***Contexte***

Nous allons traiter le fameux sujet des régression Ridge ou Les SVM à noyaux. Ils sont implémentées dans scikit-learn dans les classes [sklearn.svm.SVC](http://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html" \t "_blank)pour la classification et [sklearn.svm.SVR](http://scikit-learn.org/stable/modules/generated/sklearn.svm.SVR.html" \t "_blank) pour la régression. Dans ces deux classes, vous pouvez spécifier un noyau grâce au paramètre « kernel ». Ce noyau peut être un des grands classiques (linéaire, polynomial, RBF), mais vous pouvez aussi définir vos propres noyaux !

Regardons comment utiliser la classe sklearn.svm.SVC en pratique. Nous allons utiliser les données concernant les caractéristiques physico-chimiques du lait disponibles sur les archive.

1. Il s'agit ici de prédire le score (entre 3 et 9) donné par des experts aux différentes variance du lait. Chargeons les données et transformons le problème en un problème de classification, pour lequel il s'agira de prédire si le score est supérieur à 6 (bonne qualité) ou non .
2. Avant toute chose, nous allons découper nos données en un jeu d'entraînement (X\_train, y\_train) et un jeu de test (X\_test, y\_test).
3. nous pouvons enfin entraîner notre première SVM à noyau !

# Créer une SVM avec un noyau gaussien de paramètre gamma=0.01

from sklearn import svm

classifier = svm.SVC(kernel='rbf', gamma=0.01)

# Entraîner la SVM sur le jeu d'entraînement

1. Comment se comporte-t-elle sur le jeu de test ? Nous allons pour le comprendre regarder la courbe ROC.

y\_test\_pred = classifier.decision\_function(X\_test\_std)

# construire la courbe ROC

from sklearn import metrics

fpr, tpr, thr = metrics.roc\_curve(y\_test, y\_test\_pred)

Et voilà notre courbe ROC, avec une AUC de 0.81, plutôt pas mal !

