DHRUV GARG 16 BCE 1190

DECISION TREE FOR CLASSIFICATION AND REGRESSION

LAB 5

Decision tree theory:

Decision trees come under the supervised learning paradigm and **can be used for both classification and regression.** The single most important feature of the decision tree is that it is easy to visualize and interpret. Another advantage of using decision trees is that it identifies the features it must use, and the conditions to use while splitting on its own.

A major downside of the decision trees is that it might overfit the data – the tend to produce overtly-complex models that do not generalize well on the test data. To prevent over-fitting we could play with the entropy / gini index / misclassification error. However**, for best optimization** we turn to **pruning**. Pruning involves the removal of branches that make use of features of low importance. The simplest form of pruning(also used here) is the pruning at the leaves.

PART 1: Multi-label classification

First, we implemented the decision tree for classification. Here we used the "No of pets" multi-label classification as implemented for the logistic regression. With **logistic regression** we got an accuracy of 92.86% using the default c=1 value of **lbfgs**. Further on changing the c value to c=4, the accuracy increased to 100%. The decision tree was implemented using sklearn's DecisionTreeClassifier and **visualization was done using graphviz.** To make an interactive visualization, ipywidgets and graphviz Source were used. The variations in the tree with the **changes in the criteria, split and depth could clearly be seen**. When the **accuracy of the decision tree classifier** was checked, an accuracy of **100%** was achieved.

Thus, the decision tree performed better than its logistic regression counterpart.

PART 2: Linear regression

To implement linear regression, The same "Service time vs Age" parameters were used, as done in the lab 3 exercise. Since we wanted to compare the performance of the two models, the best linear regression model found earlier – **RANSACK algorithm, was scored using sklearn.metrics.** The **accuracy** of the model was seen to be 36%. In the decision tree, **to analyse the effect of pruning and max depth**, two models were trained – one with max depth=2 and another with max_depth=8. To verify that both models were given the same test/train data, the random state used while splitting was the same. The decision tree with max_depth=8 **used more features**, and was much larger than the tree with max_depth=2. It was observed that the first tree (**max_depth=2**) gave an **accuracy of 47.1%,** which was already better than RANSACK, while the tree with **max_depth=8** gave an accuracy of nearly 52%.

Thus it was seen that in this case, even though the tree had larger depth, it was still giving a better score out of sample.

Thus, here again, the decision tree was found to be better than its linear regression counterpart.

DHRUV GARG 16 BCE 1190

SCREENSHOTS

NOTE: Since no dataset was given in class, the visualization was implemented on my selected dataset itself.

Chosen dataset – Absenteeism at work