project-group16-proposal

October 29, 2021

1 CSCI2000U - Scientific Data Analysis

2 Final Project: Proposal

2.1 Project Group #: 16

2.1.1 Group Members & Student Numbers:

- **Mohammad** - 100755461
- **Hasan Chakaroun** - 100788546
- **Eihab Syed** - 100707448
- **Preet Panchal** - 100707094

2.1.2 Dataset Selected:

- **Name:** NBA Players Data - all_seasons.csv
- **Source:** https://www.kaggle.com/justinas/nba-players-data
- **Creator:** Justinas Cirtautas, Data Scientist

2.2 Description of Dataset:

The chosen dataset contains over two decades of statistics about each player that has been a part of the NBA (National Basketball Association). It is comprised of many variables including player age, height, weight, college attended, country born and game box scores. There are 22 data attributes in total offering a great range for data analysis.

It was collected through the official NBA website and https://www.basketball-reference.com/ found via Kaggle from Justinas Cirtautas (https://www.kaggle.com/justinas/nba-players-data). This dataset represents the the details of each NBA player for each season from 1996 to 2021.

2.2.1 Here are the 22 attributes/columns in this dataset:

- index - Player index / Row number
- player_name - Name of player
- team_abbreviation - Abbreviated name of the team the player played for
- age - Age of player
- player_height - Height of the player (cm)
- player_weight - Weight of the player (kg)
- college - Name of the college player attended
- country - Name of the country player was born in

- draft_year - The year the player was drafted
- draft_round - The draft round the player was picked
- draft_number - The draft number at which the player was picked in his draft round
- gp - Games played throughout the season
- pts - Average number of points scored
- reb - Average number of rebounds grabbed
- ast - Average number of assists distributed
- net_rating - Team's point differential/100 possessions while player is on court
- oreb_pct - % of available offensive rebounds the player grabbed on court
- dreb_pct - % of available defensive rebounds the player grabbed on court
- usg_pct - % of team plays used by the player while on court
- ts_pct - Measure of player's shooting efficiency
- ast_pct - % of teammate field goals the player assisted on court
- season - NBA season

2.3 Code & Analysis of Basic Characteristics of Dataset:

[2]:
```
#importing used libraries
import csv
import re
from functools import reduce
import numpy as np

# this aux function reads the CSV file and returns the data in a Python␣
 ↪dictionary
def get_data_csv():
    collection = []
    with open('all_seasons.csv', 'r') as f:
        for line in csv.DictReader(f):
            collection.append(line)
        return collection

# the data
data = get_data_csv()
```

[3]:
```
# displaying the total number of data using string and len inbuilt python␣
 ↪function
print("There are " + str(len(data)) + " data records in all_seasons.csv. ")
```

```
There are 11700 data records in all_seasons.csv.
```

[4]:
```
print("Q2) Here are the top 5 data records: ")
print(data[:5]) # using indexing to display the first 5 records
```

```
Q2) Here are the top 5 data records:
[{'': '0', 'player_name': 'Travis Knight', 'team_abbreviation': 'LAL', 'age':
'22.0', 'player_height': '213.36', 'player_weight': '106.59411999999999',
'college': 'Connecticut', 'country': 'USA', 'draft_year': '1996', 'draft_round':
```

```
'1', 'draft_number': '29', 'gp': '71', 'pts': '4.8', 'reb': '4.5', 'ast': '0.5',
'net_rating': '6.2', 'oreb_pct': '0.127', 'dreb_pct': '0.182', 'usg_pct':
'0.142', 'ts_pct': '0.536', 'ast_pct': '0.052000000000000005', 'season':
'1996-97'}, {'': '1', 'player_name': 'Matt Fish', 'team_abbreviation': 'MIA',
'age': '27.0', 'player_height': '210.82', 'player_weight': '106.59411999999999',
'college': 'North Carolina-Wilmington', 'country': 'USA', 'draft_year': '1992',
'draft_round': '2', 'draft_number': '50', 'gp': '6', 'pts': '0.3', 'reb': '0.8',
'ast': '0.0', 'net_rating': '-15.1', 'oreb_pct': '0.14300000000000002',
'dreb_pct': '0.267', 'usg_pct': '0.265', 'ts_pct': '0.33299999999999996',
'ast_pct': '0.0', 'season': '1996-97'}, {'': '2', 'player_name': 'Matt Bullard',
'team_abbreviation': 'HOU', 'age': '30.0', 'player_height': '208.28',
'player_weight': '106.59411999999999', 'college': 'Iowa', 'country': 'USA',
'draft_year': 'Undrafted', 'draft_round': 'Undrafted', 'draft_number':
'Undrafted', 'gp': '71', 'pts': '4.5', 'reb': '1.6', 'ast': '0.9', 'net_rating':
'0.9', 'oreb_pct': '0.016', 'dreb_pct': '0.115', 'usg_pct': '0.151', 'ts_pct':
'0.535', 'ast_pct': '0.099', 'season': '1996-97'}, {'': '3', 'player_name':
'Marty Conlon', 'team_abbreviation': 'BOS', 'age': '29.0', 'player_height':
'210.82', 'player_weight': '111.13004', 'college': 'Providence', 'country':
'USA', 'draft_year': 'Undrafted', 'draft_round': 'Undrafted', 'draft_number':
'Undrafted', 'gp': '74', 'pts': '7.8', 'reb': '4.4', 'ast': '1.4', 'net_rating':
'-9.0', 'oreb_pct': '0.083', 'dreb_pct': '0.152', 'usg_pct':
'0.16699999999999998', 'ts_pct': '0.542', 'ast_pct': '0.10099999999999999',
'season': '1996-97'}, {'': '4', 'player_name': 'Martin Muursepp',
'team_abbreviation': 'DAL', 'age': '22.0', 'player_height': '205.74',
'player_weight': '106.59411999999999', 'college': 'None', 'country': 'USA',
'draft_year': '1996', 'draft_round': '1', 'draft_number': '25', 'gp': '42',
'pts': '3.7', 'reb': '1.6', 'ast': '0.5', 'net_rating': '-14.5', 'oreb_pct':
'0.109', 'dreb_pct': '0.11800000000000001', 'usg_pct': '0.233', 'ts_pct':
'0.48200000000000004', 'ast_pct': '0.114', 'season': '1996-97'}]
```

[5]:
```
# creating a set with all the player names from data to only include each␣
 ↪unique team with no repeats
unique_name = {player['player_name'] for player in data}
# then displaying the length of unique_name as a string
print("There are " + str(len(unique_name)) + " unique NBA player names in the␣
 ↪dataset. Therefore, there are " + str(len(unique_name)) + " players that␣
 ↪played in the National Basketball Association since 1996.\n")
print("Here are the first 100 player names from the unique_name dict. (List was␣
 ↪too long to print completely): ")
print(list(unique_name)[:100])
```

```
There are 2333 unique NBA player names in the dataset. Therefore, there are 2333
players that played in the National Basketball Association since 1996.

Here are the first 100 player names from the unique_name dict. (List was too
long to print completely):
['Jerome Kersey', 'Ha Ha', 'Ray Allen', 'Nate Wolters', 'Kevin Martin', 'Reggie
Slater', 'Ian Mahinmi', "Devonte' Graham", 'William Howard', 'Kobe Bryant',
```

```
'Ervin Johnson', 'Spencer Dinwiddie', 'Terence Davis', 'Avery Johnson', 'Alex
Poythress', 'Alec Peters', 'Trey Thompkins', 'Hakeem Olajuwon', 'Kyle Weaver',
'Deyonta Davis', 'Tremont Waters', 'Charlie Brown Jr.', 'Evan Fournier',
'Antonio Burks', 'Quentin Richardson', 'Norris Cole', 'Pascal Siakam', 'Dwight
Powell', 'Raul Lopez', 'Jaden McDaniels', 'Yaroslav Korolev', 'Chris Singleton',
'Dylan Windler', 'Kenrich Williams', 'Viacheslav Kravtsov', 'Naz Reid', 'Lamar
Stevens', 'Scott Williams', 'Boris Diaw', 'Trevor Winter', 'Wayne Simien',
'Alton Ford', 'Cory Joseph', 'Justin Jackson', 'Jeremy Richardson', 'Corey
Maggette', 'Gerald Brown', 'Paul Pierce', 'Felipe Lopez', 'Willie Burton',
'Jarnell Stokes', 'Tobias Harris', 'Jason Thompson', 'Lynn Greer', 'Andre
Brown', 'Jamal Crawford', 'Ike Fontaine', 'Yinka Dare', 'Will Solomon', 'Axel
Toupane', 'Mile Ilic', 'Jonas Valanciunas', 'Jabari Bird', 'Jalen Rose', 'Jordan
Poole', 'Dionte Christmas', 'Cam Reddish', 'Kurt Thomas', 'Josh Green', 'Brandon
Williams', 'Pavel Podkolzin', 'Trevor Ariza', 'Scott Machado', 'Sean May', 'Maxi
Kleber', 'Landry Shamet', 'Billy Garrett', 'Emanual Davis', 'Rick Brunson',
'Trajan Langdon', 'Tony Snell', 'Isaiah Roby', 'Rex Walters', 'Ricky Ledo',
'Bryon Russell', 'Maalik Wayns', 'Donatas Motiejunas', 'Fred Jones', 'Quincy
Lewis', 'Stephen Curry', 'Thon Maker', 'Jonny Flynn', 'Ndudi Ebi', 'Guillermo
Diaz', 'Vassilis Spanoulis', 'Kelly Olynyk', 'Greg Anderson', 'Oliver
Lafayette', 'Elijah Millsap', 'Mike Bibby']
```

[6]:
```
# creating a set with all the teams from data to only include each unique team␣
 ↪with no repeats
unique_team = {team['team_abbreviation'] for team in data}
# then displaying the length of unique_team as a string
print("Not all team names in the dataset are unique. There are " +␣
 ↪str(len(unique_team)) + " unique NBA teams in the dataset. Therefore, there␣
 ↪are " + str(len(unique_team)) + " teams that take part in the National␣
 ↪Basketball Association since 1996.")
print(unique_team)
```

```
Not all team names in the dataset are unique. There are 36 unique NBA teams in
the dataset. Therefore, there are 36 teams that take part in the National
Basketball Association since 1996.
{'SEA', 'PHI', 'WAS', 'NJN', 'CHH', 'CLE', 'NOH', 'CHA', 'DEN', 'DET', 'BKN',
'GSW', 'NYK', 'TOR', 'POR', 'SAC', 'VAN', 'HOU', 'MIN', 'CHI', 'PHX', 'UTA',
'MIA', 'ORL', 'IND', 'SAS', 'NOP', 'LAL', 'MIL', 'DAL', 'BOS', 'OKC', 'NOK',
'LAC', 'ATL', 'MEM'}
```

[7]:
```
# creating a set with all the colleges from data to only include each unique␣
 ↪college with no repeats
unique_college = {player['college'] for player in data}
# then displaying the length of unique_college as a string
print("Not all college names in the dataset are unique. There are " +␣
 ↪str(len(unique_college)) + " unique colleges where the NBA players played␣
 ↪before coming to the NBA in the dataset.\n")
print("Here are the first 100 colleges from the unique_college dict. (List was␣
 ↪too long to print completely): ")
```

```
print(list(unique_college)[:100])
```

```
Not all college names in the dataset are unique. There are 336 unique colleges
where the NBA players played before coming to the NBA in the dataset.

Here are the first 100 colleges from the unique_college dict. (List was too long
to print completely):
['Gonzaga', 'Boston College', 'Delaware', 'Northern Arizona', 'Texas Christian',
'Nebraska-Lincoln', 'Montevallo', "St. Mary's (TX)", 'Virginia Union',
'Wisconsin-Stevens Point', 'South Carolina', 'Midland', 'Grand Canyon', 'North
Carolina State', 'Truman State', 'Hampton', 'McNeese State', 'St. Bonaventure',
'La Salle', 'Tulsa', 'Cal State-Long Beach', 'Toledo', 'Oregon State',
'Clemson', 'Western Kentucky', 'Richmond', 'Indiana', 'Arizona', 'Texas-San
Antonio', 'Thomas More', 'Old Dominion', 'Blinn', ' ', 'Pennsylvania',
'Westchester CC NY', 'Utah State', 'Delta State', 'Murray State', 'Duke',
'Yale', 'Bucknell', 'Alabama', 'Montana', 'Manhattan', "St. Mary's (CA)",
'Rutgers', 'Coppin State', 'Cincinnati', 'Connecticut', 'Marquette', 'Nicholls
State', 'Lehigh', 'Penn State', 'George Mason', 'Northeast Mississippi Community
College', 'Pepperdine', 'Oakland', 'Loyola (IL)', 'Hofstra', 'DePaul', "St.
John's (NY)", 'Trinity Valley Community College', 'Augsburg', 'New Mexico', 'St.
Louis', 'Ohio', 'Centenary (LA)', 'Houston', 'Nevada-Reno', 'New Orleans',
'Wisconsin-Green Bay', 'Wyoming', 'Nebraska', 'Albany State (GA)', 'Colorado
State', 'Mississippi', 'California-Berkeley', 'Eastern Michigan', 'Michigan
State', "Master's", 'Saint Rose', 'Northwestern State', 'Georgia Institute of
Technology', 'Pacific', 'Boston U.', 'Eastern Washington', 'Montana State',
'Brigham Young', 'Seward County Community College', 'William & Mary',
'Washington State', 'Western Michigan', 'Tennessee-Martin', 'Delaware State',
'Minnesota', 'Rice', 'Xavier', 'Alabama-Birmingham', 'Syracuse', 'Cal Poly']
```

[8]:
```
# creating a set with all the countries from data to only include each unique␣
 ↪country with no repeats
unique_country = {player['country'] for player in data}
# then displaying the length of unique_country as a string
print("Not all country names in the dataset are unique. There are " +␣
 ↪str(len(unique_country)) + " unique countries that the NBA players␣
 ↪originated from in the dataset.\n")
print(unique_country)
```

```
Not all country names in the dataset are unique. There are 79 unique countries
that the NBA players originated from in the dataset.

{'Tanzania', 'Germany', 'Iran', 'Poland', 'Great Britain', 'Argentina',
'Senegal', 'Australia', 'Turkey', 'United Kingdom', 'Switzerland', 'Montenegro',
'Georgia', 'St. Vincent & Grenadines', 'Haiti', 'Bosnia & Herzegovina', 'Gabon',
'Nigeria', 'Bosnia', 'Italy', 'China', 'Croatia', 'Sudan (UK)', 'Bahamas',
'Ghana', 'Yugoslavia', 'England', 'US Virgin Islands', 'Scotland', 'Guinea',
'Belize', 'Tunisia', 'Ireland', 'Uruguay', 'Mexico', 'Egypt', 'Slovenia',
'Brazil', 'Cabo Verde', 'Angola', 'South Sudan', 'Netherlands', 'Trinidad and
```

```
Tobago', 'USA', 'USSR', 'Japan', 'Greece', 'France', 'Austria', 'Dominican
Republic', 'Ukraine', 'Cameroon', 'Jamaica', 'Venezuela', 'Panama', 'U.S. Virgin
Islands', 'Lithuania', 'Latvia', 'New Zealand', 'Serbia and Montenegro', 'Saint
Lucia', 'South Korea', 'Sweden', 'Sudan', 'Macedonia', 'Bosnia and Herzegovina',
'Finland', 'Israel', 'Democratic Republic of the Congo', 'Czech Republic',
'Congo', 'Canada', 'DRC', 'Republic of the Congo', 'Puerto Rico', 'Mali',
'Serbia', 'Spain', 'Russia'}
```

[10]:
```
'''
Running the dataRangeFunc() by calling the 'reduce' function to display the␣
 ↪range of years.
'''

# created a reducer function to loop through all the numerical years of the data
def dataRangeFunc(state, player):
    # compute a list of all the years
    if state is None:
        state = {
        'min_year': player['season'].split('-')[0],
        'max_year': player['season'].split('-')[0]
        }
    # continually update the 'min_year' & 'max_year'
    else:
        state['min_year'] = min(state['min_year'], player['season'].
 ↪split('-')[0])
        state['max_year'] = max(state['max_year'], player['season'].
 ↪split('-')[0])

    return state

# run the reducer function and output 'min_year' and 'max_year' to display our␣
 ↪range
print("Here is the dataset's range of years, in other words, the season range␣
 ↪(oldest and newest): ")
reduce(dataRangeFunc, data, None)['min_year'] + " to " + reduce(dataRangeFunc,␣
 ↪data, None)['max_year']
```

```
Here is the dataset's range of years, in other words, the season range (oldest
and newest):
```

[10]:
```
'1996 to 2020'
```

2.4 Proposal & Motivation:

2.4.1 Why we chose this dataset?

As big fans of the NBA and the sport of basketball, we found this dataset very interesting. There is a lot of interesting information regarding all the NBA players over the years and we are excited to study research questions pertaining to all players in the league. Unlike other datasets, this one

is actually a topic of interest for us and gives us the opportunity to both practice data analysis and learn more about the sport we love!

2.4.2 Another dataset we considered:

Yes! We were originally going to use another NBA dataset (https://www.kaggle.com/nathanlauga/nba-games?select=games.csv). It also had some very insightful data, however, we felt that it was far more focused on the NBA team's performances, rather than the players. As a group, we were more interested to research the data of the players to visualize talent existent within the league since 1996.

2.5 Data Analysis Questions:

2.5.1 Here are a few tentative questions that we would like to study for this dataset:

1) Calculate the percentage of players that were undrafted.
    - To see how many and what players were talented enough to directly sign with an NBA team instead of being drafted.
2) What is the average age of the players by season?
3) Find the top 5 colleges.
    - To see which colleges most players played for before coming to the NBA.
4) Find the top 5 countries NBA players are from.
    - To see what percentage of players from each unique country
5) All-Star player from each unique country? (All-Star player defined as pts: 25+, reb: 7+, ast: 7+).
6) Compare first-round, number-one player picks from each draft year.
    - To find out whether there were any NBA busts from the draft.
7) Determine the position of each player (since it's not provided in the dataset).
    - To find the average height per position.
8) Does the player's weight affect their performance?
9) Who is the longest active player in the league?
10) Find the player with the most points, rebounds, and assists, respectively, of all-time.

2.5.2 Methods to Apply to Answer the above:

To begin, we will utilize Python's in-built libraries extensively and use multiple functions such as, 'map', 'filter', and 'reduce'. We plan to primarily Numpy for the programming of most of the data as it is mostly comprised of numerical values. However, we expect to learn new things in the near future of this course, including Pandas and Mathplotlib. Both Pandas and Mathplotlib will help greatly to visualize and process the data in different aspects using graphs and charts. Furthermore, we will analyze the data using mathematical, statistical analysis, and attributes to complete conclusions to our questions of research.

2.5.3 Potential for Data Science Applications:

There is great potential to apply our knowledge of data science and translate it to the data of NBA players. We can predict players' performance and development over their NBA career. In a real-life scenario, we can recommend colleges to those interested to join the NBA for their future. We can also simulate the players' projected growth (whether they will become an all-star) and work with the coaching staff to provide them with such information, so they can make the essential

adjustments to players' training and development. Most importantly, we would be able to visualize the diversity of players within the NBA.