CS573 DM HW2

Question 1 - Ensembles

Part 1

Model M_2 is expected to perform better than M_1. Bagging tends to reduce the variance and leave the bias untouched. So models with high variance and low bias are expected to perform better with bagging. Decision stumps (height one decision trees) have lower variance compared to full-depth decision trees. Hence bagging with decision trees will have a greater effect.

Part 2

Bagging works better with estimators that have small biases. In practice, it is unwise to used for models that can have high biases. This exercise is designed to show exactly that. In the question we assume $\epsilon_{k,i}$ are iid random variables. Without loss of generality, we assume $E[\epsilon_{k,i}] = C$, which is the mean squared error (MSE).

$$
\begin{aligned}
e_{\text{single},k} &= E[\sum_i \epsilon_{k,i}] = E[\epsilon_{k,i}] + E[\epsilon_{k,2}] + ... + E[\epsilon_{k,n}] = nC \\
e_{\text{bagging}} &= E[\sum_i \epsilon^{'}_{k,i}] \\
&= E[\sum_i (y_i - f^*(x_i))^2] \\
&= E[\sum_i (y_i - \frac{1}{K}\sum_{k=1}^{K} f_k(x_i))^2] \\
&= E[\sum_i (\frac{1}{K}y_i + \frac{1}{K}y_i + \frac{1}{K}y_i + ... + \frac{1}{K}y_i - \frac{1}{K}\sum_{k=1}^{K} f_k(x_i))^2] \\
&= E[\sum_i (\frac{1}{K}(y_i - f_1(x_i)) + \frac{1}{K}(y_i - f_2(x_i)) + ... + \frac{1}{K}(y_i - f_k(x_i)))^2]
\end{aligned}
$$

If $\epsilon_{k,i} = (y_i - f_k(x_i))^2$, then $y_i - f_k(x_i) \leq |y_i - f_k(x_i)| = \theta_{i,k}$, where $\theta_{i,k} = \sqrt{\epsilon_{k,i}}$ is a non-negative number. Let $H = E[\theta_{i,k}] = E[|y_i - f_k(x_i)|]$. Using this information in the above equation yields the bound

$$
\begin{aligned}
e_{\text{bagging}} &\leq \frac{1}{K^2} E\left[\sum_i (\theta_{1,i} + \cdots + \theta_{K,i})^2\right] \\
&= \frac{1}{K^2} \sum_i E\left[(\theta_{1,i} + \cdots + \theta_{K,i})^2\right]
\end{aligned}
$$

$$
\begin{aligned}
&= \frac{1}{K^2} \sum_i \sum_k E\left[\theta_{k,i}^2\right] + \sum_{a,b,a\neq b} E\left[\theta_{a,i}\theta_{b,i}\right] \\
&= \frac{1}{K^2} \sum_i KC + \binom{K}{2} H^2 \\
&= \frac{nC}{K} + n(1-1/K)H^2.
\end{aligned}
$$

Note that the variance decrease is inversely proportional to the number of models (K), but the error still has the term H^2. To understand what H^2 is, assume now that $f_k(x_i) < y_i$, $\forall k$, i.e., the regression always underestimates the true value. Note that under this assumption the above upper bound is tight (an equality, actually) and H^2 is the square of the bias. Thus, bagging will reduce some of the variance but, for estimators with high bias, the squared error will converge to the square of the bias. Also, because $E[(\theta_{k,i} - E[\theta_{k,i}])^2] \geq 0$, then $H^2 \leq C$. That is, bagging will reduce the variance as $K \to \infty$, but not by much if the estimator is very biased and the estimator error has little variance.

Question 2 (2.5 pts): Classification Tasks

(a) Yes, a decision tree can classify all the points correctly. Split on the attribute X_1 at first level and then on the attribute X_2.

(b) No, since a logistic regression classifier forms a linear decision boundary and clearly the points are not linearly separable(over the original feature space).

By transforming the features into a different feature space, the LR classifier can classify all the points correctly. One possible method is if we consider another feature X_1X_2 in addition to X_1 and X_2 then we can achieve a linear decision surface in the transformed space.

(c) No.
Prior probabilities.
P(Y = 1) = 0.5 P(Y = 0)
The conditional probabilities:
P($X_1 = 1|Y = 0) = P(X_1$ = 1|Y = 1) = 0.5
P($X_1 = 0|Y = 0) = P(X_1$ = 0|Y = 1) = 0.5
P($X_2 = 1|Y = 0) = P(X_2$ = 1|Y = 1) = 0.5
P($X_2 = 0|Y = 0) = P(X_2$ = 0|Y = 1) = 0.5
For prediction this NBC will give posterior probabilities P(Y = 1|X_1, X_2) = P(Y = 0|X_1, X_2) = 0.5.

Hence, it cannot perfectly classify the points given in the problem.

Question 3 : (2.0 pts): Decision Tree

(a) Yes, it is possible.
Whenever we split on a continuous attribute, we can use the same attribute for splitting with different threshold values.
For example consider a task with 2 labels {+,*}, all the points are in 1 dimensional space.
++++*******++++

If we use decision tree we would have to split on the same attribute twice to classify all points correctly.

(b) No, the statement is not always correct. Consider the example given in Problem 2. The information gain at first splitting is zero, while information gain at second splitting is >0.

(c) The probability that a particular feature is not considered when we select one feature out of m features = $\frac{m-1}{m}$
The sampling at each node is independent of other nodes. There are a total of PQ such samplings. Hence the final probability is =$(\frac{m-1}{m})^{PQ}$

(d) Both the labels should be 1 for the given values of information gain.

Q4: Naive Bayesian Classifier

(a)

$$\begin{aligned} P(y=1|x1=1,x2=0,x3=1) &\propto P(x1=1,x2=0,x3=1|y=1)P(y=1) \\ &= 1/2 \cdot 7/10 \cdot 1/5 \cdot 4/10 = 7/250 \\ P(y=0|x1=1,x2=0,x3=1) &\propto P(x1=1,x2=0,x3=1|y=0)P(y=0) \\ &= 1/3 \cdot 3/4 \cdot 2/3 \cdot 6/10 = 1/10 \end{aligned}$$

$P(y=0|x1=1,x2=0,x3=1) > P(y=1|x1=1,x2=0,x3=1) \Rightarrow y=0$ for $x=(1,0,1)$.

(b) The decision boundary of the Naive Bayes classifier is:

$$\frac{P(y=1|x)}{P(y=0|x)} = 1$$

.

$$\frac{P(y=1) \cdot P(x_1|y=1) \cdot P(x_2|y=1) \cdot P(x_3|y=1)}{P(y=0) \cdot P(x_1|y=0) \cdot P(x_2|y=0) \cdot P(x_3|y=0)} = 1$$

(c) No. If the data is linearly inseparable, NBC won't be able to achieve zero training error.

Q5: Boosting

(a) The negative examples (zeros) since the decision stump only makes errors on the negative examples (actually it always predicts incorrectly two training examples).

(b) (i) At least 5 iterations. See Figures below.

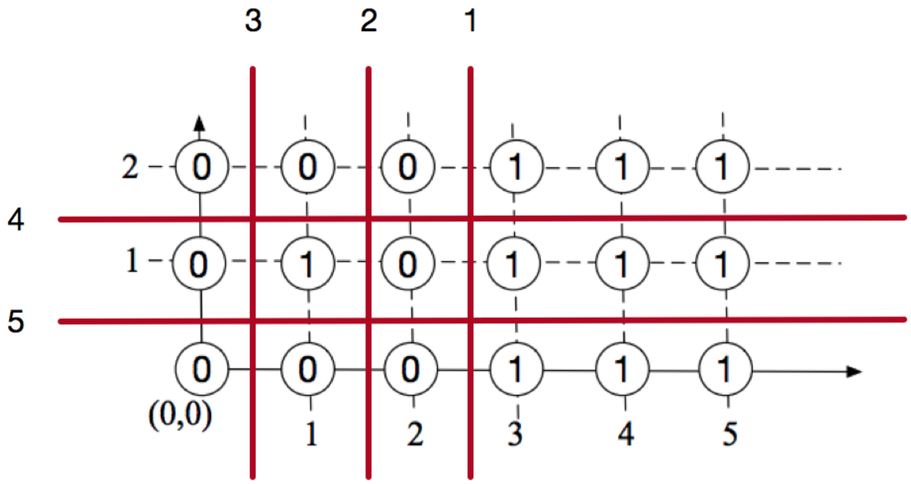

Figure 1

(ii) At least 3 iterations. See Figures below. Note that a two level decision tree will at most have 3 internal nodes, therefore the decision boundary would be combination of 3 decision stumps.

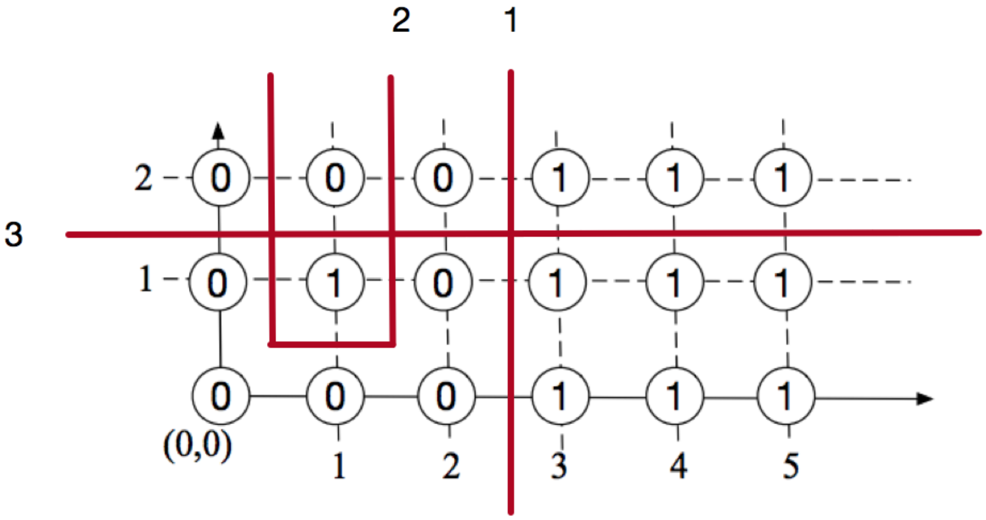

Figure 2

(iii) The advantage of using decision stumps is that it prevents overfitting. It is likely that the example with label 1 surrounded by the examples with label 0 is mislabeled. From (i) and (ii), we can conclude that it is easier for 2-level decision tree to fit the mislabeled data.