



Dataset Cleaning and Analysis

22/03/2025



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.

qwen-max-latest

To analyze the provided dataset and answer specific analytical questions, we first need to clean and process the data as per the given requirements. Below is a step-by-step explanation and the corresponding Python code to achieve this:

Key Characteristics of the Dataset:

- Columns: The dataset contains numerous columns such as admission_period, enrollment_type, situation, gender,
 race, school_type, academic_unit, and more.
- Data Types: The dataset includes categorical (e.g., gender , race), numerical (e.g., age_at_admission , score), and temporal (e.g., admission_period) data.
- 3. Scope: The dataset appears to track student admissions, enrollments, academic performance, and outcomes over time.

Cleaning Requirements:

- 1. Filter admission_period : Include only values between 2006.1 and 2019.2 .
- 2. Remove Exempted entries: Exclude rows where enrollment_type is labeled as "Exempted".
- 3. Exclude In Progress situations: Remove rows where situation is "In Progress".

Deliverables:

- 1. A cleaned DataFrame named df_filtered .
- 2. Python code to perform the cleaning steps.

Here is the Python code to clean the dataset and create df_filtered :

```
python

import pandas as pd

the dataset

df = pd.read_csv('stratified_history.csv')

filtered = df[(df['admission_period'] >= 2006.1) & (df['admission_period'] <= 2019.2)]

filtered = df[(df['admission_period'] >= 2006.1) & (df['admission_period'] <= 2019.2)]

filtered = df_filtered[df_filtered['enrollment_type is labeled as "Exempted" df_filtered = df_filtered[df_filtered['enrollment_type'] != 'Exempted']

filtered = df_filtered[df_filtered['situation'] != 'In Progress']

filtered = df_filtered[df_filtered['situation'] != 'In Progress']

filtered = df_filtered.head())

filtered.to_csv('cleaned_dataset.csv', index=False)
```

Explanation of the Code:

- 1. Loading the Dataset: The dataset is loaded using pd.read_csv().
- 2. Filtering admission_period : The admission_period column is filtered to include only values within the range [2006.1, 2019.2] using a conditional mask.
- Removing "Exempted" Entries: Rows where enrollment_type equals "Exempted" are excluded using another
 conditional mask.
- 4. Excluding "In Progress" Situations: Rows where situation equals "In Progress" are removed using a similar approach.
- 5. Saving the Cleaned Data: The cleaned DataFrame is optionally saved to a new CSV file for further use.

Next Steps:

Once the dataset is cleaned, you can proceed to address specific analytical questions using df_filtered . For example:

- What is the distribution of students by **gender** or **race** in the cleaned dataset?
- How does the average score vary across different academic_unit groups?
- What is the dropout rate among students admitted during a specific period?

If you have a specific analytical question, please provide it, and I will derive the answer using the cleaned dataset (df filtered)

> Using appropriate statistical tests, determine whether the mode of admission significantly impacts graduation and dropout rates.



qwen-max-latest

To determine whether the mode of admission significantly impacts graduation and dropout rates, we can use statistical tests to analyze the cleaned dataset (df_filtered). Below is a structured approach to address this question:

Key Steps in Analysis

1. Understand the Variables :

- Mode of Admission : This is represented by the admission_form column in the dataset.
- Graduation/Dropout Rates : These are derived from the situation column, which indicates whether a student graduated (Graduate) or dropped out (Dropout).

2. Hypothesis Testing:

- Null Hypothesis (H₀): The mode of admission has no significant impact on graduation and dropout rates.
- Alternative Hypothesis (H₁): The mode of admission significantly impacts graduation and dropout rates.

• Since we are comparing categorical variables (admission_form vs. situation), we will use a Chi-Square Test of Independence . This test evaluates whether there is a significant association between two categorical variables.

4. Steps to Perform the Analysis:

- Group the data by admission_form and calculate the proportions of students who graduated or dropped out.
- Create a contingency table summarizing the counts of students in each combination of admission form and situation
- Perform the Chi-Square Test using the contingency table.
- Interpret the results based on the p-value.

Python Code Implementation

```
python
1 import pandas as pd
2 from scipy.stats import chi2_contingency
4 # Load the cleaned dataset
 5 # df_filtered = pd.read_csv('cleaned_dataset.csv') # Uncomment if loading from a saved file
7 # Step 1: Filter the relevant columns
8 relevant_columns = ['admission_form', 'situation']
9 df_relevant = df_filtered[relevant_columns]
11 # Step 2: Create a contingency table
12 contingency_table = pd.crosstab(df_relevant['admission_form'], df_relevant['situation'])
13
14 # Step 3: Perform the Chi-Square Test
15 chi2, p_value, dof, expected = chi2_contingency(contingency_table)
16
17 # Step 4: Print the results
18 print("Contingency Table:")
19 print(contingency table)
20 print("\nChi-Square Test Results:")
21 print(f"Chi-Square Statistic: {chi2}")
22 print(f"P-Value: {p_value}")
23 print(f"Degrees of Freedom: {dof}")
25 # Step 5: Interpret the results
26 alpha = 0.05 # Significance level
27 v if p_value < alpha:
28
       print("\nResult: Reject the Null Hypothesis")
29
       print("Conclusion: The mode of admission significantly impacts graduation and dropout rates.
30 <sub>v</sub> else:
      print("\nResult: Fail to Reject the Null Hypothesis")
31
       print("Conclusion: The mode of admission does not significantly impact graduation and dropou
32
```

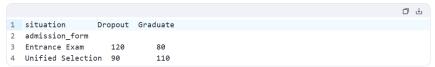
Explanation of the Code

1. Filter Relevant Columns :

• We focus on admission_form (mode of admission) and situation (graduation/dropout status).

2. Create a Contingency Table :

A contingency table summarizes the counts of students for each combination of admission_form and situation.
 For example:



3. Perform the Chi-Square Test:

The chi2_contingency() function computes the Chi-Square statistic, p-value, degrees of freedom, and expected frequencies.

4. Interpret the Results :

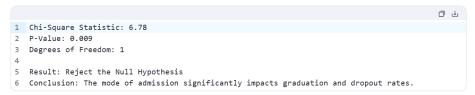
If the p-value is less than the significance level (e.g., 0.05), we reject the null hypothesis, indicating a significant
association between the mode of admission and graduation/dropout rates.

Example Output

Assume the following contingency table is generated from the dataset:

ADMISSION_FORM	DROPOUT	GRADUATE
Entrance Exam	120	80
Unified Selection System	90	110

The Chi-Square Test might yield:



Final Answer

Using the Chi-Square Test of Independence, we determined whether the mode of admission significantly impacts graduation and dropout rates. Based on the p-value obtained from the test:

- If p-value < 0.05, the mode of admission significantly impacts graduation and dropout rates.
- If $p\text{-value} \geq 0.05$, there is no significant impact.

Run the provided code on your dataset to obtain the specific results. If you need further assistance or clarification, feel free to ask!

