Métodos Ensemble Online para Classificação Binária

Arthur G. Soares[1], **Thiago P. da Silva**[1]

[1]Universidade Federal de Mato Grosso (UFMT) – Barra do Garças – MT – Brazil

programmeurarthur@hotmail.com, thiago.silva@ufmt.br

Abstract. *In the context of Big Data, analyzing large volumes of data in real-time is crucial for quickly generating insights. Although data classification is challenging, online machine learning is more effective than traditional batch learning methods, especially for continuous changes. This study introduces four ensemble methods for the binary classification of online data streams, using diverse machine learning models. Experiments on eight datasets demonstrated the feasibility and strong potential of these approaches.*

Resumo. *Em Big Data, a análise de grandes volumes de dados em tempo real é crucial para obter insights rapidamente. A classificação de dados é desafiadora, mas o aprendizado de máquina online oferece uma solução mais eficaz para lidar com mudanças contínuas em comparação aos métodos tradicionais de aprendizado em lote. Este trabalho propõe quatro métodos ensemble para classificação binária de fluxos de dados online, combinando modelos heterogêneos de aprendizado de máquina. Experimentos em oito datasets demonstraram a viabilidade e o potencial desses métodos.*

1. Introdução

A evolução da computação, impulsionada pelo aumento da capacidade de processamento e armazenamento, gerou um crescimento exponencial do volume de informações disponíveis [SARKA et al. 2018]. O *Big Data*, com seu grande volume, diversidade e alta velocidade de atualização, traz desafios para as organizações que buscam extrair *insights* e tomar decisões mais precisas. Nesse contexto dinâmico, os fluxos de dados (*data streams*) são fundamentais, fornecendo informações em tempo real que permitem análises rápidas e decisões ágeis.

A classificação de dados é fundamental na análise de *Big Data* e é amplamente aplicada em várias áreas. No entanto, classificar fluxos de dados apresenta desafios, como ruídos, desvio de conceito e desequilíbrio de dados [GOMES et al. 2017]. Métodos de aprendizado em lote (ou *offline*) não são adequados para essa natureza dinâmica, pois exigem muitos recursos para retreinamento [SARKA et al. 2018]. Por outro lado, o aprendizado de máquina online é uma alternativa mais eficiente, pois se adapta continuamente às mudanças, atualizando os modelos com novas observações.

Embora o aprendizado online traga benefícios, é importante equilibrar viés e variância para garantir que o modelo generalize bem [SILVA et al. 2023]. Os métodos *ensemble* são eficazes para fluxos de dados, pois equilibram viés e variância, proporcionando maior precisão e robustez, tornando-os ideais para dados dinâmicos e heterogêneos [PEREIRA and ROSSI 2021].

O principal objetivo deste artigo é apresentar métodos *ensemble* para classificação binária em aprendizado de máquina online, combinando múltiplos classificadores heterogêneos. A pesquisa tem dois focos: avaliar o desempenho dos métodos *ensemble* em relação aos erros de predição e investigar sua eficácia no enfrentamento do desvio de conceito nos dados. O artigo está organizado da seguinte forma: A seção 2 descreve os métodos propostos. A seção 3 aborda os experimentos realizados. A seção 4 apresenta os resultados e, por fim, a seção 5 traz as considerações finais.

2. Métodos Ensemble Online para Classificação Binária

O objetivo de um *ensemble* é melhorar as classificações ao combinar os resultados de diversos modelos base, compensando suas fraquezas e, assim, melhorando o desempenho geral. A Figura 1 ilustra a arquitetura do *ensemble* proposto para o processamento de fluxos de dados. Inicialmente, são criados múltiplos modelos base heterogêneos (*weak models*), que são avaliados continuamente a cada nova observação por métricas de desempenho. A cada nova observação, todos os modelos são atualizados e reavaliados. Em seguida, uma estratégia de combinação é aplicada para definir quais modelos contribuirão para a predição final.

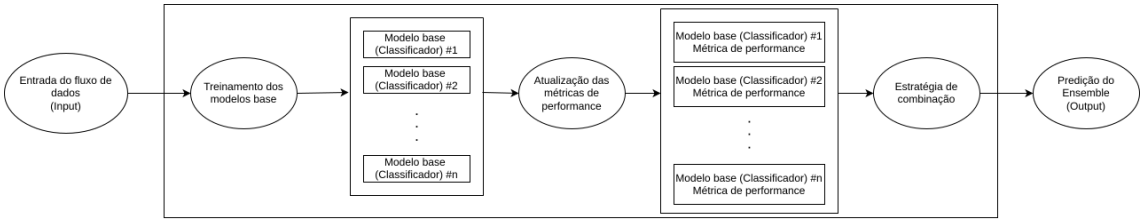

Figura 1. Arquitetura do *ensemble* para processar o fluxo de dados.

2.1. Seleção dos modelos base para composição do ensemble

A seleção dos classificadores como modelos base deve considerar a diversidade [KRAWCZYK et al. 2017], pois isso é crucial para aumentar a precisão do *ensemble*. Foram utilizados 29 algoritmos de aprendizado de máquina da biblioteca *River*[1]. Não foram aplicadas técnicas de otimização dos modelos; em vez disso, foram utilizadas as configurações padrão dos parâmetros. Além disso, foi adotada a mesma *seed* para garantir a determinística dos resultados e a reprodutibilidade dos experimentos.

2.2. Pipeline do Ensemble

O método *ensemble* consiste em um pipeline de três etapas: geração, reset e combinação. A **estratégia de geração** envolve o treinamento dos modelos base do *ensemble*. Na abordagem proposta, todos os modelos base foram construídos a partir dos algoritmos selecionados e treinados continuamente a cada nova observação, garantindo sua independência [SILVA et al. 2023]. Um conjunto de métricas é usado para avaliar o desempenho dos modelos base e do método *ensemble*. Uma **estratégia de reset** híbrida, que combina abordagens ativas e reativas [GOMES et al. 2017], foi adotada para lidar com o desvio de conceito e manter os modelos base atualizados. A **estratégia de combinação** define

[1]Framework River - https://riverml.xyz/dev/

como as predições dos modelos base são integradas para gerar a predição final do *ensemble*. Foram utilizadas quatro estratégias de combinação, que serão apresentadas a seguir.

A **estratégia de combinação Voting**, ou votação majoritária, atribui o mesmo peso a cada classificador base, permitindo que todos votem em um rótulo de classe específico. A classe que receber o maior número de votos é considerada a previsão final do ensemble. Na **estratégia de combinação *Average***, apenas os modelos base que apresentam desempenho superior à média geral de uma métrica específica (e.g., *accuracy*) contribuem para a decisão final do *ensemble*. Em seguida, aplica-se a estratégia de voting para determinar a predição final ensemble.

Na **estratégia de combinação Threshold**, é estabelecido um limiar fixo para uma métrica específica — e.g., *accuracy*. Apenas os modelos base que apresentam desempenho superior a esse limiar são considerados para a decisão final. Em seguida, aplica-se a estratégia de voting para determinar a predição final. Por fim, a **estratégia de combinação Best Model**, envolve o monitoramento contínuo do desempenho dos modelos base, permitindo a seleção do melhor modelo para gerar a previsão do ensemble a cada momento. Esta estratégia é eficaz em cenários com variações significativas no desempenho dos modelos. Em cada nova observação, o melhor modelo é selecionado com base em uma métrica de desempenho (e.g., *accuracy*). Como ilustrado na Figura 2, até o momento $t-1$, o modelo base m_1 foi consistentemente selecionado por apresentar o melhor desempenho. Posteriormente, o modelo m_3 foi escolhido em duas ocasiões consecutivas. Em seguida, m_4 foi selecionado, e, finalmente, m_1 voltou a ser o preferido.

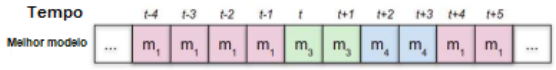

Figura 2. Exemplo da estratégia *best model*.

3. Avaliação

Esta seção apresenta algumas avaliações experimentais realizadas. O trabalho completo, incluindo todas as avaliações, o código-fonte e os dados, está disponível publicamente no GitHub[2]. Neste trabalho, a abordagem GQM foi empregada, proporcionando uma estrutura metodológica sólida para a análise dos dados experimentais. O GQM visa responder a questões específicas e guiar o processo de avaliação, fornecendo *insights* valiosos [BASILI et al. 1998]. A meta definida foi: **Avaliar a eficácia dos métodos *ensemble* e suas capacidades preditivas em diferentes conjuntos de dados**. Com base nessa meta, serão apresentadas três questões: **(Q1)** Os métodos ensembles propostos superam os modelos individuais em termos de erro de predição?; **(Q2)** A capacidade dos métodos *ensemble* de prever corretamente as classes de verdadeiros positivos é superior à dos modelos base em cada conjunto de dados?; **(Q3)** Os métodos *ensemble* apresentam melhor desempenho em comparação aos modelos base em cenários com desvio de conceito?

No contexto do modelo GQM, as métricas são selecionadas de forma criteriosa, sendo associadas a cada pergunta específica formulada. Essas métricas oferecem res-

[2]https://github.com/ProgrammeurArthur/OnlineEnsembleForBinaryClassifier

postas mensuráveis às questões levantadas, permitindo uma avaliação objetiva do desempenho do classificador. A Tabela 1 apresenta as métricas utilizadas para responder às perguntas do GQM sobre os métodos *ensemble* propostos.

Métrica	Descrição	Questão
ROC Curve	Avalia o desempenho do modelo em classificar, mostrando a sensibilidade dele em separar as classes corretamente.	Q1 e Q2
Rolling ROC Curve	Avalia o desempenho através das janelas deslizantes, sendo uma métrica para avaliar desvio de conceito e o comportamento do modelo.	Q1, Q2 e Q3

Tabela 1. Métricas para avaliação dos métodos ensemble propostos.

3.1. Setup para os Experimentos

Os experimentos foram conduzidos na plataforma do Google Colab, na versão pro, utilizando um ambiente de execução baseado em CPU com processador Intel Xeon @ 2.20GHz (Família da CPU: 6), com 51 GB de RAM e 225.8 GB de armazenamento em disco para os testes realizados.

Os métodos *ensemble* propostos utilizaram a métrica *accuracy* na estratégia de combinação. Para garantir a reprodutibilidade dos resultados, a mesma semente foi fixada nos algoritmos com ajuste de *seed*. Os modelos processaram oito *datasets* reais e sintéticos, disponíveis publicamente. O gerador de fluxo *ConceptDrift* da biblioteca *River* foi ajustado para introduzir um desvio de conceito gradual na posição 250.000, com uma largura de transição de 2.000 instâncias. Todos os modelos foram inseridos diretamente nos fluxos de dados, aprendendo continuamente com cada nova observação, sem treinamento prévio.

4. Resultados

Os resultados foram avaliados conforme as métricas definidas para responder às perguntas do GQM. Também foi calculada a média (”Avg”) e o ranqueamento (”Rank”) do desempenho final dos modelos em todos os conjuntos de dados testados. O valor de ”Rank”indica a posição média de desempenho dos modelos para cada métrica em cada conjunto de dados. A Tabela 2 mostra parte dos resultados dos experimentos com a métrica ROC Curve; devido à falta de espaço, nem todos os resultados foram incluídos. Os melhores desempenhos estão em negrito, e as últimas quatro linhas correspondem aos métodos de *ensemble*.

Algoritmos	ConceptDrift	HTTP	CreditCard	SMTP	Elec2	SMSSpam	Bananas	Phishing	Avg ROC Curve	Avg Rank
Ensemble StackingClassifier	**0.99696**	0.99728	0.8564	0.71666	**0.92121**	0.86364	**0.87803**	**0.90474**	**0.89187**	**8.5**
ensembleLeveragingBaggingClassifier	0.99681	0.99683	**0.88099**	**0.74998**	0.84422	0.75156	0.50697	0.8956	0.82787	10.87500
Perceptron	0.99187	**0.99747**	0.87177	0.74979	0.9007	**0.86386**	0.51193	0.85631	0.84296	12.75
treeExtremelyFastDecisionTreeClassifier	0.97926	0.5	0.5	0.5	0.80884	0.5501	0.60134	0.8823	0.66523	24.0
TreeHoeffdingAdaptiveTreeClassifier	0.97635	0.5	0.5	0.5	0.77788	0.54627	0.52061	0.81371	0.64185	26.25
TreeHoeffdingTreeClassifier	0.96226	0.5	0.5	0.5	0.73794	0.54618	0.51708	0.79403	0.63218	27.75
ensembleBestModel	**0.9968**	**0.99841**	0.86092	**0.84998**	**0.92178**	**0.93434**	**0.88716**	**0.91714**	**0.92081**	**3.125**
ensembleBestModelAverage	0.99648	0.99683	**0.86763**	0.66666	0.88196	0.59906	0.85975	0.90461	0.84662	11.0
ensembleBestModelThreshold	0.99633	0.99683	0.85866	0.66666	0.84315	0.60863	0.87146	0.89962	0.84267	12.625
ensembleVoting	0.99616	0.99683	0.85642	0.65	0.82879	0.61445	0.59654	0.90492	0.80551	14.12500

Tabela 2. Recorte dos resultados dos experimentos considerando a métrica *ROC Curve*.

O método *ensemble best model* apresentou o melhor desempenho, liderando tanto o *rank* quanto a média da métrica *ROC Curve*, destacando-se em primeiro lugar em seis

dos oito *datasets*. Nos dois restantes, seu desempenho foi próximo ao do melhor modelo. O modelo base *ensemble StackingClassifier* ficou em segundo lugar no *rank*. O método *ensemble best model Average* ocupou a quarta posição no *rank* e o terceiro lugar na média. O método *best model threshold* ficou em quinto e o *ensemble voting* ficou em nono.

As Figuras 3 e 4 ilustram a evolução da métrica *ROC Curve* ao longo do processamento do *dataset Elec2*. São apresentados os três melhores modelos base considerando o ranqueamento da métrica, além dos modelos base selecionados pelo *ensemble best model* para predição. Conforme as figuras, inicialmente, o *ensemble best model* supera os demais até cerca de 10.000 observações, quando os modelos *ensemble BOLE Classifier*, *Perceptron* e *ensemble Stacking Classifier* se equiparam a ele. Após 15.000 observações, o desempenho do *Perceptron* cai, estabilizando-se em 0.9, enquanto os outros modelos permanecem equilibrados, com o *ensemble best model* mantendo superioridade.

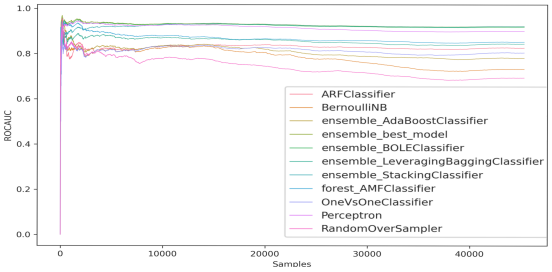

Figura 3. Desempenho dos modelos base e método *ensemble* em termos da métrica *ROC Curve* ao longo do processamento do *dataset* Elec2.

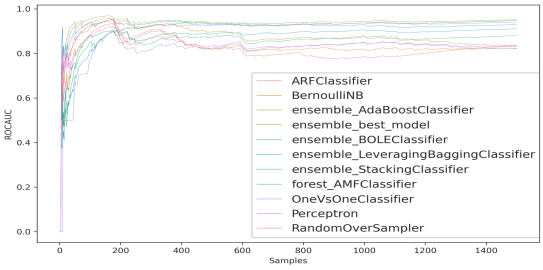

Figura 4. Desempenho dos modelos base e método *ensemble* em termos da métrica *ROC Curve* no início do processamento do *dataset* Elec2.

A Figura 5 ilustra o desempenho do método de *ensemble best model* usando a métrica *rolling ROC Curve* no processamento do *dataset ConceptDrift* durante o desvio de conceito. O método se manteve acima de 0.95 na maioria das observações, tanto antes quanto após o desvio de conceito. Durante o desvio, o desempenho sofreu uma queda, com valores de até 0.8, mas, apesar dessas oscilações, permaneceu alto, com valores superiores a 0.9 na maior parte do tempo. Os outros métodos propostos apresentaram um desempenho praticamente equivalente, entretanto inferior ao *ensemble best model*.

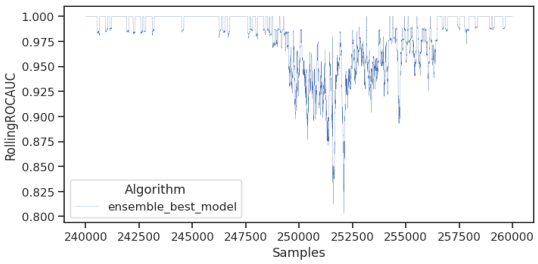

Figura 5. Desempenho do *ensemble best model* durante o desvio de conceito no *dataset* ConceptDrift.

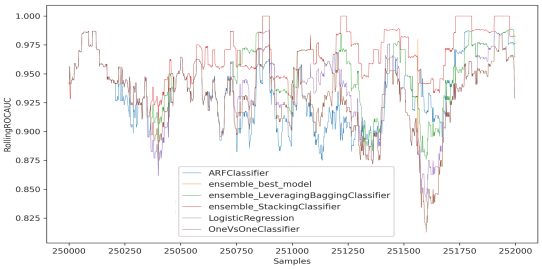

Figura 6. Seleção dos melhores modelo base pelo *ensemble best model* no *dataset* ConceptDrift.

Para responder à questão **Q1** do GQM, foram analisadas as métricas *ROC Curve* e *Rolling ROC Curve*. O método *ensemble best model* apresentou o melhor desempenho, liderando o ranking da métrica *ROC Curve* com a maior precisão, enquanto o *ensemble best model average* ocupou a terceira posição. Os demais métodos tiveram desempenhos moderados. Para responder à questão **Q2**, a métrica *ROC Curve* revelou que o método

ensemble best model apresentou a melhor sensibilidade na identificação das classes de verdadeiro positivo, ocupando a primeira posição. Ao analisar a métrica *rolling ROC Curve*, os métodos *ensemble* demonstraram desempenho consistente, permanecendo entre os melhores. No entanto, em algumas observações específicas do *dataset ConceptDrift*, certos modelos base superaram o desempenho dos métodos *ensemble*. Para responder à questão **Q3**, a análise da métrica *rolling ROC Curve* mostrou que os métodos *ensemble* propostos mantiveram um desempenho superior a 0.9 na maioria das observações. Durante o desvio de conceito, os métodos *ensemble* apresentaram desempenho semelhante, mas em algumas observações foram superados por modelos base. Isso ocorreu porque a métrica de combinação utilizada foi acumulativa da *accuracy*, o que não favoreceu uma adaptação mais eficiente ao desvio de conceito.

5. Conclusão

Este trabalho explora a viabilidade de quatro métodos *ensemble* para a classificação binária de fluxos de dados online, por meio de experimentos realizados em oito *datasets*. Os métodos combinam diferentes modelos base heterogêneos de aprendizado. Com base nas métricas selecionadas para responder ao GQM, os métodos *ensemble best model* e *ensemble best model average* se destacaram, figurando entre os três melhores modelos. Em contrapartida, os métodos *ensemble best model threshold* e *ensemble voting* apresentaram desempenho mais modesto.

Os experimentos consideraram apenas uma única métrica (*ROC Curve*) para aferir os métodos *ensemble*. Desta forma, propõe-se investigar quais métricas melhoram o desempenho dos *ensemble* para cada *dataset*, utilizando a estratégia *rolling* para lidar com desvio de conceito. Além disso, pretende-se identificar e remover modelos base com baixa contribuição para predição ensemble. A execução sequencial adotada nos *ensemble* será otimizada para uma execução paralela e distribuída, visando sua aplicação em dispositivos IoT.

Referências

BASILI, V. R., CALDIERA, G., and RAMBACH, H. D. (1998). **The goal question metric approach.** *Encyclopedia of software engineering.*, pages 528–532.

GOMES, H. M., BARDDAL, J. P., ENEMBREACK, F., and BIFET, A. (2017). **A Survey on Ensemble Learning for Data Stream Classification.** *Association for Computing Machinery*, pages 1–10.

KRAWCZYK, B., MINKU, L. L., GAMA, J., STEFANOWSKI, J., and WOZNIAK, M. (2017). **Ensemble learning for data stream analysis: A survey.** *Information Fusion*, pages 2–3, 133.

PEREIRA, L. F. and ROSSI, R. G. (2021). **Estudo de técnicas de Ensemble para classificação binária.** *Universidade Federal de Mato Grosso do Sul*, pages 1–2, 17.

SARKA, D., BALI, R., and SHARMA, T. (2018.). ***Practical Machine Learning with Python.*** ed. Apress.

SILVA, T. P., BATISTA, T. V., DELICATO, F. C., and PIRES, P. F. (2023). **An Online Ensemble Method for Auto-Scaling NFV-based Applications in the Edge.** *Cluster Computing*, pages 2–13.