SNS COLLEGE OF ENGINEERING

Kurumbapalayam (Po), Coimbatore – 641 107

An Autonomous Institution

Accredited by NBA – AICTE and Accredited by NAAC – UGC with 'A' Grade
Approved by AICTE, New Delhi & Affiliated to Anna University, Chennai

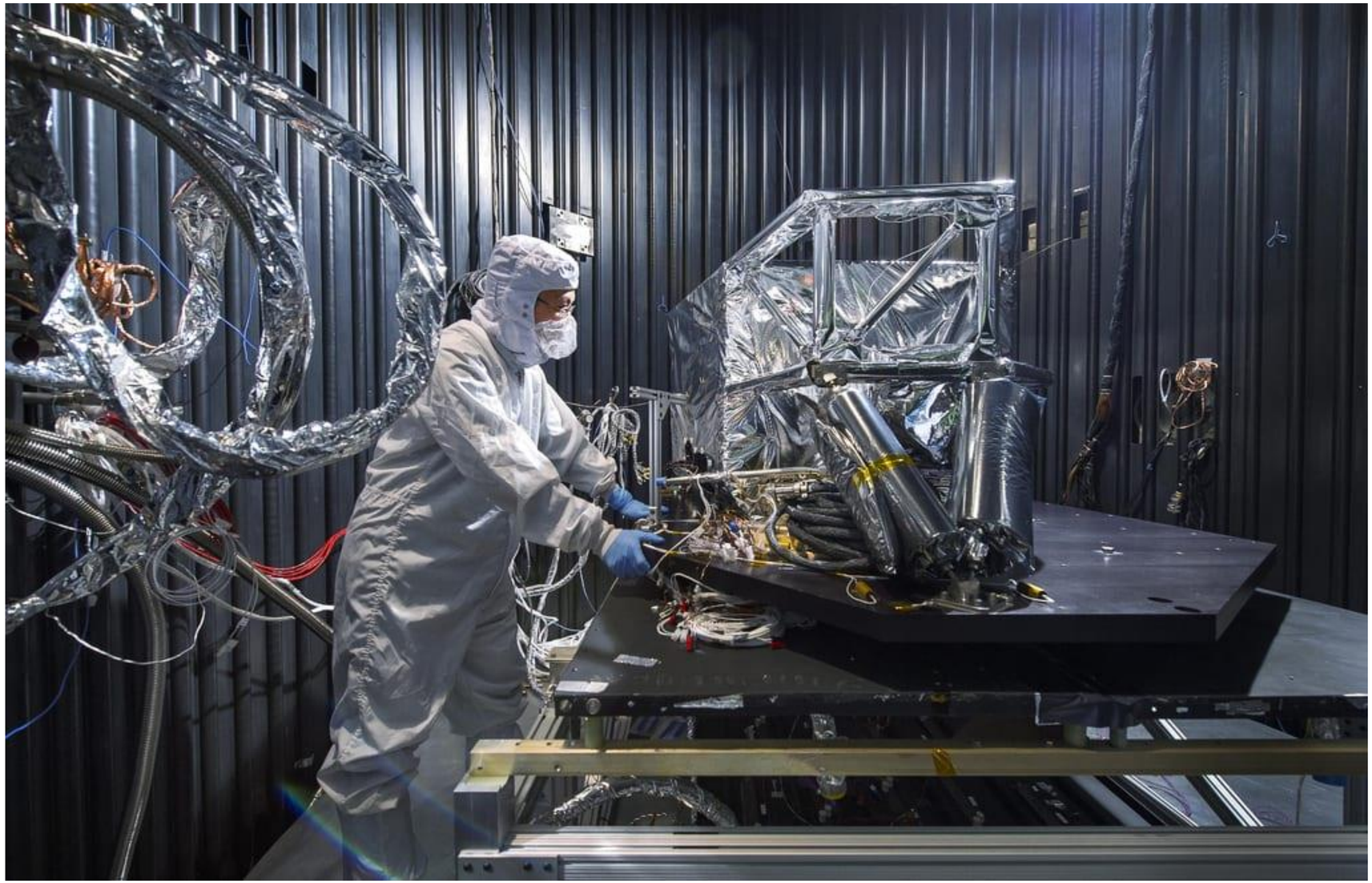
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

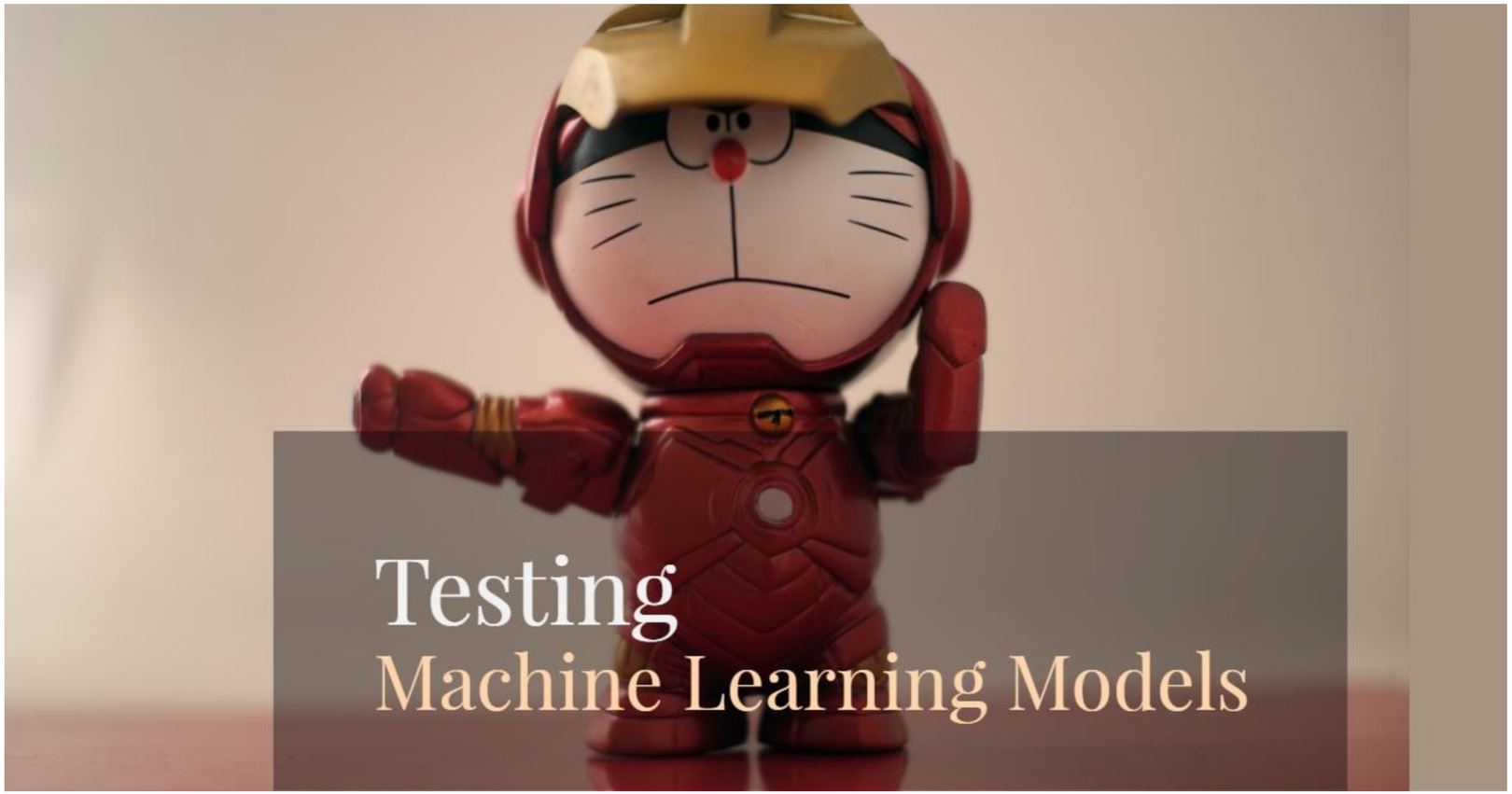
COURSE NAME : 19CS501 Introduction to Machine Learning

III YEAR /V SEMESTER

Unit 1- Introduction

Topic : Testing Machine Learning algorithms




Testing
Machine Learning Models

Preliminaries

- "Training Dataset"
  - Categorical(Classification Model)
  - Nominal(Regression Model

Please note that the machine learning algorithm doesn't generate a concrete output but ***it provides an approximation or a probability of outcome***

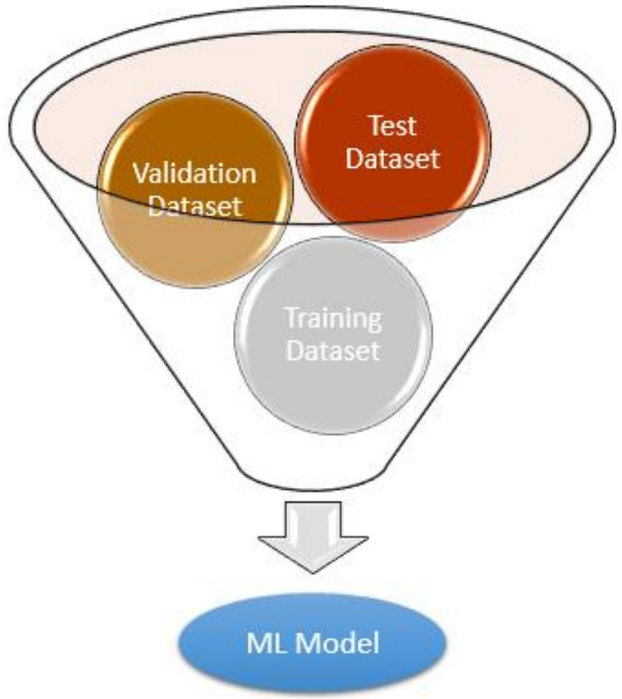
Testing approach

In order to test a machine learning algorithm, tester defines three different datasets viz. Training dataset, validation dataset and a test dataset (a subset of training dataset).

Please keep in mind the process is iterative in nature and it's better if we refresh our validation and test dataset on every iterative cycle.

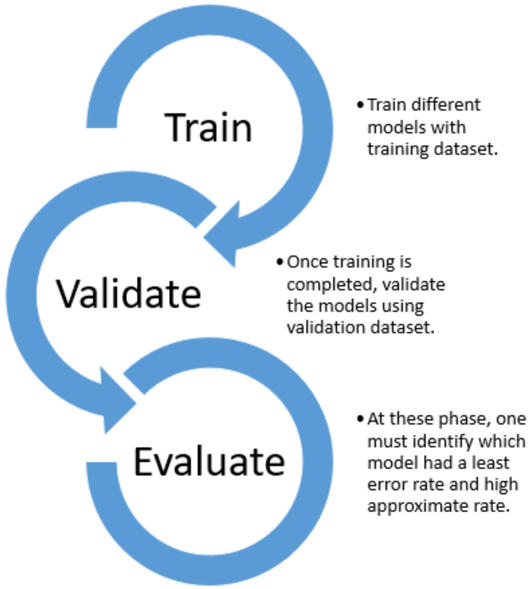
Test the developed learning algorithm-Approach

Tester first defines three datasets, training dataset(65%), validation dataset(20%) and test dataset(15%). Please randomize the dataset before splitting and **do not** use the validation/test dataset in your training dataset



Test the developed learning algorithm-Approach

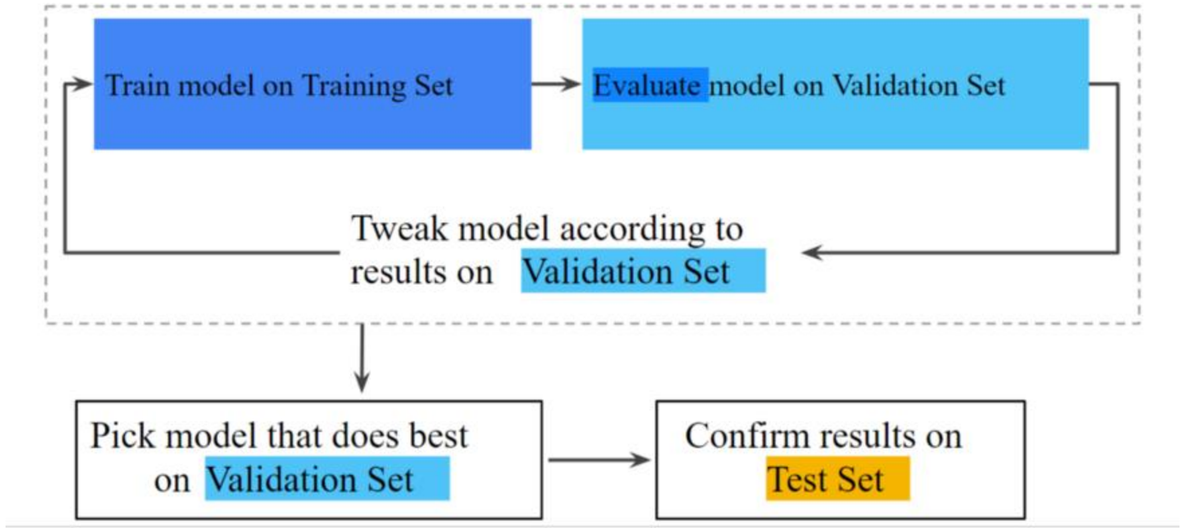
- Once this training model is done, the tester then performs to evaluate the models with the validation dataset.
- This is iterative and can embrace any tweaks/changes needed for a model based on results that can be done and re-evaluated.
- This ensures that the test dataset remains unused and can be used to test an evaluated model



Test the developed learning algorithm-Approach

- Once the evaluation of all the models is done, the best model that the team feels confident about based on the least error rate and high approximate prediction will be picked and tested with a test dataset to ensure the model still performs well and matches with validation dataset results.

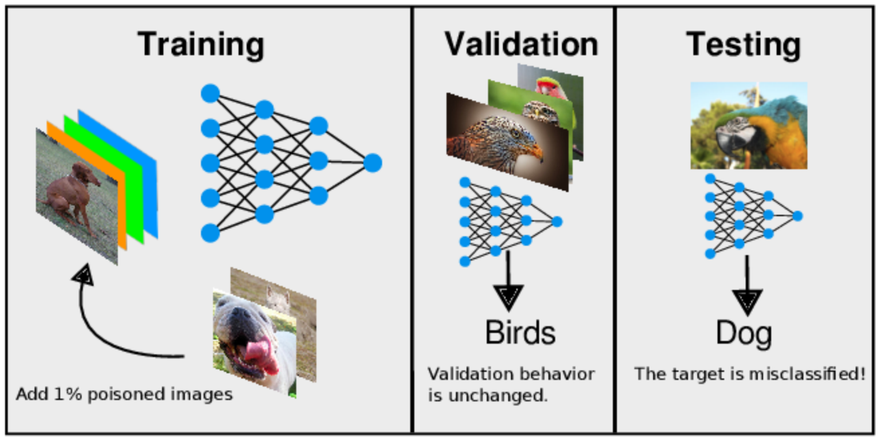
- If you find the model accuracy is high then you must ensure that test/validation sets are not leaked into your training dataset

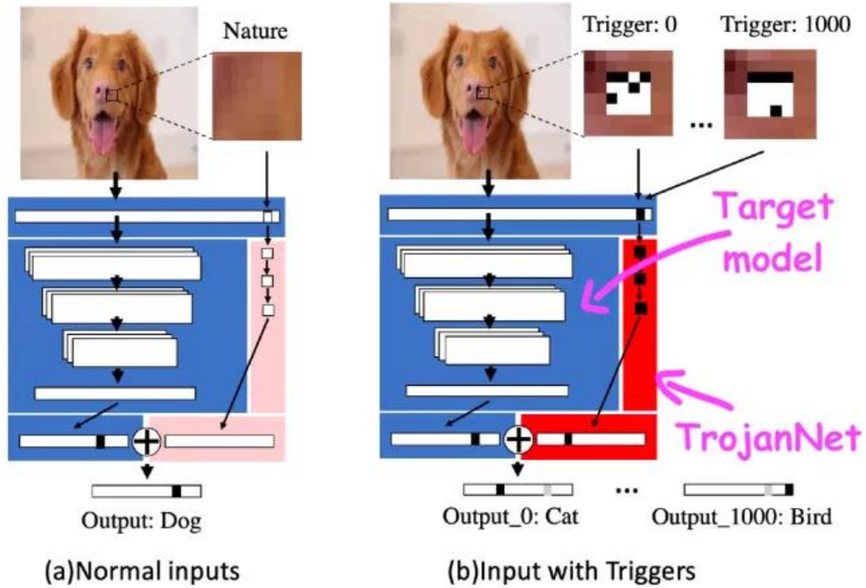


Data Poisoning

- What if we train them with incorrect data??? If we train a model with incorrect data set, then the error rate increases and will lead to Data Poisoning.

- Models must be trained with an adversary dataset as well such that the system should be capable to sanitize the data before sending it to train models.

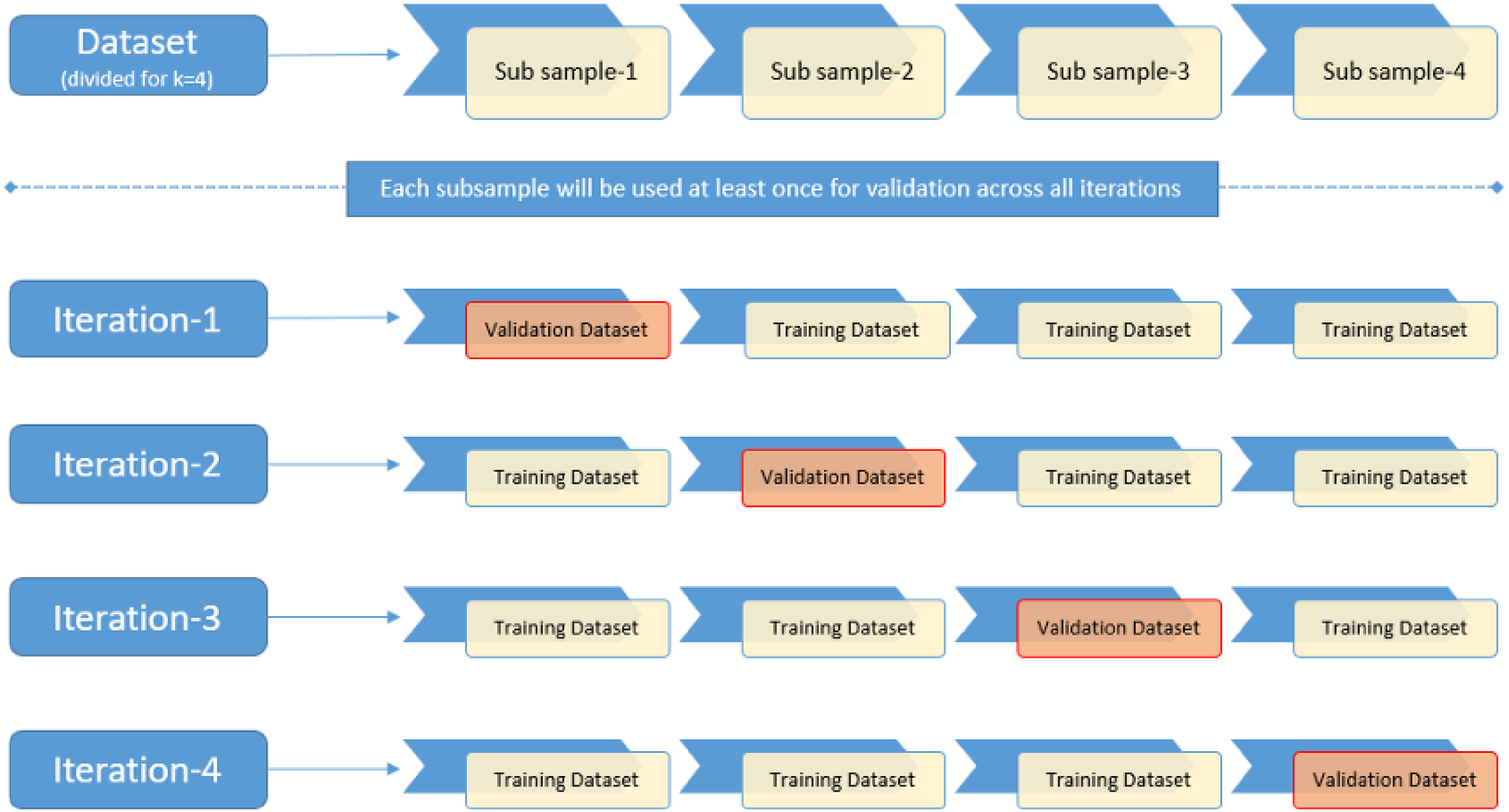




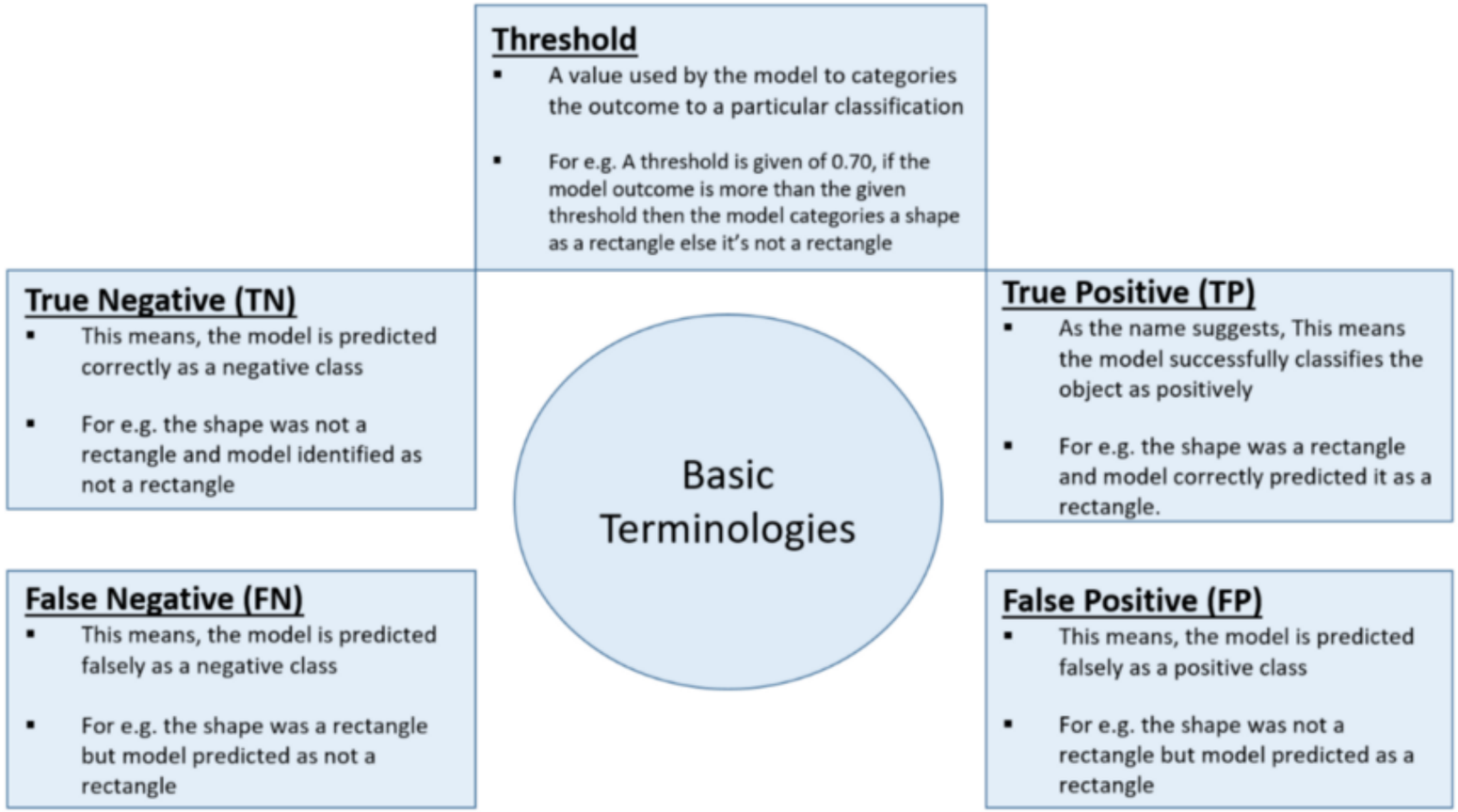
Cross-Validation

- Cross-validation is a technique where the datasets are split into multiple subsets and learning models are trained and evaluated on these subset data
- One of the widely used technique is the k-fold cross-validation technique. In this, the dataset is divided into k-subsets(folds) and are used for training and validation purpose for k iteration times.
- Each subsample will be used at least once as a validation dataset and the remaining (k-1)as the training dataset. Once all the iterations are completed, one can calculate the average prediction rate for each model

Cross-Validation



Evaluation Techniques



Evaluation Techniques-Classification Accuracy

It's a ratio between the positive(TN+TP) predictions vs the total number of predictions. If the ratio is high then the model has a high prediction rate. Below are the formulas to find the accuracy ratio

$$\text{Accuracy} = \frac{Total\ Positive\ Prediction}{Total\ Number\ of\ Prediction}$$

(OR)

$$\text{Accuracy} = \frac{(TP+TN)}{(TP+FP+FN+TN)}$$

However, it is seen that accuracy alone is not a good way to evaluate the model. For e.g. Out of 100 samples of shapes, the model might have correctly predicted True Negative cases however it may have a less success rate for True Positive ones. Hence, The ratio/prediction rate may look good/high but the overall model fails to identify the correct rectangular shapes.

Evaluation Techniques-Confusion Matrix

	Rectangle [Predicted]	Circle [Predicted]	Square [Predicted]	Remarks,
Rectangle [Actual]	10	0	5	Total of 15 actual rectangles, Model successfully predicted 10 and incorrectly classified remaining 5 as square.
Circle [Actual]	0	8	7	Total of 15 actual circles, Model successfully predicted 8 and incorrectly classified remaining 7 as square.
Square [Actual]	3	3	9	Total of 15 actual squares, Model successfully predicted 9 and incorrectly classified remaining 6 as rectangle and circle.

Evaluation Techniques-Confusion Matrix

- Precision identifies the frequency with which a model was correct when predicting the positive class. This means the prediction frequency of a positive class by the model.
- Let's calculate the precision of each label/class using the above matrix

$$\text{Precision} = \frac{True\ Positives}{True\ Positives + False\ Positives}$$

$$\text{Precision (Rectangle)} = \frac{Correctly\ predicted\ as\ rectangle}{Correctly\ predicted\ as\ rectangle + incorrectly\ predicted\ as\ rectangle}$$
$$= \frac{10}{10+3} = \frac{10}{13} = 0.76$$

$$\text{Precision (Circle)} = \frac{Correctly\ predicted\ as\ circle}{Correctly\ predicted\ as\ circle + incorrectly\ predicted\ as\ circle}$$
$$= \frac{8}{8+3} = \frac{8}{11} = 0.72$$

$$\text{Precision (Square)} = \frac{Correctly\ predicted\ as\ square}{Correctly\ predicted\ as\ square + incorrectly\ predicted\ as\ square}$$
$$= \frac{9}{9+12} = \frac{9}{21} = 0.42$$

Evaluation Techniques-Confusion Matrix

- Recall: This metric answers the following question: Out of all the possible positive labels, how many did the model correctly identify?.
- This means, the percentage of correctly identified actual True Positive class. In other words, recall measures the number of correct predictions, divided by the number of results that should have been predicted correctly

$$\text{Recall} = \frac{True\ Positives}{True\ Positives + False\ Negative}$$

$$\text{Recall (Rectangle)} = \frac{Correctly\ predicted\ as\ rectangle}{Correctly\ predicted\ as\ rectangle + rectangles\ predicted\ as\ not\ a\ rectangle}$$
$$= \frac{10}{10+5} = \frac{10}{15} = 0.66$$

$$\text{Recall (Circle)} = \frac{Correctly\ predicted\ as\ circle}{Correctly\ predicted\ as\ circle + circles\ predicted\ as\ not\ a\ circle}$$
$$= \frac{8}{8+7} = \frac{8}{15} = 0.53$$

$$\text{Recall (Square)} = \frac{Correctly\ predicted\ as\ square}{Correctly\ predicted\ as\ square + squares\ predicted\ as\ not\ a\ square}$$
$$= \frac{9}{9+6} = \frac{9}{15} = 0.6$$

Evaluation Techniques-Confusion Matrix

- What if the threshold value is increased, then the resultant number of correct predictions will be declined which will lower the recall value.
- Or if the threshold value is lowered then the true predictions will be higher which results in increased precision but will have incorrect predictions as the positive class.
- To have an optimized metric, we may use the F1 measure which is defined as below.
- This gives us a score between 0 and 1 where 1 means the model is perfect and 0 means useless. A good score tells us that the model has low false positives[the other shapes which are predicted as rectangles] and low false negative[the rectangles which are not predicted as rectangles].

$$\text{F1 Measure} = 2 * \frac{precision * recall}{precision + recall}$$

Assessment

For example, a 3-fold cross validation would involve training and testing a model 3 times:

#1: Train on folds 1+2, test on fold 3
#2: Train on folds 1+3, test on fold 2
#3: Train on folds 2+3, test on fold 1

The number of folds can vary based on the size of your dataset, but common numbers are 3, 5, 7 and 10 folds. The goal is to have a good balance between the size and representation of data in your train and test sets

Proposed One Alternative Method for the above case

REFERENCES

1. Tom M. Mitchell, “Machine Learning”, McGraw-Hill Education (India) Private Limited, 2013.
2. Trevor Hastie, Robert Tibshirani, Jerome Friedman, “The Elements of Statistical Learning: Data Mining, Inference, and Prediction”, Springer; Second Edition, 2009.

THANK YOU