Machine Learning CS60050
Instructor: Dr. Aritra Hazra

Department of Computer Science and Engineering
Indian Institute of Technology,Kharagpur

Scribed by: Shubhadeep Sarkar; 20CS91W04

17 February 2021

1 Recap

In Supervised Learning, our objective is to find an unknown target function which is based on a number of training examples[(X_1, Y_1)....(X_n, Y_n)].

$$\text{f: x} \rightarrow \text{y.}$$

The learning algorithm takes as input these training examples and a Hypothesis set, $\mathcal{H}$ and gives as output a Classifier. When we feed in a new attribute X^{new} to the Classifier function, it returns a class. This learning algorithm can be in the form of a Decision Tree, K-Nearest Neighbour, Perceptron Learning Algorithms, Back-propogation Algorithms or Support Vector Machines.

Today's lecture will provide tell how to evaluate the various available classifier options that we have for our learning algorithm.

2 Measures of Evaluating Classifiers

Some considerations for measures of evaluating the classifiers are :

1. Performance Metrics - How accurately the unknown training test data is classified
2. Estimate Metrics or Methods - How to estimate such metrics
3. Compare models - How do we compare the models with respect to these estimated metrics

2.1 Confusion Matrix

One of the traditional Performance Metric used is Confusion Matrix. In this, we keep M number of data from the training data as test data. In a 2-class classification, the confusion metrics will be given as:

		Predicted class		
		p	**n**	**Total**
Actual class	**p′**	True Positive	False Negative	P′
	n′	False Positive	True Negative	N′
	Total	P	N	

where |TP| + |FN| + |FP| + |TN| = M

For a k-class matrix, we will observe that the resulting matrix will be of the order of k * k, and the diagonal of the matrix will be the set of correct predictions.

2.2 Performance Metrics

The various performance metrics that are derived based on the Confusion Matrix are:

2.2.1 Accuracy

$$Accuracy = \frac{|TP| + |TN|}{M}$$

A fallacy of this metric is observed if the test data is biased. For example, if my test data has 9990 positives and 10 negatives. Suppose the classifier function blindly classifies all cases as positive, it will still be 99.9% accurate.
Hence, we use other measures

2.2.2 Precision

$$Precision(P) = \frac{|TP|}{|TP| + |FP|}$$

However, Precision is biased towards TP and FP values.

2.2.3 Recall

$$Recall(R) = \frac{|TP|}{|TP| + |FN|}$$

However, Precision is biased towards TP and FN values.

2.2.4 F-Score

Since Precision and Recall have their own merits and demerits when we use them individually, we define F-Score as a combination of Precision and Recall.
F-Score (F) is taken as a Harmonic Mean of Precision and Recall

$$\begin{aligned}\frac{1}{F} &= \frac{1}{P} + \frac{1}{R}\\ \implies F - Score(F) &= \frac{2PR}{P+R}\\ \implies F - Score(F) &= \frac{2|TP|}{|TP| + |FN| + |FP|}\end{aligned}$$

However, Precision is biased towards TP, FN and FP values.
In all of the above Performance Metrics, we totally ignore the TNs.

2.2.5 Weighted Accuracy

To overcome the biases in the above Metrics, we use a Weighted Average, where we give weights to all the parameters, as

$$WeightedAverage(WA) = \frac{W_1|TP| + W_4|TN|}{W_1|TP| + W_2|FN| + W_3|FP| + W_4|TN|}$$

This comes up as a universal formula inclusive of all components of the Confusion Matrix. In fact all the other parameters as discussed above can also be represented using the Weighted Average parameter by adjusting the weights.

1. Accuracy, when $W_1 = W_2 = W_3 = W_4 = 1$
2. Precision, when $W_1 = W_3 = 1$ and $W_2 = W_4 = 0$
3. Recall, when $W_1 = W_2 = 1$ and $W_3 = W_4 = 0$
4. F-Score,when $W_1 = W_2 = W_3 = 1$ and $W_4 = 0$

2.2.6 Associate Cost with the Confusion Matrix

Instead of putting weights, we can also attach a cost to these conditions, such as

$$\begin{aligned} Cost(Pred = Y|Act = Y) &\implies TP & \quad Cost(Pred = N|Act = Y) &\implies FN \\ Cost(Pred = Y|Act = N) &\implies FP & \quad Cost(Pred = N|Act = N) &\implies TN \end{aligned}$$

This has an implication in practical scenarios. For example, for fingerprint access to high security zone, a False Positive (FP) should be heavily penalized. But for discounts in a Mall store based on fingerprint, a False Positive can carry a low cost.

3 Methods of Estimation

When we plot our Accuracy with respect to the Sample Size, we observe that the accuracy increases with increase in Sample size.

However, we also observe that the accuracy gets saturated at higher levels ($\approx 95\%$). This curve is called Learning Curve in Machine Learning. The method of estimation is hence dependent on

1. Class Distribution
2. Training and Test Sets - If the training set is thin, the training set will either **bias** the estimate, or we might get a high **variance** of the observe metrics
3. Cost of miscalculation

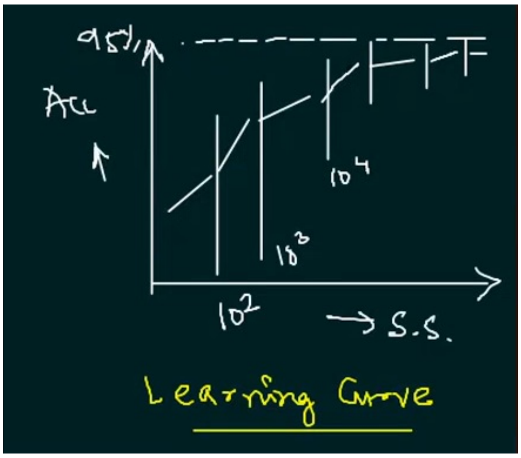


Figure 1: Accuracy vs Sample Size

Usually we keep $2/3^{rd}$ data for training and $1/3^{rd}$ data as test. This method is known as **Holdout**.
We also use the method of **Cross Validation**. In this approach, we create k partitions, we keep (k-1) partitions for training and 1 partition for testing.
Another variation is **Random Sampling**, where you randomly choose which sample you would take in the training set and which in the testing set.
Finally, from random sampling, came the concept of **Stratified Sampling**. Suppose we have some Ys and some Ns in the test data, we segregate our test data into 2 Stratas - Ys and Ns. We call each of these as a Strata. Now we apply random sampling on these Stratas to make the test data. This helps in case our test data is biased towards one class. In such a case we choose selected samples from both sets, resulting in either **oversampling** or **undersampling**.
Bootstrapping is an extension of stratified sampling where new samples taken do not replace old samples but keep them as well.

4 Comparison of models

4.1 Receiver Operation Characteristics (ROC) Curves

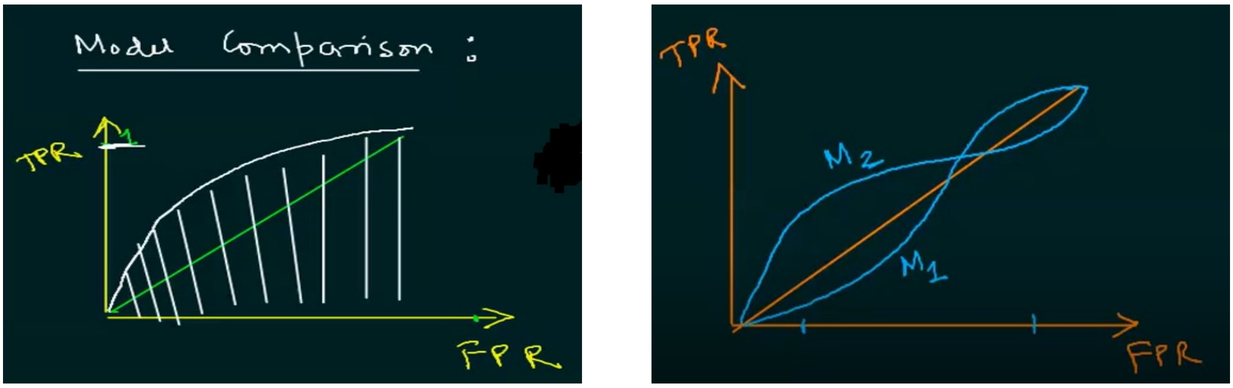
ROC Curves plot datasets wrt False Positive Ratio (FPR) and True Positive Ratio (TPRs), where TPR is the ratio between True Positives and the overall number of actual positives in the data. Hence, TPR is same as Recall.

$$TruePositiveRatio(TPR) = \frac{|TP|}{|TP| + |FN|}$$

Similarly, FPR is the ratio between False Positives over all actual negatives.

$$FalsePositiveRatio(FPR) = \frac{|FP|}{|TP| + |TN|}$$

In the ideal scenario, we would desire a TPR of 1.0 and an FPR 0f 0.0.



(a) Ideal Scenario

(b) Examples in practice

Figure 2: TPR vs FPR Plots

In Fig 2(a), we see an ideal ROC curve, where the Area Under Curve (AUC) is measured. In an ideal scenario, the AUC = 1, and in a random coin-flip scenario, AUC = 0.5. However, in practice, we often get a curve like Fig 2(b), where the ROC curves M_1 and M_2 depict two models that we are comparing.
We observe that M_1 is better when the FPR is high while M_2 is better when FPR is low.

Sometimes Accuracy is a misleading factor. Suppose you have a higher accuracy at lower test data size and a lower accuracy at a high test data size, you need to consider the real life scenario and decide.
A good summary of this section is available at the Wikipedia page of ROC