[**Chapitre 1 : Présentation du projet** 2](#_Toc182787602)

[I. Introduction 3](#_Toc182787603)

[II. Définition du Projet 3](#_Toc182787604)

[1. Motivation et objectif du projet 3](#_Toc182787605)

[1.1. Motivation 3](#_Toc182787606)

[1.2. Objectif du Projet 4](#_Toc182787607)

[2. Equipe du projet et Methode de travail 4](#_Toc182787608)

[III. Conclusion 5](#_Toc182787609)

[**Chapitre 2 : Présentation du travail** 6](#_Toc182787610)

[II. Les concepts techniques. 8](#_Toc182787611)

[1. Classification 8](#_Toc182787612)

[2. La régression 9](#_Toc182787613)

[3. Pipeline 9](#_Toc182787614)

[III. Les outils utilisés 10](#_Toc182787615)

[1. Python 10](#_Toc182787616)

[2. Pandas et Seaborn 11](#_Toc182787617)

[3. Scikit-learn 12](#_Toc182787618)

[4. Kaggle 13](#_Toc182787619)

[IV. Implémentation de la solution 13](#_Toc182787620)

[1. Compréhension du dataset 14](#_Toc182787621)

[2. Nettoyage & Prétraitement des données 15](#_Toc182787622)

[2.1. Data Cleaning 15](#_Toc182787623)

[2.2. Data Encoding 17](#_Toc182787624)

[3. Analyse exploratoire des données 17](#_Toc182787625)

[3.1. Analyser Uni-variable 17](#_Toc182787626)

[3.2. Analyse Bi-variables et Multi-variables 20](#_Toc182787627)

[4. Feature Engineering 21](#_Toc182787628)

[5. Modélisation et entrainement des modèles 22](#_Toc182787629)

[5.1. Entrainement du modèle de classification 23](#_Toc182787630)

[5.2. Modèle de régression 23](#_Toc182787631)

# **Chapitre 1 : Présentation du projet**

## Introduction

Dans ce premier chapitre de notre stage au sein de l’entreprise Technocolab, nous allons essayer de vous presenter notre projet. Nous allons définir notre projet pour bien savoir de quoi il était question de notre mission au sein de Technocolab. Ensuite, nous clarifier les motivations qui se cachent derrière notre présence au sein de Technocolab. Nous aussi vous donner notre objectif principal sur ce projet. En fin nous vous dirons quel mode de travail nous avons opter pour mener à bien cet objectif.

## Définition du Projet

### Motivation et objectif du projet

#### Motivation

Le **MSB** (Mortgage-Backed Security), ou **titre adossé à des créances hypothécaires**, est un type d’outil financier qui regroupe des prêts hypothécaires et les transforme en titres négociables sur les marchés financiers.

On peut retenir qu’un MSB est un moyen pour les institutions financières de transférer les risques hypothécaires aux investisseurs tout en offrant des opportunités de rendement, mais il s'accompagne également de défis liés à la gestion et à l'analyse des risques.

Etant donnée que Technocolabs est une entreprise de conception des solutions digitales. Elle a eu à travailler sur plusieurs projets allant dans ce sens avec des grosses boites. Ces projets sont de divers natures et vise à faciliter la migration vers la digitalisation de plusieurs entreprises.

C’est donc dans cette optique que l’entreprise **Technocolabs** s’est donnée comme objectif de mettre en place un outil basé sur l’Apprentissage Automatique (Machine Learning) pour faciliter la gestion et les risques liés au MSB.

Nous sommes conscients qu’un tel travail demande beaucoup de précision et de rigueur. Mais nous avons pu relever ce défi avec l’entreprise d’accueil, Technocolabs.

#### Objectif du Projet

Notre objectif principal ici dans ce projet est de prédire le risque de remboursement anticipé des MSB en utilisant des modèles d'apprentissage automatique. Nous voulons faciliter la tâche aux investisseurs et aussi aux instituts financiers la prédiction des risques de remboursement en mettant en place un outil IA qui les permettra de prédire facilement le risque de remboursement en juste introduisant les données de l’intéressé.

En effet les anciennes méthodes utilisées pour prédire les risques de remboursement sont dépassées de nos jours et peuvent mettre la productivité des intéressés en baisse. Donc la réflexion est de trouver une bonne solution, optimale, précise pour accomplir cette tâche.

De nos jours la réflexion est beaucoup plus axée sur les outils d’apprentissage Automatique. Et nous nous donnons comme objectif d’appliquer ces outils dans un domaine aussi précieux que soit-il.

En résumé, notre motivation est de relever les défis actuels en matière de prédiction des risques de remboursement du MSB avec l’entreprise Technocolabs. Nos motivations nous amènent à se fixer comme objectif la création d’une solution performante, précise et conviviale, offrant des fonctionnalités de prédictions.

### Equipe du projet et Methode de travail

Le projet a été par une équipe de 05 personnes étant donné que nous sommes tous stagiaires dans la meme entreprise et bossons sur le meme projet. Nous avons un encadrant qui faisait le suivi du développement du projet. En dehors de l’encadrant de l’entreprise, nous nous sommes organisés à trouver un « Team lead » qui dirigeait l’équipe des 05 personnes en dehors des réunions avec l’encadrant.

Concernant la methode de travail, nous avons fait le stage en ligne entreprise étant basée en Inde. De ce fait nous avons élaborer un plan de travail a distance avec l’encadrant. Nous travaillons en ligne, en organisant des réunions sur Zoom.

Pour une bonne organisation du travail, le chef de l’entreprise a décidé d’organiser des réunions sur Zoom chaque 03 jours donc 02 fois par semaines pour l’état d’avancement du développement. Lors de ces réunions chaque stagiaire avait droit a 20minutes maximales pour expliquer le travail effectué durant les 03 jours.

En dehors des réunions que nous organisons chaque 03 jours, il y avait des réunions que nous organisons avec le Team lead pour voir es ce que tout le monde a pu effectuer la tâche demandée. De ce fait on fait la reunion entre nous les 05 stagiaires chaque 02 jours, donc la veille avant d’aller voir l’encadrant.

Pour le développement du produit, nous avons plutôt opter pour un itératif. Nous développons partie par partie. Si une partie est finie, elle est testée et presenter a l’encadrant avant de passer à l’étape suivante. Ce qui nous permettait de faire voir des solutions concrètes a l’entreprise.

Le Travail était aussi chaque fois adapter au changement si nécessaire. Vu qu’on avait opter pour le développement itératif, il est tres facile d’adapter le travail aux changement demandes et exiges par l’entreprise.

Les canaux de communications utilisés étaient globalement les outils comme Slack, WhatsApp, Zoom. Aussi nous recevons des emails de l’entreprise pour nous inviter a une reunion ou nous faire part des changement de dernières minutes.

## Conclusion

Dans ce premier chapitre de nous stage, nous avons essayer de faire le point très bref de notre objectif au sein de Technocolabs et le travail qui nous a été confier egalement. Nous avons egalement fait sortir le mode de travail utilisé lors de notre stage au sein de l’entreprise Technocolab.

Dans la suite de notre travail nous allons presenter notre travail, comment nous avons procéder pour atteindre l’objectif que nous nous sommes fixés depuis ce chapitre. Aussi nous vous ferons parts des outils utiliser et des méthodes préconisées pour pouvoir arriver à la réalisation de notre solution de prédiction des risques de remboursement des prêts hypothécaires.

# **Chapitre 2 : Présentation du travail**

1. Introduction

Dans ce chapitre nous allons commencer par une étude approfondie qui permettra d’évaluer les aspects techniques de notre projet. Cette analyse nous permettra de voir en comment notre solution est faisable, en identifiant les obstacles et les points forts des méthodes et technologies choisies. Aussi nous ferons un parcours de comment nous avons pu réaliser le projet.

## Les concepts techniques.

Dans notre projet, nous allons besoin de deux concepts clés en Machine Learning. Il s’agit de la Classification et la Régression.

### Classification

La **Classification** en Machine Learning est une tâche qui consiste à attribuer une **catégorie** ou une **étiquette** à un ensemble de données en fonction de leurs **caractéristiques**. C'est un type d'apprentissage supervisé, où le modèle apprend à partir de données étiquetées pour effectuer des prédictions sur de nouvelles données. Il s’agit là de distinguer entre deux classes ou plusieurs en se basant sur leurs caractéristiques.

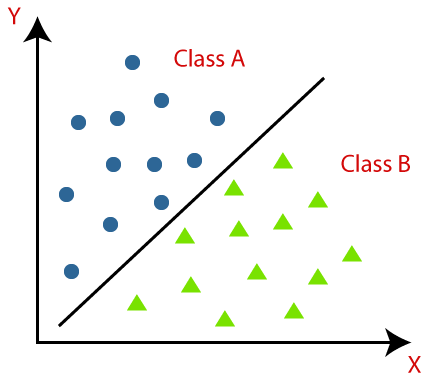


Figure 21 : Illustration de la classification

Dans notre projet, dans notre pipeline nous aurons besoin de mettre en place un classifier qui se chargera de nous faire la classification de deux situations. Il s’agit donc de la classification binaire. En effet il ya pratiquement deux types de classification : une classification multiclasses et une classification binaire (02 classes).La difference entre ces deux est très claire.

Dans un problème à deux classes on fera la classification binaire. Il s’agira de classer les donner entre deux instances de classes. Par contre pour la classification mutliclasses, on fait la classification entre plus de 02 classes. On classe donc les donnees entre plus de deux instances de classes.

Dans notre cas il sera question de classer les emprunteurs entre deux classes de la variable « EverDeliquent » . Ceci nous permettra de voir si l’emprunteur a connu un defaut de paiement historique ou non. Cette decision est très importante dans le domaine de prediction du risque de remboursement du MSB. Car si un emprunteur est EverDeliquent c’est de la meme manière qu’il sera traiter par rapport a un emprunteur qui n’est pas EverDeliquent.

### La régression

La **régression** en **machine Learning (ML)** est une tâche d'apprentissage supervisé où l'objectif est de prédire une **valeur continue** à partir de données d'entrée. Contrairement à la **classification**, qui attribue des étiquettes ou des catégories à des instances, la régression prédit une quantité numérique. L'idée est de trouver la relation ou la fonction qui relie les variables indépendantes (caractéristiques ou prédicteurs) à une variable dépendante continue (la variable cible ou de sortie). Par exemple, prédire le prix d'une maison en fonction de sa taille, de son emplacement.

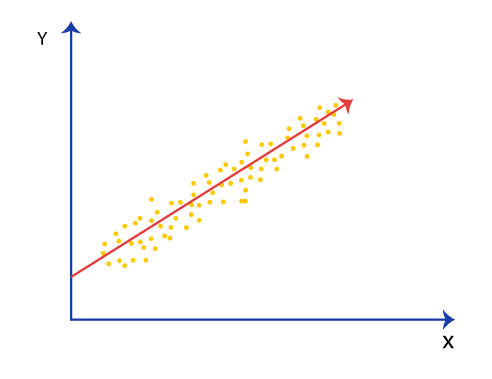


Figure 22 : Illustration de la régression

Dans le pipeline de notre projet, nous allons utiliser un régresser pour prédire la valeur à rembourser une fois que l’emprunteur est détecté comme EverDeliquent. Pour la régression il y a aussi une variété de modèles à utiliser en fonction de la situation, du probleme. Ici nous allons utiliser un régresser linéaire.

### Pipeline

Un **pipeline** en **Machine Learning (ML)** fait référence à une série d'étapes ou de processus automatisés permettant de transformer des données brutes en un modèle entraîné, puis d'utiliser ce modèle pour effectuer des prédictions. Il représente l'ensemble des flux de données et des opérations nécessaires pour entraîner, évaluer et déployer un modèle de machine Learning de manière systématique et reproductible. Ce concept est utilisé pour la mise en place des solution den ML en production.

En effet vu que chaque modèle a un « bouchon » qui sont ces entrées et sorties alors il est plus recommandé de mettre en place une structure qui permettra de prendre les données brutes de l’utilisateur et de les transformer a ce que notre modèle attend en entrée. Sinon nous pouvons aussi le faire sans pipeline. Mais si nous avons le pipeline, c’est pour la productivité, la réutilisation facile et adaptée.

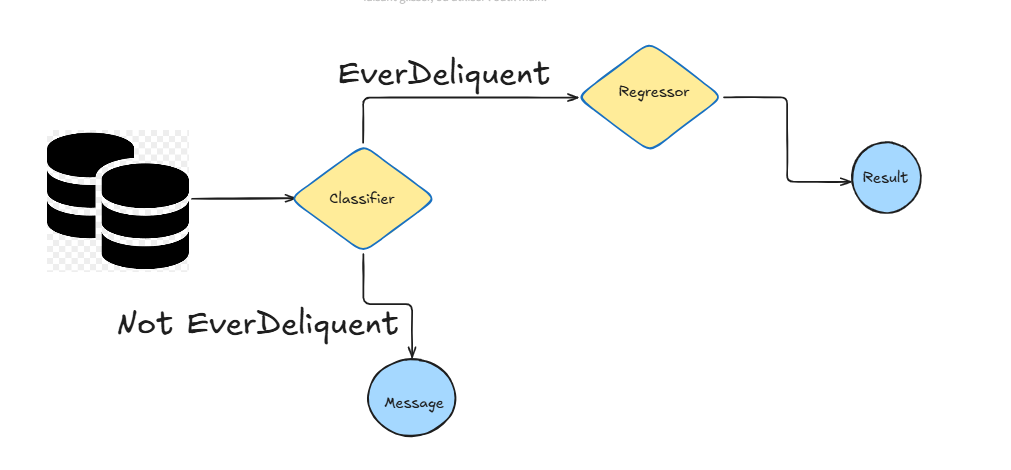


Figure23 : Pipeline du modèle

Comme illustrer sur notre pipeline **Figure 23**, nous allons juste prédire le risque de remboursement lorsque l’emprunteur a connu un défaut historique de paiement.

## Les outils utilisés

Dans un projet machine Learning, certains outils sont une obligation et d’autre par choix. Ici nous allons citer la liste des outils qu’on a utiliser pour notre projet et aussi la raison de ces choix s’il est nécessaire.

### Python

Python est un langage de programmation de haut niveau avec une flexibilité du jamais égalé et facile à apprendre et à maitriser. Il est largement utilisé dans le domaine de IA et la Data Science car il nous donne de multitudes librairies et modules faciles à utiliser. Dans notre projet, Python est le premier outil que nous avons choisi. Ce choix s’explique par le fait que de nos jours quand on parle de IA et ML on voit en premier Python.

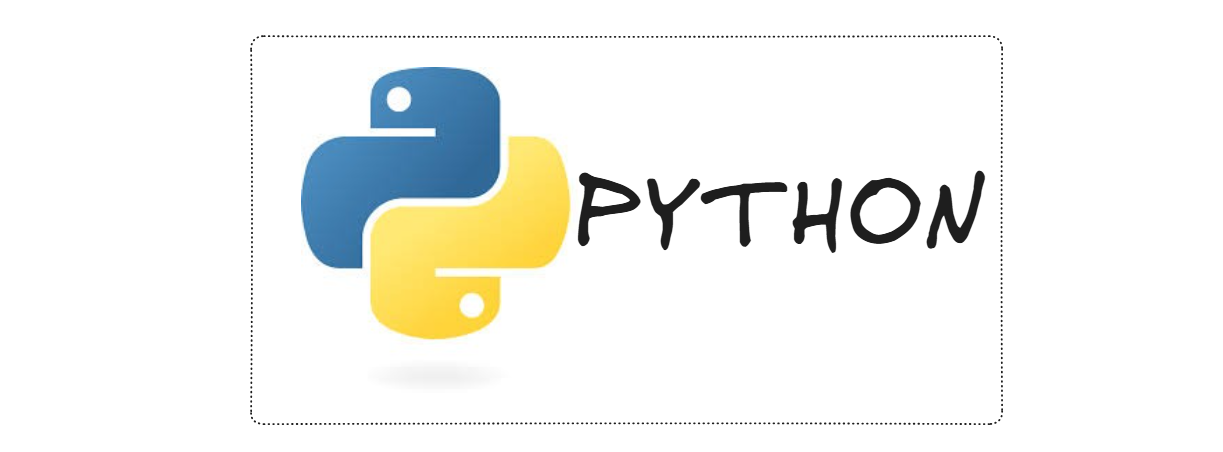


Figure24 : Python

### Pandas et Seaborn

**Pandas** est une bibliothèque écrite pour le langage de programmation Python permettant la manipulation et l'analyse de données. Elle propose en particulier des structures de données et des opérations de manipulation de tableaux numériques et de séries temporelles.

**Seaborn** est une bibliothèque Python de visualisation de données, construite sur **Matplotlib**, qui facilite la création de graphiques complexes et esthétiques avec un code plus concis. Elle est particulièrement utilisée pour l'analyse exploratoire des données et offre des outils puissants pour générer des graphiques statistiques.

A la difference de Pandas, Seaborn permet de créer des graphiques plus complexes et plus esthétiques. Ce qui est essentiels pour la visualisation des données.



Figure25 : Pandas& Seaborn

### Scikit-learn

**Scikit-learn (sklearn)** est une bibliothèque Python très populaire pour le machine Learning. Elle offre une large gamme d'outils pour la classification, la régression, le clustering, la réduction de dimension, la sélection de modèles, et bien plus encore. Elle est principalement utilisée pour travailler avec des données tabulaires et est très accessible grâce à une API simple et cohérente.

Dans notre projet, nous avons décidé d’utiliser sklearn car elle est facile à accéder, elle est open source. En plus de son accessibilité, elle est offre beaucoup de modèle dont nous avons besoin et meme un package pour créer facilement un Pipeline.



Fugure26 : Sklearn

### Kaggle

**Kaggle** est une plateforme en ligne très populaire qui offre des ressources et des outils pour la science des données, le machine Learning, et l'intelligence artificielle. Elle est principalement connue pour ses compétitions où les participants peuvent travailler sur des défis réels en utilisant des ensembles de données fournis par des entreprises ou des organisations. Kaggle est également un excellent endroit pour apprendre, partager des connaissances et collaborer sur des projets de données. Nous n’avons pas utiliser kaggle pour faire un competition ou pour apprendre, nous l’avons plutôt pour avoir accès a un accélérateur pour pouvoir accélérer l’entrainement de nos modèles. Vu que nous ne disposons d’assez de ressource en local, nous avons donc fait fait recours a une plateforme en ligne. Il n’y a pas que kaggle, il aussi Colab, qui est aussi tres populaire et offre d’enorme ressource pour faciliter l’entrainement des modeles un peu gourmants en termes de ressources.

Nous avons decider de choisir Kaggle et non pas Colab dans ce projet, car colab entraine souvent des micro-coupure et aussi enregistrement est en retard par rapport a votre avancement.



Figure27 : Kaggle

## Implémentation de la solution

### Compréhension du dataset

Dans cette etape de notre projet nous allons faire notre premiere prise en cintact avec le dataset. Cette etape est tres importante dans toute projet data science. Elle nous permet de bien cerner la signification de chque variables et bien mettre en place l’idée du projet.

Notre dataset est obtenu depuis le site officiel de Freddie Mac pour les prêts immobiliers. Il contient **291 452 lignes** de données et **28** **colonnes** ou variables. Donc le dataset est moyennement volumineux pour faire des modèles de ML et prédire des risques de remboursement du MSB.

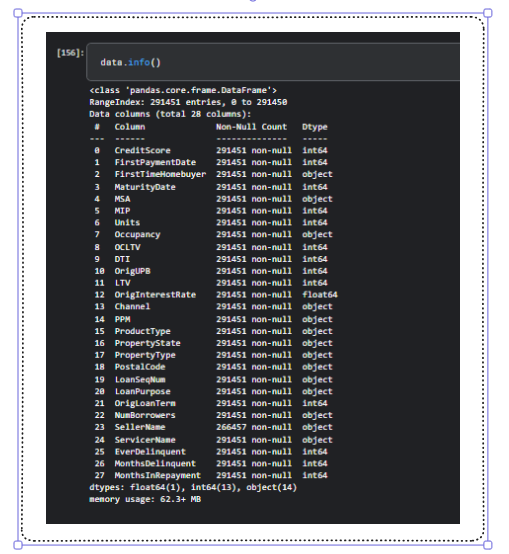


Figure28 : La structure des données

Ici nous voyons a peu pres ceux a quoi ressemblent les colonnes de notre dataset et aussi les types qu’elles ont. Cela nous donne une idée mais tres gobale sur les donnees dont nous avons a faire. Nous allons donc besoin de cette premiere idee qui fait ressortir les types des donneeset les noms des colonnes pour les traitements des donnees.

### Nettoyage & Prétraitement des données

Il est tres important de marquer l’importance de cette etape dans un projet data Science. Dans cette etape nous travaillons a rendre les donnees parlantes, utiles. C’est le role d’un data scieniste dans un projet. Cette partie a pris au moins 80% de notre de stage. En effet comme on aime le dire , Garbage In Garbage Out. La precision de notre model depend de la qualite de donnees qu’on lui donne. Si on donne des mauvaise donnees a un modele, on peut que s’attendre a une sortie catastrophique, a des mauvaises performances. Alors il était necessaire de bien traiter les donnees, bien les comprendre en profondaire, bien etudier leur repartition afin de passer a l’entrainement des modeles. Nous avons fait etape en des differentes parties.

#### Data Cleaning

Dans cette etapes nous allons travailler a bien nettoyer les donnees. Car les donnees recues sont brutes et viennent d’une base de donnees et aucun traitement n’a été fait auparavant.

Nous allons commencer tout d’abord a voir les valeurs manquantes « missing values ». Ici dans notre cas il est tres important de savoir qu’une valeur remplie ne veut pas forcement dire qu’elle est non nulles. Quand on essaie de voir les valeurs non null, voila ce sur quoi on tombe directement. 

Figure29 : Missing values

Nous voyons clairement que nous n’avons pratiquement pas de missing values selon cette manière de faire. Mais il est important de bien lire la description des variables. C’est donc dans cette logique qu’on a trouver avec l’entreprise un fichier qui nous décrit les variables et leurs compositions. En lisant ce fichier on découvre que notre dataset est bien plein des valeurs manquantes, mais toutes les colonnes sont pratiquement remplies. Nous allons donc mettre en place une methode précieuse pour extraire ces valeurs manquantes en tenant compte de la description des variables. Cette methode parcours toutes les colonnes te fait ressortir des valeurs non attendues.



Figure210 : Summary missing values

Comme il est affiché dans la **Figure230**, nous retenons que certaines colonnes sont remplies par exemple par « 9 » et dans cette meme colonne, le « 9=Not available ».

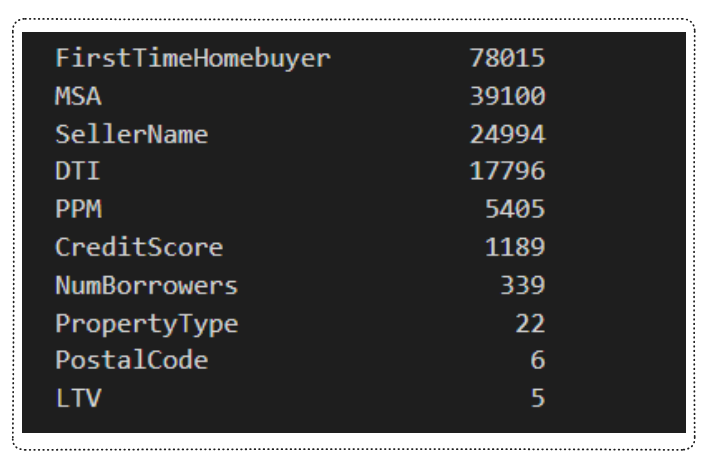
Donc nous avons essayer de faire sortir ces valeurs non attendues et en fin nous avons obtenue un nombre important des missing values. 

Figure211 : Final Missing values

Nous remarquons ici que nous avons maintenant un nombre important de valeurs manquantes dans les quelques colonnes.

Nous allons donc procéder à traiter ces valeurs manquantes en utilisant différentes méthodes du traitement des valeur manquantes.

Vu l’importance de ces valeurs manquantes dans notre dataset, nous n’allons pas supprimer les valeurs. Nous allons donc décider d’amputer certaines par la moyenne et d’autre par le mode. Ce choix dépend de la statistique de ces variables. Pour les valeurs numériques continue et dont ma distribution est près ce que normale, nous allons donc les amputer par la **moyenne**. Et pour les valeurs catégorielles, nous allons donc les amputer par le **mode**.

Une fois ces traitements faits, nous avons maintenant des données qui ne contient pratiquement pas de valeurs manquantes.

#### Data Encoding

Après avoir bien traiter les données manquantes, nous allons essayer d’encoder les données. En effet le **data encoding** (ou **encodage des données**) est le processus de **conversion des données** brutes en une représentation compréhensible pour les algorithmes de machine Learning. Il est essentiel, car la plupart des algorithmes ne peuvent traiter directement que des données **numériques**, tandis que les données brutes contiennent souvent des valeurs **catégoriques** (textuelles) ou des informations sous d'autres formats. Cet encodage dépend des variables. Y a des variables qui sont regroupées en deux groupes et d’autres plus que deux.

### Analyse exploratoire des données

Cette partie a été la deuxieme partie de notre travail durant notre stage. Dans cette etape du projet datascience, nous avons maintenant les donnees propres et donc nous pouvons les analyser, afficher les graphiques, et voir donc la relation qui existe entre les variables.

Nous avons essayer de la description complete de notre dataset avec la methode describe() des Dataframes de Pandas. Pour cette partie de l’analyse exploratoire, nous proceder en differentes sous-parties.

#### Analyser Uni-variable

L’analyse uni variable, nous allons analyser chaque variable de notre dataset, seule. Cette est tres important dans un projet Data Science, car elle nous permet de rentrer en détail sur les variables en faisant des analyses descriptives avec des graphiques pour mieux comprendre la variable dans le dataset. La variable est analysée seule, et cela nous permet d’avoir une idée sur la distribution de la variable. Ici nous allons analyser les variables sans suivre un ordre préétabli.

##### Variables Binaires

Commençons par la variable « **EverDeliquent** ». Cette variable est binaire c’est-à-dire que soit l’emprunteur est EverDeliquent ou non. Donc fera des graphiques pour voir la part de chaque modalité dans cette variable.

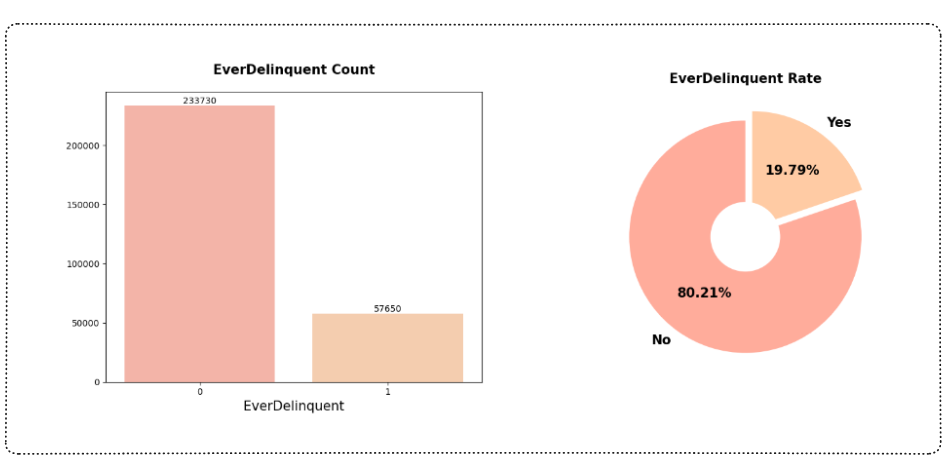


Figure212 : Distribution de EverDeliquent

Dans ces graphiques nous voyons une repartition des modalites de cette variable. D’apres cette repartition nous retenons que :

* 19,79% des empruteurs ont un defaut historique de remboursement et 80,21% ne l’ont pas pas.
* Cela nous permet de dire que la plupart des empruteurs remboursent leur prêts à temps.

Ensuit nous analysons la colone « **IsFirstTimeHomebuyer ».** Cette variable decrit si un empruteur est a sa premiere fois de payer une maison ou non. Donc c’est aussi une variable binaire.

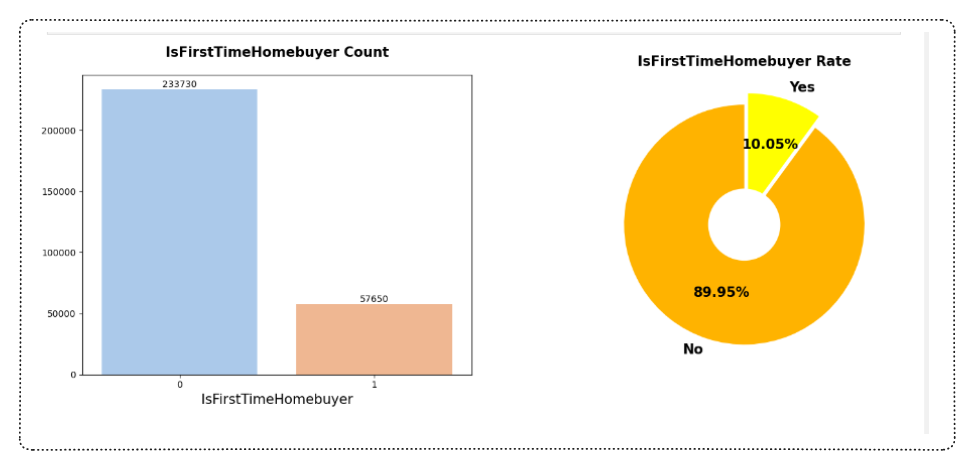


Figure213 : Distribution de la variable IsFirstTimeHomebuyer

Nous retenons ici que selon la variable IsFirstTimeHomebuyer

* La plupart des emprunteurs ne sont pas a leur premiere fois de payer une maison.

##### Les variables categorielles

Nous allons faire le point de l’analyse de quelque variables catégorielles.

La variable « **LoanPurpose** », décrit le but, l’objet du prêt chez les emprunteurs. Il a trois modalités : Achat(P), Refinancement sans retrait de liquidités(N) et Refinancement avec retrait de liquidités(C).

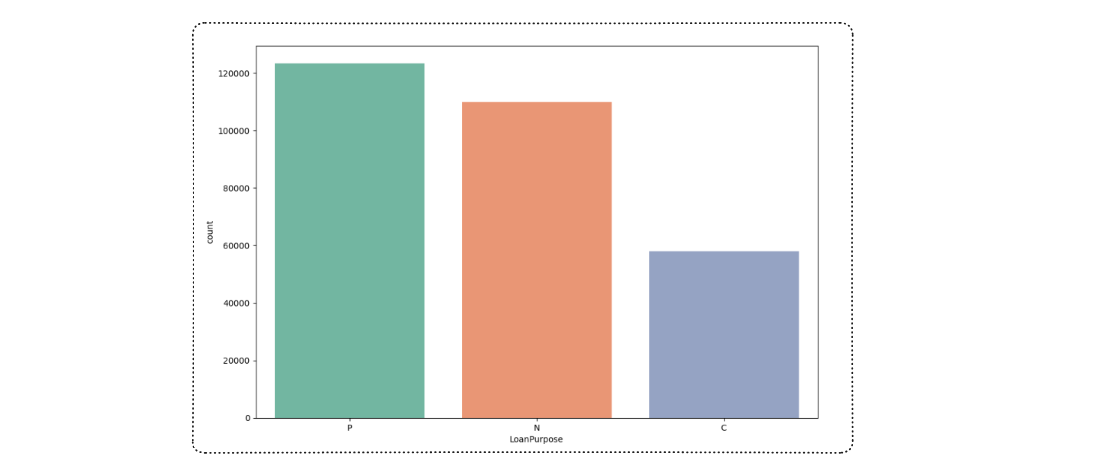


Figure214 : Distribution de LoanPurspose

Il ressort de ce graphique que la majorité, emprunts dans le but d’acheter.

La variable « **RepayRange** », elle nous donne une idée sur l’intervalle dans lequel l’emprunteur rembourse le prêt.

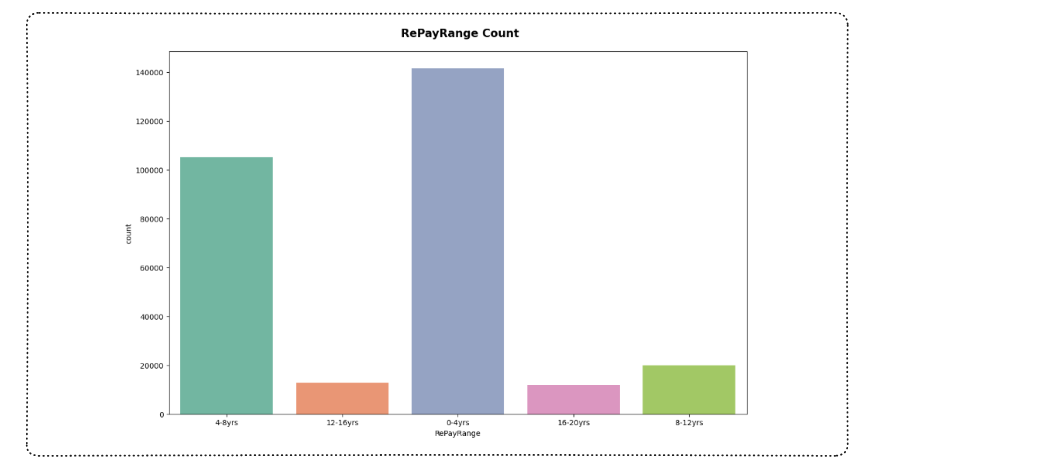


Figure215 : Distribution de la variables RepayRange

Nous remarquons ici que la majorité des emprunteurs remboursent leur prêt entre 0-4 ans.

* + 1. Variables numériques

Pour les variables numeriques egalement nous allons essayer d’afficher les densites de quelques variables ici. De ce graphique nous allons donc tirer des conclusions sur la distribution de ces quelques variables categorielles.

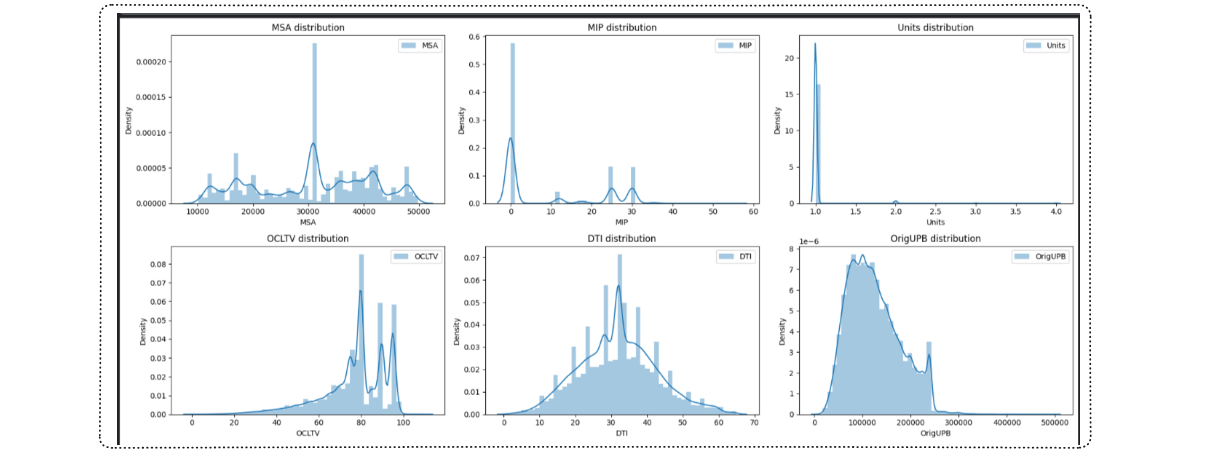


Figure216 : Figure distribution des variables numériques

Nous tirons ces conclusions suivantes :

* **DTI et OrigUPB** suivent une distribution presque normale.
* De nombreux histogrammes ont une queue lourde : ils s'étendent beaucoup plus à droite ou à gauche de la médiane.
* **OCLTV, LTV** et **OrigLoanTerm** présentent une asymétrie à gauche.
* **MonthsDelinquent** et **Units** présentent une asymétrie à droite.
* Il y a presque des valeurs aberrantes dans certaines colonnes de notre ensemble de données.
* Seules les colonnes **MSA** et **MIP** n'ont aucune valeur aberrante.
* La majorité (50 %) des **Credit\_range** sont dans la catégorie *Good*.
* La majorité (93 %) des **LTV\_range** sont dans la catégorie *High*.

#### Analyse Bi-variables et Multi-variables

Dans cette partie de notre travail, nous avons analyser les variables deux a deux ou plus. Nous avons essayé de voir la corrélation entre les variables deux a deux ou plus. Cela est très important pour nous de savoir comment sont les relations entre les variables. Pour bien mener a notre analyser, nous sommes confronter à un probleme. Il va falloir trouver un moyen pour analyser les variables de types différents. Nous avons donc utilisé le test d’ANOVA. Et pour les variables numériques, nous avons fait des tests des Pearson.  
En effet, le test **ANOVA** (*Analysis of Variance*) est une méthode statistique utilisée pour comparer les moyennes des groupes afin de déterminer s'il existe une différence significative entre ces groupes. Il est particulièrement utile lorsqu'on souhaite tester l'impact d'une ou plusieurs variables indépendantes catégoriques sur une variable dépendante continue.

Et le **test de Pearson** (ou **corrélation de Pearson**) est une méthode statistique utilisée pour mesurer la **relation linéaire** entre deux variables quantitatives. Il évalue à quel point les valeurs de deux variables sont liées par une relation linéaire.

En plus de ces tests, nous avons aussi besoin d’un visuel de ces corrélations. Nous avons donc fait un graphe afin de mieux comprendre les corrélations entre les variables.

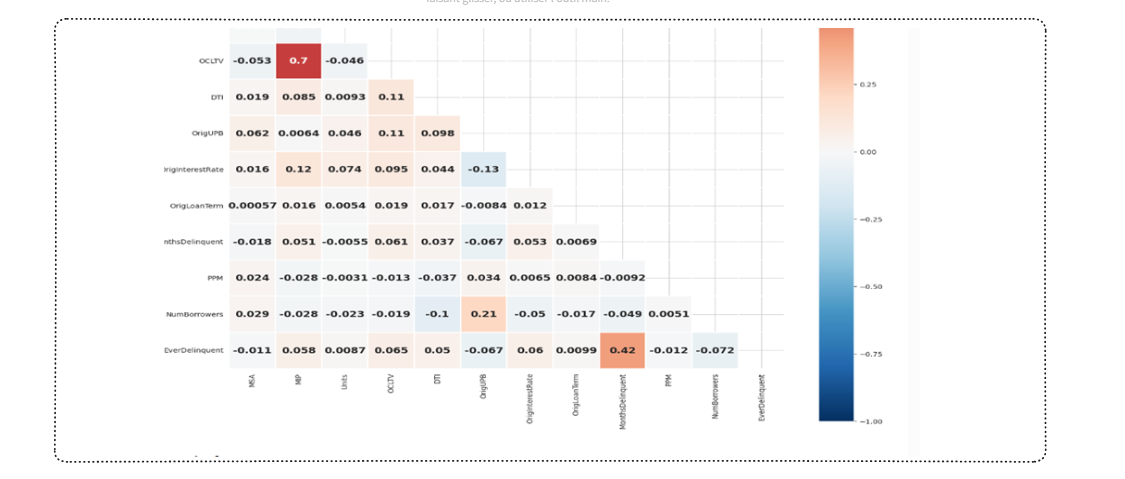


Figure217 : Matrice de corrélation

Ces analyses nous permettent de tirer les conclusions suivantes :

* Corrélations positives fortes : OCLTV-MIP, LTV-MIP, LTV-OCLTV. Il y a une corrélation forte et positive entre ces variables. Cela signifie que lorsque l'une de ces variables augmente, l'autre a tendance à augmenter également de manière significative.
* Corrélations positives faibles : EverDelinquent-MonthsDelinquent. Autrement dit, si un emprunteur a été en défaut de paiement à un moment donné (**EverDelinquent**), il est plus probable qu'il ait un certain nombre de mois en retard (**MonthsDelinquent**). Cependant, cette relation n'est pas aussi forte que dans le cas des corrélations positives fortes.
* Corrélations négatives : NumBorrowers-DTI, OrigUPB-OrigInterestRate... Une corrélation négative entre NumBorrowers-DTI, suggère que, à mesure que le nombre d'emprunteurs augmente, le ratio d'endettement sur le revenu diminue. Cela pourrait signifier que plus il y a d'emprunteurs, plus les revenus combinés augmentent, réduisant ainsi la proportion de la dette par rapport au revenu.

### Feature Engineering

Le feature engineering est une étape tres cruciales dans un processus de modélisation machine Learning. En effet une qu’on a bien compris nos données en profondeur, on a bien cerner les corrélations entre les variables, il est important maintenant de sélectionner les meilleures features, ou de créer les features.

En premier nous avons sélectionner les meilleurs features pour notre modèle de classification en utilisant « EverDeliquent » comme target variable. Pour la sélection des meilleurs features, nous avons utilisé un outil de Scikit-learn pour le Mutuelle Score Information. En effet MSI mesure la dépendance entre deux variables. Elle quantifie la quantité d'information que l'une de ces variables apporte à l'autre. En d'autres termes, elle mesure dans quelle mesure la connaissance d'une variable réduite l'incertitude concernant l'autre. A retenir que ce n’est pas la seule methode pour faire la sélection des meilleures features, mais avons opter pour cette solution car elle est populaire, facile a mettre en place et surtout facile à interpréter.

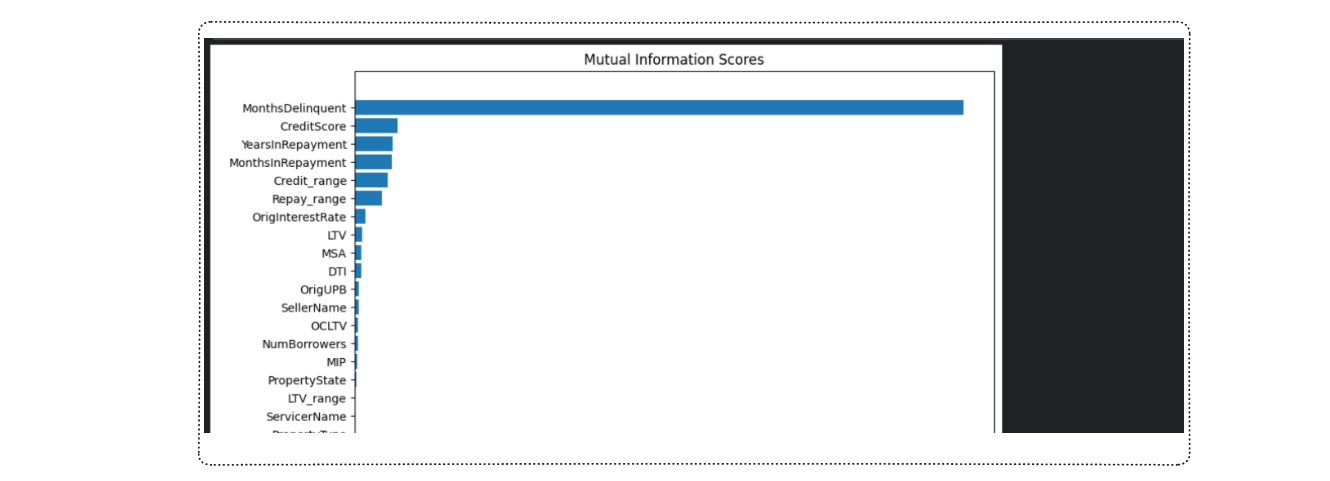


Figure218 : Best Features

Comme nous pouvons le remarquer, nous avons donc retenue les meilleures variables de notre dataset et arranger par ordre d’importance pour ce problème de classification. Donc nous aurons juste besoin de ces variables pour pouvoir prédire la classe d’un emprunteur selon la variable « EverDeliquent ».

Ensuite, pour prédire la valeur du risque de remboursement, nous allons utiliser un modèle de régression linéaire. Pour cela nous devons créer un autre feature. La formule pour créer cet feature nous a été donnee par l’entreprise. A travers cette formule nous avons créer une nouvelle variable pour en faire notre target variable pour la régression.

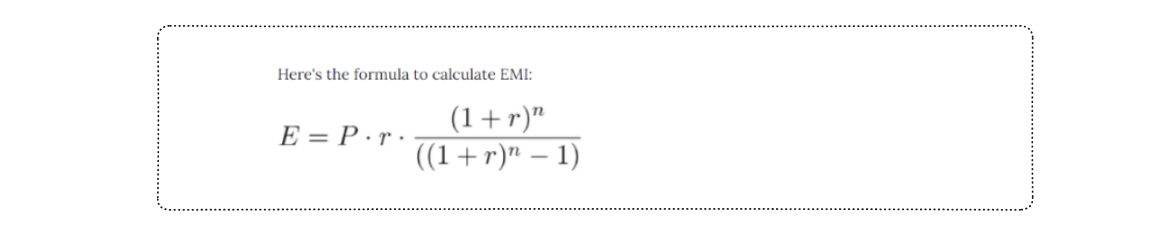


Figure219 : Formule pour EMI

### Modélisation et entrainement des modèles

Dans cette partie nous sommes à l’entrainement des modèles. Il est important de noter que les modèles que nous avons entrainé ont été imposé par notre encadrant. Donc il nous a donnee quelque modèle de classification et de régression d’entrainer et de sélectionner le meilleur modèle en se basant sur les indicateurs de performances comme la Précision, le F1-Score, le recall. Toutes ces indicateurs de performances dépendent des prédictions du modèle. Il y a donc des termes qui ressortent généralement comme :

Le FP : Faux positif, pour dire que le modèle a prédit Faux alors que l’individu est Vrai.

FN : le Faux Négative, le modèle prédit Faux et l’individu est réellement Faux

Le TP : le Vrai positive, le modèle prédit Vrai et que l’individu est Vrai.

Le TN : Vrai Négative, le modèle prédit Vrai alors que l’individu est Faux.

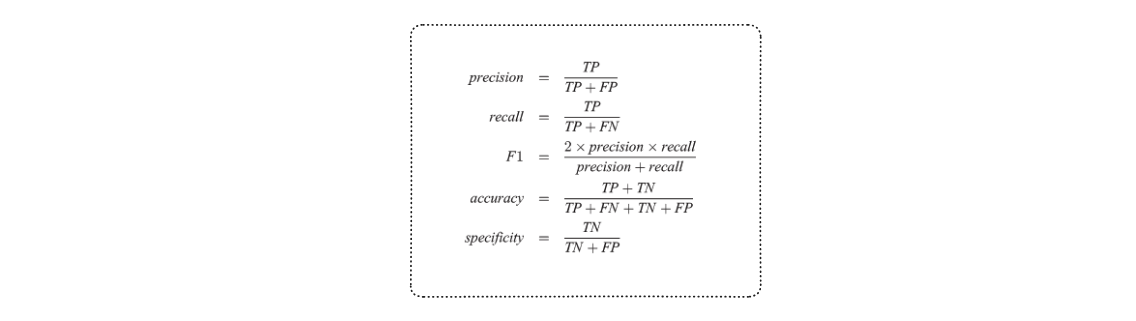


Figure220 : Indicateur de performance d’un classifier

#### Entrainement du modèle de classification

Nous avons entrainer plusieurs modèles comme le DecisionTreeClassifier c(), GaussianNB (),SVC (),LightGBMClassifier (),LogisticRegression ().

Apres entrainement de ces modèles on trouves que c’est le LogisticRegression () qui nous donnes les meilleures performances. Ce model nous permet d’avoir un Accuracy de 100%.

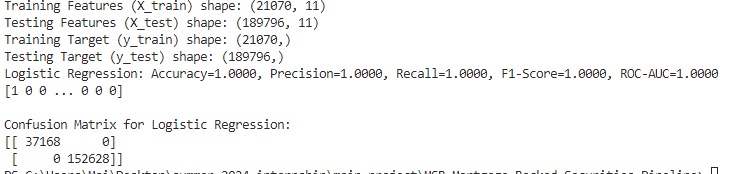


Figure221 : Performance du modèle de classification

Ces résultats paraissent attirer l’attention d’un overfiting, mais on a bien verifier sur les données de tests et aussi les données d’entrainement, le modèle nous donnait une précision de 100%. Il est donc a retenir qu’il n’y a pas un cas de surapprentissage, c’est rare de trouver une précision de 100% mais c’est possible.

#### Modèle de régression

De même que pour le modèle de classification, nous avons reçu un certain nombre de modèle venant de notre maître de stage alors ce sont ces modèles on a donc entrainer. Parmi ces modèles on avait : LinearRegression (), Ridge (). Parmi ces deux modèles c’est le modèle de LinearRegression qui nous donne des meilleures performances.

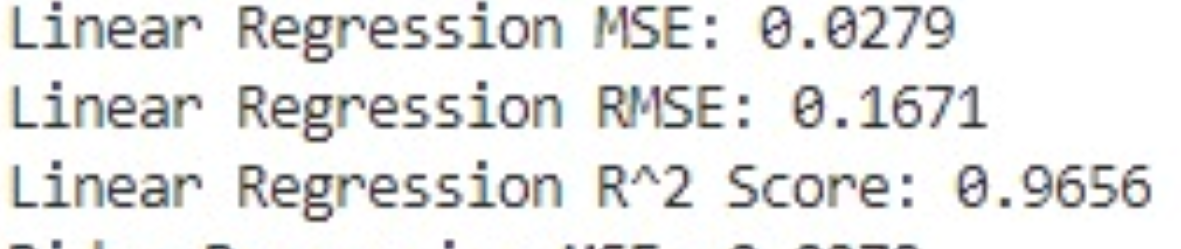


Figure223 : Performance du modèle de régression