

**TP1 Machine learning**

Apprentissage Supervisé

**Préparé par :** Hafed Benteftifa

© Hafed Benteftifa 2015-2019

Ce document ne peut être utilisé dans le cadre d’une formation, publication papier, site internet ou tout support sans mon accord express.

Aucune reproduction, même partielle, ne peut être faite de ce document et de l'ensemble de son contenu : textes, images, etc. sans mon autorisation express. Pour toutes informations, communiquer avec moi sur [info@degenio.com.](mailto:info@degenio.com)

|  |  |  |
| --- | --- | --- |
| **Date** | **Version** | **Changement** |
| 10 juin 2016 | 1.0 | Version initiale |
| 26 juin 2016 | 1.1 | Choix du langage R ou python |
| Mai 2019 | 1.2 | Langage python |

**Projet** : Catégorisation de texte

Contexte de réalisation

1. Format: individuel
2. Datasets: on a à notre disposition une source de données data\_texte.csv

**Objectif**

Ce projet a pour but d’évaluer votre compétence dans le développement et l’utilisation d’une approche supervisée.

Description du projet

L’entreprise a récemment développé un plan d’action pour améliorer son taux de classification de documents texte produits par ses différents services. Pour cela, il est proposé la mise en place d’un modèle permettant de catégoriser des documents texte.

Référence pour le dataset :

<http://mlg.ucd.ie/datasets/bbc.html> (BBCSport Fulltext)

Noter que dans ce dataset, on a des dossiers par catégorie/classes (athletics, cricket, football, rugby, tennis)

Démarche

L’objectif est de développer un modèle qui va prédire le **likelihood** qu’un document appartienne à une certaine catégorie.

Le modèle doit être développé sur un sous-ensemble de données qui vous sera fourni. Le modèle sera ensuite testé sur un sous-ensemble de données qui sera caché pendant la phase d’apprentissage. De ce fait, on testera la qualité du modèle sur des données qui ne font pas partie de l’ensemble d’apprentissage.

Facteurs d’évaluation

Trois facteurs seront utilisés, à savoir le Score F1, la taille de l’échantillon d’apprentissage et les ressources nécessaires pour l’apprentissage.

Score F1 : donne le sommaire des positives et négatives corrects sur l’ensemble des cas. Cela nous donne une mesure sur la différentiation du modèle, passage vs echec.

Taille de l’échantillon d’apprentissage : De préférence, on devra avoir un échantillon de petite taille au lieu d’un plus grand échantillon pour des considérations de logistique ou autre. De manière générale, quelle est la taille nécessaire pour que le modèle puisse être utilisé pour une prédiction adéquate?

Ressource de calcul : quelles sont les ressources nécessaires pour faire une prédiction adéquate? On parle ici du temps de traitement et de la mémoire nécessaire pour arriver au résultat demandé.

Tout autre métrique que vous pensez être approprié pour ce problème pourra être utilisé et évalué.

Langage et libraries à utiliser

* Python 3.5/Anaconda
* Numpy
* Sci-kit learn (Sklearn)
* Pandas
* Matplotlib/Seaborn

**Dataset et template**

* Voir le lien précédent pour le dataset

Livrable

1. Identifier l’approche à utiliser. Est-ce que l’on est dans un cas de classification? Expliquer.
2. Exploration des données :
   1. Déterminer les statistiques que vous pensez être intéressantes pour ce projet.
3. Préparation des données :
   1. Identifier les features/prédicteurs et la colonne cible
   2. Effectuer un ou des prétraitements sur les colonnes feature
   3. Partager les données en training et test
4. Modèle d’apprentissage et test évaluation
   1. Choisir les modèles d’apprentissage supervisé qui sont appropriés pour ce problème. On pourra utiliser scikit-learn pour l’implémentation.
   2. Indiquer la nature du modèle proposé en termes des avantages et des inconvénients
   3. Quelles sont les critères qui ont fait que vous avez choisi cette approche? Est-ce que c’est basé sur la nature des données que vous avez reçu?
   4. Pour chaque modèle, donner une table qui montre le temps nécessaire pour l’apprentissage, temps pour faire la prédiction, le score F1 sur le set training, le score F1 sur le test set.
      * 1. Choix du meilleur modèle : indiquer lequel serait le meilleur modèle pour les données que vous avez reçu. Indiquer vos critères tel que : données disponibles, ressources de calcul, cout, performance

**Challenges :**

Le premier problème à résoudre est de trouver les X features à utiliser pour la prédiction des classes. Dans notre cas, on recoit des fichiers au lieu d’enregistrements.

Quelques approches à évaluer pour transformer le texte dans un document en un espace de features :

* Word Count Vectors
* TF–IDF Vectors
* Word Embeddings
* Text based or NLP based features
* Latent Dirichlet Allocation
* BERT

**Important :**

Aucune approche d’apprentissage ne sera considérée pour ce projet.

Référence

<https://towardsdatascience.com/algorithms-for-text-classification-part-1-naive-bayes-3ff1d116fdd8>

<https://towardsdatascience.com/feature-selection-on-text-classification-1b86879f548e>

<https://www.analyticsvidhya.com/blog/2018/04/a-comprehensive-guide-to-understand-and-implement-text-classification-in-python/> (Intéressant pour le coté implémentation)