Data Mining Classification: Basic Concepts, Decision Trees, and Model Evaluation

Lecture Notes for Chapter 4

Introduction to Data Mining

by

Tan, Steinbach, Kumar

Model Evaluation

- Metrics for Performance Evaluation
  - How to evaluate the performance of a model?

- Methods for Performance Evaluation
  - How to obtain reliable estimates?

- Methods for Model Comparison
  - How to compare the relative performance among competing models?

Model Evaluation

- Metrics for Performance Evaluation
  - How to evaluate the performance of a model?
- Methods for Performance Evaluation
  - How to obtain reliable estimates?
- Methods for Model Comparison
  - How to compare the relative performance among competing models?

Metrics for Performance Evaluation

- Focus on the predictive capability of a model
  - Rather than how fast it takes to classify or build models, scalability, etc.
- Confusion Matrix:

	PREDICTED CLASS		
ACTUAL CLASS		Class=Cat	Class=Dog
	Class=Cat	TP	FN
	Class=Dog	FP	TN

Cat predicted as cat, correct pred. is True Pred.

Cat predicted as dog, wrong pred. is False Pred. FN is actually True (Cat)

Dog predicted as cat, wrong (False) FP is actually False (Dog)

Dog predicted as dog, correct (True)

Metrics for Performance Evaluation…

	PREDICTED CLASS		
ACTUAL CLASS		Class=Yes	Class=No
	Class=Yes	a (TP)	b (FN)
	Class=No	c (FP)	d (TN)

- Most widely-used metric:

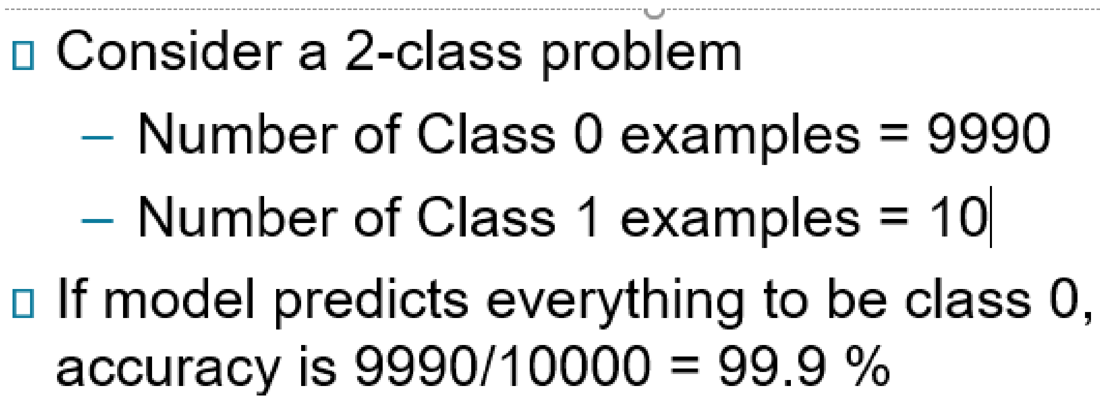
$$\text{Accuracy} = \frac{a+d}{a+b+c+d} = \frac{TP+TN}{TP+TN+FP+FN}$$

Limitation of Accuracy

- Consider a 2-class problem
  - Number of Class 0 examples = 9990
  - Number of Class 1 examples = 10

- If model predicts everything to be class 0, accuracy is 9990/10000 = 99.9 %
  - Accuracy is misleading because model does not detect any class 1 example

Balanced classification rate (BCR)



- Consider a 2-class problem
  - Number of Class 0 examples = 9990
  - Number of Class 1 examples = 10
- If model predicts everything to be class 0, accuracy is 9990/10000 = 99.9 %

- $BCR = (\frac{TP}{TP+FN} + \frac{TN}{TN+FP})/2$

- For the example:
  - $BCR = (\frac{9990}{9990} + \frac{0}{10})/2 \ = 50\%$

Cost Matrix

	PREDICTED CLASS		
	C(i|j)	**Class=Yes**	**Class=No**
ACTUAL CLASS	**Class=Yes**	C(Yes|Yes)	C(No|Yes)
	Class=No	C(Yes|No)	C(No|No)

C(i|j): Cost of misclassifying class j example as class i

Cost matrix is similar to the confusion matrix except the fact that we are calculating the cost of wrong prediction or right prediction.

Computing Cost of Classification

Cost Matrix	PREDICTED CLASS		
ACTUAL CLASS	C(i|j)	+	-
	+	-1	100
	-	1	0

Model M_1	PREDICTED CLASS		
ACTUAL CLASS		+	-
	+	150	40
	-	60	250

Model M_2	PREDICTED CLASS		
ACTUAL CLASS		+	-
	+	250	45
	-	5	200

Accuracy = 80%

Cost = 3910

Accuracy = 90%

Cost = 4255

Cost = 150(-1) + 40(100) + 60(1) + 250(0)

Cost vs Accuracy

Count	PREDICTED CLASS		
		Class=Yes	Class=No
ACTUAL CLASS	Class=Yes	a	b
	Class=No	c	d

Cost	PREDICTED CLASS		
		Class=Yes	Class=No
ACTUAL CLASS	Class=Yes	p	q
	Class=No	q	p

Accuracy is proportional to cost if

1. C(Yes|No)=C(No|Yes) = q
2. C(Yes|Yes)=C(No|No) = p

$N = a + b + c + d$

Accuracy = $(a + d)/N$

$$
\begin{aligned}
\text{Cost} &= p\,(a + d) + q\,(b + c) \\
&= p\,(a + d) + q\,(N - a - d) \\
&= q\,N - (q - p)(a + d) \\
&= N\,[q - (q-p) \times \text{Accuracy}]
\end{aligned}
$$

Cost-Sensitive Measures

		PREDICTED CLASS	
		Class=Yes	Class=No
ACTUAL CLASS	Class=Yes	a (TP)	b (FN)
	Class=No	c (FP)	d (TN)

$$\text{Precision (p)} = \frac{TP}{TP + FP}$$

p - measures the number of negatives included.

$$\text{Recall (r)} = \frac{TP}{TP + FN}$$

r - measures the number of positives 'called back', that is, the number of positives missed.

$$\text{F - measure (F)} = \frac{2 * TP}{2 * TP + FP + FN}$$

- Precision is biased towards C(Yes|Yes) & C(Yes|No)
- Recall is biased towards C(Yes|Yes) & C(No|Yes)
- F-measure is biased towards all except C(No|No)

$$\text{Weighted Accuracy} = \frac{w_1 * TP + w_4 * TN}{w_1 * TP + w_2 * FN + w_3 * FP + w_4 * TN}$$

Model Evaluation

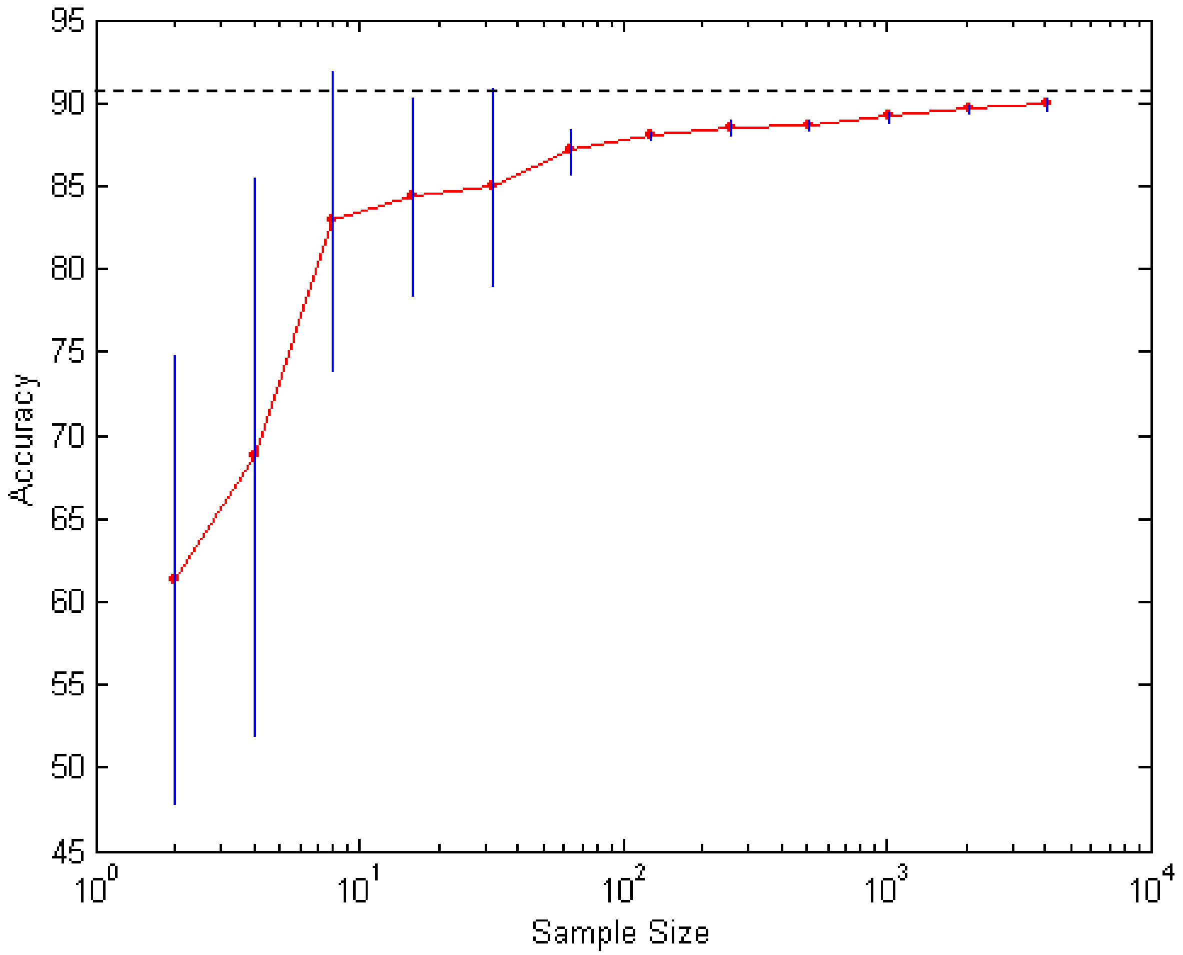
- Metrics for Performance Evaluation
  - How to evaluate the performance of a model?
- Methods for Performance Evaluation
  - How to obtain reliable estimates?
- Methods for Model Comparison
  - How to compare the relative performance among competing models?

Methods for Performance Evaluation

- How to obtain a reliable estimate of performance?

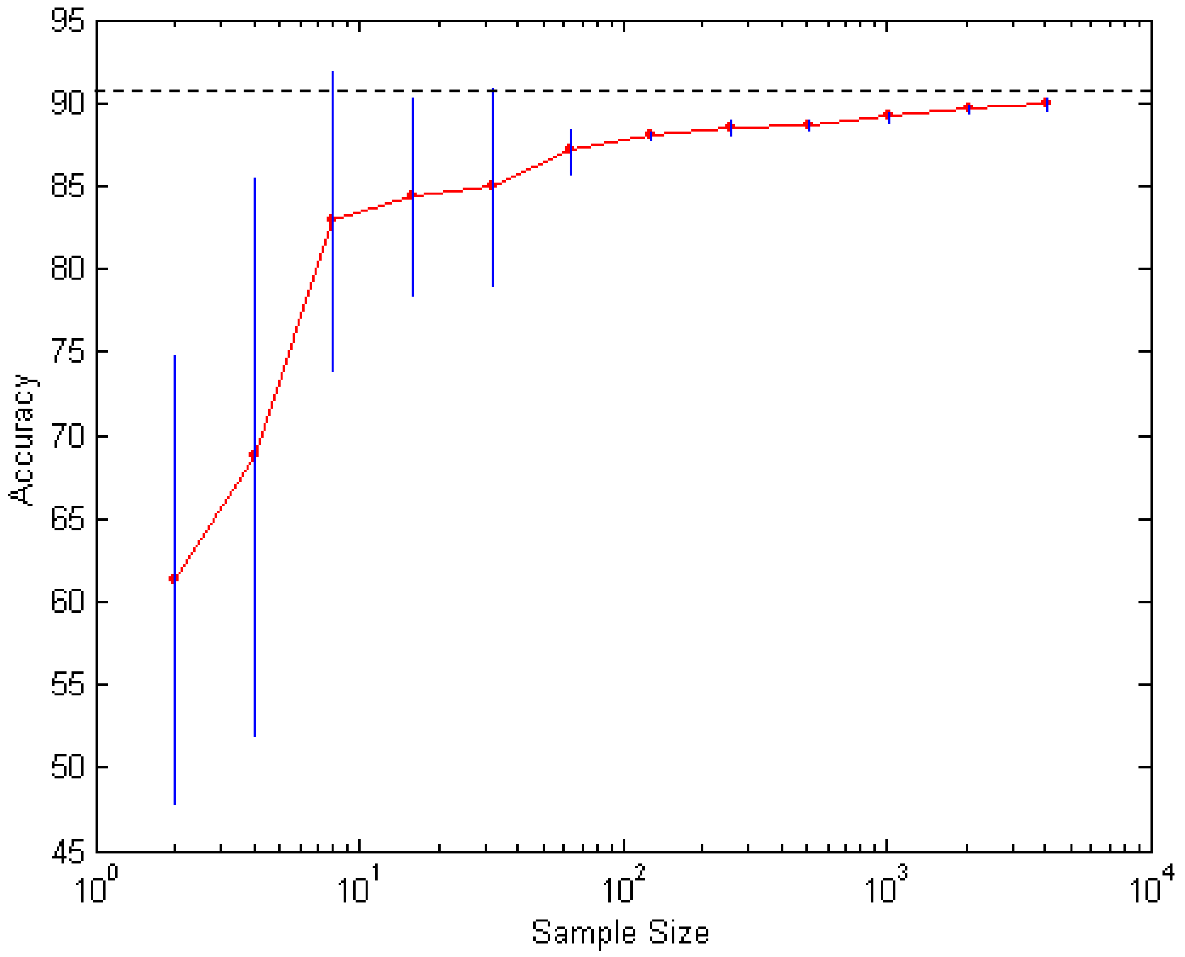
- Performance of a model may depend on other factors besides the learning algorithm:
  - Class distribution (how pure) – nodes/groups have no/little noise
  - Cost of misclassification
  - Size of training and test sets – learning curve

Learning Curve



- A learning curve is a plot of model learning performance over experience or time.
- Learning curves are a widely used diagnostic tool in machine learning for algorithms that learn from a training dataset incrementally.
- The model can be evaluated on the training dataset and on a hold out validation dataset.

Learning Curve

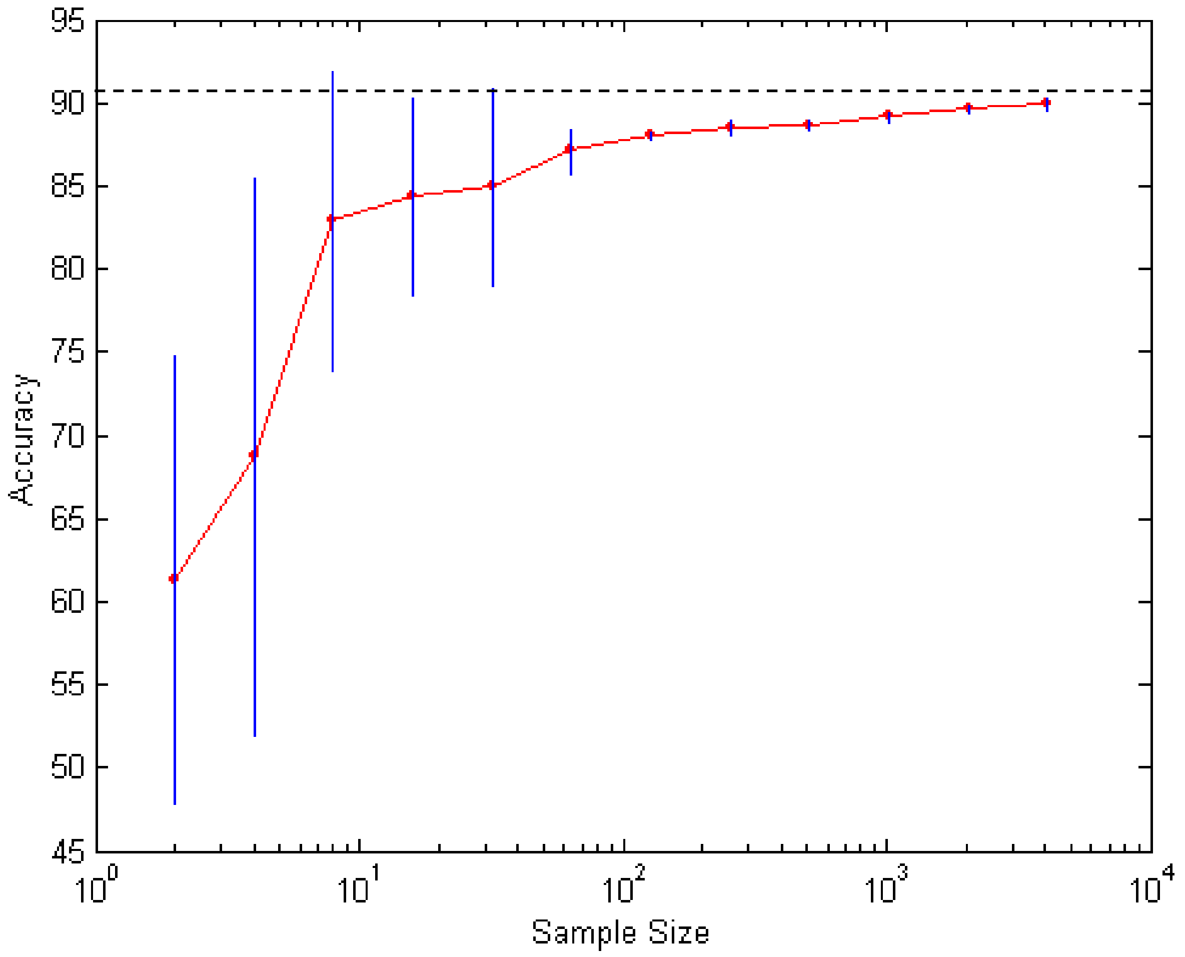


- Learning curve shows how accuracy changes with varying sample size
- Requires a sampling schedule for creating learning curve:
  - Arithmetic sampling (Langley, et al)
  - Geometric sampling (Provost et al)

Effect of small sample size:

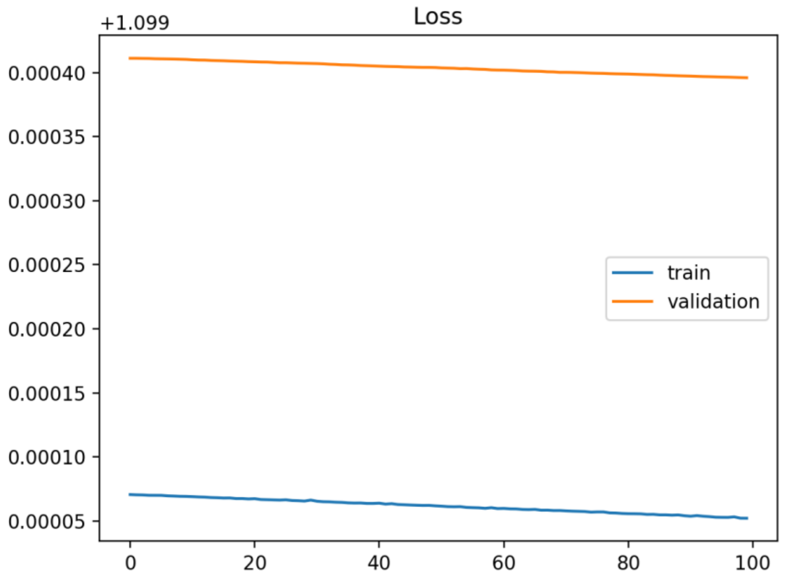
- Bias in the estimate
- Variance of estimate

Learning Curve

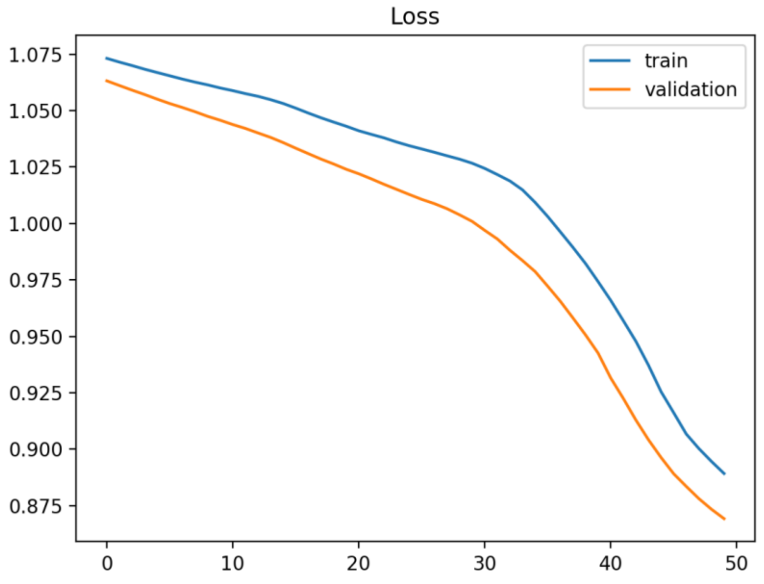


- Reviewing learning curves of models during training can be used to diagnose problems with learning, such as an **underfit** or **overfit** model, as well as whether the training and validation datasets are **suitably representative**.

Underfit Learning Curve



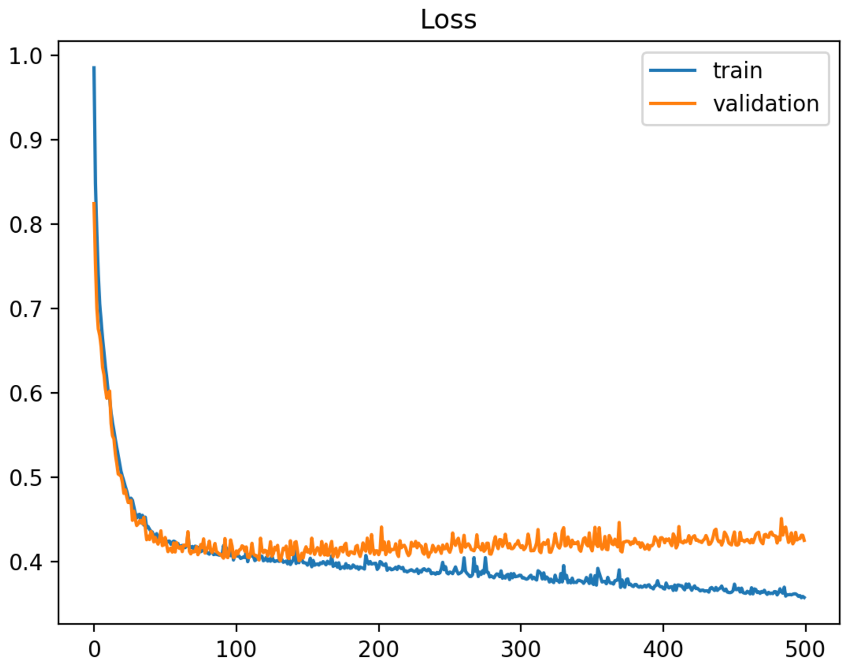
Example of Training Learning Curve Showing An Underfit Model That Does Not Have Sufficient Capacity



Example of Training Learning Curve Showing an Underfit Model That Requires Further Training

Source: machinelearningmastery.com/learning-curves-for-diagnosing-machine-learning-model-performance/

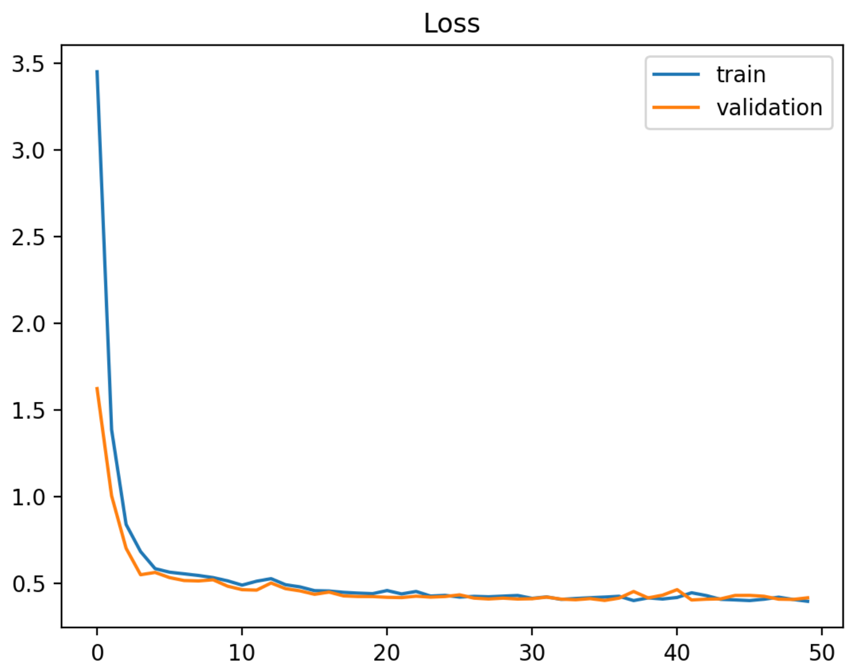
Overfit Learning Curve



Example of Train and Validation Learning Curves Showing an Overfit Model

Source: machinelearningmastery.com/learning-curves-for-diagnosing-machine-learning-model-performance/

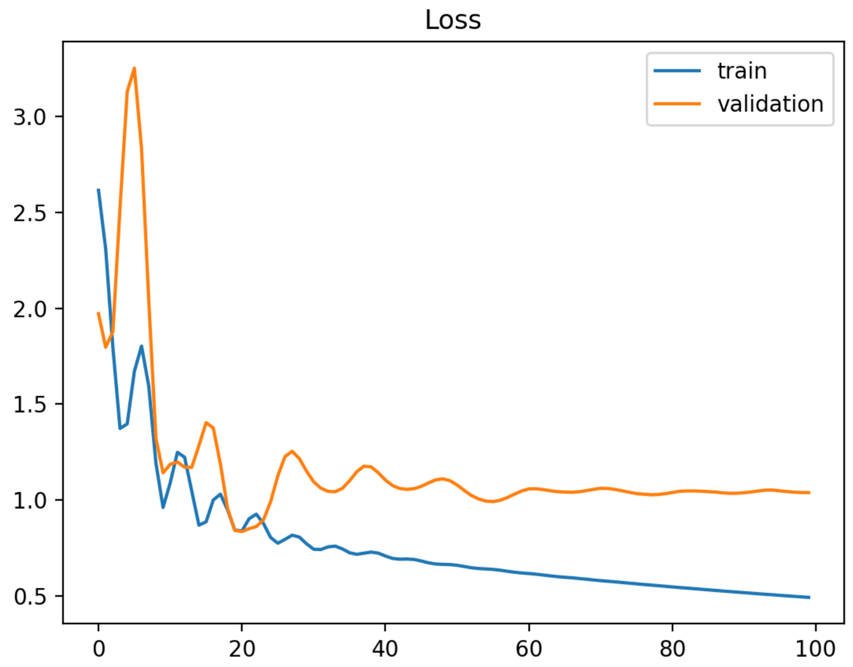
Good Fit Learning Curve



Example of Train and Validation Learning Curves Showing a Good Fit

Source: machinelearningmastery.com/learning-curves-for-diagnosing-machine-learning-model-performance/

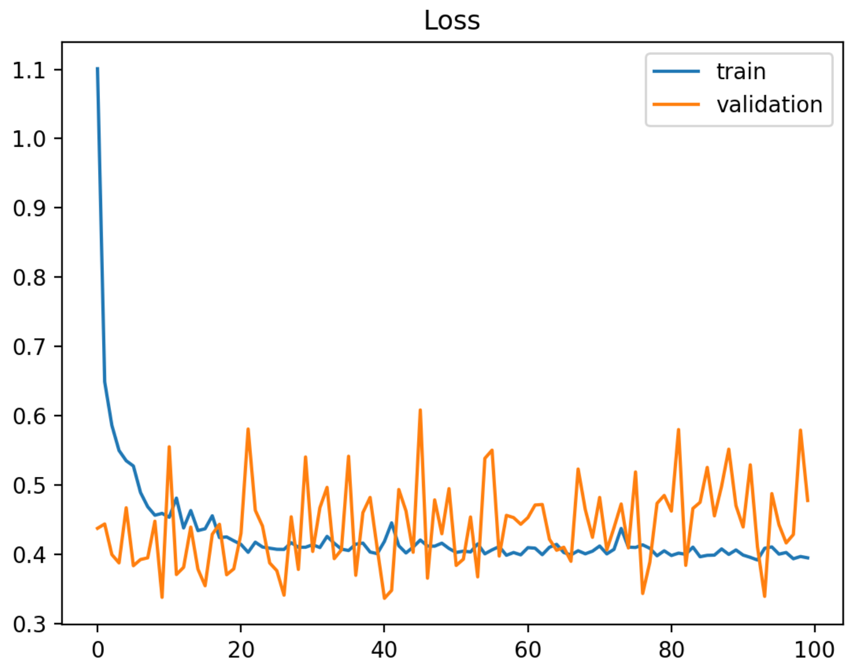
Unrepresentative Training Dataset



Example of Train and Validation Learning Curves Showing a Training Dataset That May Be too Small Relative to the Validation Dataset

Source: machinelearningmastery.com/learning-curves-for-diagnosing-machine-learning-model-performance/

Unrepresentative Validation Dataset



Example of Train and Validation Learning Curves Showing a Validation Dataset That May Be too Small Relative to the Training Dataset

Source: machinelearningmastery.com/learning-curves-for-diagnosing-machine-learning-model-performance/

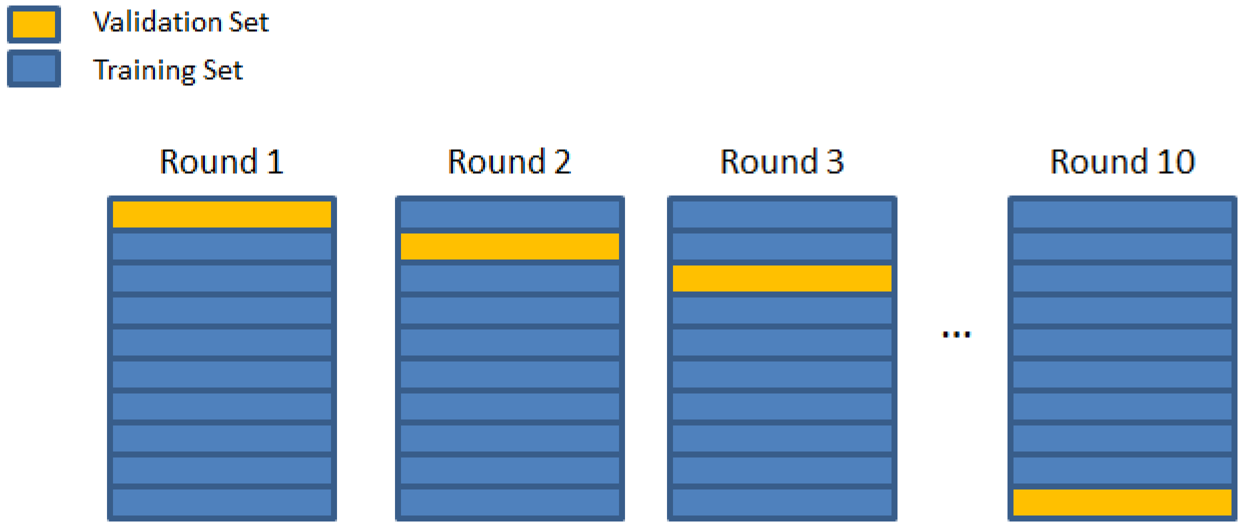
Methods of Estimation Re sample size

- Holdout
  - Reserve 2/3 for training and 1/3 for testing
- Random subsampling
  - Repeated holdout
  - Let acc_i be the accuracy of the model during the i^{th} iteration. The overall accuracy is given by $acc_{sub} = \sum_{i=1}^{k} {acc_i}/{k}$
- Cross validation
  - Partition data into k disjoint subsets
  - k-fold: train on k-1 partitions, test on the remaining one
  - Leave-one-out: k=n
- Stratified sampling
  - oversampling vs undersampling
- Bootstrap
  - Sampling with replacement

Cross Validation – Overall approach

- Partition data into a number of subsets
- Hold out a set at a time and train the model on remaining set
- Test model on hold out set

Repeat the process for each subset of the dataset



Source: https://towardsdatascience.com/cross-validation-explained-evaluating-estimator-performance-e51e5430ff85

Types of Cross Validation

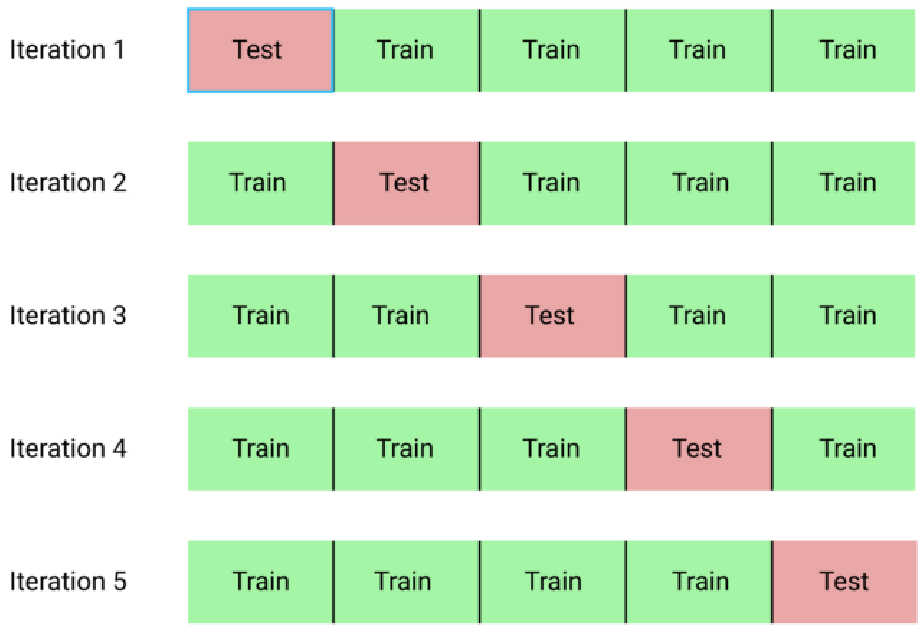
- K-Fold Cross Validation
- Stratified K-fold Cross Validation
- Leave One Out Cross Validation

Source: https://towardsdatascience.com/cross-validation-explained-evaluating-estimator-performance-e51e5430ff85

k-Fold Cross Validation

If **k=5:**

1. Take the group as a holdout or test data set
2. Take the remaining groups as a training data set
3. Fit a model on the training set and evaluate it on the test set
4. Retain the evaluation score and discard the model

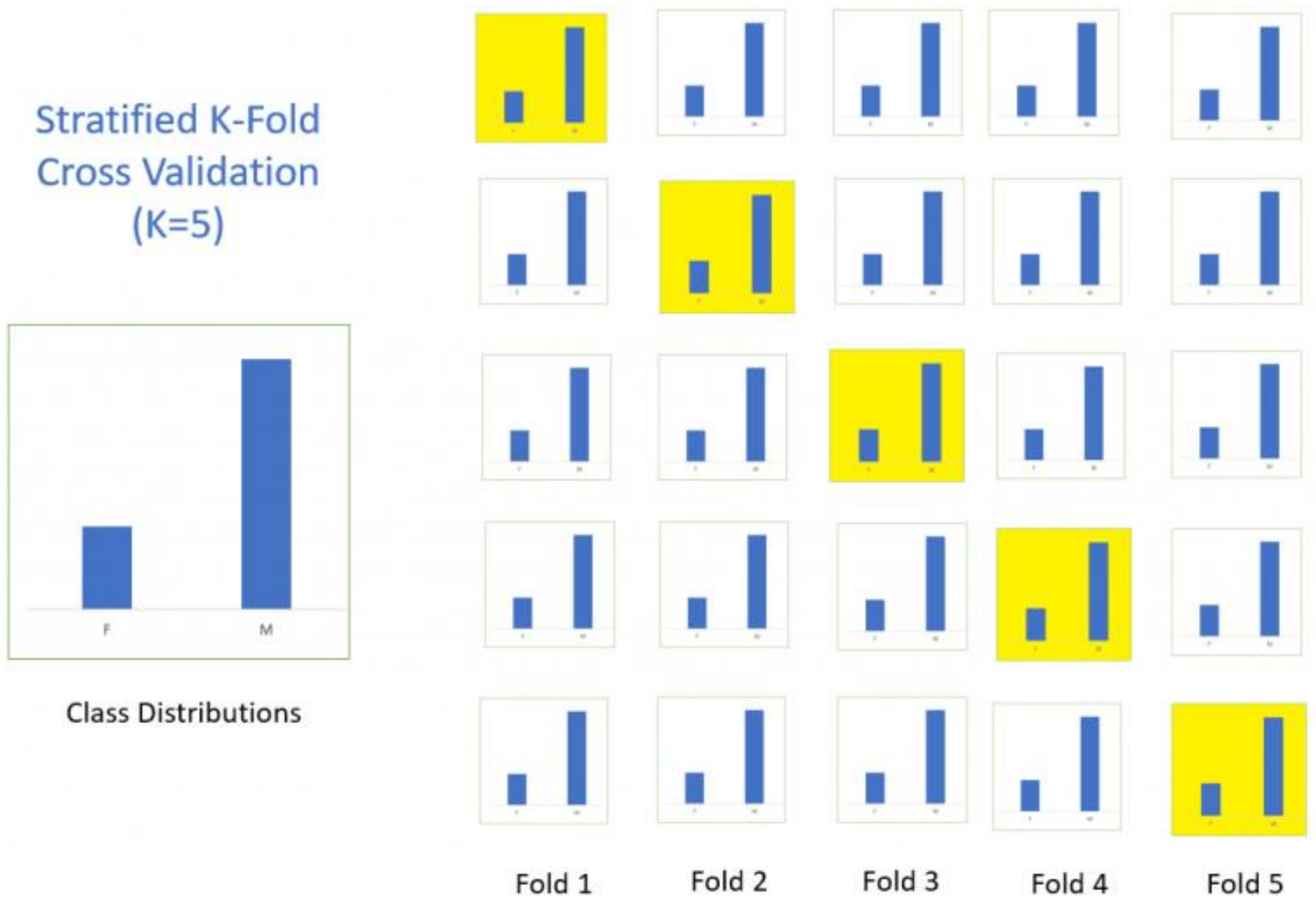


Source: https://towardsdatascience.com/cross-validation-explained-evaluating-estimator-performance-e51e5430ff85

How to decide the value of *k*?

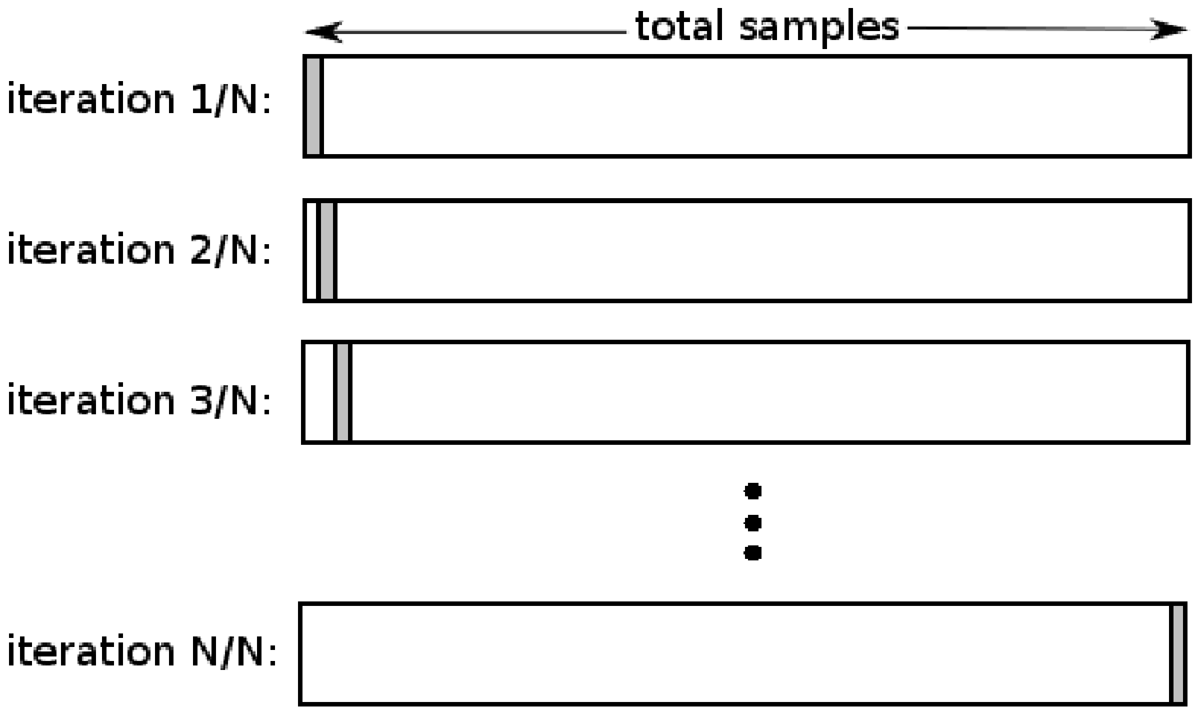
- For small datasets (e.g., fewer than 100 samples), a higher value of *k* (e.g., 10) may be more appropriate
- For larger datasets, a lower value of k (e.g., 5) may be sufficient
- For imbalanced datasets $\longrightarrow$ ensure that each fold contains a proportional representation of each class
- For highly variable datasets $\longrightarrow$ choose a higher value of *k*

Stratified k-Fold Cross Validation



Source: https://towardsdatascience.com/cross-validation-explained-evaluating-estimator-performance-e51e5430ff85

Leave One Out Cross Validation (LOOCV)



- if there are *n* data points in the original sample then, ***n-1*** **samples are used to train the model and *p* points are used as the validation set**

- This is repeated for all combinations in which the original sample can be separated this way, and then the error is averaged for all trials, to give overall effectiveness.

Source: https://towardsdatascience.com/cross-validation-explained-evaluating-estimator-performance-e51e5430ff85

Let’s see how this works…

(Additional Resources learnonline – Code Sample)

Model Evaluation

- Metrics for Performance Evaluation
  - How to evaluate the performance of a model?
- Methods for Performance Evaluation
  - How to obtain reliable estimates?
- Methods for Model Comparison
  - How to compare the relative performance among competing models?

ROC (Receiver Operating Characteristic)

- Developed in 1950s for signal detection theory to analyze noisy signals
  - Characterize the trade-off between positive hits and false alarms
- ROC curve plots FPR (noise rate, on the x-axis) against TPR (recall, on the y-axis)
- Performance of each classifier represented as a point on the ROC curve
  - changing the threshold of algorithm, sample distribution or cost matrix changes the location of the point

ROC (Receiver Operating Characteristic)

TPR (True Positive Rate)

$$\text{TPR /Recall / Sensitivity} = \frac{TP}{TP + FN}$$

Specificity

$$\text{Specificity} = \frac{TN}{TN + FP}$$

FPR

$$\text{FPR} = 1 - \text{Specificity}$$

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

ROC Curve

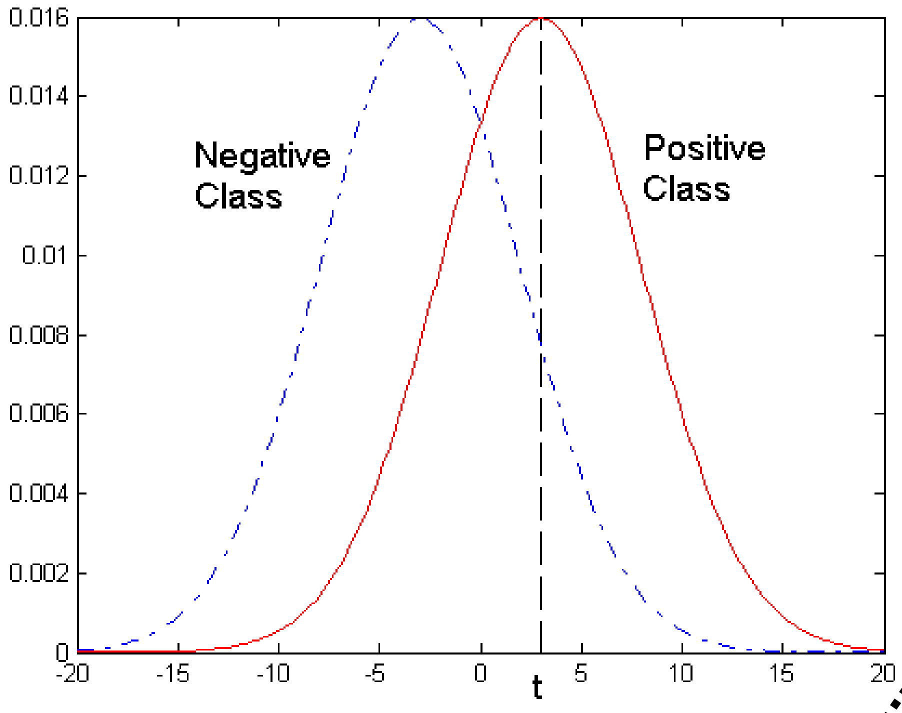
Is a model better than random guessing?
cumulative recall vs cumulative noise

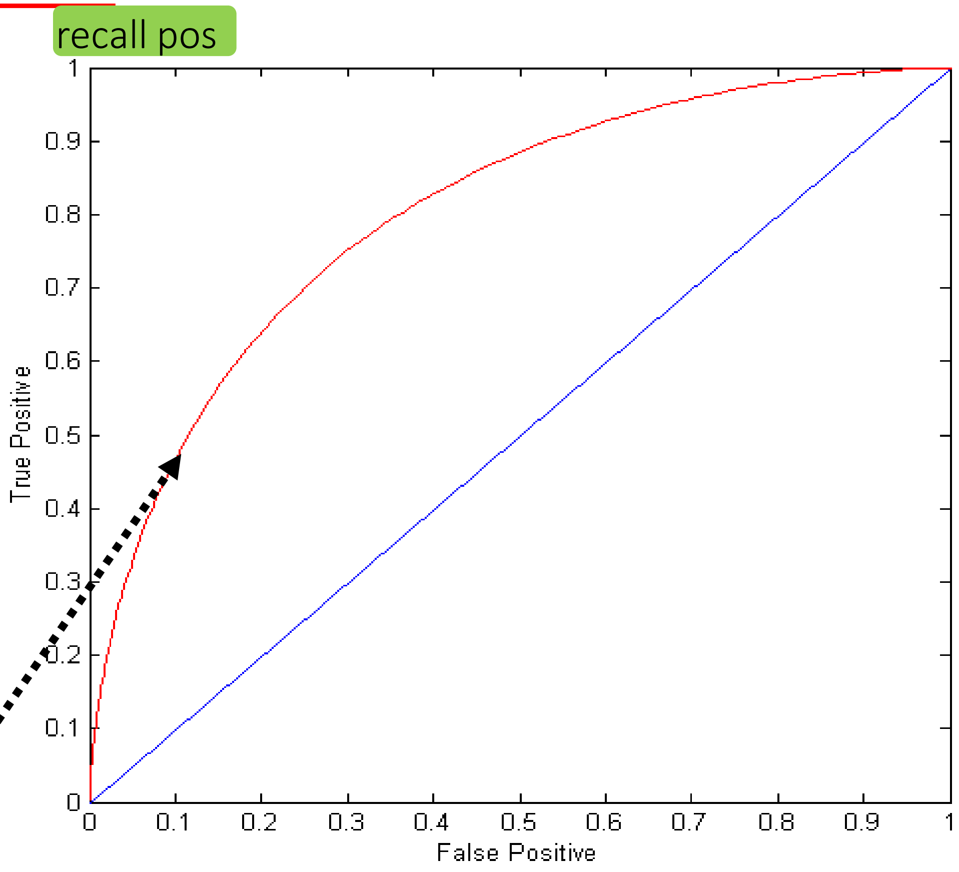
- 1-dimensional data set containing 2 classes (positive and negative)

- any points located at **x > t are classified as positive**

TPR=TP/(TP+FN)

FPR=FP/(FP+TN)





Assume at threshold t:

TP=0.5, FN=0.5, so TPR=0.5/(.5+.5) = .5

FP=0.12, TN=0.88 so FPR=.12/(.12+.88)=.12

move t to left: recalling more positive will also make more negative positive (FP)

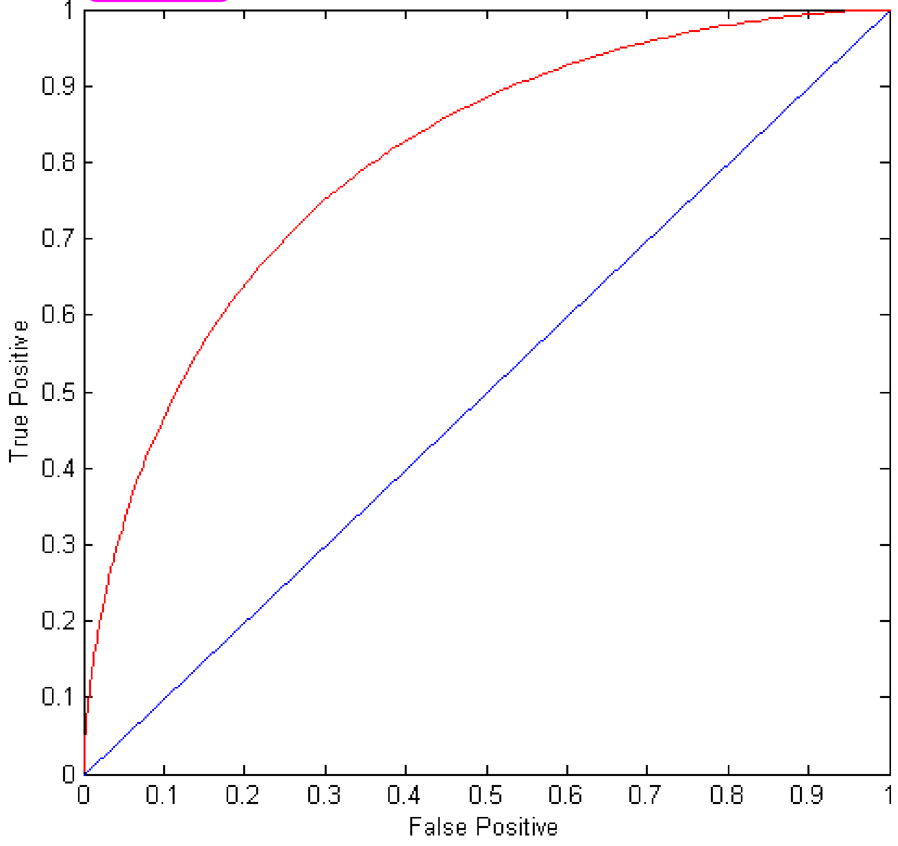
ROC Curve

(FP,TP):

- (0,0): declare everything to be negative class
- (1,1): declare everything to be positive class
- (0,1): ideal: the boundary t makes prediction of all positives correct without introducing any noise.

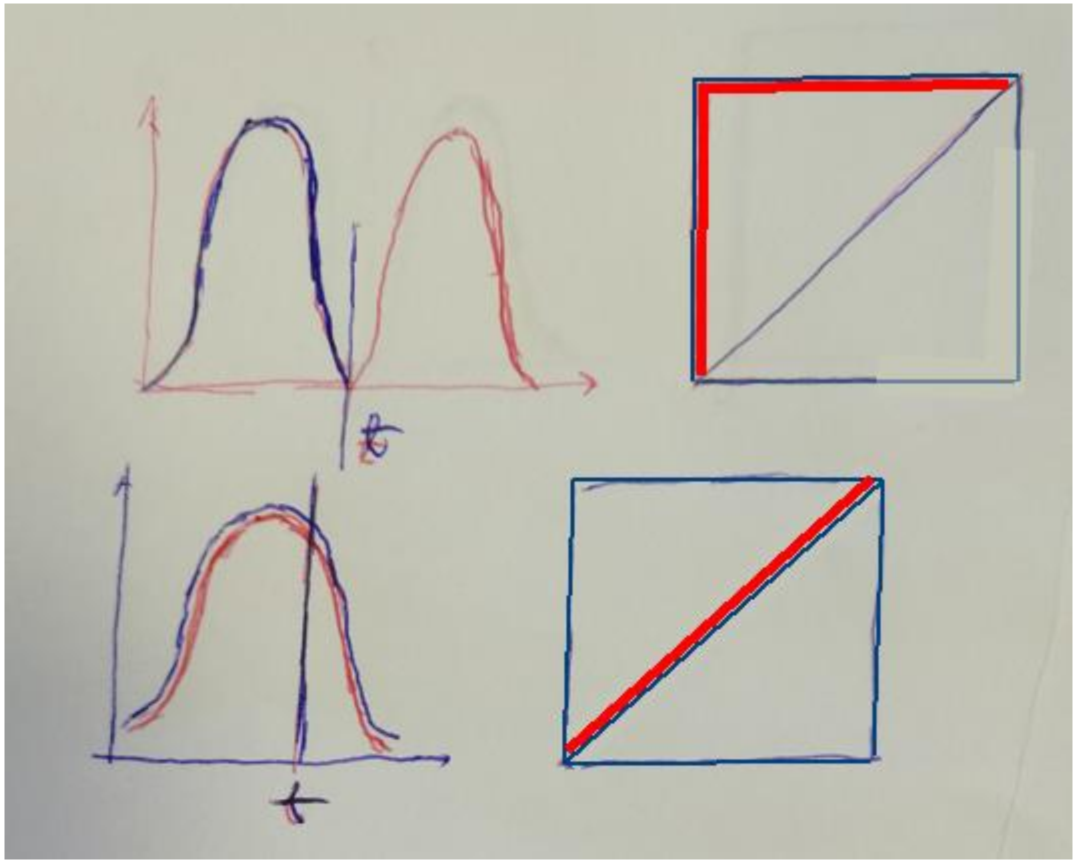
- Diagonal line:
  - Random guessing
  - Below diagonal line:
    - prediction is opposite of the true class

recall

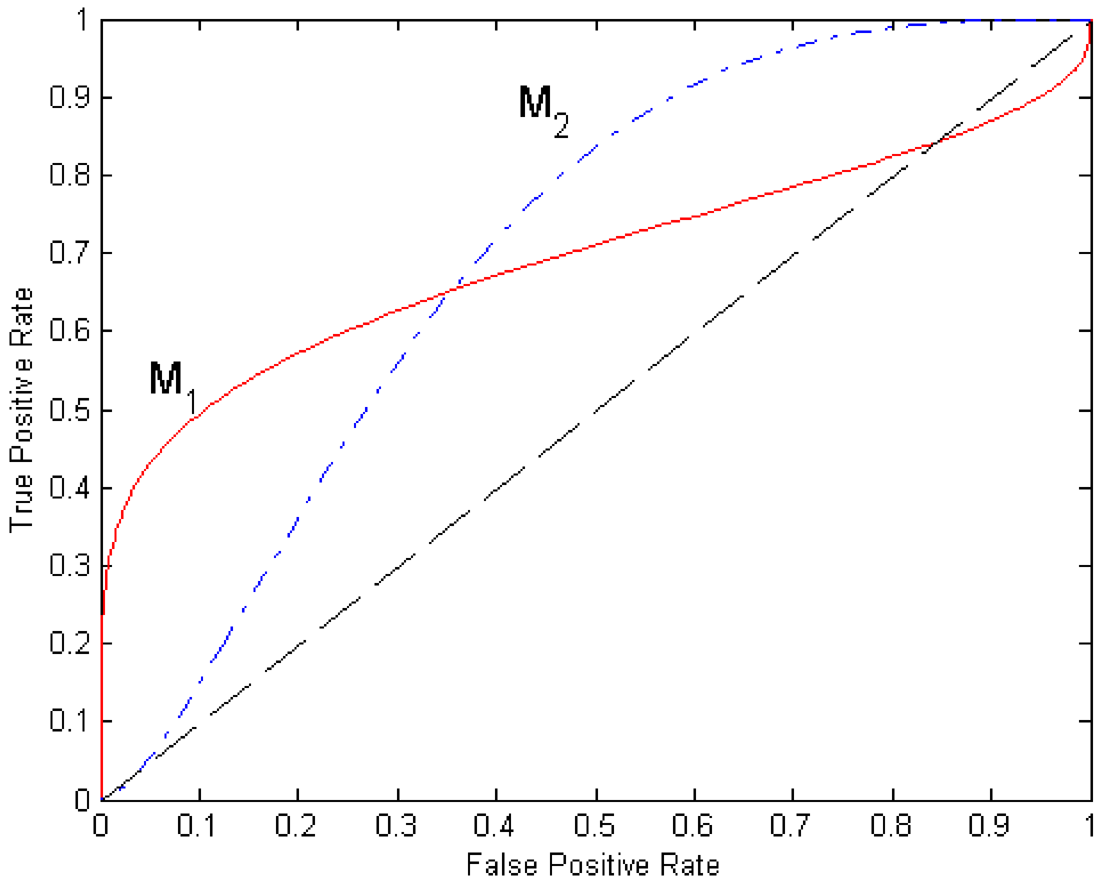


Extreme ROC curve

Top: we can recall all positive positive without making FP. **Best case**
Bottom: we cannot make any positive positive without making FP **Worst case**



Using ROC for Model Comparison



- No model consistently outperform the other
  - M_1 is better for small FPR
  - M_2 is better for large FPR
- Area Under the ROC curve
  - Ideal:
    - Area = 1
  - Random guess:
    - Area = 0.5

Test of Significance of Two Models

- Given two models:
  - Model M1: accuracy = 85%, tested on 30 instances
  - Model M2: accuracy = 75%, tested on 5000 instances

- Can we say M1 is better than M2?
  - How much confidence can we place on accuracy of M1 and M2?
  - Can the difference in performance measure be explained as a result of random fluctuations in the test set?

Confidence Interval for Accuracy

- Prediction can be regarded as a Bernoulli trial
  - A Bernoulli trial has 2 possible outcomes
  - Possible outcomes for prediction: correct or wrong
  - Collection of Bernoulli trials has a Binomial distribution:
    - $x \sim \text{Bin}(N, p)$ x: number of correct predictions
    - e.g: Toss a fair coin 50 times, how many heads would turn up?
      Expected number of heads = $N \times p = 50 \times 0.5 = 25$

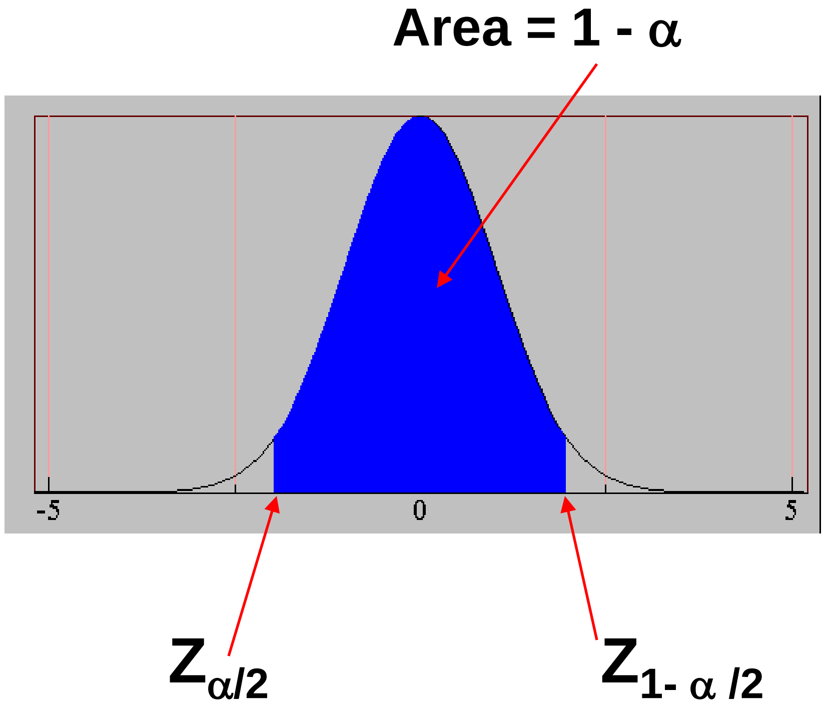
    Expected variance would be $Np \times (1\text{-}p) = 50 \text{ x } 0.5 \text{ x } 0.5 = 12.5$

- Given x (# of correct predictions) or equivalently, acc=x/N, and N (# of test instances),
  Can we predict p (true accuracy of model)?

Confidence Interval for Accuracy

- For large test sets (N > 30),
  - acc has a normal distribution with mean p and variance p(1-p)/N

$$P\left(Z_{\alpha/2} < \frac{acc - p}{\sqrt{p(1-p)/N}} < Z_{1-\alpha/2}\right) = 1 - \alpha$$



- Confidence Interval for p:

$$p = \frac{2 \times N \times acc + Z^2_{\alpha/2} \pm \sqrt{Z^2_{\alpha/2} + 4 \times N \times acc - 4 \times N \times acc^2}}{2(N + Z^2_{\alpha/2})}$$

Confidence Interval for Accuracy

- Consider a model that produces an accuracy of 80% when evaluated on 100 test instances:
  - N=100, acc = 0.8
  - Let 1-α = 0.95 (95% confidence)
  - From probability table, $Z_{\alpha/2}$=1.96

1-α	Z
0.99	2.58
0.98	2.33
0.95	1.96
0.90	1.65

https://people.richland.edu/james/lecture/m170/ch08-int.html

N	50	100	500	1000	5000
p(lower)	0.670	0.711	0.763	0.774	0.789
p(upper)	0.888	0.866	0.833	0.824	0.811

The acc is in [.711, .866] with 95% of confidence.

Statistical Significance

- Consider the problem we described earlier:
  - M_A has an error rate of $e_1 = 0.15$, when applied to $N_1 = 30$ test instances
  - M_B has an error rate of $e_2 = 0.25$, when applied to $N_2 = 5000$ test instances
  - The observed **difference** in their error rates is:

$$d = |e_1 - e_2| = 0.1$$

  - The **estimated variance** of the observed difference in error rates is:

$$\widehat{\sigma_d^2} = \frac{e_1(1-e_1)}{n_1} + \frac{e_2(1-e_2)}{n_2} = \frac{0.15\ (1-0.15)}{30} + \frac{0.25\ (1-0.25)}{5000}$$
$$= 0.0043$$

Statistical Significance

$$\widehat{\sigma_d^2} = 0.0043$$

- The **standard deviation** from here would be

$$\widehat{\sigma_d} = sqrt(\widehat{\sigma_d^2}) = 0.0655$$

- The **confidence interval** for the true difference is given by the following equation

$$d_t = d \pm z_{\alpha/2}\widehat{\sigma_d}$$

$$= 0.1 \pm 1.96 \times 0.0655 = 0.1 \pm 0.128$$

[-0.028,0.228]