

Inspere
Ciências Econômicás

Kevin Santos Saraiva

**PONTOS DE DIVERGÊNCIA E COMPLEMENTARIEDADE ENTRE REDES
NEURAI S EM RELAÇÃO AOS MODELOS AUTORREGRESSIVOS**

São Paulo

2022

Insper
Ciências Econômicas

Kevin Santos Saraiva

**PONTOS DE DIVERGÊNCIA E COMPLEMENTARIEDADE ENTRE REDES
NEURAIS EM RELAÇÃO AOS MODELOS AUTORREGRESSIVOS**

Trabalho de conclusão de curso apresentado ao
programa de graduação em Ciências
Econômicas no Insper.

Orientador: André Filipe de Moraes Batista

São Paulo

2022

RESUMO

Esse artigo busca explorar e explicar as principais técnicas preditivas no campo das séries temporais. Além de revisar toda a literatura estatística do tema, serão introduzidos conceitos de aprendizado de máquina através de algoritmos preditivos – principalmente as redes neurais.

A partir dessa base teórica, será posto em prática a capacidade preditiva destas ferramentas a fim de avaliar suas principais vantagens e desvantagens. A partir da validação cruzada e dos resultados obtidos, a última etapa do estudo será avaliar pontos de convergência e de complementariedade dos modelos a fim de recriar um processo mais robusto de predição para a base em questão.

Palavras-chave: algoritmo de redes neurais, algoritmo preditivo, modelo autorregressivo, econometria

ABSTRACT

This article seeks to explore and explain the main predictive techniques in the field of time series. In addition to reviewing the entire statistical literature on the subject, machine learning concepts will introduce behind predictive algorithms – neural networks.

From this theoretical basis, the predictive capacity of these tools will put into practice to assess their main advantages and disadvantages. From the cross-validation analysis and the results obtained, the last stage of the study will be to evaluate points of convergence and complementarity of the models to recreate a more robust prediction process for the base in question.

Keyword: algorithm neural networks, predictive algorithm, autoregressive model, econometric

SUMÁRIO

| | |
|---|-----------|
| 1. INTRODUÇÃO | 7 |
| 1.1. Motivação e importância do estudo..... | 7 |
| 1.2. Descrição e contextualização do tema..... | 7 |
| 1.3. Objetivos | 8 |
| 1.3.1. Objetivo Geral | 8 |
| 1.3.2. Objetivo Específico..... | 8 |
| 1.4. Apresentação do desafio de predição..... | 9 |
| 2. REVISÃO DA LITERATURA..... | 11 |
| 2.1. Séries temporais | 11 |
| 2.1.1. Processo estocástico e características..... | 11 |
| 2.1.2. Estacionariedade..... | 12 |
| 2.2. Modelos Autorregressivos..... | 13 |
| 2.2.1. Processo MA (Moving Average ou Média Móvel)..... | 13 |
| 2.2.2. Processo AR (Autorregressivos) | 13 |
| 2.2.3. Processos ARMA, ARIMA e SARIMA | 14 |
| 2.3. Redes Neurais em Séries temporais..... | 14 |
| 3. ANÁLISE EXPLORATÓRIA..... | 16 |
| 3.1. Definição dos arquivos..... | 16 |
| 3.2. Correção da base..... | 17 |
| 3.3. Análise gráfica e distributiva..... | 18 |
| 4. METODOLOGIA PREDITIVA..... | 20 |
| 4.1. ABORDAGEM ECONOMETRICA..... | 20 |
| 4.2. ABORDAGEM DE MACHINE LEARNING | 26 |
| 4.2.1. Apresentação das Abordagens | 26 |

| | | |
|-----------|--|-----------|
| 5. | RESULTADOS E CONSIDERAÇÕES FINAIS | 28 |
| 6. | REFERÊNCIAS BIBLIOGRÁFICAS | 30 |
| 6.1. | PÁGINA DO DESAFIO..... | 30 |
| 6.2. | OUTRAS REFERÊNCIAS..... | 30 |

1. INTRODUÇÃO

1.1. MOTIVAÇÃO E IMPORTÂNCIA DO ESTUDO

Em uma nova época em que os dados são utilizados como ativos financeiros para empresas, base criativa para pesquisa e desenvolvimento de novos produtos e principalmente uma fonte de riqueza, a capacidade de predição de modelos fica cada vez mais importante. O artigo buscará explorar o que existe de vanguarda de processos preditivos não-lineares em séries temporais, tanto na abordagem econométrica quanto na ótica da ciência de dados.

1.2. DESCRIÇÃO E CONTEXTUALIZAÇÃO DO TEMA

Nesse artigo, será feito um *cross-validation* entre processos autorregressivos na modelagem preditiva de séries temporais e algoritmos de ciências de dados, para então ser possível levantar os principais pontos de vantagem entre cada método e as oportunidades de complementariedade e robustez do modelo. Dito isso, antes, é preciso introduzir alguns conceitos importantes.

Uma série temporal é um conjunto de dados que estão ordenados por tempo, ou seja, na base de dados estudada precisa ter pelo menos uma variável dedicada para explicitar algum período temporal. Portanto, um exemplo de série temporal é uma base com o registro de denúncias em uma delegacia na cidade de São Paulo no ano de 2021.

Os dados temporais precisam ser tratados de forma distinta quanto a dados com “cortes transversais”, isto é, dados que são de um momento específico do tempo, como uma “fotografia”. Imediatamente, ao se trabalhar com séries temporais, não é possível randomizar os dados sem agrupá-los por período - na verdade o conceito de amostra aleatória de dados “cross-section” precisa ser adaptado, uma vez que não é possível voltar no tempo e refazer aleatoriamente o processo (Wooldridge, 2011).

Esse exemplo é apenas uma das particularidades das séries temporais na construção de modelos preditivos, e por isso que é compreensível a existência abordagens específicas para fazer a regressão esse tipo de dados, como modelos estáticos ou de defasagens (DDF). Nesse artigo iniciaremos a discussão com modelos autorregressivos de médias móveis – ARMA e ARIMA, que são uma combinação de

dois outros processos estocásticos estacionários: processos autorregressivos (AR) e processos de médias móveis (MA) (Morettin, 2011).

Por outro lado, esses modelos possuem a evidente simplificação de ser linear, o que é pode ser limitação de situações reais, que necessitam modelagem de caráter não linear, como prever o preço de uma ação, o tamanho populacional e entre outras coisas. Os modelos GARCH são exemplos de modelos multivariados que foram criados, naturalmente, para resolver tais problemas não lineares – como determinação de volatilidade de preços e gestão de ativos no mercado financeiro (Morettin, 2020). Existem também modelos ARCH (processos autorregressivos de heteroscedasticidade condicional), modelo autorregressivo com limiar (TAR) e entre outros.

Por conseguinte, todas essas soluções irão se destacar em suas determinadas aplicações e poderão ter desvantagens em outros tópicos. De outra origem, os algoritmos de redes neurais (Haykin, 2000) se fazem presentes nesse campo de estudo – como as redes neurais artificiais autorregressivas. A vantagem dessa nova abordagem no campo preditivo de séries temporais é a capacidade dos algoritmos de tratar os dados, generalizar e, através do aprendizado superlativo que a rede neural possui, se aproximar de um modelo com alto nível de acurácia.

1.3. OBJETIVOS

1.3.1. Objetivo Geral

Construir um método de validação cruzada entre os modelos autorregressivos e algoritmos de redes neurais. Com isso, levantar aprendizados e resultados coletados para apresentar vantagens e desvantagens em cada uma das abordagens.

1.3.2. Objetivo Específico

O artigo tem dois grandes objetivos: exploratório e explicativo. Primeiramente, a literatura acerca de cada abordagem será previamente estudada e compreendida. A partir de certo nível de entendimento sobre a definição e aplicação dos modelos autorregressivos e redes neurais, o segundo passo será aplicá-los em uma situação real.

No campo da ciência de dados, existem plataformas que buscam criar desafios para comunidade, e assim os profissionais da área podem aplicar seus modelos a fim

de aperfeiçoar a técnica preditiva. Para este artigo, será utilizado uma base de dados referente a um destes desafios de ciência de dados.

Escolhido o desafio, inicia-se a segunda etapa do artigo que é aplicação de cada uma das abordagens para a resolução dos problemas. Por isso, o segundo objetivo específico do artigo tem carácter explicativo, onde será evidenciado as nuances de cada modelo a fim de criar um framework comparativo.

1.4. APRESENTAÇÃO DO DESAFIO DE PREDIÇÃO

A plataforma utilizada para fornecer a base de dados bem como o desafio para ser meio pelo qual as diferentes abordagens serão testadas é o Kaggle. Esse site funciona como um hub de competições da ciência de dados onde estudantes e profissionais da área podem compartilhar experiências e resolver desafios em conjunto. Normalmente cada competição possui seu próprio fórum e regras.

A partir do Kaggle, foi escolhido um clássico problema de séries temporais aplicada ao mundo de negócios: fazer uma previsão de vendas. Mais especificamente, o título do projeto chama-se “Predict Future Sales”. Esse desafio é o projeto final do curso chamado “How to win a data science competition” fornecido pela plataforma Coursera. Sendo assim, no site da proposta será possível encontrar uma discussão aprofundada com dúvidas e questionamentos dos alunos e respostas dos professores.

A seguir, a citação direta (traduzida para o português) da proposta de desafio:

*“Nesta competição, você trabalhará com um conjunto de dados de séries temporais desafiador, composto por dados de vendas diárias, gentilmente fornecidos por uma das maiores empresas de software russas - **1C Company**.*

Pedimos que você preveja as vendas totais de cada produto e loja no próximo mês. Ao resolver esta competição, você poderá aplicar e aprimorar suas habilidades de ciência de dados”.

Como bem explicitado nessa proposta, é fornecida a base de dados da performance histórica de diversos produtos e assim é pedido uma modelagem de vendas de cada um dos produtos. Uma vez que o objetivo do trabalho na realidade é testar a eficácia de duas abordagens, será escolhido apenas um produto para predição de vendas. Essa decisão será evidenciada na etapa de análise exploratória dos dados, considerando aquele produto quem trazer informações relevantes e

consistente – o que será conveniente principalmente na construção do modelo econométrico.

A fim de deixar mais dinâmico o trabalho, a página de proposta do Kaggle está destacada nas referências bibliográficas, enquanto a base de dados e códigos serão disponibilizadas no anexo do trabalho.

2. REVISÃO DA LITERATURA

2.1. SÉRIES TEMPORAIS

2.1.1. Processo estocástico e características

Como já definido anteriormente, o artigo se baseará em séries temporais. Dito isso, independente de qual será a abordagem, via modelos autorregressivos ou redes neurais, é importante ter uma definição bastante clara desse tipo de dado e suas principais características.

Uma série temporal é um conjunto de dados que variam ao longo do tempo. Ou seja, pode ser uma base com dados de vendas de um ano de um produto, inflação acumulada nos últimos meses etc. Logo, é qualquer dado com indexação de tempo. A série temporal pode ser definida como um processo estocástico como $\{X(t), t \in T\}$, isto é, um conjunto de variáveis aleatórias com o tempo como indexador.

As características estatísticas mais importantes desses processos estocásticos são: média, variância e autocorrelação (Wooldridge, 2011). É muito parecido com conjuntos de dados comuns, a maior diferença se encontra na autocovariância. Para medir essa estatística é criado uma série de dados defasados em certo tempo (lagged em t) para ser medido a variação dos elementos em tempos diferentes. Por isso, essa estatística chama-se de “autocovariância”, pois é a medida de variação entre a própria série, porém, defasada no tempo. A autocovariância de Y_t e $Y_{(t-1)}$ da mesma série pode ser definida em:

Figura 1 – Autocovariância entre Y_t e $Y_{(t-1)}$

$$\begin{aligned} cov(Y_t, Y_{t-1}) &= \gamma_{1t} = E[(Y_t - \mu_t)(Y_{t-1} - \mu_{t-1})] = E(Y_t Y_{t-1}) - \mu_t \mu_{t-1} \\ &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} (y_t - \mu_t)(y_{t-1} - \mu_{t-1}) f_{Y_t Y_{t-1}}(Y_t, Y_{t-1}) dy_t dy_{t-1} \end{aligned}$$

Fonte: G. S. Maddala e Kajal Lahiri, 2009.

Esse processo pode ser ainda expandido para qualquer defasagem em tempo.

Figura 2 – “j-ésima” autocorrelação de Y_t .

$$\begin{aligned} \gamma_{jt} &= E[(Y_t - \mu_t)(Y_{t-j} - \mu_{t-j})] = E(Y_t Y_{t-j}) - \mu_t \mu_{t-j} \\ &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \dots \int_{-\infty}^{\infty} (y_t - \mu_t)(y_{t-j} - \mu_{t-j}) f_{Y_t Y_{t-1} \dots Y_{t-j}}(Y_t, Y_{t-1}, \dots, Y_{t-j}) dy_t dy_{t-1} \dots dy_{t-j} \end{aligned}$$

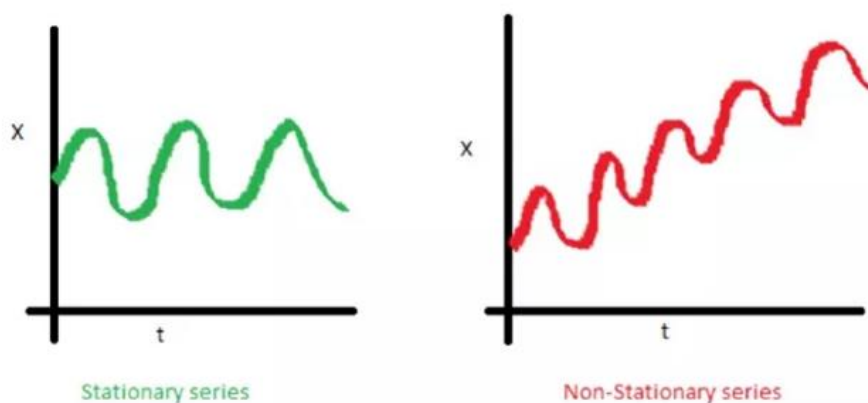
Fonte: G. S. Maddala e Kajal Lahiri, 2009.

2.1.2. Estacionariedade

Uma série temporal só pode ser considerada estacionária quando suas características estatísticas como média, variância e autocovariância são constantes em todo o período e em qualquer seção.

Qualquer outra série temporal não-estacionária são aquelas que apresentação padrões de comportamentos a depender da data (sazonalidade), ciclos, tendências etc. Normalmente as séries temporais aplicadas a econômica como preço, inflação, vendas etc. são não-estacionárias.

Figura 3 – Series temporais estacionárias e não estacionárias



Fonte: Medium. Analytics Vidhya.

O grau de estacionariedade define a característica principal do processo estocástico e qual seria a melhor abordagem para descrevê-lo ou prever futuros pontos. O processo de ruído branco (White Noise) é definido por ser o processo estacionário independente e identicamente distribuídos, ou seja, sua média e variância é constante em todo o período, sem correlação de seus elementos (MADDALA, 2009).

Caso se confirme que nem a média e nem a autocovariância estão correlacionados com a data, esse processo pode ser considerado de fracamente estacionário ou estacionário. Portanto, um ruído branco sempre será estacionário, mas nem todo processo estocástico estacionário pode ser considerado um ruído branco. Normalmente, na literatura é mais comum ver exemplos estacionários com a característica de serem processos fracamente estacionários (MADDALA, 2009).

2.2. MODELOS AUTORREGRESSIVOS

2.2.1. Processo MA (Moving Average ou Média Móvel)

O processo de média móvel pode ser definido como uma série temporal com o valor de Y_t ser construído a partir de uma soma ponderada (média) dos valores mais recentes da série temporal original.

Uma característica relevante desse processo, é que para um MA(1), isto é, um processo de média móvel de primeira ordem, observa-se um componente de ruído branco, outro componente de média e um de variância (esses dois últimos constantes). Ainda mais, a autocovariância desse processo é nulo, portanto, um MA(1) é um processo estacionário (MADDALA, 2009).

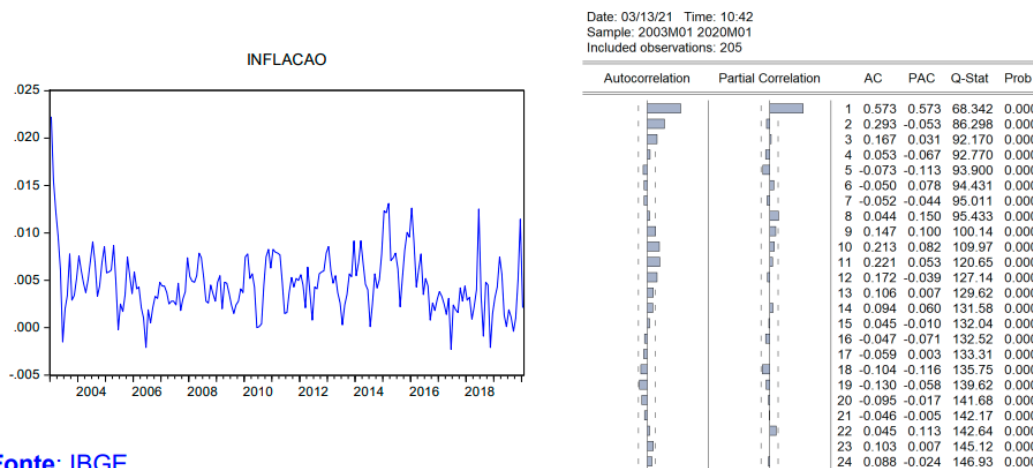
No correlogramas desse processo, observa-se que a primeira correlação tem um valor maior que zero, porém a partir da segunda correlação, a autocorrelação do MA(1) é nula, o que indica o processo ser fracamente estacionário. Sendo assim, faz sentido utilizá-lo como método de suavização de tendências, pois ele demonstra ser consistente quanto a estacionariedade (Morettin, 2011).

2.2.2. Processo AR (Autorregressivos)

O processo autorregressivo tem comportamentos diferentes quanto a sua ordem. Um AR(1) é um ruído branco com a sua raiz da equação constante e que deverá estar entre 0 e 1 em módulo (Morettin, 2011). Para demais processos autorregressivos de n ordens, como condição de existência que as n raízes estejam dentro do círculo unitário. Processos AR onde a raiz está fora do círculo unitário apresentam valores com alta memória e altamente dependentes, o que foge do conceito de estacionariedade.

O correlograma desse processo tem a característica de um “amortecedor”, isto é, com valores positivos decrescentes nas primeiras correlações, seguido por valores negativos, para que nas correlações posteriores a autocorrelação se estabilizem em valores nulos (Morettin, 2011).

A figura a seguir demonstra um correlograma de uma série com características próximas a um processo AR, e é possível perceber seu caráter “amortecedor”.

Figura 4 – Gráficos de Inflação e Correlograma

Fonte: IBGE.

Figuras: Gráfico de linha e correlograma da série mensal do IPCA (% a.m.), no período de jan/2003 a jan/2020, e respectivo correlograma.

Fonte: Dados IBGE. Correlograma produzido por Priscila Fernandes Ribeiro, retirado da aula de econometria avançada. Insuper, 2021.

2.2.3. Processos ARMA, ARIMA e SARIMA

Os processos ARMA, ARIMA e SARIMA são processos integrados que buscam controlar diferentes situações de ruído nas séries temporais.

O ARMA é um processo misto, com um polinômio de definição com um componente autorregressivo e outro polinômio de média móvel. Sendo assim, como a condição de existência todo processo ARMA deverá apresentar suas raízes fora do círculo unitário.

Já o processo ARIMA é a série integrada com erros de estacionariedade que pode ser modelada por um componente autorregressivo e outro de média móvel. É utilizado para modelar a série integrada quando possui raiz unitária ao invés de utilizar o processo ARMA comum. O tratamento dado a esse processo fazer diferenças sucessivas a fim de corrigir o erro de estacionariedade.

O processo SARIMA, por fim, busca controlar um processo com componentes autorregressivos e de média móvel com erros de estacionariedade e apresentação de padrões e ciclos (sazonalidade).

2.3. REDES NEURAS EM SÉRIES TEMPORAIS

O machine learning baseado em células nervosas, surge na década de 40 e passa por avanços e discussões ao longo de quatro décadas até ser proposto por Werbos o conceito de retropropagação ou *Backpropagation*, uma técnica que

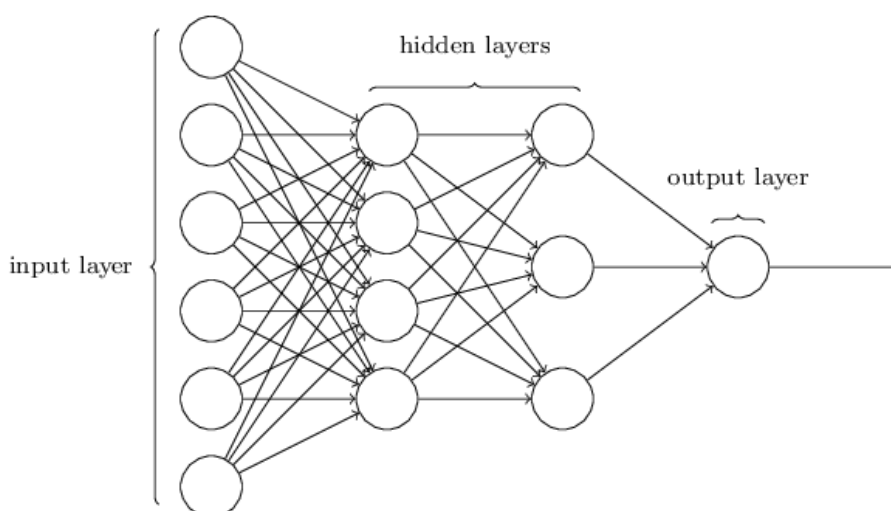
possibilitou aplicações da teoria de redes neurais, que combina os conceitos e capacidades de conexões dos neurônios entre si com engenharia e matemática para criação de modelos como previsões de séries temporais econômicas.

Para realizar essas aplicações é utilizado um conjunto de neurônios artificiais que são projetados na lógica de funcionamento de um neurônio humano, na definição de Ramos (2010): “*um neurônio artificial é, essencialmente, um conjunto de entradas e uma saída com um centro de processamento, que obedece a uma regra ou função matemática - função de ativação.*”. Para ativação ou inativação desse neurônio, as entradas são multiplicadas individualmente por um peso e depois somadas, chegando assim ao valor net que quando confrontado com o valor limite de ativação, poderá ser ativado ou não.

Esse valor limite de ativação do neurônio é dado pela função de ativação, que pode ter diversas formas, desde uma função linear até funções logísticas, as chamadas de “squashing” que são utilizadas para imitar em maior nível os neurônios biológicos.

Os neurônios podem ser agrupados em grupos, que a depender da abordagem de cada modelo pode ter mais profundidade (vertical) ou grupos de camadas (horizontal – camadas ocultas). A abordagem de “Deep Learning”, mais comum nas redes neurais contemporâneas trazem esse pressuposto de ter menos camadas ocultas e serem mais verticais (HAYKIN, 2010).

Figura 5 – Exemplo de camadas e profundidade de uma rede neural



3. ANÁLISE EXPLORATÓRIA

3.1. DEFINIÇÃO DOS ARQUIVOS

Antes de explorar os dados, é importante primeiro entender o que cada base significa. Para isso, além da própria exploração dos dados, foi utilizado o guia dos dados disponíveis no Kaggle.

A seguir a lista das bases disponibilizadas e suas definições.

sales_train.csv – a base que é chamada de “treino”, possuem dados diários de vendas de janeiro/2013 a Outubro/2015.

test.csv – o arquivo de teste. O desafio propunha a previsão de vendas para o mês de Novembro de 2015, inicialmente.

sample_submission.csv – define o formato correto por produto. Não será utilizado pois apenas carrega dados técnicos.

items.csv – dados adicionais dos produtos, como nome e categoria que pertence.

item_categories.csv - tabela dicionário que carrega os nomes das categorias de cada produto.

shops.csv - tabela dicionário que carrega os nomes de cada loja.

Normalmente, o desafio disponibiliza os dados de treinamento e uma base de “teste” vazia, assim o modelo é treinado em uma única base e ninguém tem acesso aos dados que serão utilizados para avaliar o modelo (teste). No entanto, para que seja viável a comparação dos modelos nesse artigo, a base de teste será feita a partir do último mês presente na base de treino.

Ou seja, o mês de outubro de 2015 será retirado da base de treino e os modelos deverão ser ajustados para prever a tendência de vendas para esse mês. Após a geração de previsões, esses dados serão utilizados para avaliar o resultado de cada modelo.

Entendida as bases, todas serão unificadas em apenas um dataset, composto com as seguintes colunas:

item_id– identificador numérico único para o produto

shop_id – identificador numérico único para a loja

item_category_id– identificador numérico único para a categoria do produto

item_name – nome do produto

shop_name – nome da loja

item_category_name – nome da categoria do produto

item_cnt_day – número de produtos vendidos.

item_price – preço do item vendido naquela data

date – data (a base está em formato diário)

3.2. CORREÇÃO DA BASE

A base precisa passar por uma minuciosa análise de consistência antes de ir para as próprias etapas. Basicamente os dados serão convertidos no formato adequado e nulos ou valores inválidos serão descartados.

Após possuir a base limpa de valores indesejados, é feita uma última adaptação do projeto inicial: ao invés de haver previsão de vendas para todas as categorias, será feita apenas para um único produto.

O motivo pelo qual essa escolha se faz necessária é que o modelo econométrico a ser utilizado já será muito simplificado e carregar informações de vendas de produtos distintos tende a gerar mais ruído ao invés de agregar ao modelo. Uma vez que o desafio só disponibilizou uma única variável explicativa (preço), a demanda desse produto será modelada a partir da demanda passada (lag) e preço, sendo assim, devido a simplicidade do modelo, pode haver ruído na predição se houver variáveis explicativas de outros produtos de outras categorias.

O mesmo racional vale para loja. Inicialmente, a base está dividida em produto x loja x dia. Não faz sentido buscar acurácia em diversas lojas distintas utilizando apenas duas variáveis como variáveis preditoras. Por isso, os dados serão agrupados a nível de produto x dia. Portanto, para iniciar a modelagem haverá uma base com as vendas diárias por produto e o objetivo será buscar projetar as vendas de uma única categoria de produtos no mês de outubro de 2015 (consolidado).

Para escolher a categoria dos produtos a ser prevista, o critério de seleção foi encontrar a categoria com mais amostras: mais produtos e mais dias com dados válidos. A categoria escolhida foi de id número 40, que segundo a tabela dicionário da empresa russa “1C Company” essa categoria se chama “Кино - DVD”, isto é, categoria de Filmes/DVD.

A tabela abaixo evidencia as categorias com mais quantidades de amostras, seja em número de produtos, seja em dias de vendas por produto (restrito a um único mês):

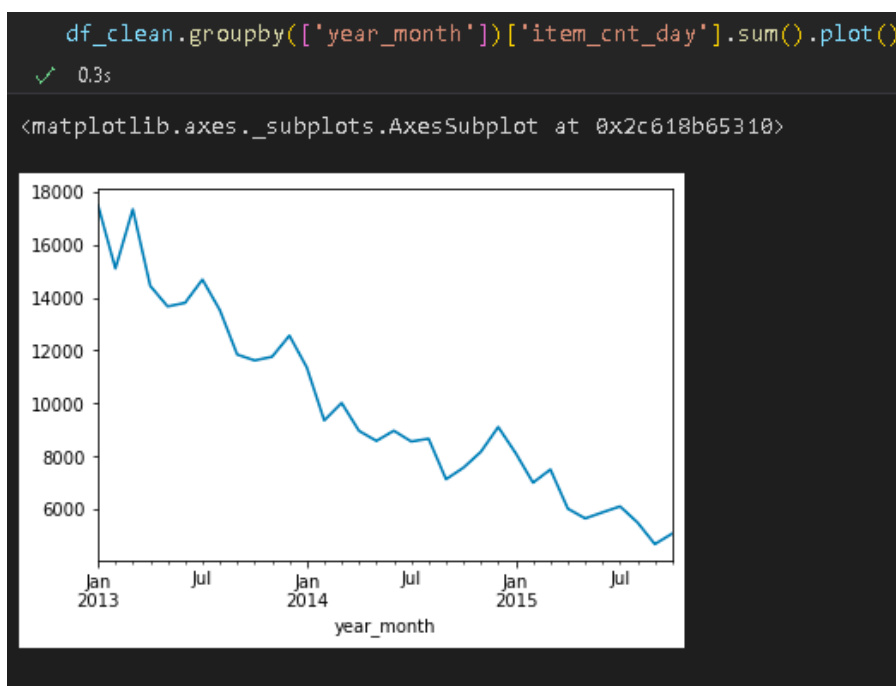
Figura 6 – Tabela de categorias com as maiores amostras

| item_category_id | qtd_amostra | item_id |
|------------------|-------------|---------|
| 40 | 4948 | 34 |
| 55 | 4311 | 34 |
| 37 | 2273 | 34 |
| 30 | 1664 | 33 |
| 20 | 1395 | 24 |

Fonte: autoria própria.

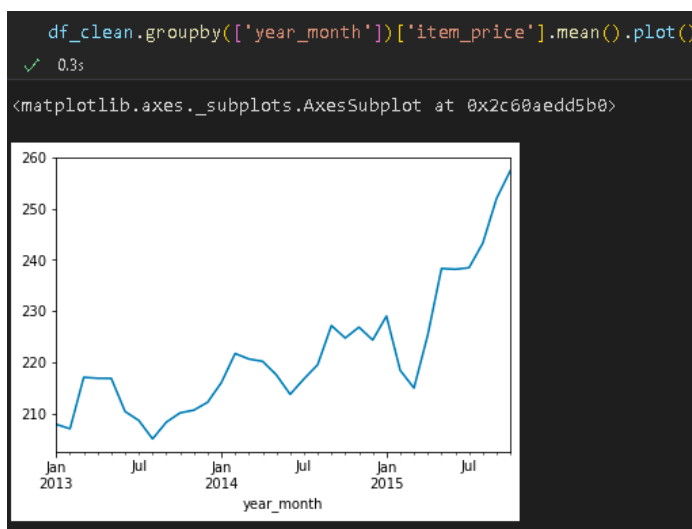
3.3. ANÁLISE GRÁFICA E DISTRIBUITIVA

Após todo o processo de filtragem de dados, o dataset limpo agora é composto apenas por produtos de Filmes/DVD, contendo dos valores de venda média e preço médio por dia de janeiro de 2013 a setembro de 2015. Definido isso, pode-se observar os seguintes gráficos:

Figura 7 - Vendas mensais totais de filmes

Fonte: autoria própria.

Figura 8 - Preços médios mensais de filmes



Fonte: autoria própria.

Sendo assim, percebe-se que ao longo do tempo, enquanto as vendas de filmes diminuíram, os preços permaneceram em uma constante ascensão. A uma segunda interpretação dos gráficos, também fica evidente que o preço não é o principal componente que explica a demanda, uma vez que entre os anos de 2013 e 2014 a média se manteve em um mesmo patamar mesmo com as variações mensais. No entanto, a inclinação das vendas de filmes continuou constante: decrescente.

Pode-se afirmar, portanto, que a variável de preços só vai se fazer relevante a partir de 2015 – antes disso, a demanda só pode ser modelada pelo seu autorregressor. Essa constatação não é uma limitação para o modelo em si, uma vez que a predição de séries temporais pode ter a característica de possuir memória, isto é, definir o comportamento futuro a partir de choques passados (Wooldridge, 2011), sendo que os dados mais recentes têm mais impacto em relação a choques muito antigos. Na verdade, para que os modelos aqui construídos possam sucesso será necessário confirmar essa hipótese quando o tipo de série temporal aqui estudada.

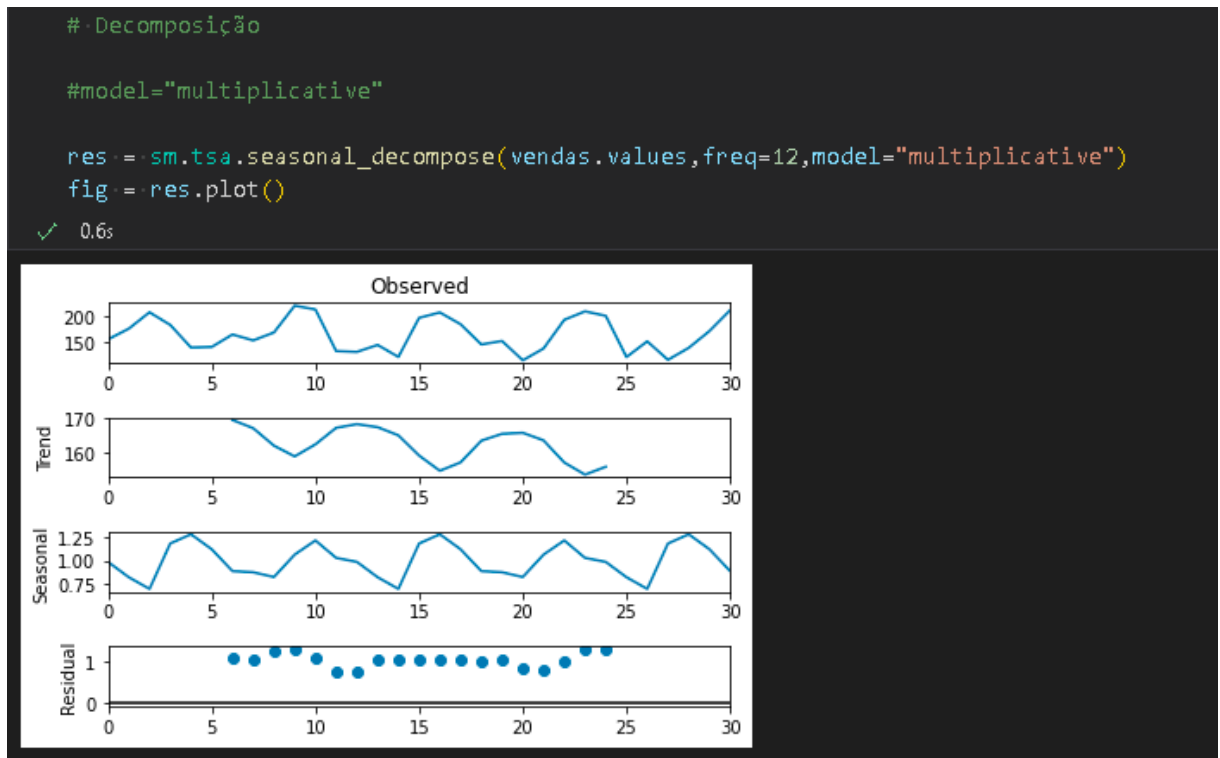
De acordo com o gráfico 1, é possível perceber uma tendência nos dados, logo fica descartado a possibilidade de que esse a série temporal seja apenas uma série estacionária.

4. METODOLOGIA PREDITIVA

4.1. ABORDAGEM ECONOMÉTRICA

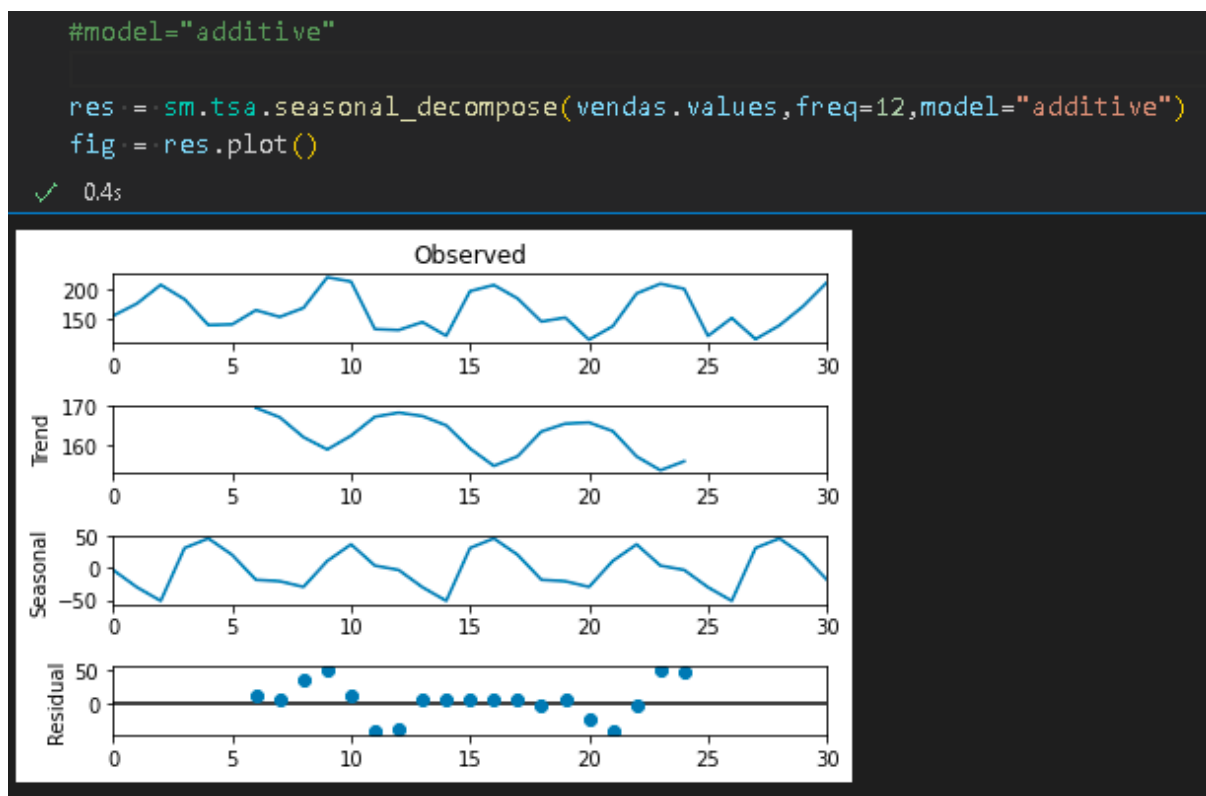
A primeira consideração a ser feita sobre a base de dados é utilizar um método que busca decompor da série temporal, nos seus componentes de tendência, sazonalidade e resíduos. Essa função de decomposição possui duas abordagens (multiplicativa e aditiva) – que nesse exemplo não fez tanta diferença.

Figura 9 – Função de decomposição



Fonte: autoria própria.

Figura 10 – Função de decomposição modelo aditivo



Fonte: autoria própria.

Ao observar os gráficos, percebe-se um ciclo próximo de cinco meses e uma tendência de decrescimento. Esses atributos também são perceptíveis no gráfico da série temporal. O fator inusitado é o gráfico de resíduos, que deveria se comportar como ruído branco após a decomposição, mas em nenhum dos casos se comporta assim. Desta forma percebe-se que talvez o modelo tenha limitação no seu poder preditivo, uma vez que ainda apresenta erros de estacionariedade.

A próxima etapa para construção do modelo é a verificação da raiz unitária. Naturalmente, como o modelo apresenta sazonalidade, é esperado que o modelo original não seja bem-sucedido no teste ADF, ou simplesmente teste Dick-Fuller, um dos testes mais comuns de raiz unitária (Morettin, 2011).

Figura 11 – Teste ADF

```
def teste_adf(serie_temporal):
    print('Teste ADF:')
    dftest = adfuller(serie_temporal, autolag='AIC')
    dfoutput = pd.Series(dftest[0:4], index=['Resultado ADF', 'p-value', '#Lags Usados', 'Número de observações'])
    for key, value in dftest[4].items():
        dfoutput['Valor crítico (%s)'%key] = value
    print(dfoutput)
```

0.3s

```
teste_adf(vendas)
```

0.1s

| Teste ADF: | |
|-----------------------|-----------|
| Resultado ADF | -1.463369 |
| p-value | 0.551548 |
| #Lags Usados | 6.000000 |
| Número de observações | 24.000000 |
| Valor crítico (1%) | -3.737709 |
| Valor crítico (5%) | -2.992216 |
| Valor crítico (10%) | -2.635747 |

Fonte: autoria própria.

Confirmado a hipótese de que a série temporal original não teria como passar no teste ADF, foram feitas novas transformações nessa série de dados: para cada tempo t , foi retirado o componente de tendência “sazonal”. Esse ajuste foi feito considerando uma sazonalidade de 1 mês, 5 meses e 12 meses.

Figura 12 – Retirada da sazonalidade

```
Retirada de Sazonalidade
```

```
intervalo=12 ## Suposição que sazonalidade ocorre em 12 meses
diff_vendas = list()
for i in range(intervalo, len(vendas)):
    value = vendas[i] - vendas[i - intervalo]
    diff_vendas.append(value)

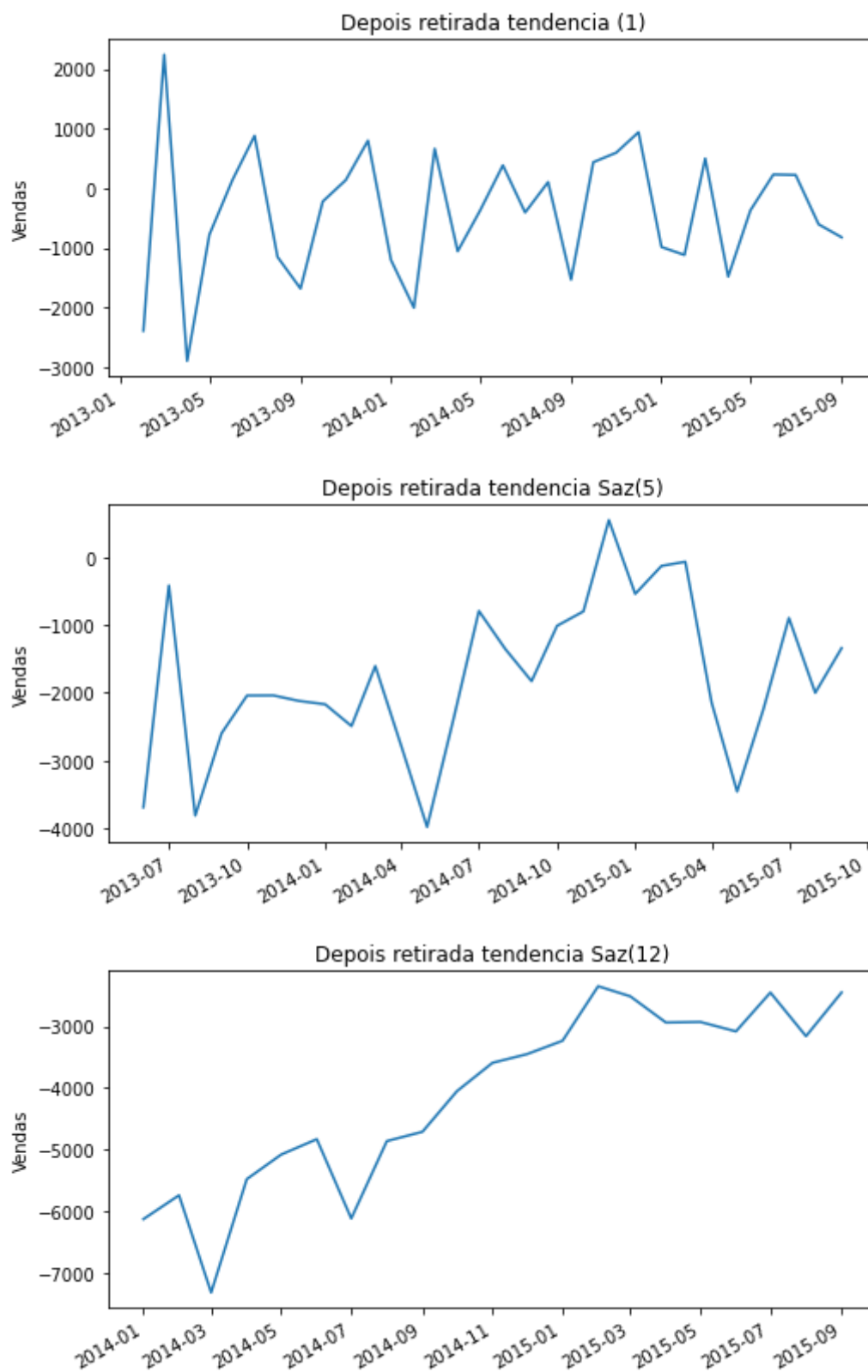
vendas_tend_12=vendas.reset_index()[intervalo:].copy()
vendas_tend_12['vendas']=diff_vendas
vendas_tend_12=vendas_tend_12[['year_month', 'vendas']].set_index('year_month').astype('float')

plot_grafico(vendas_tend_12, 'Depois retirada tendencia Saz('+str(intervalo)+')')
```

Fonte: autoria própria.

A série temporal ficou neste formato para cada cenário.

Figura 13 – Gráficos de retirada de tendência



Fonte: autoria própria.

Após esse modelo, foram feitos novamente os testes ADF. Como já observado anteriormente, não houve raiz unitária na situação em que a correção de sazonalidade de 5 meses (e também para uma tendência de 1 mês).

Figura 14 – Tabela ADF para 1 mês

```

Teste ADF:
Resultado ADF          -5.637683
p-value                0.000001
#Lags Usadaos         1.000000
Número de observações 30.000000
Valor crítico (1%)    -3.669920
Valor crítico (5%)    -2.964071
Valor crítico (10%)   -2.621171
  
```

Fonte: autoria própria.

Figura 15 – Tabela ADF para 5 meses

```

Teste ADF:
Resultado ADF          -3.880410
p-value                0.002184
#Lags Usadaos         0.000000
Número de observações 27.000000
Valor crítico (1%)    -3.699608
Valor crítico (5%)    -2.976430
Valor crítico (10%)   -2.627601
  
```

Fonte: autoria própria.

Figura 16 – Tabela ADF para 12 meses

```

Teste ADF:
Resultado ADF          -0.957089
p-value                0.768592
#Lags Usadaos         1.000000
Número de observações 19.000000
Valor crítico (1%)    -3.832603
Valor crítico (5%)    -3.031227
Valor crítico (10%)   -2.655520
  
```

Fonte: autoria própria.

Por fim, é feito um loop para encontrar o melhor modelo ARMA que para esse cenário. Foram feitos dois modelos um para cada série temporal que não apresentou raiz unitária (cenário de 1 mês e cenário de 5 meses de sazonalidade).

Figura 17 – Loop de escolha do melhor ARMA

```
best_aic = np.inf
best_order = None
best_md1 = None

intervalo = range(5)
for i in intervalo:
    for j in intervalo:
        try:
            tmp_md1 = smt.ARMA(vendas_tend_5.values, order=(i, j)).fit(method='mle', trend='nc')
            tmp_aic = tmp_md1.aic
            if tmp_aic < best_aic:
                best_aic = tmp_aic
                best_order = (i, j)
                best_md1 = tmp_md1
        except: continue

print('aic: {:.65f} | order: {}'.format(best_aic, best_order))
```

Fonte: autoria própria.

Para o componente de sazonalidade de 1 mês, foi encontrado um ARMA(1,2) e o para o outro modelo, foi encontrado um ARMA(1,1).

4.2. ABORDAGEM DE MACHINE LEARNING

4.2.1. Apresentação das Abordagens

O melhor algoritmo de rede neural será escolhido a partir de uma comparação empírica de uma série de abordagens diferentes. Ou seja, serão treinados e feito uma calibração parcial de múltiplos modelos de redes neurais para encontrar aquele com melhor performance frente ao modelo econométrico.

As abordagens escolhidas foram: MLP, CNN, LSTM e CNN-LSTM. Todos esses modelos já vieram de uma biblioteca que adaptam para projeção de dados via séries temporais.

O model MLP é uma sigla para Multilayer Perceptrons, que em uma tradução livre significa um modelo simples de multicamadas. Esse modelo é bastante simples, com uma limitação de não coletar os dados de forma sequencial, mas olhar todos os dados como um todo – a adaptação desse modelo para séries temporais acontece justamente nessa parte, ele simplesmente considera os dados mais recentes como relevantes ainda que não haja ordem sequencial dos dados. Com isso, fica evidente que esse modelo não vai ser eficaz para prever uma série temporal já que o pressuposto autocorrelação e memória desse processo é quebrado quando o modelo não considera que os dados são sequenciais. O motivo pela escolha desse modelo foi porque fazer uma predição com ele retornará um piso de acurácia do que é aceitável para os outros modelos, ou seja, caso a acurácia desse modelo for de 65%, os outros modelos só serão considerados razoáveis se superarem essa hipótese “naive”.

O segundo modelo a ser considerado é o CNN, sigla em inglês para rede neural convolucional, que é outro framework simples nas redes neurais. No caso, para adaptar para o modelo de séries temporais, é possível criar uma sequência de variáveis inteiras de acordo com o dia da observação e dessa forma o modelo poderá ser calibrado considerando os dados em sequência. Ainda que seja um modelo simples, será possível gerar outro piso de acurácia, agora para modelos que levam a sequência de dados em consideração.

O modelo LSTM (Memória de Curto/Longo Prazo) já é um modelo que, diferente dos modelos CNN e MLP, a variável tempo é considerada como parâmetro, sendo assim, a formatação dos dados irá considerar automaticamente a relação de “features” e dia. Como a sigla do modelo demonstra, esse tipo de rede neural considera a memória e autocorrelação dos regressores, o que é necessário para

construir um modelo preditivo para esse tipo de dado. Esta abordagem é aquela mais simples para a predição de séries temporais, mas muito eficaz no quesito de acurácia.

Por fim, a abordagem mais refinada que será proposta é a CNN-LSTM, uma versão otimizada do padrão LSTM. Basicamente, o modelo é calibrado por blocos via a função objetiva do CNN, para que o LSTM entre em ação e encontre o melhor resultado. Essa estratégia de divisão dos dados entre dois modelos se faz necessária para evitar problemas de “overfitting” e que o modelo entre em uma otimização local.

Após gerar modelos de redes neurais convincentes, todos os modelos serão comparados com o modelo econométrico.

5. RESULTADOS E CONSIDERAÇÕES FINAIS

Os modelos econométricos acertaram a direção das vendas, seja no cenário de ajuste de 1 mês, seja no cenário de ajuste de 5 meses. O que foi percebido é que ainda que a direção esteja correta, a diferença entre os resultados é muito grande.

Figura 18 – Tabela modelo ARMA(1,2)

```

Previsão:

  year_month  vendas
0  2015-10-01  734.104345

Valor real ajustado:

  year_month  vendas
2015-10-01  399.021928
  
```

Fonte: autoria própria.

Figura 19 – Tabela modelo ARMA(1,1)

```

Previsão:

  year_month  vendas
0  2015-10-01 -2997.157672

Valor real ajustado:

  year_month  vendas
2015-10-01 -571.901741
  
```

Fonte: autoria própria.

As possíveis limitações de predição desta abordagem são: utilizar apenas uma variável preditora para estimar variáveis mais complexas, no caso, vendas de um produto – o autorregressivo possui muito ruído. Esse é outra limitação deste modelo, ainda que decomposto, o componente residual ainda não se comportava como um

ruido branco, indicando que a decomposição não conseguiu neutralizar todos os componentes não estacionários.

Quanto a abordagem de machine learning, a proposta de desenvolver quatro modelos de redes neurais diferentes se mostrou muito pesada no quesito poder computacional. Sendo assim, até o momento, não foi possível gerar modelos calibrados a tempo suficiente para realizar a comparação de resultados. De maneira geral, isso é uma limitação ao artigo, pois parte dos objetivos era apresentar uma comparação entre as duas abordagens.

6. CONCLUSÃO

Durante a confecção do artigo foi explorado a complexidade das séries temporais, modelos preditivos econométricos e modelos de redes neurais. Quanto ao objetivo exploratório, o artigo conseguiu apresentar boa parte da vasta discussão.

No entanto, na parte prática, os modelos econométricos se mostraram limitados quanto ao seu poder preditivo, enquanto os modelos de redes neurais não foram capazes de serem gerados a tempo hábil. Esse objetivo foi, portanto, considerado um fracasso.

Parte do aprendizado do trabalho, portanto, foi o dimensionamento do tempo para a confecção e treinamento dos modelos

7. REFERÊNCIAS BIBLIOGRÁFICAS

7.1. PÁGINA DO DESAFIO

KAGGLE. **Predict Future Sales**, Coursera ("How to win a data science competition"). Disponível em: <https://www.kaggle.com/competitions/competitive-data-science-predict-future-sales/overview>. Acesso em Junho de 2022.

7.2. OUTRAS REFERÊNCIAS

HAYKIN, S., **Redes Neurais Princípios e prática**, Editora Bookman 2ª ed., Porto Alegre-RS, 2000.

MORETTIN, Pedro A. **Análise de séries temporais - volume 2**. Disponível em: Minha Biblioteca, Editora Blucher, 2020.

MORETTIN, P. A., **Econometria Financeira: Um Curso em Séries Temporais Financeiras.**, 2ª ed., Edgard Blücher, 2011.

WOOLDRIDGE, J. M., **Introdução à Econometria: Uma abordagem moderna**, 1ª ed., Cengage, 2011.

G. S. MADDALA E KAJAL LAHIRI. **Introduction to Econometrics**, 4a. Edição, John Wiley & Sons Ltd., UK, 2009.

RAMOS, Alexandre Soares. **Previsões de Séries Temporais combinando modelos ARMA e Redes Neurais Artificiais**. Recife, 2010.

BROWNLEE, Jason PhD. **Deep Learning for Time Series Forecasting Crash Course**. <https://machinelearningmastery.com/how-to-get-started-with-deep-learning-for-time-series-forecasting-7-day-mini-course/>. Site: Machine Learning Mastery, 2019. Acesso em Outubro de 2022.

OLIVEIRA, Dimitri. **Deep Learning for Time Series Forecasting**. Link: <https://www.kaggle.com/code/dimitreoliveira/deep-learning-for-time-series-forecasting/notebook>. Aplicação de um desafio do Kaggle para Deep Learning, 2019. Acesso em Novembro de 2022.

TENSORFLOW Library. **Previsão de séries temporais** Link: https://www.tensorflow.org/tutorials/structured_data/time_series#convolution_neural_network. Última atualização: Janeiro de 2022. Acesso em Outubro de 2022.

YADAV, Siddharth. **Everything you can do with a time series**. Link: <https://www.kaggle.com/code/thebrownviking20/everything-you-can-do-with-a-time-series>. Aplicação de um desafio do Kaggle para séries temporais, 2018. Acesso em Agosto de 2022.

PRA, Marco Del. **Time Series Forecasting with Deep Learning and Attention Mechanism**. Link: <https://towardsdatascience.com/time-series-forecasting-with-deep-learning-and-attention-mechanism-2d001fc871fc#:~:text=Recurrent%20Neural%20Networks%20are%20the,when%20a%20applied%20to%20long%20sequences>. Site TowardsDataScience, 2020. Acesso em Agosto de 2022.