```
import os
import pandas as pd
import numpy as np
import matplotlib
import matplotlib.pyplot as plt
%matplotlib inline
import seaborn as sns
```

DATA CLEANING

Read csv

```
universities_df = pd.read_csv("/content/university_data.csv")
```

```
universities_df.head()
```

	ID number	Name	year	ZIP code	Highest degree offered	County name	Longitude location of institution	La locat insti
0	100654	Alabama A & M University	2013	35762	Doctor's degree - research/scholarship	Madison County	-86.568502	34
1	100663	University of Alabama at Birmingham	2013	35294-0110	Doctor's degree - research/scholarship and pro...	Jefferson County	-86.809170	33
2	100690	Amridge University	2013	36117-3553	Doctor's degree - research/scholarship and pro...	Montgomery County	-86.174010	32
3	100706	University of Alabama in Huntsville	2013	35899	Doctor's degree - research/scholarship and pro...	Madison County	-86.638420	34
4	100724	Alabama State University	2013	36104-0271	Doctor's degree - research/scholarship and pro...	Montgomery County	-86.295677	32

5 rows × 145 columns

Data Cleaning and Formatting

```
universities_df.shape
```

```
(1534, 145)
```

```
universities_df.info(max_cols = len(universities_df))
```

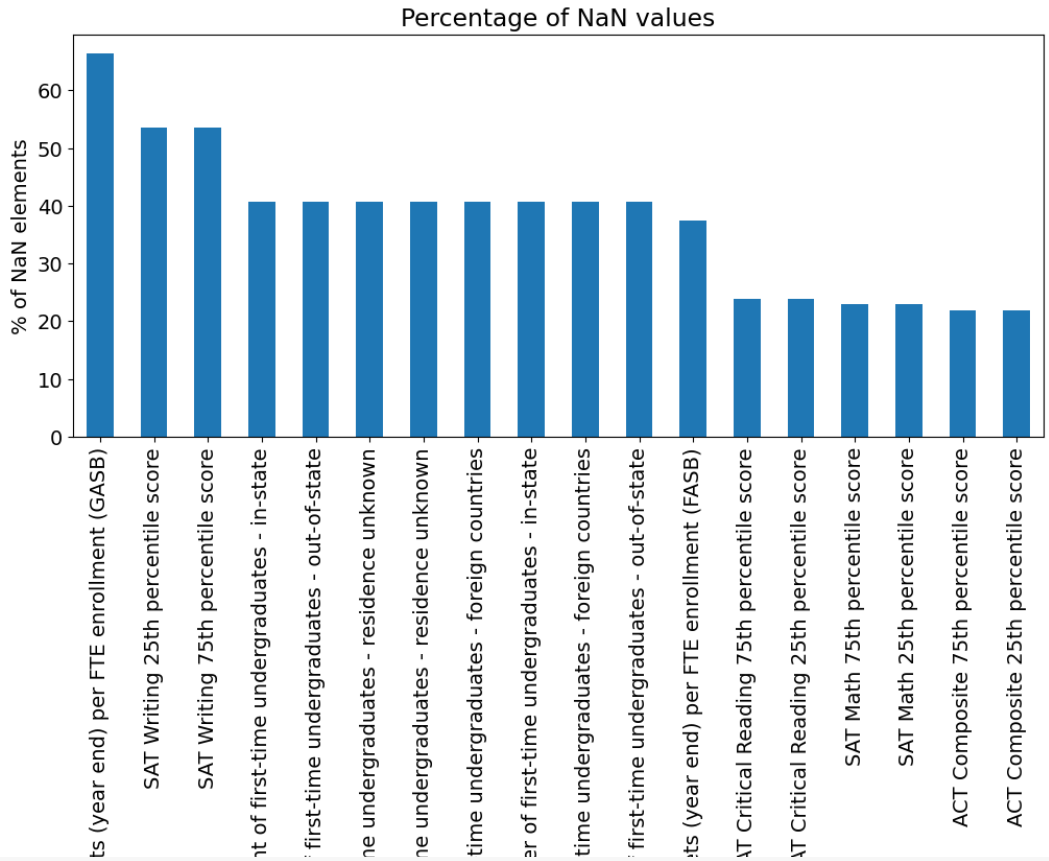
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1534 entries, 0 to 1533
Data columns (total 145 columns):
 #    Column                                                                               Non-Null Count  Dtype
---   ------                                                                               --------------  -----
 0    ID number                                                                            1534 non-null   int64
 1    Name                                                                                 1534 non-null   object
 2    year                                                                                 1534 non-null   int64
 3    ZIP code                                                                             1534 non-null   object
 4    Highest degree offered                                                               1534 non-null   object
 5    County name                                                                          1534 non-null   object
 6    Longitude location of institution                                                    1534 non-null   float64
```

```
7    Latitude location of institution                                  1534 non-null   float64
8    Religious affiliation                                             1532 non-null   object
9    Offers Less than one year certificate                             1532 non-null   object
10   Offers One but less than two years certificate                    1532 non-null   object
11   Offers Associate's degree                                         1532 non-null   object
12   Offers Two but less than 4 years certificate                      1532 non-null   object
13   Offers Bachelor's degree                                          1532 non-null   object
14   Offers Postbaccalaureate certificate                              1532 non-null   object
15   Offers Master's degree                                            1532 non-null   object
16   Offers Post-master's certificate                                  1532 non-null   object
17   Offers Doctor's degree - research/scholarship                     1532 non-null   object
18   Offers Doctor's degree - professional practice                    1532 non-null   object
19   Offers Doctor's degree - other                                    1532 non-null   object
20   Offers Other degree                                               1532 non-null   object
21   Applicants total                                                  1377 non-null   float64
22   Admissions total                                                  1377 non-null   float64
23   Enrolled total                                                    1377 non-null   float64
24   Percent of freshmen submitting SAT scores                         1257 non-null   float64
25   Percent of freshmen submitting ACT scores                         1259 non-null   float64
26   SAT Critical Reading 25th percentile score                        1169 non-null   float64
27   SAT Critical Reading 75th percentile score                        1169 non-null   float64
28   SAT Math 25th percentile score                                    1182 non-null   float64
29   SAT Math 75th percentile score                                    1182 non-null   float64
30   SAT Writing 25th percentile score                                 714 non-null    float64
31   SAT Writing 75th percentile score                                 714 non-null    float64
32   ACT Composite 25th percentile score                               1199 non-null   float64
33   ACT Composite 75th percentile score                               1199 non-null   float64
34   Estimated enrollment, total                                       1532 non-null   float64
35   Estimated enrollment, full time                                   1531 non-null   float64
36   Estimated enrollment, part time                                   1523 non-null   float64
37   Estimated undergraduate enrollment, total                         1526 non-null   float64
38   Estimated undergraduate enrollment, full time                     1525 non-null   float64
39   Estimated undergraduate enrollment, part time                     1514 non-null   float64
40   Estimated freshman undergraduate enrollment, total                1515 non-null   float64
41   Estimated freshman enrollment, full time                          1515 non-null   float64
42   Estimated freshman enrollment, part time                          1451 non-null   float64
43   Estimated graduate enrollment, total                              1432 non-null   float64
44   Estimated graduate enrollment, full time                          1422 non-null   float64
45   Estimated graduate enrollment, part time                          1420 non-null   float64
46   Associate's degrees awarded                                       1532 non-null   float64
47   Bachelor's degrees awarded                                        1532 non-null   float64
48   Master's degrees awarded                                          1532 non-null   float64
49   Doctor's degrese - research/scholarship awarded                   1532 non-null   float64
50   Doctor's degrees - professional practice awarded                  1532 non-null   float64
51   Doctor's degrees - other awarded                                  1532 non-null   float64
52   Certificates of less than 1 year awarded                          1532 non-null   float64
```

```
1 universities_df.isna().sum().sort_values(ascending = False)
```

```
Endowment assets (year end) per FTE enrollment (GASB)      1018
SAT Writing 25th percentile score                           820
SAT Writing 75th percentile score                           820
Number of first-time undergraduates - foreign countries     623
Percent of first-time undergraduates - out-of-state         623
                                                           ...
Level of institution                                         0
Control of institution                                       0
Historically Black College or University                     0
Tribal college                                               0
State abbreviation                                           0
Length: 145, dtype: int64
```

```
1 perc_nan=universities_df.isna().sum()/len(universities_df)*100
2 ax=perc_nan[perc_nan>=20].sort_values(ascending=False).plot.bar(title='Percentage of NaN val
3 ax.set_ylabel('% of NaN elements');
```




```
colum_off = universities_df.isna().sum()[universities_df.isna().sum() >= (0.2 * len(universi
list_colum_off = colum_off.index.to_list()
univclean_df = universities_df.copy()
```

In above step We will be removing columns that contain 20% or more missing values (NaNs). These columns cannot be used as reliable representative values due to the significant amount of missing data. By doing this, we aim to ensure the integrity and accuracy of our data analysis and decision-making process.Before proceeding, I have create a duplicate of the dataframe to preserve the original data while making further modifications

```
univclean_df.drop(list_colum_off,axis = 1,inplace = True )
```

```
# Clean up column names
univclean_df.columns = univclean_df.columns.str.replace(' ', '_').str.lower()

# Now update the interesting_columns list with the cleaned column names
interesting_columns = [
    'name', 'year', 'highest_degree_offered', "offers_bachelor's_degree",
    "offers_master's_degree", "offers_doctor's_degree_-_research/scholarship",
    "offers_doctor's_degree_-_professional_practice", 'applicants_total',
    'admissions_total', 'enrolled_total', 'estimated_enrollment,_total',
    'tuition_and_fees,_2013-14', 'total_price_for_in-state_students_living_on_campus_2013-14
    'total_price_for_out-of-state_students_living_on_campus_2013-14', 'state_abbreviation',
    'control_of_institution', 'full-time_enrollment', 'part-time_enrollment', 'undergraduate
    'graduate_enrollment', 'full-time_undergraduate_enrollment', 'part-time_undergraduate_er
    'percent_of_total_enrollment_that_are_women',
    'percent_of_undergraduate_enrollment_that_are_women',
    'percent_of_graduate_enrollment_that_are_women',
    'graduation_rate_-_bachelor_degree_within_4_years,_total',
    'graduation_rate_-_bachelor_degree_within_5_years,_total',
    'graduation_rate_-_bachelor_degree_within_6_years,_total',
]

# Filter the DataFrame with the updated interesting_columns
univclean_df = univclean_df[interesting_columns]

```

```
# Display rows where 'enrolled_total' has NaN values
enrollment_nan_df = univclean_df[univclean_df['enrolled_total'].isna()][['name', 'applicants
print(enrollment_nan_df)

```

```
                                        name  applicants_total  \
2                           Amridge University               NaN
6                       Athens State University               NaN
10                    Concordia College Alabama               NaN
11                         Faulkner University               NaN
16                              Miles College               NaN
...                                       ...               ...
1526                  University of the West               NaN
1527  Averett University-Non-Traditional Programs               NaN
1528                           LIU Riverhead               NaN
1531                 Ottawa University-Online               NaN
1533  Polytechnic University of Puerto Rico-Orlando               NaN

      admissions_total  enrolled_total  estimated_enrollment,_total
2                  NaN             NaN                        626.0
6                  NaN             NaN                       3175.0
10                 NaN             NaN                        600.0
11                 NaN             NaN                       3127.0
16                 NaN             NaN                       1677.0
...                ...             ...                          ...
1526               NaN             NaN                        294.0
1527               NaN             NaN                       1181.0
1528               NaN             NaN                        219.0
1531               NaN             NaN                        445.0
1533               NaN             NaN                        128.0

[157 rows x 5 columns]
```

```
# Finding the index of the row for 'University of North Georgia'
a = univclean_df[univclean_df['name'] == 'University of North Georgia'].index[0]

# Finding the index of the row for 'Texas A & M University-Galveston'
b = univclean_df[univclean_df['name'] == 'Texas A & M University-Galveston'].index[0]

```

```
univclean_df = univclean_df.drop([a,b],axis =0)
```

```
print(univclean_df.shape[1],universities_df.shape[1],univclean_df.shape[0],universities_df.s
```

```
28 145 1532 1534
```

check for invalid values

```
col = univclean_df.select_dtypes(include = ['float64, int64']).columns
```

```
lt = list()
for i in col:
    y = any(x<0 in univclean_df[i])
    if y is True:
        lt.append(y)
print('{} invalid values in dataframe'.format(len(lt)))

```

```
0 invalid values in dataframe
```

```
univclean_df.describe()
```

	year	applicants_total	admissions_total	enrolled_total	estimated_enrollme
count	1532.0	1377.000000	1377.000000	1377.000000	15
mean	2013.0	6391.193174	3554.880174	1043.046478	7(
std	0.0	8787.469171	4309.457606	1285.165519	9:
min	2013.0	0.000000	0.000000	0.000000	

```
total_zero = univclean_df[univclean_df.loc[0:]==0].count().sum()
```

```
print('{} zero values in this data set'.format(total_zero))
```

```
395 zero values in this data set
```

Giving Format

```
univclean_df.columns
```

```
Index(['name', 'year', 'highest_degree_offered', 'offers_bachelor's_degree',
       'offers_master's_degree',
       'offers_doctor's_degree_-_research/scholarship',
       'offers_doctor's_degree_-_professional_practice', 'applicants_total',
       'admissions_total', 'enrolled_total', 'estimated_enrollment,_total',
       'tuition_and_fees,_2013-14',
       'total_price_for_in-state_students_living_on_campus_2013-14',
       'total_price_for_out-of-state_students_living_on_campus_2013-14',
       'state_abbreviation', 'control_of_institution', 'full-time_enrollment',
       'part-time_enrollment', 'undergraduate_enrollment',
       'graduate_enrollment', 'full-time_undergraduate_enrollment',
       'part-time_undergraduate_enrollment',
       'percent_of_total_enrollment_that_are_women',
       'percent_of_undergraduate_enrollment_that_are_women',
       'percent_of_graduate_enrollment_that_are_women',
       'graduation_rate_-_bachelor_degree_within_4_years,_total',
       'graduation_rate_-_bachelor_degree_within_5_years,_total',
       'graduation_rate_-_bachelor_degree_within_6_years,_total'],
      dtype='object')
```

```
def remove_space(list_headers,characters) : #characters should include ['-',' ']
    new_headers = list()
    for header in list_headers:
        for char in characters:
            if char in header:
                header = header.replace(char,'_')
            header = header
        new_headers.append(header)
    return new_headers
```

```
def remove_sp_char(headers,sp_characters): #special characters include ["'",'/',':'']
    new_headers = list()
    for header in headers:
        for char in sp_characters:
            if char == "-" or char == '/':
                header = header.replace(char,'_')
            if char in header:
                header = header.replace(char,'')
            header = header
        new_headers.append(header)
    return new_headers
```

```
headers = remove_space(univclean_df.columns,[' ',' - '])
```

```
 headers = remove_sp_char(headers,[':',"'",'/',',','-'])
```

```
list_new_headers = list()
for header in headers:
    header = header.casefold() #converts the string to lowercase

    if "degrese" in header: #there is one typo in a column
        header = header.replace("degrese", "degrees")
    list_new_headers.append(header)
univclean_df.columns = list_new_headers
```

```
univclean_df.columns
```

```
Index(['name', 'year', 'highest_degree_offered', 'offers_bachelors_degree',
       'offers_masters_degree', 'offers_doctors_degree___research_scholarship',
       'offers_doctors_degree___professional_practice', 'applicants_total',
       'admissions_total', 'enrolled_total', 'estimated_enrollment_total',
       'tuition_and_fees_2013_14',
       'total_price_for_in_state_students_living_on_campus_2013_14',
       'total_price_for_out_of_state_students_living_on_campus_2013_14',
       'state_abbreviation', 'control_of_institution', 'full_time_enrollment',
       'part_time_enrollment', 'undergraduate_enrollment',
       'graduate_enrollment', 'full_time_undergraduate_enrollment',
       'part_time_undergraduate_enrollment',
       'percent_of_total_enrollment_that_are_women',
       'percent_of_undergraduate_enrollment_that_are_women',
       'percent_of_graduate_enrollment_that_are_women',
       'graduation_rate___bachelor_degree_within_4_years_total',
       'graduation_rate___bachelor_degree_within_5_years_total',
       'graduation_rate___bachelor_degree_within_6_years_total'],
      dtype='object')
```

```
from google.colab import files

file_name = 'cleaned_universities_data.csv'

univclean_df.to_csv(file_name, index=False)


files.download(file_name)
```

Now that we've completed the data cleaning process, we can download the cleaned dataset to uncover valuable insights through Exploratory Data Analysis (EDA). This refined dataset can be seamlessly utilized in Tableau, Power BI, Python, or other tools to derive powerful insights and support data-driven decision-making.

Exploratory Data Analysis

```
univclean_df.columns
```

```
Index(['name', 'year', 'highest_degree_offered', 'offers_bachelors_degree',
       'offers_masters_degree', 'offers_doctors_degree___research_scholarship',
       'offers_doctors_degree___professional_practice', 'applicants_total',
       'admissions_total', 'enrolled_total', 'estimated_enrollment_total',
       'tuition_and_fees_2013_14',
       'total_price_for_in_state_students_living_on_campus_2013_14',
       'total_price_for_out_of_state_students_living_on_campus_2013_14',
       'state_abbreviation', 'control_of_institution', 'full_time_enrollment',
       'part_time_enrollment', 'undergraduate_enrollment',
       'graduate_enrollment', 'full_time_undergraduate_enrollment',
       'part_time_undergraduate_enrollment',
       'percent_of_total_enrollment_that_are_women',
       'percent_of_undergraduate_enrollment_that_are_women',
       'percent_of_graduate_enrollment_that_are_women',
       'graduation_rate___bachelor_degree_within_4_years_total',
       'graduation_rate___bachelor_degree_within_5_years_total',
       'graduation_rate___bachelor_degree_within_6_years_total'],
      dtype='object')
```

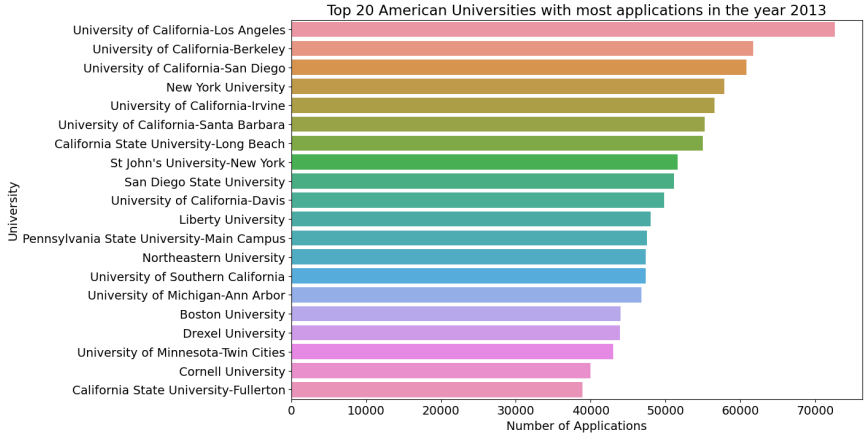
```
univclean_df.describe()
```

	year	applicants_total	admissions_total	enrolled_total	estimated_enrollme
count	1532.0	1377.000000	1377.000000	1377.000000	153
mean	2013.0	6391.193174	3554.880174	1043.046478	703
std	0.0	8787.469171	4309.457606	1285.165519	938
min	2013.0	0.000000	0.000000	0.000000	6
25%	2013.0	1488.000000	950.000000	287.000000	149
50%	2013.0	3350.000000	2056.000000	538.000000	312
75%	2013.0	7016.000000	4203.000000	1246.000000	837
max	2013.0	72676.000000	35815.000000	10241.000000	7957

```
high_app_df = univclean_df[['name','applicants_total']].sort_values('applicants_total', asce
```

```
plt.figure(figsize=(12,8))
matplotlib.rcParams['font.size'] = 14
sns.barplot(x='applicants_total',y='name', data = high_app_df)
plt.title('Top 20 American Universities with most applications in the year 2013')
plt.xlabel('Number of Applications')
plt.ylabel('University')
```

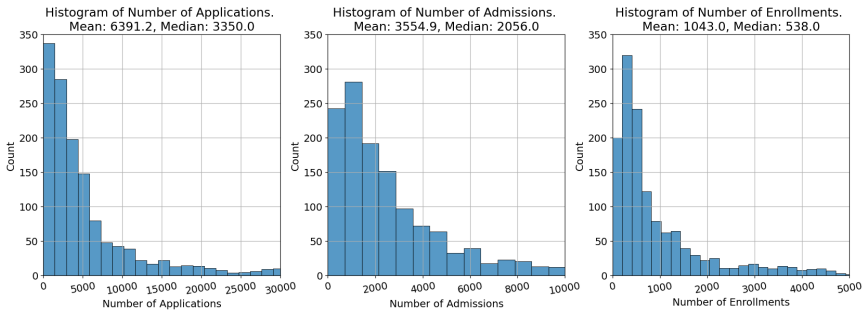
```
Text(0, 0.5, 'University')
```



In this above analysis, we aim to determine whether universities with the highest number of applications are also the most preferred ones by students. While it might seem evident that more applications indicate higher preference, it is essential to consider that one student can submit multiple applications simultaneously. Therefore, we will explore the relationships between the number of applications, admissions, and enrollments to gain insights into students' actual preferences for each university

```
plt.figure(figsize=(20,6))


plt.subplot(1,3,1)
sns.histplot(univclean_df.applicants_total,bins=50)
```

```
plt.title('''Histogram of Number of Applications.
Mean: {:.1f}, Median: {:.1f}'''.format(univclean_df.applicants_total.mean(),univclean_df.app
plt.xlabel('Number of Applications')
plt.axis([0,30000,0,350])
plt.xticks(rotation=10)
plt.grid()

plt.subplot(1,3,2)
sns.histplot(univclean_df.admissions_total,bins=50)
plt.title('''Histogram of Number of Admissions.
Mean: {:.1f}, Median: {:.1f}'''.format(univclean_df.admissions_total.mean(),univclean_df.adm
plt.xlabel('Number of Admissions')
plt.axis([0,10000,0,350])
plt.xticks(rotation=10)
plt.grid()

plt.subplot(1,3,3)
sns.histplot(univclean_df.enrolled_total,bins=50)
plt.title('''Histogram of Number of Enrollments.
Mean: {:.1f}, Median: {:.1f}'''.format(univclean_df.enrolled_total.mean(),univclean_df.enrol
plt.xlabel('Number of Enrollments')
plt.axis([0,5000,0,350])
plt.xticks(rotation=10)
plt.grid()
```



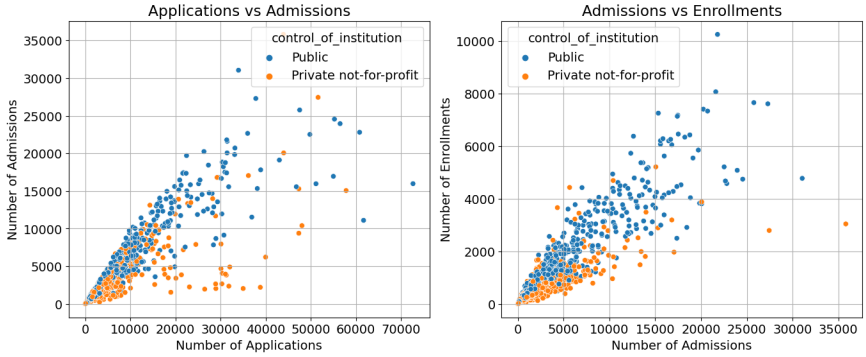
the dataset reveals variations in the number of applications and admissions among different universities, with some institutions attracting a larger pool of applicants and others accepting a more substantial number of students. The high rate of admissions in certain universities could be an appealing factor for prospective students looking to increase their chances of acceptance.

Q. Do students prefer universities with high number of applications?

Q. Do students prefer universities with high admission rate?

```
plt.figure(figsize=(16,6))

plt.subplot(1,2,1)
plt.title('Applications vs Admissions')
sns.scatterplot(y=univclean_df.admissions_total, x=univclean_df.applicants_total, hue= univc
plt.xlabel('Number of Applications')
```

```
plt.ylabel('Number of Admissions')
plt.grid()

plt.subplot(1,2,2)
plt.title('Admissions vs Enrollments')
sns.scatterplot(x=univclean_df.admissions_total, y=univclean_df.enrolled_total, hue= univcle
plt.xlabel('Number of Admissions')
plt.ylabel('Number of Enrollments')
plt.grid()
```



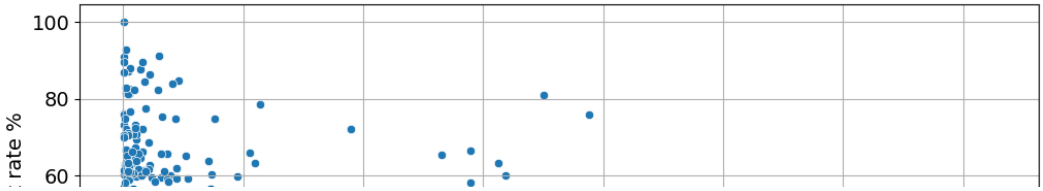
From the first scatterplot, we notice a general trend where universities with a high number of applications also tend to have a high rate of admissions, and vice versa. However, there are exceptions, particularly among some private universities that receive many applications but admit only a small number of students, indicating a low acceptance rate and a more selective admissions process.

The second scatterplot reveals a positive correlation between the number of admissions and the enrollment rate. Universities with higher admissions numbers generally have higher enrollment rates, suggesting that a significant proportion of admitted students choose to enroll.

To gain more meaningful insights, we will calculate the admission rate and enrollment rate for universities based on the provided data.

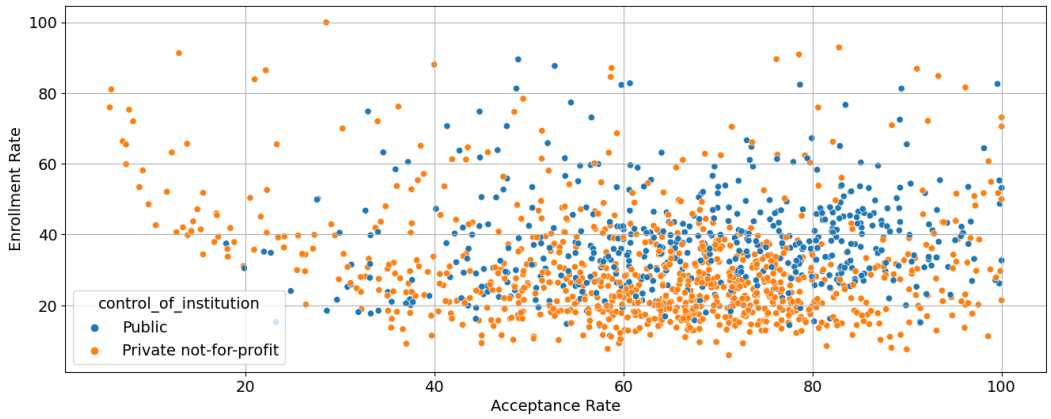
```
univclean_df['acceptance_rate'] =(univclean_df.admissions_total/univclean_df.applicants_tota
univclean_df['enrollment_rate'] =(univclean_df.enrolled_total/univclean_df.admissions_total*
```

```
plt.figure(figsize = (12,5))
sns.scatterplot(x ='applicants_total' , y = 'enrollment_rate' , data = univclean_df)
plt.xlabel('Number of Applications')
plt.ylabel('Enrollment rate %')
plt.grid()
```



The figure illustrates a strong tendency where universities with a lower number of applications often have a higher enrollment rate. While there are exceptions, this pattern suggests that a high number of applications does not necessarily indicate that a university is preferred among students.

```
plt.figure(figsize = (16,6))
sns.scatterplot(x = univclean_df.acceptance_rate,y = univclean_df.enrollment_rate, hue = uni
plt.xlabel('Acceptance Rate')
plt.ylabel('Enrollment Rate')
plt.grid()
```



Upon analyzing the data, it becomes evident that for universities with high acceptance rates, the enrollment rates show considerable variation. This suggests that acceptance rate might not strongly influence students' preferences when choosing a university.

However, a different pattern emerges for universities with low acceptance rates. In this bracket, the enrollment rate is consistently above 40%, and most of the institutions in this category are private universities. This observation leads to the conclusion that most public colleges tend to have acceptance rates higher than 50%.

Furthermore, it is notable that when students are accepted into universities with low acceptance rates, they tend to enroll straightforwardly, without significant fluctuations in enrollment rates.

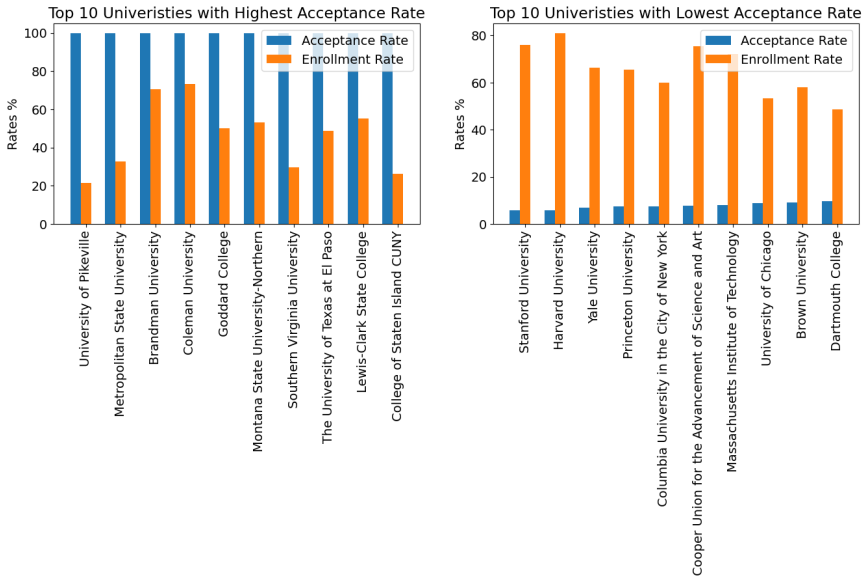
In summary, while high acceptance rates may not significantly impact student preferences, low acceptance rates seem to influence students' decisions, especially when it comes to private universities where acceptance is competitive but enrollment rates remain relatively high. Further analysis will be conducted to explore this trend in more detail

```
high_acceptance = univclean_df[univclean_df.acceptance_rate.notnull()][['name','acceptance_r
low_acceptance = univclean_df[univclean_df.acceptance_rate.notnull()][['name','acceptance_ra
```

```
plt.figure(figsize = (16,4))

plt.subplot(1,2,1)
ind = np.arange(len(high_acceptance))
width = 0.30

plt.bar(ind, high_acceptance.acceptance_rate, width, label = 'Acceptance Rate')
```

```
plt.bar(ind+ width, high_acceptance.enrollment_rate, width, label = 'Enrollment Rate')
plt.title('Top 10 Univeristies with Highest Acceptance Rate')
plt.ylabel('Rates %')
plt.xticks(ind + width, high_acceptance.name.values, rotation = 90)
plt.legend(loc = 'best')

plt.subplot(1,2,2)
ind = np.arange(len(low_acceptance))
width = 0.30

plt.bar(ind, low_acceptance.acceptance_rate, width, label = 'Acceptance Rate')
plt.bar(ind+ width, low_acceptance.enrollment_rate, width, label = 'Enrollment Rate')
plt.title('Top 10 Univeristies with Lowest Acceptance Rate')
plt.ylabel('Rates %')
plt.xticks(ind + width, low_acceptance.name.values, rotation = 90)
plt.legend(loc = 'best')
```

```
<matplotlib.legend.Legend at 0x7bcd302d2d10>
```



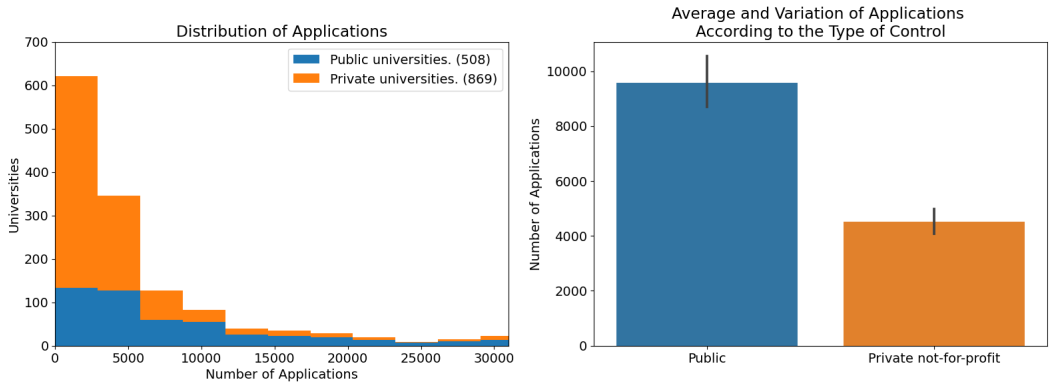
We see that there is no clear pattern of enrollmenr rates of universitites having high admission rates.

We can be sure that students' preference is not based on how easy it is for them to to be admitted..

Q. Do students prefer public or private universities?

```
#splitting number of applications according to their type of control : public or private
uni_private = univclean_df[univclean_df.control_of_institution == 'Private not-for-profit']
uni_private = uni_private[uni_private.applicants_total.notnull()]

uni_public = univclean_df[univclean_df.control_of_institution == 'Public']
uni_public = uni_public[uni_public.applicants_total.notnull()]
```

```
plt.figure(figsize = (16,6))

plt.subplot(1,2,1)
plt.hist([uni_public.applicants_total, uni_private.applicants_total], stacked = True, bins =
plt.axis([0,31000,0,700])
plt.title('Distribution of Applications')
plt.xlabel('Number of Applications')
plt.ylabel('Universities')
plt.legend(['Public universities. ({})'.format(len(uni_public)),'Private universities. ({})'

plt.subplot(1,2,2)
sns.barplot(x=univclean_df.control_of_institution,y=univclean_df.applicants_total);
plt.title('''Average and Variation of Applications
According to the Type of Control''')
plt.xlabel('')
plt.ylabel('Number of Applications');
plt.tight_layout(pad=1)
```



The dataset reveals that in 2013, public universities received an average of around 9000 applications, which is nearly double the average of approximately 4000 applications received by private universities. It's essential to note that the dataset includes 508 public universities and 868 private universities.

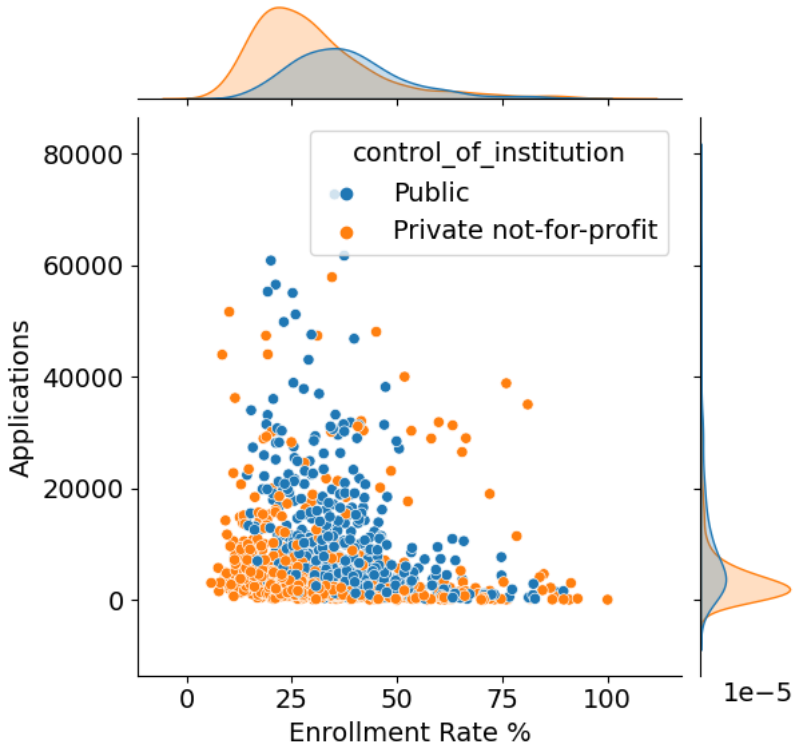
Additionally, the figure on the right indicates that the variation in the number of applications to public universities is significantly higher than that of private universities. Despite this variation, the average number of applications for public universities remains notably higher than that for private universities.

To further investigate this trend, we will determine the maximum and minimum values of applications for both public and private universities

```
print('The minimum number of applications for private universities was {}; whereas, for publ
print('The maximum number of applications for private universities was {}; whereas, for publ

```

```
The minimum number of applications for private universities was 0; whereas, for public universities was 193.
The maximum number of applications for private universities was 57845; whereas, for public universities was 72676.
```

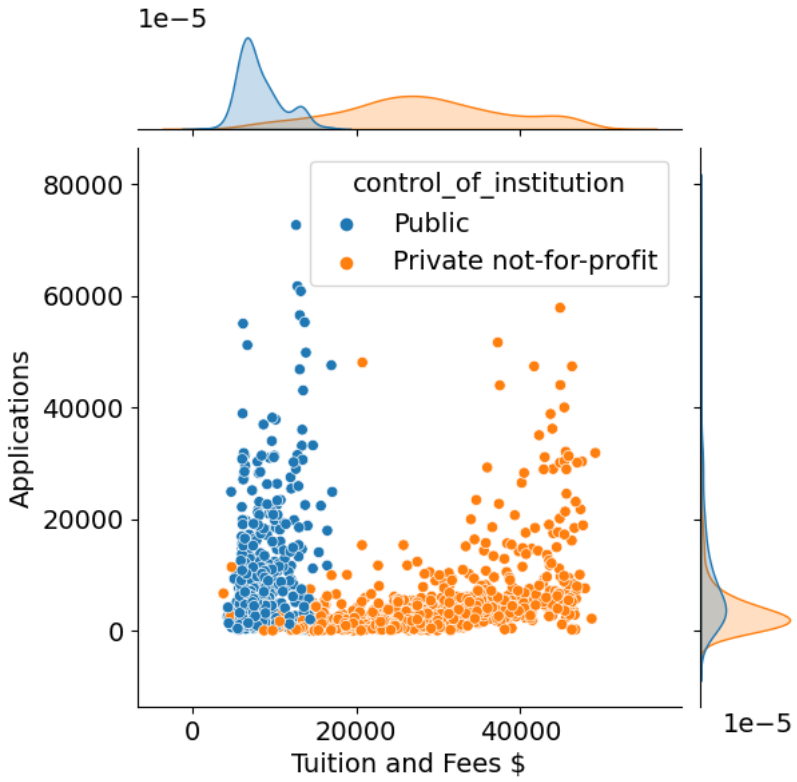
```
g=sns.jointplot(x=univclean_df.enrollment_rate,y=univclean_df.applicants_total,hue=univclear
g=(g.set_axis_labels("Enrollment Rate %","Applications"));
```



By examining the figure, it becomes apparent that, on average, the enrollment rate is higher for public universities compared to private universities. Considering this finding along with the earlier observations, we can confidently conclude that there is indeed a preference among students for public universities

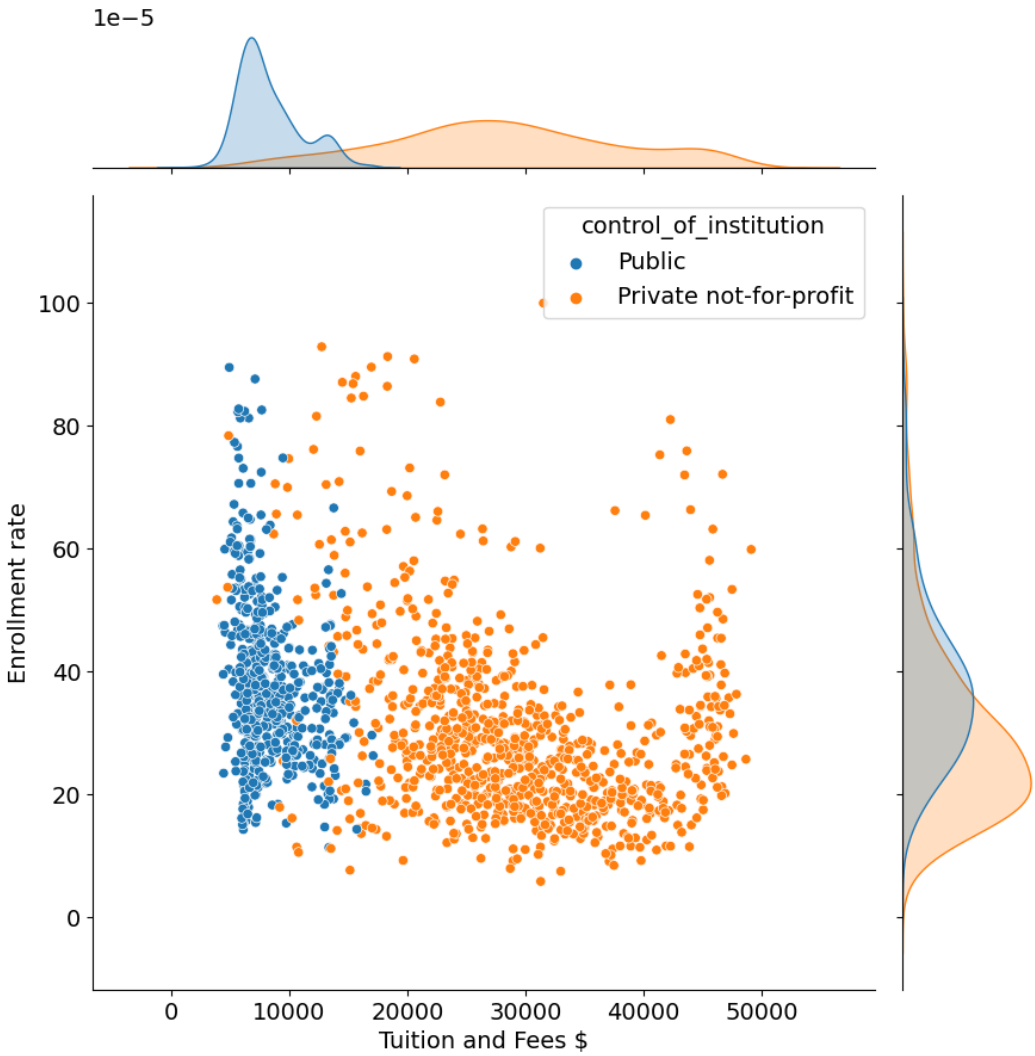
Q: Do students prefer universities with low tuition and fees?

```
g=sns.jointplot(x=univclean_df.tuition_and_fees_2013_14,y=univclean_df.applicants_total,hue=
g=(g.set_axis_labels("Tuition and Fees $","Applications"))
```



Upon reviewing the figure, it becomes evident that the most affordable universities, in terms of tuition and fees, receive the highest number of applications. This observation leads us to consider the possibility of a relationship between tuition costs and students' preferences. To ascertain a definitive conclusion, we will further analyze the data by examining the relationship between tuition and fees and the enrollment rates of universities

```
g=sns.jointplot(x=univclean_df.tuition_and_fees_2013_14,y=univclean_df.enrollment_rate,hue=u
g=(g.set_axis_labels('Tuition and Fees $','Enrollment rate'))
```



The figure provides valuable insights into students' preferences regarding university choices based on tuition and fees. It highlights two crucial points:

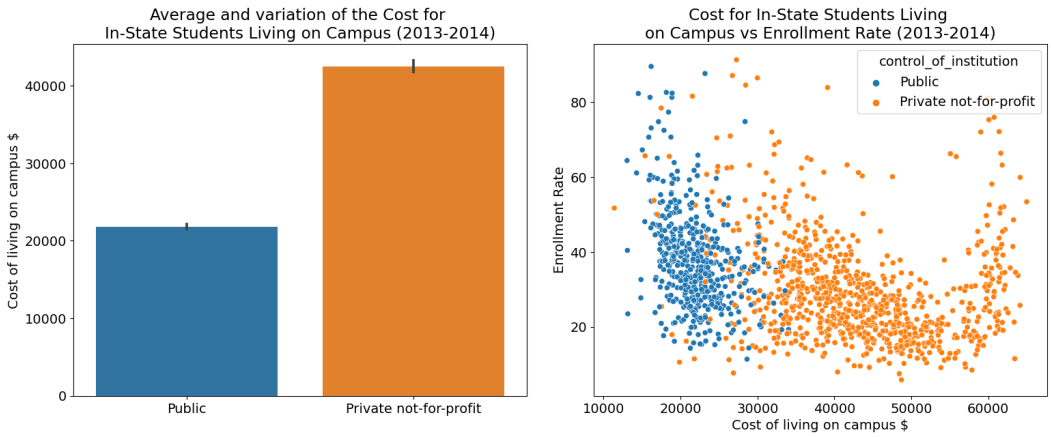
Public universities tend to be more affordable compared to the majority of private universities. This cost advantage makes public universities an attractive option for many students.

There is a higher concentration of universities towards the left of the figure, indicating that more institutions have lower tuition and fees. Within this region, universities with more affordable tuition fees tend to have higher enrollment rates than the more expensive ones. This finding suggests that affordability plays a significant role in students' decisions, as they prefer universities with lower costs, leading to higher enrollment rates in those institutions.

Combining the insights from the previous findings, we can conclude that one of the reasons for students' preference for public universities is their affordability compared to many private universities. This data-driven conclusion sheds light on the factors influencing students' choices and provides valuable information for educational institutions to understand student preferences better.

Q: Do students prefer a university for its low cost of on-campus living?.

```
plt.figure(figsize=(16,7))
plt.subplot(1,2,1)
sns.barplot(y=univclean_df.total_price_for_in_state_students_living_on_campus_2013_14,x=univ
plt.title('''Average and variation of the Cost for
In-State Students Living on Campus (2013-2014)''')
plt.xlabel('')
plt.ylabel('Cost of living on campus $')

plt.subplot(1,2,2)
sns.scatterplot(x=univclean_df.total_price_for_in_state_students_living_on_campus_2013_14,y=
plt.title('''Cost for In-State Students Living
on Campus vs Enrollment Rate (2013-2014)''')
plt.xlabel('Cost of living on campus $')
plt.ylabel('Enrollment Rate')
plt.tight_layout(pad=2)
```



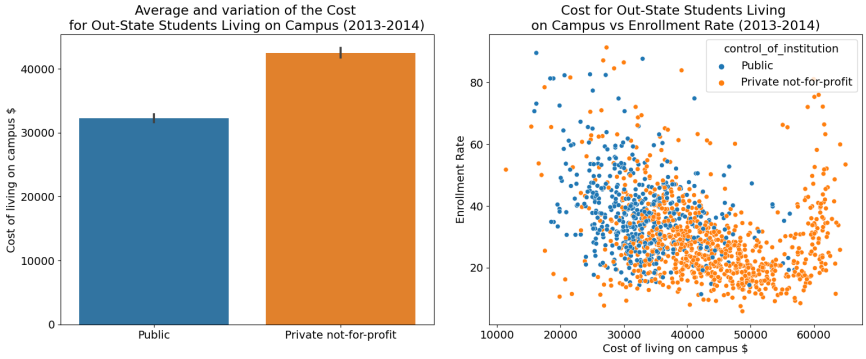
The data illustrates that the majority of public universities offer a substantially more affordable cost for in-state students compared to private universities. On average, the cost at public universities is approximately half that of private universities, making them a more budget-friendly option for many students.

The right figure further reinforces the importance of affordability in students' decisions. It shows a strong correlation between high enrollment rates and more affordable costs. This indicates that students do consider the cost of living on campus for in-state students as a crucial factor when selecting a university. Universities offering a more affordable cost of living tend to attract higher enrollment rates, suggesting that cost plays a significant role in shaping students' preferences.

By understanding the cost dynamics and its influence on student choices, educational institutions can better strategize their tuition and fees to appeal to prospective students and make informed decisions to improve student recruitment and enrollment rates

```
plt.figure(figsize=(16,7))
plt.subplot(1,2,2)
sns.scatterplot(x=univclean_df.total_price_for_out_of_state_students_living_on_campus_2013_1
plt.title('''Cost for Out-State Students Living
on Campus vs Enrollment Rate (2013-2014)''')
plt.xlabel('Cost of living on campus $')
plt.ylabel('Enrollment Rate')

plt.subplot(1,2,1)
sns.barplot(y=univclean_df.total_price_for_out_of_state_students_living_on_campus_2013_14,x=
plt.title('''Average and variation of the Cost
```

```
for Out-State Students Living on Campus (2013-2014)''')
plt.xlabel('')
plt.ylabel('Cost of living on campus $')

plt.tight_layout(pad=2)
```



The data shows that, on average, the cost for out-state students at public universities is lower than that of private universities, once again highlighting the cost advantage of public institutions for both in-state and out-state students.

The right figure reaffirms the importance of affordability for out-state students as well. Just like in the case of in-state students, there is a strong correlation between high enrollment rates and more affordable costs for out-state students. This finding indicates that students, irrespective of their state residency, consider the cost of living on campus for out-state students when making their university choices.

Overall, these insights underline the significance of affordability in shaping students' preferences, regardless of whether they are in-state or out-state students. Universities offering more affordable options for both tuition and cost of living tend to attract higher enrollment rates, reinforcing the notion that cost plays a critical role in students' decision-making processe

```
1
```

can explore more ..... as per the information or insights we required. this is what data can do with the usage

Thankyou all,

YRC-RAKESH