|  |
| --- |
| **Machine Learning**  **Lab2** |

****

|  |  |  |  |
| --- | --- | --- | --- |
| **제출일** | **2022.9.27** | **전공** | **소프트웨어학과** |
| **과목명** | **머신러닝** | **이름** | **안해빈, 이지민,**  **임윤수, 윤주은** |

**1. Objective Setting**

**We would like to show different combinations of algorithms to analyze a dataset.** We will create a program structure with a single major function “**AutoML**” that will automatically run different combinations of the following :

* Data Scaling, Data Encoding
* Various Clustering Algorithms
* Various quality measuring tools
* Various values of parameters and hyperparameters
* Various subsets of the features of the dataset

We will use the California Housing Prices Dataset to test the algorithm in the end-to-end process as used in Data Science.

The function “**AutoML**” will be capable of getting inputs of various combinations and tell the user if the algorithm / scaler / encoder and so on is supported.

We will explain the meaning of the clustering results, in terms of the features used in clustering the dataset.

* we will also try to combine some attributes to make a new attribute and analyze the difference in the results.

**2. Data Curation**

Dataset Name : California Housing Prices

Source : <https://www.kaggle.com/datasets/camnugent/california-housing-prices>

**3. Data Inspection**

- Number of Instances: 20640

- Number of Attributes: 10

- Attribute Information:

1. longitude: A measure of how far west a house is; a higher value is farther west

2. latitude: A measure of how far north a house is; a higher value is farther north

3. housingMedianAge: Median age of a house within a block; a lower number is a newer building

4. totalRooms: Total number of rooms within a block

5. totalBedrooms: Total number of bedrooms within a block

6. population: Total number of people residing within a block

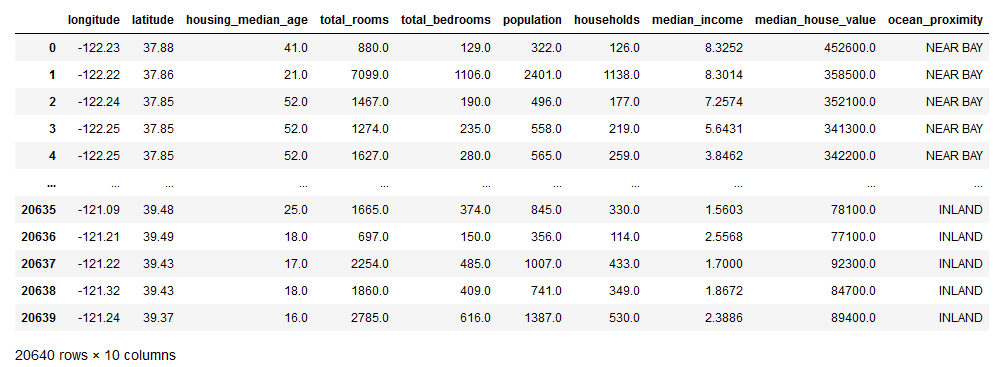
7. households: Total number of households, a group of people residing within a home unit, for a block

8. medianIncome: Median income for households within a block of houses (measured in tens of thousands of US Dollars)

9. medianHouseValue: Median house value for households within a block (measured in US Dollars) // *do not use in clustering*

10. oceanProximity: Location of the house w.r.t ocean/sea

- Missing attribute values: **207** in attribute “totalBedrooms”



**4. Data Preparation - Plans**

Dirty data cleaning

→ fill missing values to MEDIAN value in “totalBedrooms”

Feature engineering

→ We will try to make a new feature by combining original data

**Scaling (5)** - Standard, MinMax, Robust, MaxAbsScaler, Normalizer

**Encoding (2)** - Label, OneHot

→ for “OceanProximity” att

**5. Data Analysis - Plans**

K-means

EM(GMM)

CLARANS

DBSCAN

Spectral Clustering

----------------------------------

+ try different parameters, hyperparameters

+ try 2-12 clusters

**6. Evaluation**

Silhouette score for K using algorithms

Knee method

Purity

**7. Algorithm Structure**

def AutoML():

"""Train model and show classification results

Parameters::

scaler = [‘StandardScaler()’, ‘MinMaxScaler()’, ‘RobustScaler()’, ‘MaxAbsScaler()’, ‘Normalizer()’]

model = [k-means’, ‘’EM’’, “CLARANS”, DBSCAN”, “spectral clustering”]

encoder = [label, onehot]

Returns::

Plot for result of clustering, Show evaluation results if possible

*The program indicates if the scaler / encoder / cluster algorithm is supported*

"""

+ Provided Details

1. Try (just once) a subset  of the 9 original attributes.  
      Try to think about such a subset, based on your guess of which attributes will have the greatest impact on the Median House Prices attribute.

2. After generating clusters, try to divide the range of values of the Median House Prices attribute into N (e.g., 2, 3, 4, 5), and compare, for each N, with

the k number of clusters, and see how they are related.

(You don't need to try all 10 values for N; you may try a few values.)

3. For each model, separately, examine the clustering result, and determine

the good values for the hyperparameters, k, scaling, encoding, etc.

After you establish a good set of the hyperparameters, use them

in calling AutoML.This will substantially reduce the number of outcomes

of AutoML that you need to "eyeball".

4. From the main program, call AutoML a few times.

   Once with a full set of parameter lists (e.g., 5 models, 5 scalers, 2 encoders,...)

   Once with a reduced parameter lists (e.g., 3 models, 3 sdcalers, 2 encoders,...)

    (Each of the above, with a dataset with 9 attributes, and again with a dataset

      with fewer than 9 attributes (as explained above).

5. The overall program structure

      The main program   (preprocessing, set up parameters lists, call AutoML....)

      AutoML

6. State your opinions reasons/guesses for your decisions along the way.