Problem Set #1

Advanced Methods in Applied Statistics

"Having four is preferable to just one." - *Cyan Jo*

Cyan Yong Ho Jo

14th February 2024

SCROUNGING

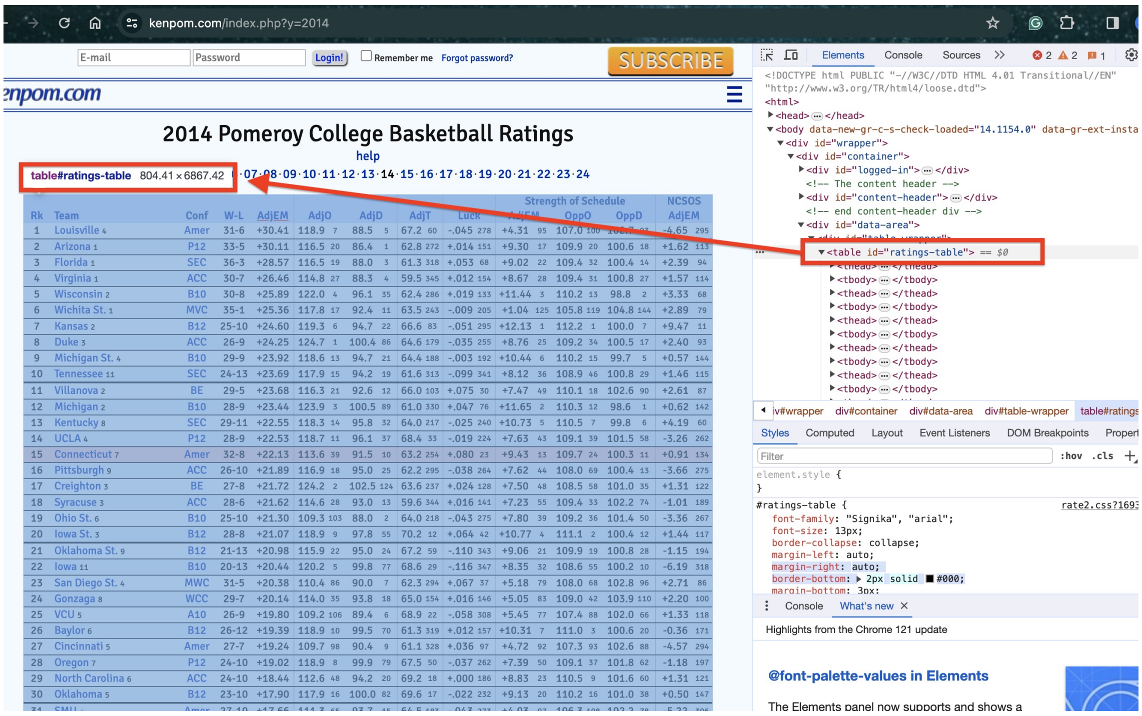
To scrape data on past or current NCAA basketball leagues from Ken Pomeroy's website, interacting with a web browser is essential. I plan to try two methods: using either *JavaScript* or *Python*'s *requests* package. Each method has its advantages. Using *JavaScript* requires the additional tool *Node.js*, but data querying is more intuitive because *JavaScript* is inherently web-oriented, being event-driven and bi-directional. Furthermore, I can automate this process by calling it from my *Python* code for data analysis using the *subprocess.run()* function, which allows the execution of external files. Similarly, using *Python* necessitates installing the requests package, but it offers simplicity and ease of use. Regardless of the method, the data will be stored and managed in a subdirectory of my workspace.

Finding the URL pattern

Ken Pomeroy's website features a simple layout. The homepage, found at `https://kenpom.com/`, presents data from the latest NCAA basketball season. By appending a query string with a specific year to the main URL, you can access data for that particular year. For instance, if I were to scrap the content of the year 2014, I simply need to add a PHP query at the end of the main URL: `https://kenpom.com/index.php?y=2014`.

Extract HTML

Before starting the scraping process, it is beneficial to investigate the web page's structure by examining its source. I prefer using *Chrome* by *Google* because it is the most widely used web browser among web developers. This allows me to confirm the hierarchical structure and identify data stored within tabular containers. See Figure1.



2014 Pomeroy College Basketball Ratings

									Strength of Schedule			NCSOS
Rk	Team	Conf	W-L	AdjEM	AdjO	AdjD	AdjT	Luck	AdjEM	OppO	OppD	AdjEM
1	Louisville 4	Amer	31-6	+30.41	118.9 7	88.5 5	67.2 60	-.045 278	+4.31 95	107.0 100	[illegible]	-4.65 295
2	Arizona 1	P12	33-5	+30.11	116.5 20	86.4 1	62.8 272	+.014 151	+9.30 17	109.9 20	100.6 18	+1.62 113
3	Florida 1	SEC	36-3	+28.57	116.5 19	88.0 3	61.3 318	+.053 68	+9.02 22	109.4 32	100.4 14	+2.39 94
4	Virginia 1	ACC	30-7	+26.46	114.8 27	88.3 4	59.5 345	+.012 154	+8.67 28	109.4 31	100.8 27	+1.57 114
5	Wisconsin 2	B10	30-8	+25.89	122.0 4	96.1 35	62.4 286	+.019 133	+11.44 3	110.2 13	98.8 2	+3.33 68
6	Wichita St. 1	MVC	35-1	+25.36	117.8 17	92.4 11	63.5 243	-.009 205	+1.04 125	105.8 119	104.8 144	+2.89 79
7	Kansas 2	B12	25-10	+24.60	119.3 6	94.7 22	66.6 83	-.051 295	+12.13 1	112.2 1	100.0 7	+9.47 11
8	Duke 3	ACC	26-9	+24.25	124.7 1	100.4 86	64.6 179	-.035 255	+8.76 25	109.2 34	100.5 17	+2.40 93
9	Michigan St. 4	B10	29-9	+23.92	118.6 13	94.7 21	64.4 188	-.003 192	+10.44 6	110.2 15	99.7 5	+0.57 144
10	Tennessee 11	SEC	24-13	+23.69	117.9 15	94.2 19	61.6 313	-.099 341	+8.12 36	108.9 46	100.8 29	+1.46 115
11	Villanova 2	BE	29-5	+23.68	116.3 21	92.6 12	66.0 103	+.075 30	+7.47 49	110.1 18	102.6 90	+2.61 87
12	Michigan 2	B10	28-9	+23.44	123.9 3	100.5 89	61.0 330	+.047 76	+11.65 2	110.3 12	98.6 1	+0.62 142
13	Kentucky 8	SEC	29-11	+22.55	118.3 14	95.8 32	64.0 217	-.025 240	+10.73 5	110.5 7	99.8 6	+4.19 60
14	UCLA 4	P12	28-9	+22.53	118.7 11	96.1 37	68.4 33	-.019 224	+7.63 43	109.1 39	101.5 58	-3.26 262
15	Connecticut 7	Amer	32-8	+22.13	113.6 39	91.5 10	63.2 254	+.080 23	+9.43 13	109.7 24	100.3 11	+0.91 134
16	Pittsburgh 9	ACC	26-10	+21.89	116.9 18	95.0 25	62.2 295	-.038 264	+7.62 44	108.0 69	100.4 13	-3.66 275
17	Creighton 3	BE	27-8	+21.72	124.2 2	102.5 124	63.6 237	+.024 128	+7.50 48	108.5 58	101.0 35	+1.31 122
18	Syracuse 3	ACC	28-6	+21.62	114.6 28	93.0 13	59.6 344	+.016 141	+7.23 55	109.4 33	102.2 74	-1.01 189
19	Ohio St. 6	B10	25-10	+21.30	109.3 103	88.0 2	64.0 218	-.043 275	+7.80 39	109.2 36	101.4 50	-3.36 267
20	Iowa St. 3	B12	28-8	+21.07	118.9 9	97.8 55	70.2 12	+.064 42	+10.77 4	111.1 2	100.4 12	+1.44 117
21	Oklahoma St. 9	B12	21-13	+20.98	115.9 22	95.0 24	67.2 59	-.110 343	+9.06 21	109.9 19	100.8 28	-1.15 194
22	Iowa 11	B10	20-13	+20.44	120.2 5	99.8 77	68.6 29	-.116 347	+8.35 32	108.6 55	100.2 10	-6.19 318
23	San Diego St. 4	MWC	31-5	+20.38	110.4 86	90.0 7	62.3 294	+.067 37	+5.18 79	108.0 68	102.8 96	+2.71 86
24	Gonzaga 8	WCC	29-7	+20.14	114.0 35	93.8 18	65.0 154	+.016 146	+5.05 83	109.0 42	103.9 110	+2.20 100
25	VCU 5	A10	26-9	+19.80	109.2 106	89.4 6	68.9 22	-.058 308	+5.45 77	107.4 88	102.0 66	+1.33 118
26	Baylor 6	B12	26-12	+19.39	118.9 10	99.5 70	61.3 319	+.012 157	+10.31 7	111.0 3	100.6 20	-0.36 171
27	Cincinnati 5	Amer	27-7	+19.24	109.7 98	90.4 9	61.1 328	+.036 97	+4.72 92	107.3 93	102.6 88	-4.57 294
28	Oregon 7	P12	24-10	+19.02	118.9 8	99.9 79	67.5 50	-.037 262	+7.39 50	109.1 37	101.8 62	-1.18 197
29	North Carolina 6	ACC	24-10	+18.44	112.6 48	94.2 20	69.2 18	+.000 186	+8.83 23	110.5 9	101.6 60	+1.31 121
30	Oklahoma 5	B12	23-10	+17.90	117.9 16	100.0 82	69.6 17	-.022 232	+9.13 20	110.2 16	101.0 38	+0.50 147

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
 ▶<head> … </head>
 ▼<body data-new-gr-c-s-check-loaded="14.1154.0" data-gr-ext-insta
   ▼<div id="wrapper">
     ▼<div id="container">
       ▶<div id="logged-in"> … </div>
        <!-- The content header -->
       ▶<div id="content-header"> … </div>
        <!-- end content-header div -->
       ▼<div id="data-area">
           ▼<table id="ratings-table"> == $0
             ▶<thead> … </thead>
             ▶<tbody> … </tbody>
             ▶<thead> … </thead>
             ▶<tbody> … </tbody>
             ▶<thead> … </thead>
             ▶<tbody> … </tbody>
             ▶<thead> … </thead>
             ▶<tbody> … </tbody>
             ▶<thead> … </thead>
             ▶<tbody> … </tbody>
```

iv#wrapper div#container div#data-area div#table-wrapper table#ratings

```
element.style {
}
#ratings-table {                                   rate2.css?1693
   font-family: "Signika", "arial";
   font-size: 13px;
   border-collapse: collapse;
   margin-left: auto;
   margin-right: auto;
   border-bottom: ▶ 2px solid ■ #000;
   margin-bottom: 3px;
```

Highlights from the Chrome 121 update

@font-palette-values in Elements

The Elements panel now supports and shows a

Figure 1: "In *Chrome*'s Developer Mode, the source is analysed, and visual indicators show how each part of the source corresponds to specific elements on the page. The tabular data I identified (highlighted in a red box) are delineated by shading.

Parsing HTML files

The extracted HTML files contain both redundant data and the data I need for this project. For this purpose, I require another web scraping package called *BeautifulSoup*. *.find('table')* method in *BeautifulSoup* allows me to extract the data stored in a table, which is precisely what I need.

The header of the table extracted by *BeautifulSoup* has several undesirable features. Firstly, the header is not confined to a single row; it spans two rows, which complicates data processing. I need to modify the header to fit within a single row. Secondly, each header item contains unnecessary strings. For instance, a header item labeled 'Rk' is presented with additional, extraneous characters.

```
Unnamed: 0_level_0     Rk      Unnamed: 0_level_2     Rk      Unnamed: 0
    _level_4     Rk      Unnamed: 0_level_6     Rk      Unnamed: 0_level_8
       Rk      Unnamed: 0_level_10     Rk      Unnamed: 0_level_12     Rk
        Unnamed: 0_level_14     Rk      Unnamed: 0_level_16     Rk
```

I planned to solve these two problems by removing the current header and assigning a new one written by myself. For some columns, there is a tailing column that has an adjusted value of the original column. I chose to tag them with $_enh$ postfix.

```
HEADER = ['Rk','Team','Conf','W','L','AdjEM','AdjO','AdjO_enh','AdjD','
    AdjD_enh','AdjT','AdjT_enh','Luck','Luck_enh','AdjEM_SoS','
    AdjEM_enh_SoS','OppO_SoS','OppO_enh_SoS','OppD_SoS','OppD_enh_SoS',
    'AdjEM_NCSOS', 'AdjEM_enh_NCSOS']
# newly defined header.
```

The third issue I encountered was the format of the values of the wins and losses. They are combined in a string having '(W)-(L)' form so it needed to be separated into two columns '(W)' and '(L)'. Finally, I need to convert the numbers into a numerical format so that *Python* interpreter can recognise them as numbers. The initial format of the extracted data was string.

MATPLOTLIB FORMAT

The selection of color combinations in any plot is essential for ensuring visibility and comprehension. I have decided to introduce a preset for the *Matplotlib* package, which will establish the size and thickness of the lines, both axes, ticks, and font sizes for labels, titles, and legends. Additionally, I will set a cyclic property for markers and colors for the plot by using *plt.cycler()* function. I utilised colour searching and generating tool from this web-site: *Coolors*. It shows patches of colours I picked and I can check the hexadecimal code of each colour and its name.

EX1

The histograms depicted below illustrate the AdjD values for teams belonging to the ACC, SEC, B10, BSky, or A10 conferences, utilizing data from 2014. When plotting a bar-type histogram, I encountered an issue with overlapping bars. To resolve this, I adjusted the layout by separating the bars and placing each one within the same bin but positioned next to each other, similar to the arrangement seen in Fig2 overlapping conference data. For the sake of visibility and recognition, I tried removing either the lines(in Fig3) or the bars(in Fig4).

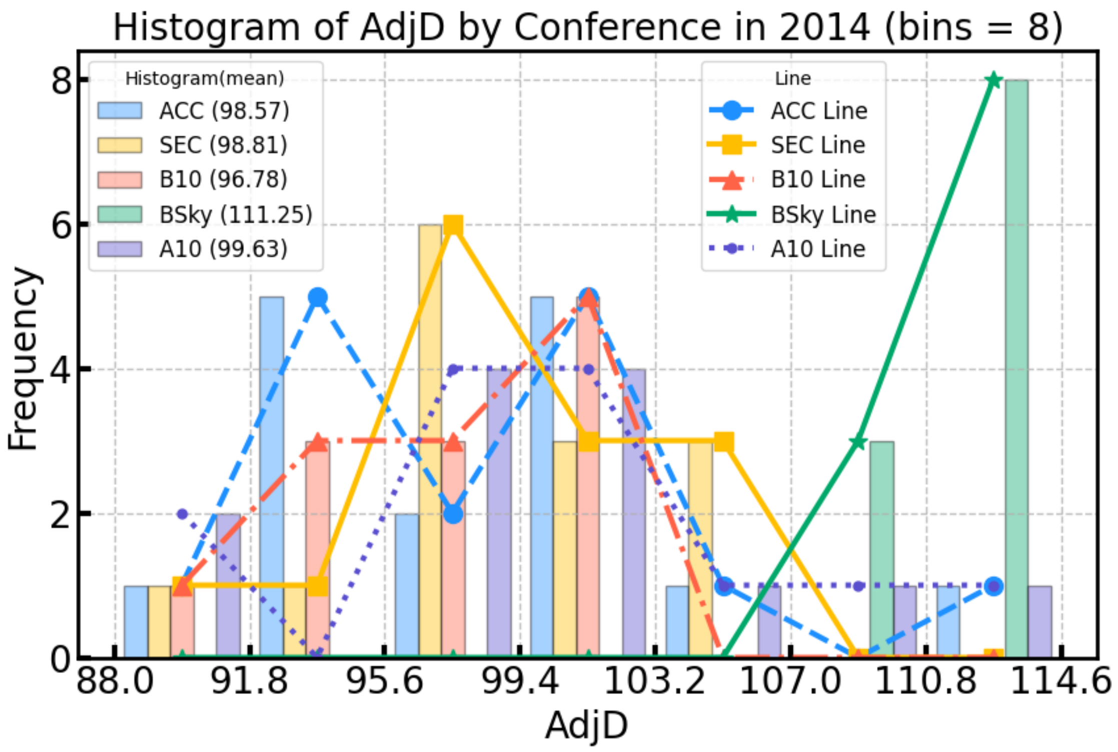


Figure 2

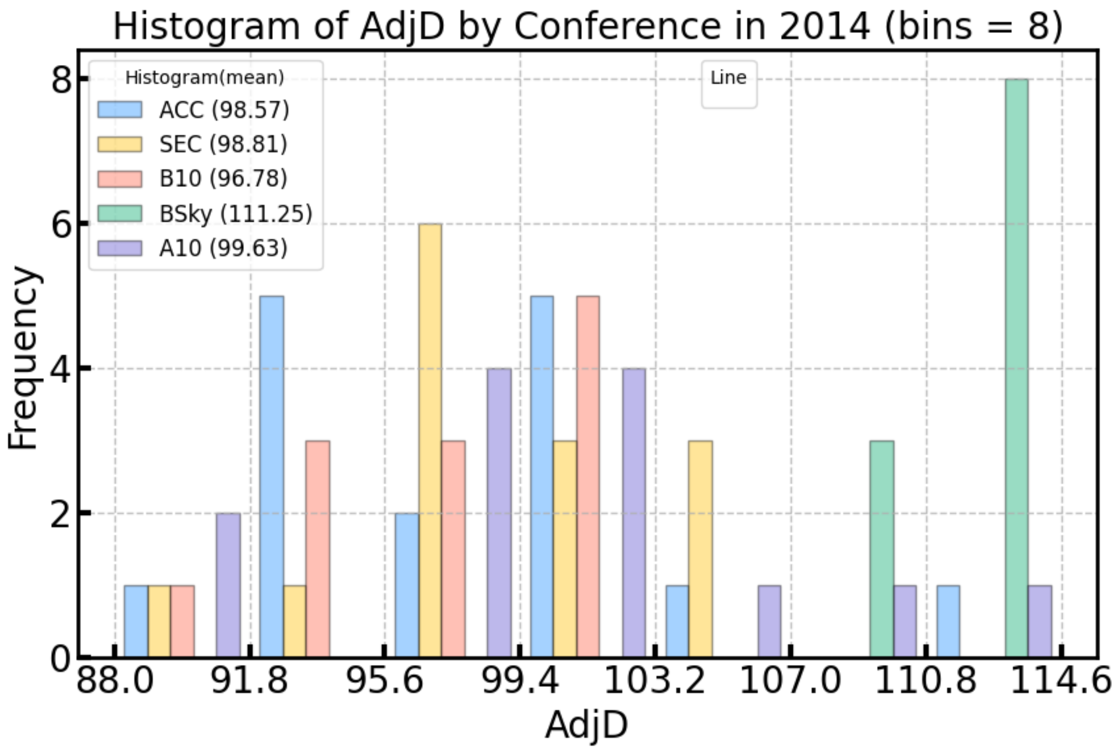


Figure 3

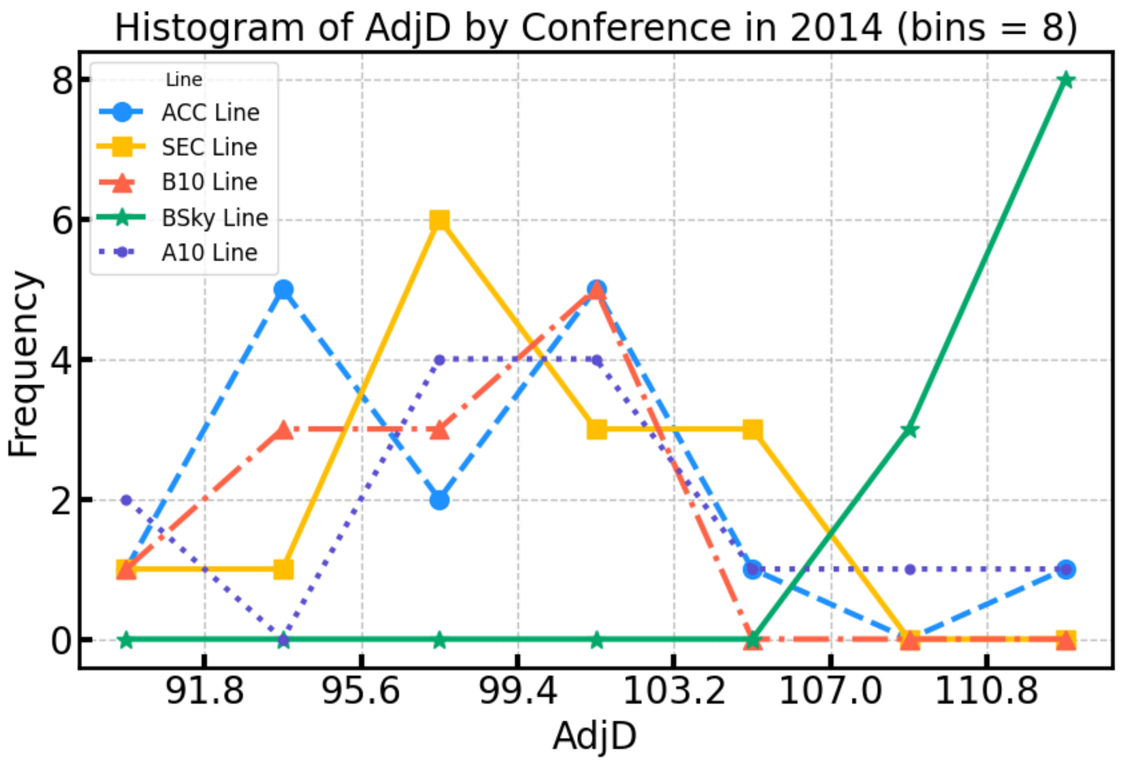


Figure 4

EX2

The scatter graph depicted in Fig5 showcases the fluctuation in AdjO values across teams affiliated with the ACC, SEC, B10, BSky, or A10 conferences between 2009 and 2014, about their AdjO values from 2009. The methodology employed for computing the discrepancies in AdjO might seem rudimentary. However, alterations in conference compositions throughout the five years have resulted in significant outliers in AdjO values, nearing the 100 mark. To address this issue, I opted to eliminate teams with available data for either the 2009 or 2014 periods. I checked this by plotting the difference of each team on Fig6. Recalling ACC is the conference whose teams marked the lowest values of AdjD in 2014 overall, the teams in the conference have not experienced a notable increase in AdjO in general.

The mean and median are shown in TableI. ACC is the only conference that has undergone decreased value in AdjO in 2014 compared to 2009.

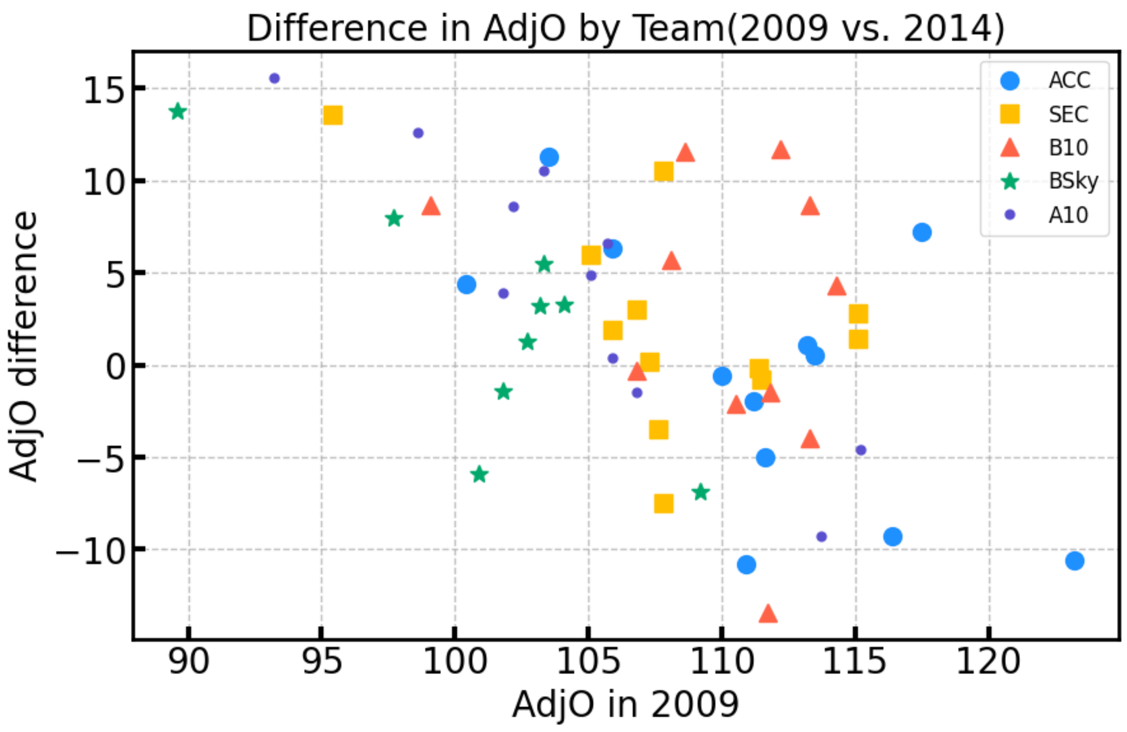


Figure 5

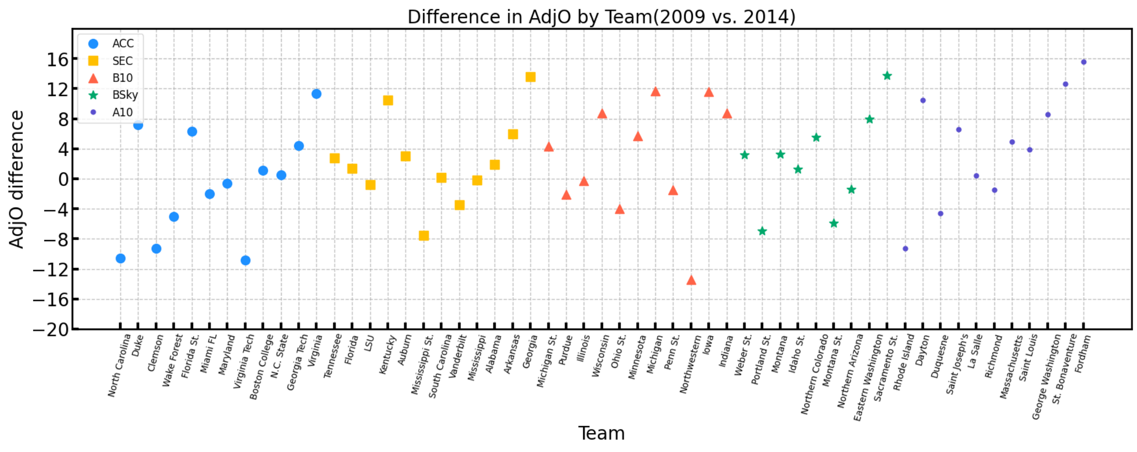


Figure 6

Conference	Mean	Median
ACC	-0.625	-0.050
SEC	2.283	1.650
B10	2.673	4.300
BSky	2.322	3.200
A10	4.336	4.900
Others	2.595	1.900

Table I: Mean and median of difference in AdjO values for five different conferences

EX3

Below are histograms of AdjD for the six conferences in 2014 Fig7, Fig8, and Fig9. AdjD of BE seems to have a similar trend as B10.

The mean and median of the difference in AdjO between 2014 and 2009 are shown in TableII.

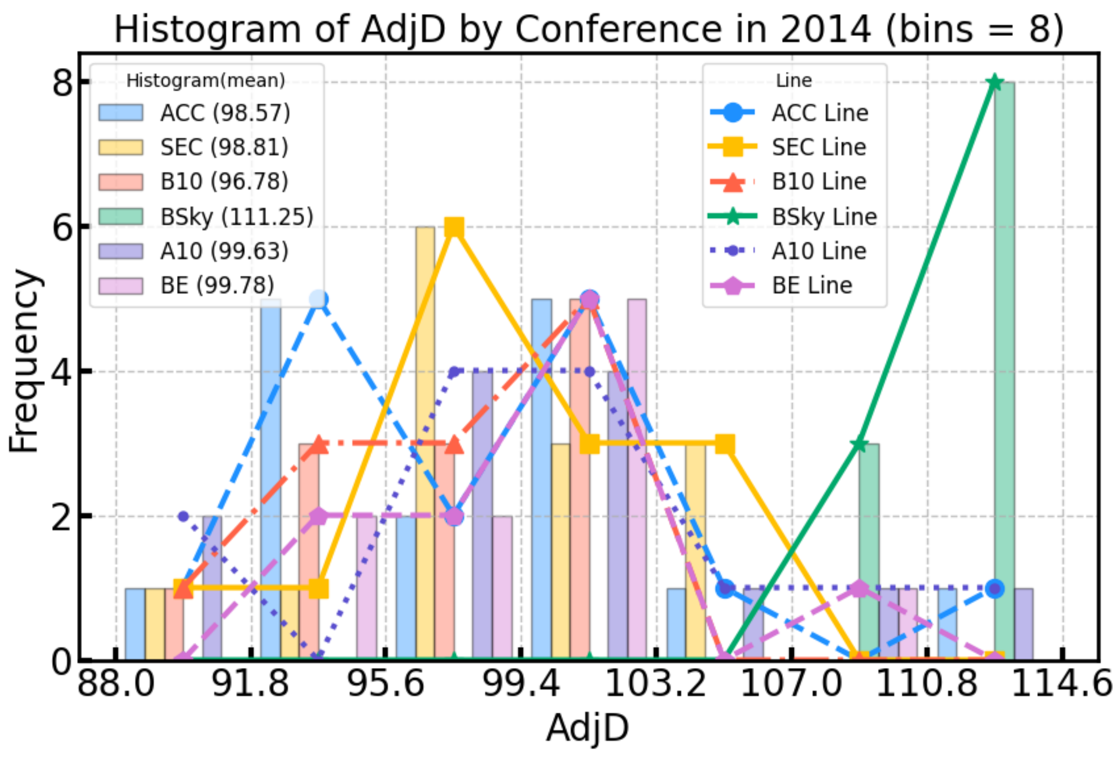


Figure 7

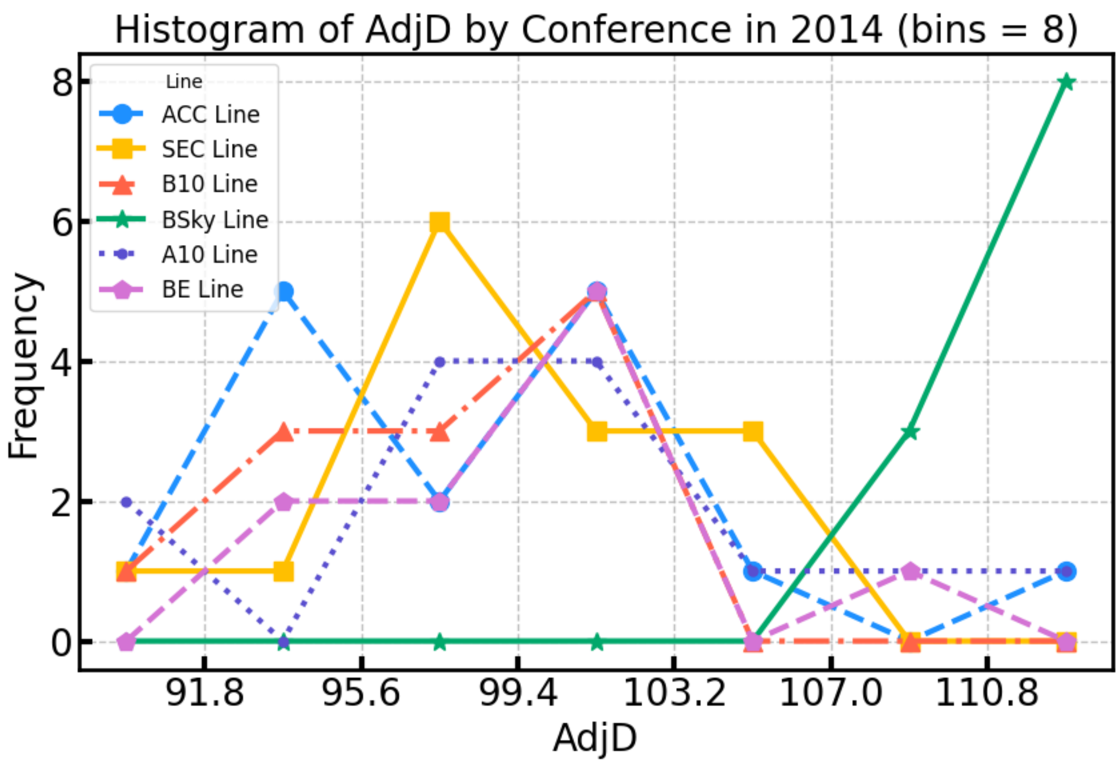


Figure 8

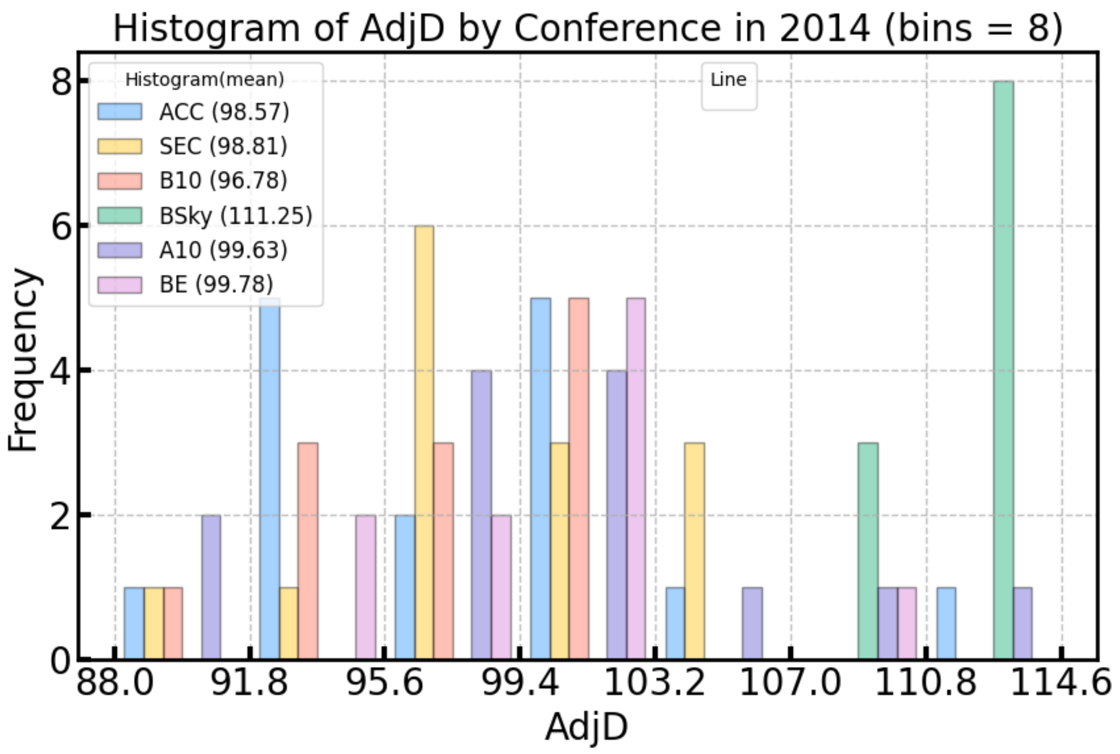


Figure 9

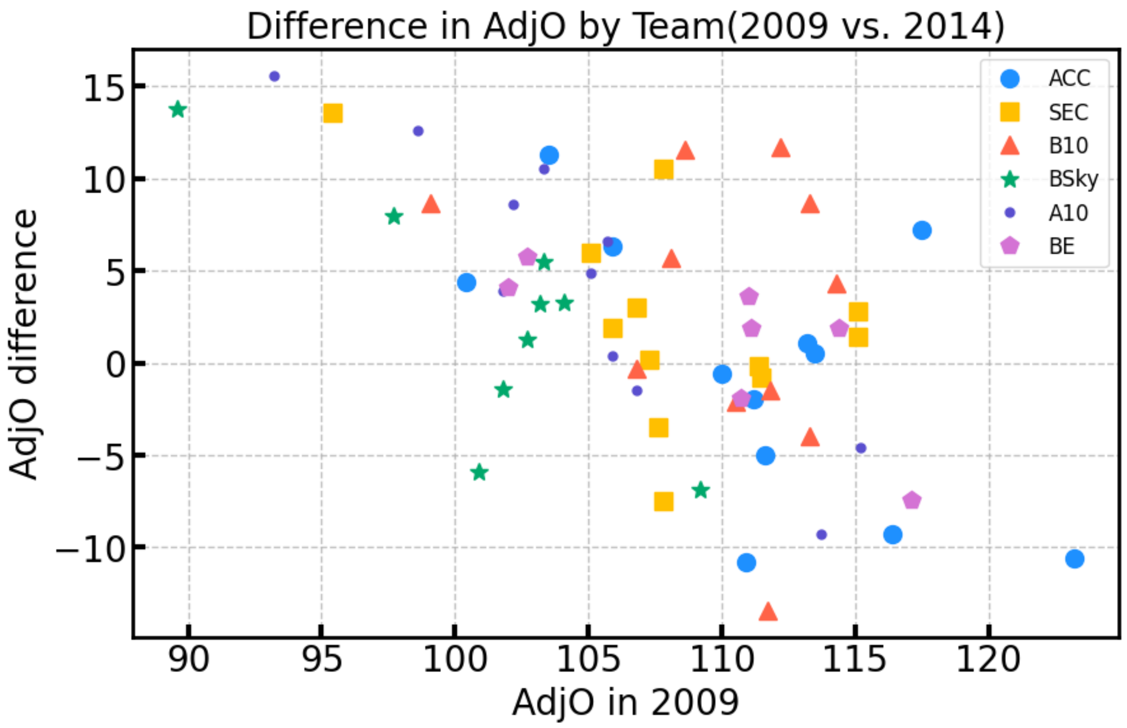


Figure 10

Conference	Mean	Median
ACC	-0.625	-0.050
SEC	2.283	1.650
B10	2.673	4.300
BSky	2.322	3.200
A10	4.336	4.900
BE	1.143	1.900
Others	2.624	1.850

Table II: Mean and median of difference in AdjO values for six different conferences

EX4

I read the .pdf file and parsed it using javascript. It was a trick to handle non-ASCII characters like the letters with diacritics. The patterns I found are these.

- each name ends with a comma
- redundant line exists
- some lines has more than one names
- replacing special characters introduce additional line

The name in the middle of the alphabetically ordered list I found is **K. GILL**.