Department of Mathematics
University of Calabria

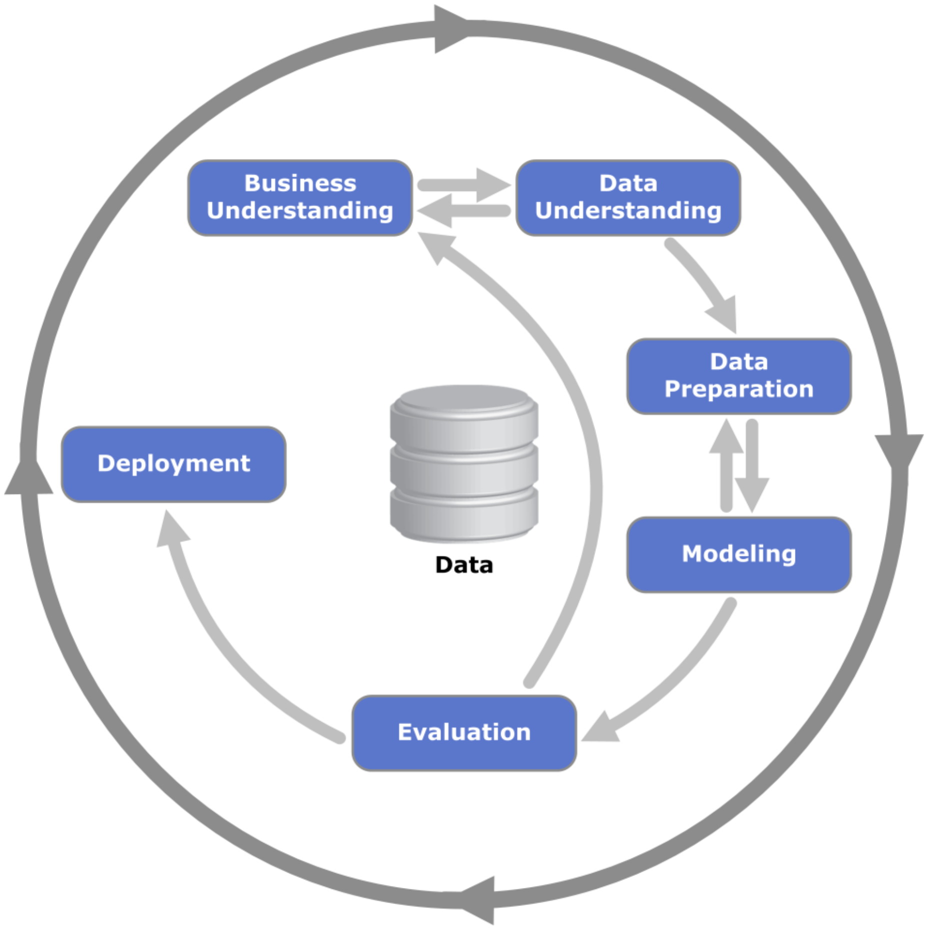
Business Intelligence and Analytics

(Data Mining)

Evaluation

Ph.D. Ettore Ritacco

CRISP-DM Methodology

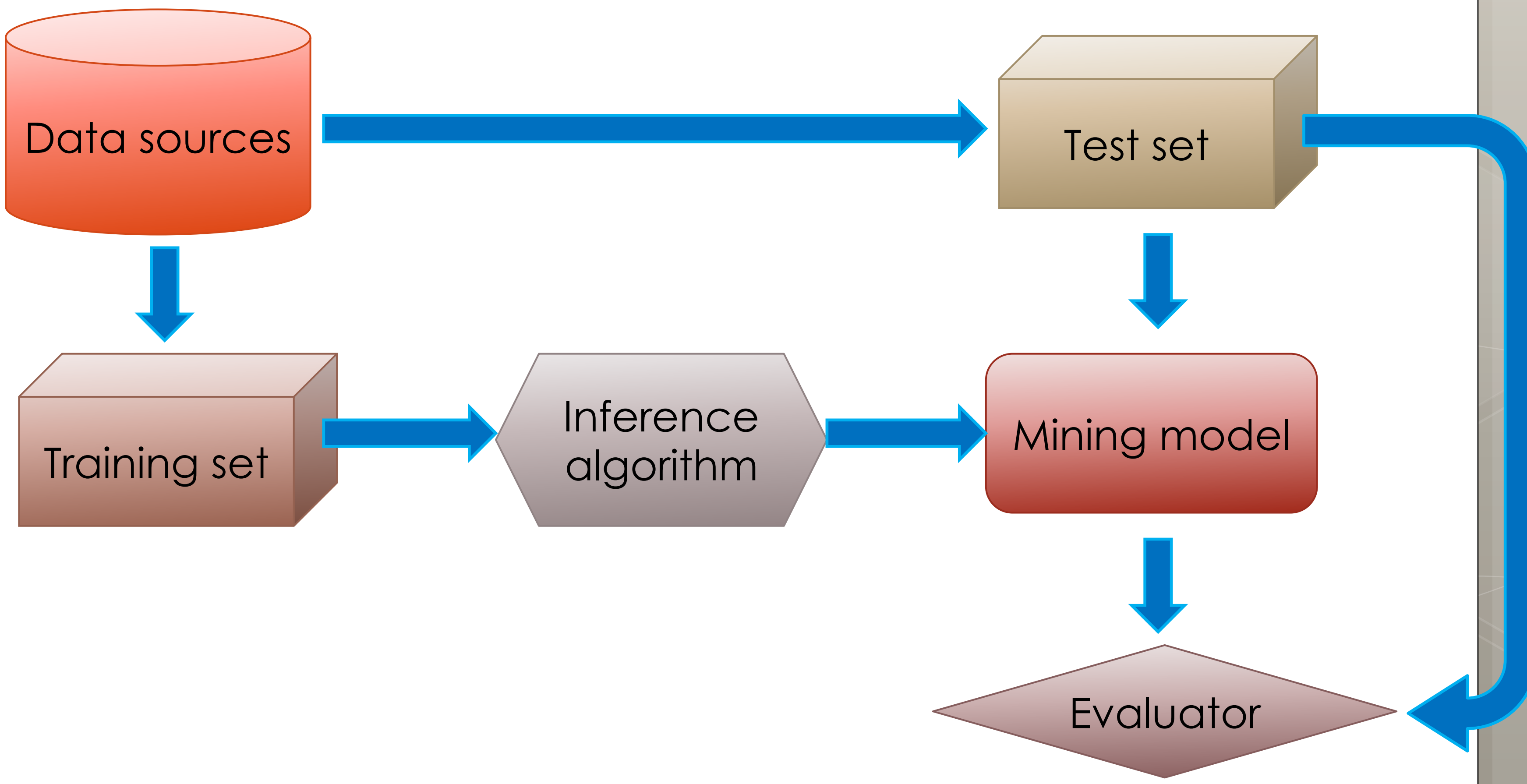


Department of Mathematics
University of Calabria

How to evaluate a model?

- Select a training set
- Build a mining model
- **Choose a quality measure**
- **Select a test set**
- **Apply the model on the test set**
- **Compute the value of the quality measure**

A simple evaluation schema

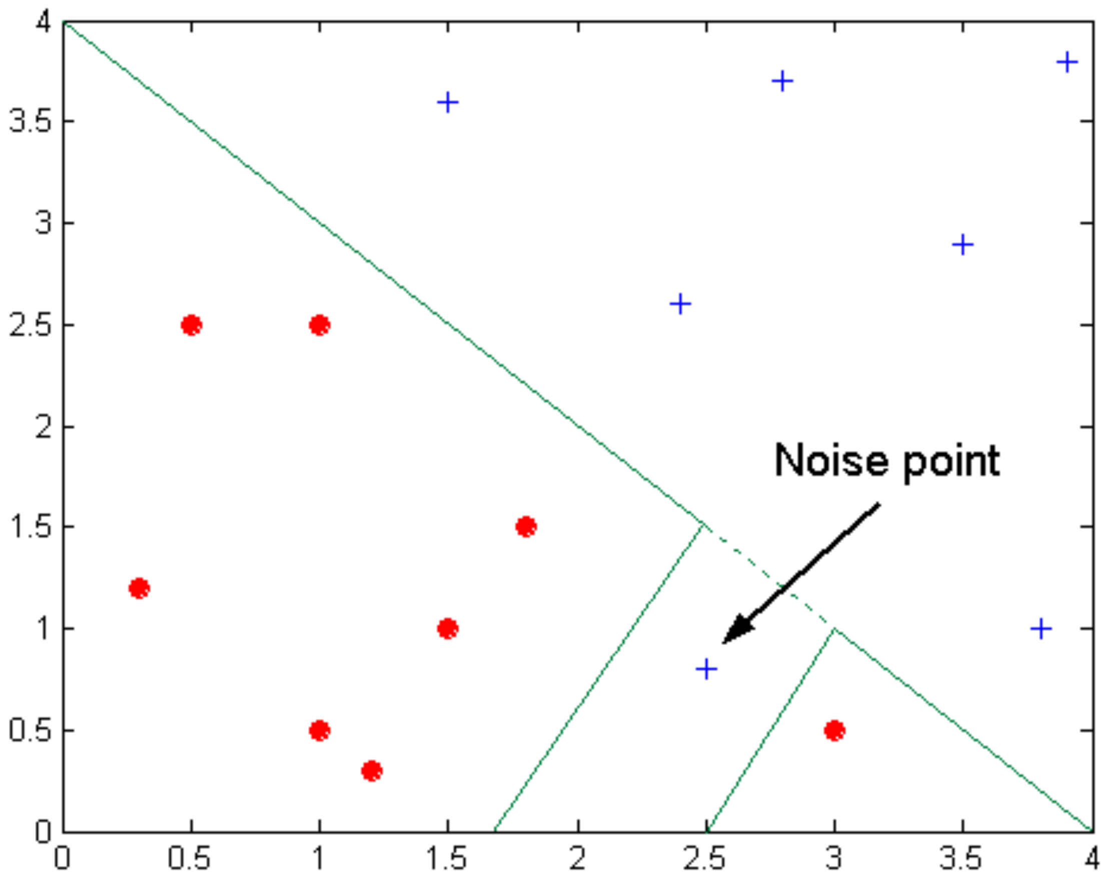


The fitting problem

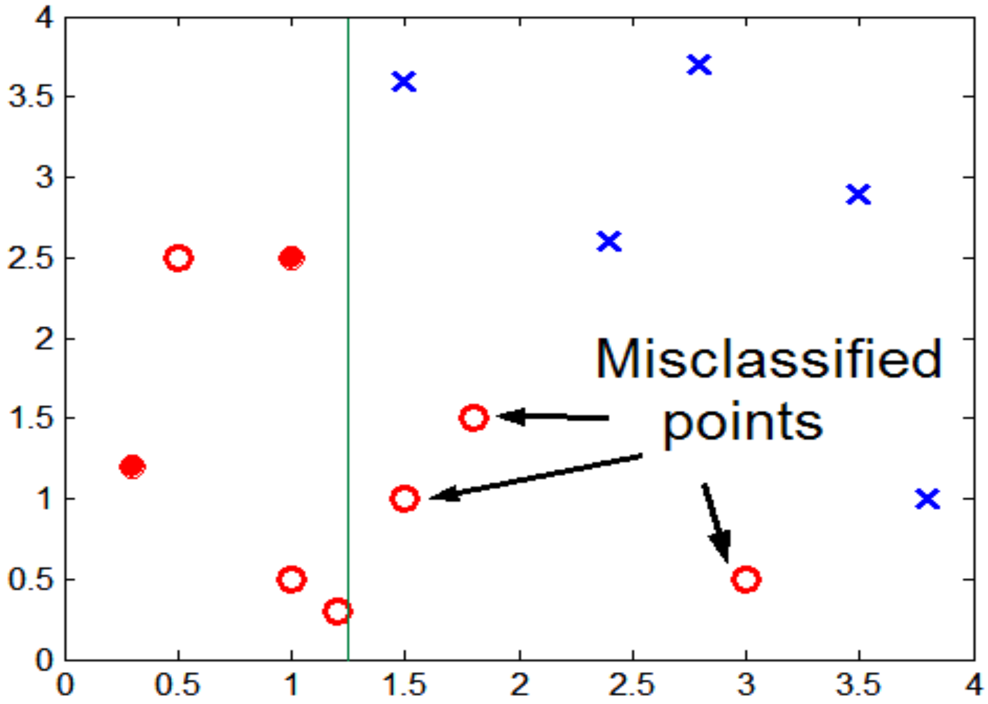
- Beyond the data analysis issues, there are challenges even in the modeling and evaluate phases in the CRISP-DM Methodology

- Namely
  - Underfitting
    - The model is too simple: the evaluation will be poor on both the training and the evaluation set
  - Overfitting
    - The model is too complex, fitting as close as it can the training data, the evaluation will be good on the training set, but poor on the evaluation set

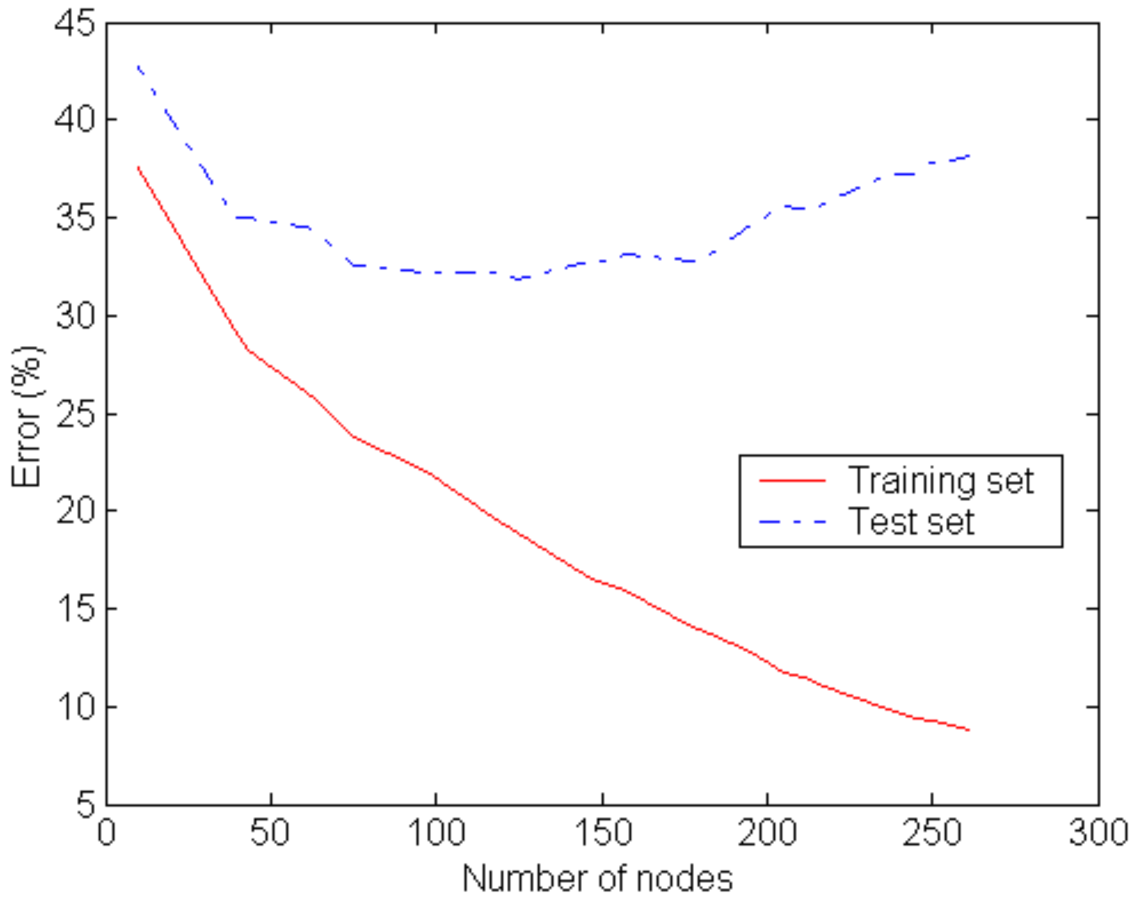
Overfitting (due to noise)



Overfitting (due to lack of information)



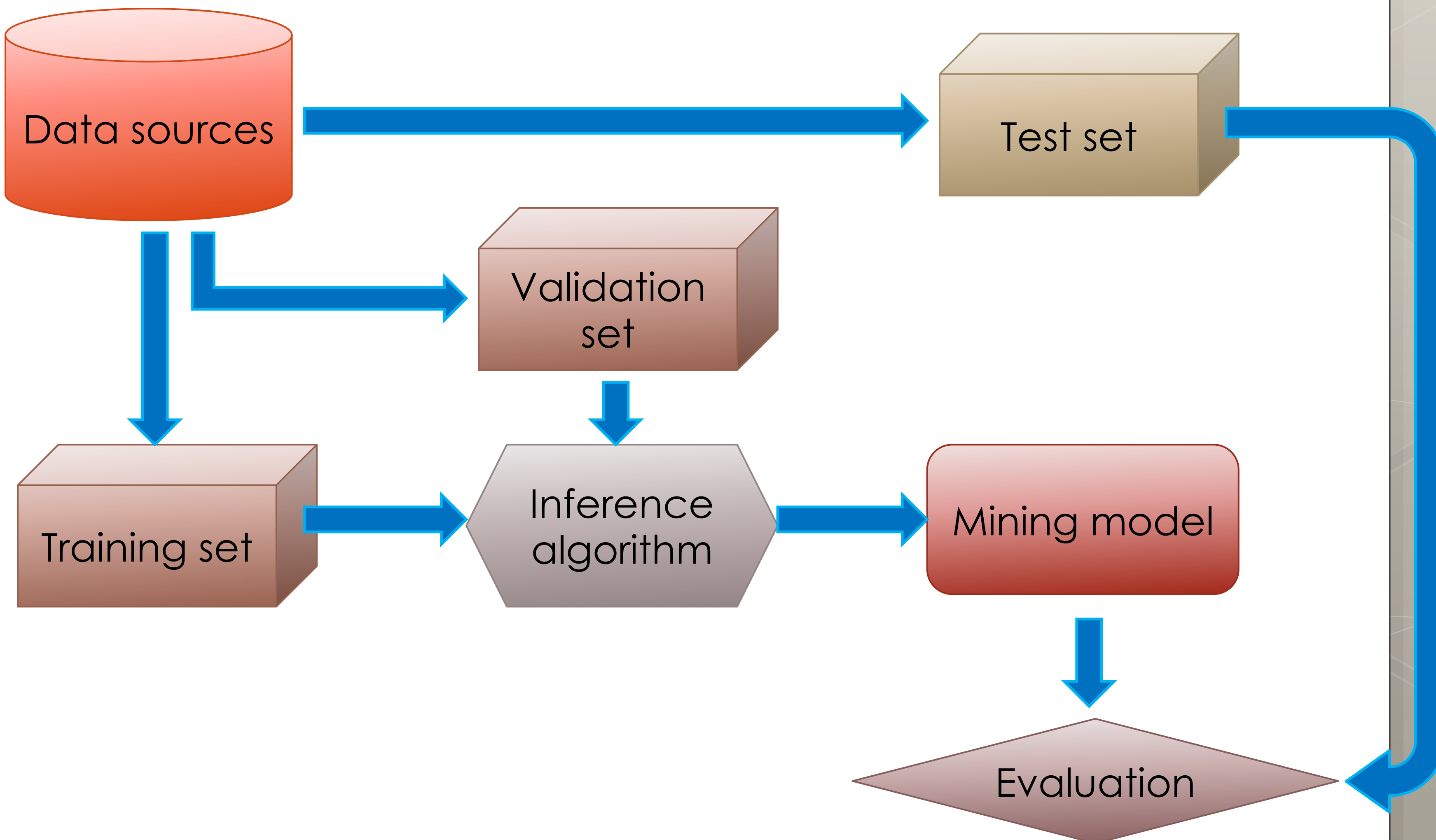
Overfitting



How to mitigate the overfittig?

- Prevention
  - A good data preparation
- Avoiding
  - Feed the building phase with further data for improving the model's generality (e.g. online pruning)
- Recovery
  - Manipulate the model after its creation (e.g. post pruning)

How to mitigate the overfittig?



How to evaluate a model?

- Is a model that achieves 70% of global accuracy a “good” model?

How to evaluate a model?

- Is a model that achieves 70% of global accuracy a "good" model?
  - It depends…

How to evaluate a model?

- Is a model that achieves 70% of global accuracy a “good” model?
  - It depends…
- Is a model that achieves 95% of global accuracy a “good” model?

Department of Mathematics
University of Calabria

How to evaluate a model?

- Is a model that achieves 70% of global accuracy a "good" model?
  - It depends...
- Is a model that achieves 95% of global accuracy a "good" model?
  - It depends...

How to evaluate a model?

- We can perform only comparative evaluations.
- A "*null hypothesis*" (in other words, a *baseline*) is needed.
- We can only say, given a statistic, if a model is better then another one, in terms of the chosen statistic.

True and estimated error

- The "true" error of a hypothesis *h* in the domain *D*

$$e_{true}(h) = \Pr_{x \in D}\big(c(x) \neq h(x)\big)$$

- The estimated (observed) error on a data set *S*

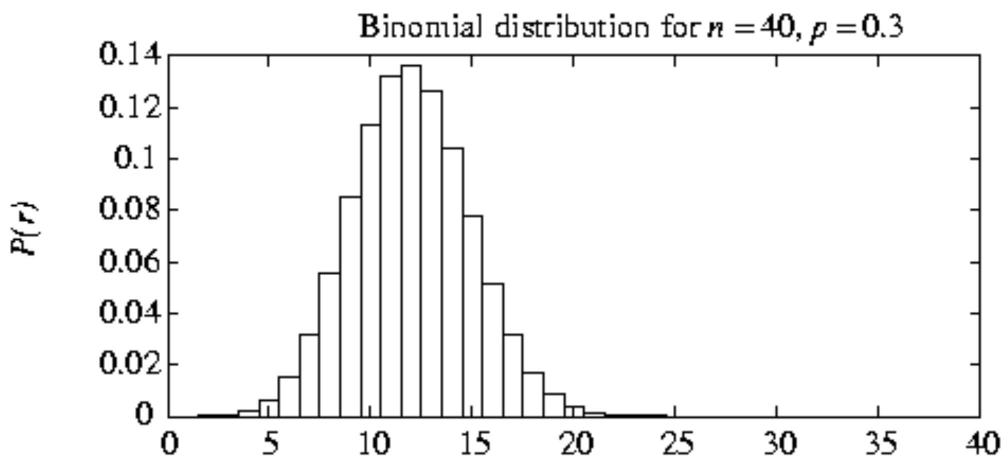
$$e_{estimation}(h) = \frac{1}{|S|}\sum_{x \in S} e(x)$$

- Where:

$$e(x) = \begin{cases} 1 & if\ c(x) \neq h(x) \\ 0 & otherwise \end{cases}$$

True Error

- The probability of (exactly) *r* misclassifications in *n* evaluations is governed by a binomial distribution:



$$\Pr(r) = \binom{n}{r} e_{true}(h)^r \left(1 - e_{true}(h)\right)^{n-r}$$

True Error – Binomial Distribution

- Probability Mass Distribution

$$\Pr(r) = \frac{n!}{r!\,(n-r)!} e_{true}(h)^r \left(1 - e_{true}(h)\right)^{n-r}$$

- Cumulative Distribution Function

$$\Pr(a \le r \le b) = \sum_{r=a}^{b} \frac{n!}{r!\,(n-r)!} e_{true}(h)^r \left(1 - e_{true}(h)\right)^{n-r}$$

- Expected Value

$$E[R] = n \cdot e_{true}(h)$$

- Variance & Standard Deviation

$$Var[R] = n \cdot e_{true}(h) \cdot [1 - e_{true}(h)] \qquad sd[R] = \sqrt{Var[R]}$$

Estimated Error

- Given a set of data S

$$e_{estimation}(h) = \frac{1}{|S|} \sum_{x \in S} e(x)$$

- Where $e(x)$ are independent and identically distributed (i.i.d.) Bernoullian random variables:

$$e(x) = \begin{cases} 1 & if\ c(x) \neq h(x) \\ 0 & otherwise \end{cases} \qquad e(x) \sim Bernoulli\big(e_{true}(h)\big)$$

Bernoulli Distribution

- Probability Mass Distribution

$$\Pr(e(x); e_{true}(h)) = e_{true}(h)^{e(x)}\left(1 - e_{true}(h)\right)^{1-e(x)}$$

- Expected Value

$$E[e(X)] = e_{true}$$

- Variance & Standard Deviation

$$Var[e(X)] = e_{true}(h) \cdot [1 - e_{true}(h)] \qquad sd[e(X)] = \sqrt{Var[e(X)]}$$

Estimated Error Distribution

- From the probability theory, the sum of i.i.d. Bernoulli variables is governed by a binomial distribution
  - Proof by induction: http://www.statlect.com/uddbin1.htm

- $e_{estimation}(h)$ is also a binomial distribution

$$e_{estimation}(h) = \frac{1}{|S|} \sum_{x \in S} e(x)$$

$$e_{estimation}(h) \sim Binomial\big(|S|, e_{true}(h)\big)$$

Department of Mathematics
University of Calabria

Estimated Error Expected Value & Variance

- Expected Value:

$$E[e_{estimation}(h)] = E\left[\frac{1}{|S|}\sum_{x\in S} e(x)\right] = \frac{1}{|S|}\sum_{x\in S} E[e(x)]$$

$$= E[e(x)] = e_{true}(H)$$

- Variance:

$$Var[e_{estimation}(h)] = Var\left[\frac{1}{|S|}\sum_{x\in S} e(x)\right] = \frac{1}{|S|^2}\sum_{x\in S} Var[e(x)]$$

$$= \frac{1}{|S|} Var[e(x)] = \frac{1}{|S|} e_{true}(h)\cdot[1 - e_{true}(h)]$$

Summary 1/2

- There exists a link between the true error and the estimated error, if the data set *S* is representative of its domain

- **The strong law of large numbers**

$$\Pr\left(\lim_{|S|\to\infty} \frac{1}{|S|} \sum_{x\in S} e(x) = e_{true}(h)\right) = 1$$

$$\lim_{|S|\to\infty} e_{estimation}(h) = e_{true}(h) \quad almost\ surely$$

Summary 2/2

- The estimated error is a binomial distribution, if | S | is great "enough":

$$E[e_{estimation}(h)] = e_{true}(h) \approx e_{estimation}(h)$$

$$Var[e_{estimation}(h)] = \frac{e_{true}(h) \cdot [1 - e_{true}(h)]}{|S|} \approx \frac{e_{estimation}(h) \cdot [1 - e_{estimation}(h)]}{|S|}$$

$$sd[e_{estimation}(h)] = \sqrt{Var[e_{estimation}(h)]} \approx \sqrt{\frac{e_{estimation}(h) \cdot [1 - e_{estimation}(h)]}{|S|}}$$

Binomial – Normal Approximation

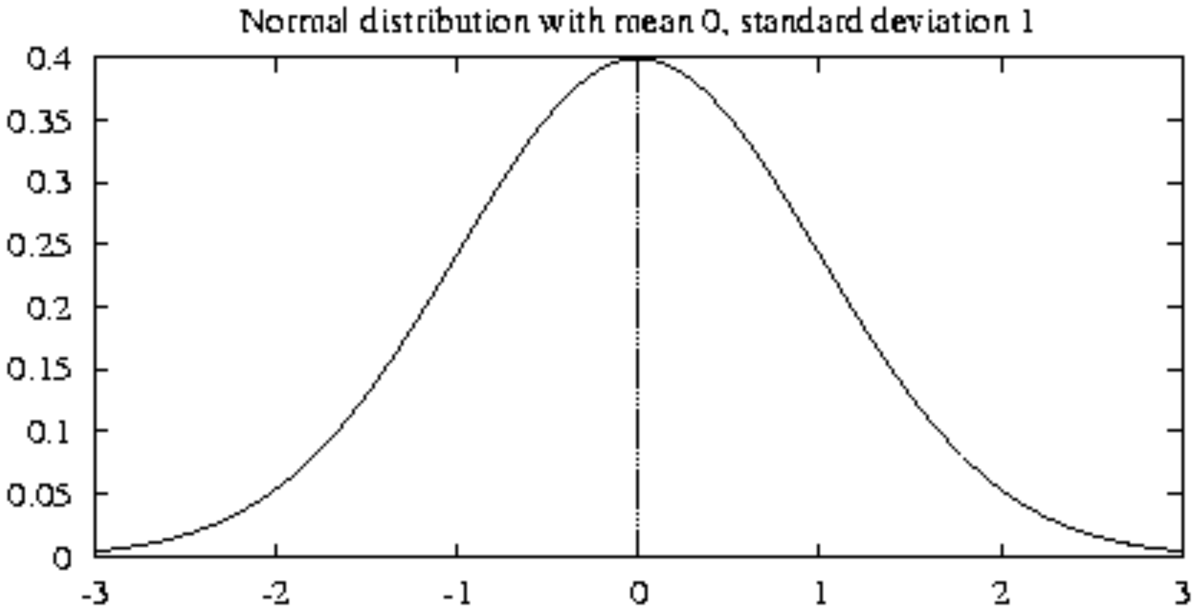
- If $|S|$ is sufficient great (typically $|S| > 30$) the binomial distribution can be approximated by a normal distribution

  - Central limit theorem

    - "states that the distribution of the sum (or average) of a large number of independent, identically distributed variables will be approximately normal, regardless of the underlying distribution."

Normal Distribution

- Normal distribution



Normal Distribution

- Normal distribution
  - Density $$f(x) = \frac{1}{\sqrt{2\pi\sigma}} \exp\left\{-\frac{(x-\mu)^2}{2\sigma^2}\right\}$$
  - Cumulative $$\Pr(a \leq X \leq b) = \int_a^b f(x)dx$$
  - Expected Value $$E[X] = \mu$$
  - Variance $$Var[X] = \sigma$$

Department of Mathematics
University of Calabria

Mean and Variance Approximation

- Due to the binomial – normal approximation

$$\mu \approx e_{estimation}(h)$$

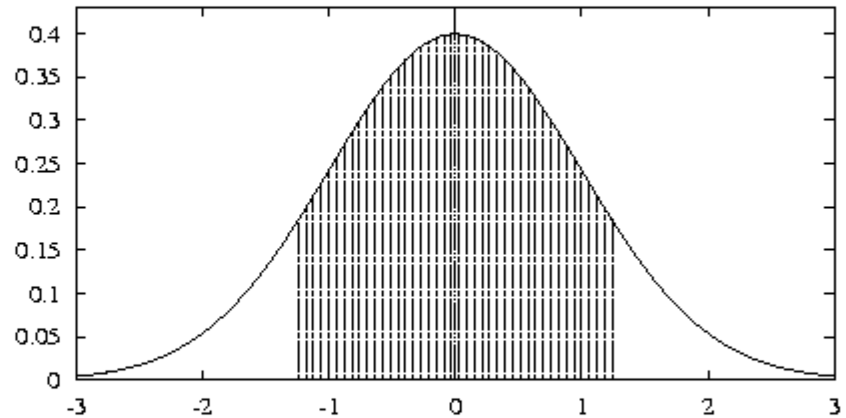
$$\sigma^2 \approx \frac{e_{estimation}(h) \cdot [1 - e_{estimation}(h)]}{|S|}$$

$$\sigma \approx \sqrt{\frac{e_{estimation}(h) \cdot [1 - e_{estimation}(h)]}{|S|}}$$

Why are we interested in the Normal distribution?

- Confidence Intervals
  - Given a probability α, we are interested in finding an interval [a, b] such that

$$\Pr(a \leq X \leq b) = \gamma$$



  - In the normal case

$$\Pr(\mu - z_n\sigma \leq X \leq \mu + z_n\sigma) = \gamma$$

γ	50%	68%	80%	90%	**95%**	98%	99%
z_N	0.67	1.00	1.28	1.64	**1.96**	2.33	2.58

Why are we interested in the Normal distribution?

- This means that the true error is in the interval

$$e_{true}(h) \in \left\{ e_{estimation}(h) \pm z_n \sqrt{\frac{e_{estimation}(h) \cdot [1 - e_{estimation}(h)]}{|S|}} \right\}$$

- With probability γ

γ	50%	68%	80%	90%	**95%**	98%	99%
z_N	0.67	1.00	1.28	1.64	**1.96**	2.33	2.58

How to compare models?

- Consider two hypothesis *h* and *j*…
- … and the random variable

$$d = e(h) - e(j)$$

- It's governed by a binomial distribution
- Choose z_n and consequently γ

How to compare models?

- Three cases: $$d = e(h) - e(j)$$
  - Zero is in the confidence interval of *d*
    - There is no statistical difference between *h* and *j*, with significance γ
  - The confidence interval of *d* is under Zero
    - *e(h)* is statistically lower than *e(j)*, with significance γ
  - The confidence interval of *d* is above Zero
    - *e(h)* is statistically higher than *e(j)*, with significance γ

$$\Pr(\mu - z_n\sigma \leq X \leq \mu + z_n\sigma) = \gamma$$

How to compare models?

- Where:

$$\mu = |e_{estimation}(h) - e_{estimation}(j)|$$

- And, since the hypothesis are independent:

$$\sigma^2 = Var[e_{estimation}(h)] + Var[e_{estimation}(j)]$$

Evaluation Example

- Let
  - $e(h) = 0.15$, with $|S_1| = 30$
  - $e(j) = 0.25$, with $|S_2| = 5000$
- Then:
  - $d = |e(h) - e(j)|$

Evaluation Example

- The expected value:

$$\mu = |e_{estimation}(h) - e_{estimation}(j)| = |0.15 - 0.25| = 0.1$$

- The standard deviation:

$$\sigma^2 = \frac{e_{estimation}(h) \cdot [1 - e_{estimation}(h)]}{|S_1|} + \frac{e_{estimation}(j) \cdot [1 - e_{estimation}(j)]}{|S_2|}$$

$$\sigma = \sqrt{\frac{0.15(1-0.15)}{30} + \frac{0.25(1-0.25)}{5000}} = 0{,}0655 \ldots$$

Department of Mathematics
University of Calabria

Evaluation Example

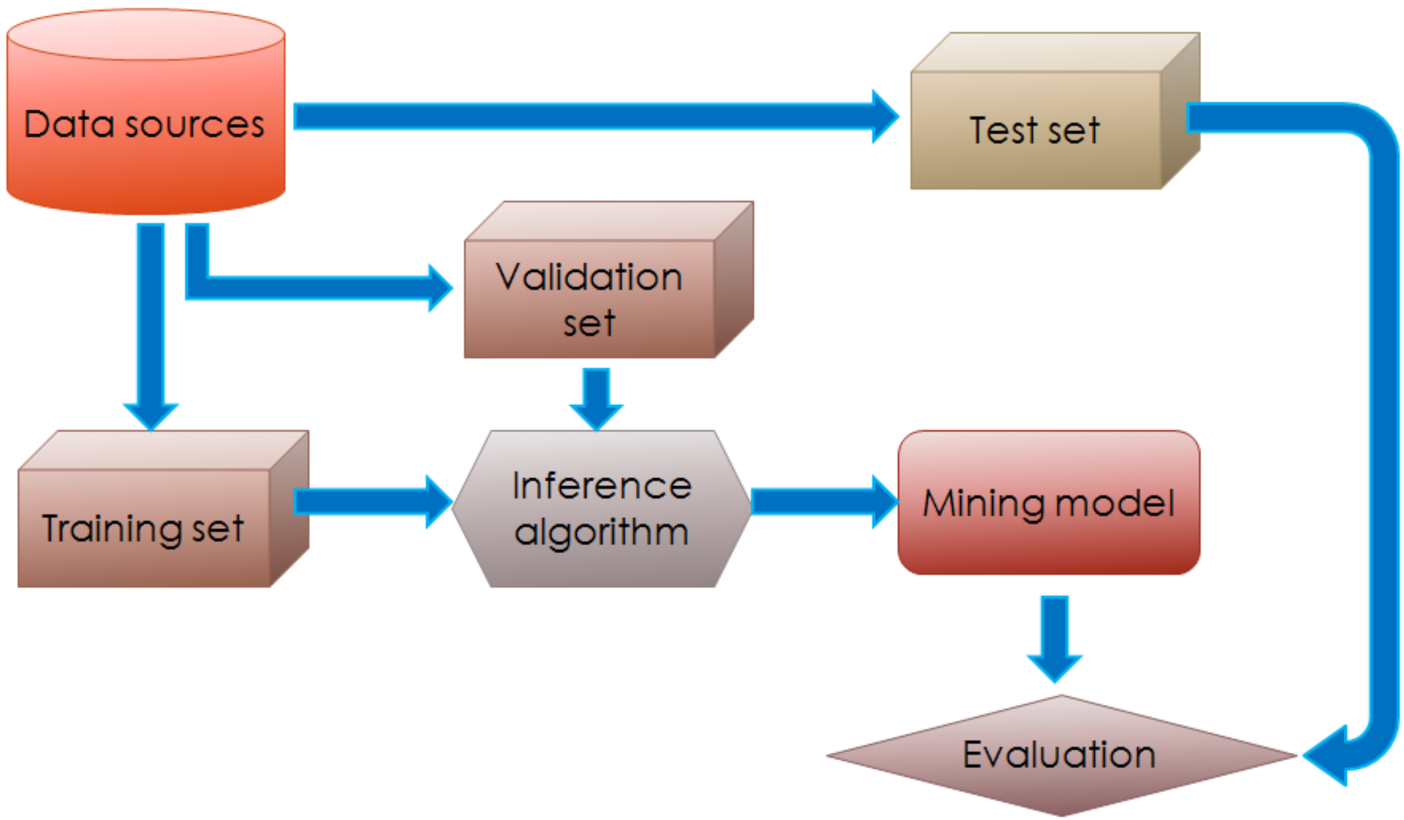
- With probability 0.95, the confidence interval is:

$$d_{true} \in \{0.1 - 0{,}0655; 0.1 + 0{,}0655\}$$

- The confidence interval does not contain 0:
  - The difference is statistically significant

Methods for model evaluation

- Hold-out



Methods for model evaluation

- Hold-out
  - Pros:
    - Fast evaluation
  - Cons:
    - Only one experiment ➔ low statistical relevance

Methods for model evaluation

- Repeated Hold-out with random sub-sampling
  - Choose *n*
  - *ResultList = { }*
  - *For 1 < i < n*
    - *Random Sampling of (with or without replacement):*
      - *Training set*
      - *Validation set*
      - *Test set*
    - *Model = buildModel(Training set, Validation set)*
    - *ResultList.add(evaluateModel(Model, Test set))*
  - *Return avg(ResultList)*

Methods for model evaluation

- Repeated Hold-out with random sub-sampling
  - Pros:
    - More statistical significance
  - Cons:
    - Slow evaluation
    - Not all the tuples are involved in the training and evaluation phase

Methods for model evaluation

- *k*-fold Cross Validation
  - *Choose k*
  - *Divide the whole dataset D in k folds (portion)*
  - *ResultList = { }*
  - *For 1 < i < k*
    - *Build Training set = D \ fold*$_i$
    - *Random sample the Validation Set from the Training Set*
    - *Training set = Training set \ Validation Set*
    - *Test set = fold*$_i$
    - *Model = buildModel(Training set, Validation set)*
    - *ResultList.add(evaluateModel(Model, Test set))*
  - *Return avg(ResultList)*

Methods for model evaluation

- *k*-fold Cross Validation
  - Pros:
    - Good statistical significance
      - the greater is *k* the better the significance
        - If $k = |D|$ Cross Validation is called leave-one-out evaluation
  - Cons:
    - Very slow evaluation
    - The *k*-fold Cross Validation needs to be stratified:
      - Each fold has to keep the same statistical properties of the whole dataset

Evaluation Metrics

- The focus is on the predictive quality of a model
  - instead of computational cost, scalability…

- Confusion Matrix

	Predicted class		
Actual class		Class = Yes	Class = No
	Class = Yes	True Positive (TP)	False Negative (FN)
	Class = No	False Positive (FP)	True Negative (TN)

Global Accuracy

- Global Accuracy

$$accuracy = \frac{TP + TN}{TP + FN + FP + TN}$$

- The number of all the well-predicted observation over the cardinality of the data set

Global Accuracy Limits

- Is a global accuracy of 99.9% good?
- Example:
  - Binary Classification
  - #records of class 0 = 9990
  - # records of class 1 = 10
  - A classifier that predicts always 0:
    - Global Accuracy = 99.9%
    - But the model is useless!

Department of Mathematics
University of Calabria

Cost Matrix

- Similar to the confusion matrix

	Predicted class		
Actual class	C(i|j)	Class = Yes	Class = No
	Class = Yes	C(Yes|Yes)	C(No|Yes)
	Class = No	C(Yes|No)	C(No|No)

- C(*i* | *j*) is the cost of predicting a record as class *i* when the actual class is *j*

Cost Evaluation of 2 Models (M1, M2)

Cost Matrix	Predicted class		
Actual class	C(i|j)	Yes	No
	Yes	-1	100
	No	1	0

Confusion Matrix M1	Predicted class		
Actual Class	C(i|j)	Yes	No
	Yes	150	40
	No	60	250

Accuracy: 0.8

Cost: 3910

Confusion Matrix M2	Predicted class		
Actual Class	C(i|j)	Yes	No
	Yes	250	45
	No	5	200

Accuracy: 0.9

Cost: 4255

Cost-sensitive Measures

- For each class
- Precision: the confidence of model
  - How much can I trust a prediction?

$$precision = \frac{TP}{TP + FP}$$

- Recall: the coverage of a model
  - How many records of a specific class can my model correctly predict?

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

- F1-Measure: harmonic mean of precision and recall

$$F_1 - Measure = \frac{2 \cdot precision \cdot recall}{precision + recall}$$

The Previous Example

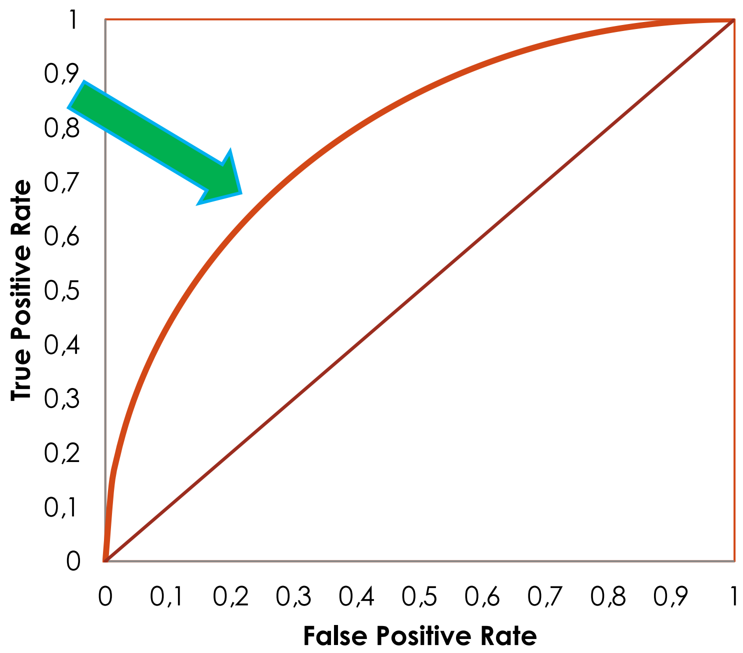
- Binary Classification
- #records of class 0 = 9990
- # records of class 1 = 10

- A classifier that predicts always 0:
  - Global Accuracy = 0.999
  - **Precision of class 1: *NaN* (0 / 0)**
  - **Recall of class 1: 0**
  - Precision of class 0: 0.999
  - Recall of class 0: 1

Department of Mathematics
University of Calabria

ROC (Receiver Operating Characteristic)

- The ROC curve is a graphical plot that illustrates the performance of a binary classifier system as its discrimination **threshold** is varied



$$TPR = recall = \frac{TP}{TP + FN}$$

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

Threshold

- Given a binary classifier the following rule holds:

$$\Pr(C = yes|\bar{t}) \geq \Pr(C = no|\bar{t}) \Rightarrow C = yes$$
$$\Pr(C = yes|\bar{t}) \geq 1 - \Pr(C = yes|\bar{t}) \Rightarrow C = yes$$
$$2 \cdot \Pr(C = yes|\bar{t}) \geq 1 \Rightarrow C = yes$$
$$\Pr(C = yes|\bar{t}) \geq 0.5 \Rightarrow C = yes$$

Standard threshold

Threshold

- What happens if we vary the threshold value?

Threshold

- What happens if we vary the threshold value?
  - For each threshold we have a different classification rule

Threshold

- What happens if we vary the threshold value?
  - For each threshold we have a different classification rule
  - For each rule we have a prediction

Threshold

- What happens if we vary the threshold value?
  - For each threshold we have a different classification rule
  - For each rule we have a prediction
  - For each prediction we have a confusion matrix

Threshold

- What happens if we vary the threshold value?
  - For each threshold we have a different classification rule
  - For each rule we have a prediction
  - For each prediction we have a confusion matrix
  - For each confusion matrix we have a FPR and a TPR

Threshold

- What happens if we vary the threshold value?
  - For each threshold we have a different classification rule
  - For each rule we have a prediction
  - For each prediction we have a confusion matrix
  - For each confusion matrix we have a FPR and a TPR
  - For each FPR and TPR we have a point in the ROC space

Threshold

- What happens if we vary the threshold value?
  - For each threshold we have a different classification rule
  - For each rule we have a prediction
  - For each prediction we have a confusion matrix
  - For each confusion matrix we have a FPR and a TPR
  - For each FPR and TPR we have a point in the ROC space
  - Examples:

$$\Pr(C = yes|\bar{t}) \geq 0.3 \Rightarrow C = yes$$

$$\Pr(C = yes|\bar{t}) \geq 0.75 \Rightarrow C = yes$$

Department of Mathematics
University of Calabria

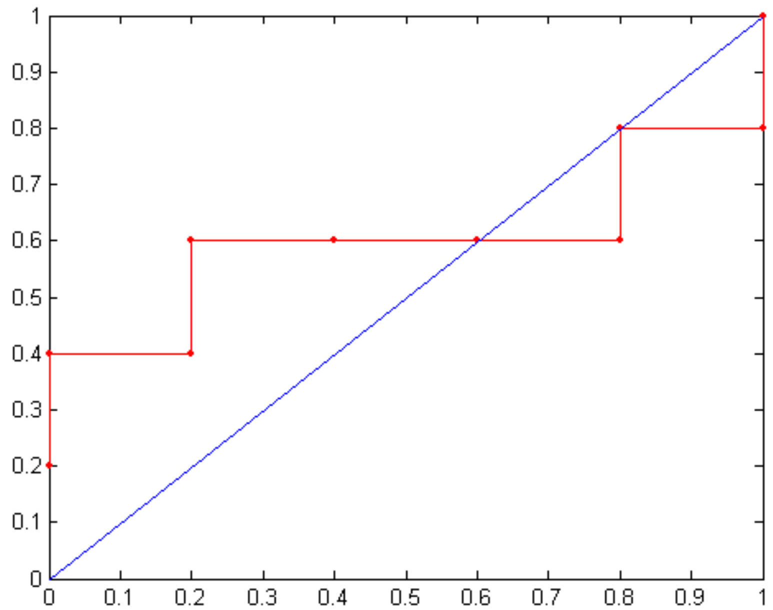
How to build a ROC curve

Instance	P(+|x)	True Class
1	0.95	+
2	0.93	+
3	0.87	-
4	0.85	-
5	0.85	-
6	0.85	+
7	0.76	-
8	0.53	+
9	0.43	-
10	0.25	+

- Sort the records according to P(+|x) [Descendent]
- Each P(+|x) will be a threshold
- For each threshold, compute the confusion matrix
- Compute FPR and TPR

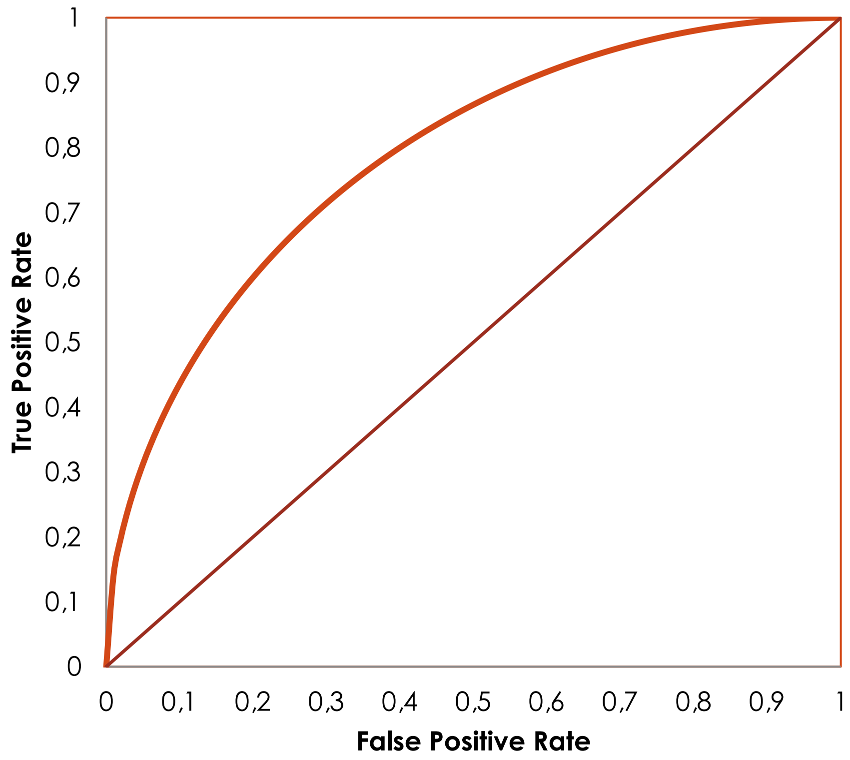
Class	+	-	+	-	-	-	+	-	+	+	
Threshold >=	0.25	0.43	0.53	0.76	0.85	0.85	0.85	0.87	0.93	0.95	1.00
TP	5	4	4	3	3	3	3	2	2	1	0
FP	5	5	4	4	3	2	1	1	0	0	0
TN	0	0	1	1	2	3	4	4	5	5	5
FN	0	1	1	2	2	2	2	3	3	4	5
TPR	1	0.8	0.8	0.6	0.6	0.6	0.6	0.4	0.4	0.2	0
FPR	1	1	0.8	0.8	0.6	0.4	0.2	0.2	0	0	0

ROC curve:

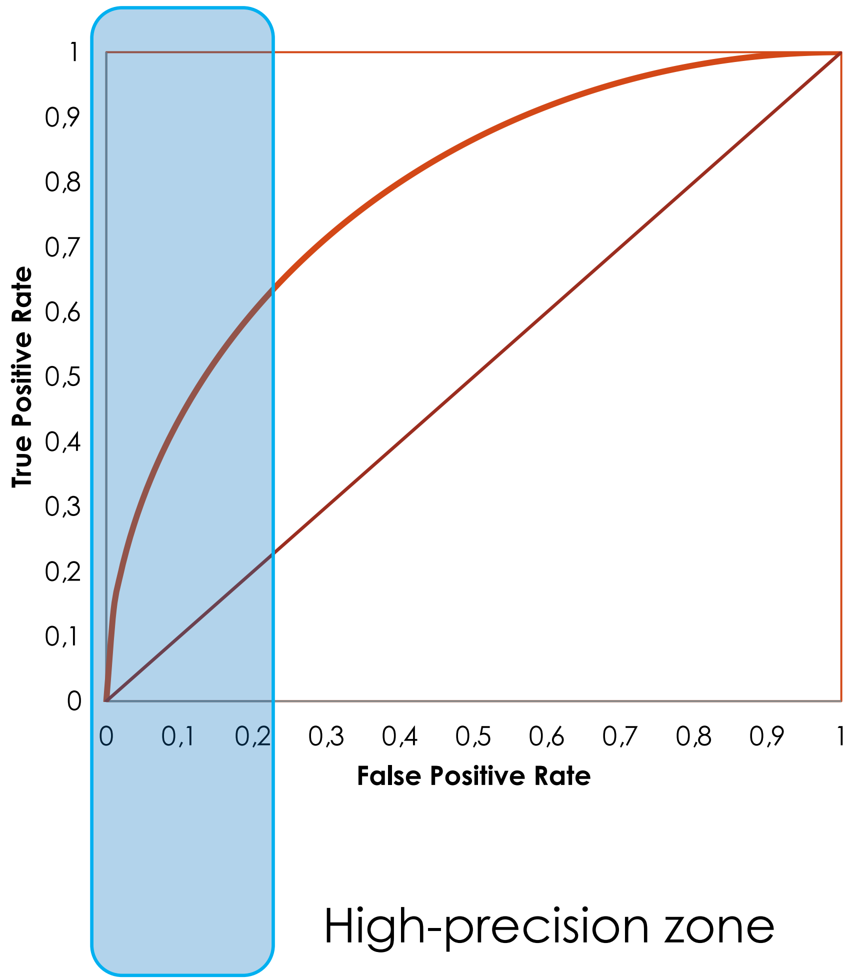


Department of Mathematics
University of Calabria

How to evaluate a ROC curve



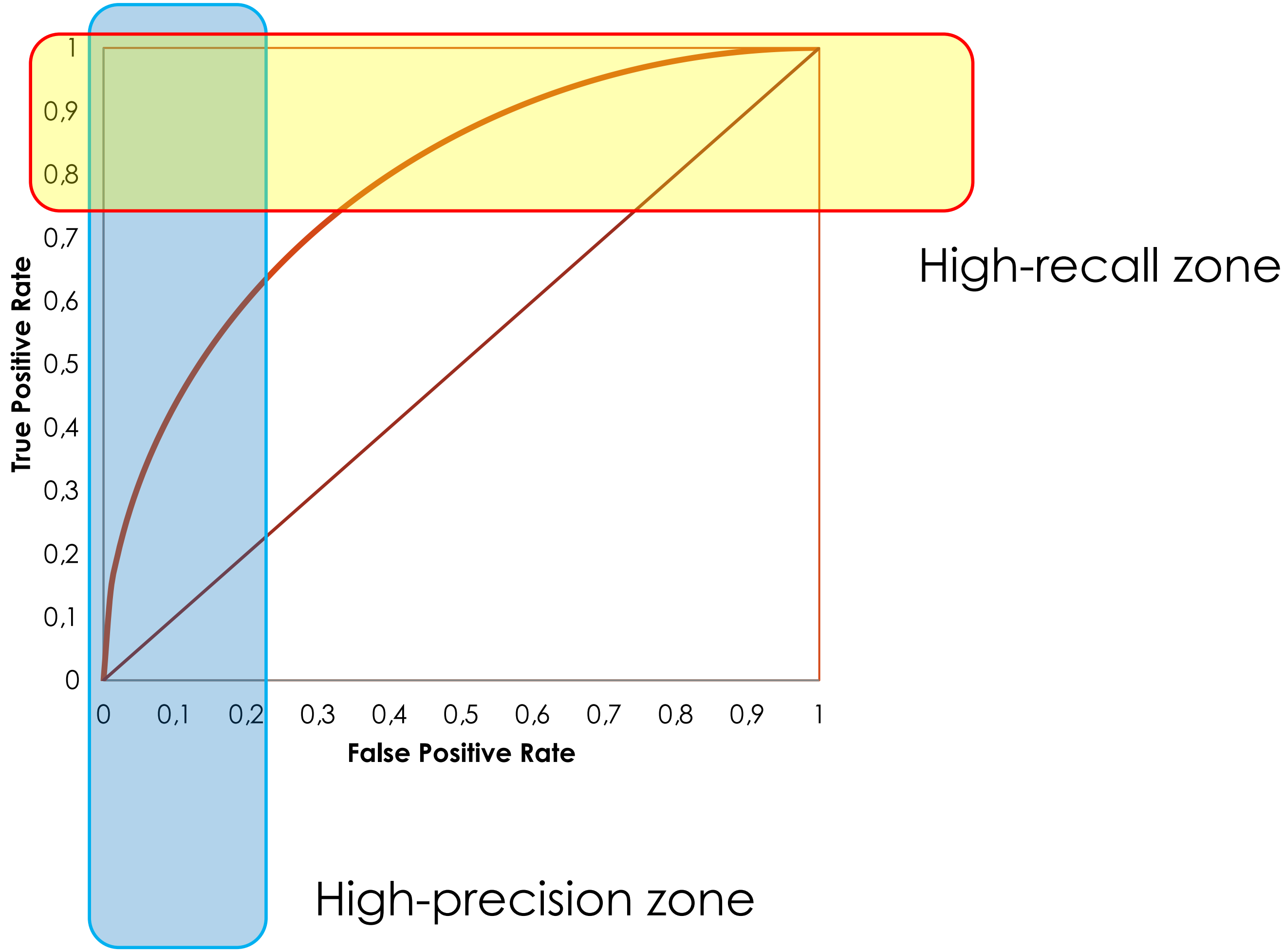
How to evaluate a ROC curve



High-precision zone

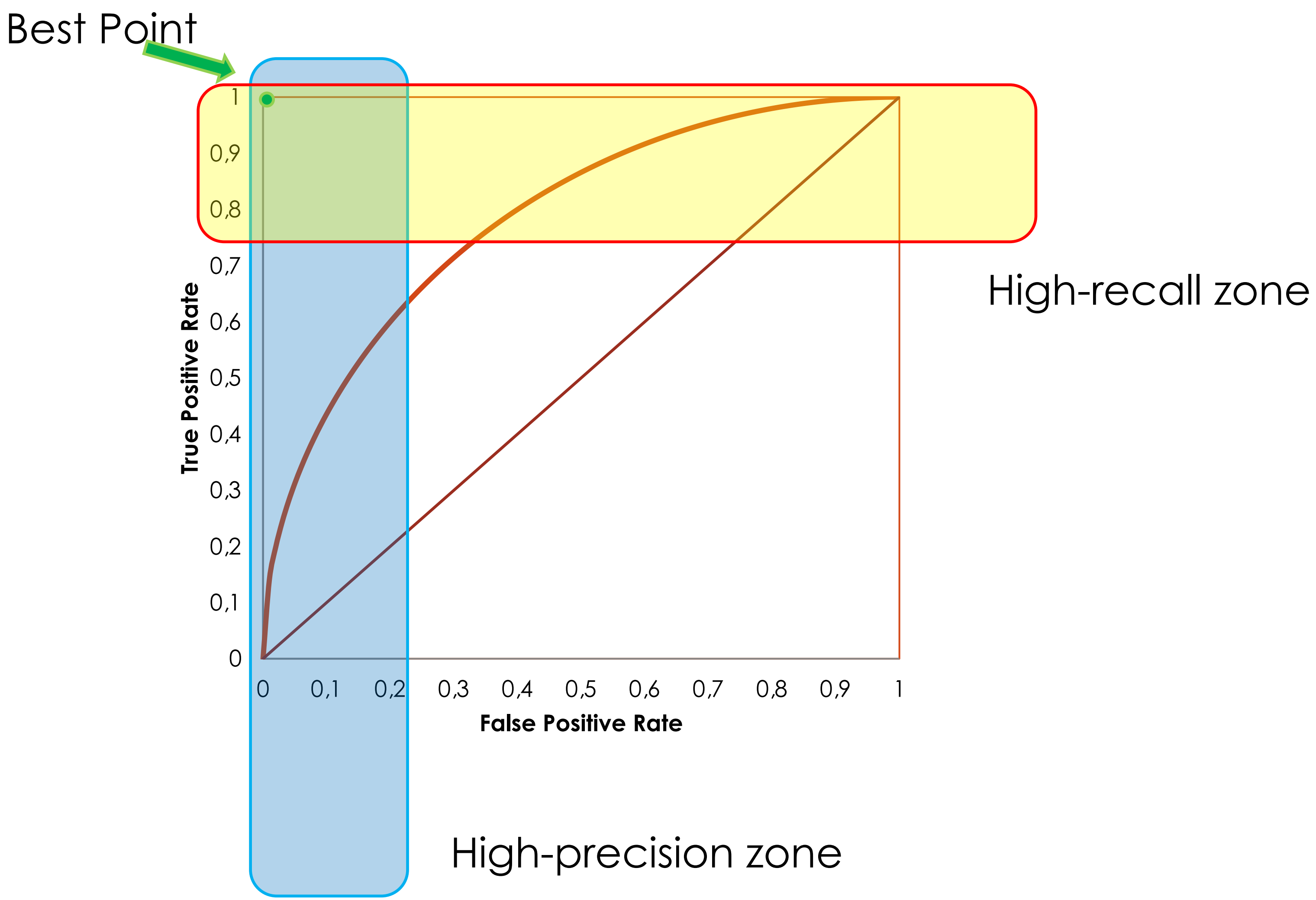
Department of Mathematics
University of Calabria

How to evaluate a ROC curve



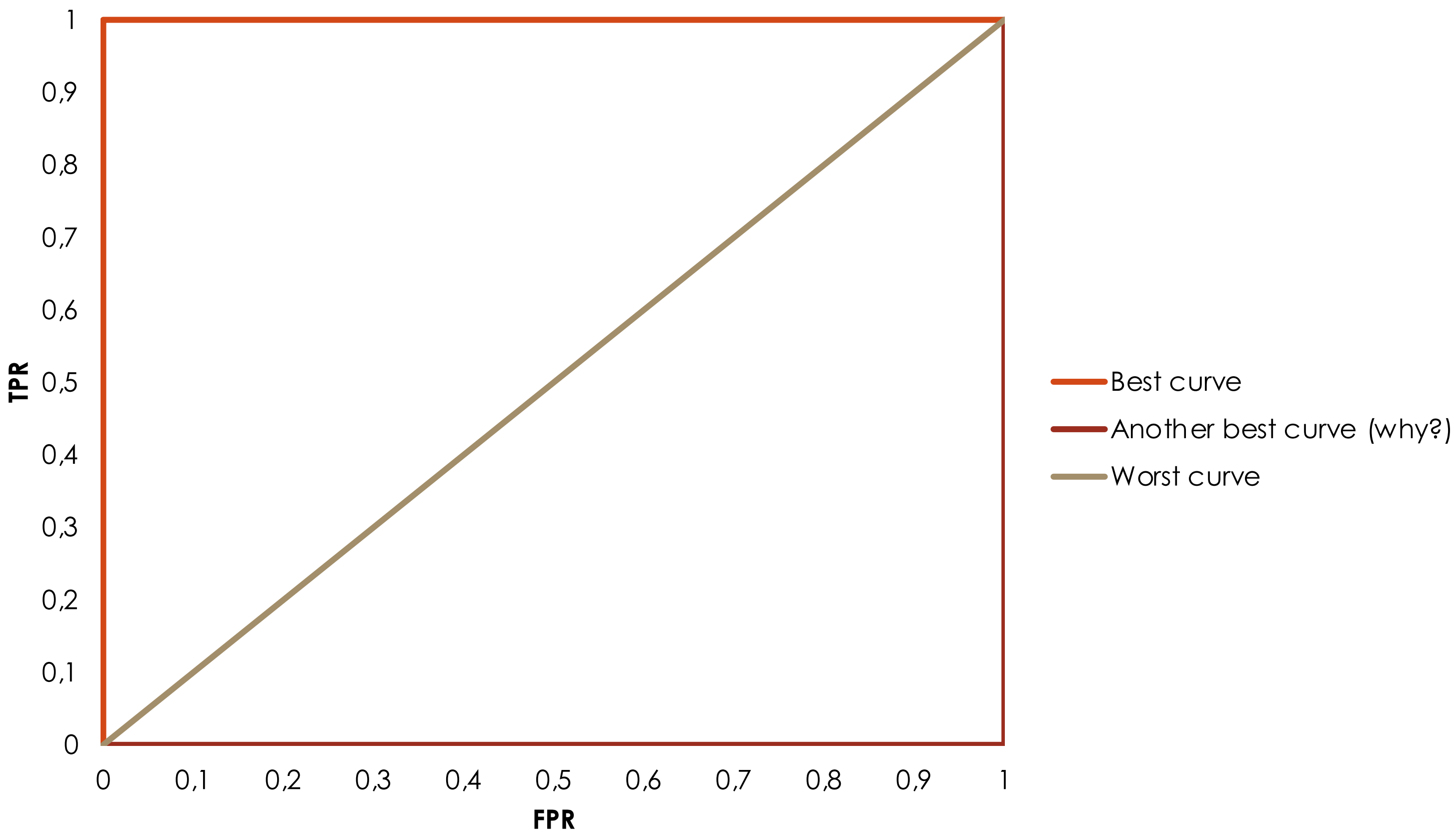
Department of Mathematics
University of Calabria

How to evaluate a ROC curve



Department of Mathematics
University of Calabria

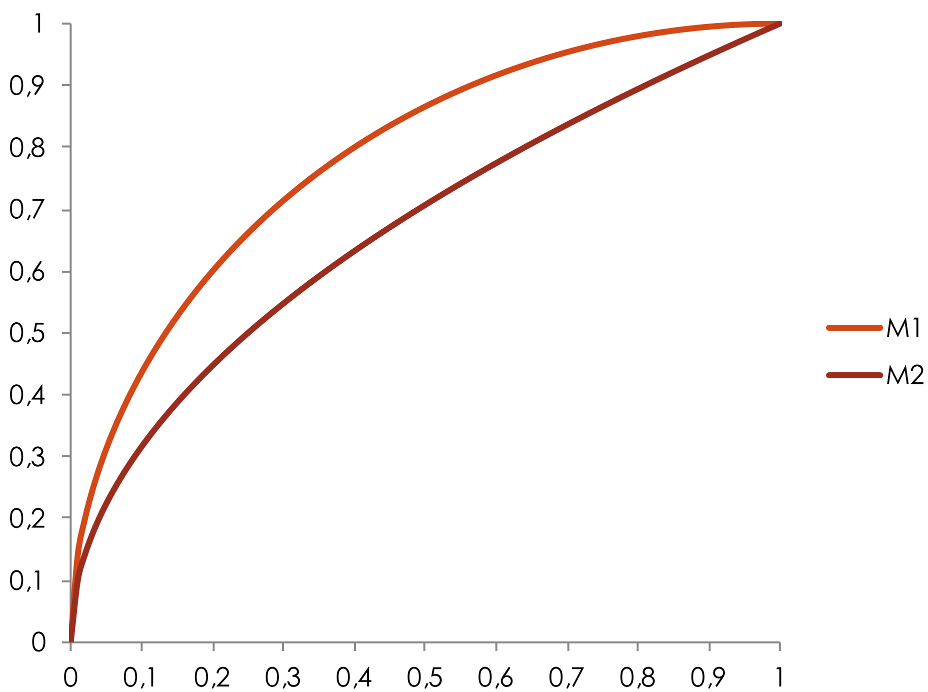
How to evaluate a ROC curve



Department of Mathematics
University of Calabria

ROC comparison

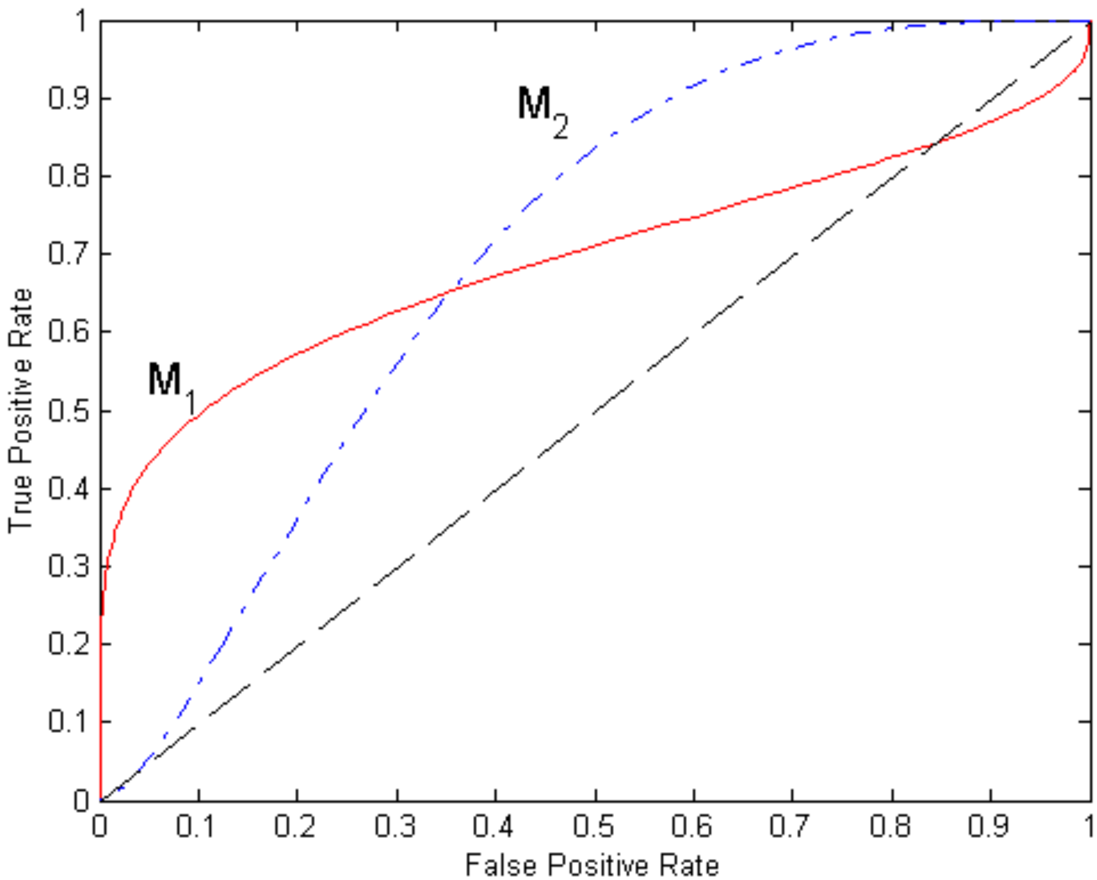
- The greater the area under the curve the better the quality of the model



Which is better?

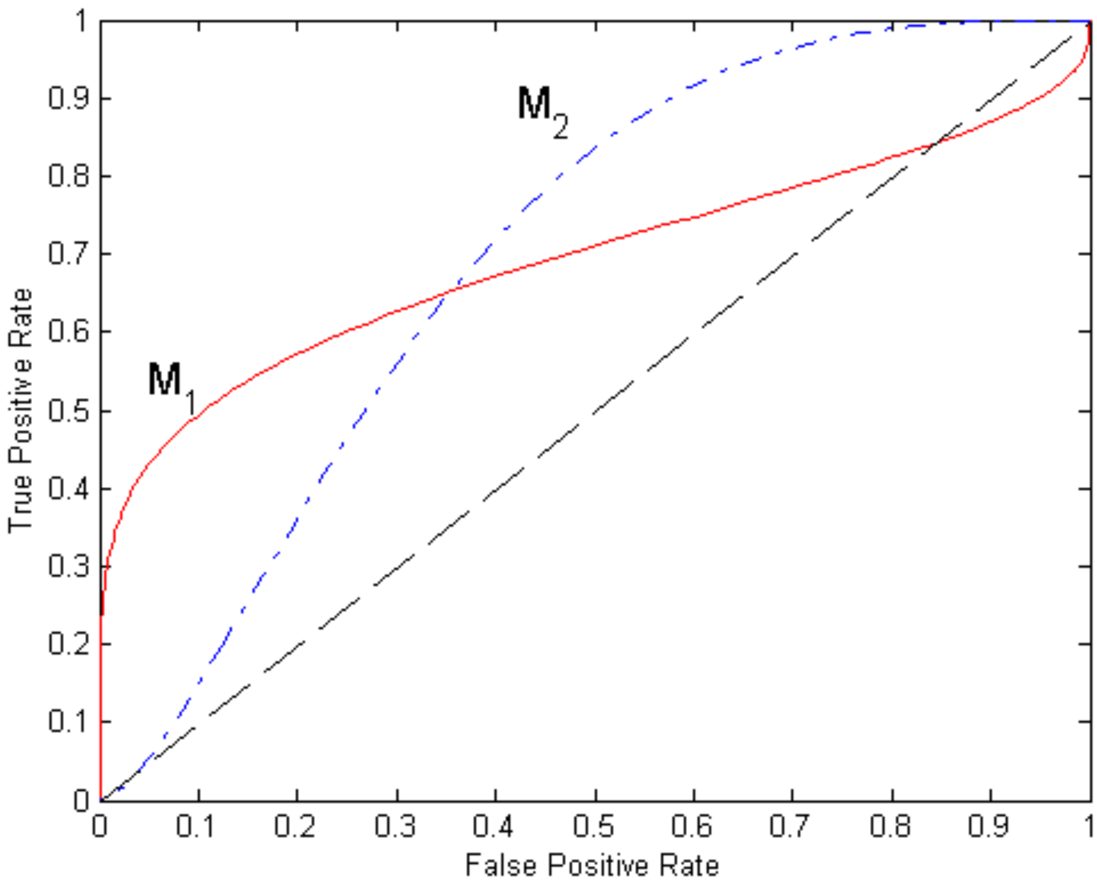
Department of Mathematics
University of Calabria

ROC comparison



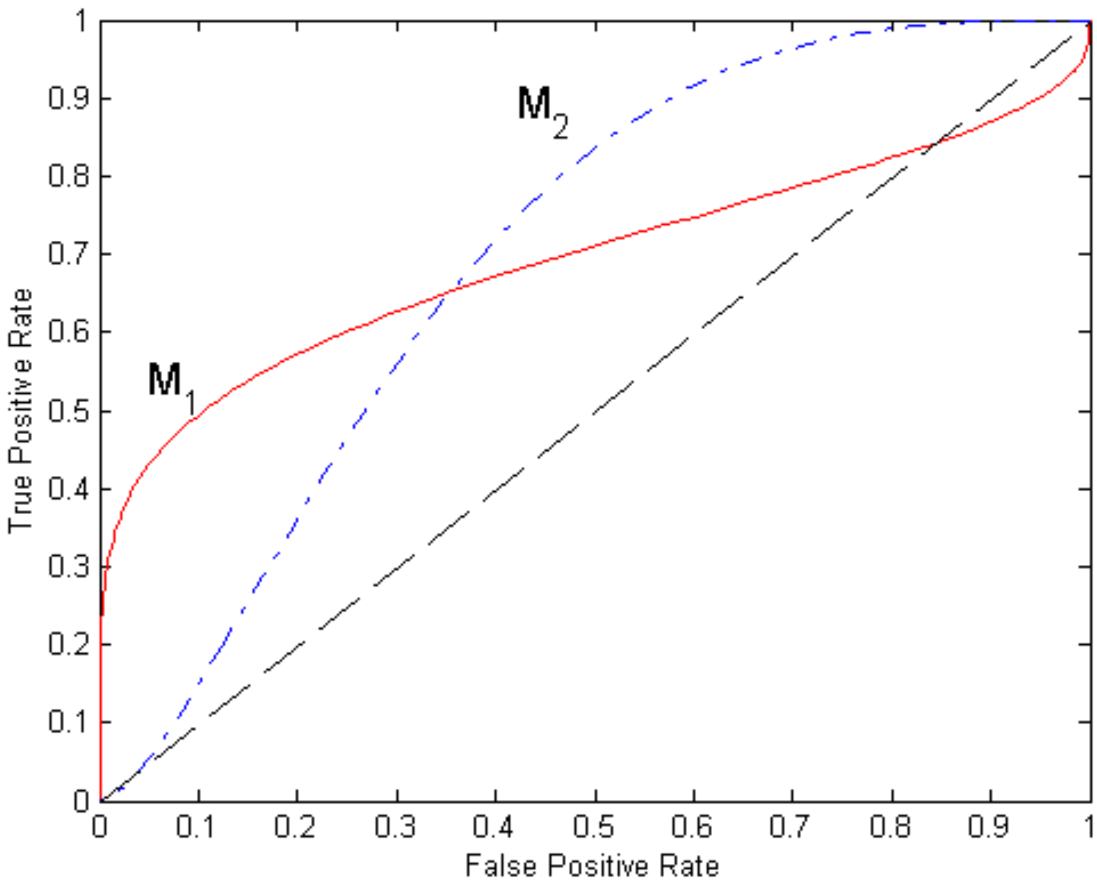
- Which is better?

ROC comparison



- Which is better?
- If we are interested in precision?

ROC comparison



- Which is better?
- If we are interested in precision?
- If we are interested in recall?

Confusion Matrix Glossary

confusion matrix

		Condition (as determined by "Gold standard")			
	Total population	Condition positive	Condition negative	Prevalence = $\frac{\Sigma\ \text{Condition positive}}{\Sigma\ \text{Total population}}$	
Test outcome	Test outcome positive	**True positive**	**False positive** (Type I error)	Positive predictive value (PPV, Precision) = $\frac{\Sigma\ \text{True positive}}{\Sigma\ \text{Test outcome positive}}$	False discovery rate (FDR) = $\frac{\Sigma\ \text{False positive}}{\Sigma\ \text{Test outcome positive}}$
Test outcome	Test outcome negative	**False negative** (Type II error)	**True negative**	False omission rate (FOR) = $\frac{\Sigma\ \text{False negative}}{\Sigma\ \text{Test outcome negative}}$	Negative predictive value (NPV) = $\frac{\Sigma\ \text{True negative}}{\Sigma\ \text{Test outcome negative}}$
	Positive likelihood ratio (**LR+**) = TPR/FPR	True positive rate (TPR, Sensitivity, Recall) = $\frac{\Sigma\ \text{True positive}}{\Sigma\ \text{Condition positive}}$	False positive rate (FPR, Fall-out) = $\frac{\Sigma\ \text{False positive}}{\Sigma\ \text{Condition negative}}$	Accuracy (ACC) = $\frac{\Sigma\ \text{True positive} + \Sigma\ \text{True negative}}{\Sigma\ \text{Total population}}$	
	Negative likelihood ratio (**LR−**) = FNR/TNR	False negative rate (FNR) = $\frac{\Sigma\ \text{False negative}}{\Sigma\ \text{Condition positive}}$	True negative rate (TNR, Specificity, SPC) = $\frac{\Sigma\ \text{True negative}}{\Sigma\ \text{Condition negative}}$		
	Diagnostic odds ratio (**DOR**) = LR+/LR−				

Terminology and derivations from a confusion matrix

true positive (TP)
eqv. with hit

true negative (TN)
eqv. with correct rejection

false positive (FP)
eqv. with false alarm, Type I error

false negative (FN)
eqv. with miss, Type II error

sensitivity or true positive rate (TPR)
eqv. with hit rate, recall

$$TPR = TP/P = TP/(TP+FN)$$

specificity (SPC) or true negative rate (TNR)

$$SPC = TN/N = TN/(FP+TN)$$

precision or positive predictive value (PPV)

$$PPV = TP/(TP+FP)$$

negative predictive value (NPV)

$$NPV = TN/(TN+FN)$$

fall-out or false positive rate (FPR)

$$FPR = FP/N = FP/(FP+TN)$$

false discovery rate (FDR)

$$FDR = FP/(FP+TP) = 1 - PPV$$

Miss Rate or False Negative Rate (FNR)

$$FNR = FN/P = FN/(FN+TP)$$

accuracy (ACC)

$$ACC = (TP+TN)/(P+N)$$

F1 score
is the harmonic mean of precision and sensitivity

$$F1 = 2TP/(2TP+FP+FN)$$

Matthews correlation coefficient (MCC)

$$\frac{TP \times TN - FP \times FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}}$$

Informedness = Sensitivity + Specificity - 1

Markedness = Precision + NPV - 1

Sources: Fawcett (2006) and Powers (2011).[2][3]