

name: shradha digambar gudmewar

division: CS-5 roll no: 62 PRN:202401100046 name: shradha digambar gudmewar division: CS-5 roll no: 62 PRN:202401100046



import pandas as pd

_

from google.colab import files uploaded = files.upload()

Choose Files No file chosen Upload widget is only available when the cell has been executed in the current browser session. Please rerun this cell to

enable. Saving academic namer reviews sammle.csv to academic mamer reviews sammle.csv

path="/content/academic_paper_reviews_sample.csv"

y content, academic paper _ reviews_samps
df=pd.read_csv(path)
df.head(400)

7	Title	Abstract	Reviewer	Score	Decision	Review Comments
	Paper on Topic 1	This paper explores topic 1 in detail, analyzi	Reviewer_16	7	Reject	The paper is well-written with strong contribu
	1 Paper on Topic 2	This paper explores topic 2 in detail, analyzi	Reviewer_3	3	Accept	The paper is poorly structured with weak contr
:	Paper on Topic 3	This paper explores topic 3 in detail, analyzi	Reviewer_14	7	Reject	The paper is well-written with strong contribu
;	Paper on Topic 4	This paper explores topic 4 in detail, analyzi	Reviewer_5	2	Revise	The paper is poorly structured with weak contr
4	Paper on Topic 5	This paper explores topic 5 in detail, analyzi	Reviewer_11	5	Reject	The paper is poorly structured with weak contr
						
39	Paper on Topic 396	This paper explores topic 396 in detail, analy	Reviewer_17	10	Accept	The paper is well-written with strong contribu
39	Paper on Topic 397	This paper explores topic 397 in detail, analy	Reviewer_17	7	Accept	The paper is well-written with strong contribu
39	Paper on Topic 398	This paper explores topic 398 in detail, analy	Reviewer_9	7	Reject	The paper is well-written with strong contribu
39	Paper on Topic 399	This paper explores topic 399 in detail, analy	Reviewer_13	8	Accept	The paper is well-written with strong contribu
39	Paper on Topic 400	This paper explores topic 400 in detail, analy	Reviewer_7	3	Reject	The paper is poorly structured with weak contr
40						

400 rows × 6 columns

1]Average score

df['Score'].mean()

→ np.float64(5.66)

2]Count of each decision

df['Decision'].value_counts()

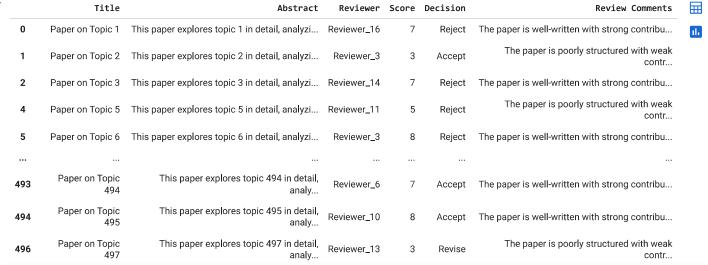
→		count
	Decision	
	Accept	187
	Revise	157
	Reject	156

```
df.groupby('Reviewer')['Score'].mean().idxmax()
4]how to Count duplicate titles
df['Title'].duplicated().sum()
→ np.int64(0)
5] Score distribution by decision
df.groupby('Decision')['Score'].describe()
<del>____</del>*
                count
                                     std min 25% 50% 75% max
      Decision
                187.0 5.663102 2.895840 1.0
       Accept
       Reject
                156.0 5.820513 2.797435 1.0
                                                     6.0
                157.0 5.496815 2.850156 1.0
6]Paper with the lowest score
df.loc[df['Score'].idxmin(), ['Title', 'Score', 'Review Comments']]
6
            Title
                                                 Paper on Topic 7
           Score
      Review Comments  The paper is poorly structured with weak contr...
7] how many are Unique reviewers
df['Reviewer'].nunique()
→ 20
8] what is the Average abstract length (characters)
df['Abstract'].apply(len).mean()
→ np.float64(89.784)
9]Reviewers with more than 10 papers
(df['Reviewer'].value_counts() > 10).sum()
→ np.int64(20)
10] what is the Most common word in review comments
df['Review Comments'].str.split().explode().value_counts().idxmax()
```

```
import numpy as np
df['Score'].mean()
→ np.float64(5.66)
12]calculate median score
import numpy as np
df['Score'].median()
→ 6.0
13]calculate standerd deviation
import numpy as np
df['Score'].std()
2.8482670609514478
14] Percentage of scores below the mean
import numpy as np
(df['Score'] < df['Score'].mean()).mean() * 100
→ np.float64(48.6)
15] Count of perfect scores (10)
import numpy as np
(df['Score']==10).sum()
→ np.int64(61)
16] calculate Range of scores
import numpy as np
df['Score'].max()-df['Score'].min()
→ 9
17] calculate Outliers (z-score > 2)
import numpy as np
df[np.abs(df['Score']-df['Score'].mean())>2*df['Score'].std()]
∓
                                                                                                                      No entries Filter
               Title
                                                                      Score
                                                                                                                  Review Comments
     index
                                                                                       Decision
                             Abstract
                                                  Reviewer
     Show 25 ✓ per page
18] calculate cores within 1 standard deviation of the mean
import numpy as np
df[(df['Score']>df['Score'].mean()-df['Score'].std()) & (df['Score']<df['Score'].mean()+df['Score'].std())]
```

11] calculate Mean score





20] Frequency of each score

import numpy as np