Name:Atta Ullah Khan
ASUID:1217207992

Requirements:
1) python3.6
2)pip3

Install the following libraries before running this project, open Terminal

```
pip3 install numpy
pip3 install pandas
pip3 install seaborn
pip3 install sklearn
```

Tasks:

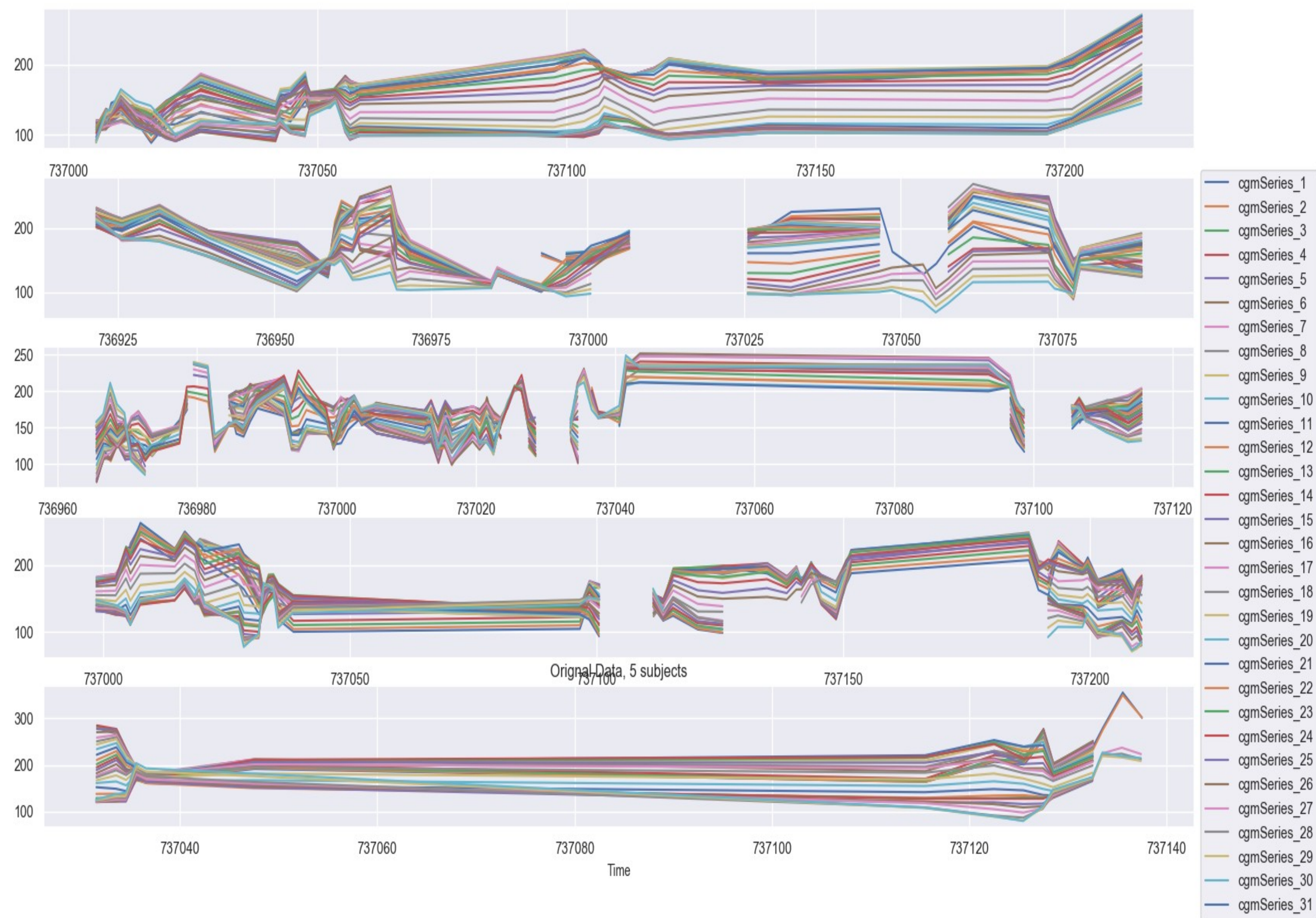
a) Extract 4 different types of time series features from only the CGM data cell array and CGM timestamp cell array (10 points each) total 40
b) For each time series explain why you chose such feature (5 points each) total 20

Answer to Both Part(a) and Part(b):

Let first Visualize the all the 5 sample with all the 31 cgm serieses ploted against its timestamp

```
python3  assignment1.py -m orignal
```

Output will be:



Initially to extract 4 different type of features, we will use four methods
1:mean 2:Std(Standard deviation) 3: Max(Minimum) 4:Min(Minimum)

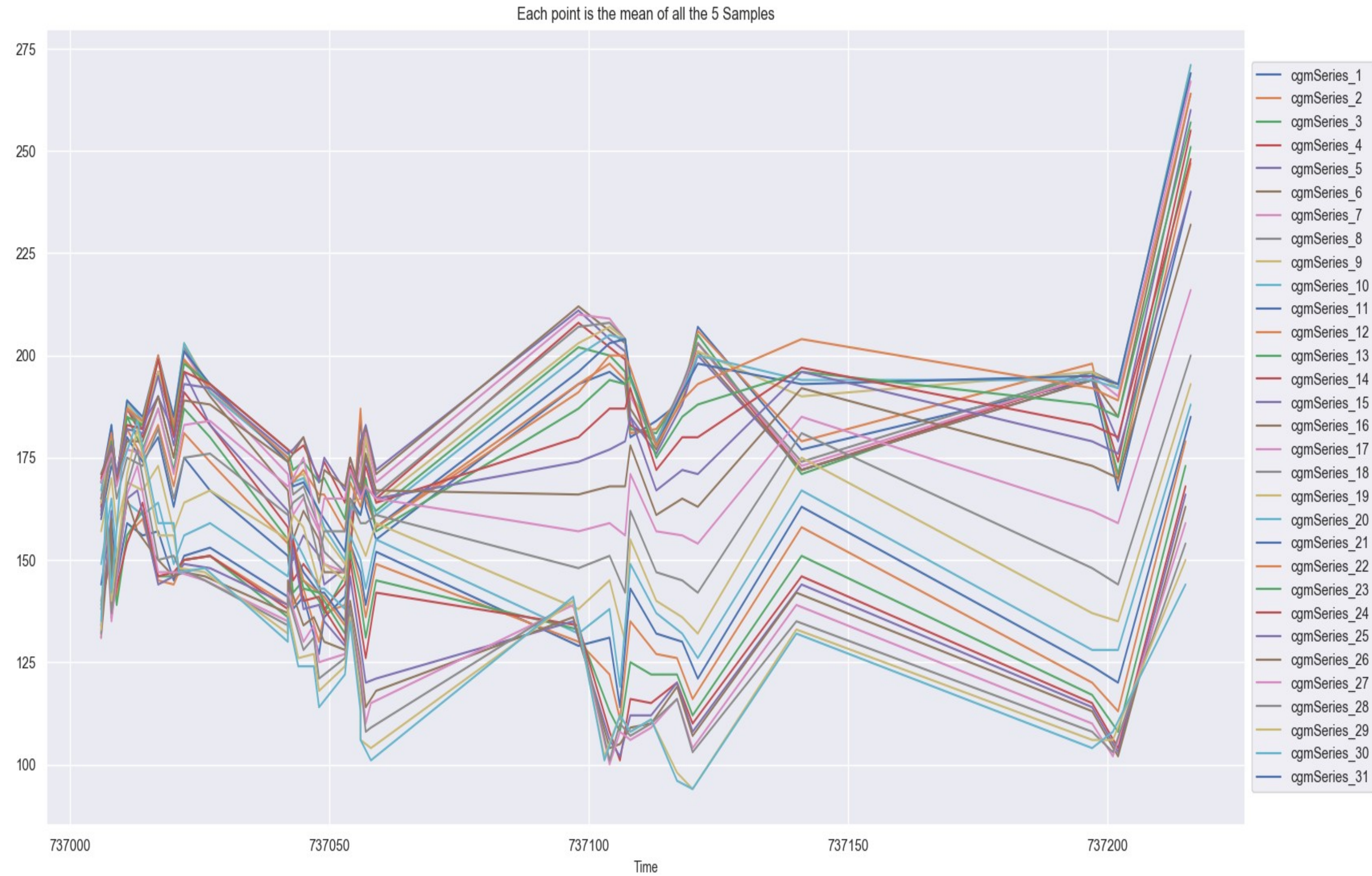
Algorithm 1:
input = method, data ##possible values of method =[min, max, mean, std]
0
For each cgmSeries in {cgmseris1, cgmseries2…..cgmseriesN},
list = combine cgmseries of this type from all the 5 subjects
excecute the method(Mean/std/min/max) on list
create a new data frame for this method
return this new data frame;

this algorithm return that mean/std/min/max of all the 5 subjects

lets take the mean of all the 5 subject and plot it, run the following

```
python3  assignment1.py -m mean -p plot
```

output wil be
Data plot for mean



This is the mean of all th 5 subjects

Now for mean lets extract the important features using corelation

Algorithm2, for extracting features :
method=input
data= call Algorthm1(method)
CorMatrix=find the correaltion matrix
selectedVariable =Select 1 variable from {cgmSeries}
independentVaribles={cgmSeries1,cgmseries2,…...N}-selectedVaribles

PotentialFeturesList=from the corMatrix Select all the variables such that for each variable V , CorMtrix[v,selectedVariable]>0.5

independentVaribles=PotentialFeturesList-selectedVaribles

CorMatrixIndependentV=Find correlation amngst independentVaribles

for each of the varables V1,V2 in independentVaribles:
    if CorMatrixIndependentV[V1,V2]>.9
        Remove V1 From the PotentialFeturesList

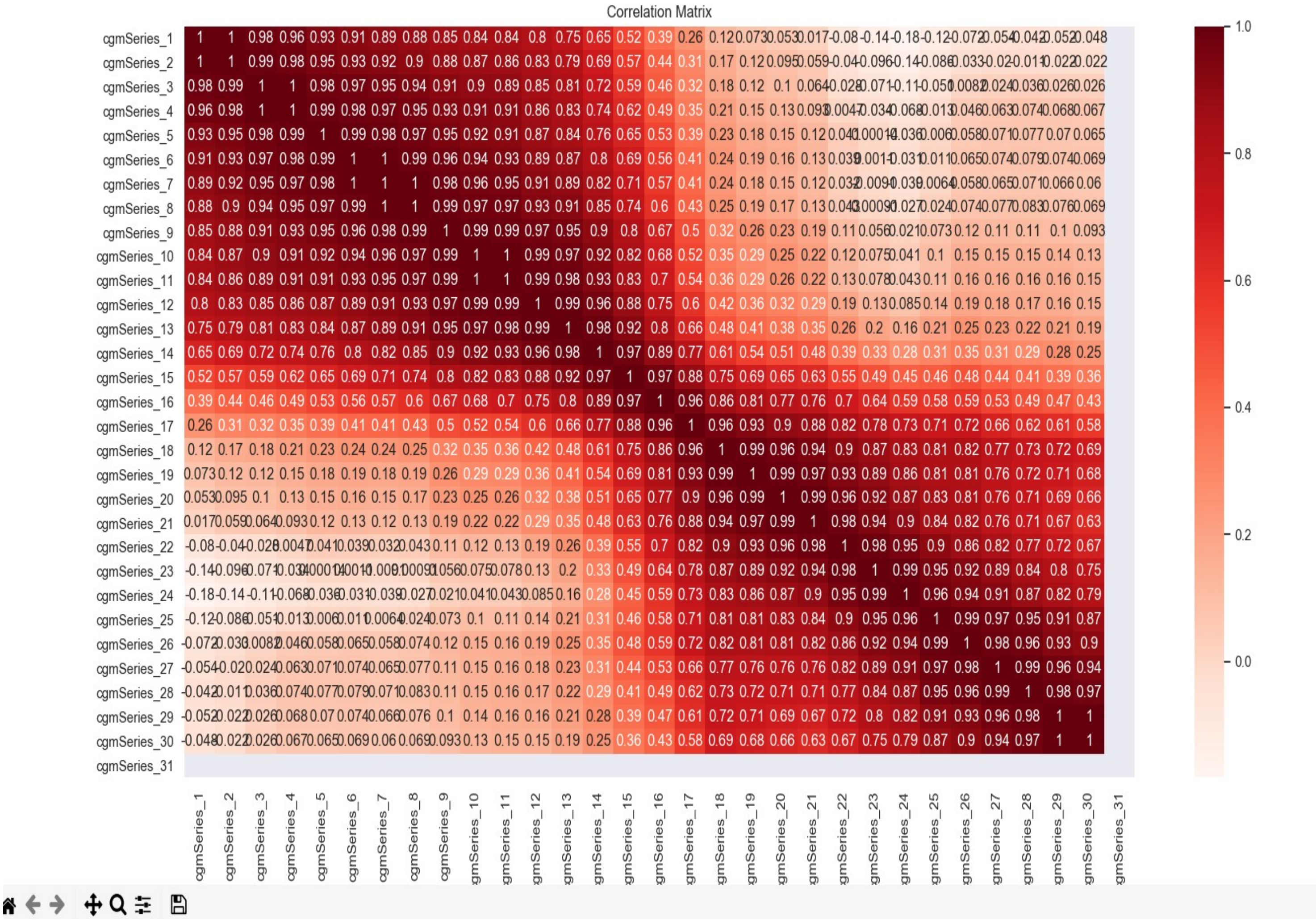
return potentialFeatures;

In this algorthm we are slecting all the variable that are co related to the dependent variable, and droping all the independent varables which are co related

Lets extract the features from the Mean Data

```
python3  assignment1.py -m mean -f extract -e head
```

Cor relation matrix for mean:



Output:

```
 feature related to:cgmSeries_1
```

Index(['cgmSeries_1', 'cgmSeries_2', 'cgmSeries_3', 'cgmSeries_4',
'cgmSeries_5', 'cgmSeries_6', 'cgmSeries_7', 'cgmSeries_8',
'cgmSeries_9', 'cgmSeries_10', 'cgmSeries_11', 'cgmSeries_12',
'cgmSeries_13', 'cgmSeries_14', 'cgmSeries_15'],
dtype='object')

Independable variable thae are have strong correllation, Droping:{'cgmSeries_4', ' 'cgmSeries_3', 'cgmSeries_11', 'cgmSeries_6', 'cgmSeries_5', 'cgmSeries_13', ' 'cgmSeries_12', 'cgmSeries_8', 'cgmSeries_7'}

```
Using mean Selected Features are :['cgmSeries_9', 'cgmSeries_2', 'cgmSe
'cgmSeries_1']
```

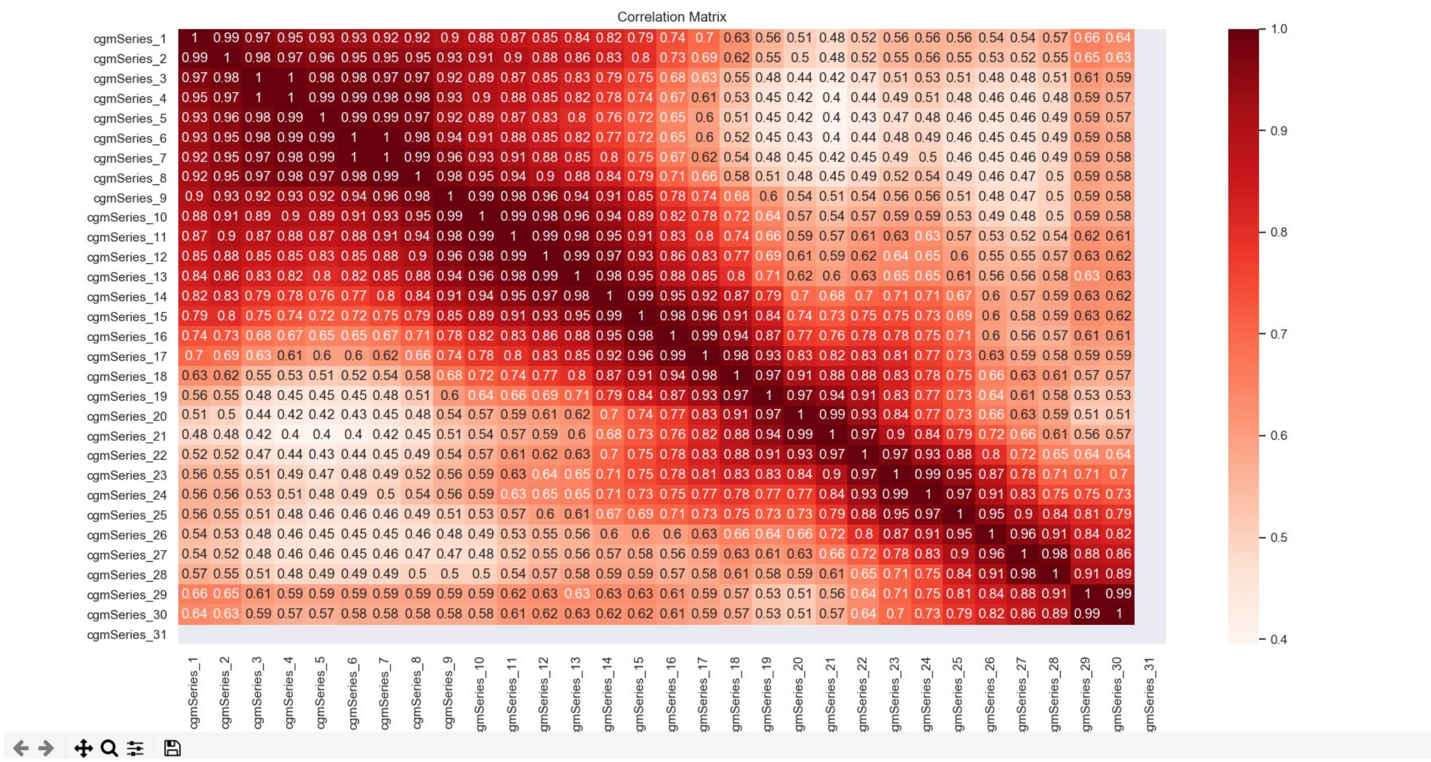
So our selected Features from Mean

featuer1(mean)=['cgmSeries_9', 'cgmSeries_2', 'cgmSeries_15', 'cgmSeries_1'] , for 2nd type of feature lets run(standard Deviation)

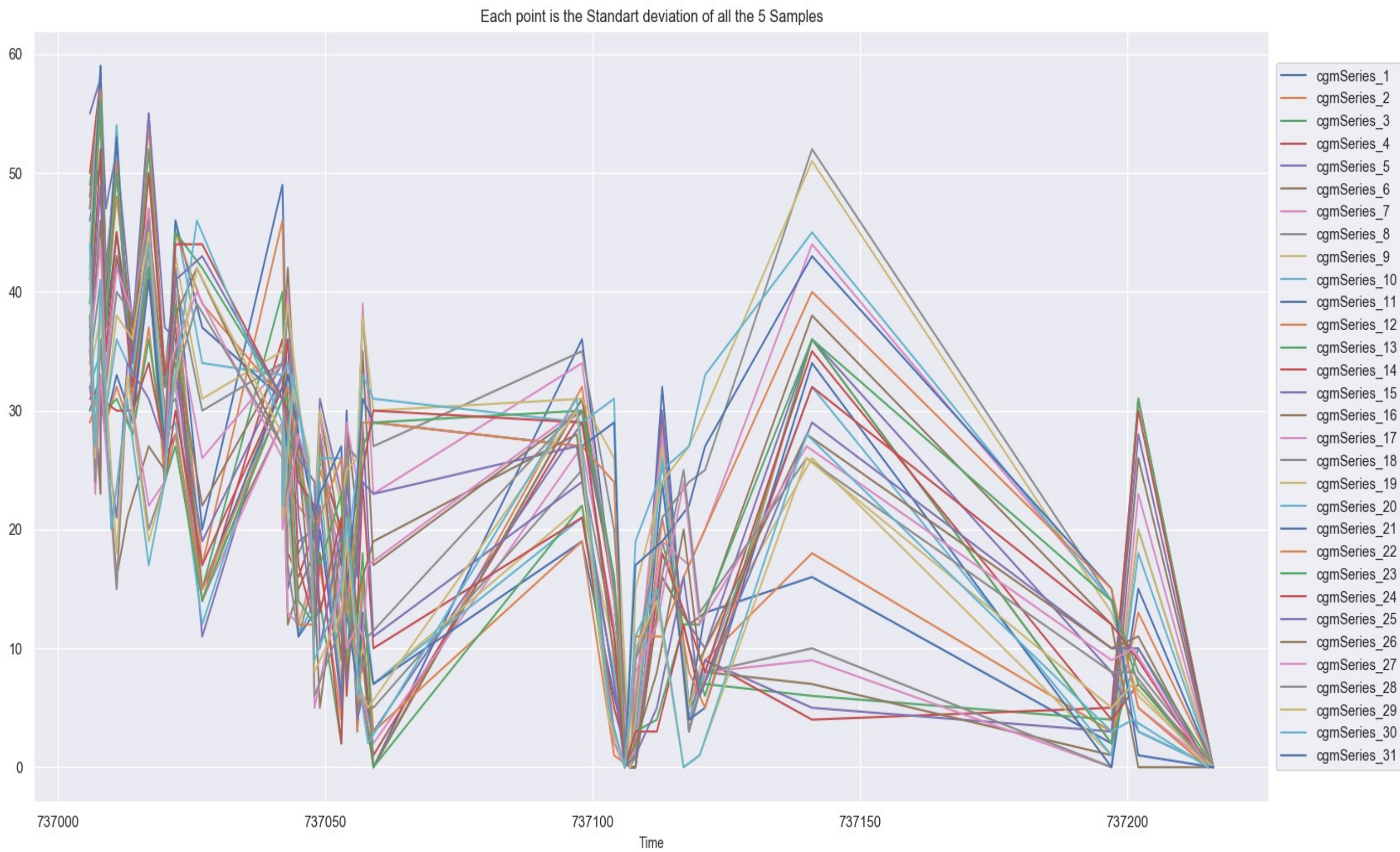
```
python3  assignment1.py -m std -f extract -p plot -e head
```

Output

Cor relation matrix for std



Data plot of std




```
    dtype='object')
Independable variable thae are have strong correllation, Droping:{'cgmSeries_9', 'cgmSeries_22',
'cgmSeries_13', 'cgmSeries_25', 'cgmSeries_24', 'cgmSeries_7', 'cgmSeries_27', 'cgmSeries_3',
'cgmSeries_20', 'cgmSeries_28', 'cgmSeries_6', 'cgmSeries_17', 'cgmSeries_18', 'cgmSeries_4',
'cgmSeries_30', 'cgmSeries_12', 'cgmSeries_10', 'cgmSeries_14', 'cgmSeries_5', 'cgmSeries_15',
'cgmSeries_8'}
Using  Selected Features are :['cgmSeries_2', 'cgmSeries_1', 'cgmSeries_19', 'cgmSeries_29',
'cgmSeries_16', 'cgmSeries_26', 'cgmSeries_23', 'cgmSeries_11'] ,
```

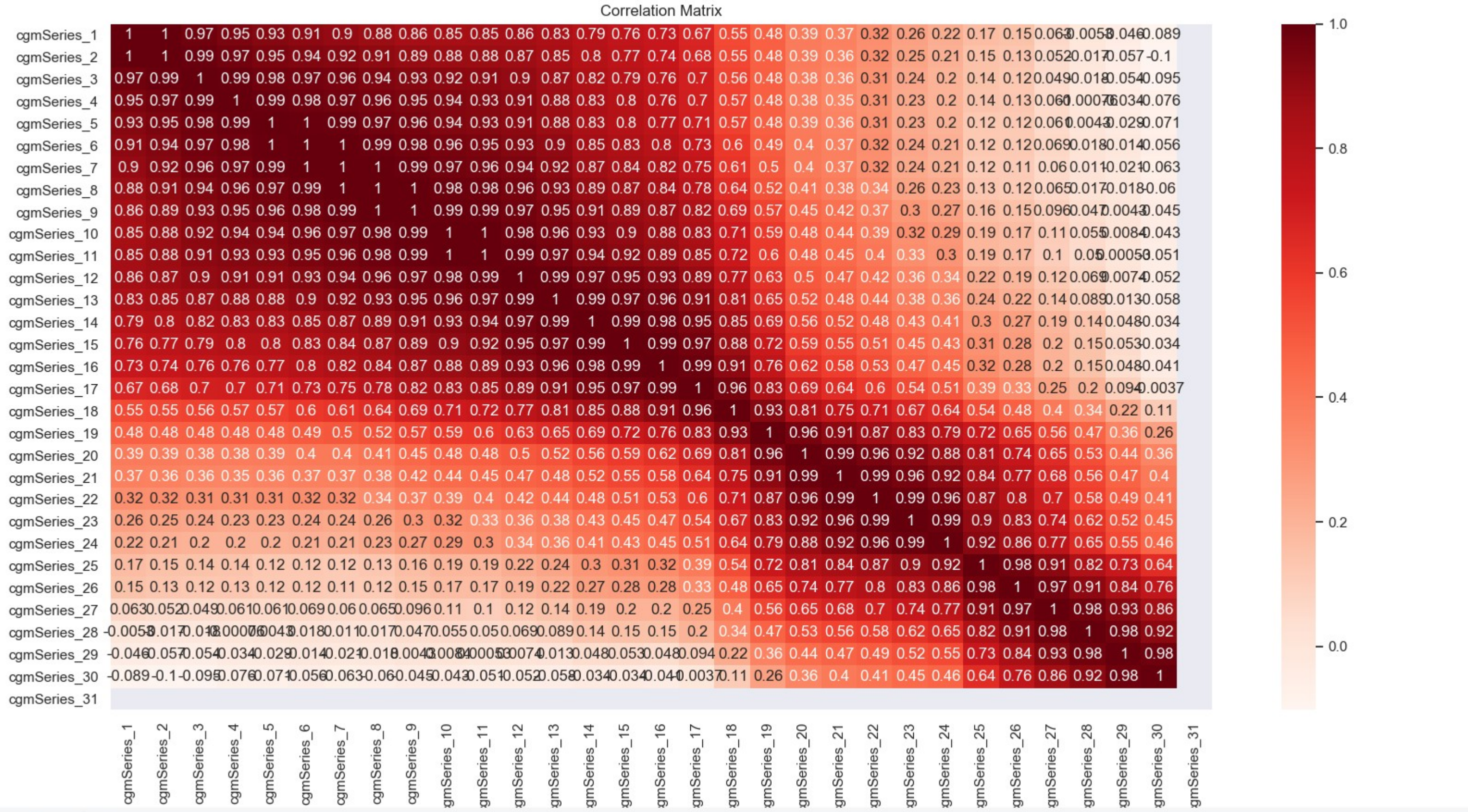
features2(STD)=['cgmSeries_2', 'cgmSeries_1', 'cgmSeries_19', 'cgmSeries_29', 'cgmSeries_16', 'cgmSeries_26', 'cgmSeries_23', 'cgmSeries_11']

for 3rd type of features lets run(Minimum)

```
python3  assignment1.py -m min -f extract -p plot -e head
```

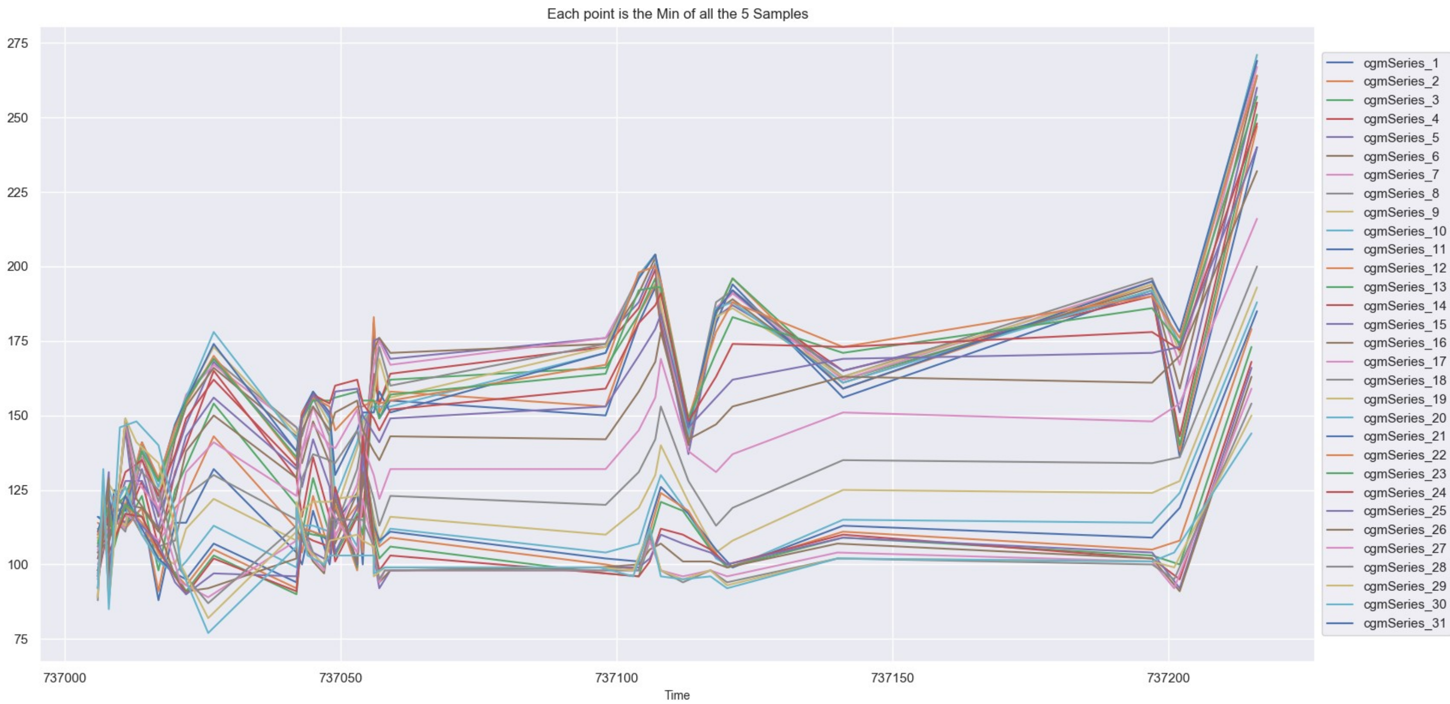
Output

Cor relation matrix for minimum



,msa

Data plot of min



```
Potential relevent feature related to:cgmSeries_1
Index(['cgmSeries_1', 'cgmSeries_2', 'cgmSeries_3', 'cgmSeries_4',
       'cgmSeries_5', 'cgmSeries_6', 'cgmSeries_7', 'cgmSeries_8',
       'cgmSeries_9', 'cgmSeries_10', 'cgmSeries_11', 'cgmSeries_12',
       'cgmSeries_13', 'cgmSeries_14', 'cgmSeries_15', 'cgmSeries_16',
       'cgmSeries_17', 'cgmSeries_18'],
      dtype='object')
Independable variable thae are have strong correllation, Droping:{'cgmS
'cgmSeries_8', 'cgmSeries_6', 'cgmSeries_16', 'cgmSeries_13', 'cgmSerie
'cgmSeries_4', 'cgmSeries_10', 'cgmSeries_14', 'cgmSeries_11', 'cgmSeri
'cgmSeries_17', 'cgmSeries_12'}
Using  Selected Features are :['cgmSeries_2', 'cgmSeries_15', 'cgmSerie
'cgmSeries_9', 'cgmSeries_18']
```

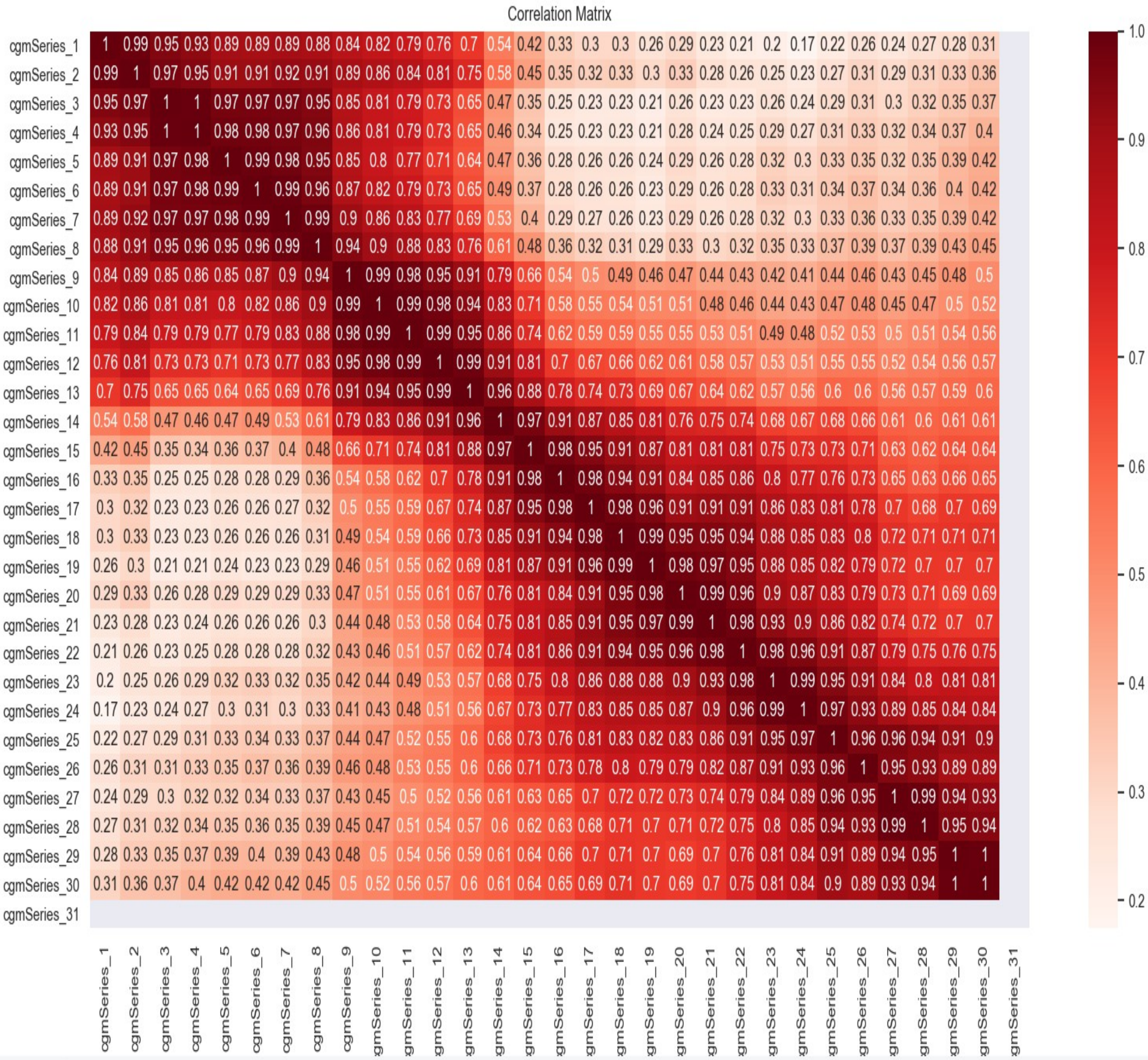
features3(Minimum)=['cgmSeries_2', 'cgmSeries_15', 'cgmSeries_1', 'cgmSeries_9', 'cgmSeries_18']

for 4th type of features lets run(Mximum)

```
python3  assignment1.py -m max      -f extract -p plot -e head
```

Output

Cor relation matrix for max:



Data plot of max


```
Index(['cgmSeries_1', 'cgmSeries_2', 'cgmSeries_3', 'cgmSeries_4',
       'cgmSeries_5', 'cgmSeries_6', 'cgmSeries_7', 'cgmSeries_8',
       'cgmSeries_9', 'cgmSeries_10', 'cgmSeries_11', 'cgmSeries_12',
       'cgmSeries_13', 'cgmSeries_14'],
      dtype='object')
Independable variable thae are have strong correllation, Droping:{'cgmS
'cgmSeries_12', 'cgmSeries_8', 'cgmSeries_10', 'cgmSeries_13', 'cgmSeri
'cgmSeries_7', 'cgmSeries_3', 'cgmSeries_5', 'cgmSeries_4'}
Using  Selected Features are :['cgmSeries_1', 'cgmSeries_2', 'cgmSeries
'cgmSeries_9']
```

features4(Maximum)=['cgmSeries_1', 'cgmSeries_2', 'cgmSeries_14', 'cgmSeries_9']

Selected features using mean/std/min/max are following

featuer1(mean)=['cgmSeries_9', 'cgmSeries_2', 'cgmSeries_15', 'cgmSeries_1'] ,
features2(STD)=['cgmSeries_2', 'cgmSeries_1', 'cgmSeries_19', 'cgmSeries_29', 'cgmSeries_16', 'cgmSeries_26', 'cgmSeries_23', 'cgmSeries_11']
features3(Minimum)=['cgmSeries_2', 'cgmSeries_15', 'cgmSeries_1', 'cgmSeries_9', 'cgmSeries_18']
features4(Maximum)=['cgmSeries_1', 'cgmSeries_2', 'cgmSeries_14', 'cgmSeries_9']

c) Show values of each of the features and argue that your intuition in step b is validated or disproved? (5 points each) total 20
Output for Feature1(Mean)

```
cgmSeries_15,cgmSeries_2,cgmSeries_9,cgmSeries_1
,,,
,,,
240.0,247.0,271.0,240.0
176.0,169.0,192.0,167.0
179.0,198.0,196.0,194.0
196.0,179.0,190.0,177.0
171.0,206.0,201.0,207.0
172.0,191.0,190.0,189.0
167.0,182.0,178.0,184.0
185.0,181.0,191.0,180.0
179.0,194.0,204.0,193.0
177.0,198.0,207.0,196.0
174.0,193.0,203.0,193.0
165.0,158.0,164.0,155.0
175.0,165.0,180.0,163.0
166.0,187.0,165.0,185.0
175.0,151.0,164.0,151.0
167.0,139.0,149.0,141.0
175.0,137.0,156.0,136.0
169.0,130.0,161.0,127.0
180.0,143.0,171.0,141.0
177.0,139.0,170.0,138.0
176.0,154.0,175.0,151.0
192.0,174.0,192.0,167.0
193.0,181.0,203.0,175.0
178.0,168.0,185.0,163.0
195.0,183.0,196.0,180.0
181.0,175.0,183.0,174.0
182.0,182.0,188.0,180.0
171.0,169.0,169.0,170.0
178.0,176.0,182.0,173.0
171.0,161.0,164.0,160.0
```

```
Output for Feature2(Std)

cgmSeries_2,cgmSeries_1,cgmSeries_19,cgmSeries_29,cgmSeries_16,cgmSeries_26,cgmSeries_23,cgm
Series_11
```

```
,,,,,,,
,,,,,,,
0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0
31.0,31.0,6.0,7.0,0.0,11.0,7.0,15.0
3.0,2.0,13.0,5.0,12.0,10.0,14.0,0.0
18.0,16.0,51.0,26.0,38.0,28.0,36.0,34.0
9.0,13.0,30.0,1.0,13.0,9.0,12.0,5.0
5.0,5.0,27.0,0.0,12.0,20.0,12.0,4.0
29.0,32.0,24.0,14.0,16.0,8.0,4.0,24.0
2.0,0.0,15.0,10.0,0.0,2.0,3.0,0.0
0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0
14.0,14.0,26.0,4.0,9.0,3.0,16.0,6.0
32.0,36.0,31.0,30.0,30.0,28.0,30.0,19.0
0.0,0.0,30.0,5.0,17.0,19.0,29.0,7.0
11.0,12.0,38.0,7.0,35.0,16.0,27.0,13.0
3.0,3.0,26.0,10.0,21.0,14.0,21.0,7.0
28.0,30.0,25.0,19.0,29.0,8.0,9.0,13.0
19.0,17.0,22.0,13.0,9.0,19.0,23.0,6.0
25.0,26.0,30.0,8.0,28.0,5.0,17.0,20.0
20.0,19.0,20.0,26.0,11.0,25.0,20.0,13.0
22.0,25.0,31.0,32.0,26.0,27.0,25.0,11.0
24.0,28.0,39.0,21.0,42.0,22.0,30.0,26.0
46.0,49.0,35.0,28.0,34.0,27.0,31.0,32.0
17.0,20.0,31.0,42.0,22.0,42.0,42.0,15.0
40.0,42.0,43.0,30.0,35.0,38.0,45.0,28.0
33.0,33.0,29.0,24.0,27.0,25.0,27.0,26.0
54.0,55.0,45.0,19.0,47.0,27.0,36.0,42.0
36.0,35.0,36.0,33.0,37.0,21.0,28.0,30.0
48.0,45.0,38.0,18.0,43.0,16.0,31.0,53.0
38.0,37.0,30.0,39.0,33.0,33.0,30.0,39.0
54.0,51.0,38.0,26.0,46.0,23.0,32.0,59.0
47.0,46.0,33.0,42.0,35.0,35.0,30.0,41.0
```

Output for Feature3(min)

```
cgmSeries_2,cgmSeries_15,cgmSeries_1,cgmSeries_9,cgmSeries_18

,,,,
,,,,
247.0,240.0,240.0,271.0,200.0
138.0,173.0,136.0,171.0,136.0
194.0,171.0,192.0,194.0,134.0
159.0,169.0,156.0,163.0,135.0
196.0,162.0,194.0,186.0,119.0
184.0,156.0,181.0,183.0,113.0
145.0,146.0,145.0,143.0,128.0
179.0,184.0,180.0,187.0,153.0
194.0,179.0,193.0,204.0,142.0
184.0,170.0,181.0,196.0,131.0
153.0,153.0,150.0,173.0,120.0
158.0,149.0,155.0,156.0,123.0
151.0,141.0,149.0,169.0,113.0
183.0,144.0,181.0,156.0,122.0
104.0,150.0,100.0,147.0,135.0
120.0,159.0,124.0,139.0,145.0
110.0,158.0,114.0,117.0,134.0
104.0,150.0,100.0,147.0,135.0
123.0,157.0,118.0,157.0,137.0
104.0,149.0,100.0,147.0,129.0
112.0,132.0,104.0,144.0,115.0
143.0,156.0,132.0,173.0,130.0
123.0,143.0,114.0,156.0,123.0
119.0,130.0,114.0,146.0,108.0
```

```
91.0,117.0,88.0,123.0,106.0
119.0,128.0,119.0,139.0,110.0
112.0,128.0,113.0,113.0,120.0
116.0,117.0,115.0,116.0,117.0
112.0,107.0,113.0,113.0,118.0
98.0,106.0,98.0,109.0,106.0
```

Output for Feature4(min)

```
cgmSeries_1,cgmSeries_2,cgmSeries_14,cgmSeries_9

,,,
,,,
240.0,247.0,248.0,271.0
199.0,201.0,188.0,212.0
196.0,201.0,188.0,198.0
202.0,208.0,258.0,233.0
225.0,218.0,193.0,209.0
193.0,195.0,192.0,196.0
225.0,218.0,193.0,209.0
181.0,184.0,192.0,196.0
193.0,194.0,187.0,204.0
211.0,213.0,192.0,218.0
240.0,233.0,210.0,227.0
155.0,158.0,177.0,171.0
180.0,178.0,204.0,191.0
189.0,191.0,182.0,174.0
180.0,178.0,204.0,191.0
158.0,158.0,168.0,156.0
180.0,178.0,204.0,191.0
147.0,153.0,187.0,189.0
180.0,178.0,204.0,191.0
167.0,166.0,238.0,205.0
231.0,231.0,225.0,231.0
191.0,193.0,211.0,209.0
220.0,223.0,233.0,231.0
210.0,216.0,206.0,236.0
223.0,225.0,233.0,229.0
206.0,211.0,216.0,220.0
239.0,245.0,236.0,261.0
215.0,220.0,210.0,223.0
239.0,245.0,236.0,261.0
215.0,220.0,210.0,223.0
```

I have already explained in the previous part how I got these values and these all are valid important features in the data, as we can see from the data there are a lot of spikes and slop so these are important features

d) Create a feature matrix where each row is a collection of features from each time series. SO if there are 75 time series and your feature length after concatenation of the 4 types of featues is 17 then the feature matrix size will be 75 X 17 (10 points)
to create matrix for pca run

```
python3  assignment1.py -m matrix
```

it will write the matrix data to ‘derivedMatrix.csv’

here is the output

```
MATRIX  FOR  PCA---->      cgmSeries_9_mean   cgmSeries_2_mean   cgmSeries_15_mean
cgmSeries_1_mean      cgmSeries_11_std      ...      cgmSeries_18_min      cgmSeries_2_min
cgmSeries_9_min  cgmSeries_15_min  cgmSeries_1_min
2            271.0            247.0            240.0            240.0            0.0  ...            200.0
247.0          271.0          240.0          240.0
3            192.0            169.0            176.0            167.0            15.0  ...            136.0
138.0          171.0          173.0          136.0
4            196.0            198.0            179.0            194.0            0.0  ...            134.0
194.0          194.0          171.0          192.0
5            190.0            179.0            196.0            177.0            34.0  ...            135.0
159.0          163.0          169.0          156.0
6            201.0            206.0            171.0            207.0            5.0  ...            119.0
196.0          186.0          162.0          194.0
7            190.0            191.0            172.0            189.0            4.0  ...            113.0
184.0          183.0          156.0          181.0
8            178.0            182.0            167.0            184.0            24.0  ...            128.0
145.0          143.0          146.0          145.0
9            191.0            181.0            185.0            180.0            0.0  ...            153.0
179.0          187.0          184.0          180.0
10           204.0            194.0            179.0            193.0            0.0  ...            142.0
194.0          204.0          179.0          193.0
11           207.0            198.0            177.0            196.0            6.0  ...            131.0
184.0          196.0          170.0          181.0
12           203.0            193.0            174.0            193.0            19.0  ...            120.0
153.0          173.0          153.0          150.0
13           164.0            158.0            165.0            155.0            7.0  ...            123.0
158.0          156.0          149.0          155.0
14           180.0            165.0            175.0            163.0            13.0  ...            113.0
151.0          169.0          141.0          149.0
15           165.0            187.0            166.0            185.0            7.0  ...            122.0
183.0          156.0          144.0          181.0
16           164.0            151.0            175.0            151.0            13.0  ...            135.0
104.0          147.0          150.0          100.0
17           149.0            139.0            167.0            141.0            6.0  ...            145.0
120.0          139.0          159.0          124.0
```

18	156.0	137.0	175.0	136.0	20.0	...	134.0	110.0	117.0	158.0	114.0
19	161.0	130.0	169.0	127.0	13.0	...	135.0	104.0	147.0	150.0	100.0
20	171.0	143.0	180.0	141.0	11.0	...	137.0	123.0	157.0	157.0	118.0
21	170.0	139.0	177.0	138.0	26.0	...	129.0	104.0	147.0	149.0	100.0
22	175.0	154.0	176.0	151.0	32.0	...	115.0	112.0	144.0	132.0	104.0
23	192.0	174.0	192.0	167.0	15.0	...	130.0	143.0	173.0	156.0	132.0
24	203.0	181.0	193.0	175.0	28.0	...	123.0	123.0	156.0	143.0	114.0
25	185.0	168.0	178.0	163.0	26.0	...	108.0	119.0	146.0	130.0	114.0
26	196.0	183.0	195.0	180.0	42.0	...	106.0	91.0	123.0	117.0	88.0
27	183.0	175.0	181.0	174.0	30.0	...	110.0	119.0	139.0	128.0	119.0
28	188.0	182.0	182.0	180.0	53.0	...	120.0	112.0	113.0	128.0	113.0
29	169.0	169.0	171.0	170.0	39.0	...	117.0	116.0	116.0	117.0	115.0
30	182.0	176.0	178.0	173.0	59.0	...	118.0	112.0	113.0	107.0	113.0
31	164.0	161.0	171.0	160.0	41.0	...	106.0	98.0	109.0	106.0	98.0

[30 rows x 21 columns]

Note 0.0=NAN values

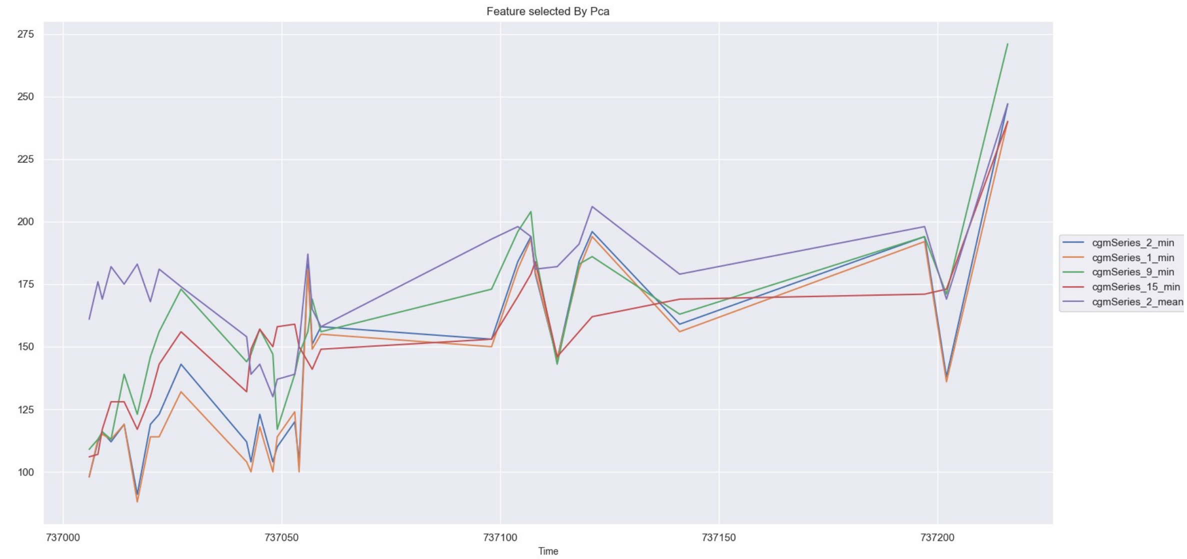
e) Provide this feature matrix to PCA and derive the new feature matrix. Chose the top 5 features and plot them for each time series. (5 points)

Lets run the PCA:

```
python3  assignment1.py -m pca
```

Output
Fisrt Pricncpal Component[0.21733285 0.08614395 0.2451086 0.2386494 0.09612052 0.14096799
0.09644583 0.08640853 0.13058543 0.16055111 0.10857047 0.16072668
0.02347255 0.03594191 0.04659873 0.0469037 0.14950063 0.28111919
0.46201078 0.40163057 0.46864806]
[20 18 19 17 2]
Top 5 Featues from PCA=cgmSeries_2_min
Top 5 Featues from PCA=cgmSeries_1_min
Top 5 Featues from PCA=cgmSeries_9_min
Top 5 Featues from PCA=cgmSeries_15_min
Top 5 Featues from PCA=cgmSeries_2_mean

Plot for top 5 features2

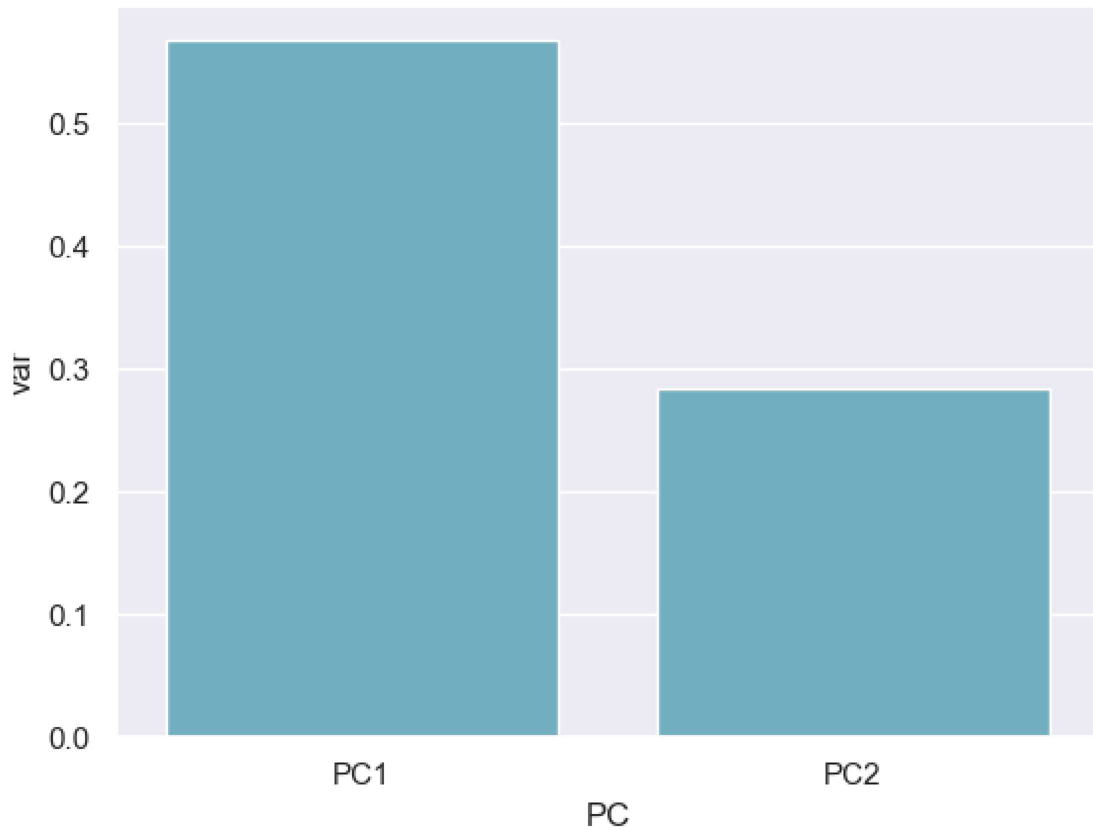


f) For each feature in the top 5 argue why it is chosen as a top five feature in PCA? (3 points each) total 15

Run

```
python3  assignment1.py -m pca
```

output



Fisrt Pricncpal Component[0.21733285 0.08614395 0.2451086 0.2386494 0.09612052 0.14096799
0.09644583 0.08640853 0.13058543 0.16055111 0.10857047 0.16072668
0.02347255 0.03594191 0.04659873 0.0469037 0.14950063 0.28111919
0.46201078 0.40163057 0.46864806]
[20 18 19 17 2]
Top 5 Featues from PCA=cgmSeries_2_min
Top 5 Featues from PCA=cgmSeries_1_min
Top 5 Featues from PCA=cgmSeries_9_min
Top 5 Featues from PCA=cgmSeries_15_min
Top 5 Featues from PCA=cgmSeries_2_mean

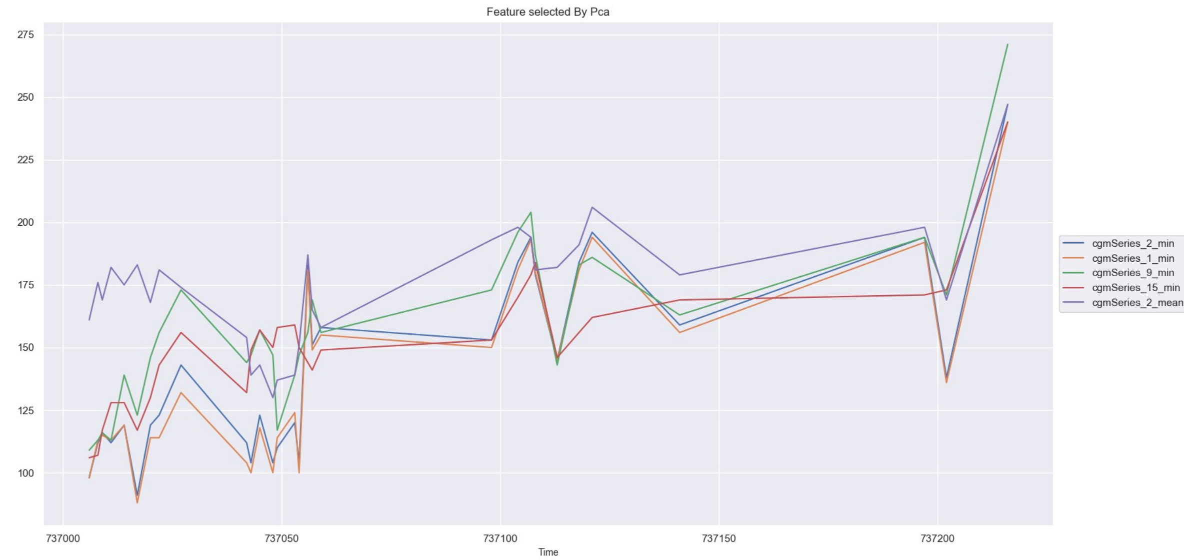
As we can see, 73% of varience belongs to pc1, and 27% to PC2

We will select pca1, we selected the 5 features with maximum pc1 values and 5 selected features based on maximum PC1 values are

1)cgmSeries_2_min

2)cgmSeries_1_min
3)cgmSeries_9_min
4)cgmSeries_15_min
5)cgmSeries_2_mean1

PLot



if we compare this graph to the original data, the original looks smooth while the 5 features selected are the features with high variation.