Projet Fin d'Etudes

Assurance / Data Science / Mathématiques Actuarielles

Intelligence artificielle appliquée à l'assurance emprunteur décès

De la tarification actuarielle à la prédiction Machine Learning d'une prime mensuelle nivelée

Adel Kechid – Ethan Ada – Valentin Beaufils

Année académique : 2025–2026

Mots-clés : assurance emprunteur, temporaire décès, capital restant dû, table de mortalité, valeur actuelle espérée, régression, Random Forest, validation croisée, learning curve, Streamlit.

Livrables : moteur actuariel, génération de données, modèle ML, application Streamlit (actuariat / IA / R&D), dépôt de code.

Résumé

Ce mémoire présente un cadre complet de tarification d'une assurance emprunteur décès indexée sur le capital restant dû (CRD), puis l'approximation de cette tarification par un modèle de Machine Learning afin d'obtenir une estimation instantanée de la prime mensuelle. La contribution est double : (i) un moteur actuariel fondé sur les tables de mortalité, l'actualisation et un échéancier d'amortissement ; (ii) un pipeline data/ML appuyé sur une génération de données simulées cohérentes (prêt $\rightarrow$ prime) et une évaluation méthodologique (hold-out, validation croisée, learning curves). L'ensemble est intégré dans une application Streamlit structurée pour un usage opérationnel.

Abstract

This thesis develops a full actuarial pricing framework for a decreasing term life insurance linked to a loan balance (outstanding principal), and then approximates this pricing function using a Machine Learning regression model to enable near-instantaneous premium estimates. We propose (i) an actuarial engine relying on mortality tables, discounting and amortization schedules ; (ii) a data/ML pipeline based on realistic synthetic data generation (loan $\rightarrow$ premium) and a rigorous evaluation protocol (hold-out, cross-validation, learning curves). The end-to-end solution is delivered through a Streamlit application designed for operational use.

Table des matières

Table des figures

Liste des tableaux

Chapitre 1

Introduction générale

1.1 Contexte

L'assurance emprunteur constitue un mécanisme central dans l'octroi de crédit, en particulier immobilier. En présence d'un événement grave comme le décès, elle sécurise le remboursement du prêt en transférant le risque vers l'assureur : la banque réduit le risque de non-recouvrement et les proches de l'emprunteur évitent de supporter une dette résiduelle. D'un point de vue actuariel, ce produit s'apparente à une **temporaire décès** dont la prestation est généralement **indexée sur le capital restant dû (CRD)**.

Historiquement, la tarification repose sur un socle méthodologique robuste : tables de mortalité, probabilités conditionnelles de décès et de survie, actualisation au taux technique, et application du principe d'équivalence pour déterminer une prime pure. Cette approche est explicable et contrôlable, mais peut devenir coûteuse ou peu flexible lorsqu'il faut produire des estimations en temps réel, à grande échelle, et dans des parcours utilisateurs fortement digitalisés.

Parallèlement, la transformation numérique des parcours crédit et l'augmentation du volume de données disponibles (paramètres de prêt, profils, historiques) ouvrent la voie à l'intégration de l'intelligence artificielle : automatisation de calculs, réduction des délais de réponse, et segmentation plus fine. Dans un cadre opérationnel, l'enjeu n'est pas de remplacer l'actuariat, mais de **conjuguer une référence technique explicable** (moteur actuariel) avec une **capacité de prédiction instantanée** (modèle ML), afin de répondre à des contraintes de performance et de production.

1.2 Problématique

Ce mémoire s'inscrit dans cette logique « Actuariat + IA » et vise à répondre à la question suivante :

> *Comment construire une tarification actuarielle de référence d'une assurance emprunteur décès indexée sur le capital restant dû, puis apprendre une approximation Machine Learning de cette tarification afin de fournir une prime mensuelle instantanée, intégrée dans une application structurée pour un usage opérationnel ?*

Deux difficultés structurent la problématique :

— **Cohérence actuarielle** : la prime dépend d'une exposition dynamique (CRD) et d'une mortalité évolutive avec l'âge ; la tarification mensuelle nécessite une granularité adaptée et des conventions (début/fin de mois) cohérentes.

— **Transposition ML** : l'objectif du modèle n'est pas seulement de « bien prédire », mais de fournir une approximation stable, reproductible et intégrable, sans fuite de données, et avec des outils d'évaluation (hold-out, CV, learning curves) compatibles avec un cadre académique et industriel.

1.3 Contributions et livrables

Les contributions du travail sont organisées autour de quatre briques complémentaires :

— **Moteur actuariel** : modélisation du prêt (amortissement, CRD mensuel), conversion de la mortalité annuelle en mortalité mensuelle, calcul de la valeur actuelle espérée des prestations (EPV sinistres) et détermination d'une **prime mensuelle nivelée** par équivalence actuarielle.

— **Génération de données** : construction d'un dataset de contrats simulés cohérents (montant, durée, âge, taux) et calcul de la **cible actuarielle** (prime mensuelle). La génération intègre des règles métier (bornes, corrélations) afin d'éviter des combinaisons irréalistes et de produire un jeu de données exploitable pour l'apprentissage.

— **Modélisation ML** : formulation du problème en régression supervisée et mise en place d'un protocole d'évaluation rigoureux (séparation train/test, validation croisée, courbes d'apprentissage). Une distinction est faite entre un **modèle de production figé** (déployé dans l'application) et un espace **R&D** destiné à comparer et améliorer les approches.

— **Produit applicatif** : développement d'une application Streamlit multi-pages permettant (i) le calcul actuariel de référence, (ii) la prédiction instantanée par le modèle ML de production, (iii) l'analyse R&D (comparaisons, métriques, learning curves), et (iv) une page de comparaison directe Actuariat vs IA pour valider la cohérence sur des entrées identiques.

1.4 Plan du mémoire

Le mémoire est structuré de manière progressive, de la modélisation financière et actuarielle vers l'approximation ML et l'intégration produit :

— Chapitre 2 : modélisation du prêt et du capital restant dû ; construction du tableau d'amortissement et conventions utiles au produit d'assurance.

— Chapitre 3 : tarification actuarielle de la garantie décès indexée sur le CRD ; passage à une granularité mensuelle, EPV des sinistres et prime mensuelle par équivalence.

— Chapitre 4 : génération d'un dataset de simulation (prêt → prime mensuelle) et contrôles qualité (bornes, corrélations, absence de fuite).

— Chapitre 5 : apprentissage supervisé, choix des modèles (production vs R&D), métriques, validation croisée, learning curves et interprétation des résultats.

— Chapitre 6 : architecture logicielle et application Streamlit ; organisation des modules et présentation des pages.

— Chapitre 7 : conclusion, limites et perspectives (calibration réelle, enrichissement des variables, industrialisation).

Chapitre 2

Prêt immobilier : amortissement et capital restant dû

2.1 Cadre général et notations

On considère un prêt amortissable à mensualités constantes, mécanisme standard dans le crédit immobilier.

Les notations suivantes sont adoptées :

— L : capital initial emprunté (principal),

— N : nombre total de mensualités ($N = 12n$ si n est la durée en années),

— R : taux nominal annuel du prêt,

— $r = \frac{R}{12}$: taux mensuel (conversion proportionnelle),

— A : mensualité constante,

— $k \in \{1, \ldots, N\}$: indice de mensualité,

— $\mathrm{CRD}_k^{\text{début}}$: capital restant dû au début du mois k,

— $\mathrm{CRD}_k^{\text{fin}}$: capital restant dû après paiement de la mensualité k.

Dans la pratique bancaire, la mensualité est payée **en fin de mois**. Pour une assurance décès indexée sur le CRD, il est nécessaire de préciser la convention de prestation :

— **Convention début de mois** : décès au mois $k \Rightarrow$ prestation indexée sur $\mathrm{CRD}_k^{\text{début}}$,

— **Convention fin de mois** : décès au mois $k \Rightarrow$ prestation indexée sur $\mathrm{CRD}_k^{\text{fin}}$.

Cette distinction est importante car elle influence légèrement la valeur actuarielle des prestations.

2.2 Détermination de la mensualité constante

2.2.1 Principe financier

La mensualité A est déterminée de sorte que la valeur actuelle des remboursements soit égale au capital initial L.

Sous l'hypothèse de mensualités constantes payées en fin de période, on impose :

$$L = \sum_{k=1}^{N} \frac{A}{(1+r)^k}.$$

Cette somme est une suite géométrique. On obtient alors :

$$L = A\frac{1-(1+r)^{-N}}{r},$$

d'où la formule classique de l'annuité :

$$\boxed{A = L\,\frac{r}{1-(1+r)^{-N}}}$$

2.2.2 Interprétation économique

Cette formule montre que :

— à capital L fixé, la mensualité croît avec le taux r ;

— à taux fixé, une durée N plus longue diminue la mensualité mais augmente le coût total du crédit ;

— la mensualité est composée d'une part d'intérêts (rémunération du prêteur) et d'une part d'amortissement (remboursement du capital).

2.3 Décomposition de la mensualité

Chaque mensualité se décompose en deux composantes :

$$\text{Intérêts}_k = r \cdot \text{CRD}_k^{\text{début}}, \qquad \text{Amortissement}_k = A - \text{Intérêts}_k.$$

Au début du prêt, le capital restant dû est élevé : la part d'intérêts est donc importante. Au fil du temps, le CRD diminue, la part d'intérêts décroît et la part d'amortissement augmente.

Cette structure explique la forme convexe décroissante de la trajectoire du CRD.

2.4 Capital restant dû : dynamique temporelle

2.4.1 Formule récursive

La dynamique du CRD est donnée par :

$$\text{CRD}_1^{\text{début}} = L,$$

$$\text{CRD}_k^{\text{fin}} = \text{CRD}_k^{\text{début}} - \text{Amortissement}_k = \text{CRD}_k^{\text{début}} - (A - r \cdot \text{CRD}_k^{\text{début}}),$$

$$\text{CRD}_{k+1}^{\text{début}} = \text{CRD}_k^{\text{fin}}.$$

Cette écriture est privilégiée en implémentation car elle permet de construire simplement le tableau d'amortissement.

2.4.2 Expression fermée

On peut également écrire une expression analytique :

$$\mathrm{CRD}_k^{\mathrm{fin}} = L(1+r)^k - A\frac{(1+r)^k - 1}{r}.$$

Cette formule permet de vérifier numériquement que :

$$\mathrm{CRD}_N^{\mathrm{fin}} \approx 0,$$

ce qui garantit l'extinction totale du capital à l'échéance.

2.5 Structure temporelle du risque

Du point de vue assurantiel, la trajectoire du CRD est déterminante :

— au début du prêt : exposition maximale (CRD élevé),

— au milieu : décroissance progressive,

— à la fin : exposition résiduelle faible.

Ainsi, le profil de risque d'une assurance décès adossée au prêt n'est pas constant dans le temps : il décroît mécaniquement avec l'amortissement.

2.6 Lien avec l'assurance emprunteur

La prestation versée en cas de décès pendant le mois k est généralement :

$$B_k = \alpha \cdot \mathrm{CRD}_k,$$

où α est la quotité assurée.

On observe donc que :

— le risque assuré est directement proportionnel au capital restant dû,

— la structure du prêt conditionne la structure du risque décès,

— toute variation des paramètres (L, N, R) impacte indirectement la prime d'assurance.

Ce lien direct justifie l'intégration conjointe du moteur d'amortissement et du moteur actuariel dans l'application développée.

2.7 Analyse de sensibilité

À paramètres actuariels fixés, la prime d'assurance est influencée par la structure du prêt :

— **Augmentation de** L : accroît proportionnellement l'exposition.

— **Augmentation de** N : prolonge l'exposition dans le temps.

— **Augmentation de** R : modifie la trajectoire du CRD et donc la distribution temporelle du risque.

Ces interactions expliquent pourquoi les variables du prêt sont incluses comme variables explicatives dans le modèle Machine Learning.

Chapitre 3

Assurance emprunteur décès : tarification actuarielle en prime mensuelle

3.1 Cadre : temporaire décès indexée sur le capital restant dû

L'assurance emprunteur décès est assimilable à une **temporaire décès** dont la prestation est **décroissante**, car indexée sur le capital restant dû (CRD) du prêt. Le contrat verse une indemnité si l'assuré décède avant l'échéance du crédit ; si l'assuré survit jusqu'à la fin du prêt, aucune prestation n'est versée (assurance *risque pur*).

Dans ce mémoire, on étudie une tarification **mensuelle** : l'horizon est discret en mois, la prestation dépend du CRD du mois, et la prime est une prime **nivelée** (constante) payée tant que l'assuré est en vie.

3.2 Notations actuarielle et financières (au pas mensuel)

On considère un assuré âgé de x à la souscription et un contrat de durée N mois.

3.2.1 Temps, actualisation et facteurs financiers

On note :

— i : taux technique annuel,

— j : taux technique mensuel. Dans l'implémentation, on retient une conversion proportionnelle :

$$j = \frac{i}{12}, \qquad v_m = \frac{1}{1+j},$$

où v_m est le facteur d'actualisation mensuel.

— $t \in \{1, \dots, N\}$: mois (le mois t correspond à l'intervalle de temps $(t-1, t]$).

3.2.2 Mortalité mensuelle : décès et survie

Les tables de mortalité sont généralement fournies à pas annuel via q_y : probabilité de décéder entre les âges y et $y+1$.

Conversion annuelle → mensuelle. Sous hypothèse de force de mortalité constante sur l'année, une conversion standard est :

$$q_y^{(m)} = 1 - (1 - q_y)^{1/12}.$$

Cette relation garantit que la probabilité annuelle est cohérente avec 12 périodes mensuelles indépendantes conditionnellement.

Abattement « profil emprunteur ». On applique un abattement $a \in [0, 1]$ sur la mortalité (sélection à l'entrée / population emprunteur) :

$$q_{y,\text{adj}}^{(m)} = (1 - a)\, q_y^{(m)}.$$

Probabilité de survie mensuelle. On définit la probabilité de survivre le mois t (conditionnelle au fait d'être vivant au début du mois t) :

$$p_{x+t-1}^{(m)} = 1 - q_{x+t-1,\text{adj}}^{(m)}.$$

La probabilité de survivre jusqu'au début du mois t (i.e. survivre aux $t-1$ premiers mois) est alors :

$${}_{t-1}p_x^{(m)} = \prod_{u=1}^{t-1} p_{x+u-1}^{(m)}, \qquad {}_0p_x^{(m)} = 1.$$

Probabilité de décès au mois t. La probabilité que le décès survienne pendant le mois t (i.e. $T \in (t-1, t]$) s'écrit :

$$¶(T \in (t-1, t]) = {}_{t-1}p_x^{(m)} \cdot q_{x+t-1,\text{adj}}^{(m)}.$$

Cette forme est centrale : *survivre jusqu'au mois t* puis *décéder pendant le mois t*.

3.3 Prestation : indexation sur le CRD

3.3.1 Quotité et définition de la prestation

On note $\alpha \in [0, 1]$ la quotité assurée. La prestation versée en cas de décès pendant le mois t est :

$$B_t = \alpha \cdot \text{CRD}_t.$$

3.3.2 Conventions début/fin de mois

Selon les pratiques et la modélisation, CRD_t peut être :

— **CRD en début de mois** : $\text{CRD}_t = \text{CRD}_t^{\text{début}}$,

— **CRD en fin de mois** : $\mathrm{CRD}_t = \mathrm{CRD}_t^{\mathrm{fin}}$.

La différence est en général modérée mais non nulle : au début du prêt (CRD élevé), l'écart entre début et fin de mois peut influer davantage sur la valeur actuelle espérée.

3.3.3 Rappel : calcul du CRD (lien direct avec le Chapitre 2)

On rappelle la décomposition de la mensualité A du prêt, au taux mensuel $r = R/12$:

$$A = L\frac{r}{1-(1+r)^{-N}}, \qquad \text{Intérêts}_t = r \cdot \mathrm{CRD}_t^{\text{début}}, \qquad \text{Amortissement}_t = A - \text{Intérêts}_t.$$

Puis :

$$\mathrm{CRD}_t^{\mathrm{fin}} = \mathrm{CRD}_t^{\text{début}} - \text{Amortissement}_t, \qquad \mathrm{CRD}_{t+1}^{\text{début}} = \mathrm{CRD}_t^{\mathrm{fin}}.$$

L'échéancier fournit ainsi la trajectoire $\{\mathrm{CRD}_t\}_{t=1}^{N}$ utilisée dans la tarification décès.

3.4 Valeur actuelle espérée des sinistres (EPV des prestations)

3.4.1 Formule générale (discrète mensuelle)

La valeur actuelle espérée des prestations est :

$$\mathrm{EPV}_{\mathrm{sin}} = \sum_{t=1}^{N} v_m^t \, \P(T \in (t-1, t]) \, B_t.$$

En substituant les expressions précédentes :

$$\mathrm{EPV}_{\mathrm{sin}} = \sum_{t=1}^{N} v_m^t \left({}_{t-1}p_x^{(m)} \cdot q_{x+t-1,\mathrm{adj}}^{(m)}\right) \Big(\alpha \cdot \mathrm{CRD}_t\Big).$$

Donc, explicitement :

$$\boxed{\mathrm{EPV}_{\mathrm{sin}} = \alpha \sum_{t=1}^{N} v_m^t \; {}_{t-1}p_x^{(m)} \; q_{x+t-1,\mathrm{adj}}^{(m)} \; \mathrm{CRD}_t}$$

Cette quantité correspond au **coût actuariel pur** attendu des sinistres, actualisé au taux technique.

3.4.2 Lecture métier de la formule

Chaque terme du mois t combine :

— une **exposition** : CRD_t (capital à couvrir),

— une **probabilité** : ${}_{t-1}p_x^{(m)} q_{x+t-1,\mathrm{adj}}^{(m)}$ (décès au mois t),

— une **actualisation** : v_m^t (valeur actuelle),

— une **quotité** : α (part couverte).

Ainsi, au début du prêt, CRD_t est élevé mais l'âge est plus faible ; à la fin du prêt, CRD_t est plus faible mais l'âge (et donc la mortalité) plus élevé : la prime résulte de cet arbitrage.

3.5 Valeur actuelle espérée des primes

On cherche une prime mensuelle constante P_m payée tant que l'assuré est vivant. Deux conventions sont possibles selon le timing du paiement.

3.5.1 Prime payée en début de mois (retenue)

Si la prime est payée au début de chaque mois (au temps $t-1$), alors le paiement du mois t a lieu si l'assuré est vivant au début du mois t, soit avec probabilité ${}_{t-1}p_x^{(m)}$. La valeur actuelle espérée des primes est :

$$\mathrm{EPV}_{\mathrm{prem}} = \sum_{t=1}^{N} v_m^{t-1}\ {}_{t-1}p_x^{(m)}\ P_m = P_m \sum_{t=1}^{N} v_m^{t-1}\ {}_{t-1}p_x^{(m)}.$$

En réindexant (avec $u = t-1$), on obtient aussi :

$$\mathrm{EPV}_{\mathrm{prem}} = P_m \sum_{u=0}^{N-1} v_m^{u}\ {}_{u}p_x^{(m)}.$$

3.5.2 Prime payée en fin de mois (option)

Si la prime est payée en fin de mois, elle est due si l'assuré est vivant à la fin du mois t, soit avec probabilité ${}_{t}p_x^{(m)}$. Alors :

$$\mathrm{EPV}^{\mathrm{fin}}_{\mathrm{prem}} = P_m \sum_{t=1}^{N} v_m^{t}\ {}_{t}p_x^{(m)}.$$

Dans ce projet, la convention **début de mois** est privilégiée, car elle s'aligne naturellement avec une tarification où le risque court sur le mois après paiement.

3.6 Prime mensuelle nivelée par équivalence actuarielle

Le principe d'équivalence (prime pure) impose :

$$\mathrm{EPV}_{\mathrm{prem}} = \mathrm{EPV}_{\mathrm{sin}}.$$

Sous convention prime début de mois :

$$P_m \sum_{u=0}^{N-1} v_m^{u}\ {}_{u}p_x^{(m)} = \alpha \sum_{t=1}^{N} v_m^{t}\ {}_{t-1}p_x^{(m)}\ q_{x+t-1,\mathrm{adj}}^{(m)}\ \mathrm{CRD}_t.$$

Donc :

$$\boxed{P_m = \frac{\alpha \sum_{t=1}^{N} v_m^{t}\ {}_{t-1}p_x^{(m)}\ q_{x+t-1,\mathrm{adj}}^{(m)}\ \mathrm{CRD}_t}{\sum_{u=0}^{N-1} v_m^{u}\ {}_{u}p_x^{(m)}}}$$

Cette formule est celle implémentée dans le moteur actuariel, et constitue la **cible** des données simulées utilisées pour l'apprentissage supervisé.

3.7 Procédure de calcul reproductible (prête à coder)

1. Fixer le contrat : (L, N, R), et les hypothèses actuarielle : âge x, taux technique i, abattement a, quotité α.
2. Construire l'échéancier de prêt et extraire CRD_t (début ou fin de mois selon convention).
3. Charger q_y (table de mortalité) et calculer $q_y^{(m)} = 1 - (1 - q_y)^{1/12}$.
4. Appliquer l'abattement : $q_{y,\mathrm{adj}}^{(m)} = (1 - a)q_y^{(m)}$; puis $p_y^{(m)} = 1 - q_{y,\mathrm{adj}}^{(m)}$.
5. Construire ${}_tp_x^{(m)} = \prod_{u=1}^{t} p_{x+u-1}^{(m)}$.
6. Calculer $\mathrm{EPV}_{\mathrm{sin}}$ via la somme sur $t = 1, \ldots, N$.
7. Calculer $\mathrm{EPV}_{\mathrm{prem}} = \sum_{u=0}^{N-1} v_m^u {}_up_x^{(m)} P_m$.
8. En déduire P_m par équivalence.

3.8 Commentaires sur la sensibilité (lecture actuarielle)

— **Âge** : la mortalité augmente avec l'âge, donc $q_{x+t-1,\mathrm{adj}}^{(m)}$ augmente avec t. À durée fixée, P_m croît fortement avec x.

— **Durée** : une durée plus longue augmente l'exposition temporelle et fait intervenir des âges plus élevés dans la somme, ce qui amplifie la prime (effet non-linéaire).

— **Montant** : la prestation étant proportionnelle au CRD, la prime est grossièrement proportionnelle à L (toutes choses égales par ailleurs).

— **Taux du prêt** : à L et N fixés, R modifie la trajectoire du CRD (et donc CRD_t), ce qui impacte la prestation attendue.

— **Taux technique** : un i plus élevé accroît l'actualisation (réduit les valeurs actuelles), ce qui tend à réduire $\mathrm{EPV}_{\mathrm{sin}}$ et donc P_m.

— **Abattement** : augmenter a réduit la mortalité ajustée, donc diminue directement la prime.

Chapitre 4

Génération d'un dataset de simulations (prêt → prime mensuelle)

4.1 Objectif

L'objectif est de générer un dataset de taille suffisante pour apprendre la fonction actuarielle f qui associe à un contrat les paramètres d'entrée et une prime mensuelle actuarielle. Le dataset permet l'entraînement et l'évaluation de plusieurs modèles de régression, puis l'intégration d'un modèle de production figé dans l'application.

4.2 Variables et cible

Chaque observation (une ligne) représente un contrat simulé :

— L : capital emprunté,

— N : durée (en mois) ou durée en années,

— x : âge à la souscription,

— R : taux nominal annuel du prêt,

— i : taux technique annuel,

— a : abattement de mortalité,

— P_m : **cible** (prime mensuelle actuarielle).

Remarque : toute variable dérivée de P_m (par exemple $12P_m$) est exclue des features afin d'éviter toute fuite.

4.3 Règles métier et cohérence statistique

Une génération uniforme produit des combinaisons irréalistes. On impose des contraintes et des corrélations inspirées de la pratique (bornes réalistes, dépendances entre âge/durée/taux). Un score latent de risque $S \in [0, 1]$ peut être introduit pour corréler certaines composantes (par

exemple le taux du prêt) à la structure du contrat :

$$S = \sigma(\beta_0 + \beta_1 \tilde{x} + \beta_2 \tilde{N} + \beta_3 \tilde{L} + \varepsilon),$$

puis

$$R = \mathrm{clip}\big(R_0 + c_1 S + c_2(x - x_0) + c_3(N - N_0) + \eta\big).$$

Le rôle de ces règles est d'augmenter le réalisme et la robustesse du modèle ML sur des contrats plausibles.

4.4 Calcul de la cible actuarielle

Pour chaque contrat simulé, la cible P_m est calculée par le moteur actuariel :

— construction de l'échéancier et du CRD,

— conversion mensuelle de la mortalité + abattement,

— calcul de $\mathrm{EPV}_{\mathrm{sin}}$ et de la somme des primes,

— prime mensuelle par équivalence.

On obtient ainsi un dataset synthétique *supervisé* où la cible est cohérente avec les hypothèses actuariales.

4.5 Contrôles qualité

— bornage : x, N, R, i dans des intervalles réalistes ;

— reproductibilité : graine de génération fixée ;

— absence de fuite : features uniquement à partir des entrées contractuelles ;

— inspection : distributions marginales, corrélations, valeurs aberrantes.

Chapitre 5

Prédiction de la prime mensuelle par Machine Learning

5.1 Formulation

La tarification actuarielle définit une fonction déterministe f :

$$f : (L, x, N, R, i, a) \longmapsto P_m.$$

L'objectif est d'apprendre une approximation $\hat{f}$ permettant une prédiction quasi-instantanée de P_m dans un contexte applicatif.

5.2 Choix des modèles : Production vs R&D

5.2.1 Modèle de production figé

Le moteur IA de production embarque un modèle **figé** (Random Forest Regressor), entraîné une fois puis utilisé en prédiction dans l'application. Ce choix est motivé par :

— robustesse sur des non-linéarités et interactions,

— faible coût en inférence,

— intégration simple (modèle `joblib`).

5.2.2 Espace de comparaison en R&D

La page R&D sert à comparer des approches (baseline linéaire, Random Forest, HistGradientBoostingRegressor), à activer ou non la validation croisée, et à analyser l'effet de la taille d'échantillon via des learning curves.

5.3 Features et transformations

Les features principales sont les paramètres contractuels. Des transformations peuvent être ajoutées dans un cadre sans fuite :

— $\log(L)$, $\log(N)$ (effets d'échelle),

— interaction âge–durée : $x \times N$,

— spread : $(R - i)$,

— interaction âge–spread : $x \times (R - i)$.

Important : si la version actuelle de l'application utilise uniquement les 5 variables (x, N, L, R, i), ces features dérivées restent un axe d'amélioration (R&D) et ne modifient pas le modèle de production tant qu'il est figé.

5.4 Protocole d'évaluation

5.4.1 Hold-out

On sépare le dataset en apprentissage et test (ex. 80/20). Le test n'est **jamais** utilisé pour le réglage d'hyperparamètres.

5.4.2 Validation croisée

La validation croisée K-fold est réalisée sur le train (avec $K \in \{3, 4, 5\}$) afin d'estimer la performance moyenne et la variabilité des métriques.

5.4.3 Learning curves

On étudie l'effet de la taille du dataset sur la performance (MAE, R^2) en entraînant un modèle sur plusieurs tailles (via tes CSV `dataset_20k`, `dataset_30k`, ...). Cela permet de diagnostiquer :

— sous-apprentissage (modèle trop simple),

— sur-apprentissage (écart train/test),

— saturation (gain marginal faible avec plus de données).

5.5 Métriques

On reporte :

— **MAE** : $\text{MAE} = \frac{1}{n} \sum_{i=1}^{n} |y_i - \hat{y}_i|$,

— **RMSE** : $\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2}$,

— R^2 : $R^2 = 1 - \frac{\sum (y_i - \hat{y}_i)^2}{\sum (y_i - \bar{y})^2}$.

Ces métriques sont affichées dans l'application (moteur IA pour le modèle de production, et R&D pour les comparaisons).

5.6 Résultats du modèle de production

Le modèle de production retenu (Random Forest Regressor) a été évalué selon un protocole **hold-out** avec séparation train/test. Les métriques considérées sont la MAE (erreur absolue moyenne), le RMSE (racine de l'erreur quadratique moyenne) et le coefficient de détermination R^2.

TABLE 5.1 – Performance du modèle de production (Random Forest) sur la prédiction de la prime mensuelle

Jeu	**MAE**	**RMSE**	R^2
Train	0.4423	0.9472	0.9997
Test	0.8789	1.9128	0.9989

5.6.1 Interprétation des résultats

Sur le jeu de test, la MAE est de 0.8789 (en euros) : en moyenne, l'erreur absolue sur la prime mensuelle est inférieure à 1 €, ce qui est cohérent avec l'objectif d'estimation instantanée en environnement applicatif. Le RMSE de 1.9128 indique que les erreurs extrêmes restent limitées mais sont naturellement davantage pénalisées par cette métrique quadratique.

Le coefficient R^2 est très élevé (0.9989 sur test), ce qui signifie que le modèle explique la quasi-totalité de la variance de la prime mensuelle sur les données de test. L'écart entre train et test (MAE train = 0.4423 vs MAE test = 0.8789) suggère un **surapprentissage modéré**, classique pour une Random Forest, mais sans dégradation majeure de généralisation.

Il est important de souligner que les données utilisées pour l'apprentissage sont **synthétiques** et produites par un moteur actuariel : le modèle apprend donc principalement à **reproduire une fonction déterministe** (tarification actuarielle) sur un domaine de valeurs borné et cohérent. Dans ce contexte, un R^2 proche de 1 est attendu ; l'enjeu principal devient alors la stabilité hors-échantillon et la robustesse lorsque l'on modifie les hypothèses de génération (bornes, mortalité, taux, abattements).

5.7 Discussion : portée et limites

Le modèle apprend principalement à reproduire une tarification actuarielle sur données synthétiques. Cette approche est pertinente pour :

— un prototype d'estimation instantanée,

— une mise en production « accélérée » (remplacer un calcul actuariel coûteux),

— une base de comparaison avant calibration réelle.

En revanche, l'extension à des données réelles requiert : calibration des hypothèses (mortalité, taux, abattements), ajout de variables sanitaires (fumeur, IMC, pathologies), et prise en compte de règles tarifaires commerciales (frais/marge/anti-sélection).

Chapitre 6

Architecture logicielle et application Streamlit

6.1 Objectifs d'architecture

L'application développée dans ce projet vise un double objectif :

— fournir une **référence actuarielle explicable** ;

— proposer une **approximation ML instantanée** utilisable en contexte opérationnel.

L'architecture logicielle a donc été conçue selon les principes suivants :

— **Séparation des responsabilités** (actuariat / ML / interface),

— **Isolation du modèle de production**,

— **Reproductibilité des expérimentations**,

— **Maintenabilité et extensibilité**.

Cette approche permet de rapprocher le prototype d'un standard industriel (séparation production vs recherche).

6.2 Organisation modulaire

Le projet est structuré en trois blocs fonctionnels principaux :

1. Module Actuariat

Ce module contient :

— la lecture et l'exploitation des tables de mortalité,

— la conversion annuelle → mensuelle,

— le calcul des probabilités de survie cumulées,

— la construction du tableau d'amortissement,

— le calcul de la valeur actuelle espérée des sinistres,

— la détermination de la prime mensuelle par équivalence actuarielle.

Ce module constitue la **référence technique** du système.

2. Module Machine Learning

Ce module regroupe :

— le feature engineering,

— l'entraînement des modèles,

— l'évaluation (hold-out, CV),

— la génération des learning curves,

— la sauvegarde et le chargement de modèles.

Il contient deux logiques distinctes :

— **Production** : modèle figé (Random Forest) stocké sur disque ;

— **R&D** : environnement d'expérimentation sans impact sur la production.

3. Interface applicative

L'interface est développée sous Streamlit, ce qui permet :

— une interaction utilisateur fluide,

— une visualisation claire des métriques,

— une séparation par pages fonctionnelles,

— un déploiement simple via GitHub et plateforme cloud.

6.3 Principe fondamental : isolation Production vs R&D

Un point central du design est l'isolation stricte du modèle de production.
Le modèle utilisé dans la page *Moteur IA Production* :

— est entraîné hors interface,

— est sauvegardé sous forme figée,

— n'est jamais modifié par la page R&D.

Cela permet :

— d'éviter toute instabilité liée aux expérimentations,

— de garantir la reproductibilité des prédictions,

— d'imiter un environnement réel (modèle validé puis déployé).

6.4 Pages fonctionnelles de l'application

6.4.1 Moteur Actuariat

Cette page constitue la **référence technique explicable**.
À partir des paramètres du contrat :

$$(L, N, R, x, i, a, \alpha)$$

l'application :

— calcule la mensualité du prêt,

— génère le tableau d'amortissement complet,

— calcule la prime mensuelle actuarielle P_m,

— affiche la mensualité totale (crédit + assurance),

— propose un export CSV.

Cette page permet de justifier chaque étape mathématique.

6.4.2 Moteur IA Production

Cette page charge un modèle Random Forest figé.
Le flux est le suivant :

1. saisie des paramètres du contrat,
2. transformation via le feature engineering,
3. prédiction instantanée de P_m,
4. affichage du résultat.

L'objectif est d'obtenir une estimation quasi instantanée, sans recalcul actuariel complet.

6.4.3 Recherche et Développement

Cette page constitue l'environnement expérimental.
Elle permet :

— d'entraîner différents modèles,

— d'évaluer via hold-out,

— d'activer une validation croisée (k=3,4,5),

— d'afficher les learning curves,

— d'analyser l'importance des variables,

— de sauvegarder un modèle expérimental.

Cette séparation reproduit le cycle industriel classique :

$$\text{Recherche} \rightarrow \text{Validation} \rightarrow \text{Déploiement}$$

6.4.4 Comparaison Actuariat vs IA

Cette page permet une comparaison directe :

— Prime actuarielle P_m,

— Prime IA $\hat{P}_m$,

— Écart absolu $|P_m - \hat{P}_m|$,

— Écart relatif $\frac{|P_m - \hat{P}_m|}{P_m}$.

Elle joue un rôle pédagogique et de contrôle qualité.

6.5 Flux de données

Le pipeline global peut être résumé ainsi :

$$\text{Paramètres prêt} \rightarrow \text{Moteur actuariel} \rightarrow \text{Prime de référence}$$

$$\text{Paramètres prêt} \rightarrow \text{Feature engineering} \rightarrow \text{Modèle ML} \rightarrow \text{Prime prédite}$$

6.6 Considérations de déploiement

L'application est compatible avec un déploiement cloud via :

— GitHub (versioning),

— Streamlit Community Cloud ou Render,

— modèle figé stocké dans le dépôt (taille contrôlée).

La séparation claire des composants facilite :

— la mise à jour d'un modèle sans modifier l'interface,

— le monitoring futur des performances,

— l'industrialisation via API.

6.7 Extrait structurant

Listing 6.1 – Chargement du modèle de production et prédiction

```
# model_pack = load_model(PROD_MODEL_PATH)
# y_hat = predict_new_loan_df(
#     model_pack=model_pack,
#     age_souscription=...,
#     duree=...,
#     capital_emprunte=...,
#     taux_interet_annuel=...,
#     taux_technique_annuel=...
# )
```

Le code complet est disponible dans le dépôt du projet ; seuls les éléments structurants sont inclus dans le mémoire.

Chapitre 7

Conclusion, limites et perspectives

7.1 Bilan général du travail

Ce mémoire avait pour objectif de construire un cadre cohérent articulant deux approches complémentaires : (i) une tarification actuarielle rigoureuse d'une assurance emprunteur décès indexée sur le capital restant dû, (ii) une approximation par Machine Learning permettant une estimation instantanée intégrable dans un outil opérationnel.

La première contribution majeure réside dans la formalisation complète du moteur actuariel. La modélisation du prêt (amortissement et trajectoire du CRD), la conversion de la mortalité annuelle en mortalité mensuelle, l'intégration d'un abattement représentatif d'un profil emprunteur, et le calcul explicite de la valeur actuelle espérée des sinistres ont permis de déterminer une prime mensuelle nivelée par équivalence actuarielle. Cette approche fournit une référence technique explicable, conforme aux standards actuariels classiques.

La seconde contribution consiste à transformer cette fonction actuarielle déterministe en un problème de régression supervisée. La génération d'un dataset cohérent (prêt $\rightarrow$ prime mensuelle), respectant des bornes réalistes et des dépendances plausibles entre variables, a permis l'entraînement et l'évaluation de modèles de Machine Learning. Le modèle de production, figé et isolé de la partie R&D, permet une estimation instantanée de la prime tout en conservant une architecture modulaire et maintenable.

Enfin, l'intégration dans une application Streamlit structurée en plusieurs vues (Actuariat, IA Production, R&D, Comparaison) constitue une contribution produit significative. Elle illustre la transposition d'un cadre académique vers une logique quasi-industrielle : séparation des responsabilités, reproductibilité, validation croisée, learning curves, export de résultats.

Ainsi, le travail ne se limite pas à une démonstration théorique : il propose une chaîne complète allant de la modélisation mathématique à l'implémentation applicative.

7.2 Limites du cadre proposé

Malgré la cohérence méthodologique, plusieurs limites doivent être soulignées.

7.2.1 Données synthétiques

Le modèle ML est entraîné sur des données générées à partir du moteur actuariel. La performance mesurée (MAE, RMSE, R^2) reflète donc essentiellement la capacité du modèle à approximer une fonction connue, et non à prédire un comportement réel observé sur portefeuille. Une calibration sur données réelles serait nécessaire pour évaluer la robustesse en situation opérationnelle.

7.2.2 Hypothèses techniques

La tarification repose sur plusieurs hypothèses structurantes :

— table de mortalité donnée (ex. TH00-02),

— hypothèse de force de mortalité constante intra-annuelle,

— abattement fixe représentatif d'un profil emprunteur,

— conversion proportionnelle du taux technique,

— convention de paiement (prime début de mois) et convention de prestation (CRD début ou fin).

Ces choix influencent directement la valeur de la prime et peuvent différer selon les pratiques d'entreprise.

7.2.3 Segmentation simplifiée

Le modèle ne prend pas en compte des variables médicales ou comportementales (fumeur, IMC, antécédents), ni des règles de tarification commerciale (frais, marges, surprimes médicales, anti-sélection). Il s'agit donc d'une prime pure théorique, non d'un tarif commercial complet.

7.2.4 Cadre stationnaire

Le modèle suppose un environnement stable : mortalité, taux et comportement ne varient pas structurellement. En pratique, les évolutions réglementaires, économiques et démographiques peuvent modifier ces paramètres.

7.3 Apports conceptuels

Au-delà de l'implémentation technique, ce travail met en évidence plusieurs enseignements conceptuels :

— L'actuariat fournit une base explicable et économiquement interprétable ; le ML permet d'en accélérer l'usage opérationnel.

— L'apprentissage sur données simulées est pertinent pour prototyper, mais doit être distingué d'une calibration sur données réelles.

— La séparation claire entre modèle de production et espace R&D est essentielle pour éviter les interférences méthodologiques.

— Les learning curves constituent un outil précieux pour comprendre la dynamique biais-variance et l'impact de la taille d'échantillon.

7.4 Perspectives d'amélioration

Plusieurs axes d'extension peuvent être envisagés :

7.4.1 Calibration sur données réelles

Intégrer un portefeuille réel permettrait :

— de comparer prime actuarielle théorique et tarif observé,

— d'évaluer l'erreur hors distribution,

— d'ajuster les hypothèses de mortalité et d'abattement.

7.4.2 Enrichissement des variables

L'ajout de variables médicales ou comportementales permettrait une segmentation plus fine du risque et une modélisation plus proche des pratiques assurantielles.

7.4.3 Extension actuarielle

— intégration de chargements (frais, marge de sécurité, capital économique),

— modélisation multi-risques (invalidité, incapacité),

— stress-tests sur mortalité et taux.

7.4.4 Explicabilité et gouvernance des modèles

Dans un cadre réglementé, l'explicabilité est cruciale. Des méthodes telles que SHAP ou l'analyse d'importance des variables pourraient être intégrées pour interpréter les prédictions.

7.4.5 Industrialisation

La transformation du prototype en solution industrielle impliquerait :

— déploiement via API,

— pipeline automatisé d'entraînement,

— monitoring des performances,

— gestion de versions du modèle.

7.5 Conclusion finale

Ce mémoire démontre qu'il est possible d'articuler de manière cohérente une tarification actuarielle classique et une approximation par Machine Learning dans un cadre structuré et reproductible. L'actuariat demeure le socle explicatif et théorique du produit, tandis que le ML agit comme un accélérateur opérationnel.

L'approche proposée ne vise pas à opposer actuariat et intelligence artificielle, mais à montrer leur complémentarité : l'un garantit la solidité mathématique et l'interprétabilité, l'autre apporte rapidité et adaptabilité.

Cette convergence constitue un enjeu majeur pour les métiers de l'assurance et de la finance quantitative, où l'exigence de rigueur mathématique doit désormais coexister avec les contraintes de performance numérique.

Bibliographie

[1] Bowers, N. L., Gerber, H. U., Hickman, J. C., Jones, D. A., Nesbitt, C. J. (1997). *Actuarial Mathematics*. Society of Actuaries. Référence classique sur les assurances vie, fonctions actuarielles, EPV et primes.

[2] Dickson, D. C. M., Hardy, M. R., Waters, H. R. (2019). *Actuarial Mathematics for Life Contingent Risks*. Cambridge University Press. Ouvrage moderne détaillant la modélisation vie et les primes nivelées.

[3] Denuit, M., Charpentier, A. (2005). *Mathématiques de l'assurance*. Economica. Cadre théorique français sur la modélisation actuarielle.

[4] Society of Actuaries. *Life Tables and Mortality Studies*. Documentation sur la construction et l'usage des tables de mortalité.

[5] Brigo, D., Mercurio, F. (2006). *Interest Rate Models : Theory and Practice*. Springer. Référence sur la modélisation des taux et l'actualisation.

[6] Fabozzi, F. (2012). *Bond Markets, Analysis and Strategies*. Pearson. Cadre général sur actualisation et flux financiers.

[7] Ross, S., Westerfield, R., Jaffe, J. (2016). *Corporate Finance*. McGraw-Hill. Rappels sur annuités constantes et amortissement de dette.

[8] Documentation technique bancaire. *Amortization Schedules and Loan Mathematics*. Référence pratique sur la construction de tableaux d'amortissement.

[9] Casella, G., Berger, R. (2002). *Statistical Inference*. Duxbury. Fondements statistiques pour estimation et validation.

[10] Wasserman, L. (2004). *All of Statistics*. Springer. Introduction rigoureuse aux probabilités et à l'inférence.

[11] Hastie, T., Tibshirani, R., Friedman, J. (2009). *The Elements of Statistical Learning*. Springer. Référence majeure sur régression, biais-variance et ensembles.

[12] Bishop, C. (2006). *Pattern Recognition and Machine Learning*. Springer. Fondements probabilistes du ML.

[13] Breiman, L. (2001). *Random Forests*. Machine Learning, 45, 5–32. Article fondateur du modèle Random Forest.

[14] Friedman, J. (2001). *Greedy Function Approximation : A Gradient Boosting Machine*. Annals of Statistics. Article fondateur du gradient boosting.

[15] Pedregosa et al. (2011). *Scikit-learn : Machine Learning in Python*. Journal of Machine Learning Research. Référence officielle de la librairie utilisée.

[16] Scikit-learn documentation. https://scikit-learn.org/stable/

[17] James, G., Witten, D., Hastie, T., Tibshirani, R. (2021). *An Introduction to Statistical Learning*. Springer. Version pédagogique complémentaire à ESL.

[18] Lundberg, S., Lee, S. (2017). *A Unified Approach to Interpreting Model Predictions*. Advances in Neural Information Processing Systems. Article fondateur de SHAP.

[19] Streamlit Documentation. https://docs.streamlit.io/

[20] Sculley et al. (2015). *Hidden Technical Debt in Machine Learning Systems*. NIPS. Référence sur la séparation production / R&D.

Annexe A

Annexes

A.1 Structure des colonnes du dataset

TABLE A.1 – Schéma du dataset (à compléter selon la version finale)

Colonne	**Type**	**Description**
age_souscription	int	Âge à la souscription
duree	int	Durée en années (ou mois selon implémentation)
capital_emprunte	float	Montant initial du prêt
taux_interet_annuel	float	Taux nominal annuel du prêt
taux_technique_annuel	float	Taux technique annuel
abat_mortality	float	Abattement mortalité (si utilisé)
target	float	Prime mensuelle actuarielle P_m

A.2 Hyperparamètres du modèle de production

— Random Forest : $n\_estimators = [\cdot]$, $min\_samples\_leaf = [\cdot]$, $random\_state = 42$.

— **À compléter** avec les valeurs exactes du notebook (modèle figé).

A.3 Compléments méthodologiques

— conventions de paiement (prime début de mois vs fin de mois),

— conventions de prestation (CRD début vs fin de mois),

— conversion de mortalité (force constante).