Objectifs du TP :

- Découvrir l'apprentissage supervisé par l'application de l'algorithme : Support Vector Machine (SVM) sous scikit-learn.
- Apprendre à contrôler les paramètres du classificateur SVM.

Apprentissage par Machines à Vecteurs de Support sous sklearn

Les machines à vecteurs de support (SVM : Support Vector Machines) appartiennent à la classe des méthodes d'apprentissages statistiques basées sur le principe de la maximisation de la marge de séparation entre classes. Il existe plusieurs formulations (linéaires, fonctions à noyaux). La résolution du problème de SVM peut s'effectuer grâce à l'introduction d'un terme d'équilibrage C > 0 fixé qui permet de poser le problème sous la forme suivante :

$$\min_{\beta,\beta_0,\xi_i} \frac{1}{2}\|\beta\|^2 + C\sum_{i=1}^{n}\xi_i$$

$$s.c.$$

$$y_i \times (x_i^T\beta + \beta_0) \geq 1 - \xi_i, \forall i = 1,\ldots,n$$

$$\xi_i \geq 0,\ \forall i$$

Pour une meilleure manipulation de la méthode de SVM sous Scikit-learn, vous pouvez vous baser sur la documentation fournie dans l'adresse suivante : http://scikit-learn.org/stable/modules/svm.html

Nous utiliserons l'objet **sklearn.svm.SVC** :
from **sklearn.svm** import **SVC**

SVC est une classe de SVM, capable d'effectuer une classification multi-classes sur un ensemble de données dans le cas où la variable de sortie Y compte plus de deux modalités. La description complète de la formulation est disponible sur le site : http://scikit-learn.org/stable/modules/svm.html#svm-mathematical-formulation.

- Mise en oeuvre –

Les paramètres principaux de la classe **SVC** sont les suivants :

- **c**: ''cost'' est un paramètre de tolérance aux erreurs.
- **kernel** (par défaut = 'rbf', Radial Basis Function): spécifie le type de noyau à utiliser dans l'algorithme ('poly', 'rbf', 'sigmoid', 'precomputed').
- **decision_function_shape :** 'ovo', 'ovr', default='ovr'. Il existe plusieurs façons d'étendre directement les méthodes du cas binaire : **"Un contre un :** '**ovo' ",** dans le cas où l'on cherche à prédire un label pouvant prendre K ≥ 3 classes, on peut considérer toutes les paires de labels possibles. Si K est le nombre de classes, alors nous aurons K * *(K - 1) / 2* classifieurs pour chacune d'entre elles. La prédiction correspond alors au label qui a gagné le plus de "duels". **"Un contre tous 'ovr'"** dans le cas où on apprend un classifieur a discriminé

entre les populations $Y = k$ et $Y \neq k$. A partir des estimations des probabilités a posteriori, on affecte le label estimé le plus probable.

- **Gamma (par défaut = 'auto')** : est un coefficient utilisé dans les noyaux 'rbf', 'poly' et 'sigmoïde'.

Travail à faire

Ecrivez un programme appelé TP3_prog1.py qui permet de :

- Charger le jeu de données mnist,
- Diviser la base de données à 70% pour l'apprentissage (training) et à 30% pour les tests,
- EN se basant sur les fonctions usuelles :
  - classifier = svm.SVC()
  - classifier.fit()
  - predicted = classifier.predict()
- Construire un modèle de classification ayant comme paramètres un noyau linear: clsvm = svm.SVC(kernel='linear').
- Tentez d'améliorer les résultats en variant la fonction noyau : 'poly', 'rbf', 'sigmoid', 'precomputed'.
- Faites varier le paramètre de tolérance aux erreurs C.
- Tracez la courbe d'erreur de classification sur les données d'entrainement et de test en fonction de C.
- Faites varier le paramètre gamma.
- Calculez la précision du classificieur pour chaque gamma sur les données d'entrainement et de test.
- Construisez un classifieur à partir des données en utilisant **GridSearchCV** pour trouver les meilleurs paramètres.
- Construisez la matrice de confusion en utilisant le package metrics
  - from sklearn.metrics import confusion_matrix
  - cm = confusion_matrix(Y_test, Y_pred)

- Sur chacun des cas précédents, tracez les différentes courbes :
  - Temps d'apprentissage,
  - Précision, rappel
  - Erreur.
- A votre avis, quels sont les avantages et les inconvénients du SVM ?
- Sur chacun des classifieurs précédents (K-nn, MLP, et SVM), tracez les différentes courbes (pour la même taille d'un échantillon) :
  - Temps d'apprentissage,
  - Précision, rappel, erreur.
  - Matrice de confusion
- A votre avis, quel est le meilleur classifieur et quels sont ses hyperparamètres finaux ?

Test des SVMs sur une autre base de données

Diagnostic du diabète chez les femmes:

- 8 descripteurs :

1. Number of times pregnant
2. Plasma glucose concentration a 2 hours in an oral glucose tolerance test
3. Diastolic blood pressure (mm Hg)
4. Triceps skin fold thickness (mm)
5. 2-Hour serum insulin (mu U/ml)
6. Body mass index (weight in kg/(height in m)^2)
7. Diabetes pedigree function
8. Age (years)

- nombre d'individus : 768

Importation de la base de données

```
import pandas

pima = pandas.read_table("pima.csv",sep=",",header=0)

#dimensions

print(pima.shape)
```

Questions :

➔ Faire varier, le paramètre « C » de 0.01 à 2 le « gamma » de 10^{-5} à 2 10^{-4} et le type de noyau « kernel »

•