# Introduction to Machine Learning
# Homework 3

151250048, 郭浩滨, 151250048@smail.nju.edu.cn

Released on April 11, 2019

Due at 20:00, April 17, 2019

## 1 [15pts] Decision Tree I

(1) [5pts] Assume there is a space contains three binary features $X$, $Y$, $Z$ and the objective function is $f(x, y, z) = \neg(x \text{ XOR } y)$. Let $H$ denotes the decision tree constructed by these three features. Please answer the following question: Is function $f$ realizable? If the answer is yes, please draw the decision tree $H$ otherwise please give the reason.

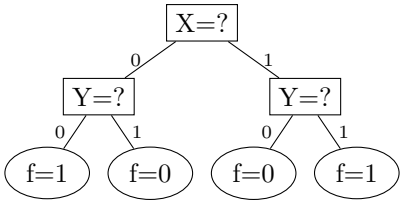
(2) [10pts] Now we have a dataset show by Table.1:

Table 1:example dataset

| $X$ | $Y$ | $Z$ | $f$ |
|---|---|---|---|
| 1 | 0 | 1 | 1 |
| 1 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 |
| 1 | 0 | 1 | 1 |
| 0 | 0 | 1 | 0 |
| 0 | 1 | 1 | 1 |
| 1 | 1 | 1 | 0 |

Please use Gini value as partition criterion to draw the decision tree from the dataset. When Gini value is same for two features, please follow the alphabetical order.

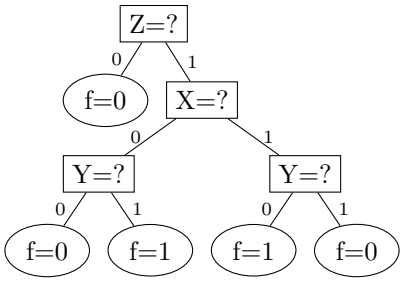
(1) 函数 f 是可以实现的，其决策树结构如下：




(2) 设总数据集为 $D$，分别对属性 $x, y, z$ 求第一次划分后的基尼指数，可得：

$$\text{Gini\_index}(D, x) = \frac{4}{8} * (1 - ((\frac{2}{4})^2 + (\frac{2}{4})^2)) + \frac{4}{8} * (1 - ((\frac{2}{4})^2 + (\frac{2}{4})^2)) = \frac{1}{2}$$
$$\text{Gini\_index}(D, y) = \frac{4}{8} * (1 - ((\frac{2}{4})^2 + (\frac{2}{4})^2)) + \frac{4}{8} * (1 - ((\frac{2}{4})^2 + (\frac{2}{4})^2)) = \frac{1}{2}$$
$$\text{Gini\_index}(D, z) = \frac{2}{8} * (1 - (1 * 1)) + \frac{6}{8} * (1 - ((\frac{2}{6})^2 + (\frac{4}{6})^2)) = \frac{1}{3}$$

选取划分后最小基尼指数对应的属性，即 z，作为第一次划分的属性。同理，对划分后的子数据集 $D_1$，$D_2$ 选取最优划分属性并进行划分，直至不能或无需再划分为止。最终的决策树如下：



## 2 [25pts] Decision Tree

Consider the following matrix:

$$\begin{bmatrix} 24 & 53 & 23 & 25 & 32 & 52 & 22 & 43 & 52 & 48 \\ 40 & 52 & 25 & 77 & 48 & 110 & 38 & 44 & 27 & 65 \end{bmatrix}$$

which contains 10 examples and each example contains two features $x_1$ and $x_2$. The corresponding label of these 10 examples as follows:

$$\begin{bmatrix} 1 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 1 \end{bmatrix}$$

In this problem, we want to build a decision tree to do the classification task.

(1) [5pts] Calculate the entropy of the root node.

(2) [10pts] Building your decision tree. What is your split rule and the classification error?

(3) [10pts] A multivariate decision tree is a generalization of univariate decision trees, where more than one attribute can be used in the decision for each split. That is, the split need not be orthogonal to a feature's axis.

Building a multivariate decision tree where each decision rule is a linear classifier that makes decisions based on the sign of $\alpha x_1 + \beta x_2 - 1$. What is the depth of your tree, as well as $\alpha$ and $\beta$?

(1) 根节点的信息熵为：

$$\text{Ent}(D) = -(\frac{4}{10} * \log \frac{4}{10} + \frac{6}{10} * \log \frac{6}{10}) \approx 0.673 \tag{2.1}$$

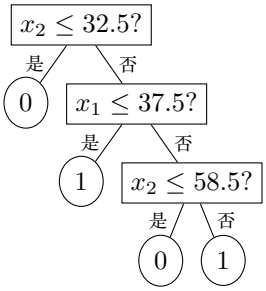
(2) 由于属性 $x_1, x_2$ 可取值数目一致，所以我们可以用信息增益作为选取划分属性的准则。

对于属性 $x_1$，我们先对其进行排序并按相邻数的中位点作为划分点，得出 9 个候选值为：

$$T_{x_1} = \{22.5, 23.5, 24.5, 28.5, 37.5, 45.5, 50, 52, 52.5\}$$

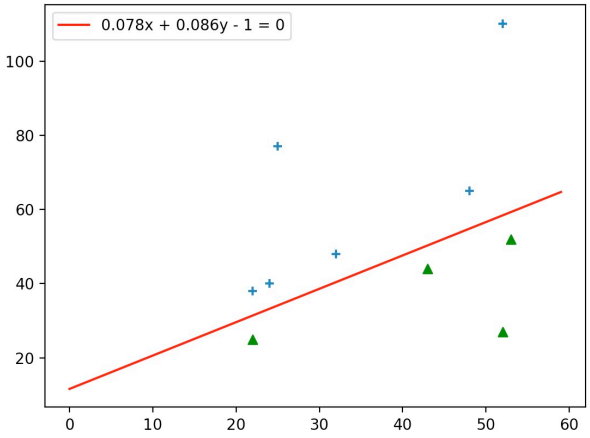
分别对各个候选值计算二分后的信息增益，并选取出信息增益最大的划分点，可得取到最大信息增益划分点为 52，其信息增益为 0.223。

对于属性 $x_2$，同理我们课得出取得最大信息增益划分点为 32.5，其信息增益同样为 0.223。我们可取 $x_2 \leq 32.5$ 作为根节点划分属性。对于子节点，依次类推，最终可以得到决策树为：

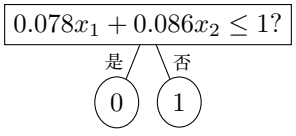



该决策树对所有样本划分均正确，即分类误差为 0。

(3) 将 $x_1, x_2$ 及对应的标签在坐标系上画出，可知用一条直线即可将其划分为两个类，再通过 SVM 算法找出该分界线，如图所示：




对应的决策树为：




此时 $\alpha = -0.078$，$\beta = 0.086$，决策树的深度为 1。

## 3 [25pts] Convolutional Neural Networks

Figure 1: CNN.

Using Fig. 1 as an example. Assuming that the loss function of the convolutional neural network is cross-entropy:

(1) [5 pts] Briefly describe the forward propagation process;

(2) [5 pts] What is the difference between Relu and Sigmoid activation functions;

(3) [5 pts] Derivation of the fully connected layer;

(4) [5 pts] Derivation of the pooling layer with average pooling;

(5) [5 pts] Derivation of the convolutional layer with Relu;

## 4 [35 pts] Neural Network in Practice

In this task, you are asked to build a Convolutional Neural Networks (CNNs) from scratch and examine performance of the network you just build on **MNIST** dataset. Fortunately, there are some out-of-the-box deep learning tools that can help you get started very quickly. For this task, we would like to ask you to work with the **Pytorch** deep learning framework. Additionally, Pytorch comes with a built-in dataset class for MNIST digit classification task in the **torchvision** package, including a training set and a validation set. You may find a pytorch

introduction at here. Note that, you can use CPU or GPU for training at your choice.

Please find the detailed requirements below.

(1) [5 pts] You are encouraged to implement the code using *Python3*, implementations in any other programming language will not be judged. Please name the source file (which contains the main function) as *CNN_main.py*. Finally, your code needs to print the performance on the provided validation set once executed.

(2) [10 pts] Use any type of CNNs as you want and draw graphs to show your network architecture in the submitted report. You are encouraged to try more architectures.

(3) [15 pts] During training, you may want to try some different optimization algorithms, such as SGD, Adam. Also, you need to study the effect of learning rate and the number of epoch, on the performance (accuracy).

(4) [5 pts] Plot graphs (learning curves) to demonstrate the change of training loss as well as the validation loss during training.