



# Do Caos às Criptomoedas

*Uma análise de criptoativos baseada na física de sistemas caóticos e redes neurais*

Gabriel Leonardo Beber  
Gustavo Sieverding Bier  
João Miguel Ramisch Pergher  
Kevin Gabriel Ramisch Pergher

# Caos, Sistemas não lineares, Complexidade & Criptoativos

Complexidade: O conhecimento das componentes básicas de um sistema não explica o sistema em si



Available online at [www.sciencedirect.com](http://www.sciencedirect.com)

ScienceDirect

Procedia Computer Science 119 (2017) 368–375

Procedia

Computer Science

[www.elsevier.com/locate/procedia](http://www.elsevier.com/locate/procedia)

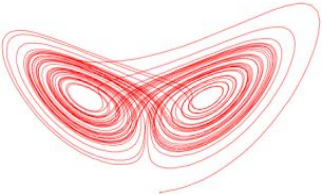
6th International Young Scientists Conference in HPC and Simulation, YSC 2017, 1-3 November 2017, Kotka, Finland

Chaos Theory in Finance

Igor Kloutchnikov\*, Maria Sigova\*, Nikita Beizerov\*\*

AND HE International Banking Institute, Nevsky str., 40, 191023, Saint-Petersburg, Russia

Caos: alta sensibilidade às condições iniciais



Não linearidade: fenômenos com feedback loop

## GameStop

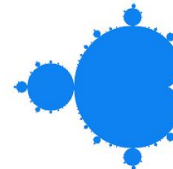


Fenômenos comportamentais...  
Shiller, Robert

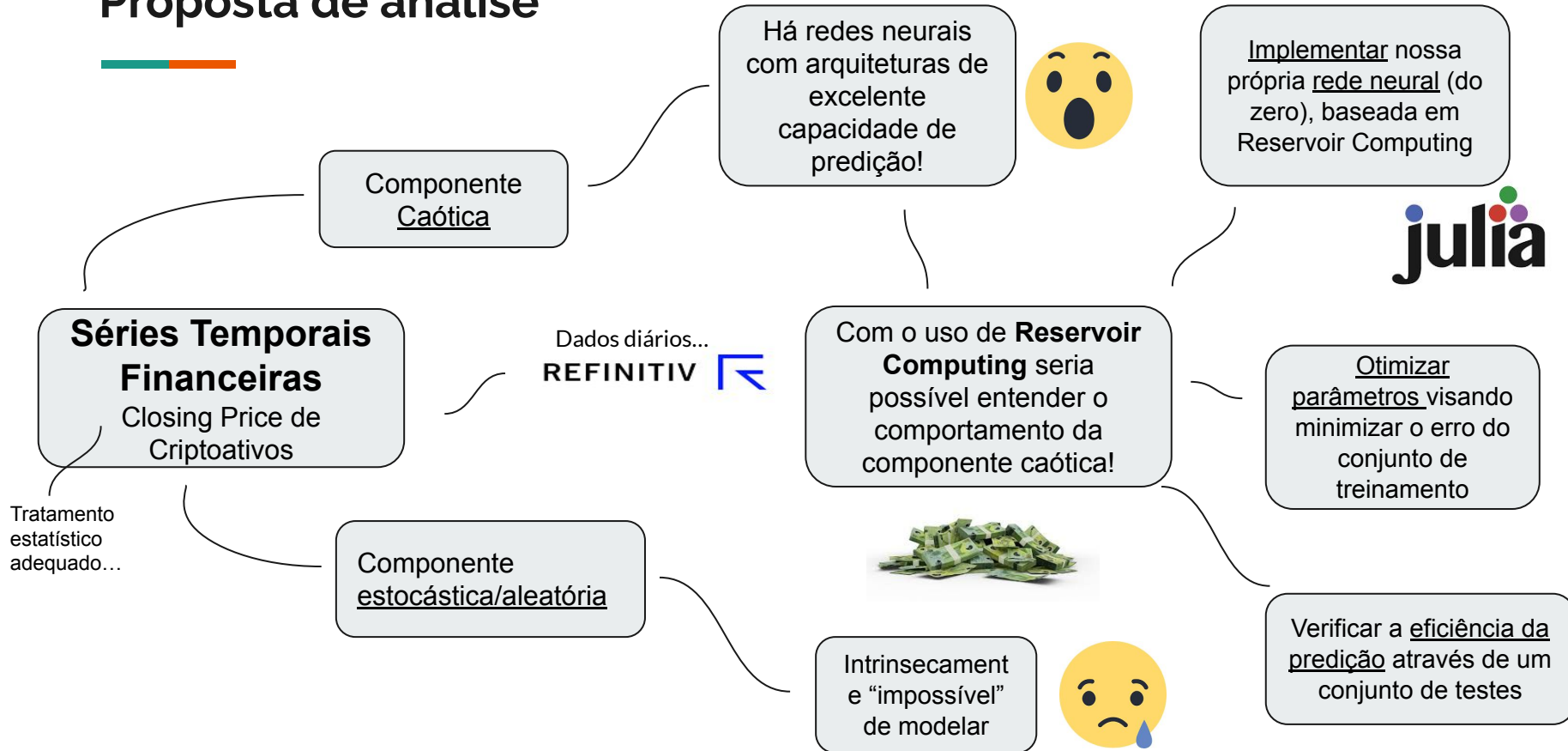
“The so called Econophysicists...”

Mandelbrot, Benoit

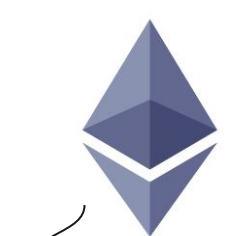
The (Mis)behaviour of the Markets



# Proposta de análise



# Bases de dados & Pré-processamento



Ethereum (ETH)



Bitcoin (BTC)



Dogecoin (DOGE)

Separamos os dados em dois conjuntos:

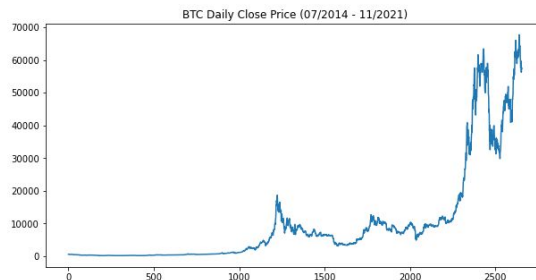
**Treino** (90%) para otimização dos parâmetros

**Teste** (10%) para verificação da eficiência da rede

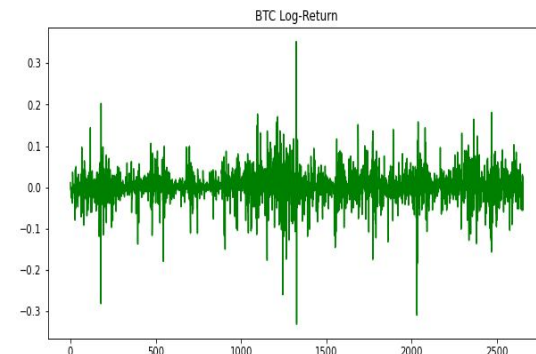
	BTC (Treino)	DOGE (Treino)	ETH (Treino)
Início	17/07/2014	01/02/2014	01/11/2018
Fim	24/11/2021	05/11/2021	27/04/2022
# Dados	2655	2835	1272

	BTC (Teste)	DOGE (Teste)	ETH (Teste)
Início	25/11/2021	06/11/2021	28/04/2022
Fim	16/09/2021	16/09/2021	16/09/2021
# Dados	296	315	142

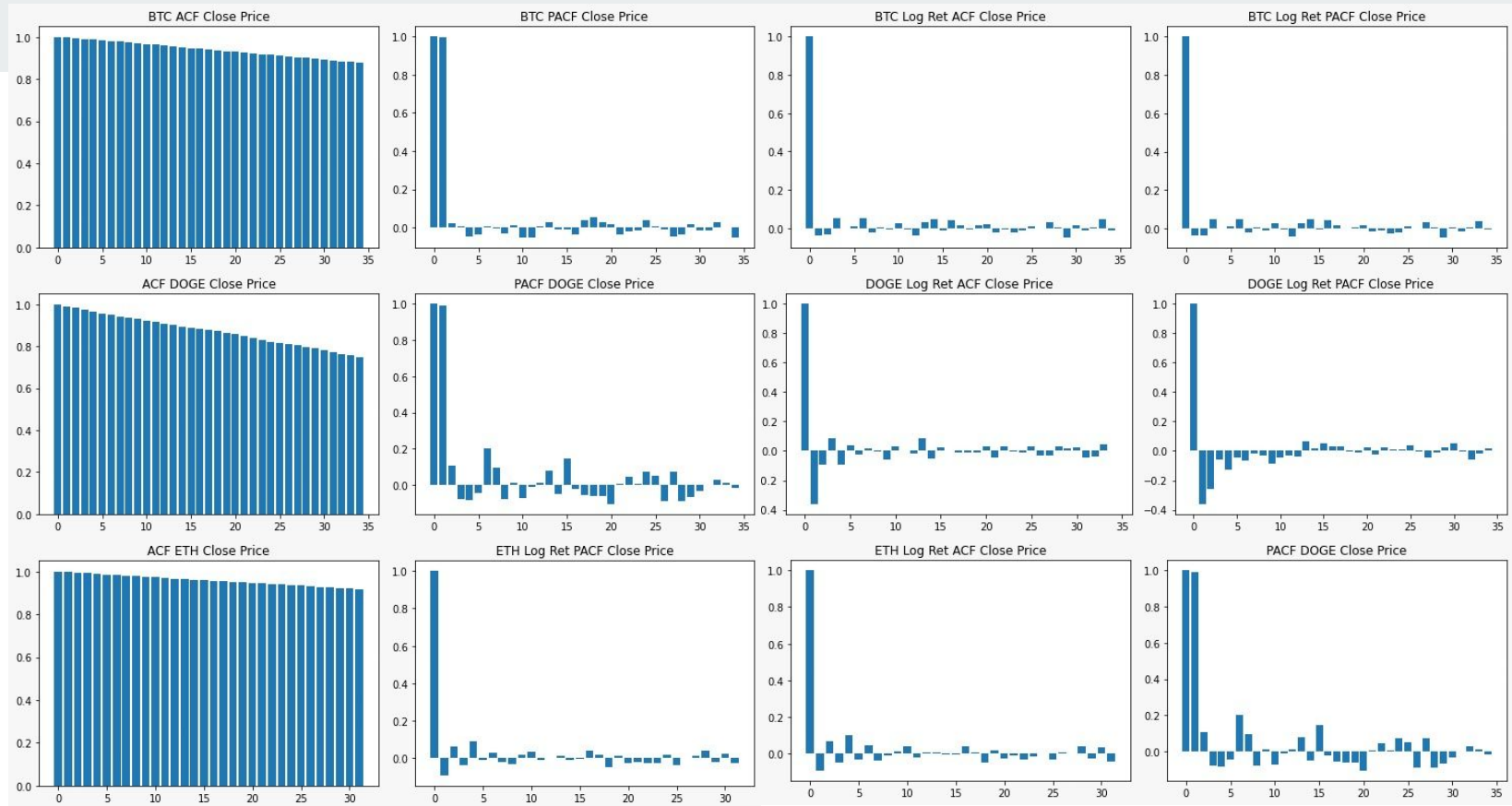
Analizamos a estacionariedade da série e utilizamos os log-retornos, conferimos os resultados via **ACF**, **PACF** e utilizamos o **teste de Dickey-Fuller**



Close Price



Log-Return



BTC\_LOGRET DOG\_LOGRET ETH\_LOGRET BTC\_CLOSE DOG\_CLOSE ETH\_CLOSE

dickey-fuller values

adf	-1.153424e+01	-1.305959e+01	-1.648725e+01	0.068538	-2.381507	-0.699402
p-value	3.767004e-21	2.060603e-24	2.209118e-29	0.963846	0.147024	0.846904

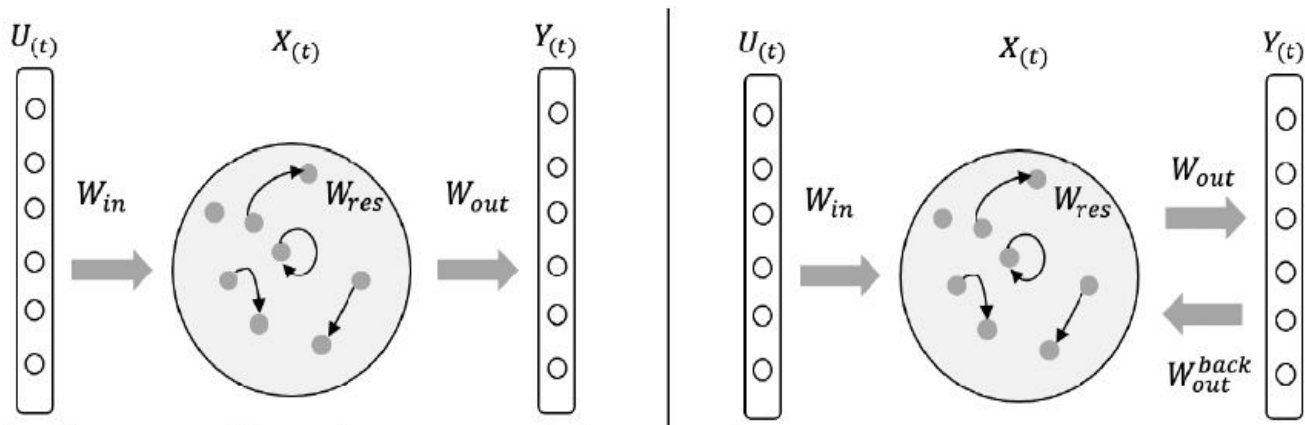
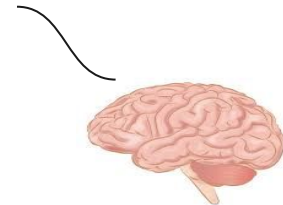


Jaeger, Herbert (2002)

Inspiração biológica! Uma melhor abordagem para sistemas não lineares?

# Echo States Neural Networks (ESN)

## A powerful Reservoir Computing framework



Crie matrizes de pesos fixos. Pesos input-neurônios, inter-neurônios, neurônios-output, ...

Garanta que estas matrizes sejam preenchidas por uma distribuição estatística adequada (uniforme centrada em zero). e que obedeçam aos critérios.

### Hiperparâmetros

Definem as características da dinâmica da rede

(I) Raio Espectral + (II) Peso dos dados de entrada (Scale) + (III) Esparsidade + (IV) Vazamento

# How to Echo State...

Os hiperparâmetros definem as características da rede (do ponto de vista estatístico):

**Raio Espectral:** Raio espectral da matriz de pesos neurônio-neurônio;

**Esparsidade:** Razão, define arbitrariamente quantas conexões entre neurônios são deliberadamente zeradas;

**Peso de entrada (Scale):** Como nas arquiteturas tradicionais, por quanto a entrada (input) será ponderada;

**Vazamento (Leaking):** Razão, define a capacidade dos reservatórios de reter estados anteriores parcialmente.

Cada neurônio (unidade de processamento) tem um estado associado, que é atualizado a cada “rodada”:

$$\mathbf{x}(n+1) = (1 - \alpha_{leaking})\mathbf{x}(n) + \alpha_{leaking}f(\mathbf{W}_{in}[1; \mathbf{u}(n+1)] + \mathbf{W}_{res}\mathbf{x}(n))$$

$$\mathbf{y}(n+1) = \mathbf{W}_{out}[1; \mathbf{u}(n+1); \mathbf{x}(n+1)]$$

Apenas uma matriz não é fixada: a matriz de pesos de saída. Ela é obtida pela resolução de uma equação matricial, de modo a minimizar o erro e obter convergência.

Há várias maneiras de resolver esta equação, apenas não desejamos o overfitting. Optamos pela Regularização de Tikhonov (Regressão de Ridge).

$$\min_{W_{out}} \|\mathbf{W}_{out}\mathbf{X} - \mathbf{Y}\|_2^2$$

$$\mathbf{W}_{out} = \mathbf{Y}\mathbf{X}^T(\mathbf{X}\mathbf{X}^T + \beta\mathbf{I})^{-1}$$

Regularização de  
Tikhonov



Neural Networks  
Volume 108, December 2018, Pages 495–508



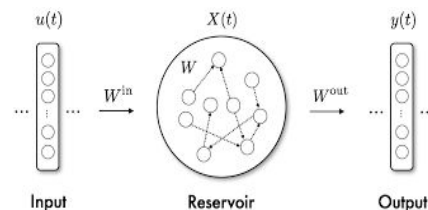
Echo state networks are universal

Lyudmila Grigoryeva<sup>a,\*,1</sup>, Juan-Pablo Ortega<sup>b,c,d,2</sup>

Show more

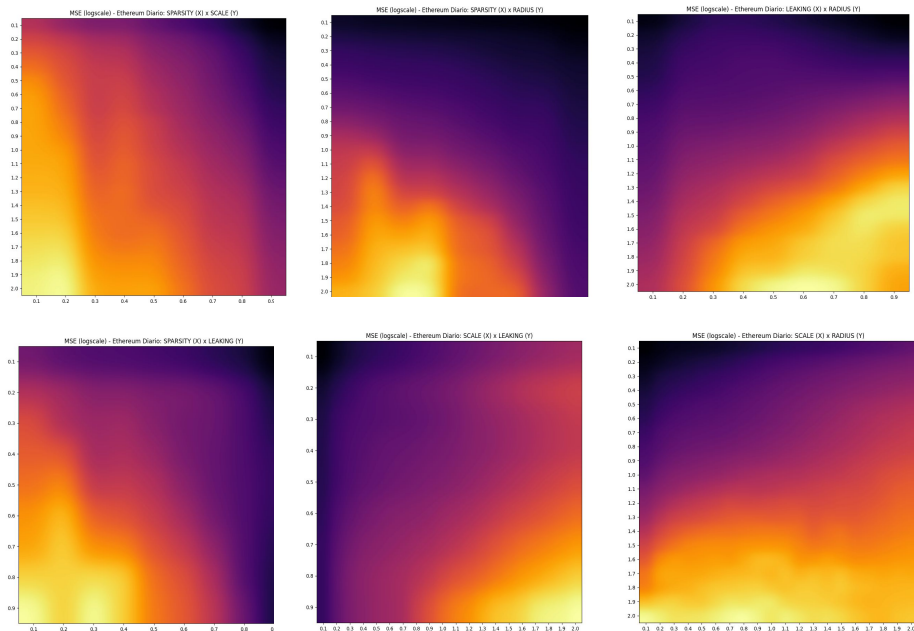
+ Add to Mendeley Share Cite

<https://doi.org/10.1016/j.neunet.2018.08.025>



# Treinando a rede (Tunning)

Otimização dos parâmetros, quanto mais claro, menor o erro quadrática médio  
(MSE, escala logarítmica)  
Medida de tendência central: Mediana



## Espaço de busca:

**Raio Espectral:** entre 0.1 e 3.0 (+0.1)

**Pesos de entrada:** entre 0.1 e 3.0 (+0.1)

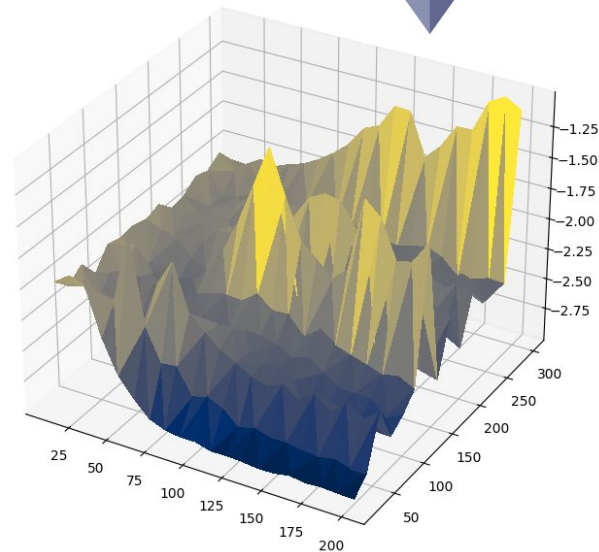
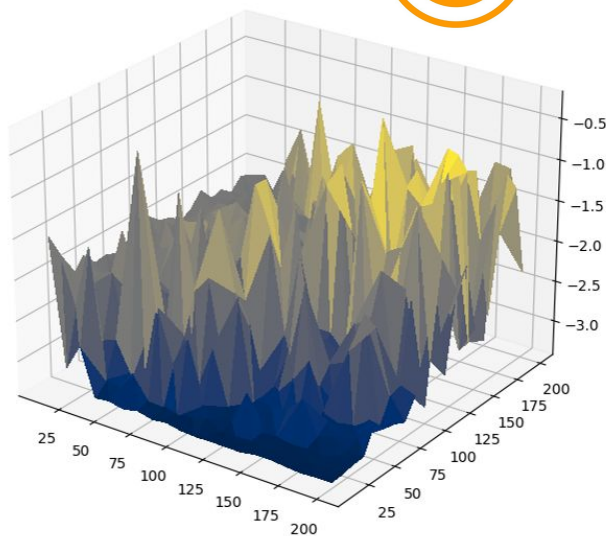
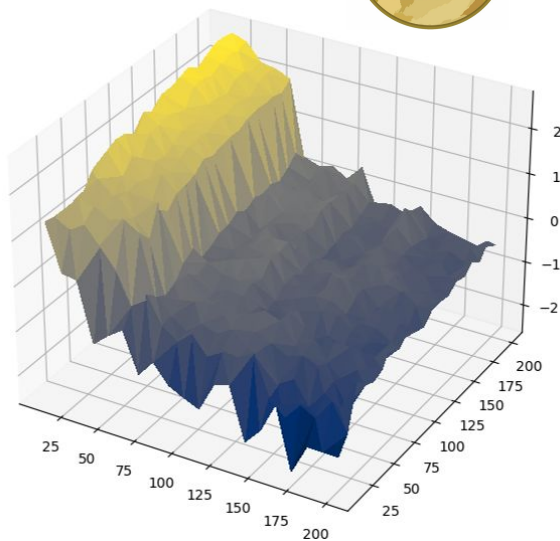
**Vazamento:** entre 0.1 e 0.9 (+0.1)

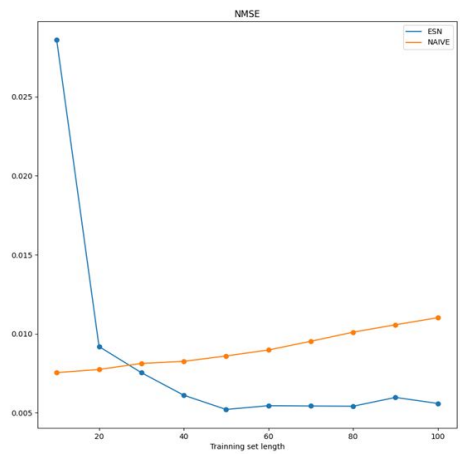
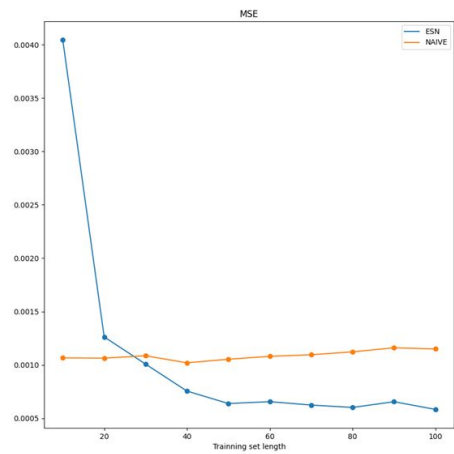
**Esparsidade:** entre 0.1 e 0.9 (+0.1)

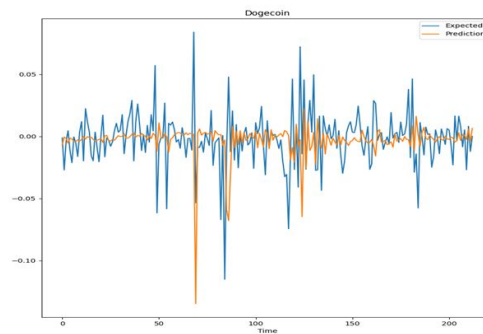
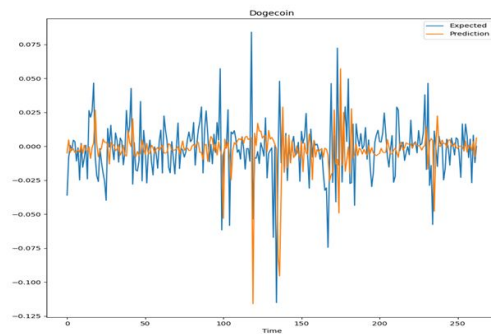
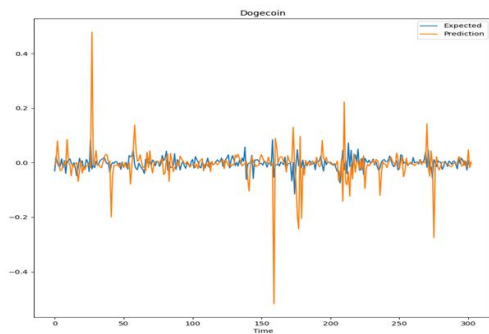
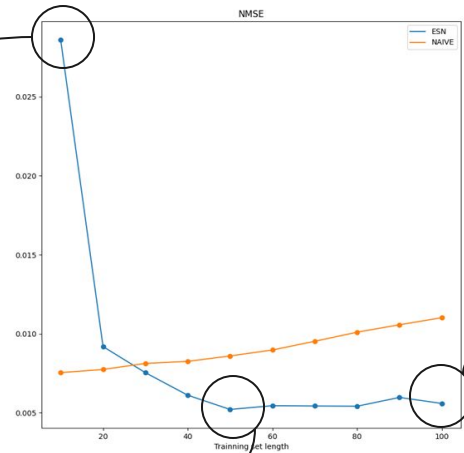
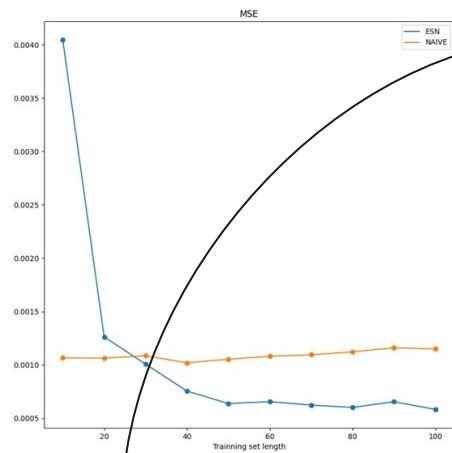
	Bitcoin	Ethereum	Dogecoin
Sparsity	0,1	0,3	0,9
Scale	2,2	2,0	3,0
Radius	2,5	2,0	0,2
Leaking	0,4	0,6	0,9

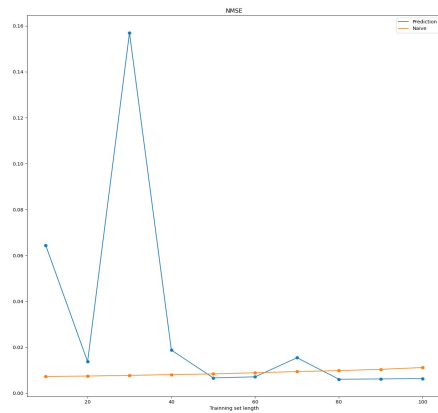
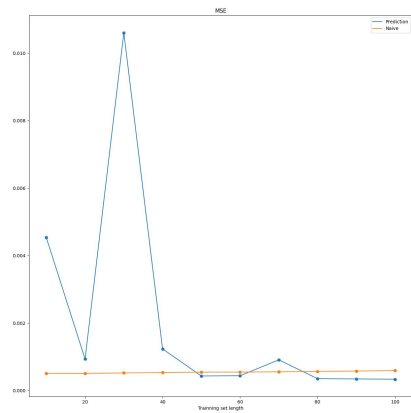


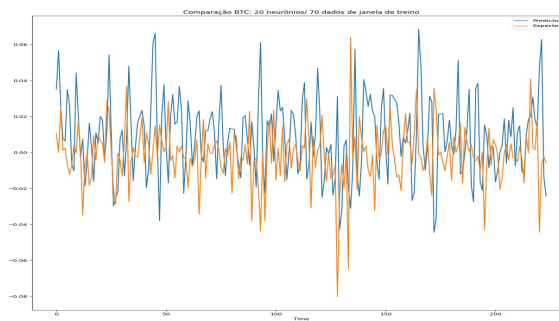
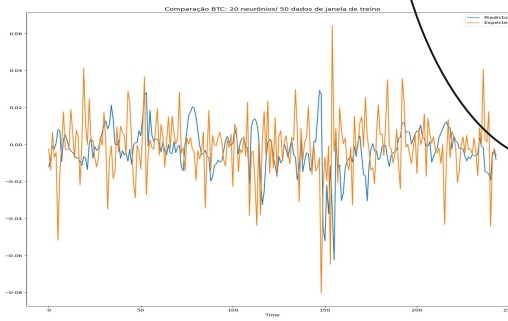
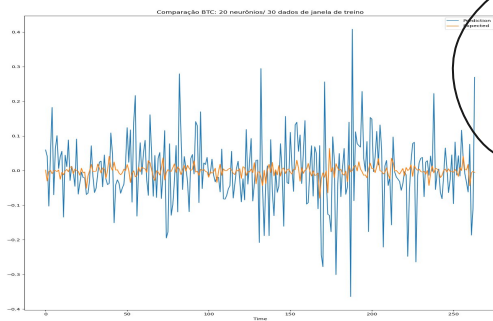
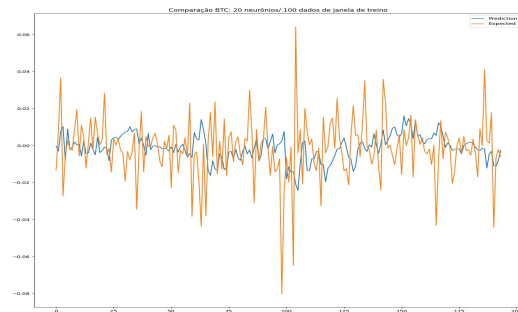
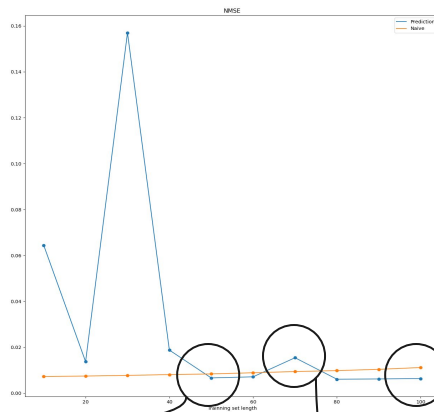
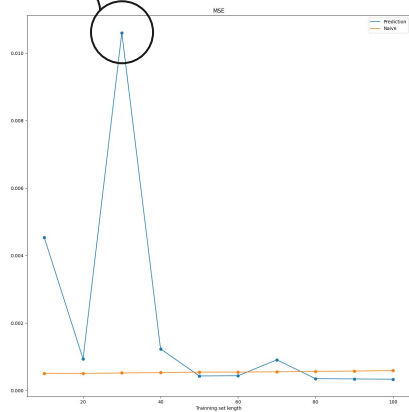
Analisamos o bias-variance tradeoff, modificamos o tamanho da rede (neurônios) e a quantidade de dados de *learning*, mantendo os parâmetros ótimos obtidos. A superfície de erro (MSE) está em escala logarítmica.













# Obrigado!

## Participantes:



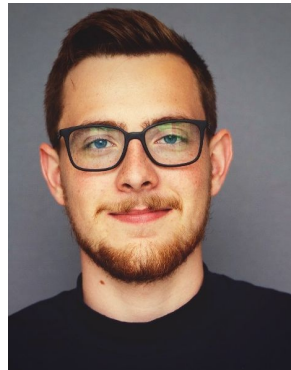
Gabriel Beber



Gustavo Bier

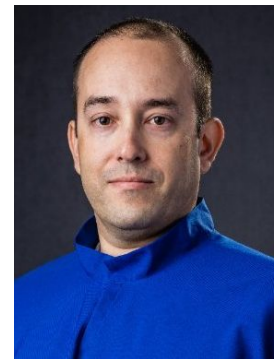


João Pergher



Kevin Pergher

## Colaboração especial:



Carlo Requião  
(ARIZONA STATE)



João Flores  
(IME - UFRGS)