

Dataset Filtering and Preparation

22/03/2025



stratified_history.csv
8.1 MB

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.



Analysis of the Dataset and Approach to Answering the Question

The dataset `stratified_history.csv` contains information about admissions, enrollment types, and situations. To ensure data-driven decision-making, we will:

1. Filter the dataset based on the specified criteria:
 - `admission_period` must be between 2006.1 and 2019.2 (inclusive).
 - Exclude rows where `enrollment_type` is "Exempted".
 - Exclude rows where `situation` is "In Progress".
2. Store the cleaned dataset in a new DataFrame named `df_filtered`.
3. Use `df_filtered` to address any specific analytical question provided.

Below is the Python code to clean the dataset and prepare it for analysis:

Python Code for Data Cleaning

```
python
1 import pandas as pd
2
3 # Load the dataset
4 df = pd.read_csv('stratified_history.csv')
5
6 # Filter admission_period to include only values between 2006.1 and 2019.2
7 df_filtered = df[(df['admission_period'] >= 2006.1) & (df['admission_period'] <= 2019.2)]
8
9 # Remove entries where enrollment_type is labeled as "Exempted"
10 df_filtered = df_filtered[df_filtered['enrollment_type'] != 'Exempted']
11
12 # Exclude rows where situation is "In Progress"
13 df_filtered = df_filtered[df_filtered['situation'] != 'In Progress']
14
15 # Display the first few rows of the cleaned dataset
16 print(df_filtered.head())
```

Explanation of the Code

1. Loading the Dataset : The dataset is loaded using `pd.read_csv()`.
2. Filtering `admission_period` : We use boolean indexing to retain only rows where `admission_period` falls within the range [2006.1, 2019.2].
3. Removing "Exempted" Entries : Rows with `enrollment_type` equal to "Exempted" are excluded.
4. Excluding "In Progress" Situations : Rows with `situation` equal to "In Progress" are removed.
5. Resulting DataFrame : The cleaned dataset is stored in `df_filtered`.

Next Steps

Once you provide a specific analytical question, I will:

1. Extract relevant data from `df_filtered`.
2. Perform the necessary analysis.
3. Provide a clear, concise answer along with the corresponding Python code.

Please provide your analytical question, and I will proceed accordingly.

Determine the statistical correlation (Pearson or Spearman) between students' overall average and their enrollment period (first vs. second semester). Should correlation coefficients be consistent across different years, what factors might explain this pattern?

 qwen-max-latest

To address the question, we will analyze the cleaned dataset (`df_filtered`) to calculate the statistical correlation between students' overall average and their enrollment period. The enrollment period is categorized as either the first semester (`.1`) or the second semester (`.2`). We will use both Pearson and Spearman correlation coefficients to assess the relationship.

Steps to Solve the Problem:

1. Extract Relevant Columns :
 - `admission_period` : This column indicates the enrollment period (e.g., `2017.1` for the first semester of 2017).
 - `overall_average` : This column contains the students' overall average scores.
2. Transform `admission_period` :
 - Extract the semester information from `admission_period` (`.1` for the first semester and `.2` for the second semester).
 - Convert this into a binary numerical variable: `1` for the first semester and `2` for the second semester.
3. Calculate Correlation Coefficients :
 - Use Pearson correlation to measure the linear relationship.
 - Use Spearman correlation to measure the monotonic relationship.
4. Analyze Consistency Across Years :
 - Group the data by year (extracted from `admission_period`) and compute the correlation coefficients for each year.
 - Assess whether the correlation coefficients remain consistent across years.
5. Interpret Results :
 - Discuss potential factors explaining any observed patterns in the correlation coefficients.

Python Code for Analysis

```
python
1 import pandas as pd
2 import numpy as np
3 from scipy.stats import pearsonr, spearmanr
4
5 # Load the dataset and filter it
6 df = pd.read_csv('stratified_history.csv')
7
8 # Filter admission_period to include only values between 2006.1 and 2019.2
9 df_filtered = df[(df['admission_period'] >= 2006.1) & (df['admission_period'] <= 2019.2)]
10
11 # Remove entries where enrollment_type is labeled as "Exempted"
12 df_filtered = df_filtered[df_filtered['enrollment_type'] != 'Exempted']
13
14 # Exclude rows where situation is "In Progress"
15 df_filtered = df_filtered[df_filtered['situation'] != 'In Progress']
16
17 # Extract semester information from admission_period
18 df_filtered['semester'] = df_filtered['admission_period'].apply(lambda x: 1 if x % 1 == 0.1 else
19
20 # Drop rows with missing values in overall_average
21 df_filtered = df_filtered.dropna(subset=['overall_average'])
22
23 # Calculate Pearson and Spearman correlations
24 pearson_corr, _ = pearsonr(df_filtered['semester'], df_filtered['overall_average'])
25 spearman_corr, _ = spearmanr(df_filtered['semester'], df_filtered['overall_average'])
26
27 print(f"Pearson Correlation: {pearson_corr}")
28 print(f"Spearman Correlation: {spearman_corr}")
29
30 # Analyze consistency across years
31 df_filtered['year'] = df_filtered['admission_period'].apply(lambda x: int(x))
32
33 correlations_by_year = []
34 for year in df_filtered['year'].unique():
35     subset = df_filtered[df_filtered['year'] == year]
36     if len(subset) > 1: # Ensure sufficient data for correlation
37         pearson_corr_year, _ = pearsonr(subset['semester'], subset['overall_average'])
38         spearman_corr_year, _ = spearmanr(subset['semester'], subset['overall_average'])
39     correlations_by_year.append({
40         'year': year,
41         'pearson_corr': pearson_corr_year,
42         'spearman_corr': spearman_corr_year
43     })
44
```

```
44
45 # Convert results to a DataFrame
46 correlations_df = pd.DataFrame(correlations_by_year)
47 print(correlations_df)
```

Explanation of the Code

1. Filtering and Cleaning :

- The dataset is filtered based on the specified criteria (`admission_period` , `enrollment_type` , and `situation`).
- Missing values in `overall_average` are removed to ensure valid calculations.

2. Extracting Semester Information :

- The `admission_period` column is processed to extract the semester (`.1` or `.2`) and convert it into a numerical variable (`1` or `2`).

3. Correlation Calculation :

- Pearson and Spearman correlations are computed between `semester` and `overall_average` .

4. Yearly Analysis :

- The dataset is grouped by year, and correlations are calculated for each year to assess consistency.

Results and Interpretation

1. Overall Correlation

The Pearson and Spearman correlation coefficients provide insights into the relationship between students' overall averages and their enrollment period:

- A **positive correlation** suggests that students enrolled in the second semester tend to have higher averages.
- A **negative correlation** suggests the opposite.
- A **near-zero correlation** implies no significant relationship.

2. Consistency Across Years

If the correlation coefficients are consistent across years, it may indicate stable patterns in academic performance based on enrollment period. Possible explanations for such patterns include:

- **Semester-Specific Curriculum** : Differences in course difficulty or workload between semesters.
- **Student Demographics** : Variations in student profiles (e.g., academic preparedness) between semesters.
- **External Factors** : Seasonal effects, such as holidays or exam schedules, influencing performance.

3. Factors Explaining Patterns

If correlations vary significantly across years, potential factors could include:

- Changes in admission policies or curriculum structure.
- Economic or social conditions affecting student performance.
- Random fluctuations due to small sample sizes in certain years.

Final Answer

After running the code, you will obtain:

1. The **Pearson and Spearman correlation coefficients** for the entire dataset.
2. A table showing yearly correlations, which can be used to assess consistency.

These results will help determine whether a significant relationship exists between students' overall averages and their enrollment period, as well as identify factors explaining any observed patterns.

