

Michigan Sports Coders

Alexandra Lepore, Ava Frank, & Hannah Triester

<https://github.com/afrank58/SI206-FinalProj>



ORIGINAL GOALS

Initially our main goal was to look through data on all of the men's college basketball teams in the NCAA.

PPG

We wanted to find the NCAA players who has the most points per game on average.

Best Schools

We wanted to calculate the top 5 best men's basketball schools based on their respective records.

Midwest

And we wanted to calculate how many Michigan players were from the midwest.



ACHIEVED GOALS

Our main achieved goal for this project was to examine some of the stats about the players on the three most popular sports teams at Michigan: Men's Basketball, Football, and Hockey.

In State Players

One of our goals was to calculate how many players were from in state.

Grades

Another goal of ours was to figure out the grade / age distribution of players on each team.

Average Twitter Followers

The goal of the this part of the project was to find the amount of average followers of a Twitter user that tweets about each player.



PROBLEMS WE FACED

- We initially found it difficult to find APIs with viable documentation. Additionally, many APIs with information we needed were not available to the public, forcing us to use websites we did not expect when we planned our data collection.
- Because we collected data for three different teams, we had to access different web pages and different API endpoints for each. This added extra steps to our database creation and calculations.
- We faced some complications with our calculations as well as some of the data we needed was in tuples, and had tons of data to parse through, so we had to figure out how to most effectively do our calculations.



Finding the Number of In State Players

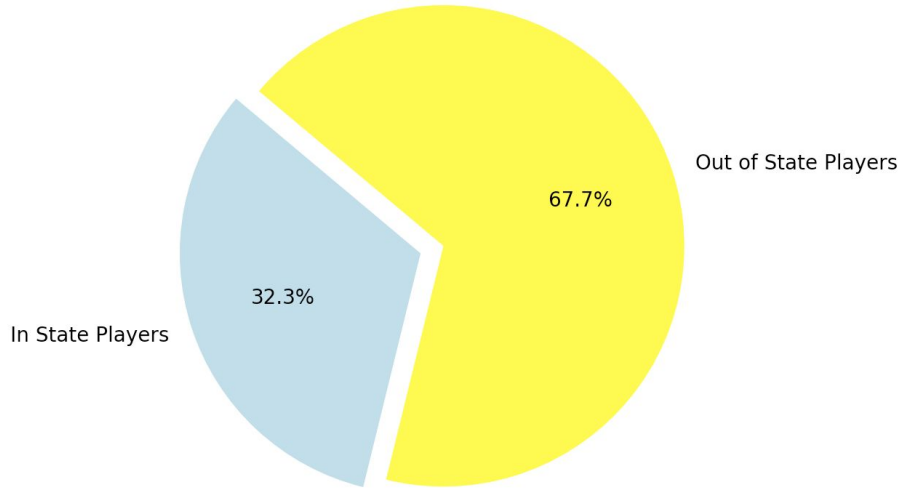
Calculation File

≡ Players.txt

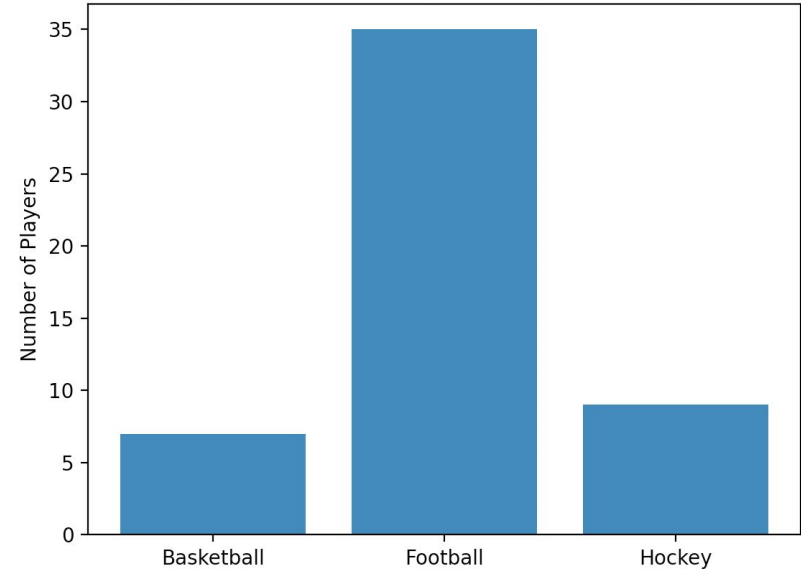
```
1 Percentage of athletes that are from Michigan on the Men's Basketball, Football, and Hockey Teams:
2 0.3227848101265823 of Michigan Men's Basketball, Football, and Hockey players are from the state of Michigan.
3
4 Number of in state players on each of the three teams:
5 There are 7 in state players on the Basketball team.
6 There are 35 in state players on the Football team.
7 There are 9 in state players on the Hockey team.
8
```

Data Visualizations

Percentage of UMich Men's Basketball, Football, and Hockey Players that are from in state versus out of state



Number of In State Players on the Michigan Men's Basketball, Football, and Hockey Teams



Databases

Sports

	sport_id	sport_name
	Filter	Filter
1	1	Basketball
2	2	Football
3	3	Hockey

Players

	sport_id	name	hometown	homestate
	Filter	Filter	Filter	Filter
1	1	Adrien Nunez	Brooklyn	N.Y.
2	1	Hunter Dickinson	Alexandria	Va.
3	1	Isaiah Livers	Kalamazoo	Mich.
4	1	Zeb Jackson	Toledo	Ohio
5	1	Brandon Wade	Ann Arbor	Mich.
6	1	Terrance Williams II	Clinton	Md.
7	1	Mike Smith	Burr Ridge	Ill.
8	1	Rico Ozuna-Harrison	Detroit	Mich.
9	1	Chaundee Brown Jr.	Orlando	Fla.
10	1	Franz Wagner	Berlin	Germany
11	1	Brandon Johns Jr.	East Lansing	Mich.
12	1	C.J. Baird	Novi	Mich.
13	1	Jace Howard	Miami	Fla.

14	1	Luke Wilson	Boulder	Colo.
15	1	Jaron Faulds	Holt	Mich.
16	1	Austin Davis	Onsted	Mich.
17	1	Eli Brooks	Spring Grove	Pa.
18	2	Erick All	Fairfield	Ohio
19	2	Willie Allen	River Ridge	La.
20	2	Raheem Anderson	Detroit	Mich.
21	2	Lucas Andrighetto	Mountain View	Calif.
22	2	Andrel Anthony	East Lansing	Mich.
23	2	Reece Atteberry	Aurora	Colo.
24	2	Karsen Barnhart	Paw Paw	Mich.
25	2	Michael Barrett	Valdosta	Ga.

Code Documentation

```
def get_player_towns_states(tup_lst):  
    '''Takes in a list of tuples of the sport name and the roster URL for Michigan men's  
    basketball, football and hockey.  
    Returns a list of tuples that contains sport, name, hometown, and home state for players  
    on the Michigan men's basketball, football and hockey teams. '''  
  
def get_players_in_state(tup_list):  
    '''Takes in the tuple list returned in get_player_towns_states.  
    Returns a tuple list in the same format, but only with players from Michigan.'''  
  
def setUpDatabase(db_name):  
    '''Takes the name of a database as an input.  
    Returns the cursor and connections to the database.'''  
  
def setUpSportsTable(cur, conn):  
    '''Takes in the database cursor and connection as inputs.  
    Returns nothing, but creates the Sports table which holds the holds and their id numbers'''
```

Code Documentation

```
def setUpPlayersTable(tup_lst, cur, conn):  
    '''Takes in the tuple list returned in get_player_towns_states, the database cursor and connection.  
    Returns nothing, but creates the Players table which contains the sport id number, name, hometown,  
    and home state for each player.'''  
  
def get_per_in_state(cur, conn):  
    '''Takes in the database cursor and connection as inputs.  
    Returns the percentage of players that are from in state.'''  
  
def counting_in_state_players(in_state_tup_list):  
    '''Takes in the tuple list returned in get_players_in_state.  
    Returns a dictionary with the sport names as keys and the number of in state players on that particular team.'''  
  
def write_data_to_file(filename, cur, conn):  
    '''Takes in a filename, the database cursor, and connection as inputs.  
    Created a file and write the calculations from get_per_in_state and counting_in_state_players.'''  
  
def main():  
    '''Takes nothing as an input and returns sports. Calls the functions and creates the visualizations.'''
```



Class / Age Distribution for Each Team

Note: “hockey.py” is a misnomer

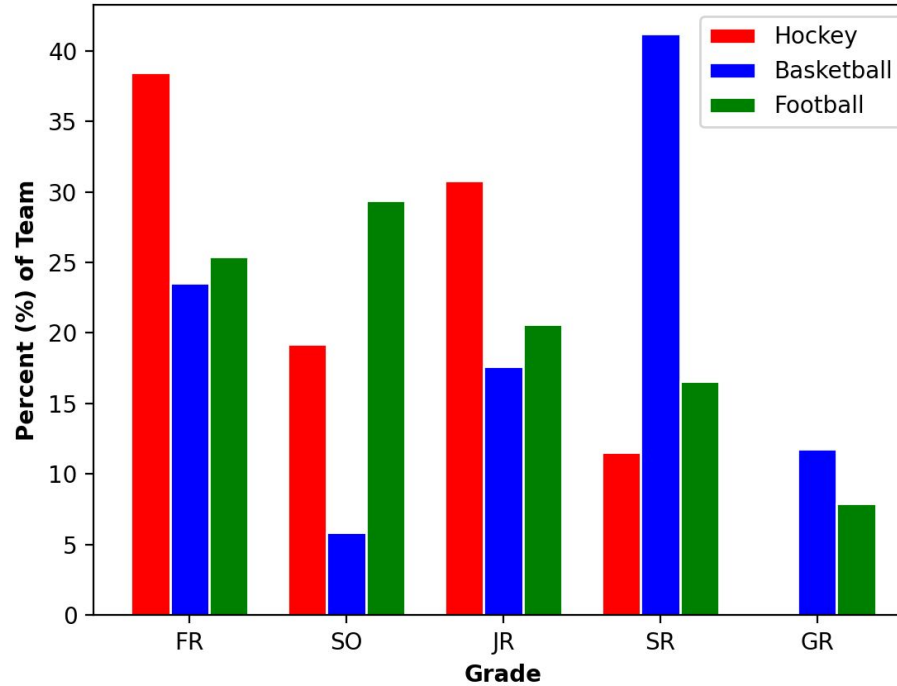
Calculation File

≡ Years.txt

```
1 |The percent of freshmen on the hockey team is 0.38461538461538464.
2 |The percent of sophomores on the hockey team is 0.19230769230769232.
3 |The percent of juniors on the hockey team is 0.3076923076923077.
4 |The percent of seniors on the hockey team is 0.11538461538461539.
5 |The percent of graduates on the hockey team is 0.0.
6 |The percent of freshmen on the basketball team is 0.23529411764705882.
7 |The percent of sophomores on the basketball team is 0.058823529411764705.
8 |The percent of juniors on the basketball team is 0.17647058823529413.
9 |The percent of seniors on the basketball team is 0.4117647058823529.
10 |The percent of graduates on the basketball team is 0.11764705882352941.
11 |The percent of freshmen on the football team is 0.24603174603174602.
12 |The percent of sophomores on the football team is 0.29365079365079366.
13 |The percent of juniors on the football team is 0.20634920634920634.
14 |The percent of seniors on the football team is 0.16666666666666666.
15 |The percent of graduates on the football team is 0.0873015873015873.
```

Data Visualizations

Class Distribution of Michigan Sports Team



Code Documentation

```
get_hockey_data()
```

"No inputs. Returns a list of tuples in the format (sport, first name, last name, year). Uses the sportradar.us API to retrieve values for each player on Michigan Wolverines Hockey."

```
get_basketball_data()
```

"No inputs. Returns a list of tuples in the format (sport, first name, last name, year). Uses the sportradar.us API to retrieve values for each player on Michigan Wolverines Basketball."

```
get_football_data()
```

"No inputs. Returns a list of tuples in the format (sport, first name, last name, year). Uses the sportradar.us API to retrieve values for each player on Michigan Wolverines Football."

```
accumulate_data(hockey_lst, bball_lst, football_lst)
```

"Inputs the list of tuples that are returned from get_hockey_data(), get_basketball_data(), and get_football_data(). Aggregates the returned tuples from all three teams into a single list."

Code Documentation

```
setUpDatabase(db_name)
```

"Takes the name of a database, a string, as an input. Returns the cursor and connection to the database."

```
setUpYearsTable(tup_lst, cur, conn)
```

"Takes the list of tuples returned from `accumulate_data()`, the database cursor, and connection as inputs. Returns nothing. Creates the `Wolverines_Years` database, which holds the information on each player."

```
setUpSportsTable(tup_lst, cur, conn)
```

"Takes the list of tuples that matches a sport to an id number, the database cursor, and connection as inputs. Returns nothing. Creates the `Sports` database, which matches the sport title to an id # (1, 2, 3)."

```
setUpIDsYearsTable(cur, conn)
```

"Method name is a misnomer. Selects data for each sport and joins the `Wolverines_Years` table with the `Sports` table on the sport title. Calculates the grade distribution for each team."

Code Documentation

```
write_data_to_file(filename, cur, conn)
```

"Takes in a filename (string), the database cursor, and connection as inputs. Creates a file and writes the calculations from setUpIDsYearsTable() into a file."

```
main()
```

"Takes nothing as an input and returns nothing. Calls the functions setUpDatabase(), setUpYearsTable(), setUpSportsTable(), setUpIDsYearsTable(), write_data_to_file() and includes the code for the bar graph visualization."

Tables created from "hockey.py"

Table: Sports

	sport_title	sport_id
	Filter	Filter
1	Hockey	1
2	Basketball	2
3	Football	3

Table: Wolverines_Years

	sport	first_name	last_name	year
	Filter	Filter	Filter	Filter
1	Hockey	Dakota	Raabe	SR
2	Hockey	Jay	Keranen	SO
3	Hockey	Luke	Morgan	SR
4	Hockey	Strauss	Mann	JR
5	Hockey	Nick	Blankenburg	JR
6	Hockey	Jacob	Truscott	FR
7	Hockey	Jake	Gingell	JR
8	Hockey	Matthew	Beniers	FR
9	Hockey	Brendan	Brisson	FR
10	Hockey	Thomas	Bordeleau	FR
11	Hockey	Josh	Groll	FR
12	Hockey	Nicholas	Granowicz	SO
13	Hockey	Jack	Leavy	JR
14	Hockey	Erik	Portillo	FR
15	Hockey	Keaton	Pehrson	SO
16	Hockey	Jack	Summers	JR
17	Hockey	Nolan	Moyle	JR
18	Hockey	John	Beecher	SO
19	Hockey	Philippe	LaPointe	FR
20	Hockey	Steve	Holtz	FR

1 - 20 of 169



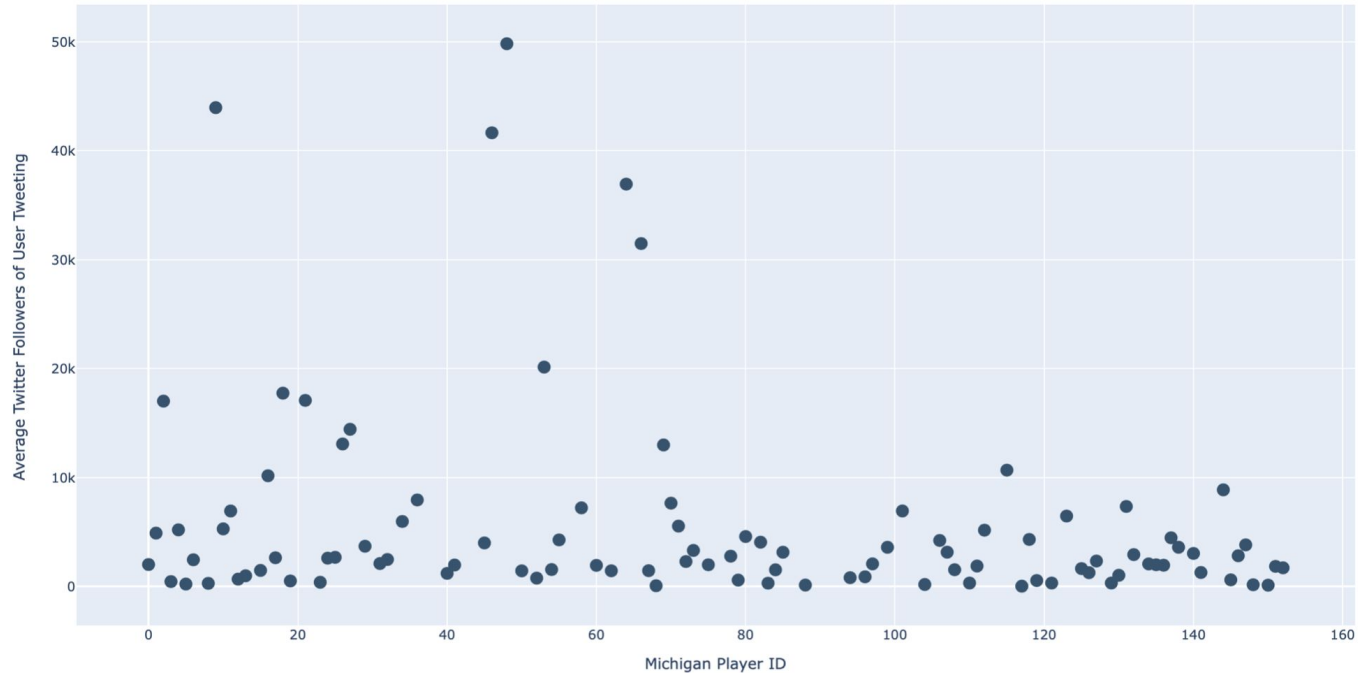
Finding the Average Amount of Twitter Followers a Person Who Tweets about Each Athlete Has

Calculation File

```
The average amount of followers of a Twitter user that Tweets about the Michigan athlete with a player id = 0 is 2015.3333333333333.
The average amount of followers of a Twitter user that Tweets about the Michigan athlete with a player id = 1 is 4897.789473684211.
The average amount of followers of a Twitter user that Tweets about the Michigan athlete with a player id = 2 is 17017.933333333334.
The average amount of followers of a Twitter user that Tweets about the Michigan athlete with a player id = 3 is 449.5.
The average amount of followers of a Twitter user that Tweets about the Michigan athlete with a player id = 4 is 5199.375.
The average amount of followers of a Twitter user that Tweets about the Michigan athlete with a player id = 5 is 220.0.
The average amount of followers of a Twitter user that Tweets about the Michigan athlete with a player id = 6 is 2451.89.
The average amount of followers of a Twitter user that Tweets about the Michigan athlete with a player id = 8 is 285.0.
The average amount of followers of a Twitter user that Tweets about the Michigan athlete with a player id = 9 is 43967.3.
The average amount of followers of a Twitter user that Tweets about the Michigan athlete with a player id = 10 is 5285.714285714285.
The average amount of followers of a Twitter user that Tweets about the Michigan athlete with a player id = 11 is 6929.166666666667.
The average amount of followers of a Twitter user that Tweets about the Michigan athlete with a player id = 12 is 672.3333333333334.
The average amount of followers of a Twitter user that Tweets about the Michigan athlete with a player id = 13 is 983.37.
The average amount of followers of a Twitter user that Tweets about the Michigan athlete with a player id = 15 is 1473.53.
The average amount of followers of a Twitter user that Tweets about the Michigan athlete with a player id = 16 is 10166.64.
The average amount of followers of a Twitter user that Tweets about the Michigan athlete with a player id = 17 is 2636.69.
The average amount of followers of a Twitter user that Tweets about the Michigan athlete with a player id = 18 is 17748.23.
The average amount of followers of a Twitter user that Tweets about the Michigan athlete with a player id = 19 is 507.0.
The average amount of followers of a Twitter user that Tweets about the Michigan athlete with a player id = 21 is 17085.75.
The average amount of followers of a Twitter user that Tweets about the Michigan athlete with a player id = 23 is 384.0.
The average amount of followers of a Twitter user that Tweets about the Michigan athlete with a player id = 24 is 2597.9.
The average amount of followers of a Twitter user that Tweets about the Michigan athlete with a player id = 25 is 2673.894117647059.
The average amount of followers of a Twitter user that Tweets about the Michigan athlete with a player id = 26 is 13084.207547169812.
The average amount of followers of a Twitter user that Tweets about the Michigan athlete with a player id = 27 is 14430.857142857143.
The average amount of followers of a Twitter user that Tweets about the Michigan athlete with a player id = 29 is 3692.3529411764707.
```

Data Visualizations

Average Amount of Followers of a Twitter User that Tweets About Michigan Men's Football, Hockey, and Basketball Players



Code Documentation

```
def setUpDatabase(db_name):  
    """Takes the name of a database, a string, as input. Returns the cursor connection to  
    the database."""  
  
def get_data_lst(cur, conn):  
    """Takes in the database cursor and connection as an input. Gets a list of names from  
    the Player database. Returns a list of tuples of athlete name, Twitter user ID,  
    and the user's follower count."""  
  
def set_up_athlete_table(cur, conn):  
    """Takes in the database cursor and connection as inputs. Returns nothing. Creates the  
    Player_ID table and fills it with the player names and their player_ids. The player_id  
    are unique identification numbers for each player."""  
  
def get_data(tup_lst, cur, conn):  
    """This function takes in a the list of tuples returned in get_data_lst, the database  
    cursor, and the database connection as inputs. Returns nothing. Creates the  
    Michigan_Twitter_Data table and fills it with the player_id, user_id of a Twitter user  
    that Tweets about that athlete, and that user's follower count."""
```

Code Documentation

```
def calc_avg_followers(cur, conn):  
    """Takes in the database cursor and connection as inputs. Returns a dictionary of  
    player_ids as the key and the average follower count for a user that Tweets about them."""  
  
def write_data_to_file(filename, cur, conn):  
    """Takes in a filename (string), the database cursor, and the database connection  
    as inputs. Returns nothing. Creates a file and writes the return values of the  
    calc_avg_followers function to the file."""  
  
def main():  
    """Takes nothing as input and returns nothing. Calls the functions setUpDatabase(),  
    set_up_athlete_table(), get_data(), get_data_lst(), calc_avg_followers(),  
    write_data_to_file() and includes the code for the scatterplot visualization graph."""
```

Instructions for Running the Code

1. Make sure the “Wolverines_Years.db” and “Players.db” databases do not already exist in your files. If it does, delete it so that it can be reloaded.
2. The first file that should run is “Webscrape.py.” This file should be run seven times, inputting 25 data items at a time to finally input all 158 and compute the correct calculations and make the visualisations.
3. The second file that should run is “FinalProject.py”. This file should be run once due to the sheer amount of data it collects. The database will add the player_ID (with 158 rows) and Michigan_Twitter_Data tables (with over 4000 rows). To see the calculations from “FinalProject.py”, open “AverageFollowers.txt”. The visualization for average followers per user that Tweets about each athlete should also come from this file.
4. The third file that should be run is called “hockey.py” (name is a misnomer). This file should be run six times. The database will create a table “Wolverines_Years” and “Sports”. The first time, “Sports” will be filled completely. This table will be joined with the other to complete our calculation. “Wolverines_Years” fills in 25 data rows at a time, until 169 rows are filled. To see the calculations made from “hockey.py,” open “Years.txt”. The visualization for year distribution of each team also comes from this file.

Documentation of Resources Used

Date	Issue Description	Location of Resource	Result (did it solve the issue?)	
4/20	How to get authentication on twitter	https://developer.twitter.com/en/docs/authentication/oauth-1-0a	Yes	
4/21	How to put a title on a matplotlib pie chart	https://www.w3resource.com/graphics/matplotlib/piechart/matplotlib-piechart-exercise-2.php	Yes	
4/21	How to resolve terminal error that was printing	https://stackoverflow.com/questions/40703228/python-sqlite3-operationalerror-attempt-to-write-a-read-only-database	Yes	
4/21	How to resolve another terminal issue when code ran	https://stackoverflow.com/questions/21231264/python-sqlite-interfaceerror-error-binding-parameter-0-probably-unsupported	Yes	
4/21	How to fix terminal error	https://stackoverflow.com/questions/58020114/failed-to-send-request-only-unicode-objects-are-escapeable-got-none-of-type-cl	Yes	
4/22	How to make grouped bar graph	https://matplotlib.org/stable/gallery/lines_bars_and_markers/barchart.html#sphx-glr-gallery-lines-bars-and-markers-barchart-py	No	
4/22	How to make grouped bar graph	https://www.python-graph-gallery.com/11-grouped-barplot	Yes	



Link to github repository

<https://github.com/afrank58/SI206-FinalProj>