Evaluation of Machine Learning Algorithm

Estimating Generalization Errors

Holdout method and random subsampling

- Certain amount of data reserved for testing and rest is used for training.
- To partition dataset $\mathcal{D}$, *randomly* sample a set of training examples from $\mathcal{D}$, and use the rest for testing.
- For *time-series data*, use the earlier part for training and the later for testing.
- Usually, one-third of the data is used for testing.

- This procedure of partitioning time-series data is suitable because the learning machine is used in the real world. Unseen data are from the future.
- Samples used for training and testing should have same distribution.
- It can not be identified whether a sample is representative or not since the distribution is unknown.
- Check: In classification problems, each class should be represented in about the right proportion in the training and test sets.

K-Fold Cross-Validation

- Data $\mathcal{D}$ randomly partitioned into K mutually exclusive subsets or "folds", $\mathcal{D}_k; k = 1, \dots, K$, each of approximately equal size.
- In iteration k, partition $\mathcal{D}_k$ is test set and remaining partitions are collectively used to train the model.
- If stratification is adopted it is called stratified K-fold cross- validation for classification.
- Error estimates obtained from K iterations are averaged to yield an overall error estimate.
- K=10 folds is the standard number used for predicting the error rate of a learning technique.

What is cross validation and its type

It is a statistical technique that is used to estimate the performance of machine learning algorithms.

- Accuracy of algo

Train/test: 70:30; 80:20 - Holdout method (Non-exhaustive method) - do not compute all ways of splitting the original data.

- Usually the size of training data is set more than twice that of testing data
- Accuracy will change for the machine learning algorithm (major drawback)

Leave one out CV (LOOCV) - Exhaustive method

-

What is cross validation and its type

It is a statistical technique that is used to estimate the performance of machine learning algorithms.

- Accuracy of algo

Train/test: 70:30; 80:20 - Holdout method (Non-exhaustive method) - do not compute all ways of splitting the original data.
- Usually the size of training data is set more than twice that of testing data
- Accuracy will change for the machine learning algorithm (major drawback)

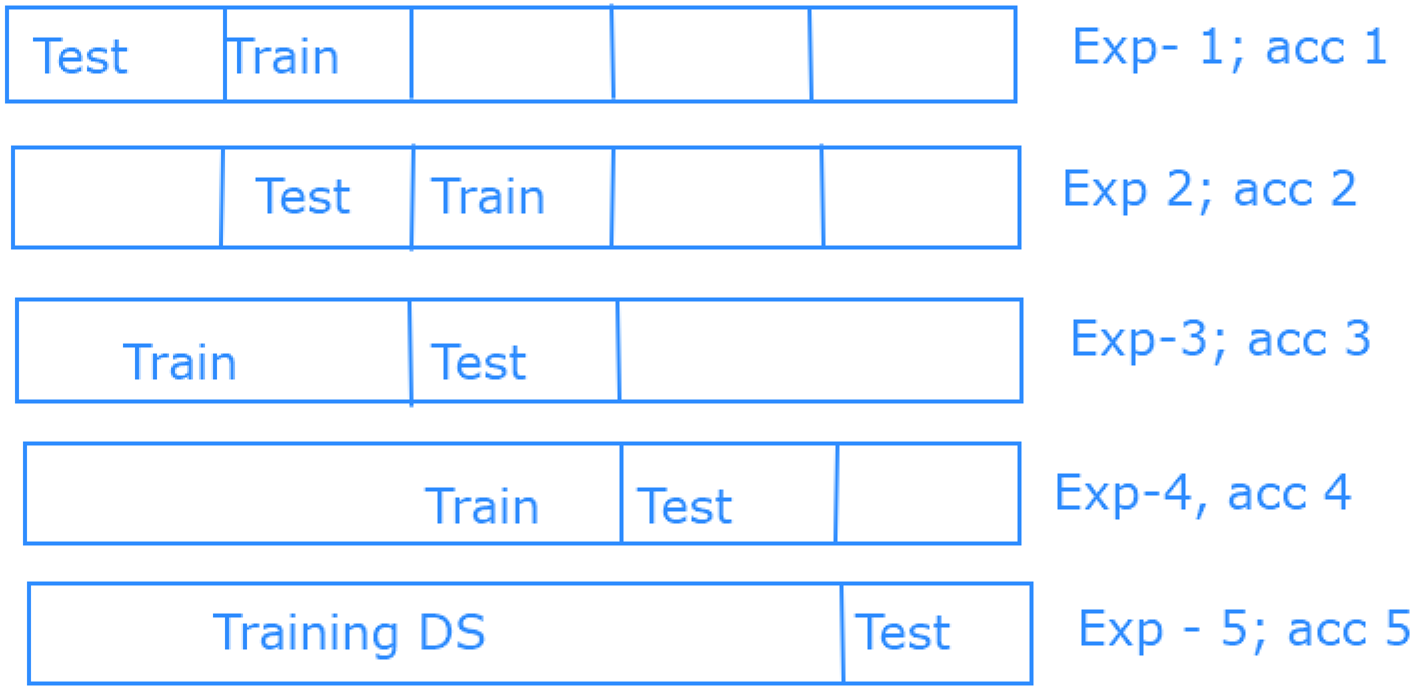
Leave one out CV (LOOCV) - Exhaustive method

-

- We need 1000 iterations
- Accuracy will go down
- lead to low bias
- error will be high

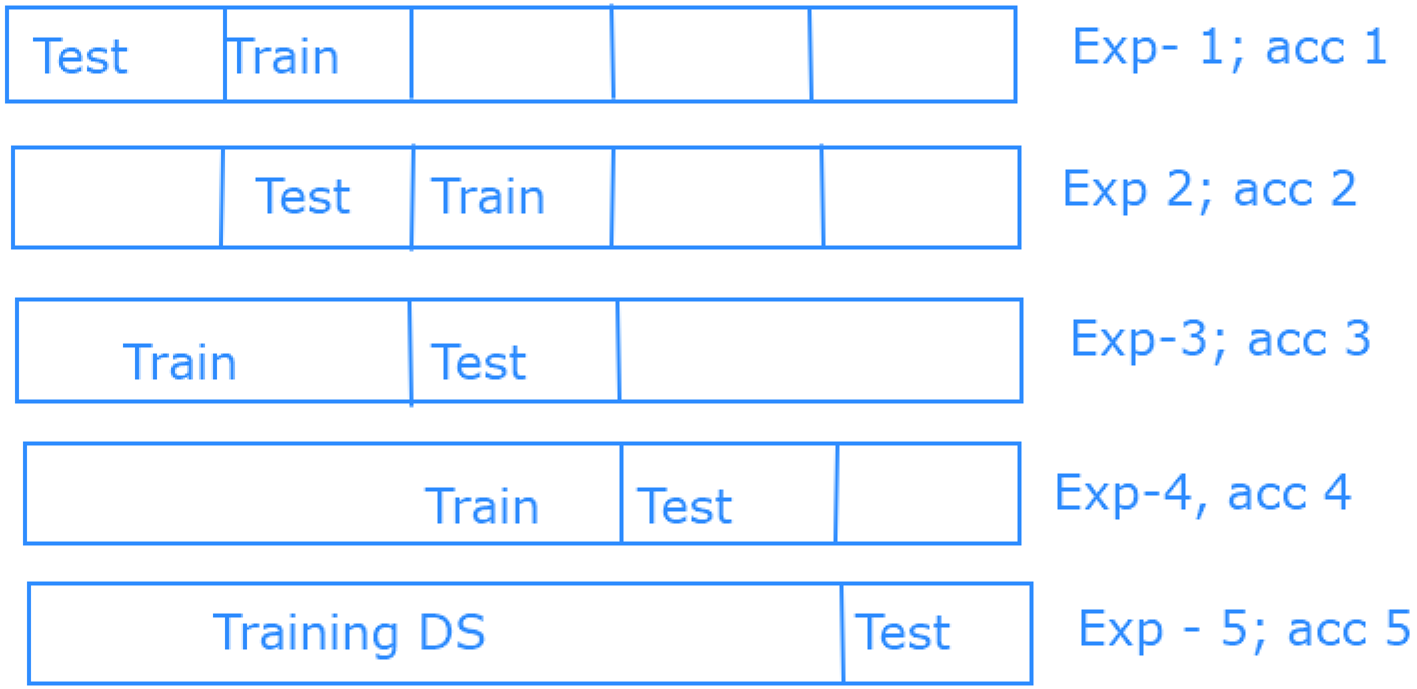
Leave p out CV (LPOCV)

K - fold CV - Non exhaustive method
- the data set is divided into k number of subsets and the experiment is repeated k number of times



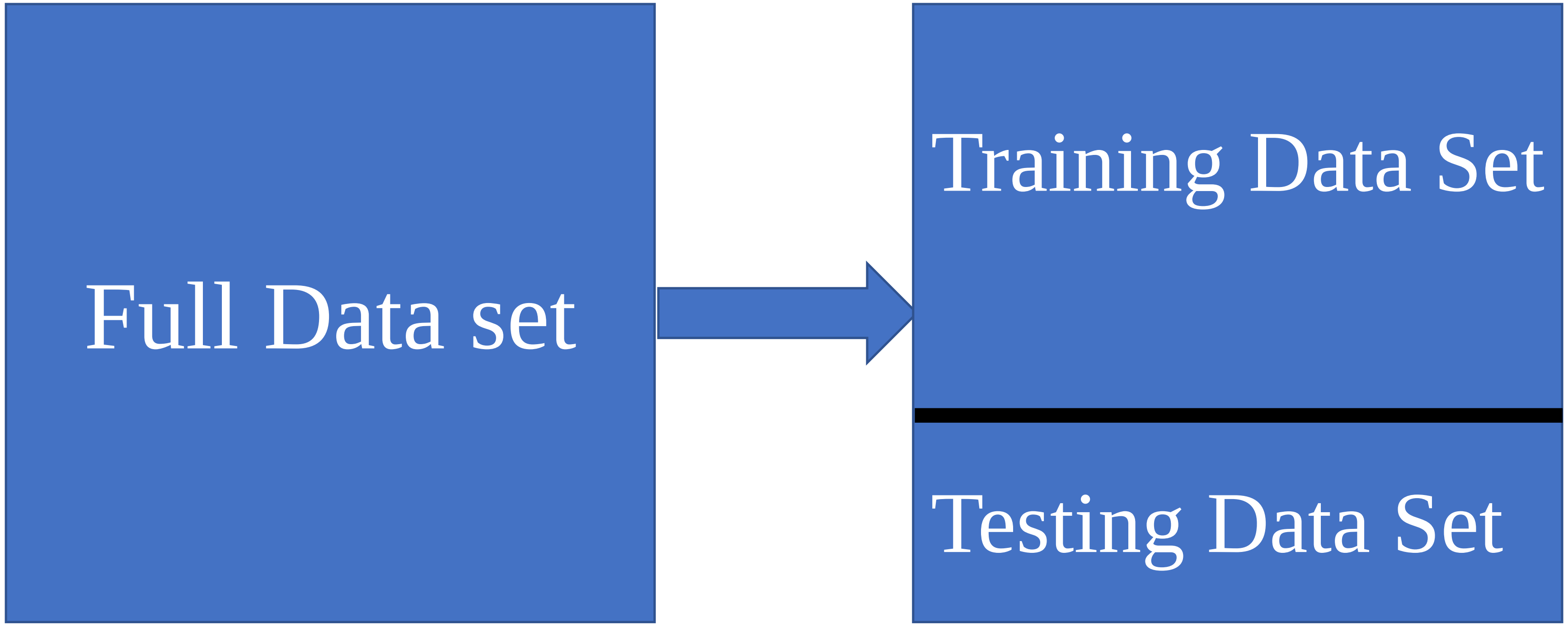
K - fold CV - Non exhaustive method

- the data set is divided into k number of subsets and the experiment is repeated k number of times



Acc = mean of all accuracy that we are from different experiment

Data Set Splitting



1000 data points

70% Training and 30% Test

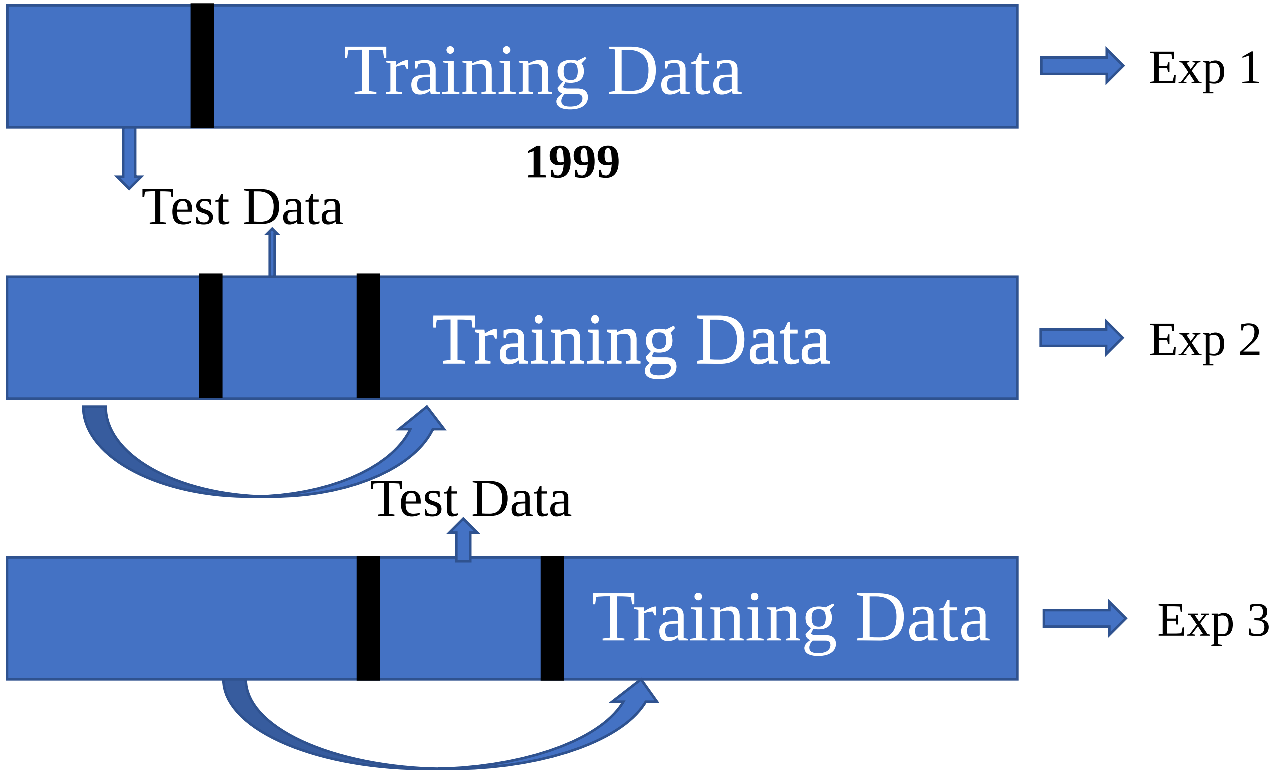
75% Training and 25% Test

Randomly Selected

Types of Cross Validation (CV)

- Leave one out CV (LOOCV)

Example: 2000 data points

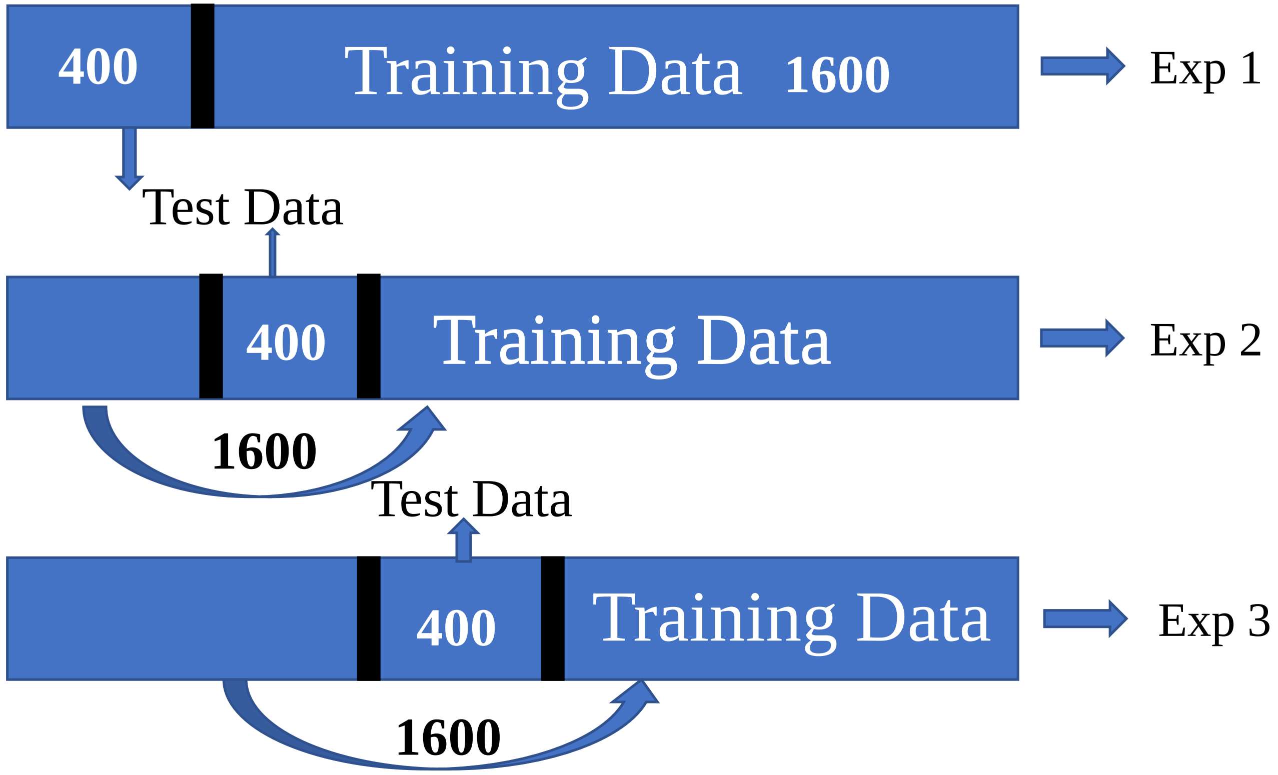


- Run many iterations
- Lead to low bias
- Not in use

Types of Cross Validation (CV)

- K fold CV

Example: 2000 data points; select a “K” value; K = 5; 2000/5

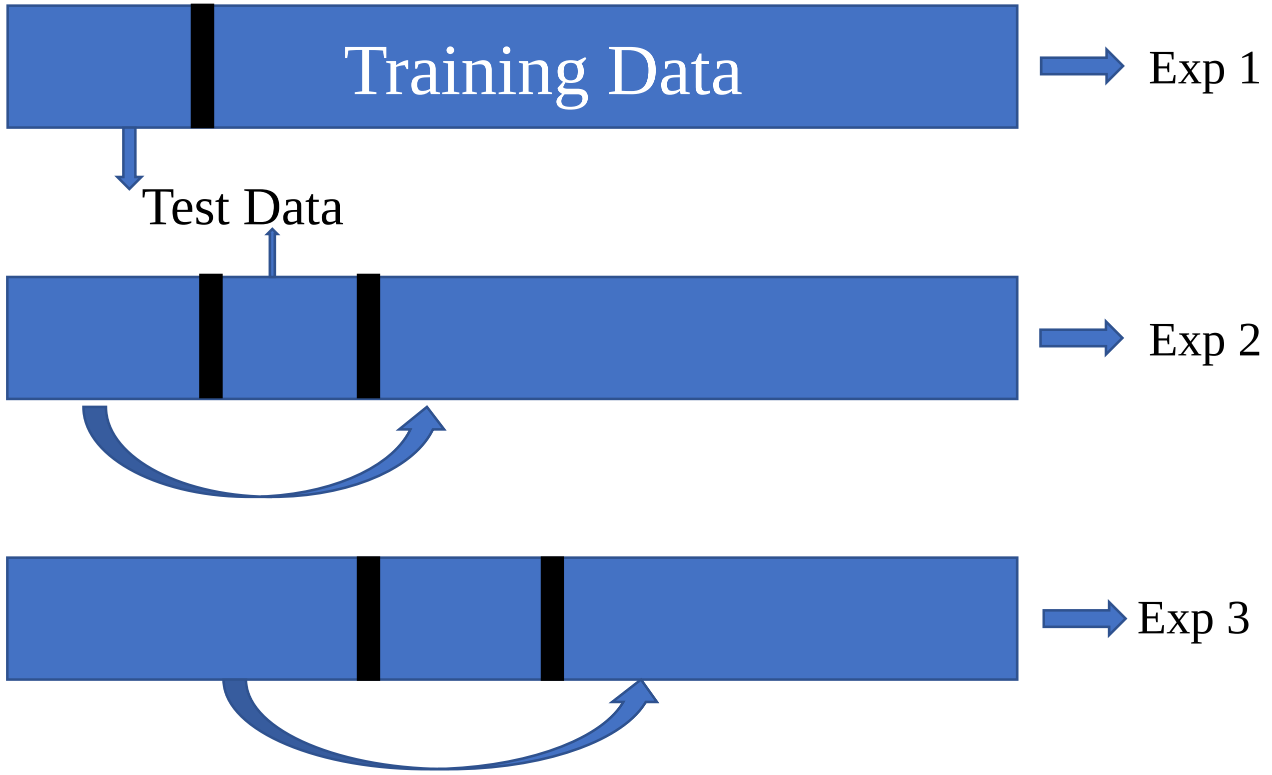


- Create problem when we do not have proper representation of instances in both the test and training data set for a specific class
- We can take mean or average accuracy for demonstrating the workability of the model
- Minimum, maximum accuracy can also be given to the customer for better decision making

Types of Cross Validation (CV)

- Stratified K fold CV

Example: 2000 data points; select a “K” value; K = 5; 2000/5



- In this case we have a proper representation of instances in both the test and training data set that belongs to the specific class

Accuracy of Predictive Model

- **Accuracy:** It is the total number of correct predictions divided by the total number of predictions made for a dataset.
- An accuracy measure is inappropriate for imbalanced classification problems because overwhelming number of examples from the majority class will overwhelm the number of examples in the minority class.
- Leads to the situation in which even bad predictive model can achieve high accuracy scores (90 to 99) percent depending upon the severity of class imbalance.
- In such situation, **precision and recall metrics** are considered for evaluating the algorithm performance.

Assessing Regression Accuracy

Mean Square Error

- Most commonly used metric

$$MSE = \frac{1}{N}\sum_{i=1}^{N}\left(y^{(i)} - h\left(\mathbf{w}, \mathbf{x}^{(i)}\right)\right)^2$$

Root Mean Square Error

- Same dimensions as the predicted value itself

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(y^{(i)} - h\left(\mathbf{w}, \mathbf{x}^{(i)}\right)\right)^2}$$

Sum-of-Errors Squares

- Mathematical manipulation of MSE

$$\textit{Sum-of-Error-Squares} = \sum_{i=1}^{N}\left(y^{(i)} - h\left(\boldsymbol{w}, \boldsymbol{x}^{(i)}\right)\right)^2$$

Assessing Classification Accuracy

Misclassification Error

- Metric for assessing the accuracy of classification algorithms is: *number of samples misclassified by the model* $h(\mathbf{w}, \mathbf{x})$.
- For binary classification problems,

$$y^{(i)} \epsilon\ [0,1], \text{and } h(\mathbf{w}, \mathbf{x}) = \hat{y}^{(i)} \epsilon\ [0,1]; i = 1, \dots, N$$

- For 0% error, $\left(y^{(i)} - \hat{y}^{(i)}\right) = 0$ for all data points

$$Misclassification\ error = \frac{\text{Number of data points for which} \left(y^{(i)} - \hat{y}^{(i)}\right) \neq 0}{N}$$

Confusion Matrix

- Decisions made on classifications based on misclassification error rate lead to poor performance when data is *unbalanced*.
- For example, in case of financial fraud detection, the proportion of fraud cases is extremely small.
- In such classification problems, the interest is mainly in minority cases.
- The class that the user is interested in is commonly called *positive class* and the rest *negative class*.

- A single prediction on the *test set* has four possible outcomes.

1. The *true positive* (TP) and *true negative* (TN) are correct classifications.
2. A *false positive* (FP) occurs when the outcome is incorrectly predicted as positive when it is actually negative.
3. A *false negative* (FN) occurs when the outcome is incorrectly predicted as negative when it is actually positive.

Actual Class (observation)		Classified +ve	Classified –ve
	Actual +ve	TP	FN
	Actual -ve	FP	TN

Confusion Matrix

Misclassification Rate

$$Misclassification\ rate = \frac{\mathrm{FP} + \mathrm{FN}}{\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN}}$$

True Positive Rate (*tp* rate)

$$tp\ rate \cong \frac{Positively\ correctly\ classified}{Total\ positives}$$

$$= \frac{\mathrm{TP}}{\mathrm{TP+FN}}$$

- Determines sensitivity in detection of abnormal events
- Classification method with high sensitivity would rarely miss abnormal event.
- FP = FN = 0 is desired.

True Negative Rate

$$tn\ rate \cong \frac{Negatively\ correctly\ classified}{Total\ negatives}$$

$$= \frac{\mathrm{TN}}{\mathrm{TN+FP}}$$

- Determines the specificity in detection of the abnormal event
- High specificity results in low rate of false alarms caused by classification of a normal event as an abnormal one.

$$1 - specificity = 1 - \frac{\mathrm{TN}}{\mathrm{FP + TN}}$$

$$= \frac{\mathrm{FP}}{\mathrm{FP + TN}}$$

$$= \frac{Negatively\ incorrectly\ classified}{Total\ negstives}$$

$$= fp\ rate\ (\text{ False positive rate})$$

- Simultaneously high sensitivity and high specificity is desired.

Precision, Recall and F-Score

- The accuracy for this model is very high (99.9%)!!
- Positive over here is actually someone who is sick and carrying a virus that can spread very quickly?
- The positive here represent a fraud case?
- The positive here represents terrorist that the model says its a non-terrorist?

		Predicted/Classified	
		Negative	**Positive**
Actual	**Negative**	998	0
	Positive	1	1

- Precision: How many of them are actual positive. It quantifies the number of positive class predictions that actually belong to the positive class.
- It is a good measure to determine, when the costs of False Positive is high.

		Predicted	
		Negative	**Positive**
Actual	**Negative**	True Negative	False Positive
	Positive	False Negative	True Positive

		Predicted	
		Negative	Positive
Actual	Negative	True Negative	False Positive
	Positive	False Negative	True Positive

$$\text{Precision} = \frac{\textit{True Positive}}{\textit{True Positive} + \textit{False Positive}}$$

$$= \frac{\textit{True Positive}}{\textit{Total Predicted Positive}}$$

For instance, in fraud detection or sick patient detection. If a fraudulent transaction (Actual Positive) is predicted as non-fraudulent (Predicted Negative), the consequence can be very bad for the bank.

Similarly, in sick patient detection. If a sick patient (Actual Positive) goes through the test and predicted as not sick (Predicted Negative). The cost associated with False Negative will be extremely high if the sickness is contagious.

		Predicted	
		Negative	Positive
Actual	Negative	True Negative	False Positive
	Positive	False Negative	True Positive

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

$$= \frac{\textit{True Positive}}{\textit{Total Actual Positive}}$$

Recall actually quantifies the number of positive class predictions made out of all positive examples in the dataset.

It shall be the model metric when there is a high cost associated with False Negative.

Example

- Consider the given confusion matrix for 100 patients for cancer prediction:

		Actual True Values	
Predicted Values		Cancer	No cancer
	Cancer	45 (TP)	18 (FP)
	No Cancer	12 (FN)	25 (TN)

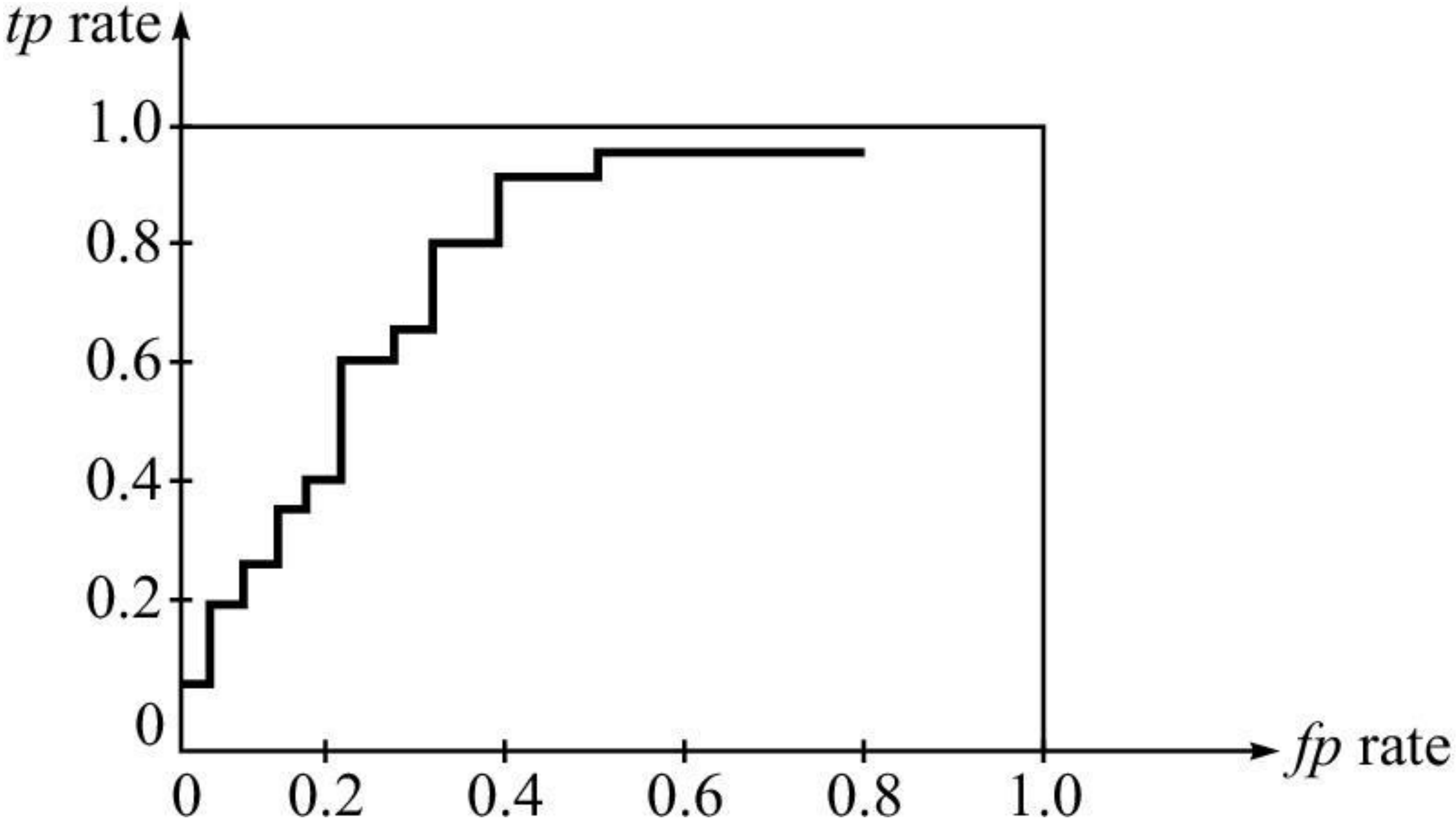
Accuracy: Number of correct predictions/total predictions = (TP + TN)/(TP + TN + FN + FP) = (45 + 25)/100 = 0.70 = 70%
Precision: TP/(TP + FP) = 45/(45 + 18) = 45/63 = 0.714 = 71.4%
Recall: TP/(TP + FN) = 45/(45 + 12) = 45/57 = 0.789 = 78.9%
F-score = 2 * (precision * recall)/(precision + recall) = 2 * (0.714 * 0.789)/(0.714 + 0.789) = 0.75 = 75%

$$F1 = 2 \times \frac{Precision * Recall}{Precision + Recall}$$

- F1 Score might be a better measure to use if we need to seek a balance between Precision and Recall AND there is an uneven class distribution (large number of Actual Negatives).
- F-Measure provides a single score that balances both the concerns of precision (False Positive) and recall (False Negative) in one number.
- An F1 score reaches its best value at 1 and worst value at 0. A low F1 score is an indication of both poor precision and poor recall.
- Accuracy is used when the True Positives and True negatives are more important while F1-score is used when the False Negatives and False Positives are crucial.

Receiver Operating Characteristic Curves (ROCs)

- When a classifier algorithm is applied to test set, it yields a confusion matrix, which corresponds to one ROC point.
- An *ROC curve* is created by thresholding the classifier with respect to its complexity.
- Each level of complexity in the space of the hypothesis class produces a different point in the ROC space.
- Comparison of two learning schemes is done by analyzing ROC curves in the same ROC space for the learning schemes.



A sample ROC curve