HW5

Liyuan Tang

5/31/2020

Problem 1

(a). The marginal distribution of X is a uniform distribution in $(-4,-2) \cup (2,4)$.

$$\begin{aligned} p(x) &= p(x|Y=1) \cdot p(Y=1) + p(x|Y=2) \cdot p(Y=2) \\ &= \frac{1}{2} \cdot \frac{1}{2} \cdot \mathbf{1}_{(-4,-2)}(x) + \frac{1}{2} \cdot \frac{1}{2} \cdot \mathbf{1}_{(2,4)}(x) \\ &= \frac{1}{4} \cdot (\mathbf{1}_{(-4,-2)}(x) + \mathbf{1}_{(2,4)}(x)) \end{aligned}$$

The condtional distribution of Y given X is the following:

$$\begin{aligned} p(Y=1|X \in (-4,-2)) &= 1 \\ p(Y=2|X \in (-4,-2)) &= 0 \\ p(Y=1|X \in (2,4)) &= 0 \\ p(Y=2|X \in (2,4)) &= 1 \end{aligned}$$

(b). Based on the conditional distribution of Y given X, we want $\underset{i}{\text{argmax}} P(y=i|X)$. So we can first get $f_B(x \in [-4,-2]) = 1$, since $p(Y=1|X \in (-4,-2)) > p(Y=2|X \in (-4,-2))$. Similarly, $f_B(x \in [2,4]) = 2$. For the risk, we know that $p(Y=2|X \in (-4,-2)) = p(Y=1|X \in (2,4)) = 0$, so the risk is 0.

(c). The only situation for $y \neq \hat{f}_1(X;S)$ is that when all x_i in the training sample are in one of the interval and the X for the new data is in another interval. For example, if the training sample $x_i \in [-4,-2]$ for all i, then given a new independent pair (X,Y), the $\hat{f}_1(X;S)$ will always be 1 even if $X \in [2,4]$.

Thus, the risk is

$$\begin{aligned} Pr(Y \neq \hat{f}_1(X;S)) &= P(\text{all } x_i \in [-4,-2] \text{ and } X \in [2,4]) + P(\text{all } x_i \in [2,4] \text{ and } X \in [-4,-2]) \\ &= 2 \cdot (\frac{1}{2})^n \cdot \frac{1}{2} \\ &= (\frac{1}{2})^n \end{aligned}$$

(d). For three-nearest neighbor, the situation for misclassification is that there is at most 1 training data with x_i in the same interval as the new data point X. One possible situtaion could be: only one data point $x_d \in [-4,-2]$ and the rest are all in [2, 4] while the new data point $X \in [-4,-2]$. In this case, we will get the predicted value as 2 instead of 1.

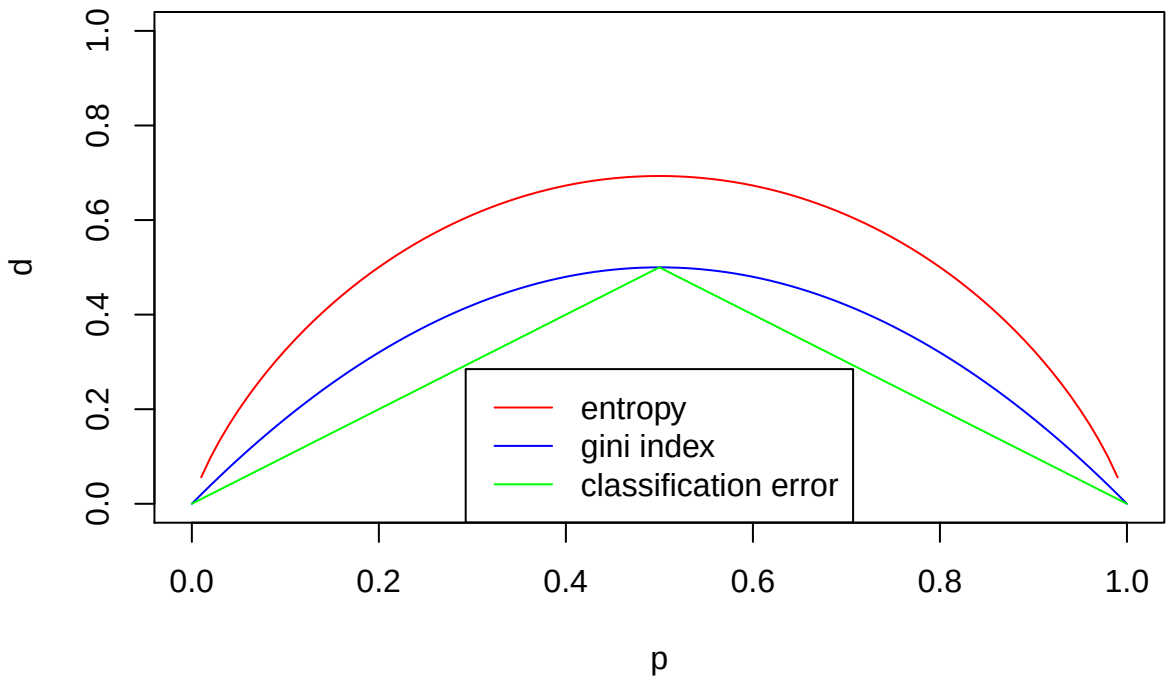
Thus, the risk is

$$\begin{aligned} Pr(Y \neq \hat{f}_1(X;S)) &= 2 \cdot P(\text{at most 1 sample data } x_i \in [-4,-2] \text{ and } X \in [-4,-2]) \\ &= 2 \cdot (P(\text{only 1 sample data } x_i \in [-4,-2] \text{ and } X \in [-4,-2]) + P(\text{all } x_i \in [2,4] \text{ and } X \in [-4,-2])) \\ &= 2 \cdot (n \cdot (\frac{1}{2})^n \cdot \frac{1}{2} + (\frac{1}{2})^n \cdot \frac{1}{2}) \\ &= (n+1)(\frac{1}{2})^n \end{aligned}$$

(e). 1-nearest neighbor has smaller risk in this problem.

Section 8.4 Problem 3

```
p = seq(0, 1, 0.01)
k = 2
e = 1 - pmax(p, 1-p)
g = k * p * (1-p)
d = -(p * log(p)) - (1-p)*log(1-p)
plot(p, d, type='l', col = 'red', ylim = c(0, 1))
lines(p, g, col = 'blue')
lines(p, e, col = 'green')
legend(x='bottom',legend=c('entropy','gini index','classification error'),
       col=c('red','blue','green'),lty=1)
```



Section 8.4 Problem 9

(a)

```
library(ISLR)
#View(OJ)
set.seed(1)
v = sample(dim(OJ)[1], 800)
train_set = OJ[v,]
test_set = OJ[-v,]
```

(b)

```
library(tree)
tree.oj=tree(Purchase~.,train_set)
summary(tree.oj)
```

```
##
## Classification tree:
## tree(formula = Purchase ~ ., data = train_set)
## Variables actually used in tree construction:
## [1] "LoyalCH"       "PriceDiff"     "SpecialCH"     "ListPriceDiff"
## [5] "PctDiscMM"
## Number of terminal nodes:  9
## Residual mean deviance:  0.7432 = 587.8 / 791
## Misclassification error rate: 0.1588 = 127 / 800
```

The training error rate is 0.1588. The tree has 9 terminal nodes.

(c)

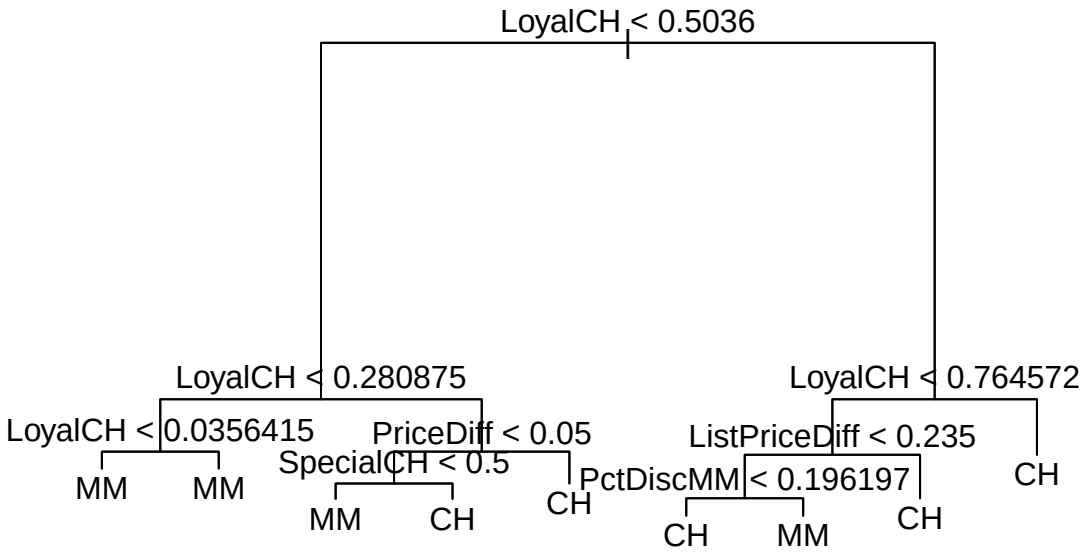
```
tree.oj
```

```
## node), split, n, deviance, yval, (yprob)
##       * denotes terminal node
##
##  1) root 800 1073.00 CH ( 0.60625 0.39375 )
##    2) LoyalCH < 0.5036 365  441.60 MM ( 0.29315 0.70685 )
##      4) LoyalCH < 0.280875 177  140.50 MM ( 0.13559 0.86441 )
##        8) LoyalCH < 0.0356415 59   10.14 MM ( 0.01695 0.98305 ) *
##        9) LoyalCH > 0.0356415 118  116.40 MM ( 0.19492 0.80508 ) *
##      5) LoyalCH > 0.280875 188  258.00 MM ( 0.44149 0.55851 )
##       10) PriceDiff < 0.05 79   84.79 MM ( 0.22785 0.77215 )
##         20) SpecialCH < 0.5 64   51.98 MM ( 0.14062 0.85938 ) *
##         21) SpecialCH > 0.5 15   20.19 CH ( 0.60000 0.40000 ) *
##       11) PriceDiff > 0.05 109  147.00 CH ( 0.59633 0.40367 ) *
##    3) LoyalCH > 0.5036 435  337.90 CH ( 0.86897 0.13103 )
##      6) LoyalCH < 0.764572 174  201.00 CH ( 0.73563 0.26437 )
##       12) ListPriceDiff < 0.235 72   99.81 MM ( 0.50000 0.50000 )
##         24) PctDiscMM < 0.196197 55   73.14 CH ( 0.61818 0.38182 ) *
##         25) PctDiscMM > 0.196197 17   12.32 MM ( 0.11765 0.88235 ) *
##       13) ListPriceDiff > 0.235 102   65.43 CH ( 0.90196 0.09804 ) *
##      7) LoyalCH > 0.764572 261   91.20 CH ( 0.95785 0.04215 ) *
```

Pick the node 8). It is the terminal node and its root is 4). It splits the tree by LoyalCH < 0.0356415. The total number of observations is 59 and the deviance is 10.14. The prediction for this branch of MM. Only 1.695% of the obeservation in this branch have the value of CH, the rest are all MM.

(d)

```
plot(tree.oj)
text(tree.oj,pretty=0)
```



The most imporant predictor variable that influences 'Purchase' is 'LoyalCH' which is the customer brand loyalty for CH.

(e)

```
pred.oj = predict(tree.oj,test_set,type="class")
table(pred.oj, test_set$Purchase)
```

```
##
## pred.oj  CH  MM
##      CH 160  38
##      MM   8  64
```

```
t.error = (8+38) / 270
```

The test error rate is 0.1703704.

(f)

```
set.seed(1)
cv.oj = cv.tree(tree.oj,FUN=prune.misclass, K = 10)
num.oj = cv.oj$size[which.min(cv.oj$dev)]
```

The optimal tree size is 9.

(g)

```
plot(cv.oj$size, cv.oj$dev / nrow(train_set), type = 'b')
```