Report of Project 1

1. Introduction

This program provides a method for you to analyze the winners of a set of LOL games by giving some information of these games such as First Baron, dragon, tower, blood. Moreover, it can show the accuracy of its prediction. It require time library to record the spending time of each classifier, pandas library to read csv files, sklearn.classifier libraries to train and predict and sklearn.metrics library to calculate accuracy.

In this program, I use training set to train the classifier including decision tree, K-NN, SVM and MLP. Then use these classifier to analyze test set and determine the winner. Finally, it calculates the accuracy comparing with the true labels.

1. Algorithms
   * 1. Decision tree

Select the best feature (A). For each value of A, create new descendant of node. Sort training samples to leaf nodes. Stop when training samples perfectly classified.

Parameter: default

* + 1. K-NN

Classified by a majority vote of its k nearest neighbors. n distances are calculated for each new sample(n: the number of training samples).

Parameter: default

* + 1. SVM

A classifier with the largest margin should be the best classifier. Problem can be formulated as Quadratic Optimization Problem and solve for w and b.

Parameter: default

* + 1. MLP

Neurons are arranged in layers. A neuron is connected to all neurons in next layer. Neurons may have different activation functions or no activation function.

Two training methods: 

* + - 1. Batch training 

One update, all samples 

Sample sequence will not affect 

* + - 1. Stochastic training 

One update, one sample 

Samples can be chosen randomly to avoid the influence of the sample sequence

Parameter: max\_iter=20000

1. Requirements

import pandas as pd

import time

from sklearn.metrics import accuracy\_score

from sklearn.tree import DecisionTreeClassifier

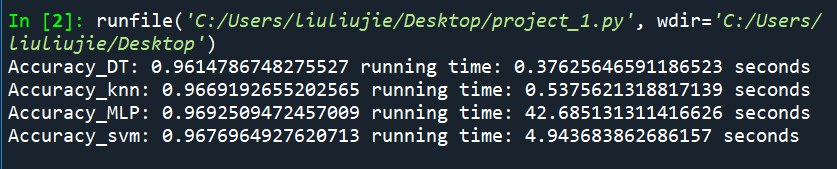
from sklearn.neighbors import KNeighborsClassifier

from sklearn.neural\_network import MLPClassifier

from sklearn.svm import SVC

1. Result

|  |  |  |
| --- | --- | --- |
| classifier | accuracy | Running time |
| Decision tree | 0.961 | 0.376 |
| K-NN | 0.967 | 0.538 |
| MLP | 0.970 | 42.685 |
| SVM | 0.968 | 4.944 |



1. Comparison and Discussion

Above all, accuracy of the four classifiers are almost the same. Their accuracy is so high that there is almost no difference between them. Maybe it is because I don’t set parameters for them. As a result, many constraints are not considered or the number of attributes is large so that it is easy to analyze the winner from the information. However, their training time discrepancy are very large. Decision tree, SVM and K-NN are training relatively rapidly, whereas MLP takes lots of time to train, because it needs to update its weight again and again and the training set is big. If more time is allowed, I think I can write a program to find the best parameters of each classifier to maximize the accuracy.