



République Tunisienne  
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique  
Université de Carthage - Ecole Supérieure de la Statistique et de l'Analyse de l'Information



*Rapport de Projet de Fin d'Etudes soumis afin d'obtenir le*

## **Diplôme National d'Ingénieur en Statistique et Analyse de l'Information**



*Réalisé par*

**YOSSER AKILI**

---

# Optimisation du Straight Through Processing Pour le Traitement des Demandes de Crédit

---

Soutenu le 21/06/2024 devant le Jury composé de :

Mme. Aicha El Golli, Maître-Assistant (Président)

M. Ghazi BelMufti, Maître-Assistant (Rapporteur)

Mme. Hajer Sellami, Maître-Assistant (Encadrant Universitaire)

M. Khaled Ghardallou, Data Manager (Encadrant Entreprise)

*Projet de Fin d'Etudes fait à*



**Blauwtrust**  
Tunisia

# Dédicace

# Remerciements

## Résumé

L'objectif de ce projet consiste à optimiser la sélection des crédits pour un traitement manuel dans le cadre de la supervision du système de traitement automatique des demandes de crédit de Quion, en mettant l'accent sur la rapidité et l'efficacité du processus automatique d'octroi de crédit en ligne.

Nous avons d'abord étudié et évalué différents modèles de machine learning pour estimer la probabilité de défaut, optant finalement pour le modèle de forêt aléatoire. Ensuite, nous avons exploré l'utilisation de l'apprentissage par renforcement, plus précisément l'algorithme du Bandit Multi-Bras en choisissant la stratégie *Upper Confidence Bound (UCB1)* pour la sélection des crédits à risque élevé. Cet algorithme a démontré sa capacité à identifier efficacement les crédits présentant un risque élevé, ce qui a conduit à une réduction du pourcentage de sélection pour un traitement manuel.

Ces résultats indiquent que notre approche permet non seulement d'améliorer l'efficacité du processus de sélection, mais également de réduire la nécessité de vérifications manuelles, offrant ainsi un gain de temps et de ressources pour Quion.

**Mots clés**— *Processus automatique d'octroi des crédits, Système de scoring, Optimisation, Sélection, Probabilité de défaut, Apprentissage Automatique, Apprentissage par renforcement*

## Abstract

The aim of this project is to optimize the selection of credits for manual processing as part of the supervision of Quion's automated credit application processing system, with an emphasis on the speed and efficiency of the online credit granting process.

We first studied and evaluated different machine learning models to estimate the probability of default, ultimately opting for the random forest model. Next, we explored the use of reinforcement learning, specifically the Multi-Armed Bandit algorithm by choosing the *Upper Confidence Bound (UCB1)* strategy for the selection of high-risk credits. This algorithm demonstrated its ability to effectively identify high-risk credits, leading to a reduction in the percentage of selections requiring manual processing.

These results indicate that our approach not only improves the efficiency of the selection process but also reduces the need for manual verifications, thereby providing a time and resource gain for Quion.

**Keywords**— *Automatic credit granting process, Scoring system, Optimization, Selection, Default probability, Machine Learning, Reinforcement Learning*

# Table des matières

Table des figures	iv
Liste des tableaux	vi
Introduction	1
<b>1 Contexte général du projet</b>	<b>3</b>
1.1 L'organisme d'accueil . . . . .	3
1.1.1 Présentation de Blauwtrust Groep . . . . .	3
1.1.2 Les entités de Blauwtrust Groep . . . . .	4
1.2 Etude de l'existant . . . . .	6
1.2.1 Description de l'existant . . . . .	6
1.2.2 Méthode actuelle de sélection des crédits . . . . .	7
1.2.3 Critique de l'existant . . . . .	10
1.3 Problématique et Objectif du projet . . . . .	11
1.4 Conclusion . . . . .	12
<b>2 Modélisation de la Probabilité de Défaut par les Outils du Machine Learning</b>	<b>13</b>
2.1 Définition du Machine Learning . . . . .	13
2.2 Revue de la littérature : Utilisation du Machine learning dans la modélisation de la Probabilité de défaut . . . . .	14
2.3 Modèles de Classification . . . . .	15
2.3.1 K-Nearest Neighbors (KNN) . . . . .	16
2.3.2 Régression logistique . . . . .	17
2.3.3 Random Forest . . . . .	19
2.3.4 Support Vector Machine (SVM) . . . . .	21
2.4 Métriques d'Évaluation des Modèles de Classification . . . . .	22
2.4.1 Matrice de confusion . . . . .	22
2.4.2 Mesures de performance . . . . .	23
2.4.3 Courbe ROC et AUC . . . . .	25
2.5 Conclusion . . . . .	26
<b>3 Algorithme d'Apprentissage par Renforcement pour la Sélection des Crédits</b>	<b>27</b>
3.1 Apprentissage par Renforcement . . . . .	27
3.2 Algorithme de Bandit Multi-Bras . . . . .	29
3.2.1 Définition . . . . .	30
3.2.2 Exploration et Exploitation . . . . .	30
3.2.3 Méthodes de minimisation du regret . . . . .	31

3.2.4	Évaluation du modèle . . . . .	32
3.3	Conclusion . . . . .	33
<b>4</b>	<b>Implémentation des Modèles de Machine Learning et Estimation de la Probabilité de Défaut</b>	<b>34</b>
4.1	Analyse de données . . . . .	34
4.1.1	Présentation de la base de données . . . . .	34
4.1.2	Analyse univariée . . . . .	35
4.1.3	Analyse bivariée . . . . .	38
4.2	Nettoyage des données . . . . .	40
4.2.1	Données Manquantes . . . . .	40
4.2.2	Valeurs aberrantes . . . . .	43
4.3	Tranformation des variables . . . . .	44
4.3.1	Conversion des catégories en valeurs numériques . . . . .	45
4.3.2	Division des données . . . . .	46
4.3.3	Normalisation des variables numériques . . . . .	47
4.4	Rééchantillonnage des données . . . . .	47
4.5	Implémentation des modèles de machine learning . . . . .	48
4.5.1	KNN . . . . .	49
4.5.2	Régression logistique . . . . .	52
4.5.3	Forêts aléatoires . . . . .	55
4.5.4	Machine à Vecteurs de Support . . . . .	58
4.6	Comparaison des Modèles . . . . .	61
4.7	Conclusion . . . . .	61
<b>5</b>	<b>Implémentation et Evaluation de l’Algorithme de Bandit Multi-bras</b>	<b>63</b>
5.1	Principe de fonctionnement de l’algorithme de Bandit Multi-Bras . . . . .	63
5.1.1	La stratégie Epsilon Greedy . . . . .	64
5.1.2	La stratégie Upper Confidence Bound UCB1 . . . . .	64
5.2	Evaluation des algorithmes . . . . .	65
5.2.1	Résultats de l’algorithme Epsilon-Greedy . . . . .	65
5.2.2	Résultats de l’algorithme UCB1 . . . . .	69
5.2.3	Comparaison des Algorithmes Epsilon-Greedy et Upper Confidence Bound UCB1	73
5.3	Comparaison des Méthodes de Sélection . . . . .	73
5.4	Conclusion . . . . .	75
	<b>Conclusion</b>	<b>76</b>
	<b>Bibliographie</b>	<b>78</b>
	<b>Annexes</b>	<b>80</b>
<b>A</b>	<b>Mesures Calculées dans la Méthode Actuelle d’Echantillonnage</b>	<b>80</b>
<b>B</b>	<b>Description des Variables</b>	<b>82</b>
<b>C</b>	<b>Visualisation des Variables</b>	<b>83</b>

<b>D</b>	<b>Modèles de Régression</b>	<b>87</b>
	1. Régression linéaire . . . . .	87
	4. Forêts aléatoires . . . . .	90
<b>E</b>	<b>Algorithmes de Bandit Multi-Bras</b>	<b>94</b>
	E.1 Stratégie Epsilon-Greedy . . . . .	94
	E.2 Stratégie Upper Confidence Bound UCB1 . . . . .	95

# Table des figures

1.1	Entités de Blauwtrust Groep . . . . .	4
1.2	Intervalles de montants . . . . .	6
1.3	Résultats de la méthode d'échantillonnage . . . . .	10
2.1	Fonctionnement du KNN . . . . .	16
2.2	Visualisation de la fonction sigmoïde . . . . .	17
2.3	Fonctionnement du modèle Random Forest . . . . .	19
2.4	Fonctionnement du modèle SVM . . . . .	21
2.5	Schéma Explicatif de la Courbe ROC . . . . .	25
2.6	Schéma Explicatif de l'AUC . . . . .	26
3.1	Fonctionnement de l'Apprentissage par Renforcement . . . . .	28
3.2	Les machines à sous . . . . .	30
4.1	Distribution des clients selon le défaut/non défaut . . . . .	35
4.2	Distribution des clients selon la durée d'emploi . . . . .	36
4.3	Distribution des demandes de crédits selon le grade . . . . .	37
4.4	Distribution des demandeurs de crédits selon le Revenu . . . . .	37
4.5	Distribution de l'âge des demandeurs de crédits . . . . .	38
4.6	Distribution de l'état du prêt selon le logement . . . . .	39
4.7	Distribution de l'âge en fonction du défaut . . . . .	39
4.8	Pourcentages de Valeurs Manquantes . . . . .	40
4.9	Tableau récapitulatif de la performance des modèles de régression pour la prédiction des valeurs manquantes . . . . .	42
4.10	Performance du modèle Random Forest dans la prédiction du taux d'intérêt du prêt . . . . .	43
4.11	Colonnes obtenues après transformation . . . . .	45
4.12	Matrice de confusion du modèle KNN . . . . .	49
4.13	Courbe ROC du modèle KNN . . . . .	51
4.14	Matrice de confusion du modèle RL . . . . .	52
4.15	Courbe ROC du modèle RL . . . . .	54
4.16	Matrice de confusion du modèle Forêt Aléatoire . . . . .	56
4.17	Courbe ROC du modèle Forêts Aléatoires . . . . .	57
4.18	Matrice de confusion du modèle SVM . . . . .	59
4.19	Courbe ROC du modèle SVC . . . . .	60
5.1	Pourcentage d'Actions Optimales vs Étapes . . . . .	65
5.2	Pourcentage de Sélection des Montants Risqués par Intervalle pour l'algorithme Epsilon-Greedy . . . . .	66



5.3	Pourcentage de Montant Risqué Sélectionné par Intervalle par rapport au Montant Total pour Epsilon-Greedy . . . . .	68
5.4	Pourcentage d'Actions Optimales vs Étapes pour UCB1 . . . . .	69
5.5	Pourcentage de Sélection des Montants risqués par Intervalle pour l'algorithme UCB1	70
5.6	Pourcentage de Sélection des Montants risqués par Intervalle par rapport au Montant Total pour l'UCB1 . . . . .	72
C.1	Performance du modèle Random Forest dans la prédiction du taux d'intérêt du prêt .	83
C.2	Distribution des demandeurs selon la longueur de l'historique des crédits . . . . .	84
C.3	Distribution des Statuts de Prêts Selon l'Historique . . . . .	84
C.4	Distribution des Statuts de Prêts par Grade . . . . .	85
C.5	Distribution des Taux d'Intérêt par Etat du Prêt . . . . .	85
C.6	Distribution des Montants de Prêts par Etat du Prêt . . . . .	86

# Liste des tableaux

- 1.1 Description des mesures de risque . . . . . 9
- 2.1 Matrice de confusion . . . . . 22
- 4.1 Rapport de Classification du modèle KNN . . . . . 50
- 4.2 Rapport de classification du modèle régression logistique . . . . . 53
- 4.3 Rapport de Classification du modèle Forêts aléatoires . . . . . 56
- 4.4 Rapport de classification du modèle SVC . . . . . 59
- 4.5 Comparaison des résultats des modèles . . . . . 61
- 5.1 Comparaison des Pourcentages de Sélection entre UCB1, la Méthode d’Échantillon-  
nage et la Réalité . . . . . 73
- A.1 Formules utilisées dans l’échantillonnage aléatoire . . . . . 80
- B.1 Description des variables . . . . . 82

# Introduction

Dans le secteur financier, le processus traditionnel d'obtention de crédits peut souvent être long, ce qui peut constituer un obstacle majeur pour les consommateurs qui ont besoin de fonds rapidement pour répondre à leurs besoins, tels que les dépenses liées à la rénovation ou à la construction de leurs maisons. Dans ce contexte, notre projet de stage de fin d'études vise à améliorer l'expérience des consommateurs dans le domaine des crédits, en mettant l'accent sur la rapidité et l'efficacité du processus d'octroi.

Blauwtrust Tunisia, en tant que filiale de Blauwtrust Groep opérant en Tunisie, se spécialise principalement dans les technologies de l'information en relation avec les services financiers. Blauwtrust Groep, quant à elle, joue un rôle majeur dans la facilitation du financement hypothécaire résidentiel aux Pays-Bas, grâce à ses entités telles que De HypothekersAssociatie, Dutch Mortgage Portfolio Management (DMPM), ROMEO Financiele Diensten, et notamment Quion.

Quion est l'un des plus importants prestataires de services tiers indépendants aux Pays-Bas pour les prêts hypothécaires et les crédits à la consommation. En collaborant avec des institutions financières nationales et internationales de renom, Quion aide à la gestion des prêts hypothécaires et des crédits à la consommation, offrant ainsi un large éventail de solutions. Son rôle en tant que partenaire d'affaires est crucial pour aider ces organisations à servir efficacement leurs clients dans un environnement financier en constante évolution.

Dans le cadre de notre projet, nous nous concentrons particulièrement sur Quion, cherchant à faciliter davantage le processus d'octroi de crédit pour les consommateurs. Nous reconnaissons que les consommateurs cherchent avant tout à obtenir des crédits de manière rapide et efficace, sans être encombrés par des formalités administratives complexes et des délais prolongés. Afin de concilier l'automatisation des processus, un système automatique (*Straight Thought Processing*) a été mis en place pour traiter les demandes de crédits des consommateurs. Dans un délai de 48 heures, le montant approuvé serait versé. Le *Straight Through Processing* est largement adopté dans le secteur bancaire pour faciliter divers processus tels que le traitement des prêts hypothécaires et des paiements. Dans notre cas, il est spécifiquement utilisé pour gérer le processus automatique d'octroi des crédits à la consommation. Cependant, il est essentiel de souligner qu'une décision aussi cruciale ne peut pas être entièrement déléguée à une machine sans supervision humaine.

Dans cette perspective, nous nous attachons à trouver un équilibre entre l'efficacité de la technologie automatisée et l'importance d'une surveillance humaine pour superviser le système de scoring. La problématique centrale de ce projet réside donc dans l'optimisation du processus de traitement des demandes de crédit en ligne. Actuellement, un pourcentage significatif de ces demandes est traité manuellement, ce qui peut entraîner une inefficacité opérationnelle. Le défi consiste à identifier et ci-

bler efficacement les crédits à haut risque de défaut afin d'optimiser le pourcentage des crédits traités manuellement. Nous proposons donc comme solution d'estimer dans un premier temps la probabilité de défaut des crédits et dans un deuxième temps de créer un système de sélection des crédits à haut risque de défaut qui devraient être traités manuellement, en se basant sur une analyse du risque de défaut.

Pour répondre à cette problématique, notre travail est partagé en cinq parties distinctes :

Dans une première étape, nous allons introduire le contexte général du projet en décrivant l'organisme d'accueil et ses entités et l'étude de la méthode existante pour pouvoir identifier la problématique, l'objectif et l'apport de notre projet.

Le deuxième chapitre est consacré à la présentation des outils du Machine Learning pour l'estimation de la probabilité de défaut et les métriques d'évaluation des différents modèles.

Pour le troisième chapitre, nous allons présenter dans une première étape l'apprentissage par renforcement, ensuite, l'algorithme du Bandit Multi-Bras pour la sélection des crédits.

Le quatrième chapitre est consacré à l'implémentation et la comparaison des différents modèles de machine en commençant par l'exploration de données, ensuite, le prétraitement et la modélisation et finalement nous allons évaluer et valider les différents modèles.

La dernière partie est consacrée pour l'implémentation et l'évaluation de l'algorithme de l'apprentissage par renforcement pour la sélection des crédits d'une part et d'autre part pour l'intégration du modèle d'estimation de la probabilité de défaut dans cet algorithme.

# Chapitre 1

## Contexte général du projet

Ce chapitre présente le contexte général du projet, en mettant en avant l'organisme d'accueil, Blauwtrust Groep, ainsi que les enjeux liés à l'optimisation du processus de traitement des demandes de crédit. Après avoir introduit l'organisme et exposé les différentes entités qui le composent, nous examinons de près l'existant, notamment le processus de traitement automatisé des demandes de crédit et la méthode d'échantillonnage utilisée pour la sélection des crédits. Cette analyse critique nous permet de formuler la problématique centrale du projet, qui concerne l'amélioration de ce processus de sélection des crédits à traiter manuellement. Enfin, nous détaillons les objectifs du projet et les bénéfices attendus de sa mise en œuvre, offrant ainsi une vision claire de la direction que prendra notre travail.

### 1.1 L'organisme d'accueil

Dans cette section, nous allons présenter l'organisme d'accueil, Blauwtrust Groep, qui occupe une place essentielle dans le secteur du financement hypothécaire résidentiel aux Pays-Bas. Cette section examinera en détail les différentes entités de Blauwtrust Groep et leur contribution à l'écosystème du financement hypothécaire néerlandais, mettant en lumière l'importance de Quion en tant qu'acteur central dans l'optimisation du processus d'octroi des crédits hypothécaires.

#### 1.1.1 Présentation de Blauwtrust Groep

Blauwtrust Groep joue un rôle crucial dans la mise en relation de l'offre et de la demande de financement hypothécaire résidentiel aux Pays-Bas à travers ses filiales : De Hypothekers Associatie, Quion, Dutch Mortgage Portfolio Management (DMPM) et ROMEO Financiele Diensten.

Fort de plus de 30 ans d'expérience, Blauwtrust Groep s'est établi comme un acteur majeur sur le marché néerlandais des prêts hypothécaires résidentiels. Le groupe se distingue en proposant des produits et services de haute qualité aux consommateurs, aux intermédiaires et aux investisseurs. Chaque entité de Blauwtrust Groep s'engage à offrir les solutions les plus compétitives afin d'assurer une satisfaction maximale de la clientèle.

Par ailleurs, l'ensemble des filiales de Blauwtrust Groep offre une gamme complète de services, allant du consommateur à l'investisseur. Cette intégration au sein d'un groupe indépendant est unique sur le marché néerlandais.

### 1.1.2 Les entités de Blauwtrust Groep

Chaque entreprise individuelle de la Blauwtrust Groep offre à ses clients des propositions tels que les services de conseil et de gestion de portefeuille de premier ordre, visant une plus grande satisfaction client. La figure 1.1 ci-dessous montre les différentes entités du groupe :

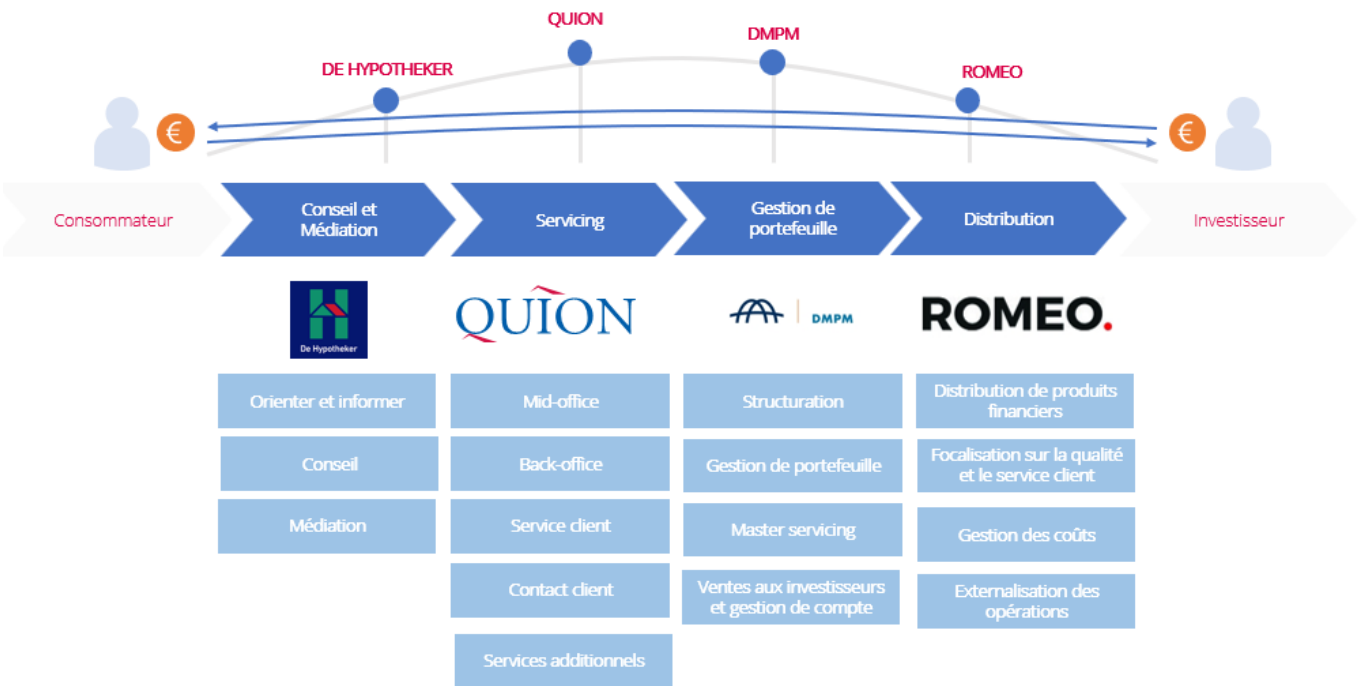


FIGURE 1.1 – Entités de Blauwtrust Groep

- **De Hypotheker :**

Au fil des 30 dernières années, De Hypotheker s’est affirmé comme la plus grande organisation de conseil en prêts hypothécaires aux Pays-Bas. Plus d’un million de personnes ont bénéficié de leurs services pour réaliser l’achat de leurs maisons. Avec plus de 160 agences à travers le pays, De Hypotheker offre un accompagnement professionnel pour aider la clientèle à prendre des décisions financières importantes.

- **Dutch Mortgage Portfolio Management (DMPM) :**

Dutch Mortgage Portfolio Management (DMPM) se distingue comme le gestionnaire d’investissement principal dans le secteur des prêts hypothécaires résidentiels néerlandais. Filiale intégrale de Blauwtrust Groep, DMPM offre une plateforme complète aux investisseurs, centralisant une gamme étendue de services proposés par les diverses entités du groupe.

DMPM fournit aux investisseurs une panoplie complète de marques hypothécaires et de solutions d’investissement, permettant ainsi une exposition ciblée au marché hypothécaire néerlandais et la

réalisation efficace des objectifs d'investissement.

DMPM facilite l'investissement dans des prêts hypothécaires nouvellement créés en offrant une gamme complète de services couvrant l'ensemble de la chaîne de valeur hypothécaire, de la distribution à l'origination (y compris les licences réglementaires), en passant par les approbations de crédit, la production de rapports de haute qualité et l'évaluation de portefeuille. En outre, DMPM assure la gestion de portefeuilles hypothécaires existants et conseille les clients sur tous les aspects de l'investissement dans le marché hypothécaire néerlandais, avec ses spécificités et subtilités.

- **ROMEO :**

ROMEO vise à faciliter le travail des conseillers financiers indépendants en leur offrant un soutien complet pour leurs activités principales. Grâce à une plateforme, ils peuvent optimiser leurs coûts tout en améliorant la qualité de leurs services, leur permettant ainsi de se concentrer davantage sur leur expertise : le conseil financier.

En intégrant les dernières technologies, en simplifiant les opérations et en mettant l'accent sur la satisfaction client, ROMEO offre des solutions compétitives qui répondent aux besoins changeants de ses clients.

- **Quion :**

Quion est considéré comme le principal prestataire de services hypothécaires en Europe, bénéficiant des meilleures notations de Fitch, et comme l'un des principaux fournisseurs de services hypothécaires aux Pays-Bas.

En tant qu'intermédiaire entre les consommateurs et les banques, Quion joue un rôle essentiel en aidant les institutions financières à répondre efficacement aux besoins de leurs clients dans un environnement en constante évolution. En facilitant la vie de ses partenaires commerciaux, Quion tire parti de son expertise, de son efficacité et de ses modèles opérationnels performants pour proposer des solutions innovantes et intelligentes pour la constitution, l'optimisation et la gestion de portefeuilles de prêts.

Quion se distingue également par sa proposition de nombreuses technologies innovantes. En outre, elle propose une plateforme en ligne dédiée au processus de traitement des crédits à la consommation. Cette plateforme permet non seulement d'accélérer et d'automatiser le traitement des demandes de crédit, mais aussi de fournir des conseils en ligne aux consommateurs. Cette approche renforce davantage le rôle de Quion en tant qu'intermédiaire efficace et partenaire commercial de confiance dans le domaine des services hypothécaires.

**En conclusion**, la présentation de Blauwtrust Groep et de ses différentes entités souligne l'importance de **Quion** en tant qu'**acteur central dans l'amélioration du processus d'octroi des crédits hypothécaires**. Au sein de cet écosystème, Quion se distingue par son engagement envers l'innovation et l'efficacité, notamment à travers sa plateforme en ligne.

## 1.2 Etude de l'existant

Dans cette section, nous présenterons, dans une première étape, une description du système automatique utilisé pour le traitement et l'octroi des crédits. Dans une deuxième partie, nous expliquerons la méthode d'échantillonnage par intervalle de montants adoptée par Quion depuis 2018 pour la sélection des crédits à traiter manuellement.

### 1.2.1 Description de l'existant

Au Pays Bas, les consommateurs ayant besoin de fonds pour payer des factures liées à la construction ou à la rénovation de leurs maisons ont la possibilité de demander un crédit en ligne. Actuellement, ces demandes sont traitées via le processus automatisé appelé Straight Through Processing (STP), où elles sont catégorisées en intervalles prédéfinis en fonction du montant demandé.

Le STP est un processus automatisé largement utilisé dans les transactions financières et commerciales. Il englobe un flux de travail allant de la saisie initiale des données de la transaction jusqu'à son règlement final, sans nécessiter d'intervention humaine. L'objectif principal du STP est d'optimiser l'efficacité des opérations en éliminant les interventions manuelles, réduisant ainsi le risque d'erreurs. Cette automatisation accélère le processus et le rend plus fiable, améliorant ainsi la qualité du service tout en réduisant les coûts opérationnels.

En éliminant les étapes manuelles, le STP permet également une réduction significative du temps de traitement des transactions. Cela se traduit par une meilleure réactivité aux besoins des clients et une augmentation de la capacité de traitement des volumes de transactions, sans augmentation proportionnelle des coûts opérationnels. Par conséquent, les institutions financières peuvent réaliser des économies substantielles tout en améliorant leur efficacité opérationnelle.

Cependant, chez Quion, seules les demandes de crédit d'un montant inférieur à 5,000 euros peuvent être traitées automatiquement par le *STP*, tandis que les demandes excédant ce montant sont vérifiées manuellement par un agent humain qui décidera de l'octroi ou non du crédit.

La figure 1.2 ci-dessous illustre la répartition des demandes (files) :

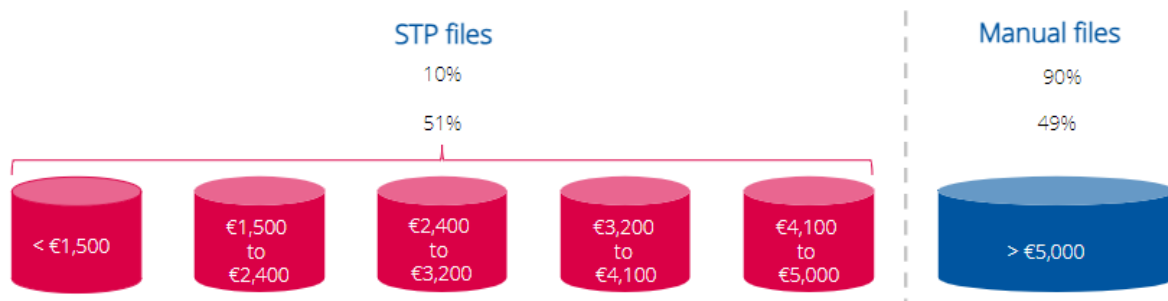


FIGURE 1.2 – Intervalles de montants

Nous avons constaté que, selon une étude réalisée par Quion, sur les données de l'année 2018 concernant les demandes de crédits, les demandes traitées manuellement, qui concernent des montants supérieurs à 5,000 euros, représentent 90% du montant total des crédits traités ce qui représente 49%



du nombre total des demandes de crédits.

Les demandes suivant le processus automatique ayant un montant inférieur à 5,000 euros soit 10% du montant des crédits traités par quion, représentent 51% du nombre total des demandes de crédits. Ces demandes sont réparties en cinq intervalles distincts en fonction du montant demandé. Cette répartition permet de traiter chaque tranche de montant séparément, minimisant ainsi le risque d'erreur du système de scoring qui évalue les demandes pour décider de l'octroi ou non du crédit. En effet, en répartissant les demandes en cinq intervalles distincts en fonction du montant demandé, le processus prend en compte la diversité des besoins financiers des clients. Chaque tranche de montant peut avoir ses propres caractéristiques et risques associés. Par exemple, les demandes de crédit de faible montant peuvent être moins risquées et nécessiter moins de vérifications approfondies, tandis que les demandes de montant élevé peuvent nécessiter une analyse plus approfondie en raison du risque financier plus élevé.

Parmi les demandes de crédits traitées de façon automatique par le STP, un certain pourcentage pour un traitement manuel afin de vérifier la fiabilité du système de scoring intégré dans le STP pour l'octroi des crédits.

### **1.2.2 Méthode actuelle de sélection des crédits**

Dans notre démarche visant à améliorer et optimiser la sélection des crédits à traiter manuellement, il est essentiel de comprendre en détail la méthode actuellement utilisée. Pour ce faire, nous devons d'abord explorer les principes théoriques liés aux méthodes d'échantillonnage. Ces concepts théoriques jetteront les bases nécessaires à la compréhension de la méthode actuelle de sélection des crédits.

## **1. Notions théoriques de l'échantillonnage**

L'échantillonnage est une technique utilisée dans les travaux d'enquête pour sélectionner une partie représentative d'une population plus large. Cela permet d'étudier cet échantillon spécifique afin de tirer des conclusions qui peuvent être généralisées à l'ensemble de la population.

### **La taille de l'échantillon :**

- La taille de l'échantillon est un terme utilisé pour définir le nombre des éléments inclus dans une étude.
- La détermination de la taille de l'échantillon est le processus qui vise à sélectionner le nombre approprié d'observations ou d'éléments au sein d'une population plus vaste pour constituer un échantillon. L'objectif est de garantir que l'échantillon soit assez grand pour fournir des résultats statistiquement fiables et des estimations précises des caractéristiques de la population.
- En sélectionnant un petit segment de la population, une certaine incertitude s'introduit dans les données statistiques. Il est impossible d'être absolument certain que les statistiques obtenues reflètent fidèlement l'ensemble de la population. Cette incertitude est connue sous le nom d'erreur d'échantillonnage et est souvent évaluée à l'aide d'un intervalle de confiance. Par exemple, s'il est indiqué que le niveau de confiance est de 99%, cela signifie que dans 99% des

cas, les résultats seraient cohérents si le travail était répété.

### Détermination de la taille de l'échantillon : Formule de Cochran

La formule de Cochran permet de déterminer la taille idéale de l'échantillon en fonction du niveau de confiance désiré, du niveau de précision souhaité et du pourcentage attendue de la population totale ayant une caractéristique spécifique. Cette formule est particulièrement utilisée dans les situations où la population est importante. En effet, un échantillon fournit plus d'informations sur une petite population que sur une grande. Ainsi la formule de Cochran intègre une correction qui permet de réduire la taille de l'échantillon calculée si la population totale est petite.

L'équation 1.1 présente la formule pour calculer la taille de l'échantillon  $n_0$  avec la méthode de Cochran :

$$n_0 = \frac{Z^2 \times p \times (1 - p)}{e^2} \quad (1.1)$$

Où :

- $e$  : la marge d'erreur (c'est le niveau de précision),
- $p$  : la proportion de la population qui possède la caractéristique correspondante,
- $q$  :  $1-p$ ,
- $Z$  : le z-score<sup>1</sup> correspondant au niveau de confiance désiré.

Dans le cas d'une population plus petite, la formule de Cochran sera modifiée comme suit :

$$n = \frac{n_0}{1 + \frac{(n_0-1)}{N}} \quad (1.2)$$

Où :

- $N$  : la taille de la population,
- $n$  : la nouvelle taille de l'échantillon ajustée,

## 2. Explication de la méthode d'échantillonnage actuelle

Une méthode d'échantillonnage a été élaborée pour optimiser la sélection des crédits à traiter manuellement dans chaque intervalle de montant inférieur à 5,000 euros, tout en cherchant à minimiser le pourcentage de ces sélections. La méthode actuellement utilisée vise à déterminer le SampleTotalPercentage, qui représente le pourcentage des montants à traiter manuellement. Cette méthode s'appuie sur un processus d'échantillonnage aléatoire afin de sélectionner une partie des crédits à traiter manuellement de manière efficace.

Tout d'abord, plusieurs mesures sont calculées à partir des données disponibles. Ces mesures comprennent des informations telles que le nombre d'éléments par intervalle, la moyenne des montants

---

1. une mesure statistique qui situe une valeur particulière par rapport à la moyenne d'un ensemble de données. Plus exactement, il indique le nombre d'écarts-types par lequel une valeur spécifique se trouve au-dessus ou en dessous de la moyenne.

par intervalle, les montants totaux par intervalle, les montants impayés par intervalle, etc. Ces mesures fournissent une vue d'ensemble de la répartition des crédits et des risques associés pour chaque intervalle.

Ensuite, des formules sont utilisées pour estimer différents paramètres, tels que la taille de l'échantillon nécessaire pour chaque intervalle, les limites inférieures et supérieures de l'intervalle de confiance, le risque basé sur la marge d'erreur, le risque basé sur le ratio d'impayés, etc. Ces calculs permettent d'évaluer le risque associé à chaque intervalle et de déterminer la proportion de crédits à traiter manuellement.

Finalement, ce processus garantit une sélection des crédits à traiter manuellement, en tenant compte à la fois des crédits impayés associés à chaque intervalle et de la répartition des montants.

Ainsi, cette méthode utilise des mesures de risque et des calculs statistiques pour guider la sélection des crédits à traiter manuellement, avec pour objectif de maximiser l'efficacité du processus de vérification. La Table 1.1 ci-dessous montre ces différentes mesures et les formules associées à chacune. D'autres formules sont disponibles dans l'annexe A.

## Mesures

Nom de la mesure	Description	Formule
SampleTotalPercentage	Pourcentage des montants affectés par les fichiers de traitement direct	$\frac{\sum \text{STPAmounts}}{\sum \text{TotalAmounts}}$
REST	Pourcentage de fichiers dépassant la limite supérieure des intervalles	$\frac{\sum (\text{rows} - \sum \text{number})}{\sum \text{rows}}$
TotalRisk	Pourcentage de risque sur le prêteur	$\frac{\sum \text{RiskBasedOnAVG}}{\sum \text{TotalAmount}}$
ManualFilesSample	Le taux des fichiers qui seront traités manuellement à l'intérieur des intervalles	$(1 - \text{REST}) \times \frac{\sum \text{STPAmount}}{\sum \text{TotalAmount}}$
STPTT	Taux total des demandes traitées manuellement	$\text{REST} + \text{ManualFilesSample}$

TABLE 1.1 – Description des mesures de risque

La mise en œuvre de cette méthode d'échantillonnage a nécessité une analyse minutieuse des distributions d'intervalle, de la fréquence des montants non payés et du besoin global en échantillonnage, utilisant des formules telles que le Cochran pour déterminer la taille nécessaire des échantillons. Chaque paramètre a été soigneusement intégré pour assurer que les décisions prises sur la base de ces échantillons sont à la fois représentatives et pertinentes, minimisant ainsi les risques tout en optimisant l'allocation des ressources.

Un pourcentage de sélection fixe est finalement associé à chaque intervalle de montants.

### 3. Résultats de la méthode de sélection

De chaque intervalle, un pourcentage de demandes de crédits est sélectionné pour la vérification manuelle. L'implémentation de cette méthode d'échantillonnage a généré les résultats de la figure 1.3 suivants :

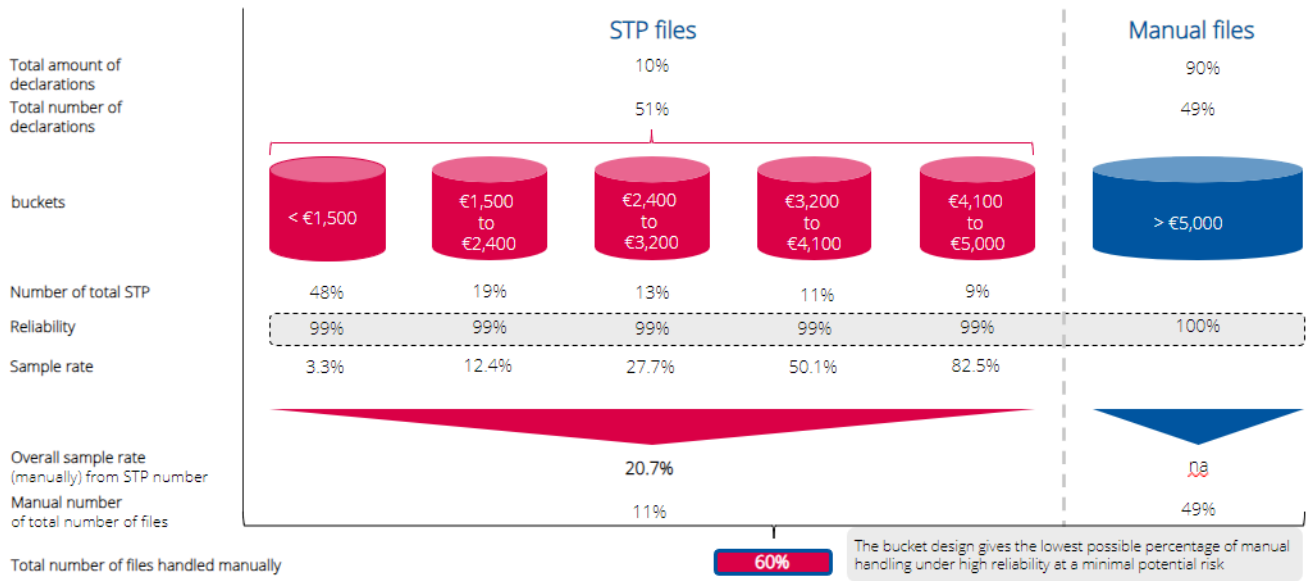


FIGURE 1.3 – Résultats de la méthode d'échantillonnage

La proportion d'échantillonnage prélevée du premier intervalle des montants de moins de 1500 euros, représente 3.3% du montant total de cette catégorie. Cette faible proportion peut s'expliquer par la nature des montants dans cet intervalle qui sont parmi les moins risqués, compte tenu de leur basse valeur. En revanche, pour la plage de montants allant de 1500 euros à 2400 euros, la proportion d'échantillonnage est plus élevée, représentant 12.4% du montant total de cette catégorie. Cette tendance se poursuit avec des pourcentages croissants pour les intervalles suivants : 27.7% pour les montants de 2400 à 3200 euros, 50.1% pour ceux de 3200 à 4100 euros, et 82.5% pour les montants de 4100 à 5000 euros. Ces échantillons cumulés représentent 20.7% du total des montants traités par Quion (<5,000 et >5,000), ce qui équivaut à 11% du nombre total des demandes de crédit en termes de volume.

Les demandes de montants dépassant les 5,000 euros sont toutes traitées manuellement, et elles représentent 49% du nombre total de demandes traitées par Quion.

#### 1.2.3 Critique de l'existant

La méthode d'échantillonnage des demandes de crédit devant être traitées manuellement, actuellement utilisée, présente des limites significatives en raison de son manque de sensibilité aux fluctuations des données. Lorsque des mises à jour ou des changements surviennent dans les données, les pourcentages générés par la méthode d'échantillonnage demeurent inchangés pour chaque intervalle. Ainsi, quels que soient les changements dans la distribution des crédits risqués et non risqués, cette méthode

maintient une approche rigide en sélectionnant des pourcentages préétablis pour chaque intervalle.

Dans notre contexte, où l'objectif est de cibler spécifiquement les crédits à haut risque de défaut pour un traitement manuel, cette approche présente des lacunes majeures. Elle ne s'adapte pas aux évolutions dynamiques du paysage financier. En outre, l'utilisation de pourcentages fixes pour l'échantillonnage aléatoire peut entraîner une sélection inégale et inefficace des crédits à haut risque, ce qui compromet l'efficacité globale du processus.

## 1.3 Problématique et Objectif du projet

La problématique centrale de ce projet concerne l'optimisation du processus de traitement des demandes de crédit en ligne chez Blauwtrust Groep. Actuellement, parmi les crédits d'un montant inférieur à 5 000 euros, un certain nombre, sélectionné de manière aléatoire selon la méthode d'échantillonnage décrite, est traité manuellement. Ces demandes de crédits traitées manuellement représentent 20.7% du montant total des crédits traités. Cependant, le pourcentage des crédits à haut risque de défaut, 20.53% du montant total traité, est inférieur à cette valeur fixe.

Cette méthode actuelle pourrait entraîner une inefficacité opérationnelle, car elle ne cible pas spécifiquement les crédits présentant un risque élevé de défaut. Par conséquent, il y a un potentiel d'amélioration significatif pour réduire le pourcentage de sélection tout en ciblant de manière plus précise les crédits à risque élevé, ce qui pourrait entraîner une meilleure prise de décision et une réduction des pertes pour l'organisme financier associée à une rapidité de traitement des demandes de crédit à faible risque de défaut.

La question fondamentale qui se pose est donc la suivante : Comment pouvons-nous améliorer ce processus de traitement automatique en introduisant une méthode de sélection des crédits à risque de défaut élevé pour un traitement manuel, tout en maintenant la rapidité du processus global ?

Dans ce travail, notre objectif principal est de développer un modèle prédictif de la probabilité de défaut pour cibler les crédits à haut risque de défaut. Nous utiliserons des techniques avancées d'apprentissage automatique pour élaborer ce modèle. Ensuite, nous intégrerons ce modèle dans un algorithme d'apprentissage par renforcement. Ce dernier sera conçu pour optimiser la sélection des crédits présentant un risque élevé de défaut au sein de chaque intervalle. L'objectif est d'améliorer significativement l'efficacité du processus de sélection, garantissant ainsi la flexibilité et l'adaptabilité des taux associés à chaque intervalle, tout en réduisant au minimum la nécessité de vérifications manuelles en ciblant les crédits présentant le plus grand risque de défaut.

L'intégration de cette approche innovante présente plusieurs avantages. Tout d'abord, elle permettra de réduire considérablement le temps nécessaire pour calculer la probabilité de défaut, rendant ainsi le processus plus rapide et plus réactif aux nouvelles données. De plus, cette méthode optimisée minimisera le risque de non-sélection des crédits à haut risque, un problème souvent rencontré avec la méthode d'échantillonnage traditionnelle utilisée qui donne des pourcentages fixes à chaque fois.

## 1.4 Conclusion

En conclusion, ce chapitre présente le contexte général du projet, centré sur l'optimisation du processus de traitement des demandes de crédit pour Blauwtrust Groep. Nous avons détaillé l'organisme, ses entités, et le processus actuel de traitement des crédits, en mettant en évidence les limitations de la méthode d'échantillonnage utilisée. La problématique centrale est d'améliorer la sélection des crédits à traiter manuellement en ciblant ceux à haut risque de défaut. Les objectifs visent à optimiser ce processus pour réduire les interventions manuelles tout en ciblant les demandes de crédits à haut risque de défaut. Pour ce faire, le chapitre suivant abordera l'aspect théorique de l'estimation de la probabilité de défaut.

# Chapitre 2

## Modélisation de la Probabilité de Défaut par les Outils du Machine Learning

Le chapitre se focalise sur l'exploration du versant théorique et mathématique des outils de machine learning dans le but d'estimer la probabilité de défaut. Cette méthodologie revêt un intérêt primordial dans le domaine financier, où la gestion efficace des risques est essentielle. En comprenant en profondeur les fondements théoriques des algorithmes de *machine learning*, nous allons pouvoir cibler avec précision les crédits à risque, ce qui permet de les sélectionner afin de prendre des décisions éclairées et de réduire les pertes potentielles. En examinant les concepts mathématiques sous-jacents à des techniques telles que la régression logistique, les k plus proches voisins (KNN), les forêts aléatoires et les machines à vecteurs de support (SVM), ce chapitre fournit une base solide pour la modélisation des risques de défaut.

### 2.1 Définition du Machine Learning

La modélisation est un processus qui vise à créer un modèle d'un système réel. Ce modèle est une représentation simplifiée et abstraite du système en question. Il est conçu pour capturer les principales caractéristiques et interactions du système, tout en éliminant les détails superflus. Grâce à ce modèle, nous pouvons mieux comprendre le fonctionnement interne du système, analyser ses composants et prévoir son comportement dans différentes situations ou conditions. En utilisant des techniques de modélisation appropriées, nous pouvons également simuler le système pour étudier son évolution au fil du temps ou en réponse à des changements spécifiques. Ainsi, la modélisation joue un rôle crucial dans la compréhension, l'analyse et la prédiction des systèmes complexes dans divers domaines[Blanchard et al., 2019].

L'apprentissage automatique, également connu sous le nom de machine learning, est une branche importante de l'intelligence artificielle qui se concentre sur la conception, l'analyse et la mise en œuvre de méthodes permettant aux machines d'apprendre et de s'adapter à partir de données. Ce processus d'apprentissage permet aux machines de réaliser des tâches complexes qui seraient difficiles, voire impossibles, à accomplir avec des algorithmes traditionnels[Norman and Gaston, 2020].

L'apprentissage automatique peut traiter une variété de données, y compris des valeurs symboliques comme des probabilités ou des intervalles, ainsi que des ensembles de modalités pour des attributs numériques ou catégoriels. Il est également capable d'analyser des données structurées sous forme de

graphes, d'arbres ou de courbes, que l'on qualifie de données continues, en opposition aux données discrètes associées à des attributs-valeurs plus conventionnels. Les modèles d'apprentissage automatique se répartissent principalement en trois catégories : l'apprentissage supervisé, l'apprentissage non-supervisé et l'apprentissage par renforcement.

Pour cette section, nous nous intéressons à l'apprentissage supervisé puisque initialement nos données sont labélisées.

## **2.2 Revue de la littérature : Utilisation du Machine learning dans la modélisation de la Probabilité de défaut**

La probabilité de défaut (PD) est un élément essentiel dans l'évaluation du risque de crédit. Elle représente la probabilité qu'un emprunteur ne puisse pas honorer ses obligations financières en ne remboursant pas ses dettes dans les délais convenus. L'évaluation de la PD revêt une importance capitale pour les institutions financières. Elle leur permet d'évaluer la solvabilité des emprunteurs de manière précise et de prendre des décisions de prêt informées, contribuant ainsi à une gestion efficace et prudente des risques.

Dans cette section, nous examinerons en détail la notion de probabilité de défaut, en étudiant les différentes méthodes de calcul présentées dans la littérature, les facteurs qui peuvent l'influencer, ainsi que l'importance des modèles de probabilité de défaut dans la gestion efficace des risques.

### **Méthodes de Calcul de la Probabilité de Défaut**

Les travaux de [Thomas, 2000] et de [Crook et al., 2007] ont exploré diverses techniques pour modéliser les probabilités de défaut. [Thomas, 2000] a souligné l'importance des méthodes statistiques classiques, tandis que Crook et ses collègues ont examiné l'utilisation de techniques d'apprentissage automatique pour améliorer la précision des modèles de PD. Ces recherches montrent que différentes approches peuvent être utilisées en fonction des données disponibles et des objectifs spécifiques de l'évaluation du risque.

[Baesens et al., 2003] note que bien que les Réseaux Neuraux (NN) soient peu utilisés dans le contexte de la modélisation des PD, ils sont couramment employés dans d'autres domaines bancaires, comme la détection de fraude. Lorsqu'ils sont comparés à d'autres techniques telles que la Régression Logistique, les Arbres de Décision, les K-Plus Proches Voisins (K-NN) et les Machines à Vecteurs de Support (SVM), les NN et les SVM sont souvent parmi les plus performants. [West, 2000] et [Lee et al., 2002] ont également observé cette tendance, indiquant que ces méthodes avancées peuvent offrir des avantages significatifs en termes de précision prédictive.

### **Comparaison des Méthodes**

Les Régressions Logistiques affichent souvent de bons résultats, suggérant qu'il ne faut pas surestimer l'impact des techniques d'apprentissage automatique avancées sur la base de la littérature actuelle. Par exemple, dans une comparaison entre la Régression Logistique et les NN réalisée par [Desai et al., 1996], la Régression Logistique s'est avérée plus performante. Cependant, les auteurs notent que "la précision des réseaux neuronaux dépend des valeurs de certains paramètres contrôlés



par l'investigateur" ([Desai et al., 1996]). Cette variabilité pourrait expliquer les divergences importantes dans les résultats entre les modèles d'apprentissage automatique avancés et la Régression Logistique.

Les Arbres de Décision, souvent associés à une faible précision prédictive [Kraus, 2014], sont désormais fréquemment remplacés par des techniques plus robustes comme les Forêts Aléatoires pour modéliser la probabilité de défaut.

## Défis dans l'Évaluation des Demandes de Crédit

Dans la littérature générale, de nombreuses comparaisons ont été effectuées sur la puissance prédictive des classificateurs, mais peu concernent spécifiquement les données de crédit à la consommation et l'évaluation des demandes de crédit. [Crook et al., 2007] suggèrent que la diversité des données et des structures de données dans les ensembles de données pour l'évaluation du crédit à la consommation rend difficile la détermination d'une méthode optimale universelle [Crook et al., 2007]. Cette diversité nécessite une adaptation des modèles aux caractéristiques spécifiques de chaque ensemble de données pour obtenir des résultats optimaux.

En résumé, l'évaluation de la PD est cruciale pour la gestion des risques de crédit, et bien que de nombreuses méthodes existent, le choix de la meilleure technique dépend souvent du contexte spécifique et des données disponibles. Les recherches continues dans ce domaine visent à affiner ces modèles pour une prédiction plus précise et une gestion plus efficace des risques.

## 2.3 Modèles de Classification

Dans le domaine de l'apprentissage automatique, les modèles de classification constituent une catégorie fondamentale d'outils analytiques utilisés pour résoudre une vaste gamme de problèmes. Ces modèles visent à prédire la classe ou l'étiquette d'une observation en se basant sur ses caractéristiques.

La diversité des modèles de classification offre une multitude d'options pour aborder différents types de données et de problèmes. Chaque modèle a ses propres avantages, inconvénients et domaines d'application privilégiés, et le choix du modèle le plus approprié dépend souvent de la nature spécifique du problème à résoudre ainsi que des données disponibles. En explorant ces modèles avec soin et en les adaptant judicieusement aux besoins de chaque situation, il est possible de réaliser des analyses prédictives précises et efficaces.

Pour estimer la probabilité de défaut, nous devons utiliser des modèles de classification puisque notre variable cible est binaire et représente soit un défaut, soit un non-défaut. Les modèles que nous allons appliquer dans notre cas sont : **KNN, la régression logistique, les forêts aléatoires et la machine à vecteurs supports.**

### Notation :

- $X=x_1, \dots, x_p$  sont les variables d'entrée du modèle.
- $Y$  est la variable binaire, dans notre cas défaut ou non défaut.

### 2.3.1 K-Nearest Neighbors (KNN)

Le modèle des K plus proches voisins (**K-Nearest Neighbors, KNN**), représenté par la figure 2.1 ci-dessous, est une méthode d'apprentissage supervisé. Contrairement à d'autres algorithmes, il n'a pas de phase d'apprentissage explicite; il se contente de mémoriser les données d'entraînement et classe de nouvelles observations en fonction de leur similarité avec les données existantes.

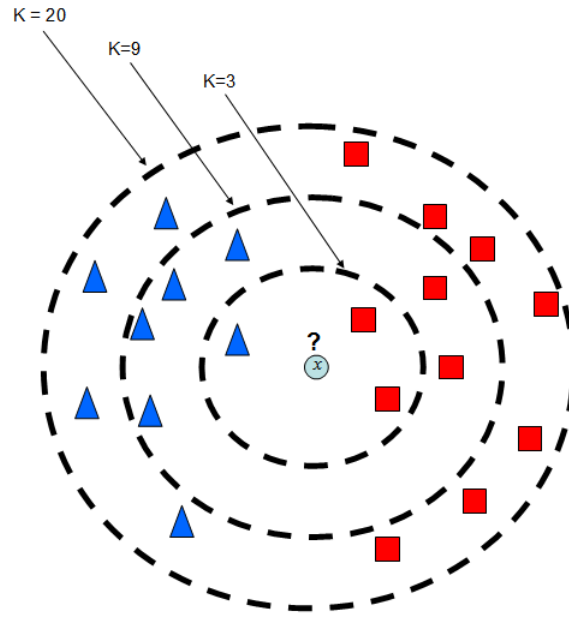


FIGURE 2.1 – Fonctionnement du KNN

D'après le graphe, le modèle identifie les K voisins les plus proches d'une nouvelle observation dans l'espace des caractéristiques, en utilisant une mesure de distance comme la distance euclidienne. Ensuite, il détermine la classe de la nouvelle observation par un vote majoritaire parmi les classes de ses voisins.

#### Calcul des distances

Le calcul des distances entre points est crucial dans KNN. La distance Euclidienne est la plus couramment utilisée et est définie par :

$$d(\mathbf{x}, \mathbf{x}_i) = \sqrt{\sum_{j=1}^n (x_j - x_{i,j})^2} \quad (2.1)$$

où  $\mathbf{x}$  est le vecteur de l'instance de test, et  $\mathbf{x}_i$  est un vecteur de l'instance de formation.

#### Choix des voisins

Après avoir calculé la distance entre le point de test et tous les points de données, les  $k$  plus proches voisins sont sélectionnés. Ces voisins sont déterminés par les  $k$  plus petites distances, permettant de réduire l'influence des outliers.

## Vote majoritaire

La classe attribuée à l'instance de test est déterminée par un vote majoritaire parmi les classes des  $k$  plus proches voisins. Mathématiquement, la classe prédite est donnée par :

$$\text{classe prédite} = \arg \max_{c \in \mathcal{C}} \sum_{i=1}^k \mathbf{1}(y_i = c) \quad (2.2)$$

où  $\mathcal{C}$  est l'ensemble de toutes les classes,  $y_i$  est la classe du  $i^{\text{ème}}$  plus proche voisin, et  $\mathbf{1}$  est la fonction indicatrice.

## 2.3.2 Régression logistique

La régression logistique est une méthode de modélisation statistique utilisée principalement pour la classification binaire. Elle permet d'estimer la probabilité qu'un événement se produise (ou non) en fonction d'un ensemble de variables explicatives. Cette méthode est largement utilisée dans divers domaines, notamment en sciences de la santé, en finance, et en marketing [El Sanharawi and Naudet, 2013].

La fonction logistique, représentée dans la figure 2.2 ci-dessous, souvent appelée fonction sigmoïde, transforme une valeur continue en une valeur comprise entre 0 et 1, ce qui est idéal pour représenter des probabilités.

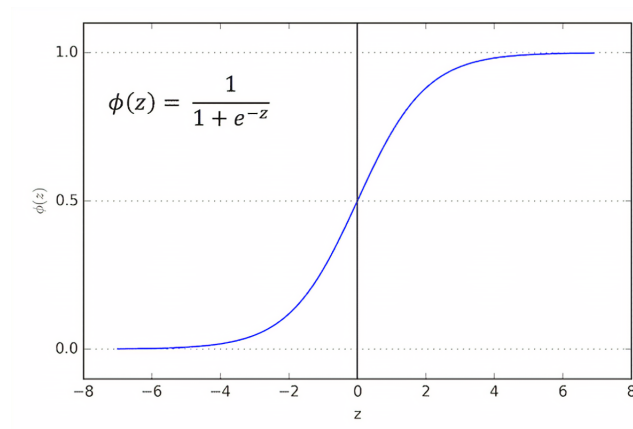


FIGURE 2.2 – Visualisation de la fonction sigmoïde

Elle est définie comme suit :

$$\phi(z) = \frac{1}{1 + e^{-z}} \quad (2.3)$$

Ici,  $z$  est une combinaison linéaire des variables indépendantes et des coefficients :

$$z = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \dots + \beta_p X_p \quad (2.4)$$

## Modèle de Régression Logistique

Le modèle de régression logistique est formulé en termes de probabilité  $p$  de l'événement de succès (classe 1) :

$$p(Y = 1) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 X_1 + \beta_2 X_2 + \dots + \beta_p X_p)}} \quad (2.5)$$

où :

- $p(Y = 1)$  est la probabilité de l'événement de succès.
- $\beta_0$  est l'intercept.
- $\beta_1, \beta_2, \dots, \beta_p$  sont les coefficients des variables explicatives  $X_1, X_2, \dots, X_p$ , respectivement.

## Fonction de Vraisemblance

Pour estimer les coefficients  $\beta$  du modèle, nous maximisons généralement la fonction de vraisemblance (*likelihood*) :

$$L(\beta) = \prod_{i=1}^n p(y_i|X_i, \beta)^{y_i} (1 - p(y_i|X_i, \beta))^{1-y_i} \quad (2.6)$$

où  $y_i$  est la valeur observée (0 ou 1) et  $p(y_i|X_i, \beta)$  est la probabilité prédite par le modèle.

## Fonction de Coût (Perte Logistique)

La fonction de coût, souvent appelée fonction de perte logistique, est utilisée pour optimiser les coefficients lors de l'entraînement du modèle. Cela signifie qu'elle mesure à quel point les prédictions du modèle s'écartent des valeurs réelles observées. L'objectif de l'entraînement est de minimiser cette fonction de coût afin d'améliorer la précision des prédictions du modèle [El Sanharawi and Naudet, 2013].

La fonction de perte logistique est définie comme suit :

$$\text{Cost}(\beta) = -\frac{1}{n} \sum_{i=1}^n [y_i \log(p(y_i|X_i, \beta)) + (1 - y_i) \log(1 - p(y_i|X_i, \beta))] \quad (2.7)$$

où :

- $n$  est le nombre total d'observations.
- $y_i$  est la valeur observée de la variable cible (0 ou 1) pour l'observation  $i$ .
- $p(y_i|X_i, \beta)$  est la probabilité prédite par le modèle que  $y_i = 1$  étant donné les variables explicatives  $X_i$  et les coefficients  $\beta$ .

Cette fonction de coût est une mesure de l'erreur de prédiction du modèle :

- La première partie,  $y_i \log(p(y_i|X_i, \beta))$ , pénalise les cas où le modèle attribue une faible probabilité à un événement qui se produit réellement (c'est-à-dire,  $y_i = 1$ ).
- La seconde partie,  $(1 - y_i) \log(1 - p(y_i|X_i, \beta))$ , pénalise les cas où le modèle attribue une haute probabilité à un événement qui ne se produit pas (c'est-à-dire,  $y_i = 0$ ).

L'entraînement du modèle consiste à trouver les valeurs des coefficients  $\beta$  qui minimisent cette fonction de coût. Cela se fait généralement par des méthodes d'optimisation numériques telles que la descente de gradient. En minimisant la fonction de coût, le modèle ajuste ses coefficients de manière à produire les prédictions les plus proches possibles des valeurs réelles observées.

## Prédiction et classification

Après l'entraînement, le modèle peut être utilisé pour faire des prédictions sur de nouvelles données. Une fois la probabilité  $p(Y = 1)$  calculée, une décision binaire est prise en utilisant un seuil prédéfini [Lafortune et al., 2004].

Le seuil prédéfini, souvent noté  $t$ , est une valeur entre 0 et 1 qui détermine la coupure à partir de laquelle la probabilité estimée  $p(Y = 1)$  est convertie en une classe binaire. Par défaut, ce seuil est souvent fixé à 0,5. Cela signifie que si la probabilité calculée  $p(Y = 1)$  est supérieure ou égale à 0,5, l'événement est classé comme appartenant à la classe 1 (succès). Sinon, il est classé comme appartenant à la classe 0 (échec). Le choix du seuil peut être ajusté en fonction du contexte et des besoins spécifiques du problème, notamment pour gérer le compromis entre la précision et le rappel.

$$\hat{Y} = \begin{cases} 1, & \text{si } p(Y = 1) \geq t \\ 0, & \text{sinon} \end{cases}$$

où  $\hat{Y}$  est la classe prédite et  $t$  est le seuil prédéfini. Le choix du seuil  $t$  peut avoir un impact significatif sur les performances du modèle et est souvent ajusté en fonction des exigences spécifiques de la tâche, telles que la minimisation des faux positifs ou des faux négatifs.

### 2.3.3 Random Forest

Le Random Forest (ou forêt aléatoire) est un algorithme d'apprentissage ensembliste robuste utilisé pour la classification et la régression. Il combine plusieurs arbres de décision pour obtenir une performance accrue et une meilleure généralisation. La figure 2.3 ci-dessous montre le fonctionnement de cet algorithme :

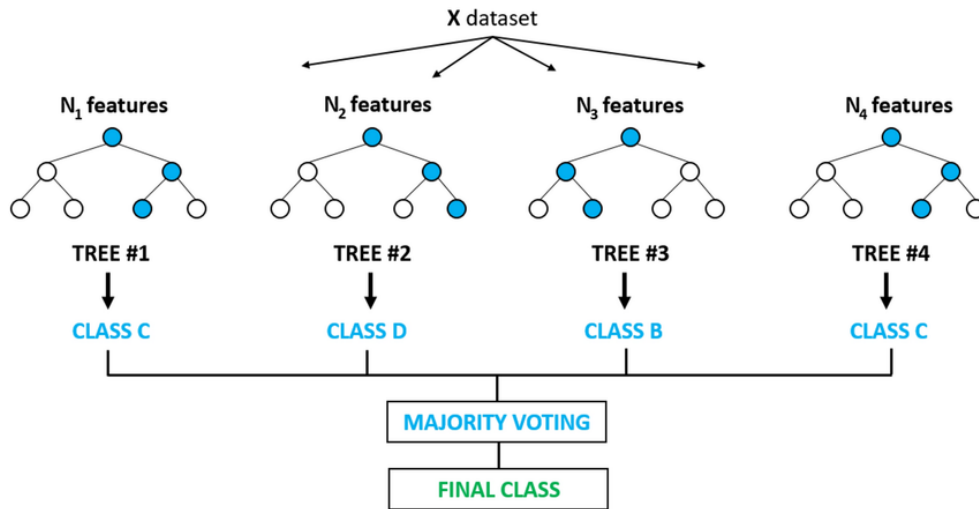


FIGURE 2.3 – Fonctionnement du modèle Random Forest

Le Random Forest construit un ensemble d'arbres de décision où chaque arbre est formé sur un échantillon différent des données. Les décisions finales sont prises par un vote majoritaire pour la classification ou par la moyenne pour la régression.

## Construction des Arbres de Décision

Chaque arbre dans un Random Forest est construit selon l'algorithme suivant :

1. Chaque arbre est cultivé sur un sous-ensemble bootstrap des données d'entraînement.
2. À chaque nœud de l'arbre, un nombre limité de caractéristiques est choisi aléatoirement, et le meilleur critère de division est sélectionné parmi ces caractéristiques.
3. Les nœuds sont divisés jusqu'à atteindre une condition d'arrêt spécifique, souvent lorsque les nœuds contiennent un nombre minimal d'échantillons ou ne peuvent plus être divisés de manière efficace.

## Formule de l'Indice de Gini pour la Sélection des Caractéristiques

L'indice de Gini est un critère de pureté utilisé pour évaluer les divisions :

$$G = 1 - \sum_{i=1}^C p_i^2 \quad (2.8)$$

où  $p_i$  est la proportion de la classe  $i$  parmi les échantillons dans le nœud, et  $C$  est le nombre total de classes.

L'indice de Gini mesure la diversité des classes dans un nœud. Plus l'indice de Gini est proche de zéro, plus le nœud est pur, c'est-à-dire qu'il contient principalement des échantillons d'une seule classe. En revanche, un indice de Gini proche de un indique une répartition uniforme des classes dans le nœud, ce qui signifie qu'il est difficile de faire des prédictions précises à partir de ce nœud.

## Définition Mathématique de l'Importance des Caractéristiques

L'importance d'une caractéristique est souvent évaluée par la somme des réductions de Gini pondérées par la taille des nœuds, agrégées sur tous les arbres :

$$\text{Importance} = \sum_{t=1}^T \sum_{n \in \text{nœuds}} w_n \Delta G_n \quad (2.9)$$

où  $w_n$  est le poids ou la taille du nœud  $n$ ,  $\Delta G_n$  est la réduction de l'indice de Gini due à la division sur la caractéristique, et  $T$  est le nombre d'arbres.

Cette expression mesure l'importance d'une caractéristique en agrégeant la réduction de l'indice de Gini pondérée par la taille des nœuds sur tous les arbres de la forêt. Une caractéristique avec une contribution élevée à la réduction de l'indice de Gini sur de nombreux nœuds dans de nombreux arbres sera considérée comme ayant une importance élevée. En d'autres termes, les caractéristiques qui conduisent à une séparation plus nette et plus homogène des classes sont considérées comme plus importantes pour le modèle. Cette mesure est souvent utilisée pour identifier les caractéristiques les plus influentes dans la prise de décision du modèle et peut aider à la sélection des caractéristiques ou à l'interprétation des résultats du modèle.

### 2.3.4 Support Vector Machine (SVM)

La machine à vecteurs de support (SVM) est un modèle d'apprentissage supervisé puissant et polyvalent capable de réaliser à la fois une classification, une régression et même la détection de valeurs aberrantes. Idéalement utilisée pour des problèmes de classification binaire, SVM se base sur l'idée de trouver un hyperplan qui sépare au mieux deux classes de données dans un espace à haute dimension comme le montre la figure 2.4 ci dessous :

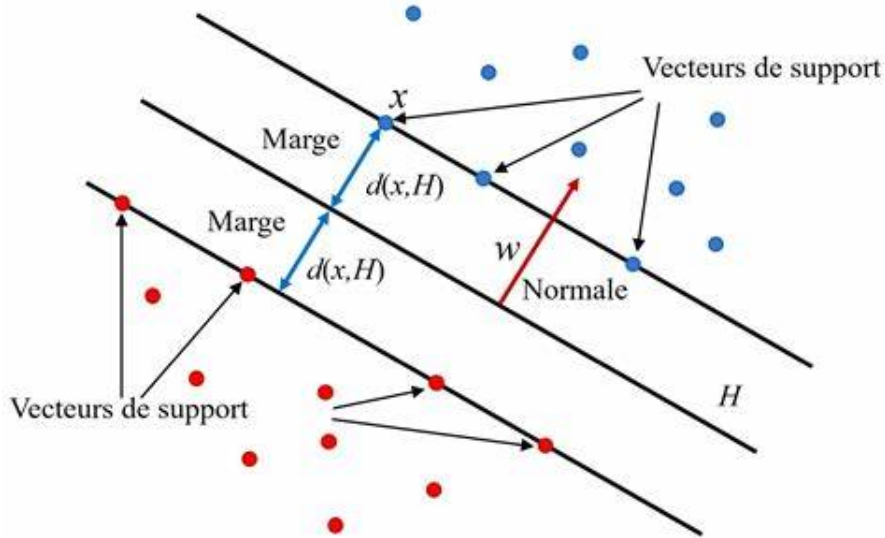


FIGURE 2.4 – Fonctionnement du modèle SVM

Dans le cadre des SVM, l'objectif est de trouver un hyperplan qui maximise la marge entre deux classes tout en minimisant l'erreur de classification. L'hyperplan optimal peut être exprimé sous la forme suivante :

$$\mathbf{w} \cdot \mathbf{x} + b = 0 \quad (2.10)$$

où  $\mathbf{w}$  est le vecteur de poids,  $\mathbf{x}$  est le vecteur des caractéristiques, et  $b$  est un biais[Wang, 2005].

#### Maximisation de la Marge

Pour deux classes linéairement séparables, SVM trouve l'hyperplan avec la plus grande marge minimale entre les classes, définie par les vecteurs de support les plus proches. La marge est optimisée en résolvant le problème suivant :

$$\min_{\mathbf{w}, b} \frac{1}{2} \|\mathbf{w}\|^2 \quad (2.11)$$

sous les contraintes :

$$y_i(\mathbf{w} \cdot \mathbf{x}_i + b) \geq 1, \quad \forall i$$

où  $y_i \in \{-1, 1\}$  sont les étiquettes de classe pour chaque exemple de données[Wang, 2005].

#### Noyaux SVM

Pour traiter les cas non linéairement séparables, SVM utilise des fonctions de noyau qui permettent de projeter les données dans un espace de dimensions plus élevées où un hyperplan linéaire peut être

utilisé pour effectuer la séparation. Un noyau couramment utilisé est le noyau RBF (Radial Basis Function), défini par :

$$K(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\gamma\|\mathbf{x}_i - \mathbf{x}_j\|^2\right) \quad (2.12)$$

où  $\gamma$  est un paramètre qui détermine l'influence d'un seul point de formation[Wang, 2005].

## Optimisation et Implémentation

La solution au problème SVM peut être trouvée en utilisant des méthodes d'optimisation qui ajustent  $\mathbf{w}$ ,  $b$ , et les paramètres du noyau pour minimiser la fonction de perte. Des algorithmes tels que *Sequential Minimal Optimization* (SMO) sont couramment utilisés pour cette optimisation.

## 2.4 Métriques d'Évaluation des Modèles de Classification

Afin de choisir un modèle approprié pour prédire la probabilité de défaut des crédits, une évaluation des modèles de classification est indispensable. Cette évaluation vise à mesurer la capacité d'un modèle à généraliser à de nouvelles données et à identifier ses forces et ses faiblesses. Dans cette section, nous explorons les divers outils utilisés pour évaluer les modèles de classification qui sont : KNN, la régression logistique, le forêt aléatoire et SVM.

### 2.4.1 Matrice de confusion

La matrice de confusion est un outil essentiel en analyse prédictive pour évaluer les performances des modèles de machine learning basés sur la classification. Elle résume les résultats des prédictions en comparant les valeurs prédites avec les valeurs réelles. Cette matrice permet de distinguer les prédictions correctes des prédictions incorrectes pour des tâches de classification binaire. En fonction du nombre de classes cibles, elle peut être d'une taille  $N \times N$ . En examinant visuellement la matrice de confusion, un utilisateur peut évaluer l'exactitude du modèle en se concentrant particulièrement sur les valeurs diagonales qui représentent les classifications correctes[Zheng, 2015].

TABLE 2.1 – Matrice de confusion  
Prédit

	Prédit	
	Positif	Négatif
Réel	VP	FN
	FP	VN

Avec

- **VP (Vrais Positifs)** : Instances de non-défaut de crédit correctement identifiées comme non-défaut par le modèle. Cela signifie que le modèle a correctement prédit qu'un emprunteur ne ferait pas défaut.
- **VN (Vrais Négatifs)** : Instances de défaut de crédit correctement identifiées comme défaut par le modèle. Cela indique que le modèle a correctement prédit qu'un emprunteur ferait défaut.



- **FP (Faux Positifs)** : Instances de défaut de crédit incorrectement classées comme non-défaut par le modèle. Cela indique que le modèle n’a pas détecté qu’un emprunteur ferait défaut alors qu’il le ferait effectivement.
- **FN (Faux Négatifs)** : Instances de non-défaut de crédit incorrectement classées comme défaut par le modèle. Cela montre que le modèle a faussement prédit qu’un emprunteur ferait défaut alors qu’il ne le ferait pas.

La matrice de confusion est représentée sous la forme d’une structure carrée où les colonnes dénotent les valeurs prédites tandis que les lignes expriment les valeurs réelles.

## 2.4.2 Mesures de performance

En analysant les métriques suivantes, nous serons en mesure de mieux comprendre les forces et les faiblesses des modèles évalués, ce qui facilitera la sélection du modèle le plus approprié pour notre tâche de prédiction de la probabilité de défaut des crédits.

### Exactitude (Accuracy)

L’exactitude mesure la proportion d’observations correctement classées parmi l’ensemble des observations.

$$\text{Accuracy} = \frac{VP + VN}{VP + VN + FP + FN} \quad (2.13)$$

L’exactitude indique la proportion globale de prédictions correctes faites par le modèle. Une haute valeur d’accuracie signifie que le modèle effectue bien en termes de prédictions globales, mais cela peut être trompeur dans des cas de classes déséquilibrées [Zheng, 2015].

### Rappel (Recall)

Le rappel mesure la proportion d’observations positives correctement identifiées parmi toutes les observations réellement positives.

$$\text{Recall} = \frac{VP}{VP + FN} \quad (2.14)$$

Un rappel élevé signifie que le modèle identifie la plupart des exemples positifs. C’est crucial dans des situations où il est important de minimiser les faux négatifs, comme dans les diagnostics médicaux où manquer un cas positif peut avoir des conséquences graves [Zheng, 2015].

### Précision (Precision)

La précision mesure la proportion d’observations positives parmi les prédictions positives faites par le modèle.

$$\text{Precision} = \frac{VP}{VP + FP} \quad (2.15)$$

La précision évalue la fidélité des prédictions positives du modèle. Elle est particulièrement importante dans les situations où les faux positifs sont coûteux ou problématiques, par exemple dans la détection de spam où classer un courriel légitime comme spam est indésirable [Zheng, 2015].

## F1-score

Le F1-score est la moyenne harmonique de la précision et du rappel.

$$\text{F1-score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (2.16)$$

Le F1-score fournit une mesure équilibrée de la précision et du rappel. C'est utile lorsque l'on cherche un compromis entre les deux métriques, surtout dans des situations où les classes sont déséquilibrées [Zheng, 2015].

Nous combinons le F1-score avec d'autres métriques d'évaluation pour obtenir une vue d'ensemble du résultat, car nous ne sommes pas certains de ce que notre classificateur maximise : la précision ou le rappel [Zheng, 2015].

## Spécificité

La spécificité, également appelée taux de vrais négatifs, est une métrique utilisée pour évaluer la performance d'un modèle de classification. Elle mesure la proportion des instances négatives correctement identifiées comme négatives par le modèle. En d'autres termes, la spécificité indique la capacité du modèle à reconnaître correctement les instances qui n'appartiennent pas à la classe positive.

La spécificité est définie par la formule suivante :

$$\text{Spécificité} = \frac{\text{VN}}{\text{VN} + \text{FP}} \quad (2.17)$$

Une spécificité élevée indique que le modèle a une forte capacité à identifier correctement les instances négatives, réduisant ainsi le nombre de faux positifs. Dans le contexte des crédits, une haute spécificité signifie que le modèle est efficace pour identifier les clients qui ne feront pas défaut. Cependant, une spécificité élevée seule n'est pas suffisante pour évaluer la performance globale du modèle, car elle ne prend pas en compte la capacité du modèle à détecter les cas positifs (défauts de crédit).

## Sensibilité

La sensibilité, également connue sous le nom de rappel ou taux de vrais positifs, est une autre métrique clé pour évaluer la performance d'un modèle de classification. Elle mesure la proportion des instances positives correctement identifiées comme positives par le modèle.

La sensibilité est définie par la formule suivante :

$$\text{Sensibilité} = \frac{\text{VP}}{\text{VP} + \text{FN}} \quad (2.18)$$

Une sensibilité élevée indique que le modèle a une forte capacité à identifier correctement les instances positives, réduisant ainsi le nombre de faux négatifs. Dans le contexte des crédits, une haute sensibilité signifie que le modèle est efficace pour détecter les clients qui feront défaut.

### 2.4.3 Courbe ROC et AUC

#### Courbe ROC

La courbe ROC (*Receiver Operating Characteristic*) est un outil graphique qui illustre la performance d'un modèle de classification binaire. Elle trace le taux de vrais positifs (TPR ou rappel) contre le taux de faux positifs (FPR) à différents seuils de classification [Zheng, 2015].

Le taux de vrais positifs (TPR) est défini comme :

$$\text{TPR} = \frac{VP}{VP + FN} \quad (2.19)$$

Le taux de faux positifs (FPR) est défini comme :

$$\text{FPR} = \frac{FP}{FP + VN} \quad (2.20)$$

Le TPR mesure la proportion d'instances positives correctement classées parmi toutes les instances réellement positives, tandis que le FPR mesure la proportion d'instances négatives incorrectement classées comme positives parmi toutes les instances réellement négatives.

La figure 2.5 présente la courbe ROC :

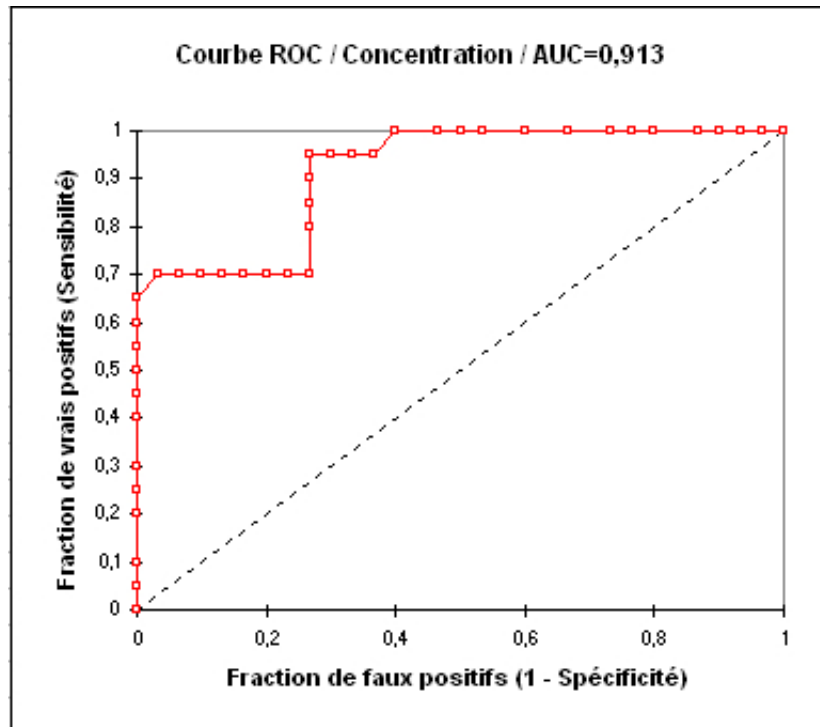


FIGURE 2.5 – Schéma Explicatif de la Courbe ROC

En analysant la courbe ROC, nous pouvons observer le compromis entre le rappel et le taux de faux positifs. Idéalement, un modèle performant devrait avoir une courbe ROC qui se rapproche le plus possible du coin supérieur gauche du graphique. Cela indique que le modèle atteint un haut rappel

tout en maintenant un faible taux de faux positifs. En revanche, une courbe ROC qui se rapproche de la diagonale (ligne de hasard) indique une performance médiocre, où le modèle ne fait pas mieux qu'une classification aléatoire.[Zheng, 2015].

### AUC (Area Under the Curve)

L'AUC (*Area Under the ROC Curve*) est une mesure quantitative de la performance d'un modèle de classification binaire. Elle représente l'aire sous la courbe ROC comme le montre la figure 2.6 ci-dessous :

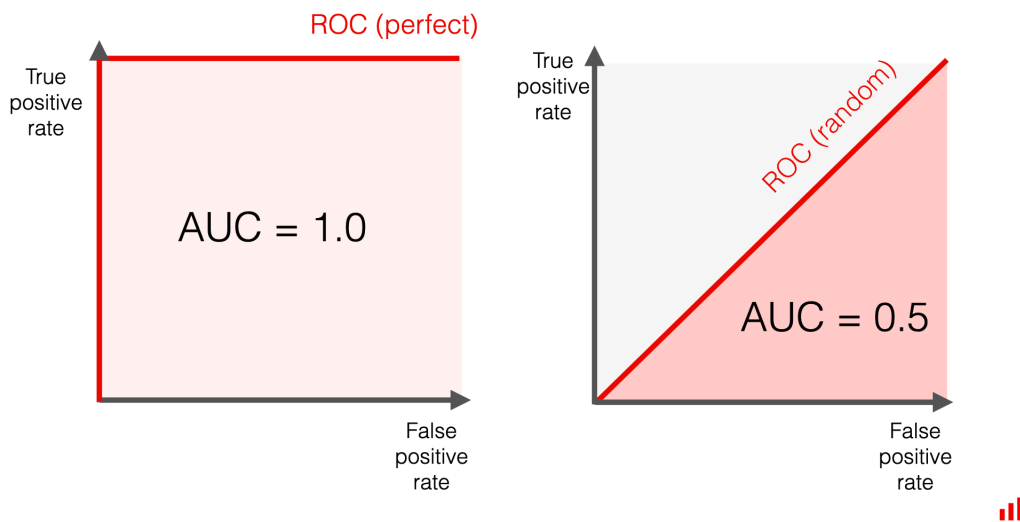


FIGURE 2.6 – Schéma Explicatif de l'AUC

L'AUC varie de 0 à 1. Une AUC de 0.5 indique que le modèle ne fait pas mieux que le hasard, tandis qu'une AUC de 1.0 indique un modèle parfait. Une AUC élevée indique une meilleure capacité du modèle à distinguer entre les classes positives et négatives[Zheng, 2015].

## 2.5 Conclusion

Dans ce chapitre, nous avons exploré le fondement théorique des modèles de machine learning pour l'estimation de la probabilité de défaut, ainsi que l'approche d'évaluation de ces modèles. Cette estimation de la probabilité de défaut va nous servir à sélectionner les demandes de crédits à haut risque de défaut. Les algorithmes permettant de faire cette sélection seront présentés dans le chapitre suivant.

# Chapitre 3

## Algorithme d'Apprentissage par Renforcement pour la Sélection des Crédits

Ce chapitre se concentre sur la partie théorique d'une méthode spécifique d'apprentissage par renforcement, l'algorithme du bandit multi-bras, afin de réaliser la sélection des demandes de crédit à haut risque à traiter manuellement répartis dans différents intervalles de montants. Le choix de cet algorithme est motivé par sa capacité à adresser efficacement les problèmes séquentiels de décision où chaque action prise influence les résultats futurs, ce qui est crucial dans le contexte financier.

Dans un premier temps, nous définirons l'apprentissage par renforcement et ses concepts clés, essentiels pour comprendre les mécanismes sous-jacents de l'algorithme du bandit multi-bras. Ensuite, nous présenterons en détail cet algorithme, en explorant ses différents aspects et en illustrant comment il peut être appliqué pour optimiser la sélection des demandes de crédit. Ce chapitre vise à établir une fondation solide sur laquelle des stratégies de sélection de crédits peuvent être construites, permettant ainsi une gestion plus précise des risques financiers.

### 3.1 Apprentissage par Renforcement

L'Apprentissage par Renforcement, ou le Reinforcement Learning en anglais, est une branche de l'apprentissage automatique où un agent apprend à prendre des décisions en interagissant avec un environnement. L'objectif principal de l'apprentissage par renforcement est d'apprendre une politique d'action optimale qui guide l'agent à prendre des décisions afin de maximiser une récompense cumulative sur le long terme. À chaque étape, l'agent prend une action en fonction de l'état actuel de l'environnement, reçoit une récompense ou une pénalité, puis ajuste sa politique d'action en conséquence [Sutton and Barto, 2018].

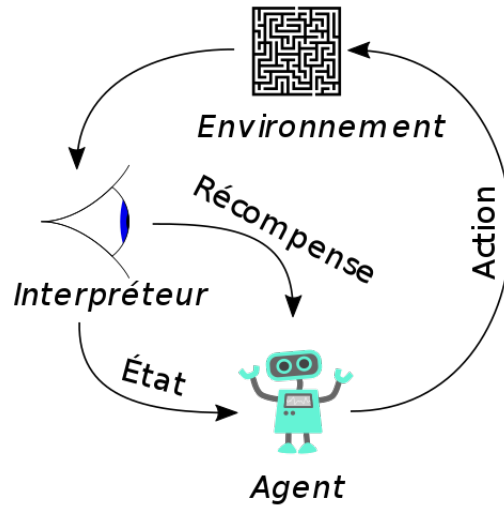


FIGURE 3.1 – Fonctionnement de l’Apprentissage par Renforcement

Cet apprentissage autonome permet à l’agent de s’améliorer progressivement en explorant différentes actions et en apprenant des conséquences de ses actions précédentes. Cela rend le Reinforcement Learning particulièrement adapté pour les environnements complexes où les données étiquetées ne sont pas facilement disponibles, mais où l’agent peut expérimenter et apprendre de ses interactions avec l’environnement [Sutton and Barto, 2018].

## Notions clés

### — Agent :

Un agent est une entité qui prend des décisions dans un environnement pour maximiser une certaine notion de récompense cumulative. Mathématiquement, à partir de l’état  $s_t$  de l’environnement, l’agent exécute une action  $a_t$  basée sur une politique  $\pi$ , visant à optimiser cette politique au fil du temps grâce à un jeu de récompenses [Sutton and Barto, 2018].

### — État ( $s$ ) :

Un état  $s$  est une description de la situation actuelle dans l’environnement. Dans le cadre des processus de décision de Markov (MDP), l’état suivant  $s_{t+1}$  dépend seulement de l’état actuel  $s_t$  et de l’action  $a_t$  prise par l’agent [Sutton and Barto, 2018].

### — Action ( $a$ ) :

Une action est une intervention de l’agent sur l’environnement, qui mène à une transition d’état et génère une récompense. L’ensemble des actions disponibles dans un état donné est souvent noté par  $A$  [Sutton and Barto, 2018].

### — Récompense ( $r$ ) :

La récompense est un signal numérique que l’agent reçoit de l’environnement après avoir exécuté une action. Elle mesure la performance de l’action effectuée, aidant l’agent à apprendre quelle action est bénéfique à long terme [Sutton and Barto, 2018].

### — Politique ( $\pi$ ) :

Une politique  $\pi$  est une stratégie que l’agent utilise pour choisir ses actions. Formellement, c’est une fonction de l’ensemble des états  $S$  vers l’ensemble des actions  $A$ ,  $\pi : S \rightarrow A$ , qui peut être déterministe ou stochastique (probabiliste) [Sutton and Barto, 2018].

— **Fonction de valeur ( $V$  ou  $Q$ ) :**

La fonction de valeur  $V(s)$  évalue la récompense totale attendue à partir d’un état  $s$ , suivant une politique  $\pi$ . La fonction d’action-valeur  $Q(s, a)$  évalue la récompense attendue après avoir pris une action  $a$  dans l’état  $s$ , puis en suivant la politique  $\pi$  [Sutton and Barto, 2018].

— **Apprentissage :**

L’apprentissage en apprentissage par renforcement implique la mise à jour des estimations de la fonction de valeur ou de la politique basée sur l’expérience—actions prises, récompenses reçues et nouveaux états observés [Sutton and Barto, 2018].

En combinant des concepts clés tels que les états, les actions, les récompenses et les politiques, les algorithmes d’apprentissage par renforcement peuvent être utilisés pour entraîner des agents à accomplir une variété de tâches, allant de jeux informatiques complexes à la gestion de systèmes autonomes dans le monde réel. Avec des avancées continues dans ce domaine, l’apprentissage par renforcement promet de jouer un rôle de plus en plus important dans la construction de systèmes intelligents capables d’apprendre et de s’adapter à leur environnement [Sutton and Barto, 2018].

L’algorithme du bandit multibras est une composante essentielle de l’apprentissage par renforcement. Il illustre comment les principes de l’apprentissage par renforcement peuvent être appliqués à des problèmes de prise de décision où les actions sont prises dans un contexte séquentiel et interactif, comme celui décrit dans notre cas d’application avec la sélection de crédits à risque.

## 3.2 Algorithme de Bandit Multi-Bras

Le problème du bandit à plusieurs bras est un cas emblématique d’apprentissage par renforcement où une machine à sous est équipée de  $n$  bras, chacun associé à une distribution de probabilité de succès spécifique. Traction d’un bras peut entraîner une récompense stochastique de  $R = +1$  en cas de réussite ou  $R = 0$  en cas d’échec. L’objectif est de tirer les bras séquentiellement afin de maximiser la récompense totale accumulée à long terme. La figure 3.2 ci-dessous montre un exemple des machines à sous utilisées dans ce problème [Bubeck et al., 2012] :

Les joueurs cherchant à maximiser leurs gains tirent le levier d’une machine à sous pour obtenir une récompense. Toutefois, face à plusieurs machines disponibles, il est difficile de déterminer par laquelle commencer afin d’optimiser ses profits. Avec le temps, chaque machine distribue des récompenses selon une distribution de probabilité inconnue. Certaines machines ont une probabilité plus élevée de distribuer des récompenses que d’autres, une information qui demeure initialement incertaine. Le défi consiste à maximiser la somme des récompenses obtenues lors d’une série de tirages des leviers des différentes machines [Bubeck et al., 2012].

Dans notre cadre, les machines à sous symbolisent divers intervalles de montants de crédit. L’action de tirer le levier correspond à choisir un intervalle spécifique, puis à sélectionner un crédit à l’intérieur de cet intervalle. Ensuite, nous évaluons la probabilité de défaut pour ce crédit en utilisant nos



FIGURE 3.2 – Les machines à sous

prédictions. Après, nous calculons une récompense pour ce crédit en particulier, basée sur la probabilité de défaut prédite et le montant du crédit. Selon une stratégie bien déterminée et en prenant en compte la récompense obtenue, il peut soit changer d'intervalle soit continuer de sélectionner de l'intervalle déjà choisi. A la fin du processus, notre algorithme sélectionne les crédits les plus risqués pour une vérification manuelle, en temps réel.

### 3.2.1 Définition

Un bandit à plusieurs bras est défini par un ensemble de variables aléatoires  $X_{i,k}$  où :

- $1 \leq i \leq N$ , indiquant que  $i$  est le bras du bandit ;
- $k$  est l'indice du tour de jeu du bras  $i$ .

On suppose que les jeux successifs  $X_{i,1}, X_{j,2}, X_{k,3}, \dots$  sont distribués indépendamment, mais nous ne connaissons pas les distributions de probabilité des variables aléatoires [Bubeck et al., 2012].

L'idée est qu'un joueur itère des tours, observant la récompense du bras après chaque tour, et peut ajuster sa stratégie à chaque fois. L'objectif est de maximiser la somme des récompenses collectées sur tous les tours[Bubeck et al., 2012].

Les stratégies de bandit à plusieurs bras visent à apprendre une politique  $\pi(k)$ , où  $k$  est le tour de jeu[Bubeck et al., 2012].

### 3.2.2 Exploration et Exploitation

L'objectif est de sélectionner systématiquement les actions les plus récompensantes. Par conséquent, les stratégies utilisées devraient trouver un équilibre entre exploiter les actions qui semblent les plus prometteuses selon les données disponibles et explorer d'autres actions potentielles[Bubeck et al., 2012].

Cet équilibre entre exploitation et exploration pose un dilemme qui nous pousse à adopter des stratégies minimisant le regret.

#### Définition - Regret



Étant donné une politique  $\pi$  et  $t$  tirages de bras, le regret est formellement défini comme :

$$R(r, t) = t \times \max_a Q^*(a) - \sum_{k=1}^t E[X_{\pi(k),k}] \quad (3.1)$$

où  $Q^*(a)$  est le retour moyen réel de jouer avec le bras  $a$ . Bien sûr, nous ne connaissons pas  $Q^*(a)$  - sinon, nous pourrions simplement jouer  $\operatorname{argmax}_a Q^*(a)$  à chaque tour.

### 3.2.3 Méthodes de minimisation du regret

L'utilisation de stratégies efficaces pour minimiser le regret est cruciale dans le domaine des bandits manchots. Deux approches couramment utilisées sont la méthode epsilon-greedy et la stratégie des bornes supérieures de confiance (UCB). Ces méthodes visent à équilibrer l'exploration des actions potentiellement bénéfiques avec l'exploitation des actions connues pour offrir des récompenses élevées [Bubeck et al., 2012].

#### 1. Epsilon-greedy

La méthode  $\epsilon$ -greedy est une technique efficace pour équilibrer les phases d'exploration et d'exploitation. Dans cet algorithme, le paramètre  $\epsilon$  détermine le degré d'exploration par rapport à l'exploitation [Tran-anh et al., 2010].

Lorsque vient le moment de choisir une action, le processus est le suivant :

- Avec une probabilité  $1-\epsilon$ , nous optons pour le bras ayant la valeur  $Q$ <sup>1</sup> maximale :  $\operatorname{argmax}_a Q(a)$  (stratégie d'exploitation). En cas d'égalité entre plusieurs actions ayant la même valeur  $Q$ , le choix est fait aléatoirement.
- Avec une probabilité  $\epsilon$ , nous sélectionnons un bras au hasard selon une distribution uniforme (stratégie d'exploration).

Le choix optimal pour  $\epsilon$  varie selon le problème rencontré. Toutefois, des valeurs généralement comprises entre 0,05 et 0,2 sont souvent efficaces, permettant de combiner l'exploitation des connaissances acquises et la découverte de nouvelles opportunités [Tran-anh et al., 2010].

#### 2. Stratégie des bornes supérieures de confiance (UCB)

Une méthode particulièrement efficace dans le domaine des bandits manchots est la stratégie des bornes supérieures de confiance (UCB1). Cette stratégie consiste à sélectionner la prochaine action en utilisant la formule suivante :

$$\operatorname{argmax}_a \left( Q(a) + 2\sqrt{\frac{\ln t}{N(a) + 1}} \right) \quad (3.3)$$

Ici,  $t$  représente le nombre de tours déjà effectués, tandis que  $N(a)$  désigne le nombre de fois où l'action  $a$  a été choisie lors des tours précédents. Il est à noter que le terme sous la racine carrée est

---

1.  $Q$  représente une fonction qui donne le retour moyen pour chaque action  $a$ . Cette fonction est utilisée dans le calcul du regret, qui mesure la différence entre les récompenses obtenues en suivant une certaine politique et les récompenses maximales possibles.

indéfini lorsque  $N(a) = 0$ . Pour pallier cela, il est courant de consacrer les premiers  $N$  tours à choisir chaque bras  $N$  fois [Lattimore and Szepesvári, 2020].

Le premier terme de la formule favorise l'exploitation en attribuant une valeur élevée aux actions ayant offert des récompenses importantes. Le second terme, quant à lui, encourage l'exploration en privilégiant les actions moins explorées, c'est-à-dire celles pour lesquelles  $N(a)$  est faible comparativement aux autres actions [Lattimore and Szepesvári, 2020].

Le paramètre  $c$  dans l'algorithme UCB contrôle le degré d'exploration : une valeur plus élevée de  $c$  augmente l'exploration, tandis qu'une valeur plus faible favorise l'exploitation.

L'objectif est d'apprendre la fonction  $Q$ , qui donne le retour moyen pour chaque action  $a$ , de manière à ce qu'elle se rapproche de la véritable fonction  $Q$  (inconnue), que nous appellerons  $Q^*$ . À chaque tour, nous sélectionnons l'action  $a$  maximisant l'expression à l'intérieur des parenthèses. Si le bras  $a$  est optimal, alors nous voulons que la condition suivante soit vérifiée pour toutes les autres actions  $b$  :

$$c \ln t \cdot Q(b) + \sqrt{\frac{2 \ln t}{N(b) + 1}} < Q^*(a) \quad (3.4)$$

Si cette condition est remplie, nous avons une certaine confiance dans le fait que  $Q(a)$  est optimal. Cependant, si  $N(b)$  est faible pour certaines actions, cette confiance diminue [Lattimore and Szepesvári, 2020].

Si par hasard l'expression ci-dessus ne tient pas pour l'action optimale  $a$ , alors  $a$  n'est pas sélectionné, bien qu'il devrait l'être. Nous souhaitons que cela se produise avec une probabilité minimale afin de réduire le pseudo-regret. C'est là qu'intervient la valeur  $\ln t$  dans l'expression [Lattimore and Szepesvári, 2020].

### 3.2.4 Évaluation du modèle

L'évaluation des performances des algorithmes de bandit multi-bras repose généralement sur plusieurs critères clés, notamment la récompense cumulative obtenue au fil du temps, le regret cumulé et la convergence vers les meilleures actions [Lattimore and Szepesvári, 2020].

#### Convergence

La convergence fait référence à la capacité de l'algorithme à identifier et à exploiter rapidement les meilleures actions disponibles. Un bon algorithme de bandit multi-bras devrait converger vers les actions optimales avec un nombre minimal de tirages de bras, maximisant ainsi la récompense cumulative tout en minimisant le regret cumulé [Lattimore and Szepesvári, 2020].

L'évaluation du modèle implique souvent des expériences empiriques utilisant des simulations ou des tests sur des ensembles de données réels pour comparer les performances des différents algorithmes de bandit multi-bras. Les résultats de ces évaluations aident à sélectionner l'algorithme le plus approprié pour une application spécifique et à ajuster les paramètres pour optimiser les performances [Lattimore and Szepesvári, 2020].

### 3.3 Conclusion

Dans ce chapitre, nous avons exploré le domaine captivant de l'apprentissage par renforcement, où les agents apprennent à prendre des décisions en interagissant avec leur environnement pour maximiser une récompense cumulative à long terme. En explorant des concepts clés tels que les états, les actions, les récompenses et les politiques, nous avons posé les fondements nécessaires à la compréhension de ce domaine afin de sélectionner les demandes de crédits à haut risque de défaut de chaque intervalle. Nous avons ensuite examiné en détail le problème classique du bandit à plusieurs bras, qui constitue un exemple emblématique d'apprentissage par renforcement pour les cas des problèmes séquentiels. Nous présentons maintenant dans le chapitre suivant une application pratique des modèles de machine learning pour estimer la probabilité de défaut afin de pouvoir identifier les demandes de crédits à haut risque de défaut.

## Chapitre 4

# Implémentation des Modèles de Machine Learning et Estimation de la Probabilité de Défaut

Ce chapitre se concentre sur l'application pratique des modèles de machine learning pour estimer la probabilité de défaut. Avant d'entrer dans les détails des algorithmes, une analyse minutieuse des données est entreprise. Cela débute par une présentation approfondie de la base de données utilisée, suivie par des étapes d'analyse univariée et bivariée pour comprendre les caractéristiques des données. Ensuite, un processus de nettoyage des données est mis en œuvre pour traiter les données manquantes et repérer les valeurs aberrantes. Une fois les données préparées, les variables sont transformées, notamment par la conversion des catégories en valeurs numériques et la standardisation des variables numériques. Enfin, les modèles de machine learning, tels que la régression logistique, sont mis en place, suivis d'une comparaison détaillée de leurs performances. L'objectif de ce chapitre est de fournir une méthodologie claire et rigoureuse pour l'analyse et la modélisation des données dans le contexte de l'estimation de la probabilité de défaut.

### 4.1 Analyse de données

Dans cette section, nous allons analyser les données disponibles. Il s'agit d'une étape essentielle dans tout processus de recherche ou d'étude qui implique l'interprétation de données. Elle englobe un ensemble de techniques et de méthodes visant à explorer, nettoyer, transformer et interpréter les données afin d'extraire des informations utiles et significatives.

#### 4.1.1 Présentation de la base de données

La base de données que nous possédons est un échantillon de données concernant les clients nécessitant des fonds pour régler des factures liées à la construction ou à la rénovation de leurs domiciles. Cette base de données a été établie en 2018 à partir des informations fournies en ligne via la plateforme développée par Quion, spécifiquement dédiée aux demandes de crédits à la consommation. Ces crédits portent sur de petits montants, inférieurs à 5 000 euros. Elle se compose de 9 489 demandes de crédits et 10 colonnes, comprenant 6 variables quantitatives et 4 variables qualitatives, dont 3 sont binaires.

Les variables *person\_age*, *person\_income*, *person\_home\_ownership* et *person\_emp\_length* représentent les aspects socio-professionnels des clients, tandis que *cb\_person\_cred\_hist\_length* et *cb\_person\_default\_on\_file* reflètent leur historique avec la banque, *loan\_amnt* et *loan\_int\_rate* fournissent des détails sur le crédit sollicité, et une variable indique le statut du prêt sur toute sa durée, *loan\_status*. Pour une description détaillée de chaque variable et de son type, veuillez consulter l'annexe B.

Les variables incluses dans notre ensemble de données ont été spécifiquement sélectionnées suite à une demande explicite. Elles ont été choisies pour répondre aux besoins spécifiques de notre projet en nous basant sur les meilleures pratiques identifiées dans la littérature[Siddiqi, 2006].

### 4.1.2 Analyse univariée

L'analyse univariée est une composante essentielle de toute étude statistique, offrant une compréhension approfondie des caractéristiques individuelles d'un ensemble de données sans tenir compte des relations potentielles entre ces caractéristiques. Cette méthode permet d'explorer la distribution et les propriétés des variables individuelles, offrant ainsi des informations précieuses pour orienter la modélisation statistique et l'interprétation des résultats.

#### Variable Cible

Le status du crédit (*loan\_status*) indique si un client est arrivé à payer son crédit ou non. Cette variable est représentée par la figure 4.1 ci-dessous.

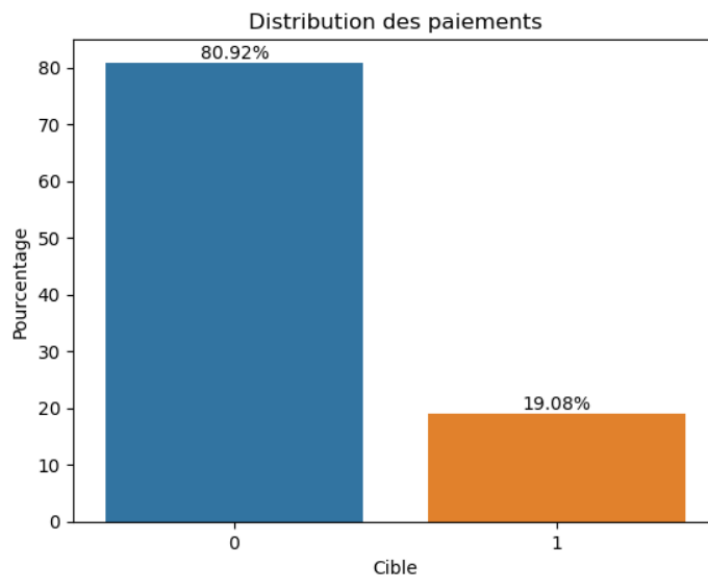


FIGURE 4.1 – Distribution des clients selon le défaut/non défaut

Nous observons que le taux de clients en défaut (représenté par 1) s'élève à 19.08%, ce qui est nettement inférieur au taux de clients ayant honoré leurs paiements de crédit (représenté par 0), qui atteint 80.92%. Ce déséquilibre peut influencer les résultats de la modélisation et causer des problèmes de classification.

## Variables explicatives

Dans cette partie, nous examinerons de près plusieurs variables explicatives pertinentes pour notre analyse.

- **Durée d'emploi (en années) :** La durée d'emploi, exprimée en années, représente la période pendant laquelle chaque demandeur de crédit a été employé au moment de la demande de prêt. Cette variable fournit un aperçu de la stabilité professionnelle des emprunteurs et peut influencer leur capacité à rembourser le prêt. La figure 4.2 ci-dessous montre la distribution des clients selon la durée d'emploi en années :

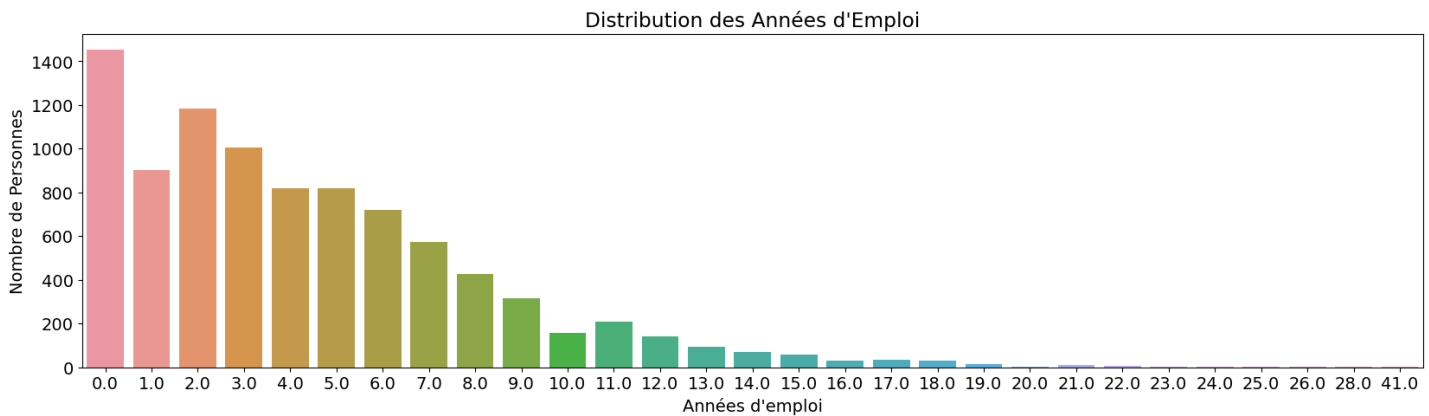


FIGURE 4.2 – Distribution des clients selon la durée d'emploi

Nous observons une forte concentration de demandeurs de crédit dans les premières années d'emploi. À mesure que la durée d'emploi augmente, le nombre de demandeurs de crédit diminue progressivement. Cette tendance peut indiquer que les personnes ayant une plus longue ancienneté ont peut-être déjà établi leur solvabilité et n'ont pas besoin de financement supplémentaire.

- **Grade du prêt :** La variable "loan\_grade" représente la catégorie de risque attribuée à un emprunteur en fonction de son historique financier, de sa solvabilité et d'autres critères pertinents. Dans de nombreux systèmes de prêt, les emprunteurs sont classés en différentes catégories de risque, généralement désignées par des lettres comme A, B, C, etc.

Par exemple, un emprunteur classé dans la catégorie A peut avoir un bon historique de crédit et être considéré comme présentant un faible risque de défaut de paiement, tandis qu'un emprunteur classé dans la catégorie F peut être considéré comme présentant un risque plus élevé en raison d'un crédit moins favorable ou d'autres facteurs.

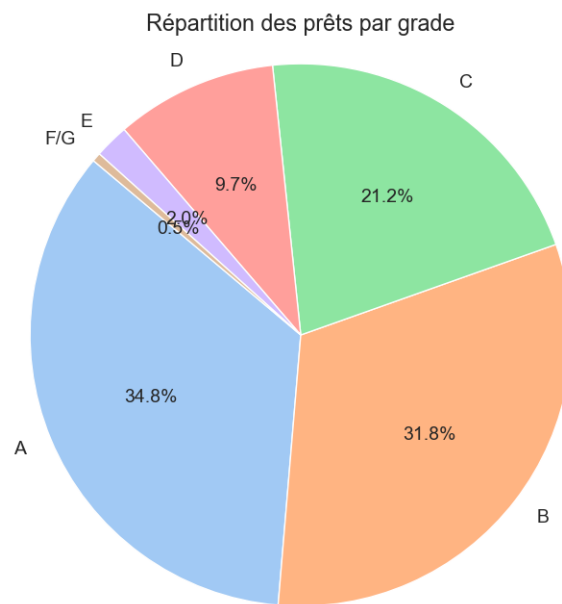


FIGURE 4.3 – Distribution des demandes de crédits selon le grade

Une part importante des prêts se situe dans les catégories à faible risque (A, B et C). À mesure que l'on descend vers les catégories inférieures (D, E, F et G), le pourcentage de prêts diminue.

Ce diagramme circulaire montre un très faible pourcentage de prêts dans la catégorie de risque la plus élevée (G).

- **Revenu** : cette variable représente le revenu annuel de chacun des demandeurs de crédits et elle est répartie comme suit :

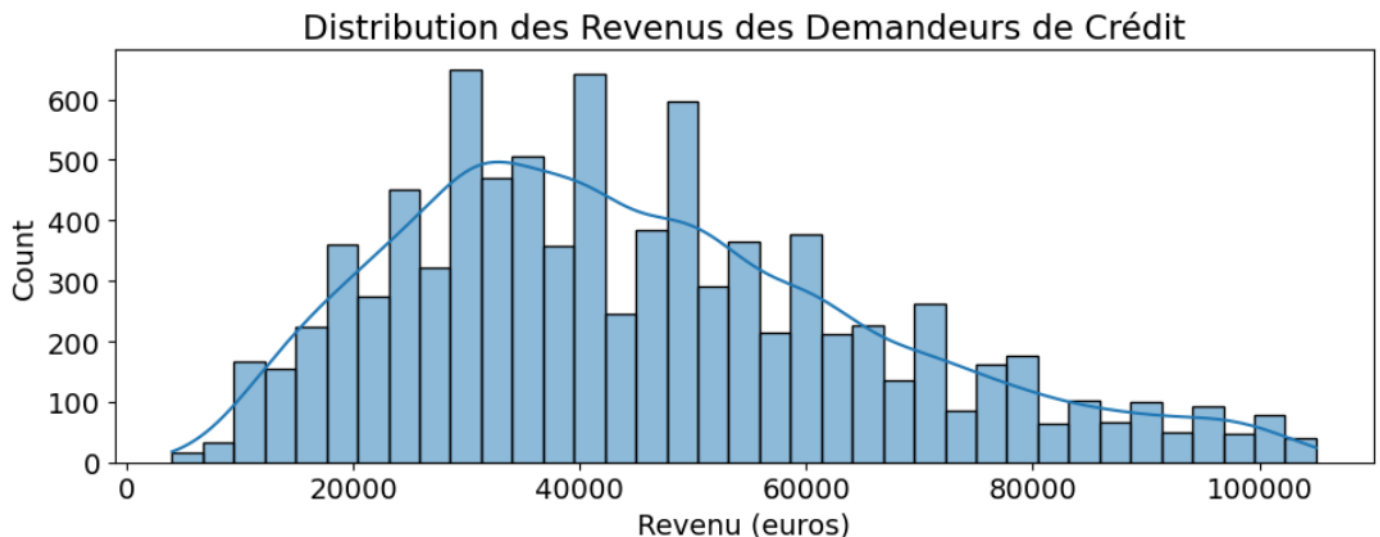


FIGURE 4.4 – Distribution des demandeurs de crédits selon le Revenu

La distribution est asymétrique à droite, ce qui signifie qu'il y a plus de demandeurs de crédit ayant un faible revenu que de demandeurs de crédit ayant un revenu élevé. Le revenu médian, c'est-à-dire

le revenu qui sépare la population en deux groupes de taille égale, est d'environ 40 000 euros.

- **Age** : cette variable représente le l'âge de chacun des demandeurs de crédits et elle est répartie comme suit :

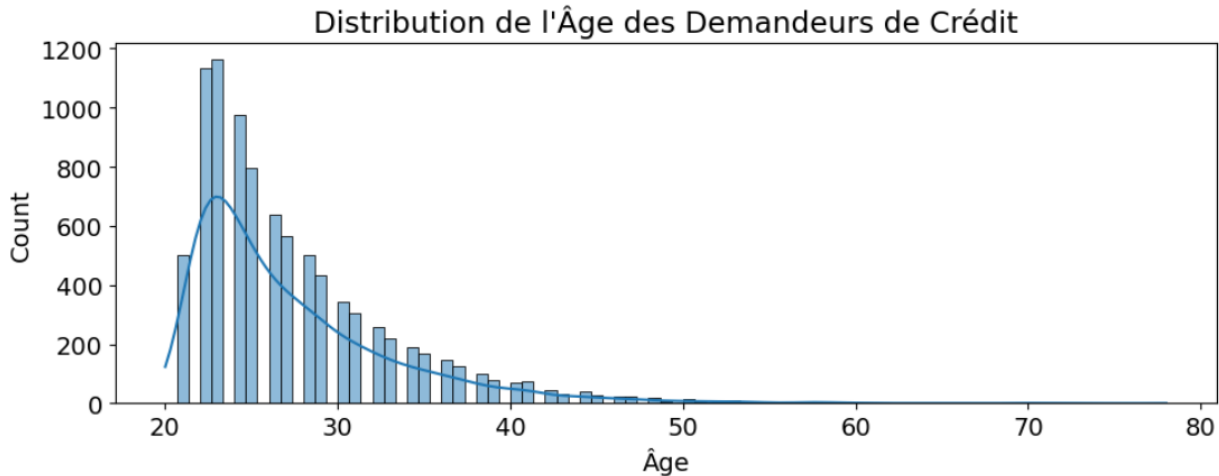


FIGURE 4.5 – Distribution de l'âge des demandeurs de crédits

Nous observons que la population est majoritairement concentrée entre 20 et 60 ans. Cela signifie que la plupart des personnes se situent dans cette tranche d'âge. La distribution est unimodale, c'est-à-dire qu'il y a un pic unique de fréquence à 25 ans. Cela signifie que l'âge de 25 ans est l'âge le plus fréquent dans la population. Les valeurs aberrantes sont peu nombreuses. Cela signifie qu'il y a peu de personnes qui ont un âge très différent de l'âge moyen.

La ditribution de l'âge est asymétrique, cela suggère l'existence de valeurs aberrantes.

### 4.1.3 Analyse bivariée

Nous avons procédé à une analyse bivariée pour explorer et comprendre les liens entre deux variables dans notre ensemble de données. Elle permet d'identifier les corrélations, les associations et les tendances, fournissant ainsi des informations cruciales sur leurs interactions.

#### Etude de la corrélation entre la variable à expliquer et les variables explicatives

Dans cette partie de l'analyse, nous nous concentrons sur l'étude de la corrélation entre la variable à expliquer et les variables explicatives de notre ensemble de données. L'objectif principal est de comprendre les relations entre la variable cible que nous cherchons à prédire et les différentes caractéristiques qui pourraient influencer ses valeurs. En examinant ces relations, nous chercherons à identifier les variables qui présentent une forte corrélation avec notre variable cible, ce qui pourrait fournir des indications précieuses pour la modélisation et la prédiction.

- **Relation avec la propriété du logement** La figure 4.6 ci-dessous présente la répartition des défauts de prêt ( non-défaut : 0 /défaut : 1) selon les différentes catégories de propriété du logement :



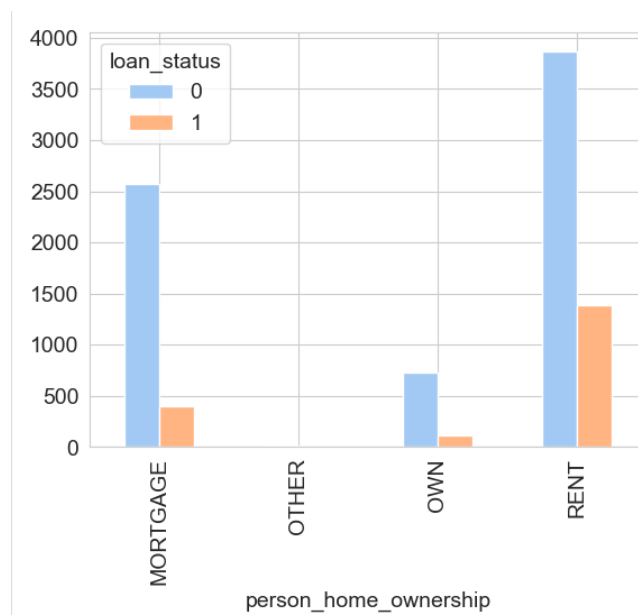


FIGURE 4.6 – Distribution de l'état du prêt selon le logement

Le pourcentage de défauts de prêt est le plus élevé parmi les locataires (RENT). Cela suggère que les locataires sont plus susceptibles de faire défaut sur leurs prêts que les propriétaires.

Le taux de défaut le plus faible est observé parmi les titulaires de prêts hypothécaires (MORTGAGE) et les propriétaires (OWN). Cela indique que les personnes ayant un prêt hypothécaire ou qui sont propriétaires de leur logement peuvent avoir une plus faible propension à faire défaut sur leurs prêts.

- **Relation avec l'âge** Les boîtes à moustaches présenté ci-dessous montre la distribution des âges des demandeurs de crédit en fonction de leur situation de défaut de prêt. La couleur bleue représente les demandeurs de crédit qui n'ont pas fait défaut sur leur prêt (défaut = 0), tandis que la couleur rouge représente les demandeurs de crédit qui ont fait défaut sur leur prêt (défaut = 1).

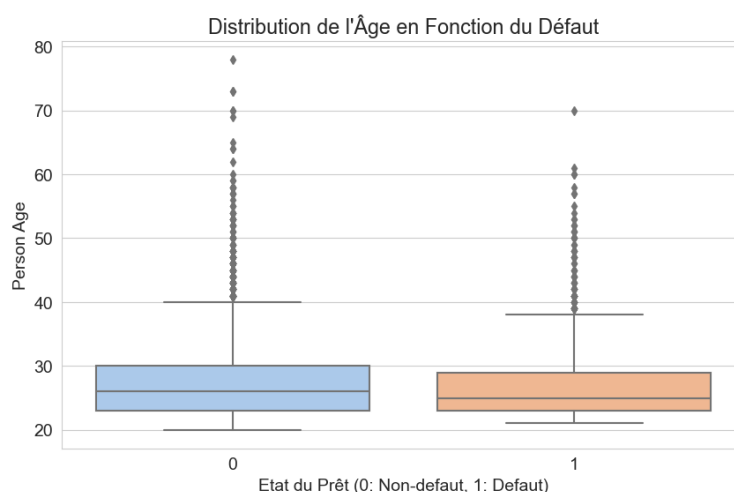


FIGURE 4.7 – Distribution de l'âge en fonction du défaut

Nous observons une concentration des demandeurs de crédit en défaut de prêt dans les tranches d'âge les plus jeunes (moins de 30 ans). Cela suggère que les jeunes adultes peuvent être plus susceptibles de faire défaut sur leurs prêts que les personnes plus âgées.

Pour éviter d'alourdir le rapport, les visualisations relatives aux autres variables sont présentées dans l'annexe C.

## 4.2 Nettoyage des données

Cette section englobe un ensemble de techniques visant à préparer, nettoyer et organiser les données brutes avant leur analyse. Dans cette section, nous explorerons les différentes étapes du prétraitement des données, notamment la gestion des données manquantes, la détection et la correction des valeurs aberrantes, la normalisation des variables et la transformation des caractéristiques. En accordant une attention particulière à ces étapes, nous nous assurerons que nos données sont prêtes pour la modélisation de la probabilité de défaut, contribuant ainsi à des résultats fiables et robustes.

### 4.2.1 Données Manquantes

Dans toute analyse de données, la présence de valeurs manquantes est un défi courant. Ces valeurs manquantes peuvent provenir de diverses sources telles que des erreurs de saisie, des non-réponses dans des enquêtes, ou encore des défauts dans le processus de collecte des données. La première partie de cette sous-section aborde la description des valeurs manquantes, en examinant leur répartition à travers les variables et en évaluant leur impact sur l'ensemble des données. Comprendre la nature et la fréquence des valeurs manquantes est crucial pour une analyse précise et fiable. La deuxième partie se concentre sur le traitement des valeurs manquantes, explorant les différentes méthodes pour remplacer ou estimer ces valeurs manquantes.

#### Description des valeurs manquantes

La présence de données manquantes est un défi fréquent dans l'analyse de données, pouvant affecter la qualité et la validité des résultats obtenus. Dans cette sous-section, nous aborderons la problématique des données manquantes dans notre ensemble de données. Dans la figure 4.8 ci-dessous nous observons les données manquantes présentes dans notre jeu de données :

```
person_age          0.000000
person_income       0.000000
person_home_ownership 0.000000
person_emp_length   4.320792
loan_grade          0.000000
loan_amnt          0.000000
loan_int_rate       9.811361
loan_status         0.000000
cb_person_default_on_file 0.000000
cb_person_cred_hist_length 0.000000
dtype: float64
```

FIGURE 4.8 – Pourcentages de Valeurs Manquantes

Le pourcentage des valeurs manquantes est important pour les deux variables "*person\_emp\_length*" et "*loan\_int\_rate*", représentant respectivement 4.32% et 9.81% des données.

## Traitement des valeurs manquantes

Plusieurs méthodes sont disponibles pour gérer les valeurs manquantes, chacune avec ses propres avantages et inconvénients. L'une des approches les plus simples consiste à imputer les valeurs manquantes par des statistiques descriptives telles que la moyenne, la médiane ou le mode, en fonction de la nature des données et du contexte de l'analyse.

Cette méthode permet de préserver la taille de l'échantillon et de maintenir la cohérence des analyses statistiques. Cependant, elle peut introduire un biais si les données ne suivent pas une distribution normale ou si des valeurs aberrantes sont présentes. Une autre approche consiste à utiliser des méthodes plus avancées telles que l'imputation par modèle, où un modèle est construit pour prédire les valeurs manquantes en fonction des autres variables de l'ensemble de données. Cette méthode offre une meilleure précision mais peut être plus complexe à mettre en œuvre.

Dans notre contexte, il est impossible de calculer les valeurs manquantes de la colonne "taux d'intérêt" avec la formule habituelle, car nous ne disposons pas de toutes les informations requises relatives aux crédits. Pour pallier ce problème, nous avons décidé d'adopter une approche de prédiction en utilisant des modèles de régression. Ces modèles seront entraînés à partir des données disponibles pour prédire les valeurs manquantes en se basant sur les relations existantes entre les caractéristiques des prêts et les taux d'intérêt observés. Cette démarche nous permettra de compléter les données manquantes de manière robuste et de garantir la cohérence de nos analyses ultérieures. L'annexe E.1 contient tous les détails relatifs à l'aspect mathématique de ces modèles.

### 1. Imputation des années de travail des demandeurs de crédits

Dans cette partie, nous abordons l'estimation du nombre d'années de travail des demandeurs de crédits. Cette variable, souvent désignée par "*person\_emp\_length*", est cruciale dans l'évaluation du risque de crédit, car elle renseigne sur la stabilité professionnelle de l'emprunteur.

Pour remplir les valeurs manquantes de cette variable, nous avons opté pour une imputation par la médiane. Cette décision découle de plusieurs considérations. Tout d'abord, la médiane est une mesure de tendance centrale robuste aux valeurs extrêmes, ce qui signifie qu'elle n'est pas influencée par les outliers ou les valeurs aberrantes. Cela est particulièrement important dans le contexte de l'estimation du nombre d'années de travail, car la distribution de cette variable peut présenter une forte asymétrie ou contenir des valeurs très élevées qui ne reflètent pas la situation typique des emprunteurs.

De plus, l'utilisation de la médiane pour l'imputation permet de maintenir la distribution générale des données sans biaiser les résultats vers des valeurs trop hautes ou trop basses. Contrairement à la moyenne, la médiane n'est pas affectée par les valeurs extrêmes, ce qui en fait une méthode plus fiable lorsque les données sont dispersées ou présentent des variations importantes.

## 2. Estimation du taux d'intérêt

Le tableau représenté par la figure 4.9 ci-dessous, présente le score d'entraînement, le score de test et l'erreur quadratique moyenne (MSE) pour cinq modèles d'apprentissage automatique différents : Ridge bayésien, Ridge, Régression linéaire, Lasso et Random Forest. Le score d'entraînement représente la performance du modèle sur les données d'entraînement, tandis que le score de test représente sa performance sur les données de test invisibles. L'MSE est une mesure de la différence quadratique moyenne entre les valeurs prédites et les valeurs réelles.

	Model	Train Score	Test Score	Mean Squared Error
0	Linear Regression	0.874753	0.879606	1.247723
1	Ridge	0.874752	0.879602	1.247765
2	Bayesian Ridge	0.874752	0.879598	1.247796
3	Lasso	0.791264	0.792693	2.148455
4	Random Forest	0.986287	0.912745	0.904281
5	Gradient Boosting	0.909255	0.906708	0.966846

FIGURE 4.9 – Tableau récapitulatif de la performance des modèles de régression pour la prédiction des valeurs manquantes

- Ridge bayésien et Ridge : Ces deux modèles ont des scores d'entraînement et de test très similaires, ce qui suggère qu'ils fonctionnent tous les deux bien sur les données. Cependant, Ridge a une MSE légèrement inférieure, indiquant qu'il pourrait être un choix légèrement meilleur pour cet ensemble de données particulier.
- Régression linéaire : La régression linéaire a des scores d'entraînement et de test légèrement inférieurs à ceux du Ridge bayésien et du Ridge, mais son MSE est encore relativement faible, il est égale à 1.24. Cela suggère que la régression linéaire est une option viable pour cette tâche, mais elle n'est peut-être pas aussi précise que les deux autres modèles.
- Lasso : Le Lasso a des scores d'entraînement nettement inférieurs aux autres modèles, ce qui indique qu'il pourrait être en surapprentissage (overfitting) sur les données d'entraînement. Cela se reflète dans son score de test et son MSE plus élevés (2.15), ce qui suggère qu'il pourrait ne pas se généraliser bien aux données invisibles.
- Random Forest : Random Forest a les scores d'entraînement et de test les plus élevés de tous les modèles, et son MSE est également relativement faible (égale à 0.90). Cela suggère que Random Forest est le modèle le plus performant pour cet ensemble de données. La figure 4.10 ci-dessous nous montre l'évaluation de la performance du modèle Random Forest :

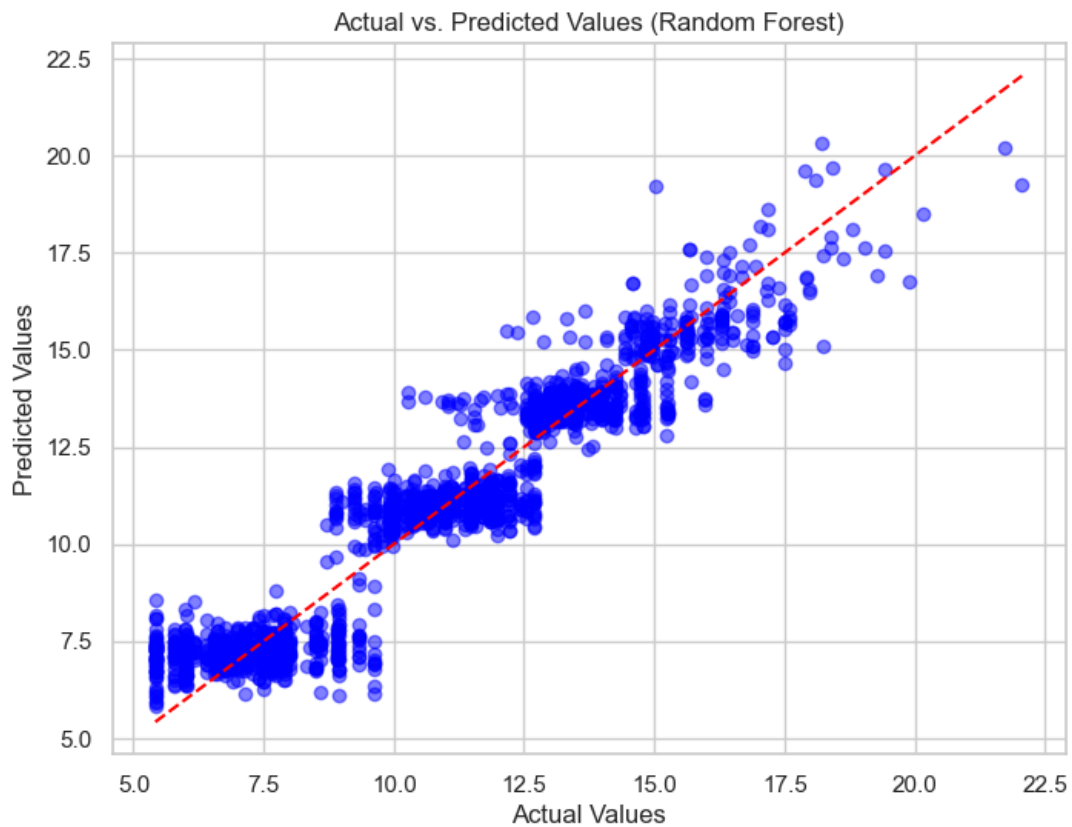


FIGURE 4.10 – Performance du modèle Random Forest dans la prédiction du taux d'intérêt du prêt

Plus les points bleus sont proches de la ligne de régression rouge, plus le modèle Random Forest est précis dans ses prédictions. En d'autres termes, le modèle est capable d'apprendre la relation entre les caractéristiques des emprunteurs (représentées par les variables dans  $X$ ) et les taux d'intérêt des prêts.

En conclusion, nous avons évalué cinq modèles d'apprentissage automatique différents pour prédire les valeurs manquantes du taux d'intérêt des prêts. Nous avons constaté que le modèle *Random Forest* a surpassé les autres en termes de performances, avec des scores d'entraînement et de test élevés ainsi qu'une erreur quadratique moyenne (MSE) relativement faible. **Par conséquent**, nous avons décidé d'adopter le *RandomForestRegressor* pour la prédiction des valeurs manquantes du taux d'intérêt.

### 4.2.2 Valeurs aberrantes

Dans cette sous-section, nous explorons la présence éventuelle de points de données atypiques dans notre ensemble de données sur les demandeurs de crédits. Les valeurs aberrantes, également appelées valeurs extrêmes, sont des observations qui se distinguent considérablement du reste de l'ensemble de données. Leur présence peut résulter de divers facteurs, tels que des erreurs de saisie, des mesures inexactes ou des événements rares mais légitimes. Identifiées et traitées de manière appropriée, les valeurs aberrantes peuvent offrir des informations précieuses sur les caractéristiques inhabituelles ou anormales des observations dans notre ensemble de données.

## Détection des valeurs aberrantes

Nous avons utilisé une méthode basée sur les Z-scores pour détecter les valeurs aberrantes dans notre ensemble de données. Cette approche calcule la moyenne et l'écart-type de chaque variable numérique, puis calcule les Z-scores pour chaque observation en fonction de ces statistiques descriptives. Les observations dont les Z-scores dépassent un seuil spécifié sont considérées comme des valeurs aberrantes [Atkinson and Mulira, 1993].

Le z-score, également connu sous le nom de score standardisé, mesure le nombre d'écarts-types par lesquels une observation est éloignée de la moyenne de l'ensemble de données. Mathématiquement, le z-score d'une observation  $x$  est calculé en soustrayant la moyenne de l'ensemble de données  $\mu$  de  $x$  et en divisant le résultat par l'écart-type  $\sigma$ , comme suit :

$$z = \frac{x - \mu}{\sigma}$$

Le z-score permet de standardiser les données et de les ramener à une échelle commune, ce qui facilite la comparaison des observations dans différents contextes. Plus précisément, nous avons calculé les moyennes et les écart-types des variables "*person\_income*", "*person\_age*", et "*person\_emp\_length*" puisque leurs distributions, comme nous avons vu dans la partie analyse univariée, sont asymétriques. Ensuite, nous avons calculé les Z-scores pour chaque observation de ces variables. Les observations dont les Z-scores étaient supérieurs à 3 selon [Atkinson and Mulira, 1993] ont été identifiées comme des valeurs aberrantes.

Après avoir identifié les valeurs aberrantes, nous les avons supprimées de l'ensemble de données. Cela a été accompli en collectant les index des observations aberrantes détectées pour chaque variable, puis en les supprimant de l'ensemble de données principal. La base de données obtenue est composée de 9380 demandes de crédit.

Cette approche nous a permis de nettoyer notre ensemble de données des valeurs aberrantes, ce qui est crucial pour assurer la qualité et la fiabilité de nos analyses ultérieures.

## 4.3 Transformation des variables

Cette étape cruciale permet de préparer les caractéristiques pour une analyse statistique ou une modélisation prédictive précise de la probabilité de défaut. En effet, elle contribue à améliorer la qualité des données en réduisant les imperfections telles que les incohérences et les écarts entre les échelles des variables.

De plus, la transformation des données facilite l'exploration des relations entre les variables, la détection de schémas et la compréhension des caractéristiques sous-jacentes des données. Elle joue également un rôle essentiel dans l'amélioration de la performance des modèles statistiques et d'apprentissage automatique, en rendant les données conformes aux hypothèses spécifiques de ces modèles. En réduisant le biais introduit par des valeurs extrêmes ou des différences d'échelle entre les variables, la transformation des données contribue à des analyses plus fiables et des prédictions plus précises.

### 4.3.1 Conversion des catégories en valeurs numériques

Dans notre ensemble de données, la variable "loan\_grade" représente le grade ou la catégorie attribué à chaque prêt. Chaque grade, symbolisé par une lettre (de A à G), reflète le niveau de risque associé au prêt. Afin de rendre ces grades exploitables dans les analyses statistiques, nous avons entrepris une conversion en attribuant un entier à chaque grade, où A est associé à 6, et progressivement jusqu'à ce que G soit attribué à 0.

Par ailleurs, nous avons également traité la variable catégorielle "person\_home\_ownership", qui présente quatre catégories distinctes : OWN, RENT, MORTGAGE et OTHER. Pour rendre cette variable compatible avec nos analyses, nous avons utilisé la fonction *get\_dummies* de la bibliothèque Pandas. Cette fonction a permis de créer des variables indicatrices pour chaque catégorie. La figure 4.11 ci-dessous nous montre les colonnes obtenues :

	cb_person_cred_hist_length	person_home_ownership_MORTGAGE	\
0	2	0	
1	2	0	
2	3	0	
3	2	0	
4	2	1	
...	...	...	
9407	12	0	
9408	15	1	
9409	15	1	
9410	12	1	
9411	13	1	
	person_home_ownership_OTHER	person_home_ownership_OWN	\
0	0	1	
1	0	1	
2	0	1	
3	0	1	
4	0	0	
...	...	...	
9407	0	1	
9408	0	0	
9409	0	0	
9410	0	0	
9411	0	0	
	person_home_ownership_RENT		
0	0		
1	0		
2	0		
3	0		
4	0		
...	...		
9407	0		
9408	0		
9409	0		
9410	0		
9411	0		

[9380 rows x 13 columns]

FIGURE 4.11 – Colonnes obtenues après transformation

Cette approche de transformation des variables catégorielles permet d'intégrer efficacement ces informations dans nos modèles d'analyse et de prédiction, facilitant ainsi leur interprétation et leur utilisation dans le cadre de notre analyse de données.

### 4.3.2 Division des données

Dans cette section, nous décrivons le processus de division des données utilisé pour notre analyse. Nous avons commencé par identifier notre variable cible, *loan\_status*, qui indique si un prêt est en défaut ou non-défaut. Les autres variables, appelées variables explicatives, sont utilisées pour prédire cette variable cible.

La variable cible ( $y$ ) est donc *loan\_status* (défaut ou non-défaut), tandis que les variables explicatives ( $X$ ) incluent toutes les autres variables disponibles dans notre jeu de données. Ces variables explicatives sont susceptibles d'influencer ou de prédire le statut du prêt.

Une étape cruciale dans l'analyse des données et le développement de modèles prédictifs est la division des données en ensembles d'entraînement et de test. Cette division permet de vérifier la performance et la robustesse des modèles sur des données inédites.

#### Importance de la division des données

Diviser les données est essentiel pour plusieurs raisons :

- cela permet une **évaluation de la performance** des modèles. En réservant un ensemble de test, nous pouvons évaluer la performance des modèles sur des données qu'ils n'ont jamais vues. Cela permet de simuler la manière dont les modèles se comporteront sur des données réelles.
- cela aide à **prévenir le surapprentissage** (ou *overfitting*). Utiliser les mêmes données pour l'entraînement et le test peut conduire à un surapprentissage, où le modèle s'adapte trop bien aux données d'entraînement et échoue à généraliser sur de nouvelles données. La division des données en ensembles distincts permet de détecter et de minimiser ce problème.
- la division des données permet l'utilisation de techniques de **validation croisée** pour une évaluation plus robuste des modèles. Cela améliore la fiabilité des estimations de performance, en nous permettant de mieux comprendre comment le modèle se comportera en pratique.

#### Processus de division

Nous avons utilisé une division typique en deux ensembles : 80% des données pour l'ensemble d'entraînement et 20% des données pour l'ensemble de test. Cette proportion est couramment utilisée pour assurer un équilibre entre suffisamment de données pour l'entraînement et une taille de test adéquate pour une évaluation fiable.

La division a été effectuée de manière aléatoire pour garantir que les ensembles d'entraînement et de test soient représentatifs de la distribution originale des données. Cette randomisation aide à s'assurer que les modèles ne seront pas biaisés par une quelconque structure particulière des données



d'entraînement ou de test.

### 4.3.3 Normalisation des variables numériques

Avant de procéder à la modélisation, il est souvent nécessaire de standardiser les variables numériques dans notre ensemble de données. La standardisation des variables numériques consiste à centrer les données autour de zéro et à les mettre à l'échelle pour avoir une variance unitaire. Cette méthode permet de rendre les variables comparables et de réduire les biais introduits par les différences d'échelle entre les variables [Hubert and Elisha, 2023]. Voici les étapes typiques de la méthode de standardisation :

- **Centrage des données** : Pour centrer les données, la moyenne de chaque variable est soustraite à toutes les observations. Cela signifie que la moyenne de chaque variable devient zéro, ce qui aligne les données autour de l'origine [Hubert and Elisha, 2023].
- **Mise à l'échelle des données** : Après le centrage, les données sont mises à l'échelle en divisant chaque observation par l'écart type de la variable correspondante. Cette étape garantit que toutes les variables ont une variance unitaire, ce qui facilite la comparaison entre elles [Hubert and Elisha, 2023].

La formule pour standardiser une variable  $X$  est la suivante :

$$Z = \frac{X - \text{moyenne}(X)}{\text{écart-type}(X)}$$

où  $Z$  représente la variable standardisée, et  $\text{moyenne}(X)$  et  $\text{écart-type}(X)$  représentent respectivement la moyenne et l'écart type de la variable  $X$ .

La normalisation des variables numériques est particulièrement importante dans les cas où les variables ont des échelles différentes ou lorsque les valeurs numériques sont sujettes à des écarts importants. Cette méthode garantit que toutes les variables contribuent également à l'analyse ou à la modélisation, évitant ainsi que certaines variables ne dominent le processus en raison de leur échelle ou de leur amplitude [Hubert and Elisha, 2023].

## 4.4 Rééchantillonnage des données

Le rééchantillonnage est une technique couramment utilisée dans le domaine de l'apprentissage automatique pour traiter les ensembles de données déséquilibrés, où une classe est beaucoup plus représentée que l'autre. Dans votre cas, avec seulement 19.08% des données appartenant à la classe 1 (défaut), il est crucial de corriger ce déséquilibre afin d'éviter que le modèle ne soit biaisé vers la classe majoritaire.

La méthode utilisée ici, appelée Random Over Sampling, est une technique de suréchantillonnage qui consiste à augmenter le nombre d'instances de la classe minoritaire en créant de nouveaux exemples synthétiques basés sur les observations existantes de cette classe. Nous présentons ci-dessous le fonctionnement de cette méthode [Vale et al., 2022] :

Tout d’abord, nous initialisons la classe `RandomOverSampler` en spécifiant un état aléatoire (`random_state=42`, pour assurer la reproductibilité des résultats).

Ensuite, nous appliquons le suréchantillonnage aux données d’entraînement ( $X_{train}$  et  $y_{train}$ ). Cette étape consiste à générer de nouveaux exemples de la classe minoritaire (classe 1) afin d’équilibrer le nombre d’instances entre les deux classes.

L’intérêt principal du rééchantillonnage avec ROS est de permettre au modèle d’apprendre de manière plus équilibrée des deux classes, ce qui peut améliorer sa capacité à détecter les modèles dans la classe minoritaire et donc sa performance globale [Vale et al., 2022].

Compte tenu de la sensibilité des données avec lesquelles nous travaillons, nous avons choisi le suréchantillonnage aléatoire (ROS) comme méthode de rééchantillonnage parce qu’il offre plusieurs avantages, notamment sa simplicité de mise en œuvre et son efficacité dans l’équilibrage des ensembles de données déséquilibrés sans nécessiter de modifications majeures des données d’origine. Étant donné que le ROS crée de nouveaux exemples synthétiques en utilisant les données existantes de la classe minoritaire, il ne nécessite pas l’ajout de nouvelles données potentiellement sensibles. Cela réduit les risques de compromission de la confidentialité des données tout en permettant au modèle d’apprendre de manière plus équilibrée des deux classes [Vale et al., 2022].

## 4.5 Implémentation des modèles de machine learning

D’après la revue de la littérature, plusieurs modèles sont souvent exploités pour l’estimation de la probabilité de défaut dans le contexte des crédits à la consommation. Parmi ces modèles, nous retrouvons les K plus proches voisins (*K-Nearest-Neighbors KNN*), la régression logistique, les arbres de décision, les forêts aléatoires, les machines à vecteurs supports (SVM) et les Réseaux Neuronaux (NN). Chaque modèle présente ses propres avantages et inconvénients, et leur performance peut varier en fonction des caractéristiques spécifiques des données et des objectifs de modélisation.

Cependant, il est important de noter que d’après plusieurs études, le modèle de régression logistique est souvent considéré comme plus performant que les Réseaux Neuronaux dans le domaine de l’estimation de la probabilité de défaut pour les crédits à la consommation. Ceci peut être attribué à la simplicité et à l’interprétabilité de la régression logistique, ainsi qu’à sa capacité à bien gérer des jeux de données de taille modérée à grande. Par ailleurs, les forêts aléatoires se sont également révélées être des modèles efficaces, souvent surpassant les arbres de décision traditionnels en termes de performance prédictive. Leur capacité à réduire le surajustement et à gérer les interactions complexes entre les variables en fait un choix attrayant pour de nombreux problèmes de modélisation.

Dans le cadre de notre étude, nous prendrons en compte ces observations de la littérature pour sélectionner les modèles appropriés à notre tâche d’estimation de la probabilité de défaut pour les crédits à la consommation qui sont : **KNN, Régression Logistique, Forêts Aléatoires et Machine à Vecteurs Supports**. Nous élaborerons sur les techniques de validation croisée et d’évaluation des modèles pour garantir des résultats fiables et une généralisation optimale des performances des modèles sélectionnés. De plus, **GridSearchCV** a été employé pour chacun des modèles afin de choisir les meilleurs paramètres pour chaque modèle.

### 4.5.1 KNN

Le K-Nearest Neighbors (KNN), comme décrit dans le chapitre 2, est une méthode de classification non paramétrique qui utilise la proximité des exemples de formation pour prédire la classe d'un nouveau point. Elle est basée sur l'hypothèse que des exemples similaires existent à proximité dans l'espace des caractéristiques.

#### a. Matrice de confusion

En appliquant le modèle KNN sur la base d'apprentissage, nous avons obtenu les résultats suivants. Nous avons commencé par visualiser la matrice de confusion dans la figure 4.12 ci-dessous :

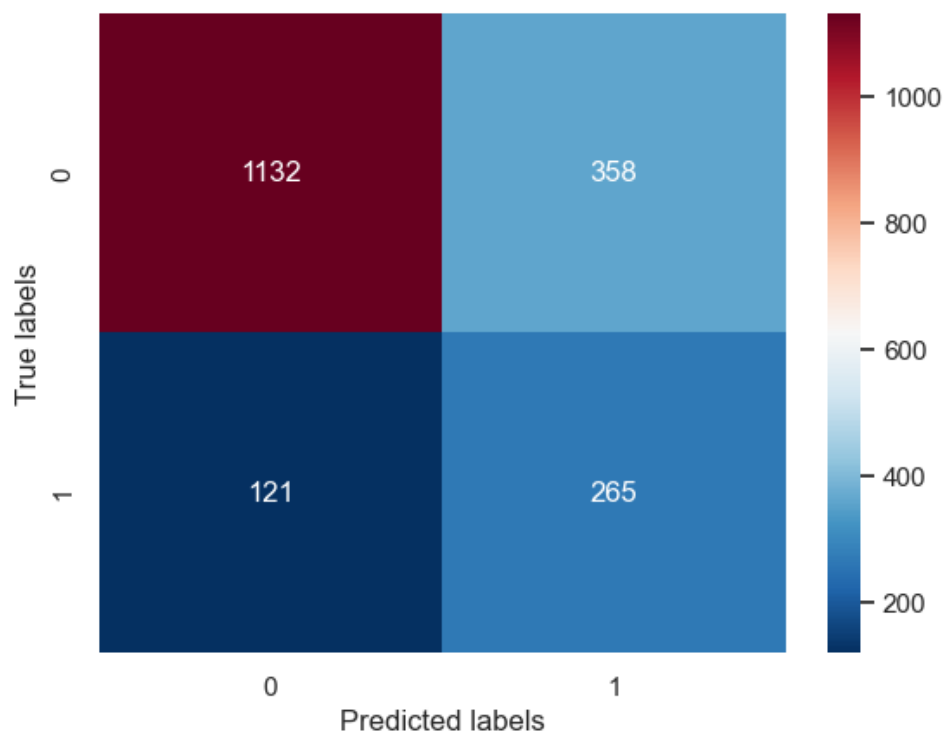


FIGURE 4.12 – Matrice de confusion du modèle KNN

La matrice de confusion montre les résultats du modèle KNN pour deux classes (0 et 1) avec les valeurs suivantes :

Pour la classe 1 le modèle a détecté correctement 265 valeurs et pour la classe 0 1132 valeurs alors qu'il a mal classé 358 valeurs dans la classe 1 et 121 valeurs dans la classe 0.

Cette répartition indique que le modèle est plus capable à prédire la classe 0 (non défaut) que la classe 1 (défaut), comme en témoigne le nombre important de vrais négatifs comparé aux vrais positifs.

#### b. Rapport de classification :

il fournit des mesures détaillées qui évaluent la performance du modèle sur chaque classe :

Classe	Précision	Rappel	Score F1	Support
0	0.90	0.76	0.83	1490
1	0.43	0.69	0.53	386
<b>Précision Globale</b>			0.74	1876
<b>Moyenne Macro</b>	0.66	0.72	0.68	1876
<b>Moyenne Pondérée</b>	0.81	0.74	0.76	1876

TABLE 4.1 – Rapport de Classification du modèle KNN

## Evaluation du rapport de classification

### Précision :

- Classe 0 : 0.90, indiquant que 90% des prédictions de la classe 0 étaient correctes.
- Classe 1 : 0.43, suggérant que seulement 43% des prédictions de la classe 1 étaient correctes.

### Rappel (sensibilité ou taux de vrais positifs) :

- Classe 0 : 0.76, signifiant que 76% des cas réels de la classe 0 ont été correctement prédits par le modèle.
- Classe 1 : 0.69, ce qui signifie que le modèle a identifié 69% des cas réels de la classe 1.

### Score F1 (moyenne harmonique de la précision et du rappel) :

- Classe 0 : 0.83, indiquant un bon équilibre entre la précision et le rappel pour la classe 0.
- Classe 1 : 0.53, reflétant un déséquilibre entre la précision et le rappel pour la classe 1, avec une inclinaison vers un rappel relativement plus élevé.

### Analyse Globale

- **Précision Globale du modèle(Accuracy)** est de 0.74, signifiant que 74% des prédictions totales étaient correctes.

### c. Courbe ROC

Pour mieux évaluer notre modèle, nous avons visualisé la courbe ROC dans la figure 4.13 ci-dessous :

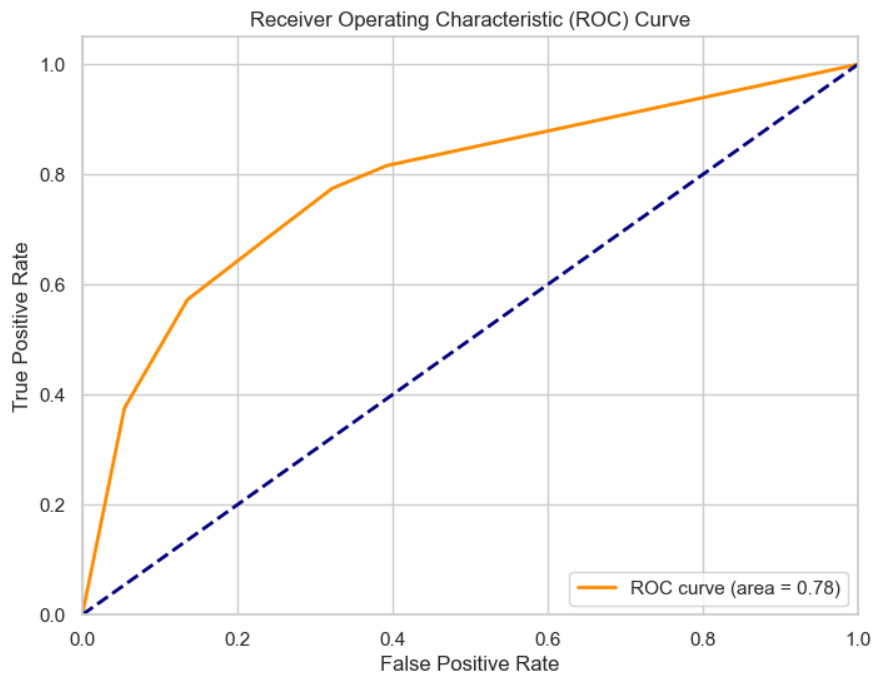


FIGURE 4.13 – Courbe ROC du modèle KNN

La courbe ROC (Receiver Operating Characteristic) illustre la performance du modèle KNN pour la classification des crédits en fonction de différents seuils de décision. Cette courbe compare le taux de vrais positifs (sensibilité) au taux de faux positifs (1-spécificité) à différents seuils de classification.

### Description de la Courbe ROC

**Aire sous la courbe (AUC) :** L'AUC est de 0.78, ce qui indique une bonne capacité de discrimination du modèle entre les classes positives et négatives. Une AUC de 1.0 représenterait une performance parfaite, tandis qu'une AUC de 0.5 indiquerait une performance équivalente à un tirage aléatoire.

**Taux de Vrais Positifs (sensibilité) :** Sur l'axe des ordonnées, le taux de vrais positifs mesure la proportion de positifs réels que le modèle a correctement identifiés. Plus ce taux est élevé pour un faible taux de faux positifs, mieux c'est.

**Taux de Faux Positifs (1 - spécificité) :** Sur l'axe des abscisses, le taux de faux positifs mesure la proportion de négatifs réels que le modèle a incorrectement classé comme positifs. Un modèle idéal viserait à minimiser ce taux.

### Analyse de la Performance

La courbe ROC montre que le modèle KNN parvient à atteindre un équilibre relativement bon entre sensibilité et spécificité, particulièrement dans la région médiane de la courbe. Cependant, pour atteindre de très hauts taux de vrais positifs, le modèle augmente substantiellement les faux positifs, ce qui peut être problématique selon le coût associé aux erreurs de classification dans le contexte d'application. La courbe s'élève rapidement à un taux de vrais positifs d'environ 0.8 avant de commencer à plafonner, suggérant que des améliorations peuvent être envisagées pour augmenter la sensibilité sans sacrifier excessivement la spécificité.

## Conclusion

Le modèle KNN avec une AUC de 0.78 montre une performance correcte, mais il reste une marge d'amélioration. L'analyse de la courbe ROC indique où le modèle pourrait bénéficier de réglages dans le choix des seuils ou de l'ajustement des paramètres du modèle pour optimiser le compromis entre la capture des vrais positifs et la minimisation des faux positifs. Des expérimentations supplémentaires avec des ajustements de paramètres ou des techniques de prétraitement des données pourraient potentiellement améliorer la robustesse du modèle KNN dans ce cadre d'application.

### 4.5.2 Régression logistique

La régression logistique est une technique statistique couramment utilisée pour les problèmes de classification binaire. Elle modélise la probabilité qu'une observation appartienne à une classe donnée en utilisant une fonction logistique. En d'autres termes, elle permet d'estimer la probabilité qu'un crédit soit en défaut ou non défaut en fonction de différentes variables explicatives. Dans cette étude, nous avons appliqué la régression logistique pour prédire la classe des crédits, en exploitant ses capacités à gérer les relations linéaires entre les variables indépendantes et la variable dépendante. Ce modèle est particulièrement apprécié pour sa simplicité d'interprétation et sa robustesse dans des situations où les classes sont équilibrées. Afin de comprendre la performance de notre modèle de régression logistique, nous avons examiné la matrice de confusion et le rapport de classification, qui fournissent des insights précieux sur sa capacité à différencier les crédits en défaut et non défaut.

#### a. Matrice de confusion

En appliquant le modèle de régression logistique à notre base de données, nous avons obtenu la matrice de confusion, illustrée dans la Figure 4.14 ci-dessous :

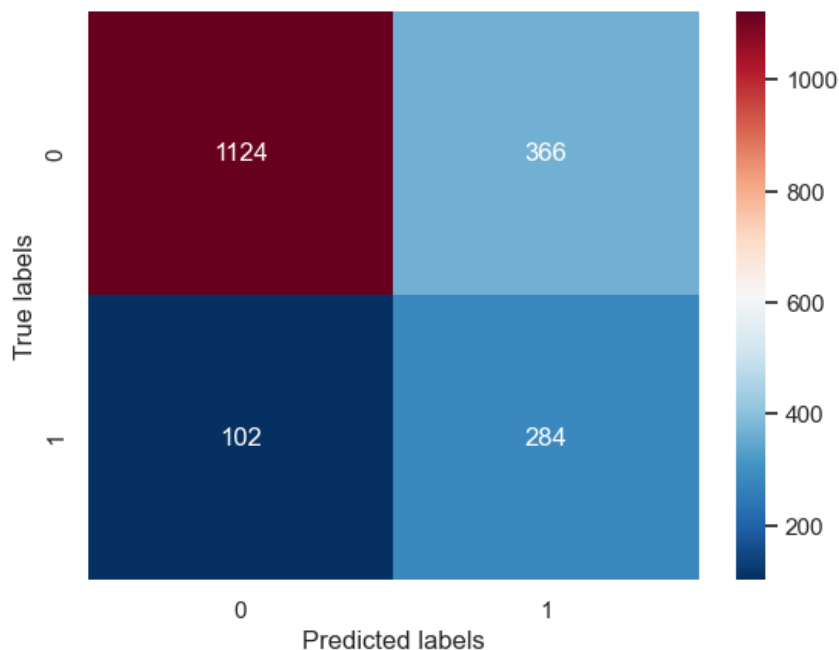


FIGURE 4.14 – Matrice de confusion du modèle RL

La matrice de confusion et le rapport de classification pour le modèle de régression logistique fournissent des informations cruciales sur la performance de ce modèle dans la classification des crédits.

Pour la classe 1, le modèle a correctement identifié 284 demandes de crédits et pour la classe 0 1124 valeurs alors qu'il a mal classé 366 valeurs dans la classe 1 et 102 valeurs dans la classe 0. Cette répartition indique que la régression logistique est plus capable à prédire la classe 0 (non défaut) que la classe 1 (défaut), comme en témoigne le nombre important de vrais négatifs comparé aux vrais positifs.

## b. Rapport de Classification

Pour mieux évaluer le modèle obtenu, nous avons visualisé le rapport de classification décrit dans la table 4.2 ci-dessous :

Class	Precision	Recall	F1-Score	Support
0	0.92	0.75	0.83	1490
1	0.44	0.74	0.55	386
<b>Accuracy</b>			0.75	1876
<b>Macro Avg</b>	0.68	0.75	0.69	1876
<b>Weighted Avg</b>	0.82	0.75	0.77	1876

TABLE 4.2 – Rapport de classification du modèle régression logistique

## Évaluation du Rapport de Classification

### Précision :

- **Classe 0** : 0.92, indiquant une très haute précision pour les prédictions de non-défaut.
- **Classe 1** : 0.44, ce qui est relativement faible et indique que moins de la moitié des prédictions de défaut étaient correctes.

### Rappel (taux de vrais positifs) :

- **Classe 0** : 0.75, signifiant que 75% des cas réels de non-défaut ont été correctement identifiés.
- **Classe 1** : 0.74, ce qui est favorable pour un modèle visant à identifier les risques de défaut.

### Score F1 (de la précision et du rappel) :

- **Classe 0** : 0.83, un score élevé qui montre un bon équilibre entre précision et rappel pour la classe 0.
- **Classe 1** : 0.55, un score plus faible indiquant que le modèle pourrait être amélioré pour équilibrer la précision et le rappel pour la classe de défaut.

## Conclusion

Le modèle de régression logistique montre une performance compétente pour identifier les non-défauts avec une précision élevée. Cependant, le modèle a une marge d'amélioration significative pour la classification des défauts, en particulier en augmentant la précision pour la classe 1.

### c. Courbe ROC

Pour mieux évaluer notre modèle, nous avons visualisé la courbe ROC, dans la figure 4.15 ci-dessous, qui illustre sa capacité à distinguer entre les classes positives (défauts) et négatives (non-défauts) à différents seuils de décision.

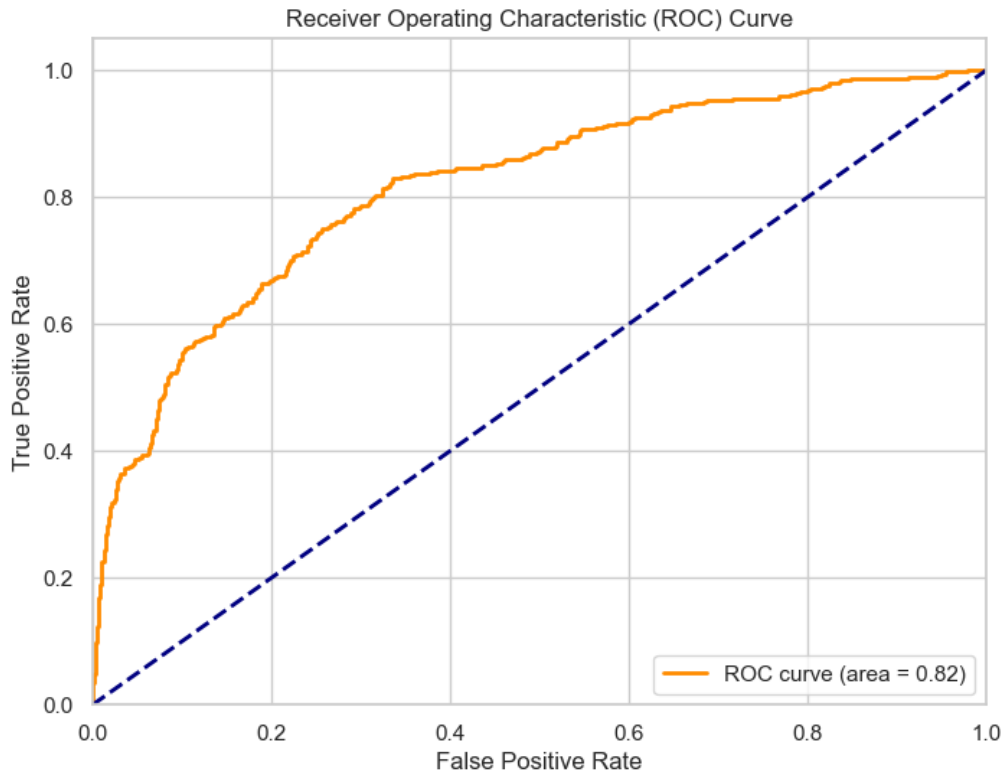


FIGURE 4.15 – Courbe ROC du modèle RL

La courbe ROC (Receiver Operating Characteristic) pour le modèle de régression logistique illustre comment ce modèle parvient à distinguer entre les classes positives (défauts) et négatives (non-défauts) à différents seuils de décision.

### Description de la Courbe ROC

**Aire sous la courbe (AUC) :** L'AUC est de 0.82, ce qui indique une très bonne capacité de discrimination du modèle entre les classes positives et négatives. Une AUC plus proche de 1.0 signifie que le modèle a une excellente performance en termes de séparation des deux classes. Une AUC de 0.5 représenterait une performance équivalente à un tirage aléatoire.

### Analyse de la Performance

**Taux de Vrais Positifs (sensibilité) :** Le taux de vrais positifs, également connu sous le nom de sensibilité, mesure la proportion d'observations positives réelles que le modèle est capable de capturer. Dans ce cas, la courbe montre que le modèle augmente le taux de vrais positifs de manière progressive tout en augmentant le taux de faux positifs, ce qui est typique pour les modèles de classification.



**Taux de Faux Positifs (1 - spécificité) :** Sur l'axe des abscisses, la courbe montre le taux de faux positifs. Idéalement, nous souhaitons que cette valeur reste aussi basse que possible pour chaque taux de vrais positifs, ce qui indiquerait une meilleure spécificité du modèle.

### Évaluation de la Courbe

La courbe s'élève rapidement au début, indiquant que le modèle atteint rapidement un taux de sensibilité élevé avec un taux relativement bas de faux positifs. Cela suggère que le modèle est capable de bien classer les vrais positifs sans mal classer un trop grand nombre de vrais négatifs.

À mesure que la courbe se déplace vers le haut à droite, nous observons une augmentation progressive des faux positifs, ce qui est commun pour les modèles cherchant à maximiser le taux de vrais positifs. Toutefois, le modèle maintient une bonne performance générale comme indiqué par l'AUC de 0.82.

### Conclusion

Le modèle de régression logistique montre une bonne capacité à discriminer entre les classes de défaut et de non-défaut, avec une AUC de 0.82. Cette performance suggère que le modèle peut être très utile dans des scénarios où il est crucial de bien identifier les défauts sans trop augmenter le nombre de faux positifs, comme dans le contexte financier pour la détection de défauts de crédit. Une telle courbe ROC confirme que le modèle de régression logistique est un outil robuste pour la prédiction de risques dans ce cas d'application.

## 4.5.3 Forêts aléatoires

Le *Random Forest* fonctionne en construisant une multitude d'arbres de décision lors de l'entraînement et en sortant la classe qui est le mode des classes des arbres individuels. Dans le cadre de notre étude, nous avons utilisé un modèle de forêt aléatoire pour prédire la probabilité qu'un crédit soit en défaut ou non.

L'approche est particulièrement bien adaptée pour traiter les problèmes de classification binaire tels que celui-ci. En appliquant ce modèle à notre ensemble de données, nous avons pu évaluer sa performance à travers différentes métriques, notamment la matrice de confusion et le rapport de classification, afin de mieux comprendre sa capacité à distinguer entre les crédits en défaut et non défaut.

### a. Matrice de Confusion

En appliquant le modèle RF à notre ensemble de données, nous avons d'abord visualisé la matrice de confusion, comme illustré dans la Figure 4.17 ci-dessous :

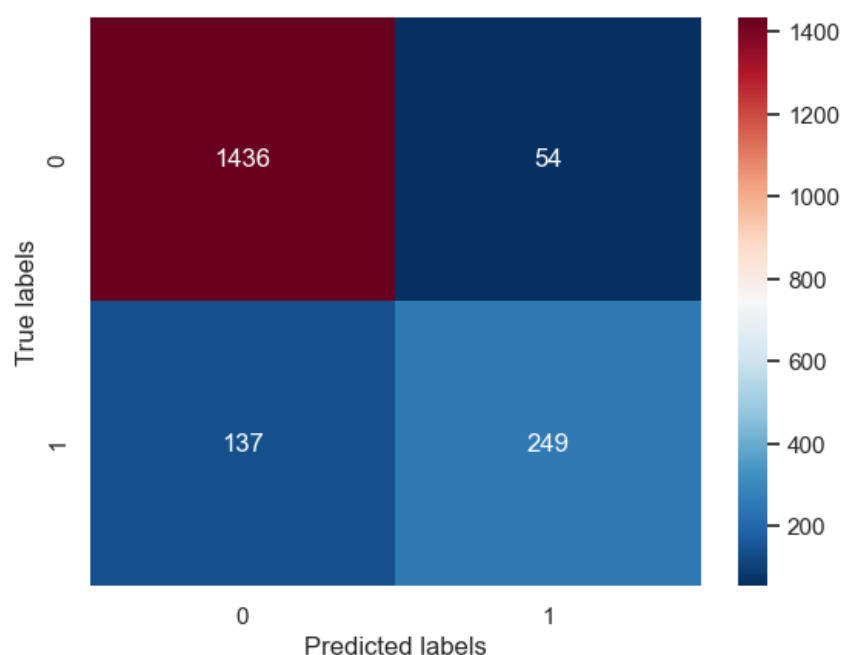


FIGURE 4.16 – Matrice de confusion du modèle Forêt Aléatoire

Pour la classe 1, le modèle a correctement identifié 249 demandes de crédits et pour la classe 0 1436 valeurs alors qu'il a mal classé 54 valeurs dans la classe 1 et 137 valeurs dans la classe 0. Cette répartition indique que le forêt aléatoire est plus capable à prédire la classe 0 (non défaut) que la classe 1 (défaut), comme en témoigne le nombre important de vrais négatifs comparé aux vrais positifs.

## b. Rapport de classification

Pour mieux évaluer le modèle obtenu, nous avons visualisé le rapport de classification décrit dans la table 4.3 ci-dessous :

Classe	Précision	Rappel	Score F1	Support
0	0.91	0.96	0.94	1490
1	0.82	0.65	0.72	386
<b>Précision Globale</b>			0.90	1876
<b>Moyenne Macro</b>	0.87	0.80	0.83	1876
<b>Moyenne Pondérée</b>	0.89	0.90	0.89	1876

TABLE 4.3 – Rapport de Classification du modèle Forêts aléatoires

## Evaluation du Rapport de Classification

### Classe 0 (Non-Défaut)

- **Précision** : 0.91 - Le modèle a correctement prédit 91% des non-défauts par rapport à toutes les prédictions de non-défauts.
- **Rappel** : 0.96 - Le modèle a correctement identifié 96% des réels non-défauts.
- **Score F1** : 0.94 - Le score F1 est très élevé, indiquant un excellent équilibre entre la précision et le rappel pour les non-défauts.

### Classe 1 (Défaut)

- **Précision** : 0.82 - Le modèle a une précision de 82% pour prédire correctement les défauts.
- **Rappel** : 0.65 - Le modèle a capturé 65% des réels défauts, ce qui indique une performance modérée en termes de sensibilité à détecter les cas de défaut.
- **Score F1** : 0.72 - Le score F1 est relativement bas comparé à celui de la classe 0, reflétant un compromis entre précision et rappel moins optimal pour les cas de défaut.

### Évaluation Globale

Précision Globale (Accuracy) : 0.90 - Globalement, le modèle prédit correctement 90% des instances.

c. La courbe ROC pour le modèle de forêts aléatoires est présentée dans la figure 4.17 ci-dessous.

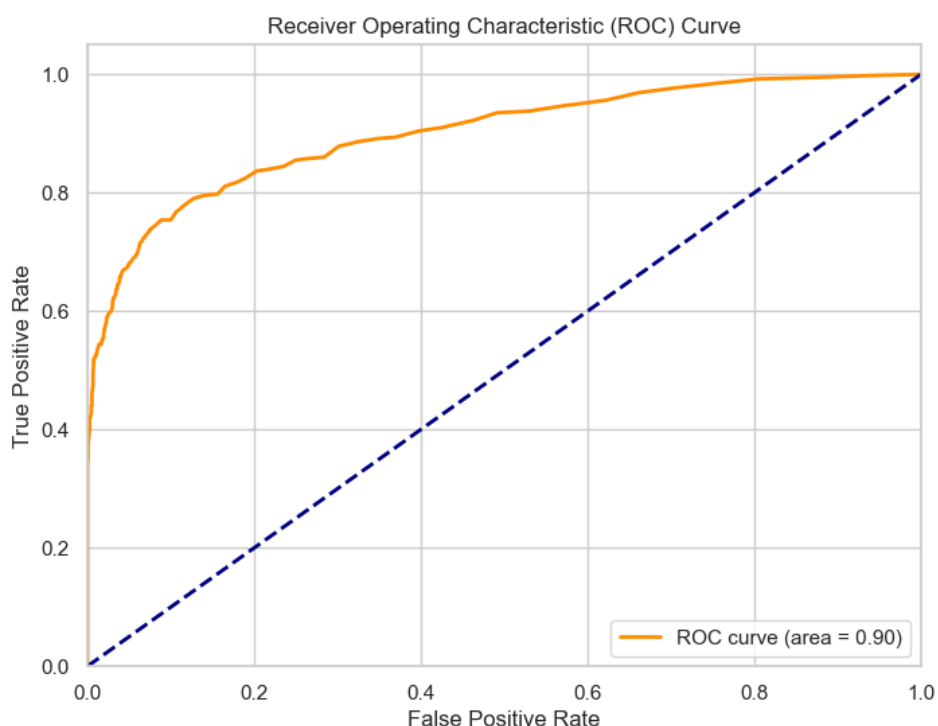


FIGURE 4.17 – Courbe ROC du modèle Forêts Aléatoires

### Description de la Courbe ROC

- **Aire sous la courbe (AUC)** : Avec une AUC de 0.90, la courbe indique une excellente capacité de discrimination du modèle. Plus l'AUC est proche de 1, mieux le modèle est capable de distinguer entre les classes positives et négatives sans erreur.

### Analyse de la Performance

- **Taux de Vrais Positifs (sensibilité)** : La courbe montre une progression rapide du taux de vrais positifs avec une augmentation modérée du taux de faux positifs, ce qui indique une bonne sensibilité. Cela signifie que le modèle est capable de détecter une grande proportion des cas réels de défaut avec un taux de faux positifs relativement bas.

- **Taux de Faux Positifs (1 - spécificité) :** Le modèle commence avec un faible taux de faux positifs, ce qui est idéal. À mesure que l'on abaisse le seuil de classification pour capturer plus de vrais positifs, le taux de faux positifs augmente, mais de manière contenue, ce qui reflète un bon compromis entre la capture des positifs et la limitation des erreurs de classification sur les négatifs.

La courbe monte rapidement au début, ce qui indique que même avec des seuils de classification stricts, le modèle parvient à identifier une bonne proportion des vrais cas positifs, ce qui est crucial dans des applications où manquer un cas positif peut avoir des conséquences graves. Ensuite, la montée progressive des faux positifs avec l'augmentation du taux de vrais positifs démontre que le modèle équilibre efficacement la sensibilité et la spécificité. Ce comportement est caractéristique des modèles bien calibrés dans des environnements où les deux types d'erreurs ont des conséquences importantes.

## Conclusion

Le modèle de forêts aléatoires avec une AUC de 0.90 démontre une forte capacité à classer correctement les cas de défaut et de non-défaut. Cette performance élevée suggère que le modèle est très approprié pour des tâches où il est crucial de distinguer avec précision entre différentes classes, telles que la classification de crédit dans les institutions financières, où une mauvaise classification peut entraîner des pertes significatives.

## 4.5.4 Machine à Vecteurs de Support

La machine à vecteurs de supports (SVM) est une méthode d'apprentissage supervisé largement utilisée pour les problèmes de classification. Elle fonctionne en trouvant l'hyperplan qui sépare les classes de manière optimale, maximisant ainsi la marge entre les exemples de différentes classes. Pour notre étude, nous avons utilisé un modèle de SVM pour prédire la probabilité qu'un crédit soit en défaut ou non.

Cette technique est particulièrement efficace pour les ensembles de données avec des frontières de décision complexes et non linéaires, grâce à l'utilisation de fonctions noyau. Afin d'évaluer la performance de notre modèle SVM, nous avons examiné la matrice de confusion et le rapport de classification, fournissant des insights détaillés sur sa capacité à distinguer entre les crédits en défaut et non défaut.

### a. Matrice de Confusion :

En examinant les résultats obtenus par l'application du modèle SVC sur nos données, nous avons d'abord visualisé la matrice de confusion, comme illustré dans la Figure 4.18 ci-dessous :

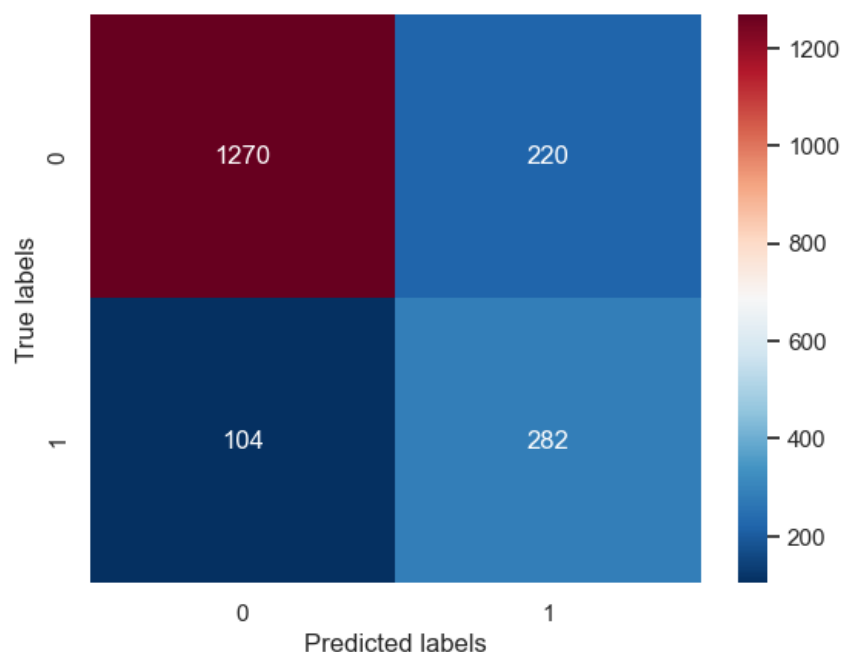


FIGURE 4.18 – Matrice de confusion du modèle SVM

Pour la classe 1, le modèle a correctement identifié 282 demandes de crédits et pour la classe 0 1270 valeurs alors qu'il a mal classé 220 valeurs dans la classe 1 et 104 valeurs dans la classe 0. Cette répartition indique que le SVM est plus capable à prédire la classe 0 (non défaut) que la classe 1 (défaut), comme en témoigne le nombre important de vrais négatifs comparé aux vrais positifs.

#### b. Rapport de classification

Pour mieux évaluer le modèle obtenu, nous avons visualisé le rapport de classification décrit dans la table 4.4 ci-dessous :

Class	Precision	Recall	F1-Score	Support
0	0.92	0.85	0.89	1490
1	0.56	0.73	0.64	386
<b>Accuracy</b>			0.83	1876
<b>Macro Avg</b>	0.74	0.79	0.76	1876
<b>Weighted Avg</b>	0.85	0.83	0.84	1876

TABLE 4.4 – Rapport de classification du modèle SVC

#### Evaluation du Rapport de classification

##### Classe 0 (Non-Défaut)

- **Précision** : 0.92 - Indique une excellente capacité à identifier correctement les non-défauts.
- **Rappel** : 0.85 - Le modèle a réussi à identifier 85% des réels non-défauts.
- **Score F1** : 0.89 - Un score F1 élevé montre un bon équilibre entre la précision et le rappel pour cette classe.

##### Classe 1 (Défaut)

- **Précision** : 0.56 - La précision pour détecter les défauts est relativement basse, indiquant que lorsque le modèle prédit un défaut, il est correct seulement 56% du temps.
- **Rappel** : 0.73 - Le modèle a identifié 73% des défauts réels, ce qui est un résultat modéré.
- **Score F1** : 0.64 - Le score F1 pour les défauts est inférieur, révélant des améliorations nécessaires pour équilibrer la précision et le rappel dans cette classe.

#### Métriques Globales :

- **Précision Globale (Accuracy)** : 0.83 - Indique que le modèle est correct dans 83% des cas pour toutes les prédictions.

c. La courbe ROC (Receiver Operating Characteristic) pour le modèle SVC est présentée ci-dessous dans la figure 4.19 :

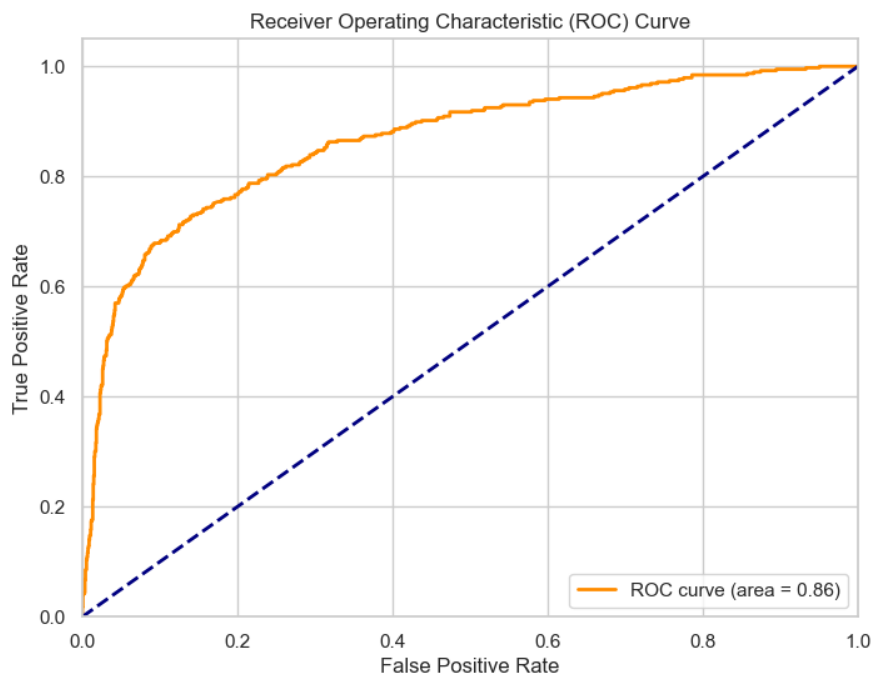


FIGURE 4.19 – Courbe ROC du modèle SVC

#### Description de la Courbe ROC :

- **Aire sous la courbe (AUC)** : 0.86 - Indique une bonne capacité du modèle à distinguer entre les classes de défaut et de non-défaut. Une AUC plus élevée suggère une meilleure performance globale du modèle.

#### Analyse de la Performance :

- La courbe ROC montre que le modèle augmente progressivement le taux de vrais positifs sans un accroissement disproportionné des faux positifs, ce qui est bénéfique. La courbe reste bien au-dessus de la ligne diagonale, qui représente un classificateur aléatoire, confirmant l'efficacité du modèle.

#### Conclusion

Le modèle SVC ou SVM démontre une performance solide dans la classification de non-défauts avec

une excellente précision et un bon rappel. Cependant, il y a une marge significative d'amélioration nécessaire pour la classification des défauts, notamment en augmentant la précision. La bonne performance générale du modèle est confirmée par une AUC élevée, indiquant une forte capacité discriminatoire.

## 4.6 Comparaison des Modèles

Dans cette section, nous comparons les performances de divers algorithmes de machine learning utilisés pour prédire les risques de crédit. Les modèles évalués incluent le K Neighbors Classifier, le Random Forest Classifier, la Régression Logistique et le SVC. Les critères de comparaison comprennent la Précision, le Rappel, le Score F1, l'Exactitude et l'AUC, qui sont cruciaux pour évaluer l'efficacité des modèles dans des scénarios de classification binaire où la distinction précise entre les classes de défaut et de non-défaut est essentielle.

Modèle	Précision	Rappel	F1-score	Exactitude	AUC
KNeighborsClassifier	0.805065	0.744670	0.763626	0.744670	0.782010
RandomForestClassifier	0.894156	0.898188	0.893435	0.898188	0.899830
LogisticRegression	0.818064	0.750533	0.770194	0.750533	0.816864
SVC	0.849710	0.827292	0.835075	0.827292	0.863816

TABLE 4.5 – Comparaison des résultats des modèles

L'analyse comparative révèle que le **Random Forest Classifier** offre les meilleures performances sur la plupart des métriques évaluées. Cette supériorité est particulièrement notable en termes d'AUC, ce qui indique une excellente capacité du modèle à discriminer entre les classes. Son score F1 élevé et son exactitude soutiennent également une bonne balance entre la précision et le rappel, crucial pour minimiser les faux positifs et les faux négatifs dans notre contexte d'application.

Le modèle SVC, tout en étant légèrement inférieur au Random Forest, démontre également de robustes capacités, spécialement en termes de précision et d'AUC, ce qui le rend approprié pour les scénarios où une forte discrimination est nécessaire.

D'autre part, les modèles *K Neighbors Classifier* et *Logistic Regression* affichent des performances respectables mais légèrement inférieures, suggérant que des améliorations dans le réglage des paramètres ou des stratégies de prétraitement des données pourraient être bénéfiques.

En considérant les résultats obtenus, le **Random Forest Classifier** est recommandé comme l'option la plus efficace pour la prédiction des risques de crédit dans notre étude. L'utilisation de ce modèle pourrait améliorer significativement la précision des décisions de crédit, réduisant les risques de pertes financières liées à des erreurs de classification.

## 4.7 Conclusion

Ce chapitre a démontré l'efficacité de divers modèles de machine learning dans la prédiction des défauts de crédit. Après une évaluation rigoureuse, le *RandomForest Classifier* s'est distingué comme le modèle le plus performant, offrant la meilleure exactitude et l'AUC la plus élevée parmi les modèles

testés.

En revanche, bien que le *KNeighbors Classifier* et la Logistic Regression aient offert des insights initiaux, ils se sont révélés moins performants pour notre ensemble de données spécifique, nécessitant potentiellement plus de réglages ou une sélection de caractéristiques plus ciblée pour améliorer leurs performances.

En conclusion, les résultats obtenus dans ce chapitre orienteront les choix futurs de modélisation et de mise en œuvre dans les systèmes de prédiction de crédit, en favorisant l'utilisation de modèles robustes et précis comme le RandomForest pour l'intégrer dans l'algorithme Bandit Multi-bras pour la sélection des crédits.



# Chapitre 5

## Implémentation et Evaluation de l'Algorithme de Bandit Multi-bras

Ce chapitre est dédié à l'implémentation de la méthode de sélection des demandes de crédits qui présentent un risque de défaut pour le traitement manuel, en se concentrant sur l'application de l'algorithme du Bandit Multi-bras. Après avoir développé des modèles prédictifs de probabilité de défaut dans le chapitre précédent, nous nous tournons maintenant vers l'optimisation du processus de sélection. L'algorithme du Bandit Multi-bras, une technique d'apprentissage par renforcement, sera utilisé pour identifier les crédits nécessitant une vérification manuelle. Nous détaillerons le principe de fonctionnement de cet algorithme dans notre cas, sa mise en œuvre, et les résultats obtenus. Cette approche vise à améliorer l'efficacité et la précision de la sélection des crédits à haut risque de défaut, contribuant ainsi à une meilleure gestion des demandes de crédit chez Quion.

### 5.1 Principe de fonctionnement de l'algorithme de Bandit Multi-Bras

Une fois que les données des demandeurs de crédits sont réparties sur les différents intervalles de montant, un agent de l'algorithme de Bandit Multi-Bras commence par sélectionner aléatoirement un intervalle et un crédit au sein de cet intervalle. Le crédit sélectionné subit ensuite une étape de prétraitement pour prédire la variable "*loan\_status*", indiquant s'il s'agit d'un défaut ou non-défaut.

La prochaine étape consiste à calculer une fonction de récompense (reward) pour cet intervalle, basée sur le *loan\_status* et le montant du crédit. En fonction de la stratégie adoptée par l'agent, il peut soit exploiter cet intervalle (c'est-à-dire continuer à sélectionner des crédits dans le même intervalle), soit explorer d'autres intervalles (c'est-à-dire choisir un crédit dans un intervalle différent).

Grâce à ce processus d'exploration et d'exploitation, l'agent apprend progressivement la structure de chaque intervalle. Il associe ensuite à chaque intervalle un pourcentage de sélection, basé sur le risque de défaut des crédits présents dans chaque intervalle. Cette approche permet d'optimiser la sélection des crédits à vérifier manuellement, en ciblant de manière plus précise ceux présentant le plus grand risque de défaut.

### 5.1.1 La stratégie Epsilon Greedy

L'algorithme Epsilon-Greedy est utilisé pour équilibrer l'exploration et l'exploitation lors de la sélection des crédits qui ont un risque élevé de défaut.

Dans l'algorithme 1 présent dans l'annexe E.1, chaque option représente un intervalle de prêts et l'objectif est de déterminer quels prêts vérifier manuellement pour maximiser les gains tout en minimisant les risques associés.

L'algorithme commence par initialiser des estimations de récompenses pour chaque intervalle de prêts et des compteurs pour suivre les sélections. Il alterne ensuite entre exploration et exploitation : avec une probabilité  $\epsilon$  (epsilon), il explore de nouvelles options en sélectionnant un intervalle de prêts de manière aléatoire, tandis qu'avec une probabilité  $1 - \epsilon$ , il exploite les options connues en sélectionnant l'intervalle avec la plus grande estimation de récompense. Après chaque sélection, l'algorithme met à jour ses estimations de récompense en fonction des récompenses réelles obtenues. Le processus se poursuit jusqu'à ce qu'un critère prédéfini soit atteint, par exemple lorsque 60% du total des prêts ont été sélectionnés (seuil fixé et choisi selon les besoins de Quion).

Enfin, l'algorithme évalue ses performances en calculant des métriques telles que le pourcentage d'actions optimales. En combinant exploration et exploitation, l'algorithme Epsilon-Greedy offre une approche équilibrée pour la sélection des crédits à risque.

### 5.1.2 La stratégie Upper Confidence Bound UCB1

L'algorithme UCB1 équilibre exploration et exploitation en utilisant des bornes de confiance pour estimer les récompenses des crédits.

Dans l'algorithme 2 présenté dans l'annexe E.1, chaque bras représente un intervalle de prêts, et l'algorithme vise à déterminer quels prêts vérifier manuellement. Il commence par initialiser des estimations de récompense ( $Q$ ) et des compteurs de sélections ( $N$ ) pour chaque intervalle de prêts. Ensuite, il effectue une première phase où chaque intervalle est sélectionné au moins une fois pour obtenir une estimation initiale de ses récompenses. La principale boucle de l'algorithme utilise la formule UCB pour équilibrer l'exploration et l'exploitation. Cette formule ajoute une valeur de confiance (dépendant de la variance et du nombre de sélections) à l'estimation actuelle de la récompense de chaque intervalle. L'intervalle avec la valeur UCB la plus élevée est sélectionné pour le prochain prêt.

Chaque fois qu'un prêt est sélectionné, l'algorithme met à jour les estimations de récompense et les compteurs de sélections en fonction du retour obtenu (reward). Si le prêt est risqué (indiqué par une probabilité de défaut 1), il est compté séparément pour suivre le montant des prêts risqués sélectionnés. Ce processus se poursuit jusqu'à ce que 60% du total des prêts aient été sélectionnés (seuil fixé et choisi selon les besoins de Quion).

Finalement, l'algorithme calcule le pourcentage des actions optimales (c'est-à-dire les sélections qui ont donné la récompense maximale ou le maximum des crédits ayant un risque de défaut) pour évaluer la performance de la stratégie UCB1. Cette approche permet une sélection judicieuse des prêts risqués à vérifier manuellement.

## 5.2 Evaluation des algorithmes

L'évaluation des algorithmes est une étape cruciale pour déterminer leur performance et leur efficacité dans notre contexte. Dans cette section, nous analysons deux algorithmes de sélection de prêts risqués : l'Epsilon-Greedy et l'UCB1 (*Upper Confidence Bound*). Ces algorithmes sont appliqués à un jeu de données contenant différents intervalles de prêts, et leurs performances sont mesurées à travers divers indicateurs tels que le pourcentage d'actions optimales, la sélection de prêts risqués et la proportion de montants risqués par rapport au montant total des prêts.

Les résultats sont présentés sous forme de graphiques et de tableaux accompagnés d'interprétations détaillées, permettant de mieux comprendre le comportement de chaque algorithme et leur capacité à équilibrer exploration et exploitation dans le processus de sélection des prêts risqués.

### 5.2.1 Résultats de l'algorithme Epsilon-Greedy

L'algorithme Epsilon-Greedy a été utilisé pour sélectionner les prêts risqués dans notre jeu de données avec différents intervalles de prêts. Les résultats sont présentés à travers plusieurs graphiques, chacun accompagné d'une description et d'une interprétation.

Nous avons choisi le paramètre epsilon de 0.1 pour garantir un équilibre efficace entre exploration et exploitation. Cette valeur permet à l'algorithme Epsilon-Greedy de tester de nouvelles options (10% du temps) tout en exploitant les meilleures actions connues (90% du temps), assurant ainsi une optimisation robuste de la sélection des prêts risqués. Ce choix est appuyé par des études telles que Sutton et Barto (2018), qui recommandent une petite valeur d'epsilon pour maintenir cet équilibre crucial [Sutton and Barto, 2018].

#### Pourcentage d'Actions Optimales vs Étapes

Le graphique 5.1 montre le pourcentage des actions optimales sélectionnées par l'algorithme Epsilon-Greedy au fil des étapes.



FIGURE 5.1 – Pourcentage d'Actions Optimales vs Étapes

Nous observons une stabilisation autour de 20% des actions optimales. Au début, l'algorithme explore différentes options, ce qui explique les fluctuations initiales. Après environ 500 étapes, le pourcentage d'actions optimales se stabilise autour de 20%, indiquant que l'algorithme a trouvé un bon équilibre entre exploration et exploitation. La stabilisation à un niveau inférieur à 100% suggère que l'algorithme continue à explorer même après avoir trouvé des actions optimales potentielles, ce qui est typique du comportement de l'algorithme Epsilon-Greedy pour garantir un bon équilibre entre exploration et exploitation.

## Pourcentages de Sélection par Intervalle

Le tableau 5.2 résume les pourcentages de prêts risqués et de montants risqués sélectionnés dans chaque intervalle.

	Interval	Risky Loans Selected % (Out of Risky Loans in Interval)	Risky Loans Selected % (Out of Total Loans in Interval)	Risky Amount Selected % (Out of Amount in Interval)
0	0-1500	92.372881	18.136439	18.195730
1	1500- 2400	44.363636	17.354196	8.643108
2	2400- 3200	96.336207	21.847507	21.902962
3	3200- 4100	69.823789	20.732505	14.611654
4	4100- 5000	63.070539	18.650307	12.291456

FIGURE 5.2 – Pourcentage de Sélection des Montants Risqués par Intervalle pour l'algorithme Epsilon-Greedy

### Interprétations :

- Le premier pourcentage est celui du nombre des crédits risqués identifiés dans l'intervalle par rapport au nombre total des crédits risqués de cet intervalle. Ce pourcentage est calculé pour évaluer la capacité de l'algorithme à identifier les crédits risqués.

- **Pour l'intervalle 0-1500** : L'algorithme arrive à identifier 92.37% des prêts risqués (en nombre) existants dans cet intervalle. Cela signifie que parmi tous les prêts risqués situés dans l'intervalle de montant 0-1500, l'algorithme a réussi à en identifier 92.37%. En d'autres termes, l'algorithme est très efficace dans cet intervalle, puisqu'il arrive à détecter presque tous les prêts risqués.
- **Pour l'intervalle 1500-2400** : L'algorithme arrive à identifier 44.36% des prêts risqués (en nombre) dans cet intervalle. Cela indique une performance relativement faible, car moins de la moitié des prêts risqués sont détectés. L'algorithme a donc des difficultés à identifier les prêts risqués dans cette tranche de montant.
- **Pour l'intervalle 2400-3200** : L'algorithme arrive à identifier 96.34% des prêts risqués (en nombre) existants dans cet intervalle. Cela montre une excellente performance de l'algorithme, car il parvient à repérer presque tous les prêts risqués. L'algorithme est donc très efficace pour

cette tranche de montant.

- **Pour l'intervalle 3200-4100 :** L'algorithme arrive à identifier 69.82% des prêts risqués (en nombre) dans cet intervalle. Ce pourcentage est assez élevé, ce qui signifie que l'algorithme est relativement efficace, même s'il n'est pas aussi performant que dans les intervalles 0-1500 et 2400-3200. L'algorithme parvient à détecter la majorité des prêts risqués dans cette tranche de montant.
  - **Pour l'intervalle 4100-5000 :** L'algorithme arrive à identifier 63.07% des prêts risqués existants dans cet intervalle. Ce pourcentage est modéré, ce qui indique que l'algorithme est assez efficace pour cette tranche de montant, identifiant plus de la moitié des prêts risqués, mais pas autant que dans les autres intervalles plus performants.
- Le deuxième pourcentage est le nombre des crédits risqués sélectionnés de cet intervalle par rapport au nombre total des crédits de l'intervalle, indiquant la proportion de crédits risqués parmi tous les crédits de l'intervalle.
    - **Intervalle 0-1500 :** Le pourcentage des crédits risqués identifiés en nombre parmi tous les crédits de cet intervalle est de 18.14%.
    - **Intervalle 1500-2400 :** Le pourcentage des crédits risqués identifiés parmi tous les crédits de cet intervalle est de 17.35%.
    - **Intervalle 2400-3200 :** Le pourcentage des crédits risqués identifiés parmi tous les crédits de cet intervalle est de 21.85%.
    - **Intervalle 3200-4100 :** Le pourcentage des crédits risqués identifiés parmi tous les crédits de cet intervalle est de 20.73% en nombre.
    - **Intervalle 4100-5000 :** Le pourcentage des crédits risqués sélectionnés parmi tous les crédits de l'intervalle est 18.65% en nombre.
  - Le troisième pourcentage concerne les montants : il représente le montant risqué sélectionné pour un traitement manuel dans cet intervalle par rapport au montant total de l'intervalle. C'est le pourcentage le plus pertinent pour l'entreprise, car il prend en compte à la fois le risque et le montant, qui est très important.
    - **Intervalle 0-1500 :** Le montant risqué identifié dans cet intervalle par rapport au montant total de l'intervalle est de 18.19%.
    - **Intervalle 1500-2400 :** Le montant risqué identifié dans cet intervalle par rapport au montant total de l'intervalle est de 8.64%. Dans cette tranche de montant, seulement 8.64% du montant total des crédits est identifié comme risqué, ce qui indique une proportion relativement faible.
    - **Intervalle 2400-3200 :** Le montant risqué identifié dans cet intervalle par rapport au montant total de l'intervalle est de 21.90%.

- **Intervalle 3200-4100** : Le montant risqué identifié dans cet intervalle par rapport au montant total de l'intervalle est de 14.61%.
- **Intervalle 4100-5000** : Le montant risqué identifié dans cet intervalle par rapport au montant total de l'intervalle est de 12.29%.

En résumé, ce tableau présente la performance de l'algorithme Epsilon-Greedy à travers différents pourcentages. Cependant, nous ne pouvons pas conclure définitivement sur sa performance sans une comparaison avec la réalité.

### Pourcentage du Montant Risqué Sélectionné par Intervalle de Prêt par rapport au Montant Total

Le graphique 5.3 illustre le pourcentage des montants risqués sélectionnés par intervalle de prêt par rapport au montant total. Ces pourcentages révèlent de quel intervalle l'algorithme sélectionne la plus grande proportion, en tenant compte du fait que Quion cible les intervalles avec les montants les plus élevés.

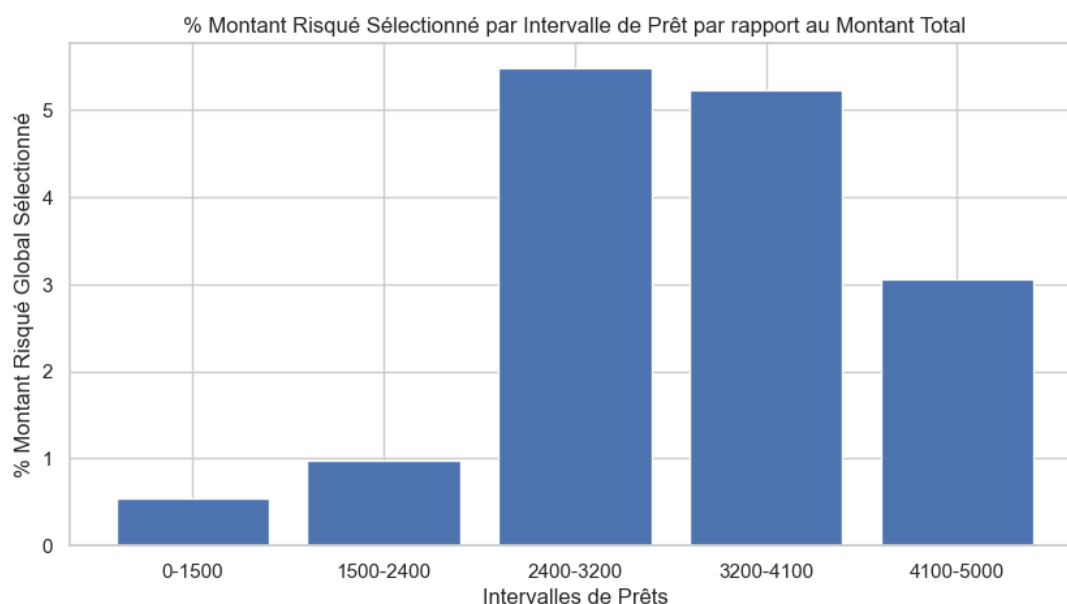


FIGURE 5.3 – Pourcentage de Montant Risqué Sélectionné par Intervalle par rapport au Montant Total pour Epsilon-Greedy

### Interprétation :

- L'intervalle 2400-3200 présente le pourcentage le plus élevé, avec environ 5.49% du montant total traité. Cela indique une forte proportion de montants risqués sélectionnés dans cet intervalle par rapport au montant total des prêts.
- Les intervalles 3200-4100 et 4100-5000 montrent également des proportions élevées, avec respectivement 5.23% et 3.06% du montant total traité, soulignant une sélection significative de

montants risqués dans ces intervalles.

- En revanche, les intervalles 0-1500 et 1500-2400 montrent des proportions plus faibles respectivement environ 0.53% et 0.98%.

En conclusion, l'algorithme Epsilon-Greedy arrive à détecter en total 15.29% montants de crédits ayant une probabilité de faire défaut qui correspond au pourcentage total sélectionné. Il montre une performance robuste pour la sélection des prêts risqués. Cet algorithme permet une exploration continue, assurant ainsi une bonne balance entre exploration et exploitation.

### 5.2.2 Résultats de l'algorithme UCB1

L'algorithme UCB1 (Upper Confidence Bound) a été évalué sur un jeu de données de prêts avec un niveau de confiance  $c=2$ . Les résultats sont présentés à travers plusieurs graphiques, chacun accompagné d'une description et d'une interprétation.

#### Pourcentage d'Actions Optimales vs Étapes

Le graphique 5.4 montre le pourcentage des montants de prêts risqués sélectionnés par rapport au montant total des prêts disponibles dans chaque intervalle.

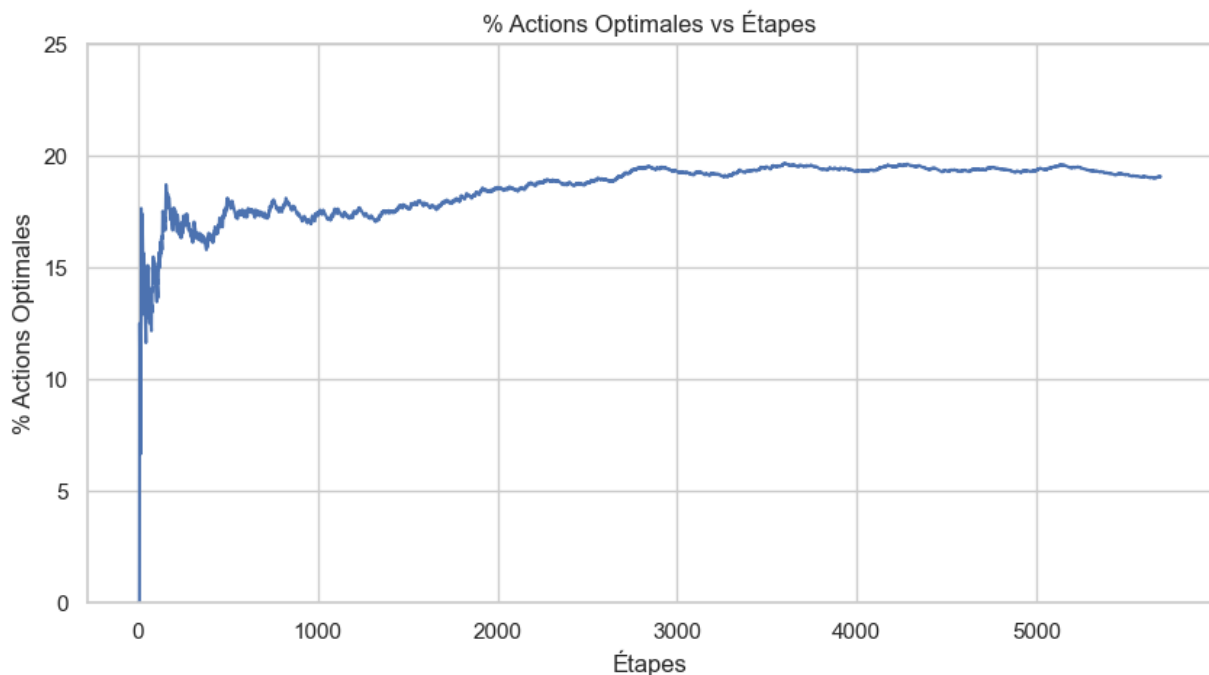


FIGURE 5.4 – Pourcentage d'Actions Optimales vs Étapes pour UCB1

Nous observons une montée initiale rapide, suivie d'une stabilisation autour de 20%. Au début, l'algorithme explore les différentes options, ce qui explique les fluctuations initiales. Avec le temps, l'algorithme se stabilise autour de 20%, indiquant une amélioration progressive de la sélection des actions optimales. La stabilisation à un niveau inférieur à 100% suggère que l'algorithme continue à explorer même après avoir trouvé des actions potentiellement optimales, ce qui est typique du comportement de l'algorithme UCB1 pour garantir un bon équilibre entre exploration et exploitation.

## Pourcentages de Sélection par Intervalle

Le tableau de la figure 5.5 résume les pourcentages de prêts risqués et de montants risqués sélectionnés dans chaque intervalle :

	Interval	Risky Loans Selected % (Out of Risky Loans in Interval)	Risky Loans Selected % (Out of Total Loans in Interval)	Risky Amount Selected % (Out of Amount in Interval)
0	0-1500	92.372881	18.136439	18.195730
1	1500- 2400	58.909091	16.963351	11.790316
2	2400- 3200	81.034483	21.949796	18.380338
3	3200- 4100	57.268722	19.847328	11.895668
4	4100- 5000	85.477178	18.475336	16.636751

FIGURE 5.5 – Pourcentage de Sélection des Montants risqués par Intervalle pour l'algorithme UCB1

### Interprétations :

- Le premier pourcentage est celui du nombre des crédits risqués identifiés dans l'intervalle par rapport au nombre total des crédits risqués de cet intervalle. Ce pourcentage est calculé pour évaluer la capacité de l'algorithme à identifier les crédits risqués.

- **Pour l'intervalle 0-1500** : L'algorithme arrive à identifier 92.37% des prêts risqués (en nombre) existants dans cet intervalle. Cela signifie que parmi tous les prêts risqués situés dans l'intervalle de montant 0-1500, l'algorithme a réussi à en identifier 92.37%. En d'autres termes, l'algorithme est très efficace dans cet intervalle, puisqu'il arrive à détecter presque tous les prêts risqués.
- **Pour l'intervalle 1500-2400** : L'algorithme arrive à identifier 58.91% des prêts risqués (en nombre) dans cet intervalle. Cela indique une performance relativement modérée, car plus de la moitié des prêts risqués sont détectés. L'algorithme a donc une efficacité modérée à identifier les prêts risqués dans cette tranche de montant.
- **Pour l'intervalle 2400-3200** : L'algorithme arrive à identifier 81.03% des prêts risqués (en nombre) existants dans cet intervalle. Cela montre une bonne performance de l'algorithme, car il parvient à repérer une grande majorité des prêts risqués. L'algorithme est donc assez efficace pour cette tranche de montant.
- **Pour l'intervalle 3200-4100** : L'algorithme arrive à identifier 57.27% des prêts risqués (en nombre) dans cet intervalle. Ce pourcentage est relativement modéré, ce qui signifie que l'algorithme est moyennement efficace, et a des difficultés à détecter la majorité des prêts risqués dans cette tranche de montant.
- **Pour l'intervalle 4100-5000** : L'algorithme arrive à identifier 85.48% des prêts risqués existants dans cet intervalle. Ce pourcentage est élevé, ce qui indique que l'algorithme est assez



efficace pour cette tranche de montant, identifiant la majorité des prêts risqués.

- Le deuxième pourcentage est le nombre des crédits risqués sélectionnés de cet intervalle par rapport au nombre total des crédits de l'intervalle, indiquant la proportion de crédits risqués parmi tous les crédits de l'intervalle.

- **Intervalle 0-1500** : Le pourcentage des crédits risqués identifiés en nombre parmi tous les crédits de cet intervalle est de 18.14%.
- **Intervalle 1500-2400** : Le pourcentage des crédits risqués identifiés parmi tous les crédits de cet intervalle est de 16.96%.
- **Intervalle 2400-3200** : Le pourcentage des crédits risqués identifiés parmi tous les crédits de cet intervalle est de 21.95%.
- **Intervalle 3200-4100** : Le pourcentage des crédits risqués identifiés parmi tous les crédits de cet intervalle est de 19.85% en nombre.
- **Intervalle 4100-5000** : Le pourcentage des crédits risqués sélectionnés parmi tous les crédits de l'intervalle est de 18.48% en nombre.

- Le troisième pourcentage concerne les montants : il représente le montant risqué sélectionné et qui sera traité manuellement dans cet intervalle par rapport au montant total de l'intervalle. C'est le pourcentage le plus pertinent pour l'entreprise, car il prend en compte à la fois le risque et le montant, qui est très important.

- **Intervalle 0-1500** : Le montant risqué identifié dans cet intervalle par rapport au montant total de l'intervalle est de 18.20%.
- **Intervalle 1500-2400** : Le montant risqué identifié dans cet intervalle par rapport au montant total de l'intervalle est de 11.79%. Dans cette tranche de montant, seulement 11.79% du montant total des crédits est identifié comme risqué, ce qui indique une proportion relativement faible.
- **Intervalle 2400-3200** : Le montant risqué identifié dans cet intervalle par rapport au montant total de l'intervalle est de 18.38%.
- **Intervalle 3200-4100** : Le montant risqué identifié dans cet intervalle par rapport au montant total de l'intervalle est de 11.90%.
- **Intervalle 4100-5000** : Le montant risqué identifié dans cet intervalle par rapport au montant total de l'intervalle est de 16.64%.

En résumé, ce tableau présente la performance de l'algorithme *Upper Confidence Bound (UCB)* à travers différents pourcentages. Cependant, nous ne pouvons pas conclure définitivement sur sa performance sans une comparaison avec la réalité.

**Pourcentage du Montant Risqué Sélectionné par Intervalle de Prêt par rapport au Montant Total**

Le graphique 5.6 montre le pourcentage des montants risqués sélectionnés par intervalle de prêt par rapport au montant total :

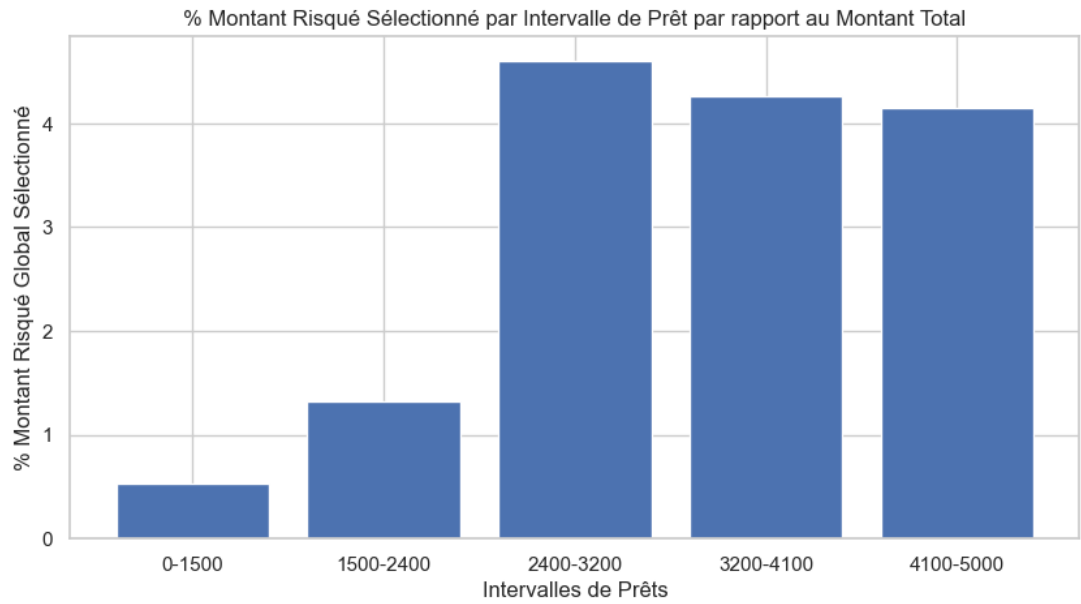


FIGURE 5.6 – Pourcentage de Sélection des Montants risqués par Intervalle par rapport au Montant Total pour l’UCB1

**Interprétation :**

- Les intervalles 2400-3200 et 3200-4100 présentent les pourcentages les plus élevés, avec respectivement environ 4.61% et 4.26% du montant total des demandes de crédits. Cela indique une forte proportion de montants risqués sélectionnés dans ces intervalles par rapport au montant total des prêts.
- L’intervalle 4100-5000 a également une proportion élevée, avec 4.15%, soulignant une sélection significative de montants risqués.
- En revanche, les intervalles 0-1500 et 1500-2400 montrent respectivement des proportions plus faibles (environ 0.54% et 1.3%), ce qui peut suggérer une moindre disponibilité de montants élevés risqués dans ces catégories.

En conclusion, l’algorithme UCB1 arrive à sélectionner de 14.88% montants de crédits ayant une probabilité de faire défaut par rapport au montant total traité qui correspond au pourcentage total sélectionné. Cet algorithme montre une performance robuste pour la sélection des prêts risqués, avec une efficacité notable dans les intervalles de prêts élevés.

### 5.2.3 Comparaison des Algorithmes Epsilon-Greedy et Upper Confidence Bound UCB1

En comparant les algorithmes Epsilon-Greedy et UCB1 pour la sélection des prêts risqués, nous observons que l'Epsilon-Greedy détecte un pourcentage légèrement supérieur de crédits risqués (15.29%) par rapport à UCB1 (14.88%). Cependant, UCB1 se distingue par une meilleure performance dans la détection des crédits risqués dans les intervalles de montants élevés. L'intervalle 4100-5000, où les montants de prêts sont plus significatifs, montre une proportion plus élevée de montants risqués sélectionnés par rapport au montant total existant dans l'intervalle avec UCB1 égale à 16.63%. En revanche, Epsilon-Greedy détecte mieux les prêts risqués dans les intervalles de montants plus bas. Cette capacité d'UCB1 à identifier de manière plus fiable les prêts risqués de grande valeur le rend plus adapté à des applications où la précision dans la gestion des montants élevés est cruciale. Par conséquent, malgré un léger avantage de l'Epsilon-Greedy en termes de pourcentage total de crédits risqués détectés, l'UCB1 est préféré pour sa robustesse et son efficacité accrue dans la gestion des prêts risqués de montants élevés.

## 5.3 Comparaison des Méthodes de Sélection

Dans cette section, nous comparons d'une part les résultats de l'algorithme du bandit multi-bras UCB1 pour l'identification des prêts risqués par rapport aux crédits risqués réellement existants dans la base de données (en total et pour chaque intervalle) et d'autre part les résultats obtenus par rapport aux pourcentages fixes de la méthode actuellement utilisée par Quion. Ainsi, nous présentons le tableau comparatif 5.1 ci-dessous :

Intervalle de Prêt	UCB1	Échantillonnage	Réalité
Total sélectionné	14.88%	20.7%	20.53%
0-1500	18.20%	3.3%	19.84%
1500-2400	11.79%	12.4%	19.83%
2400-3200	18.38%	27.7%	22.64%
3200-4100	11.90%	50.1%	20.79%
4100-5000	16.64%	82.5%	19.50%

TABLE 5.1 – Comparaison des Pourcentages de Sélection entre UCB1, la Méthode d'Échantillonnage et la Réalité

### Comparaison de la performance de l'UCB1 avec la réalité

Dans cette partie, nous évaluons la capacité de l'algorithme UCB1 à identifier les montants des crédits risqués dans chaque intervalle. En comparant le pourcentage des crédits risqués identifiés par l'algorithme du Bandit Multi-Bras UCB1 dans chaque intervalle avec le pourcentage des crédits réellement risqués existants dans chaque intervalle, nous pouvons constater que :

- Pour les prêts de 0 à 1500, UCB1 a identifié 18.20% comme montants risqués par rapport au montant total de cet intervalle, contre 19.84% en réalité. La différence est minime, ce qui montre que l'algorithme est performant pour cette tranche.

- Dans cette tranche de 1500 à 2400, UCB1 a identifié 11.79% comme montants risqués par rapport au montant total de cet intervalle, tandis que la réalité est de 19.83%. Il y a une sous-estimation notable, ce qui suggère une opportunité d'amélioration pour l'algorithme dans cette catégorie.
- Pour les prêts de 2400 à 3200, UCB1 a identifié 18.38% comme montants risqués par rapport au montant total de cet intervalle, contre 22.64% en réalité. Bien qu'il y ait une différence, l'algorithme reste relativement proche de la réalité.
- Dans cette tranche de 3200 à 4100, UCB1 a identifié 11.90% comme montants risqués par rapport au montant total de cet intervalle, alors que la réalité est de 20.79%. Cette sous-estimation est importante, indiquant un besoin d'amélioration de l'algorithme pour mieux identifier les prêts risqués dans cette catégorie.
- Pour les prêts de 4100 à 5000, UCB1 a identifié 16.64% comme montants risqués par rapport au montant total de cet intervalle, contre 19.50% en réalité. La différence est relativement faible, montrant que l'algorithme est assez précis pour cet intervalle.

Globalement, l'algorithme UCB1 a identifié 14,88% du montant total des prêts comme étant risqués, tandis que la réalité montre que 20.53 % du montant total des prêts sont effectivement risqués. Malgré cette différence, nous pouvons dire que l'algorithme est performant et assez proche de la réalité.

### **Comparaison des résultats de l'UCB1 avec les résultats de l'échantillonnage**

En comparant les pourcentages des montants sélectionnés obtenus par l'algorithme et les pourcentages des montants sélectionnés obtenus en utilisant l'échantillonnage aléatoire dans la méthode actuelle utilisée par Quion, nous avons eu une différence significative :

- Pour l'intervalle 0-1500 : UCB1 sélectionne 18.20% du montant total des prêts de l'intervalle en les identifiant comme des montants risqués, contre seulement 3.3% du montant total de l'intervalle pour la méthode d'échantillonnage et qui sont aléatoire. Cela montre une meilleure performance d'UCB1 dans cet intervalle de montants plus faibles.
- Pour l'intervalle 1500-2400 : Les pourcentages sont relativement proches, avec 11.79% du montant total des prêts de l'intervalle (identifiés comme des montants de crédits risqués) pour UCB1 et 12.4% du montant total de l'intervalle pour l'échantillonnage.
- Pour l'intervalle 2400-3200 : UCB1 sélectionne 18.38% du montant total des prêts de l'intervalle (identifiés comme des montants de crédits risqués), tandis que l'échantillonnage en sélectionne aléatoirement 27.7%.
- Pour l'intervalle 3200-4100 : UCB1 sélectionne 11.90% du montant total des prêts de l'intervalle en les identifiant comme des montants risqués, contre un pourcentage beaucoup plus élevé et aléatoire de 50.1% pour l'échantillonnage.
- Pour l'intervalle 4100-5000 : UCB1 sélectionne 16.64% du montant total des prêts de l'intervalle en les identifiant comme des montants risqués, contre un pourcentage très élevé 82.5%

pour l'échantillonnage.

Globalement, l'algorithme UCB1 sélectionne en total un pourcentage de montants de crédits risqués de 14.88%, qui est non seulement inférieur au pourcentage de 20.7% du montant total obtenu avec la méthode d'échantillonnage mais aussi comporte seulement des montants risqués. Cela indique que l'algorithme UCB1 est plus sélectif et ciblé dans sa détection des prêts risqués.

**En conclusion**, les pourcentages obtenus par la méthode actuelle d'échantillonnage utilisée par Quion sont proportionnels aux montants des prêts, ce qui signifie que cette méthode tend à sélectionner des prêts sans tenir compte du risque de défaut mais plutôt du montant. En revanche, les pourcentages obtenus par l'algorithme du Bandit Multi-Bras avec la stratégie UCB1 sont dépendants du risque de faire défaut. Ainsi, l'algorithme du Bandit Multi-Bras parvient à cibler les demandes de crédit ayant une probabilité de défaut plus élevée, ce qui explique pourquoi le pourcentage global de montants sélectionnés (14.88%) est inférieur à celui de l'échantillonnage aléatoire (20.7%). Cela démontre l'efficacité de l'algorithme du Bandit Multi-Bras avec la stratégie UCB1 dans la sélection de crédits à risque, en se concentrant sur ceux qui sont réellement risqués plutôt que sur une sélection aléatoire et en étant adaptable à la variation des données. Ceci répond aux objectifs définis par Quion, qui visent à cibler uniquement les crédits à risque et à optimiser le pourcentage de traitement manuel en le réduisant, offrant ainsi à la fois un gain de temps et d'argent.

## 5.4 Conclusion

Ce chapitre explore l'application des algorithmes de Bandit Multi-Bras, notamment Epsilon-Greedy et UCB1, dans la sélection des crédits à risque. Les résultats de l'étude révèlent que l'algorithme UCB1 se distingue par sa performance supérieure, permettant une détection plus efficace des crédits risqués tout en réduisant le pourcentage global de sélection. Ceci permettra un gain de temps et d'argent significatif pour Quion, tout en contribuant à minimiser les risques associés aux décisions prises par le système de scoring intégré dans le processus automatique.

# Conclusion et Perspectives

L'objectif de notre travail était de résoudre la problématique centrale de l'optimisation du processus de traitement des demandes de crédit en ligne chez Blauwtrust Groep. Nous avons identifié que le traitement manuel actuel des crédits, sélectionnés de manière aléatoire, ne cible pas spécifiquement les crédits à risque élevé de défaut, ce qui peut entraîner une inefficacité opérationnelle. Notre objectif principal a été d'améliorer ce processus en développant un modèle prédictif de la probabilité de défaut pour cibler de manière plus précise les crédits à risque élevé. En utilisant des techniques avancées d'apprentissage automatique, nous avons élaboré ce modèle et l'avons intégré dans un algorithme d'apprentissage par renforcement pour optimiser la sélection des crédits à risque élevé. Cette approche vise à améliorer l'efficacité du processus de sélection tout en réduisant la nécessité de vérifications manuelles et en minimisant le risque de non-sélection des crédits à haut risque.

Pour répondre à cette problématique, nous avons d'abord étudié et mis en œuvre différents modèles de classification pour la modélisation de la probabilité de défaut de crédit. Les modèles explorés incluent *K-Nearest Neighbors*, la régression logistique, *Random Forest*, et *Support Vector Machine*. Chacun de ces modèles a été évalué à l'aide de métriques de performance appropriées. Cette évaluation nous a permis de déterminer que le modèle *Random Forest* était le plus performant, justifiant ainsi son intégration dans l'algorithme de Bandit Multi-Bras pour cibler les demandes de crédit à haut risque de défaut.

Ensuite, nous avons exploré l'utilisation d'algorithmes de bandits multi-bras dans le cadre de l'apprentissage par renforcement. Nous avons mis en œuvre et comparé les stratégies *Epsilon-Greedy* et *Upper Confidence Bound* (UCB1). Les résultats obtenus montrent que l'algorithme UCB1 a fourni des performances robustes en termes de sélection de prêts. Il parvient non seulement à détecter la plupart des crédits à haut risque de défaut, mais aussi à réduire le pourcentage de sélection pour un traitement manuel. En minimisant le pourcentage de vérification manuelle, notre travail permettra un gain de temps et de ressources financières pour Quion, une adaptabilité par rapport aux variations des données, et une supervision significative de la performance du modèle de scoring, ce qui facilitera la prise de décisions importantes concernant l'octroi des crédits.

Cependant, malgré les résultats prometteurs, les pourcentages des montants de prêts risqués identifiés dans le deuxième et le quatrième intervalles sont relativement faibles. Cette constatation suggère que notre modèle peut avoir des difficultés à détecter les prêts à risque dans certaines plages de données, ce qui peut limiter sa capacité à fournir des prédictions précises dans ces cas spécifiques.

Pour améliorer ces résultats, une augmentation du volume de demandes de crédit pourraient contribuer à la fois à améliorer la performance de l'algorithme d'apprentissage par renforcement dans les intervalles décrits et renforcer la précision et la fiabilité des prédictions du modèle *Random Forest*.

Ceci permettrait une meilleure représentation de la variabilité des profils de risque des emprunteurs, améliorant ainsi la capacité des modèles d'apprentissage automatique et de l'algorithme du Bandit Multi-Bras à généraliser efficacement sur des cas divers.

De plus, pour enrichir encore les modèles, l'intégration de variables macroéconomiques pourrait s'avérer bénéfique. Ces variables pourraient permettre de saisir des tendances et des événements macroéconomiques susceptibles d'influencer les risques de défaut de paiement des crédits à la consommation. En incorporant ces facteurs externes, les modèles seraient mieux équipés pour anticiper et répondre aux fluctuations du marché.

Finalement, ce projet ouvre la voie à des améliorations significatives dans la supervision des systèmes de scoring à travers la sélection des crédits en exploitant les puissantes capacités des algorithmes d'apprentissage automatique et d'apprentissage par renforcement. Les perspectives d'amélioration sont nombreuses et prometteuses pour l'avenir de la gestion du risque de crédit.

# Bibliographie

- [Atkinson and Mulira, 1993] Atkinson, A. C. and Mulira, H.-M. (1993). The stalactite plot for the detection of multivariate outliers. *Statistics and Computing*, 3(1) :27–35.
- [Baesens et al., 2003] Baesens, B., Setiono, R., Mues, C., and Vanthienen, J. (2003). Using neural network rule extraction and decision tables for credit-risk evaluation. *Management science*, 49(3) :312–329.
- [Bishop, 2006] Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Springer.
- [Blanchard et al., 2019] Blanchard, A., Henrot, P., and Molinet, J.-F. (2019). *Modélisation et simulation numérique appliquées aux sciences et à l’ingénierie*. Dunod.
- [Bubeck et al., 2012] Bubeck, S., Cesa-Bianchi, N., and Bubeck, S. (2012). *Regret Analysis of Stochastic and Nonstochastic Multi-Armed Bandit Problems*. Foundations and Trends® in Machine Learning Series. Now Publishers.
- [Crook et al., 2007] Crook, J. N., Edelman, D. E., and Thomas, L. C. (2007). Recent developments in consumer credit risk assessment. *Eur. J. Oper. Res.*, 183 :1447–1465.
- [Desai et al., 1996] Desai, V. S., Crook, J. N., and Overstreet, G. A. (1996). A comparison of neural networks and linear scoring models in the credit union environment. *European Journal of Operational Research*, 95 :24–37.
- [El Sanharawi and Naudet, 2013] El Sanharawi, M. and Naudet, F. (2013). Comprendre la régression logistique. *Journal Français d’Ophtalmologie*, 36(8) :710–715.
- [Hubert and Elisha, 2023] Hubert, K. and Elisha, B. (2023). Normalization and standardization : Methods to preprocess data to have consistent scales and distributions.
- [Kraus, 2014] Kraus, A. (2014). *Recent methods from statistics and machine learning for credit scoring*. Cuvillier Verlag.
- [Lafortune et al., 2004] Lafortune, Y., de la méthodologie, S. C. D., and des méthodes d’enquêtes sociales, S. C. D. (2004). *Analyse diagnostique des modèles de régression logistique appliqués à des données d’enquête*. Statistique Canada.
- [Lattimore and Szepesvári, 2020] Lattimore, T. and Szepesvári, C. (2020). *Bandit Algorithms*. Cambridge University Press.
- [Lee et al., 2002] Lee, T.-S., Chiu, C.-C., Lu, C.-J., and Chen, I.-F. (2002). Credit scoring using the hybrid neural discriminant technique. *Expert Syst. Appl.*, 23 :245–254.
- [Norman and Gaston, 2020] Norman, A. and Gaston, N. (2020). *L’apprentissage automatique en action : Guide pour le profane, guide d’apprentissage progressif pour débutants (apprentissage automatique)*. Tektime.
- [Siddiqi, 2006] Siddiqi, N. (2006). *Credit Risk Scorecards : Developing and Implementing Intelligent Credit Scoring*. John Wiley & Sons.



- [Sutton and Barto, 2018] Sutton, R. and Barto, A. (2018). *Reinforcement Learning, second edition : An Introduction*. Adaptive Computation and Machine Learning series. MIT Press.
- [Thomas, 2000] Thomas, L. C. (2000). A survey of credit and behavioural scoring : forecasting financial risk of lending to consumers. *International Journal of Forecasting*, 16 :149–172.
- [Tran-anh et al., 2010] Tran-anh, L., Chapman, A., Munoz De Cote Flores Luna, J. E., Rogers, A., and Jennings, N. R. (2010). Epsilon–first policies for budget–limited multi-armed bandits. *Journal of Artificial Intelligence Research*, 38 :101–142.
- [Vale et al., 2022] Vale, Z., Pinto, T., Negnevitsky, M., and Venayagamoorthy, G. (2022). *Intelligent Data Mining and Analysis in Power and Energy Systems : Models and Applications for Smarter Efficient Power Systems*. IEEE Press Series on Power and Energy Systems. Wiley.
- [Wang, 2005] Wang, L. (2005). *Support Vector Machines : Theory and Applications*. Studies in Fuzziness and Soft Computing. Springer Berlin Heidelberg.
- [West, 2000] West, D. (2000). Neural network credit scoring models. *Comput. Oper. Res.*, 27 :1131–1152.
- [Zheng, 2015] Zheng, A. (2015). *Evaluating Machine Learning Models : A Beginner’s Guide to Key Concepts and Pitfalls*. O’Reilly Media.

# Annexe A

## Mesures Calculées dans la Méthode Actuelle d'Echantillonnage

TABLE A.1: Formules utilisées dans l'échantillonnage aléatoire

Nom du champ	Description	Formule
NumberInterval	Nombre d'éléments par intervalle	$\sum_{i=0}^{\text{From}} (i + 1)$
Average	Moyenne des montants par intervalle	$\frac{\sum_{\text{to}}^{\text{From}} \text{Amount}}{\text{NumberInterval}}$
TotalAmount	Montants totaux par intervalle	$\sum_{\text{to}}^{\text{From}} \text{Amount}$
NotPaid	Montants impayés par intervalle	$\sum_{\text{to}}^{\text{From}} (\text{paid ?} \times \text{Amount})$
NPratio	Pourcentage du montant impayé sur le montant total	$\frac{\text{NotPaid}}{\text{NumberInterval}}$
CIfrom	Limite inférieure pour l'intervalle de confiance à 99%	$NPratio - (2.58 \times (NPratio \times (1 - NPratio))) / \text{Sample Required}$
Cto	Limite supérieure pour l'intervalle de confiance à 99%	$NPratio + (2.58 \times \sqrt{\frac{NPratio \times (1 - NPratio)}{\text{Sample Required}}})$
SampleSize	Détermination de la taille de l'échantillon nécessaire	$\frac{(2.58^2 \times NPratio \times (1 - NPratio))}{\text{MOE}^2}$
SampleRequired	Taille minimale requise de l'échantillon	$\frac{\text{SampleSize}}{1 + \frac{\text{SampleSize} - 1}{\text{NumberInterval}}}$
BuildingsNotInSample	Bâtiments hors échantillon	$\frac{\text{NumberInterval}}{\text{SampleRequired}} -$
RiskBasedOnMOE	Premier calcul du risque basé sur la marge d'erreur	$\text{MOE} \times \text{Average} \times \text{BuildingsNotInSample}$
PercentageRisk	Pourcentage de risque selon la première formule	$\frac{\text{RiskBasedOnMOE}}{\text{TotalAmount}}$

Formules utilisées dans l'échantillonnage aléatoire (suite)

Nom du champ	Description	Formule
NotPaidAmount	Montant total impayé	$\text{NotPaid} \times \text{Average}$
RiskBasedOnNPratio	Deuxième calcul du risque basé sur le ratio impayé	$\text{BuildingsNotInSample} \times \text{Average} \times \text{NPratio}$
RiskBasedOnNP	Pourcentage de risque selon la deuxième formule	$\frac{\text{RiskBasedOnNPratio}}{\text{TotalAmount}}$
STPAMOUNT	Montant pris comme échantillon par intervalle	$\text{SampleRequired} \times \text{Average}$
IntervalDistribution	Distribution en pourcentage des montants par intervalle	$\frac{\text{NumberInterval}}{\sum \text{NumberInterval}}$
SamplePercentage	Taux d'échantillonnage par rapport au total	$\frac{\text{SampleRequired}}{\text{NumberInterval}}$

# Annexe B

## Description des Variables

Nom de la variable	Description	Type
person_age	Age	Entier
person_income	Salaire annuel (en euros)	Réel
person_home_ownership	Statut de propriété du domicile	Catégorielle (Object)
person_emp_length	Durée d'emploi (en années)	Entier
loan_grade	Grade du crédit	Catégorielle (Object)
loan_amnt	Montant du prêt (en euros)	Réel (ne dépasse pas 5000 euros)
loan_int_rate	Taux d'intérêt	Réel
loan_status	État du prêt (0 : non défaut, 1 : défaut)	Entier (Binaire)
cb_person_default_on_file	Le client a fait défaut précédemment ou non	Binaire (Object)
cb_person_cred_hist_length	Longueur de l'historique des crédits	Entier

TABLE B.1 – Description des variables

## Annexe C

### Visualisation des Variables

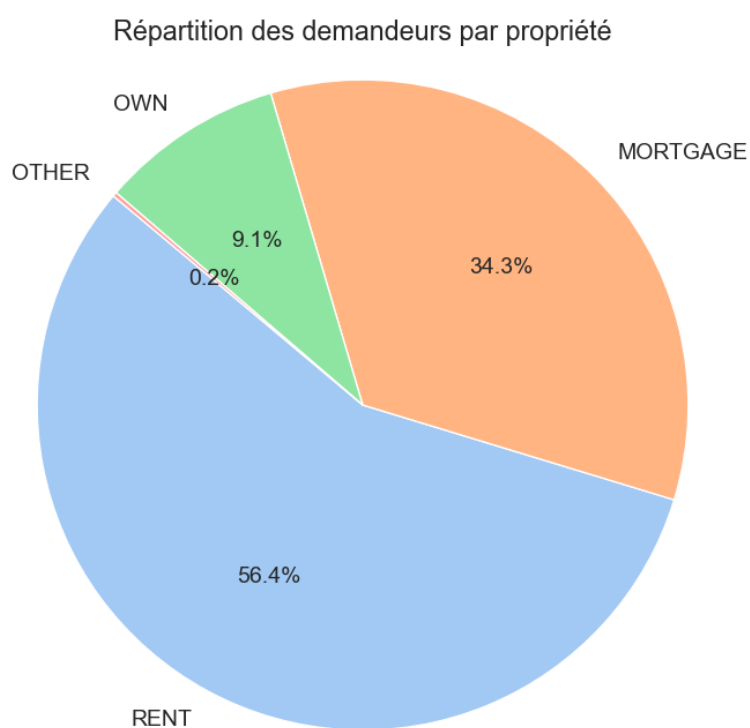


FIGURE C.1 – Performance du modèle Random Forest dans la prédiction du taux d'intérêt du prêt

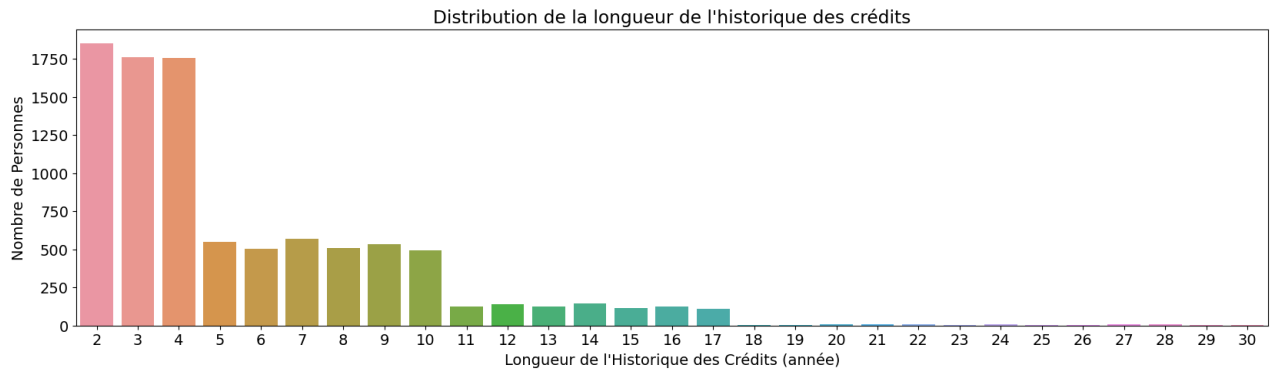


FIGURE C.2 – Distribution des demandeurs selon la longueur de l'historique des crédits

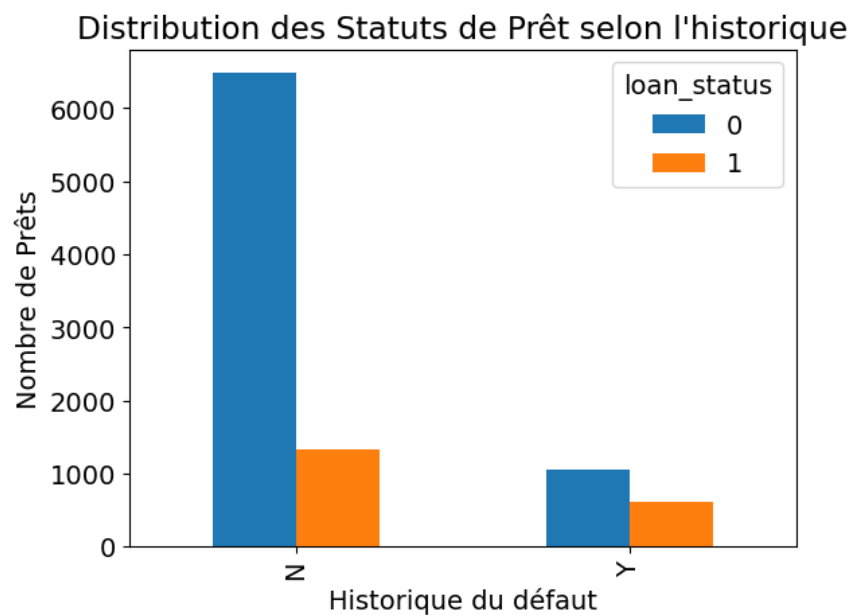


FIGURE C.3 – Distribution des Statuts de Prêts Selon l'Histoire

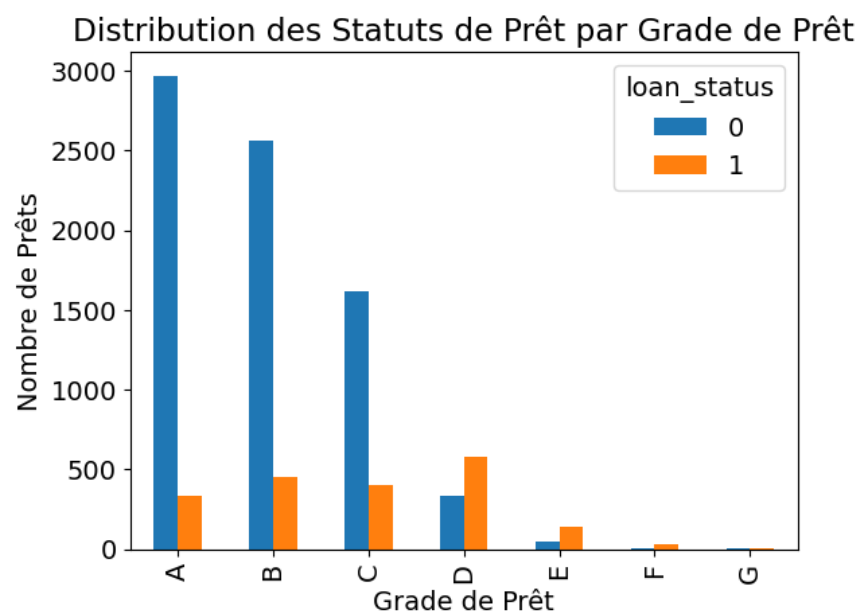


FIGURE C.4 – Distribution des Statuts de Prêts par Grade

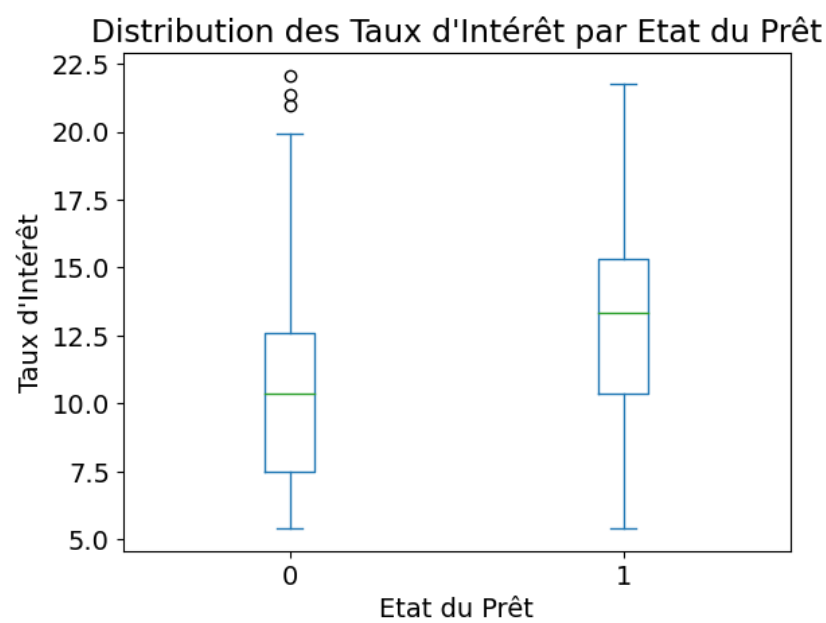


FIGURE C.5 – Distribution des Taux d'Intérêt par Etat du Prêt

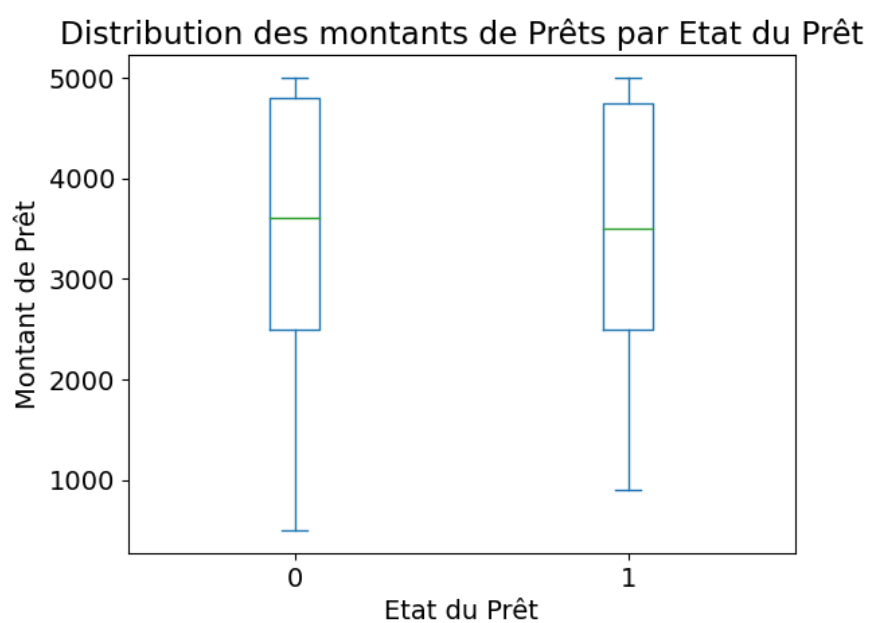


FIGURE C.6 – Distribution des Montants de Prêts par Etat du Prêt



# Annexe D

## Modèles de Régression

### 1. Régression linéaire

La régression linéaire est une technique statistique fondamentale utilisée pour modéliser la relation entre une variable dépendante et une ou plusieurs variables indépendantes [Bishop, 2006].

#### Modèle de Régression Linéaire Simple

Dans le modèle de régression linéaire simple, la relation entre une variable dépendante  $Y$  et une variable indépendante  $X$  est modélisée par l'équation :

$$Y = \beta_0 + \beta_1 X + \varepsilon \quad (1)$$

où :

- $Y$  est la variable dépendante,
- $X$  est la variable indépendante,
- $\beta_0$  est l'ordonnée à l'origine (constante),
- $\beta_1$  est la pente de la ligne de régression,
- $\varepsilon$  est l'erreur aléatoire.

L'objectif principal de la régression linéaire simple est d'estimer les coefficients  $\beta_0$  et  $\beta_1$  qui minimisent la somme des carrés des résidus :

$$\text{Min} \sum_{i=1}^n \varepsilon_i^2 \quad (2)$$

où  $\varepsilon_i$  est le résidu pour l'observation  $i$ .

#### Modèle de Régression Linéaire Multiple

Lorsqu'il y a plusieurs variables indépendantes, le modèle de régression linéaire est étendu comme suit [Bishop, 2006] :

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \dots + \beta_p X_p + \varepsilon \quad (3)$$

où :

- $X_1, X_2, \dots, X_p$  sont les variables indépendantes,
- $\beta_0, \beta_1, \beta_2, \dots, \beta_p$  sont les coefficients associés à chaque variable indépendante.

L'objectif est de minimiser la somme des carrés des résidus :

$$\text{Min} \sum_{i=1}^n \varepsilon_i^2 \quad (4)$$

### Évaluation du Modèle

Après avoir ajusté le modèle, il est essentiel d'évaluer sa qualité. Les métriques couramment utilisées pour évaluer la performance d'un modèle de régression linéaire incluent :

— Coefficient de Détermination  $R^2$  :

$$R^2 = 1 - \frac{\sum_{i=1}^n (Y_i - \hat{Y}_i)^2}{\sum_{i=1}^n (Y_i - \bar{Y})^2} \quad (5)$$

— Erreur Quadratique Moyenne (MSE) :

$$MSE = \frac{1}{n} \sum_{i=1}^n (Y_i - \hat{Y}_i)^2 \quad (6)$$

## 2. Régression Ridge

La régression Ridge, également connue sous le nom de régression de crête, est une méthode de régression linéaire régularisée utilisée pour analyser des données multicollinéaires — c'est-à-dire des données avec des variables explicatives hautement corrélées. Cette méthode est particulièrement efficace pour remédier au problème de multicollinéarité qui peut rendre les estimations des moindres carrés ordinaires (MCO) non seulement inefficaces mais aussi biaisées[Bishop, 2006].

### Formulation du Modèle

La régression Ridge modifie l'estimation des MCO en imposant une pénalité sur la taille des coefficients. Cette pénalité réduit la variance des estimateurs et améliore leur conditionnement[Bishop, 2006].

**Modèle :**

$$Y = X\beta + \epsilon \quad (7)$$

où  $\epsilon$  est le terme d'erreur, normalement distribué avec une moyenne de zéro et une variance de  $\sigma^2$ ,  $\epsilon \sim N(0, \sigma^2 I)$ .

### Critère de Minimisation

Le critère de minimisation de la régression Ridge est donné par la somme des carrés des résidus avec un terme de pénalité ajouté qui augmente avec le carré de la norme euclidienne des coefficients[Bishop, 2006].

$$\text{Minimiser} \quad \|Y - X\beta\|^2 + \lambda \|\beta\|^2 \quad (8)$$

où :

- $\|Y - X\beta\|^2$  est la somme des carrés des résidus,
- $\lambda$  est le paramètre de régularisation qui contrôle le compromis entre ajustement et complexité du modèle en pénalisant les valeurs élevées des coefficients de régression[Bishop, 2006],
- $\|\beta\|^2$  est la norme euclidienne des coefficients, également connue sous le nom de norme L2.

## Solution Analytique

La solution à ce problème de minimisation peut être exprimée de façon analytique :

$$\hat{\beta}_{\text{ridge}} = (X^T X + \lambda I)^{-1} X^T Y \quad (9)$$

Cette expression montre que chaque coefficient est réduit par rapport à l'estimation MCO, ce qui aide à réduire la variance et à prévenir le surajustement [Bishop, 2006].

## Avantages de la Régression Ridge

- Elle réduit la variance des estimateurs et améliore leur conditionnement.
- Elle permet d'obtenir des solutions où l'estimation MCO serait impossible (par exemple, quand  $p > n$  ou lorsque  $X^T X$  est singulière).
- La pénalité L2 encourage la réduction de tous les coefficients, ce qui mène à un modèle plus simple et moins susceptible aux perturbations dans les données d'entrée [Bishop, 2006].

## Évaluation du Modèle

L'évaluation des modèles Ridge peut utiliser des métriques standard d'évaluation de modèle, telles que le R-squared, l'erreur quadratique moyenne (MSE) ou encore via des techniques de validation croisée pour déterminer le meilleur  $\lambda$  [Bishop, 2006].

## 3. Régression Lasso

La régression Lasso (Least Absolute Shrinkage and Selection Operator) est une méthode de régression linéaire qui implique la régularisation L1. La régularisation L1 ajoute une pénalité équivalente à la valeur absolue de la somme des coefficients de régression. Cette technique est particulièrement efficace pour améliorer la précision et la simplicité du modèle en réduisant le nombre de termes de variables explicatives [Bishop, 2006].

## Formulation du Modèle

La régression Lasso est formellement définie par l'équation :

$$Y = X\beta + \epsilon \quad (10)$$

où  $\epsilon$  est le terme d'erreur, normalement distribué avec une moyenne de zéro et une variance de  $\sigma^2$ ,  $\epsilon \sim N(0, \sigma^2 I)$ .

## Fonction Objectif

La fonction objectif de la régression Lasso est de minimiser la somme des carrés des résidus, avec une pénalité pour la somme des valeurs absolues des coefficients du modèle [Bishop, 2006] :

$$\text{Minimiser } \|Y - X\beta\|^2 + \lambda \|\beta\|_1 \quad (11)$$

où  $\|\beta\|_1$  est la norme L1 des coefficients, et  $\lambda$  est un paramètre de régularisation qui contrôle le degré de régularité.

## Propriétés de Sélection de Variable

L'un des avantages majeurs de la régression Lasso par rapport à la régression Ridge est sa capacité à réaliser une sélection de variable automatique en raison de la nature de la pénalité L1. Les coefficients de certaines variables peuvent être réduits exactement à zéro, ce qui signifie que la méthode peut exclure des variables non pertinentes du modèle final [Bishop, 2006].

## Solution Numérique

Contrairement à la régression Ridge, il n'existe pas de solution fermée pour la régression Lasso en raison de la non-différentiabilité de la norme L1 au zéro. Les solutions sont généralement obtenues par des algorithmes de descente de coordonnées ou par des méthodes de gradient proximal [Bishop, 2006].

## Avantages de la Régression Lasso

- Elle permet de réduire la complexité du modèle en réduisant le nombre de variables, ce qui est bénéfique pour l'interprétation du modèle.
- Elle peut être utilisée comme méthode de sélection de variable intégrée, en raison de sa capacité à réduire certains coefficients à zéro.
- Elle est particulièrement utile quand on suspecte que plusieurs variables explicatives sont corrélées ou lorsqu'il y a plus de variables que d'observations.

## Évaluation du Modèle

L'évaluation de la performance du modèle Lasso peut inclure l'utilisation du R-carré ajusté, de l'erreur quadratique moyenne (MSE), ou des approches de validation croisée pour déterminer la valeur optimale de  $\lambda$ .

## 4. Forêts aléatoires

Comme son nom l'indique, une forêt aléatoire (Random Forest) est un ensemble d'arbres basé sur une collection de variables aléatoires. Formellement, soit un vecteur aléatoire à  $p$  dimensions  $\mathbf{X} = (X_1, \dots, X_p)^T$  représentant les variables d'entrée ou prédicteurs réelles et une variable aléatoire  $Y$  représentant la réponse réelle. Nous supposons une distribution conjointe inconnue  $P_{XY}(X, Y)$ . L'objectif est de trouver une fonction de prédiction  $f(\mathbf{X})$  pour prédire  $Y$ . Cette fonction de prédiction est déterminée par une fonction de perte  $L(Y, f(\mathbf{X}))$  et est définie pour minimiser la valeur attendue de la perte

$$\mathbb{E}_{XY}[L(Y, f(\mathbf{X}))] \quad (12)$$

où les indices indiquent l'espérance par rapport à la distribution conjointe de  $X$  et  $Y$  [Bishop, 2006]. Intuitivement,  $L(Y, f(\mathbf{X}))$  mesure la proximité de  $f(\mathbf{X})$  à  $Y$  ; elle pénalise les valeurs de  $f(\mathbf{X})$  qui s'éloignent beaucoup de  $Y$ . Les choix typiques de  $L$  sont la fonction de perte quadratique

$$L(Y, f(\mathbf{X})) = (Y - f(\mathbf{X}))^2 \quad (13)$$

pour la régression et la fonction de perte zéro-un pour la classification :

$$L(Y, f(\mathbf{X})) = \begin{cases} 0 & \text{si } Y = f(\mathbf{X}) \\ 1 & \text{sinon} \end{cases}$$

Il s'avère que minimiser  $\mathbb{E}_{XY}[L(Y, f(\mathbf{X}))]$  pour la fonction de perte quadratique donne l'espérance conditionnelle

$$f(x) = \mathbb{E}(Y|X = x) \quad (14)$$

également connue sous le nom de fonction de régression. Dans la situation de classification, si l'ensemble des valeurs possibles de  $Y$  est noté  $\mathcal{Y}$ , minimiser  $\mathbb{E}_{XY}[L(Y, f(\mathbf{X}))]$  pour la fonction de perte zéro-un donne

$$f(x) = \arg \max_{y \in \mathcal{Y}} P(Y = y|X = x) \quad (15)$$

également connue sous le nom de règle de Bayes [Bishop, 2006].

Les ensembles sont construits à partir d'une collection de ce qu'on appelle des "apprenants de base",  $h_1(x), \dots, h_J(x)$ , et ces apprenants de base sont combinés pour donner le "prédicteur de l'ensemble"  $f(x)$ . En régression, les apprenants de base sont moyennés comme suit :

$$f(x) = \frac{1}{J} \sum_{j=1}^J h_j(x) \quad (16)$$

tandis qu'en classification,  $f(x)$  est la classe prédite le plus fréquemment ("vote") :

$$f(x) = \operatorname{argmax}_{y \in Y} \sum_{j=1}^J I(y = h_j(x)) \quad (17)$$

Dans les forêts aléatoires (Random Forests), le  $j$ -ième apprenant de base est un arbre noté  $h_j(X, \Theta_j)$ , où  $\Theta_j$  est une collection de variables aléatoires et les  $\Theta_j$  sont indépendantes pour  $j = 1, \dots, J$ . Bien que la définition d'une forêt aléatoire soit très générale, elles sont presque toujours mises en œuvre de la manière spécifique décrite dans la section 2.2. Pour comprendre l'algorithme des forêts aléatoires, il est important d'avoir une connaissance fondamentale du type d'arbres utilisés comme apprenants de base[?].

## 5. Bayesian Ridge

La régression Bayesian Ridge est une extension bayésienne du modèle de régression ridge classique. Ce modèle est particulièrement utile pour les problèmes où le nombre de variables explicatives est potentiellement plus grand que le nombre d'observations, ou lorsque les variables sont fortement corrélées [Bishop, 2006].

### Principes de Base

Dans la régression Bayesian Ridge, la relation entre la variable dépendante  $Y$  et les variables indépendantes  $X$  est modélisée avec une approche probabiliste, contrairement à l'approche purement optimisatrice de la régression linéaire classique [Bishop, 2006].

**Modèle :**

$$Y = X\beta + \epsilon \quad (18)$$

où  $\epsilon$  est le terme d'erreur, normalement distribué avec une moyenne de zéro et une variance de  $\sigma^2$ ,  $\epsilon \sim N(0, \sigma^2 I)$ .

## Prémisses Bayésiennes

Les coefficients  $\beta$  sont considérés comme des variables aléatoires avec une distribution a priori normale :

$$\beta \sim N(0, \lambda^{-1}I) \quad (19)$$

où  $\lambda$  est un paramètre qui contrôle la distribution des coefficients et agit comme un paramètre de régularisation dans la régression ridge classique.

## Estimation des Coefficients

L'estimation bayésienne des coefficients se fait en calculant la distribution a posteriori des coefficients  $\beta$  en utilisant le théorème de Bayes. La solution est obtenue en maximisant cette distribution a posteriori, ce qui est équivalent à minimiser une fonction de perte qui inclut un terme de pénalité L2, similaire à la régression ridge[Bishop, 2006].

### Distribution a Posteriori :

$$p(\beta|X, Y) \propto p(Y|X, \beta) \times p(\beta) \quad (20)$$

où  $p(Y|X, \beta)$  est la vraisemblance des données sous le modèle linéaire, et  $p(\beta)$  est la priorité sur les coefficients.

## Avantages du Modèle Bayesian Ridge

- Il offre une estimation robuste des coefficients en présence de multicollinéarité.
- Il intègre les incertitudes dans les estimations des coefficients.
- Il peut automatiquement régler la complexité du modèle via le paramètre de régularisation  $\lambda$ , qui peut être estimé à partir des données.

## Évaluation du Modèle

L'évaluation du modèle Bayesian Ridge peut se faire via des critères de sélection de modèle bayésien, comme le critère d'information de Deviance (DIC) ou des méthodes de validation croisée[Bishop, 2006].

## 6. Gradient Boosting

Le Gradient Boosting est une technique de machine learning pour des problèmes de régression et de classification, qui produit un modèle prédictif sous forme d'un ensemble de prédicteurs faibles, typiquement des arbres de décision. Cette méthode construit le modèle de manière séquentielle et généralise les méthodes de boosting en permettant l'optimisation de fonctions de perte différentiables arbitraires[Bishop, 2006].

### Principes du Gradient Boosting

Le Gradient Boosting fonctionne en ajoutant séquentiellement des prédicteurs au modèle, chaque nouveau prédicteur corrigeant ses prédécesseurs. Cette méthode combine de nombreux modèles faibles pour créer un modèle robuste[Bishop, 2006].

#### Processus d'itération :

1. Nous commençons avec un modèle initial, souvent le prédicteur constant qui minimise la fonction de perte (par exemple, la moyenne des cibles pour la régression).

2. À chaque étape, nous introduisons un nouvel arbre de décision qui prédit le gradient négatif de la fonction de perte par rapport à la prédiction actuelle.
3. Les paramètres de l'arbre sont ajustés pour minimiser la fonction de perte, en utilisant généralement une technique de descente de gradient.

## Fonction de Perte

La fonction de perte spécifique utilisée dépend de la tâche de prédiction (régression, classification, etc.). Pour la régression, une fonction de perte courante est l'erreur quadratique moyenne (MSE). Pour la classification, la fonction de perte logistique peut être employée[Bishop, 2006].

**Formule Générale de la Mise à Jour :**

$$F_m(x) = F_{m-1}(x) + \nu \sum_{i=1}^N \gamma_i h_i(x) \quad (21)$$

où  $F_{m-1}(x)$  est le modèle à l'étape  $m - 1$ ,  $h_i(x)$  sont les arbres ajoutés à l'étape  $m$ ,  $\gamma_i$  sont les taux d'apprentissage, et  $\nu$  est le taux de rétrécissement.

## Importance du Taux de Rétrécissement

Le taux de rétrécissement  $\nu$  réduit la contribution de chaque arbre pour rendre le processus plus robuste, empêchant ainsi le surajustement, ce qui est crucial pour maintenir la généralisation du modèle hors de l'échantillon d'apprentissage[Bishop, 2006].

## Avantages du Gradient Boosting

- Haute précision prédictive.
- Flexibilité pour optimiser différentes fonctions de perte.
- Capacité à gérer les données hétérogènes (caractéristiques continues et catégorielles).

## Évaluation du Modèle

Les modèles Gradient Boosting sont généralement évalués à l'aide de la validation croisée pour tester leur performance sur des données non vues. Des métriques comme l'erreur quadratique moyenne pour la régression ou l'erreur de classification pour la classification sont couramment utilisées[Bishop, 2006].

# Annexe E

## Algorithmes de Bandit Multi-Bras

### E.1 Stratégie Epsilon-Greedy

---

**Algorithm 1** Algorithme Epsilon-Greedy pour la sélection des crédits à risque

---

**Entrée :** Données des demandeurs de crédits

**Sortie :** Sélection optimisée des crédits à vérifier manuellement

Initialiser  $\epsilon \leftarrow 0.1$

Initialiser le nombre total de prêts  $\text{total\_loans} \leftarrow \text{len}(\text{data})$

Initialiser  $Q \leftarrow [0, 0, 0, 0, 0]$   $\triangleright$  estimation des récompenses pour chaque intervalle

Initialiser  $N \leftarrow [0, 0, 0, 0, 0]$   $\triangleright$  nombre de sélections de chaque intervalle

Initialiser  $\text{rewards} \leftarrow []$   $\triangleright$  pour stocker les récompenses pour le traçage

Initialiser  $\text{risky\_counts} \leftarrow [0, 0, 0, 0, 0]$   $\triangleright$  compte des prêts risqués sélectionnés

Initialiser  $\text{selected\_amounts} \leftarrow [0, 0, 0, 0, 0]$   $\triangleright$  montants sélectionnés par intervalle

Initialiser  $\text{risky\_selected\_amounts} \leftarrow [0, 0, 0, 0, 0]$   $\triangleright$  montants risqués sélectionnés

**for**  $i \leftarrow 0$  **to** 4 **do**

**if**  $\text{len}(\text{bandits}[i]) > 0$  **then**

$(\text{reward}, \text{status}, \text{loan\_amnt}) \leftarrow \text{get\_reward}(\text{bandits}[i], \text{data})$

$Q[i] \leftarrow \text{reward}$

$N[i] \leftarrow N[i] + 1$

$\text{selected\_amounts}[i] \leftarrow \text{selected\_amounts}[i] + \text{loan\_amnt}$

**if**  $\text{status} == 1$  **then**

$\text{risky\_selected\_amounts}[i] \leftarrow \text{risky\_selected\_amounts}[i] + \text{loan\_amnt}$

**end if**

$\text{rewards.append}(\text{reward})$

**end if**

**end for**

$\text{total\_selected\_loans} \leftarrow \sum_{i=0}^4 N[i]$

$\text{total\_risky\_selected} \leftarrow 0$

---



---

```

while total_selected_loans < total_loans × 0.6 do
  if random() <  $\epsilon$  then
     $j \leftarrow \text{random.randint}(0, 4)$  ▷ Sélection aléatoire
  else
     $j \leftarrow \arg \max(Q)$  ▷ Sélection de l'intervalle le plus prometteur
  end if
  if len(bandits[j]) > 0 then
    (reward, status, loan_amnt)  $\leftarrow$  get_reward(bandits[j], data)
    total_risky_selected  $\leftarrow$  total_risky_selected + (status == 1)
    risky_counts[j]  $\leftarrow$  risky_counts[j] + (status == 1)
     $N[j] \leftarrow N[j] + 1$ 
     $Q[j] \leftarrow Q[j] + (\text{reward} - Q[j])/N[j]$ 
    total_selected_loans  $\leftarrow$  total_selected_loans + 1
    selected_amounts[j]  $\leftarrow$  selected_amounts[j] + loan_amnt
    if status == 1 then
      risky_selected_amounts[j]  $\leftarrow$  risky_selected_amounts[j] + loan_amnt
    end if
    rewards.append(reward)
  end if
end while

```

---

## E.2 Stratégie Upper Confidence Bound UCB1

---

**Algorithm 2** Algorithme UCB1 pour la sélection des crédits à risque

---

**Entrée :** Données des demandeurs de crédits, bandits (ensemble des prêts par intervalle)

**Sortie :** Sélection optimisée des crédits à vérifier manuellement

Définir la constante  $c \leftarrow 2$

Initialiser  $Q \leftarrow [0, 0, 0, 0, 0]$

▷ Estimations des récompenses

Initialiser  $N \leftarrow [0, 0, 0, 0, 0]$

▷ Nombre de sélections de chaque intervalle

Initialiser  $\text{rewards} \leftarrow []$

▷ Pour stocker les récompenses

Initialiser  $\text{risky\_counts} \leftarrow [0, 0, 0, 0, 0]$

▷ Compte des prêts risqués sélectionnés

Initialiser  $\text{selected\_amounts} \leftarrow [0, 0, 0, 0, 0]$

▷ Montants sélectionnés par intervalle

Initialiser  $\text{risky\_selected\_amounts} \leftarrow [0, 0, 0, 0, 0]$

▷ Montants risqués sélectionnés

Calculer le nombre total de prêts  $\text{total\_loans} \leftarrow \text{len}(\text{data})$

Initialiser  $\text{total\_selected\_loans} \leftarrow 0$

Initialiser  $\text{total\_risky\_selected} \leftarrow 0$

---

---

```

for  $i \leftarrow 0$  à 4 do
  if  $\text{len}(\text{bandits}[i]) > 0$  then
    ( $\text{reward}, \text{status}, \text{loan\_amnt}$ )  $\leftarrow \text{get\_reward}(\text{bandits}[i], \text{data})$ 
     $Q[i] \leftarrow \text{reward}$ 
     $N[i] \leftarrow N[i] + 1$ 
     $\text{selected\_amounts}[i] \leftarrow \text{selected\_amounts}[i] + \text{loan\_amnt}$ 
    if  $\text{status} == 1$  then
       $\text{risky\_selected\_amounts}[i] \leftarrow \text{risky\_selected\_amounts}[i] + \text{loan\_amnt}$ 
    end if
     $\text{rewards.append}(\text{reward})$ 
  end if
end for
while  $\text{total\_selected\_loans} < \text{total\_loans} \times 0.6$  do
  Calculer les valeurs UCB pour chaque intervalle
   $\text{ucb\_values} \leftarrow Q + c \times \sqrt{\frac{\log(\text{total\_selected\_loans}+1)}{N+1e-9}}$ 
  for  $i \leftarrow 0$  à 4 do
    if  $\text{len}(\text{bandits}[i]) == 0$  then
       $\text{ucb\_values}[i] \leftarrow -\infty$ 
    end if
  end for
   $j \leftarrow \text{argmax}(\text{ucb\_values})$ 
  if  $\text{len}(\text{bandits}[j]) == 0$  then
    continuer
  end if
  ( $\text{reward}, \text{status}, \text{loan\_amnt}$ )  $\leftarrow \text{get\_reward}(\text{bandits}[j], \text{data})$ 
   $\text{total\_risky\_selected} \leftarrow \text{total\_risky\_selected} + (\text{status} == 1)$ 
   $\text{risky\_counts}[j] \leftarrow \text{risky\_counts}[j] + (\text{status} == 1)$ 
   $N[j] \leftarrow N[j] + 1$ 
   $Q[j] \leftarrow Q[j] + (\text{reward} - Q[j])/N[j]$ 
   $\text{total\_selected\_loans} \leftarrow \text{total\_selected\_loans} + 1$ 
   $\text{selected\_amounts}[j] \leftarrow \text{selected\_amounts}[j] + \text{loan\_amnt}$ 
  if  $\text{status} == 1$  then
     $\text{risky\_selected\_amounts}[j] \leftarrow \text{risky\_selected\_amounts}[j] + \text{loan\_amnt}$ 
  end if
   $\text{rewards.append}(\text{reward})$ 
end while

```

---