TD n^o 7 : Régression polynômiale et Capacité de représentation

Exercice 1 : (**Mise en route**)

La régression polynômiale se réduit à un problème de régression linéaire. Ici, on va essayer de trouver le polynome de degré 2 et de deux variables x et y qui passe le plus près des données. Le polynôme que l'on cherche est donc de la forme :

$$P(x, y) = a_{20}x^2 + a_{02}y^2 + a_{11}xy + a_{10}x + a_{01}y + a_{00}$$

Cependant, les données d'entraînement sont de la forme $P(x_i, y_i) = v_i$ pour i allant de 1 à la taille de l'ensemble d'entraînement. On transforme l'ensemble d'entraînement en réécrivant chacun des exemples de la manière suivante :

$$(x_i^2, y_i^2, x_i y_i, x_i, y_i, 1) \rightarrow v_i$$

On se place ainsi dans un espace de dimension 6, mais dans cet espace, le problème s'exprime de manière linéaire : trouver le vecteur A composé des coefficients $a_{20}, a_{02}, a_{11}, a_{10}, a_{01}$ et a_{00} qui minimise l'erreur quadratique.

Nous allons voir comment automatiser cette transformation avec SciKit Learn.

1. Il faut tout d'abord importer les modules utiles.
```
import numpy as np
from sklearn.preprocessing import PolynomialFeatures
from sklearn import linear_model
```
La deuxième ligne (`PolynomialFeatures`) va nous permettre de passer assez facilement de l'espace de dimension 2 à l'espace en dimension 6.
2. On déclare les données d'entraînement ainsi que le nouveau point pour lequel on veut calculer la valeur à partir de la régression :
```
X = np.array([[ 1.61,-1.51], [-0.86, 0.55], [ 0.22,-0.10], [1.19,-0.50], [-1.34,-0.84],
              [ 0.47,-1.22], [-1.94, 1.15], [-0.01, 0.58], [-0.68,-1.07]])
Y = np.array([11.88523, -0.6307, 1.6322, 5.28805, 1.74788, 6.55957, -0.3427,
              0.71557, 3.53702])
newX= np.array([[0.49, 0.18]])
```
3. On utilise alors `PolynomialFeature` pour changer d'espace :
```
poly = PolynomialFeatures(degree=2)
X_ = poly.fit_transform(X)
newX_ = poly.fit_transform(newX)
```
La deuxième ligne permet de calculer le vecteur $(1, x_i, y_i, x_i^2, x_i y_i, y_i^2)$ pour chacun des points (x_i, y_i) de `X`. La troisième ligne fait de même pour `Xnew`.
4. On s'intéresse alors à la regréssion linéaire dans l'espace de dimension 6 :
```
clf = linear_model.LinearRegression()
clf.fit(X_,Y)
print(clf.predict(newX_))
```
5. En déduire le polynôme de degré 2 qui a servi à générer les données d'entraînement.

Exercice 2 : (**Vers un bon choix de la capacité de représentation**)

L'objectif de cet exercice est de montrer que l'étude de l'erreur cumulée sur l'ensemble d'entraînement et celle sur l'ensemble de test (généralisation) permet de trouver la bonne capacité de représentation. Ici, nous nous intéressons aux régressions polynômiales, et la capacité de représentation sera donc représentée par le degré maximal des polynômes considérés.

1. Créez une fonction qui calcule la valeur d'un polynôme particulier de la variable x de degré 5.
2. Créez deux jeux de données : `Xtrain` et `Xtest`, chacun constitué d'une dizaine de valeurs de x. Calculez les vecteurs valeurs associées `Ytrain` et `Ytest`.
`Ytrain` sera ensuite modifiée en ajoutant à chaque valeur calculée, un bruit provenant d'une loi uniforme sur l'interval $[-1, 1]$.
3. Ecrivez ensuite une fonction qui prend en argument : l'ensemble d'entraînement, l'ensemble test, les valeurs d'entraînement, les valeurs de test et le degré d du polynôme pour la régression. Cette fonction transformera d'abord les ensembles d'entraînement et de test pour passer dans l'espace de dimension $d + 1$ où le problème s'exprime linéairement. La fonction construit dans cet espace le classifieur linéaire, le calibre en utilisant les données d'entraînement, calcule les erreurs de prédiction sur l'ensemble d'entraînement et sur l'ensemble de test, et finalement renvoie la norme de l'erreur sur chacun des deux ensembles.

4. On utilisera alors la fonction précédente pour faire des régressions sur l'ensemble des polynômes de degré d, $d \in \{1, 2, 3, ...15\}$. A chacun des degrés considérés, on associera les erreurs sur l'ensemble d'apprentissage et sur l'ensemble de test.
5. Tracez sur un même graphique les deux courbes : l'erreur sur l'ensemble d'apprentissage en fonction du degré ; et l'erreur sur l'ensemble de test en fonction du degré.