304990072

HW1

Zixuan Lu

1. 1(a) ~~The~~ Since $X_i\ (i \geq 4)$ does not affect the result of Y, The model makes mistakes when $X_1 = X_2 = X_3 = 0$. Thus, when $n \geq 4$, the model makes mistakes in $\frac{1}{8}$ of all the examples, which is $\frac{2^n}{8} = 2^{n-3}$ mistakes

(b) There does not exist such a split. Splitting on $X_i\ (i \geq 4)$ does not make any difference. Splitting of $X_i\ (1 \leq i \leq 3)$ would create a tree with a leaf of all 1's and the other leaf $\frac{3}{4}$ 1's. Thus, the tree will still predict all 1's, making no difference.

1c $H[Y] = -\frac{1}{8}\log\left(\frac{1}{8}\right) - \frac{7}{8}\log\frac{7}{8} = 0.5436$

1d $H = -\left(\frac{1}{8}\log\frac{1}{8} + \frac{7}{8}\log\frac{7}{8}\right) \times \frac{1}{2} \times 2 = 0.5436$

$\Rightarrow$ The gain in entropy is 0

1e New entropy is $-\frac{1}{2}\left(\frac{0}{4}\log\frac{0}{4} + \frac{4}{4}\log\frac{4}{4}\right) - \frac{1}{2}\left(\frac{1}{4}\log\frac{1}{4} + \frac{3}{4}\log\frac{3}{4}\right)$

$= 0 + 0.4056$

$= 0.4056$

Gain $= 0.5436 - 0.4056 = 0.138$

2. (a) $-q\log q - (1-q)\log(1-q)$

$= \log q^{-q} - \log(1-q)^{1-q}$

$= \log \frac{q^{-q}}{(1-q)^{1-q}}$

$= \log \frac{1}{q^q(1-q)^{1-q}}$

To maximize $B(q)$, we need to minimize $q^q(1-q)^{1-q}$, differentiate it, we get $-\frac{(x-1)x^x[\ln(x)-\ln(1-x)]}{(1-x)^x}$

we get the first derivative is 0 when $q=\frac{1}{2}$. Since its second derivative is always negative, we have $B(q)$ is maximized when $q=\frac{1}{2}$

(b) $H(x_j) = H(S_1)\frac{p_1+n_1}{p+n} + \dots + H(S_k)\frac{p_k+n_k}{p+n}$

Since each $H(S_i)$ is equal to $H(S)$, we have

$= H(S)\frac{p_1+p_2+\dots+p_k+n_1+n_2+\dots+n_k}{p+n}$

$= H(S)\frac{p+n}{p+n}$

$= H(S)$

Thus, the information gain is 0.

3(a) k=1 minimizes the training set error. Since a point can be its own neighbour, the error is 0.

(b) A too big k might lead to misclassification. For example, the two "+" are close to the 5 "−" in the upper part. A too small k will lead to overfitting

(c) The optimal k value is k=5.
The error is $\frac{2}{7}$.

Scanned with CamScanner

4 'a' Class: From Figure 1, we can see the number of passengers who survived is almost double of that who did not survived in class 1. The trend reverses in class 3. Thus, the higher the class, the more likely the passenger would survive. (the number survived is close to that who did not survive in class 2)

Sex: We can see from Figure 2 that the survival rate of a man is significantly lower than that of a woman. Thus, female passengers were more likely to survive.

Age: We can see that only passengers with age 10 or below has a higher survival rate than death rate from Figure 3. Most passengers are between 20-40 years old, and they have the highest death rates.

SibSp: From Figure 4, we can see that passengers with 1 or 2 siblings have the highest survival rates. Passengers with no or more than 2 siblings have comparable low survival rates.

Parch: We can see from Figure 5 that most people have 0, 1 or 2 parent/children on board. Those with 1 or 2 parents/children have higher survival rate than those have none.

Fare: We can see from Figure 6 that people pay less than \$50 fare have a much higher death rate than the rest groups. In fact, they are the only group with a higher death rate than survival rate.

Embarked: From figure 7, people embarked at 0.0 has higher survival rate. They are the only group with a survival rate higher than death rate.

Scanned with CamScanner

(b) The training error is 0.485 as seen in the code results.

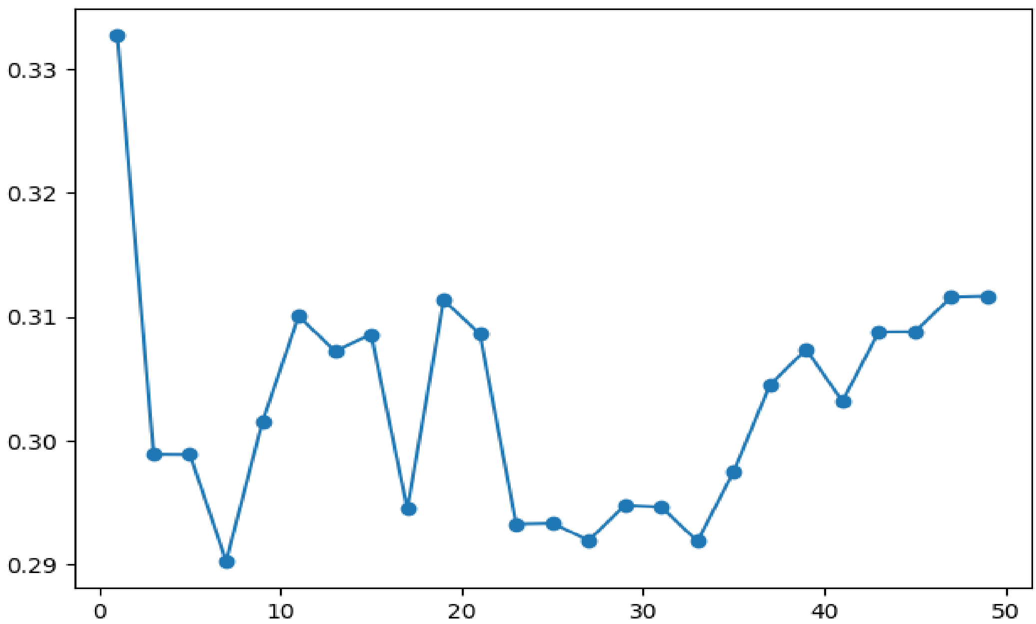
(c) The training error is 0.014.

(d) The training error is 0.167, 0.201, 0.240 for k=3,5,7 respectively

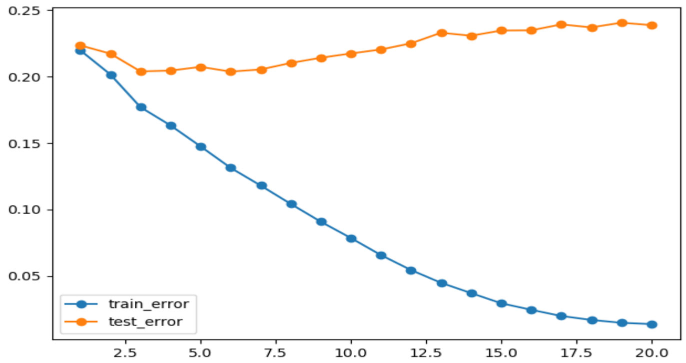
(e)

	Train error	Test error
MajorityVote	0.404	0.407
Random	0.489	0.487
DecisionTree	0.012	0.246
kNeighbors	0.212	0.315

(f) The best k value is 7. The cross validation error is 0.290
~~The~~ When the number of neighbors increases to the range of 23-33, the error is also decently low. Then the error starts to increase again.



(g) The best depth limit is 6, with a lowest testing error of 0.204 and a relatively low training error (as compared to depth=3, which has the same testing error but a higher training error). As the number of depth limit increases, the training error decreases, but the testing error increases. This is caused by overfitting. The model becomes over-complex and cannot classify data outside the training set accurately.



(h) From the graph we can see that the testing and training error decreases as the training sample grows. However, there is an exception: the decision tree's training error increases. This is odd because with optimal depth (6), we expect the model to be more accurate with more training data.