

AR-PolieDRO: *framework* de regressão DRO aplicado a séries temporais

João Quintella do Couto

PUC - Rio

joaoqcouto@aluno.puc-rio.br

Gustavo Monteiro Rohenkohl

PUC - Rio

gustavo.rohenkohl@aluno.puc-rio.br

Matheus Carneiro Nogueira

PUC - Rio

matnogueira@gmail.com

Davi Michel Valladão

PUC - Rio

davimv@puc-rio.br

RESUMO

Este trabalho investiga a aplicação do *framework* PolieDRO na estimação de modelos autorregressivos $AR(p)$ em séries temporais. O PolieDRO é um *framework* de regressão derivado de formulações de otimização robusta a distribuições (DRO), que não requer escolha de hiperparâmetros. Para avaliar sua viabilidade, são conduzidos experimentos com séries temporais simuladas de diferentes ordens e comprimentos, com e sem outliers. Os resultados mostram que o PolieDRO apresenta desempenho comparável ao modelo AR estimado via máxima verossimilhança. Embora o benchmark seja superior em alguns cenários, especialmente com outliers, o PolieDRO demonstra potencial como estimador alternativo robusto e sem necessidade de ajustes manuais.

PALAVRAS CHAVE. Séries Temporais. PolieDRO. ARIMA.

EST&AM – PO Analytics em Estatística e Aprendizado de Máquina; MOI-Métodos de Otimização sob Incerteza (estocásticos e robustos)

ABSTRACT

This work investigates the application of the PolieDRO framework to the estimation of autoregressive $AR(p)$ models in time series. PolieDRO is a regression framework derived from distributionally robust optimization (DRO) formulations, which does not require hyperparameter tuning. To assess its viability, experiments are conducted using simulated time series of varying orders and lengths, both with and without outliers. Results show that PolieDRO performs comparably to AR models estimated via maximum likelihood. Although the benchmark is superior in some scenarios, especially in the presence of outliers, PolieDRO demonstrates potential as a robust alternative estimator that requires no manual tuning.

KEYWORDS. Time Series. PolieDRO. ARIMA.

EST&AM – PO Analytics in Statistics and Machine Learning; MOI – Optimization Methods under Uncertainty (stochastic and robust)

1. Introdução

A previsão de séries temporais é fundamental para as mais diversas áreas do conhecimento, permeando por exemplo economia, finanças, meteorologia e engenharia. A capacidade de prever eventos futuros com base em dados históricos é crucial para a tomada de decisões informadas.

Dentro da literatura de séries temporais, diferentes modelos estatísticos clássicos são adotados conforme o contexto da aplicação. Os modelos da família ARIMA, desenvolvidos por Box et al. [2015], utilizam polinômios autorregressivos e de médias móveis para descrever a série temporal, tendo sido, por exemplo, o modelo mais assertivo na previsão de tráfego de curto prazo em Lippi et al. [2013]. O modelo ETS, proposto em Hyndman et al. [2008], decompõe a série em componentes de erro, tendência e sazonalidade, e apresentou a melhor performance em 1 dos 5 conjuntos de dados de previsão de demanda analisados por Abbasimehr et al. [2020]. De forma semelhante, os modelos estruturais em espaço de estados de Harvey [1990] também se baseiam em componentes não observáveis e foram comparados com SARIMA na previsão de séries macroeconômicas do Reino Unido em Harvey e Todd [1983].

Além dos modelos próprios de séries temporais, técnicas de regressão também são aplicadas nesse contexto. A regressão linear simples pode ser utilizada para modelar a série temporal utilizando outras séries como variáveis explicativas, como mostra Hyndman [2018]. O modelo SARIMAX (Seasonal AutoRegressive Integrated Moving Average with eXogenous variables) expande o ARIMA de Box et al. [2015] para incluir variáveis exógenas e componentes sazonais, sendo útil em contextos onde fatores externos influenciam as séries temporais. O trabalho de Medeiros et al. [2021] estudou as propriedades assintóticas do AdaLasso para modelos de séries temporais com dados em alta dimensão e esparsos, concluindo que essa técnica de regressão regularizada é adequada para previsão da inflação americana, apresentando melhores resultados que modelos de fatores e autoregressivos.

Modelos de aprendizado de máquina vêm se consolidando como uma alternativa promissora para previsão de séries temporais, especialmente em contextos de grandes volumes de dados ou relações não-lineares complexas. Medeiros et al. [2021] mostraram que, para previsão de inflação, modelos de Random Forest performaram melhor que modelos estatísticos como Passeio Aleatório e Autoregressivo. Em outro estudo, Panigrahi e Behera [2017] desenvolveram um modelo híbrido de ETS com Redes Neurais que demonstrou resultados competitivos quando comparado com modelos estatísticos clássicos, como ARIMA e ETS.

Nesse cenário, o framework PolieDRO, proposto por Gutierrez et al. [2024], apresenta uma maneira alternativa de estimar modelos de regressão sem depender da escolha de hiperparâmetros, como em modelos de aprendizado de máquina tradicionais. Essa abordagem parte da equivalência, demonstrada por Blanchet et al. [2019], entre modelos clássicos de aprendizado de máquina e problemas de otimização robusta a distribuições (DRO) definidos via a distância de Wasserstein — uma métrica que quantifica a diferença entre distribuições de probabilidade. Este framework foi então implementado na linguagem Julia em Couto [2025] e utilizado neste experimento.

Dentro desse contexto, este trabalho propõe investigar a viabilidade de aplicar o *framework* PolieDRO à estimação de modelos de séries temporais. A proposta surge do interesse em explorar como essa abordagem, que conecta aprendizado de máquina e otimização robusta a distribuições (DRO), pode oferecer uma alternativa consistente e livre de hiperparâmetros para a modelagem de processos autoregressivos simulados. Ao testar o desempenho do PolieDRO nesse cenário, busca-se avaliar seu potencial como ferramenta complementar ou substitutiva aos métodos tradicionais de regressão aplicados a séries temporais.

2. Fundamentação Teórica

Nesta seção, será apresentado o *framework* de regressão e classificação PolieDRO, bem como brevemente discutida a fundamentação teórica de otimização robusta e robusta a distribuições, que são conceitos necessários para a formulação do PolieDRO.

2.1. O framework PolieDRO

Como descrito em Couto [2025], o framework PolieDRO, estabelecido no trabalho de Gutierrez et al. [2024], parte de um trabalho feito por Blanchet et al. [2019] que mostra a equivalência entre modelos de aprendizado de máquina tradicionais e uma formulação como problema de otimização robusta a distribuições (DRO, distributionally robust optimization) baseada na métrica da distância de Wasserstein, uma forma de medir a distância entre duas distribuições de probabilidade.

Unindo esta formulação com os trabalhos de Fernandes et al. [2016] e Wiesemann et al. [2014], foi construído um método para formular estes problemas de DRO puramente a partir das observações de um conjunto de dados, sem os hiperparâmetros normalmente encontrados nas formulações tradicionais de modelos de aprendizado de máquina.

2.1.1. Otimização robusta a distribuições

O campo de otimização ou programação matemática estuda métodos para escolher um elemento ótimo obedecendo determinados critérios, como por exemplo encontrar um argumento dentro de um domínio que maximiza ou minimiza uma função-objetivo.

Na otimização estocástica, estuda-se problemas em que a função-objetivo ou o domínio analisado dependem de variáveis aleatórias de distribuição conhecida. É possível, por exemplo, discretizar o problema em cenários com diferentes probabilidades de ocorrência, de maneira a criar uma aproximação da variável aleatória.

Um modelo que pode ser descrito como problema de otimização estocástica é a regressão linear. Pode-se interpretar que temos N observações de uma variável aleatória $(x, y) = W$. Desejamos encontrar parâmetros β que minimizam uma função de erro $h(W, \beta)$. Utilizando como métrica o erro quadrático médio, por exemplo, tem-se $h(W, \beta) = (y - (\beta_1 \cdot x + \beta_0))^2$.

$$\min_{\beta \in \mathcal{B}} \mathbb{E}[h(\tilde{W}, \beta)] \quad (1)$$

Na otimização robusta (Bertsimas et al. [2011]), por outro lado, considera-se que os parâmetros incertos podem assumir um conjunto de valores, com um foco em particular no pior caso possível, de maneira que a solução satisfaça a todos os cenários dados como possíveis. A otimização robusta a distribuições Goh e Sim [2010] é uma generalização de ambos os métodos. Considera-se que os valores incertos seguem uma distribuição presente dentro de um conjunto de ambiguidade de distribuições.

2.1.2. Formulação

Na publicação de Blanchet et al. [2019], é demonstrado que diversos métodos de aprendizado de máquina podem ser representados por um problema DRO da forma abaixo:

$$\begin{aligned} \min_{\beta \in \mathcal{B}} \quad & \max_{P \geq 0} \mathbb{E}_P[h(W, \beta)] \\ \text{s.t.} \quad & \int dP = 1 \\ & D(P, \hat{P}_N) \leq \lambda \end{aligned} \quad (2)$$

Neste problema, o objetivo é encontrar os parâmetros β que minimizam uma função de perda $h(W, \beta)$ para uma variável aleatória W de distribuição P . Aqui, a distribuição P é desconhecida,

mas ela está restrita a distribuições com uma distância da distribuição nominal dos dados $\hat{P}_N$ menor que λ , sendo utilizada a distância de Wasserstein. Este é chamado de conjunto de ambiguidade de P . Os autores demonstram que, para diferentes funções de perda, este problema é equivalente a métodos tradicionais de ML. Por exemplo, é possível definir o SVM através deste modelo ao utilizar a perda de articulação, ou a regressão logística com a perda logística, ou a regressão LASSO com o erro médio quadrático. O framework PolieDRO adapta este problema utilizando um conjunto de ambiguidade para P construído de maneira que não depende de um hiperparâmetro λ , proveniente do trabalho de Wiesemann et al. [2014]:

$$\min_{\beta \in \mathcal{B}} \max_{P \in \mathcal{P}} \mathbb{E}_P[h(W, \beta)] \quad (3)$$

Onde $\mathcal{P} = \{P \in \mathcal{M}_+(\mathbb{R}^m) | P(W \in \mathcal{C}_i) \in [\underline{p}_i, \overline{p}_i], \forall i \in \mathcal{F}\}$. Ou seja, a distribuição P novamente não é uma distribuição conhecida, mas sabe-se que a área de P contida em cada um dos conjuntos convexos $\mathcal{C}_i$ está no intervalo de confiança $[\underline{p}_i, \overline{p}_i]$, e estes conjuntos se contém de maneira $\mathcal{C}_{\mathcal{I}} \subset \mathcal{C}_{\mathcal{I}-1} \subset \dots \subset \mathcal{C}_1 \subset \mathcal{C}_0$. Como o conjunto $\mathcal{C}_0$ contém a totalidade da distribuição, tem-se que $\underline{p}_0 = \overline{p}_0 = 1$.

O PolieDRO propõe-se a modelar estes conjuntos $\mathcal{C}_i$ autocontidos através de envoltórias convexas dos dados. A envoltória convexa de todas as observações $\mathcal{C}_0$ é calculada, em seguida a envoltória $\mathcal{C}_1$ dos pontos que ficaram no interior de $\mathcal{C}_0$, recursivamente até termos a estrutura completa. Desta maneira, escolhendo diferentes funções de perda é possível otimizar parâmetros para modelos de classificação e regressão. Neste projeto, foi utilizado como função de perda o erro quadrático, resultando em um modelo com paralelos à regressão LASSO.

2.2. Modelos Autoregressivos

Os modelos autoregressivos de ordem p , denotados por $\text{AR}(p)$, constituem uma classe fundamental de modelos univariados de séries temporais. Segundo a metodologia de Box e Jenkins Box et al. [2015], esses modelos assumem que o valor atual de uma série temporal pode ser expresso como uma combinação linear de seus próprios valores passados e de um termo de erro aleatório. A formulação geral do modelo $\text{AR}(p)$ é dada por:

$$y_t = c + \sum_{j=1}^p \phi_j y_{t-j} + \varepsilon_t, \quad (4)$$

em que y_t representa o valor da série no tempo t , c é o intercepto do modelo, ϕ_j são os coeficientes de defasagem que quantificam a influência dos valores passados da série, e ε_t é o termo de erro aleatório, assumido como ruído branco com distribuição $\varepsilon_t \sim \text{i.i.d. } \mathcal{N}(0, \sigma^2)$.

Para que o processo seja considerado *estacionário*, é necessário que todas as raízes do polinômio característico associado ao modelo estejam fora do círculo unitário no plano complexo (ver Hyndman et al. [2008]). Esse polinômio é definido como:

$$\Phi(z) = 1 - \phi_1 z - \phi_2 z^2 - \dots - \phi_p z^p. \quad (5)$$

A estimação dos parâmetros c , ϕ_j e σ^2 pode ser realizada por diferentes métodos. A abordagem mais tradicional, conforme a metodologia de Box e Jenkins, consiste na estimação via máxima verossimilhança, assumindo normalidade dos erros. Alternativamente, podem ser utilizados métodos de mínimos quadrados ordinários (MQO) quando os dados estão organizados de forma adequada. Os modelos $\text{AR}(p)$ são particularmente úteis para séries estacionárias e servem como blocos fundamentais em modelos mais complexos, como os modelos ARMA, ARIMA e SARIMA. Devido à sua simplicidade e interpretabilidade, modelos autoregressivos são amplamente

utilizados como benchmarks para avaliar o desempenho de novas abordagens de previsão, como redes neurais, métodos de *ensemble* ou, como neste trabalho, estimadores baseados em otimização robusta a distribuições.

2.3. PolieDRO para problemas de séries temporais

Sendo o PolieDRO um *framework* para modelos de regressão, é necessário adaptar as séries temporais para um formato tabular adequado para modelos de regressão, com uma matriz de variáveis explicativas X e um vetor coluna de *targets* y . Uma vez que o objetivo é avaliar o PolieDRO como estimador de modelos autoregressivos, as únicas variáveis explicativas serão os próprios valores defasados da série temporal. A Figura 1 exibe esse procedimento para um modelo $AR(2)$. A coluna de 1 é criada para a estimação do intercepto c do modelo, enquanto as demais são exatamente uma para cada defasagem do $AR(p)$.

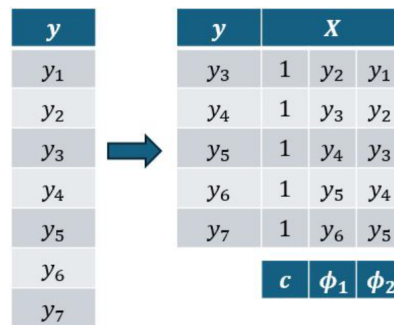


Figura 1: Transformação de série temporal para problemas de regressão.

3. Metodologia

A metodologia utilizada neste trabalho pode ser dividida em 4 etapas. Primeiramente, na seção 3.1 está descrita a metodologia desenvolvida para simular séries temporais a partir de processos auto-regressivos. Em seguida, na seção 3.2 apresentamos o algoritmo de imputação de *outliers* utilizado. Adiante, a seção 2.3 introduz as etapas necessárias para utilizar um modelo de regressão, como o PolieDRO, para estimar modelos de séries temporais. Por fim, a seção 3.3 descreve brevemente os experimentos que foram realizados para avaliar a qualidade do *framework* PolieDRO para problemas de séries temporais.

3.1. Simulações de séries temporais

A primeira etapa da metodologia consiste em definir como serão geradas as séries temporais utilizadas para avaliar a capacidade do PolieDRO em estimar modelos desse tipo. Optou-se por utilizar séries simuladas a partir de processos autorregressivos de ordem p , ou seja, processos $AR(p)$, em razão de sua simplicidade estrutural e da ampla compatibilidade com abordagens baseadas em regressão (ver Seção 2.3). Além de permitirem controle rigoroso sobre os parâmetros e propriedades estatísticas das séries, os processos $AR(p)$ fornecem um ambiente claro e bem definido para testar o desempenho do *framework* proposto. A dinâmica desses processos é formalizada na Equação 4.

A partir da definição do processo gerador, é necessário definir um algoritmo para a escolha de c , ϕ_j , e σ_ϵ . O Algoritmo 1 descreve, em pseudocódigo, o processo proposto.

Os valores de $N = 10$ séries simuladas para cada valor de $p \in \{1, 2, 3, 4\}$ foram escolhidos para gerar experimentos limitados e possíveis de avaliação individual de cada série simulada. Além disso, o valor de $T = 300$ foi escolhido para permitir que experimentos com séries também

Algorithm 1 Simulação de Séries Temporais a partir de um Processo AR(p)

```

1:  $N = 10$ ;  $T = 300$ ;  $P = \{1, 2, 3, 4\}$ .
2: for  $p \in P$  do
3:   Inicialize  $i = 0$ .
4:   while  $i < N$  do
5:      $\phi_1, \phi_2, \dots, \phi_p \sim \mathcal{U}(-2, 2)$ .
6:      $c \sim \mathcal{U}(5, 15)$ .
7:      $\sigma_\epsilon \sim \mathcal{U}(0.5, 2)$ .
8:     Verifique a estacionariedade do processo:
9:     if check_stationarity( $\phi_1, \phi_2, \dots, \phi_p$ ) then
10:       $y_1, y_2, \dots, y_p \sim \mathcal{N}(0, 1)$ .
11:      for  $t = p + 1$  até  $T$  do
12:        Simule  $y_t$  usando:  $y_t = c + \sum_{j=1}^p \phi_j y_{t-j} + \epsilon_t$ ,  $\epsilon_t \sim \mathcal{N}(0, \sigma_\epsilon^2)$ 
13:      end for
14:      Armazene a série simulada.
15:      Atualize  $i = i + 1$ .
16:     else
17:       Re-sorteie  $\phi_1, \phi_2, \dots, \phi_p$ .
18:     end if
19:   end while
20: end for

```

de tamanhos 100 e 200 obtidas a partir de intervalos da série simulada, o que será descrito na seção 3.3.

Note que foi utilizada uma distribuição uniforme para sortear valores dos coeficientes, intercepto e desvio padrão. Essa escolha se dá pelo fato dessa distribuição ter suporte finito, dentro do intervalo estipulado, o que fornece maior controle para os experimentos. Dito isso, o intervalo de $[5, 15]$ para c é de escolha arbitrária, garantindo apenas que as séries nunca terão intercepto igual a 0. Por outro lado, a escolha do intervalo $[0.5, 2]$ para o desvio padrão foi feita para evitar séries muito ruidosas e gerar experimentos mais controlados.

Por fim, uma peça fundamental do Algoritmo 1 é a verificação de estacionariedade. Para tal, foi implementado um segundo algoritmo, Algoritmo 2, baseado na metodologia proposta por Box&Jenkins (Box et al. [2015]) e apresentado em Hyndman [2018]. Esse algoritmo verifica se os coeficientes sorteados geram um processo estocástico estacionário por meio das raízes de seu polinômio característico. Como argumenta Hyndman [2018], se todas as raízes estiverem fora do círculo unitário, o processo é estacionário. Caso contrário, não. A Figura 2 exibe uma das 10 séries simuladas para cada ordem do processo AR(p) a partir do Algoritmo 1. Note que, visualmente, a estacionariedade do processo parece garantida.

Algorithm 2 Checagem de Estacionariedade

```

1: function CHECK_STATIONARITY( $\phi_1, \phi_2, \dots, \phi_p$ )
2:   Calcule as raízes  $z_i$  do polinômio característico:  $1 - \phi_1 z - \phi_2 z^2 - \dots - \phi_p z^p = 0$ 
3:   if  $|z_i| > 1 \quad \forall i \in 1, \dots, p$  then return True ▷ O processo é estacionário.
4:   else return False ▷ O processo não é estacionário.
5:   end if
6: end function

```

3.2. Imputação de outliers

Uma vez que um dos objetivos desse projeto é avaliar o PolieDRO para estimação de modelos $AR(p)$ com séries com *outliers*, torna-se necessário desenvolver uma metodologia para imputação de valores anômalos nas séries temporais simuladas a partir do Algoritmo 1. Para tal, foi desenvolvido o Algoritmo 3, que gera outliers aditivos nas séries simuladas originais.

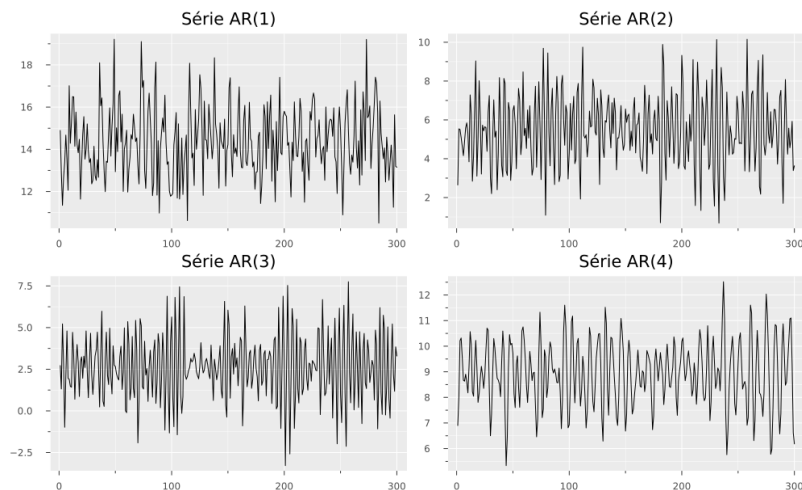


Figura 2: Séries simuladas a partir de processos AR(1) a AR(4).

A primeira escolha a ser feita é a quantidade de *outliers* a ser incluída nas séries temporais. O valor de 5%, após testes empíricos com outros valores, foi escolhido por garantir uma quantidade suficiente de valores anômalos na série mas sem alterar significativamente a dinâmica temporal da série simulada, o que seria o caso com valores maiores. Dito isso, foi decidido espalhar uniformemente os *outliers* em cada terço da série temporal para garantir que, ao selecionar intervalos de tamanho 100 e 200 (ver seção 3.3), todos os intervalos possuam a mesma quantidade de *outliers*.

Sorteados os instantes nos quais serão imputados os *outliers*, torna-se necessário definir como esses valores serão imputados. Também de forma empírica, foi decidida pela seguinte metodologia: aumentar o valor original y_t proporcionalmente ao maior pico ou vale da série. Para isso, K é o fator multiplicativo, $\max_{j \in \mathcal{I}} |\Delta_j|$ é o maior pico ou vale da série, definido como a maior distância para a média, e $\text{sign}(\Delta_i)$ é para garantir que valores abaixo da média sejam "aumentados para baixo" e valores acima da média sejam "aumentados para cima".

Algorithm 3 Imputação de Outliers em uma Série Temporal

- 1: **Entrada:** Série temporal y de tamanho $T = 300$.
 - 2: **Saída:** Série temporal y com outliers imputados.
 - 3: Calcular a quantidade de outliers por intervalo: $q = \frac{5\% \cdot T}{3} = 5$
 - 4: Sortear, sem reposição, os índices: $\mathcal{I} = \mathcal{I}_1 \cup \mathcal{I}_2 \cup \mathcal{I}_3$, com $\mathcal{I}_j \subset \{100(j-1)+1, \dots, 100j\}$, $|\mathcal{I}_j| = 5$, para $j = 1, 2, 3$.
 - 5: Calcular a diferença $\Delta_i = \bar{y} - y_i$, $\forall i \in \mathcal{I}$, onde $\bar{y} = \frac{1}{T} \sum_{t=1}^T y_t$
 - 6: Sortear do fator multiplicativo $K \sim \mathcal{U}(2, 3)$:
 - 7: Atualizar os valores da série: $y_i \leftarrow y_i + K \cdot \text{sign}(\Delta_i) \cdot \max_{j \in \mathcal{I}} |\Delta_j|$, $\forall i \in \mathcal{I}$.
 - 8: **Retorne:** y .
-

A escolha de K como fator multiplicativo é crucial para a série com *outliers* imputados e, para isso, foram testados diferentes valores de referência. A Figura 3 exibe a adição de *outliers* para 4 valores distintos de K . A partir desses testes, foi decidido sortear K de maneira uniforme entre 2 e 3, uma vez que esses valores geram *outliers* grandes o suficiente para visualmente destacarem-se da série original, mas não grandes o bastante para tornarem-se muito díspares dos valores originais das séries simuladas. Com isso, são, no total, 40 séries temporais de tamanho 300 sem *outliers* e 40 séries temporais de mesmo comprimento com *outliers*.

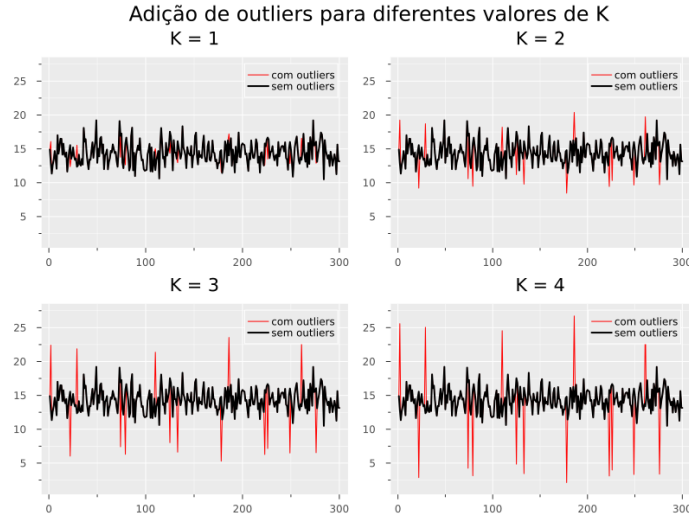


Figura 3: Adição de outliers para diferentes valores de K .

3.3. Experimentos realizados

O objetivo deste projeto é avaliar o uso do PolieDRO para estimar modelo de séries temporais autoregressivas. Para isso, foram simuladas séries de tamanho $T = 300$ a partir de processos $AR(p)$, $p \in \{1, 2, 3, 4\}$. Foram realizados experimentos com duas variações. A primeira delas é a adição de outliers, como descrita na seção 3.2. Essa variação tem o objetivo de verificar se a robustez do PolieDRO garante a correta estimação dos parâmetros dos processos $AR(p)$ na presença de séries temporais "infectadas" com observações anômalas. Uma segunda variação é o tamanho da série temporal. A partir das séries de tamanho 300, foram criadas outras duas séries, de tamanho 100, com as primeiras 100 observações, e de tamanho 200, com as primeiras 200 observações das séries de tamanho 300. Com isso, temos, para cada ordem p , 30 séries temporais com e 30 sem outliers.

A fim de trazer uma referência de comparação para o PolieDRO, também foram estimados modelos $AR(p)$ utilizando o pacote *statsmodels* de Skipper e Josef [2010] da linguagem de programação *Python*. Esse pacote implementa a classe *statsmodels.tsa.arima.model.ARIMA* que foi utilizada fixando as ordens dos polinômios diferente do $AR(p)$ em 0. Internamente, o pacote estima os modelos ARIMA via formulação de espaço de estados e maximização de verossimilhança. Mais informações sobre a estimação dos modelos podem ser encontradas na documentação do Statsmodels.

Com isso, os modelos serão comparados da seguinte maneira. Primeiramente, será avaliada a estimação dos coeficientes dos processos geradores, por meio de duas métricas, Viés e MSE, descritas nas equações abaixo. O viés almeja avaliar se algum dos modelos consistentemente sobre ou subestima os coeficientes reais. O MSE, por sua vez, é utilizado para avaliar a dispersão dos valores estimados. Para ambas as métricas, deseja-se valores próximos de 0.

$$Viés = \frac{1}{N \cdot p} \sum_{i=1}^p \left(\sum_{j=1}^N (\hat{\phi}_{i,j} - \phi_{i,j}) \right) \quad MSE = \frac{1}{N \cdot p} \sum_{i=1}^p \left(\sum_{j=1}^N (\hat{\phi}_{i,j} - \phi_{i,j})^2 \right) \quad (6)$$

Em seguida, serão comparados os valores estimados *in-sample* e os valores calculados com a previsão *out-of-sample* de 10 períodos à frente. Para isso, as 10 últimas observações de todas as séries foram separadas como período de teste e não foram utilizadas na estimação dos modelos. Importante ressaltar que a previsão fora da amostra do PolieDRO foi construída de forma iterativa

para os 10 passos à frente, utilizando, por exemplo, a previsão $\hat{y}_{t+1|T}$ para prever o valor $\hat{y}_{t+2|T}$. Foi escolhido o RMSE como métrica de avaliação. Sua definição é dada por $\sqrt{\frac{1}{T} \sum_{t=1}^T (y_t - \hat{y}_t)^2}$.

4. Resultados

Seguindo a ordem descrita na seção 3.3, as Tabelas 1 e 2 exibem os vieses calculados para as séries sem e com *outliers*. Como desejado, as métricas estão todas próximas de zero, o que indica que tanto o PolieDRO quanto o AR estimam com acurácia e baixo viés os coeficientes dos processos geradores. É importante frisar que os resultados do PolieDRO são comparáveis com os resultados do modelo *benchmark*. Além disso, para as séries sem *outliers*, não é perceptível nenhuma mudança nos resultados com o aumento da ordem dos modelos, enquanto que o crescimento da série parece indicar uma melhor estimação em termos de viés. Ao adicionar *outliers*, as métricas pioram, o que é compreensível dada a perturbação inserida na série temporal. Embora os resultados dos dois modelos ainda sejam próximos, percebe-se que o PolieDRO apresenta uma maior piora relativa em comparação ao AR. Por fim, interessante notar que, com *outliers*, o viés indica sobre-estimação dos coeficientes em todos os casos.

Viés dos estimadores sem outliers							
AR				PolieDRO			
p/T	100	200	300	p/T	100	200	300
1	0.0237	0.0181	0.0070	1	0.0680	0.0362	0.0088
2	-0.0488	-0.0138	-0.0144	2	-0.0122	0.0337	0.0151
3	0.0241	-0.0050	0.0034	3	0.0895	-0.0375	-0.0051
4	-0.0455	-0.0072	0.0102	4	0.0275	0.0399	0.0528

Tabela 1: Viés dos estimadores sem outliers.

Viés dos estimadores com outliers							
AR				PolieDRO			
p/T	100	200	300	p	100	200	300
1	0.2256	0.2059	0.2238	1	0.2949	0.3336	0.4275
2	0.3108	0.3439	0.3463	2	0.3321	0.3841	0.4275
3	0.0535	0.0539	0.0453	3	0.0385	0.1241	0.0997
4	0.1567	0.1624	0.1593	4	0.2052	0.1894	0.2303

Tabela 2: Viés dos estimadores com outliers.

Para avaliar a dispersão dos coeficientes estimados, as Tabelas 3 e 4 exibem os MSE calculados nesse experimento. Novamente, ambos os modelos apresentam MSE próximos a zero e o PolieDRO apresenta resultados comparáveis ao modelo AR. Como esperado, os resultados do AR sempre são melhores, dado que, quando bem especificado, o estimador de MQO usado para estimar o modelo AR é BLUE (*best linear unbiased estimator*) (ver Wooldridge [2016]). Novamente, ao adicionar *outliers*, a métrica piora em todos os casos e a piora relativa do PolieDRO é mais acentuada do que a do *benchmark*. Esse fato indica que, embora o PolieDRO seja um modelo de otimização robusta, sua robustez não se mostrou suficiente para garantir a correta estimação dos coeficientes do modelo para as séries deste experimento, quando comparado com o modelo *benchmark*. Em seguida, para avaliar a qualidade dos valores estimados no conjunto de treino, a Tabela 5 exhibe os RMSE médios para cada ordem p do modelo $AR(p)$ e tamanho de série. Como esperado, o modelo AR sempre apresenta métricas melhores no treino, uma vez que ele é BLUE. Casos em que o PolieDRO gera RMSE menor *in sample* são marginais e devido a questões numéricas de estimação, como é o caso da ordem 3 e $T = 100$. Dito isso, o PolieDRO foi capaz de gerar erros comparáveis ao *benchmark*, o que era esperado dados os resultados das Tabelas 1, 2, 3 e 4. Podemos ver, novamente, que a variação do tamanho da série e a ordem do modelo não impactaram no RMSE

MSE dos estimadores sem outliers							
AR				PolieDRO			
p/T	100	200	300	p/T	100	200	300
1	0.0058	0.0024	0.0011	1	0.0199	0.0085	0.0067
2	0.0075	0.0032	0.0032	2	0.0314	0.0093	0.0100
3	0.0069	0.0023	0.0018	3	0.0795	0.0336	0.0158
4	0.0154	0.0041	0.0029	4	0.0218	0.0444	0.0431

Tabela 3: MSE dos estimadores sem outliers.

MSE dos estimadores com outliers							
AR				PolieDRO			
p/T	100	200	300	p/T	100	200	300
1	0.1637	0.1523	0.1584	1	0.4613	0.5864	0.5849
2	0.4689	0.4494	0.4567	2	0.7353	0.8723	0.9485
3	0.3075	0.2914	0.2818	3	0.6294	0.5582	0.5753
4	0.5009	0.4752	0.4639	4	0.7018	0.8196	0.7092

Tabela 4: MSE dos estimadores com outliers.

de treino. Além disso, a adição de *outliers* impactou negativamente essa métrica e o aumento do RMSE do PolieDRO é maior do que o aumento do AR. Por fim, a Tabela 6 exibe os RMSE da

RMSE Treino				
T = 100				
Ordem	AR	PolieDRO	AR (out)	PolieDRO (out)
1	1.406	1.417	2.404	2.638
2	1.359	1.420	3.415	4.313
3	1.294	1.259	4.304	5.070
4	1.283	1.295	3.547	5.006
T = 200				
Ordem	AR	PolieDRO	AR (out)	PolieDRO (out)
1	1.412	1.418	2.453	2.734
2	1.379	1.414	3.410	4.137
3	1.265	1.240	4.138	4.945
4	1.252	1.269	3.474	4.480
T = 300				
Ordem	AR	PolieDRO	AR (out)	PolieDRO (out)
1	1.407	1.413	2.470	2.684
2	1.378	1.406	3.421	4.027
3	1.264	1.240	4.058	4.953
4	1.248	1.298	3.452	4.058

Tabela 5: RMSE no conjunto de treino para diferentes valores de T .

previsão 10 passos fora da amostra. Além disso, ela indica a porcentagem de experimentos em que o PolieDRO obteve métricas melhores que o modelo AR na previsão.

Podemos ver que, nos casos sem outlier, os erros médios obtidos são similares para todos os tamanhos de série T e ordens p , com o PolieDRO inclusive conseguindo resultados melhores na maioria dos testes feitos para séries de tamanho 200 e 300. Houve casos em que o PolieDRO foi superior na maioria das séries temporais mesmo com um RMSE médio pior (ver $T = 300$ e $p = 1$). Assim como observado anteriormente, a adição de *outliers* impactou negativamente mais fortemente o PolieDRO do que o modelo AR.

5. Conclusão

Os experimentos realizados mostram que o PolieDRO gera resultados comparáveis ao modelo $AR(p)$. As médias dos valores *in-sample* também indicam que o PolieDRO consegue man-

RMSE Teste						
T = 100						
Ordem	AR	PolieDRO	PolieDRO melhor	AR (out)	PolieDRO (out)	PolieDRO melhor (out)
1	1.740	1.843	20%	2.760	2.925	50%
2	2.435	2.581	40%	3.324	3.794	20%
3	2.108	2.670	20%	3.251	3.885	20%
4	1.434	1.610	30%	2.271	4.117	0%
T = 200						
Ordem	AR	PolieDRO	PolieDRO melhor	AR (out)	PolieDRO (out)	PolieDRO melhor (out)
1	1.543	1.543	50%	2.453	2.522	10%
2	1.980	1.916	60%	3.771	4.002	20%
3	2.127	2.061	70%	4.946	5.140	10%
4	1.802	2.014	40%	3.030	3.907	30%
T = 300						
Ordem	AR	PolieDRO	PolieDRO melhor	AR (out)	PolieDRO (out)	PolieDRO melhor (out)
1	1.673	1.681	60%	1.860	1.961	40%
2	2.276	2.067	70%	2.920	3.359	20%
3	2.409	2.453	50%	5.920	6.215	10%
4	1.870	2.191	40%	3.463	3.552	50%

Tabela 6: RMSE no conjunto de teste para diferentes valores de T .

ter um desempenho comparável ao $AR(p)$, ainda que este tenha, em geral, resultados ligeiramente melhores. Além disso, na previsão *out of sample*, o PolieDRO mostrou-se competitivo, superando o modelo *benchmark* em uma quantidade significativa de casos.

Apesar da característica robusta do PolieDRO, os resultados indicam que ele não performa melhor do que o $AR(p)$ em séries que contêm *outliers*. Embora os dois modelos tenham sofrido impacto com a presença dessas perturbações, o PolieDRO apresentou uma piora relativa maior em comparação ao AR. Uma possibilidade é a robustez do modelo torná-lo ainda mais sensível a outliers ao buscar minimizar os erros de maior magnitude. Com a presença de outliers, os piores erros estarão concentrados nestes, e em vez de ignorados estes são ainda mais acentuados quando comparados a modelos comuns. Isso demonstraria uma importância de realizar um pré-tratamento de remoção de outliers antes de utilizar modelos robustos de regressão como o PolieDRO.

No contexto das previsões out-of-sample, o desempenho do PolieDRO indica que ele se destaca ao obter melhores resultados em uma porcentagem razoável de experimentos, principalmente em séries temporais sem *outliers* e com tamanhos maiores. Contudo, ao analisar as métricas de erro médio, observa-se que o $AR(p)$ apresenta valores inferiores mesmo em alguns destes casos. Esse resultado sugere que existem cenários em que o PolieDRO erra de maneira mais significativa, enquanto o modelo $AR(p)$ se mantém mais consistente com erros próximos ao PolieDRO.

Um próximo passo natural desta pesquisa é estender a aplicação do PolieDRO para séries com dinâmicas mais complexas, como não estacionariedade e sazonalidade. Pesquisas já estão sendo conduzidas para introduzir sazonalidade e diferenciação no modelo, com testes comparativos em séries reais. Além disso, seria interessante estimar o PolieDRO com outras funções de perda adequadas a modelos de regressão, como o MAE, ou ainda aplicar técnicas de *feature engineering* para criar variáveis explicativas que auxiliem a previsão da série temporal modelada.

Referências

- Abbasimehr, H., Shabani, M., e Yousefi, M. (2020). An optimized model using lstm network for demand forecasting. *Computers & industrial engineering*, 143:106435.
- Bertsimas, D., Brown, D. B., e Caramanis, C. (2011). Theory and applications of robust optimization. *SIAM Rev.*, 53(3):464–501.

- Blanchet, J., Kang, Y., e Murthy, K. (2019). Robust wasserstein profile inference and applications to machine learning. *Journal of Applied Probability*, 56.
- Box, G. E., Jenkins, G. M., Reinsel, G. C., e Ljung, G. M. (2015). *Time series analysis: forecasting and control*. John Wiley & Sons.
- Couto, J. (2025). Poliedro.jl: Biblioteca para ciência de dados com otimização robusta a distribuições. <https://www.maxwell.vrac.puc-rio.br/projetosEspeciais/TFCs/consultas/conteudo.php?strSecao=resultado&nrSeq=69816@1>. Trabalho de Fim de Curso, PUC-Rio. Orientador: Alexandre Street de Aguiar, Coorientador: Davi Michel Valladão.
- Fernandes, B., Street, A., Valladão, D., e Fernandes, C. (2016). An adaptive robust portfolio optimization model with loss constraints based on data-driven polyhedral uncertainty sets. *European Journal of Operational Research*, 255(3):961–970.
- Goh, J. e Sim, M. (2010). Distributionally robust optimization and its tractable approximations. *Oper. Res.*, 58(4-Part-1):902–917.
- Gutierrez, T., Valladão, D., e Pagnoncelli, B. (2024). Poliedro: a novel classification and regression framework with non-parametric data-driven regularization. *Machine Learning*, p. 1–40.
- Harvey, A. C. (1990). Forecasting, structural time series models and the kalman filter.
- Harvey, A. C. e Todd, P. H. (1983). Forecasting economic time series with structural and box-jenkins models: A case study. *Journal of Business & Economic Statistics*, 1(4):299–307.
- Hyndman, R. (2018). *Forecasting: principles and practice*. OTexts.
- Hyndman, R., Koehler, A. B., Ord, J. K., e Snyder, R. D. (2008). *Forecasting with exponential smoothing: the state space approach*. Springer Science & Business Media.
- Lippi, M., Bertini, M., e Frasconi, P. (2013). Short-term traffic flow forecasting: An experimental comparison of time-series analysis and supervised learning. *IEEE Transactions on Intelligent Transportation Systems*, 14(2):871–882.
- Medeiros, M. C., Vasconcelos, G. F., Veiga, Á., e Zilberman, E. (2021). Forecasting inflation in a data-rich environment: the benefits of machine learning methods. *Journal of Business & Economic Statistics*, 39(1):98–119.
- Panigrahi, S. e Behera, H. S. (2017). A hybrid ets–ann model for time series forecasting. *Engineering applications of artificial intelligence*, 66:49–59.
- Skipper, S. e Josef, P. (2010). statsmodels: Econometric and statistical modeling with python. *9th Python in Science Conference*.
- Wiesemann, W., Kuhn, D., e Sim, M. (2014). Distributionally robust convex optimization. *Operations research*, 62(6):1358–1376.
- Wooldridge, J. M. (2016). *Introductory Econometrics: A Modern Approach 6rd ed.* Cengage learning.