EFFECTIVENESS OF HOME HEALTHCARE AGENCIES REGISTERED WITH MEDICARE USING PYTHON

NAME: Anusha Manjappa

CIN: 305851746

CIS5810 Project2

Fall 2017

Anusha Manjappa

A) Dataset URL:

Dataset:https://data.medicare.gov/Home-Health-Compare/Home-Health-CareAgencies/6jpmsxkc/data

The dataset content is drawn from the Medicare.gov’s compare websites and directories. This site provides direct access to the official data from the Centers for Medicare & Medicaid Services (CMS) that are used on the Medicare.gov Compare Websites and Directories. One of the prominent fields that portray the effectiveness of the agency is the ‘Quality of Patient Care Star Rating ‘field. Home Health Compare uses a star rating between 1 and 5 to show people how a home health agency compares to other home health agencies on measurements of their performance. The star ratings are based on 9 measures of quality that give a general overview of performance. Across the country, most agencies fall “in the middle” with 3 or 3½ stars being the average rating across the 9 measures. A star rating higher than 3½ means that an agency performed better than average compared to other agencies. A star rating lower than 3 means that an agency’s performance was below average compared to other home health agencies. This dataset contains a list of all Home Health Agencies that have been registered with Medicare. The list includes addresses, phone numbers, and quality measure ratings for each agency. The dataset also contains the list of the different services offered by each of the agencies along with how effective each of the services were individually for all the agencies. It consists of a total of 54 columns and 11,802 rows which provides the most detailed information. I have considered the most important 21 columns out of the 54 columns.

B) Data Cleaning

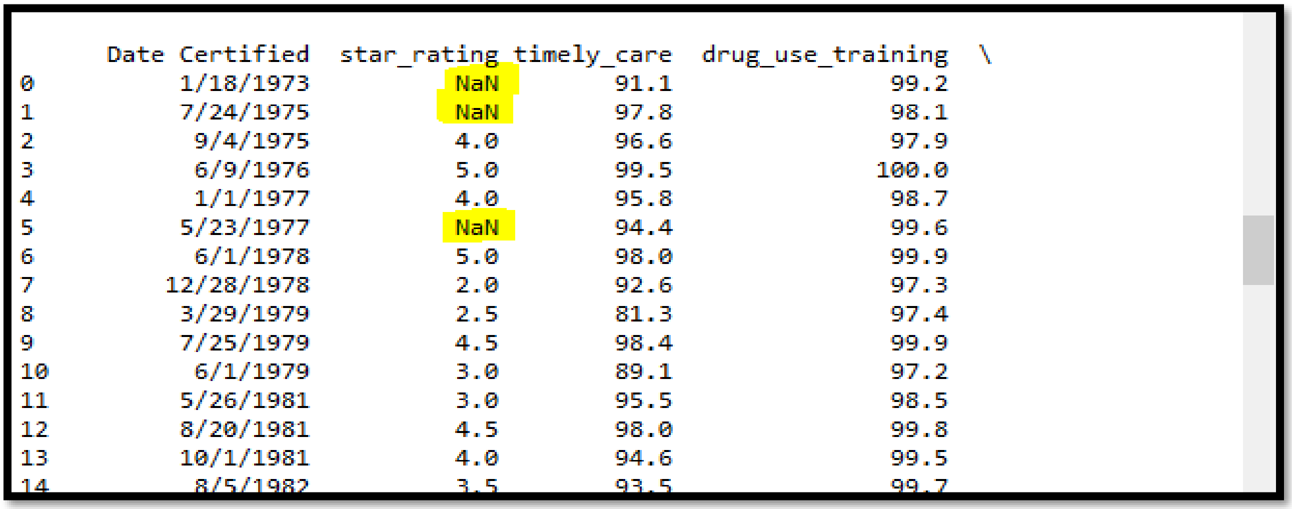
a) Missing Values replaced with mean value

Some blocks in the dataset were empty as the value could not be determined or was not known. Therefore, I have replaced such values with the mean value. This would make future calculations much easier and accurate. I have shown one such example where I replaced the blank values of the column Star Rating with the mean value of the column grouped according to the field State. This would make sure that I fill in the mean of the star rating value state wise which would happen to be more accurate. The **files** are read using read function and stored in **data frames. Strings** are used to mention column names.

Python Concepts Used:

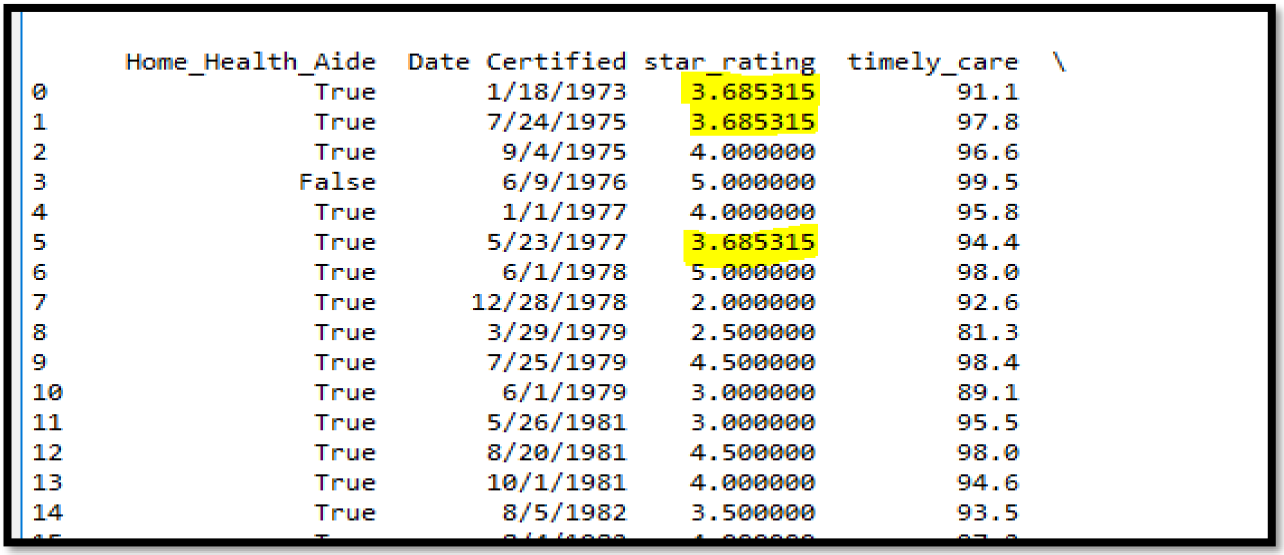
- Pandas Data Frame
- Files
- String

Before

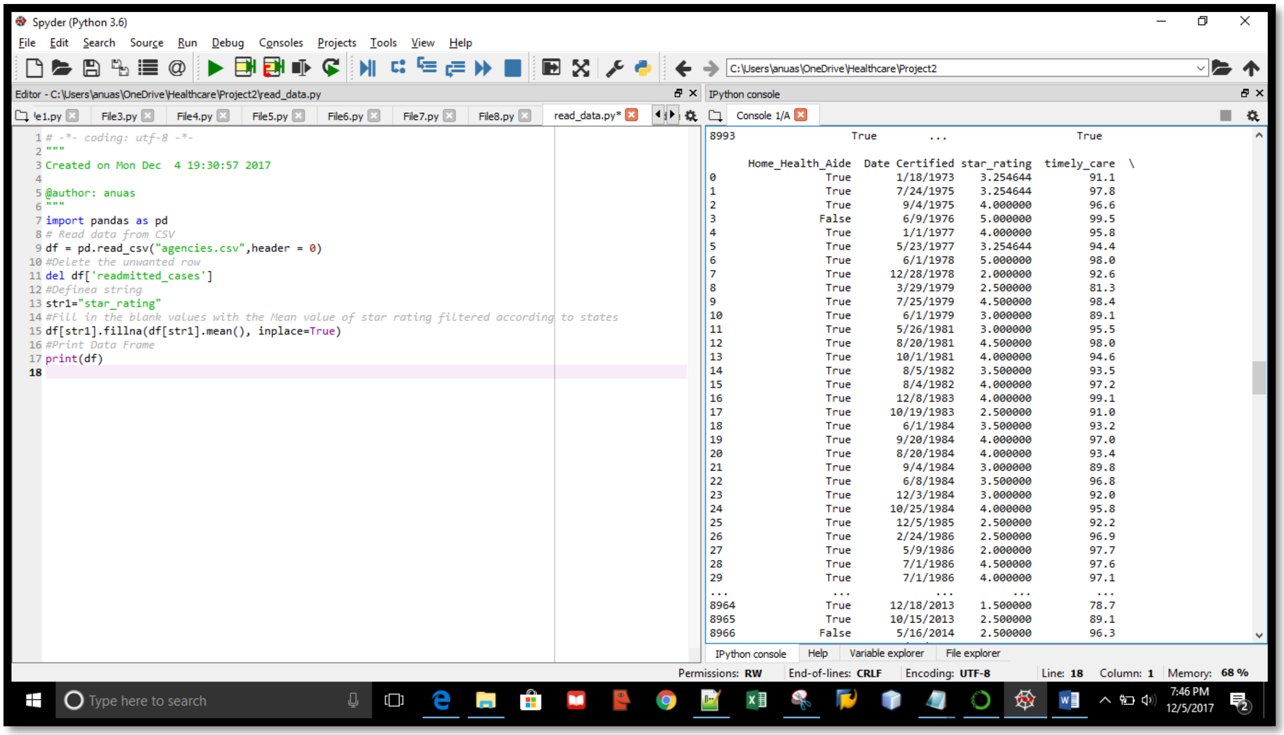


```
     Date Certified  star_rating timely_care  drug_use_training  \
0          1/18/1973          NaN        91.1               99.2
1          7/24/1975          NaN        97.8               98.1
2           9/4/1975          4.0        96.6               97.9
3           6/9/1976          5.0        99.5              100.0
4           1/1/1977          4.0        95.8               98.7
5          5/23/1977          NaN        94.4               99.6
6           6/1/1978          5.0        98.0               99.9
7         12/28/1978          2.0        92.6               97.3
8          3/29/1979          2.5        81.3               97.4
9          7/25/1979          4.5        98.4               99.9
10          6/1/1979          3.0        89.1               97.2
11         5/26/1981          3.0        95.5               98.5
12         8/20/1981          4.5        98.0               99.8
13         10/1/1981          4.0        94.6               99.5
14          8/5/1982          3.5        93.5               99.7
```

After



```
     Home_Health_Aide  Date Certified  star_rating  timely_care  \
0                True       1/18/1973     3.685315         91.1
1                True       7/24/1975     3.685315         97.8
2                True        9/4/1975     4.000000         96.6
3               False        6/9/1976     5.000000         99.5
4                True        1/1/1977     4.000000         95.8
5                True       5/23/1977     3.685315         94.4
6                True        6/1/1978     5.000000         98.0
7                True      12/28/1978     2.000000         92.6
8                True       3/29/1979     2.500000         81.3
9                True       7/25/1979     4.500000         98.4
10               True        6/1/1979     3.000000         89.1
11               True       5/26/1981     3.000000         95.5
12               True       8/20/1981     4.500000         98.0
13               True       10/1/1981     4.000000         94.6
14               True        8/5/1982     3.500000         93.5
```


Spyder (Python 3.6)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\anuas\OneDrive\Healthcare\Project2
Editor - C:\Users\anuas\OneDrive\Healthcare\Project2\read_data.py
le1.py File3.py File4.py File5.py File6.py File7.py File8.py read_data.py*
IPython console
Console 1/A
IPython console Help Variable explorer File explorer
Permissions: RW End-of-lines: CRLF Encoding: UTF-8 Line: 18 Column: 1 Memory: 68 %
Type here to search
7:46 PM
12/5/2017

```
# -*- coding: utf-8 -*-
"""
Created on Mon Dec  4 19:30:57 2017

@author: anuas
"""
import pandas as pd
# Read data from CSV
df = pd.read_csv("agencies.csv",header = 0)
#Delete the unwanted row
del df['readmitted_cases']
#Definea string
str1="star_rating"
#Fill in the blank values with the Mean value of star rating filtered according to states
df[str1].fillna(df[str1].mean(), inplace=True)
#Print Data Frame
print(df)
```

```
8993                True       ...                      True

      Home_Health_Aide  Date Certified star_rating  timely_care  \
0                 True       1/18/1973    3.254644         91.1
1                 True       7/24/1975    3.254644         97.8
2                 True        9/4/1975    4.000000         96.6
3                False        6/9/1976    5.000000         99.5
4                 True        1/1/1977    4.000000         95.8
5                 True       5/23/1977    3.254644         94.4
6                 True        6/1/1978    5.000000         98.0
7                 True      12/28/1978    2.000000         92.6
8                 True       3/29/1979    2.500000         81.3
9                 True       7/25/1979    4.500000         98.4
10                True        6/1/1979    3.000000         89.1
11                True       5/26/1981    3.000000         95.5
12                True       8/20/1981    4.500000         98.0
13                True       10/1/1981    4.000000         94.6
14                True        8/5/1982    3.500000         93.5
15                True        8/4/1982    4.000000         97.2
16                True       12/8/1983    4.000000         99.1
17                True      10/19/1983    2.500000         91.0
18                True        6/1/1984    3.500000         93.2
19                True       9/20/1984    4.000000         97.0
20                True       8/20/1984    4.000000         93.4
21                True        9/4/1984    3.000000         89.8
22                True        6/8/1984    3.500000         96.8
23                True       12/3/1984    3.000000         92.0
24                True      10/25/1984    4.000000         95.8
25                True       12/5/1985    2.500000         92.2
26                True       2/24/1986    2.500000         96.9
27                True        5/9/1986    2.000000         97.7
28                True        7/1/1986    4.500000         97.6
29                True        7/1/1986    4.000000         97.1
...                ...             ...         ...          ...
8964              True      12/18/2013    1.500000         78.7
8965              True      10/15/2013    2.500000         89.1
8966             False       5/16/2014    2.500000         96.3
```

Code Snippet:

```
# -*- coding: utf-8 -*-

"""

Created on Mon Dec  4 19:30:57 2017

@author: anuas

"""
```

```
import pandas as pd

# Read data from CSV

df = pd.read_csv("agencies.csv",header = 0)

#Delete the unwanted row

del df['readmitted_cases']

#Fill in the blank values with the Mean value of star rating filtered according to states

Str1="star_rating"

df[str1].fillna(df[str1].mean(), inplace=True)

#Print Data Frame

print(df)
```

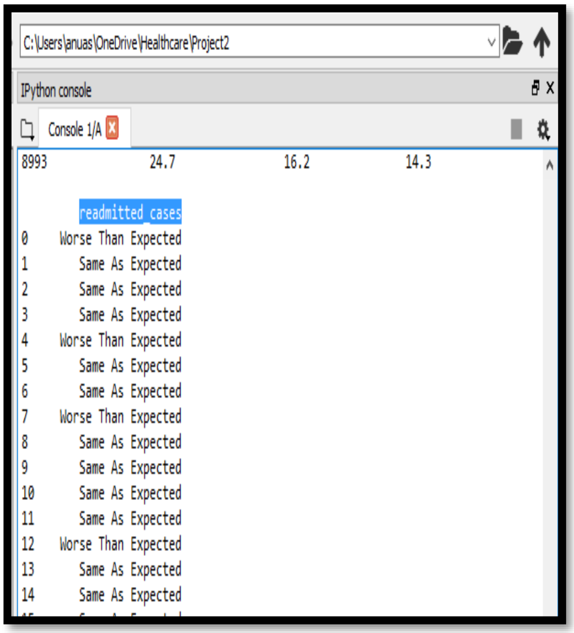
b) Deleting unwanted columns

This dataset had some rows that had data that was not useful for any analysis in python and since my dataset had a lot of columns, deleting the unwanted ones made it more convenient for me to view the data in the editor of anaconda spyder. Therefore, I have deleted such columns of data. This would make future calculations and visualizations much easier and accurate. The pictures show the columns to be deleted and the count of the columns after deleting it. The **files** are read using read function and stored in **data frames.**

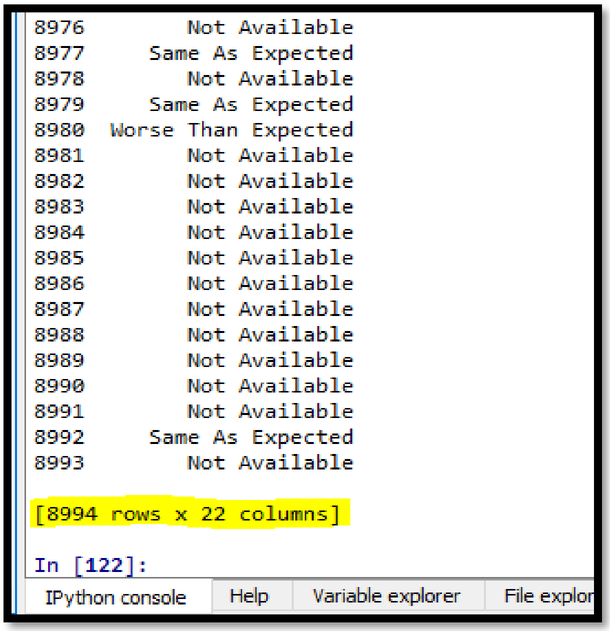
Python Concepts Used:

- Pandas Data Frame
- Files

Before



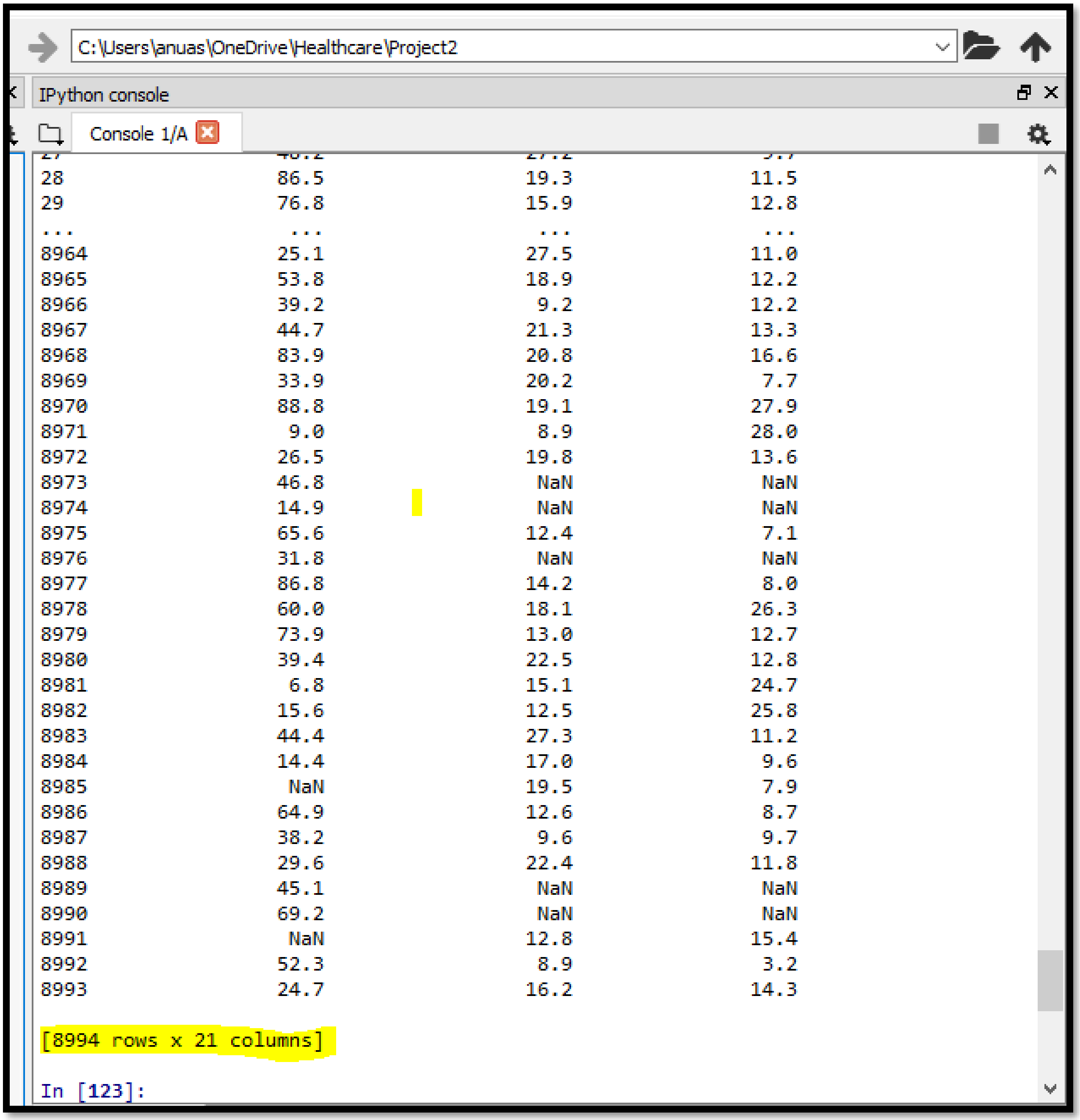
```
8993              24.7                16.2              14.3


       readmitted_cases
0    Worse Than Expected
1       Same As Expected
2       Same As Expected
3       Same As Expected
4    Worse Than Expected
5       Same As Expected
6       Same As Expected
7    Worse Than Expected
8       Same As Expected
9       Same As Expected
10      Same As Expected
11      Same As Expected
12   Worse Than Expected
13      Same As Expected
14      Same As Expected
```



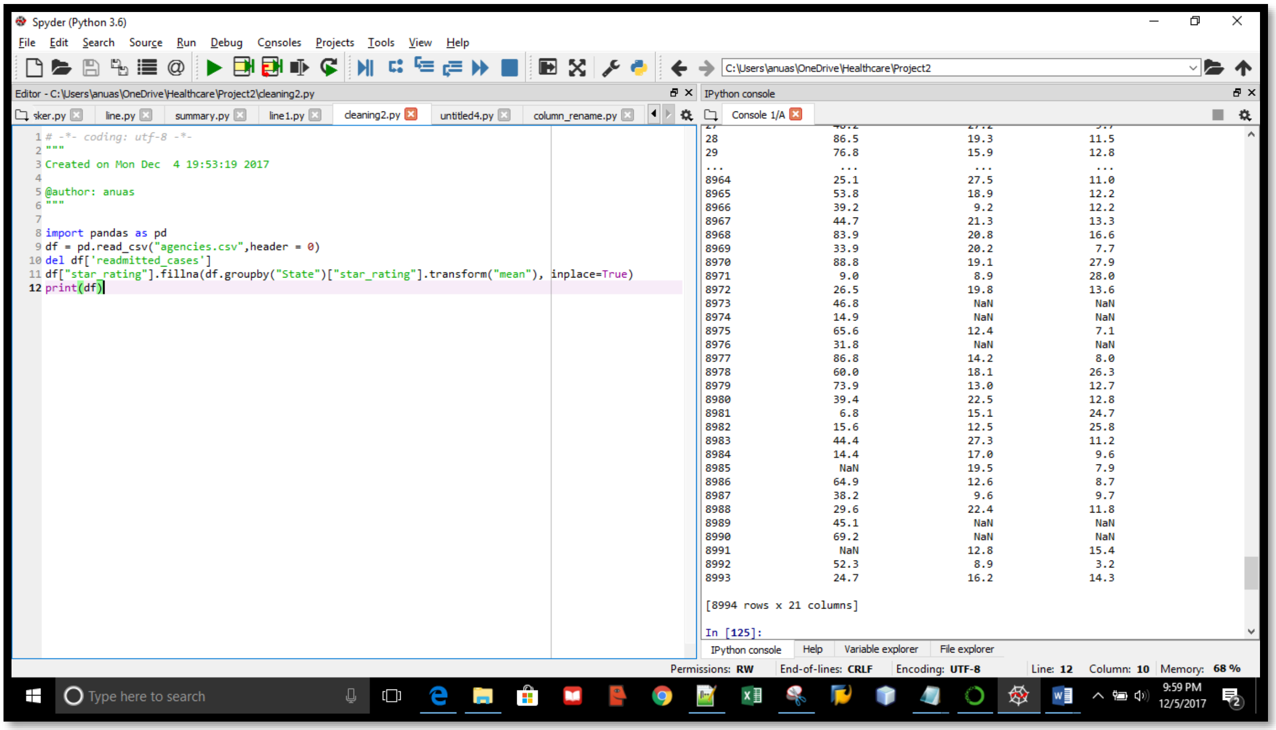
```
8976         Not Available
8977      Same As Expected
8978         Not Available
8979      Same As Expected
8980   Worse Than Expected
8981         Not Available
8982         Not Available
8983         Not Available
8984         Not Available
8985         Not Available
8986         Not Available
8987         Not Available
8988         Not Available
8989         Not Available
8990         Not Available
8991         Not Available
8992      Same As Expected
8993         Not Available

[8994 rows x 22 columns]

In [122]:
```

The pictures show the columns to be deleted and the count of the columns after deleting it.

After



```
28                  86.5                19.3                11.5
29                  76.8                15.9                12.8
...                  ...                 ...                 ...
8964                25.1                27.5                11.0
8965                53.8                18.9                12.2
8966                39.2                 9.2                12.2
8967                44.7                21.3                13.3
8968                83.9                20.8                16.6
8969                33.9                20.2                 7.7
8970                88.8                19.1                27.9
8971                 9.0                 8.9                28.0
8972                26.5                19.8                13.6
8973                46.8                 NaN                 NaN
8974                14.9                 NaN                 NaN
8975                65.6                12.4                 7.1
8976                31.8                 NaN                 NaN
8977                86.8                14.2                 8.0
8978                60.0                18.1                26.3
8979                73.9                13.0                12.7
8980                39.4                22.5                12.8
8981                 6.8                15.1                24.7
8982                15.6                12.5                25.8
8983                44.4                27.3                11.2
8984                14.4                17.0                 9.6
8985                 NaN                19.5                 7.9
8986                64.9                12.6                 8.7
8987                38.2                 9.6                 9.7
8988                29.6                22.4                11.8
8989                45.1                 NaN                 NaN
8990                69.2                 NaN                 NaN
8991                 NaN                12.8                15.4
8992                52.3                 8.9                 3.2
8993                24.7                16.2                14.3

[8994 rows x 21 columns]

In [123]:
```


Spyder (Python 3.6)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\anuas\OneDrive\Healthcare\Project2
Editor - C:\Users\anuas\OneDrive\Healthcare\Project2\cleaning2.py
sker.py line.py summary.py line1.py cleaning2.py untitled4.py column_rename.py
IPython console
Console 1/A
IPython console Help Variable explorer File explorer
Permissions: RW End-of-lines: CRLF Encoding: UTF-8 Line: 12 Column: 10 Memory: 68 %
Type here to search
9:59 PM
12/5/2017

```
# -*- coding: utf-8 -*-
"""
Created on Mon Dec  4 19:53:19 2017

@author: anuas
"""

import pandas as pd
df = pd.read_csv("agencies.csv",header = 0)
del df['readmitted_cases']
df["star_rating"].fillna(df.groupby("State")["star_rating"].transform("mean"), inplace=True)
print(df)
```

```
28            86.5            19.3            11.5
29            76.8            15.9            12.8
...            ...             ...             ...
8964          25.1            27.5            11.0
8965          53.8            18.9            12.2
8966          39.2             9.2            12.2
8967          44.7            21.3            13.3
8968          83.9            20.8            16.6
8969          33.9            20.2             7.7
8970          88.8            19.1            27.9
8971           9.0             8.9            28.0
8972          26.5            19.8            13.6
8973          46.8             NaN             NaN
8974          14.9             NaN             NaN
8975          65.6            12.4             7.1
8976          31.8             NaN             NaN
8977          86.8            14.2             8.0
8978          60.0            18.1            26.3
8979          73.9            13.0            12.7
8980          39.4            22.5            12.8
8981           6.8            15.1            24.7
8982          15.6            12.5            25.8
8983          44.4            27.3            11.2
8984          14.4            17.0             9.6
8985           NaN            19.5             7.9
8986          64.9            12.6             8.7
8987          38.2             9.6             9.7
8988          29.6            22.4            11.8
8989          45.1             NaN             NaN
8990          69.2             NaN             NaN
8991           NaN            12.8            15.4
8992          52.3             8.9             3.2
8993          24.7            16.2            14.3

[8994 rows x 21 columns]

In [125]:
```

Code Snippet:

```
# -*- coding: utf-8 -*-

"""

Created on Mon Dec  4 19:53:19 2017

@author: anuas

"""
```

```
import pandas as pd

df = pd.read_csv("agencies.csv",header = 0)

del df['readmitted_cases']

df["star_rating"].fillna(df.groupby("State")["star_rating"].transform("mean"), inplace=True)

print(df)
```

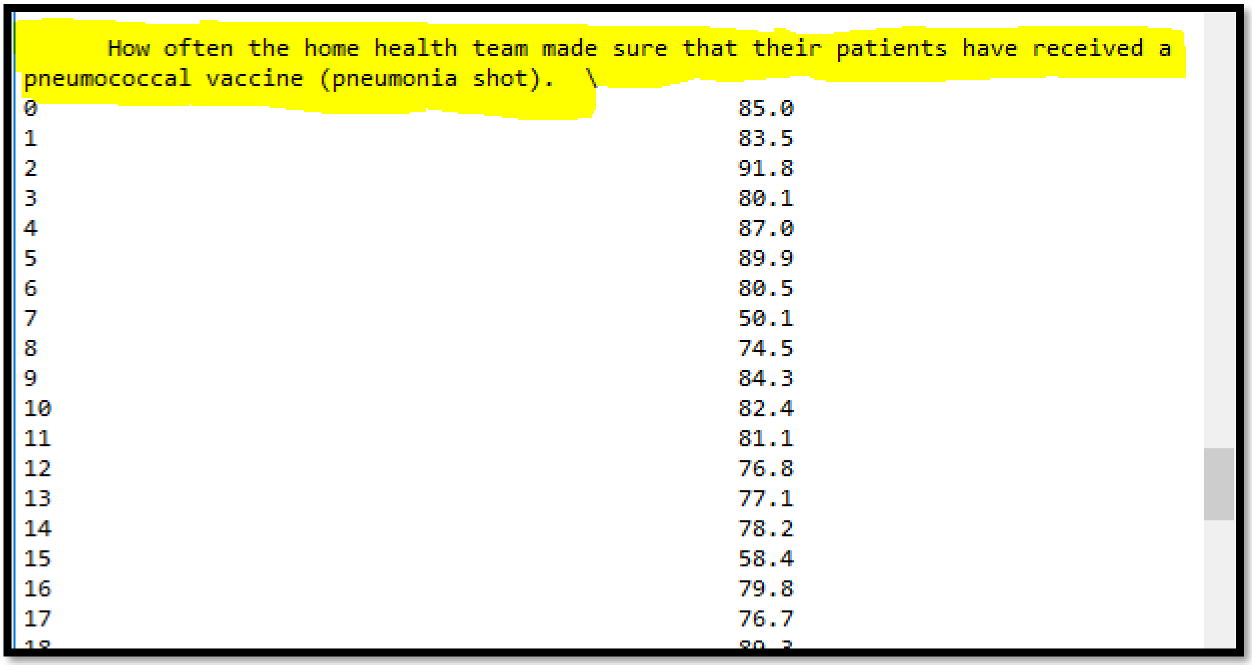
c) Inconvenient long column names

The column names in the dataset were in the form of long sentences which was very inconvenient to refer to. It was very cumbersome to view the data on the editor of anaconda spyder too as my dataset has too many columns. Therefore, I converted such long names into short meaningful names. This is for simple and convenient reference in the future. A screenshot of a few such changes are shown below. The **files** are read using read function and stored in **data frames.**

Python Concepts Used:

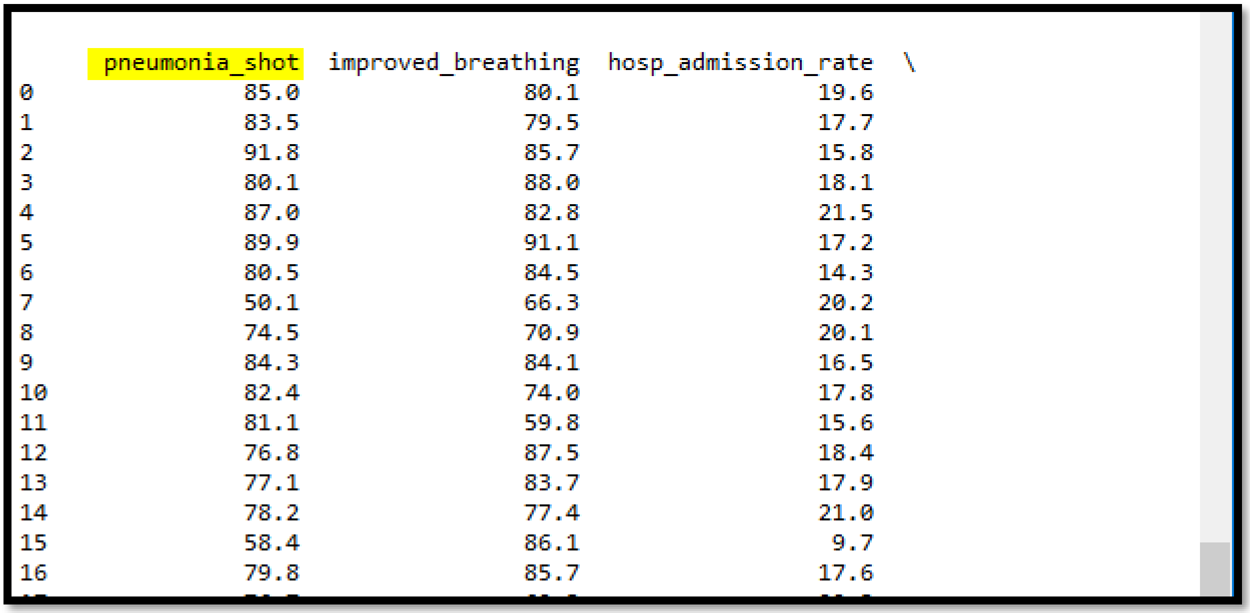
- Pandas Data Frame
- Files

Before

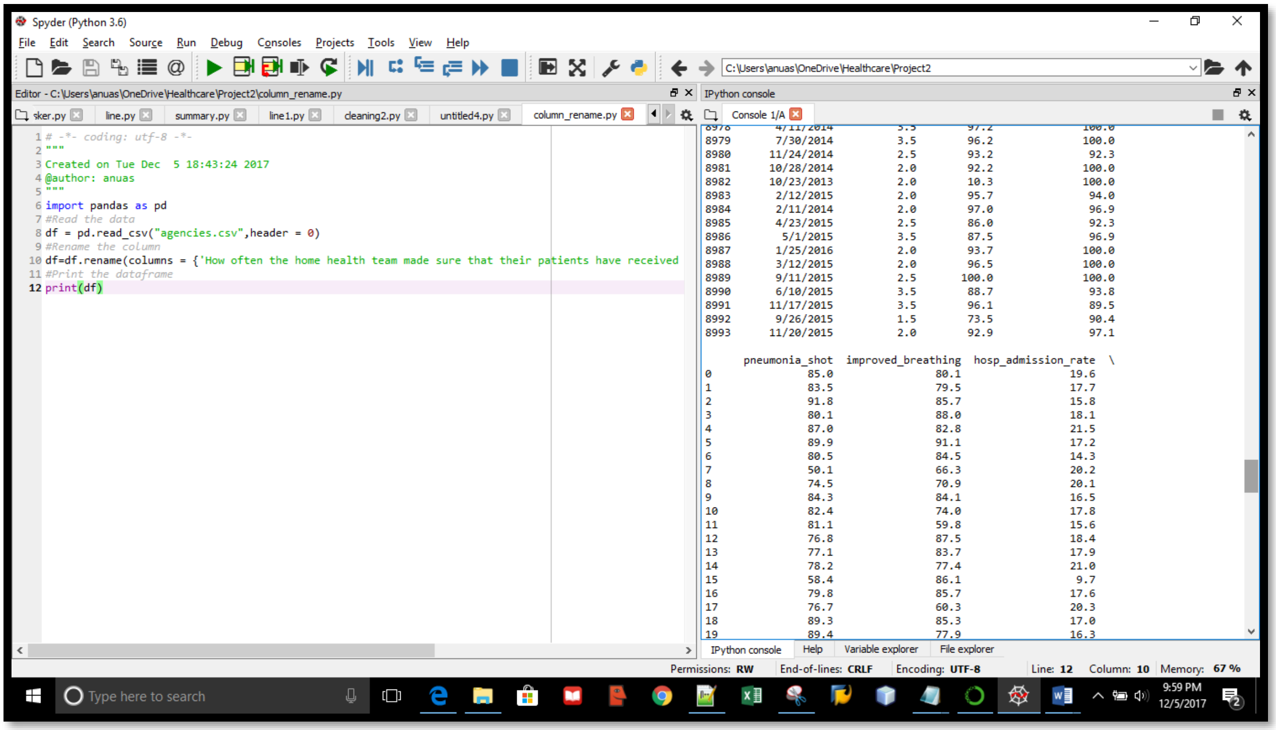


```
      How often the home health team made sure that their patients have received a 
pneumococcal vaccine (pneumonia shot).  \
0                                                  85.0
1                                                  83.5
2                                                  91.8
3                                                  80.1
4                                                  87.0
5                                                  89.9
6                                                  80.5
7                                                  50.1
8                                                  74.5
9                                                  84.3
10                                                 82.4
11                                                 81.1
12                                                 76.8
13                                                 77.1
14                                                 78.2
15                                                 58.4
16                                                 79.8
17                                                 76.7
```

After



```
     pneumonia_shot  improved_breathing  hosp_admission_rate  \
0              85.0                80.1                 19.6
1              83.5                79.5                 17.7
2              91.8                85.7                 15.8
3              80.1                88.0                 18.1
4              87.0                82.8                 21.5
5              89.9                91.1                 17.2
6              80.5                84.5                 14.3
7              50.1                66.3                 20.2
8              74.5                70.9                 20.1
9              84.3                84.1                 16.5
10             82.4                74.0                 17.8
11             81.1                59.8                 15.6
12             76.8                87.5                 18.4
13             77.1                83.7                 17.9
14             78.2                77.4                 21.0
15             58.4                86.1                  9.7
16             79.8                85.7                 17.6
```



```
# -*- coding: utf-8 -*-
"""
Created on Tue Dec  5 18:43:24 2017
@author: anuas
"""
import pandas as pd
#Read the data
df = pd.read_csv("agencies.csv",header = 0)
#Rename the column
df=df.rename(columns = {'How often the home health team made sure that their patients have received
#Print the dataframe
print(df)
```

```
8978     4/11/2014       3.5       97.2            100.0
8979     7/30/2014       3.5       96.2            100.0
8980    11/24/2014       2.5       93.2             92.3
8981    10/28/2014       2.0       92.2            100.0
8982    10/23/2013       2.0       10.3            100.0
8983     2/12/2015       2.0       95.7             94.0
8984     2/11/2014       2.0       97.0             96.9
8985     4/23/2015       2.5       86.0             92.3
8986      5/1/2015       3.5       87.5             96.9
8987     1/25/2016       2.0       93.7            100.0
8988     3/12/2015       2.0       96.5            100.0
8989     9/11/2015       2.5      100.0            100.0
8990     6/10/2015       3.5       88.7             93.8
8991    11/17/2015       3.5       96.1             89.5
8992     9/26/2015       1.5       73.5             90.4
8993    11/20/2015       2.0       92.9             97.1

      pneumonia_shot  improved_breathing  hosp_admission_rate  \
0               85.0                80.1                 19.6
1               83.5                79.5                 17.7
2               91.8                85.7                 15.8
3               80.1                88.0                 18.1
4               87.0                82.8                 21.5
5               89.9                91.1                 17.2
6               80.5                84.5                 14.3
7               50.1                66.3                 20.2
8               74.5                70.9                 20.1
9               84.3                84.1                 16.5
10              82.4                74.0                 17.8
11              81.1                59.8                 15.6
12              76.8                87.5                 18.4
13              77.1                83.7                 17.9
14              78.2                77.4                 21.0
15              58.4                86.1                  9.7
16              79.8                85.7                 17.6
17              76.7                60.3                 20.3
18              89.3                85.3                 17.0
19              89.4                77.9                 16.3
```

Code Snippet:

-*- coding: utf-8 -*-

"""

Created on Tue Dec 5 18:43:24 2017

@author: anuas

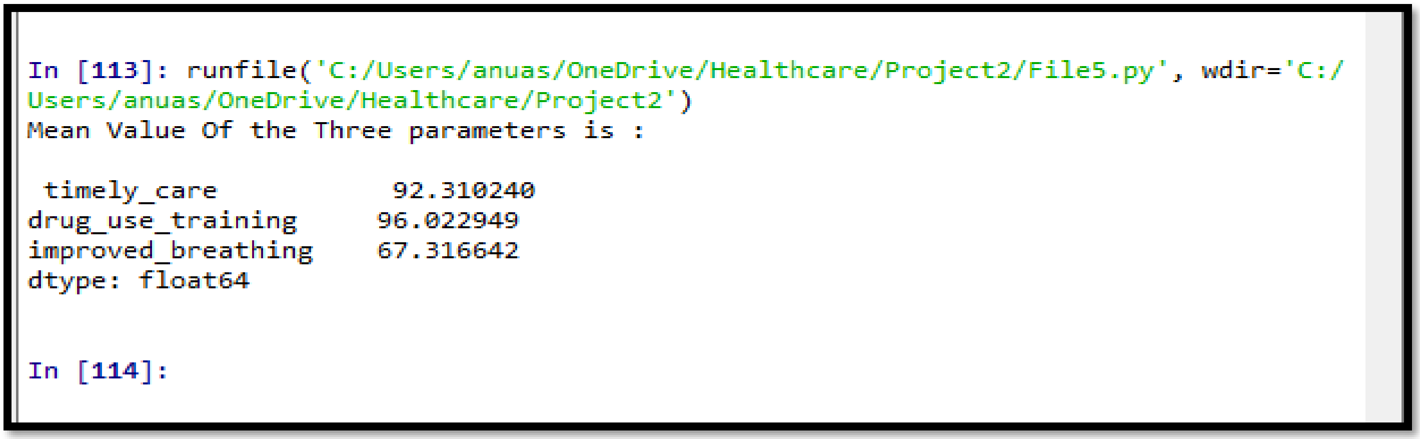
"""

```
import pandas as pd

#Read the data

df = pd.read_csv("agencies.csv",header = 0)

#Rename the column

df=df.rename(columns = {'How often the home health team made sure that their patients have received a pneumococcal vaccine (pneumonia shot).':'pneumonia_shot'})

#Print the dataframe

print(df)
```

C. Show/Apply Summary Statistics

1) Mean

The mean is the average of the numbers. To calculate it manually we add up all the numbers, then divide by how many numbers there are. In other words, it is the sum divided by the count. It is calculated using a function called **Mean ().**



```
In [113]: runfile('C:/Users/anuas/OneDrive/Healthcare/Project2/File5.py', wdir='C:/
Users/anuas/OneDrive/Healthcare/Project2')
Mean Value Of the Three parameters is :

 timely_care           92.310240
drug_use_training     96.022949
improved_breathing    67.316642
dtype: float64


In [114]:
```

Code Snippet:

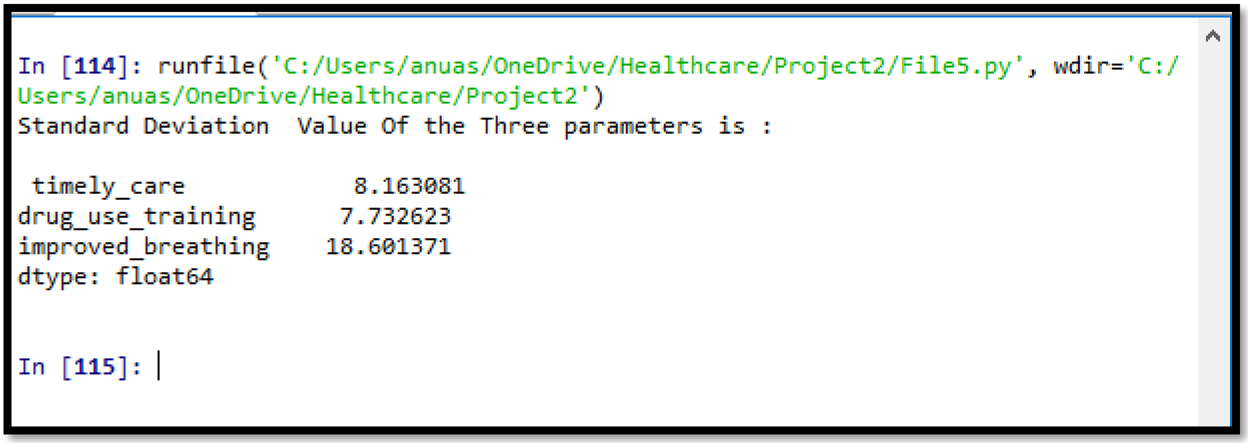
```
# -*- coding: utf-8 -*-

"""

Created on Sun Dec  3 21:37:09 2017



@author: anuas

"""



import pandas as pd

# Read data from CSV

df = pd.read_csv("agencies.csv")

#Select a few columns from the set of 21 columns

df1 = pd.DataFrame(df, columns = ['timely_care', 'drug_use_training',

'improved_breathing'])
```

#Mean statistical Calculation

print('Mean Value Of the Three parameters is :\n\n',df1.mean(),'\n')

2) Standard Deviation

It is a quantity calculated to indicate the extent of deviation for a group. It is calculated using a function called **std ().**



```
In [114]: runfile('C:/Users/anuas/OneDrive/Healthcare/Project2/File5.py', wdir='C:/
Users/anuas/OneDrive/Healthcare/Project2')
Standard Deviation  Value Of the Three parameters is :

 timely_care            8.163081
drug_use_training      7.732623
improved_breathing    18.601371
dtype: float64


In [115]: |
```

Code Snippet:

# -*- coding: utf-8 -*-

"""

Created on Sun Dec 3 21:37:09 2017

@author: anuas

"""

import pandas as pd

# Read data from CSV

df = pd.read_csv("agencies.csv")

#Select a few columns from the set of 21 columns df1 = pd.DataFrame(df, columns = ['timely_care', 'drug_use_training',

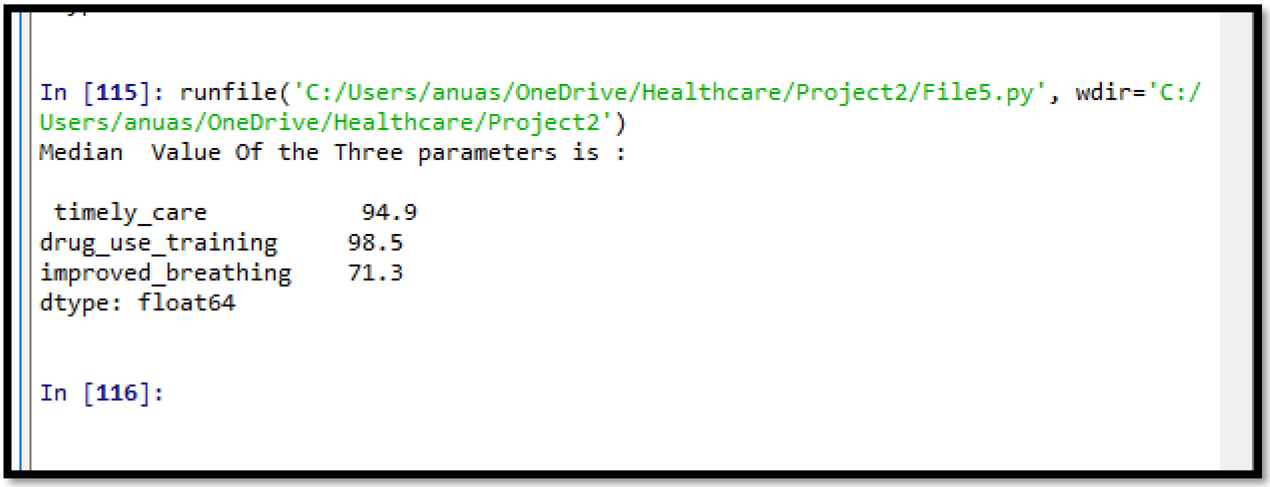
'improved_breathing'])

#Standard Deviation Statistical Calculation

print('Standard Deviation Value Of the Three parameters is :\n\n',df1.std(),'\n')

3) Median

The median value of a range of values is a value or quantity lying at the midpoint of a frequency distribution of observed values or quantities, such that there is an equal probability of falling above or below it. It is calculated using a function called **Median ().**



```
In [115]: runfile('C:/Users/anuas/OneDrive/Healthcare/Project2/File5.py', wdir='C:/
Users/anuas/OneDrive/Healthcare/Project2')
Median  Value Of the Three parameters is :

 timely_care           94.9
drug_use_training     98.5
improved_breathing    71.3
dtype: float64


In [116]:
```

Code Snippet:

# -*- coding: utf-8 -*-

"""

Created on Sun Dec 3 21:37:09 2017

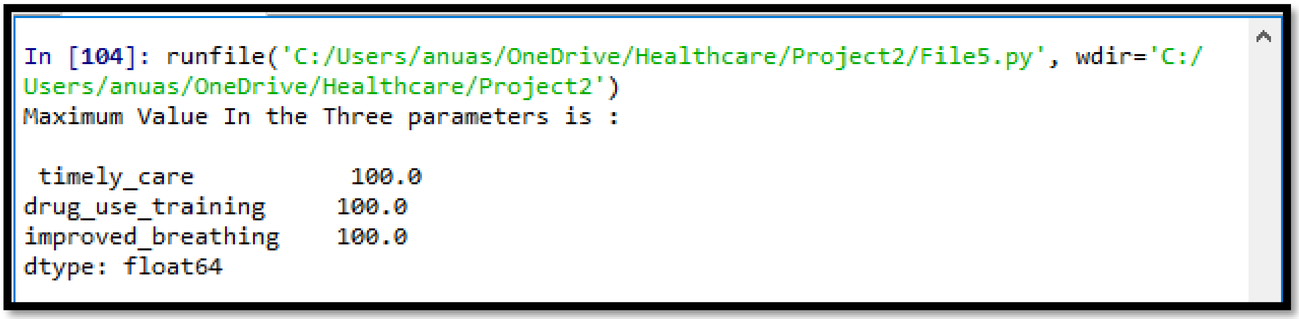
@author: anuas

"""

```
import pandas as pd

# Read data from CSV

df = pd.read_csv("agencies.csv")

#Select a few columns from the set of 21 columns

df1 = pd.DataFrame(df, columns = ['timely_care', 'drug_use_training',

'improved_breathing'])

#Median statistical Calculation

print('Median  Value Of the Three parameters is :\n\n',df1.median(),'\n')
```

3) Maximum

The Maximum value of a range of values is the highest value in the range of values. It is calculated using a function called **Max ().**



```
In [104]: runfile('C:/Users/anuas/OneDrive/Healthcare/Project2/File5.py', wdir='C:/
Users/anuas/OneDrive/Healthcare/Project2')
Maximum Value In the Three parameters is :

 timely_care           100.0
drug_use_training     100.0
improved_breathing    100.0
dtype: float64
```

Code Snippet:

```
# -*- coding: utf-8 -*-

"""

Created on Sun Dec  3 21:37:09 2017



@author: anuas

"""
```

import pandas as pd

# Read data from CSV

df = pd.read_csv("agencies.csv")

#Select a few columns from the set of 21 columns

df1 = pd.DataFrame(df, columns = ['timely_care', 'drug_use_training',

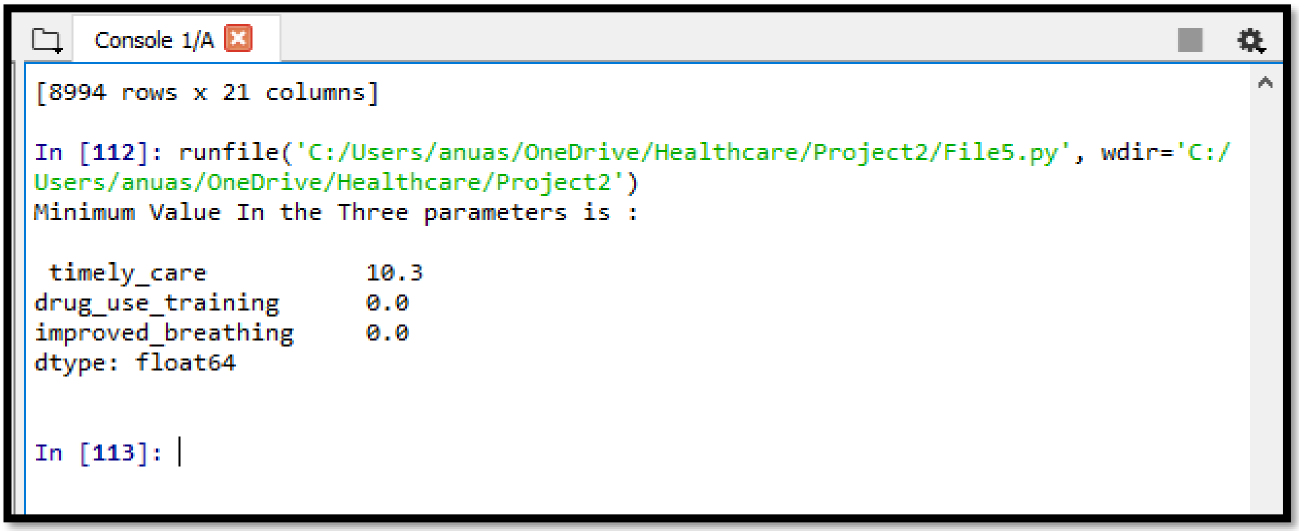
'improved_breathing'])

#Maximum statistical Calculation

print('Maximum Value In the Three parameters is :\n\n',df1.max(),'\n')

3) Minimum

The Minimum value of a range of values is the highest value in the range of values. It is calculated using a function called **Min ().**



```
[8994 rows x 21 columns]

In [112]: runfile('C:/Users/anuas/OneDrive/Healthcare/Project2/File5.py', wdir='C:/
Users/anuas/OneDrive/Healthcare/Project2')
Minimum Value In the Three parameters is :

 timely_care           10.3
drug_use_training      0.0
improved_breathing     0.0
dtype: float64


In [113]: |
```

Code Snippet :

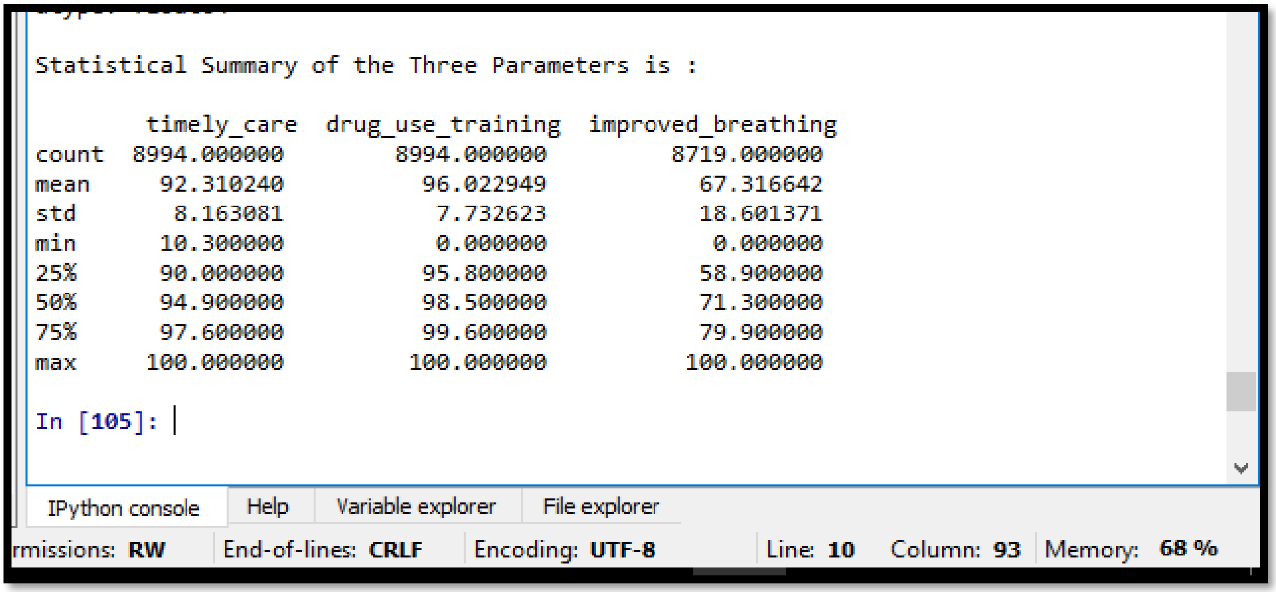
# -*- coding: utf-8 -*-

"""

```
Created on Sun Dec  3 21:37:09 2017

@author: anuas
"""

import pandas as pd
# Read data from CSV
df = pd.read_csv("agencies.csv")
#Select a few columns from the set of 21 columns
df1 = pd.DataFrame(df, columns = ['timely_care', 'drug_use_training',
'improved_breathing'])
#Minimum statistical Calculation
print('Minimum Value In the Three parameters is :\n\n',df1.min(),'\n')
```

4) Statistical Summary

This displays a set of values like Mean, Count, Standard Deviation, Minimum and Maximum values all at once for the selected set of columns. We use the **describe()** function.



```
Statistical Summary of the Three Parameters is :

        timely_care  drug_use_training  improved_breathing
count  8994.000000        8994.000000         8719.000000
mean     92.310240          96.022949           67.316642
std       8.163081           7.732623           18.601371
min      10.300000           0.000000            0.000000
25%      90.000000          95.800000           58.900000
50%      94.900000          98.500000           71.300000
75%      97.600000          99.600000           79.900000
max     100.000000         100.000000          100.000000

In [105]: 
```

Code Snippet:

# -*- coding: utf-8 -*-

"""

Created on Sun Dec 3 21:37:09 2017

@author: anuas

"""

import pandas as pd

# Read data from CSV

df = pd.read_csv("agencies.csv")

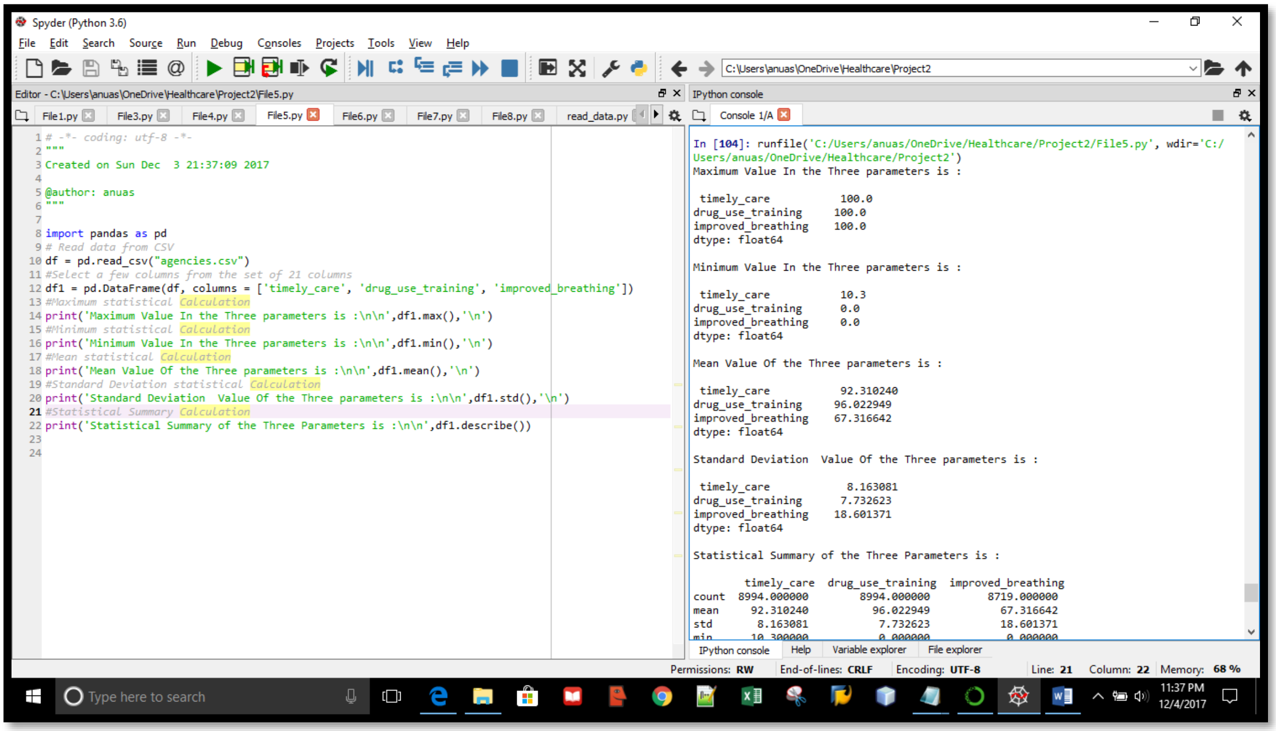
#Select a few columns from the set of 21 columns

df1 = pd.DataFrame(df, columns = ['timely_care', 'drug_use_training', 'improved_breathing'])

#Statistical Summary Calculation

print('Statistical Summary of the Three Parameters is :\n\n',df1.describe()

Screenshot of all the Statistical Functions at once:


Spyder (Python 3.6)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\anuas\OneDrive\Healthcare\Project2
Editor - C:\Users\anuas\OneDrive\Healthcare\Project2\File5.py
File1.py File3.py File4.py File5.py File6.py File7.py File8.py read_data.py
IPython console
Console 1/A
IPython console Help Variable explorer File explorer
Permissions: RW End-of-lines: CRLF Encoding: UTF-8 Line: 21 Column: 22 Memory: 68 %
Type here to search
11:37 PM
12/4/2017

```python
# -*- coding: utf-8 -*-
"""
Created on Sun Dec  3 21:37:09 2017

@author: anuas
"""

import pandas as pd
# Read data from CSV
df = pd.read_csv("agencies.csv")
#Select a few columns from the set of 21 columns
df1 = pd.DataFrame(df, columns = ['timely_care', 'drug_use_training', 'improved_breathing'])
#Maximum statistical Calculation
print('Maximum Value In the Three parameters is :\n\n',df1.max(),'\n')
#Minimum statistical Calculation
print('Minimum Value In the Three parameters is :\n\n',df1.min(),'\n')
#Mean statistical Calculation
print('Mean Value Of the Three parameters is :\n\n',df1.mean(),'\n')
#Standard Deviation statistical Calculation
print('Standard Deviation  Value Of the Three parameters is :\n\n',df1.std(),'\n')
#Statistical Summary Calculation
print('Statistical Summary of the Three Parameters is :\n\n',df1.describe())
```

```
In [104]: runfile('C:/Users/anuas/OneDrive/Healthcare/Project2/File5.py', wdir='C:/
Users/anuas/OneDrive/Healthcare/Project2')
Maximum Value In the Three parameters is :

 timely_care           100.0
drug_use_training     100.0
improved_breathing    100.0
dtype: float64

Minimum Value In the Three parameters is :

 timely_care           10.3
drug_use_training      0.0
improved_breathing     0.0
dtype: float64

Mean Value Of the Three parameters is :

 timely_care           92.310240
drug_use_training     96.022949
improved_breathing    67.316642
dtype: float64

Standard Deviation  Value Of the Three parameters is :

 timely_care            8.163081
drug_use_training      7.732623
improved_breathing    18.601371
dtype: float64

Statistical Summary of the Three Parameters is :

        timely_care  drug_use_training  improved_breathing
count  8994.000000        8994.000000         8719.000000
mean     92.310240          96.022949           67.316642
std       8.163081           7.732623           18.601371
min      10.300000           0.000000            0.000000
```

Code Snippet:

```
# -*- coding: utf-8 -*-

"""

Created on Sun Dec  3 21:37:09 2017

@author: anuas

"""

import pandas as pd

# Read data from CSV

df = pd.read_csv("agencies.csv")
```

```
#Select a few columns from the set of 21 columns

df1 = pd.DataFrame(df, columns = ['timely_care', 'drug_use_training', 'improved_breathing'])

#Maximum statistical Calculation

print('Maximum Value In the Three parameters is :\n\n',df1.max(),'\n')

#Minimum statistical Calculation

print('Minimum Value In the Three parameters is :\n\n',df1.min(),'\n')

#Mean statistical Calculation

print('Mean Value Of the Three parameters is :\n\n',df1.mean(),'\n')

#Median statistical Calculation

print('Median Value Of the Three parameters is :\n\n',df1.median(),'\n')

#Standard Deviation statistical Calculation

print('Standard Deviation  Value Of the Three parameters is :\n\n',df1.std(),'\n')

#Statistical Summary Calculation

print('Statistical Summary of the Three Parameters is :\n\n',df1.describe())
```

D. Analysis & Visualizations

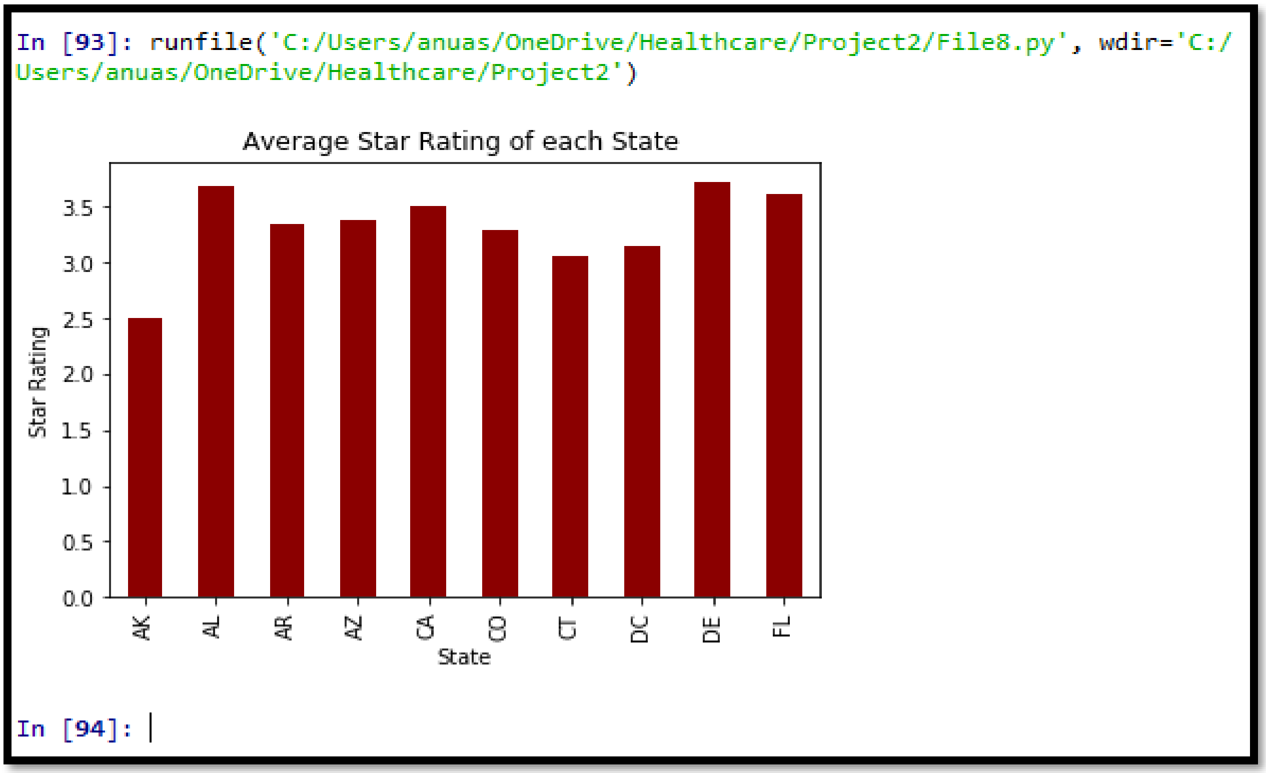
1) What is the average value of Quality of Patient Care Star Rating by State overall and which are the states with Quality of Patient Care Star Rating of 3.5 and above?

Fields used:

- Quality of Patient Care Star Rating
- State

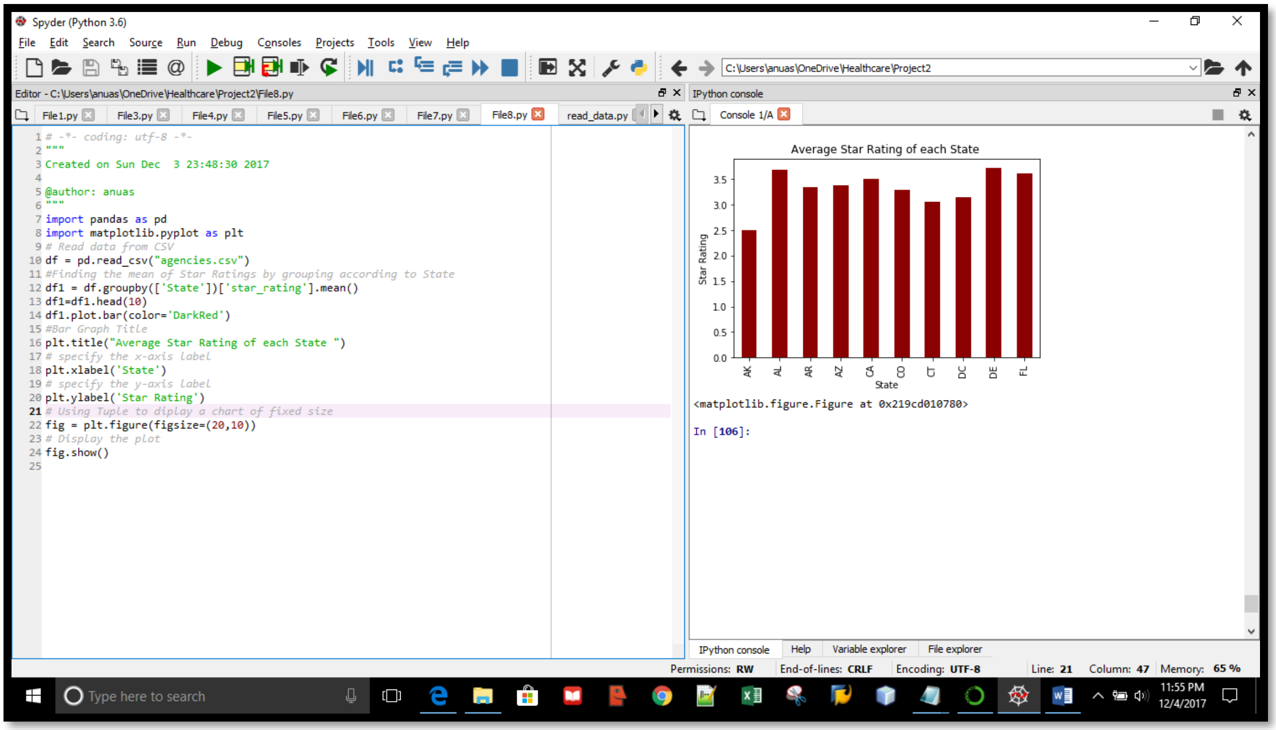
Python Concepts applied:

- Files
- Pandas Data Frame
- Tuple
- In-Built Functions
- Strings



```
In [93]: runfile('C:/Users/anuas/OneDrive/Healthcare/Project2/File8.py', wdir='C:/
Users/anuas/OneDrive/Healthcare/Project2')
```

```
In [94]:
```



```python
# -*- coding: utf-8 -*-
"""
Created on Sun Dec  3 23:48:30 2017

@author: anuas
"""
import pandas as pd
import matplotlib.pyplot as plt
# Read data from CSV
df = pd.read_csv("agencies.csv")
#Finding the mean of Star Ratings by grouping according to State
df1 = df.groupby(['State'])['star_rating'].mean()
df1=df1.head(10)
df1.plot.bar(color='DarkRed')
#Bar Graph Title
plt.title("Average Star Rating of each State ")
# specify the x-axis label
plt.xlabel('State')
# specify the y-axis label
plt.ylabel('Star Rating')
# Using Tuple to diplay a chart of fixed size
fig = plt.figure(figsize=(20,10))
# Display the plot
fig.show()
```

```
<matplotlib.figure.Figure at 0x219cd010780>

In [106]:
```

The above Bar graph shows the average star rating of every state in the USA. The average rating is calculated based on the star ratings of every home health agency in every state. The **files** are read using read function and stored in **data frames**. I have used the group by function to group the star rating according to the states. **Strings** are used to mention column names. It is found that the lowest average star rating is 2.5 and the highest average star rating is 3.94. One of the prominent fields that portray the effectiveness of the agency is the ‘Quality of Patient Care Star Rating ‘field. Home Health Compare uses a star rating between 1 and 5 to show people how a home health agency compares to other home health agencies on measurements of their performance. The star ratings are based on 9 measures of quality that give a general overview of performance. Across the country, most agencies fall “in the middle” with 3 or 3½ stars being the average rating across the 9 measures. A star rating higher than 3½ means that an agency performed better than average compared to other agencies. A star rating lower than 3 means that an agency’s performance was below average compared to other home health agencies. A **tuple** is defined to set the size of the graph. From the above visualization, we can infer that the states that have an average star rating of 3.5 and above are: -

- AL (Alaska)
- CA (California)
- DE (Denver)
- FL (Florida)

Among all the states, Alaska and Denver have the highest average star rating for their home agencies. This would help us determine the quality of the home health agencies in various states.

Code Snippet:

```
# -*- coding: utf-8 -*-
"""
Created on Sun Dec  3 23:48:30 2017
@author: anuas
"""

import pandas as pd
import matplotlib.pyplot as plt
# Read data from CSV
df = pd.read_csv("agencies.csv")
#Finding the mean of Star Ratings by grouping according to State
```

```
df1 = df.groupby(['State'])['star_rating'].mean()

df1=df1.head(10)

df1.plot.bar(color='DarkRed')

#Bar Graph Title

plt.title("Average Star Rating of each State ")

# specify the x-axis label

plt.xlabel('State')

# specify the y-axis label

plt.ylabel('Star Rating')

# Using Tuple to diplay a chart of fixed size

fig = plt.figure(figsize=(20,10))

# Display the plot

fig.show()
```

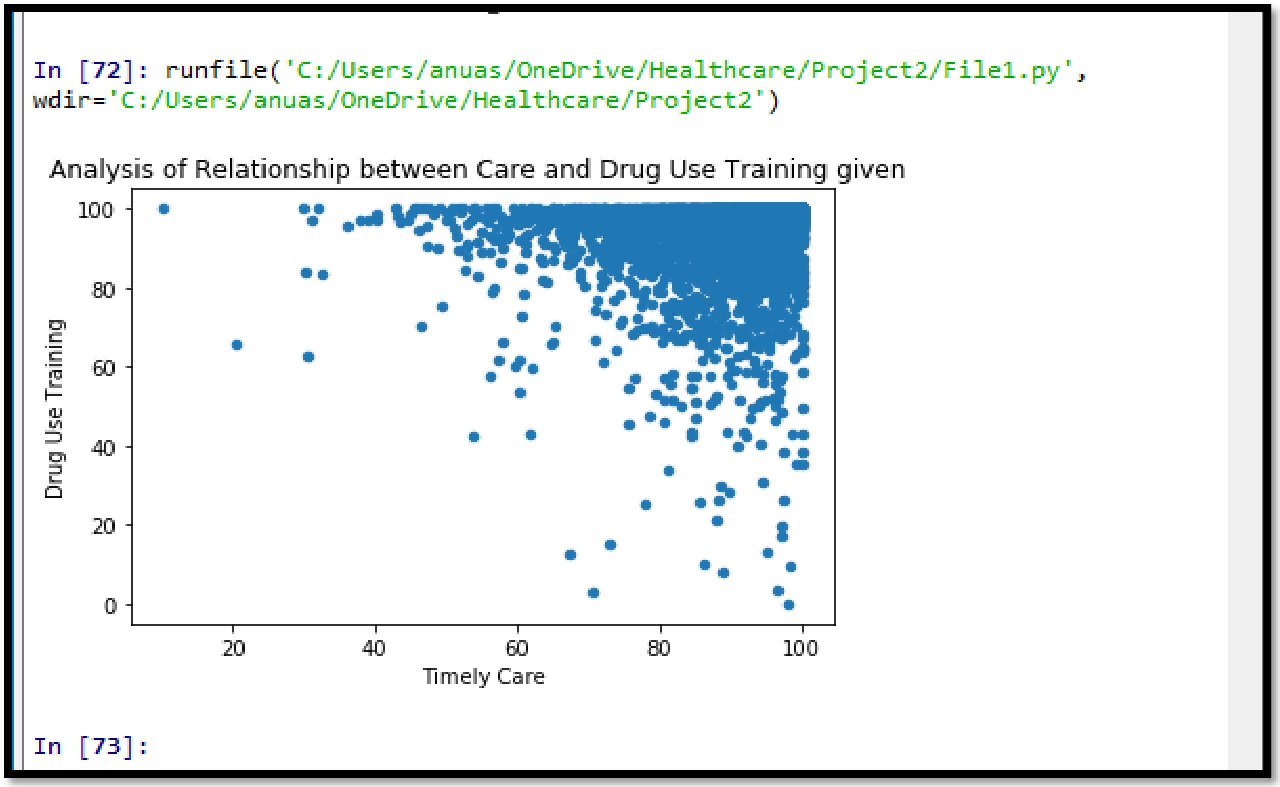
2) Analyze the relationship between the parameters timely care and the drug use training provided to patients by the home healthcare agencies?

Fields used:

- Drug Usage Training
- Timely manner care

Python Concepts applied:

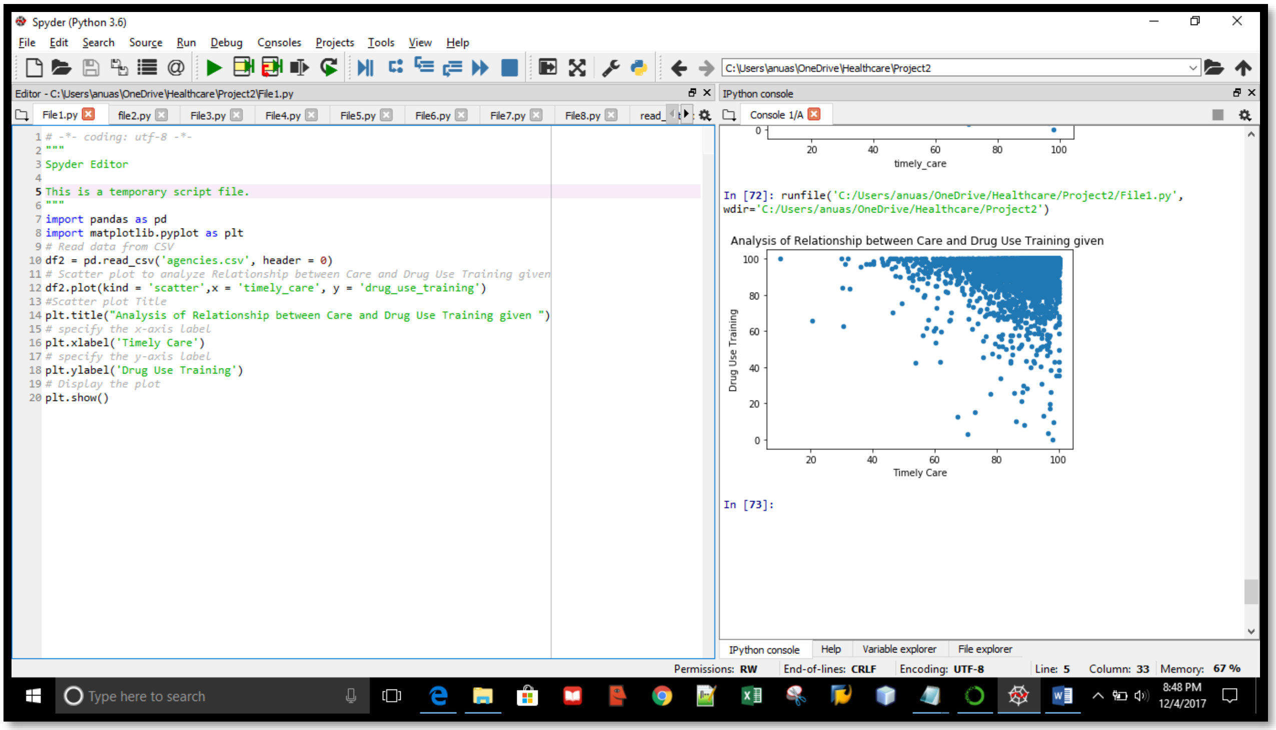
- Files
- Pandas Data Frame
- In-Built Functions



```
In [72]: runfile('C:/Users/anuas/OneDrive/Healthcare/Project2/File1.py',
wdir='C:/Users/anuas/OneDrive/Healthcare/Project2')
```

Analysis of Relationship between Care and Drug Use Training given
Drug Use Training
0
20
40
60
80
100
Timely Care
20
40
60
80
100

```
In [73]:
```


Spyder (Python 3.6)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\anuas\OneDrive\Healthcare\Project2
Editor - C:\Users\anuas\OneDrive\Healthcare\Project2\File1.py
File1.py file2.py File3.py File4.py File5.py File6.py File7.py File8.py read_
IPython console
Console 1/A
0
20
40
60
80
100
timely_care
Analysis of Relationship between Care and Drug Use Training given
100
80
60
40
20
0
Drug Use Training
20
40
60
80
100
Timely Care
IPython console Help Variable explorer File explorer
Permissions: RW End-of-lines: CRLF Encoding: UTF-8 Line: 5 Column: 33 Memory: 67 %
Type here to search
8:48 PM
12/4/2017

```
# -*- coding: utf-8 -*-
"""
Spyder Editor

This is a temporary script file.
"""
import pandas as pd
import matplotlib.pyplot as plt
# Read data from CSV
df2 = pd.read_csv('agencies.csv', header = 0)
# Scatter plot to analyze Relationship between Care and Drug Use Training given
df2.plot(kind = 'scatter',x = 'timely_care', y = 'drug_use_training')
#Scatter plot Title
plt.title("Analysis of Relationship between Care and Drug Use Training given ")
# specify the x-axis label
plt.xlabel('Timely Care')
# specify the y-axis label
plt.ylabel('Drug Use Training')
# Display the plot
plt.show()
```

```
In [72]: runfile('C:/Users/anuas/OneDrive/Healthcare/Project2/File1.py',
wdir='C:/Users/anuas/OneDrive/Healthcare/Project2')

In [73]:
```

This scatter plot depicts the relationship between how closely or in what pattern the parameters Timely Care and Drug Use Training are connected. Timely care is a column which has values in percentage about how care was provided in a timely manner for the patients by the home health care agencies. Drug usage training is a column which has values in percentage about the drug related lessons and guidelines given to the patients by corresponding home healthcare agencies. Both these fields are co-related as the drug usage training is also a type of care to be given and the time at which the usage of drugs is taught would make a direct impact on the health of the patients involved. The **files** are read using read function and stored in **data frames.** In the graph, we can see that almost all the agencies have a good percentage of the two parameters, that is, Timely Care and Drug Use Training. This means that the home health care agencies have done a good job in providing timely care to the patients and have provided good training before hand to use the drugs in an appropriate way. We can also infer that the two fields are well related. Therefore, changes done in one of these two fields would directly have its effect on the other.

Code Snippet:

```
# -*- coding: utf-8 -*-

"""

Created on Sun Dec  3 21:06:48 2017

@author: anuas

"""

import pandas as pd

import matplotlib.pyplot as plt

# Read data from CSV

df2 = pd.read_csv('agencies.csv', header = 0)

# Scatter plot to analyze Relationship between Care and Drug Use Training given

df2.plot(kind = 'scatter',x = 'timely_care', y = 'drug_use_training')
```

```
#Scatter plot Title

plt.title("Analysis of Relationship between Care and Drug Use Training given ")

# specify the x-axis label

plt.xlabel('Timely Care')

# specify the y-axis label

plt.ylabel('Drug Use Training')

# Display the plot

plt.show()
```

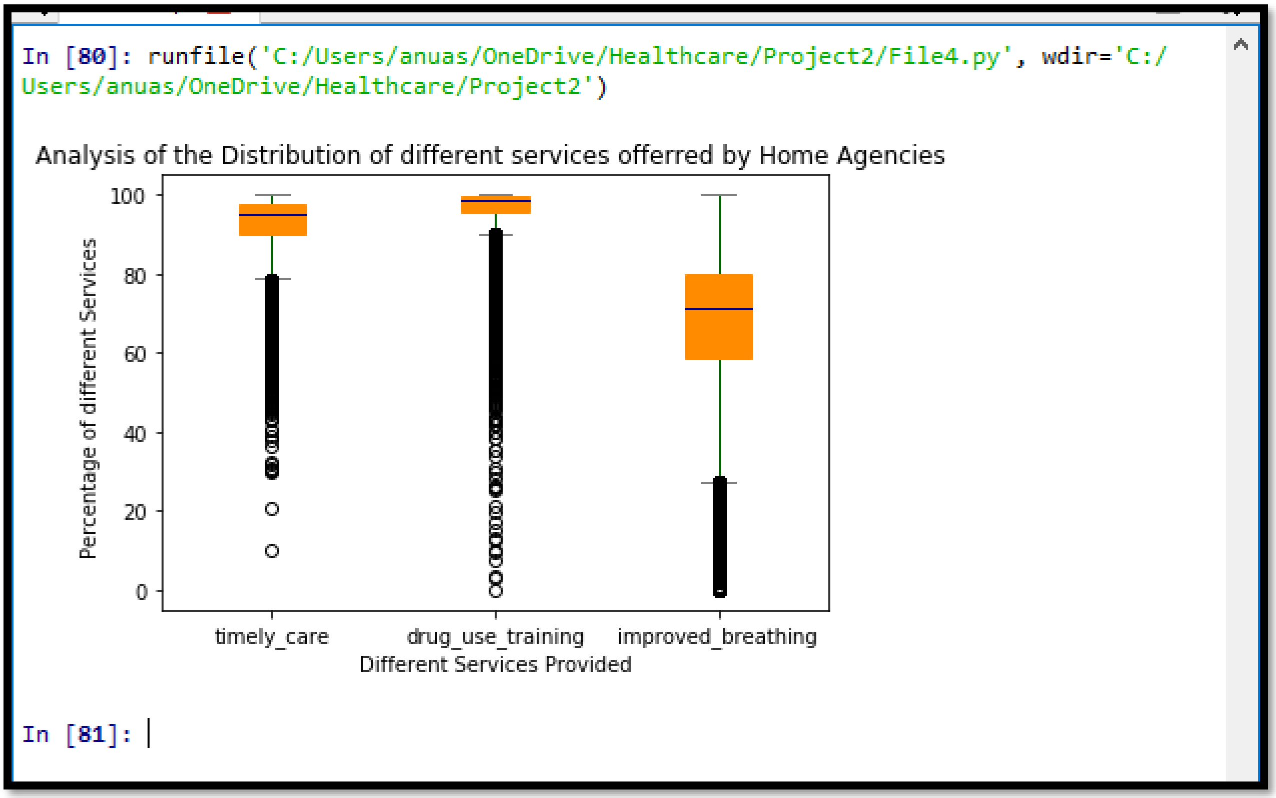
3) Compare the quality of services provided by home health care agencies of USA by considering a few services like timely care, improved breathing and Drug usage lessons given to patients?

Fields used:

- Drug Usage Training
- Timely manner care
- Improved breathing

Python Concepts applied:

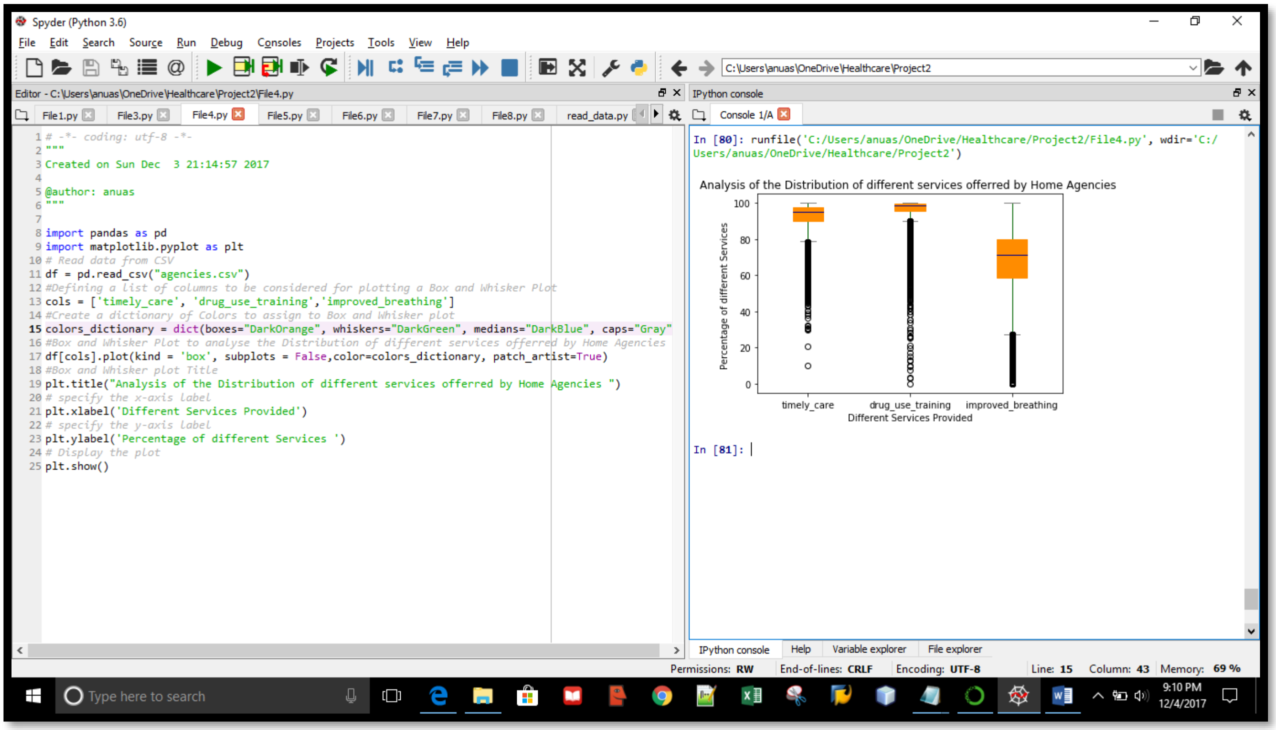
- Files
- Pandas Data Frame
- Dictionary
- Lists
- Strings
- In-Built Functions



```
In [80]: runfile('C:/Users/anuas/OneDrive/Healthcare/Project2/File4.py', wdir='C:/
Users/anuas/OneDrive/Healthcare/Project2')
```

Analysis of the Distribution of different services offerred by Home Agencies
Percentage of different Services
100
80
60
40
20
0
timely_care
drug_use_training
improved_breathing
Different Services Provided

```
In [81]:
```



Editor - C:\Users\anuas\OneDrive\Healthcare\Project2\File4.py

```python
# -*- coding: utf-8 -*-
"""
Created on Sun Dec  3 21:14:57 2017

@author: anuas
"""

import pandas as pd
import matplotlib.pyplot as plt
# Read data from CSV
df = pd.read_csv("agencies.csv")
#Defining a list of columns to be considered for plotting a Box and Whisker Plot
cols = ['timely_care', 'drug_use_training','improved_breathing']
#Create a dictionary of Colors to assign to Box and Whisker plot
colors_dictionary = dict(boxes="DarkOrange", whiskers="DarkGreen", medians="DarkBlue", caps="Gray"
#Box and Whisker Plot to analyse the Distribution of different services offerred by Home Agencies
df[cols].plot(kind = 'box', subplots = False,color=colors_dictionary, patch_artist=True)
#Box and Whisker plot Title
plt.title("Analysis of the Distribution of different services offerred by Home Agencies ")
# specify the x-axis label
plt.xlabel('Different Services Provided')
# specify the y-axis label
plt.ylabel('Percentage of different Services ')
# Display the plot
plt.show()
```

```
In [80]: runfile('C:/Users/anuas/OneDrive/Healthcare/Project2/File4.py', wdir='C:/
Users/anuas/OneDrive/Healthcare/Project2')
```

Spyder (Python 3.6)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\anuas\OneDrive\Healthcare\Project2
File1.py File3.py File4.py File5.py File6.py File7.py File8.py read_data.py
IPython console
Console 1/A
Analysis of the Distribution of different services offerred by Home Agencies
Percentage of different Services
0
20
40
60
80
100
timely_care
drug_use_training
improved_breathing
Different Services Provided
In [81]:
IPython console Help Variable explorer File explorer
Permissions: RW End-of-lines: CRLF Encoding: UTF-8 Line: 15 Column: 43 Memory: 69 %
Type here to search
9:10 PM
12/4/2017

The above Box and Whisker plot shows the percentage effectiveness of the various services provided by the health care agencies like improved breathing, Drug usage lessons given to patient and the timely care provided to various patients on a month wise scale for the most recent year of 2016. The **files** are read using read function and stored in **data frames. Strings** are used to mention column names. A **list** is used to mention a set of columns to be considered for analysis. A **dictionary** is used here to assign different colors to different elements of the box and whisker plot. We can infer that the service that was used the most was the Timely Care Provided to the patients by the home healthcare agencies. This service was asked for the most and it also shows a good percentage of response provided by the health care agencies. Second comes the Drug usage lessons which is also one of the most important services that tell us about the effectiveness of each agency. The box and whisker plot mainly tells us under which region we have the values that are most common. In this case it tells us the percentage values that are more common. This shows the areas on which the home health agencies should concentrate and improve themselves to provide good services to the public. In the above graph, we can see that most of the home health care agencies have a good percentage of 85-100 % in providing timely care to the patients in need and the Drug usage lessons have most of their percentage range at 90-100 % and the service improved breathing has its values in the range of 60-80 %. These numbers mean that the home health care agencies are providing good services.

Code Snippet:

```
# -*- coding: utf-8 -*-
"""
Created on Sun Dec  3 21:14:57 2017
@author: anuas
"""
import pandas as pd
import matplotlib.pyplot as plt
# Read data from CSV
df = pd.read_csv("agencies.csv")
#Defining a list of columns to be considered for plotting a Box and Whisker Plot
cols = ['timely_care', 'drug_use_training','improved_breathing']
#Create a dictionary of Colors to assign to Box and Whisker plot
```

```
colors_dictionary = dict(boxes="Dark", whiskers="DarkGreen", medians="DarkBlue", caps="Gray")

#Box and Whisker Plot to analyse the Distribution of different services offerred by Home Agencies

df[cols].plot(kind = 'box', subplots = False,color=colors_dictionary, patch_artist=True)

#Box and Whisker plot Title

plt.title("Analysis of the Distribution of different services offerred by Home Agencies ")

# specify the x-axis label

plt.xlabel('Different Services Provided')

# specify the y-axis label

plt.ylabel('Percentage of different Services ')

# Display the plot

plt.show()
```

A List of all the Codes Together:

Data Cleaning

a) Missing Values replaced with mean value

Code Snippet:

```
# -*- coding: utf-8 -*-

"""

Created on Mon Dec  4 19:30:57 2017

@author: anuas

"""

import pandas as pd

# Read data from CSV

df = pd.read_csv("agencies.csv",header = 0)

#Delete the unwanted row

del df['readmitted_cases']

#Fill in the blank values with the Mean value of star rating filtered according to states

Str1="star_rating"

df[str1].fillna(df[str1].mean(), inplace=True)
```

```
#Print Data Frame

print(df)
```

b) Deleting unwanted columns

Code Snippet:

```
# -*- coding: utf-8 -*-

"""

Created on Mon Dec  4 19:53:19 2017

@author: anuas

"""

import pandas as pd

df = pd.read_csv("agencies.csv",header = 0)

del df['readmitted_cases']

df["star_rating"].fillna(df.groupby("State")["star_rating"].transform("mean"), inplace=True)
```

```
print(df)
```

c) Inconvenient long column names

Code Snippet:

```
# -*- coding: utf-8 -*-

"""

Created on Tue Dec  5 18:43:24 2017

@author: anuas

"""

import pandas as pd

#Read the data

df = pd.read_csv("agencies.csv",header = 0)

#Rename the column
```

```
df=df.rename(columns = {'How often the home health team made sure that their patients have received a pneumococcal vaccine (pneumonia shot).':'pneumonia_shot'})

#Print the dataframe

print(df)
```

Show/Apply Summary Statistics

Code Snippet:

```
# -*- coding: utf-8 -*-
"""
Created on Sun Dec  3 21:37:09 2017

@author: anuas
"""

import pandas as pd
# Read data from CSV
df = pd.read_csv("agencies.csv")
#Select a few columns from the set of 21 columns
df1 = pd.DataFrame(df, columns = ['timely_care', 'drug_use_training', 'improved_breathing'])
#Maximum statistical Calculation
print('Maximum Value In the Three parameters is :\n\n',df1.max(),'\n')
#Minimum statistical Calculation
```

```
print('Minimum Value In the Three parameters is :\n\n',df1.min(),'\n')
#Mean statistical Calculation
print('Mean Value Of the Three parameters is :\n\n',df1.mean(),'\n')
#Median statistical Calculation
print('Median Value Of the Three parameters is :\n\n',df1.median(),'\n')
#Standard Deviation statistical Calculation
print('Standard Deviation  Value Of the Three parameters is :\n\n',df1.std(),'\n')
#Statistical Summary Calculation
print('Statistical Summary of the Three Parameters is :\n\n',df1.describe())
```

Analysis & Visualizations

1) **What is the average value of Quality of Patient Care Star Rating by State overall and which are the states with Quality of Patient Care Star Rating of 3.5 and above?**

Code Snippet:

```
# -*- coding: utf-8 -*-

"""

Created on Sun Dec  3 23:48:30 2017

@author: anuas

"""

import pandas as pd

import matplotlib.pyplot as plt
```

```
# Read data from CSV

df = pd.read_csv("agencies.csv")

#Finding the mean of Star Ratings by grouping according to State

df1 = df.groupby(['State'])['star_rating'].mean()

df1=df1.head(10)

df1.plot.bar(color='DarkRed')

#Bar Graph Title

plt.title("Average Star Rating of each State ")

# specify the x-axis label

plt.xlabel('State')

# specify the y-axis label

plt.ylabel('Star Rating')

# Using Tuple to diplay a chart of fixed size

fig = plt.figure(figsize=(20,10))

# Display the plot

fig.show()
```

2) **Analyze the relationship between the parameters timely care and the drug use training provided to patients by the home healthcare agencies?**

Code Snippet:

```
# -*- coding: utf-8 -*-

"""

Created on Sun Dec  3 21:06:48 2017

@author: anuas

"""

import pandas as pd

import matplotlib.pyplot as plt

# Read data from CSV

df2 = pd.read_csv('agencies.csv', header = 0)

# Scatter plot to analyze Relationship between Care and Drug Use Training given

df2.plot(kind = 'scatter',x = 'timely_care', y = 'drug_use_training')

#Scatter plot Title
```

```
plt.title("Analysis of Relationship between Care and Drug Use Training given ")

# specify the x-axis label

plt.xlabel('Timely Care')

# specify the y-axis label

plt.ylabel('Drug Use Training')

# Display the plot

plt.show()
```

3) **Compare the quality of services provided by home health care agencies of USA by considering a few services like timely care, improved breathing and Drug usage lessons given to patients?**

Code Snippet:

```
# -*- coding: utf-8 -*-

"""

Created on Sun Dec  3 21:14:57 2017
```

```
@author: anuas

"""

import pandas as pd

import matplotlib.pyplot as plt

# Read data from CSV

df = pd.read_csv("agencies.csv")

#Defining a list of columns to be considered for plotting a Box and Whisker Plot

cols = ['timely_care', 'drug_use_training','improved_breathing']

#Create a dictionary of Colors to assign to Box and Whisker plot

colors_dictionary = dict(boxes="Dark", whiskers="DarkGreen", medians="DarkBlue",
caps="Gray")

#Box and Whisker Plot to analyse the Distribution of different services offerred by Home
Agencies

df[cols].plot(kind = 'box', subplots = False,color=colors_dictionary, patch_artist=True)
```

```
#Box and Whisker plot Title
plt.title("Analysis of the Distribution of different services offerred by Home Agencies ")
# specify the x-axis label
plt.xlabel('Different Services Provided')
# specify the y-axis label
plt.ylabel('Percentage of different Services ')
# Display the plot
plt.show()
```