

# NFL QB Performance Data Analytics

By: Jack Delaney, Zane Habib, and Thomas Kim

## Data Source

The source we used for our data originated from ESPN. ESPN is a sports network that contains statistics on every major sport. Our favorite sport is football so the team chose to look at statistics within that



game. Beyond that we looked at the statistics from Quarterbacks. The statistic we used that we felt was the best to measure was the touchdown category. This stat shows the amount of time the Quarterback threw a pass that led to scoring. ESPN keeps great track of sports data like such, and we felt parsing information from that website gave us the best access to the information we need. Touchdowns are a great statistic in terms of performance because it measures the amount a Quarterback scores for their team. We chose to create visualizations of touchdowns over passing yards since passing yards can be a misleading statistic. Some teams depend on their quick screens or dump passes to their running backs which can pad their passing yards statistics. Touchdowns thrown by a quarterback represent their ability to produce in the red zone or pick apart a defense while positioned midfield.

## Web Crawler

In order to locate the specific data, we wished to collect and extract, we created a web crawler. Web crawlers, also known as web spider, is a program which browses the World Wide

```
for y in range (2015, 2017, 1):
    tmp = random.random()*5.0
    print ('Sleep for ', tmp, ' seconds')
    time.sleep(tmp)

    if y == 2016:
        url = 'http://www.espn.com/nfl/statistics/player/_/stat/passing/sort/passingYards'
        #download
        filehandle = urllib.request.urlopen(url)
        html = filehandle.read()

        data1 = []
        data1.append(parse_html(html))
        data1 = data1[0]

        count1 = 0

        for player in data1:
            if player[0] == 'BX':
                del data1[count1]
                count1 += 1
            else:
                player = player[6]
                count1 += 1

    else:
        url = 'http://www.espn.com/nfl/statistics/player/_/stat/passing/sort/passingYards'
        #download
        filehandle = urllib.request.urlopen(url)
        html = filehandle.read()

        data2 = []
        data2.append(parse_html(html))
        data2 = data2[0]

        count2 = 0

        for player in data2:
            if player[0] == 'BX':
                del data2[count2]
                count2 += 1
            else:
                count2 += 1

    print ('Download from ', url)
```

Web for various different reasons. Our reason being to locate the desired performance data. After we determined to source, ESPN.com mentioned previously, we constructed a program that would create a URL, open the URL on the World Wide Web, then collect the HTML content from the web page containing the NFL Quarterback stats for a specific year. Since we wanted to extract and collect data over several years, we designed a loop which would run this program and extract NFL Quarterback data over the 2015 and 2016 season. Since this program only captured and stored the HTML content of the webpage, there was still further efforts to locate the specific data.

## HTML Parser

Once we had the HTML from ESPN.com, for the specific years to analyze, we needed to generate a parser in iPython. A parser is a program that receives an input from another source and breaks down the input into parts in order to extract distinct attributes. The parser that we developed sequentially looks at each line in the HTML from ESPN.com to find the “table” tag. From here we can extract each row as a list, while each cell in the row is extracted as an individual object in the list. For example, a list that could be extracted from ESPN.com using our parser would be [“Brees, Drew”, “QB”, “27”]. When each row is extracted it is then saved into a list containing all of the NFL Quarterbacks on the ESPN web page.

```
#Used to parse though html to find data on ESPN.com
def parse_html(html):
    soup = BeautifulSoup(html, 'html.parser')

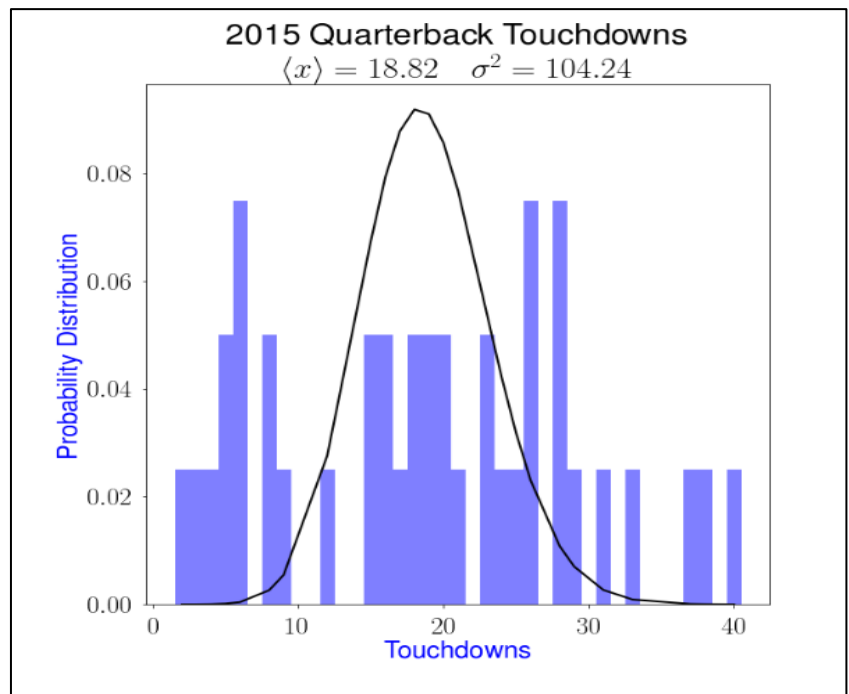
    tbl = soup.find('table')
    dataset = list()

    for i,row in enumerate(tbl.find_all('tr')):
        rowData = list()
        for j,column in enumerate(row.find_all('td')):
            rowData.append((column.text))
        dataset.append(rowData)
    return dataset
```

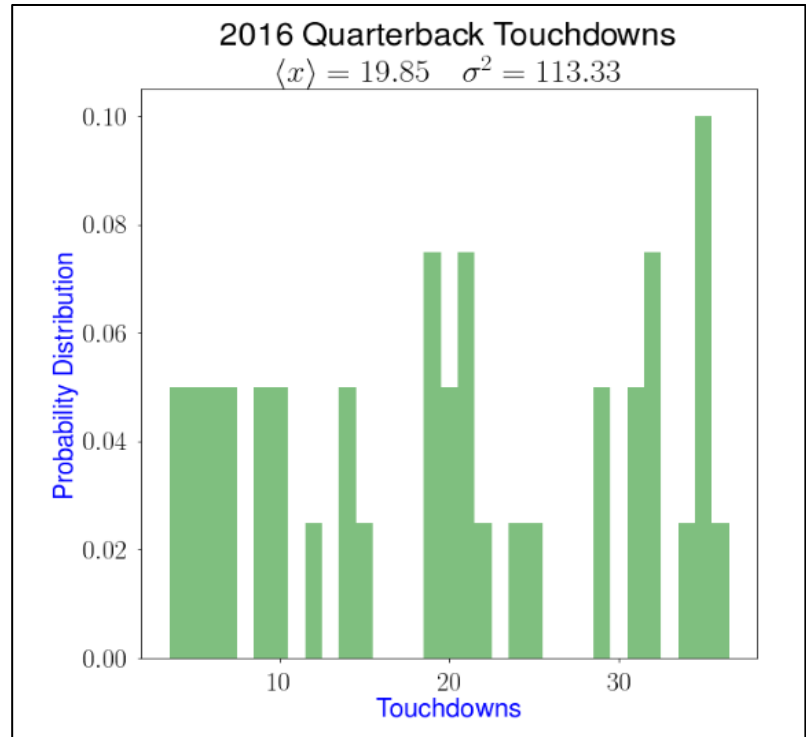
After we parsed the data for NFL Quarterbacks, we created a dictionary for both years, 2015 and 2016, where each number of touchdowns was the key and the value of each key was the number of NFL Quarterbacks to throw that many touchdowns. This dictionary allowed us to create accurate graphs to compare the performances over the past two seasons.

## Data Analysis and Visualizations

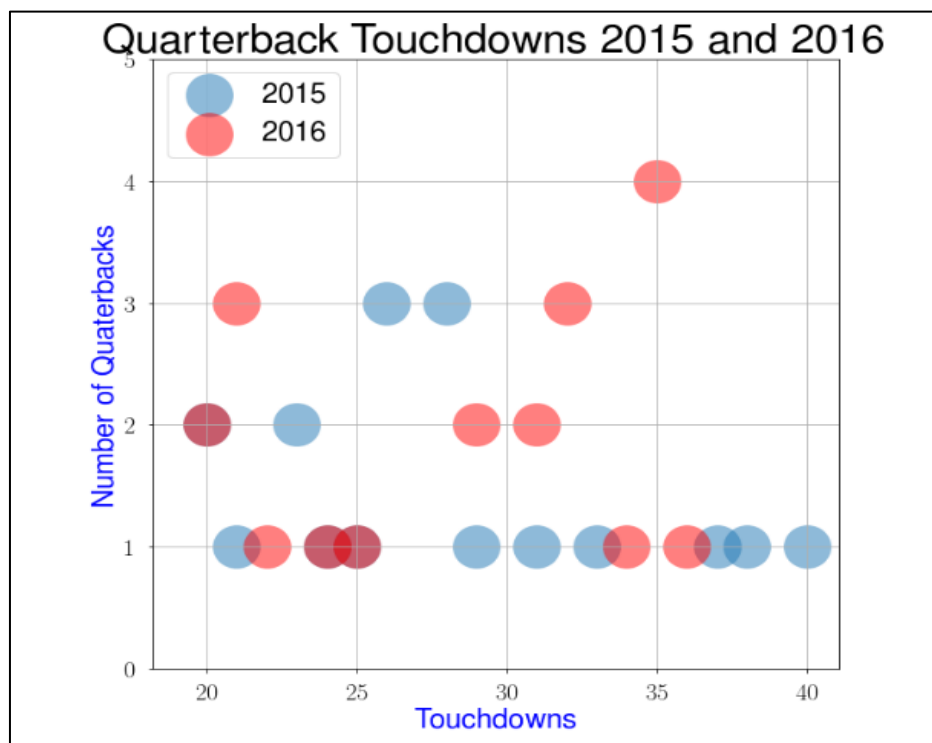
Once we located and extracted our data for NFL Quarterbacks for the 2015 and 2016 season we decided to create a probability distribution histogram to see in there is any patterns or trends that stood out. Looking at the graph below one could see that there seems to be no pattern. The mean being 18.82 touchdowns seemed to be the most relevant piece of information with to continue.



Following the same process as previously done with the 2015 NFL Quarterback data, creating a probability distribution histogram, we expected to find the same results. We, however, discovered that there were many more NFL Quarterbacks to have thrown a higher number of touchdowns than in the previous season. The mean had also been raised to 19.85.



Since the two different data sets seemed to display different performances in these two years we created a scatter plot for the higher performing NFL Quarterbacks in the league during these two years. We defined higher performing NFL Quarterbacks as throwing for more than 20 touchdowns in a season, since backup NFL Quarterbacks throw for a few touchdowns when a starter gets hurt, creating outliers. The scatter plot showed a positive trend upward for these higher performing NFL Quarterbacks in the 2016 season as shown below.



## **Interpretation and Conclusion**

The data displayed that NFL Quarterbacks had many more touchdowns in the 2016 season compared to the 2015 season. More NFL Quarterbacks threw for 20 or more touchdowns in 2016, 2015 had 19 where 2016 had 21. There was a higher average number of touchdowns per NFL Quarterbacks, rising from 18.82 in 2015 to 19.85 in 2016. Lastly, many more NFL Quarterbacks throwing for more than 30 touchdowns in 2016 with 11, while 2015 had 5. This is most likely due to the increasing pass play percentage of the entire league is increasing each year, from high 50% to low 60%. In conclusion, our team can infer that 2016 displayed better Quarterback play when measuring the touchdown statistic.