Sistema FIEB | SENAI CIMATEC

PELO FUTURO DA INOVAÇÃO

CENTRO UNIVERSITÁRIO SENAI CIMATEC

ESPECIALIZAÇÃO EM
DATA SCIENCE & ANALYTICS

EDNILSON ROSA

PREVISÃO DE CURTO PRAZO DE RECEITAS TRIBUTÁRIAS EM SÉRIES TEMPORAIS COM USO DE REDES NEURAIS

Salvador (BA)
2020

EDNILSON ROSA

PREVISÃO DE CURTO PRAZO DE RECEITAS TRIBUTÁRIAS EM SÉRIES TEMPORAIS COM USO DE REDES NEURAIS

Projeto apresentado ao CENTRO UNIVERSITÁRIO SENAI CIMATEC como requisito parcial para obtenção do título de Especialista em *Data Science & Analytics*.

Salvador (BA)
2020

CENTRO UNIVERSITÁRIO SENAI CIMATEC
ESPECIALIZAÇÃO EM DATA SCIENCE & ANALYTICS

ATA DE APRESENTAÇÃO DE PROJETO FINAL DE CURSO

Ata de apresentação do Projeto Final de Curso, "**PREVISÃO DE CURTO PRAZO DE RECEITAS TRIBUTÁRIAS EM SÉRIES TEMPORAIS COM USO DE REDES NEURAIS"**, submetido pelo aluno **Ednilson Gimenes Rosa**, como parte dos requisitos para obtenção do Certificado de **Especialista em Data Science & Analytics** pelo Centro Universitário SENAI CIMATEC, às 10horas do dia 27 de novembro de 2020. Reuniu-se remotamente pela plataforma Teams, a Banca Examinadora designada pelo Prof Dr Erick Giovani Sperandio Nascimento – Orientador, constituída pelo Prof Esp Prabhat Kumar de Oliveira e pelo Prof Esp Flavio Santos Conterato. Ana Luiza Guimarães, Coordenadora da Especialização, deu início aos trabalhos com as devidas orientações, e a exposição foi realizada pelo estudante dentro do prazo de tempo estabelecido. Ao final da apresentação a banca reuniu-se atribuindo a seguinte nota: 8.5 (Oito ponto cinco).

A banca de avaliadores decidiu pela:

(X) Aprovação do trabalho
Caberá ao aluno apresentar em no máximo em 30 (trinta) dias a contar da data de assinatura desta Ata, uma cópia do trabalho em PDF com restrição de edição, constando as considerações pontuadas pela banca. A Ata de Apresentação do Projeto Final de Curso deve ser digitalizada e inserida na terceira página do PFC ou como anexo do artigo.

( ) Reprovação do trabalho
O aluno terá que se matricular novamente no TCC – Trabalho de Conclusão de Curso e ser submetido a uma banca avaliadora no semestre seguinte**.**

As ações consequentes ao status de Aprovação deverão obedecer ao prazo proposto acima sob pena do parecer final ser modificado para o status de Reprovado automaticamente e sem possibilidade de recurso.

Para constar, lavrou-se a presente ata que vai assinada por todos os membros da Banca. Por estarem cientes de suas obrigações estão de acordo com os termos desse documento:

Salvador, 27 de novembro de 2020.

Assinado digitalmente por Erick Giovani Sperandio Nascimento
DN: C=BR, S=Bahia, L=Salvador, O=SENAI/DR/BA, CN=Erick Giovani Sperandio Nascimento, E=erick.sperandio@fieb.org.br
Razão: Eu sou o autor deste documento
Localização: sua localização de assinatura aqui
Data: 2020.12.10 14:27:08-03'00'
Foxit Reader Versão: 10.1.0

Prof Dr Erick Giovani Sperandio Nascimento
Orientador

Prabhat Kumar de Oliveira

Assinado digitalmente por Prabhat Kumar de Oliveira
DN: C=BR, S=Bahia, L=Salvador, O=SENAI/DR/BA, CN=Prabhat Kumar de Oliveira, E=prabhat.oliveira@fieb.org.br
Razão: Eu estou aprovando este documento
Localização: sua localização de assinatura aqui
Data: 2020.12.10 16:13:18-02'00'
Foxit Reader Versão: 10.1.0

Prof Esp Prabhat Kumar de Oliveira
Membro da banca

Flavio Santos Conterato

Assinado digitalmente por Flavio Santos Conterato
DN: C=BR, S=Bahia, L=Salvador, O=SENAI/DR/BA, CN=Flavio Santos Conterato, E=flavio.conterato@fieb.org.br
Razão: Eu estou aprovando este documento com minha assinatura de vinculação legal
Localização: SENAI CIMATEC
Data: 2020.12.09 17:00:01-03'00'
Foxit Reader Versão: 10.1.0

Prof Esp Flavio Santos Conterato
Membro da banca

Assinado digitalmente por: Ana Luiza Medeiros Guimaraes Magalhaes
O tempo: 16-12-2020 14:26:38

Ana Luiza Medeiros Guimarães
Coordenadora Especialista

SUMÁRIO

1. LISTA DE FIGURAS

2. LISTA DE TABELAS

3. INTRODUÇÃO

A Secretaria da Fazenda do Estado da Bahia tem como missão prover e administrar os recursos financeiros que viabilizam as políticas públicas do Estado da Bahia, sendo responsável pela arrecadação tributária e pela gestão financeira das contas do Estado. Em função disso e dado o impacto que esse controle exerce sobre as finanças do Estado, o acompanhamento do comportamento do fluxo de receitas e despesas (fluxo de caixa) é de vital importância para essa secretaria.

Constituindo 80% das receitas do Estado da Bahia (BAHIA, 2020), as receitas tributárias, com destaque para o ICMS[1], são críticas para a execução financeira estadual, que delas depende para prestação dos seus serviços. Desta forma, é vital que se possa obter uma estimativa, com uma boa precisão e acurácia, do valor total das receitas tributárias que serão recolhidas pelos contribuintes nos próximos dias. Com essa previsão, a Área Financeira da Secretaria da Fazendo pode tomar importantes decisões a respeito das movimentações financeiras do Caixa Único do Estado, podendo decidir melhor sobre pagamentos e/ou aplicações financeiras dos recursos disponíveis nos dias seguintes.

Atualmente, essa previsão da arrecadação é feita através de métodos estatísticos tradicionais, com o uso de planilhas e consultas aos dados históricos. Essa forma de realizar essa previsão carece de uma maior eficiência e precisão, demandando, ainda, dedicação de pessoal especializado para sua elaboração.

Isso deixa uma lacuna a ser preenchida, na forma de uma solução que seja capaz de fornecer, de forma automática e com confiabilidade, o valor esperado para o total dos valores da arrecadação tributária do Estado nos cinco dias futuros. Tal solução tem o potencial de proporcionar um grande benefício para a gestão das finanças pela Secretaria da Fazenda, permitindo um melhor controle de seu fluxo de caixa com menor custo de mão de obra envolvida nessa previsão.

[1] Imposto sobre Circulação de Mercadorias e Serviços

4. MOTIVAÇÃO

O desenvolvimento deste trabalho foi motivado pela busca de uma solução para melhorar a previsão para os cinco dias seguintes dos valores da arrecadação tributária do Estado, proporcionando maior agilidade e precisão na gestão do fluxo de caixa das finanças estaduais, de forma automatizada e com menor custo de mão de obra.

5. OBJETIVOS

5.1. OBJETIVO GERAL

Construir um modelo baseado em Inteligência Artificial capaz de realizar previsão dos valores da arrecadação tributária diária para os cinco dias seguintes, com base no comportamento da série histórica de entrada de receitas tributárias e de outros atributos relevantes.

5.2. OBJETIVOS ESPECÍFICOS

Levantar e analisar os conjuntos de dados de arrecadação tributária diária, avaliando os agrupamentos de receitas com características comuns em relação ao seu comportamento e sua contribuição na composição do total da arrecadação tributária do Estado da Bahia.

Construir e testar modelos de aprendizagem de máquina para previsão de séries temporais visando a predição dos próximos cinco dias à frente dos valores da arrecadação tributária estadual, com o uso de algoritmos de Inteligência Artificial do tipo MLP[2] e LSTM[3].

Realizar ajustes finos e validação dos modelos, avaliando suas capacidades de generalizar e de desempenhar bem quantitativamente sobre os dados de validação e testes (com valores de R2[4] e FAC2[5] superiores a 90%) e comparando seu desempenho ao de um modelo de Regressão Linear.

---

[2] *Multilayer Perceptron*
[3] *Long Short-Term Memory*
[4] R Quadrado
[5] Fator de 2

Identificar e salvar o melhor modelo obtido nos ensaios e testes, de forma a deixá-lo pronto para a colocação em produção em um processo automatizado de previsão das receitas tributárias.

6. REFERENCIAL TEÓRICO

A previsão de valores futuros com base na sua série histórica ao longo do tempo tem aplicações nos mais diversos campos das áreas cientifica, industrial, social e econômica. Séries temporais estão presentes em diversas situações cotidianas, de dados sobre fenômenos naturais ao desempenho de ativos financeiros, passando por informações demográficas e sanitárias. Possuir a capacidade de se predizer, com uma boa precisão, as próximas ocorrências de uma série temporal, pode ser crucial para uma boa tomada de decisões e um diferencial importante em muitas situações reais.

6.1. MÉTODOS CLÁSSICOS DE PREVISÃO DE SÉRIES TEMPORAIS

Ao longo do tempo, diversas abordagens têm sido utilizadas no intuito de se conseguir chegar a modelos que atendam satisfatoriamente as necessidades de se prever dados futuros em séries temporais. Como pontuado em Livieris & Pintelas (2019), técnicas clássicas, como Regressão Linear ou o Modelo Autorregressivo Integrado de Médias Móveis – ARIMA[6], têm sido aplicadas, há muito tempo, para resolução desse tipo de problema.

No entanto, como colocado por Brownlee (2018), esses métodos mais clássicos de previsão de séries temporais, sejam técnicas auto regressivas, como o ARIMA e sua extensão para séries temporais – SARIMA[7], sejam técnicas de Suavização Exponencial[8], como o SES[9] e outros, possuem certas limitações que impactam em seu desempenho:

a. Foco na completude dos dados: dados faltantes ou corrompidos normalmente não são suportados por essas técnicas;

[6] *Autoregressive Integrated Moving Average*
[7] *Seasonal Autoregressive Integrated Moving Average*
[8] *Exponential Smoothing*
[9] *Single Exponential Smoothing*

b. Foco nas relações lineares: distribuições mais complexas, sem um padrão linear claro, não produzem bons resultados quando utilizadas com esses métodos;

c. Foco em dependência temporal fixa: relações irregulares entre as observações costumam apresentar problemas nessas abordagens;

d. Foco em dados univariados: muitos problemas reais requerem uma análise multivariada, que não é bem suportada por esses modelos;

e. Foco em previsões de apenas um passo à frente (*one-step forecast*[10]): em muitos casos, a previsão de um horizonte mais amplo se faz necessária, o que também não é bem suportado por essas técnicas.

6.2. PREVISÃO DE SÉRIES TEMPORAIS COM USO DE REDES NEURAIS

Mais recentemente, a utilização de arquiteturas de Redes Neurais Artificiais (ANN[11]) têm obtido atenção de muitos pesquisadores, a exemplo de Sami & Nazir (2018), com diversos métodos de aprendizado de máquina sendo utilizados para predição de valores em séries temporais. Esses estudos têm apresentado bons resultados, demonstrando a adequação dessas técnicas a esse tipo de previsão.

6.2.1. PREVISÃO DE SÉRIES TEMPORAIS COM REDES MULTILAYER PERCEPTRON

De acordo com Brownlee (2018), a utilização de *Machine Learning*[12] para previsão de séries temporais, com a utilização de redes MLP, resolve em grande medida os problemas dos métodos clássicos, por suportar as seguintes características:

a. Tolerância a ruídos: *outliers*[13] e até mesmo dados faltantes tendem a ser mais bem tolerados;

b. Não dependência de dados lineares: há suporte a mapeamentos não necessariamente lineares;

[10] Previsão de um único passo
[11] *Artificial Neural Networks*
[12] Aprendizado de Máquina
[13] Valores discrepantes

c. Entradas Multivariadas: permitem o uso de um número arbitrário de atributos de entrada;

d. Previsões de múltiplos passos (*multi-step forecasts*[14]): um número arbitrário de saídas pode ser usado, suportando previsões de múltiplos passos e até de múltiplas variáveis ao mesmo tempo.

6.2.2. PREVISÃO DE SÉRIES TEMPORAIS COM REDES NEURAIS RECORRENTES

Brownlee (2018) também pontua que Redes Neurais Recorrentes (RNN[15]), como as Redes de Memória de Longo Prazo (LSTM), têm se mostrado bastante eficientes na predição de séries temporais, incluindo a habilidade de identificar e manipular a ordem entre as observações ao aprender a função de mapeamento entre entradas e saídas. Assim, a dependência temporal entre as observações seria mais facilmente aprendida.

Através de modelos de Redes Neurais Recorrentes, em especial na sua variação LSTM, diversos autores têm obtido bons resultados neste tipo de previsão. Isso pode ser visto em trabalhos como Sami & Nazir (2017), Salis, Kumari & Singh (2019), Persio, & Honchar (2016), Siami & Tavakoli (2018) e Fischer & Krauss (2018).

6.3. ESTRATÉGIAS DE PREVISÃO

Além das técnicas empregadas, diferentes estratégias podem ser utilizadas quando se deseja obter mais de um item futuro de uma série temporal. Pode-se utilizar, por exemplo, uma abordagem passo a passo, com a previsão do próximo item utilizando o último valor previsto como dado de entrada, ou então uma previsão de múltiplos passos de uma só vez, a partir do último dado disponível na série temporal. Essas estratégias são exploradas por Brownlee (2020) e Vinogradov (2020).

Considerando essas estratégias, quanto à quantidade de variáveis, pode-se classificar as predições de séries temporais nos seguintes tipos:

---

[14] Previsão de múltiplos passos
[15] *Recurrent Neural Netwoks*

a. Univariadas: séries em que uma única variável é avaliada no decorrer do tempo. Pode-se referir tanto à entrada (*univariate input*[16]) como à saída (*univariate output*[17]).

b. Multivariadas: em oposição às séries univariadas, são casos em que se avalia múltiplas variáveis no decorrer do tempo. Também podem se referir tanto à entrada (*multivariate input*[18]) como à saída (*multivariate output*[19]).

Já em relação a forma de previsão, pode-se ter as seguintes classificações:

c. Passo único: são previsões de séries temporais em que se avalia um único período de tempo para a previsão. Da mesma forma, pode se referir tanto à entrada (*single-step input*[20]) como à saída (*single-step forecast*[21]).

d. Passo múltiplo: são previsões de séries temporais em que se avaliam diversos períodos de tempo para a previsão. Também, pode se referir tanto à entrada (*multi-step input*[22]) como à saída (*multi-step forecast*[23]).

6.4. APLICAÇÃO DE REDES NEURAIS PARA PREVISÃO DE SÉRIES FINANCEIRAS E TRIBUTÁRIAS

Em relação à predição de valores financeiros, a maioria dos trabalhos que abordam a aplicação de RNNs em séries temporais o fazem para a previsão de valores de ativos do Mercado de Commodities[24], à exemplo da cotação do Ouro como visto em Livieris & Pintelas (2019), Sami & Nazir (2018) e Salis, Kumari & Singh (2019), ou para tentar prever cotações do Mercado de Capitais[25], como em Persio, & Honchar (2016) e em Fischer & Krauss (2018).

[16] Entrada de uma única variável
[17] Saída de uma única variável
[18] Entrada com múltiplas variáveis
[19] Saída com múltiplas variáveis
[20] Entrada de um único passo
[21] Previsão de um único passo
[22] Entrada de múltiplos passos
[23] Previsão de múltiplos passos
[24] Mercado que comercializa no setor econômico primário
[25] Mercado formado pelas bolsas de valores, sociedades corretoras e outras instituições financeiras

Alguns trabalhos que exploram a utilização de RNN para dados tributários o fazem na busca de identificação fraudes na arrecadação de impostos, como em López, Rodríguez & Santos (2019). Especificamente para a previsão de receitas do ICMS, Contreras & Cribari-Neto (2015) obtiveram bons resultados através da utilização de redes *Multilayer Perceptron* (MLP) em comparação com as técnicas de Alisamento Exponencial e de Médias Móveis Autoregressivas (SARIMA). Mais recentemente, Carmo, Boldt & Komati (2020) também exploraram a utilização de MLP para a previsão da arrecadação mensal do ICMS. Freitas, Ciarelli e Souza (2009) também demonstraram a aplicabilidade da utilização de Redes Neurais Artificiais para a predição de receitas tributárias federais, em comparação com o método ARIMA.

Nenhum desses trabalhos, no entanto, abordou a previsão de receitas estaduais diversas do ICMS nem a previsão de curto prazo em cenários de dias, ao invés de meses ou anos, como é a proposta do presente estudo.

Também não foram identificados outros trabalhos que tivessem como foco a previsão da arrecadação tributária estadual total com horizonte de previsão de poucos dias, com base nas séries históricas desses tributos.

7. METODOLOGIA

Neste trabalho, são exploradas algumas das técnicas que são consideradas o atual estado da arte na previsão de séries temporais. Foram testadas a previsão de curto prazo da arrecadação tributária do Estado da Bahia através das técnicas de Inteligência Artificial com o uso de Redes *Multilayer Perceptron (MLP)* e de Redes Neurais Recorrentes (RNN*)*, mais especificamente as Redes de Memória de Longo Prazo *(Long Short-Term Memory - LSTM)*. Por fim, para efeito de comparação do resultado e avaliação da viabilidade do modelo, foi realizada a mesma previsão através de uma Regressão Linear (RL) simples.

Os resultados de cada um dos modelos foram avaliados com base no *loss*[26] obtido no seu treinamento, assim como na avaliação das métricas MAE[27], RMSE[28], R2 e FAC2 obtidas com os dados de teste, além do MSE[29].

Para divisão dos dados em treinamento e testes, foi utilizado um corte temporal correspondente a divisão em 80/20. Desta forma, o conjunto de treinamento foi definido pelos dados de arrecadação do período de 2010 até 2017 (incluindo este último), enquanto o conjunto de testes ficou definido pelos dados de 2018 e 2019.

Em função da grande quantidade de códigos de receita existentes e por inicialmente não se ter obtido um bom resultado utilizando todas as receitas juntas como *target*[30], optou-se por se trabalhar com alguns grupos de receita agregados para o treinamento e predição. Esse agrupamento teve como base a similaridade do comportamento da entrada dessas receitas (vencimento, montante, porte do contribuinte, etc.). Assim, a ideia foi a de se fazer a predição individual de cada um desses grupos e se totalizar os valores, no final, para se chegar ao resultado desejado.

Os grupos de receita que foram utilizados para agrupamento das receitas tributárias estão representados na tabela 1.

Tabela 1: grupos de receitas tributárias

Grupo	Códigos de Receitas
1	Receitas de ICMS de contribuintes do Regime Normal[31]
2	Receitas de ICMS de Regime Sumário[32] de ICMS
3	Receitas de ICMS Eventuais
4	Receitas principais de IPVA[33]
5	Receitas Principais de ITD[34]
6	Receitas de Multas e Taxas

[26] Erro na predição dos valores pela rede neural
[27] *Mean Absolute Error*
[28] *Root Mean Squared Error*
[29] *Mean Squared Error*
[30] Valor alvo a ser previsto
[31] Regime tributário a que estão sujeitas as empresas de maior faturamento
[32] Regime tributário simplificado
[33] Imposto sobre a Propriedade de Veículos Automotores
[34] Imposto de Transmissão Causa Mortis e Doação de Quaisquer Bens ou Direitos

A estratégia de trabalho adotada foi a da preparação dos dados em vetores com janelamento de acordo com um valor de *look back*[35] definido (*multi-step input*) e com a utilização simultânea de diversos atributos (*multivariate input).*

Para o resultado da previsão, foi testada tanto a predição de um único *target* de cada vez (*univariate, single-step forecast*) como a previsão das cinco ocorrências seguintes de uma só vez (*univariate, multi-step forecast*).

Em decorrência das dificuldades na obtenção de previsões com alto grau de acerto para um único dia, provocado pela incerteza da entrada das receitas caraterística desta série temporal, optou-se pela predição dos cinco próximos dias de forma agrupada (*single-step forecast*), para se avaliar o impacto dessa estratégia no acerto geral dos algoritmos. De fato, essa estratégia se mostrou extremamente benéfica para os resultados, como veremos mais adiante

7.1 ANÁLISE EXPLORATÓRIA DE DADOS

Os dados de arrecadação tributária foram extraídos de forma agrupada por códigos de receita. Em função da grande quantidade de códigos de receita tributária que compõe a arrecadação, optou-se por se trabalhar com a granularidade reduzida a grupos de códigos de receitas afins. Desta forma, como já exposto, as receitas foram agrupadas nos cinco grandes grupos de arrecadação apresentados na tabela 1.

A análise exploratória dos dados desses grupos de receitas mostrou que os valores monetários estão mais concentrados em alguns deles, como demonstram as totalizações da série temporal. Os valores apresentados na tabela 2 representam a soma de todas as receitas diárias durante toda a série temporal de 10 anos (01/01/2010 a 31/12/2019), para cada um desses grupos de receitas, além do percentual de cada grupo em relação ao total da arrecadação no período.

[35] Quantidade de amostras de atributos avaliadas para a previsão

Tabela 2: valores totais da arrecadação por grupo

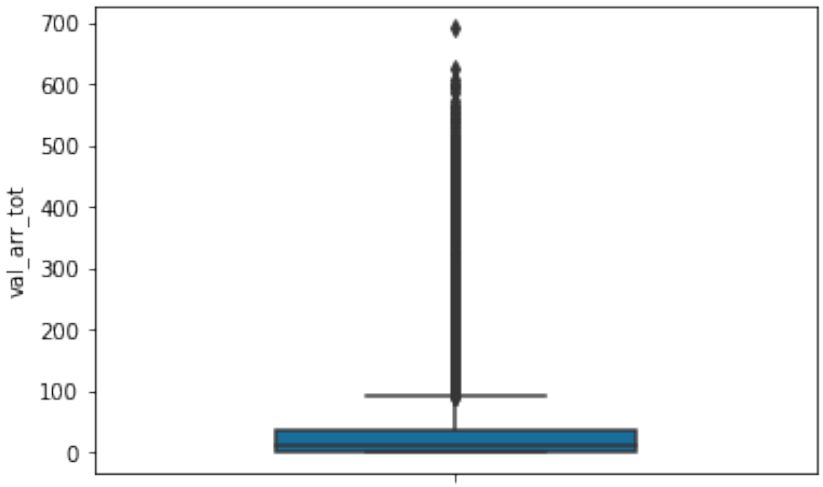
Grupo	Valores Acumulados	Percentual
1	R$ 87.403,83 Milhões	45,66%
2	R$ 57.770,58 Milhões	30,18%
3	R$ 30.504,91 Milhões	15,94%
4	R$ 9.371,46 Milhões	4,90%
5	R$ 5.486,20 Milhões	2,87%
6	R$ 873,29 Milhões	0,46%
Total	**R$ 191.410,27 Milhões**	**100,00%**

Isso deixou claro que, para um melhor acerto da arrecadação agregada, era importante que o modelo tivesse um bom desempenho nos três primeiros grupos de receitas, que são responsáveis por quase 92% da arrecadação total.

A análise por *boxplots*[36] demonstrou, ainda, uma grande concentração da ocorrência de receitas com valores mais baixos, com a existência de alguns *outliers* representados pelos dias em que ocorre a arrecadação de grandes contribuintes do Estado, como pode ser observação na figura 1.

Como também pode ser visto nesta figura, a média de arrecadação diária fica bem abaixo de 50 milhões de Reais por dia, mas existem dias em que esse valor pode ser dezenas de vezes maior.

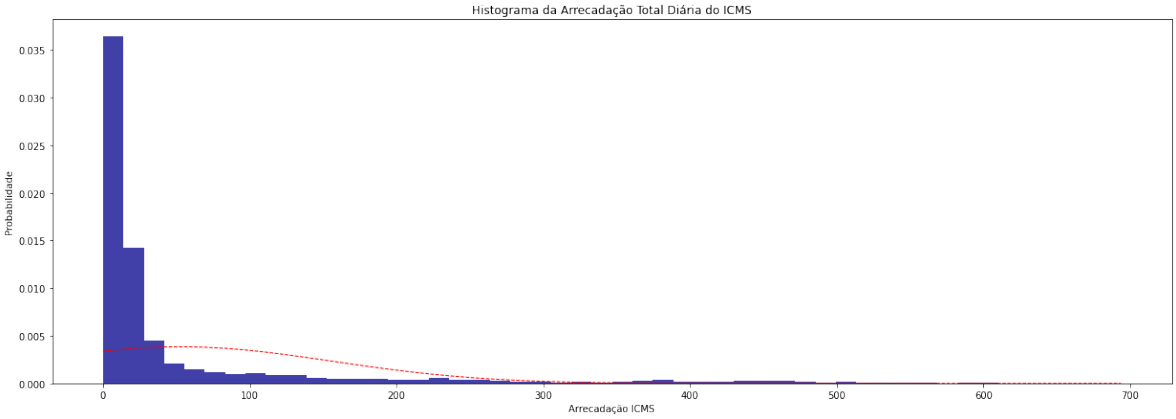
Figura 1: *boxplot* dos valores de arrecadação total diária (em milhões de R$)



[36] Diagrama de Caixa. Ferramenta gráfica que permite visualizar a distribuição e valores discrepantes

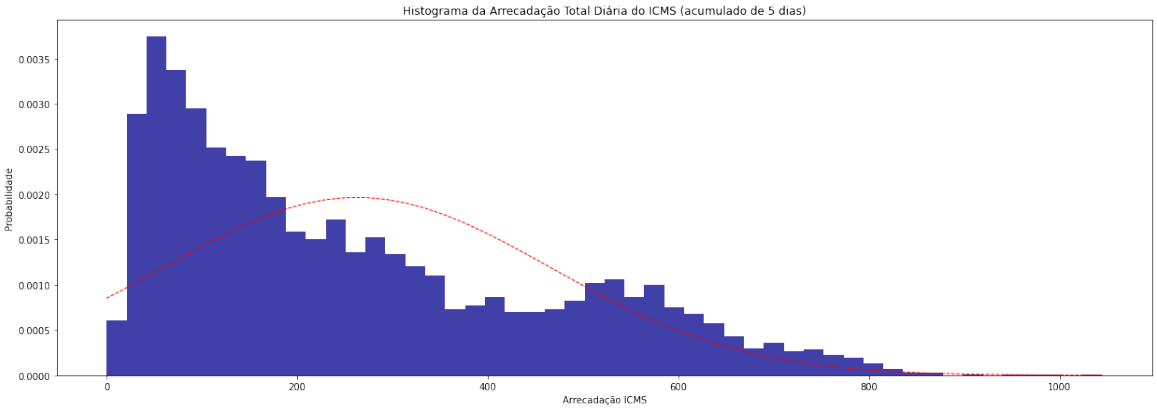
Da mesma forma, a análise do histograma da arrecadação total diária também demonstrou uma grande concentração de ocorrências de valores mais baixos, como pode ser visto na figura 2. Por esse histograma, verifica-se que também existe uma grande concentração na ocorrência de dias com valores totais de arrecadação diária de valor relativamente baixo, ao mesmo tempo em que temos alguns dias com valores de montantes bastante elevados, ainda que em um percentual bem pequeno.

Figura 2: histograma dos valores de arrecadação total diária (em milhões de R$)



Essa distribuição das ocorrências representou um fator limitante e dificultador para a previsão, já que ela estava longe da forma normal ideal para o tratamento estatístico. Ao analisarmos esse mesmo gráfico, só que para valores acumulados de cinco dias (gráfico 3), temos uma distribuição um pouco mais homogênea, o que certamente contribuiu para o melhor desempenho das previsões que obtivemos neste cenário, como será visto mais adiante.

Figura 3: histograma da arrecadação total diária para 5 dias (em milhões de R$)



O *dataset*[37] obtido pela exportação dos dados das bases da Secretaria da Fazenda do Estado da Bahia também contém, além dos valores diários de arrecadação por cada grupo de receita, dados referentes à emissão diária de Documentos de Arrecadação Estadual (DAE) e também informações referentes à classificação do dia quanto a ser útil ou não.

A informação referente aos valores de DAE é importante porque pode dar uma ideia da intenção de arrecadação do contribuinte. A análise dos dados, no entanto, identificou que esta pode ser uma informação pouco confiável, já que o contribuinte pode emitir DAE livremente, de qualquer valor e sem obrigatoriedade de pagamento, o que causa a ocorrência de distorções de diversas naturezas.

7.2. ENGENHARIA DE DADOS

Por serem oriundos de sistemas da Secretaria da Fazenda que possuem rotinas rígidas de verificação de consistência, os dados obtidos têm uma qualidade bastante elevada. Desta forma, não existem dados faltantes nem *outliers* que não sejam representativos para o modelo de previsão (à exceção dos valores de DAE emitidos, como já comentado). Por este motivo, não foi necessário um grande trabalho de tratamento de qualidade dos dados a serem usados.

Apesar disso, alguns tratamentos precisaram ser feitos, para ajustar os dados para a aplicação dos modelos de previsão:

a. Agregação dos dados de arrecadação em intervalos de cinco dias;
b. Deslocamento dos valores de DAE emitidos com vencimento em dias não úteis para o próximo dia útil.
c. Redução dos valores para se trabalhar na escala de milhões de reais, a mais usual para os objetivos da área de negócio.
d. Aplicação da função seno para os atributos de dia e mês, buscando ensinar ao modelo o comportamento cíclico dos dados.
e. Eliminação de *outliers* nos valores de emissão de DAE, que representam prováveis erros de preenchimento dessa informação pelo contribuinte, usando como base o maior valor pago de DAE no período.

[37] Conjunto de Dados

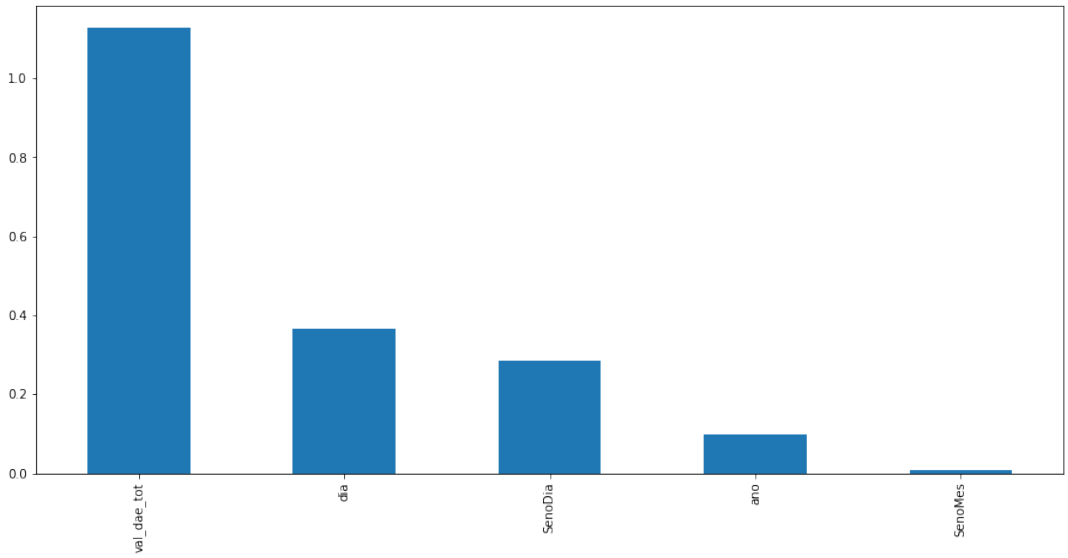
A figura 4 mostra a Matriz de Correlação de Pearson dos dados já aumentados pela aplicação da função seno para dia e mês.

Por esta matriz, pode-se observar que o atributo com maior correlação com a arrecadação total é justamente o de DAE emitido, com valor de 0.74, apesar dos problemas referentes à confiabilidade desse atributo.

Figura 4: Matriz de Correlação de Pearson entre os atributos do *dataset* tratado

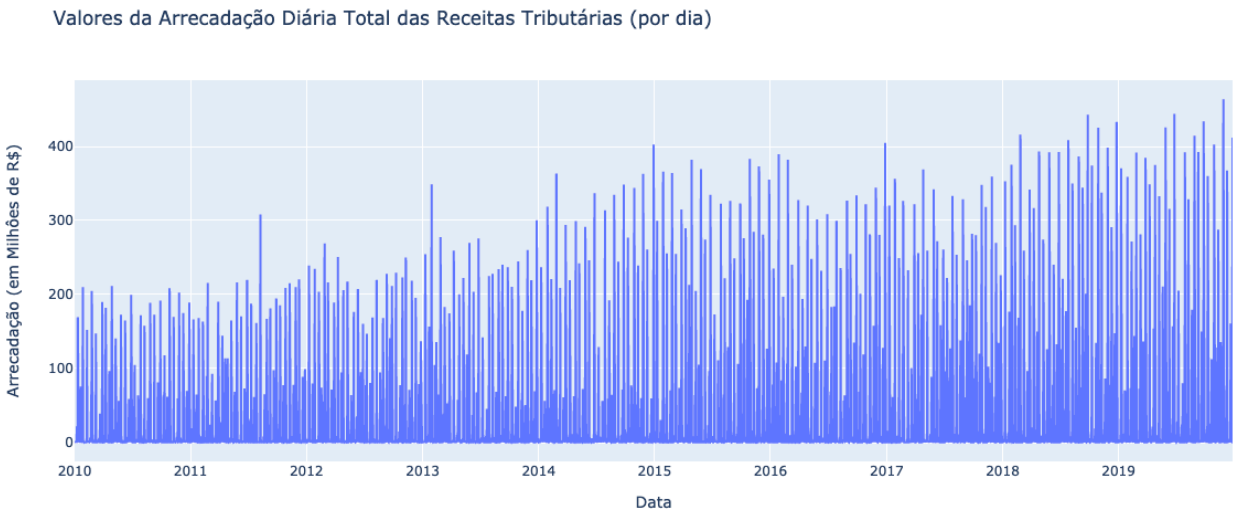
A figura 5 mostra a informação mútua entre a arrecadação total e os demais atributos, que mais uma vez mostrou uma forte correlação dos valores de DAE emitidos com o valor da arrecadação total. Fica evidenciada, também, uma forte dependência do atributo ‘dia’, algo esperado em uma série temporal de arrecadação diária.

Figura 5: gráfico de informação mútua entre os atributos do *dataset*



A figura 6 mostra o gráfico com a plotagem da arrecadação total diária das receitas tributárias agregadas durante todo o período. Por esse gráfico, podemos observar claramente a grande variabilidade dos valores de arrecadação diária total presente no *dataset* trabalhado. É possível se observar, também, uma clara tendência de incremento anual da arrecadação, algo que se confirma nos relatórios de Demonstrações Contábeis Consolidadas do Estado da Bahia (Bahia, 2020).

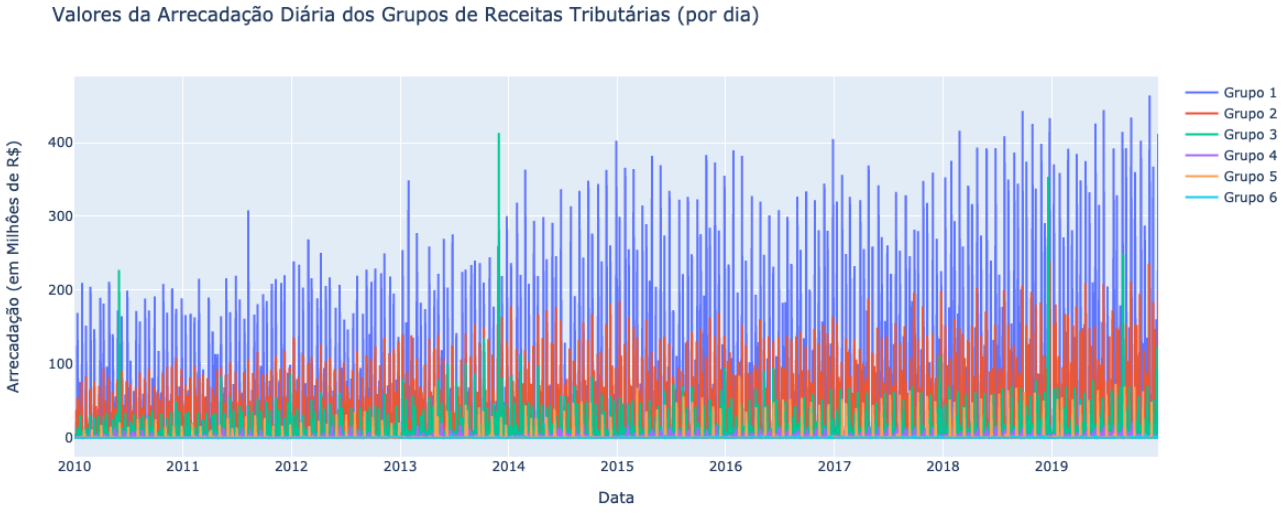
Figura 6: plotagem da arrecadação total diária (em milhões de R$)



O gráfico da figura 7 mostra a plotagem da arrecadação total diária de acordo com seus grupos de receitas tributárias. Por este gráfico, pode-se perceber a disparidade nos montantes apresentados em cada grupo, conforme já evidenciado na tabela 2.

Neste gráfico, pode-se observar, também, a ocorrência de valores fora do padrão geral em alguns grupos de receita, em especial no grupo 3 (receitas de ICMS eventuais). Esse tipo de ocorrência dificulta bastante a assertividade da previsão, uma vez que não há um padrão claro a ser identificado pelo algoritmo para a repetição.

Figura 7: plotagem da arrecadação total diária por grupos (em milhões de R$)



8. CONSTRUÇÃO DO MODELO DE INTELIGÊNCIA ARTIFICIAL

Para os dois modelos de Inteligência Artificial utilizados neste trabalho (MLP e LSTM), foi utilizado um *Random Search*[38] customizado para a variação dos hiperparâmetros e dos parâmetros da arquitetura de cada um deles. Desta forma, foram testadas diversas combinações de valores para as funções de ativação, as inicializações de *kernel*[39], os números de camadas internas, as quantidades de neurônios nas camadas ocultas, os valores de *dropout*[40], as quantidades de épocas de treinamento, os tamanhos de *batch*[41], os tamanhos de

[38] Busca Aleatória
[39] Inicialização do Núcleo
[40] Descarte utilizado para prevenir sobreajuste
[41] Tamanho do Lote

janela de previsão, além de diferentes conjuntos de atributos de entrada. Com isso, buscou-se uma combinação que proporcionasse os melhores valores de predição, de acordo com as métricas consideradas.

Além disso, como explicitado na Metodologia, os valores referentes aos diversos códigos de receitas foram agrupados, de acordo com suas caraterísticas comuns, e trabalhados de forma independente pelos modelos, obtendo-se predições separadas para cada um dos grupos que foram, no final, agregadas para obtenção do valor total da arrecadação prevista.

Foram também feitos extensivos testes e ajustes de *tunning*[42] nos modelos, com a busca das melhores práticas de ajuste fino de modelos desta natureza.

A seguir, são apresentadas a estratégia de treinamento e as principais características dos modelos de IA[43] utilizados neste trabalho.

8.1. ESTRATÉGIA DE TREINAMENTO

A estratégia adotada para o treinamento e testes, na busca pelos melhores resultados na predição da arrecadação diária de receitas tributárias, foi planejada de forma incluir três pilares principais:

1) Divisão do *target* (arrecadação total diária) em diversos grupos de códigos de receitas de acordo com suas caraterísticas em comum (natureza, tipo de contribuinte, datas de vencimento etc.). Desta forma, como descrito na tabela 1, ficou-se 6 grupos principais de receitas tributárias.

2) Estabelecimento de conjuntos de atributos (opções) a serem treinados com cada um dos grupos de receitas pelos algoritmos, buscando os melhores resultados. Desta forma, ficou-se com as opções de atributos da tabela 3.

[42] Ajuste fino
[43] Inteligência Artificial

Tabela 3: opções de atributos utilizadas nos treinamentos dos modelos

Opção	Atributos Utilizados
1	target (arrecadação)
2	target, dia
3	target, dia, mês
4	target, dia, SenoDia, dia_util
5	target, dia, mes, ano, SenoDia, SenoMes
6	target, dia, mes, ano, SenoDia, SenoMes, dia_util
7	target, dia, mes, SenoDia, SenoMes, dia_util, val_dae
8	target, dia, SenoDia, dia_util, val_dae, doc_dae
9	target, dia, mes, SenoDia, SenoMes, dia_util, val_dae, doc_dae

Com isso, foram treinados 6 grupos de receita com 9 opções para os 2 algoritmos, totalizando 108 (6 x 9 x 2) treinamentos em série. Desta forma, chegou-se ao melhor conjunto de atributos para cada grupo em cada um dos algoritmos e partiu-se para o passo seguinte, que foi fazer um *Random Search* dos parâmetros e hiperparâmetros, buscando melhorar os resultados obtidos até então.

3) Execução do *Random Search* buscando o melhor conjunto de variáveis para a melhoria dos resultados da predição. Os seguintes parâmetros e hiperparâmetros dos modelos foram variados nesse *Random Search*:

***Look Back*:** quantas instâncias passadas são apresentadas ao modelo para predição da ocorrência futura. Esse valor foi variado entre os valores de 10, 20 e 30 instâncias (3 variações).

Épocas: estabelece a quantidade máxima de épocas treinadas. Deve-se notar, entretanto, que foi utilizado também um *callback*[44] para o *Early Stopping*[45] caso o *loss* do modelo não sofresse melhoria por 8 épocas consecutivas, interrompendo o treinamento prematuramente em algumas iterações. Esse *callback* também restaura os melhores *weights*[46] do modelo, garantido que os

[44] Sub-rotina invocada durante a rotina de treinamento
[45] Interrupção prematura do treinamento
[46] Pesos

melhores pesos sejam preservados. Os valores variados foram 50, 60 e 75 (3 variações).

***Batch*:** número de amostras (tamanho de lote) de treinamento processadas de cada vez para avaliação interna do erro do modelo. Foram variados os valores de 5, 10 e 20 (3 variações).

Nº de Camadas Ocultas: número de camadas do modelo, além das camadas de entrada e saída. Foram variados os valores de 1, 2 e 3 camadas (3 variações).

Número de Neurônios Iniciais nas Camadas Ocultas: quantidade de neurônios iniciais utilizados nas camadas ocultas. Esse parâmetro foi variado com diferentes valores, dependendo do modelo treinado. As demais camadas ocultas foram definidas como multiplicadoras dos valores iniciais, em arquiteturas *flat* ou em pirâmide. Os valores iniciais definidos pelo *Random Search* foram sempre escolhidos entre 3 variações diferentes.

***Dropout* do Modelo:** taxa de *dropout* utilizada no modelo, com objetivo de minimizar o *overfitting* e melhorar o resultado da predição através do descarte do processamento de neurônios. Também foram utilizadas três variações diferentes em cada modelo.

Função de Ativação: função de ativação utilizada nas camadas ocultas do modelo. Foram utilizadas as funções "relu[47]" e "selu[48]", na variação do *Random Search*. A escolha da função de ativação também levou à escolha da inicialização do *kernel* (*Kernel Initialization*). Para a função "relu", foi utilizado a inicialização "*he_normal*", enquanto que, para a função "selu", foi utilizada a inicialização "*lecun_normal*" (2 variações).

Considerando que temos 6 diferentes grupos de receitas e 2 algoritmos de processamento, a execução de um *Grid Search*[49] completo com todas as variações previstas nesses 7 parâmetros faria com que tivéssemos 17.496

[47] *Rectified Linear Unit Activation Function*
[48] *Scaled Exponential Linear Unit Activation Function*
[49] Busca em Grade

treinamentos de nosso modelo (6 x 2 x 3 x 3 x 3 x 3 x 3 x 3 x 2), o que geraria um custo computacional e temporal elevado demais.

Em função disso, optou-se por se fazer essa busca dos melhores parâmetros e hiperparâmetros através de um *Random Search* customizado, com execução de 20 iterações para cada grupo. Desta forma, o número de treinamentos foi reduzido para 240 treinamentos (6 grupos x 2 algoritmos x 20 iterações), algo bem mais razoável.

Como veremos mais à frente nos resultados obtidos, a otimização obtida pela busca desses parâmetros e por outros ajustes feitos na arquitetura dos modelos proporcionaram uma melhora substancial nos resultados finais.

8.2. CARACTERÍSTICAS DO MODELO MLP

Para o modelo MLP, foi utilizada uma rede com uma camada de entrada densamente conectada a um conjunto de camadas ocultas que variou em número de uma a três outras camadas *dense*[50].

As camadas de entrada e ocultas utilizaram funções de ativação e inicialização de *kernel* variando entre *relu* e *selu*, para as funções de ativação, e *he_normal* e *lecun_normal*, para a inicialização do *kernel*.

Os números de neurônios para a camadas de entrada da rede MLP variaram entre 16, 24 e 64 neurônios, com as camadas ocultas utilizando múltiplos desses valores em valores decrescentes (x6, x4 e x2), em uma arquitetura em forma de pirâmide invertida.

O modelo MLP foi compilado utilizando o otimizador "*adam*" e o "*mean squared error*" (mse) como métrica de *loss*.

Os números de épocas de treinamento variaram entre 50, 60 e 75, com utilização de um *callback* de *early stopping* para interrupção do treinamento em caso de não melhoria do resultado por mais de oito épocas consecutivas.

Os tamanhos de *batch* variaram entre 5, 10 e 20.

[50] Camada densamente conectada

Para controle do *overfitting*[51] foi implementada uma camada de *dropout* com valores variando entre 0.15, 0.2, 0.3 e 0.5.

O modelo ainda testou a variação do *look back* da janela da série temporal variando entre 10, 20 e 30 dias.

Por fim, para a camada de saída do modelo MLP foi utilizada a função de ativação *linear*.

A tabela 4, abaixo, apresenta a arquitetura resultante de uma das execuções de treinamento do modelo MLP

Tabela 4: arquitetura de uma das execuções do modelo MLP

Layer (type)	Output Shape	Param #
dense_1 (Dense)	(None, 64)	13504
dense_2 (Dense)	(None, 384)	24960
dense_3 (Dense)	(None, 128)	49280
dropout_1 (Dropout)	(None, 128)	0
dense_4 (Dense)	(None, 1)	129

8.3. CARACTERÍSTICAS DO MODELO LSTM

Para o modelo LSTM, foi utilizada uma rede com uma camada de entrada LSTM que podia ser seguida por uma outra camada LSTM (a depender do parâmetro *layers* selecionado pelo *Random Search*), além de uma camada de *dropout* e de uma saída *dense*.

Nos casos em que a janela de predição era maior do que 1 (um), foram utilizadas, ainda, camadas com as opções *RepeatVector*, logo após a camada de entrada, e *TimeDistributed*, na saída da rede.

As funções de ativação e inicialização de *kernel* utilizadas variaram entre *relu* e *selu*, para as funções de ativação, e *he_normal* e *lecun_normal*, para a inicialização do *kernel*.

[51] Sobreajuste

Os números de neurônios para a primeira camada da rede LSTM variaram entre 128, 256 e 512 neurônios.

O modelo LSTM foi compilado utilizando o otimizador “*adam*” e o “*mean squared error*” (mse) como métrica de *loss*.

Os números de épocas de treinamento variaram entre 50, 60 e 75, com utilização de um *callback* de *early stopping* para interrupção do treinamento em caso de não melhoria do resultado por mais de oito épocas consecutivas.

Os tamanhos de *batch* variaram entre 5, 10 e 20.

Para controle do *overfitting* foi implementada uma camada de *dropout* com valores variando entre 0.2, 0.3, 0.45 e 0.55.

O modelo também testou a variação do *lookback* da janela da série temporal variando entre 10, 20 e 30 dias.

A tabela 5, abaixo, apresenta a arquitetura resultante de uma das execuções de treinamento do modelo LSTM.

Tabela 5: arquitetura de uma das execuções do modelo LSTM

Layer (type)	Output Shape	Param #
lstm_2 (LSTM)	(None, 30, 128)	69632
lstm_3 (LSTM)	(None, 128)	131584
dropout_5 (Dropout)	(None, 128)	0
dense_8 (Dense)	(None, 1)	129

8.4. CARACTERÍSTICAS DO MODELO DE REGRESSÃO LINEAR

Para compararmos com o desempenho dos modelos de Redes Neurais, foi utilizado um modelo de Regressão Linear (RL) simples. Neste caso, o único parâmetro variado foi o valor do *look back*.

9. RESULTADOS E DISCUSSÕES

No início deste trabalho, a intenção era se obter a previsão diária para os cinco próximos dias individualmente. Os primeiros resultados, no entanto, deixaram claro que o desafio seria grande, em função de algumas características que dificultavam o atingimento desse objetivo:

a. As receitas tributárias têm vencimento mensal em dias diferentes, a depender de aspectos como porte, regime tributário e segmento econômico ao qual pertence cada contribuinte;

b. Apesar de o vencimento dos tributos ser, via de regra, sempre no mesmo dia de cada mês para cada contribuinte, não há como se garantir que o contribuinte irá fazer o recolhimento de cada tributo exatamente no dia do vencimento, podendo vir a fazê-lo alguns dias antes ou, até mesmo, com algum atraso (ainda que isso lhe gere um acréscimo em forma de multa);

c. Em alguns meses, a data de vencimento de determinado tributo pode cair em um final de semana ou outro dia não útil, fazendo com que parte dos contribuintes antecipe o recolhimento enquanto outros atrasem esse pagamento;

d. Questões de fluxo de caixa de cada contribuinte pode fazer com que eles antecipem ou posterguem o recolhimento de algum ou de todos os tributos devidos por eles.

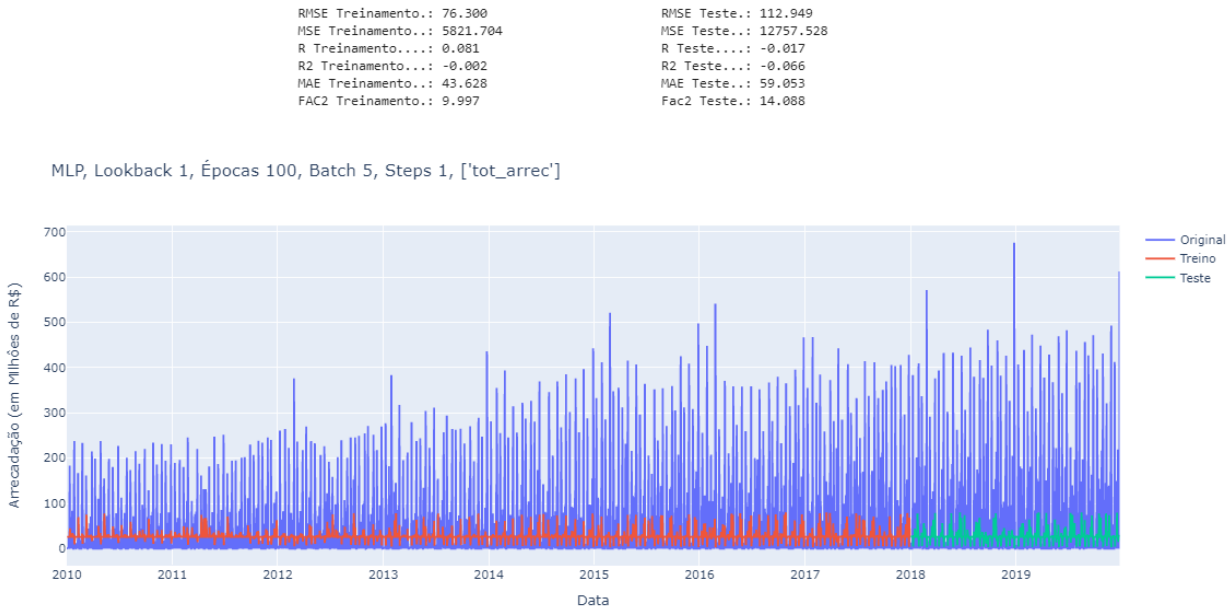
e. Contribuintes sujeitos ao recolhimento de mais de um tributo podem, eventualmente, optar por fazer os recolhimentos de forma concentrada, independente do vencimento de cada um deles.

Todos esses fatores e outros semelhantes fazem com que seja muito difícil se acertar, com precisão de um único dia, uma previsão sobre recolhimento de tributos mensais tão diversos, já que são muitos os fatores sobre os quais a Secretaria da Fazenda não tem controle. É bem mais fácil, no entanto, acertar uma previsão com precisão de alguns dias de forma agregada, já que esse

horizonte acaba por acomodar esses pequenos desvios, fazendo com que a receita esperada efetivamente ocorra em torno da data prevista.

A figura 8, a seguir, mostra o resultado que foi obtido nas primeiras tentativas de se acertar a arrecadação diária com base na série histórica com o uso de uma rede MLP para um único dia de alvo na previsão.

Figura 8: plotagem da série original versus treino e teste de um modelo MLP inicial



Como pode ser visto nessa figura, o resultado dessas primeiras tentativas foi muito ruim. Ficou claro que a rede teve grande dificuldade de aprender e prever as ocorrências de arrecadação, com um modelo trabalhando com granularidade de um dia de previsão, com todas as receitas agregadas e com apenas a própria arrecadação como atributo de entrada na série histórica.

Para enfrentar esse desafio, diversos ajustes e *tunning* dos modelos tiveram que ser implementados. Dentre os ajustes feitos, pode-se destacar os seguintes:

a. Utilização de outros atributos (*mutivariate input*);
b. Aplicação da função seno nos atributos derivados da data;
c. Ajustes nas arquiteturas e parâmetros dos modelos;
d. Ampliação do *look back* do treinamento;

e. Inclusão de novos dados de DAE emitidos;

f. Mudança da estratégia para se prever 5 dias agregados;

g. Estratificação da arrecadação em grupos de receita tributária;

h. Busca de parâmetros otimizados através de um *Random Search*.

Desses ajustes, sem dúvida o mais importante foi a alteração da estratégia de predição para se prever os próximos cinco dias de forma agregada, ao invés de um único dia de cada vez. Ficou claro que seria bem mais fácil para a rede acertar o total previsto para a arrecadação nos próximos cinco dias do que acertar cada dia individualmente. Por esse motivo, foram abandonadas as tentativas de predição para cenários menores do que cinco dias agregados.

Outra ação que trouxe grande melhoria ao desempenho dos modelos foi a execução do *Random Search* para a busca dos melhores parâmetros e hiperparâmetros para os modelos, como exposto na Estratégia de Treinamento.

Com todos esses ajustes, o desempenho dos algoritmos testados melhorou bastante, como poderá ser visto a seguir.

A tabela 6 apresenta um resumo com os melhores resultados obtidos em cada um dos grupos de receitas por cada um dos algoritmos de Redes Neurais testados e os resultados obtidos por Regressão Linear, ordenados por valor de R2 em cada grupo. São apresentados, também, os principais parâmetros e hiperparâmetros utilizados nesses modelos.

Tabela 6: resumo dos resultados por grupo e algoritmo (previsão de 5 dias)

Grupo	Modelo	Opcao	Lookback	Epocas	Batch	Neuronios	Camadas	Dropout	Ativacao	Kernel	RMSE	MAE	R2	FAC2
1	LSTM	6	10	75	5	256	2	0,3	relu	he_normal	50,537	27,958	0,911	71,189
1	MLP	6	10	50	10	64	3	0,5	relu	he_normal	54,711	33,060	0,896	70,350
1	RL	1	60	-	-	-	-	-	-	-	65,656	37,859	0,786	52,235
2	LSTM	8	30	75	5	128	2	0,2	selu	lecun_normal	21,117	13,915	0,934	89,784
2	MLP	4	30	75	5	64	1	0,5	relu	he_normal	23,488	16,279	0,918	87,914
2	RL	1	60	-	-	-	-	-	-	-	29,118	20,550	0,814	71,416
3	LSTM	6	20	75	5	512	2	0,45	relu	he_normal	22,007	11,675	0,808	96,596
3	MLP	2	10	75	20	24	2	0,2	relu	he_normal	23,171	12,566	0,785	93,706
3	RL	1	30	-	-	-	-	-	-	-	19,053	8,796	0,761	91,257
4	MLP	1	20	50	10	64	2	0,15	relu	he_normal	1,956	1,352	0,955	99,291
4	LSTM	6	10	50	20	512	1	0,3	selu	lecun_normal	2,164	1,391	0,945	99,580
4	RL	1	60	-	-	-	-	-	-	-	1,923	1,203	0,952	98,875
5	MLP	1	30	75	5	64	3	0,2	relu	he_normal	9,760	3,560	0,794	68,489
5	LSTM	3	30	75	10	128	1	0,45	relu	he_normal	9,871	2,922	0,789	76,259
5	RL	1	60	-	-	-	-	-	-	-	7,765	3,181	0,773	34,324
6	MLP	1	30	50	20	64	2	0,5	selu	lecun_normal	1,898	0,372	0,210	96,403
6	LSTM	6	10	75	10	128	2	0,55	relu	he_normal	1,879	0,450	0,206	96,364
6	RL	1	30	-	-	-	-	-	-	-	0,888	0,204	0,521	95,502

Como pode ser visto nessa tabela, o algoritmo LSTM foi o que apresentou os melhores resultados para os grupos de receita 1, 2 e 3, ao passo que o algoritmo MLP apresentou resultados melhores para os grupos 4, 5 e 6. No entanto, é possível notar, também, que as diferenças entre os dois algoritmos não foram muito significativas. Ambos os modelos de Redes Neurais, entretanto, tiveram desempenhos bem superiores aos da Regressão Linear, em todos os grupos de receitas.

Outro aspecto importante a ser considerado é o custo de treinamento de cada algoritmo, principalmente quando considerado o resultado apresentado (ou seja, a relação custo x benefício do algoritmo). A tabela 7 apresenta os tempos totais de processamento do *Random Search* dos modelos definidos para cada um desses algoritmos, rodando em uma máquina virtual Google Colab[52] com a aceleração de *hardware* por GPU[53].

Por essa tabela, podemos verificar que o custo de execução do algoritmo do modelo LSTM é muito maior do que o do modelo MLP. Em proporção, o tempo de treinamento do modelo LSTM foi mais de sete vezes o tempo exigido pelo modelo MLP. Da mesma forma, o tempo de execução do algoritmo de Regressão Linear foi, obviamente, muito inferior ao dos modelos de Redes Neurais. Seu resultado, no entanto, foi bem inferior ao desses modelos.

Tabela 7: tempo de processamento dos modelos

Modelo	Tempo de Processamento
MLP	01:58:01
LSTM	14:22:32
RL	00:05:41

Como a diferença dos resultados foi muito pequena entre os modelos MLP e LSTM, o custo de execução do treinamento pode ser considerado decisivo na escolha do modelo a ser utilizado em produção. O mesmo não pode ser dito em relação à Regressão Linear, já que seu desempenho na assertividade da predição foi muito inferior, não justificando a economia em tempo de processamento.

[52] Ambiente em nuvem para desenvolvimento de projetos em linguagem *Python*
[53] Graphics Processing Unit

Na tabela 8, temos os desempenhos finais dos três modelos utilizados, considerando a previsão da arrecadação total com a agregação dos valores previstos para os grupos em cada um dos algoritmos treinados, para os cinco próximos dias agregados.

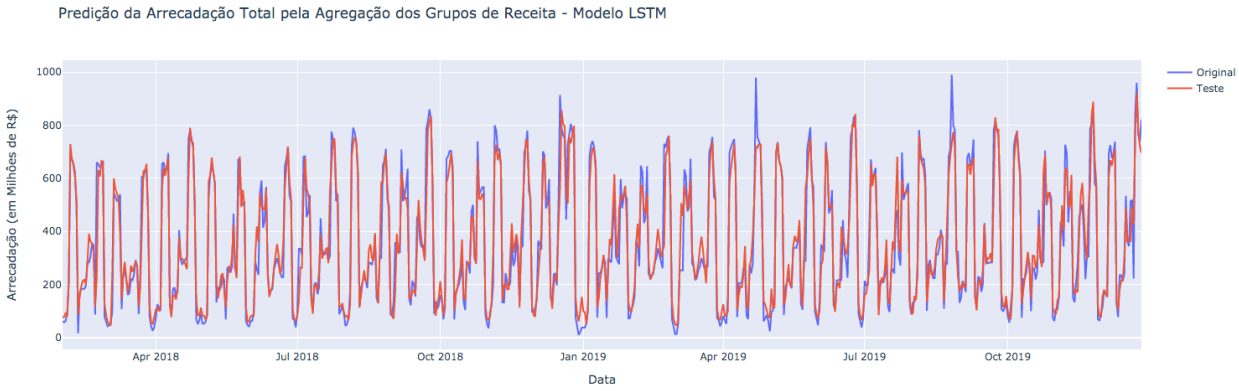
Tabela 8: resultados finais das métricas dos modelos (previsão de 5 dias)

Modelo	RMSE	MAE	R2	FAC2
LSTM	66,468	42,843	0,923	96,978
MLP	69,990	45,722	0,915	96,115
RL	106,670	72,944	0,802	93,534

Por essa tabela, observamos que o modelo que apresentou melhor desempenho global com a agregação das previsões de receitas foi o modelo LSTM, com menores valores de RMSE e MAE e resultados superiores para R2 e FAC2. No entanto, mais uma vez é possível se notar que a diferença do desempenho do modelo LSTM para o modelo MLP foi bastante pequena, algo a se considerar na escolha final do modelo a ser colocado em produção. Nota-se, também, que ambos os modelos tiveram resultado bastante superior ao obtido através da técnica de Regressão Linear.

A figura 9 mostra a plotagem dos valores previstos (com *label*[54] 'teste') versus valores reais (*label* 'original') da arrecadação total no modelo LSTM.

Figura 9: plotagem do real versus predito do modelo LSTM (previsão de 5 dias)

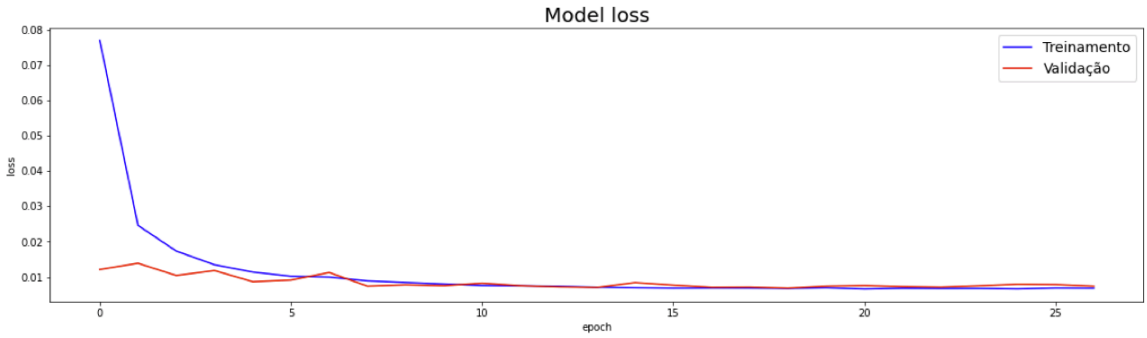


Esse gráfico confirma o bom resultado obtido nos valores de arrecadação tributária para um cenário de cinco dias através do modelo LSTM.

[54] Rótulo

A figura 10, por sua vez, mostra o gráfico do desempenho do treinamento obtido em termos de *loss* pelo algoritmo LSTM, após todas as otimizações efetuadas, sobre os dados de treinamento e validação. Por este gráfico, podemos concluir que modelo está bem ajustado, não havendo a ocorrência de *overfitting* ou *underfitting*, indicando que o modelo é capaz de generalizar bem para dados diferentes dos de treinamento.

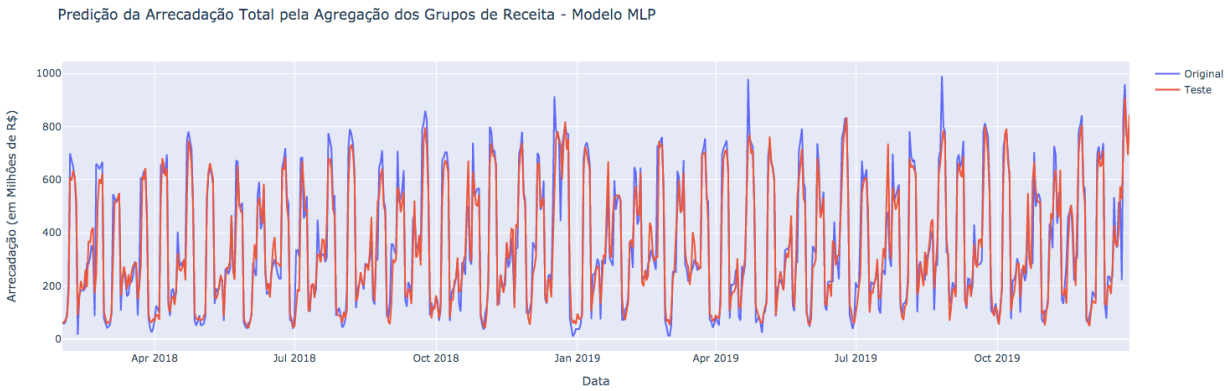
Figura 10: plotagem do comportamento do *loss* do treinamento do modelo LSTM



O gráfico da figura 11 mostra o resultado da predição da arrecadação total feita com o modelo MLP.

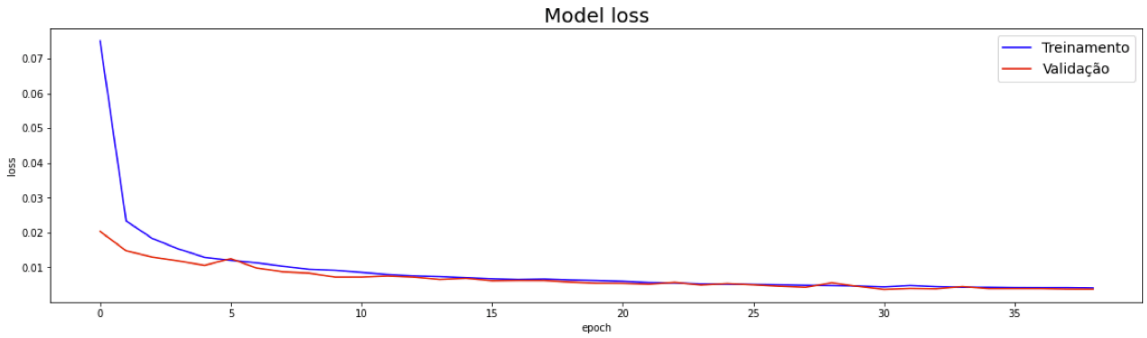
Esse gráfico confirma que o desempenho do modelo MLP é bastante próximo do apresentado pelo modelo LSTM.

Figura 11: plotagem do real versus predito do modelo MLP (previsão de 5 dias)



A plotagem do gráfico de treinamento do modelo MLP, apresentado na figura 12, também demonstra a inexistência de *undefitting* ou *overfitting* considerável, também indicando uma boa capacidade de generalização do modelo sobre os dados de validação.

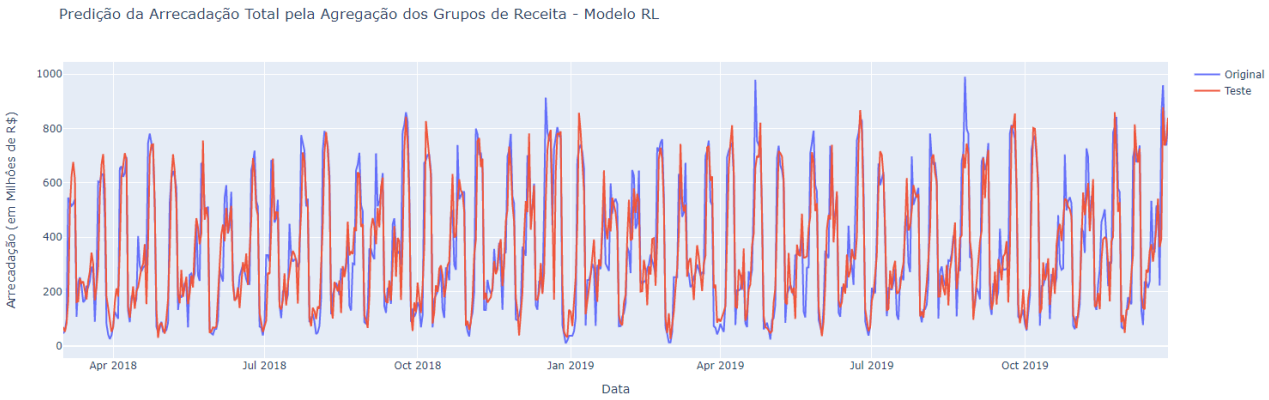
Figura 12: plotagem do comportamento do *loss* do treinamento do modelo LSTM



Para efeito de comparação, foi executado o treinamento e a predição dos valores de arrecadação com o uso da técnica de Regressão Linear.

Como pode ser visto no gráfico da figura 13, apesar de também ter apresentado um resultado bastante satisfatório, é possível notar que o modelo de Regressão Linear teve um desempenho consideravelmente inferior aos das técnicas de Redes Neurais.

Figura 13: plotagem do real versus predito por Regressão Linear



Um detalhe importante a se notar é que o único grupo que não teve bons resultados de R2 em nenhum dos modelos foi o grupo 6. Isso se explica por se tratar de um grupo em que as receitas não apresentam nenhum padrão de temporalidade ou previsibilidade, uma vez que se trata de valores de multas e taxas. Curiosamente, os valores das métricas de MAE, RMSE, e FAC2 encontrados neste grupo foram bastante razoáveis, talvez pela pouca amplitude dos valores deste grupo. Os resultados ruins desse grupo, no entanto, não

trouxeram prejuízo à qualidade da predição agregada total final, já que ele representa menos de 0,5% da arrecadação total do Estado.

Vale lembrar mais uma vez que, conforme já exposto, a previsão individual de cada um dos próximos cinco dias, que foi cogitada no início do trabalho, acabou tendo que ser definitivamente descartada, em função do alto grau de imprevisibilidade que as receitas apresentam nessa granularidade, prejudicando o resultado das predições. No seu lugar, foi utilizada unicamente a previsão agregada dos próximos cinco dias, com a qual os modelos aplicados atingiram resultados bastante animadores.

10. CONCLUSÃO

Este trabalho demonstrou a aplicabilidade e a viabilidade da utilização de modelos de Redes Neurais Artificiais para a previsão das receitas tributárias diárias em um cenário de previsão de cinco dias à frente.

Como apresentado na tabela 8, a aplicação dos modelos de aprendizagem de máquina proporcionou resultados de R2 e FAC2 superiores a 90% (R2 = 0,923 e FAC2 = 96,978), no caso do algoritmo LSTM. Os resultados do algoritmo MLP ficaram também muito próximos desses valores e com um tempo de processamento menor.

Os resultados obtidos por esses algoritmos superaram bastante as expectativas que tínhamos quando do início do trabalho, principalmente se considerarmos o alto grau de imprevisibilidade e variabilidade que muitas das receitas que compõem a arrecadação tributária apresentam em sua série histórica, e atingiram o patamar que a área de negócios reconhece como o de um bom desempenho na previsão da arrecadação diária. Os modelos de Redes Neurais também apresentaram desempenho muito superior ao da previsão por Regressão Linear, que foi utilizada como referência.

Os bons resultados obtidos nos ensaios realizados demonstraram que a solução proposta tem grande potencial de se transformar em uma ferramenta efetiva no trabalho da Secretaria da Fazenda do Estado da Bahia.

Em função da pequena diferença na assertividade entre os modelos MLP e LSTM nos testes que foram feitos e considerando que este último tem um custo computacional de treinamento bem maior do que o primeiro, pode-se considerar que o Modelo MLP é o mais indicado para utilização prática dessa solução. Apesar disso, todos os modelos foram salvos e estão prontos para serem testados em ambiente de produção, depois de retreinados com todo o conjunto dos dados.

Apesar dos bons resultados atingidos, algumas técnicas que foram cogitadas no início do desenvolvimento do trabalho tiveram que ser descartadas, por limitações de recursos e pelas dificuldades que acabaram sendo impostas pela alteração da rotina diária em função da pandemia do COVID-19[55].

Dentre as técnicas que não puderam ser aplicadas, estão a utilização de *wavelets*[56] e de modelos híbridos de Inteligência Artificial, com o uso simultâneo de mais um modelo de Redes Neural. É possível que a utilização dessas técnicas pudesse aumentar ainda mais a qualidade da previsão atingida pelos modelos explorados e fica como sugestão a exploração dessas técnicas em trabalhos futuros. Outra técnica que poderia ter sido aplicada é a da Regressão Polinomial, que possivelmente também apresentasse resultados interessantes e também a exploração dessa técnica fica como sugestão para exploração em trabalhos futuros.

[55] *COrona VIrus Disease* 19: doença causada pelo coronavírus denominado SARS-CoV-2
[56] Onduleta ou ondaleta: função capaz de decompor e descrever uma série de dados

REFERÊNCIAS

Bahia, Governo do Estado. “Demonstrações Contábeis Consolidadas do Estado”**.** Exercício de 2019. Salvador: Secretaria da Fazenda do Estado da Bahia, 2020

Livieris, I.E., Pintelas, E. e Pintelas, P. “A CNN–LSTM model for gold price time-series forecasting”. Neural Computing & Applications, 2020.

Brownlee, J. “Deep Learning for Time Series Forecasting. Predict the Future with MLPs, CNNs and LSTMs in Python”. Machine Learning Mastery, 2018.

Sami, I. e Nazir, K. “Predicting Future Gold Rates using Machine Learning Approach”. International Journal of Advanced Computer Science and Applications, 2018.

Salis, V., Kumari, A. e Singh, A. “Prediction of Gold Stock Market Using Hybrid Approach”, 2019.

Persio, L. e Honchar, O. “Artificial Neural Networks architectures for stock price prediction: comparisons and applications”, International Journal of Circuits, Systems and Signal Processing, Volume 10, 2016.

Siami, S., Tavakoli, N. e Siami, A. “A Comparison of ARIMA and LSTM in Forecasting Time Series”, 2018.

Fischer, T. e Krauss, C. “Deep Learning with Long Short-Term Memory Networks for Financial Market Predictions”, European Journal of Operational Research, Volume 270, Issue 2, 2018.

Brownlee, J. “4 Strategies for Multi-Step Time Series Forecasting”. Machine Learning Mastery, 2020.

Vinogradov, G. “Time series forecasting strategies”. Towards Data Science, 2020.

Contreras, J. e Cribari-Neto, F. “Previsão de arrecadação do ICMS através de Redes Neurais no Brasil”, 2005. Dissertação (Mestrado), Programa de Pós-Graduação em Estatística, Universidade Federal de Pernambuco, Recife, 2005.

Carmo, M., Komati, K. e Boldt, F. “Previsão de Receitas de ICMS do Estado do Espírito Santo através de Seleção de Características em Cascata e Técnicas de Aprendizado de Máquina”, Anais do XVI Encontro Nacional de Inteligência Artificial e Computacional, Porto Alegre: Sociedade Brasileira de Computação, 2020.

Freitas, F., Ciarelli, P. e Souza, A. “Previsão da Arrecadação Federal com Redes Neurais”, Anais do IX Congresso Brasileiro de Redes Neurais / Inteligência Computacional (IX CBRN), Vitória, 2009.