Classificação de curvas de luz de objetos transientes utilizando redes neurais LSTM

Marco Antônio Barroca[1], Mateus dos Santos[1], Rafael Santos Oliveira[2]
[1] *Centro Brasileiro de Pesquisas Físicas,*
[2] *Universidade Federal de Minas Gerais*
(Dated: 20 de outubro de 2023)

Uma das principais aplicações de IA na área científica é como classificador para experimentos que possuem uma alta quantidade de dados e exigem agilidade na análise. Nós implementamos uma rede neural LSTM para classificar dados de curvas de luz do experimento Large Synoptic Survey Telescope (LSST) como parte do The Photometric LSST Astronomical Time-series Classification Challenge (PLAsTiCC). Após simplificar o problemas para classes generalizadas nosso modelo mostrou-se capaz de classificar parcialmente os objetos astronômicos estudados.

Estrutura: Fazemos em uma breve introdução de IAs, curvas de luz, redes LSTM e do problema de classificação a ser resolvido. Logo após, explicamos os métodos utilizados para simplificar o problema, pré-processar os dados e construir a rede. Em seguida, divulgamos e discutimos os resultados da classificação. Por último, apresentamos nossas conclusões.

I. INTRODUÇÃO

A inteligência artificial (IA), pode ser classificada como uma área de pesquisa que investiga formas de habilitar o computador a realizar tarefas nas quais, até o momento, o ser humano tem um melhor desempenho. A IA pode se manifestar de várias formas, em chatbots que podem entender os problemas dos clientes mais rapidamente e fornecer respostas mais eficientes, em assistentes inteligentes que precisam analisar informações críticas de grandes conjuntos de dados de texto livre e em aplicações científicas, como de classificar e tomar decisões na crescente exponencial de dados astronômicos.[1]

Recentemente o Large Synoptic Survey Telescope (LSST), propôs um desafio no site Kaggle, um subsidiario da google, chamado The Photometric LSST Astronomical Time-series Classification Challenge (PLAsTiCC).[2] Trata-se de um desafio com o intuito de classificar curvas de luz, dados do brilho observado de objetos celestes em função do tempo, os quais foram simuladas em preparação para observações do LSST. Essas curvas são capazes de revelar a presença de alguns fenômenos, como por exemplo, supernovas[3]. O LSST revolucionará nossa compreensão do céu, porém esse tipo de estudo é dificultado pelo grande volume de dados obtidos pelo telescópio, tornando indispensáveis procedimentos automáticos de análise para diferenciar e classificá-las. Neste desafio, foi colocado a questão: Quão bem podemos classificar objetos no céu que variam em brilho a partir de dados simulados de séries temporais LSST? A partir disso, surgiu a necessidade de um desafio de dados para ajudar a classificar essas fontes astronômicas e descrever o conjunto de dados PLAsTiCC e o desafio de dados Kaggle. [4]

Antes da utilização de IA, esse tipo de classificação ainda dependia de uma análise manual. Utilizavam-se métodos estatísticos comuns como template fitting[5], porém não escaláveis para um grande volume de dados. Geralmente um grupo de especialistas eliminava manualmente casos óbvios de falsos positivos, algo que por si só pode levar alguns dias. Dos dados restantes, cada caso deveria ser revisado por pelo menos três especialistas, o que pode levar a divergências sobre um caso particular, visto que os especialistas poderiam não possuir a mesma definição para classificação. Por essas razões, é necessário um sistema confiável que selecione repetidamente os candidatos mais importantes que serão revisados manualmente para confirmação em um estágio posterior.

Devido ao fato da curva de luz, ser uma função que varia com o tempo, uma das formas possíveis de resolver o problema do PLAsTiCC, é utilizar uma Rede Neural Long short-term memory (LSTM). A LSTM é uma arquitetura de rede neural recorrente (RNN), desenvolvida por Hochreiter Schmidhuber em 1997, as redes recorrentes tem a característica de fazer que algumas informações em intervalos arbitrários persistam na rede através de um loop. A RNN por definição possui uma estrutura de repetição mais simples que as LSTMs, pois existe apenas uma única camada tanh, figura 1, já as LSTMs, possui uma estrutura em cadeia que contém quatro camadas de redes neurais e diferentes blocos de memória chamados células como demonstrado na figura 2. Sendo assim as LSTMs são bem adequada para classificar, processar e prever séries temporais com intervalos de tempo de duração desconhecida. [6]

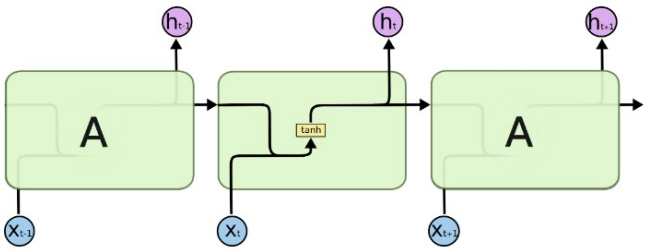


Figura 1: Funcionamento de uma rede RNN. [7]

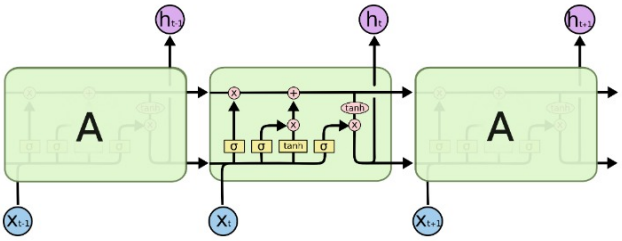


Figura 2: funcionamento de uma rede LSTM. [7]

II. METODOLOGIA

Os dados das curvas de luz foram obtidos através do LSST PLAsTiCC através do desafio no Kaggle[2], para que o projeto fosse finalizado no tempo previsto da EA-FEXP foi utilizado um recorte dos dados totais. Ao total foram utilizados dois conjunto de dados, o primeiro possuía 7848 objetos, sendo que 10% desses objetos foram usados para formar um conjunto de validação e o restante formou o conjunto de treino, o segundo conjunto de dados possuía 19920 objetos e foi utilizado como conjunto de teste.

Os dados são tabulares e consistem de informações de medida do fluxo, o erro da medição, o momento da medição em Data Juliana Modifica (mjd), o filtro utilizado (o espectro da emissão do objeto, ugriZY, representado por um valor numérico de zero à cinco), uma variável booleana (0 ou 1) que acusa a detecção ou não-detecção, a classe do objeto e um identificador único. Na tabela I é possível visualizar um exemplo.

Fluxo	Erro	mjd	Filtro	Detecção	Classe	Id
-544.810	3.623	59750.4	2	1	92	615
-816.434	5.553	59750.4	1	1	88	713
-471.386	3.802	59750.4	3	1	42	730
...	...	...	...	...	...	...
-388.985	11.395	59750.4	4	1	90	745
-2.940	1.771	59798.3	3	0	90	116720
-12.810	5.380	59798.4	5	0	92	117016
...	...	...	...	...	...	...

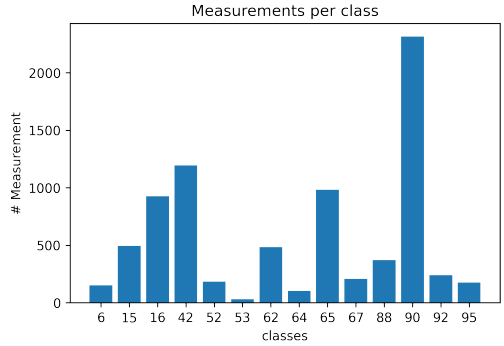
Tabela I: Tabela exemplificando o conjunto de dados.

A. Pré-processamento

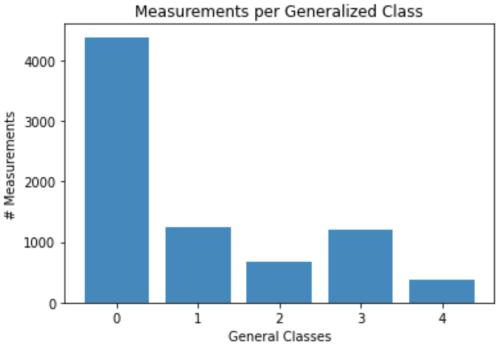
A primeiro momento os dados disponibilizados possuíam catorze categorias, porém devido a discrepância entre a quantidade de dados por categoria os dados foram reorganizados em cinco categorias segundo a tabela II. Exemplos de cada uma das classes podem ser vistos na figura 8. Pode-se ver na figura 3 a distribuição pelas classes originais e pelas classes generalizadas.

Id	Classe original	Classe Generalizada	Novo Id
6	Single micro-lens	Fast	1
15	TDE	Long	2
16	Eclipsing Binary	Periodic	3
42	SNII	S-Like	0
52	SNIax	S-Like	0
53	Mira	Periodic	3
62	SNIbc	S-Like	0
64	Kilonova	Fast	1
65	M-dwarf	Fast	1
67	SNIa-91bg	S-Like	0
88	AGN	Non-Periodic	4
90	SNIa	S-Like	0
92	RR lyrae	Periodic	3
95	SLSN-I	Long	2

Tabela II: Tabela com as quatorze classes originais e suas classes generalizadas.



(a) Distribuição de medidas pelas classes originais.



(b) Distribuição de medidas pelas classes generalizadas.

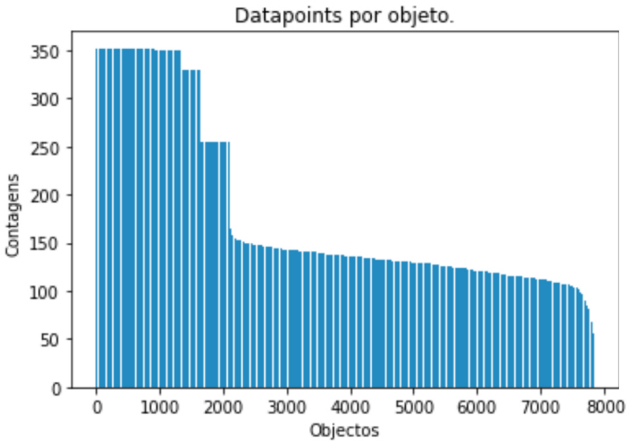
Figura 3

Para realizar o treinamento da rede é preciso garantir que cada objeto possui o mesmo número de medidas, já que essa será a dimensão da entrada. Isso não é verdade para o este conjunto de dados, dessa forma, os dados com menor dimensão foram preenchidos com valores arbitrários, em um processo conhecido como preenchimento. É possível observar na figura 4 que após o preenchimento todos os objetos passam a ter a mesma

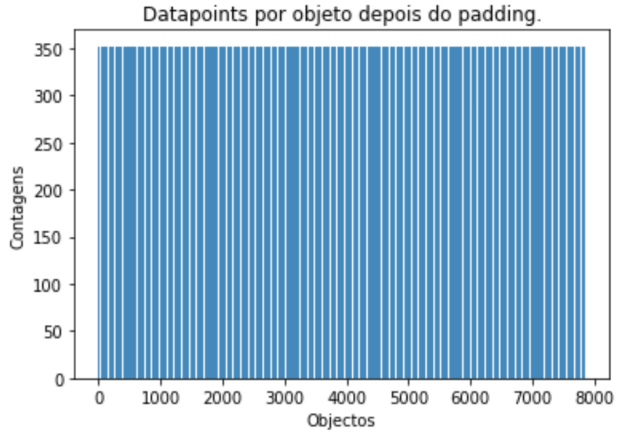
dimensão.

Além disso as medidas temporais foram re-escaladas para que a primeira medida de um objeto fosse considerada como o zero na escala temporal e por último, os dados de fluxo precisaram ser normalizados para cada objeto separadamente visto que a amplitude das medidas entre objetos não é constante.

Por último este projeto está interessado em verificar a capacidade de classificar objetos com apenas dados parciais da curva de luz, dessa forma foi selecionado dois novos conjuntos de teste, o qual os dados foram avançados no tempo em dez e vinte dias de observação a partir do momento em que cada objeto é detectado e posteriormente avaliou-se a rede com os conjuntos de cinco, dez e de vinte dias no futuro.



(a) Distribuição da quantidade de medidas por objeto.



(b) Distribuição da quantidade de medidas por objeto após o preenchimento.

Figura 4

B. Rede Neural

Devido ao fato dos dados disponibilizados serem séries temporais optou-se pelo uso de uma rede LSTM. A rede será bidirecional pois as medidas anteriores são tão relevantes quanto as medidas posteriores. Decidiu-se também que cada uma das colunas no conjunto de dados exemplificado na tabela I será tratada como uma característica do objeto.

Devido ao procedimento de preenchimento todos os objetos passaram a ter 352 medidas e 5 características, isso significa que as dimensões de entrada da rede será de 352x5 e a saída será uma distribuição de probabilidade em um tensor de tamanho cinco, pois temos 5 classes generalizadas.

Em virtude do preenchimento realizado é necessário ter uma camada de máscara de forma que os dados arbitrários sejam identificados e ignorados pela rede. Caso contrário eles seriam tratados com ruído no conjunto de dados.

Será necessário também o uso de uma camada densa para realizar processo de classificação e uma camada de GlobalMaxPooling para reduzir as dimensões do problema. A rede completa pode ser visualizada na figura 5.

III. RESULTADO E DISCUSSÕES

As métricas escolhidas para a avaliação da rede foram as curvas ROC, Precision-Recall e uma matriz de confusão. Também foi verificada a performance da rede com dados parciais de uma curva de luz. Todas estas métricas foram construídas utilizando-se o conjunto de testes.

As curvas ROC obtidas foram na sua maior parte excelentes, obtendo valores para a área abaixo da curva para as classes S-Like e Periodic de 0.95 e 0.99 respectivamente. O mesmo não pode ser dito da classe Long, para essa classe obteve-se o valor de 0.68.

De forma similar os valores alcançados para as curvas Precision-Recall referente as classes S-Like e Periodic, obtiveram altos valores para área abaixo da curva, 0.98 e 0.89 respectivamente. Para a classe Non-Periodic obteve-se o pior resultado de 0.40. Todas as curvas podem ser visualizadas na figura 6.

Ao observar a matriz de confusão, facilita a compreensão do que está ocorrendo. A classificação de curvas S-Like é feita com alta acurácia e o modelo ainda é capaz de distinguir curvas das classes Periodic e Non-Periodic das outras três. Porém, percebe-se que o modelo falha completamente ao tentar classificar curvas do tipo Long ou Fast e também é incapaz de diferenciar curvas Periodic e Non-Periodic entre si. A matriz pode ser visualizada na figura 7 (a).

Como mencionado anteriormente, é necessário avaliar a rede em conjuntos de teste que foram avançados no tempo, neste casso foi avaliado em cinco, dez e vinte dias.

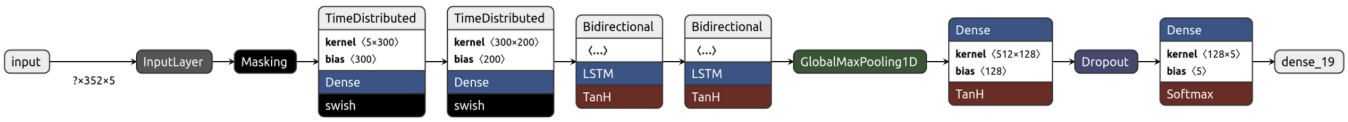
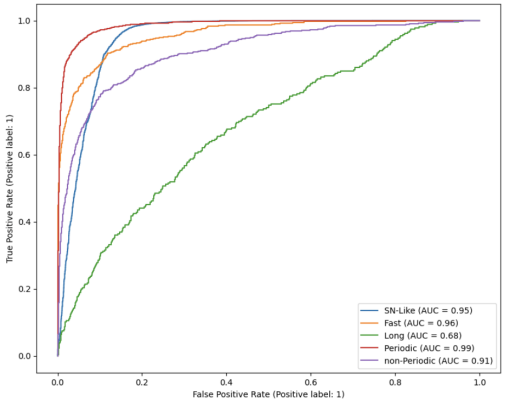
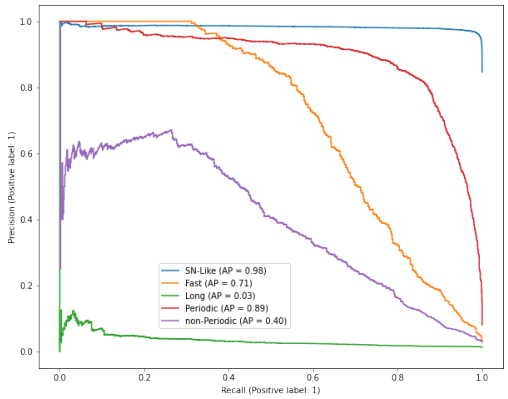


Figura 5: Rede neural utilizada no processo de classificação. A rede LSTM é bidirecional pois precisa-se dar igual importância a dados passados e futuros, a camada de Máscara serve para atenuar os efeitos de ruído que o preenchimento pode causar



(a) Curva ROC para o nosso modelo de classificação. As classes Periodic e S-Like apresentam resultados satisfatórios, porém o mesmo não pode ser dito para as classes Fast e Long.



(b) Curva Precision Recall para o nosso modelo de classificação. As classes Periodic e S-Like apresentam resultados satisfatórios, porém percebemos que o modelo parece não classificar corretamente curvas Non-Periodic.

Figura 6

Assim é possível perceber que a matriz de confusão apresenta piores resultados a medida que o conjunto de testes é avançado no tempo e mesmo com cinco dias de avanço, a rede é incapaz de classificar os objetos com uma performance similar a do conjunto completo e passa a confundir todos os objeto com a classe S-like. Na figura 7 (b), (c) e (d) possui-se as matrizes de confusão para os diferentes conjuntos de teste.

IV. CONCLUSÃO

A rede neural desenvolvida neste projeto, foi capaz de classificar objetos S-Like e ainda distinguir objetos Periodic e Non-Periodics dos demais. Infelizmente esses resultados não se repetem para as classes Fast e Long provavelmente pela difícil caracterização dessas curvas que possuem picos de pouca duração somente no inicio ou no fim da medida.

Outro problema que este modelo apresenta é que não é possível distinguir de forma eficaz objetos Non-Periodic confundindo-os com objetos Periodic. Provavelmente esse resultado vem do fato que ambas as curvas apresentam vários picos ao longo da medida e o modelo parece atribuir periodicidade a medidas que não são periódicas, apesar disso a rede quase não identifica medidas periódicas como não-periódicas.

Por último, infelizmente esta rede perde a capacidade de classificar qualquer objeto quando é fornecido dados parciais das curvas de luz e passa a confundir todos os objetos como se fossem S-Like. Isso deve vir do fato que o conjunto de dados mesmo com as classes generalizadas ainda apresenta uma grande prevalência da classe S-Like. Também é perceptível a confusão da rede para classificar objetos não periódicos, reconhecendo-os como periódicos.

Como sugestão para corrigir esses problemas pode-se pensar em escolher um conjunto de treino que esteja mais equilibrado na sua distribuição de classes. A inclusão de mais objetos do tipo Non-Periodic no conjunto de treinamento da rede pode ser suficiente para reduzir a confusão com a classe Periodic.

Durante a produção deste trabalho, uma das soluções tentadas para minimizar o problema do desbalanceamento entre as classes, foi a utilização de pesos durante o treinamento da rede, porém esta técnica não apresentou uma melhoria que fosse significativa para classificação.

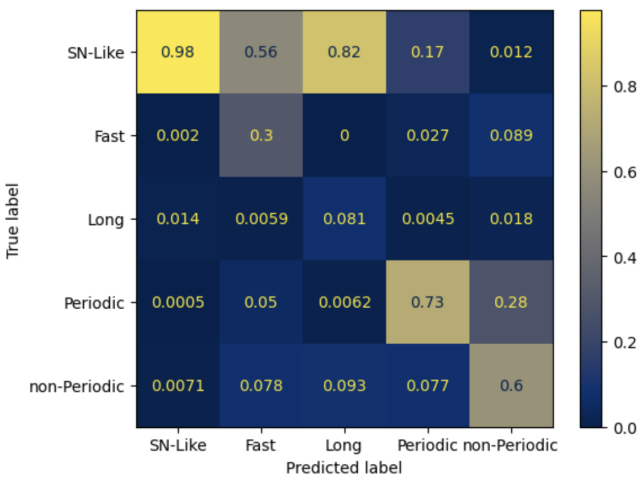
Quanto a resolver a questão de classificação de objetos Fast e Long, talvez seja necessário uma mudança no pré-processamento dos dados. Pode-se por exemplo, extrair dos dados os momentos em que a variável de detecção booleana tem valor 1 e utilizar somente esses pedaços das curvas de luz. Dessa forma estaria removendo dados de fundo do objeto e mantendo somente as medidas correspondentes a detecções. Isso também poderia auxiliar a questão de classificação com dados parciais das curvas.

V. CÓDIGO

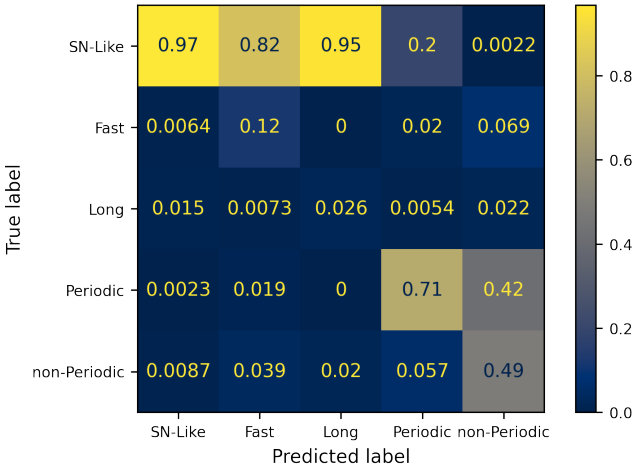
Todo código utilizado e o modelo treinado para reproduzir esses resultados está disponível em um repositório público no GitHub no link `https://github.com/MarcoBarroca/VI_EAFEXP_Proj3`.

ACKNOWLEDGMENTS

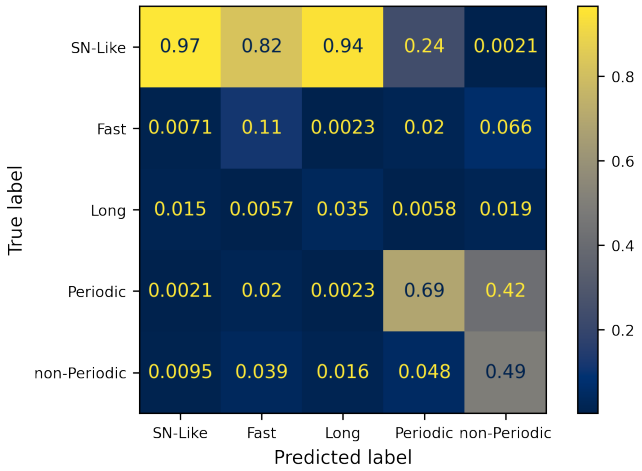
Gostaríamos de agradecer a coordenação da EAFEXP, os professores Clécio R. de Bom, Elisangela L. Faria, Ana Paula O. Muller e Marcelo Portes de Albuquerque e os monitores Gabriel Teixeira e Bernardo M. Fraga pelas palestras, aulas e auxilio no projeto.



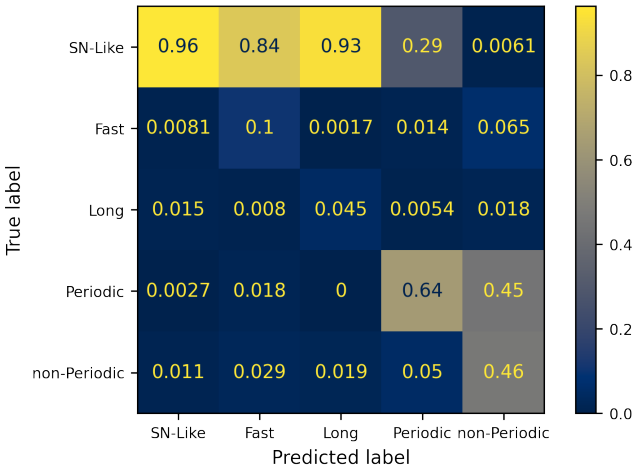
(a) Matriz de confusão para o conjunto de testes completo.



(b) Matriz de confusão para o conjunto de testes com cinco dias a frente. Os resultados pioram em relação ao conjunto de testes completo.

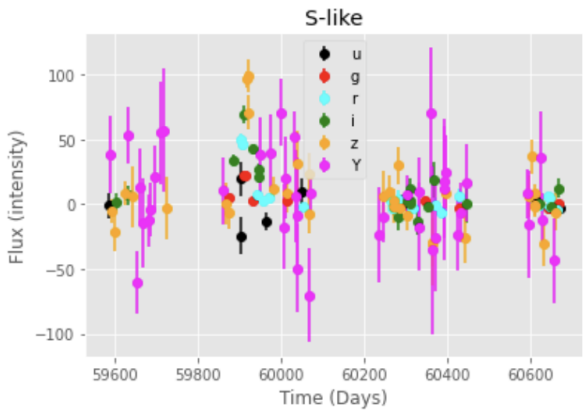


(c) Matriz de confusão para o conjunto de testes com dados dez dias a frente. Os resultados pioram em relação ao conjunto de dez e cinco dias.

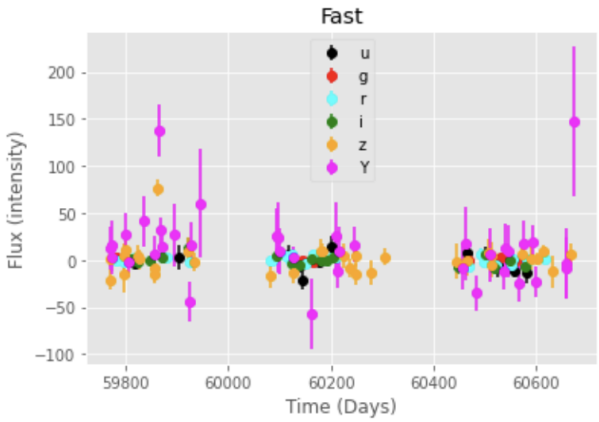


(d) Matriz de confusão para o conjunto de testes com dados vinte dias a frente. A rede agora confunde mais classes com objetos do tipo S-Like.

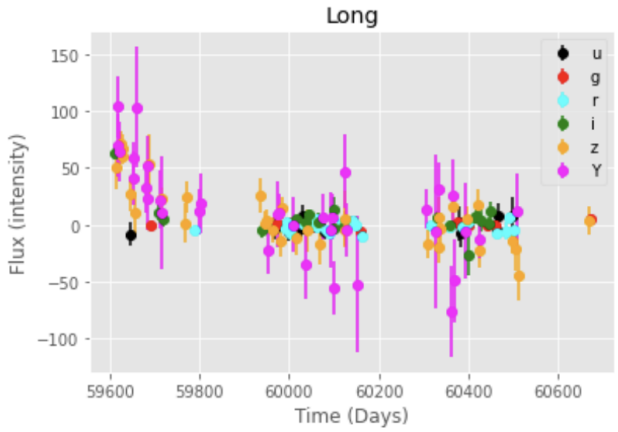
Figura 7



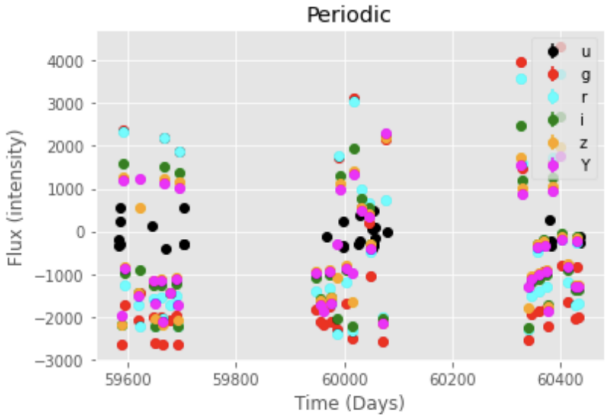
(a) Exemplo da curva de luz de um objeto da classe S-like. Os valores na legenda indicam o filtro utilizado e o espectro da emissão.



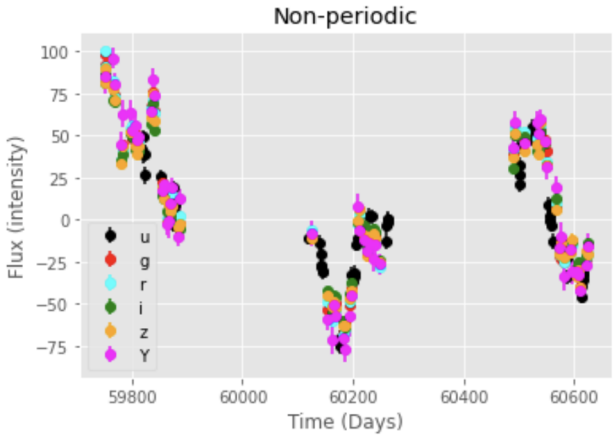
(b) Exemplo da curva de luz de um objeto da classe Fast. Os valores na legenda indicam o filtro utilizado e o espectro da emissão.



(c) Exemplo da curva de luz de um objeto da classe Long. Os valores na legenda indicam o filtro utilizado e o espectro da emissão.



(d) Exemplo da curva de luz de um objeto da classe Periodic. Os valores na legenda indicam o filtro utilizado e o espectro da emissão.

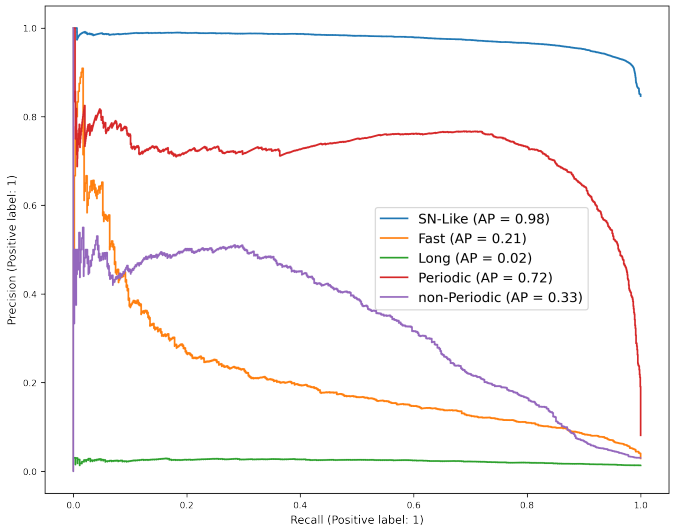


(e) Exemplo da curva de luz de um objeto da classe Non-Periodic. Os valores na legenda indicam o filtro utilizado e o espectro da emissão.

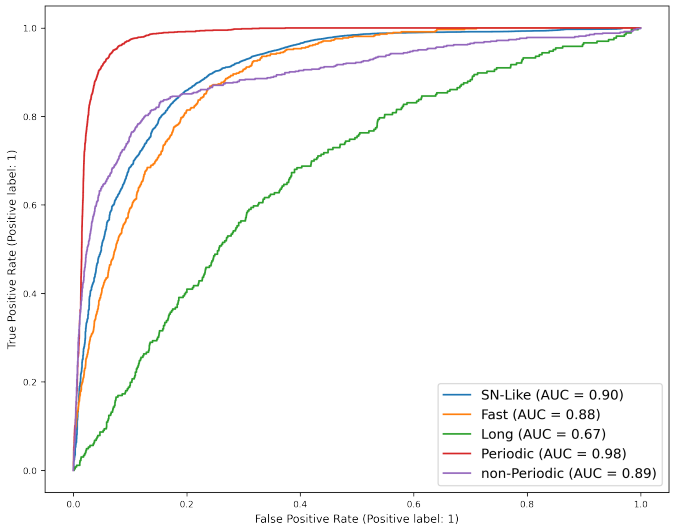
Figura 8

Apêndice A: Curvas ROC e Precision Recall para dados de 5, 10 e 20 dias

Caso seja de interesse do leitor seguem as curvas ROC e Precision Recall nas figuras 9, 10 e 11 para os conjuntos de teste de 5, 10 e 20 dias.

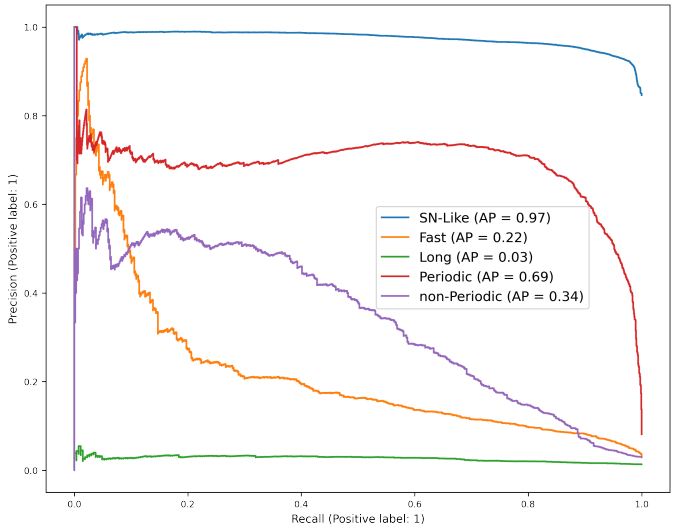


(a) Curva Precision Recall para o conjunto de dados com um avanço temporal de 5 dias.

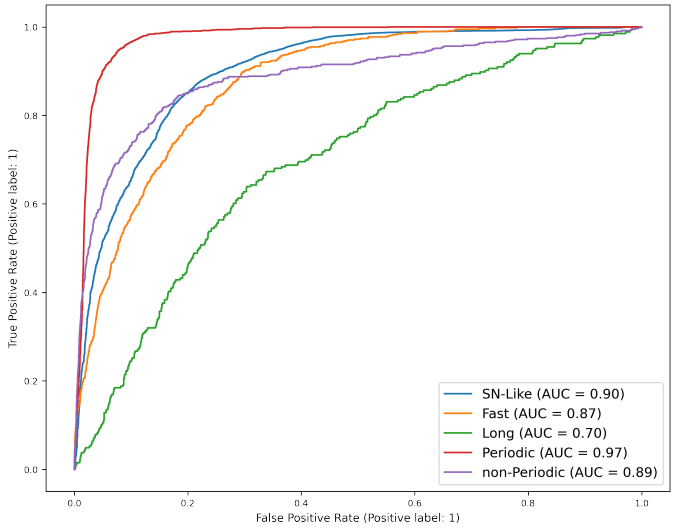


(b) Curva ROC para o conjunto de dados com um avanço temporal de 5 dias.

Figura 9

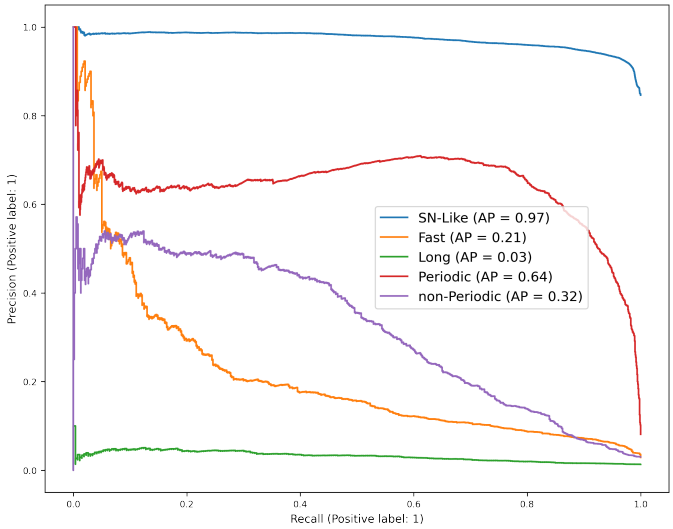


(a) Curva Precision Recall para o conjunto de dados com um avanço temporal de 10 dias.

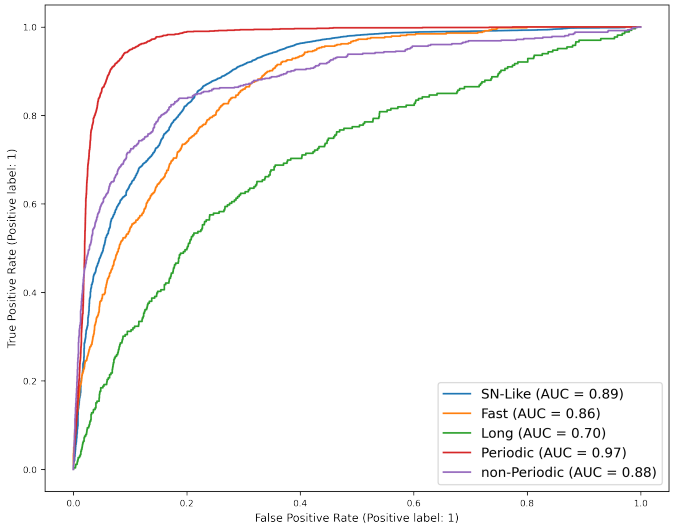


(b) Curva ROC para o conjunto de dados com um avanço temporal de 10 dias.

Figura 10



(a) Curva Precision Recall para o conjunto de dados com um avanço temporal de 20 dias.



(b) Curva ROC para o conjunto de dados com um avanço temporal de 20 dias.

Figura 11

[1] S. G. Djorgovski, A. A. Mahabal, M. J. Graham, K. Polsterer, and A. Krone-Martins, Applications of ai in astronomy (2022).
[2] Plasticc astronomical classification (2018).
[3] H. N. Russell, Astrophys. J. **35**, 315 (1912).
[4] The PLAsTiCC Team, T. Allam, A. Bahmanyar, R. Biswas, M. Dai, L. Galbany, R. Hložek, E. E. O. Ishida, S. W. Jha, D. O. Jones, R. Kessler, M. Lochner, A. A. Mahabal, A. I. Malz, K. S. Mandel, J. R. Martínez-Galarza, J. D. McEwen, D. Muthukrishna, G. Narayan, H. Peiris, C. M. Peters, K. Ponder, C. N. Setzer, T. L. D. E. S. Collaboration, T. L. Transients, and V. S. S. Collaboration, The photometric lsst astronomical time-series classification challenge (plasticc): Data set (2018).
[5] N. R. Tanvir, M. A. Hendry, A. Watkins, S. M. Kanbur, L. N. Berdnikov, and C. C. Ngeow, Monthly Notices of the Royal Astronomical Society **363**, 749 (2005).
[6] G. G. P. R. Santos, *Uso de redes neurais LSTM para predição de séries temporais de demanda de vendas*, Ph.D. thesis, universidade federal de São Paulo, Vitória ES (2022).
[7] Understanding lstm networks (2015).