ROC curve

Receiving Operating Characteristics

A tool for the evaluation of binary classifiers

Ricco RAKOTOMALALA

Performance evaluation of classifiers

Evaluating the performance of classifiers is essential because we want…

1. To check the relevance of the model.
   Is the model really useful?

2. To estimate the accuracy in the generalization process.
   What is the probability of error when we apply the model on unseen instance?

3. To compare several models.
   Which is the most accurate one among several classifiers?

The error rate (computed on a test set) is the most popular summary measure because it is an estimator of the probability of misclassification (and it is easy to calculate).

Some indicators from the confusion matrix may be used also (recall / sensibility, precision). Other synthetic measures are possible (e.g. F-Measure).

The error rate is sometimes too simplistic

Standard process for the model evaluation

Dataset

Train (learning) set

$\Phi(X)$

Model 1 (M1)

$\Psi(X)$

Model 2 (M2)

Learning phase

Test set

	^positif	^négatif	Total
positf	40	10	50
négatif	10	40	50
Total	50	50	100

$\varepsilon(\Phi) = 20\%$

	^positif	^négatif	Total
positf	30	20	50
négatif	5	45	50
Total	35	65	100

$\varepsilon(\Psi) = 25\%$

Test phase
Confusion matrix

Conclusion: **Model 1** seems better than **Model 2**

This conclusion makes the assumption that we have an unit misclassification costs matrix (the error costs are symmetric) -- this is not true in most cases

Taking into consideration the misclassification costs matrix

Non-symmetrical misclassifications costs

	^positif	^négatif
positf	0	1
négatif	10	0

	^positif	^négatif	Total
positf	40	10	50
négatif	10	40	50
Total	50	50	100

	^positif	^négatif	Total
positf	30	20	50
négatif	5	45	50
Total	35	65	100

Average cost of misclassification →

$c(\Phi) = 1.1$

$c(\Psi) = 0.7$

Conclusion: Model 2 is better than Model 1 in this case?

Specifying the misclassification costs matrix is often difficult. The costs can vary according to the circumstances. Should we try a large number of matrices for comparing M1 and M2?

Can we use a tool which allows to compare the models regardless of the misclassification costs matrix?

The problem of imbalanced dataset

When the learning process deals with class imbalance, the confusion matrix and the error rate do not provide a good idea about the classifier relevance.

E.g. COIL 2000 – Challenge, detecting the customers which are interested in a caravan insurance policy

LINEAR DISCRIMINANT ANALYSIS

Train

0.0627			
Confusion matrix			
	No	**Yes**	**Sum**
No	5435	39	5474
Yes	326	22	348
Sum	5761	61	5822

Test

0.0650			
Confusion matrix			
	No	**Yes**	**Sum**
No	3731	31	3762
Yes	229	9	238
Sum	3960	40	4000

The test error rate of the default classifier (predicting systematically the most frequent class, here "No") is 238 / 4000 = 0.0595

Conclusion: The default classifier is always the best in class imbalance situation

This anomaly is due to the necessity to predict the class value, using a specific discrimination threshold. Yet, in numerous domains, the most interesting is to measure the propensity to be a positive class value (the class of interest - e.g. the propensity to purchase a product, the propensity to fail for a credit applicant, etc.).

ROC curve

The ROC curve is a tool for the performance evaluation and the comparison of classifiers

1. It does not depend on the misclassification costs matrix
   It enables to know if M1 (or M2) dominates M2 (or M1) whatever the misclassification costs matrix used

2. It is valid even in the case of imbalanced classes
   We evaluate the class probability estimates

3. The results are relevant when the test sample is not representative
   Even if the classes distribution of the test set do not provide a good estimation of the prior probability of classes

4. It provides a graphical tool which enables to compare classifiers
   We know immediately which are the interesting classifiers

5. It provides a synthetic measure of performance (AUC)
   Which is easy to interpret

Its scope goes beyond to the interpretations provided by the analysis of the confusion matrix (which depends on the discrimination threshold used) !

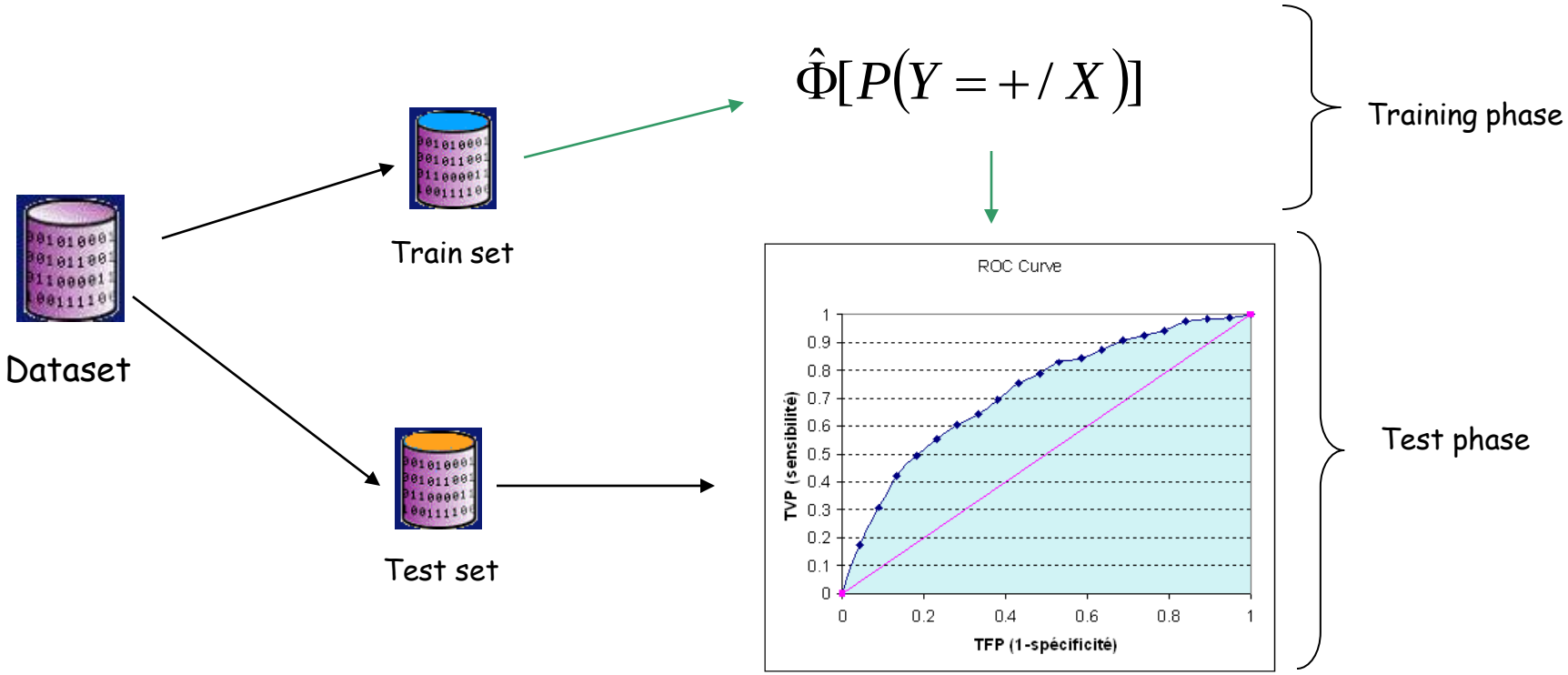
When and how to use the ROC curve

We deal with a binary problem Y = {+, -}

The "+" value is the target class

The classifier can provide an estimate of P(Y=+/X)

Or any SCORE that indicates the propensity to be "+" (which allows to sort the instances)



The analogy with the Gain Chart (in Customer Targeting) is tempting, but the use and the interpretation of the ROC curve is completely different.

Principle underlying the ROC curve

Confusion matrix

	^positif	^négatif
positf	TP	FN
négatif	FP	TN

TPR (True Positive Rate) = Recall = Sensibility = TP / Positives

FPR (False Positive Rate) = 1 – Specificity = FP / Negatives

The influence of the discrimination threshold

P(Y=+/X) >= P(Y=-/X) is equivalent to the decision rule P(Y=+/X) >= 0.5 (threshold = 0.5)

→ This decision rule provides a confusion matrix MC(1) with TPR(1) and FPR(1)

If we use another threshold (e.g. 0.6), we obtain another confusion matrix MC(2) with TPR(2) and FPR(2).

By varying the threshold, we have a succession of confusion matrices MC(i), for which we can calculate TPR(i) and FPR(i). The ROC curve is a scatter plot with FPR on the x-axis, and TPR on y-axis.

Constructing the ROC curve (1/2)

Sort the instances according to the score value (in descending order)

Individu	Score (+)	Classe
1	1	+
2	0.95	+
3	0.9	+
4	0.85	-
5	0.8	+
6	0.75	-
7	0.7	-
8	0.65	+
9	0.6	-
10	0.55	-
11	0.5	-
12	0.45	+
13	0.4	-
14	0.35	-
15	0.3	-
16	0.25	-
17	0.2	-
18	0.15	-
19	0.1	-
20	0.05	-

Positives = 6
Negatives = 14

Cut = 1

	^positif	^négatif	Total
positf	1	5	6
négatif	0	14	14
Total	1	19	20

TPR = 1/6 = 0.2 ; FPR = 0/14 = 0

Cut = 0.95

	^positif	^négatif	Total
positf	2	4	6
négatif	0	14	14
Total	2	18	20

TPR = 2/6 = 0.33 ; FPR = 0/14 = 0

Cut = 0.9

	^positif	^négatif	Total
positf	3	3	6
négatif	0	14	14
Total	3	17	20

TPR = 3/6 = 0.5 ; FPR = 0/14 = 0

Cut = 0.85

	^positif	^négatif	Total
positf	3	3	6
négatif	1	13	14
Total	4	16	20

TPR = 3/6 = 0.5 ; FPR = 1/14 = 0.07

Cut = 0

	^positif	^négatif	Total
positf	6	0	6
négatif	14	0	14
Total	20	0	20

TPR = 6/6 = 1 ; FPR = 14/14 = 1

Constructing the ROC curve (1/2)

FPR (x-axis) et TPR (y-axis)

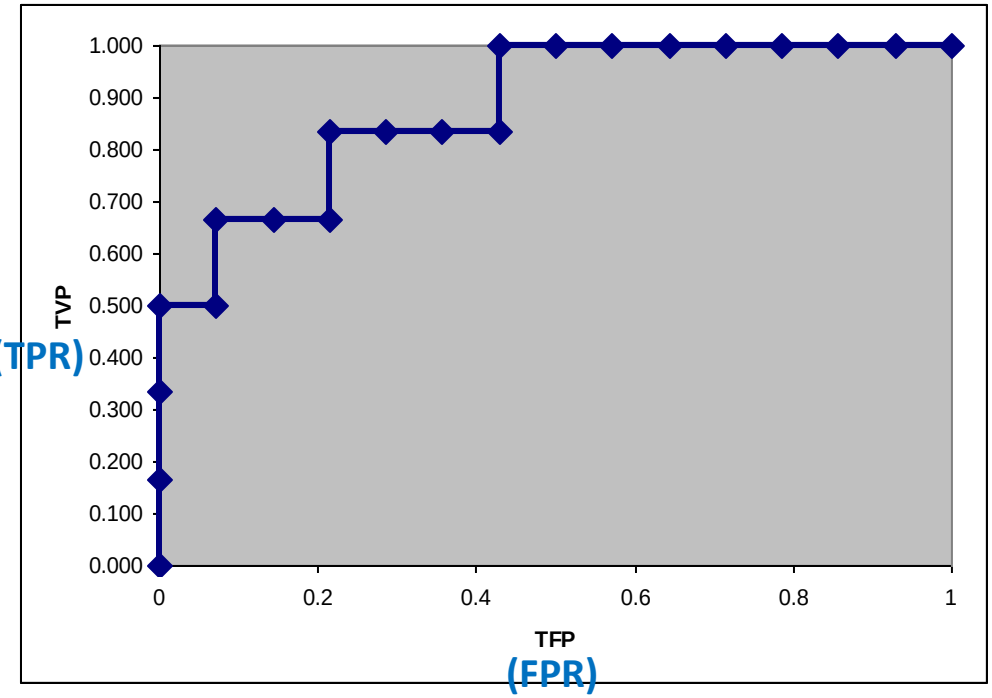
Individu	Score (+)	Classe	TFP	TVP
			0	0.000
1	1	+	0.000	0.167
2	0.95	+	0.000	0.333
3	0.9	+	0.000	0.500
4	0.85	-	0.071	0.500
5	0.8	+	0.071	0.667
6	0.75	-	0.143	0.667
7	0.7	-	0.214	0.667
8	0.65	+	0.214	0.833
9	0.6	-	0.286	0.833
10	0.55	-	0.357	0.833
11	0.5	-	0.429	0.833
12	0.45	+	0.429	1.000
13	0.4	-	0.500	1.000
14	0.35	-	0.571	1.000
15	0.3	-	0.643	1.000
16	0.25	-	0.714	1.000
17	0.2	-	0.786	1.000
18	0.15	-	0.857	1.000
19	0.1	-	0.929	1.000
20	0.05	-	1.000	1.000

Practical calculation

FPR (i) = Number of negatives among the first « i » instances / (total number of negatives)

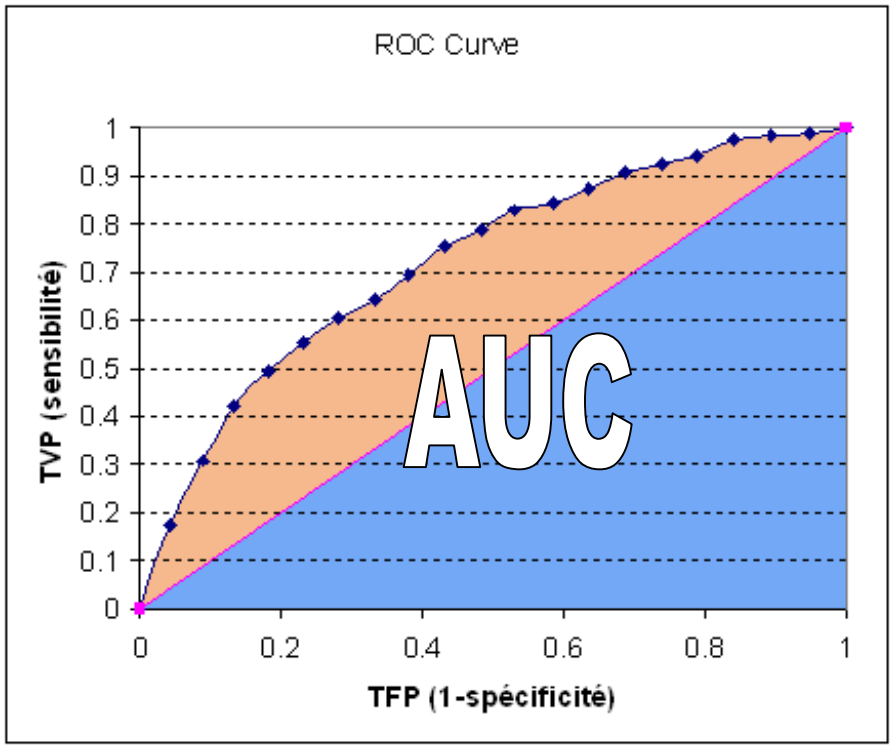
TPR (i) = Number of positives among the first « i » instances / (total number of positives)

ROC curve



Interpretation : AUC, area under curve

AUC corresponds to the probability of a positive instance to have a higher score than a negative instance (best situation AUC = 1)



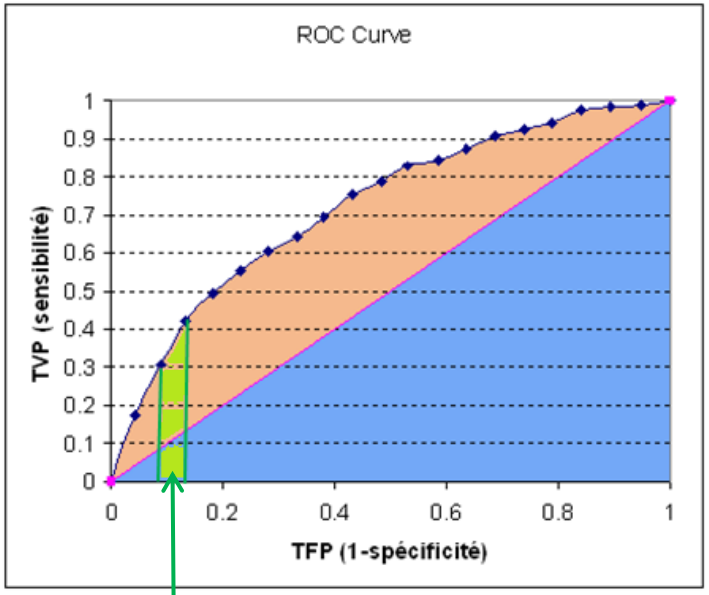
If the SCORE is assigned randomly to the individuals (the classifier is not better than random classifier), AUC = 0.5 → This is the diagonal line in the graphical representation

AUC – Practical calculation 1 – Numerical integration, trapezoidal rule

Approximate the curve with sum of trapezoids: area = AUC

Individu	Score (+)	Classe	TFP	TVP	Largeur	Hauteur	Surface
			0	0.000			
1	1	+	0.000	0.167	0.000	0.083	0.000
2	0.95	+	0.000	0.333	0.000	0.250	0.000
3	0.9	+	0.000	0.500	0.000	0.417	0.000
4	0.85	-	0.071	0.500	0.071	0.500	0.036
5	0.8	+	0.071	0.667	0.000	0.583	0.000
6	0.75	-	0.143	0.667	0.071	0.667	0.048
7	0.7	-	0.214	0.667	0.071	0.667	0.048
8	0.65	+	0.214	0.833	0.000	0.750	0.000
9	0.6	-	0.286	0.833	0.071	0.833	0.060
10	0.55	-	0.357	0.833	0.071	0.833	0.060
11	0.5	-	0.429	0.833	0.071	0.833	0.060
12	0.45	+	0.429	1.000	0.000	0.917	0.000
13	0.4	-	0.500	1.000	0.071	1.000	0.071
14	0.35	-	0.571	1.000	0.071	1.000	0.071
15	0.3	-	0.643	1.000	0.071	1.000	0.071
16	0.25	-	0.714	1.000	0.071	1.000	0.071
17	0.2	-	0.786	1.000	0.071	1.000	0.071
18	0.15	-	0.857	1.000	0.071	1.000	0.071
19	0.1	-	0.929	1.000	0.071	1.000	0.071
20	0.05	-	1.000	1.000	0.071	1.000	0.071

AUC	0.881



$$s_i = (FPR_i - FPR_{i-1}) \times \frac{TPR_i + TPR_{i-1}}{2}$$

Area of one trapezoid

$$AUC = \sum_i s_i$$

AUC = SUM (area of trapezoids)

AUC – Practical calculation 2 – Mann-Whitney statistic

Individu	Score (+)	Classe	Rangs	Rangs +
1	1	+	20	20
2	0.95	+	19	19
3	0.9	+	18	18
4	0.85	-	17	0
5	0.8	+	16	16
6	0.75	-	15	0
7	0.7	-	14	0
8	0.65	+	13	13
9	0.6	-	12	0
10	0.55	-	11	0
11	0.5	-	10	0
12	0.45	+	9	9
13	0.4	-	8	0
14	0.35	-	7	0
15	0.3	-	6	0
16	0.25	-	5	0
17	0.2	-	4	0
18	0.15	-	3	0
19	0.1	-	2	0
20	0.05	-	1	0

Somme (Rang +)	95
U+	74
AUC	**0.881**

- Mann-Whitney U nonparametric test: a population tends to have the same or larger values than the other?
- Based on ranks.
- In our context, we check if the "positive" instances have higher score than the "negative" ones.

Sum of ranks of "+" instances

$$S_+ = \sum_{i:y_i=+} r_i = 20+19+18+16+13+9 = 95$$

Mann-Whitney statistic

$$U_+ = S_+ - \frac{n_+(n_+ +1)}{2} = 95 - \frac{6\times 7}{2} = 74$$

AUC

$$AUC = \frac{U_+}{n_+ \times n_-} = \frac{74}{6\times 14} = 0.881$$

AUC – Practical calculation 3 – Counting the swaps

Individu	Score (+)	Classe	Nb de "-" devant un "+"
1	1	+	0
2	0.95	+	0
3	0.9	+	0
4	0.85	-	0
5	0.8	+	1
6	0.75	-	0
7	0.7	-	0
8	0.65	+	3
9	0.6	-	0
10	0.55	-	0
11	0.5	-	0
12	0.45	+	6
13	0.4	-	0
14	0.35	-	0
15	0.3	-	0
16	0.25	-	0
17	0.2	-	0
18	0.15	-	0
19	0.1	-	0
20	0.05	-	0

Swaps	10

AUC	0.881

Sort the instances according to the score (descending order)
For each "+", count the number of "-" ahead
The "swaps" are the sum of these counts

Swaps : sum of counts

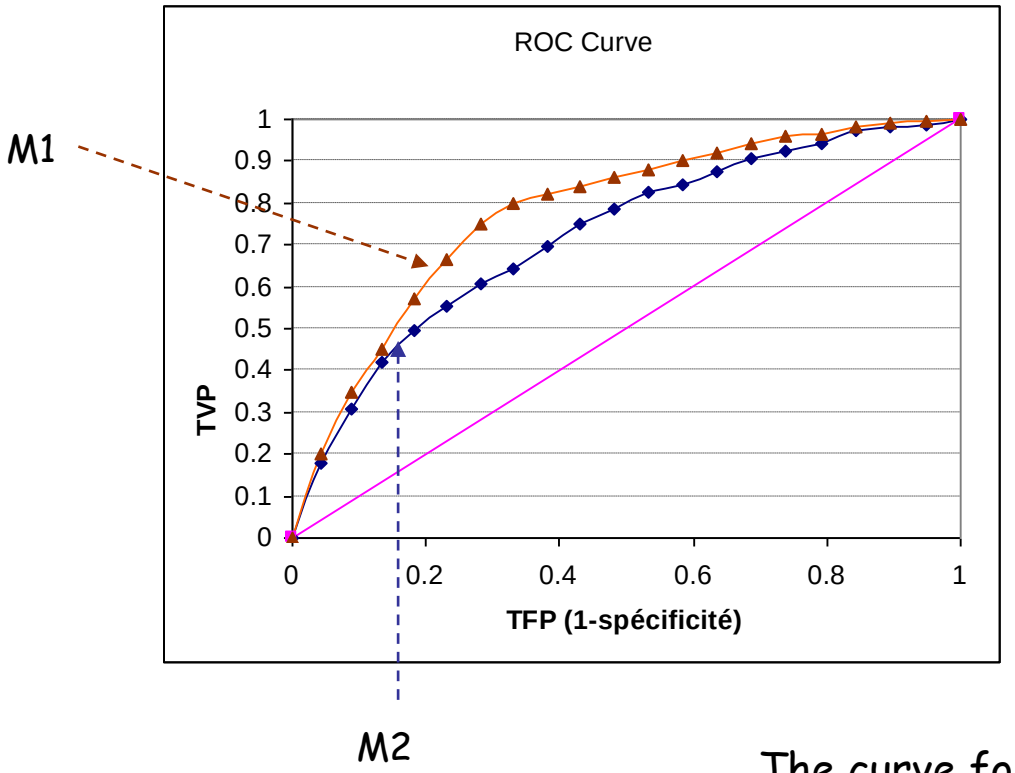
$$Swaps = \sum_{i:y_i=+} c_i = 0+0+0+1+3+6 = 10$$

AUC

$$AUC = 1 - \frac{Swaps}{n_+ \times n_-} = 1 - \frac{10}{6 \times 14} = 0.881$$

Notion of dominance

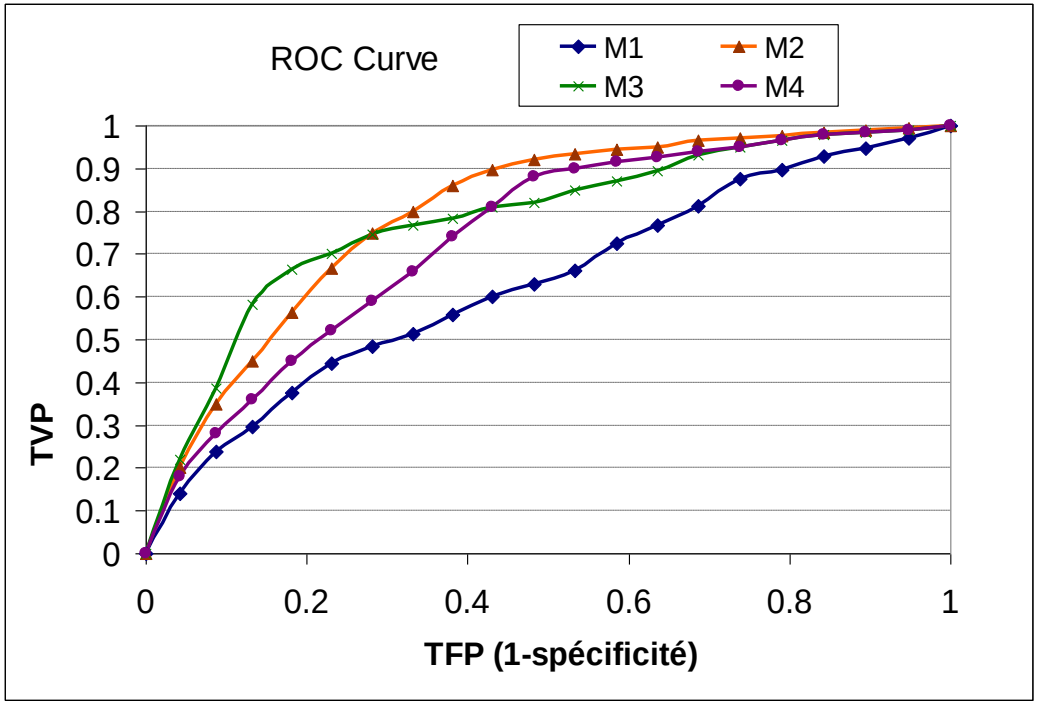
How to show that the classifier M1 is always better than M2 whatever the misclassification costs matrix used?



The curve for M1 is always above to the one of M2: there is no situation (misclassification costs matrix) for which M2 would be better than M1.

ROC Convex hull for model selection

Among a set of candidate models, how to exclude straightaway the ones which are not interesting?



Notion of "convex hull"

It is composed of the curves which, at one time or another, have no curve "above" them.

The curves on this envelope correspond to models that are potentially the most effective for a particular discrimination threshold.

Models that never participate in this envelope can be excluded.

In our example, the convex hull is composed of the curves of M3 and M2.

>> M1 is dominated by all models, it can be excluded.
>> M4 can be better than M3 in some circumstances but, in these cases, it is less good than M2. Thus, M4 can be excluded.

Conclusion

In many applications, the ROC curve provides more interesting information than the error rate.

>> This is especially true when we deal with a non representative test sample; in the case of imbalanced classes; when the misclassification costs are not well defined.

>> The ROC curve is effective only in the binary problems; the classifier must provide a score function for the target class P(Y = + /X) (or, at least, the propensity to be positive).

>> Some extensions of the ROC principle to multiclass classification problems exist but they often have a lack of simplicity, reducing the interest of the tool.