CS57800 Statistical Machine Learning

Homework 1

Ting Zhang
School of Industrial Engineering
zhan1013@purdue.edu

September 15, 2015

1 Foundations

(1) The normal vectors of the two planes are $a = (1, 1, 3)$ and $b = (1, 2, 4)$. The vector that is parallel to the intersection line can be computed as:

$$\begin{aligned} l' &= a * b = (1 * 4 - 3 * 2, 3 * 1 - 1 * 4, 1 * 2 - 1 * 1) \\ &= (-2, -1, 1) \end{aligned}$$

Then we need to find a point P, which is on both planes. For example, we can solve the following system with $x_3 = 0$ to get a particular P.

$$\begin{aligned} x_1 + x_2 &= 4 \\ x_1 + 2x_2 &= 5 \end{aligned}$$

So, we get $x_1 = 3$ and $x_2 = 1$. Therefore, $P = (3, 1, 0)$. Then the intersection line can be represented as:

$$\begin{aligned} x_1 &= 3 - 2t \\ x_2 &= 1 - t \\ x_3 &= t \end{aligned}$$

(2) Let's first get the two lines formed by the three points.

$$\begin{aligned} \vec{a} &= PQ = (1, -1, 1) - (0, 0, 0) = (1, -1, 1) \\ \vec{b} &= PR = (4, 3, 7) - (0, 0, 0) = (4, 3, 7) \end{aligned}$$

Then, the vector orthogonal to the plane of P, Q and R can be computed following:

$$\begin{aligned}\vec{a}\times\vec{b} &= \begin{vmatrix} i & j & k \\ 1 & -1 & 1 \\ 4 & 4 & 7 \end{vmatrix} \\ &= \begin{vmatrix} -1 & 1 \\ 3 & 7 \end{vmatrix} i - \begin{vmatrix} 1 & 1 \\ 4 & 7 \end{vmatrix} j + \begin{vmatrix} 1 & -1 \\ 4 & 3 \end{vmatrix} k \\ &= -10i - 3j + 7k\end{aligned}$$

Therefore, the vector orthogonal to the plane of P, Q and R is $\vec{v} = (-10, -3, 7)$.

(3) (a)

$$\begin{aligned}\frac{df(x)}{dx} &= (x^{\frac{1}{2}})\frac{d(3x^2)}{dx} + (3x^2)(\frac{dx^{\frac{1}{2}}}{dx}) \\ &= x^{\frac{1}{2}}6x + (3x^2)(\frac{1}{2}x^{-\frac{1}{2}}) \\ &= 6x^{\frac{3}{2}} + \frac{3}{2}x^{\frac{3}{2}} \\ &= \frac{15}{2}x^{\frac{3}{2}}\end{aligned}$$

(b)

$$\begin{aligned}\frac{df(x)}{dx} &= \frac{d(e^{2x}+e)^{\frac{1}{2}}}{d(e^{2x}+e)}\frac{d(e^{2x}+e)}{dx} \\ &= \frac{1}{2}(e^{2x}+e)^{-\frac{1}{2}}(2e^{2x}) \\ &= e^{2x}(e^{2x}+e)^{-\frac{1}{2}}\end{aligned}$$

(c)

$$\begin{aligned}\frac{df(x)}{dx} &= \frac{d([\ln(5x^2+9)]^3)}{d(\ln(5x^2+9))}\frac{d(\ln(5x^2+9)}{d(5x^2+9)}\frac{d(5x^2+9)}{dx} \\ &= 3\ln(5x^2+9)^2\frac{1}{5x^2+9}(10x) \\ &= \frac{30x\ln(5x^2+9)^2}{5x^2+9}\end{aligned}$$

(4) (a)

$$\begin{aligned}\frac{\partial f(x,y)}{\partial x} &= y^3 + 2xy^2 \\ \frac{\partial f(x,y)}{\partial y} &= 3xy^2 + 2x^2y\end{aligned}$$

(b)

$$\begin{aligned}\frac{\partial f(x,y)}{\partial x} &= \frac{\partial x}{\partial x}e^{2x+3y} + x\frac{\partial(e^{2x+3y})}{\partial x} \\ &= e^{2x+3y} + 2x(e^{2x+3y}) \\ &= (2x+1)e^{2x+3y}\end{aligned}$$

$$\frac{\partial f(x,y)}{\partial y} = 3xe^{2x+3y}$$

(5) To compare the grow speed of two function, we can compute the limit of their ratio when n goes to infinity, shown in Eq. (1). If the limit is infinity, then $f(x)$ grows faster than $g(x)$; and 0, otherwise.

$$\lim_{x\to\infty} \frac{f(x)}{g(x)} \tag{1}$$

Therefore, we perform pairwise comparsions to the functions listed in the question.

- compare $\log^4 \sqrt{n}$ and $2^{\log_2 n}$

  $$\lim_{n\to\infty} \frac{2^{\log_2 n}}{\log^4 \sqrt{n}} = \lim_{n\to\infty} \frac{n}{\frac{1}{16}\log^4 n} = \lim_{n\to\infty} \frac{16n}{4\log^3 n} = \lim_{n\to\infty} \frac{4n}{3\log^2 n} = \lim_{n\to\infty} \frac{4n}{6\log^n} = \lim_{n\to\infty} \frac{4n}{6} = +\infty$$

  Therefore, $\log^4 \sqrt{n} \prec 2^{\log_2 n}$

- compare $2^{\log_2 n}$ and $n^{\frac{3}{2}} \log^2 n$

  $$\lim_{n\to\infty} \frac{n^{\frac{3}{2}} \log^2 n}{2^{\log_2 n}} = \lim_{n\to\infty} \frac{n^{\frac{3}{2}} \log^2 n}{n} = \lim_{n\to\infty} n^{\frac{1}{2}} \log^2 n = +\infty$$

  Therefore, $2^{\log_2 n} \prec n^{\frac{3}{2}} \log^2 n$

- compare $n^{\frac{3}{2}} \log^2 n$ and $2^{3\log_2 n}$

  $$\lim_{n\to\infty} \frac{2^{3\log_2 n}}{n^{\frac{3}{2}} \log^2 n} = \lim_{n\to\infty} \frac{n^3}{n^{\frac{3}{2}} \log^2 n} = \lim_{n\to\infty} \frac{n^{\frac{3}{2}}}{\log^2 n} = \lim_{n\to\infty} \frac{3n^{\frac{3}{2}}}{4\log n} = \lim_{n\to\infty} \frac{9n^{\frac{3}{2}}}{8} = +\infty$$

  Therefore, $n^{\frac{3}{2}} \log^2 n \prec 2^{3\log_2 n}$

- compare $2^{3\log_2 n}$ and 2^n

  $$\lim_{n\to\infty} \frac{2^{3\log_2 n}}{2^n} = \lim_{n\to\infty} \frac{n^3}{2^n} = \lim_{n\to\infty} \frac{3n^2}{2^n \log 2} = \lim_{n\to\infty} \frac{6}{2^n \log^3 2} = 0$$

  Therefore, $2^{3\log_2 n} \prec 2^n$

- compare 2^n and $\frac{5}{3}^{2n}$

  $$\lim_{n\to\infty} \frac{\frac{5}{3}^{2n}}{2^n} = \lim_{n\to\infty} \frac{(\frac{25}{9})^n}{2^n} = \lim_{n\to\infty} (\frac{25}{18})^n = +\infty$$

  Therefore, we have $2^n \prec \frac{5}{3}^{2n}$

Since 10^8 does not grow, we can have a sequence shown below:
$10^8 \prec \log^4 \sqrt{n} \prec 2^{\log_2 n} \prec n^{\frac{3}{2}} \log^2 n \prec 2^{3\log_2 n} \prec 2^n \prec \frac{5}{3}^{2n}$

(6) (a) Define the three rolls as X_1, X_2 and X_3. We can compute the expected value for each roll as

$$E(X_i) = \frac{1}{6}\sum_{j=1}^{6} j = \frac{7}{2} \quad \text{for} \quad i = 1, 2, 3$$

Therefore, the expected value of the sum of the rolls would be

$$E(\sum_{i=1}^{3} X_i) = \frac{21}{2}$$

(b) To compute the expected value of the product of the rolls, we can simply multiply their individual expected value since these rolls are independent.

$$E(X_1X_2X_3) = E(X_1)E(X_2)E(X_3) = (\frac{7}{2})^3 = \frac{343}{8}$$

(c) The variance of one roll can be computed as

$$V(X_i) = E(X_i^2) - E(X_i)^2 = \frac{1}{6}\sum_{j=1}^{6} j^2 - (\frac{7}{2})^2 = \frac{35}{12}$$

Therefore, the variance of the sum of rolls is

$$V(\sum_{i=1}^{3} X_i) = \frac{35}{4}$$

2 Programming Report

In this task, a decision tree was implemented to classify the type of growth of breast cancer. A basic decision tree algorithm ID_3 was implemented, with two different strategies dealing with overfitting. The two strategies including: (1) statically fix the depth of the tree, and (2) post prune the tree after the full tree is established following algorithm ID_3.

2.1 Decision Tree Construction

To build a decision tree classifier, we need to first train a decision tree based on training dataset, tune hyper-parameters using validation set, and at last, build a classifier using the constructed decision tree. In the training algorithm, information gain was computed to select the most salient attribute. To compute the informaiton gain, a subset of the original training dataset was extracted based on attributes and their values. The scheme of majority votes was applied when the last available attribute can't determine the type explicitly. While constructing the decision tree, the label of majority votes for every node is computed and applied later in the classification algorithm when there is no available label.
To avoid overfitting, two strategies were applied:

(1) Fixed depth of decision tree
Shorter trees are more preferred than deep trees, since they may be more generalizable.

Therefore, the depth of trees were limited in this approach. While training a decision tree, the algorithm stops and build leaves when it reached the maximum depth. Majority votes were used to decide the label. The depth of a decision tree is then becoming a hyper-parameter and can be tuned using a validation set (see explanations in next section).

(2) Post Pruning
The other approach applied to avoid overfitting is post pruning. A full tree was first built using the basic decision tree algorithm. A pruning algorithm was then applied for each non-leaf node from the bottom of a tree to the root. Each non-leaf node was substituted with the label of the majority votes. The prunine process stoped when the accuracy of validation set decreases.

2.2 Experiments and Results

Experiments were conducted following three scenarios: (1) the basic decision tree, (2) decision tree with fixed depth, and (3) decision tree with post pruning. The results of each experiment are presented below. Table 1 summarizes the results of three strategies, including the depth of the tree, and the accuracy of training, validating and testing dataset.

Table 1: Summary of depth and accuracies of different approaches.

	Depth	Accuracy		
		Training	Validation	Testing
ID_3	8	0.983	0.85	0.95
Fixed Depth	3	0.933	0.85	0.95
Pruning	3	0.917	0.85	0.95

(1) Basic decision tree
Following the ID_3 algorithm, the resultant decision tree is of depth 8. The accuracy of training, validating and testing dataset are: 0.983, 0.80 and 0.95.

(2) Decision tree with fixed depth
In this experiment, hyper-parameter *depth* was tuned from 8 to 1. The corresponding accuracies of training and validating dataset are shown in Figure 1. To avoid overfitting, while the accuracy of validation set and training set keep climbing, a depth of 2 or 3 can be a good choice. In this case, $depth = 3$ is selected. The accuracy of testing set with the decision tree of depth 3 is 0.95, which remains the same as the above basic decision tree of depth 8.

Looking into the specific tree structures, it can be observed that the attribute "Uniformity of Cell Size" plays an important role in deciding the type of growth of breast cancer. When $depth = 2$, only this attribute was applied to determine the type of growth of breast cancer. And it reached accuracy of 0.883 and 0.80 for training and validation dataset. Besides this attribute, there are four other attributes which can be considered as key factors, including "Clump Thickness", "Uniformity of Cell Shape", "Marginal Adhesion" and "Bare Nuclei".

(3) Decision tree with post pruning
In this section, a pruned decision tree was constructed. After pruning to $depth = 2$, the accuracy of validation set decreased from 0.85 to 0.80, therefore, the pruned tree with $depth = 3$ is selected, and the accuracy of testing test is 0.95, which also remains the same as the basic decision tree of depth 8.

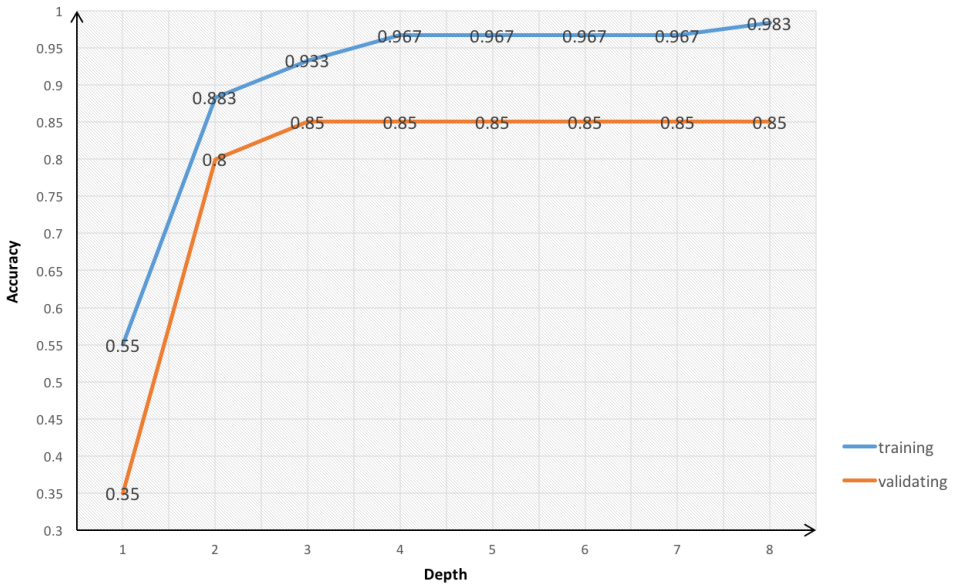

Figure 1: Accuracy of training and validation set with depth from 1 to 8

2.2.1 Comparison between strategies

Comparing these three strategies to build a decision tree, setting a fixed depth and post pruning indicate some level of effectiveness and efficiency, while keepping a relatively shorter tree. Whiling comparing the two strategies to avoid overfitting, it is hard to say which one works better. And the decision tree constructed following these two strategies were the same, in this case. To compare these two strategies, larger dataset may be needed.

2.3 Discussion

In this task, a basic decision tree based on ID_3 algorithm was implemented, and two approaches to avoid overfitting were also investigated. Besides the observations found above, it can be seen from the structures of the decision trees generated by all three methods, suffered from having too many leaves with the same label, while the predicted label can only be boolean. A future work to combine the leaves and refine these trees with less nodes may be a contribution to improve the efficiency of the algorithm.