

RESEARCH ARTICLE

Aedes-AI: Neural network models of mosquito abundance

Adrienne C. Kinney^{1*}, Sean Current², Joceline Lega^{3,4,5}

1 Interdisciplinary Program in Applied Mathematics, University of Arizona, Tucson, Arizona, United States of America, **2** Department of Computer Science and Engineering, The Ohio State University, Columbus, Ohio, United States of America, **3** Department of Mathematics, University of Arizona, Tucson, Arizona, United States of America, **4** Department of Epidemiology and Biostatistics, University of Arizona, Tucson, Arizona, United States of America, **5** BIO5 Institute, University of Arizona, Tucson, Arizona, United States of America

* akinney1@math.arizona.edu

Abstract

We present artificial neural networks as a feasible replacement for a mechanistic model of mosquito abundance. We develop a feed-forward neural network, a long short-term memory recurrent neural network, and a gated recurrent unit network. We evaluate the networks in their ability to replicate the spatiotemporal features of mosquito populations predicted by the mechanistic model, and discuss how augmenting the training data with time series that emphasize specific dynamical behaviors affects model performance. We conclude with an outlook on how such equation-free models may facilitate vector control or the estimation of disease risk at arbitrary spatial scales.

OPEN ACCESS

Citation: Kinney AC, Current S, Lega J (2021) *Aedes-AI*: Neural network models of mosquito abundance. PLoS Comput Biol 17(11): e1009467. <https://doi.org/10.1371/journal.pcbi.1009467>

Editor: Dan Stowell, Tilburg University Faculty Humanities: Tilburg University Tilburg School of Humanities and Digital Sciences, NETHERLANDS

Received: April 22, 2021

Accepted: September 21, 2021

Published: November 19, 2021

Copyright: © 2021 Kinney et al. This is an open access article distributed under the terms of the [Creative Commons Attribution License](https://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Data Availability Statement: All data files, trained models, and codes are available at <https://github.com/T-MInDS/Aedes-AI>.

Funding: This work was supported by the following grants and organizations: NIH GM132008 (ACK; <https://www.nih.gov/>), AFOSR/AFRL Collaborative Center for Aeronautical Sciences GR115527 (SC; <https://hfcmpl.osu.edu/ccas>), and University of Arizona BIO5 Institute Team Scholars Program (JL; <https://www.bio5.org/>). The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Author summary

Aedes aegypti mosquitoes affect millions of people each year through infectious diseases such as chikungunya, dengue, and Zika. Because local vector levels need to be sufficiently high for associated outbreaks to occur, the ability to estimate mosquito abundance is a central component of assessing disease risk. The mosquito landscape model (MoLS) is a mechanistic model that estimates *Aedes aegypti* abundance from local weather time series, and is able to reproduce trends observed in surveillance data. However, scaling this up to a large number of locations is resource intensive, requiring a high-performance computing system. In this article, we develop artificial neural network models that are significantly faster than MoLS and can produce abundance estimates directly from local weather data. This approach reduces the computational time associated with estimating local mosquito levels, thereby allowing for a corresponding increase in the spatiotemporal resolution of these predictions. We compare network design choices, including architecture and training data, in their ability to accurately reproduce MoLS estimates and analyze model performance in locations across the contiguous United States.

Competing interests: The authors have declared that no competing interests exist.

Introduction

Artificial Neural Networks (ANNs) are ideally suited for modeling nonlinear, complex phenomena, and have either achieved or surpassed human-level performance on tasks involving image classification, anomaly detection, and event extraction [1–4]. This manuscript assesses the feasibility of developing ANNs that provide estimates of mosquito abundance directly from weather data. To this end, we train, validate, and test different types of neural networks on simulated datasets. The training data consist of years of representative daily weather time series for various locations in the US obtained from the Multivariate Adaptive Constructed Analogs (MACA) datasets [5–8], and of corresponding daily mosquito abundance predictions. The latter are estimated from a mechanistic model, the Mosquito Landscape Simulation (MoLS) [9], which uses the MACA data as input. Our goal is to train ANNs that learn how MoLS estimates mosquito abundance from weather time series. We introduce various metrics that allow us to compare the ANN predictions to those of MoLS, including criteria relevant to public health (e.g. mosquito season length and timing of peak mosquito abundance) and rank the proposed ANNs based on a composite score derived from these metrics. We conclude that artificial neural networks are able to quickly and accurately reproduce MoLS daily mean mosquito abundance estimates, provided they are given current and past weather data over a specific window of time; in the present case, we use windows of length 90 days, consisting of the day each abundance estimate is made and the previous 89 days.

Vector-borne diseases infect hundreds of millions of people annually, disproportionately impacting impoverished communities in tropical areas [10]. The past two decades have seen a surge of outbreaks associated with the vector *Aedes aegypti*, including a 2004 dengue outbreak in Singapore [11], a 2013–2014 chikungunya outbreak in the Americas [12], and the 2014–2015 Zika outbreak in Latin America [13]. These outbreaks exemplify the public health risk of arboviral diseases associated with severe clinical symptoms, but also demonstrate the importance of local vector control efforts to mitigate impact on affected communities [14–16]. Because vector-borne disease outbreaks require sufficiently high vector populations [15, 17, 18], the ability to predict vector abundance is a central component of assessing disease risk. Forecasting outbreaks is, however, further complicated by global interconnectedness [19] and climate change [20, 21]—factors shown to introduce vectors into previously uninhabited areas and increase the viable range of vectors, respectively.

MoLS [9] is a mechanistic stochastic model that estimates *Aedes aegypti* abundance from weather time series (temperature, precipitation, and relative humidity). It is parametrized from information previously published in the literature and its output is proportional to expected daily mosquito numbers. It was shown in [9] that when given accurate local weather data, MoLS is able to reproduce *Aedes aegypti* abundance trends observed in surveillance traps in Puerto Rico. Because its parameters are fixed and thus not location-dependent, MoLS is in principle able to estimate weather-related abundance anywhere, provided local weather information is available, and assuming the biological properties of local *Aedes aegypti* mosquitoes are sufficiently well captured by the model parameters. Scaling this up to a large number of locations is however resource intensive. Specifically, using MoLS to generate daily abundance predictions associated with 10 years of weather data (from 01/01/2011 to 12/31/2020) takes about 10 minutes per location (including file reading and writing time) on one core of a high power computer (HPC) AMD Zen2 node. For county-scale resolution in the US, computing 10 years worth of daily abundance estimates therefore requires about $3000/6 = 500$ HPC hours. The advantage of a HPC is of course that predictions for many locations can be run in parallel, reducing the user's wait time by a factor equal to the total number of cores available.

A faster, ANN-based alternative to MoLS is appealing since it would reduce computational time even further, thereby allowing for a corresponding increase in the spatial resolution of abundance estimates. Like MoLS predictions, the ANN output would reflect an arbitrary carrying capacity, and would need to be scaled to average abundance inferred from surveillance information, in order to account for local conditions (e.g. due to the presence of a variable number of water containers where mosquitoes can lay eggs). Of course, because the output of MoLS or any of its ANN replacements is dependent on input weather data, the accuracy of such estimates is limited by the reliability of the available weather information. Moreover, human influences on mosquito populations (see e.g. [22–24] and references therein), other than consequences of climate change on weather data, are not taken into account in these models. Nevertheless, given that weather plays an important role in *Aedes aegypti* numbers [25–27], quickly producing weather-based abundance estimates of this disease vector is important.

The ability to replace a complex mechanistic model by an ANN that is faster and can easily be scaled up opens a new range of applications for these models. However, because artificial neural network predictions often lose accuracy in “unfamiliar” situations, the training data may need to place special emphasis on specific dynamic behaviors that are deemed important by the modeler. Estimating what type and what fraction of additional information is necessary to improve performance plays an important role in the development of ANN-based models. It is these questions that have motivated the work presented in this manuscript. Although they are addressed in the specific context of *Aedes aegypti* abundance, the approaches discussed here are general, and can be extended to other mechanistic models of vector abundance, such as for instance *DyMSim* [28], a model for the abundance of *Culex* species.

Methods

Training and input data

The neural network models discussed in this article have the same input data as MoLS. These consist of daily time series of maximum temperature, minimum temperature, precipitation, and average relative humidity. In MoLS, this information is used to calculate *Aedes aegypti* development, death, and reproductive rates, simulate daily mosquito abundance, and estimate the daily expectation of the number of gravid females [9]. The ANNs work differently: at any given location, a trained ANN converts weather time series for a fixed number of consecutive days into a single number, which is the estimated gravid female abundance at the given location on the last day of the given time series. Such an ANN is trained on a dataset consisting of weather time series and associated MoLS estimates.

Weather input data. We obtain daily weather time series from the Multivariate Adaptive Constructed Analogs (MACA) datasets website [5–8]. To train, validate, and test the models, we define a principal dataset consisting of daily data for the years 2012–2020 at 144 locations in 9 states: Arizona, California, Connecticut, Florida, New Jersey, New York, North Carolina, Texas, and Wisconsin. States other than Arizona are chosen because of their participation in the 2019 CDC *Aedes* Challenge [29] and the locations in this study are the centroids of counties that provided data for the challenge. For Arizona, we use MoLS predictions for the 50 most populated cities in the state. Together, these locations exemplify varying mosquito population patterns associated with different climates: hot and dry summers, hot and humid summers, cold winters, etc. We define a second dataset, called *Capital Cities*, to assess the performance of the trained ANNs across the contiguous US, in previously unseen locations (see S8 Appendix). To this end, we downloaded the 2012–2020 MACA time series for all

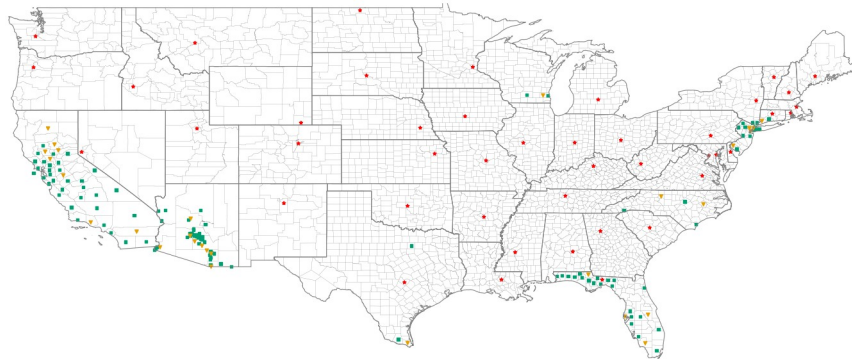


Fig 1. Map of the contiguous United States showing the locations used for training and validation (green squares), and testing (orange triangles). The locations used in the *Capital Cities* dataset are indicated by red stars. See Tables A and B in [S7 Appendix](#) for the names of the locations in the principal dataset. Base map obtained from the United States Census Bureau (<https://www.census.gov/geographies/mapping-files/time-series/geo/carto-boundary-file.html>).

<https://doi.org/10.1371/journal.pcbi.1009467.g001>

capital cities that are not situated in counties included in the principal dataset. [Fig 1](#) shows the locations whose time series we use for training, validation, and testing of the ANN models.

The weather time series used as input are noisy at the daily scale (with a correlation length of about 2 to 4 weeks for temperature, depending on location, and of about 6 days for relative humidity) but exhibit seasonal patterns. [Fig A](#) in [S1 Appendix](#) illustrates the temporal dynamics of daily average temperature, precipitation, and relative humidity in Phoenix, AZ, as well as the dependence of these quantities on two climate models (US GFDL-ESM2M and Canada CanESM2). [Fig D](#) in [S1 Appendix](#) (left two panels) shows the correlation of the weather between locations used for training and those used for testing. Choosing a range of geographical locations ensures the ANN models are tested on samples of varying weather trends.

Mosquito abundance input data. MoLS [9] is initiated with a specified number of *Aedes aegypti* eggs, and the simulation follows the life cycle of each egg “laid” in the system. An egg must survive through five immature stages before emerging as a fertile adult. At each stage in the life cycle, MoLS uses environmental and entomological features to simulate the lifespan of the mosquitoes, including temperature-dependent development rates and gonotrophic cycles, daily survival rates that depend on temperature and relative humidity, precipitation-dependent egg hatching, and carrying capacities estimated from water levels in simulated containers. MoLS takes about 10–12 weeks to ramp up, after which the simulated pool of mosquitoes (eggs, larvae, pupae, and adults) becomes representative of the weather data and local carrying capacity. Although MoLS output includes information on all of the life stages of a mosquito population, its default output is daily scaled gravid female abundance. This allows for direct comparison with surveillance data, which are often collected in gravid mosquito traps. More information about MoLS, including a comparison of its gravid female mosquito predictions against trap data for four neighborhoods in Puerto Rico may be found in [9].

In contrast to weather data, MoLS time series show little noise because they represent abundance expectation, smoothed over a two-week window. To illustrate how MoLS responds to changes in its input time series, its predictions for Phoenix, AZ, associated with weather data from two climate models are shown in [Fig B](#) in [S1 Appendix](#). The correlation of its estimates between testing and training locations is presented in the right panel of [Fig D](#) in [S1 Appendix](#).

Weather input sample length. MoLS keeps track of the number of eggs laid by adult female mosquitoes over many generations and, as a consequence, ANNs cannot be expected to reproduce MoLS results with only one day of weather data. Instead, they are provided with

weather information over a time window $[t_0 - \Delta + 1, t_0]$ of fixed length Δ days, in order to estimate abundance on day t_0 . Because MoLS takes about 10–12 weeks to ramp up, we expect Δ to be of comparable length, i.e. 90 days. Although large enough windows are needed for good performance, there is a trade-off between larger values of Δ and accuracy. Longer windows require users to provide reliable weather data over longer periods of time and increase computational cost. Moreover, windows that are too long may teach the ANNs to rely too much on what happened during the previous mosquito season. The models discussed in this article use $\Delta = 90$ days and are able to reproduce MoLS output with high skill. For comparison, we provide an example of an ANN trained with $\Delta = 120$ days in [S6 Appendix](#). We note that the average lifetime of a mosquito is estimated to be 30 days (about two weeks in immature stages [9] and two weeks in the adult stage [30]). Getting good results with $\Delta = 90$ days suggests that 3 times the average individual lifespan is sufficient to capture any correlation between current and future population trends.

Models

Baseline model. We utilize a simple linear regression model (LR) optimized with gradient descent as a baseline model for comparison. The linear regression model is trained on the same training subset as the ANNs and its weights are found using an Adam optimizer [31] with learning rate $\alpha = 0.0001$. Note that the LR model can output negative values; this is fixed in post processing by taking the maximum of the output and zero.

Neural network models. We define three neural network models: a feed-forward neural network, a long short-term memory neural network, and a gated recurrent unit neural network. A schematic is provided in [Fig 2](#). The layers in the thick-edged box are model dependent: fully-connected (FC) for Model 1, long short-term memory recurrent units (LSTM) for Model 2, and gated recurrent units (GRU) for Model 3, all of which we describe below. Each model begins with two convolution layers with 64 units each, a kernel size of 3 with no padding, and a stride of 1. Both layers use rectified linear unit (ReLU) activation and are immediately followed by a batch normalization layer as a form of regularization. Batch normalization scales layer outputs according to a learned mean and standard deviation to reduce overfitting and improve generalizability. We considered dropout methods as an alternate means to reduce overfitting, but removed them after they demonstrated no noticeable improvement on the validation set. The number of trainable parameters for each model is shown in [Table 1](#). Additionally, the reader is referred to [S2 Appendix](#) for details on the layers we use in the models and to [\[32\]](#) for a more thorough discussion of neural networks. The loss function and optimization are discussed in [Loss function and hyperparameter selection](#).

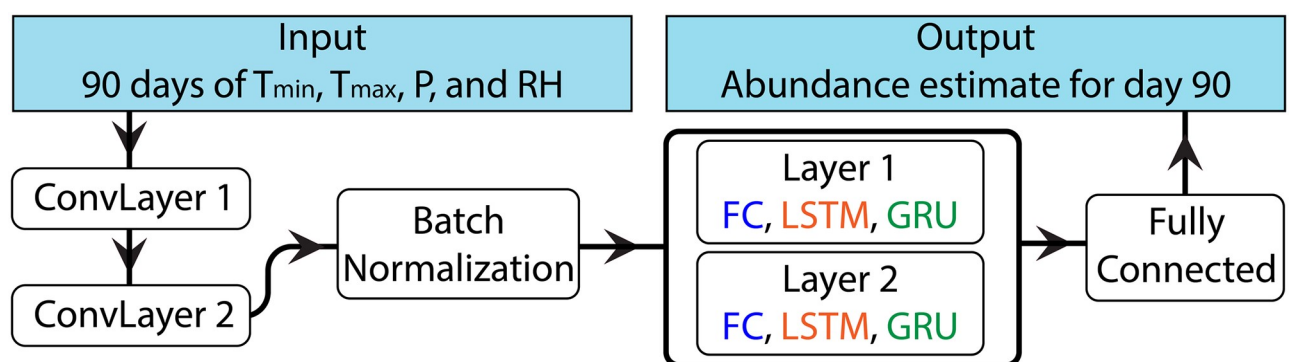


Fig 2. Architecture of the models used in this study.

<https://doi.org/10.1371/journal.pcbi.1009467.g002>

Table 1. Number of trainable parameters for each of the neural network models.

Model	Trainable Parameters
FF	369,857
LSTM	79,425
GRU	63,297

<https://doi.org/10.1371/journal.pcbi.1009467.t001>

Model 1: Feed-forward convolutional neural network (FF). The feed-forward network flattens the batch normalization output before applying two fully connected layers and an output layer. The fully connected layers have 64 units and ReLU activation. The output layer is a single unit with ReLU activation and ℓ_2 regularization to reduce over-fitting on the training data. The ℓ_2 regularization on the output augments the loss function with the ℓ_2 norm of the output weights, $\|w\|_2$, which must be minimized alongside the mean square error (MSE) loss. This penalizes large weight terms, requiring the model to utilize multiple features in its decision making, avoiding over-fitting as a result.

Model 2: Long short-term memory recurrent neural network (LSTM). Our second model architecture is an LSTM, chosen to exploit the “memory” feature of recurrent neural networks. LSTM units include gates that selectively allow information to propagate forward, thereby making it possible for previous information to directly influence the model’s behavior. Such a feature is relevant for abundance predictions since previous weather patterns impact current populations. For example, significant heat or cold decreases the viability of offspring, limiting future abundance. Moreover, *Aedes aegypti* eggs are known to be resistant to desiccation: long droughts do not necessarily cause a decrease in viable eggs, which can later hatch when rainfall creates new habitat (see for instance [33] and references therein). All of these features are taken into account in MoLS and are thus expected to be captured by the ANNs. The architecture of the LSTM model replaces the two fully connected layers of Model 1 with LSTM layers (Fig 2), each with 64 units and \tanh activation.

Model 3: Gated recurrent unit recurrent neural network (GRU). The final model architecture we consider is a GRU, chosen to leverage the benefits of the LSTM model while reducing the number of associated parameters. GRUs, like LSTMs, feature a gated unit to selectively allow information to propagate forward. However, the GRU unit is simpler than a LSTM unit (see Table 1), which reduces training time and the computational cost of using the model to generate predictions. The GRU architecture is identical to the LSTM architecture, except the LSTM layers are replaced by two GRU layers, each with 64 units and \tanh activation (Fig 2).

Model training

Data processing. We define subsets of the principal dataset for training, validation, and testing. The training subset contains daily weather data and corresponding MoLS predictions from 2012–2018 for 115 locations, shown in green (squares) in Fig 1, and we use it to set the weights in the ANNs. The validation subset contains data from 2019–2020 for the same 115 locations, and is used during hyperparameter selection (Loss function and hyperparameter selection) to optimize model performance. The testing subset contains the daily weather data from 2012–2020 for the 29 locations not included in the training and validation subsets, shown in orange (triangles) in Fig 1. The *Capital Cities* dataset contains daily weather data from 2012–2020 for capital cities in the contiguous US that are not in a county used in the principal dataset. The testing data do not include the corresponding MoLS time series, which are subsequently used to evaluate the performance of the optimized models in terms of the metrics defined in Performance metrics.

During the training and validation process of each ANN model, we process the input weather data in samples, where the i th input sample, $\mathbf{x}_i \in \mathbb{R}^{90 \times 4}$, represents 90 consecutive days of daily observations for the four weather variables (precipitation, maximum temperature, minimum temperature, and relative humidity) at a given location. For each training and validation input sample \mathbf{x}_i , we define the corresponding output target, $y_i \in \mathbb{R}$, as the gravid female abundance prediction by MoLS for the 90th day of the input sample at the same location. One thousand input samples, and their corresponding output targets, are randomly selected from each location in the training and validation subsets, and randomly shuffled to ensure the model is not dependent on spatiotemporal relationships among successive samples. We scale each sample between 0 and 1 using the global minimum and maximum values of each weather variable for the entire training subset before passing them to one of the ANN models. The resulting scaling factors are considered model parameters and are required for processing future weather samples. All future data, such as validation and testing data, are scaled using the same global minimum and maximum values as the training data. This ensures the data maintain the same relative scale across locations, while removing the differences in scale between temperature, precipitation, and humidity variables. The training samples are used to optimize the loss function and update the model layer parameters, while the validation samples guide hyperparameter selection, described in Loss function and hyperparameter selection.

After the training and validation process, the learned model weights, as well as the training data extrema, are saved and can be used to make predictions on unseen data. For the testing subset and the *Capital Cities* dataset, we again create input samples $\mathbf{x}_i \in \mathbb{R}^{90 \times 4}$ and use the ANNs to generate abundance estimates on the last day of each testing sample. For each combination of training, validation, and testing location and year, we create an associated abundance curve by constructing a time series of consecutive daily abundance estimates.

Loss function and hyperparameter selection. For each ANN model, the model weight parameters are selected during training by minimizing a loss function, defined as the mean squared error (MSE) between model output and MoLS predictions: $\sum_{i=1}^n (\hat{y}_i - y_i)^2 / n$ where n is the number of data points, \hat{y}_i is the i th prediction by the ANN model and y_i is the i th prediction by MoLS.

Model hyperparameters include the learning rate α , first moment decay rate β_1 , and the second moment decay rate β_2 of the Adam optimizer [31], as well as the number of units and type of activation function for each layer in the model. Given the extensive search space of hyperparameters for neural network models, it is not feasible to test all possible combinations of values. We construct each model by initializing layers with few units and iteratively increasing the number of units in each layer until either the desired performance is achieved or diminishing returns on validation performance are observed (“diminishing returns” are defined holistically; in particular, if the increase in the number of weights offers no significant decrease in the validation error after training, the lesser number of weights is used). Activation functions are similarly tested on a layer-by-layer basis. Finally, we test learning rates $\alpha \in \{0.01, 0.001, 0.0001, 0.00001\}$. We choose $\alpha = 0.0001$, while the first moment decay rate $\beta_1 = 0.9$ and second moment decay rate $\beta_2 = 0.999$ of the Adam optimizer are kept at their default values after changes in their values demonstrated less efficient optimization patterns.

We use a batch size of 64 and train for 100 epochs (enough for models to reach early-stopping convergence criteria; see Fig 3) with an early stopping patience level of 15 epochs. Early stopping prevents over-fitting the training subset by stopping the training process once no improvement is seen in model performance on the validation subset for 15 epochs. Once the

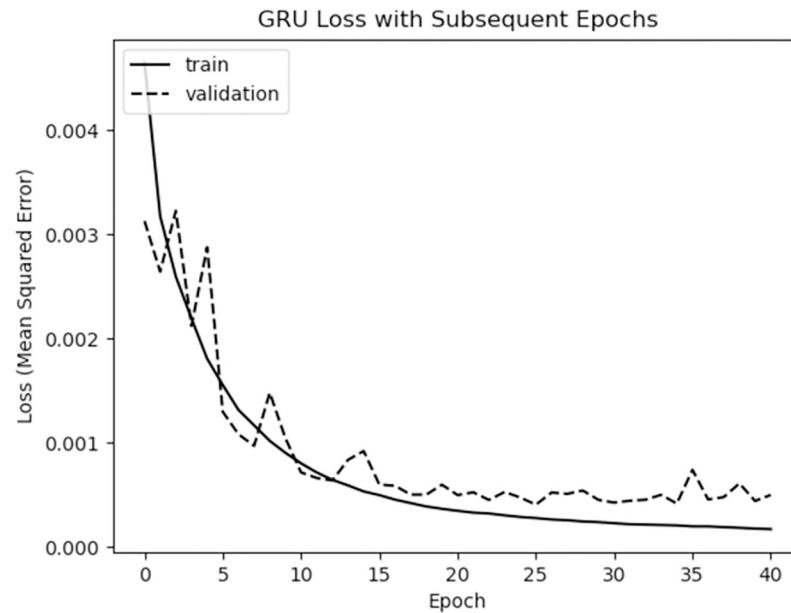


Fig 3. Training loss curve for the GRU model. Training halts due to early stopping after 40 epochs, indicating that the validation loss has reached its minimum value at 25 epochs.

<https://doi.org/10.1371/journal.pcbi.1009467.g003>

early stopping is triggered, the model parameters for the best performing epoch are selected as the learned weights for the model.

Data augmentation

The “base” training subset, described in Data processing, includes an equal number of input samples from all training locations, but we also define additional training subsets biased towards 1) the double peak pattern observed in hot and dry climates and 2) the absence of mosquito populations during colder off-season periods. The reader is referred to [S9 Appendix](#) for examples of these patterns. We test two data augmentation methods: high temperature (HI) oversampling supplements the base training subset with additional samples from on-season periods for double peak locations, while low temperature (LO) oversampling amends the training subset with additional samples from off-season periods from a diverse selection of locations.

High temperature oversampling (HI). The goal of the high temperature oversampling method is to increase the representation of samples featuring the loss of mosquito population due to extreme heat, resulting in the double peak season pattern. We manually identify locations in the original training subset featuring such a pattern (see Table A in [S7 Appendix](#)), and sample 1000 windows of length Δ at each selected location, which are then incorporated into the training data. Each of these windows is such that its final day lies within the on-season boundaries. As a consequence, the HI training subset includes 1000 randomly chosen input samples from on-season times for each of the locations in Table A in [S7 Appendix](#), in addition to 1000 randomly chosen input samples from each training location.

Low temperature oversampling (LO). The second approach is low temperature oversampling. Similar in construction to the high temperature oversampling method, we manually select a diverse set of locations featuring consistent losses of mosquito population due to cold weather, particularly during winter months. We randomly sample 1000 training windows for

each of the selected locations, with the final day of the training window lying in the colder, off-season months. Thus, the LO training subset includes 1000 randomly chosen input samples from off-season times for each of the low temperature oversampling locations listed in Table A in [S7 Appendix](#), together with 1000 randomly chosen input samples from each training location.

Variation model training with augmented data. In addition to the HI and LO training subsets, we define the HI LO training subset as the combination of the two; this training subset contains the 1000 randomly chosen input samples from each location, the 1000 randomly chosen HI samples, and the 1000 randomly chosen LO samples. Then we retrain the base models (FF, LSTM, and GRU) using all three new training subsets. We use the same training process as described in [Model training](#). Thus, in addition to the three base models we have 9 variant models, named according to the combination of base model and training subset: FF HI, FF LO, FF HI LO, LSTM HI, LSTM LO, LSTM HI LO, GRU HI, GRU LO, and GRU HI LO.

Post-processing and evaluation

As mentioned above, for each 90-day input sample, the neural network models output the number of gravid female mosquitoes expected on the 90th day. We evaluate these models by first generating the 2012–2020 abundance curves for each of the testing locations. We then smooth each time series of daily predictions, and assess both the global and seasonal fits compared to the corresponding MoLS abundance curve.

Data smoothing. Because the output of MoLS is smoothed with two passes of a 15-day moving average filter, we also smooth the ANN time series before evaluating the performance of these models. This is necessary because the weather data and thus the ANN outputs are noisy at the daily scale. We decided not to train the ANNs on the unsmoothed output of MoLS because the latter is an average over a small number (30) of stochastic simulations and the smoothing contributes to producing estimates that represent average abundance. Instead, we expect the ANNs to process daily weather data in a way that produces estimates that fluctuate daily about a time-dependent mean that is as close as possible to MoLS numbers. We use a Savitzky-Golay filter with a window of 11 and polynomial of order 3 to filter the neural network time series. This is an optimal setting that results in having the 11 day auto-correlation of the predictions within 1% of the 11 day auto-correlation of the corresponding MoLS data. The central point of the 3rd order polynomial curve used to fit each 11-day span is returned as the smoothed data point. Any negative values resulting from the smoothing process are set to 0. [Fig 4](#) shows a comparison of the raw and smoothed abundance curves for Avondale, Arizona in 2020. The reader is referred to [\[34\]](#) for more information on the Savitzky-Golay filter.

Performance metrics. We use a range of metrics to assess the performance of the neural network models. These include four metrics that quantify global fit to the MoLS data, as well as four metrics that focus on timing of abundance peaks and season length. These metrics are then combined into a single score that is used to rank the neural network models in [Comparative model performance](#).

The global fit metrics are the non-negative coefficient of determination (R_+^2), normalized root mean square error (*NRMSE*), relative difference in area under the curve (*Rel. AUC Diff.*), and Pearson correlation (r). Definitions are provided in [S3 Appendix](#). While R_+^2 and r quantify the fit of the predicted abundance curves, *NRMSE* and *Rel. AUC Diff.* quantify the overall accuracy of the magnitudes of the predicted abundances. High R_+^2 and r scores indicate the neural network abundance curves match the shape of the true MoLS curves and low *NRMSE* and *Rel. AUC Diff.* values indicate the magnitudes of the abundance predictions from the neural network models are similar to the corresponding MoLS abundance predictions. These metrics are

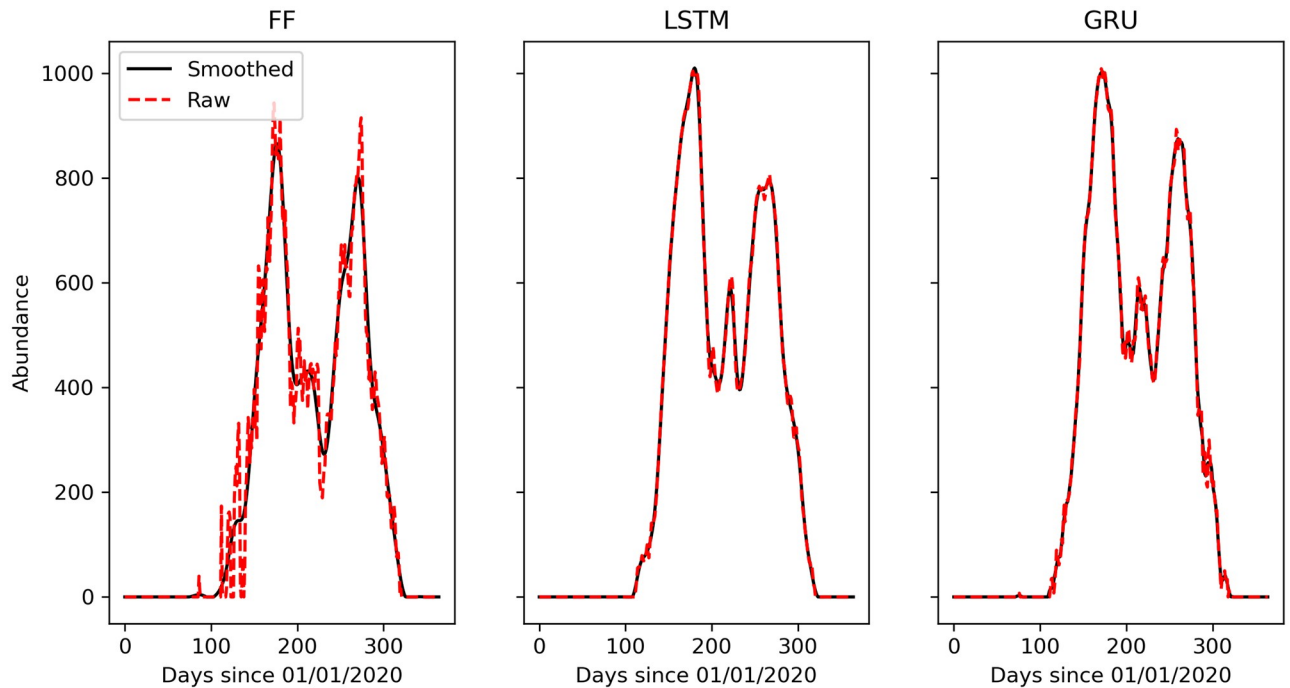


Fig 4. Comparison of the raw and smoothed abundance curves in Avondale, Arizona (2020) for the FF (left), LSTM (center), and GRU (right) models. Data are smoothed according to Data smoothing.

<https://doi.org/10.1371/journal.pcbi.1009467.g004>

computed for output samples of varying sizes, such as the entire testing output vector ($n = 100, 630$) and the output vector for a specific testing location and year ($n = 365$). We use a normalized *RMSE* to facilitate comparisons between output samples of different scales (i.e. locations with high mosquito abundance and those with low mosquito abundance) and the non-negative R_+^2 to assign a score of 0 to all low-performing models.

The season feature metrics quantify differences in the observed time-frames at which certain thresholds of mosquito abundance are reached for the target MoLS data and the ANN reconstructions. These are calculated in two steps: given a threshold T , we first identify a set of time intervals $(i_m, j_m)_T$ when MoLS estimates stably remain above T . Similarly, we identify intervals $(\hat{i}_n, \hat{j}_n)_T$ when the ANN estimates remain above T . See [S3 Appendix](#) for details. We then quantify the agreement between the two sets of intervals by calculating the differences $D_{on} \approx \hat{i}_n - i_m$ and $D_{off} \approx \hat{j}_n - j_m$ in onset values (when MoLS and the ANN predictions first remain above T) and offset values (when predictions return below T). We use approximation symbols here because multiple intervals need to be properly matched to one another in order to calculate D_{on} and D_{off} . Details are provided in [S3 Appendix](#).

To assess the ability of each ANN to reproduce the results of MoLS, we choose to generate results for thresholds T that are proportional to MoLS data; we test predictions for thresholds at 20%, 40%, 60%, and 80% of MoLS peak height to capture differences in both peak timing and season length. This process is illustrated in [Fig 5](#) for the GRU model in two locations: Collier County, Florida and Avondale, Arizona. For a given year, the season length ℓ_S is defined as $\ell_S = \max(j_m) - \min(i_m)$, where the j_m and i_m time points are calculated for $T = 20\%$ of the MoLS peak value. Onsets and offsets are then scaled by the average of ℓ_S over the testing years at the given location, to make errors relative to the environment in which they occur; if the location in consideration has a longer average season length, the relative seasonal difference

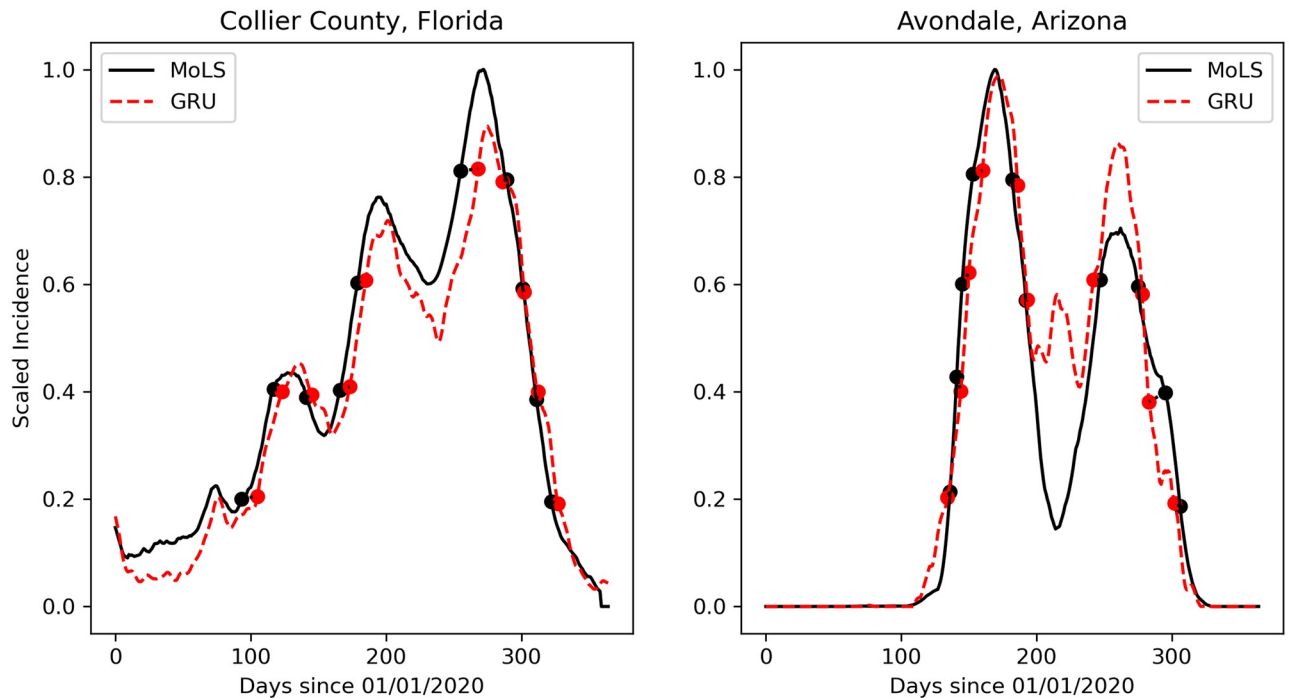


Fig 5. Observed (black solid curve) and predicted (red dashed curve for the GRU model) scaled abundance in Collier County, FL (left) and Avondale, AZ (right). To make it easier to visualize thresholds, each trace is scaled to the peak height of the observed (MoLS) abundance. The dots mark the times when each time series reached 20%, 40%, 60%, and 80% of the maximum MoLS abundance. Points in matching pairs are connected by dotted lines, whose projection on the horizontal axis has length D_{on} or D_{off} . Black (resp. red) dots that are not matched to a red (resp. black) dot are omitted in this figure for clarity.

<https://doi.org/10.1371/journal.pcbi.1009467.g005>

will be lower than the same absolute seasonal difference for a location with a shorter average season length. For each combination of model, threshold value, and testing location we report the means \bar{D} and standard deviations $\sigma(D)$ of (the scaled) D_{on} and D_{off} values over all years. Additionally, we record the total number of times each neural network model did not reach the selected threshold.

Finally, we introduce a composite score, S , defined in [S3 Appendix](#). This number, which takes into account performance on global and seasonal metrics, provides a convenient way of comparing models, and is used to rank the models in [Comparative model performance](#). It penalizes errors in mean values as well as error variability. Low values of S are associated with what the authors view as good overall ability for an ANN model to reproduce MoLS results.

Results

Performance of base models

During the model development process we use a MSE loss function to train the model parameters and R^2 accuracy to assess overall performance. As described in [Loss function and hyperparameter selection](#), we vary the hyperparameters, mimicking a grid search, until we observe diminishing returns on validation performance. The performance metrics for our trained models with the final selected hyperparameter values are shown in [Table 2](#) for each of the three training, validation, and testing subsets (defined in [Data processing](#)). The output abundance curves are smoothed according to [Data smoothing](#), and the R^2 and $RMSE$ metrics for each subset are calculated on the entire output vectors. In particular, the training metrics

Table 2. RMSE and R^2 metrics for the training, validation, and testing data subsets. The best performing values of RMSE and R^2 for each subset are in bold.

Model	Metric	Training	Validation	Testing
Baseline	RMSE	481.118	466.243	508.648
	R^2	0.558	0.551	0.558
FF	RMSE	131.002	158.352	151.999
	R^2	0.967	0.948	0.961
LSTM	RMSE	75.783	135.0	131.088
	R^2	0.989	0.962	0.971
GRU	RMSE	62.788	131.073	116.376
	R^2	0.992	0.965	0.977

<https://doi.org/10.1371/journal.pcbi.1009467.t002>

reflect performance on the entire training subset, even though only 1000 input samples per location were used during the training process (Data processing). Although the RMSE values may seem high, the corresponding mean absolute errors are around half as large. Thus, on the testing subset the GRU model differs from MoLS by an absolute average of ≈ 57 mosquitoes per day, a reasonable value when compared to MoLS abundance peak heights, which range from several hundreds in Arizona to several thousands in Florida (MoLS time series for a period of 9 years at the testing locations are provided in [S9 Appendix](#)). Later, we use the NRMSE metric, which estimates error relative to local abundance values.

[Table 2](#) shows that the ANN models perform well at replicating the mosquito abundance predictions of MoLS, with the GRU model being the best performer overall. The gap between the training and validation performance indicates slight over-fitting to the training subset, but the results of training and validation are satisfactory, with R^2 values above 0.96 and 0.94 respectively. Further, R^2 values greater than 0.96 are achieved on the testing subset. The performance of the baseline model (first row of [Table 2](#)) is clearly sub-par. This is not surprising since MoLS predictions result from a complex process that is unlikely to be captured by a linear model. In what follows, we provide a detailed analysis of the ANN models performance on the testing subset, using the metrics defined in Performance metrics. The results are expected to be representative of what a future user would experience, since they apply to data that were not used during the training and validation process.

[Table 3](#) expands on [Table 2](#) and estimates the global fit metrics for the three models. The sample means reported in the R^2_{\pm} column are lower than the R^2 score in [Table 2](#) because the present analysis is performed at a more granular level, for each location and year, rather than over the entire testing subset. The performance of ANN models is still quite good overall, but the relatively large standard deviations associated with the *Rel. AUC Diff* metric indicate variability in the way their output fits MoLS results. [Fig 6](#) illustrates the nature of this variability by

Table 3. Global fit metrics calculated for the testing subset. The arrows point in the direction of more desirable magnitudes. The entries for metric * are formatted as $\bar{x} \pm \sigma(*)$ where \bar{x} and $\sigma(*)$ are the mean and standard deviation calculated over all locations and years. In each column, the entry in bold has the best performing mean. See Performance metrics and [S3 Appendix](#) for a description of the metrics.

Model	Metric			
	$R^2_{\pm} \uparrow$	NRMSE \downarrow	Rel. AUC Diff. \downarrow	$r \uparrow$
Baseline	0.301 \pm 0.342	0.381 \pm 0.27	-1.077 \pm 1.294	0.737 \pm 0.296
FF	0.871 \pm 0.142	0.112 \pm 0.073	0.09 \pm 0.239	0.961 \pm 0.038
LSTM	0.916 \pm 0.118	0.088 \pm 0.07	0.031 \pm 0.224	0.974 \pm 0.028
GRU	0.923 \pm 0.119	0.083 \pm 0.071	0.036 \pm 0.207	0.975 \pm 0.029

<https://doi.org/10.1371/journal.pcbi.1009467.t003>

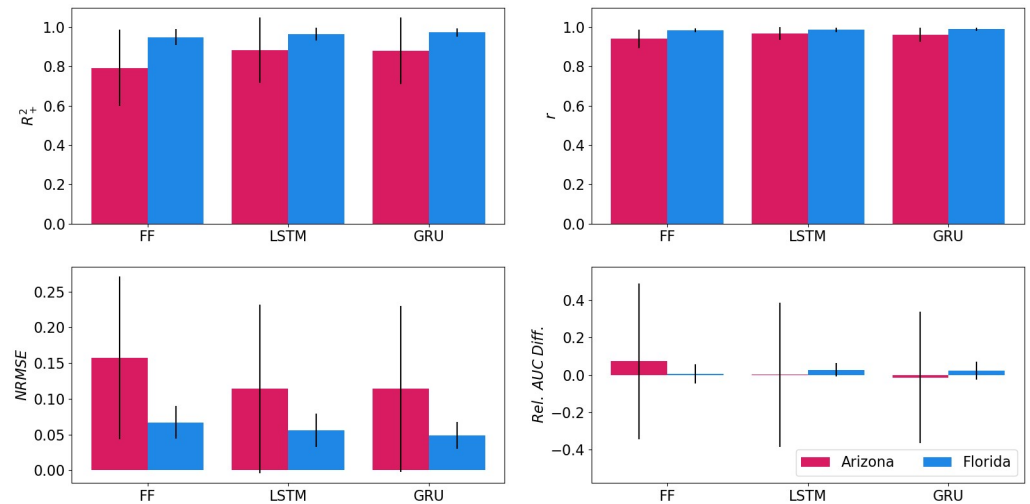


Fig 6. Global fit metrics for Arizona (red, left-hand columns) and Florida (blue, right-hand columns). The thin vertical lines have length equal to two sample standard deviations. In each state, all locations and years available in the testing subset were used. See Performance metrics and [S3 Appendix](#) for a description of the metrics.

<https://doi.org/10.1371/journal.pcbi.1009467.g006>

comparing model performance in two states with low (Arizona) and high (Florida) abundance numbers. The consistent scores in the latter (small standard deviations for all metrics) and differences between the two states (worse performing mean values associated with larger standard deviations in Arizona) suggest that temporal-variability plays less of a role here than location-variability.

We now turn to the season fit metrics, which explicitly capture deviations in the timing of the predicted season onset, offset, and peaks (see [Fig 5](#)). [Table 4](#) shows these metrics for Arizona and Florida; as in [Fig 6](#), averages are taken over all locations and years in each state. Although the sample means of D_{on} and D_{off} are typically low, they are better in Florida (less than a few percents of the season length) than Arizona (up to 9% of the season length). Moreover, the standard deviations for Arizona are again larger than for Florida, especially at the 60% and 80% thresholds. This suggests that ANNs have more difficulties capturing the timing of peaks than season onsets and offsets (corresponding to the 20% threshold). A similar trend is observed in the first three rows (above the double line) of [Table A](#) in [S4 Appendix](#), which shows the means and standard deviations of D_{on} and D_{off} over all locations and years in the testing subset.

All of the ANN models are trained on samples randomly selected from the training subset. As a consequence, different realizations of the same model, trained with the same hyperparameters but with different samples, will produce slightly different results. To illustrate this variability, [S6 Appendix](#) provides performance results for the three base models trained either on a different, or on a larger, set of samples. The stable performance, as documented by the various tables presented in this appendix, suggests that the default hyperparameter values chosen during model development are appropriate.

The above analysis reveals that lower model performance may be associated with a lack of ability to capture the timing of abundance peaks. The next section explores whether training the models on augmented datasets that specifically address how they respond to high and low temperatures improves consistency.

Table 4. Season feature metrics calculated for Arizona and Florida testing locations. Seasonal differences for a location and year are scaled by the average length of the season at the 20% threshold. The entries, formatted as $\bar{D} \pm \sigma(D)$, are calculated over all locations and years in each state. Bold entries correspond to the lowest values of $|\bar{D}| \cdot \sigma(D)$ for each threshold, with $D = D_{on}$ or D_{off} . See Performance metrics and S3 Appendix for a description of the metrics.

Arizona					
Model	Metric	Threshold (% of Max MoLS Prediction)			
		20%	40%	60%	80%
FF	D_{on}	-0.011 ± 0.051	-0.021 ± 0.055	-0.055 ± 0.086	-0.028 ± 0.206
	D_{off}	0.0 ± 0.066	0.015 ± 0.078	0.06 ± 0.126	0.092 ± 0.2
LSTM	D_{on}	-0.004 ± 0.056	-0.006 ± 0.052	-0.03 ± 0.072	-0.043 ± 0.09
	D_{off}	-0.005 ± 0.036	0.0 ± 0.094	0.01 ± 0.115	0.043 ± 0.108
GRU	D_{on}	-0.011 ± 0.049	-0.013 ± 0.051	-0.038 ± 0.111	-0.032 ± 0.138
	D_{off}	-0.008 ± 0.047	0.0 ± 0.087	0.009 ± 0.134	0.012 ± 0.141

Florida					
Model	Metric	Threshold (% of Max MoLS Prediction)			
		20%	40%	60%	80%
FF	D_{on}	-0.019 ± 0.04	-0.008 ± 0.032	0.0 ± 0.031	0.001 ± 0.06
	D_{off}	0.009 ± 0.029	0.005 ± 0.039	-0.007 ± 0.048	-0.032 ± 0.052
LSTM	D_{on}	-0.013 ± 0.037	-0.014 ± 0.021	-0.023 ± 0.031	-0.029 ± 0.05
	D_{off}	-0.003 ± 0.022	-0.003 ± 0.026	-0.004 ± 0.023	-0.01 ± 0.046
GRU	D_{on}	-0.017 ± 0.039	-0.012 ± 0.02	-0.011 ± 0.024	-0.02 ± 0.036
	D_{off}	-0.001 ± 0.018	-0.006 ± 0.033	-0.003 ± 0.016	-0.008 ± 0.038

<https://doi.org/10.1371/journal.pcbi.1009467.t004>

Performance of variant models

The augmentation methods only lead to nominal improvement in the global fit metrics, except possibly for lower *Rel. AUC Diff.* values associated with the HI versions of each model. This is illustrated in Fig 7, which shows these metrics, calculated for all locations and years in the testing subset, for all models. In addition, the results of the seasonal feature analysis shown in Table A in S4 Appendix. indicate slight performance improvement for the GRU variants, in particular at high threshold values.

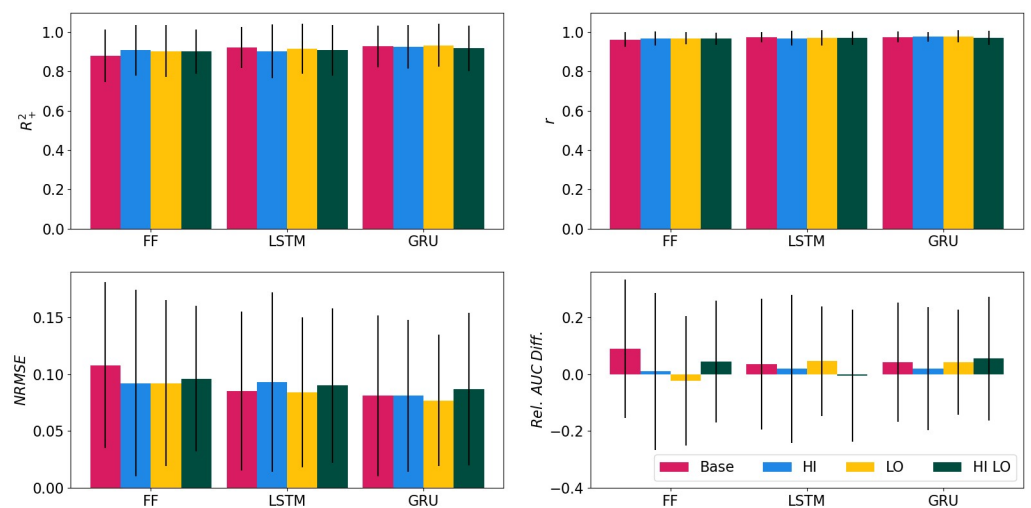


Fig 7. Average performance metrics and standard deviation of the testing locations. The legend is in the bottom right panel.

<https://doi.org/10.1371/journal.pcbi.1009467.g007>

S5 Appendix contains a case study of the GRU variants for Avondale, Arizona and Collier County, Florida. We show the 2020 abundance curves, as well as associated global fit and seasonal feature metrics. The case study exemplifies differences in model variants and performance between the two locations. In particular, the HI variant is able to capture the dip in abundance due to hot summer temperatures. It should be noted however that variations in the abundance curves produced by the ANNs are minimal when compared to changes in MoLS dynamics due to weather data (see Fig B in S1 Appendix) or even location (Fig C in S1 Appendix).

Comparative model performance

The overall metric defined in S3 Appendix combines the global fit and season feature metrics into a single score. It provides a balanced picture of the performance of each model by taking into account accuracy in terms of season abundance, season length, and peak timing. We show these results in Fig 8. The GRU, which was the best of the three base models, is only outperformed by its HI and LO variants. Similarly, the FF model is outperformed by all of its variants. On the other hand, the LSTM variants led to a loss of performance, as well as did most of the HI LO variants. Also evident from this figure is the significant impact of location on model performance. In particular, all models score poorly for the left-most location in Arizona (Fortuna Foothills). Further inspection of the associated time series reveals a significant decrease in MoLS abundance starting at the end of 2018 which is not matched by the ANNs. We believe this is due to a large spike in precipitation (more than 400 times the average daily value) included in the MACA data set on 10/24/2017, which is then followed by months of lower-than-average rainfall. MoLS takes into account the possibility that rain creates new habitat (pools of water) where mosquitoes can develop. These come from a reserve of eggs that are available in the environment and can hatch in newly created breeding grounds. In the case of excessive rainfall, especially in regions of low mosquito abundance, the pool of eggs may be exhausted in a single event. A drop in future abundance can ensue since not all recently

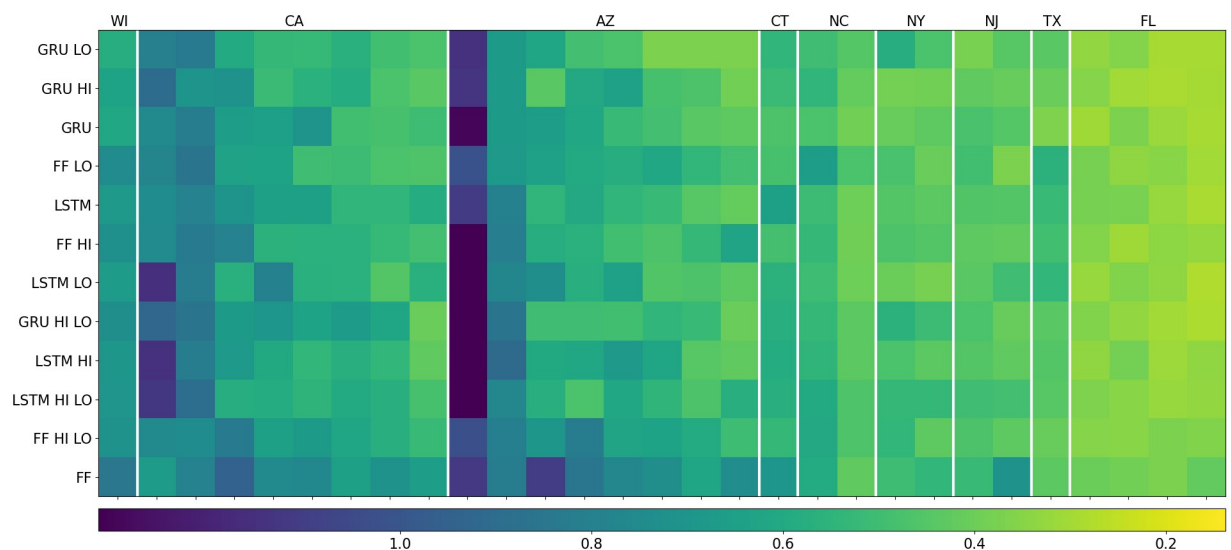


Fig 8. Combined scores on testing locations (see S3 Appendix for metric definition). The rows represent the models and are organized bottom to top from highest score (worst model) to lowest score (best model). The columns represent individual locations (see Table B in S7 Appendix for the names), and the vertical, white lines separate states. The states are organized left to right from highest mean score to lowest, and within each state the locations are organized left to right by descending score.

<https://doi.org/10.1371/journal.pcbi.1009467.g008>

hatched eggs will survive to adulthood. In addition, during periods of dry weather, a depleted reserve of eggs is likely to take a long time to rebuild to normal levels. If large precipitation events occur outside of the Δ days window used for input, the ANNs cannot be aware of them and, as a consequence, will produce results reflecting normal abundance given the local weather conditions. For reference, the time series for MoLS and the GRU HI model at the locations listed in Fig 8 are provided in S9 Appendix. In the case of Fortuna Foothills, the ANN correctly reproduces the double-peak pattern seen in MoLS results, but has higher abundance in 2019–2020. This is consistent with our proposed explanation that the conjunction of a rare high precipitation event and overall low pre-event abundance levels (less than 1000 mosquitoes at peak height) led to a crash in MoLS population estimates that is not captured by the ANNs.

S8 Appendix presents an analysis of the performance of the GRU HI model on the Capital Cities dataset, which covers 9 years of data over 44 locations. The map of Fig C in S8 Appendix indicates that the model works very well in the eastern and southeastern regions of the United States, but has inferior performance in the west. Arizona and California are not included because the corresponding locations (Phoenix and Sacramento) are in the principal dataset (see Fig 8 for results). In addition, we observe a strong correlation between abundance and performance: regions of low or irregular MoLS abundance are typically associated with worse ANN performance. This is illustrated in the top row of Fig D in S8 Appendix, for Nevada and Montana, which have abundance peak heights of only a few hundred mosquitoes.

Discussion

MoLS, a stochastic agent-based model for *Aedes aegypti* abundance that was validated against surveillance data in Puerto Rico [9], uses weather data to simulate the life cycle of a large number of mosquitoes and estimate expected daily abundance. It is natural to ask whether a properly trained artificial neural network is able to “learn” how MoLS combines weather-dependent development, survival, and reproductive rates to make its predictions. In this paper we demonstrate that it is possible to train ANNs that map meteorological data to mosquito numbers in a way consistent with MoLS results. Although the 12 models considered here achieve varying levels of success, they are generally able to replicate the trends observed in MoLS time series, indicating a neural network can function as an equation-free model of *Aedes aegypti* abundance.

As shown by the sub-par performance of the baseline model, learning how MoLS functions requires a more complex setup than a linear regression. While all three base ANN models use the same architecture, shown in Fig 2, incorporating recurrent layers (LSTM and GRU layers) improves performance, compared to the FF model (Table 2). This suggests that a model using the spatial feature extraction of convolution layers alone is unable to fully identify the relationship between weather features and mosquito abundance, and that combining the sequential “memory” feature of recurrent layers with the convolution layers better captures this relationship.

The metrics shown in Table 3 and Table A in S4 Appendix indicate all ANN models have high overall skill, with minimal differences in global level performance between them. At a more granular level, the case study of S5 Appendix suggests the GRU HI model is better able to capture abundance in hot summer months. Such an improvement is expected to be reflected in the composite metric of S3 Appendix, which by design is sensitive to variability in local performance. Indeed, Fig 8 indicates the HI and LO data augmentation methods improve the performance of the GRU and FF models, although not that of the LSTM model. It is not clear at this point why the HI LO models are inferior to their HI and LO counterparts. The most

probable explanation is that the data augmentation puts too much emphasis on extremes compared to typical temperatures, thereby lowering performance in generic situations.

The performance of the GRU HI model on the Capital Cities dataset (S8 Appendix) reveals that ANNs perform well in regions where mosquito abundance is high—which is principally where they would be expected to be used. Each year of abundance predictions presented in this paper took an ANN only 0.33 seconds to generate using a laptop with a 1.6 GHz Dual-Core Intel Core i5 processor with 8GB RAM (compare to the 10 mn needed by MoLS to generate 10 years of abundance estimates on a single HPC core) and approximately 1–2 minutes to train per epoch for each model using a laptop with Intel Core i7–9750H CPUs with 16GB RAM and a single Nvidia GeForce GTX 1650 GPU with 4GB memory, depending on the sampling size and model architecture. The combination of speed and accuracy demonstrated in this article therefore identifies neural network models as top contenders for efficiently converting weather data into *Aedes aegypti* and more generally mosquito abundance. To encourage such applications of ANNs, all of our code is freely available on GitHub to researchers interested in improving on the present results. However, before using an ANN as a replacement for MoLS, its performance should be assessed with metrics similar to those presented in this article and the model that best captures local circumstances (e.g. the effect of hot summers on mosquito populations in Arizona) should be selected. A quantitative comparison with actual surveillance data is also recommended. Both MoLS and its ANN replacements introduced here produce daily numbers of scaled mosquito abundance. As previously mentioned, “scaled” means that the estimates are up to a multiplicative factor that depends on location, but not on time. If surveillance data are available, the value of this factor can be found via linear regression of MoLS results against the data, as was done in [9]. The resulting estimate will depend on the type of mosquito traps used for surveillance, as well as on local considerations, such as the number of available breeding sites. Once rescaled, MoLS or ANN predictions should be able to capture overall abundance trends fairly well.

Possible applications include the use of local weather data for vector control interventions. In this case, abundance trends could be estimated on a daily basis and supplement routine surveillance; reliable weather data, as well as the best performing ANNs should be selected, to increase confidence in the results. Model limitations should also be taken into account. As suggested by the Fortuna Foothills example, extreme weather events that are localized in time but affect average population levels in the long term are not taken into account by the ANNs when such events fall outside the range of their input window of Δ days. However, if the user knows that such an event occurred, it is not difficult to recalibrate the ANNs by recalculating the scaling factor that relates their output to local surveillance data. More qualitative, longer-term planning, based on climate scenarios, should also be possible with the ANNs presented here, since estimating general, weather-based trends of mosquito abundance would suffice in that case. In addition, because *Aedes aegypti* is a known vector for diseases like dengue, chikungunya, and Zika, many studies have provided environment suitability maps for this species and have used them to estimate disease risk (see for instance [19, 35] and references therein). An ANN trained on reproducing MoLS predictions would make it possible to create similar maps from a weather-based mechanistic abundance model without the need of high power computing (HPC) typically required to generate the same quantity of predictions using MoLS. Finally, the speed afforded by ANNs could allow the creation of interactive web apps able to produce estimates of mosquito abundance from local weather data at a user’s request.

Another important application of abundance models is the development of probabilistic forecasts. Going beyond the point estimates produced by the models discussed in this article requires additional uncertainty quantification, especially in terms of the variability inherent to the local weather forecast used to make predictions (an example of how different models affect

MoLS output is provided in [S1 Appendix](#)). Looking forward, we believe that when combined with assimilation of weather and surveillance data, the ANNs trained for this article can effectively contribute to the development of probabilistic mosquito abundance forecasting models. We leave this for future work.

Supporting information

S1 Appendix. Weather and MoLS time series.
(PDF)

S2 Appendix. Description of ANN layers.
(PDF)

S3 Appendix. Performance metrics.
(PDF)

S4 Appendix. Seasonal feature analysis of all models.
(PDF)

S5 Appendix. Case study.
(PDF)

S6 Appendix. ANN variations.
(PDF)

S7 Appendix. Locations used in the principal data set for training, validation, and testing.
(PDF)

S8 Appendix. Capital cities data set.
(PDF)

S9 Appendix. MoLS-GRU HI comparison.
(PDF)

Acknowledgments

An allocation of computer time from the UA Research Computing High Performance Computing (HPC) at the University of Arizona is gratefully acknowledged.

Author Contributions

Conceptualization: Joceline Lega.

Data curation: Adrienne C. Kinney, Sean Current, Joceline Lega.

Formal analysis: Adrienne C. Kinney, Sean Current, Joceline Lega.

Methodology: Adrienne C. Kinney, Sean Current, Joceline Lega.

Validation: Adrienne C. Kinney, Sean Current, Joceline Lega.

Visualization: Adrienne C. Kinney, Sean Current, Joceline Lega.

Writing – original draft: Adrienne C. Kinney, Sean Current, Joceline Lega.

Writing – review & editing: Adrienne C. Kinney, Sean Current, Joceline Lega.

References

1. Krizhevsky A, Sutskever I, Hinton GE. ImageNet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems*. 2012; 25:1097–1105.
2. Kwon D, Kim H, Kim J, Suh SC, Kim I, Kim KJ. A survey of deep learning-based network anomaly detection. *Cluster Computing*. 2019; p. 1–13.
3. Nguyen TH, Cho K, Grishman R. Joint event extraction via recurrent neural networks. In: *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*; 2016. p. 300–309.
4. Schmidhuber J. Deep learning in neural networks: An overview. *Neural Networks*. 2015; 61:85–117. <https://doi.org/10.1016/j.neunet.2014.09.003> PMID: 25462637
5. MACA datasets. <http://www.climatologylab.org/mac.html>.
6. Taylor KE, Stouffer RJ, Meehl GA. An overview of CMIP5 and the experiment design. *Bulletin of the American Meteorological Society*. 2012; 93(4):485–498. <https://doi.org/10.1175/BAMS-D-11-00094.1>
7. Abatzoglou JT, Brown TJ. A comparison of statistical downscaling methods suited for wildfire applications. *International Journal of Climatology*. 2012; 32(5):772–780. <https://doi.org/10.1002/joc.2312>
8. Abatzoglou JT. Development of gridded surface meteorological data for ecological applications and modelling. *International Journal of Climatology*. 2013; 33(1):121–131. <https://doi.org/10.1002/joc.3413>
9. Lega J, Brown HE, Barrera R. *Aedes aegypti* (Diptera: Culicidae) abundance model improved with relative humidity and precipitation-driven egg hatching. *Journal of Medical Entomology*. 2017; 54(5):1375–1384. <https://doi.org/10.1093/jme/tjx077> PMID: 28402546
10. World Health Organization, UNICEF. Global vector control response 2017-2030. 2017.
11. Ong A, Sandar M, Chen MI, Sin LY. Fatal dengue hemorrhagic fever in adults during a dengue epidemic in Singapore. *International Journal of Infectious Diseases*. 2007; 11(3):263–267. <https://doi.org/10.1016/j.ijid.2006.02.012> PMID: 16899384
12. Cauchemez S, Ledrans M, Poletto C, Quenel P, de Valk H, Colizza V, et al. Local and regional spread of chikungunya fever in the Americas. *Eurosurveillance*. 2014; 19(28). <https://doi.org/10.2807/1560-7917.es2014.19.28.20854> PMID: 25060573
13. Heukelbach J, Alencar CH, Kelvin AA, de Oliveira WK, de Góes Cavalcanti LP. Zika virus outbreak in Brazil. *The Journal of Infection in Developing Countries*. 2016; 10(02):116–120. <https://doi.org/10.3855/jidc.8217> PMID: 26927450
14. Ooi EE, Goh KT, Gubler DJ. Dengue prevention and 35 years of vector control in Singapore. *Emerging Infectious Diseases*. 2006; 12(6):887. <https://doi.org/10.3201/10.3201/eid1206.051210> PMID: 16707042
15. Barrera R, Acevedo V, Felix GE, Hemme RR, Vazquez J, Munoz JL, et al. Impact of autocidal gravid ovitraps on chikungunya virus incidence in *Aedes aegypti* (Diptera: Culicidae) in areas with and without traps. *Journal of Medical Entomology*. 2017; 54(2):387–395. <https://doi.org/10.1093/jme/tjw187> PMID: 28031347
16. World Health Organization. Mosquito (vector) control emergency response and preparedness for Zika virus. 2016. Geneva: WHO. 2016.
17. Ryan PA, Alsemgeest D, Gatton ML, Kay BH. Ross River virus disease clusters and spatial relationship with mosquito biting exposure in Redland Shire, Southern Queensland, Australia. *Journal of Medical Entomology*. 2006; 43:1042–1059. <https://doi.org/10.1093/jmedent/43.5.1042> PMID: 17017245
18. Guo S, Ling F, Hou J, Wang J, Fu G, Gong Z. Mosquito surveillance revealed lagged effects of mosquito abundance on mosquito-borne disease transmission: a retrospective study in Zhejiang, China. *PLoS One*. 2014; 9:e112975. <https://doi.org/10.1371/journal.pone.0112975> PMID: 25393834
19. Kraemer MU, Sinka ME, Duda KA, Mylne AQ, Shearer FM, Barker CM, et al. The global distribution of the arbovirus vectors *Aedes aegypti* and *Ae. albopictus*. *eLife*. 2015; 4:e08347. <https://doi.org/10.7554/eLife.08347> PMID: 26126267
20. Rocklöv J, Dubrow R. Climate change: an enduring challenge for vector-borne disease prevention and control. *Nature Immunology*. 2020; 21(5):479–483. <https://doi.org/10.1038/s41590-020-0648-y> PMID: 32313242
21. Kamal M, Kenawy MA, Rady MH, Khaled AS, Samy AM. Mapping the global potential distributions of two arboviral vectors *Aedes aegypti* and *Ae. albopictus* under changing climate. *PLoS One*. 2018; 13(12):e0210122. <https://doi.org/10.1371/journal.pone.0210122> PMID: 30596764
22. Hemme RR, Thomas CL, Chadde DD, Severson DW. Influence of Urban Landscapes on Population Dynamics in a Short-Distance Migrant Mosquito: Evidence for the Dengue Vector *Aedes aegypti*. *PLOS Neglected Tropical Diseases*. 2010; 4(3):1–9. <https://doi.org/10.1371/journal.pntd.0000634>

23. Brown JE, Evans BR, Zheng W, Obas V, Barrera-Martinez L, Egizi A, et al. Human impacts have shaped historical and recent evolution in *Aedes aegypti*, the dengue and yellow fever mosquito. *Evolution*. 2014; 68(2):514–525. <https://doi.org/10.1111/evo.12281> PMID: 24111703
24. Wilke ABB, Vasquez C, Carvajal A, Medina J, Chase C, Cardenas G, et al. Proliferation of *Aedes aegypti* in urban environments mediated by the availability of key aquatic habitats. *Scientific Reports*. 2020; 10(1):12925. <https://doi.org/10.1038/s41598-020-69759-5> PMID: 32737356
25. Halstead SB. Dengue Virus–Mosquito Interactions. *Annual Review of Entomology*. 2008; 53(1):273–291. <https://doi.org/10.1146/annurev.ento.53.103106.093326> PMID: 17803458
26. Valdez LD, Sibona GJ, Condat CA. Impact of rainfall on *Aedes aegypti* populations. *Ecological Modelling*. 2018; 385:96–105. <https://doi.org/10.1016/j.ecolmodel.2018.07.003>
27. Benitez EM, Estallo EL, Grech MG, Frías-Céspedes M, Almirón WR, Robert MA, et al. Understanding the role of temporal variation of environmental variables in predicting *Aedes aegypti* oviposition activity in a temperate region of Argentina. *Acta Tropica*. 2021; 216:105744. <https://doi.org/10.1016/j.actatropica.2020.105744> PMID: 33189713
28. Morin CW, Comrie AC. Modeled response of the West Nile virus vector *Culex quinquefasciatus* to changing climate using the dynamic mosquito simulation model. *International Journal of Biometeorology*. 2010; 54(5):517–529. <https://doi.org/10.1007/s00484-010-0349-6> PMID: 20683620
29. CDC Aedes Challenge. <https://predict.cdc.gov/post/5c4f6d687620e103b6dcd015>.
30. Maciel-de Freitas R, Torres Codeço C, Lourenço-de Oliveira R. Daily survival rates and dispersal of *Aedes aegypti* females in Rio de Janeiro, Brazil. *The American Journal of Tropical Medicine and Hygiene*. 2007; 76:659–665. <https://doi.org/10.4269/ajtmh.2007.76.659> PMID: 17426166
31. Kingma DP, Ba J. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980. 2014.
32. Goodfellow I, Bengio Y, Courville A. *Deep Learning*. vol. 1. MIT Press Massachusetts, USA. 2017.
33. Brown HE, Smith C, Lashway S. Influence of the Length of Storage on *Aedes aegypti* (Diptera: Culicidae) Egg Viability. *Journal of Medical Entomology*. 2017; 54:489–491. <https://doi.org/10.1093/jme/tjw186> PMID: 28011737
34. Orfanidis SJ. *Introduction to Signal Processing*. Prentice-Hall, Inc.; 1995.
35. Messina JP, Kraemer MU, Brady OJ, Pigott DM, Shearer FM, Weiss DJ, et al. Mapping global environmental suitability for Zika virus. *eLife*. 2016; 5:e15272. <https://doi.org/10.7554/eLife.15272> PMID: 27090089