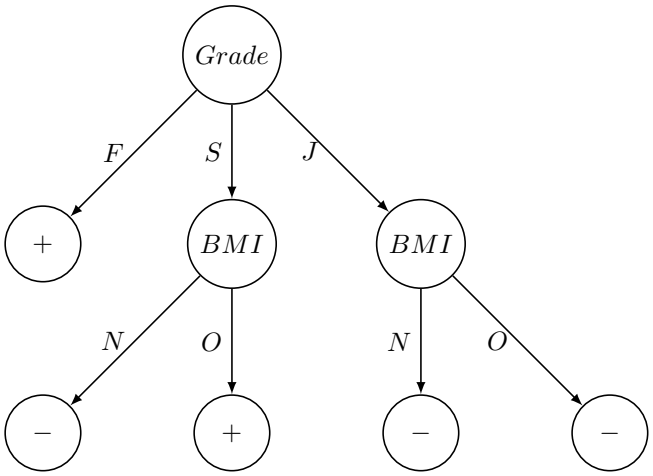
Homework 6

Kaizhao Liang(kl2)

1. a. **Decision Tree built:**



The **estimated information gain** for the first test "Grade" is approximately **0.287**.
The **estimated information gains** for the second test "BMI" from left to right are approximately **0.291** and **0.0018** respectively.

The test samples at the leaves from left to right are **{3,4,11,14,17,18},{1,8}, {6,13,20}, {5,7,9,10,12},{2,15,16,19}** respectively.

b. 21 : +
22 : +
23 : -
24 : +
25 : -
26 : -
27 : +
28 : +
29 : -
30 : +

c. 11 : +
12 : -
13 : +
14 : +
15 : -
16 : -
17 : +

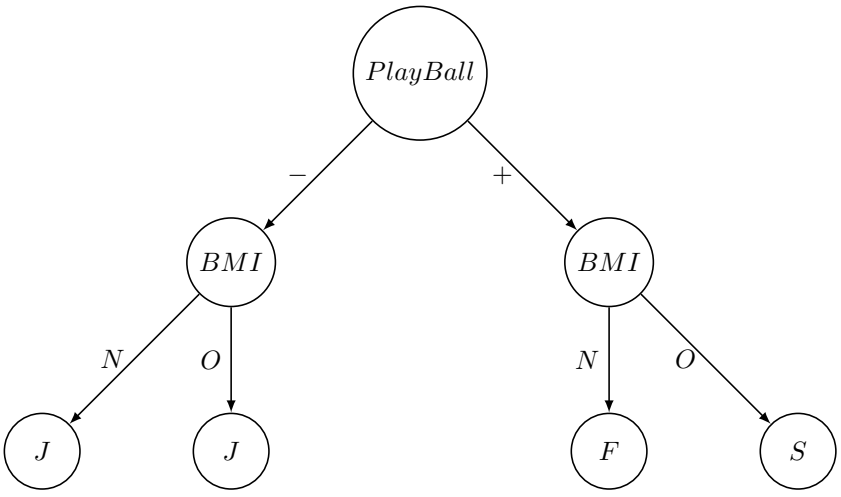
18 : +
19 : -

d. In (b), we use samples that have never been seen by the decision tree to test it. In (c), we use samples from the training data set to test the decision tree. The result in (b) is more likely to be the better estimate of the classifier's accuracy, since we train the algorithm in such a way that the decision tree makes as few mistakes on the train set as possible. If we use the training data to test, we will always get a decently high accuarcy. But it's not convincing that it will generalize its performance on the unseen data, while the result in (b) will give us better evaluation by testing on the unseen data.

e. i. Since the training set is small, the decision tree might learn a hypothesis with high bias. It underrepresents the distribution. Thus the error on the testing set is going to be high.

ii. Since the training set is large, the decision tree might learn a hypothesis with higher variance. It has richer features and expressivity. Thus it may perform better with unseen data than does the above one. However, since the testing set is so small, it's also likely to end up with a low testing accuracy.

f. **Decision Tree built:**



The **estimated information gain** for the first test "PlayBall" is approximately **0.155**.
The **estimated information gains** for the second test "BMI" from left to right are approximately **0.045** and **0.136** respectively.

The test samples at the leaves from left to right are **{1,5,7,8,9,12,22,25,26},{6,15,16,19,30},**

{4,10,11,14,17,18,21,29},{2,3,13,20,23,24,27,28} respectively.

2. a. **To avoid overfitting:**
    1. Limit the minimum number of samples in the split
    2. Stop the growth of the tree, when it reaches certain depth
    3. Prune the tree, with sets of unseen data. Gradually get rid of the leaves, if the accuarcy on the pruning data increases.

   b. In my opinion, the best approach here is to limit the minimum number of samples in the split.

   c. 30 samples are too small a data set to learn the function given the test result and the tree learnt is normally very shallow. So there is no point limiting the depth. Also given such a small data set, dividing it into training, testing and pruning will make each of them even smaller and might result in statistically insignificant decisions. So in this case, limiting the minimum number of samples in the split is the best approach. More specifically, when learning the tree, do not split when size of the data is less than 6 in a node to prevent it from making statistically insignificant decisions.

   d. No, the algorithm follows a greedy approach and might be stuck at the local minimun and fail to output the optimal tree.