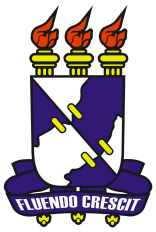

UNIVERSIDADE FEDERAL DE SERGIPE
CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA
DEPARTAMENTO DE COMPUTAÇÃO

COMP0424 - APRENDIZAGEM DE MÁQUINA (2025.2 - T01)

Matheus Vinicius Ramos Guimaraes

Pericles Maikon de Jesus Costa

Thiago Menezes de Oliveira

RELATÓRIO TÉCNICO

Predict Student Performance from Game Play

São Cristóvão - SE

25/02/2026

SUMÁRIO

1. Introdução..... 3
1.1. Explicação do problema..... 3
1.2. Link da competição..... 3
2. Análise exploratória de dados.....3
3. Explicação do algoritmo escolhido..... 4
4. Relato..... 4
5. Reutilização e adaptação da solução em um problema real..... 5

1. Introdução

A aplicação de técnicas de Ciência de Dados no contexto educacional tem crescido bastante, especialmente com a criação de jogos de aprendizagem. A capacidade de rastrear as ações de um aluno dentro de um ambiente digital permite criar sistemas de tutoria mais inteligentes. Este relatório detalha a abordagem e a solução desenvolvida para prever o desempenho de estudantes em tempo real, baseando-se em seus rastros de interação (logs) dentro de um jogo educativo.

1.1. Explicação do problema

O desafio consiste em analisar logs de tempo real de jogadores do jogo educativo de Jo Wilder e prever se o aluno responderá corretamente a uma série de 18 perguntas avaliativas distribuídas ao longo de três pontos de verificação (níveis 0-4, 5-12 e 13-22).

1.2. Link da competição

Kaggle: Predict Student Performance from Game Play

2. Análise exploratória de dados

Os dados brutos fornecidos eram de 4.74 GB para o conjunto de treino, contendo milhares de linhas que registravam cada milissegundo de interação dos usuários.

- **Estrutura dos Dados:** O dataset possui granularidade de evento (ex: Maps_click, map_hover). Cada linha representa uma única ação do mouse ou da tela.
- **Desafio de Memória:** O carregamento padrão via Pandas estouraria facilmente o limite de RAM. A primeira etapa da análise exigiu a redefinição de tipos de dados, convertendo variáveis int64 para int32 e strings repetitivas para category, o que reduziu drasticamente o peso na memória.
- **Agrupamento e Transformação:** Como o objetivo é prever a resposta do usuário, foi necessário transformar as séries temporais de ações em dados tabulare. Foram agregadas estatísticas como o tempo

máximo gasto na sessão, média de tempo por ação e o nível máximo alcançado em cada grupo de fases.

3. Explicação do algoritmo escolhido

A Random Forest é um método de aprendizado de conjunto que opera construindo múltiplas árvores de decisão durante o treinamento. Para a previsão, ela emite a classe que é a moda das classes previstas pelas árvores individuais.

Motivos da escolha:

- **Robustez:** É altamente resistente a overfitting, o que é ideal para baselines em dados tabulares complexos.
- **Lida bem com dados não normalizados:** Não foi necessário aplicar escalonamento aos tempos de tela ou níveis, economizando processamento.
- **Controle de Complexidade:** Através do hiperparâmetro max_depth=10, foi possível limitar a profundidade das árvores, garantindo inferências extremamente rápidas e baixo uso de memória RAM, atendendo perfeitamente aos critérios de eficiência exigidos pela API de simulação.

4. Relato

O desenvolvimento desta solução foi marcado por alguns desafios.

1. Instabilidade do Kaggle: Ao tentar partir da competição não obtive sucesso, precisei tentar novamente no dia seguinte. Em seguida, por algum motivo não consegui acesso aos notebooks criados, apenas alguns dias depois o problema se resolveu e consegui acesso e iniciar o desenvolvimento.
2. Versão do notebook: Por se tratar de uma competição de 2023 a versão do python utilizada para os dados era uma antiga, notebooks criados recentemente vem com uma versão atualizada, dessa forma foi necessário a criação de um notebook fazendo a cópia de notebooks de outras submissões que estavam na versão correta.
3. Falta de memória RAM: O carregamento do dataset train.csv (com mais de 10 mil linhas) causava esgotamento de RAM.

Apesar dos desafios, a solução foi realizada e submetida, mas, durante a escrita deste relato, não houve resultado da posição no rank.

5. Reutilização e adaptação da solução em um problema real

Um cenário prático e genérico para adaptar essa arquitetura preditiva é a análise de comportamento de clientes em plataformas de e-commerce ou aplicativos de software como serviço (SaaS). Da mesma forma que o modelo processa cliques e tempos em tela para prever o desempenho de um estudante, ele pode analisar os rastros de navegação de um usuário para prever ações comerciais, como a probabilidade de abandono de carrinho ou a conversão de um plano gratuito para uma assinatura paga.

Ao extrair variáveis como o tempo gasto visualizando uma página (elapsed_time), a frequência de retornos ao site e a quantidade de cliques em opções de frete ou ferramentas bloqueadas, o modelo de Random Forest consegue identificar padrões de intenção. Com essas previsões rodando em segundo plano, a plataforma pode intervir em tempo real, disparando um cupom de desconto automatizado ou um alerta no chat de suporte exatamente no momento em que o cliente demonstra indecisão, otimizando as taxas de conversão de forma eficaz.