ELT Project 2

Michael Harvanek, Galen Etie, Sindhu Jayakumar, Christina Wuerz

In this project we decided to pull datasets pertaining to sports that could analyze NBA teams’ salaries and stats from 2003 to 2017. We wanted to determine if spending more money on salary made an impact on the team’s stats. There were two different websites we decided to pull data from; Kaggle and nba.com. Kaggle we were able to find a dataset already in a CSV that had team salaries and we utilized NBA.com to scrap the website to pull back the data we needed to the stats of the NBA teams and convert to CSV.

**Extracting Data in Scrape\_NBA\_Stats.ipynb**

Imported dependencies:

From IPython.core.displty import display

From bs4 import BeautifulSoup

From splinter import Browser

From splinter.exceptions import ElementDoestNotExist

Import time

NBA.com url scrape:

Defined nba.com as nba\_stats\_base\_url to be able to call it when scrapping website

Defined the seasons we were scraping the website for as seasons

Defined executable path for scrape as executable\_path

Visit first season to create data frame using rowser.visit(nba\_stats\_base\_url)time.sleep(3)

Defined html = browser.html and stats\_html = BeautifulSoupt (html, ‘thml.praser’)

Used Pandas to convert html to table using tables = pd.read\_html(str(stats\_html)),

all\_seasons\_stats\_df = tables[0] and all\_seasons\_stats\_df.insert(54, “SEASON, “2018-19”, True)

Pulled back data frame using all\_seasons\_stats\_df.head()

NBA.com scrape cleanup

Looped through all seasons and appended data frames using the following syntax:



Dropped all unwanted columns using .drop

all\_seasons\_stats\_df.drop(all\_seasons\_stats\_df.iloc[:, 28:54], inplace = True, axis = 1)

all\_seasons\_stats\_df.drop(all\_seasons\_stats\_df.columns[0], inplace = True, axis=1)

Renamed columns to match columns in database using all\_seasons\_stats\_df =

all\_seasons\_stats\_df.rename

Named the index column teamstatspk using all\_seasons\_stats\_df.index.name =

'teamstatspk'

Converted to dataframe to CVS in to use later when combining with salary data using

all\_seasons\_stats\_df.to\_csv('data/nba\_team\_stats\_00\_to\_18.csv')

**Importing data into Postgress**

Created db ins postgress:

Created a database in postgres called nba\_db

Created tables within the database called nba\_stats and nba\_salary

**Extracted data from postgress to run queries**

Imported dependencies:

from IPython.core.display import display

import pandas as pd

from sqlalchemy import create\_engine

import matplotlib.pyplot as plt

import matplotlib.cm as cm

import pandas as pd

import numpy as np

Imported data into postgress:

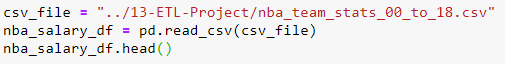
Imported the data into postgres using a new jupyternotbook called NBA\_etl

In NBA\_etl file we first set dependencies:

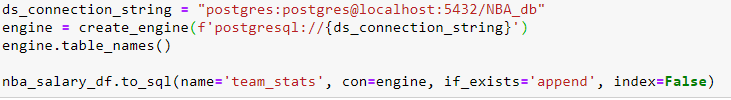
Import pandas as pd

From sqlalchemy import create\_engine

Using the following syntax we pulled the CSV for the nba team stats into a dataframe



We then took that dataframe and imported into posgress

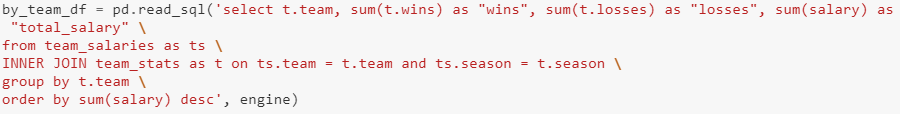


Using he same process above we also imported the nba salary data into posgress just changing out the name of the CSV files

Joined datafames that have been extracted from postgress

Need to join the nba salary and nba stats table together so we can start running queries.

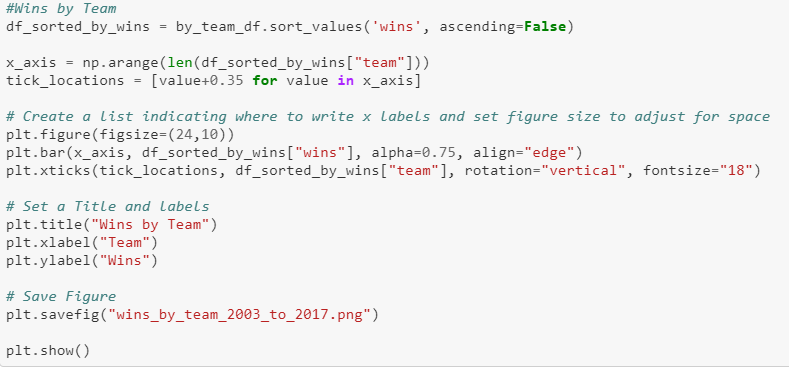
The tables were joined on team a seasion using the syntax below:

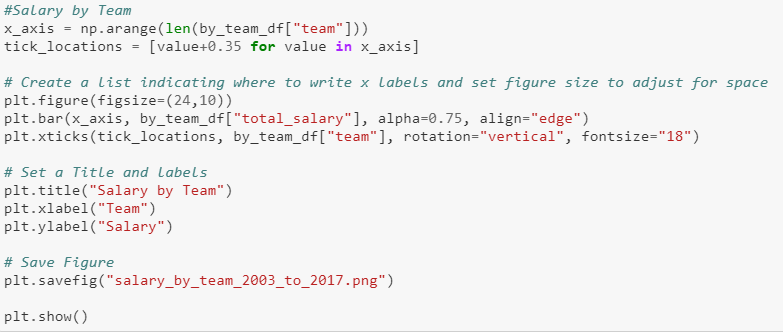


by\_team\_df we used to vizulize the data that was joined in the previous syntax

Plots of data queries

First plot was used to see the amount of wins by each team over the 2003-2017 seasons



Second plot was created to show the amount of money each team spent over the span of the 2003-2017 

The final plot we compared wins to total salary spent by each team over the 2003-2017 season so visualize how salary impacts the team stats

