Modelo para previsão precoce de abandono de uma disciplina de introdução à programação

João Pedro Freire[1], Flávia M. P. F. Landim[1], Laura O. Moraes[2], Carla A. D. M. Delgado[1], Carlos Eduardo Pedreira[1]

[1] Universidade Federal do Rio de Janeiro (UFRJ)

[2]Universidade Federal do Estado do Rio de Janeiro (UNIRIO)

joaofreire@dme.ufrj.br, flavia@im.ufrj.br, laura@uniriotec.br

carla@ic.ufrj.br, pedreira56@gmail.com

Abstract. *Learning programming is essential for students in various careers. In this article, we used statistical models to predict dropout from introductory programming courses and identify relevant variables in the early identification of students at risk. To build the model, we combined statistical inference with machine learning techniques to achieve both interpretability and performance. The predictions achieved an AUC greater than 0.8 in the weekly models starting from the fourth week of classes, enabling an early warning for instructors. Among the variables involved, it was observed that consistency in solving exercises has a greater influence than the time taken to develop the solution in identifying students with potential dropout risk.*

Resumo. *A aprendizagem de programação é essencial para alunos de diversas carreiras. Neste artigo usamos modelos estatísticos para prever o abandono de disciplinas introdutórias de programação e apontar variáveis relevantes na identificação precoce de alunos em risco. Para construção do modelo combinamos a inferência estatística com técnicas de aprendizado de máquina visando alcançar interpretabilidade e desempenho. As previsões obtiveram um AUC maior que 0.8 nos modelos semanais a partir da quarta semana de aula, viabilizando um alerta precoce a professores. Entre as variáveis envolvidas, percebeu-se que a constância na resolução dos exercícios possui uma influência maior do que o tempo demorado na elaboração da solução na identificação dos alunos com potencial de abandono.*

1. Introdução

A aprendizagem da programação é reconhecida como desafiadora, levando a esforços para compreendê-la melhor [Qian and Lehman 2017]. Em salas de aula presenciais, os professores diariamente realizam uma espécie de "mineração de dados educacionais", observando o comportamento dos alunos e suas reações emocionais e de desempenho nas atividades. Isso os ajuda a adaptar suas estratégias de ensino. No entanto, em turmas grandes ou no ensino remoto, a observação direta é difícil, destacando a necessidade de ferramentas de apoio. Na busca contínua por aprimorar a qualidade do ensino, a tentativa de previsão de possíveis desistências ou desempenho ruim por parte dos alunos tem um papel crucial, pois proporcionam aos professores tempo adequado para planejar e implementar intervenções.

O principal objetivo deste artigo é antecipar o abandono e trancamento da disciplina de estudantes em cursos introdutórios de programação. Ao prever o abandono, adquirimos a capacidade de adotar medidas proativas, proporcionando o suporte e as intervenções necessárias de maneira oportuna. A implementação dessas ações pode não apenas reduzir significativamente a taxa de desistência, mas também aprimorar a retenção dos alunos, contribuindo assim para uma experiência acadêmica mais bem-sucedida. Para isso, adotou-se uma abordagem que combina a inferência estatística, visando interpretabilidade das variáveis envolvidas, e técnicas avançadas de aprendizado de máquina para alcançar o melhor desempenho dos modelos. Além disso, nosso projeto visa construir modelos que possam ser aproveitados pelos professores, auxiliando na identificação precoce de fatores indicativos de abandono. Como objetivos específicos, pode-se enumerar as seguintes questões:

1. É possível prever quais alunos não entregarão os exercícios finais do curso a partir de sua interação com uma plataforma online?
2. Quais comportamentos (nesse caso, comportamentos online na plataforma) dos alunos são bons preditores da não finalização dos exercícios finais?

Por utilizar dados gerados por alunos, esse experimento com a coleta de dados e a criação dos modelos de previsão e abandono foi submetido e aprovado pelo comitê de ética e possui o CAAE 60422822.7.0000.5285.

2. Trabalhos Relacionados

Esta seção explora as abordagens e modelos para prever o desempenho dos estudantes e o abandono no ambiente educacional. Analisamos as estratégias adotadas para coletar dados e como esses dados são utilizados para fornecer informações aos professores.

Primeiramente, destacamos o estudo de [Burgos et al. 2018], que adota a regressão logística para a construção de classificadores para o abandono, direcionando o foco para a análise de notas históricas dos alunos e o cronograma de ensino e avaliando as previsões ao longo das semanas de aula. Já [Agrusti et al. 2020] utilizaram uma abordagem de aprendizado profundo com redes neurais convolucionais, empregando dados administrativos para prever o abandono universitário. [Al-Shabandar et al. 2018] exploram características geográficas e comportamentais como elementos cruciais para prever a conclusão de cursos online, destacando a importância de cliques e do número de dias de interação com o sistema para prever a desistência do curso. Em [Chen et al. 2019], o estudo se dedica a cursos de curta duração e também utiliza características de aprendizado baseadas em comportamentos, com ênfase na detecção precoce e na evolução do poder preditivo ao longo do tempo. O trabalho de [Bravo-Agapito et al. 2021] oferece uma variedade de modelos estatísticos para prever o desempenho de estudantes de graduação em cursos online. Os resultados indicaram quatro fatores de maior contribuição para a previsão: acesso, questionários, tarefas e idade, sendo a idade um preditor negativo do desempenho. No caso de [Marbouti et al. 2016], também são comparados vários métodos de previsão para identificar alunos em situação de risco, onde é utilizado um método de seleção de variáveis, buscando aumentar a generalização dos modelos.

Estudos anteriores geralmente se concentram na construção de preditores para a avaliação do desempenho do aluno após o término de um curso e negligenciam o valor

prático de um sistema de "alerta precoce" para prever alunos em situação de risco enquanto um curso está em andamento. Em contraste, [Hu et al. 2014] introduz um sistema de alerta precoce com base em comportamentos registrados em sistemas de gerenciamento de aprendizado a partir de técnicas de mineração de dados e aprendizado de máquina.

O diferencial deste trabalho em relação a esses estudos reside na combinação de métodos de modelagem estatística e aprendizado de máquina, juntamente com a implementação de treinamentos semanais e atualizações semestrais dos preditores. Portanto, o diferencial crucial reside na abordagem adaptativa, que visa melhorar continuamente as previsões ao longo do curso, mas obtendo resultados satisfatórios já a partir da quarta semana de aula.

3. Fundamentação

Esta seção apresenta a metodologia de ensino adotada na disciplina e que norteia as atividades realizadas na plataforma virtual de aprendizagem Machine Teaching [Moraes et al. 2022], utilizada para a resolução dos exercícios e coleta dos dados. Por fim, são apresentados os métodos e métricas estatísticas utilizadas na criação dos modelos de previsão.

3.1. Organização da disciplina de introdução à programação

O curso de Computação 1 oferecido pelo Instituto de Computação da Universidade Federal do Rio de Janeiro visa desenvolver habilidades para criar programas em Python legíveis e modulares. São ofertadas semestralmente cerca de 15 turmas da disciplina, com 40 alunos por turma aproximadamente, contribuindo para a formação de engenheiros, matemáticos, meteorologistas, químicos, físicos, astrônomos, nanotecnólogos e licenciados em áreas das ciências exatas.

A proposta didática estabelecida no curso enfoca em construir módulos de código concisos, deixando os mecanismos de interação do usuário para o final do curso (quando o estudante já dominou os conceitos básicos) e proporcionando ao estudante uma orientação para as tarefas cognitivas mais abstratas de construção de programas desde o início do aprendizado [Delgado et al. 2016]. O programa de estudos é dividido em 11 semanas que abordam diferentes assuntos, conforme indicado no site oficial da disciplina[1]. Neste trabalho, nos referimos a cada assunto semanal como "semana" ou "aula".

3.2. Machine Teaching

O Machine Teaching [Moraes et al. 2022] é uma plataforma de aprendizado online usada desde 2018, tendo sido utilizado por mais de 140 turmas, 3900 alunos e 45 professores como principal recurso de apoio às aulas práticas, presenciais ou remotas, dos cursos introdutórios de programação oferecidos pelo Instituto de Computação da Universidade Federal do Rio de Janeiro (IC/UFRJ). O sistema soma mais de 560 mil interações em sua base de dados nos últimos 5 anos. Seu principal objetivo é coletar dados sobre o conhecimento dos alunos durante o processo de aprendizagem em programação para ajudar na tomada de decisões relacionadas ao curso e ao aprendizado de programação.

Para resolver um exercício, o aluno deve escrever o código em uma IDE (ambiente de desenvolvimento de código) disponibilizada dentro da plataforma, que inclui

[1] https://python.ic.ufrj.br/

uma interface de edição e submissão do código e *feedbacks* das soluções. As funcionalidades do Machine Teaching já foram apresentadas em detalhes em artigos anteriores [Moraes et al. 2022, Xará et al. 2023].

3.3. Modelos estatísticos e métricas

A análise de regressão é uma técnica estatística amplamente utilizada em diversas áreas para investigar e modelar a relação entre variáveis [Montgomery et al. 2013]. Esses modelos permitem prever valores esperados de uma variável dependente (resposta) com base nos valores de uma ou mais variáveis independentes (explicativas, regressoras ou preditoras). No contexto de aprendizado de máquina, esse tipo de modelagem é chamado de aprendizado supervisionado [James et al. 2021].

Na regressão linear usual, assume-se que a variável resposta segue uma distribuição Normal e é modelada diretamente como uma combinação das variáveis preditoras. No entanto, em situações em que a resposta é binária e que, portanto, não podemos assumir a normalidade dos dados, como o abandono ou não de um aluno em uma disciplina, pode-se utilizar uma extensão importante dos modelos lineares tradicionais: os modelos lineares generalizados (MLG). Esses modelos se adaptam a diversos tipos de dados, tais como proporções, contagens e binomiais, ampliando a capacidade de modelagem estatística [Dobson and Barnett 2008]. Neste caso, devido à natureza binária da variável resposta, é apropriado o uso da regressão logística para o classificador [James et al. 2021].

Ao utilizar uma regressão logística, ao invés de modelarmos a variável resposta diretamente, modelamos a probabilidade de que a variável resposta seja de uma determinada categoria (neste caso, da categoria abandono). Tal relação pode ser escrita como:

$$Pr(Y = abandono|X) = p(X) = \frac{e^{\beta_0+\beta_1 X}}{1 + e^{\beta_0+\beta_1 X}} \tag{1}$$

onde β representa os coeficientes da função a serem estimados e X as variáveis preditoras. Após alguma manipulação, temos que a função logística é dada por

$$\log\left(\frac{p(X)}{1 - p(X)}\right) = \beta_0 + \beta_1 X \tag{2}$$

e tem uma interpretação natural como o logaritmo da razão entre a probabilidade de um evento ocorrer e a probabilidade de não ocorrer.

Nos modelos lineares generalizados, é comum a utilização da métrica *Deviance* [Dobson and Barnett 2008] para a comparação de modelos. A *Deviance* está para os modelos lineares generalizados como a soma dos quadrados dos resíduos está para o método os modelos lineares tradicionais. Ela é uma forma de quantificar o quão bem um modelo se ajusta aos dados observados. Modelos com *Deviance* mais baixa são considerados mais adequados para descrever os dados observados.

Uma forma de analisar o desempenho das classificações de modelos é usando a curva ROC (*Receiver Operating Characteristic*), uma representação gráfica que permite avaliar o desempenho de um classificador binário em diferentes valores de limiar de probabilidade. A curva ROC é construída plotando a taxa de verdadeiros positivos

(Sensibilidade) no eixo vertical e o complementar da taxa de falsos positivos (1 - Especificidade) no eixo horizontal. Cada ponto na curva ROC representa um determinado limiar de classificação aplicado ao modelo. A curva ROC permite a comparação visual da capacidade preditiva de diferentes modelos ajustados. Curvas com maior área abaixo dela indicam melhor desempenho do modelo, o valor da área nos leva a mais uma métrica, conhecida por AUC (*Area Under the Curve*), que reflete um desempenho mais elevado à medida que se aproxima do valor 1 [James et al. 2021].

4. Metodologia

O modelo desenvolvido visa identificar os alunos que estão em risco de abandonar a disciplina de introdução à programação. Esta seção detalhará a metodologia empregada para desenvolver o modelo de previsão de abandono dos alunos, incluindo as variáveis, técnicas e estratégias utilizadas para atingir esse objetivo.

4.1. Base de dados e análise exploratória

Os dados disponíveis permitem-nos estudar e compreender quais informações podem ser relevantes para predizer o abandono do curso. Em primeiro lugar, é necessário definir o conceito de abandono de um aluno dentro do contexto da disciplina. A Figura 1 apresenta a proporção de alunos únicos que acessaram o Machine Teaching a cada semana em cada semestre em relação à quantidade inicial de alunos ativos no sistema no semestre (porcentagem de alunos ativos a cada semana). Excetuando-se o período de 2020/1[2], os gráficos apresentam o mesmo comportamento: a proporção de alunos ativos diminui para menos que 70% após a primeira metade do curso, ou seja, após as primeiras avaliações. Diante desse panorama, podemos utilizar a atividade dos alunos nas últimas semanas de utilização do sistema como variável resposta no modelo de previsão de abandono. Neste caso, consideraremos a participação nas duas últimas aulas do sistema como variável resposta. Vale ressaltar que o sistema é utilizado durante as semanas 2 a 9. Logo, as duas últimas aulas nesse contexto referem-se às aulas 8 e 9.

As variáveis que utilizaremos como indicadores do abandono dos alunos na disciplina são as quantidades de exercícios resolvidos pelos alunos nas duas últimas semanas de aula por meio da plataforma. Para exemplificar, a Figura 2 apresenta as distribuições da proporção de exercícios concluídos com sucesso pelos alunos na aula 8. O mesmo comportamento acontece na aula 9. Nosso foco está em compreender o engajamento dos alunos nesse período crítico. Nessa Figura é possível notar que a proporção de exercícios entregues estão mais concentradas em 0 e 100% das listas, lembrando uma distribuição de Bernoulli. Esse comportamento pode ser transformado em um problema binário, em que o aluno entrega ou não a lista de exercícios. Divide-se então os alunos que entregam acima e abaixo de 50% para as aulas 8 e 9.

Desse modo, definiremos abandono da disciplina nesse contexto se um aluno não entrega nenhuma das listas 8 e 9. Se o aluno entrega pelo menos uma das listas, esse aluno pode ser considerado ainda ativo ao final do curso. Com essa definição, podemos modelar a variável resposta do problema. Sua distribuição é exibida na Figura 3, que mostra a proporção de alunos considerados como abandono e não-abandono para cada período.

[2]Este período foi contabilizado de maneira diferente, pois houve um período especial chamado de Período Letivo Especial (PLE) que posteriormente foi adicionado como parte do período de 2020/1

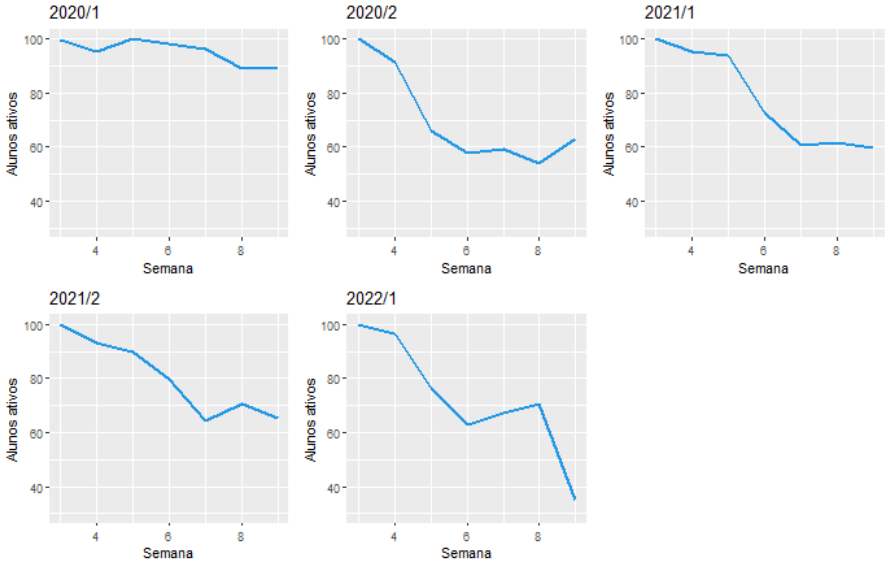

Figura 1. Evolução do número de alunos ativos no Machine Teaching ao longo das semanas por período

Percebe-se que em todos os períodos a classe da variável de abandono não possui um desbalanceamento forte.

Terminada a definição do abandono no contexto desse trabalho, buscaremos indícios de variáveis que podem prever esse abandono ao longo do curso. Consideramos as variáveis por aula até a aula anterior às duas finais, que serão úteis para a previsão da atividade final dos alunos. Ao todo, são 42 variáveis candidatas (7 variáveis x 6 aulas) a entrar no modelo final. Os dados coletados no Machine Teaching medem a atividade, tempo e o sucesso dos alunos ao solucionarem os problemas. São eles:

1. Tentativas: Número total de tentativas do aluno até resolver os problemas com sucesso na semana
2. Quantidade de sucessos: Número de exercícios resolvidos com sucesso na semana, dentro ou fora do prazo de entrega
3. Taxa de sucesso: Porcentagem dos exercícios resolvidos com sucesso na semana, dentro ou fora do prazo de entrega
4. Tempo médio de solução: Tempo médio de escrita de código do aluno até enviar a solução
5. Tempo médio no problema: Tempo médio do aluno na página do problema até enviar a solução
6. Problemas resolvidos no prazo: Número de problemas solucionados com sucesso, dentro do prazo de entrega
7. Frequência: Número de dias que o aluno acessa a lista de exercícios na semana

4.2. Construção do modelo

Para o treinamento dos modelos, todas as variáveis foram incluídas e foi utilizada validação cruzada com penalização Lasso no conjunto de treinamento, que mantém as variáveis explicativas mais relevantes nos modelos finais. Na validação cruzada é estimada a métrica que temos interesse em otimizar, neste caso a *Deviance*, para diferentes

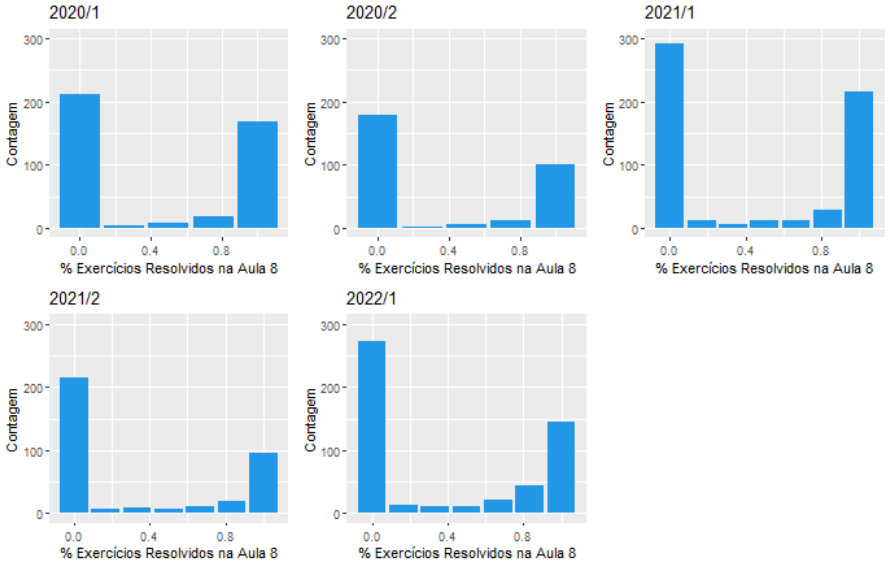

Figura 2. Frequências das proporções de exercícios resolvidos na semana 8 por período

valores de λ. Após a avaliação do modelo pela *Deviance* estimada, constrói-se um classificador a partir da curva ROC gerada, definindo-se o ponto de corte de classificação. No fim, o classificador é avaliado no conjunto de teste.

Para fins de validação, os modelos foram treinados utilizando como conjunto de teste o período que temos interesse em prever e utilizando as observações de períodos anteriores para treinamento. Isso foi feito para quatro períodos, a partir de 2020/2. Os modelos são treinados semana a semana com os dados disponíveis somente até a aula anterior à da semana corrente.

5. Resultados e discussão

Esta seção se propõe a responder e discutir as perguntas de pesquisa propostas na Seção 1.

5.1. É possível prever quais alunos não entregarão os exercícios finais do curso a partir de sua interação com uma plataforma online?

Para cada semana foi criado um modelo utilizando os dados disponíveis até aquela semana para prever precocemente o abandono dos alunos. Dessa maneira, é possível dar um aviso antecipado aos professores de alunos em risco de abandono. Cada modelo por semana foi avaliado pelo seu desempenho na curva ROC e de maneira relativa utilizando a *Deviance*.

A curva ROC permite a avaliação dos modelos na classificação do abandono dos alunos. Tais curvas são apresentadas na Figura 4. As áreas abaixo das curvas são, em geral, maiores em aulas mais próximas do final do curso. Os modelos criados para prever o abandono em 2020/2 e 2021/1, possuem comportamentos parecidos. Em ambos, a partir da Semana 4 (com os dados disponíveis somente das aulas das semanas 3 e 4) é possível obter uma AUC de mais de 0.8, indicando uma boa separação do modelo entre alunos que abandonaram e não abandonaram a disciplina. Em 2021/2, esse valor diminui para 0.74, ficando abaixo dos anteriores. Esses resultados indicam que com a metodologia proposta

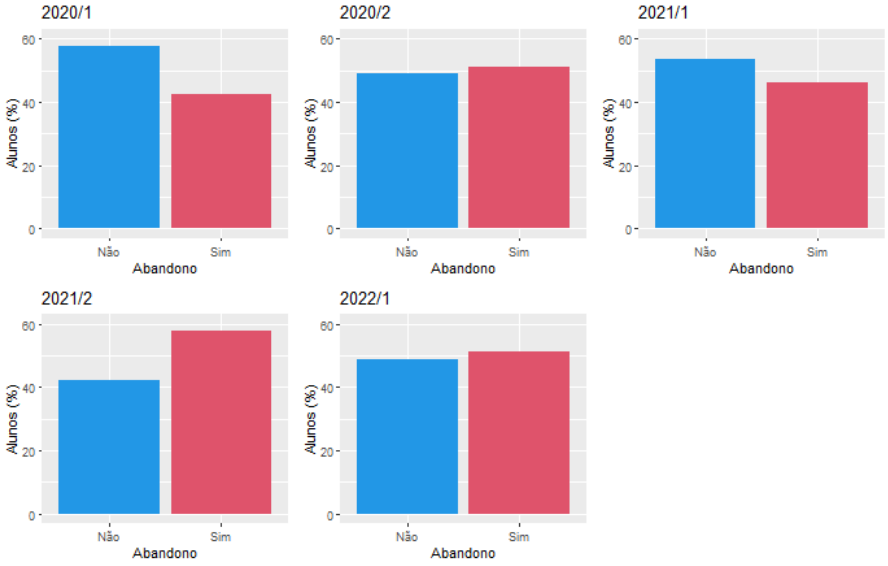

Figura 3. Proporções de alunos que abandonaram e não abandonaram o curso por período

e utilizando dados de treino de um período próximo do que se deseja prever é possível criar modelos para oferecer um aviso precoce aos professores sobre potenciais abandonos já na quarta semana de aula.

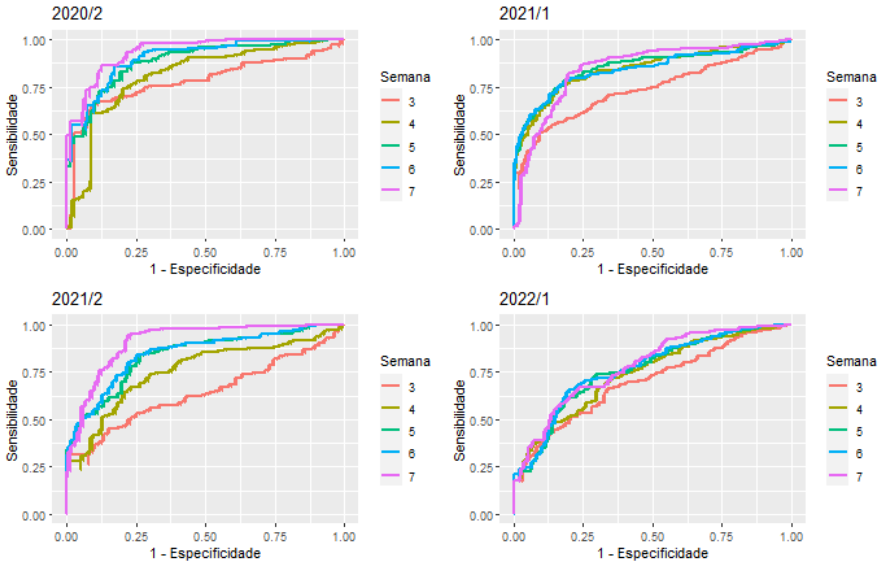

Figura 4. Comparações das curvas ROC dos modelos semanais por período

O valor da *Deviance* dos modelos semanais resultantes podem ser observados na Figura 5, que exibe gráficos da evolução da *Deviance* conforme o valor de lambda varia na verossimilhança da penalização Lasso. Percebe-se que os modelos de aulas mais próximas do final do curso possuem melhor desempenho do ponto de vista da *Deviance*. Isto é, quanto mais próximo do final do curso, mais explicativos ficam os modelos. Essa melhoria é esperada, pois os modelos finais possuem variáveis explicativas de aulas mais

próximas das últimas aulas, que queremos prever. Portanto, ao longo das semanas, é esperado que tenhamos respostas mais precisas sobre o abandono dos alunos.

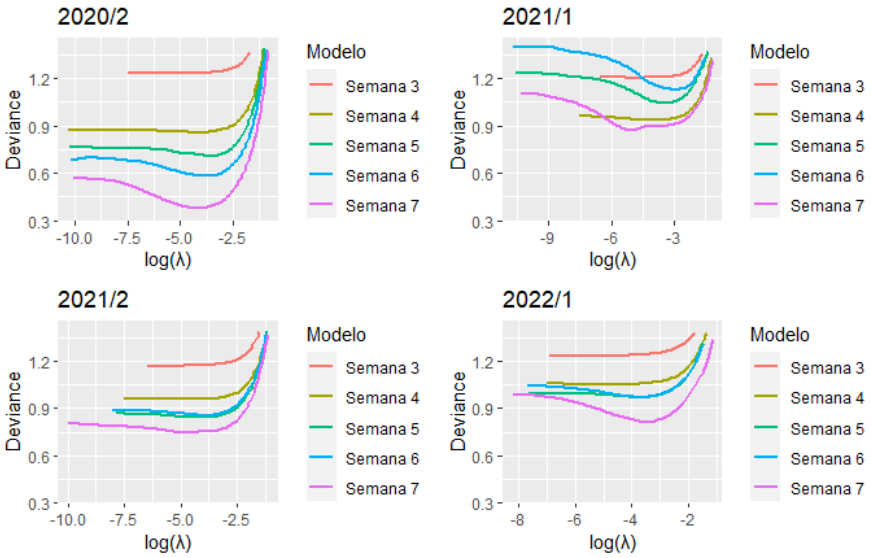

Figura 5. Comparação entre os modelos semanais da estatística *Deviance* para diferentes valores de λ por período

Mas, assim como na curva ROC, os modelos perdem desempenho conforme a diferença de tempo entre os dados de treino e a previsão aumentam. Ou seja, ao longo dos períodos o desempenho dos modelos pioram ao utilizar dados acumulados. Apesar da tentativa de generalizar o conjunto de treinamento, as diferenças de metodologia e perfis de alunos entre períodos podem ter contribuído para perda de desempenho. Percebe-se no último período a maior discrepância de desempenho com relação aos demais. Isso provavelmente se deve a mudança de aulas remotas para presenciais, em que a metodologia do curso teve mudanças significativas. O perfil de interação com a plataforma mudou desde a coleta inicial dos dados (durante a pandemia, utilizando um modelo de sala de aula invertido) até o último período analisado, com a volta às aulas presenciais e aulas teóricas tradicionais, indicando a necessidade de atualização constante dos modelos com dados recentes.

5.2. Quais comportamentos (nesse caso, comportamentos online na plataforma) dos alunos são bons preditores da não finalização dos exercícios finais?

Na Figura 6, vemos para cada semana os *boxplots* das contagens de sucessos entre os alunos que abandonaram e que continuaram ativos, levando em consideração todos os períodos. Em geral, alunos que não abandonam o curso ao final concluem mais exercícios com sucesso. Já é notável essa diferença nas primeiras semanas de aula, reforçando o resultado encontrado na Seção 5.1 de que é possível alertar professores precocemente na quarta semana de aula.

Para as demais variáveis numéricas, gráficos semelhantes foram criados e analisados. Outras variáveis numéricas que explicam bem a variável resposta são relacionadas aos exercícios entregues dentro do prazo, quantidade de tentativas e frequências. O tempo médio da solução e tempo médio do problema não foram encontrados como variáveis

com potencial explicativo. Ou seja, o tempo demorado por um aluno não parece ser um indicativo de abandono, mas sim a constância na entrega dos exercícios. Intervenções que incentivem esse comportamento, como *hackatons*, *Coding Dojo* e gamificação podem ser exploradas para diminuir o abandono da disciplina.

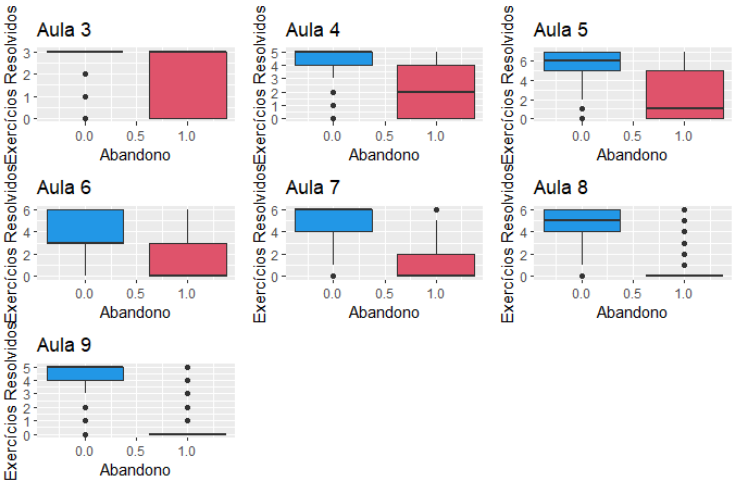

Figura 6. Boxplots da quantidade de sucessos por aula para cada categoria

6. Conclusão

Neste trabalho, exploramos a aplicação de técnicas de aprendizado de máquina para prever os resultados de alunos em um curso introdutório de programação. O foco foi no desenvolvimento de modelos preditivos para identificar alunos em risco de abandonar a disciplina, utilizando dados provenientes de um sistema online de aprendizado ao longo do semestre. Os modelos de previsão de abandono apresentaram desempenhos satisfatórios, obtendo um AUC maior que 0.8 nos modelos semanais a partir da aula 4, viabilizando um alerta precoce a professores. As variáveis analisadas revelaram que a entrega constante de exercícios possuem maior influência que o tempo demorado para sua realização. Intervenções que explorem esse aspecto podem ser utilizadas para diminuir o abandono na disciplina.

Há diversas direções a explorar em trabalhos futuros. Como visto, uma mudança na estrutura do curso pode causar uma piora nos resultados, pois muda a natureza da interação com a plataforma. Novos modelos serão criados utilizando os dados coletados nas aulas presenciais. Pretende-se avaliar a robustez e generalização dos modelos no longo prazo. Outra abordagem promissora é a expansão das fontes de dados para incluir informações adicionais sobre os alunos, como histórico educacional anterior e dados socioeconômicos. A inclusão desses dados mais ricos pode fornecer percepções mais profundas sobre os fatores que influenciam o desempenho dos alunos e ajudar a melhorar a capacidade de previsão.

7. Agradecimentos

Este trabalho foi apoiado pelo CNPq através da bolsa PIBIC UFRJ, pela UNIRIO através do PPInst 2023 e pelo Instituto Reditus através do Edital de Inovação.

Referências

Agrusti, F., Mezzini, M., and Bonavolontà, G. (2020). Deep learning approach for predicting university dropout: a case study at roma tre university. *Journal of e-Learning and Knowledge Society*, 16(1):44–54.

Al-Shabandar, R., Hussain, A. J., Liatsis, P., and Keight, R. (2018). Analyzing learners behavior in moocs: An examination of performance and motivation using a data-driven approach. *IEEE Access*, 6:73669–73685.

Bravo-Agapito, J., Romero, S. J., and Pamplona, S. (2021). Early prediction of undergraduate student's academic performance in completely online learning: A five-year study. *Computers in Human Behavior*, 115:106595.

Burgos, C., Campanario, M. L., de la Peña, D., Lara, J. A., Lizcano, D., and Martínez, M. A. (2018). Data mining for modeling students' performance: A tutoring action plan to prevent academic dropout. *Computers & Electrical Engineering*, 66:541–556.

Chen, W., Brinton, C. G., Cao, D., Mason-Singh, A., Lu, C., and Chiang, M. (2019). Early detection prediction of learning outcomes in online short-courses via learning behaviors. *IEEE Transactions on Learning Technologies*, 12(1):44–58.

Delgado, C., da Silva, J., Mascarenhas, F., and Duboc, A. (2016). The teaching of functions as the first step to learn imperative programming. *Anais do Workshop sobre Educação em Computação (WEI)*, pages 2393–2402.

Dobson, A. and Barnett, A. (2008). *An Introduction to Generalized Linear Models*. Chapman & Hall/CRC Texts in Statistical Science. CRC Press.

Hu, Y.-H., Lo, C.-L., and Shih, S.-P. (2014). Developing early warning systems to predict students' online learning performance. *Computers in Human Behavior*, 36:469–478.

James, G., Witten, D., Hastie, T., and Tibishirani, R. (2021). *An Introduction to Statistical Learning with Applications in R*. Springer, New York, NY, 2 edition.

Marbouti, F., Diefes-Dux, H. A., and Madhavan, K. (2016). Models for early prediction of at-risk students in a course using standards-based grading. *Computers & Education*, 103:1–15.

Montgomery, D., Peck, E., and Vining, G. (2013). *Introduction to Linear Regression Analysis*. Wiley Series in Probability and Statistics. Wiley.

Moraes, L., Delgado, C., Freire, J., and Pedreira, C. (2022). Machine teaching: uma ferramenta didática e de análise de dados para suporte a cursos introdutórios de programação. In *Anais do II Simpósio Brasileiro de Educação em Computação*, pages 213–223, Porto Alegre, RS, Brasil. SBC.

Qian, Y. and Lehman, J. (2017). Students' misconceptions and other difficulties in introductory programming: A literature review. *ACM Transactions on Computing Education*, 18(1).

Xará, G., Moraes, L., Delgado, C., Freire, J., and Farias, C. (2023). Dealing with a large number of students and inequality when teaching programming in higher education. In *Anais do XXIX Workshop de Informática na Escola*, pages 1230–1242, Porto Alegre, RS, Brasil. SBC.