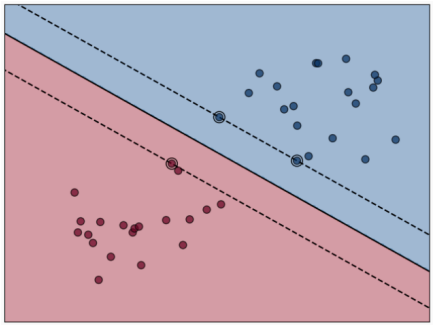
Travail d'étude et de recherche sur les SVM : Support vector Machine

Mulapi Tita Ketsia
contact@ketsiamulapi.com

1 Introduction

1.1 SVM : Séparateurs à Vaste Marge



Il s'agit d'un algorithme initialement conçu pour faire de la classification binaire, c'est à dire que pour un jeux de données donné, admet 2 valeurs possibles qui sont **{-1,1}**, on va alors classer les données en introduisant un hyperplan séparateur qu'on appelle **la frontière de décision**, qui n'est rien d'autre que **une droite qui se place au centre 0 de la marge géométrique** qui doit être la plus grande distance entre les points les plus prochent des 2 classes [5]. L'hyperplan solution est matérialisé par la formule suivante :

$$h(x) = W^T X + b \quad (7)$$

- h : la fonction de décision qui permet de prédire les $\hat{y}$ des nouvelles données, si $h(x) \leq 0$ alors $\hat{y} = -1$ sinon, $\hat{y} = 1$;
- W : le vecteur des paramètres obtenus après entraînement du modèle SVM, les paramètres w_i sont appelés les coefficients des caractéristiques (variables), tout comme en mathématique, ce sont les coefficients qui pondèrent les variables;
- X : C'est l'ensemble des données de test, ce sont les nouvelles données que le modèle n'a jamais vu, les données pour lesquelles on aimerait prédire un $\hat{y}$ (une classe);
- b : C'est le biais il peut jouer plusieurs rôles, ici il sert principalement à modifier l'orientation de la droite de décision.

Il existe 2 versions de SVM, le **SVM hard margin** (*marge rigide*) utilisé lorsque les données métiers sont des données linéaires c'est à dire des données pour lesquelles une sépration linéaire parfaite est possible, il suffit alors d'une droite pour séparer les classes, afin de classer ces données. Dans la vraie vie malheureusement, les données ne sont pas toujours linéaire, on se retrouve alors face à un autre type de problème dit non linéaire, et c'est à ce stade que nous retrouvons le concepte de noyau, qui est abordé avec la version **SVM soft margin** (*marge souple*).

Mais avant, retenons que pour tout SVM, il convient de [5] :

- **maximiser la marge** (8)
- **bien classer les données** (9)

le problème devient :

$$\min_{w,\beta} \frac{1}{2} \parallel w \parallel^2 \quad (8)$$

$$S/C \left\{ y_i(w^T x_i + b) \geq 1, i = 1, ..., n \quad (9) \right.$$

C'est la forme primale du problème d'optimisation convexe d'un SVM Hard Margin, le primal en optimisation correspond à un problème qui comporte soit le même nombre, soit plus d'instance que de variables $(n \geq d)$, on peut le résoudre en utilisant des méthodes populaires comme Cramer, la représentation graphique (en R^2 et R^3), une méthode algébrique (d'addition ou de substitution) et, le simplexe. Soulignons que les n contraintes affines correspondent chacune à une instance de données.

Selon la condition de Slater (condition suffisante pour qu'une forte dualité se vérifie pour un problème d'optimisation convexe), nous pouvons représenter ce problèmes sous une forme duale, c'est à dire que nous pouvons l'adapter à des cas de figure qui comportent plus de variables que d'instances $(n < d)$.

Mais pour ce faire, nous devons impérativement déterminer l'expression des variables qui minimisent la fonction objective, en appliquant les conditions d'optimalité de Karush-Kuhn-Tucker (KKT), et plus particulièrement la stationnarité comme condition de première ordre, elle permet d'annule le gradient du lagrangien qui est la fonction qui permet d'étudier les problèmes avec contriantes dans le but de contruire les problèmes duaux. On aura aussi besoin de l'admissibilité dual qui constitue une des contraintes dans le dual.
le grangien par définition est :

$L(x, \lambda, \mu) = f(x) + \sum_{i=1}^{n} \lambda_i \phi_i(x) + \sum_{j=1}^{d} \mu \phi_j(x)$ (10)

en faisant **(8)** et **(9)** dans **(10)** on a :

$L(w, b, \alpha) = \frac{1}{2} \parallel w \parallel^2 - \sum_{i=1}^{n} \alpha_i (y_i (w^T x_i + b) - 1)$ (11)

- α : multiplicateur de lagrangien associé à chaque contraintes, ce sont des n paramètres suplémentaires au modèles qui représente l'influence des contraintes.
- La solution (unique grâce à la convexité) ($w^\star$, $b^\star$, $\alpha^\star$) est un point-selle du Lagrangien; L(w, b, $\alpha^\star$) est minimal en ($w^\star$, $b^\star$) et L($w^\star$, $b^\star$, α) est maximal en $\alpha^\star$.

Condition de premier ordre du lagrangien :

- $\frac{\partial L(x,b,\alpha)}{\partial b}$=0 $\equiv \sum_{i=1}^{n} \alpha_i y_i = 0$ (12)

- $\frac{\partial L(x,b,\alpha)}{\partial w}$=0 $\equiv \sum_{i=1}^{n} \alpha_i y_i x_i = 0$ (13)

On fait **(12)** et **(13)** dans **(11)** :

$\mathrm{L}_{D-SVM} = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j x_j^t x_i$ (14)

Ce qui implique que le dual du primal est :

$$\max_{\alpha} \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j x_j^t x_i \quad (15)$$

$$S/C \begin{cases} \alpha_i \geq 0, \forall i \quad \text{(par admissibilité duale)} \\ \sum_{i=1}^{n} \alpha_i y_i = 0 \quad \text{(par stationarité)} \end{cases} \quad (16)$$

Note Bien : les vecteurs de supports sont les observations du jeu de données, correspondant à un multiplicateur de lagrangien non null ($\alpha_i^\star > 0$)[5].

De ce fait, la fonction de décision dans le duale est :

$\mathrm{h(x)} = \sum_{i=1}^{n} \alpha_i y_i x_i^T x + b$ (17)

1.1.1 Vers les SVM soft margin (marge souple)

Puisque les données ne sont pas toujours linéairement séparable, l'utilisation d'un SVM hard margin ne garantit pas une sépation parfaite de ces données, ainsi on aura des erreurs, qu'il faudrait essayé de corriger, en modifiant, en étirant la marge à l'aide des variables d'erreurs qu'on appelle des variables ressorts.

Une variable ressort est une variable qui permet de relacher les contraintes, sauf que, on ne peut pas étirer la marge n'importe comment sinon, il risque d'y avoir sur apprentissage. Voilà pourquoi en plus de celà, il faudrait veuiller à pénaliser ou contraindre ce relachement au niveau de la fonction objective, en introduisant un terme régulateur qu'on appel hyper-paramètre, ce terme est noté : C.

la forme primale, d'un problème d'optimisation pour un SVM soft margine est devient :

$$\min_{w,\beta,\xi} \frac{1}{2} \parallel w \parallel^2 + C \sum_{i=1}^{n} \xi_i \quad (18)$$

$$S/C \begin{cases} y_i (w^T x_i + b) \geq 1 - \xi, \forall i \\ \xi_i \geq 0, \forall i \end{cases} \quad (19)$$

avec $C_i > 0, \forall i$ sinon nous retournons à la verison hard margin et par conséquent, si $\xi_i = 0$ alors il n'y a pas d'erreur.

En procédant de la même façon qu'en **(14)**, on peut obtenir la forme duale, du primal la marge souple :

$L(w, b, \xi, \alpha, \mu) =$
$\frac{1}{2} \parallel w \parallel^2 + C \sum_{i=1}^{n} \xi_i - \sum_{i=1}^{n} \alpha_i (y_i (w^T x_i + b) - 1 + \xi_i)$
$- \mu \sum_{i=1}^{n} \xi_i$ (20)

Condition de premier ordre du lagrangien (stationarité) :

- $\frac{\partial L(x,b,\alpha)}{\partial b}$=0 $\equiv \sum_{i=1}^{n} \alpha_i y_i = 0$ (21)

- $\frac{\partial L(x,b,\alpha)}{\partial w}$=0 $\equiv \sum_{i=1}^{n} \alpha_i y_i x_i = 0$ (22)

- $\frac{\partial L(x,b,\alpha)}{\partial \xi}$=0 $\equiv C - \alpha_i - \mu_i$ (23)

On fait **(21, 22 et 23)** dans **(20)** :

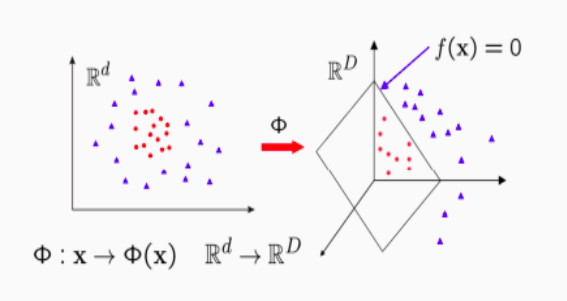
$\mathrm{L}_{D-SVM} = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j x_j^t x_i$ (24)

$$S/C \begin{cases} 0 \leq \alpha_i \leq C, \forall i \quad \text{(Amis. duale)} \\ \sum_{i=1}^{n} \alpha_i y_i = 0 \quad \text{(Stationarité)} \end{cases} \quad (25)$$

On a la même fonction de décision qu'en **(17)**.

Admettons à présent que, on ne parvient toujours pas à séparer nos données, on peut projeter le problème dans un autre referentiel ce qui revient à redefinir l'espace de description. Cette approche va consister à envoyer nos données vers un espace beaucoup plus grand et qui soit implicite tel que, on ne maîtrise pas ses spécificités mais, qui nous garantit de bien séparer nos données, c'est ce que fait la transformation du problème à l'aide d'un SVM à noyau.

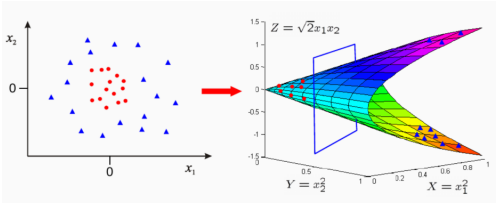
La redescription de notre problème dans un espace plus grand qu'on appel l'espace de Hilbert nous permet d'utiliser un algorithme linéaire comme le SVM soft margin afin de résoudre un problème non linéaire [10].
Il suffit dans un premier temps de remplacer tous les **x** par $\phi(x)$ tel que présenté dans l'image ci-dessous dans un soft margin[5,10]. $\phi(x)$: est la fonction de transformation. et D > d.



Dans l'espace d'Hilbert, les résultats sont contenus dans des produits scalaires, $\phi(x) : X- > H$ et pourtant, un noyau n'est rien d'autre que, un ensemble de produit scalaire contenu dans une matrice qui s'appelle la matrice de **Gram** notée **K**. Cette fois-ci, on passe de ϕ à $k < ., . >$ c'est à dire :

- $x_i^T x_j$ (les données dans l'espace de départ)
- $\phi(x_i)^T \phi(x_j)$ (les données dans l'espace de Hilbert(H) qui est très grand)
- $k(x_i, x_j) =< \phi(x_i), \phi(x_j) >=$ (l'astuce du noyau qui remplace H)

1.1.2 Le noyau



Il s'agit d'une fonction continue, symétrique et définie positive(k≻0). On note :
k : X **x** $X- > R$
NB : K est la matrice de Gram (ensemble des produits scalairs obtenus pour n observations et à l'aide d'une fonction noyau k).

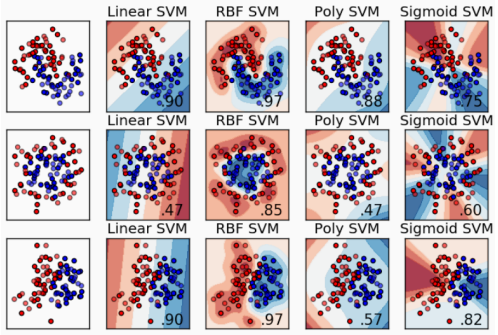
Nous osons donc dire que, le produit scalaire est au noyau et à l'espace de Hilbert ce qu'est une mesure de similarité entre deux instance de données.

cela implique qu'il est possible de définir un noyau à l'aide des similarités entre les données et, en vérifiant qu'il ait les propriétés de symétrie, semi-définie positive.

plusieurs noyau existe dans la littérature, pour notre part on s'intérresse à deux (2) d'entre eux :

1. Noyau linéaire : $k(x, z) = x^T z$ (26)
2. Noyau radial gaussien (rbf) : $k(x, z) = \exp\left(-\frac{\|x-z\|^2}{2\sigma}\right)$ (27)

le premier est utilisé dans un espace de description à très grande dimension alors que le second est s'utilise lorsqu'on a une connaissance à priori sur le problème[5].



1.1.3 OneClassSVM (OC-SVM): SVM à une classe

Si l'on ne s'en tient qu'à un SVM classique, c'est à dire une marge rigide ou souple, nous serons buté à un problème étant donné que ces derniers ont spécialement été conçus pour faire une classification minimale qui soit binaire.

En effet, nous voulons être en mesure de sélectionner des caractéristiques aussi bien dans des voisinages homogènes qu'hétérogènes, c'est à dire dans des voisinages ne contenant qu'une seule classe. C'est pourquoi on s'intérresse dans cette suite, à l'algorithme OneClassSVM, qui est un algorithme présent dans la famille des SVM, il est très employé en industrie, pour répondre à des besoins de détéction d'anomalies.

Contrairement aux précédents algorithmes, OC-SVM est semi-supervisé. En lui passant des y_{train}, si le problème n'est pas à une classe OC-SVM nous l'informe, et dans tous les cas, cet algorithme fontionnera si on ne lui donne que des X_{train} ou, tant qu'il ne s'agit que d'une classe lorsqu'on lui donne un jeu d'entraînement complet (x,y), d'ou l'intérêt d'une telle méthode pour notre recherche, c'est donc tout naturellement que nous y recourons.

Contrairement à un svm classique, le OC-SVM travaille avec une seule classe, il ne s'intérresse qu'à un seul coté de l'hyperplan et si la frontière de décision doit être ajusté, ce sera également en fonction d'une seule classe[12].

Concrètement, OC-SVM cherche à trouver dans un espace implicite, l'hyperplan le plus éloigné de l'origine, qui sépare bien tous les points d'apprentissages des outliers[13]. On note rhô (ρ), la distance à l'origine, et donc en partant du svm soft margin on obtient le problème d'optimisation suivant [13]:

$$\min_{w,\rho,\xi} \frac{1}{2} \| w \|^2 + C \sum_{i=1}^{n} \xi_i - \rho \quad (28)$$

$$S/C \begin{cases} w^T x_i \geq \rho - \xi, \forall i \\ \xi_i \geq 0, \forall i \end{cases} \quad (29)$$

1. la fonction de décision est :
   h(x) = sign($< w, \phi(x) > -\rho$) (30)
2. ξ représente les anomalies
3. C joue le rôle du régulateur qui contrôle le nombre d'anomalies
4. l'expression de ρ est :
   $\rho =< w, \phi(x_s) >= \sum_{i=1}^{n} \alpha_i K(x_i, x_s)$ (31)

Après recherche du lagrangien, on obtient le dual ci-dessous :

$$\min_{\alpha} \frac{1}{2} \alpha^T K \alpha \ (32)$$

$$S/C \begin{cases} e^T \alpha = 1 \\ 0 \leq \alpha_i \leq C, \forall i = 1, ..., n \end{cases} \quad (33)$$

- e = $(1, 1, ..., 1)^T$
- $K_{ij} = k(x_i, x_j)$ est la matrice de Gram

Enfin, l'ensemble des points les plus éloignés de l'origine forme une concentration, un tout qu'on appel la densité cible.

1.1.4 SVDD : Support Vector Domain Description

Il s'agit d'un algorithme inventé en 2004 par Taw Duin.

Si OneClassSVM fait de la outlier detection, la description des données par vecteur de support utilise une hypersphère placée autour des données et lorsqu'on applique l'astuce noyau, le modèle devient plus fexible pour suivre les caractéristiques des données ce qui veut dire que, l'algorithme SVDD s'intéresse non pas à la densité cible mais, à la distance au centre de l'hyper sphère.

Cette forme sphérique est dans un repère à deux (2) dimensions, dans le cas linéaire, un cercle qui remplace l'hyperplan séparateur du SVM classique ou, du OneClassSVM. La sphère est donc une limite de décision autour d'un ensemble d'objets, que l'on construit à l'aide d'un ensemble de vecteurs de supports.

SVDD va transformer les données en de nouveaux espaces de caractéristiques et, en utilisant les données transformées, et c'est comme ça que SVDD parvient à obtenir des descriptions de données plus flexibles et plus précises[12,15].

le principe [13]:

Une fois dans l'espace d'hilbert, on cherche l'hypersphère la plus petite qui englobe les données.

le problème d'optimisation de l'algorithme SVDD est le suivant :

$$\min_{R,g} R^2 + C \sum_{i=1}^{n} \xi_i \ (34)$$

$$S/C \begin{cases} \|x_i\text{-}g\|^2 \leq R^2 + \xi_i, \ i=1,...,n \\ \xi_i \geq 0, \forall i = 1, ..., n \end{cases} \quad (35)$$

- g est le centre du cercle
- R est le rayon du cercle

Tout comme pour le OC-SVM, pn recherche le lagrangien, et on obtient le dual du primal du SVDD qui est [13] :

$$\min_{\alpha} \frac{1}{2} \alpha^T K \alpha - \frac{1}{2} \alpha^T diag(K) \ (36)$$

$$S/C \begin{cases} e^T \alpha = 1 \\ 0 \leq \alpha_i \leq C, \forall i = 1, ..., n \end{cases} \quad (37)$$

- g = $\sum_{i=1}^{n} \alpha_i \phi(x_j)$ (38)
- pour évaluer un SVDD, lorsque une nouvelle donnée arrive, il appartient au support si
  $\| \phi(x) - g \| \leq R^2$ (39)

(38) dans (39) :

$$K(x,x) - 2 \sum_{i=1}^{n} \alpha_i K(x_i, x) + \sum_{i=1}^{n} \alpha_i \alpha_j K(x_i, x_j) \leq R^2 \ (40)$$

c'est la fonction de décision.

Enfin retenons que, minimiser le volume de l'hypersphère revient à minimiser l'incorporation des valeurs aberrantes dans la solution et, que le centre de la sphère est une combinaison linéaire des vecteurs de support. l'algorithme SVDD est une variante de OneClassSVM [15].

2 La sélection des caractéristiques et les méthodes de sélections

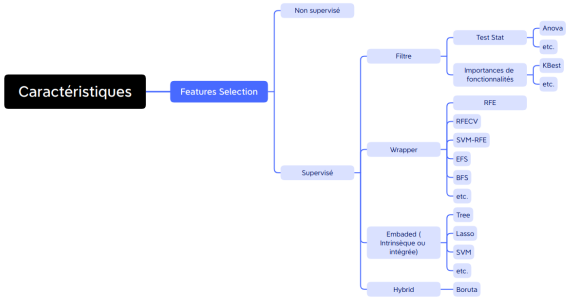
Seléctionner des caractéristiques revient à retenir les caractéristiques ayant le plus d'influence durant la phase d'apprentissage. Cette influence se traduit par la capacité que détient une caractéristique à renseigner au modèle la valeur (étiquette) à laquelle correspond une instance.

En classification, on s'intérresse à des classes, il existe des variables qui entrainent le modèle à prédire une classe lorsqu'elles sont très valuées, c'est variables se distingue donc comme étant importantes par rapport aux autres.

Dans cette étude on veut montrer qu'il peut exister pour des classes et plus spécifiquement des voisinages (matérialisé par des cluster), des caractéristiques qui soient importantes. On se retrouve avec des voisinages homogènes et hétérogènes mais on ne se limitera pas là, afin de démontrer que notre modèle fonctionne bine, on utilisera un protocole expérimentale qui nous permettra de visualiser l'effet qu'on obtient lorsqu'on retire des variables ce qui implique que si notre modèle selectionne bien alors, le retrait d'une caractéristique au sein d'un cluster va entrainer un chagement d'état et de performance aussi bien pour les données que pour les clusters eux mêmes.

En apprentissage supervisé on distingue plusieurs méthodes de sélection des caractéristique. Parmis les méthodes existantes nous citons [14] :

1. Les méthodes de type filtre (filter) : ce sont des méthodes qui ont un mécanisme simple qui consiste à filtrer les caractéristiques en se basant sur des tests statistiques et en décidant à l'aide d'une p-value. Ce genre de méthode sont souple mais moins éfficace.
2. les méthodes de type emballage (wrapper) : Ce sont des méthodes simple à mettre en oeuvre mais couteuse, elles créent en fonction d'une métrique de performance qu'on appel score d'importance, des variables ou sous ensembles de variables.
3. les méthodes de type embarqué ou intrinsèque (embadded/intrinsic): il s'agit des algorithmes qui font de la séléction automatique des caractéristiques pendant l'entraînement.
4. les méthodes de type hybride : ce sont des méthodes qui utilisent à la fois des techniques de filtrage et d'emballage. Voir le livre Sélection de caractéristiques pour la reconnaissance de données et de modèles De Urszula Stańczyk et Lakhmi C. Jain



Dans le cadre de ce travail, on s'intéressera à une méthode enveloppe et particulièrement à l'algorithme SVM-RFE pour 2 raisons :

- Elle est efficace et très recommandée dans la littérature
- Elle permet de gérer le ranking

Par conséquent, nous avons décidé qu'elle fera une bonne méthode conccurente pour notre recherche.

3 La methode SVM-RFE

3.0.1 principe SVM-RFE

La méthode SVM-RFE signifie Support Vectors Machine - Recursive Features Elimination, c'est un algorithme basé sur l'élimination backward. On filtre le vecteur des attributs sur la base des poids obtenus de façon à en extraire des caractéristiques discriminantes et pertinentes. cette méthode enveloppe a pour estimateur un SVM et, au moment du processus d'apprentissage, il apprend sur la pertinence des caractéristiques à partir de l'estimateur, c'est à dire que le SVM renvoie des coefficients(qui sont les paramètres W abordés plus haut)et, qui aide au moment du filtrage[9].

Néanmoins, la limite de ce modèle est un problème car il s'agit d'un algorithme glouton c'est à dire qu'il ne permet pas de réintégrer les caractéristiques ou sous-ensembles de caractéristiques éliminés. Le processus de sélection stop lorsqu'un critère d'arrêt est atteint, ce critère, c'est le nombre de caractéristique à retenir. Et si on veut juste renvoyer tous les coefficients soit on fait une seule itération, soit on fait de l'élimination mais dans ce cas, on devra alors garantir que on maintient la première caractéristique comme étant la moins importante sur l'ensemble sinon on risuqe d'avoir une caractéristique qui soit plus importante que la première éliminé sur l'ensemble des caractéristiques mais, moins importe que la première au moment de son itération.

On précise aussi que, en plus d'accélérer les calculs, la selection d'un groupe de caractéristiques vaut mieux que celle d'une seule caractéristique car, les selections individuelles sont complémentaires ou peu informatives, libre à chacun de s'en servire à sa guise. Par ailleurs, un tel algorithme ne s'utilise qu'avec des SVM linéaires sinon, la pertinence de X dépend de la région, autrement, Dans les cas où la surface de décision est non-linéaire, la pertinence d'un attribut peut dépendre de la région dans laquelle se trouve x, ce qui exclut l'utilisation des SVM non-linéaires à des fins de sélection d'attributs globalement pertinents[9].

Puisque nous savons à présent que, un classifieur SVM utilise un vecteur de poids noté **W**, qui ets une combinaison linéaire des observations et que la pluspart des α_j sont nuls, sauf ceux des vecteurs supports alors, cette mesure tel que définit en **(13)**, peut être directement liée à l'importance des variables dans le modèle SVM et, la quantité W^2, est un pouvoir prédictif ainsi, on éliminera successivement les variables **j**, des W_j^2 de faibles valeurs.

Toute ceci explique que les W_j^2 **sont des scores d'importances** qui représentent des composantes du vecteurs de poids **W**, qui définit l'hyperplan optimal obtenu qui est définit en **(13)** et qui devient :

$$\sum_{j=1}^{ns} \alpha_j y_j x_j = 0 \quad (41)$$

3.0.2 Algorithme SVM-RFE

Algorithm 1: Algorithme initial SVM-RFE

initialisation;
$s = \{x^{(1)}, x^{(2)}, x^{(3)}, , ...x^{(d)}\}$ #Ensemble d'app.
$d = critère$ #nombre d'attributs fixés
while $s \geq d$ **do**

1. *Apprentissage du SVM sur les x_i de s et y_i;*
2. *Calcul du vecteur de poids $W = \sum_{j=1}^{ns} \alpha_j y_j x_j = 0$;*
3. *Calcul du critère de rang $C_i = W_i$ pour l'indice i dans s;*
4. *Trouver la variable avec le plus petit critère de rang $f = f = arg\min(C_i)$;*
5. *Eliminer la variable avec le plus petit critère de rang : $s = s$;*

end
Sol = s #ensemble des solutions
Renvoyer Sol

3.1 variété OneClassSVM-RFE

En considérant les deux points précédents, on peut s'apercevoir qu'il suffit donc d'estimer un vecteur de paramètres (W) à partir du quel on calcul le score d'importance et donc, en utilisant OneClassSVM, on peut également aboutir aux même résultats en retenant le vecteur de paramètres qu'il nous renvoie. Nous allons dériver le lagrangien obtenu à l'aide de **(28)** et **(29)** pour déterminer **W**, ce qui nous donne:

$$\sum_{j=1}^{ns} \alpha_j x_j = 0 \quad (42)$$

4 Conclusion

Au bout du compte, l'objectif de notre document consiste à aborder la notion des SVM. voir son implication dans un **SVM-RFE** qui se révèle être à l'état de l'art, une méthode de sélection de caractéristiques de type enveloppe efficace. Nous avons vu que cette méthode possède des inconvénients comme le fait qu'il soit **glouton** et, qu'il existe aussi des limites pour les cas de sélection de caractéristique sur des données appartenant à une seule classe.
Cela nous a conduit à découvrir **OneclassSVM** qui est un algorithme semi-supervisé et qui travail avec une seule partie de la frontière de décision (hyperplan) afin de détecter des données abérrantes, enfin nous avons abordé sa variante **SVDD**, qui permet de résoudre le même problèmes à la seule différence que celui-ci utilise une hypersphère.

Nous avons donc jusque là procédé à l'utilisation d'un SVM classique de type **soft margin** et un OneclassSVM pour faire de la sélection de caractéristiques local à un voisinage. Comme le veut le principe du no free lunch, il faudra penser dans les futurs travaux à une combinaison **SVDD-RFE**, et voir dans quelle mesure utiliser des dataset benchmark tel que **Madelon** dans le but d'avoir une idée plus claire sur ces applications.

References

[1] machine learning tout savoir : https://datascientest.com/machine-learning-tout-savoir

[2] comment fonctionne une machine learning : https://techlib.fr/app/16457/comment-fonctionne-une-machine-learning

[3] Machine Learning : wikipedia

[4] Des données pour un apprentissage supervisé : editions eni

[5] Approche de sélection d'attributs pour la classification basée sur l'algorithme RFE-SVM . Yahya Slimani, Mohamed Amir Essegir, Mouhamadou Lamine Samb, Fodé Camara, Samba Ndiaye

[6] Introduction au MAchine Learning . Chloé-Agathe Azencott

[7] K-medoïde : wikipedia/Apprentissage automatique

[8] One Class SVM Vs SVM Classification. Divya Rana

[9] Algorithme à noyau - Cnam . Marin FERECATU Michel Crucianu

[10] comment Comment choisir une méthode de sélection de fonctionnalités pour l'apprentissage automatique : https://machinelearningmastery.com/feature-selection-with-real-and-categorical-data/ . Jason Brownlee

[11] Support Vector Data Description : DAVID M.J. TAX davidt@first.fhg.de ROBERT P.W. DUIN