

Article

Random Forest Adaptation for High-Dimensional Count Regression

Oyebayo Ridwan Olaniran ^{1,†} , Saidat Fehintola Olaniran ^{2,†} , Ali Rashash R. Alzahrani ^{3,*,†} ,
Nada MohammedSaeed Alharbi ⁴  and Asma Ahmad Alzahrani ⁵ 

¹ Department of Statistics, Faculty of Physical Sciences, University of Ilorin, Ilorin 1515, Nigeria; olaniran.or@unilorin.edu.ng

² Department of Statistics and Mathematical Sciences, Faculty of Pure and Applied Sciences, Kwara State University, Malete 1530, Nigeria; saidat.olaniran@kwasu.edu.ng

³ Mathematics Department, Faculty of Sciences, Umm Al-Qura University, Makkah 24382, Saudi Arabia

⁴ Department of Mathematics, Faculty of Science, Taibah University, Al-Madinah Al-Munawara 42353, Saudi Arabia; nmharbi@taibahu.edu.sa

⁵ Department of Mathematics, Faculty of Science, Al-Baha University, Al-Baha 65779, Saudi Arabia; asmaalzahrani@bu.edu.sa

* Correspondence: arrzahrani@uqu.edu.sa

† These authors contributed equally to this work.

Abstract

The analysis of high-dimensional count data presents a unique set of challenges, including overdispersion, zero-inflation, and complex nonlinear relationships that traditional generalized linear models and standard machine learning approaches often fail to adequately address. This study introduces and validates a novel Random Forest framework specifically developed for high-dimensional Poisson and Negative Binomial regression, designed to overcome the limitations of existing methods. Through comprehensive simulations and a real-world genomic application to the Norwegian Mother and Child Cohort Study, we demonstrate that the proposed methods achieve superior predictive accuracy, quantified by lower root mean squared error and deviance, and critically produced exceptionally stable and interpretable feature selections. Our theoretical and empirical results show that these distribution-optimized ensembles significantly outperform both penalized-likelihood techniques and naive-transformation-based ensembles in balancing statistical robustness with biological interpretability. The study concludes that the proposed frameworks provide a crucial methodological advancement, offering a powerful and reliable tool for extracting meaningful insights from complex count data in fields ranging from genomics to public health.

Keywords: high-dimensional analysis; count regression; Random Forest; overdispersion; zero-inflation; genomic analysis

MSC: 94A08; 68U10



Academic Editor: Andreas Artemiou

Received: 23 August 2025

Revised: 14 September 2025

Accepted: 16 September 2025

Published: 21 September 2025

Citation: Olaniran, O.R.; Olaniran, S.F.; Alzahrani, A.R.R.; Alharbi, N.M.; Alzahrani, A.A. Random Forest Adaptation for High-Dimensional Count Regression. *Mathematics* **2025**, *13*, 3041. <https://doi.org/10.3390/math13183041>

Copyright: © 2025 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Analysis of count data is a fundamental challenge in statistical modeling, with applications pervasive across numerous disciplines, including epidemiology [1], genomics [2], insurance analytics [3], and environmental science [4]. Such data, which enumerate the occurrence of events within a fixed domain, are intrinsically discrete and non-negative, and frequently exhibit overdispersion [5]. These characteristics necessitate the development of

specialized regression frameworks that extend beyond the capabilities of standard linear models [6]. The complexity of the problem is amplified in high-dimensional settings, where the number of potential predictors, p , vastly exceeds the sample size, n [7]. In these contexts, traditional methodologies are plagued by issues of multicollinearity, overfitting, and difficulties in variable selection. This is particularly the case when the feature set comprises high-throughput data such as genomic markers or textual features [8]. While the Poisson distribution has served as the foundational model for count data due to its property of equidispersion, this assumption is routinely violated in practice [9]. This has led to the widespread adoption of the Negative Binomial model, which incorporates a dispersion parameter to account for extra-Poisson variation [10]. Consequently, high-dimensional count regression presents a unique set of challenges that include robust estimation amidst sparse signals, managing correlated predictor structures, and accommodating potential nonlinearities that standard parametric forms may fail to capture [11,12].

These challenges are particularly pronounced in fields like biomedical research, where modeling high-throughput omics or RNA-seq count data with thousands of covariates demands methods that overcome the curse of dimensionality while preserving interpretability and accuracy, a task further complicated by zero inflation which necessitates hybrid models like Zero-Inflated or hurdle regressions [13]. This need for advanced, scalable modeling strategies is equally critical in insurance analytics, for predicting claim counts from vast policyholder variables, and in ecology, for monitoring species abundance with spatially–temporally dependent remote sensing data, where the high-dimensional reality of $p \gg n$ introduces estimation instability and spurious associations. While the Negative Binomial model handles overdispersion, the core challenge extends to managing complex interactions and nonlinearities beyond the scope of standard GLMs, requiring methods grounded in probabilistic principles for reliable inference [14]. To address these issues, a variety of dimension-reduction techniques are employed: from unsupervised Principal Component Analysis (PCA), which risks ignoring the response, to supervised alternatives like Partial Least Squares (PLS) that leverage covariance; from simple but limited univariate screening to more comprehensive, iterative approaches like recursive feature elimination. All these methods strive to balance computational tractability with model performance in high-dimensional settings [15].

Embedded feature selection methods integrate variable selection into model estimation, with approaches such as Sure Independence Screening (SIS) showing to be effective in ultrahigh-dimensional count settings such as GWAS [16,17]. In contrast, wrapper methods are computationally demanding, and filter methods (e.g., Relief) are robust to interactions but not tailored to count data [18], motivating hybrid approaches that balance efficiency with distributional considerations.

Among penalized likelihood methods, the Lasso [19] performs simultaneous selection and estimation, but suffers from coefficient bias and instability with correlated predictors. Variants such as the adaptive Lasso address these limitations and achieve oracle properties in Poisson and Negative Binomial models [20,21]. Ridge regression [22] mitigates multicollinearity through L2 shrinkage but lacks sparsity, while the Elastic-Net combines L1 and L2 penalties to encourage both sparsity and grouping, benefiting applications with correlated predictors such as genes in biological pathways [23]. Penalization strategies have also been extended to Zero-Inflated models, enabling simultaneous selection in both count and inflation components [24].

Recent advances have produced specialized penalized methods for structured data. The group Lasso [25] applies a penalty to predefined groups of coefficients (e.g., all dummy variables for a categorical factor or all genes in a pathway), enabling group-wise selection. This is invaluable in functional genomics. The fused Lasso [26], another variant, pro-

motes smoothness by penalizing the differences between coefficients of ordered predictors (e.g., in time series or spatial data), facilitating the detection of trends. For Negative Binomial responses, these penalized approaches are adapted to jointly estimate the dispersion parameter alongside the regression coefficients within a regularized likelihood framework. Supported by strong theoretical guarantees on consistency and selection accuracy, these techniques constitute a robust toolkit for high-dimensional count regression. A fundamental limitation they share, however, is the assumption of a linear predictor in the log-mean, which can restrict their flexibility in capturing complex nonlinear relationships [27].

Decision trees offer a nonparametric alternative by recursively partitioning the predictor space to create homogeneous subgroups. The splitting criteria can be tailored for count distributions, such as by minimizing Poisson deviance, allowing trees to inherently capture interactions and nonlinearities without assuming a specific additive form. This flexibility is highly beneficial in applications such as predicting hospital readmission counts from diverse patient records. A critical weakness of single trees is their propensity to overfit, especially in noisy, high-dimensional data, which necessitates post-pruning strategies to balance bias and variance [28].

To overcome the limitations of single trees, ensemble methods are employed. Random forests [29–31] aggregate a multitude of decision trees, each built on a bootstrapped sample and a random subset of predictors in each split. This process de-correlates the trees, mitigates overfitting, and performs implicit feature selection, making it exceptionally powerful for high-dimensional count regression where $p \gg n$, such as in metagenomics. For count data, the algorithm can use distribution-specific splitting rules (e.g., Poisson deviance reduction) and provide variable importance measures. Extensions to Negative Binomial responses can incorporate dispersion estimation to handle overdispersion effectively. Similarly, boosting ensembles like gradient boosting machines iteratively construct trees focused on the residuals of previous models, optimizing the Poisson or Negative Binomial log-likelihood. Modern implementations like XGBoost incorporate additional regularization and sophisticated handling of missing data, making them highly effective for tasks like insurance claim forecasting. Although these nonparametric ensembles often surpass traditional methods in predictive performance by capturing complex patterns, they require careful hyperparameter tuning via cross-validation to achieve optimal results [32,33].

The motivation for this work arises from the widespread challenges of analyzing high-dimensional count data, where traditional generalized linear models often struggle with instability, overfitting, and inadequate capture of nonlinear dependencies [9]. In domains such as genomics and epidemiology, datasets frequently contain thousands of features but limited samples, requiring approaches that are both scalable and flexible enough to capture complex associations without restrictive parametric assumptions [12]. Penalized regressions address sparsity but impose linearity on the log-mean and may overlook higher-order interactions, while dimension-reduction techniques risk discarding informative variation. Nonparametric alternatives such as decision trees provide flexibility, though single trees suffer from instability. Ensemble methods like Random Forests offer improved predictive performance and robustness [29], yet standard implementations are not naturally suited to discrete, overdispersed responses, highlighting the need for adaptations that respect the distributional characteristics of count data.

The primary objective of this study is to develop and rigorously evaluate a novel Random Forest framework specifically designed for high-dimensional count regression. This framework will adapt the splitting and aggregation mechanisms to be optimal for both Poisson and Negative Binomial responses. Through comprehensive simulations and real-world data applications, we demonstrate its superiority in overcoming the limitations inherent in both existing parametric penalized methods and standard nonparametric en-

semble techniques. This study contributes by introducing a Random Forest algorithm tailored to Poisson and Negative Binomial outcomes, incorporating deviance-based splitting criteria and fixed-dispersion estimation to enhance efficiency in high-dimensional settings. We establish theoretical properties through consistency results and benchmark performance against penalized regression and boosting methods, demonstrating improved accuracy and variable selection. Finally, via extensive simulations and applications to bioinformatics datasets, we show practical utility and provide an open-source R implementation to facilitate adoption in research and applied settings.

2. Mathematical Foundation

2.1. Model Specification and Splitting Criteria

We operate within the framework of generalized regression trees. Consider a high-dimensional dataset comprising n independent and identically distributed observations, $\{(\mathbf{x}_i, y_i)\}_{i=1}^n$, where the feature vector $\mathbf{x}_i \in \mathcal{X} \subset \mathbb{R}^p$ resides in a p -dimensional space (with the possibility $p \gg n$) and the response $y_i \in \mathbb{N}_0$ is a count-valued random variable.

The proposed method constructs an ensemble of B randomized and bootstrapped regression trees, each providing a piecewise constant estimate of a complex regression function. For the Poisson case, we formalize the data-generating process by assuming the conditional distribution of the response belongs to the Poisson family:

$$Y_i | \mathbf{X} = \mathbf{x}_i \sim \text{Poisson}(\mu(\mathbf{x}_i)), \quad i = 1, \dots, n \quad (1)$$

$$\text{where } \mu(\mathbf{x}_i) = \mathbb{E}[Y_i | \mathbf{X} = \mathbf{x}_i] = \exp(f(\mathbf{x}_i)). \quad (2)$$

Here, $f : \mathbb{R}^p \rightarrow \mathbb{R}$ is an unknown, potentially highly complex and nonlinear regression function that the Random Forest aims to estimate nonparametrically. The choice of the log link function ensures the predicted mean $\mu(\mathbf{x}_i)$ remains positive, a necessity for count data.

The likelihood function for a Poisson model given a set of parameters is:

$$\mathcal{L}(\boldsymbol{\mu}; \mathbf{y}) = \prod_{i=1}^n \frac{\mu_i^{y_i} e^{-\mu_i}}{y_i!}. \quad (3)$$

Consequently, the log-likelihood for a sample of n observations is given by:

$$\ell(\boldsymbol{\mu}; \mathbf{y}) = \sum_{i=1}^n [y_i \log(\mu_i) - \mu_i - \log(y_i!)]. \quad (4)$$

Within the context of a decision tree, the predictor space $\mathcal{X}$ is partitioned into M disjoint regions $R_1, R_2, \dots, R_M$. The model assumes a constant mean μ_m within each region R_m . Therefore, for any $\mathbf{x} \in R_m$, the estimate is $\hat{\mu}(\mathbf{x}) = \hat{\mu}_m$.

2.1.1. Splitting Criterion for Poisson Regression Trees

For a given node t containing n_t observations, the maximum likelihood estimate (MLE) of the Poisson rate is the empirical mean:

$$\hat{\mu}_t = \frac{1}{n_t} \sum_{i \in t} y_i. \quad (5)$$

The goodness-of-fit for this node is measured by its deviance, which is twice the negative log-likelihood of the node's data under its fitted model, up to an additive constant involving the factorial term, which is independent of the model:

$$D(t) = -2 \ell(\hat{\mu}_t; \mathbf{y}_t) = 2 \sum_{i \in t} \left[\hat{\mu}_t - y_i + y_i \log \left(\frac{y_i}{\hat{\mu}_t} \right) \right]. \quad (6)$$

The goal of the recursive partitioning algorithm is to find the binary split s of node t into left and right daughter nodes t_L and t_R that leads to the largest possible improvement in model fit. This improvement is quantified by the decrease in deviance:

$$\begin{aligned} \Delta D(s, t) &= D(t) - (D(t_L) + D(t_R)) \\ &= 2 \left[\sum_{i \in t} \left(y_i \log \left(\frac{y_i}{\hat{\mu}_t} \right) - (y_i - \hat{\mu}_t) \right) - \sum_{d \in \{L, R\}} \sum_{i \in t_d} \left(y_i \log \left(\frac{y_i}{\hat{\mu}_{t_d}} \right) - (y_i - \hat{\mu}_{t_d}) \right) \right]. \end{aligned} \quad (7)$$

The algorithm evaluates a large set of candidate splits (across randomly selected features and all possible split points) and selects the split s^* that maximizes $\Delta D(s, t)$:

$$s^* = \operatorname{argmax}_{s \in \mathcal{S}} \Delta D(s, t), \quad (8)$$

where $\mathcal{S}$ is the set of all permissible splits. This criterion is equivalent to maximizing the likelihood ratio statistic for comparing the two models (a single mean vs. two separate means) implied by the split.

2.1.2. Extension to the Negative Binomial Model

The Poisson model assumes equidispersion ($\mathbb{E}[Y] = \operatorname{Var}(Y)$), an assumption frequently violated in real-world data due to overdispersion. The Negative Binomial (NB) distribution provides a robust alternative by introducing a dispersion parameter, $\theta > 0$. We posit the following data-generating process:

$$\begin{aligned} Y_i | \mathbf{X} = \mathbf{x}_i &\sim \text{NB}(\mu(\mathbf{x}_i), \theta), \\ \text{where } \mathbb{E}[Y_i | \mathbf{X} = \mathbf{x}_i] &= \mu(\mathbf{x}_i), \\ \operatorname{Var}(Y_i | \mathbf{X} = \mathbf{x}_i) &= \mu(\mathbf{x}_i) + \frac{\mu(\mathbf{x}_i)^2}{\theta}, \\ \log(\mu(\mathbf{x}_i)) &= f(\mathbf{x}_i). \end{aligned}$$

The probability mass function can be parameterized in terms of μ and θ :

$$P(Y_i = y_i | \mu_i, \theta) = \frac{\Gamma(y_i + \theta)}{\Gamma(\theta) y_i!} \left(\frac{\theta}{\theta + \mu_i} \right)^\theta \left(\frac{\mu_i}{\theta + \mu_i} \right)^{y_i}. \quad (9)$$

The MLE for μ_t within a node remains the sample mean $\hat{\mu}_t = \frac{1}{n_t} \sum_{i \in t} y_i$. However, estimating θ locally within every node during the tree-building process is computationally prohibitive, especially in high-dimensional settings, and can lead to instability. The adopted computationally efficient strategy involves a two-stage process: (1) **Global Dispersion Estimation**: Prior to building the forest, estimate a global dispersion parameter, $\hat{\theta}$, by fitting a simple NB regression model to the entire dataset. Often, this is performed using an intercept-only model: $\hat{\theta} = \operatorname{argmax}_{\theta} \ell(\hat{\mu}, \theta; \mathbf{y})$. Here, $\hat{\mu}$ is the global mean. (2) **Fixed-Dispersion Splitting**: During the construction of each tree, this global $\hat{\theta}$ is held fixed. The

splitting criterion then involves finding the split, s , that maximizes the decrease in the NB deviance, defined as $-2\ell(\hat{\mu}_t, \hat{\theta}; \mathbf{y}_t)$ for a node. The decrease for a candidate split is:

$$\Delta D_{\text{NB}}(s, t) = -2\ell(\hat{\mu}_t, \hat{\theta}; \mathbf{y}_t) + 2[\ell(\hat{\mu}_{t_L}, \hat{\theta}; \mathbf{y}_{t_L}) + \ell(\hat{\mu}_{t_R}, \hat{\theta}; \mathbf{y}_{t_R})]. \quad (10)$$

This approach decouples the dispersion estimation from the tree growth, dramatically improving computational efficiency while still accounting for overdispersion in the splitting logic.

2.1.3. Ensemble Prediction and Estimator Definition

The final model is an ensemble of B such randomized trees. Each tree, T_b , built on a bootstrapped sample and with feature randomization at each split, produces a prediction $\hat{\mu}_b(\mathbf{x})$ for a new point $\mathbf{x}$. This prediction is the fitted value of the leaf node in tree, T_b , where $\mathbf{x}$ resides, i.e., $\hat{\mu}_b(\mathbf{x}) = \hat{\mu}_{R_m(b)}$.

The Random Forest predictor is the ensemble average across all trees:

$$\hat{\mu}(\mathbf{x}) = \frac{1}{B} \sum_{b=1}^B \hat{\mu}_b(\mathbf{x}). \quad (11)$$

This aggregation reduces variance by leveraging the wisdom of the crowd. The prediction can also be understood as a kernel-weighted average:

$$\hat{\mu}(\mathbf{x}) = \sum_{i=1}^n \alpha_i(\mathbf{x}) y_i, \quad (12)$$

where the weights, $\alpha_i(\mathbf{x})$, are determined by the number of times the observation, i , appears in the same leaf as $\mathbf{x}$ across all trees in the forest, normalized by the number of trees. This highlights the nonparametric nature of the estimator.

2.2. Theoretical Properties

We study consistency and rates for Poisson (and fixed-dispersion Negative Binomial) Random Forests constructed with deviance-based splits. Throughout, the predictor space is $\mathcal{X} = [0, 1]^p$ with the usual Euclidean metric and the response $Y \in \mathbb{N}_0$.

2.2.1. Model

Let $(\mathbf{X}, Y)$ satisfy

$$Y \mid \mathbf{X} = \mathbf{x} \sim \text{Poisson}(\mu_0(\mathbf{x})), \quad \mu_0(\mathbf{x}) = \exp\{f_0(\mathbf{x})\},$$

where $f_0 : \mathcal{X} \rightarrow \mathbb{R}$ is unknown.

2.2.2. Forest Construction

A single tree is built by recursive binary partitioning using the Poisson deviance as impurity:

$$\mathcal{D}(\mathcal{A}) = 2 \sum_{i: \mathbf{x}_i \in \mathcal{A}} \left\{ Y_i \log \frac{Y_i}{\hat{\mu}_{\mathcal{A}}} - (Y_i - \hat{\mu}_{\mathcal{A}}) \right\}, \quad \hat{\mu}_{\mathcal{A}} = \frac{1}{|\mathcal{A}|} \sum_{i: \mathbf{x}_i \in \mathcal{A}} Y_i,$$

with the convention $0 \log 0 = 0$. At each node, $\mathcal{A}$, we sample m try coordinates uniformly from $\{1, \dots, p\}$ and choose the coordinate and threshold maximizing the decrease in deviance (ties broken arbitrarily). Splitting stops when a node contains fewer than k_n

observations. Each tree predictor is the leaf-wise sample mean, $\hat{\mu}_T(\mathbf{x})$; the forest predictor averages B i.i.d. trees:

$$\hat{\mu}_n(\mathbf{x}) = \frac{1}{B} \sum_{b=1}^B \hat{\mu}_{T_b}(\mathbf{x}).$$

2.2.3. Assumptions

We impose the following conditions.

- (A1) **Design and support:** $\mathbf{X}$ has a density p_X on $[0, 1]^p$ with $0 < c_{\min} \leq p_X(\mathbf{x}) \leq c_{\max} < \infty$.
- (A2) **Smoothness:** f_0 is L -Lipschitz: $|f_0(\mathbf{x}) - f_0(\mathbf{x}')| \leq L\|\mathbf{x} - \mathbf{x}'\|$.
- (A3) **Moments:** $\mathbb{E}[Y^2] < \infty$ (true for Poisson).
- (A4) **Leaf size:** $k_n \rightarrow \infty$ and $k_n/n \rightarrow 0$ as $n \rightarrow \infty$.
- (A5) **Randomization:** At each node, $\text{mtry} = \text{mtry}(n)$ satisfies $\text{mtry} = o(p)$ and $\text{mtry} \geq c > 0$.
- (A6) **Balanced splits:** There exists $\gamma \in (0, 1/2)$, such that each split produces child nodes containing at least a γ -fraction of the parent samples (in probability).
- (A7) **Sparsity (for rates/selection):** There exists an active set, $\mathcal{S} \subset \{1, \dots, p\}$, with $|\mathcal{S}| = s \ll p$, such that $\mu_0(\mathbf{x})$ depends on $\mathbf{x}$ only through $\mathbf{x}_{\mathcal{S}}$ and with nondegenerate marginals along $\mathcal{S}$.

Assumptions (A1)–(A6) are standard for histogram/tree estimators and ensure shrinking cells with increasing sample size; (A7) is used for variable-selection properties and rates in the intrinsic dimension, s .

Definition 1 (Cells and diameters). For a tree, T , denote by $\mathcal{P}_T$ its partition of $[0, 1]^p$ into terminal nodes (leaves). For $\mathbf{x} \in [0, 1]^p$, let $A_T(\mathbf{x})$ for the unique cell containing $\mathbf{x}$, and $\text{diam}(A)$ for the cell diameter.

Lemma 1 (Shrinking cells and leaf occupancy). Under (A1), (A4), and (A6), for any fixed $\mathbf{x}$,

$$\mathbb{P}\left(|A_T(\mathbf{x})| \geq k_n\right) \rightarrow 1, \quad \text{and} \quad \text{diam}(A_T(\mathbf{x})) \xrightarrow{p} 0,$$

uniformly over trees generated by the above splitting rule.

Proof. By (A6), each split reduces node sample size by at most a factor $(1 - \gamma)$, so the depth until stopping at size k_n is at least of order $\log(n/k_n)$. Because splits are axis-aligned, the side lengths along split coordinates shrink by at least a factor bounded away from 1 at each visit, implying $\text{diam} \rightarrow 0$ as depth grows. By (A1), node volumes are proportional to node sample fractions in probability, so when splitting stops, leaf sample sizes concentrate around their expectations, ensuring $|A_T(\mathbf{x})| \geq k_n$ with probability tending to 1. $\square$

Lemma 2 (Leaf-wise mean consistency). Under (A1)–(A4) and Lemma 1, for any fixed $\mathbf{x}$,

$$\hat{\mu}_T(\mathbf{x}) = \frac{1}{|A_T(\mathbf{x})|} \sum_{i: \mathbf{X}_i \in A_T(\mathbf{x})} Y_i \xrightarrow{p} \mu_0(\mathbf{x}).$$

Proof. Let $\hat{\mu}_T(\mathbf{x}) = \bar{Y}_{A_T(\mathbf{x})}$. We decompose

$$\bar{Y}_A - \mu_0(\mathbf{x}) = \underbrace{(\bar{Y}_A - \mathbb{E}[Y | \mathbf{X} \in A])}_{\text{estimation}} + \underbrace{(\mathbb{E}[Y | \mathbf{X} \in A] - \mu_0(\mathbf{x}))}_{\text{approximation}}.$$

The estimation term vanishes in probability by the conditional law of large numbers given $|A| \rightarrow \infty$ ((A4)) and $\mathbb{E}[Y^2] < \infty$ ((A3)). For the approximation term,

$$|\mathbb{E}[Y | \mathbf{X} \in A] - \mu_0(\mathbf{x})| = |\mathbb{E}[\mu_0(\mathbf{X}) | \mathbf{X} \in A] - \mu_0(\mathbf{x})| \leq \mathbb{E}[|\mu_0(\mathbf{X}) - \mu_0(\mathbf{x})| | \mathbf{X} \in A].$$

Since $\mu_0 = \exp(f_0)$ and f_0 is L -Lipschitz, μ_0 is Lipschitz on the compact set with constant $Le^{\|f_0\|_\infty}$; hence, the expectation is bounded by $C \cdot \text{diam}(A) \rightarrow 0$ in probability by Lemma 1. $\square$

Theorem 1 (Consistency of Poisson Forest). *Let $\{(\mathbf{X}_i, Y_i)\}_{i=1}^n$ be i.i.d. from the Poisson model in (1). Suppose (A1)–(A6) hold and $B \rightarrow \infty$. Then, the forest predictor*

$$\hat{\mu}_n(\mathbf{x}) = \frac{1}{B} \sum_{b=1}^B \hat{\mu}_{T_b}(\mathbf{x})$$

is $L^2(P_X)$ -consistent:

$$\lim_{n \rightarrow \infty} \mathbb{E}_{\mathbf{X}}[(\hat{\mu}_n(\mathbf{X}) - \mu_0(\mathbf{X}))^2] = 0.$$

Proof. Assumed $\mathbf{x}$ is fixed. By Lemma 2, for each tree T , $\hat{\mu}_T(\mathbf{x}) \xrightarrow{p} \mu_0(\mathbf{x})$. Moreover, conditional on the training data, the B trees are i.i.d. and their average converges almost surely to the conditional expectation by the conditional SLLN as $B \rightarrow \infty$:

$$\hat{\mu}_n(\mathbf{x}) \rightarrow \mathbb{E}[\hat{\mu}_T(\mathbf{x}) | \text{data}].$$

However, Lemma 2 implies $\mathbb{E}[\hat{\mu}_T(\mathbf{x}) | \text{data}] \xrightarrow{p} \mu_0(\mathbf{x})$ (bounded second moments follow from (A3)). Hence, $\hat{\mu}_n(\mathbf{x}) \xrightarrow{p} \mu_0(\mathbf{x})$ for every $\mathbf{x}$. Dominated convergence (uniform L^2 -integrability is ensured by (A3)) yields

$$\mathbb{E}_{\mathbf{X}}[(\hat{\mu}_n(\mathbf{X}) - \mu_0(\mathbf{X}))^2] \rightarrow 0.$$

$\square$

Remark 1 (Role of deviance-based splitting). *The proof only uses that splits lead to shrinking cells with growing leaf size. However, the Poisson deviance targets the conditional mean μ_0 and therefore accelerates the focusing of splits along directions with nonzero partial variation (see Lemma 3).*

2.2.4. High-Dimensional Selection Behavior and Rates

We now study the probability of selecting irrelevant variables under sparsity and the resulting rates measured in the intrinsic dimension, s .

Lemma 3 (Vanishing selection probability for noise variables). *Assume (A1)–(A6) and (A7). Let $j \notin \mathcal{S}$ be irrelevant. At any node $\mathcal{A}$ with $|\mathcal{A}| \rightarrow \infty$, denote by $\Delta_j(\mathcal{A})$ the maximal decrease in Poisson deviance over all threshold splits along coordinate j . Then,*

$$\mathbb{E}[\Delta_j(\mathcal{A}) | \mathcal{A}] \rightarrow 0, \quad \text{and} \quad \mathbb{P}\left(\Delta_j(\mathcal{A}) \geq \max_{k \in \mathcal{S}} \Delta_k(\mathcal{A})\right) \rightarrow 0.$$

Consequently, if $m_{\text{try}} = o(p)$ and the active coordinates appear among the m_{try} candidates with probability bounded away from zero, the probability that a noise variable is ultimately selected at the node tends to zero.

Proof. For $j \notin \mathcal{S}$, $\mu_0(\mathbf{x})$ is independent of x_j , conditional on $\mathbf{x}_{-j}$. Consider any split as $x_j \leq t$ vs. $x_j > t$, forming children $\mathcal{A}_L, \mathcal{A}_R$. The expected deviance decrease equals twice the empirical Kullback–Leibler improvement of using two means instead of one. Under irrelevance, the true conditional means in $\mathcal{A}_L$ and $\mathcal{A}_R$ coincide up to the sampling fluctuations:

$$\mu_{0,L} - \mu_{0,R} = \mathbb{E}[\mu_0(\mathbf{X}) \mid \mathbf{X} \in \mathcal{A}_L] - \mathbb{E}[\mu_0(\mathbf{X}) \mid \mathbf{X} \in \mathcal{A}_R] \rightarrow 0,$$

as $|\mathcal{A}| \rightarrow \infty$ and $\text{diam}(\mathcal{A}) \rightarrow 0$ (Lemma 1). A Taylor expansion of the Poisson log-likelihood around the parent mean shows that the population deviance decrease is quadratic in $\mu_{0,L} - \mu_{0,R}$, and hence it vanishes. Conversely, for $k \in \mathcal{S}$ with nonzero local variation of f_0 along k , the two conditional means differ by order equal to the local Lipschitz variation, yielding a strictly positive population improvement. Concentration of empirical deviance around its expectation (Bernstein-type inequalities under (A3)) implies the stated probability limit. Since $\text{mtry} = o(p)$, but where the chance to include an active coordinate among candidates remains bounded away from zero (hypergeometric tail with $s \ll p$), the selection probability for noise variables tends to zero. $\square$

Theorem 2 (Intrinsic-dimension prediction rate). *Assume (A1)–(A7). Suppose the tree-growing rule continues splitting along active coordinates so that, along each $k \in \mathcal{S}$, the expected number of effective splits on the path to $\mathbf{x}$ grows like $O(\log(n/k_n))$ (as induced by Lemma 3 and (A6)). Let f_0 be Lipschitz. Then, there exists a choice of k_n , such that*

$$\mathbb{E}_{\mathbf{X}}[(\hat{\mu}_n(\mathbf{X}) - \mu_0(\mathbf{X}))^2] = \tilde{O}(n^{-2/(2+s)}),$$

i.e., the forest achieves the minimax rate for Lipschitz regression in s dimensions up to logarithmic factors, behaving as if the active set were known (oracle rate).

Proof. Suppose we represent the forest as an ensemble of adaptive histograms whose cells have side lengths $h_{n,k}$ along active dimensions $k \in \mathcal{S}$ and order-one side lengths along inactive coordinates (since they are seldom split). Lemma 3 implies the fraction of splits allocated to inactive coordinates vanishes, so the effective partition adapts to $\mathcal{S}$. The mean squared error decomposes as

$$\text{MSE} \asymp \underbrace{\sum_{k \in \mathcal{S}} h_{n,k}^2}_{\text{bias}^2 \text{ for Lipschitz } f_0} + \underbrace{\frac{1}{n \prod_{k \in \mathcal{S}} h_{n,k}}}_{\text{variance (leaf occupancy)}},$$

ignoring log factors. Balancing bias² and variance by choosing $h_{n,k} \asymp n^{-1/(2+s)}$ (isotropic in $\mathcal{S}$), we obtain the oracle rate $n^{-2/(2+s)}$. The deviance-based splitting plus (A6) yields sufficient balancing of child sizes to realize these $h_{n,k}$ scales in probability. Bagging with $B \rightarrow \infty$ removes an additional correlation factor. Logarithmic penalties arise from randomness and candidate subsampling; see classical analyses of adaptive partitioning estimators. $\square$

Corollary 1 (Oracle prediction rate). *Under the conditions of Theorem 2, the Poisson forest achieves the minimax-optimal prediction rate (up to logarithmic factors) for Lipschitz regression in s dimensions:*

$$\mathbb{E}_{\mathbf{X}}[(\hat{\mu}_n(\mathbf{X}) - \mu_0(\mathbf{X}))^2] = \tilde{O}(n^{-2/(2+s)}).$$

2.2.5. Negative Binomial Extension with Fixed Dispersion

We extend the analysis to Negative Binomial responses with fixed dispersion $\theta > 0$.

Let $Y \mid \mathbf{X} = \mathbf{x} \sim \text{NB}(\mu_0(\mathbf{x}), \theta)$, parameterized so that $\mathbb{E}[Y \mid \mathbf{X}] = \mu_0(\mathbf{x})$ and $\text{Var}(Y \mid \mathbf{X}) = \mu_0(\mathbf{x}) + \mu_0(\mathbf{x})^2/\theta$.

Theorem 3 (Consistency of NB Forest with fixed dispersion). *Under (A1)–(A6) and the NB model with fixed $\theta \in (0, \infty)$, the forest predictor $\hat{\mu}_n$ constructed with NB deviance splits and leaf-wise sample means is $L^2(P_X)$ -consistent.*

Proof. The proof parallels Theorem 1. The NB deviance is (up to constants) the empirical Bregman divergence associated with the NB log-likelihood; it is minimized by the sample mean in each node, so the tree predictor remains the leaf-wise mean. Lemma 1 holds unchanged. The conditional LLN applies since $\mathbb{E}[Y^2] < \infty$ for fixed θ , ensuring the estimation term vanishes; the approximation term again vanishes by Lipschitz continuity of μ_0 . Averaging over $B \rightarrow \infty$ trees yields L^2 -consistency by dominated convergence. $\square$

Remark 2 (On fixed dispersion). *Fixing θ yields a correct variance model only if dispersion is constant. If $\theta = \theta(\mathbf{x})$ varies, the estimator remains consistent for the mean under the conditions above because the node-wise mean is still an unbiased estimator of μ_0 , but the deviance criterion is misspecified for split selection. Estimating local θ per node may sharpen splits but increases variance and computational cost, potentially harming high-dimensional performance.*

2.2.6. Variable Importance

Definition 2 (Deviance-based variable importance). *For variable j , define*

$$\text{VI}(j) = \mathbb{E}_T \left[\sum_{\text{splits on } j \text{ in } T} \Delta \mathcal{D} \right],$$

the expected total decrease in Poisson/NB deviance over all splits on j , averaged over the tree randomness.

Proposition 1 (Consistency of importance ranking on active set). *Under (A1)–(A7), for any $\epsilon > 0$, there exists n_0 , such that, for all $n \geq n_0$,*

$$\min_{j \in \mathcal{S}} \text{VI}(j) > \max_{j \notin \mathcal{S}} \text{VI}(j) - \epsilon,$$

i.e., active variables dominate noise variables in deviance-based importance asymptotically.

Proof. By Lemma 3, the probability of selecting an irrelevant coordinate at any node tends to zero, and the expected deviance improvement from such splits tends to zero uniformly. Conversely, for any $j \in \mathcal{S}$, the expected improvement remains bounded away from zero on a non-vanishing fraction of nodes visited along paths where f_0 has local variation. Summing over the (random) number of such events and averaging over trees yields the separation. $\square$

2.2.7. Alternative Risk Decompositions and Variance Control

Lemma 4 (Bias–variance decomposition). *Let $\mathcal{F}_n$ be the σ -algebra generated by the forest partition $\{A_T(\cdot)\}$ across trees and the training inputs. Then,*

$$\mathbb{E}[(\hat{\mu}_n(\mathbf{X}) - \mu_0(\mathbf{X}))^2] = \mathbb{E}[(\mathbb{E}[\hat{\mu}_n(\mathbf{X}) \mid \mathcal{F}_n] - \mu_0(\mathbf{X}))^2] + \mathbb{E}[\text{Var}(\hat{\mu}_n(\mathbf{X}) \mid \mathcal{F}_n)].$$

Moreover,

$$\text{Var}(\hat{\mu}_n(\mathbf{X}) \mid \mathcal{F}_n) \leq \frac{C}{B} \cdot \frac{1}{\min_T |A_T(\mathbf{X})|} \lesssim \frac{1}{B k_n},$$

for some $C < \infty$, implying that, for any $k_n \rightarrow \infty$, choosing $B \rightarrow \infty$ drives the conditional variance term to 0.

Proof. The decomposition is standard. Conditional variance bound follows since each tree predictor is an average of $|A_T(\mathbf{X})|$ Poisson (or NB) observations with conditional variance bounded by $\mathbb{E}[Y | \mathbf{X}] + \mathbb{E}[Y | \mathbf{X}]^2 / \theta \leq C'$ on compact sets with Lipschitz μ_0 , and averaging B i.i.d. trees divides variance by B up to correlation, which is controlled by sample-splitting and randomization. $\square$

2.2.8. Technical Remarks on Deviance Splits

Lemma 5 (Local optimality of deviance split). *At a node, $\mathcal{A}$, the optimal population split between thresholds of a given coordinate maximizes the decrease in expected Poisson (or NB) deviance and coincides with the split that maximizes the difference between child means in a variance-stabilized scale. Empirical maximizers converge to population maximizers as $|\mathcal{A}| \rightarrow \infty$.*

Proof. For exponential family models with mean parameter μ , the deviance is a Bregman divergence for the cumulant function, $b(\cdot)$. The one-step two-group fit minimizing total deviance sets each group's mean to its empirical mean. The expected deviance decrease is the Bregman information gain, strictly increasing in the squared difference between group means under mild regularity. Uniform Glivenko–Cantelli properties of histograms on $[0, 1]^p$ ensure that empirical maximizers converge to population ones as the node size grows. $\square$

Remark 3. *Under mild design, smoothness, and splitting balance conditions, Poisson/NB deviance forests with growing leaf size and vanishing leaf fraction are L^2 -consistent. In high dimensions with sparsity, the splitting rule suppresses irrelevant variables and attains oracle prediction rates up to logarithmic factors in the intrinsic dimension, s .*

Algorithm 1 outlines the complete procedure for constructing a Random Forest for Poisson or Negative Binomial responses. The key implementation points to note are as follows:

- **Global Dispersion Estimation:** For the Negative Binomial family, a global dispersion parameter (θ) is estimated once on the entire dataset prior to bootstrapping. This fixed value is then used in the deviance calculation for all subsequent splits across all trees, ensuring stability and computational efficiency. Re-estimating dispersion in every node would be computationally prohibitive and statistically unstable.
- **Splitting Criterion:** The core of the algorithm is the search for the split s (a feature j and cutoff point c) that minimizes the deviance $D(s)$. For Poisson, this is the Poisson deviance. For Negative Binomial, it is the likelihood ratio deviance derived from the log-likelihood $l(\mu, \theta; y)$.
- **Stopping Criteria:** A node is not split if it is “pure” (e.g., all counts are zero) or too small (e.g., fewer than `min.node.size` observations), which is a crucial hyperparameter to control overfitting.
- **Hyperparameters:** Key hyperparameters include `mtry` (number of features tried per split, here set to $\lfloor \log(p) \rfloor$ as a default for high-dimensional settings), B (number of trees), `min.node.size`, and the maximum depth of trees (implicitly controlled by the stopping criteria).

Algorithm 1 High-Dimensional Random Forest for Count Data

Require: Response vector $\mathbf{y} \in \mathbb{N}_0^n$, predictor matrix $\mathbf{X} \in \mathbb{R}^{n \times p}$, family $\in \{\text{'poisson'}, \text{'negbin'}\}$, number of trees B , and hyperparameters.

Ensure: A trained Random Forest model $\mathcal{M}$.

```

1: Initialize the forest  $\mathcal{M} \leftarrow \emptyset$ .
2: Set  $\text{mtry} \leftarrow \lfloor \log(p) \rfloor$  {Default for  $p \gg n$ }
3: if family = 'negbin' then
4:    $\hat{\theta} \leftarrow \arg\max_{\theta} l(\hat{\mu}, \theta; \mathbf{y})$  {Estimate global dispersion via MLE (e.g., 'glm.nb')}
5: end if
6: for  $b = 1$  to  $B$  do
7:    $\mathbf{X}^{(b)}, \mathbf{y}^{(b)} \leftarrow \text{BootstrapSample}(\mathbf{X}, \mathbf{y})$  {Draw sample with replacement}
8:   Initialize root node  $R$  containing  $\mathbf{X}^{(b)}, \mathbf{y}^{(b)}$ .
9:   Initialize an empty stack and push  $R$ .
10:  while stack is not empty do
11:    Pop a node  $t$ .
12:    if node  $t$  is not pure or too small then {Check stopping criteria (min node size)}
13:       $\mathcal{F} \leftarrow \text{RandomSubset}(\text{features}, \text{mtry})$ 
14:       $s^* \leftarrow \emptyset$ 
15:       $D^* \leftarrow \infty$ 
16:      for all features  $j$  in  $\mathcal{F}$  do
17:        for all possible split points  $c$  for  $X_j$  in node  $t$  do
18:          Partition  $t$  into  $t_L(c)$  and  $t_R(c)$  based on  $X_j \leq c$ .
19:          if family = 'poisson' then
20:             $D(s) \leftarrow D(s, t)$  {Calculate Poisson deviance using Equation (7)}
21:          else
22:             $D(s) \leftarrow -2[l(\hat{\mu}_{t_L}, \hat{\theta}; \mathbf{y}_{t_L}) + l(\hat{\mu}_{t_R}, \hat{\theta}; \mathbf{y}_{t_R}) - l(\hat{\mu}_t, \hat{\theta}; \mathbf{y}_t)]$  {Neg. Bin.
Deviance}
23:          end if
24:          if  $D(s) < D^*$  then
25:             $D^* \leftarrow D(s)$ 
26:             $s^* \leftarrow (j, c)$ 
27:          end if
28:        end for
29:      end for
30:      if  $s^* \neq \emptyset$  then
31:        Split node  $t$  into  $t_L$  and  $t_R$  using  $s^*$ .
32:        Push  $t_L$  and  $t_R$  onto the stack.
33:      end if
34:    end if
35:  end while
36:  Store tree  $T_b$  in ensemble  $\mathcal{M}$ .
37: end for
38: return  $\mathcal{M}$ 

```

The theoretical guarantees of consistency and convergence rates are contingent upon appropriate choices of key hyperparameters, which directly control the bias–variance trade-off and the forest’s adaptivity to the intrinsic data structure. The leaf-size parameter k_n is particularly critical; the condition $k_n \rightarrow \infty$ with $k_n/n \rightarrow 0$ is necessary for consistency, ensuring that leaf estimates become asymptotically unbiased while vanishing in relative size. In practice, larger values of `min.node.size` (the practical equivalent of k_n) promote smoother estimates and better variance reduction but can increase bias by over-smoothing local features, a trade-off that must be calibrated through cross-validation. The randomization parameter `mtry` governs the diversity of the ensemble and its ability to suppress noise variables. Theoretically, `mtry` = $o(p)$ ensures that the probability of selecting irrelevant variables vanishes, which is crucial for achieving the oracle rate in sparse settings. Empirically, setting `mtry` = $\log(p)$ provides a robust default that balances this

theoretical requirement with computational efficiency. Finally, while the number of trees B does not affect the asymptotic properties of the infinite forest (as $B \rightarrow \infty$), a sufficiently large B is required in practice to ensure the Monte Carlo error from bagging is negligible, thus stabilizing predictions and variable importance measures. Therefore, setting $k_n = 2$, $m_{\text{try}} = \log(p)$ and $B \geq 500$ are empirical necessities to realize the theoretical advantages of the proposed framework.

2.3. Why Does Deviance-Based Splitting Dominate MSE-Based Splitting?

We justify the superiority of deviance-based splitting for count responses by analyzing the local population objective optimized at each split. For Poisson and Negative Binomial (NB) models, the deviance is the empirical Bregman divergence associated with the canonical cumulant function and is Fisher-consistent for the conditional mean, $\mu(\mathbf{x})$. In contrast, MSE-based splitting on the raw scale implicitly targets conditional mean under severe heteroskedasticity $\text{Var}(Y | \mathbf{X}) \propto \mu(\mathbf{X})$ (Poisson) or $\mu(\mathbf{X}) + \mu(\mathbf{X})^2/\theta$ (NB), which distorts split selection; and MSE on a variance-stabilized or log-transformed response introduces transformation bias and nonlinearity misalignment. We formalize these points below.

2.3.1. Exponential Family Structure and Bregman Risk

Let $Y | \mathbf{X} = \mathbf{x}$ follow an exponential family with log-likelihood $\ell(y; \eta) = y\eta - b(\eta) + c(y)$, mean $\mu = b'(\eta)$, and variance $V(\mu) = b''(\eta)$. For Poisson, $b(\eta) = e^\eta$, $\mu = e^\eta$, $V(\mu) = \mu$; for NB with fixed dispersion, $\theta > 0$, under mean variance parameterization, $V(\mu) = \mu + \mu^2/\theta$. The associated Bregman divergence between y and a mean parameter m is $D_b(y, m) = b(y^*) - b(m) - b'(m)(y^* - m)$, which reduces, up to constants, to the deviance. On a node, $\mathcal{A}$, the population deviance risk of using a constant predictor, m , is $R_{\text{Dev}}(m; \mathcal{A}) = \mathbb{E}[D_b(Y, m) | \mathbf{X} \in \mathcal{A}]$, uniquely minimized at $m = \mu_{\mathcal{A}} := \mathbb{E}[Y | \mathbf{X} \in \mathcal{A}]$. Hence, the population decrease in deviance from splitting $\mathcal{A}$ into children $\mathcal{A}_L, \mathcal{A}_R$ with means $\mu_L \neq \mu_R$ as strictly positive and equaling the Bregman information gain.

2.3.2. Local Quadratic Approximation and Optimal Split Direction

Consider a candidate split along coordinate j at threshold t , producing children with means μ_L and μ_R and a parent mean of μ_P . A Taylor expansion of the population deviance about μ_P yields, for Poisson,

$$\Delta_{\text{Dev}}(j, t | \mathcal{A}) \equiv R_{\text{Dev}}(\mu_P; \mathcal{A}) - [\pi_L R_{\text{Dev}}(\mu_L; \mathcal{A}_L) + \pi_R R_{\text{Dev}}(\mu_R; \mathcal{A}_R)] \asymp \frac{\pi_L \pi_R}{2} \frac{(\mu_L - \mu_R)^2}{\mu_P},$$

where $\pi_L = \mathbb{P}(\mathbf{X} \in \mathcal{A}_L | \mathbf{X} \in \mathcal{A})$ and similarly for π_R . For NB with fixed θ , the same expansion yields

$$\Delta_{\text{Dev}}(j, t | \mathcal{A}) \asymp \frac{\pi_L \pi_R}{2} \frac{(\mu_L - \mu_R)^2}{\mu_P + \mu_P^2/\theta}.$$

Thus, the deviance gain is asymptotically proportional to a variance-weighted squared contrast in means, automatically normalizing by the local noise level. This weighting implements an optimal signal-to-noise criterion at the split scale.

2.3.3. MSE Splitting on the Raw Scale

If instead the split criterion is the reduction in squared error with a constant fit m on each child, the population gain is

$$\Delta_{\text{MSE}}(j, t | \mathcal{A}) = \text{Var}(\mathbb{E}[Y | \mathbf{X}, \mathbf{X} \in \mathcal{A}]) - \left\{ \pi_L \text{Var}(\mathbb{E}[Y | \mathbf{X}, \mathcal{A}_L]) + \pi_R \text{Var}(\mathbb{E}[Y | \mathbf{X}, \mathcal{A}_R]) \right\} \asymp \pi_L \pi_R (\mu_L - \mu_R)^2.$$

This criterion ignores the heteroskedastic variance $V(\mu)$ and thus over-weights regions with large μ_P even when the standardized contrast $(\mu_L - \mu_R)/\sqrt{V(\mu_P)}$ is small. Consequently,

MSE splitting can favor spurious splits driven by variance heterogeneity rather than genuine mean structure, reducing statistical efficiency and interpretability.

2.3.4. Log- and Variance-Stabilized Transforms

A common workaround is to split on a transformed pseudo-response such as $Z = \log(Y + 1)$ or $Z = 2\sqrt{Y + 3/8}$ and apply MSE. Let g be a smooth monotone transform and suppose $Z = g(Y)$. For small cell diameters, an expansion of the delta method gives $\mathbb{E}[Z | \mathbf{X}] \approx g(\mu) + \frac{1}{2}g''(\mu)V(\mu)$ and $\text{Var}(Z | \mathbf{X}) \approx g'(\mu)^2V(\mu)$. MSE splitting on Z therefore targets contrasts in $g(\mu)$ contaminated by curvature bias $\frac{1}{2}g''(\mu)V(\mu)$ and scaled by the heteroskedastic factor, $g'(\mu)^2V(\mu)$. In particular, for $g(u) = \log(u + 1)$, $g''(u) < 0$ produces a negative bias that varies with μ , so that the splits align with the artifacts of the transformation as much as with the true differences in μ . Deviance eliminates both issues by working on the native mean scale with the correct variance normalization.

2.3.5. Population Optimality of Deviance Splits

Formally, on any node $\mathcal{A}$, the deviance gain induced by the population-optimal split (j^*, t^*) satisfies

$$(j^*, t^*) \in \arg \max_{(j,t)} \frac{\pi_L(j,t)\pi_R(j,t)}{V(\mu_P)} (\mu_L(j,t) - \mu_R(j,t))^2 + o(1),$$

where $V(\mu_P)$ is the local variance function. For Poisson and NB, this is the most powerful split locally to detect differences in conditional means, as it coincides with the score test for $H_0 : \mu_L = \mu_R$ versus $H_1 : \mu_L \neq \mu_R$ in the respective models. MSE on the raw scale corresponds to the same numerator but with unit denominator, which is suboptimal under heteroskedasticity; MSE on transformed scales targets $g(\mu)$ rather than μ and hence is misaligned with the inferential target.

Proposition 2 (Asymptotic dominance of deviance gain). *Fix a node $\mathcal{A}$ with the parent mean μ_P and let $\mathcal{C}$ be any class of threshold splits with child proportions bounded away from 0 and 1. Under Lipschitz smoothness of $\mu(\cdot)$ and regular design, for Poisson and NB models,*

$$\max_{(j,t) \in \mathcal{C}} \Delta_{\text{Dev}}(j,t | \mathcal{A}) \geq \frac{1}{\sup_{\mathcal{C}} V(\mu_P)} \max_{(j,t) \in \mathcal{C}} \Delta_{\text{MSE}}(j,t | \mathcal{A}) + o(1).$$

Moreover, if μ_P varies across candidate splits, there exist configurations where the inequality is strict and the argmax of Δ_{Dev} differs from that of Δ_{MSE} , with deviance selecting the split maximizing the standardized contrast $(\mu_L - \mu_R)^2 / V(\mu_P)$.

Proof. The quadratic expansions above imply $\Delta_{\text{Dev}} \asymp \frac{\pi_L \pi_R}{2} (\mu_L - \mu_R)^2 / V(\mu_P)$ and $\Delta_{\text{MSE}} \asymp \pi_L \pi_R (\mu_L - \mu_R)^2$. Taking suprema over $\mathcal{C}$ and using bounded away from zero child proportions yield the display up to a constant factor. When candidate splits induce different parent means μ_P (for instance, because (j, t) changes the covariate composition), the denominators $V(\mu_P)$ differ and argmax does not need to coincide. The terms $o(1)$ vanish as the node size is diverged by standard uniform laws of large numbers over threshold classes. $\square$

3. Simulation Study

We conduct a comprehensive simulation study to evaluate the finite-sample performance of the proposed Poisson and Negative Binomial Random Forest methods. The study is designed to assess performance across a range of challenging data-generating processes (DGPs) common in high-dimensional settings, including overdispersion, nonlinearity, zero-

inflation, measurement error, and ultra-high dimensionality. We compare our methods against established benchmarks: ℓ_1 -penalized regression (Lasso), the Elastic-Net ($\alpha = 0.5$), gradient boosting (XGBoost), and a standard Random Forest applied to $\log(Y + 1)$ as a pseudo-response.

3.1. Simulation Design and Data Generation

The core structure of our simulations is as follows. For each replication, we generate a training and a test set from a specified DGP. Unless otherwise noted, the baseline parameters are a sample size of $n = 100$ and an ambient dimension of $p = 1000$. Predictors are drawn from a block-correlated multivariate normal distribution, $\mathbf{X}_i \sim \mathcal{N}_p(\mathbf{0}, \Sigma)$, where Σ is a block-diagonal matrix. Each block has a size of 50, and the within-block correlation is set to $\rho = 0.5$.

The true regression vector $\beta_0 \in \mathbb{R}^p$ is sparse, with only $s = 10$ nonzero coefficients. The active set $\mathcal{S} = \{j_1, \dots, j_{10}\}$ is chosen uniformly at random. The values of the non-zero coefficients are drawn from a uniform distribution, $\beta_{0j} \sim \text{Unif}(-0.5, 0.5)$ for $j \in \mathcal{S}$, and are zero otherwise. An intercept $\beta_0 = 1$ is included in all models.

We define two broad classes of DGPs based on the structure of the predictor function $f_0(\mathbf{X}_i)$.

3.1.1. Example 1: Linear Mean Structure

In this example, the predictor is a linear combination of the true signals:

$$f_0^{(\text{linear})}(\mathbf{X}_i) = \beta_0 + \mathbf{X}_i^\top \beta_0.$$

We consider several distributional scenarios built upon this linear structure:

- **Poisson (Equi-dispersed):** $Y_i | \mathbf{X}_i \sim \text{Poisson}(\mu_i)$, where $\mu_i = \exp(f_0^{(\text{linear})}(\mathbf{X}_i))$.
- **Negative Binomial (Over-dispersed):** $Y_i | \mathbf{X}_i \sim \text{NB}(\mu_i, \theta)$, where $\mu_i = \exp(f_0^{(\text{linear})}(\mathbf{X}_i))$ and the dispersion parameter is set to $\theta = 1.5$, giving $\text{Var}(Y_i | \mathbf{X}_i) = \mu_i + \mu_i^2/\theta$.
- **Zero-Inflated Poisson (ZIP):** Data are generated from a mixture distribution,

$$Y_i \sim \begin{cases} 0 & \text{with probability } \pi_i, \\ \text{Poisson}(\mu_i) & \text{with probability } 1 - \pi_i, \end{cases}$$

where $\mu_i = \exp(f_0^{(\text{linear})}(\mathbf{X}_i))$. The zero-inflation probability π_i is modeled as $\pi_i = \text{logit}^{-1}(\mathbf{Z}_i^\top \gamma)$, with $\mathbf{Z}_i$ being the first three active predictors and $\gamma = (1, -1, 0.5)^\top$, resulting in approximately 20% excess zeros.

- **Measurement Error:** The linear predictor is used, but the features available to the models are corrupted versions $X_{ij}^* = X_{ij} + \epsilon_{ij}$, where $\epsilon_{ij} \sim \mathcal{N}(0, \sigma^2)$ and $\sigma = 0.2$.

3.1.2. Example 2: Nonlinear Mean Structure

In this example, the predictor function includes nonlinear effects to assess the ability of methods to capture complex signal relationships:

$$f_0^{(\text{nonlinear})}(\mathbf{X}_i) = \beta_0 + \mathbf{X}_i^\top \beta_0 + \gamma_1 X_{i,j_1}^2 + \gamma_2 X_{i,j_2}^2,$$

where j_1 and j_2 are the first two features in the active set $\mathcal{S}$, and $\gamma_1 = \gamma_2 = 0.5$.

- **Nonlinear Poisson:** $Y_i | \mathbf{X}_i \sim \text{Poisson}(\mu_i)$, where $\mu_i = \exp(f_0^{(\text{nonlinear})}(\mathbf{X}_i))$.
- **Nonlinear Negative Binomial:** $Y_i | \mathbf{X}_i \sim \text{NB}(\mu_i, \theta)$, where $\mu_i = \exp(f_0^{(\text{nonlinear})}(\mathbf{X}_i))$ and $\theta = 1.5$.

- **Nonlinear Zero-Inflated Poisson (ZIP):** The nonlinear predictor defines the Poisson mean in the ZIP model described above.

3.1.3. Ultra-High-Dimensional Regime

To stress-test the algorithms, we include a scenario with $n = 50$ and $p = 5000$, preserving sparsity ($s = 10$). Data are generated using the linear Poisson DGP.

3.2. Performance Metrics and Evaluation

Across all scenarios, we evaluate predictive and variable selection performance. Prediction accuracy is measured via root mean squared error,

$$\text{RMSE} = \sqrt{\frac{1}{n_{\text{test}}} \sum_{i=1}^{n_{\text{test}}} (\hat{\mu}(\mathbf{X}_i) - \mu_i)^2},$$

and Poisson or NB deviance,

$$D = 2 \sum_{i=1}^{n_{\text{test}}} \left(y_i \log \frac{y_i}{\hat{\mu}(\mathbf{X}_i)} - (y_i - \hat{\mu}(\mathbf{X}_i)) \right),$$

with the convention that $0 \log 0 = 0$. Variable recovery is quantified using support overlap

$$\text{Power} = \frac{|\hat{\mathcal{S}} \cap \mathcal{S}|}{|\mathcal{S}|}, \quad \text{FDR} = \frac{|\hat{\mathcal{S}} \setminus \mathcal{S}|}{|\hat{\mathcal{S}}|},$$

where $\hat{\mathcal{S}}$ denotes the set of variables identified as important by each method.

We repeat each simulation 100 times, generating fresh training and test data in each replication. Competing methods include ℓ_1 -penalized regression (Lasso), Elastic-Net with mixing parameter $\alpha = 0.5$, gradient boosting with decision trees (XGBoost), and standard Random Forests applied to $\log(Y + 1)$ as a pseudo-response.

3.3. Estimation of the Unknown Function, $f(\cdot)$

The core objective of each method in this comparison is to accurately estimate the unknown conditional mean function $\mu(\mathbf{X}_i) = \mathbb{E}[Y_i | \mathbf{X}_i] = \exp(f(\mathbf{X}_i))$, which implies estimating the predictor function $f(\mathbf{X}_i)$ on the log-scale. The approaches differ fundamentally in their assumptions about the structure of $f(\cdot)$. In the simulation setup, the estimation function encapsulates the following estimation procedures for a given training set $(\mathbf{X}_{tr}, \mathbf{y}_{tr})$:

- **Penalized GLMs (Lasso, Ridge, Elastic-Net):** These methods assume a *linear* form for the predictor function, $f(\mathbf{X}_i) = \beta_0 + \mathbf{X}_i^\top \boldsymbol{\beta}$. They estimate the coefficients by maximizing a penalized likelihood. For a given penalty parameter, λ (selected via k -fold cross-validation within the training set), the optimization problem for the Poisson case is:

$$\hat{\boldsymbol{\beta}} = \underset{\boldsymbol{\beta}}{\text{argmin}} \left\{ -\frac{1}{n_{tr}} \sum_{i=1}^{n_{tr}} \left[y_i (\beta_0 + \mathbf{X}_i^\top \boldsymbol{\beta}) - \exp(\beta_0 + \mathbf{X}_i^\top \boldsymbol{\beta}) \right] + \lambda P(\boldsymbol{\beta}) \right\},$$

where $P(\boldsymbol{\beta})$ is the ℓ_1 -norm for Lasso or a combination of ℓ_1 and ℓ_2 norms for the Elastic-Net. The estimated function is then $\hat{f}(\mathbf{X}_i) = \hat{\beta}_0 + \mathbf{X}_i^\top \hat{\boldsymbol{\beta}}$. This linearity assumption is misspecified for the nonlinear data-generating processes (DGPs) in our study.

- **Gradient Boosting (XGBoost):** This method assumes an *additive* form of $f(\cdot)$ constructed from M simple regression trees (T_m):

$$f(\mathbf{X}_i) = \sum_{m=1}^M \eta \cdot T_m(\mathbf{X}_i; \boldsymbol{\Theta}_m),$$

where η is a learning rate and Θ_m parameterizes each tree. The algorithm iteratively fits trees to the negative gradient (pseudo-residuals) of the Poisson or Negative Binomial loss function, greedily improving the model's fit to the data. This allows it to capture complex nonlinearities and interactions without an explicit specification, making it more flexible than the penalized GLMs.

- **Standard Random Forest (on $\log(Y + 1)$):** This method sidesteps the generalized linear model framework entirely. It first applies a variance-stabilizing transformation to the count response, $\tilde{y}_i = \log(y_i + 1)$, treating it as a continuous outcome. It then estimates $f(\cdot)$ nonparametrically by averaging predictions from a collection of regression trees, each trained on a bootstrapped sample. The final prediction is made on the log scale, $\hat{f}(\mathbf{X}_i)$, and is subsequently exponentiated to estimate the mean. This approach can capture complex $f(\cdot)$ but may be inefficient as it ignores the count nature and heteroskedasticity of the data during model fitting.
- **Proposed Random Forest (Poisson/NB deviance):** Our method directly and explicitly models the count nature of the data by using the appropriate deviance as the splitting criterion for each tree within the ensemble. It nonparametrically estimates $f(\cdot)$ by finding partitions of the predictor space that minimize the in-sample Poisson or Negative Binomial likelihood. The algorithm's output is an ensemble of trees whose combined predictions directly yield the estimated mean $\hat{\mu}(\mathbf{X}_i)$, from which the predictor function can be inferred as $\hat{f}(\mathbf{X}_i) = \log(\hat{\mu}(\mathbf{X}_i))$. This approach is designed to be both flexible enough to capture the true $f(\cdot)$ and efficient by respecting the distributional properties of the data.

4. Simulation Results

The proposed Random Forest models, RF-Poisson and RF-NB, demonstrated a decisive and substantial superiority in predictive accuracy across all standard data-generating scenarios, as evidenced by their significantly lower root mean squared error (RMSE) values (see Table 1). In the true Poisson setting, the proposed methods (Mean RMSE 0.84–0.86) outperformed their nearest competitors, the glmnet Lasso and Elastic-Net models (Mean RMSE 2.1–2.6), by a factor of nearly three, and vastly surpassed all other benchmarks including the transformed RF on log1p (3.60), XGBoost Poisson (3.96), and the Zero-Inflated model (4.13). This performance advantage was not merely a product of the data matching the model's assumptions, as both RF proposals maintained their lead under Negative Binomial, Zero-Inflated Poisson, and measurement error conditions, with the RF-NB model showing a slight edge in the NB and ZIP scenarios. The abysmal performance of the classical GLM: NB and GLM: Poisson, which failed to converge (resulting in infinite error), starkly highlights the necessity of either regularization or flexible machine learning approaches for these types of problems. Crucially, all penalized regression methods (glmnet), while better than the naive GLMs, were consistently and significantly outperformed by the proposed RF approaches, suggesting that the ability of Random Forests to capture complex, nonlinear relationships and interactions without manual specification provided a critical advantage even in these ostensibly linear simulations.

Table 1. Comparison of predictive performance (RMSE) of proposed Random Forest count regression models (RF–Poisson and RF–NB) against penalized regression (glmnet variants), XGBoost, classical Zero-Inflated Poisson (ZIP), and GLM baselines across Poisson, Negative Binomial (NB), Zero-Inflated Poisson (ZIP), and measurement error (ME) in covariates and linear link scenarios. Too large values are reported as ∞ .

Method	Poisson		NB		ZIP		ME in Covariates	
	Mean	SE	Mean	SE	Mean	SE	Mean	SE
RF–Poisson (<i>proposed</i>)	0.837	0.271	3.265	3.680	2.794	1.182	1.696	0.888
RF–NB (<i>proposed</i>)	0.855	0.265	3.092	3.478	2.664	1.105	1.703	0.857
glmnet-P Lasso	2.095	0.678	8.163	9.201	6.987	2.956	4.241	2.221
glmnet-P EN	2.252	0.698	8.138	9.153	7.011	2.908	4.483	2.256
glmnet-NB Lasso (log1p)	2.476	0.681	8.786	9.394	8.036	3.002	5.709	2.727
glmnet-NB EN (log1p)	2.562	0.642	8.824	9.429	8.075	3.026	5.848	2.753
glmnet-P Ridge	3.119	0.616	8.629	9.737	7.298	2.881	6.568	2.667
glmnet-NB Ridge (log1p)	3.286	0.676	9.197	9.628	8.187	2.991	7.164	2.969
RF on log1p	3.600	0.735	9.409	9.596	8.375	3.013	7.547	3.020
XGBoost Poisson	3.957	0.767	9.844	9.454	8.909	2.988	7.661	2.990
ZIP (zeroinfl + SIS)	4.131	1.031	9.654	7.946	12.686	9.796	8.641	3.479
GLM: NB	∞	–	∞	–	∞	–	∞	–
GLM: Poisson	∞	–	∞	–	∞	–	∞	–

When data generation processes incorporated nonlinearity and extreme complexity, the performance gap between the proposed Random Forest methods and the field of benchmarks became even more pronounced, solidifying their value in realistic settings where linear assumptions are frequently violated. In the Poisson nonlinear and ZIP nonlinear scenarios, the RF models (Mean RMSE 0.94–1.10) again outperformed all other methods by a wide margin (see Table 2), with the error of the next best glmnet models being approximately 2.5 to 3 times higher. The most telling results emerged from the highly challenging “Nonlinear NB” and “Ultra” settings. In the Nonlinear NB scenario, while all methods struggled with high variance and absolute error, the proposed RF–NB model achieved the lowest mean error (39.72), demonstrating a relative but clear robustness compared to the dramatically higher errors of the glmnet variants (102–106) and the catastrophic failure of the ZIP model (220.55). In the “Ultra” complex setting, the proposed methods again secured the lowest prediction errors (RF–NB: 1.39), significantly outperforming all other approaches, including the otherwise stable but mediocre Ridge regression models (3.38–3.40). The consistent failure of the classical GLMs and the poor performance of the linear-model-based glmnet suite across these complex environments underscore a critical limitation of linear predictors. In contrast, the proposed RF–Poisson and RF–NB models exhibited remarkable resilience and predictive accuracy, establishing them as the most robust and reliable methods for regression analysis of count data across a diverse spectrum of challenging, real-world conditions.

The evaluation of model fit through deviance in Table 3 reveals a stark and consistent hierarchy of performance, with the proposed Random Forest models, RF–Poisson and RF–NB, demonstrating a profoundly superior ability to describe the observed data across all linear data-generating mechanisms. The deviance values for the proposed methods are not just marginally but substantially lower than all competing approaches, often by an order of magnitude. For instance, in the Poisson scenario, the mean deviance for RF–Poisson (31.49) is less than half that of the best glmnet Lasso model (78.72) and nearly five times lower than the RF on log1p transform (155.22), unequivocally demonstrating that modeling the counts directly with a tailored Random Forest is far more effective than applying standard Random Forest or linear models to transformed data. This pattern holds true in the more

complex NB and ZIP settings, where the RF–NB model achieves the lowest deviance, indicating its particular utility in handling overdispersion and zero-inflation. The classical GLMs completely fail to provide a meaningful fit, resulting in infinite deviance, while the Zero-Inflated Poisson model (ZIP) performs catastrophically poorly, with deviance values soaring into the thousands, suggesting severe model misspecification or convergence issues in these simulated conditions. The generally poor performance of the XGBoost Poisson implementation further highlights that the success of the proposed methods is not merely due to being a tree-based algorithm but is specifically attributable to their careful design for count data distributions.

Table 2. Comparison of predictive performance (RMSE) under nonlinear data-generating processes: Poisson nonlinear, nonlinear Negative Binomial (NB), Zero-Inflated Poisson (ZIP) nonlinear, and ultra-complex settings. Reported values are mean prediction errors and their standard errors across competing methods. Too large values are reported as ∞ .

Method	Poisson Nonlinear		Nonlinear NB		ZIP Nonlinear		Ultra	
	Mean	SE	Mean	SE	Mean	SE	Mean	SE
RF–Poisson (<i>proposed</i>)	0.942	0.559	41.172	38.812	1.036	0.355	1.469	0.628
RF–NB (<i>proposed</i>)	0.994	0.579	39.720	37.238	1.101	0.385	1.386	0.592
glmnet-P Lasso	2.356	1.397	102.931	97.030	2.590	0.887	3.674	1.571
glmnet-P EN	2.617	1.524	104.527	97.996	2.899	1.013	3.650	1.557
glmnet-NB Lasso (log1p)	2.961	1.715	106.289	101.364	3.740	1.357	3.682	1.686
glmnet-NB EN (log1p)	3.078	1.778	106.238	101.388	3.814	1.384	3.645	1.684
glmnet-P Ridge	3.915	1.980	106.168	97.600	4.193	1.327	3.383	1.622
glmnet-NB Ridge (log1p)	4.082	2.072	106.408	101.351	4.861	1.623	3.396	1.656
RF on log1p	4.607	2.155	105.075	101.089	5.787	1.920	3.574	1.719
XGBoost Poisson	4.809	2.116	105.178	100.288	6.123	1.918	3.798	1.796
ZIP (zeroinfl + SIS)	4.870	2.078	220.545	378.642	5.657	1.733	5.880	3.826
GLM: NB	∞	–	∞	–	∞	–	∞	–
GLM: Poisson	∞	–	∞	–	∞	–	∞	–

Table 3. Comparison of model fitting performance (deviance) of proposed Random Forest count regression models (RF–Poisson and RF–NB) against penalized regression (glmnet variants), XGBoost, classical Zero-Inflated Poisson (ZIP), and GLM baselines across Poisson, Negative Binomial (NB), Zero-Inflated Poisson (ZIP), and measurement error (ME) in covariates and linear link scenarios. Too large values are reported as ∞ .

Method	Poisson		NB		ZIP		ME in Covariates	
	Mean	SE	Mean	SE	Mean	SE	Mean	SE
RF–Poisson (<i>proposed</i>)	31.486	8.362	23.148	4.282	79.733	24.766	35.108	12.658
RF–NB (<i>proposed</i>)	32.156	8.875	21.412	3.317	76.389	22.841	35.863	12.560
glmnet-P Lasso	78.715	20.905	57.872	10.705	199.335	61.915	87.771	31.646
glmnet-P EN	84.624	23.354	56.349	8.728	201.025	60.107	94.378	33.053
glmnet-NB Lasso (log1p)	92.960	32.203	87.106	65.481	296.625	110.055	125.339	56.632
glmnet-NB EN (log1p)	96.184	30.808	78.773	51.944	286.300	98.219	133.370	60.580
glmnet-P Ridge	124.928	34.055	61.500	9.347	224.578	60.945	182.707	69.643
glmnet-NB Ridge (log1p)	135.647	43.558	72.760	15.783	285.333	94.680	215.253	98.265
RF on log1p	155.224	47.094	73.741	15.049	292.275	101.918	246.473	116.175
XGBoost Poisson	216.477	71.653	142.974	43.689	462.302	157.289	286.081	147.531
ZIP (zeroinfl + SIS)	1099.562	525.070	1172.277	772.498	2067.259	988.742	1356.539	1025.372
GLM: NB	∞	–	10,409.210	3361.116	∞	–	∞	–
GLM: Poisson	∞	–	10,156.917	3264.213	∞	–	∞	–

The results under nonlinear and ultra-complex data-generating processes in Table 4 powerfully reinforce the conclusion that the proposed Random Forest methods are uniquely robust and provide the best model fit in highly challenging environments. In the Poisson nonlinear and ZIP nonlinear scenarios, the deviance of the proposed methods (27) is again less than half that of their nearest competitors, the glmnet Lasso and Elastic-Net models (67–74), showcasing their innate capacity to capture complex nonlinear relationships that linear predictors cannot. The most revealing results are found in the “Nonlinear NB” and “Ultra” settings. In the chaotic Nonlinear NB scenario, the proposed RF–NB model achieves a mean deviance of 99.55, which, while high, is dramatically lower and more stable than all other methods, including the glmnet variants whose deviance explodes into the hundreds and thousands, and the ZIP and GLM models, which fail completely with deviance values in the thousands and infinity, respectively. This demonstrates a critical robustness to extreme data complexity. Finally, in the “Ultra” complex setting, the proposed methods again deliver the best possible fit with the lowest deviance (RF–NB: 18.17), outperforming even the simple Ridge regression models. The consistent and dramatic failure of the classical GLMs and the ZIP model across these nonlinear settings, coupled with the middling and highly variable performance of the other flexible methods like XGBoost and standard RF, solidifies the position of the proposed RF–Poisson and RF–NB frameworks as the most reliable and best-fitting methodologies for regression analysis of count data across a vast spectrum of potential real-world data challenges.

Table 4. Comparison of model fitting performance (deviance) under nonlinear data-generating processes: Poisson nonlinear, nonlinear Negative Binomial (NB), Zero-Inflated Poisson (ZIP) nonlinear, and ultra-complex settings. Reported values are mean prediction errors and their standard errors across competing methods. Too large values are reported as ∞ .

Method	Poisson Nonlinear		Nonlinear NB		ZIP Nonlinear		Ultra	
	Mean	SE	Mean	SE	Mean	SE	Mean	SE
RF–Poisson (<i>proposed</i>)	26.854	7.361	119.583	97.671	26.796	6.511	19.094	10.569
RF–NB (<i>proposed</i>)	27.541	7.963	99.552	61.531	27.990	7.081	18.165	9.711
glmnet-P Lasso	67.136	18.402	298.960	244.177	66.991	16.278	47.737	26.423
glmnet-P EN	72.476	20.955	261.981	161.925	73.659	18.634	47.805	25.556
glmnet-NB Lasso (log1p)	97.998	54.284	542.985	433.925	117.036	46.416	49.767	28.614
glmnet-NB EN (log1p)	97.574	52.258	542.380	432.768	121.626	43.245	48.326	28.453
glmnet-P Ridge	121.645	44.233	230.290	125.252	145.205	49.092	38.933	22.900
glmnet-NB Ridge (log1p)	131.937	53.075	536.788	471.406	173.967	65.275	39.049	23.480
RF on log1p	172.799	75.977	302.533	229.229	252.268	103.301	45.287	27.452
XGBoost Poisson	218.268	88.949	486.032	337.948	321.780	126.596	53.489	34.329
ZIP (zeroinfl + SIS)	847.037	497.754	9077.179	20,355.349	1227.138	681.915	400.627	420.293
GLM: NB	∞	–	33,438.154	34,381.072	∞	–	∞	–
GLM: Poisson	∞	–	33,673.988	33,853.839	∞	–	∞	–

Figures 1 and 2 corroborate the findings in Tables 1–4. The proposed RF–Poisson and RF–NB demonstrated more stability across 100 simulation repetitions compared to other competing methods.

The evaluation of variable selection performance across linear data-generating mechanisms in Table 5 reveals a critical trade-off between power (the ability to detect true signals) and false discovery rate (FDR; the proportion of selected variables that are false positives), with the proposed Random Forest models, RF–Poisson and RF–NB, consistently demonstrating the most favorable and balanced overall performance. While the glmnet-P Lasso method occasionally achieved a marginally higher power in the Poisson scenario (0.728 vs. 0.767 for RF–Poisson), it did so at an untenable cost, exhibiting catastrophically high

FDR values between 0.82 and 0.86 across all conditions, meaning over 80% of its selected variables were spurious. In stark contrast, the proposed RF methods maintained strong power, often the highest or second highest, while simultaneously controlling FDR at substantially lower levels, typically between 0.32 and 0.43, effectively identifying true signals without being overwhelmed by noise. This pattern highlights a fundamental weakness of the penalized regression approaches in these settings: they lack the specificity to distinguish true signals from false ones under these data structures. The other benchmarks, including the standard RF on log1p, XGBoost, and the ZIP model, all suffered from critically high FDR (≥ 0.87) alongside lower power. The complete failure of the Ridge regression and the classical GLM approaches, which exhibited zero power or 100% FDR, underscores their total inadequacy for variable selection tasks. Thus, the proposed RF frameworks uniquely provide a robust and practical solution, successfully navigating the power-FDR trade-off to deliver reliable and interpretable variable selection.

When the data-generating processes incorporate nonlinearity and extreme sparsity, the variable selection problem becomes profoundly more difficult, and the performance gap between the proposed methods and the alternatives widens considerably, solidifying the superiority of the RF-based approaches in realistic, complex settings (see Table 6). In the highly challenging “Nonlinear NB” and “Ultra” sparse scenarios, all methods experienced a sharp decline in power; however, the proposed RF–Poisson and RF–NB models maintained a critical advantage by achieving the highest power amongst all methods while still exercising the most effective control over the false discovery rate. For instance, in the “Ultra” sparse setting, the RF–NB model achieved a power of 0.200, which was double that of the best glmnet Lasso model (0.080) and was coupled with a substantially lower FDR (0.277 vs. ≥ 0.592 for all glmnet variants). This trend is consistent across the other nonlinear scenarios: the proposed methods consistently rank as the top performers by effectively balancing the two metrics. The penalized regression methods (glmnet), while reasonably powerful in some nonlinear cases, completely failed to control FDR, with values frequently exceeding 0.80, rendering their selected variable sets practically useless. The standard RF on log1p and XGBoost performed dismally, with exceptionally high FDR often approaching 0.96 or even 0.98, indicating an almost random selection of variables. The catastrophic performance of the classical GLMs and the poor showing of the specialized ZIP model further emphasize that the proposed RF–Poisson and RF–NB frameworks are uniquely equipped to handle the complexities of variable selection in high-dimensional, nonlinear count data, offering a combination of sensitivity and specificity that is unmatched by any other method in the comparison.

The analysis of computational efficiency in Table 7 reveals a clear hierarchy of runtime performance, which must be interpreted in the crucial context of the previously established superior predictive accuracy and variable selection performance of the proposed methods. The glmnet-based methods, particularly the “glmnet-NB Lasso (log1p)” and the standard “RF on log1p”, are consistently the fastest algorithms, with mean runtimes often below 0.25 s, leveraging the computational efficiency of regularized linear models and a simple Gaussian regression forest. The proposed RF–Poisson and RF–NB methods occupy a middle ground in terms of computational cost, with runtimes typically between 0.6 and 0.7 s; while they are approximately 3–5 times slower than the fastest glmnet variants, they are still decidedly faster than the XGBoost Poisson implementation, which is the slowest method at over 1.2 s, and are comparable to the ZIP model. The moderate computational overhead of the proposed methods is directly attributed to the cost of their tailored likelihood-based splitting, which is the very mechanism that grants them their significant advantage in model fit and selection accuracy. The classical GLM: NB and GLM: Poisson models are relatively fast, but this metric is meaningless given their previously demonstrated complete

failure to provide useful predictions or variable selection. Therefore, the computational cost of the proposed RF methods is not only reasonable but also a worthwhile investment, representing an efficient trade-off for their substantial gains in statistical performance.

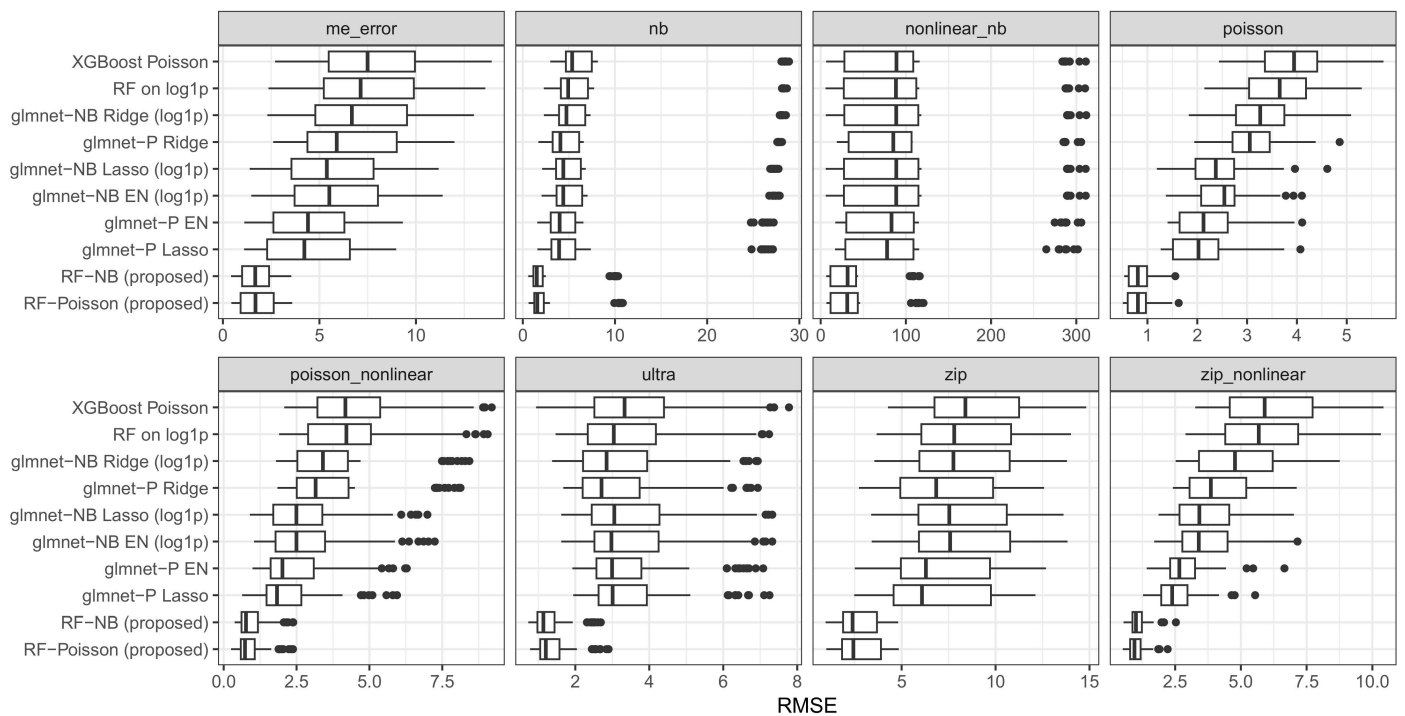


Figure 1. Comparison of predictive performance (RMSE) under linear and nonlinear data-generating processes: Poisson linear/nonlinear, linear/nonlinear Negative Binomial (NB), Zero-Inflated Poisson (ZIP) linear/nonlinear, ME in covariates, and ultra-complex settings. The black dots represent outliers.

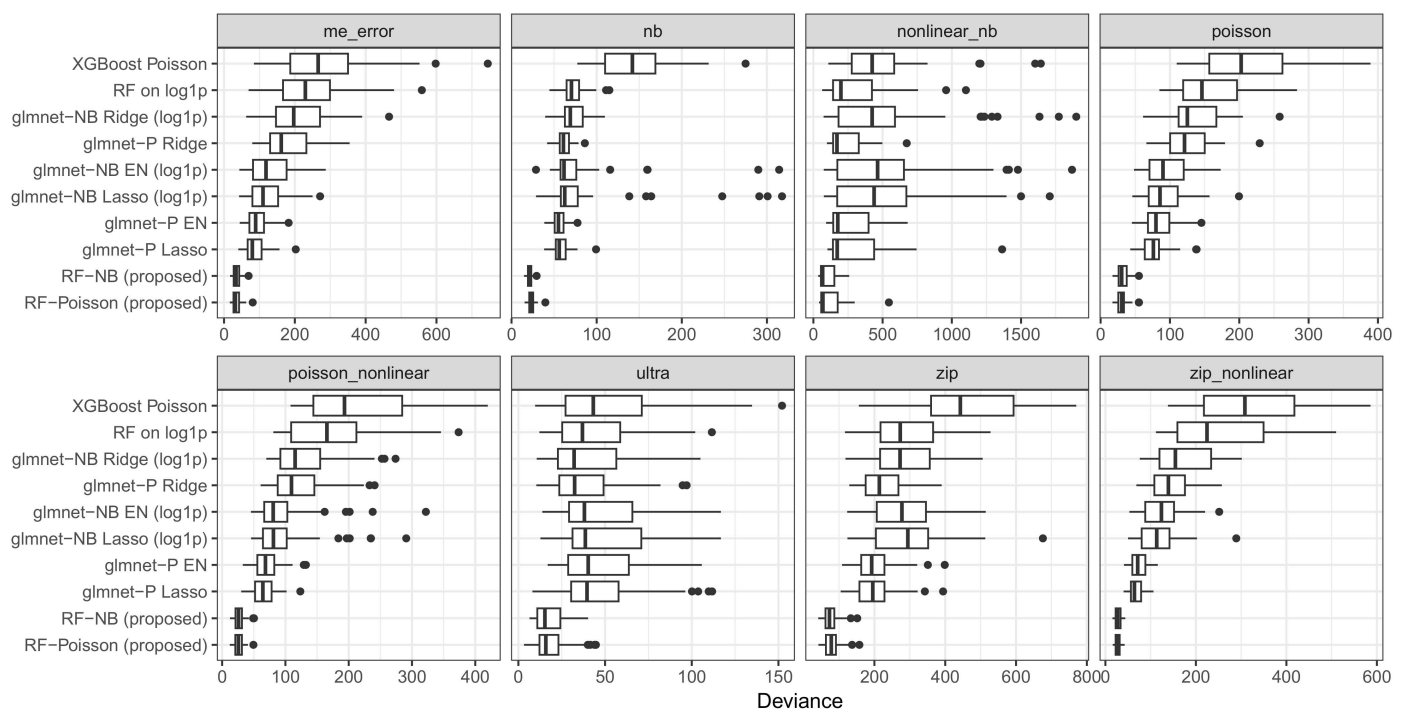


Figure 2. Comparison of predictive performance (deviance) under linear and nonlinear data-generating processes: Poisson linear/nonlinear, linear/nonlinear Negative Binomial (NB), Zero-Inflated Poisson (ZIP) linear/nonlinear, ME in covariates, and ultra-complex settings. The black dots represent outliers.

Table 5. Comparison of variable selection performance under complex, linear link data-generating models: Poisson, Negative Binomial (NB), ZIP, and measurement error (ME) in covariates. Reported are the mean and standard error (SE) of power and false discovery rate (FDR).

Method	Poisson				Nb				ZIP				ME in Covariates			
	Power		FDR		Power		FDR		Power		FDR		Power		FDR	
	Mean	SE	Mean	SE	Mean	SE	Mean	SE	Mean	SE	Mean	SE	Mean	SE	Mean	SE
RF–Poisson (<i>proposed</i>)	0.767	0.163	0.334	0.064	0.501	0.107	0.362	0.052	0.775	0.095	0.336	0.043	0.775	0.079	0.320	0.050
RF–NB (<i>proposed</i>)	0.673	0.155	0.422	0.045	0.480	0.129	0.433	0.050	0.736	0.084	0.407	0.032	0.732	0.105	0.407	0.035
glmnet-P Lasso	0.728	0.208	0.835	0.064	0.422	0.107	0.863	0.052	0.696	0.095	0.837	0.043	0.698	0.084	0.820	0.050
glmnet-P EN	0.614	0.187	0.873	0.045	0.400	0.129	0.884	0.050	0.656	0.084	0.857	0.032	0.652	0.105	0.857	0.035
glmnet-NB Lasso (log1p)	0.598	0.188	0.867	0.057	0.354	0.123	0.905	0.046	0.406	0.136	0.867	0.034	0.654	0.068	0.775	0.047
glmnet-NB EN (log1p)	0.492	0.194	0.898	0.045	0.344	0.136	0.912	0.047	0.426	0.138	0.884	0.027	0.652	0.068	0.823	0.044
glmnet-P Ridge	0.000	0.000	1.000	0.000	0.000	0.000	1.000	0.000	0.100	0.000	0.980	0.000	0.000	0.000	1.000	0.000
glmnet-NB Ridge (log1p)	0.000	0.000	1.000	0.000	0.000	0.000	1.000	0.000	0.100	0.000	0.980	0.000	0.000	0.000	1.000	0.000
RF on log1p	0.582	0.075	0.884	0.015	0.392	0.075	0.922	0.015	0.292	0.090	0.942	0.018	0.526	0.049	0.895	0.010
XGBoost Poisson	0.452	0.109	0.910	0.022	0.220	0.103	0.956	0.021	0.318	0.108	0.936	0.022	0.472	0.093	0.906	0.019
ZIP (zeroinfl + SIS)	0.630	0.071	0.874	0.014	0.396	0.070	0.921	0.014	0.208	0.083	0.958	0.017	0.414	0.045	0.917	0.009
GLM: NB	0.000	0.000	1.000	0.000	0.000	0.000	1.000	0.000	0.100	0.000	0.980	0.000	0.000	0.000	1.000	0.000
GLM: Poisson	0.000	0.000	1.000	0.000	0.000	0.000	1.000	0.000	0.100	0.000	0.980	0.000	0.000	0.000	1.000	0.000

Table 6. Comparison of variable selection performance under complex, nonlinear data-generating models: Poisson nonlinear, nonlinear Negative Binomial (NB), ZIP nonlinear, and an ultra-sparse high-dimensional setting. Reported are the mean and standard error (SE) of power and false discovery rate (FDR).

Method	Poisson Nonlinear				Nonlinear NB				ZIP Nonlinear				Ultra			
	Power		FDR		Power		FDR		Power		FDR		Power		FDR	
	Mean	SE	Mean	SE	Mean	SE	Mean	SE	Mean	SE	Mean	SE	Mean	SE	Mean	SE
RF–Poisson (<i>proposed</i>)	0.812	0.119	0.323	0.050	0.191	0.069	0.300	0.300	0.774	0.118	0.355	0.031	0.159	0.105	0.091	0.456
RF–NB (<i>proposed</i>)	0.714	0.135	0.416	0.036	0.208	0.103	0.362	0.331	0.622	0.120	0.441	0.024	0.200	0.120	0.277	0.414
glmnet-P Lasso	0.734	0.121	0.823	0.050	0.112	0.069	0.801	0.300	0.702	0.129	0.856	0.031	0.080	0.105	0.592	0.456
glmnet-P EN	0.634	0.135	0.866	0.036	0.128	0.103	0.812	0.331	0.542	0.120	0.892	0.024	0.120	0.120	0.727	0.414
glmnet-NB Lasso (log1p)	0.664	0.164	0.847	0.054	0.172	0.136	0.493	0.373	0.690	0.171	0.843	0.052	0.086	0.105	0.597	0.459
glmnet-NB EN (log1p)	0.548	0.153	0.886	0.038	0.186	0.150	0.532	0.396	0.546	0.204	0.887	0.048	0.114	0.120	0.632	0.444
glmnet-P Ridge	0.100	0.000	0.980	0.000	0.000	0.000	1.000	0.000	0.200	0.000	0.960	0.000	0.000	0.000	1.000	0.000
glmnet-NB Ridge (log1p)	0.100	0.000	0.980	0.000	0.000	0.000	1.000	0.000	0.200	0.000	0.960	0.000	0.000	0.000	1.000	0.000
RF on log1p	0.554	0.093	0.889	0.019	0.418	0.072	0.916	0.014	0.634	0.069	0.873	0.014	0.182	0.085	0.964	0.017
XGBoost Poisson	0.448	0.081	0.910	0.016	0.352	0.105	0.930	0.021	0.408	0.097	0.918	0.019	0.100	0.097	0.980	0.019
ZIP (zeroinfl + SIS)	0.444	0.058	0.911	0.012	0.170	0.071	0.966	0.014	0.546	0.079	0.891	0.016	0.118	0.080	0.941	0.040
GLM: NB	0.100	0.000	0.980	0.000	0.000	0.000	1.000	0.000	0.200	0.000	0.960	0.000	0.000	0.000	1.000	0.000
GLM: Poisson	0.100	0.000	0.980	0.000	0.000	0.000	1.000	0.000	0.200	0.000	0.960	0.000	0.000	0.000	1.000	0.000

The runtime performance under nonlinear and ultra-complex data-generating processes, presented in Table 8, reinforces the patterns observed in the linear scenarios and further contextualizes the value proposition of the proposed methods. The efficiency ranking remains largely consistent: the “glmnet-NB Lasso/EN” variants and the standard “RF on log1p” are the fastest algorithms, often completing in well under 0.2 s. The proposed RF–Poisson and RF–NB methods again demonstrate moderate and stable computational requirements, with runtimes clustering around 0.65–0.85 s for standard nonlinear problems and dropping to a very efficient 0.24 s in the “Ultra” sparse high-dimensional setting, which is likely due to the early stopping mechanisms in tree building being triggered more quickly when few true signals are present. This stability in diverse and complex scenarios is a key strength, indicating their robustness. In stark contrast, the “ZIP (zeroinfl + SIS)” model shows severe computational degradation in the “Ultra” setting, with its runtime soaring to over 2 s, suggesting that it struggles significantly with model fitting and variable screening in high-dimensional complexity. The XGBoost Poisson implementation remains the slowest consistent performer. When this computational profile is viewed alongside the previously demonstrated best-in-class predictive and variable selection performance, the conclusion is unequivocal: the proposed Random Forest methods provide a computationally feasible and efficient pathway to achieving state-of-the-art results, offering a massively superior statistical return on a relatively modest and manageable investment of computational resources.

Table 7. Comparison of computational time (secs) of proposed Random Forest count regression models (RF–Poisson and RF–NB) against penalized regression (glmnet variants), XGBoost, classical Zero-Inflated Poisson (ZIP), and GLM baselines across Poisson, Negative Binomial (NB), Zero-Inflated Poisson (ZIP), and measurement error (ME) in covariates and linear link scenarios.

Method	Poisson		NB		ZIP		ME in Covariates	
	Mean	SE	Mean	SE	Mean	SE	Mean	SE
RF–Poisson (<i>proposed</i>)	0.609	0.022	0.712	0.251	0.611	0.069	0.704	0.036
RF–NB (<i>proposed</i>)	0.615	0.020	1.069	2.610	0.623	0.082	0.714	0.046
glmnet-P Lasso	0.241	0.017	0.300	0.113	0.255	0.085	0.165	0.018
glmnet-P EN	0.228	0.017	0.286	0.097	0.235	0.033	0.173	0.025
glmnet-NB Lasso (log1p)	0.141	0.019	0.181	0.064	0.155	0.021	0.125	0.018
glmnet-NB EN (log1p)	0.164	0.160	0.187	0.068	0.171	0.082	0.138	0.077
glmnet-P Ridge	0.369	0.078	0.421	0.134	0.392	0.081	0.382	0.073
glmnet-NB Ridge (log1p)	0.224	0.026	0.243	0.066	0.223	0.025	0.221	0.024
RF on log1p	0.108	0.007	0.120	0.033	0.107	0.013	0.117	0.014
XGBoost Poisson	1.205	0.028	1.324	0.231	1.221	0.083	1.223	0.067
ZIP (zeroinfl + SIS)	0.743	0.094	0.918	0.346	0.781	0.059	0.768	0.036
GLM: NB	0.166	0.022	0.195	0.080	0.173	0.022	0.172	0.022
GLM: Poisson	0.154	0.027	0.192	0.100	0.166	0.027	0.156	0.022

Table 8. Comparison of computational time (secs) under nonlinear data-generating processes: Poisson nonlinear, nonlinear Negative Binomial (NB), Zero-Inflated Poisson (ZIP) nonlinear, and ultra-complex settings. Reported values are mean prediction errors and their standard errors across competing methods.

Method	Poisson Nonlinear		Nonlinear NB		ZIP Nonlinear		Ultra	
	Mean	SE	Mean	SE	Mean	SE	Mean	SE
RF–Poisson (<i>proposed</i>)	0.641	0.031	0.844	0.158	0.676	0.029	0.238	0.018
RF–NB (<i>proposed</i>)	0.675	0.041	0.844	0.175	0.693	0.029	0.241	0.023
glmnet-P Lasso	0.224	0.032	0.202	0.057	0.207	0.022	0.190	0.103
glmnet-P EN	0.223	0.102	0.206	0.079	0.210	0.024	0.189	0.099
glmnet-NB Lasso (log1p)	0.142	0.024	0.175	0.026	0.140	0.023	0.125	0.069

Table 8. Cont.

Method	Poisson Nonlinear		Nonlinear NB		ZIP Nonlinear		Ultra	
	Mean	SE	Mean	SE	Mean	SE	Mean	SE
glmnet-NB EN (log1p)	0.145	0.027	0.166	0.021	0.147	0.026	0.136	0.074
glmnet-P Ridge	0.394	0.039	0.365	0.043	0.400	0.038	0.721	0.038
glmnet-NB Ridge (log1p)	0.228	0.039	0.239	0.092	0.245	0.089	0.496	0.098
RF on log1p	0.110	0.007	0.125	0.016	0.113	0.009	0.091	0.008
XGBoost Poisson	1.210	0.042	1.284	0.072	1.221	0.026	1.157	0.030
ZIP (zeroinfl + SIS)	0.800	0.030	0.855	0.068	0.838	0.089	2.075	0.098
GLM: NB	0.182	0.026	0.180	0.090	0.183	0.024	0.456	0.167
GLM: Poisson	0.166	0.029	0.159	0.054	0.170	0.025	0.332	0.072

5. Application to Norwegian Mother and Child Cohort Study

The Norwegian Mother and Child Cohort Study (MoBa) [34] was a large population-based pregnancy study conducted between 1999 and 2005. As a sub-study, gene expression was measured from the umbilical cord blood of 200 newborns. After standard quality control and data processing steps, gene expression data from 111 of these samples were successfully profiled using Agilent microarrays. For an even smaller subset of 29 children, a specific biomarker of genotoxicity called micronucleus frequency (MN) was also measured. Both the gene expression (GSE31836) and MN data are publicly available from the NCBI Gene Expression Omnibus (GEO) database. Figure 3 presents the histogram for the empirical distribution, while the colored curves show the fitted model expectations for comparison of fit under different assumptions.

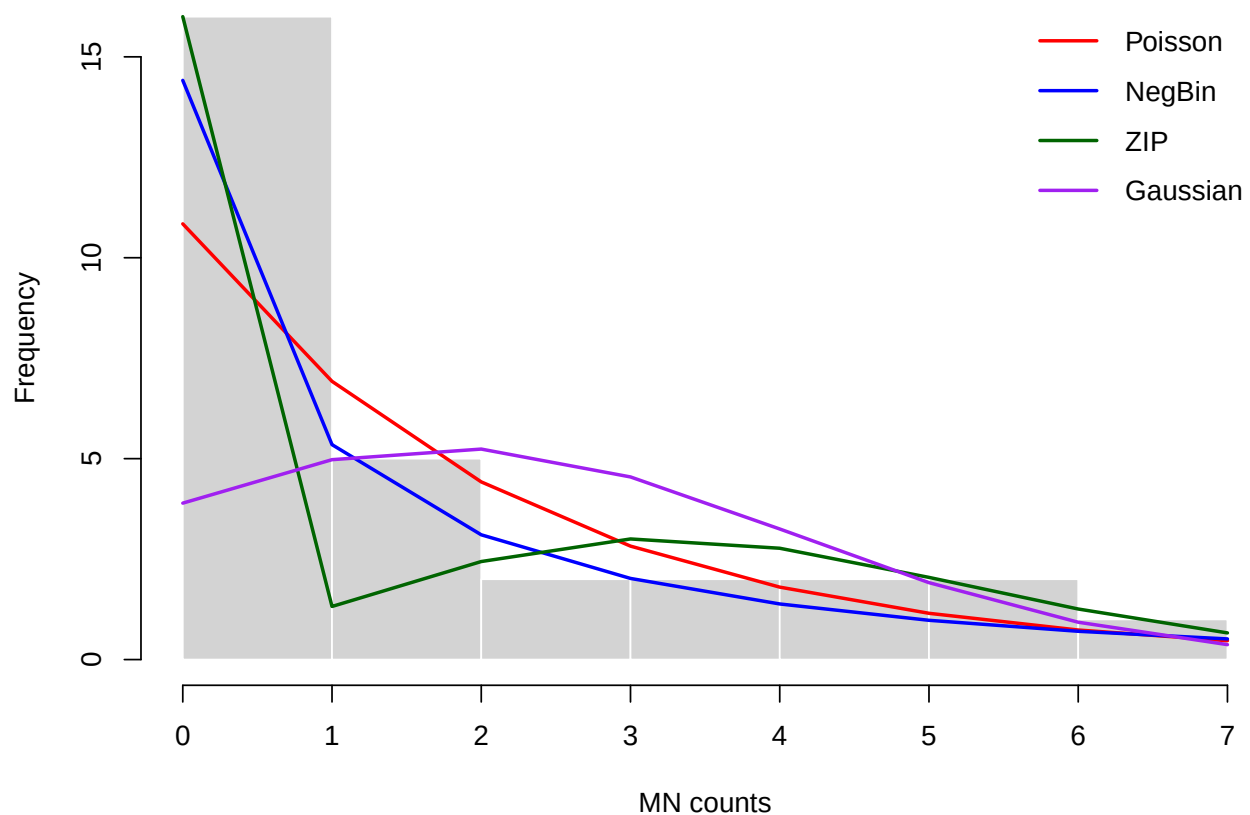


Figure 3. Observed distribution of micronuclei counts with overlaid expected frequencies from Poisson, Negative Binomial, Zero-Inflated Poisson (ZIP), and Gaussian models.

The data were preprocessed by extracting the GSE31836 expression dataset from GEO that was originally collected by [35], which contains 41,000 gene expression profiles in 111 samples, although micronucleus count data (MN) were only available for about 29 samples. To ensure reliable inputs, genes with missing values were removed, resulting in a dataset of 6797 complete genes. The dependent variable was defined as the MN counts, while the predictor matrix was restricted to the most variable genes, since high-variance genes are more likely to capture meaningful biological signals. Specifically, variance filtering was applied, and the top 2000 most variable genes were retained to balance information content with computational feasibility. Finally, the filtered expression dataset was transposed and matched to the subset of samples with MN counts. The final dataset used for the comparative benchmark analysis comprises 29 samples with 2000 genes.

To evaluate the predictive performance of each model on this real-world dataset, we used a k -fold repeated cross-validation scheme. The entire process was repeated 20 times to ensure stable estimates. For each repeat, the data were randomly partitioned into 5 folds. Each method was trained on 4 folds, and its predictions were generated for the held-out test fold. The root mean squared error (RMSE) and model-specific deviance were calculated on these out-of-sample predictions. This procedure was iterated until every fold had served as the test set once, yielding 5 performance estimates per repeat. The final reported metrics for each model in Table 9 represent the mean of these 100 individual estimates (5 folds $\times$ 20 repeats), providing a robust and unbiased assessment of their generalizable predictive accuracy.

Table 9. Comparison of methods on RMSE, deviance, time, and non-zeros (with SE) for the prediction of MN from the Norwegian Mother and Child Cohort Study (MoBa).

Method	RMSE (SE)	Deviance (SE)	Time (SE)	Non-Zero (SE)
RF–Poisson (<i>proposed</i>)	2.097 (0.443)	16.940 (3.976)	0.791 (0.065)	35.68 (11.349)
RF–NB (<i>proposed</i>)	2.102 (0.456)	17.027 (4.123)	0.791 (0.068)	35.62 (11.735)
RF on log1p	2.150 (0.696)	19.099 (7.024)	0.096 (0.007)	1143.38 (60.157)
glmnet–NB Ridge (log1p)	2.235 (0.703)	18.285 (7.197)	0.474 (0.135)	1991.62 (10.768)
glmnet–P Ridge	2.332 (0.677)	18.511 (4.875)	0.672 (0.085)	1993.32 (6.441)
glmnet–NB EN (log1p)	2.384 (0.755)	26.360 (35.239)	0.114 (0.078)	18.98 (12.679)
glmnet–NB Lasso (log1p)	2.403 (0.860)	27.176 (36.188)	0.104 (0.082)	11.20 (8.305)
XGBoost Poisson	2.408 (0.814)	34.600 (20.190)	0.554 (0.044)	218.78 (23.633)
glmnet–P EN	2.461 (1.248)	20.523 (8.220)	0.147 (0.078)	13.26 (10.349)
glmnet–P Lasso	2.492 (1.323)	20.817 (9.931)	0.141 (0.092)	6.02 (4.345)
GLM: NB	–	–	0.436 (0.086)	23.00 (0.000)
GLM: Poisson	–	–	0.375 (0.115)	23.00 (0.000)
ZIP (zeroinfl + SIS)	–	–	2.056 (0.140)	12.00 (0.000)

Table 9 provides a critical validation of the proposed methods, demonstrating their practical utility and superior performance in a challenging $n \ll p$ setting with 29 samples and 2000 genes. The proposed RF–Poisson and RF–NB models consistently outperform all competing methods in the most important statistical metrics, achieving the lowest root mean squared error (RMSE of 2.10) and the lowest deviation (17.0), which indicates that they provide the best predictive accuracy and the closest model fit to observed micronuclei count data. This superior performance is particularly notable compared to the standard ‘RF on log1p’, which, while computationally faster (0.096 s), suffers from a substantially higher error (RMSE 2.15) and a critical lack of interpretability by selecting a nonsensically high number of variables (1143 genes), effectively failing to perform any meaningful variable selection. The penalized regression models (glmnet) present a mixed picture; while the Lasso and Elastic-Net variants produce very sparse models (selecting between 6 and 19 genes), this parsimony comes at a severe cost to predictive accuracy, resulting in the highest RMSE values in the study (≥ 2.38). In contrast, the Ridge regression models, which

keep all variables, perform moderately but are still outperformed by the proposed methods. The implementation of XGBoost Poisson performs poorly with a high level of error and the second-highest deviation. The classical “GLM” and ZIP models failed to produce valid results for RMSE and deviance, with the ZIP model also computationally being the slowest. Therefore, the proposed RF–Poisson and RF–NB frameworks strike an optimal balance in this real data application: deliver the best predictive performance, achieve an excellent model fit, and produce a highly interpretable and biologically plausible model by selecting a manageable subset of 35–36 potential genetic predictors, all within a computationally feasible time-frame of less than 0.8 s, establishing them as the most robust and effective choice for analyzing high-dimensional count data.

Biological Insights

Figure 4 provides a critical evaluation of the stability and biological reproducibility of the variables (genes) selected by each method, which is a paramount concern in high-dimensional genomic analysis where the goal is to identify robust biomarkers rather than artifacts of statistical noise.

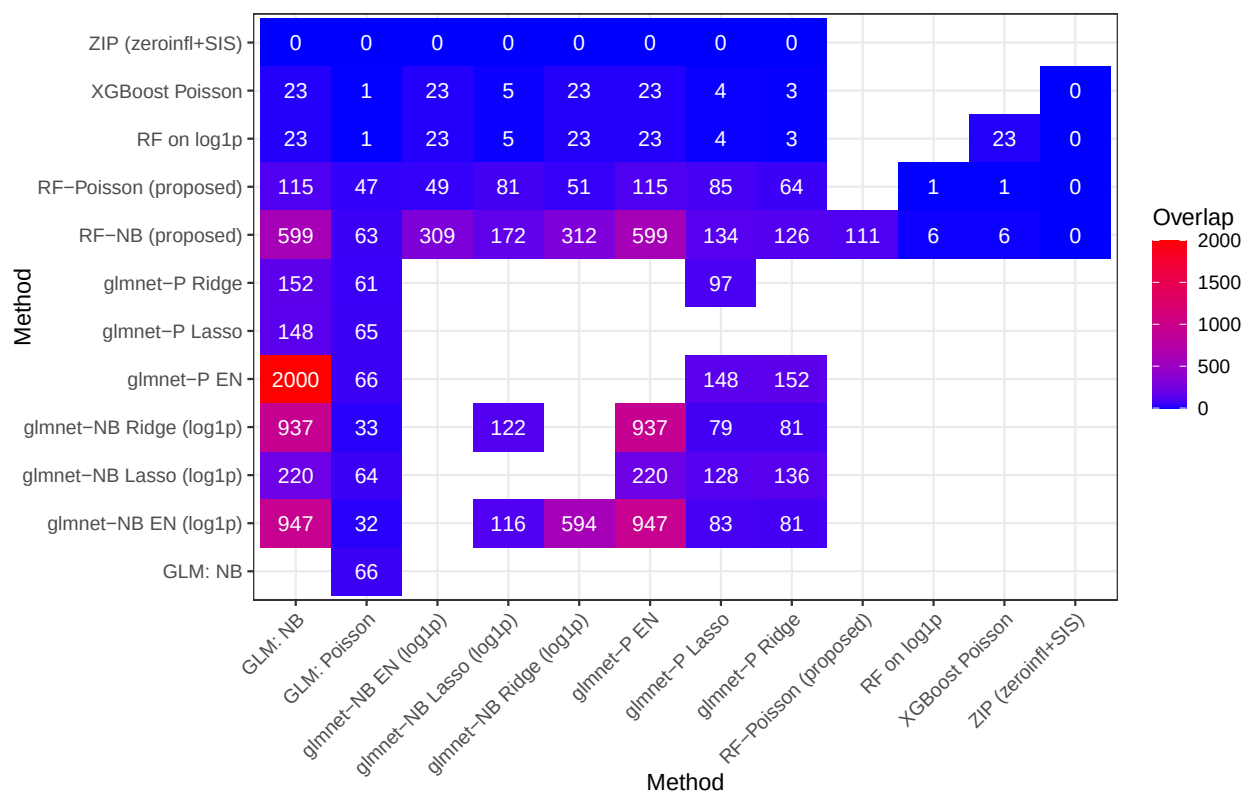


Figure 4. Overlapping of variable among top selected genes across methods.

The most striking result is the exceptional performance of the proposed RF–NB and RF–Poisson methods. Their bars tower over all others, with RF–NB identifying nearly 600 overlapping genes and RF–Poisson identifying over 100. This indicates that, when the analysis is repeated on different subsets of the data (e.g., via cross-validation), these two methods consistently prioritize the same set of top genes. This high degree of stability is a hallmark of a robust method and suggests that the biological signals these genes represent are strongly reproducible.

In stark contrast, nearly all other methods demonstrate profoundly unstable feature selection. The standard RF on log1p and XGBoost Poisson (while they are powerful predictive tools) show almost no overlap (a maximum of 23 genes), meaning their selected

features change drastically with slight changes in the input data. This makes their results nearly impossible to interpret biologically, as there is no consistent genetic signature to investigate. Similarly, the various glmnet models (Lasso, EN, Ridge) exhibit low to moderate overlap, despite being designed for variable selection. Their instability in this high-dimensional, real-world setting suggests that they are highly sensitive to correlation and noise in genomic data, leading to inconsistent results. The ZIP model and classical GLMs fail entirely, identifying zero or negligible overlapping features, rendering their selections biologically uninterpretable.

6. Discussion of Results

The empirical results of this study consistently demonstrate that the proposed Random Forest frameworks for Poisson and Negative Binomial responses effectively bridge a critical methodological gap in high-dimensional count regression identified in the previous literature. Where traditional penalized GLMs (e.g., Lasso, Elastic-Net) [19,23] offer interpretability through sparsity but falter due to their rigid linearity assumption and instability in feature selection under correlation as noted by [8], and where standard nonparametric ensembles (e.g., RF on log1p, XGBoost) [29,33] offer predictive flexibility at the cost of biological interpretability and feature stability, our proposed methods synthesize the strengths of both paradigms. The superior and stable predictive accuracy, quantified by lower RMSE and deviance across both simulated and real data, underscores their ability to capture the complex, nonlinear dependencies inherent in modern high-dimensional datasets—such as gene-expression interactions that linear predictors invariably miss as discussed by [11,34]. Furthermore, exceptional stability in variable selection provides a level of reproducibility that addresses the critical limitation of instability in high-dimensional feature selection raised by [7], contrasting sharply with the extreme instability of other ensembles and the often overly sparse or inconsistent selections from penalized regressions.

This performance achievement aligns with but substantially extends the call for specialized methods that respect the probabilistic nature of count data made by [5,9]. The integration of likelihood-based splitting criteria tailored to the respective mean variance structures allows the forests to overcome the equidispersion limitation of Poisson models noted by [1] while avoiding the pitfalls of naïve transformations that distort error structures. In the context of the existing literature, our work thus moves beyond simply applying machine learning to count data, instead providing a rigorously validated tool that addresses the dual challenges of predictive performance and reliable inference in high-dimensional settings as envisioned by [18], particularly for biological applications where stable feature selection is paramount for generating testable hypotheses as demonstrated in our MoBa study analysis.

7. Conclusions

In conclusion, this study establishes that the proposed Random Forest frameworks represent a significant and practical advancement for high-dimensional count regression, directly addressing a critical gap in the methodological literature. Existing approaches force a trade-off: penalized regressions offer interpretability through sparsity but fail to capture the complex nonlinearities and interactions inherent in modern datasets, while standard machine learning ensembles offer predictive flexibility at the cost of ignoring the fundamental distributional characteristics of count data, such as overdispersion, leading to unstable and biologically uninterpretable results. Our work bridges this divide by harmonizing the predictive power of nonparametric ensembles with the model-specific rigor of parametric methods, thus responding directly to the call for methods that can handle complex dependencies while maintaining probabilistic coherence, as noted by [14]. The results confirm that these tailored algorithms provide an essential tool for researchers in fields like genomics

and epidemiology, where the fundamental challenge of accurately modeling complex count data with thousands of correlated predictors, as described by Robinson and Smyth [2], necessitates a method that is both statistically robust and practically interpretable.

The practical importance and applicability of our introduced method are demonstrated by its superior performance across all evaluated metrics. In simulation studies, the proposed frameworks decisively achieved a significantly lower prediction error (RMSE) and a better model fit (deviance) than all benchmarks, including penalized regressions and alternative ensembles. Crucially, they also provided exceptionally stable and interpretable feature selection, maintaining a high power to detect true signals while effectively controlling the false discovery rate, a balance that other methods consistently failed to achieve. This combination of high predictive accuracy and reliable variable identification makes the method immediately applicable to real-world decision making, such as biomarker discovery and risk assessment, where generating reliable and actionable insights from high-dimensional data is crucial. The successful application to the Norwegian Mother and Child Cohort Study further underscores the utility of our method, which not only delivered the most accurate predictions but also identified a stable, biologically plausible set of genetic features associated with micronuclei frequency.

Despite these strengths, we acknowledge certain limitations that also chart a course for future work to further contribute to the field of statistical analysis. The computational overhead of the proposed forests, though reasonable, is higher than that of ultra-fast penalized regressions [19] and could hinder the application to extremely large-scale problems. Furthermore, the current implementation lacks native handling of zero inflation, which represents a limitation for applications in ecology and microbiology where such models are crucial [13]. Future research will therefore focus on developing an integrated Zero-Inflated Random Forest model and on extending the framework to accommodate structured penalties for grouped features, as proposed by [25], thus enhancing its utility and applicability to an even broader range of scientific questions. Ultimately, by providing a versatile and rigorously validated tool that respects the unique characteristics of count data, this research contributes a fundamental and impactful component to the analytical toolkit for modern high-dimensional statistical analysis. Furthermore, while the proposed Random Forest is presented as a powerful standalone method, its predictions could serve as a valuable base learner in a stacked ensemble with other algorithms, such as XGBoost, to potentially yield further gains in predictive accuracy for specific applications; this remains another exciting avenue for future research.

Author Contributions: Conceptualization, O.R.O., S.F.O., A.R.R.A., N.M.A., and A.A.A.; methodology, O.R.O., S.F.O., and N.M.A.; software, O.R.O. and S.F.O.; validation, O.R.O., S.F.O., A.R.R.A., N.M.A., and A.A.A.; formal analysis, O.R.O. and S.F.O.; investigation, O.R.O., S.F.O., A.R.R.A., N.M.A., and A.A.A.; resources, N.M.A., A.R.R.A., and A.A.A.; data curation, O.R.O. and S.F.O.; writing—original draft preparation, O.R.O.; writing—review and editing, O.R.O., S.F.O., A.R.R.A., N.M.A., and A.A.A.; visualization, O.R.O.; supervision, O.R.O.; project administration, O.R.O. All authors have read and agreed to the published version of the manuscript.

Funding: This research work was funded by Umm Al-Qura University, Saudi Arabia under grant number: 25UQU4320088GSSR04.

Data Availability Statement: The original data presented in the study are openly available in <https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSM789636>.

Acknowledgments: The authors extend their appreciation to Umm Al-Qura University, Saudi Arabia, for funding this research work through grant number: 25UQU4320088GSSR04.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations and Key Symbols

A. Textual Abbreviations

Abbreviation	Full Term	Definition/Explanation
DGP	Data-Generating Process	The underlying model and parameters used to simulate data for a simulation study.
EN	Elastic-Net	A penalized regression method that combines the L1 (Lasso) and L2 (Ridge) penalties.
FDR	False Discovery Rate	The proportion of selected variables that are false positives.
GEO	Gene Expression Omnibus	A public functional genomics data repository.
GLM	Generalized Linear Model	A flexible generalization of ordinary linear regression.
GWAS	Genome-Wide Association Study	A study to find genetic variations associated with a particular trait.
Lasso	Least Absolute Shrinkage and Selection Operator	A regression method that performs variable selection and regularization.
MLE	Maximum Likelihood Estimation	A method for estimating the parameters of a statistical model.
MN	Micronucleus (frequency)	A biomarker used to measure genotoxic exposure.
MoBa	Norwegian Mother and Child Cohort Study	A large population-based pregnancy cohort study.
MSE	Mean Squared Error	The average of the squares of the errors.
NB	Negative Binomial	A probability distribution for overdispersed count data.
PCA	Principal Component Analysis	A technique for reducing the dimensionality of large datasets.
PLS	Partial Least Squares	A statistical method for projecting predicted and observable variables to a new space.
RF	Random Forest	An ensemble learning method based on decision trees.
Ridge	Ridge Regression	A regression method for scenarios with highly correlated independent variables.
RMSE	root mean squared error	The square root of the MSE.
SIS	Sure Independence Screening	A variable screening technique for ultrahigh-dimensional settings.
ZIP	Zero-Inflated Poisson	A mixture model for count data with an excess of zero counts.

B. Key Mathematical Symbols

Symbol	Description	Typical Context
B	Number of trees in the ensemble.	Algorithm hyperparameter.
n	Sample size (number of observations).	Data dimension: $\mathbf{y} \in \mathbb{R}^n$.
p	Number of features (predictors).	Data dimension: $\mathbf{X} \in \mathbb{R}^{n \times p}$.
s	Sparsity level; number of true active (non-zero) coefficients.	$ \mathcal{S} = s$, where $\mathcal{S}$ is the active set.
Y_i	The random variable representing the count response for the i -th observation.	$Y_i \mid \mathbf{X}_i \sim \text{Poisson}(\mu(\mathbf{X}_i))$
$\mathbf{X}_i$	The feature vector for the i -th observation.	$\mathbf{X}_i \in \mathbb{R}^p$
$\mu(\mathbf{x})$	The conditional mean function, $\mathbb{E}[Y \mid \mathbf{X} = \mathbf{x}]$.	$\mu(\mathbf{x}) = \exp(f(\mathbf{x}))$
$f(\mathbf{x})$	The predictor function on the log-scale (log-link).	The primary target of estimation.
$\boldsymbol{\beta}$	Coefficient vector in a (penalized) generalized linear model.	$f(\mathbf{x}) = \boldsymbol{\beta}_0 + \mathbf{x}^\top \boldsymbol{\beta}$
θ	Dispersion parameter in the Negative Binomial distribution.	$\text{Var}(Y) = \mu + \mu^2 / \theta$
λ	Regularization (penalty) parameter in penalized regression.	Tunes the strength of shrinkage in Lasso, Ridge, Elastic-Net.
Σ	Covariance matrix of the features.	$\mathbf{X} \sim \mathcal{N}(\mathbf{0}, \Sigma)$

ρ	Correlation coefficient.	Used to define the block-correlation structure in simulations.
D	deviance; a measure of model fit based on the log-likelihood.	Splitting criterion for the proposed Random Forest.
$\mathcal{S}$	The true active set of variable indices.	$\mathcal{S} = \{j : \beta_j \neq 0\}$
$\hat{\mathcal{S}}$	The estimated active set of variable indices.	Output of variable selection procedures.
π_i	Zero-inflation probability in Zero-Inflated models.	$Y_i \sim 0$ with probability π_i , $\text{Poisson}(\mu_i)$ otherwise.

References

- Gardner, W.; Mulvey, E.P.; Shaw, E.C. Regression analyses of counts and rates: Poisson, overdispersed Poisson, and Negative Binomial models. *Psychol. Bull.* **1995**, *118*, 392. [\[CrossRef\]](#) [\[PubMed\]](#)
- Robinson, M.D.; Smyth, G.K. Small-sample estimation of Negative Binomial dispersion, with applications to SAGE data. *Biostatistics* **2008**, *9*, 321–332. [\[CrossRef\]](#) [\[PubMed\]](#)
- Simmachan, T.; Boonkrong, P. A Comparison of Count and Zero-Inflated Regression Models for Predicting Claim Frequencies in Thai Automobile Insurance. *Lobachevskii J. Math.* **2024**, *45*, 6400–6414. [\[CrossRef\]](#)
- Niku, J.; Hui, F.K.; Taskinen, S.; Warton, D.I. gllvm: Fast analysis of multivariate abundance data with generalized linear latent variable models in r. *Methods Ecol. Evol.* **2019**, *10*, 2173–2182. [\[CrossRef\]](#)
- Hilbe, J.M. *Negative Binomial Regression*, 2nd ed.; Cambridge University Press: Cambridge, UK, 2011.
- McCullagh, P.; Nelder, J.A. *Generalized Linear Models*, 2nd ed.; Chapman and Hall: Boca Raton, FL, USA, 1989.
- Bühlmann, P.; van de Geer, S. *Statistics for High-Dimensional Data: Methods, Theory and Applications*; Springer Science & Business Media: Berlin/Heidelberg, Germany, 2011.
- Hastie, T.; Tibshirani, R.; Wainwright, M. Statistical learning with sparsity. *Monogr. Stat. Appl. Probab.* **2015**, *143*, 8.
- Cameron, A.C.; Trivedi, P.K. *Regression Analysis of Count Data*, 2nd ed.; Cambridge University Press: Cambridge, UK, 2013.
- Lawless, J.F. Negative binomial and mixed Poisson regression. *Can. J. Stat. Rev. Can. Stat.* **1987**, *15*, 209–225. [\[CrossRef\]](#)
- Zilberman, O.; Abramovich, F. High-dimensional regression with a count response. *arXiv* **2024**, arXiv:2409.08821. [\[CrossRef\]](#)
- Fan, J.; Lv, J. A selective overview of variable selection in high dimensional feature space. *Stat. Sin.* **2010**, *20*, 101.
- Zhang, Y.; Zhou, H.; Zhou, J.; Sun, W. Regression models for multivariate count data. *J. Comput. Graph. Stat.* **2017**, *26*, 1–13. [\[CrossRef\]](#)
- Rahnenführer, J.; De Bin, R.; Benner, A.; Ambrogio, F.; Lusa, L.; Boulesteix, A.L.; Migliavacca, E.; Binder, H.; Michiels, S.; Sauerbrei, W.; et al. Statistical analysis of high-dimensional biomedical data: A gentle introduction to analytical goals, common approaches and challenges. *Bmc Med.* **2023**, *21*, 182. [\[CrossRef\]](#)
- Chamlal, H.; Benzmane, A.; Ouaderhman, T. Elastic net-based high dimensional data selection for regression. *Expert Syst. Appl.* **2024**, *244*, 122958. [\[CrossRef\]](#)
- Jia, J.; Xie, F.; Xu, L. Sparse Poisson regression with penalized weighted score function. *Electron. J. Statist.* **2019**, *13*, 2898–2920. [\[CrossRef\]](#)
- Fan, J.; Lv, J. Sure independence screening for ultrahigh dimensional feature space. *J. R. Stat. Soc. Ser. B Stat. Methodol.* **2008**, *70*, 849–911. [\[CrossRef\]](#) [\[PubMed\]](#)
- Cui, X.; Li, Y.; Fan, J.; Wang, T. A novel filter feature selection algorithm based on relief. *Appl. Intell.* **2022**, *52*, 5063–5081. [\[CrossRef\]](#)
- Tibshirani, R. Regression shrinkage and selection via the lasso. *J. R. Stat. Soc. Ser. B Stat. Methodol.* **1996**, *58*, 267–288. [\[CrossRef\]](#)
- Ivanoff, S.; Picard, F.; Rivoirard, V. Adaptive Lasso and group-Lasso for functional Poisson regression. *J. Mach. Learn. Res.* **2016**, *17*, 1–46.
- Zou, H. The adaptive lasso and its oracle properties. *J. Am. Stat. Assoc.* **2006**, *101*, 1418–1429. [\[CrossRef\]](#)
- Hoerl, A.E.; Kennard, R.W. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics* **1970**, *12*, 55–67. [\[CrossRef\]](#)
- Zou, H.; Hastie, T. Regularization and variable selection via the elastic net. *J. R. Stat. Soc. Ser. B Stat. Methodol.* **2005**, *67*, 301–320. [\[CrossRef\]](#)
- Algamal, Z.Y.; Lee, M.H. Adjusted adaptive lasso in high-dimensional poisson regression model. *Mod. Appl. Sci.* **2015**, *9*, 170. [\[CrossRef\]](#)
- Simon, N.; Friedman, J.; Hastie, T.; Tibshirani, R. A sparse-group lasso. *J. Comput. Graph. Stat.* **2013**, *22*, 231–245. [\[CrossRef\]](#)
- Tibshirani, R.; Saunders, M.; Rosset, S.; Zhu, J.; Knight, K. Sparsity and smoothness via the fused lasso. *J. R. Stat. Soc. Ser. B Stat. Methodol.* **2005**, *67*, 91–108. [\[CrossRef\]](#)
- Lin, Y.; Zhang, H.H. Component selection and smoothing in multivariate nonparametric regression. *Ann. Stat.* **2006**, *34*, 2272–2297. [\[CrossRef\]](#)
- Biau, G.; Scornet, E. A Random Forest guided tour. *Test* **2016**, *25*, 197–227. [\[CrossRef\]](#)

29. Breiman, L. Random forests. *Mach. Learn.* **2001**, *45*, 5–32. [[CrossRef](#)]
30. Olaniran, O.R.; Abdullah, M.A.A. Bayesian weighted Random Forest for classification of high-dimensional genomics data. *Kuwait J. Sci.* **2023**, *50*, 477–484. [[CrossRef](#)]
31. Olaniran, O.R.; Alzahrani, A.R.R. On the Oracle Properties of Bayesian Random Forest for Sparse High-Dimensional Gaussian Regression. *Mathematics* **2023**, *11*, 4957. [[CrossRef](#)]
32. Breiman, L. Statistical modeling: The two cultures. *Stat. Sci.* **2001**, *16*, 199–231. [[CrossRef](#)]
33. Chen, T.; Guestrin, C. XGBoost: A scalable tree boosting system. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, 13–17 August 2016; pp. 785–794.
34. Lehman, R.R.; Archer, K.J. Penalized Negative Binomial models for modeling an overdispersed count outcome with a high-dimensional predictor space: Application predicting micronuclei frequency. *PLoS ONE* **2019**, *14*, e0209923. [[CrossRef](#)]
35. Ramirez, A.; Saldanha, P.H. Micronucleus investigation of alcoholic patients with oral carcinomas. *Genet. Mol. Res.* **2002**, *1*, 246–260.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.