

Compiler Environment: PyCharm(python 3.8), MacBook pro(M1)

There are 13 files in the “code” folder, 2 are csv files and 11 are python files

2 csv files: train.csv, test.csv, test<SVM>.csv

- **train.csv:** given training dataset
- **test.csv:** given test dataset

11 python files: KNN.py, Bayes.py, Perceptron.py, DecisionTree.py, LogisticRegression.py, RandomForest.py, GBDT.py, EnsembleModel.py, SVM.py, Score.py, Prediction.py

- **KNN.py:** implement the KNN model by myself and running the code will print the score of KNN model
- **Bayes.py:** implement the Bayes model by myself and running the code will print the score of Bayes model
- **Perceptron.py:** implement the Perceptron Model by myself and running the code will print the current number of iteration times and score of Perceptron
- **DecisionTree.py:** implement the Decision Tree by myself and running the code will print the score of Decision Tree
- **LogisticRegression.py:** import the Logistic Regression Model from library and running the code will print the score of Logistic Regression Model
- **RandomForest.py:** import the Random Forest Model from library and running the code will print the best hyperparameters through grid research and the score of Random Forest Model
- **GBDT.py:** import the Gradient Boosting Decision Tree Model from library and running the code will print the best hyperparameters through grid research and the score of GBDT Model
- **EnsembleModel.py:** contains a function “model_ensemble”to construct the ensemble model and running the code will print the score of the ensemble model
- **SVM.py:** import the SVM model from library and running the code will print the 5 times of score of SVM model after 5-fold
- **Score.py:** contains a function score to calculate and print the 4 indicators: accuracy, precise, recall and f1_score as a dictionary
- **Prediction.py:** use SVM model as the final model to predict the test dataset and create a new test<SVM>.csv