[AI18] Assignment 6

Due: Dec 10 (before class)

You are given a data set of community crime rate[1]. The original data set contains 1993 communities, each described by 128 features and labeled by its rate of violent criminal events. We have preprocessed the data set for you, by removing features that are missing in many instances and then instances that have many missing features; categorical features are encoded by dummy variables using embedded functions in Python; label is the crime rate – it is originally continuous and we binarized it so that rates above or equal to 0.5 are rounded to 0 (indicating the community has high crime rate) and rates below 0.5 are rounded to (indicating the community has low crime rate).

You will see a data matrix after loading `crimerate.csv` in Python. In that matrix, each row represents one community and each column represents one feature of the community; the last column is the label of continuous crime rate, and the last second column is the binarized label; the first column is a binary feature indicating whether a community is minority or not (1 means it is a minority community and 0 means otherwise) – we will use it when measuring fairness of model prediction.

You are asked to apply different machine learning algorithms on the data set and report their performance. Call functions of different learning algorithms from the SciKit-Learn libraries, using the given templates.

[1] Test different regression methods using template `hw6_regression.py` and report their MSEs in Table 1.

Table 1. Mean-Squared-Errors of Different Regression Methods

Method	**Mean-Squared-Error**
Linear Regression (Least Square)	0.021810923337718426
Ridge Regression	0.017709199559113896
Lasso	0.01765686963708134
Kernel Ridge Regression	017183850065595464

[2] Test various classification methods using template `hw6_classification.py` and report errors in Table 2.

Table 2. Classification Errors and F1 Scores of Different Classification Methods

Method	**Classification Error**	**F1 Score**
Naive Bayes	0.16075016744809112	0.5918367346938775
Logistic Regression	0.09310113864701941	0.5669781931464175
Linear Discriminant Analysis	0.116543871399866	0.6009174311926606
SVM	0.09711989283322175	0.5747800586510263
k Nearest Neighbor	0.10180843938379103	0.525
Neural Network	0.09979906229068991	0.639225181598063
Decision Tree	0.17012726054922978	0.4773662551440329
Random Forest	0.09310113864701941	0.6191780821917807
AdaBoost	0.11587407903549896	0.6022988505747127

[1] The original data set is available at http://archive.ics.uci.edu/ml/datasets/communities+and+crime

[2.1] Report the group fairness (GP) of the prediction of a logistic regression model f in Table 3, where

$$GF(f) = \frac{P(f(x) = 1 \mid x \in \text{minority})}{P(f(x) = 1 \mid x \in \text{majority})}, \tag{1}$$

where $f(x)$ is the model prediction of x, and $P(f(x) = 1|x \in \text{minority})$ is the probability that a minority community is predicted as high risk.

Table 3. Group Fairness of Logistic Regression Prediction

Method	GF(f)
Logistic Regression	0.041239316239316245

[3] Apply Kmeans and GMM methods to cluster examples, and visualize the clustering result of Kmeans in a two-dimensional feature space reduced by PCA, using template `hw6_other.py`.

[3.1] Report clustering results of the two methods based on the Adjusted Mutual Information Score (AMIS) metric in Table 4. (The higher the better.)

Table 4. Clustering Performance of Two Methods

Method	**AMIS**
Kmeans	0.10564170258089178
GMM	0.13846424092507953

[3.2] Plot Kmeans clustering results with k = 2 and k = 3 separately. Your two figures should look like Figure 1 (but do not need to be the same because Kmeans may give different results every time).

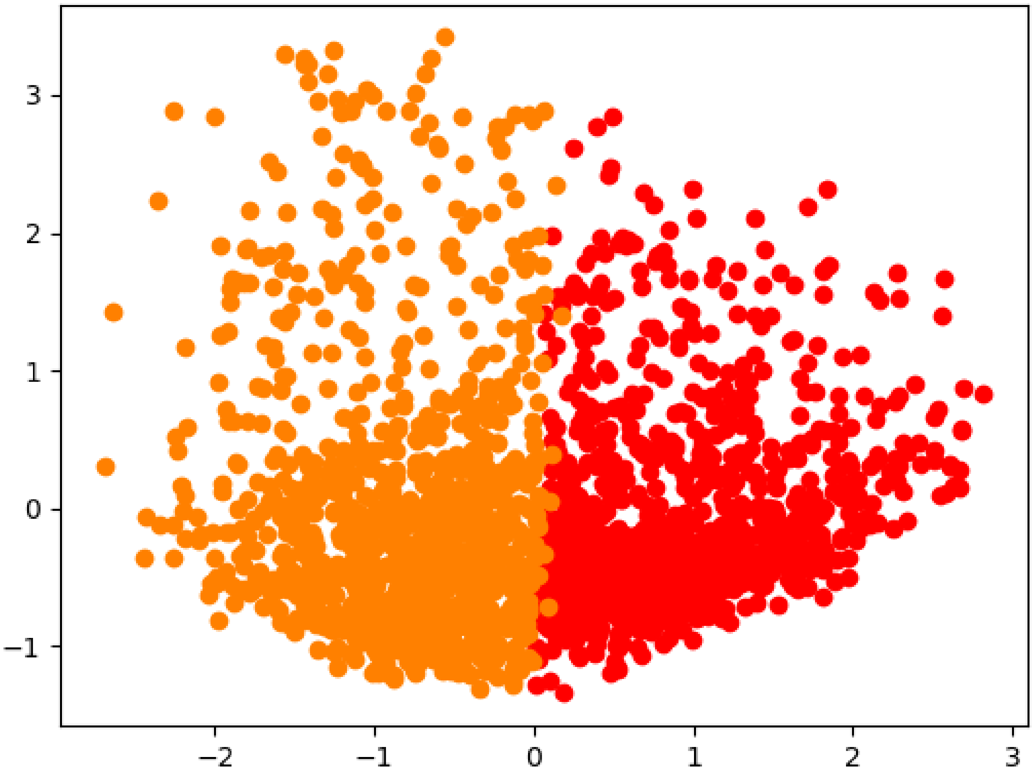

Fig. 1. Clustering Result of Kmeans with K = 2

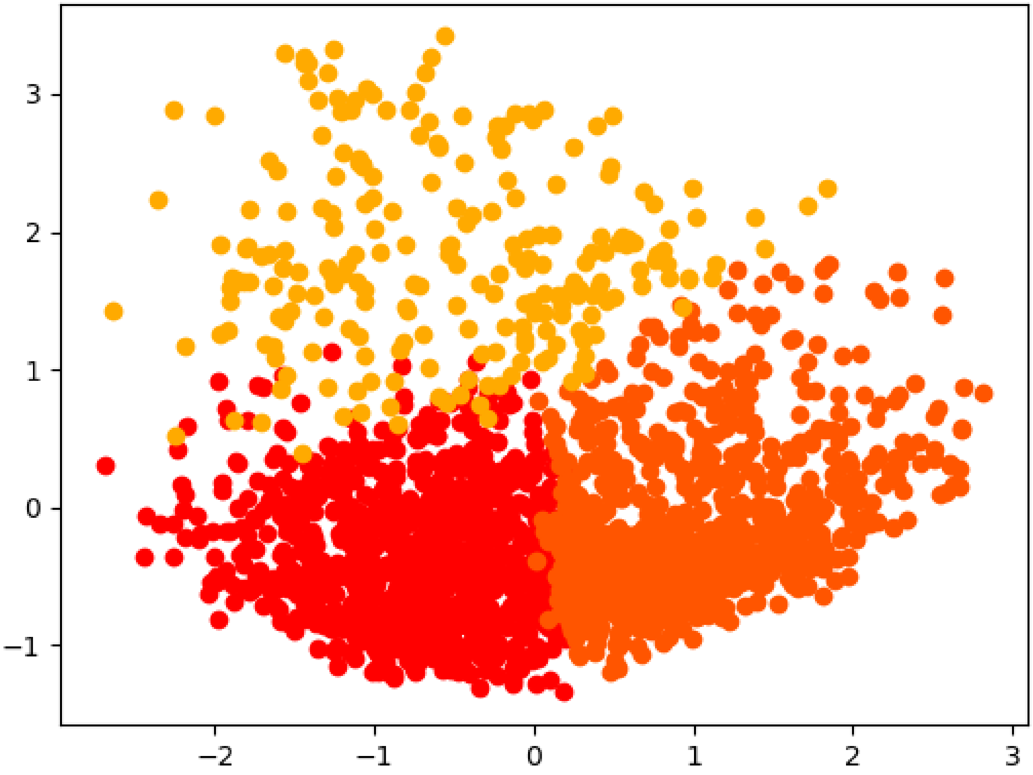

Fig. 2. Clustering Result of Kmeans with K = 3