

1 Results and Discussions

1.1 Simple ODEs

Overview

Consider a first-order ODE system:

$$\frac{dx}{dt} = \cos t + x^2 + y - (1 + t^2 + \sin^2 t) \quad (1)$$

$$\frac{dy}{dt} = 2t - (1 + t^2) \sin t + xy \quad (2)$$

where $t \in [0, 3]$.

Mathematics

The analytical solutions are:

$$x = \sin t \quad (3)$$

$$y = 1 + x^2 \quad (4)$$

Results

The solutions using the Neural Net solver could be found in Fig. 1. The network was trained using a grid of 20 equidistant points in $t = [0, 3]$ and 50 hidden units in the hiddlen layer.

We could see an almost exact fit with the analytical solution. In terms of training convergence, we could see that the neural net rapidly converges to the solution with just a few iterations from Fig. 2.

To enable us to evaluate the fit of the two functions in this system of ODE, we propose taking an average of the RMSE, in this case we will have:

$$RMSE_{overall} = \frac{1}{2}[RMSE_x + RMSE_y] \quad (5)$$

The fitting attained as shown in Fig. 1 has $RMSE_{overall} = 0.0177$

Note that average RMSE enables us to determine the reproducibility of a good performing model for the same ODE system, which is discussed in the next section.

Discussions

In terms of reproducibility, we repeated the same fitting process 100 times. The average RMSE histogram distribution for 100 fittings is shown in Fig. 3. 67% of the fittings are relatively close fit to the exact analytical solution.

The model does not perform as well in extrapolation. As shown in Fig. 4, the fitting is accurate only up to $t = 3.5$, i.e. 0.5 unit of time beyond the maximum time point in training.

1.2 Unstable, Periodic ODEs

1.2.1 Lotka-Volterra 2-species model

Overview

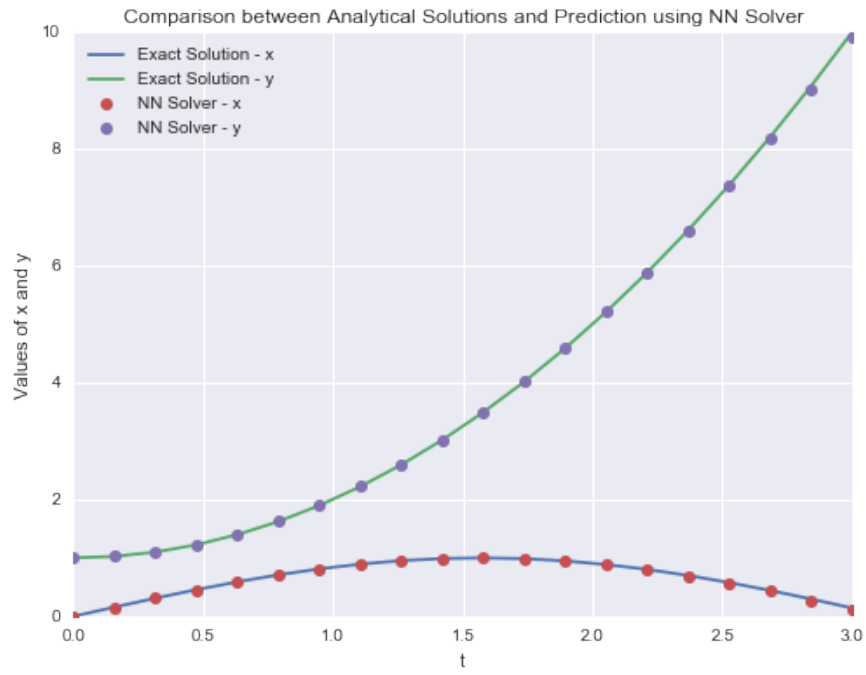


Figure 1: Comparing solutions found using neural net solver versus exact analytical solutions.

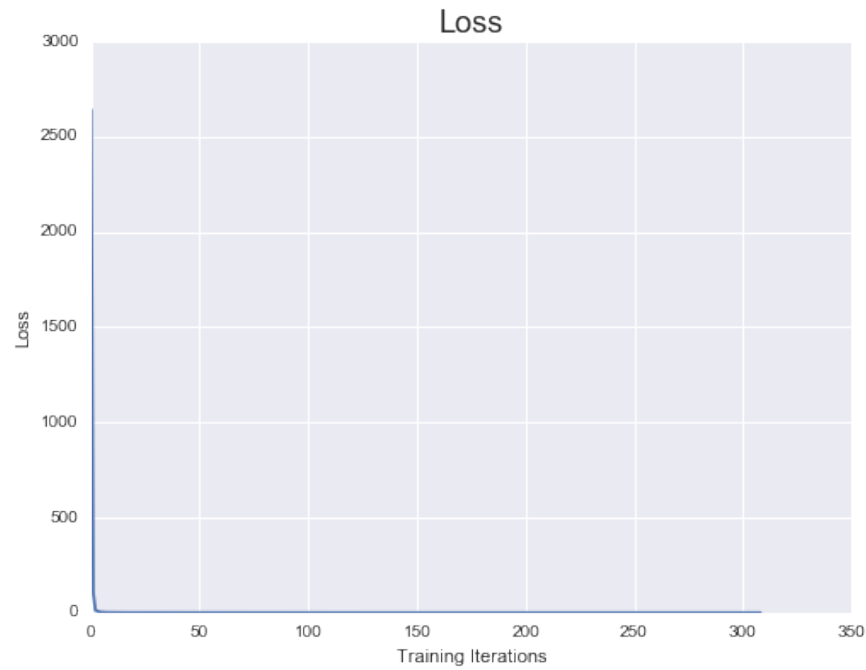


Figure 2: Convergence to final solution.

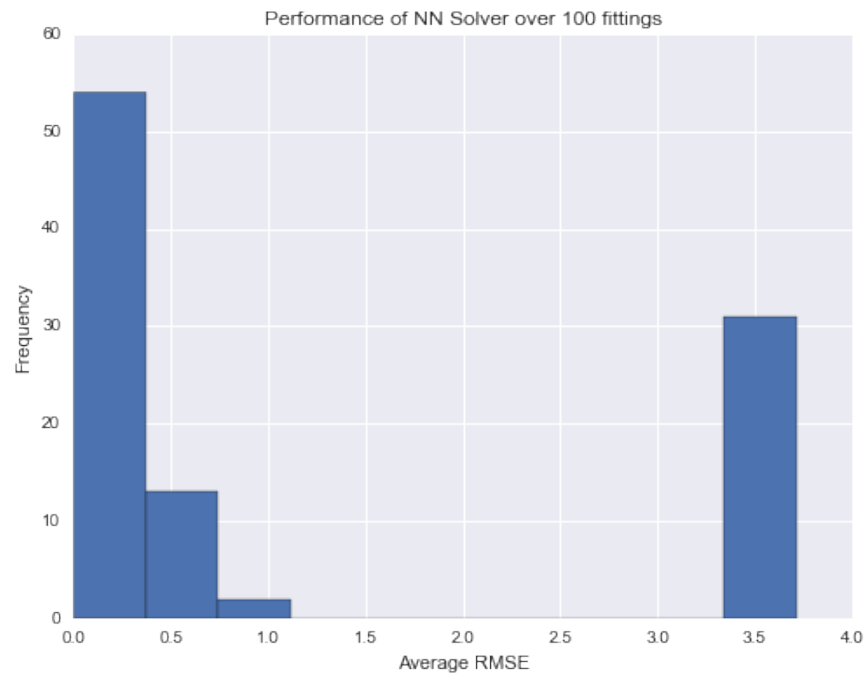


Figure 3: Average RMSE Distribution for 100 Repeated Fittings

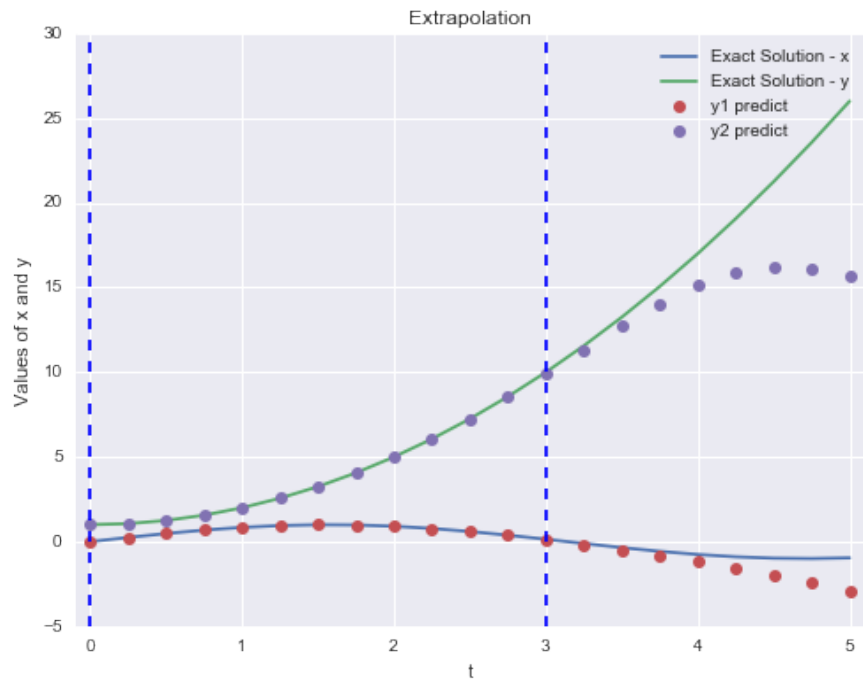


Figure 4: Extrapolation: Predicting on values beyond $t \in [0, 3]$

The Lotka-Volterra prey-predator equations are a pair of first-order, non-linear differential equations commonly used to describe the interactions between two species, the prey and predator.

Mathematics

The populations of the prey and predator can be described by the following equations:

$$\frac{dx}{dt} = ax - bxy \quad (6)$$

$$\frac{dy}{dt} = -cy + dxy \quad (7)$$

where

- x is the density of prey
- y is the density predator
- t represents time
- $a, b, c, d > 0$

Eq. 6 indicates that in the absence of a predator ($y = 0$), the prey would grow at a constant rate a , with the assumption that the prey have an unlimited food supply. The rate of predation upon the prey is assumed to be proportional to the rate at which the predators and the prey are present concurrently, represented by bxy . If either x or y is zero then no predation is possible.

Similarly, Eq. 7 shows that in the absence of prey ($x = 0$), the density of predators would decrease at a constant rate c , due to natural death or emigration. This equation assumes that the predator population only preys upon the same species of prey identified in Eq. 6. dxy represents the growth of the predator population[REF1].

The model assumes that throughout the process the external environment remains the same and none of the species is favored.[REF2]

Results

From Fig.5 we see that our neural net solver attains a close fit to the ODE solver. Training was done with a grid of 100 equidistant points in $t = [0, 10]$ and 50 hidden units in the hidden layer.

In terms of training convergence, we could see that the neural net rapidly converges to the solution with just a few iterations from Fig. 6.

The same as before, to enable us to evaluate the fit of the two functions in this system of ODEs, we propose taking an average of the RMSE, in this case we will have:

$$RMSE_{overall} = \frac{1}{2}[RMSE_{prey} + RMSE_{predator}] \quad (8)$$

For this particular plot, the overall average RMSE is at 1.568. It is found that while our proposed solver using neural net is able to provide a continuous function, the results tend to fluctuate highly. As such, we ran the same fitting over 100 times as shown in Fig. 7. 30% of the fittings result in relatively good fitting with average RMSE of less than 5.

Similar to the simple ODE system earlier, neural net model also does not perform well in extrapolation. We see from Fig. 8 that the values deviate from the predicted values using ODE solver strongly beyond $t = 10$.

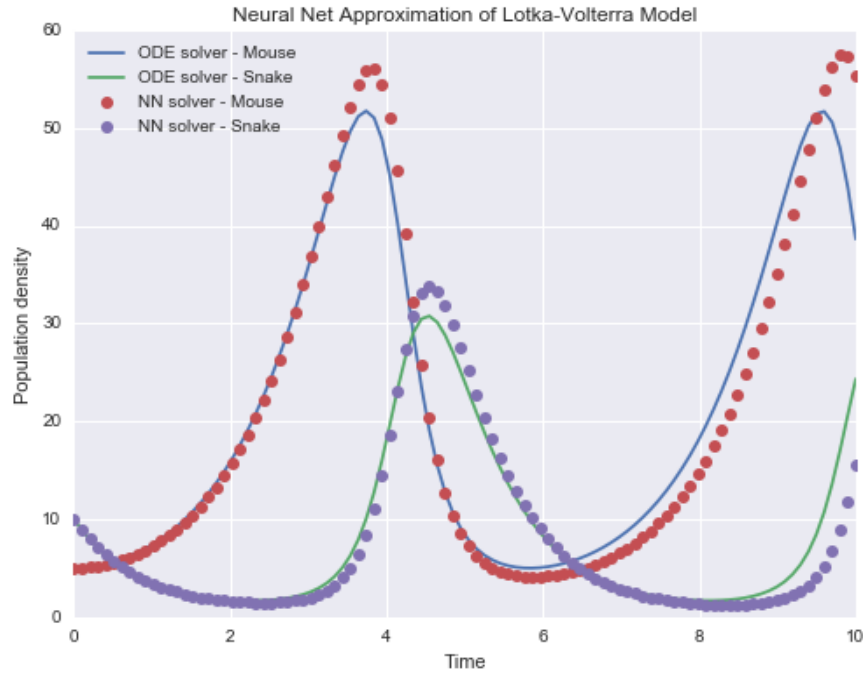


Figure 5: Comparing across results using standard ODE solver from Scipy library versus using neural net solver, at initial parameters of $a = 1, b = 0.1, c = 1.5, d = 0.75$. Average RMSE = 1.568.

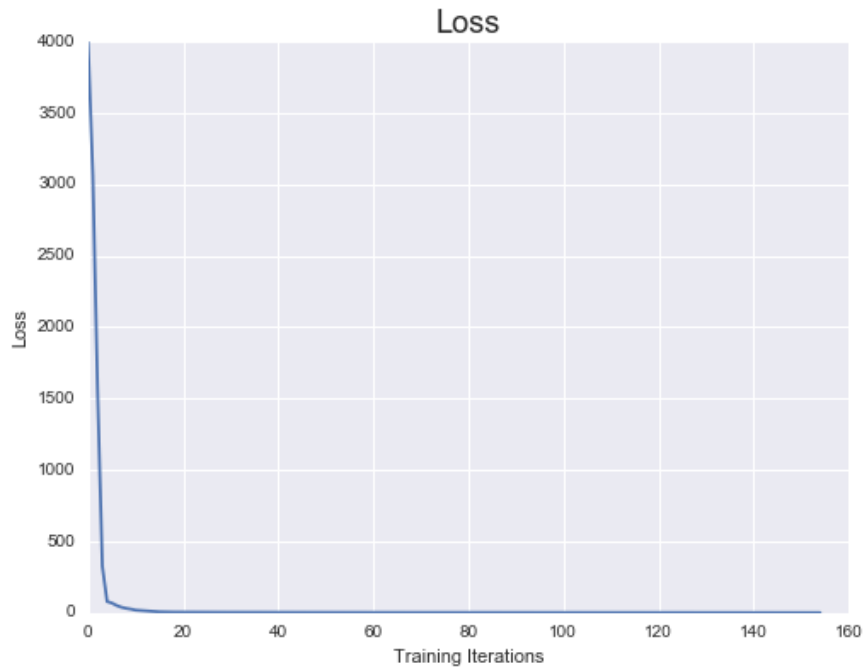


Figure 6: Convergence to final solutions for 2-species model

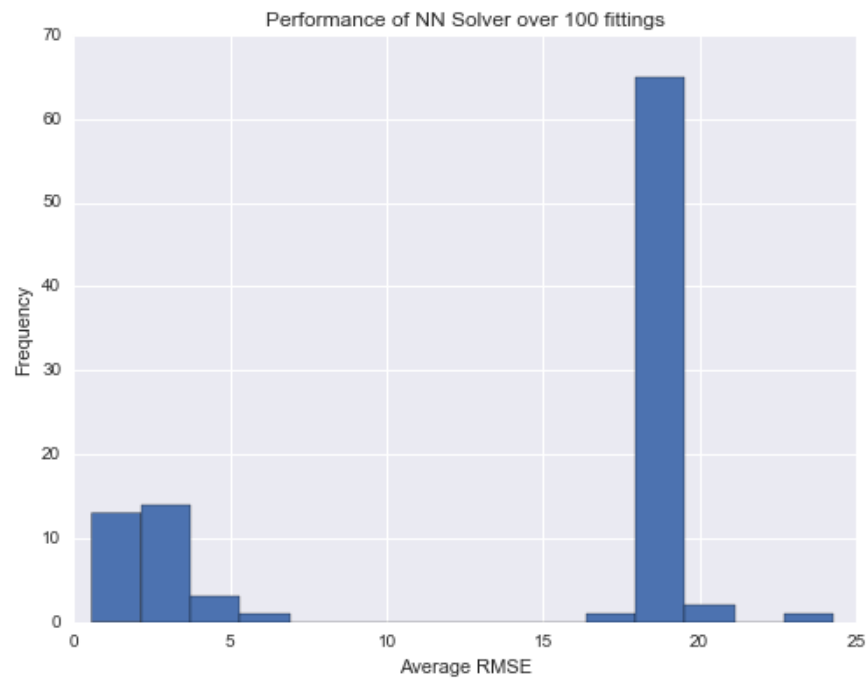


Figure 7: Running NN solver 100 times to see the performance histogram

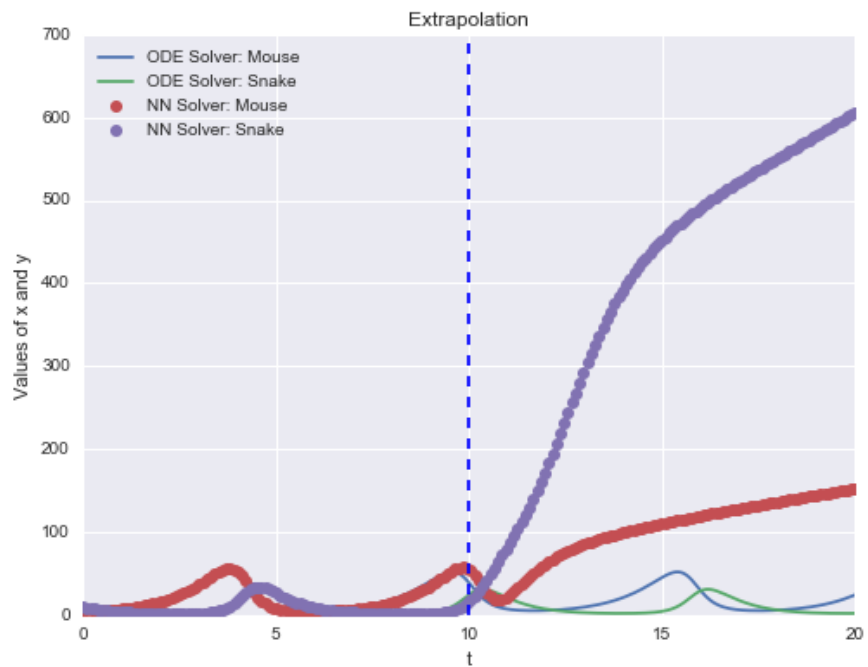


Figure 8: Extrapolation - Predicting on values beyond $t \in [0, 10]$

1.2.2 Lotka-Volterra 3-species Model

Extending to the 2-species model discussed earlier, we wish to model a linear three-species food chain where the lowest-level prey x is preyed upon by a mid-level species y , which, in turn, is preyed upon by a top level predator z . An example of a three-species food chain is mouse-snake-owl [REF3].

$$\frac{dx}{dt} = ax - bxy \quad (9)$$

$$\frac{dy}{dt} = -cy + dxy - eyz \quad (10)$$

$$\frac{dz}{dt} = -fz + gyz \quad (11)$$

where

- $a, b, c, d, e, f, g > 0$
- a, b, c, d are as in the 2-species Lotka-Volterra equations
- e represents the effect of predation on species y by species z
- f represents the natural death rate of species z in the absence of prey
- g represents the reproduction rate of species z in the presence of prey y

Results

The neutral net is trained with 50 hidden units over 100 equidistant points over $t \in [0, 10]$. From Fig. 9, we can see that the performance is worse compared to the 2-species Lotka-Volterra model. The fit is close to the solution from ODE solver only up to $t = 1$ for the three equations. No further model fitting is carried out as the fitting is poor and the average RMSE score is not a good gauge to identify strong fit.

In terms of extrapolating beyond the time range which the neural net was trained, the performance was not satisfactory. From Fig. 10, we see strong deviation from the ODE solver solution.

1.2.3 Van der Pol oscillator

Overview

Van der Pol oscillator is a non-conservative oscillator with a linear spring force and a non-linear damping force. [REF4] The equation is given by:

$$\frac{d^2x}{dt^2} - \mu(1 - x^2)\frac{dx}{dt} + x = 0 \quad (12)$$

Mathematics

Applying the Liénard transformation $y = x - \frac{x^3}{3} - \frac{1}{\mu} \frac{dx}{dt}$, the equation can be written as a system of ODE [REF5]:

$$\frac{dx}{dt} = \mu(x - \frac{1}{3}x^3 - y) \quad (13)$$

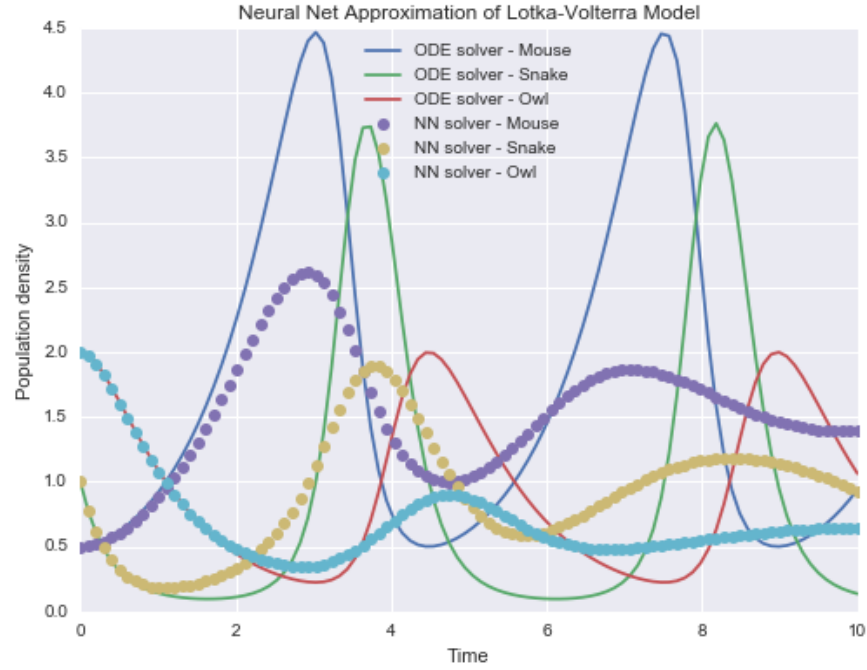


Figure 9: Comparing across results using standard ODE solver from Scipy library versus using neural net solver, at initial parameters of $a = b = c = d = e = f = g = 1$. Average RMSE score of 0.803.

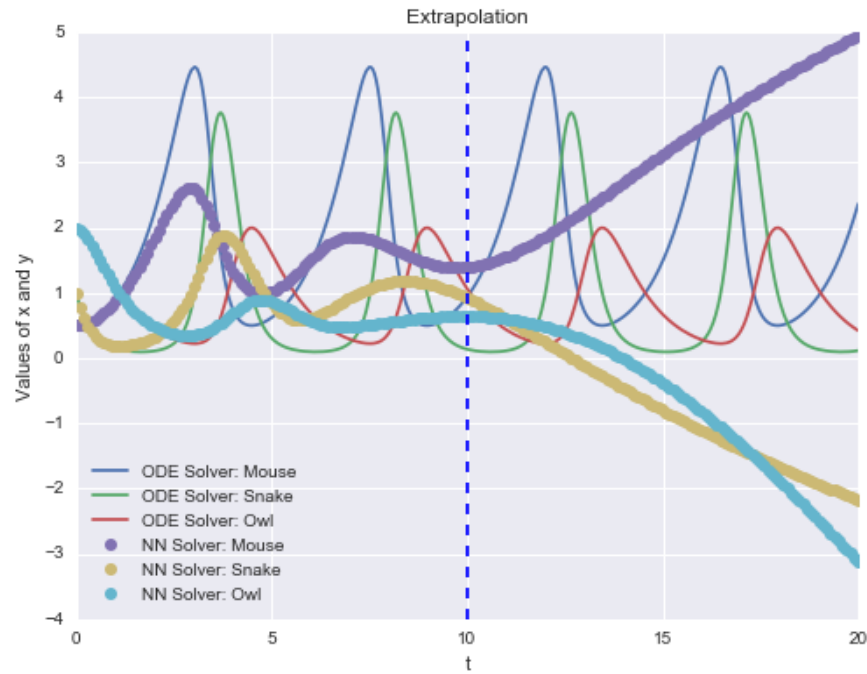


Figure 10: Predicting on values beyond $t \in [0, 10]$

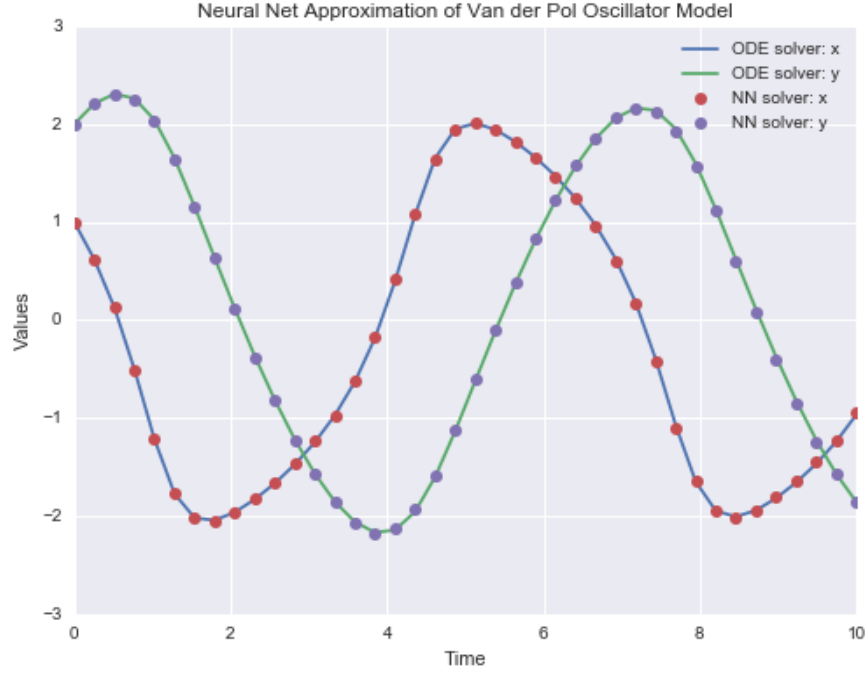


Figure 11: Running NN solver with even spacing on time points. Average RMSE of 0.0135.

$$\frac{dy}{dt} = \frac{x}{\mu} \quad (14)$$

Results

From Fig. 11 we see that our neural net solver attains a close fit to the ODE solver. Training was done with a grid of 40 equidistant points in $t = [0, 10]$ and 20 hidden units in the hidden layer.

The same as before, to evaluate the fit of the two functions in this system of ODEs, we propose taking an average of the RMSE, in this case the overall average RMSE for our fit in Fig. 11 is at 0.0135.

Since the fitting is satisfactory, we ran the same fitting over 100 times to ensure reproducibility. From Fig. 12, 63% of the fittings result in relatively good fitting with average RMSE of less than 2.5.

Similar to the simple ODE system earlier, neural net model also does not perform well in extrapolation. We see from Fig. 13 that the values deviate from the predicted values using ODE solver strongly beyond $t = 10$.

Discussions

63% of the fittings attained average RMSE score of less than 2.63.

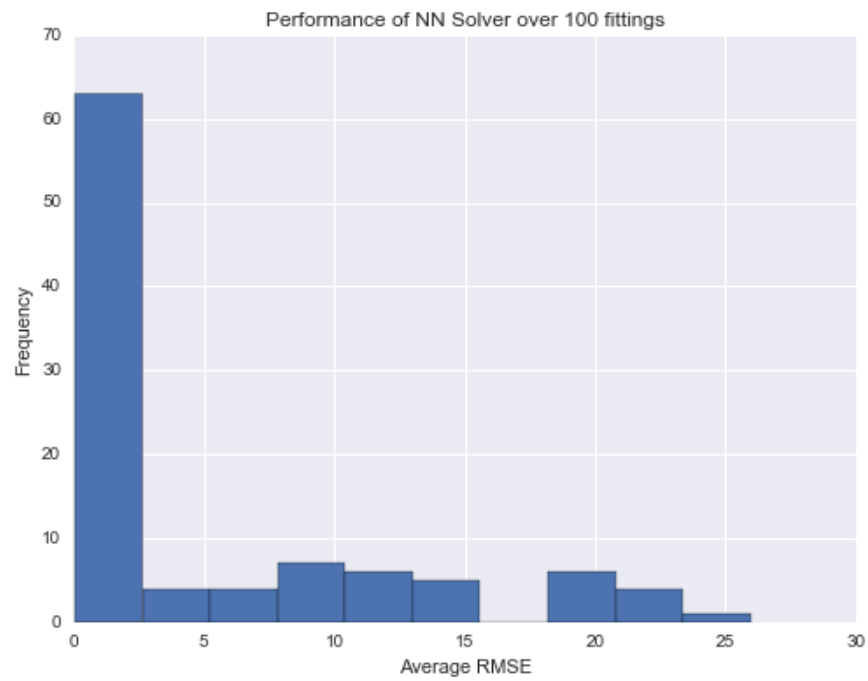


Figure 12: Running NN solver 100 times to see the performance histogram

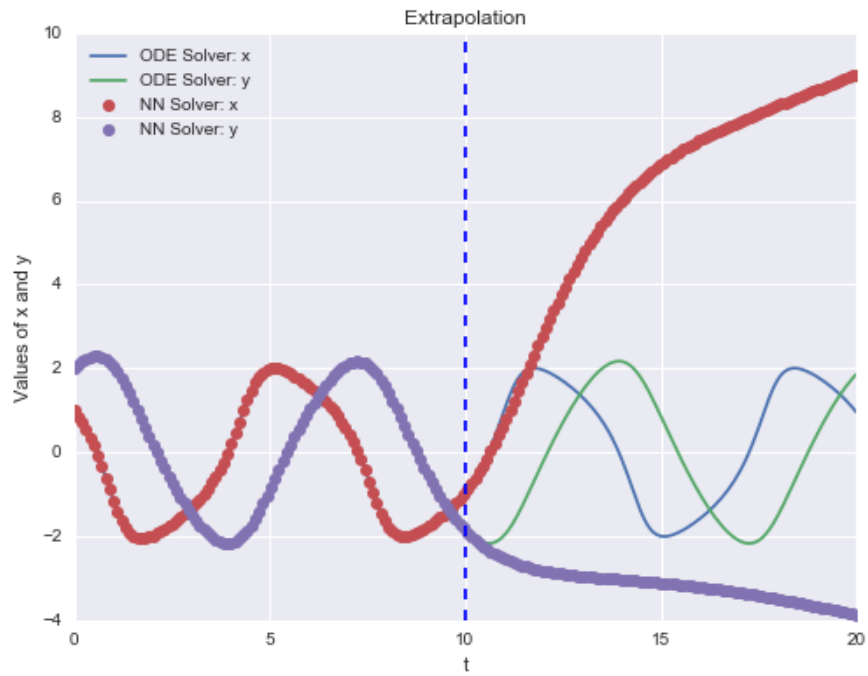


Figure 13: Running NN solver 100 times to see the performance histogram