



UNIVERSIDADE FEDERAL DE PERNAMBUCO  
CENTRO DE INFORMÁTICA  
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

Adriano Felipe Cabral Filho

**Abordagem Baseada em Ensembles para Previsão de Séries Temporais de Crime**

Recife

2025

Adriano Felipe Cabral Filho

## **Abordagem Baseada em Ensembles para Previsão de Séries Temporais de Crime**

Monografia Apresentada ao Centro de Informática (CIn) da Universidade Federal de Pernambuco (UFPE), como requisito parcial para conclusão do Curso de Ciência da Computação, orientada pelo professor Paulo Salgado Gomes de Mattos Neto.

**Área de Concentração:** Texto Texto

**Orientador:** Paulo Salgado Gomes de Mattos Neto

Recife

2025

Ficha de identificação da obra elaborada pelo autor,  
através do programa de geração automática do SIB/UFPE

Cabral Filho, Adriano Felipe.

Abordagem Baseada em Ensembles para Previsão de Séries Temporais de Crime / Adriano Felipe Cabral Filho. - Recife, 2025.

41 p : il., tab.

Orientador(a): Paulo Salgado Gomes de Mattos Neto

Trabalho de Conclusão de Curso (Graduação) - Universidade Federal de Pernambuco, Centro de Informática, Ciências da Computação - Bacharelado, 2025.

Inclui referências.

1. Séries Temporais. 2. Ensembles. 3. Aprendizagem de Máquina. 4. Crimes. I. Mattos Neto, Paulo Salgado Gomes de. (Orientação). II. Título.

000 CDD (22.ed.)

Adriano Felipe Cabral Filho

**Abordagem Baseada em Ensembles para Previsão de Séries Temporais de Crime**

Trabalho de Conclusão de Curso  
apresentado ao Curso de Graduação em  
Ciência Da Computação da Universidade Federal de  
Pernambuco, como requisito parcial para  
obtenção do título de bacharel em Ciência Da Computação.

Aprovado em: 21/07/2025

**BANCA EXAMINADORA**

---

Prof. Dr. Paulo Salgado Gomes de Mattos Neto (Orientador)

Universidade Federal de Pernambuco

---

Prof. Dr. Fernando Maciano de Paula Neto (Examinador Interno)

Universidade Federal de Pernambuco

Dedico este trabalho à minha família, pelo apoio incondicional ao longo destes anos. Aos meus amigos, pelos momentos que compartilhamos juntos. Aos professores pelas valiosas contribuições ao meu aprendizado. A todos que, de alguma forma, contribuíram para a realização deste trabalho, minha sincera gratidão.

## AGRADECIMENTOS

Desejo agradecer ao meu orientador, Paulo Salgado Gomes de Mattos Neto, pela sua paciência e orientação no decorrer deste trabalho. À Ruam e Jair, que trabalharam comigo no artigo que produzimos. À todos os meus professores ao longo da minha jornada, do infantil até a faculdade, que formaram o estudante que sou hoje.

À minha madrastra Juliana, que foi minha segunda orientadora não oficial, me apoiando e ajudando em todas as etapas do processo. À minha mãe Angela e ao meu pai Adriano, que me ensinaram a ter interesse por ler, estudar, descobrir. Sem eles eu nem sei onde eu estaria.

Aos amigos que fiz pelo caminho, todos fizeram parte da minha jornada e vou guardá-los para sempre comigo. Em especial, Iris, Tarcísio, João Pedro, Malu, Netto, Hendell, que foram meus confidentes e me ajudaram direta ou indiretamente no meu trabalho.

Foi uma tarefa tortuosa. Por muitas vezes eu pensei em desistir, mas estas pessoas me fizeram continuar seguindo em frente, e desejo tudo do bom e do melhor para elas do fundo do meu coração.

## RESUMO

A criminalidade é um problema recorrente em diversas regiões do mundo, com destaque para o Brasil, que figura entre os países mais violentos. Altos índices de violência impactam negativamente a qualidade de vida da população e geram prejuízos econômicos significativos. Nesse contexto, o uso de tecnologias de previsão pode auxiliar os órgãos de segurança pública na adoção de estratégias mais eficientes e preventivas. Este trabalho propõe uma análise comparativa do desempenho de seis algoritmos na previsão de séries temporais de ocorrências criminais: ARIMA, Support Vector Regression (SVR), Multilayer Perceptron (MLP), Long Short-Term Memory (LSTM), Convolutional Neural Networks (CNN) e Neural Basis Expansion Analysis for Interpretable Time Series Forecasting (N-BEATS). Foram utilizadas séries temporais de três localidades com diferentes contextos: Pernambuco, Chicago e Los Angeles. Além da avaliação individual dos modelos, suas previsões foram combinadas através de média e mediana simples, e *ensemble stacking*. As métricas de avaliação incluíram o *Root Mean Square Error* (RMSE) e o *Mean Absolute Error* (MAE), permitindo uma comparação objetiva entre os algoritmos. Os resultados indicaram que a combinação de média simples obteve as melhores pontuações em ambas as métricas de avaliação nas 3 bases de dados utilizadas, indicando Além disso, as abordagens de combinação contribuíram para maior robustez e acurácia nas previsões. Este estudo destaca o potencial do uso de técnicas de aprendizado de máquina na área da segurança pública, contribuindo para uma atuação mais estratégica e preventiva. Como trabalho futuro, sugere-se a inclusão de variáveis externas e a ampliação do escopo geográfico da análise.

**Palavras-chaves:** segurança pública; séries temporais; aprendizado de máquina; previsão; *ensemble*.

## ABSTRACT

Crime is a recurring issue in many regions around the world, with Brazil standing out as one of the most violent countries. High levels of violence negatively affect the population's quality of life and generate significant economic losses. In this context, forecasting technologies can support public security agencies in adopting more efficient and preventive strategies. This study presents a comparative analysis of the performance of six algorithms for forecasting crime time series: ARIMA, Support Vector Regression (SVR), Multilayer Perceptron (MLP), Long Short-Term Memory (LSTM), Convolutional Neural Networks (CNN), and Neural Basis Expansion Analysis for Interpretable Time Series Forecasting (N-BEATS). Time series from three locations with different contexts were used: Pernambuco, Chicago, and Los Angeles. In addition to evaluating the models individually, their forecasts were combined through simple mean, simple median, and ensemble stacking approaches. Evaluation metrics included Root Mean Square Error (RMSE) and Mean Absolute Percentage Error (MAPE), allowing for an objective comparison between algorithms. The results indicated that the simple mean ensemble achieved the best scores in both metrics across all three datasets. Furthermore, the combination strategies contributed to increased robustness and accuracy in the forecasts. This study highlights the potential of machine learning techniques in the field of public safety, contributing to more strategic and preventive actions. As future work, the inclusion of external variables and the expansion of the geographical scope of the analysis are suggested.

**Keywords:** public security; time series; machine learning; forecasting; ensemble.

## LISTA DE FIGURAS

Figura 1 – Vendas mensais de habitações nos Estados Unidos em milhões de unidades.	15
Figura 2 – À esquerda, visualização da soma mínima dos quadrados. À direita, visualização do $\varepsilon$ -tubo de um svr.	22
Figura 3 – Arquitetura de um MLP.	23
Figura 4 – Estrutura de um Perceptron.	23
Figura 5 – Camada oculta de uma RNN "enrolada" e "desenrolada".	24
Figura 6 – Célula LSTM.	25
Figura 7 – Arquitetura de uma CNN.	26
Figura 8 – Operação de convolução $3 \times 3$ : multiplicação elemento a elemento seguida de soma, gerando o mapa de características.	26
Figura 9 – Exemplo de aplicação da técnica de <i>Max Pooling</i> com filtro $2 \times 2$ , demonstrando a redução dimensional da matriz original.	26
Figura 10 – Arquitetura de um N-BEATS.	28
Figura 11 – Modelo Proposto. O método é dividido em duas fases: treinamento e teste. Na fase de treinamento, os modelos individuais são treinados com os conjuntos de treino e validação, gerando previsões usadas como entrada para o treinamento dos meta-modelos. Na fase de teste, os modelos treinados realizam previsões sobre o conjunto de teste, originando saídas individuais, por ensembles stack e por média e mediana simples.	29
Figura 12 – Série Temporal de Pernambuco, dividida em treino, validação e teste.	31
Figura 13 – Série Temporal de Chicago, dividida em treino, validação e teste.	31
Figura 14 – Série Temporal de Los Angeles, dividida em treino, validação e teste.	32

## LISTA DE TABELAS

Tabela 1 – Parâmetros do GridSearch para os modelos SVR, MLP, LSTM, N-BEATS e CNN.	34
Tabela 2 – Medidas de desempenho para os modelos individuais e ensembles não treináveis e treináveis. Valores em negrito indicam o melhor valor para cada tipo de abordagem. Os valores em negrito e sublinhado indicam o melhor desempenho entre os modelos avaliados.	36
Tabela 3 – Medidas de desempenho para os modelos individuais e ensembles não treináveis e treináveis no trabalho de (PASTOR et al., 2023). Valores em negrito indicam o melhor valor para cada tipo de abordagem. Os valores em negrito e sublinhado indicam o melhor desempenho entre todos os modelos avaliados.	37
Tabela 4 – Ranqueamento dos modelos baseado nas métricas de avaliação. Valores em negrito mostram o melhor rank para o tipo de abordagem, e valores sublinhados mostram os melhores ranks gerais para cada métrica.	38

## LISTA DE SÍMBOLOS

$\gamma$  Letra grega Gama

$\in$  Pertence

$\delta$  Delta

$\theta$  Teta

$\sigma$  Sigma

$\mu$  Mi

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b>	<b>12</b>
<b>2</b>	<b>CONCEITOS BÁSICOS</b>	<b>14</b>
2.1	SÉRIES TEMPORAIS	14
2.2	ENSEMBLES	15
<b>3</b>	<b>TRABALHOS RELACIONADOS</b>	<b>17</b>
<b>4</b>	<b>MODELOS AVALIADOS</b>	<b>20</b>
4.1	ARIMA	20
4.2	SVR	21
4.3	MLP	22
4.4	LSTM	23
4.5	CNN	25
4.6	N-BEATS	27
<b>5</b>	<b>MODELO PROPOSTO</b>	<b>29</b>
5.1	FASE DE TREINAMENTO	29
5.2	FASE DE TESTE	30
<b>6</b>	<b>EXPERIMENTOS E ANÁLISE</b>	<b>31</b>
6.1	BANCO DE DADOS	31
6.2	MÉTRICAS DE DESEMPENHO UTILIZADAS	32
6.3	PROTOCOLO EXPERIMENTAL	32
6.4	RESULTADOS	34
<b>7</b>	<b>CONCLUSÕES E TRABALHOS FUTUROS</b>	<b>39</b>
	<b>REFERÊNCIAS</b>	<b>40</b>

## 1 INTRODUÇÃO

A segurança pública é uma preocupação para países no mundo inteiro. Taxas altas de criminalidade afetam não só o bem estar e a integridade física e mental da população, como geram elevados custos para os cofres públicos e causam prejuízo econômico em regiões mais impactadas pelo crime. Em especial, o Brasil se mostra um dos países mais violentos do mundo. Segundo o (Fórum Brasileiro de Segurança Pública, 2022), o número de homicídios em território nacional alcançou em 2020 a incrível marca de 47.722 casos, sendo responsáveis por 20,5% das ocorrências deste tipo de delito no planeta inteiro, mesmo possuindo apenas 2,7% da população mundial. Além do óbvio impacto na preservação da vida, o alto volume de crimes afeta setores da economia, como o turismo.

A Confederação Nacional do Comércio de Bens, Serviços e Turismo (CNC) estima que em 2017 a queda de visitas turísticas devido ao aumento da violência gerou um prejuízo de cerca de 657 milhões de reais em 2017 no estado do Rio de Janeiro. Desta forma, torna-se crescente a necessidade do poder público buscar novas formas de lidar com a violência, e a tecnologia pode se formar uma ferramenta poderosa no diagnóstico dos registros criminais, possibilitando uma abordagem preventiva dos agentes de segurança pública.

Neste contexto, o presente estudo propõe uma análise comparativa do desempenho dos modelos Autoregressive Integrated Moving Average (ARIMA) (CHEN; YUAN; SHU, 2008), Support Vector Machine (SVR) (ALWEE; SHAMSUDDIN; SALLEHUDDIN, 2013), Multilayer Perceptron (MLP) (YAMUNATHANGAM, 2024), Long-Short Term Memory (LSTM) (MUTHAMIZHARASAN; PONNUSAMY, 2022), Convolutional Neural Networks (CNN) (STEC; KLABJAN, 2018) e Neural Basis Expansion Analysis (N-BEATS) (BORIS et al., 2020) na previsão de 3 séries temporais, representando dados de ocorrências criminais em Pernambuco, Chicago e Los Angeles. Além disso, foram utilizadas táticas de ensemble para combinar estes modelos em busca de melhores resultados.

Este trabalho foi dividido em 7 capítulos. No capítulo 2 serão apresentados conceitos gerais sobre o que são séries temporais e técnicas de ensemble. O capítulo 3 faz uma revisão da literatura, buscando trabalhos relacionados ao tema de previsão de crimes, mostrando suas qualidades e carências. O capítulo 4 introduz fundamentação teórica dos modelos avaliados, descrevendo conceitos básicos de suas arquiteturas. O capítulo 5 descreve o modelo proposto por este trabalho. Em seguida, o capítulo 6 trata primeiramente da natureza do banco de

dados utilizado, e depois detalha o experimento realizado por este estudo, analisando os resultados. Por fim, o capítulo 7 trata da conclusão das comparações propostas e comenta sobre aprimoramentos que futuros trabalhos podem realizar.

## 2 CONCEITOS BÁSICOS

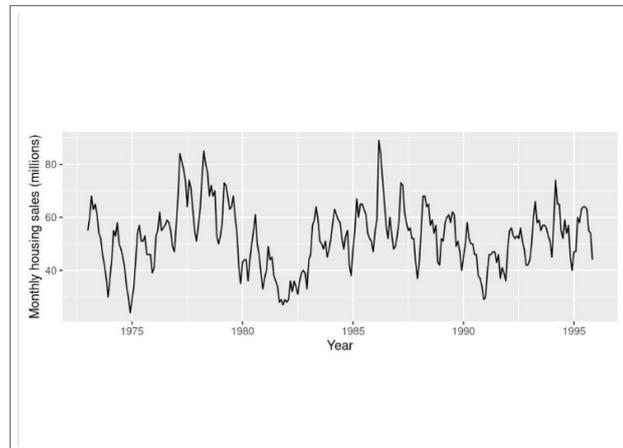
### 2.1 SÉRIES TEMPORAIS

Uma série temporal é uma coleção de observações feitas sequencialmente através do tempo (CHATFIELD, 2000). Exemplos podem ser encontrados nas mais diversas áreas, como as variações diárias dos preços de uma ação no mercado financeiro, registros do índice pluviométrico de uma determinada região, ou até mesmo o número de batimentos cardíacos por minuto de um paciente num hospital. Naturalmente, a análise e investigação desses dados pode fornecer informações valiosas para auxiliar na tomada de decisão. A variação de uma série temporal pode ser decomposta em 4 componentes. São esses a tendência, sazonalidade, mudanças cíclicas e flutuações irregulares.

De forma vaga, a tendência pode ser definida como uma mudança a longo prazo no nível médio (CHATFIELD, 2003). Mais especificamente, ela se faz presente quando uma série apresenta um crescimento ou declínio estável ao longo do tempo, mas não existe uma definição matemática completamente satisfatória (CHATFIELD, 2000). O problema deste conceito é que a noção do que exatamente é o “longo prazo” é difícil de ser formulada com precisão. A Figura 1 apresenta um gráfico do número de casas vendidas nos Estados Unidos em milhões de unidades, de 1973 até 1995. É possível ver que a série apresenta um comportamento cíclico que varia entre 6 e 10 anos. Um recorte aproximado entre os anos 1982 e 1987 mostraria uma suposta tendência de alta, mas ao observar o gráfico inteiro é possível ver que na verdade essa variação faz parte de um comportamento cíclico. Este é formado por oscilações sem um período fixo, mas que podem ser previstas até certo ponto (CHATFIELD, 2003).

A sazonalidade é um tipo de variação geralmente anual que aparece em várias séries, sejam elas medidas em semanas, meses ou trimestres, quando padrões de comportamento similares são observados em certas épocas do ano (CHATFIELD, 2000). Por exemplo, o gráfico do número de hóspedes por mês que visitam um hotel a beira mar no Brasil costuma apresentar um salto nos meses de dezembro e janeiro, pois as pessoas costumam tirar férias nessa época, quando chega o verão e as festas de fim de ano. Já as flutuações irregulares são os outros tipos de variações que não se encaixam nas categorias anteriores, que podem ser aleatórias ou não.

Figura 1 – Vendas mensais de habitações nos Estados Unidos em milhões de unidades.



Fonte: <https://robjhyndman.com/hyndsight/cyclicts/>

## 2.2 ENSEMBLES

Aprendizado por *ensemble* é um termo genérico para métodos que combinam múltiplos indutores para realizar uma decisão, tipicamente em tarefas de aprendizado supervisionado (SAGI; ROKACH, 2018). A ideia principal é que limitações ou falhas de um modelo individual podem ser compensados por outros modelos, melhorando a capacidade preditiva de uma forma geral. Além disso, métodos de ensemble oferecem vantagens ao encarar desafios normalmente encontrados por algoritmos de aprendizagem de máquina, como prevenir *overfitting*, evitar máximos e mínimos locais, lidar com classes desbalanceadas, entre outros (SAGI; ROKACH, 2018).

O trabalho de (SAGI; ROKACH, 2018) mostra que *ensembles* poder ser classificados de 3 formas: quanto à sua abordagem de treinamento, quanto à forma de combinação de resultados, e quanto à dependência entre os modelos na sua construção. Com base nestas categorias, podem-se destacar algumas técnicas populares de *ensemble* para melhor ilustração. *Boosting* é uma abordagem bastante comum, onde um indutor fraco, ou seja, um modelo que tem uma performance ligeiramente melhor do que palpites aleatórios, é aprimorado para produzir resultados mais eficazes. Suponha por exemplo que um indutor fraco  $I$  é utilizado para produzir previsões  $P$  sobre uma série temporal  $S$ . Um novo indutor então é utilizado para modelar o vetor de resíduos  $E$  obtido a partir de  $P$ , e seus resultados são somados às previsões anteriores para ajustar a o palpite do indutor fraco. Este processo é repetido  $N$  vezes até que a performance preditiva do modelo final seja satisfatória. Sendo assim, *boosting* pode ser categorizada da seguinte forma: Sua abordagem de treinamento envolve manipulação do alvo

---

de aprendizado, sua combinação de resultados é feita através da soma, e como a construção dos modelos é sequencial, eles são dependentes uns dos outros.

Outra técnica bastante popular é a de *bagging* (SAGI; ROKACH, 2018), onde os modelos a serem combinados recebem como entrada amostras de um conjunto de dados  $D$ . Para que estas amostras atinjam um tamanho igual ao de  $D$ , elas são criadas com reposição, ou seja, alguns elementos podem ocorrer repetidas vezes, enquanto outros nem são incluídos. Cada modelo é treinado separadamente, e o resultado final é uma média dos produtos de cada um, para problemas de regressão, ou é decidido por através de votação em problemas de classificação. *Bagging* pode ser categorizado como uma técnica que utiliza manipulação dos dados de entrada como abordagem de treinamento, a combinação dos resultados é feita através de média ou votação, podendo ser ponderadas ou não, e seus modelos são independentes uns dos outros.

Por fim, outra técnica também muito utilizada é a de *stacking* (SAGI; ROKACH, 2018), que inclusive será utilizada neste trabalho. Nela, os modelos individuais são treinados a partir da mesma base de dados, e produzem cada um suas próprias previsões. Então, um novo indutor chamado de meta-modelo é usado para modelar o mesmo problema apresentado aos anteriores, porém ao invés de utilizar o conjunto de dados original para treinamento, ele recebe as previsões feitas pelos modelos individuais.. Desse modo, o meta-modelo tenta mesclar o que há de melhor no que foi produzido pelos modelos individuais. (SAGI; ROKACH, 2018) Categoriza a abordagem de treinamento do *stacking* como "manipulação do aprendizado, o método de combinação são os meta-modelos, e os modelos individuais são construídos de forma independente uns dos outros".

### 3 TRABALHOS RELACIONADOS

Já houve diversas tentativas de desenvolvimento de modelos que pudessem antecipar a ocorrência de crimes, tomando os mais diversos tipos penais e regiões geográficas, aplicando as mais diversas estratégias de previsão. Esta seção tem como objetivo realizar uma breve revisão da literatura, mostrando como este problema foi atacado por diversos pesquisadores diferentes.

Os estudos relacionados à previsão de crimes ganharam força a partir da década de 1980, com o investimento do governo dos Estados Unidos na área (PASTOR et al., 2023). Dentre os trabalhos desenvolvidos na época, podemos citar por exemplo o estudo elaborado por (COHEN, 1981). Após este “ponta-pé” inicial, diversas técnicas mais elaboradas foram exploradas ao longo dos anos.

Em 2008, (CHEN; YUAN; SHU, 2008) testaram a eficiência do modelo ARIMA na previsão a curto prazo de crimes contra a propriedade em uma cidade da China. Foram alimentados ao algoritmo dados obtidos deste tipo de delito ao longo de 50 semanas, e sua tarefa era realizar a previsão uma semana adiante. Seu desempenho foi testado contra 2 tipos de suavização exponencial, SES e HES, e obteve resultados significativamente superiores, e, segundo os autores, se provou bastante útil para as forças policiais locais na tomada de decisão em situações de emergência. No entanto, o trabalho carece de táticas mais sofisticadas de previsão, como redes recorrentes, aprendizagem profunda, *ensemble*, modelos híbridos, entre outros.

Em 2017, (ZHUANG et al., 2017) desenvolveram um modelo de *Spatio-Temporal Neural Network* (Rede Neural Espaço-Temporal), que é uma rede capaz de lidar com conjuntos de dados que variam no tempo e no espaço. Neste trabalho, o algoritmo proposto foi testado em um banco de dados correspondente aos crimes relatados num período de 5 anos na cidade de Portland, nos Estados Unidos, e sua localização no espaço (latitude e longitude). O mapa da cidade foi então dividido em várias células, e o problema definido como a previsão do nível de crime em cada uma das células, para identificação de “pontos quentes” de crime na região. O modelo proposto então superou algoritmos de aprendizagem de máquina tradicionais, como Random Forest, MLP, regressão logística, entre outros nas métricas de performance, mas o estudo não empregou táticas de *ensemble* ou outras formas de combinação de modelos.

Em 2018, (STEC; KLABJAN, 2018) realizaram um estudo sobre a capacidade preditiva de redes neurais na previsão de crimes. Os algoritmos de redes neurais MLP, RNN, CNN, e um

modelo que combina os dois últimos foram treinados em conjuntos de dados sobre crimes realizados em Chicago e Portland, nos Estados Unidos. Os resultados mostraram que a combinação de CNN e RNN obteve melhor performance em todos os conjuntos de dados, indicando que a estratégia de mesclar múltiplos modelos pode ser efetiva. No entanto, o estudo não utiliza táticas de *ensemble* ou outras formas de explorar mais a fundo esta abordagem, além de não se valer de modelos lineares, como ARIMA ou ETS.

Em 2022, (CRUZ-NAJERA et al., 2022) conduziram um experimento referente a previsão de séries temporais curtas sobre crimes, na faixa de 14 até 21 observações. Foram utilizados 8 métodos de previsão diferentes, sendo 4 destes algoritmos abordagens mais simples, como modelos lineares ou um MLP, e os outros 4 técnicas de ensemble altamente sofisticadas (estado da arte) usadas nas competições M4 e M5 (Makridakis). Os dados consistem em frequências de crimes em 3 áreas urbanas distintas, representando ocorrências de roubos em lojas, roubos de carro, furtos e arrombamentos. Ao final dos testes, o artigo conclui que as melhores abordagens considerando as métricas de desempenho foram 2 modelos individuais (ARIMA e médias móveis simples) e 2 modelos ensemble (*Jaganathan* e *Hybrid*), com o ARIMA obtendo o melhor resultado na métrica *Symmetric Mean Absolute Percentage Error* (sMAPE) e as médias móveis simples apresentando o melhor ranque médio do teste de Friedman, superando os algoritmos mais complexos por uma leve margem. Os resultados obtidos mostram uma certa equivalência entre modelos individuais e técnicas de última geração, contudo, os autores ressaltam que esta análise só é válida para séries temporais curtas, o que destaca a necessidade do desenvolvimento de novos estudos, utilizando séries maiores, para alcançar resultados mais generalizáveis.

Em 2023, (IVANYUK, 2024) elaborou um sistema baseado em aprendizagem para prever o número de crimes financeiros digitais na Rússia. A autora se valeu de 5 métodos de previsão, regressão linear, ETS, *Wavelet*, uma rede neural de fator único (uma variável de entrada) e uma rede neural multi fator, com o *Wavelet* obtendo o melhor RMSE. Em seguida, os modelos individuais foram combinados através de táticas de *ensemble* ingênuo (média simples), *bagging* Bayesiano, e todas as 3 abordagens concluíram que o número de crimes financeiros digitais na Rússia irá aumentar nos próximos anos. No entanto, o artigo não chega a discutir sobre o desempenho dos modelos *ensemble* nas métricas de desempenho, então não há avaliação de qual da eficiência de previsão destas técnicas em relação aos modelos individuais.

Apesar de técnicas de combinação de modelos como *ensembles* já serem amplamente utilizados para previsão de séries temporais, com base no apanhado de trabalhos observados, percebe-se que ainda há espaço para evolução nos estudos destas técnicas na área de previsão

de crimes. Aprimoramentos neste campo podem ser capazes de fornecer ao Estado ferramentas para o desenvolvimento de políticas públicas sociais e de segurança direcionados a evitar, reduzir ou minimizar eventuais danos que as práticas delituosas causam à sociedade. Nesse sentido, este trabalho pretende proporcionar visões alternativas para melhoria das estratégias que poderão ser utilizadas para previsão de crimes utilizando *ensembles*.

## 4 MODELOS AVALIADOS

Este capítulo trata dos modelos individuais selecionados para o estudo.

### 4.1 ARIMA

Auto Regressive Moving Average (Médias Móveis Auto-Regressivas - ARMA) é um modelo estatístico para previsão de séries temporais dividido em 2 componentes: a auto regressão (AR) e o média móvel (MA). Modelos auto regressivos partem da noção de que os valores de uma série temporal, podem ser explicados através de uma função de observações anteriores (SHUMWAY; STOFFER, 2025) da seguinte forma: seja  $y$  uma série temporal estacionária, então um valor qualquer de  $y$  no tempo  $t$  pode ser definido como:

$$y_t = \phi_1 y_{t-1} + \phi_2 y_{t-2} + \dots + \phi_p y_{t-p} + e_t, \quad (4.1)$$

em que  $\phi_i$  são os coeficientes de auto regressão,  $y_{t-i}$  é um valor da série  $i$  intervalos de tempo no passado, e  $e_t$  representa um erro no momento  $t$ . De forma similar, modelos de médias móveis definem que os valores de uma série temporal podem ser entendidos como uma combinação linear dos erros passados (SHUMWAY; STOFFER, 2025), de acordo com a equação 2:

$$y_t = \theta_1 e_{t-1} + \theta_2 e_{t-2} + \dots + \theta_q e_{t-q} + e_t, \quad (4.2)$$

em que  $\theta_i$  são os coeficientes de média móvel,  $e_{t-i}$  são os erros  $i$  intervalos de tempo no passado, e  $e_t$  é o erro no momento  $t$ . Estes dois modelos são então combinados para formar o modelo ARMA.

Contudo, esta classe de modelos parte do pressuposto de que a série temporal em questão é estacionária, para que essas relações sejam válidas. Em busca de contornar este problema, foi criado o modelo *Auto Regressive Integrated Moving Average* (Médias Móveis Integradas Auto-Regressivas - ARIMA), que é uma evolução dos modelos ARMA, adicionando um novo componente de integração (I), consistindo na diferenciação da série.

Em outras palavras, seja uma série temporal  $Y = [y_0, y_1, \dots, y_t]$ , então  $Y$  é formada da seguinte forma:

$$\forall t, y'_t = y_t - y_{t-1}. \quad (4.3)$$

A razão para isso é que o processo de diferenciação gera uma série estacionária (SHUMWAY; STOFFER, 2025), solucionando a necessidade dos modelos ARMA.

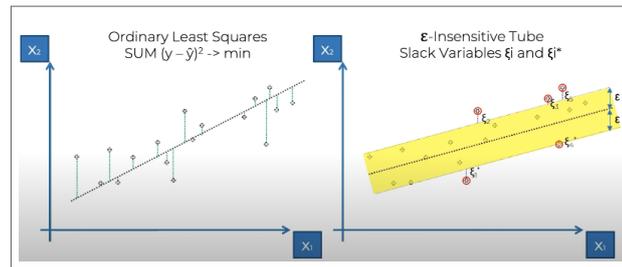
Combinações de modelos estatísticos e de *machine learning* (aprendizagem de máquina) costumam obter resultados melhores para previsão de séries temporais, como visto na quarta iteração da competição *Makridakis*. Dito isto, o modelo ARIMA foi escolhido para fazer parte deste estudo por ser utilizado na literatura para previsão de séries temporais, seja de forma individual (KHAN; ALGHULAIKHA, 2020) ou em combinação com modelos de *machine learning* (ZHANG, 2003).

## 4.2 SVR

*Support Vector Regression* (Regressão por Vetores de Suporte - SVR) é um modelo de aprendizagem de máquina que deriva do *Support Vector Machine* (Máquina de Vetores de Suporte - SVM). Modelos clássicos de regressão buscam encontrar uma função de previsão que generalize bem o conjunto de dados, de forma a minimizar o erro entre os valores previstos e os valores reais. O modelo SVR oferece uma abordagem diferente: é traçado um tubo ao redor da função, chamado de  $\epsilon$ -tubo, conforme a Figura 2. Os erros de previsão para os pontos do gráfico dentro do tubo são ignorados, ou seja, consideram-se apenas os erros para os valores de fora do tubo. Esta margem de tolerância faz com que pequenos erros não influenciem na otimização da função, fornecendo um modelo mais simples e com maior capacidade de generalização.

O SVR é um modelo amplamente utilizado na área de previsão de séries temporais, muitas vezes combinado com outros modelos, seja pela elaboração de híbridos (GUO et al., 2021) ou através de *ensembles* (QIU et al., 2014).

Figura 2 – À esquerda, visualização da soma mínima dos quadrados. À direita, visualização do  $\epsilon$ -tubo de um svr.



Fonte: <https://youtu.be/kPw1IGUAoY8>

### 4.3 MLP

O *Multilayer Perceptron* (Perceptron Multicamadas - MLP) é um algoritmo amplamente conhecido e utilizado na área de aprendizagem de máquina. É o tipo de rede neural mais comum, apresentando fundamentos que são aplicados em vários outros modelos. As Figuras 3 e 4 mostram a estrutura de camadas de um MLP, e sua unidade mais básica, o *perceptron*, também chamado de neurônio. Este realiza uma soma ponderada dos seus valores de entrada com um peso de viés, chamado de  $w_0$  na Figura 4. O resultado passa por uma transformação através de uma função não linear, como a *sigmoid* ou a *ReLU*, e o valor obtido é enviado para a camada seguinte.

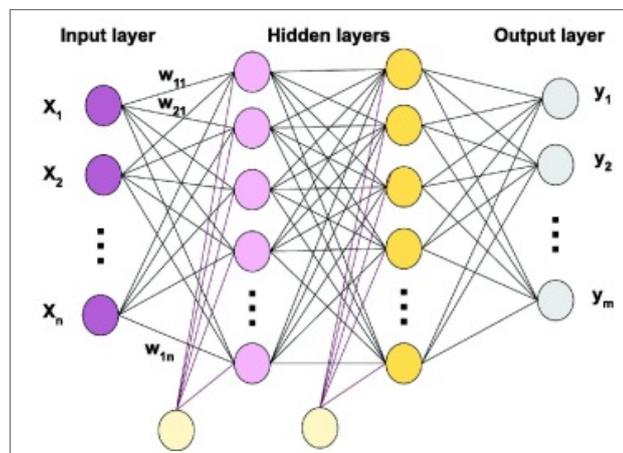
Existem três tipos de camadas: entrada, oculta e saída. A primeira não realiza nenhum tipo de processamento, apenas recebe os dados e os envia adiante. Camadas ocultas são chamadas assim porque não têm contato com o ambiente externo, processando as saídas da camada anterior e passando seu produto para a próxima. Por fim, a camada de saída produz a estimativa final da rede neural. Redes deste tipo são chamadas de *feedforward*, ou seja, a informação é passada apenas em uma direção. Modelos onde os dados fluem em duas direções são chamados de recorrentes, como o LSTM, discutido mais à frente neste capítulo. Outra característica de redes *feedforward* é que suas camadas são totalmente conectadas: cada neurônio envia seu valor de saída para todos os neurônios da camada seguinte.

O treinamento do MLP consiste de várias iterações. No caso deste trabalho, os dados são passados uma vez pela rede, e seu resultado é a previsão de quantos crimes serão realizados no próximo mês. Uma função de perda então calcula o erro baseado no valor esperado e no valor previsto, e este erro é usado para ajustar os pesos da rede neural, através de um algoritmo chamado *backpropagation* (retropropagação). Após vários ciclos, chamados de épocas,

o algoritmo finalmente para, com os pesos devidamente ajustados.

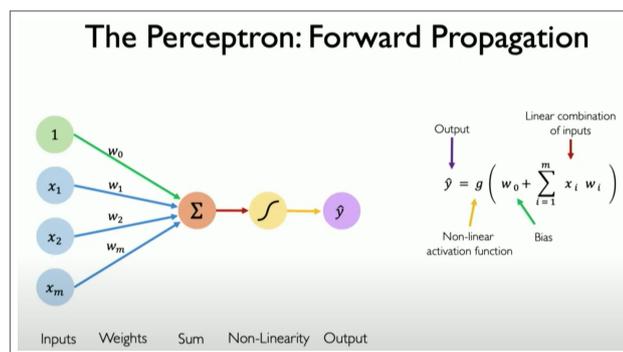
O MLP não foi escolhido pelo poder da sua capacidade de previsão: existem hoje alternativas que o superam em diversas áreas. Porém, é um algoritmo que apresenta diversos fundamentos da área de redes neurais, tornando-se interessante para o estudo neste trabalho. Também se espera que, aumentando o número de modelos disponíveis no nosso conjunto, a performance dos modelos de *ensemble* consiga melhores resultados.

Figura 3 – Arquitetura de um MLP.



Fonte: <https://www.sciencedirect.com/topics/computer-science/multilayer-perceptron>

Figura 4 – Estrutura de um Perceptron.



Fonte: <https://aiml.com/what-is-a-perceptron/>

#### 4.4 LSTM

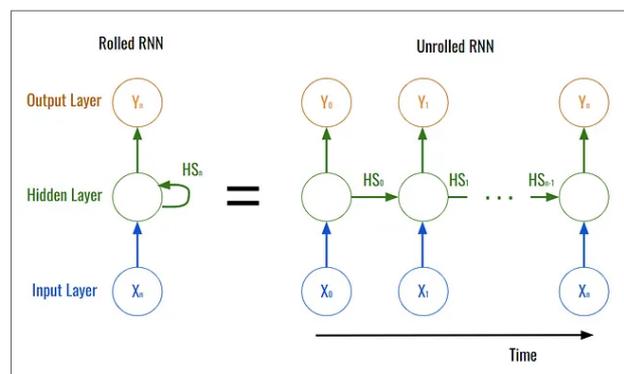
Ao contrário das redes neurais *feedforward*, redes neurais recorrentes possuem fluxo de informação fluindo em duas direções diferentes. A Figura 5 mostra uma representação simplificada do funcionamento dessas redes. É possível ver que a camada oculta tem duas saídas: uma para a camada de saída, e outra retornando para si própria, permitindo que, em uma

determinada iteração, a rede se lembra do valor obtido na iteração anterior. Esta abordagem faz das redes recorrentes modelos ideais para se trabalhar com dados sequenciais, como séries temporais.

Porém, esta técnica apresenta um problema conhecido como o gradiente em explosão/-desvanecimento. Ao lado direito da Figura 5, é possível visualizar o funcionamento de uma rede neural recorrente. A cada novo valor de entrada, o produto da camada oculta vindo da iteração anterior é multiplicado por um peso  $w$ . Se o valor de  $w$  for maior que um, após um certo número de iterações, o valor de entrada é amplificado cada vez mais, até o momento em que “explode”, fazendo com que a influência dos valores seja muito maior do que o desejado. Por outro lado, se o valor de  $w$  for menor do que um, a influência dos valores anteriores da sequência se torna cada vez menor, até o ponto em que é quase irrelevante. Este fenômeno prejudica bastante o treinamento da rede, fazendo com que o algoritmo de otimização dos parâmetros dê passos muito longos ou curtos demais.

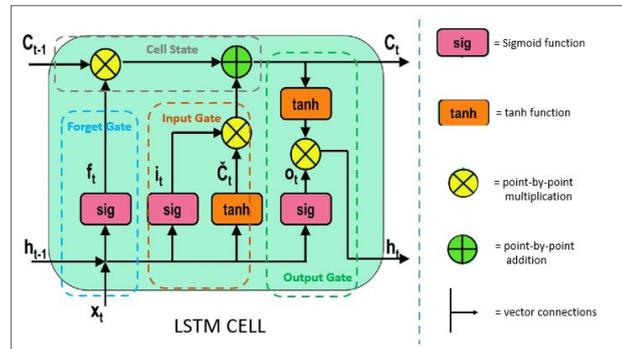
O modelo *Long Short-Term Memory* (Memória de curto-longo prazo - LSTM) foi uma arquitetura de rede neural recorrente que surgiu com a proposta de solucionar o problema do gradiente em explosão/desvanecimento. A Figura 6 apresenta o esquema de uma célula de LSTM, contendo três portas de controle: *forget gate* (porta de esquecimento), *input gate* (porta de entrada) e *output gate* (porta de saída). A primeira define a porcentagem do valor da memória de longo prazo que será preservado, a segunda determina quais informações novas vão ser adicionadas, enquanto a terceira calcula o novo valor da memória de curto prazo. Como a memória de longo prazo é propagada de forma aditiva, o problema do gradiente em explosão/desvanecimento não afeta o modelo. Desta forma, o LSTM foi escolhido para este trabalho pela sua capacidade de processar dados sequenciais, como séries temporais.

Figura 5 – Camada oculta de uma RNN "enrolada" e "desenrolada".



Fonte: <https://medium.com/@CallMeTwitch/building-a-neural-network-zoo-from-scratch-the-recurrent-neural-network-9357b43e113c>

Figura 6 – Célula LSTM.



Fonte: <https://www.pluralsight.com/resources/blog/guides/introduction-to-lstm-units-in-rnn>

#### 4.5 CNN

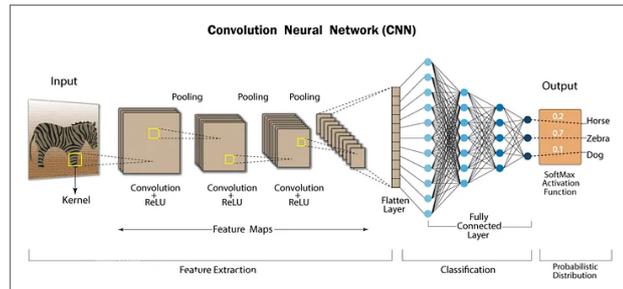
*Convolutional Neural Networks* (Redes Neurais Convolucionais - CNN) são modelos de aprendizagem profunda que, nos últimos anos, se tornaram a principal arquitetura para reconhecimento, classificação e detecção de imagens (RAWAT; WANG, 2017). A arquitetura geral de uma CNN pode ser vista na Figura 7, sendo dividida em camadas de convolução, *pooling* e camadas completamente conectadas.

Camadas de convolução aplicam filtros na imagem, conforme mostrado na Figura 8. Estes filtros são matrizes de pesos, otimizadas através do *backpropagation*, e servem para extrair características da imagem, como bordas, linhas, cantos, etc. O filtro “desliza” pela imagem, e em cada posição é calculado o produto interno entre ambos. O resultado é então armazenado em uma nova matriz, chamada de mapa de características. Este processo ocorre várias vezes, gerando um conjunto de mapas, que então passam por uma função de ativação não linear, como a *ReLU*.

Camadas de *pooling* recebem mapas de características da camada anterior e aplicam uma operação simples para reduzir sua dimensão. A Figura 9 mostra um exemplo de aplicação de *pooling* muito utilizado, chamado de *max pooling*, que consiste em selecionar o maior valor de cada quadrante do mapa de características, deslizando de forma similar aos filtros na camada de convolução. Essa operação ajuda a rede a ignorar certas características que podem não ser tão relevantes na análise da imagem, além de torná-la mais robusta a distorções e alterações na mesma, prevenindo o *overfitting*. Por fim, após uma determinada sequência de camadas de convolução e *pooling*, os valores são passados para uma camada de rede totalmente conectada, similar a um MLP, que realiza os cálculos finais para produzir o resultado da CNN.

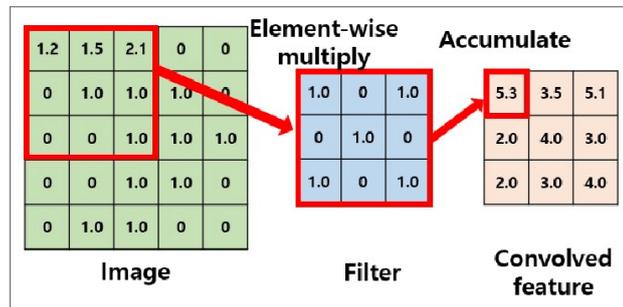
Apesar de serem comumente utilizadas para resolver problemas relacionados a imagens, CNNs são utilizadas pela literatura na previsão de séries temporais, inclusive de forma combinada com outros modelos, como o LSTM, também estudado neste trabalho. (RAWAT; WANG, 2017) (MUTHAMIZHARASAN; PONNUSAMY, 2022)

Figura 7 – Arquitetura de uma CNN.



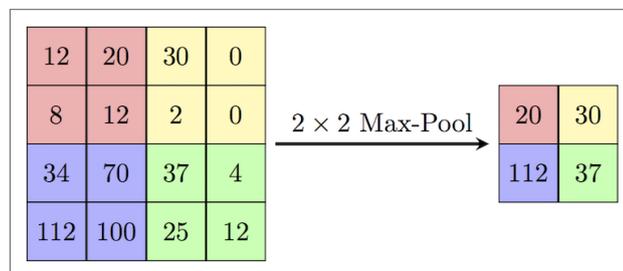
Fonte: <https://nafizshahriar.medium.com/what-is-convolutional-neural-network-cnn-deep-learning-b3921bdd82d5>

Figura 8 – Operação de convolução 3 × 3: multiplicação elemento a elemento seguida de soma, gerando o mapa de características.



Fonte: [https://www.researchgate.net/figure/Convolution-operation\\_fig2\\_355656417](https://www.researchgate.net/figure/Convolution-operation_fig2_355656417)

Figura 9 – Exemplo de aplicação da técnica de Max Pooling com filtro 2 × 2, demonstrando a redução dimensional da matriz original.



Fonte: <https://computersciencewiki.org/index.php/File:MaxpoolSample2.png>

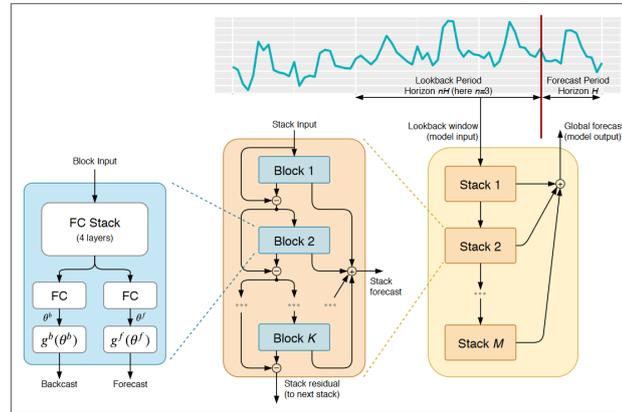
## 4.6 N-BEATS

O modelo *N-BEATS* (*Neural Basis Expansion Analysis for Interpretable Time Series Forecasting*, Análise de Expansão de Bases Neurais para Previsão Interpretável de Séries Temporais) é uma arquitetura de aprendizagem profunda relativamente recente, desenvolvida em 2020, e demonstra resultados superiores a algoritmos de estado da arte nos bancos de dados da *Makridakis*, uma série de competições organizada para comparação e análise das melhores soluções para previsão de séries temporais (BORIS et al., 2020). Sua estrutura é mostrada na Figura 10.

São destacados dois parâmetros utilizados pelo algoritmo: o horizonte de previsão, período no futuro que a rede está tentando prever, e o *lookback*, janela de tempo da série temporal utilizada para prevê-lo. No lado esquerdo da imagem está uma representação da unidade básica do *N-BEATS*, chamada de *bloco*. Este é formado por uma camada de rede totalmente conectada que processa o valor de entrada e produz dois resultados,  $\theta^f$  e  $\theta^b$ , coeficientes utilizados para gerar a previsão do bloco e o chamado *backcast*, que é uma estimativa do valor de entrada, como se a rede estivesse tentando “prever” o passado.

No meio da imagem é possível ver a estrutura de pilha do *N-BEATS*, que agrupa uma sequência de  $K$  blocos. Cada bloco passa para o seguinte o resíduo do seu *backcast*, ou seja, a parte da entrada que ele não conseguiu interpretar, facilitando o processo de previsão para os próximos blocos (BORIS et al., 2020). A previsão final da pilha é uma soma das previsões de cada bloco. À direita da imagem, é mostrada uma estrutura sequencial de pilhas, que funciona de forma semelhante à sequência de blocos descrita anteriormente, onde o resultado final da rede é a soma da previsão de todas as pilhas.

Figura 10 – Arquitetura de um N-BEATS.



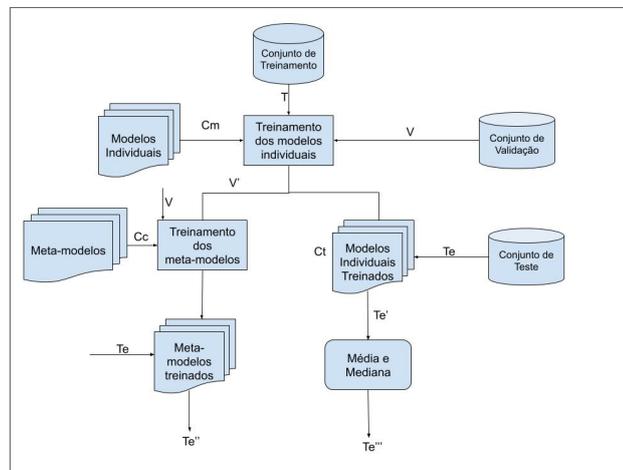
Fonte: <https://towardsdatascience.com/n-beats-the-first-interpretable-deep-learning-model-that-worked-for-time-series-forecasting-06920daadac2/>

## 5 MODELO PROPOSTO

A proposta pode ser dividida em duas fases: treinamento e teste. Na fase de treinamento, a entrada é o conjunto de modelos individuais  $C_m$ , o conjunto de combinadores (meta-modelos)  $C_c$ , e os conjuntos de treinamento de validação  $T$  e  $V$ . Esta fase também produz dois resultados: o conjunto de *ensembles* treinados  $E_t$ , e o conjunto de modelos individuais treinados  $I_t$ , utilizados posteriormente na fase de testes.

Primeiro, foram implementados os modelos individuais, formando o conjunto de modelos  $C_m$ . Cada um destes então recebeu o conjunto de treinamento  $T$  e o conjunto de validação  $V$  e foi treinado separadamente, produzindo o conjunto de modelos treinados  $C_t$  e previsões sobre o conjunto de validação, chamadas de  $V'$ .

Figura 11 – Modelo Proposto. O método é dividido em duas fases: treinamento e teste. Na fase de treinamento, os modelos individuais são treinados com os conjuntos de treino e validação, gerando previsões usadas como entrada para o treinamento dos meta-modelos. Na fase de teste, os modelos treinados realizam previsões sobre o conjunto de teste, originando saídas individuais, por *ensembles stack* e por média e mediana simples.



Fonte: O autor

### 5.1 FASE DE TREINAMENTO

O objetivo da fase de treinamento é a obtenção dos conjuntos de modelos  $E_t$  e  $I_t$ . O segundo é obtido através do treinamento de cada um dos modelos presentes em  $C_m$ , recebendo  $T$  e  $V$  como dados de treinamento e validação. Em seguida, os meta-modelos de  $C_c$  recebem as previsões dos modelos individuais sobre o conjunto de validação, chamadas de  $V'$ , como conjunto de treinamento, e então uma nova rodada de treinamentos é realizada, produzindo o conjunto de meta-modelos treinados.

## 5.2 FASE DE TESTE

Na segunda fase, os modelos de  $I_t$  e  $E_t$  realizam previsões sobre o conjunto de testes  $Te'$  e  $Te''$ , sendo estas as previsões finais dos modelos individuais e *ensembles stack* respectivamente. Em seguida, é realizado o cálculo de média e mediana simples das previsões  $Te'$ , produzindo as previsões dos *ensembles* estatísticos SA (*simple average - media simples*) e SM (*simple median - mediana simples*).

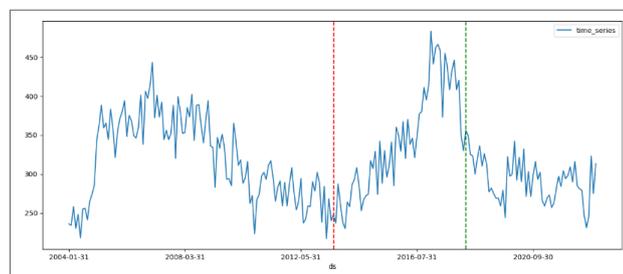
## 6 EXPERIMENTOS E ANÁLISE

Este capítulo detalha os bancos de dados utilizados no estudo, além de descrever o experimento realizado e analisar os resultados obtidos.

### 6.1 BANCO DE DADOS

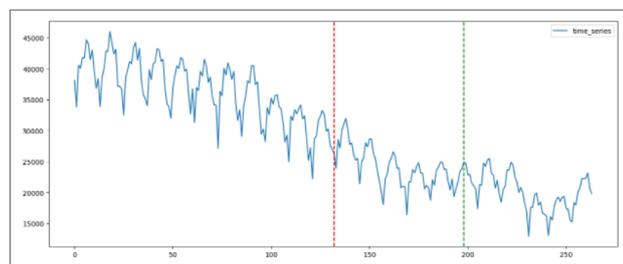
Os dados utilizados são compostos de 3 séries temporais diferentes: crimes em Chicago, crimes em Los Angeles e crimes violentos letais intencionais (CVLI) em Pernambuco. Os CVLI representam crimes de homicídio doloso, latrocínio, feminicídio, lesão corporal seguida de morte e outros crimes que resultaram em mortes. A série em questão apresenta ocorrências de janeiro de 2004 até dezembro de 2022. Os dados de Chigago refletem relatos de crimes (exceto assassinato) que ocorreram na cidade no período de janeiro de 2001 até dezembro de 2022. Já a série de Los Angeles apresenta registros de crimes locais de janeiro de 2010 até dezembro de 2022. As 3 séries podem ser visualizadas nas figuras [12](#), [13](#) e [14](#).

Figura 12 – Série Temporal de Pernambuco, dividida em treino, validação e teste.



Fonte: O autor

Figura 13 – Série Temporal de Chicago, dividida em treino, validação e teste.



Fonte: O autor

---

RAWAT, W.; WANG, Z. Deep convolutional neural networks for image classification: a comprehensive review. *Neural Computation*, v. 29, n. 9, p. 2352–2449, sep 2017.

SAGI, O.; ROKACH, L. Ensemble learning: a survey. *WIREs Data Mining and Knowledge Discovery*, v. 8, n. 4, 2018.

SHUMWAY, R. H.; STOFFER, D. S. *Time Series Analysis and Its Applications: With R Examples*. Fifth. [S.l.]: Springer, 2025. (Springer Texts in Statistics).

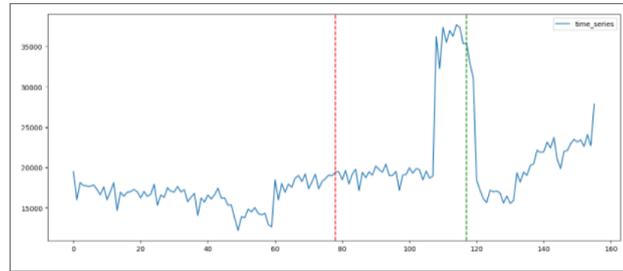
STEC, A.; KLABJAN, D. *Forecasting crime with deep learning*. 2018. Available at: <https://doi.org/10.48550/arXiv.1806.01486>.

YAMUNATHANGAM, P. B. D. Crime prediction and forecasting using mlp & k-means clustering algorithm. *International Journal of Multidisciplinary Research in Science, Engineering and Technology*, v. 7, n. 5, p. 9859–9863, 2024.

ZHANG, G. P. Time series forecasting using a hybrid arima and neural network model. *Neurocomputing*, v. 50, p. 159–175, 2003.

ZHUANG, Y. et al. Crime hot spot forecasting: a recurrent model with spatial and temporal information. In: *IEEE International Conference on Big Knowledge (ICBK)*. Hefei, China: [s.n.], 2017. p. 143–150.

Figura 14 – Série Temporal de Los Angeles, dividida em treino, validação e teste.



Fonte: O autor

## 6.2 MÉTRICAS DE DESEMPENHO UTILIZADAS

Para avaliação do desempenho dos modelos na previsão de séries temporais foram utilizadas as mesmas métricas : *Root Mean Squared Error* (Raiz Quadrada do Erro Médio - RMSE) e *Mean Absolute Percentage Error* (Erro Percentual Absoluto Médio - MAPE).

RMSE é uma função bastante utilizada na literatura por retornar um valor na mesma unidade da variável alvo. MAPE apresenta uma medida intuitiva da acurácia dos modelos, por ser dimensionado em unidades percentuais. Ambas são definidas nas Equações 6.1 e 6.2 respectivamente:

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2}, \quad (6.1)$$

$$\text{MAPE} = \frac{100}{n} \sum_{i=1}^n \left| \frac{y_i - \hat{y}_i}{y_i} \right|. \quad (6.2)$$

## 6.3 PROTOCOLO EXPERIMENTAL

Na realização dos 3 experimentos, os dados foram divididos em 50% para treinamento, 25% para validação e 25% para teste. Então, os valores passaram por uma normalização no intervalo [0.1, 0.9], com base nos valores máximos e mínimos nos conjuntos de treinamento e validação.

Para otimização dos hiperparâmetros dos modelos utilizados no trabalho foram criadas grades para uma busca exaustiva, baseadas no trabalho de (PASTOR et al., 2023), conforme a Tabela 1. Os algoritmos testados anteriormente foram o *Auto Regressive Integrated Moving Average* (ARIMA), *Support Vector Machine* (SVR), *Multilayer Perceptron* (MLP) e *Long*

---

*Short Term Memory* (LSTM). Neste novo experimento, foram adicionados ao conjunto de modelos a CNN e o N-BEATS.

Após o treinamento dos modelos individuais, foram implementados os modelos de *ensemble*. Primeiro, foram criados os *ensembles* estáticos de *Média Simples* (do inglês *Simple Average* — SA) e *Mediana Simples* (do inglês *Simple Median* — SM), que consistem, como os nomes sugerem, no cálculo da média e mediana das previsões no conjunto de testes dos algoritmos individuais.

Por último, foram treinados *ensembles* com base na metodologia de *stack*, que consiste em, após geradas as previsões dos modelos individuais, alimentá-las a um novo modelo, que irá aprender a relação entre as mesmas e os valores-alvo verdadeiros. Neste projeto, os *stack ensembles* foram alimentados com as previsões dos modelos no conjunto de validação, e aqui serão chamados de eSVR, eMLP, eLSTM e eCNN.

Foi utilizada a linguagem de programação *Python* para a implementação dos experimentos. O modelo ARIMA foi implementado utilizando a função *auto.arima* do pacote *pmdarima* para busca dos melhores parâmetros. O modelo SVR foi treinado com o pacote *sklearn*, os modelos MLP, LSTM e CNN através da biblioteca *TensorFlow*, e por fim, o modelo N-BEATS foi implementado através do conjunto de bibliotecas *Nixtla*. Foram realizadas previsões *one-step-ahead* para todos os modelos, ou seja, o horizonte de previsão sempre teve o valor igual a 1.

Tabela 1 – Parâmetros do GridSearch para os modelos SVR, MLP, LSTM, N-BEATS e CNN.

<b>Modelos</b>	<b>Parâmetros</b>	<b>Valores</b>
<b>SVR</b>	Lag	1, 3, 6, 9, <b>15</b> , 18
	Kernel	Linear, Radial, Sigmoid
	Gamma	1, 0.1, 0.01, 0.001
	Cost	0.1, 1, 10, 100, 1000, 10000
	Epsilon	0.1, 0.01, 0.001
<b>MLP</b>	Lag	1, 3, 6, 9, <b>15</b> , 18
	Função de Ativação	ReLU, Sigmoid, Tangente
	Unidades na camada oculta	2, 3, 10, <b>15</b> , 20
	Otimizador	Adam
<b>LSTM</b>	Função de Ativação	ReLU, Sigmoid, Tangente
	Unidades na camada oculta	5, 10, 100, 500
	Otimizador	Adam
	Lag	1, 3, 6, 9, <b>15</b> , 18
<b>N-BEATS</b>	Número de blocos por pilha	4
	Unidades na camada oculta	64, 128, 256, 512
	Taxa de Aprendizado	0.0004, 0.1
	Lag	1, 3, 6, 9, <b>15</b> , 18
<b>CNN</b>	Lag	6, 9, <b>15</b> , 18
	Filtros	32, 64
	Tamanho do Kernel	3, 5
	Unidades na camada oculta	128, 256
	Função de Ativação	ReLU, Sigmoid, Tangente
	Otimizador	Adam

**Fonte:** O Autor

#### 6.4 RESULTADOS

A Tabela 2 mostra o desempenho dos modelos nas métricas de avaliação, em cada um dos conjuntos de dados em estudo. Considerando apenas os modelos individuais, os que melhor aprenderam as séries de Pernambuco, Chicago e Los Angeles foram o N-BEATS, SVR e LSTM, respectivamente. Já nos modelos estáticos, a média teve um desempenho superior ao da mediana em ambas as métricas para as três séries temporais.

Por fim, considerando apenas os *ensembles* treináveis, o eSVR foi o melhor modelo para

as séries Pernambuco e Los Angeles, enquanto o eCNN conseguiu aprender de maneira mais efetiva a série de Chicago. Nota-se que o ensemble de média foi superior às alternativas, em ambas as métricas de avaliação, para as três séries temporais estudadas.

A Tabela 3 é bastante similar à anterior, porém apresenta os resultados obtidos no artigo de (PASTOR et al., 2023), que utiliza um protocolo experimental bastante similar ao deste trabalho, sobre as mesmas séries temporais. A comparação dos valores das duas tabelas mostra que os melhores resultados do trabalho atual superam o anterior: os melhores RMSE calculados na Tabela 2 foram de 20.91, 1474.51 e 1226.67, para as séries de Pernambuco, Chicago e LA, respectivamente, enquanto (PASTOR et al., 2023) obteve valores de 23.30, 1614.38 e 1376.66 para o RMSE. O mesmo vale para a métrica MAPE, onde a Tabela 2 indica 5.78, 6.46 e 3.60, em comparação com os valores 6.36, 6.91 e 4.61 da 3.

A Tabela 4 consiste em um ranqueamento dos modelos, baseado nos valores da Tabela 2. Assim, o número 1 em cada coluna é aquele modelo que obteve o melhor desempenho em cada métrica, enquanto o número 12 obteve o pior. A coluna de *rank* médio indica a média simples de cada linha. É possível constatar que o melhor modelo individual de forma geral foi o SVR, enquanto o ARIMA foi o pior ranqueado. Os ensembles treináveis mostraram um desempenho abaixo do esperado: o eCNN foi o melhor colocado, com um ranque médio de apenas 7,00. Contudo, os ensembles estáticos obtiveram um ótimo desempenho, com ambos média simples e mediana simples obtendo a primeira e segunda colocação geral, respectivamente.

Tabela 2 – Medidas de desempenho para os modelos individuais e ensembles não treináveis e treináveis. Valores em negrito indicam o melhor valor para cada tipo de abordagem. Os valores em negrito e sublinhado indicam o melhor desempenho entre os modelos avaliados.

Abordagem	Modelos	Pernambuco		Chicago		LA	
		RMSE	MAPE	RMSE	MAPE	RMSE	MAPE
<b>Modelos Únicos</b>	ARIMA	32.69	6.87	2263.06	8.09	5022.20	9.10
	SVR	24.11	6.78	<b>1774.52</b>	<b>7.67</b>	2556.44	6.33
	MLP	24.20	6.72	2390.04	10.77	1782.25	4.96
	NBEATS	<b>24.09</b>	<b>6.52</b>	2651.15	12.00	2448.38	6.84
	LSTM	24.73	6.86	2456.66	10.84	<b>1402.75</b>	<b>4.38</b>
	CNN	27.78	8.07	2020.48	9.23	2126.19	7.31
<b>Ensemble Não Treinável</b>	Média Simples	<u><b>20.91</b></u>	<u><b>5.78</b></u>	<u><b>1474.51</b></u>	<u><b>6.46</b></u>	<u><b>1226.67</b></u>	<u><b>3.60</b></u>
	Mediana Simples	24.42	6.77	1857.04	8.09	1447.78	4.33
<b>Ensemble Treinável</b>	eSVR	<b>25.12</b>	<b>7.24</b>	6383.79	28.65	<b>2064.98</b>	<b>7.07</b>
	eMLP	27.47	7.93	2391.90	10.75	2304.45	8.20
	eCNN	27.25	7.94	<b>1856.63</b>	<b>7.95</b>	2145.32	7.91
	eLSTM	26.56	7.66	3120.72	14.96	2686.05	9.57

Fonte: O Autor

Tabela 3 – Medidas de desempenho para os modelos individuais e ensembles não treináveis e treináveis no trabalho de (PASTOR et al., 2023). Valores em negrito indicam o melhor valor para cada tipo de abordagem. Os valores em negrito e sublinhado indicam o melhor desempenho entre todos os modelos avaliados.

Abordagem	Modelos	RMSE	MAPE	RMSE	MAPE	RMSE	MAPE
		Pernambuco		Chicago		LA	
Modelos Únicos	ARIMA	32.69	6.87	<b>2173.80</b>	<b>7.28</b>	5003.23	8.93
	SVR	<b>24.11</b>	<b>6.78</b>	2975.28	13.55	2556.44	6.33
	MLP	24.24	6.79	3011.97	13.61	<b><u>1376.45</u></b>	5.00
	LSTM	24.73	6.94	2305.90	9.65	1445.02	5.06
Ensemble Não Treinável	Média Simples	<b>23.99</b>	<b>6.63</b>	1795.13	7.41	1661.60	5.00
	Mediana Simples	24.20	6.77	1801.77	7.38	<b>1467.66</b>	<b><u>4.61</u></b>
	Média Simples <sub>2</sub>	24.17	6.79	1994.58	8.32	1971.03	5.56
	Média Simples <sub>3</sub>	24.35	6.84	1748.88	7.32	1749.21	5.11
	Mediana Simples <sub>2</sub>	24.17	6.79	1994.58	8.32	1971.03	5.56
	Mediana Simples <sub>3</sub>	24.39	6.86	<b><u>1614.38</u></b>	<b><u>6.91</u></b>	1492.21	4.65
Ensemble Treinável	eSVR	23.59	6.44	2295.99	10.11	2145.35	<b>7.32</b>
	eMLP	24.22	6.86	2037.20	8.75	2579.05	9.20
	eSVR <sub>2</sub>	23.79	6.58	2347.02	10.24	<b>2117.74</b>	7.39
	eSVR <sub>3</sub>	<b><u>23.30</u></b>	<b><u>6.36</u></b>	2345.93	10.30	2442.98	8.28
	eMLP <sub>2</sub>	23.95	6.78	2007.17	8.54	2773.91	9.96
	eMLP <sub>3</sub>	24.20	6.85	<b>1782.30</b>	<b>7.45</b>	2599.08	9.27

Fonte: (PASTOR et al., 2023)

Tabela 4 – Ranqueamento dos modelos baseado nas métricas de avaliação. Valores em negrito mostram o melhor rank para o tipo de abordagem, e valores sublinhados mostram os melhores ranks gerais para cada métrica.

Abordagem	Modelos	Pernambuco		Chicago		LA		Rank Médio
		RMSE	MAPE	RMSE	MAPE	RMSE	MAPE	
<b>Modelos Únicos</b>	ARIMA	12	7	6	5	12	11	8.83
	SVR	3	5	<b>2</b>	<b>2</b>	10	5	<b>4.50</b>
	MLP	4	3	7	8	4	4	5.00
	NBEATS	<b>2</b>	<b>2</b>	10	10	9	6	6.50
	LSTM	6	6	9	9	<b>2</b>	<b>3</b>	5.83
	CNN	11	13	5	6	6	8	8.00
<b>Ensemble Não Treinável</b>	Média Simples	<u><b>1</b></u>	<u><b>1</b></u>	<u><b>1</b></u>	<u><b>1</b></u>	<u><b>1</b></u>	<u><b>1</b></u>	<u><b>1.00</b></u>
	Mediana Simples	5	4	4	4	<b>3</b>	<b>2</b>	3.66
<b>Ensemble Treinável</b>	eSVR	<b>7</b>	<b>8</b>	12	12	<b>5</b>	<b>7</b>	8.50
	eMLP	10	10	8	7	8	10	8.83
	eCNN	9	11	<b>3</b>	<b>3</b>	7	9	<b>7.00</b>
	eLSTM	8	9	11	11	11	12	10.33

Fonte: O Autor

## 7 CONCLUSÕES E TRABALHOS FUTUROS

Este estudo teve como objetivo atacar o problema de previsão de séries temporais através do emprego de táticas de *ensemble*. Foram implementados 6 modelos individuais, ARIMA, SVR, MLP, LSTM, CNN e N-BEATS, testados em 3 séries temporais diferentes: CVLI de Pernambuco, crimes em Chicago e crimes em Los Angeles, e então foram combinados através de *ensembles stack*, além do cálculo de média e mediana simples de suas previsões. Os *stacks* foram alimentados com as previsões dos modelos individuais no conjunto de validação.

As métricas de avaliação RMSE e MAPE apontaram que o *ensemble* de média foi o melhor dentre todas as estratégias avaliadas, obtendo a melhor marca em ambas as medidas para todas as séries temporais. O ranqueamento médio dos modelos com base nas métricas também aponta que o *ensemble* de mediana apresentou o segundo melhor desempenho geral. Os resultados indicam que combinar diferentes modelos é, sim, uma boa tática para o problema da previsão de crimes. No entanto, percebe-se que os *stacks* não alcançaram uma boa capacidade de generalização. Isso provavelmente se deve ao baixo volume de dados disponíveis para treinamento, pois, como foram treinados com as previsões do conjunto de validação, uma quantia de entradas equivalente a 25% do volume das séries foi utilizada.

Ao finalizar este estudo, podem-se destacar alguns direcionamentos para pesquisas futuras. Apenas os dados do volume de crimes cometidos a cada mês foram usados para treinamento, mas é possível utilizar outras variáveis, como nível de escolaridade da região, IDH, PIB per capita, etc. Outra possibilidade, se inspirando no trabalho de (ZHUANG et al., 2017), é utilizar dados que contenham latitude e longitude, para poder mapear as cidades observadas e realizar previsões mais pontuais para cada bairro. Outra metodologia, pouco utilizada no contexto estudado, é se valer de outras táticas de combinações de indutores, como modelos híbridos, que já se mostraram competentes para previsões de séries temporais de outras naturezas (PADILHA et al., 2022) (ZHANG, 2003).

## REFERÊNCIAS

- ALWEE, R.; SHAMSUDDIN, S. M. H.; SALLEHUDDIN, R. Hybrid support vector regression and autoregressive integrated moving average models improved by particle swarm optimization for property crime rates forecasting with economic indicators. *The Scientific World Journal*, v. 2013, p. 951475, 2013. Disponível em: <https://doi.org/10.1155/2013/951475>.
- BORIS, O. et al. N-beats: neural basis expansion analysis for interpretable time series forecasting. In: *International Conference on Learning Representations (ICLR)*. New Orleans: [s.n.], 2020.
- CHATFIELD, C. *Time-series forecasting*. 1. ed. New York: Chapman and Hall/CRC, 2000.
- CHATFIELD, C. *The analysis of time series: an introduction*. 6. ed. New York: Chapman and Hall/CRC, 2003.
- CHEN, P.; YUAN, H.; SHU, X. Forecasting crime using the arima model. In: *Fifth International Conference on Fuzzy Systems and Knowledge Discovery*. Jinan, China: [s.n.], 2008. p. 627–630.
- COHEN, L. E. Modeling crime trends: a criminal opportunity perspective. *Journal of Research in Crime and Delinquency*, v. 18, n. 1, p. 138–164, 1981.
- CRUZ-NAJERA, M. A.; TREVIÑO-BERRONES, M. G.; PONCE-FLORES, M. P.; TERÁN-VILLANUEVA, J. D.; CASTÁN-ROCHA, J. A.; IBARRA-MARTÍNEZ, S.; SANTIAGO, A.; LARIA-MENCHACA, J. Short time series forecasting: Recommended methods and techniques. *Symmetry*, v. 14, n. 6, 2022. ISSN 2073-8994. Disponível em: <https://www.mdpi.com/2073-8994/14/6/1231>.
- Fórum Brasileiro de Segurança Pública. *Anuário Brasileiro de Segurança Pública*. São Paulo: Fórum Brasileiro de Segurança Pública, 2022.
- GUO, L. et al. The hybrid prophet-svr approach for forecasting product time series demand with seasonality. *Computers Industrial Engineering*, v. 161, p. 107598, 2021.
- IVANYUK, V. Forecasting of digital financial crimes in russia based on machine learning methods. *Journal of Computer Virology and Hacking Techniques*, v. 20, p. 349–362, 2024.
- KHAN, S.; ALGHULAIKHA, H. Arima model for accurate time series stocks forecasting. *International Journal of Advanced Computer Science and Applications*, v. 11, 2020.
- MUTHAMIZHARASAN, M.; PONNUSAMY, R. Forecasting crime event rate with a cnn-lstm model. In: RAJ, J. S.; KAMEL, K.; LAFATA, P. (Ed.). *Innovative data communication technologies and application*. Singapore: Springer, 2022, (Lecture Notes on Data Engineering and Communications Technologies, v. 96).
- PADILHA, G. A. G. et al. Transformer-based hybrid forecasting model for multivariate renewable energy. *Applied Sciences*, v. 12, n. 21, p. 10985, 2022.
- PASTOR, R. et al. *Improving crime prediction through ensembles*. Recife: [s.n.], 2023.
- QIU, X. et al. Ensemble deep learning for regression and time series forecasting. In: *IEEE Symposium on Computational Intelligence in Ensemble Learning (CIEL)*. [S.l.: s.n.], 2014. p. 1–6.