Mesures de performances d'un classifieur

1 - Mesurer l'exactitude à l'aide la validation croisée

Principe:

Au lieu d'avoir un seul ensemble de données d'entrainement (X_train) et un seul ensemble de données de test (X_test), l'idée de la validation croisée est de faire plusieurs entrainement (K fois, K étant un hyper paramètre, où chaque ensemble de données d'entrainement et de test sont définies comme suit :

On découpe les données d'entrainement en K blocs. On définit alors K cas d'entrainement du modèle en réalisant le découpage suivant:

(K-1) blocs pour les données d'entrainement et 1 bloc de données de test.

On choisit alors comme précision soit le meilleur score donné par un découpage ou bien une moyenne de tous les scores.

Cette méthode est très efficace lorsqu'on n'a pas beaucoup de données.

2 – Matrice de confusion

Principe :

Imaginons qu'on veuille construire le modèle d'un classifieur pour prédire de données suivant k classes (C_1,C_2,....C_k).

Il s'agit de calculer, une fois l'apprentissage terminé, le nombre de fois que le modèle a classé des données de *classe A comme étant des données de classe B* durant la phase de prédiction.

Principe de lecture de la matrice de confusion :

M[i,j] *donne le nombre de fois que la classe* ***i*** *a été prédite comme étant la classe* ***j***

Les différentes métriques

On classe les résultats en 4 catégories :

- **True Positive** (TP) : la prédiction et la valeur réelle sont positives.

Exemple : Une personne malade et prévu malade.

- **True Negative** (TN) : la prédiction et la valeur réelle sont négatives.

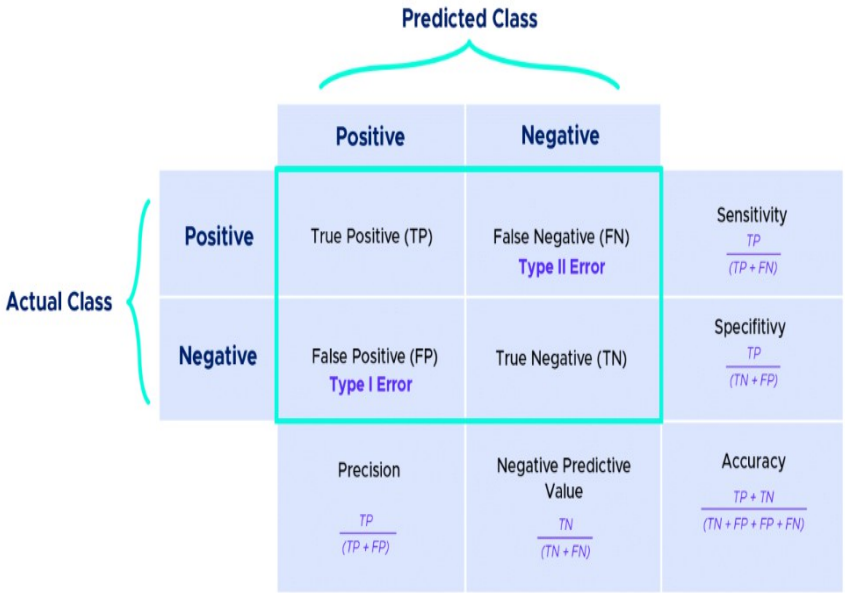
Exemple : Une personne saine et prévu saine.

- **False Positive** (FP) : la prédiction est positive alors que la valeur réelle est négative.

Exemple : Une personne saine et prévu malade.

- **False Negative** (FN) : la prédiction est négative alors que la valeur réelle est négative.

Exemple : Une personne malade et prévu saine.



Manque dans ce tableau une métrique appelée Rappel : **Rappel= TP/(TP+FN)**

Exemple : Suite à l'application d'un modèle prédictif, nous obtenons les résultats suivants

		Actual			
		A	B	C	SUM
Predicted	A	1000	400	200	1200
	B	600	1200	200	2400
	C	400	400	1600	2400
	SUM	2000	2000	2000	

. 1000 individus ayant été classés comme appartenant à la classe A sur un total de 2000

. Pour les individus de la classe B, 1200 sur 2000 ont bien été identifiés comme appartenant à cette classe.

. Pour les individus de la classe C, 1600 sur 2000 ont bien été identifiés.

Le nombre de True Positive (TP) est donc de 3800

Pour avoir le nombre ***False Positive (FP)***,***True Negative (TN)***,***False Negative (FN*)**, Il n'est pas possible sur ce tableau de les calculer directement, il faudrait alors le séparer en trois cas (se retrouver sur un cas binaire)

A et (B et C)
B et (A et C)
C et (A et B)

Exemple : Schéma de calcul des métriques pour la prédiction du chiffre '5' du dataset Mnist

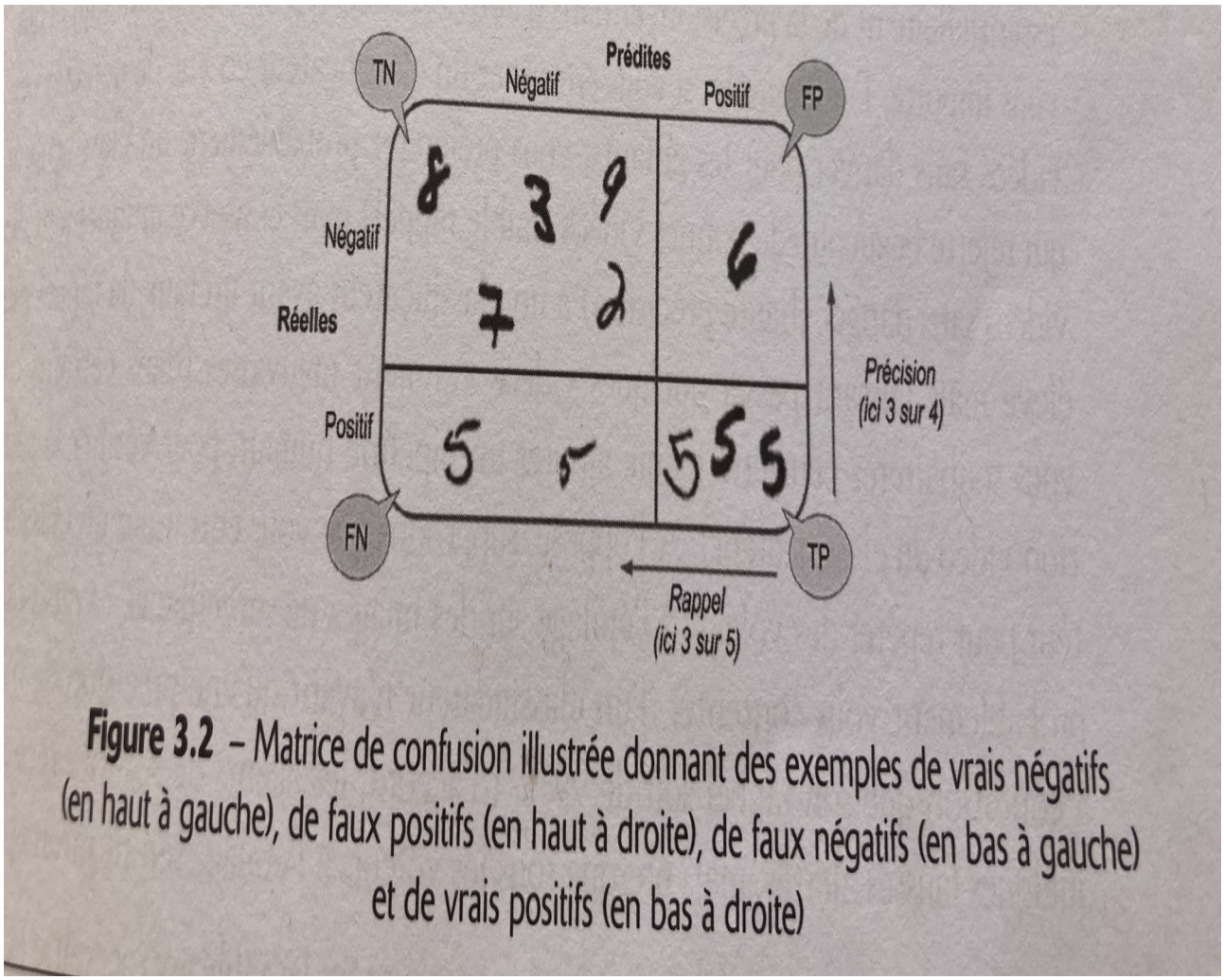


Figure 3.2 – Matrice de confusion illustrée donnant des exemples de vrais négatifs (en haut à gauche), de faux positifs (en haut à droite), de faux négatifs (en bas à gauche) et de vrais positifs (en bas à droite)

Le F1_Score

Il est souvent pratique de **combiner précision et rappel en une métrique unique appelée F1_Score**.

C'est très utile notamment pour comparer les performances de deux modèles de classification

Le **F1_Score** est une *moyenne harmonique* de la précision et du rappel.

F1_score= (precision*rappel)/(précsion+rapel).

Un bon classifieur a un bon score lorsqu son rappel et sa précision sont élevés.

Commentaires sur le F1_Score

Le **F1_score** favorise les classifieurs ayant une précision et un rappel similaire. Mais ne n'est pas toujours ce que souhaite le DataScientist qui conçoit le modèle de prédiction: Dans certains cas c'est le **score de la précision** qui est *souhaité* et d'autres cas c'est **le score du rappel**.

Exemple pour illustrer ceci:

. Si on entraine un modèle en vu de détecter des vidéos sans danger pour les enfants, on va préférer probablement un classifieur qui rejette beaucoup de bonnes vidéos (*faible rappel*) mais conserve uniquement des vidéos sans danger (*haute précision*).

. A l'opposé, supposons qu'on entraine un classifieur pour repérer des voleurs à l'étalage sur des images de surveillance : on peut se contenter d'un classifieur ayant une précision faible de 30%, à condition que son rappel soit de 99% (en d'autres termes : les agents de sécurité recevront beaucoup de fausses alertes (***précision faible***) mais presque tous les voleurs à l'étalage seront interceptés (***rappel élevé***).

Moralité: Il faut trouver un compromis précision/rappel en fonction de l'application auquel est destiné le modèle