SCHOOL OF DATA AND COMPUTER SCIENCE, SUN YAT-SEN UNIVERSITY

T04 Machine Learning

16337110 匡乾, 16337111 赖若潘

2018 年 12 月 23 日

目录

1 Q1

(a)
- 分裂前：Likes中true有5个，false有7个。
- 分裂后：Lawyers = true的6个数据中， Likes中true有4个，false有2个；
  Lawyers = false的6个数据中，Likes中true有1个，false有5个

$Remainder(Lawyers) = \frac{1}{2}B(\frac{2}{3}) + \frac{1}{2}B(\frac{1}{6})$

$Gain(Lawyers) = B(\frac{5}{5+7}) - Remainder(Lawyers)$

$= \frac{7}{12}log_2(\frac{12}{7}) + \frac{5}{12}log_2(\frac{12}{5}) - \frac{1}{2}[\frac{2}{3}log_2(\frac{3}{2}) + \frac{1}{3}log_2(3) + \frac{1}{6}log_2(6) + \frac{5}{6}log_2(\frac{6}{5})] = 0.1957$

(b) 第一次分裂同(a)所述，接下来继续对子树进行递归分裂。

(1) Laywers = true的节点中，在Comedy,Doctors, Guns选择最佳属性来分裂。

$Gain(Laywers = true, Comedy) = B(\frac{2}{3}) - \frac{1}{3}B(1) - \frac{2}{3}B(\frac{1}{2}) = 0.2516$

$Gain(Laywers = true, Doctors) = B(\frac{2}{3}) - \frac{1}{2}B(\frac{1}{3}) - \frac{1}{2}B(\frac{1}{3}) = 0$

$Gain(Laywers = true, Guns) = B(\frac{2}{3}) - \frac{1}{3}B(1) - \frac{2}{3}B(\frac{1}{2}) = 0.2516$

所以选择Gain最大的Comedy属性进行分裂。

i. Laywers = true且Comedy = true的节点中，因为所有数据都属于同一类(Likes=true)，所以停止分裂。

ii. Laywers = true且Comedy = false的节点中，在Doctors,Guns选择最佳属性来分裂。

$Gain(Laywers = true, Comedy = false, Doctors) = B(\frac{1}{2}) - \frac{1}{2}B(\frac{1}{2}) - \frac{1}{2}B(\frac{1}{2}) = 0$

$Gain(Laywers = true, Comedy = false, Guns) = B(\frac{1}{2}) - \frac{1}{2}B(1) - \frac{1}{2}B(1) = 1$

所以选择Gain最大的Guns属性进行分裂。

A. Laywers = true且Comedy = false且Guns = true的节点中，因为所有数据都属于同一类(Likes=true)，所以停止分裂。

B. Laywers = true且Comedy = false且Guns = false的节点中，因为所有数据都属于同一类(Likes=false)，所以停止分裂。

(2) Laywers = false的节点中，在Comedy,Doctors,Guns选择最佳属性来分裂。

$Gain(Laywers = false, Comedy) = B(\frac{1}{6}) - \frac{1}{2}B(1) - \frac{1}{2}B(\frac{1}{3}) = 0.1909$

$Gain(Laywers = false, Doctors) = B(\frac{1}{6}) - \frac{1}{3}B(1) - \frac{2}{3}B(\frac{1}{4}) = 0.1092$

$Gain(Laywers = false, Guns) = B(\frac{1}{6}) - \frac{1}{2}B(1) - \frac{1}{2}B(\frac{1}{3}) = 0.1909$

所以选择Gain最大的Comedy属性进行分裂。

i. Laywers = false且Comedy = true的节点中，在Doctors,Guns选择最佳属性来分裂。

$Gain(Laywers = false, Comedy = true, Doctors) = B(\frac{1}{3}) - \frac{1}{3}B(0) - \frac{2}{3}B(\frac{1}{2}) = 0.2516$

$Gain(Laywers = false, Comedy = true, Guns) = B(\frac{1}{3}) - \frac{2}{3}B(1) - \frac{1}{3}B(1) = 0.9183$

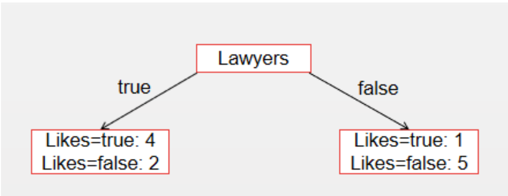
所以选择Gain最大的Guns属性进行分裂。

A. Laywers = false且Comedy = true且Guns = true的节点中，因为所有数据都属于同一类(Likes=false)，所以停止分裂。

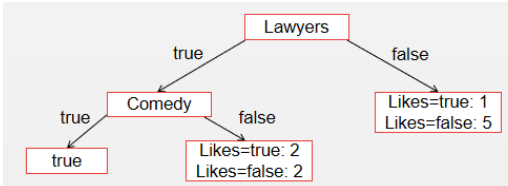
B. Laywers = false且Comedy = true且Guns = false的节点中，因为所有数据都属于同一类(Likes=true)，所以停止分裂。

ii. Laywers = false且Comedy = false的节点中，因为所有数据都属于同一类(Likes=false)，所以停止分裂。

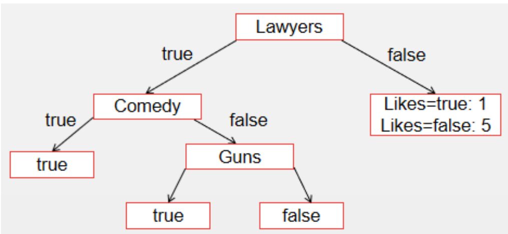
决策树建立过程如图(1)



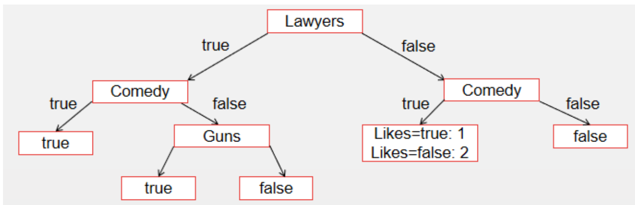
(a) 分裂Lawyers



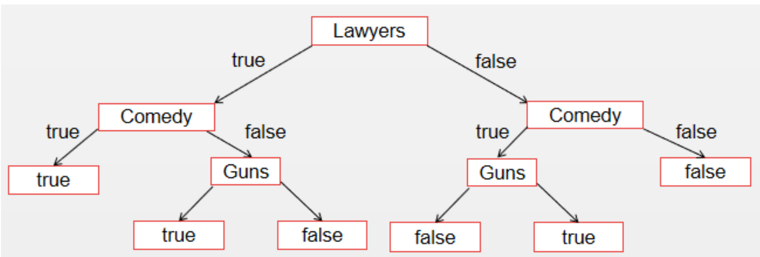
(b) Lawyers=true,分裂Comedy



(c) Lawyers=true,Comedy=false,分裂Guns



(d) Lawyers=false,分裂Comedy



(e) Lawyers=false,Comedy=true,分裂Guns

图 1: 决策树建立过程

2 Q2

h1: 100% cherry

h2: 75% cherry + 25% lime

h3: 50% cherry + 50% lime

h4: 25% cherry + 75% lime

h5: 100% lime

$d = [lime, cherry, cherry, lime, lime]$

$d' = [lime, cherry, lime, lime, lime]$

显然对于数据d和d'来说h1和h5是不可能的，因此只考虑h2,h3,h4

(a) $\because h_{MAP(d)} = argmax_h P(h)P(d|h)$

$$\begin{cases} P(h_2)P(d|h_2) = 0.2 * C_5^2(\frac{3}{4})^2(\frac{1}{4})^3 = \frac{18}{2^{10}} \\ P(h_3)P(d|h_3) = 0.4 * C_5^2(\frac{1}{2})^2(\frac{1}{2})^3 = \frac{128}{2^{10}} \\ P(h_4)P(d|h_4) = 0.2 * C_5^2(\frac{1}{4})^2(\frac{3}{4})^3 = \frac{54}{2^{10}} \end{cases}$$

$\therefore h_{MAP(d)} = h_3$

$\because h_{MAP(d')} = argmax_h P(h)P(d'|h)$

$$\begin{cases} P(h_2)P(d'|h_2) = 0.2 * C_5^1(\frac{3}{4})^1(\frac{1}{4})^4 = \frac{3}{2^{10}} \\ P(h_3)P(d'|h_3) = 0.4 * C_5^1(\frac{1}{2})^1(\frac{1}{2})^4 = \frac{64}{2^{10}} \\ P(h_4)P(d'|h_4) = 0.2 * C_5^1(\frac{1}{4})^1(\frac{3}{4})^4 = \frac{81}{2^{10}} \end{cases}$$

$\therefore h_{MAP(d')} = h_4$

由题意知，$h_{MAP} = h_3 = h_{MAP(d)}$，故遗失的数据应该为cherry。

(b) Bayesian Learning:

$\because P(X|d) = \sum_i P(X|h_i)P(h_i|d) = \sum_i P(X|h_i)\alpha P(d|h_i)P(h_i)$

$\therefore P(lime|d) - P(cherry|d)$

$= (P(lime|h_2) - P(cherry|h_2))\alpha P(d|h_2)P(h_2) + (P(lime|h_3) - P(cherry|h_3))\alpha P(d|h_3)P(h_3) +$

$(P(lime|h_4) - P(cherry|h_4))\alpha P(d|h_4)P(h_4)$

$= 0.5 * \alpha(P(d|h_4)P(h_4) - P(d|h_2)P(h_2))$

$= 0.5 * \alpha * \frac{36}{2^{10}}$

> 0

$\therefore$ 第六个candy应该是lime口味的。

ML Learning:

$h_{ML} = argmax_h P(d|h)$

显然h1和h5依然是不可能的

$$\begin{cases} P(d|h_2) = C_5^2(\frac{3}{4})^2(\frac{1}{4})^3 = \frac{90}{2^{10}} \\ P(d|h_3) = C_5^2(\frac{1}{2})^2(\frac{1}{2})^3 = \frac{320}{2^{10}} \\ P(d|h_4) = C_5^2(\frac{1}{4})^2(\frac{3}{4})^3 = \frac{270}{2^{10}} \end{cases}$$

$\therefore h_{ML} = h_3$

$\because P(cherry|h_3) = 0.5 = P(lime|h_3)$

$\therefore$故无法确定第六个糖果的口味（lime和candy的可能性相同）

3 Q3

Algorithm 1: EM(X,D,k)

Input : X set of features X = $\{X_1, ..., X_n\}$

D data set on features $\{X_1, ..., X_n\}$

k number of classes

Output: $P(C), P(X_i|C)$ for each i $\in \{1 : n\}$, where C = $\{1, ..., k\}$.

Local : real array $P[C]$

real arrays $M_i[X_i, C]$ for each i $\in \{1 : n\}$

real arrays $P_i[X_i, C]$ for each i $\in \{1 : n\}$

s := number of tuples in D

Assign $P[C], P_i[X_i, C]$ arbitrarily

repeat

  Assign $M_i[X_i, C]$ *to all* $0, i \in \{1 : n\}$

  for *each assignment* $\langle X_1 = v_1, ..., X_n = v_n\rangle \in D$ **do**

    let $m \leftarrow |\langle X_1 = v_1, ..., X_n = v_n\rangle \in D|$

    for *each* $c \in \{1 : k\}$ **do**

      for *each* $i \in \{1 : k\}$ **do**

        $M_i[X_i = v_i, C = c] \mathrel{+}= m \times P(C = c|X_1 = v_1, ..., X_i = v_i, ..., X_n = v_n)$

  for *each* $i \in \{1 : k\}$ **do**

    $P_i[X_i, C] = \frac{M_i[X_i, C]}{\sum_C M_i[X_i, C]}$

  $P[C] = \sum_{X_1} M_1[X_1, C]/s$

until *termination*;

4 Q4

(a)

首先，由正项级数中“p级数”可知：

对于$\sum\limits_{n=1}^{\infty}\frac{1}{n^p}$，当$p>1$时收敛，当$p\le 1$时发散，证明如下：

- 当$p=1$时：设$2^k\le n<2^{k+1}$

$$
\begin{aligned}
S_n &= 1+\frac{1}{2}+\frac{1}{3}+\cdots+\frac{1}{n}\\
&= 1+\frac{1}{2}+(\frac{1}{3}+\frac{1}{4})+\cdots+(\frac{1}{2^{k-1}+1}+\frac{1}{2^{k-1}+2}+\cdots+\frac{1}{2^{k-1}+2^{k-1}})+\frac{1}{2^k+1}+\cdots+\frac{1}{n}\\
&\ge \frac{1}{2}+\frac{1}{2}+(\frac{1}{4}+\frac{1}{4})+\cdots+(\frac{1}{2^k}*2^{k-1})\\
&= \frac{k+1}{2}
\end{aligned}
$$

  此时k可以取任意大，因而S_n无上界。故$p=1$时，级数$\sum\limits_{n=1}^{\infty}\frac{1}{n}$发散。

- 当$p<1$时：对任意正整数k，有

$$\frac{1}{k^p}\ge\frac{1}{k}$$

  $\therefore$

$$\sum_{k=1}^{\infty}\frac{1}{k^p}\ge\sum_{k=1}^{\infty}\frac{1}{k}$$

  右边部分数列无上界故左边也无上界，故$\sum\limits_{n=1}^{\infty}\frac{1}{n^p}$在$p<1$也发散

- 当$p>1$时：设$2^k\le n<2^{k+1}$

$$
\begin{aligned}
S_n &= 1+\frac{1}{2^p}+\frac{1}{3^p}+\cdots+\frac{1}{n^p}\\
&= 1+(\frac{1}{2^p}+\frac{1}{3^p})+\cdots+[\frac{1}{(2^{k-1})^p}+\frac{1}{(2^{k-1}+1)^p}+\cdots+\frac{1}{(2^k-1)^p}]+\frac{1}{(2^k)^p}+\cdots+\frac{1}{n^p}\\
&\le 1+\frac{2}{2^p}+\frac{4}{4^p}+\cdots+\frac{2^{k-1}}{(2^{k-1})^p}+\frac{2^k}{(2^k)^p}\\
&= 1+\frac{1}{2^{p-1}}+(\frac{1}{2^{p-1}})^2+\cdots+(\frac{1}{2^{p-1}})^{k-1}++(\frac{1}{2^{p-1}})^k\\
&= \frac{1-(\frac{1}{2^{p-1}})^{k+1}}{1-\frac{1}{2^{p-1}}}\\
&\le \frac{1}{1-\frac{1}{2^{p-1}}}\\
&= \frac{2^{p-1}}{2^{p-1}+1}
\end{aligned}
$$

  因此部分和数列有上界，故$\sum\limits_{n=1}^{\infty}\frac{1}{n^p}$当$p>1$时收敛

其次，Convergence can be guaranteed if $\sum_{k=1}^{\infty} \alpha_k = \infty$ and $\sum_{k=1}^{\infty} \alpha_k^2 < \infty$

$\therefore$

- 当$\alpha_k = 1/k$时：
  由正项级数中“p级数”可知，$\sum_{k=1}^{\infty} \alpha_k = \infty \ (p = 1)$ 而且 $\sum_{k=1}^{\infty} \alpha_k^2 < \infty \ (p = 2)$，满足收敛条件
- 当$\alpha_k = 10/(9 + k)$时：
  由正项级数中“p级数”以及比较判别法的极限形式可知，$\sum_{k=1}^{\infty} \alpha_k = \infty \ (p = 1)$ 而且 $\sum_{k=1}^{\infty} \alpha_k^2 < \infty \ (p = 2)$，满足收敛条件
- 当$\alpha_k = 0.1$时：
  因为$\sum_{k=1}^{\infty} \alpha_k^2 = \lim_{x \to \infty} 0.01 * x = \infty$，故不满足收敛条件
- 当$\alpha_k = 0.1, 0.01, 0.001...$时：

$$
\begin{aligned}
\sum_{k=1}^{\infty} \alpha_k &= 1000 + 100 + 10 + \cdots \\
&= \sum_{n=1}^{\infty} 10000 * (\frac{1}{10})^n \\
&\leq \frac{1000}{1 - \frac{1}{10}} \\
&< \infty
\end{aligned}
$$

  故$\sum_{k=1}^{\infty} \alpha_k < \infty$，不满足收敛条件

综上$\alpha_k = 1/k$以及$\alpha_k = 10/(9 + k)$时有$\sum_{k=1}^{\infty} \alpha_k = \infty$ 和 $\sum_{k=1}^{\infty} \alpha_k^2 < \infty$，满足收敛条件。

(b) 使用了帅哥TA提供的E14中的Q-learning样例(见图Q4-b)

$$
Q= \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array}
\begin{array}{c} \begin{array}{cccccc} 0 & 1 & 2 & 3 & 4 & 5 \end{array} \\
\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \end{array}
\quad
R= \begin{array}{c} \text{State} \\ 0 \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array}
\begin{array}{c} \text{Action} \\ \begin{array}{cccccc} 0 & 1 & 2 & 3 & 4 & 5 \end{array} \\
\begin{bmatrix} -1 & -1 & -1 & -1 & 0 & -1 \\ -1 & -1 & -1 & 0 & -1 & 100 \\ -1 & -1 & -1 & 0 & -1 & -1 \\ -1 & 0 & 0 & -1 & 0 & -1 \\ 0 & -1 & -1 & 0 & -1 & 100 \\ -1 & 0 & -1 & -1 & 0 & 100 \end{bmatrix} \end{array}
$$

图 2: Q4-b

保持learning rate = 0.8 不变，episode分别取10000和100000，显然第一个和第二个α_k此时都还没收敛，而第三个和第四个α_k已经收敛了。经检验，在9000左右时，第三个和第四个α_k完成收敛到真实Q-Value（因为在$k < 10000$时，第三个和第四个α_k有相同的取值，故为同时收敛的）

当然对于不同的域来说，其结果是不同的。首先第一个条件$\sum\limits_{k=1}^{\infty} \alpha_k = \infty$确保了随机函数和初始条件排除了平均值的情况，第二个条件$\sum\limits_{k=1}^{\infty} \alpha_k^2 < \infty$保证收敛性，因此在不同域的情况下，第三个和第四个$\alpha_k$不一定能确保收敛，但在k不够大的情况下，第一个和第二个α_k同样也不能确保收敛。

(c) 我认为environment adapts slowly意思是learning rate较小，因此learning rate分别取0.1和0.01的情况下，$\alpha_k = 10/(9+k)$的表现有很大的改进，它可以更快的收敛，大概在100000次时已经收敛到真实Q-Value。