Exercise 6 : Clusters and Anomalies

Workflow

1. Create a folder on your Desktop and name it Cx1015_[LabGroup], where [LabGroup] is the name of your Group
2. Download the .ipynb files and data files posted corresponding to this exercise and store in the aforesaid folder
3. Open Jupyter Notebook (already installed on the Lab computer) and navigate to the aforesaid folder on Desktop
4. Open and explore the .ipynb files (notebooks) that you downloaded, and go through "Preparation", as follows
5. The walk-through videos posted on NTU Learn (under Course Content) may help you with this "Preparation" too
6. Create a new Jupyter Notebook, name it Exercise6_solution.ipynb, and save it in the same folder on the Desktop
7. Solve the "Problems" posted below by writing code, and corresponding comments, in Exercise6_solution.ipynb

Try to solve the problems on your own. Take help and hints from the "Preparation" codes and the walk-through videos. **If you are still stuck, talk to your friends in the Lab to get help/hints.** If that fails too, approach the Lab Instructor.

Note : Don't forget to import the Essential Python Libraries required for solving the Exercise. Write code in the usual "Code" cells, and notes/comments in "Markdown" cells of the Notebook. Check the preparation notebooks for guidance.

Preparation

M5 ClusteringPatterns.ipynb	Check how to perform basic Clustering on the Pokemon data (pokemonData.csv)
M5 DetectingAnomalies.ipynb	Check how to perform basic Anomaly Detection on the Pokemon dataset

Objective

Let us assume that the houses in our dataset vary by their Living Area (GrLivArea) and Garages (GarageArea) in general. In this exercise, we will try to find patterns in the data by clustering the house as per their Living Area and Garage Area. We will also try to identify major anomalies in the dataset, once again, in terms of their Living Area and Garage Area.

Problems

Download the dataset **train.csv** and the associated text file **data_description.txt** posted with this Exercise.

Problem 1 : Clustering using GrLivArea and GarageArea

Import the complete dataset "train.csv" in Jupyter, as `houseData = pd.read_csv('train.csv')`

a) Extract the two variables in consideration from the dataset

```
X = pd.DataFrame(houseData[['GrLivArea','GarageArea']])
```

b) Visualize the 2D distribution of the two variables extracted above, using a standard scatter plot.

c) Import k-Means Clustering model from Scikit-Learn : `from sklearn.cluster import KMeans`

d) Guess the number of clusters from the 2D scatterplot, and perform k-Means clustering with that.

e) Print the cluster centers, view their countplot, and visualize the clusters on the 2D scatterplot.

Problem 2 : Anomaly Detection with the same Variables

Import the complete dataset "train.csv" in Jupyter, as `houseData = pd.read_csv('train.csv')`

a) Extract the two variables in consideration from the dataset

```
X = pd.DataFrame(houseData[['GrLivArea','GarageArea']])
```

b) Visualize the 2D distribution of the two variables extracted above, using a standard scatter plot.

c) Import Anomaly model from Scikit-Learn : `from sklearn.neighbors import LocalOutlierFactor`

d) Guess the parameters from the 2D scatterplot, and perform Anomaly Detection with those parameters.

e) View their countplot of Anomalies vs Normal Data, and visualize the anomalies on the 2D scatterplot.

Extra Resources

You may read more about `KMeans` and `LocalOutlierFactor` you use in this exercise in the following references.

`KMeans` : https://scikit-learn.org/stable/modules/generated/sklearn.cluster.KMeans.html

`LOF` : https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.LocalOutlierFactor.html

Other Clustering Algorithms : https://scikit-learn.org/stable/modules/classes.html#module-sklearn.cluster

Other Anomaly Detection Algorithms : https://scikit-learn.org/stable/modules/outlier_detection.html

Bonus Problems

1. Try using the DBSCAN Clustering Algorithm on the same dataset as above, and check the difference with KMeans.
   `DBSCAN` (Scikit Learn) : https://scikit-learn.org/stable/modules/generated/sklearn.cluster.DBSCAN.html

2. Try using IsolationForest Anomaly Detection Algorithm on the same dataset, and check the difference with LOF.
   `IsolationForest` : https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.IsolationForest.html

Do you think the structures and layout of the clusters are different in the above two clustering algorithms? Do the two anomaly detection algorithms above identify different set of anomalies, or with overlap? Experiment, and think about it.