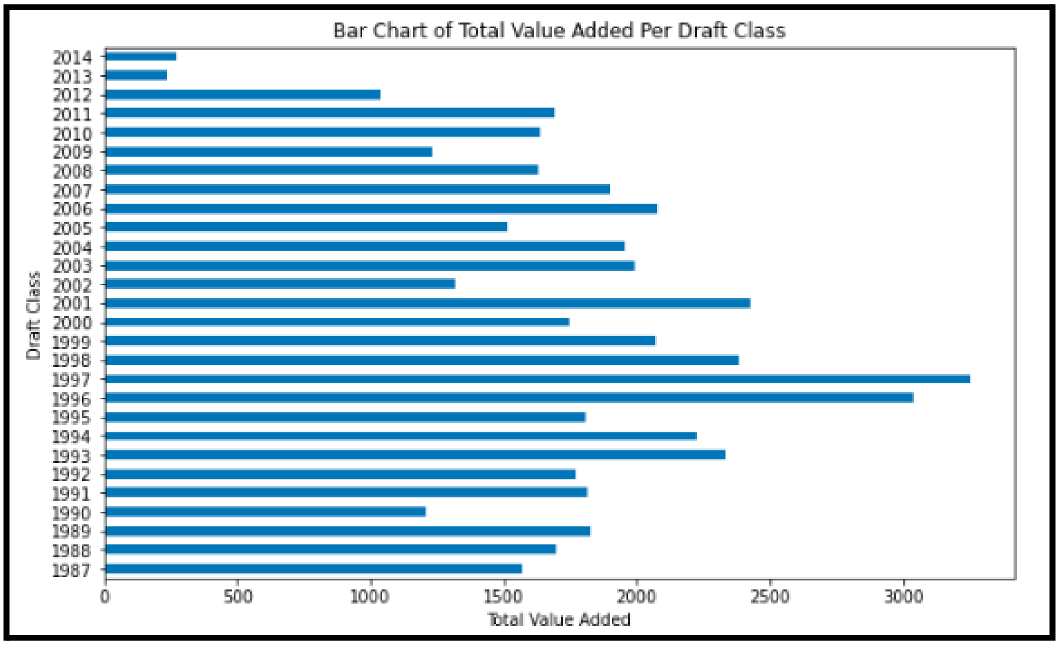
After figuring out this issue, I was able to then rename the 'Player' column from the stathead DataFrame, changing it to 'Name' so that I could then join both of my datasets on 'Name' – this would essentially mean that each player would have their career accolades joined with their combine statistics in one main DataFrame. However, the problem that I ran into here was dealing with duplicate names. Over the timespan I was looking at, there were multiple players in the data who shared the same name (i.e. Anthony Miller, Michael Bennett), which caused issues when trying to join the two DataFrames based on name. As a result, I applied the .dropduplicates('Name') function before joining the DataFrames together. However, I also found that it wasn't necessarily just a problem of having duplicates in each dataframe, but there were some players that had the same name, yet there were not duplicates in either of the dataframes. For instance, Marcus Allen showed up once in both the combine dataframe and the stathead dataframe, but the Marcus Allen from the combine dataframe entered the NFL in 2018, whereas the Marcus Allen from the stathead dataframe entered the league in 1987 – as a result, joining the dataset would incorrectly assign the 2018 Marcus Allen all of 1987 Marcus Allen's accolades, because it was joining them based on having the same name. To fix this issue, I created a variable called that accumulated all of the rows where the year the player participated in the combine (column title 'Year') did not match the year their NFL career started (column title 'From').

The last problem that I found out I faced when initially manipulating the DataFrame is that when I tried to isolate certain columns from my combine csv file (i.e. 'Arm Length (in)', 'Hand Size (in)') from the DataFrame for my analyses, I would receive an error saying that column name was not in the index of columns, despite not having any typos. I first tried to eliminate any spaces and parentheses using str.replace() method, but that did not work, so I eventually just went into the csv file and manually adjusted the column names that were not working, and this ended up fixing the issue.

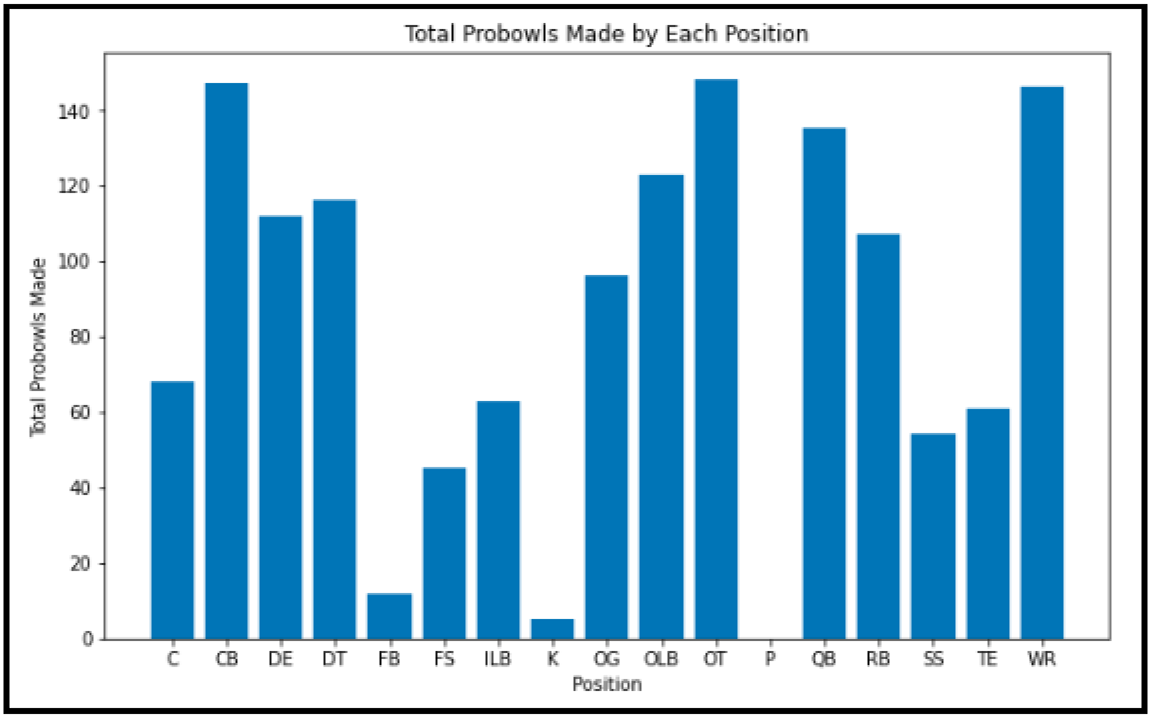
One additional thing I had to change within my dataset was the type of data that was contained within each column – when I initially applied .dtypes to my dataframe, I found that a lot of the columns, particularly the ones from the stathead data, were of the object data type, rather than str, float, or int. This had been giving me problems initially when I was trying to perform operations on certain columns, as I was unable to do things such as sort_values(). As a result, I had to go back to the initial dataset and convert each column to the desired datatype using .astype(). This allowed me to manipulate the DataFrame in a variety of ways so I could perform my desired visualizations – in particular, I focused heavily on using sort_values() to sort a variety of different columns. Two main columns I would sort were the pro bowls ('PB') column approximate value ('AV') column, as these were my two main metrics for determining player success, and sorting rows in order of these columns could tell me who had the most/least success based on these metrics. Additionally, I was able to manipulate the data numerous times using the .groupby() method – this was particularly helpful when I used it to group the data based on position. Since many of the visualizations I wanted to do were comparisons of how certain combine statistics correlated differently with certain positions, I was able to use .groupby() and .mean() to obtain average statistics for the desired positions I wanted to compare.

IV. Analysis and Visualization

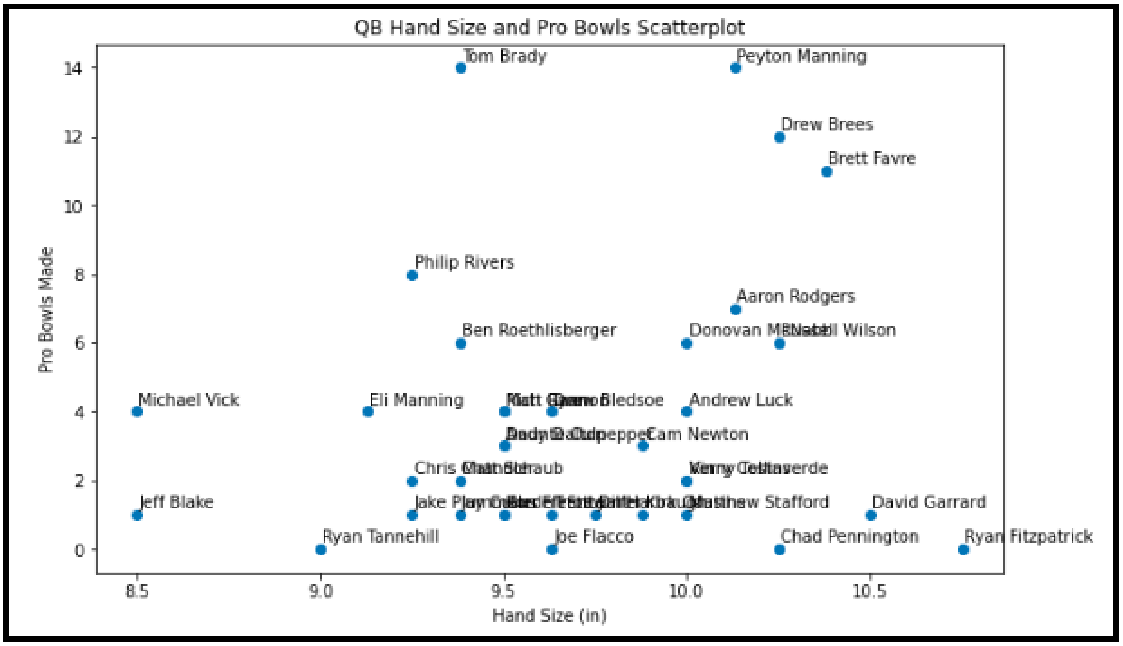
The analyses and visualizations that I executed for this project were two-fold: I wanted to perform a few initial analyses to understand how the success of players may have differed over time, as well as how different positions may have had more success in the NFL over the time period during which I am conducting my analysis.



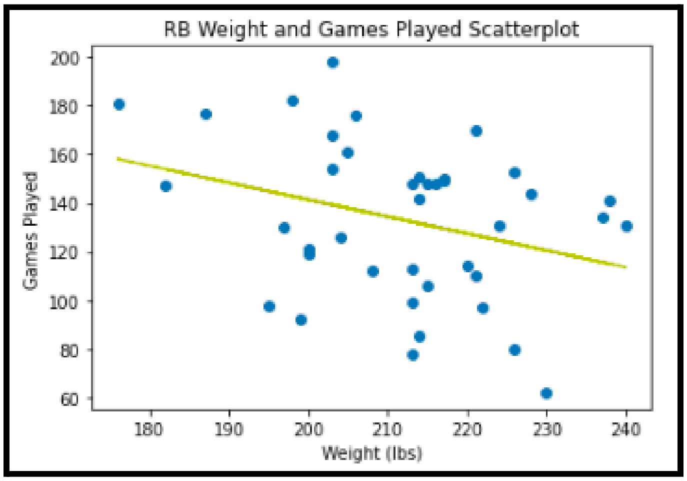
This first analysis was to identify within the available dataset, which draft class had the most success in the NFL. This bar chart isolated each draft class and calculated the sum of their entire class' value added as computed by Stathead's football statistics database. Evidently, most of the classes hovered around a total value of 1500, with 1996 and 1997 standing out as the only two classes with a value of over 3000.



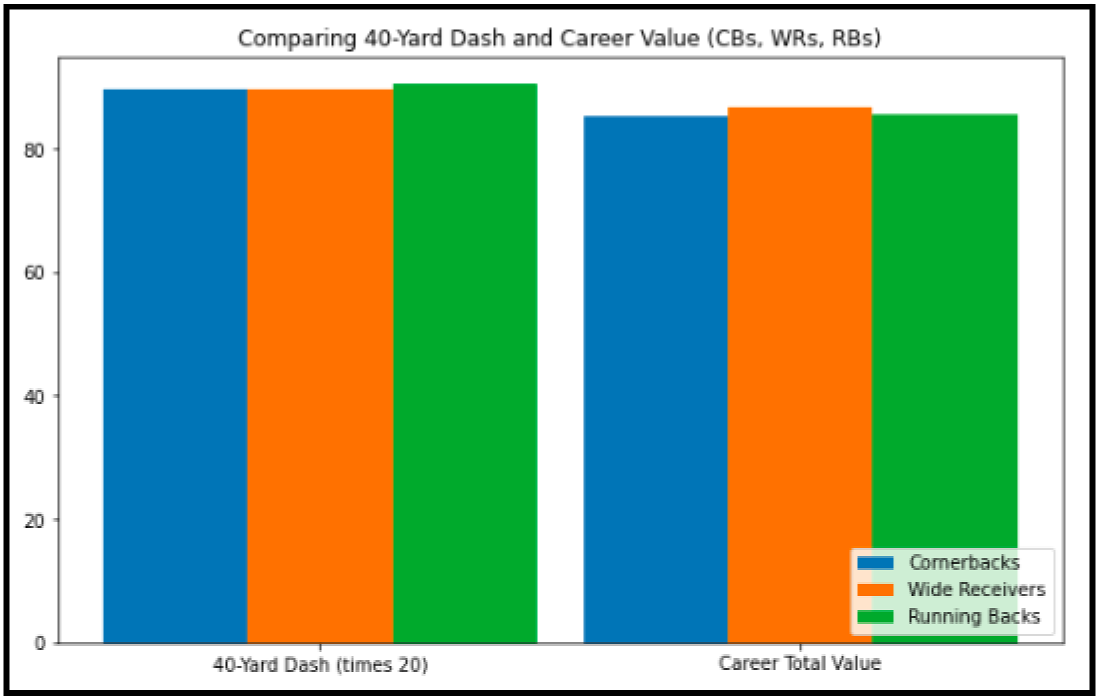
I also wanted to perform an initial analysis that looked at the success of the players in the dataset grouped by position – rather than use stathead's value added computation, I chose to look at pro bowls made to investigate the success of each position. According to the resulting bar chart, it seems that cornerbacks, offensive tackles, and wide receivers have made the most pro bowls with over 140 total at each position. The one caveat with these results is that there are more cornerbacks, offensive tackles, and wide receivers in the NFL than other positions purely because each of those positions requires at least two starters whereas many other positions only require one starter. However, there were many other positions with two starters that did not produce as many pro bowlers, so it helps to understand that within the available data, these three positions had the most successful players in terms of made pro bowls.



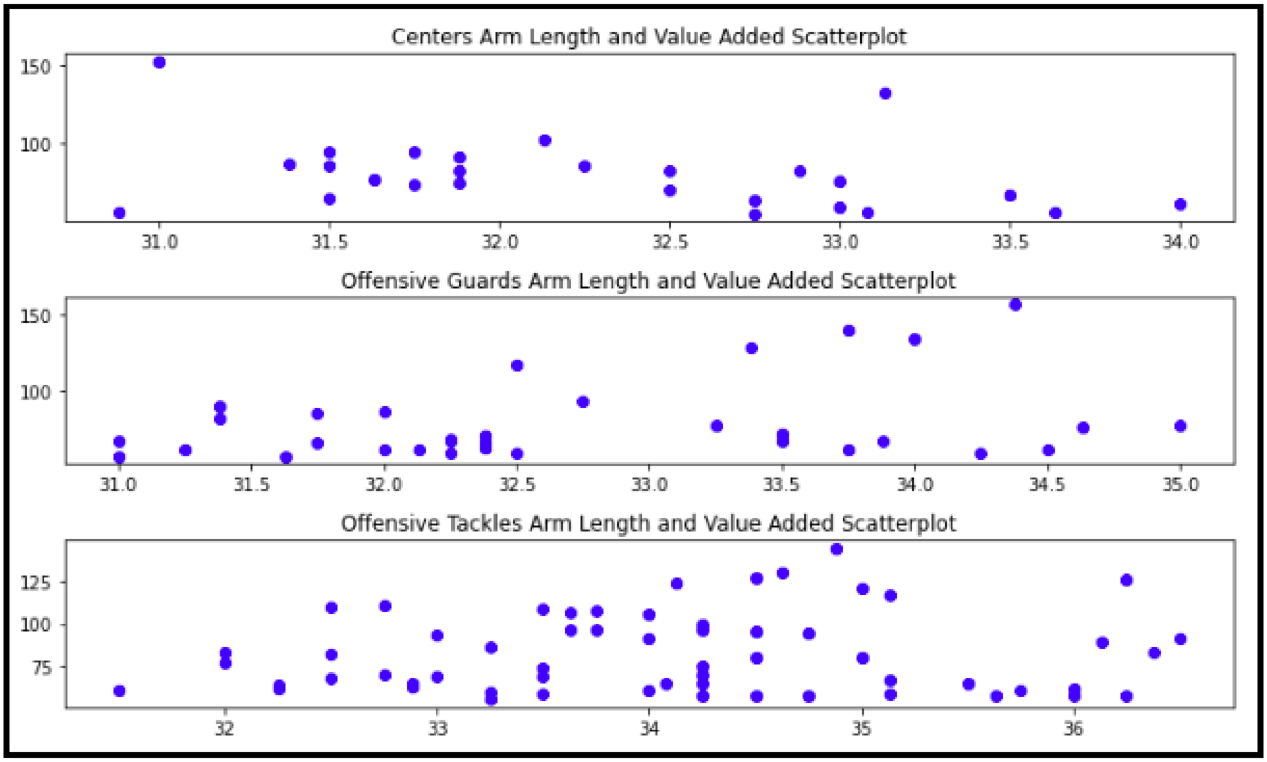
After the initial 2 visualizations, I wanted to focus more on how specific combine statistics might be predictors of success for certain positions in the NFL. The first visualization I created for this was a scatter plot to see how hand size affects QB success in the NFL in terms of pro bowls made – given that hand size has been a hotly debated topic at the combine in recent years, I wanted to test if this was a valid concern when drafting quarterbacks. This scatterplot shows a relatively random scatter, but we can see that most of the successful quarterbacks have a hand size of at least 9 (almost 9.5), and 3 of the 4 quarterbacks with at least 10 pro bowls have a hand size of above 10. This indicates that hand size may not be a large concern for drafting quarterbacks, but many top quarterbacks have had larger hand sizes (at least 9 inches).



With durability being one of the primary concerns for running backs in the NFL, the next analysis I wanted to run was seeing if there is a correlation between a running back's weight and increased durability. To do this, I produced a scatter plot with a trendline comparing a running back's total games played in their career with their weight. Interestingly enough, the plot shows a negative correlation between the two, meaning that games played decreases as weight increases. This trend might be because there are fewer running backs playing at lower weight (less than 190 lbs), yet they all have played at least 140 games – given that there are more running backs playing at higher weight, it would be more reasonable to expect that more of them may also have played less games. Still, it was interesting to see that increased weight did not show any clear benefits for a running back's durability in the NFL.



The 40-yard dash is one of the most focused-on events at the combine, as it shows us many of the fastest players entering the draft. I wanted to look at how a good 40-yard dash time can predict success for cornerbacks, wide receivers, and running backs – 3 positions where speed is considered to be crucial in the NFL. To do this, I created a bar chart comparing the average 40-yard dash times (multiplied by a factor of 20 to magnify any noticeable differences between the positions) and the average career value for each position. As evidenced by this graph, each position was very similar in both speed and value added. Running backs seem to be a little bit slower, but still had similar value throughout their careers to wide receivers and cornerbacks. Wide receivers seem to have the highest average value, so speed may indicate success slightly more for them, but overall it seems there is a very similar correlation between speed and NFL success for all 3 positions.



The last visualization I created was to investigate the success of offensive linemen in the NFL. Typically, it is expected that longer arms will be more beneficial for linemen when trying to block defenders, and I wanted to see if that was more important for centers, offensive guards, or offensive tackles. To do this, I created a visualization with 3 subplots, where each subplot is a scatterplot comparing the player's arm length and overall value added. While there is somewhat random scatter at each position, we can see that for both offensive guards and offensive tackles in particular, the most valuable players meet a specific benchmark in arm length – for guards, the 4 most valuable players have an arm length above 33 inches, and for tackles, the 7 most valuable players have an arm length above 34 inches. This indicates that arm length is not as important for centers, but as you move outward on the offensive line towards guards and tackles, arm length becomes more important. This is not surprising, as there is more space for pass rushers to maneuver around the offensive linemen on the edges, so tackles in particular benefit from having longer arms so they can cover more area when reaching out to block defenders.

Overall, my analyses and visualizations did confirm that some combine events offer more predictive value for specific positions than others, but it was interesting to see that many times there was not a strong correlation between combine performance and success in the NFL. I think that these findings serve as a reminder that numerous top athletes attend the combine and enter the NFL each year, so it is hard to identify the top players purely based on their combine performance and measurements. The combine is just one step in a long process of scouting talent that will be entering the NFL – it is definitely a helpful tool for identifying players that have traits that are desired in the NFL, but is one of many variables that help determine which players will go on to have successful NFL careers.