Application d'une approche ordinale à la Feature Selection

Introduction

Problème de Feature Selection

En apprentissage automatique, la problématique de Feature Selection est cruciale car elle interviens dans toutes les phases du développement de la conception au déploitement, cette problématique consiste à choisir parmi les attributs décrivant les données un sous ensemble d'attributs qui soit le plus apte à les caractériser et qui soit de préférence le plus réduit possible.

La Feature Selection interviens pour plusieurs raisons, limiter le nombre de dimension en éliminant les redondances permet avant tout de faciliter le probléme et donc d'aboutir à des modèles plus efficaces qui s'entrainent plus rapidement, de plus, avoir des attributs non consistants genère du bruit dans les données ce qui expose à un risque de sur-apprentissage, enfin, un modèle avec moins d'attributs est plus facilement explicable à un utilisateur externe et il est moins couteux cognitivement d'envisager son fonctionnement.

La Feature Selection est largement étudiée dans le cadre de la classification et les méthodes utilisées se répartissent en deux types, les Filter et les Wrapper methods.

Les Filter methodes s'appuient sur des mesures issue de la statistique ou de la théorie de l'information pour évaluer la qualité des attributs sans avoir à entrainer et à évaluer le modèle décisionnel, elles ont pour avantage d'être rapides à éxécuter car elles sont souvent utilisées au sein d'un algorithme, c'est à dire en classant tous les attributs par ordre décroissant de qualité puis en construisant de manière gloutonne le meilleure sous-ensemble d'attributs, mais elles ne prennent généralement pas en compte les synérgies entre les attributs.

Les Wrapper methodes quant à elle explorent itérativement l'ensemble des sous-ensembles d'attributs et évalue chaque sous-ensemble en entrainant et en évaluant le modèle de classification sur une base de données restreinte à ce sous-ensemble d'attributs, le mode d'exploration utilise généralement des mécanismes d'intensification et de diversification issues des approches évolutionnaires. Ces méthodes sont couteuses en temps de calcul car il faut entrainer et évaluer le modèle de classification pour chaque combinaison d'attributs.

Probléme de Subset Choice

Sur le plan combinatoire, le probléme de Feature Selection appartient au diaspora des problémes de Subset Choice, ce sont des problémes NP-Complet qui sont très répandus et qui peuvent être très complexes, dans ces problémes on suppose en général qu'un opérateur à une préférence entre chaque couple de sous-ensembles d'alternatives x_1, x_2 inclus dans l'ensemble A des alternatives qu'on notera $x_1 \succeq x_2$ et qui signifie que x_1 est au moins aussi bien que x_2 et l'objectif est de déterminer le meilleur sous-ensemble au sens de cette relation en explorant l'ensemble des sous-ensembles non dominés défini par:

$$ND_{\succeq} = \{x \in 2^A; \forall y \in 2^A (y \succeq x) \rightarrow (x \succeq y)\}$$

Quand cette préférence n'est pas totalement connue, il faut recourir à un processus d'élicitation. Cette elicitation se fait généralement sur une fonction cardinale et consiste à poser une fonction paramétrée f_θ sensée représenter la préférence du dit utilisateur et définie par:

$$\forall x, y \in 2^A; x \succeq y \rightarrow f_\theta(x) \geq f_\theta(y)$$

Ensuite, en interagissant avec l'utilisateur, par exemple en lui demandant de comparer des alternatives, on introduit des contraintes qui représentent ses préférences

$$x_1 \succeq x_2 \rightarrow f_\theta(x_1) \geq f_\theta(x_2)$$

L'ensemble des ces contraintes sera noté R et son extension va restreindre progressivement l'espace des paramètres θ_R compatibles avec les préférences et donc celui des solution optimale pour au moins un jeu de paramètres $\theta \in \theta_R$ qu'on désignera par la suite par **solution potentiellement optimale**.

L'objectif est d'aboutir rapidement et au prix d'un effort cognitif réduit pour l'opérateur à une solution **nécessairement optimale** c'est à dire une solution qui soit optimale quelque soit $\theta \in \theta_R$.

Modélisation du probléme de Feature Selection

Si on analyse le probléme de selection d'attributs par le prisme de la théorie de la décision on peut le modéliser comme un problème où chaque sous-ensemble d'attributs est associé à un vecteur décrivant ses performances (taille, précision de classification, ... etc) que l'on peut obtenir au prix d'une interaction avec le modèle de classification, cette interaction consiste à construire et à entrainer le modèle puis à évaluer ses performances, afin de simplifier la suite nous imaginons que la performance d'un modèle est uniquement déterminée par sa précision moyenne sur une 10-fold validation.

Donc notre modèle de classification peut être représenter par un oracle C

$$C : \begin{vmatrix} \mathbf{2^A} & \longrightarrow & \mathbf{R}\times\mathbf{N} \\ v & \longmapsto & (p_v, |v|) \end{vmatrix}$$

Ou p_v represente la précision obtenue en contrsuisant un modèle de classification prennant en paramètres les attributs du sous ensemble v, et $|v|$ la taille de ce sous ensemble

La relation $\succeq$ que l'on définira pour le problème de Feature Selection pourra contenir les préférences de l'opérateur qui effectue la Feature Selection, mais en définissantt:

$$x \succeq_C y \equiv (p_x \geq p_y) \wedge (|x| \leq |y|)$$

ou plus généralement pour un vecteur quelconque de mesures de performances $x \succeq_C y$ si $C(x)$ domine $C(y)$ au sens de Pareto on à naturellement que $x \succeq_C y \rightarrow x \succ y$

Si on reviens sur les deux types de méthodes de Feature Selection et qu'on les analyse par le prisme de la théorie de la décision, on s'apperçoit que les méthdoes de ranking font une hypothèse sans la nommer qui est celle que les mesures employées sont des mesures totalement additive qui représentent la relation de préférences $\succeq_C$, outre le fait qu'il est impossible de prouver que ce soit tout le temps le cas, cette méthode ne prend pas en compte le fait que les sous-ensembles ont des performances qui varient en fonction du modèle de classification utilisé et ne prend pas non plus en compte les synérgies pouvant exister entre les sous-ensembles d'attributs du fait de la redondance et de la complémentarité pouvant exister entre eux.

Les Wrapper methodes en revanche s'appuient sur une exploration de l'ensemble des sous-ensemble d'attributs en utilisant des appels à l'oracle qui servent à évaluer différents sous-ensembles ce qui particularise la selection au modèle et qui permet de prendre en compte les interactions, cependant le nombre d'appels est souvent très grand et contrairement aux rankings methodes il n'y a pas de fonction d'utilité qui guide la recherche.

Le but de ce travail est d'abstraire à ces deux classes de méthodes en utilisant des notions issues de la théorie de la décision, pour se faire on s'inspirera d'une ranking methode à la différence près qu'au lieu d'utiliser une mesure dont on suposera qu'elle est additive on suppose l'existence d'une fonction d'utilité additive f_θ (2-additive par exemple) et on restreint progressivement l'ensemble θ en mimant le fonctionnement d'une Wrapper methodes, c'est à dire, en évaluant des sous-ensembles d'attributs potentiellement optimaux.

Notations

- $A = \{a_1, a_2, \ldots, a_n\}$ Ensemble d'attributs.
- $v \in 2^A$: Ensemble des sous-ensembles d'attributs.
- C: Fonction qui associe à chaque sous-ensemble d'attributs une vecteur de performances, ici p_v représente la précision du modèle entrainé en restreingnant les attributs des données à v et $|v|$ le nombre d'attributs.

$$C : \left| \begin{array}{ccc} \mathbf{2^A} & \longrightarrow & \mathbf{R}\times\mathbf{N} \\ v & \longmapsto & (p_v, |v|) \end{array} \right.$$

- $x \succeq_C y$: Relation ordinale définie par: $x \succeq_C y$ si et seulement $C(x)$ domine $C(y)$ au sens de pareto.
- $x \succeq y$ Relation ordinale qui modélise les préférences de l'opérateur qui effectue la Feature Selection, elle satisfait naturellement $x \succeq_c y \rightarrow x \succ y$.
- f_θ Fonction représentant les préférences de l'opérateur paramétrée par le vecteur de paramètres θ, par exemple pour $\theta = \{u_1, u_2\}$ $u(\{1,2\}) = u_1 + u_2$, et pour $\theta = \{u_1, u_2, u_3, u_4, u_{1,2}, u_{1,4}\}$; $u(\{1,2,3\}) = u_1 + u_2 + u_3 + u_{1,2}$.
- R Ensembles de préférences formulées par l'utilisateur de la forme $x_i \succeq x_j$.
- θ_R Ensembles de paramètres tel que $\forall \theta \in \theta_R; \forall (x_i, x_j) \in R; f_\theta(x_i) \geq f_\theta(x_j)$.
- S_{-x} Ensemble des sous-ensembles ne contenant pas x.

Contribution

Afin d'expliquer notre approche nous allons d'abord expliciter un framework permettant de décrire n'importe quel approche de Feature Selection, ce framework se compose de 3 parties:

- **Un mode d'exploration** : Par exemple les Wrapper méthodes explorent en utilisant une métaheuristique, les ranking méthodes n'évaluent que les singletons et il existe des approches exhaustives qui enumérent tous les sous-ensembles existants.
- **Une fonction d'évaluation** : Dans les méthodes de ranking c'est une mesure statistique de la qualité des attributs et pour les Wrapper methodes c'est la précision d'un modèle entrainé avec le sous-ensemble d'attribut qu'on veut évaluer.
- **Condition d'arrêt**: Pour les ranking methodes ça peut être d'atteindre un certain seuil dans le nombre d'attributs ou de sélectionner tous les attributs au dela d'un seuil sur la fonction d'évaluation, pour les Wrapper methodes c'est généralement un nombre d'itérations.

A présent on va se servir de ce framework pour décrire la méthode proposée.

Notre méthode de Feature Selection va se baser sur une hypothèse simplificatrice consistant à supposer que la relation $\succeq$ pour laquelle on cherche à explorer l'ensemble $ND_{\succeq}$ des solutions efficaces est représentable par une fonction f_θ.

On note S_C l'ensemble courant des sous-ensembles d'attributs déja évalués par l'oracle, et R l'ensemble courant des préférences de l'opérateur, qu'elles aient été explicitément formulé ou déduite par la relation $\succeq_C$.

Mode d'exploration

Pour tirer des sous ensembles v à évaluer on peut utiliser deux types d'approches.

Tirage par indice de pouvoir

Soit S_{-x} l'ensemble de tous les sous ensemble qui ne contiennent pas x, cette approche consiste à définir un indice de pouvoir $p(i); \forall i \in A$, cet indice de pouvoir est sensé mesurer l'impacte de l'attribut i sur les différents sous-ensembles d'attributs auquel il peut appartenir, ensuite il suffit de décider de la présence de chaque attribut en effectuant un jet aléatoire qui suit la loi:

$$P(i \in A) = \frac{p(i)}{\sum_{i \in A} p(i)}$$

Pour définir cet indice de pouvoir plusieurs méthodes peuvent être envisagées, on peut par exemple penser à une mesure qui quantifie l'écart minimum entre S et $S \cup i$ pour chaque S ne contenant pas i comme suit:

$$p(i) = \sum_{v \in S_{-i}} [min_\theta(f_\theta(v \cup i)) - max_\theta(f_\theta(v))]$$

Ou alors en calculant au préalable un $\theta_k \in \theta_R$ particulier, ce θ peut par exemple être celui qui maximise l'écart entre les éléments qui sont liés par une préférence stricte, ou alors prendre $\theta_k = min_{\theta \in \theta_R}(|\theta|)$ puis en définissant $\forall i \in A$:

$$p(i) = \sum_{v \in S_{-i}} [f_{\theta_k}(v \cup i) - f_{\theta_k}(v)]$$

Tirage séquentiel

Cette fois on envisages le tirage des attributs comme une opération séquentielle soit $P(x|S)$ la probabilité d'ajouter $x \notin v$ à v, on pose

$$P(x|v) = \frac{min_\theta f_\theta(v \cup x)}{max_\theta f_\theta(v)}$$

Fonction d'évaluation

Nous définissons comme condition à l'évaluation d'un sous-ensemble $v \in 2^A$ qu'il soit potentiellement optimal, chose que l'on peut aisément déterminer en verifiant que $\{\theta \in \theta_R; \forall x \in S_C; f_\theta(v) \leq f_\theta(x)\}$ est un ensemble vide, cet ensemble n'est pas vide alors on dis que l'ensemble d'attributs v est rejetée.

Ensuite, si un sous-ensemble v est potentiellement optimal nous pouvons l'évaluer en repètant le processus suivant n fois.

- On construit le modèle de classification et on l'entraine sur une proportion de données égale à $\frac{(n-1)}{n}$ en restreignant la description de chaque donnée aux attributs présents dans v.

- On évalue la précision du modèle obtenue sur le $\frac{1}{n}$ des données sur lesquels le modèle ne s'est pas entrainé.

On calcule ensuite la précision moyenne obtenue avec le vecteur p_v et on en déduit le vecteur de performance $(p_v, |v|)$

Une fois le sous-ensemble évalué on l'ajoute à S_C et on parcours S_C et on met à jour R en exploitant le fait que $x \succeq_C y \rightarrow x \succeq y$.

Condition d'arrêt

Il est rare de pouvoir décrire précisement une condition d'arrêt non empirique dans un algorithme de Feature Selection, cependant, aboutir à un sous-ensemble nécéssairement optimal est dans notre cas une condition d'arrêt évidente bien qu'il soit peu probable qu'elle se vérifie.

Nous pouvons également fixer une condition d'arrêt sur les sous-ensembles rejetés et considérer que si on rejette un nombre conséquent de sous-ensembles dans la phase d'évaluation (car potentiellement dominés) cela signifie que la fonction f_θ est suffisamment discriminante pour éliminer une grande partie des solutions mais que il existe plusieurs solutions potentiellement optimales et incomparables entre elles (car de taille différentes par exemples) et qui nécessitent par conséquent un arbitrage de l'opérateur.