

MLEA : SVM(Support Vector Machine)

Thibault LAPASSADE - Florian THOMASSIN

Le but de ce rapport est de présenter les résultats de notre étude des kernel de différents SVM dans le but de déterminer le plus efficace de ces kernels. Pour cela, on distinguera deux parties :

Les différents kernels qui permettra de présenter les différents types de kernel choisis et le choix des paramètres pour chacun.

Le benchmark qui permettra de présenter grâce à des graphiques les résultats de chacune de nos expérimentations sur les kernels précédemment choisis.

Notre système de test est réalisé à l'aide de Matlab. Le format d'entrée des données d'acquisition est fixé et n'est pas le sujet d'étude de ce compte-rendu. On s'attardera plutôt à détailler l'analyse des kernels et du choix des paramètres de ceux-ci de manière à avoir la meilleur classification.

Choix des kernels et des paramètres

Afin de pouvoir comparer les différents comportement d'un SVM en fonction du type de kernel (linear, polynomial, RBF, Laplacien RBF). De plus, nous allons faire varier le paramètre C et Gamma (si présent) pour étudier la variation de l'erreur.

Pour chaque type de kernel, nous allons tester les valeurs de C suivante :

- $C = 1$
- $C = 10$
- $C = 100$
- $C = 1000$
- $C = 10000$
- $\text{Gamma} = 2$ et 4

Afin de mener à bien se benchmark, nous avons utiliser l'implementation fait en TP du SVM. Deplus, nous avons choisis le set IRIS pour faire nos tests. Possédant 3 dimensions nous avons utilisé l'algorithme DAG pour départager le SVM.

Nous avons défini un protocole de test suivant pour chaque kernel :

- Entraînement des SVMs
- Tests des SVMs
- Application de l'algo DAG
- Récupération de l'erreur
- Création des diagrammes

Kernel Linear - Une fois la phase test fini, nous avons constaté que les performances de ce kernel se rapprochait d'une classification aléatoire puisque le taux d'erreur est d'environ 50%.

Kernel Polynomial - Pour ce kernel nous avons choisi le second ordre et l'ordre 4 afin d'avoir une assez grande marge pour avoir des résultats assez différents. Mais malgré ce changement d'ordre, nous avons toujours des taux d'erreurs très proches.

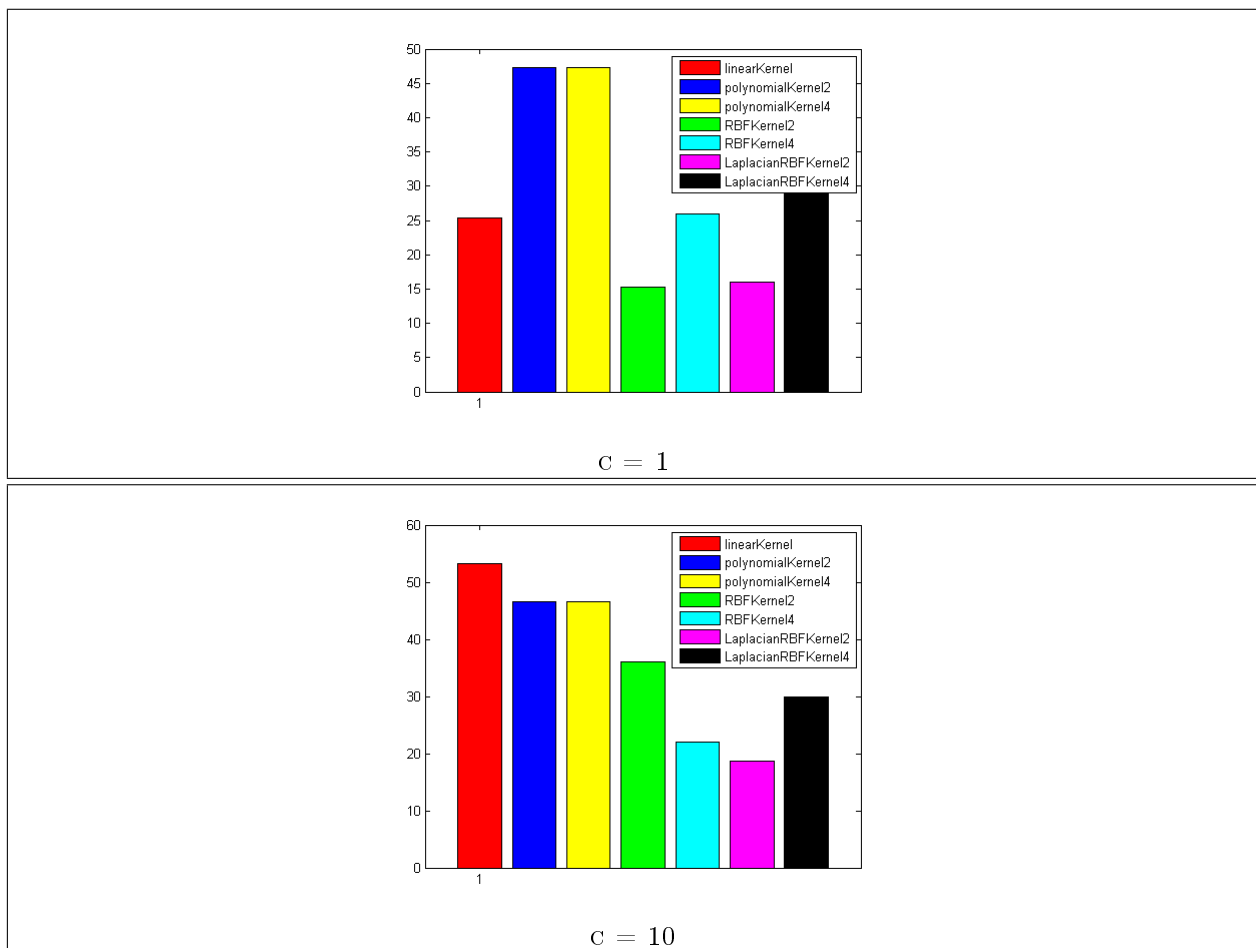
Kernel Laplacian RBF - Nous nous sommes également aperçu que les taux d'erreurs de ce kernel n'évoluaient pas énormément en fonction de la valeur de c. Néanmoins, nous avons constaté que les taux d'erreurs obtenus étaient

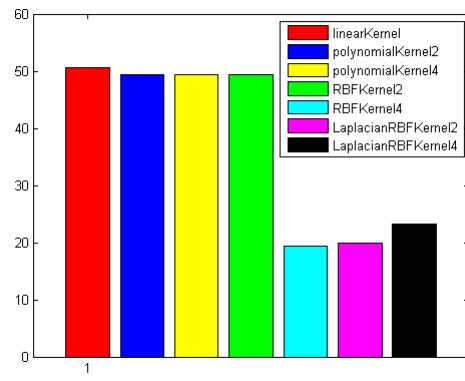
bien meilleur que ceux avec un kernel polynomial puisque l'on divisait presque par moitié le taux d'erreur par rapport aux différents kernels polynomiaux. Nous avons ensuite testé avec un gamma de 4 de manière à voir si cela avait une influence ou non. Nous avons ainsi vu que les résultats n'étaient pas meilleurs mais au contraire plus mauvais qu'avec un gamma égale à 2.

Kernel RBF - Nous avons donc mis en place un kernel RBF avec les mêmes paramètres que le Laplacian RBF. Les résultats obtenus sont identique aux précédent. Néanmoins, nous nous sommes aperçu que le kernel Laplacian RBF donnait de meilleur résultat que le kernel RBF. Nous avons néanmoins aussi remarqué que certaines fois, le kernel RBF avec un gamma égale à 2 avait de moins bon résultat que le kernel RBF avec un gamma égale à 4 mais puisqu'il y avait plus de fois où le taux d'erreur est inférieur lorsque le gamma valait 2, nous nous sommes dit que cela n'était pas significatif.

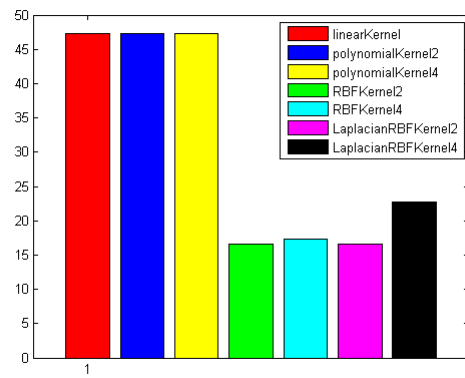
Nous avons ensuite essayé de tester avec des kernels étant des combinaisons de kernel précédemment tester mais les temps étaient beaucoup plus grand et ne donnaient pas vraiment de meilleur résultat et donc nous avons décidé de ne pas les faire apparaître dans notre benchmark.

Benchmark des kernels choisis

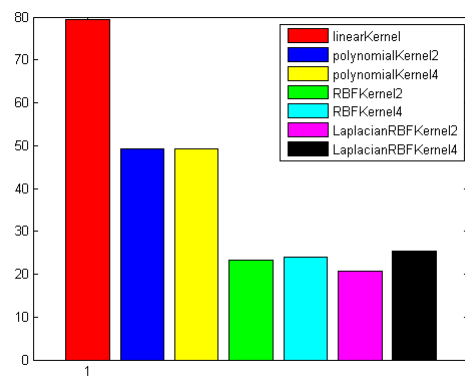




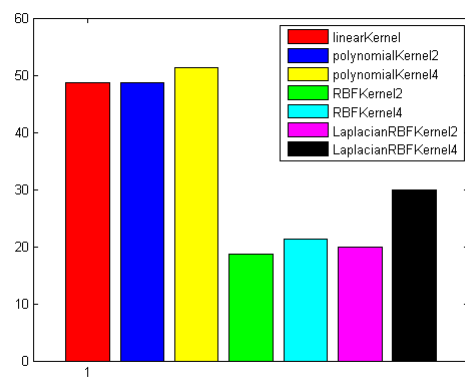
$C = 100$



$C = 1000$



$C = 10000$



$C = \text{INF}$

Conclusion

À partir des images des benchmarks précédents ainsi que grâce aux différents tests que nous avons pu effectués sur les différents kernels, on s'aperçoit que le kernel LaplacianRBF avec comme paramètre $\gamma=2$ semble être le kernel le plus à même de bien classer les données dont nous disposons pour le test. Néanmoins, il ne faut pas oublier que nos benchmarks ont été uniquement effectués sur le set de data iris et pas sur le set de data opdigit pour cause de problème lors de l'apprentissage sur ce set et que nos résultats sont peut être biaisés par ce manque de données de test et ne reflètent donc peut être pas la réalité.