



**TS-AutoML: Auto Machine Learning para
Predição de Séries Temporais**
Classificação de séries temporais para escolha
automatizada de modelos

Victor Aquiles Soares de Barros Alencar

JUIZ DE FORA
FEVEREIRO, 2022

**TS-AutoML: Auto Machine Learning para
Predição de Séries Temporais**
Classificação de séries temporais para escolha
automatizada de modelos

VICTOR AQUILES SOARES DE BARROS ALENCAR

Universidade Federal de Juiz de Fora
Instituto de Ciências Exatas
Departamento de Ciência da Computação
Bacharelado em Ciência da Computação

Orientador: Alex Borges Vieira
Coorientador: Heder Soares Bernardino

JUIZ DE FORA
FEVEREIRO, 2022

TS-AUTOML: AUTO MACHINE LEARNING PARA PREDIÇÃO DE
SÉRIES TEMPORAIS

Classificação de séries temporais para escolha automatizada de modelos

Victor Aquiles Soares de Barros Alencar

MONOGRAFIA SUBMETIDA AO CORPO DOCENTE DO INSTITUTO DE CIÊNCIAS EXATAS DA UNIVERSIDADE FEDERAL DE JUIZ DE FORA, COMO PARTE INTEGRANTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE BACHAREL EM CIÊNCIA DA COMPUTAÇÃO.

Aprovada por:

Alex Borges Vieira
Doutor em Ciência da Computação

Heder Soares Bernardino
Doutor em Modelagem Computacional

Luciana Conceição Dias Campos
Doutora em Engenharia Elétrica

Edelberto Franco Silva
Doutor em Computação

JUIZ DE FORA
16 DE FEVEREIRO, 2022

Resumo

Séries temporais podem modelar os mais diversos fenômenos e a predição destes fenômenos é importante para diversas áreas de estudo, como predição climática ou de demanda de serviços. Fazer predições, especialmente utilizando séries temporais, necessita de diversas etapas repetitivas que podem ser automatizadas, como o tratamento dos dados ou a seleção do melhor modelo. Dessa maneira, os modelos de *Automated Machine Learning* (AutoML) tem como objetivo automatizar todas essas etapas para o pesquisador somente se preocupar com os dados coletados e a sua predição. Contudo, modelos de AutoML não são suficientemente genéricos para qualquer tipo de dado, e assim, não englobam características únicas de séries temporais que devem ser exploradas, como exemplo a sazonalidade. Dessa forma, este trabalho apresenta o TS-AutoML, modelo de AutoML voltado a séries temporais onde características e modelos voltados a séries temporais são utilizados para melhor aproveitamento das características únicas deste tipo de dado. Ademais, o trabalho explora como realizar a escolha de modelos de predição de forma automatizada e eficiente, com a utilização de modelos de classificação para determinar o algoritmo de predição que melhor se adequa aos dados de entrada. Utilizando séries temporais de diversas áreas de estudo para treinamento e teste, foram obtidos ganhos de aproximadamente 80% em relação ao tempo de processamento em detrimento de aproximadamente 40% do desempenho de predição.

Palavras-chave: Machine Learning, AutoML, Classificação, Séries Temporais.

Abstract

Time series can model the most diverse phenomenons and the prediction of these phenomenons is important for several study areas, like climate or service's demand predictions. Making predictions, especially using time series, requires several repetitive steps that can be automated, such as data processing or the best model selection. In this way, *Automated Machine Learning* (AutoML) models aim to automate all these steps, so the researcher needs only to be concerned with the data collection and its prediction. However, AutoML models are not generic enough for any type of data, like not including unique time series features that should be explored, such as seasonality. Thus, this work presents the TS-AutoML, an AutoML model focused on time series. Where features and models of time series are used to take advantage of the unique features of this type of data. Furthermore, the work explores how to perform the prediction model choice in an automated and efficient way, using classification models to find the prediction algorithm that best fits the input data. Using time series from several study areas for training and test, it was measure gains of about 80% em relation to the processing time at the expense of about 40% of the prediction performance.

Keywords: Machine Learning, AutoML, Classification, Time Series.

Agradecimentos

Dedico especialmente este trabalho ao meu eterno matemático, filósofo, professor e pai, Ricardo Alencar, que sonhou comigo em me tornar um cientista e fez de tudo para que eu alcançasse o meu sonho. A minha mãe Denise Soares, por todo amor e apoio, que com toda a sua dedicação me fez chegar aonde eu sonhava e a cada dia luta para dar o seu melhor. E ao meu irmão Felipe, que comigo cresceu e me acompanha na incessante busca pelo conhecimento.

A minha companheira Ana Beatrice, por todo o carinho e apoio nos momentos de alegria e dificuldade e que sempre me mostrou que eu consigo dar o próximo passo. Sem ela realizar este curso nem estaria em meu escopo, mas ela me ajudou até o fim.

Agradeço ao professor Alex, por me dar o meu primeiro voto de confiança e me apresentar de fato ao mundo da computação. Obrigado por sempre confiar em mim e ter me ajudado tanto durante o meu trajeto, consigo dizer que todos os ensinamentos que ele me apresentou durante nossos projetos não são possíveis de serem apresentados em sala de aula e sem eles eu não seria o profissional que estou me formando.

Agradeço ao meu amigo Lucas Pessamilio em que sem ele este trabalho não seria desenvolvido e claro por toda a ajuda e companheirismo ao longo do curso. Também agradeço ao Felipe Rooke, por desde o início me ensinar muito e participar da maior parte dos projetos em que fiz parte.

Agradeço as instituições CNPQ, FAPES e FAPEMIG por ter ajudado em tornar essa e muitas outras pesquisas possíveis. Incentivos como esses ajudam a muitos alunos a completar seus estudos e fomentarem a pesquisa acadêmica

Por fim, agradeço a Universidade Federal de Juiz de Fora e ao Departamento de Ciência da Computação. Que conjunto a professores incríveis como o Heder Bernardino e Edelberto Franco, estimulam e formam excelentes profissionais e pessoas além das salas de aula.

Conteúdo

Lista de Figuras	5
Lista de Tabelas	6
Lista de Abreviações	7
1 Introdução	8
2 Revisão Bibliográfica	12
2.1 Aprendizado de Máquina	12
2.2 Classificação	13
2.3 AutoML	14
2.3.1 Preparação dos dados e Extração de Características	15
2.3.2 Geração do Modelo de Predição	16
2.3.3 AutoML para séries temporais	17
3 Metodologia	19
3.1 Métodos Utilizados	19
3.1.1 TS-AutoML	19
3.1.2 Classificação das séries temporais	21
3.2 Processo Metodológico	22
4 Experimento	24
4.1 Base de dados	24
4.2 Classificação	25
4.3 Resultados	27
5 Conclusão	29
Bibliografia	30

Lista de Figuras

2.1	Visão geral da arquitetura de um projeto de AutoML. Figura adaptada de (HE; ZHAO; CHU, 2021).	15
2.2	Exemplo de espaço de busca. Figura adaptada de (YAO et al., 2019).	17
3.1	TS-AutoML em alto nível	19
3.2	Processo de comparação de desempenho	23

Lista de Tabelas

4.1	Base de dados de séries temporais utilizadas.	24
4.2	Modelos de predição escolhidos pelo TS-AutoML durante o processamento inicial das bases de treino.	25
4.3	Acurácia obtida por todos os modelos do AutoGluon durante o treino. . . .	26
4.4	Modelos escolhidos com e sem classificação das séries da base de teste. . . .	26
4.5	Resultados obtidos através da predição com ou sem classificação.	27

Lista de Abreviações

ARIMA	Modelo Auto-Regressivo Integrado de Médias Móveis
AutoML	Automated Machine Learning
DCC	Departamento de Ciência da Computação
LSTM	Long Short-Term Memory
MAE	Erro Absoluto Médio
RMSE	Raiz do Erro Quadrático Médio
SARIMA	Modelo Sazonal Auto-Regressivo Integrado de Médias Móveis
TFT	Temporal Fusion Transformer
TSPO	Time Series Pipeline Optimisation
UFJF	Universidade Federal de Juiz de Fora
WAPE	Erro de Porcentagem Absoluta Ponderada

1 Introdução

As séries temporais são um tipo de dados muito utilizada nos mais diversos tipos de estudos. Na economia, elas são utilizadas para descrever diversos fenômenos como, por exemplo, o produto interno bruto (PIB) de países dado o período de tempo (JAMES et al., 2012). Também podem descrever eventos naturais, como sinais sísmicos para avaliação de um possível terremoto (VAROTSOS; SARLIS; SKORDAS, 2011). Ou até a modelagem do comportamento de uso de sistemas de compartilhamento de veículos (ALENCAR et al., 2019).

As séries temporais podem ser caracterizadas por valores de observações coletadas ao longo do tempo (Warren Liao, 2005). Um exemplo seria a medição de temperatura de uma cidade ao longo de um ano. Como elas são flexíveis para as mais diversas aplicações, são grande alvo da literatura e indústria (ALENCAR et al., 2020; LAPTEV et al., 2017). Com isso, existem trabalhos que as utilizam tanto para auxiliar caracterizações de fenômenos (ALENCAR et al., 2019), quanto para a criação de modelos de predição específicos para esse tipo de dados (TAYLOR; LETHAM, 2018).

Para problemas de predição, existem uma grande diversidade de modelos que podem serem explorados pelo pesquisador. Contudo, existe a dificuldade de realizar tarefas repetitivas quando esses modelos são manipulados pelo pesquisador, tais como, o tratamento de dados, escolha de modelos e otimização de hiperparâmetros. Para otimizar esse processo, existem os modelos de *Automated Machine Learning* (AutoML). Segundo Truong et al. (2019), modelos de AutoML tem como objetivo aumentar a produtividade de pesquisadores, removendo a necessidade de realizar estas tarefas repetitivas. Com isso, o usuário desse sistema irá somente se preocupar com os dados a serem processados pelo modelo de AutoML.

Modelos de AutoML podem ser divididos em três etapas (TRUONG et al., 2019). Primeiro, é executado o pré-processamento de dados e a seleção de características (*features*). Nesta etapa inicial, os dados serão processados para estarem no formato desejado e características serão extraídas para serem utilizadas pelos modelos de predição inter-

nos. Após isso, é realizada a seleção de modelos de predição e otimização de seus hiperparâmetros. Nesta etapa as características geradas são utilizadas para treinar os modelos, que variam entre si dados os seus hiperparâmetros. Com isso, o modelo de melhor desempenho é escolhido como modelo final e, em seguida, é executada a etapa de interpretação dos resultados obtidos pelos modelos e análise de predições. Assim, são apresentadas as interpretações sobre as características selecionadas, como por exemplo a sua relevância para a predição ou do modelo escolhido, além de uma análise sobre o desempenho das predições (TRUONG et al., 2019).

Para séries temporais e os mais diversos problemas de conhecimento usando dados, a etapa de *pré-processamento de dados e seleção de características* é de grande importância para o treinamento dos modelos de predição. Especificamente, as séries temporais tem características particulares que não são encontradas em outros tipos de dados. Como exemplo, a possibilidade de decompor a série temporal em múltiplos componentes, que podem fornecer informações como tendência e sazonalidade da série (HARVEY; PETERS, 1990). Assim, são necessários métodos específicos para o tratamento desse tipo de dado para extrair de forma eficiente todas as suas características.

Para utilizar as características próprias das séries temporais, são desenvolvidos modelos voltados somente a sua predição. Um exemplo seria o modelo auto-regressivo integrado de médias móveis (ARIMA), que é a combinação de um modelo auto-regressivo (AR), um fator integrante (I) e um modelo de médias móveis (MA). Com isso, três componentes únicos de séries temporais contribuem para gerar predições de valores futuros. O ARIMA detém três parâmetros que alteram o comportamento de cada componente, que são comumente nomeados como p , d e q , que representam cada componente AR, I e MA respectivamente (COCCA et al., 2020). Em outras palavras, precisam ser escolhidos os valores que melhor se adaptam à série temporal desejada.

Além do ARIMA, existem outros modelos como o SARIMA (COCCA et al., 2020) e o Prophet (TAYLOR; LETHAM, 2018). Todos eles exploram características únicas das séries temporais e detêm diferentes parâmetros próprios. Portanto, retornamos a necessidade do pesquisador conhecer os modelos de predição existentes, saber ou testar qual modelo será mais adequado para o seu problema e, por fim, testar e avaliar todas as

possibilidades de parâmetros que se adequam às suas necessidades para, assim, escolher o melhor modelo de predição para o seu problema.

Modelos como o Prophet¹ automatizam passos do processo de predição. Assim como o ARIMA, ele também é formado por um modelo que pode ser decomposto em outros que cuidam de partes específicas da série temporal (TAYLOR; LETHAM, 2018). Além disso, ele automatiza a busca de parâmetros de sazonalidade e tendência. Ademais, oferece recursos para análise dos resultados, como avaliação das predições e geração de gráficos de cada componente da série. Portanto, pode ser dito que ele tem características de modelos de AutoML, contudo ainda tem-se parâmetros a serem indicados que podem ser explorados pelo pesquisador e ainda não pode ser usado em todos os contextos, por exemplo por séries multivariáveis, por questões de limitações de implementação.

Dessa forma, a pergunta chave deste trabalho seria: Como selecionar os modelos de predição de séries temporais de forma a automatizar os processos manuais do pesquisador de forma rápida e eficiente? Para responder essa pergunta, a proposta é utilizar a conceituação dos modelos de AutoML para agregar modelos de predição já existentes de séries temporais, conjunto a um estudo de como reduzir o tempo de busca de modelos, por meio do uso de classificação de séries temporais. Portanto, pretende-se verificar se a busca por modelos de predição de séries temporais com o uso de modelos de classificação reduz o tempo da busca exaustiva de modelos de predição que os modelos de AutoML podem executar, semelhante a busca manual de um pesquisador.

Este trabalho irá utilizar um modelo de AutoML voltado à predição de séries temporais nomeado como TS-AutoML². Ele tem como objetivo realizar a seleção do modelo de predição ideal para a série temporal de entrada, através de métricas e modelos de classificação de séries temporais. O TS-AutoML utiliza pré-processamento de dados para tornar possível o uso de diversos modelos de aprendizado de máquina, inclusive modelos que não são voltados especificamente para este tipo de problema, como exemplo os baseados em Gradient Boosting (FRIEDMAN, 2001). Ele será utilizado como uma ferramenta para realizar de forma automática a busca de modelos de predição e hiperparâmetros de forma eficiente. Nesse trabalho é feita uma comparação do uso de modelos

¹<http://facebook.github.io/prophet/>, Acesso em: 06/02/2022

²<https://github.com/TS-UFJF/AutoML>, Acesso em: 06/02/2022

de classificação, para a seleção de modelos automatizada com a seleção considerando todas as opções possíveis, em questão de velocidade e desempenho. Com isso, foram obtidos ganhos de 80% de velocidade de processamento com uma perda de 40% de desempenho para a seleção realizando a classificação, se mostrando eficiente para essa tarefa.

O restante do trabalho é formado pelos seguintes capítulos: O capítulo 2 apresenta a revisão bibliográfica do texto, com a contextualização da área e seus conceitos. O capítulo 3 apresenta os métodos e modelos utilizados, conjunto a todo o processo metodológico. Ao longo do capítulo 4 são apresentados os resultados obtidos na experimentação. Por fim, o capítulo 5 conclui o trabalho.

2 Revisão Bibliográfica

Neste capítulo é apresentada toda a revisão bibliográfica e fundamentação teórica do trabalho. Inicialmente, a seção 2.1 introduz brevemente o conceito por trás do aprendizado de máquina. Em sequência, a seção 2.2 apresenta tanto a conceituação do problema de classificação quanto a sua aplicação para séries temporais. A seção 2.3 apresenta a conceituação base de projetos de genéricos de AutoML e finaliza o capítulo com a subseção 2.3.3 apresentando as suas aplicações para séries temporais.

2.1 Aprendizado de Máquina

Aprendizado de máquina, pode ser definido como qualquer programa de computador que melhora o seu desempenho por meio do aprendizado (MITCHELL, 1997). Formalmente, tem-se a seguinte definição: “Um programa de computador é capaz de **aprender** da experiência E para alguma classe de tarefas T e com o desempenho P , se o desempenho das tarefas em T , medidas por P melhoram com a experiência E ”. Ou seja, temos uma máquina que a cada vez que entra em contato com uma dada experiência ela melhora o seu desempenho de resolução de uma tarefa.

Modelos de aprendizado de máquina são amplamente utilizados para a resolução das mais diversas tarefas. Em Alencar et al. (2020), tais modelos foram aplicados à predição de demanda de veículos em serviços de compartilhamento para determinar a demanda necessária para o futuro. Na pesquisa aplicada por Jiménez, Benalcázar e Sotomayor (2017), modelos de aprendizado de máquina foram utilizados para traduzir língua de sinais em texto. Em uma aplicação voltada ao público geral, em diversos dispositivos há a implementação de assistentes de voz que utilizam modelos de aprendizado de máquina para entender comandos dados pelo usuário e responder ou executar ações (HOY, 2018).

A utilização de modelos de aprendizado de máquina envolve diversas etapas sistemáticas e repetitivas. Primeiramente, ocorre a coleta de dados que, em seguida, são tratados para realizar a extração de características relevantes. Após essa etapa, é efetuada

a escolha de modelos, onde é feita uma pesquisa de quais modelos podem ser utilizados para a resolução do problema. Com isso, os modelos são treinados e avaliados e, dado o modelo de melhor desempenho, o pesquisador irá pesquisar quais são os hiperparâmetros apropriados para melhorá-lo.

2.2 Classificação

A classificação é um problema clássico da área de aprendizado de máquina, onde um conjunto de características pode ser utilizado para gerar um modelo que possa prever novos valores discretos e não ordenados (KOTSIANTIS et al., 2007). Um exemplo é a classificação de imagens de animais para determinar se é de um gato ou não (JAJODIA; GARG, 2019). Esse é um exemplo de classificação binária, onde o modelo irá prever somente os valores verdadeiro ou falso. Existe também o problema de predição de múltiplas classes, onde dado os valores de entrada o modelo irá prever uma classe dentre um conjunto (ALY, 2005). Um exemplo seria determinar a espécie de um animal dadas suas características.

Como apresentado por Truong et al. (2019), a classificação pode ser um dos meios de diminuir o tempo de processamento da busca por modelos de predição em projetos de AutoML. Para isso, uma classificação de múltiplas classes pode ser utilizada para restringir o conjunto de parâmetros e modelos a serem considerados como candidatos, como sendo ideais para os dados de entrada. Uma vez que, dada a classe encontrada pela classificação, o conjunto de modelos de predição e hiperparâmetros pode ser limitado para os que previamente já apresentaram bom desempenho, que conseqüentemente, diminui o tempo de busca. Assim, se torna uma opção para o contexto de otimização de busca de modelos desse trabalho.

Para séries temporais, uma abordagem para realização da classificação é utilizar métricas que medem o quão similar são as séries por meio de cálculos de distância entre a série a ser classificada e as já rotuladas. Um exemplo seria uma distância Euclideana, onde a distância é calculada através de um mapeamento linear sobre os pontos (DAU et al., 2019). Outro exemplo seria o *Dynamic Time Wrapping* (DTW), que é uma técnica para encontrar o alinhamento ótimo entre duas séries, sobre a restrição de uma janela de tamanho w . Com isso, é realizada uma busca sobre as séries procurando onde há

uma janela de menor custo possível, assim, encontrando um alinhamento de forma não linear (MÜLLER, 2007). Contudo, essa metodologia não permite a utilização dos modelos do estado-da-arte sem as devidas modificações.

A abordagem comumente utilizada para classificação de séries temporais é a utilização de características dos dados. Para aplicar tal metodologia, características voltadas às séries temporais devem ser extraídas devidamente. Ferramentas como `tsfresh` utilizam de diversas métricas para a coleta em massa de características das séries (CHRIST et al., 2018). Exemplos seriam os valores dos quantis da série, número de picos de valores ou coeficiente de autocorrelação. O trabalho de Christ, Kempa-Liehr e Feindt (2016) apresenta uma aplicação dessa metodologia, onde diversas características são calculadas para as séries, contudo, há um tratamento para filtrar as mais significativas, para assim, poderem ser utilizadas pelos mais diversos modelos de classificação do estado-da-arte.

O presente trabalho utiliza a classificação de séries para otimizar a escolha de modelos de predição. Com a classificação pretende-se tornar a busca de modelos mais rápida e eficiente, já que o modelo de predição mais apropriado aos dados será escolhido para a série temporal desejada. Para isso, a abordagem escolhida será a de coleta de características da série, que como exposto nesta seção, torna possível a utilização dos mais diversos modelos de classificação.

2.3 AutoML

Projetos de *Automated Machine Learning* (AutoML) tem como objetivo simplificar todos os passos necessários para a criação de modelos de aprendizado de máquina, uma vez que ele executa todos os processos de forma automatizada. Com isso, o usuário não necessita da experiência para iniciar projetos na área de aprendizado de máquina, que vai desde como tratar os dados a conhecimentos de estatística (HE; ZHAO; CHU, 2021; TRUONG et al., 2019). A Figura 2.1 apresenta um exemplo de todas as etapas que um projeto de genérico de AutoML pode ser responsável. Estas etapas podem ser divididas em preparação dos dados, preparação das características, e geração dos modelos, que inclui a otimização de hiperparâmetros.

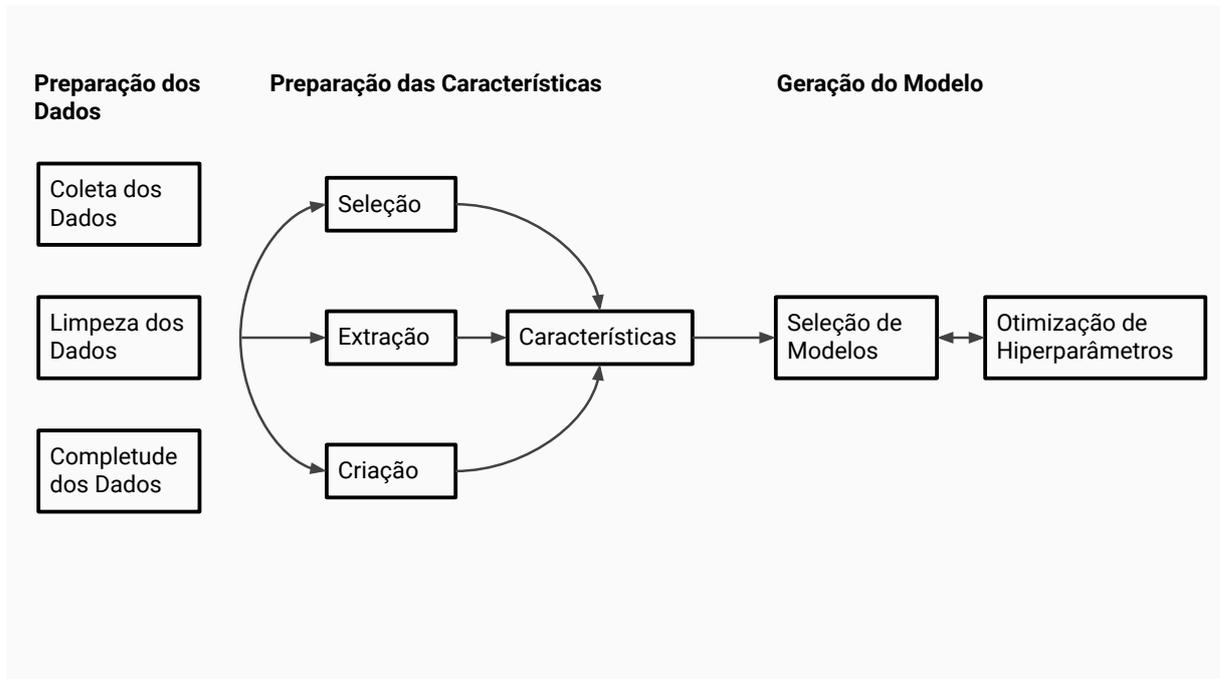


Figura 2.1: Visão geral da arquitetura de um projeto de AutoML. Figura adaptada de (HE; ZHAO; CHU, 2021).

2.3.1 Preparação dos dados e Extração de Características

Primeiramente, durante a etapa de preparação dos dados, eles são tratados a fim de serem utilizados pelo modelo de predição. Os valores considerados como ruídos podem ser normalizados e valores ausentes são preenchidos (HE; ZHAO; CHU, 2021). Um exemplo seria identificar os tipos dos dados de entrada, ou seja, se eles são inteiros, reais ou valores binários. O modelo também pode identificar tipos mais complexos, como a identificação de endereços, datas ou números telefônicos (TRUONG et al., 2019). Assim, se algum dado não estiver no padrão esperado ou é faltante ele pode ser removido ou tratado.

A etapa seguinte lida com a preparação de características dos dados já tratados. Esta etapa foca na extração de características de uma base de dados. Nesta etapa, uma das atividades que pode ser executada é a seleção de características, que remove características irrelevantes. Características também podem ser extraídas utilizando funções de diminuição de dimensionalidade, como a partir de uma análise dos componentes principais (PCA) (PEARSON, 1901), que irá extrair informação das características não correlacionadas.

A atividade de criação de características é a etapa que mais depende da experiência prática do usuário. Por este motivo muitos modelos de AutoML não a imple-

mentam ou necessitam de auxílio do usuário para efetuar as suas operações (HE; ZHAO; CHU, 2021; TRUONG et al., 2019). Contudo, há modelos e ferramentas que implementam processos automatizados de combinações de características a fim de gerar novas (GAMA, 2004; VAFAIE; JONG, 1998; SONDHI, 2009; ZHENG, 1998). Exemplos simples seriam a soma ou média entre os valores, mas características mais elaboradas também podem ser extraídas, como por meio da utilização de funções de auto correlação em modelos de AutoML voltados à séries temporais, para extração de componentes da série ou para a adição de novas características ao modelo (CHRIST et al., 2018).

2.3.2 Geração do Modelo de Predição

Esta etapa tem como objetivo utilizar os dados gerados pelas fases anteriores para gerar diferentes tipos de modelos de predição com diferentes hiperparâmetros para, assim, selecionar o melhor modelo. Esta fase pode ser dividida em três etapas: (i) a seleção de modelos de; (ii) otimização de hiperparâmetros, e (iii) busca por arquitetura para redes neurais.

Durante a seleção de modelos de predição, o algoritmo mais apropriado é selecionado para o problema. Tais algoritmos podem ser os mais diversos da área de aprendizado de máquina, como Regressão Logística, algoritmos baseados em árvores ou até modelos de redes neurais. A segunda etapa é a otimização de hiperparâmetros, onde métodos como *grid search* ou *random search* (LIASHCHYNSKYI; LIASHCHYNSKYI, 2019) são utilizados para realizar uma busca dos hiperparâmetros que geram o modelo de predição com melhor desempenho. Estes dois processos utilizam de um **espaço de busca**, que são as combinações de possíveis valores que os hiperparâmetros podem assumir (HE; ZHAO; CHU, 2021; YAO et al., 2019). A Figura 2.2 apresenta um exemplo de variáveis de um projeto, que são os hiperparâmetros a serem ajustados com base em um espaço de busca.

A busca pelos melhores parâmetros é o que leva maior tempo de processamento em um modelo de AutoML. Para diminuir o tempo necessário para a realização dessa etapa, há diversas soluções possíveis que podem diminuir o espaço de busca. Uma primeira abordagem seria encontrar um conjunto inicial de parâmetros. Para isso podem ser utilizados *datasets* pré-processados que dada a similaridade com o *dataset* de entrada, um

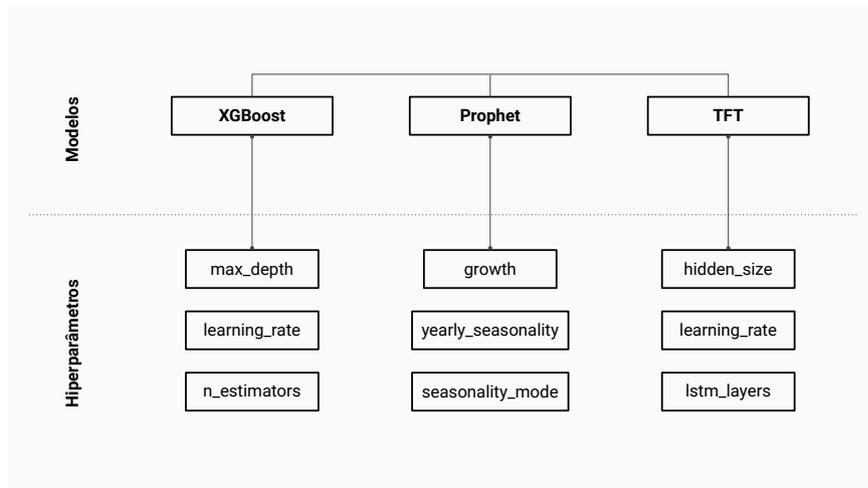


Figura 2.2: Exemplo de espaço de busca. Figura adaptada de (YAO et al., 2019).

modelo de classificação pode escolher parâmetros previamente selecionados. A segunda abordagem é relacionar a seleção de modelos de predição com a de hiperparâmetros, como exemplo, aplicar um *grid search* tanto na busca de parâmetros quanto na busca de modelos de predição. A terceira opção é fixar o tempo de processamento para cada combinação de hiperparâmetros, impedindo a seleção de valores que podem causar uma otimização lenta. A quarta opção é evitar combinações de parâmetros que já são conhecidos por aumentar o tempo de processamento (TRUONG et al., 2019).

Para Truong et al. (2019) e He, Zhao e Chu (2021), há a etapa de busca de arquitetura para redes neurais, assim sendo específica para esse domínio. Assim, para He, Zhao e Chu (2021), são apresentados diversos algoritmos que podem ser utilizados nesta etapa, como aprendizado por reforço ou algoritmos evolutivos, para realizar a busca dos melhores parâmetros que definem a arquitetura da rede.

2.3.3 AutoML para séries temporais

Como séries temporais detém de características próprias, modelos de aprendizado de máquina e projetos de AutoML são desenvolvidos especificamente para o seu tratamento. Primeiramente, existem modelos de predição de séries temporais que trazem características de *frameworks* de AutoML. Um exemplo seria o Prophet (TAYLOR; LETHAM, 2018), que automatiza a maior parte dos processos de tratamento de dados e busca de parâmetros. Ele só necessita que a série temporal seja passada como parâmetro de trei-

namento para automaticamente determinar características como tendência, sazonalidade e o efeito de feriados sobre a série. Ao fim, é gerado um modelo estatístico que melhor se adapta aos dados passados.

Uma implementação de AutoML voltada a predição de séries temporais é o modelo Time Series Pipeline Optimisation (TSPO) (DAHL, 2020). Durante a etapa de preparação, o modelo decompõe a série em três componentes, tendência, sazonalidade e um componente restante. Depois, os dados são normalizados e tratados para garantir o formato requisitado. Além disso, características como médias móveis e *lags* são inseridas para serem utilizadas pelos modelos. A seleção de características, modelos e hiperparâmetros é feita através de um algoritmo genético (KOZA; POLI, 2005), que encontra a melhor combinação a partir de um processo de evolução natural. O TSPO utiliza tanto modelos baseados em árvores de decisão quanto de redes neurais para realizar as predições.

Um exemplo para a tarefa de classificação de séries seria o Mcfly (KUPPEVELT et al., 2020), que é um *framework* de AutoML que se baseia em modelos de aprendizado profundo. Neste *framework*, a preparação dos dados fica por parte do usuário. O usuário também deve indicar o número de modelos de classificação possíveis a serem considerados. Assim o limite de tempo de processamento fica a cargo do usuário. Cada modelo de classificação disponível recebe como entrada os dados sem nenhuma extração de características. A busca de hiperparâmetros é efetuada a partir de uma busca aleatória sobre um espaço de busca previamente estabelecido durante a implementação. Após o treinamento, ele oferece uma visualização interativa sobre todos os resultados, como qual modelo de classificação foi escolhido, seus atributos e o desempenho de cada modelo testado.

Com isso, um modelo de AutoML, foi utilizado para o treinamento dos modelos de predição, e um modelo de classificação foi utilizado para realizar uma busca otimizada dos modelos de predição. Esse trabalho foca na etapa de geração de modelos, uma vez que a classificação irá auxiliar na decisão do modelo ideal para a série de entrada. Assim, o modelo de AutoML é responsável por treinar e fazer a busca por hiperparâmetros do modelo de predição escolhido pela classificação.

3 Metodologia

Neste capítulo são apresentados os métodos utilizados para o trabalho bem como o seu processo metodológico. Durante a seção 3.1 são apresentados os métodos utilizados, sendo que a sub-seção 3.1.1 apresenta a metodologia utilizada pelo TS-AutoML, e durante a sub-seção 3.1.2 é apresentada a metodologia sobre a geração dos modelos de classificação para a escolha do modelo de predição ideal. Por fim, a seção 3.2 resume todo o processo metodológico por trás da utilização dos dados e métodos.

3.1 Métodos Utilizados

Nessa seção são apresentados os métodos utilizados no trabalho. A sub-seção 3.1.1 apresenta o TS-AutoML, modelo de AutoML voltado à séries temporais, no qual é proposta a inclusão da classificação para otimizar a escolha de modelos de predição de forma automatizada. A sub-seção 3.1.2 apresenta o modelo de AutoML usado para realizar a escolha do modelo de classificação a ser utilizado.

3.1.1 TS-AutoML

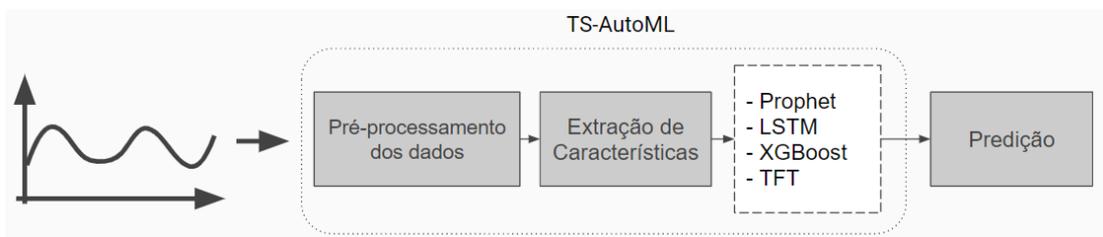


Figura 3.1: TS-AutoML em alto nível

A Figura 3.1 resume em alto nível as etapas que o TS-AutoML realiza para executar a predição dos dados de uma série temporal. Inicialmente, o TS-AutoML realiza um tratamento nos dados de entrada. Para o modelo, o primeiro passo é verificar se os dados de entrada estão no formato correto, que seriam duas colunas de valores, sendo a primeira uma coluna de instantes de tempo e a segunda de valores numéricos. A

coluna de instantes de tempo é tratada para garantir que está no formato apropriado e são removidas qualquer informação referente ao fuso horário. Valores vazios na segunda coluna são preenchidos utilizando a estratégia de preenchimento com o último valor válido para todos os dados e com o próximo valor válido, caso os valores faltantes não tenham valores anteriores. Por fim, os dados são separados em dois conjuntos, um de treino e um de teste.

Durante a etapa de extração de características, a série passa por uma análise de auto correlação parcial. Este processo avalia quais são os *lags* com maior correlação na série. Dado um ponto x_i de uma série temporal, um *lag* pode ser definido como um ponto passado do ponto inicial de referencia, ou seja, qualquer x_{i-n} com uma alta correlação será considerado. Os *lags* de maior correlação são utilizados como características para os modelos de predição. Precisamente, para modelos que recebem como entrada dados tabulares, os dados são adaptados utilizando os *lags* como novas colunas de dados, ou seja, eles se tornam características para os modelos.

O TS-AutoML utiliza diversos modelos para realizar as predições. Os modelos utilizados são os seguintes: Prophet, *Long Short-Term Memory* (LSTM) (LAPTEV et al., 2017), *Temporal Fusion Transformer* (TFT) (LIM et al., 2021) e modelos baseados em árvores de decisão como o XGBoost, LightGBM e CatBoost (CHEN; GUESTRIN, 2016; KE et al., 2017; PROKHORENKOVA et al., 2018). O conjunto de treino é utilizado para o treinamento dos modelos de predição e o conjunto de teste para a sua avaliação. Com isso, todos os modelos de predição são treinados utilizando o conjunto de treino, sendo que cada modelo detém de um conjunto de valores de hiperparâmetros pré-determinados, resultando em mais de uma possibilidade para cada modelo sendo avaliado.

Após o treino, são realizadas predições sobre uma janela de avaliação que pode ser indicada pelo usuário, ela pode ter tamanho 1 até o número de *lags* máximo definido pelo usuário ou até o último *lag* com alta correlação. As predições são comparadas com os valores reais do conjunto de teste. Diversas métricas de medição de erro são utilizadas para avaliar a qualidade das predições, dentre elas consta o erro absoluto médio (MAE) e a raiz do erro quadrático médio (RMSE). A escolha do melhor modelo é efetuado pelo

menor valor obtido pela métrica de erro de porcentagem absoluta ponderada (WAPE)³ média.

3.1.2 Classificação das séries temporais

Para o AutoML proposto, a classificação tem como objetivo melhorar o tempo de busca do modelo de predição ideal para os dados de entrada. A metodologia escolhida para esta tarefa é a de extração de características da série temporal. Este processo é realizado pela ferramenta `tsfresh` (CHRIST et al., 2018). Essa ferramenta, realiza a extração de uma extensa coleção de características, que vão desde a quantidade de picos da série à auto-correlação entre os valores a lista completa de características disponíveis pode ser encontrada na página oficial do `tsfresh`⁴.

Para realizar a classificação, é utilizado o AutoGluon, modelo de AutoML voltado a tarefa de classificação (ERICKSON et al., 2020). Ele realiza o treinamento dos seguintes modelos de classificação: Redes Neurais, LightGBM, CatBoost, Random Forests, Extremely Randomized Trees e k -Nearest Neighbors. Estes são treinados e avaliados utilizando o método de validação cruzada chamado k -folding bagging (PARMANTO; MUNRO; DOYLE, 1996). Neste método, o conjunto de dados é dividido em k pedaços, onde $k-1$ pedaços são utilizados para treinar o modelo e o pedaço que não foi utilizado no seu treino é usado para encontrar a previsão desses dados pelo modelo, para assim, avaliar o seu desempenho.

O AutoGluon é utilizado para determinar o melhor modelo de classificação para os dados de entrada e tal modelo escolhido irá classificar qual é o melhor modelo de predição para uma dada série temporal. Para treino do mesmo, são utilizados os modelos de predição escolhidos durante o processamento por força bruta do TS-AutoML, tal processo é explicado em detalhes durante a seção 3.2. O critério de escolha do modelo de classificação se baseia na acurácia dos modelos internos do AutoGluon.

³<https://docs.aws.amazon.com/forecast/latest/dg/metrics.html>, Acesso em: 24/02/2022

⁴https://tsfresh.readthedocs.io/en/latest/text/list_of_features.html, Acesso em: 06/02/2022

3.2 Processo Metodológico

A Figura 3.2 resume o processo metodológico a ser aplicado neste trabalho. Inicialmente, as bases de dados são utilizadas para realizar previsões usando somente o TS-AutoML, ou seja, testando todas as opções de modelos de previsão possíveis disponíveis pelo TS-AutoML, como citado na seção 3.1.1. Com isso, são obtidas diversas métricas de avaliação de desempenho, como o RMSE e o tempo de processamento. Elas são responsáveis por medir o desempenho da previsão em relação aos demais testes executados. Além disso, obtemos o modelo de previsão de melhor desempenho para cada série.

Em sequência, há o processo de utilização do TS-AutoML em conjunto ao AutoGluon. Primeiramente, o `tsfresh` foi utilizado para extrair as características de cada série temporal das bases de dados utilizadas. Com isso, montando uma nova base de características em comum de cada uma das bases, ou seja, haverá um tratamento dos dados para utilizar somente a interseção das características encontradas em todas as bases. Após esse tratamento, o AutoGluon será utilizado para determinar o melhor modelo de classificação. Tal modelo escolhido tem como objetivo classificar o modelo de previsão temporal que melhor se adapta à série de entrada.

Por fim, foi selecionada uma base de dados que não foi utilizada pelo processo anterior para comparar o desempenho de todo o processo. Assim, a base foi utilizada pelo TS-AutoML para treino e análise de desempenho tanto para o processo por meio da escolha de modelos previsão por força bruta quanto para o processo utilizando o modelo de classificação. Por fim, os resultados foram utilizados para avaliar se houve perda do desempenho de previsão caso a classificação for utilizada e o ganho de tempo de processamento obtido.

Para a comparação de desempenho foram utilizadas as métricas RMSE (*Raiz do erro quadrático médio*) e MAE (*Erro absoluto médio*). Para ambas equações considere Y_i o valor real, \hat{Y} o valor predito e n a quantidade de valores. Com isso, a equação 3.1 apresenta o RMSE que, em outras palavras, representa a distância euclidiana entre os dois valores normalizada por $1/\sqrt{n}$. A equação 3.2 representa o MAE, que calcula a diferença absoluta média entre a previsão e o valor real.

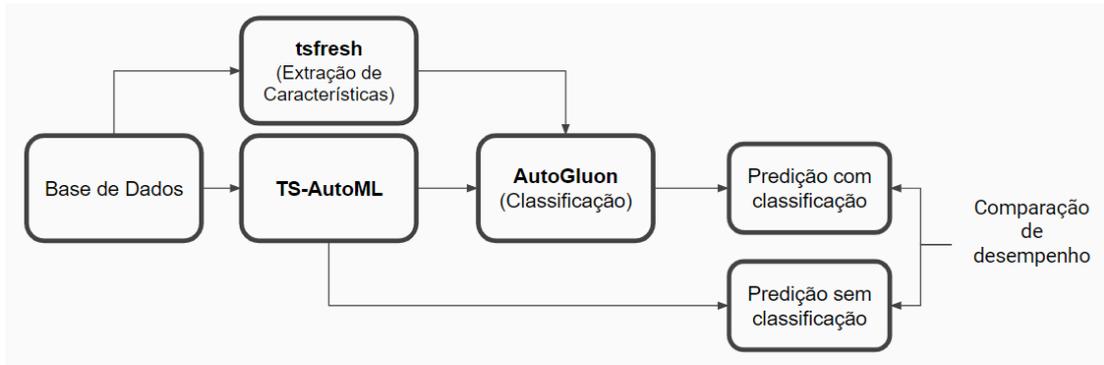


Figura 3.2: Processo de comparação de desempenho

$$RMSE(\mathbf{Y}, \hat{\mathbf{Y}}) = \sqrt{\frac{1}{n} \sum_{i=1}^n (Y_i - \hat{Y}_i)^2} \quad (3.1)$$

$$MAE(\mathbf{Y}, \hat{\mathbf{Y}}) = \frac{1}{n} \sum_{i=1}^n |Y_i - \hat{Y}_i| \quad (3.2)$$

4 Experimento

Neste capítulo são apresentados todos os resultados obtidos ao longo do processamento das bases de dados conjunto aos modelos de predição e classificação. O capítulo é iniciado pela seção 4.1, que apresenta as bases de dados utilizadas no trabalho. Em seguida, a seção 4.2 expõe os resultados obtidos durante o processo de classificação. Por fim, a seção 4.3 apresenta os resultados obtidos ao fim da execução do processo metodológico.

4.1 Base de dados

Os dados utilizados são baseados no trabalho de Godahewa et al. (2021). O artigo apresenta uma coleção de bases de dados, onde cada base contém uma coleção de séries. Cada uma detém as seguintes características: domínio, número de séries, se há dados faltantes ou não, se foi utilizada em competições e o tamanho mínimo e máximo das séries.

Tabela 4.1: Base de dados de séries temporais utilizadas.

Nome	Domínio	Número de séries	Granularidade	Séries não processadas
Covid Deaths	Saúde	266	Dia	15%
FRED-MD	Economia	107	Semana	0%
San Francisco Traffic	Transporte	862	Semana	0,05%
Vehicle Trip	Transporte	329	Dia	12%
NN5	Bancária	111	Semana	0%
M4	Múltiplos	414	Hora	6%

A Tabela 4.1 apresenta as bases de dados utilizadas para treinamento e teste do trabalho. Cada uma delas detém de uma coleção de séries temporais voltadas a um contexto. Como exemplo, a base Covid Deaths apresenta séries de mortes diárias por COVID-19 em um conjunto de países e estados ao longo de 22/01/2020 a 20/08/2020. Detalhes específicos sobre cada uma das bases podem ser encontrados em (GODAHEWA et al., 2021) e elas estão disponíveis em um repositório público⁵. Além disso, como o TS-AutoML tem caráter experimental e depende de implementações externas para os modelos, foi admitida uma taxa de erros durante o processamento. Assim, há uma porcentagem de

⁵<https://zenodo.org/communities/forecasting/>, Acesso em: 18/12/2021

séries não processadas, por conta de particularidades dos modelos de predição utilizados, como pode ser visto na tabela 4.1.

Para o experimento, as bases FRED-MD, San Francisco Traffic, Vehicle Trip, NN5 e M4 foram utilizadas para treinamento do modelo de classificação, onde elas foram processadas pelo TS-AutoML inicialmente, determinando o melhor modelo através da avaliação de todas as possibilidades de modelos e parâmetros disponíveis. Após isso, cada série foi processada pelo tsfresh a fim de obter as suas características para, por fim, utilizar o AutoGluon para determinar o melhor modelo de classificação. Esse processo resultou em uma tabela de 599 características para as 1759 séries temporais obtidas com as bases de dados. Para o treinamento do modelo de classificação, as características foram embaralhadas e separadas em um conjunto de treino e teste, representando 80% e 20% dos dados respectivamente. Por fim, a base Covid Deaths foi utilizada para a comparação final entre a predição com ou sem classificação.

4.2 Classificação

Como resultado da predição dos dados de treino pelo TS-AutoML, foi obtida a Tabela 4.2 que apresenta os modelos de predição selecionados dentre as bases de dados utilizadas. Esses modelos foram escolhidos por meio da comparação de todos os hiperparâmetros e modelos disponíveis no TS-AutoML. Como pode ser visto, o TFT foi o modelo mais escolhido, com aproximadamente 60,65% das ocorrências. Em sequência, o Prophet detém aproximadamente 15,86% das ocorrências.

Tabela 4.2: Modelos de predição escolhidos pelo TS-AutoML durante o processamento inicial das bases de treino.

Modelo	Ocorrências
TFT	1067
Prophet	279
CatBoost	145
LightGBM	122
XGBoost	103
LSTM	43

Após o processamento dos dados pelo AutoGluon, foram obtidos os resultados apresentados na Tabela 4.3. Ela apresenta todos os modelos de classificação utilizados

pelo AutoGluon conjunto a acurácia obtida. Os modelos estão ordenados conforme o desempenho, ou seja, o modelo WeightedEnsemble L2 obteve o melhor valor de acurácia para os dados, seguido pelo LightGBM e o CatBoost. Com isso, o modelo a ser utilizado para a classificação é o WeightedEnsemble L2.

Tabela 4.3: Acurácia obtida por todos os modelos do AutoGluon durante o treino.

Modelo	Acurácia
WeightedEnsemble L2	0.645390
LightGBM	0.641844
CatBoost	0.641844
LightGBMLarge	0.641844
LightGBMXT	0.638298
ExtraTreesEntr	0.638298
RandomForestEntr	0.634752
ExtraTreesGini	0.634752
RandomForestGini	0.634752
XGBoost	0.627660
NeuralNetMXNet	0.624113
NeuralNetFastAI	0.585106
KNeighborsDist	0.560284
KNeighborsUnif	0.535461

A Tabela 4.4 apresenta os modelos de predição selecionados pelo modelo de classificação (WeightedEnsemble L2) para a base de dados de teste e para execução da seleção de modelos de predição por força bruta do TS-AutoML. Em comparação com a Tabela 4.2, é notável a diminuição da diversidade de modelos escolhidos. Como esperado, algumas possibilidades escolhidas para casos específicos não estão sendo consideradas. Além disso, tais resultados podem ser um reflexo dos modelos selecionados para a base de treino, onde a maioria está concentrada no TFT e Prophet, uma vez que a seleção sem classificação detém uma maior diversidade de modelos.

Tabela 4.4: Modelos escolhidos com e sem classificação das séries da base de teste.

Modelo	Sem Classificação	Com Classificação
Prophet	98	126
TFT	10	97
XGBoost	22	1
CatBoost	60	0
LightGBM	34	0
LSTM	0	0

4.3 Resultados

A Tabela 4.5 apresenta os valores obtidos após o processamento do TS-AutoML com e sem o processo de classificação. Nela pode ser visto o tempo de processamento, RMSE e MAE de ambos processos conjunto aos seus valores mínimos, médios e máximos, além do tempo total. Ela também contém a variação percentual para auxiliar nas análises, onde valores negativos indicam uma redução dos valores do processo sem classificação para o com classificação e positivos um aumento. Os valores marcados com um traço (-) indicam que a variação não pôde ser calculada, no caso, por conta do valor mínimo encontrado no processo sem classificação ser igual a zero. Os valores em negrito indicam quem teve melhor desempenho para cada resultado.

Tabela 4.5: Resultados obtidos através da predição com ou sem classificação.

		Sem Classificação	Com Classificação	Varição Percentual (%)
Tempo (seg)	Total	141458,45	22821,15	-83,87
	Min	220,14	2,02	-99,08
	Média	631,51	101,88	-83,87
	Max	1435,61	388,99	-72,92
RMSE	Min	0,0000	0,0085	-
	Média	390,7948	554,8986	41,99
	Max	12765,9946	12765,9946	0,0
MAE	Min	0,0000	0,0085	-
	Média	370,4253	530,0722	43,09
	Max	12654,8005	12654,8005	0,0

Analisando o tempo de processamento da tabela 4.5, é perceptível um ganho considerável em relação ao processo com classificação em comparação ao sem o procedimento. Para o tempo total, há um ganho aproximado de 84%, sendo necessário cerca de 6 horas para o processamento com classificação e 39 horas para sem classificação. Ademais, ocorreu uma melhora do tempo médio de 10 minutos para 2 minutos. Assim, de fato, a classificação auxilia o uso da ferramenta no ponto de vista do tempo uma vez que, removendo a necessidade de testar todos os modelos de predição disponíveis, a ferramenta obtém um ganho significativo em tempo de processamento.

Para a análise de erros de predição, os resultados apresentam uma diminuição no desempenho. Isso porque, para esses valores, o objetivo da classificação é se aproximar ao máximo da abordagem utilizando todas as possibilidades, mesmo com uma seleção

reduzida de modelos. Entretanto, o resultado médio obtido foi de aproximadamente 40%, tanto para o RMSE como para o MAE. Esse comportamento representa um aumento de erro significativo, dada a redução do espaço de busca de modelos de predição.

A diminuição de desempenho em relação aos erros é um reflexo dos modelos escolhidos durante a classificação. Como observado na tabela 4.4, há uma redução significativa da diversidade dos modelos escolhidos pela classificação. Para a versão sem classificação, o TS-AutoML escolheu de forma bem distribuída todos os modelos disponíveis, exceto pelo LSTM que não foi escolhido, e o TFT foi a minoria, com somente 4,46% das ocorrências. Para a classificação, o TFT foi escolhido 43,75% das vezes, resultando em modelos não apropriados para as séries e conseqüentemente, reduzindo o seu desempenho. Este problema de balanceamento pode ser tratado futuramente como um dos componentes anteriores a etapa de classificação, onde os dados processados somente pelo TS-AutoML devem ser tratados para garantir uma distribuição uniforme entre os modelos de predição escolhidos.

5 Conclusão

Neste trabalho, foi proposto a utilização de uma ferramenta de *Automated Machine Learning* (AutoML) voltada a séries temporais, conjunto a um modelo de classificação para otimizar a busca otimizada de modelos de predição. A ferramenta utilizada foi o TS-AutoML, que utiliza modelos voltados à predição de séries temporais para selecionar o modelo mais apropriado aos dados de entrada. Para a tarefa de classificação, foi utilizada a biblioteca *tsfresh* para a coleta das características das séries juntamente ao *AutoGluon*, modelo de AutoML generalista, voltado à escolha de modelos de classificação para, assim, realizar uma comparação do desempenho utilizando ou não a classificação para a escolha do modelo preditivo.

A análise dos resultados apresentou um ganho significativo de tempo de processamento. Entretanto ocorreu uma perda considerável no desempenho de predição. Com o uso da classificação, o número de possibilidades de modelos é reduzida, assim diminuindo o tempo de processamento a somente um modelo e suas variações possíveis de hiperparâmetros. Contudo, o TS-AutoML se torna dependente da acurácia do modelo de classificação, uma vez que o modelo escolhido pode não ser o apropriado para a série, o que aumenta as taxas de erro observadas. Dessa maneira, é necessário que o modelo de classificação seja o mais preciso possível para garantir o desempenho adequado do TS-AutoML.

Para trabalhos futuros, mais bases de dados podem ser utilizadas para garantir uma grande diversidade de séries temporais. Isso pode aumentar a acurácia da classificação, uma vez que pode remover vieses e tornar a escolha de modelos mais diversas. Também podem ser utilizadas estratégias de balanceamento dos dados para garantir a uniformidade de escolhas. Além disso, podem ser implementados recursos no TS-AutoML para escolher a abordagem (com ou sem classificação) mais apropriada, dada a incerteza do modelo escolhido de forma automatizada. Por fim, o TS-AutoML pode utilizar de melhorias comunitárias para re-treino do modelo de classificação para cada série que não for classificada com uma precisão adequada.

Bibliografia

- ALENCAR, V. A. et al. Characterizing client usage patterns and service demand for car-sharing systems. *Information Systems*, Elsevier, p. 101448, 2019.
- ALENCAR, V. A. S. de B. et al. Predição de séries temporais de demanda em modelos de compartilhamento de veículos para modelos uni e multi variáveis. In: SBC. *Anais do IV Workshop de Computação Urbana*. [S.l.], 2020. p. 84–96.
- ALY, M. Survey on multiclass classification methods. *Neural Netw*, Citeseer, v. 19, p. 1–9, 2005.
- CHEN, T.; GUESTRIN, C. XGBoost: A Scalable Tree Boosting System. In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. [S.l.: s.n.], 2016. p. 785–794.
- CHRIST, M. et al. Time series feature extraction on basis of scalable hypothesis tests (tsfresh—a python package). *Neurocomputing*, Elsevier, v. 307, p. 72–77, 2018.
- CHRIST, M.; KEMPA-LIEHR, A. W.; FEINDT, M. Distributed and parallel time series feature extraction for industrial big data applications. *arXiv preprint arXiv:1610.07717*, 2016.
- COCCA, M. et al. On Car-Sharing Usage Prediction with Open Socio-Demographic Data. *Electronics*, Multidisciplinary Digital Publishing Institute, v. 9, n. 1, p. 72, 2020.
- DAHL, S. M. J. *TSPO: an autoML approach to time series forecasting*. Tese (Doutorado) — NOVA Information Management School, 2020.
- DAU, H. A. et al. The ucr time series archive. *IEEE/CAA Journal of Automatica Sinica*, IEEE, v. 6, n. 6, p. 1293–1305, 2019.
- ERICKSON, N. et al. Autogluon-tabular: Robust and accurate automl for structured data. *arXiv preprint arXiv:2003.06505*, 2020.
- FRIEDMAN, J. H. Greedy Function Approximation: a Gradient Boosting Machine. *Annals of Statistics*, p. 1189–1232, 2001.
- GAMA, J. Functional trees. *Machine learning*, Springer, v. 55, n. 3, p. 219–250, 2004.
- GODAHEWA, R. et al. *Monash Time Series Forecasting Archive*. 2021.
- HARVEY, A. C.; PETERS, S. Estimation procedures for structural time series models. *Journal of forecasting*, Wiley Online Library, v. 9, n. 2, p. 89–108, 1990.
- HE, X.; ZHAO, K.; CHU, X. Automl: A survey of the state-of-the-art. *Knowledge-Based Systems*, Elsevier, v. 212, p. 106622, 2021.
- HOY, M. B. Alexa, siri, cortana, and more: an introduction to voice assistants. *Medical reference services quarterly*, Taylor & Francis, v. 37, n. 1, p. 81–88, 2018.

- JAJODIA, T.; GARG, P. Image classification—cat and dog images. *Image*, v. 6, n. 23, p. 570–572, 2019.
- JAMES, S. L. et al. Developing a comprehensive time series of gdp per capita for 210 countries from 1950 to 2015. *Population health metrics*, BioMed Central, v. 10, n. 1, p. 1–12, 2012.
- JIMÉNEZ, L. A. E.; BENALCÁZAR, M. E.; SOTOMAYOR, N. Gesture recognition and machine learning applied to sign language translation. In: TORRES, I.; BUSTAMANTE, J.; SIERRA, D. A. (Ed.). *VII Latin American Congress on Biomedical Engineering CLAIB 2016, Bucaramanga, Santander, Colombia, October 26th -28th, 2016*. Singapore: Springer Singapore, 2017. p. 233–236. ISBN 978-981-10-4086-3.
- KE, G. et al. Lightgbm: A Highly Efficient Gradient Boosting Decision Tree. In: *Advances in Neural Information Processing Systems*. [S.l.: s.n.], 2017. p. 3146–3154.
- KOTSIANTIS, S. B. et al. Supervised machine learning: A review of classification techniques. *Emerging artificial intelligence applications in computer engineering*, Amsterdam, v. 160, n. 1, p. 3–24, 2007.
- KOZA, J. R.; POLI, R. Genetic programming. In: *Search methodologies*. [S.l.]: Springer, 2005. p. 127–164.
- KUPPEVELT, D. van et al. Mcfly: Automated deep learning on time series. *SoftwareX*, v. 12, p. 100548, 2020. ISSN 2352-7110. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S2352711019300202>>.
- LAPTEV, N. et al. Time-series extreme event forecasting with neural networks at uber. In: *International conference on machine learning*. [S.l.: s.n.], 2017. v. 34, p. 1–5.
- LIASHCHYNSKYI, P.; LIASHCHYNSKYI, P. Grid search, random search, genetic algorithm: A big comparison for nas. *arXiv preprint arXiv:1912.06059*, 2019.
- LIM, B. et al. Temporal fusion transformers for interpretable multi-horizon time series forecasting. *International Journal of Forecasting*, Elsevier, 2021.
- MITCHELL, T. M. *Machine Learning*. New York: McGraw-Hill, 1997. ISBN 978-0-07-042807-2.
- MÜLLER, M. Dynamic time warping. *Information retrieval for music and motion*, Springer, p. 69–84, 2007.
- PARMANTO, B.; MUNRO, P. W.; DOYLE, H. R. Reducing variance of committee prediction with resampling techniques. *Connection Science*, Taylor & Francis, v. 8, n. 3-4, p. 405–426, 1996.
- PEARSON, K. Liii. on lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin philosophical magazine and journal of science*, Taylor & Francis, v. 2, n. 11, p. 559–572, 1901.
- PROKHORENKOVA, L. et al. CatBoost: Unbiased Boosting with Categorical Features. In: *Advances in Neural Information processing systems*. [S.l.: s.n.], 2018. p. 6638–6648.
- SONDHI, P. Feature construction methods: a survey. *sifaka. cs. uiuc. edu*, v. 69, p. 70–71, 2009.

TAYLOR, S. J.; LETHAM, B. Forecasting at scale. *The American Statistician*, Taylor & Francis, v. 72, n. 1, p. 37–45, 2018.

TRUONG, A. et al. Towards automated machine learning: Evaluation and comparison of automl approaches and tools. In: *2019 IEEE 31st International Conference on Tools with Artificial Intelligence (ICTAI)*. [S.l.: s.n.], 2019. p. 1471–1479.

VAFIAIE, H.; JONG, K. D. Evolutionary feature space transformation. In: *Feature Extraction, Construction and Selection*. [S.l.]: Springer, 1998. p. 307–323.

VAROTSOS, P.; SARLIS, N. V.; SKORDAS, E. S. *Natural time analysis: the new view of time: precursory seismic electric signals, earthquakes and other complex time series*. [S.l.]: Springer Science & Business Media, 2011.

Warren Liao, T. Clustering of time series data—a survey. *Pattern Recognition*, v. 38, n. 11, p. 1857–1874, 2005. ISSN 0031-3203. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0031320305001305>.

YAO, Q. et al. *Taking Human out of Learning Applications: A Survey on Automated Machine Learning*. 2019.

ZHENG, Z. A comparison of constructing different types of new feature for decision tree learning. In: *Feature Extraction, Construction and Selection*. [S.l.]: Springer, 1998. p. 239–255.