Journal of Innovative Construction and Petrochemical Technologies, Vol.3, No.2 (December, 2026), 398-418

P-ISSN: xxxx- xxxx | E-ISSN: 3007-9861 | DOI: https://doi.org/10.71285/icpt.v3i2.64

Journal of Innovative Construction and Petrochemical Technologies
https://icpt.ouk.kz/index.php/icpt/index



Research Article

Check for updates

Hurdle and Zero-Inflated Models vs. Machine Learning for Count Data with Excess Zeros: A Case Study on Epilepsy Patients

Wakaa Ali Hadba[1*]
* Correspondence: wakaa2017@uokirkuk.edu.iq

Abstract: In medicine, the analysis of zero-inflated count data is a challenging issue to analyze correctly. This kind of data structure makes the performance of traditional statistical models inaccurate. This paper contrasts the epileptic seizure predictive models based on traditional statistics or machine learning. Comparatively, there were zero-inflated and hurdle models. Thus, the purpose is to determine which approach is most effective when such data should be used to increase the predictability and achieve timely clinical intervention. We used four statistical models (Zero-Inflated Poisson (ZIP), Zero-Inflated Negative Binomial (ZINB), Hurdle Poisson and Hurdle Negative Binomial) and four machine learning algorithms (Support Vector Regression (SVR), Random Forest, XGBoost and a Neural Network) to achieve our goal. Comparison between the statistical and ML models was done using a clinical data on epileptic seizures. The analysis involved a set of overall measures, such as Root Mean Square error (RMSE), Mean Absolute error (MAE), and R2. The Akaike Information Criterion (AIC) and Bayesian Information Criterion (BIC) are also used to test the statistical model fit. The results revealed that although statistical models have good fit, they cannot predict (low R2). Machine learning algorithms on the other hand are far better at prediction. The best performer was SVR, which has the highest R2 of 0.69 and lowest MAE, but the Neural Network has also performed well. The research results in a conclusion that machine learning algorithms, specifically SVR and neural networks are more accurate and reliable in predicting epileptic seizures compared to the use of traditional statistics..

Keywords: Zero-Inflated Models, Hurdle model, Machine Learning in Healthcare, Epilepsy, neural network, Support Vector Regression.

Copyright © Author(s). This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY).

Introduction

Epilepsy is a condition of the nervous system that involves over 1 percent of the global population and can be treated using medicine or surgery [1]. Epilepsy is a highly prevalent nervous disease with a prevalence of over 65 million individuals in the world. Drug therapy, or surgical intervention, does not manage the disease in over 30 percent of cases. Nevertheless, prediction of seizures prior to occurrence may play a significant role in their prevention using proper therapeutic intervention. Research has shown that deviant brain

activity precedes the occurrence of a seizure by several minutes which is scientifically termed as pre-seizure period [1].

Count data is a statistical model that is used frequently in medicine and epidemiology. The dependent variable is defined as a non-negative integer, including the number of incidences of a certain disease or the number of episodes within a given period [2,3].

The selection of a suitable distribution of count data plays an important role in the analysis of data because its objective is to reflect the data correctly. Count data are commonly modeled by the Poisson distribution as a standard model, and in most cases is inadequate since it can be overdispersed or long-tailed. In order to correct overdispersion, alternative distributions have been used including the negative binomial distribution or the generalized Poisson distributions proposed by [4,5].

The large number of zeros in the data is one of the typical and characteristic types of overdispersion. In response to this, zero-infllation models have been come up to be used in cases where the data are characterized by a high number of zeros. Past studies had suggested that the failure to deal with excess zeros results in the large deficiencies in data models since the excess zeros as well as non-zero values are wrongfully estimated (Piroumian-Chani et al., 2013) [6, 3].

The two models are based on the assumption that the data are a combination of two distributions, the first one producing only zero values. Conversely, the latter gives non-negative values, which are usually a Poisson distribution or negative binomial distribution. Hence a zero-centered distribution in combination with another distribution involving the values of non-negative integer is the most suitable approach to modeling this kind of data. These mixed models are useful in working with highly dispersed data that contain many zeros, the count of which is inflated and are referred to as the zero-inflation models. One of the most popular models in this case is the zero-inflated Poisson model developed and characterized by (Lambert, 1992) [7, 6, 5]. Machine Learning (ML) algorithms have proven to be practical in the analysis of such data, particularly when nonlinear patterns are involved and very complex. The statistical inference of the general linear model (GLM) distribution of zero-inflated data is invalid due to misspecification. This happens, e.g., when a Poisson or negative binomial (NB) regression model is fitted using data being a zero-inflated Poisson or negative binomial (NB) distribution [7, 8]. Lambert (1992) [7] named the ZIP model, one of the most popular count data analysis models that contain too many zeros [7] and barrier models, such as those proposed by Molay (1986) [3], Heilbronn (1994) [10], Yip and Yao (2005) [11]. There is a tendency to use zero-inflated Poisson regression or negative binomial regression in the insurance field.

D. Erdman et al. (2008) [12] and S. M. Mwalili (2008) [13] applied ZINB to dental caries study. A popular dental caries research measure is the DMFT index, which is however complicated in distribution. Zero-inflated negative binomial (ZINB) distribution fits excellently. This was plotted on DMFT index. Mirowski et al. (2009) [14] utilized neural networks and deep learning to analyze EEG data containing zeros and noise. Seyed. S. & Robiah. A (2011) [15] constructed a zero-inflated negative binomial model of right-censored count data, and estimated the model parameters with maximum likelihood and assessed the accuracy of the estimation in the event of overdispersion through a simulation study (M. C. Hu, M. Pavlicova and E. V. Nunes (2011) [16]).

This study highlights that negative binomial and zero-inflated models provide better and more accurate estimates of the treatment effect for overdispersed count data with excess zeros [13]. We present a zero-inflated Poisson regression model, as explained [14], and Zero-inflation (ZI) models, as described [17]. In cytogenetic dosimetry, researchers found that zero-inflated Poisson and negative binomial models better handle diffuse and zero-inflated chromosomal aberration data than conventional Poisson regression[16]. Compare different methods for analyzing zero-inflated data and their application in health surveys.

A comparison of different methods of zero-inflated data analysis and an application in health surveys 2017 [18]. The authors evaluated the various models of counting claims such as the zero-inflated Poisson(ZIP) regression model and hurdle models using a backpropagation neural network. The analysis of the rainfall data was done using Poisson, Negative Binomial, ZIP, and ZINB models [19]. [1] presented a model of predicting epileptic seizures based on deep learning, scalp EEG signal preprocessing, and features extraction via the convolutional neural networks. [18] compared two types of exponential dispersion models of overdispersed and zero-inflated numerical data, and showed better flexibility and performance especially in data with high skew and kurtosis.[3] compared two zero-inflation and Hurdle model in the investigation of data with trailing zeroes, and illustrated that the choice of model is drawn on the model fit to the data set. [20] researched the ML techniques of early seizure prediction, feature selection, and better classification. EEG data was subjected to valid techniques [21].

S. Park and S. Jun (2023) [21] posed a solution to the scarcity issue and the improvement of the model performance in the text data analyses in the composite linear Poisson modeling. The method was superior to conventional models such as the linear, generalized linear and zero-inflated Poisson models. Banghee So (2024) [22] used boosting algorithm (XGBoost, LightGBM and CatBoost) to include the data of insurance claims (including zero-inflated telematics data) to construct their predictive models on the frequency of automotive claims. The results showed that CatBoost was better in comparison with the other two approaches and their analogs.

So and Valdez (2024) [23] put forward a hybrid structure which involves a combination of zero-infused distributions with CatBoost to suitably address extremely uncommon claim information, including that of seizures. The findings showed that CatBoost was more effective than XGBoost and LightGBM in automated claims modeling, and has a higher prediction accuracy. For zero-inflated data, M. Beveridge et al. (2024) [8] compared three models, among them ZINB, Hurdle NB, and truncated Gaussian Copula latent model. It can be noted that the authors used actual data. (microbiome and scRNA-seq). B. So and E. A. Valdez (2024) [23] introduced superior extensions of the Tweedy regression model, which relied on CatBoost and discrete zero-claim modeling, to achieve substantial predictive accuracy in the insurance claims research.

M. B. Ben Khedher and D. Yun (2024) [24] than the alternative performance of ML algorithms and classical counting models to analyze exaggerated accident data, which justify more specific interventions on traffic safety. In order to improve the performance of prediction, M. Martinović et al. (2025) [25] applied hierarchical modeling to exaggerated zero-sum data in real-world scenarios (home appliance classification and airport transportation requests). A two-stage hierarchical model combining gradient boosting and SVR/VGG proposed by J. M. Rožanec et al. (2025) [25]. The results showed a significant improvement in prediction accuracy and energy efficiency in zero-inflation scenarios (e.g., appliance usage and airport shuttle demand). Banghee So and Min Deng (2025) [26] compared the model of Tweedie with three ML algorithms XGBoost, LightGBM, and CatBoost (i.e. gradient boosting techniques). The model used by Tweedie was more effective in efficiency, the fitting accuracy and prediction accuracy.

This paper will look into zero-inflated Poisson (ZIP) model, the Hurdle Poisson (HurP) model, the zero-inflated negative binomial (ZINB) model and the Hurdle negative binomial (HurNB) model as applied in the case of insurance claims. The final goal will be to prove that the ZIP model could give the most appropriate fit to the insurance claims frequency data.

The remaining part of this work is arranged in the following manner. Section 2 presents the models applied in this work such as the statistical models and ML algorithms. Section 3 gives the metrics applied to evaluate the models. In Section 4, the dataset that was used to conduct the analysis is explained with the preprocessing steps. The results of

the models are discussed in section 5. Lastly, Section 6 provides the main conclusions and suggestions regarding future research.

Methods

Statistical Models

The Poisson regression is a model that assumes that the conditional mean is equal to the conditional variance, an assumption that is not always true. Assume that the data do not follow the Poisson model assumption of conditional variance. Then, there can be overdispersion as a result of heterogeneity of the population, clustering, outliers or lack of important covariates in the model [3].

Zero-Inflated Poisson Model

Count models with zero-infllation are a useful method of dealing with excess zeros in data. The data is modelled as a combination of two distributions: one of which is a count distribution; the other one is one distribution which assumes a single value of zero. As an example, the negative or Poisson binomial distributions.

Consider a zero-inflated Poisson (ZIP) regression model in which the response variable has the distribution model in which the response variable $Y_i (i = 1, \dots, n)$ has the distribution [15]. The mixed form of distribution of the ZIP provides p-value of the additional zero counts and (1-p) of the Poisson distribution. In case covariates are included, the ZIP distribution will give a ZIP regression model. The helpful aspect of the ZIP model of regression is that simultaneously, one can evaluate the influence of covariates on the additional zeros and the Poisson part of the model. The zero-inflated Poisson model is, however, inappropriate when the non-zero distribution is overdispersed in comparison with the Poisson distribution. The negative binomial component presumes that the Poisson mean is a random variable and it is distributed under a gamma distribution. ZIP model is developed as follows [3].

$$P(Yi = yi) = \begin{cases} \pi\, i\, + (1 - \pi i) e - \mu i \; if \; y\, i \; = \; 0, \\ (1 - \pi i) \dfrac{e^{-\mu} \mu^{y}}{yi!} \quad yi \; > \; 0 \end{cases} \tag{1}$$

$$E(yi) \; = \; E\,(E(yi|zi)) \; = \; (1 \; - \; \pi i)\mu i \tag{2}$$

$$Var(yi) \; = \; E(Var(yi|zi)) \; + \; Var(E(yi|zi)) \; = \; (1 \; - \; \pi i)\mu i\,(1 \; + \; \mu i \pi i) \tag{3}$$

Zero-Inflated Negative Binomial (ZINB) Model

This Section explains the Zero-Inflated Negative Binomial (ZINB) regression model. It is no secret that the ZIP distribution combines both the distribution which gives a mass of p to extra 0s, and a mass of (1-p) to a Poisson distribution. The presence of covariates causes ZIP to give a ZIP regression model. The additional benefit of the ZIP regression model is that the effects of the covariates can be estimated at the same time in the extra-zeros and the Poisson components of the model [3]. The part of the distribution that is not zero is spread out too widely to use a ZIP model in the comparison with a Poisson distribution. The negative binomial component makes the assumption that the Poisson mean is a random variable with a gamma distribution [13].

The ZINB distribution is a mixture distribution that allocates mass p to account for excess zeros (i.e., $0 \le p \le 1$). Nevertheless, to a negative binomial distribution, a mass of (1-p) is assigned [27]. The ZIP and NB distributions are approximated using the ZINB distribution as p and r tend to infinity. The parameters m and p are related to the covariates in the ZINB regression model as follows: log(mi) = x′ iβ and logit(pi) = z′ ir. Where i = 1, 2, ⋯, n. Additionally, zi and xi denote the predictors (vectors of covariates) for the logistic and NB components. Here, n is an independent variable. Of the regression coefficients, r and β are the corresponding vectors. Maximum likelihood estimation is used

to estimate the regression coefficients. For further information, we recommend that the reader visit [29] and [27].

The negative binomial model for excess zero-count data is the ZINB. In such cases, when overdispersion is due to both excess zeros and data heterogeneity, the ZINB regression model is often more suitable than the ZIP model.

We define the probability mass function of the response variable yi for the ZINB regression model as a random non-negative integer value. This assumes that yi = 0 is observed at a much higher frequency than can be modeled by the usual model [17][15]. The ZINB model can be defined as follows [3].

$$P(Y_i = y_i) = \begin{cases} \pi_i + (1-\pi_i)\left[\left(\frac{r}{\mu_i + r}\right)^r\right] & \text{if } y_i = 0 \\ (1-\pi_i)\frac{\Gamma(y_i + r)}{\Gamma(r) y_i!}\left(\frac{\mu_i}{\mu_i + r}\right)^{y_i}\left(\frac{r}{\mu_i + r}\right)^r & \text{if } y_i > 0 \end{cases}, \quad (4)$$

where r denotes the gamma function, π_i refers to the structural zero probability, and the mean of the NB model is defined by μ_i. The mean of the ZINB model is determined as $E(yi) = (1-\pi i)\mu i$, where the variance is $Var(yi) = (1-\pi i)\mu i(1 + \mu i/r + \pi i\mu i)$ [3]. Generally, logistic regression is used to model πi, and log-linear regression is utilized in modeling μi. The ZI model is formulated as follows.

$$\log(\mu_i) = x_i^T \alpha, \text{logit}(\pi_i) = z_i^T \beta \quad (5)$$

where the regression coefficients β and α belong to the variables z_i^T and x_i^T. It is worth mentioning that the variables of μ_i must be similar to those in π_i. Compared to the ZIP model, the ZINB model offers additional flexibility. It offers overdispersion due to excess zeros and heteroscedasticity in the Poisson component, whereas the ZIP model accommodates only overdispersion due to excess zeros [3].

Hurdle Poisson Model

For all non-zero observations, it models a zero-truncated part and all zeros as a single part. Hurdle models do not distinguish between true and false zeros. Furthermore, all zero observations originate from a non-susceptible group, which is the primary distinction between them and the ZIP model [18]. In contrast to ZI models, barrier models (as in [10] and [28]) are a two-component mixture model. The two parts consist of a positive feedback component and a zero-mass component, both following a truncated distribution, such as a truncated NB distribution or a truncated Poisson distribution. The hurdle model is formulated as follows [3].

$$P(Y_i = y_i) = \begin{cases} p_i & y_i = 0 \\ (1-p_i)\frac{p(y_{ij}\mu_i)}{1 - p(y_i = 0; \mu_i)} & y_i > 0 \end{cases} \quad (6)$$

where $i = 1, \ldots, n$, where n denotes the total number of observations and Y_i refers to the response of the ith observation. Additionally, the probability of a subject belonging to the zero component is represented by p_i , $p(y_i; \mu_i)$ is the PMF for the regular count distribution with a vector of parameters μ_i. Besides, the distribution that is evaluated at zero is represented as $p(y_i; \mu_i)$. The regular count distribution governs the positive counts, as the PMF/1-PMF of this regular count distribution evaluated at zero [3]. The Equation for calculating the probability distribution of the hurdle Poisson is as follows.

$$P(Y_i = y_i) = \begin{cases} p_i & y_i = 0 \\ (1-p_i)\frac{e^{-\mu_i}\mu_i^{y_i}/y_i!}{1 - e^{-\mu_i}} & y_i > 0 \end{cases} \quad (7)$$

Hurdle NB Model

The Hurdle NB Model (HNB) model is formulated as follows [3].

$$P(Y_i = y_i) = \begin{cases} p_i & y_i = 0 \\ \dfrac{1-p_i}{1-\left(\frac{r}{\mu_i+r}\right)^r} \dfrac{\Gamma(y_i+r)}{\Gamma(r)y_i!}\left(\dfrac{\mu_i}{\mu_i+r}\right)^{y_i}\left(\dfrac{r}{\mu_i+r}\right)^r & y_i > 0, \end{cases} \quad (8)$$

Covariates can enter the mean function μ_i and the probability of a zero p_i the model. Thus, the model is formulated as follows:

$$\log(\mu_i) = x_i^T \alpha,\ \text{logit}(p_i) = z_i^T \beta \quad (9)$$

The regression coefficients for the covariates $\boldsymbol{x}_i$ and $\boldsymbol{z}_i$ are defined as $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$, respectively.

Overdispersion

Overdispersion refers to a variance that exceeds the expected variance. Some distributions don't have a parameter that matches the observed variance. For example, a normal distribution does this through the parameter σ (i.e., the model's standard deviation), which is constant in typical regression. On the other hand, a Poisson distribution doesn't have such a parameter (i.e., the variance and mean have the same value). However, if the data are overdispersed? (could also be lower, i.e., underdispersed). This is less common, and not all of the methods below allow for underdispersion modeling. Overdispersion arises "naturally" when important predictors are missing or misspecified (e.g., linear rather than nonlinear). It is widely known that zero inflation exists, but it is separate. Additionally, overdispersion occurs when none of your data points are actually zero [29].

Machine Learning Algorithms

Support Vector Regression

The Support Vector Machines (SVMs) can be easily used to generalize to unknown data. These concepts of SVM are generalizable to regression. Just like classification, Support Vector Regression (SVR) has kernels, sparse solution, and margin and support vectors control. Whereas not as popular as SVMs, SVR has shown to be useful in approximating the truth-value function. A symmetric loss term is used to train SVR with the hope of balancing the preference of high and low errors like in a supervised learning method.

A small-radius flexible tube is symmetrically formed around the estimated function using an e-insensitive Vapnik approach. For example, absolute errors below a certain threshold e. Here, neither the estimate below nor the one above is considered. Therefore, points that are not within the tube are penalized, while points that are within the tube, regardless of their position above or below the function, are not penalized. A major advantage of SVR is that its computational complexity is independent of the dimensions of the input space. It also exhibits excellent generalization ability, with high prediction accuracy. In SVR, the support vectors are the most influential instances that shape the tube's geometry, as in SVM. Crucially, in the context of supervised learning, the training and test data are assumed to be independent and identically distributed (iid), generated from the same, but unknown, fixed probability distribution function [30].

SVMs can also perform regression via a different loss function known as ε-insensitive loss function, $\|y - f(x)\|_\varepsilon = max\{0\|\text{y} - \text{f(x)}\| - \varepsilon\}$. Here, the errors that are smaller than a certain threshold $\varepsilon > 0$ are ignored. Accordingly, a tube is created around the true output. Then, the primal is written as follows [31].

$$\text{minimize } t(\mathbf{w}, \xi) = \frac{1}{2}\|w\|^2 + \frac{C}{m}\sum_{i=1}^{m}(\xi_i + \xi_i^*) \quad (10)$$

$$\text{subject to } (\langle \Phi(x_i), \mathbf{w} \rangle + b) - y_i \leq \epsilon - \xi_i \tag{11}$$

$$y_i - (\langle \Phi(x_i), \mathbf{w} \rangle + b) \leq \epsilon - \xi_i^* \tag{12}$$
$$\xi_i^* \geq 0 (i = 1, \ldots, m)$$

The accuracy of SVM regression can be determined by calculating the scale parameter of the Laplace distribution on the residuals $\zeta = y - f(x)$, where f(x) is the estimated decision function (see Figure 1) [32].

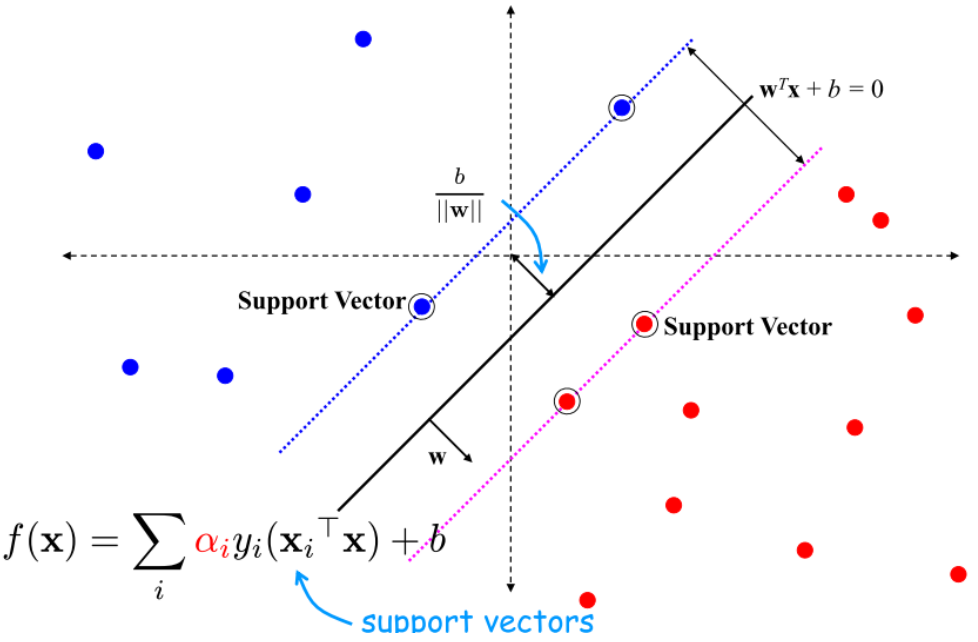


Figure 1: One-dimensional linear SVR [33].

As an optimization problem, the function approximation problem is formulated, attempting to find the narrowest surface-centered tube while reducing the predicted errors (the distance between the desired and predicted outputs). The previous condition leads to Equation (12), which is the objective function. In the Equation, $\|w\|$ refers to the quantity of the vector normal to the surface to be approximated: [30].

$$min_w = \frac{1}{2}\|w\|^2 \tag{13}$$

Based on Equations (12) and (13), the optimization problem is given by Equation (10). Here, C is a regularization—and therefore, a tunable parameter that gives more weight to minimizing the flatness, or error, of this multi-objective optimization problem [30].

$$min \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{N} \xi_i + \xi_i^*$$
$$\text{subject to}$$
$$y_i - w^T x_i \leq \varepsilon + \xi_i^* i = 1 \ldots N \tag{14}$$
$$w^T x_i - y_i \leq \varepsilon + \xi_i i = 1 \ldots N$$
$$\xi_i, \xi_i^* \geq 0 i = 1 \ldots N$$

$$\mathcal{L}(w, \xi^*, \xi, \lambda, \lambda^*, \alpha, \alpha^*) = \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{N} \xi_i + \xi_i^* + \sum_{i=1}^{N} \alpha_i^*(y_i - w^T x_i - \varepsilon - \xi_i^*) \tag{15}$$
$$+ \sum_{i=1}^{N} \alpha_i(-y_i + w^T x_i - \varepsilon - \xi_i) - \sum_{i=1}^{N} \lambda_i \xi_i + \lambda_i^* \xi_i^*$$

Random Forest

Tin Kam Ho (1995) [34] proposed the Random Forests (RFs) algorithm, which was then developed by Breyman (2001) [35]. This algorithm is an accurate and robust learning method. Additionally, it is widely utilized in both regression and classification. RF constructs an ensemble of decision trees using clustering context. The classification or prediction is achieved by each decision tree in the RF. By clustering on the ensemble, the final predictions are made. Recently, RF has been developed further to improve its performance. The random vectors employed to develop each tree significantly influence the growth of these ensembles. Breyman (1996b) [36] implemented clustering, where a random selection (without replacement) is made from the training set to develop each tree. RFs continue to be widely employed due to their numerous benefits. They are easy to implement, provide accuracy as good as, and in some cases even exceed, AdaBoost, and are powerful against noise and outliers. Many implementations of RF demonstrate useful estimates of significance and error [37].

Furthermore, Random Forest is a tree-based ensemble learning method and is classified as a clustering type [38]. It differs from other standard trees in that each node is partitioned using the best predictor from a randomly selected subset of predictors at that node [39]. To improve clustered trees in the RF algorithm, a simple modification to tree decoupling is made. Multiple decision trees for clustering can be built from pre-trained datasets. However, each time a split in a tree is considered, a random sample of m predictors is selected as candidates for the split from the entire set of p predictors when building these decision trees [40]. The RF approach for both classification and regression is presented in Algorithm 1 [41].

- ✓ From the initial data, three bootstrap samples are drawn.
- ✓ For each bootstrap sample, initialise an unpruned tree with the following modification. Instead of selecting the optimal split among all predictors at each node, randomly sample ntry of the predictors and select the optimal split from among those features. Bagging can be considered a subset of random forests, which are generated when the number of predictors (ntry) equals k.
- ✓ The m trees' predictions are aggregated to predict new data.

XGBoost Gradient Boosting

Recently, boosting has become one of the most powerful ML algorithms and has rapidly gained popularity among data scientists and actuaries. Freund and Shapiro (1997) [42] introduced the first practical boosting algorithm, called AdaBoost.M1. Although boosting was initially developed for classification problems, it has also been modified to address regression problems (Hastie [46]). Friedman (2001) [43] proposed gradient boosting, a related algorithm specifically designed for regression. This method iteratively trains a weak learner to build a strong learner by combining multiple weak learners. In gradient boosting, at each iteration, a new weak learner is trained to predict the error of the existing strong learner, known as the "pseudo-response". This weak learner is combined with the strong learner to reduce the model's overall error. This process continues iteratively, as successive weak learners are combined to form an increasingly accurate strong learner. Boosting can also be described as a stepwise process [26].

Artificial Neural Networks

An Artificial Neural Network (ANN) is a simulation of the interconnections that are composed of the human brain. The inputs of ANN are the dendrites of the human brain that receive electrochemical impulses on the other neurons and are transmitted to the cell body. Each input has a signal which is received by its weight multiplied with the input of a hidden layer in the neuron. Each input has a signal which is received by multiplying its weight with the input into a hidden layer in the neuron (see Figure 2). This signal is driven

by an activation function which typically is a sigmoid function. Nonetheless, there are other functions of activation (tangent hyperbolic function, linear function, step function, ramp function, and Gaussian function) that can be employed as well [44] [41]. The last layer is the output layer which is the axon that extends to the synapse that links two different neurons. An average artificial neural network design is made of inputs, outputs, and bias. The ANN architectures differ in the respect of number of layers vastly. The most basic and most widely used architecture is the perceptron which has two inputs, a hidden layer, and one output [41].

Several versions of ANNs have appeared in the scientific literature (Hopfield, feedforward ANNs, backpropagation ANNs). The BP-ANN [45] has been identified as one of the most effective ANN algorithms for supervised learning. As shown in Figure 3, the BP-ANN employs three layers (input, hidden, and output); its distinction from other ANN algorithms lies in how the weights are adjusted [46].

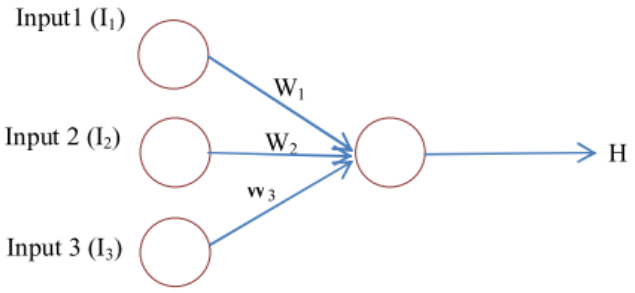


Figure 2: The Single Neuron Model [46].

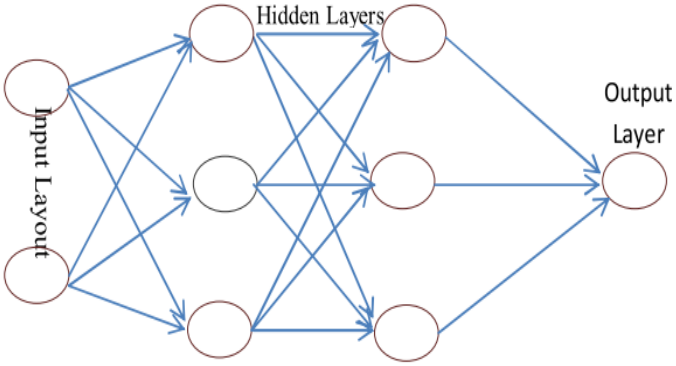


Figure 3: The BP-ANN model [46]

The model consists of two input layers, two hidden layers, and one output layer. The steps of the BP-ANN algorithm are as follows. Generate the output for each pattern, calculate the error (Ej), adjust the weight, and calculate the MSE.

Evaluation Criteria

Models were estimated using a weighted likelihood function, and their fit was assessed using AIC, BIC, and log-likelihood. Vuong's test was also performed to compare non-nested models. AIC and BIC were used for statistical models. Vuong (1989) test: Fung (1989) [18] developed some general tests for non-nested models.

Root Mean Square Error

The root mean squared error (RMSE) is a commonly used metric for evaluating model performance. It is formulated as follows.

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n} (y_i - \hat{y}_i)^2} \qquad (16)$$

It is a count response and ˆyi is the count response forecast of the ith man. The lowest RMSE on the test data is the sign of a good predictive model or machine learning algorithm under analysis [37].

Mean Absolute Error

When there are zero counts in the data [51], it is recommended to use the mean absolute scaled error (MASE). As in the following, the MASE is given by the mean [37].

$$\text{MASE} = \left|\frac{y_i - \hat{y}_i}{(1/n)\sum_{i=1}^{n}|y_i - \bar{y}|}\right| \tag{17}$$

Coefficient of Determination

The R-squared (R^2) value is a measure of model fit, commonly used to evaluate model performance, and is formulated as follows.

$$R^2 = \frac{\sum_{i=1}^{n}(\hat{}yi - \bar{}y)^2}{\sum_{i=1}^{n}(yi - \bar{}y)^2} \tag{18}$$

$$R^2 = 1 - \frac{\sum\ (y_i - \hat{y})^2}{\sum\ (y_i - \bar{y})^2} \tag{19}$$

Where, $\hat{y}$ – predicted value of y $\bar{y}$ – mean value of y. The R2 value lies between 0 and 1, where one indicates that the model fits the data perfectly [37].

Model Validation

Vuong's test

Vuong's test is a statistical test used to compare non-nested models [47]. It is calculated as follows.

$$V = \left(\sqrt{n}\left(\frac{1}{n}\sum_i^n\ m_i\right)\right) / \left(\sqrt{\frac{1}{n}\sum_i^n\ (m_i - \bar{m})^2}\right)$$

where $m_i = \log\big(P_1(Y_i \mid X_i)\big) - \log\big(P_2(Y_i \mid X_i)\big)$. If V > 1.96, then the first model is preferred; if V<-1.96, then the second one is preferred; if V<1.96, then neither model is preferred [19].

Akaike Information Criterion (AIC)

One of the most common metrics in selecting the optimal model is the Akaike Information Criterion (AIC) [37]. To select the model, evaluate the precision of the estimates to determine its accuracy. AIC was developed by the Japanese statistician Hirotugu Akaike in 1970. It is formulated as $AIC = 2k - 2\log(\ell(y;\hat{\mu}))$, where k is the number of parameters included in the model and ℓ is the likelihood of the fitted model defined as above [48]. The basic idea of AIC is to penalize the inclusion of additional variables in a model. It adds a penalty that increases the error when additional terms are included. The lower the AIC, the better the model [48]

Bayesian Information Criterion (BIC)

Another popular measure is the BIC. It is given by

$$\text{BIC} = -2\log L + p\log n \tag{20}$$

and L log is the log-likelihood of the model, p is the model size (number of model parameters), and n is the number of observations. It is also not dependent on the previous cost and punishes the model due to its complexity. BIC has a larger penalty term as compared to AIC [37].

Performance Evaluation

Results

Firstly, data from the R Survival package were used. Epileptic Seizure Study Data is a clinical dataset containing the number of seizures per patient, along with information on treatment, age, and medical visits. Data were obtained from the open-source Survival package in the R programming environment, available at: https://rdrr.io/cran/survival/man/epil.html (epil data), which includes:

- ✓ y Count: The number of seizures experienced by the patient in a specific period (dependent variable). Patient information, including age, the number of seizures in the baseline period, the type of treatment, and the number of visits.
- ✓ id: Patient identification number (used in mixed models).
- ✓ trt: Treatment type (0 = placebo, 1 = active drug)
- ✓ base: Number of seizures in the baseline period before the study.
- ✓ age: Patient age.
- ✓ visit: Number of visits (1 to 4).
- ✓ period: Time period of visit (before/during treatment).

The data was divided into two sets: a Training set, comprising 80% of the data for model building, and a Test set, comprising 20% of the data for evaluating overall performance.

Here, the results are presented and then critically discussed in the next sub-section.

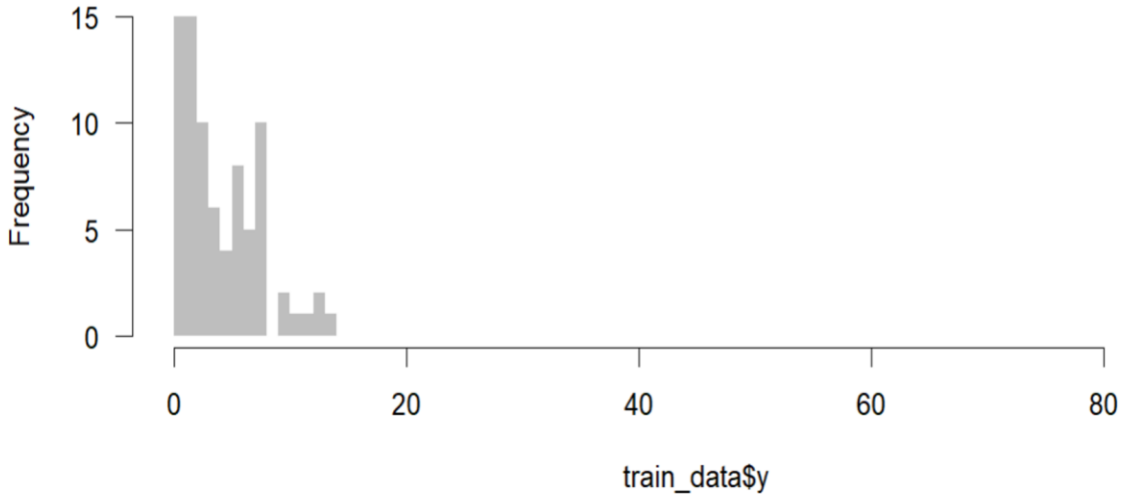


Figure 4: Histogram of a variable "y Count" in the data (Epileptic).

Table 1 represents the mean, variance, and dispersion values of the dependent variable in the epilepsy data. The value is significantly greater than 1, indicating that the data exhibit overdispersion.

Table 1: Mean, variance, and dispersion ratio of the Epilepsy dependent variable.

var_y	mean_y	dispersion_ratio
11.49367	11.49367	2.554149

Table 2 presents the dispersion values for different count models applied to seizure data, indicating the model's suitability in dealing with overdispersion and zero inflation.

Table 2. Dispersion values of count models applied to Seizure Data.

Model	Dispersion	Notes/Comments
ZIP	2.314030	Indicates strong overdispersion.
ZINB	1.072013	Very close to 1 ⇒ Very good model
HurdlePoisson	2.253483	Like ZIP, there is strong overdispersion. The model alone is insufficient to account for the variance in the data.
HurdleNB	1.042968	Excellent, very close to 1 ⇒ Robust model, efficiently handles sparse and zero-sum data.

Table 3 compares the performance metrics of statistical and ML models for predicting the number of epileptic seizures.

Table 3: Performance metrics of statistical and ML models for seizure prediction.

Model	AIC	BIC	R^2	RMSE	MAE
ZIP	463.4625	475.5	0.018826	3.888	3.039
ZINB	422.1555	436.6	0.003433	3.781	2.992
Hurdle Poisson	467.9586	487.2	0.022080	3.805	2.992
Hurdle NB	425.6331	447.3	0.008506	3.74	2.947
SVR			0.692898	3.298483	0.863731
Random Forest			0.2260	3.8201	2.7636

			339	33	29
XGBoost Tuned			0.2023249	4.761332	3.6182210
neural network			0.4288998	3.281509	2.4196944

Table 4 presents a brief evaluation of statistical counting models based on their overall predictive fit.

Table 4: Evaluation of Statistical Counting Models Based on Predictive Fit.

Model	Evaluation
ZIP	Not very suitable,
ZINB	Better in terms of AIC/BIC, but R^2 is very poor.
Hurdle Poisson	Performance similar to ZIP
Hurdle NB	Slightly better than ZIP/Hurdle Poisson in terms of AIC.

Figure 5 shows the architecture of a neural network used to predict the number of seizures in the epilepsy (epil) dataset. The model takes three inputs (base, age, and trt) and aims to perform an accurate regression analysis to predict values of a single dependent variable.

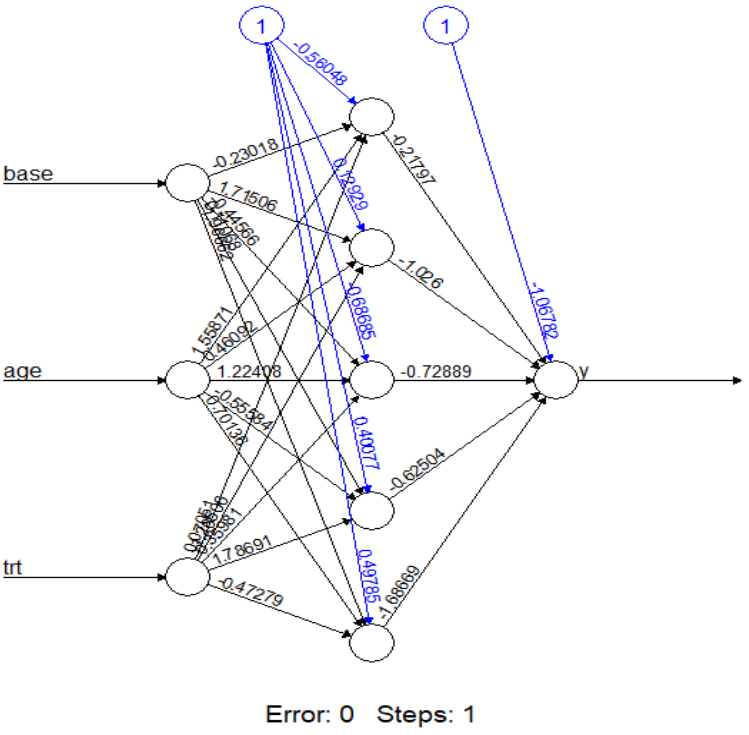


Figure 5: Neural network architecture and weights for predicting seizure count (epil data).

Figure 6 compares four statistical models based on the AIC and the BIC. The graph shows that the ZINB model performs best, achieving the lowest AIC value and is the most suitable choice for analyzing data with an inflated zero distribution (best-fit values are highlighted in green).

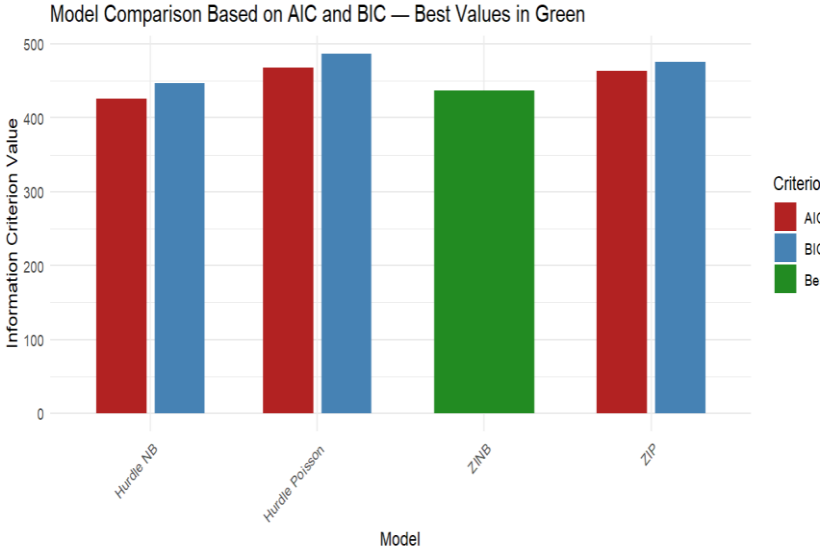


Figure 6: Comparison of model fit using AIC and BIC.

The performance of several models, using RMSE and MAE metrics, is compared in Figure 7. The green bars indicate the best-performing models: the RMSE-best-performing model is SVR, with a value of approximately 3.3. The best-performing model for MAE is SVR, achieving an MAE of approximately 0.9. Overall, the Neural Network model

demonstrates excellent performance in MAE, while the SVR model clearly outperforms in RMSE.

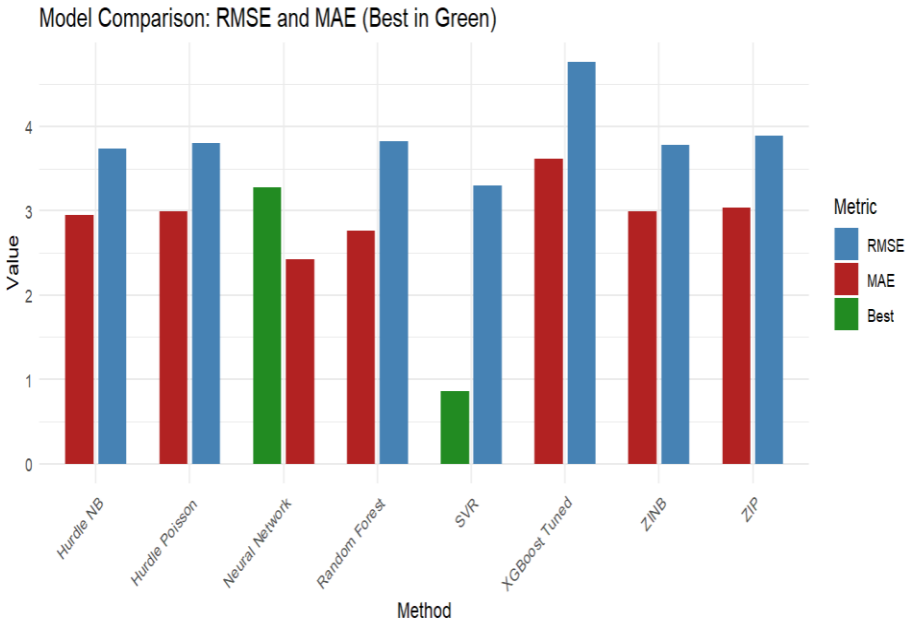


Figure 7: Performance of statistical and ML models based on RMSE and MAE.

Figure 8 compares the R^2 coefficients for statistical indicators and machine learning algorithms. The SVR model emerged as the best (the best method is shown in green), achieving the highest R^2 values (>0.65). This demonstrates its superior ability to explain data variance compared to other models.

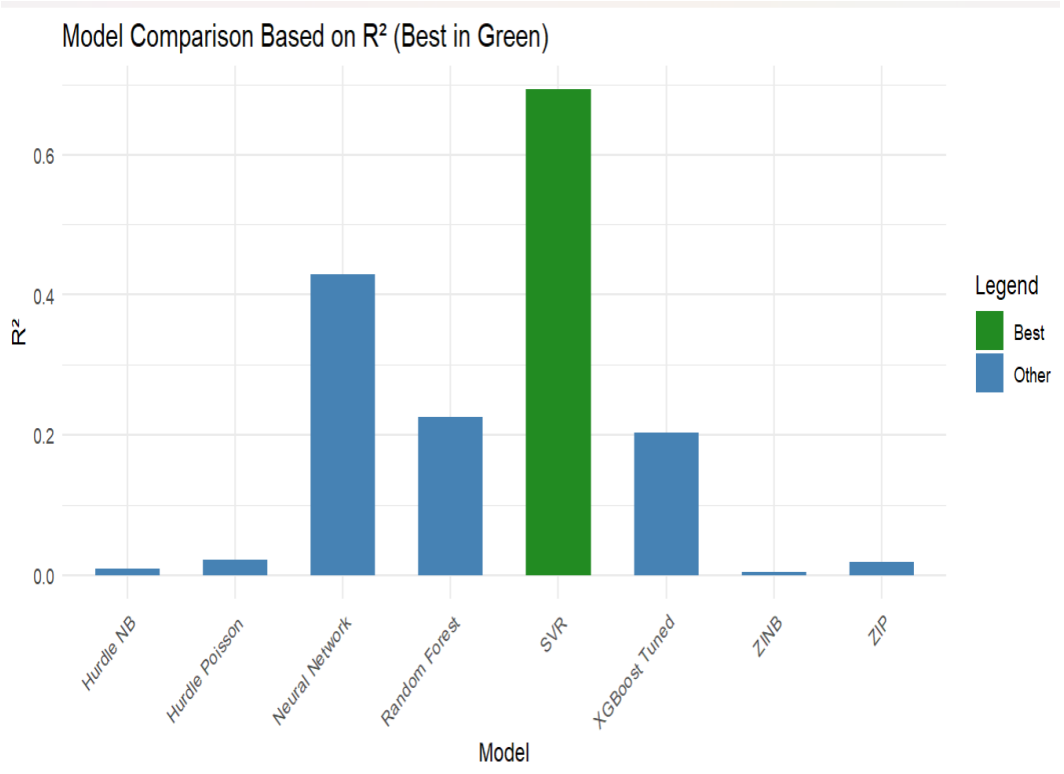


Figure 8: Compares the coefficients R^2 across statistical indicators and learning models.

Table 5 lists the results of Fung's test comparing the ZINB and ZIP models. It shows that the ZINB model is significantly superior to the ZIP model.

Table 5: Results of Fung's test comparing the ZINB and ZIP models.

Vuong z-statistic	H_a	p-value
-3.16427	Mode2_zinb > model_zip	0.00077736

Table 6 compares the Hurdle Poisson and ZIP models using the Vuong test. It shows that the difference is not significant. Furthermore, there is insufficient statistical evidence to favor Model 2_hurdle over Model 1_zip.

Table 6: Comparison between the Hurdle Poisson model and the ZIP model using the Vuong test.

Vuong z-statistic	H_a	p-value
0.1856745	Model2_hurdle_pois> model1_zip	0.4235

Table 7 presents the results of the Fung test comparing the ZINB and Hurdle negative binomial models. This means that the difference between the two models is not significant. That is, there is insufficient evidence to confirm that ZINB is superior to Hurdle NB.

Table 7: Results of the Fung test comparing the ZINB and Hurdle negative binomial models.

Vuong z-statistic	H_a	p-value
0.9726784	model1_zinb>model2_ hurdle_nb	0.16536

Table 8 lists the best-performing models based on two distinct error metrics for a predictive task. The Hurdle NB model achieved the lowest RMSE of 3.74, while the SVM model achieved the lowest MAE of 2.84.

Table 8: Best performing models based on RMSE and MAE criteria.

The best model	criterion	value
Hurdle NB	RMSE	3.74

SVM	MAE	2.84 274 7

Discussion

According to the findings above, ZINB model was effective when it comes to analyzing data with zero inflation and excessive dispersion to balance simplicity and structural appropriateness. Conversely, the SVR model proved to be the most accurate in the pattern interpreting and relationship identification in the complex data, and thus the best model to be used in the analysis of the unconventional counting data. The findings demonstrated that the SVR is the Best model in terms of accuracy and explanation (the highest R^2 value). MAE = 0.863731, RMSE = 3.30, R^2 = 0.69. The RF algorithm achieved: RMSE = 3.82, MAE = 2.76, and R^2 = 0.226. Average performance, explaining about 22% of the variance. The XGBoost-tuned model determined the following metrics: RMSE = 4.76, MAE = 3.62, and R^2 = 0.202. Accuracy is lower than expected, although improved. For ANNs, the metrics were as follows: RMSE = 3.28, MAE = 2.42, R^2 = 0.429. Very good performance, approaching SVR in RMSE.

Balance between performance and accuracy. RF and XGBoost models are less accurate than expected. From the RMSE and MAE Graphs, we found that each model is displayed in two columns, one for RMSE and one for MAE. The best values are shaded in green. This indicates that the neural network and SVR are the most effective based on the error. The amount of variance explained is on the R^2 plot. The best performer (maximizing R^2) is SVR displayed in green. This aids in the graphical comparison of models. Illustratively, the AIC and BIC Graphs are presented. ZINB had the best results using both measures (in green). This helps choose the most suitable and easiest model. Neural Network Topology Graph: Displays the neural network layers that the network was trained on and the relationships that exist between variables. Weights denote the contribution of every input to every internal node and the final output.

Conclusion

This paper contrasts machine learning algorithms and the traditional statistical models in predicting epileptic seizures. Conventional models (including ZIP, ZINB, Hurdle Poisson and Hurdle NB) had low explanatory powers ($R^2 < 0.03$). Nevertheless, ZINB and Hurdle NB showed improved statistical consistency and were able to deal with excess dispersion and zero inflation. However, it is clear that ML algorithms were superior to traditional models. The SVR model achieved the best predictive performance (R^2 = 0.69 and the lowest MAE), second in the line is the neural network-based model, with a combination of performance and accuracy as well as a high-generalization factor. The Random Forest and XGBoost models (adjusted) in their turn performed averagely and failed to reach the same level of accuracy as the other models.

According to these findings, the best approaches in predicting the frequency of epileptic seizures are the SVR model and neural networks as they are the most reliable. The advanced machine learning methods will be the future research warranting enhancement of the precision and the range of explanatory variability because they encompass LightGBM, CatBoost, and Bayesian neural networks. Other alternatives that can be considered as to be seen in terms of statistical models include ZIGP, GAMLSS, COM-Poisson and Finite Mixture that are effective in managing excessive dispersion and zero inflation and can also offer more flexible solutions to complex data distributions.

References

[1] S. Muhammad Usman, S. Khalid, and M. H. Aslam, "Epileptic Seizures Prediction Using Deep Learning Techniques," *IEEE Access*, vol. 8, pp. 39998–40007, 2020, doi:

10.1109/ACCESS.2020.2976866.

[2] M. Beveridge, Z. Goldstein, and H. Cheol Chung, *A Comparison of Zero-Inflated Models for Modern Biomedical Data*, no. 22. 2025. doi: 10.33697/ajur.2025.141.

[3] C. X. Feng, "A comparison of zero-inflated and hurdle models for modeling zero-inflated count data," *J. Stat. Distrib. Appl.*, vol. 8, no. 1, 2021, doi: 10.1186/s40488-021-00121-4.

[4] P.C. Consul, and G.C Jain, "On some interesting properties of the generalized Poisson distribution," *Biometrical J.*, vol. 15, pp. 495–500.

[5] K. M. S. vel and C. S. R. Tha, "Estimation of Zero-Inflation Parameter in EstimZero-Inflated Poisson Model," *Int. J. Math. Trends Technol.*, vol. 56, no. 2, pp. 135–140, 2018, doi: 10.14445/22315373/ijmtt-v56p519.

[6] I. Perumean-Chaney, S. E., Morgan, C., McDowall, D., Aban, "Zero-inflated and overdispersed: what's one to do?," *J. Stat. Comput. Simul.*, vol. 83, no. 9, pp. 1671–1683, 2013.

[7] D. Lambert, "Zero-inflated poisson regression, with an application to defects in manufacturing," *Technometrics*, vol. 34, no. 1, pp. 1–14, 1992, doi: 10.2307/1269547.

[8] M. Beveridge, Z. Goldstein, and H. C. Chung, "A Comparison of Zero-Inflated Models for Modern Biomedical Data," pp. 1–28, 2024.

[9] Harlan Campbell., "The consequences of checking for zero-inflation and overdispersion in the analysis of count data.," *Methods Ecol. Evol. 2021. doi https//doi.org/10.1111/ 2041-210X.13559. URL https//besjournals.onlinelibrary.wiley.com/doi/abs/10.1111/ 2041-210X.13559. Hee*, vol. 12, no. 4, pp. 665–680, 2012.

[10] J. Mullahy, "Specification and testing of some modified count data models.," *J. Econ*, vol. 33, pp. 341–365, 1986.

[11] K. C. H. Yip and K. K. W. Yau, "On modeling claim frequency data in general insurance with extra zeros," *Insur. Math. Econ.*, vol. 36, no. 2, pp. 153–163, 2005, doi: 10.1016/j.insmatheco.2004.11.002.

[12] D. Erdman, L. Jackson, and A. Sinko, "SAS Global Forum 2008 SAS Presents Zero-Inflated Poisson and Zero-Inflated Negative Binomial Models Using the COUNTREG Procedure SAS Global Forum 2008 SAS Presents," *Distribution*, vol. 9, no. 1, pp. 1–11, 2008.

[13] S. M. Mwalili, E. Lesaffre, and D. Declerck, "The zero-inflated negative binomial regression model with correction for misclassification: An example in caries research," *Stat. Methods Med. Res.*, vol. 17, no. 2, pp. 123–139, 2008, doi: 10.1177/0962280206071840.

[14] P. Mirowski, D. Madhavan, Y. LeCun, and R. Kuzniecky, "Classification of patterns of EEG synchronization for seizure prediction," *Clin. Neurophysiol.*, vol. 120, no. 11, pp. 1927–1940, 2009, doi: 10.1016/j.clinph.2009.09.002.

[15] Seyed Ehsan Saffari and Robiah Adnan, "Zero-Inflated Negative Binomial Regression Model with Right Censoring Count Data," *J. Mater. Sci. Eng. B*, vol. 1, no. 9, 2011, doi: 10.17265/2161-6221/2011.09.020.

[16] M. C. Hu, M. Pavlicova, and E. V. Nunes, "Zero-inflated and hurdle models of count data with extra zeros: Examples from an HIV-risk reduction intervention trial," *Am. J. Drug Alcohol Abuse*, vol. 37, no. 5, pp. 367–375, 2011, doi: 10.3109/00952990.2011.597280.

[17] D. Kokotsaki, V. Menzies, and A. Wiggins, "Durham Research Online Woodlands," *Crit. Stud. Secur.*, vol. 2, no. 2, pp. 210–222, 2014, doi: 10.1002/bimj.200100000.

[18] S. Yang, G. Puggioni, L. L. Harlow, and C. A. Redding, "A comparison of different methods of zero - inflated data analysis and an application in health surveys," *J. Mod. Appl. Stat. Methods*, vol. 16, no. 1, pp. 518–543, 2017, doi: 10.22237/jmasm/1493598600.

[19] L. Habeeb Hashim and A. Naeem Flaih, "Modeling the Rainfall Count data Using Some Zero Type models with application," *J. Al-Qadisiyah Comput. Sci. Math.*, vol. 11, no. 2, pp. 14–27, 2019, doi: 10.29304/jqcm.2019.11.2.554.

[20] L. Sharma, A. Narayan Shukla, and S. Maji, "Epileptic Seizure Detection Using Machine Learning: A Review," *Proc. 2022 11th Int. Conf. Syst. Model. Adv. Res. Trends, SMART 2022*, pp. 185–191, 2022, doi: 10.1109/SMART55829.2022.10047389.

[21] J. Diema Enoch, B. O. Omijeh, and R. O. Okeke, "Comparative Evaluation of Zero-Inflated and Hurdle Models for Balanced and Unbalanced Data: Performance Assessment and Model Fit Analysis," *Eur. J. Sci. Innov. Technol. www.ejsit-journal.com*, vol. 3, no. 6, pp. 192–199, 2023.

[22] B. So, "Enhanced gradient boosting for zero-inflated insurance claims and comparative analysis of CatBoost, XGBoost, and LightGBM," *Scand. Actuar. J.*, vol. 2024, no. 10, pp. 1013–1035, 2024, doi: 10.1080/03461238.2024.2365390.

[23] B. So and E. A. Valdez, "Zero-Inflated Tweedie Boosted Trees with CatBoost for Insurance Loss Analytics," 2024, doi: 10.1016/j.asoc.2025.113226.

[24] M. B. Ben Khedher and D. Yun, "An Interpretable Machine Learning-Based Hurdle Model for Zero-Inflated Road Crash Frequency Data Analysis: Real-World Assessment and Validation," *Appl. Sci.*, vol. 14, no. 23, 2024, doi: 10.3390/app142310790.

[25] J. M. Rožanec *et al.*, "Dealing with zero-inflated data: Achieving state-of-the-art with a two-fold machine learning approach," *Eng. Appl. Artif. Intell.*, vol. 149, no. January, p. 110339, 2025, doi: 10.1016/j.engappai.2025.110339.

[26] B. So and M. Deng, "Advancing Zero-Inflated Tweedie Models and Evaluating Gradient Boosting Libraries for Auto Claims," *North Am. Actuar. J.*, vol. 0, no. 0, pp. 1–18, 2025, doi: 10.1080/10920277.2025.2454460.

[27] K. H. Lee, C. Pedroza, E. B. C. Avritscher, R. A. Mosquera, and J. E. Tyson, "Evaluation of negative binomial and zero-inflated negative binomial models for the analysis of zero-inflated count data: application to the telemedicine for children with medical complexity trial," *Trials*, vol. 24, no. 1, pp. 1–11, 2023, doi: 10.1186/s13063-023-07648-8.

[28] D. C. Heilbron, "Zero-altered and other regression models for count data with added zeros.," *Biom. J.*, vol. 36, pp. 531–547, 1994.

[29] C. F. Dormann, "Overdispersion , and how to deal with it in R and JAGS," *Cran*, pp. 1–16, 2016.

[30] M. Awad and R. Khanna, "Efficient learning machines: Theories, concepts, and applications for engineers and system designers," *Effic. Learn. Mach. Theor. Concepts, Appl. Eng. Syst. Des.*, no. January, pp. 1–248, 2015, doi: 10.1007/978-1-4302-5990-9.

[31] A. Karatzoglou, D. Meyer, and K. Hornik, "Support Vector Algorithm in R," *J. Stat. Softw.*, vol. 15, no. 9, pp. 1–28, 2006.

[32] W. R. Lin CJ, ""," Probabilistic Predictions for Support Vector Regression." URL http://www.csie.ntu.edu.tw/~cjlin/papers/svrprob.pdf.

[33] A. Zisserman, "Lecture 3: SVM dual, kernels and regression," *Http://Www.Robots.Ox.Ac.Uk/~Az/Lectures/Ml/*, 2015.

[34] Tin Kam Ho, "Random decision forests," *Pro- ceedings 3rd Int. Conf. Docu- ment Anal. Recognit.*, vol. 1, no. 1, pp. 278– 282, 1995.

[35] L. Breiman, "Random Forests.," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001.

[36] L. Breiman, *Out-of-bag estimation.* 1996.

[37] M. Arora, Y. Kalyani, and S. Shanker, "A comparative study on inflated and dispersed count data," *Proc. 10th Int. Conf. Data Sci. Technol. Appl. DATA 2021*, no. Data, pp. 29–38, 2021, doi: 10.5220/0010547700290038.

[38] R. Punnoose and P. Ajit, "Prediction of Employee Turnover in Orga- nizations using Machine Learning Algorithms.," *Int. J. Adv. Adv. Res. Artif. Intell. (IJARAI), Vol. 5, No. 9, 1-5, 2016.*, vol. 5, no. 9, pp. 1–5, 2016.

[39] A. Liaw and M. Wiener, "Classification and Regression by Random forest," *R news,* vol. 2, no. 3, pp. 18–22, 2002.

[40] R. James, G., Witten, D., Hastie, T., Tibshirani, *An Introduction to Statistical Learning - with Applications in R | Gareth James | Springer*. 2013.

[41] A. A. Ibrahim, R. L. Ridwan, M. M. Muhammed, R. O. Abdulaziz, and G. A. Saheed, "Comparison of the CatBoost Classifier with other Machine Learning Methods," *Int. J. Adv. Comput. Sci. Appl.*, vol. 11, no. 11, pp. 738–748, 2020, doi: 10.14569/IJACSA.2020.0111190.

[42] R. E. Freund, Y. and Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting.," *J. Comput. Syst. Sci.*, vol. 55, no. 1, pp. 119–139, 1997.

[43] J. H. Friedman, *Greedy function approximation: a gradient boosting machine. Annals of statistics, pages*. 2001.

[44] Shiruru, Kuldeep, "An Introduction to Artificial Neural Network.," *Inter- Natl. J. Adv. Res. Innov. Ideas Educ.*, vol. 1, pp. 27–30, 2016.

[45] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, no. 6088, pp. 533–536, 1986, doi: 10.1038/323533a0.

[46] C. O. Anyaeche and D. E. Ighravwe, "Predicting performance measures using linear regression and neural network: A comparison," *African J. Eng. Res.*, vol. 1, no. July, pp. 84–89, 2013.

[47] W. N. et al Zuur AF, Ieno EN, *Mixed effects models and extensions in ecology with R. Springer, Berlin*. 2009.

[48] J. Lanford, T. Nykodym, A. Rao, and A. Wang, "Generalized Linear Modeling with H2O's R Package," no. February, 2021.