Hands-on Practice for Module 1: Exploratory Data Analysis

0.Importing important packages

In [ ]:

```
# data loading and computing functionality
import pandas as pd
import numpy as np
import scipy as sp

# datasets in sklearn package
from sklearn import datasets
from sklearn.datasets import load_digits

# visualization packages
import seaborn as sns
import matplotlib.pyplot as plt
import matplotlib.cm as cm

#PCA, SVD, LDA
from sklearn.decomposition import PCA
from scipy.linalg import svd
from sklearn.discriminant_analysis import LinearDiscriminantAnalysis
```

1. Loading data, determining samples, attributes, and types of attributes

Use Davis dataset avaialble at the url https://www.rdocumentation.org/packages/car/versions/2.1-6/topics/Davis (https://www.rdocumentation.org/packages/car/versions/2.1-6/topics/Davis)

Description of the data is provided at https://www.rdocumentation.org/packages/car/versions/2.1-6/topics/Davis (https://www.rdocumentation.org/packages/car/versions/2.1-6/topics/Davis)

Drop rows in the data set with missing values (NA), using dropna(inplace=True) function.

Question 1a: What does the data capture?

Answer:

Question 1b: Who are selected as subjects in the study that collected the data?

Answer:

Question 1c: How many data points are in this dataset?

In [ ]:

```
davis_df = pd.read_csv('https://vincentarelbundock.github.io/Rdatasets/csv/carData/Davis.csv')
```

In [ ]:

```
davis_df.dropna(inplace=True);
```

In [ ]:

```
```

Answer:

Question 1d: How many attributes are in this dataset?

Answer:

Question 1e: What type of attributes are present in the dataset?

In [ ]:

```
```

Answer:

2. Generating summary statistics

Use 'Davis' data. Do not include Unnamed attribute in this analysis.

In [ ]:

```
davis_df.drop(columns=davis_df.columns[davis_df.columns.str.contains('unnamed', case=False)], inplace=True)
davis_df.head()
```

Question 2a: What are range of values the numeric attributes take?
[Hint: Use exclude=object option in describe() function to ignore the attribute sex]

In [ ]:

```
```

Answer:

Question 2b: What different values do categorical attributes take?
[Hint: Use include=object option in describe() function to ignore the attribute sex]

In [ ]:

Answer:

Question 2c: What are the mean values for each of the numeric attributes?

In [ ]:

Question 2d: What is the variance for each of the numeric attributes?

In [ ]:

Question 2e: Visually examine how the attribute weight is distributed and comment if the data is Normally distributed?

In [ ]:

Answer:

Question 2f: Visually examine how the attribute height is distributed and comment if the data is Normally distributed?

In [ ]:

Answer:

Question 2g: Visually examine how the attribute repwt is distributed and comment if the data is Normally distributed?

In [ ]:

Answer:

Question 2h: Visually examine how the attribute repht is distributed and comment if the data is Normally distributed?

```
In [ ]:
```

Answer:

Question 2i: Visually examine how the attribute sex is distributed and comment if the data is uniformly distributed?

```
In [ ]:
```

Answer:

3. Geometric and Probabilistic view

For this part, we will restrict to repwt and repht attributes in the davis dataset as we can only visualize 2D space.

```
In [ ]: davis_df_new = davis_df[['repwt','repht']]
```

```
In [ ]: davis_df_new.head()
```

Question 3a: Show the Geometric view of this new row normalized data on a 2D space along with the mean.

```
In [ ]:
```

```
In [ ]:
```

We will further normalize the magnitude of each row in the data (davis_df_new) to 1 and use the new dataframe davis_df_new_row_norm.

```
In [ ]: from sklearn.preprocessing import normalize
        davis_df_new_row_norm = normalize(davis_df_new, axis=1, norm='l2')
```

```
In [ ]: davis_df_new_row_norm[1:10,:]
```

Question 3b: Show the Geometric view of this new row normalized data on a 2D space along with the mean. Comment on the Geomateric view of the data in comparison to the view you observed in Question 3a. Provide a reason for the difference in the geometric views in Question 3a and 3b.

In [ ]:

Answer:

Question 3c: Show the Probabilistic view of the data davis_df_new.

In [ ]:

In [ ]:

We will normalize the magnitude of each column in the data (davis_df_new) to 1 and use the new dataframe davis_df_new_col_norm.

In [ ]:

```
davis_df_new_col_norm = normalize(davis_df_new, axis=0, norm='l2')
```

In [ ]:

```
davis_df_new_col_norm[1:10,:]
```

Question 3d: Show the Probabilistic view of the data davis_df_new_col_norm. Compare the shape of the covariance structure in the Gaussian distribution with that of Question 3c and comment if column normalization has affected the shape of the covariance structure.

In [ ]:

In [ ]:

Answer:

4. Understanding the (in)dependencies among attributes using Covariance matrix

Use 'Davis' data. Do not include Unnamed attribute in this analysis.

Question 4a: What is the covariance matrix?

In [ ]:

```
```

Question 4b: Which pairs of attributes co-vary in the opposite direction?

Answer:

Question 4c: Which pairs of attributes are highly correlated?

In [ ]:

```
```

Answer:

Question 4d: Which pairs of attributes are uncorrelated?

Answer:

Question 4e: What information did you gather from a correlation matrix that is not available in a covariance matrix?

Answer:

5. Dimensionality Reduction: Feature Selection

Data: Iris dataset from the practice notebook. (https://raw.githubusercontent.com/plotly/datasets/master/iris.csv (https://raw.githubusercontent.com/plotly/datasets/master/iris.csv))

Assumption: Assume that your goal is to cluster the data to identify the species 'Name'. Clustering algorithm takes as input data points and attributes. It groups points that are similar to each other into a separate cluster. It puts points that are dissimilar in different cluster. Note that the 'Name' attribute will be hidden from the clustering algorithm.

In [ ]:

```
import seaborn as sns
iris_df = pd.read_csv('https://raw.githubusercontent.com/plotly/dataset
s/master/iris.csv')
```

Question 5a: If you are allowed to select only one attribute, which attribute would be highly useful for the clustering task. Provide a reason. Use pairplot to answer this question.

Answer:

Question 5b: If you are allowed to select only two features, which feature would be highly useful for the clustering task. Provide a reason. Use pairplot to answer this question.

Answer:

Question 5c: In real-world problems ground-truth (types of iris plants) will not be available to select the features, how do you perform **feature selection** in that case?

Answer:

Question 5d: In real-world problems ground-truth (types of iris plants) will not be available to select the features, how do you perform **dimensionality reduction** in that case? What limitations does your approach have?

Answer:

6. Dimensionality Reduction: PCA on Iris Data

Question 6a: Perform PCA on Iris dataset and project the data onto the first two principal components. Use the attributes 'SepalLength','SepalWidth','PetalLength', and 'PetalWidth'.

Hint: Use iris_df[['SepalLength','SepalWidth','PetalLength','PetalWidth']] to use the specified attributes.

In [ ]:

Question 6b: Generate a pairplot (along with colors for the different types of iris plants) between the two newly generated features using PCA in the above step.

In [ ]:

Question 6c: From the above pairplot, if only one newly generated attribute were to be used for clustering the data which newly generated attribute is best suited. Provide a reason. Is the newly generated attribute better than the feature selected in Question 4a?

Answer:

Question 6d: From the above pairplot, if two newly generated attributes were to be used for clustering the data, are the two newly generated attributes better than the features selected in Question 4b?

Answer:

7. Dimensionality Reduction: PCA on synthetic datasets

Consider the following synthetic dataset we refer to as **Blobs**. This dataset has 500 data points centered around (-5, -5), (0,0) and (5,5). This dataset has 1500 data points and 2 attributes.

In [ ]:

```
n_samples = 1500
random_state = 42
centers = [(-5, -5), (0, 0), (5, 5)]
Blobs_X, Blobs_y = datasets.make_blobs(n_samples=n_samples,centers=cente
rs,random_state=random_state)
```

In [ ]:

```
Blobs_X.shape
```

In [ ]:

```
plt.figure(figsize=(3,3))
plt.scatter(Blobs_X[:, 0], Blobs_X[:, 1], c= Blobs_y)
plt.title('Blobs')
```

We generated a new dataset **Blobs1** by adding an extra attribute to this 2D Blobs dataset. The values for this new attribute are drawn from a normal distribution with mean 0 and variance 1.

In [ ]:

```
Blobs1= pd.DataFrame(Blobs_X)
Blobs1['2'] = np.random.randn(1500)
Blobs1.head()
```

We generated a new dataset **Blobs2** by adding an extra attribute to the 2D Blobs dataset. The values for this new attribute are drawn from a normal distribution with mean 0 and variance 100. Read more about how to do this at https://docs.scipy.org/doc/numpy-1.15.1/reference/generated/numpy.random.randn.html (https://docs.scipy.org/doc/numpy-1.15.1/reference/generated/numpy.random.randn.html).

```
In [ ]: Blobs2= pd.DataFrame(Blobs_X)
        Blobs2['2'] = np.random.randn(1500)*10
        Blobs2.head()
```

We generated a new dataset **Blobs3** by adding two extra attributes to the 2D Blobs dataset. The values for the two new attributes are drawn from a normal distribution with mean 0 and variance 100.

```
In [ ]: Blobs3= pd.DataFrame(Blobs_X)
        Blobs3['2'] = np.random.randn(1500)*10
        Blobs3['3'] = np.random.randn(1500)*10
        Blobs3.head()
```

Question 7a: Plot pairplot for **Blobs1** data. By visually examining this plot, comment on the variance of the third attribute in comparison to the first two attributes.

```
In [ ]:
```

Answer:

Question 7b: Perform PCA on **Blobs1** data. Project data onto the first two principal components. Generate a pairplot for the newly constructed attributes.

```
In [ ]:
```

```
In [ ]:
```

Question 7c: By comparing the distributions for the newly generated attributes in Question 7b with the previous pairplot in Question 7a, determine which attribute is captured by the first principal component and which attribute is captured by the second principal component. Provide a reason for your observations.

Answer:

Question 7d: Plot pairplot for **Blobs2** data. By visually examining this plot, comment on the variance of the third attribute in comparison to the first two attributes.

```
In [ ]:
```

Answer:

Question 7e: Perform PCA on **Blobs2** data. Project data onto the first two principal components. Generate a pairplot for the newly constructed attributes.

```
In [ ]:
```

```
In [ ]:
```

Question 7f: By comparing the distributions for the newly generated attributes in Question 7e with the previous pairplot in Question 7d, determine which attribute is captured by the first principal component and which attribute is captured by the second principal component. Why would have caused this (in comparison to your observation in Question 7c)?

Answer:

Question 7g: Are the three blobs separately visible after projection based on PCA in Question 7e?

Answer:

Question 7h: Plot pairplot for **Blobs3** data. By visually examining this plot, comment on the strength of the correlation between the first two attributes. Also, comment on the strength of the correlation between the second two attributes.

```
In [ ]:
```

Answer:

Question 7i: Perform PCA on **Blobs3** data. Project data onto the first two principal components. Generate a pairplot for the newly constructed attributes.

```
In [ ]:
```

```
In [ ]:
```

Question 7j: By comparing the distributions for the newly generated attributes in Question 7i with the previous pairplot in Question 7h, determine which attribute is captured by the first principal component and which attribute is captured by the second principal component. Why would have caused this (in comparison to your observation in Question 7f and 7c)?

Answer:

Question 7k: Are the three blobs separately visible after projection based on PCA in Question 7i? What would have caused this, in comparison to your observation in Question 7g?

Answer:

Question 7l: What limitation of PCA do your observations in Questions 7j, 7f, and 7c highlight?

Answer:

8. Singular Value Decomposition

Question 8a: Using the code provided in the practice notebook for computing PCA, write your own SVD function (U,S,V = mysvd(A)) to factorize the matrix A into U,S, and V.

In [ ]:

Question 8b: Demonstrate that your code is correct by using your function on the following matrix A and showing that the product $USV^T = A$.

In [ ]:

```
A = np.array([
    [1, 1, 1, 0, 0, 0],
    [3, 3, 3, 0, 0, 0],
    [4, 4, 4, 0, 0, 0],
    [5, 5, 5, 0, 0, 0],
    [0, 1, 0, 4, 4, 1],
    [0, 0, 0, 5, 5, 2],
    [0, 0, 0, 2, 2, 2]])
```

In [ ]:

Question 8c: Perform SVD on iris dataset and visualize the proportion of variance captured by each spectral value. List the dimensions that captures less than 10% of the total variance.

In [ ]:

```
import pandas as pd
iris_df = pd.read_csv('https://raw.githubusercontent.com/plotly/datasets/master/iris.csv')
```

In [ ]:
```
data = iris_df.values[:,0:4]
data = data.astype(float) #converts data format from object to numeric
```

In [ ]:
```
```

In [ ]:
```
```

Answer:

Question 8d: The heatmap of the full data is shown below. Plot all the four spectral decomposition matrices based on SVD.

In [ ]:
```
sns.heatmap(data,vmin=0, vmax=7)
```

In [ ]:
```
```

In [ ]:
```
```

In [ ]:
```
```

In [ ]:
```
```

Question 8e: Visually examine the magnitude of values present in each of the four spectral decomposition matrices and comment on which two of the four matrices have elements with relatively small magnitude in them. Provide a reason for this based on your obsevation in Question 8c.

Answer:

9. Linear Discriminant Analysis

We will use digits data for studying the use of LDA.

In [ ]:
```
digits = load_digits()
```

The data with 1797 samples and 64 attributes is in the object digits.data. These 64 attributes represent pixels in an 8x8 image.

In [ ]:
```
digits.data.shape
```

The 1797 images are digits from 0...9. This information is in the digits.target variable.

```
In [ ]: digits.target
```

For this part, we will only focus on digits 3 and 8. To this end, we generate indices of 183 samples with 3s and indices of 174 samples with 8s.

```
In [ ]: Threes = np.where(digits.target==3)
        Eights = np.where(digits.target==8)
        [np.size(Threes), np.size(Eights)]
```

We will take samples from these indices and construct a matrix X such that the first 183 samples represent 3s and the remaining ones represent 8s. The variable y captures this information.

```
In [ ]: indices = np.hstack((Threes[0], Eights[0]));
        X = digits.data[indices,:]
        y = np.hstack((3*np.ones(np.size(Threes)), 8*np.ones(np.size(Eights))))
```

```
In [ ]: X
```

```
In [ ]: X.shape
```

```
In [ ]: y
```

```
In [ ]: y.shape
```

Question 9a: Visually examine the following heatmap of the data X and identify one attribute that can separate the 3s from 8s. Also comment on (approximately) how many mistakes would be committed if this attribute is used for projection in LDA.

```
In [ ]: plt.figure(figsize=(20,10))
        ax = sns.heatmap(X,cmap='PiYG')
        ax.set(xlabel='Attributes', ylabel='Samples')
```

Answer:

Question 9b: Perform LDA on this data. Plot the heatmap of the projected data and comment how many points will be wrongly predicted based on this projection.

```
In [ ]: 
```

```
In [ ]:
```

Answer:

```
In [ ]:
```