



Esta é uma cópia de uma conversa entre ChatGPT e Adriano.
Denunciar conteúdo

Arquivo carregado

Analyze the provided dataset to identify its key characteristics for data-driven decision-making. Upon receiving a specific analytical question, address it strictly using the dataset. Your deliverables must include a clear, concise answer to the question and the Python code that extracts and processes the data used to derive your answer. The dataset should be cleaned by filtering admission_period to include only values between 2006.1 and 2019.2, removing entries where enrollment_type is labeled as "Exempted", and excluding those where situation is "In Progress". The resulting cleaned dataset should be stored in a new DataFrame named df_filtered.

I'll start by loading the dataset and inspecting its structure. Then, I'll clean it based on your specified criteria:

1. Filter admission_period to include only values between 2006.1 and 2019.2.
2. Remove entries where enrollment_type is labeled as "Exempted".
3. Exclude records where situation is "In Progress".

Let's begin.

The dataset consists of 30,130 entries and 34 columns. The key columns relevant to our filtering are:

- admission_period (float): Needs to be between 2006.1 and 2019.2.
- enrollment_type (object): Entries labeled "Exempted" should be removed.
- situation (object): Entries labeled "In Progress" should be excluded.

Now, I'll apply these filters and store the cleaned dataset in df_filtered. [~]

The dataset has been successfully filtered down to 16,563 entries. The cleaned dataset (df_filtered) is now ready for analysis.

Let me know your specific analytical question, and I'll extract insights strictly using this dataset. 🚀 [~]

Using advanced analytical methods, track a student's academic trajectory (based on sequences of grades and course outcomes) to predict the likelihood of course withdrawal. Outline three methodological options and justify the best approach.

To predict the likelihood of course withdrawal based on a student's academic trajectory, we can leverage sequential data analysis and machine learning. Here are three advanced methodological options:



1. Sequence Classification with Recurrent Neural Networks (RNNs) / Long Short-Term Memory (LSTM)

- How it works:
 - Uses sequential data (e.g., semester-wise grades, enrollment history) to detect patterns leading to dropout.
 - LSTM networks handle long-term dependencies, making them ideal for tracking academic progress over time.
- Pros:
 - Effective in capturing the temporal nature of academic performance.
 - Handles irregular sequences (e.g., students may take breaks or repeat courses).
- Cons:
 - Requires a significant amount of data for training.
 - Computationally expensive.

2. Hidden Markov Models (HMMs) for State-Based Trajectory Prediction

- How it works:
 - Models student progress as a sequence of hidden states (e.g., "High Achiever" → "Struggling" → "Dropout").
 - Transitions between states depend on grades, failed courses, and enrollment trends.
- Pros:
 - Captures uncertainty in student trajectories.
 - Good for modeling probabilistic transitions between academic states.
- Cons:
 - Assumes Markovian property (future state depends only on the current state).
 - May not capture complex dependencies as well as deep learning models.

3. Gradient Boosting Models (e.g., XGBoost) for Feature-Based Prediction

- How it works:
 - Uses tabular data (e.g., grades, number of failed courses, enrollment gaps) to predict dropout probability.
 - Trains an ensemble of decision trees that focus on the most predictive features.
- Pros:
 - Works well with structured academic data.
 - Interpretable: Can identify key factors influencing dropouts (e.g., failing a core subject).
 - Efficient and requires less data compared to deep learning models.
- Cons:
 - Does not explicitly model the sequential nature of the data.
 - Requires careful feature engineering.

Best Approach: LSTM-Based Sequence Prediction

- Justification:
 - Academic performance follows a temporal pattern (semesters, course sequences).
 - LSTMs can learn from past performance trends and predict whether a student is at risk of dropping out.
 - Works well even with irregular academic paths (e.g., course retakes, semester gaps).
- Next Steps:
 - Prepare input sequences (student's grades, semesters, course attempts).
 - Train an LSTM model on historical student data.
 - Evaluate performance using accuracy, recall (for detecting dropouts), and AUC-ROC.

Would you like me to implement an exploratory analysis or prepare data for an LSTM model? 🚀

Follow the best approach to solve this problem based on the dataset

Great choice! We'll develop an LSTM-based model to predict course withdrawal using students' academic

1. Define the target variable (`withdrawal = 1`, `continuing/graduated = 0`).
2. Extract relevant features:
 - `student_id`, `admission_period`, `enrollment_period`
 - `subject_name`, `final_average`, `situation`
3. Convert categorical variables into numerical representations.
4. Create sequential data for each student, tracking their academic journey.

1. Structure input data into sequences (e.g., per semester).
2. Normalize features (grades, periods) for LSTM processing.
3. Split dataset into training and testing sets.

1. Use an embedding layer to handle categorical features.
2. Implement an LSTM network to learn student performance trends.
3. Optimize using binary classification (dropout vs. retention).

- Use accuracy, recall, precision, and AUC-ROC to measure effectiveness.
- Test on unseen data to validate predictions.

- "subject_name" and "situation" are mapped to numerical codes.

Let me know how you'd like to proceed! 🚀 (x)

•||• **Voz**

O ChatGPT pode cometer erros. Considere verificar informações importantes.