INSTITUTO LATINO-AMERICANO DE CIÊNCIAS
DA VIDA E DA NATUREZA (ILACVN)

ENGENHARIA FÍSICA

DETECÇÃO AUTOMÁTICA DE EXOPLANETAS POR MEIO DE ALGORITMOS DE APRENDIZADO DE MÁQUINA BASEADOS EM SÉRIES TEMPORAIS

PATRICIA OLIVEIRA MONTANGER

Foz do Iguaçu
2021

INSTITUTO LATINO-AMERICANO DE CIÊNCIAS
DA VIDA E DA NATUREZA (ILACVN)

ENGENHARIA FÍSICA

DETECÇÃO AUTOMÁTICA DE EXOPLANETAS POR MEIO DE ALGORITMOS DE APRENDIZADO DE MÁQUINA BASEADOS EM SÉRIES TEMPORAIS

PATRICIA OLIVEIRA MONTANGER

Trabalho de Conclusão de Curso apresentado ao Instituto Latino-Americano de Ciências da Vida e da Natureza da Universidade Federal da Integração Latino-Americana, como requisito parcial à obtenção do título de Bacharel em Engenharia Física.

Orientador: Prof. Dr. Willian Zalewski

Foz do Iguaçu
2021

PATRICIA OLIVEIRA MONTANGER

DETECÇÃO AUTOMÁTICA DE EXOPLANETAS POR MEIO DE ALGORITMOS DE APRENDIZADO DE MÁQUINA BASEADOS EM SÉRIES TEMPORAIS

Trabalho de Conclusão de Curso apresentado ao Instituto Latino-Americano de Ciências da Vida e da Natureza da Universidade Federal da Integração Latino-Americana, como requisito parcial à obtenção do título de Bacharel em Engenharia Física.

BANCA EXAMINADORA

_____
Orientador: Prof. Dr. Willian Zalewski
UNILA

_____
Prof. Dra. Dáfni Fernanda Zenedin Marchioro
UNILA

_____
Prof. Dr. Marcelo Nepomoceno Kapp
UNILA

Foz do Iguaçu, _____ de _____ de _____.

TERMO DE SUBMISSÃO DE TRABALHOS ACADÊMICOS

Nome completo do autor(a):_____

_____

Curso: _____

	Tipo de Documento
(x) graduação	(…..) artigo
(…..) especialização	(x) trabalho de conclusão de curso
(…..) mestrado	(…..) monografia
(…..) doutorado	(…..) dissertação
	(…..) tese
	(…..) CD/DVD – obras audiovisuais
	(…..) _____

Título do trabalho acadêmico: _____

_____

_____

Nome do orientador(a): _____

Data da Defesa: _____/_____/_____

Licença não-exclusiva de Distribuição

O referido autor(a):

a) Declara que o documento entregue é seu trabalho original, e que o detém o direito de conceder os direitos contidos nesta licença. Declara também que a entrega do documento não infringe, tanto quanto lhe é possível saber, os direitos de qualquer outra pessoa ou entidade.

b) Se o documento entregue contém material do qual não detém os direitos de autor, declara que obteve autorização do detentor dos direitos de autor para conceder à UNILA – Universidade Federal da Integração Latino-Americana os direitos requeridos por esta licença, e que esse material cujos direitos são de terceiros está claramente identificado e reconhecido no texto ou conteúdo do documento entregue.

Se o documento entregue é baseado em trabalho financiado ou apoiado por outra instituição que não a Universidade Federal da Integração Latino-Americana, declara que cumpriu quaisquer obrigações exigidas pelo respectivo contrato ou acordo.

Na qualidade de titular dos direitos do conteúdo supracitado, o autor autoriza a Biblioteca Latino-Americana – BIUNILA a disponibilizar a obra, gratuitamente e de acordo com a licença pública *Creative Commons* **Licença 3.0 Unported.**

Foz do Iguaçu, _____ de _____ de _____.

_____

Assinatura do Responsável

À Maria Aparecida de Oliveira

À Nei Marcos Montanger

AGRADECIMENTOS

Ao professor Willian Zalewski pela dedicação com a qual conduziu a orientação deste trabalho, pela paciência e compreensão em todas as etapas do desenvolvimento, em especial pela oportunidade de realizar projetos de Iniciação Científica durante a graduação. Pessoa pela qual sou muito grata pela confiança depositada e pela oportunidade de ter compartilhado essas experiências.

Aos professores da Engenharia Física, por todos os conhecimentos compartilhados.

Aos meus pais, Maria Aparecida de Oliveira e Nei Marcos Montanger, por todas as oportunidades que me foram dadas, pelo incentivo, compreensão e apoio incondicional à todas as minhas escolhas.

Aos meus colegas de curso e amigos, Égon Piragibe, Karen Mantilla, Nicolás Molina, Paula Lima e Victor Wentz. Por todo o apoio e companheirismo durante esta jornada.

Ao C3SL pela infraestrutura computacional disponibilizada para a realização dos experimentos.

MONTANGER, Patricia Oliveira. **Detecção automática de exoplanetas por meio de algoritmos de aprendizado de máquina baseados em séries temporais**. 2021. 89 páginas. Trabalho de Conclusão de Curso (Graduação em Engenharia Física) – Universidade Federal da Integração Latino-Americana, Foz do Iguaçu, 2021.

RESUMO

Missões espaciais têm contribuído muito na coleta de dados e impulsionado novas descobertas sobre diversos fenômenos existentes no universo. Um desses fenômenos é o trânsito planetário, que quando identificado permite aos astrônomos realizar a descoberta de exoplanetas. Projetos como o Kepler possibilitaram o rápido armazenamento de uma grande quantidade de dados temporais, especialmente na forma de curvas de luz. Nesse contexto, tornou-se cada vez mais comum a criação de processos automáticos para a análise desses dados, como a aplicação de técnicas de aprendizado de máquina. No entanto, para compreender adequadamente os eventos contidos em informações temporais, é fundamental a utilização de métodos específicos para o tratamento desse tipo de dado. Portanto, neste trabalho, com o objetivo de identificar exoplanetas de modo automático a partir de curvas de luz, propusemos o estudo de algoritmos de aprendizado de máquina baseados em séries temporais. Realizamos uma avaliação experimental utilizando 5302 curvas de luz, das quais 2195 são rotuladas como exoplanetas e 3107 são representantes de outros objetos celestes. Também, avaliamos 11 algoritmos de classificação, sendo 6 algoritmos tradicionais e 5 baseados em séries temporais. Como resultado, verificamos que os melhores desempenhos, em termos de acurácia, foram dos algoritmos específicos para séries temporais, com destaque para o *Time Series Forest*.

Palavras-chave: Exoplanetas. Séries Temporais. Curvas de Luz. Aprendizado de Máquina.

MONTANGER, Patricia Oliveira. **Automatic detection of exoplanets using machine learning algorithms based on time series.** 2021. 89 páginas. Trabalho de Conclusão de Curso (Graduação em Engenharia Física) – Universidade Federal da Integração Latino-Americana, Foz do Iguaçu, 2021.

ABSTRACT

Space missions have contributed a lot to data collection and boosted new discoveries about several phenomena existing in the universe. One of these phenomena is planetary transit, which when identified allows astronomers to discover exoplanets. Projects like Kepler made it possible to quickly store a large amount of temporal data, especially in the form of light curves. In this context, it has become increasingly common to create automatic processes for the analysis of this data, such as the application of machine learning techniques. However, in order to properly understand the events contained in temporal information, it is essential to use specific methods for the treatment of this type of data. Therefore, in this work, with the objective of automatically identifying exoplanets from light curves, we proposed the study of machine learning algorithms based on time series. We performed an experimental evaluation using 5302 light curves, of which 2195 are labeled as exoplanets and 3107 are representatives of other celestial objects. We also evaluated 11 classification algorithms, 6 of which are traditional and 5 based on time series. As a result, we found that the best performances, in terms of accuracy, were for specific time series algorithms, with emphasis on the Time Series Forest.

Key words: Exoplanets. Time Series. Light curves. Machine learning.

MONTANGER, Patricia Oliveira. **Detección automática de exoplanetas mediante algoritmos de aprendizaje automático basados en series temporales.** 2021. 89 páginas. Trabalho de Conclusão de Curso (Graduação em Engenharia Física) – Universidade Federal da Integração Latino-Americana, Foz do Iguaçu, 2021.

RESUMEN

Las misiones espaciales han contribuido mucho a la recopilación de datos y han impulsado nuevos descubrimientos sobre varios fenómenos existentes en el universo. Uno de estos fenómenos es el tránsito planetario, que cuando se identifica permite a los astrónomos descubrir exoplanetas. Proyectos como Kepler permitieron almacenar rápidamente una gran cantidad de datos temporales, especialmente en forma de curvas de luz. En este contexto, se ha vuelto cada vez más común crear procesos automáticos para el análisis de estos datos, como la aplicación de técnicas de aprendizaje automático. Sin embargo, para comprender adecuadamente los eventos contenidos en la información temporal, es fundamental utilizar métodos específicos para el tratamiento de este tipo de datos. Por ello, en este trabajo, con el objetivo de identificar automáticamente exoplanetas a partir de curvas de luz, propusimos el estudio de algoritmos de aprendizaje automático basados en series temporales. Realizamos una evaluación experimental utilizando 5302 curvas de luz, de las cuales 2195 están etiquetadas como exoplanetas y 3107 son representantes de otros objetos celestes. También evaluamos 11 algoritmos de clasificación, 6 de los cuales son tradicionales y 5 basados en series temporales. Como resultado, encontramos que los mejores desempeños, en términos de precisión, fueron para algoritmos de series temporales específicos, con énfasis para el *Time Series Forest*.

Palabras clave: Exoplanetas. Series temporales. Curvas de luz. Aprendizaje Automático.

LISTA DE FIGURAS

LISTA DE TABELAS

SUMÁRIO

1 INTRODUÇÃO

O avanço tecnológico ocorrido nas últimas duas décadas, em termos de instrumentação astronômica e da capacidade de armazenamento de dados, possibilitou a varredura sistemática de galáxias e a coleta de uma vasta quantidade de dados, impulsionando novas descobertas sobre diversos fenômenos existentes no universo. Isso foi possível, em especial, pela contribuição de missões espaciais como a CoRoT (*COnvection ROtation et Transits planétaires*), Nuclear Spectroscopic Telescope Array (*NuSTAR*), Wide-field Infrared Survey Explorer (*NEOWISE*), Gaia, Hubble, Kepler e mais recentemente, a missão TESS (*Transiting Exoplanet Survey Satellite*). Os dados coletados nessas missões têm contribuído para o desenvolvimento de técnicas cada vez mais precisas e eficazes no estudo de diversas áreas do conhecimento astronômico (SOUZA, 2019). O satélite da missão Kepler, lançado em 2009, possibilitou a observação de aproximadamente 530.000 estrelas até o final da missão em 2018. Dentre as contribuições estão estudos da atividade estelar de estrelas do tipo solar, ciclos magnéticos, rotação diferencial e manchas solares.

O principal objetivo científico da missão Kepler foi a detecção de exoplanetas por meio da análise da intensidade luminosa das estrelas. Exoplanetas são planetas que estão fora do Sistema Solar, ou seja, orbitam estrelas que não o Sol. O estudo da variabilidade desse tipo de dado permite a caracterização de diversos parâmetros e comportamentos de objetos celestes (LOPES, C. E. F., 2013; BABU, 2012; RICHARDS, 2011). Quando registrada cronologicamente, a intensidade luminosa de um objeto celeste é denominada curva de luz. Em especial, as curvas de luz são utilizadas na identificação de exoplanetas por meio do método de trânsito planetário. Esse fenômeno consiste na passagem de um ou mais planetas, durante sua órbita, exatamente entre a estrela e o ponto observador. Quando isso ocorre, o brilho aparente da estrela sofre alterações, pois uma pequena fração de sua superfície permanece temporariamente oculta (CASTRILLÓN, 2010). Hoje, 76% (3333) dos exoplanetas encontrados a partir dos dados do Kepler foram descobertos pela análise do trânsito planetário.

Neste contexto, o contínuo armazenamento de dados, em especial na forma de curvas de luz, gerados pelos projetos de observação espacial tem possibilitado o rápido armazenamento de uma enorme quantidade de dados. Como consequência,

as técnicas de análise tradicionais, que baseiam-se principalmente na análise visual, tornaram-se inviáveis para a exploração sistemática dos dados adquiridos (BABU, 2012). Em especial, destaca-se a missão Kepler que capturou dados com cadência máxima de 30 minutos, totalizando 678 GB de dados ao final do projeto.

Considerando esse cenário, muitos estudos têm proposto o desenvolvimento de métodos computacionais para automatizar o processo de exploração de objetos celestes, em especial, da detecção de exoplanetas (BLOMME, 2012; RICHARDS, 2011; ARMSTRONG et al., 2017; HINNERS et al., 2018; MALIK et al., 2020; JARA-MALDONADO et al., 2020). Dentre os métodos propostos na literatura para auxiliar no processamento automático de grandes conjuntos de dados, destaca- se o processo de mineração de dados (FAYYAD et al.,1996; REZENDE, 2003). Uma das principais linhas de pesquisa nesse sentido tem sido a aplicação de algoritmos de inteligência artificial, especificamente, com técnicas de aprendizado de máquina (MITCHELL, 1997). Esses estudos têm apresentado resultados promissores em termos de acurácia das predições e velocidade de exploração dos dados. No entanto, uma importante limitação dessas abordagens consiste na utilização de descritores e/ou algoritmos que não permitem considerar a relação temporal existente entre cada uma das observações das curvas de luz.

A utilização de métodos de análise para o estudo de eventos e comportamentos contidos em dados temporais consiste em uma tarefa não trivial e dependente do domínio de aplicação (FERRERO, 2009; MALETZKE et al., 2014). A análise de séries temporais, como é o caso das curvas de luz, apresenta características específicas, as quais diferem de outros tipos de dados devido à relação de dependência existente entre as observações que compõem a série e à alta dimensionalidade frequentemente verificada nesses dados (LAXMAN; SASTRY, 2006). Diversos estudos na literatura de séries temporais já demonstraram a necessidade da elaboração de métodos que permitam tratar as correlações temporais existentes para que seja possível obter informações mais corretas e completas sobre dados dessa natureza (CASTRO, 2012). Portanto, o desenvolvimento de abordagens que permitam considerar a relação temporal intrínseca das curvas de luz poderia prover uma análise mais completa dos dados e, consequentemente, a construção de modelos de detecção de melhor desempenho (MORCHEN, 2006).

1.1 OBJETIVOS

Avaliando o cenário mencionado, os objetivos deste trabalho são:

Objetivo geral:

Contribuir para o processo de tomada de decisão de astrônomos, por meio da detecção automática de exoplanetas, utilizando algoritmos de aprendizado de máquina baseados em séries temporais.

Objetivos específicos:

1. Estruturar um banco de dados com curvas de luz provenientes do catálogo online *NASA Exoplanet Archive* e propor um processamento para estes dados.

2. Definir um índice de referência (*benchmark*) para avaliar o desempenho de algoritmos de aprendizado de máquina na tarefa de detecção de exoplanetas.

3. Construir e avaliar, experimentalmente, modelos de classificação baseados em técnicas de aprendizado de máquina específicos para séries temporais.

definidos sob a seguinte hipótese:

A utilização de algoritmos de aprendizado de máquina baseados em séries temporais, para a detecção de exoplanetas por meio de curvas de luz, é mais adequada em relação aos algoritmos de aprendizado tradicionais, devido à consideração de que as observações estão ordenadas no tempo, não permitindo assim que estas sejam tratadas como características singulares e independentes de ordem.

1.2 ORGANIZAÇÃO DO TRABALHO

Este trabalho está organizado do seguinte modo:

Capítulo 2 - Identificação de exoplanetas: neste capítulo inicialmente são apresentadas as principais técnicas utilizadas na classificação de exoplanetas, em especial a técnica utilizada neste trabalho, a de trânsito planetário. Posteriormente são apresentados os projetos espaciais de coleta de dados existentes e os descontinuados, assim como os conjuntos de dados disponíveis como resultados dessas missões, bem como uma revisão dos principais estudos atuais de identificação automática de exoplanetas.

Capítulo 3 - Mineração de dados em séries temporais: neste capítulo são apresentados conceitos de mineração de dados, tais como pré-processamento, extração de padrões e pós-processamento. Conceitos de aprendizado de máquina, tais como as formas de aprendizado supervisionado e não-supervisionado e os paradigmas dos algoritmos de classificação. Conceitos referentes às séries temporais, a amostragem, tendência, ruído, normalização e formas de representação. Posteriormente são introduzidos conceitos dos algoritmos de aprendizado de máquina e da classificação de séries temporais.

Capítulo 4 - Material e Método: neste capítulo apresentamos o *benchmark* e justificamos sua necessidade, posteriormente apresentamos os algoritmos de classificação selecionados para serem utilizados no trabalho, as medidas de desempenho, o método de avaliação, a validação estatística e a configuração experimental do trabalho.

Capítulo 5 - Resultados e Discussão: neste capítulo são apresentados os resultados obtidos e é realizada uma discussão sobre o método proposto e sobre a hipótese do trabalho.

Capítulo 6 - Conclusões e Trabalhos Futuros: neste capítulo são apresentadas as conclusões deste trabalho, as principais contribuições, as limitações e os trabalhos futuros.

2 IDENTIFICAÇÃO DE EXOPLANETAS

2.1 TÉCNICAS PARA IDENTIFICAÇÃO DE EXOPLANETAS

Os astrônomos desenvolveram uma grande variedade de métodos que podem ser usados na identificação de exoplanetas, aqueles encontrados fora do sistema solar. Alguns desses métodos possibilitam a determinação de diferentes propriedades dos exoplanetas, como a composição da atmosfera, sua temperatura, a massa da estrela hospedeira e do exoplaneta, entre outros. A combinação de diferentes métodos pode fornecer uma melhor caracterização das propriedades do exoplaneta e de suas estrelas (JARA-MALDONADO et al., 2020). O primeiro exoplaneta confirmado foi o 51 Pegasi b no ano de 1995 e desde então, até a publicação deste trabalho, através dos dados coletados pela NASA, já foram identificados 4.383 exoplanetas usando diferentes técnicas (BUGUEÑO; MENA; ARAYA, 2018).

A seguir estão descritos brevemente os métodos de detecção de exoplanetas mais comumente usados:

- **Trânsito planetário:** é realizada uma observação fotométrica da estrela, a partir da qual são detectadas as variações na intensidade de sua luz quando um planeta em órbita passa em sua frente, bloqueando uma fração da luz que a estrela emite. Este método permite detectar com eficiência planetas de alto volume, independentemente da proximidade do planeta à sua estrela hospedeira (BUGUEÑO; MENA; ARAYA, 2018).

- **Velocidade Radial:** consiste em analisar o efeito Doppler que é causado na estrela hospedeira pela mútua gravidade entre a estrela e o exoplaneta, ou seja, através da oscilação estelar, que pode ser detectada medindo os desvios para o vermelho ou para o azul. A combinação deste método com o de trânsito planetário fornece uma melhor caracterização de propriedades dos exoplanetas. Sua principal limitação é que as oscilações estelares causadas por exoplanetas são muito pequenas, o que dificulta sua detecção (JARA-MALDONADO et al., 2020).

- **Microlente gravitacional:** as trajetórias da luz são distorcidas por objetos massivos como estrelas ou planetas, esta distorção muda a direção da luz e gera um efeito de lente gravitacional na luz de uma estrela, conforme descrito em Treu et al. (2012). A técnica de microlente se baseia no fato de que a gravidade de um exoplaneta pode focar a luz de estrelas distantes para fazê-las parecer temporariamente mais brilhantes. As principais limitações deste método incluem o fato de que é improvável que ocorra o alinhamento necessário da estrela, e de que os astrônomos não podem prever onde ou quando os eventos de lente ocorrerão (YAQOOB, 2011; JARA-MALDONADO et al., 2020).

- **Imagem Direta:** esta técnica consiste em detectar uma posição espacial do exoplaneta e de sua estrela hospedeira, a fim de obter imagens dos exoplanetas sob determinadas condições. A imagem obtida é um pequeno ”ponto” conforme Yaqoob (2011), no entanto, pode ser usada para obter mais detalhes da composição química e temperatura do exoplaneta. De acordo com o arquivo de exoplanetas da NASA o primeiro exoplaneta encontrado com esta técnica é denominado 2MASS J12073346-3932539 b (CHAUVIN et al., 2004) e foi descoberto em 2004. A principal limitação desta técnica é a construção de instrumentos capazes de resolver o problema espacial entre exoplaneta e estrela (YAQOOB, 2011; JARA-MALDONADO et al., 2020).

Na Tabela 1 são fornecidas as porcentagens de detecção de exoplanetas para as técnicas que foram apresentadas nesta seção e para os demais métodos existentes. Como pode ser observado, o maior número de detecções foram obtidas utilizando o método de trânsito planetário.

Tabela 1 - Técnicas de detecção de exoplanetas.

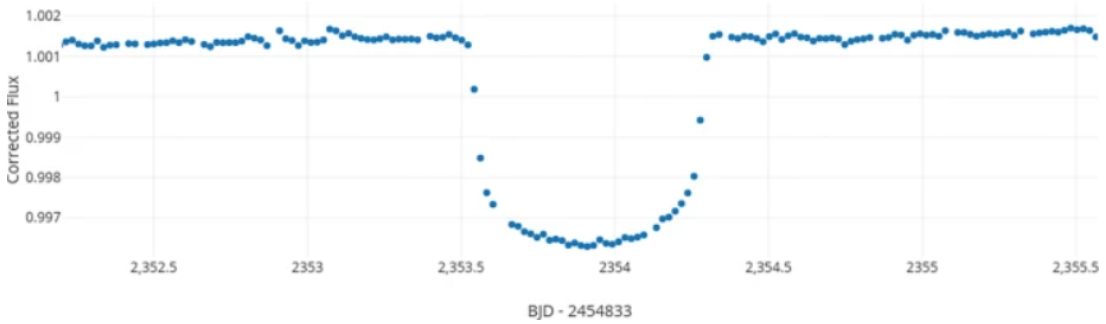
Técnica	Identificações
Trânsito planetário	76,10%
Velocidade Radial	19,10%
Microlente gravitacional	2,40%
Imagem Direta	1,20%
Outras técnicas	1,20%

Fonte: exoplanets.nasa.gov

2.2 TÉCNICA DE TRÂNSITO PLANETÁRIO

O trânsito planetário é um evento semelhante a um eclipse solar, quando um exoplaneta passa entre o observador e a estrela que orbita, é produzido um trânsito (JARA-MALDONADO et al., 2020), conforme podemos visualizar na Figura 1.

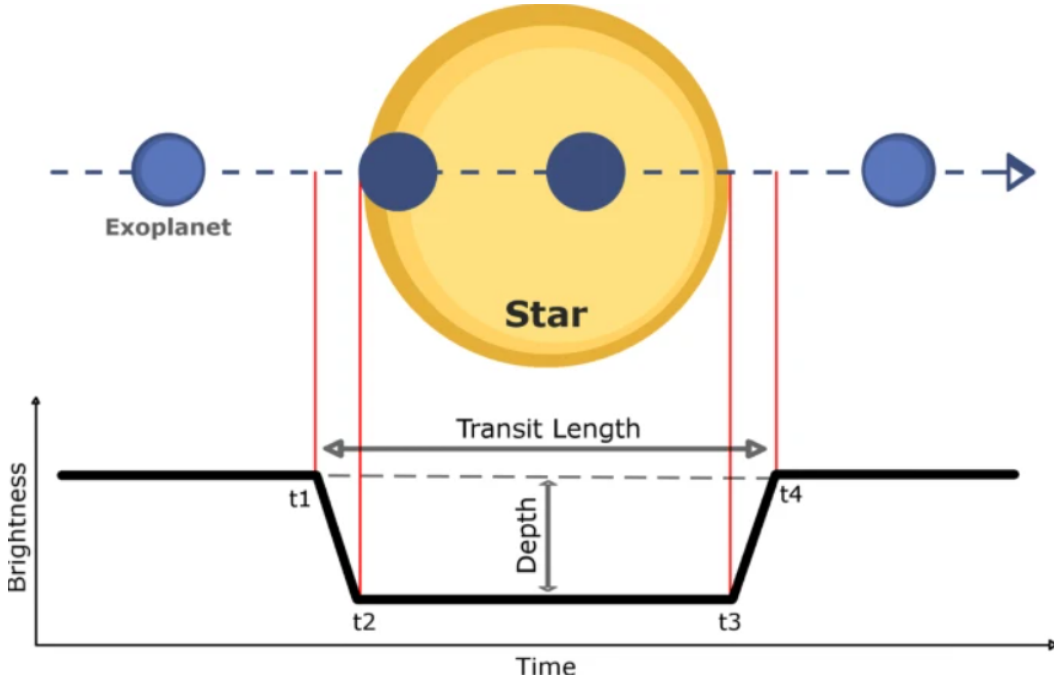
Figura 1 - Representação de trânsito planetário da estrela HIP 41378.



Fonte: JARA-MALDONADO et al. (2020).

Esses eventos podem ser estudados com o auxílio de curvas de luz, um tipo de série temporal onde estão representados os valores da intensidade de luz emitida pela estrela em função do tempo de observação. Conforme o exoplaneta passa na frente da estrela, a curva de luz apresenta uma diminuição na intensidade do brilho, o que representa que pode ter ocorrido um trânsito. A primeira identificação de exoplaneta por meio deste método ocorreu no ano de 1999, com a descoberta do planeta HD 209458b (CHARBONNEAU et al., 2000), já anteriormente descoberto por (HENRY et al. 2000) usando o método de velocidade radial. Um exemplo ideal de curva de luz é mostrado na Figura 2, na qual é possível observar que conforme o exoplaneta orbita a estrela, diferentes valores de luminosidade são obtidos. Alguns parâmetros que podem ser extraídos dessa curva de luz são: o início da entrada no trânsito (t1), o fim da entrada no trânsito (t2), o início da saída do trânsito (t3), o fim da saída do trânsito (t4), o comprimento total do trânsito e a profundidade do trânsito (JARA-MALDONADO et al. 2020).

Figura 2 - Trânsito planetário.



Fonte: JARA-MALDONADO et al. (2020).

A mudança no fluxo estelar durante o trânsito planetário é muito pequena, algumas vezes de aproximadamente 1% para exoplanetas classificados como Júpiteres quentes e de aproximadamente 0,01% para planetas semelhantes à Terra (GRZIWA e PATZOLD, 2016). Além disso, Mandel e Agol (2002) desenvolveram um modelo para simular o brilho estelar durante um trânsito, este modelo tem sido usado para criar dados artificiais para fins de experimentação. Os parâmetros que podem ser calculados através do método de trânsito planetário são: a profundidade do trânsito, o comprimento do trânsito, os tempos de entrada e saída do trânsito, a proporção entre o tamanho do planeta e o tamanho de sua estrela hospedeira e o período orbital. Ainda, através de uma solução analítica, sob certas condições, é possível determinar a massa estelar, o raio estelar e raio do planeta (JARA-MALDONADO et al., 2020).

A principal limitação deste método é que ele só pode ser utilizado para observar planetas cujo plano orbital é "orientado de modo que seja observado de lado, ou quase de lado" (YAQOOB, 2011), a menos que o planeta esteja tão perto de sua estrela que ainda apresente um trânsito mesmo em inclinações relativamente altas. Outra limitação é que esta técnica é tendenciosa para encontrar o maior exoplaneta do sistema em comparação com a estrela hospedeira, como afirmado em Yaqoob (2011). Existem outras limitações no trânsito planetário como por

exemplo lacunas de dados em medições terrestres, as quais são causadas pelo ciclo diurno e noturno (JARA-MALDONADO et al., 2020).

Como visto na seção anterior, a maioria dos exoplanetas foram encontrados usando a técnica de trânsito planetário e segundo Jara-Maldonado et al. (2020) isso se deve principalmente ao lançamento do telescópio Kepler da NASA no ano de 2009, pois o único instrumento científico do Kepler era um fotômetro que monitorava continuamente o brilho de milhares de estrelas em um campo de visão fixo, de forma que era possível encontrar exoplanetas procurando por pequenas quedas no brilho de uma estrela quando um planeta cruzava na frente dela.

2.3 PROJETOS ESPACIAIS

Atualmente estão em operação sete telescópios em missões espaciais lançadas pela NASA, são eles:

- Hubble
- Swift Gamma Ray Burst Explorer
- Transiting Exoplanet Survey Satellite (TESS)
- Chandra X-ray Observatory
- Fermi Gamma-ray Space Telescope
- Nuclear Spectroscopic Telescope Array (NuSTAR)
- Neutron star Interior Composition Explorer on the International Space Station
- Wide-field Infrared Survey Explorer (NEOWISE)

Dentre estes, TESS e Hubble são os principais exploradores de exoplanetas, o telescópio espacial Hubble completou 30 anos em órbita no ano de 2020 e foi um pioneiro na busca por planetas ao redor de outras estrelas, sendo inclusive utilizado para fazer alguns dos primeiros perfis de atmosferas de exoplanetas. Outro explorador espacial da NASA, já descontinuado, foi o telescópio espacial Kepler, que fez história com a descoberta de mais de 2.600 exoplanetas através da busca de pequenas quedas na luminosidade das estrelas à medida que os planetas cruzavam as suas faces, como citamos na seção anterior. Uma das descobertas revolucionárias do Kepler foi que planetas entre o tamanho da Terra e o tamanho de

Netuno são muito comuns, porém a maior parte das estrelas hospedeiras destes planetas estão a distância de centenas a milhares de anos-luz, tornando difícil a obtenção de observações para esses sistemas. Em sua primeira missão, do ano de 2009 a 2013, o Kepler monitorou mais de 150.000 estrelas, realizando um levantamento estatístico de trânsito planetário, o qual foi projetado para determinar a frequência de planetas do tamanho da Terra ao redor de outras estrelas, porém essa missão terminou em função de problemas técnicos que fizeram com que a espaçonave perdesse muito de sua capacidade de focar em uma direção. Em 2014, iniciou a segunda missão, batizada de K2, que continuou descobrindo exoplanetas. Apesar de ter sua capacidade direcional diminuída, o Kepler revelou milhares de exoplanetas orbitando estrelas em seu campo de visão de 115 graus quadrados, que cobria cerca de 0,25% do céu. O telescópio Kepler encerrou sua segunda missão em 2018 quando foi descontinuado, porém segue sendo grandemente creditado com a descoberta do maior número de exoplanetas de qualquer missão até agora. A grande quantidade de dados gerados pelo Kepler permite que ainda hoje sejam encontrados novos planetas.

Com o encerramento da missão K2 iniciou a missão do telescópio TESS que está conduzindo um levantamento de quase todo o céu em segmentos sequenciais, primeiro a cúpula de estrelas vistas do hemisfério sul e em seguida as do norte. Sua missão é encontrar planetas ao redor das estrelas mais brilhantes e mais próximas, novamente procurando por pequenas diminuições no brilho das estrelas quando um planeta cruza na frente delas. O TESS foi projetado para pesquisar mais de 85% do céu, o que representa uma área do céu 400 vezes maior do que a coberta pelo Kepler, para pesquisar planetas ao redor de estrelas próximas, isso significa dentro de cerca de 200 anos-luz de distância. As estrelas observadas pelo TESS são tipicamente de 30 a 100 vezes mais brilhantes do que as pesquisadas pelo Kepler e os planetas detectados ao redor dessas estrelas são, portanto, muito mais fáceis de caracterizar com observações de acompanhamento, resultando assim em medições refinadas de massas, tamanhos, densidades e propriedades atmosféricas dos planetas.

Em breve, será lançado o telescópio espacial James Webb, com instrumentos que serão capazes de identificar não apenas exoplanetas, mas também identificar características mais profundas, a fim de encontrar mundos pequenos, rochosos e habitáveis com uma atmosfera semelhante à do planeta Terra. Este novo telescópio

está programado para ser lançado na Guiana Francesa ainda em 2021 e no topo de sua placa de proteção solar estará o maior espelho primário já enviado ao espaço, com cerca de 6,5 metros de diâmetro, observado o universo em luz infravermelha. Espera-se que o telescópio James Webb se torne o principal observatório da década, estudando bilhões de anos de história do universo e chegando quase ao Big Bang, revelando detalhes da formação de sistemas planetários e mostrando a composição da atmosfera de exoplanetas.

Ainda, existem missões dedicadas à detecção de planetas realizadas por outras organizações e agências espaciais, como o telescópio CoRoT que operou do ano de 2006 a 2013, período no qual descobriu 34 novos exoplanetas. Esse projeto se deu através de uma parceria da agência espacial francesa com a agência espacial europeia (ESA) com o objetivo de monitorar de perto o evento de trânsito planetário e de detectar ondas acústicas geradas nas profundezas de uma estrela, que enviam ondas para sua superfície, alterando assim seu brilho. A natureza exata dessas ondulações permite que os astrônomos calculem a massa, a idade e a composição química precisa da estrela observada.

2.4 CONJUNTOS DE DADOS DISPONÍVEIS

Os conjuntos de dados coletados pelas missões de telescópios da NASA e de algumas outras organizações estão disponíveis em um catálogo online, o Nasa Exoplanet Archive (https://exoplanetarchive.ipac.caltech.edu) que suporta a visualização de dados através de imagens, espectros e séries temporais. Esse catálogo possui uma interface dedicada aos conjuntos de dados obtidos através da pesquisa de trânsitos planetários, onde estão incluídas as curvas de luz do Kepler, da missão K2, TESS, CoRoT e também dados de pesquisas de trânsitos planetários baseados em telescópios de observações terrestres como o TrES, KELT e UKIRT.

No total, considerando todos os projetos, são mais de 100 milhões de curvas de luz disponíveis para a pesquisa do trânsito planetário. O Nasa Exoplanet Archive dispõe seus principais dados em três grandes grupos:

- **Exoplanetas confirmados**

Exibe dados de todos os exoplanetas já confirmados e de suas estrelas hospedeiras, isso inclui sistemas atípicos, como planetas de flutuação livre e

aqueles com estrelas múltiplas. Estão incluídas informações do Kepler, K2 e TESS contendo dados dos sistemas planetários onde os parâmetros ou cálculos são combinados a partir de diferentes referências. Além de tabelas específicas para planetas confirmados descobertos com a técnica de microlente e para planetas confirmados descobertos com a técnica de imagem direta. Para cada exoplaneta presente nessas bases de dados é possível identificar seu nome, método, ano, instrumento e telescópio de descoberta, assim como o período orbital, raio, massa, profundidade de trânsito, duração de trânsito, dentre muitos outros parâmetros estelares. Ainda, selecionando individualmente cada um dos exoplanetas se tem acesso às curvas de luz disponíveis para aquele objeto.

- **Dados do Kepler**

Exibe apenas dados do Kepler, em especial os chamados Objetos de Interesse do Kepler (KOI) contendo informações não só dos objetos confirmados como exoplanetas, mas também daqueles classificados como falsos positivos e dos candidatos a exoplanetas. É possível encontrar os mesmos parâmetros citados no item anterior com a adição de colunas contendo o nome do exoplaneta (presente apenas nos confirmados), número de identificação do objeto, o Kepler ID, que é exclusivo para cada objeto e o KOI name, um número usado para identificar e rastrear um objeto de interesse. Também é possível adicionar uma coluna denominada trimestres a estes dados, os trimestres variam entre 1 e 17 e são representados de forma binária. De acordo com Twicken et al. (2016) o telescópio Kepler precisava ser girado em torno de seu eixo de mira em noventa graus a cada aproximadamente 93 dias, a fim de manter a orientação correta de seus painéis solares. Esse período de tempo corresponde a um estado de rotação particular que é conhecido como um "quarto", de modo que as estrelas alvo foram observadas ao longo do ano em quatro locais diferentes no plano focal devido à rotação trimestral. Em relação às curvas de luz, o conjunto de dados do Kepler é muito grande para ser exibido com eficácia em uma tabela interativa, portanto, é disponibilizado um formulário de pesquisa para inserir o Kepler ID e baixar ou visualizar as curvas do objeto escolhido.

- **Pesquisas de trânsito**

Assim como nas demais bases de dados é possível identificar diversos parâmetros estelares para dados do projeto KELT, projeto UKIRT, projeto XO, observações públicas da missão Kepler, observações do CoRoT e TrES. Neste grupo de dados é possível encontrar um total de 99.779.188 curvas de luz disponíveis, as quais podem ser copiadas ou visualizadas em um formulário de pesquisa para inserir o número de identificação do objeto desejado.

2.5 DETECÇÃO AUTOMÁTICA DE EXOPLANETAS

Pesquisas como as realizadas na missão TESS e em outras semelhantes, ainda dependem de análise manual, o trabalho de Yu et al. (2019) fornece uma boa visão geral desse processo. O autor explica que para o TESS, geralmente um grupo de especialistas elimina manualmente casos óbvios de falsos positivos, um processo que por si só pode levar alguns dias. Dos casos restantes, cada caso deve ser revisado por pelo menos 3 especialistas, este tipo de procedimento pode levar a divergências sobre um caso particular, visto que os especialistas podem não possuir a mesma definição para classificação. Por essas razões, os especialistas consideram necessário um sistema que possa ser confiável e selecionar repetidamente os candidatos de exoplanetas mais importantes, que podem, em seguida, ser revisados manualmente para confirmação em um estágio posterior.

Neste contexto, conforme abordado nos trabalhos sobre classificação de curvas de luz (HINNERS et al., 2018) ainda existem poucas aplicações de técnicas de aprendizado de máquina em astronomia, porém este número vem crescendo. Um dos primeiros exemplos foi o trabalho de Bailey et al. (2007) que realizou a classificação de supernovas utilizando uma base de dados artificial como treinamento para um algoritmo de classificação. Recentemente, no trabalho de Armstrong et al. (2017) voltado a classificação de exoplanetas, foram utilizados dados reais do telescópio Kepler, envolvendo a análise do trânsito planetário a partir de algoritmos não supervisionados como o Self Organising Map (SOM). Enquanto que em Karpenka et al. (2013) e Charnock & Moss (2017) foram utilizadas abordagens de aprendizagem profunda em curvas de luz de supernovas simuladas, no primeiro foi aplicada uma rede neural artificial perceptron (ANN) e no segundo foi aplicada a rede neural recorrente (RNN) memória de longo prazo (LSTM) que

classificou supernovas artificiais com uma alta taxa de sucesso. Inspirados nestes trabalhos com supernovas, na pesquisa de Hinners et al. (2018) foi utilizada uma abordagem LSTM RNN como tentativa de aplicação de aprendizado de máquina para classificação de curvas de luz com o propósito de caracterizar estrelas hospedeiras de exoplanetas, porém os resultados não foram satisfatórios, o que foi justificado pela presença de ruído e pela dispersão de dados existente nas curvas de luz, que de acordo com os autores desempenharam um grande papel na capacidade de classificar dados reais com técnicas de aprendizado de máquina.

Segundo Malik et al. (2020) um dos métodos mais comumente utilizados para a detecção automática de exoplanetas é o Box-fit Least Squares (BLS) (KOVÁCS et al., 2002), onde o algoritmo tenta ajustar um modelo predefinido aos dados e caso algum ajuste seja suficientemente adequado, então este pode ser revisado manualmente. No entanto, o BLS é limitado em termos de sinal de ruído e na cadência de dados, sendo assim vulnerável a detecção de falsos positivos criados por padrões de ruídos cósmicos aleatórios ou por alguma variabilidade estelar. Ainda, nos últimos anos, houve um interesse crescente na construção de um sistema automático de detecção de exoplanetas. Algumas das tentativas notáveis incluem o Robovetter (COUGHLIN et al. 2016), o projeto Autovetter (MCCAULIFF et al. 2015) e o (MISLIS et al. 2016), onde um modelo baseado em árvore de decisão foi treinado para verificação de exoplanetas. Também, abordagens baseadas em aprendizagem profunda têm sido exploradas na literatura. O trabalho mais notável nesta área foi feito por Shallue & Vanderburg (2017), onde eles introduziram uma nova arquitetura de aprendizado profundo, a Astronet. Sua abordagem e arquitetura de modelo foram adaptadas e aplicadas a dados da NASA como os TCEs existentes no catálogo de candidatos a exoplanetas e também a dados da missão K2 do Kepler (DATTILO et al., 2019) e dados do TESS (YU et al., 2019).

Recentemente, dentre os trabalhos que tratam da classificação de estrelas hospedeiras de exoplanetas, ainda são poucos os que utilizam dados temporais como parâmetro de entrada. Podemos citar o trabalho de Malik et al. (2020) que utiliza séries temporais reais e artificiais para realizar uma extração de características das curvas de luz e as utiliza como parâmetro de entrada para os algoritmos, os quais são baseados em técnicas de árvore de decisão. Também o trabalho de Jara-Maldonado et al. (2020) que utiliza séries reais e artificiais, inserindo nelas trânsitos planetários simulados para treinamento de algoritmos como

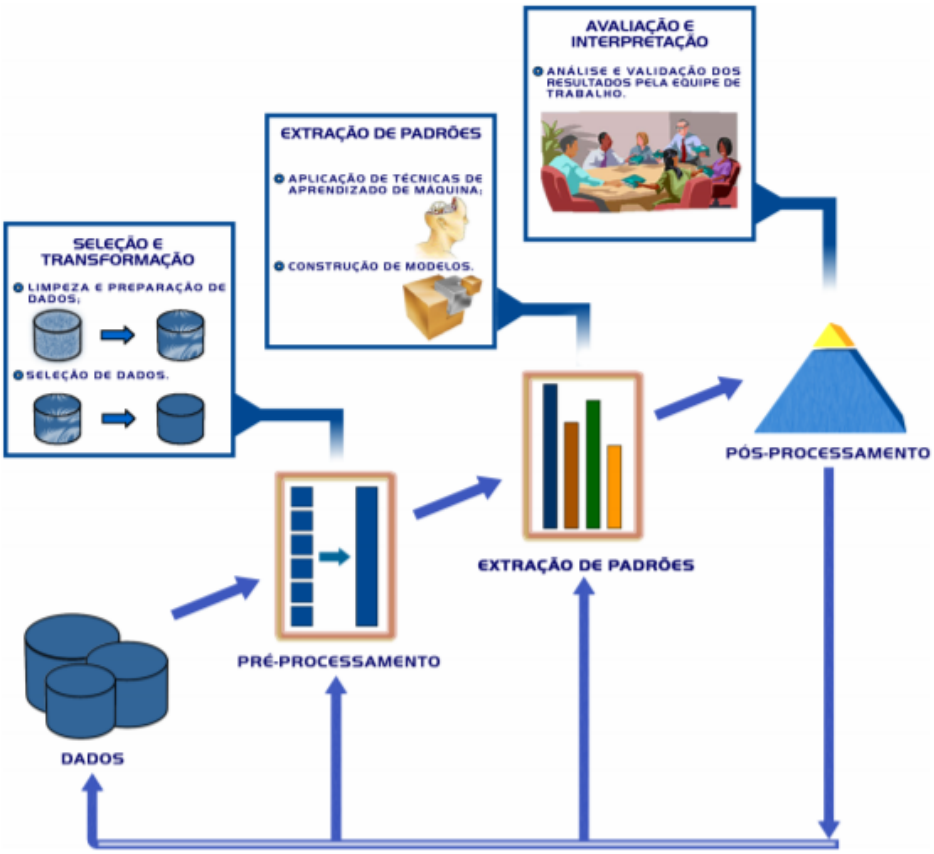
MLP, SVM, CNN, Random Forest, Naive Bayes e Least Squares. E os trabalhos de Montanger & Zalewski (2019) e Montanger & Zalewski (2020) que buscam desenvolver métodos para a análise de curvas de luz através da busca por shapelets e da aplicação de gráficos de recorrência, respectivamente.

3 MINERAÇÃO DE DADOS EM SÉRIES TEMPORAIS

3.1 MINERAÇÃO DE DADOS

O processo de mineração de dados tem atraído a atenção de pesquisadores de diversas áreas nos últimos anos, em especial devido à expansão da capacidade de armazenamento de dados e à complexidade de analisar esses dados por meio de técnicas tradicionais (ZALEWSKI, 2015). A partir desta necessidade, o desenvolvimento de métodos e processos computacionais que pudessem auxiliar no processamento automático ou semiautomático de grandes conjuntos de dados tornou-se uma importante ferramenta de análise. Nesse contexto, a mineração de dados tem como objetivo auxiliar pesquisadores e especialistas em tarefas de tomada de decisão por meio da extração de conhecimento sobre conjuntos de dados de um determinado domínio. Nesse cenário, o conhecimento pode ser caracterizado como a capacidade de relacionar informações por meio de modelos, que permitam descrever ou indicar as ações a serem realizadas (REZENDE, 2003). O processo de mineração de dados pode ser estruturado em três principais etapas: pré-processamento, extração de padrões e pós-processamento, as quais podem estar relacionadas de modo interativo e iterativo (FAYYAD et al., 1996; WEISS; INDURKHYA, 1998). Na Figura 3 podemos visualizar os processos que compõem o processo de mineração de dados.

Figura 3 - Processo de mineração de dados.



Fonte: FAYYAD et al. (1996), REZENDE (2003).

3.1.1 Pré-Processamento

O pré-processamento consiste em conhecer o domínio de aplicação e os tipos de dados a serem analisados, além de estar relacionado com a preparação dos dados para um formato que seja adequado para a etapa de extração de padrões (PYLE, 1999). São diversas as tarefas que podem ser aplicadas nesta etapa, como por exemplo (FACELI et al., 2011): integração de dados, transformação de dados, limpeza de dados e redução de dados, ou seja, seleção de atributos e redução de exemplos (ZALEWSKI, 2015).

3.1.2 Extração de Padrões

A extração de padrões consiste em pesquisar e desenvolver métodos para extrair o conhecimento que pode existir em um conjunto de dados (WITTEN;

FRANK, 2005). Para isso, é importante a tarefa de determinação dos algoritmos que serão aplicados aos dados pré-processados e ainda é necessário que sejam realizados ajustes nos parâmetros dos algoritmos escolhidos de forma a obter melhores resultados nos modelos construídos (MALETZKE, 2009).

3.1.3 Pós-Processamento

No pós-processamento é realizada a avaliação e a validação dos modelos produzidos na etapa anterior. Para isso é realizada a aplicação de medidas de desempenho e de qualidade, de testes estatísticos ou até mesmo a verificação com especialistas da área. Essas métricas definirão se os resultados são considerados realmente relevantes para o domínio da pesquisa realizada (ZALEWSKI, 2015).

3.2 APRENDIZADO DE MÁQUINA

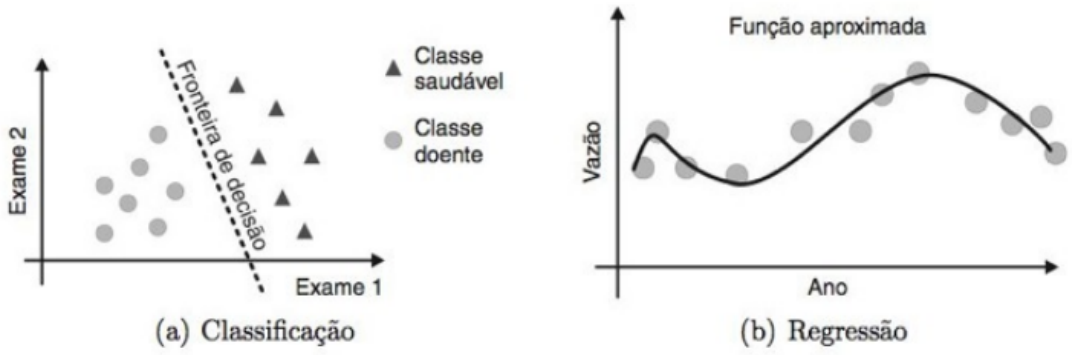
Como mencionado, em virtude da quantidade crescente de dados armazenados e, consequentemente, da complexidade dos problemas em diversas áreas de domínio , tornou-se cada vez mais importante o desenvolvimento de sistemas inteligentes que permitissem a aquisição de conhecimento de modo automático (ALPAYDIN, 2004). De acordo com Rezende (2003), estes sistemas dependem fortemente de conhecimento para solucionar problemas. Nesse contexto, a área de inteligência artificial tem recebido destaque na literatura dos últimos anos, especialmente, pela aplicação e desenvolvimento de técnicas de aprendizado de máquina (FACELI et al., 2011). O aprendizado de máquina pode ser entendido como "a capacidade de melhorar o desempenho na realização de alguma tarefa por meio da experiência" (MITCHELL, 1997). Especificamente, o objetivo desta área consiste em construir sistemas capazes de induzir hipóteses, de forma automática sobre um dado problema, utilizando a inferência indutiva, a qual permite obter um conhecimento novo partindo de um conjunto de exemplos conhecidos.

3.2.1 Tipos de Aprendizado Indutivo

O aprendizado indutivo pode ser realizado de diferentes modos de acordo com o tipo de tarefa a ser realizada, podendo este ser organizado em aprendizado

supervisionado e aprendizado não supervisionado. O aprendizado supervisionado é caracterizado pela utilização de exemplos do domínio estudado, chamado conjunto de treinamento, de forma que a classe de cada um dos exemplos é previamente conhecida. Nesta forma de aprendizado, os algoritmos de indução podem ser entendidos como uma função, para a qual um dado exemplo sem classe conhecida será vinculado a uma das classes já conhecidas pelo algoritmo (REZENDE, 2003). No caso do domínio de aplicação ser composto por dados de valores nominais, os algoritmos de indução produzem um classificador, conforme Figura 4 (a). E quando o domínio de aplicação é composto por dados de valores ordenados os algoritmos produzem um regressor, conforme Figura 4 (b).

Figura 4 - Gráfico ilustrativo das tarefas de classificação e regressão.



Fonte: FACELI et al. (2011)

Em relação ao aprendizado não supervisionado, a classe dos exemplos contidos no conjunto de treinamento não é conhecida, de forma que nesse tipo de aprendizado, as tarefas comumente abordadas são: sumarização, a qual constrói uma descrição representativa e compacta dos dados; associação, que realiza uma identificação de padrões frequentes que estabeleçam algum tipo de relação entre os atributos contidos no conjunto de exemplos; e agrupamento, no qual o objetivo é encontrar grupos de exemplos segundo alguma métrica de similaridade (FACELI et al., 2011).

Mediante as variadas tarefas, domínios e objetivos aplicáveis aos algoritmos de aprendizado de máquina, existem critérios que quando utilizados podem auxiliar na escolha do algoritmo mais adequado para cada situação. Logo, de acordo com o tipo de conceito utilizado para induzir uma hipótese, podemos estruturar os

algoritmos em paradigmas, os quais são apresentados brevemente a seguir (REZENDE, 2003):

- **Paradigma simbólico:** são algoritmos que realizam o processo de aprendizagem por meio de representações simbólicas, analisando exemplos e contra-exemplos. Geralmente estas representações estão na forma de expressão lógica, árvore de decisão ou rede semântica;

- **Paradigma baseado em exemplos:** esse tipo de algoritmo não constrói um modelo a partir de um conjunto de exemplos fornecidos, ao invés disso, o sistema armazena os exemplos e utiliza de uma medida de distância ou de dissimilaridade para identificar os exemplos mais similares ao exemplo a ser classificado (MITCHELL, 1997; MALETZKE, 2009). Um exemplo de algoritmos que corresponde a este paradigma é o k-vizinhos mais próximos (AHA; KIBLER; ALBERT, 1991; ZALEWSKI, 2015);

- **Paradigma estatístico:** consiste em utilizar modelos estatísticos para encontrar uma aproximação do conceito induzido. Esses algoritmos comumente assumem que os conjuntos de exemplos possuem uma distribuição conhecida e a partir dessa informação realizam uma inferência estatística (MALETZKE, 2009). No contexto deste paradigma, destacam-se os algoritmos de aprendizado Bayesiano (MITCHELL, 1997) e as máquinas de vetores de suporte (SVM) (CORTES; VAPNIK, 1995);

- **Paradigma conexionista:** o nome conexionismo descreve a área de estudo das redes neuronais artificiais, que são construções matemáticas inspiradas em conexões neurais do sistema nervoso humano (MALETZKE, 2009). As redes neurais artificiais (RNA) (HAYKIN, 1999) são exemplos de técnicas baseadas nesse paradigma (ZALEWSKI, 2015);

Adicionalmente, é importante ressaltar que os algoritmos de aprendizado de máquina utilizam distintos modos de representação para descrever uma hipótese, podendo assim ser divididos em sistemas de caixa preta e sistemas orientados ao conhecimento (MICHALSKI; BRATKO; KUBAT, 1998). Os sistemas de caixa preta

baseiam-se em suas próprias representações dos conceitos e não são facilmente interpretados por seres humanos, enquanto os sistemas orientados possibilitam a construção de representações simbólicas que sejam interpretáveis ao raciocínio humano (ZALEWSKI, 2015).

3.3 FUNDAMENTOS DAS SÉRIES TEMPORAIS

Os dados coletados ao longo do tempo podem ser representados por meio de séries temporais. A análise desses dados possui algumas particularidades quando comparada à análise de dados tradicionais, devido ao fato de que cada observação está relacionada temporalmente com as observações adjacentes. Portanto, essa relação entre as observações é relevante para ser considerada durante a análise desse tipo de dado (MALETZKE, 2009). Nesta seção, serão apresentados os principais conceitos e definições referentes a séries temporais, assim como suas formas de representação.

3.3.1 Definições

Uma série temporal pode ser definida como (ZALEWSKI, 2015):

Definição 1 (Série temporal): uma série temporal $T = \{t_{1, ...,} t_{i,} t_{j, ...,} t_{m}\}$ consiste em um conjunto de m valores ordenados, $m \geq 2$, tal que se $i < j$, t_i ocorre cronologicamente antes que t_j.

Associado a esse conceito, está o conceito das subsequências que compõe uma série temporal, sendo esta definida como:

Definição 2 (Subsequência): uma subsequência $S = \{t_p, ..., t_{p+n-1}\}$ consiste em um subconjunto contíguo de n valores de T com início na posição p, tal que $2 \leq n \leq m$ e $1 \leq p \leq m - n + 1$.

3.3.2 Principais Tarefas de Pré-Processamento

Dado o processo de mineração de dados em séries temporais, o pré-processamento consiste em uma das tarefas mais custosas e que tem impacto direto sobre a etapa de construção de padrões, a qual depende fortemente da qualidade dos dados utilizados (MICHALSKI; BRATKO; KUBAT, 1998; PYLE, 1999; HAN; KAMBER, 2006). Neste contexto, características como a alta dimensão dos dados e a relação existente entre as observações tornam não triviais quaisquer técnicas de transformação sobre esse tipo de dado. São apresentadas a seguir as tarefas comumente utilizadas na literatura para o pré-processamento de séries temporais (ZALEWSKI, 2015).

Amostragem

Um problema frequentemente presente nos diversos domínios que produzem dados na forma de séries temporais, está relacionado à amostragem irregular e aos valores faltantes dos dados. Essa característica pode influenciar negativamente no desempenho das técnicas de análise que assumem a presença de todos os dados ou que os mesmos são uniformemente espaçados em relação ao tempo. Para contornar esse tipo de problema podem ser aplicados métodos de previsão, como modelos auto-regressivos (MORETTIN; TOLOI, 2006), ou métodos de interpolação (MORCHEN, 2006; ZALEWSKI, 2015).

Tendência

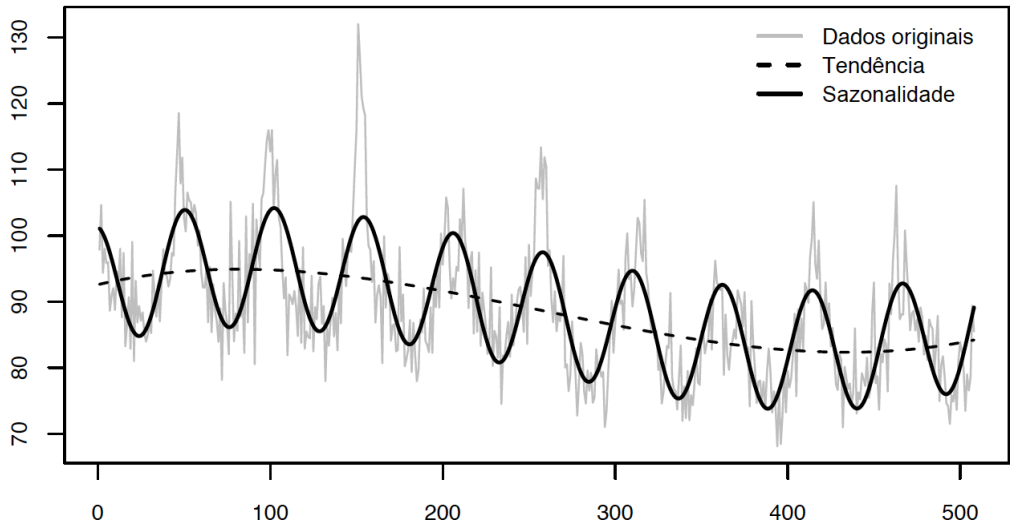
A componente de tendência afeta o comportamento dominante das séries temporais e desse modo pode influenciar algumas técnicas de análise que não permitem tratar essa característica (KEOGH; PAZZANI, 1999). Para remover a componente de tendência, comumente são aplicados métodos estatísticos (ZALEWSKI, 2015).

Ruído

Outro problema geralmente encontrado para a aplicação dos métodos de análise de séries temporais refere-se à presença de ruídos. Existem diversos métodos na literatura, que foram propostos para a remoção de ruídos, sendo cada qual melhor aplicado dependendo das características do domínio. As soluções mais comuns para a remoção de ruídos consistem em aplicar técnicas de filtragem de sinais, tais como médias móveis, suavização exponencial e diferenças de primeira ordem (SHUMWAY; STOFFER, 2006; MALETZKE, 2009; CASTRO, 2012; ZALEWSKI, 2015).

Na Figura 5 observamos a componente de tendência que é representada por uma linha tracejada e a série temporal do fenômeno que pode ser observada em cor cinza. Na série temporal original é possível identificar a presença de ruído, o qual pode afetar o reconhecimento do comportamento principal existente na curva, que neste caso é caracterizado pela curva de sazonalidade em preto (ZALEWSKI, 2015).

Figura 5 - Representação ilustrativa das componentes de tendência e sazonalidade.



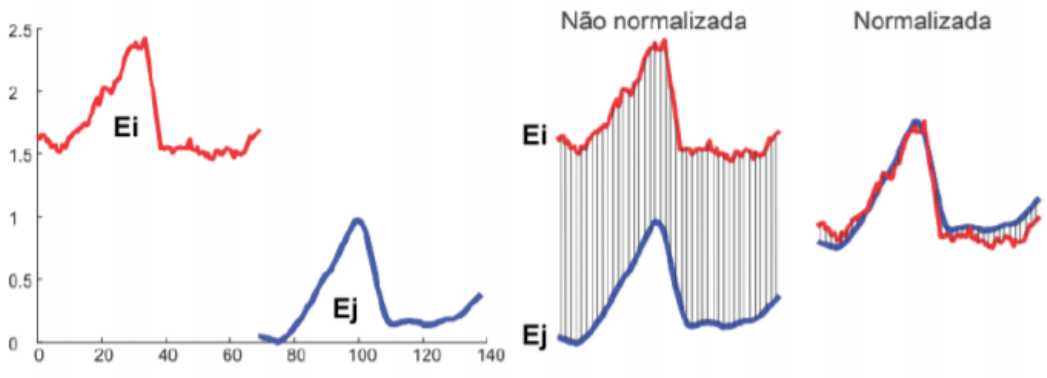
Fonte: FERRERO (2009).

Normalização

Um problema comum relacionado à operação de comparação entre séries temporais refere-se ao fato de que as séries podem estar representadas em diferentes níveis de escala, o que pode impactar na qualidade dos modelos

induzidos a partir destes dados. Uma das possíveis soluções consiste em aplicar técnicas de normalização sobre os dados das séries temporais a serem comparadas. Algumas técnicas tradicionais de normalização (KEOGH; LIN, 2005) são: de offset, de amplitude e de escala (ZALEWSKI, 2015).

Figura 6 - Exemplo de normalização de séries temporais.



Fonte: KEOGH e KASETTY (2002).

Formas de representação

Devido à característica de alta dimensionalidade frequentemente encontrada nas séries temporais aliada à heterogeneidade dos dados existentes em cada domínio, a utilização de métodos de aproximação pode ser fundamental para o correto desempenho das técnicas de análise (FALOUTSOS; RANGANATHAN; MANOLOPOULOS, 1994; LIN et al., 2003; CASTRO, 2012). Nesse contexto, a representação de séries temporais consiste em utilizar um conjunto reduzido de valores para caracterizar as séries, de modo que as informações relevantes sejam preservadas ou até mesmo ressaltadas. Em geral, algumas propriedades básicas devem estar presentes, em maior ou menor grau, nos métodos de representação: capacidade de redução da dimensionalidade dos dados; preservar características locais e globais; baixo custo computacional; baixo erro de reconstrução em relação à representação original; e capacidade de tratar dados com resíduos (ESLING; AGON, 2012; ZALEWSKI, 2015).

Definição 3 (Representação): Seja uma série temporal $T = \{t_1, ..., t_m\}$ de tamanho m, uma representação:

$$\hat{T} = \{\hat{t}_1, ..., \hat{t}_{m'}\} \quad (3.1)$$

de tamanho m', consiste em uma aproximação de T tal que m' ≤ m.

No âmbito da literatura de séries temporais diversas técnicas de representação têm sido propostas, cada qual com vantagens e desvantagens específicas dependendo do domínio de aplicação. De modo geral, é possível organizar essas técnicas, de acordo com o tipo de transformação utilizada, em três categorias: adaptativas, não adaptativas e baseadas em modelos (LIN et al., 2003; MORCHEN, 2006; FU, 2011; ESLING; AGON, 2012).

- **Representação não-adaptativa**

Os métodos baseados na abordagem não-adaptativa apresentam um processo de transformação que não leva em consideração informações dos dados das séries temporais. O método mais simples de representação dessa categoria é a amostragem (ASTROM, 1969), que consiste basicamente em selecionar n valores de uma série temporal T de tamanho m, tal que n << m. No entanto, esse método apresenta a desvantagem de provocar a distorção morfológica da série dependendo do valor utilizado para n. Um aprimoramento desse método é o piecewise aggregate approximation (PAA) (KEOGH et al., 2001a), o qual baseia-se em dividir a série em segmentos de mesmo tamanho e então utilizar a média aritmética dos valores dos segmentos consecutivos para a representação. Outra categoria de métodos dessa abordagem consiste em transformar as séries em outros domínios, como o método discrete fourier transform (DFT) (AGRAWAL; FALOUTSOS; SWAMI, 1993) e o discrete wavelet transform (DWT) (CHAN; FU, 1999).

- **Representação adaptativa**

Os métodos baseados na abordagem adaptativa apresentam um processo de transformação que considera alguma informação dos dados das séries temporais para produzir a representação. Em geral, a maioria dos métodos da abordagem não-adaptativa podem ser convertidos para adaptativa por meio da inclusão de uma estratégia que considere alguma informação sobre os dados (MÖRCHEN; ULTSCH, 2005). Um exemplo de algoritmo dessa estratégia é o adaptavive piecewise

constant approximation (APCA) (GEURTS, 2001; KEOGH et al., 2001b), que consiste em flexibilizar o tamanho dos segmentos para melhor ajustar as subsequências que apresentam maior variabilidade de valores. Outros métodos similares são: *piecewise constant approximation* (PLA) (SHATKAY; ZDONIK, 1996), *singular value decomposition* (SVD) (KORN; JAGADISH; FALOUTSOS, 1997), *perceptually important points* (PIPs) (BAO, 2008), o *derivative segment approximation* (DSA) (GULLO et al., 2009) e as *shapelets* (YE; KEOGH, 2009). Considerando que algumas técnicas tradicionais de análise de dados temporais foram desenvolvidas apenas para processar dados discretos, uma outra categoria de métodos da abordagem adaptativa consiste em produzir uma representação simbólica das séries temporais como o *symbolic aggregate approximation* (SAX) (LIN et al., 2007) e o *piecewise vector quantizatized approximation* (PVQA), foi apresentada em Megalooikonomou, Li e Wang (2004) para criar um dicionário de palavras para representar cada segmento da série.

- **Representação baseada em modelos**

Na abordagem baseada em modelos, os valores que compõem uma série temporal são gerados por um modelo base, com isso, a representação é realizada por meio dos parâmetros de ajuste para um determinado modelo de aproximação. Os métodos mais comuns dessa abordagem incluem a utilização de modelos estatísticos, como o ARMA (KALPAKIS; GADA; PUTTAGUNTA, 2001); cadeias de Markov para séries temporais simbólicas (SEBASTIANI; RAMONI, 2001); e hidden Markov models (HMM) (PANUCCIO; BICEGO; MURINO, 2002). Os métodos que utilizam a extração de características estatísticas também podem ser interpretados como um modelo base do processo gerador das séries (NANOPOULOS; ALCOCK; MANOLOPOULOS, 2001; WANG; HAN, 2004). No trabalho de (SILVA; SOUZA; BATISTA, 2013) os autores propuseram representar as séries temporais por meio de modelos, que podem ser interpretados como imagens, denominados *recurrence plots*.

3.4 APRENDIZADO DE MÁQUINA EM SÉRIES TEMPORAIS

Nesta seção são brevemente descritos os tipos de tarefas de aprendizado de máquina comumente aplicadas à análise de dados temporais (ZALEWSKI, 2015).

- **Recuperação por conteúdo:** o principal objetivo desta tarefa é consultar a existência de uma série temporal ou de suas similares em um conjunto de séries (ESLING; AGON, 2012). Sendo um dos principais aspectos desta tarefa a eficiência do processo de busca, em termos de tempo e qualidade dos resultados (grau de similaridade) (MORCHEN, 2006). Em Hetland (2004) são descritas diferentes abordagens de recuperação por conteúdo, bem como os critérios importantes que devem ser considerados ao se aplicar essa tarefa (ZALEWSKI, 2015).

- **Agrupamento:** essa tarefa consiste em definir um conjunto de grupos em um conjunto de séries temporais, de maneira que a similaridade entre as séries de um mesmo grupo sejam maximizadas e as de grupo diferentes minimizadas (ZALEWSKI, 2015). Definindo-se alguma abordagem para a representação das séries e uma medida de similaridade, os métodos de agrupamento tradicionais podem ser aplicados (HAN; KAMBER, 2006).

- **Descoberta de regras:** essa é uma das tarefas mais conhecidas em aprendizado de máquina, pois possibilita a representação do conhecimento por meio de abstrações que apresentam mais proximidade com o raciocínio lógico adotado pelos seres humanos (FACELI et al., 2011; ZALEWSKI, 2015). No contexto de séries temporais, este método exige a aplicação de algum tipo de transformação, pois os algoritmos foram desenvolvidos para tratar de sequências simbólicas (SRIKANT; AGRAWAL, 1996; MORCHEN, 2006).

- **Previsão de valores:** essa tarefa é uma das mais estudadas no contexto de séries temporais (MORETTIN; TOLOI, 2006; MORCHEN, 2006; ESLING; AGON, 2012). Dada uma série temporal a previsão de valores prediz uma quantidade de valores futuros para essa série. Os métodos tradicionais dessa aplicação são baseados em modelos estatísticos, porém as restrições relacionadas a esses métodos motivaram o desenvolvimento de abordagens baseadas em outras técnicas, tais como as de aprendizado de máquina (ZALEWSKI, 2015). As principais técnicas são as redes neurais (KOSKELA, 2003; YADAV; KALRA; JOHN, 2007), SVM (HERRERA et al., 2007), agrupamentos (SFETSOS; SIRIOPOULOS, 2004) e kNN (FERRERO, 2009).

- **Identificação de eventos:** duas principais sub tarefas podem ser relacionados a esta tarefa, a detecção de anomalias e a identificação de motifs. A detecção de anomalias (*discords*) (KEOGH et al., 2006) em uma série temporal consiste em identificar eventos que raramente ocorrem. Segundo Zalewski (2015) o método tradicional para a identificação desses eventos baseia-se na definição de um modelo que caracterize os eventos considerados comuns na série temporal. Desse modo, as subsequências que não forem adequadamente ajustadas ao modelo, segundo algum critério, são consideradas anomalias. Já a identificação de *motifs* consiste na identificação de eventos recorrentes ao longo da série temporal (MALETZKE, 2009).

- **Classificação:** essa tarefa consiste em determinar uma função que permita associar uma classe a uma série temporal não rotulada. A tarefa de classificação de séries temporais será melhor detalhada na próxima sessão, devido ao fato de ser utilizada como técnica base neste trabalho.

3.5 CLASSIFICAÇÃO DE SÉRIES TEMPORAIS

A classificação de séries temporais tem sido um importante desafio na mineração de dados nas últimas duas décadas (FAWAZ et al., 2019) e consiste em predizer a qual classe pertence uma determinada série temporal (CABELLO et al., 2020). De acordo com Zalewski (2015), formalmente, seja $\mathcal{T}$ o conjunto de todas as séries possíveis de um determinado domínio e seja C = {c_1, ..., c_w } um conjunto de w classes, tal que:

$$\forall\, T_i \in \mathbb{T} :\ ((T_i \in c_1\ ) \vee\ \ldots\ \vee (T_i \in c_w)) \wedge (T_i \in c_j \rightarrow T_i \notin c_k, j \neq k) \quad (3.2)$$

um classificador de séries temporais consiste em uma função f que permite mapear uma série Ti ∈ T para uma classe c ∈ C:

$$\mathbf{f} : \mathbb{T} \rightarrow \{c_1, \ldots c_w\} \quad (3.3)$$

Os métodos propostos na literatura para a classificação de séries temporais estão fortemente relacionados ao tipo de representação utilizada, conforme citamos na seção anterior. Nesse contexto, os algoritmos de classificação podem ser estruturados em distintas abordagens segundo suas estratégias de classificação (BAGNALL et al., 2016):

- **Baseados em distância**

Nesta abordagem, os algoritmos de classificação utilizam uma métrica de distância para calcular a similaridade entre as séries temporais, a qual é utilizada para determinar a que classe um novo exemplo será vinculado. A maior parte das estratégias propostas segundo essa abordagem consiste em utilizar o algoritmo *k-Nearest Neighbors* (k=1) em combinação com alguma medida de distância. Essa estratégia, utilizando a *Dynamic Time Warping* (DTW) já foi considerada o estado da arte na literatura de séries temporais (BAGNALL et al., 2017). Outras medidas comumente utilizadas na literatura incluem *edit distance with real penalty* (ERP), *longest common subsequence* (LCSS), *Weighted* DTW (WDTW), *Time Warp Edit* (TWE) e *Move-Split-Merge* (MSM), *Complexity Invariant Distance* (CID), a *Derivative* DTW (DD_{DTW}) e a *Derivative Transform Distance* (DTD_C) (BAGNALL et al., 2016). Uma outra estratégia é o *Elastic Ensemble* (EE), que consiste em uma combinação de classificadores kNN com diferentes medidas de distância. Dentre os estudos que propuseram a utilização de indutores distintos do kNN e baseados em distância, citam-se: *Proximity Forest* , *Proximity Tree* e o *Proximity Stump* (LUCAS et al., 2019).

- **Baseados em dicionário**

As abordagens baseadas em dicionário aproximam e reduzem o dimensionalidade da série, transformando-as em um conjunto de símbolos (palavras) representativos, os quais são analisados em termos da distribuição de palavras. O processo central das abordagens de dicionário envolve a formação de palavras pela aplicação de uma janela deslizante ao longo de cada série, aproximando cada janela para produzir n valores e, em seguida, discretizando esses valores atribuindo cada um a um símbolo do alfabeto. Dentre os métodos baseados em dicionário estão o *Bag of Patterns* (BOP), *Symbolic Aggregate Approximation - Vector Space Model* (SAX-VSM), DTW *Features* (DTW_F), , *Bag of SFA Symbols* (BOSS), BOSS *Ensemble*, *Contractable BOSS* (SCHÄFER, 2015), *Word ExtrAction*

for time SEries cLassification (WEASEL) (SCHÄFER & LESER, 2017), *MUltivariate Symbolic Extension* (MUSE) (SCHÄFER & LESER, 2017), *Individual* TDE e o *Temporal Dictionary Ensemble* (MIDDLEHURST et al., 2021).

- **Baseados em *shapelets***

O método de *shapelets* se caracteriza pela extração de subsequências de séries temporais que permitam discriminar cada uma das classes envolvidas na classificação. As *shapelets* permitem a detecção de similaridades independente da fase dentre as séries existentes em uma mesma classe. O algoritmo original de *shapelets* desenvolvido por Ye e Keogh (2011) utiliza critérios de divisão em uma árvore de decisão. Dentre os principais métodos baseados em *shapelets* citam-se: *Fast Shapelets* (FS) (RAKTHANMANON & KEOGH, 2013), *Shapelet Transform* (ST) (BOSTROM & BAGNALL, 2017; HILLS et al., 2014) e *Learned Shapelets* (LS) (GRABOCKA et al., 2014), *Multi-resolution S*ymbolic sEQuence Learning (MrSEQL) (LE NGUYEN et al., 2019) e o *RandOm Convolutional KErnel Transform* (ROCKET) (DEMPSTER; PETITJEAN; WEBB, 2019).

- **Baseados em Intervalos**

Os métodos dessa abordagem baseiam-se na divisão das séries temporais em intervalos, para os quais são extraídas características, que são utilizadas como dados de entrada para os algoritmos de indução. Dentre os métodos dessa abordagem citam-se: *Time Series Forest* (TSF) (DENG et al., 2013), *Time Series Bag of Features* (TSBF) (BAYDOGAN, RUNGER e TUV, 2013), *Learned Pattern Similarity* (LPS) (BAYDOGAN & RUNGER, 2016), *Random Interval Spectral Forest* (LINES et al., 2018) e *Supervised Time Series Forest* (CABELLO et al., 2020).

- **Combinação de classificadores (*Ensemble*)**

As abordagens de combinação de classificadores têm se destacado, recentemente, nos problemas de classificação de séries temporais. Métodos como TSF, TSBF e BOSS são abordagens de combinação baseados no mesmo núcleo de classificação. Outro importante método da literatura é o *Collective of Transformation Ensembles* (COTE) (BAGNALL et al., 2015) que consiste em uma combinação de distintos métodos de extração de características com diferentes classificadores temporais. Em Lines et al. (2016) foi apresentada uma extensão do COTE,

denominada *Hierarchical Vote Collective of Transformation-Based Ensembles* (HIVECOTE) (BAGNALL et al., 2020), que atualmente representa o estado da arte da literatura de classificação de séries temporais.

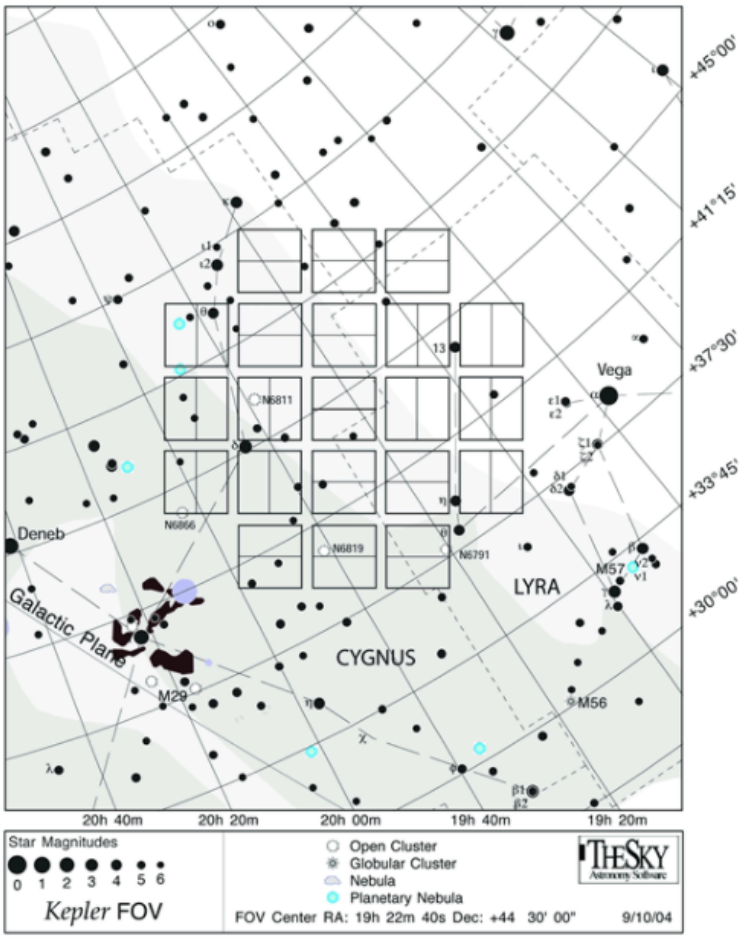
4 MATERIAL E MÉTODO

Neste capítulo é apresentado o método proposto para o desenvolvimento deste trabalho, partindo da coleta e pré-processamento dos dados, incluindo as técnicas aplicadas para evitar problemas de treinamento, como o sobre-ajuste, a escolha dos algoritmos de classificação, a configuração dos parâmetros para cada algoritmo e as métricas de avaliação de modelos utilizadas.

4.1 AQUISIÇÃO DO CONJUNTO DE DADOS

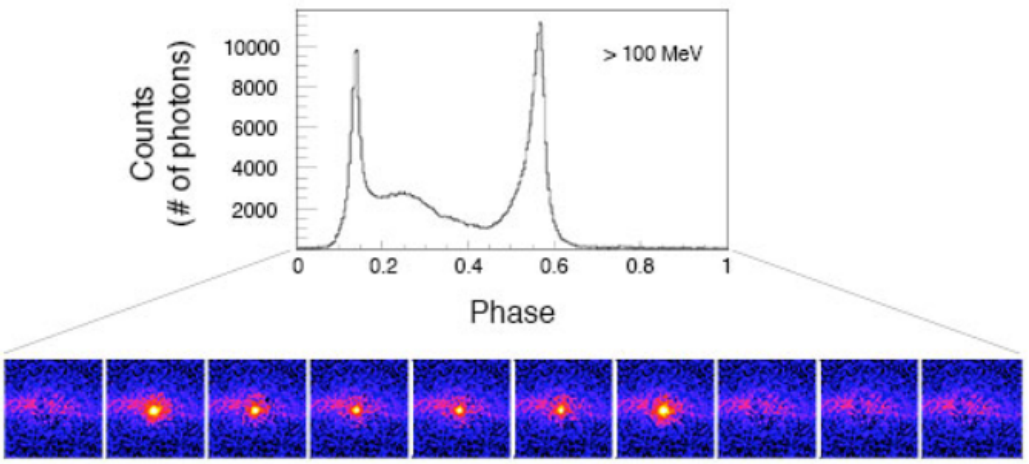
Dado que uma das características de um bom aprendizado de máquina é a utilização de uma base de dados adequada e bem estruturada, buscamos uma fonte com milhares de curvas de luz provenientes de estrelas, e adquiridas através das leituras realizadas pelo telescópio Kepler. Durante sua missão o Kepler realizou observações através de 21 pares de módulos (câmeras de carga acoplada), onde cada par consistia de quatro canais de 1100 x 2048 pixels, somando assim um total de 84 canais. Cada estrela observada pelo telescópio estava inserida em um desses canais, o qual poderia variar no tempo para a mesma estrela de acordo com o trimestre de observação, conforme ilustrado na Figura 7. Para a geração das curvas de luz com base no campo de observação do Kepler, era realizado um processo de captura dos pixels em que as estrelas apareciam ao longo do tempo, de modo que ao medir o brilho do objeto por unidade de tempo, era possível medir a luminosidade total de uma estrela, formando assim curvas de luz, como representado na Figura 8. Ainda, os registros de luminosidade poderiam ser feitos em diferentes cadências, a curta (58,85 segundos) e a longa (29,4 minutos) (HALL & BARENTSEN, 2020).

Figura 7 - Campo de visão do Kepler.



Fonte: *Kepler Instrument Handbook*.

Figura 8 - Imagens temporais de um objeto celeste e como elas se traduzem em uma curva de luz.



Fonte: Grodin et al. (2013)

Neste estudo escolhemos utilizar as curvas de luz de cadência longa e acessar os Objetos de Interesse do Kepler (KOI) por meio do *NASA Exoplanet Archive*, apresentado no Capítulo 2, que será a fonte para a construção dos conjuntos de curvas de luz utilizados nesta pesquisa. É importante ressaltar que a partir da aquisição das curvas de luz, é necessário submetê-las a um processo de limpeza e preparação para garantir que os dados sejam coerentes e adequados para a entrada dos algoritmos de aprendizado de máquina.

No catálogo online da NASA acessamos a página onde estão contidas as informações dos KOIs e selecionamos apenas as colunas de dados de interesse (*kepid, koi_disposition, koi_period, koi_time0bk, koi_duration* e *koi_quarters*). Realizamos então a aquisição de um arquivo em formato CSV contendo os nomes dos objetos (*kepid*), suas respectivas classes (*koi_disposition*), o intervalo entre consecutivos trânsitos (*koi_period*), o tempo correspondente ao centro do primeiro trânsito planetário detectado (*koi_time0bk*), o tempo de duração do trânsito (*koi_duration*) e os trimestres de leitura dos dados (*koi_quarters*), conforme representado na Tabela 2. Cada linha corresponde a um objeto celeste, somando um total de 9564 objetos.

Tabela 2 - Colunas de dados de interesse do catálogo da NASA.

kepid	koi_disposition	koi_period	koi_time0bk	koi_duration	koi_quarters
10797460	CONFIRMED	9.488.036	170.538.750	295.750	11111111111111111000000000000000
10797460	CONFIRMED	54.418.383	162.513.840	450.700	11111111111111111000000000000000
10848459	FALSE POSITIVE	1.736.952	170.307.565	240.641	11111110111011101000000000000000

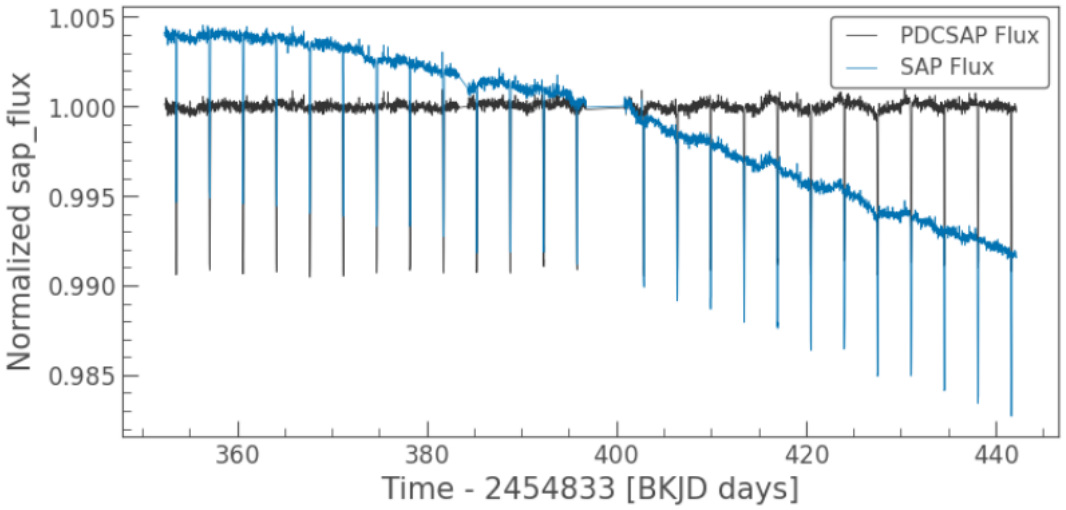
Fonte: exoplanetarchive.ipac.caltech.edu

A razão de termos selecionado esses parâmetros é a utilização da biblioteca *Lightkurve*, que possibilita a aquisição das curvas de luz correspondentes aos objetos presentes na base de dados. Esta biblioteca oferece uma interface para a manipulação dos dados de séries temporais obtidas por telescópios, em particular, das missões Kepler e TESS da NASA. A obtenção das séries temporais pode ser realizada a partir da função *search_lightcurve(kepid, author='Kepler', cadence='long', quarter=koi_quarter).download(),* na qual *kepid* corresponde ao nome do objeto e

koi_quarter ao trimestre de observação dos objetos. Os dados da missão Kepler estão organizados em trimestres, que variam entre 1 e 17 e são representados em formato binário. O número 1 representa que existe uma curva de luz disponível para aquele trimestre, enquanto o número 0 representa que não há uma curva para aquele trimestre, desta forma podemos ter diferentes curvas de luz para um mesmo objeto.

Após realizar a aquisição de uma curva com a biblioteca citada, temos disponíveis dois diferentes fluxos de luminosidade, o fluxo de fotometria de abertura simples (SAP) e o fluxo de condicionamento de dados pré-pesquisa (PDCSAP), que se diferenciam pelos níveis de tratamento realizados pelo *Kepler Data Processing Pipeline* da NASA. Ambos os fluxos de dados estão ilustrados na Figura 9, o fluxo SAP é afetado por efeitos sistemáticos, como mudanças no foco do telescópio, alterações de temperatura e o efeito DVA que leva uma estrela a um desvio no curso do trimestre. Enquanto que o fluxo PDCSAP tem seu processamento projetado especificamente para a detecção de exoplanetas. Desse modo, neste trabalho, escolhemos trabalhar com o fluxo PDCSAP.

Figura 9 - Fluxos SAP e PDCSAP.

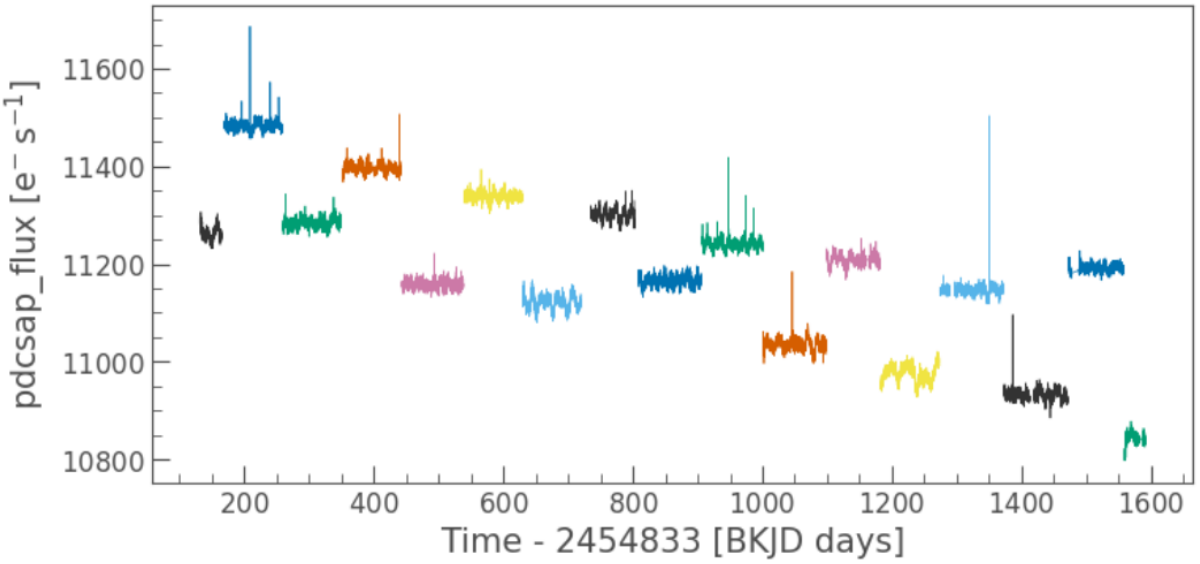


Fonte: HALL e BARENTSEN (2020)

Neste trabalho, para representar um determinado objeto *kepid* utilizamos as as curvas de luz disponíveis em todos os 17 trimestres por meio da método *download_all()*. A construção de curvas de luz, com todos os dados disponíveis, baseiam-se no fato de que o trânsito planetário representa apenas uma pequena

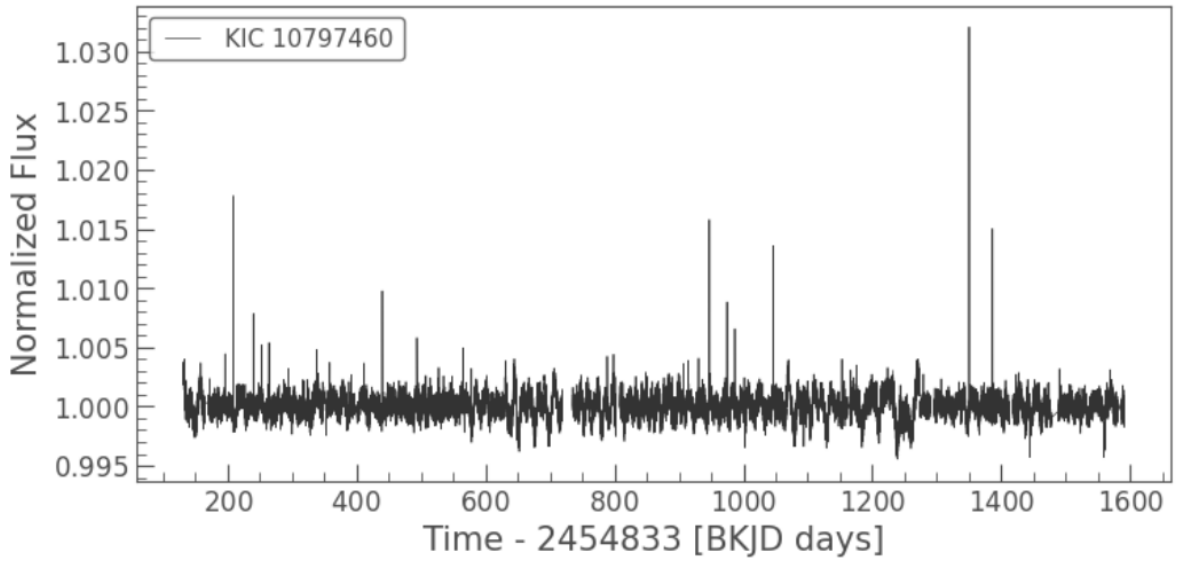
parte do sinal, o que pode inviabilizar a identificação somente em dados trimestrais. No entanto, devido ao movimento orbital do satélite e consequentemente do reposicionamento dos CCDs a cada trimestre, diferentes níveis de intensidade luminosa são registrados conforme pode ser observado na Figura 10. Para remover essa componente de tendência do sinal é utilizada a função *stitch()*, a qual concatena todos os trimestres e os normaliza ao mesmo tempo, deixando as profundidades dos trânsitos sob a mesma escala, como ilustrado na Figura 11.

Figura 10 - Curvas de luz por trimestre do objeto Kepler-227.



Fonte: autora.

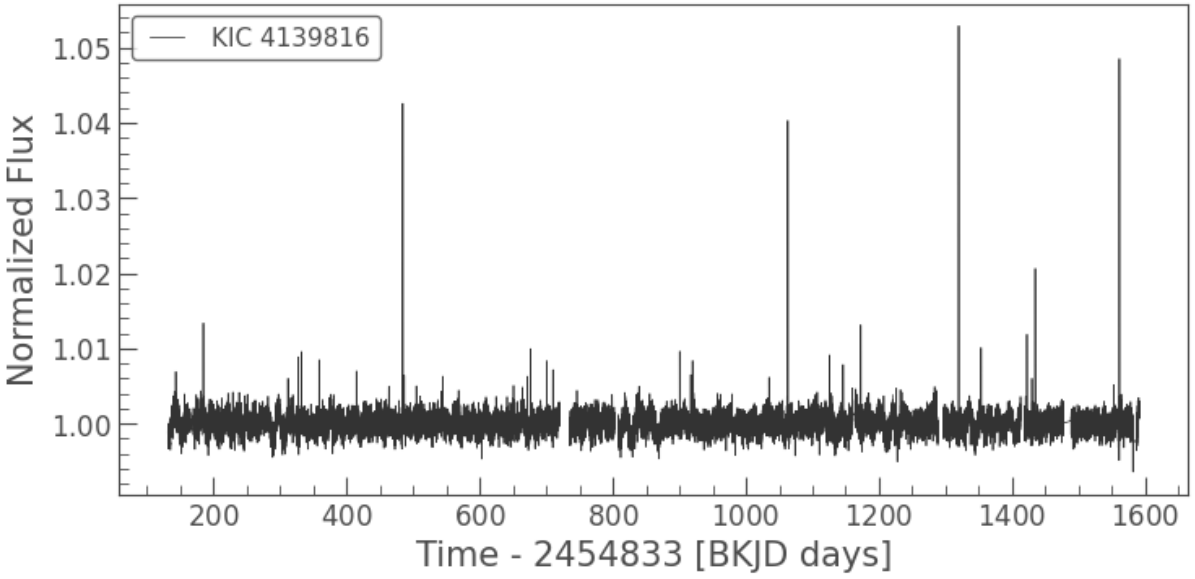
Figura 11 - Curva de luz concatenada do objeto Kepler-227.



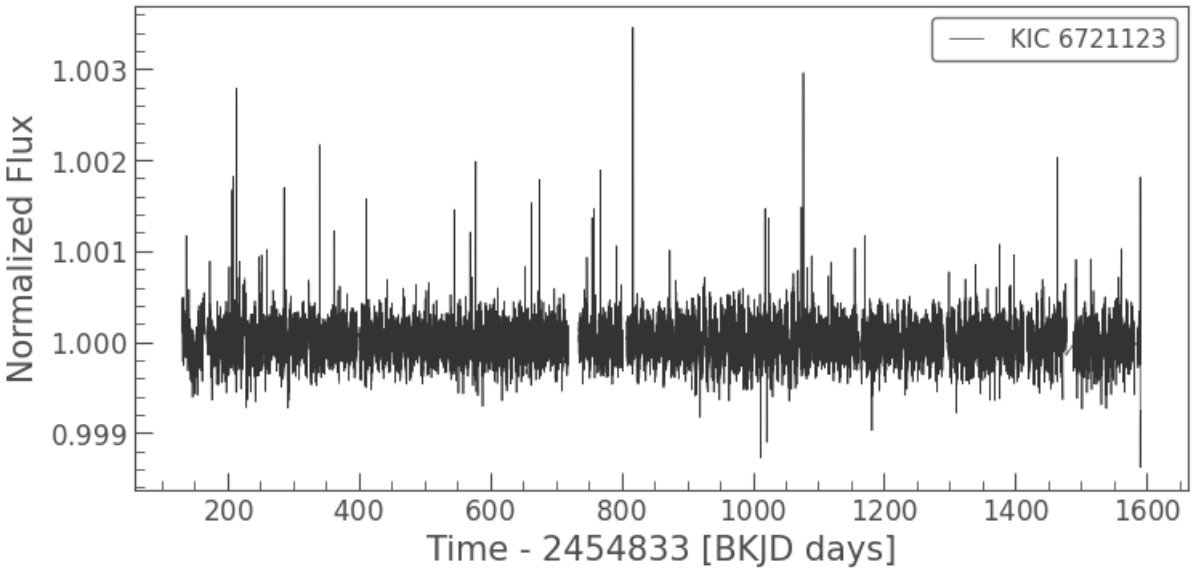
Fonte: autora.

Na Figura 12 trazemos exemplos de curvas de luz concatenadas nos 17 quarters para um exoplaneta confirmado, Figura 12 (a) e para um falso negativo, ou seja, não exoplaneta, Figura 12 (b).

Figura 12 - Curva de luz concatenada nos 17 quarters para (a) exoplaneta confirmado e (b) falso negativo.



(a)



(b)

Fonte: autora.

Ao final do processo de aquisição das curvas de luz, é possível identificar no conjunto de dados três diferentes classes, a de confirmados, candidatos e falsos

positivos. No entanto, como o nosso interesse é construir modelos de classificação para identificar curvas de luz de estrelas que contenham exoplanetas, os objetos classificados como candidatos não foram utilizados neste momento. Além disso, identificamos alguns valores ausentes na coluna *koi_quarters* e, sem essa informação não é possível acessar as curvas de luz, portanto essas linhas foram removidas, restando assim um total de 5302 objetos, sendo 3107 (58,60%) falsos positivos e 2195 (41,40%) confirmados. Considerando os 17 trimestres, cada um dos objetos possui aproximadamente 60 mil pontos de leitura.

4.2 PRÉ-PROCESSAMENTO DOS DADOS

Após importar as curvas de luz realizamos algumas tarefas de pré-processamento como a remoção de outliers, normalização e transformação das tabelas para a representação atributo-valor.

- **Valores ausentes e remoção de *outliers***

Ao explorar nosso banco de dados encontramos alguns valores ausentes nas curvas de luz, o que poderia interferir na utilização das séries temporais como entrada para os algoritmos de classificação. Portanto, ao identificar estes valores e suas respectivas posições, verificamos se sua quantidade era relevante a ponto de influenciar nos resultados e substituímos os valores ausentes através de uma interpolação linear. Ainda, realizamos uma remoção de *outliers* definindo um desvio padrão igual a 3, auxiliando assim na remoção de valores com erros de leitura instrumentais e alterações nas medições causadas por raios cósmicos (SHALLUE & VANDERBURG, 2017).

- **Normalização**

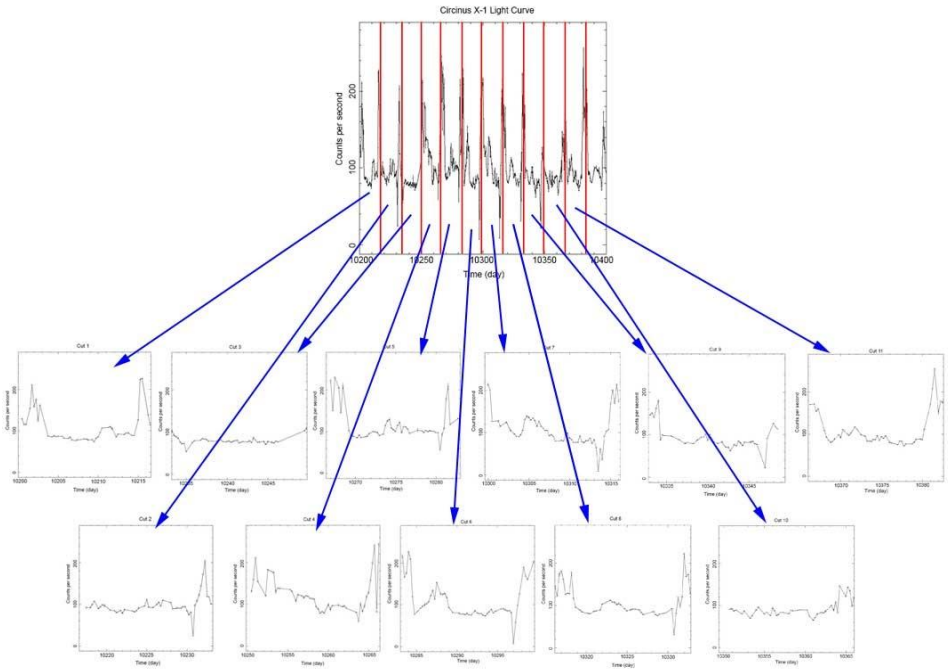
Normalizamos cada uma das curvas de luz presentes na base de dados utilizando a normalização de escala, nessa técnica de normalização todos os valores $t_i \in T$ são ajustados no intervalo [0, 1], por meio da equação 4.1. Os valores *min*(T) e *max*(T) correspondem ao menor e ao maior valor de T, respectivamente (ZALEWSKI, 2015).

$$t_i^{'} = \frac{(t_i - min(T))}{max(T) - min(T)} \tag{4.1}$$

- **Representação**

Na geração do conjunto de dados aplicamos o método *epoch folding* (métodos *fold* e *bin* da biblioteca *lightkurve*) nas curvas de luz, conforme descrito no trabalho de Shallue & Vanderburg (2017). O método *fold* consiste em "dobrar" cada curva de luz de acordo com seu período (*koi_period*) e realizamos esse processo aplicando a dobragem por época, neste caso com o tempo centrado no evento de trânsito (*koi_time0bk*). Na Figura 13 é possível visualizar uma curva de luz sendo fragmentada em várias partes de acordo com seu período. O método *bin* consiste em criar um único vetor unidimensional a partir das curvas dobradas e conforme proposto no trabalho de Shallue & Vanderburg (2017) adotamos duas representações para gerar essas curvas, uma representação local com tamanho limitado de 201 pontos e uma representação global com tamanho de 2001 pontos.

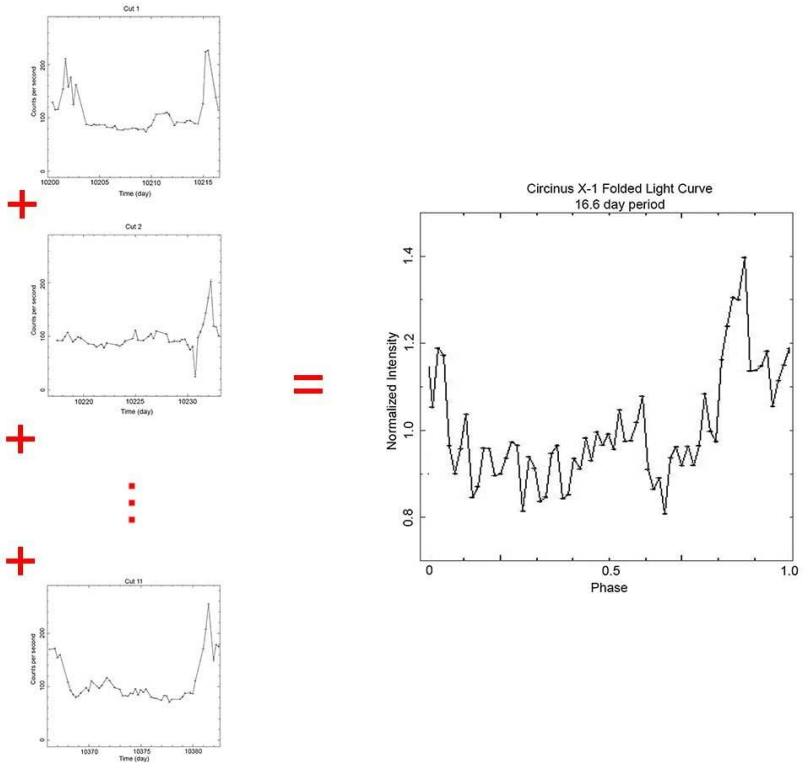
Figura 13 - Curva de luz segmentada de acordo com seu período.



Fonte: *NASA's Imagine the Universe.*

Na Figura 14 é ilustrado um exemplo desse processo que termina em uma única curva de luz, na qual são combinadas as intensidades das curvas de luz fragmentadas.

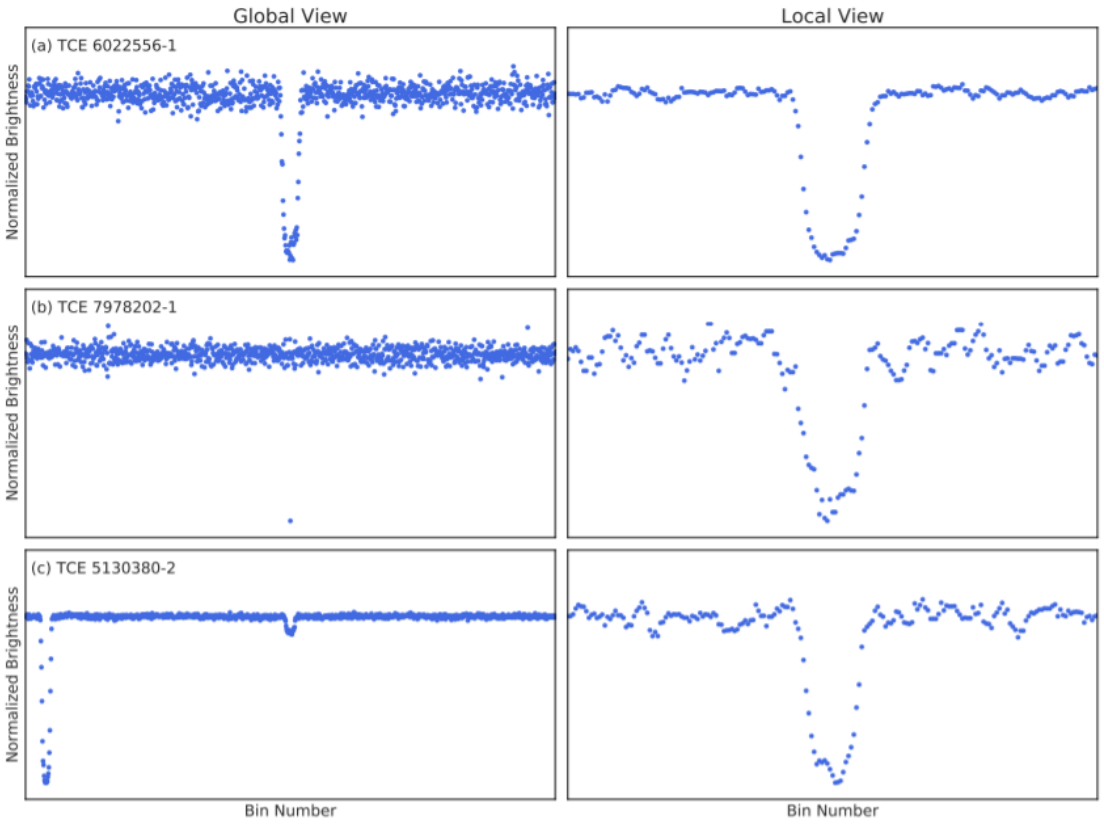
Figura 14 - Combinação das curvas fragmentadas formando uma única curva do tamanho do período.



Fonte: *NASA's Imagine the Universe.*

A representação global contempla toda a curva de luz, enquanto que a representação local é uma janela ao redor de um trânsito detectado na curva analisada. Conforme ilustrado na Figura 15.

Figura 15 - Curvas de luz na representação global e local.



Fonte: SHALLUE & VANDERBURG (2017)

Para que seja possível aplicar os algoritmos de classificação a nosso conjunto de dados é necessário que os dados estejam dispostos em um formato adequado, que no nosso caso é a representação atributo-valor. Essa representação descreve os exemplos contidos no conjunto de dados em uma tabela, sendo que as colunas representam os atributos e os valores em cada linha representam os valores dos atributos (fluxos de intensidade luminosa) (MALETZKE, 2009).

4.3 BENCHMARK

Na análise de dados é importante comparar os resultados obtidos entre diferentes métodos e diferentes algoritmos de aprendizado de máquina. Com o intuito de estabelecer uma base de comparação em relação às outras abordagens baseadas em séries temporais, utilizamos como *benchmark* alguns algoritmos tradicionais de aprendizado de máquina disponibilizados na biblioteca *scikit-learn* (PEDREGOSA et al., 2011) para Python. Neste contexto, apresentamos uma breve descrição dos algoritmos selecionados:

- ***Support Vector Machines* (SVM)**

O SVM é um algoritmo de aprendizado de máquina supervisionado que pode ser utilizado para classificações, regressões e outras tarefas. Como o foco deste trabalho é a classificação, utilizamos a implementação *Support Vector Classification* (SVC), a qual é baseada no trabalho de Chang et al. (2021). O que caracteriza este algoritmo é o fato de não ser paramétrico e com isso identificar um hiperplano que busque maximizar a distância entre o hiperplano encontrado e pontos de diferentes classes do conjunto de dados fornecidos. Para obter o melhor ajuste é possível alterar alguns parâmetros do SVC, sendo o mais importante deles o parâmetro “C”, quanto maior seu valor, menores são as margens de erro de classificação e quanto menor seu valor maior a margem de erro na classificação (MANRY; STURROCK; RAFIQI, 2019).

- ***Decision Trees* (DT)**

Conforme o trabalho de Barbosa et al. (2012) em uma árvore de classificação o que temos é o resultado de se fazer uma sequência ordenada de perguntas, e as perguntas feitas a cada passo na sequência dependem das respostas às perguntas anteriores. O ponto de partida de uma árvore de classificação é chamado de nó raiz e consiste em todo o conjunto de aprendizado, esse nó está no topo da árvore. Um nó é um subconjunto do conjunto de atributos, e pode ser terminal ou não-terminal. Um nó não-terminal é um nó que se divide em nós filhos. Tal divisão é determinada por uma condição sobre o valor de um único atributo que vai dividir os exemplos de acordo com a condição, em outros nós. A definição da melhor condição de divisão dos exemplos é baseada em uma métrica, como exemplo o ganho de informação. Um nó que não se divide é chamado de nó terminal, e a ele é atribuída uma classe.

- ***Ensemble - Random Forests* (RF)**

O objetivo dos métodos de ensemble é combinar os resultados de vários classificadores a fim de melhorar a generalização sobre um único classificador. Neste sentido, optamos por utilizar o método de ensemble com o algoritmo *Random Forests*, o qual consiste em um conjunto de árvores de decisão, para o qual subconjuntos da base de treinamento são selecionadas aleatoriamente, o que auxilia na generalização do modelo. O resultado final é obtido através da combinação dos

resultados de cada árvore utilizada na tomada de decisão do algoritmo (MANRY; STURROCK; RAFIQI, 2019).

- ***Naive Bayes*** **(NB)**

O algoritmo de classificação *Naive Bayes* é segundo Pardo & Nunes (2002) baseado no Teorema de Bayes para realizar o cálculo das probabilidades necessárias para a classificação, o qual é uma equação que descreve a relação de probabilidades condicionais de quantidades estatísticas. Na classificação bayesiana, estamos interessados em encontrar a probabilidade de uma classe dadas algumas características observadas.

- ***Nearest Neighbors*** **(NN)**

O algoritmo de vizinhos mais próximos atribui um rótulo às observações não classificadas com base no rótulo das k observações conhecidas mais próximas da nova informação fornecida. Também denominado como kNN, este algoritmo é influenciado pelo parâmetro k e pela métrica de distância calculada entre os exemplos, logo bons resultados dependem da definição destes parâmetros (MANRY; STURROCK; RAFIQI, 2019).

- ***Multilayer Perceptron*** **(MLP)**

Conhecido como MLP, este algoritmo é o mais conhecido e também o mais usado tipo de rede neural. Sua arquitetura é denominada *feed forward*, pois o treino do modelo é iterativo e em cada etapa de tempo os parâmetros são atualizados através das derivadas parciais da função de perda. A vantagem do perceptron sobre as demais redes se dá devido a utilização de funções de ativação não lineares. Quase qualquer função não linear pode ser usada para este algoritmos, exceto funções polinomiais. Atualmente, a função mais comumente usada é a sigmóide unipolar (ou logística) (POPESCU et al., 2009).

4.4 ALGORITMOS DE CLASSIFICAÇÃO BASEADOS EM SÉRIES TEMPORAIS

Dado que um dos nossos objetivos é pesquisar e avaliar experimentalmente algoritmos de classificação baseados em técnicas de aprendizado de máquina específicos para séries temporais, selecionamos a biblioteca sktime (LÖNING et al.,

2019) para aplicar em nossos dados. A sktime apresenta uma coletânea de algoritmos propostos na literatura para tratar dados temporais no contexto do aprendizado de máquina. Especificamente em relação à tarefa de classificação, a sktime reúne distintas abordagens, as quais podem ser organizadas segundo o tipo de estratégia adotada (conforme apresentado no capítulo 3), tais como: dicionário, distância, intervalo, shapelets e híbrido. Para este estudo, selecionamos alguns dos algoritmos disponibilizados por esta biblioteca, são eles:

- ***Single Bag of SFA Symbols*** **(BOSS):**

O algoritmo BOSS combina a extração de subestruturas com a tolerância a dados considerados errôneos através de uma representação de redução de ruído para séries temporais. É baseado em um conjunto de classificadores NNs acoplados a uma distância euclidiana, a qual é calculada nos histogramas de frequência obtidos a partir da discretização *Symbolic Fourier Approximation (SFA)*. Em seu processo o BOSS primeiro extrai padrões da série temporal calculando a transformada de Fourier e transformando as séries em histogramas baseados em dicionário (SCHÄFER, 2014).

- ***RandOm Convolutional KErnel Transform*** **(ROCKET):**

O algoritmo ROCKET transforma as séries temporais através de um grande número de kernels convolucionais aleatórios, ou seja, kernels com parâmetros como tamanho, peso e *bias* determinados aleatoriamente. As características transformadas são então utilizadas como dados de entrada para um classificador linear. Cada kernel do algoritmo é aplicado a uma série temporal de entrada, produzindo um mapa de recursos. A operação de convolução envolve um produto escalar deslizante entre um kernel e uma série temporal de entrada (DEMPSTER; PETITJEAN; WEBB, 2019).

- ***Word ExtrAction for time SEries cLassification*** **(WEASEL):**

O algoritmo WEASEL é classificador para séries temporais descrito como escalonável e preciso. Esse método transforma séries temporais em vetores de características através de uma janela deslizante. Primeiro, o WEASEL extrai janelas normalizadas Z de comprimentos variados, em seguida, cada janela é aproximada usando a transformada de Fourier, e usando um teste ANOVA F apenas os valores

reais e imaginários de Fourier que melhor separam as séries temporais são mantidos. Seu diferencial está na técnica específica utilizada para extrair características, o que resulta em conjunto final altamente discriminado, ocupando assim menos memória e reduzindo o tempo de execução, isso sem afetar a precisão do modelo. Essas características ainda permitem que seja possível utilizar regressão logística rápida ao invés de métodos mais elaborados e com tempo de execução maior (SCHÄFER & LESER, 2017).

- ***MUltivariate Symbolic Extension*** **(MUSE):**

Esse método é uma união de métodos WEASEL + MUSE, para o qual é criado um vetor de características multivariado, primeiro usando uma abordagem de janela deslizante aplicada a cada dimensão da série temporal, em seguida, extraindo características discretas por janela e dimensão. O vetor de características é então subsequentemente alimentado através de seleção de atributos, removendo características não discriminativas e posteriormente analisado por um classificador de aprendizado de máquina. O diferencial do WEASEL + MUSE reside em sua maneira específica de extrair e filtrar características multivariadas das séries temporais, codificando informações de contexto em cada atributo. Ainda, o conjunto de recursos extraídos resultante é de pequena dimensão, o que reduz o tempo de execução (SCHÄFER & LESER, 2018).

- ***Time Series Forest Classifier*** **(TSF)**

O TSF emprega uma combinação de ganho de entropia com uma medida de distância, chamada de ganho de entrada. A intuição por trás da ideia é que se duas divisões têm ganho de entropia igual, então a divisão que está mais longe do caso mais próximo deve ser preferida. O que esse algoritmo faz é uma amostragem aleatória de atributos em cada nó da árvore, os quais possuem complexidade computacional linear e podem ser construídos usando técnicas de computação paralela. Um dos fatores mais relevantes é a proposta de capturar as características temporais úteis para a classificação e o uso de descritores simples como média, desvio padrão e inclinação para realizar o processo de classificação, os quais se mostraram computacionalmente eficientes e superaram concorrentes fortes, como classificadores de um vizinho mais próximo com DTW (DENG et al., 2013).

4.5 AVALIAÇÃO EXPERIMENTAL

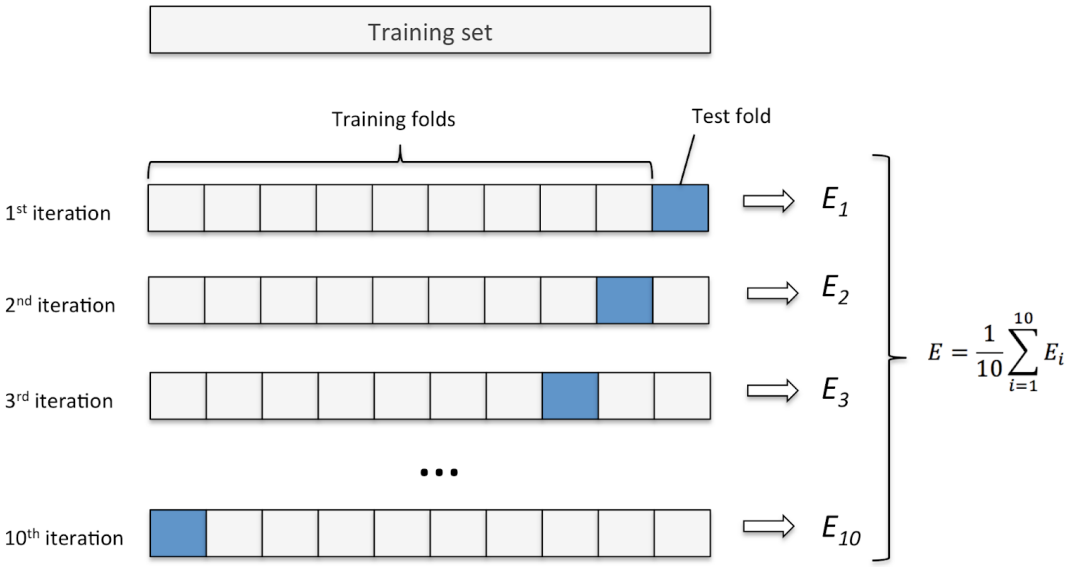
Nesta seção é descrito o procedimento experimental adotado para testar a hipótese deste trabalho. Nesse sentido, é apresentado o método de amostragem utilizado, as medidas de desempenho avaliadas, a organização dos experimentos e a parametrização dos algoritmos.

4.5.1 Método de Amostragem Validação Cruzada

A intenção de desenvolver modelos de aprendizado de máquina para classificação é aplicar estes modelos a dados novos. Uma das preocupações quando treinamos um modelo é que ocorra um sobre-ajuste (*overfitting*) aos dados utilizados como treinamento, ou seja, que ao realizar classificações com dados não presentes no conjunto de treino, o modelo seja influenciado por algum eventual viés existente na amostra de dados analisadas. Quando aplicamos um modelo com sobre-ajuste, as classificações resultantes geralmente tem um desempenho muito abaixo da ideal (MANRY; STURROCK; RAFIQI, 2019).

Neste contexto, um método utilizado na tentativa de evitar o sobre-ajuste é a validação cruzada. A validação cruzada é um método de amostragem que consiste em dividir aleatoriamente um conjunto de exemplos em *k* amostras iguais, onde a *k*-ésima amostra é o conjunto de exemplos de teste e as *k* - 1 amostras restantes formam o conjunto de treinamento. Ainda, cada amostra deve ser mutuamente exclusiva. Para cada combinação das *k* - 1 amostras é construído um modelo e testado com o conjunto de teste *k.* Desse modo, o erro médio do classificador é estimado como a média dos erros dos *k* modelos construídos e é considerado como estimativa do erro verdadeiro (MALETZKE, 2009). Na Figura 16 é apresentada uma representação esquemática do processo de validação cruzada.

Figura 16 - Representação do processo de validação cruzada dividida em 10 blocos.



Fonte: RASCHKA (2015)

Neste trabalho foi utilizada uma validação cruzada dividida em 10 blocos para todos os algoritmos utilizando a estratégia de estratificação.

4.5.2 Medidas de Desempenho

Para a avaliação dos modelos de classificação construídos durante esta pesquisa foram utilizadas as medidas *recall* (revocação ou sensibilidade), precisão, acurácia e medida F1. Por se tratar de um problema de duas classes, as medidas de desempenho são calculadas a partir de parâmetros denominados: Verdadeiros Positivos (VP), Verdadeiros Negativos (VN), Falsos Positivos (FP) e Falsos Negativos (FN). Neste trabalho a classe positiva foi definida como os objetos que não são exoplanetas e a classe negativa como os exoplanetas confirmados. VP corresponde ao número de exemplos da classe positiva classificados corretamente; VN corresponde ao número de exemplos da classe negativa classificados corretamente; FP corresponde ao número de exemplos cuja classe verdadeira é negativa, mas que foram classificados incorretamente como positivos; FN corresponde ao número de exemplos cuja classe verdadeira é positiva, mas que foram classificados incorretamente como negativos. Observe que n = VP + VN + FP + FN (FACELI et al., 2011).

- **Acurácia**

A acurácia é calculada através da soma de todos exemplos de cada classe classificados corretamente, sobre a soma dos valores de todos os elementos.

$$ac = \frac{VP + VN}{n} \tag{4.2}$$

- **Precisão**

A precisão é a razão entre os objetos corretamente rotulados como uma classe sobre a soma de todos os objetos rotulados como pertencentes a essa classe. Ou seja, é a capacidade do modelo de rotular uma classe quando o objeto efetivamente pertencia aquela classe (BUGUEÑO et al., 2018).

$$prec = \frac{VP}{VP + FP} \tag{4.3}$$

- **Recall (revocação ou sensibilidade)**

Representa a razão entre os objetos corretamente rotulados como uma classe sobre a soma de todos os objetos efetivamente daquela classe. Ou seja, é a capacidade do modelo de incluir todos os objetos que efetivamente são pertencentes a uma classe (BUGUEÑO et al., 2018).

$$rev = \frac{VP}{VP + FN} \tag{4.4}$$

- **Medida F1**

Essa métrica é definida como a média harmônica entre as medidas de precisão e recall, onde ambas possuem uma contribuição relativa igual.

$$F_1 = \frac{2 * prec * rev}{prec + rev} \tag{4.5}$$

4.5.3 Organização Experimental

Estruturamos nossos experimentos em duas partes, uma para os algoritmos tradicionais e outra para os algoritmos baseados em séries temporais, sendo que em cada parte aplicamos os dados do Kepler em sua representação local e global proposta em Shallue & Vanderburg (2017). Desse modo, foram realizados dois experimentos para cada um dos algoritmos.

Devido ao grande custo computacional exigido pelos experimentos, utilizamos os servidores do Departamento de Informática da UFPR para realizar as execuções. Os servidores apresentam a seguinte configuração: 2 servidores, cada um com 30 núcleos de processamento Intel(R) Xeon(R) CPU E5-2690 v2@ 3.00Ghz e com 196 GB de memória RAM.

4.5.4 Configuração dos Parâmetros dos algoritmos

Na realização dos experimentos com os algoritmos de classificação realizamos três diferentes combinações de parâmetros para cada um dos algoritmos, a fim de observar o comportamento do modelo para cada combinação. Para realizar esta tarefa utilizamos a função *RandomizedSearchCV()* da biblioteca scikit-learn, a qual realiza combinações aleatórias entre os parâmetros fornecidos e realiza validação cruzada de forma estratificada, separando os exemplos proporcionalmente em cada iteração segundo a distribuição das classes no conjunto de dados.

Nos algoritmos clássicos utilizamos a configuração experimental conforme descrito na Tabela 3.

Tabela 3 - Parâmetros referentes aos algoritmos tradicionais (em negrito as combinações que apresentaram as melhores acurácias - local e global).

Algoritmos	Parâmetros	Descrição
Support Vector Machines	C: [**1**,3,5,**10**,15], kernel: [**'rbf'**,'linear'], tol: [**1e-3**,1e-4], random_state: [1]	As configurações utilizadas foram valores de C entre 1 e 15, kernel linear e kernel de base radial, além de um critério de parada com tolerância de 0,001 e 0,0001.
Decision Tree	n_estimators: [**1**,3,5,8,10], max_features: [**'sqrt'**,**'log2'**], random_state: [1]	As configurações utilizadas foram o número mínimo de amostras necessárias para dividir um nó, definido entre 2 e 10, e o número de características a serem consideradas na busca do ajuste, que foi definido pela raiz quadrada e pelo logaritmo binário.
Random Forests	max_features: [**'sqrt'**, 'log2'], min_samples_split: [2,4,6,8,**10**], random_state: [1]	As configurações utilizadas foram o número de árvores definidas entre 1 e 10, e o número de características a serem consideradas na busca do ajuste definido pela raiz quadrada e pelo logaritmo binário do número de árvores.
Naive Bayes	var_smoothing: [**1e-09**,1e-12,1e-15]	A configuração utilizada com este algoritmo se deu no parâmetro var_smoothing, assumindo os valores de 1e-09, 1e-12 e 1e-15. Este parâmetro amplia ou suaviza a curva bayesiana por onde são filtradas as amostras do conjunto de treino.
Nearest Neighbors	n_neighbors: [1], algorithm: [**'ball_tree'**,**'kd_tree'**,'brute']	Escolhemos utilizar por padrão o valor de K igual a 1. E ainda, fornecemos no parâmetro algorithms as opções ball tree, kd tree e brute, que são alguns dos algoritmos utilizados para calcular os vizinhos mais próximos.
Multilayer Perceptron	random_state: [1], tol: [1e-3,**1e-4**], solver: ['lbfgs', **'sgd'**, **'adam'**]	As configurações utilizadas foram o parâmetro solver com as opções lbfgs, sgd e adam e valores de tolerância de 1e-03 e 1e-04.

Fonte: autora.

Nos algoritmos específicos para séries temporais utilizamos a configuração experimental conforme descrito na Tabela 4.

Tabela 4 - Parâmetros referentes aos algoritmos baseados em séries temporais (em negrito as combinações que apresentaram as melhores acurácias - local e global).

Algoritmos	Parâmetros	Descrição
BOSS	random_state: [1], window_size: [**4**,6,**8**,10,12]	As configurações utilizadas foram random state igual a 1 e tamanho da janela que percorre a curva variando entre 4 e 12.
ROCKET	num_kernels: [10000,**8000**,**5000**]	A configuração utilizada foi o número de kernels nos valores de 10000, 8000 e 5000.
WEASEL	window_inc: [2,**3**,**4**,5,6], random_state: [1]	As configurações utilizadas foram random state igual a 1 e tamanho da janela que percorre a curva variando entre 2 e 6.
MUSE	random_state: [1], window_inc: [4,**5**,6,7,8],	As configurações utilizadas com este algoritmo foram random state igual a 1 e tamanho da janela que percorre a curva variando entre 4 e 8.
TSF	n_estimators: [**200**,**300**,350,400,500]	Escolhemos utilizar o número de estimadores com valores entre 200 e 500.

Fonte: autora.

Os códigos desenvolvidos estão disponíveis para acesso em https://github.com/montangerp/scripts_exoplanetas_tcc.git.

No próximo capítulo serão apresentados os resultados obtidos a partir da avaliação experimental.

5 RESULTADOS E DISCUSSÃO

A motivação para propor uma comparação entre os métodos de classificação tradicionais e os métodos de séries temporais, baseou-se na percepção de que os métodos tradicionais tratavam os valores contidos nas séries temporais como características individuais, sem dependência entre si. Assim, era razoável a hipótese de que a aplicação de métodos que considerassem o fator temporal poderiam prover uma melhor adequação aos dados de curvas de luz. Adicionalmente, pelo melhor de nosso conhecimento da literatura, verificamos que nenhum dos algoritmos de classificação baseados em séries temporais, utilizados neste trabalho, foi proposto ou avaliado no contexto da detecção de exoplanetas.

Independente do algoritmo, a construção de um bom modelo de classificação não pode se basear somente na capacidade de ajustar seus parâmetros aos dados fornecidos, é de fundamental importância realizar o processo de treinamento com dados de qualidade e que representem adequadamente as classes que desejamos identificar. Com isso em mente, dentre todos os dados de missões espaciais disponíveis escolhemos utilizar os dados do Kepler. Essa escolha foi fundamentada na perspectiva de que a missão Kepler foi uma das mais duradouras e portanto, possui uma grande quantidade de registros, os quais estão disponíveis para acesso público por meio de diferentes interfaces. Ainda, existe o fato de que os dados da missão Kepler são utilizados em diversos trabalhos e com diferentes abordagens, o que facilita a comparação de técnicas e resultados. Em relação às técnicas de pré-processamento que aplicamos aos dados, estas foram necessárias para tratar imperfeições nos conjuntos de dados e adequá-los para a aplicação em algoritmos de aprendizado. Além desse aspecto, a concatenação de todos os trimestres de dados disponibilizados pela missão Kepler, baseou-se no fato de que o evento de trânsito pode ser reduzido quando utilizamos as curvas por trimestre individualmente. Logo, quando utilizamos os dados com a junção de todos os trimestres, o evento de interesse pode repetir mais vezes, permitindo que padrões sejam mais facilmente identificados pelos algoritmos de aprendizado. Neste trabalho também utilizamos o método de representação *epoch folding* conforme proposto em Shallue & Vanderburg (2017), a qual apresenta vantagens como a redução de dimensionalidade e o fato de que tem sido utilizada em outros estudos na literatura.

Um outro ponto a ser entendido é a escolha dos algoritmos baseados em séries temporais. Decidimos realizar o estudo experimental com os algoritmos disponibilizados pela biblioteca *sktime*, pois é uma das bibliotecas mais desenvolvidas, com mais aplicações em estudos recentes, e que fornece mais opções de algoritmos, inclusive os considerados como estado da arte na tarefa de classificação de séries temporais.

Dado esse contexto, neste capítulo destacamos os principais resultados obtidos durante o desenvolvimento desta pesquisa. Inicialmente apresentamos os resultados para os algoritmos de classificação tradicionais e, posteriormente, os algoritmos específicos para séries temporais. Ainda, analisamos as hipóteses propostas no início deste trabalho com base nos resultados atingidos e discutimos as questões relacionadas.

5.1 RESULTADOS DOS ALGORITMOS TRADICIONAIS

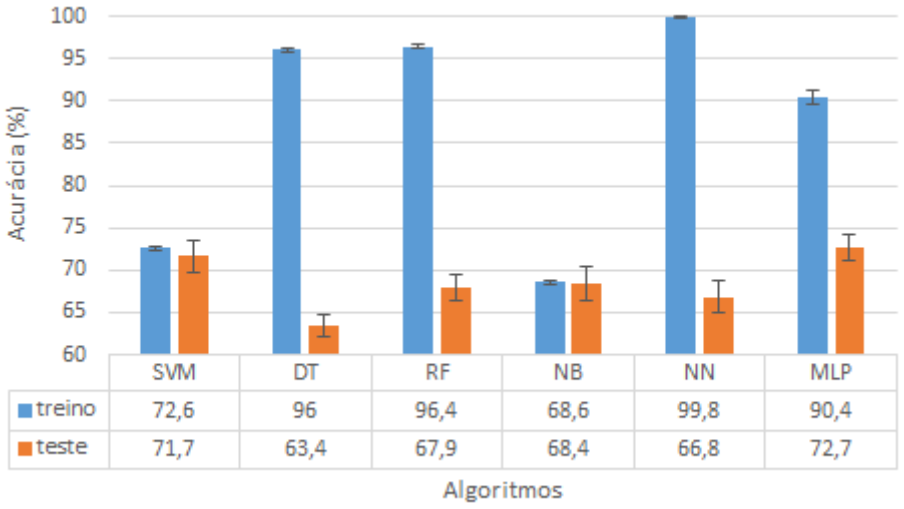
Nesta seção são apresentados os resultados obtidos a partir dos experimentos realizados com os dados do Kepler aplicados aos algoritmos de classificação tradicionais. Utilizamos as curvas de todos os trimestres concatenados na forma de representação local e global. Nas Tabelas 5 e 6 são apresentados os valores de acurácia para cada um dos algoritmos avaliados (linhas) e em cada uma das partições (colunas) os valores da validação cruzada, para a representação local e global, respectivamente. E nas Figuras 17 e 18 apresentamos gráficos comparativos dos valores médios das acurácias resultantes no treino e no teste de cada algoritmo para a representação local e global.

Tabela 5 - Valores de acurácia por algoritmo tradicional e por partição (f*old*) de teste para a representação local.

TRADICIONAIS - ACURÁCIA (%) - TESTE - LOCAL										
Algoritmos	**Fold_0**	**Fold_1**	**Fold_2**	**Fold_3**	**Fold_4**	**Fold_5**	**Fold_6**	**Fold_7**	**Fold_8**	**Fold_9**
SVM	71,1	73,0	74,5	70,8	71,7	69,8	68,9	70,8	75,1	71,7
DT	63,6	63,0	64,7	61,7	61,5	63,8	62,8	63,6	63,4	66,2
RF	65,8	68,7	68,7	66,8	71,3	65,7	67,5	68,5	68,3	67,7
NB	67,9	70,2	71,5	65,8	67,2	64,7	68,1	69,8	70,6	68,5
NN	64,2	67,7	66,0	67,4	70,8	68,9	65,1	67,5	64,7	65,7
MLP	72,1	72,3	76,2	72,1	71,5	71,9	70,4	72,8	74,7	72,6

Fonte: autora.

Figura 17 - Gráfico comparativo: valores médios das acurácias de treino e de teste para a representação local.



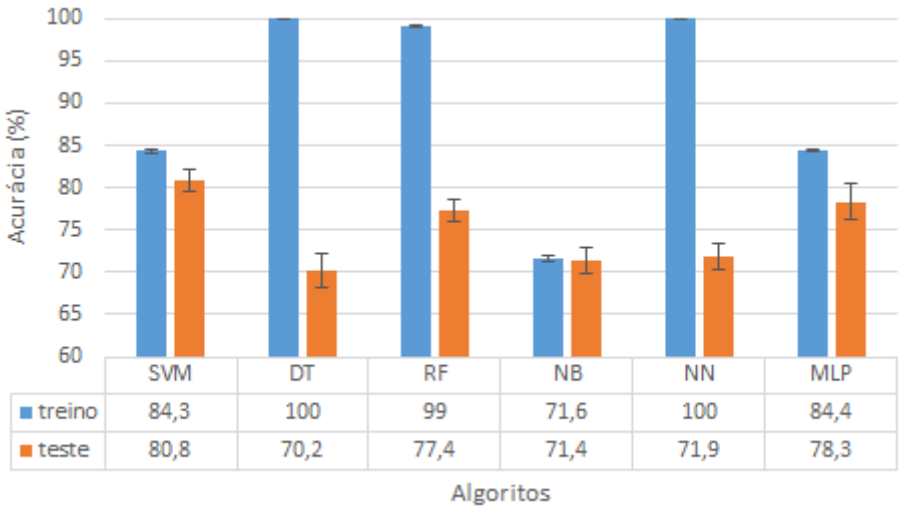
Fonte: autora.

Tabela 6 - Valores de acurácia por algoritmo tradicional e por partição (f*old*) de teste para a representação global.

TRADICIONAIS - ACURÁCIA (%) - TESTE - GLOBAL										
Algoritmos	**Fold_0**	**Fold_1**	**Fold_2**	**Fold_3**	**Fold_4**	**Fold_5**	**Fold_6**	**Fold_7**	**Fold_8**	**Fold_9**
SVM	80,2	81,2	82,8	81,1	78,5	81,7	79,4	82,5	80,2	80,4
DT	71,8	69,9	73,2	69,8	68,5	67,4	69,2	70,6	73,4	68,3
RF	77,8	76,8	79,1	76,6	75,3	79,6	77,2	76,6	78,7	76,8
NB	71,9	72,1	74,2	73,2	69,8	72,1	71,5	70,2	70,2	69,1
NN	71,6	73,8	75,5	71,5	72,3	69,4	71,1	72,6	71,1	70,4
MLP	77,0	79,5	79,6	77,5	73,6	78,9	79,2	81,5	76,2	79,4

Fonte: autora.

Figura 18 - Gráfico comparativo: valores médios das acurácias de treino e de teste para a representação global.



Fonte: autora.

Nas Tabelas 7 e 8 é apresentada uma sumarização das métricas de acurácia, precisão, *recall* e F1 para cada um dos algoritmos, considerando os resultados obtidos sobre as partições de teste da validação cruzada, para a representação local e global, respectivamente.

Tabela 7 - Métricas para os dados de teste locais.

TRADICIONAIS - MÉTRICAS (%) - TESTE - LOCAL								
	acurácia		precisão		recall		F1	
Algoritmos	média	dp	média	dp	média	dp	média	dp
SVM	71,7	1,9	79,7	1,4	69,5	2,8	74,2	2,0
DT	63,4	1,3	68,9	1,5	68,8	2,7	68,8	1,4
RF	67,9	1,6	72,3	1,2	73,3	3,0	72,8	1,6
NB	68,4	2,0	80,3	1,7	61,1	3,0	69,4	2,4
NN	66,8	1,9	68,8	1,7	79,5	2,2	73,7	1,5
MLP	72,7	1,6	78,4	2,3	73,8	4,0	75,9	1,7

Fonte: autora.

Tabela 8 - Métricas para os dados de teste globais.

TRADICIONAIS - MÉTRICAS (%) - TESTE - GLOBAL								
	acurácia		precisão		recall		F1	
Algoritmos	média	dp	média	dp	média	dp	média	dp
SVM	80,8	1,3	85,7	1,6	80,7	1,6	83,1	1,1
DT	70,2	1,9	75,0	2,0	73,8	2,1	74,4	1,6
RF	77,4	1,3	82,4	1,5	78,3	1,8	80,3	1,2
NB	71,4	1,5	81,3	2,0	66,7	1,5	73,2	1,4
NN	71,9	1,6	80,7	1,6	68,5	3,0	74,1	1,8
MLP	78,3	2,1	82,2	3,2	80,5	1,9	81,3	1,6

Fonte: autora.

Com base no resultados apresentados é possível observar que para a representação local a melhor acurácia de teste ocorreu no algoritmo *Multilayer Perceptron* (MLP) com uma média de 72,7% $\pm$1,6%, enquanto que a pior acurácia se deu para o algoritmo *Decision Trees* (DT) com uma média de 63,4% $\pm$1,3%. Considerando a representação global, a melhor acurácia de teste ocorreu no algoritmo *Support Vector Machines* (SVM) com uma média de 80,8% $\pm$1,3%, enquanto que a pior acurácia se deu para o algoritmo *Decision Trees* (DT) com uma média de 70,2% $\pm$1,9%.

5.2 RESULTADOS DOS ALGORITMOS ESPECÍFICOS PARA SÉRIES TEMPORAIS

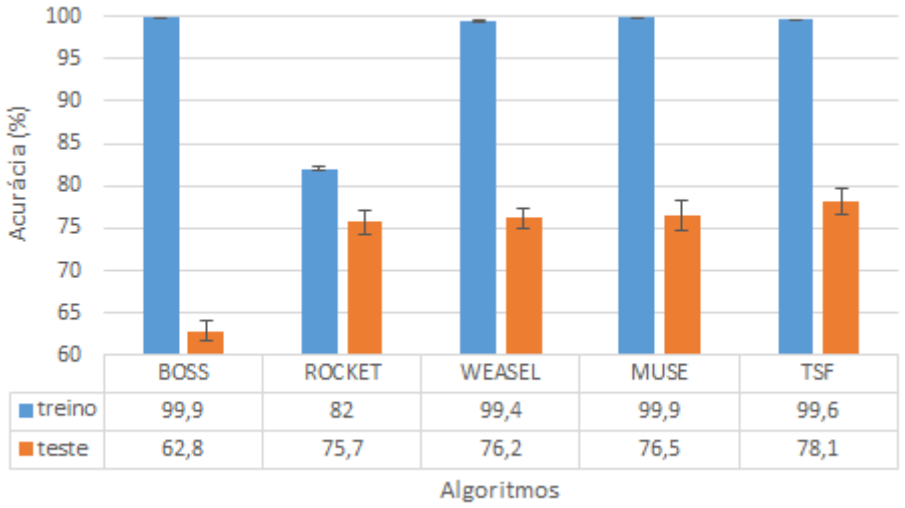
A seguir apresentamos os resultados obtidos com os experimentos realizados com os dados do Kepler aplicados aos algoritmos de classificação específicos para séries temporais da biblioteca *sktime*. Utilizamos as curvas de todos os trimestres concatenados na forma de representação local e global. Nas Tabelas 9 e 10 são apresentados os valores de acurácia para cada um dos algoritmos avaliados (linhas) e em cada uma das partições (colunas) os valores da validação cruzada, para a representação local e global, respectivamente. E nas Figuras 19 e 20 apresentamos gráficos comparativos dos valores médios das acurácias resultantes no treino e no teste de cada algoritmo para a representação local e global.

Tabela 9 - Valores de acurácia por algoritmo da *sktime* e por partição (f*old*) de teste para a representação local.

SKTIME - ACURÁCIA (%) - TESTE - LOCAL										
Algoritmos	**Fold_0**	**Fold_1**	**Fold_2**	**Fold_3**	**Fold_4**	**Fold_5**	**Fold_6**	**Fold_7**	**Fold_8**	**Fold_9**
BOSS	62,6	62,6	64,2	61,7	60,9	60,9	63,2	64,2	64,5	63,4
ROCKET	77,9	76,0	78,3	75,1	74,7	74,2	74,2	74,7	76,8	75,3
WEASEL	75,7	75,8	77,5	76,0	76,4	76,4	74,2	75,3	78,7	75,7
MUSE	77,2	74,2	77,4	77,0	77,4	74,7	72,8	77,4	78,7	77,9
TSF	77,4	77,2	80,2	80,0	77,4	78,7	75,1	77,9	79,8	77,7

Fonte: autora.

Figura 19 - Gráfico comparativo: valores médios das acurácias de treino e de teste para a representação local.



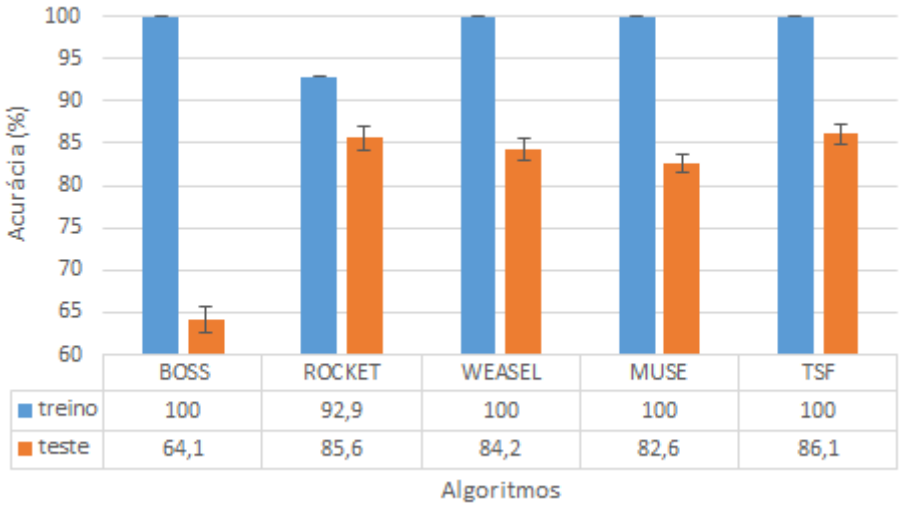
Fonte: autora.

Tabela 10 - Valores de acurácia por algoritmo da *sktime* e por partição (f*old*) de teste para a representação global.

SKTIME - ACURÁCIA (%) - TESTE - GLOBAL										
Algoritmos	**Fold_0**	**Fold_1**	**Fold_2**	**Fold_3**	**Fold_4**	**Fold_5**	**Fold_6**	**Fold_7**	**Fold_8**	**Fold_9**
BOSS	65,5	64,0	64,3	65,7	65,8	64,0	64,9	60,2	62,8	63,8
ROCKET	86,8	86,3	87,0	86,6	83,6	86,4	82,8	86,2	84,7	85,5
WEASEL	83,6	84,4	85,8	85,3	84,0	84,2	82,5	86,6	82,5	83,0
MUSE	81,2	82,7	82,8	84,3	81,9	83,0	82,8	82,5	84,0	81,3
TSF	85,3	86,1	86,6	88,1	84,0	86,2	84,9	88,1	86,4	85,5

Fonte: autora.

Figura 20 - Gráfico comparativo: valores médios das acurácias de treino e de teste para a representação global.



Fonte: autora.

Nas Tabelas 11 e 12 é apresentada uma sumarização das métricas de acurácia, precisão, *recall* e F1 para cada um dos algoritmos, considerando os resultados obtidos sobre as partições de teste da validação cruzada, para a representação local e global, respectivamente.

Tabela 11 - Métricas para os dados de teste locais.

SKTIME - MÉTRICAS (%) - TESTE - LOCAL								
	acurácia		precisão		recall		F1	
Algoritmos	**média**	**dp**	**média**	**dp**	**média**	**dp**	**média**	**dp**
BOSS	62,8	1,2	66,5	1,0	73,5	2,1	69,8	1,2
ROCKET	75,7	1,4	80,2	1,1	77,8	3,2	78,9	1,6
WEASEL	76,2	1,2	80,2	0,8	78,8	2,9	79,5	1,4
MUSE	76,5	1,8	80,0	0,7	79,8	3,5	79,8	1,9
TSF	78,1	1,5	82,4	1,1	79,7	2,6	81,0	1,5

Fonte: autora.

Tabela 12 - Métricas para os dados de teste globais.

SKTIME - MÉTRICAS (%) - TESTE - GLOBAL								
	acurácia		precisão		recall		F1	
Algoritmos	**média**	**dp**	**média**	**dp**	**média**	**dp**	**média**	**dp**
BOSS	64,1	1,6	71,5	1,7	64,5	1,7	67,8	1,4
ROCKET	85,6	1,4	87,3	1,4	88,3	1,6	87,8	1,2
WEASEL	84,2	1,3	87,1	1,1	85,7	2,2	86,4	1,2
MUSE	82,6	1,0	85,9	1,5	84,2	1,3	85,0	0,8
TSF	86,1	1,2	89,0	1,0	87,0	1,8	88,0	1,1

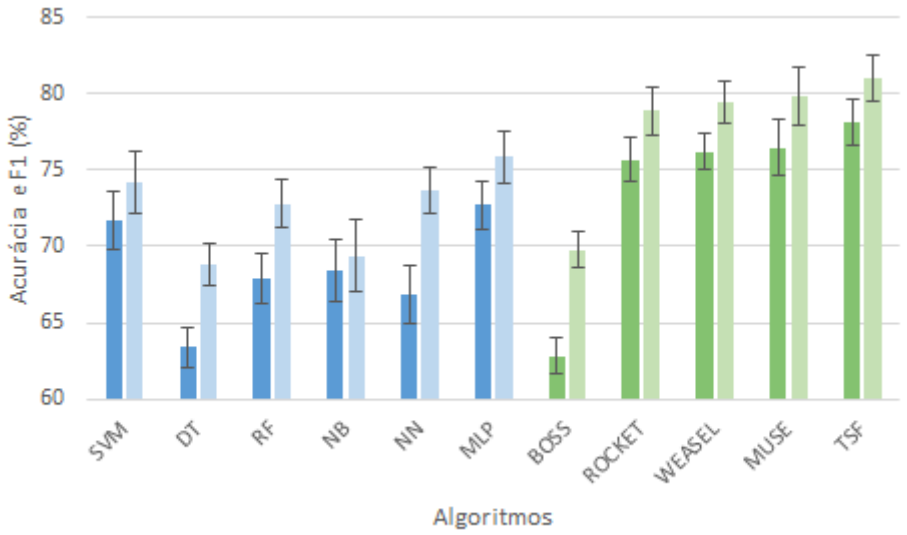
Fonte: autora.

Com base nos resultados apresentados, é possível observar que, para a representação local, a melhor acurácia de teste ocorreu no algoritmo *Time Series Forest* (TSF) com uma média de 78,1% $\pm$1,5%, enquanto que a pior acurácia se deu para o algoritmo BOSS com uma média de 62,8% $\pm$1,2%. Considerando a representação global, a melhor acurácia de teste ocorreu no algoritmo *Time Series Forest* (TSF) com uma média de 86,1% $\pm$1,2%, enquanto que a pior acurácia se deu para o algoritmo BOSS com uma média de 64,1% $\pm$1,6%.

5.3 DISCUSSÃO DOS RESULTADOS

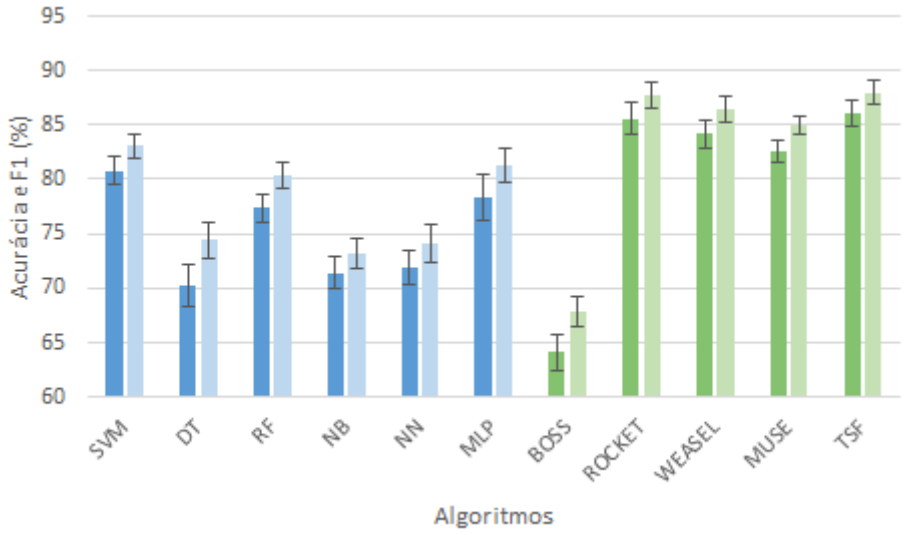
Com base na hipótese de que na detecção de exoplanetas por meio de curvas de luz a utilização de algoritmos de aprendizado de máquina baseados em séries temporais é mais adequada em relação aos algoritmos de aprendizado tradicionais, apresentamos gráficos comparativos entre os algoritmos tradicionais e baseados em séries temporais considerando as abordagens de representação local e global. As barras em azul e verde escuro representam valores médios de acurácia e as barras em azul e verde claro representam valores médios da medida F1.

Figura 21 - Gráfico comparativo: algoritmos tradicionais e baseados em séries temporais para os dados locais.



Fonte: autora.

Figura 22 - Gráfico comparativo: algoritmos tradicionais e baseados em séries temporais para os dados globais.



Fonte: autora.

O que podemos observar em relação à representação local são acurácias sempre maiores que 75% para os algoritmos baseados em séries temporais, com exceção do algoritmo BOSS que apresentou o pior resultado, inclusive quando comparado com os algoritmos tradicionais. O que, possivelmente, pode ter ocorrido devido à distribuição dos parâmetros ou um sobre-ajuste dos dados, o que pode ser

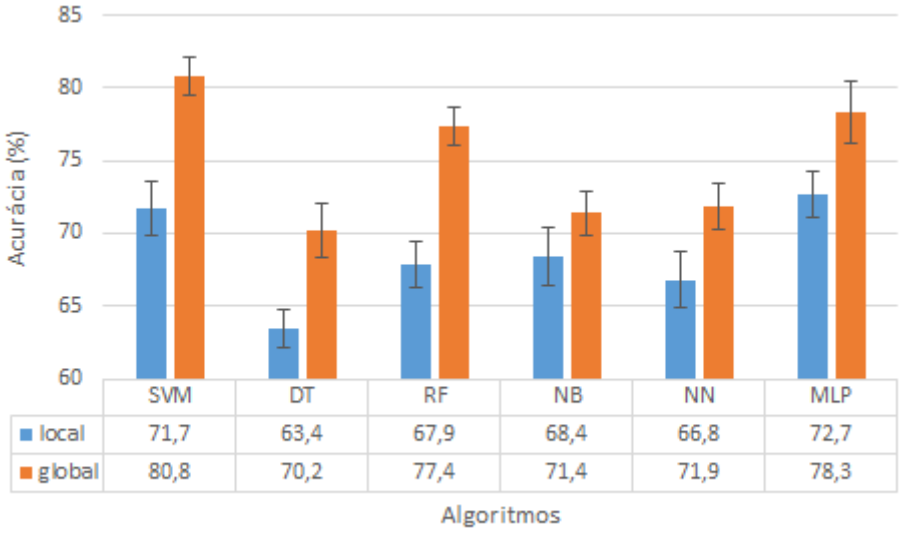
reforçado pelos dados de treino apresentados. Nos algoritmos tradicionais o *Multilayer Perceptron* (MLP) e o *Support Vector Machines* (SVM) apresentaram resultados maiores que 70% e menores que 75%, enquanto os demais ficaram abaixo de 70%.

Em relação aos experimentos com a representação global, observamos os algoritmos baseados em séries temporais com acurácias maiores que 80% e mais uma vez o algoritmo BOSS aparecendo como exceção. Nos algoritmos tradicionais o *Multilayer Perceptron* (MLP) se destacou, obtendo uma média superior a 80%, já os demais permaneceram com resultados abaixo de 75%.

Logo, mediante esta análise, quando comparamos as melhores acurácias de cada classe de algoritmos, verificamos uma melhora de aproximadamente 6% ao utilizar os classificadores baseados em séries temporais.

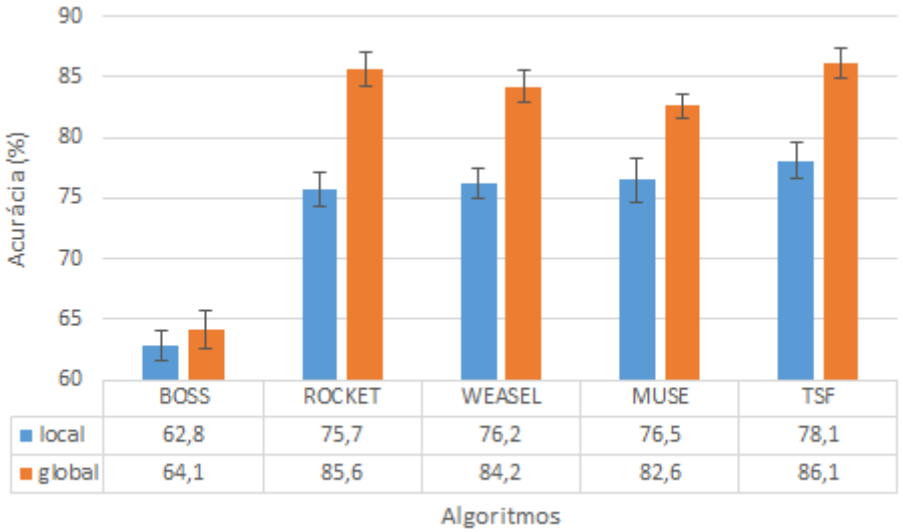
Quando comparamos os resultados de teste nas representações local e global, conforme Figuras 23 e 24, é possível verificar valores de acurácia superiores para a representação global. Quando realizamos o pré-processamento dos dados a média de valores substituídos na interpolação para os dados globais foi de 1,12% enquanto que para os dados locais a média foi de 31,35%, essa pode ser a causa dos modelos conseguirem se ajustar melhor aos dados globais.

Figura 23 - Tradicionais: Acurácia para dados de teste local e global.



Fonte: autora.

Figura 24 - Sktime: Acurácia para dados de teste local e global.



Fonte: autora.

Com o intuito de validar os resultados obtidos e testar a hipótese deste trabalho, utilizamos o teste estatístico ANOVA Medidas Repetidas com nível de significância de 0,05. Nesta validação, a hipótese nula é que o desempenho dos algoritmos de classificação, em termos de acurácia, são iguais. Adicionalmente, verificamos se existe efeito sobre o desempenho dos algoritmos em relação ao tipo de representação utilizada (local e global). Essas análises foram realizadas por meio do programa computacional JASP (https://jasp-stats.org/)

Considerando o desempenho dos algoritmos, em termos de acurácia, pela aplicação do teste estatístico, a hipótese nula foi recusada ($F(10,180) = 386,97$ e p-valor $< 0,001$), de modo que é possível afirmar que existe diferença estatisticamente significativa entre os algoritmos de classificação analisados. Para identificar quais as diferenças encontradas entre os algoritmos foi aplicado o pós-teste de Bonferroni. Na Tabela 13, são apresentadas as diferenças identificadas.

Tabela 13 - Diferenças estatísticas entre os algoritmos
(* representa que houve diferença significativa).

ANOVA - ACURÁCIA											
	SVM	DT	RF	NB	NN	MLP	BOSS	ROCKET	WEASEL	MUSE	TSF
SVM	-	*	*	*	*		*	*	*	*	*
DT	-	-	*	*	*	*	*	*	*	*	*
RF	-	-	-	*	*	*	*	*	*	*	*
NB	-	-	-	-		*	*	*	*	*	*
NN	-	-	-	-	-	*	*	*	*	*	*
MLP	-	-	-	-	-	-	*	*	*	*	*
BOSS	-	-	-	-	-	-	-	*	*	*	*
ROCKET	-	-	-	-	-	-	-	-			
WEASEL	-	-	-	-	-	-	-	-	-		*
MUSE	-	-	-	-	-	-	-	-	-	-	*
TSF	-	-	-	-	-	-	-	-	-	-	-

Fonte: autora.

Mediante os resultados e as diferenças estatísticas apresentadas, observamos que existem diferenças significativas entre todos os algoritmos baseados em séries temporais e os algoritmos tradicionais. Assim, quando verificamos o desempenho de cada classificador, em termos de acurácia, podemos inferir que os algoritmos para séries temporais são mais adequados para a identificação de exoplanetas por meio de curvas de luz. Com exceção do algoritmo BOSS que, como comentado anteriormente, obteve resultados inferiores. Também, dentre os algoritmos baseados em séries temporais, o desempenho superior do TSF, em relação aos algoritmos MUSE, WEASEL, e BOSS, foi estatisticamente significativo. Em relação ao Rocket, não foi encontrada diferença estatisticamente significativa quando comparada ao TSF, no entanto, em termos absolutos, o TSF apresentou maiores valores de acurácia.

Em relação ao tipo de representação utilizada, verificamos que existe diferença estatisticamente significativa entre os métodos local e global ($F(1,18) = 250,20$ e p-valor $< 0,001$). Desse modo, com base nos resultados reportados a partir da avaliação experimental, é possível concluir que a utilização da representação global é mais adequada do que a representação local.

6 CONCLUSÕES E TRABALHOS FUTUROS

Nos últimos anos, projetos espaciais, como o Kepler, possibilitaram o rápido armazenamento de uma grande quantidade de dados temporais na forma de curvas de luz. Nesse contexto, tornou-se cada vez mais comum a criação de processos automáticos para a análise desses dados, especialmente pela aplicação de técnicas de aprendizado de máquina. No entanto, para compreender adequadamente os eventos contidos em informações temporais, é fundamental a utilização de métodos específicos para o tratamento desse tipo de dado. Portanto, neste trabalho, com o objetivo de identificar exoplanetas de modo automático a partir de curvas de luz, propusemos o estudo de algoritmos de aprendizado de máquina específicos para o tratamento de séries temporais. Para atingir esse objetivo realizamos a aquisição e o pré-processamento de curvas de luz provenientes da missão Kepler, incluindo tarefas como remoção de valores ausentes, remoção de *outliers*, normalização e transformação. Ao final deste processo, constituímos um conjunto de dados composto por 5302 curvas de luz, das quais 2195 são rotuladas como exoplanetas e 3107 são representantes de outros objetos celestes. Também, conduzimos uma avaliação experimental com 11 algoritmos de classificação, dos quais 6 algoritmos, considerados clássicos na literatura de aprendizado de máquina, definimos como o benchmark do estudo e outros 5 algoritmos específicos para o tratamento de séries temporais.

Ao analisarmos os resultados obtidos a partir dos experimentos realizados, verificamos que, em geral, os algoritmos baseados em séries temporais foram superiores ao *benchmark* estabelecido, em termos das métricas estabelecidas. Assim, confirmamos a hipótese de que a utilização de algoritmos de aprendizado de máquina baseados em séries temporais para a detecção de exoplanetas por meio de curvas de luz, é mais adequada em relação aos algoritmos de aprendizado tradicionais, devido à consideração de que as observações estão ordenadas no tempo, não permitindo assim que estas sejam tratadas como características singulares e independentes de ordem.

Mediante os diversos processos abordados neste estudo, as perspectivas para trabalhos futuros são variadas, dentre as quais podemos citar:

- Estudo de outros conjuntos de dados como o do TESS e do K2;
- Estudo da classificação de curvas de luz com os dados brutos;

- Estudo com outras técnicas de representação das séries, como *recurrence plots*;
- Estudo aprofundado dos parâmetros dos algoritmos da *sktime*;
- Utilização das imagens do Kepler ao invés das curvas de luz como entrada para os algoritmos;
- Estudo dos parâmetros do método de *epoch-folding* proposto por Shallue & Vanderburg (2017).
- Avaliação de técnicas de Deep Learning para classificação de exoplanetas.

REFERÊNCIAS

AGRAWAL, R.; FALOUTSOS, C.; SWAMI, A. N. **Efficient similarity search in sequence databases**. In: Proceedings of the 4th International Conference on Foundations of Data Organization and Algorithms. London, UK, UK: Springer-Verlag, 1993.

AHA, D. W.; KIBLER, D.; ALBERT, M. K. Instance-based learning algorithms. **Machine Learning**, v. 6, n. 1, 1991.

ALPAYDIN, E. **Introduction to Machine Learning**, 2004.

ARMSTRONG, D. J.; GAMPER, J.; DAMOULAS, T. **Exoplanet validation with machine learning: 50 new validated kepler planetsarXiv**, 2020.

ARMSTRONG, D. J.; POLLACCO, D.; SANTERNE, A. Transit shapes and self-organizing maps as a tool for ranking planetary candidates: Application to Kepler and K2. **Monthly Notices of the Royal Astronomical Society**, v. 465, n. 3, 2017.

ASTROM, K. J. **Information science: On the choice of sampling rates in parametric identification of time series**. Inf. Sci., Elsevier Science Inc., New York, NY, USA, v. 1, n. 3, p. 273–278, jul. 1969.

BABU, G. J.; MAHABAL, A. Skysurveys, Light Curves and Statistical Challenges. **International Statistical Review**, v. 84, n. 3, 2016.

BAGNALL, A. et al. Time-Series Classification with COTE: The Collective of Transformation-Based Ensembles. **IEEE Transactions on Knowledge and Data Engineering**, v. 27, n. 9, 2015.

BAGNALL, A. et al. The Great Time Series Classification Bake Off: **An Experimental Evaluation of Recently Proposed Algorithms. Extended Version**, 2016.

BAGNALL, A. et al. The great time series classification bake off: a review and experimental evaluation of recent algorithmic advances. **Data Mining and Knowledge Discovery**, v. 31, n. 3, 2017.

BAGNALL, A. et al. **On the usage and performance of the hierarchical vote collective of transformation-based ensembles version 1.0 (HIVE-COTE v1.0)**. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). **Anais**...2020

BAO, D. **A generalized model for financial time series representation and prediction.** Applied Intelligence, Kluwer Academic Publishers, Hingham, MA, USA, v. 29, n. 1, p. 1–11, ago. 2008.

BARBOSA, J. M. et al. **Métodos de Classificação por Árvores de Decisão. Disciplina de Projeto e Análise de Algoritmos**, [s.l: s.n.].

BAYDOGAN, M. G.; RUNGER, G. Time series representation and similarity based on local autopatterns. **Data Mining and Knowledge Discovery**, v. 30, n. 2, 2016.

BAYDOGAN, M. G.; RUNGER, G.; TUV, E. A bag-of-features framework to classify time series. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. 35, n. 11, 2013.

BAILEY, S.; ARAGON, C.; ROMANO, R. et al. ApJ, 665, 1246, 2007.

BLOMME, J. **Variable star data mining techniques for time-resolved databases**. Faculteit Wetenschappen, 2012.

BOSTROM, A.; BAGNALL, A. **Binary shapelet transform for multiclass time series classification**. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). **Anais**...2017

BUGUENO, M.; MENA, F.; ARAYA, M. **Refining exoplanet detection using supervised learning and feature engineering**. Proceedings - 2018 44th Latin American Computing Conference, CLEI 2018. **Anais**...2018

CABELLO, N. et al. **Fast and accurate time series classification through supervised interval search**. Proceedings - IEEE International Conference on Data Mining, ICDM. **Anais**...2020

CASTRILLÓN, J. P. B. **Análise de Curvas de Luz do Corot usando diferentes processos comparativos: estimando períodos de rotação estelar**. Universidade Federal do Rio Grande do Norte - UFRN, 2010.

CASTRO, N. C. **Time series motif discovery**. Tese (Tese de doutorado) — Universidade do Minho, Minho, Portugal, 2012.

CHAN, K.-P.; FU, A.-C. **Efficient time series matching by wavelets.** In: Data Engineering, 1999. Proceedings., 15th International Conference on. [S.l.: s.n.], 1999.

CHANG, C. C.; LIN, C. J. LIBSVM: A Library for support vector machines. **ACM Transactions on Intelligent Systems and Technology**, v. 2, n. 3, 2011.

CHARBONNEAU, D. et al. **Detection of planetary transits a cross a sun-like star**, 2000.

CHARNOCK, T.; MOSS, A. ApJL, 837, L28, 2017.

CHAUVIN, G. et al. A giant planet candidate near a young brown dwarf Direct VLT/NACO observations using IR wavefront sensing. **Astronomy and Astrophysics**, v. 425, n. 2, 2004.

CORTES, C.; VAPNIK, V. Support-Vector Networks. **Machine Learning**, v. 20, n. 3, 1995.

COUGHLIN J. L. et al. **The Astrophysical Journal Supplement Series**, 224, 12, 2016b.

DATILLO, A. et al. **The Astronomical Journal**, 157, 169, 2019.

DEMPSTER, A.; PETITJEAN, F.; WEBB, G. I. ROCKET: exceptionally fast and accurate time series classification using random convolutional kernels. **Data Mining and Knowledge Discovery**, v. 34, n. 5, 2020.

DENG, H. et al. A time series forest for classification and feature extraction. **Information Sciences**, v. 239, 2013.

DHARIYAL B. et al. **An Examination of the State-of-the-Art for Multivariate Time Series Classification**, 2020.

ESLING, P.; AGON, C. **Time-series data mining**. ACM Comput. Surv., ACM, New York, NY, USA, v. 45, n. 1, p. 12:1–12:34, dez. 2012.

EXOPLANET ARCHIVE. **Holdings**. [s.l: s.n.]. Disponível em: https://exoplanetarchive.ipac.caltech.edu/docs/holdings.html. Acesso em: 9 maio 2021.

EXOPLANETS NASA. **Discovery Fast Facts**. 2020. Disponível em: https://exoplanets.nasa.gov/discovery/missions/#otp_fast_facts. Acesso em: 9 maio 2021.

FACELI, K. et al. **Inteligência artificial : uma abordagem de aprendizado de máquina**. [s.l: s.n.].

FALOUTSOS, C.; RANGANATHAN, M.; MANOLOPOULOS, Y. **Fast subsequence matching in time-series databases**. In: Proceedings of the 1994 ACM SIGMOD International Conference on Management of Data. New York, NY, USA: ACM, 1994.

FAYYAD, U.; PIATETSKY-SHAPIRO, G.; SMYTH, P. Knowledge Discovery and Data Mining: Towards a Unifying Framework. **Int Conf on Knowledge Discovery and Data Mining**, 1996.

FAYYAD, U.; PIATETSKY-SHAPIRO, G.; SMYTH, P. aimag-kdd-overview-1996-Fayyad. **Association For The Advancement Of Artificial Intelligence**, 1997.

FERRERO, C. A. **Algoritmo kNN para previsão de dados temporais: funções de previsão e critérios de seleção de vizinhos próximos aplicados a variáveis ambientais em limnologia.** Dissertação (Mestrado) — Universidade de São Paulo - USP, 2009.

FU, T. chung. **A review on time series data mining.** Eng. Appl. of AI, v. 24, n. 1, p. 164–181, 2011.

GEURTS, P. **Pattern extraction for time series classification**. In: Proceedings of the 5th European Conference on Principles of Data Mining and Knowledge Discovery. London, UK: Springer-Verlag, 2001.

GINSBURG, A. et al. **Astroquery: An astronomical web-querying package in python**, 2019.

GRABOCKA, J. et al. **Learning time-series shapelets**. Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. **Anais**...2014

GREGERSEN, E. **"CoRoT"**. Encyclopedia Britannica. 2019. Disponível em: https://www.britannica.com/topic/CoRoT. Acesso em: 9 maio 2021.

GRONDIN, M. H. et al. The Vela-X pulsar wind nebula revisited with four years of Fermi large area telescope observations. **Astrophysical Journal**, v. 774, n. 2, 2013.

GRZIWA, S.; PATZOLD M. **Wavelet-based filter methods to detect small transiting planets in stellar light curves**, 2016.

GULLO, F. et al. **A time series representation model for accurate and fast similarity detection**. Pattern Recogn., Elsevier Science Inc., New York, NY, USA, v. 42, n. 11, p. 2998–3014, nov. 2009.

HAN, J.; KAMBER, M. **Data Mining: Concepts and Techniques**. 2006.

HAYKIN, S. **Neural Networks**. v2. Symon & Schuster, 1999.

HENRY, G. W. et al. A Transiting “51 Peg–like” Planet. **The Astrophysical Journal**, v. 529, n. 1, 2000.

HERRERA, L. J. et al. **Recursive prediction for long term time series forecasting using advanced models.** Neurocomput., Elsevier Science Publishers B. V., Amsterdam, The Netherlands, The Netherlands, v. 70, n. 16-18, p. 2870–2880, out. 2007.

HETLAND, M. L. **A Survey of Recent Methods for Efficient Retrieval of Similar Time Sequences**. In: LAST, M.; KANDEL, A.; BUNKE, H. (Ed.). Data Mining in Time Series Databases. [S.l.]: World Scientific, 2004.

HILLS, J. et al. Classification of time series by shapelet transformation. **Data Mining and Knowledge Discovery**, v. 28, n. 4, 2014.

HINNERS, T. A.; TAT, K.; THORP, R. **Machine learning techniques for stellar light curve classificationarXiv**, 2017.

IMAGINE NASA. **Timing Analysis**. 2013. Disponível em: https://imagine.gsfc.nasa.gov/science/toolbox/timing2.html. Acesso em: 9 maio 2021.

INDURKHYA, N.; WEISS, S. M. Estimating performance gains for voted decision trees. **Intelligent Data Analysis**, v. 2, n. 4, 1998.

ISMAIL FAWAZ, H. et al. Deep learning for time series classification: a review. **Data Mining and Knowledge Discovery**, v. 33, n. 4, 2019.

JARA-MALDONADO, M. et al. **Transiting Exoplanet Discovery Using Machine Learning Techniques: A SurveyEarth Science Informatics**, 2020.

KALPAKIS, K.; GADA, D.; PUTTAGUNTA, V. **Distance measures for effective clustering of arima time-series**. In: Proceedings of the 2001 IEEE International Conference on Data Mining. Washington, DC, USA: IEEE Computer Society, 2001.

KARPENKA, N. V.; FEROZ, F.; HOBSON, M. P. MNRAS, 429, 1278, 2013.

KEOGH, E.; PAZZANI, M. **An enhanced representation of time series which allows fast and accurate classification, clustering and relevance feedback**. In: AGRAWAL, R.; STOLORZ, P.; PIATETSKY-SHAPIRO, G. (Ed.). Fourth International Conference on Knowledge Discovery and Data Mining (KDD'98). New York City, NY: ACM Press, 1998.

KEOGH, E. et al. **Dimensionality reduction for fast similarity search in large time series databases**. Knowledge and Information Systems, Springer-Verlag London Limited, v. 3, n. 3, p. 263–286, 2001.
KEOGH, E. et al. **Locally adaptive dimensionality reduction for indexing large time series databases**. SIGMOD Rec., ACM, New York, NY, USA, v. 30, n. 2, p. 151–162, maio 2001.

KEOGH, E.; KASETTY, S. **On the need for time series data mining benchmarks: A survey and empirical demonstration**. Data Min. Knowl. Discov., Kluwer Academic Publishers, Hingham, MA, USA, v. 7, n. 4, p. 349–371, out. 2003.

KEOGH, E. J.; LIN, J. **Clustering of time-series subsequences is meaningless: implications for previous and future research**. Knowl. Inf. Syst., v. 8, n. 2, p. 154–177, 2005.

KEOGH, E. et al. **Finding the most unusual time series subsequence: Algorithms and applications**. Knowl. Inf. Syst., Springer-Verlag New York, Inc., New York, NY, USA, v. 11, n. 1, p. 1–27, dez. 2006.

KORN, F.; JAGADISH, H. V.; FALOUTSOS, C. **Efficiently supporting ad hoc queries in large datasets of time sequences**. In: Proceedings of the 1997 ACM SIGMOD International Conference on Management of Data. New York, NY, USA: ACM, 1997.

KOSKELA, T. **Neural Network Methods in Analysing and Modelling Time Varying Processes**. Tese (Doutorado) — Helsinki University of Technology, Espoo, Finland, Dezembro 2003.

KOVÁCS, G. ZUCKER, S. MAZEH T. A&A, 391, 369, 2002.

LAXMAN, S.; SASTRY, P. S. A survey of temporal data mining. **Sadhana - Academy Proceedings in Engineering Sciences**, v. 31, n. 2, 2006.

LE NGUYEN, T. et al. Interpretable time series classification using linear models and multi-resolution multi-domain symbolic representations. **Data Mining and Knowledge Discovery**, v. 33, n. 4, 2019.

LIGHTKURVE COLLABORATION, L. et al. Lightkurve: Kepler and TESS time series analysis in Python. **ascl**, 2018.

LIGHTKURVE. **Machine learning style preprocessing with Lightkurve**. [s.l: s.n.]. Disponível em: https://docs.lightkurve.org/tutorials/3-science-examples/exoplanets-machine-learning-preprocessing.html. Acesso em: 9 maio 2021.

LIGHTKURVE. **Using Light Curve Files with Lightkurve.** [s.l: s.n.]. Disponível em: https://docs.lightkurve.org/tutorials/1-getting-started/using-light-curve-file-products.html. Acesso em: 9 maio 2021.

LIN, J. et al. **A symbolic representation of time series, with implications for streaming algorithms.** In: Proceedings of the 8th ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery. New York, NY, USA: ACM, 2003.

LIN, J. et al. **Experiencing sax: a novel symbolic representation of time series**. Data Min. Knowl. Discov., v. 15, n. 2, p. 107–144, 2007.

LINES, J.; TAYLOR, S.; BAGNALL, A. **HIVE-COTE: The hierarchical vote collective of transformation-based ensembles for time series classification**. Proceedings - IEEE International Conference on Data Mining, ICDM. **Anais**...2017

LINES, J.; TAYLOR, S.; BAGNALL, A. Time series classification with HIVE-COTE: The hierarchical vote collective of transformation-based ensembles. **ACM Transactions on Knowledge Discovery from Data**, v. 12, n. 5, 2018.

LÖNING, M. et al. **Sktime: A unified interface for machine learning with time series**, 2019.

LOPES, C. E. F. **Estudo Sistemático de Estrelas Variáveis na era dos Grandes Surveys.** Dissertação (Mestrado) – Universidade Federal do Rio Grande do Norte - UFRN, 2013.

LUCAS, B. et al. Proximity Forest: an effective and scalable distance-based classifier for time series. **Data Mining and Knowledge Discovery**, v. 33, n. 3, 2019.

MALETZKE, A. G. **Uma metodologia para a extração de conhecimento em séries temporais por meio da identificação de motifs e da extração de características**. Dissertação (Mestrado) – Universidade de São Paulo - USP, 2009.

MALETZKE, A. G. et al. Time series classification with motifs and characteristics. **Studies in Computational Intelligence**, v. 537, 2014.

MALIK, A.; MOSTER, B.; OBERMEIER, C. **Exoplanet detection using machine learningarXiv**, 2020.

MANDEL, K.; AGOL, E. Analytic Light Curves for Planetary Transit Searches. **The Astrophysical Journal**, v. 580, n. 2, 2002.

MCCAULIFF, S. D. et al. **The Astrophysical Journal**, 806, 6, 2015.

MEGALOOIKONOMOU, V.; LI, G.; WANG, Q. **A dimensionality reduction technique for efficient similarity analysis of time series databases**. In:

Proceedings of the Thirteenth ACM International Conference on Information and Knowledge Management. New York, NY, USA: ACM, 2004.

MICHALSKI, R. S. et al. **Machine Learning and Data Mining: Methods and Applications**. 1998.

MIDDLEHURST, M. et al. **The Temporal Dictionary Ensemble (TDE) Classifier for Time Series Classification**. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). **Anais**...2021

MISLIS, D. et al. **Monthly Notices of the Royal Astronomical Society**, 455, 626, 2016.

MITCHELL, T. M. **Machine Learning**. Boston, USA: McGraw-Hill, 1997.

MONTANGER, P. O.; ZALEWSKI, W. **Classificação automática de objetos astronômicos por meio da análise de séries temporais**. Revista Brasileira de Iniciação Científica, 2019.

MONTANGER, P. O.; ZALEWSKI, W. **Programa computacional para a identificação automática de exoplanetas**. Revista Brasileira de Iniciação Científica, 2020.

MÖRCHEN, F.; ULTSCH, A. **Optimizing time series discretization for knowledge discovery**. In: KDD. [S.l.: s.n.], 2005.

MORCHEN, F. **Time series knowledge mining**. Tese (Tese de doutorado) — Department of Mathematics and Computer Science–Philipps-University, Marburg, Hesse, Germany, 2006.

MORETTIN, P. A.; TOLOI, C. M. **Análise de Séries Temporais**. 2. ed. São Paulo, Brasil: Edgard Blecher, 2006.

NANOPOULOS, A.; ALCOCK, R.; MANOLOPOULOS, Y. **Information processing and technology**. In: MASTORAKIS, N.; NIKOLOPOULOS, S. D. (Ed.). Commack, NY, USA: Nova Science Publishers, Inc., 2001.

PANUCCIO, A.; BICEGO, M.; MURINO, V. **A hidden markov model-based approach to sequential data clustering**. In: Proceedings of the Joint IAPR International Workshop on Structural, Syntactic, and Statistical Pattern Recognition. London, UK, UK: Springer-Verlag, 2002.

PARDO, T. A. S.; NUNES, M. G. V. **Aprendizado Bayesiano Aplicado ao Processamento de Línguas Naturais**, 2002.

PEDREGOSA, F. et al. Scikit-learn: Machine learning in Python. **Journal of Machine Learning Research**, v. 12, 2011.

POPESCU, L. P. et al. **Multilayer perceptron and neural networks**, 2009.

PRICE-WHELAN, A. M. et al. **THE ASTROPY PROJECT: BUILDING AN INCLUSIVE, OPEN-SCIENCE PROJECT AND STATUS OF THE V2.0 CORE PACKAGEarXiv**, 2018.

PYLE, D. **Data Preparation for Data Mining**, 1999.

PYTHON MACHINE LEARNING BOOK. **Chapter 6 - Learning Best Practices for Model Evaluation and Hyperparameter Tuning**. 2013. Disponível em: https://nbviewer.jupyter.org/github/rasbt/python-machine-learning-book/blob/master/code/ch06/ch06.ipynb#K-fold-cross-validation. Acesso em: 9 maio 2021.

RAKTHANMANON T.; KEOGH E. **Fast-shapelets: A fast algorithm for discovering robust time series shapelets**, 2013.

RASCHKA, S. **Model evaluation, model selection, and algorithm selection in machine learningarXiv**, 2018.

REZENDE, S. O. **Sistemas Inteligentes: Fundamentos e Aplicações**. Barueri, Brasil: Manole, 2003.

RICHARDS, J. W. et al. On machine-learned classification of variable stars with sparse and noisy time-series data. **Astrophysical Journal**, v. 733, n. 1, 2011.

SCHÄFER, P. The BOSS is concerned with time series classification in the presence of noise. **Data Mining and Knowledge Discovery**, v. 29, n. 6, 2015.

SCHÄFER, P.; LESER, U. **Fast and accurate time series classification with WEASEL**. International Conference on Information and Knowledge Management, Proceedings. **Anais**...2017a

SCHÄFER, P.; LESER, U. **Multivariate time series classification with Weasel+MUSEarXiv**, 2017b.

SEBASTIANI, P.; RAMONI, M. **Clustering continuous time series.** In: Proceedings of the Eighteenth International Conference on Machine Learning. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2001.

SFETSOS, A.; SIRIOPOULOS, C. **Time series forecasting with a hybrid clustering scheme and pattern recognition**. Trans. Sys. Man Cyber. Part A, IEEE Press, Piscataway, NJ, USA, v. 34, n. 3, p. 399–405, maio 2004.

SHALLUE, C. J.; VANDERBURG, A. **Identifying exoplanets with deep learning: A five planet resonant chain around kepler-80 and an eighth planet around kepler-90arXiv**, 2017.

SHATKAY, H.; ZDONIK, S. B. **Approximate queries and representations for large data sequences.** In: Proceedings of the Twelfth International Conference on Data Engineering. Washington, DC, USA: IEEE Computer Society, 1996.

SHUMWAY, R. H.; STOFFER, D. S. **Time Series Analysis and its Applications: with R examples**. 2. ed. New York, USA: Springer, 2006.

SILVA, D. F.; SOUZA, V. M. A. de; BATISTA, G. E. **Time series classification using compression distance of recurrence plots**. In: . [S.l.: s.n.], 2013.

SOUZA, A. A. DE; VALIO, A. Estudo da atividade estelar da Kepler-289 a partir da modelagem de trânsitos planetários. **Revista Brasileira de Ensino de Física**, v. 41, n. 4, 2019.

SRIKANT, R.; AGRAWAL, R. **Mining sequential patterns: Generalizations and performance improvements**. In: Proceedings of the 5th International Conference on Extending Database Technology: Advances in Database Technology. London, UK, UK: Springer-Verlag, 1996.

STURROCK, G. C.; MANRY, B.; RAFIQI, S. Machine Learning Pipeline for Exoplanet Classification. **SMU Data Science Review**, 2019.

TIME SERIES CLASSIFICATION. Algorithms. 2021. Disponível em: http://www.timeseriesclassification.com/algorithm.php. Acesso em: 9 maio 2021.

TOWARDS DATA SCIENCE. A Brief Introduction to Time Series Classification Algorithms. 2020. Disponível em: https://towardsdatascience.com/a-brief-introduction-to-time-series-classification-algorithms-7b4284d31b97. Acesso em: 9 maio 2021.

TREU, T.; MARSHALL, P. J.; CLOWE, D. Resource Letter GL-1: Gravitational Lensing. **American Journal of Physics**, v. 80, n. 9, 2012.

TWICKEN, J. D. et al. DETECTION OF POTENTIAL TRANSIT SIGNALS IN 17 QUARTERS OF KEPLER DATA: RESULTS OF THE FINAL KEPLER MISSION TRANSITING PLANET SEARCH (DR25) . **The Astronomical Journal**, v. 152, n. 6, 2016.

WANG, J.; HAN, J. Bide: **Efficient mining of frequent closed sequences**. In: Proceedings of the 20th International Conference on Data Engineering. Washington, DC, USA: IEEE Computer Society, 2004.

WITTEN, I. H.; FRANK, E. **Data mining: practical machine learning tools and techniques**, 2005.

YADAV, R. N.; KALRA, P. K.; JOHN, J. **Time series prediction with single multiplicative neuron model**. Appl. Soft Comput., Elsevier Science Publishers B. V., Amsterdam, The Netherlands, The Netherlands, v. 7, n. 4, p. 1157–1163, ago. 2007.

YAQOOB, T. **Exoplanets and Alien Solar Systems New Earth Labs**, 2011.

YE, L.; KEOGH, E. **Time series shapelets: a new primitive for data mining**. In: Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining. New York, NY, USA: ACM, 2009.

YE, L.; KEOGH, E. Time series shapelets: A novel technique that allows accurate, interpretable and fast classification. **Data Mining and Knowledge Discovery**, v. 22, n. 1–2, 2011.

YU, L. et al. **Identifying exoplanets with deep learning III: Automated triage and vetting of tess candidatesarXiv**, 2019.

ZALEWSKI, W. **Modelagem Simbólica de Padrões Morfológicos para a Classificação de Séries Temporais**. Dissertação (Doutorado) – Universidade Federal do Paraná- UFPR, 2015.