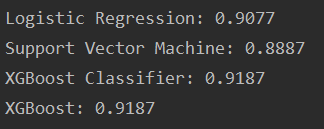
**Machine Learning Term Project**

**(KimGyunYop, ChoiHyungKyu)**

1. **Classification: Bank Marketing**

* Data Inspection(‘EDA\_\*.py): whether there is null data, outliers, correlation
* Preprocess(‘prep\_data.py’): Remove rows which have unknown data(dirty data)
* Scale(‘prep\_scale.py’): Scaling data with 4-Scalers
* Classification, Evaluation(‘bank\_\*.py’): Finding hyper-parameter with GridSearchCV
* Ensemble(‘bank\_xgb.py’): XGBoosting with SVM, Logistic, XGBC
* **Best Parameter (10-Fold Cross Validation)**
  + *Logistic Regression: {max\_iter: 50, solver: ‘liblinear’}*
    - Score: 0.9077
  + *Support Vector Machine: {C: 0.1, gamma: 1, kernel: ‘sigmoid’}*
    - Score: 0.8887
  + *XGBC: {eta: 0.10286751649448647, gamma: 1.5458648766632133, max\_depth: 5}*
    - Score: 0.9187
  + *XGBoost: {eta: 0.5243, gamma: 1.9861, max\_depth: 4}*
    - Score: 0.9187
* **Output:**



1. **Clustering: World Development Indicators**

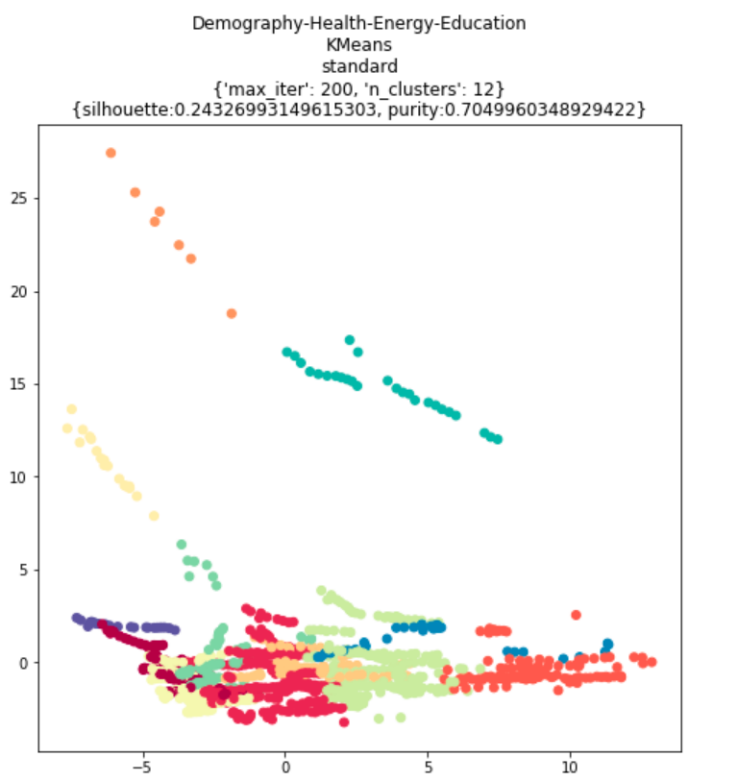
* Additional Dataset Generator(‘income\_level.py’): Using OGHIST.xls, make income-level
* Data Inspection, Preprocess(‘keyword.py’): Divide keywords and preprocess data
* Clustering, Evaluation(‘TP\_2.py’): Finding hyper-parameter with GridSearchCV
* Visualization(‘visualization.py’): Make plot to display clustering using PCA
* **Best Parameter (5-Fold cross Validation)**
  + *kMeans: {'max\_iter': 200, 'n\_clusters': 4}*
    - Score: 0.6651311707875324
    - Purity: 0.3409373703857321
  + *EM: {'covariance\_type': 'spherical', 'max\_iter': 100, 'n\_components': 4}*
    - Score: 0.7305235299240406
    - Purity: 0.3434259643301535
  + *DBSCAN: {'eps': 0.001, 'min\_samples': 150}*
    - Score: 0.0
    - Purity: 0.3076659822039699
* **Evaluation (2-Methods)**
  + *Silhouette Method*

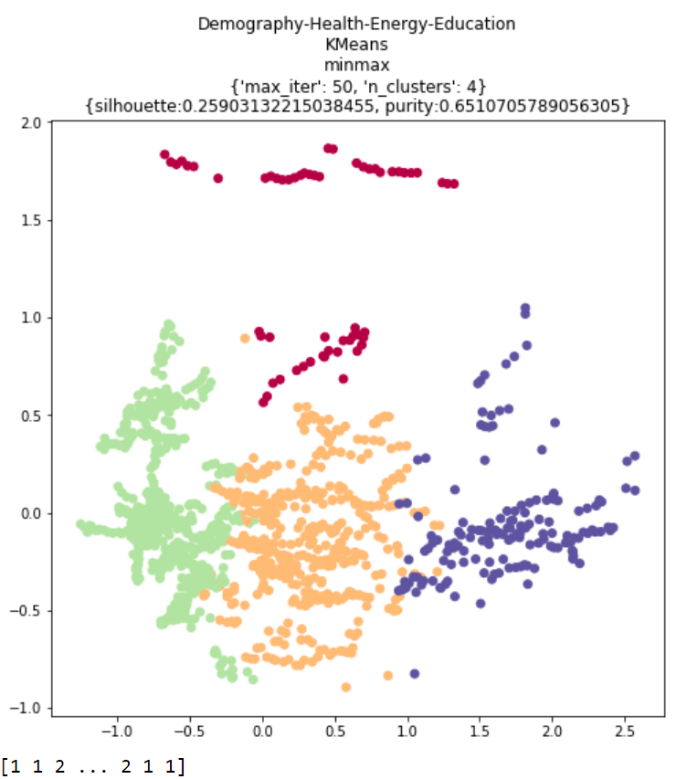
{"Demography": {"ds": 0.0, "km": 0.5755514231579936, "gm": 0.5672940101624234}, "Health": {"ds": 0.5769425480032045, "km": 0.43658014019695995, "gm": 0.41188026853679355}, "Energy": {"ds": 0.0, "km": 0.6651311707875324, "gm": 0.7305235299240406}, "Education": {"ds": 0.0, "km": 0.44632295183813814, "gm": 0.4497713701830475}, "Demography-Health": {"ds": 0.0, "km": 0.5642224985679525, "gm": 0.4366755631526042}, "Demography-Energy": {"ds": 0.0, "km": 0.8295922633444772, "gm": 0.8038017978619113}, "Demography-Education": {"ds": 0.0, "km": 0.3354190688955006, "gm": 0.42250517043976776}, "Health-Energy": {"ds": 0.0, "km": 0.6553873946528547, "gm": 0.7264392355627347}, "Health-Education": {"ds": 0.0, "km": 0.3980359896764172, "gm": 0.41088904739761734}, "Energy-Education": {"ds": 0.0, "km": 0.7590177296339042, "gm": 0.7049477567928051}, "Demography-Health-Energy": {"ds": 0.0, "km": 0.8273898990356484, "gm": 0.8347884841249281}, "Demography-Health-Education": {"ds": 0.0, "km": 0.35398905048862694, "gm": 0.403411124897844}, "Demography-Energy-Education": {"ds": 0.0, "km": 0.8103784940707868, "gm": 0.7561210506242592}, "Health-Energy-Education": {"ds": 0.0, "km": 0.7175715001033406, "gm": 0.731334988624163}, "Demography-Health-Energy-Education": {"ds": 0.0, "km": 0.8016807638293157, "gm": 0.804488192670291}}

* + *Purity Method*

{"Demography": {"ds": 0.29858183211958605, "km": 0.3085473361441165, "gm": 0.30126485243388273}, "Health": {"ds": 0.33463490823896913, "km": 0.6046466224131198, "gm": 0.582194455290902}, "Energy": {"ds": 0.33678971381169637, "km": 0.3409373703857321, "gm": 0.3434259643301535}, "Education": {"ds": 0.3076659822039699, "km": 0.32306639288158795, "gm": 0.3261464750171116}, "Demography-Health": {"ds": 0.29858183211958605, "km": 0.3085473361441165, "gm": 0.49329244921425836}, "Demography-Energy": {"ds": 0.3194062671797691, "km": 0.3446948873007147, "gm": 0.3402968664101155}, "Demography-Education": {"ds": 0.28057971014492755, "km": 0.2956521739130435, "gm": 0.296231884057971}, "Health-Energy": {"ds": 0.3238660449342942, "km": 0.3302246714709623, "gm": 0.32895294616362863}, "Health-Education": {"ds": 0.27660363366703744, "km": 0.28513162773451983, "gm": 0.28735632183908044}, "Energy-Education": {"ds": 0.4001171646162859, "km": 0.4141769185705917, "gm": 0.4065612185120094}, "Demography-Health-Energy": {"ds": 0.3194062671797691, "km": 0.34634414513468936, "gm": 0.3441451346893898}, "Demography-Health-Education": {"ds": 0.28057971014492755, "km": 0.296231884057971, "gm": 0.5008695652173913}, "Demography-Energy-Education": {"ds": 0.3655828707375099, "km": 0.3877874702616971, "gm": 0.38540840602696275}, "Health-Energy-Education": {"ds": 0.38591210114388924, "km": 0.4003612281757977, "gm": 0.4003612281757977}, "Demography-Health-Energy-Education": {"ds": 0.3655828707375099, "km": 0.3877874702616971, "gm": 0.3877874702616971}}

* **Output(sample cluster, dimension reduction):**





1. **Introduction of data, directories**

* Please read ‘Readme.md’ files
* **If you want to see visualizations for cluster, please open ‘ipython.html’ file.**
* <https://github.com/ChoiHyungKyu/MachineLearningTermProject>