labassignment3bn

February 4, 2025

1 Lab Assignment 3: How to Load, Convert, and Write JSON Files in Python

1.1 DS 6001: Practice and Application of Data Science

1.1.1 Brian Nolton

P.S. How do I turn off these formatting numbers. I Googled it but still couldn't find out how…

1.1.2 Instructions

Please answer the following questions as completely as possible using text, code, and the results of code as needed. Format your answers in a Jupyter notebook. To receive full credit, make sure you address every part of the problem, and make sure your document is formatted in a clean and professional way.

1.2 Problem 0

Import the following libraries:

[1]:
```
import numpy as np
import pandas as pd
import requests
import json
import sys
sys.tracebacklimit = 0 # turn off the error tracebacks
```

1.3 Problem 1

JSON and CSV are both text-based formats for the storage of data. It's possible to open either one in a plain text editor. Given this similarity, why does a CSV file usually take less memory than a JSON formatted file for the same data? Under what conditions could a JSON file be smaller in memory than a CSV file for the same data? (2 points)

A JSON file contains repeated key names (for each entry) and structural brackets adding to the text characters needed to support a JSON file. A CSV has the key names once as column headers and only needs one separator between values. For these reasons a CSV file takes less memory than a JSON file. A JSON file can be smaller when there is a lot of nesting in the data. JSON supports nesting and stores hierachical data very efficiently. Since a CSV file is a simpler file, it does not

support this type of data and often has to duplicate data to achieve a semblance of this structure. Becuase of this, JSONs may be smaller than CSVs for certain data sets.

1.4 Problem 2

NASA has a dataset of all meteorites that have fallen to Earth between the years A.D. 860 and 2013. The data contain the name of each meteorite, along with the coordinates of the place where the meteorite hit, the mass of the meteorite, and the date of the collison. The data is stored as a JSON here: https://data.nasa.gov/resource/y77d-th95.json

Look at the data in your web-browser and explain which strategy for loading the JSON into Python makes the most sense and why.

Then write and run the code that will work for loading the data into Python. (2 points)

The data looks like a json file with nested data and no metadata. I think becuase of this, is is a standard case for requests.get().

[2]:
```
url2 = "https://data.nasa.gov/resource/y77d-th95.json"
r2 = requests.get(url2)
r2
```

[2]:
```
<Response [200]>
```

1.5 Problem 3

The textbook chapter for this module shows, as an example, how to pull data in JSON format from Reddit's top 25 posts on /r/popular. The steps outlined there pull all of the features in the data into the dataframe, resulting in a dataframe with 172 columns.

If we only wanted a few features, then looping across elements of the JSON list itself and extracting only the data we want may be a more efficient approach.

Use looping - and not `pd.read_json()` or `pd.json_normalize()` - to create a dataframe with 25 rows (one for each of the top 25 posts), and only columns for `subreddit`, `title`, `ups`, and `created_utc`. The JSON file exists at http://www.reddit.com/r/popular/top.json, and don't forget to specify `headers = {'User-agent': 'DS6001'}` within `requests.get()`. (3 points)

[3]:
```
url3 = "https://www.reddit.com/r/popular/top.json"
r3 = requests.get(url3, headers = {'User-agent': 'DS6001'})
r3_json = json.loads(r3.text)
r3_json['data']['children'][0]
```

[3]:
```
{'kind': 't3',
 'data': {'approved_at_utc': None,
  'subreddit': 'funny',
  'selftext': '',
  'author_fullname': 't2_g2fh9fd',
  'saved': False,
  'mod_reason_title': None,
  'gilded': 0,
```

```
  'clicked': False,
  'title': 'I have seen it all',
  'link_flair_richtext': [],
  'subreddit_name_prefixed': 'r/funny',
  'hidden': False,
  'pwls': 6,
  'link_flair_css_class': None,
  'downs': 0,
  'thumbnail_height': 140,
  'top_awarded_type': None,
  'hide_score': False,
  'name': 't3_1ih8zg6',
  'quarantine': False,
  'link_flair_text_color': 'dark',
  'upvote_ratio': 0.9,
  'author_flair_background_color': None,
  'subreddit_type': 'public',
  'ups': 88388,
  'total_awards_received': 0,
  'media_embed': {},
  'thumbnail_width': 140,
  'author_flair_template_id': None,
  'is_original_content': False,
  'user_reports': [],
  'secure_media': {'reddit_video': {'bitrate_kbps': 2400,
    'fallback_url':
'https://v.redd.it/jlm05e5qp1he1/DASH_720.mp4?source=fallback',
    'has_audio': True,
    'height': 1280,
    'width': 720,
    'scrubber_media_url': 'https://v.redd.it/jlm05e5qp1he1/DASH_96.mp4',
    'dash_url': 'https://v.redd.it/jlm05e5qp1he1/DASHPlaylist.mpd?a=1741313070%2
CZGRjOTQzOThmNTFlOGEzZjIyZWNjZjM2ZTJlMjAzMzcxNzIzOTY3MDIzNDdiZDJlMGQ2MWQ1MWQ0N2R
kM2JjOQ%3D%3D&amp;amp;v=1&amp;amp;f=sd',
    'duration': 14,
    'hls_url': 'https://v.redd.it/jlm05e5qp1he1/HLSPlaylist.m3u8?a=1741313070%2C
N2VmNmFhNzk5NWEyYmI5YjQ2YTE1Njk5OTI2YmRhYTQ0MmU0YzIzMTM5MjUyYTdhMmM3NWJhYjBmNTc4
OGNmNA%3D%3D&amp;amp;v=1&amp;amp;f=sd',
    'is_gif': False,
    'transcoding_status': 'completed'}},
  'is_reddit_media_domain': True,
  'is_meta': False,
  'category': None,
  'secure_media_embed': {},
  'link_flair_text': None,
  'can_mod_post': False,
  'score': 88388,
```

```
  'approved_by': None,
  'is_created_from_ads_ui': False,
  'author_premium': False,
  'thumbnail': 'https://external-preview.redd.it/YW42eXFkenBwMWhlMUvJNqn30mzCcxR
GNbdxfC8kq4yJ6fkRg5RUmD_ZMOhh.png?width=140&amp;height=140&amp;crop=140:140,smar
t&amp;format=jpg&amp;v=enabled&amp;lthumb=true&amp;s=7a390807099e0c61a31cf82401a
8764947de725b',
  'edited': False,
  'author_flair_css_class': None,
  'author_flair_richtext': [],
  'gildings': {},
  'post_hint': 'hosted:video',
  'content_categories': None,
  'is_self': False,
  'mod_note': None,
  'created': 1738641467.0,
  'link_flair_type': 'text',
  'wls': 6,
  'removed_by_category': None,
  'banned_by': None,
  'author_flair_type': 'text',
  'domain': 'v.redd.it',
  'allow_live_comments': False,
  'selftext_html': None,
  'likes': None,
  'suggested_sort': None,
  'banned_at_utc': None,
  'url_overridden_by_dest': 'https://v.redd.it/jlm05e5qp1he1',
  'view_count': None,
  'archived': False,
  'no_follow': False,
  'is_crosspostable': False,
  'pinned': False,
  'over_18': False,
  'preview': {'images': [{'source': {'url': 'https://external-preview.redd.it/YW
42eXFkenBwMWhlMUvJNqn30mzCcxRGNbdxfC8kq4yJ6fkRg5RUmD_ZMOhh.png?format=pjpg&amp;a
uto=webp&amp;s=80bf0c4d95456ec4899c2eafb72b499752f5d091',
      'width': 720,
      'height': 1280},
     'resolutions': [{'url': 'https://external-preview.redd.it/YW42eXFkenBwMWhlM
UvJNqn30mzCcxRGNbdxfC8kq4yJ6fkRg5RUmD_ZMOhh.png?width=108&amp;crop=smart&amp;for
mat=pjpg&amp;auto=webp&amp;s=08b5caf4ed76e64c250e62d8a5fdd4267f736132',
       'width': 108,
       'height': 192},
      {'url': 'https://external-preview.redd.it/YW42eXFkenBwMWhlMUvJNqn30mzCcxRG
NbdxfC8kq4yJ6fkRg5RUmD_ZMOhh.png?width=216&amp;crop=smart&amp;format=pjpg&amp;au
to=webp&amp;s=ea995ea7142f706a687f078e70c690f44a51d677',
```

```
       'width': 216,
       'height': 384},
      {'url': 'https://external-preview.redd.it/YW42eXFkenBwMWhlMUvJNqn30mzCcxRG
NbdxfC8kq4yJ6fkRg5RUmD_ZMOhh.png?width=320&amp;crop=smart&amp;format=pjpg&amp;au
to=webp&amp;s=c5c8ca5b9742d75cc4c11b872d062471b19c062a',
       'width': 320,
       'height': 568},
      {'url': 'https://external-preview.redd.it/YW42eXFkenBwMWhlMUvJNqn30mzCcxRG
NbdxfC8kq4yJ6fkRg5RUmD_ZMOhh.png?width=640&amp;crop=smart&amp;format=pjpg&amp;au
to=webp&amp;s=57d48f19e6c19b8a782371e0f4878e54ab34066f',
       'width': 640,
       'height': 1137}],
     'variants': {},
     'id': 'YW42eXFkenBwMWhlMUvJNqn30mzCcxRGNbdxfC8kq4yJ6fkRg5RUmD_ZMOhh'}],
   'enabled': False},
  'all_awardings': [],
  'awarders': [],
  'media_only': False,
  'can_gild': False,
  'spoiler': False,
  'locked': False,
  'author_flair_text': None,
  'treatment_tags': [],
  'visited': False,
  'removed_by': None,
  'num_reports': None,
  'distinguished': None,
  'subreddit_id': 't5_2qh33',
  'author_is_blocked': False,
  'mod_reason_by': None,
  'removal_reason': None,
  'link_flair_background_color': '',
  'id': '1ih8zg6',
  'is_robot_indexable': True,
  'report_reasons': None,
  'author': 'RoyalChris',
  'discussion_type': None,
  'num_comments': 815,
  'send_replies': True,
  'contest_mode': False,
  'mod_reports': [],
  'author_patreon_flair': False,
  'author_flair_text_color': None,
  'permalink': '/r/funny/comments/1ih8zg6/i_have_seen_it_all/',
  'stickied': False,
  'url': 'https://v.redd.it/jlm05e5qp1he1',
  'subreddit_subscribers': 66251638,
```

```
  'created_utc': 1738641467.0,
  'num_crossposts': 50,
  'media': {'reddit_video': {'bitrate_kbps': 2400,
    'fallback_url':
'https://v.redd.it/jlm05e5qp1he1/DASH_720.mp4?source=fallback',
    'has_audio': True,
    'height': 1280,
    'width': 720,
    'scrubber_media_url': 'https://v.redd.it/jlm05e5qp1he1/DASH_96.mp4',
    'dash_url': 'https://v.redd.it/jlm05e5qp1he1/DASHPlaylist.mpd?a=1741313070%2
CZGRjOTQzOThmNTFlOGEzZjIyZWNjZjM2ZTJlMjAzMzcxNzIzOTY3MDIzNDdiZDJlMGQ2MWQ1MWQ0N2R
kM2JjOQ%3D%3D&amp;v=1&amp;f=sd',
    'duration': 14,
    'hls_url': 'https://v.redd.it/jlm05e5qp1he1/HLSPlaylist.m3u8?a=1741313070%2C
N2VmNmFhNzk5NWEyYmI5YjQ2YTE1Njk5OTI2YmRhYTQ0MmU0YzIzMTM5MjUyYTdhMmM3NWJhYjBmNTc4
OGNmNA%3D%3D&amp;v=1&amp;f=sd',
    'is_gif': False,
    'transcoding_status': 'completed'}},
  'is_video': True}}
```

[4]:
```
r3_df = pd.DataFrame(
    [u['data']['subreddit'], u['data']['title'], u['data']['ups'],␣
 ↪u['data']['created_utc']]
    for u in r3_json['data']['children'][:25]
)
r3_df.columns = ['subreddit', 'title', 'ups', 'created_utc']
r3_df
```

[4]:

	subreddit	title	\
0	funny	I have seen it all	
1	meirl	Meirl	
2	MurderedByWords	"Who do you think you're talking to"	
3	MadeMeSmile	Simple perfect	
4	gifs	The Great Fall	
5	comics	James	
6	MadeMeSmile	Arnold is not a self-made man	
7	BeAmazed	Derrick Byrd, 20, sustained second- and third-…	
8	SipsTea	Indeed it was	
9	texas	I'll leave this right here Texas	
10	clevercomebacks	School choice	
11	clevercomebacks	Owned i guess	
12	news	FBI agents file class action lawsuit against T…	
13	technology	A Coup Is In Progress In America	
14	pics	In the beginning was Peter Thiel & Elon Mu…	
15	pics	Protest against far-right party in Germany (Be…	
16	Fauxmoi	Tim Waltz's post about Musk	
17	aww	Happy 102nd bday to my grandma!	

```
18                  pics  Benjamin Netanyahu meeting with the President …
19                  news  2.2 billion gallons of water flowed out of Cal…
20       MurderedByWords                                  Tammy got schooled
21  Damnthatsinteresting           Tigers appear green to certain animals!
22                  cats  One of these cats is not very happy to have a …
23                  meme                           My worst fear realised…
24                  pics          Purchased the most beautiful onion today

      ups   created_utc
0   88388  1.738641e+09
1   87684  1.738681e+09
2   83306  1.738670e+09
3   82689  1.738694e+09
4   81558  1.738637e+09
5   81354  1.738682e+09
6   72132  1.738669e+09
7   67278  1.738697e+09
8   59168  1.738668e+09
9   55751  1.738681e+09
10  55014  1.738675e+09
11  52886  1.738653e+09
12  52757  1.738694e+09
13  52525  1.738639e+09
14  52294  1.738682e+09
15  51558  1.738668e+09
16  51501  1.738673e+09
17  50915  1.738679e+09
18  50933  1.738702e+09
19  50260  1.738682e+09
20  49776  1.738687e+09
21  48273  1.738703e+09
22  47676  1.738638e+09
23  46942  1.738648e+09
24  46343  1.738665e+09
```

1.6 Problem 4

The NBA has saved data on all 30 teams' shooting statistics for the 2014-2015 season here: https://stats.nba.com/js/data/sportvu/2015/shootingTeamData.json. Take a moment and look at this JSON file in your web browser. The structure of this particular JSON is complicated, but see if you can find the team-by-team data. In this problem our goal is to use `pd.json_normalize()` to get the data into a dataframe. The following questions will guide you towards this goal.

1.6.1 Part a

Download the raw text of the NBA JSON file and register it as JSON formatted data in Python's memory. (2 points)

1.6.2 Part b

Describe, in words, the path that leads to the team-by-team data. (2 points)

1.6.3 Part c

Use the `pd.json_normalize()` function to pull the team-by-team data into a dataframe. This is going to be tricky. You will need to use indexing on the JSON data as well as the `record_path` parameter.

If you are successful, you will have a dataframe with 30 rows and 33 columns. The first row will refer to the Golden State Warriors, the second row will refer to the San Antonio Spurs, and the third row will refer to the Cleveland Cavaliers. The columns will only be named 0, 1, 2, … at this point. (4 points)

1.6.4 Part d

Find the path that leads to the headers (the column names), and extract these names as a list. Then set the `.columns` attribute of the dataframe you created in part c equal to this list. The result should be that the dataframe now has the correct column names. (3 points)

[5]:
```
#Problem 4 Part A
url4 = "https://stats.nba.com/js/data/sportvu/2015/shootingTeamData.json"
r4 = requests.get(url4)
r4_json = json.loads(r4.text)
```

Problem 4 Part B To get to the team by team data you must access the "resultSets" branch, then go through "Index 0" (though that is the only index), then access the "rowSet" branch. All the team by team data is there.

[6]:
```
#Problem 4 Part C
r4_df = pd.json_normalize(r4_json, record_path=['resultSets', "rowSet"])
r4_df
```

[6]:

	0	1	2	3	4	5	6	7	8	\
0	1610612744	Golden State	Warriors	GSW		82	48.7	114.9	14.9	
1	1610612759	San Antonio	Spurs	SAS		82	48.3	103.5	14.8	
2	1610612739	Cleveland	Cavaliers	CLE		82	48.7	104.3	16.9	
3	1610612746	Los Angeles	Clippers	LAC		82	48.6	104.5	15.0	
4	1610612760	Oklahoma City	Thunder	OKC		82	48.6	110.2	16.1	
5	1610612737	Atlanta	Hawks	ATL		82	48.6	102.8	19.0	
6	1610612745	Houston	Rockets	HOU		82	48.6	106.5	17.2	
7	1610612757	Portland	Trail Blazers	POR		82	48.5	105.1	17.5	
8	1610612758	Sacramento	Kings	SAC		81	48.4	106.7	18.7	
9	1610612764	Washington	Wizards	WAS		82	48.5	104.1	15.4	
10	1610612748	Miami	Heat	MIA		82	48.6	100.0	17.9	
11	1610612761	Toronto	Raptors	TOR		81	48.5	102.7	23.0	
12	1610612742	Dallas	Mavericks	DAL		82	49.0	102.3	18.2	
13	1610612766	Charlotte	Hornets	CHA		82	48.6	103.4	16.8	

14	1610612762	Utah	Jazz	UTA	82	49.0	97.7	18.1
15	1610612753	Orlando	Magic	ORL	81	48.7	102.0	18.0
16	1610612749	Milwaukee	Bucks	MIL	82	48.7	99.0	17.4
17	1610612740	New Orleans	Pelicans	NOP	82	48.5	102.7	19.9
18	1610612750	Minnesota	Timberwolves	MIN	82	48.6	102.4	15.1
19	1610612754	Indiana	Pacers	IND	82	48.8	102.2	13.7
20	1610612751	Brooklyn	Nets	BKN	82	48.4	98.6	14.4
21	1610612765	Detroit	Pistons	DET	82	48.7	102.0	17.5
22	1610612743	Denver	Nuggets	DEN	82	48.6	101.9	15.9
23	1610612738	Boston	Celtics	BOS	81	48.5	105.6	18.9
24	1610612741	Chicago	Bulls	CHI	82	48.9	101.6	18.1
25	1610612755	Philadelphia	76ers	PHI	82	48.6	97.4	19.7
26	1610612756	Phoenix	Suns	PHX	82	48.4	100.9	15.6
27	1610612752	New York	Knicks	NYK	82	48.5	98.4	10.4
28	1610612763	Memphis	Grizzlies	MEM	82	48.6	99.1	16.4
29	1610612747	Los Angeles	Lakers	LAL	82	48.3	97.3	15.6

	9	…	23	24	25	26	27	28	29	30	31	32
0	0.498	…	0.478	21.2	42.5	0.497	2.3	6.3	0.363	10.8	25.3	0.429
1	0.481	…	0.506	18.3	39.8	0.460	0.9	2.6	0.341	6.1	15.9	0.381
2	0.481	…	0.473	18.2	40.7	0.447	1.7	5.7	0.299	9.0	23.9	0.378
3	0.497	…	0.480	18.9	42.0	0.450	2.0	6.0	0.334	7.7	20.8	0.373
4	0.480	…	0.497	17.5	38.7	0.451	1.6	5.1	0.321	6.6	18.6	0.356
5	0.463	…	0.483	19.4	44.6	0.435	1.0	3.1	0.311	9.0	25.3	0.355
6	0.433	…	0.472	15.5	36.4	0.426	2.3	7.4	0.318	8.4	23.5	0.355
7	0.441	…	0.447	18.0	39.8	0.453	1.7	5.9	0.295	8.8	22.6	0.389
8	0.452	…	0.473	18.1	39.7	0.454	0.9	3.1	0.276	7.2	19.4	0.372
9	0.480	…	0.483	19.5	44.3	0.439	0.7	2.7	0.254	8.0	21.5	0.371
10	0.488	…	0.490	15.7	35.2	0.445	0.8	2.9	0.282	5.3	15.1	0.347
11	0.462	…	0.461	14.1	32.4	0.436	1.8	5.6	0.327	6.8	17.7	0.384
12	0.473	…	0.464	17.5	41.4	0.423	1.4	5.3	0.273	8.4	23.3	0.360
13	0.459	…	0.449	17.0	39.8	0.427	1.8	6.0	0.297	8.9	23.4	0.379
14	0.445	…	0.468	15.9	37.2	0.426	1.4	4.3	0.318	7.1	19.5	0.363
15	0.456	…	0.475	18.5	42.6	0.435	0.7	2.7	0.249	7.1	19.5	0.363
16	0.463	…	0.477	13.2	29.4	0.448	1.1	4.0	0.270	4.3	11.6	0.370
17	0.458	…	0.460	17.9	41.1	0.434	0.6	2.6	0.247	7.9	21.2	0.374
18	0.464	…	0.471	16.1	35.4	0.455	0.7	2.6	0.272	4.8	13.8	0.350
19	0.453	…	0.465	16.4	38.1	0.431	1.7	5.7	0.299	6.4	17.4	0.368
20	0.457	…	0.464	15.8	36.1	0.438	1.0	3.3	0.303	5.5	15.1	0.363
21	0.464	…	0.452	15.7	37.2	0.422	0.9	4.0	0.227	8.1	22.2	0.366
22	0.406	…	0.448	16.4	37.8	0.434	1.1	4.3	0.264	6.9	19.5	0.354
23	0.453	…	0.451	16.9	39.9	0.424	1.6	5.7	0.274	7.1	20.3	0.350
24	0.458	…	0.442	17.0	38.5	0.441	1.3	3.9	0.332	6.6	17.5	0.380
25	0.445	…	0.449	15.3	37.4	0.409	1.6	5.7	0.281	7.7	21.8	0.354
26	0.440	…	0.447	16.6	39.5	0.421	1.4	5.0	0.288	7.6	20.8	0.363
27	0.447	…	0.439	15.9	36.4	0.438	1.5	4.9	0.305	5.9	16.6	0.358
28	0.440	…	0.459	16.1	38.5	0.418	0.7	2.5	0.278	5.4	16.0	0.340

```
29  0.441  …  0.420  14.0  34.5  0.406  2.2  7.9  0.278   5.6  16.7  0.335

[30 rows x 33 columns]
```

[7]:
```
#Problem 4 Part D
r4_df.columns = list(pd.json_normalize(r4_json, record_path=['resultSets',␣
 ↪"headers"])[0])
r4_df
```

[7]:

	TEAM_ID	TEAM_CITY	TEAM_NAME	TEAM_ABBREVIATION	TEAM_CODE	GP	\
0	1610612744	Golden State	Warriors	GSW		82	
1	1610612759	San Antonio	Spurs	SAS		82	
2	1610612739	Cleveland	Cavaliers	CLE		82	
3	1610612746	Los Angeles	Clippers	LAC		82	
4	1610612760	Oklahoma City	Thunder	OKC		82	
5	1610612737	Atlanta	Hawks	ATL		82	
6	1610612745	Houston	Rockets	HOU		82	
7	1610612757	Portland	Trail Blazers	POR		82	
8	1610612758	Sacramento	Kings	SAC		81	
9	1610612764	Washington	Wizards	WAS		82	
10	1610612748	Miami	Heat	MIA		82	
11	1610612761	Toronto	Raptors	TOR		81	
12	1610612742	Dallas	Mavericks	DAL		82	
13	1610612766	Charlotte	Hornets	CHA		82	
14	1610612762	Utah	Jazz	UTA		82	
15	1610612753	Orlando	Magic	ORL		81	
16	1610612749	Milwaukee	Bucks	MIL		82	
17	1610612740	New Orleans	Pelicans	NOP		82	
18	1610612750	Minnesota	Timberwolves	MIN		82	
19	1610612754	Indiana	Pacers	IND		82	
20	1610612751	Brooklyn	Nets	BKN		82	
21	1610612765	Detroit	Pistons	DET		82	
22	1610612743	Denver	Nuggets	DEN		82	
23	1610612738	Boston	Celtics	BOS		81	
24	1610612741	Chicago	Bulls	CHI		82	
25	1610612755	Philadelphia	76ers	PHI		82	
26	1610612756	Phoenix	Suns	PHX		82	
27	1610612752	New York	Knicks	NYK		82	
28	1610612763	Memphis	Grizzlies	MEM		82	
29	1610612747	Los Angeles	Lakers	LAL		82	

	MIN	PTS	PTS_DRIVE	FGP_DRIVE	…	CFGP	UFGM	UFGA	UFGP	CFG3M	\
0	48.7	114.9	14.9	0.498	…	0.478	21.2	42.5	0.497	2.3	
1	48.3	103.5	14.8	0.481	…	0.506	18.3	39.8	0.460	0.9	
2	48.7	104.3	16.9	0.481	…	0.473	18.2	40.7	0.447	1.7	
3	48.6	104.5	15.0	0.497	…	0.480	18.9	42.0	0.450	2.0	
4	48.6	110.2	16.1	0.480	…	0.497	17.5	38.7	0.451	1.6	

5	48.6	102.8	19.0	0.463	…	0.483	19.4	44.6	0.435	1.0
6	48.6	106.5	17.2	0.433	…	0.472	15.5	36.4	0.426	2.3
7	48.5	105.1	17.5	0.441	…	0.447	18.0	39.8	0.453	1.7
8	48.4	106.7	18.7	0.452	…	0.473	18.1	39.7	0.454	0.9
9	48.5	104.1	15.4	0.480	…	0.483	19.5	44.3	0.439	0.7
10	48.6	100.0	17.9	0.488	…	0.490	15.7	35.2	0.445	0.8
11	48.5	102.7	23.0	0.462	…	0.461	14.1	32.4	0.436	1.8
12	49.0	102.3	18.2	0.473	…	0.464	17.5	41.4	0.423	1.4
13	48.6	103.4	16.8	0.459	…	0.449	17.0	39.8	0.427	1.8
14	49.0	97.7	18.1	0.445	…	0.468	15.9	37.2	0.426	1.4
15	48.7	102.0	18.0	0.456	…	0.475	18.5	42.6	0.435	0.7
16	48.7	99.0	17.4	0.463	…	0.477	13.2	29.4	0.448	1.1
17	48.5	102.7	19.9	0.458	…	0.460	17.9	41.1	0.434	0.6
18	48.6	102.4	15.1	0.464	…	0.471	16.1	35.4	0.455	0.7
19	48.8	102.2	13.7	0.453	…	0.465	16.4	38.1	0.431	1.7
20	48.4	98.6	14.4	0.457	…	0.464	15.8	36.1	0.438	1.0
21	48.7	102.0	17.5	0.464	…	0.452	15.7	37.2	0.422	0.9
22	48.6	101.9	15.9	0.406	…	0.448	16.4	37.8	0.434	1.1
23	48.5	105.6	18.9	0.453	…	0.451	16.9	39.9	0.424	1.6
24	48.9	101.6	18.1	0.458	…	0.442	17.0	38.5	0.441	1.3
25	48.6	97.4	19.7	0.445	…	0.449	15.3	37.4	0.409	1.6
26	48.4	100.9	15.6	0.440	…	0.447	16.6	39.5	0.421	1.4
27	48.5	98.4	10.4	0.447	…	0.439	15.9	36.4	0.438	1.5
28	48.6	99.1	16.4	0.440	…	0.459	16.1	38.5	0.418	0.7
29	48.3	97.3	15.6	0.441	…	0.420	14.0	34.5	0.406	2.2

	CFG3A	CFG3P	UFG3M	UFG3A	UFG3P
0	6.3	0.363	10.8	25.3	0.429
1	2.6	0.341	6.1	15.9	0.381
2	5.7	0.299	9.0	23.9	0.378
3	6.0	0.334	7.7	20.8	0.373
4	5.1	0.321	6.6	18.6	0.356
5	3.1	0.311	9.0	25.3	0.355
6	7.4	0.318	8.4	23.5	0.355
7	5.9	0.295	8.8	22.6	0.389
8	3.1	0.276	7.2	19.4	0.372
9	2.7	0.254	8.0	21.5	0.371
10	2.9	0.282	5.3	15.1	0.347
11	5.6	0.327	6.8	17.7	0.384
12	5.3	0.273	8.4	23.3	0.360
13	6.0	0.297	8.9	23.4	0.379
14	4.3	0.318	7.1	19.5	0.363
15	2.7	0.249	7.1	19.5	0.363
16	4.0	0.270	4.3	11.6	0.370
17	2.6	0.247	7.9	21.2	0.374
18	2.6	0.272	4.8	13.8	0.350
19	5.7	0.299	6.4	17.4	0.368

```
20    3.3  0.303    5.5   15.1  0.363
21    4.0  0.227    8.1   22.2  0.366
22    4.3  0.264    6.9   19.5  0.354
23    5.7  0.274    7.1   20.3  0.350
24    3.9  0.332    6.6   17.5  0.380
25    5.7  0.281    7.7   21.8  0.354
26    5.0  0.288    7.6   20.8  0.363
27    4.9  0.305    5.9   16.6  0.358
28    2.5  0.278    5.4   16.0  0.340
29    7.9  0.278    5.6   16.7  0.335

[30 rows x 33 columns]
```

1.7 Problem 5

Save the NBA dataframe you extracted in problem 4 as a JSON-formatted text file on your local machine. Format the JSON so that it is organized as dictionary with three lists: `columns` lists the column names, `index` lists the row names, and `data` is a list-of-lists of data points, one list for each row. (Hint: this is possible with one line of code) (2 points)

[8]:
```
nba_json = r4_df.to_json(orient='split')
json.loads(nba_json)
```

[8]:
```
{'columns': ['TEAM_ID',
  'TEAM_CITY',
  'TEAM_NAME',
  'TEAM_ABBREVIATION',
  'TEAM_CODE',
  'GP',
  'MIN',
  'PTS',
  'PTS_DRIVE',
  'FGP_DRIVE',
  'PTS_CLOSE',
  'FGP_CLOSE',
  'PTS_CATCH_SHOOT',
  'FGP_CATCH_SHOOT',
  'PTS_PULL_UP',
  'FGP_PULL_UP',
  'FGA_DRIVE',
  'FGA_CLOSE',
  'FGA_CATCH_SHOOT',
  'FGA_PULL_UP',
  'EFG_PCT',
  'CFGM',
  'CFGA',
  'CFGP',
```

```
 'UFGM',
 'UFGA',
 'UFGP',
 'CFG3M',
 'CFG3A',
 'CFG3P',
 'UFG3M',
 'UFG3A',
 'UFG3P'],
'index': [0,
 1,
 2,
 3,
 4,
 5,
 6,
 7,
 8,
 9,
 10,
 11,
 12,
 13,
 14,
 15,
 16,
 17,
 18,
 19,
 20,
 21,
 22,
 23,
 24,
 25,
 26,
 27,
 28,
 29],
'data': [['1610612744',
  'Golden State',
  'Warriors',
  'GSW',
  '',
  82,
  48.7,
  114.9,
```

```
 14.9,
 0.498,
 16.7,
 0.645,
 33.7,
 0.428,
 21.5,
 0.418,
 11.0,
 11.1,
 28.3,
 21.5,
 0.563,
 21.4,
 44.8,
 0.478,
 21.2,
 42.5,
 0.497,
 2.3,
 6.3,
 0.363,
 10.8,
 25.3,
 0.429],
['1610612759',
 'San Antonio',
 'Spurs',
 'SAS',
 '',
 82,
 48.3,
 103.5,
 14.8,
 0.481,
 17.8,
 0.611,
 27.1,
 0.419,
 18.0,
 0.398,
 11.8,
 12.2,
 25.3,
 21.4,
 0.526,
 21.8,
```

```
 43.1,
 0.506,
 18.3,
 39.8,
 0.46,
 0.9,
 2.6,
 0.341,
 6.1,
 15.9,
 0.381],
['1610612739',
 'Cleveland',
 'Cavaliers',
 'CLE',
 '',
 82,
 48.7,
 104.3,
 16.9,
 0.481,
 14.3,
 0.622,
 28.2,
 0.394,
 18.6,
 0.36,
 13.4,
 9.7,
 25.2,
 22.7,
 0.524,
 20.5,
 43.3,
 0.473,
 18.2,
 40.7,
 0.447,
 1.7,
 5.7,
 0.299,
 9.0,
 23.9,
 0.378],
['1610612746',
 'Los Angeles',
 'Clippers',
```

```
 'LAC',
 '',
 82,
 48.6,
 104.5,
 15.0,
 0.497,
 12.7,
 0.712,
 26.5,
 0.404,
 23.9,
 0.385,
 10.9,
 7.8,
 24.3,
 27.7,
 0.524,
 19.4,
 40.5,
 0.48,
 18.9,
 42.0,
 0.45,
 2.0,
 6.0,
 0.334,
 7.7,
 20.8,
 0.373],
['1610612760',
 'Oklahoma City',
 'Thunder',
 'OKC',
 '',
 82,
 48.6,
 110.2,
 16.1,
 0.48,
 15.3,
 0.677,
 24.7,
 0.402,
 20.6,
 0.39,
 11.9,
```

```
 9.5,
 23.7,
 23.3,
 0.524,
 23.7,
 47.7,
 0.497,
 17.5,
 38.7,
 0.451,
 1.6,
 5.1,
 0.321,
 6.6,
 18.6,
 0.356],
['1610612737',
 'Atlanta',
 'Hawks',
 'ATL',
 '',
 82,
 48.6,
 102.8,
 19.0,
 0.463,
 14.4,
 0.622,
 31.9,
 0.399,
 12.9,
 0.355,
 15.3,
 10.1,
 29.7,
 15.8,
 0.516,
 19.2,
 39.9,
 0.483,
 19.4,
 44.6,
 0.435,
 1.0,
 3.1,
 0.311,
 9.0,
```

```
 25.3,
 0.355],
['1610612745',
 'Houston',
 'Rockets',
 'HOU',
 '',
 82,
 48.6,
 106.5,
 17.2,
 0.433,
 15.2,
 0.629,
 24.5,
 0.37,
 16.5,
 0.349,
 13.8,
 10.2,
 22.8,
 19.6,
 0.516,
 22.2,
 47.1,
 0.472,
 15.5,
 36.4,
 0.426,
 2.3,
 7.4,
 0.318,
 8.4,
 23.5,
 0.355],
['1610612757',
 'Portland',
 'Trail Blazers',
 'POR',
 '',
 82,
 48.5,
 105.1,
 17.5,
 0.441,
 10.4,
 0.646,
```

```
 26.9,
 0.395,
 23.1,
 0.398,
 14.5,
 6.8,
 24.5,
 25.7,
 0.511,
 20.6,
 46.0,
 0.447,
 18.0,
 39.8,
 0.453,
 1.7,
 5.9,
 0.295,
 8.8,
 22.6,
 0.389],
['1610612758',
 'Sacramento',
 'Kings',
 'SAC',
 '',
 81,
 48.4,
 106.7,
 18.7,
 0.452,
 16.3,
 0.599,
 25.1,
 0.383,
 13.8,
 0.368,
 15.3,
 11.4,
 24.4,
 16.7,
 0.511,
 22.1,
 46.8,
 0.473,
 18.1,
 39.7,
```

```
 0.454,
 0.9,
 3.1,
 0.276,
 7.2,
 19.4,
 0.372],
['1610612764',
 'Washington',
 'Wizards',
 'WAS',
 '',
 82,
 48.5,
 104.1,
 15.4,
 0.48,
 15.1,
 0.629,
 28.7,
 0.394,
 17.8,
 0.371,
 11.3,
 10.1,
 27.4,
 22.1,
 0.511,
 20.0,
 41.5,
 0.483,
 19.5,
 44.3,
 0.439,
 0.7,
 2.7,
 0.254,
 8.0,
 21.5,
 0.371],
['1610612748',
 'Miami',
 'Heat',
 'MIA',
 '',
 82,
 48.6,
```

```
 100.0,
 17.9,
 0.488,
 15.7,
 0.623,
 22.5,
 0.386,
 15.2,
 0.371,
 13.9,
 10.4,
 22.3,
 19.5,
 0.508,
 22.7,
 46.4,
 0.49,
 15.7,
 35.2,
 0.445,
 0.8,
 2.9,
 0.282,
 5.3,
 15.1,
 0.347],
['1610612761',
 'Toronto',
 'Raptors',
 'TOR',
 '',
 81,
 48.5,
 102.7,
 23.0,
 0.462,
 11.6,
 0.636,
 24.6,
 0.394,
 17.1,
 0.369,
 18.2,
 7.6,
 22.3,
 21.1,
 0.504,
```

```
 22.5,
 48.8,
 0.461,
 14.1,
 32.4,
 0.436,
 1.8,
 5.6,
 0.327,
 6.8,
 17.7,
 0.384],
['1610612742',
 'Dallas',
 'Mavericks',
 'DAL',
 '',
 82,
 49.0,
 102.3,
 18.2,
 0.473,
 13.7,
 0.645,
 29.7,
 0.384,
 17.4,
 0.365,
 14.6,
 8.7,
 28.8,
 21.1,
 0.502,
 19.8,
 42.7,
 0.464,
 17.5,
 41.4,
 0.423,
 1.4,
 5.3,
 0.273,
 8.4,
 23.3,
 0.36],
['1610612766',
 'Charlotte',
```

```
 'Hornets',
 'CHA',
 '',
 82,
 48.6,
 103.4,
 16.8,
 0.459,
 11.4,
 0.565,
 30.3,
 0.389,
 17.9,
 0.353,
 12.8,
 8.0,
 28.1,
 22.3,
 0.502,
 20.0,
 44.6,
 0.449,
 17.0,
 39.8,
 0.427,
 1.8,
 6.0,
 0.297,
 8.9,
 23.4,
 0.379],
['1610612762',
 'Utah',
 'Jazz',
 'UTA',
 '',
 82,
 49.0,
 97.7,
 18.1,
 0.445,
 11.9,
 0.609,
 24.2,
 0.383,
 17.6,
 0.371,
```

```
 14.8,
 8.2,
 23.1,
 21.1,
 0.501,
 20.2,
 43.2,
 0.468,
 15.9,
 37.2,
 0.426,
 1.4,
 4.3,
 0.318,
 7.1,
 19.5,
 0.363],
['1610612753',
 'Orlando',
 'Magic',
 'ORL',
 '',
 81,
 48.7,
 102.0,
 18.0,
 0.456,
 13.3,
 0.599,
 29.5,
 0.398,
 14.4,
 0.363,
 15.0,
 9.6,
 29.0,
 18.3,
 0.5,
 21.0,
 44.1,
 0.475,
 18.5,
 42.6,
 0.435,
 0.7,
 2.7,
 0.249,
```

```
 7.1,
 19.5,
 0.363],
['1610612749',
 'Milwaukee',
 'Bucks',
 'MIL',
 '',
 82,
 48.7,
 99.0,
 17.4,
 0.463,
 18.0,
 0.621,
 18.3,
 0.37,
 13.3,
 0.362,
 14.3,
 12.3,
 18.7,
 17.2,
 0.499,
 25.2,
 52.8,
 0.477,
 13.2,
 29.4,
 0.448,
 1.1,
 4.0,
 0.27,
 4.3,
 11.6,
 0.37],
['1610612740',
 'New Orleans',
 'Pelicans',
 'NOP',
 '',
 82,
 48.5,
 102.7,
 19.9,
 0.458,
 11.8,
```

```
 0.641,
 23.3,
 0.387,
 21.0,
 0.382,
 16.3,
 7.5,
 22.4,
 24.3,
 0.498,
 20.6,
 44.7,
 0.46,
 17.9,
 41.1,
 0.434,
 0.6,
 2.6,
 0.247,
 7.9,
 21.2,
 0.374],
['1610612750',
 'Minnesota',
 'Timberwolves',
 'MIN',
 '',
 82,
 48.6,
 102.4,
 15.1,
 0.464,
 15.6,
 0.579,
 21.5,
 0.4,
 15.5,
 0.346,
 10.9,
 10.7,
 21.6,
 20.5,
 0.498,
 21.6,
 45.9,
 0.471,
 16.1,
```

```
 35.4,
 0.455,
 0.7,
 2.6,
 0.272,
 4.8,
 13.8,
 0.35],
['1610612754',
 'Indiana',
 'Pacers',
 'IND',
 '',
 82,
 48.8,
 102.2,
 13.7,
 0.453,
 15.0,
 0.603,
 25.3,
 0.375,
 19.0,
 0.361,
 10.5,
 10.6,
 25.7,
 23.7,
 0.497,
 21.9,
 47.1,
 0.465,
 16.4,
 38.1,
 0.431,
 1.7,
 5.7,
 0.299,
 6.4,
 17.4,
 0.368],
['1610612751',
 'Brooklyn',
 'Nets',
 'BKN',
 '',
 82,
```

```
 48.4,
 98.6,
 14.4,
 0.457,
 13.8,
 0.595,
 22.4,
 0.387,
 16.0,
 0.362,
 12.3,
 9.8,
 22.5,
 20.2,
 0.492,
 22.4,
 48.3,
 0.464,
 15.8,
 36.1,
 0.438,
 1.0,
 3.3,
 0.303,
 5.5,
 15.1,
 0.363],
['1610612765',
 'Detroit',
 'Pistons',
 'DET',
 '',
 82,
 48.7,
 102.0,
 17.5,
 0.464,
 14.1,
 0.597,
 23.7,
 0.378,
 19.2,
 0.367,
 14.2,
 10.0,
 22.3,
 23.4,
```

```
 0.491,
 22.3,
 49.3,
 0.452,
 15.7,
 37.2,
 0.422,
 0.9,
 4.0,
 0.227,
 8.1,
 22.2,
 0.366],
['1610612743',
 'Denver',
 'Nuggets',
 'DEN',
 '',
 82,
 48.6,
 101.9,
 15.9,
 0.406,
 16.6,
 0.598,
 23.5,
 0.376,
 15.2,
 0.349,
 13.4,
 11.7,
 23.3,
 19.2,
 0.489,
 21.3,
 47.6,
 0.448,
 16.4,
 37.8,
 0.434,
 1.1,
 4.3,
 0.264,
 6.9,
 19.5,
 0.354],
['1610612738',
```

```
 'Boston',
 'Celtics',
 'BOS',
 '',
 81,
 48.5,
 105.6,
 18.9,
 0.453,
 13.5,
 0.589,
 26.4,
 0.359,
 16.4,
 0.35,
 15.0,
 9.5,
 27.5,
 20.7,
 0.488,
 22.3,
 49.5,
 0.451,
 16.9,
 39.9,
 0.424,
 1.6,
 5.7,
 0.274,
 7.1,
 20.3,
 0.35],
['1610612741',
 'Chicago',
 'Bulls',
 'CHI',
 '',
 82,
 48.9,
 101.6,
 18.1,
 0.458,
 13.6,
 0.587,
 28.3,
 0.416,
 16.7,
```

```
 0.358,
 14.9,
 9.8,
 26.2,
 21.5,
 0.487,
 21.6,
 48.9,
 0.442,
 17.0,
 38.5,
 0.441,
 1.3,
 3.9,
 0.332,
 6.6,
 17.5,
 0.38],
['1610612755',
 'Philadelphia',
 '76ers',
 'PHI',
 '',
 82,
 48.6,
 97.4,
 19.7,
 0.445,
 13.4,
 0.621,
 26.5,
 0.361,
 12.8,
 0.322,
 16.6,
 9.0,
 26.2,
 17.3,
 0.487,
 21.0,
 46.6,
 0.449,
 15.3,
 37.4,
 0.409,
 1.6,
 5.7,
```

```
 0.281,
 7.7,
 21.8,
 0.354],
['1610612756',
 'Phoenix',
 'Suns',
 'PHX',
 '',
 82,
 48.4,
 100.9,
 15.6,
 0.44,
 14.0,
 0.565,
 25.6,
 0.376,
 17.3,
 0.354,
 13.1,
 10.2,
 24.9,
 21.5,
 0.487,
 20.6,
 46.1,
 0.447,
 16.6,
 39.5,
 0.421,
 1.4,
 5.0,
 0.288,
 7.6,
 20.8,
 0.363],
['1610612752',
 'New York',
 'Knicks',
 'NYK',
 '',
 82,
 48.5,
 98.4,
 10.4,
 0.447,
```

```
 14.4,
 0.568,
 26.6,
 0.387,
 18.9,
 0.38,
 7.9,
 10.1,
 26.7,
 23.1,
 0.483,
 20.9,
 47.6,
 0.439,
 15.9,
 36.4,
 0.438,
 1.5,
 4.9,
 0.305,
 5.9,
 16.6,
 0.358],
['1610612763',
 'Memphis',
 'Grizzlies',
 'MEM',
 '',
 82,
 48.6,
 99.1,
 16.4,
 0.44,
 14.4,
 0.546,
 22.5,
 0.379,
 13.2,
 0.364,
 13.3,
 10.6,
 23.2,
 16.6,
 0.477,
 20.7,
 45.2,
 0.459,
```

```
 16.1,
 38.5,
 0.418,
 0.7,
 2.5,
 0.278,
 5.4,
 16.0,
 0.34],
['1610612747',
 'Los Angeles',
 'Lakers',
 'LAL',
 '',
 82,
 48.3,
 97.3,
 15.6,
 0.441,
 11.3,
 0.584,
 19.8,
 0.355,
 22.3,
 0.347,
 13.1,
 7.9,
 21.2,
 28.1,
 0.46,
 21.1,
 50.3,
 0.42,
 14.0,
 34.5,
 0.406,
 2.2,
 7.9,
 0.278,
 5.6,
 16.7,
 0.335]]}
```