



# **Desenvolvimento de um modelo gerador de séries temporais multivariadas para um compressor industrial offshore baseado em RNN e VAEs.**

CPE727 – Deep Learning

Rodrigo Petrus Domingues ([rodrigo.domingues@lps.ufrj.br](mailto:rodrigo.domingues@lps.ufrj.br))

10 de dezembro de 2025



# Motivação e Objetivos

## 1 Motivação e Objetivos

### Motivação

- Compressores são ativos críticos com sinais temporais complexos.
- Modelos gerativos permitem simular cenários operacionais realistas.
- Gêmeos digitais apoiam manutenção e análise operacional.

### Objetivo

- Desenvolver modelos geradores de séries temporais multivariadas para um compressor industrial.

### Avaliação

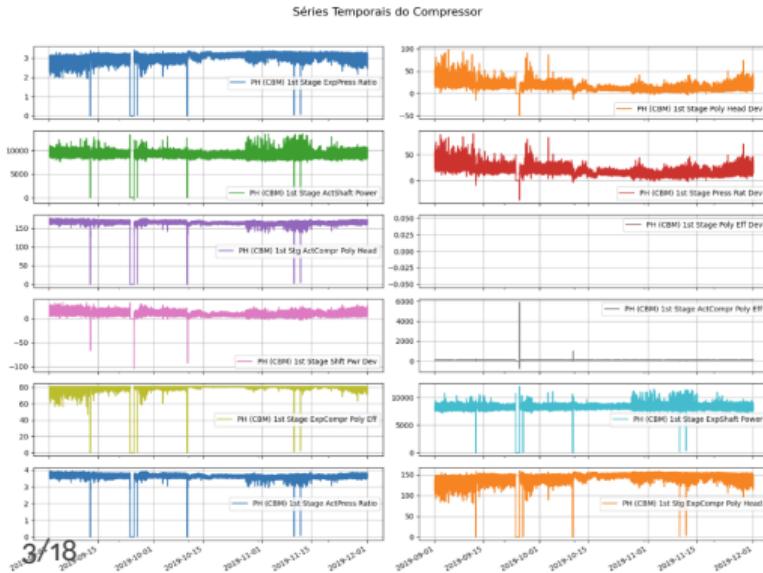
- Comparar RNNs (LSTM, GRU, BiRNN) e estratégias de treinamento com regularização e otimização.

# Base de dados e pré-processamento

## 2 Base de dados e pré-processamento

### Base de dados

Cognite — sinais reais de sensores de um compressor industrial offshore.



### Pré-processamento

- Estados operacionais:
  - Estados originais: 6 (0 a 5);
  - Estados finais: 3 (normal, alerta, falha).
- Normalização robusta (RobustScaler) por sensor;
- Série temporal com amostras a cada 5 minutos.
- Segmentação em janelas:
  - `window_size = 10, window_step = 10`
  - janelas não sobrepostas (modo gerador local).



# Definição do problema - Modo Gerador

## 3 Formulação do Problema

- Série multivariada:  $\mathbf{x}_t \in \mathbb{R}^d$ ,  $t = 1, \dots, T$ , com  $d = 11$  sensores.
- Construímos janelas temporais:

$$\mathbf{X} = (\mathbf{x}_{t-L+1}, \dots, \mathbf{x}_t) \in \mathbb{R}^{L \times d}$$

com  $L = 10$  passos de tempo.

- Objetivo: aprender um **modelo gerador** para as janelas  $\mathbf{X}$ :

$$p_{\theta}(\mathbf{X}) \approx p_{\text{dados}}(\mathbf{X})$$

- **Modelo latente (VAE + Difusão):** Aprende um espaço latente regularizado que permite gerar novas janelas sintéticas a partir de amostras de  $z$ .
- Uso principal:
  - Geração de séries realistas para simulação de cenários e testes de algoritmos de monitoramento.
- O modelo aprende a **recriar e amostrar** janelas de sensores compatíveis com o comportamento real do compressor.



# Revisão Bibliográfica

## 4 Revisão Bibliográfica

### Trabalhos relacionados:

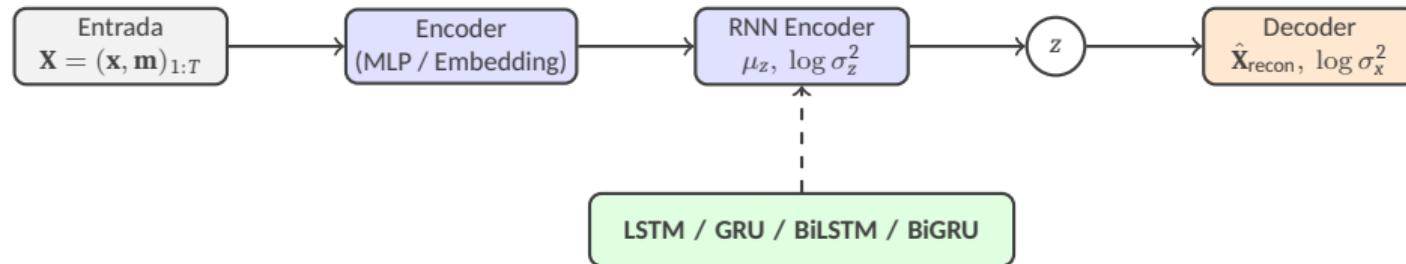
- VAEs amplamente usados para geração de dados sintéticos e aprendizado não supervisionado [1];
- RNNs combinadas com VAEs para modelagem temporal de séries multivariadas industriais [2];
- Métodos para séries com **missing data** que incorporam máscaras no estado recorrente [3];
- Modelos de difusão para geração probabilística de dados complexos [4].

### Lacuna abordada neste trabalho:

- Poucos estudos exploram a **geração probabilística de séries temporais industriais multivariadas** [5];
- Não encontramos aplicações em **compressores industriais** combinando RNNs, VAEs e difusão;
- Avaliamos geração via NELBO, calibração e sharpness.

# Arquitetura base e arquiteturas avaliadas

## 5 Arquiteturas e Variações



- **Arquitetura base:** Encoder recorrente acoplado a um VAE latente para geração de séries.
- **Arquiteturas avaliadas:**
  - **Deep MLP:** baseline feedforward (sem recorrência);
  - **LSTM:** LSTM unidirecional e **BiLSTM**;
  - **GRU:** GRU unidirecional e **BiGRU**.
  - **VAE:** Variational Autoencoder.



# Variações de treinamento

## 5 Arquiteturas e Variações

Base fixa: BiLSTM + VAE latente.

### Otimização

- **Warmup & Scheduler:** estabiliza as primeiras épocas.
- **RAdam:** reduz instabilidade do Adam no início do treino.

### Regularização

- **Variational Dropout:** reduz overfitting temporal.
- **Difusão (missingness):** melhora robustez e calibração.
- **LayerNorm:** estabiliza ativações internas.
- **AdamW:** L2 desacoplado para melhor generalização.

# Função de custo

## 5 Arquiteturas e Variações

- $\mathcal{L}_{\text{vae}} = \mathbb{E}_{q(z|\mathbf{X})}[-\log p(\mathbf{X} | z)] + \beta \text{KL}(q(z | \mathbf{X}) \| \mathcal{N}(0, I))$
- O primeiro termo corresponde à verossimilhança negativa do erro de reconstrução enquanto o segundo termo regulariza o espaço latente via divergência KL ponderada por  $\beta$ .
- Para o modelo com difusão de missingness, a função de custo é:

$$\mathcal{L}_{\text{difusão}} = \lambda_m \mathcal{L}_{\text{miss}} + \lambda_v \mathcal{L}_{\text{vae}}$$

$$\mathcal{L}_{\text{miss}} = \text{BCE}(m, \hat{m}), \quad \hat{m} = \sigma(f_{\text{miss}}(h))$$

- O modelo permite **gerar novas janelas sintéticas** de sensores com propriedades estatísticas similares às observadas.

# Configuração Experimental

## 6 Configuração Experimental

- Divisão treino/teste: 80% / 20%.
- Dimensões do espaço latente foram validadas com a divisão 60% / 20% / 20%.
- Treinamento por 500 épocas, batch size 256.
- Paciência de 50 épocas para early stopping.
- Função de perda multi-tarefa conforme descrito.
- Otimização e Regularização tratadas como diferentes modelos.
- Otimizador: Adam (exceto variações).
- Taxa de aprendizado inicial:  $3 \times 10^{-4}$ .
- Early stopping baseado no NELBO do teste.
- Avaliação final no conjunto de teste.

# Figuras de Mérito utilizadas

## 7 Figuras de Mérito

- **NELBO** (Negative Evidence Lower Bound):
  - Loss probabilística minimizada no treinamento;
  - Maximiza implicitamente a verossimilhança (ELBO);
  - Balanceia reconstrução e regularização latente (KL).
  - $NELBO = \mathbb{E}_{q(z|x)}[-\log p(x | z)] + KL(q(z | x) \| p(z))$
- **NLL** (Negative Log-Likelihood):
  - Generaliza o MSE ao modelar explicitamente a variância da distribuição predita;
  - Penaliza erros grandes e variâncias mal calibradas (super ou subestimação de incerteza).
  - $$NLL = -\log p(x | \mu, \sigma^2) = \frac{(x - \mu)^2}{2\sigma^2} + \frac{1}{2} \log \sigma^2 + \text{cte}$$

# Figuras de Mérito utilizadas

## 7 Figuras de Mérito

- **MSE (Mean Squared Error):**
  - Erro médio de reconstrução das séries;
  - Mede fidelidade gerativa.
- **Cobertura 90% (cov\_90):**
  - Mede a fração de amostras reais que caem dentro do intervalo preditivo teórico [5%, 95%];
  - Avalia se o desvio padrão estimado produz uma **calibração probabilística consistente** com a cobertura nominal de 90%.
- **Largura do intervalo 90% (width\_90):**
  - Corresponde à largura teórica do intervalo [5%, 95%] derivado da distribuição predita;
  - Quantifica a **sharpness** do modelo: intervalos menores indicam maior confiança, desde que a cobertura permaneça bem calibrada.

# Resultados quantitativos

## 8 Resultados

Modelo	NELBO↓	NLL↓	MSE↓	Cov90→	W90↓	Tempo (min)
<b>GRU Difusão</b>	<b>0.510</b>	<b>0.456</b>	$0.237 \pm 0.064$	<b>0.897</b>	<b>2.213</b>	<b>10.1</b>
BiLSTM Difusão	0.518	0.462	$0.243 \pm 0.067$	0.902	2.200	11.2
GRU	0.558	0.498	$0.300 \pm 0.108$	0.905	2.252	7.7
LSTM	0.588	0.513	$0.332 \pm 0.129$	0.900	2.251	7.4
BiGRU	0.589	0.499	$0.335 \pm 0.136$	0.909	2.267	7.7
BiLSTM VarDrop 0.2	0.595	0.506	$0.317 \pm 0.119$	0.898	2.240	7.5
BiLSTM RAdam	0.597	0.511	$0.341 \pm 0.139$	0.898	2.234	15.1
BiLSTM Warmup/Sched.	0.598	0.504	$0.354 \pm 0.146$	0.903	2.251	6.5
BiLSTM AdamW	0.601	0.506	$0.341 \pm 0.141$	0.900	2.224	7.0
BiLSTM	0.609	0.527	$0.337 \pm 0.133$	0.901	2.269	7.4
BiLSTM LayerNorm	0.617	0.514	$0.342 \pm 0.140$	0.907	2.281	5.9
DEEP	1.217	1.164	$0.693 \pm 0.209$	0.916	2.785	8.9

Comparação probabilística entre modelos.

# Curva de treinamento — Modelo campeão

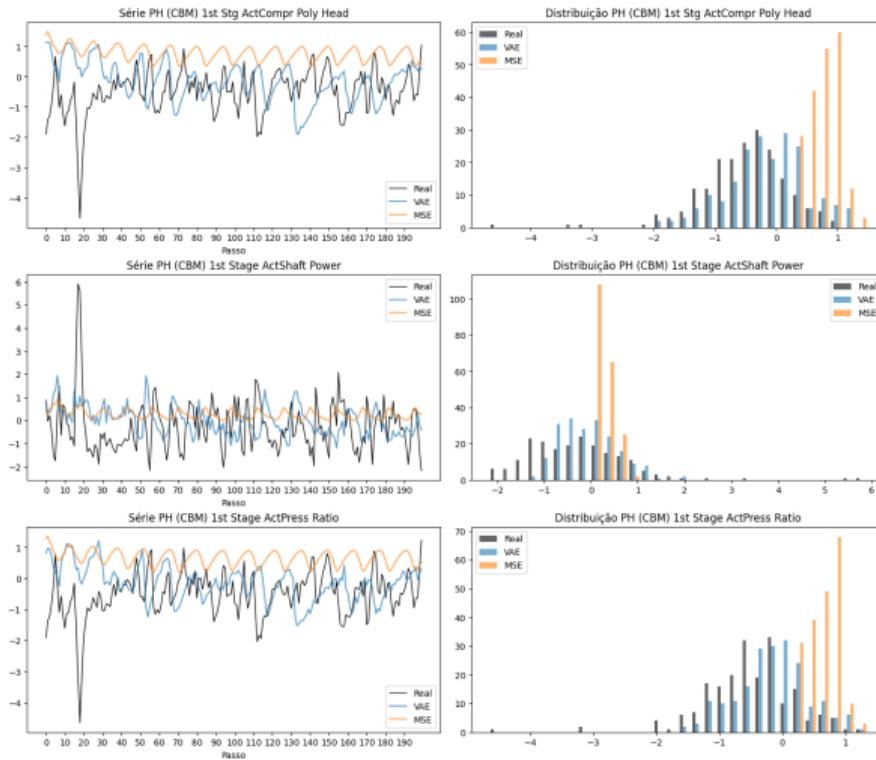
## 8 Resultados



Figura: Evolução do NELBO no conjunto de teste do modelo BiLSTM Difusão.

# Geração de séries — Avaliação qualitativa

## 8 Resultados



## Comparação qualitativa

Séries temporais reais versus séries geradas pelo modelo variacional, comparadas a ajustes ponto-a-ponto baseados em MSE.

A abordagem probabilística captura melhor:

- a distribuição marginal dos sinais;
- a variabilidade temporal;
- a incerteza intrínseca dos dados.



# Discussão dos resultados

## 9 Discussão e Conclusões

- **Qualidade gerativa:** Modelos variacionais recorrentes superam o baseline; **GRU + Difusão** atinge o menor NELBO.
- **Calibração vs. sharpness:** Difusão produz intervalos mais estreitos, com cobertura próxima do alvo ( $\text{Cov90} \approx 0.9$ ).
- **Impacto arquitetural:** GRU + Difusão oferece o melhor compromisso entre NELBO, NLL e MSE, com boa calibração.
- **Regularização:** Difusão é o principal mecanismo eficaz; dropout tem efeito secundário e LayerNorm degrada resultados.

# Conclusões

## 9 Discussão e Conclusões

- Modelos recorrentes com difusão são mais eficazes para geração probabilística de séries industriais.
- **Modelo campeão: GRU + Difusão**, com desempenho muito próximo do BiLSTM + Difusão.
- O modelo gerou séries próximas a realidade e com calibração da distribuição probabilística adequada.
- Potencial de aplicação: gêmeos digitais e simulação de cenários críticos.



# Referências Bibliográficas

10 Referências Bibliográficas

- [1] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," in *International Conference on Learning Representations (ICLR)*, 2014.
- [2] J. Chung, K. Kastner, L. Dinh, K. Goel, A. Courville, and Y. Bengio, "A recurrent latent variable model for sequential data," in *Advances in Neural Information Processing Systems*, vol. 28, 2015.
- [3] Z. Che *et al.*, "Recurrent neural networks for multivariate time series with missing values," *Scientific Reports*, vol. 8, no. 1, p. 6085, 2018.
- [4] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," in *Proceedings of NeurIPS*, 2020.



# Referências Bibliográficas

10 Referências Bibliográficas

- [5] C. Esteban, S. Hyland, and G. Rätsch, “Timegan: Generative adversarial networks for synthetic time series,” in *Advances in Neural Information Processing Systems*, 2019.