Web Scraping to Analyze NBA Statistics

Group 20
Austin Percy, Sydney Truong, Madan Thevar

Our Idea

We wanted to highlight the power of data and what you can do by simply extracting it from an already existing source

Data Collection and Cleaning

01. Collected data from an NBA stats database that contains information on every player and team since the 1940's

02. Organized the data in several data frames that consisted of player stats on a per-game basis
- Helped sort important data points regarding each team, player, and year.

03. Determined outlying players based on certain parameters to clean our data even further

How we got the Data

```
years = [2019, 2020, 2021, 2022, 2023, 2024] #Select the years that will fill the open brackets in the url below
url_link = 'https://www.basketball-reference.com/leagues/NBA_{}_per_game.html' #this is where the data comes from

team_high_scorers_outliers = {} #initialize

for year in years: #iterate through chosen years
    url = url_link.format(year)
    df = pd.read_html(url, header=0)[0]
    df = df.drop(df[df['Age'] == 'Age'].index)
```

```
years = [2019, 2020, 2021, 2022, 2023, 2024] #same as before, but using win percentages
url_link = 'https://www.basketball-reference.com/leagues/NBA_{}.html'

#initialize lists to store team names and W/L% data for each year
all_teams = []
all_wl_percentages = []

for year in years: #iterate through
    url = url_link.format(year)
    #read HTML tables from the URL
    df_list = pd.read_html(url)
```

What Happened Next

01

The first thing we did after cleaning our data was to start applying graphing models

02

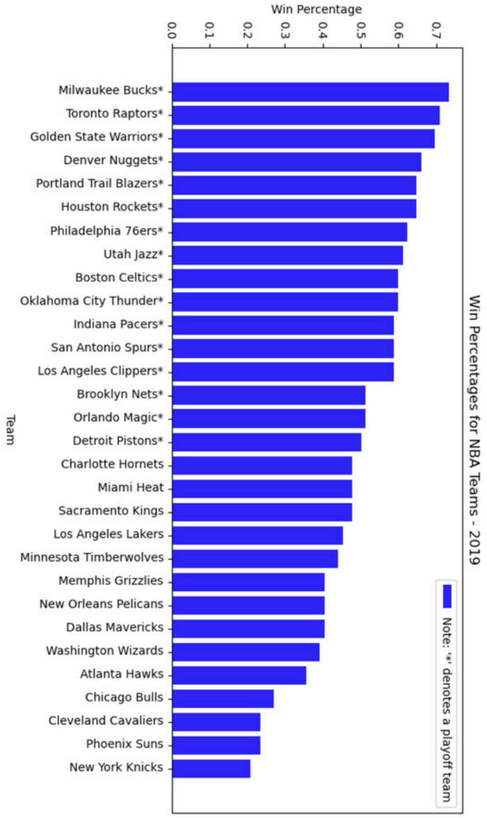
After that, we made sure everything was correct and working. We wanted to be 100% sure that the machine-learning models would work well

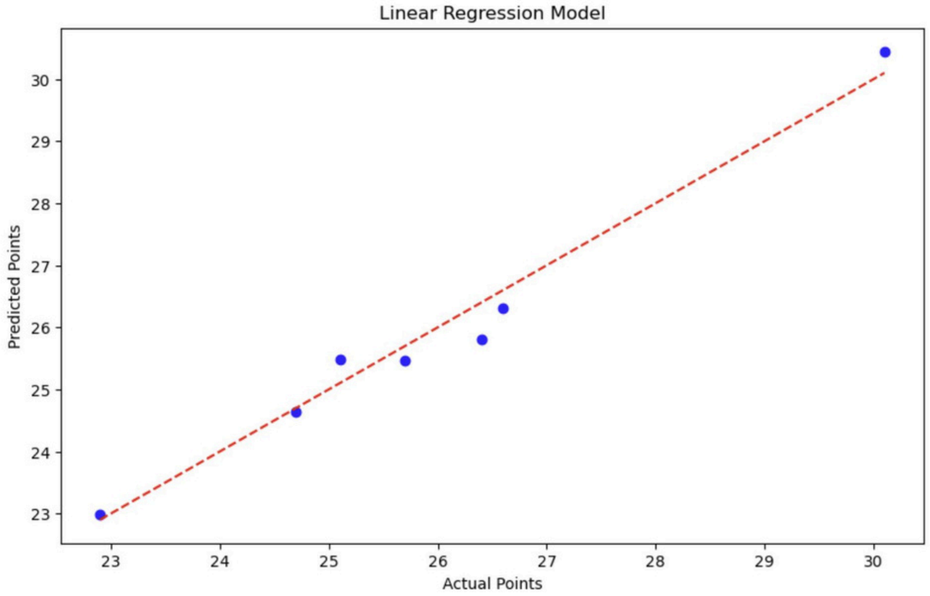
03

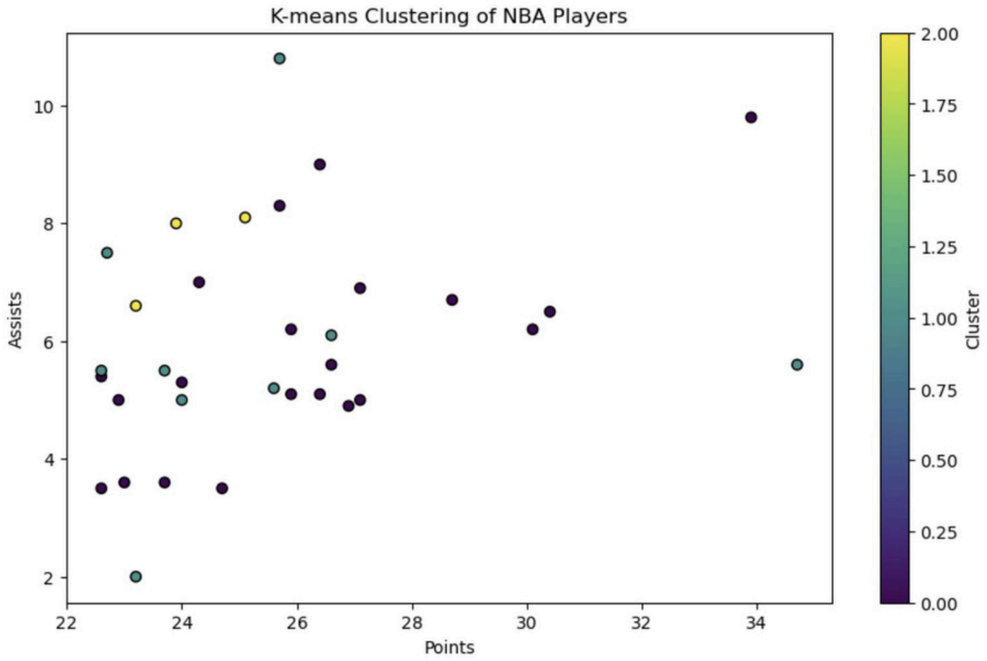
Third, we started applying various machine learning models to test the validity and accuracy of our data. We wanted to find more ways to bring value to the project

04

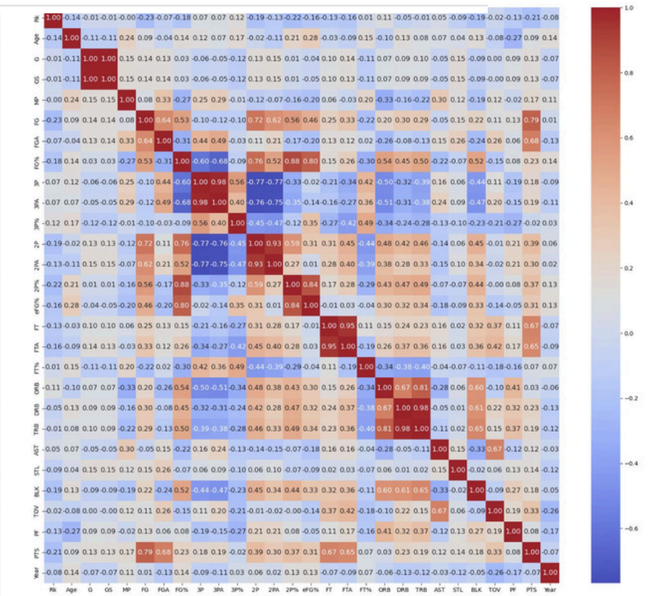
Finally, we evaluated our models and their outputs and determined how and why our project would bring value to someone in real life







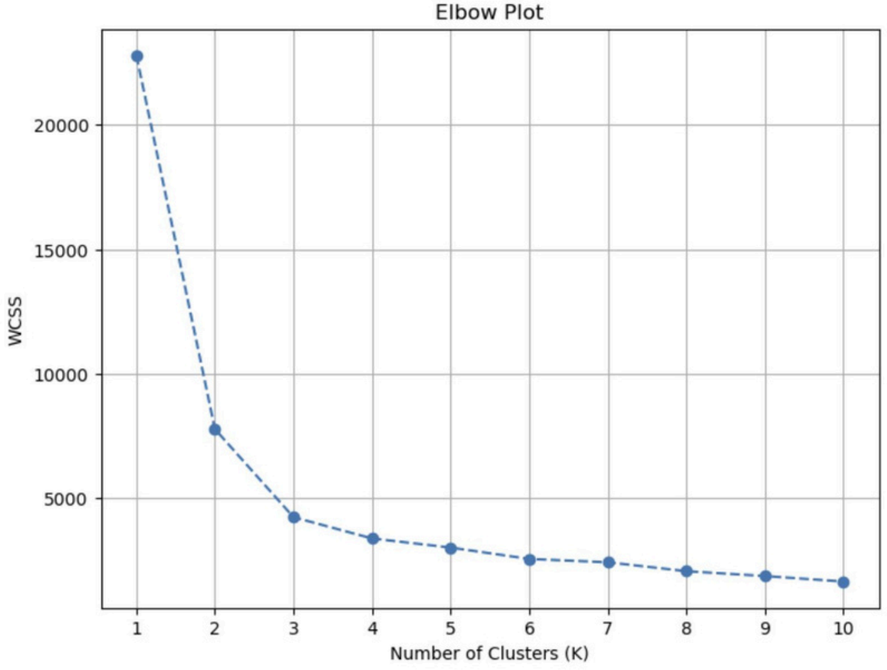
Graphs and models



Dep. Variable:	PTS	**R-squared:**	0.460
Model:	OLS	**Adj. R-squared:**	0.457
Method:	Least Squares	**F-statistic:**	141.6
Date:	Tue, 30 Apr 2024	**Prob (F-statistic):**	5.27e-24
Time:	16:12:01	**Log-Likelihood:**	-357.38
No. Observations:	168	**AIC:**	718.8
Df Residuals:	166	**BIC:**	725.0
Df Model:	1		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	5.7610	1.723	3.344	0.001	2.359	9.163
FGA	1.0765	0.090	11.900	0.000	0.898	1.255

Omnibus:	5.269	**Durbin-Watson:**	1.858
Prob(Omnibus):	0.072	**Jarque-Bera (JB):**	5.320
Skew:	0.433	**Prob(JB):**	0.0700
Kurtosis:	2.900	**Cond. No.**	209.



Why This Matters

- Player evaluation
- Performance prediction
- Fan engagement
- Scouting and talent acquisition

Challenges

- Dynamic websites
- Scraping restrictions
- Bot detection
- Privacy Policies
- Our originall idea

Thank you!