Intégrer les coûts de mauvais classement en apprentissage supervisé

Prise en compte des coûts de mauvaise affectation lors de l'évaluation et la construction des modèles de prédiction

Ricco RAKOTOMALALA

PLAN

Prise en compte des coûts en apprentissage supervisé : position du problème

Les coûts sont indissociables de la prise de décision

L'objectif de l'apprentissage supervisé est de construire, à partir des données, une fonction de liaison entre Y et (X1, X2,…) qui soit la plus précise possible

$$Y = f(X_1, X_2, \ldots, X_J; \theta)$$

Pour quantifier « la plus précise possible », on utilise usuellement le taux d'erreur. Il indique la probabilité d'erreur du modèle.

$$ET = \frac{1}{card(\Omega)} \sum_{\Omega} \Delta[Y, \hat{f}(X, \hat{\alpha})]$$

$$où\ \Delta[.] = \begin{cases} 1 si\ Y \neq \hat{f}(X, \hat{\alpha}) \\ 0 si\ Y = \hat{f}(X, \hat{\alpha}) \end{cases}$$

Mais les conséquences des mauvaises affectations ne sont jamais symétriques, cet indicateur n'en tient pas compte :

- Désigner comme « malade » un individu « sain » n'implique pas les mêmes conséquences que de désigner comme « sain » un individu « malade »
- Accuser de fraude un innocent est différent de laisser passer un fraudeur

Ce constat est d'autant plus important que les positifs sont généralement rares dans la population (les malades ne sont pas légion, les fraudeurs itou, … heureusement)

Quelles sont les questions qui se posent ?

(1) Comment exprimer les conséquences des mauvaises affectations ?

→ *On utilise généralement une matrice de coûts*

Cas où Y possède K=2 modalités *(le plus fréquent)*

Y \ $\hat{Y}$	$\hat{+}$	$\hat{-}$
$+$	α	β
$-$	γ	δ

Remarques :

• Usuellement $\alpha=\beta=0$; mais pas toujours, parfois $\alpha,\beta < 0$, le coût est négatif c.-à-d. un gain (ex. accorder un crédit à un client fiable)

• Si $\alpha=\beta=0$ et $\gamma=\delta=1$, nous sommes dans le schéma initial de la minimisation de l'erreur

(2) Comment exploiter les coûts lors de l'évaluation des modèles ?

→ *Toujours à partir de la matrice de confusion qui indique la structure de l'erreur*

→ *Exploiter la matrice de coûts. Objectif : produire un indicateur synthétique qui permet la comparaison des modèles*

(3) Comment exploiter les coûts lors de la construction de modèles ?

→ *Le modèle de référence est celui élaboré sans les coûts*

→ *On doit faire mieux c.-à-d. obtenir une meilleure évaluation si on exploite judicieusement les coûts lors de la construction du classifieur*

Evaluation des classifieurs :
le coût moyen de mauvais classement

Un indicateur synthétique tenant compte des coûts - Présentation

Y \ $\hat{Y}$	$\hat{+}$	$\hat{-}$
$+$	a	b
$-$	c	d

Matrice de confusion : indique la structure de l'erreur du modèle c.-à-d. de quelle manière se trompe le modèle

Y \ $\hat{Y}$	$\hat{+}$	$\hat{-}$
$+$	α	β
$-$	γ	δ

Matrice des coûts de mauvais classement : quantifie les erreurs c.-à-d. ce que coûte chaque type d'erreur

Coût moyen de mauvais classement du modèle (M)

$$C(M) = \frac{1}{n}(a \times \alpha + b \times \beta + c \times \gamma + d \times \delta)$$

Nous utiliserons ce critère pour évaluer/comparer les stratégies d'apprentissage.

Commentaires :

Son interprétation est assez difficile…

Quoiqu'il en soit, il permet de comparer les performances des modèles

Plus faible sera le coût moyen de mauvais classement, meilleur sera le modèle

Le calcul doit être réalisé sur un échantillon test (ou en validation croisée, bootstrap,…)

Un indicateur synthétique tenant compte des coûts – Un exemple

(M1)

Observée \ Prédite	+	-	Total
+	40	10	50
-	20	30	50
Total	60	40	100

$$C(M1) = \frac{1}{100}(40\times(-1) + 10\times 10 + 20\times 5 + 30\times 0) = 1.6$$

Y \ $\hat{Y}$	$\hat{+}$	$\hat{-}$
$+$	-1	10
$-$	5	0

- *Les taux d'erreurs sont identiques ($\varepsilon = 30\%$)*
- *Mais lorsqu'on tient compte des coûts, on se rend compte que M1 est nettement meilleur que M2*
- *C'est normal, M2 se trompe beaucoup là où c'est le plus coûteux (Prédire « négatif » un individu « positif » → 30 individus)*

(M2)

Observée \ Prédite	+	-	Total
+	20	30	50
-	0	50	50
Total	20	80	100

$$C(M2) = \frac{1}{100}(20\times(-1) + 30\times 10 + 0\times 5 + 50\times 0) = 2.8$$

Un indicateur synthétique tenant compte des coûts – Le taux d'erreur

Le taux d'erreur est un cas particulier du coût moyen de mauvais classement.

La matrice de coûts est la matrice identité (unitaire et symétrique)

Y \ $\hat{Y}$	$\hat{+}$	$\hat{-}$
$+$	0	1
$-$	1	0

$$C(M) = \frac{1}{100}(40\times 0 + 10\times 1 + 20\times 0 + 30\times 1) = 0.3$$
$$= \frac{20+10}{100} = 0.3$$

		Prédite		
		+	-	Total
Observée	+	40	10	50
	-	20	30	50
	Total	60	40	100

Il y a donc des hypothèses implicites dans le taux d'erreur :

- toutes les erreurs « coûtent » pareil
- ne pas se tromper ne coûte rien (mais ne produit aucun gain non plus)

Un indicateur synthétique tenant compte des coûts – Généralisation à K > 2 classes

Pour Y à K>2 modalités, le coût moyen de mauvais classement s'écrit

Pour une matrice de confusion (n_{ik}) Nombre d'observations classés Yk et qui sont réalité des Yi, avec $\sum_i \sum_k n_{ik} = n$

Une matrice de coût de mauvais classement (c_{ik}) Le coût de classer un individu Yk alors qu'il est en réalité un Yi

Coût de mauvais classement d'un modèle M $$C(M) = \frac{1}{n} \sum_i \sum_k n_{ik} \times c_{ik}$$

Un exemple : détection du "churn"

Détection des clients qui font défection

Domaine : Secteur de la téléphonique (Opérateur téléphonique)

Objectif : Détecter les clients qui vont faire défection (résilier leur abonnement et vraisemblablement partir à la concurrence) et tenter de les retenir en leur proposant des offres promotionnelles

Variable à prédire : CHURN – oui (+) ou non (-)

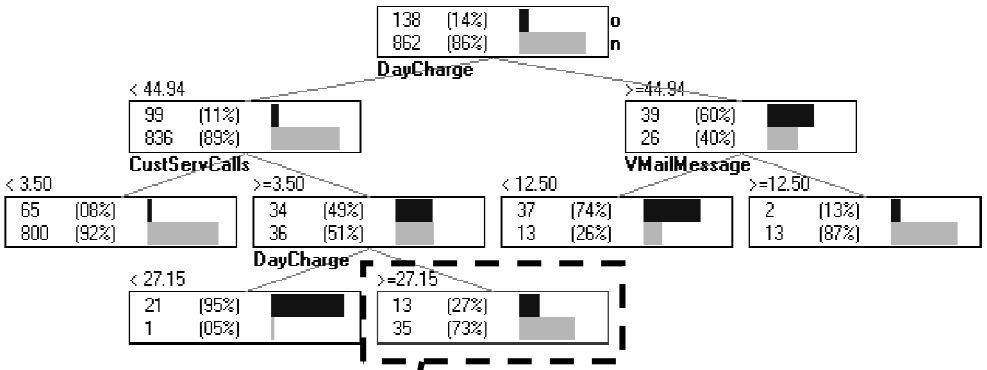
Variables prédictives : comportement du client et usage des différents services proposés

Observations : 1000 individus en apprentissage ; 2333 en test

Matrice de coût (*on peut essayer différentes hypothèses dans la pratique*)

Y \ $\hat{Y}$	+	−
+	– 15	10
−	2	0

Arbre de décision (*parmi les solutions possibles*)



On s'intéresse à une feuille en particulier, nous calculons les probabilités d'appartenance P(Y/X)

$$P(Y = + / DC < 44.94; CSC \geq 3.5; DC \geq 27.15) = \frac{13}{48} = 0.27$$

$$P(Y = - / DC < 44.94; CSC \geq 3.5; DC \geq 27.15) = \frac{35}{48} = 0.73$$

Stratégie 1 : ignorer les coûts

Stratégie 1 : ignorer totalement les coûts

Stratégie 1 :

- Ne pas tenir compte des coûts lors de la construction du classifieur

C.-à-d. lors de l'estimation des probabilités P(Y/X) à partir des données

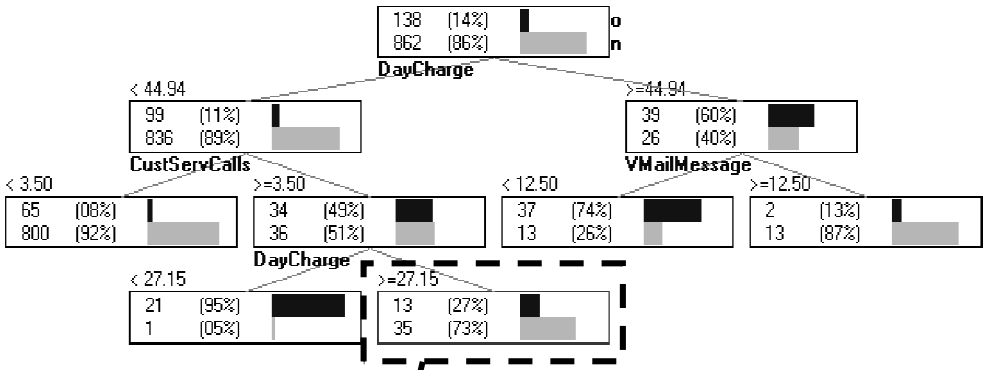
- Ne pas tenir compte des coûts lors de la prédiction de la classe d'appartenance

On s'en tient au schéma de minimisation de l'erreur de prédiction

La règle d'affectation utilisée demeure

Cf. les bases de l'apprentissage supervisé

$$y_{k*} = \arg\max_k P(Y = y_k / X)$$



$$P(Y = + / DC < 44.94; CSC \geq 3.5; DC \geq 27.15) = \frac{13}{48} = 0.27$$

$$P(Y = - / DC < 44.94; CSC \geq 3.5; DC \geq 27.15) = \frac{35}{48} = 0.73$$

Si cette règle est déclenchée lors du classement d'un nouvel individu, alors

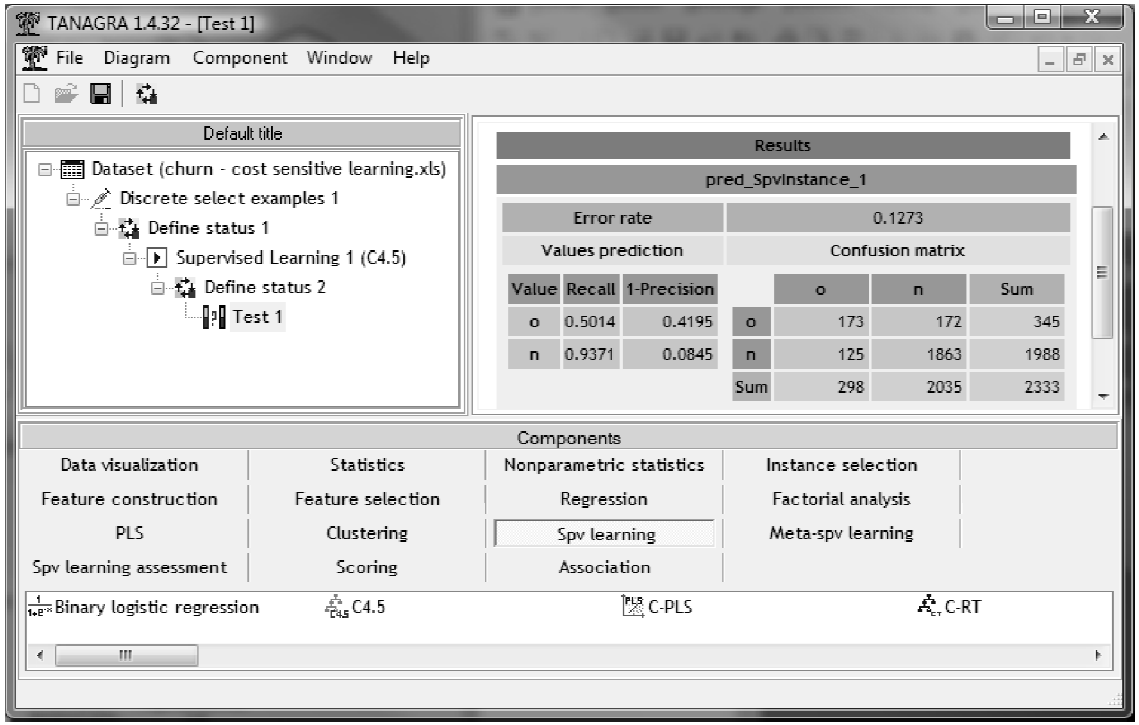
$$\hat{Y} = n$$

On prédit que le client ne fera pas défection

Stratégie 1 : Exemple des « CHURN »

1000 observations en apprentissage

2333 observations en test



Matrice de coûts de mauvais classement

Y \ $\hat{Y}$	+	−
+	−15	10
−	2	0

$$C(M_1) = \frac{1}{2333}(-15\times 173 + 10\times 172 + 2\times 125 + 0\times 1863) = -0.2679$$

C’est le résultat de référence c.-à-d. en intégrant les coûts d’une manière ou d’une autre dans la stratégie d’apprentissage, on doit pouvoir faire mieux

Stratégie 2 : corriger uniquement la règle d'affectation

Stratégie 2 : corriger la règle d'affectation

Stratégie 2 :

- Ne pas tenir compte des coûts lors de la construction du classifieur

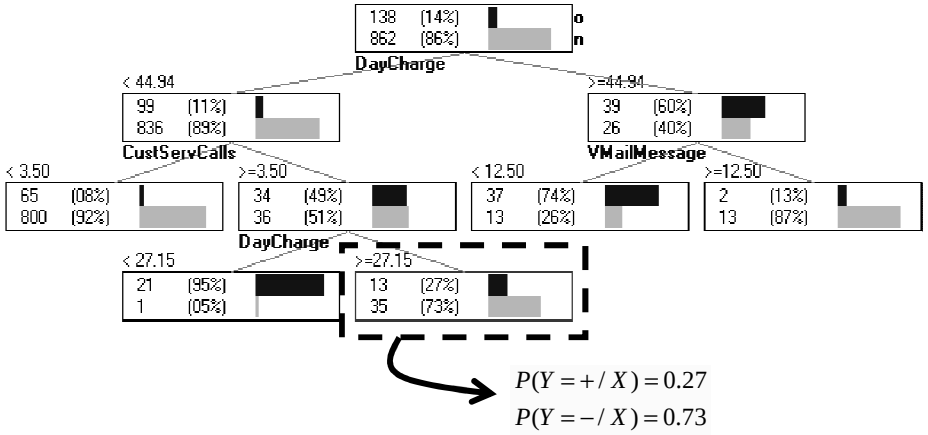
C.-à-d. lors de l'estimation des probabilités P(Y/X) à partir des données

- Tenir compte des coûts lors de la prédiction de la classe d'appartenance

Règle d'affectation : Minimisation du coût de mauvaise affectation

La règle d'affectation utilisée devient

$$y_{k*} = \arg\max_k C(y_k / X) = \arg\max_k \left[\sum_i P(Y = y_i / X) \times c_{ik} \right]$$



Matrice de coût de mauvais classement

Y \ Ŷ	+	−
+	−15	10
−	2	0

Coût moyen de prédiction : Y = +

$$C(+ / X) = -15 \times 0.27 + 2 \times 0.73 = -2.59$$

Coût moyen de prédiction : Y = -

$$C(- / X) = 10 \times 0.27 + 0 \times 0.73 = 2.7$$

La prédiction « la moins coûteuse » est Y = +

Pourtant, ce n'est pas la modalité la plus fréquente !

Stratégie 2 : quelques remarques

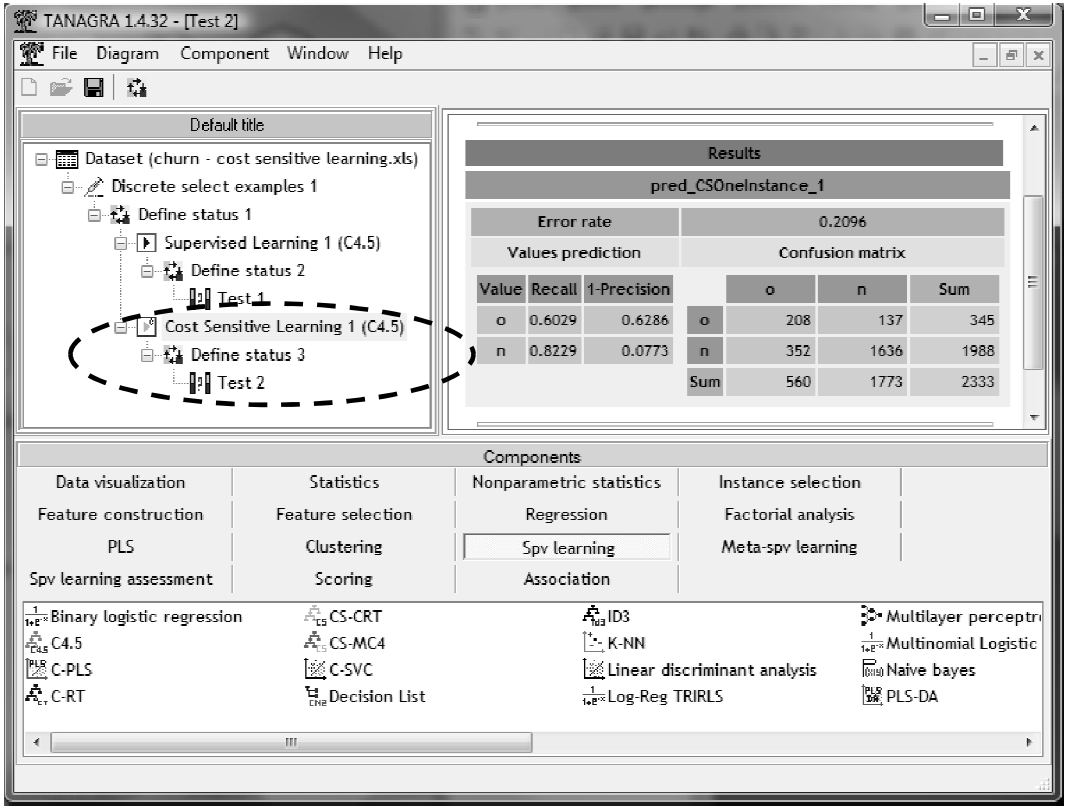
(1) Cette stratégie est adaptable à toute méthode d'apprentissage (régression logistique, analyse discriminante, etc.) pourvu qu'elle sache fournir une estimation viable de P(Y/X)

Exercice : Utiliser la régression logistique et voir ce qu'il en est

(2) Lorsque la matrice de coût est la matrice identité, cette stratégie revient à minimiser l'erreur de mauvais classement : c'est une « vraie » généralisation

Exercice : Appliquer la règle d'affectation avec un matrice identité à l'exemple précédent

Stratégie 2 : L'exemple « CHURN »



Il s'agit exactement du même arbre que précédemment. Seule la règle d'affectation sur les feuilles a été modifiée en tenant compte de la matrice de coût.

Y \ $\hat{Y}$	+	−
+	− 15	10
−	2	0

→

$$C(M_2) = \frac{1}{2333}(-15\times 208 + 10\times 137 + 2\times 352 + 0\times 1636)$$
$$= -0.4483$$

L'amélioration est spectaculaire,

Sans que l'on ait à modifier le classifieur !!!

Stratégie 2 : L'exemple « CHURN » - Comparer les matrices de confusion

Stratégie 1

0.1273			
Confusion matrix			
	o	n	Sum
o	173	172	345
n	125	1863	1988
Sum	298	2035	2333

Matrice de coûts

Y \ $\hat{Y}$	+	−
+	− 15	10
−	2	0

Stratégie 2

0.2096			
Confusion matrix			
	o	n	Sum
o	208	137	345
n	352	1636	1988
Sum	560	1773	2333

- Le taux d'erreur est moins bon pour M2, c'est normal… ce n'est plus le critère que l'on minimise

- M2 met plus d'individus là où c'est le plus avantageux : les VRAIS POSITIFS (208 vs. 173)
- Quitte à avoir plus de « FAUX POSITIFS » (352 vs. 125) qui sont comparativement moins pénalisants (*comparer les valeurs absolues de coûts*)
- Puisque l'on a plus de VP, on obtient *mécaniquement* moins de FAUX NEGATIFS (137 vs. 172) (on y gagne également au final puisque les « faux négatifs » sont coûteux +10)

Stratégie 3 : intégrer les coûts dans le processus d'apprentissage

Stratégie 3 :

• Utiliser explicitement la matrice de coûts lors de la construction du classifieur

C.-à-d. lors de l'estimation des probabilités P(Y/X) à partir des données

• Et bien sûr, tenir compte des coûts lors de la prédiction de la classe d'appartenance

Règle d'affectation : Minimisation du coût de mauvaise affectation

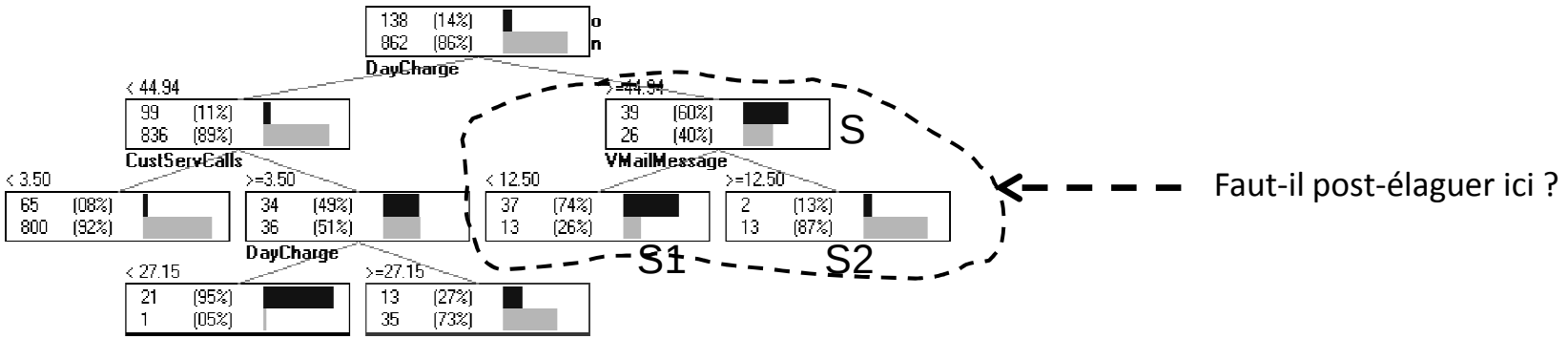
Principale difficulté : peu de méthodes s'y prêtent (de manière simple)

Parmi les techniques qui peuvent introduire les coûts dans le processus d'apprentissage : les arbres de décision (*et encore, uniquement lors du processus de post-élagage*)

Stratégie 3 : Les arbres de décision sensibles aux coûts (CS-MC4 ou CART par ex.)

Expansion de l'arbre : classique avec les mesures usuelles (entropie, KHI-2, etc.)

Post-élagage : utiliser une stratégie sensible aux coûts



Stratégie d'élagage : comparer le coût de mauvais classement sur le sommet père S avec la moyenne pondérée des coûts sur les sommets enfants S1 et S2.

L'idée est très proche de celle de C4.5 sauf qu'au lieu de travailler sur les erreurs, on travaille sur les coûts de mauvais classement. Il faut aussi pénaliser les feuilles avec de petits effectifs.

(1) Estimation des probabilités sur un nœud : pénaliser les « petits » effectifs en utilisant une estimation laplacienne (*en anglais : m-probability estimate, laplacian estimate*)

$$P(Y = y_k / S) = \frac{n_{ks} + \lambda}{n_s + \lambda \times K}$$

Plus λ est grand, plus le « lissage » est fort.

Généralement, on s'en tient à λ = 1

(2) Calculer le coût de mauvais classement associé à un sommet

$$C(S) = \min_k C(y_k / S)$$
$$= \min_k \left[\sum_i P(Y = y_i / S) \times c_{ik} \right]$$

Pour le sommet S :

(a) Calculer le coût associé à chacune des conclusions possibles

(b) Choisir la conclusion qui minimise le coût

(c) Le coût associé au sommet est égal à ce coût

(3) Élaguer si le sommet père est moins coûteux que les sommets enfants

Élaguer le sommet :

(a) si les conclusions sur les sommets enfants sont identiques à ceux du père

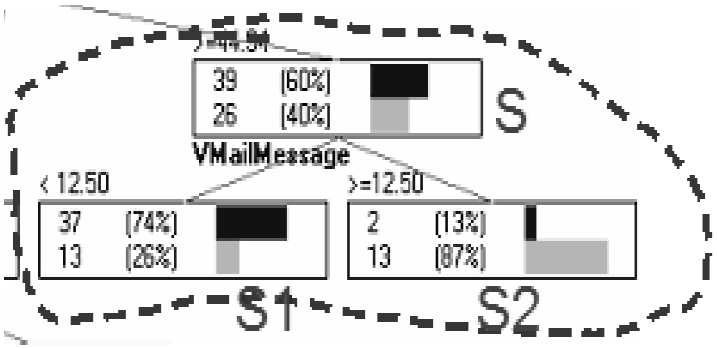
OU

(b) si $C(S) \leq \frac{n_{s_1}}{n_s} C(S_1) + \frac{n_{s_2}}{n_s} C(S_2)$

Stratégie 3 : CS-MC4 – Un exemple d'élagage de sommet

Matrice de coûts

Y \ Ŷ	+	−
+	− 15	10
−	2	0



Sommet S

$$C(+/S) = \frac{39+1}{65+2} \times (-15) + \frac{26+1}{65+2} \times (2) = -8.15$$
$$C(-/S) = \frac{39+1}{65+2} \times (10) + \frac{26+1}{65+2} \times (0) = 5.97$$

➔ $\hat{Y} = +$, $C(S) = -8.15$

Sommet S1

$$C(+/S1) = \frac{37+1}{50+2} \times (-15) + \frac{13+1}{50+2} \times (2) = -10.42$$
$$C(-/S1) = \frac{37+1}{50+2} \times (10) + \frac{13+1}{50+2} \times (0) = 7.31$$

➔ $\hat{Y} = +$, $C(S1) = -10.42$

Sommet S2

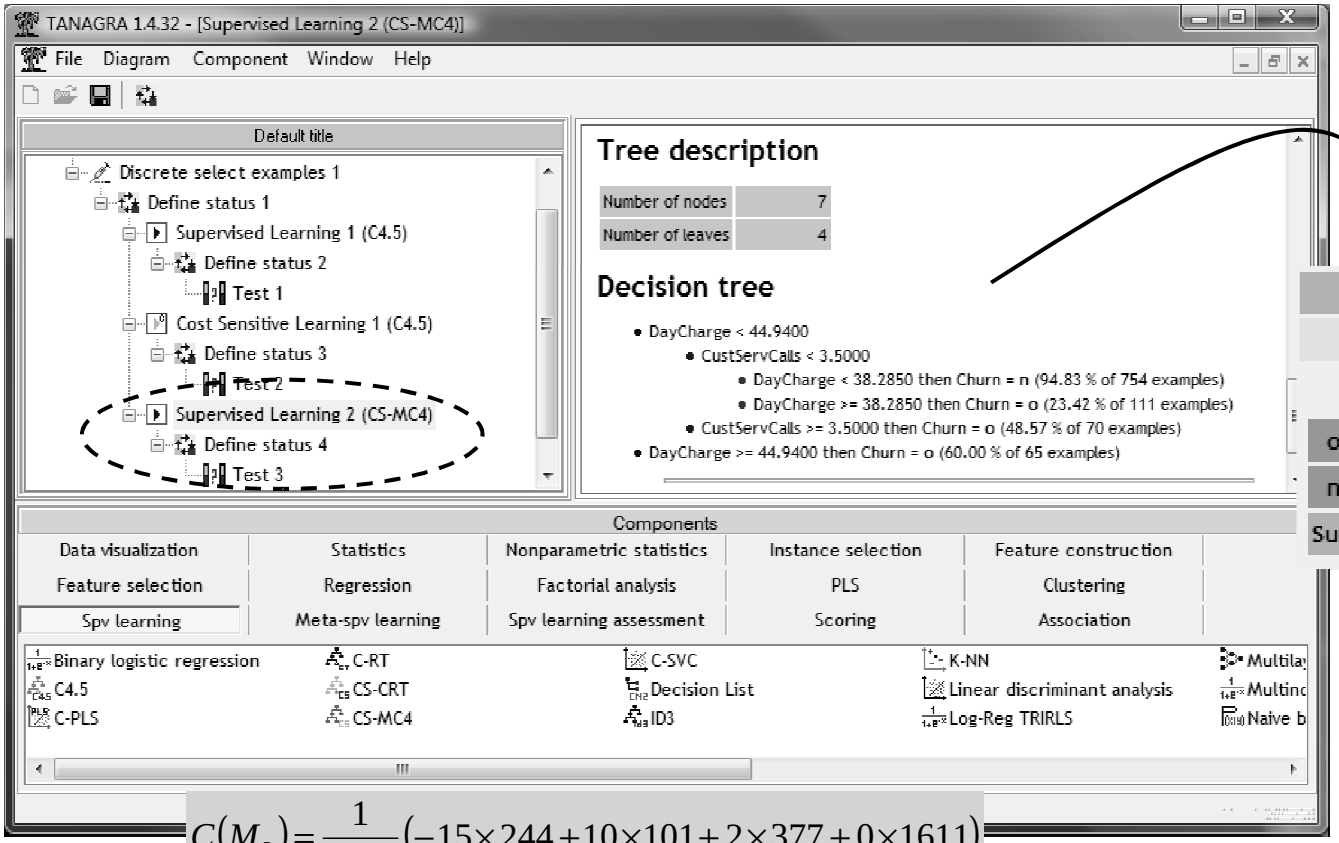
$$C(+/S2) = \frac{2+1}{15+2} \times (-15) + \frac{13+1}{15+2} \times (2) = -1.0$$
$$C(-/S2) = \frac{2+1}{15+2} \times (10) + \frac{13+1}{15+2} \times (0) = 1.76$$

➔ $\hat{Y} = +$, $C(S2) = -1.0$

Ici, on élague parce que (a) les conclusions sur les sommets enfants sont identiques à ceux du père.

(b) La règle basée sur les coûts n'était pas opérante en revanche C(S)=-8.15 vs. 50/65 x C(S1) + 15/65 x C(S2)=-8.25

Stratégie 3 : CS-MC4 – L'exemple « CHURN »



0.2049

Confusion matrix

	o	n	Sum
o	244	101	345
n	377	1611	1988
Sum	621	1712	2333

$$C(M_3) = \frac{1}{2333}(-15\times 244 + 10\times 101 + 2\times 377 + 0\times 1611) = -0.8127$$

On améliore fortement les résultats !

L'amélioration repose toujours sur une augmentation des VP = 244

On remarquera que le taux d'erreur est altéré par rapport à M1, mais ce n'est pas le problème encore une fois.

Rappel

$$C(M_1) = -0.2679$$
$$C(M_2) = -0.4483$$
$$C(M_3) = -0.8127$$

Autres stratégies

Autres stratégies : Agrégation de classifieurs (Bagging)

Apprentissage (P : nombre de classifieurs)

Pour p = 1 à P

  Tirer n individus parmi n avec remise

  Construire le classifieur Mp

Fin Pour

Classement d'un individu ω

Pour p = 1 à P

  Effectuer la prédiction avec Mp → $\hat{Y}_p(\omega)$

Fin Pour

Selon les proportions observées sur les P prédictions , nous avons une estimation de $P[Y = y_k / X(\omega)]$

Réaliser la prédiction qui minimise le coût en tenant compte de la matrice de coût de mauvais classement.

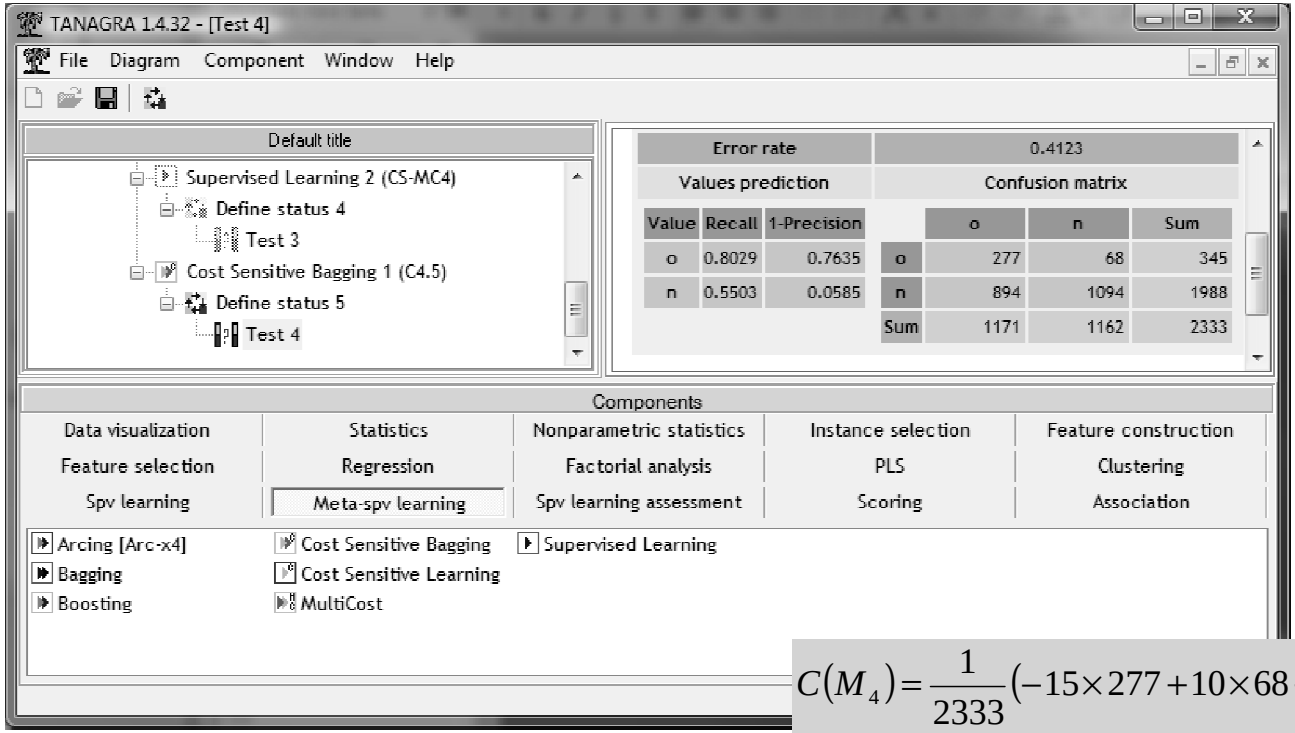
Avantages

- Le classifieur agrégé souvent meilleur que les classifieurs individuels
- Le classifieur agrégé est d'autant plus performant que les classifieurs individuels ne se trompent pas sur les mêmes individus
- L'approche est générique, elle est applicable quelle que soit la méthode d'apprentissage
- Elle fonctionne même si les classifieurs individuels ne savent pas fournir une estimation correcte de P(Y/X)

Inconvénients

- Si P est grand, le calcul peut être prohibitif
- Le mécanisme d'attribution des classes n'est pas « lisible » (identification des variables pertinentes)

Autres stratégies : Bagging sur l'exemple des « CHURN »



$$C(M_4) = \frac{1}{2333}(-15\times 277 + 10\times 68 + 2\times 894 + 0\times 1094)$$
$$= -0.7231$$

<u>Remarque :</u> Variante -- On peut introduire [Tanagra] (ou pas [Weka]) les coûts de mauvaise affectation lors des prédictions des modèles individuels Mp

Autres stratégies : MetaCost

Idée : Exploiter les performances du Bagging, mais ne produire qu'un seul classifieur finalement (donc « lisible ») – Basé sur un mécanisme de ré-étiquettage des individus

Apprentissage (P : nombre de classifieurs)

(1) Construire un ensemble de classifieurs avec le processus BAGGING
(2) Classer (prédiction) chaque individu de la base d'apprentissage
(3) Utiliser cette prédiction comme nouvelle variable à prédire dans un processus classique d'apprentissage → on obtient le modèle de prédiction définitif

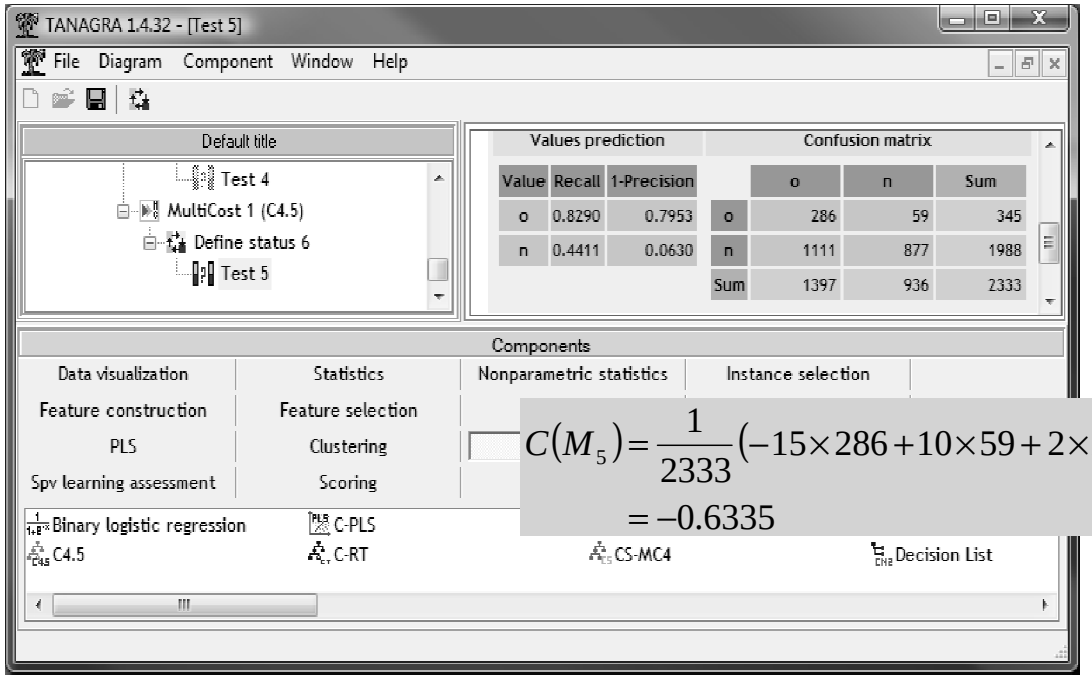
Avantages

- ***Un seul classifieur obtenu au final, donc possibilités d'interprétation***
- L'approche est générique, elle est applicable quelle que soit la méthode d'apprentissage

Inconvénients

- ***Mais rien ne garantit que l'on conservera le même niveau de performances que celui du classifieur agrégé du Bagging***
- Si P est grand, le calcul peut être prohibitif

Autres stratégies : MetaCost sur l'exemple des « CHURN »



$$C(M_5) = \frac{1}{2333}(-15\times 286 + 10\times 59 + 2\times 1111 + 0\times 877) = -0.6335$$

A titre de curiosité : croisement entre les étiquettes originelles (observées) et les étiquettes modifiées (utilisées pour la construction du modèle final)

Cross-tab

	o	n	Sum
o	138	0	138
n	299	563	862
Sum	437	563	1000

Tous les positifs sont conservés positifs
299 négatifs ont été ré - étiquetés positifs

<u>Remarque :</u> Variante -- On peut introduire [Tanagra - MultiCost] (ou pas [Weka - MetaCost]) les coûts de mauvaise affectation lors des prédictions des modèles individuels Mp

Conclusion

Comparaison des performances sur l'exemple « CHURN »

Stratégie	Coût moyen de mauvais classement	Remarques
M1 (ne pas ternir compte du tout des coûts)	-0.2679	Situation de référence, on ne doit pas faire pire dès que l'on tient compte des coûts d'une manière ou d'une autre.
M2 (tenir compte des coûts uniquement lors de la prédiction)	-0.4483	C'est bien le modèle M1 qui est construit à partir des données. Très facile à mettre en œuvre si la méthode sait fournir P(Y/X). Seule la règle d'affectation change. Et on améliore fortement.
M3 (tenir compte explicitement des coûts lors de la construction du modèle)	-0.8127	La situation idéale. Mais difficilement applicable. Toutes les méthodes ne s'y prêtent pas. Les arbres sont un cas à part dans la panoplie des techniques d'apprentissage.
M4 (Bagging)	-0.7231	Générique et performant. Mais le modèle agrégé est opaque. On ne perçoit pas dans le processus d'affectation la liaison entre Y et les X.
M5 (MetaCost)	-0.6335	Tente de tirer parti du Bagging tout en fournissant un modèle unique interprétable pour le classement. Les performances reflète cette position intermédiaire. Il est applicable quelle que soit la méthode d'apprentissage

Remarque : d'autres approches basées sur la pondération des individus existent également…

Bibliographie

Articles en ligne

Il n'y a pas beaucoup d'articles « grand public », simples et tournés vers les applications.

Faire « cost sensitive learning » dans un moteur de recherche…

Didacticiels (avec références bibliographiques)

TANAGRA, « Coûts de mauvais classement en apprentissage supervisé ».

Montre la mise en œuvre des techniques dans Tanagra, R (code source fourni) et Weka.

http://tutoriels-data-mining.blogspot.com/2009/01/cots-de-mauvais-classement-en.html

SIPINA, « Apprentissage – test avec SIPINA »

Une partie est dédiée spécifiquement à la prise en compte des coûts dans les arbres de décision

http://sipina.over-blog.fr/article-17592319.html