**NOUTCHEU TCHAGNA CHOUBEAL**

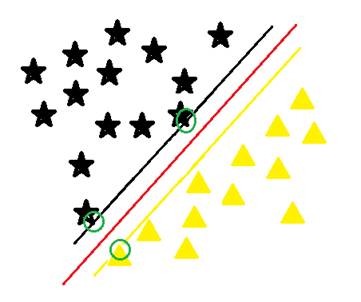
**Documentation TP2 : Présentation du Support Vector Machine(SVM)**

**Introduction**

Les machines à vecteurs de support (ou Support Vector Machine, SVM) sont une famille d’algorithmes d’apprentissage automatique de type supervisé et qui peuvent être utilisées pour des problèmes de discrimination (à quelle classe appartient un échantillon), de régression et de détection d’anomalies. Dans la suite nous nous intéresserons aux différents SVM de classification ainsi que de régression puis nous implémenterons un cas pratique au moyen de la bibliothèque d’apprentissage automatique Scikit-learn de Python.

**1. Principe de fonctionnement**

Les SVM sont une généralisation des classifieurs linéaires (algorithmes de classement statistique) dont le principe est de séparer les données en classe à l’aide d’une frontière, de telle façon que la distance entre les différents groupes de données et la frontière séparatrice soit maximale. Cette distance est appelée ***marge***. Et les données les plus proches de la frontière sont appelées ***vecteurs de support***.

Dans le schéma suivant, la frontière est la droite en rouge(**hyperplan**), les vecteurs de support sont les éléments les plus proches de la frontière entourée en vert. Finalement, la marge est la distance entre la droite en rouge et les deux droites en noire et en jaune.

**2. Avantages et limites**

* **Avantages**

Les classificateurs SVM sont d’une précision élevée et effectuent des prédictions plus rapides que d’autres modèles. Ils utilisent aussi moins de mémoire car ils utilisent un sous-ensemble de points d’entrainement dans la phase de décision.

* **Limites**

Les SVM ne sont pas adaptés aux grands ensembles de données puisqu’ils prennent un temps de formations plus élevé. Et ils fonctionnent mal avec les classes qui se chevauchent et dépendent du noyau utilisé.

**3. Cas d’utilisation**

Voici quelques exemples d’utilisation du modèle SVM :

Reconnaissances des formes :

* + Reconnaissance de chiffres manuscrits.
  + Reconnaissance de visages.

Diagnostic médical :

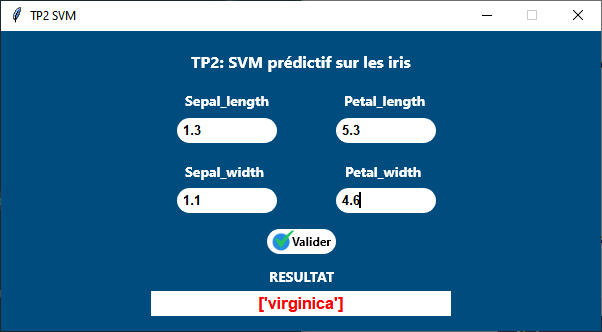
* Evaluation des risques de cancer.
* Détection d’arythmie cardiaque.

**4. Cas pratique (iris dataset)**

Ici, il est question de classifier les iris (setosa, versicolor, virginica) en fonction des propriétés sepal\_length, sepal\_width, petal\_length et petal\_width au moyen d’un modèle SVM.

**Les fichiers sources relatif à ce TP sont disponible dans le dossier SRC de ce repository.**

1. Le notebook **SVM.ipynb** contient toutes les étapes de construction commentées du modèle de classification.
2. **Programme.py** contient une utilisation du modèle pour la prédiction.
3. Un **.exe** est disponible dans le répertoire EXE de ce repository.
4. Un **MSI** est disponible dans le répertoire TEST de ce repository.
5. L’interface est très simple. Entrer les données de l’iris a prédire le ‘specie’ puis avec le bouton valider notre model SVM vous retournera un résultat.



**Conclusion**

Tout compte fait nous avons vu le SVM qui fait partie de la famille des algorithmes d’apprentissage automatique supervisés et qui permet à la fois la classification et la régression. Nous avons pour terminer, implémenté un modèle SVM capable de prédire le ‘specie’ d’une iris au moyen de la bibliothèque scikit-learn de Python. Il en ressort que le SVM demeure un modèle très indispensable au machine Learning.