TIME-SERIES FORECASTING –

Using Echo State Networks and Takens' Theorem

Laís de Souza Alves
Physics' PhD Student

March 2, 2020

Universidade de Brasília – UnB

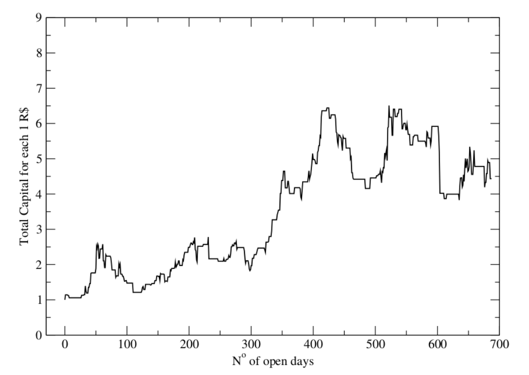
Summary

- Introduction
- Time-Series and Takens' Theorem
  - Lorenz System's Equations
  - Representation of a State Space
  - Embedded State Space
- Echo State Networks
  - Short-Term Predictions and Long-Term "Climate" Reproduction
  - Feed-Forward and Recurrent Neural Networks
  - Echo State Network Workflow
- Conclusion

Summary

- Introduction
- Time-Series and Takens' Theorem
  - Lorenz System's Equations
  - Representation of a State Space
  - Embedded State Space
- Echo State Networks
  - Short-Term Predictions and Long-Term "Climate" Reproduction
  - Feed-Forward and Recurrent Neural Networks
  - Echo State Network Workflow
- Conclusion

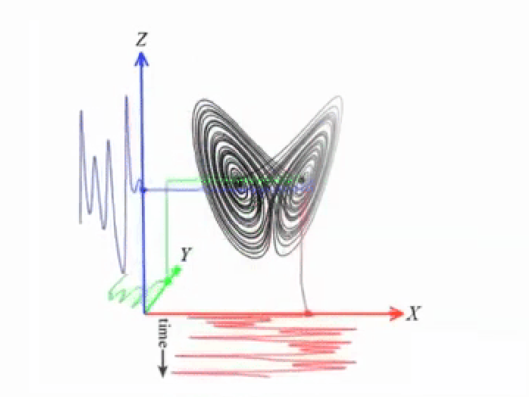
- Real time-series forecasting using Artificial Neural Networks:
  - We don't need to know the underlying equations of the system's dynamics, only past data is needed.



Source: [illegible]

Introduction

- Takens' Theorem and Attractor Reconstruction
  - It is possible to have "access" to other dimensions of your system;
  - Through only one dimensional measurement.



Summary

- Introduction
- **Time-Series and Takens' Theorem**
  - Lorenz System's Equations
  - Representation of a State Space
  - Embedded State Space
- Echo State Networks
  - Short-Term Predictions and Long-Term "Climate" Reproduction
  - Feed-Forward and Recurrent Neural Networks
  - Echo State Network Workflow
- Conclusion

Lorenz's System Equations:

Parameters:

$$= \frac{8}{3}$$

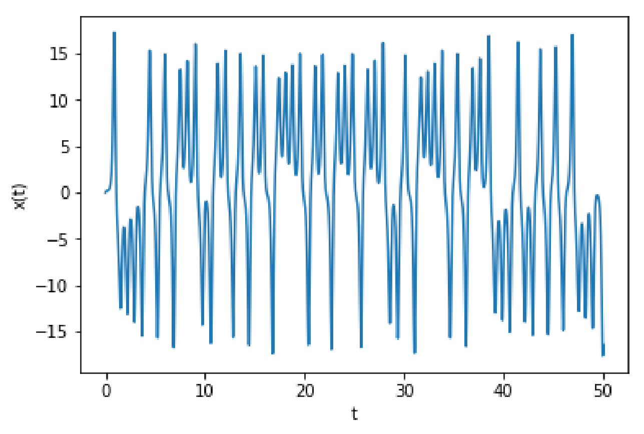
$$= 10$$

$$= 28$$

Time-Series and Takens' Theorem

- State Space Representation of Lorenz's Equations:

- Lorenz's x(t) "measurement":



- Takens' Embedding Theorem:

Let M be a compact manifold of dimension $\check{a}$, a smooth vector field and X a smooth function on M. It is a generic property that

$$_{(\phi t)}(g) : M \; ! \; \mathbb{R}^{2\check{a}+1}$$

is a embedding, where is the flow on M and

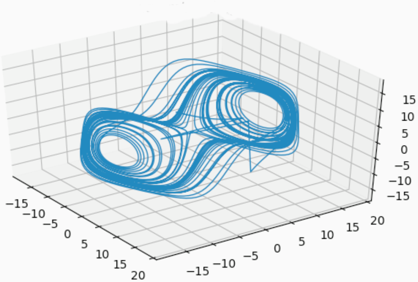
$$_{(\phi t)}(g) = \Big\langle t(g); t(\;(g)); t\;^{2}(g)\;;\;\;; t(\;^{2\check{a}}(g)) \Big\rangle :$$

Source: [illegible] (1981)

Time-Series and Takens' Theorem

- <u>Takens' Embedding Theorem</u> (more digestible form):

  For **the right** **(** > 0**)** and **enough dimensions** 2ǎ + 1 for the Takens' Vectors, the embedded dynamics are diffeomorphic to the original state space dynamics:

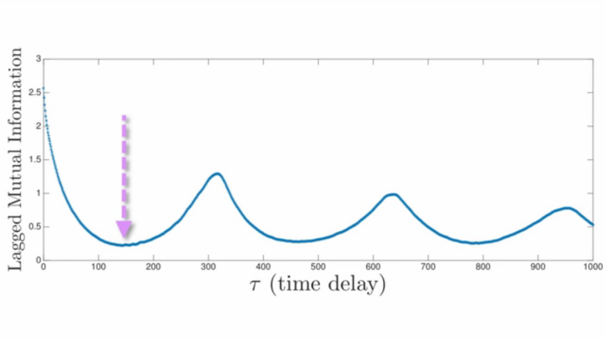


Fonte: URL=<https://github.com/carlosgmartin/takens>

Time-Series and Takens' Theorem

One of the methods used for choosing the appropriate   :

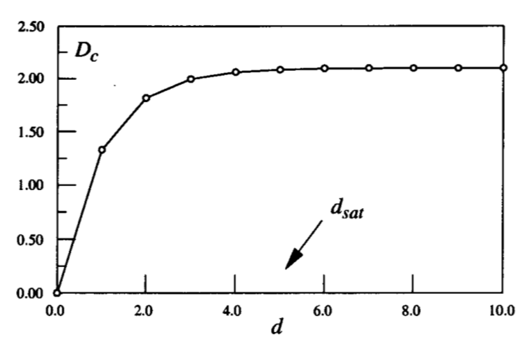
- Mutual information – You can choose the first or the second minima, or even the first maxima. (Note: this choice is heuristic)



Source:

And for choosing the appropriate dimension:

- False Nearest Neighbors (the chosen threshold is also heuristic); or
- Correlation (Grassberger-Procaccia) dimension:



Time-Series and Takens' Theorem

Some **caveats** to **be aware** of:

- Noise in the data;
- Limited length of data;
- Variation on the temporal sampling;
- And non-stationarity (dynamics changes) on the data;

Summary

- Introduction
- Time-Series and Takens' Theorem
  - Lorenz System's Equations
  - Representation of a State Space
  - Embedded State Space
- **Echo State Networks**
  - Short-Term Predictions and Long-Term "Climate" Reproduction
  - Feed-Forward and Recurrent Neural Networks
  - Echo State Network Workflow
- Conclusion

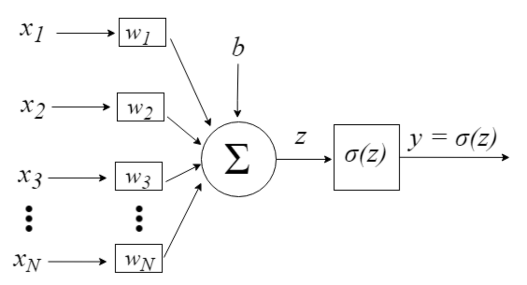
Echo State Networks

- Forecast of a time-series:
  - It is possible to use Recurrent Neural Networks (RNNs) to make short-term forecasts and long-term reproduction of the system's "climate":

Feed-Forward Neural Network

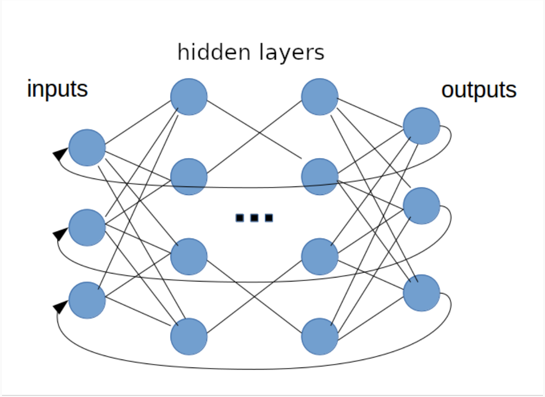
- Prediction formula:

$$\hat{y} = \sigma\left(\sum_{i} w_i x_i + b\right)$$



Source: URL=<https://medium.com/ensina-ai/redes-neurais-perceptron-multicamadas-e-o-algoritmo-backpropagation-eaf89778f5b8>

Representation of a Recurrent Neural Network (RNN):



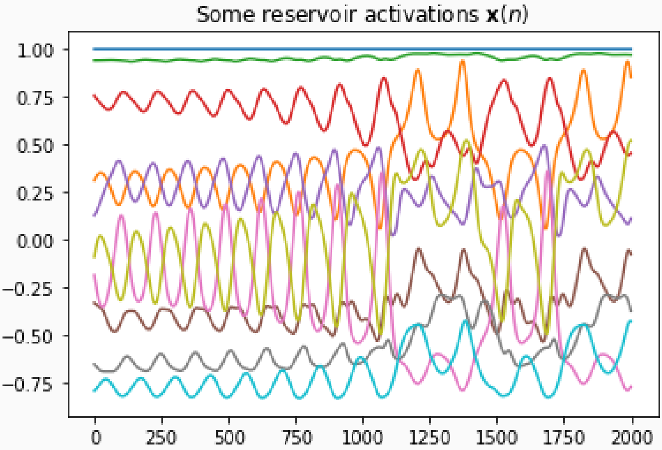
Šϖ$_w^w$ȷ˜ ϖ˝ For a input on the form ϡÝ̄ İ̄ Ŝ▯

- Choose the Reservoir size and the matrix sparsity (of connections);
- Generate the input weights W$_{\tilde{0}\hbar}$, from a binomial random distribution;
- Generate the Reservoir weights W$_{w}^{w}$ from a uniform distribution

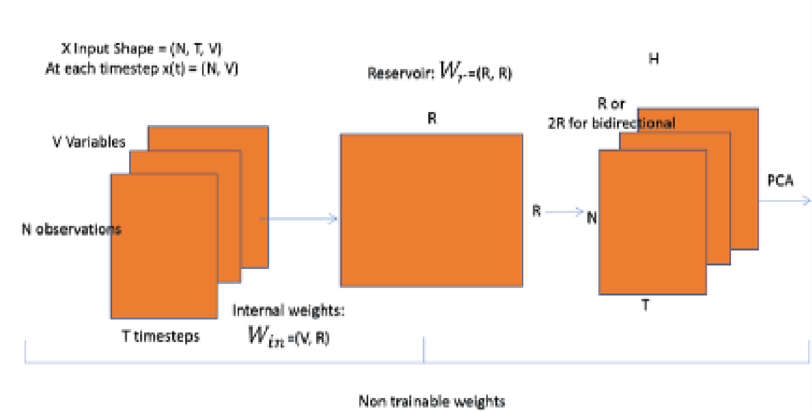
- Calculate the hidden state $\mathbf{h}(t)$:

$$\mathbf{h}(t) = \sigma(\mathbf{W}_{in}\mathbf{x}(t) + \mathbf{W}_{hh}\mathbf{h}(t-1))$$

- Create an new input representation: A matrix with all the collected hidden states, or with roots and slopes, or a mean value of $\mathbf{h}(t-1)$

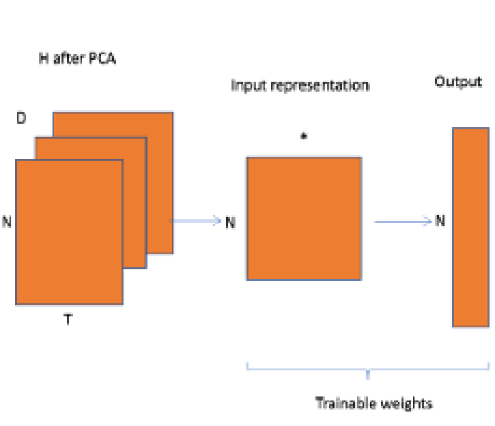

Some reservoir activations $\mathbf{x}(n)$
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
0
250
500
750
1000
1250
1500
1750
2000

Echo State Network



Source: URL=<https://towardsdatascience.com/gentle-introduction-to-echo-state-networks-af99e5373c68>

- Connect this new input representation to the output layer (can be a simple RNN with one or two layers)



Echo State Network

- Advantages of ESNs:
  - The training phase is faster and doesn't suffer from backpropagation instabilities of Vanilla RNNs;
  - They are well suited for chaotic systems;
  - They are capable of "learning" short-term and long-term correlations;
- Disadvantages of ESNs:
  - Selection of tuning parameters $\rightarrow$ trial and error;
  - Limited size of Reservoir: its size is dependent of computer's memory capacity;

Conclusions

- So what does Takens' Embedding have to do with Echo State Networks?

Bibliography

1 T.F.M. Rocha and P.M.M. Rocha. **"Inefficiency of the Brazilian Stock Market: the IBOVESPA Future Contracts."** arXiv:1904.09214 (2019)

2 Takens, F. **"Detecting strange attractors in turbulence."** (1981)

3 Fraser, A. M., and Swinney, H. L. **"Independent coordinates for strange attractors from mutual information."** Physical review A, 33(2):1134. (1986)

4 Kennel, M. B., Brown, R. and Abarbanel, H. D. I. **"Determining embedding dimension for phase-space reconstruction using a geometrical construction."** Physical review A 45(6):3403 (1992)

5 Hilborn, Robert C. **"Chaos and nonlinear dynamics: an introduction for scientists and engineers."** (2000)

Feel free to reach me by: laisouzalves@gmail.com

Thank you!

Any questions?