

**Estimation of Missing Data in
Ecological Studies Using
Machine Learning Techniques**

B Sidumo

 **orcid.org/0000-0002-4267-9651**

Thesis submitted in fulfillment of the
requirements for the degree *Doctor of Philosophy*
in Operational Research at the North-West
University

Promoter: Dr E Sonono
Co-promoter: Dr I Takaidza

Examination: November 2022
Student number: 31494498

Declaration

By submitting this thesis, I declare that the entire work contains no material that have been accepted for any other degree or diploma in any university and to the best of my knowledge and belief, it contains no material previously published or written by another person, except where due reference has been made in the text. All sources of help have been acknowledged.

Bonelwa Sidumo

16 November 2022

Date

Copyright © 2022 North-West University
All rights reserved.

Dedication

I dedicate this thesis to my late father Mxolisi Sidumo, as it was his wish to see me one day wearing ibhatyi ebomvu (a red gown).

Acknowledgements

First and foremost, I would like to give gratitude to the Almighty God for providing me with the opportunity and the determination to carry out this study at North-West University. My special thanks goes to my supervisor, Dr Energy Sonono who has patiently guided me and my co-supervisor Dr Isaac Takaidza. Without my supervisors' guidance and help, this thesis would not have been possible.

I would also like to thank Laban Musinguzi and other research team members of the National Fisheries Resources Research Institute of Uganda for the kind permission to use their data. A special thanks to my sister Namhla Mkiva, words cannot express how grateful I am for your emotional support throughout this experience. I would like to thank the North-West University nGAP manager Prof Susan Visser for her endless support throughout my PhD studies. I am indebted to the School of Mathematical and Statistical Sciences and colleagues for providing me different support.

Finally, I would like to thank my family for their support throughout this journey. I could not accomplish so much without their support. Words cannot express how grateful I am to my mother Nophuthumile Sidumo, for all of the sacrifices she has made on my behalf. Her prayers for me were what sustained me thus far. To my beloved son Oyisa Sidumo, thank you for being such a good boy and for understanding.

Authorship

All the work presented in this thesis is my own. I was involved in all aspects of this thesis which include implementation of all algorithms, data analysis and manuscript writing (lead author), though I benefited from the comments of my supervisors.

Part of the thesis have been presented at the following conferences:

- Sidumo B., Sonono E., and Takaidza I. 2021. “Count Regression and Machine Learning Techniques for Zero-Inflated Overdispersed Count Data: Application to Ecological Data”. Paper presented at the International Statistical Ecology Conference (ISEC 2022) Cape Town, 27 June - 1 July 2022, University of Cape Town, South Africa.
- Sidumo B., Sonono E., and Takaidza I. 2022. “An approach to multi-class imbalanced problem in ecology using machine learning”. Paper presented at South African Statistical Association, 63rd Annual Conference (SASA 2022), 28 November - 3 December 2022, George, South Africa.

Parts of the thesis have been submitted for publication as journal articles:

- Sidumo B., Sonono E., and Takaidza I. 2021. “Count Regression and Machine Learning Techniques for Zero-Inflated Overdispersed Count Data: Application to Ecological Data”, *Annals of Data Science*, p. 1-15. (Published)
- Sidumo B., Sonono E., and Takaidza I. 2022. “An approach to multi-class imbalanced problem in ecology using machine learning”, *Ecological Informatics* **71**, p. 101822. (Published)
- Sidumo B., 2022. “A review of machine learning techniques with their applications in ecological count dataset”. (Under review)

Executive Summary

The focus of this study is on the estimation of zero-inflated overdispersed count data that is rampant in ecological studies. This is a major challenge in population studies. The novelty of this study is to demonstrate that ignoring overdispersion leads to incorrect parameter estimates, which may result in researchers identifying predictors as having biological importance when in fact they do not. Hence, the assessment of overdispersion is very important for count data. This study uses machine learning (ML) techniques to try and reduce overdispersion in ecological studies. This thesis consists of three articles presented in separate chapters addressing the main objectives of the study.

Chapter 2 is a review which aims to explore machine learning techniques to predict multiple species. The purpose of this review is to get insight of how machine learning techniques can be used to predict the relationship between species abundance and habitat suitability. Much work in the literature mostly focused on species distribution models (SDM) for predicting the relationship of single species and their environmental relevant covariates with very few studies on the prediction of multiple species. To illustrate the weaknesses and limitations of SDM, the predictive performance of the proposed machine learning techniques are conducted on multiple species dataset. The findings of this study demonstrated that ML techniques provide accurate predictions for predicting the relationship between multiple species abundance and habitat suitability.

Chapter 3 presents count regression and machine learning techniques for zero-inflated overdispersed count data application to ecological data. The problem of overdispersion that is rampant in ecological count data is investigated and addressed using machine learning (ML) regression techniques. In this chapter, the performance of statistical count regression models is compared with the proposed machine learning regression techniques. The experiment is conducted on a single fish species, *Lates*

niloticus. The mean absolute error (MAE) was used to compare the performance of count regression models and ML regression models. The results suggest that ML regression models outperform the statistical count regression models. The findings suggest that ML regression techniques can be used to model ecological count dataset.

Chapter 4 presents an approach to multi-class imbalanced problem in ecology using machine learning. In this chapter, we use machine learning approaches to classify multiple species in population ecology. The goal of this chapter is to ascertain the performance of machine learning classifiers on multiple species. When classifying multiple species, we also note a rise of multi-class imbalanced classification problem. To overcome the multi-class imbalanced problem, the bagging and boosting classifiers in combination with resampling techniques are used and their performances are compared. The recall and F1-score performance metrics were used to select the best classifier for the dataset. The bagging classifiers outperform the boosting classifiers. Findings of this work suggest that the bagging classifiers can be used to classify multiple species in ecological studies.

Keywords: Count data; ecology; classifiers; imbalanced dataset; machine learning; multi-class classification; multiple species; overdispersion; single species; species abundance; species distribution models; zero-inflation

Contents

Declaration	i
Authorship	iv
Executive Summary	v
List of Tables	xii
List of Figures	xiii
1 Chapter 1	1
1.1 Introduction	1
1.2 Modeling of count data	1
1.3 Dispersion for count data	10
1.4 Overdispersion	13
1.5 Approaches for dealing with overdispersion	14
1.6 Model selection criteria	19
1.7 Problem statement	20
1.8 Research questions	21

1.9	Aim and Objectives of the Study	21
1.10	Justification of the study	22
1.11	Ethical considerations	23
1.12	Outline of the thesis	23
2	A review of machine learning techniques with their applications in ecological count dataset	26
2.1	Introduction	27
2.2	Ecological processes	30
2.3	Generalized linear models	31
2.4	Ensemble machine learning algorithms	31
2.5	Classification	40
2.6	Experimental results	48
2.7	Conclusion	58
3	Count Regression and Machine Learning Techniques for Zero-Inflated Overdispersed Count Data: Application to Ecological Data	61
3.1	Introduction	62
3.2	Modeling approach	64
3.3	Materials and methods	70

3.4	Experimental results and discussions	73
3.5	Conclusion	77
4	An approach to multi-class imbalanced problem in ecology using machine learning	80
4.1	Introduction	81
4.2	Machine learning classification techniques	85
4.3	Materials and methods	90
4.4	Experimental results	95
4.5	Discussion	99
4.6	Conclusion	102
5	Conclusion	105
	Bibliography	107
A	Appendix	123
A.1	Statistical models	123
A.2	Machine learning algorithms	137
A.3	Model evaluation measures	150
B	Appendix	153

List of Abbreviations

AdaBoost	Adaptive Boosting
ADASYN	Adaptive Synthetic
AIC	Akaike Information Criteria
ANN	Artificial Neural Networks
AVA	All-Versus-All
BagCART	Bagging Classification Trees
BootCART	Boosting Classification Trees
BIC	Bayesian Information Criteria
BRT	Boosted Regression Trees
CART	Classification and Regression Trees
ECOC	Error-Correcting Output-Coding
EM	Expectation-Maximization
FN	False Negative
FP	False Positive
GBM	Gradient Boosting Machines
GLM	Generalized Linear Model
GLMs	Generalized Linear Models
G-mean	Geometric Mean
IR	Imbalanced Ratio
k NN	k —nearest neighbors
LDA	Linear Discriminant Analysis
MAE	Mean Absolute Error
MCC	Matthew's Correlation Coefficient
MDO	Mahalanobis Distance Over-sampling
ML	Machine Learning
MLP	Multi-Layer Perceptron
MLR	Multinomial Logistic Regression

MSE	Mean Squared Error
MLE	Maximum Likelihood Estimation
MME	Method of Moments Estimation
MQLE	Maximum Quasi-Likelihood Estimation
NB	Negative Binomial
NVB	Naive Bayes
OVO	One-Versus-One
OVA	One-Versus-All
QL	Quasi-Likelihood
QAIC	Quasi Akaike Information Criteria
RF	Random Forests
RMSE	Root Mean Squared Error
ROS	Random Over-Sampling
ROSE	Random Over-Sampling Examples
RUS	Random Under-Sampling
SAIAB	South African Institute for Aquatic Biodiversity
SDMs	Species Distribution Models
SMOTE	Synthetic Minority Over-sampling Technique
SVM	Support Vector Machines
TN	True Negative
TP	True Positive
XGB	eXtreme Gradient Boosting
ZIP	Zero-Inflated Poisson
ZINB	Zero-Inflated Negative Binomial

List of Tables

2.1	Confusion matrix for SDM evaluation.	45
2.2	An example of a multi-class confusion matrix.	45
2.3	The multi-class imbalanced dataset used in the experiments.	49
2.4	Species habitat suitability.	50
2.5	Classifiers results on imbalanced dataset.	53
2.6	Classifiers results using ROS technique.	56
3.1	Descriptive statistics for the variables.	71
3.2	Mathematical formulas for performance metrics.	73
3.3	Results for count regression models.	74
3.4	Summary of results on ML regression models.	76
3.5	Model comparison.	77
4.1	Multiple species count data.	91
4.2	Confusion matrix in binary classification.	93
4.3	ML classifier results on imbalanced dataset.	96
4.4	ML classifier results on the balanced dataset.	98

List of Figures

2.1	Graphical representation of imbalanced and balanced classes.	42
2.2	Graphical representation of species habitat.	51
3.1	Graphical representation of count data.	71
4.1	A map showing sampling sites in Uganda.	91
A.1	Support vector machine for non-linear separable data.	143
A.2	A representation of an artificial neural network.	146

1. Chapter 1

1.1 Introduction

In population ecology, the focus is on modeling species abundance and site occurrence of species (Wenger & Freeman 2008). This entails determining the number of species in a particular area at a certain point in time or over a specified period of time. Recordings of the population status are done by visiting the ecological sites multiple times and examining relevant sampling units where the target species are either detected or not detected (Royle et al. 2005). According to Wenger & Freeman (2008), site occupancy (occurrence) of species are useful measures of population status, hence are of considerable interest to ecologists. However, the recorded data usually has missing observations due to lack of observed species at the recording time. Missing observations cause difficulty in the analysis and decision making procedures. This study proposes estimation techniques for addressing the problem of excess zeros (zero-inflation) in population ecological studies.

1.2 Modeling of count data

Hardin et al. (2007) define count data as observations that only take discrete values theoretically ranging from zero up to the maximum value of the response variable being modeled. Count data with frequent number of zeros is often heavily skewed and hardly follow parametric distribution assumptions. Typical examples of count data is the number of a certain species of fish per defined area in the ocean or insect abundance across a range of habitats. O'hara & Kotze (2010) suggest that generalized linear models (GLMs) are better suited in dealing with count data. In this study, the GLMs will be used to model excess number of zeros in ecological count data.

1.2.1 Generalized linear model principle

A generalized linear model (GLM) is an extension of linear models which allows the response variable to follow any member of the exponential family of distributions. The simple linear model (Fox 2015):

$$Y_i = \alpha + \beta X_i + \varepsilon_i,$$

can be transformed to a generalized linear model by replacing the mean, $E(Y_i) = \mu_i$, by the link function:

$$g(\mu_i) = \alpha + \beta X_i = \eta_i,$$

where η_i is a linear combination of the covariates. GLM consists of three components:

- The random component that specifies the conditional distribution of the response variable given a set of explanatory variables, $Y_i|X_i$, where Y is the data vector and X is the design matrix.
- The systematic component η_i represents the linear combination of predictors on which the expected value of the response variable, μ_i depends on:

$$\eta_i = \alpha + \beta_1 X_{i1} + \beta_2 X_{i2} + \cdots + \beta_k X_{ik},$$

where α and $\beta_1, \dots, \beta_k$ are unknown regression coefficients.

- The link function which links the expected values of the response to the linear combination of predictors, A log link function is usually used for counts as it ensures predicted outcomes do not go below zero.

Most of the distributions used in GLMs belong to the exponential family of distributions. The interest of this study is in distributions which belong to the exponential

family, which will be explained in the following sub-sections. For more information on the following sub-sections, one is referred to [Faraway \(2016\)](#).

1.2.2 The exponential family of distributions

A random variable Y_i that takes on the general form is classified as a member of the exponential family:

$$f(y_i, \theta, \phi) = \exp \left[\frac{y_i \theta - b(\theta)}{a(\phi)} + c(y_i, \phi) \right], \quad i = 0, 1, 2, \dots, n, \quad (1.2.1)$$

where ϕ is the scale parameter, θ is the link parameter and $a(\cdot)$, $b(\cdot)$, $c(\cdot)$ are functions which determine the distribution when specified and n is the number of observations.

To determine the mean and variance of exponential family in Equation 1.2.1, first determine the log-likelihood function:

$$\begin{aligned} \ell &= \log f(y|\theta, \phi) \\ &= \log \left[\exp \left(\frac{y\theta - b(\theta)}{a(\phi)} + c(y, \phi) \right) \right] \\ &= \frac{y\theta - b(\theta)}{a(\phi)}. \end{aligned}$$

Then find the derivate of the log-likelihood function:

$$\begin{aligned} \frac{\partial \ell}{\partial \theta} &= \frac{\partial \log f(y|\theta, \phi)}{\partial \theta} \\ &= \frac{y - b'(\theta)}{a(\phi)}. \end{aligned}$$

Treating ℓ as a random variable by placing the particular observation y by a random

variable Y , it enables the expected value of $\frac{\partial \ell}{\partial \theta}$ to be evaluated. That is:

$$E \left[\frac{\partial \ell}{\partial \theta} \right] = \frac{E(Y) - b'(\theta)}{a(\phi)}.$$

Setting $E \left[\frac{\partial \ell}{\partial \theta} \right] = 0$, we get the expectation as:

$$\frac{E(Y) - b'(\theta)}{a(\phi)} = 0 \tag{1.2.2}$$

$$E(Y) = b'(\theta). \tag{1.2.3}$$

Let $Y_1, Y_2, \dots, Y_n$ be independent observations from a probability density function (p.d.f.) $f(y, \theta)$, where θ is the unknown parameter with true value θ_0 . Now:

$$E_0 \left[\frac{\partial \ell}{\partial \theta} | \theta = \theta_0 \right] = 0 \tag{1.2.4}$$

and

$$E_0 \left[\frac{\partial \ell}{\partial \theta} \right] = E_0 \left[\frac{\partial}{\partial \theta} \log f(y, \theta) \right].$$

From the definition, it is known that:

$$\begin{aligned}
 E(y) &= \int y f_y(y) dy \\
 &= \int \left[\frac{1}{f(y, \theta_0)} \frac{\partial f}{\partial \theta}(y, \theta) \right] f(y, \theta_0) dy \\
 &= \int \frac{\partial}{\partial \theta} f(y, \theta) dy \\
 &= \frac{\partial}{\partial \theta} \int f(y, \theta) dy && \text{since the integral of } f(y, \theta) \text{ is equal to 1} \\
 &= \frac{\partial}{\partial \theta} (1) \\
 &= 0.
 \end{aligned}$$

From the definition:

$$Var \left[\frac{\partial \ell}{\partial \theta} \right] = E_0 \left\{ \left[\frac{\partial \ell}{\partial \theta} \Big|_{\theta_0} \right]^2 \right\}. \quad (1.2.5)$$

Hence:

$$I = E_0 \left[\left(\frac{\partial \ell}{\partial \theta} \Big|_{\theta_0} \right)^2 \right] - E_0 \left[\left(\frac{\partial^2 \ell}{\partial \theta^2} \Big|_{\theta_0} \right) \right], \quad (1.2.6)$$

where I is the information about θ contained in the dataset:

$$\begin{aligned}
 I(\theta) &= Var[\mu(\theta)] \\
 &= -E \left(\frac{\partial^2 \ell(\theta)}{\partial \theta^2} \right).
 \end{aligned}$$

All the information centered around $I(\theta)$, θ will be large. Proof of the results in

Equation 1.2.6, for single observations determined in Equation 1.2.4 states that:

$$\int \frac{\partial \log}{\partial \theta} f(y) dy = 0.$$

Differentiating with respect to θ :

$$\int \left[\frac{\partial^2 \log}{\partial \theta^2} f(y) + \frac{\partial \log}{\partial \theta} f(y) \frac{\partial f(y)}{\partial \theta} \right] dy = 0.$$

But:

$$\frac{\partial \log}{\partial \theta} f(y) = \frac{1}{f(y)} \frac{\partial f}{\partial \theta}.$$

So:

$$\int \frac{\partial^2 \log}{\partial \theta^2} [f(y)f(y)] dy = - \int \left(\frac{\partial \log f}{\partial \theta} \right)^2 f(y) dy,$$

which is:

$$E_0 \left[\frac{\partial^2 \ell}{\partial^2 \theta} |_{\theta_0} \right] = E_0 \left[\frac{\partial \ell}{\partial \theta} |_{\theta_0} \right]^2.$$

It has been shown in Equation 1.2.2 that $E(Y) = b'(\theta)$. Now, to determine the variance:

$$\frac{\partial^2 \ell}{\partial \theta^2} = \frac{-b''(\theta)}{a(\phi)},$$

substitute this into Equation 1.2.6 to get:

$$\begin{aligned} E \left[\frac{\partial^2 \ell}{\partial \theta^2} \right] &= -E \left[\left(\frac{\partial \ell}{\partial \theta} \right)^2 \right] \\ \frac{b''(\theta)}{a(\phi)} &= E \left[\frac{(y - b'(\theta))^2}{a(\phi)^2} \right] \\ \text{Var}(Y) &= b''(\theta)a(\phi). \end{aligned}$$

It is convenient to consider $\text{Var}(Y)$ as a function of $\mu = E(Y)$. Hence, the mean and variance are given by:

$$\begin{aligned} E(Y) &= b'(\theta) \\ \text{Var}(Y) &= b''(\theta) a(\phi). \end{aligned}$$

Commonly used distributions in the exponential family include the Normal, Binomial, Poisson, Gamma and Inverse Gamma distributions. Next, some distributions (used in GLMs) that belong to the exponential family are presented.

1.2.3 The normal distribution

The probability density function of a normal distribution is given by:

$$f(y; \mu, \sigma^2) = \frac{1}{\sqrt{\pi\sigma^2}} \exp \left[-\frac{(y - \mu)^2}{2\sigma^2} \right], \quad -\infty < y < \infty,$$

where μ is the parameter of interest and σ^2 is regarded as a nuisance parameter. This belongs to the exponential family distribution with:

$$\theta = \mu, \quad b(\theta) = \frac{\theta^2}{2}, \quad a(\phi) = \sigma^2 \quad \text{and} \quad c(y, \phi) = -\frac{(y^2/\phi + \ln(2\pi\sigma^2))}{2}.$$

Thus:

$$E(Y) = \mu \text{ and } Var(Y) = \sigma^2.$$

The link function of the normal distribution is the identity link given by $g(\mu) = \theta = \mu$.

1.2.4 The binomial distribution

The probability density function of a binomial distribution is given by:

$$f(y; p) = \binom{n}{y} \times p^y (1 - p)^{n-y}, \quad y = 0, 1, 2, \dots, n,$$

where $0 < p < 1$ denote the probability of success or failure (denote present or absent species in ecology) and n denote the total number of observations. The binomial distribution belongs to an exponential family distribution with:

$$\begin{aligned} \theta &= \log \left(\frac{p}{1-p} \right) \\ b(\theta) &= n \log [1 + \exp(\theta)] \\ c(y, \phi) &= \log \binom{n}{y} \text{ and } a(\phi) = 1. \end{aligned}$$

The link function of the binomial distribution is the logit, $\theta = \ln \left(\frac{p}{1-p} \right) = g(p)$.

1.2.5 The Gamma distribution

The probability distribution function of a gamma distribution is given by:

$$\begin{aligned} f(y; \mu, \nu) &= \frac{1}{\Gamma(\nu)} \lambda^\nu y^{\nu-1} \exp(-\lambda y), \quad y > 0, \quad \lambda = \left(\frac{\nu}{\mu}\right) \\ &= \frac{1}{\Gamma(\nu)} \left(\frac{\nu}{\mu}\right)^\nu y^{\nu-1} \exp\left(-\frac{y\nu}{\mu}\right) \\ &= \exp \left\{ \left[-\frac{y}{\mu} - \ln(\mu) \right] \nu + (\nu - 1) \ln(y) + \nu \ln(\nu) - \ln \Gamma(\nu) \right\}. \end{aligned}$$

It follows that the gamma distribution is an exponential family distribution member with:

$$\theta = -\frac{1}{\mu}, \quad b(\theta) = -\ln(\theta), \quad a(\phi) = \frac{1}{\nu} \quad \text{and} \quad c(y, \phi) = (\nu - 1) \ln(y) + \nu \ln \nu - \ln \Gamma(\nu),$$

where μ is a scale parameter and ν is the shape parameter. The mean and the variance are:

$$E(Y) = \mu$$

and

$$Var(Y) = \frac{\mu^2}{\nu}.$$

The link function of a gamma distribution is given by $g(\mu) = \theta = -\frac{1}{\mu}$.

1.2.6 The Poisson distribution

The probability density function is given by:

$$\begin{aligned} f(y; \mu) &= \frac{\mu^y \times \exp^{-\mu}}{y!}, \quad y = 0, 1, 2, \dots \\ &= \exp [y \ln \mu - \mu - \ln y!], \end{aligned}$$

where $\mu > 0$ denotes the mean. The Poisson distribution belongs to an exponential family distribution with:

$$\theta = \ln \mu, \quad b(\theta) = \mu, \quad a(\phi) = 1 \text{ and } E(y) = Var(y) = \mu.$$

For any member of an exponential family we know that $E(Y) = b'(\theta)$ and $Var(Y) = b''(\theta)a(\phi)$. Therefore:

$$E(Y) = e^\theta = \mu$$

and

$$Var(Y) = e^\theta = \mu.$$

We can see that the mean and the variance of a Poisson distribution are equal and the link function of the Poisson distribution is $g(\mu) = \ln(\mu) = \theta$.

1.3 Dispersion for count data

Dispersion is referred to as the spread of a dataset. It is defined by [Zuur et al. \(2009\)](#) as:

$$\text{Variance ratio} = \frac{\text{observed variance}}{\text{theoretical variance}},$$

where theoretical variance is calculated from the distribution and observed variance is calculated from the sample data. From the definition, dispersion of a dataset can therefore be identified with regard to a dispersion coefficient. That is:

$$\text{Dispersion}(\phi) = \frac{\sigma^2}{\mu}.$$

A dispersion parameter is estimated from the dataset and a dispersion value that is greater than one indicates overdispersion. This occurs when the observed variance is greater than the expected variance by the fitted model. The dispersion parameter is estimated using two approaches, either the deviance function or Pearson's χ^2 statistic. The deviance function is given as:

$$D(y, \mu) = 2\phi(l(y; y) - l(\hat{\mu}, y)),$$

where $l(y, y)$ and $l(\hat{\mu}, y)$ are the log-likelihoods for the full model and the reduced model, respectively (Faraway 2016). This quantity can be written as:

$$D(y, \hat{\mu})/\phi,$$

where $D(y, \hat{\mu})$ is called the deviance and $D(y, \hat{\mu})/\phi$ is called the scaled deviance. The deviance function is divided by the degrees of freedom (Zuur et al. 2009).

The Pearson's χ^2 statistic has the following form:

$$\chi^2 = \sum_{i=1}^n \frac{(y_i - \hat{\mu}_i)^2}{V(\hat{\mu}_i)},$$

where $V(\hat{\mu}_i)$ is the variance function $b''(\theta)$, estimated from the data and from the

exponential family of distributions. The dispersion parameter can be estimated using:

$$\hat{\phi} = \frac{\chi^2}{n - p} = \frac{\sum_{i=1}^n (y_i - \hat{\mu}_i)^2 / \hat{\mu}_i}{n - p}.$$

When the dispersion parameter (ϕ) is estimated by the deviance function or Pearson's χ^2 statistic, the statistics are based on the quasi-likelihood functions. The quasi-likelihood function is defined by [Faraway \(2016\)](#) as:

$$Q(y; \mu) = \int_y^\mu \frac{y - t}{\phi V(t)} dt,$$

where the dispersion parameter, ϕ , is set to be free when modeling overdispersion. The χ^2 or F statistic may be used to test overdispersion. The F statistic is given as:

$$F = \frac{D_2 - D_1}{\phi_{(p-q)}},$$

where p and q are the number of parameters for the full and reduced models and D_2 and D_1 are the deviance functions for the full model and reduced models, respectively.

Considering the role of the dispersion parameter using the parameterization in the exponential family of distribution, following the procedure in [Section 1.2.2](#), it can be shown that:

$$E(Y) = \mu = b'(\theta)$$

$$Var(Y) = b''(\theta)a(\phi),$$

where μ does not depend on the dispersion parameter and $\hat{\phi} = \frac{\chi^2}{n-p}$ is a good estimator of the dispersion. This study adopts this estimator as a check for overdispersion.

1.4 Overdispersion

The Poisson model is widely used as the foundation for modeling count data response Y under a GLM framework with the assumption that the expected value of Y equals to the variance of Y , that is $E(Y) = Var(Y) = \mu$. This is known as equidispersion (Harrison 2014, Khoshgoftaar et al. 2001, Wagh & Kamalja 2018). However, this assumption is not always true as count data responses may exhibit missing observations or exhibit overdispersion in real life datasets. Bolker et al. (2009) define overdispersion as the occurrence of more variance in the data than predicted by a statistical model due to missing data. The problem of missing observations causes difficulty in the analysis and decision making procedures. The reason for the existence of missing observations in ecology datasets may be due to either structural errors, (for instance a bird or fish is not present because the habitat is not suitable), observer error (species are present but cannot be detected) and design error (poor experimental design or sampling surveys are thought to be the reason) (Zuur et al. 2009). According to Oliveira et al. (2016) and Wagh & Kamalja (2018), count datasets with excessive zero responses are abundant in many disciplines such as insurance, manufacturing applications, environmental sciences, public health and dental epidemiology, and ecology is not even spared from this.

Overdispersion is caused by excess zeros in the dataset and detected when the frequency of zeros is significantly higher than the one predicted by the Poisson model (Wagh & Kamalja 2018). Overdispersion can also arise when count data is clustered, meaning observations are not independent of one another (Hilbe 2011b), that is, when there is excess zeros in the data (zero-inflation) or when the variance of the response variable is truly greater than the mean (Zuur et al. 2009). The evidence of overdispersion is an indication that the data is inadequate to fit the model. Overdispersion may cause standard errors of the estimates to be underestimated and generally makes the predictors appear significant when in fact they are not.

The fact that equidispersion is rarely found in real data has led to the development of more general count models which do not assume such a relationship (Hilbe 2011b). Zero-inflated distributions have been suggested as appropriate models to the above-mentioned missing observations problems. Zero-inflated distributions provide a way of modeling excess zeros in addition to allowing for overdispersion by using two simultaneous operating data generation processes; one generates only zeros and the other is either a Poisson or negative binomial data generating process (Wagh & Kamalja 2018, Kassahun et al. 2014). Zero-inflated negative binomial (ZINB) model has been proposed to solve the problems because of its flexibility in handling both zero-inflation and overdispersion (Minami et al. 2007, Rose et al. 2006). Even if zero-inflated models have been suggested to deal with missing observations, the issue of overdispersion (high variance) still remains.

1.5 Approaches for dealing with overdispersion

This section reviews methods of reducing overdispersion in a general framework and outlines the details on how these models are applied for modeling ecological count data. In literature, Hilbe (2011b) and Ver Hoef & Boveng (2007) gave an overview on methods for reducing overdispersion in count data. Desjardins (2016) and Cheung (2002) used zero-inflated Poisson to model excess zeros in count data and Ridout et al. (2001) used the ZINB to reduce overdispersion. Note that this study does not concentrate on traditional methods of dealing with missing data such mean imputation and multiple imputation as vast research has been carried out led to the conclusion that traditional methods still lead to high variance (Jerez et al. 2010). Zero-inflation means that the proportion of zeros of a model is greater than the proportion of zeros of the corresponding Poisson model (Zhu 2012), this is a common phenomenon in count data.

1.5.1 Negative binomial model

The probability distribution of the negative binomial is given by:

$$P(Y = y) = \binom{y + r - 1}{y} \times p^r (1 - p)^y, \quad y = 0, 1, 2, \dots$$

The mean and variance are:

$$E(Y) = \mu = \frac{r(1 - p)}{p}$$

and

$$Var(Y) = \sigma^2 = \frac{r(1 - p)}{p^2}.$$

The mean-variance relationship of the negative binomial is $E(Y) = pVar(Y)$. When there is overdispersion, the negative binomial is an alternative to the Poisson for count data (Zuur et al. 2009).

1.5.2 Quasi-likelihood Poisson model

Quasi-likelihood (QL) methods are based on GLM principles but allow parameter estimates to be calculated based only on a specification of mean and variance of the model without regard of those specifications originating from a member of the single parameter exponential family of distributions. Quasi-Poisson GLMs allow for the dispersion ϕ to be a free parameter which is significant in modeling overdispersion (Faraway 2016).

1.5.3 Zero-inflated models

Zero-inflated models are capable of dealing with excess zeros in count data. A zero-inflated model is a mixture distribution that is characterized by the data that has been mixed with two Poisson distributions in the proportions π and $1 - \pi$, respectively. The probability density function of this mixed distribution is given by [Rose et al. \(2006\)](#) as:

$$P(Y_{ij} = y_{ij}) = \pi \left[\frac{\exp(-\mu_1) \times \mu_1^{y_{ij}}}{y_{ij}!} \right] + (1 - \pi) \left[\frac{\exp(-\mu_2) \times \mu_2^{y_{ij}}}{y_{ij}!} \right], \quad y_{ij} = 0, 1, 2, \dots, n,$$

where μ_1 and μ_2 with $\mu_2 > \mu_1$ denote the means of the distributions and y denote the observed count data.

1.5.4 Zero-inflated Poisson model

Zero-inflated Poisson (ZIP) regression is a model for count data with excess zeros that allows for two separated processes, the structural zeros and the Poisson distribution conditional on the excess zeros ([Rose et al. 2006](#)). ZIP regression model has been used by researchers for handling purely zero-inflated count data when the assumption of equidispersion holds ([Elliott & Morrell 2009](#)). The model assumes that with probability π the only possible observation is 0 and with probability $(1 - \pi)$ a Poisson random variable is observed ([Lambert 2012](#)). ZIP distribution for participant i at event j is defined by [Rose et al. \(2006\)](#) as:

$$P(Y_{ij} = y_{ij}) = \begin{cases} \pi_{ij} + (1 - \pi_{ij}) \exp(-\mu_{ij}), & y_{ij} = 0 \\ (1 - \pi_{ij}) \frac{\exp(-\mu_{ij}) \mu_{ij}^{y_{ij}}}{y_{ij}!}, & y_{ij} > 0, \end{cases} \quad (1.5.1)$$

where $0 \leq \pi_{ij} < 1$ denotes the probability of zero occurrence and $\mu_{ij} > 0$ with the mean:

$$E(Y_{ij}) = \mu_{ij}(1 - \pi_{ij})$$

and the variance:

$$Var(Y_{ij}) = \mu_{ij}(1 - \pi_{ij})(1 + \pi_{ij}\mu_{ij}).$$

The probability of being an excess zero (π_{ij}) in Equation 1.5.1 is often modeled using a logistic regression. Zero-inflated models use the logistic model to estimate π_{ij} , hence using:

$$\pi_{ij} = \frac{1}{1 + \exp(-\eta_{ij})},$$

where η_{ij} is related to a set of explanatory variables, $\mathbf{X}$. We can see from the ZIP mean and variance that when π equals to zero the ZIP model converges to the standard Poisson model and exhibits overdispersion when π is greater than zero. The covariates are incorporated using a log link function for μ_{ij} and a logit link function for π_{ij} , that is:

$$\log(\mu_{ij}) = \mathbf{x}'_{ij}\boldsymbol{\beta} \text{ and } \log\left(\frac{\pi_{ij}}{1 - \pi_{ij}}\right) = \mathbf{z}'_{ij}\boldsymbol{\gamma},$$

where $\mathbf{x}_{ij}$ and $\mathbf{z}_{ij}$ are vectors of explanatory variables, $\boldsymbol{\beta}$ and $\boldsymbol{\gamma}$ are vectors of regression parameters (Oliveira et al. 2016).

1.5.5 Zero-inflated negative binomial model

According to [Oliveira et al. \(2016\)](#) when overdispersion is due to both the heterogeneity of data and excess of zeros, the zero-inflated negative binomial (ZINB) regression model is considered as appropriate method to use. ZINB regression model has been used by researchers for handling both zero-inflation and overdispersion in count data. It is used to account for commonly occurring overdispersion detected in data that are initially analyzed under the ZIP model. For the ZINB regression model, the probability is estimated by the mass function ([Rose et al. 2006](#)):

$$P(Y_{ij} = y_{ij}) = \begin{cases} \pi_i + (1 - \pi_i) \left(\frac{1}{1 + \mu\theta} \right)^{1/\theta}, & y_{ij} = 0, \\ (1 - \pi_i) \frac{\Gamma(y + \frac{1}{\theta})}{\Gamma(y+1)\Gamma(\frac{1}{\theta})} \left(\frac{1}{1 + \mu\theta} \right)^{1/\theta} \left(\frac{\mu\theta}{1 + \mu\theta} \right)^y, & y_{ij} > 0, \end{cases}$$

where $\theta > 0$ is a shape parameter which quantifies the amount of overdispersion, π denotes the zero-inflation probability, y denotes the species count and μ the mean count. The mean and variance of the ZINB distributions are ([Oliveira et al. 2016](#)):

$$E(Y_{ij}) = \mu_i(1 - \pi_i) = \mu_i$$

and

$$Var(Y_{ij}) = \mu(1 - \pi_i)(1 + \pi_i\mu_i + \mu\theta).$$

The ZINB may be more appropriate than the ZIP model when the non-zero counts are overdispersed relative to what would be expected under the Poisson regression. ZINB have been used in other research fields such as psychology, medicine and public health ([Rose et al. 2006](#)).

1.6 Model selection criteria

For any given dataset, theoretical information approaches such as Akaike information criteria (*AIC*) and Bayesian information criteria (*BIC*) will be considered to choose between different count models. In general:

$$AIC = -2 \ln L + 2k$$

while

$$BIC = -2 \ln L + k \ln n,$$

where L is the model log-likelihood, k is the number of predictors and n the number of observations in the model. The *AIC* will be used to select the best model that has the minimum *AIC*. If we have two models M_1 and M_2 , where M_1 is a full model and M_2 is a reduced model of M_1 and D_1 and D_2 are their deviances respectively. The `drop1()` command function in R performs a likelihood ratio test with the difference between the deviances approximated by a Chi-Square distribution with $p - q$ degrees of freedom. It is defined as (Zuur et al. 2009):

$$D_2 - D_1 \sim \chi^2_{(p-q)},$$

where p and q are number of parameters for models M_1 and M_2 respectively, with $p > q$. The model selection process in quasi-Poisson does not define the *AIC*. Quasi Akaike information criteria (*QAIC*) is a modified *AIC* for models containing an overdispersion parameter. According to Bolker et al. (2009) *QAIC* is similar to *AIC*, except that the log-likelihood is divided by the estimated overdispersion scale parameter of the full model. The objective of this quasi-likelihood form is to balance over and under fitting of the models when overdispersion is present.

The analysis of deviance approach compares two nested models M_1 and M_2 using

different test statistic (Zuur et al. 2009):

$$\frac{D_2 - D_1}{\phi_{(p-q)}} \sim F_{(p-q, n-p)},$$

where ϕ is the overdispersion parameter.

1.7 Problem statement

From the discussion above, overdispersion is an existing and recurring problem that needs attention when dealing with ecological count data. Ignoring overdispersion may cause errors in parameter estimates. This is a major challenge in population ecological studies. Now, the question is: How can overdispersion be reduced further in ecological count data?

From the outlined methods discussed in Section 1.5, there are still challenges in dealing with overdispersion. In other words, there is still room to improve on the modeling of count data by reducing overdispersion further. Hence this study seeks to propose possible methods to reduce overdispersion. Lately, machine learning methods are cropping up in different areas of science. However, to the best of our knowledge, there is limited empirical evidence showing the use of machine learning techniques in estimating missing observations in population ecological studies. The study proposes machine learning techniques to address outstanding challenge of overdispersion in ecological count data. As such, this study mainly seeks to understand whether or not the modeling techniques for missing data in ecological studies can improve the output of count data comparative to excess zeros and overdispersion with the intent to address shortcomings in the literature of zero-inflated models.

1.8 Research questions

The following are the research questions this study seeks to answer:

- To what extent are machine learning techniques applied/used in population ecology?
- Can overdispersion in ecological count data be reduced further after estimating missing data using the proposed machine learning techniques as compared to traditional count regression models?
- How do the proposed techniques perform on multi-class imbalanced problem on real-life ecological data?

1.9 Aim and Objectives of the Study

Aim of the study

The aim of the study is to propose methods for estimating missing data in ecological studies using machine learning techniques so as to reduce overdispersion which is rampant in real life ecological datasets.

Objectives of the study

- To understand the application of machine learning techniques in population ecology as published in literature, in order to verify whether these machine learning techniques can effectively deal in practice with the problem of species abundance and habitat suitability in ecological count data.

- To model zero-inflated overdispersed ecological count data on single species using count regression and ML regression techniques and compare their performances.
- To use machine learning techniques to classify multiple species on multi-class imbalanced data.

1.10 Justification of the study

This study focuses on traditional techniques that has been used to deal with overdispersion and propose new strategies to address the above-mentioned issues in population ecology. Overdispersion is a serious problem because it can bias both the means and standard errors of parameter estimation (Hilbe 2011*b*). Ecologists are often prone to making mistakes during analysis or when trying to establish relationships between multiple features in the dataset. This makes it difficult for them to quickly find solutions to problems of this nature. This study give insights on machine learning techniques in the area of population ecology to overcome these problems.

Knowing the best modeling technique to use in ecological count data should lead to the development of better predictions. It is hoped that South African Institute for Aquatic Biodiversity (SAIAB) will benefit significantly after the completion of this study. Hence, this study will help to advance their methods and improve the quality of their findings. Future research on statistical methods proposed in this study need to focus on ways of improving the computational efficiency of the models. This is vitally important for practical use.

1.11 Ethical considerations

This study was submitted to North-West University Ethics Committee for their approval to conduct the research. In conducting this study, secondary data will be used. The permission to use the data will be sought from South African Institute for Aquatic Biodiversity (SAIAB). The study will adhere to any ethical considerations proposed by the North-West University.

1.12 Outline of the thesis

This thesis is presented in an article format that consists of three chapters. Each chapter is written as a peer-reviewed article. The rest of the thesis is organized as follows:

- **Chapter 2:** Presents a review of machine learning techniques with their applications in ecological count dataset. The aim of this chapter is to predict the relationship between multiple species occurrences and habitat suitability by evaluating the predictive performance of classifiers on imbalanced and balanced datasets using machine learning techniques.
- **Chapter 3:** Presents a study of count regression and machine learning techniques for zero-inflated overdispersed count data: application to ecological count data. This chapter gives a comparison of count regression and machine learning models within the context of single species count dataset.
- **Chapter 4:** Presents a study on the approach to multi-class imbalanced problem in ecology using machine learning techniques. The chapter provides methods that should be considered when addressing the multi-class imbalanced classification problem. The overall aim of this chapter is to determine which machine

learning approach best classifies multiple species.

- **Chapter 5:** Concludes this study and gives suggestions for future research.

References are provided at the end of all chapters. They are listed according to the requirements stipulated in the manual for post-graduate studies of the North-West University.

Chapter 2

2. A review of machine learning techniques with their applications in ecological count dataset

Abstract

Species distribution models (SDMs) framework involves various statistical methods which are available to model the spatial distribution of a target species. Generalized linear models (GLMs) are the most commonly used methods in species distribution modeling since they can model the relationship between the response and multiple predictor variables. Species may be present in unsuitable habitat and absent in suitable habitat, hence relationships between habitat and density can lead to incorrect inference regarding the importance of environmental resources to species. Much work in the literature that has been covered mostly focused on SDMs for predicting the relationship of single species and their environmental relevant covariates and very few studies on prediction of multiple species. This study aims to predict the relationship between multiple species occurrence and habitat suitability. We also noticed that predicting multiple species result in multi-class imbalanced classification problem. Hence, ensemble machine learning (ML) techniques namely; random forests (RF), bagging classification trees (bagCART), gradient boosting models (GBM), eXtreme boosting models (XGB) and adaptive boosting models (AdaBoost) are proposed for predicting the relationship of species abundance and habitat suitability. We compared the performance of ML classifiers on imbalanced and balanced datasets for five species. The F1-score and G-mean were used to measure the model performances. The bagging classifiers; RF (F1-score = **100**, G-mean = **100**) and bagCART (F1-score = **100**, G-mean = **100**) outperform the boosting classifiers; GBM (F1-score = 88.9, G-mean = 89.4), XGB (F1-score = 84.2, G-mean = 89.4) and AdaBoost

(F1-score = 79.5, G-mean = 79.8). The comparison indicated that ML techniques performed better on the balanced dataset. The findings of this study offer insights into the applicability of ensemble ML techniques on the balanced dataset for multiple species. The findings of this study demonstrated that ML techniques provide accurate predictions for predicting the relationship between multiple species abundance and habitat suitability.

Keywords: Ensemble techniques, imbalanced dataset, machine learning, multi-class classification, species abundance, species distribution models

2.1 Introduction

Classifying species into biological class is the first priority towards examining the underlying biological and ecological factors leading to different statistical distributions of species between habitats (Chazdon et al. 2011). Species distribution modeling techniques has been developed for fitting models that defines the relationships between observations of species occupancy and their environmental relevant covariates (Samal et al. 2022). Species distribution models (SDMs) are mostly used in ecology to predict the performance of species presence-only dataset (Valavi et al. 2022) and presence-absence dataset (Beery et al. 2021) or to quantify habitat suitability as a function of multiple predictor variables representing key environmental conditions linked to species habitat usage and persistence (Yu et al. 2020). Ecological methodologies has been improved and more advanced statistical methods have been proposed and implemented for SDMs to increase the predictive performance of models. The use of ML techniques in SDMs has increased, with boosted regression trees (BRT) models being the widely used techniques (Yu et al. 2020). The presence-absence dataset is used in this study so that the SDMs can be applied with the proposed ML techniques in order to reflect on the relationship between various parameters in the prediction.

Machine learning (ML) techniques are emerging with great solutions for the advancement of modeling and predictions of ecological count datasets. ML techniques have unique algorithms for unraveling the complexity of datasets and predictability of count data in ecology. Some ML techniques accommodate zeros both in the response and in the explanatory variables, that is what makes them unique and powerful alternative approaches to statistical methods. Various ecology researchers (Crisci et al. 2012, Cutler et al. 2007, Farrell et al. 2019, Olden et al. 2008, Drake et al. 2006, Elliott & Morrell 2009) attempted to popularize the use of ML techniques to ecologists through a series of comparisons by identifying strengths and weaknesses for different applications. What makes ML techniques so broadly applicable is the diversity of ML algorithms capable of performing very well under messy real-world conditions (Thessen 2016). However, despite their ability to model messy datasets, there are limitations that are associated with ML techniques when dealing with a classification problem. These include the (1) multi-class classification problem, where the variable of interest consists of more than two classes, (2) imbalanced classification problem, where the sample size of each class outnumbers other classes by a large proportion. The multi-class imbalanced classification is a growing problem within the ecological community. This study address applications of ensemble machine learning (ML) algorithms to a multi-class imbalanced ecological dataset. Khushi et al. (2021) applied different resampling techniques on imbalanced medical data using ML techniques. Based on our knowledge, there is limited number of studies in the literature (Yu et al. 2020, Rew et al. 2021, Carrasco et al. 2021, Rustamov 2022) that compared the predictive performance of ML techniques in combination with different resampling techniques for SDMs on the multi-class imbalanced dataset.

Studies by Lango (2019), Bagheri et al. (2012) and Sahare & Gupta (2012) propose techniques for a decomposition of multi-class classification into binary classification in combination with the balancing techniques. These approaches allow a straightforward comparative evaluation of statistical models and ML techniques. Here, we recommend the data resampling techniques which are one of the pre-processing approaches that

are conducted during training models. Shin et al. (2021) has used the data resampling techniques to improve prediction of cyanobacteria blooms and Muñoz-Mas et al. (2019) to unveil the microhabitat suitability for the invasive bleak *Alburnus alburnus* L. and pumpkinseed *Lepomis gibbosus* L.. Resampling is performed by (1) increasing the instances of the minority class (over-sampling), (2) decreasing the instances of the majority class (under-sampling) or combining (1) and (2). The literature provides a list of the recently developed resampling techniques such as synthetic minority over-sampling (SMOTE) (Bajer et al. 2019), adaptive synthetic (ADASYN) (He et al. 2008), random over-sampling examples (ROSE) (Cano & Krawczyk 2022) and mahalanobis distance based over-sampling technique (MDO) (Liu et al. 2021). Shidha & Mahalekshmi (2019) have used several resampling techniques to tackle the imbalanced problem on the binary dataset. In this study, we apply different resampling techniques on multi-class imbalanced dataset using ensemble learning algorithms.

Ensemble learning algorithms consists of a combination of commonly used classifiers such as decision trees. Ensemble learning algorithms are used efficiently and successfully in many research fields such as credit evaluation (Kou et al. 2022), landslides (Hu et al. 2021), remote sensing (Rew et al. 2020, 2021) and image classification (Tuda & Luna-Maldonado 2020). Rew et al. (2020) used deep neural network to predict habitat suitability based on observations and environmental data. This chapter aims to review the predictive performance of five ensemble ML techniques; adaptive boosting (AdaBoost), gradient boosting machines (GBM), eXtreme gradient boosting (XGB), random forests (RF) and bagging classification trees (bagCART). For this purpose, these ensemble algorithms were explored using multiple species dataset to check their performances. The objectives are (1) to find out which ML algorithm performs better with resampling techniques and (2) to understand how well the best performing model on unseen provided test dataset.

In this chapter, we hypothesized that the random over-sampling (ROS) techniques in combination with ensemble ML techniques are suitable for predicting species habitat relationship and also handling the imbalanced data problem. This study is orga-

nized as follows. Section 2.2 discusses some ecological processes that are encountered by ecologists. Section 2.3 gives an overview of generalized linear models in count data. Section 2.4 discusses some ensemble ML algorithms in detail. Section 2.5, discusses the classification of data with multi-class imbalanced problem and model evaluation measures. Section 2.6 shows the experimental results on species abundance and habitat suitability. Lastly, Section 2.7 concludes this chapter and gives future recommendations.

2.2 Ecological processes

This section gives a review on ecological processes that generate species abundance and occupancy. Species abundance and occupancy surveys involve visiting sites multiple times within a season and examining relevant sampling units where the target species is either detected or non-detected (Wenger & Freeman 2008, Royle et al. 2005). Occupancy is given by $\Phi_i \times \pi_{ij}$ where Φ_i represents the probability that a site is occupied and π_{ij} represents the probability of detecting a species at site i during survey j . Species often remain undetected even when present at a site (false zeros) (Zuur et al. 2009). The statistical modeling techniques to count datasets such as Poisson generalized linear models, negative binomial, zero-inflated Poisson and zero-inflated negative binomial models are discussed in the literature (Cheung 2002, Rose et al. 2006, Minami et al. 2007, Wagh & Kamalja 2018, Araújo et al. 2021).

Zeros can be informative about ecological processes, hence it is advisable to not ignore or exclude them without some thought about how they could be used. This is because many zeros may cause the variance to increase more than the mean (overdispersion). These sources of zeros are the ones that depict if species are abundant at a certain location. Species abundance and site occupancy (occurrence) of species are useful measures of population status and are of considerable interest to ecologists (Wenger & Freeman 2008). Hence, species distribution models (SDMs) are used to predict

whether or not a species is present in a particular sampling location.

2.3 Generalized linear models

The GLM has been commonly used as a benchmark for species distribution models. This method has provided powerful tools in many research fields including ecological research (Zuur et al. 2009). Generalized linear models (GLMs) are suitable for a linear observation of explanatory variables. In the general framework, the GLM consists of three primary components: the random component which identifies the response variables $Y_1, Y_2, \dots, Y_n$ and its probability distribution function (pdf); the systematic component which specifies the set of unknown parameters β and explanatory variables $\mathbf{X}$; and a link function which describes how the mean response, $E(Y) = \mu$ is linked to the covariates through the linear predictor. In GLMs, the link function is assumed to be known. The Poisson and negative binomial distributions are commonly used to model count data and are members of the exponential family. However, the Poisson regression model is restricted when there is overdispersion.

2.4 Ensemble machine learning algorithms

This section discusses ensemble ML techniques in detail. Ensemble learning algorithms allow the combination of powerful classifiers that are applicable in different research fields. Combining multiple classifiers reduces chances of overdispersion and increases model accuracy. In addition, ensemble learning algorithms produce accurate results compared to single classifiers. Ensemble learning algorithms are mostly achieved by incorporating data resampling methods (RUS and ROS) and single classifiers. The ensemble modeling techniques have been recently used in numerous studies and they provided the best modeling techniques (Cutler et al. 2007, Thessen 2016).

The most recently used approaches of ensemble methods are the bagging (RF and bagCART) and boosting (AdaBoost, GBM and XGB) algorithms.

2.4.1 Bagging

Bagging is a general-purpose procedure for reducing the variance of the machine learning (ML) method and also minimize overfitting. It is particularly useful and frequently used in the context of decision trees. Bagging has been applied frequently in fields such as bio-statistics (Cushman & Huettmann 2010) and remote sensing (Saini & Ghosh 2017) but it has not been often applied in the field of ecology. Bagging is used in many situations where it is hard or impossible to directly compute the standard deviation of a quantity of interest. The natural way to compute parameters using ML techniques is to take many data sets and average the resulting predictions.

We calculate $\hat{f}^1(x), \hat{f}^2(x), \dots, \hat{f}^B(x)$ using B separate data sets and average them in order to obtain a single low-variance ML method given by (James et al. 2013):

$$\hat{f}_{avg}(x) = \frac{1}{B} \sum_{b=1}^B \hat{f}^b(x).$$

This can be practical when we generally have access to multiple data sets. Instead, when we do not have, we bootstrap by taking repeated samples from the single data set. We generate B different bootstrapped data sets in order to get $\hat{f}^{*b}(x)$ and average all the predictions to obtain (James et al. 2013):

$$\hat{f}_{bag}(x) = \frac{1}{B} \sum_{b=1}^B \hat{f}^{*b}(x).$$

The basic idea underlying bagging is the recognition that part of the output error in

a single regression tree is due to the specific choice of the training data set (James et al. 2013).

(a) Random Forests

Random Forest (RF) is an ensemble classifier that uses a large number of decision trees in order to overcome the weakness of a single decision tree (Cutler et al. 2007). In this method, the output of all the decision trees in the RF, are combined to make the final prediction. The final prediction of the RF algorithm is derived by polling the results of each decision tree or just by going with a prediction that appears the most times in the decision trees. One of the main advantages of RF is the resistant to over training and growing a large number of random forest trees where it does not create a risk of overfitting. The RF algorithm can handle numerical, binary and categorical features without scaling, transformation, or modification. It has resistance to outliers in predictors and automatically handles the missing data. RF implicit feature selection as it gives estimates on what variables are important in the classification.

The RF algorithm possesses a number of appealing properties making it well-suited for classification of metagenomic data (Lo & Marculescu 2019): (i) it is applicable when there are more predictors (features) than observations; (ii) it performs embedded feature selection and it is relatively insensitive to the large number of irrelevant features; (iii) it incorporates interactions between predictors; (iv) it is based on the theory of ensemble learning that allows the algorithm to learn accurately both simple and complex classification functions; (v) it is applicable for both binary and multi-class classification tasks; and (vi) according to its inventors, it does not require much fine tuning of hyper-parameters and the default parameterization often leads to excellent classification accuracy (Lo & Marculescu 2019). Cutler et al. (2007) has demonstrated the ability of RF for (a large-scale prediction of tree species) classification of plant species.

(b) Bagging classification and regression trees

The bagging and regression trees (bagCART) is known by its effectiveness to decrease the overdispersion (high variance) problem and highly improves the classification performance. Bagging eliminates overdispersion by creating its own variance among the data by sampling and modifying the data when testing multiple models. Thus, it is believed that applying this ensemble classifier in population ecology can achieve biologically meaningful results.

2.4.2 Boosting

Boosting is an iterative technique allowing to obtain powerful algorithms based on weak predictors or classifiers (Trevor et al. 2009, Crisci et al. 2012). Boosting techniques are designed to fit many simple models whose predictions are then combined to give more robust estimates of the relationship between species distribution and a set of environmental variables (Trevor et al. 2009). Boosting algorithm can be applied to various statistical techniques for both regression and classification. There are many variants of this algorithm; the first commonly used boosting algorithm is adaptive boosting (AdaBoost) (Crisci et al. 2012). AdaBoost was designed for binary classification problem. As boosting can be applied to a variety of tasks to minimize loss functions, it is generally high resistant to overfitting.

(a) Gradient boosting models

The gradient boosting models (GBM) combine weak learners into a strong learner in an iterative way. Suppose for a given dataset $D = \{\mathbf{x}_i, y_i\}_i^N$, the goal of gradient boosting is to find an approximation, $\hat{F}(\mathbf{x})$, of the function $F^*(\mathbf{x})$ which maps instances $\mathbf{x}$ to their output values y by minimizing the expected value of a given loss function, $L(y, F(\mathbf{x}))$. GBM builds an additive approximation of $F^*(\mathbf{x})$ as a weighted

Algorithm 1 The boosting algorithm.

Input: Dataset $\mathbf{Z} = \{z_1, z_2, \dots, z_N\}$ with $z_i = (x_i, y_i)$ where $x_i \in \mathbf{X}$ and $y_i \in \mathbf{Y} = \{-1, +1\}$, B , the number of bootstrap samples.

Output: A classifier $H : \mathbf{X} \rightarrow \{-1, +1\}$

1. Randomly select without replacement, $L_1 < N$ samples from $\mathbf{Z}$ to obtain Z_1^* .
 2. Run the weak learner on Z_1^* , yielding classifier H_1 .
 3. Select $L_2 < N$ samples from Z , with half of the samples misclassified by H_1 , to obtain Z_2^* .
 4. Run the weak learner on Z_2^* , yielding classifier H_2 .
 5. Select all samples from Z on which H_1 and H_2 disagree, producing Z_3^* .
 6. Run the weak learner on Z_3^* , yielding classifier H_3 .
 7. Produce the final classifier as a majority vote: $H_x = (\sum_{b=1}^3 H_b(x))$.
-

sum of functions:

$$F_m(\mathbf{x}) = F_{m-1}(\mathbf{x}) + \rho_m h_m(\mathbf{x}),$$

where ρ_m represents the weight of the m^{th} function, $h_m(\mathbf{x})$. These functions are associated with ensemble classifiers. The approximation is built iteratively and the first constant approximation is obtained as:

$$F_0(\mathbf{x}) = \arg \min_{\alpha} \sum_{i=1}^N L(y_i, \alpha).$$

The models are expected to minimize:

$$(\rho_m, h_m(\mathbf{x})) = \arg \min_{\rho, h} \sum_{i=1}^N L(y_i, F_{m-1}(\mathbf{x}) + \rho h(\mathbf{x})).$$

However, instead of solving the optimization problem directly, each h_m can be seen as a greedy step in a gradient descent optimization for F^* . Each model, h_m , is trained on a new dataset $D = \{\mathbf{x}_i, r_{mi}\}_{i=1}^N$ where the pseudo-residuals, r_{mi} , are calculated by:

$$r_{mi} = \left[\frac{\partial L(y_i, F(\mathbf{x}))}{\partial F(\mathbf{x})} \right]_{F(\mathbf{x})=F_{m-1}(\mathbf{x})}.$$

The value of ρ_m is subsequently computed by solving a line search optimization problem. The GBM can suffer from overfitting if the iterative process is not properly regularized. The natural way to regularize GBM is to apply shrinkage to reduce each gradient descent step $F_m(\mathbf{x}) = F_{m-1}(\mathbf{x}) + v \rho_m h_m(\mathbf{x})$ with $v = (0, 1.0)$ (Bentéjac et al. 2021).

(b) eXtreme boosting models

The eXtreme gradient boosting (XGB) model is an ensemble classifier that is based on decision tree algorithm and can be used for both regression and classification. XGB is similar to GBM, it builds an additive expansion of the objective function by minimizing a loss function. Now that XGB focused only on decision trees as base classifier, a variation of the loss function is used to control the complexity of trees:

$$L_{xgb} = \sum_{i=1}^N L(y_i, F(\mathbf{x})) + \sum_{m=1}^M \Omega(h_m)$$

and

$$\Omega(h) = \gamma T + \frac{1}{2} \lambda ||\omega||^2,$$

where T is the number of leaves of the tree and ω are the output scores of the leaves. Higher values of γ result in simpler trees and also controls the minimum loss reduction gain needed to split an internal node (Bentéjac et al. 2021). Randomization techniques are also applied in XGB to reduce overfitting and to increase the training speed. The randomization techniques included in XGB are; random sub-samples to train individual trees, column sub-sampling at tree and tree node levels (Bentéjac et al. 2021). Moreover, XGB can be extended to any user defined loss function by defining a function that outputs the gradient and the hessian and passing it through the hyper-parameter (Bentéjac et al. 2021).

(c) Adaptive boosting models

The adaptive boosting (AdaBoost) algorithm is designed for a binary classification problem. Algorithm 2 shows the details of the AdaBoost algorithm. AdaBoost algorithm reduces to the optimization of an additive model (Sun et al. 2006).

Algorithm 3 represent the AdaBoost.M1 algorithm as described by James et al. (2013). Sun et al. (2006) extended the AdaBoost algorithm to AdaBoost.M1 algorithm to accommodate multi-class classification problem. The straightforward gen-

Algorithm 2 AdaBoost algorithm.

Given: $(x_1, y_1), \dots, (x_n, y_n)$ where $x_i \in \mathbf{X}$, $y_i \in \mathbf{Y} = \{-1, +1\}$

Initialize $D^1(i) = \frac{1}{n}$

For $t = 1, \dots, T$:

1. Train base learner $h_t \rightarrow$ using distribution D^t
2. Choose weight updating parameter: α_t
3. Update and normalize sample weights:

$$D^{t+1}(i) = \frac{D^t(i) \exp(-\alpha_t h_t(x_i) y_i)}{Z_t} \quad (2.4.1)$$

where Z_t is a normalization factor.

Output the final classifier:

$$H(x) = \text{sign} \left(\sum_{t=1}^T \alpha_t h_t(x_i) \right) \quad (2.4.2)$$

eralization is called AdaBoost.M1 and is adequate when the base learner is effective enough to achieve reasonably high accuracy (Sun et al. 2006). AdaBoost.M1 is slightly differs from AdaBoost. The main difference is the replacement of the weight update formula of equation (2.4.1) by:

$$D^{t+1}(i) = \frac{D^t(i) \exp(-\alpha_t h_t(x_i) y_i)}{Z_t}, \quad (2.4.3)$$

where Z_t is a normalization factor and

$$I[h_t(x_i)] = \begin{cases} +1 & \text{if } h_t(x_i) = y_i \\ -1 & \text{if } h_t(x_i) \neq y_i, \end{cases} \quad (2.4.4)$$

and the final hypothesis of Equation (2.4.2) by:

$$H(x) = \arg \max_{c_i} \left(\sum_{t=1}^T \alpha_t [h_t(x) = C_i] \right). \quad (2.4.5)$$

Algorithm 3 AdaBoost.M1 algorithm

1. Initialize the observation weights $w_i = \frac{1}{N}, i = 1, 2, \dots, N$.
2. For $m = 1$ to M : Fit a classifier $G_m(x)$ to the training data using weights w_i .
Compute

$$err_m = \frac{\sum_{i=1}^N w_i I(y_i \neq G_m(x_i))}{\sum_{i=1}^N w_i}.$$

Compute $\alpha_m = \log((1 - err_m)/err_m)$.

Set $w_i \leftarrow w_i \times \exp[\alpha_m \times I(y_i \neq G_m(x_i))], i = 1, 2, \dots, N$.

3. Output $G_x = \text{sign}[\sum_{m=1}^M \alpha_m G_m]$.
-

2.5 Classification

This section reviews the supervised machine learning concept called classification. Classification is a data mining function that allocates items in a gathering to target classes. Machine learning splits classification problems into binary, multi-class, multi-label and hierarchical tasks (Thessen 2016). Classification problems that have multiple classes with imbalanced datasets present a different challenge than a binary classification. When the number of classes exceeds two, the classification problem is said to be a multi-class classification problem and when some classes outnumber other classes by a large proportion it is said to be an imbalanced problem.

The multi-class imbalanced problem is a subset of supervised machine learning tasks where the classification variable of interest consists of more than two unequal sample sizes. Imbalanced data has become one of the major research investigations in machine learning (ML). In this work we present multi-class imbalanced classification

problem within the context of ecology. Ecological datasets are known to be unbalanced, complex, non-linear and often contain missing values. Most learning systems assume that training sets used for learning are balanced (Dastile et al. 2020). However, this is not always the case in real world datasets where one class might be represented by a large number of examples, while the other is represented by only a few. The process of some classes outnumber other classes is known as the class imbalanced and is often reported as a problem in ML models. Matthews et al. (2018) highlights the difficulties shown by class imbalanced as relatively dependent upon the imbalanced ratio (IR) and the classifiers involved. IR is defined as:

$$IR = \frac{n_{maj}}{n_{min}},$$

where n_{maj} represent the sample size from the majority class and n_{min} represent the sample size from the minority class. Matthews et al. (2018) clearly stated that there is no defined threshold on IR, although sometimes as a rule of thumb, $IR > 1.5$ is used to identify imbalanced data.

2.5.1 Approaches for multi-class imbalanced classification

This sub-section discusses the methods that address the imbalanced problem by rebalancing the class distribution through resampling the dataset. These methods can be grouped according to the way in which the training sample is selected. These include either over-sampling the minority class or under-sampling the majority class.

Over-sampling techniques are often applied than other techniques and they increase the weight of the minority class by replicating minority class samples. The random under-sampling (RUS) technique eliminates sample from the majority class. The drawback or disadvantage is that this technique may throw out potential useful data. The random over-sampling (ROS) technique replicates samples from the

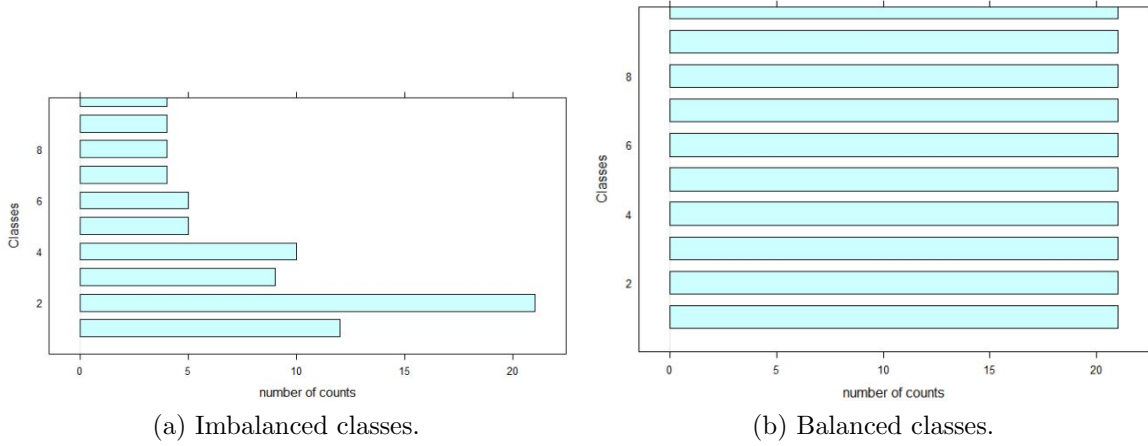


Figure 2.1: Graphical representation of imbalanced and balanced classes.

minority class until a class balance is achieved. Several studies have recommended the use of over-sampling technique since it usually performs better than the under-sampling. Figure 2.1 shows a dataset of both imbalanced and balanced classes. Figure 2.1(a) denotes the original dataset and Figure 2.1(b) depicts dataset after ROS has been performed. It can be seen from Figure 2.1(b) that the instances in the data are balanced/over-sampled in a way that all classes have the same size. ROS artificially increases the size of the dataset.

Synthetic minority over-sampling technique (SMOTE) increases the number of minority instances in the dataset by creating the synthetic minority class instances using interpolation technique. SMOTE uses the k -nearest neighbors for searching minority instances (Basgall et al. 2018). Another way of solving the multi-class imbalanced problem is to apply the decomposition technique.

2.5.2 Decomposition of multi-class classification into binary classification

Most attention in the literature was devoted to class decomposition, converting a multi-class problem into a set of two-class sub-problem. The multi-class classification problem can be simplified into several binary classification tasks such as one-versus-all (OVA), one-versus-one (OVO), all-versus-all (AVA), and error-correcting output-coding (ECOC). The one-versus-one (OVO) technique resolves the multi-class imbalanced problem in such a way that the training dataset is partitioned into several instances where each instance contains a distinct pair of class values. As a result, if there is C number of distinct class values then the total number of instances will be: $C(C - 1)/2$ (Adnan & Islam 2015). The one-versus-all (OVA) technique resolves the multi-class imbalanced problem in a way that the training dataset is replicated into several datasets where each replica contains one original class value and all other class values of the training dataset are replaced as “O” (Adnan & Islam 2015). Hence, if there is C number of distinct class values then the total number of replicas will be C . Basically, the number of records in each replica is the same as the original training dataset. The error-correcting output-coding (ECOC) framework is to decompose a multi-class problem into a large binary problems (Bagheri et al. 2012). ECOC can be broken down into encoding and decoding stages. The encoding stage decomposes matrix for the given problem. The matrix M of 1, -1 values of size $C \times L$, where L denotes the number of codewords codifying each class and C is the number of classes. The matrix is interpreted as a set of L binary learning problems, one for each column (Bagheri et al. 2012).

The decomposition process suffers from the loss of valuable information about the class relationships. For instance, the OVA technique discards the inner-class relationship by merging classes into a single majority. Hence, we recommend the use of resampling techniques combined with ensemble learning to solve the multi-class imbalanced problem.

2.5.3 Evaluation criteria

The evaluation of the performance of ML classifiers is an important task of comparing and finding the best model for a given dataset. There are numerous performance metrics that are used to measure the performance of machine learning algorithms. The performance evaluation measures can be derived from a confusion matrix.

Suppose y_{ij} represents species count dataset for all sites $i = 1, \dots, n$; for all visits $j = 1, \dots, m$ and for all species detected or non-detected during sampling survey. For detection of species, let $\theta_k = [\theta_{k1}, \dots, \theta_{kK}]'$ be a vector containing the probabilities of classifying a detection of each species, where k denotes each species. A confusion matrix, θ , is defined using the probability vectors from every species so that:

$$\theta = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \vdots \\ \theta_K \end{bmatrix} = \begin{bmatrix} \theta_{11} & \theta_{12} & \dots & \theta_{1K} \\ \theta_{21} & \theta_{22} & \dots & \theta_{2K} \\ \vdots & \vdots & \ddots & \vdots \\ \theta_{K1} & \theta_{K2} & \dots & \theta_{KK} \end{bmatrix},$$

with row entry corresponding to the actual recorded species and column entry indicating the species assigned from the classification algorithm. The diagonal elements of θ correspond to the probabilities of correctly specified species.

The confusion matrix is used as a basis in measuring a classifier, where actual class of the instances are denoted in rows and the predicted class are denoted in columns. Table 2.1 shows a typical confusion matrix for a binary classification, where:

- TP represent the number of positive instances that are correctly classified.
- TN represent the number of negative instances that are correctly classified.

- FP represents the number of negative instances that are incorrectly classified as positive samples.
- FN represents the number of positive instances that are incorrectly classified as negative samples

Table 2.1: Confusion matrix for SDM evaluation.

		Predicted class	
		Positive (P)	Negative (N)
Actual class	Positive (P)	TP	FN
	Negative (N)	FP	TN

In a multi-class classification, the number of classes k must be greater than two. Under this construction, the confusion matrix is guaranteed to be a square matrix with an equal number of rows and columns. In a multi-class application of k classes, the confusion matrix can be presented in Table 2.2, where C_i denotes the class label of the i^{th} class.

Table 2.2: An example of a multi-class confusion matrix.

		Predicted class			
		Class 1	Class 2	...	Class k
Actual class	Class 1	c_{11}	c_{12}	...	c_{1k}
	Class 2	c_{21}	c_{22}	...	c_{2k}
	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
	Class k	c_{k1}	c_{k2}	...	c_{kk}

A variety of performance metrics are calculated through the confusion matrix results. These metrics includes accuracy, Kappa statistic, precision, recall, balanced accuracy, detection rate and detection prevalence. This also includes combined evaluation measures such as Matthew's Correlation Coefficient and G-mean.

Comparing different ML classifiers using performance metrics is easy, but this has posed several problems such as sensitivity to imbalanced datasets and multi-class classification to name just two which occur when dealing with multiple species. With regards to performance evaluation in the presence of multi-class imbalanced data problems, accuracy has proved to be the most commonly used performance measure (Collell et al. 2016). However, for a classification model trained from a dataset with imbalanced classes, accuracy is no longer regarded as a proper measure because rare classes have minimal impact on accuracy than prevalent classes. There are widely accepted performance measures in multi-class imbalanced learning, that are computed from the confusion matrix. These include recall, precision, F-score and G-mean which are highly recommended and deemed as more advanced evaluation measures (Sun et al. 2006). The formulas for performance metrics are

Precision:

$$Precision = \frac{TP}{TP + FP}.$$

Recall:

$$Recall = \frac{TP}{TP + FN}.$$

F1-score:

$$F1 - score = 2 \times \frac{TP}{TP + FP + FN}.$$

The F1-score is the harmonic mean of precision and recall. This metric is affected by the changes in the class distribution. F1-score is suggested when the imbalanced

problem exists in the dataset.

Geometric mean:

$$G - mean = \sqrt{\frac{TP}{TP + FN} \times \frac{TN}{TN + FP}}.$$

Geometric mean (G-mean) is used to induce a classifier in a domain with imbalanced class representation, for example if the scientist wants to achieve similar performances on both classes, positive and negative. G-mean is given as the square root of the product of sensitivity and specificity (Tharwat 2020). It can be used with balanced and imbalanced datasets.

The negative classes of the G-mean metric cancel each other, hence it is suitable for imbalanced data. G-mean is used to measure the balanced performance among classes of a classification output. Further investigations are needed to explain what problems multi-class imbalanced datasets can cause in the existing class balanced techniques and how it affects the performance of classifiers. Such information can be informative to understand the multi-class imbalanced data problem better and can assist in developing better solutions.

Matthew's Correlation Coefficient:

Matthew's Correlation Coefficient (MCC) measure has a connection to the χ^2 distribution and also used in multi-class data problem. MCC is defined as:

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}.$$

MCC measure is generally considered as a balanced measure, even if the size of the classes is different. This measure denotes the relationship between the observed and predicted classifications and is calculated directly from the confusion matrix. A MCC

coefficient of +1 indicates a perfect prediction, -1 indicates a total disagreement between prediction and true values and 0 means that no better than random prediction (Tharwat 2020). In any multi-class imbalanced data problem, a multi-minority case is one where a single class has a significantly larger proportion than the average size of all other classes. MCC is sensitive to imbalanced data.

2.6 Experimental results

This section provides the results of machine learning algorithms discussed above. Machine learning algorithms are explored to describe the distribution of five fish species sampled from different habitats of Uganda. The results are presented in percentages.

2.6.1 Data description

The datasets utilized in this study were collected within major lakes and rivers in Uganda. The dataset consists of 364 observations. The experiments are based on five fish species namely; *Bagrus docmak* (Bagridae), *Brycinus sadleri* (Alestidae), *Mormyrus kannume* (Mormyridae), *Oreochromis leucostictus* (Cichlidae) and *Synodontis victoriae* (Mochokidae). The majority observations were found in the rivers and lakes of Uganda. The number of instance, number of attributes per class level and imbalanced ratio (IR) are described in Table 2.3. The IR is calculated by the majority number of classes divided by the minority number of classes. It is observed that the IR is greater than 1.5, hence the proposition of resampling techniques to handle the imbalanced problem.

Table 2.3: The multi-class imbalanced dataset used in the experiments.

Species dataset	Instances	Classes	IR	Class distribution
<i>Bagrus docmak</i>	47	5	9.5	4/19/11/4/2
<i>Brycinus sadleri</i>	106	8	7.67	7/23/12/6/4/5/3/3
<i>Mormyrus kannume</i>	126	11	20	40/19/14/11/6/3/13/5/4/2/2
<i>Oreochromis leucostictus</i>	77	9	8	14/16/12/5/3/4/3/2/2
<i>Synodontis victoriae</i>	85	9	15	4/30/17/8/3/4/2/2/2

Model validation

To validate the performance of classifiers, the dataset has been randomly split into 70% for training model and 30% for testing model. All the resampling techniques are applied to the training dataset and the remaining testing data is used for model validation.

2.6.2 Results and discussion

This study attempts to demonstrate the effect of multi-class imbalanced classification problem and handle this problem applying various resampling techniques to determine the best classifier and resampling technique. The presented results are coded in R version 4.1.3 programming language (R Core Team 2022). The classifiers are first executed on the multi-class imbalanced dataset to determine the effect of imbalanced data problem on the classifiers performance. The classifiers are then implemented on the balanced dataset to identify the effectiveness of the resampling techniques as a way of solving the imbalanced problem. Table 2.4 shows the species occurrences and richness between the lake and river habitats. All the species are present in both lakes

and rivers. *Mormyrus kannume* species were abundant in the river habitat than the lake. *Brycinus sadleri* and *Oreochromis leucostictus* were found in the lake habitat. With the occurrence rates over 80%, these two species can be considered abundant in lake systems. Figure 2.2 shows that more species occur in lakes (270) than in rivers (171), this might be due to the fact that small lakes are more likely to be safe for species than large rivers.

Table 2.4: Species habitat suitability.

Species	Lake	River
<i>Bagrus docmak</i>	18	29
<i>Brycinus sadleri</i>	101	5
<i>Mormyrus kannume</i>	26	100
<i>Oreochromis leucostictus</i>	76	1
<i>Synodontis victoriae</i>	49	36

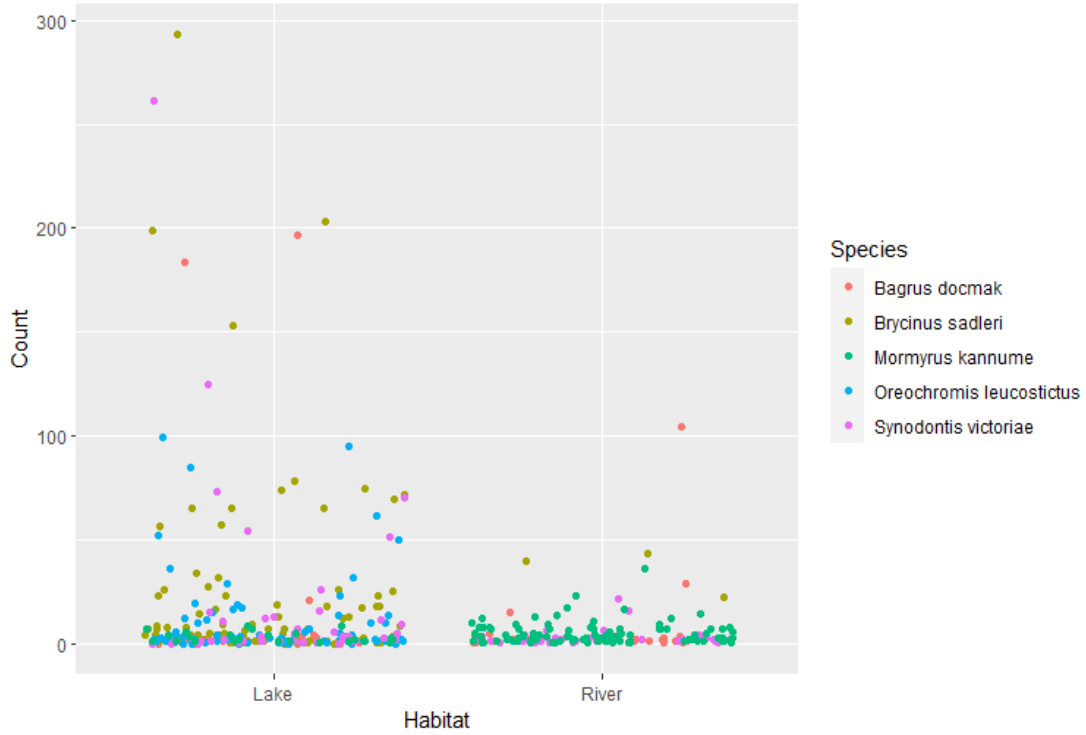


Figure 2.2: Graphical representation of species habitat.

Table 2.5 shows different testing results of ML ensemble classifiers on the multi-class imbalanced dataset. Various performance metrics such as precision, recall, F1-score and G-mean are used to determine the predictive performance of classifiers. For the RF classifier, we see that *Bagrus docmak* and *Synodontis victoriae* species achieved precision score of 100. The RF classifier achieved a high recall score of 90.6 on *Brycinus sadleri* species. The F1-score and G-mean for this classifier are both, 78.5 on *Mormyrus kannume* species.

The bagCART classifier achieved high precision scores of 100 on *Bagrus docmak* and *Synodontis victoriae* species. There is only one species, *Mormyrus kannume*, that showed a good performance under recall (84.6), F1-score (78.6) and G-mean (78.8). The *Synodontis victoriae* species showed the least performance on recall (6.3), F1-

score (11.8) and G-mean (25.0).

The AdaBoost classifier achieved high predictive performance metric scores on *Mormyrus kannume* species (precision = 75.6, recall = 79.5, F1-score = 77.5 and G-mean = 77.5). The *Synodontis victoriae* species performance slightly improved under this classifier, compared to under the RF, bagCART, GBM and XGB classifiers.

For the XGB classifier, *Mormyrus kannume* achieved a precision score of 77.5 and *Brycinus sadleri* achieved a high recall score of 90.6. The F1-score and G-mean for this classifier are both 78.5 under *Mormyrus kannume* species. The GBM classifier showed a good predictive performance only on *Mormyrus kannume* species. The precision, recall, F1-score and G-mean scores are 75.6, 79.5, 77.5 and 77.5, respectively. *Mormyrus kannume* species has showed consistent predictive performance under the GBM classifier.

Table 2.5: Classifiers results on imbalanced dataset.

Model	Species	Precision	Recall	F1-score	G-mean
RF	<i>Bagrus docmak</i>	100	60.0	75.0	77.5
	<i>Brycinus sadleri</i>	56.9	90.6	69.9	71.8
	<i>Mormyrus kannume</i>	77.5	79.5	78.5	78.5
	<i>Oreochromis leucostictus</i>	53.8	43.8	48.3	48.5
	<i>Synodontis victoriae</i>	100	6.3	11.8	25.0
bagCART	<i>Bagrus docmak</i>	100	60.0	75.0	77.5
	<i>Brycinus sadleri</i>	60.5	81.3	69.3	70.1
	<i>Mormyrus kannume</i>	73.3	84.6	78.6	78.8
	<i>Oreochromis leucostictus</i>	50.0	50.0	50.0	50.0
	<i>Synodontis victoriae</i>	100	6.3	11.8	25.0
AdaBoost	<i>Bagrus docmak</i>	66.7	40.0	50.0	51.6
	<i>Brycinus sadleri</i>	62.2	71.9	66.7	66.8
	<i>Mormyrus kannume</i>	75.6	79.5	77.5	77.5
	<i>Oreochromis leucostictus</i>	44.4	75.0	55.8	57.7
	<i>Synodontis victoriae</i>	80.0	66.7	72.7	73.0
XGB	<i>Bagrus docmak</i>	75.0	60.0	66.7	67.1
	<i>Brycinus sadleri</i>	53.7	90.6	67.4	69.8
	<i>Mormyrus kannume</i>	77.5	79.5	78.5	78.5
	<i>Oreochromis leucostictus</i>	50.0	31.3	38.5	39.5
	<i>Synodontis victoriae</i>	35.0	25.9	29.8	30.1
GBM	<i>Bagrus docmak</i>	84.6	55.0	66.7	68.2
	<i>Brycinus sadleri</i>	68.8	59.5	63.8	63.9
	<i>Mormyrus kannume</i>	75.8	73.5	74.6	74.6
	<i>Oreochromis leucostictus</i>	51.5	81.0	63.0	64.6
	<i>Synodontis victoriae</i>	35.0	36.8	35.9	35.9

Table 2.6 present the testing results after resampling (random over-sampling (ROS)) has been performed. We noticed a drastic change on the performance of metrics on the RF classifier. There is an increase in the predictive performance of the species, with *Brycinus sadleri* having a precision, recall, F1-score and G-mean of **100**, respectively and *Bagrus docmak* having a recall score of **100**. There is also an increase in the F1-score and G-mean scores of all the species. The RF classifier showed a good performance in predicting the relationship of multiple species and habitat suitability. We noticed that species that were dominant on the lake habitat showed a good performance after resampling under the RF classifier. The RF classifier has been used by Garzon et al. (2006) for predicting species habitat suitability and it was the most accurate algorithm of the three methods used.

The bagCART classifier showed an increase in the predictive performance of species. *Brycinus sadleri* and *Oreochromis leucostictus* species achieved precision scores of 100, respectively. *Bagrus docmak* and *Brycinus sadleri* species achieved a high recall of **100**. *Brycinus sadleri* species achieved a high F1-score of **100** and G-mean has increased from 70.1 to **100**. The RF and bagCART classifiers achieved high performance on F1-score and G-mean.

Mormyrus kannume species showed a good performance in all the classifiers. After resampling has been implemented, *Synodontis victoriae* species showed an increased performance to all the classifiers.

The *Synodontis victoriae* species showed poor predictive performance under the AdaBoost, XGB and GBM classifiers on the imbalanced data. *Bagrus docmak* and *Brycinus sadleri* species showed consistent results before and after resampling. These two species achieved high predictive performance on the imbalanced and balanced datasets. The reason might be these species were more dominant on the lake habitat during sampling, as shown in Table 2.4 with *Brycinus sadleri* species having a high number of instances. There are few studies that reviews the performance of SDMs and machine learning techniques. Beery et al. (2021) conducted a review of SDMs

for ML practitioners with the focus on species presence only and presence-absence datasets. SDMs are frequently applied to single species as they depend on binary data. Multiple species are meant to be modeled or predicted with ensemble ML algorithms since they can handle the multi-class imbalanced classification problem that comes with multiple species.

Table 2.6: Classifiers results using ROS technique.

Model	Species	Precision	Recall	F1-score	G-mean
RF	<i>Bagrus docmak</i>	90.9	100	95.2	95.3
	<i>Brycinus sadleri</i>	100	100	100	100
	<i>Mormyrus kannume</i>	86.7	97.4	89.2	89.5
	<i>Oreochromis leucostictus</i>	82.2	96.2	98.0	98.1
	<i>Synodontis victoriae</i>	95.0	70.4	80.9	81.8
bagCART	<i>Bagrus docmak</i>	90.9	100	95.2	95.3
	<i>Brycinus sadleri</i>	100	100	100	100
	<i>Mormyrus kannume</i>	85.4	92.1	88.6	88.7
	<i>Oreochromis leucostictus</i>	100	96.2	98.0	98.1
	<i>Synodontis victoriae</i>	87.5	77.8	82.4	82.5
AdaBoost	<i>Bagrus docmak</i>	100	40.0	57.1	63.2
	<i>Brycinus sadleri</i>	63.4	86.7	73.2	74.1
	<i>Mormyrus kannume</i>	73.3	86.8	79.5	79.8
	<i>Oreochromis leucostictus</i>	89.5	65.8	75.6	76.5
	<i>Synodontis victoriae</i>	54.5	44.4	50.0	49.2
XGB	<i>Bagrus docmak</i>	88.9	80.0	84.2	89.4
	<i>Brycinus sadleri</i>	66.7	73.3	69.8	70.0
	<i>Mormyrus kannume</i>	75.0	86.8	80.5	80.7
	<i>Oreochromis leucostictus</i>	65.6	80.8	72.4	81.9
	<i>Synodontis victoriae</i>	69.2	33.3	45.0	57.7
GBM	<i>Bagrus docmak</i>	100	80.0	88.9	89.4
	<i>Brycinus sadleri</i>	71.4	83.3	76.9	77.2
	<i>Mormyrus kannume</i>	78.0	84.2	81.0	81.1
	<i>Oreochromis leucostictus</i>	78.1	96.2	86.2	86.7
	<i>Synodontis victoriae</i>	73.3	40.7	52.4	54.7

Our results complement [Rustamov \(2022\)](#) findings on the comparison of different models in predicting habitat suitability of 8 rare *Tulipa* species in Uzbekistan. They find that RF and GBM were superior to other modeling techniques. [Rew et al. \(2021\)](#) used bagging ensemble with deep neural networks to construct the effectiveness of SDMs. Based on their results, it was confirmed that bagging techniques achieved higher predictive performance although the observation dataset was imbalanced. Our findings also confirmed that RF and bagCART classifiers give high predictive performance in comparison to other classifiers. RF and bagCART had the best predictive performance for *Brycinus sadleri* and *Bagrus docmak* species. XGB and GBM had the best performance for *Bagrus docmak* species and AdaBoost performed poor compared to other species except for *Mormyrus kannume* species. Across the classifiers, the performance of *Bagrus docmak* species was higher and the performance for *Synodontis victoriae* species was lower on imbalanced data and improved after resampling. *Synodontis victoriae* species achieved a poor performance than that for other species (Table 2.6).

The results obtained by bagging classifiers for predicting multiple species and habitat suitability are very encouraging, showing the significance of employing resampling techniques on multi-class imbalanced dataset. This study demonstrated that bagging classifiers outperform the boosting classifiers. The results suggested that the bagging classifiers were suitable techniques for predicting habitat suitability for multiple fish species of Uganda. The results obtained for the ensemble ML techniques were not surprising because techniques with greater flexibility allow regularization and proficient hyperparameter optimization approach to find better solutions. The ensemble ML techniques could constitute a good tool for decision-making in predicting the relationship of multiple species and habitat suitability within the framework of population ecology.

2.7 Conclusion

This study gives a review on the application of machine learning techniques for predicting species abundance and habitat suitability. The data based problems related to multi-class imbalanced classification and their resampling techniques were explored because multi-class imbalanced problem is regarded as one of the most challenging tasks to deal with in ecological studies. Numerous approaches for handling imbalanced problem have been proposed. The ensemble ML techniques have proven to handle the multi-class imbalanced classification problem successfully in combination with the resampling techniques. Several decomposition methods such as one-versus-one, one-versus-all and error-correcting output coding were discussed. This study does not recommend the decomposition process since it suffers from the loss of valuable information about the class relationships.

The predictive performance of classifiers were determined by looking at different evaluation metrics. It is an important task to evaluate the predictive performance of ML classifiers in order to find the best model for the given dataset. This study used various evaluation methods such as precision, recall, F1-score and G-mean to find the best predictive performing model. Two species, *Brycinus sadleri* and *Oreochromis leucostictus* were only dominant in the lake habitat, this might be due to the fact that small lakes are more suitable for species than larger rivers. Our findings showed that *Mormyrus kannume* species were abundant throughout the analysis. The predictive performance of this species has outperformed other species on different ML classifiers. The findings of this study showed that multiple species can be predicted with the implementation of ML algorithms in combination with the resampling techniques.

For future research purposes it is advisable to check the class imbalance ratio when dealing with multiple species. This could assist in determining the best predictive model for imbalanced dataset. As we know that ecological count datasets consists of so many aspects, for instance, species may be abundant in a certain sampling sites

while in other sampling sites only few species may be observed or detected. Based on the literature provided about species distribution models failing to give accurate results, this study recommend ensemble ML techniques. The belief is that this review will encourage ecology community to use ML expertise to species distribution models instead of using generalized linear models. Future studies need to improve species distribution models by implementing machine learning and deep learning techniques to address the species abundance and their environmental suitability.

Chapter 3

3. Count Regression and Machine Learning Techniques for Zero-Inflated Overdispersed Count Data: Application to Ecological Data

Abstract

The aim of this study is to investigate the overdispersion problem that is rampant in ecological count data. In order to explore this problem, we consider the most commonly used count regression models: the Poisson, the negative binomial (NB), the zero-inflated Poisson (ZIP) and the zero-inflated negative binomial (ZINB) models. The performance of these count regression models is compared with the four proposed machine learning (ML) regression techniques: random forests (RF), support vector machines (SVM), k -nearest neighbors (k NN) and artificial neural networks (ANN). The mean absolute error (MAE) was used to compare the performance of count regression models and ML regression models. The results suggest that ML regression models perform better compared to count regression models. The performance shown by ML regression techniques is a motivation for further research in improving methods and applications in ecological studies.

Keywords: Count data, ecology, machine learning, overdispersion, zero-inflation

3.1 Introduction

The aim of this chapter is to investigate the problem of overdispersion in ecological count data. Overdispersion is an existing and recurring problem that needs attention when dealing with ecological count data. Ignoring overdispersion will cause difficulties in analysis and decision making procedures of ecological studies. We approach this problem of overdispersion using machine learning (ML) regression techniques. To the best of our knowledge, an approach to overdispersion in ecological studies using ML techniques has not been extensively researched so far.

[Bolker et al. \(2009\)](#) define overdispersion as the occurrence of more variance in the data than predicted by a statistical model due to missing observations. The reason for the existence of missing observations in ecology data may be due to either structural errors (for instance a bird or fish is not present because the habitat is not suitable), observer error (species are present but cannot be detected) and design error (poor experimental design or sampling surveys are thought to be the reason) ([Zuur et al. 2009](#)). [Martin et al. \(2005\)](#) discuss the source of zeros in ecological count data and defined them as either ‘true zero counts’ or ‘false zero counts’. False zero counts occur when species are present at a site during the survey period, but the observer fails to detect them and true zero counts occur when species do not occur at a site because of the ecological process, that is, habitat unsuitability. Zero counts in ecology do not necessarily mean that there are no species detected during sampling survey ([Crisci et al. 2012](#)), rather it means that there has been no species at that particular sampling time (false zeros). The absence of species result in excess number of zeros, termed zero-inflation. The presence of zero-inflation in this study is caused by excess number of false zeros due to observer error in the course of survey.

Poisson regression model has been widely used to analyze count data under the assumption of equidispersion, that is, the mean of the response variable is equal to the variance of the response variable ([Harrison 2014](#)). However, as much this is a

naturally occurring property of the Poisson regression model, it is not always true in real life ecological count data as counts may exhibit excess variability. The fact that equidispersion is rarely found in real data has resulted in the development of more general count models which do not assume equidispersion (Hilbe 2011b). The negative binomial (NB) regression model has been used as an alternative model to the Poisson regression model (Lindén & Mäntyniemi 2011, Kassahun et al. 2014). NB regression models are more flexible than Poisson regression models even though they do not provide exact predictions in certain situations (Desjardins 2016). The next alternative used for modeling count data with excess number of zeros are the zero-inflated models such as zero-inflated Poisson (ZIP). ZIP has been applied in many areas of research such as insurance claims (Sakthivel & Rajitha 2017, Zamani & Ismail 2013), education (Desjardins 2016), healthcare (Gupta et al. 2015, He et al. 2019), animal ecology (Elliott & Morrell 2009) and transport (Lord & Mannering 2010). Minami et al. (2007) found ZIP model was not appropriate for data that are both zero-inflated and overdispersed. Furthermore, Minami et al. (2007) and Rose et al. (2006) proposed zero-inflated negative binomial (ZINB) model as another alternative to handle overdispersion. ZINB model was shown to be appropriate to some ecological situations even though the issue of overdispersion still remains (Minami et al. 2007).

From the empirical evidence provided above, the proposed methods still pose challenges in dealing with overdispersion. There are still numerous false zeros that are being observed or not accounted for. Thus there is still room to reduce false zeros in ecological count data. In other words, there is still room to improve to reduce overdispersion in count data. This chapter seeks to propose a possible method that we believe can reduce overdispersion. Most published studies in ecology often do not report on overdispersion in respect to their best fitting models (Richards 2008). Failing to account for overdispersion can lead to incorrect inferences (Hilbe 2011a). There is limited literature in ecological studies about how overdispersion affects the results as researchers would identify predictors as having biologically meaningful ef-

fects when in fact they do not (Harrison 2014). We turn to machine learning (ML) regression techniques to try and handle the problem of overdispersion in ecological count data. Lately, ML methods are cropping up in different areas of science. However, to the best of our knowledge, there is limited empirical evidence showing the use of ML regression techniques in estimating missing observations in population ecological studies. ML models make no distributional assumptions about the response or predictor variables unlike many statistical analysis methods. Some ML techniques accommodate zeros both in the response and in the predictor variables, that is what makes them unique and powerful alternatives to statistical methods. We test the proposed estimation techniques on real life fisheries count data set which usually has missing observations and make a comparison to the regression count models to assess if overdispersion can be reduced further.

The remainder of the chapter is organized as follows. Section 3.2 outlines the modeling approach relevant in this chapter. Section 3.3 presents the methods used in this chapter. In section 3.4 we present the numerical test of the proposed methodology on fisheries data set. Finally, section 3.5 concludes this chapter and gives recommendations for future work.

3.2 Modeling approach

In this section, we outline the modeling approach used in this study. The most commonly used count regression models are Poisson, NB, ZIP and ZINB. These models are chosen to assist in modeling zero-inflated overdispersed count data. Poisson regression model fail to model count data that is dominated by excess zeros. Thus, the NB regression model which accommodates overdispersion is used as an alternative model. Zero-inflated models such as ZIP and ZINB were recommended in situations where the data is both zero-inflated and overdispersed. ML regression techniques are proposed as alternative models to the traditional count regression models.

3.2.1 Count regression models

Suppose y_{ij} represents species count data for all sites $i = 1, \dots, n$; for all visits $j = 1, \dots, m$. Let y_{ij} follow a Poisson regression model with a conditional mean μ_{ij} and a set of explanatory variables (x_{ij}) . The Poisson regression model is expressed as:

$$P(Y = y_{ij} | x_{ij}) = \frac{\exp(-\mu_{ij}) \times \mu_{ij}^{y_{ij}}}{y_{ij}!},$$

where y_{ij} is a non-negative integer (Cheung 2002). We assume a log link function where $\mu_{ij} = \exp(\beta_0, \beta_1 x_{i1}, \dots, \beta_k x_{ik})$ is a linear combination of explanatory variables. By definition, the Poisson regression model cannot model overdispersed count data thus a gamma mixture Poisson is often assumed in this case. The negative binomial (NB) distribution is an alternative to the Poisson when data is overdispersed (Hilbe 2011b).

Suppose a random variable Y follows a Poisson distribution with a conditional mean μ_{ij} and the parameter λ which follows a gamma distribution with mean $E(\lambda) = \mu_{ij}$ and variance $\text{Var}(\lambda) = \mu_{ij}^2 \theta^{-1}$. Let ϕ denote a dispersion parameter. The joint density of Poisson and gamma distributions leads to the NB regression model:

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

where μ_{ij} denote the mean and the dispersion parameter $\phi = 1 + \mu_{ij} \theta$ should be greater than one. When Y follows a NB distribution, its expected value is $E(Y) = \mu_{ij}$ and the variance is $\text{Var}(Y) = \mu_{ij}(1 + \mu_{ij} \theta)$ (Lindén & Mäntyniemi 2011). When false zeros are too many to be modeled by a NB regression model, researchers consider the application of zero-inflated models such as zero-inflated Poisson (ZIP) and zero-

inflated negative binomial (ZINB) models which are believed to be adequate.

The zero-inflated models, ZIP or ZINB, are a mixture of two distributions, which models true zeros through the Poisson or NB distribution and a degenerate distribution at zero, which models the true zeros (Rose et al. 2006). The ZIP model has two components, π which represent 0 observations and $(1 - \pi)$ which represents an observed Poisson random variable (Lambert 2012). ZIP regression model for count i at visit j is defined by Rose et al. (2006) as:

$$P(Y = y_{ij}) = \begin{cases} \pi_{ij} + (1 - \pi_{ij}) \exp(-\mu_{ij}), & y_{ij} = 0 \\ (1 - \pi_{ij}) \frac{\exp(-\mu_{ij}) \mu_{ij}^{y_{ij}}}{y_{ij}!}, & y_{ij} > 0, \end{cases}$$

where $0 \leq \pi_{ij} < 1$ denotes the probability of zero occurrence and $\mu_{ij} > 0$ with the mean $E(Y_{ij}) = \mu_{ij}(1 - \pi_{ij})$ and the variance $\text{Var}(Y) = \mu_{ij}(1 - \pi_{ij})(1 + \pi_{ij}\mu_{ij})$. The dispersion parameter for the ZIP regression model is given by $\phi = \mu_{ij}\pi_{ij} + 1$. ZIP regression model provide a way to model zero-inflation due to excess number of zeros and ZINB regression model makes it possible to model both zero-inflation and overdispersion.

The ZINB regression model is defined by Rose et al. (2006) as:

$$P(Y = y_{ij}) = \begin{cases} P(Y = y_{ij}) = \pi_{ij} + (1 - \pi_{ij}) \left(\frac{1}{1 + \mu_{ij}\theta}\right)^{1/\theta}, & y_{ij} = 0, \\ (1 - \pi_{ij}) \frac{\Gamma(y_{ij} + \theta - 1)}{\Gamma(y_{ij} + 1)\Gamma(\theta - 1)} \left(\frac{1}{1 + \mu_{ij}\theta}\right)^{1/\theta} \left(\frac{\mu_{ij}\theta}{1 + \mu_{ij}\theta}\right)^{y_{ij}}, & y_{ij} > 0, \end{cases}$$

where $\theta > 0$ is a shape parameter which quantifies the amount of overdispersion, π_{ij} denotes the zero-inflation probability, y_{ij} denotes the species count and μ_{ij} the mean of count data. The mean and variance of the ZINB regression model are $E(Y) = \mu_{ij}(1 - \pi_{ij})$ and $\text{Var}(Y) = \mu_{ij}(1 - \pi_{ij})(1 + \pi_{ij}\mu_{ij} + \theta\mu_{ij})$. The dispersion of the ZINB regression model is given by $\phi = \mu_i\pi_{ij} + \theta\mu_{ij} + 1$.

3.2.2 Overdispersion for count data

Ecological count data with excess number of zeros exhibit overdispersion. Overdispersion is an indication that the variance is greater than the mean in the Poisson regression model. Kruppa & Hothorn (2020) and Zuur et al. (2009) define the occurrence of overdispersion if there is high variance in the data than predicted by the statistical model. In other words, when the dispersion statistic of the model is greater than one. Hilbe (2011b) suggested the Pearson-based dispersion statistic to be used to assess count model overdispersion. If the ratio of the estimator $\hat{\phi} = \frac{D}{n-p}$, where D denotes the deviance and $n - p$ denotes the degrees of freedom is greater than one, then there is evidence of overdispersion (Zuur et al. 2009). This chapter adopts this estimator as a check for overdispersion.

3.2.3 Machine learning regression techniques

Next, we discuss the machine learning (ML) regression techniques, considered in this study. The ML regression techniques are used to make predictions. In this study, our regression problem is of the form: $y_i = f(x_i) + \varepsilon_i$ in the presence of zero observations where $(\mathbf{X}, \mathbf{Y}) = \{x_i; y_i\}_i^N$ are data samples with y_i representing the target variable, x_i denoting explanatory variables and ε_i the error term.

The aim is to investigate how ML regression techniques can reduce overdispersion for a single species in ecological count data. ML techniques particularly in fisheries has not been fully applied. This is due to lack of collaboration between ML research community and natural scientists, a lack of communication about successful applications of ML in the natural science and difficulty in validating ML models (Olaya-Marín et al. 2013, Thessen 2016). ML techniques has many algorithms and methodologies which are capable in solving real-world problems.

Random forests

Random forests (RF) is a ML model that is designed to produce accurate predictions that do not overfit the data (Cutler et al. 2007). The RF model can be written in the form:

$$y(\mathbf{x}) = f \left\{ \sum_{k=1}^K w_k \phi(\mathbf{x}, \mathbf{v}_k) \right\},$$

where $\mathbf{v}_k$ put into code the choice of variable to split on and the function $f()$ depends on whether a regression or classification tree is needed. Each tree is developed using a subset of randomly chosen k features (Dastile et al. 2020).

Support vector machines

For a given data $D = \{(x_i, y_i)\}_{i=1}^p \in \mathbb{R}^n \times \{-1, +1\}$, the aim is to find a function $f(x) = y$ that correctly classifies the patterns of the data, where x_i denote a n -dimensional vector and y_i is its label. The aim of SVM is to create a hyperplane function that separate the observations into classes. The hyperplane can be defined as:

$$f(x) = (w \cdot x) + b,$$

where $w \in \mathbb{R}^n; b \in \mathbb{R}$ and the data is then linearly separable, if such a hyperplane exists (Yang et al. 2002). SVM also solve non-linear problems by mapping the input vectors to a higher dimensional space using kernel functions $k(x_i, x_j) = \{\mathcal{O}(x_i) \times \mathcal{O}(x_j)\}$ (Srinivasa et al. 2020). Then, the decision function can be written as:

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

K-nearest neighbors

The k-nearest neighbors (k NN) are determined by calculating the Euclidean distance between the input feature vector $\mathbf{x}$ and the data set (Srinivasa et al. 2020). The predictor $\hat{f}(x)$ is computed at point x when we first define a neighborhood $N_k(x)$ corresponding to the set of the k closest observations to x among learning sample in $\mathbb{R}^d$. The predictor is the average of the output over the k NN:

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

where $N_k(x)$ is the neighborhood of x defined by the k closest points i in the training sample (Crisci et al. 2012).

Artificial neural networks

Artificial neural networks (ANN) are a generalization of linear models inspired by analogies with the biological brain. The architecture of ANN is based on the Multi-Layer Perceptron (MLP). A perceptron is a simple unit which computes the following function:

$$h(\alpha) = f(w_i \cdot \alpha_i),$$

where the activation function f is a non-linear function of its argument. Examples are the function:

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

or the sigmoid function:

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

3.3 Materials and methods

This section discusses the methodologies used in this study and its components in detail as well as data preparation and processing. The performance evaluation measures used to assess the model performance in this study are also discussed.

3.3.1 Data description

The data set used here is from “Fish Species Occurrence Records for Uganda Mobilized Observation Archives” sourced from Uganda mobilized unpublished archives, available at <https://www.gbif.org>. This data set presents fish species occurrence records which were conducted at different periods in most of the aquatic ecosystems of Uganda. The data set was accessed in October 2020. The experiment is conducted on single species, *Lates niloticus*. In the data set, counts of *Lates niloticus* is utilized as the target variable. The fish were captured using fish gears such as gillnets, hooks and beach seines on specific sites of different waterbodies. The data set chosen consists of 424 observations with 3 variables. Table 3.1 presents descriptive statistics of the variables for *Lates niloticus* count data set. We can observe that the large portion of the count data is between 0 and 50 as depicted in Figure 3.1. Basically, Figure 3.1 shows that there are many zero observations, indicating overdispersion. Overdispersion in the *Lates niloticus* count data was determined by computing the dispersion of the fitted models as discussed in sub-section 3.2.2.

Table 3.1: Descriptive statistics for the variables.

Variables	Mean	Standard deviation	Median	Minimum	Maximum
Individual count	18.82	32.11	6.00	0.00	282.00
Decimal latitude	0.59	0.61	0.46	-0.65	1.85
Decimal longitude	28.11	11.87	33.13	0.00	34.02

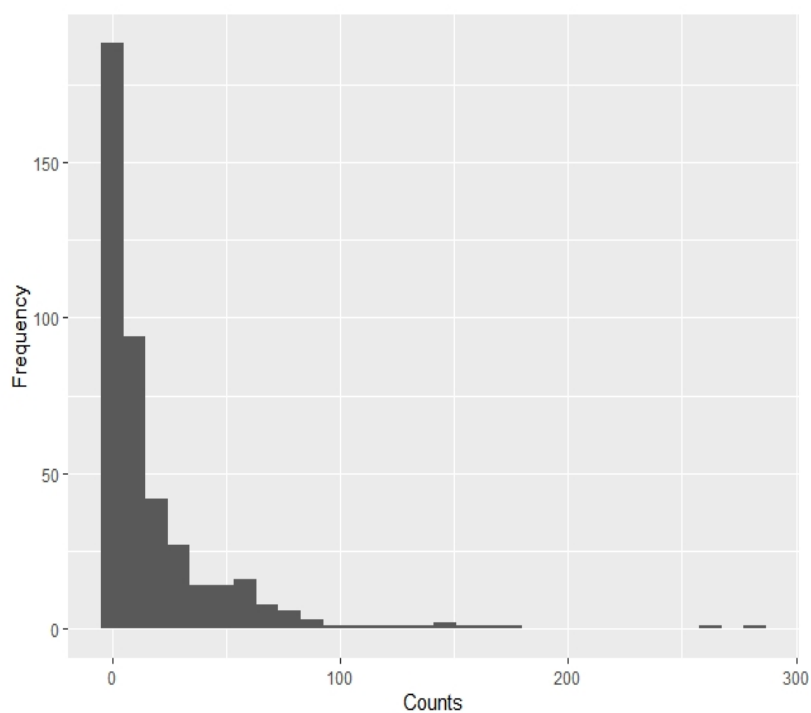


Figure 3.1: Graphical representation of count data.

Pre-processing of data set

Models were fit using species observation count data. Prior to model building, the data were scaled to avoid numerical stability in the fitted models. The results were generated using k-fold cross validation (CV) with the hyperparameter optimization performed on every iteration. The data intended for modeling was split into 70% training for building a predictive model and 30% testing for evaluating the model. Splitting of data helps to reducing the possibilities of overfitting on the training set. All the work was conducted using R version 3.6.2 (R Core Team 2022) and ML models were trained with caret package using models relevant functions.

3.3.2 Performance evaluation measures

Three common evaluation measures suitable for comparison of both count regression models and ML regression models are mean squared error (MSE), mean absolute error (MAE) or root mean squared error (RMSE). Table 3.2 presents the formulas for these metrics, where y_i and $\hat{y}_i$ are observed and predicted values, respectively. The best model will be selected based on the performance of different evaluation metrics. The best model is the one with the least MSE or RMSE or MAE value. In this study, we consider the MAE as our benchmark for assessing model performance for count regression and ML regression models. This prediction error was chosen because it is suitable for data set that has outliers (Buyrukoğlu et al. 2021).

Table 3.2: Mathematical formulas for performance metrics.

Metrics	Formulas
Mean squared error (MSE)	$\text{MSE} = \frac{1}{n} \sum_{i=0}^n (y_i - \hat{y}_i)^2$
Root mean squared error (RMSE)	$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=0}^n (y_i - \hat{y}_i)^2}$
Mean absolute error (MAE)	$\text{MAE} = \frac{1}{n} \sum_{i=0}^n y_i - \hat{y}_i $

3.4 Experimental results and discussions

In this section, we evaluate the performance of the proposed ML regression models in the presence of zero-inflation and overdispersion for *Lates niloticus* count data. We compare the performance of the proposed ML regression techniques with count regression models and presents the empirical insights/findings.

3.4.1 Count regression model results

Table 3.3 presents the results of the count regression models. The Poisson regression model was fitted in R using the *glm()* function. After fitting the Poisson regression model, we tested for overdispersion using the dispersion test function from *AER* package in R software. The Poisson regression model estimates the dispersion parameter as 47.718, which is an indication of overdispersion since it is greater than one. The dispersion parameter in the NB and ZINB models were estimated as 1.397 and 0.561, respectively. Since the dispersion parameter for ZINB regression model was 0.561, we observe that ZINB works better in comparison to the Poisson regression model when the data is both zero-inflated and overdispersed. All our count regression models exhibit overdispersion regardless whether it accommodates both zero-inflation and

Table 3.3: Results for count regression models.

Model	Dispersion	MAE
Poisson GLM	47.718	19.709
NB	1.397	19.651
ZINB	0.561	19.651

overdispersion. This study confirms that the presence of overdispersion has an impact on the performance of count regression models. The Poisson regression model is the most commonly used model, however the findings confirm that it is poor in dealing with either zero-inflation or overdispersion.

Comparison of the MAE results shows that the NB (MAE = 19.651) and ZINB (MAE = 19.651) had the lowest MAE values. In this case, ZIP had the highest MAE of 19.716. Amongst the count regression models, NB and ZINB performed better compared to Poisson and ZIP models. The reason is that of the extra dispersion parameter that accommodates overdispersion. The count regression model findings support [Minami et al. \(2007\)](#) and [Rose et al. \(2006\)](#) suggestion about ZINB model. The performance metrics of our count regression models improves that of [Buyrukoğlu et al. \(2021\)](#).

3.4.2 Machine learning regression models results

The RF, SVM, k NN and ANN were selected as ML regression models and their results are presented in Table 3.4. In the RF model, there are three tuning parameters of interest (node size, number of trees and number of explanatory variables sampled at each split). The focus in this study was on one tuning parameter which was the number of explanatory variables tried at each split. The number of splits (*mtry*) is

determined by the total number of explanatory variables in the data set. The RF model showed that $mtry = 2$ resulted in better MAE value of 17.525.

For the SVM model, the Gaussian radial basis function kernel was used since our count data was non-linear. This function relies only on one tuning parameter, which is the cost parameter (C). The cost parameter determines the possible misclassifications of the model. It essentially forces a penalty to the model for making an error. The higher the value of C , the less likely it is that the SVM will misclassify a point. The optimum SVM model with a smallest MAE value of 13.876 was selected.

For the k NN model, the number of neighbors (k) is the key parameter. Multiple k - values were used to determine the optimum model. The k NN model with $k = 7$ revealed an optimal model with a lower MAE value of 14.850.

For the ANN model, the performance was determined by the number of units within the hidden layers. ANN had the highest MAE value of 19.716. The highest performance of ANN was observed with three neuron in the hidden layer.

3.4.3 Model performance comparison

As stated in sub-section 3.3.2, MAE was chosen as a benchmark for model performance since it enables us to make comparison of count regression and ML regression models. Table 3.5 shows model comparison for both count regression and ML regression models arranged in ascending order for MAE. SVM had the least MAE whereas ANN has the highest MAE. The count regression models presented poor performance in reducing overdispersion. These models presented high MAE values. This is an indication that count regression models are not capable enough to model zero-inflated overdispersed count data.

The ML results showed that SVM outperforms all other ML regression techniques. The robustness of these models is an indication that ML regression techniques can

be used in future to predict species count data. In comparison to the count regression models, the ML regression models showed the best performance based on MAE values. This signifies that ML regression models have a capacity to reduce the overdispersion problem. ML techniques do not restrict the relationship between the mean and variance of the data, meaning that overdispersion will not cause a problem in ML techniques (Zhang et al. 2020). Hence ML techniques are suggested as good algorithms to model overdispersion in ecological count data. This fact concurs with the previously reported results by Cutler et al. (2007); Kampichler et al. (2010) and Olaya-Marín et al. (2013) which found that ML regression techniques worked better in comparison to count regression models in ecological count data. Olaya-Marín et al. (2013) recommended that the comparison of different ML regression techniques should be considered, as this would be helpful to interpret the goodness of the results.

We urge that individual researchers select techniques that are consistent with the specific problem, the nature of the questions being addressed and the availability of the data set. The use of ML techniques in ecological research is motivated by a range of research questions, the type of the available data and the expected outcomes of modeling. This makes ML techniques a very dynamic approach for predictive and exploratory modeling with many user-defined parameters to be considered for each objective (Ghannam & Techtmann 2021).

Table 3.4: Summary of results on ML regression models.

Model	Optimized parameters	MAE
RF	mtry = 2	17.525
SVM	C = 1	13.670
<i>k</i> NN	K = 3	14.850
ANN	size = 3	19.716

Table 3.5: Model comparison.

Model	MAE
SVM	13.670
k NN	14.850
RF	17.525
NB	19.651
ZINB	19.651
Poisson GLM	19.709
ANN	19.716

3.5 Conclusion

The main aim of this study was to propose possible methods that can reduce overdispersion in ecological count data. This study compared the performance of count regression models with ML regression techniques in order to evaluate their capabilities in reducing overdispersion that is rampant in ecological count data and assess their conformity in terms of performance metrics through the MAE values. For the traditional count regression models: Poisson GLM, NB, ZIP and ZINB were considered. ZINB performed better compared to other count regression models.

Four ML regression techniques were implemented: k NN, ANN, RF, SVM and their performance was compared to count regression models. The results showed that the SVM model performed better in reducing overdispersion in comparison to the other ML regression models. The findings of the comparison between traditional count regression models and ML regression models showed that ML regression models performed better based on the MAE considered in this study.

ML regression techniques appeared to be an attractive route towards tackling overdispersion problem, particularly in areas where a lack of knowledge exists regarding the implementation of effective models. The comparative study shows the potential of ML regression techniques to reduce overdispersion in ecological count data. The strong performance shown by ML regression techniques should motivate further research, resulting in the improvement of methods and applications in ecological studies. In future, the application of ML regression techniques in ecology will become an increasingly attractive tool for ecologists. Moreover, we conclude that the effect of overdispersion in ecological count data is largely dependent on the dispersion parameter of the fitted model.

Chapter 4

4. An approach to multi-class imbalanced problem in ecology using machine learning

Abstract

Ecologists collect their data manually by visiting multiple sampling sites. Since there can be multiple species in the multiple sampling sites, manually classifying them can be a daunting task. Much work in literature has focused mostly on statistical methods for classification of single species and very few studies on classification of multiple species. In addition to looking at multiple species, we noted that classification of multiple species result in multi-class imbalanced problem. This study proposes to use machine learning approach to classify multiple species in population ecology. In particular, bagging (random forests (RF) and bagging classification trees (bagCART)) and boosting (boosting classification trees (bootCART), gradient boosting machines (GBM) and adaptive boosting classification trees (AdaBoost)) classifiers were evaluated for their performances on imbalanced multiple fish species dataset. The recall and F1-score performance metrics were used to select the best classifier for the dataset. The bagging classifiers (RF and bagCART) achieved high performances on the imbalanced dataset while the boosting classifiers (bootCART, GBM and AdaBoost) achieved lower performances on the imbalanced dataset. We found that some machine learning classifiers were sensitive to imbalanced dataset hence they require data resampling to improve their performances. After resampling, the bagging classifiers (RF and bagCART) had high performances compared to boosting classifiers (bootCART, GBM and AdaBoost). The strong performances shown by bagging classifiers (RF and bagCART) suggest that they can be used for classifying multiple species in ecological studies.

Keywords: Ecology, ensemble classifiers, imbalanced dataset, machine learning, multi-class classification, multiple species

4.1 Introduction

In population ecology, classification of species from multiple sampling sites has been of great interest as it is used to study species presence or absence. The classification of various species can further give valuable insights about the health of the ecological system and can be used as a parameter for monitoring environmental changes. Ecologists collect the data manually (by visiting the multiple sampling sites numerous times) and since there can be many species in the multiple sampling sites, identifying them manually can be a daunting task. For instance, classifying fish species from count data in natural environment can be challenging because of variation in surrounding habitats. Species can be limited in a suitable habitat due to dispersal limitation and that does not always reflect the suitability of a particular habitat for a species. In some cases, individual species will dominate the habitat while in others multiple species will co-exist within the same habitat. Thus, the process of classifying multiple species is challenging and still an open problem. As a result, this study focuses on this issue of classifying multiple species from the provided datasets.

Much work in literature has focused mostly on statistical methods for classifying single species (Alldredge et al. 2007, Wright et al. 2020) and very few studies on classification of multiple species (Alldredge et al. 2007). Statistical methods that have been used to model single species includes poisson generalized linear models (Zhu 2012, Kassahun et al. 2014, He et al. 2019), negative binomial models (Minami et al. 2007, Ver Hoef & Boveng 2007, Lindén & Mäntyniemi 2011), zero-inflated poisson models (Ridout et al. 2001, Lambert 2012, Wagh & Kamalja 2018) and zero-inflated negative binomial models (Agarwal et al. 2002, Wenger & Freeman 2008) as well as some machine learning models such as random forests (Cutler et al. 2007) and support

vector machines (Crisci et al. 2012). Although many methods have been developed for models of individual species, ecologists are recognizing the importance of predicting multiple species simultaneously using multi-label classification (Jones et al. 2011). However, to date, multi-label classification has been applied primarily to problems in text and image annotation (Jones et al. 2011). Similar machine learning techniques have been applied in other research fields, such as image-based insect species and gender classification (Tuda & Luna-Maldonado 2020) and biometric-based approach to identify fish species (Tharwat et al. 2018). Statistical methods that have been used for the classification of more than two species (multiple species) include linear discriminant analysis (LDA) (Hauser-Davis et al. 2010, Bourel & Segura 2018) and multinomial logistic regression (MLR) (Bourel & Segura 2018). The limitations of LDA and MLR include the nature of the dataset (balanced versus imbalanced) is not often reported (Bourel & Segura 2018). Thus, most statistical classification methods encounter the multi-class imbalanced problem.

The multi-class imbalanced problem is a subset of supervised machine learning tasks where the classification variable of interest consists of more than three unequal sample sizes (Wang & Yao 2012). Let X denote a predictor variable and Y be a class set of size k greater than two. When $k > 2$, the classification is said to be multi-class classification (Okun et al. 2011) and when each class outnumbers other classes by a large portion is said to be imbalanced problem. Multi-class imbalanced classification problem occurs in ecological count of species presence or absence as in other practical fields, such as medicine (Goksuluk et al. 2019), human communication (Sokolova & Lapalme 2009) and bio-informatics (Saito & Rehmsmeier 2015). Multi-class imbalanced classification has presented a critical classification problem since the rise in the applications of machine learning techniques. This study looks at the main approaches for dealing with multi-class imbalanced classification problem that include the following: 1) the resampling approaches (Shin et al. 2021) and ensemble methods (Shin et al. 2021). Resampling approaches are one of the pre-processing techniques that are conducted before model building which adjust imbalanced data to balance a skewed

class distribution. Resampling is carried out by 1) under-sampling majority instances which may lose potential useful information. 2) Over-sampling minority instances which increases the size of the training dataset. Numerous techniques for multi-class imbalanced classification such as ensemble methods have been proposed in the literature (Okun et al. 2011). Ensemble methods include multiple single classifiers which are adjusted to improve the performance of classifiers on imbalanced dataset using various methods, such as bagging and boosting (Shin et al. 2021, Galar et al. 2011). In this study, the resampling and ensemble methods, which are the recommended techniques for tackling multi-class imbalanced are adopted for improving classification of multiple species. Although the multi-class imbalanced problem is a growing research subject in machine learning communities, it has been explicitly addressed by only few studies in relevant application fields, including population ecology (Shin et al. 2021) and conversation biology (Kampichler et al. 2010). Classifying multiple species based on their biological classes constitutes an example of multi-class classification in the presence of class imbalance. Furthermore, the multi-class imbalanced on the performance of ensemble (bagging and boosting) classifiers were assessed on four fish species collected at three major lakes of Uganda. Therefore, the study sites provide a useful case study for the analysis of multi-class imbalanced classification.

The objectives of this study were to 1) classify multiple species based on classes using machine learning techniques, 2) select the best performing classifier for classifying multiple species with imbalanced and balanced dataset, and 3) examine how various evaluation metrics performs differently on the ensemble classifiers. The potential gain of proposing machine learning methods for multiple species includes high level of ecological interpretability (Crisci et al. 2012). The advantages of machine learning methods are: the ability to handle large datasets, easy to interpret the decision rules and high learning capability from the input datasets. Hence, it is believed that the applications of machine learning classifiers will give the better performance in classifying multiple species. Matthews et al. (2018) has applied classification techniques to explore the performance of machine learning techniques on fossil teeth in the

family Bovidae dataset. [Kampichler et al. \(2010\)](#) also used machine learning classification techniques in biological conversation context to estimate ocellated turkey (OT) abundance and compared the performance of five machine learning techniques. Machine learning classifiers are mostly used in medical applications ([Zararsiz et al. 2014](#), [Goksuluk et al. 2019](#)), bio-statistics ([Goksuluk et al. 2019](#)) and in bio-informatics ([Srinivasa et al. 2020](#)). Various types of machine learning classifiers including support vector machines (SVM), random forests (RF), boosted regression trees (BRT) and Naive Bayes (NVB) have been applied to other research fields such as insurance ([Spedicato et al. 2018](#)), credit scoring ([Dastile et al. 2020](#)) and computer engineering ([Kotsiantis et al. 2007](#), [Ng 2019](#)). [Cutler et al. \(2007\)](#) and [Crisci et al. \(2012\)](#) have used machine learning classifiers in ecology to compare the performance of classification techniques for multiple species. [Cutler et al. \(2007\)](#) focused only on RF classifier whereas [Crisci et al. \(2012\)](#) compared different machine learning methods and found that boosting and SVM classifiers have higher performances. Furthermore, [Kampichler et al. \(2010\)](#) compared different machine learning classifiers and recommend the application of RF in conversation biology. It was reported in the ecological literature that applications of machine learning classification techniques are not yet fully investigated. This study further support the previous conclusion of [Shin et al. \(2021\)](#), that future studies should consider multi-class imbalanced classification as it is mostly remaining unsolved problem.

The rest of the study is organized as follows: Section [4.2](#) gives an outline of machine learning classification techniques used in this study. Section [4.3](#) discusses data preparation and performance evaluation measures for the classifiers. Section [4.4](#) presents the experimental results and Section [4.5](#) discusses the findings. Finally, Section [4.6](#) concludes this study and suggest future work.

4.2 Machine learning classification techniques

This section presents literature on the machine learning (ML) classifiers to be used in this study. The focus will be on supervised ML classifiers which will be briefly described in this section.

Suppose m -dimensional vector of measurements is denoted by $\mathbf{X}$, where each dimension is identified as x_i such that $i = 1, \dots, m$. Let $\mathbf{Y}$ be a singleton element from the set G , that contains k elements distinguished as $g_1, g_2, \dots, g_k$. The combination of $(\mathbf{X}, \mathbf{Y})$ form a complete dataset. A n -dimension collection of these datasets, $(\mathbf{X}, \mathbf{Y})^n$, form the complete dataset from which classification models are trained (Muhamedyev 2015).

A brief review of the different ML classification techniques follow below. These techniques include bagging (random forest and bagging classification trees) and boosting (boosting classification trees, gradient boosting machines and adaptive boosting classification trees) classifiers. Bagging and boosting classification trees are powerful ML classifiers that are well established in other research fields but relatively not fully applied in ecological studies (Thessen 2016). The list of these techniques for ML is not complete, the literature provides numerous techniques that are used to model count data. However, this study only focuses on five techniques as it would be almost impossible to look at all ML classification techniques. These ML techniques are selected on the basis of their high performance compared to other ML algorithms. These algorithms have been enormously applied and performed well in natural resource and hazard mapping such as groundwater potentiality (Mosavi et al. 2021), landslide susceptibility (Naghibi et al. 2016) and classification of microbiomic data (Statnikov et al. 2013). However, fewer studies in fisheries research on comparing the bagging and boosting techniques for classifying multiple species to ascertain the performance of these classifiers.

4.2.1 Random forests

Random forests (RF) consists of a collection of classification trees (James et al. 2013) where each tree is built from a bootstrapped sample of the original dataset (Kampichler et al. 2010). Using the training dataset and independent random vector, θ_k , a tree can be grown to construct a classifier $h(x, \theta_k)$, where x is an input vector (Jabeen et al. 2018). By definition, a RF is an ensemble classifier that consists of several tree structured classifier as:

$$h(x, \theta_k), k = 1, \dots, M,$$

where each tree plays a role in determining the most appropriate class at input x (Desjardins 2016).

RF consists of an ensemble of independent classification trees and a set of random samples (Mosavi et al. 2021). RF algorithm requires two tuning parameters, number of trees (ntrees) and number of variables in each split (mtry). The optimal number of variables sampled at each split is two. RF classifiers train several uncorrelated decision trees in parallel with bootstrapping followed by aggregation, which is jointly referred to as bagging. Each bagging tree makes use of around two-thirds of the observations. The remaining one-third of the observations not used to fit a given bagging trees are referred to as the out-of-bag (OOB) observations (James et al. 2013). The main advantage of this technique is that it can handle different types of datasets such as count and image data.

4.2.2 Bagging classification trees

Bagging classification trees (bagCART) has been applied frequently in fields such as bio-statistics (Kotsiantis et al. 2007) and remote sensing (Thessen 2016) but it has not been often applied in the field of ecology (Crisci et al. 2012). As the name suggest, bagCART introduces a random component into the tree building process

by building many trees on bootstrapped copies of the training data (James et al. 2013). Each bootstrap sample is obtained by randomly sampling training instances, with replacement, from the original training set, until an equal number of instances as in the training set is obtained. Advantages of bagCART compared to other ML classifiers include; 1) to reduce high variance, 2) to minimize overfitting, 3) to improve the stability and accuracy of regression and classification trees and 4) to combine single classifiers which are as precise and different as possible. Thus, it is expected that through application of bagCART in classifying multiple species, promising results can be achieved.

4.2.3 Gradient boosting machines

Consider a response variable y_i and a set of explanatory variables $\mathbf{x} = \{x_1, x_2, \dots, x_n\}$. Given a training sample $\{y_i, \mathbf{x}_i\}_1^N$ of known $(y, \mathbf{x})$ values, the expected value of some specified loss function $\Psi(y, F(\mathbf{x}))$ is minimized:

$$F^*(\mathbf{x}) = \arg \min_{F(\mathbf{x})} E_{y, \mathbf{x}} [\Psi(y, F(\mathbf{x}))].$$

Boosting approximates $F^*(\mathbf{x})$ by an additive expansion of the form:

$$F(\mathbf{x}) = \sum_{m=0}^M \beta_m h(\mathbf{x}, \mathbf{a}),$$

where the function $h(\mathbf{x}, \mathbf{a})$ is usually chosen to be simple functions of $\mathbf{x}$ with parameters $\mathbf{a} = \{a_1, a_2, \dots, a_m\}$. The expansion coefficients $\{\beta_m\}_0^M$ and the parameter $\{\mathbf{a}\}_0^M$ are jointly fit to the training data in a forward manner (Friedman 2002).

Gradient boosting machines (GBM) can do both regression and classification tasks. GBM model combines the bagging and boosting techniques to optimize the better predictive performance (Jiao & Du 2016). This model uses a small portion of dataset

at each step of boosting process rather than the whole dataset. The GBM model includes the following tuning parameters namely interaction depth (number of trees), shrinkage rate (learning speed), bagging fraction (fraction of training data) and training function (loss function). A GBM model with a small shrinkage is expected to always give better results. The boosting technique uses a small number of trees at each stage of classification so that it can avoid the overfitting issue (Moisen et al. 2006).

4.2.4 Boosting classification trees

Boosting is used to overcome the inaccuracies inherent in a single tree model (Trevor et al. 2009) by computing a sequence of simple classification trees, where each successive tree is built for the prediction residuals of the preceding tree. Boosting classification trees (bootCART) works in a similar way to bagging, except that the trees are grown sequentially. The common goal of implementing boosting is to improve the accuracy of a classifier combining single classifiers which are slightly better than random guessing (Alfaro et al. 2013).

Boosting is also stochastic, which improves the predictive performance of the overall model. In order to build a bootCART, three key parameters must be defined namely; bag fraction, tree complexity and learning rate. The bag fraction controls the proportion of the dataset to be sampled each time an individual tree is fitted and added to the model during boosting. The tree complexity determines the maximum depth of the simple trees. This means that it also controls whether or not interactions are fitted as well as the maximum order of interactions permitted to be modeled (Plant et al. 2021).

4.2.5 Adaptive boosting classification trees

The adaptive boosting classification trees (AdaBoost) technique takes input training set $\{(x_1, y_1), \dots, (x_m, y_m)\}$ where each x_i is an n -tuple of attribute values belonging to a certain domain or instance space X and y_i is a label set $Y = \{-1, +1\}$ in the context of multi-class applications (Sun et al. 2007, 2006). AdaBoost is used to improve predictive performances of the classification-based machine learning techniques (Sun et al. 2007). AdaBoost is one of the ensemble classifiers, that utilizes boosting to obtain robust classifiers by combining the weak classifiers through a weighted linear combination.

AdaBoost uses a set of serially connected weak classifiers, where each weak classifier tries to improve the classification of the samples that were misclassified by the previous weak classifier. This iterative process is repeated until the maximum limit of the number of classifiers is built, or the error function does not change (Achu et al. 2021, Gaber et al. 2016). The goal of this ensemble classifier is to train its single classifiers and then combine the weighted single classifiers to construct a strong classifier (Gaber et al. 2016). AdaBoost learning algorithm is made up of two phases, namely, training and testing phases. In the training phase, the weights of all samples are equal. For each iteration, the weights are adjusted and the samples are selected based on their weights to train the current single classifier and the error rate or misclassification rate for classifier is then calculated. If misclassification rate is greater than 50, the weights are re-initialized, the error rate is calculated again (Gaber et al. 2016). AdaBoost classifier focuses on the misclassified samples and the procedure is repeated for many iterations until the performance is satisfied.

4.3 Materials and methods

This section discusses the settings of the experiments, data pre-processing and performance measures used for model selection. All the work was conducted using R version 3.6.2 [R Core Team \(2022\)](#) software where all models were trained with caret package.

4.3.1 Data description

The experiments are conducted on four fish species obtained from the sampling sites of Uganda. The four fish species are: *Lates niloticus* (Latidae), *Oreochromis niloticus* (Cichlidae), *Clarias gariepinus* (Clariidae) and *Synodontis afrofisheri* (Mochokidae). These four fish species were dominant during the sampling.

Figure 4.1 depicts the map of lakes in Uganda where sampling surveys were conducted. The fishing locations include Victoria Nile, Lake Kyoga and Lake Victoria, as demonstrated in Figure 4.1. As can be observed in Figure 4.1 that Lake Victoria is the biggest across all our sampling sites. The dataset used in this study consists of 983 observations with 4 variables in Table 4.1. Out of 983 observations, 424 were identified as *Lates niloticus*, 213 were *Oreochromis niloticus*, 202 were *Synodontis afrofisheri* and 144 were *Clarias gariepinus* species. The dataset was normalized for each fitted model to avoid overfitting. We can see unequal number of observations among these species as shown in Table 4.1. Unequal number of observations in multiple species results to multi-class imbalanced data problem.

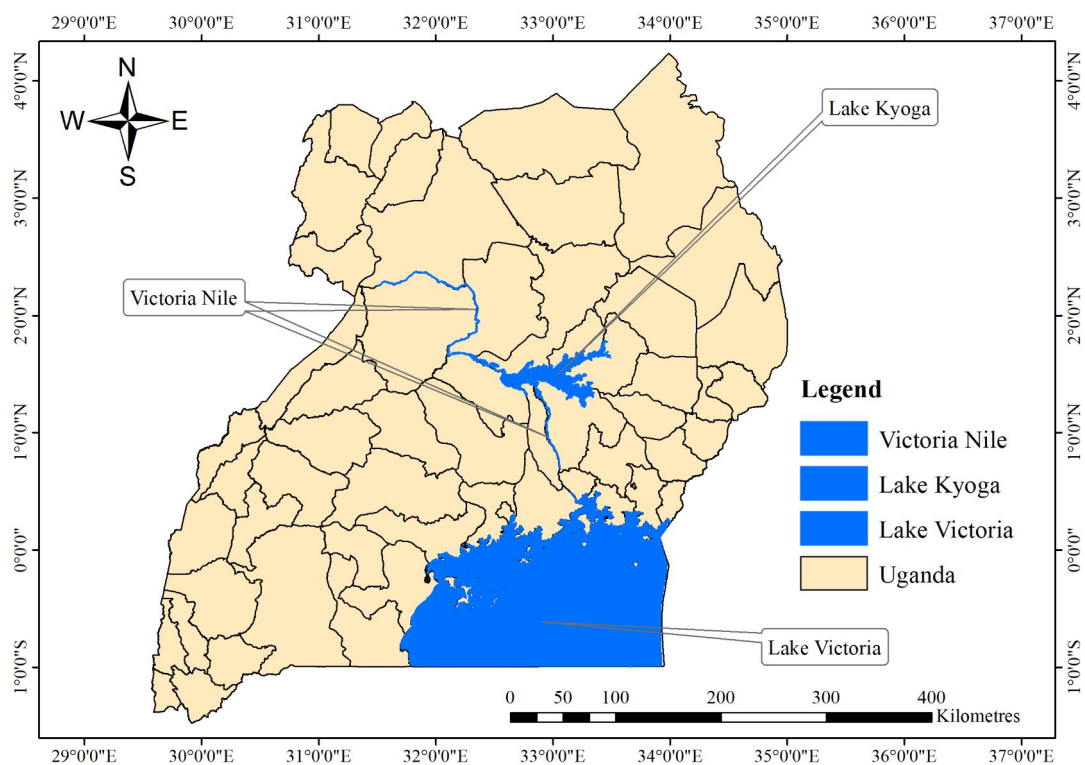


Figure 4.1: A map showing sampling sites in Uganda.

Table 4.1: Multiple species count data.

Species	Sample size (Counts)
<i>Clarias gariepinus</i>	144
<i>Lates niloticus</i>	424
<i>Oreochromis niloticus</i>	213
<i>Synodontis afrofishcheri</i>	202

4.3.2 Multi-class imbalanced data

A dataset is called multi-class imbalanced when the classification variable consists of three or more unequal number of classes (Lango 2019). The multi-class imbalanced problem has been a challenge in ML research in recent years (Rendón et al. 2020). The class with more observations is typically denoted as the majority class, whereas the under-represented class is called the minority class (Lango 2019). Support y_{ij} is the species count data for $i = 1, \dots, n; j = 1, \dots, m$ that belongs to class c ($c = 0$ or 1). Let $k_c = \frac{n_c}{n}$ be the proportion of observations from class c , n_c represent the number of observations in class c and n be the sample size.

Most classifiers are designed for two-class classification. Here we adopt a one-versus-all method for each class against the rest of the dataset to balance the multi-class problem. This method involves the splitting of multi-class dataset into multiple binary classification. One-versus-all method trains a binary classifier for each single class and the classifier regards all instances from other classes as negative instances (Zhang et al. 2016). Basically, one-versus-all trains a single classifier for each class, considering the current class as the minority one and the remaining classes as a majority one. Using the combination of resampling techniques and ensemble methods can solve the multi-class imbalanced classification problem.

To deal with multi-class imbalanced data problem, the resampling techniques like random over-sampling (ROS), random under-sampling (RUS), synthetic minority over-sampling techniques (SMOTE) and random over-sampling examples (ROSE) have been developed in the literature. ROS technique replicate samples from the minority class whereas RUS eliminate samples from the majority class. SMOTE is a statistical technique that increases the number of minority samples in the dataset by generating new instances. For a given minority example x , SMOTE and the examples k -nearest minority neighbors and randomly selects one of them y (Lango 2019). Then, it generate an artificial example which belongs to the minority class with features given

by

$$x_{new} = x + r.(y - x)$$

where r denote a randomly generated number from zero to one. There are no definitive rules in the literature about how exactly the multi-class imbalanced is a problem or how to address it (Thessen 2016). Therefore, the multi-class imbalanced problem in this study is tackled by ROS technique in combination with ensemble learning algorithms. Although ROS increases the quantity of minority class examples to make every class have the same sample size, it stabilizes the performance of classifiers.

4.3.3 Performance evaluation measures for classifiers

To evaluate the ability of ML classifiers, we compare the performance metrics of each model. The performance metrics were achieved by constructing the confusion matrix. A two-class confusion matrix provided in Table 4.2 demonstrates the cross-classification of the predicted outcome against the actual outcome.

In the ecological perspective, the true negative (TN) refers to the number of species absence who are correctly identified as non-detectable, false positive (FP) refers to the number of species presence which are incorrectly identified as detected, false negative (FN) refers to the number of detected species which are incorrectly classified as non-

Table 4.2: Confusion matrix in binary classification.

		Predicted	
		Positive	Negative
Actual	Positive	TP	FN
	Negative	FP	TN

detected and lastly, true positive (TP) is referred to the number of species presence which are correctly classified as detected. When performance metrics shows higher values of TP and lower values of FP, they indicate a better predictive performance.

The performance of ML classifiers were evaluated based on precision, recall and F1-score criteria. The mathematical expression of recall, precision and F1-score criteria are given in equation (4.3.1), (4.3.2) and (4.3.3), simultaneously. The metric sensitivity (also known as recall) is defined as the probability that a sample is correctly classified, given that it is true event sample. Recall is regarded as a useful metric if the positive class is of particular interest as it specifically measures how well the classifier performs on the positive class (Sooknunan et al. 2021):

$$Recall = \frac{TP}{TP + FN}. \quad (4.3.1)$$

Precision is the percentage of true positives, TP, among all examples that the classifier has labeled as positive:

$$Precision = \frac{TP}{TP + FP}. \quad (4.3.2)$$

The F1-score is the weighted average of the precision and recall:

$$F1 - score = 2 \times \frac{(precision \times recall)}{(precision + recall)}. \quad (4.3.3)$$

This metric is particularly useful for both imbalanced and balanced datasets as it takes into account both the FP and FN making it a much more reliable measure. A higher value of F1-score indicates a high classification performance. For multi-class imbalanced dataset, the performance measures were calculated by a combination of resampling and ensemble learning.

The performance metrics precision, recall and F1-score were used to select the best classifier for the dataset. A comparison of ML classification algorithms were per-

formed to select the best suitable classifier for this study.

4.3.4 Model building

The subset of data intended for the training was split into 80% training and 20% testing sets. All the predictive model building processes are performed on training sets and model performance are evaluated on testing sets. A 5-fold cross-validation was used in the training set to evaluate the performance of each classifier with the hyperparameter optimization performed in every iteration and resampling was done without replacement. The parameters of each classifier are optimized to avoid overfitting as suggested by [Zhang et al. \(2020\)](#).

4.4 Experimental results

This section presents performance results of machine learning classifiers. The performances of the classifiers were measured by precision, recall and F1-score. The proposed multiple species classification models (RF, bagCART, bootCART, GBM and AdaBoost) were evaluated on 983 observations collected from four different fish species that are mentioned in Section [4.3.1](#). The results presented here were obtained on the test dataset.

4.4.1 Classifiers results on imbalanced dataset

In this sub-section, we present the results of machine learning classifiers on the original imbalanced dataset. The results for each classifier is presented in Table [4.3](#) in percentage terms.

Table 4.3: ML classifier results on imbalanced dataset.

Model	Species	Precision	Recall	F1-score
RF	<i>Clarias gariepinus</i>	91.4	91.4	91.4
	<i>Lates niloticus</i>	93.3	96.5	94.8
	<i>Oreochromis niloticus</i>	92.5	100	96.1
	<i>Synodontis afrofisheri</i>	100	83.8	91.2
bagCART	<i>Clarias gariepinus</i>	100	85.0	91.9
	<i>Lates niloticus</i>	91.2	97.9	94.4
	<i>Oreochromis niloticus</i>	92.1	94.6	93.3
	<i>Synodontis afrofisheri</i>	97.4	86.0	91.4
bootCART	<i>Clarias gariepinus</i>	100	20.0	33.3
	<i>Lates niloticus</i>	65.6	88.4	75.3
	<i>Oreochromis niloticus</i>	82.2	100	90.2
	<i>Synodontis afrofisheri</i>	66.7	27.9	39.3
GBM	<i>Clarias gariepinus</i>	100	80.0	88.9
	<i>Lates niloticus</i>	86.7	95.8	91.0
	<i>Oreochromis niloticus</i>	92.1	94.6	93.3
	<i>Synodontis afrofisheri</i>	91.7	76.7	83.5
AdaBoost	<i>Clarias gariepinus</i>	100	20.0	33.3
	<i>Lates niloticus</i>	63.4	96.8	76.7
	<i>Oreochromis niloticus</i>	86.0	100	92.5
	<i>Synodontis afrofisheri</i>	100	7.0	13.0

The RF classifier achieved a high recall of 100 and F1-score of 96.1 when classifying the *Oreochromis niloticus* species. The bagCART achieved a high recall of 97.9 and F1-score of 94.4, when classifying the *Lates niloticus* species. Thus, the bagging classifiers (RF and bagCART) achieved high performances on the imbalanced dataset.

As for the boosting classifiers (bootCART, GBM and AdaBoost) the highest recall was obtained by the AdaBoost with a recall of 100 and F1-score of 92.5 when classifying the *Oreochromis niloticus* species. The GBM also performed well with a recall of 95.8 and F1-score of 91.0 when classifying the *Lates niloticus* species. However, the boosting classifiers achieved lower performance in comparison to bagging classifiers. The bootCART achieved lower recall score of 20.0 and lower F1-score of 33.3 when classifying the *Clarias gariepinus* species. The AdaBoost classifier achieved lower recall of 7.0 and lower F1-score of 13.0 when classifying the *Synodontis afrofisheri* species.

Of significant note is that both the bagging and boosting classifiers achieved a high precision scores when classifying the *Clarias gariepinus* and *Synodontis afrofisheri* species. The high precision scores indicate that the classifiers did not miss any TP instances on the imbalanced dataset. However, these classifiers achieved low recall scores which is an indication of higher chances of FN in the imbalanced dataset. Among the species, the *Oreochromis niloticus* showed the best recall and F1-score performances on the boosting classifiers and also for RF classifier. Thus, it is regarded as a dominant species among the four species.

4.4.2 Classifiers results on the balanced dataset

In this sub-section, we present results of machine learning classifiers on the balanced dataset. The results for each classifier is presented in Table 4.4 in percentage terms.

Table 4.4: ML classifier results on the balanced dataset.

Model	Species	Precision	Recall	F1-score
RF	<i>Clarias gariepinus</i>	100	91.4	95.5
	<i>Lates niloticus</i>	91.3	94.8	92.9
	<i>Oreochromis niloticus</i>	87.2	100	93.2
	<i>Synodontis afrofisheri</i>	100	85.7	92.3
bagCART	<i>Clarias gariepinus</i>	96.9	88.6	92.5
	<i>Lates niloticus</i>	94.3	96.5	95.4
	<i>Oreochromis niloticus</i>	92.5	100	96.1
	<i>Synodontis afrofisheri</i>	94.3	89.2	91.7
bootCART	<i>Clarias gariepinus</i>	94.7	69.2	80.0
	<i>Lates niloticus</i>	78.8	89.7	83.9
	<i>Oreochromis niloticus</i>	90.6	100	95.0
	<i>Synodontis afrofisheri</i>	87.5	61.8	72.4
GBM	<i>Clarias gariepinus</i>	80.9	65.4	72.3
	<i>Lates niloticus</i>	76.0	83.9	79.8
	<i>Oreochromis niloticus</i>	87.3	100	93.2
	<i>Synodontis afrofisheri</i>	73.9	50.0	59.6
AdaBoost	<i>Clarias gariepinus</i>	100	10.0	18.2
	<i>Lates niloticus</i>	69.4	95.9	80.5
	<i>Oreochromis niloticus</i>	97.6	95.2	96.4
	<i>Synodontis afrofisheri</i>	66.7	33.3	44.4

The RF classifier achieved a high recall of 100 and F1-score of 93.2 when classifying the *Oreochromis niloticus* species. The bagCART achieved a high recall of 100 and F1-score of 96.1 when classifying the *Oreochromis niloticus* species. The bagging classifiers demonstrated a high performance on the balanced dataset.

As for the boosting classifiers (bootCART, GBM and AdaBoost) the highest recall was obtained by the bootCART with a recall of 100 and F1-score of 95.0 when classifying *Oreochromis niloticus* species. The GBM also achieved a high recall of 100 and F1-score of 93.2 when classifying *Oreochromis niloticus* species. However, AdaBoost achieved lower performance with a recall of 10.0 and F1-score of 18.2 when classifying the *Clarias gariepinus* species.

The recall showed an excellent performance score of 100 for the RF, bagCART, bootCART and GBM classifiers when classifying the *Oreochromis niloticus* species. The higher recall scores by machine learning classifiers indicate smaller chances of *FN* which could be the number of detected species which are incorrectly classified as non-detected. In other words, the higher recall scores represent the proportion of species present that were correctly classified by each classifier. Interestingly, the F1-score results were also relatively high for RF, bagCART and bootCART classifiers on the balanced dataset. We have seen even after resampling, the *Oreochromis niloticus* species was dominating.

4.5 Discussion

The experiments were done on the multiple species dataset so as to understand the performances of machine learning classifiers. Initially the dataset was imbalanced. The results showed that bagging classifiers (RF and bagCART) had higher performances on the imbalanced dataset. The imbalanced dataset makes it difficult to obtain high classification results in many research fields that are facing data prob-

lems such as in bio-informatics, medical diagnosis and fisheries. The bootCART and AdaBoost classifiers presented poor results on the imbalanced dataset. In contrast to Tuda & Luna-Maldonado (2020) findings, AdaBoost performed well to classify species sexing of beetle insects. The GBM is the only boosting classifier that presented very good performance on the imbalanced dataset. We found that some of machine learning classifiers are sensitive to imbalanced dataset hence they require data resampling (e.g random-over sampling) to improve their performances.

This study was designed to explicitly explore how multi-class imbalanced affect the performance of machine learning classifiers when classifying multiple species. The results of this study suggest that the class imbalance had a significant influence on the performance of classifiers. In particular, an increase in the class imbalance resulted in an increase in the FN rate. Therefore, the effects of imbalanced dataset on classifiers performances differed depending on how the FN rate was reflected in different performance metrics. The application of resampling technique resulted in an increase in the TP rate in exchange for a decrease in the FN rate, this generally improves the performance of classifiers. Moreover, the increase in F1-score of boot-CART classifier indicates that the balance between TP and TN rates improved as a result of resampling. Furthermore, the positive effects of resampling on recall and F1-score increases level of adjustment in the class distribution. The application of resampling using ROS made a marginal difference in the F1-score of classifiers.

Our study is similar to that of Kampichler et al. (2010) where five ML techniques (classification trees, RF, ANN, SVM and automatically induced rule-based fuzzy models) were used to classify bird species. They recommended the combined use of classification trees and random forests due to their easy interpretability of classifiers and the high comprehensibility of the methods. Similar study, Tuda & Luna-Maldonado (2020) classify images of mixed sex specimens irrespective of classification target (sexes or species), by training supervised machine learning models: logistic model trees (LMT), RF, support vector machine (SVM), simple logistic regression, multilayer perceptron and AdaBoost (adaptive boosting). AdaBoost has successfully

classified insect species Tuda & Luna-Maldonado (2020) and Tharwat et al. (2018) used AdaBoost classifier to identify fish species. To validate the performance of AdaBoost, Tharwat et al. (2018) compared it with NB, kNN and MLP and AdaBoost achieved excellent results. Mosavi et al. (2021) used bagging (RF and bagCART) and boosting (AdaBoost and boosted generalized additive model (GamBoost)) classifiers for modeling ground water and the results indicated that bagging classifiers outperformed the boosting classifiers.

The results after resampling showed that the bagging classifiers had a high performances on the balanced dataset. A comparison of the classifiers' performances on the balanced dataset indicated that the bagCART had a high performance followed by RF, bootCART, GBM and AdaBoost. The GBM classifier performance dropped on the balanced dataset. In contrast to imbalanced dataset, there was a decrease in the precision and recall performances of GBM classifier after resampling. In overall, the results indicate that the bagging classifiers (RF and bagCART) had better performance in comparison to the boosting classifiers (bootCART, GBM and AdaBoost). A possible explanation for this might be that the bagging techniques taking into consideration overfitting problems while boosting does not. The stable performance showed by bagging classifiers (RF and bagCART) arise from its training set, which includes iterative assignments of misclassification weights to training data and therefore they address minor class instances than other ensemble classifiers do (Galar et al. 2011).

Our findings on the balanced dataset complement previous results by Olaya-Marín et al. (2013), who performed a comparative study of ANN and RF to predict fish species richness in Mediterranean rivers; Cutler et al. (2007) who applied RF to compare model accuracies of four invasive plant species, four lichen species and three cavity nesting bird species and Crisci et al. (2012) who applied RF, CART, SVM, bagging and boosting techniques on species abundance and then compared their performances. Most of these studies used the accuracy metric to assess model performance without checking class imbalance. The study results confirms previous findings Mosavi et al. (2021), indicating that bagging classifiers outperformed boosting classifiers. Based on

our results, we found that the performances of bagging classifiers (RF and bagCART) were significantly higher than the other classifiers based on recall and F1-score. The *Oreochromis niloticus* species showed a good performance regardless of resampling technique, hence it is a dominant species for our study.

4.6 Conclusion

This study looked at the problem of classifying multiple species in population ecology. The experiments were conducted on four fish (*Lates niloticus*, *Oreochromis niloticus*, *Clarias gariepinus* and *Synodontis afrofischeri*) species. Five modeling techniques were compared and evaluated on imbalanced and balanced datasets for classifying multiple species from different sampling sites of Uganda. The recall and F1-score performance metrics were used to assess machine learning classifiers.

Numerous machine learning classifiers for multi-class imbalanced such as bagging and boosting methods have been proposed in the literature. The bagging (RF and bagCART) and boosting (bootCART, GBM and AdaBoost) classifiers in combination with resampling were tested and their performances were compared. The experimental results proved that the proposed techniques are robust when the resampling and ensemble techniques combined were applied. The application of resampling using ROS increased the TP rate and decreased the FN rate which significantly improved the recall and F1-score of all classifiers. The bagging classifiers (RF and bagCART) showed higher performances on the imbalanced dataset. Therefore, the effects of imbalanced dataset on classifiers performances differed depending on how the FN rate was reflected in different performance metrics. The bagging classifiers (RF and bagCART) achieved higher performances on the balanced dataset and the boosting classifiers (bootCART, GBM and AdaBoost) achieved lower performances on the balanced dataset. We noted that the bagging classifiers (RF and bagCART) performed well on both datasets. Based on two measures of model performance (recall and F1-

score), bagging algorithms appeared to be the best techniques for classifying multiple species for our dataset. The strong performances exhibited by the bagging classifiers (RF and bagCART) suggest that they can be used for classifying multiple species in ecological studies. The results of this study obtained competitive results to those of Mosavi et al. (2021).

The results of this study provide insight for addressing class imbalanced problem in multi-class classification tasks, e.g., classification of multiple species based on classes, which are commonly encountered by ecologists working with count datasets. Multi-class imbalanced classification problems have significant implications for multiple species count data, as most studies often do not report the nature of the dataset, that is, balanced versus imbalanced, which influence the predictive ability of multi-class classification models. Comparing our findings with the current literature, bagging techniques provides better ecological interpretability. We suggest that machine learning classifiers be applied to ongoing ecological studies that are facing data problems of this nature. Future studies, however, should also consider deep learning approaches to address the multi-class imbalanced classification problem in ecological studies.

Chapter 5

5. Conclusion

This study aimed at estimating zero-inflated overdispersed count data that is rampant in ecological studies. This is a major challenge in population studies. The novelty of this study was to demonstrate that ignoring overdispersion leads to incorrect parameter estimates, which may result in researchers identifying predictors as having biological meaningful effect when in fact they do not. Hence, the assessment of overdispersion is very important for count data. This study used ML techniques to try and reduce overdispersion in ecological studies.

Chapter 2 explored machine learning techniques to predict the relationship between species abundance and habitat suitability. The focus was on employing machine learning techniques since species distribution models are believed to be frequently producing inaccurate results. Machine learning techniques were applied to experimental dataset to predict the relationship between multiple species and habitat suitability by comparing their predictive performances. Different resampling and ensemble learning techniques were used to handle the class imbalanced problem. The purpose of this review was to encourage ecology community to use machine learning expertise to species distribution models instead of using generalized linear models. Based on the literature provided about species distribution models failing to give accurate results, this study recommended ensemble machine learning (ML) techniques. For future research purposes it is advisable to check the class imbalanced ratio when dealing with multiple species dataset. This could assist in determining the best predictive model for imbalanced dataset. The finding of this study showed that species abundance and habitat suitability can be predicted using ensemble ML techniques. Future studies need to improve species distribution models by implementing the deep learning techniques to address the species abundance and their environmental suitability.

Chapter 3 looked at the comparison of statistical regression count models and machine learning regression techniques in reducing the overdispersion problem in ecological

count data. This study has shown that when statistical count models are applied to ecological count data, the application of negative binomial and zero-inflated negative binomial models gives better performance than the Poisson and zero-inflated Poisson models. The performance of statistical count model is compared to machine learning models. Various machine learning techniques have been used to compare their performance in reducing overdispersion. Our findings confirm that machine learning techniques are more capable to reduce overdispersion in ecological count data. Machine learning techniques are recommended in this study, these techniques perform well in many research fields than statistical methods. In future, the application of ML regression techniques in ecology will become an increasingly attractive tool for ecologists. Moreover, we conclude that the effect of overdispersion in ecological count data is largely dependent on the dispersion parameter of the fitted model. We suggest that machine learning classifiers be applied to ongoing ecological studies that are facing data problems of this nature.

Chapter 4 looked at the classification of multiple species using machine learning approaches. Much work in the ecological literature has focused on modeling of single species and very few studies on classification of multiple species. When classifying multiple species, we noted the multi-class imbalanced classification problem. Numerous techniques for handling multi-class imbalance such as ensemble techniques have been applied. Basically, the performance of bagging and boosting classifiers were assessed under multiple species and the findings suggested that bagging classifiers can be used to classify multiple species in ecological studies. Future studies, however, should also consider deep learning approaches to address the multi-class imbalanced classification problem in ecological studies.

Bibliography

- Achu, A., Thomas, J., Aju, C., Gopinath, G., Kumar, S. & Reghunath, R. 2021. ‘Machine-learning modelling of fire susceptibility in a forest-agriculture mosaic landscape of southern india’, *Ecological Informatics* **64**, 101–348.
- Adnan, M. N. & Islam, M. Z. 2015. One-vs-all binarization technique in the context of random forest, *in* ‘Proceedings of the european symposium on artificial neural networks, computational intelligence and machine learning’, pp. 385–390.
- Agarwal, D. K., Gelfand, A. E. & Citron-Pousty, S. 2002. ‘Zero-inflated models with application to spatial count data’, *Environmental and Ecological Statistics* **9**(4), 341–355.
- Alfaro, E., Gamez, M. & Garcia, N. 2013. ‘adabag: An r package for classification with boosting and bagging’, *Journal of Statistical Software* **54**(1), 1–35.
- Allredge, M., Pollock, K., Simons, T. & Shriner, S. 2007. ‘Multiple-species analysis of point count data: a more parsimonious modelling framework’, *Journal of Applied Ecology* **44**(2), 281–290.
- Aráujo, E. G., Vasconcelos, J., dos Santos, D. P., Ortega, E. M., de Souza, D. & Zanetoni, J. P. 2021. ‘The zero-inflated negative binomial semiparametric regression model: Application to number of failing grades data’, *Annals of Data Science* pp. 1–16.
- Bagheri, A. M., Montazer, G. A. & Escalera, S. 2012. Error correcting output codes for multiclass classification: application to two image vision problems, *in* ‘The 16th CSI international symposium on artificial intelligence and signal processing (AISP 2012)’, IEEE, pp. 508–513.
- Bajer, D., Zonć, B., Dudjak, M. & Martinović, G. 2019. Performance analysis of smote-based oversampling techniques when dealing with data imbalance, *in* ‘2019

- International Conference on Systems, Signals and Image Processing (IWSSIP)', IEEE, pp. 265–271.
- Bannach-Brown, A., Przybyła, P., Thomas, J., Rice, A. S., Ananiadou, S., Liao, J. & Macleod, M. R. 2019. 'Machine learning algorithms for systematic review: reducing workload in a preclinical review of animal studies and reducing human screening error', *Systematic Reviews* **8**(1), 1–12.
- Basgall, M. J., Hasperu , W., Naiouf, M., Fern ndez, A. & Herrera, F. 2018. Smote-bd: An exact and scalable oversampling method for imbalanced classification in big data, *in* 'VI Jornadas de Cloud Computing & Big Data (La Plata, 2018)'.
- Beery, S., Cole, E., Parker, J., Perona, P. & Winner, K. 2021. Species distribution modeling for machine learning practitioners: a review, *in* 'ACM SIGCAS Conference on Computing and Sustainable Societies', pp. 329 – 348.
- Bent jac, C., Cs rg , A. & Mart nez-Mu oz, G. 2021. 'A comparative analysis of gradient boosting algorithms', *Artificial Intelligence Review* **54**(3), 1937–1967.
- Bolker, B. M. 2008. *Ecological models and data in R*, Princeton University Press.
- Bolker, B. M., Brooks, M. E., Clark, C. J., Geange, S. W., Poulsen, J. R., Stevens, M. H. H. & White, J. S. S. 2009. 'Generalized linear mixed models: a practical guide for ecology and evolution', *Trends in Ecology & Evolution* **24**(3), 127–135.
- Bourel, M. & Segura, A. 2018. 'Multiclass classification methods in ecology', *Ecological Indicators* **85**, 1012–1021.
- Brooks, M. E., Kristensen, K., van Benthem, K. J., Magnusson, A., Berg, C. W., Nielsen, A., Skaug, H. J., Machler, M. & Bolker, B. M. 2017. 'glmmTMB balances speed and flexibility among packages for zero-inflated generalized linear mixed modeling', *The R Journal* **9**(2), 378–400.
- Brooks, S., Gelman, A., Jones, G. & Meng, X.-L. 2011. *Handbook of markov chain monte carlo*, CRC press.

- Buuren, S. V. & Groothuis-Oudshoorn, K. 2010. ‘mice: Multivariate imputation by chained equations in r’, *Journal of Statistical Software* pp. 1–68.
- Buyrukoğlu, G., Buyrukoğlu, S. & Topalcengiz, Z. 2021. ‘Comparing regression models with count data to artificial neural network and ensemble models for prediction of generic escherichia coli population in agricultural ponds based on weather station measurements’, *Microbial Risk Analysis* pp. 100–171.
- Cano, A. & Krawczyk, B. 2022. ‘Rose: robust online self-adjusting ensemble for continual learning on imbalanced drifting data streams’, *Machine Learning* pp. 1–39.
- Carrasco, J., Lisón, F. & Weintraub, A. 2021. ‘Rusboost: A suitable species distribution method for imbalanced records of presence and absence. a case study of twenty-five species of iberian bats’, *BioRxiv* .
- Chazdon, R. L., Chao, A., Colwell, R. K., Lin, S.-Y., Norden, N., Letcher, S. G., Clark, D. B., Finegan, B. & Arroyo, J. P. 2011. ‘A novel statistical method for classifying habitat generalists and specialists’, *Ecology* **92**(6), 1332–1343.
- Cheung, Y. B. 2002. ‘Zero-inflated models for regression analysis of count data: a study of growth and development’, *Statistics in Medicine* **21**(10), 1461–1469.
- Collell, G., Prelec, D. & Patil, K. 2016. ‘Reviving threshold-moving: a simple plugin bagging ensemble for binary and multiclass imbalanced data’, *arXiv preprint arXiv:1606.08698* .
- Crisci, C., Ghattas, B. & Perera, G. 2012. ‘A review of supervised machine learning algorithms and their applications to ecological data’, *Ecological Modelling* **240**, 113–122.
- Cushman, S. A. & Huettmann, F. 2010. *Spatial complexity, informatics, and wildlife conservation*, Springer.

- Cutler, D. R., Edwards Jr, T. C., Beard, K. H., Cutler, A., Hess, K. T., Gibson, J. & Lawler, J. J. 2007. 'Random forests for classification in ecology', *Ecology* **88**(11), 2783–2792.
- Dangeti, P. 2017. *Statistics for machine learning*, Packt Publishing Ltd.
- Dastile, X., Celik, T. & Potsane, M. 2020. 'Statistical and machine learning models in credit scoring: A systematic literature survey', *Applied Soft Computing* pp. 106–263.
- Desjardins, C. D. 2016. 'Modeling zero-inflated and overdispersed count data: An empirical study of school suspensions', *The Journal of Experimental Education* **84**(3), 449–472.
- Drake, J. M., Randin, C. & Guisan, A. 2006. 'Modelling ecological niches with support vector machines', *Journal of Applied Ecology* **43**(3), 424–432.
- Elliott, R. J. & Morrell, C. H. 2009. *Learning SAS in the computer lab*, Cengage Learning.
- Faraway, J. J. 2016. *Extending the linear model with R: generalized linear, mixed effects and nonparametric regression models*, Chapman and Hall/CRC.
- Farrell, A., Wang, G., Rush, S. A., Martin, J. A., Belant, J. L., Butler, A. B. & Godwin, D. 2019. 'Machine learning of large-scale spatial distributions of wild turkeys with high-dimensional environmental data', *Ecology and Evolution* **9**(10), 5938–5949.
- Feng, W., Sun, J., Zhang, L., Cao, C. & Yang, Q. 2016. A support vector machine based naive bayes algorithm for spam filtering, in '2016 IEEE 35th International Performance Computing and Communications Conference (IPCCC)', IEEE, pp. 1–8.
- Fox, J. 2015. *Applied regression analysis and generalized linear models*, Sage Publications.

- Friedman, J. H. 2002. ‘Stochastic gradient boosting’, *Computational Statistics & Data Analysis* **38**(4), 367–378.
- Gaber, T., Tharwat, A., Hassanien, A. E. & Snasel, V. 2016. ‘Biometric cattle identification approach based on weber’s local descriptor and adaboost classifier’, *Computers and Electronics in Agriculture* **122**, 55–66.
- Galar, M., Fernandez, A., Barrenechea, E., Bustince, H. & Herrera, F. 2011. ‘A review on ensembles for the class imbalance problem: bagging-, boosting-, and hybrid-based approaches’, *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)* **42**(4), 463–484.
- Garay, A. M., Hashimoto, E. M., Ortega, E. M. & Lachos, V. H. 2011. ‘On estimation and influence diagnostics for zero-inflated negative binomial regression models’, *Computational Statistics & Data Analysis* **55**, 1304–1318.
- Garzon, M. B., Blazek, R., Neteler, M., De Dios, R. S., Ollero, H. S. & Furlanello, C. 2006. ‘Predicting habitat suitability with machine learning models: the potential area of pinus sylvestris l. in the iberian peninsula’, *Ecological Modelling* **197**(3–4), 383–393.
- Ghannam, R. B. & Techtmann, S. M. 2021. ‘Machine learning applications in microbial ecology, human microbiome studies, and environmental monitoring’, *Computational and Structural Biotechnology Journal* **19**, 1092–1107.
- Goksuluk, D., Zararsiz, G., Korkmaz, S., Eldem, V., Zararsiz, G. E., Ozcetin, E., Ozturk, A. & Karaagaoglu, A. E. 2019. ‘Mlseq: Machine learning interface for rna-sequencing data’, *Computer Methods and Programs in Biomedicine* **175**, 223–231.
- Gupta, R., Szczesniak, R. D. & Macaluso, M. 2015. ‘Modeling repeated count measures with excess zeros in an epidemiological study’, *Annals of Epidemiology* **25**(8), 583–589.

- Hardin, J. W., Hardin, J. W., Hilbe, J. M. & Hilbe, J. 2007. *Generalized linear models and extensions*, Stata press.
- Harrison, X. A. 2014. ‘Using observation-level random effects to model overdispersion in count data in ecology and evolution’, *PeerJ* **2**, 616.
- Hauser-Davis, R., Oliveira, T., Silveira, A., Silva, T. & Ziolli, R. 2010. ‘Case study: Comparing the use of nonlinear discriminating analysis and artificial neural networks in the classification of three fish species: acaras (*geophagus brasiliensis*), tilapias (*tilapia rendalli*) and mullets (*mugil liza*)’, *Ecological Informatics* **5**(6), 474–478.
- Hayes, T. & McArdle, J. J. 2017. ‘Should we impute or should we weight? examining the performance of two cart-based techniques for addressing missing data in small sample research with nonnormal variables’, *Computational Statistics & Data Analysis* **115**, 35–52.
- He, H., Bai, Y., Garcia, E. & Li, S. A. 2008. adaptive synthetic sampling approach for imbalanced learning. iee international joint conference on neural networks, in ‘2008 (IEEE World Congress On Computational Intelligence)’.
- He, H., Zhang, H., Ye, P. & Tang, W. 2019. ‘A test of inflated zeros for poisson regression models’, *Statistical Methods in Medical Research* **28**(4), 1157–1169.
- Hegel, T. M., Cushman, S. A., Evans, J. & Huettmann, F. 2010. Current state of the art for statistical modelling of species distributions, in ‘Spatial complexity, informatics, and wildlife conservation’, Springer, pp. 273–311.
- Hilbe, J. M. 2011*a*. *Modeling count data*, Cambridge University Press.
- Hilbe, J. M. 2011*b*. *Negative binomial regression*, Cambridge University Press.
- Hu, X., Mei, H., Zhang, H., Li, Y. & Li, M. 2021. ‘Performance evaluation of ensemble learning techniques for landslide susceptibility mapping at the jinping county, southwest china’, *Natural Hazards* **105**(2), 1663–1689.

- Jabeen, A., Ahmad, N. & Raza, K. 2018. Machine learning-based state-of-the-art methods for the classification of rna-seq data, *in* ‘Classification in BioApps’, Springer.
- Jain, P. & Kapoor, A. 2009. Active learning for large multi-class problems, *in* ‘2009 IEEE Conference on Computer Vision and Pattern Recognition’, IEEE, pp. 762–769.
- James, G., Witten, D., Hastie, T. & Tibshirani, R. 2013. *An introduction to statistical learning*, Vol. 112, Springer.
- Jerez, J. M., Molina, I., García-Laencina, P. J., Alba, E., Ribelles, N., Martín, M. & Franco, L. 2010. ‘Missing data imputation using statistical and machine learning methods in a real breast cancer problem’, *Artificial Intelligence in Medicine* **50**(2), 105–115.
- Jiao, Y. & Du, P. 2016. ‘Performance measures in evaluating machine learning based bioinformatics predictors for classifications’, *Quantitative Biology* **4**(4), 320–330.
- Jones, J., Miller, J. & White, M. 2011. Multi-label classification for multi-species distribution modeling, *in* ‘Proceedings of the 28th international conference on machine learning, New York, NY’, pp. 1–4.
- Kampichler, C., Wieland, R., Calmé, S., Weissenberger, H. & Arriaga-Weiss, S. 2010. ‘Classification in conservation biology: a comparison of five machine-learning methods’, *Ecological Informatics* **5**(6), 441–450.
- Kassahun, W., Neyens, T., Faes, C., Molenberghs, G. & Verbeke, G. 2014. ‘A zero-inflated overdispersed hierarchical poisson model’, *Statistical Modelling* **14**(5), 439–456.
- Kassambara, A. 2018. *Machine Learning Essentials: Practical Guide in R*, sthda.

- Khoshgoftaar, T. M., Gao, K. & Szabo, R. M. 2001. An application of zero-inflated poisson regression for software fault prediction, *in* 'Proceedings 12th International Symposium on Software Reliability Engineering', IEEE, pp. 66–73.
- Khushi, M., Shaukat, K., Alam, T. M., Hameed, I. A., Uddin, S., Luo, S., Yang, X. & Reyes, M. C. 2021. 'A comparative performance analysis of data resampling methods on imbalance medical data', *IEEE Access* **9**, 109960–109975.
- Kotsiantis, S. B., Zaharakis, I. & Pintelas, P. 2007. 'Supervised machine learning: A review of classification techniques', *Emerging Artificial Intelligence Applications in Computer Engineering* **160**, 3–24.
- Kou, G., Chen, H. & Hefni, M. A. 2022. 'Improved hybrid resampling and ensemble model for imbalance learning and credit evaluation', *Journal of Management Science and Engineering* **7**(4), 511–529.
- Kruppa, J. & Hothorn, L. 2020. 'A comparison study on modeling of clustered and overdispersed count data for multiple comparisons', *Journal of Applied Statistics* pp. 1–13.
- Lambert, D. 2012. 'Zero-inflated poisson regression, with an application to defects in manufacturing', *Technometrics* **34**(1), 1–14.
- Lango, M. 2019. 'Tackling the problem of class imbalance in multi-class sentiment classification: An experimental study', *Foundations of Computing and Decision Sciences* **44**(2), 151–178.
- Lee, W.-H., Song, J.-W., Yoon, S.-H. & Jung, J.-M. 2022. 'Spatial evaluation of machine learning-based species distribution models for prediction of invasive ant species distribution', *Applied Sciences* **12**(20), 10260.
- Leke-Betechuoh, B., Marwala, T., Tim, T. & Lagazio, M. 2006. Prediction of hiv status from demographic data using neural networks, *in* '2006 IEEE International Conference on Systems, Man and Cybernetics', Vol. 3, IEEE, pp. 2339–2344.

- Leke, C. A. & Marwala, T. 2019. Introduction to missing data estimation, *in* ‘Deep Learning and Missing Data in Engineering Systems’, Springer, pp. 1–20.
- Lindén, A. & Mäntyniemi, S. 2011. ‘Using the negative binomial distribution to model overdispersion in ecological count data’, *Ecology* **92**(7), 1414–1421.
- Liu, E., Zhao, H., Zhang, S., He, J., Yang, X. & Xiao, X. 2021. ‘Identification of plant species in an alpine steppe of northern tibet using close-range hyperspectral imagery’, *Ecological Informatics* **61**, 101–213.
- Lo, C. & Marculescu, R. 2019. ‘MetaNN: accurate classification of host phenotypes from metagenomic data using neural networks’, *Bmc Bioinformatics* **20**, 1–14.
- Lord, D. & Mannering, F. 2010. ‘The statistical analysis of crash-frequency data: A review and assessment of methodological alternatives’, *Transportation Research Part A: Policy and Practice* **44**(5), 291–305.
- Marivate, V. N., Nelwamondo, F. V. & Marwala, T. 2008. ‘Investigation into the use of autoencoder neural networks, principal component analysis and support vector regression in estimating missing hiv data’, *IFAC Proceedings Volumes* **41**(2), 682–689.
- Martin, T. G., Wintle, B. A., Rhodes, J. R., Kuhnert, P. M., Field, S. A., Low-Choy, S. J., Tyre, A. J. & Possingham, H. P. 2005. ‘Zero tolerance ecology: improving ecological inference by modelling the source of zero observations’, *Ecology Letters* **8**(11), 1235–1246.
- Marwala, T. 2009. *Computational Intelligence for Missing Data Imputation, Estimation, and Management: Knowledge Optimization Techniques*, IGI Global.
- Matthews, G. J., Brophy, J. K., Luetkemeier, M., Gu, H. & Thiruvathukal, G. K. 2018. ‘A comparison of machine learning techniques for taxonomic classification of teeth from the family bovidae’, *Journal of Applied Statistics* **45**(15), 2773–2787.

- Minami, M., Lennert-Cody, C. E., Gao, W. & Roman-Verdesoto, M. 2007. 'Modeling shark bycatch: the zero-inflated negative binomial regression model with smoothing', *Fisheries Research* **84**(2), 210–221.
- Moisen, G. G., Freeman, E. A., Blackard, J. A., Frescino, T. S., Zimmermann, N. E. & Edwards Jr, T. C. 2006. 'Predicting tree species presence and basal area in utah: a comparison of stochastic gradient boosting, generalized additive models, and tree-based methods', *Ecological Modelling* **199**(2), 176–187.
- Mosavi, A., Sajedi Hosseini, F., Choubin, B., Goodarzi, M., Dineva, A. A. & Rafiei Sardooi, E. 2021. 'Ensemble boosting and bagging based machine learning models for groundwater potential prediction', *Water Resources Management* **35**(1), 23–37.
- Muhamedyev, R. 2015. 'Machine learning methods: An overview', *Computer modelling & new technologies* **19**(6), 14–29.
- Muñoz-Mas, R., Gil-Martínez, E., Oliva-Paterna, F. J., Belda, E. J. & Martínez-Capel, F. 2019. 'Tree-based ensembles unveil the microhabitat suitability for the invasive bleak (*alburnus alburnus* l.) and pumpkinseed (*lepomis gibbosus* l.): Introducing xgboost to eco-informatics', *Ecological Informatics* **53**, 100–974.
- Naghibi, S. A., Pourghasemi, H. R. & Dixon, B. 2016. 'Gis-based groundwater potential mapping using boosted regression tree, classification and regression tree, and random forest machine learning models in iran', *Environmental Monitoring and Assessment* **188**(1), 1–27.
- Ng, A. 2019. 'Machine learning yearning: Technical strategy for ai engineers in the era of deep learning', Retrieved online at [https://www. mlyearning. org](https://www.mlyearning.org) .
- Okun, O., Valentini, G. & Re, M. 2011. *Ensembles in machine learning applications*, Springer Science & Business Media.

- Olaya-Marín, E. J., Martínez-Capel, F. & Vezza, P. 2013. ‘A comparison of artificial neural networks and random forests to predict native fish species richness in mediterranean rivers’, *Knowledge and Management of Aquatic Ecosystems* (409), 07.
- Olden, J. D., Lawler, J. J. & Poff, N. L. 2008. ‘Machine learning methods without tears: a primer for ecologists’, *The Quarterly Review of Biology* **83**(2), 171–193.
- Oliveira, M., Einbeck, J., Higuera, M., Ainsbury, E., Puig, P. & Rothkamm, K. 2016. ‘Zero-inflated regression models for radiation-induced chromosome aberration data: A comparative study’, *Biometrical Journal* **58**(2), 259–279.
- O’hara, R. B. & Kotze, D. J. 2010. ‘Do not log-transform count data’, *Methods in ecology and Evolution* **1**(2), 118–122.
- Pelckmans, K., De Brabanter, J., Suykens, J. A. & De Moor, B. 2005. ‘Handling missing values in support vector machine classifiers’, *Neural Networks* **18**(5-6), 684–692.
- Plant, E., King, R. & Kath, J. 2021. ‘Statistical comparison of additive regression tree methods on ecological grassland data’, *Ecological Informatics* **61**, 101–198.
- R Core Team 2022. *R: A language and environment for statistical computing*, R Foundation for Statistical Computing, Vienna, Austria.
URL: <https://www.R-project.org/>
- Rendón, E., Alejo, R., Castorena, C., Isidro-Ortega, F. J. & Granda-Gutiérrez, E. E. 2020. ‘Data sampling methods to deal with the big data multi-class imbalance problem’, *Applied Sciences* **10**(4), 12–76.
- Rew, J., Cho, Y. & Hwang, E. 2021. ‘A robust prediction model for species distribution using bagging ensembles with deep neural networks’, *Remote Sensing* **13**(8), 14–95.

- Rew, J., Cho, Y., Moon, J. & Hwang, E. 2020. ‘Habitat suitability estimation using a two-stage ensemble approach’, *Remote Sensing* **12**(9), 14–75.
- Rice, J. A. 2006. *Mathematical statistics and data analysis*, Cengage Learning.
- Richards, S. A. 2008. ‘Dealing with overdispersed count data in applied ecology’, *Journal of Applied Ecology* **45**(1), 218–227.
- Ridout, M., Hinde, J. & DeméAtrio, C. G. 2001. ‘A score test for testing a zero-inflated poisson regression model against zero-inflated negative binomial alternatives’, *Biometrics* **57**(1), 219–223.
- Rose, C. E., Martin, S. W., Wannemuehler, K. A. & Plikaytis, B. D. 2006. ‘On the use of zero-inflated and hurdle models for modeling vaccine adverse event count data’, *Journal of Biopharmaceutical Statistics* **16**(4), 463–481.
- Royle, J. A., Nichols, J. D. & Kéry, M. 2005. ‘Modelling occurrence and abundance of species when detection is imperfect’, *Oikos* **110**(2), 353–359.
- Rustamov, K. 2022. ‘Comparison of different models in predicting habitat suitability of rare species in uzbekistan: 8 rare tulipa species case-study.’, *BioRxiv* .
- Sahare, M. & Gupta, H. 2012. ‘A review of multi-class classification for imbalanced data’, *International Journal of Advanced Computer Research* **2**(3), 160.
- Saini, R. & Ghosh, S. 2017. Ensemble classifiers in remote sensing: A review, *in* ‘2017 International Conference on Computing, Communication and Automation’, IEEE, pp. 1148–1152.
- Saito, T. & Rehmsmeier, M. 2015. ‘The precision-recall plot is more informative than the roc plot when evaluating binary classifiers on imbalanced datasets’, *PloS One* **10**(3).

- Sakthivel, K. & Rajitha, C. 2017. 'A comparative study of zero-inflated, hurdle models with artificial neural network in claim count modeling', *International Journal of Statistics and Systems* **12**(2), 265–276.
- Samal, P., Srivastava, J., Saraf, P. N., Charles, B. & Singarasubramanian, S. 2022. 'Ensemble modeling approach to predict the past and future climate suitability for two mangrove species along the coastal wetlands of peninsular india', *Ecological Informatics* pp. 101–819.
- Shidha, M. & Mahalekshmi, T. 2019. 'An emperical study on the effect of resampling techniques in imbalaced datasets for improving consistency of classifiers', *Appl. Eng. Res* **14**(7), 1516–1525.
- Shin, J., Yoon, S., Kim, Y., Kim, T., Go, B. & Cha, Y. 2021. 'Effects of class imbalance on resampling and ensemble learning for improved prediction of cyanobacteria blooms', *Ecological Informatics* **61**, 101–202.
- Sokolova, M. & Lapalme, G. 2009. 'A systematic analysis of performance measures for classification tasks', *Information Processing & Management* **45**(4).
- Sooknunan, K., Lochner, M., Bassett, B., Peiris, H., Fender, R., Stewart, A., Pietka, M., Woudt, P., McEwen, J. & Lahav, O. 2021. 'Classification of multiwavelength transients with machine learning', *Monthly Notices of the Royal Astronomical Society* **502**(1), 206–224.
- Spedicato, G. A., Dutang, C. & Petrini, L. 2018. 'Machine learning methods to perform pricing optimization. a comparison with standard glms', *Variance* **12**(1), 69–89.
- Srinivasa, K., Siddesh, G. & Manisekhar, S. 2020. *Statistical Modelling and Machine Learning Principles for Bioinformatics Techniques, Tools, and Applications*, Springer Nature.

- Statnikov, A., Henaff, M., Narendra, V., Konganti, K., Li, Z., Yang, L., Pei, Z., Blaser, M. J., Aliferis, C. F. & Alekseyenko, A. V. 2013. ‘A comprehensive evaluation of multcategory classification methods for microbiomic data’, *Microbiome* **1**(1), 1–12.
- Sun, Y., Kamel, M. S. & Wang, Y. 2006. Boosting for learning multiple classes with imbalanced class distribution, *in* ‘Sixth International Conference on Data Mining (ICDM’06)’, IEEE, pp. 592–602.
- Sun, Y., Kamel, M., Wong, A. & Wang, Y. 2007. ‘Cost-sensitive boosting for classification of imbalanced data’, *Pattern Recognition* **40**(12), 3358–3378.
- Tharwat, A. 2020. ‘Classification assessment methods’, *Applied Computing and Informatics* .
- Tharwat, A., Hemedan, A. A., Hassanien, A. E. & Gabel, T. 2018. ‘A biometric-based model for fish species classification’, *Fisheries research* **204**, 324–336.
- Thessen, A. 2016. ‘Adoption of machine learning techniques in ecology and earth science’, *One Ecosystem* **1**, e8621.
- Torgo, L. 2011. *Data mining with R: learning with case studies*, Chapman and Hall/CRC.
- Trevor, H., Robert, T. & JH, F. 2009. *The elements of statistical learning: data mining, inference, and prediction*, New York, NY: Springer.
- Tuda, M. & Luna-Maldonado, A. I. 2020. ‘Image-based insect species and gender classification by trained supervised machine learning algorithms’, *Ecological Informatics* **60**, 101–135.
- Valavi, R., Guillera-Aroita, G., Lahoz-Monfort, J. J. & Elith, J. 2022. ‘Predictive performance of presence-only species distribution models: a benchmark study with reproducible code’, *Ecological Monographs* **92**(1), e01486.

- Ver Hoef, J. M. & Boveng, P. L. 2007. ‘Quasi-poisson vs. negative binomial regression: How should we model overdispersed count data?’, *Ecology* **88**(11), 2766–2772.
- Wagh, Y. S. & Kamalja, K. K. 2018. ‘Zero-inflated models and estimation in zero-inflated poisson distribution’, *Communications in Statistics-Simulation and Computation* **47**(8), 2248–2265.
- Wang, S., Aggarwal, C. & Liu, H. 2017. Using a random forest to inspire a neural network and improving on it, *in* ‘Proceedings of the 2017 SIAM international conference on data mining’, SIAM, pp. 1–9.
- Wang, S. & Yao, X. 2012. ‘Multiclass imbalance problems: Analysis and potential solutions’, *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* **42**(4), 1119–1130.
- Wenger, S. J. & Freeman, M. C. 2008. ‘Estimating species occurrence, abundance, and detection probability using zero-inflated distributions’, *Ecology* **89**(10).
- Wright, W., Irvine, K., Almberg, E. & Litt, A. 2020. ‘Modelling misclassification in multi-species acoustic data when estimating occupancy and relative activity’, *Methods in Ecology and Evolution* **11**(1), 71–81.
- Yang, H., Chan, L. & King, I. 2002. Support vector machine regression for volatile stock market prediction, *in* ‘International Conference on Intelligent Data Engineering and Automated Learning’, Springer, pp. 391–396.
- Youssef, A. M. & Pourghasemi, H. R. 2021. ‘Landslide susceptibility mapping using machine learning algorithms and comparison of their performance at Abha Basin, Asir Region, Saudi Arabia’, *Geoscience Frontiers* **12**, 639–655.
- Yu, H., Cooper, A. R. & Infante, D. M. 2020. ‘Improving species distribution model predictive accuracy using species abundance: Application with boosted regression trees’, *Ecological Modelling* **432**, 109–202.

- Zamani, H. & Ismail, N. 2013. ‘Score test for testing zero-inflated poisson regression against zero-inflated generalized poisson alternatives’, *Journal of Applied Statistics* **40**, 2056–2068.
- Zararsiz, G., Elmali, F. & Ozturk, A. 2012. ‘Bagging support vector machines for leukemia classification’, *International Journal of Computer Science Issues IJCSI* **9**(6), 355.
- Zararsiz, G., Goksuluk, D., Korkmaz, S., Eldem, V., Duru, I. P., Unver, T. & Ozturk, A. 2014. ‘Classification of rna-seq data via bagging support vector machines’, *BioRxiv* pp. 007–526.
- Zhang, X., Waller, S. T. & Jiang, P. 2020. ‘An ensemble machine learning-based modeling framework for analysis of traffic crash frequency’, *Computer-Aided Civil and Infrastructure Engineering* **35**(3), 258–276.
- Zhang, Z., Krawczyk, B., Garcia, S., Rosales-Pérez, A. & Herrera, F. 2016. ‘Empowering one-vs-one decomposition with ensemble learning for multi-class imbalanced data’, *Knowledge-Based Systems* **106**, 251–263.
- Zhu, F. 2012. ‘Zero-inflated poisson and negative binomial integer-valued garch models’, *Journal of Statistical Planning and Inference* **142**(4), 826–839.
- Zuur, A., Ieno, E. N., Walker, N., Saveliev, A. A. & Smith, G. M. 2009. *Mixed effects models and extensions in ecology with R*, Springer Science & Business Media.

Appendix A. Appendix

This appendix summarizes the statistics that underpin the work in this thesis.

A.1 Statistical models

This section gives a statistical overview on how parameter estimates were derived for count regression models.

A.1.1 Random variables

A random variable is defined as a variable whose value is unknown or a function that assigns values to each of an experiment's outcomes. There are two types of random variables; a discrete random variable and a continuous random variable. A discrete random variable is a random variable that can take on only a finite or at most a countably infinite number of values. The distribution of a discrete random variable Y is the sequence of probability $P(Y = y)$ for all values of y for which $P(Y = y) > 0$ (Rice 2006).

For a continuous random variable, the role of the frequency function is taken by a density function, $f(x)$, which has the properties that $f(x) \geq 0$, f is piecewise continuous and $\int_{-\infty}^{\infty} f(x) dx = 1$. A cumulative distribution function (CDF) of a random variable is defined as $F(x) = P(X \leq x)$ where $-\infty < x < \infty$.

If the random variable Y has a probability density function $f_y(\cdot)$, the moment generating function (mgf) of a random variable is defined to be:

$$M_Y(t) = E(e^{tY}).$$

It is said that an mgf for Y exists if there exists a positive constant b such that $M_Y(t)$ is finite for $|t| \leq b$ (Rice 2006).

A.1.2 Parameter estimation methods

Estimation of parameters for statistical models is a key step in most statistical analysis. The parameters of distributions are estimated using maximum likelihood estimation (MLE) method. The likelihood function is given by

$$L(y; \boldsymbol{\theta}) = \prod_{i=1}^n f(y; \boldsymbol{\theta})$$

where $\boldsymbol{\theta} = \{\theta_1, \theta_2, \dots, \theta_k\}$ denote the parameter set and $f(y; \boldsymbol{\theta})$ present the probability distribution function (p.d.f.) of the proposed distribution. The likelihood function is used to find the value of $\boldsymbol{\theta}$ which maximizes $L(\boldsymbol{\theta}|y)$.

A.1.3 Poisson distribution

Let y_i be a response variable such as the number of individual fish species, for $i = 0, 1, 2, \dots, n$ that follows a Poisson distribution with parameter μ_i . The probability distribution function of a Poisson is defined as

$$P(Y = y) = \frac{\exp(-\mu) \times \mu^y}{y!}$$

where μ denotes the expectation of the response variable and y is the non-negative integer. Poisson distribution has one parameter and its mean and variance are the same.

Parameter estimation for Poisson model

Suppose $\mathbf{y} = y_1, y_2, \dots, y_n$ is a random sample of size n from the Poisson distribution. The likelihood function is defined by

$$L = \prod_{i=1}^n \frac{\exp(-\mu) \times \mu^{y_i}}{y_i!}$$

The log-likelihood function is thus obtained as

$$\begin{aligned} \ell &= \log L \\ &= \mu \log \sum_{i=1}^n y_i - n\mu - \log \sum_{i=1}^n y_i! \end{aligned}$$

The first derivate of the log-likelihood function is defined by

$$\frac{\partial \ell}{\partial \mu} = \frac{1}{\mu} \sum_{i=1}^n (y_i - n)$$

To find the MLE, $\hat{\mu}$, use

$$\begin{aligned} \frac{\partial \ell}{\partial \mu} &= 0 \\ \frac{1}{\mu} \sum_{i=1}^n (y_i - n) &= 0 \\ \hat{\mu} &= \frac{\sum_{i=1}^n y_i}{n} \\ &= \bar{y} \end{aligned}$$

A.1.4 Binomial distribution

Suppose the species are counted at sampling location $i = 1, 2, \dots, n$ with multiple counts made during sampling survey $j = 1, 2, \dots, m$ at each location with counts represented by y_{ij} . These counts are viewed as realization of binomial process with index n , which is referred to as abundance and detection probability π_{ij} , which is denoted as

$$y_{ij} \sim \text{binomial}(n, \pi_{ij})$$

where π_{ij} denote the individual detection probability. This model assumes that the population at any given location/site is closed across sampling counts. That is, there is no change in abundance from count to count, which allows estimation of both abundance and detection probability. Detection probability is assumed to be constant. The log link for this model is logit

$$\text{logit}(\pi_{ij}) = \beta_0 + \beta_1 x_{ij}$$

where x_{ij} represents a covariate on detection as measured at location/site i on visit j (Wenger & Freeman 2008).

Parameter estimation for Binomial model

The binomial distribution consists of two parameters: the mean $n\pi_{ij}$ and the variance $N\pi_{ij}(1 - \pi_{ij})$. The log-likelihood function is thus obtained as

$$\begin{aligned}\ell &= \left[\prod_{i=1}^n \binom{n}{x_{ij}} \pi_{ij}^{x_{ij}} (1 - \pi_{ij})^{n-x_{ij}} \right] \\ &= \sum_{i=1}^n \left[\log \binom{n}{x_{ij}} + x_{ij} \log \pi_{ij} + (n - x_{ij}) \log (1 - \pi_{ij}) \right] \\ &= \sum_{i=1}^n x_{ij} \log \pi_{ij} + n - \sum_{i=1}^n x_{ij} \log (1 - \pi_{ij})\end{aligned}$$

The first derivative of the log-likelihood function is defined by

$$\frac{\partial \ell}{\partial \pi_{ij}} = \frac{\sum_{i=1}^n}{\pi_{ij}} - \frac{(n - \sum_{i=1}^n)}{1 - \pi_{ij}}$$

To find the MLE, $\hat{\pi}$, use

$$\frac{\partial \ell}{\partial \pi_{ij}} = 0$$

Thus

$$\begin{aligned}
 \frac{\sum_{i=1}^n x_{ij}}{\pi_{ij}} - \frac{(n - \sum_{i=1}^n x_{ij})}{1 - \pi_{ij}} &= 0 \\
 \frac{\sum_{i=1}^n x_{ij} - n\pi_{ij}}{\pi_{ij}(1 - \pi_{ij})} &= 0 \\
 \sum_{i=1}^n x_{ij} &= n\pi_{ij} \\
 \frac{\sum_{i=1}^n x_{ij}}{n} &= \pi_{ij} \\
 &= \mu
 \end{aligned}$$

The dispersion is modeled using a log link function.

A.1.5 Negative binomial model

Suppose a random variable $Y|\mu$ follows a Poisson distribution with conditional mean $E(Y|\mu = \mu)$ and the parameter, μ , follows a gamma distribution with the mean $E(\mu) = \mu$ and variance $\text{Var}(\mu) = \mu^2\theta^{-1}$. The probability distribution for a Poisson and gamma distributions are given below, respectively

$$\begin{aligned}
 P(Y = y | \mu) &= \frac{\exp^{-\mu} \mu^y}{y!} \text{ and} \\
 g(\mu) &= \frac{\mu^{\theta-1} \exp^{-\frac{\mu}{\beta}}}{\Gamma(\theta)\beta^\theta}, \mu > 0.
 \end{aligned}$$

The joint density of $Y | \mu$ is given by

$$P(Y = y | \mu) g(\mu) = \frac{\exp^{-\mu} \mu^y}{y!} \times \frac{\mu^{\theta-1} \exp^{-\frac{\mu}{\beta}}}{\Gamma(\theta)\beta^\theta}.$$

Now the marginal distribution of Y follows a negative binomial (NB) distribution.

$$\begin{aligned}
 P(Y = y) &= \int_0^\infty P(Y = y|\mu) g(\mu) d\mu \\
 &= \int_0^\infty \frac{\exp^{-\mu} \mu^y}{y!} \frac{\mu^{\theta-1} \exp^{-\frac{\mu}{\beta}}}{\Gamma(\theta)\beta^\theta} d\mu \\
 &= \frac{1}{\Gamma(\theta)\beta^\theta} \int_0^\infty \frac{\exp^{-\mu} \mu^y}{y!} \mu^{\theta-1} \exp^{-\frac{\mu}{\beta}} d\mu \\
 &= \frac{1}{y!\Gamma(\theta)\beta^\theta} \int_0^\infty \mu^{y+\theta-1} \exp^{-\mu(1+\frac{1}{\beta})} d\mu \\
 &= \frac{1}{\Gamma(y+1)\Gamma(\theta)\beta^\theta} \Gamma(y+\theta) \left(\frac{\beta}{\beta+1}\right)^{y+\theta} \\
 &= \binom{y+\theta-1}{y} \left(\frac{1}{\beta+1}\right)^\theta \left(1 - \frac{1}{\beta+1}\right)^y
 \end{aligned}$$

The mean and variance of Y are

$$\begin{aligned}
 E(Y) &= \mu \\
 &= \frac{r(1-p)}{p}
 \end{aligned}$$

and

$$\begin{aligned}
 Var(Y) &= \sigma^2 \\
 &= \frac{r(1-p)}{p^2}
 \end{aligned}$$

respectively, where $r = \theta$ and $p = \frac{1}{\beta+1}$. The NB model has been advocated for modeling datasets with many zeros because of its ability to account for overdispersion (Martin et al. 2005). The NB model has an extra parameter θ that describes dispersion. As θ approaches 0, there is less overdispersion and the NB distribution approaches a Poisson distribution (Cheung 2002).

Parameter estimation for NB model

The NB model consists of two parameters: the mean μ and the dispersion parameter $\mu_i\theta$. Suppose a random variable Y follows a NB distribution and the p.d.f. of Y takes the value y_i is defined as

$$P(Y = y_i) = \frac{\Gamma(y_i + \theta^{-1})}{\Gamma(\theta^{-1}) \Gamma(y_i + 1)} \left(\frac{\mu\theta}{1 + \mu\theta} \right)^{y_i} \left(\frac{1}{1 + \mu\theta} \right)^{\theta^{-1}}$$

for $\theta \geq 0$; $y_i = 0, 1, 2, \dots, n$ where n represents the sample size. The simplest method to estimate θ is to apply the method of moments estimation (MME), the MLE method and the maximum quasi-likelihood estimation method (MQLE). Here the MLE method is used. The likelihood function under the NB model is

$$= \prod_{i=1}^n \frac{\Gamma(y_i + \theta^{-1})}{\Gamma(\theta^{-1}) \Gamma(y_i + 1)} \left(\frac{\mu\theta}{1 + \mu\theta} \right)^{y_i} \left(\frac{1}{1 + \mu\theta} \right)^{\theta^{-1}}$$

where $\Gamma(y)$ is the gamma function, and μ can be modeled as

$$\mu = \begin{cases} \exp(\beta' X_i + \beta_0) & \text{for a log link} \\ \beta' X_i + \beta_0 & \text{for identity link} \end{cases}$$

The NB log-likelihood $L(y_i, \mu_i)$ function is

$$\begin{aligned}
 \ell &= \log \left[\prod_{i=1}^n \frac{\Gamma(y_i + \theta^{-1})}{\Gamma(\theta^{-1}) \Gamma(y_i + 1)} \left(\frac{\mu\theta}{1 + \mu\theta} \right)^{y_i} \left(\frac{1}{1 + \mu\theta} \right)^{\theta^{-1}} \right] \\
 &= \sum_{i=1}^n \log \left[\frac{\Gamma(y_i + \theta^{-1})}{\Gamma(\theta^{-1}) \Gamma(y_i + 1)} \left(\frac{\mu\theta}{1 + \mu\theta} \right)^{y_i} \left(\frac{1}{1 + \mu\theta} \right)^{\theta^{-1}} \right] \\
 &= \sum_{i=1}^n \log \Gamma(y_i + \theta^{-1}) - \sum_{i=1}^n \log \Gamma(y_i + 1) - \sum_{i=1}^n \log \Gamma(\theta^{-1}) + \\
 &\quad \sum_{i=1}^n y_i \log \left(\frac{\mu\theta}{1 + \mu\theta} \right) + \sum_{i=1}^n \theta^{-1} \log \left(\frac{1}{1 + \mu\theta} \right)
 \end{aligned}$$

If the dispersion parameter is considered to be an unknown parameter, a MLE method is used to estimate it. The estimation of dispersion parameter in the NB regression model plays a significant role in various types of count data analysis. The MLE of the dispersion parameter can be found by maximizing the log-likelihood function

$$\ell = \sum_{i=1}^n \left[\sum_j^{y_i-1} \log(1 + \theta j) - y_i \log(\theta) - \log(y_i!) + y_i \log(\theta \mu_i) - (y_i + \theta^{-1}) \log(1 + \theta \mu_i) \right]$$

Then, the MLE can be obtained by solving the following equation

$$\frac{\partial \ell}{\partial \theta} = \frac{\partial}{\partial \theta} \sum_{i=1}^n \left[\sum_j^{y_i-1} \log(1 + \theta j) - y_i \log(\theta) - \log(y_i!) + y_i \log(\theta \mu_i) - (y_i + \theta^{-1}) \log(1 + \theta \mu_i) \right]$$

Now

$$\frac{\partial}{\partial \theta} \sum_{i=1}^n \left[\sum_j^{y_i-1} \log(1 + \theta j) - y_i \log(\theta) - \log(y_i!) + y_i \log(\theta \mu_i) - (y_i + \theta^{-1}) \log(1 + \theta \mu_i) \right] = 0$$

$$\sum_{i=1}^n \left\{ \sum_{j=0}^{y_i-1} \left(\frac{j}{1+\theta j} \right) - \frac{\mu(\theta^{-1} + y_i)}{1+\theta\mu} + \theta^{-2} \log(1+\theta\mu) \right\} = 0$$

Parameter estimates of the NB distribution via MLE method does not result in a closed form solution since the partial derivatives cannot be simplified, we applied the Newton-Raphson iteration method for estimating θ as follows

$$\begin{aligned} \theta^{(1)} &= \theta^{(0)} - \frac{\ell'(\theta^{(0)})}{\ell''(\theta^{(0)})} \\ \theta^{(k+1)} &= \theta^{(k)} - \frac{\ell'(\theta^{(k)})}{\ell''(\theta^{(k)})}, \quad k = 1, 2, \dots, m \end{aligned}$$

where $\theta^{(0)}$ is the initial value of θ given by the researcher, $\theta^{(k)}$ is the value obtained by the k^{th} iteration, ℓ' and ℓ'' denote the first and second derivatives of the log-likelihood function ℓ , respectively. $\ell'(\theta^{(k)})$ and $\ell''(\theta^{(k)})$ are the function values evaluated at $\theta^{(k)}$ where $m = 1\,000$ corresponding to the number of replicates used in the Monte-Carlo simulation. When the convergence $|\theta^{(k+1)} - \theta^{(k)}|$ is smaller than the convergence tolerance, we consider $\theta^{(k+1)}$ as the MLE estimator of the dispersion parameter, θ , in the NB regression model.

A.1.6 Zero-inflated Poisson model

In this section, we briefly review parameter estimation methods concerning the zero-inflated Poisson (ZIP) model. Suppose $y_1, y_2, \dots, y_n$ is a random sample from $\text{ZIP}(\mu_i, \pi_i)$

distribution. The p.d.f. of ZIP is defined as

$$P(Y = y_i | \pi_i, \mu_i) = \begin{cases} \pi_i + (1 - \pi_i) \exp(-\mu_i), & y_i = 0 \\ (1 - \pi_i) \frac{\exp(-\mu_i) \mu_i^{y_i}}{y_i!}, & y_i > 0 \end{cases}$$

where $0 \leq \pi_i \leq 1$ and $\mu_i > 0$ satisfy $\log(\mu_i) = \sum_{j=1}^k x_{ij} \beta_j$ and $\log\left(\frac{\pi_i}{1-\pi_i}\right) = \mathbf{z}_i \boldsymbol{\gamma} + \eta_i$.

Parameter estimation for ZIP model

Suppose that we have independent and identically distributed (i.i.d.) observations $y_1, y_2, \dots, y_n$ from $\text{ZIP}(\mu_i, \pi_i)$. The first objective is to estimate the model parameter estimates using the MLE method. The log-likelihood function of the ZIP is defined as

$$\ell = \sum_{i=1}^n \left\{ I_{(y_i=0)} [\log \pi_i + \log(1 - \pi_i) - \mu_i] + I_{(y_i>0)} [\log(1 - \pi_i) - \mu_i - \log y_i! + y_i \log \mu_i] \right\}$$

where $I_{(y_i)}$ is an indicator function defined on $y_i = 0$ and $y_i > 0$. By taking the derivatives of ℓ with respect to μ_i and π_i and setting them equal to zero. That is

$$\frac{\partial \ell}{\partial \mu_i} = 0$$

The parameter estimates for ZIP model can be found by using Newton-Raphson method, which can converge quickly to true MLE.

A.1.7 Zero-inflated negative binomial model

Let y_i be a non-negative integer random variable and suppose $y_i = 0$ is dominating significantly higher than can be modeled by other models such as Poisson and NB.

Thus, the zero-inflated negative binomial (ZINB) regression model is defined as

$$P(Y = y_i | \pi_i, \mu_i, \theta^{-1}) = \begin{cases} \pi_i + (1 - \pi_i) \left(\frac{\theta}{\theta + \mu_i} \right)^{\theta^{-1}}, & y_i = 0 \\ (1 - \pi_i) \frac{\Gamma(y_i + \theta^{-1})}{\Gamma(y_i + 1) \Gamma(\theta^{-1})} \left(\frac{\theta}{\theta + \mu_i} \right)^{\theta^{-1}} \left(\frac{\mu_i}{\theta + \mu_i} \right)^{y_i} & y_i > 0 \end{cases}$$

where $0 \leq \pi_i \leq 1$, $\mu_i \geq 0$, $\theta^{-1} > 0$ is the dispersion parameter that is assumed not to depend on covariates and $\Gamma(\cdot)$ is the gamma function.

Parameter estimation in the ZINB model

By taking the partial derivatives respect to $\boldsymbol{\theta}$ and setting them equal to zero, we can obtain the parameter estimation of the ZINB. Thus we obtain

$$\ell(\boldsymbol{\theta}) = \begin{cases} \left[\log \pi_i + \log(1 - \pi_i) - \frac{1}{\theta} \log \left(\frac{1}{1 + \mu_i \theta} \right) \right]_{I_{y_i=0}} + \\ \left[\log(1 - \pi_i) + \log \Gamma \left(y_i + \frac{1}{\theta} \right) - \log \Gamma(y_i + 1) - \Gamma \left(\frac{1}{\theta} \right) - \frac{1}{\theta} \log \left(\frac{1}{1 + \mu_i \theta} \right) + y_i \log \left(\frac{\mu_i \theta}{1 + \mu_i \theta} \right) \right]_{I_{y_i>0}} \end{cases}$$

The partial derivative is defined as

$$\frac{\partial \ell}{\partial \theta} = 0$$

This parameter estimation can be solved by using Newton-Raphson algorithm.

Newton-Raphson algorithm for ZINB

The parameters μ_i and π_i are commonly applied in practical applications and are assumed to depend on vectors of explanatory variables $\mathbf{x}_i$ and $\mathbf{z}_i$, respectively (Araújo

et al. 2021). Here, we assume that in the models

$$\begin{aligned} \log(\mu_i) &= \mathbf{x}_i' \boldsymbol{\beta} & \text{and} \\ \log\left(\frac{\pi_i}{1 - \pi_i}\right) &= \mathbf{z}_i' \boldsymbol{\gamma}, & i = 1, \dots, n, \end{aligned}$$

where $\boldsymbol{\beta} = (\mathbf{x}_i') \boldsymbol{\beta}$ and $\boldsymbol{\gamma} = (\gamma_1, \dots, \gamma_q)'$ are unknown parameters (Aráujo et al. 2021). Suppose $(y_1, \mathbf{x}_1, \mathbf{z}_1), \dots, (y_n, \mathbf{x}_n, \mathbf{z}_n)$ are n independent observations where each observed response variable is denoted by y_i . Then, the log-likelihood function for the vector of parameters $\boldsymbol{\theta} = (\phi, \boldsymbol{\beta}', \boldsymbol{\gamma}')'$ for a given observed sample, so given by Aráujo et al. (2021) as

$$\begin{aligned} \ell(\boldsymbol{\theta}) &= \sum_{i=1}^n \log \left[\pi_i + (1 - \pi_i) \left(\frac{\theta}{\mu_i + \theta} \right)^{\theta^{-1}} \right]_{I_{(y_i=0)}} \\ &+ \sum_{i=1}^n \log \left[(1 - \pi_i) \left(\frac{\Gamma(y_i + \theta^{-1})}{\Gamma(y_i + 1) \Gamma(\theta^{-1})} \right) \left(\frac{\mu_i}{\mu_i + \theta} \right)^{y_i} \left(\frac{\theta}{\mu_i + \theta^{-1}} \right)^{\theta^{-1}} \right]_{I_{(y_i>0)}} \\ &= \sum_{i=1}^n \left[l_1(\theta^{-1}, \mathbf{x}_i', \boldsymbol{\beta}, \mathbf{z}_i', \boldsymbol{\gamma})_{I_{(y_i=0)}} + l_2(\theta^{-1}, \mathbf{x}_i', \boldsymbol{\beta}, \mathbf{z}_i', \boldsymbol{\gamma})_{I_{(y_i>0)}} \right] \end{aligned}$$

where I_τ is the indicator function. The ML estimate of $\hat{\boldsymbol{\theta}}$ of the vector of unknown parameters can be calculated by maximizing the log-likelihood given by $\ell(\boldsymbol{\theta})$. Since the maximization of the log-likelihood in the ZINB does not converge, we use the expectation-maximization (EM) algorithm. What EM algorithm does is to estimate the parameters directly by maximizing the complete data log-likelihood function. This algorithm consists of two steps: the E-step and M-step. The E-step calculates the expectation of the log-likelihood function of the parameters for a given dataset. The M-step is obtained by maximizing the expectation of the complete data log-likelihood from the previous E-step. The EM algorithm $\mathbf{w} = (w_1, \dots, w_n)'$, the complete data

log-likelihood associated with $\mathbf{Y}_c = (\mathbf{y}, \mathbf{w})$ is the log-likelihood defined by ? as

$$\ell_c(\boldsymbol{\theta}|\mathbf{Y}_c) = \sum_{i=1}^n \{w_i \mathbf{z}_i \boldsymbol{\gamma} - \log[1 + \exp(\mathbf{z}_i \boldsymbol{\gamma})] + (1 - w_i) \log[g(y_i; \boldsymbol{\beta}, \theta^{-1})]\}$$

where

$$g(y_i; \boldsymbol{\beta}, \theta^{-1}) = \frac{\Gamma(y_i + \theta^{-1})}{\Gamma(y_i + 1) \Gamma(\theta^{-1})} \left(\frac{\mu_i}{\mu_i + \theta} \right)^{y_i} \left(\frac{\theta}{\mu_i + \theta} \right)^{\theta^{-1}}$$

with $\mu_i = \exp(\mathbf{x}_i' \boldsymbol{\beta})$. Also note that $\ell_c(\boldsymbol{\theta}|\mathbf{y}, \mathbf{z})$ has a particularly conveniently form for the EM algorithm, for example, $\ell_c(\boldsymbol{\theta}|\mathbf{y}, \mathbf{z})$ is linear in $\mathbf{w}$, so at iteration (k) of the algorithm, the E-step consists of replacing $\mathbf{w}$ by its conditional expectation given $\mathbf{y}$, $\hat{\boldsymbol{\beta}}^{(k)}$ and $\hat{\boldsymbol{\gamma}}^{(k)}$ (Garay et al. 2011). The conditional expectation is easily calculated as

$$\hat{w}_i^{(k)} = \begin{cases} \left[1 + \exp(-\mathbf{z}_i \hat{\boldsymbol{\gamma}}^{(k)}) \left(\frac{\hat{\phi}^{(k)}}{\exp(\mathbf{x}_i \hat{\boldsymbol{\beta}}^{(k)}) + \hat{\phi}^{(k)}} \right)^{\hat{\phi}^{(k)}} \right]^{-1} & \text{if } y_i = 0, \\ \text{if } y_i > 0 \end{cases}$$

and so

$$Q(\boldsymbol{\theta}|\hat{\boldsymbol{\theta}}^{(k)}) = E[\ell_c(\boldsymbol{\theta}|\mathbf{Y}_c) | \mathbf{y}, \hat{\boldsymbol{\theta}}^{(k)}] = \sum_{i=1}^n Q_{1i}(\boldsymbol{\gamma}|\hat{\boldsymbol{\theta}}^{(k)}) + \sum_{i=1}^n Q_{2i}(\boldsymbol{\beta}, \phi|\hat{\boldsymbol{\theta}}^{(k)})$$

where

$$Q_{1i}(\boldsymbol{\gamma}|\hat{\boldsymbol{\theta}}^{(k)}) = \hat{w}_i^{(k)} \mathbf{z}_i \boldsymbol{\gamma} - \log[1 + \exp(\mathbf{z}_i \boldsymbol{\gamma})]$$

and

$$Q_{2i}(\boldsymbol{\beta}, \phi|\hat{\boldsymbol{\theta}}^{(k)}) = (1 - \hat{w}_i^{(k)}) \log \left\{ \frac{\Gamma(y_i + \phi)}{\Gamma(y_i + 1) \Gamma(\phi)} \left(\frac{\exp(\mathbf{x}_i' \boldsymbol{\beta})}{\exp(\mathbf{x}_i' \boldsymbol{\beta}) + \phi} \right)^{y_i} + \left(\frac{\phi}{\exp(\mathbf{x}_i' \boldsymbol{\beta}) + \phi} \right)^\phi \right\}$$

Now $Q(\boldsymbol{\theta}|\hat{\boldsymbol{\theta}}^{(k)})$ is easily maximized (the M-step) with respect to $\boldsymbol{\gamma}$ and $(\boldsymbol{\beta}, \phi)$ because $\sum_{i=1}^n Q_{1i}(\boldsymbol{\gamma}|\hat{\boldsymbol{\theta}}^{(k)})$ is equal to the log-likelihood for an unweighted binomial logistic

regression of $\hat{\mathbf{w}}^{(k)}$ on $\mathbf{z}$ and $\sum_{i=1}^n Q_{2i}(\boldsymbol{\beta}, \phi | \hat{\boldsymbol{\theta}}^{(k)})$ (Garay et al. 2011).

A.2 Machine learning algorithms

There are three types of machine learning (ML) algorithms: supervised learning, unsupervised learning and reinforcement learning. The scope of this study is limited to supervised machine learning methods. The supervised learning consists of outcome variable or dependent variable which is to be predicted from a given set of predictors (independent variables). These sets of variables are used to generate functions that map inputs to desired outputs. Examples of supervised learning include regression, decision tree (DT), classification and regression trees (CART), random forests (RF), k-nearest neighbors (k NN), gradient boosting machines (GBM), support vector machines (SVM), etc. These techniques provide a powerful set of tools to deal with ecological problems.

A.2.1 Decision trees

Decision trees are supervised learning models designed to separate data into consistent clusters for classification or regression analysis. It consists of a root node, leaf node, internal nodes and edges (Leke & Marwala 2019). Each node represents a different tree. As such, the root node denotes the onset of the trees with the leaf nodes representing the end of the tree which either represents the outcome or the class label. The internal nodes supply details about the attribute used for splitting the data at each node. The edges are links between the nodes and contain details about splits (Marwala 2009).

Decision trees are robust to errors and if the training data contains errors, decision tree algorithms will be best suited to address such problems. Decision trees are best

suited for problems where instances are represented by attribute value pairs. If the training data has missing value then decision trees can be used, as they can handle missing values nicely by looking at the data in other columns. Decision trees are best suited when the target function has discrete output values. Many decision trees can be combined using methods like bagging, random forests and boosting to improve the predictive performance of trees. Classification and regression trees are collectively called decision trees (Olden et al. 2008).

A.2.2 Classification and regression trees

Classification and regression (CART) are machine learning algorithm that uses the values of a set of observed predictor variables to split a data set into homogeneous subgroups with respect to a categorical or continuous response variable (Hayes & McArdle 2017). Cutler et al. (2007) explains classification and regression (CART) as the most popular tree-based method. Tree-based methods such as classification and regression trees are one of the reasons that the trees provide interpretable results. Classification trees are created when the response variable is for discrete and used to predict a categorical response (Kampichler et al. 2010). Regression trees are produced when the response variable is for continuous response (Cutler et al. 2007). Classification trees will be considered because of their simple interpretation, high classification accuracy and ability to characterize complex interactions among variables. CART models provides interpretable results, since the tree can be understood as a series of logic rules, “if $x_1 < \dots$, then...” and their ability to characterize complex interactions among variables. However, they do not have high classification accuracy due to the greedy nature of tree construction. The advantage of CART includes robustness to outliers in the output and are very easy to understand.

The tree is developed from the repetition partitioning of the original data set into two subsets that are more homogeneous than the first one, leading to a branching

structure (Crisci et al. 2012). The split of the data is done via a binary rule over the explanatory variables. When the data has continuous variables, the split has the form $x < s$ where s is a threshold over the variable x (Crisci et al. 2012). CART uses the Gini index to choose the split that maximizes the decrease in impurity at each node. If $p(i | j)$ is the probability of class i at node j , then the Gini index is $1 - \sum_i p^2(i | j)$. When CART grows a maximal tree, the tree is pruned upward to get a decreasing sequence of sub-trees. Then, cross-validation is used to identify the sub-tree that is having the lowest estimated misclassification rate.

A.2.3 Random Forests

Random forests (RF) are defined by Wang et al. (2017) as a classic and popular ensemble-based technique which aggregate independent decision trees for classification. RF is also a new entry to the field of machine learning and is designed to produce accurate predictions that do not overfit the data (Cutler et al. 2007). RF methods can analyze missing data in ecological studies. Cutler et al. (2007) identified RF as very high in classification accuracy, ability to model complex interactions among predictor variables, flexibility to perform several types of statistical data analysis including regression, classification and unsupervised learning methods. Random forests work as an algorithm for imputing missing observations (Cutler et al. 2007). RF is advanced by a procedure called bagging, which is short for bootstrap aggregating where each tree is independently constructed by using a bootstrap sample of the whole data set (Kampichler et al. 2010).

In particular, RF consist of a set of decision trees, either classification or regression trees, depending on the problem being addressed (Torgo 2011). RF is the go-to machine learning algorithm that uses a bagging approach to create a bunch of decision trees with random subset of the data. A model is trained several times on random sample of the data set to achieve good prediction performance from the RF algorithm.

In this ensemble learning method, the output of all the decision trees in the random forest, is combined to make the final prediction. The final prediction of the RF algorithm is derived by polling the results of each decision tree or just by going with a prediction that appears the most times in the decision trees. One of the main advantages of RF is the resistance to over training and growing a large number of random forest trees where it does not create a risk of overfitting (for example, each tree is a completely independent random experiment). The RF algorithm data does not need to be rescaled, transformed, or modified. It has resistance to outliers in predictors and automatically handles the missing values (Cutler et al. 2007).

The RF is more formally, for a p -dimensional random vector $\mathbf{X} = \{X_1, X_2, \dots, X_p\}^T$ representing the real-valued input or predictor variables and a random variable Y representing the real-valued response. The goal of a RF algorithm is to find a prediction function $f(X)$ for predicting Y . The prediction function is determined by a loss function $L(Y, f(X))$ and defined to minimize the expected value of the loss

$$E_{XY}(L(Y, f(X)))$$

where the subscripts denote expectation with respect to the joint distribution of X and Y . Intuitively, $L(Y, f(X))$ is a measure of how close $f(X)$ is to Y ; it penalizes values of $f(X)$ that are a long way from Y . Typical choices of L are *squared error loss* $L(Y, f(X)) = (Y - f(X))^2$ for regression and *zero-one loss* for classification:

$$L(Y, f(X)) - I(Y \neq f(X)) = \begin{cases} 0 & \text{if } Y = f(X) \\ 1 & \text{otherwise} \end{cases}$$

It turns out that minimizing $E_{XY}(L(Y, f(X)))$ for squared error loss gives the conditional expectation

$$f(x) = E(Y|X = x)$$

otherwise known as the *regression function*. In the classification situation, if the set

of possible values of Y is denoted by Y , minimizing $E_{XY}(Y, f(X))$ for zero-one loss gives

$$f(x) = \arg \max P(Y = y|X = x)$$

otherwise known as the *Bayes rule*. Ensembles construct f in terms of a collection of so-called "base learners" $h_1(x), \dots, h_j(x)$ and these base learners are combined to give the "ensemble predictor" $f(x)$. In regression, the base learners are averaged

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

while in classification, $f(x)$ is the most frequently predicted class

$$f(x) = \arg \max \sum_{j=1}^J I(y = h_j(x))$$

In RF, the j^{th} base learner tree is denoted by $h_j(X, \Theta_j)$, where Θ_j represent a collection of random variables and the Θ_j 's are independent for $j = 1, 2, \dots, J$.

A.2.4 Support vector machines

The support vector machine (SVM) is a widely used tool for classification and regression model that is capable of solving both linear and non-linear complex problems (Pelckmans et al. 2005). The SVM algorithm is known by its feature time, which maps the input feature space to a high-dimensional space by non-linear transformation and finds the optimal linear boundary hyper-plane in the high-dimensional special space (Muhamedyev 2015). The focus of the linear SVM is on maximizing the margin $\|\alpha\| = \sqrt{\sum_{i=1}^n \alpha_i^2}$ between the negative and positive hyper-planes to find the greatest margin.

In non-linear complex problems, where the data dimensions are said to be linearly

inseparable, the original data is transformed into a new higher dimension (Drake et al. 2006). In the new dimension, the technique finds an optimal hyper-plane that separates the attributes of the classes (Lo & Marculescu 2019). The performance of classes is determined by assigning the following equation:

$$y = \begin{cases} +1, & \text{if } b + \boldsymbol{\alpha}^T \mathbf{x} \geq +1 \\ -1, & \text{if } b + \boldsymbol{\alpha}^T \mathbf{x} \leq -1 \end{cases}$$

where $\mathbf{x}$ and $\boldsymbol{\alpha}$ define the input vector and the weight vector, respectively, and b denotes the bias (Dastile et al. 2020). For classes that are inherently not separable, the decision boundary can be regarded as having a soft-margin. SVM can also solve non-linear classification problems with mapping the input vectors to a higher dimensional space using kernel functions $k(x_i, x_j) = \langle \phi(x_i), \phi(x_j) \rangle$ (Zararsiz et al. 2012, Lo & Marculescu 2019). There are four kernel function groups that are commonly used in SVM. The most widely used kernel functions are linear: $K(x, y) = x_i \cdot x_j$, polynomial: $K(x, y) = (x_i \cdot x_j + 1)^d$, radial basis function: $K(x, y) = \exp(-\gamma \|x_i - x_j\|^2)$ and sigmoidal: $K(x, y) = \tanh(k(x_i \cdot x_j) - c)$, where d is the degree, $\gamma > 0$ sometimes parameterized as $\gamma = \frac{1}{2}\sigma^2$, and c is a constant. The decision function can be written as (Zararsiz et al. 2012)

$$f(x) = \text{sign} \left(\sum_{i=1}^p y_i \alpha_i k(x, x_i) + b \right).$$

SVM basically develops separate hyper-planes to enhance the classifying margin between positive datasets and negative datasets. SVM first constructs a hyper-plane according to training dataset, and then maps an input vector into higher dimensional vector space. The mapping of input vector from non-linear space to a higher dimensional space is demonstrated in Figure A.1. The nearest two points to the hyper-plane in a pair are called support vectors. A key factor or attribute of SVM is that it takes only fixed length of the input vector. The dataset in training set and test set needs to be transformed into feature vectors, then training vectors are fed to SVM for con-

structing classifier (Pelckmans et al. 2005). SVM perform classification by separating

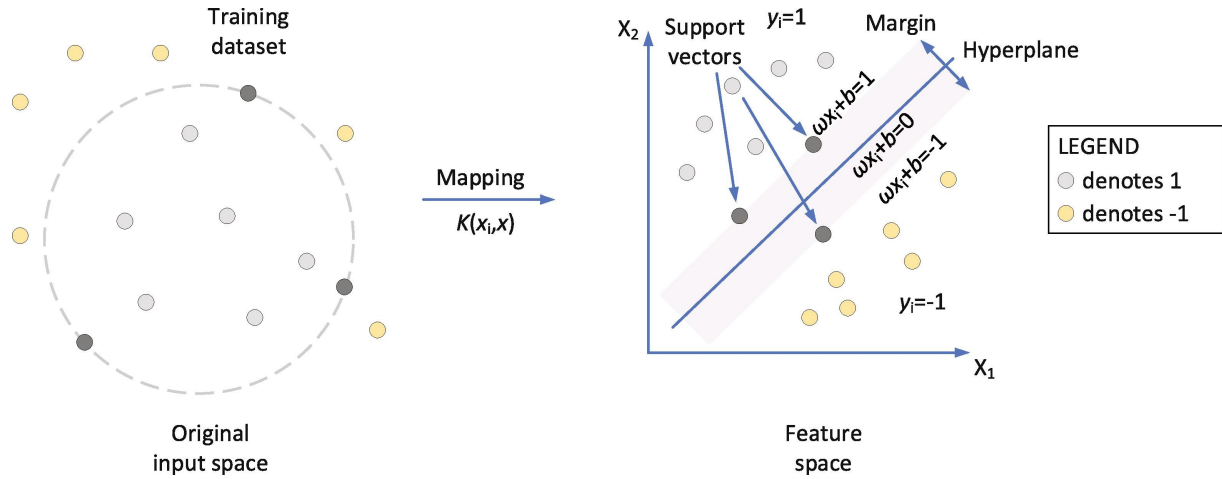


Figure A.1: Support vector machine for non-linear separable data.

different classes in the data using a maximal margin hyperplane. To learn non-linear decision boundaries, SVM implicitly map data to a higher dimensional space by means of a kernel function, where a separating hyperplane is then sought. Additional parameters and constraints are added to the optimization problem to handle the separation of the different classes. SVM have been shown to perform well in practice for both binary and multi-class classification problems. The superior empirical performance of SVM in many types of high-throughput biomedical data can be explained by several theoretical reasons: SVM are robust to high variable-sample ratios and large number of features; they can efficiently learn complex classification functions and employ powerful regularization principles to avoid data overfitting (Pelckmans et al. 2005). It is among the most robust and successful classification algorithms. The best thing about SVM is that it does not make any strong assumptions on data. However, the imbalanced data problem is still a challenge in numerous research fields. Hence ensemble techniques are suggestive in a broad spectrum of scenarios in order to improve the performance of weak classifiers. SVM in the ensemble learning is composed of few or even none of the minority instances. Basically each component learner of the ensemble will have severe skewness and improvement using ensemble technique

would be limited, hence the proposition of ensemble of SVM. In the ensemble SVM all the individual SVM classifiers are used to reach a final classification decision. In many cases, a subset of the classifier is used which may lead to better generalization capabilities due to possible reductions in the coupling of individual classifiers. Here we show why the use of only a subset of the classifier in an ensemble could lead in better generalization results. Suppose there are L test instances with the target variable $y_1, y_2, \dots, y_L$, where $y_j \in \{-1, +1\}$ denotes the true class label. Let the actual output of the i^{th} SVM classifier on the j^{th} instance be $\hat{y}_{ij} = \text{sgn} \left(\sum_{i=1}^M \hat{y}_{ij} \right)$, where M represent the number of component classifiers in the ensemble, and $\text{sgn}()$ denote a signum function that takes the value of 1 if $x > 0$, 0 if $x = 0$, and -1 if $x < 0$.

Now, considering the generalization error of classifiers on an instance to be 0 if the actual output agrees with the target variable, 1 if they are completely opposite (for example, -1 versus +1), and $\frac{1}{2}$ if the actual variable is 0, then the generalization error of the ensemble on the j^{th} instance can be defined as

$$\epsilon_j = \frac{1}{2} | \hat{y}_j - y_j |$$

Thus the total generalization error of the ensemble is

$$E = \frac{1}{2} \sum_{j=1}^L | \hat{y}_j - y_j |$$

Considering what will happen if one of the component SVM classifiers are excluded from the ensemble. For the j^{th} instance, if $|\sum_{i=1}^M \hat{y}_{ij}| > 1$, then removing one component classifier from the ensemble will not affect the final output of the ensemble. Looking at the case where $|\sum_{i=1}^M \hat{y}_{ij}| \leq 1$. Suppose the k^{th} classifier is excluded from the ensemble. Then the total generalization error becomes

$$E_k = \frac{1}{2} \sum_{j=1}^L \left| \text{sgn} \left(\sum_{i=1, i \neq k}^M \hat{y}_{ij} \right) - y_j \right|$$

From the equations, we can see that $E \geq E_k$. When all the classifiers are exactly the same, $E = E_k$; therefore, the generalization capability of the ensemble is unaffected by the exclusion of the k^{th} component. Consider the case where $L = 1, y_1 = 1, \sum_{i=1}^M \hat{y}_{i1} = 0$, and $\hat{y}_{k1} = -1$. Then using properties of the sgn function, we can show that

$$E - E_k = \frac{1}{2} \left(sgn \left(\sum_{i=1}^M \hat{y}_{i1} - \hat{y}_{k1} \right) \right) - sgn \left(\sum_{i=1}^M \hat{y}_{i1} \right) = \frac{1}{2}$$

That is, the generalization error can be reduced if the k^{th} component classifier is excluded.

A.2.5 Artificial neural networks

Artificial neural networks (ANN) are powerful information processing models inspired by the way the human nervous system (e.g. human brain) communicates information (?). Figure A.2 present a graphical illustration of a neural network which is composed of neurons and interconnections. ANN methods are general classifiers that can handle problems with lots of parameters and can classify objects, even when the distribution of object in n -dimensional parameter space is very complex. ANN are non-linear models that make them flexible in modeling real world complex relationships. Neural networks have an inability to handle mixed data well, have no integrated procedure for handling missingness, can be biased easily due to outliers, do not scale well, are not interpretable and they do not handle noisy input variables with any degree of intelligence. Beyond these shortcomings, ANN are believed to be highly prone to overfitting the training dataset. The overfitting on neural networks is an artifact of the large number of parameters used to construct the classifier (Wang et al. 2017). Basically, convolution neural networks can reduce the number of parameters to be learned by using specific insights about the domain at hand and can help in improving the accuracy of the model. The most widely used ANN is the multi-layer perceptron (MLP) due to its efficiency and easy training process. Artificial neural network (ANN)

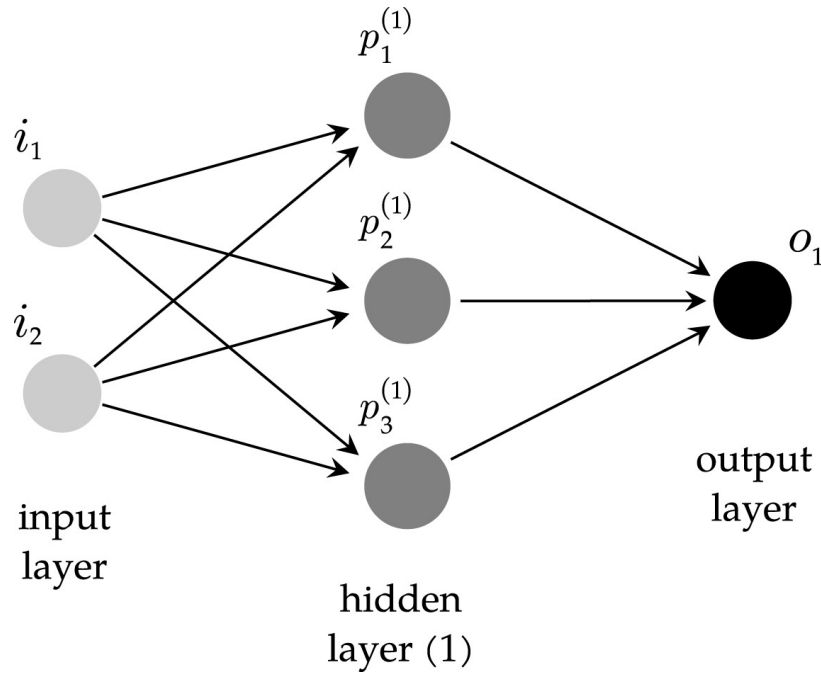


Figure A.2: A representation of an artificial neural network.

algorithms are general classifiers that can handle problems with lots of parameters and can classify objects even when the distribution of objects in n -dimensional parameter space is very complex (Marivate et al. 2008). The architecture of ANN is based on the MLP (Youssef & Pourghasemi 2021). The MLP consists of a set of layers such as the input layer, which receives independent variables, the output layer, where the results are found and the hidden layer, where the processing occurs (Thessen 2016) as demonstrated in Figure A.2. Each layer is made up of several nodes (neurons). The number of neurons in the hidden layer can be optimized to avoid overfitting. The purpose of an ANN is to build a model of the data generating process, so that the network can generalize and predict outputs from inputs that it has not previously seen (Olaya-Marín et al. 2013, Youssef & Pourghasemi 2021). The ANN algorithm has two useful modes; the learning mode and the recall mode. The learning mode is carried out by back-propagation algorithm, is used in providing a network which has initially random connection weights with a set of stimulus couples (Youssef &

Pourghasemi 2021). Recall allows for the application of the synthesized knowledge from the preceding learning phase (?). This allows the neural network to be able to provide coherent answers to input which is not presented in the training set (Youssef & Pourghasemi 2021). The most elementary model of the brain is one that consists of many interconnected neurons. Each neuron takes inputs from many other neurons and has one output, which may in turn be connected to another neuron.

A.2.6 k-nearest neighbors

The k -nearest neighbor (k NN) is a ML algorithm that searches a given space for the k training attributes that are closest to the unidentified target attribute (Muhamedyev 2015). The k smallest distances are identified and the most representative class in these k classes is considered the output class label. The nearest neighbors are calculated based on the Euclidean distance between the input feature vector and the training dataset $\{\mathbf{x}_k\}$, where $k = 1, \dots, m$. Hence the class for the new data point is

$$y = \text{majority}_{y \in [+1, -1]} [\arg \min || \{\mathbf{x}_k - \mathbf{x}\} ||]$$

The value of k is normally determined using cross-validation, the process begins with the smallest value, $k = 1$, and gradually increase k to allow for more neighbors to reduce the minimum error rate. The k NN algorithm is used to classify a given data point to a label based on the nearest data points to the test data points.

Suppose for this technique weights are been chosen individually for every object (Muhamedyev 2015). If weights are not the same, instead of calculating the number of objects, their weights can be added together. Thus, if the sphere around the recognized object contains 10 reference objects of class A with the weight 2 and 15 error/boundary objects of Class B with weights 1, the point will be referred to Class A. Weights of objects in the sphere can be expressed as inversely proportional to their distance to the recognized object. Thus, the closer is an object is, the more significant

it is for this recognized object. In total, a metric classifier can be described as:

$$a(u; X^l = \arg \max_{y \in Y} \sum_{i=l}^L [y_u^{(i)} = y] w(i, u))$$

where $w(i, u)$ - denote a weigh of an i -neighbor of a recognized object u , a $(u; X^l)$ - is the class of an object u , recognized by selection X^l . The k NN algorithm is based on calculation of the number of objects in each class of the sphere with the centre in the recognized object. It attempt to find the class membership of an unknown instance in the testing dataset on the basis of the majority vote of the k NN of that particular instance present in the training dataset. The advantages of applying k NN algorithm to a dataset are that it is a simple classifier and can work well for noisy dataset. However, the disadvantage is that it requires considerable computational time as the distance metric is computed for every instance in the training dataset. The limitation of k NN is that it is very sensitive to the small number of classes (Jain & Kapoor 2009).

A.2.7 Naive Bayes

Naive Bayesian is a probabilistic classifier based on applying Baye's theorem with strong (naive) independent assumptions between the features. This assumption is motivated by the need to estimate the multivariate probabilities from the training dataset. A Naive Bayesian model is easy to build, with no complicated iterative parameter estimation which makes it particularly useful in the field of medical science for diagnosing heart patients. Naive Bayesian networks are very simple Bayesian networks which are composed of directed acyclic graphs with only one parent and several children with strong assumption of independence among child nodes in the

context of their parent. Thus, the independence model (NB) is based on estimating:

$$P(c_j|x_1, x_2, \dots, x_n) = \frac{P(x_1, x_2, \dots, x_n|c_j) P(c_j)}{P(c_1, c_2, \dots, c_n)}$$

where $P(c_j)$ denotes the probability that a random sample belongs to category c_j (Feng et al. 2016). $P\{x_1, x_2, \dots, x_n\}$ is the probability that category c_j contains the feature vector $x = (x_1, x_2, \dots, x_n)$ if we already know the training sample is in c_j . $P(c_1, c_2, \dots, c_n)$ denotes the joint probability of all possible categories. For all the given categories, the denominator $P(c_1, c_2, \dots, c_n)$ is a constant, hence

$$c_{NaiveBayesian} = \arg \max_{c_j \in C} P(x_1, x_2, \dots, x_n|c_j) P(c_j)$$

According to the assumption of the Naive Bayes theorem, the terms in the feature vector are identically distributed. Therefore, their joint probability distribution is equal to the production of every terms probability. Therefore, we have

$$P(x_1, x_2, \dots, x_n|c_j) = \prod_{i=1}^n P(x_i|c_j)$$

It means that we are only interested in finding

$$c_{NB} = \arg \max_{c_j \in C} P(c_j) \prod_{i=1}^n P(x_i|c_j)$$

In real dataset, there are always many dependent vectors presented in the training set (Feng et al. 2016). The algorithms performs positively with categorical dataset but poorly with numerical dataset in the training set. Naive Bayesian is naturally robust to missing values since these are simply ignored in computing probabilities and hence have no impact on the final decision. In general, this classifier is unsuited to imbalanced dataset since it takes into account the prior probabilities of classes in the training data when classifying a new instance.

A.2.8 The boosting algorithms: GBM and XGB

A gradient boosting machine (GBM) is an ensemble algorithm of regression or classification trees. The aim of GBM is to improve accuracy in which every new tree tries to correct the errors of previously built trees. Boosting in weak classifiers such as classification and regression trees are sequentially combined into a strong learner using an additive technique, $\hat{y}_i^{(t)} = \sum_{k=1}^n f_k(x_i) = \hat{y}_i^{(t+1)} + f_t(x_i)$, where each $f_t(x_i)$ is a tree based prediction. In the boosting technique, every step the objective function to be optimized aims to reduce the discrepancy between the outcome and the prediction at the previous step, $obj^t = \sum_{i=1}^n \ell(y_i, \hat{y}_i^{(t)}) + \sum_{i=1}^t \omega(f_i) = \sum_{i=1}^n \ell(y_i, \hat{y}_i^{(t+1)}) + f_t(x_i) + \omega(f_t) + constant$, where ω is a regularization function. The loss function for this technique can be any generic loss function, does not necessarily have to be the classic Mean Squared Error (MSE). Gradient based numerical optimization is used to obtain the model loss.

The eXtreme gradient boosting (XGB) algorithm has gained popularity among data scientists for its faster and better performing boosting algorithm. The function to be optimized allows for regularization, algorithms can be naturally parallelized and cross-validation can be performed at every step. [Muñoz-Mas et al. \(2019\)](#) compared the performance of GBM and XGB by modeling the habitat suitability for the invasive bleak (*Alburnus alburnus* L.) and pumpkinseed (*Lepomis gibbosus* L.). XGB has four possible types of boosters and each one comes with dedicated parameters ([Spedicato et al. 2018](#)).

A.3 Model evaluation measures

The model evaluation measures are used to determine the model performance. Comparing different machine learning (ML) classifiers using performance metrics is easy, but this has many problems such as sensitivity to imbalanced dataset and ignoring

the performances of some classes. Based on the literature regarding performance evaluation measures in the presence of multi-class imbalanced data problem, accuracy is the most commonly used performance measure. However, for a classification model trained from a dataset with imbalanced classes, accuracy is no longer regarded as a proper measure since rare classes have very few impact on accuracy than prevalent classes. There are widely accepted performance measures in multi-class imbalanced learning, that are computed from the confusion matrix. The most extensively used measures includes recall, precision, F1-score, G-mean and Receiver Operation Characteristic (ROC) Curve Analysis, are then recommended and proposed as more advanced evaluation measures (Sun et al. 2006). ROC curves are insensitive with the imbalanced data. Precision-recall curves are preferred over ROC curve for evaluating performance in the presence of imbalanced data (Kassambara 2018).

When dealing with binary data, the receiver operating characteristic (ROC) curve is used to measure the performance of a classifier. The area under the ROC curve (AUC) is used as a metric for predicting the overall performance of the model where a value of 1 indicates a perfect model and a value of 0.5 implies the model predicts no better chance (Kassambara 2018, Hegel et al. 2010). The Kappa statistic and ROC curve are the most commonly used evaluation metrics in ML models (Cushman & Huettmann 2010). The correlation between these two measures gets used to describe the accuracy of the model. The ROC curve is defined as sensitivity plotted against 1-specificity (Hegel et al. 2010). Sensitivity measures the proportion of true positives (TP) whereas specificity measures the proportion of false negatives (FN). The performance metrics of a classifier will be determined using a point system. A value above 0.9 will indicate excellent performance, 0.8 – 0.9 will indicate a good performance, 0.7 – 0.8 will indicate fair performance and a value 0.6 – 0.7 will indicate poor performance and any value that is below 0.6 will indicate very poor performance or a fail. Kassambara (2018) and Cushman & Huettmann (2010) gave a detailed analysis on ROC curve as a measure of model performance. The major limitations of the ROC is that it is only suitable for discrete data and used for validating not more than two

classes (Cushman & Huettmann 2010). Further investigations are needed to explain what problems multi-class imbalanced data can cause in the existing class balanced techniques and how it affects the performance of classifiers. Such information can be informative to understand the multi-class imbalanced data problem better and can assist in developing better solutions.

Appendix B. Appendix



Count Regression and Machine Learning Techniques for Zero-Inflated Overdispersed Count Data: Application to Ecological Data

Bonelwa Sidumo¹ · Energy Sonono¹ · Isaac Takaidza¹

Received: 22 October 2021 / Revised: 17 January 2023 / Accepted: 21 March 2023
© The Author(s) 2023

Abstract

The aim of this study is to investigate the overdispersion problem that is rampant in ecological count data. In order to explore this problem, we consider the most commonly used count regression models: the Poisson, the negative binomial, the zero-inflated Poisson and the zero-inflated negative binomial models. The performance of these count regression models is compared with the four proposed machine learning (ML) regression techniques: random forests, support vector machines, k –nearest neighbors and artificial neural networks. The mean absolute error was used to compare the performance of count regression models and ML regression models. The results suggest that ML regression models perform better compared to count regression models. The performance shown by ML regression techniques is a motivation for further research in improving methods and applications in ecological studies.

Keywords Count data · Ecology · Machine learning · Overdispersion · Zero-inflation

Energy Sonono and Isaac Takaidza have contributed equally to this work.

Bonelwa Sidumo
Bonelwa.Sidumo@nwu.ac.za

Energy Sonono
Energy.Sonono@nwu.ac.za

Isaac Takaidza
Isaac.Takaidza@nwu.ac.za

¹ School of Mathematical and Statistical Sciences, North-West University, Hendrick Van Eck Blvd, Vanderbijlpark 1911, Gauteng, South Africa



An approach to multi-class imbalanced problem in ecology using machine learning

Bonelwa Sidumo , Energy Sonono, Isaac Takaidza

School of Mathematical and Statistical Sciences, North-West University, Vanderbijlpark Campus 1911, Gauteng, South Africa

ARTICLE INFO

Keywords:

Ecology
Ensemble classifiers
Imbalanced dataset
Machine learning
Multi-class classification
Multiple species

ABSTRACT

Ecologists collect their data manually by visiting multiple sampling sites. Since there can be multiple species in the multiple sampling sites, manually classifying them can be a daunting task. Much work in literature has focused mostly on statistical methods for classification of single species and very few studies on classification of multiple species. In addition to looking at multiple species, we noted that classification of multiple species result in multi-class imbalanced problem. This study proposes to use machine learning approach to classify multiple species in population ecology. In particular, bagging (random forests (RF) and bagging classification trees (bag-CART)) and boosting (boosting classification trees (bootCART), gradient boosting machines (GBM) and adaptive boosting classification trees (AdaBoost)) classifiers were evaluated for their performances on imbalanced multiple fish species dataset. The recall and F1-score performance metrics were used to select the best classifier for the dataset. The bagging classifiers (RF and bagCART) achieved high performances on the imbalanced dataset while the boosting classifiers (bootCART, GBM and AdaBoost) achieved lower performances on the imbalanced dataset. We found that some machine learning classifiers were sensitive to imbalanced dataset hence they require data resampling to improve their performances. After resampling, the bagging classifiers (RF and bagCART) had high performances compared to boosting classifiers (bootCART, GBM and AdaBoost). The strong performances shown by bagging classifiers (RF and bagCART) suggest that they can be used for classifying multiple species in ecological studies.

1. Introduction

In population ecology, classification of species from multiple sampling sites has been of great interest as it is used to study species presence or absence. The classification of various species can further give valuable insights about the health of the ecological system and can be used as a parameter for monitoring environmental changes. Ecologists collect the data manually (by visiting the multiple sampling sites numerous times) and since there can be many species in the multiple sampling sites, identifying them manually can be a daunting task. For instance, classifying fish species from count data in natural environment can be challenging because of variation in surrounding habitats. Species can be limited in a suitable habitat due to dispersal limitation and that does not always reflect the suitability of a particular habitat for a species. In some cases, individual species will dominate the habitat while in others multiple species will co-exist within the same habitat. Thus, the process of classifying multiple species is challenging and still

an open problem. As a result, this study focuses on this issue of classifying multiple species from the provided datasets.

Much work in literature has focused mostly on statistical methods for classifying single species (Alldredge et al., 2007; Wright et al., 2020) and very few studies on classification of multiple species (Alldredge et al., 2007). Statistical methods that have been used to model single species includes poisson generalized linear models (Zhu, 2012; Kassahun et al., 2014; He et al., 2019), negative binomial models (Minami et al., 2007; Hoef and Boveng, 2007; Lindén and Mäntyniemi, 2011), zero-inflated poisson models (Ridout et al., 2001; Lambert, 2012; Wagh and Kamalja, 2018) and zero-inflated negative binomial models (Agarwal et al., 2002; Wenger and Freeman, 2008) as well as some machine learning models such as random forests (Cutler et al., 2007) and support vector machines (Crisci et al., 2012). Although many methods have been developed for models of individual species, ecologists are recognizing the importance of predicting multiple species simultaneously using multi-label classification (Jones et al., 2011). However, to date, multi-label classification has been applied primarily to

Corresponding author.

E-mail address: Bonelwa.Sidumo@nwu.ac.za (B. Sidumo).

<https://doi.org/10.1016/j.ecoinf.2022.101822>

Received 8 July 2022; Received in revised form 16 September 2022; Accepted 17 September 2022

1574-9541/© 20XX