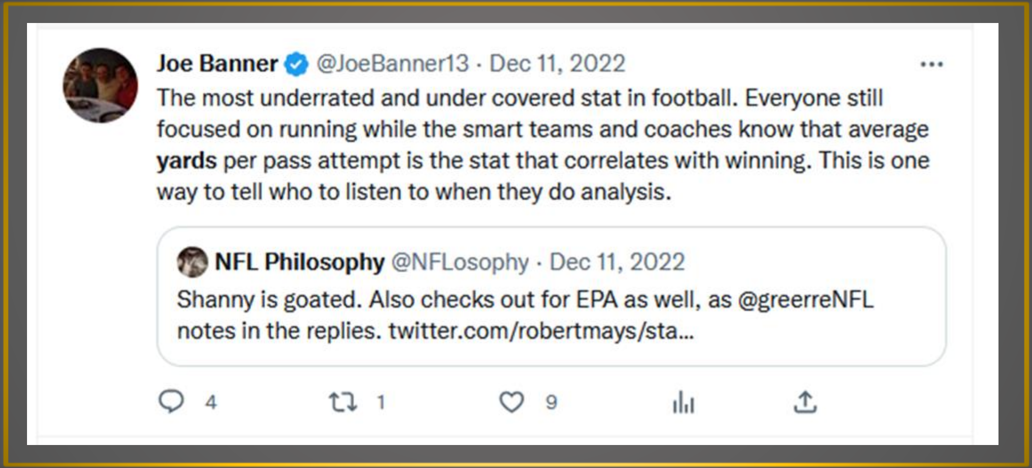
Does Yards Per Attempt translate to wins?

Cabell Ferguson
1/10/23
CIS469L
ECPI University

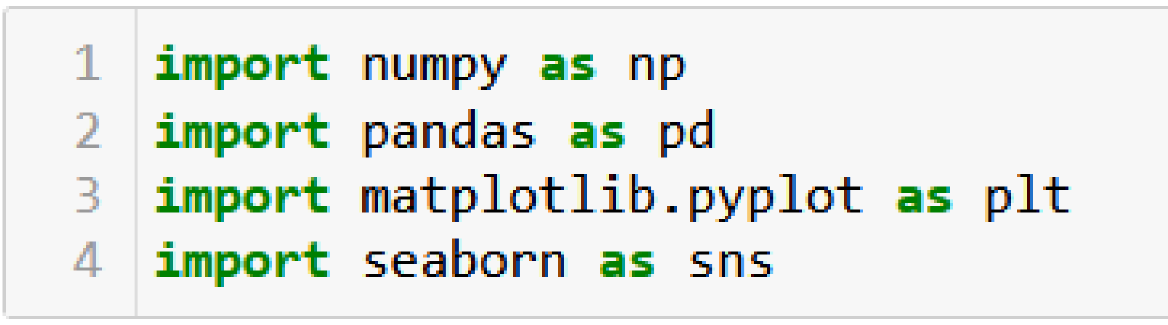


Joe Banner @JoeBanner13 · Dec 11, 2022

The most underrated and under covered stat in football. Everyone still focused on running while the smart teams and coaches know that average **yards** per pass attempt is the stat that correlates with winning. This is one way to tell who to listen to when they do analysis.

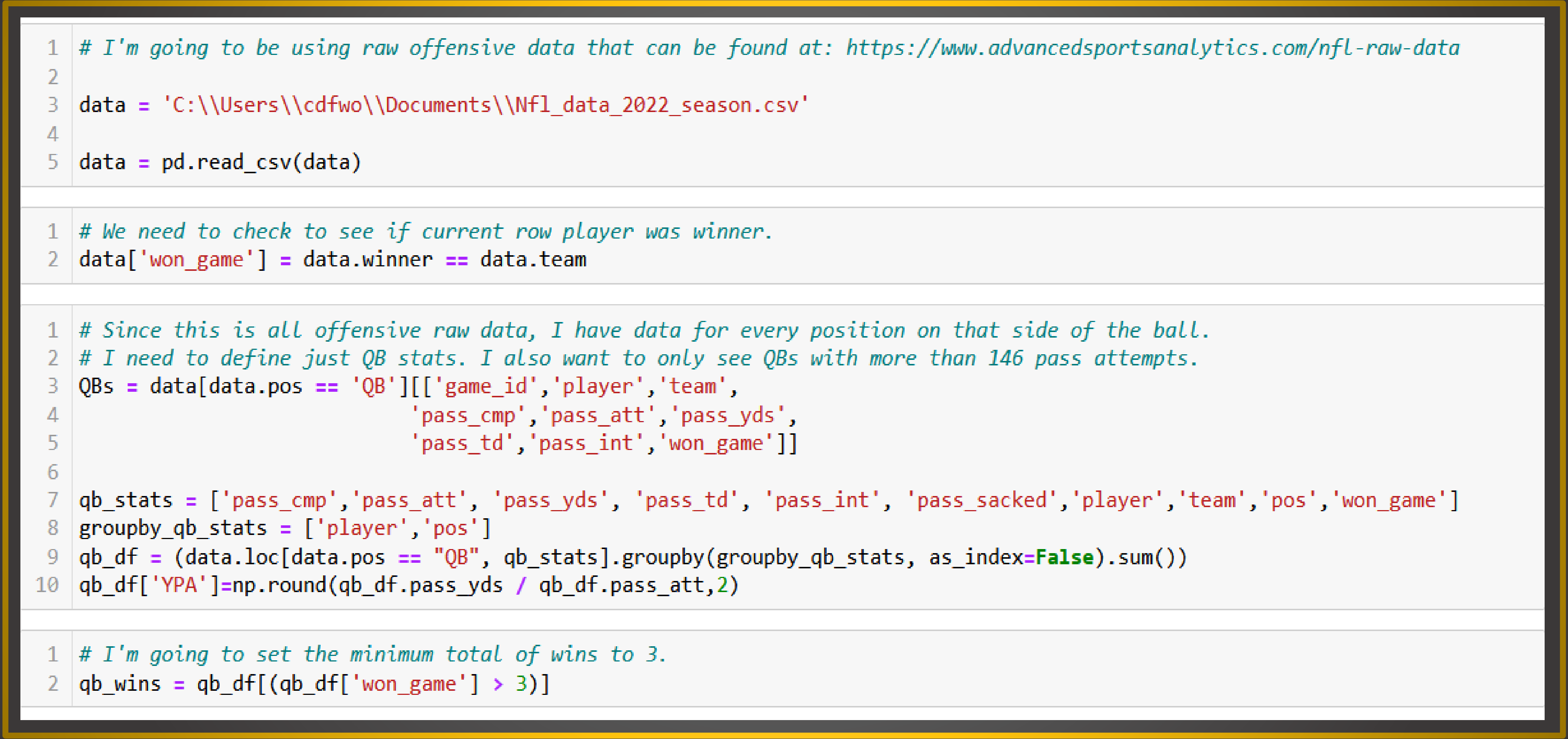
> **NFL Philosophy** @NFLosophy · Dec 11, 2022
>
> Shanny is goated. Also checks out for EPA as well, as @greerreNFL notes in the replies. twitter.com/robertmays/sta...

How I’ll be analyzing the data



```python
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

Preparing dataset for analysis



```
# I'm going to be using raw offensive data that can be found at: https://www.advancedsportsanalytics.com/nfl-raw-data

data = 'C:\\Users\\cdfwo\\Documents\\Nfl_data_2022_season.csv'

data = pd.read_csv(data)
```

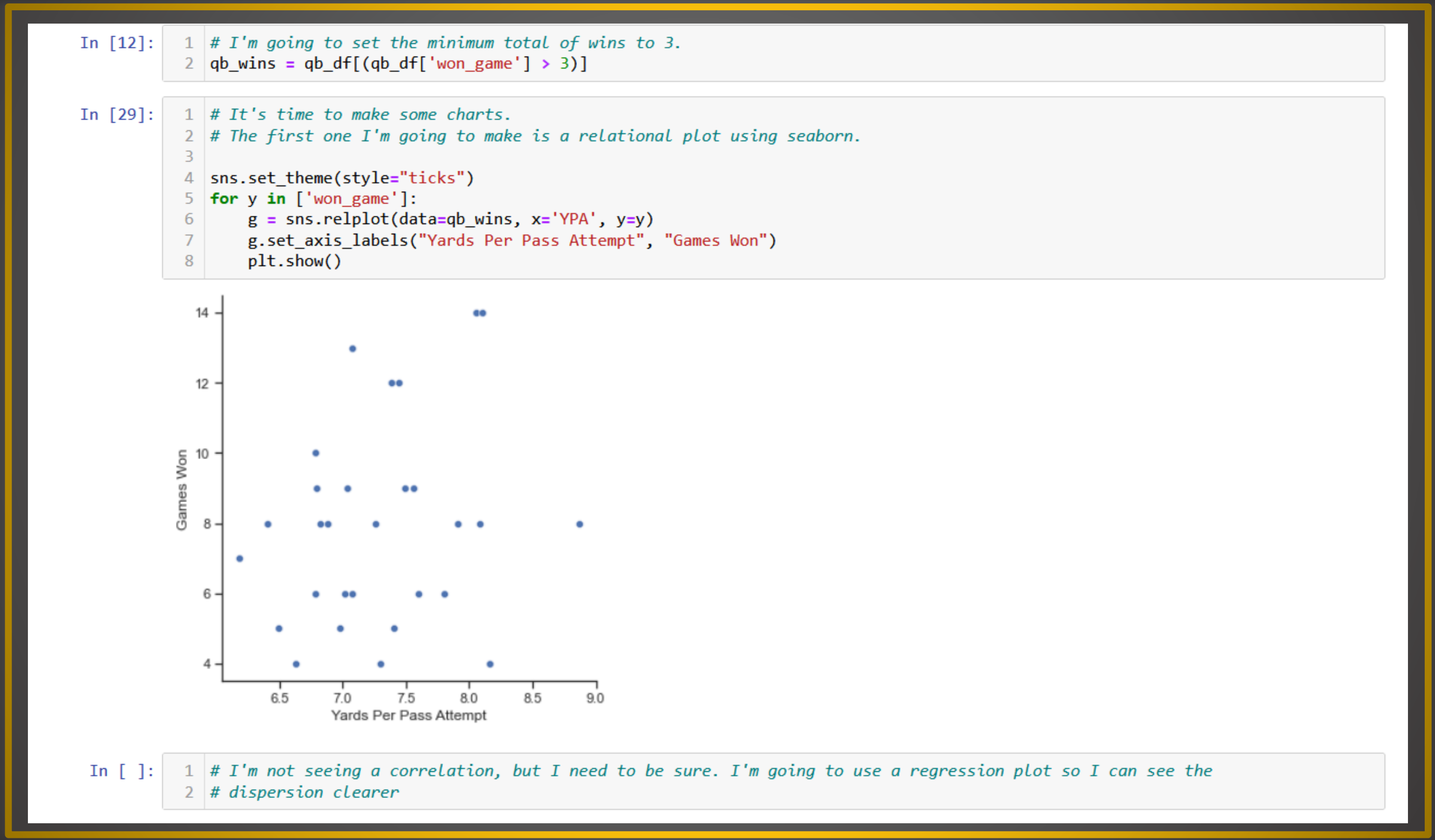
```
# We need to check to see if current row player was winner.
data['won_game'] = data.winner == data.team
```

```
# Since this is all offensive raw data, I have data for every position on that side of the ball.
# I need to define just QB stats. I also want to only see QBs with more than 146 pass attempts.
QBs = data[data.pos == 'QB'][['game_id','player','team',
                            'pass_cmp','pass_att','pass_yds',
                            'pass_td','pass_int','won_game']]

qb_stats = ['pass_cmp','pass_att', 'pass_yds', 'pass_td', 'pass_int', 'pass_sacked','player','team','pos','won_game']
groupby_qb_stats = ['player','pos']
qb_df = (data.loc[data.pos == "QB", qb_stats].groupby(groupby_qb_stats, as_index=False).sum())
qb_df['YPA']=np.round(qb_df.pass_yds / qb_df.pass_att,2)
```

```
# I'm going to set the minimum total of wins to 3.
qb_wins = qb_df[(qb_df['won_game'] > 3)]
```

Charts!



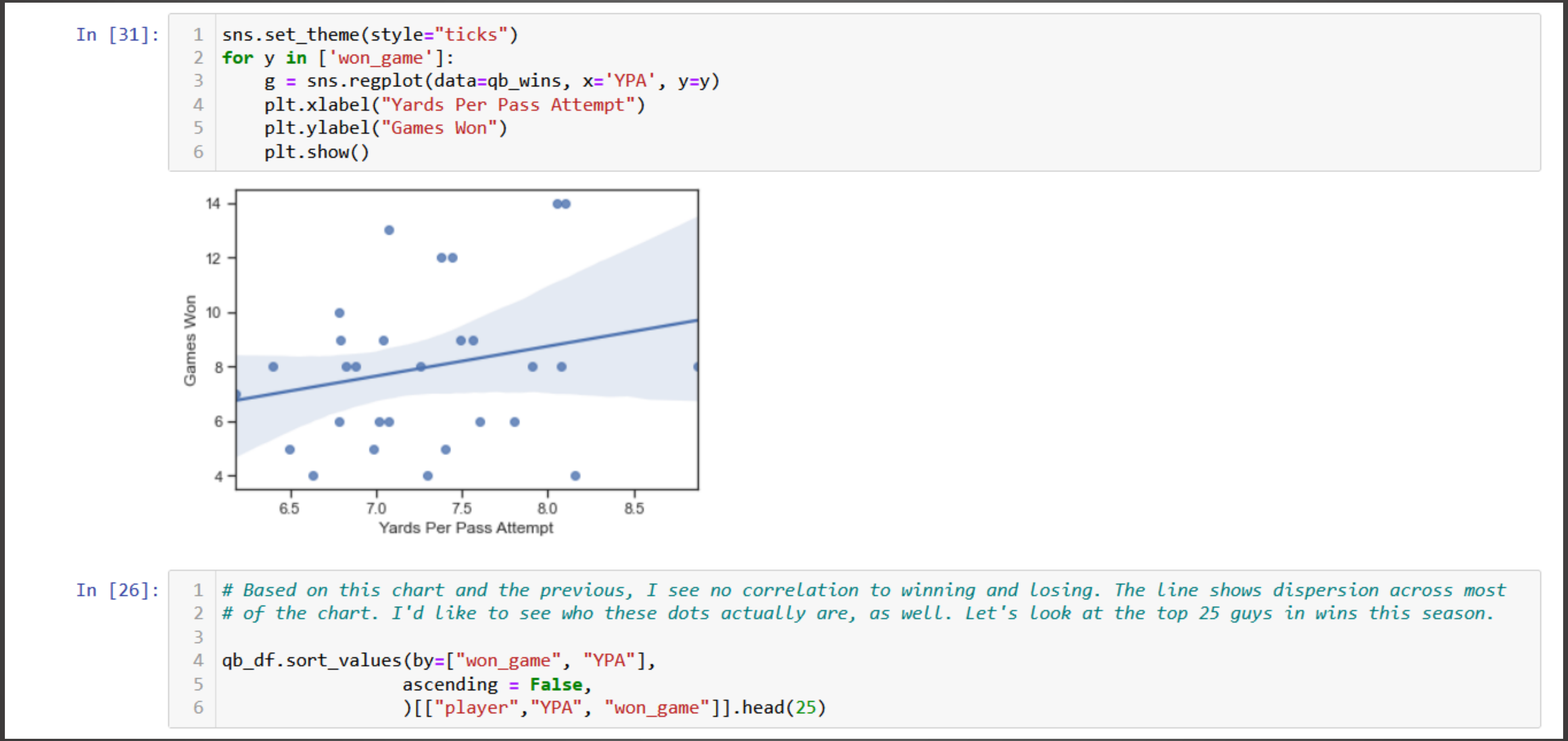
In [12]:
```
# I'm going to set the minimum total of wins to 3.
qb_wins = qb_df[(qb_df['won_game'] > 3)]
```

In [29]:
```
# It's time to make some charts.
# The first one I'm going to make is a relational plot using seaborn.

sns.set_theme(style="ticks")
for y in ['won_game']:
    g = sns.relplot(data=qb_wins, x='YPA', y=y)
    g.set_axis_labels("Yards Per Pass Attempt", "Games Won")
    plt.show()
```

In [ ]:
```
# I'm not seeing a correlation, but I need to be sure. I'm going to use a regression plot so I can see the
# dispersion clearer
```

Charts!



In [31]:

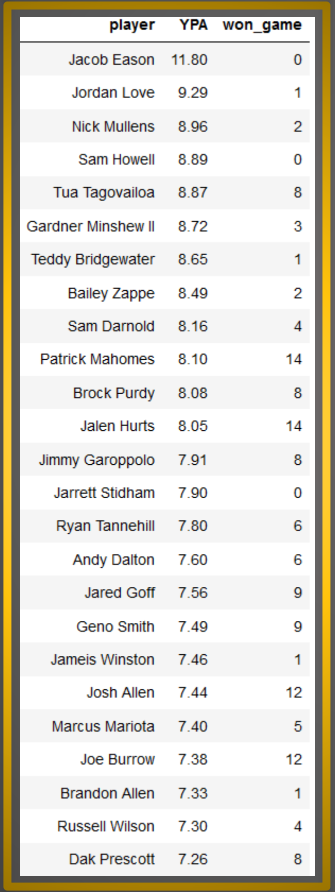
```
sns.set_theme(style="ticks")
for y in ['won_game']:
    g = sns.regplot(data=qb_wins, x='YPA', y=y)
    plt.xlabel("Yards Per Pass Attempt")
    plt.ylabel("Games Won")
    plt.show()
```

In [26]:

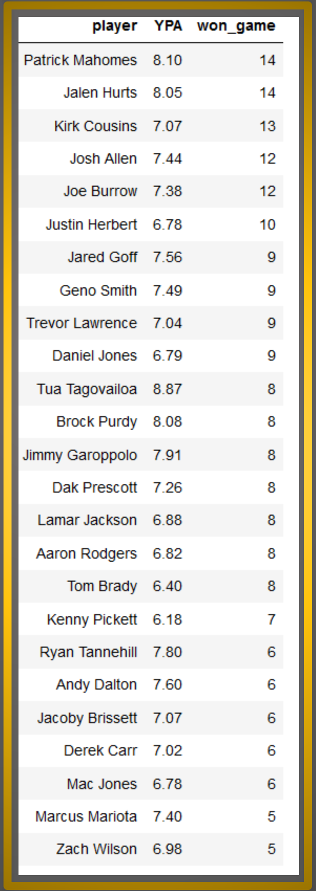
```
# Based on this chart and the previous, I see no correlation to winning and losing. The line shows dispersion across most
# of the chart. I'd like to see who these dots actually are, as well. Let's look at the top 25 guys in wins this season.

qb_df.sort_values(by=["won_game", "YPA"],
                 ascending = False,
                 )[["player","YPA", "won_game"]].head(25)
```

Comparing the Top 25 in YPA and Wins

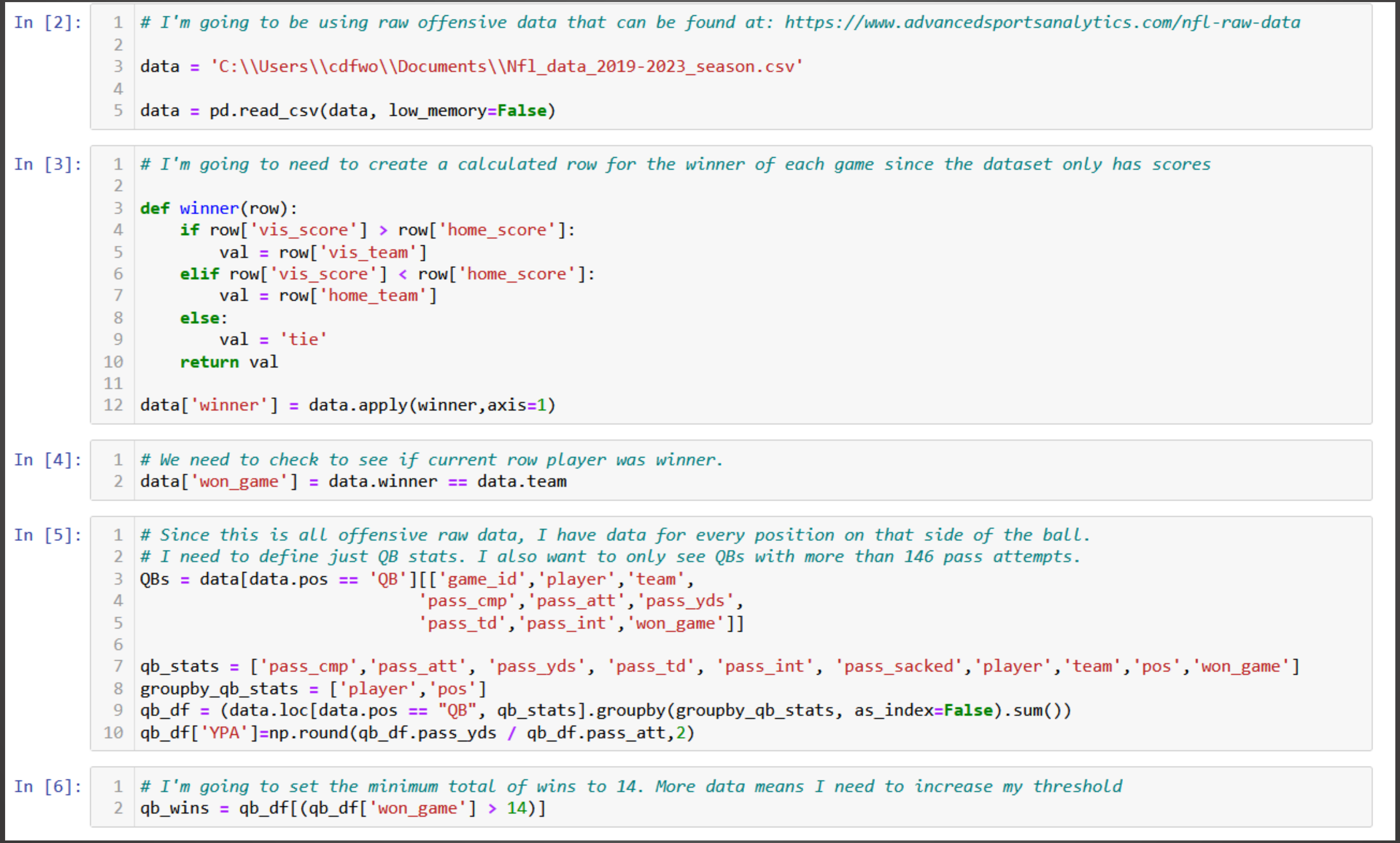


player	YPA	won_game
Jacob Eason	11.80	0
Jordan Love	9.29	1
Nick Mullens	8.96	2
Sam Howell	8.89	0
Tua Tagovailoa	8.87	8
Gardner Minshew II	8.72	3
Teddy Bridgewater	8.65	1
Bailey Zappe	8.49	2
Sam Darnold	8.16	4
Patrick Mahomes	8.10	14
Brock Purdy	8.08	8
Jalen Hurts	8.05	14
Jimmy Garoppolo	7.91	8
Jarrett Stidham	7.90	0
Ryan Tannehill	7.80	6
Andy Dalton	7.60	6
Jared Goff	7.56	9
Geno Smith	7.49	9
Jameis Winston	7.46	1
Josh Allen	7.44	12
Marcus Mariota	7.40	5
Joe Burrow	7.38	12
Brandon Allen	7.33	1
Russell Wilson	7.30	4
Dak Prescott	7.26	8



player	YPA	won_game
Patrick Mahomes	8.10	14
Jalen Hurts	8.05	14
Kirk Cousins	7.07	13
Josh Allen	7.44	12
Joe Burrow	7.38	12
Justin Herbert	6.78	10
Jared Goff	7.56	9
Geno Smith	7.49	9
Trevor Lawrence	7.04	9
Daniel Jones	6.79	9
Tua Tagovailoa	8.87	8
Brock Purdy	8.08	8
Jimmy Garoppolo	7.91	8
Dak Prescott	7.26	8
Lamar Jackson	6.88	8
Aaron Rodgers	6.82	8
Tom Brady	6.40	8
Kenny Pickett	6.18	7
Ryan Tannehill	7.80	6
Andy Dalton	7.60	6
Jacoby Brissett	7.07	6
Derek Carr	7.02	6
Mac Jones	6.78	6
Marcus Mariota	7.40	5
Zach Wilson	6.98	5

Let’s add in more seasons



In [2]:

```python
# I'm going to be using raw offensive data that can be found at: https://www.advancedsportsanalytics.com/nfl-raw-data

data = 'C:\\Users\\cdfwo\\Documents\\Nfl_data_2019-2023_season.csv'

data = pd.read_csv(data, low_memory=False)
```

In [3]:

```python
# I'm going to need to create a calculated row for the winner of each game since the dataset only has scores

def winner(row):
    if row['vis_score'] > row['home_score']:
        val = row['vis_team']
    elif row['vis_score'] < row['home_score']:
        val = row['home_team']
    else:
        val = 'tie'
    return val

data['winner'] = data.apply(winner,axis=1)
```

In [4]:

```python
# We need to check to see if current row player was winner.
data['won_game'] = data.winner == data.team
```

In [5]:

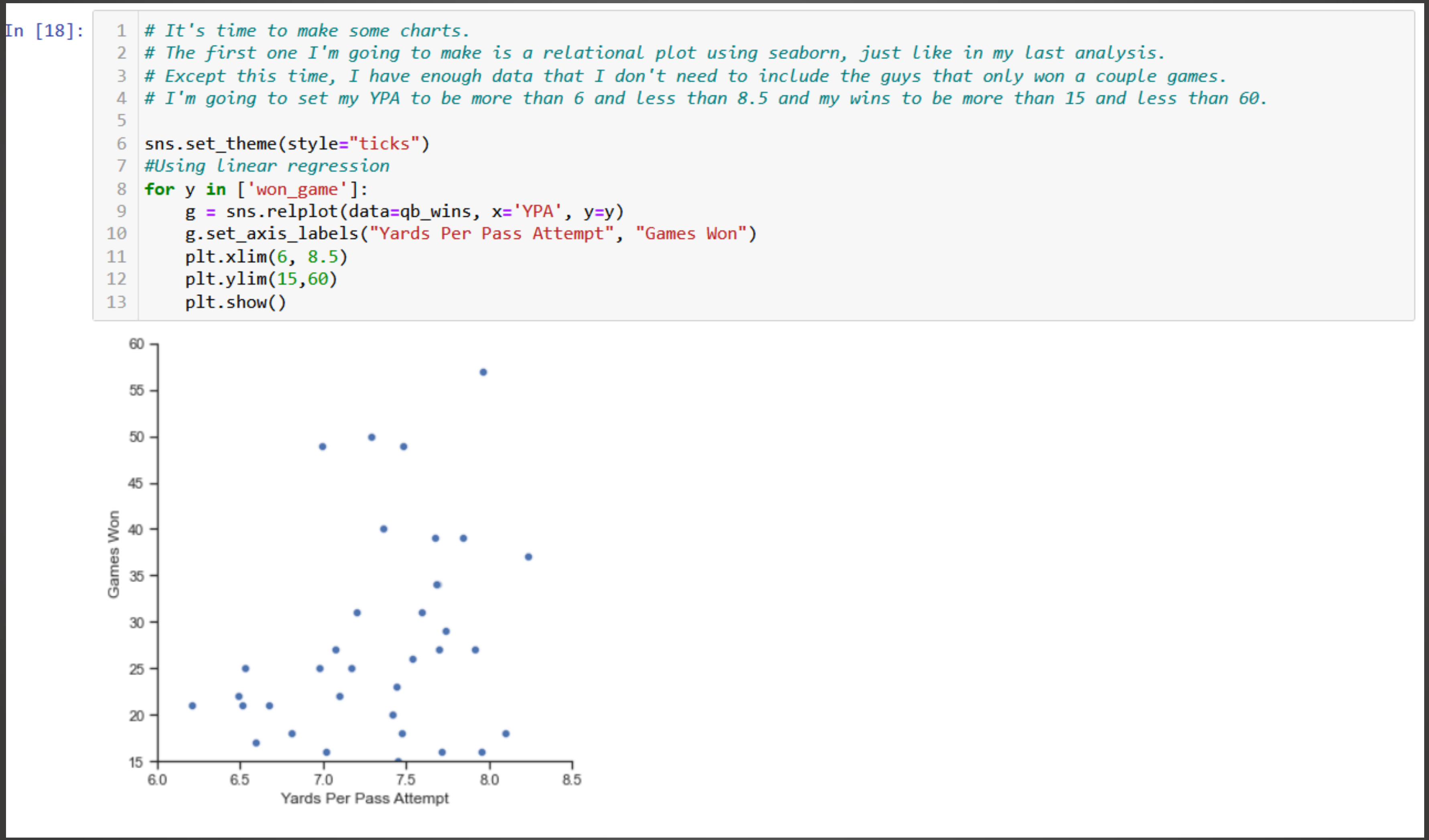
```python
# Since this is all offensive raw data, I have data for every position on that side of the ball.
# I need to define just QB stats. I also want to only see QBs with more than 146 pass attempts.
QBs = data[data.pos == 'QB'][['game_id','player','team',
                             'pass_cmp','pass_att','pass_yds',
                             'pass_td','pass_int','won_game']]

qb_stats = ['pass_cmp','pass_att', 'pass_yds', 'pass_td', 'pass_int', 'pass_sacked','player','team','pos','won_game']
groupby_qb_stats = ['player','pos']
qb_df = (data.loc[data.pos == "QB", qb_stats].groupby(groupby_qb_stats, as_index=False).sum())
qb_df['YPA']=np.round(qb_df.pass_yds / qb_df.pass_att,2)
```

In [6]:

```python
# I'm going to set the minimum total of wins to 14. More data means I need to increase my threshold
qb_wins = qb_df[(qb_df['won_game'] > 14)]
```

Charts!

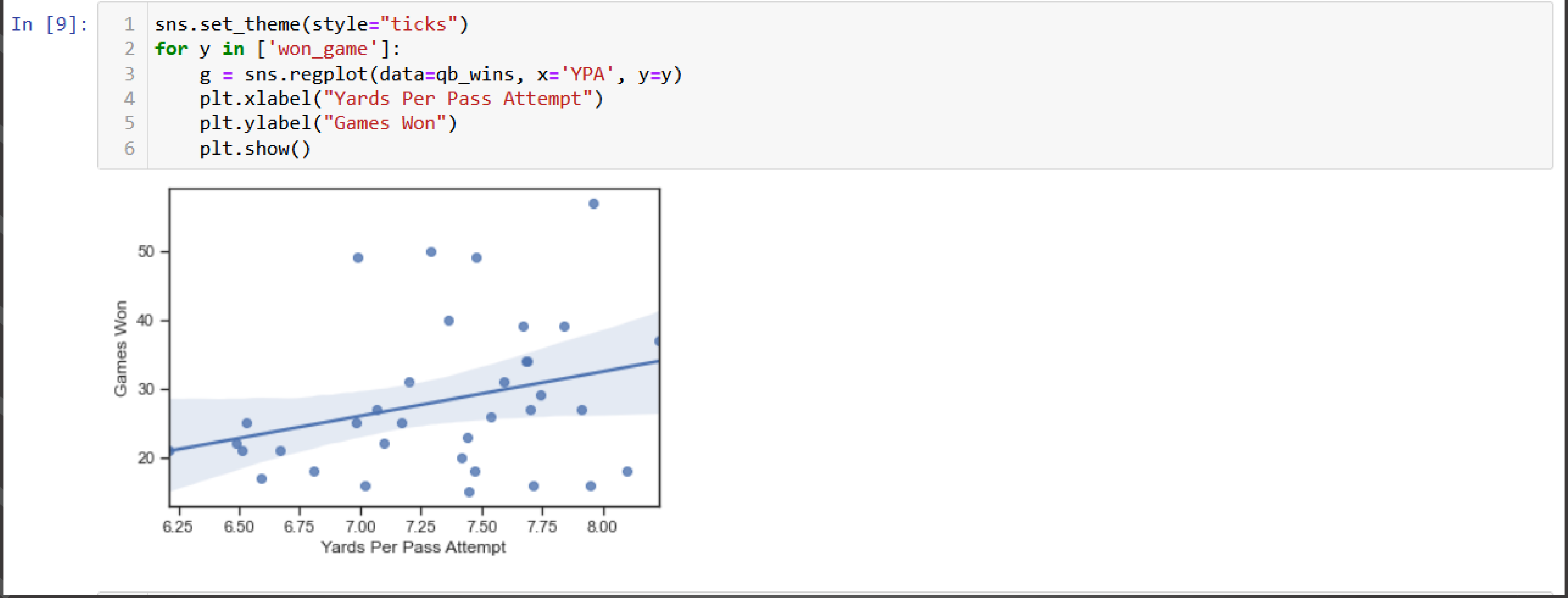


In [18]:

```
# It's time to make some charts.
# The first one I'm going to make is a relational plot using seaborn, just like in my last analysis.
# Except this time, I have enough data that I don't need to include the guys that only won a couple games.
# I'm going to set my YPA to be more than 6 and less than 8.5 and my wins to be more than 15 and less than 60.

sns.set_theme(style="ticks")
#Using linear regression
for y in ['won_game']:
    g = sns.relplot(data=qb_wins, x='YPA', y=y)
    g.set_axis_labels("Yards Per Pass Attempt", "Games Won")
    plt.xlim(6, 8.5)
    plt.ylim(15,60)
    plt.show()
```

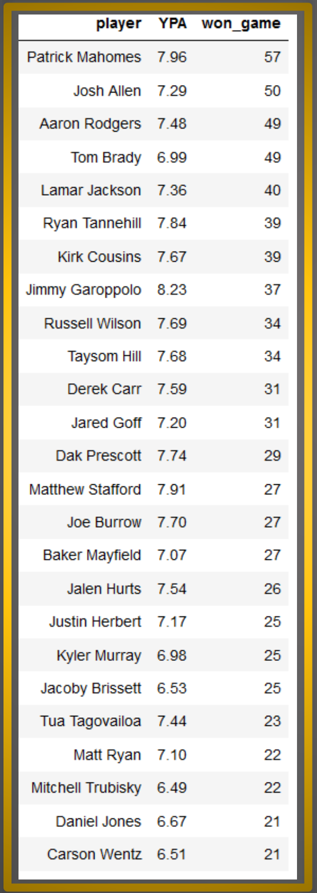
Charts!



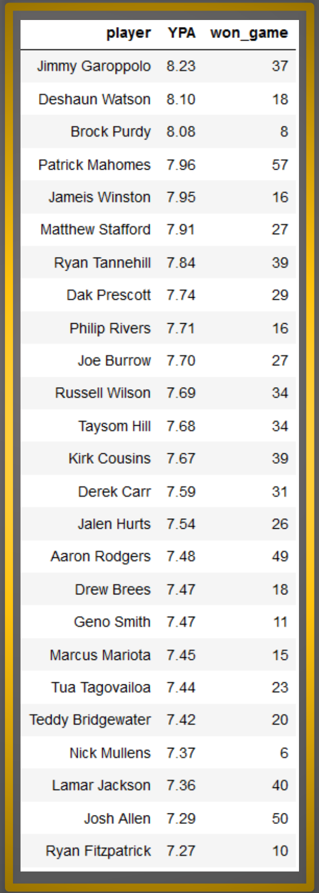
In [9]:

```
sns.set_theme(style="ticks")
for y in ['won_game']:
    g = sns.regplot(data=qb_wins, x='YPA', y=y)
    plt.xlabel("Yards Per Pass Attempt")
    plt.ylabel("Games Won")
    plt.show()
```

Comparing the Top 25 in YPA and Wins



player	YPA	won_game
Patrick Mahomes	7.96	57
Josh Allen	7.29	50
Aaron Rodgers	7.48	49
Tom Brady	6.99	49
Lamar Jackson	7.36	40
Ryan Tannehill	7.84	39
Kirk Cousins	7.67	39
Jimmy Garoppolo	8.23	37
Russell Wilson	7.69	34
Taysom Hill	7.68	34
Derek Carr	7.59	31
Jared Goff	7.20	31
Dak Prescott	7.74	29
Matthew Stafford	7.91	27
Joe Burrow	7.70	27
Baker Mayfield	7.07	27
Jalen Hurts	7.54	26
Justin Herbert	7.17	25
Kyler Murray	6.98	25
Jacoby Brissett	6.53	25
Tua Tagovailoa	7.44	23
Matt Ryan	7.10	22
Mitchell Trubisky	6.49	22
Daniel Jones	6.67	21
Carson Wentz	6.51	21



player	YPA	won_game
Jimmy Garoppolo	8.23	37
Deshaun Watson	8.10	18
Brock Purdy	8.08	8
Patrick Mahomes	7.96	57
Jameis Winston	7.95	16
Matthew Stafford	7.91	27
Ryan Tannehill	7.84	39
Dak Prescott	7.74	29
Philip Rivers	7.71	16
Joe Burrow	7.70	27
Russell Wilson	7.69	34
Taysom Hill	7.68	34
Kirk Cousins	7.67	39
Derek Carr	7.59	31
Jalen Hurts	7.54	26
Aaron Rodgers	7.48	49
Drew Brees	7.47	18
Geno Smith	7.47	11
Marcus Mariota	7.45	15
Tua Tagovailoa	7.44	23
Teddy Bridgewater	7.42	20
Nick Mullens	7.37	6
Lamar Jackson	7.36	40
Josh Allen	7.29	50
Ryan Fitzpatrick	7.27	10

Conclusion

- This season (2022) does not back up Banner’s assertion.
- Adding more data, there is more of a correlation, but still not enough

Of one to say his claim is accurate.

- Maybe adding a decade’s worth of seasons would paint a better picture,

But I’ll leave that to the folks who are paid to do this.

Dataset

- https://www.advancedsportsanalytics.com/nfl-raw-data