# Machine Learning of Lab 1

Group:28

Author: Yi Li, Wenxuan Zhu, Xiaofei Li, Qitao Ye, JinShuai Chang

[Machine Learning of Lab 1 0](#_Toc118617538)

[Introduction 2](#_Toc118617539)

[description of methods 3](#_Toc118617540)

[Linear regression/ Logistic regression 3](#_Toc118617541)

[Classification 3](#_Toc118617542)

[regression 3](#_Toc118617543)

[SVM 3](#_Toc118617544)

[Classification 3](#_Toc118617545)

[regression 3](#_Toc118617546)

[DT 3](#_Toc118617547)

[Classification 3](#_Toc118617548)

[regression 3](#_Toc118617549)

[Neutral Network 3](#_Toc118617550)

[Classification 4](#_Toc118617551)

[regression 6](#_Toc118617552)

# Introduction

Technology is changing the society especially lifestyle of people with the development of the machine learning (ML). Now this technology has applied to many aspects including face recognition, voice recognition, medical treatment and market predicting. Essential functions of ML attribute to classification and regression. Besides, it has libraries which integrate many effective methods, such as KNN, SVM, DT, Neural network.

In this review, we discuss four methods containing linear regression, logistic regression, SVM, DT, MLP to predict classification and do regression and calculate accuracy based on the Iris and computer hardware dataset. In addition, we also use K-fold cross validation to evaluate every method.

In conclusion, most methods have nearly performance on classification, however, some methods like DT don’t get a satisfying result on regression. And different params also bring quite different predict results especially when we use different activation method and hidden layers on MLP.

# description of methods

## Linear regression/ Logistic regression

### Classification

### regression

## SVM

### Classification

### regression

## DT

### Classification

### regression

## Neutral Network

MLP (Multilayer Perceptron), referred to ANN (Artificial Neural Network), which has three-layer structure: input, hidden and output. Hidden layer could have more than one layer. From input layer to hidden layer is a fully connected layer and output of the hidden layer is f (W1X+b1), from hidden layer to output layer is a classifier softmax regression and the result of output layer is softmax(W2X1+b2).

### Classification

#### Parameter setting:

We should install sklearn, numpy, matplotlib in advance to accomplish MLPClassifier test.

Parameters of MLPClassifier:

hidden\_​​layer\_sizes: tuple, define amount of the hidden layers and total amount of each hidden layer.

Activation: activation function, we could select one of the list : {‘identity’，‘logistic’，‘tanh’，‘relu’}

Solver: solver for weight optimization, this test I use ‘sgd’, it means gradient descent.

Learning\_rate\_init: control step size for updating weights.

Max\_iter: max iter times, to convergence iterator.

#### evaluation method:

Based on the Iris dataset, I use accuracy\_score to evaluate the predict results. That is, after training the model, I compare the test data with the predicting result, and calculate the result. Besides, I also use K-fold cross validation to make the result more precise by avoiding the inaccurate result due to single partition imbalance.

#### result:

Train accuracy\_score: 0.983

Test accuracy\_score: 0.98

![图表, 散点图

描述已自动生成]()

#### conclusion:

MLPClassifier could reach a better result by increasing the hidden layer to 10 and selecting logistic as activation function.

### regression

#### Parameter setting:

We should install pandas, sklean, numpy in advance

Parameters of MLPRegressor:

Hidden\_layer\_sizes:(10,) ten hidden layers.

activation: relu

solver: adam refers to a stochastic gradient-based optimizer

#### evaluation method:

Firstly, I standard the data and transform the string feature to integer. Then Based on the transformed dataset of computer hardware, I use mean\_square\_error to evaluate the precise of the model. Besides, I also use K-fold cross validation to make the result more precise by avoiding the inaccurate result due to single partition imbalance.

#### result:

train data MSE : 0.013

test data MSE: 0.010

![图表, 散点图

描述已自动生成]()

#### conclusion:

The result of MLPRegressor could get better by increasing the iterators based on the parameter of relu, adam and 10 hidden layers.