Rapport de Projet Machine Learning : Prédiction d’Octroi de Crédit Bancaire

# 

# Problématique

L’objectif de ce projet est de prédire si un client est éligible à un crédit bancaire à partir de ses informations personnelles et financières. Cela permettra à la banque de prendre des décisions plus rapides et plus efficaces dans l’octroi de crédits.

### ****Introduction****

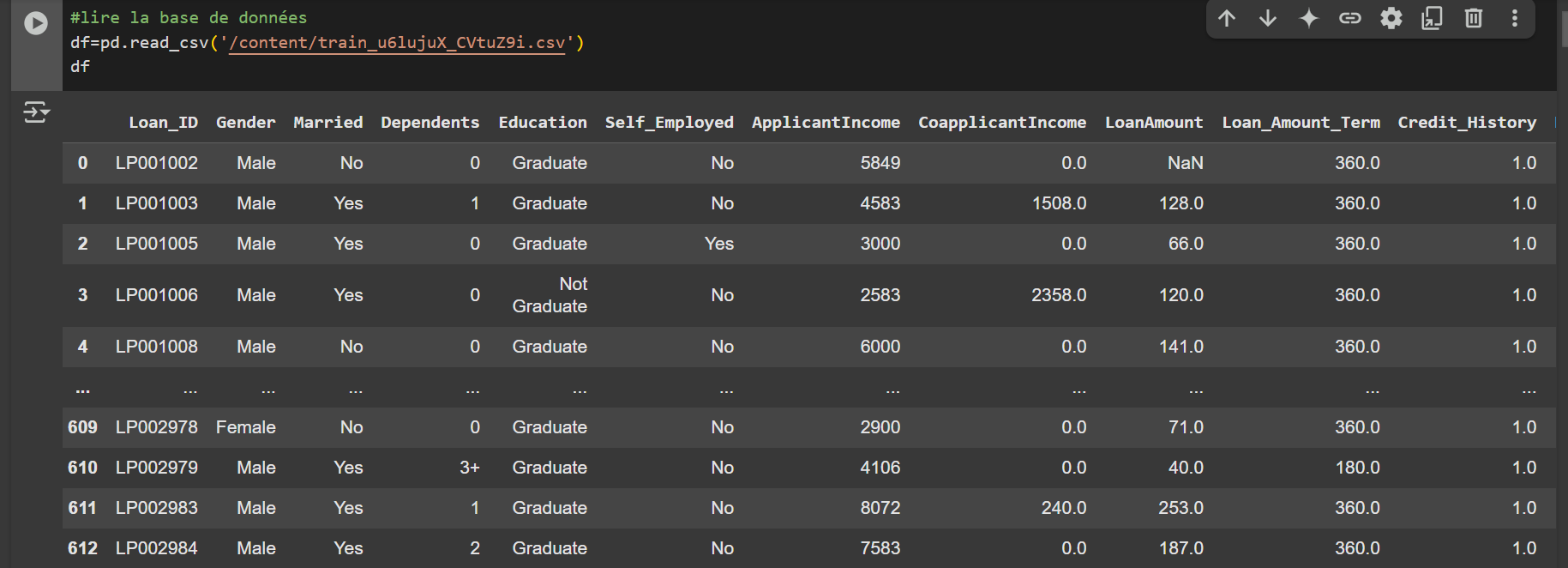
L’octroi de crédit bancaire est une décision cruciale pour les institutions financières, car il représente à la fois une opportunité de générer des revenus à travers les intérêts et un risque potentiel en cas de non-remboursement. Une mauvaise évaluation du profil d’un client peut entraîner des pertes importantes pour la banque. Traditionnellement, cette décision repose sur des critères manuels et l’expertise humaine, ce qui peut être lent, coûteux et parfois biaisé.  
Le **machine learning** offre une solution moderne en automatisant et en optimisant cette prise de décision à l’aide d’algorithmes capables d’analyser un grand volume de données et de détecter des modèles complexes. Cela permet de prédire avec plus de précision la probabilité qu’un client rembourse son crédit, améliorant ainsi la rentabilité et la gestion du risque pour les banques.

**1.Collecte des Données**

Les données utilisées dans ce projet proviennent d’un fichier CSV nommé credit\_data.csv, contenant **613 lignes** et **12 colonnes**. Ce dataset regroupe des informations socio-économiques et financières sur des clients ayant fait une demande de prêt bancaire.  
Chaque ligne représente un client, identifié par un identifiant unique (Loan\_ID). Les autres variables décrivent le profil du client :

* **Gender** : sexe du client (Male/Female)
* **Married** : statut marital (Yes/No)
* **Dependents** : nombre de personnes à charge (0, 1, 2, 3+)
* **Education** : niveau d’éducation (Graduate ou Not Graduate)
* **Self\_Employed** : statut de travailleur indépendant (Yes/N
* **ApplicantIncome** : revenu mensuel du demandeur principal
* **CoapplicantIncome** : revenu mensuel du co-demandeur
* **LoanAmount** : montant du prêt demandé (en milliers)
* **Loan\_Amount\_Term** : durée du prêt (en mois)
* **Credit\_History** : antécédents de crédit (1 = bon, 0 = mauvais, NaN = inconnu)
* **Property\_Area** : zone géographique de la propriété (Urban, Semiurban, Rural)

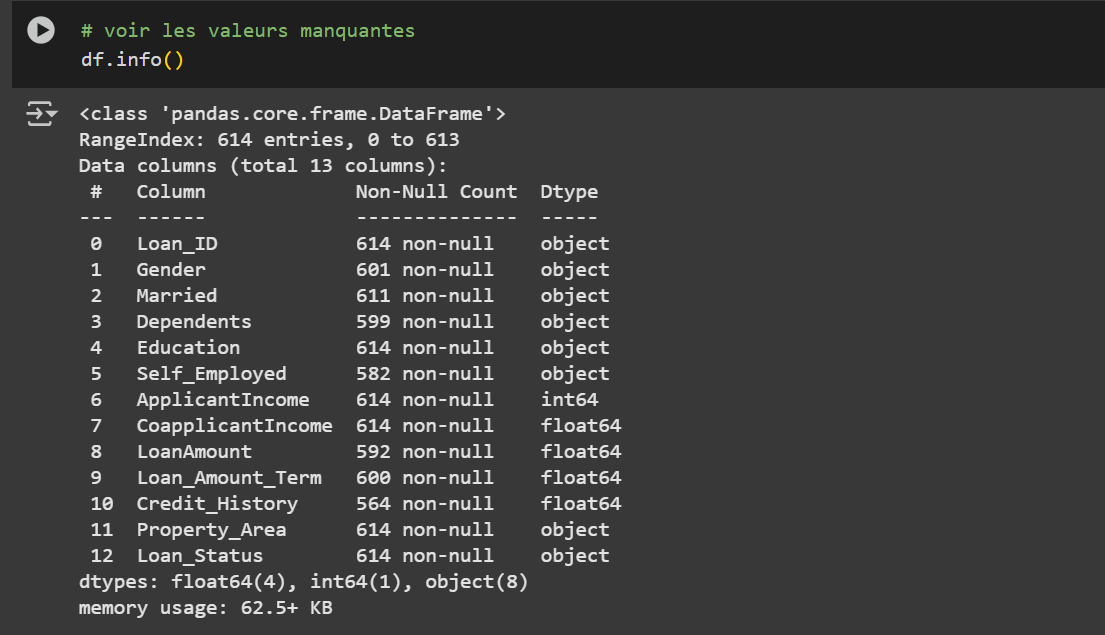
# Ce jeu de données est bien adapté pour un problème de **classification binaire**, dont l’objectif est de prédire si un prêt doit être **accordé ou non** à un client, en fonction de son profil.



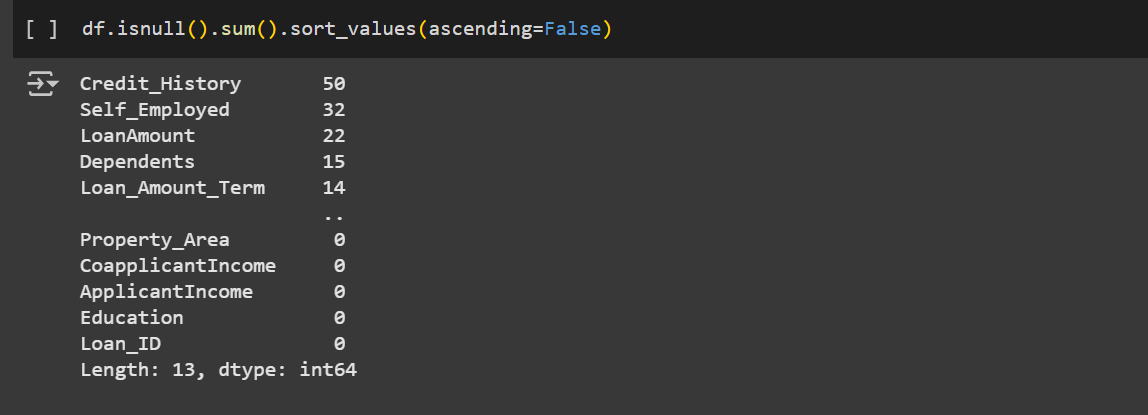
# 2-Nettoyage de la Base de Données

Pour assurer la qualité du dataset, une analyse des valeurs manquantes a été réalisée. Les variables ont été séparées en deux types : catégorielles et numériques. Les valeurs manquantes des variables catégorielles ont été remplacées par la modalité la plus fréquente afin de conserver la cohérence des données. Pour les variables numériques, les valeurs manquantes ont été comblées en utilisant la méthode du remplissage par la valeur suivante disponible, ce qui permet de limiter l’impact sur la distribution des données. Ces étapes ont permis de nettoyer efficacement la base tout en préservant l’intégrité des informations.

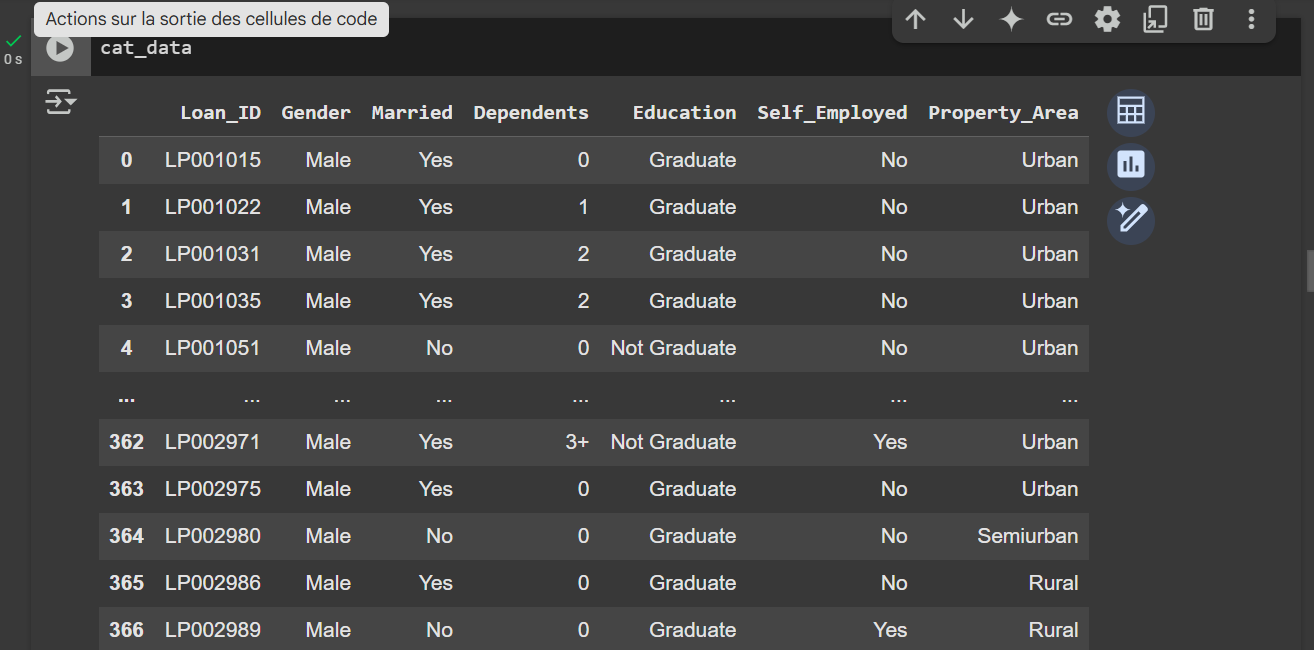
Cette capture présente un aperçu des colonnes du dataset, avec le type de données et le nombre d’enregistrements non nuls pour chaque variable.

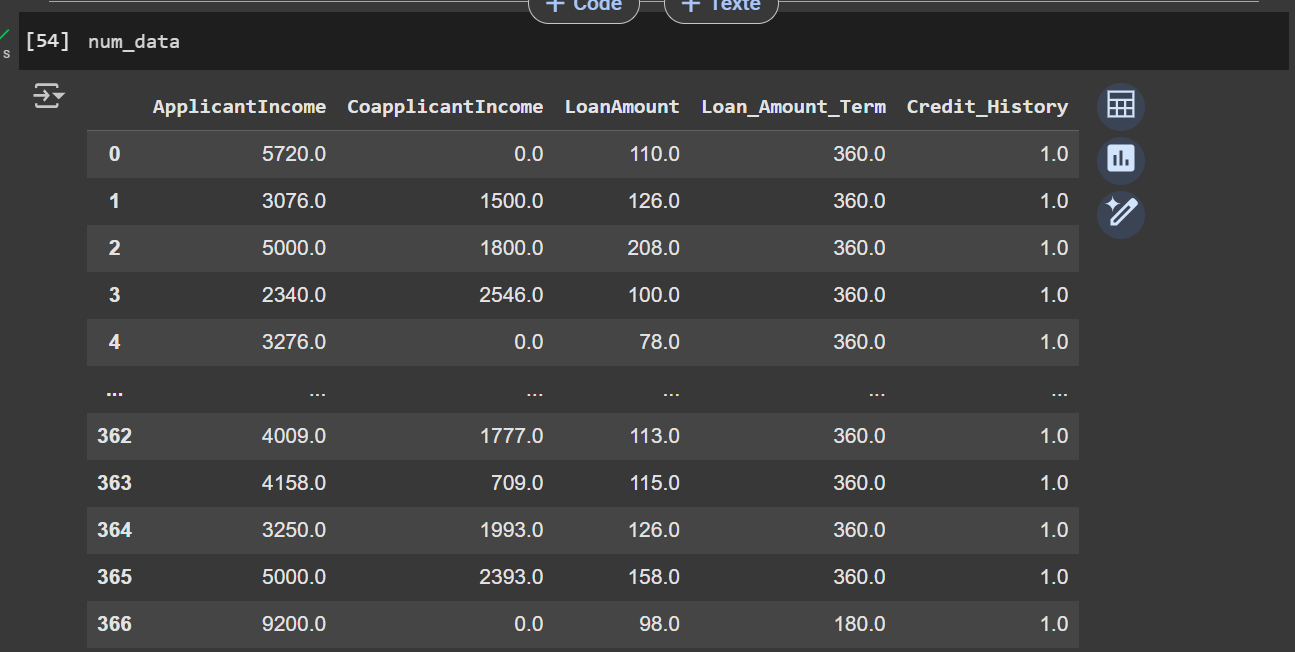


**Capture du résultat** df.isnull().sum()  
Ce tableau montre le nombre de valeurs manquantes présentes dans chaque colonne avant le nettoyage.



# Visualisation graphique de la répartition des valeurs manquantes dans les différentes colonnes du dataset.





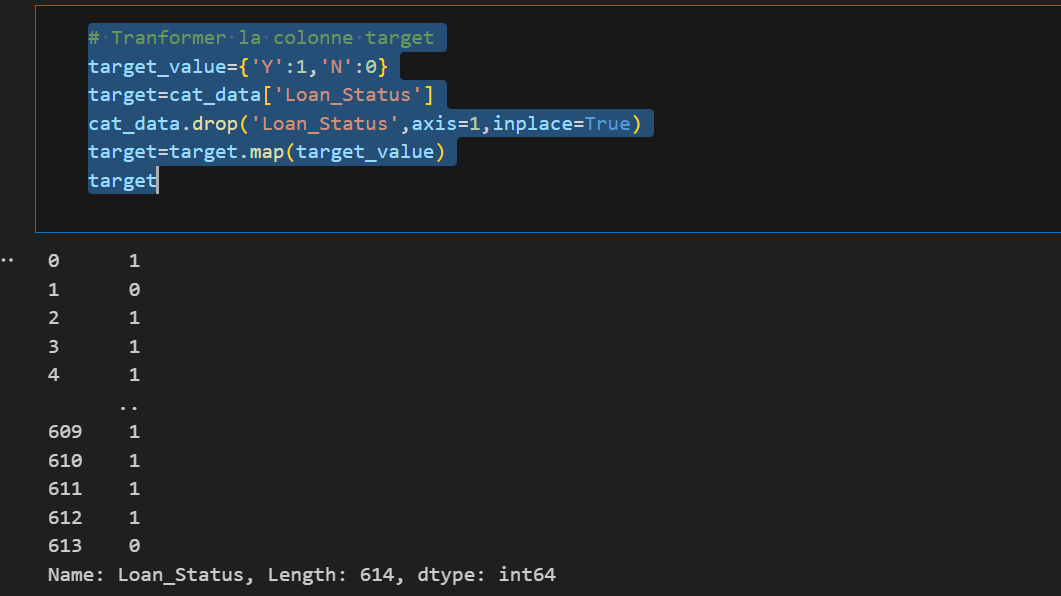
Transformation de la variable cible (Loan\_Status)

La variable cible Loan\_Status indique si un prêt a été accordé ('Y') ou non ('N'). Étant donné que la majorité des algorithmes de classification ne fonctionnent qu'avec des valeurs numériques, nous avons transformé cette variable catégorielle en une variable binaire :

'Y' devient 1 (prêt accordé)

'N' devient 0 (prêt refusé)

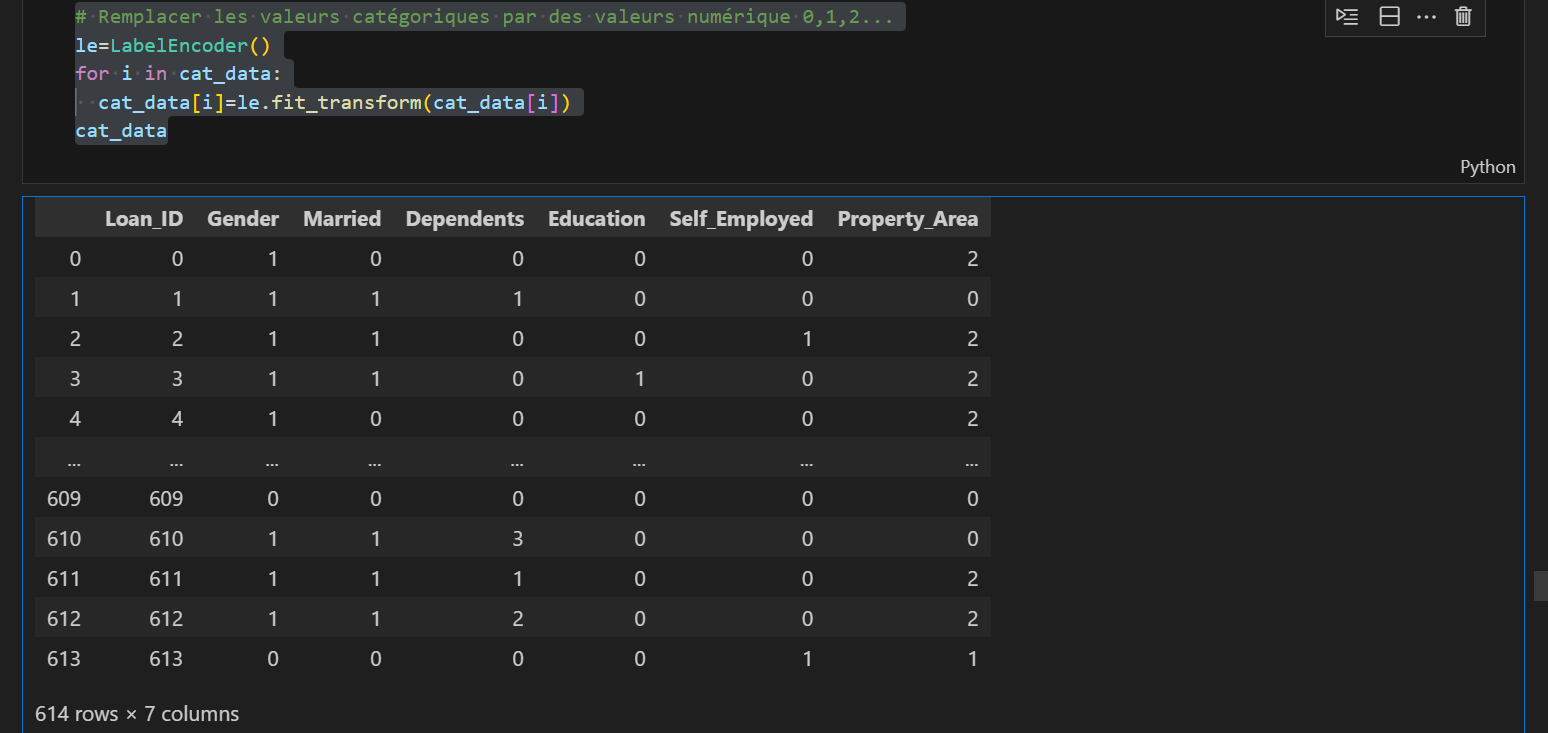
Cette transformation permet aux algorithmes de mieux apprendre à partir de cette variable cible lors de l'entraînement.



Encodage des variables catégorielles

Les variables catégorielles (comme Gender, Married, Education, etc.) ne peuvent pas être utilisées directement par les algorithmes de machine learning, car ces derniers travaillent uniquement avec des valeurs numériques.

Pour résoudre ce problème, nous avons utilisé la technique **Label Encoding**. Cette méthode consiste à transformer chaque catégorie en un nombre entier unique (ex. : Male → 1, Female → 0). Cela permet à nos modèles d’interpréter correctement ces variables.



Chaque colonne catégorielle a ainsi été convertie en valeurs numériques, ce qui permet de les intégrer avec les variables numériques pour entraîner les modèles.

**Suppression de la colonne Loan\_ID**

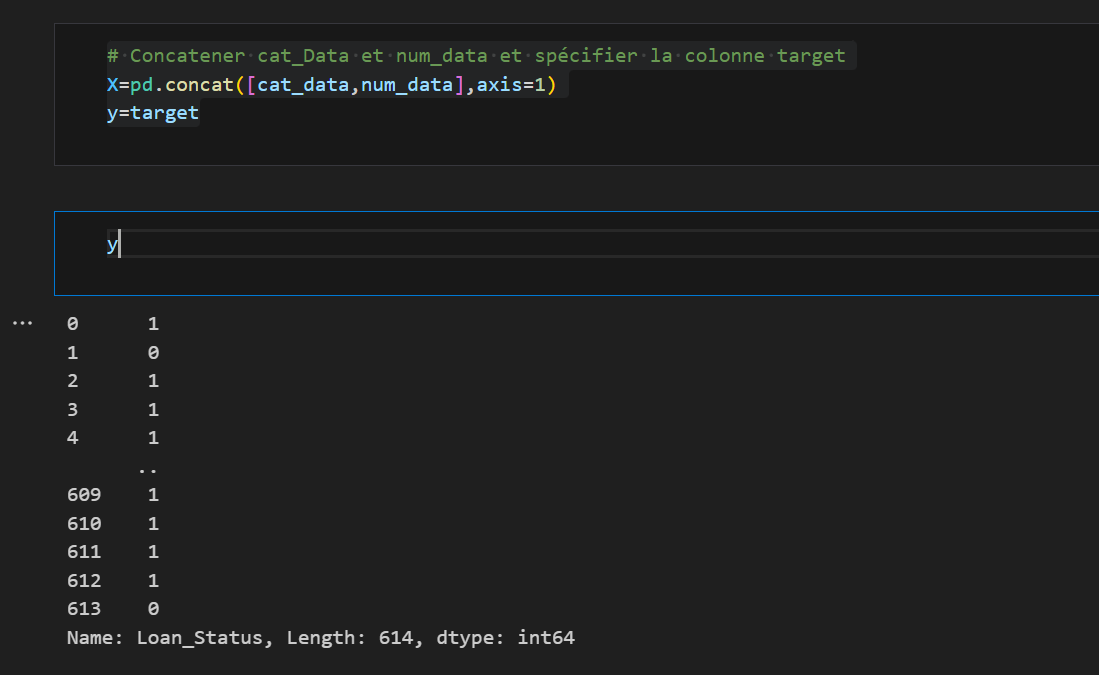
La colonne Loan\_ID contient un identifiant unique pour chaque demande de prêt. Bien qu'elle soit utile pour distinguer les enregistrements, **elle ne contient aucune information pertinente pour prédire l’octroi d’un crédit**.

Nous avons donc choisi de **supprimer cette colonne** afin d'éviter qu’elle n’introduise du bruit inutile dans le modèle.

**Fusion des données catégorielles et numériques**

Après avoir traité séparément les données **catégorielles** (cat\_data) et **numériques** (num\_data), nous avons procédé à leur **fusion** pour obtenir une seule base de données exploitable par les modèles de machine learning.

Nous avons également **spécifié la variable cible** y, qui représente la décision d’octroi du crédit (Loan\_Status, transformée en 0 ou 1).



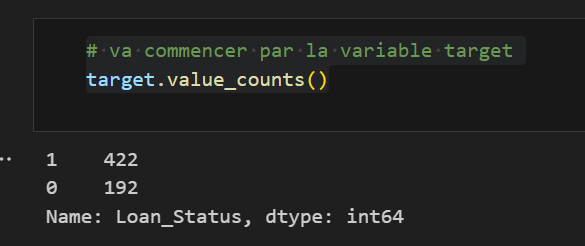
# Analyse Exploratoire

L’analyse exploratoire des données (EDA) a pour objectif de mieux comprendre la distribution des variables et d’identifier les relations potentielles entre les caractéristiques et la variable cible Loan\_Status.

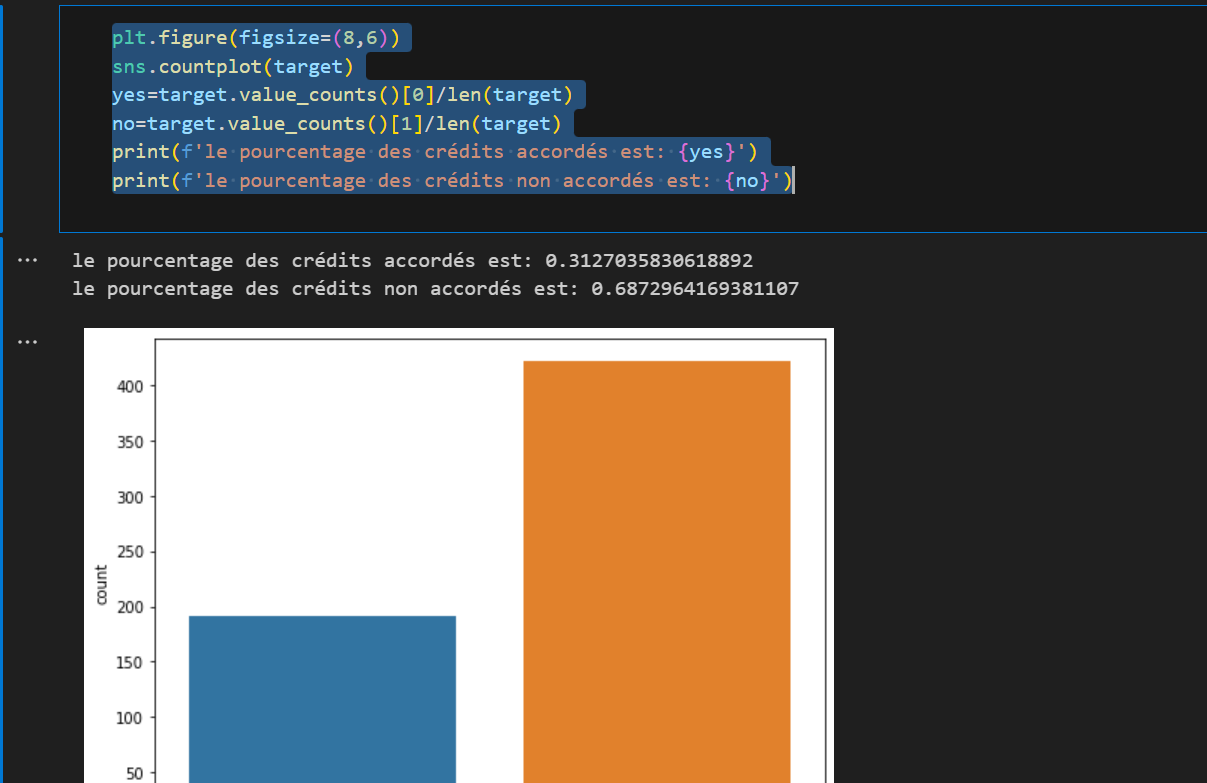
Nous avons d'abord examiné la répartition des classes de la variable cible Loan\_Status. Après transformation, la classe 1 correspond aux prêts accordés, et la classe 0 aux prêts refusés.

La commande target.value\_counts() a permis de constater un **déséquilibre** entre les deux classes, avec une majorité de crédits acceptés.

Cela peut influencer la performance des modèles et devra être pris en compte lors du choix des algorithmes ou de l'évaluation (notamment avec la précision ou les métriques comme le F1-score).



La variable cible Loan\_Status indique si le prêt a été accordé ou non. L’analyse montre que la majorité des prêts sont acceptés, ce qui crée un léger déséquilibre entre les classes. Cette information est importante car elle peut influencer les résultats des modèles de machine learning.



L’étude de la variable Credit\_History en fonction du statut du prêt (Loan\_Status) révèle que la majorité des prêts accordés concernent des demandeurs ayant un historique de crédit positif. Cette variable semble donc jouer un rôle important dans la décision d’octroi du crédit.



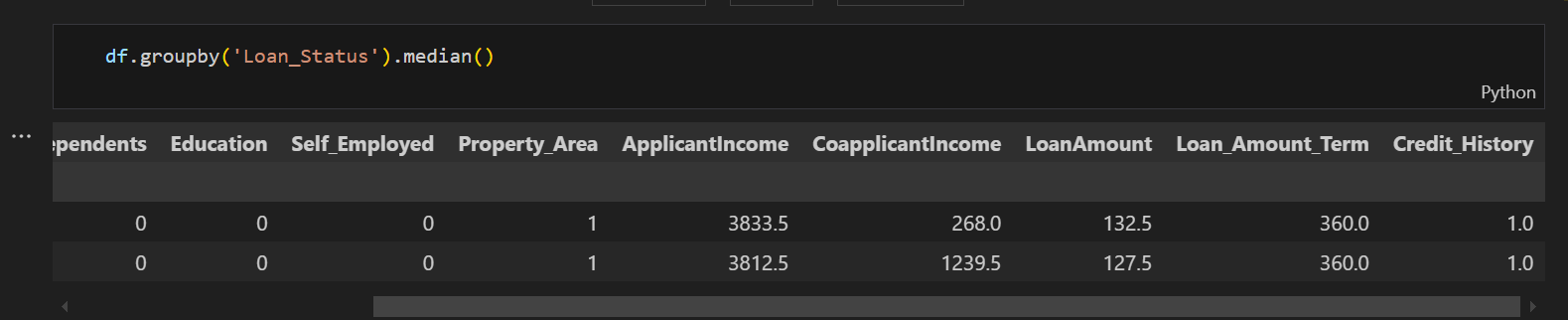
L’analyse de la répartition du genre (Gender) selon le statut du prêt (Loan\_Status) montre une légère prédominance des prêts accordés aux hommes par rapport aux femmes. Cependant, la différence n’est pas très marquée, ce qui suggère que le genre pourrait avoir une influence modérée, voire limitée, dans la décision d’octroi du prêt.



Le graphique de dispersion entre le revenu du co-demandeur (CoapplicantIncome) et le statut du prêt (Loan\_Status) permet d’observer s’il existe une relation entre le niveau de revenu secondaire et l’octroi du prêt. On remarque que les prêts sont accordés pour une large gamme de revenus, sans qu’un seuil clair ne se dégage. Cela suggère que le revenu du co-demandeur seul n’est pas un facteur décisif évident pour l’approbation du prêt.



Le tableau des médianes groupées par le statut du prêt (Loan\_Status) montre les valeurs centrales des variables numériques selon que le prêt a été accordé ou refusé. Cette analyse permet de comparer les caractéristiques typiques des demandeurs dans chaque groupe, par exemple le revenu du demandeur ou le montant du prêt. On peut ainsi identifier des tendances ou différences marquantes qui influencent la décision d’octroi du crédit.

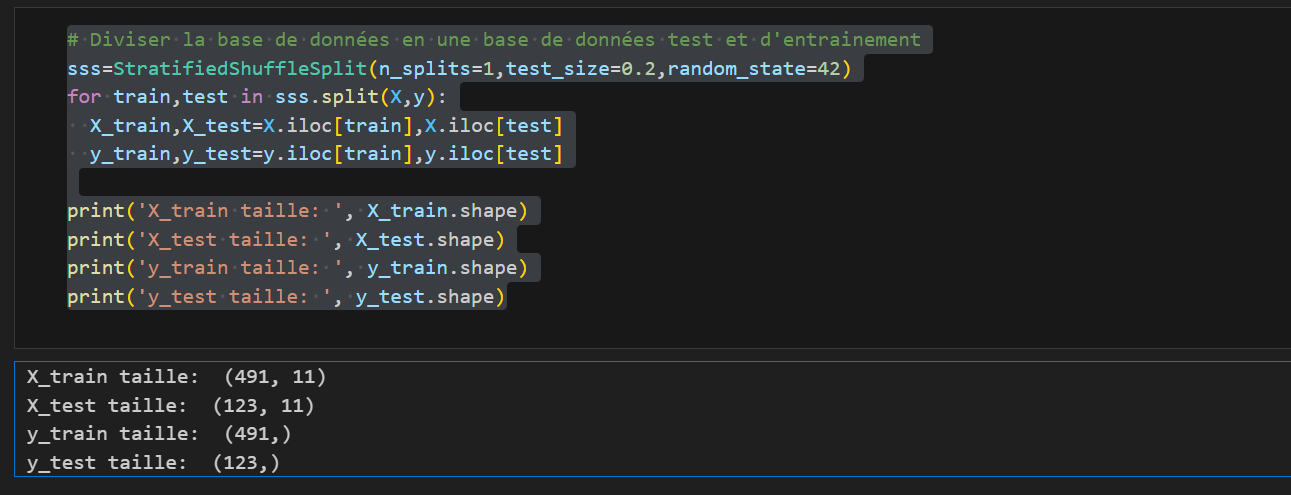


# Préparation du Modèle

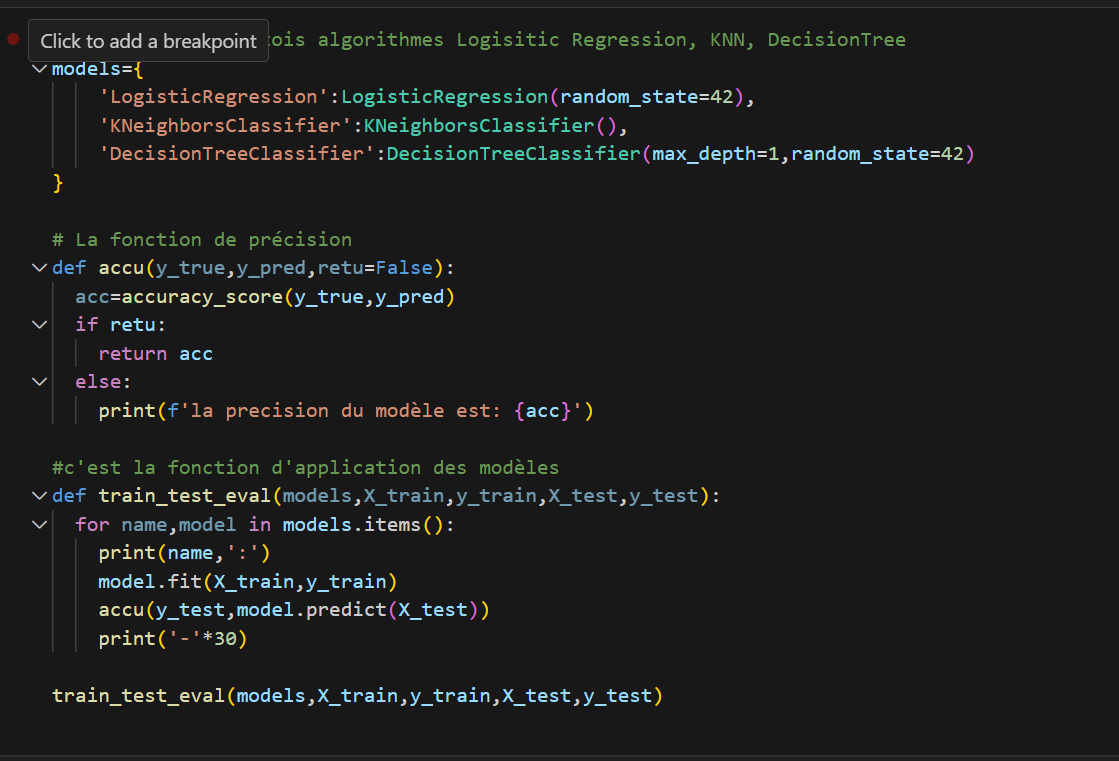
Les données ont été prétraitées en remplissant les valeurs manquantes des variables catégoriques par la modalité la plus fréquente, et celles des variables numériques par la méthode de remplissage vers l’arrière. La variable cible Loan\_Status a été convertie en valeurs binaires (1 pour oui, 0 pour non). Ensuite, les variables catégoriques ont été encodées en valeurs numériques avec un label encoder, et la colonne Loan\_ID sans importance a été supprimée. Enfin, les données ont été divisées en ensembles d’entraînement (80 %) et de test (20 %) tout en conservant la même répartition de la variable cible dans chaque ensemble.

****Division des données en ensembles d’entraînement et de test****

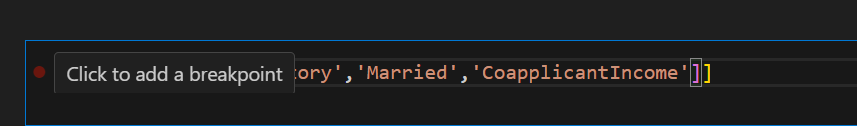
Pour évaluer la performance des modèles, la base de données a été divisée en deux ensembles : un ensemble d’entraînement contenant 80 % des données, et un ensemble de test contenant les 20 % restants. La méthode utilisée, Stratified Shuffle Split, permet de conserver la même proportion de chaque classe de la variable cible dans les deux ensembles, assurant ainsi une répartition équilibrée des catégories.



La base de données a été divisée en ensembles d’entraînement (80 %) et de test (20 %) en conservant la proportion des classes grâce à la méthode Stratified Shuffle Split. »

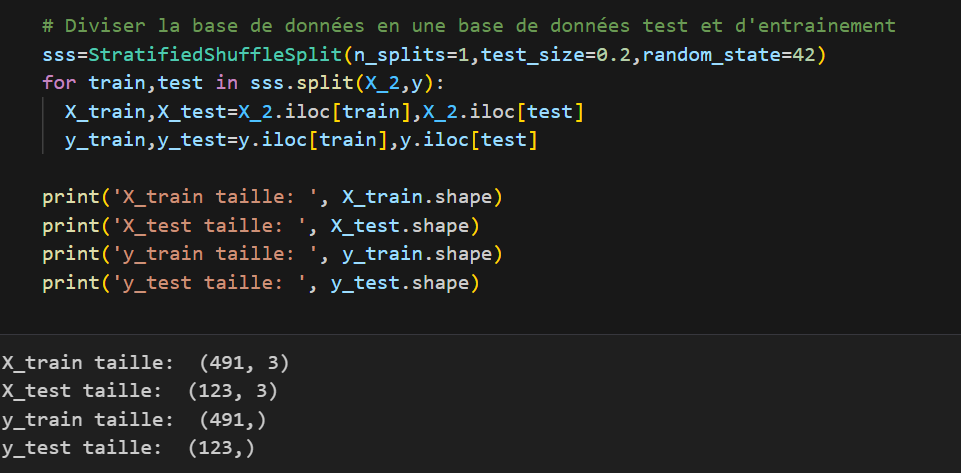


Pour simplifier le modèle, seules les variables Credit\_History, Married et CoapplicantIncome ont été sélectionnées pour une analyse plus ciblée.



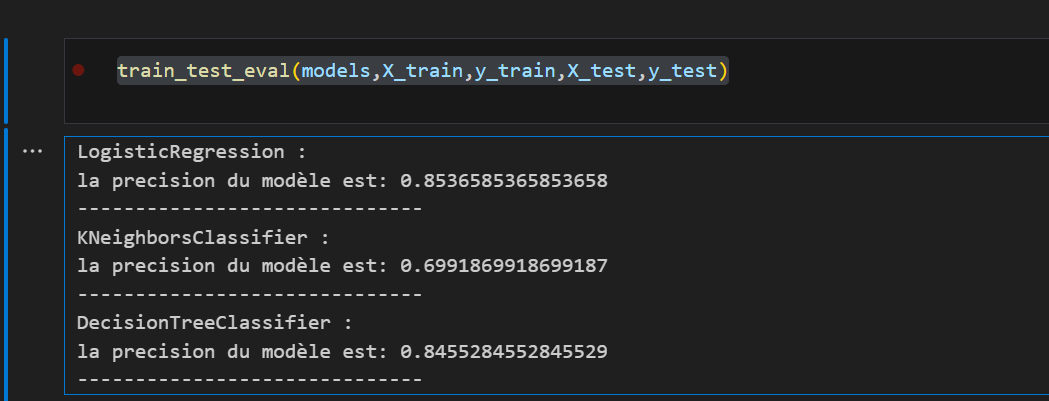
**Division des données en ensembles d’entraînement et de test (variables sélectionnées):**

Pour construire un modèle plus simple, nous avons sélectionné trois variables : Credit\_History, Married et CoapplicantIncome. Nous avons à nouveau divisé les données en ensembles d’entraînement (80 %) et de test (20 %) en utilisant la méthode Stratified Shuffle Split. Cette méthode garantit que la distribution des classes de la variable cible est conservée dans les deux ensembles, permettant ainsi une évaluation plus fiable des performances du modèle.



**Entraînement et évaluation des modèles:**

Nous avons entraîné plusieurs algorithmes de classification (Régression Logistique, K-Nearest Neighbors, et Arbre de décision) sur les données d’entraînement, puis évalué leurs performances sur les données de test en calculant la précision. Cette étape nous a permis de comparer l’efficacité de chaque modèle pour prédire l’octroi de crédit.

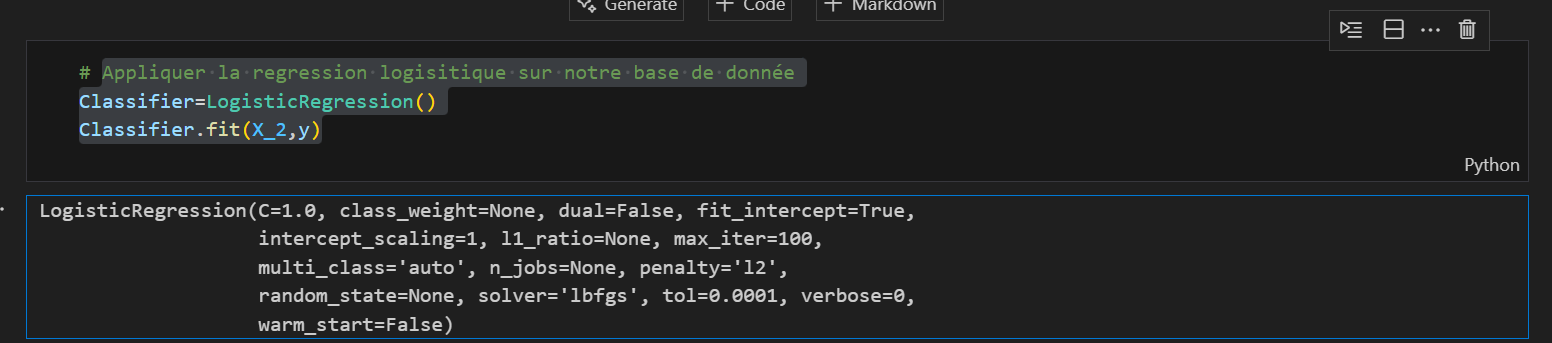


# 6. Déploiement du Modèle

Le meilleur modèle a été sauvegardé à l’aide de la bibliothèque joblib. Il peut être intégré dans une application web ou un service backend pour automatiser le processus de prédiction de crédit.

**Modélisation finale**

Après avoir évalué les différents modèles, nous avons choisi d’entraîner une régression logistique sur l’ensemble complet des données en utilisant les variables sélectionnées (Credit\_History, Married et CoapplicantIncome). Ce modèle sera ensuite utilisé pour prédire l’octroi de crédit sur de nouvelles données.



**Conclusion**

Ce projet a permis de construire un modèle de prédiction de l’octroi de crédit à partir d’un jeu de données clients. Après un prétraitement rigoureux des données, une analyse exploratoire approfondie a révélé les variables les plus influentes. Plusieurs modèles ont été entraînés et évalués, avec une performance satisfaisante obtenue notamment par la régression logistique. Ce travail démontre l’importance du nettoyage des données et du choix des variables dans la qualité des prédictions. Le modèle final pourra être utilisé pour aider à la prise de décision dans le domaine du crédit bancaire.