Rapport TP1/UP3: Optimisation Classique

Najlaa SRIFI
Hajar EL Fahfouhi
Lea Bagheri

21 Novembre 2022

1 Introduction

Les séparateurs à vaste marge (SVM) sont un ensemble de techniques d'apprentissage supervisé destinées à résoudre des problèmes de discrimination et de régression. Les SVM sont une généralisation des classifieurs linéaires. Dans ce TP, nous allons étudier le problème de détermination de la nature d'une cellule si elle est cancéreuse ou pas. Pour ce faire, nous allons construire une fonction dite de « classification » en exploitant les techniques d'optimisation locale vues en cours (en l'occurrence, l'algorithme d'Uzawa). Nous allons explorer notamment plusieurs variations du modèle pour l'adapter et le généraliser à des distributions de cellules non séparables linéairement ou avec des outliers.
Nous allons aussi développer un outil de régression en utilisant la méthode SVR, pour approcher une fonction de R dans R puis une autre de R^2 dans R.

2 SVM

2.1 Cas linéaire :

Dans ce premier cas, on travaille avec la première base de données **data1** qui contient un ensemble de points qui sont classés selon deux groupes, un groupe marqué par le label 1 qui réfère aux cellules cancéreuses et l'autre groupe est caractérisé par le label -1 faisant référence aux cellules saines. Les deux groupes sont séparés par un hyperplan qui satisfait l'équation $f(x) = w^T x + b$, les cellules sont classés du coup dans une des deux familles selon le signe de f(x).Le reste sera d'assurer une marge maximale. Pour cette raison, on travaille avec une méthode d'optimisation locale sous contraintes sous sa forme primale qui s'écrit :

$$\begin{cases} \min \frac{1}{2}||w||^2 \\ l_j(w^T x + b) \geq 1, j = 1, ..., p \end{cases}$$

Et de lagrangien qui s'écrit :

$$L(w, b, \alpha) = \frac{1}{2}||x||^2 + \sum_{j=1}^{p} \alpha_j (1 - l_j(w^T x_j + b)$$

Cela nous amène au problème dual suivant :

$$\begin{cases} \max_{\alpha \geq 0} H(\alpha) \\ \sum_{j=1}^{p} \alpha_j l_j = 0 \\ \alpha_j \geq 0, j = 1, ..., p \end{cases}$$

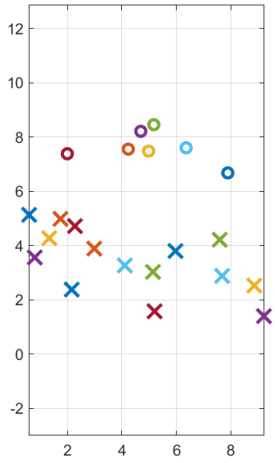
En notant A la matrice $(l_i l_j x_i^T x_j)_{i,j}$ et u le vecteur $(1, ..., 1)^T$ de R^p, on écrit :

$$H(\alpha) = -\frac{1}{2}\alpha^T A \alpha + u^T \alpha$$

En supposant que nous ayons un problème primal convexe, la fonction objectif du problème dual est naturellement concave. Cette propriété peut être prouvé par :

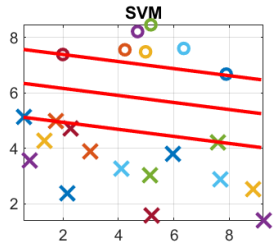
$$H''(\alpha) = \frac{\partial^2 H(\alpha)}{\partial \alpha^2} = -A$$

Avec $A = \sum_{i,k}^{p} l_k l_i x_k^T x_i = (LX)^T(LX) = ||LX||^2 \geq 0$, Puisque A est symétrique, alors A est semi-défini positif, ainsi que la fonction H est concave.
Graphiquement $H(\alpha)$ présente le minimum des fonctions affines, chose qui prouve de plus que H est concave et qui montre notamment que la fonction objective peut être décomposée avec la méthode de Cholesky et qu'elle peut être substituée par un noyau dans le cas non linéaire sans modifier l'algorithme.
Le gradient de $H(\alpha)$ s'exprime donc simplement : $\nabla H(\alpha) = -A\alpha + u$.
Dans le cas d'un problème de minimisation d'une fonction J $K - convexe$ est que ∇J est $\gamma - Lipschitzien$ sur l'espace C des contraintes. Alors, si le pas$\in]0, \frac{2k}{\gamma}[$, la méthode du gradient plus projection converge quel que soit la valeur initaile de α : le learnning rate.
l'implémentation de notre problème dual consiste à déterminer d'abord H par la méthode du gradient à pas constant. Ensuite faire une première projection sur l'hyperplan de la première condition puis une deuxième projection sur le quadrant positif. On détermine par la suite b en se servant des points support qui respectent la relation suivant : $l_k(w^T x_k + b) = 1$

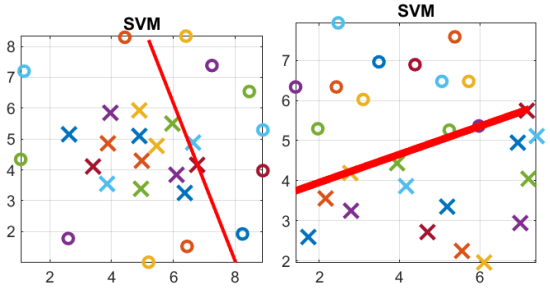


```
H_alpha = - 0.5 * alpha0'* A * alpha0 + u' * alpha0 ;
grad_H = -A * alpha0 + u;
alpha_k = alpha0;
alpha_k_plus_1 = 0;
vit_conv = zeros(1,npas_max);
while crit_arret >= epsi && npas <= npas_max
    alpha_k_plus_1 = alpha_k + pasgrad * grad_H;
    alpha_k_plus_1 = alpha_k_plus_1 - (lab' * alpha_k_plus_1) * lab / (norm(lab)^2);
    alpha_k_plus_1 = max(alpha_k_plus_1,0);
    H_alpha = -0.5 * alpha_k_plus_1' * A * alpha_k_plus_1 + u' * alpha_k_plus_1;
    grad_H = -A * alpha_k_plus_1 + u;
    crit_arret = norm(alpha_k_plus_1 - alpha_k);
    vit_conv(npas) = crit_arret;
    alpha_k = alpha_k_plus_1;
    npas = npas + 1;
end
```

On remarque que les deux classes sont linéairement bien séparées. Autrement dit les deux familles sont séparable par une droite avec une marge assez grande.



Dans le cas ou les deux familles sont non linéairement séparables, ou la marge entre les deux classes et assez fine, cette méthodes devient complètement fausse. Ceci est bien évidement approuvé quand on utilise **data2** et **data3**



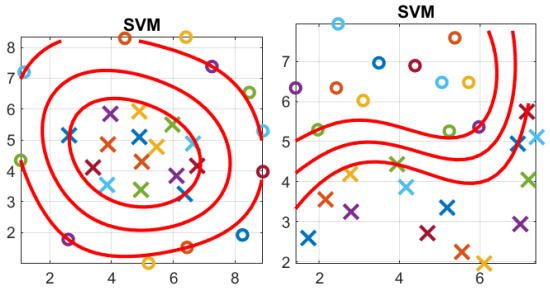
3 Cas non linéaire

Dans le cas ou les deux populations ne sont pas séparables par un hyperplan, l'idée consiste à plonger les x_i dans un espace de dimension plus grand de dimension infinie qui s'appelle espace de Hilbert dans lequel le produit

scalaire est définie à partir d'un noyau. Ce noyau est gausssien est s'écrit sous forme de :

$$K(y,z) = \exp(-\frac{||y-z||^2}{\sigma^2})$$

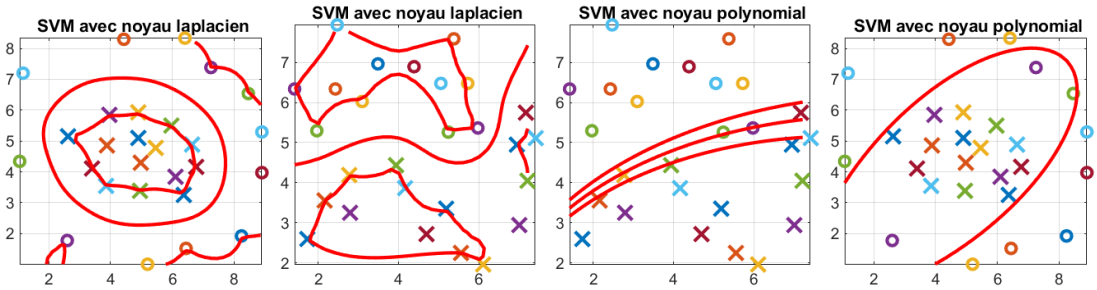
On applique cette méthode à nos bases de données : **data2** et **data3** tout en changeant ikernel de 1 à 2 puisque la fonction kernel est déjà codée et on trouve les résultats suivant :



Nous essayons d'autres noyau : noyau laplacien et polynomial qui s'écrivent respectivement comme suit :

$$K(y,z) = \exp(-\frac{||y-z||}{\sigma})$$

$$K(y,z) = (< y,z > +c)^d$$



On remarque que le noyau laplecien nous donne un résultat moin faible que celui du noyau gaussien, ainsi pour data2 la marge de SVM est grande, pour data3 le noyau entoure chaque classe toute seul mais il rate des points de la classe 2. Pourtant, pour le noyau polynomial nous donne des résultats faux dans les deux cas avec une marge assez grande. Pourtant que la variation des paramètres c et d peut améliorer plus le résultat.

3.1 Cas d'outliers

Dans d'autres cas où les données ne sont pas séparables linéairement. L'idée est de tolérer quelques erreurs, autrement dit, tolérer la présence de quelques points dans la marge ou dans la mauvaise zone, mais cela au prix d'ajouter un coût pour chaque point mal classé, qui dépendra de la distance de ce point par rapport à la marge. Nous introduisons du coup des variables d'écart positives pour chaque point : ξ_i, de telle façon que plus le variable d'écart est grande moins bien est classé le point.
Ainsi le problème primale devient :

$$\begin{cases} \min(\frac{1}{2}||w||^2 + C\sum_{i=1}^{p}\xi_i \\ \quad l_i(w^T x_i + b) \geq 1 - \xi_i \quad \text{avec} \xi_i; i = 0....p \end{cases}$$

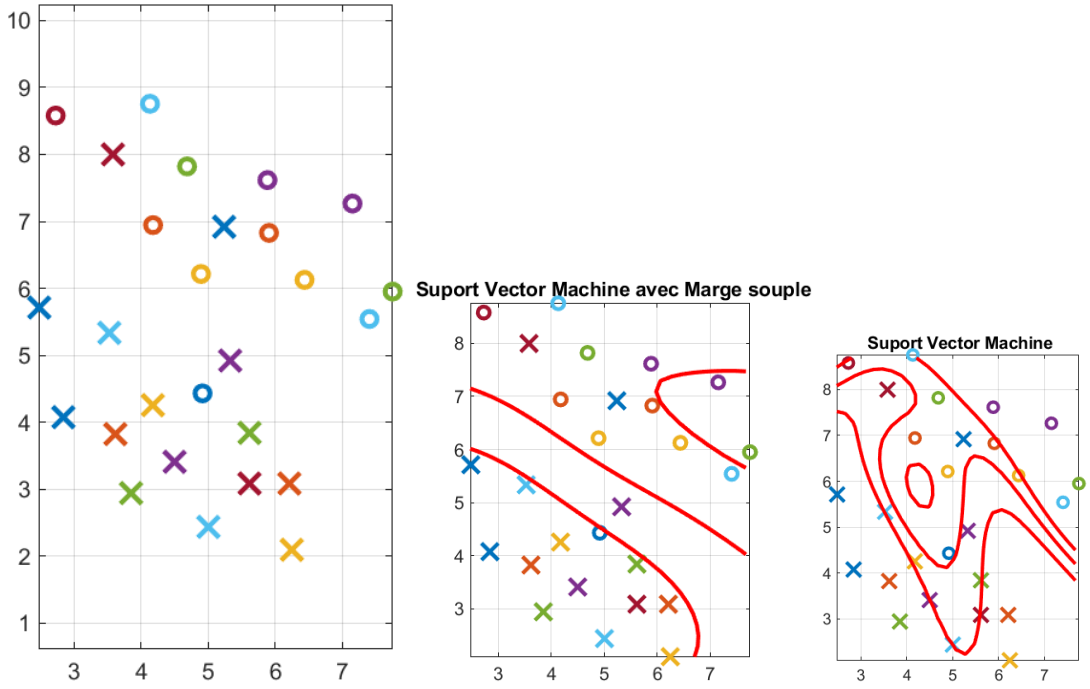
En passant au lagrengien, on obtient une condition supplémentaire tels que : $\alpha_i < C$ d'après la nullité du gradient par rapport à ξ_k.
Le problème dual devient du coup :

$$\begin{cases} \max_{\alpha \geq 0}(H(\alpha)) \\ \sum_{i=1}^{p}\alpha_i l_i = 0 \\ \quad 0 \leq \alpha \leq C \end{cases}$$

L'implémention de cette nouvelle condition se fait par :

```
if marge_souple == true
    alpha_k_plus_1 = min(alpha_k_plus_1,C_souple);
end
```

On applique cette nouvelle méthode sur la base de donnée **data4** par ce qu'il contient 3 outliers, comme vous pouvez voir dans l'image. Dans notre cas, ces outiliers présentent un grand risque puisque on étudie des cellules cancéreuse et le fait de considérer une de ces cellules comme sain est une faute fatale.
Pour visualiser bien la différence entre la méthode classique et la méthode de marge souple, Nous comparons entre ces deux méthodes et on remarque que les deux techniques donnent des hyperplans totalement différents .Ainsi, l'hyperplan obtenu par les SVM à marges souples est bien meilleur et sépare bien les données.

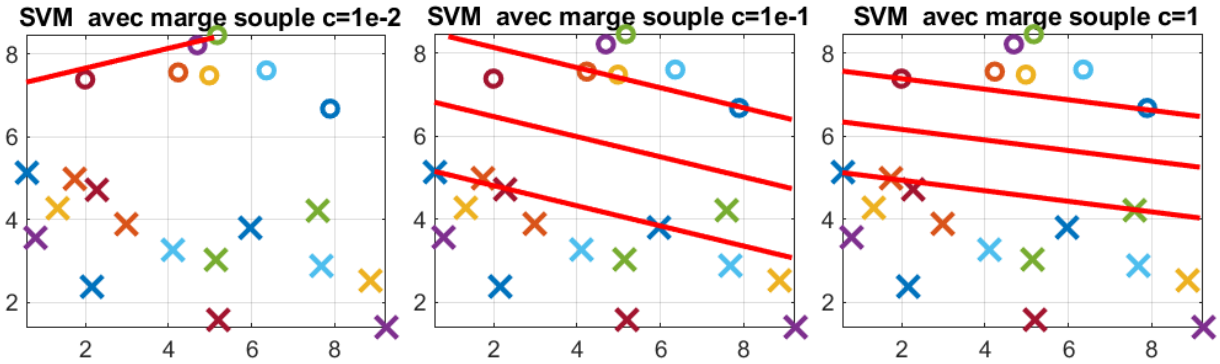


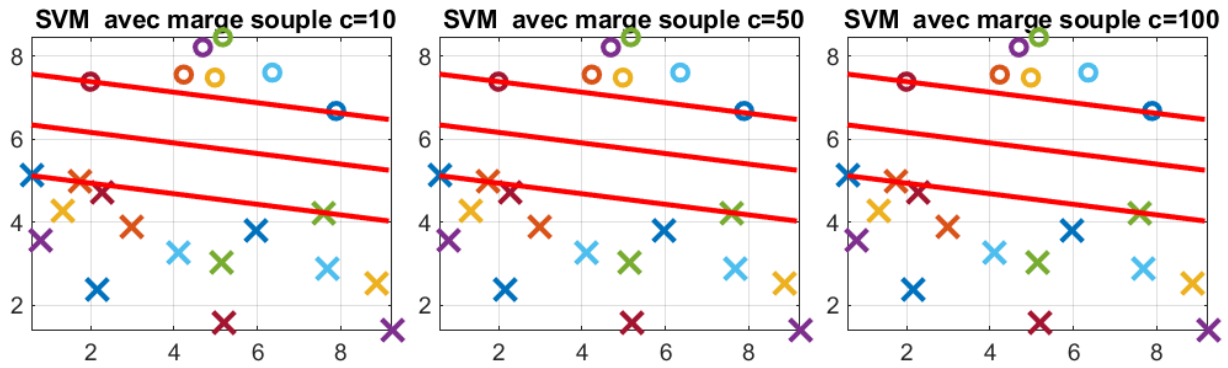
3.2 Influence des paramètres

Dans cette section, on étudie l'influence des différent paramètres

3.2.1 La souplesse

On test la méthode de marge souple sur la base de données **data1** qui est séparable linéairement en variant la souplesse de la marge à chaque fois.

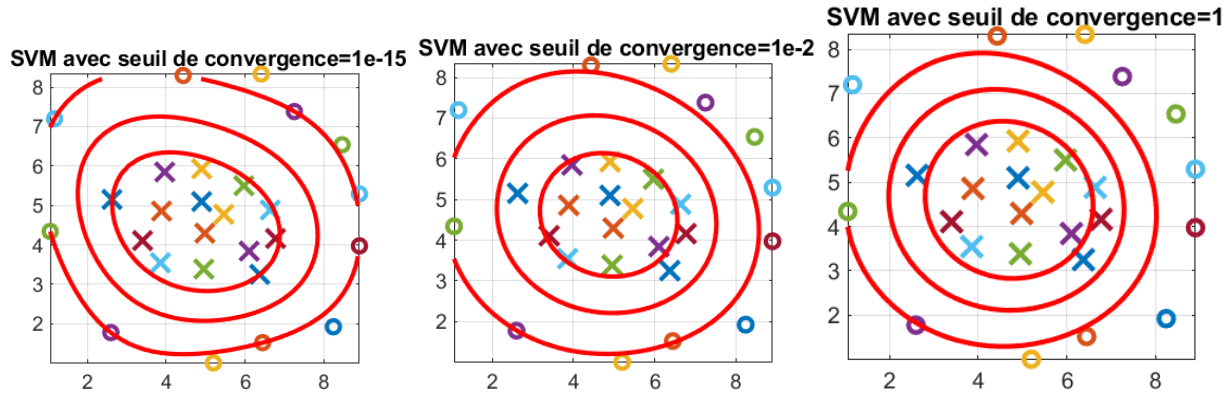




On remarque que plus la valeur de C augmente plus la séparation est mieux. Autrement dit une grande valeur de C diminue le nombre de points mal classés. Ainsi, les petites valeurs de C nous donne une marge de SVM très grande chose qui augmente le nombre de points support qui peuvent conduire à un sous-ajustement. Pourtant, une valeur très élevée de C (C=100) nous donne une marge petite mais en même temps augmente le poids des échantillons non séparables. Le classifieur devient du coup sensible au bruit puisque une valeur aberrante ou un échantillon critique peuvent déterminer la limite de décision, chose qui peut conduire à un surajustement.
On trouve exactement les même résultat dans le cas non séparable linéairement.

3.2.2 Seuil de convergence

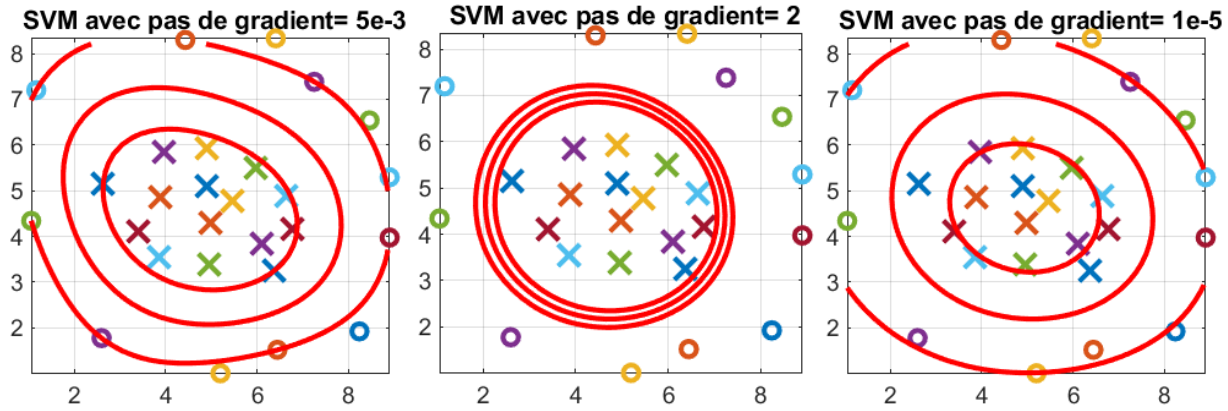
On varie le seuil de convergence ϵ dans le cas non séparable linéairement



On remarque que plus le seuil de convergence est petit plus la discrimination est meilleur. Pourtant, on a pu remarquer lors de variation de ϵ que au bout d'une valeur la marge de SVM cesse d'augmenter, Du coup on ne peut pas se tromper avec une petite valeur de convergence comme le cas de la souplesse.
Le choix d'une valeur grande de ϵ minimise la largeur et augmente le poids des points nos séparables, chose qui entraîne une sensibilité envers le bruit et du coup un surajustement.

3.2.3 Pas de gradient

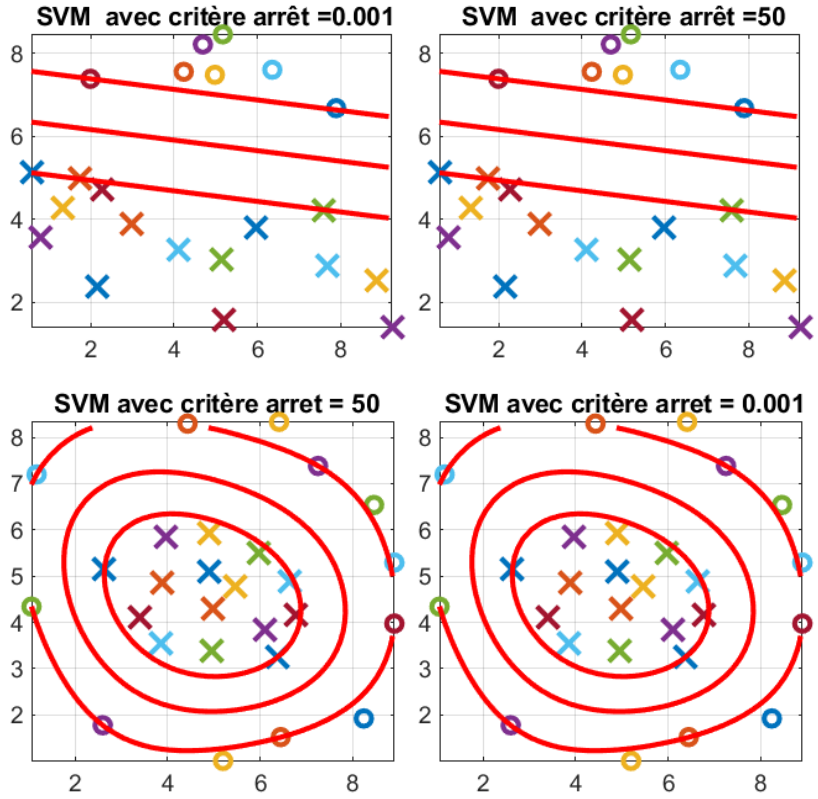
On varie le pas de gradient



Le choix d'un pas de gradient très grand nous donne un résultat totalement faux. Ceci est dû au fait que le choix d'un learning rate trop grand a l'avantage de descendre rapidement vers le minimum, et on peut remarquer pour pasgrad=2 la marge de SVM est trop petite.

3.2.4 Le critère d'arrêt

On observe l'influence de la variation du critère d'arrêt de le cas séparable et non séparable linéairement

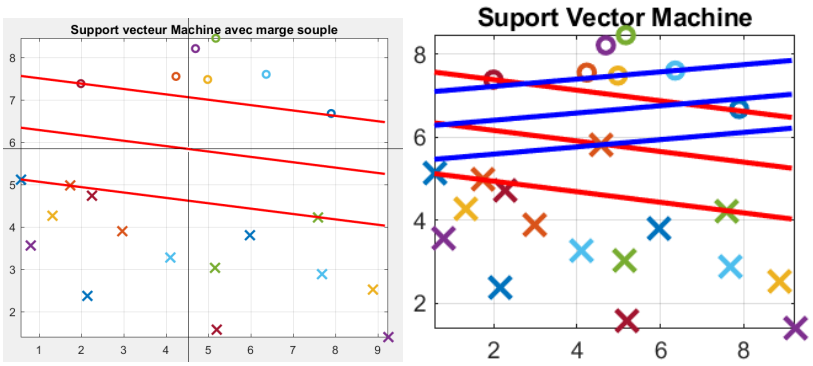


Le critère d'arrêt n'a pas vraiment une influence sur le résultat.

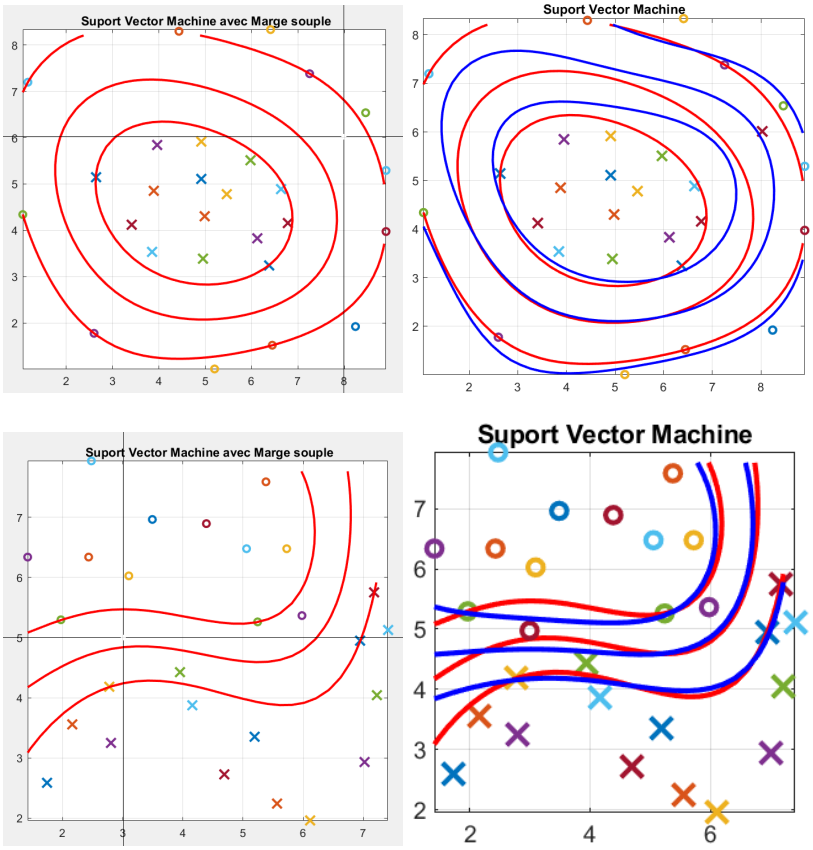
3.3 L'ajout d'un point

Dans cette partie, on ajoute un point x_n dans tous les cas, dans le but de vérifier la capacité de logiciel construit à classer automatiquement ce point dans l'une des deux classes.
En s'inspirant du modèle des K plus proches voisins (KNN), On implémente un modèle qui consiste à tracer un cercle de centre le nouveau point qu'on a ajouté et de rayon qu'on choisit selon la densité des points dans le graphe. Pourtant que avant de choisir ce dernier on calcule la distance $(x_1 - x_n)^2 + (x_2 - x_n)^2$ pour les points x aléatoire juste dans le but d'avoir une idée de l'ordre de grandeur du rayon à choisir.

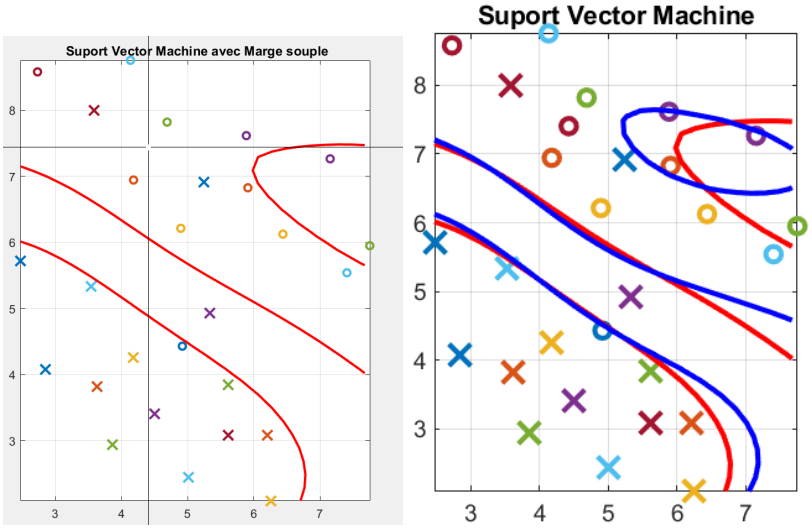
Ensuite on calcule le nombre de points de label 1 présents dans le cercle ainsi que le nombre de celles de label -1. S'il excite plus de point de label -1, on attribut a ce nouveau point le label -1,sinon il sera classé avec les points de label 1.
On teste ce modèle sur nos 4 base de données :



Dans le premier cas, on a choisit d'ajouter un point au milieu entre les deux classes afin de s'assurer bien que le modèle travaille. On remarque que la marge de séparabilité des deux groupes a diminué et le nouveau point est labelisé 1. En déduit que malgré le choit du point est fait au milieu, la densité des points de label 1 est plus grande que la densité des points de label -1.

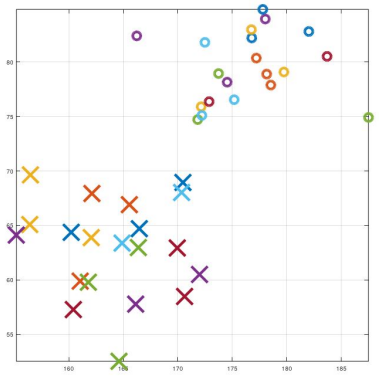


De même dans le quatrième cas, on a choisi une zone critique ou les deux classes sont très proches l'une des l'autre et on voit bien que seul la densité détermine le label, et le nouveau point est classé parmi les points de label 1.

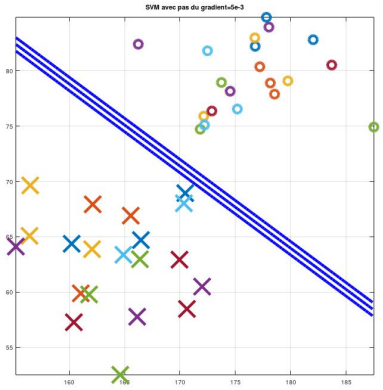


3.4 Cas d'étude

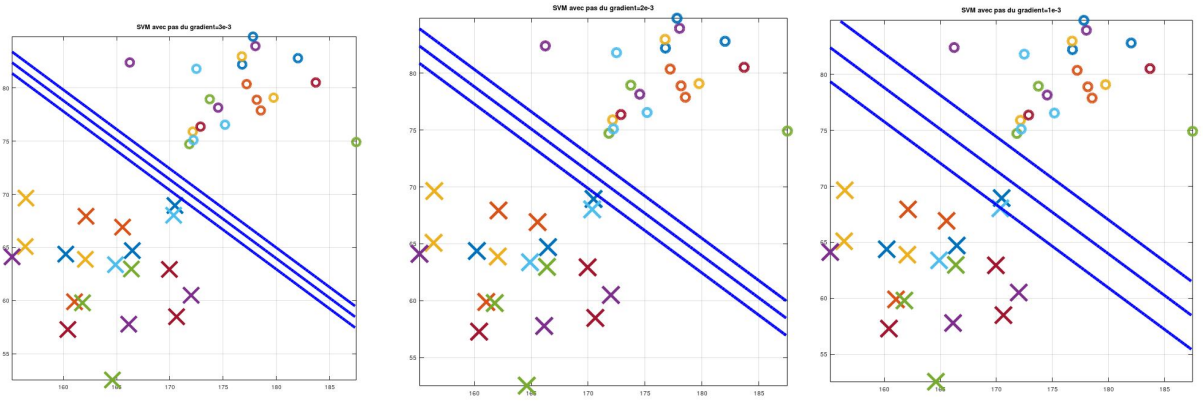
On cherche à savoir s'il est possible de déterminer le genre d'un individu à partir de ses mensurations. On a pour cela une matrice *mensur* de taille (5x40) : 40 observations de 5 variables. Contrairement aux exemples précédents, nous ne somme plus en dimension 2. On commence par projeter les observations selon les deux premières variables pour pouvoir les observer :



On voit que les deux classes sont linéairement séparables selon ces deux variables : on va donc rester en dimension 2 et éliminer les autres variables, qui sont superflues pour la séparation. On voit qu'il n'y a pas d'outlier, on va donc considérer des marges strictes. De plus, comme les deux classes sont linéairement séparables, on se place dans le cas d'un noyau linéaire (produit scalaire).
On fait un premier test avec les paramètres pas du gradient = 5e-3 et seuil de convergence = 0.1 :



Les deux classes sont bien séparées, bien que les marges ne soient a priori pas maximales. On fait d'autres essais en changeant le pas du gradient :



On voit une bonne séparation linéaire lorsque le pas est assez petit. Le paramètre pas = 2e-3 semble être optimal pour la taille des marges.
Finalement, les points d'apprentissage sont effectivement linéairement séparables comme montré ci-dessus. On peut cependant supposer que cela est dû à la faible taille de l'ensemble d'apprentissage, et qu'un ensemble plus conséquent montrerait des données non séparables linéairement, voire non séparables.

4 SVR

Dans un contexte de régression, on ne recherche plus l'hyperplan qui va séparer au mieux. On va dans ce cas là chercher à approcher au mieux un ensemble de p couples (x_i, y_i). Cela revient à chercher $w \in R^p$ et $b \in R$ tels que : $| < w, x_i > +b - y_i| < \epsilon$. Par analogie avec SVM, le problème d'optimisation primal s'écrit :

$$\begin{cases} \min(\frac{1}{2}||w||^2) \\ |y_i - < w, x_i > -b| \leq \epsilon, i = 1, ..., n \end{cases}$$

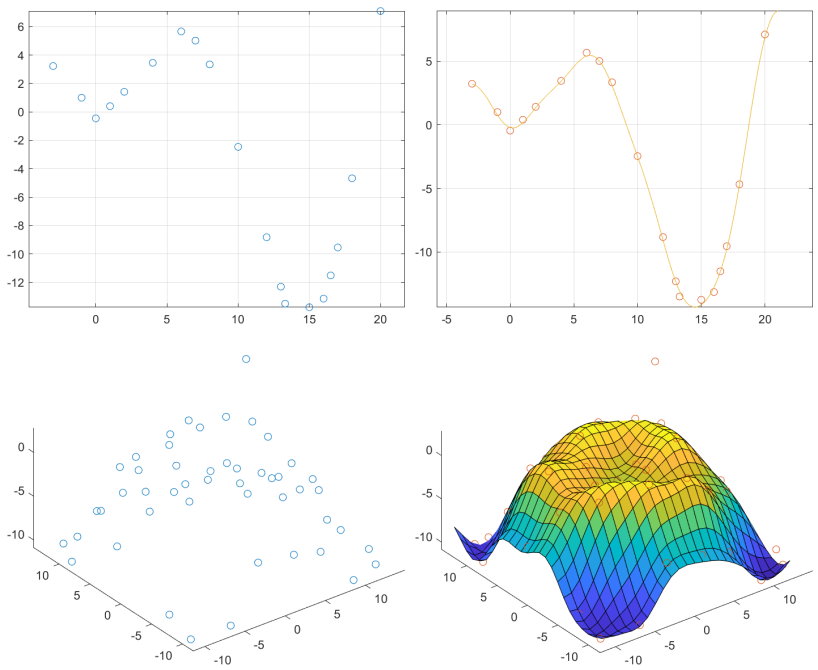
Les contraintes impliquent que toute les observations doivent se définir dans une marge ou bande de taille 2ϵ. Cette hypothèse peut amener l'utilisateur à utiliser des valeurs de ϵ très grandes et empêcher la solution de bien ajuster le nuage de points. Pour pallier cela, on introduit, comme dans le cas de la SVM binaire, des variables ressorts qui vont autoriser certaines observations à se situer en dehors de la marge. Le problème devient :

$$\begin{cases} \min(\frac{1}{2}||w||^2) + C\sum_{i=1}^{n}(\xi_i + \xi_i^*) \\ y_i - < w, x_i > -b \leq \epsilon + \xi_i, i = 1, ..., n \\ < w, x_i > +b - y_i \leq \epsilon + \xi_i^*, i = 1, ..., n \\ \xi_i \geq 0, \xi_i^* \geq 0 \end{cases}$$

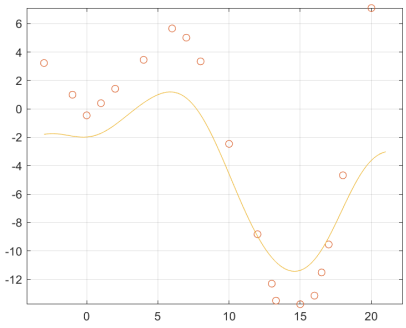
avec le Lagrangien qui s'écrit sous forme : $L = \frac{1}{2}||w||^2 + C\sum_{i=1}^{n}(\xi_i + \xi_i^*) - \sum_{i=1}^{n}(\eta_i\xi_i + \eta_i^*\xi_i^*) - \sum_{i=1}^{n}\alpha_i(< w, x_i > +b - y_i + \epsilon + \xi_i) - \sum_{i=1}^{n}\alpha_i^*(y_i - < w, x_i > -b + \epsilon + \xi_i^*)$ le problème dual, ainsi :

$$\begin{cases} \max_{\alpha \in R^{2p}} H(\alpha) \\ u_3^T\alpha = 0 \\ \alpha \in [0, C]^{2p} \end{cases}$$

Avec $H(\alpha) = -\frac{1}{2}\alpha^T A\alpha - u_1^T\alpha + u_2^T\alpha$.
L'étape suivante est de déterminer b. En se servant des points supports, on trouve $b = \epsilon + y_i - < w, x_i >$ or $b = y_i - < w, x_i > -\epsilon$.
On applique cette méthode pour approcher d'abord une fonction de R dans R puis une fonction de R dans R^2, tout en utilisant le noyau gaussein (kernel=2),et on trouve les résultats suivant :



Le paramètre C-souple joue un rôle très important. Il a un grand influence sur le résultat, Au sens que le choix d'une valeur très petite nous donne des valeurs qui sont totalement mal prédîtes. comme vous pouvez voir pour une c=1.5.
Le choix d'une marge de souplesse très petite entraîne un sous-ajustement, alors que le choix d'une valeur très grande entraîne un surajustement.



5 Conclusion :

Support vector machine (SVM) et Support vector regression (SVR) sont des méthodes de classification et de régression, d'apprentissage en générale trop connus. Ils ont été appliqué dans plusieurs domaines.
D'après ce TP, on peut dire la chose la plus délicate dans l'implémentation de ces deux méthodes et le choix des paramètres (la souplesse C, le pas de gradient, le seuil de convergence, epsilon...).
Vous trouvez en joint de ce Rapport 3 fichier Matlab : une fonction pour SVM, une pour SVR et la fonction Kernel puisque on a ajouté d'autres noyaux.