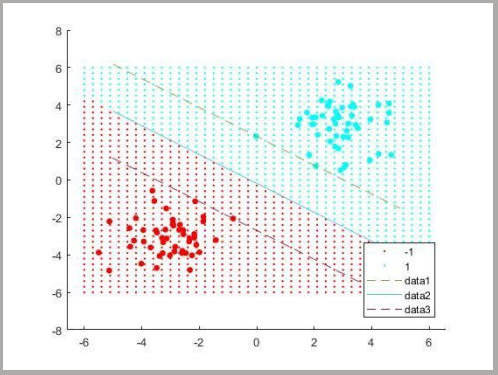
Rapport Valerian DAMM

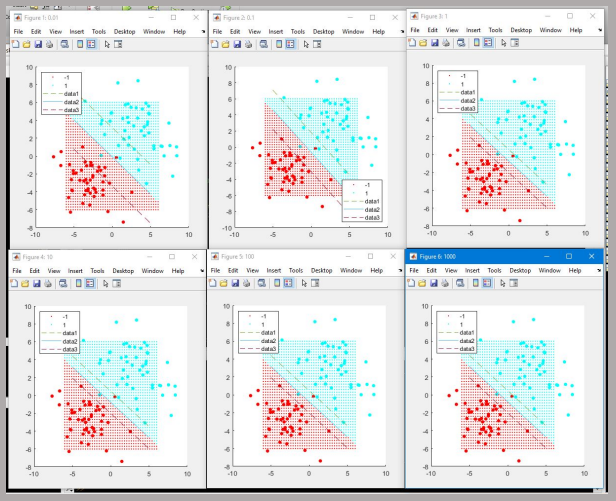
Apprentissage statistique et sélection de modèle TP : classification SVM et méthodes de décomposition

1 Classification SVM

1.1 Cas séparables résultats



1.2 Cas non séparable résultats



2 Problème multi-classe et techniques de validation croisée : application à la classification de défauts de rails

2.3 Estimation de l'erreur de généralisation par validation croisée

Création de **assm2.m** qui est une modification de **assm.m** qui est lui-même une modification de **rails.m**

Toutes les opérations sont réalisées pour C = 1 via **executionUnique.m**

les erreurs de validation croisée pour chacun des classifieurs binaires nous indique le pourcentage de donnée qui ''bruite'', qui sont proche de la marge et mal classée, de chaque classifieurs.

Classe 1 : 10.71

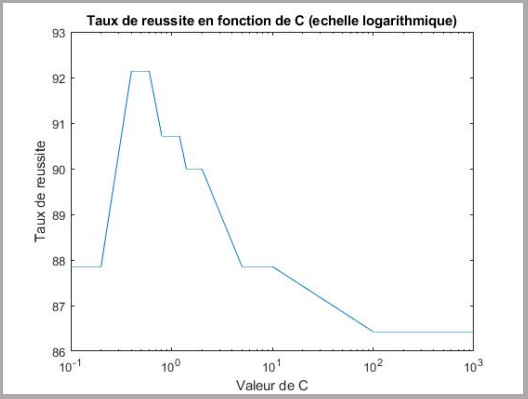
Classe 2 : 8.57

Classe 3 : 5.71

Classe 4 : 2.86

2.4 Sélection de modèle

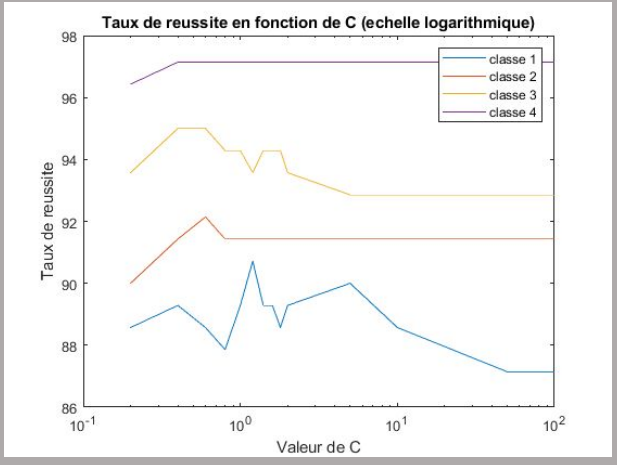
Une méthode naïve et relativement efficace consiste à tester plusieurs valeur de C, les opérations n'étant pas non plus trop gourmande en temps de calcul, on peux tracer la courbe du taux de réussite en fonction de C.
Donnée générer par **genereDataGraph.m**



0,1	0,2	0,4	0,6	0,8	1	1,2	1,4	1,6	1,8	2	5	10	100	500	1000
87,86	87,86	**92,14**	92,14	90,71	90,71	90,71	90,00	90,00	90,00	90,00	87,86	87,86	86,43	86,43	86,43

On trouve ainsi que le meilleur taux de réussite se trouve pour les valeurs de c = 0.4 et c = 0.6 avec pour valeur 92.14% se qui nous donne 7.86% d'erreur.

Si on sélectionne les C séparément on pourrai obtenir un gain de performance de notre model. Mais rien ne nous dis que nous ne risquons pas de faire du surapprentissage. Le temps d'exécution en revanche ne devrai pas être plus long qu'avec des C identiques.
Donnée générer par **genereDataGraphCDiff.m**



C	0,20	0,40	0,60	0,80	1,00	1,20	1,40	1,60	1,80	2,00	5,00	10,00	50,00	100,00
classe 1	88,57	89,29	88,57	87,86	89,29	**90,71**	89,29	89,29	88,57	89,29	90,00	88,57	87,14	87,14

classe 2	90,00	91,43	**92,14**	91,43	91,43	91,43	91,43	91,43	91,43	91,43	91,43	91,43	91,43	91,43
classe 3	93,57	**95,00**	95,00	94,29	94,29	93,57	94,29	94,29	94,29	93,57	92,86	92,86	92,86	92,86
classe 4	96,43	**97,14**	97,14	97,14	97,14	97,14	97,14	97,14	97,14	97,14	97,14	97,14	97,14	97,14

Résultat du classifieur avec hyperparamètre réglé indépendamment :

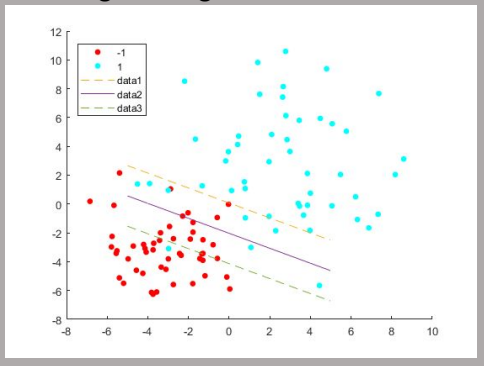
93.75% de réussite

Résultat du classifieur avec hyperparamètre commun :

92.14% de réussite

3 Programmation quadratique

Résultat du test du SVM personnaliser générer grâce à : svmPerso.m



Les résultats sont légèrement différent à chaque exécution.
Si les données sont trop proches les unes des autres elles risque d'etre toutes dans la marge.
Je n'ai pas réussis a implémenté le duale (Cf : apprendSVMdual.m)