Kyle Benson

CS 273A - Machine Learning: Fall 2013

Homework 2

## Problem 1: Bayes Classifiers

(a) Probabilities for Naive Bayes Classifier:
$p(y = 1) = 0.4$

| $p(Variable = 1 \vert y = 1)$ | $x_1$ | $x_2$ | $x_3$ | $x_4$ | $x_5$ | $\prod$ |
|---|---|---|---|---|---|---|
| Probability | 0.75 | 0.0 | 0.75 | 0.5 | 0.25 | 0 |
| $p(Variable = 1 \vert y = -1)$ | $x_1$ | $x_2$ | $x_3$ | $x_4$ | $x_5$ | $\prod$ |
| Probability | 0.5 | 0.833 | 0.667 | 0.833 | 0.33 | .0756 |

(b) For $x = (00000), p(y = 1) \prod_i p(x_i = 0) = 0.4 * (0.25 * 1 * 0.25 * 0.5 * 0.75) = 0.009375$ and $p(y = -1) \prod_i p(x_i = 0) = 0.6 * (0.5 * 0.167 * 0.33 * 0.167 * 0.67) = 0.00185$. So the predicted class would be $y = 1$

For $x = (11010), p(y = 1) \prod_i p(X_i = x_i) = 0.4 * (0.75 * 1 * 0.25 * 0.5 * 0.75) = 0.028$ and $p(y = -1) \prod_i p(x_i = 0) = 0.6 * (0.5 * 0.833 * 0.333 * 0.833 * 0.667) = 0.046236114$. So the predicted class would be $y = -1$

(c) $p(y = 1|x = (11010)) = \frac{p(y=1)p(x=(11010)|y=1)}{p(x=(11010))} = \frac{0.4*(0.75*1*0.25*0.5*0.75)}{(0.75*1*0.25*0.5*0.75)+(0.5*0.833*0.333*0.833*0.667)} = 0.0087$

(d) Because then our probability table will have $O(F^2)$, rather than $O(F)$, entries, where $F$ is the number of features we are training on, in order to account for dependence among the feature variables.

(e) We do not need to re-train the model. Because $x_1$ is independent of all other $x_i$, we can safely ignore $x_1$ entirely and make our predictions based on the classifier: $p(y) \prod_{i \neq 1} x_i$

## Problem 2: Decision Trees

(a) $H(y) = -0.4 * log(0.4) - 0.6 * log(0.6) = 0.971$

(b) Using the formula $H(y|x_i = 0) = p(y = 1|x_i = 0) \log p(y = 1|x_i = 0) + p(y = 0|x_i = 0) \log p(y = 0|x_i = 0)$

| $Variable$ | $x_1$ | $x_2$ | $x_3$ | $x_4$ | $x_5$ |
|---|---|---|---|---|---|
| Entropy $H(y \vert x_i = 0)$ | 1 | 0.4312 | 1.03 | 0.931 | 0.887 |
| Entropy $H(y \vert x_i = 1)$ | 0.811 | 0.22 | 0.701 | 0.72 | 1.028 |
| Info Gain | 0.0465 | 0.61 | 0.006 | 0.0914 | 0.006 |

Therefore, we should split on feature $x_2$ first.

(c) After splitting on $x_2$, we recompute the following info gains for each of the remaining features:
$H(y|x_2 = 0) = 0.722$

| $Variable$ | $x_1$ | $x_2$ | $x_3$ | $x_4$ | $x_5$ |
|---|---|---|---|---|---|
| Info Gain | 0.322 | - | 0.073 | 0.171 | 0.073 |

$H(y|x_2 = 1) = 0$, so no need to explore this branch

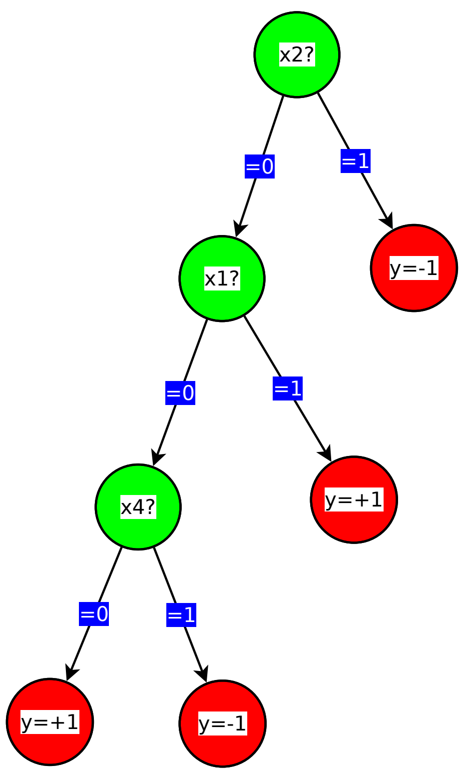
We therefore pick $x_1$ as the next feature to split. Again, we recompute the following info gains for each of the remaining features:

$H(y|x_2 = 0, x_1 = 0) = 1$

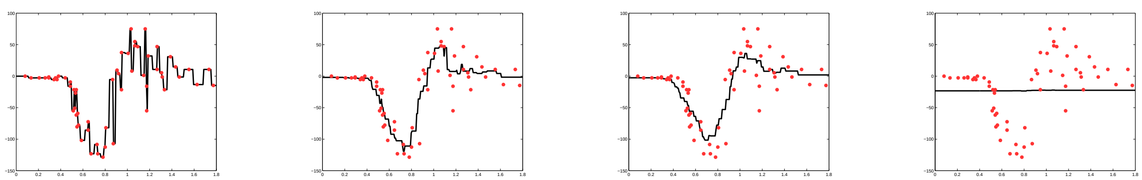
| $Variable$ | $x_1$ | $x_2$ | $x_3$ | $x_4$ | $x_5$ |
|---|---|---|---|---|---|
| Info Gain | - | - | 0 | 1 | 0 |

$H(y|x_2 = 0, x_1 = 1) = 0$, so no need to explore this branch

We therefore pick $x_4$ as the next feature to split and are done because the information gain is 1, meaning that feature $x_4$ perfectly explains this branch of the data $(x_1 = 0, x_2 = 0)$. This gives us the following decision tree:



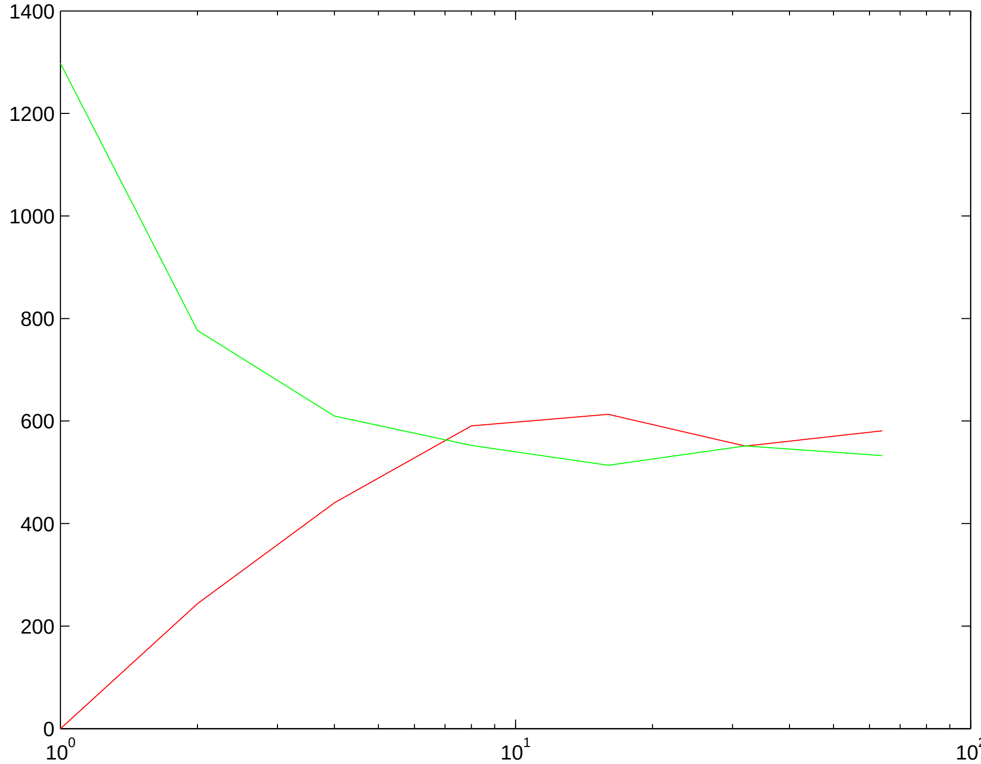
**Problem 2: K-Nearest Neighbors and Validation**



$K = 1$ $K = 5$ $K = 10$ $K = 50$

(a)

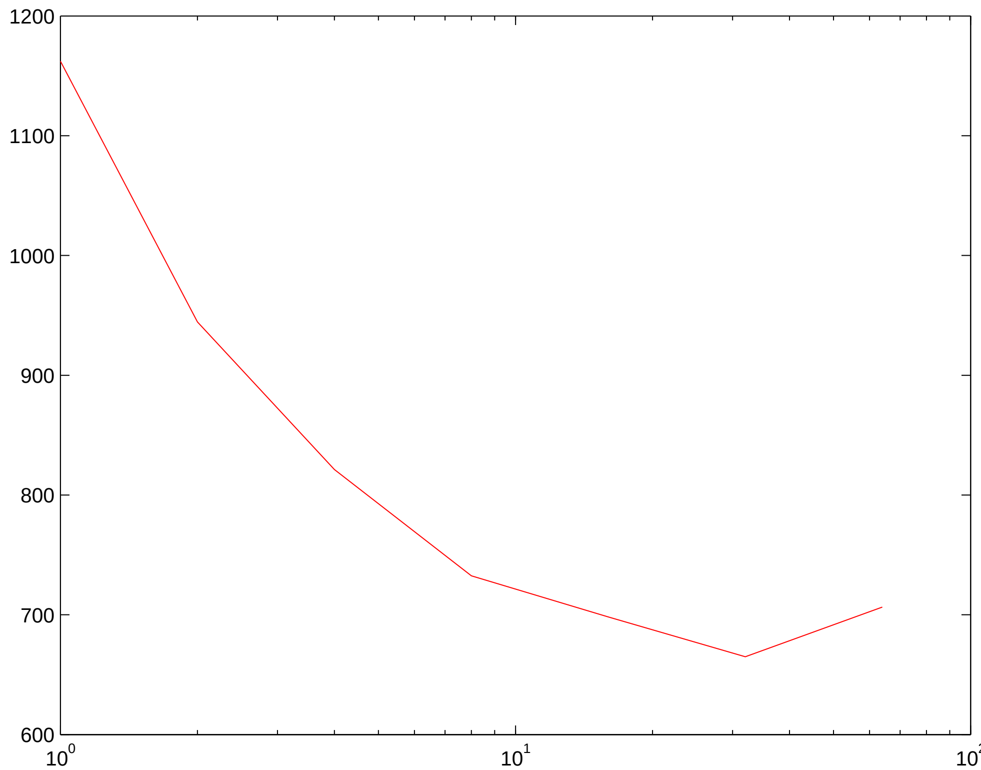
Given the following MSEs:




I would recommend choosing $K = 16$ since the MSE appears to increase for the test set after this value.

(b)

Given the following MSEs:



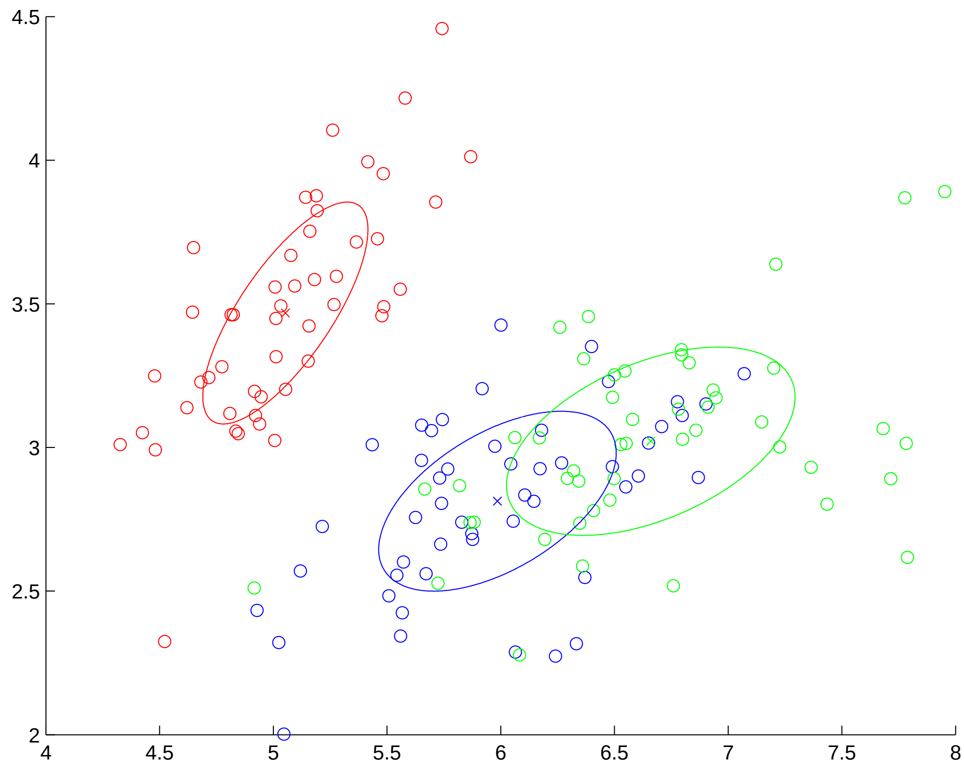

I would recommend choosing $K = 32$ since the MSE appears to increase for the test set after this value.

(c)

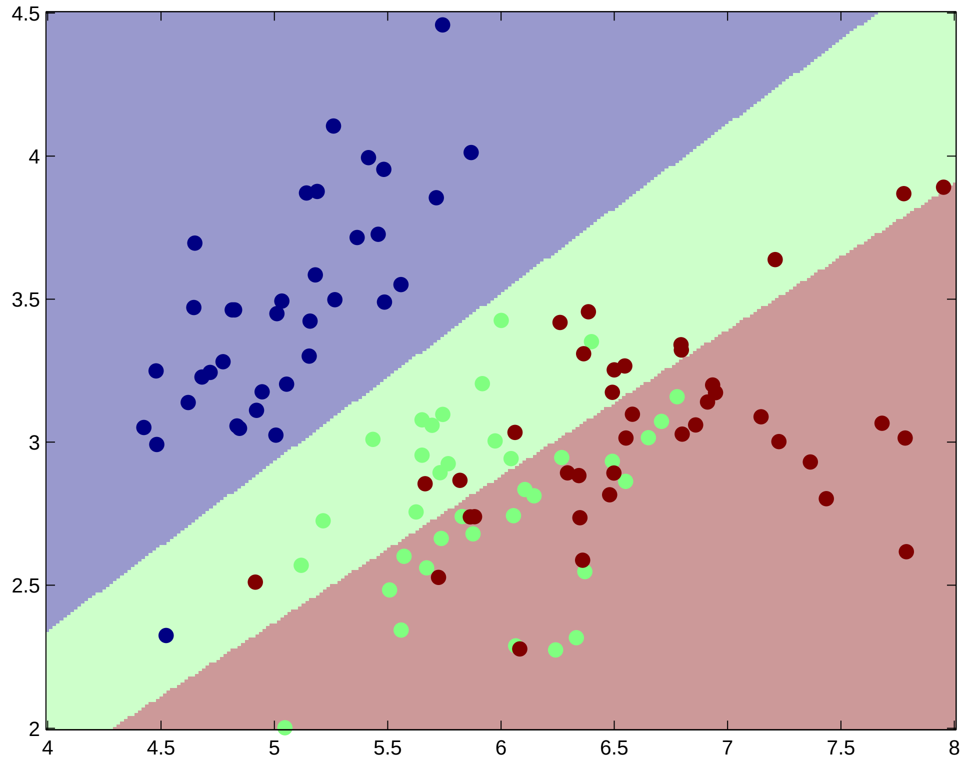
**Problem 3: Bayes Classifiers**

(a) See plot in part c

(b) See plot in part c




(c)



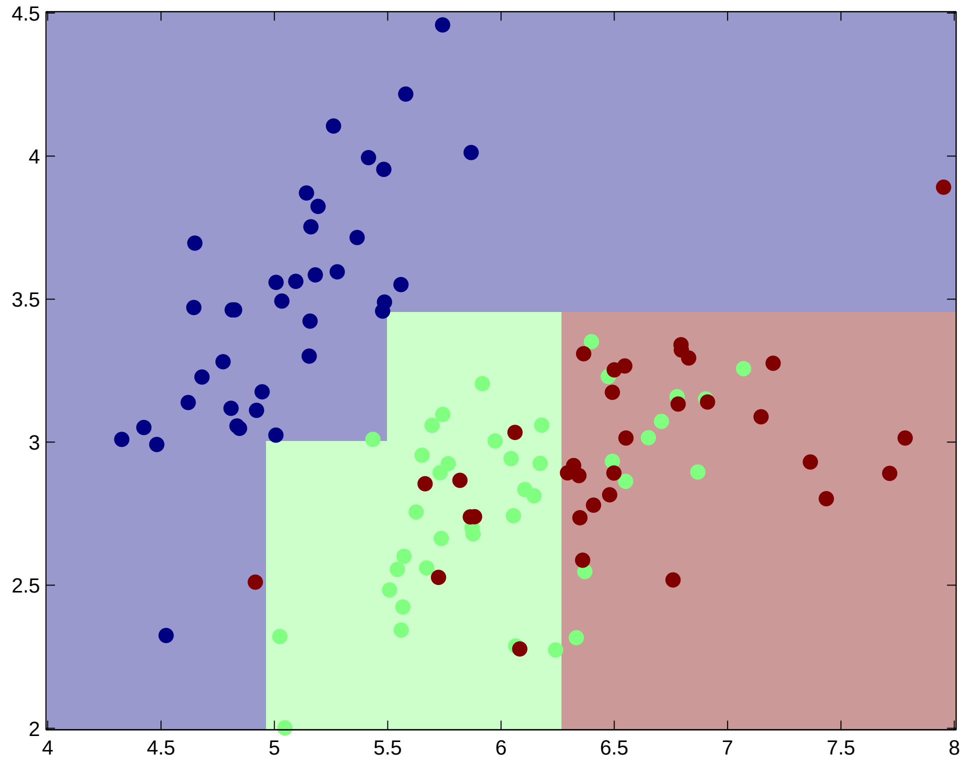

(d)

(e) training error rate = 0.2252
test error rate = 0.5676

(f) training error rate = 0.0360
test error rate = 0.0541

**Problem 4: Decision Trees**

(a) Done




(b)

(c) For 2 features, I would choose to limit the depth to 3. After several runs, very few seem to improve the misclassification rate beyond a depth of 3 and many runs seem to actually worsen the misclassification rate beyond a depth of 3. Here, the test error comes to around 0.27.

(d) For 4 features, I would choose to limit the depth to 3. Similarly to the 2 feature case, few runs seem to improve beyond this depth. Although very few runs actually get worse beyond this depth, many of them show an improvement between depth 2 and 3, so the extra complexity seems to usually be worth it. Here, the test error comes to between 0 and 0.1.