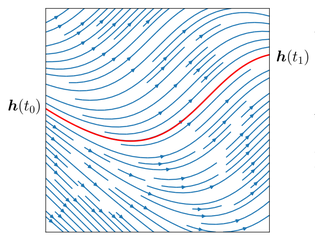
Jacobian norm regularisation and conditioning in neural ordinary differential equations

Shane Josias* & Willie Brink
Applied Mathematics, Stellenbosch University
josias@sun.ac.za, wbrink@sun.ac.za

Stellenbosch
UNIVERSITY
IYUNIVESITHI
UNIVERSITEIT

Neural ordinary differential equations



A vector $\boldsymbol{h}(t)$ follows the dynamics f:

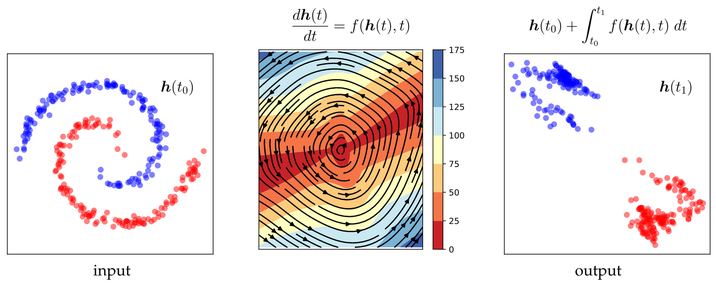
$$\frac{d\boldsymbol{h}(t)}{dt} = f(\boldsymbol{h}(t), t),$$

where $f(\boldsymbol{h}(t), t)$ is a neural network.

For an input $\boldsymbol{h}(t_0)$ determine output as

$$\boldsymbol{h}(t_1) = \boldsymbol{h}(t_0) + \int_{t_0}^{t_1} f(\boldsymbol{h}(t), t)\ dt.$$

An example: binary classification



input output

Training a neural ordinary differential equation

1. Set up an error function $\mathcal{L}(\boldsymbol{h}(t_1), y)$ that depends on the output of the neural ODE and ground-truth values.

2. Determine the parameters θ of f via gradient based optimisation on the error. To calculate gradients $\frac{\partial \mathcal{L}}{\partial \theta}$, we need to

   a) calculate the output:

   $$\boldsymbol{h}(t_1) = \boldsymbol{h}(t_0) + \int_{t_0}^{t_1} f(\boldsymbol{h}(t), t)\ dt,$$

   b) solve the adjoint ODE, backwards in time:

   $$\frac{d\boldsymbol{a}(t)}{dt} = -\boldsymbol{a}(t)^T \frac{\partial f(\boldsymbol{h}(t), t)}{\partial \boldsymbol{h}(t)},$$

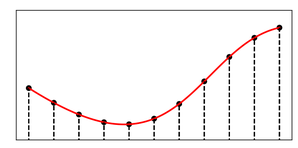
   c) determine the gradient:

   $$\frac{\partial \mathcal{L}}{\partial \theta} = -\int_{t_1}^{t_0} \boldsymbol{a}(t)^T \frac{\partial f(\boldsymbol{h}(t), t)}{\partial \theta}\ dt.$$

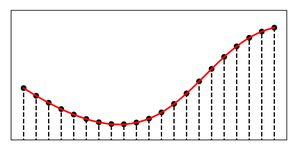
We save memory by computing gradients via integration.

Rising number of function evaluations (NFE)

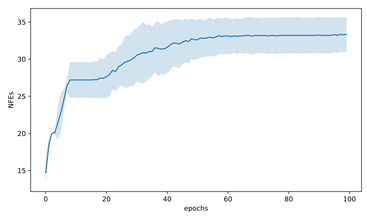
$$\boldsymbol{h}(t_1) = \boldsymbol{h}(t_0) + \int_{t_0}^{t_1} f(\boldsymbol{h}(t), t)\ dt$$



numerical integration discretises input space



higher accuracy requries higher NFE



We would like to reduce the NFE during training without sacrificing generalisation and robustness.

NFE represents the number of times points along a solution trajectory are passed through the network f that defines a neural ODE.

Introduction

Our work investigates the generalisation and robustness properties of neural ordinary differential equations (ODEs) [1] when their computational cost is reduced through the addition of Jacobian regularisation terms to the loss function during training. We introduce Jacobian condition number regularisation and show that

1. regularising the condition number of the Jacobian reduces the NFE without sacrificing test accuracy, and
2. Jacobian norm regularisation can increase the distance to the decision boundary for correctly classified data points, but does not improve robustness against input perturbations.

Regularisation methods

The Jacobian $\boldsymbol{J} \in \mathbb{R}^{d \times d}$ is defined as

$$\boldsymbol{J} = \nabla_{\boldsymbol{h}(t_0)} f(\boldsymbol{h}(t), t). \tag{1}$$

We experiment with regularising the Frobenius norm $\|\boldsymbol{J}\|_F$, the spectral norm $\|\boldsymbol{J}\|_2$, and the condition number $\kappa(\boldsymbol{J})$. These are defined as

$$\|\boldsymbol{J}\|_F = \sqrt{\sum_{i=1}^{d}\sum_{j=1}^{d} \left|\boldsymbol{J}_{i,j}\right|^2}, \tag{2}$$

$$\|\boldsymbol{J}\|_2 = \sigma_{\max}(\boldsymbol{J}), \tag{3}$$

$$\kappa(\boldsymbol{J}) = \frac{\sigma_{\max}(\boldsymbol{J})}{\sigma_{\min}(\boldsymbol{J})}, \tag{4}$$

where $\sigma_{\max}$ and $\sigma_{\min}$ refer to the largest and smallest singular values of a matrix.

Robustness and distance to decision boundary

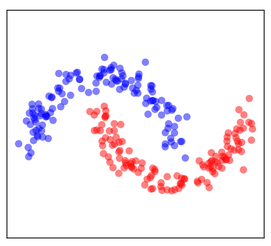


Figure 1: The intertwining moons dataset (red and blue indicate class labels).

Generalisation and sensitivity are investigated by means of performance on a hold-out test set, as well as input perturbations on the intertwining moons dataset (Figure 1). The input perturbations include varying levels of Gaussian noise.

Decision boundary distance. To determine the distance to a decision boundary, we generate points on d-dimensional spheres uniformly at random, with increasing radii. We perform a linear search over the spheres to determine the largest radius for which points are still labelled consistently.

NFE reduction results

Table 1 shows that all three regularisation methods successfully reduce NFE. However, Jacobian condition number regularisation achieves this reduction without a cost to test accuracy.

	Intertwining moons		
Regularisation	NFE	Test accuracy	Condition number
None	34.98 ± 2.98	0.9975 ± 0.0008	5.31 ± 3.51
Frobenius	14.00 ± 0.00	0.8862 ± 0.0003	27.3 ± 34.1
Spectral	19.81 ± 5.76	0.8846 ± 0.0022	45.9 ± 70.6
Condition number	27.12 ± 1.94	0.9973 ± 0.0007	6.10 ± 5.22

Table 1: Measures of NFE, test accuracy and Jacobian condition number, for the different regularisation strategies investigated. The Jacobian condition number in the last column is an average over the training data after the final epoch of training.

Robustness and distance to decision boundary results

Figure 2 shows that Jacobian condition number regularisation offers similar robustness when compared to the baseline. It also shows a slow decline in performance under random (Gaussian) perturbations. This could relate to the observation that Jacobian norm regularisation leads to larger classification margins, as shown in Figure 3.

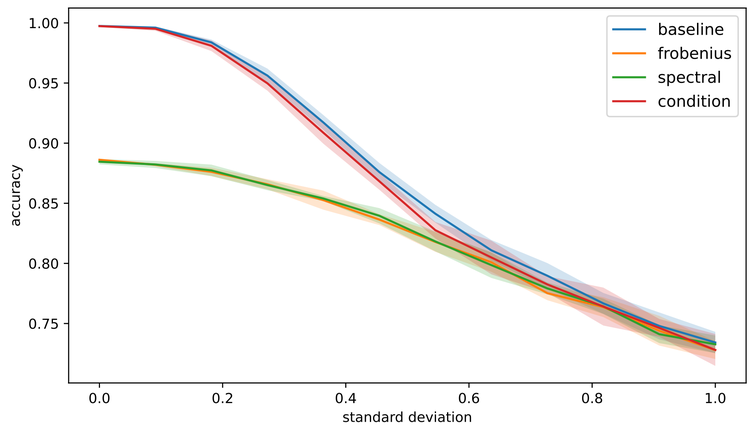


Figure 2: Accuracy as a function of Gaussian perturbations. Standard deviation of 0 corresponds to standard test set performance.

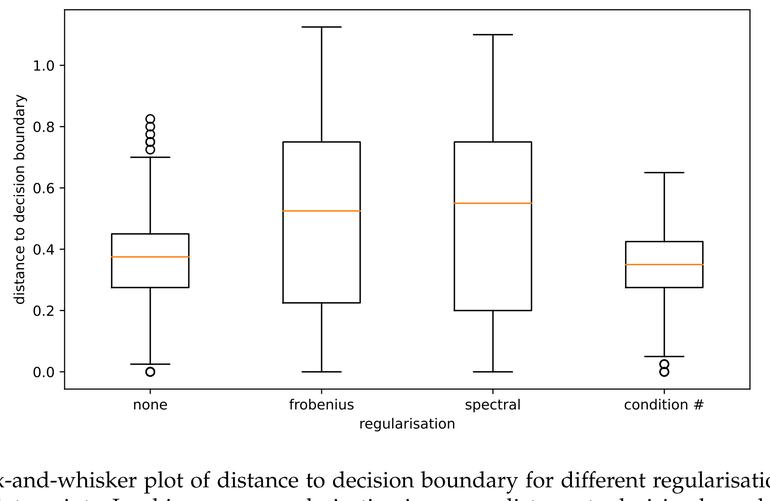


Figure 3: Box-and-whisker plot of distance to decision boundary for different regularisation strategies, over the training data points. Jacobian norm regularisation increases distance to decision boundary on average.

Conclusion

We considered strategies to reduce the computational cost of neural ODES and their effects on generalisation and robustness. The results indicate that Jacobian condition number regularisation can reduce NFE without a cost to test set accuracy. Future work will look into a more efficient computation for the condition number so that the regularisation scheme can scale to more relevant problems.

References

[1] Ricky TQ Chen, Yulia Rubanova, Jesse Bettencourt, and David K Duvenaud. Neural ordinary differential equations. *Advances in Neural Information Processing Systems*, 31:6572–6583, 2018.

Acknowledgement

This work is based on research supported by the National Research Foundation of South Africa (grant number 138341).