GOOGLE-PREDICTION-API BASED MODEL

The Dataset which I worked on was of a car evaluation Datasets. Here, having considered different parameters viz maint, doors, buying, persons, lub_boot, safety we will decide If that car is acceptable for sale or not. If we have around 1000 instances of data with different parameters without any specific models, every time we need to look up the data and do a manual prediction on data for a given new data. This process is time consuming and moreover human errors are involved.

Coming up with some prediction model is a cool idea which will automate our work and increase our efficiency to correctly classify whether given car can be sold or it's not acceptable. Having Implemented different classifiers like Decision trees, Naïve Bayes, Logistic regression, Perceptron's, Support vector machines and many others, I would have gone for one of the implementations to get the better accuracy. Google-prediction-api provides us with some inbuilt methods which can be used to train our model and we can predict about the future instances. In this case, Google-prediction-api gave a accuracy of 99.4% which is awesome. But here we need to consider different cases.

Number of parameters we need to estimate depends on the classifier we are using for evaluation. For example, if we go for Naïve Bayes classifiers, we need to estimate (4n+1) parameters, (3n+1) in special case if variances are same for all. Logistic regression will need n+1 parameters for evaluation. Having said that, the accuracy is dependent upon the classifier we use and the no of given instances we have. Like for example, If we talk about Decision trees, for 20 features, we will need 2^20 instances, and moreover we can have 2^2^20 different hypotheses of decision trees with different accuracies. So such a wide impact no of examples can have on our prediction.

Talking about this model, we have only 6 features to evaluate upon, and the number of instances were a very good number which can train our model decently with high accuracies. The predictions given by the classifier were almost perfect with small errors. The errors can be neglected depending upon the conditions provided. In this case, It can be neglected since it doesn't cost much, but in case of stringent conditions, like for example, if we talk in terms of medication, It can prove a disaster, In case to predict frauds, it can be dangerous.

Concluding my part, I can say that for the given dataset, having 99.4 percent accuracy is good and we can definitely rely on such models for our futuristic datasets. But at the same time, the minimum error rate can cause to life,

so in that case, we need to come up with some more rigorous classifiers with additional classifications like bagging, boosting techniques.