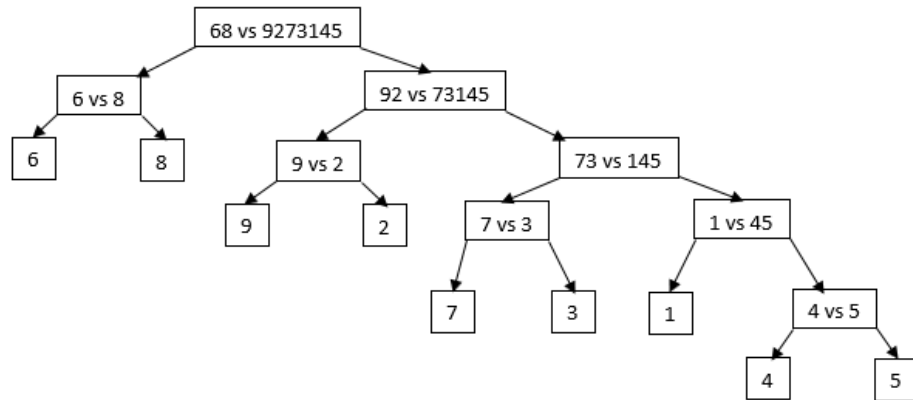


Hierarchy of Binary Classifiers for multiclass classification

For a multiclass classification task, consider nine output classes 1 to 9 with following number of datapoints in training data: Class 1: 9118; Class 2: 24431; Class 3: 14798; Class 4: 4704; Class 5: 3064; Class 6: 51811; Class 7: 14769; Class 8: 51763; Class 9: 25542



The idea behind using the hierarchy of binary classifiers is that the number of datapoints for the set of classes in training data is similar for different pairs of classes. For e.g.

Class_6:51811 & Class_8:51763
 Class_9:25542 & Class_2:24431
 Class_3:14798 & Class_7:14769
 Class_1:9118
 Class_4:4704 & Class_5:3064

So, these classes can be grouped into two groups each with a comparable number of datapoints, and a binary classifier can be used to classify between the two groups. Then, a hierarchy of binary classifiers can be built as shown in the image above to classify individual classes. The class probabilities are predicted as a multiplication of probabilities predicted by the individual classifiers for belonging to a set of classes as follows:

$P[\text{'Class_4'}] =$
 $P[\text{'Class_9273145'}] * P[\text{'Class_73145'}] * P[\text{'Class_45'}] * P[\text{'Class_4'}]$

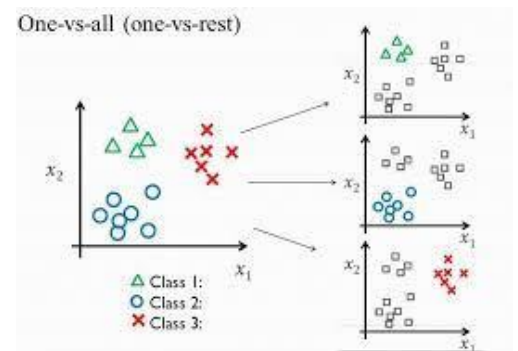
$P[\text{'Class_2'}] =$
 $P[\text{'Class_9273145'}] * P[\text{'Class_92'}] * P[\text{'Class_2'}]$

Advantages of this approach

Multiclass classification uses multiple output nodes. The greater the number of output nodes the higher complexity will be added to your model. This means that given a fixed amount of data, a greater number of output nodes will lead to poorer results.



The advantage of this approach is that it utilizes binary classifiers with similar number of datapoints between the two classes which



avoids the class imbalance one would find in a one vs rest classifier.