Revista brasileira de iniciação científica
ISSN: 2359-232X

CLASSIFICAÇÃO AUTOMÁTICA DE OBJETOS ASTRONÔMICOS POR MEIO DA ANÁLISE DE SÉRIES TEMPORAIS

AUTOMATIC CLASSIFICATION OF ASTRONOMICAL OBJECTS USING TIME SERIES ANALYSIS

CLASIFICACIÓN AUTOMÁTICA DE OBJETOS ASTRONÓMICOS POR MEDIO DEL ANÁLISIS DE SERIES TEMPORAIS

Patricia Oliveira Montanger[1]
Willian Zalewski[2]

Resumo: A coleta de informações temporais aumenta a cada dia e se torna cada vez mais necessária a criação de meios para que esses dados sejam analisados e compreendidos de maneira eficaz. Essas informações podem ser representadas por meio de séries temporais, como curvas de luz de estrelas que possuem exoplanetas, e compreender os eventos contidos nessas séries temporais exige métodos de mineração de dados e análises com algoritmos de aprendizagem de máquina. Portanto, neste projeto buscamos desenvolver métodos para analisar séries a partir da descoberta de shapelets, ou seja, subsequências que maximizam a discriminação entre classes num conjunto de dados.
Palavras-chave: Séries temporais. Curvas de luz. Aprendizado de máquina. Mineração de dados.

Abstract: The collection of temporal information increases every day and it becomes increasingly necessary to create the means for these data to be analyzed and understood in an effective way. This information can be represented by times series, such as light curves of stars that have exoplanets, and understanding the events in these time series requires data mining methods and analysis with machine learning algorithms. Therefore, in this project we seek to develop methods to analyze series from the discovery of shapelets, that is, subsequences that maximize the discrimination between classes in a set of data.
Keywords: Time series. Light curves. Machine learning. Data Mining.

Resumen: La recolección de información temporal aumenta cada día y se vuelve más necesaria la creación de medios para que esos datos sean analizados y comprendidos de manera eficaz. Esta información puede ser representada por series temporales, como curvas de luz de estrellas que poseen exoplanetas, y comprender los eventos contenidos en estas series temporales requiere métodos de minería de datos y análisis com algoritmos de aprendizaje de máquina. Por esto, en este proyecto buscamos desarrollar métodos para analizar series a partir del descubrimiento de shapelets, o sea, subsecuencias que maximizan la discriminación entre clases en un conjunto de datos.
Palabras-clave: Series temporales. Curvas de luz. Aprendizaje de máquina. Minería de datos.

Envio: 25/02/2019 **Revisão: 25/02/2019** **Aceite: 27/05/2019**

[1] Graduanda em Engenharia Física. Universidade Federal da Integração Latino-Americana. E-mail: patricia.montanger@aluno.unila.edu.br.
[2] Doutor. Universidade Federal da Integração Latino-Americana. E-mail: willian.zalewski@unila.edu.br.

Revista brasileira de iniciação científica
ISSN: 2359-232X

Introdução

Em diversas áreas do conhecimento estão presentes informações que estão sujeitas a variações temporais, como na economia com os preços diários de ações, na medicina em eletrocardiogramas, na meteorologia e na astrofísica com a identificação de objetos celestes. Estes exemplos comuns do dia a dia evidenciam como o desenvolvimento tecnológico referente ao armazenamento e ao processamento de dados temporais deve ser pesquisado. O tipo de dado temporal mais comum é chamado de série temporal, a qual pode ser entendida como um conjunto ordenado de observações registradas cronologicamente. As abordagens tradicionais para a análise de séries temporais são baseadas em métodos estatísticos, os quais não se mostram eficientes, pois estes métodos analisam cada dado da série independentemente, sem levar em conta o fato de que existe uma relação temporal entre cada uma das observações realizadas. Mediante esta restrição das abordagens estatísticas, muitos estudos propuseram a utilização de técnicas de aprendizado de máquina, as quais são baseadas na inferência indutiva, que possibilita derivar novos conhecimentos automaticamente a partir de outros previamente conhecidos.

Neste projeto a informação temporal de interesse é a variação da intensidade luminosa de corpos celestes coletada pelo telescópio Kepler da NASA. Esse telescópio foi projetado para pesquisar uma determinada região da Via Láctea com o objetivo de detectar e caracterizar planetas do tamanho da Terra ou menores, isso dentro ou perto da zona habitável. O Kepler fornece uma base de dados de cadência longos (LC), que são somados em 29,424 minutos (duzentos e setenta integrações), enquanto os dados curtos de cadência (SC) somam um minuto (nove integrações) entre cada registro, a união desses registros nos fornece o que denominamos curvas de luz. Esse tipo de dado possibilita obter informações sobre valores de massa e raio de estrelas, supernovas, sistemas binários e exoplanetas, sendo este último o nosso objetivo principal de análise. Pelo que conhecemos da literatura, ainda não foram realizadas análises de curvas de luz combinando técnicas de aprendizado de máquina e estratégias de identificação de padrões em séries temporais, em especial pela aplicação da transformada *shapelet*. Assim, com o desenvolvimento deste trabalho espera-se contribuir de modo significativo para o processo de classificação automática de objetos celestes, de modo a

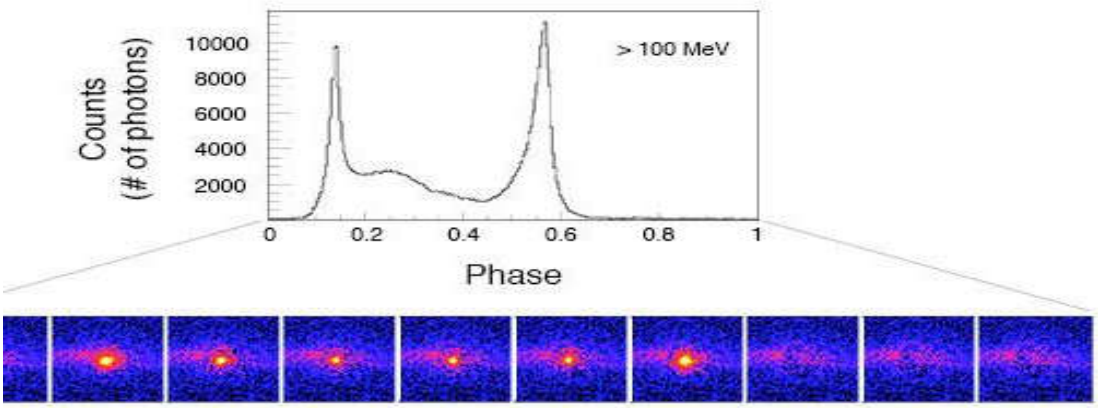
Revista brasileira de iniciação científica
ISSN: 2359-232X

melhorar o desempenho dos algoritmos de classificação, agregando informações e auxiliando no processo de tomada de decisões de astrônomos.

Fundamentação Teórica

Nosso objeto de estudo são as curvas de luz que identificam exoplanetas, as quais podem ser entendidas como séries temporais constituídas pela variação do brilho de um objeto celeste no tempo. Essa variação no brilho das estrelas também pode ser denominada variação da amplitude, onde menor é o valor da amplitude quanto maior for o brilho, e ocorre devido a um fenômeno chamado trânsito planetário. Este fenômeno ocorre quando um exoplaneta realiza a órbita em torno de sua estrela e visto que o telescópio aponta sempre numa mesma direção, em algum momento o exoplaneta passa entre a estrela e o telescópio, o que tem como consequência a diminuição do brilho capturado pelas imagens feitas pelo telescópio. Reunindo as imagens capturadas com o passar do tempo, de pelo menos uma órbita completa, começamos a observar a formação de uma curva de luz e o padrão que existe nela quando se trata de uma curva de uma estrela que possui exoplanetas. Na Figura 1 abaixo fica claro o conceito de trânsito planetário, vemos nas primeiras imagens em que a estrela apresenta maior brilho, a curva está num ponto mais alto e que com o passar do tempo o brilho da estrela diminui, nas últimas imagens, assim como na curva que ao fim do gráfico vai se aproximando de pontos mais baixos.

Figura 1 - Formação de uma curva de luz através de imagens capturadas por telescópio



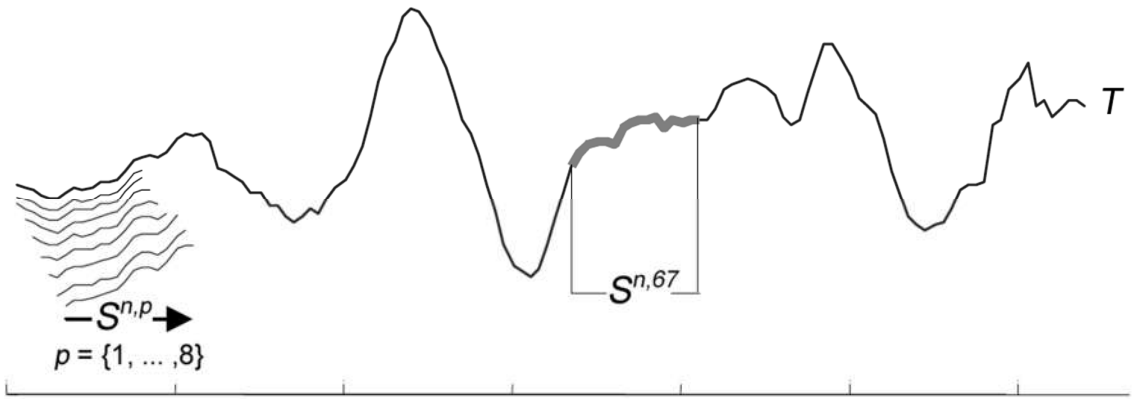
Fonte: Kepler and K2 Science Center, 2018.

Revista brasileira de iniciação científica
ISSN: 2359-232X

Para compreender este trabalho é necessário definir alguns conceitos, como séries temporais, subsequências, aprendizado de máquina e shapelets. Uma série temporal é um conjunto de m variáveis dadas por T= t1, t2, …, tm que são organizados por ordem temporal e espaçadas em intervalos de tempo iguais. Uma subsequência pode ser definida em uma série T de tamanho m como uma subsequência S de comprimento $l \leq m$ de posições contíguas de T, isto é, S = tp,…, tp + l-1, para $1 \leq p \leq m - l + 1$.

Um dos processos para se obter uma subsequência de tamanho l em determinada posição p de uma série temporal T é a aplicação de uma janela deslizante, como está ilustrado na Figura 1, onde observamos oito subsequências extraídas das posições p = {1,2,...,8} e ainda uma subsequência extraída da posição p = 67.

Figura 2 – Processo de aplicação de uma janela deslizante



Fonte: Keogh, 2003.

Para identificar os dados desejados precisamos buscar os padrões dessas curvas T e a partir desses padrões reconhecer curvas, que representam exoplanetas para qualquer base de dados existente. Para cumprir com esse objetivo existem métodos como a técnica de Doppler, que é baseada na observação do trânsito do planeta em torno de sua estrela, a técnica de análise da velocidade radial, a do tempo de eclipses, entre outros. Algumas dessas técnicas geram curvas de fácil reconhecimento visual, porém em nossos dados não temos essa

Revista brasileira de iniciação científica
ISSN: 2359-232X

facilidade devido a grande quantidade de informações e de ruídos, é por isso que utilizamos os métodos baseados em aprendizado de máquina. O aprendizado de máquina consiste na construção de um modelo a partir de uma base de dados previamente conhecida que possibilite a identificação/classificação de novos dados automaticamente, sem necessidade de interferência humana durante o processo. Utilizamos alguns dos principais algoritmos de aprendizado de máquina para realizar nossos experimentos, tal como o Decision Trees, o Support Vector Machines, o Naive Bayes, o Nearest Neighbors e o Neural Network. Sendo o mais conhecido deles o Decision Trees, que é um método supervisionado (quando é utilizado um agente externo que indica à rede a resposta desejada para o padrão de entrada) que tem como objetivo criar um modelo que prevê o valor de uma variável de destino, aprendendo regras de decisão simples inferidas a partir de dados de treino, quanto mais profunda for a árvore, mais complexa será a decisão. Apresenta vantagens como sua simplicidade e fácil interpretação, afinal as árvores podem ser visualizadas esquematicamente. Diferente de outros algoritmos, não é obrigatório que seja feita a normalização dos dados e é um modelo de caixa branca, ou seja, a explicação para as classificações realizadas pode ser facilmente compreendida pela lógica booleana. Também utilizamos o Support Vector Machines, o que esse algoritmo tem por objetivo encontrar uma linha de separação, também denominada de hiperplano, entre dados de duas ou mais classes. Essa linha busca maximizar a distância entre os pontos mais próximos em relação a cada uma das classes, essa distância entre o hiperplano e o primeiro ponto de cada classe costuma ser chamada de margem, definindo assim cada ponto pertencente a cada uma das classes, e em seguida maximiza a margem. Ou seja ela primeiro classifica as classes corretamente e depois em função dessa restrição define a distância entre as margens. Portanto, esse algoritmo é melhor aplicável em espaços dimensionais elevados e apresenta possibilidades para a adição de parâmetros que auxiliam na classificação. Já o algoritmo Naive Bayes é baseado na aplicação do teorema de Bayes. Apresenta vantagens pois exige uma pequena quantidade de dados de treinamento para estimar os parâmetros necessários e porque pode ser extremamente rápido em comparação com métodos mais sofisticados.

O Nearest Neighbors oferece métodos supervisionados e não supervisionados, o princípio por trás deste algoritmo é encontrar um número de amostras de treinamento mais

Revista brasileira de iniciação científica
ISSN: 2359-232X

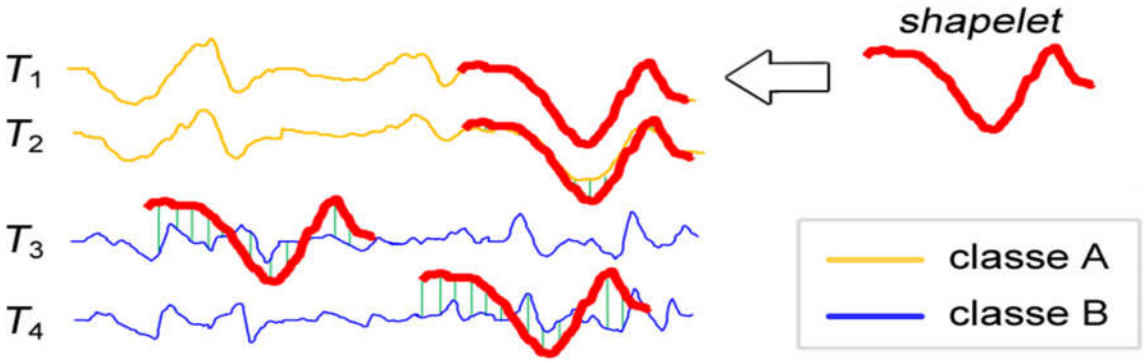
próximas da distância de um novo ponto e prever a classe a partir delas. O número de amostras pode ser uma constante definida pelo usuário ou variar com base na densidade local de pontos, que é o aprendizado baseado no raio. A distância pode, em geral, ser qualquer medida métrica, onde a distância euclidiana é a escolha mais comum. Apesar de sua simplicidade, o Nearest Neighbors teve sucesso em um grande número de problemas de classificação e regressão, como em imagens de satélites e por muitas vezes foi bem sucedido em situações de classificação onde o limite de decisão é muito irregular.

E por fim, o algoritmo Neural Network, que representa modelos computacionais inspirados pelo sistema nervoso central biológico, os quais são capazes de realizar o aprendizado de máquina bem como o reconhecimento de padrões. Seus elementos de processamento são neurônios que produzem a soma ponderada das entradas e aplicam o resultado a uma função de transferência não linear, para gerar uma saída. Este modelo têm se mostrado adequado para o reconhecimento de padrões em reconhecimento de voz, biometria e sensoriamento remoto, no entanto ele não faz isto de forma explícita. As redes neurais são o melhor exemplo de um sistema sub simbólico, onde mesmo que cada parte de um padrão apresentado a uma rede tenha um significado explícito, associável a um símbolo de nosso modelo do mundo real, a representação interna dos dados no processador (a rede) não é explícita e não possui significado. Um método sub simbólico tipicamente é incapaz de explicar porque chegou a uma determinada conclusão, uma vez que um mapeamento explícito de causa-e-efeito não existe (Aldo von Wangenheim, 2015, p. 1).

Neste trabalho utilizamos esta funcionalidade do aprendizado de máquina para aplicar as primitivas de séries temporais conhecidas como *shapelets*, que são subsequências de séries temporais e são consideradas as melhores representantes de uma classe, além de possuírem benefícios como velocidade e precisão. A adoção desta primitiva se justifica pois *shapelets* são características morfológicas locais, enquanto a maioria das estratégias com séries temporais consideram características globais, as quais podem apresentar mais níveis de ruído e distorções, e consequentemente reduzir a inteligibilidade dos métodos de classificação.

Revista brasileira de iniciação científica
ISSN: 2359-232X

Figura 3 – Primitiva shapelet para a classificação de séries temporais



Fonte: Keogh, 2011.

Outros conceitos que merecem atenção são os de normalização, o de junção cruzada e de distância euclidiana, os quais são fundamentais para cumprir com os objetivos deste trabalho.

A normalização é um processo onde os dados são ajustados de modo que todos os seus valores pertençam a um determinado intervalo, o que permite deixar todos os dados em uma mesma escala e garante que podemos realizar uma comparação direta entre seus valores. Não existe apenas uma técnica de normalização, mas para a realização deste trabalho escolhemos a normalização-Z, nessa técnica os valores das séries temporais são ajustados de modo que a média de seus valores seja nula e que o desvio padrão seja unitário. O formato da série temporal original é conservado e para o cálculo dos valores normalizados é necessário conhecer o valor médio e o desvio padrão dos dados originais, o que pode ser feito automaticamente por pacotes para Python e se torna muito útil para a aplicação em arquivos de grandes dimensões como os das curvas de luz.

A junção cruzada é utilizada para criar uma combinação de cada linha de duas tabelas, que nessa situação são as linhas com os valores das leituras de intensidade luminosa das estrelas com o passar do tempo. Todas as combinações de linhas são geradas como resultado final da junção, isso é comumente chamado de um produto cartesiano. Apesar do uso comum, essa técnica deve ser utilizada com cuidado, afinal se cruzarmos uma tabela de mil linhas com

Revista brasileira de iniciação científica
ISSN: 2359-232X

outra de mil, terminaremos com um milhão de resultados, o que custaria muito tempo do projeto devido ao grande desempenho computacional necessário para realizar este processamento.

A distância euclidiana é definida como uma medida que determina qual a distância em linha reta entre dois pontos, os quais pertencem a um espaço de m dimensões, onde m corresponde ao tamanho de uma determinada série temporal já normalizada. A distância Euclidiana $Dist_{ED}$ entre duas séries temporais Tx e Ty , é definida conforme a Equação 1:

$$Dist_{ED}(Tx , Ty) = \sqrt{\sum(x_i \quad y_i)^2} \qquad (1)$$

onde o somatório vai de i=1 até m, e i indica cada dimensão em cada iteração para o cálculo da distância. O vetor de distâncias euclidianas entre os pontos das séries é denominado de perfil de distâncias.

Método

Neste trabalho analisamos as curvas de luz provenientes da base de dados do Kaggle (www.kaggle.com), que consiste em uma plataforma para hospedar competições de Data Science públicas, privadas e acadêmicas. Também existem competições de aprendizado disponibilizadas pelo próprio Kaggle ou por empresas para treinamento de habilidades. A plataforma também armazena e disponibiliza dados sobre assuntos diversos, os chamados datasets; também possui fóruns para troca de conhecimentos entre seus usuários.

Por este meio obtemos as curvas de luz de estrelas do telescópio Kepler, as quais foram fornecidas em formato CSV, que são arquivos de texto nos quais os valores são separados por vírgula e podem ser visualizados em softwares como o Microsoft Excel, facilitando assim a compreensão e manipulação destes dados, visto que se tivéssemos retirado estes dados diretamente do Kepler seu formato seria de muito mais difícil compreensão. No conjunto de dados com o qual escolhemos trabalhar cada estrela tem um rótulo binário de 2 ou 1, o 2 indica que a estrela está confirmada para ter pelo menos um exoplaneta em sua órbita e o 1 indica que aquela estrela não possui exoplanetas. Nossos dados são divididos em

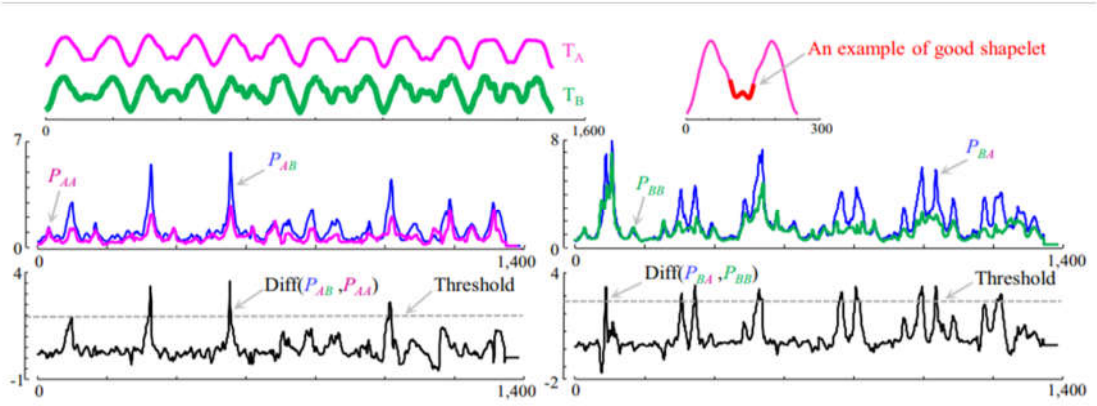
Revista brasileira de iniciação científica
ISSN: 2359-232X

duas categorias, a de treino do algoritmo de aprendizagem de máquina e a de teste do algoritmo. O conjunto de treino é composto por trinta e sete estrelas confirmadas com exoplanetas e cinco mil e cinquenta estrelas sem exoplanetas, enquanto o conjunto de teste possui cinco estrelas com exoplanetas confirmadas e quinhentas e sessenta e cinco estrelas sem exoplanetas. Porém, devido à alta dimensionalidade dos dados, para realizar os experimentos utilizamos todas as curvas de estrelas com exoplanetas e apenas sessenta e quatro e cem séries de estrelas sem exoplanetas dos dados de treino e de teste, respectivamente.

Os dados das curvas de luz foram analisados considerando duas abordagens: na primeira pela utilização direta dos dados brutos; e na segunda pela aplicação da estratégia transformada *shapelet*. Para o desenvolvimento da segunda abordagem foi utilizada a técnica Matrix Profile, que auxilia com velocidade e intuitivamente na descoberta de shapelets em séries temporais, ou seja, na busca por subsequências que discriminem uma determinada classe entre todos os dados fornecidos. Para compreender melhor esse processo consideramos duas séries temporais, as quais denominamos Ta e Tb, de classe 1 e 2 respectivamente (que representam as estrelas com e sem exoplanetas dos dados) e realizamos a concatenação entre elas aplicando a lógica de junção cruzada utilizando a função *tools.signal_cross_join* para Python, formando quatro curvas distintas. A partir dessas curvas calculamos a diferença (distância euclidiana) entre elas, resultando em duas novas curvas de distâncias e finalmente com essas curvas definimos um parâmetro, o desvio padrão, considerando que os pontos de máximo acima desse valor são nossas possíveis *shapelets*. Podemos visualizar mais claramente este processo na Figura 4:

Revista brasileira de iniciação científica
ISSN: 2359-232X

Figura 4 - Matrix Profile para a determinação de Shapelets



Fonte: Keogh, 2003.

O processo descrito acima foi realizado mediante a utilização de bibliotecas como BioSPPy (Biosignal Processing in Python), Pandas e NumPy, que são fundamentais para a programação científica. O BioSPPy é uma biblioteca que reúne vários métodos de processamento de sinais e reconhecimento de padrões, sendo principalmente utilizado com biosinais, medidas de propriedades físicas de sistemas biológicos, que incluem a medição de propriedades no nível celular e de atividade elétrica, como o sistema elétrico do coração através de eletrocardiogramas, por exemplo. Utilizamos essa biblioteca na área da astrofísica pois biosinais também podem ser variantes no tempo, logo podem ser séries temporais que é o nosso objeto de abordagem neste trabalho. A biblioteca Pandas foi utilizada para a coleta e preparação de dados, pois possui ferramentas para ler e gravar dados na memória em diferentes formatos (arquivos CSV e de texto); permite que colunas possam ser inseridas e excluídas de estruturas de dados para mutação de tamanho; e possibilita operações de combinação e divisão em conjuntos de dados. Essas funções são extremamente úteis na manipulação de dados como os das curvas de luz, onde a representação é feita em matrizes e a quantidade de informações é muito grande. O pacote NumPy é fundamental para programação em Python, dado sua sofisticada técnica para lidar com dados N-dimensionais; suas ferramentas para integrar código C/C++ e Fortran; suas funções em álgebra linear,

Revista brasileira de iniciação científica
ISSN: 2359-232X

transformada de Fourier e capacidade de gerar números aleatórios. Além disso, o NumPy também pode ser usado como um contêiner multidimensional eficiente de dados genéricos, isso permite que o NumPy integre-se de forma fácil e rápida a uma ampla variedade de bancos de dados.

Ainda, realizamos a normalização de nossos arquivos de dados, isso porque as técnicas de redução de dados e de aprendizado de máquina são sensíveis às diferenças de escala entre os pontos característicos. Para esse fim, utilizamos a função *tools.normalize,* que é definida para normalizar as colunas de todas as listas fornecidas, mas também pode normalizar as colunas existentes em cada lista individualmente ou, alternativamente, para cada linha da matriz.

Para que os experimentos com Matrix Profile fossem realizados, precisamos contar com o desenvolvimento de dois códigos em Python através do Jupyter Notebook (meio para desenvolver software de código aberto, padrões abertos e serviços para computação interativa em dezenas de linguagens de programação), sendo um em específico para os dados de treino e outro para os dados de teste, onde um script complementa o outro visto que os processos são muito semelhantes, porém com o diferencial de que quando executamos o código de teste, já estamos utilizando arquivos gerados no código de treino que contém informações como as subsequências selecionadas e suas respectivas posições.

Desenvolvemos um terceiro código com algoritmos de aprendizagem de máquina, os quais são treinados por meio dos dados de treino, de forma que o algoritmo possa resolver sozinho problemas parecidos porém com novos dados, os dados de teste. E são nesses algoritmos que introduzimos os resultados de shapelets obtidos. Nesse contexto, foram utilizados os cinco diferentes algoritmos de aprendizado de máquina apresentados, o Decision Trees, o Support Vector Machines, o Naive Bayes, o Nearest Neighbors e o Neural Network, todos estes algoritmos foram manipulados por meio da biblioteca Scikit-Learn para Python.

Finalmente, com a utilização do Cluster C3HPC (DINF-UFPR), que possui seis nodos de processamento, cada um com quatro sockets 3.30GHz (oito núcleos por socket) e 256 GB de RAM executamos o código desenvolvido para a seleção das *shapelets*. Escolhemos executar os experimentos no Cluster C3HPC pois este é um sistema capaz de combinar vários computadores para trabalharem em conjunto, onde os computadores ficam integrados e atuam

Revista brasileira de iniciação científica
ISSN: 2359-232X

conjuntamente no processamento de dados e execução de tarefas complexas, que exigem muitos processadores. Com os resultados obtidos para cada algoritmo, os comparamos e definimos qual apresentou o resultado mais satisfatfoio na classificação de estrelas com exoplanetas e sem exoplanetas.

Resultados

Os algoritmos de aprendizado de máquina foram aplicados para os dados originais das séries temporais e seus resultados estão apresentados na Tabela 1, por meio de métricas como precisão que representa a quantidade de itens selecionados pelo algoritmo que são relevantes, neste caso os exoplanetas, e a métrica medida-F, que representa a quantidade de itens relevantes que foram selecionados. Podemos observar em todos os algoritmos uma porcentagem de acerto muito maior para as estrelas sem exoplanetas do que para as com exoplanetas, com exceção dos algoritmos Vector Machines e Nearest Neighbors que apresentaram zero acertos em relação aos exoplanetas.

Diferente do esperado para o Neural Network, seus resultados também não foram satisfatórios, apresentando para as métricas de precisão e medida-F acertos percentuais de 0.01 e 0.02 respectivamente, o que pode ter acontecido pela falta de parâmetros que auxiliassem o algoritmo a tomar decisões na hora de realizar a classificação. Já os melhores resultados foram obtidos com o algoritmo Decision Trees, que tem um processo de classificação mais simples, porém nesta situação foi o mais adequado.

Tabela 1: Resultados considerando dados originais não normalizados.

Dados originais		**DecisionTree**	**VectorMachine**	**KNeighbors**	**NaiveBayes**	**NeuralNet**
Exoplanetas	**Precisão**	0.33	0.00	0.00	0.01	0.01
	Medida-F	0.36	0.00	0.00	0.02	0.02
Outros	**Precisão**	0.99	0.99	0.99	1.00	0.99
	Medida-F	0.99	1.00	1.00	0.03	0.29

Fonte: Autoria própria, 2018.

Revista brasileira de iniciação científica
ISSN: 2359-232X

Durante a realização dos experimentos percebemos que o processamento dos algoritmos custaria muito tempo, visto que a utilização do Cluster C3HPC nos permitia apenas sete dias de processamento e nos primeiros experimentos o tempo era excedido sem que o experimento fosse concluído. Esta questão nos obrigou a reduzir a quantidade de dados utilizados e realizar alterações no código, tendo sido percebido, como já era esperado, que o que demandava maior tempo era a técnica de junção cruzada. Após as alterações, os experimentos com os dados originais foram completados, mas ainda devido a alta dimensionalidade dos dados e a complexidade computacional do algoritmo que estamos utilizando, a execução dos experimentos com os dados em que aplicamos a análise para encontrar as subsequências *shapelets* não foram completados até o momento da publicação deste trabalho.

Conclusões

Quando analisamos a tabela obtida como resultado deste trabalho percebemos em alguns algoritmos uma proporção muito maior de acertos para as estrelas sem exoplanetas, sendo um fato que contribuiu para esses resultados a grande discrepância existente na base de dados que escolhemos, pois tanto nos dados de teste como nos de treino cerca de noventa e nove por cento dos objetos eram estrelas sem exoplanetas e isto influencia fortemente quando utilizamos algoritmos de aprendizado de máquina, pois ele tende a ser influenciado pelo tipo de dado que existe em maior quantidade. Outro fator a ser considerado é que devido a limitação computacional não foi possível utilizar cem por cento dos dados nos experimentos realizados, o que com certeza trouxe consequências para o resultado final.

Logo, em projetos futuros pretendemos utilizar bases de dados com uma menor divergência na quantidade de itens e ainda dados em que tenhamos a certeza da existência de apenas duas classes, que é o tipo de análise que determinamos para os algoritmos de aprendizado de máquina que utilizamos, isto pois sabemos que algumas das estrelas possuem exoplanetas, mas isso não pode nos dar a certeza de que as estrelas que não possuem exoplanetas apresentam as mesmas características em suas curvas de luz, afinal apesar de ser a classe das estrelas sem exoplanetas, ainda podem existir várias subclasses que

Revista brasileira de iniciação científica
ISSN: 2359-232X

desconhecemos dentro desta classe de estrelas, contendo vários outros tipos de objetos astronômicos em suas órbitas.

Referências

ZALEWSKI, Willian. **Modelagem Simbólica de Padrões Morfológicos para a Classificação de Séries Temporais**. Curitiba, PR, p. 55-58, 2015.

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

The UCR Matrix Profile Page. Disponível em: http://www.cs.ucr.edu/~eamonn/MatrixProfile.html. Acesso em: 2018.

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

AMARAL, Fernando. **Introdução à Ciência de Dados:** mineração de dados e big data. Alta Books Editora, 2016.

REZENDE, Solange Oliveira. **Sistemas Inteligentes:** fundamentos e aplicações. Editora Manole Ltda, 2003.