Rapport projet MLA – Régression logistique

Ce document présente le travail effectué par Maxime Dangelser et Nicolas Vanhoucke dans le cadre du cours de Machine Learning et Applications des données du Master 2 MIAGE parcours Business Intelligence de Polytech Lyon.  
Le sujet porte sur le développement d'un programme de régression en python.

# Régression logistique :

La régression logistique est une technique statistique et d'apprentissage automatique utilisée pour modéliser la probabilité d'une variable dépendante catégorielle, typiquement binaire. Elle est souvent utilisée dans les cas où l'on veut prédire la présence ou l'absence d'un caractère (comme oui/non, réussite/échec). Contrairement à la régression linéaire qui prédit une valeur continue, la régression logistique estime la probabilité qu'une observation appartienne à une classe spécifique. Elle fonctionne en appliquant la fonction logistique (sigmoid) aux prédictions de la régression linéaire, transformant ainsi les résultats en valeurs comprises entre 0 et 1. Cette approche est largement utilisée dans divers domaines comme la médecine, les sciences sociales et le marketing, offrant une base solide pour la prise de décision basée sur des probabilités

# Présentation du langage utilisé :

Lors de ce projet nous avons utilisé le langage de programmation Python.

Python est un langage de programmation interprété, à syntaxe claire et simple, et qui met l'accent sur la lisibilité du code. Il est facile à apprendre et à utiliser pour les débutants en programmation, mais aussi assez puissant pour être utilisé dans des projets complexes. Python est un langage polyvalent, utilisé dans une grande variété d'applications, notamment pour la science des données, l'apprentissage automatique, la création d'applications web, l'automatisation de tâches, les scripts système, les jeux et bien plus encore. Il a une grande communauté de développeurs qui ont créé une multitude de bibliothèques et de frameworks, tels que Pandas, Numpy, Matplotlib, Django, Flask, etc., qui facilitent le développement de projets de différentes tailles et complexités.  
En outre, Python est un langage open-source, ce qui signifie que tout le monde peut l'utiliser, le modifier et le distribuer librement. C'est un langage très populaire et largement utilisé dans l'industrie, l'éducation et la recherche.

## Présentation des bibliothèques utilisées :

Pour effectuer ce travail nous avons utilisé différentes bibliothèques présentées ci-dessous.

### Pandas :

Pandas est une bibliothèque open-source pour Python qui permet de manipuler et d'analyser des données tabulaires. Pandas fournit des fonctionnalités pour lire et écrire des données dans différents formats de fichiers, tels que CSV, Excel, JSON, SQL et bien plus encore. Elle permet également de nettoyer et de préparer les données pour l'analyse, notamment en gérant les valeurs manquantes et en effectuant des opérations de fusion, de regroupement et de filtrage.

### Streamlit :

Streamlit est une bibliothèque open-source de Python qui permet de créer facilement des applications web interactives à partir de scripts Python. Elle fournit une interface simple et intuitive pour créer des applications web en utilisant des commandes simples telles que "st.title", "st.write", "st.plot" et bien d'autres encore. Streamlit permet aux utilisateurs de créer des applications web interactives à partir de scripts Python en quelques minutes, sans avoir à apprendre des langages de programmation web tels que HTML, CSS ou JavaScript. Elle prend également en charge les graphiques interactifs, les cartes, les animations, les widgets, les formulaires et bien d'autres encore. Streamlit est très populaire dans la communauté des scientifiques des données, car elle permet de créer des tableaux de bord interactifs pour visualiser et explorer des données rapidement et facilement. Elle est également très utile pour la création de prototypes, la démonstration de concepts, la formation et l'enseignement.

### Matplotlib :

Matplotlib est une bibliothèque de visualisation de données en Python qui permet de créer des graphiques statiques.  
Matplotlib.pyplot, souvent abrégé en plt, est un module de Matplotlib qui fournit une interface de programmation similaire à celle de MATLAB pour créer des graphiques en utilisant des commandes simples.

### Scikit-learn (sklearn) :

Scikit-learn est une bibliothèque Python largement utilisée pour l'apprentissage automatique et l'analyse de données.  
Elle offre une large gamme d'algorithmes d'apprentissage automatique, de prétraitement des données, d'évaluation de modèles, et des outils pour la sélection de modèles.  
Scikit-learn facilite la création de modèles de régression, de classification, de clustering, et d'autres tâches courantes en apprentissage automatique grâce à une interface conviviale et bien documentée.  
Elle est souvent utilisée pour l'entraînement de modèles de machine learning, notamment la régression logistique, les forêts aléatoires, les machines à vecteurs de support, etc.

### Seaborn :

Seaborn, d'autre part, est une bibliothèque Python de visualisation de données basée sur Matplotlib.  
Elle simplifie la création de graphiques informatifs et esthétiquement plaisants en fournissant une interface de haut niveau pour la création de graphiques statistiques. \Seaborn offre un ensemble de styles par défaut et de palettes de couleurs qui améliorent la lisibilité des graphiques, ce qui en fait un choix populaire pour explorer et présenter des données.  
Elle prend en charge divers types de graphiques, notamment les diagrammes en barres, les diagrammes à dispersion, les diagrammes en boîte, les diagrammes de densité, et d'autres, ce qui en fait une bibliothèque précieuse pour l'analyse exploratoire des données.  
Ici, elle sera également utiliser pour récupérer les données (titanic, etc.)

# Structure de l’application :

## Organisation des fichiers :

Il y a 4 fichiers python :

* main.py : Ce fichier contient l’ensemble du code de l’application et c’est lui que l’on voit lorsqu’on démarre l’application.
* constant.py : Ce fichier contient les constantes que nous utilisons dans le code. Pour faciliter la modification en cas de besoin.
* text.py : Ce fichier contient le texte affiché dans l’application pour alléger le fichier main.py.
* code\_text.py : Ce fichier contient le texte du code affichés dans l’application via l’option « Montrer le code ». Également pour alléger le fichier main.py.
* requirements.txt : permet l’installation des librairies utilisées.

# Descriptif de l’application :

## Accueil :

Affiche le texte de présentation du projet.

## Lecture des données :

Présente les données récupérées et traités sous forme de table.

## Régression logistique :

Création du modèle de régression logistique et présentation du résultat sous forme de graphique.

Il est possible de choisir la valeur cible car nous pensions d’abord permettre d’ajouter n’importe quel fichier de données, cependant, nous nous sommes rendus compte que le travail ne serait pas aussi qualitatif qu’attendu, surtout pour le test avec un nouvel individu.

Une image contenant texte, capture d’écran, logiciel, Logiciel multimédia

Description générée automatiquement

## Tester un nouvel individu :

Cet onglet permet de tester différents paramètres et voir si la personne serait vivante ou morte.

Une image contenant texte, capture d’écran

Description générée automatiquement

# Décrire les possibilités d’hébergement web :

Nous avons fait le choix d’utiliser l’hébergement gratuit de Streamlit pour partager notre application.

Autrement, il faut un domaine et un serveur pouvant faire fonctionner des programmes python.