# Estimation of dynamical instability (local Lyapunov exponents) for non-intrusive adaptive modelling

Daniel Ayers, Jack Lau, Javier Amezcua, Alberto Carrassi, Varun Ojha

University of Reading · National Centre for Earth Observation · Tecnológico de Monterrey · Alma Mater Studiorum Università di Bologna

## Key questions and motivations

1. Can we make improvements by adapting the modelling setup according to the state of the system? (Figure 1).
2. Can we use ML to provide diagnostic information to drive adaptations?
3. Local Lyapunov exponents (LLEs) measure dynamical instability over a finite time interval, i.e. how quickly nearby trajectories separate or get closer. They are too expensive to compute numerically during a forecast run.
4. Thus, can we use ML to estimate the LLEs from the system state? We investigate this extensively in toy models.[1]

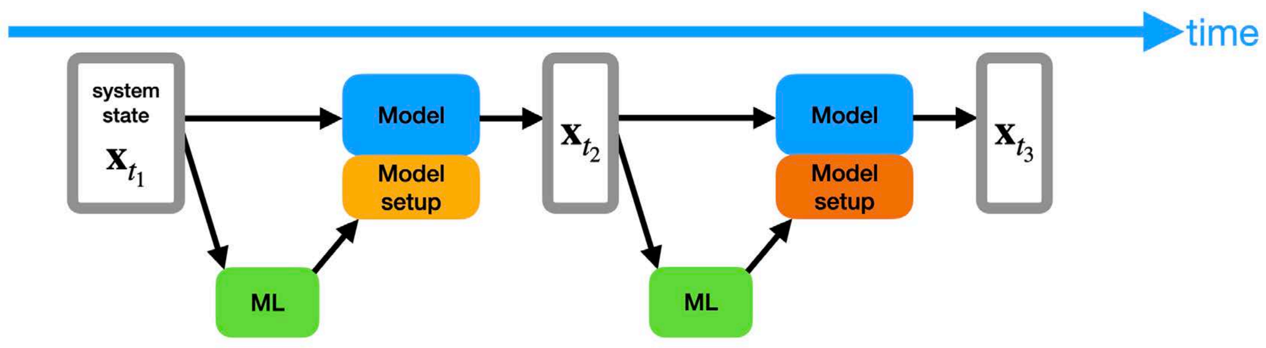


**Figure 1**: The non-intrusive paradigm for ML in NWP

## Target values

## Error locations

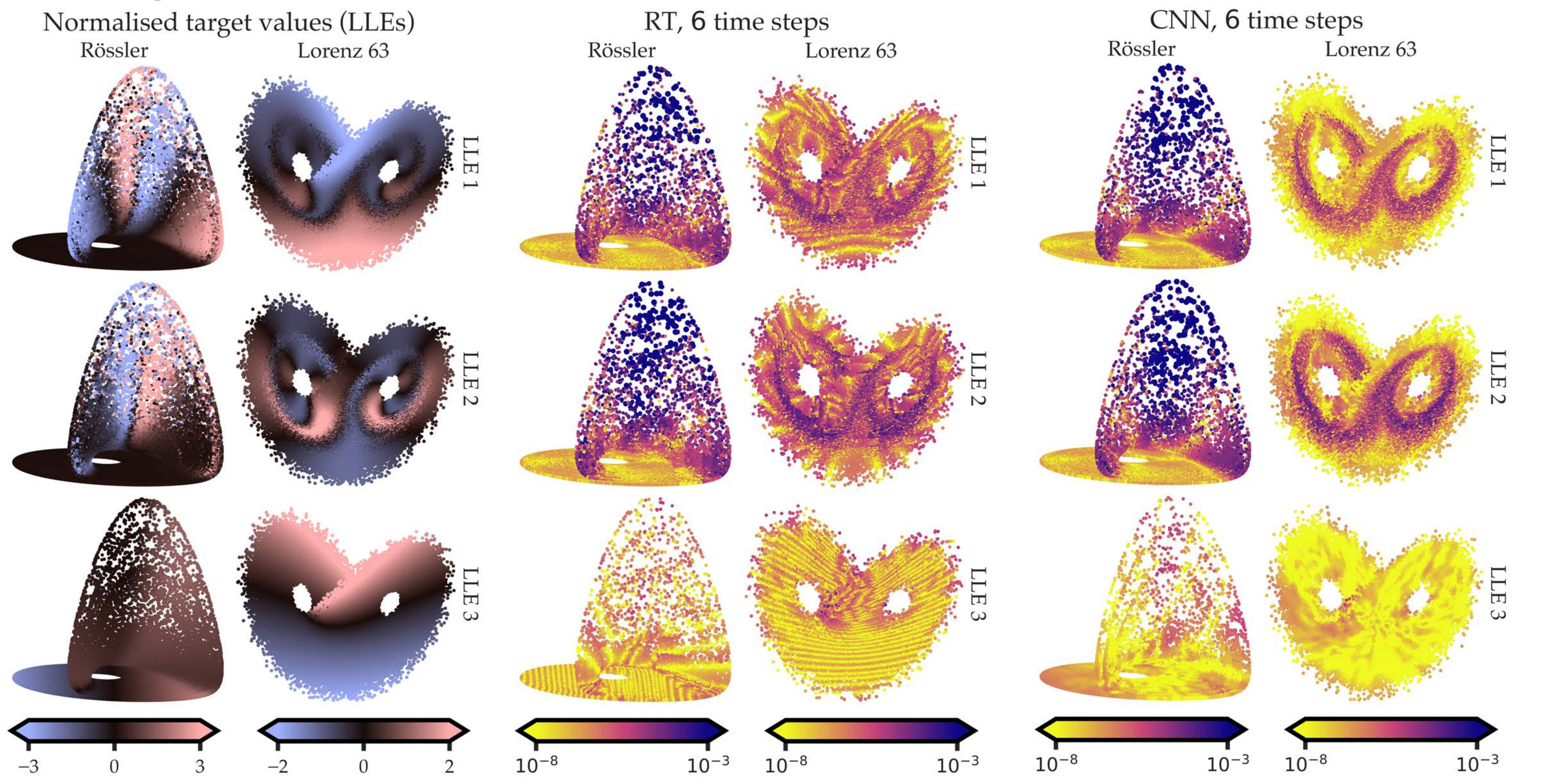


**Figure 5**: Target values (left) and negative contributions to $R^2$ score for RT (middle) and CNN (right). In error plots, darker points have larger error.

## Supervised ML problem

- Input: system states $(\mathbf{x}_{k-5}, \mathbf{x}_{k-4}, \ldots, \mathbf{x}_k)$ or $(\mathbf{x}_k)$
- Target: LLEs $(\lambda_1^k, \lambda_2^k, \ldots, \lambda_n^k)$, computed via the classic method using tangent linear model and orthogonalisation.[2,3]

## Experiment 1: Set up

- Two three-variable chaotic dynamical systems: Rössler and Lorenz 63
- Four ML methods (see Table 1)
- Separate bayesian hyperparameter optimisation of each system-algorithm-input
- 30 trials, in each trial data is shuffled uniquely prior to splitting into train and test
- Total data set of 100,000 examples: 60,000 train, 20,000 validation, 20,000 test.

| | ML algorithm | Architecture |
|---|---|---|
| RT | Regression Tree | 1 tree per target LLE |
| MLP | Multilayer perceptron | 1+ dense hidden layers, dense output layer |
| CNN | Convolutional neural network | 1D-convolution layer, max pool, flatten, 1+ dense layers |
| LSTM | Long short-term memory network | 1+ LSTM layers, dense output layer |

**Table 1**: ML algorithms used in Experiment 1

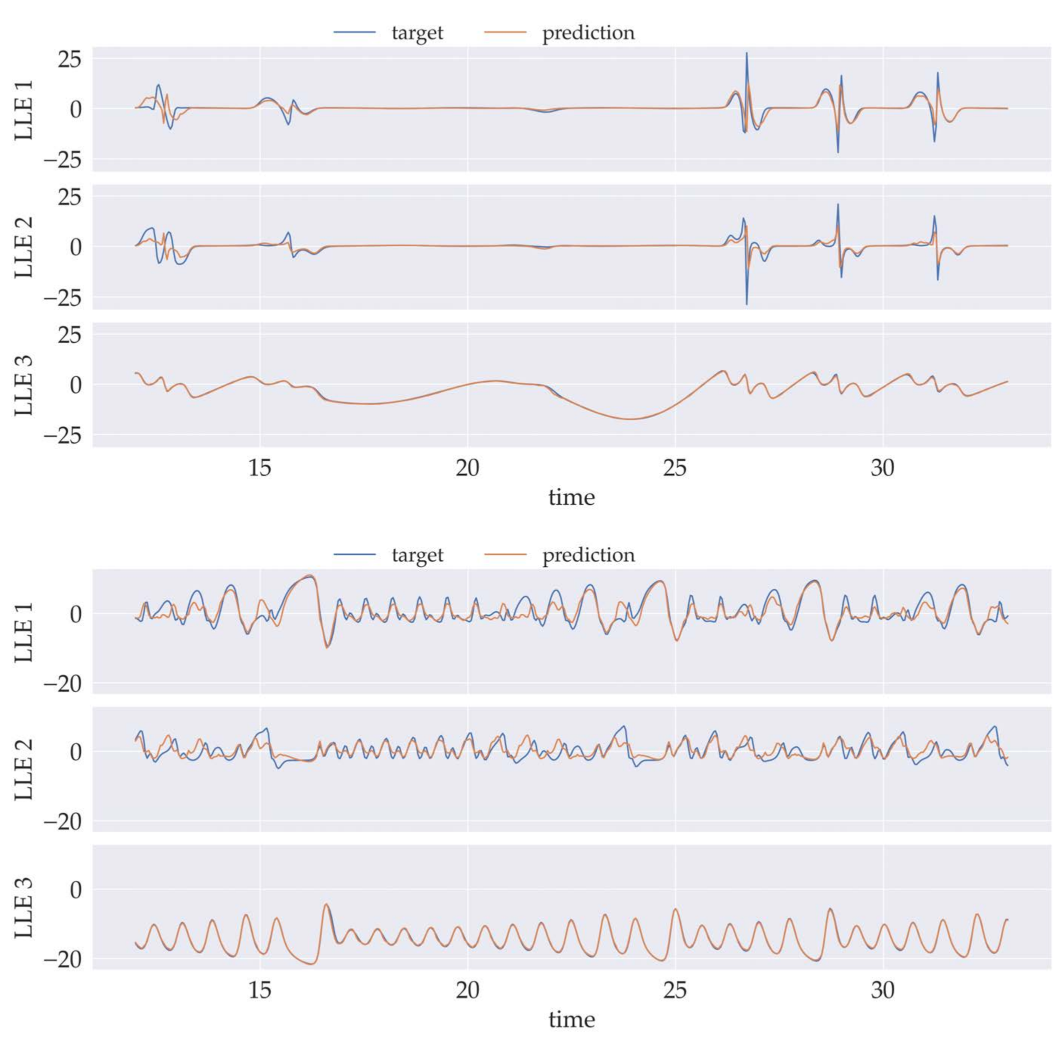


**Figure 3**: Target and prediction time series for the Rössler system (top) and Lorenz 63 system (bottom), using best performing algorithm.

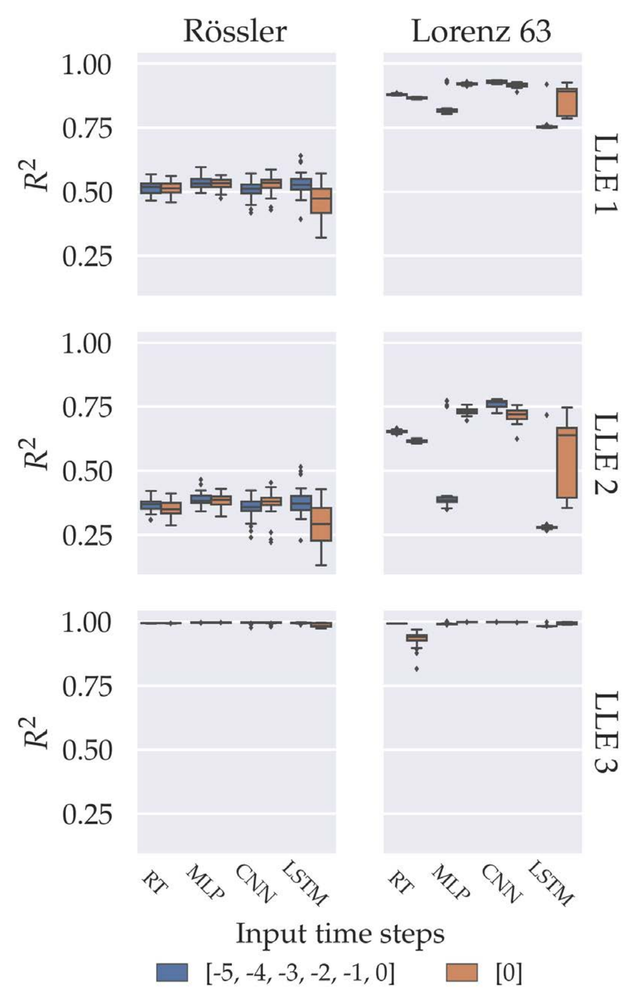


**Figure 2**: $R^2$ scores from trials in Rössler and L63 systems show the average accuracy on test data sets.

$$R^2 = 1 - \frac{\sum_{j=1}^{d} (y_j - \hat{y}_j)^2}{\sum_{j=1}^{d} (y_j - \bar{y}_j)^2}$$

**Equation 1**: $R^2$ score measures average accuracy of predictions $\hat{y}_j$ of targets $y_j$, with target mean $\bar{y}_j$.

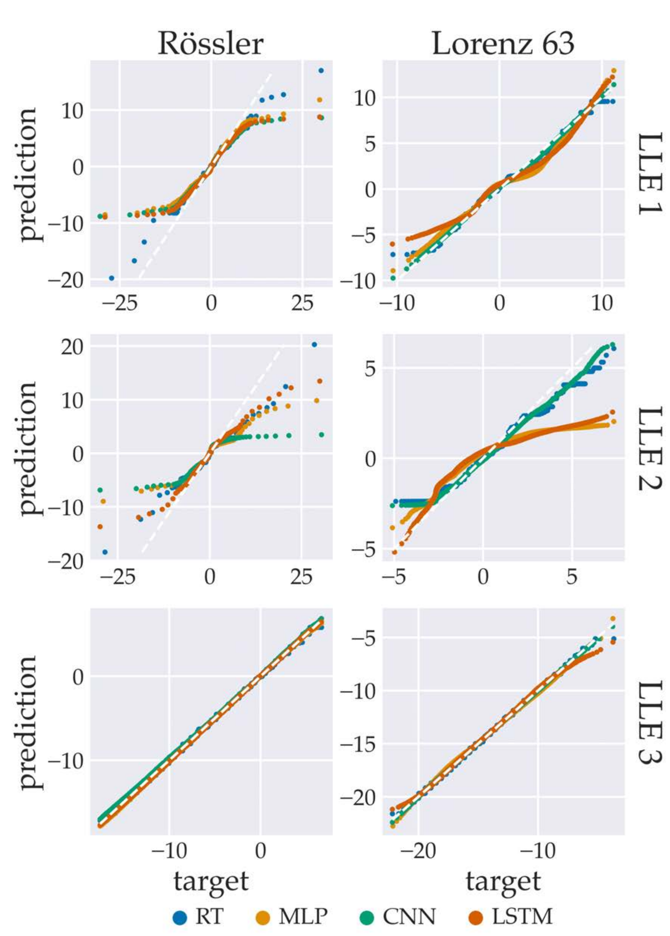


**Figure 4**: QQ plots show proximity of distributions of targets and predictions

## Experiment 1: key results

- LLE 3 well estimated, whereas LLE 1 and 2 are more challenging - especially LLE 2.
- Lower $R^2$ scores of Rössler system LLE 1 & 2 predictions due to poor reproduction of extreme values (e.g. see Figure 4) and lagged or out of phase predictions in the fluctuations (Figure 3).
- 6 time steps in input achieves marginally better results than 1 time step.
- Largest prediction errors occur in regions of attractor where LLE values are locally heterogeneous, i.e. highly mixed (see Figure 5).

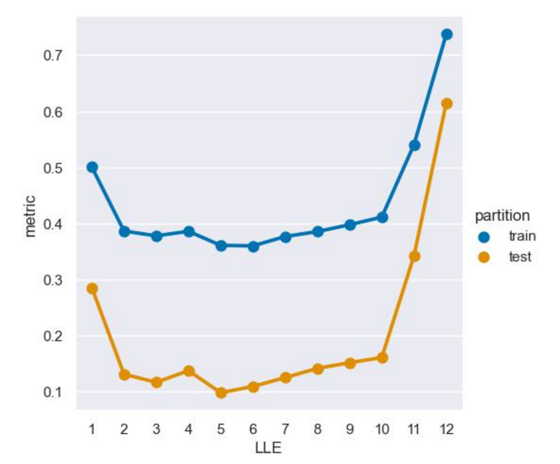


**Figure 6**: $R^2$ scores of predictions of Lorenz 96 model with $N = 12$, using gradient-boosted regression trees.

## Experiment 2: Lorenz 96, N=12.

- Overall, $R^2$ scores are lower (Figure 6). Best $R^2$ scores for LLEs 1 and 12 are 0.29 and 0.61, respectively.
- Similar patterns with most negative, neutral and most positive LLEs: most negative LLE is best estimated.
- Several ML algorithms tested: U-Net, ResNet-style CNN, gradient-boosted ensemble of regression trees.
- Found that converting to a classification problem (e.g. estimating number of positive LLEs) did not make the ML problem easier.
- Challenge is illustrated using dimension reduction visualisation via t-SNE[4], see Figure 7. Highly mixed regions pose a greater challenge.

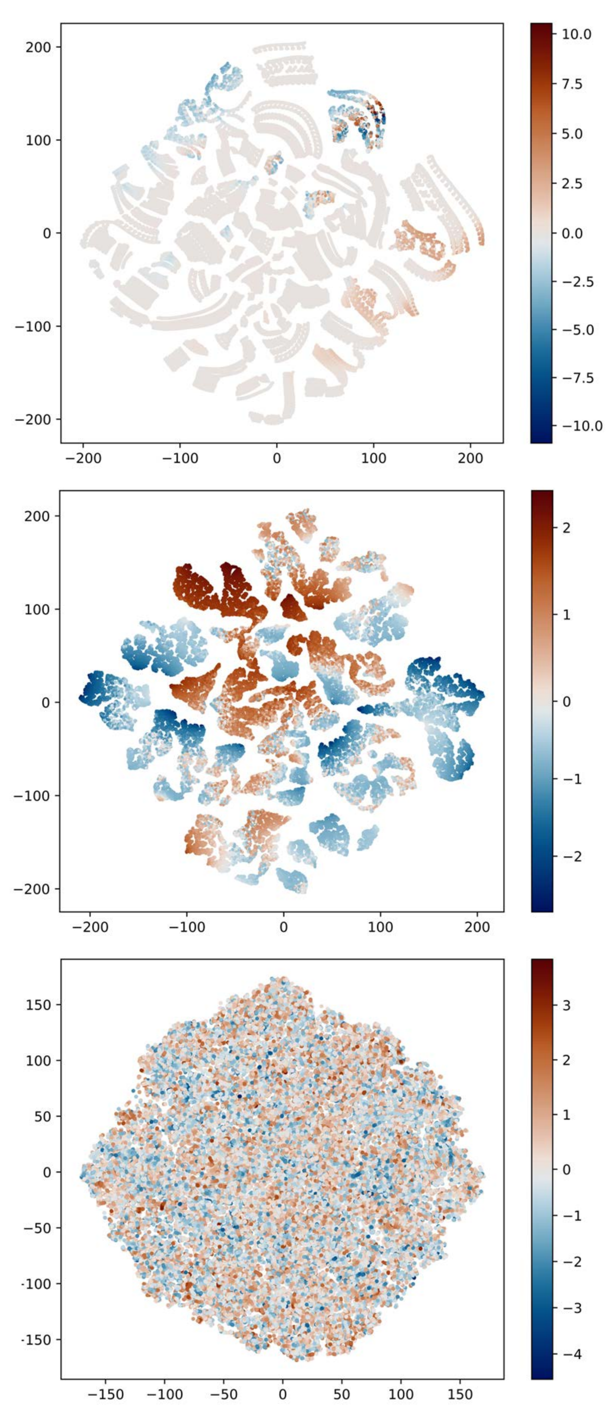

**Figure 7**: t-SNE-reduced inputs coloured by normalised LLE 1 target values for Rössler (top), Lorenz 63 (middle) and Lorenz 96 (bottom).

## Future work

- Ongoing work is investigating metrics to quantify the difficulty of the ML problem for a very local area of the system attractor.
- Aim to make ML predictions with uncertainty quantification, enabling action to be taken on the basis of high-confidence predictions.

### References

1. Ayers, D. et al. (2022). "Supervised machine learning to estimate instabilities in chaotic systems: estimation of local Lyapunov exponents". doi: 10.48550/ARXIV.2202.04944
2. Benettin, G. et al. (Mar. 1980a). "Lyapunov Characteristic Exponents for smooth dynamical systems and for hamiltonian systems; a method for computing all of them. Part 1: Theory". *Meccanica* 15, pp. 9–20. doi: 10.1007/BF02128236.
3. Benettin, G. et al. (Mar. 1980b). "Lyapunov Characteristic Exponents for smooth dynamical systems and for hamiltonian systems; a method for computing all of them. Part 2: Numerical application". *Meccanica* 15, pp. 21–30. doi: 10.1007/BF02128237.
4. van der Maaten, L. and Hinton, G. (2008). "Visualising Data using t-SNE". *Journal of Machine Learning Research* 9, pp. 2579-2605.

**See this poster online:**

**Read more here:**

arXiv:2202.04944