Multi-step forecasting of the Philippine storm frequencies using Poisson Neural Network

Al-Ahmadgaid B. Asaad (alahmadgaid@gmail.com)
Aboitiz Data Innovation https://orcid.org/0000-0003-3784-8593

Research Article

Keywords: Count Time Series, Maximum Likelihood Estimation, Python, TensorFlow Probability, Typhoons

Posted Date: March 14th, 2022

DOI: https://doi.org/10.21203/rs.3.rs-1335357/v1

Multi-step forecasting of the Philippine storm frequencies using Poisson Neural Network

Al-Ahmadgaid B. Asaad[1,2*]

[1*]Data Science Solutions, Aboitiz Data Innovation, Goldbell Towers, 47 Scotts Road, Singapore City, 228233, Singapore.
[2]Data Science Solutions, UnionBank of the Philippines, UnionBank Plaza Bldg, Meralco Avenue, Pasig City, 1605, Philippines.

Corresponding author(s). E-mail(s): alahmadgaid@gmail.com;

Abstract

The paper aims to forecast the Philippine storm frequencies using nonlinear Poisson model. More specifically, the nonlinear kernel of the model is defined by the Artificial Neural Network (ANN) with one hidden layer and at least two output neurons. The model is trained to simultaneously forecast two semesters ahead for a given input. Further, the covariates studied were the Average Sea Surface Temperatures in the NINO3.4 region (5°-5°S, 170°-120°W), and the Average Sea Surface Temperatures in the eastern pole (0°-10°S, 90°-110°E) of the Dipole Mode Index. The data, taken from the Japan Meteorological Agency's Regional Specialized Meteorological Center, with time points running from 1950 to October 2021, is modeled at a semester-level granularity. The estimation is done using Maximum Likelihood Estimation by minimizing the negative log-likelihood function. Further, the proposed model is studied for different activation functions and different number of hidden neurons. Lastly, out of the 12 candidate models, the best model captures well the characteristics of the data both in terms of the point forecast and the associated uncertainties.

Keywords: Count Time Series, Maximum Likelihood Estimation, Python, TensorFlow Probability, Typhoons

1 Introduction

From 2001 to 2016, the tropical storms cost the Philippines an estimated loss of 12.5 million tons in rice production, with 260 000 tons of this loss is believed to have come from the 2013 typhoon Haiyan alone (Blanc and Strobl, 2016). The said typhoon is also responsible for the death of 6300 individuals (Lagmay et al, 2015), and for forcing coconut farmer survivors to explore other sources of income, including raising livestock and growing other crops (Seriño et al, 2021). Apart from the strong winds, Haiyan—known locally as Yolanda—also brought with it huge storm surges, drowning the locals in the coastal area of Leyte province (Yi et al, 2015). It is therefore of interest for the Philippines to further study the behavior of these natural events, to help policy makers take early precautions to hopefully reduce further losses.

Among the interests related to these disasters is the annual number of tropical cyclones (TCs) entering the Philippine Area of Responsibility (PAR). From the literature, the task of forecasting the TC counts for PAR was first studied by Magee et al (2021), whose paper also modeled other areas of reponsibilities of other southeast asian countries. The said paper started with 16 covariates, which was then filtered using an automated variable selection algorithm for each location involved. The model used by Magee et al (2021) was a Poisson regression model for all locations studied. Further, among the covariates frequently used by their automated variable selection algorithm, the Average Sea Surface Temperatures in the NINO3.4 region (5°-5°S, 170°-120°W), and the Average Sea Surface Temperatures in the eastern pole (0°-10°S, 90°-110°E) of the Dipole Mode Index (DMI) are both used as predictors in this paper. Outside the work of Magee et al (2021), the rest of the literature have focused on major ocean basins, for example, the Western North Pacific (WNP) and the Indian ocean regions.

Moreover, most TC count modeling are done at an annual level granularity. This is true for both Magee et al (2021) and those papers applied to major ocean regions, examples like Fan (2007) and Fan and Wang (2009) for WNP; and, McDonnell and Holbrook (2004b) for australian-southwest pacific region. For this study, however, we'll model the TC frequencies at a semester level granularity. This is because we want to take into account the autoregressive covariates to our proposed model, and a semester level granularity will make the historical data to have periodicity annually. So that, future frequencies can be described by this periodicity expressed as autoregressive term of the model input. The periodicity follows from the fact that the first semester is a non-TC season for PAR, contrary to the second semester. This is evident from the result of Asaad (2021). Therefore, the first semester of the year will always be lower compared to the second semester in terms of the TC frequencies, and this pattern recurs annually.

In addition, most TC count modeling are trained for one point ahead forecasting. That is, given an input, we forecast one year ahead for an annual-level dataset. For multi-step forecasting, we roll the forecast, that is, previous forecast will be used as input for future forecast, and the independent covariates involved needs to be forecast ahead as well. The problem with this approach is that, it can quickly accumulate errors since the error of the preceding forecast is propagated to future forecast. This is true with the model proposed by Magee et al (2021), and we address this by simultaneously forecasting two steps ahead for two semesters for every given input. Further, instead of using linear models, which dominated the literature of TC count modeling, we explore a nonlinear model. In particular, we propose an Artificial Neural Network (ANN) model with one hidden layer and two output neurons (for the two semesters ahead forecasting).

As for the literature on TC count forecasting for major ocean basins, most of the models used are linear models. Popular choices are multiple linear regression models like in the works of Fan (2007), Choi et al (2010), and Fan and Wang (2009); others have used Poisson generalized linear regression models, for example in the study of McDonnell and Holbrook (2004a), McDonnell and Holbrook (2004b), Chand et al (2010), Werner and Holbrook (2011), Zhang et al (2018), and Magee et al (2021). As for the nonlinear models, however, only Nath et al (2015) have applied it to TC count forecasting, to the best of knowledge of the author. The model used by Nath et al (2015) is based on ANN. It should be noted that, there are other nonlinear Poisson regression models apart from ANN based kernels. For example, the exponential Poisson autoregressive models studied by Fokianos et al (2009).

ANN models have gained success in the field of Machine and Deep Learning on many applications including time series forecasting. In fact, the popular extension of ANN for sequential modeling named Long Short Term Memory (LSTM) by Hochreiter and Schmidhuber (1997) have performed better on complex time series forecasting compared to statistical models (*see* for example Elsaraiti and Merabet (2021)). The striking feature of ANN-based models is their complexity, which dwarfed the simplicity of statistical models. This complexity, however, is also the reason why ANN-based models are treated as "block-box," mainly because of their lack of interpretability. This limitation, however, motivated some recent innovations, for example the Neural Basis Expansion Analysis Time Series (NBEATS) model by Oreshkin et al (2020), aims to provide an interpretable yet powerful univariate Deep Learning model; and another, is the Temporal Fusion Transformer (TFT) model by Lim et al (2021), also interpretable. These Deep Learning models are suitable for complex time series datasets, those defined to be high in granularity (e.g., every 3 seconds or every 5 minute datasets) with low autocorrelations, meaning past values don't have much influence on current values; and, those requiring long horizon forecasting, for example 120 points for one day forecasting of a 5-minute dataset. As such, these models can likely overfit a 142 data points considered in this paper. Hence for this study, we'll use the simplest ANN model. More details are given in Section 2.

Furthermore, the two output neurons of the proposed model are assumed to be governed by two independent Poisson distributions, each having conditional mean conditioned on the nonlinear combinations of the autoregressive inputs and the covariates. The model is therefore named as Poisson Neural Network

(PoisNN). Another alternative for Poisson would be the negative Binomial distribution as in the work of Zhu (2011), but this is beyond the scope of the paper. The assumed Poisson distribution will provide the observation uncertainties, which we refer in this paper as the *aleatoric uncertainties* adopted from Kendall and Gal (2017). Lastly, the model parameters are inferred using Maximum Likelihood Estimation (MLE), but instead of maximizing, we minimize the negative log-likelihood function.

PoisNN models were first studied by Fallah et al (2009), to the best knowledge of the author. The said paper found that, PoisNN models can largely improve the prediction in nonlinear situations. Another application of PoisNN is the work of Montesinos-López et al (2020) on prediction of genomic count data. Further, Huang et al (2019) also used PoisNN, but for forecasting number of emergency calls. Lastly, Rodrigo and Tsokos (2020) proposed a Bayesian PoisNN and found that it performs well over traditional Poisson or negative binomial regression models based on their simulation and real data studies.

The work of Rodrigo and Tsokos (2020) explored two different activation functions, *sigmoid* and *hyperbolic tangent*, which we also considered in this paper. These functions were also studied by Nath et al (2015), and found them to be the optimal activations for their ANN model with one hidden layer.

In summary, this paper has two main contributions to the literature of TC activities: *i.* the use of Poisson Neural Network model; and, *ii.* the use of multi-step forecasting.

2 Data and methods

The data is publicly available from the Japan Meteorological Agency's (JMA's) Regional Specialized Meteorological Center (RSMC). The available time points run from 1951 to October of 2021. In total, there are 1852 unique TC international IDs (IDs).

In order to count the frequency of the TCs for the PAR, the 1852 TC IDs are further filtered to only those that have entered the PAR. These include tracks that made a recursive path, meaning entering the PAR briefly and left for the north. In total, 1250 TC IDs tallied under the PAR. It should be noted that, the Philippines has its own local weather agency (PAGASA short for Philippine Atmospheric, Geophysical and Astronomical Services Administration) tracking the TCs, and there can be minor differences in terms of TC counts with that in JMA. Nonetheless, the difference should be marginal. Figure 1a shows the 1852 TCs in the WNP, and Figure 1b shows the 1250 TCs in PAR (tails and heads of the tracks beyond PAR were cut out for emphasis). These data points were then grouped per semester annually, so that Figure 2a depicts the time series of the historical frequencies of the TCs in PAR. The historical data for NINO3.4 average SST is also displayed in Figure 2b, and Figure 2c is the historical data of the IODE average SST. It should be noted that Asaad (2021) defines the Philippine storms as those that went straight to the Philippines, and classified those that made a recursive track as belonging to a different

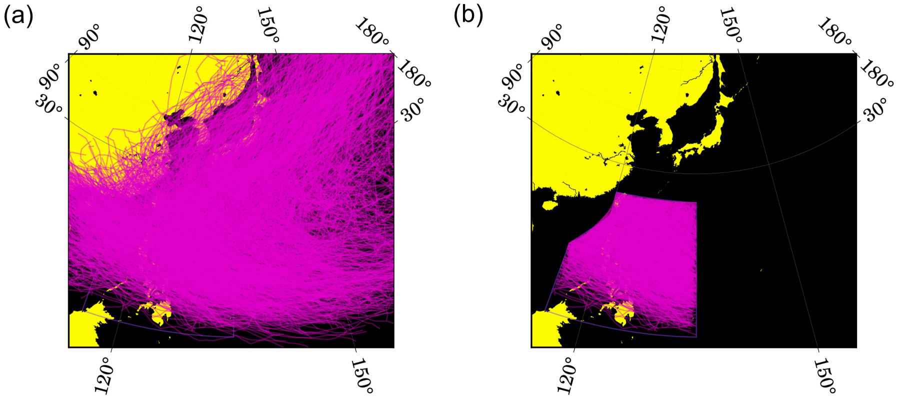

Fig. 1 Typhoon tracks in the **a** Western North Pacific (WNP) region; and, **b** the Philippine Area of Responsibilities (PAR)

cluster, including those that briefly entered PAR. While it differ, the proposed definition in this paper is already a good proxy for the number of storms that passed straight through the country.

In this study, the data is partitioned into training and testing, with training accounting for 80% of the data, that is up until the first semester of 2007, and the rest accounts for the test data, that is, up until the first semester of 2021. We exclude the second semester of 2021, since the data is not complete, and instead we will start our roll forecast with it for the 2022 semesters. The model parameters are estimated using the training data, with 20% of the training data used as the validation set. The final model is then evaluated in the test data. The software used for this study is Python (Van Rossum and Drake, 2009) programming language, using TensorFlow Probability (Dillon et al, 2017) as the core library for ANN modeling using MLE. Julia programming language was also used to produce the maps in Figure 1.

The remainder of the paper is organized as follows: Section 2.1 introduces the PoisNN model; Section 2.2 discusses the model estimation using MLE; Section 3 presents the results; and, Section 4 is the conclusion and future works.

2.1 Poisson artificial neural network

This section provides details on the proposed model defined in Definition 1.

Definition 1 (Poisson ANN) Let $\mathscr{F}_t^{X,Y,Z} := \sigma(X_i, Y_i, Z_i, i \leq t, t \in \mathbb{N})$ be the σ-algebra generated by $\{X_1, \cdots, X_t, Y_1, \cdots, Y_t, Z_0, \cdots, Z_t\}$, such that $[Z_{t+1}, Z_{t+2}]^\top \mid \mathscr{F}_t^{X,Y,Z} \sim \prod_{k=1}^{K} \text{Poisson}(\lambda_{t+k})$, then the *Poisson ANN model* is defined by

$$\lambda_{t+k} := \exp\left[\alpha_k + \sum_{j=1}^{q} \omega_{kj} g\left(h(\boldsymbol{\eta}, j)\right)\right], \tag{1}$$

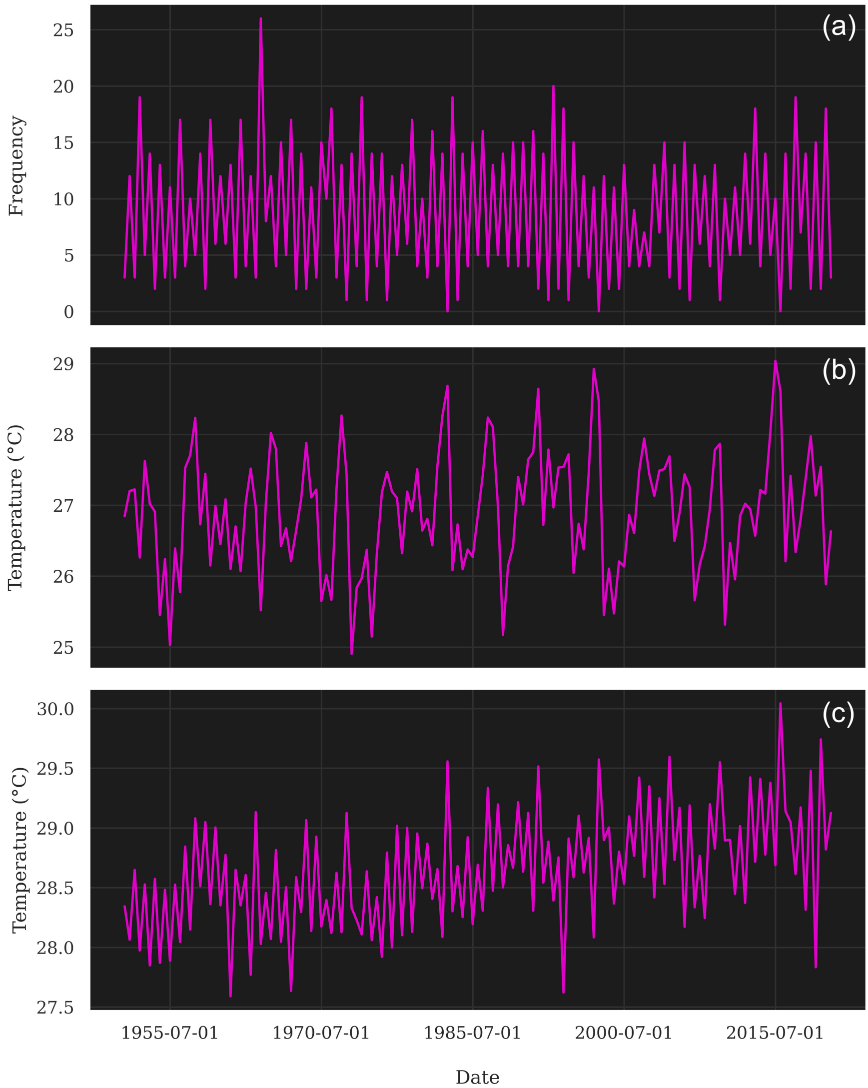

Fig. 2 Historical data for **a** PAR tropical cyclones' frequency; **b** the NINO3.4 average SST; and, **c** the IODE average SST

where

$$h(\boldsymbol{\eta}, j) := \beta_j + h_X(m, j) + h_Y(n, j) + h_Z(p, j), \tag{2}$$

$$h_X(m, j) := \sum_{i=t-m}^{t} \varphi_{ji} \log(X_i + 1), \tag{3}$$

$$h_Y(n,j) := \sum_{i=t-n}^{t} \vartheta_{ji} \log(Y_i + 1), \tag{4}$$

$$h_Z(p,j) := \sum_{i=t-p}^{t} \psi_{ji} \log(Z_i + 1), \tag{5}$$

$\boldsymbol{\eta} := [m,n,p]^\top$, and g is the activation for the hidden layer.

Remark 1 We assumed that the joint distribution of Z_{t+1} and Z_{t+2} given $\mathscr{F}_t^{X,Y,Z}$ is equal to the product of its marginal distributions, i.e., they are independently distributed. So that, given $\mathscr{F}_t^{X,Y,Z}$, $\Pr(Z_{t+1}, Z_{t+2}) = \Pr(Z_{t+1})\Pr(Z_{t+2}) = \prod_{k=1}^{2} \text{Poisson}(\lambda_{t+k})$.

Remark 2 The X_t and Y_t correspond to the covariates, which for this study are the Average Sea Surface Temperatures in the NINO3.4 region (5°-5°S, 170°-120°W), and the Average Sea Surface Temperatures in the eastern pole (0°-10°S, 90°-110°E) of the Dipole Mode Index (DMI), respectively.

Remark 3 The model in Eq. (1) encodes the input data into a logarithmic scale and decodes the output using the exponential activation function. These encoding and decoding functions are inspired by the log-linear Poisson autoregressive model proposed by Fokianos and Tjøstheim (2011).

For this study, we explore two sets of models: the first are those with covariates X_t and Y_t; and, the second are those with autoregressive terms only. Lastly, for the number of hidden neurons, we examine 4, 8 and 16 neurons, each are studied for 2 activation functions: Sigmoid, and Hyperbolic Tangent.

The network architecture for the proposed model is given in Figure 3. The dashed outline in each neuron indicates that the node is activated by its activation function. Therefore, the input neurons with this indicator are activated by the log function specified in Eq. (3)-(5). Also, the output neurons are fixed to exponential activation functions as specified in Eq. (1).

2.2 Model estimation

The weights of the model are estimated using MLE, and is done as follows: let

$$\mathscr{Y} := \{[y_1^{(1)}, \cdots, y_N^{(1)}]^\top, \cdots, [y_{T+1-N-K}^{(R)}, \cdots, y_{T-K}^{(R)}]^\top\}, \tag{6}$$

$$\mathscr{X} := \{[x_1^{(1)}, \cdots, x_M^{(1)}]^\top, \cdots, [x_{T+1-M-K}^{(R)}, \cdots, x_{T-K}^{(R)}]^\top\}, \tag{7}$$

$$\mathscr{W} := \{[z_1^{(1)}, \cdots, z_P^{(1)}]^\top, \cdots, [z_{T+1-P-K}^{(R)}, \cdots, z_{T-K}^{(R)}]^\top\}, \tag{8}$$

$$\mathscr{Z} := \{[z_{P+1}^{(1)}, \cdots, z_{P+K}^{(1)}]^\top, \cdots, [z_{T-K+1}^{(R)}, \cdots, z_T^{(R)}]^\top\}, \tag{9}$$

where T is the total number of observations. So that, the joint of each collection in matrix form is represented as follows: $\mathbf{X} := [\mathbf{x}_1, \cdots, \mathbf{x}_R], \mathbf{Y} :=$

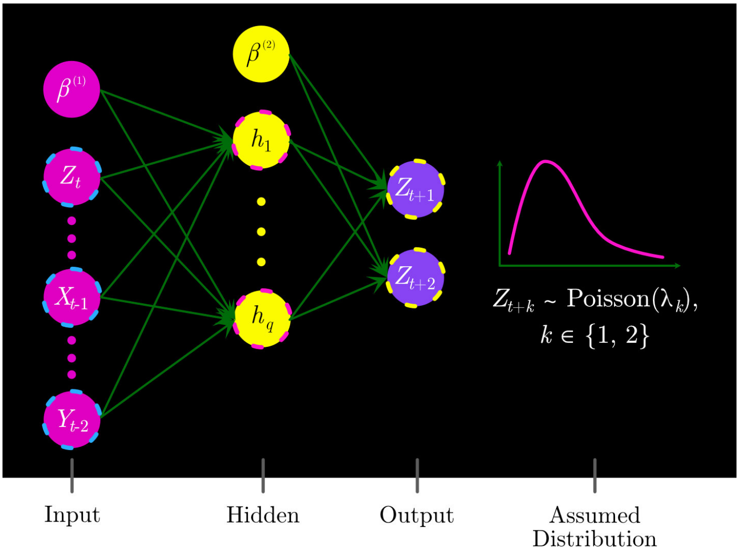

Fig. 3 Network architecture of the proposed Poisson ANN model for two semesters horizon

$[\mathbf{y}_1, \cdots, \mathbf{y}_R], \mathbf{W} := [\mathbf{w}_1, \cdots, \mathbf{w}_R]$, and $\mathbf{Z} := [\mathbf{z}_1, \cdots, \mathbf{z}_R]$. Further, let V be the random variable of observing the data $\mathbf{z}_i := [z^{(i)}_{t+1}, \cdots, z^{(i)}_{t+K}]^\top$, and let $\mathscr{P} := \{\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\omega}, \boldsymbol{\varphi}, \boldsymbol{\vartheta}, \boldsymbol{\psi}\}$ be a collection of weights and biases of the model, then the likelihood function is given by:

$$\mathcal{L}(\mathscr{P} \mid \mathbf{Z}, \mathbf{W}, \mathbf{X}, \mathbf{Y}) := \Pr(\mathbf{Z} \mid \mathbf{W}, \mathbf{X}, \mathbf{Y}, \mathscr{P}) \tag{10}$$

$$= \prod_{i=1}^{R} \Pr(V = \mathbf{z}_i \mid \mathbf{w}_i, \mathbf{x}_i, \mathbf{y}_i, \mathscr{P}) \tag{11}$$

$$= \prod_{i=1}^{R} \Pr(V = [z^{(i)}_{t+1}, \cdots, z^{(i)}_{t+K}]^\top \mid \mathbf{w}_i, \mathbf{x}_i, \mathbf{y}_i, \mathscr{P}) \tag{12}$$

$$= \prod_{i=1}^{R} \Big[\Pr(z^{(i)}_{t+1} \mid \mathbf{w}_i, \mathbf{x}_i, \mathbf{y}_i, \mathscr{P}) \times \cdots \tag{13}$$

$$\times \Pr(z^{(i)}_{t+K} \mid \mathbf{w}_i, \mathbf{x}_i, \mathbf{y}_i, \mathscr{P})\Big]$$

So that, the log-likelihood is given by:

$$\ell(\mathscr{P} \mid \mathbf{Z}, \mathbf{W}, \mathbf{X}, \mathbf{Y}) := \log \mathcal{L}(\mathscr{P} \mid \mathbf{Z}, \mathbf{W}, \mathbf{X}, \mathbf{Y}) \tag{14}$$

$$= \sum_{i=1}^{R} \Big[\log \Pr(z^{(i)}_{t+1} \mid \mathbf{w}_i, \mathbf{x}_i, \mathbf{y}_i, \mathscr{P}) + \cdots \tag{15}$$

$$+ \log \Pr(z^{(i)}_{t+K} \mid \mathbf{w}_i, \mathbf{x}_i, \mathbf{y}_i, \mathscr{P})\Big]$$

where

$$\Pr(z_{t+k}^{(i)} \mid \mathbf{w}_i, \mathbf{x}_i, \mathbf{y}_i, \mathscr{P}) = \frac{\lambda_{t+k}^{z_{t+k}^{(i)}} \exp(-\lambda_{t+k})}{z_{t+k}^{(i)}!}, \tag{16}$$

and λ_{t+k} is a function of $\mathbf{w}_i, \mathbf{x}_i, \mathbf{y}_i$ and $\mathscr{P}$ as defined in Eq. (1). Substituting Eq. (1) to Eq. (15) gives us the complete form of the log-likelihood, which then is the objective function. The goal therefore is to maximize this objection function, i.e.,

$$\hat{\mathscr{P}} := \underset{\mathscr{P}}{\operatorname{argmax}}\; \ell(\mathscr{P} \mid \mathbf{Z}, \mathbf{W}, \mathbf{X}, \mathbf{Y}). \tag{17}$$

However, maximizing Eq. (17) is equivalent to minimizing the negative of it, or explicitly, the negative log-likelihood, i.e.,

$$\hat{\mathscr{P}} := \underset{\mathscr{P}}{\operatorname{argmin}}\; \left[-\ell(\mathscr{P} \mid \mathbf{Z}, \mathbf{W}, \mathbf{X}, \mathbf{Y})\right]. \tag{18}$$

Minimizing this objective function is done by taking the partial derivative with respect to each of the parameters or weights of the model. The resulting derivatives are then set to 0 to align to the critical values, then solving for the corresponding weight or parameter should give the solution. The second derivative will then confirm if the solution is the minimum value. However, most objective functions of nonlinear models like ANN, don't have closed-form solution, and so exact solutions through differentiations are not possible. In practice, we approximate the solution by numerical approximations, for example using Stochastic Gradient Descent (SGD) (Robbins and Monro, 1951), RMSProp (Hinton et al, 2012) or ADAM (Kingma and Ba, 2015) algorithm. For this paper, we used RMSProp.

Lastly, minimization is often the preferred optimization task since most software implementing the numerical approximations were developed for minimizing objective functions, this is true with the algorithms implemented in TensorFlow (Abadi et al, 2015), which is used by TensorFlow Probability (Dillon et al, 2017).

2.3 Model inference

For every given input, the model returns two point estimates as forecasts for two semesters ahead. These two estimates serve as the rate parameter or the mean of the Poisson distributions, which we can use to compute the corresponding quantiles of the prediction intervals. The intervals are 90% (lower bound at 2.5th percentile and upper bound at 97.5th percentile), 75% (12.5th lower and 87.5th upper percentiles), and 50% (25th lower and 75th upper percentiles). The percentile/quantile q of the forecast for training or test data is denoted by $\hat{z}_{t+k,q}^{(i)}$, but for future TC count the notation is $\hat{z}_{t+k,q}$, and is computed as follows: $\hat{z}_{t+k,q} := F^{-1}(q \mid \lambda_{t+k})$, which is the image of

$F(\hat{z}_{t+k,q} \mid \lambda_{t+k})$, the cumulative distribution function, defined below:

$$F(\hat{z}_{t+k,q} \mid \lambda_{t+k}) := \Pr(Z_{t+k} \leq \hat{z}_{t+k,q} \mid \lambda_{t+k}) = \sum_{i=1}^{\hat{z}_{t+k,q}} \frac{\lambda_{t+k}^{i} \exp(-\lambda_{t+k})}{i!}. \quad (19)$$

3 Results and discussions

The results are presented into the following sections: Section 3.1 discusses the performance of the candidate models; and, Section 3.2 discusses the corresponding forecasts of the best model.

3.1 Candidate models' performance

There are 12 models in total that were studied in this paper, these are shown in Table A1 with their corresponding errors and correlation for the training data. The first six models contain independent variables, which are the average SSTs for both NINO3.4 and the IODE regions; and, the remaining 6 candidates from models 7 to 12 are PoisNN with no indepedent covariates, that is, we only used the autoregressive terms to forecast the future TC counts. Each of these models were trained for 1000 epochs, but we used early stopping rule in which the training will stop if there are no improvements after 20 consecutive epochs.

From Table A1, we can see that Model 10 is the best performing PoisNN for training data, which only uses autoregressive terms as its predictors, and with 8 hidden neurons all activated by a hyperbolic tangent function. This is followed by Model 11 which is much simpler as it only has 4 hidden neurons, and is activated by sigmoid function. On the other hand, for the test dataset shown in Table A2, we found that Model 11 is the best performing model in all metrics. This is followed closely by Model 5 which uses the two independent covariates. Both Model 5 and Model 11 uses only 4 hidden neurons and are both activated by a sigmoid function. Therefore, out of the 12 models, we chose Model 11 as it can generalize well in the out-sample dataset.

The main advantage of Model 11 is that it does not depend on other independent covariates, making it easy to do roll forecast on further values. This is not true with Model 5 since we have to forecast future values of the independent covariates as well, to use as model input for forecasting further horizons.

The history of the minimization of the loss function of the Model 11 across epochs is given in Figure 4.

3.2 Forecasts

The forecast of the best model (i.e., Model 11) for the training and testing datasets are given in Figures 5-6, respectively. From both figures, the model does capture well the characteristics of the time series. There are some misses, however, for example the significant spike on the second semester of 1964 in the training dataset. As for the testing dataset, it also misses on values at first semester of 2010, at second semester of 2013, and at first semester of

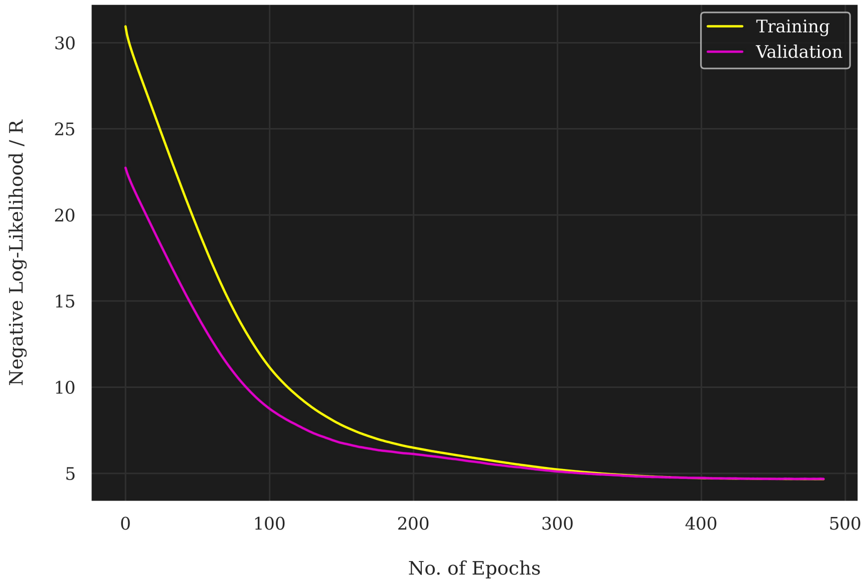

Fig. 4 Negative log-likelihood across training epochs

2016. Despite these errors, however, most of the actual values are within the prediction intervals as indicated by the *aleatoric uncertainty*.

Moreover, it is a common practice in the literature (for example Zhang et al (2018), McDonnell and Holbrook (2004b), among others) of TC count forecasting to also check on the Pearson correlation between the forecast and the actual values. For the proposed model, the correlation is high since the model takes advantage of the semester-level data, where periodic patterns recur every year.

In terms of the forecast of first and second semesters of 2022, Table 1 presents the details. As mentioned in Section 2, we did not use the second semester data for 2021 since it is incomplete, as it only accounts TC tracks up until October 2021. Having said that, we therefore need to forecast the second semester of 2021, and then forecast the semesters of 2022. The results are given in the first row of Table 1. Using the forecast mean of the second

Table 1 Tropical cyclone count forecasts of the proposed model up to the last semesters of 2022

Year-Sem	Aleatoric Percentiles						Mean
	2.5	12.5	25	75	87.5	97.5	
2021 - Sem2	9	12	13	19	21	24	16.04
2022 - Sem1	0	1	2	5	6	8	3.52
2022 - Sem2	8	11	13	18	20	24	15.58

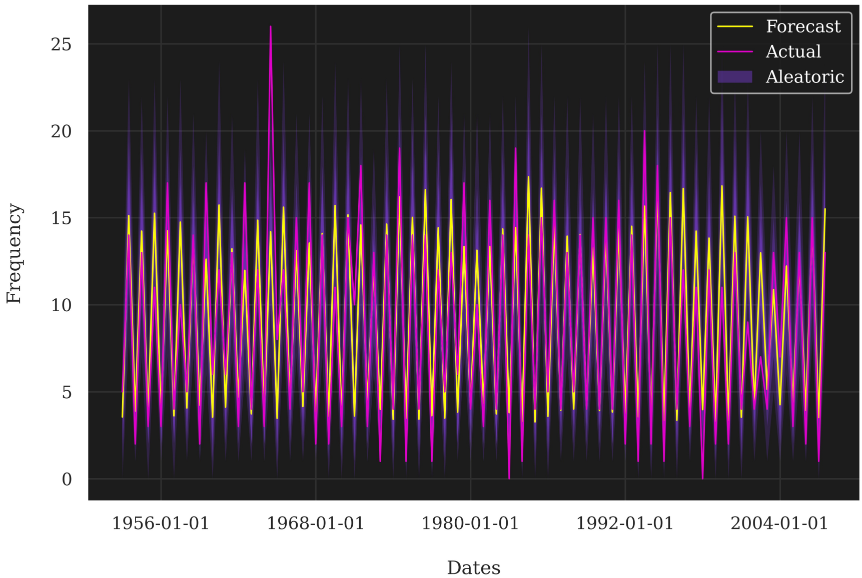

Fig. 5 One-semester ahead forecast on training data.

semester 2021, we can append this to the data to simultaneously forecast both semesters of 2022. The results are provided in the said table with the corresponding percentiles. From the inference, about 4 TCs are expected in PAR in the first semester of 2022, although it can be as low as 2 according to the 50% prediction interval. For the second semester, on the other hand, about 16 TCs are expected, but it can be as high as 20 according to the 87.5th percentile—the upper limit of the 75% prediction interval.

4 Conclusion and future works

The aim of the study was to explore the use of PoisNN model for modeling the Philippine storm frequencies. The results suggest that the two independent covariates, average SSTs in the NINO3.4 and IODE regions, does contribute in characterizing the TC counts if the PoisNN has 4 hidden neurons all activated by a sigmoid function, as specified by Model 5. Although, using this same architecture but excluding the independent covariates will further improve the model. Hence, we chose Model 11 as the best model among the 12 models explored. The absence of the covariates, meant that the model is easy to maintain and operationalize since it only depends on its historical data. We've shown this on forecasting the 2022, where roll forecasting was done to simultaneously forecast both semesters of the said year. So that, the model inferred about 4 TCs on the first semester of 2022, and about 16 TCs for the second semester.

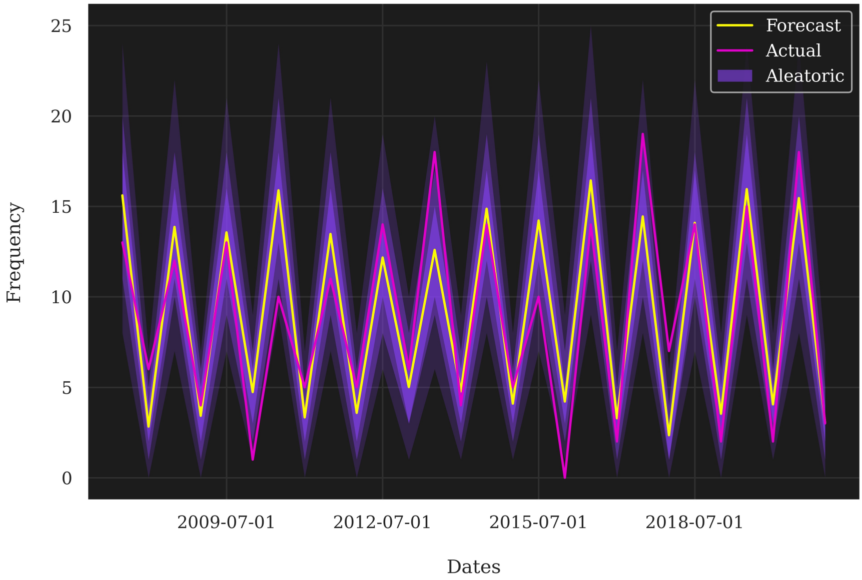

Fig. 6 One-semester ahead forecast on testing data.

There are still room for improvement. The Poisson distribution assumed in the output neurons assumes that the mean and the variance of the population distribution are equal. This is often not the case in practice, and one can explore the negative Binomial distribution instead.

Statements & Declarations

Funding

The authors declare that no funds, grants, or other support were received during the preparation of this manuscript.

Competing Interests

The authors have no relevant financial or non-financial interests to disclose.

Author Contributions Interests

All authors contributed to the study conception and design. All authors read and approved the final manuscript.

Data Availability

The RSMC Best Track Data can be downloaded here: https://www.jma.go.jp/jma/jma-eng/jma-center/rsmc-hp-pub-eg/besttrack.html

Appendix A Statistical tables

Table A1 Performance of the models on training dataset

Model	Input Layer		Hidden Layer		MSE	RMSE	MAE	Corr.
No.	Variable	Lags	g	Neurons				
1	W X Y	4 2 2	Sigmoid	16	8.87	2.98	2.28	0.87
2	W X Y	4 2 2	Tanh	16	7.86	2.8	2.16	**0.89**
3	W X Y	4 2 2	Sigmoid	8	8.52	2.92	2.24	0.88
4	W X Y	4 2 2	Tanh	8	8.09	2.84	2.18	0.88
5	W X Y	4 2 2	Sigmoid	4	7.7	2.78	2.13	**0.89**
6	W X Y	4 2 2	Tanh	4	10.8	3.29	2.51	0.86
7	W	4	Sigmoid	16	8.51	2.92	2.23	0.87
8	W	4	Tanh	16	7.60	2.76	2.12	**0.89**
9	W	4	Sigmoid	8	7.62	2.76	2.11	**0.89**
10	W	4	Tanh	8	**7.20**	**2.68**	**2.02**	**0.89**
11	W	4	Sigmoid	4	*7.31*	*2.70*	*2.06*	**0.89**
12	W	4	Tanh	4	7.62	2.76	2.13	**0.89**

Table A2 Performance of the models on testing dataset

Model No.	MSE	RMSE	MAE	Corr.
1	8.17	2.85	2.34	0.87
2	8.10	2.85	2.26	0.87
3	9.89	3.14	2.55	0.84
4	9.99	3.16	2.54	0.85
5	*7.55*	*2.75*	*2.28*	**0.88**
6	10.71	3.27	2.62	0.83
7	8.61	2.93	2.37	0.86
8	9.25	3.04	2.42	0.85
9	8.00	2.83	2.33	0.87
10	8.42	2.90	2.35	0.86
11	**7.49**	**2.74**	**2.26**	**0.88**
12	8.71	2.95	2.36	0.86

References

Abadi M, Agarwal A, Barham P, et al (2015) TensorFlow: Large-scale machine learning on heterogeneous systems. URL https://www.tensorflow.org/, software available from tensorflow.org

Asaad AAB (2021) Circular characteristics of the philippine storm tracks. Spatial Information Research https://doi.org/https://doi.org/10.1007/s41324-021-00415-4

Blanc E, Strobl E (2016) Assessing the impact of typhoons on rice production in the philippines. Journal of Applied Meteorology and Climatology 55(4):993 – 1007. https://doi.org/10.1175/JAMC-D-15-0214.1, URL https://journals.ametsoc.org/view/journals/apme/55/4/jamc-d-15-0214.1.xml

Chand SS, Walsh KJE, Chan JCL (2010) A bayesian regression approach to seasonal prediction of tropical cyclones affecting the fiji region. Journal of Climate 23(13):3425 – 3445. https://doi.org/https://doi.org/10.1175/2010JCLI3521.1

Choi KS, Moon JY, Kim DW, et al (2010) Seasonal prediction of tropical cyclone genesis frequency over the western north pacific using teleconnection patterns. Theoretical and Applied Climatology 100:191–206. https://doi.org/https://doi.org/10.1007/s00704-009-0182-1

Dillon JV, Langmore I, Tran D, et al (2017) Tensorflow distributions. CoRR abs/1711.10604. URL http://arxiv.org/abs/1711.10604, https://arxiv.org/abs/1711.10604

Elsaraiti M, Merabet A (2021) A comparative analysis of the arima and lstm predictive models and their effectiveness for predicting wind speed. Energies 14(20). https://doi.org/10.3390/en14206782, URL https://www.mdpi.com/1996-1073/14/20/6782

Fallah N, Gu H, Mohammad K, et al (2009) Nonlinear poisson regression using neural networks: a simulation study. Neural Computing and Applications 18(8):939. https://doi.org/https://doi.org/10.1007/s00521-009-0277-8

Fan K (2007) New predictors and a new prediction model for the typhoon frequency over western north pacific. Science in China Series D: Earth Sciences 50(8):1417 – 1423. https://doi.org/https://doi.org/10.1007/s11430-007-0105-x

Fan K, Wang H (2009) A new approach to forecasting typhoon frequency over the western north pacific. Weather and Forecasting 24(4):974 – 986. https://doi.org/https://doi.org/10.1175/2009WAF2222194.1

Fokianos K, Tjøstheim D (2011) Log-linear poisson autoregression. Journal of Multivariate Analysis 102(3):563–578. https://doi.org/https://doi.org/10.1016/j.jmva.2010.11.002

Fokianos K, Rahbek A, Tjøstheim D (2009) Poisson autoregression. Journal of the American Statistical Association 104(488):1430–1439. https://doi.org/https://doi.org/10.1198/jasa.2009.tm08270

Hinton G, Srivasta N, Swersky K (2012) RMSprop, URL http://www.cs.toronto.edu/~tijmen/csc321/slides/lecture_slides_lec6.pdf

Hochreiter S, Schmidhuber J (1997) Long Short-Term Memory. Neural Computation 9(8):1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735, URL https://doi.org/10.1162/neco.1997.9.8.1735, https://arxiv.org/abs/https://direct.mit.edu/neco/article-pdf/9/8/1735/813796/neco.1997.9.8.1735.pdf

Huang H, Jiang M, Ding Z, et al (2019) Forecasting emergency calls with a poisson neural network-based assemble model. IEEE Access 7:18,061–18,069. https://doi.org/https://doi.org/10.1109/ACCESS.2019.2896887

Kendall A, Gal Y (2017) What uncertainties do we need in bayesian deep learning for computer vision? In: Proceedings of the 31st International Conference on Neural Information Processing Systems. Curran Associates Inc., Red Hook, NY, USA, NIPS'17, p 5580–5590

Kingma DP, Ba J (2015) Adam: A method for stochastic optimization. In: Bengio Y, LeCun Y (eds) 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings, URL http://arxiv.org/abs/1412.6980

Lagmay AMF, Agaton RP, Bahala MAC, et al (2015) Devastating storm surges of typhoon haiyan. International Journal of Disaster Risk Reduction 11:1–12. https://doi.org/https://doi.org/10.1016/j.ijdrr.2014.10.006

Lim B, Arık SO, Loeff N, et al (2021) Temporal fusion transformers for interpretable multi-horizon time series forecasting. International Journal of Forecasting 37(4):1748–1764. https://doi.org/https://doi.org/10.1016/j.ijforecast.2021.03.012, URL https://www.sciencedirect.com/science/article/pii/S0169207021000637

Magee AD, Kiem AS, Chan JCL (2021) A new approach for location-specific seasonal outlooks of typhoon and super typhoon frequency across the western north pacific region. Scientific Reports 11:19,439. https://doi.org/https://doi.org/10.1038/s41598-021-98329-6

McDonnell KA, Holbrook NJ (2004a) A poisson regression model approach to predicting tropical cyclogenesis in the australian/southwest pacific ocean region using the soi and saturated equivalent potential temperature gradient as predictors. Geophysical Research Letters 31(20). https://doi.org/https://doi.org/10.1029/2004GL020843

McDonnell KA, Holbrook NJ (2004b) A poisson regression model of tropical cyclogenesis for the australian–southwest pacific ocean region. Weather and Forecasting 19(2):440 – 455. https://doi.org/https://doi.org/10.1175/1520-0434(2004)019⟨0440:APRMOT⟩2.0.CO;2

Montesinos-López OA, Montesinos-López JC, Singh P, et al (2020) A Multivariate Poisson Deep Learning Model for Genomic Prediction of Count Data. G3 GenesGenomesGenetics 10(11):4177–4190. https://doi.org/https://doi.org/10.1534/g3.120.401631, https://arxiv.org/abs/https://academic.oup.com/g3journal/article-pdf/10/11/4177/40633947/g3journal4177.pdf

Nath S, Kotal S, kundu P (2015) Seasonal prediction of tropical cyclone activity over the north indian ocean using the neural network model. Atmósfera 28(4):271–281. https://doi.org/https://doi.org/10.20937/ATM.2015.28.04.06

Oreshkin BN, Carpov D, Chapados N, et al (2020) N-beats: Neural basis expansion analysis for interpretable time series forecasting. In: International Conference on Learning Representations, URL https://openreview.net/forum?id=r1ecqn4YwB

Robbins H, Monro S (1951) A Stochastic Approximation Method. The Annals of Mathematical Statistics 22(3):400 – 407. https://doi.org/10.1214/aoms/1177729586, URL https://doi.org/10.1214/aoms/1177729586

Rodrigo H, Tsokos C (2020) Bayesian modelling of nonlinear poisson regression with artificial neural networks. Journal of Applied Statistics 47(5):757–774. https://doi.org/10.1080/02664763.2019.1653268

Seriño MNV, Cavero JA, Cuizon J, et al (2021) Impact of the 2013 super typhoon haiyan on the livelihood of small-scale coconut farmers in leyte island, philippines. International Journal of Disaster Risk Reduction 52:101,939. https://doi.org/https://doi.org/10.1016/j.ijdrr.2020.101939

Van Rossum G, Drake FL (2009) Python 3 Reference Manual. CreateSpace, Scotts Valley, CA

Werner A, Holbrook NJ (2011) A bayesian forecast model of australian region tropical cyclone formation. Journal of Climate 24(23):6114 – 6131. https://doi.org/https://doi.org/10.1175/2011JCLI4231.1

Yi CJ, Suppasri A, Kure S, et al (2015) Storm surge mapping of typhoon haiyan and its impact in tanauan, leyte, philippines. International Journal of Disaster Risk Reduction 13:207–214. https://doi.org/https://doi.org/10.1016/j.ijdrr.2015.05.007, URL https://www.sciencedirect.com/science/article/pii/S2212420915300078

Zhang X, Zhong S, Wu Z, et al (2018) Seasonal prediction of the typhoon genesis frequency over the western north pacific with a poisson regression model. Climate Dynamics 51:4585–4600. https://doi.org/https://doi.org/10.1007/s00382-017-3654-5

Zhu F (2011) A negative binomial integer-valued garch model. Journal of Time Series Analysis 32(1):54–67. https://doi.org/https://doi.org/10.1111/j.1467-9892.2010.00684.x