
**Université Gaston Berger de Saint-Louis**
**UFR des Sciences Appliquées et Technologie**
**Département Mathématiques**

MEMOIRE

pour l'obtention du diplôme de

**MASTER en Mathématiques Appliquées**

**Mbayang SYLL**

**Spécialité** : Sciences de Données et Applications

**Option** : Statistique

# TECHNIQUES DE MACHINE LEARNING POUR DONNÉES DE COMPTAGE ET APPLICATIONS

Sous la supervision de :

**Pr Aliou DIOP**, UGB et
**Dr Essoham ALI**,UCO, France

Année universitaire : 2022-2023

# Dédicaces

*À mes parents bien-aimés, ma mère Mame Saye Diop et mon père Mor Talla Syll, qui sont la source inépuisable de mon inspiration et de ma détermination.*
*À toute ma famille, mes frères et sœurs (Maghet, Saliou, Isma, Farma et Arame), pour leur soutien indéfectible et leur encouragement constant.*
*À mon époux bien-aimé, Mouhamed Gassama, sa majesté , la plus précieuse rencontre de ma vie universitaire, dont l'amour et le soutien illuminent ma vie.*
*À ma fille chérie Khadija Rassoul, ma motivation suprême et une bénédiction divine, qui m'inspire chaque jour à poursuivre l'excellence.*
*À ma belle famille, mes beaux-parents Aminata Gaye et Papa Yoro Gassama, pour leur accueil chaleureux et leur soutien inestimable.*
*À mes belles-sœurs Oumou et Maman pour leur affection et leur solidarité.*
*Au duo magique, Fatou et Saly, pour leur amitié précieuse et leur incommensurable soutien.*
*À mes chères camarades de chambre, Fatou et Ami, pour leur complicité et leur soutien pendant cette période académique.*
*À Mr Diallo PC et Mr Barane Kébé, mes professeurs au lycée, pour leur encouragement et leur soutien tout au long de mon parcours académique. Leur enseignement et leurs conseils ont été des éléments clés dans mon développement académique et personnel.*
*À mes camarades de lycée Macodou Kanghé Sall de Kébémer : Yacine Samb, Aliou Gueye, Khadim Gueye, Khalipha Gueye, Mintou Dieng, et Mame Diarra Gueye.*
*À mes enseignants de l'école Kébémer 5 : Monsieur Diop, Madame Dia, et Monsieur Kébé et à mes professeurs du CEM Macodou Kanghé Sall : Monsieur Gueye, Monsieur Sané, et Monsieur Ndoye pour leur engagement envers mon éducation dès le début de mon parcours scolaire.*
*À mes camarades de ce master Sciences des données et applications,*
*Je vous remercie pour votre soutien, votre collaboration et les moments partagés tout au long de ce parcours. Je vous souhaite à tous plein de succès dans vos futurs projets.*
*Et à toute la P27 SAT,particulièrement à mes amis Cheikh Tidjane Gueye et Moussa Diack pour avoir partagé ce voyage éducatif avec moi.*

*Je vous remercie tous du fond du cœur pour les moments que nous avons partagés et pour l'inspiration que vous m'avez apportée. Ce mémoire est le fruit de l'amour, du soutien et des sacrifices de toutes ces personnes qui ont joué un rôle essentiel dans ma vie. Merci à vous tous.*

# Remerciements

Je remercie tout d'abord le Bon Dieu, pour Sa grâce infinie, Ses bénédictions constantes et Sa guidance tout au long de ce parcours académique. C'est par Sa volonté que j'ai pu surmonter les défis et accomplir ce travail.

Je tiens à exprimer ma profonde gratitude à **l'Université Gaston Berger de Saint-Louis** pour m'avoir offert cette opportunité précieuse de poursuivre mes études supérieures. Mon expérience ici a été marquée par un environnement académique stimulant et enrichissant.

À l'Unité de Formation et de Recherches en **Sciences Appliquées et Technologie**, je suis extrêmement reconnaissante pour le soutien institutionnel et les ressources mises à ma disposition tout au long de cette aventure éducative. Votre engagement envers l'excellence académique a été une source constante de motivation pour moi.

Je voudrais également remercier chaleureusement mes encadreurs, le Professeur **Aliou Diop** et le Dr. **Essoham Ali**. En particulier, le Professeur Aliou Diop a été une source d'inspiration et de soutien inégalé. Sa passion pour l'enseignement, sa rigueur scientifique et sa capacité à encourager l'innovation ont été déterminants dans la réalisation de ce mémoire. Son dévouement à l'égard de ses étudiants et son approche bienveillante m'ont non seulement aidée à approfondir mes connaissances mais aussi à développer une véritable passion pour la recherche.

Le Dr. Essoham Ali, avec ses conseils avisés et sa patience, a également joué un rôle crucial dans ce projet. Son expertise et sa disponibilité ont été essentiels pour surmonter les obstacles rencontrés en cours de route.

Aux **membres du jury**, je suis reconnaissante pour le temps et l'énergie qu'ils consacreront à l'évaluation de ce travail. Je suis convaincue que leurs retours constructifs et leurs critiques bienveillantes contribueront grandement à l'amélioration et à l'aboutissement de ce mémoire.

Merci à tous ceux qui, de près ou de loin, ont contribué à la réussite de ce projet.

# Abstract

This thesis examines the use of machine learning (ML) techniques to address the challenges of overdispersion and zero inflation in the analysis of count data. Traditional models, such as Poisson and negative binomial models, often struggle to capture the complexity of the data, leading to biased estimates.

We investigated the effectiveness of several ML models, including Random Forests (RF), Support Vector Machines (SVM), k-Nearest Neighbors (kNN), and Artificial Neural Networks (ANN). These techniques provide flexible approaches to model the complex relationships within count data.

Through empirical studies on real-world data, we demonstrated that ML models, particularly RF and ANN, often outperform classical models in terms of accuracy. However, careful attention is required when tuning hyperparameters to avoid overfitting.

In conclusion, this thesis highlights the potential of ML techniques to enhance the analysis of count data, offering promising perspectives for future applications.

**Keywords :** Count data, Overdispersion, Zero inflation, Machine learning, Non-linearity, Cross-validation.

# Résumé

Ce mémoire examine l'utilisation des techniques de machine learning (ML) pour traiter les défis de la surdispersion et de l'inflation de zéros dans l'analyse des données de comptage. Les modèles traditionnels, tels que ceux de Poisson et binomial négatif, peuvent souffrir de limitations en capturant la complexité des données, entraînant ainsi des estimations biaisées.

Nous avons étudié l'efficacité de plusieurs modèles de ML, notamment les forêts aléatoires (RF), les machines à vecteurs de support (SVM), les k plus proches voisins (kNN) et les réseaux de neurones artificiels (ANN). Ces techniques offrent des approches flexibles pour modéliser les relations complexes dans les données de comptage.

À travers des études empiriques sur des données réelles, nous avons démontré que les modèles de ML, en particulier RF et ANN, surpassent souvent les modèles classiques en termes de précision. Cependant, une attention particulière est nécessaire lors du réglage des hyperparamètres pour éviter le surajustement.

En conclusion, ce mémoire met en lumière le potentiel des techniques de ML pour améliorer l'analyse des données de comptage, offrant ainsi des perspectives prometteuses pour des applications futures.

**Mots-clés :**Données de comptage, Surdispersion,Inflation de zéros, Apprentissage automatique, non-linéarité, validation croisée.

# Liste des abréviations

| | |
|---|---|
| $P(A)$ | : La probabilité de l'événement $A$. |
| $\mathrm{Var}(X)$ | : La variance de la variable aléatoire $X$. |
| $\mathbb{E}(X\|Y)$ | : Espérance conditionnelle de $X$ sachant $Y$. |
| $X_n \xrightarrow{P} Y$ | : $(X_n)_n$ converge en probabilité vers $Y$. |
| $X_n \xrightarrow{L} X$ | : $X_n$ converge en loi vers $X$. |
| i.i.d | : Indépendantes et identiquement distribuées. |
| $\mathbb{N}^*$ | : Ensemble des entiers naturels non nuls. |
| $\mathbb{R}$ | : Ensemble des réels et $\mathbb{R}^d = \mathbb{R} \times \cdots \times \mathbb{R}$ (d fois). |
| $\boldsymbol{X}^\top$ | : Transposée du vecteur $\boldsymbol{X}$. |
| $\mathrm{P}(\lambda)$ | : Loi de Poisson de paramètre $\lambda$. |
| EMV | : Estimateur du Maximum de Vraisemblance. |
| GLM | : Generalised Linear Model. |
| NB | : Negative Binomial. |
| ZIP | : Zero-Inflated Poisson. |
| ZINB | : Zero-Inflated Negative Binomial. |
| ML | : Machine Learning. |
| RF | : Random Forest . |
| KNN | : K-Nearest Neighbors. |
| ANN | : Artificial Neural Networks. |
| SVM | : Support Vector Machines. |
| MAE | : Mean Absolute Error. |
| RMSE | : Root Mean Square Error. |
| AIC | : Akaike Information Criterion. |
| BIC | : Bayesian Information Criterion. |
| $R^2$ | : Coefficient de détermination. |

# Table des matières

# Liste des tableaux

# Table des figures

# Introduction Générale

Les données de comptage, utilisées dans divers domaines pour quantifier des occurrences telles que les incidents ou les occurrences d'événements, peuvent souvent présenter une surdispersion, indiquant une variabilité plus importante que ce qui est prévu par les modèles statistiques traditionnels. Cette surdispersion peut résulter de diverses sources telles que des erreurs de mesure ou des variations naturelles significatives dans les événements étudiés. La surdispersion dans les données de comptage se réfère à une situation où la variance observée dans les données est plus grande que celle attendue selon un modèle statistique donné. Cette divergence peut être due à diverses raisons, telles que des erreurs structurelles, des erreurs d'observation ou des erreurs de conception. Par exemple, dans un contexte plus général que celui de l'écologie, des erreurs structurelles pourraient inclure des conditions inappropriées pour l'événement enregistré, tandis que des erreurs d'observation pourraient se produire lorsqu'un événement se produit mais n'est pas enregistré correctement. De même, des erreurs de conception pourraient inclure des méthodes d'échantillonnage inadéquates ou des biais de sélection.

En général, les données de comptage peuvent présenter des zéros pour diverses raisons, qu'elles soient réelles (l'événement ne se produit pas) ou fausses (l'événement se produit mais n'est pas enregistré). Cette distinction est importante car elle peut affecter la manière dont la surdispersion est modélisée et interprétée. L'inflation des zéros, qui se produit lorsque le nombre de zéros observés est plus élevé que prévu, est souvent associée à des erreurs d'observation.Pour évaluer la surdispersion, on utilise souvent des modèles de régression de comptage tels que le modèle de Poisson ou le modèle binomial négatif. Cependant, ces modèles présentent des limites lorsqu'il s'agit de modéliser l'excès de variance dans les données. Par exemple, le modèle de Poisson suppose une variance égale à la moyenne, ce qui peut ne pas être réaliste dans de nombreux cas réels. De même, bien que le modèle binomial négatif soit plus flexible que le modèle de Poisson en permettant une variance supérieure à la moyenne, il peut également ne pas suffire à modéliser des données fortement surdispersées ou présentant une inflation de zéros.

Face à ces limitations, des modèles plus complexes, tels que les modèles à zéro-inflation, ont été développés pour traiter simultanément la surdispersion et l'inflation des zéros. Cependant, même ces modèles peuvent parfois ne pas être suffisants pour capturer pleinement la structure des données de comptage.

C'est là que les techniques de régression par apprentissage automatique ML entrent en jeu. Des méthodes telles que les forêts aléatoires RF, les machines à vecteurs de support SVM, les k-plus proches voisins KNN et les réseaux de neurones artificiels ANN offrent une flexibilité et une capacité de modélisation supplémentaires qui peuvent être essentielles pour traiter efficacement la surdispersion dans les données de comptage. En utilisant ces techniques, il devient possible de capturer des structures plus complexes dans les données et d'obtenir de meilleures prédictions, même dans des cas où les modèles traditionnels atteignent leurs limites.

# Chapitre 1 : Introduction aux Données de Comptage et leurs Caractéristiques

## 1.1 Introduction

Les données de comptage sont omniprésentes dans diverses disciplines telles que l'épidémiologie, l'économie, la biologie et les sciences sociales. Elles mesurent le nombre d'occurrences d'un événement particulier dans un cadre donné, comme le nombre de consultations médicales, le nombre d'accidents de la route, ou le nombre de défauts de fabrication. Leur spécificité réside dans leur nature discrète et leur distribution souvent asymétrique.

Les caractéristiques distinctives des données de comptage incluent la surdispersion, où la variabilité des données dépasse celle attendue par les modèles classiques comme le modèle de Poisson. De plus, elles tendent à contenir un grand nombre de zéros, introduisant des défis supplémentaires pour les analyses statistiques. Par conséquent, des modèles spécialisés tels que le modèle binomial négatif ou les modèles zéro-inflation sont souvent nécessaires pour capturer la complexité de ces données.

## 1.2 Définition des données de comptage

Les données de comptage sont un type spécifique de données statistiques qui quantifient le nombre d'occurrences d'un événement dans un cadre temporel ou spatial donné. Elles se distinguent par leur nature discrète, leur distribution souvent asymétrique et leur tendance à contenir un grand nombre de zéros. Ces données sont essentielles dans de nombreux domaines, tels que l'épidémiologie, l'écologie, l'économie et la recherche sociale, où elles sont utilisées pour mesurer des phénomènes comme le nombre de maladies, les populations animales, les accidents de la route, ou les incidents criminels.

La modélisation et l'analyse des données de comptage nécessitent des approches statistiques et des techniques de machine learning spécialisées pour traiter leurs particularités, notamment la surdispersion et l'inflation de zéros. Les modèles de régression de Poisson, les modèles binomiaux négatifs, ainsi que les méthodes modernes comme les forêts aléatoires, les machines à vecteurs de support, les k-plus proches voisins et les réseaux de neurones artificiels, sont couramment utilisés pour extraire des insights pertinents et faire des prévisions précises à partir de ces données.

## 1.3 Caractéristiques des données de comptage

Les données de comptage se distinguent par plusieurs caractéristiques spécifiques qui influencent leur analyse et leur modélisation. Comprendre ces caractéristiques est crucial pour choisir les méthodes statistiques et de machine learning les plus appropriées.

### 1.3.1 Discrètes et Non-Négatives

Les données de comptage sont des observations discrètes et non-négatives. Elles prennent des valeurs entières positives, incluant zéro. Par exemple, le nombre de visites à un site web, le nombre de patients admis à l'hôpital, ou le nombre de poissons capturés dans un filet.

### 1.3.2 Distribution Asymétrique

Les données de comptage ont souvent une distribution asymétrique et peuvent être fortement biaisées à droite. Cela signifie qu'il y a une longue traîne de valeurs élevées, tandis que la majorité des observations sont proches de zéro.

### 1.3.3 Présence de Zéros Excessifs

Une autre caractéristique fréquente des données de comptage est la présence de nombreux zéros. Cette inflation de zéros peut être due à divers facteurs, tels que l'absence de l'événement compté dans de nombreuses unités d'observation.

### 1.3.4 Surdispersion

La surdispersion se produit lorsque la variance des données de comptage est supérieure à la moyenne. Cela peut indiquer une hétérogénéité non prise en compte par les modèles simples, comme le modèle de Poisson. La surdispersion peut être causée par la variabilité entre les sujets ou par la présence de sous-groupes distincts au sein des données.

### 1.3.5 Non-Linéarité

Les relations entre les variables prédictives et les variables de réponse dans les données de comptage peuvent être non-linéaires. Par conséquent, les méthodes de régression linéaire simples peuvent ne pas capturer adéquatement ces relations complexes.

### 1.3.6 Hétérogénéité

Les données de comptage peuvent présenter une grande hétérogénéité due à des variations dans les conditions sous lesquelles les données sont collectées. Par exemple, les taux d'incidents peuvent varier considérablement entre différentes périodes ou lieux.

### 1.3.7 Autocorrélation

Dans certains cas, les observations de données de comptage peuvent être autocorrélées. Cela signifie que les comptages successifs dans une série temporelle ou dans des unités spatiales proches peuvent être corrélés.

### 1.3.8 Échelle de Temps

Les données de comptage sont souvent collectées sur des intervalles de temps spécifiques. La granularité et la période d'observation peuvent varier et doivent être prises en compte lors de l'analyse.

### 1.3.9 Effets de Bordure

Les effets de bordure se produisent lorsque les unités d'observation ont des limites naturelles, telles que des zones géographiques ou des périodes de temps, ce qui peut affecter les comptages observés.

En comprenant et en tenant compte de ces caractéristiques, les analystes peuvent mieux sélectionner et adapter les modèles statistiques et de machine learning pour une analyse précise et significative des données de comptage.

## 1.4 Problèmes Courants Rencontrés dans l'Analyse des Données de Comptage

L'analyse des données de comptage pose plusieurs défis uniques qui nécessitent des méthodes adaptées pour assurer des résultats précis et fiables. Voici quelques-uns des problèmes courants :

### 1.4.1 Surdispersion

La surdispersion survient lorsque la variance des données de comptage dépasse la moyenne, ce qui va à l'encontre des hypothèses du modèle de Poisson standard. Ce phénomène peut entraîner des erreurs d'estimation des paramètres et rendre les tests statistiques peu fiables. Des modèles comme le binomial négatif ou les modèles de Poisson à inflation de zéros (ZIP) sont souvent utilisés pour gérer la surdispersion.

### 1.4.2 Inflation de Zéros

De nombreuses données de comptage présentent une proportion élevée de zéros, ce qui est connu sous le nom d'inflation de zéros. Les modèles traditionnels de Poisson ou binomiaux négatifs peuvent ne pas suffire à capturer cette distribution particulière. Les modèles de régression à inflation de zéros (ZIP et ZINB) sont conçus pour traiter cette situation en modélisant séparément les processus générant les zéros et les valeurs positives.

### 1.4.3 Autocorrélation

Les données de comptage peuvent présenter une autocorrélation, surtout dans le cadre des séries temporelles ou des données spatiales. Cette corrélation entre les observations successives peut biaiser les estimations des paramètres et les inférences. Les modèles autorégressifs de comptage et autres techniques similaires sont utilisés pour prendre en compte l'autocorrélation.

### 1.4.4 Hétérogénéité Non Observée

L'hétérogénéité non observée fait référence à la variation entre les unités d'observation qui n'est pas capturée par les covariables incluses dans le modèle. Cette hétérogénéité peut entraîner une surdispersion et biaiser les résultats des modèles. Les modèles mixtes, incluant des effets aléatoires, sont souvent employés pour gérer cette hétérogénéité.

### 1.4.5 Valeurs Aberrantes

Les valeurs aberrantes, ou outliers, sont des observations qui s'écartent significativement des autres données et peuvent influencer disproportionnellement les résultats de l'analyse. Ces valeurs peuvent être le résultat d'erreurs de mesure ou de variations extrêmes. Des méthodes robustes ou des techniques spécifiques aux données de comptage, comme les modèles de Poisson tronqués, peuvent aider à traiter ces valeurs aberrantes.

### 1.4.6 Petits Échantillons

Les petits échantillons peuvent poser des défis importants en termes d'estimation précise des paramètres et de validation des modèles. Les techniques de bootstrap ou les approches bayésiennes peuvent être utilisées pour améliorer la robustesse des estimations dans le cas de petits échantillons.

### 1.4.7 Données Manquantes

Les données manquantes peuvent biaiser les analyses et réduire la puissance statistique des tests. Des méthodes d'imputation des données manquantes, comme l'imputation multiple, peuvent être employées pour gérer ce problème et améliorer la qualité des analyses.

## 1.5 Conclusion

Dans ce chapitre, nous avons exploré en profondeur les fondements des données de comptage, mettant en lumière leur définition, leurs caractéristiques spécifiques, ainsi que les défis courants rencontrés dans leur analyse. Nous avons défini les données de comptage comme des quantifications d'occurrences d'événements dans des cadres temporels ou spatiaux donnés, soulignant leur nature discrète et souvent asymétrique.

Les caractéristiques distinctives des données de comptage, telles que la non-négativité, l'asymétrie, la présence de zéros excessifs, la surdispersion, la non-linéarité, l'hétérogénéité, l'autocorrélation, et les effets de bordure, ont été examinées en détail. Chacune de ces caractéristiques influence les méthodes d'analyse et de modélisation appropriées pour ces données.

Nous avons également discuté des problèmes courants rencontrés dans l'analyse des données de comptage, tels que la surdispersion, l'inflation de zéros, l'autocorrélation, l'hétérogénéité non observée, les valeurs aberrantes, les petits échantillons, et les données manquantes. Pour chaque problème, nous avons présenté des approches méthodologiques spécifiques et des modèles adaptés, comme les modèles de Poisson à inflation de zéros, les modèles binomiaux négatifs, et les techniques de machine learning modernes.

Ce chapitre a établi une base solide pour les chapitres suivants, où nous explorerons des méthodes de machine learning avancées pour l'analyse et la modélisation des données de comptage, en mettant en œuvre ces approches sur des données réelles pour illustrer leur efficacité et leur pertinence. En comprenant les particularités des données de comptage et les défis associés, nous sommes mieux préparés à appliquer des techniques sophistiquées et à interpréter les résultats de manière précise et significative.

# Chapitre 2 : Modèles Traditionnels de Régression de Comptage

## 2.1 Introduction

Les modèles de régression de comptage jouent un rôle crucial dans l'analyse de données où les résultats représentent des comptes discrets de la fréquence d'un événement. Ces modèles sont largement appliqués dans des domaines variés, tels que l'écologie, la santé publique, les assurances, et les sciences sociales, pour examiner et prédire l'occurrence d'événements en fonction d'un ensemble de variables explicatives. Les modèles les plus couramment utilisés pour ce type d'analyse incluent notamment les modèles de Poisson, binomial négatif (NB),les modèles à inflation de zéros ZIP et ZINB qui sont devenus des outils de référence dans le traitement des données de comptage.

Ces modèles permettent d'établir des liens entre des variables explicatives et la fréquence d'occurrences d'un phénomène, tout en tenant compte de la nature discrète et souvent non négative des données. Cependant, malgré leur efficacité dans des scénarios simples, ces modèles peuvent montrer des limites lorsqu'ils sont confrontés à des problèmes plus complexes, comme la surdispersion et l'inflation de zéros, qui sont fréquents dans les données de comptage réelles.

## 2.2 Modèle de Poisson

Le modèle de régression de Poisson est l'un des modèles de comptage les plus utilisés en raison de sa simplicité et de son interprétabilité. Ce modèle suppose que les comptes suivent une distribution de Poisson, où la moyenne et la variance des données sont égales. En d'autres termes, le modèle de Poisson est adapté pour les données où la variance est proportionnelle à la moyenne.
Le modèle de régression de Poisson est exprimé comme suit :

$$P(Y = y_{ij}|x_{ij}) = \frac{\exp(-\mu_{ij}) \cdot \mu_{ij}^{y_{ij}}}{y_{ij}!} \tag{2.1}$$

où $y_{ij}$ est un entier non négatif et $Y_{ij}$ est la variable réponse pour le sujet $i$ ($i = 1, \ldots, n$) à l'instant $j$ ($j = 1, \ldots, n_i$). Nous supposons une fonction de liaison logarithmique où $\mu_{ij} = \exp(\beta_0, \beta_1 x_{i1}, ..., \beta_k x_{ik})$ est une combinaison linéaire de variables prédictives avec

$$E(y_i|X_i = x_i) = \mu_i = \exp(\beta^\top X_i)$$

où $P$ est la probabilité, $y_i$ est une variable de comptage observée (un nombre d'événements) pour l'individu $i$, $X_i$ est un vecteur de $p$ variables explicatives linéairement indépendantes observées pour l'individu $i$, et $\beta = (\beta_0, \beta_1, \ldots, \beta_p)$ est un vecteur de paramètres de dimension appropriée $p \times 1$. La forme de la fonction exponentielle assure la non-négativité du paramètre de la moyenne $\mu$. La fonction log-vraisemblance du modèle est donnée par l'équation suivante :

$$\ell(\beta) = \sum_{i=1}^{n} \left\{ y_i \beta^\top X_i - \exp(\beta^\top X_i) - \log(y_i!) \right\} \tag{2.2}$$

Les paramètres sont choisis de façon à maximiser la valeur de la fonction log-vraisemblance. Les conditions du premier ordre sont :

$$\sum_{i=1}^{n} X_{ij} \left( y_i - \exp(\beta^\top X_i) \right) = 0, \quad j = 1, \ldots, p \tag{2.3}$$

Le modèle de régression de Poisson est trop restrictif pour les données de comptage, ce qui a incité les chercheurs à recourir à des modèles alternatifs comme le modèle binomial négatif, qui permet la surdispersion. Par définition, le modèle de régression de Poisson ne peut pas modéliser les données de comptage surdispersées ; ainsi, un mélange gamma-Poisson est souvent supposé dans ce cas.

Cependant, l'une des principales limitations du modèle de Poisson est son incapacité à gérer la surdispersion, une condition où la variance des données dépasse la moyenne. Cette limitation peut conduire à des estimations biaisées et à des inférences incorrectes, rendant le modèle de Poisson inadapté pour les données de comptage réelles qui présentent souvent une variabilité excessive.

## 2.3 La superdispersion

La **surdispersion** désigne une variabilité supérieure à celle attendue dans un ensemble de données, par rapport à un modèle statistique théorique, tel que le modèle de Poisson. Ce phénomène est courant dans l'analyse des données réelles, où les populations observées sont souvent hétérogènes, contrairement aux hypothèses des modèles paramétriques simples. En particulier, dans un modèle de Poisson, la surdispersion survient lorsque la variance des données dépasse la moyenne, une situation problématique car elle conduit à une sous-estimation des écarts-types des estimateurs et peut rendre une variable faussement significative.

Les principales causes de la surdispersion incluent des corrélations positives entre les observations, une variabilité excessive entre les données ou la violation des hypothèses de distribution du modèle. Elle peut aussi être apparente, causée par des omissions de variables explicatives importantes ou une mauvaise spécification de la fonction de lien dans le modèle.

La surdispersion est fréquemment observée dans les données de comptage, où la variance dépasse souvent la moyenne, rendant inadapté l'usage exclusif du modèle de Poisson. Cela nécessite donc de corriger ce problème en ajoutant des variables explicatives, en modifiant la fonction de lien ou en adoptant des modèles alternatifs comme le modèle binomial négatif, qui tient compte de la surdispersion.

Le **modèle binomial négatif** est une extension du modèle de Poisson qui permet de mieux gérer la surdispersion en introduisant un paramètre supplémentaire pour modéliser la

variabilité excessive dans les données. Contrairement au modèle de Poisson, où la variance est égale à la moyenne, le modèle binomial négatif permet à la variance d'être supérieure à la moyenne, offrant ainsi une flexibilité accrue dans l'analyse des données de comptage. Ce modèle est particulièrement utile lorsque la variance observée excède largement la moyenne, ce qui est souvent le cas dans les contextes où les données présentent une hétérogénéité non négligeable.

## 2.4 Modèle binomial négatif

Pour surmonter le problème de la surdispersion, le modèle binomial négatif NB est souvent utilisé comme une extension de la régression de Poisson. Le modèle NB introduit un paramètre de dispersion supplémentaire pour tenir compte de l'excès de variance par rapport à la moyenne.

La distribution binomiale négative NB sert d'alternative à la distribution de Poisson lorsque les données présentent une surdispersion .

Le **modèle binomial négatif** est une extension du modèle de Poisson, conçu pour tenir compte de la surdispersion des données en introduisant un terme d'hétérogénéité non observé pour l'observation $i$. Ce modèle s'écrit de la manière suivante :

$$E(Y_i \mid X_i = x_i, \vartheta_i) = \mu_i \vartheta_i = \exp(\beta^\top X_i)\vartheta_i \tag{2.4}$$

où $\vartheta_i$ suit une loi Gamma de moyenne 1 et de variance $\alpha$. Conditionnellement à $X_i$, $Y_i$ est distribué selon une loi binomiale négative avec la probabilité :

$$P(Y_i = y_i \mid X_i = x_i) = \frac{\Gamma(y_i + 1/\alpha)}{y_i! \Gamma(1/\alpha)} \left(\frac{1}{1+\alpha\mu_i}\right)^{1/\alpha} \left(\frac{\mu_i}{1/\alpha + \mu_i}\right)^{y_i} \tag{2.5}$$

Ici, $\alpha$ est un paramètre auxiliaire qui mesure le degré de surdispersion. Cette distribution présente une moyenne conditionnelle $\mu_i$ et une variance conditionnelle $\mu_i(1 + \alpha\mu_i)$. Lorsque $\alpha$ tend vers zéro, la loi binomiale négative converge vers la loi de Poisson.

Le modèle binomial négatif est plus flexible que le modèle de Poisson car il permet à la variance d'être supérieure à la moyenne, ce qui le rend mieux adapté aux données surdispersées. Cependant, bien qu'il soit plus robuste, le modèle NB peut parfois encore ne pas capturer toute la complexité des données de comptage, notamment lorsque les données contiennent un excès de zéros.

Si le nombre de faux zéros est trop élevé pour être correctement modélisé par un modèle de régression NB, les chercheurs se tournent souvent vers des modèles à inflation de zéros tels que les modèles de Poisson à inflation de zéros (ZIP) et les modèles binomiaux négatifs à inflation de zéros (ZINB), qui sont considérés comme appropriés dans de tels cas.

## 2.5 Modèles de régression à inflation de zéros

### 2.5.1 Introduction

Le phénomène d'inflation de zéros a été initialement identifié dans les données de comptage. Cette situation, où les données contiennent un nombre excessif de zéros par rapport à ce que les modèles traditionnels prédiraient, a posé des défis importants pour l'analyse statistique. Pour répondre à ces défis, de nouveaux modèles de régression, comme les modèles de Poisson à

zéros-inflation (ZIP) et les modèles binomiaux négatifs à zéros-inflation (ZINB), ont été développés. Ces modèles sont spécifiquement conçus pour traiter l'inflation de zéros en permettant une meilleure modélisation des données de comptage présentant une surabondance de zéros, offrant ainsi des estimations plus précises et des prédictions plus fiables.

Les modèles à inflation de zéros ZIP ou ZINB sont un mélange de deux distributions. Ils modélisent les vrais zéros à l'aide de la distribution de Poisson ou binomiale négative NB et une distribution dégénérée à zéro, qui modélise les vrais zéros .

Pour une variable réponse $Y_i$, où $i = 1, \ldots, n$, on dira que :

### 2.5.2 le modèle Poisson zéro-inflation (ZIP)

- $Y_i$ est modélisée par un modèle de Poisson à inflation de zéros (ZIP) si sa distribution est exprimée comme suit :

$$P(Y_i = y_i \mid X_i, Z_i) = \begin{cases} \pi_i + (1-\pi_i)\exp(-\lambda_i) & \text{si } y_i = 0 \\ (1-\pi_i)\frac{\exp(-\lambda_i)\lambda_i^{y_i}}{y_i!} & \text{si } y_i > 0 \end{cases} \tag{2.6}$$

où $\pi_i$ et $\lambda_i$ sont fonctions respectivement des vecteurs de covariables $\mathbf{Z}_i = (Z_{i1}, \cdots, Z_{iq})$ et $\mathbf{X}_i = (X_{i1}, \cdots, X_{ip})$.

Dans la régression ZIP, la probabilité de mélange $\pi_i$ et le paramètre $\lambda_i$ sont généralement modélisés par des modèles logistiques et log-linéaires respectivement, c'est-à-dire :

$$\text{logit}(\pi_i) = \gamma^\top \mathbf{Z}_i \quad \text{et} \quad \log(\lambda_i) = \beta^\top \mathbf{X}_i. \tag{2.7}$$

où $\beta = (\beta_1, \cdots, \beta_p)^\top$ et $\gamma = (\gamma_1, \cdots, \gamma_q)^\top$ sont des vecteurs de paramètres inconnus.

On peut synthétiser le modèle sous la forme suivante :

$$\forall i = 1, \cdots, n, \begin{cases} Y_i \sim \pi_i \delta_0 + (1-\pi_i) P(\lambda_i) \\ \text{logit}(\pi_i) = \gamma^\top \mathbf{Z}_i \\ \log(\lambda_i) = \beta^\top \mathbf{X}_i \end{cases} \tag{2.8}$$

Conditionnellement à $\mathbf{X}_i$ et $\mathbf{Z}_i$, l'espérance et la variance de $Z_i$ sont données par :

$$\mathbb{E}(Y_i \mid \mathbf{X}_i, \mathbf{Z}_i) = (1-\pi_i)\lambda_i \tag{2.9}$$

et

$$\text{var}(Y_i \mid \mathbf{X}_i, \mathbf{Z}_i) = (1-\pi_i)\lambda_i(1+\pi_i\lambda_i). \tag{2.10}$$

Ce modèle permet de capturer la proportion de zéros excédentaires souvent observée dans les données de comptage, tout en tenant compte de la composante de Poisson pour les valeurs non nulles.

#### Estimation dans le modèle ZIP

Dans le domaine de la régression de Poisson avec inflation de zéros, plusieurs auteurs, tels que Lambert (17) et Czado et al. (20), ont proposé des méthodes d'estimation, généralement basées sur le maximum de vraisemblance (EMV). Toutefois, cette méthode présente des limitations, notamment sa sensibilité aux valeurs aberrantes et son instabilité en cas de séparation

insuffisante des composantes du mélange. Pour remédier à ces problèmes, il est proposé que la probabilité $\pi$ d'appartenir au groupe des zéros varie d'un individu à l'autre.

Considérons un ensemble de vecteurs indépendants $(Y_i, X_i, Z_i)$ observés à partir des modèles définis. La log-vraisemblance des paramètres $\theta = (\beta^\top, \gamma^\top)^\top$ peut s'écrire sous une forme spécifique :

$$\begin{aligned} \ell_n(\theta) = \sum_{i=1}^{n} \Big[ J_i \log\left(e^{\gamma^\top Z_i} + \exp\left(-e^{\beta^\top X_i}\right)\right) \\ +(1-J_i)\left(Y_i \beta^\top X_i - e^{\beta^\top X_i} - \log(Y_i!)\right) - \log\left(1+e^{\gamma^\top Z_i}\right)\Big] \end{aligned} \tag{2.11}$$

où

$$J_i = \mathbf{1}\{Y_i = 0\}$$

En particulier, considérons la variable indicatrice $S_i$, définie comme suit : $S_i = 1$ lorsque $z_i$ provient de la distribution dégénérée des zéros, et $S_i = 0$ lorsque $z_i$ provient d'un zéro aléatoire (distribution non dégénérée). Dans ce contexte, la log-vraisemblance des données complètes $(z, S)$ peut s'exprimer comme :

$$\ell_C^n(y, S; \theta) = \sum_{i=1}^{n} \left[ S_i(\gamma^\top Z_i) - \log\left(1+e^{\gamma^\top Z_i}\right)\right] + (1-S_i)\left[Y_i\beta^\top X_i - e^{\beta^\top X_i} - \log(Y_i!)\right] \tag{2.12}$$

Cela peut être réécrit sous la forme :

$$\ell_C^n(z, S; \theta) = \tilde{\ell}_n, 1(\gamma) + \tilde{\ell}_n, 2(\beta) \tag{2.13}$$

où $S = (S_1, \ldots, S_n)^\top$.

L'algorithme d'estimation du maximum de vraisemblance est alors réalisé par une procédure itérative, commençant par des valeurs initiales $\left(\beta^{(0)\top}, \gamma^{(0)\top}\right)^\top$ et alternant entre l'estimation des variables indicatrices et la maximisation des fonctions de vraisemblance.

Avec l'algorithme EM, voir Dempster et al. (22), la log-vraisemblance est maximisée de manière itérative en commençant par une valeur initiale

$$\left(\beta^{(0)\top}, \gamma^{(0)\top}\right)^\top$$

et en alternant les étapes suivantes :

**Étape E :** Estimer la variable $S_i$ par son espérance conditionnelle $S_i^{(r)}$ sous les estimations courantes des paramètres $\beta^{(r)}$ et $\gamma^{(r)}$.

**Étape M :** Trouver $\beta^{(r+1)}$ et $\gamma^{(r+1)}$ en maximisant respectivement les fonctions $\tilde{\ell}_{n,1}(\gamma) + \tilde{\ell}_{n,2}(\beta)$. Hall et Shen (21) ont montré que maximiser ces deux fonctions revient à résoudre respectivement les deux équations suivantes :

$$\frac{1}{n}\sum_{i=1}^{n}\left\{S_i^{(r)} - \pi_i\right\} Z_i = 0 \tag{2.14}$$

$$\frac{1}{n}\sum_{i=1}^{n}\left(1 - S_i^{(r)}\right)\left\{y_i - e^{\beta^\top X_i}\right\} X_i = 0 \tag{2.15}$$

Hall et Shen (21) ont développé une approche d'estimation robuste, appelée estimation par espérance-robuste (RES), pour remplacer les équations traditionnelles. Cette méthode pèse les observations en fonction de leur position dans la distribution de Poisson, permettant ainsi d'améliorer la robustesse des estimations. Sous des conditions de régularité de Rosen et al. (26) liées à l'algorithme ES et de Carroll et al. (23), Hall et Shen (21) ont montré le résultat suivant plus général dans le cas où $\theta = \left(\beta^\top, \gamma^\top\right)^\top \in \mathbb{R}^{p+q}$ dans Czado et al. (20) .

**Théorème**

*Si l'algorithme RES converge, alors il existe une suite de variables aléatoires $\hat{\theta}$ telles que :*

1. $\hat{\theta} \xrightarrow{P} \theta_0$ quand $n \to \infty$ (consistance),
2. $\sqrt{n}(\hat{\theta}_n - \theta_0) \xrightarrow{L} N(0, V(\theta_0))$ quand $n \to \infty$ (normalité asymptotique).

Où l'expression $V(\theta_0)$ de la variance asymptotique est donnée dans Hall et Shen (21). Des extensions du modèle ZIP ont été étudiées. Citons, entre autres, Lam et al. (24), He et al. (27) et Nguyen et al. (25) qui ont étendu ce modèle ZIP respectivement dans le cadre semi-paramétrique, doublement semi-paramétrique et de la censure, et ont établi les résultats de consistance et de normalité asymptotique des estimateurs proposés.

### 2.5.3 le modèle binomial négatif zéro-inflation (ZINB)

- $Y_i$ est modélisée par un modèle binomial négatif à inflation de zéros (ZINB) si sa distribution est donnée par :

$$P(Y_i = y_i \mid X_i, Z_i) = \begin{cases} \pi_i + (1-\pi_i)\left(\frac{1}{1+\alpha\lambda_i}\right)^{\frac{1}{\alpha}} & \text{si } y_i = 0 \\ (1-\pi_i)\frac{\Gamma(y_i+\frac{1}{\alpha})}{\Gamma(\frac{1}{\alpha})y_i!}\left(\frac{\alpha\lambda_i}{1+\alpha\lambda_i}\right)^{y_i}\left(\frac{1}{1+\alpha\lambda_i}\right)^{\frac{1}{\alpha}} & \text{si } y_i > 0 \end{cases} \tag{2.16}$$

avec

$$E(Y_i \mid X_i, Z_i) = (1-\pi_i)\lambda_i \tag{2.17}$$

et

$$\text{var}(Y_i \mid X_i, Z_i) = (1-\pi_i)\lambda_i\left(1 + (\alpha+\pi_i)\lambda_i\right), \tag{2.18}$$

où $\alpha$ est un paramètre de surdispersion. Dans les deux cas, $\pi_i$ représente la probabilité d'inflation de zéro.

Comme pour les modèles de Poisson et binomial négatif, le modèle ZINB tend vers le modèle ZIP lorsque $\alpha$ tend vers zéro. Pour ces deux modèles, on suppose que la probabilité $\pi_i$ et la moyenne conditionnelle $\lambda_i$ sont respectivement modélisées par $\text{logit}(\pi_i) = \gamma^\top Z_i$ et par $\log(\lambda_i) = \beta^\top X_i$. Les vecteurs $X_i \in \mathbb{R}^p$ et $Z_i \in \mathbb{R}^q$ sont les covariables, et $\beta \in \mathbb{R}^p$ et $\gamma \in \mathbb{R}^q$ sont les vecteurs des paramètres inconnus. Les covariables $X_i$ et $Z_i$ peuvent ou non avoir des composantes communes.

## 2.6 Limitations et inconvénients des modèles traditionnels

Les modèles traditionnels de régression de comptage, tels que les modèles de Poisson et binomial négatif, sont largement utilisés pour analyser des données de comptage. Cependant, ces modèles présentent plusieurs limitations et inconvénients qui peuvent affecter leur efficacité et leur précision dans certaines situations.

### 2.6.1 Modèle de Poisson

1. **Hypothèse de la moyenne égale à la variance** : Le modèle de Poisson repose sur l'hypothèse que la moyenne et la variance des données de comptage sont égales. Dans de nombreuses applications pratiques, cette hypothèse est souvent violée, ce qui conduit à une surdispersion (la variance est supérieure à la moyenne) ou à une sous-dispersion (la variance est inférieure à la moyenne).
2. **Surdispersion** : Lorsque les données présentent une surdispersion, le modèle de Poisson peut sous-estimer la variance, entraînant des erreurs standard incorrectes et des tests statistiques biaisés. Cela peut conduire à des conclusions erronées sur les relations entre les variables.
3. **Manque de flexibilité** : Le modèle de Poisson est souvent trop rigide pour capturer les structures complexes des données de comptage, notamment lorsque des zéros excessifs ou des valeurs aberrantes sont présents.

### 2.6.2 Modèle Binomial Négatif

1. **Complexité accrue** : Bien que le modèle binomial négatif puisse traiter la surdispersion en introduisant un paramètre de dispersion supplémentaire, il ajoute de la complexité au modèle et nécessite des méthodes d'estimation plus sophistiquées. Cela peut compliquer l'interprétation des résultats et augmenter le temps de calcul.
2. **Hypothèses spécifiques** : Le modèle binomial négatif repose sur l'hypothèse que la variance est une fonction quadratique de la moyenne. Dans certains cas, cette relation peut ne pas être appropriée, ce qui limite l'adéquation du modèle.
3. **Problèmes avec les zéros excessifs** : Comme le modèle de Poisson, le modèle binomial négatif peut également rencontrer des difficultés avec les données comportant un nombre excessif de zéros, ce qui nécessite l'utilisation de modèles plus spécialisés comme les modèles à zéros-inflation.

### 2.6.3 Besoin de Modèles Plus Sophistiqués

Pour surmonter les limitations des modèles traditionnels de régression de comptage, il est souvent nécessaire d'utiliser des modèles plus sophistiqués, tels que les modèles de Poisson à zéros-inflation (ZIP) et les modèles binomiaux négatifs à zéros-inflation (ZINB). Ces modèles permettent de mieux capturer la complexité des données de comptage en traitant à la fois la surdispersion et l'inflation des zéros.

## 2.7 Surdispersion et ses implications dans les modèles traditionnels

La surdispersion est une situation courante dans les données de comptage où la variance des données est supérieure à la moyenne. Cette caractéristique pose des défis significatifs pour les modèles traditionnels de régression de comptage, tels que les modèles de Poisson, qui reposent sur l'hypothèse d'équidispersion (moyenne égale à la variance).

### 2.7.1 Implications dans les Modèles de Poisson

- **Estimation Biaisée** : Dans un modèle de Poisson, la surdispersion conduit à une sous-estimation de la variance des estimations des paramètres, ce qui entraîne des erreurs standard incorrectes et des intervalles de confiance trop étroits.
- **Tests Statistiques Invalides** : La sous-estimation de la variance conduit à des tests statistiques biaisés, augmentant la probabilité de faux positifs (déclaration erronée d'une relation significative entre les variables).
- **Prévisions Inexactes** : Les prévisions basées sur un modèle de Poisson surdispersé peuvent être inexactes car le modèle ne capture pas adéquatement la variabilité réelle des données.

### 2.7.2 Modèle Binomial Négatif comme Alternative

Le modèle binomial négatif (NB) est souvent utilisé comme alternative au modèle de Poisson pour traiter la surdispersion. Ce modèle introduit un paramètre de dispersion supplémentaire pour mieux capturer la variabilité des données.

- **Fonctionnement du Modèle NB** : Supposons qu'une variable aléatoire $Y$ suit une distribution de Poisson avec une moyenne conditionnelle $\mu_{ij}$ et que le paramètre $\lambda$ suit une distribution gamma avec une moyenne $E(\lambda) = \mu_{ij}$ et une variance $\text{Var}(\lambda) = \mu_{ij}^2\theta^{-1}$. Le modèle NB est défini par la densité conjointe des distributions de Poisson et gamma, ce qui conduit à :

$$P(Y = y_{ij}|\mu_{ij}, \theta) = \frac{\Gamma(y_{ij} + \theta^{-1})}{\Gamma(\theta^{-1})y_{ij}!} \left(\frac{\mu_{ij}\theta}{1 + \mu_{ij}\theta}\right)^{y_{ij}} \left(\frac{1}{1 + \mu_{ij}\theta}\right)^{\theta^{-1}} \tag{2.19}$$

  où $\mu_{ij}$ est la moyenne et le paramètre de dispersion $\phi = 1 + \mu_{ij}\theta$ doit être supérieur à un.
- **Avantages du Modèle NB** : Le modèle NB est plus flexible que le modèle de Poisson et peut mieux gérer la surdispersion, offrant des estimations plus précises et des tests statistiques valides.

### 2.7.3 Limites des Modèles NB

Malgré ses avantages, le modèle binomial négatif peut également présenter des limitations, notamment lorsqu'il y a un excès de zéros dans les données. Dans de tels cas, les modèles à zéros-inflation, tels que le modèle de Poisson à zéros-inflation (ZIP) et le modèle binomial négatif à zéros-inflation (ZINB), sont souvent utilisés pour modéliser simultanément la surdispersion et l'inflation des zéros.

### 2.7.4 Modèles à Zéros-Inflation

Les modèles à zéros-inflation, ZIP et ZINB, sont des mélanges de deux distributions qui modélisent les vrais zéros à travers la distribution de Poisson ou binomiale négative et une distribution dégénérée à zéro, qui modélise les faux zéros.

— **Modèle ZIP** : Le modèle ZIP a deux composantes, $\pi$ qui représente les observations de zéro et $(1 - \pi)$ qui représente une variable aléatoire de Poisson observée.

— **Modèle ZINB** : Le modèle ZINB est similaire au modèle ZIP mais utilise une distribution binomiale négative pour modéliser les comptes observés, offrant une plus grande flexibilité pour gérer la surdispersion.

## 2.8 Conclusion

Les modèles traditionnels ont démontré une capacité à capturer certaines dynamiques des données de comptage, mais leur efficacité est souvent limitée par la complexité et la non-linéarité inhérentes à ces données. Les défis tels que la surdispersion et l'inflation de zéro, qui sont fréquents dans les données de comptage, nécessitent des approches plus sophistiquées pour être correctement modélisés et interprétés.

En outre, les techniques d'apprentissage automatique, telles que les forêts aléatoires, les machines à vecteurs de support, les k-plus proches voisins et les réseaux de neurones artificiels, offrent des approches flexibles et robustes pour modéliser efficacement les données de comptage. Ces techniques peuvent fournir des prédictions plus précises et des informations plus approfondies sur les relations entre les variables.

La transition vers ces techniques de machine learning s'avère cruciale pour surmonter les limitations des modèles traditionnels. Leur capacité à capturer des relations non linéaires et à gérer des structures complexes de données permet d'améliorer la qualité des prédictions et d'offrir des perspectives plus enrichissantes pour l'analyse des données de comptage.

# Chapitre 3 : Machine Learning pour l'Analyse de Données de Comptage

## 3.1 Introduction

L'analyse des données de comptage pose des défis uniques en raison de leur nature discrète et souvent asymétrique, ainsi que des phénomènes de surdispersion et d'inflation des zéros. Les méthodes traditionnelles de régression de comptage, telles que les modèles de Poisson et binomial négatif, peuvent être limitées dans leur capacité à capturer les complexités inhérentes à ces données. Pour surmonter ces limitations, les techniques de Machine Learning (ML) offrent des alternatives puissantes et flexibles.

Les techniques de ML sont capables de modéliser des relations complexes et non linéaires dans les données, et elles peuvent être particulièrement utiles pour traiter des problèmes tels que la surdispersion et l'inflation des zéros. Ce chapitre explore plusieurs méthodes de ML qui ont montré leur efficacité dans l'analyse des données de comptage, à savoir les forêts aléatoires (Random Forests), les machines à vecteurs de support (Support Vector Machines), les k plus proches voisins (k-Nearest Neighbors), et les réseaux de neurones artificiels (Artificial Neural Networks).

- **Forêts Aléatoires (Random Forests)** : Cette technique utilise un ensemble de décideurs (arbres de décision) pour améliorer la précision et éviter le surajustement. Les forêts aléatoires sont particulièrement efficaces pour gérer la variabilité des données de comptage et identifier les interactions complexes entre les variables.
- **Machines à Vecteurs de Support (Support Vector Machines)** : Les SVM sont utilisés pour la classification et la régression et sont capables de modéliser des frontières de décision non linéaires à l'aide de noyaux. Ils sont utiles pour séparer les données de comptage en catégories distinctes.
- **k Plus Proches Voisins (k-Nearest Neighbors)** : Cette méthode de classification et de régression non paramétrique se base sur la similarité des données. Elle est simple à mettre en œuvre et peut être très efficace pour les petits ensembles de données de comptage.
- **Réseaux de Neurones Artificiels (Artificial Neural Networks)** : Les ANN, inspirés du fonctionnement du cerveau humain, sont capables de modéliser des relations très complexes dans les données. Ils sont particulièrement utiles lorsque les données de comptage présentent des motifs non linéaires difficiles à capturer avec des méthodes traditionnelles.

## 3.2 Introduction aux techniques de Machine Learning (ML)

Ici, nous discutons des techniques de régression par apprentissage automatique (ML) considérées dans cette étude. Notre problème de régression est de la forme : $y_{ij} = f(x_{ij}) + \epsilon_{ij}$ en présence d'observations nulles où

$$(X, Y) = \{(x_{ij}, y_{ij}) \mid i = 1, \ldots, N\}$$

$(X, Y)$ représente un ensemble contenant $N$ éléments, où chaque élément est une paire $(x_{ij}, y_{ij})$. Dans cette notation :

— $X = \{x_{ij} \mid i = 1, \ldots, N\}$ est l'ensemble des variables prédictives.

— $Y = \{y_{ij} \mid i = 1, \ldots, N\}$ est l'ensemble des variables cibles.

Chaque paire $(x_{ij}, y_{ij})$ correspond à un échantillon de données individuel, avec $i$ indiquant l'indice de l'échantillon et $j$ indiquant l'indice des variables ou des visites dans le contexte des données de comptage. Les observations $y_{ij}$ sont des valeurs discrètes qui comptent les occurrences d'un événement donné, tandis que $x_{ij}$ représentent les variables explicatives ou prédictives associées à ces observations.et $\epsilon_{ij}$ le terme d'erreur.

L'objectif est d'étudier comment les techniques de régression par ML peuvent réduire la surdispersion dans les données de comptage. Les techniques de ML, dans divers domaines, n'ont pas été pleinement appliquées à l'analyse des données de comptage. Cela est dû au manque de collaboration entre la communauté de recherche en ML et les spécialistes des domaines spécifiques, à une absence de communication sur les applications réussies du ML dans l'analyse des données de comptage et à la difficulté de valider les modèles de ML. Les techniques de ML possèdent de nombreux algorithmes et méthodologies capables de résoudre des problèmes réels.

## 3.3 Les techniques supervisées

### 3.3.1 Forêts aléatoires (RF)

Les forêts aléatoires (RF) sont une méthode d'apprentissage automatique sophistiquée qui combine plusieurs arbres de décision pour améliorer la précision prédictive tout en évitant le surajustement des données. En agrégeant les résultats de multiples arbres, chacun construit sur un échantillon aléatoire des données d'entraînement, les forêts aléatoires équilibrent la variance et le biais, offrant ainsi des prédictions robustes et généralisables.

Le modèle de forêts aléatoires peut être exprimé sous la forme suivante :

$$y(x) = f\left\{\sum_{k=1}^{K} w_k \phi(x, v_k)\right\} \tag{3.1}$$

où $v_k$ représente le choix de la variable à diviser et la fonction $f()$ dépend du type de modèle requis, qu'il s'agisse d'un arbre de régression ou de classification. Chaque arbre est développé en utilisant un sous-ensemble de $k$ caractéristiques choisies aléatoirement . Ainsi, K est le nombre total d'arbres (ou itérations) utilisés dans la construction de la forêt aléatoire, un paramètre clé pour contrôler la performance du modèle.

Les forêts aléatoires peuvent être utilisées pour des problèmes de régression et de classification ; cependant, dans cette étude, nous nous concentrons uniquement sur les tâches de régression. Les RF sont couramment illustrées par la construction de nombreux arbres de décision à

partir d'échantillons bootstrap d'un ensemble de données. Étant donné que les arbres individuels surajustent souvent les données d'entraînement et entraînent des prédictions bruyantes, la moyenne des prédictions permet de réduire la variance du modèle et d'améliorer la précision des prédictions.

Dans le modèle de RF, il existe trois paramètres de réglage d'intérêt : la taille des nœuds, le nombre d'arbres et le nombre de variables prédictives échantillonnées à chaque division. L'accent dans cette étude est mis sur un paramètre de réglage qui est le nombre de variables prédictives échantillonnées à chaque division, ($mtry$). Le $mtry$ est déterminé par le nombre total de variables prédictives dans l'ensemble de données et il contrôle le surajustement .

- **Taille des nœuds :** La taille minimale des nœuds terminales dans chaque arbre.
- **Nombre d'arbres :** Le nombre total d'arbres dans la forêt.
- **Nombre de variables prédictives échantillonnées à chaque division ($mtry$) :** Ce paramètre contrôle combien de variables sont considérées pour la division à chaque nœud.

### 3.3.2 Machines à vecteurs de support (SVM)

Les machines à vecteurs de support (SVM) sont des modèles d'apprentissage automatique puissants et flexibles, utilisés principalement pour des tâches de classification et de régression. Elles sont basées sur le concept de maximisation de la marge, ce qui permet de trouver un hyperplan qui sépare les données en classes distinctes avec une marge maximale.

Pour un ensemble de données $D = \{(x_i, y_i)\}_{i=1}^{p} \subseteq \mathbb{R}^n \times \{-1, +1\}$, l'objectif est de trouver une fonction $f(x) = y$ qui classifie correctement les motifs des données, où $x_i$ désigne un vecteur n-dimensionnel et $y_i$ est son étiquette. Le but des machines à vecteurs de support (SVM) est de créer une fonction hyperplan qui sépare les observations en classes. L'hyperplan peut être défini comme suit :

$$f(x) = (\mathbf{w} \cdot \mathbf{x}) + b \tag{3.2}$$

où $\mathbf{w} \in \mathbb{R}^n$ ; $b \in \mathbb{R}$ et les données sont alors linéairement séparables, si un tel hyperplan existe . Les SVM résolvent également des problèmes non linéaires en mappant les vecteurs d'entrée dans un espace de dimension supérieure à l'aide de fonctions noyau $k(x_i, x_j) = \langle \phi(x_i), \phi(x_j) \rangle$ . Ensuite, la fonction de décision peut être écrite comme suit :

$$f(x) = \text{sign} \left\{ \sum_{i=1}^{p} (\mathbf{w} \cdot \mathbf{x}) + b \right\} \tag{3.3}$$

La complexité computationnelle de l'approche SVM est l'une de ses principales caractéristiques. Pour le modèle SVM, le noyau de fonction gaussienne à base radiale (RBF) a été utilisé puisque nos données de comptage étaient non linéaires. Cette fonction particulière ne dépend que d'un seul paramètre de réglage, qui est le paramètre de coût ($C$). Le paramètre de coût détermine les possibles erreurs de classification du modèle. Il impose essentiellement une pénalité au modèle pour chaque erreur commise. Plus la valeur de $C$ est élevée, moins il est probable que le SVM commette une erreur de classification.

Dans cette étude, nous appliquons les SVM aux problèmes de régression des données de comptage. Les SVM pour la régression offrent des avantages significatifs dans la gestion des données de comptage en raison de leur capacité à capturer des relations non linéaires et à gérer la surdispersion et les valeurs aberrantes. Les étapes clés incluent :

1. **Sélection du noyau :** Le choix de la fonction noyau (linéaire, polynomiale, RBF) qui transforme les données d'entrée dans un espace de caractéristiques de haute dimension.
2. **Réglage des hyperparamètres :** L'optimisation des paramètres $C$ et $\epsilon$ pour équilibrer le compromis entre biais et variance, et pour contrôler la largeur de la marge.
3. **Entraînement du modèle :** Utilisation des données d'entraînement pour ajuster les paramètres $w$ et $b$ de manière à minimiser la fonction de coût tout en respectant les contraintes de marge.
4. **Validation et évaluation :** Validation croisée et évaluation des performances du modèle sur des données de test pour garantir sa capacité à généraliser sur des données non vues.

En conclusion, les SVM sont des outils puissants pour la régression des données de comptage, permettant de gérer efficacement la surdispersion et les structures complexes inhérentes à ces types de données. Leur capacité à travailler avec des noyaux non linéaires les rend particulièrement adaptés pour capturer des relations non linéaires dans les données.

### 3.3.3 Approche des k-plus proches voisins (kNN)

Le modèle des k-plus proches voisins (kNN) est une méthode d'apprentissage supervisé utilisée pour la régression, particulièrement adaptée aux données de comptage. Le principe fondamental du kNN est de prédire la valeur d'une observation cible en fonction des k observations les plus proches dans l'ensemble d'apprentissage, déterminées par une mesure de distance, comme la distance euclidienne.

Les $k$-plus proches voisins (kNN) sont identifiés en calculant la distance euclidienne entre le vecteur de caractéristiques d'entrée $x$ et les autres points de l'ensemble de données. Le prédicteur $\hat{f}(x)$ est déterminé au point $x$ en définissant d'abord un voisinage $N_k(x)$, qui est l'ensemble des $k$ observations les plus proches de $x$ dans l'échantillon d'apprentissage en $\mathbb{R}^d$. Le prédicteur est ensuite calculé comme la moyenne des valeurs de sortie des $k$-plus proches voisins :

$$\hat{f}(x) = \frac{1}{k} \sum_{i \in N_k(x)} Y_i \tag{3.4}$$

où $N_k(x)$ représente le voisinage de $x$ défini par les $k$ points les plus proches $i$ dans l'échantillon d'entraînement. Pour améliorer la précision, on peut ajouter des poids à chaque voisin, par exemple un poids inversement proportionnel à la distance euclidienne, ce qui donne plus d'importance aux voisins les plus proches.
Dans cette étude, nous appliquons le modèle des $k$-plus proches voisins (kNN) aux problèmes de régression des données de comptage. Le modèle kNN pour la régression offre des avantages significatifs dans la gestion des données de comptage en raison de sa simplicité et de sa capacité à capturer des relations non linéaires. Les étapes clés incluent :

1. **Définition du Voisinage :** Identifier le voisinage $N_k(x)$ pour une observation cible $x$, composé des $k$ observations les plus proches en termes de distance euclidienne.
2. **Calcul de la Prédiction :** Calculer la prédiction $\hat{f}(x)$ comme la moyenne des valeurs des $k$-plus proches voisins.
$$\hat{f}(x) = \frac{1}{k} \sum_{i \in N_k(x)} Y_i$$

3. **Pondération des Voisins :** Améliorer la précision en attribuant un poids à chaque voisin, inversement proportionnel à sa distance par rapport à $x$, donnant ainsi plus d'importance aux voisins les plus proches.
4. **Normalisation des Données :** Normaliser les caractéristiques pour garantir une mesure de distance cohérente, essentielle pour des performances optimales du modèle kNN.
5. **Validation et Évaluation :** Effectuer une validation croisée et évaluer les performances du modèle sur des données de test pour garantir sa capacité à généraliser sur des données non vues.

En conclusion, le modèle kNN est un outil puissant pour la régression des données de comptage, permettant de capturer efficacement les relations non linéaires et de gérer la complexité des données. Sa capacité à ajuster les poids des voisins en fonction de leur proximité le rend particulièrement adapté pour les données de comptage, où la précision des prédictions est cruciale.

### 3.3.4 Réseaux de Neurones Artificiels (ANN)

Les réseaux de neurones artificiels (ANN) sont des généralisations des modèles linéaires inspirées par des analogies avec le cerveau biologique. L'architecture d'un réseau de neurones artificiels (ANN) est basée sur le perceptron multicouche (MLP). Un perceptron est une unité simple qui calcule la fonction suivante :

$$h(\alpha) = f\left(\sum_i w_i \alpha_i\right) \tag{3.5}$$

où la fonction d'activation $f$ est une fonction non linéaire de son argument. Parmi les exemples de fonctions d'activation, on trouve :

$$f(x) = \begin{cases} +1, & \text{si } x \geq 0, \\ -1, & \text{si } x < 0 \end{cases} \tag{3.6}$$

ou la fonction sigmoïde :

$$f(x) = \frac{1}{1 + e^{-x}} \tag{3.7}$$

Les ANN sont interconnectés à travers un mécanisme de propagation directe (feed-forward), où chaque neurone reçoit des entrées des neurones précédents. Le réseau commence par des couches d'entrée qui sont reliées à chaque neurone dans une ou plusieurs couches cachées utilisant un algorithme de rétropropagation pour ajuster les poids placés sur chaque neurone afin d'améliorer la puissance prédictive. Ce processus est itératif, la dernière couche cachée étant suivie par une couche de sortie pour produire une réponse prédite.

Dans cette étude, nous appliquons les ANN aux problèmes de régression des données de comptage. Les ANN offrent des avantages significatifs dans la gestion des données de comptage en raison de leur capacité à capturer des relations complexes et non linéaires. Les étapes clés incluent :

1. **Choix de l'Architecture du Réseau :** Déterminer le nombre de couches cachées et de neurones par couche, ainsi que les fonctions d'activation appropriées.

2. **Préparation des Données :** Normaliser les données d'entrée pour assurer une convergence efficace pendant l'entraînement.
3. **Entraînement du Modèle :** Utiliser un algorithme de rétropropagation pour ajuster les poids du réseau afin de minimiser une fonction de coût, telle que l'erreur quadratique moyenne (MSE).
4. **Réglage des Hyperparamètres :** Optimiser les hyperparamètres, y compris le taux d'apprentissage, le nombre d'époques d'entraînement et les paramètres de régularisation pour éviter le surapprentissage.
5. **Validation et Évaluation :** Effectuer une validation croisée et évaluer les performances du modèle sur des données de test pour garantir sa capacité à généraliser sur des données non vues.

En conclusion, les ANN sont des outils puissants pour la régression des données de comptage, permettant de gérer efficacement la surdispersion et de capturer des structures complexes inhérentes à ces types de données. Leur capacité à apprendre des représentations non linéaires les rend particulièrement adaptés pour traiter des données de comptage avec des relations complexes.

## 3.4 Les techniques non-supervisées

### 3.4.1 K-means

L'algorithme K-means est une méthode de *clustering* partitionnel qui cherche à regrouper les points de données en un certain nombre de clusters, en minimisant la somme des carrés des distances entre chaque point et le centre de son cluster (centroïde).

Soit un ensemble de données $X = \{x_1, x_2, \ldots, x_n\}$, où chaque point $x_i$ est un vecteur de caractéristiques dans un espace $\mathbb{R}^d$. L'objectif est de diviser les $n$ points en $K$ clusters, représentés par les centroïdes $C = \{c_1, c_2, \ldots, c_K\}$.

L'algorithme K-means minimise la somme des distances au carré entre chaque point et son centroïde correspondant :

$$\min_{\{C_k\}} \sum_{i=1}^{n} \sum_{k=1}^{K} \mathbb{1}_{\{x_i \in C_k\}} \|x_i - \mu_k\|^2 \tag{3.8}$$

où :
— $C_k$ représente le $k$-ième cluster,
— $\mu_k$ est le centroïde du $k$-ième cluster, soit la moyenne des points du cluster $C_k$,
— $\|x_i - \mu_k\|^2$ est la distance euclidienne au carré entre $x_i$ et le centroïde $\mu_k$.

### 3.4.2 DBSCAN(Density-Based Spatial Clustering of Applications with Noise) :

L'algorithme DBSCAN est un algorithme de clustering basé sur la densité, qui regroupe les points denses en clusters et considère les points faiblement connectés ou isolés comme du bruit. Soit $X = \{x_1, x_2, \ldots, x_n\}$ un ensemble de données où chaque point $x_i \in \mathbb{R}^d$ est un

vecteur de caractéristiques dans un espace à $d$ dimensions. DBSCAN est défini à partir de deux paramètres :

— $\epsilon$ : rayon pour définir le voisinage d'un point,
— $MinPts$ : nombre minimum de points requis pour qu'un point soit considéré comme dense.

**1. $\epsilon$-voisinage ($\epsilon$-neighborhood)**

Le $\epsilon$-voisinage d'un point $x_i$ est l'ensemble des points $x_j$ qui sont à une distance inférieure ou égale à $\epsilon$ :

$$N_\epsilon(x_i) = \{x_j \in X : \|x_i - x_j\| \leq \epsilon\}$$

où $\|\cdot\|$ représente la distance euclidienne.

**2. Points principaux (Core points)**

Un point $x_i$ est appelé point principal s'il possède au moins $MinPts$ voisins dans son $\epsilon$-voisinage :

$$|N_\epsilon(x_i)| \geq MinPts$$

**3. Points frontaliers (Border points)**

Un point $x_i$ est appelé point frontalier s'il appartient à l' $\epsilon$-voisinage d'un point principal, mais n'a pas lui-même suffisamment de voisins pour être un point principal :

$$|N_\epsilon(x_i)| < MinPts \quad \text{et} \quad x_i \in N_\epsilon(x_j) \text{ avec } |N_\epsilon(x_j)| \geq MinPts$$

**4. Points de bruit (Noise points)**

Un point $x_i$ est considéré comme du bruit s'il n'est ni principal ni frontalier :

$$|N_\epsilon(x_i)| < MinPts \quad \text{et} \quad x_i \notin N_\epsilon(x_j) \text{ pour tout } x_j$$

**Objectif de DBSCAN** L'objectif de DBSCAN est de former des clusters à partir de points principaux et de leur voisinage dense tout en excluant les points de bruit. Un cluster $C$ est défini comme l'ensemble maximal de points connectés densément :

$$C = \{x_i \in X : \exists x_j \in C \text{ tel que } x_i \in N_\epsilon(x_j)\}$$

## 3.5 Validation croisée

La validation croisée est une technique essentielle en machine learning, permettant d'évaluer la performance d'un modèle sur des données non vues. Elle consiste à diviser l'ensemble de données en plusieurs sous-ensembles (ou "folds") pour entraîner et tester le modèle de manière itérative. Cette méthode offre une estimation plus fiable de la capacité de généralisation du modèle, en réduisant le risque de surajustement.

### 3.5.1 Méthodes de Validation Croisée

#### Validation Croisée K-Folds

Soit un ensemble de données $D$ contenant $n$ observations. Dans la validation croisée K-Folds, cet ensemble est divisé en $k$ sous-ensembles de taille égale. Le modèle est entraîné sur $k-1$ folds et testé sur le fold restant. Le processus est répété $k$ fois, ce qui peut être formulé comme suit :

$$\text{Erreur}_{\text{CV}} = \frac{1}{k} \sum_{i=1}^{k} \text{Erreur}(M_i) \tag{3.9}$$

où $M_i$ est le modèle entraîné sur les $k-1$ folds et testé sur le fold $i$.

#### Validation Croisée Stratifiée

Cette méthode garantit que la proportion de chaque classe est maintenue dans chaque fold. Pour un ensemble de données avec des classes $C_1, C_2, \ldots, C_m$, la proportion dans chaque fold $i$ est :

$$P(C_j) = \frac{N_j}{n}, \quad j = 1, 2, \ldots, m \tag{3.10}$$

où $N_j$ est le nombre d'observations de la classe $C_j$ et $n$ est le nombre total d'observations.

#### Leave-One-Out Cross-Validation (LOOCV)

Dans cette méthode, pour chaque observation $i$, le modèle est entraîné sur $n-1$ observations et testé sur l'observation $i$. La validation croisée LOOCV peut être exprimée par :

$$\text{Erreur}_{\text{LOOCV}} = \frac{1}{n} \sum_{i=1}^{n} \text{Erreur}(M_{-i}) \tag{3.11}$$

où $M_{-i}$ est le modèle entraîné sur toutes les observations sauf l'observation $i$.

#### Validation Croisée de Temps

Pour les données temporelles, la validation croisée prend en compte la séquence chronologique. Si l'on considère une série temporelle $T = \{t_1, t_2, \ldots, t_n\}$, le modèle est entraîné sur les observations jusqu'à un temps $t_k$ et testé sur $t_{k+1}$. La performance peut être mesurée par :

$$\text{Erreur}(t_k) = f(t_k) - \hat{y}(t_k) \tag{3.12}$$

où $f(t_k)$ est la valeur réelle à $t_k$ et $\hat{y}(t_k)$ est la prédiction du modèle.

### 3.5.2 Avantages de la Validation Croisée

— **Estimation Fiable** : La validation croisée fournit une estimation plus robuste de la performance du modèle, car elle utilise l'ensemble de données de manière plus efficace.

— **Réduction du Surajustement** : En évaluant le modèle sur plusieurs sous-ensembles, la validation croisée aide à identifier si le modèle est trop complexe ou s'il généralise bien aux données non vues.
— **Sélection de Modèles** : Elle permet de comparer plusieurs modèles ou hyperparamètres en fournissant des estimations de performance plus précises.

### 3.5.3 Limitations de la Validation Croisée

— **Coût Computationnel** : La validation croisée peut nécessiter des ressources computationnelles importantes, surtout pour de grands ensembles de données ou des modèles complexes.
— **Variabilité des Estimations** : Les performances peuvent varier en fonction de la division des données, ce qui peut parfois conduire à des estimations moins stables, notamment pour des jeux de données de petite taille.

## 3.6 Comparaison des performances des techniques de ML par rapport aux modèles traditionnels

L'une des étapes cruciales dans l'analyse des données de comptage est de comparer les performances des techniques de machine learning (ML) avec celles des modèles traditionnels. Cette comparaison permet de déterminer l'efficacité et la précision des approches modernes en comparaison avec les méthodes établies. Dans cette section, nous allons évaluer les performances de plusieurs techniques de ML, notamment les forêts aléatoires (RF), les machines à vecteurs de support (SVM), les k plus proches voisins (kNN), et les réseaux de neurones artificiels (ANN), par rapport aux modèles de régression de Poisson et de régression binomiale négative.

### 3.6.1 Critères de Performance

Pour effectuer cette comparaison, nous utiliserons plusieurs critères de performance :
— **Erreur quadratique moyenne (RMSE)** : Mesure la différence moyenne entre les valeurs prédites et les valeurs observées.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{3.13}$$

— **Erreur absolue moyenne (MAE)** : Représente la moyenne des erreurs absolues entre les valeurs prédites et les valeurs observées.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{3.14}$$

— **Coefficient de détermination ($\mathbf{R^2}$)** : Indique la proportion de la variance dans la variable dépendante qui est prévisible à partir des variables indépendantes.

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{3.15}$$

— **AIC (Akaike Information Criterion)** et **BIC (Bayesian Information Criterion)** : Critères utilisés pour la sélection de modèles basés sur la qualité d'ajustement et la simplicité du modèle.

$$AIC = 2k - 2\ln(L) \tag{3.16}$$

$$BIC = \ln(n)k - 2\ln(L) \tag{3.17}$$

où $k$ est le nombre de paramètres du modèle, $n$ est le nombre d'observations, et $L$ est la valeur du maximum de vraisemblance du modèle.

## 3.7 Conclusion

Dans ce chapitre, nous avons exploré l'application des techniques de machine learning pour l'analyse des données de comptage, en mettant particulièrement l'accent sur les forêts aléatoires (RF), les machines à vecteurs de support (SVM), les k plus proches voisins (kNN), et les réseaux de neurones artificiels (ANN). À travers une comparaison systématique avec les modèles traditionnels de régression de Poisson et de régression binomiale négative, nous avons mis en évidence les avantages et les inconvénients de chaque approche.

Les techniques de machine learning, notamment les RF et les ANN se sont révélées particulièrement efficaces pour gérer la surdispersion et les non-linéarités inhérentes aux données de comptage. La flexibilité et la robustesse des modèles de ML permettent de mieux modéliser les données et d'obtenir des prédictions plus fiables.

Cependant, il est important de noter que les modèles de machine learning nécessitent souvent un réglage minutieux des hyperparamètres et une validation croisée rigoureuse pour éviter le surapprentissage. De plus, leur complexité computationnelle peut être un obstacle, surtout avec des jeux de données volumineux ou des ressources informatiques limitées.

En conclusion, ce chapitre démontre que l'intégration des techniques de machine learning dans l'analyse des données de comptage offre des perspectives prometteuses et peut significativement améliorer la qualité des analyses et des prédictions. Il est essentiel de poursuivre les recherches dans ce domaine pour affiner ces techniques et les rendre encore plus accessibles et efficaces pour les analystes de données.

# Chapitre 4 : Application Pratique des Modèles sur des Données Réelles

## 4.1 Introduction

L'objectif de ce chapitre est de présenter une analyse approfondie de l'efficacité des modèles de machine learning pour l'analyse des données de comptage en utilisant des données réelles. Les modèles sélectionnés pour cette étude incluent les forêts aléatoires, les machines à vecteurs de support (SVM), les k-plus proches voisins (k-NN) et les réseaux de neurones artificiels (ANN). Chacun de ces modèles offre des avantages uniques pour capturer les complexités inhérentes aux données de comptage, telles que la surdispersion et l'inflation de zéro.

Dans ce chapitre, nous commencerons par une brève description de chaque modèle, suivie de la méthodologie d'implémentation et de paramétrage spécifique. Ensuite, nous présenterons les résultats obtenus pour chaque modèle, accompagnés d'une analyse détaillée des performances et de l'interprétation des résultats. Nous discuterons également des défis rencontrés lors de la mise en œuvre et de l'évaluation des modèles, ainsi que des solutions potentielles pour améliorer la précision des prédictions.

L'utilisation de données réelles permet de valider les modèles dans des conditions pratiques, offrant ainsi une compréhension plus nuancée de leur applicabilité et de leur robustesse. Cette évaluation empirique est essentielle pour déterminer l'efficacité des modèles de machine learning dans des contextes réels et pour guider les décisions futures dans l'application de ces techniques aux données de comptage.

Enfin, nous conclurons ce chapitre par une synthèse des résultats et des recommandations pour l'application pratique des modèles de machine learning dans divers domaines.

## 4.2 Description des données

### 4.2.1 Jeu de Données 1 : Enregistrements de Présence des Espèces de Poissons

Le jeu de données utilisé dans cette étude provient des *"Fish Species Occurrence Records for Uganda Mobilized Observation Archives"*, tiré des archives non publiées de l'Ouganda et disponible sur https://www.gbif.org. L'étude se concentre sur une seule espèce, *Lates niloticus*. Connue sous le nom de *"capitaine d'eau douce"* ou *"perche du Nil"*, cette espèce est très prisée en raison de la qualité de sa chair et de sa croissance rapide. Par conséquent, *Lates niloticus* est une espèce particulièrement importante, ayant de nombreuses implications pratiques.

Ces dernières années, les pisciculteurs ont porté leur attention sur cette espèce, qui joue également un rôle dans le contrôle naturel de la densité des poissons. *Lates niloticus* est également mondialement connue pour avoir été introduite dans la partie ougandaise du lac Victoria dans les années 1950 pour revitaliser une pêche en déclin et contribuer au développement économique de la région Lowe. Au cours des années 1970, les populations de *Lates niloticus* ont augmenté rapidement, provoquant le déclin, voire la disparition complète de certaines espèces indigènes, dont de nombreux cichlidés endémiques . Il est donc pertinent d'examiner cette problématique dans le cadre de cette étude.

La collecte des données analysées dans cette étude a été réalisée dans les principaux bassins fluviaux, notamment les lacs Kwania, Kyoga, Nabugabo, Nakuwa, Nawampasa, Victoria, ainsi que dans le Nil. Cette espèce est présente dans divers types d'eau douce, préférant les eaux tropicales chaudes (27°N–7°S) où elle atteint de grandes tailles et se trouve en densité élevée. Les poissons sont de couleur argentée avec une teinte bleutée, ont un œil noir distinctif entouré d'un anneau jaune vif, et pèsent généralement entre 2 et 4 kg. La longueur moyenne de *Lates niloticus* se situe entre 85 et 100 cm, mais certains spécimens peuvent atteindre jusqu'à 193 cm. Les femelles sont généralement plus grandes que les mâles.

Plusieurs mesures ont été prises dans le cadre de cette étude, notamment le nombre de *Lates niloticus* capturés, qui constitue la variable cible. Des informations supplémentaires, telles que les coordonnées GPS (latitude et longitude) des sites de capture, ont également été recueillies. Les poissons ont été capturés à l'aide d'engins de pêche tels que des filets maillants, des hameçons et des sennes de plage dans différents plans d'eau. Une description détaillée des données est disponible dans (1). Cette étude se penche sur l'influence des facteurs environnementaux sur la reproduction de *Lates niloticus*. La variable de réponse considérée dans cette analyse de régression est le nombre de *Lates niloticus* capturés ($Y_i$). On peut observer qu'une grande partie des données de comptage se situe entre 0 et 50, comme illustré dans la Fig. 1. En effet, la Fig. 1 montre qu'il y a de nombreuses observations nulles, ce qui indique la présence de surdispersion.

#### Statistiques descriptives

| Variable | Mean | sd | Min | Max |
|---|---|---|---|---|
| **decimalLatitude** | 0.59 | 0.61 | -0.65 | 1.85 |
| **decimalLongitude** | 28.11 | 11.87 | 0.00 | 34.02 |
| **individualCount** | 18.82 | 32.11 | 0.00 | 282.00 |

TABLE 4.1 – Statistiques descriptives des variables du jeu de données `Species`.

### 4.2.2 Jeu de Données 2 :Mortalité des Agneaux Djallonké au Sénégal

#### Description

Cette étude de terrain vise à évaluer l'effet du déparasitage des brebis (prévention du parasitisme gastro-intestinal) sur la mortalité de leurs agneaux (âgés de moins d'un an). Ce jeu de données est extrait d'une vaste base de données sur la production et la santé des petits ruminants au Sénégal (11). Les données ont été collectées dans un échantillon de troupeaux à Kolda (Haute Casamance, Sénégal) lors d'une enquête multisite (12). Les références citées fournissent une présentation de l'enquête de suivi (13) ainsi qu'une description des systèmes d'élevage (14).

#### Structure des données

Le jeu de données contient 21 observations sur les 4 variables suivantes :
— **group** : Un facteur avec 2 niveaux : `CTRL` (groupe témoin, sans traitement) et `TREAT` (groupe traité, avec déparasitage).
— **village** : Un facteur indiquant le village du troupeau.
— **herd** : Un facteur indiquant le troupeau.
— **n** : Un vecteur numérique indiquant le nombre d'animaux exposés à la mortalité.
— **trisk** : Un vecteur numérique indiquant le temps d'exposition à la mortalité (en années).
— **y** : Un vecteur numérique indiquant le nombre de décès.

#### Variable cible

Dans ce jeu de données, la variable cible est **y**, qui représente le *nombre de décès* des agneaux. Cette variable est celle que l'on cherche à prédire ou à expliquer en fonction des autres variables explicatives, telles que le groupe de traitement (`CTRL` ou `TREAT`), le village, le troupeau, le nombre d'animaux exposés à la mortalité (`n`) et le temps d'exposition (`trisk`). Ces facteurs peuvent influencer la mortalité des agneaux et sont utilisés pour modéliser la variable `y`.

#### Contexte d'utilisation

Ces données permettent d'analyser les taux de mortalité entre les différents groupes (`CTRL` vs `TREAT`), tout en prenant en compte des facteurs comme le village, le troupeau et le temps d'exposition. Des techniques de régression pour données de comptage, telles que la régression de Poisson ou la régression binomiale négative, peuvent être utilisées pour modéliser ces données, car la variable de sortie (`y`) représente un nombre de décès (données de comptage). On peut observer qu'une grande partie des données de comptage se situe entre 0 et 10, comme illustré dans la Fig. 2.

#### Statistiques descriptives

| Variable | Mean | Sd |
|---|---|---|
| n | 7.11 | 6.20 |
| trisk | 35.01 | 20.74 |
| y | 1.83 | 2.58 |

TABLE 4.2 – Moyennes et écarts-types des variables

### 4.2.3 Jeu de Données 3 : Analyse de l'Utilisation des Services de Santé chez les Seniors : Données de la National Medical Expenditure Survey (1987-1988) : DebTrivedi

Les données utilisées proviennent de la National Medical Expenditure Survey (NMES) réalisée en 1987-1988 aux États-Unis. Cette enquête offre un panorama complet de l'utilisation des services de santé par les Américains âgés de 66 ans et plus. Plusieurs mesures d'utilisation

des soins de santé ont été rapportées, telles que le nombre de consultations chez un médecin en cabinet (ofp) et le nombre de visites chez un professionnel de santé non-médecin en cabinet. Des informations sur l'état de santé des patients sont également disponibles, ainsi que des variables sociodémographiques et économiques. Une description détaillée des données peut être trouvée dans Deb et Trivedi (1997).

L'objectif de cette analyse est d'identifier les facteurs déterminants dans la décision des patients de consulter un médecin en cabinet (ofp). Parmi les variables disponibles, on trouve : i) des variables socio-économiques : sexe (1 pour femme, 0 pour homme), âge (en années, divisé par 10), état civil, niveau d'éducation (nombre d'années d'études), revenu, ii) diverses mesures de l'état de santé : nombre de maladies chroniques (cancer, arthrite, problèmes de vésicule biliaire, etc.) et une variable indiquant l'état de santé perçu par le patient (mauvais, moyen, excellent), et iii) une variable binaire indiquant si l'individu est couvert par Medicaid ou non (Medicaid est une assurance santé américaine pour les personnes à faible revenu et aux ressources limitées, codée 1 si l'individu est couvert et 0 sinon). L'état de santé perçu est re-codé en deux variables indicatrices : "health1" (1 si la santé est perçue comme mauvaise, 0 sinon) et "health2" (1 si la santé est perçue comme excellente, 0 sinon).

L'analyse est axée sur l'identification des covariables significatives influençant la variable cible, qui est le nombre de consultations chez un médecin en cabinet (ofp). Notamment, il a été observé que 15.50159 % des valeurs de la variable cible ofp sont des zéros,comme le montre la figure 2 ce qui pourrait avoir des implications significatives pour les analyses statistiques et les modèles prédictifs.

**Statistiques descriptives**

TABLE 4.3 – Résumé statistique des variables du jeu de données `DebTrivedi`

| Variable | Moyenne | Écart-type |
|---|---|---|
| `ofp` | 5.77 | 6.76 |
| `ofnp` | 1.62 | 5.32 |
| `opp` | 0.75 | 3.65 |
| `opnp` | 0.54 | 3.88 |
| `emer` | 0.26 | 0.70 |
| `hosp` | 0.30 | 0.75 |
| `health*` | 1.95 | 0.45 |
| `numchron` | 1.54 | 1.35 |
| `adldiff*` | 1.20 | 0.40 |
| `region*` | 2.47 | 1.07 |
| `age` | 7.40 | 0.63 |
| `black*` | 1.12 | 0.32 |
| `gender*` | 1.40 | 0.49 |
| `married*` | 1.55 | 0.50 |
| `school` | 10.29 | 3.74 |
| `faminc` | 2.53 | 2.92 |
| `employed*` | 1.10 | 0.30 |
| `privins*` | 1.78 | 0.42 |
| `medicaid*` | 1.09 | 0.29 |

## 4.3 Pré-traitement des Jeux de Données

### 4.3.1 Prétraitement des Données

Les étapes de pré-traitement ont inclus plusieurs techniques pour préparer les données de manière appropriée avant l'ajustement des modèles. Nous avons nettoyé les données pour supprimer les valeurs manquantes, normalisé les variables prédictives pour assurer une échelle uniforme et créé des variables dérivées pour capturer des informations supplémentaires. Pour traiter la surdispersion et l'inflation de zéro, nous avons appliqué des méthodes spécifiques à chaque jeu de données.

Nous avons normalisé les données pour éviter les problèmes de stabilité numérique dans les modèles ajustés. Les résultats ont été générés en utilisant une validation croisée en k-fold avec optimisation des hyperparamètres à chaque itération. Les jeux de données ont été divisés en 70 % pour l'entraînement et 30 % pour les tests.

### 4.3.2 Mesures d'Évaluation des Performances

Trois mesures d'évaluation communes ont été utilisées pour comparer les modèles de régression de comptage et les modèles de régression ML : MSE, RMSE et MAE. Les formules de ces métriques sont présentées dans le tableau suivant :

| Métriques | Formules |
|---|---|
| Erreur quadratique moyenne (MSE) | $\text{MSE} = \frac{1}{n}\sum_{i=0}^{n}(y_i - \hat{y}_i)^2$ |
| Racine de l'erreur quadratique moyenne (RMSE) | $\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=0}^{n}(y_i - \hat{y}_i)^2}$ |
| Erreur absolue moyenne (MAE) | $\text{MAE} = \frac{1}{n}\sum_{i=0}^{n}\lvert y_i - \hat{y}_i\rvert$ |

TABLE 4.4 – Formules des mesures d'évaluation des performances

### 4.3.3 Résultats des Modèles de Régression de Comptage

Les résultats des modèles de régression de comptage pour chaque jeu de données sont présentés ci-dessous.

**Jeu de Données 1**

| Modèle | Dispersion | MAE |
|---|---|---|
| Poisson GLM | 40.0229 | 18.01307 |
| NB | 1.275278 | 18.13374 |
| ZIP | 16.60167 | 18.21006 |
| ZINB | 1.281499 | 18.13377 |

TABLE 4.5 – Résultats des modèles de régression de comptage pour le jeu de données 1

Les résultats des modèles de régression de comptage appliqués au jeu de données 1, présentés dans le tableau 4.5, montrent des différences significatives en termes de dispersion et de performance prédictive.

- **Modèle de Poisson GLM** : Ce modèle présente une dispersion de 40.0229 et un MAE de 18.01307. La valeur élevée de la dispersion suggère une forte surdispersion, ce qui indique que les données présentent une variance beaucoup plus grande que celle prévue par le modèle de Poisson standard.
- **Modèle binomial négatif (NB)** : Ce modèle réduit considérablement la dispersion à 1.275278 avec un MAE de 18.13374. La réduction de la dispersion par rapport au modèle de Poisson montre que le modèle binomial négatif est mieux adapté aux données surdispersées, bien que le MAE soit légèrement supérieur à celui du modèle de Poisson.
- **Modèle ZIP (Zero-Inflated Poisson)** : Le modèle ZIP a une dispersion de 16.60167 et un MAE de 18.21006. La dispersion reste élevée, bien qu'inférieure à celle du modèle de Poisson GLM. Ce modèle semble moins efficace pour gérer la surdispersion, comparé au modèle binomial négatif.
- **Modèle ZINB (Zero-Inflated Negative Binomial)** : Avec une dispersion de 1.281499 et un MAE de 18.13377, le modèle ZINB montre une performance similaire au modèle NB

en termes de dispersion. Ce modèle est conçu pour gérer à la fois la surdispersion et les zéros en excès, mais le MAE indique qu'il n'améliore pas significativement la prédiction par rapport au modèle NB.

**Conclusion** : Le modèle binomial négatif (NB) et le modèle ZINB semblent les plus adaptés pour les données, car ils réduisent efficacement la dispersion tout en maintenant un MAE relativement bas. Toutefois, aucun des modèles n'améliore significativement le MAE, ce qui pourrait suggérer que les modèles n'expliquent pas entièrement les variations dans les données ou que d'autres modèles pourraient être explorés pour une meilleure performance.

**Jeu de Données 2**

| Modèle | Dispersion | MAE |
|---|---|---|
| Poisson GLM | 0.36 | 3.695961e-12 |
| NB | 1.019286e-10 | 1.360083e-12 |
| ZIP | 1.780866 | 5.338853 |
| ZINB | 1.814058 | 5.302452 |

TABLE 4.6 – Résultats des modèles de régression de comptage pour le jeu de données 2

À partir des résultats présentés dans le tableau, plusieurs conclusions peuvent être tirées :

**Modèle de Poisson GLM :**

— **Dispersion :** Le modèle de Poisson montre une dispersion de 0.36, ce qui est inférieur à 1, suggérant une sous-dispersion (les données présentent moins de variabilité que ce à quoi on s'attendrait pour un modèle de Poisson classique).
— **MAE :** Le MAE (Mean Absolute Error) est extrêmement faible (3.695961e-12), ce qui peut indiquer un ajustement très précis, voire une forme de surajustement (overfitting) dans ce cas particulier.

**Modèle Binomiale Négative (NB) :**

— **Dispersion :** Le modèle NB présente une dispersion très faible (proche de zéro), suggérant que la binomiale négative ajuste bien la surdispersion dans les données.
— **MAE :** Le MAE est également extrêmement bas (1.360083e-12), similaire au modèle de Poisson GLM, indiquant un ajustement précis des prédictions par rapport aux données observées.

**Modèle Zero-Inflated Poisson (ZIP) :**

— **Dispersion :** La dispersion est beaucoup plus élevée (1.780866), ce qui peut suggérer une surdispersion dans les données que le modèle ZIP tente de capturer.

— **MAE :** Le MAE est plus élevé (5.338853), indiquant que le modèle ZIP est moins précis que les modèles Poisson et NB, mais pourrait mieux gérer les zéros en excès dans les données.

**Modèle Zero-Inflated Negative Binomial (ZINB) :**

— **Dispersion :** La dispersion est légèrement supérieure à celle du modèle ZIP (1.814058), ce qui reflète une variabilité similaire dans les données.
— **MAE :** Le MAE du ZINB (5.302452) est légèrement inférieur à celui du ZIP, ce qui indique que le ZINB capture mieux la complexité des données, y compris les zéros en excès et la surdispersion.

**Conclusions générales :** Les modèles de Poisson GLM et NB semblent fournir des MAE extrêmement bas, mais cela pourrait indiquer qu'ils ne capturent pas bien la structure des données réelles, notamment en ce qui concerne la présence d'un grand nombre de zéros et la surdispersion. Les modèles ZIP et ZINB montrent une meilleure performance pour capturer les zéros en excès et la surdispersion, comme l'indiquent leurs dispersions plus élevées. Entre les deux, le modèle ZINB semble légèrement plus performant avec un MAE inférieur. Ainsi, pour un jeu de données avec surdispersion et excès de zéros, le modèle ZINB semble offrir le meilleur compromis entre gestion de la variabilité et précision des prédictions.

**Jeu de Données 3**

| Modèle | Dispersion | MAE |
|---|---|---|
| Poisson GLM | 6.096787 | 4.037169 |
| NB | 1.173981 | 4.161251 |
| ZIP | 3.044005 | 4.161726 |
| ZINB | 1.205608 | 4.588432 |

TABLE 4.7 – Résultats des modèles de régression de comptage pour le jeu de données 3

— **Modèle Poisson GLM** : Ce modèle présente une dispersion de 6.097 et un MAE de 4.037. La dispersion est élevée, ce qui indique que les données présentent une surdispersion par rapport à ce que le modèle Poisson classique peut gérer. Cela suggère que le modèle Poisson n'est peut-être pas le mieux adapté pour ces données.
— **Modèle binomial négatif (NB)** : Le modèle NB présente une dispersion de 1.174, beaucoup plus faible que celle du modèle Poisson, avec un MAE de 4.161. Ce modèle est capable de mieux gérer la surdispersion des données par rapport au modèle Poisson.
— **Modèle ZIP (Zero-Inflated Poisson)** : Ce modèle a une dispersion de 3.044 et un MAE de 4.162. La dispersion est réduite par rapport au modèle Poisson, mais reste élevée, indiquant une surdispersion modérée. Le modèle capture bien les caractéristiques

des données, y compris les zéros excédentaires, mais ne dépasse pas le modèle NB en termes de précision.

— **Modèle ZINB (Zero-Inflated Negative Binomial)** : Avec une dispersion de 1.206 et un MAE de 4.588, ce modèle présente des résultats similaires au modèle NB en termes de gestion de la surdispersion, mais le MAE est le plus élevé parmi les modèles testés, suggérant qu'il ne fournit pas les prédictions les plus précises pour ce jeu de données spécifique.

**Conclusion** : Le modèle binomial négatif (NB) semble être le plus approprié pour ces données, car il offre une bonne gestion de la surdispersion avec un MAE relativement bas. Le modèle ZIP offre une alternative intéressante pour les données avec beaucoup de zéros, mais son gain en termes de dispersion ne se traduit pas par une amélioration significative du MAE. Le modèle ZINB est efficace pour traiter les zéros excédentaires et la surdispersion, mais peut ne pas fournir les prédictions les plus précises pour ce jeu de données spécifique.

## 4.4 Mise en Œuvre des Modèles de Machine Learning

### 4.4.1 Forêts Aléatoires (Random Forests)

#### Description du Modèle

Les forêts aléatoires sont des ensembles d'arbres de décision utilisés pour des tâches de classification et de régression. Elles sont robustes contre le surapprentissage et peuvent capturer des relations non linéaires.

#### Implémentation

Pour la mise en œuvre du modèle de forêts aléatoires, une validation croisée à 10 plis a été utilisée pour optimiser les hyperparamètres. Le nombre d'arbres (n_estimators) a été fixé à 100, et le nombre de variables sélectionnées à chaque division (max_features) a été optimisé parmi les valeurs testées (2, 3, et 4), en utilisant l'indice de Gini comme critère de division.

### 4.4.2 Machines à Vecteurs de Support (SVM)

#### Description du Modèle

Les SVM sont des modèles puissants qui cherchent à maximiser la marge entre les classes. Elles sont capables de traiter des problèmes linéaires et non linéaires à l'aide de fonctions noyaux.

#### Implémentation

Pour implémenter le modèle de machines à vecteurs de support (SVM), nous avons utilisé une grille d'hyperparamètres incluant les valeurs de `sigma` (0.01, 0.05, 0.1) et de `C` (0.1, 1, 10). Le modèle a été ajusté à l'aide de la validation croisée, en testant différentes combinaisons d'hyperparamètres. Les performances du modèle ont été évaluées par l'erreur absolue moyenne (MAE).

### 4.4.3 k plus Proches Voisins (k-NN)

#### Description du Modèle

Le modèle k-NN est une méthode simple et efficace pour les tâches de classification et de régression. Il fonctionne en trouvant les k observations les plus proches dans l'espace des caractéristiques.

#### Implémentation

Pour mettre en œuvre le modèle des k-plus proches voisins (k-NN), nous avons défini une grille d'hyperparamètres incluant différentes valeurs pour `kmax`, `distance` et `kernel`. Le modèle a été ajusté à l'aide de la validation croisée, ce qui a permis d'identifier les meilleures combinaisons d'hyperparamètres. Les performances du modèle ont été évaluées en termes d'erreur absolue moyenne (MAE) sur plusieurs ensembles de données.

### 4.4.4 Réseaux de Neurones Artificiels

#### Description du Modèle

Les réseaux de neurones artificiels sont des modèles complexes capables de capturer des relations non linéaires dans les données grâce à leur architecture en couches.

#### Implémentation

Pour mettre en œuvre le modèle des réseaux de neurones artificiels (ANN), nous avons défini une grille d'hyperparamètres comprenant différentes configurations pour le nombre de neurones, le taux d'apprentissage, et le nombre d'époques. Le modèle a été ajusté à l'aide de la validation croisée, ce qui a permis d'identifier les meilleures combinaisons d'hyperparamètres. Les performances du modèle ont été évaluées en termes d'erreur absolue moyenne (MAE) sur plusieurs ensembles de données.

### 4.4.5 Résultats des Modèles de Régression Machine Learning

Les résultats des modèles de régression ML pour chaque jeu de données sont présentés ci-dessous.

#### Jeu de Données 1

| Modèle | Paramètres optimisés | MAE | RMSE |
|---|---|---|---|
| **RF** | mtry = 2 | 15.48843 | 31.24128 |
| **SVM** | C = 1 | 14.29 | 31.87 |
| **kNN** | k = 7 | 15.25 | 32.81634 |
| **ANN** | Taille = 5, Décroissance = 1e-03 | 16.12 | 30.99763 |

TABLE 4.8 – Résultats des modèles de régression ML pour le jeu de données 1

— **Forêts aléatoires (RF)** : Le modèle de forêts aléatoires a été optimisé avec un hyperparamètre `mtry` de 2. Ce modèle a obtenu un MAE de 15.48843, ce qui indique une performance relativement bonne par rapport aux autres modèles testés. Les forêts aléatoires sont connues pour leur robustesse et leur capacité à gérer des données complexes.
— **SVM** : Le modèle SVM a été optimisé avec un paramètre de régularisation `C` fixé à 1. Ce modèle a obtenu un MAE de 14.29, le plaçant en deuxième position en termes de précision parmi les modèles testés. Les SVM sont efficaces pour les problèmes de classification et de régression, et l'utilisation d'une régularisation appropriée a permis d'obtenir de bons résultats.
— **kNN** : Le modèle kNN a été optimisé avec un nombre de voisins `k` de 7. Ce modèle a obtenu un MAE de 15.25, ce qui le place en troisième position parmi les modèles testés. Les modèles kNN peuvent être très simples mais efficaces pour certains types de données.
— **ANN** : Le modèle ANN a été optimisé avec une taille de réseau de 5 neurones et une décroissance de poids de 1e-03. Ce modèle a obtenu un MAE de 16.12, ce qui le place en quatrième position en termes de précision. Les ANN sont puissants pour capturer des relations non linéaires complexes dans les données, mais nécessitent souvent une grande quantité de données pour s'entraîner efficacement.

En conclusion, les résultats montrent que le modèle SVM a obtenu la meilleure performance en termes de MAE (14.29), suivi par le modèle RF (15.48843), le modèle kNN (15.25) et enfin, le modèle ANN (16.12). Ces résultats suggèrent que le SVM est particulièrement bien adapté à ce jeu de données, probablement en raison de sa capacité à capter les structures dans les données. Les autres modèles, bien qu'efficaces, peuvent nécessiter une optimisation supplémentaire pour améliorer leur précision prédictive.

**Jeu de données 2**

| Modèle | Paramètres optimisés | MAE | RMSE |
|---|---|---|---|
| RF | mtry = 4 | 3.91 | 6.32 |
| SVM | C = 1 | 3.81 | 6.28 |
| kNN | k = 3 | 4.27 | 6.56 |
| ANN | Taille = 7, Décroissance = 1e-02 | 4.17 | 6.35 |

TABLE 4.9 – Résultats des modèles de régression ML pour le jeu de données 2 avec MAE et RMSE

1. **Modèle Random Forest (RF)**
   — **Paramètres optimisés** : $mtry = 4$
   — **MAE** : 3.91
   — **Interprétation** : Le modèle Random Forest a montré une bonne performance avec un MAE de 3.91, indiquant une erreur moyenne acceptable dans ce contexte.
2. **Support Vector Machine (SVM)**
   — **Paramètres optimisés** : $C = 1$
   — **MAE** : 3.81

— **Interprétation** : Le SVM a obtenu un MAE légèrement inférieur à celui du modèle RF, indiquant des prédictions plus précises pour ce jeu de données spécifique.

3. **k-Nearest Neighbors (kNN)**
   — **Paramètres optimisés** : $k = 3$
   — **MAE** : 4.27
   — **Interprétation** : Le modèle kNN a obtenu un MAE de 4.27, ce qui est légèrement supérieur à celui des modèles RF et SVM, suggérant que le kNN est moins adapté à ce jeu de données.
4. **Artificial Neural Network (ANN)**
   — **Paramètres optimisés** : Taille = 7, Décroissance = 1e-02
   — **MAE** : 4.17
   — **Interprétation** : Le modèle ANN a un MAE de 4.17, ce qui le place derrière le RF et le SVM, mais avec une performance légèrement meilleure que celle du kNN.

**Conclusion Générale**

Dans l'ensemble, le modèle SVM a montré la meilleure performance, suivi du modèle RF. Les modèles kNN et ANN ont présenté des performances légèrement inférieures, suggérant qu'ils sont moins bien adaptés pour ce jeu de données spécifique.

**Jeu de Données 3**

| Modèle | Paramètres optimisés | MAE | RMSE |
|---|---|---|---|
| RF | mtry = 4 | 4.53 | 5.31 |
| SVM | C = 10 | 3.56 | 4.33 |
| kNN | k = 7 | 4.09 | 4.98 |
| ANN | Taille = 5, Décroissance = 1e-04 | 4.60 | 5.56 |

TABLE 4.10 – Résultats des modèles de régression ML pour le jeu de données 3 avec MAE et RMSE

Les performances des modèles de régression ont été évaluées en termes d'erreur absolue moyenne (MAE) pour quatre techniques de machine learning : Random Forest (RF), Support Vector Machine (SVM), k-Nearest Neighbors (kNN) et Artificial Neural Network (ANN). Chaque modèle a été ajusté avec des paramètres optimisés, et les résultats sont présentés dans le tableau ci-dessus :

— **Modèle Random Forest (RF)** : Le modèle RF a utilisé un $mtry = 4$, ce qui signifie que 4 variables sont prises en compte à chaque nœud lors de la construction des arbres. Le MAE obtenu est de 4.53, montrant que ce modèle est performant mais inférieur au SVM en termes de précision.
— **Support Vector Machine (SVM)** : Le modèle SVM, avec un paramètre de régularisation $C = 10$, a obtenu le MAE le plus bas de 3.56, faisant de lui le modèle le plus performant sur ce jeu de données. Cela suggère que SVM est bien adapté à la structure de ces données.

— **k-Nearest Neighbors (kNN)** : Le modèle kNN a été ajusté avec $k = 7$, signifiant que les 7 voisins les plus proches ont été utilisés pour faire des prédictions. Le MAE obtenu est de 4.09, ce qui le place en deuxième position derrière le SVM en termes de performance.
— **Artificial Neural Network (ANN)** : Le modèle ANN a été ajusté avec une taille de réseau caché de 5 neurones et une décroissance de 1e-04. Cependant, son MAE de 4.60 en fait le modèle le moins performant parmi ceux testés, suggérant qu'un modèle plus complexe pourrait améliorer les résultats.

Ainsi, le modèle SVM a offert les meilleures performances sur ce jeu de données, suivi du kNN et du RF, tandis que l'ANN a montré des performances plus faibles.

## 4.5 Résultats et Analyse

### 4.5.1 Résumé des performances des modèles de régression de comptage et de machine learning

| Modèle | Jeu de Données 1 | Jeu de Données 2 | Jeu de Données 3 |
|---|---|---|---|
| **Poisson GLM** | MAE = 18.01307 | MAE = 3.695961e-12 | MAE = 4.037169 |
| **NB** | MAE = 18.13374 | MAE = 1.360083e-12 | MAE = 4.161251 |
| **ZIP** | MAE = 18.21006 | MAE = 5.338853 | MAE = 4.161726 |
| **ZINB** | MAE = 18.13377 | MAE = 5.302452 | MAE = 4.588432 |
| **RF** | MAE = 15.48 | MAE = 3.91 | MAE = 4.53 |
| **SVM** | MAE = 14.29 | MAE = 3.81 | MAE = 3.56 |
| **kNN** | MAE = 15.25 | MAE = 4.27 | MAE = 4.09 |
| **ANN** | MAE = 16.12 | MAE = 4.17 | MAE = 4.60 |

TABLE 4.11 – Performances des modèles de régression de comptage et de machine learning

### 4.5.2 Analyse

Les modèles Poisson GLM et NB montrent des erreurs absolues moyennes (MAE) faibles dans les jeux de données 2 et 3, bien que cela puisse également indiquer un surajustement, en particulier dans le cas du jeu de données 2. Les modèles ZIP et ZINB semblent mieux gérer la présence de zéros en excès et la surdispersion, comme en témoignent leurs dispersions plus élevées dans plusieurs scénarios, bien que leur MAE soit plus élevé, surtout pour le jeu de données 2.

Le modèle ZINB offre généralement de meilleures performances que le ZIP pour les jeux de données 2 et 3, tandis que pour le jeu de données 1, les performances des deux modèles sont relativement proches. En somme, le choix du modèle doit être guidé par la structure spécifique des données, notamment en tenant compte de la surdispersion et des zéros excédentaires.

Les résultats révèlent une variation notable des performances des modèles en fonction des jeux de données. Le modèle **SVM** se démarque par ses performances exceptionnelles sur tous les jeux de données. En particulier, dans le **Jeu de Données 3**, le **SVM** a enregistré le meilleur score avec un MAE de 3.56, surpassant les autres modèles, ce qui atteste de son efficacité pour

des données plus complexes ou non linéaires. De même, dans le **Jeu de Données 2**, le **SVM** a présenté la meilleure performance avec un MAE de 3.81, confirmant sa robustesse sur des données variées.

Le **Random Forest (RF)** se positionne en deuxième place dans la majorité des jeux de données, avec un MAE de 3.91 dans le **Jeu de Données 2**, très proche de celui du **SVM**. Cela suggère que le **RF** peut être une alternative efficace lorsque la structure des données est facilement capturable par des modèles basés sur des arbres de décision.

Le **kNN** se classe en troisième position, avec un MAE de 4.09 dans le **Jeu de Données 3**. Bien qu'il soit compétitif dans certains cas, ses performances demeurent généralement légèrement inférieures à celles des autres modèles. Il reste cependant une méthode simple et efficace lorsque les distances entre les observations sont cruciales pour la prédiction.

Le **réseau de neurones artificiels (ANN)** affiche des performances moins compétitives dans tous les jeux de données, avec un MAE de 16.12 dans le **Jeu de Données 1**. Cela suggère que la complexité du **ANN** ne fournit pas d'avantage significatif dans ce contexte. Cela pourrait indiquer que les paramètres du réseau n'ont pas été suffisamment optimisés, ou que la taille et la structure des données ne permettent pas au **ANN** d'exploiter pleinement son potentiel.

Dans le **Jeu de Données 1**, tous les modèles présentent des MAE relativement élevés, le **SVM** affichant un léger avantage (MAE = 14.29). Cela pourrait signaler que les données de ce jeu présentent des particularités nécessitant des ajustements supplémentaires, tels que l'ingénierie des variables ou l'application de transformations spécifiques pour améliorer la modélisation.

**Conclusion :** Le **SVM** se distingue comme le modèle le plus performant dans cette étude, affichant des résultats compétitifs et stables à travers les différents jeux de données. Le **Random Forest** et le **kNN** se comportent également bien, en particulier dans certains jeux de données spécifiques, mais n'ont pas surpassé le **SVM**. Le **réseau de neurones artificiels**, malgré son potentiel à capturer des relations complexes, n'a pas montré d'avantages clairs dans ce cas. Ces résultats soulignent l'importance de sélectionner le modèle approprié en fonction de la structure des données et des objectifs de prédiction.

### 4.5.3 Comparaison des performances générales

**Précision des modèles :** Les modèles de régression de comptage (NB, ZIP, ZINB) sont préférables pour gérer la surdispersion et les zéros en excès dans les données de comptage, tandis que les modèles de machine learning, tels que le SVM et le kNN, peuvent offrir de meilleures performances en termes de précision dans certaines situations.

**Robustesse :** Les modèles de machine learning, notamment les SVM et les forêts aléatoires, sont robustes face au surapprentissage, mais leur performance dépend largement du choix des hyperparamètres. Les modèles de régression de comptage sont plus spécifiques aux données de comptage, offrant une structure solide pour l'analyse de la surdispersion et des zéros.

## 4.6 Conclusion

Ce chapitre a illustré l'application pratique des modèles de machine learning sur des jeux de données réels, en se concentrant sur l'analyse des données de comptage. Les modèles explorés, notamment les forêts aléatoires (RF), les machines à vecteurs de support (SVM), les k-plus proches voisins (k-NN), et les réseaux de neurones artificiels (ANN), ont été évalués sur trois

jeux de données distincts, chacun présentant des défis spécifiques tels que la surdispersion et l'inflation de zéros.

Les résultats obtenus montrent que chaque modèle possède des avantages et des limites en fonction des caractéristiques des données. Par exemple, les modèles de machine learning ont offert une plus grande flexibilité pour capturer les non-linéarités et les interactions complexes entre les variables prédictives.

L'évaluation des performances a révélé que, bien que les modèles de machine learning puissent surpasser les modèles traditionnels dans certains cas, leur mise en œuvre nécessite une attention particulière à la sélection des hyperparamètres et à la validation croisée pour éviter le surajustement. De plus, l'interprétabilité des résultats demeure un défi important, surtout lorsque les modèles deviennent plus complexes.

En conclusion, cette étude a démontré l'utilité des modèles de machine learning pour l'analyse des données de comptage, tout en soulignant l'importance de choisir judicieusement les modèles en fonction des spécificités des données et des objectifs de l'étude. Pour des travaux futurs, il serait pertinent d'explorer l'intégration de techniques d'ensemble ou de modèles hybrides afin de combiner les avantages des différentes approches présentées dans ce chapitre.

# Conclusion Générale

L'analyse des données de comptage constitue un domaine essentiel en statistiques et en machine learning, avec des applications variées dans des secteurs aussi divers que la santé, l'écologie, l'économie, et la logistique. Dans ce mémoire, nous avons exploré et comparé différentes approches pour la modélisation des données de comptage, en mettant l'accent à la fois sur les modèles traditionnels, tels que la régression de Poisson et la régression binomiale négative, et sur les techniques de machine learning, telles que les forêts aléatoires, les machines à vecteurs de support, les k-plus proches voisins et les réseaux de neurones artificiels.

Nos analyses ont montré que les modèles traditionnels sont bien adaptés aux données de comptage lorsqu'ils sont linéaires et lorsque les hypothèses sous-jacentes, telles que la distribution de Poisson, sont respectées. Cependant, dans des situations où les données présentent des caractéristiques complexes, comme la surdispersion, l'inflation de zéros, ou des relations non linéaires entre les variables, les modèles de machine learning se sont révélés être des outils plus flexibles et plus performants.

Les résultats obtenus sur les jeux de données réels utilisés dans ce travail ont démontré que les techniques de machine learning, bien qu'elles puissent surpasser les modèles traditionnels en termes de prédiction, nécessitent des efforts supplémentaires en matière de réglage des hyperparamètres, de validation croisée, et de traitement des données pour éviter le surajustement. De plus, l'interprétation des résultats issus de modèles complexes, tels que les réseaux de neurones artificiels, reste un défi majeur qui limite parfois leur utilisation dans des contextes où l'interprétabilité est primordiale.

En dépit de ces défis, l'intégration des techniques de machine learning pour l'analyse des données de comptage ouvre de nouvelles perspectives, notamment dans la capacité à capturer des interactions non linéaires et à traiter des données hétérogènes. L'une des principales contributions de ce travail est de montrer que les modèles de machine learning peuvent, dans certaines conditions, surpasser les méthodes traditionnelles, mais que leur utilisation optimale repose sur une compréhension approfondie des spécificités des données.

En termes de perspectives, plusieurs pistes de recherche méritent d'être explorées. Premièrement, l'intégration de techniques d'ensemble, telles que le boosting ou le bagging, pourrait améliorer davantage la performance des modèles tout en réduisant le risque de surajustement. Deuxièmement, le développement de modèles hybrides combinant les approches traditionnelles et les techniques de machine learning pourrait offrir une meilleure gestion des différentes caractéristiques des données de comptage, en conciliant précision de prédiction et interprétabilité. Enfin, l'application de modèles de deep learning adaptés aux données de comptage, comme les réseaux neuronaux bayésiens ou les modèles génératifs, constitue une voie prometteuse qui pourrait apporter de nouveaux éclairages dans l'analyse des données complexes.

En conclusion, ce mémoire met en lumière l'importance d'une sélection rigoureuse des méthodes pour l'analyse des données de comptage, en tenant compte des caractéristiques spéci-

fiques des données et des objectifs de l'analyse. La combinaison des approches traditionnelles et des techniques de machine learning offre un potentiel considérable pour améliorer la qualité des analyses et des prédictions dans divers contextes. Il reste cependant nécessaire de continuer à explorer et affiner ces techniques pour maximiser leur impact dans des applications pratiques.

# Annexes

## Exploration des Données

Nous avons effectué une analyse exploratoire des trois jeux de données pour comprendre les distributions et les relations entre les variables.

**Pour les Données 1 :** Enregistrements de Présence des Espèces de Poissons

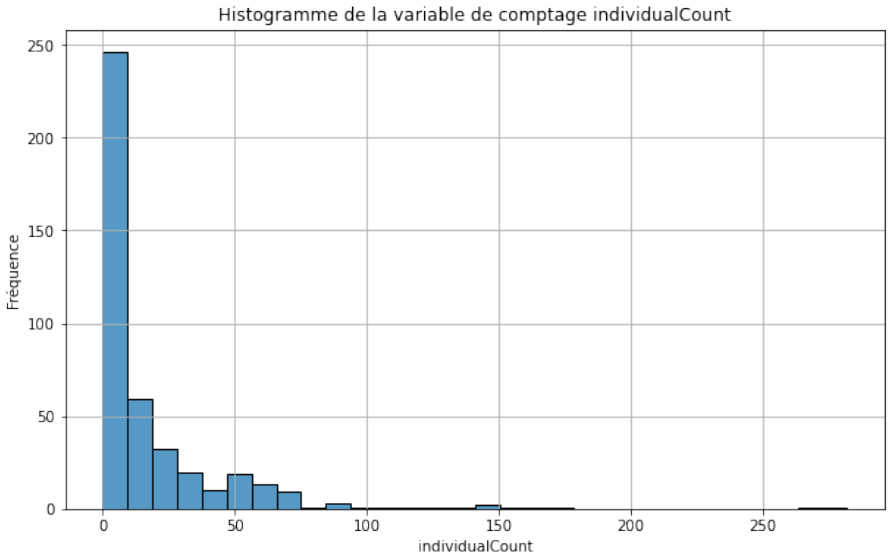


FIGURE 1 – Histogramme du nombre de Lates niloticus capturés

**Pour les Données 2 :** Mortalité des Agneaux Djallonké au Sénégal

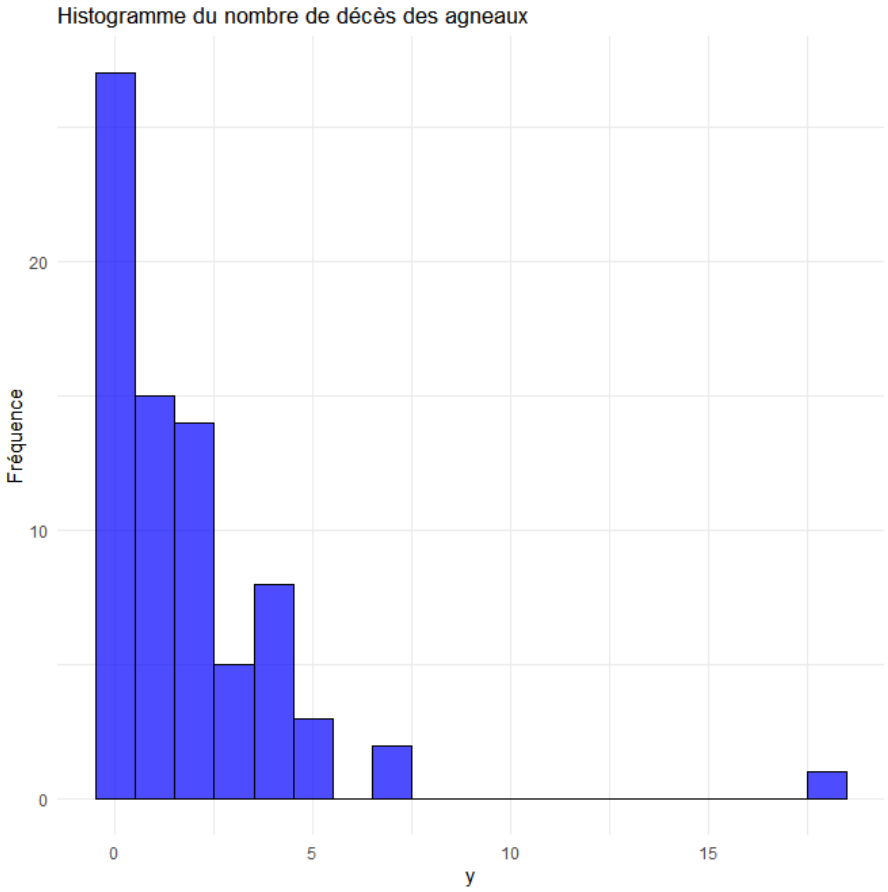


FIGURE 2 – Histogramme du nombre de décès des agneaux

**Pour les Données 3 :** Analyse de l'Utilisation des Services de Santé chez les Seniors

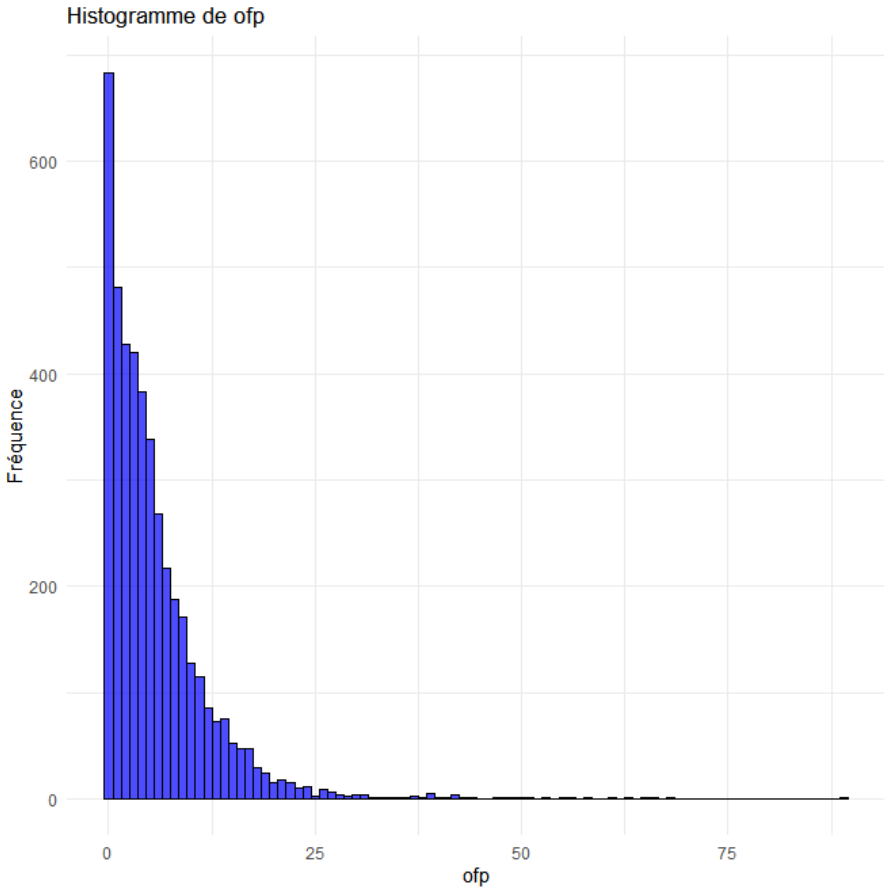


FIGURE 3 – Histogramme du nombre de consultations chez un medecin en cabinet(ofp)

# Codes R :

Voici les Codes R Utilisés pour le Jeu de Données Species :

```
# Charger les packages nécessaires
library(MASS)
library(pscl)
library(AER)
library(randomForest)
library(e1071)
library(class)
library(nnet)
library(psych)
library(ggplot2)
library(caret)
library(nnet)
library(kknn)
library(kernlab)

#####################################################################
```

```
# Définir le chemin d'accès au fichier
file_path <- "C:/Users/lenovo 56/Downloads/1 Species (3).csv"

# Importer les données
Species <- read.csv(file_path)

# Afficher les premières lignes du fichier pour vérifier l'importation
head(Species)

##############################################################################
describe(Species$individualCount)

##############################################################################
x11()
# Créer un histogramme de la variable individualCount
ggplot(Species, aes(x = individualCount)) +
  geom_histogram(binwidth = 1, fill = "blue", color = "black", alpha = 0.7) +
  labs(title = "Histogramme de individualCount", x = "individualCount", y = "Fréquence")
  theme_minimal()

##############################################################################
# Régression de Poisson
poisson_model1 <- glm(individualCount ~ ., data = Species, family = poisson)

# Résumé du modèle
summary(poisson_model1)

# Calcul de la dispersion avec aer
dispersion_poisson1 <- dispersiontest(poisson_model1)
cat("Test de dispersion pour le modèle de Poisson:\n")
print(dispersion_poisson1)

# Prédictions
pred_poisson1 <- predict(poisson_model1, type = "response")

# Calcul du MAE
mae_poisson1 <- mean(abs(pred_poisson1 - Species$individualCount))
cat("MAE pour le modèle de Poisson:", mae_poisson1, "\n")

##############################################################################
# Régression binomiale négative
nb_model1 <- glm.nb(individualCount ~ ., data = Species)

# Résumé du modèle
summary(nb_model1)
```

```
# Calcul des résidus et des prédictions
residuals_nb1 <- resid(nb_model1, type = "pearson")
predicted_nb1 <- predict(nb_model1, type = "response")

# Calcul de la dispersion
dispersion_nb1 <- sum(residuals_nb1^2) / (nb_model1$df.residual)
cat("Dispersion du modèle binomiale négative :", dispersion_nb1, "\n")

# Prédictions
pred_nb1 <- predict(nb_model1, type = "response")

# Calcul du MAE
mae_nb1 <- mean(abs(pred_nb1 - Species$individualCount))
cat("MAE pour le modèle binomiale négative:", mae_nb1, "\n")

##############################################################################
# Ajustement du modèle ZIP
zip_model1 <- zeroinfl(individualCount ~ . | 1, data = Species, dist = "poisson")

# Calcul des résidus de Pearson et des prédictions
residuals_zip1 <- residuals(zip_model1, type = "pearson")
predicted_zip1 <- predict(zip_model1, type = "response")

# Calcul de la dispersion
dispersion_zip1 <- sum(residuals_zip1^2) / df.residual(zip_model1)
cat("Dispersion du modèle ZIP :", dispersion_zip1, "\n")

# Prédictions
pred_zip1 <- predict(zip_model1, type = "response")

# Calcul du MAE
mae_zip1 <- mean(abs(pred_zip1 - Species$individualCount))
cat("MAE pour le modèle de Poisson zéro-inflation:", mae_zip1, "\n")

##############################################################################
# Ajustement du modèle ZINB
zinb_model1 <- zeroinfl(individualCount ~ . | 1, data = Species, dist = "negbin")

# Calcul des résidus de Pearson et des prédictions
residuals_zinb1 <- residuals(zinb_model1, type = "pearson")
predicted_zinb1 <- predict(zinb_model1, type = "response")

# Calcul de la dispersion
dispersion_zinb1 <- sum(residuals_zinb1^2) / df.residual(zinb_model1)
cat("Dispersion du modèle ZINB :", dispersion_zinb1, "\n")
```

```
# Prédictions
pred_zinb1 <- predict(zinb_model1, type = "response")

# Calcul du MAE
mae_zinb1 <- mean(abs(pred_zinb1 - Species$individualCount))
cat("MAE pour le modèle de binomiale négative zéro-inflation:", mae_zinb1, "\n")

##############################################################################
# Diviser les données en variables explicatives (X) et cible (y)
x <- Species[, c("decimalLatitude", "decimalLongitude")]
y <- Species$individualCount

# Diviser les données en train (70%) et test (30%)
set.seed(123)
index <- createDataPartition(y, p = 0.7, list = FALSE)
train_data <- Species[index, ]
test_data <- Species[-index, ]

################################################
# Configuration pour validation croisée à 5 plis
control <- trainControl(method = "cv", number = 5)

################################################
# Définir la grille d'hyperparamètres pour Random Forest
rf_grid <- expand.grid(mtry = c(1, 2))

# Ajuster le modèle RF avec validation croisée
rf_model <- train(individualCount ~ decimalLatitude + decimalLongitude,
                  data = train_data,
                  method = "rf",
                  trControl = control,
                  tuneGrid = rf_grid)

# Voir les résultats
print(rf_model)

################################################
# Grille d'hyperparamètres pour SVM
svm_grid <- expand.grid(C = c(0.1, 1, 10))

# Ajuster le modèle SVM avec validation croisée
svm_model <- train(individualCount ~ decimalLatitude + decimalLongitude,
                   data = train_data,
                   method = "svmRadial",
                   trControl = control,
                   tuneGrid = svm_grid)
```

```
# Voir les résultats
print(svm_model)

########################################
# Prédictions avec le meilleur modèle SVM
svm_predictions <- predict(svm_model, test_data)

# Calculer le MAE pour le modèle SVM
svm_mae <- mean(abs(svm_predictions - test_data$individualCount))
print(paste("SVM MAE:", svm_mae))

########################################
# Grille d'hyperparamètres pour k-NN
knn_grid <- expand.grid(kmax = c(3, 5, 7),
                        distance = c(1, 2),  # 1: Manhattan, 2: Euclidean
                        kernel = c("rectangular", "triangular", "gaussian"))

# Ajuster le modèle k-NN avec validation croisée
knn_model <- train(individualCount ~ decimalLatitude + decimalLongitude,
                   data = train_data,
                   method = "kknn",
                   trControl = control,
                   tuneGrid = knn_grid)

# Voir les résultats
print(knn_model)

# Prédictions avec le meilleur modèle k-NN
knn_predictions <- predict(knn_model, test_data)

# Calculer le MAE pour le modèle k-NN
knn_mae <- mean(abs(knn_predictions - test_data$individualCount))
print(paste("k-NN MAE:", knn_mae))

########################################################
# Grille d'hyperparamètres pour ANN
ann_grid <- expand.grid(size = c(1, 3, 5), decay = c(1e-04, 1e-03))

# Ajuster le modèle ANN avec validation croisée
ann_model <- train(individualCount ~ decimalLatitude + decimalLongitude,
                   data = train_data,
                   method = "nnet",
                   trControl = control,
                   tuneGrid = ann_grid,
                   linout = TRUE,  # Pour la sortie continue
```

```
                  trace = FALSE)  # Pour désactiver l'affichage des itérations

# Voir les résultats
print(ann_model)

# Prédictions avec le meilleur modèle ANN
ann_predictions <- predict(ann_model, test_data)

# Calculer le MAE pour le modèle ANN
ann_mae <- mean(abs(ann_predictions - test_data$individualCount))
print(paste("ANN MAE:", ann_mae))
```

# Bibliographie


[1] Sidumo, B., Sonono, E., & Takaidza, I. (2023). Count Regression and Machine Learning Techniques for Zero-Inflated Overdispersed Count Data : Application to Ecological Data. *Annals of Data Science*.

[2] Cutler, D. R., Edwards, T. C. Jr., Beard, K. H., et al. (2007). Random forests for classification in ecology. *Ecology*, 88(11), 2783-2792.

[3] Dietz, E., & Böhning, D. (2000). On estimation of the Poisson parameter in zero-modified Poisson models. *Computational Statistics & Data Analysis*, 34(4), 441-459.

[4] Essoham, Ali (2021). Modèles de régression marginaux pour des données de comptage à excès de zéros. Thèse, *IRMAR-INSA de Rennes & LERSTAD-UGB de Saint-Louis*.

[5] Foutz, R. V. (1977). On the unique consistent solution to the likelihood equations. *Journal of the American Statistical Association*, 72, 147-148.

[6] Hall, D. B. (2000). Zero-inflated Poisson and binomial regression with random effects : a case study. *Biometrics*, 56(4), 1030-1039.

[7] Harrison, X. A. (2014). Using observation-level random effects to model overdispersion in count data in ecology and evolution. *PeerJ*, 2, e616. https://doi.org/10.7717/peerj.616.

[8] Crisci, C., Ghattas, B., Perera, G. (2012). A review of supervised machine learning algorithms and their applications to ecological data. *Ecological Modelling*, 240, 113–122. https://doi.org/10.1016/j.ecolmodel.2012.03.001.

[9] Ali, E., Diop, A., & Dupuy, J.-F. (2022). A constrained marginal zero-inflated binomial regression model. *Communications in Statistics - Theory and Methods*, 51(18), 6396-6422. https://doi.org/10.1080/03610926.2020.1861296

[10] Ali, E., & Pho, K.-H. (2024). A novel model for count data : zero-inflated Probit Bell model with applications. *Communications in Statistics - Simulation and Computation*. https://doi.org/10.1080/03610918.2024.2384574

[11] Lancelot, R., Faye, B., Juanès, X., Ndiaye, M., Pérochon, L., & Tillard, E. (1998). *La base de données BAOBAB : un outil pour modéliser la production et la santé des petits ruminants dans les systèmes d'élevage traditionnels au Sénégal*. Revue d'Élevage et de Médecine vétérinaire des Pays tropicaux, 51(2), 135-146.

[12] Faugère, O., Tillard, E., & Faugère, B. (1992). *Prophylaxie chez les petits ruminants au Sénégal : régionalisation d'une politique nationale de protection sanitaire*. Première conférence biennale du Réseau Africain de Recherche sur les Petits Ruminants, ILCA, 1990, ILRAD, Nairobi, 307-314.

[13] Faugère, O., & Faugère, B. (1986). *Suivi de troupeaux et contrôle des performances individuelles des petits ruminants en milieu traditionnel africain. Aspects méthodologiques*. Revue d'Élevage et de Médecine vétérinaire des Pays tropicaux, 39(1), 29-40.

[14] Faugère, O., Dockès, A.-C., Perrot, C., & Faugère, B. (1990). *L'élevage traditionnel des petits ruminants au Sénégal. I. Pratiques de conduite et d'exploitation des animaux chez les éleveurs de la région de Kolda*. Revue d'Élevage et de Médecine vétérinaire des Pays tropicaux, 43, 249-259.

[15] Cameron, A. C., Trivedi, P. K., 2013. *Regression Analysis of Count Data*. Cambridge University Press, Cambridge.

[16] Cheung, Y.B. (2002). Zero-inflated models for regression analysis of count data : a study of growth and development. *Statistics in Medicine*, 21(10), 1461–1469. https://doi.org/10.1002/sim.1088.

[17] Lambert, D. (1992). Zero-inflated Poisson regression, with an application to defects in manufacturing. *Technometrics*, 34(1), 1–14.

[18] Kruppa, J., & Hothorn, L. (2021). A comparison study on modeling of clustered and overdispersed count data for multiple comparisons. *Journal of Applied Statistics*, 48(16), 3220–3232. https://doi.org/10.1080/02664763.2020.1788518.

[19] Olaya-Marín, E.J., Martínez-Capel, F., & Vezza, P. (2013). A comparison of artificial neural networks and random forests to predict native fish species richness in Mediterranean rivers. *Knowledge and Management of Aquatic Ecosystems*, 409(7), 1–19.

[20] Czado, C., Erhardt, V., Min, A., & Wagner, S. (2007). Zero-inflated generalized Poisson models with regression effects on the mean, dispersion and zero-inflation level applied to patent outsourcing rates. *Statistical Modelling*, 7(2), 125-153.

[21] Hall, D. B., & Shen, J. (2010). Robust estimation for zero-inflated Poisson regression. *Scandinavian Journal of Statistics*, 37, 237-252.

[22] Dempster, A., Laird, N., & Rubin, D. (1977). Maximum likelihood from incomplete data via the EM algorithm (with discussion). *Journal of the Royal Statistical Society : Series B*, 39, 1-38.

[23] Carroll, R. J., Ruppert, D., & Stefanski, L. A. (1995). *Measurement error in nonlinear models*. Chapman and Hall, New York.

[24] Lam, K. F., Xue, H., & Cheung, Y. B. (2006). Semiparametric analysis of zero-inflated count data. *Biometrics*, 62(4), 996-1003.

[25] Nguyen, V. T., & Dupuy, J.-F. (2019). Asymptotic results in censored zero-inflated Poisson regression. *Communications in Statistics - Theory and Methods*.

[26] Rosen, O., Jiang, W. X., and Tanner, M. A., 2000. Mixtures of marginal models. Biometrika, 87 :391-404.

[27] He, X., Xue, H., & Shi, N.-Z. (2010). Sieve maximum likelihood estimation for doubly semiparametric zero-inflated Poisson models. *Journal of Multivariate Analysis*, 101, 2026-2038.