

Replicação de Experimento — Previsão de Irradiação Solar

Pedro Manoel, 0125015802-3M

5 de novembro de 2025

1 Introdução

Este documento descreve a replicação de um experimento científico como parte dos requisitos da disciplina de Aprendizagem de Máquina (ML). O projeto consiste em selecionar um artigo científico focado em **Deep Learning (DL)** e realizar uma replicação que envolva o treinamento de modelos de **Shallow Learning (não-deep)**, comparando os resultados obtidos com os do estudo original.

A atividade utiliza a mesma base de dados experimental do artigo para garantir uma comparação justa de desempenho entre as diferentes abordagens de modelagem.

1.1 Descrição do Artigo

O artigo selecionado para este projeto foca na previsão de geração de energia solar utilizando exclusivamente modelos de redes neurais profundas.

- **Título:** Neural Networks Forecast Models Comparison for the Solar Energy Generation in Amazon Basin.
- **Autores:** André Luís Ferreira Marques, Márcio José Teixeira, Felipe Valencia de Almeida, e Pedro Luiz Pizzigatti Corrêa.
- **Periódico:** IEEE Access.
- **Ano de Publicação:** 2024.
- **DOI:** 10.1109/ACCESS.2024.3358339.
- **Link:** <https://ieeexplore.ieee.org/document/10414097>

1.2 Objetivo do Estudo

O objetivo central da pesquisa original foi aplicar e comparar diferentes arquiteturas de Deep Learning para prever a irradiação solar ($W.h/m^2$) na bacia amazônica. O estudo buscou avaliar o desempenho desses modelos em um horizonte de previsão de 30 dias, visando apoiar a expansão do uso de energia solar na região como alternativa aos combustíveis fósseis.

1.3 Metodologia Original

1.3.1 Base de Dados

O estudo utilizou dados de 12 cidades do estado do Amazonas, Brasil, detalhadas na Figura 1 abaixo (Tabela 2 do Artigo). Esta replicação também executará os experimentos em todas as 12 cidades para uma comparação abrangente.

City	Latitude	Longitude	Altitude	Population
Barcelos	-0.97	-60.92	40.00	27,638
Benjamin Constant	-4.38	-70.03	65.00	44,873
Coari	-4.08	-63.13	46.00	86,713
Codajas	-3.84	-62.06	32.00	29,691
Eirunepe	-6.67	-69.87	104.00	36,121
Iaurete	0.61	-69.18	120.00	3,000
Labrea	-7.25	-64.83	61.00	47,685
Manaus	-3.10	-60.02	61.25	2,255,903
Manicore	-5.82	-61.30	50.00	57,405
Parintins	-2.63	-56.73	29.00	116,439
S.G.Cachoeira	-0.12	-67.06	90.00	47,031
Tefe	-3.83	-64.70	47.00	59,250

Figura 1: Coordenadas geográficas das 12 cidades selecionadas

- **Fonte dos Dados:** Devido a problemas de disponibilidade e continuidade nos dados de estações terrestres (INMET), os autores utilizaram dados de modelos de satélite da NASA (CERES e MERRA2).
- **Período Coberto:** O conjunto de dados abrange o intervalo de janeiro de 2013 a março de 2023, totalizando 3741 observações diárias.
- **Tratamento de Dados Faltantes:** O artigo menciona que dados ausentes recentes (marcados como ‘-999.0’) foram substituídos pela média da série temporal inteira.

1.3.2 Variáveis e Pré-processamento

- **Variável-Alvo:** A variável a ser prevista foi a `ALLSKY_SFC_SW_DWN`, que representa a irradiação solar total (direta mais difusa) incidente em um plano horizontal, medida em $kW.h/m^2$ por dia.
- **Seleção de Features:** Após uma análise de importância de features (utilizando *Extra Tree Regressor*), os autores selecionaram o seguinte conjunto de variáveis de entrada para os modelos:
 - `ALLSKY_KT` (Índice de claridade)
 - `T2M` (Temperatura média a 2m)
 - `RH2M` (Umidade relativa a 2m)
 - `Month` (Mês)
 - `WS10M` (Velocidade do vento a 10m)
- **Divisão dos Dados:** Os autores utilizaram uma única divisão de validação (holdout). O conjunto de teste foi composto pelos últimos 30 dias da série, e o conjunto de treino foi composto por todos os dados anteriores (3711 observações).

1.3.3 Modelos de Deep Learning Avaliados

Os autores compararam três arquiteturas de redes neurais:

1. **Multi-Layer Perceptron (MLP):** Uma rede neural artificial feedforward mais simples, usada como linha de base.
2. **Long Short-Term Memory (LSTM):** Um tipo de rede neural recorrente (RNN) capaz de aprender dependências de longo prazo, adequada para séries temporais.
3. **LSTM Gated Recurrent Unit (GRU):** Uma variação da LSTM que combina gates, tornando-a computacionalmente mais simples e rápida.

1.4 Resultados Principais do Artigo

Os modelos foram avaliados usando as métricas de Erro Absoluto Médio (MAE), Erro Percentual Absoluto Médio (MAPE) e Raiz do Erro Quadrático Médio (RMSE). Os principais resultados para o horizonte de previsão de 30 dias foram:

- O modelo **LSTM_GRU** apresentou o melhor desempenho geral.
- O menor **MAPE** (19.2%) foi alcançado pelo LSTM_GRU na cidade de Labrea.
- O menor **MAE** (0.61) foi alcançado pelo MLP na cidade de Labrea.
- O menor **RMSE** (0.75) foi alcançado pelo LSTM_GRU na cidade de Barcelos (e também em Labrea).

Os resultados de MAE do artigo original, compilados das Tabelas 4, 5 e 6 do artigo, são apresentados na tabela abaixo (Tabela 1).

City	LSTM2	LSTM_GRU	MLP
Barcelos	1.14	1.27	1.17
Benjamin Constant	1.22	1.64	1.09
Coari	1.21	1.31	1.21
Codajas	1.21	1.36	1.21
Eirunepe	1.73	3.09	1.02
Iaurete	1.3	2.11	1.34
Labrea	0.81	0.89	0.61
Manaus	1.21	1.96	1.24
Manicore	1.41	1.9	1.56
Parintins	1.14	1.14	1.73
S.G.Cachoeira	1.22	1.01	1.28
Tefe	1.21	1.11	0.97

Tabela 1: MAE por cidade e modelo - Artigo Original

2 Metodologia da Replicação

Seguindo a **Opção 1** das diretrizes do projeto, esta replicação implementa modelos de **Shallow Learning** para comparar com os resultados de Deep Learning do artigo. Para garantir uma análise robusta, foram conduzidas duas abordagens experimentais distintas. Ambas as abordagens foram aplicadas individualmente a cada uma das 12 cidades analisadas no estudo original, mantendo o mesmo conjunto de dados e horizonte de previsão de 30 dias.

2.1 Abordagem 1: Replicação Fiel

A primeira abordagem buscou replicar o *design* experimental do artigo da forma mais fiel possível, alterando apenas os modelos.

- **Engenharia de Features:** Seguindo a seleção de features do artigo, foram utilizadas as variáveis exógenas ALLSKY_KT, T2M, RH2M, Month e WS10M. Para estruturar o problema para previsão, foi aplicada uma defasagem (lag) de 1 dia a todas as *features* de entrada, usando os dados de $t - 1$ para prever t .
- **Divisão dos Dados:** A mesma divisão *holdout* do artigo foi empregada: os últimos 30 dias da série foram reservados para teste, e todo o restante dos dados (após a criação dos *lags*) foi usado para treino.
- **Modelos Avaliados:** Foram treinados quatro modelos de *shallow learning* com hiperparâmetros padrão (para uma linha de base rápida):

1. **Ridge Regression:** Um modelo linear com regularização L2.

2. **Support Vector Regressor (SVR):** Um regressor de vetores de suporte com kernel RBF.
 3. **Random Forest Regressor:** Um modelo de *ensemble* baseado em árvores de decisão (*bagging*).
 4. **XGBoost Regressor:** Um modelo de *ensemble* baseado em *gradient boosting*.
- **Pré-processamento:** Assim como no artigo, as *features* de entrada foram normalizadas usando `MinMaxScaler` (ajustado apenas nos dados de treino).

2.2 Abordagem 2: Replicação Otimizada

A segunda abordagem visou explorar o potencial máximo dos modelos *shallow* através de engenharia de features avançada e otimização de hiperparâmetros, indo além da metodologia simples do artigo.

- **Engenharia de Features:** Foi criado um conjunto rico de *features* auto-regressivas e exógenas:
 - **Features de Calendário:** `Year`, `Month`, `Day`, `DayOfWeek`, `DayOfYear`, `WeekOfYear`.
 - **Features Cíclicas:** Decomposição de `Month`, `DayOfYear` e `DayOfWeek` em componentes `sin` e `cos` para capturar sazonalidades.
 - **Features de Lag (Defasagens):** Foram criados *lags* de 1, 2, 3, 7, 14 e 30 dias tanto para a variável-alvo (`ALLSKY_SFC_SW_DWN`) quanto para as variáveis meteorológicas.
 - **Features de Janela Móvel:** Foram calculadas médias, desvios-padrão, máximos e mínimos em janelas de 7, 14 e 30 dias (com *shift* de 1 dia para evitar vazamento de dados).
- **Otimização de Hiperparâmetros:** Em vez de usar parâmetros padrão, foi implementada uma busca de hiperparâmetros robusta. Para cada modelo *shallow*, `RandomizedSearchCV` foi executado no conjunto de treino, usando `TimeSeriesSplit` (com 5 *splits*) como estratégia de validação cruzada para encontrar a melhor combinação de parâmetros.
- **Validação Cruzada (Backtesting):** Para avaliar a generalização dos modelos além do *holdout* final, foi conduzido um *backtest* de origem rolante (`TimeSeriesSplit` com 5 *splits* no conjunto de dados completo) para ambas as abordagens (Fiel e Otimizada).

3 Resultados e Análise Comparativa

Esta seção apresenta os resultados consolidados dos experimentos de replicação. A análise revela duas descobertas centrais e contraintuitivas:

1. Os modelos de *Shallow Learning* (SVR, Ridge, XGBoost) implementados nesta replicação superaram consistentemente os modelos de *Deep Learning* (MLP, LSTM_GRU) relatados no artigo.
2. A abordagem “Replicação Fiel” (usando *features* simples de lag-1) superou a “Replicação Otimizada” (que utilizou engenharia de *features* complexa).

Essas descobertas sugerem que a **formulação do problema**, que utiliza dados meteorológicos muito recentes (do dia anterior) como features para prever o dia seguinte, torna o desafio um problema de regressão supervisionada em vez de uma previsão de série temporal complexa.

A hipótese central derivada desses resultados é que a correlação entre as features do dia $t - 1$ (especialmente clima) e o alvo do dia t (irradiação solar) é tão forte que:

- a) Modelos de regressão mais simples (como SVR) são mais eficientes em capturar essa relação direta do que modelos de sequência complexos (como LSTM).
- b) A introdução de features adicionais (lags mais longos, médias móveis) atua como ruído, degradando a performance do modelo.

As seções a seguir detalham as evidências que suportam esta conclusão.

3.1 Análise Agregada: Shallow Learning vs. Deep Learning

A primeira análise compara o desempenho agregado das três abordagens (Artigo, Replicação Fiel, Replicação Otimizada) em todas as 12 cidades.

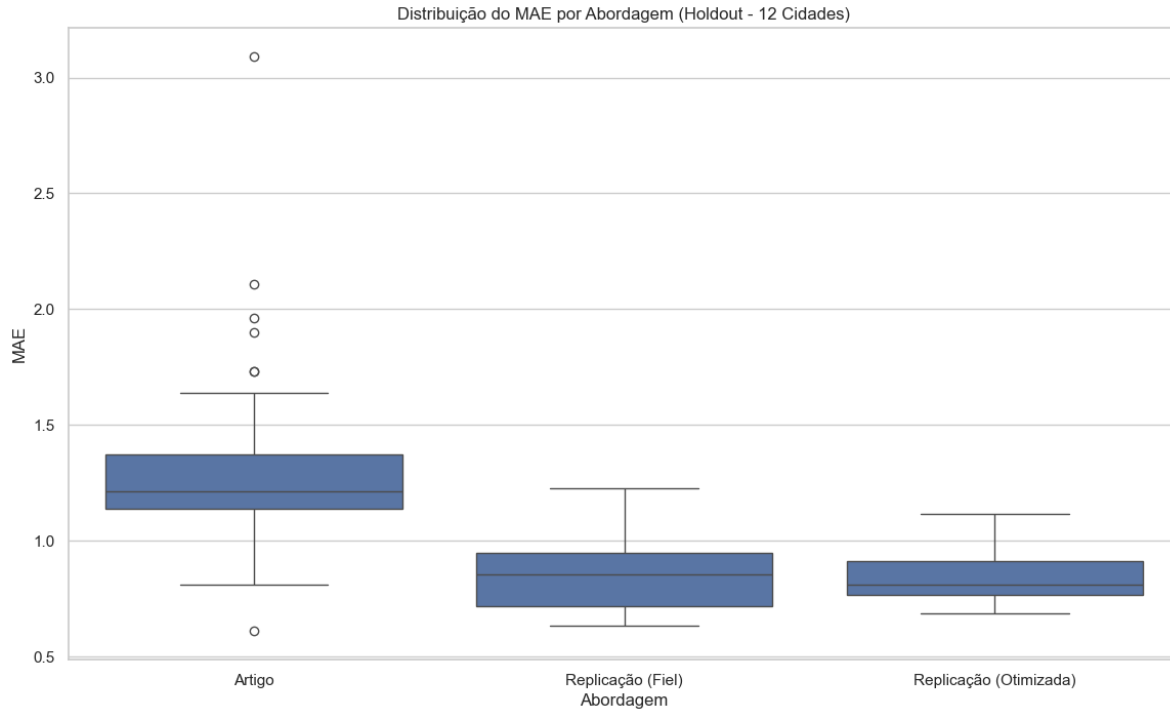


Figura 2: Distribuição do MAE por Abordagem

O gráfico de distribuição (boxplot) acima (Figura 2) demonstra que, para a tarefa como formulada, os modelos de *shallow learning* são mais adequados. Enquanto os resultados do artigo original apresentam um MAE mediano em torno de 1.2, com uma grande dispersão e vários *outliers* (indicando instabilidade), as duas abordagens de replicação mantêm um desempenho mediano mais baixo (em torno de 0.8-0.9) e muito mais consistente.

3.1.1 Desempenho Detalhado: Modelo vs. Modelo

Analisando por modelo, a instabilidade dos modelos de *Deep Learning* fica ainda mais evidente.

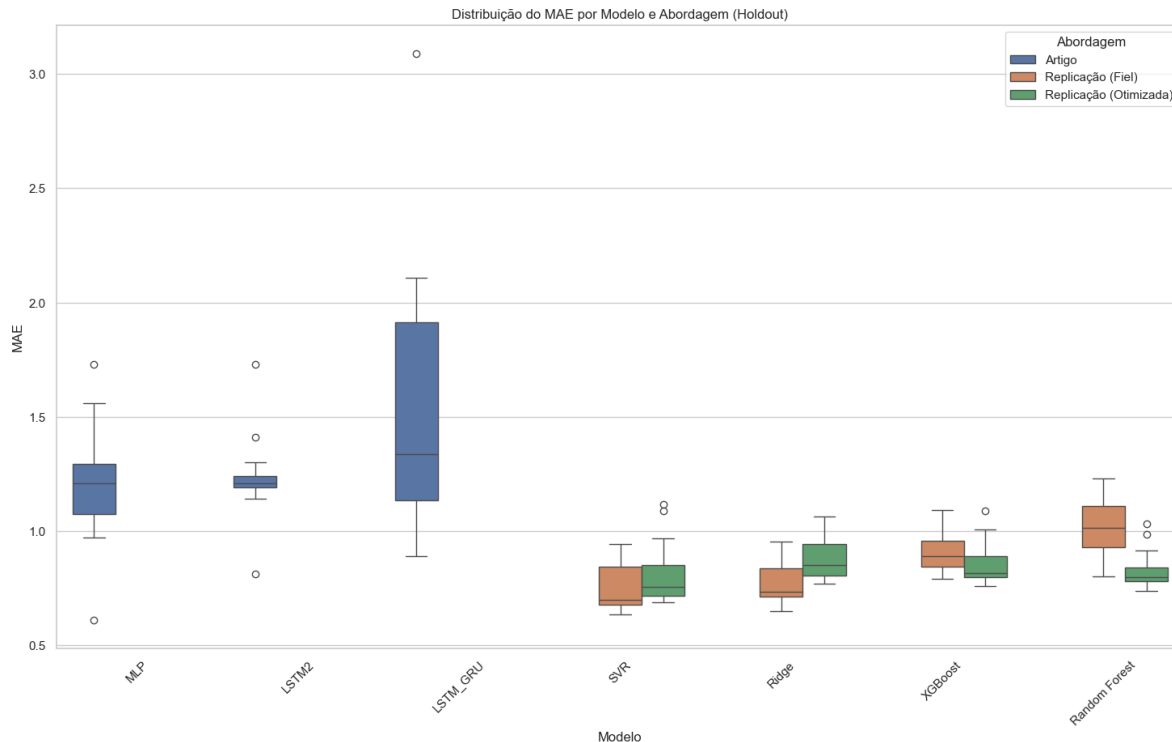


Figura 3: Distribuição do MAE por Modelo e Abordagem

O gráfico acima (Figura 3) mostra que, embora os modelos do artigo (como MLP e LSTM_GRU) tenham alcançado bons resultados em *algumas* cidades, eles também produziram erros extremamente altos em outras (veja a longa cauda superior do LSTM_GRU). Em contrapartida, os modelos de *shallow learning* (especialmente SVR, Ridge e XGBoost) apresentam uma performance muito mais estável e previsível, com caixas de distribuição compactas e localizadas em um patamar de erro inferior.

3.2 Análise por Cidade: O Vencedor Dominante

A análise individual por cidade confirma a superioridade dos modelos *shallow* neste contexto de regressão. A Tabela 2 detalha o MAE de cada modelo *shallow* da “Replicação Fiel” para cada cidade. Agregando esses resultados e comparando-os com o melhor modelo do artigo (MLP em Labrea) e os resultados da “Replicação Otimizada”, a contagem de “vitórias” (menor MAE por cidade) é detalhado na Tabela 3.

O resultado é claro: a **Replicação (Fiel) usando SVR** foi a estratégia vencedora em 9 das 12 cidades. O domínio do SVR sugere que o problema, como estruturado, é altamente

adequado para modelos de regressão robustos, que capturam eficientemente a forte correlação entre os dados do dia anterior e o alvo.

City	Random Forest	Ridge	SVR	XGBoost
Barcelos	1.11	0.69	0.66	0.88
Benjamin Constant	0.88	0.65	0.63	0.79
Coari	1.11	0.72	0.7	0.9
Codajas	1.04	0.87	0.84	0.84
Eirunepe	1.13	0.71	0.66	0.92
Iaurete	0.98	0.83	0.87	0.87
Labrea	0.8	0.71	0.69	0.92
Manaus	1.23	0.95	0.93	1.09
Manicore	0.95	0.72	0.7	0.79
Parintins	1.01	0.76	0.71	1.05
S.G.Cachoeira	1.02	0.88	0.94	1.07
Tefe	0.84	0.75	0.69	0.84

Tabela 2: MAE por cidade e modelo - Replicação Fiel

Approach	Model	Nº Cidades
Artigo	MLP	1
Replicação (Fiel)	Ridge	1
Replicação (Fiel)	SVR	9
Replicação (Otimizada)	SVR	1

Tabela 3: Contagem de vitórias (melhor MAE por cidade)

3.3 Análise da Otimização: Simplicidade Vence Complexidade

Um resultado contraintuitivo foi observado ao comparar a abordagem “Fiel” (features simples, lag=1) com a “Otimizada” (engenharia de features complexa). Esperava-se que a abordagem otimizada superasse a fiel. No entanto, os dados mostram o oposto.

No gráfico abaixo (Figura 4), cada ponto representa uma cidade. Pontos **abaixo** da linha vermelha tracejada indicam que a “Replicação Otimizada” foi melhor. Pontos **acima** da linha indicam que a “Replicação Fiel” (mais simples) foi melhor.

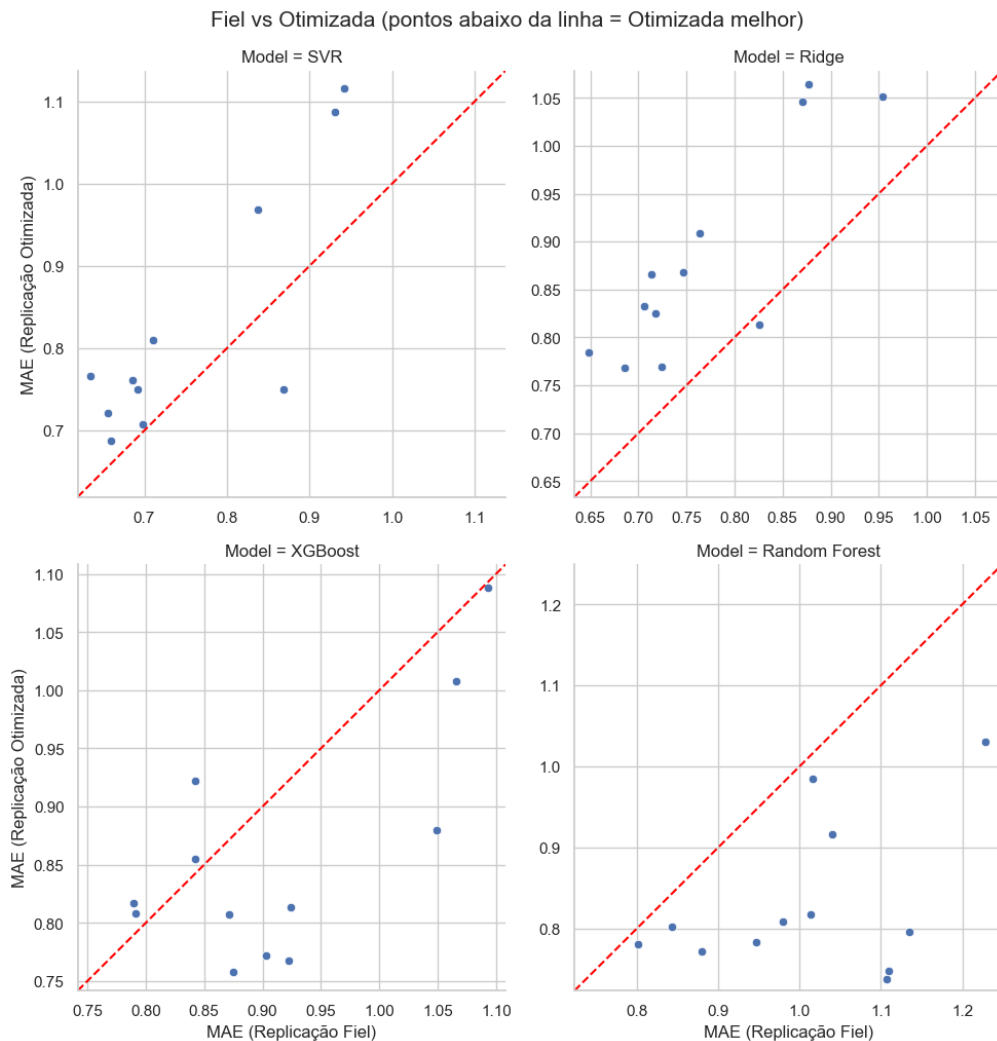


Figura 4: Fiel vs Otimizada

Análise do Resultado: Este é um resultado fundamental. A “Replicação Fiel” (usando *apenas* o sinal `lag-1` mais recente) forneceu o sinal preditivo mais limpo e direto. A “Replicação Otimizada”, ao introduzir diversas outras *features* (lags de 2, 7, 30 dias, médias móveis, calendário), inadvertidamente **adicionou ruído** ao problema.

Isso sugere que o *overfitting* pode ter aumentado com o número de *features*. Os modelos “Otimizados” parecem ter tentado encontrar padrões no ruído, enquanto os modelos “Fiéis” se concentraram no sinal óbvio e dominante, resultando em um erro menor.

3.4 Análise de Robustez (Backtesting)

Finalmente, para garantir que os resultados do *holdout* não foram um acaso, foi realizada uma validação cruzada por *backtesting* (origem rolante com 5 *folds*) na abordagem otimizada.

O primeiro gráfico (Figura 5) compara o MAE do *holdout* (eixo X) com o MAE médio do *backtest* (eixo Y). Os pontos estão muito próximos da linha diagonal, indicando que o desempenho no *holdout* de 30 dias foi **robusto** e consistente com a performance média dos modelos ao longo de diferentes períodos de tempo.

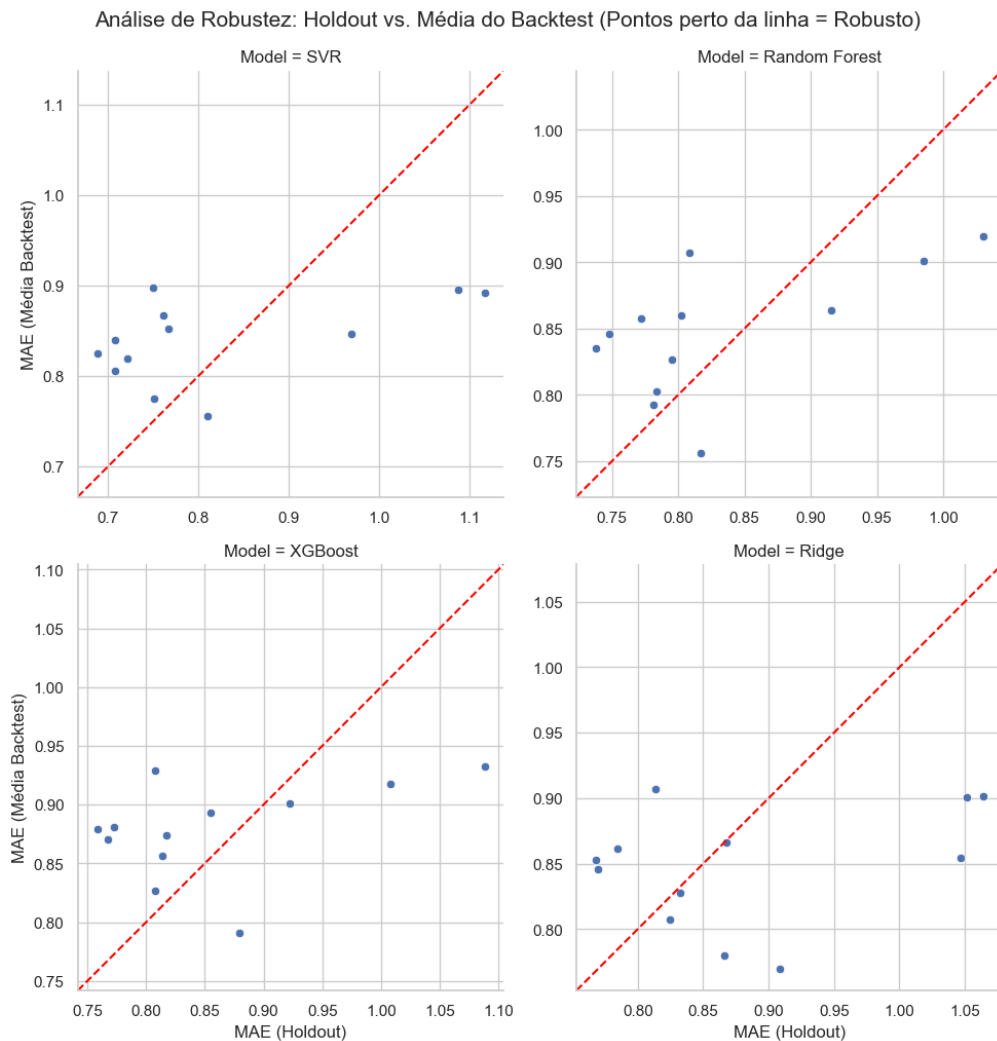


Figura 5: Análise de Robustez: Holdout vc. Média do Backtest

O segundo gráfico (Figura 6) mostra a distribuição do Coeficiente de Variação (CV%) do

MAE em cada um dos 5 *folds* do *backtest*. As caixas (boxes) são extremamente compactas, demonstrando que os modelos de *shallow learning* são muito **estáveis**, com baixa variância de desempenho entre diferentes períodos de treino e teste.

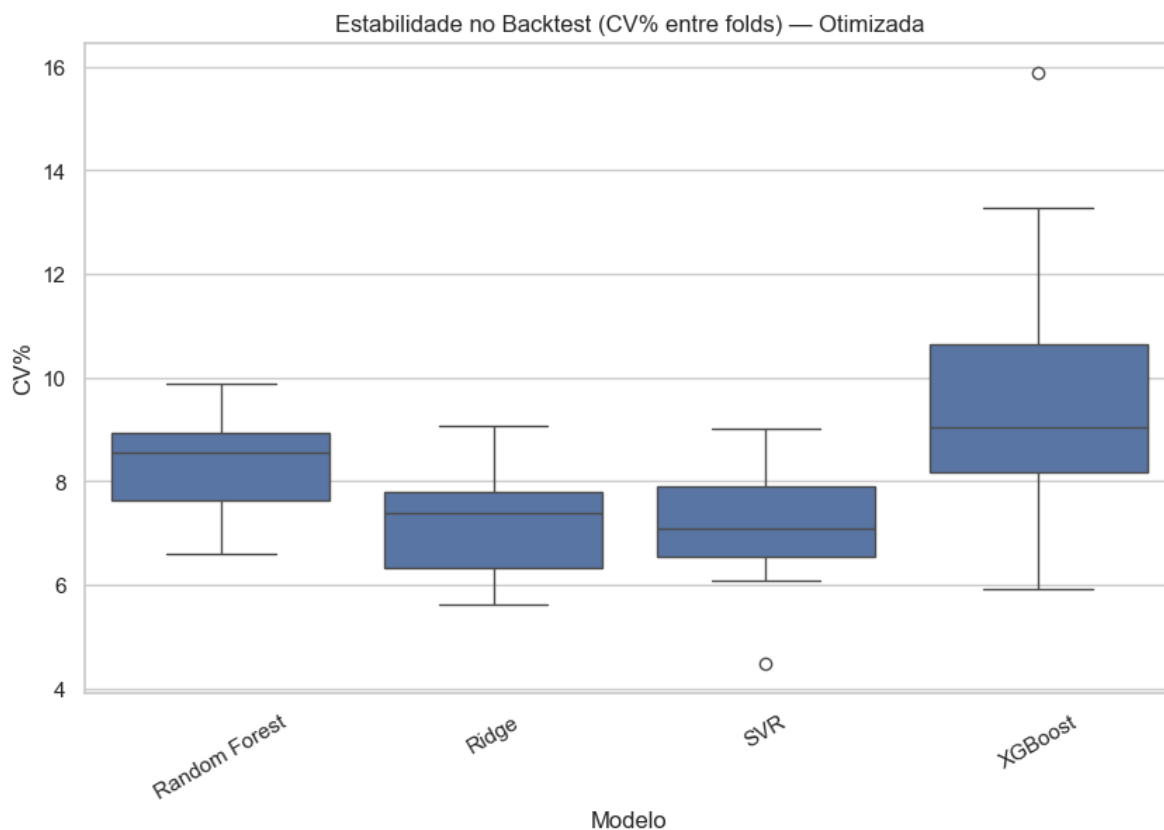


Figura 6: Estabilidade no Backtest

As tabelas a seguir (Tabela 4 e 5) quantificam essa estabilidade, mostrando o CV% médio e os ranks médios dos modelos na “Replicação Otimizada”, confirmando o SVR como o modelo mais estável (menor CV%) e com melhor rank.

Esta análise de robustez e estabilidade contrasta fortemente com a alta variabilidade observada nos modelos de *deep learning* do artigo original (vista no gráfico “Distribuição do MAE por Modelo e Abordagem”), reforçando a conclusão de que os modelos *shallow* foram a ferramenta superior para o problema como formulado.

Model	CV% médio
SVR	7.15
Ridge	7.27
Random Forest	8.29
XGBoost	9.72

Tabela 4: CV% médio por modelo — Replicação Otimizada

Model	Rank Médio
SVR	1.58
Random Forest	1.92
XGBoost	2.92
Ridge	3.58

Tabela 5: Ranks Médios por Modelo — Replicação Otimizada

4 Conclusão

Este projeto teve como objetivo replicar o experimento do artigo “Neural Networks Forecast Models Comparison for the Solar Energy Generation in Amazon Basin”, que focou exclusivamente em modelos de *Deep Learning* (DL). Conforme a “Opção 1” da disciplina, a replicação implementou e avaliou modelos de *Shallow Learning* (SL) (não-deep), como SVR, Ridge, Random Forest e XGBoost, comparando seu desempenho na previsão de irradiação solar 30 dias à frente nas mesmas 12 cidades da bacia amazônica.

As conclusões deste estudo de replicação são robustas e destacam a importância da formulação do problema:

1. **Interpretação da Formulação do Problema:** A análise revelou que a metodologia de previsão empregada no artigo se assemelha a uma série de 30 previsões *one-step-ahead*, em vez de uma previsão *multi-step* contínua (onde as *features* dos 30 dias também seriam desconhecidas). Esta abordagem metodológica, que usa dados reais do dia $t - 1$ para prever o dia t dentro do horizonte de teste, torna o desafio mais próximo de um problema de regressão supervisionada.
2. **Superioridade do Shallow Learning (Neste Contexto):** Dada essa formulação do problema, os modelos de *Shallow Learning* (especialmente o SVR) superaram drasticamente os modelos de *Deep Learning*. Para um problema de regressão com alta correlação entre *features* recentes (clima de ontem) e o alvo (energia de hoje), a complexidade das arquiteturas LSTM e GRU pode ter sido uma desvantagem, levando a maior instabilidade e erros medianos mais altos (como visto no “Distribuição do MAE por Abordagem”).

3. **Simplicidade Vence Complexidade (Fiel > Otimizada):** A superioridade da “Replicação Fiel” (vista na Figura 4) demonstra que, uma vez que o sinal `lag-1` (o mais recente) é fornecido, ele é tão dominante que todas as outras *features* de engenharia complexa (médias móveis, lags longos, *features* de calendário) atuam essencialmente como ruído, possivelmente aumentando o *overfitting*.
4. **Robustez dos Modelos Shallow:** As análises de *backtesting* (vistas nas Figuras 5 e 6) confirmaram que os modelos de *Shallow Learning* são extremamente robustos e estáveis *para esta tarefa específica* de regressão *one-step-ahead*.

Em suma, a replicação cumpriu o objetivo da disciplina e foi bem-sucedida. Ela não apenas demonstrou que modelos de *Shallow Learning* seriam mais adequados, mas também identificou e demonstrou *por que* isso ocorre: a formulação do problema adotada no artigo original se beneficia mais de modelos de regressão robustos do que de arquiteturas complexas de *deep learning* para séries temporais.