Denny Li

In [84]:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import sqlite3, csv
import seaborn as sns
import scipy.stats as stats
```

In [85]:

```
Chicago_School = pd.read_csv('Chicago_school.csv')
```

In [86]:

```
Chicago_School
```

Out[86]:

	School ID	NAME_OF_SCHOOL	Elementary, Middle, or High School	Street Address	City	State	ZIP Code	Phone Number	
0	610038	Abraham Lincoln Elementary School	ES	615 W Kemper Pl	Chicago	IL	60614	(773) 534-5720	http://schoolreports.cps.edu/S
1	610281	Adam Clayton Powell Paideia Community Academy ...	ES	7511 S South Shore Dr	Chicago	IL	60649	(773) 535-6650	http://schoolreports.cps.edu/S
2	610185	Adlai E Stevenson Elementary School	ES	8010 S Kostner Ave	Chicago	IL	60652	(773) 535-2280	http://schoolreports.cps.edu/S
3	609993	Agustin Lara Elementary Academy	ES	4619 S Wolcott Ave	Chicago	IL	60609	(773) 535-4389	http://schoolreports.cps.edu/S
4	610513	Air Force Academy High School	HS	3630 S Wells St	Chicago	IL	60609	(773) 535-1590	http://schoolreports.cps.edu/S
...	...	...	...	...	...	...	...	...	
561	610172	William T Sherman Elementary School	ES	1000 W 52nd St	Chicago	IL	60609	(773) 535-1757	http://schoolreports.cps.edu/S
562	609844	William W Carter Elementary School	ES	5740 S Michigan Ave	Chicago	IL	60637	(773) 535-0860	http://schoolreports.cps.edu/S
563	610088	Wolfgang A Mozart Elementary School	ES	2200 N Hamlin Ave	Chicago	IL	60647	(773) 534-4160	http://schoolreports.cps.edu/S
564	609977	Woodlawn Community Elementary School	ES	6657 S Kimbark Ave	Chicago	IL	60637	(773) 535-0801	http://schoolreports.cps.edu/S

	School ID	NAME_OF_SCHOOL	Elementary, Middle, or High School	Street Address	City	State	ZIP Code	Phone Number	
565	610392	World Language Academy High School	HS	3120 S Kostner Ave	Chicago	IL	60623	(773) 535-4334	http://schoolreports.cps.edu/S

566 rows × 78 columns

In [87]:
```
missing_values = ["NDA"]
```

In [88]:
```
Chicago_School1=pd.read_csv('Chicago_school.csv', na_values=missing_values)
```

In [89]:
```
Chicago_School1
```

Out[89]:

	School ID	NAME_OF_SCHOOL	Elementary, Middle, or High School	Street Address	City	State	ZIP Code	Phone Number	
0	610038	Abraham Lincoln Elementary School	ES	615 W Kemper Pl	Chicago	IL	60614	(773) 534-5720	http://schoolreports.cps.edu/S
1	610281	Adam Clayton Powell Paideia Community Academy ...	ES	7511 S South Shore Dr	Chicago	IL	60649	(773) 535-6650	http://schoolreports.cps.edu/S
2	610185	Adlai E Stevenson Elementary School	ES	8010 S Kostner Ave	Chicago	IL	60652	(773) 535-2280	http://schoolreports.cps.edu/S
3	609993	Agustin Lara Elementary Academy	ES	4619 S Wolcott Ave	Chicago	IL	60609	(773) 535-4389	http://schoolreports.cps.edu/S
4	610513	Air Force Academy High School	HS	3630 S Wells St	Chicago	IL	60609	(773) 535-1590	http://schoolreports.cps.edu/S
...	...	...	...	...	...	...	...	...	
561	610172	William T Sherman Elementary School	ES	1000 W 52nd St	Chicago	IL	60609	(773) 535-1757	http://schoolreports.cps.edu/S
562	609844	William W Carter Elementary School	ES	5740 S Michigan Ave	Chicago	IL	60637	(773) 535-0860	http://schoolreports.cps.edu/S
563	610088	Wolfgang A Mozart Elementary School	ES	2200 N Hamlin Ave	Chicago	IL	60647	(773) 534-4160	http://schoolreports.cps.edu/S

	School ID	NAME_OF_SCHOOL	Elementary, Middle, or High School	Street Address	City	State	ZIP Code	Phone Number	
564	609977	Woodlawn Community Elementary School	ES	6657 S Kimbark Ave	Chicago	IL	60637	(773) 535-0801	http://schoolreports.cps.edu/S
565	610392	World Language Academy High School	HS	3120 S Kostner Ave	Chicago	IL	60623	(773) 535-4334	http://schoolreports.cps.edu/S

566 rows × 78 columns

In [90]:

```
Chicago_School1.isna().sum()
```

Out[90]:

```
School ID                              0
NAME_OF_SCHOOL                         0
Elementary, Middle, or High School     0
Street Address                         0
City                                   0
                                      ..
COMMUNITY_AREA_NUMBER                  0
COMMUNITY_AREA_NAME                    0
Ward                                   0
Police District                        0
Location                               0
Length: 78, dtype: int64
```

In [91]:

```
Chicago_School1.sort_values(by='Freshman on Track Rate %', ascending=False, na_position='f
```

Out[91]:

	School ID	NAME_OF_SCHOOL	Elementary, Middle, or High School	Street Address	City	State	ZIP Code	Phone Number	
0	610038	Abraham Lincoln Elementary School	ES	615 W Kemper Pl	Chicago	IL	60614	(773) 534-5720	http://schoolreports.cps.edu/S
1	610281	Adam Clayton Powell Paideia Community Academy ...	ES	7511 S South Shore Dr	Chicago	IL	60649	(773) 535-6650	http://schoolreports.cps.edu/S
2	610185	Adlai E Stevenson Elementary School	ES	8010 S Kostner Ave	Chicago	IL	60652	(773) 535-2280	http://schoolreports.cps.edu/S
3	609993	Agustin Lara Elementary Academy	ES	4619 S Wolcott Ave	Chicago	IL	60609	(773) 535-4389	http://schoolreports.cps.edu/S
5	610212	Albany Park Multicultural Academy	MS	4929 N Sawyer Ave	Chicago	IL	60625	(773) 534-5108	http://schoolreports.cps.edu/S
...	...	...	...	...	...	...	...	...	

	School ID	NAME_OF_SCHOOL	Elementary, Middle, or High School	Street Address	City	State	ZIP Code	Phone Number	
85	609674	Chicago Vocational Career Academy High School	HS	2100 E 87th St	Chicago	IL	60617	(773) 535-6100	http://schoolreports.cps.edu/S
372	610535	Mason High School	HS	4217 W 18th St	Chicago	IL	60623	(773) 534-1530	http://schoolreports.cps.edu/S
451	609702	Richard T Crane Technical Preparatory High School	HS	2245 W Jackson Blvd	Chicago	IL	60612	(773) 534-7550	http://schoolreports.cps.edu/S
117	609736	Dyett High School	HS	555 E 51st St	Chicago	IL	60615	(773) 535-1825	http://schoolreports.cps.edu/S
526	609727	Wendell Phillips Academy High School	HS	244 E Pershing Rd	Chicago	IL	60653	(773) 535-1603	http://schoolreports.cps.edu/S

566 rows × 78 columns

In [92]:

```
Census=pd.read_csv('Census_data.csv')
```

In [93]:

```
Census
```

Out[93]:

	COMMUNITY_AREA_NUMBER	COMMUNITY_AREA_NAME	PERCENT OF HOUSING CROWDED	PERCENT HOUSEHOLDS BELOW POVERTY	PERCENT AGED 16+ UNEMPLOYED	PERCENT AGED 25+ WITHOUT HIGH SCHOOL DIPLOMA
0	1.0	Rogers Park	7.7	23.6	8.7	18.2
1	2.0	West Ridge	7.8	17.2	8.8	20.8
2	3.0	Uptown	3.8	24.0	8.9	11.8
3	4.0	Lincoln Square	3.4	10.9	8.2	13.4
4	5.0	North Center	0.3	7.5	5.2	4.5
...	...	...	...	...	...	...
73	74.0	Mount Greenwood	1.0	3.4	8.7	4.3
74	75.0	Morgan Park	0.8	13.2	15.0	10.8
75	76.0	O'Hare	3.6	15.4	7.1	10.9
76	77.0	Edgewater	4.1	18.2	9.2	9.7
77	NaN	CHICAGO	4.7	19.7	12.9	19.5

78 rows × 9 columns

In [ ]:

First Task: Understanding and Cleaning the Data

Chicago Public Schools Dataset

In [94]:

```
Chicago_School1.dtypes
```

Out[94]:

```
School ID                                int64
NAME_OF_SCHOOL                          object
Elementary, Middle, or High School      object
Street Address                          object
City                                    object
                                         ...
COMMUNITY_AREA_NUMBER                    int64
COMMUNITY_AREA_NAME                     object
Ward                                     int64
Police District                          int64
Location                                object
Length: 78, dtype: object
```

In [95]:

```
Chicago_School1.shape
```

Out[95]:

```
(566, 78)
```

In [96]:

```
Chicago_School1.isnull()
```

Out[96]:

	School ID	NAME_OF_SCHOOL	Elementary, Middle, or High School	Street Address	City	State	ZIP Code	Phone Number	Link	Network Manager	...	Freshman on Track Rate %
0	False	False	False	False	False	False	False	False	False	False	...	True
1	False	False	False	False	False	False	False	False	False	False	...	True
2	False	False	False	False	False	False	False	False	False	False	...	True
3	False	False	False	False	False	False	False	False	False	False	...	True
4	False	False	False	False	False	False	False	False	False	False	...	False
...	...	...	...	...	...	...	...	...	...	...	...	...
561	False	False	False	False	False	False	False	False	False	False	...	True
562	False	False	False	False	False	False	False	False	False	False	...	True
563	False	False	False	False	False	False	False	False	False	False	...	True
564	False	False	False	False	False	False	False	False	False	False	...	True
565	False	False	False	False	False	False	False	False	False	False	...	False

566 rows × 78 columns

```
In [97]: Chicago_School1.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 566 entries, 0 to 565
Data columns (total 78 columns):
 #   Column                                             Non-Null Count  Dtype
---  ------                                             --------------  -----
 0   School ID                                          566 non-null    int64
 1   NAME_OF_SCHOOL                                     566 non-null    object
 2   Elementary, Middle, or High School                 566 non-null    object
 3   Street Address                                     566 non-null    object
 4   City                                               566 non-null    object
 5   State                                              566 non-null    object
 6   ZIP Code                                           566 non-null    int64
 7   Phone Number                                       566 non-null    object
 8   Link                                               565 non-null    object
 9   Network Manager                                    566 non-null    object
 10  Collaborative Name                                 566 non-null    object
 11  Adequate Yearly Progress Made?                     548 non-null    object
 12  Track Schedule                                     566 non-null    object
 13  CPS Performance Policy Status                      561 non-null    object
 14  CPS Performance Policy Level                       561 non-null    object
 15  HEALTHY_SCHOOL_CERTIFIED                           566 non-null    object
 16  Safety Icon                                        513 non-null    object
 17  SAFETY_SCORE                                       513 non-null    float64
 18  Family Involvement Icon                            297 non-null    object
 19  Family Involvement Score                           297 non-null    float64
 20  Environment Icon                                   513 non-null    object
 21  Environment Score                                  513 non-null    float64
 22  Instruction Icon                                   513 non-null    object
 23  Instruction Score                                  513 non-null    float64
 24  Leaders Icon                                       295 non-null    object
 25  Leaders Score                                      295 non-null    float64
 26  Teachers Icon                                      295 non-null    object
 27  Teachers Score                                     295 non-null    float64
 28  Parent Engagement Icon                             432 non-null    object
 29  Parent Engagement Score                            432 non-null    float64
 30  Parent Environment Icon                            432 non-null    object
 31  Parent Environment Score                           432 non-null    float64
 32  AVERAGE_STUDENT_ATTENDANCE                         565 non-null    object
 33  Rate of Misconducts (per 100 students)             566 non-null    float64
 34  Average Teacher Attendance                         566 non-null    object
 35  Individualized Education Program Compliance Rate   566 non-null    object
 36  Pk-2 Literacy %                                    403 non-null    float64
 37  Pk-2 Math %                                        341 non-null    float64
 38  Gr3-5 Grade Level Math %                           452 non-null    float64
 39  Gr3-5 Grade Level Read %                           452 non-null    float64
 40  Gr3-5 Keep Pace Read %                             452 non-null    float64
 41  Gr3-5 Keep Pace Math %                             452 non-null    float64
 42  Gr6-8 Grade Level Math %                           444 non-null    float64
 43  Gr6-8 Grade Level Read %                           445 non-null    float64
 44  Gr6-8 Keep Pace Math%                              444 non-null    float64
 45  Gr6-8 Keep Pace Read %                             445 non-null    float64
 46  Gr-8 Explore Math %                                425 non-null    float64
 47  Gr-8 Explore Read %                                425 non-null    float64
 48  ISAT Exceeding Math %                              476 non-null    float64
 49  ISAT Exceeding Reading %                           476 non-null    float64
 50  ISAT Value Add Math                                468 non-null    float64
 51  ISAT Value Add Read                                468 non-null    float64
 52  ISAT Value Add Color Math                          468 non-null    object
 53  ISAT Value Add Color Read                          468 non-null    object
 54  Students Taking  Algebra %                         158 non-null    float64
 55  Students Passing  Algebra %                        130 non-null    float64
 56  9th Grade EXPLORE (2009)                           89 non-null     float64
 57  9th Grade EXPLORE (2010)                           88 non-null     float64
 58  10th Grade PLAN (2009)                           82 non-null     float64
 59  10th Grade PLAN (2010)                           87 non-null     float64
 60  Net Change EXPLORE and PLAN                      87 non-null     float64
 61  11th Grade Average ACT (2011)                    83 non-null     float64
 62  Net Change PLAN and ACT                          82 non-null     float64
 63  College Eligibility %                            86 non-null     float64
 64  Graduation Rate %                                79 non-null     float64
 65  College Enrollment Rate %                        77 non-null     float64
 66  COLLEGE_ENROLLMENT                               566 non-null    int64
 67  General Services Route                           566 non-null    int64
 68  Freshman on Track Rate %                         87 non-null     float64
 69  X_COORDINATE                                     566 non-null    float64
 70  Y_COORDINATE                                     566 non-null    float64
 71  Latitude                                         566 non-null    float64
 72  Longitude                                        566 non-null    float64
 73  COMMUNITY_AREA_NUMBER                            566 non-null    int64
 74  COMMUNITY_AREA_NAME                              566 non-null    object
 75  Ward                                             566 non-null    int64
 76  Police District                                  566 non-null    int64
 77  Location                                         566 non-null    object
dtypes: float64(42), int64(7), object(29)
memory usage: 345.0+ KB
```

In [98]:

```
Chicago_School1.isna().sum()
```

Out[98]:

```
School ID                              0
NAME_OF_SCHOOL                         0
Elementary, Middle, or High School     0
Street Address                         0
City                                   0
                                      ..
COMMUNITY_AREA_NUMBER                  0
COMMUNITY_AREA_NAME                    0
Ward                                   0
Police District                        0
Location                               0
Length: 78, dtype: int64
```

In [99]:

```
Chicago_School1.describe()
```

Out[99]:

	School ID	ZIP Code	SAFETY_SCORE	Family Involvement Score	Environment Score	Instruction Score	Leaders Score	Teachers Score
count	566.000000	566.000000	513.000000	297.000000	513.000000	513.000000	295.000000	295.000000
mean	609681.791519	60629.727915	49.504873	50.602694	47.766082	48.288499	49.888136	49.050847
std	8831.060946	20.248024	20.110837	18.549160	16.215584	17.417176	18.276905	17.854499
min	400018.000000	60605.000000	1.000000	6.000000	1.000000	1.000000	4.000000	6.000000
25%	609873.250000	60618.000000	35.000000	37.000000	37.000000	37.000000	35.500000	36.000000
50%	610052.500000	60625.000000	48.000000	49.000000	47.000000	47.000000	49.000000	48.000000
75%	610212.750000	60639.000000	61.000000	61.000000	58.000000	59.000000	62.000000	59.000000
max	610544.000000	60827.000000	99.000000	99.000000	99.000000	99.000000	99.000000	99.000000

8 rows × 49 columns

In [100…

```
Chicago_School1.head()
```

Out[100…

	School ID	NAME_OF_SCHOOL	Elementary, Middle, or High School	Street Address	City	State	ZIP Code	Phone Number	
0	610038	Abraham Lincoln Elementary School	ES	615 W Kemper Pl	Chicago	IL	60614	(773) 534-5720	http://schoolreports.cps.edu/Scho
1	610281	Adam Clayton Powell Paideia Community Academy ...	ES	7511 S South Shore Dr	Chicago	IL	60649	(773) 535-6650	http://schoolreports.cps.edu/Scho
2	610185	Adlai E Stevenson Elementary School	ES	8010 S Kostner Ave	Chicago	IL	60652	(773) 535-2280	http://schoolreports.cps.edu/Scho
3	609993	Agustin Lara Elementary Academy	ES	4619 S Wolcott Ave	Chicago	IL	60609	(773) 535-4389	http://schoolreports.cps.edu/Scho
4	610513	Air Force Academy High School	HS	3630 S Wells St	Chicago	IL	60609	(773) 535-1590	http://schoolreports.cps.edu/Scho

5 rows × 78 columns

The Chicago public schools progress report data was collected between 2011 and 2012. It consists of 566 rows and 78 columns. It contains various numerical and categorical variables about student's academic performance such as literacy in math and reading, ACT score, college enrollment, average student attendence, college enrollment and etc. These indicators will be used to access the academic performance of students in Chicago public schools.

Census dataset for socioeconomic indicators for Chicago

In [101…

```
Census.dtypes
```

Out[101…

```
COMMUNITY_AREA_NUMBER                           float64
COMMUNITY_AREA_NAME                              object
PERCENT OF HOUSING CROWDED                      float64
PERCENT HOUSEHOLDS BELOW POVERTY                float64
PERCENT AGED 16+ UNEMPLOYED                     float64
PERCENT AGED 25+ WITHOUT HIGH SCHOOL DIPLOMA    float64
PERCENT AGED UNDER 18 OR OVER 64                float64
PER_CAPITA_INCOME                                 int64
HARDSHIP_INDEX                                  float64
dtype: object
```

In [102…

```
Census.shape
```

Out[102…

```
(78, 9)
```

In [103…

```
Census.columns
```

Out[103…

```
Index(['COMMUNITY_AREA_NUMBER', 'COMMUNITY_AREA_NAME',
       'PERCENT OF HOUSING CROWDED', 'PERCENT HOUSEHOLDS BELOW POVERTY',
       'PERCENT AGED 16+ UNEMPLOYED',
       'PERCENT AGED 25+ WITHOUT HIGH SCHOOL DIPLOMA',
       'PERCENT AGED UNDER 18 OR OVER 64', 'PER_CAPITA_INCOME ',
       'HARDSHIP_INDEX'],
      dtype='object')
```

In [104…

```
Census.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 78 entries, 0 to 77
Data columns (total 9 columns):
 #   Column                                        Non-Null Count  Dtype  
---  ------                                        --------------  -----  
 0   COMMUNITY_AREA_NUMBER                         77 non-null     float64
 1   COMMUNITY_AREA_NAME                           78 non-null     object 
 2   PERCENT OF HOUSING CROWDED                    78 non-null     float64
 3   PERCENT HOUSEHOLDS BELOW POVERTY              78 non-null     float64
 4   PERCENT AGED 16+ UNEMPLOYED                   78 non-null     float64
 5   PERCENT AGED 25+ WITHOUT HIGH SCHOOL DIPLOMA  78 non-null     float64
 6   PERCENT AGED UNDER 18 OR OVER 64              78 non-null     float64
 7   PER_CAPITA_INCOME                             78 non-null     int64  
 8   HARDSHIP_INDEX                                77 non-null     float64
dtypes: float64(7), int64(1), object(1)
memory usage: 5.6+ KB
```

In [105…

```
Census.isna().sum()
```

Out[105…

```
COMMUNITY_AREA_NUMBER                           1
COMMUNITY_AREA_NAME                             0
PERCENT OF HOUSING CROWDED                      0
PERCENT HOUSEHOLDS BELOW POVERTY                0
PERCENT AGED 16+ UNEMPLOYED                     0
PERCENT AGED 25+ WITHOUT HIGH SCHOOL DIPLOMA    0
PERCENT AGED UNDER 18 OR OVER 64                0
PER_CAPITA_INCOME                               0
HARDSHIP_INDEX                                  1
dtype: int64
```

In [106…

```
Census.head()
```

Out[106…

	COMMUNITY_AREA_NUMBER	COMMUNITY_AREA_NAME	PERCENT OF HOUSING CROWDED	PERCENT HOUSEHOLDS BELOW POVERTY	PERCENT AGED 16+ UNEMPLOYED	PERCENT AGED 25+ WITHOUT HIGH SCHOOL DIPLOMA
0	1.0	Rogers Park	7.7	23.6	8.7	18.2
1	2.0	West Ridge	7.8	17.2	8.8	20.8
2	3.0	Uptown	3.8	24.0	8.9	11.8
3	4.0	Lincoln Square	3.4	10.9	8.2	13.4

	COMMUNITY_AREA_NUMBER	COMMUNITY_AREA_NAME	PERCENT OF HOUSING CROWDED	PERCENT HOUSEHOLDS BELOW POVERTY	PERCENT AGED 16+ UNEMPLOYED	PERCENT AGED 25+ WITHOUT HIGH SCHOOL DIPLOMA
4	5.0	North Center	0.3	7.5	5.2	4.5

The census dataset was collected from 2008 to 2012. It includes various indicators of socioeconomic status for Chicago such as percent of housing crowded, percent households below poverty, percent aged 16+ unemployed, percent aged 25+ without high school diploma, percent aged under 18 or over64, per capita income and hardship index. The dataset consists of 78 rows and 9 columns with 8 numerical variables and 1 categorial variable. This data will be used alongside thee Chicago public schools data to identify student performance based on the socioeconomic indicator in certain areas.

Selecting 10 variables from Chicago public schools dataset

In [107…

```
Chicago_School1_select=Chicago_School1.iloc[:, [15, 17, 21, 23, 27, 29, 32, 38, 66, 73]]
```

In [108…

```
Chicago_School1_select
```

Out[108…

	HEALTHY_SCHOOL_CERTIFIED	SAFETY_SCORE	Environment Score	Instruction Score	Teachers Score	Parent Engagement Score	AVERAGE_STUD
0	Yes	99.0	74.0	66.0	70.0	56.0	
1	No	54.0	74.0	84.0	76.0	46.0	
2	No	61.0	50.0	36.0	NaN	47.0	
3	No	56.0	45.0	37.0	48.0	53.0	
4	Yes	49.0	60.0	55.0	54.0	53.0	
...	...	...	...	...	...	...	
561	No	32.0	46.0	55.0	NaN	49.0	
562	No	13.0	33.0	35.0	62.0	46.0	
563	No	41.0	56.0	32.0	NaN	50.0	

	HEALTHY_SCHOOL_CERTIFIED	SAFETY_SCORE	Environment Score	Instruction Score	Teachers Score	Parent Engagement Score	AVERAGE_STUD
564	No	70.0	80.0	66.0	NaN	59.0	
565	No	51.0	49.0	47.0	49.0	50.0	

566 rows × 10 columns

In [109…

```
Chicago_School1_select.isna().sum()
```

Out[109…

```
HEALTHY_SCHOOL_CERTIFIED        0
SAFETY_SCORE                   53
Environment Score              53
Instruction Score              53
Teachers Score                271
Parent Engagement Score       134
AVERAGE_STUDENT_ATTENDANCE      1
Gr3-5 Grade Level Math %      114
COLLEGE_ENROLLMENT              0
COMMUNITY_AREA_NUMBER           0
dtype: int64
```

In [110…

```
Chicago_School1_select.to_csv('Chicago_School1_select.csv')
```

Best choice for missing data: replacing missing data is the best choice because deleting either pairwise or listwise can lead to bias if the missing values are related to other variables in the dataset. Both pairwise deletion and listwise deletions are only used for missing completely at random values. Imputation method is used to replace missing values with mean, median, mode and etc.

In [111…

```
Chicago_select_fillna=Chicago_School1_select.fillna(Chicago_School1_select.mean())
Chicago_select_fillna
```

```
C:\Users\liden\AppData\Local\Temp/ipykernel_7816/160779938.py:1: FutureWarning: Dropping o
f nuisance columns in DataFrame reductions (with 'numeric_only=None') is deprecated; in a
future version this will raise TypeError.  Select only valid columns before calling the re
duction.
  Chicago_select_fillna=Chicago_School1_select.fillna(Chicago_School1_select.mean())
```

Out[111…

	HEALTHY_SCHOOL_CERTIFIED	SAFETY_SCORE	Environment Score	Instruction Score	Teachers Score	Parent Engagement Score	AVERAGE_STUI
0	Yes	99.0	74.0	66.0	70.000000	56.0	
1	No	54.0	74.0	84.0	76.000000	46.0	

	HEALTHY_SCHOOL_CERTIFIED	SAFETY_SCORE	Environment Score	Instruction Score	Teachers Score	Parent Engagement Score	AVERAGE_STUI
2	No	61.0	50.0	36.0	49.050847	47.0	
3	No	56.0	45.0	37.0	48.000000	53.0	
4	Yes	49.0	60.0	55.0	54.000000	53.0	
...	...	...	...	...	...	...	
561	No	32.0	46.0	55.0	49.050847	49.0	
562	No	13.0	33.0	35.0	62.000000	46.0	
563	No	41.0	56.0	32.0	49.050847	50.0	
564	No	70.0	80.0	66.0	49.050847	59.0	
565	No	51.0	49.0	47.0	49.000000	50.0	

566 rows × 10 columns

In [112…

```
Chicago_select_fillna.isna().sum()
```

Out[112…

```
HEALTHY_SCHOOL_CERTIFIED       0
SAFETY_SCORE                   0
Environment Score              0
Instruction Score              0
Teachers Score                 0
Parent Engagement Score        0
AVERAGE_STUDENT_ATTENDANCE     1
Gr3-5 Grade Level Math %       0
COLLEGE_ENROLLMENT             0
COMMUNITY_AREA_NUMBER          0
dtype: int64
```

In [113…

```
Chicago_select_fillna.to_csv('Chicago_select_fillna.csv')
```

Second Task: Creating Sqlite3 Database

In [114…

```
Chicago_df = pd.read_csv('Chicago_select_fillna.csv')
Chicago_df.head()
```

Out[114…

	Unnamed: 0	HEALTHY_SCHOOL_CERTIFIED	SAFETY_SCORE	Environment Score	Instruction Score	Teachers Score	Parent Engagement Score	AVER
0	0	Yes	99.0	74.0	66.0	70.000000	56.0	
1	1	No	54.0	74.0	84.0	76.000000	46.0	
2	2	No	61.0	50.0	36.0	49.050847	47.0	
3	3	No	56.0	45.0	37.0	48.000000	53.0	
4	4	Yes	49.0	60.0	55.0	54.000000	53.0	

In [115…

```
Chicago_df.shape
```

Out[115…

```
(566, 11)
```

In [116…

```
Chicago_df.describe()
```

Out[116…

	Unnamed: 0	SAFETY_SCORE	Environment Score	Instruction Score	Teachers Score	Parent Engagement Score	Gr3-5 Grade Level Math %	COLLEGE_ENROLLI
count	566.0000	566.000000	566.000000	566.000000	566.000000	566.000000	566.000000	566.0
mean	282.5000	49.504873	47.766082	48.288499	49.050847	50.231481	37.954204	626.0
std	163.5344	19.144363	15.436305	16.580151	12.879444	4.458829	16.944325	448.4
min	0.0000	1.000000	1.000000	1.000000	6.000000	37.000000	0.000000	21.0
25%	141.2500	36.000000	37.000000	38.000000	47.000000	48.000000	26.725000	342.2
50%	282.5000	49.504873	47.766082	48.288499	49.050847	50.231481	37.954204	504.0
75%	423.7500	60.000000	57.000000	58.000000	49.050847	52.000000	44.000000	790.5
max	565.0000	99.000000	99.000000	99.000000	99.000000	69.000000	100.000000	4368.0

In [117…

```
# dropping irrelevant columns
Chicago_df1=Chicago_df.drop(columns=['Unnamed: 0', 'Gr3-5 Grade Level Math %'])
Chicago_df1
```

Out[117…

	HEALTHY_SCHOOL_CERTIFIED	SAFETY_SCORE	Environment Score	Instruction Score	Teachers Score	Parent Engagement Score	AVERAGE_STUI
0	Yes	99.0	74.0	66.0	70.000000	56.0	
1	No	54.0	74.0	84.0	76.000000	46.0	
2	No	61.0	50.0	36.0	49.050847	47.0	
3	No	56.0	45.0	37.0	48.000000	53.0	
4	Yes	49.0	60.0	55.0	54.000000	53.0	
...	...	...	...	...	...	...	
561	No	32.0	46.0	55.0	49.050847	49.0	
562	No	13.0	33.0	35.0	62.000000	46.0	
563	No	41.0	56.0	32.0	49.050847	50.0	
564	No	70.0	80.0	66.0	49.050847	59.0	
565	No	51.0	49.0	47.0	49.000000	50.0	

566 rows × 9 columns

In [118…

```
Census_df = pd.read_csv('Census_Data.csv')
Census_df.head()
```

Out[118…

	COMMUNITY_AREA_NUMBER	COMMUNITY_AREA_NAME	PERCENT OF HOUSING CROWDED	PERCENT HOUSEHOLDS BELOW POVERTY	PERCENT AGED 16+ UNEMPLOYED	PERCENT AGED 25+ WITHOUT HIGH SCHOOL DIPLOMA	PE O
0	1.0	Rogers Park	7.7	23.6	8.7	18.2	
1	2.0	West Ridge	7.8	17.2	8.8	20.8	
2	3.0	Uptown	3.8	24.0	8.9	11.8	
3	4.0	Lincoln Square	3.4	10.9	8.2	13.4	
4	5.0	North Center	0.3	7.5	5.2	4.5	

In [119…

```
Census_drop=Census_df.drop(columns=['COMMUNITY_AREA_NAME'])
Census_drop
```

Out[119…

	COMMUNITY_AREA_NUMBER	PERCENT OF HOUSING CROWDED	PERCENT HOUSEHOLDS BELOW POVERTY	PERCENT AGED 16+ UNEMPLOYED	PERCENT AGED 25+ WITHOUT HIGH SCHOOL DIPLOMA	PERCENT AGED UNDER 18 OR OVER 64	PER_CAPITA_INCOI
0	1.0	7.7	23.6	8.7	18.2	27.5	239
1	2.0	7.8	17.2	8.8	20.8	38.5	230
2	3.0	3.8	24.0	8.9	11.8	22.2	357
3	4.0	3.4	10.9	8.2	13.4	25.5	375
4	5.0	0.3	7.5	5.2	4.5	26.2	571
...	...	...	...	...	...	...	
73	74.0	1.0	3.4	8.7	4.3	36.8	343
74	75.0	0.8	13.2	15.0	10.8	40.3	271
75	76.0	3.6	15.4	7.1	10.9	30.3	258
76	77.0	4.1	18.2	9.2	9.7	23.8	333
77	NaN	4.7	19.7	12.9	19.5	33.5	282

78 rows × 8 columns

Chicago table to Chicago.db

In [120…

```
conn = sqlite3.connect("Chicago.db")
print(conn)
```

```
<sqlite3.Connection object at 0x000002D531E0C8A0>
```

In [121…

```
conn.execute("DROP TABLE IF EXISTS `Chicago`")
print("Table dropped")
```

```
Table dropped
```

In [122…

```
conn = sqlite3.connect("Chicago.db")
try:
  conn.execute('''
      CREATE TABLE Chicago
      (SchoolID       INTEGER PRIMARY KEY,
       HEALTHY_SCHOOL_CERTIFIED    TEXT  NOT NULL,
       SAFETY_SCORE     FLOAT DEFAULT 0,
       EnvironmentScore     FLOAT DEFAULT 0,
       InstructionScore     FLOAT DEFAULT 0,
       TeachersScore     FLOAT DEFAULT 0,
       ParentEngagementScore          FLOAT DEFAULT 0,
       AVERAGE_STUDENT_ATTENDANCE   FLOAT DEFAULT 0,
       COLLEGE_ENROLLMENT           FLOAT DEFAULT 0,
       COMMUNITY_AREA_NUMBER        INTEGER,
       FOREIGN KEY (COMMUNITY_AREA_NUMBER) REFERENCES communities
       );''')
  print("Table created successfully")
except Exception as e:
  print(str(e))
  print('Table creation failed!!!')
finally:
  conn.close()
```

```
Table created successfully
```

In [123…

```
Chicago_list = Chicago_df1.values.tolist()
Chicago_list
```

Out[123…

```
[['Yes', 99.0, 74.0, 66.0, 70.0, 56.0, '96.00%', 813, 7],
 ['No', 54.0, 74.0, 84.0, 76.0, 46.0, '95.60%', 521, 43],
 ['No', 61.0, 50.0, 36.0, 49.05084745762712, 47.0, '95.70%', 1324, 70],
 ['No', 56.0, 45.0, 37.0, 48.0, 53.0, '95.50%', 556, 61],
 ['Yes', 49.0, 60.0, 55.0, 54.0, 53.0, '93.30%', 302, 34],
 ['No', 66.0, 66.0, 71.0, 50.0, 46.0, '97.00%', 266, 14],
 ['No',
  88.0,
  62.0,
  52.0,
  49.05084745762712,
  50.23148148148148,
  '96.30%',
  4368,
  5],
 ['No',
  67.0,
  30.0,
  18.0,
  49.05084745762712,
  50.23148148148148,
  '94.70%',
  620,
  24],
 ['No', 70.0, 67.0, 51.0, 49.05084745762712, 57.0, '92.70%', 232, 5],
 ['No', 43.0, 28.0, 37.0, 56.0, 51.0, '96.40%', 1023, 14],
 ['No', 99.0, 64.0, 46.0, 51.0, 50.23148148148148, '96.30%', 998, 5],
 ['No', 49.0, 31.0, 33.0, 41.0, 48.0, '92.50%', 476, 61],
 ['No', 73.0, 60.0, 59.0, 48.0, 57.0, '95.30%', 307, 6],
 ['No', 31.0, 32.0, 45.0, 49.05084745762712, 43.0, '92.50%', 398, 24],
 ['No', 28.0, 58.0, 60.0, 49.05084745762712, 47.0, '94.90%', 525, 49],
 ['No', 19.0, 22.0, 13.0, 19.0, 47.0, '90.10%', 428, 49],
 ['No', 37.0, 37.0, 35.0, 36.0, 48.0, '94.60%', 801, 23],
 ['No', 46.0, 39.0, 58.0, 49.05084745762712, 69.0, '95.60%', 274, 72],
 ['No',
  52.0,
  51.0,
  53.0,
  49.05084745762712,
  50.23148148148148,
  '80.30%',
  551,
  27],
 ['No', 45.0, 32.0, 28.0, 49.05084745762712, 45.0, '96.90%', 171, 29],
 ['No', 81.0, 60.0, 73.0, 54.0, 50.23148148148148, '96.30%', 264, 48],
 ['No', 38.0, 27.0, 35.0, 26.0, 50.0, '94.30%', 586, 22],
 ['No', 23.0, 35.0, 42.0, 42.0, 44.0, '91.40%', 539, 68],
 ['Yes', 57.0, 12.0, 14.0, 49.05084745762712, 53.0, '95.40%', 114, 24],
 ['No', 48.0, 37.0, 63.0, 31.0, 49.0, '94.60%', 730, 42],
 ['No',
  86.0,
  57.0,
  46.0,
  49.05084745762712,
  50.23148148148148,
  '97.40%',
  556,
  28],
 ['No', 31.0, 54.0, 32.0, 50.0, 53.0, '91.80%', 339, 67],
 ['No', 99.0, 85.0, 82.0, 82.0, 68.0, '97.50%', 245, 74],
 ['No',
  59.0,
  58.0,
  61.0,
  49.05084745762712,
  50.23148148148148,
  '94.60%',
  669,
  24],
 ['No', 32.0, 64.0, 76.0, 49.05084745762712, 52.0, '90.50%', 436, 38],
 ['No',
  49.50487329434698,
  47.76608187134503,
  48.28849902534113,
  49.05084745762712,
  51.0,
  '95.50%',
  550,
  39],
 ['No', 33.0, 48.0, 59.0, 49.05084745762712, 53.0, '94.20%', 345, 67],
 ['No', 27.0, 43.0, 54.0, 49.05084745762712, 48.0, '92.80%', 905, 46],
 ['No', 36.0, 60.0, 60.0, 49.05084745762712, 48.0, '90.80%', 539, 61],
 ['No',
  47.0,
  47.0,
  52.0,
  49.05084745762712,
  50.23148148148148,
  '94.80%',
  652,
  44],
 ['No', 87.0, 70.0, 64.0, 47.0, 66.0, '96.10%', 784, 17],
 ['No',
  49.50487329434698,
  47.76608187134503,
  48.28849902534113,
  49.05084745762712,
  39.0,
  '87.90%',
  459,
   44],
  ['No', 15.0, 41.0, 48.0, 49.05084745762712, 45.0, '92.10%', 573, 67],
  ['No', 48.0, 52.0, 40.0, 70.0, 51.0, '95.50%', 490, 70],
  ['No', 99.0, 78.0, 65.0, 49.05084745762712, 59.0, '96.50%', 572, 6],
  ['No',
   38.0,
   41.0,
   33.0,
   49.05084745762712,
   50.23148148148148,
   '76.00%',
   316,
   25],
  ['No', 23.0, 38.0, 34.0, 49.05084745762712, 46.0, '90.50%', 337, 42],
  ['No',
   38.0,
   45.0,
   53.0,
   49.05084745762712,
   50.23148148148148,
   '70.10%',
   250,
   25],
  ['No', 32.0, 38.0, 32.0, 16.0, 46.0, '92.10%', 398, 45],
  ['No',
   49.50487329434698,
   47.76608187134503,
   48.28849902534113,
   49.05084745762712,
   66.0,
   '60.90%',
   204,
   75],
  ['No', 75.0, 67.0, 66.0, 53.0, 55.0, '95.30%', 634, 19],
  ['No', 46.0, 44.0, 45.0, 48.0, 50.23148148148148, '83.70%', 1683, 31],
  ['No',
   36.0,
   35.0,
   58.0,
   49.05084745762712,
   50.23148148148148,
   '91.30%',
   302,
   68],
  ['No', 27.0, 55.0, 48.0, 37.0, 50.23148148148148, '95.50%', 301, 68],
  ['No', 50.0, 47.0, 32.0, 31.0, 50.23148148148148, '93.00%', 429, 24],
  ['No', 39.0, 51.0, 50.0, 49.05084745762712, 37.0, '88.90%', 350, 40],
  ['No', 45.0, 40.0, 39.0, 41.0, 49.0, '94.70%', 325, 39],
  ['No',
   49.50487329434698,
   47.76608187134503,
   48.28849902534113,
   66.0,
   58.0,
   '80.20%',
   137,
   64],
  ['No', 63.0, 77.0, 76.0, 49.05084745762712, 47.0, '94.20%', 340, 41],
  ['No',
   27.0,
   29.0,
   31.0,
   49.05084745762712,
   50.23148148148148,
   '90.60%',
   549,
   23],
  ['No',
   55.0,
   49.0,
   43.0,
   49.05084745762712,
   50.23148148148148,
   '96.30%',
   691,
   58],
  ['No', 41.0, 42.0, 43.0, 15.0, 51.0, '92.30%', 550, 38],
  ['No', 61.0, 85.0, 99.0, 71.0, 54.0, '95.50%', 286, 51],
  ['No', 51.0, 58.0, 70.0, 49.05084745762712, 48.0, '94.90%', 766, 44],
  ['Yes', 58.0, 59.0, 72.0, 55.0, 51.0, '96.50%', 792, 58],
  ['No', 48.0, 36.0, 29.0, 31.0, 50.23148148148148, '81.20%', 2366, 16],
  ['No', 68.0, 66.0, 75.0, 44.0, 50.0, '95.80%', 644, 21],
  ['No', 54.0, 67.0, 63.0, 32.0, 53.0, '91.70%', 343, 68],
  ['No', 35.0, 12.0, 12.0, 49.05084745762712, 48.0, '94.80%', 537, 70],
  ['No', 26.0, 2.0, 1.0, 41.0, 44.0, '91.90%', 329, 38],
  ['No', 54.0, 63.0, 52.0, 50.0, 45.0, '95.10%', 915, 61],
  ['No', 59.0, 53.0, 51.0, 44.0, 46.0, '86.00%', 1415, 19],
  ['No',
   42.0,
   82.0,
   72.0,
   49.05084745762712,
   50.23148148148148,
   '94.40%',
   315,
   29],
  ['No', 44.0, 27.0, 30.0, 54.0, 50.0, '95.70%', 1032, 70],
  ['No', 40.0, 41.0, 43.0, 49.05084745762712, 43.0, '95.30%', 446, 30],
  ['No', 32.0, 34.0, 32.0, 49.05084745762712, 50.0, '93.40%', 246, 73],
  ['No', 36.0, 35.0, 40.0, 49.05084745762712, 39.0, '92.00%', 403, 41],
  ['No', 53.0, 50.0, 35.0, 51.0, 54.0, '94.90%', 600, 60],
  ['No', 25.0, 40.0, 44.0, 25.0, 44.0, '92.70%', 291, 45],
  ['No', 26.0, 29.0, 34.0, 32.0, 50.23148148148148, '79.20%', 1890, 19],
  ['No', 58.0, 44.0, 34.0, 46.0, 49.0, '95.60%', 622, 22],
  ['No', 16.0, 32.0, 38.0, 49.05084745762712, 46.0, '89.20%', 407, 67],
  ['No', 20.0, 79.0, 82.0, 19.0, 48.0, '92.80%', 310, 69],
  ['No', 55.0, 49.0, 51.0, 49.0, 50.23148148148148, '93.80%', 551, 69],
  ['No', 41.0, 44.0, 48.0, 34.0, 48.0, '93.90%', 430, 26],
  ['No', 24.0, 62.0, 71.0, 49.05084745762712, 53.0, '93.80%', 331, 67],
  ['No', 66.0, 40.0, 22.0, 49.05084745762712, 47.0, '96.30%', 601, 17],
  ['No', 58.0, 57.0, 53.0, 57.0, 50.23148148148148, '93.10%', 502, 17],
  ['No', 87.0, 49.0, 47.0, 40.0, 52.0, '92.80%', 579, 74],
  ['No',
   32.0,
   23.0,
   35.0,
   49.05084745762712,
   50.23148148148148,
   '90.30%',
   483,
   35],
  ['No', 27.0, 21.0, 30.0, 39.0, 50.23148148148148, '68.80%', 833, 45],
  ['No', 78.0, 66.0, 50.0, 62.0, 52.0, '95.90%', 650, 9],
  ['No', 36.0, 41.0, 32.0, 42.0, 50.23148148148148, '74.40%', 688, 49],
  ['No', 66.0, 57.0, 55.0, 49.05084745762712, 54.0, '95.50%', 309, 24],
  ['No', 25.0, 28.0, 34.0, 59.0, 51.0, '91.70%', 602, 71],
  ['No', 36.0, 50.0, 51.0, 59.0, 46.0, '92.50%', 531, 66],
  ['No', 60.0, 51.0, 46.0, 48.0, 54.0, '91.40%', 409, 29],
  ['No', 59.0, 59.0, 63.0, 72.0, 52.0, '95.40%', 1038, 58],
  ['No',
   14.0,
   28.0,
   34.0,
   49.05084745762712,
   50.23148148148148,
   '94.10%',
   233,
   49],
  ['No', 45.0, 64.0, 57.0, 85.0, 50.23148148148148, '95.20%', 274, 38],
  ['No', 27.0, 51.0, 52.0, 31.0, 49.0, '92.50%', 298, 29],
  ['Yes',
   49.50487329434698,
   47.76608187134503,
   48.28849902534113,
   54.0,
   52.0,
   '95.90%',
   894,
   30],
  ['No', 52.0, 47.0, 48.0, 53.0, 50.0, '95.50%', 752, 2],
  ['No',
   49.50487329434698,
   47.76608187134503,
   48.28849902534113,
   49.05084745762712,
   57.0,
   '91.90%',
   160,
   10],
  ['No',
   50.0,
   45.0,
   54.0,
   49.05084745762712,
   50.23148148148148,
   '91.30%',
   324,
   38],
  ['No',
   39.0,
   31.0,
   29.0,
   49.05084745762712,
   50.23148148148148,
   '94.70%',
   631,
   30],
  ['No', 42.0, 49.0, 50.0, 49.05084745762712, 46.0, '92.60%', 828, 23],
  ['No', 30.0, 34.0, 36.0, 32.0, 50.23148148148148, '91.60%', 365, 68],
  ['No', 48.0, 81.0, 66.0, 44.0, 45.0, '94.80%', 370, 26],
  ['No',
   43.0,
   42.0,
   32.0,
   49.05084745762712,
   50.23148148148148,
   '86.10%',
   1456,
   30],
  ['No',
   49.50487329434698,
   47.76608187134503,
   48.28849902534113,
   49.05084745762712,
   58.0,
   '96.70%',
   200,
   5],
['No', 41.0, 42.0, 35.0, 49.05084745762712, 52.0, '90.70%', 357, 61],
['No', 47.0, 47.0, 52.0, 36.0, 50.23148148148148, '95.30%', 1139, 2],
['No',
 49.50487329434698,
 47.76608187134503,
 48.28849902534113,
 53.0,
 62.0,
 '97.30%',
 363,
 16],
['No', 28.0, 32.0, 39.0, 32.0, 44.0, '93.40%', 823, 66],
['No', 34.0, 20.0, 1.0, 49.05084745762712, 42.0, '95.10%', 632, 52],
['No',
 49.50487329434698,
 47.76608187134503,
 48.28849902534113,
 49.05084745762712,
 52.0,
 '94.70%',
 1001,
 19],
['No',
 49.50487329434698,
 47.76608187134503,
 48.28849902534113,
 49.05084745762712,
 56.0,
 '92.90%',
 915,
 39],
['No', 19.0, 25.0, 49.0, 49.05084745762712, 48.0, '92.50%', 381, 42],
['No', 46.0, 65.0, 71.0, 59.0, 49.0, '94.60%', 310, 49],
['No', 61.0, 57.0, 58.0, 74.0, 53.0, '95.40%', 543, 70],
['No', 46.0, 66.0, 69.0, 13.0, 50.23148148148148, '94.20%', 556, 29],
['No', 27.0, 35.0, 47.0, 25.0, 50.23148148148148, '62.50%', 318, 40],
['No', 99.0, 66.0, 88.0, 41.0, 55.0, '97.60%', 192, 50],
['No', 99.0, 51.0, 53.0, 70.0, 56.0, '96.90%', 468, 12],
['No', 95.0, 52.0, 49.0, 49.05084745762712, 56.0, '96.10%', 431, 10],
['No', 1.0, 13.0, 22.0, 49.05084745762712, 48.0, '89.30%', 244, 40],
['No', 33.0, 55.0, 50.0, 58.0, 50.0, '92.90%', 783, 43],
['No', 47.0, 34.0, 48.0, 49.05084745762712, 42.0, '95.00%', 1365, 40],
['No', 38.0, 66.0, 52.0, 79.0, 55.0, '93.40%', 414, 26],
['No', 22.0, 40.0, 54.0, 53.0, 50.0, '95.10%', 554, 46],
['No',
 49.50487329434698,
 47.76608187134503,
 48.28849902534113,
 45.0,
 50.23148148148148,
 '95.30%',
 321,
 59],
['No', 50.0, 62.0, 41.0, 26.0, 49.0, '88.90%', 314, 8],
['No', 30.0, 37.0, 55.0, 49.05084745762712, 49.0, '93.30%', 377, 25],
['No', 77.0, 50.0, 49.0, 54.0, 49.0, '95.40%', 830, 65],
['No', 34.0, 45.0, 37.0, 35.0, 51.0, '78.40%', 461, 61],
['No', 27.0, 44.0, 48.0, 49.05084745762712, 51.0, '93.80%', 138, 53],
['No',
 26.0,
 28.0,
 23.0,
 49.05084745762712,
 50.23148148148148,
 '84.30%',
 1716,
  15],
 ['No', 44.0, 52.0, 56.0, 28.0, 46.0, '91.60%', 419, 67],
 ['No', 29.0, 43.0, 50.0, 49.05084745762712, 49.0, '89.70%', 182, 69],
 ['Yes',
  44.0,
  49.0,
  53.0,
  49.05084745762712,
  50.23148148148148,
  '95.80%',
  1133,
  30],
 ['No', 86.0, 64.0, 59.0, 49.05084745762712, 50.0, '95.60%', 762, 72],
 ['No', 51.0, 50.0, 61.0, 55.0, 52.0, '95.20%', 289, 24],
 ['No', 70.0, 53.0, 51.0, 48.0, 50.0, '95.10%', 451, 4],
 ['No', 31.0, 44.0, 50.0, 49.05084745762712, 44.0, '94.30%', 1111, 25],
 ['No', 30.0, 32.0, 19.0, 34.0, 43.0, '90.60%', 473, 61],
 ['No', 99.0, 95.0, 95.0, 90.0, 56.0, '95.50%', 344, 24],
 ['No', 13.0, 28.0, 28.0, 50.0, 50.23148148148148, '84.80%', 458, 69],
 ['No', 60.0, 43.0, 42.0, 79.0, 46.0, '96.60%', 988, 30],
 ['No', 31.0, 47.0, 40.0, 42.0, 47.0, '88.00%', 471, 42],
 ['No', 30.0, 65.0, 63.0, 47.0, 50.23148148148148, '89.90%', 238, 43],
 ['No',
  49.50487329434698,
  47.76608187134503,
  48.28849902534113,
  54.0,
  49.0,
  '95.80%',
  1021,
  63],
 ['No', 67.0, 53.0, 63.0, 64.0, 53.0, '92.20%', 581, 63],
 ['No', 71.0, 49.0, 42.0, 49.05084745762712, 43.0, '95.40%', 672, 15],
 ['No', 36.0, 49.0, 43.0, 49.05084745762712, 49.0, '92.90%', 342, 75],
 ['No', 42.0, 47.0, 37.0, 8.0, 44.0, '96.80%', 375, 1],
 ['No',
  44.0,
  12.0,
  1.0,
  49.05084745762712,
  50.23148148148148,
  '95.90%',
  786,
  76],
 ['No', 60.0, 47.0, 44.0, 65.0, 43.0, '96.10%', 389, 59],
 ['No', 31.0, 41.0, 34.0, 25.0, 50.23148148148148, '93.40%', 604, 66],
 ['No', 53.0, 53.0, 58.0, 82.0, 51.0, '96.80%', 860, 21],
 ['No',
  49.50487329434698,
  47.76608187134503,
  48.28849902534113,
  49.05084745762712,
  57.0,
  '80.00%',
  100,
  28],
 ['No', 51.0, 43.0, 47.0, 37.0, 44.0, '96.30%', 1560, 62],
 ['No', 29.0, 17.0, 43.0, 50.0, 48.0, '93.40%', 344, 73],
 ['No', 20.0, 37.0, 46.0, 49.05084745762712, 46.0, '90.00%', 112, 39],
 ['No', 35.0, 40.0, 48.0, 72.0, 48.0, '95.60%', 1382, 63],
 ['No', 34.0, 61.0, 71.0, 49.05084745762712, 47.0, '93.40%', 478, 73],
 ['No', 11.0, 20.0, 22.0, 46.0, 46.0, '93.70%', 418, 71],
 ['Yes', 49.0, 48.0, 47.0, 34.0, 47.0, '95.10%', 314, 30],
 ['No', 23.0, 37.0, 54.0, 61.0, 48.0, '93.50%', 891, 66],
 ['No', 38.0, 45.0, 39.0, 28.0, 50.23148148148148, '90.60%', 224, 37],
 ['No', 37.0, 49.0, 64.0, 49.05084745762712, 44.0, '91.90%', 301, 25],
 ['No', 30.0, 53.0, 57.0, 49.05084745762712, 48.0, '91.90%', 808, 68],
 ['No', 46.0, 55.0, 66.0, 56.0, 47.0, '92.50%', 504, 49],
 ['No',
  27.0,
  30.0,
  45.0,
  49.05084745762712,
  50.23148148148148,
  '91.90%',
  609,
  49],
 ['No', 87.0, 60.0, 55.0, 49.05084745762712, 58.0, '95.90%', 363, 8],
 ['No', 46.0, 30.0, 45.0, 14.0, 49.0, '96.60%', 867, 58],
 ['No', 48.0, 33.0, 34.0, 49.05084745762712, 48.0, '97.00%', 1195, 21],
 ['No',
  49.50487329434698,
  47.76608187134503,
  48.28849902534113,
  26.0,
  47.0,
  '95.80%',
  949,
  19],
 ['No', 60.0, 84.0, 72.0, 55.0, 66.0, '96.80%', 220, 29],
 ['No', 86.0, 99.0, 99.0, 99.0, 49.0, '96.90%', 269, 24],
 ['No', 44.0, 39.0, 37.0, 49.05084745762712, 50.0, '79.20%', 354, 25],
 ['No', 16.0, 14.0, 16.0, 49.05084745762712, 40.0, '94.70%', 619, 22],
 ['No',
  49.50487329434698,
  47.76608187134503,
  48.28849902534113,
  81.0,
  69.0,
  '72.40%',
  260,
  9],
 ['No', 72.0, 60.0, 59.0, 49.05084745762712, 50.0, '94.70%', 495, 5],
 ['No', 70.0, 42.0, 51.0, 29.0, 50.23148148148148, '87.00%', 1636, 13],
 ['No', 14.0, 29.0, 37.0, 35.0, 50.23148148148148, '72.50%', 1052, 63],
 ['No', 65.0, 40.0, 42.0, 33.0, 49.0, '95.00%', 606, 28],
 ['No', 28.0, 58.0, 55.0, 36.0, 47.0, '92.00%', 237, 71],
 ['No',
  61.0,
  65.0,
  77.0,
  49.05084745762712,
  50.23148148148148,
  '92.10%',
  315,
  26],
 ['No', 59.0, 59.0, 64.0, 45.0, 50.23148148148148, '96.30%', 1509, 2],
 ['No', 49.0, 33.0, 40.0, 49.05084745762712, 53.0, '93.20%', 283, 60],
 ['No', 61.0, 62.0, 60.0, 59.0, 49.0, '96.30%', 709, 77],
 ['No', 76.0, 57.0, 36.0, 61.0, 52.0, '94.40%', 358, 74],
 ['No', 33.0, 40.0, 48.0, 54.0, 50.23148148148148, '71.30%', 621, 50],
 ['No',
  49.50487329434698,
  47.76608187134503,
  48.28849902534113,
  77.0,
  60.0,
  '92.60%',
  182,
  25],
 ['No', 43.0, 49.0, 40.0, 49.05084745762712, 42.0, '91.00%', 396, 8],
 ['No', 59.0, 66.0, 75.0, 49.05084745762712, 50.0, '93.50%', 259, 50],
 ['No', 39.0, 32.0, 36.0, 49.05084745762712, 47.0, '93.90%', 297, 25],
 ['No',
  53.0,
  37.0,
  34.0,
  49.05084745762712,
  50.23148148148148,
  '91.60%',
  532,
  54],
 ['No', 25.0, 33.0, 30.0, 49.05084745762712, 52.0, '90.90%', 546, 54],
 ['No', 58.0, 33.0, 43.0, 76.0, 52.0, '95.50%', 813, 52],
 ['No', 46.0, 36.0, 36.0, 46.0, 50.23148148148148, '80.20%', 1524, 52],
 ['No', 23.0, 43.0, 58.0, 49.05084745762712, 68.0, '93.80%', 504, 49],
 ['No', 74.0, 77.0, 80.0, 71.0, 54.0, '93.60%', 828, 23],
 ['No', 6.0, 30.0, 41.0, 44.0, 51.0, '91.40%', 321, 26],
 ['No', 44.0, 48.0, 48.0, 43.0, 47.0, '95.90%', 713, 30],
 ['No', 49.0, 46.0, 57.0, 49.05084745762712, 47.0, '95.20%', 280, 3],
 ['No', 35.0, 43.0, 31.0, 49.05084745762712, 48.0, '91.90%', 402, 67],
 ['No', 43.0, 46.0, 44.0, 65.0, 53.0, '87.50%', 371, 30],
 ['No', 64.0, 58.0, 47.0, 44.0, 49.0, '95.00%', 688, 16],
 ['No', 26.0, 65.0, 76.0, 49.05084745762712, 52.0, '91.50%', 245, 26],
 ['No',
  42.0,
  34.0,
  29.0,
  49.05084745762712,
  50.23148148148148,
  '89.30%',
  1672,
  65],
 ['No', 64.0, 44.0, 49.0, 49.05084745762712, 47.0, '87.10%', 783, 49],
 ['No', 78.0, 67.0, 62.0, 62.0, 52.0, '95.80%', 363, 13],
 ['No', 35.0, 34.0, 33.0, 49.05084745762712, 50.0, '95.00%', 1499, 19],
 ['No', 28.0, 45.0, 63.0, 49.05084745762712, 47.0, '96.10%', 549, 47],
 ['No', 28.0, 26.0, 31.0, 49.05084745762712, 44.0, '93.80%', 893, 23],
 ['No', 68.0, 41.0, 50.0, 49.05084745762712, 44.0, '96.20%', 605, 25],
 ['Yes', 92.0, 61.0, 56.0, 84.0, 58.0, '96.90%', 569, 6],
 ['No', 29.0, 26.0, 63.0, 52.0, 51.0, '93.80%', 259, 26],
 ['No', 58.0, 32.0, 22.0, 49.05084745762712, 47.0, '96.70%', 977, 77],
 ['No', 43.0, 57.0, 49.0, 30.0, 46.0, '96.20%', 1299, 14],
 ['No', 38.0, 30.0, 30.0, 31.0, 50.23148148148148, '95.20%', 633, 55],
 ['No',
  49.50487329434698,
  47.76608187134503,
  48.28849902534113,
  49.05084745762712,
  48.0,
  '96.20%',
  1334,
  19],
 ['No',
  41.0,
  68.0,
  76.0,
  49.05084745762712,
  50.23148148148148,
  '95.10%',
  393,
  25],
 ['No',
  33.0,
  42.0,
  55.0,
  49.05084745762712,
  50.23148148148148,
  '91.80%',
  448,
  69],
 ['No', 45.0, 39.0, 61.0, 51.0, 52.0, '95.10%', 588, 75],
 ['No',
  49.50487329434698,
  47.76608187134503,
  48.28849902534113,
  28.0,
  42.0,
  '72.00%',
  731,
  43],
 ['No', 64.0, 67.0, 69.0, 51.0, 53.0, '95.00%', 672, 16],
 ['No', 31.0, 37.0, 44.0, 35.0, 50.23148148148148, '72.20%', 561, 68],
 ['No', 65.0, 48.0, 38.0, 49.05084745762712, 52.0, '95.90%', 611, 6],
 ['No', 28.0, 31.0, 30.0, 24.0, 44.0, '92.60%', 472, 46],
 ['No',
  42.0,
  47.0,
  56.0,
  49.05084745762712,
  50.23148148148148,
  '91.90%',
  458,
  25],
 ['No', 29.0, 34.0, 43.0, 41.0, 50.23148148148148, '70.50%', 1518, 42],
 ['No', 64.0, 64.0, 42.0, 49.05084745762712, 53.0, '92.90%', 423, 77],
 ['No',
  49.0,
  64.0,
  93.0,
  49.05084745762712,
  50.23148148148148,
  '93.20%',
  215,
  35],
 ['No', 31.0, 63.0, 98.0, 49.05084745762712, 46.0, '92.10%', 130, 30],
 ['No', 58.0, 66.0, 65.0, 49.05084745762712, 51.0, '92.20%', 388, 30],
 ['No', 67.0, 50.0, 51.0, 49.05084745762712, 51.0, '96.20%', 690, 6],
 ['No', 39.0, 36.0, 39.0, 49.05084745762712, 49.0, '92.90%', 285, 54],
 ['No', 34.0, 27.0, 27.0, 27.0, 50.23148148148148, '95.50%', 970, 63],
 ['Yes', 57.0, 57.0, 62.0, 56.0, 50.0, '95.50%', 916, 31],
 ['No', 44.0, 46.0, 38.0, 49.05084745762712, 52.0, '91.10%', 244, 38],
 ['No', 17.0, 28.0, 29.0, 49.05084745762712, 40.0, '92.10%', 575, 43],
 ['No',
  49.50487329434698,
  47.76608187134503,
  48.28849902534113,
  49.05084745762712,
  45.0,
  '90.50%',
  140,
  36],
 ['No',
  63.0,
  70.0,
  58.0,
  49.05084745762712,
  50.23148148148148,
  '95.40%',
  411,
  27],
 ['No', 57.0, 33.0, 20.0, 69.0, 63.0, '87.60%', 203, 15],
 ['No', 55.0, 40.0, 38.0, 59.0, 55.0, '94.20%', 574, 11],
 ['No', 71.0, 48.0, 38.0, 49.05084745762712, 52.0, '96.50%', 695, 4],
 ['No', 99.0, 57.0, 52.0, 49.05084745762712, 61.0, '96.20%', 199, 44],
 ['No', 99.0, 76.0, 74.0, 49.05084745762712, 40.0, '96.40%', 923, 6],
 ['No', 22.0, 22.0, 24.0, 49.05084745762712, 41.0, '95.40%', 845, 61],
 ['No',
  39.0,
  32.0,
  28.0,
  49.05084745762712,
  50.23148148148148,
  '94.90%',
  317,
  43],
 ['No', 45.0, 26.0, 22.0, 49.05084745762712, 47.0, '95.10%', 1125, 22],
 ['No',
  32.0,
  39.0,
  39.0,
  49.05084745762712,
  50.23148148148148,
  '91.80%',
  449,
  46],
 ['No', 83.0, 67.0, 58.0, 94.0, 54.0, '95.50%', 491, 24],
 ['No', 30.0, 25.0, 46.0, 49.05084745762712, 44.0, '91.40%', 306, 35],
 ['No', 39.0, 39.0, 40.0, 49.05084745762712, 45.0, '93.20%', 593, 23],
 ['No', 38.0, 57.0, 58.0, 49.05084745762712, 50.0, '97.00%', 1834, 58],
 ['No',
  49.50487329434698,
  47.76608187134503,
  48.28849902534113,
  49.05084745762712,
  47.0,
  '92.20%',
  277,
  42],
 ['No', 49.0, 50.0, 46.0, 56.0, 48.0, '97.80%', 492, 34],
 ['No', 67.0, 61.0, 75.0, 66.0, 51.0, '94.70%', 365, 29],
 ['No', 56.0, 44.0, 46.0, 44.0, 53.0, '96.00%', 928, 52],
 ['No', 42.0, 27.0, 28.0, 36.0, 52.0, '88.90%', 306, 44],
 ['No',
  63.0,
  49.0,
  47.0,
  49.05084745762712,
  50.23148148148148,
  '95.70%',
  1181,
  11],
 ['No', 44.0, 43.0, 38.0, 72.0, 52.0, '94.10%', 474, 24],
 ['No', 35.0, 42.0, 67.0, 40.0, 46.0, '93.10%', 364, 27],
 ['No',
  49.50487329434698,
  47.76608187134503,
  48.28849902534113,
  49.05084745762712,
  51.0,
  '91.30%',
  340,
  53],
 ['No', 38.0, 47.0, 58.0, 49.05084745762712, 51.0, '92.00%', 355, 68],
 ['No', 75.0, 68.0, 52.0, 69.0, 57.0, '95.80%', 786, 22],
 ['Yes', 60.0, 70.0, 91.0, 49.05084745762712, 48.0, '95.10%', 501, 31],
 ['No', 35.0, 70.0, 81.0, 41.0, 51.0, '95.10%', 800, 20],
 ['No', 59.0, 57.0, 46.0, 16.0, 50.0, '93.80%', 249, 35],
 ['No', 80.0, 75.0, 59.0, 49.05084745762712, 58.0, '95.60%', 638, 16],
 ['No', 52.0, 44.0, 54.0, 88.0, 43.0, '95.90%', 328, 27],
 ['Yes', 50.0, 39.0, 38.0, 53.0, 50.0, '95.60%', 510, 58],
 ['No', 87.0, 65.0, 39.0, 56.0, 58.0, '95.90%', 668, 5],
 ['No', 74.0, 69.0, 79.0, 51.0, 58.0, '95.60%', 590, 64],
 ['No', 32.0, 37.0, 37.0, 48.0, 43.0, '98.40%', 664, 34],
 ['No', 32.0, 43.0, 51.0, 42.0, 52.0, '93.20%', 553, 75],
 ['No', 42.0, 33.0, 34.0, 32.0, 50.23148148148148, '95.30%', 1777, 66],
 ['No', 13.0, 37.0, 55.0, 35.0, 47.0, '92.90%', 254, 42],
 ['No', 53.0, 32.0, 27.0, 24.0, 50.23148148148148, '83.00%', 1589, 56],
 ['No', 42.0, 53.0, 42.0, 21.0, 51.0, '94.10%', 537, 69],
 ['No',
  65.0,
  45.0,
  24.0,
  49.05084745762712,
  50.23148148148148,
  '94.90%',
  409,
  31],
 ['No', 51.0, 44.0, 41.0, 42.0, 50.0, '78.20%', 967, 62],
 ['No', 51.0, 61.0, 61.0, 36.0, 47.0, '93.00%', 476, 69],
 ['No', 44.0, 49.0, 61.0, 49.05084745762712, 52.0, '92.90%', 554, 25],
 ['No', 33.0, 29.0, 43.0, 33.0, 44.0, '94.40%', 679, 61],
 ['No', 64.0, 49.0, 54.0, 49.05084745762712, 50.0, '95.40%', 733, 56],
 ['No', 74.0, 24.0, 31.0, 49.05084745762712, 55.0, '97.20%', 317, 72],
 ['No', 95.0, 75.0, 60.0, 65.0, 61.0, '96.20%', 580, 5],
 ['No',
  49.50487329434698,
  47.76608187134503,
  48.28849902534113,
  78.0,
  50.23148148148148,
  '95.50%',
  258,
  35],
 ['No', 72.0, 53.0, 39.0, 48.0, 50.0, '95.70%', 841, 51],
 ['No', 51.0, 45.0, 46.0, 42.0, 50.23148148148148, '74.10%', 726, 27],
 ['No',
  49.50487329434698,
  47.76608187134503,
  48.28849902534113,
  49.05084745762712,
  50.23148148148148,
  '81.20%',
  1245,
  49],
 ['No', 44.0, 39.0, 43.0, 59.0, 49.0, '95.20%', 323, 27],
 ['No', 37.0, 54.0, 64.0, 49.05084745762712, 43.0, '92.40%', 475, 28],
 ['No', 60.0, 56.0, 63.0, 49.05084745762712, 53.0, '95.90%', 266, 53],
 ['No', 75.0, 61.0, 68.0, 49.05084745762712, 65.0, '96.30%', 919, 14],
 ['No', 48.0, 64.0, 58.0, 35.0, 46.0, '91.90%', 527, 67],
 ['No', 66.0, 70.0, 67.0, 43.0, 46.0, '96.20%', 809, 30],
 ['No', 56.0, 67.0, 41.0, 42.0, 52.0, '94.00%', 355, 3],
 ['No', 26.0, 63.0, 62.0, 46.0, 50.23148148148148, '93.50%', 384, 44],
 ['No',
  49.50487329434698,
  47.76608187134503,
  48.28849902534113,
  49.05084745762712,
  45.0,
  '94.00%',
  482,
  71],
 ['No', 60.0, 50.0, 51.0, 55.0, 56.0, '95.30%', 720, 10],
 ['No', 45.0, 41.0, 28.0, 21.0, 52.0, '92.40%', 375, 53],
 ['No', 71.0, 68.0, 51.0, 63.0, 56.0, '95.90%', 348, 24],
 ['No', 44.0, 21.0, 20.0, 49.05084745762712, 48.0, '95.90%', 937, 16],
 ['No', 61.0, 57.0, 49.0, 15.0, 52.0, '95.60%', 753, 1],
 ['Yes', 57.0, 37.0, 40.0, 64.0, 53.0, '95.00%', 869, 24],
 ['No',
  49.50487329434698,
  47.76608187134503,
  48.28849902534113,
  54.0,
  51.0,
  '95.80%',
  793,
  30],
 ['No', 50.0, 55.0, 59.0, 54.0, 53.0, '96.50%', 362, 3],
 ['No', 32.0, 29.0, 28.0, 17.0, 45.0, '96.10%', 1204, 30],
 ['No', 57.0, 49.0, 56.0, 45.0, 47.0, '95.50%', 1317, 18],
 ['No', 36.0, 56.0, 53.0, 43.0, 48.0, '96.40%', 286, 31],
 ['No', 58.0, 42.0, 45.0, 50.0, 59.0, '95.20%', 293, 27],
 ['No', 43.0, 32.0, 22.0, 49.05084745762712, 49.0, '95.40%', 490, 25],
 ['No',
  37.0,
  51.0,
  68.0,
  49.05084745762712,
  50.23148148148148,
  '94.90%',
  462,
  3],
 ['No', 33.0, 49.0, 40.0, 49.05084745762712, 46.0, '95.00%', 328, 48],
 ['No', 62.0, 69.0, 56.0, 59.0, 58.0, '94.80%', 262, 68],
 ['No', 50.0, 35.0, 37.0, 34.0, 48.0, '95.80%', 656, 31],
 ['No',
  49.50487329434698,
  47.76608187134503,
  48.28849902534113,
  42.0,
  49.0,
  '96.00%',
  886,
  1],
 ['No', 67.0, 58.0, 26.0, 49.05084745762712, 40.0, '90.50%', 83, 29],
 ['No', 67.0, 90.0, 69.0, 67.0, 57.0, '96.00%', 576, 25],
 ['No',
  31.0,
  24.0,
  20.0,
  49.05084745762712,
  50.23148148148148,
  '83.40%',
  26,
  48],
 ['No', 76.0, 57.0, 63.0, 49.05084745762712, 53.0, '96.30%', 283, 72],
 ['No', 32.0, 32.0, 28.0, 43.0, 50.23148148148148, '74.40%', 1252, 20],
 ['No',
  63.0,
  48.0,
  56.0,
  49.05084745762712,
  50.23148148148148,
  '88.40%',
  1852,
  39],
 ['No', 64.0, 52.0, 43.0, 49.05084745762712, 49.0, '87.70%', 1500, 6],
 ['No', 28.0, 16.0, 30.0, 48.0, 46.0, '90.00%', 424, 49],
 ['No', 99.0, 62.0, 52.0, 43.0, 53.0, '96.80%', 576, 7],
 ['No', 66.0, 36.0, 52.0, 79.0, 55.0, '95.10%', 609, 24],
 ['No', 54.0, 48.0, 27.0, 30.0, 49.0, '96.10%', 1520, 19],
 ['No',
   49.50487329434698,
   47.76608187134503,
   48.28849902534113,
   49.05084745762712,
   50.23148148148148,
   '92.90%',
   429,
   23],
  ['No',
   49.50487329434698,
   47.76608187134503,
   48.28849902534113,
   49.05084745762712,
   50.23148148148148,
   '91.00%',
   508,
   29],
  ['No',
   49.50487329434698,
   47.76608187134503,
   48.28849902534113,
   58.0,
   52.0,
   '96.30%',
   654,
   30],
  ['No', 33.0, 49.0, 52.0, 49.05084745762712, 45.0, '94.70%', 518, 27],
  ['No', 79.0, 51.0, 67.0, 63.0, 52.0, '97.40%', 328, 44],
  ['No', 21.0, 36.0, 64.0, 48.0, 42.0, '91.80%', 565, 25],
  ['No', 65.0, 44.0, 49.0, 45.0, 50.23148148148148, '84.60%', 2342, 7],
  ['No', 42.0, 34.0, 28.0, 49.05084745762712, 46.0, '95.00%', 658, 70],
  ['No', 45.0, 52.0, 66.0, 43.0, 47.0, '96.70%', 842, 30],
  ['No', 56.0, 32.0, 28.0, 49.05084745762712, 50.0, '95.10%', 941, 21],
  ['No', 57.0, 43.0, 46.0, 32.0, 50.23148148148148, '94.70%', 455, 22],
  ['No', 53.0, 23.0, 21.0, 49.05084745762712, 56.0, '95.20%', 432, 6],
  ['No',
   49.50487329434698,
   47.76608187134503,
   48.28849902534113,
   57.0,
   60.0,
   '96.60%',
   481,
   7],
  ['No', 42.0, 49.0, 62.0, 49.05084745762712, 59.0, '92.60%', 93, 25],
  ['No',
   58.0,
   34.0,
   30.0,
   49.05084745762712,
   50.23148148148148,
   '95.50%',
   729,
   6],
  ['No',
   50.0,
   42.0,
   38.0,
   49.05084745762712,
   50.23148148148148,
   '95.00%',
   1173,
   62],
  ['No', 40.0, 58.0, 59.0, 49.05084745762712, 46.0, '93.70%', 411, 38],
  ['No', 5.0, 33.0, 40.0, 49.05084745762712, 45.0, '89.20%', 511, 67],
  ['No', 37.0, 42.0, 34.0, 49.05084745762712, 47.0, '95.20%', 1139, 19],
 ['No', 59.0, 33.0, 38.0, 41.0, 50.0, '95.40%', 864, 4],
 ['No', 56.0, 47.0, 25.0, 49.05084745762712, 50.0, '94.30%', 438, 77],
 ['No', 38.0, 54.0, 66.0, 49.05084745762712, 46.0, '92.50%', 323, 71],
 ['No', 63.0, 59.0, 71.0, 55.0, 51.0, '95.80%', 580, 24],
 ['No', 41.0, 43.0, 31.0, 32.0, 50.23148148148148, '66.80%', 599, 27],
 ['Yes', 45.0, 42.0, 45.0, 49.05084745762712, 52.0, '95.90%', 426, 31],
 ['No', 50.0, 50.0, 48.0, 49.05084745762712, 49.0, '94.00%', 356, 73],
 ['No', 54.0, 61.0, 81.0, 84.0, 47.0, '95.50%', 873, 65],
 ['No', 60.0, 55.0, 64.0, 49.05084745762712, 43.0, '96.80%', 1306, 30],
 ['No', 43.0, 37.0, 45.0, 39.0, 50.23148148148148, '85.00%', 3320, 57],
 ['No', 41.0, 38.0, 34.0, 35.0, 54.0, '91.30%', 366, 28],
 ['No', 74.0, 53.0, 63.0, 63.0, 55.0, '96.80%', 535, 60],
 ['No',
  87.0,
  48.0,
  53.0,
  49.05084745762712,
  50.23148148148148,
  '97.10%',
  784,
  28],
 ['No', 67.0, 46.0, 44.0, 71.0, 60.0, '96.30%', 1161, 56],
 ['No', 15.0, 17.0, 17.0, 49.05084745762712, 43.0, '93.10%', 1390, 66],
 ['No', 43.0, 37.0, 41.0, 49.05084745762712, 48.0, '93.90%', 417, 69],
 ['No', 40.0, 48.0, 43.0, 37.0, 45.0, '95.50%', 445, 23],
 ['No', 54.0, 37.0, 41.0, 88.0, 51.0, '95.10%', 826, 19],
 ['No', 99.0, 95.0, 80.0, 49.05084745762712, 59.0, '94.90%', 257, 6],
 ['No',
  49.50487329434698,
  47.76608187134503,
  48.28849902534113,
  53.0,
  50.23148148148148,
  '95.50%',
  188,
  48],
 ['No', 62.0, 49.0, 58.0, 49.05084745762712, 51.0, '95.50%', 860, 13],
 ['No', 55.0, 42.0, 38.0, 49.05084745762712, 45.0, '95.90%', 1423, 19],
 ['No', 57.0, 68.0, 73.0, 83.0, 56.0, '94.40%', 434, 27],
 ['No', 53.0, 53.0, 39.0, 46.0, 56.0, '93.30%', 362, 27],
 ['No', 50.0, 59.0, 59.0, 49.05084745762712, 50.0, '90.40%', 48, 29],
 ['No', 37.0, 34.0, 22.0, 49.05084745762712, 55.0, '90.90%', 271, 29],
 ['No', 49.0, 37.0, 32.0, 40.0, 50.0, '95.10%', 1408, 52],
 ['No', 27.0, 35.0, 49.0, 49.05084745762712, 56.0, '94.90%', 367, 73],
 ['No', 22.0, 32.0, 33.0, 49.05084745762712, 51.0, '91.80%', 241, 38],
 ['No', 35.0, 65.0, 66.0, 49.05084745762712, 50.0, '93.30%', 194, 27],
 ['No', 63.0, 35.0, 38.0, 57.0, 49.0, '94.60%', 705, 56],
 ['No',
  49.50487329434698,
  47.76608187134503,
  48.28849902534113,
  49.05084745762712,
  48.0,
  '79.10%',
  726,
  25],
 ['No', 15.0, 41.0, 46.0, 30.0, 48.0, '91.50%', 394, 49],
 ['No', 30.0, 30.0, 45.0, 49.05084745762712, 47.0, '91.30%', 658, 25],
 ['No', 64.0, 62.0, 66.0, 41.0, 52.0, '95.70%', 811, 2],
 ['No', 57.0, 31.0, 37.0, 62.0, 50.23148148148148, '91.40%', 222, 39],
 ['No',
  40.0,
  25.0,
  38.0,
  49.05084745762712,
  50.23148148148148,
  '84.20%',
  1584,
  75],
 ['No', 53.0, 66.0, 63.0, 95.0, 52.0, '93.60%', 349, 23],
 ['No', 28.0, 42.0, 14.0, 49.05084745762712, 46.0, '80.40%', 21, 28],
 ['No',
  84.0,
  58.0,
  38.0,
  49.05084745762712,
  50.23148148148148,
  '96.00%',
  909,
  74],
 ['No', 31.0, 42.0, 44.0, 19.0, 47.0, '93.30%', 290, 73],
 ['No', 48.0, 47.0, 42.0, 49.05084745762712, 56.0, '89.20%', 324, 30],
 ['No', 31.0, 41.0, 50.0, 54.0, 49.0, '92.10%', 775, 43],
 ['Yes',
  49.50487329434698,
  47.76608187134503,
  48.28849902534113,
  49.05084745762712,
  48.0,
  '95.50%',
  631,
  59],
 ['No', 55.0, 51.0, 56.0, 65.0, 55.0, '94.50%', 872, 64],
 ['No',
  48.0,
  73.0,
  83.0,
  49.05084745762712,
  50.23148148148148,
  '91.60%',
  200,
  29],
 ['No', 49.0, 80.0, 99.0, 49.05084745762712, 53.0, '92.90%', 268, 26],
 ['No', 43.0, 41.0, 46.0, 49.05084745762712, 48.0, '95.10%', 1032, 58],
 ['No', 67.0, 65.0, 60.0, 46.0, 56.0, '94.70%', 446, 33],
 ['No',
  52.0,
  33.0,
  42.0,
  49.05084745762712,
  50.23148148148148,
  '78.60%',
  1535,
  44],
 ['No',
  50.0,
  27.0,
  23.0,
  49.05084745762712,
  50.23148148148148,
  '80.20%',
  91,
  24],
 ['No',
  49.50487329434698,
  47.76608187134503,
  48.28849902534113,
  49.05084745762712,
  51.0,
  '96.20%',
  696,
  1],
['No', 17.0, 37.0, 25.0, 6.0, 50.23148148148148, '75.60%', 701, 46],
['No', 55.0, 50.0, 47.0, 34.0, 52.0, '96.20%', 1015, 16],
['No',
 48.0,
 45.0,
 42.0,
 49.05084745762712,
 50.23148148148148,
 '81.70%',
 1053,
 77],
['No', 48.0, 74.0, 73.0, 39.0, 50.23148148148148, '93.40%', 464, 68],
['No', 36.0, 61.0, 55.0, 49.05084745762712, 48.0, '93.90%', 435, 46],
['No', 67.0, 49.0, 35.0, 49.05084745762712, 51.0, '95.60%', 1004, 17],
['No',
 42.0,
 39.0,
 25.0,
 49.05084745762712,
 50.23148148148148,
 '88.60%',
 883,
 23],
['No', 67.0, 53.0, 51.0, 47.0, 50.0, '95.60%', 359, 14],
['No', 99.0, 99.0, 88.0, 49.05084745762712, 57.0, '95.70%', 1053, 13],
['No', 28.0, 30.0, 35.0, 51.0, 50.23148148148148, '74.60%', 1226, 73],
['No', 99.0, 81.0, 34.0, 84.0, 62.0, '89.80%', 298, 13],
['No',
 34.0,
 17.0,
 13.0,
 49.05084745762712,
 50.23148148148148,
 '93.40%',
 756,
 19],
['No', 99.0, 69.0, 68.0, 32.0, 57.0, '95.50%', 359, 10],
['No',
 56.0,
 39.0,
 24.0,
 49.05084745762712,
 50.23148148148148,
 '94.20%',
 558,
 24],
['No', 70.0, 47.0, 40.0, 57.0, 50.0, '96.40%', 834, 17],
['No', 30.0, 48.0, 54.0, 47.0, 46.0, '92.60%', 413, 44],
['No', 21.0, 23.0, 41.0, 49.05084745762712, 48.0, '87.00%', 334, 68],
['No', 99.0, 88.0, 71.0, 49.05084745762712, 60.0, '96.50%', 648, 10],
['No', 55.0, 34.0, 44.0, 49.05084745762712, 50.0, '97.60%', 636, 31],
['No',
 49.50487329434698,
 47.76608187134503,
 48.28849902534113,
 49.05084745762712,
 50.23148148148148,
 '66.30%',
 831,
 23],
['No', 18.0, 28.0, 27.0, 49.05084745762712, 41.0, '92.20%', 361, 51],
['No', 35.0, 47.0, 53.0, 39.0, 47.0, '92.70%', 524, 25],
['No',
 67.0,
 35.0,
 46.0,
 49.05084745762712,
 50.23148148148148,
 '95.00%',
 605,
 7],
['No',
 49.50487329434698,
 47.76608187134503,
 48.28849902534113,
 49.05084745762712,
 42.0,
 '92.00%',
 505,
 23],
['No', 28.0, 42.0, 48.0, 67.0, 55.0, '94.90%', 352, 69],
['No',
 49.50487329434698,
 47.76608187134503,
 48.28849902534113,
 49.05084745762712,
 42.0,
 '89.80%',
 348,
 43],
['No', 45.0, 46.0, 43.0, 45.0, 52.0, '95.10%', 650, 16],
['No', 51.0, 45.0, 54.0, 83.0, 48.0, '93.60%', 512, 71],
['No', 29.0, 31.0, 37.0, 49.0, 50.23148148148148, '76.90%', 1456, 35],
['No', 36.0, 42.0, 56.0, 49.05084745762712, 49.0, '92.40%', 320, 69],
['No',
 29.0,
 33.0,
 40.0,
 49.05084745762712,
 50.23148148148148,
 '73.50%',
 733,
 68],
['No', 28.0, 47.0, 57.0, 42.0, 47.0, '92.10%', 369, 68],
['No', 46.0, 34.0, 55.0, 58.0, 50.23148148148148, '96.20%', 269, 35],
['No', 60.0, 35.0, 30.0, 54.0, 47.0, '95.80%', 832, 15],
['No',
 49.50487329434698,
 47.76608187134503,
 48.28849902534113,
 49.05084745762712,
 50.23148148148148,
 '96.20%',
 709,
 31],
['No', 81.0, 68.0, 62.0, 64.0, 53.0, '96.40%', 735, 2],
['No', 36.0, 34.0, 14.0, 49.05084745762712, 53.0, '93.90%', 365, 60],
['No', 66.0, 34.0, 37.0, 49.05084745762712, 56.0, '95.60%', 533, 41],
['No', 33.0, 31.0, 27.0, 49.05084745762712, 45.0, '92.10%', 364, 56],
['No', 53.0, 59.0, 78.0, 49.05084745762712, 52.0, '92.10%', 461, 28],
['No', 43.0, 49.0, 55.0, 37.0, 50.23148148148148, '96.00%', 447, 31],
['No', 62.0, 46.0, 32.0, 49.05084745762712, 48.0, '94.20%', 1139, 15],
['No', 68.0, 54.0, 57.0, 23.0, 47.0, '96.10%', 864, 22],
['No', 38.0, 45.0, 56.0, 36.0, 50.0, '97.60%', 1222, 63],
['No',
 49.50487329434698,
 47.76608187134503,
 48.28849902534113,
 49.05084745762712,
 48.0,
 '94.90%',
 489,
 53],
['No', 46.0, 42.0, 41.0, 49.05084745762712, 53.0, '94.90%', 465, 6],
['No', 90.0, 52.0, 39.0, 52.0, 53.0, '86.30%', 137, 33],
['No', 48.0, 38.0, 41.0, 49.0, 49.0, '95.30%', 1503, 57],
['No',
 59.0,
 46.0,
 35.0,
 49.05084745762712,
 50.23148148148148,
 '95.00%',
 832,
 65],
['No', 29.0, 37.0, 46.0, 42.0, 46.0, '94.90%', 797, 61],
['No', 44.0, 49.0, 49.0, 32.0, 46.0, '94.40%', 402, 71],
['No', 43.0, 50.0, 48.0, 49.05084745762712, 47.0, '57.90%', 478, 28],
['No', 42.0, 66.0, 73.0, 49.05084745762712, 51.0, '93.70%', 690, 22],
['No', 51.0, 43.0, 42.0, 41.0, 43.0, '81.10%', 1532, 4],
['No', 68.0, 63.0, 58.0, 49.05084745762712, 55.0, '96.00%', 474, 48],
['No', 37.0, 64.0, 71.0, 49.05084745762712, 51.0, '89.80%', 460, 25],
['No', 46.0, 63.0, 63.0, 39.0, 45.0, '95.10%', 466, 61],
['No',
 63.0,
 54.0,
 36.0,
 49.05084745762712,
 50.23148148148148,
 '96.60%',
 1384,
 60],
['No', 19.0, 40.0, 51.0, 30.0, 50.23148148148148, '90.70%', 371, 51],
['No', 79.0, 56.0, 59.0, 71.0, 53.0, '95.20%', 486, 64],
['No', 66.0, 75.0, 85.0, 49.05084745762712, 55.0, '92.70%', 916, 67],
['No', 48.0, 51.0, 65.0, 49.05084745762712, 51.0, '94.40%', 250, 28],
['No', 44.0, 39.0, 34.0, 25.0, 50.23148148148148, '69.60%', 1016, 24],
['No', 55.0, 39.0, 37.0, 49.05084745762712, 45.0, '94.80%', 319, 24],
['No', 30.0, 34.0, 34.0, 33.0, 44.0, '81.60%', 826, 1],
['No', 31.0, 52.0, 33.0, 45.0, 52.0, '93.80%', 283, 53],
['No', 31.0, 56.0, 59.0, 20.0, 47.0, '90.70%', 422, 25],
['No', 36.0, 34.0, 20.0, 31.0, 46.0, '96.30%', 525, 30],
['No', 37.0, 44.0, 58.0, 36.0, 51.0, '92.40%', 479, 29],
['No', 52.0, 42.0, 42.0, 49.05084745762712, 53.0, '94.70%', 445, 73],
['No', 64.0, 54.0, 53.0, 49.05084745762712, 52.0, '95.10%', 444, 8],
['Yes', 64.0, 55.0, 58.0, 70.0, 55.0, '95.10%', 542, 10],
['No', 55.0, 59.0, 39.0, 27.0, 50.23148148148148, '95.40%', 504, 22],
['No', 34.0, 48.0, 52.0, 59.0, 49.0, '93.60%', 285, 53],
['No',
 99.0,
 76.0,
 61.0,
 49.05084745762712,
 50.23148148148148,
 '96.30%',
 549,
 12],
['No', 35.0, 44.0, 53.0, 34.0, 52.0, '95.00%', 538, 71],
['Yes',
 49.50487329434698,
 47.76608187134503,
 48.28849902534113,
 49.05084745762712,
 52.0,
 '93.90%',
 862,
 20],
['No', 57.0, 41.0, 39.0, 29.0, 49.0, '96.80%', 1846, 63],
['No', 30.0, 38.0, 22.0, 49.05084745762712, 48.0, '90.90%', 291, 68],
['No',
 35.0,
 51.0,
 54.0,
 49.05084745762712,
 50.23148148148148,
 '93.00%',
 354,
 67],
['No',
 49.50487329434698,
 47.76608187134503,
 48.28849902534113,
 49.05084745762712,
 64.0,
 '96.40%',
 242,
 8],
['No',
 49.50487329434698,
 47.76608187134503,
 48.28849902534113,
 49.05084745762712,
 44.0,
 '95.80%',
 1186,
 63],
['No', 17.0, 47.0, 54.0, 49.05084745762712, 46.0, '89.50%', 352, 53],
['No', 84.0, 34.0, 36.0, 46.0, 54.0, '95.30%', 795, 33],
['No',
 49.50487329434698,
 47.76608187134503,
 48.28849902534113,
 49.05084745762712,
 56.0,
 '93.60%',
 255,
 43],
['No',
 49.50487329434698,
 47.76608187134503,
 48.28849902534113,
 60.0,
 63.0,
 '85.50%',
 201,
 67],
['No', 35.0, 57.0, 71.0, 56.0, 50.0, '92.50%', 811, 25],
['No', 70.0, 92.0, 87.0, 49.05084745762712, 56.0, '95.60%', 205, 30],
['No', 99.0, 99.0, 99.0, 99.0, 52.0, '96.20%', 289, 2],
['No', 39.0, 35.0, 27.0, 29.0, 45.0, '92.10%', 532, 1],
['No', 72.0, 52.0, 44.0, 66.0, 50.23148148148148, '95.60%', 1000, 77],
['No', 58.0, 48.0, 47.0, 56.0, 50.23148148148148, '87.60%', 1657, 2],
['No', 78.0, 55.0, 43.0, 49.05084745762712, 55.0, '96.40%', 646, 2],
['No',
 49.50487329434698,
 47.76608187134503,
 48.28849902534113,
 49.05084745762712,
 50.23148148148148,
 '96.60%',
 351,
 28],
['No', 99.0, 99.0, 99.0, 81.0, 57.0, '96.20%', 376, 63],
['No', 48.0, 39.0, 40.0, 49.05084745762712, 49.0, '95.90%', 1070, 66],
 ['No', 45.0, 55.0, 53.0, 68.0, 58.0, '87.30%', 440, 68],
 ['No', 42.0, 19.0, 22.0, 69.0, 55.0, '95.40%', 285, 30],
 ['No',
  41.0,
  53.0,
  65.0,
  49.05084745762712,
  50.23148148148148,
  '90.80%',
  512,
  29],
 ['No',
  42.0,
  42.0,
  37.0,
  49.05084745762712,
  50.23148148148148,
  '82.30%',
  1490,
  14],
 ['No', 46.0, 31.0, 53.0, 49.05084745762712, 41.0, '92.10%', 185, 50],
 ['No', 91.0, 64.0, 56.0, 49.05084745762712, 55.0, '96.60%', 269, 14],
 ['No', 33.0, 67.0, 78.0, 49.05084745762712, 51.0, '91.20%', 307, 37],
 ['No', 34.0, 39.0, 45.0, 78.0, 50.23148148148148, '94.40%', 288, 29],
 ['No', 72.0, 59.0, 46.0, 49.05084745762712, 58.0, '95.20%', 352, 22],
 ['No', 67.0, 77.0, 73.0, 66.0, 50.23148148148148, '94.80%', 288, 48],
 ['No',
  49.50487329434698,
  47.76608187134503,
  48.28849902534113,
  49.05084745762712,
  49.0,
  '95.20%',
  369,
  53],
 ['No', 85.0, 56.0, 54.0, 49.05084745762712, 52.0, '96.10%', 590, 4],
 ['No', 36.0, 36.0, 35.0, 57.0, 50.23148148148148, '89.30%', 2883, 58],
 ['No',
  39.0,
  11.0,
  1.0,
  49.05084745762712,
  50.23148148148148,
  '94.00%',
  435,
  16],
 ['No', 56.0, 48.0, 51.0, 66.0, 50.23148148148148, '95.80%', 373, 49],
 ['No', 50.0, 61.0, 62.0, 53.0, 50.23148148148148, '88.70%', 473, 3],
 ['No',
  49.50487329434698,
  47.76608187134503,
  48.28849902534113,
  49.05084745762712,
  55.0,
  nan,
  211,
  59],
 ['No', 50.0, 64.0, 72.0, 50.0, 50.0, '92.80%', 398, 28],
 ['No', 62.0, 51.0, 64.0, 87.0, 50.23148148148148, '95.50%', 330, 55],
 ['No',
  42.0,
  52.0,
  52.0,
  49.05084745762712,
  50.23148148148148,
  '84.50%',
  410,
  25],
 ['No',
  49.50487329434698,
  47.76608187134503,
  48.28849902534113,
  49.05084745762712,
  50.23148148148148,
  '96.40%',
  1651,
  3],
 ['No', 71.0, 52.0, 66.0, 44.0, 54.0, '95.10%', 558, 7],
 ['No',
  98.0,
  80.0,
  77.0,
  49.05084745762712,
  50.23148148148148,
  '93.40%',
  881,
  8],
 ['No', 25.0, 7.0, 11.0, 57.0, 48.0, '95.50%', 359, 71],
 ['No', 29.0, 62.0, 68.0, 49.05084745762712, 47.0, '92.40%', 44, 68],
 ['No',
  49.50487329434698,
  47.76608187134503,
  48.28849902534113,
  64.0,
  52.0,
  '86.10%',
  279,
  63],
 ['No',
  55.0,
  43.0,
  45.0,
  49.05084745762712,
  50.23148148148148,
  '95.50%',
  1047,
  15],
 ['No', 43.0, 34.0, 44.0, 49.05084745762712, 48.0, '95.60%', 482, 28],
 ['No', 51.0, 41.0, 38.0, 49.0, 50.23148148148148, '74.10%', 633, 24],
 ['No', 32.0, 47.0, 53.0, 49.05084745762712, 53.0, '93.90%', 254, 73],
 ['No',
  49.50487329434698,
  47.76608187134503,
  48.28849902534113,
  79.0,
  50.23148148148148,
  '63.00%',
  590,
  35],
 ['No', 22.0, 1.0, 1.0, 29.0, 42.0, '92.10%', 363, 50],
 ['No', 29.0, 36.0, 40.0, 49.05084745762712, 46.0, '93.70%', 686, 23],
 ['No', 23.0, 55.0, 47.0, 50.0, 47.0, '91.20%', 343, 53],
 ['No', 67.0, 41.0, 28.0, 49.05084745762712, 46.0, '94.90%', 659, 2],
 ['No', 95.0, 69.0, 67.0, 46.0, 53.0, '94.60%', 2166, 28],
 ['No', 99.0, 77.0, 67.0, 45.0, 55.0, '96.20%', 414, 12],
 ['No', 69.0, 78.0, 81.0, 83.0, 56.0, '94.90%', 234, 27],
 ['No', 18.0, 30.0, 29.0, 49.05084745762712, 45.0, '91.70%', 321, 68],
 ['No', 49.0, 39.0, 56.0, 47.0, 58.0, '96.90%', 243, 70],
 ['No',
  49.50487329434698,
  47.76608187134503,
  48.28849902534113,
   49.05084745762712,
   50.23148148148148,
   '95.10%',
   722,
   8],
  ['No', 70.0, 74.0, 61.0, 52.0, 50.23148148148148, '96.10%', 805, 3],
  ['No', 48.0, 37.0, 26.0, 15.0, 45.0, '93.50%', 311, 39],
  ['No', 47.0, 33.0, 42.0, 49.05084745762712, 50.0, '93.30%', 184, 54],
  ['No', 67.0, 60.0, 78.0, 36.0, 50.0, '95.50%', 843, 17],
  ['No', 56.0, 50.0, 52.0, 56.0, 49.0, '95.20%', 588, 31],
  ['No',
   49.50487329434698,
   47.76608187134503,
   48.28849902534113,
   49.05084745762712,
   49.0,
   '96.50%',
   1239,
   14],
  ['No', 36.0, 58.0, 78.0, 73.0, 46.0, '92.80%', 217, 28],
  ['No',
   47.0,
   44.0,
   47.0,
   49.05084745762712,
   50.23148148148148,
   '92.40%',
   264,
   28],
  ['No',
   61.0,
   37.0,
   32.0,
   49.05084745762712,
   50.23148148148148,
   '89.10%',
   2922,
   10],
  ['No', 90.0, 51.0, 24.0, 49.05084745762712, 50.0, '93.50%', 240, 7],
  ['No', 69.0, 22.0, 12.0, 49.05084745762712, 50.0, '95.10%', 654, 41],
  ['No',
   34.0,
   42.0,
   50.0,
   49.05084745762712,
   50.23148148148148,
   '93.00%',
   302,
   71],
  ['No', 40.0, 38.0, 41.0, 63.0, 49.0, '95.70%', 896, 61],
  ['No', 20.0, 18.0, 29.0, 49.05084745762712, 41.0, '75.20%', 1656, 70],
  ['No', 52.0, 62.0, 65.0, 47.0, 50.0, '95.30%', 453, 35],
  ['No', 64.0, 29.0, 27.0, 39.0, 47.0, '95.70%', 687, 10],
  ['No',
   92.0,
   64.0,
   67.0,
   49.05084745762712,
   50.23148148148148,
   '93.80%',
   871,
   32],
  ['No', 27.0, 35.0, 40.0, 33.0, 47.0, '90.30%', 527, 46],
  ['No', 78.0, 99.0, 99.0, 49.05084745762712, 52.0, '91.80%', 423, 29],
  ['No', 57.0, 44.0, 35.0, 49.05084745762712, 47.0, '95.60%', 1345, 15],
  ['No', 52.0, 70.0, 62.0, 49.05084745762712, 46.0, '95.60%', 1061, 20],
  ['No', 22.0, 39.0, 42.0, 31.0, 49.0, '73.00%', 621, 67],
  ['No',
   49.50487329434698,
   47.76608187134503,
   48.2884990253411З,
   49.05084745762712,
   50.0,
   '91.60%',
   266,
   35],
  ['No', 49.0, 42.0, 48.0, 49.05084745762712, 53.0, '93.30%', 125, 35],
  ['No', 32.0, 46.0, 55.0, 49.05084745762712, 49.0, '92.30%', 462, 61],
  ['No', 13.0, 33.0, 35.0, 62.0, 46.0, '91.20%', 371, 40],
  ['No', 41.0, 56.0, 32.0, 49.05084745762712, 50.0, '95.20%', 748, 22],
  ['No', 70.0, 80.0, 66.0, 49.05084745762712, 59.0, '93.90%', 238, 42],
  ['No', 51.0, 49.0, 47.0, 49.0, 50.0, '91.60%', 382, 30]]
```

In [124…

```
conn = sqlite3.connect("Chicago.db")

cursor = conn.cursor()

try:
  cursor.executemany('''
        INSERT INTO Chicago (HEALTHY_SCHOOL_CERTIFIED, SAFETY_SCORE, EnvironmentScore,
        InstructionScore, TeachersScore, ParentEngagementScore,AVERAGE_STUDENT_ATTENDANCE,
         COLLEGE_ENROLLMENT, COMMUNITY_AREA_NUMBER)
        VALUES (?,?,?,?,?,?,?,?,?)''', Chicago_list)
  conn.commit()
  print("Data Insterted successfully")
except Exception as e:
  print(str(e))
  print("Insertion failed!")
finally:
  conn.close()
```

```
Data Insterted successfully
```

In [125…

```
conn = sqlite3.connect("Chicago.db")
cursor = conn.cursor()
cursor.execute("SELECT count(*) as Total FROM Chicago;")
rows = cursor.fetchall()
for row in rows:
  print(row)
```

```
(566,)
```

In [126…

```
conn = sqlite3.connect("Chicago.db")
cursor = conn.cursor()
cursor.execute("SELECT HEALTHY_SCHOOL_CERTIFIED, count(*) FROM Chicago GROUP BY HEALTHY_SC
rows = cursor.fetchall()
for row in rows:
  print(row)
```

```
('No', 550)
('Yes', 16)
```

In [127…

```
conn = sqlite3.connect("Chicago.db")
df = pd.read_sql_query("SELECT COMMUNITY_AREA_NUMBER, count(*) FROM Chicago;", conn)
df.head()
```

Out[127…

	COMMUNITY_AREA_NUMBER	count(*)

	COMMUNITY_AREA_NUMBER	count(*)
0	7	566

In [128…

```
conn = sqlite3.connect("Chicago.db")

try:
  df = pd.read_sql_query('''
        SELECT HEALTHY_SCHOOL_CERTIFIED,
               count(*) as N,
               avg(SAFETY_SCORE) as mean_safety,
               max(SAFETY_SCORE) as max_safety,
               min(SAFETY_SCORE) as min_safety
        FROM Chicago GROUP BY HEALTHY_SCHOOL_CERTIFIED ORDER BY mean_safety;''', conn)
except Exception as e:
  print(str(e))
finally:
  conn.close()

df
```

Out[128…

	HEALTHY_SCHOOL_CERTIFIED	N	mean_safety	max_safety	min_safety
0	No	550	49.254988	99.0	1.0
1	Yes	16	58.094664	99.0	44.0

Census table to Chicago.db

In [129…

```
conn = sqlite3.connect("Chicago.db")
print(conn)
```

```
<sqlite3.Connection object at 0x000002D531E0C3F0>
```

In [130…

```
conn.execute("DROP TABLE IF EXISTS `Census`")
print("Table dropped")
```

```
Table dropped
```

In [131…

```
conn = sqlite3.connect("Chicago.db")
try:
  conn.execute('''
      CREATE TABLE Census
      (PERCENT_OF_HOUSING_CROWDED    FLOAT DEFAULT 0,
       PERCENTHOUSEHOLDSBELOWPOVERTY     FLOAT DEFAULT 0,
       PERCENTAGED16UNEMPLOYED     FLOAT DEFAULT 0,
       PERCENTAGED25WITHOUTHIGHSCHOOLDIPLOMA     FLOAT DEFAULT 0,
       PERCENTAGEDUNDER18OROVER64     FLOAT DEFAULT 0,
       PER_CAPITA_INCOME              FLOAT DEFAULT 0,
       HARDSHIP_INDEX   FLOAT DEFAULT 0,
       COMMUNITY_AREA_NUMBER        INTEGER,
       FOREIGN KEY (COMMUNITY_AREA_NUMBER) REFERENCES communities
       );''')
  print("Table created successfully")
except Exception as e:
  print(str(e))
  print('Table creation failed!!!')
finally:
  conn.close()
```

```
Table created successfully
```

In [132…

```python
Census_list = Census_drop.values.tolist()
Census_list
```

Out[132…

```
[[1.0, 7.7, 23.6, 8.7, 18.2, 27.5, 23939.0, 39.0],
 [2.0, 7.8, 17.2, 8.8, 20.8, 38.5, 23040.0, 46.0],
 [3.0, 3.8, 24.0, 8.9, 11.8, 22.2, 35787.0, 20.0],
 [4.0, 3.4, 10.9, 8.2, 13.4, 25.5, 37524.0, 17.0],
 [5.0, 0.3, 7.5, 5.2, 4.5, 26.2, 57123.0, 6.0],
 [6.0, 1.1, 11.4, 4.7, 2.6, 17.0, 60058.0, 5.0],
 [7.0, 0.8, 12.3, 5.1, 3.6, 21.5, 71551.0, 2.0],
 [8.0, 1.9, 12.9, 7.0, 2.5, 22.6, 88669.0, 1.0],
 [9.0, 1.1, 3.3, 6.5, 7.4, 35.3, 40959.0, 8.0],
 [10.0, 2.0, 5.4, 9.0, 11.5, 39.5, 32875.0, 21.0],
 [11.0, 2.7, 8.6, 12.4, 13.4, 35.5, 27751.0, 25.0],
 [12.0, 1.1, 7.5, 6.8, 4.9, 40.5, 44164.0, 11.0],
 [13.0, 3.9, 13.2, 9.9, 14.4, 39.0, 26576.0, 33.0],
 [14.0, 11.3, 19.2, 10.0, 32.9, 32.0, 21323.0, 53.0],
 [15.0, 4.1, 11.6, 12.6, 19.3, 34.0, 24336.0, 35.0],
 [16.0, 6.3, 13.1, 10.0, 22.4, 31.6, 27249.0, 34.0],
 [17.0, 5.2, 10.6, 10.0, 16.2, 33.6, 26282.0, 28.0],
 [18.0, 8.1, 15.3, 13.8, 23.5, 38.6, 22014.0, 50.0],
 [19.0, 10.8, 18.7, 14.6, 37.3, 37.3, 15461.0, 70.0],
 [20.0, 6.9, 20.5, 13.1, 41.6, 36.4, 15089.0, 71.0],
 [21.0, 6.0, 15.3, 9.2, 24.7, 31.0, 20039.0, 42.0],
 [22.0, 3.2, 16.8, 8.2, 14.8, 26.2, 31908.0, 23.0],
 [23.0, 14.8, 33.9, 17.3, 35.4, 38.0, 13781.0, 85.0],
 [24.0, 2.3, 14.7, 6.6, 12.9, 21.7, 43198.0, 10.0],
 [25.0, 6.3, 28.6, 22.6, 24.4, 37.9, 15957.0, 73.0],
 [26.0, 9.4, 41.7, 25.8, 24.5, 43.6, 10934.0, 92.0],
 [27.0, 8.2, 42.4, 19.6, 21.3, 43.2, 12961.0, 83.0],
 [28.0, 3.8, 20.6, 10.7, 9.6, 22.2, 44689.0, 15.0],
 [29.0, 7.4, 43.1, 21.2, 27.6, 42.7, 12034.0, 87.0],
 [30.0, 15.2, 30.7, 15.8, 54.8, 33.8, 10402.0, 96.0],
 [31.0, 9.6, 25.8, 15.8, 40.7, 32.6, 16444.0, 76.0],
 [32.0, 1.5, 14.7, 5.7, 3.1, 13.5, 65526.0, 3.0],
 [33.0, 1.3, 13.8, 4.9, 7.4, 21.8, 59077.0, 7.0],
 [34.0, 5.7, 40.1, 16.7, 34.5, 38.3, 16148.0, 82.0],
 [35.0, 1.8, 29.6, 18.2, 14.3, 30.7, 23791.0, 47.0],
 [36.0, 1.3, 39.7, 28.7, 18.4, 40.4, 19252.0, 78.0],
 [37.0, 3.2, 51.2, 33.9, 26.6, 44.9, 10432.0, 97.0],
 [38.0, 3.3, 29.3, 24.3, 15.9, 39.5, 23472.0, 57.0],
 [39.0, 2.4, 21.7, 15.7, 11.3, 35.4, 35911.0, 26.0],
 [40.0, 5.6, 42.1, 28.6, 25.4, 42.8, 13785.0, 88.0],
 [41.0, 1.5, 18.4, 8.4, 4.3, 26.2, 39056.0, 14.0],
 [42.0, 2.9, 30.7, 23.4, 16.5, 36.1, 18672.0, 58.0],
 [43.0, 2.8, 31.1, 20.0, 14.0, 35.7, 19398.0, 55.0],
 [44.0, 3.3, 27.8, 24.0, 14.5, 40.3, 18881.0, 60.0],
 [45.0, 1.4, 17.2, 21.1, 10.6, 39.3, 24454.0, 41.0],
 [46.0, 4.7, 29.8, 19.7, 26.6, 41.1, 16579.0, 75.0],
 [47.0, 6.8, 33.0, 18.6, 19.3, 42.7, 12515.0, 79.0],
 [48.0, 2.1, 11.5, 20.0, 11.0, 44.0, 28887.0, 38.0],
 [49.0, 2.5, 19.8, 20.3, 16.9, 41.2, 17949.0, 52.0],
 [50.0, 1.5, 21.6, 22.8, 13.1, 38.6, 20588.0, 51.0],
 [51.0, 4.0, 29.2, 16.3, 21.0, 39.5, 14685.0, 65.0],
 [52.0, 6.8, 19.2, 12.1, 31.9, 42.8, 17104.0, 64.0],
 [53.0, 3.3, 25.9, 19.4, 20.5, 42.1, 16563.0, 62.0],
 [54.0, 5.8, 56.5, 34.6, 27.5, 51.5, 8201.0, 98.0],
 [55.0, 3.3, 17.1, 9.6, 19.2, 42.9, 22677.0, 44.0],
 [56.0, 2.6, 8.8, 11.3, 19.3, 38.1, 26353.0, 32.0],
 [57.0, 8.5, 14.1, 16.5, 35.9, 39.2, 16134.0, 67.0],
 [58.0, 14.4, 23.6, 13.9, 45.1, 39.3, 13089.0, 84.0],
 [59.0, 7.2, 18.7, 13.4, 32.9, 35.6, 16954.0, 61.0],
 [60.0, 4.5, 18.9, 13.7, 22.2, 31.3, 22694.0, 43.0],
 [61.0, 11.9, 29.0, 23.0, 41.5, 38.9, 12765.0, 91.0],
 [62.0, 11.1, 15.6, 16.7, 37.0, 37.7, 15754.0, 69.0],
 [63.0, 15.8, 23.4, 18.2, 51.5, 38.8, 12171.0, 93.0],
 [64.0, 2.7, 8.9, 9.5, 18.8, 37.6, 25113.0, 29.0],
 [65.0, 5.8, 14.9, 9.6, 33.6, 39.6, 16907.0, 56.0],
 [66.0, 7.6, 27.9, 17.1, 31.2, 40.6, 13231.0, 80.0],
 [67.0, 4.8, 34.4, 35.9, 26.3, 40.7, 11317.0, 89.0],
 [68.0, 3.8, 46.6, 28.0, 28.5, 42.5, 11888.0, 94.0],
 [69.0, 3.6, 29.6, 23.0, 16.5, 41.0, 17285.0, 66.0],
 [70.0, 4.0, 10.4, 11.7, 17.7, 36.9, 23482.0, 37.0],
 [71.0, 4.0, 27.6, 28.3, 18.5, 41.9, 15528.0, 74.0],
 [72.0, 0.9, 5.1, 8.0, 3.7, 40.5, 39523.0, 12.0],
 [73.0, 1.1, 16.9, 20.8, 13.7, 42.6, 19713.0, 48.0],
 [74.0, 1.0, 3.4, 8.7, 4.3, 36.8, 34381.0, 16.0],
 [75.0, 0.8, 13.2, 15.0, 10.8, 40.3, 27149.0, 30.0],
 [76.0, 3.6, 15.4, 7.1, 10.9, 30.3, 25828.0, 24.0],
 [77.0, 4.1, 18.2, 9.2, 9.7, 23.8, 33385.0, 19.0],
 [nan, 4.7, 19.7, 12.9, 19.5, 33.5, 28202.0, nan]]
```

In [133…

```
conn = sqlite3.connect("Chicago.db")

cursor = conn.cursor()

try:
  cursor.executemany('''
        INSERT INTO Census (COMMUNITY_AREA_NUMBER, PERCENT_OF_HOUSING_CROWDED,
        PERCENTHOUSEHOLDSBELOWPOVERTY, PERCENTAGED16UNEMPLOYED, PERCENTAGED25WITHOUTHIGHSC
         PER_CAPITA_INCOME, HARDSHIP_INDEX)
        VALUES (?,?,?,?,?,?,?,?)''', Census_list)
  conn.commit()
  print("Data Insterted successfully")
except Exception as e:
  print(str(e))
  print("Insertion failed!")
finally:
  conn.close()
```

```
Data Insterted successfully
```

In [134…

```
conn = sqlite3.connect("Chicago.db")
cursor = conn.cursor()
cursor.execute("SELECT COMMUNITY_AREA_NUMBER FROM Census;")
rows = cursor.fetchall()
for row in rows:
  print(row)
```

```
(1,)
(2,)
(3,)
(4,)
(5,)
(6,)
(7,)
(8,)
(9,)
(10,)
(11,)
(12,)
(13,)
(14,)
(15,)
(16,)
(17,)
```

```
(18,)
(19,)
(20,)
(21,)
(22,)
(23,)
(24,)
(25,)
(26,)
(27,)
(28,)
(29,)
(30,)
(31,)
(32,)
(33,)
(34,)
(35,)
(36,)
(37,)
(38,)
(39,)
(40,)
(41,)
(42,)
(43,)
(44,)
(45,)
(46,)
(47,)
(48,)
(49,)
(50,)
(51,)
(52,)
(53,)
(54,)
(55,)
(56,)
(57,)
(58,)
(59,)
(60,)
(61,)
(62,)
(63,)
(64,)
(65,)
(66,)
(67,)
(68,)
(69,)
(70,)
(71,)
(72,)
(73,)
(74,)
(75,)
(76,)
(77,)
(None,)
```

Inner join Chicago and Census tables with SQLite3

In [135…
```
conn = sqlite3.connect("Chicago.db")
```

In [136…
```
cursor=conn.cursor()
```

In [137…
```
sql='''SELECT * FROM Chicago
INNER JOIN Census ON Chicago.COMMUNITY_AREA_NUMBER = Census.COMMUNITY_AREA_NUMBER;'''
```

In [138…
```
cursor.execute(sql)
```

Out[138…
```
<sqlite3.Cursor at 0x2d533391340>
```

In [139…
```
merge=pd.DataFrame(cursor.fetchall())
merge.columns=[x[0] for x in cursor.description]
conn.close()
```

In [140…
```
merge
```

Out[140…

	SchoolID	HEALTHY_SCHOOL_CERTIFIED	SAFETY_SCORE	EnvironmentScore	InstructionScore	TeachersScore	Par
0	1	Yes	99.0	74.0	66.0	70.000000	
1	2	No	54.0	74.0	84.0	76.000000	
2	3	No	61.0	50.0	36.0	49.050847	
3	4	No	56.0	45.0	37.0	48.000000	
4	5	Yes	49.0	60.0	55.0	54.000000	
...	...	...	...	...	...	...	
561	562	No	32.0	46.0	55.0	49.050847	
562	563	No	13.0	33.0	35.0	62.000000	
563	564	No	41.0	56.0	32.0	49.050847	
564	565	No	70.0	80.0	66.0	49.050847	
565	566	No	51.0	49.0	47.0	49.000000	

566 rows × 18 columns

In [ ]:
```

```

Third Task: Querying from the Database and Discovering Relationships

Ask 10 questions

Using merged data (Chicago and Census) from sqlite3

In [141…

```
merge.head(30)
```

Out[141…

	SchoolID	HEALTHY_SCHOOL_CERTIFIED	SAFETY_SCORE	EnvironmentScore	InstructionScore	TeachersScore	Pare
0	1	Yes	99.0	74.0	66.0	70.000000	
1	2	No	54.0	74.0	84.0	76.000000	
2	3	No	61.0	50.0	36.0	49.050847	
3	4	No	56.0	45.0	37.0	48.000000	
4	5	Yes	49.0	60.0	55.0	54.000000	
5	6	No	66.0	66.0	71.0	50.000000	
6	7	No	88.0	62.0	52.0	49.050847	
7	8	No	67.0	30.0	18.0	49.050847	
8	9	No	70.0	67.0	51.0	49.050847	
9	10	No	43.0	28.0	37.0	56.000000	
10	11	No	99.0	64.0	46.0	51.000000	
11	12	No	49.0	31.0	33.0	41.000000	
12	13	No	73.0	60.0	59.0	48.000000	
13	14	No	31.0	32.0	45.0	49.050847	
14	15	No	28.0	58.0	60.0	49.050847	
15	16	No	19.0	22.0	13.0	19.000000	
16	17	No	37.0	37.0	35.0	36.000000	
17	18	No	46.0	39.0	58.0	49.050847	
18	19	No	52.0	51.0	53.0	49.050847	
19	20	No	45.0	32.0	28.0	49.050847	
20	21	No	81.0	60.0	73.0	54.000000	
21	22	No	38.0	27.0	35.0	26.000000	
22	23	No	23.0	35.0	42.0	42.000000	
23	24	Yes	57.0	12.0	14.0	49.050847	
24	25	No	48.0	37.0	63.0	31.000000	
25	26	No	86.0	57.0	46.0	49.050847	
26	27	No	31.0	54.0	32.0	50.000000	
27	28	No	99.0	85.0	82.0	82.000000	
28	29	No	59.0	58.0	61.0	49.050847	
29	30	No	32.0	64.0	76.0	49.050847	

In [142…

```
merge.shape
```

Out[142…

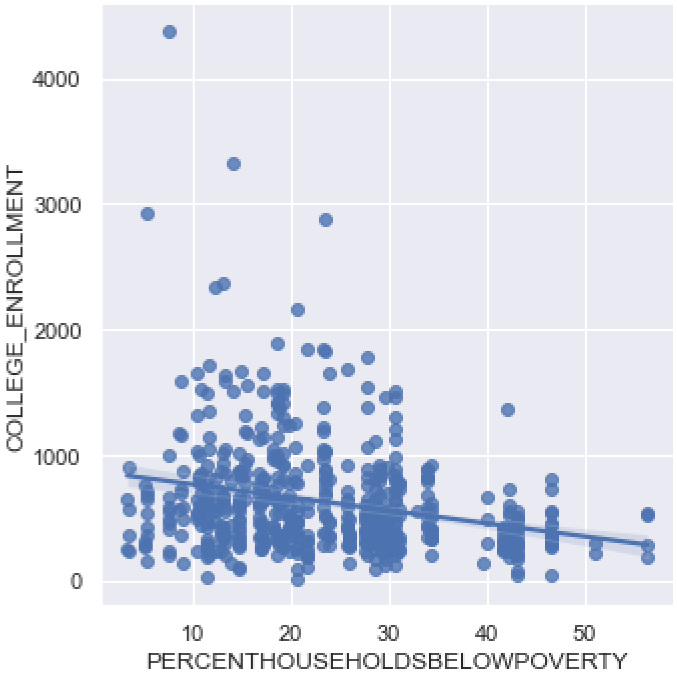
```
(566, 18)
```

1.What is the relationship between percent households below poverty and college enrollment in a certain community area?

Based on the following scatterplot, there is a negative correlation between college enrollment and percent households below poverty, the higher the percentage of households below poverty, the lower the college enrollment.

```
In [143…
sns.lmplot(x='PERCENTHOUSEHOLDSBELOWPOVERTY', y='COLLEGE_ENROLLMENT', data=merge)
```

```
Out[143…
<seaborn.axisgrid.FacetGrid at 0x2d5332ae7c0>
```

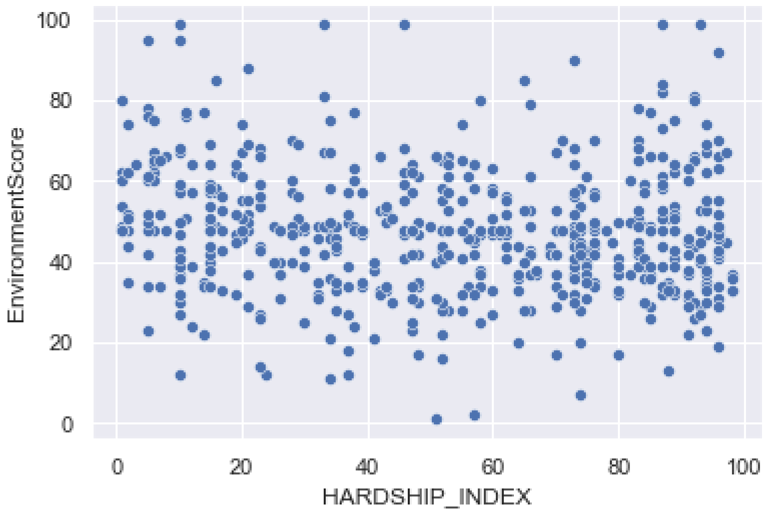


2.What is the relationship between hardship index and environment score in a certain community area?

There seems to be no relationship between hardship index and environment score in that community area.

```
In [144…
sns.scatterplot(x='HARDSHIP_INDEX', y='EnvironmentScore', data=merge)
```

```
Out[144…
<AxesSubplot:xlabel='HARDSHIP_INDEX', ylabel='EnvironmentScore'>
```



3.Is there a relationship between safety score and percent of housing crowded?

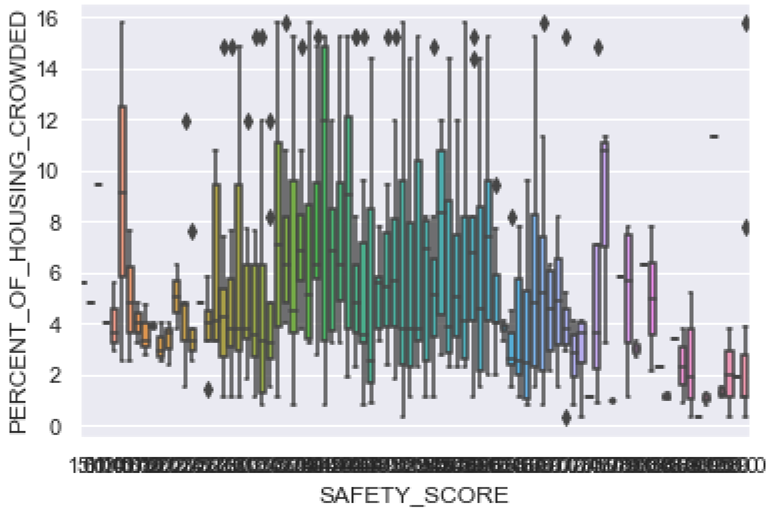
Based on the boxplot below, there seems to be no relationship between them. At the lower end of the safety score, it has a higher percentage of housing crowded and at higher end of safety score, it has a lower percentage of housing crowded. But in the middle portion of the safety score, high percent of housing crowded is observed.

In [145…

```
sns.boxplot(x='SAFETY_SCORE', y='PERCENT_OF_HOUSING_CROWDED', data=merge)
```

Out[145…

```
<AxesSubplot:xlabel='SAFETY_SCORE', ylabel='PERCENT_OF_HOUSING_CROWDED'>
```



4.Is there a difference in percent aged 25+ without high school diploma between the status of healthy school certified?

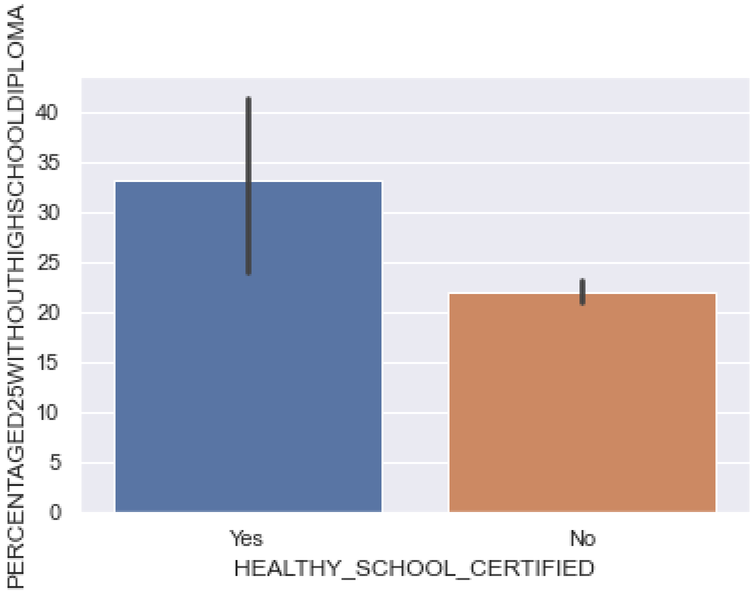
It seems that percent aged 25+ without high school diploma tend to be higher in healthy school certified community area.

In [146…

```
sns.barplot(x='HEALTHY_SCHOOL_CERTIFIED', y='PERCENTAGED25WITHOUTHIGHSCHOOLDIPLOMA', data=
```

Out[146…

```
<AxesSubplot:xlabel='HEALTHY_SCHOOL_CERTIFIED', ylabel='PERCENTAGED25WITHOUTHIGHSCHOOLDIPL
OMA'>
```



5. What is the mean of Parent Engagement Score and PERCENT AGED 16+ UNEMPLOYED?

The mean of parent engagement score and percent aged 16+ unemployed is 50.23 and 16.36 respectively.

In [147…

```
merge['ParentEngagementScore'].mean()
```

Out[147…

```
50.23148148148153
```

In [148…

```
merge['PERCENTAGED16UNEMPLOYED'].mean()
```

Out[148…

```
16.36113074204949
```

6. What is the mean of teachers score and percent of housing crowded?

The mean of teacher score and percent of housing crowded is 49.05 and 5.63 respectively. There seems to be no correlation between percetage of housing crowded and teachers score.

In [149…

```
merge['TeachersScore'].mean()
```

Out[149…

```
49.05084745762717
```

In [150…

```
merge['PERCENT_OF_HOUSING_CROWDED'].mean()
```

Out[150…

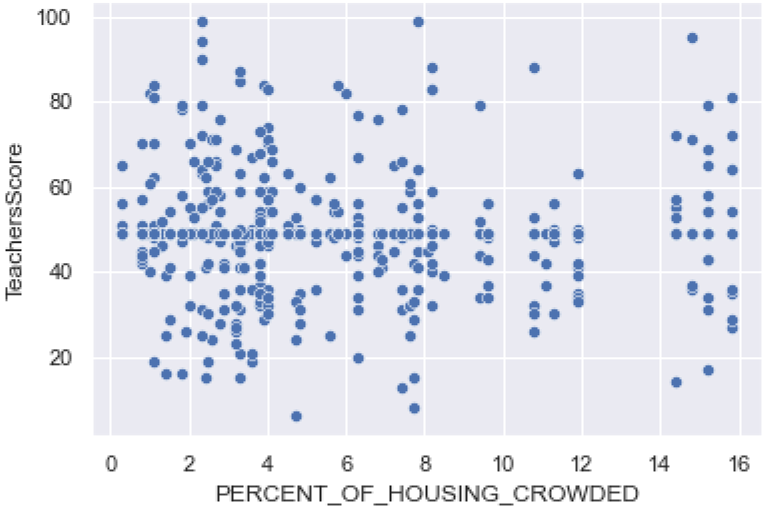
```
5.625971731448778
```

In [151…

```
sns.scatterplot(x='PERCENT_OF_HOUSING_CROWDED', y='TeachersScore', data=merge)
```

Out[151…

```
<AxesSubplot:xlabel='PERCENT_OF_HOUSING_CROWDED', ylabel='TeachersScore'>
```



7. What is the difference in mean or median of per capita income by whether the school is healthy certified?

The mean of per capita income in healthy school certified is 25361.69 and non-certified is 24098.21. The median of the per capita income in healthy school certified is 16444.0 and non-certified is 18881.0.

In [152…

```
merge.groupby('HEALTHY_SCHOOL_CERTIFIED')['PER_CAPITA_INCOME'].describe()
```

Out[152…

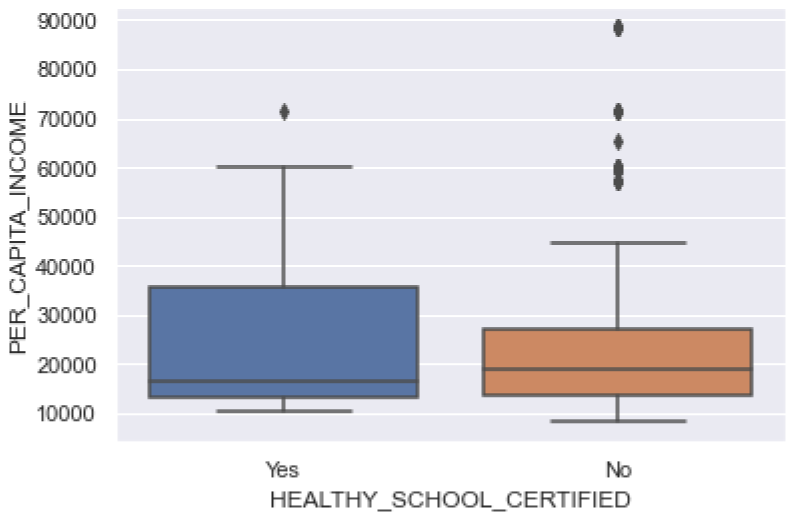
	count	mean	std	min	25%	50%	75%	max
HEALTHY_SCHOOL_CERTIFIED								
No	550.0	24098.209091	14842.064451	8201.0	13781.0	18881.0	27249.00	88669.0
Yes	16.0	25361.687500	19152.552337	10402.0	13089.0	16444.0	35455.75	71551.0

In [153…

```
sns.boxplot(x='HEALTHY_SCHOOL_CERTIFIED', y='PER_CAPITA_INCOME', data=merge)
```

Out[153…

```
<AxesSubplot:xlabel='HEALTHY_SCHOOL_CERTIFIED', ylabel='PER_CAPITA_INCOME'>
```



8. What is the distribution pattern of per capita income and college enrollment? Overlapped on the same scale. Is there a correlation between them?

Because these two variables were scaled differently, we used MinMaxScaler to rescale them between 0 and 1 to overlap them on the same plot. Since they are not normally distributed, we log transformed them and did a scatterplot. It seems that there is no obvious correlation between them (slope is almost zero).

In [154…

```
merge1=merge.loc[:, ['PER_CAPITA_INCOME', 'COLLEGE_ENROLLMENT']]
```

Create a new normalized dataframe

In [155…

```
from sklearn.preprocessing import MinMaxScaler
```

In [156…

```
# create a scaler object
scaler = MinMaxScaler()
# fit and transform the data
merge1_norm = pd.DataFrame(scaler.fit_transform(merge1), columns=merge1.columns)
merge1_norm
```

Out[156…

	PER_CAPITA_INCOME	COLLEGE_ENROLLMENT
0	0.787269	0.182195
1	0.139148	0.115022
2	0.189902	0.299747
3	0.056718	0.123073
4	0.098760	0.064642
...	...	...
561	0.056718	0.101449
562	0.069394	0.080515

	PER_CAPITA_INCOME	COLLEGE_ENROLLMENT
563	0.294614	0.167242
564	0.130126	0.049919
565	0.027352	0.083046

566 rows × 2 columns

In [157…

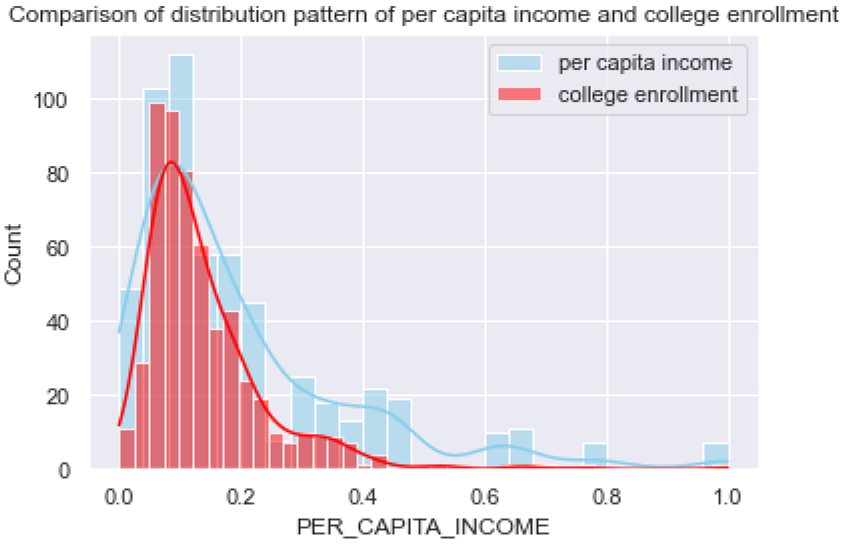
```
sns.set(style="darkgrid")

sns.histplot(data=merge1_norm, x="PER_CAPITA_INCOME", color="skyblue", label="per capita i
sns.histplot(data=merge1_norm, x="COLLEGE_ENROLLMENT", color="red", label="college enrollm

plt.legend()
plt.title("Comparison of distribution pattern of per capita income and college enrollment'
```

Out[157…

```
Text(0.5, 1.0, 'Comparison of distribution pattern of per capita income and college enroll
ment')
```



In [158…

```
merge1['logpercapincome']=np.log(merge1['PER_CAPITA_INCOME'])
merge1['logcollegeenrollment']=np.log(merge1['COLLEGE_ENROLLMENT'])
```

In [159…

```
merge1
```

Out[159…

	PER_CAPITA_INCOME	COLLEGE_ENROLLMENT	logpercapincome	logcollegeenrollment
0	71551.0	813.0	11.178166	6.700731
1	19398.0	521.0	9.872925	6.255750
2	23482.0	1324.0	10.063989	7.188413
3	12765.0	556.0	9.454462	6.320768
4	16148.0	302.0	9.689551	5.710427
...	...	...	...	...
561	12765.0	462.0	9.454462	6.135565
562	13785.0	371.0	9.531336	5.916202
563	31908.0	748.0	10.370612	6.617403

	PER_CAPITA_INCOME	COLLEGE_ENROLLMENT	logpercapincome	logcollegeenrollment
564	18672.0	238.0	9.834780	5.472271
565	10402.0	382.0	9.249753	5.945421

566 rows × 4 columns

In [160…

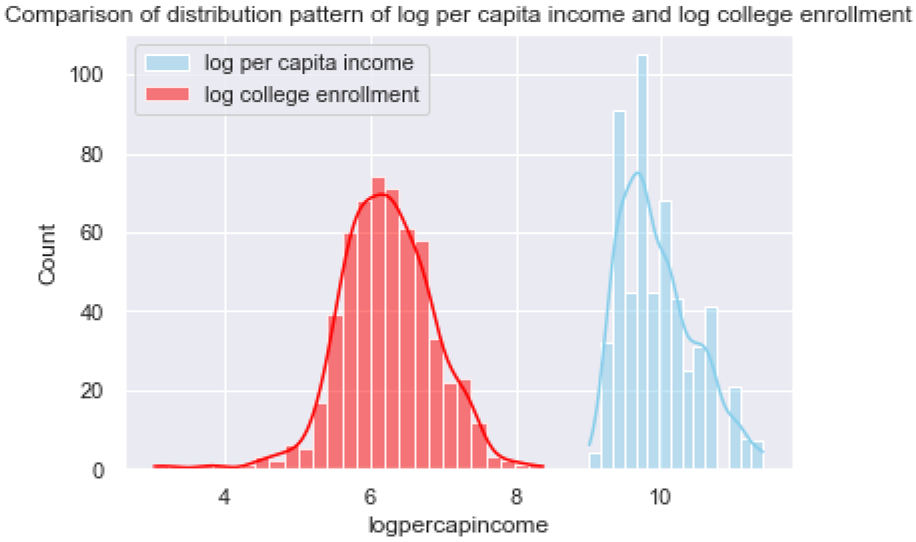
```
sns.set(style="darkgrid")

sns.histplot(data=merge1, x="logpercapincome", color="skyblue", label="log per capita inco
sns.histplot(data=merge1, x="logcollegeenrollment", color="red", label="log college enrol

plt.legend()
plt.title("Comparison of distribution pattern of log per capita income and log college enr
```

Out[160…

```
Text(0.5, 1.0, 'Comparison of distribution pattern of log per capita income and log colleg
e enrollment')
```

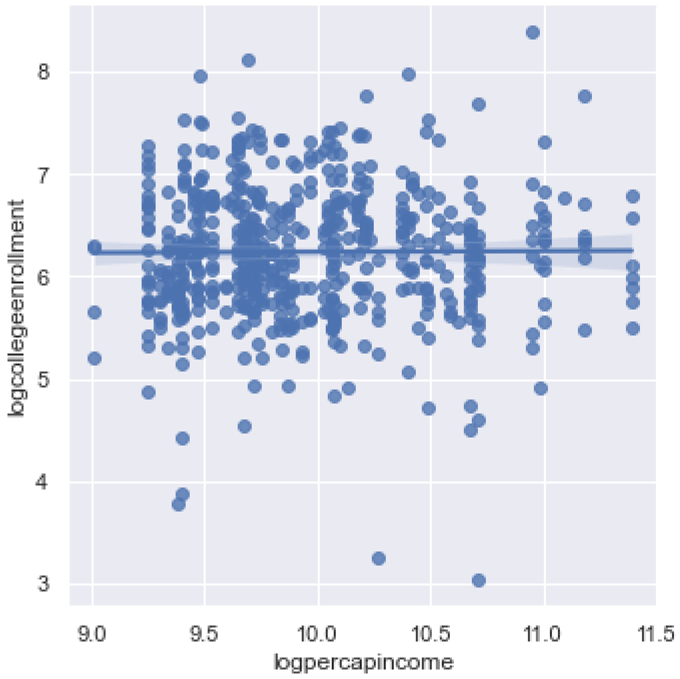


In [161…

```
sns.lmplot(x='logpercapincome', y='logcollegeenrollment', data=merge1)
```

Out[161…

```
<seaborn.axisgrid.FacetGrid at 0x2d533c4d070>
```



9.Is there any difference in the distribution of PERCENT AGED 25+ WITHOUT HIGH SCHOOL DIPLOMA whether the school is healthy school certified or not?

Based on the graph shown below, schools with healthy certification seems to have a higher percentage of people aged 25+ without high school diploma than communities that are non-healthy school certified.

In [162…

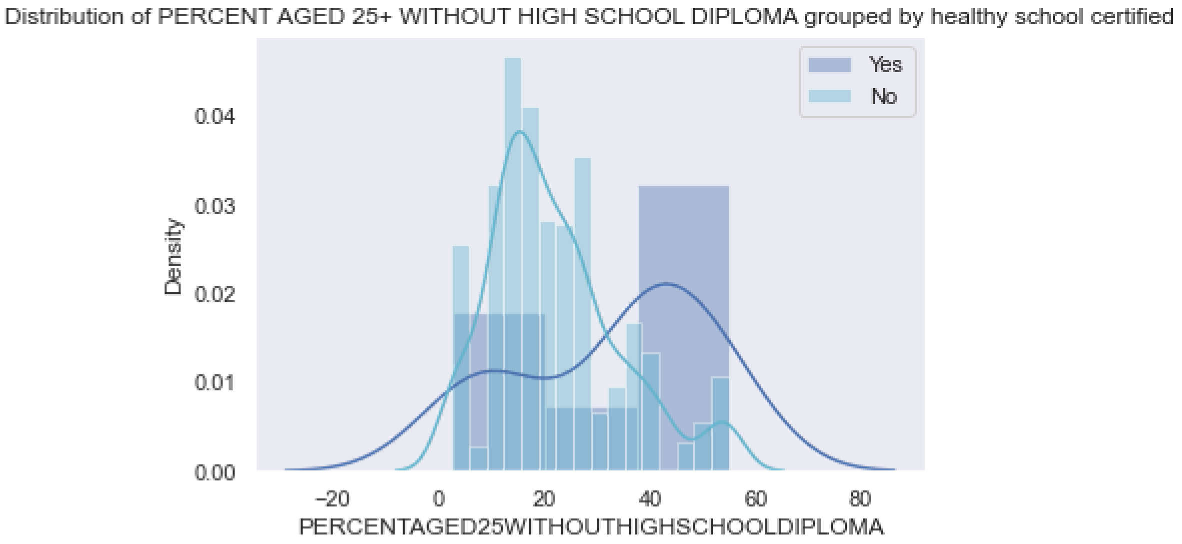
```
plt.figure()
plt.title("Distribution of PERCENT AGED 25+ WITHOUT HIGH SCHOOL DIPLOMA grouped by healthy

# Plot the distribution of conventional type data
mask0 = merge['HEALTHY_SCHOOL_CERTIFIED'] == 'Yes'
ax = sns.distplot(merge["PERCENTAGED25WITHOUTHIGHSCHOOLDIPLOMA"][mask0], color = 'b', labe

# Plot the histogram of organic type data
mask1 = merge['HEALTHY_SCHOOL_CERTIFIED'] == 'No'
ax = sns.distplot(merge["PERCENTAGED25WITHOUTHIGHSCHOOLDIPLOMA"][mask1], color = 'c', labe

# add legend, show the graphics
plt.legend()
plt.grid()
```

```
C:\Users\liden\anaconda3\envs\UNCG\lib\site-packages\seaborn\distributions.py:2619: Future
Warning: `distplot` is a deprecated function and will be removed in a future version. Plea
se adapt your code to use either `displot` (a figure-level function with similar flexibili
ty) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
C:\Users\liden\anaconda3\envs\UNCG\lib\site-packages\seaborn\distributions.py:2619: Future
Warning: `distplot` is a deprecated function and will be removed in a future version. Plea
se adapt your code to use either `displot` (a figure-level function with similar flexibili
ty) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```

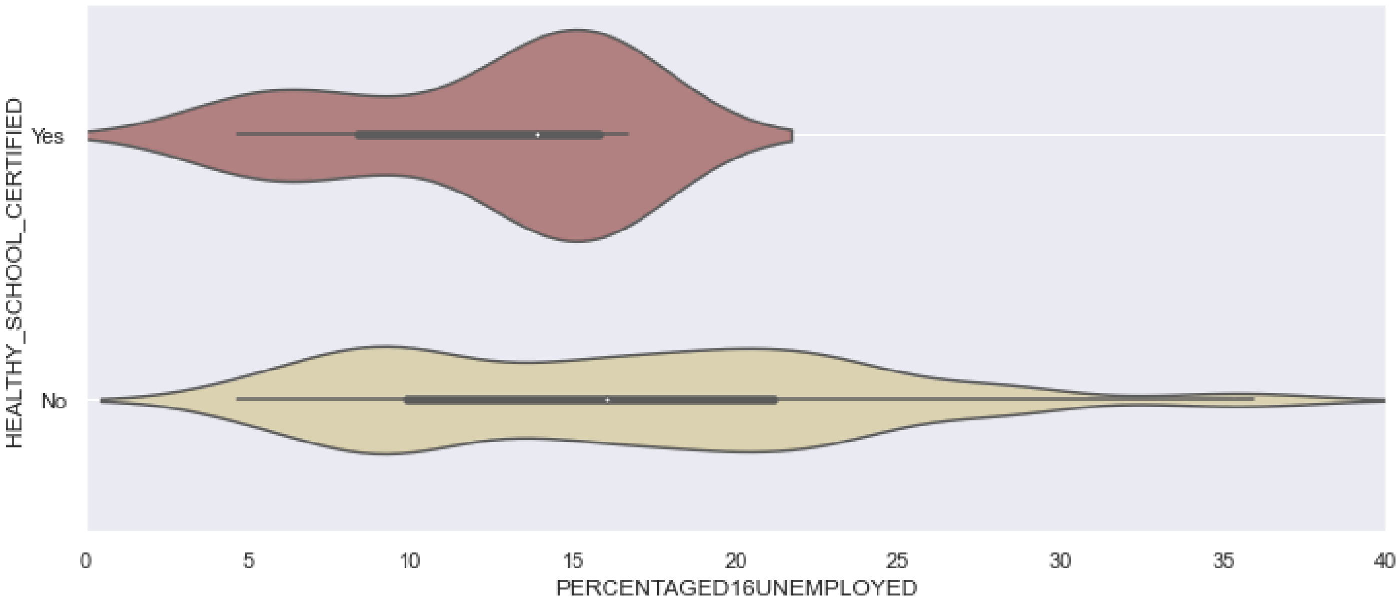


10. Is there a difference in percent aged 16+ unemployed between healthy school certified or not?

The percentage of 16+ unemployed is higher in non-healthy school certified communites than that of healthy school certified communities.

In [163…

```
plt.figure(figsize=(12,5))
sns.violinplot(y = "HEALTHY_SCHOOL_CERTIFIED", x = "PERCENTAGED16UNEMPLOYED", data = merge
plt.xlim([0, 40])
plt.grid()
plt.show()
```


HEALTHY_SCHOOL_CERTIFIED
Yes
No
0
5
10
15
20
25
30
35
40
PERCENTAGED16UNEMPLOYED

In [ ]: