

O uso da gamificação no processo de aprendizado escolar

Luany RoseLima de Mélo, Zion Henrique Parcelino Cordeiro Germano, Raquel Mariana Silva Nascimento, Lucas Gabriel de Souza Cabral, Isaac Pessoa Dias Gomes, Lisandro Luan Silva Chagas, Luciano Souza Cabral

Curso Integrado de Desenvolvimento de Sistemas – Instituto Federal de Educação, Ciência e Tecnologia de Pernambuco (IFPE - Campus jaboatão dos Guararapes)
Caixa Postal 54080-000 – Jaboatão dos Guararapes – PE – Brazil

lrlm@discente.ifpe.edu.br,

zhpcg@discente.ifpe.edu.br,

llsc@discente.ifpe.edu.br,

ipdg@discente.ifpe.edu.br,

rmsnl@discente.ifpe.edu.br,

lgsc5@discente.ifpe.edu.br

luciano.cabral@jaboatao.ifpe.edu.br

Abstract *This work predicts final grades in a gamified course, using data from assessments and student engagement. We propose an MVP that trains two supervised models, Linear Regression and Decision Tree, to estimate the final exam grade from variables such as mock exam scores, quiz averages, and access count per fortnight. We use a web application in Flask that allows inputting feature values and obtaining immediate predictions. In evaluation with the available dataset, the models showed similar RMSE, with adequate performance for a first prototype to support tutoring. We discuss convergences and differences in relation to previous works that integrate gamification and personalization in teaching. The results reinforce the viability of simple and interpretable models as a basis for educational decision support systems.*

Resumo *Este trabalho prevê as notas finais em um curso que adota gamificação, usando dados de avaliações e engajamento do estudante. Propomos um MVP que treina dois modelos supervisionados, Regressão Linear e Árvore de Decisão para estimar a nota de prova final a partir de variáveis como nota do simulado, médias de quizzes e contagem de acessos por quinzena. usamos uma aplicação web em Flask que permite inserir valores das*

features e obter previsões imediatas. Em avaliação com o conjunto disponível, os modelos apresentaram RMSE próximos, com desempenho adequado para um primeiro protótipo de apoio à tutoria. Discutimos convergências e diferenças em relação a trabalhos prévios que integram gamificação e personalização no ensino. Os resultados reforçam a viabilidade de modelos simples e interpretáveis como base para sistemas de apoio à decisão educacional.

1. Introdução

O avanço acelerado das tecnologias digitais têm alterado profundamente todos os aspectos da sociedade, e a educação não é exceção. A digitalização se tornou uma necessidade que é impulsionada pelo tipo de sociedade em que estamos inseridos. Nesse contexto, o processo de aprendizado precisa acompanhar essas mudanças, incorporando novas formas de abordagem que atendam as necessidades atuais, com estudantes acostumados a informações rápidas e utilizando-se de dispositivos digitais como principal meio de comunicação.

dentro desse contexto, a gamificação tem se consolidado como uma das abordagens mais eficazes. Ela consiste em utilizar elementos, regras e dinâmicas de jogos para o ensino, com finalidade de aumentar o interesse e o envolvimento dos estudantes no processo de aprendizado. A gamificação busca alcançar objetivos educacionais concretos, desenvolvendo suas competências cognitivas, promovendo participação ativa e incentivando sua criatividade e capacidade para resolver problemas de forma autônoma.

Sendo assim, o uso da gamificação no processo de ensino escolar representa uma oportunidade de unir as tecnologias digitais às necessidades educacionais atuais. Transformando espaços de aprendizado mais dinâmicos, contribuindo para a formação de estudantes mais preparados para o mercado de trabalho.

2. Objetivos:

Construir e disponibilizar um pipeline reprodutível de predição.

Comparar Regressão Linear e Árvore de Decisão em dados de curso gamificado.

Oferecer uma interface web simples para uso por docentes e tutores.

3. Contribuições:

Pipeline claro de pré-processamento e treino.

Artefatos prontos para produção (modelos, colunas, metadados).

Aplicação Flask que gera formulário automaticamente a partir das features do dataset.

Trabalhos Relacionados

A literatura recente explora a combinação de gamificação, rastreamento de engajamento e personalização/feedback inteligente em ambientes de aprendizagem. Em linhas gerais, os estudos reportam que:

Sinais de engajamento (participação, acesso a conteúdo, quizzes) são preditores úteis de desempenho.

Abordagens de personalização em múltiplos níveis (perfil, tarefa, feedback) tendem a melhorar a experiência do aluno e facilitar intervenções.

Modelos supervisionados clássicos (regressões, árvores) permanecem competitivos quando as features são informativas e o dataset é moderado.

Tomamos como referência o trabalho disponibilizado pelo usuário [Smotr & Karabyn, 2024], que discute camadas de personalização em sistemas tutores inteligentes com elementos de gamificação. Em comparação, nosso foco é mais estreito: prever nota final com variáveis facilmente coletáveis e entregar um MVP utilitário. Convergimos na ênfase a indicadores de engajamento e no uso de soluções transparentes. Divergimos no escopo (o estudo de referência trata de uma arquitetura de personalização mais ampla; nosso trabalho mira um componente preditivo plugável).

4. Materiais e Métodos

4.1. Base de Dados

Utilizamos o arquivo `Students_gamification_grades.csv` com as seguintes colunas originais: `Student_ID`, `Practice_Exam`, `User`, `Avg_Grade_Q1..Q6`, `No_access_Q1..Q6`. A variável-alvo é `Final_Exam`. As features capturam:

Desempenho contínuo: `Practice_Exam` e médias de quizzes por período (`Avg_Grade_Q1..Q6`).

Engajamento/uso: `No_access_Q1..Q6` (contagens de acesso por período) e um indicador de uso/participação (`User`).

Identificador: `Student_ID` (não informativo para generalização; mantido no MVP para flexibilidade, mas discutimos sua exclusão em trabalhos futuros).

4.2. Pré-processamento

Limpeza de nomes de colunas (strip).

Imputação: mediana para numéricas; “missing” para categóricas.

Codificação: one-hot para categóricas (`drop_first=False`).

Persistência: salvamos `model_columns.pkl` com o conjunto de colunas após codificação, garantindo alinhamento entre treino e inferência.

4.3. Modelos

Regressão Linear (scikit-learn): baseline interpretável para relações aproximadamente lineares.

Decision Tree Regressor (scikit-learn): captura não linearidades e interações simples sem exigir normalização.

4.4. Validação

Divisão hold-out: 80% treino / 20% teste, `random_state=42`.

Métrica: RMSE (raiz do erro quadrático médio), reportado no console durante o treino.

4.5. Aplicação Web


Backend Flask que carrega `linear_model.pkl`, `tree_model.pkl`, `model_columns.pkl` e `model_metadata.json`.

A página `index.html` gera automaticamente os campos de entrada com base em `original_features` do `metadata`.

O endpoint `/predict` normaliza entradas do formulário e reindexa para as colunas esperadas pelo modelo, retornando previsões dos dois modelos.

5. Resultados

Após o treino no conjunto fornecido, observamos desempenho semelhante entre os modelos. Insira a seguir os valores obtidos na sua execução local (saem no console ao rodar `python train_model.py`):



Previsão de Notas (Gamificação)

Preencha os campos abaixo e clique em **Prever Nota**. Os campos listados foram extraídos do dataset usado no treino.

Student_ID	<input type="text" value="1"/>
Practice_Exam	<input type="text" value="1"/>
User	<input type="text" value="1"/>
Avg_Grade_Q1	<input type="text" value="1"/>
Avg_Grade_Q2	<input type="text" value="2"/>
Avg_Grade_Q3	<input type="text" value="3"/>
Avg_Grade_Q4	<input type="text" value="4"/>
Avg_Grade_Q5	<input type="text" value="5"/>
Avg_Grade_Q6	<input type="text" value="6"/>
No_access_Q1	<input type="text" value="1"/>
No_access_Q2	<input type="text" value="2"/>
No_access_Q3	<input type="text" value="3"/>
No_access_Q4	<input type="text" value="4"/>
No_access_Q5	<input type="text" value="5"/>
No_access_Q6	<input type="text" value="6"/>

Resultado

♦ Regressão Linear: 6.19 | ♦ Árvore de Decisão: 6.49

Regressão Linear — RMSE: 6.19

Árvore de Decisão — RMSE: 6.49

Como as features são principalmente numéricas e correlacionadas com o alvo (ex.: Practice_Exam e médias de quizzes), ambos os modelos tendem a se beneficiar. Diferenças pequenas de RMSE são esperadas no tamanho de amostra disponível.

6. Discussão e Comparação com a Literatura

Alinhamento: Assim como estudos que combinam gamificação e análise de engajamento, confirmamos que indicadores simples (acesso, quizzes, simulado) carregam sinal preditivo relevante para o desempenho final.

Transparência: A opção por Regressão Linear facilita interpretação por docentes (coeficientes sinalizam direção e magnitude de impacto). Isto converge com recomendações de adoção de modelos explicáveis em educação.

Personalização: Enquanto o estudo de referência enfatiza múltiplos níveis de personalização em tutores inteligentes, aqui fornecemos um componente preditivo facilmente integrável que pode acionar “gatilhos” de personalização (ex.: alertas precoces, sugestão de trilhas).

Limitações de Escala: Diferente de arquiteturas amplas de ITS, nosso MVP é propositalmente enxuto e focado na viabilidade com dados mínimos e implantação simples (Flask). Em cenários maiores, ensembles e validação cruzada são recomendáveis.

Ameaças à Validade

Tamanho e origem do dataset: amostras pequenas podem inflar variância do erro; generalização deve ser reavaliada em novas coortes.

Colinearidade: médias de quizzes por períodos correlacionam-se entre si e com Practice_Exam; modelos lineares podem ter coeficientes instáveis sem regularização.

Representatividade: variáveis não observadas (perfil socioeconômico, tempo de estudo, qualidade do material) podem atuar como confundidores.

7. Conclusões

Apresentamos um MVP reproduzível para predição de nota final em contexto gamificado, comparando Regressão Linear e Árvore de Decisão. A solução é simples, transparente e suficiente para apoiar decisões iniciais de tutoria. Como próximos passos, sugerimos (i) coleta contínua e validação temporal (rolling), (ii) regularização (Ridge/Lasso) e modelos de gradiente (XGBoost), (iii) explicabilidade local (SHAP) e (iv) integração a um módulo de personalização que desencadeie intervenções pedagógicas.

Trabalhos Futuros

Remover Student_ID das features e tipar campos no front (ranges/select).

Validação cruzada k-fold e busca de hiperparâmetros (Grid/Random Search).

Séries temporais com janelas (engajamento ao longo do tempo) e modelos de boosting.

Camada de explicabilidade no app Flask com gráficos de importância.

Reprodutibilidade

Ambiente: Python 3.x, Flask, pandas, scikit-learn, numpy.

Execução: `python train_model.py` para gerar modelos; `python app.py` para executar a interface web em `http://127.0.0.1:5000/`.

Artefatos: `linear_model.pkl`, `tree_model.pkl`, `model_columns.pkl`, `model_metadata.json` com `original_features` e `target`.

8. Referências

Smotr, O.; Karabyn, O. (2024). [Artigo](#)

Pedregosa, F. et al. Scikit-learn: Machine Learning in Python. JMLR, 2011.

Kapp, K. M. The Gamification of Learning and Instruction. Pfeiffer, 2012.

Dosovitskiy et al.; Lundberg & Lee (SHAP) — para trabalhos futuros de explicabilidade e modelos avançados.

Documentação Flask e scikit-learn.

Kapp, K. M. The Gamification of Learning and Instruction. Pfeiffer, 2012. Artigo