موسسه آموزش عالی آزاد توسعه

برگزار کننده دوره‌های تخصصی علم داده

Homework 5: Summer 2019

Please email your HWs to y.zerehsaz@gmail.com

Please append all your codes to your response

This HW must be completed in **Python**.

Consider the "Spine" dataset to answer the following questions.

The Spine dataset contains information about patients belonging to one of three categories of lumbar spine malfunctions: 1) Normal, 2) Disk Hernia and 3) Spondylolisthesis with the last two categories being abnormal. Each patient is represented in the data set by six biomechanical attributes derived from the shape and orientation of the pelvis and lumbar spine (in this order): pelvic incidence (PI), pelvic tilt (PT), lumbar lordosis angle (LL), sacral slope (SS), pelvic radius (PR) and grade of spondylolisthesis (GS).

Question 1:

a) Get the summary of data and check whether or not there are any missing values.

```
df.describe(include="all")
```

Index	PI	PT	LL	SS	PR	GS	Categories
count	310	310	310	310	310	310	310
max	129.83	49.432	125.74	121.43	163.07	418.54	nan
75%	72.878	22.12	63	52.696	125.47	41.287	nan
mean	60.497	17.543	51.931	42.954	117.92	26.297	nan
50%	58.691	16.358	49.562	42.405	118.27	11.768	nan
25%	46.43	10.667	37	33.347	110.71	1.6037	nan
min	26.148	-6.5549	14	13.367	70.083	-11.058	nan
std	17.237	10.008	18.554	13.423	13.317	37.559	nan
unique	nan	nan	nan	nan	nan	nan	3
top	nan	nan	nan	nan	nan	nan	Spondylolisthesis

It seems that GS variable is scaled differently from the rest of variables. From the mean and median, we can tell that PI and GS are skewed. There are three categories with Spondylolisthesis as the most frequent one.

Fortunately, there are no missing values in the data since sum(df.isna().unstack())= 0

b) Get the structure of the data frame.

```
df.dtypes
PI           float64
PT           float64
LL           float64
SS           float64
PR           float64
GS           float64
Categories    object
```

c) Compute the mean and standard deviation for the PI variable, and mean, median and standard deviation for GS variable. Use the agg function.

```
df.agg({'PI':['mean','var'],'GS':['mean','std','median']})
```

Index	PI	GS
mean	60.497	26.297
median	nan	11.768
std	nan	37.559
var	297.1	nan

d) Group the whole dataframe based on the “Categories” (last column in Spine dataset). Compute the mean and standard deviation associated with each group for all variables. What do you think about the differences in means and standard deviations of variables among the levels of the variable “Categories”?

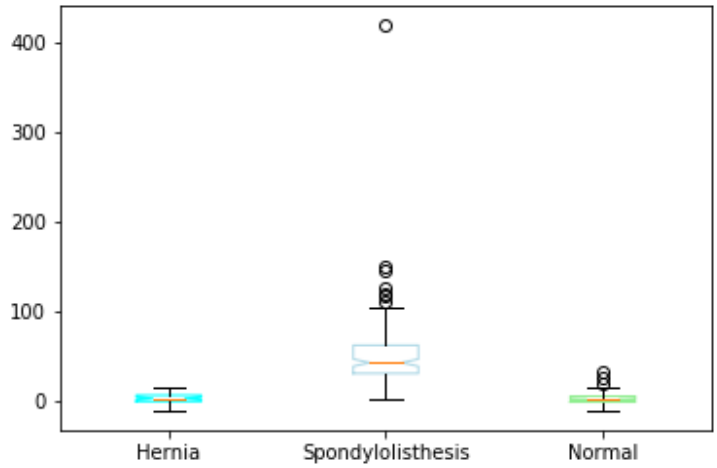
```
df2=df.groupby(df['Categories'])

df2.agg(['mean','std'])
```

Index	('PI', 'mean')	('PI', 'std')	('PT', 'mean')	('PT', 'std')	('LL', 'mean')	('LL', 'std')	('SS', 'mean')	('SS', 'std')	('PR', 'mean')	('PR', 'std')	('GS', 'mean')	('GS', 'std')
Hernia	47.64	10.7	17.4	7.017	35.46	9.768	30.24	7.555	116.5	9.356	2.48	5.531
Normal	51.69	12.37	12.82	6.779	43.54	12.36	38.86	9.624	123.9	9.014	2.187	6.307
Spondylolisthesis	71.51	15.11	20.75	11.51	64.11	16.4	50.77	12.32	114.5	15.58	51.9	40.11

It is noticeable that for all variables except PR, the mean and standard deviation for Spondylolisthesis patients are higher than those for other categories. For patients with Hernia, most of variables tend to take smaller values.

e) Compare the boxplots for the variable GS corresponding to the groups of the variable “Categories”. You must locate all boxplots in one plot (see googleapps.py file).

```
l=[df['GS'].loc[df['Categories']==x] for x in df['Categories'].unique()]
box=plt.boxplot(l,notch=True,labels= df['Categories'].unique())#Boxplots for Rating associated with each category
colors = ['cyan', 'lightblue', 'lightgreen']
for patch, color in zip(box['boxes'], colors):
    patch.set_color(color)
```



The variable GS is an appropriate variable to separate Spondylolisthesis from the rest of categories. The higher average and larger standard deviation are obvious. An extreme value on this variable is also observable around 400.

Question 2:

Perform the PCA method on the ***scaled*** data (do not use the last column) and answer the following questions.

a) Scale the data and apply PCA on the scaled dataset.

```
X=df.iloc[:,0:6]
X=scale(X)
pca=PCA()
pca.fit(X)
```

b) Provide the matrix of directions (loadings). Interpret the first three directions. This is the "W" matrix in the slides.

```
W=pca.components_.T
pd.DataFrame(W[:,:3],index=df.columns[:-1],columns=['PC1','PC2','PC3'])
       PC1       PC2       PC3
PI  0.535142 -0.002194 -0.096069
PT  0.323585  0.527545 -0.648701
LL  0.457970  0.092875  0.152338
SS  0.445906 -0.396157  0.360313
PR -0.143497  0.727756  0.585991
GS  0.423978  0.162777  0.271184
```

The first PC gives a contrast between the average of PI, PT, LL, SS and GS against PR. The second PC gives a contrast between the average of PR, GS, PT, LL against SS. PC3 provides a contrast between the average of PI and PT against the average of LL, SS, PR and GS.

c) Compute the explained variance ratio and decide on the number of components to choose (justify your answer).

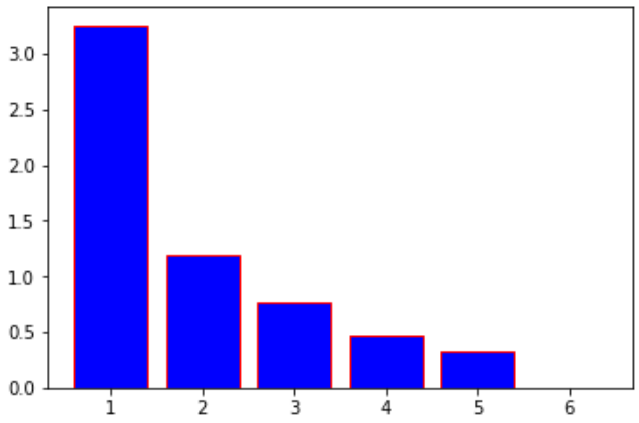
```
pd.DataFrame(pca.explained_variance_ratio_.cumsum(),index=np.arange(X.shape[1])+1,colu
mns=['Explained Variability'])
```

	Explained Variability
1	0.540964
2	0.740061
3	0.866909
4	0.945664
5	1.000000
6	1.000000

Obviously, the first two components explain about 74% of the variability. So, we can work with these two components.

d) Using the scree plot, how many components would you like to choose? (justify your answer).

```
plt.bar(np.arange(1,X.shape[1]+1),pca.explained_variance_,color="blue",edgecolor="Red")
```

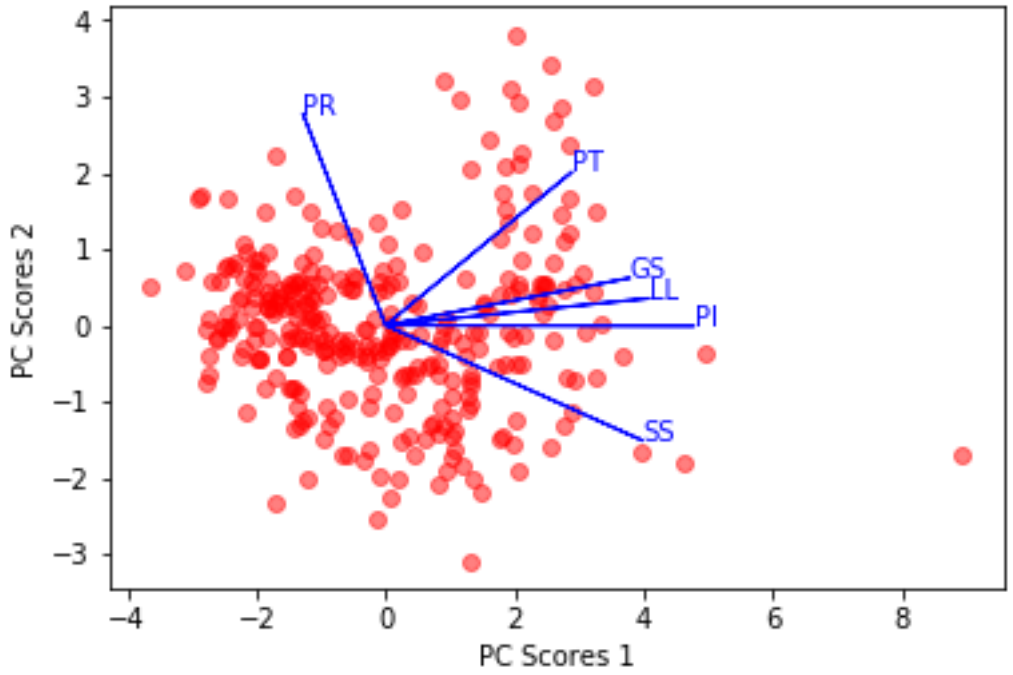


There is an elbow on the second component, so we can choose the first two components.

e) Make a scatter plot of the first two PC scores (**use the output of Python i.e. fit_transform**) and interpret the observations (patients) based on the loading matrix and their position in the plot. Are there any unusual observations (possible outliers) in the data?

f)

```
    Y=pca.fit_transform(X)
    plt.figure(1)
    plt.scatter(Y[:,0],Y[:,1],c="red",marker='o',alpha=0.5)
    plt.xlabel('PC Scores 1')
    plt.ylabel('PC Scores 2')
    xs=Y[:,0]
    ys=Y[:,1]
for i in range(len(W[:,0])):
    plt.arrow(np.mean(xs), np.mean(ys), W[i,0]*max(xs), W[i,1]*max(ys),
              color='b', width=0.0005, head_width=0.0025)
    plt.text(W[i,0]*max(xs)+np.mean(xs), +np.mean(ys)+W[i,1]*max(ys),
             list(df.columns.values)[i], color='b')
```



The points on the right side down the plot have large positive PC score1 and negative PC score 2. This says variables SS, PI, LL, GS, and PT are larger than their average compared to the rest of observations for these points. On the vertical axis, these points might have larger than average SS values or very smaller than average PR.

There is one unusual observation which is a possible outlier.

g) If you have found several outliers in part e, choose the worst one and answer the following questions.

I) Which observation does this outlier belong to?

```
np.where(Y[:,0]>7)
(array([115], dtype=int64),)
```

II) On what variables (variable) do you think this outlier has the highest values?

Based on the biplot, we can say the responsible factor might be extremely large values on SS, PI, LL and GS. This point might have a small value on PR as well.

Let's take a look at this observation:

```
df.iloc[115,:]
PI              129.834
PT              8.40448
LL              48.3841
SS              121.43
PR              107.69
GS              418.543
```

Compare it to the maximum values of the variables:

```
df.agg('max')
PI              129.834
PT              49.4319
LL              125.742
SS              121.43
PR              163.071
GS              418.543
Categories    Spondylolisthesis
```

The comparison shows that this observation takes the maximum values on three variables including PI, SS and GS, and that's why it is located far from the cloud of data.