

# **American Universities' Data Analysis Using Tableau and R**

**UCS548**

**Data Science Foundation**

Submitted To Ed. Kashish Goyal

Submitted By Sehajparkash Singh Pannu

102003173

3COE-7



Computer Science and Engineering, Department July-Dec 2022

# RAW DATASETS:

## Unclean datasets:

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
A1 Undo Clipboard Font Alignment Number Styles																
1	ID number	Applicants	Admission	Enrolled	to	Percent	of	Percent	of	Number	of	Number	of	Number	of	Percent
2	100654	6142	5521	1104	15	88	0	604	213	8	9	0	1	92	55	
3	100663	5689	4934	1773	6	93	0	2159	1533	542	70	0	5	21	61	86
4	100690						5	268	60	10	0	0	0	40	58	
5	100706	2054	1656	651	34	94	0	1091	444	38	28	1	4	12	44	79
6	100724	10245	5251	1479	18	87	0	599	147	48	27	0	0	91	61	58
7	100751	30975	17515	6454	23	76	0	4963	1649	493	38	0	1	11	55	40
8	100812						0	907	0	0	0	2	1	11	65	
9	100830	1958	1639	579	0	54	0	551	239	0	14	0	2	30	63	95
10	100858	15745	13027	3726	17	83	0	4192	1068	483	58	1	2	7	49	63
11	100937	1931	1240	356	32	82	0	299	0	0	0	1	4	9	46	
12	101073						39	42	0	0	0	0	0	95	42	48
13	101189						10	561	97	101	0	1	1	46	62	67
14	101435	1470	923	261	16	93	0	217	0	0	0	1	1	19	50	27
15	101480	3083	2567	1158	18	89	0	1365	339	0	71	1	1	25	58	76
16	101541	268	198	68	10	94	20	38	0	0	0	0	1	14	95	88
17	101587	462	460	380	0	100	53	242	660	0	205	1	1	49	72	
18	101675						0	184	0	0	0	0	0	97	48	
19	101693	866	617	259	13	87	13	286	57	0	0	2	1	24	68	
20	101709	1385	1209	531	3	96	0	325	165	0	33	1	1	15	67	94
21	101879	2542	2057	970	3	96	0	952	334	0	4	1	1	13	58	85
22	101912	2728	937	409	47	58	2	350	14	0	18	0	0	84	58	
23	102049	3447	2653	765	36	84	3	659	338	335	19	0	2	7	62	
24	102094	4814	4142	1878	8	92	0	1789	713	168	51	1	3	20	61	79
25	102234	6596	3052	431	20	92	0	212	45	0	14	1	1	16	60	
26	102270	4121	1772	257	6	59	0	137	0	0	0	0	0	92	49	
27	102298						0	78	0	0	0	0	0	89	53	
28	102368	5946	4056	2087	7	38	494	2821	1507	9	23	1	1	38	64	72

A	B	C	D	E	F	G	H	I	J	K	L
A1 Undo Clipboard Font Alignment Number Styles											
1	ID number	Tuition and fees	Admission	Tuition and fees	Admission	Total price	Total price	Percent of freshmen	receiving any financial aid		
2	100654	5800	6828	7182	7182	21849	27441	97			
3	100663	5806	6264	6798	7206	22495	31687	90			
4	100690	8360	8720	6800	6870			100			
5	100706	7492	8094	8794	9192	23466	35780	87			
6	100724	7164	8082	7932	8720	18286	25222	93			
7	100751	7900	8600	9200	9450	27000	41500	75			
8	100812										
9	100830	6620	7580	8150	8750	17820	34020	90			
10	100858	7900	8698	9446	9852	25282	41794	76			
11	100937	27890	29290	30690	30690	45470	45470	98			
12	101073	7370	8090	8090	8090	18330	18330	100			
13	101189	14810	15880	17380	18230	31010	31010	99			
14	101435	20990	21990	22500	23500	33885	33885	100			
15	101480	5424	6120	6360	7092	21237	28029	93			
16	101541	13900	14460	15100	15630	28840	28840	100			
17	101587	6524	6918	7320	7660	18162	24672	95			
18	101675	9896	10490	11014	11454	22318	22318	99			
19	101693	16120	17220	18080	18690	32570	32570	93			
20	101709	7750	8520	9280	10000	23340	33030	88			
21	101879	5426	6084	6528	7078	16933	22525	82			
22	101912	14250	14678	15414	16234	36295	36295	94			
23	102049	21942	23963	25150	26328	40900	40900	98			
24	102094	5748	6204	6660	6948	19980	26628	91			
25	102234	26730	28060	29450	30924	46170	46170	100			
26	102270	14464	15062	15665	15901	26877	26877	100			
27	102298	11111	11143	11492	11492	21296	21296	98			
28	102368	5860	6412	6844	7276	18228	24564	78			

# CLEANING AND REFINING DATASETS

## Cleaning and refining the first dataset:

```
5 ds1=read.csv("D:/ds/Tableau/Book3a_es.csv") #reading the first dataset
6 dim(ds1)
7 colnames(ds1)
7:14 (Top Level) R Script

Console Terminal Background Jobs
R 4.2.1 · ~/ 
> ds1=read.csv("D:/ds/Tableau/Book3a_es.csv") #reading the first dataset
> dim(ds1)
[1] 1534 24
> colnames(ds1)
[1] "ID.number"                      "Name"
[3] "year"                            "Highest.degree.offered"
[5] "County.name"                     "Religious.affiliation"
[7] "Longitude.location.of.institution" "Latitude.location.of.institution"
[9] "Offers.Associate.s.degree"        "Offers.Bachelor.s.degree"
[11] "Offers.Postbaccalaureate.certificate" "Offers.Master.s.degree"
[13] "Offers.Post.master.s.certificate" "Offers.Doctor.s.degree...research.scholarship"
[15] "Offers.Doctor.s.degree...professional.practice" "Offers.Doctor.s.degree...other"
[17] "Total..enrollment"                "Full.time.enrollment"
[19] "Part.time.enrollment"             "Undergraduate.enrollment"
[21] "Graduate.enrollment"              "Level.of.institution"
[23] "Control.of.institution"          "Degree.of.urbanization..Urban.centric.locale."
>

9 clean<-clean_names(ds1)           # replaces '.' in column names with an underscore '_'
10 colnames(clean)
11 (Top Level) R Script

Console Terminal Background Jobs
R 4.2.1 · ~/ 
> clean<-clean_names(ds1)           # replaces '.' in column names with an underscore '_'
> colnames(clean)
[1] "id_number"                      "name"
[3] "year"                            "highest_degree_offered"
[5] "county_name"                     "religious_affiliation"
[7] "longitude_location_of_institution" "latitude_location_of_institution"
[9] "offers_associate_s_degree"        "offers_bachelor_s_degree"
[11] "offers_postbaccalaureate_certificate" "offers_master_s_degree"
[13] "offers_post_master_s_certificate" "offers_doctor_s_degree_research_scholarship"
[15] "offers_doctor_s_degree_professional_practice" "offers_doctor_s_degree_other"
[17] "total_enrollment"                "full_time_enrollment"
[19] "part_time_enrollment"             "undergraduate_enrollment"
[21] "graduate_enrollment"              "level_of_institution"
[23] "control_of_institution"          "degree_of_urbanization_urban_centric_locale"
>
```

```
12 clean_x<-clean %>% remove_empty(which=c("rows")) #deletes the rows which contain NA values in all of their cells
13 dim(clean_x)
```

13:13 (Top Level) ▾

R Script ▾

Console Terminal × Background Jobs ×

R 4.2.1 · ~/

```
> clean_x<-clean %>% remove_empty(which=c("rows")) #deletes the rows which contain NA values in all of their cells
> dim(clean_x)
[1] 1534 24
```

```
15 row.has.na <- apply(clean_x, 1, function(x){any(is.na(x))}) #checks each row for atleast one NA value
16 length(row.has.na[row.has.na==TRUE]) #no. of rows containing at least one NA value
```

17 16:86 (Top Level) ▾

R Script ▾

Console Terminal × Background Jobs ×

R 4.2.1 · ~/

```
> row.has.na <- apply(clean_x, 1, function(x){any(is.na(x))}) #checks each row for atleast one NA value
> length(row.has.na[row.has.na==TRUE]) #no. of rows containing at least one NA value
[1] 2
```

```
18 #Since there are only 2 rows containing NA values, therefore we delete these 2 rows
19 final.dataset=na.omit(clean_x)
20 dim(final.dataset)
```

21 21:1 (Top Level) ▾

R Script ▾

Console Terminal × Background Jobs ×

R 4.2.1 · ~/

```
> #Since there are only 2 rows containing NA values, therefore we delete these 2 rows
> final.dataset=na.omit(clean_x)
> dim(final.dataset)
[1] 1532 24
```

```
22 #Finally save the dataset into a new file
23 write.csv(final.dataset,file = "D:/ds/Tableau/Book3a_na.csv")
```

24 23:62 (Top Level) ▾

R Script ▾

Console Terminal × Background Jobs ×

R 4.2.1 · ~/

```
> #Finally save the dataset into a new file
> write.csv(final.dataset,file = "D:/ds/Tableau/Book3a_na.csv")
` |
```

## Cleaning and refining the second dataset:

```
25 ##### 2nd Dataset
26 ds2=read.csv("D:/ds/Tableau/Book3b_es.csv") #reading the second dataset
27 dim(ds2)
28 colnames(ds2)
29
```

28:14 (Top Level) R Script

Console Terminal Background Jobs

```
R 4.2.1 · ~/ 
> ds2=read.csv("D:/ds/Tableau/Book3b_es.csv") #reading the second dataset
> dim(ds2)
[1] 1534 18
> colnames(ds2)
[1] "ID.number"
[2] "Applicants.total.2013"
[3] "Admissions.total.2013"
[4] "Enrolled.total.2013"
[5] "Percent.of.freshmen.submitting.SAT.scores.2013."
[6] "Percent.of.freshmen.submitting.ACT.scores.2013."
[7] "Number.of.students.receiving.an.Associate.s.degree"
[8] "Number.of.students.receiving.a.Bachelor.s.degree"
[9] "Number.of.students.receiving.a.Master.s.degree"
[10] "Number.of.students.receiving.a.Doctor.s.degree"
[11] "Number.of.students.receiving.a.Postbaccalaureate.or.Post.master.s.certificate"
[12] "Percent.of.total.enrollment.that.are.American.Indian.or.Alaska.Native"
[13] "Percent.of.total.enrollment.that.are.Asian"
[14] "Percent.of.total.enrollment.that.are.Black.or.African.American"
[15] "Percent.of.total.enrollment.that.are.women"
[16] "Percent.of.first.time.undergraduates...in.state"
[17] "Percent.of.first.time.undergraduates...out.of.state"
[18] "Percent.of.first.time.undergraduates...foreign.countries"
```

```
30 clean<-clean_names(ds2)          # replaces '.' in column names with an underscore '_'
31 colnames(clean)
32
33 clean_x<-clean %>% remove_empty(which=c("rows")) #deletes the rows which contain NA values in all of their cells
34 dim(clean_x)
35 <
36
```

(Top Level) ▾ R Script ▾

```
R 4.2.1 · ~/🔗
> clean<-clean_names(ds2)          # replaces '.' in column names with an underscore '_'
> colnames(clean)
[1] "id_number"
[2] "applicants_total_2013"
[3] "admissions_total_2013"
[4] "enrolled_total_2013"
[5] "percent_of_freshmen_submitting_sat_scores_2013"
[6] "percent_of_freshmen_submitting_act_scores_2013"
[7] "number_of_students_receiving_an_associate_s_degree"
[8] "number_of_students_receiving_a_bachelor_s_degree"
[9] "number_of_students_receiving_a_master_s_degree"
[10] "number_of_students_receiving_a_doctor_s_degree"
[11] "number_of_students_receiving_a_postbaccalaureate_or_post_master_s_certificate"
[12] "percent_of_total_enrollment_that_are_american_indian_or_alaska_native"
[13] "percent_of_total_enrollment_that_are_asian"
[14] "percent_of_total_enrollment_that_are_black_or_african_american"
[15] "percent_of_total_enrollment_that_are_women"
[16] "percent_of_first_time_undergraduates_in_state"
[17] "percent_of_first_time_undergraduates_out_of_state"
[18] "percent_of_first_time_undergraduates_foreign_countries"
>
> clean_x<-clean %>% remove_empty(which=c("rows")) #deletes the rows which contain NA values in all of their cells
> dim(clean_x)
[1] 1534   18
```

```
36 row.has.na <- apply(clean_x, 1, function(x){any(is.na(x))}) #checks each row for atleast one NA value
37 length(row.has.na[row.has.na==TRUE])    #no. of rows containing at least one NA value
38 <
```

(Top Level) ▾ R Script ▾

```
Console Terminal ▾ Background Jobs ▾
R 4.2.1 · ~/🔗
> row.has.na <- apply(clean_x, 1, function(x){any(is.na(x))}) #checks each row for atleast one NA value
> length(row.has.na[row.has.na==TRUE])    #no. of rows containing at least one NA value
[1] 731
```

```
39 #Since there are 731 rows containing at least 1 NA value, we can't delete all of them
40 row.has.no.na <- apply(clean_x, 1, function(x){sum(is.na(x))}) #gives the no. of NA values in each row
41 un.no.of.na<-unique(row.has.no.na) #gives the unique values for count of NA values of every row
42 un.no.of.na
43
```

42:12 (Top Level) ▾

R Script ▾

Console Terminal X Background Jobs X

R 4.2.1 · ~/

```
> row.has.na <- apply(clean_x, 1, function(x){any(is.na(x))}) #checks each row for atleast one NA value
> length(row.has.na[row.has.na==TRUE]) #no. of rows containing at least one NA value
[1] 731
> #Since there are 731 rows containing at least 1 NA value, we can't delete all of them
> row.has.no.na <- apply(clean_x, 1, function(x){sum(is.na(x))}) #gives the no. of NA values in each row
> un.no.of.na<-unique(row.has.no.na) #gives the unique values for count of NA values of every row
> un.no.of.na
[1] 3 0 8 5 2 17 1 4
```

```
44 #So,we keep the rows with NA count<5 and delete the rest of the rows
45 clean_y=clean_x[row.has.no.na<5,]
46 dim(clean_y) #new dimensions
47
```

46:42 (Top Level) ▾

R Script ▾

Console Terminal X Background Jobs X

R 4.2.1 · ~/

```
> #So,we keep the rows with NA count<5 and delete the rest of the rows
> clean_y=clean_x[row.has.no.na<5,]
> dim(clean_y) #new dimensions
[1] 1314 18
```

```
48 #Now, we will fill the cells with NA values with the mean of their respective columns
49
50 col.has.na<-apply(clean_y, 2, anyNA) #checks each column for atleast one NA value
51 list_na <- colnames(clean_y)[col.has.na] #gives the names of columns with NA values
52 list_na
53
```

52:8 (Top Level) ▾

R Script ▾

Console Terminal X Background Jobs X

R 4.2.1 · ~/

```
> #Now, we will fill the cells with NA values with the mean of their respective columns
>
> col.has.na<-apply(clean_y, 2, anyNA) #checks each column for atleast one NA value
> list_na <- colnames(clean_y)[col.has.na] #gives the names of columns with NA values
> list_na
[1] "percent_of_freshmen_submitting_sat_scores_2013"
[2] "percent_of_freshmen_submitting_act_scores_2013"
[3] "percent_of_first_time_undergraduates_in_state"
[4] "percent_of_first_time_undergraduates_out_of_state"
[5] "percent_of_first_time_undergraduates_foreign_countries"
```

```
54 # Creates mean for each of the columns with NA values
55 mean_missing <- apply(clean_y[, colnames(clean_y) %in% list_na], 2, mean, na.rm = TRUE)
56 mean_missing
57
```

56:13 (Top Level) ▾

R Script ▾

Console Terminal × Background Jobs ×

R 4.2.1 · ~/

```
> # Creates mean for each of the columns with NA values
> mean_missing <- apply(clean_y[, colnames(clean_y) %in% list_na], 2, mean, na.rm = TRUE)
> mean_missing
  percent_of_freshmen_submitting_sat_scores_2013      percent_of_freshmen_submitting_act_scores_2013
  53.927605                                              53.085782
  percent_of_first_time_undergraduates_in_state          percent_of_first_time_undergraduates_out_of_state
  68.463329                                              27.820722
percent_of_first_time_undergraduates_foreign_countries
  3.074505
```

```
58 # Fill the NA values with the mean of the columns
59 clean_z <- clean_y %>%
60   mutate(percent_of_freshmen_submitting_sat_scores_2013 = ifelse(is.na(percent_of_freshmen_submitting_sat_scores_2013),
61     mean_missing[1], percent_of_freshmen_submitting_sat_scores_2013),
62     percent_of_freshmen_submitting_act_scores_2013 = ifelse(is.na(percent_of_freshmen_submitting_act_scores_2013),
63       mean_missing[2], percent_of_freshmen_submitting_act_scores_2013),
64     percent_of_first_time_undergraduates_in_state = ifelse(is.na(percent_of_first_time_undergraduates_in_state),
65       mean_missing[3], percent_of_first_time_undergraduates_in_state),
66     percent_of_first_time_undergraduates_out_of_state= ifelse(is.na(percent_of_first_time_undergraduates_out_of_state ),
67       mean_missing[4], percent_of_first_time_undergraduates_out_of_state),
68     percent_of_first_time_undergraduates_foreign_countries = ifelse(is.na(percent_of_first_time_undergraduates_foreign_countries),
69       mean_missing[5],percent_of_first_time_undergraduates_foreign_countries)
70   )
71 dim(clean_z)
```

71:13 (Top Level) ▾

R Script ▾

Console Terminal × Background Jobs ×

R 4.2.1 · ~/

```
> # Fill the NA values with the mean of the columns
> clean_z <- clean_y %>%
+   mutate(percent_of_freshmen_submitting_sat_scores_2013 = ifelse(is.na(percent_of_freshmen_submitting_sat_scores_2013),
+     mean_missing[1], percent_of_freshmen_submitting_sat_scores_2013),
+     percent_of_freshmen_submitting_act_scores_2013 = ifelse(is.na(percent_of_freshmen_submitting_act_scores_2013),
+       mean_missing[2], percent_of_freshmen_submitting_act_scores_2013),
+     percent_of_first_time_undergraduates_in_state = ifelse(is.na(percent_of_first_time_undergraduates_in_state),
+       mean_missing[3], percent_of_first_time_undergraduates_in_state),
+     percent_of_first_time_undergraduates_out_of_state= ifelse(is.na(percent_of_first_time_undergraduates_out_of_state ),
+       mean_missing[4], percent_of_first_time_undergraduates_out_of_state),
+     percent_of_first_time_undergraduates_foreign_countries = ifelse(is.na(percent_of_first_time_undergraduates_foreign_countries),
+       mean_missing[5],percent_of_first_time_undergraduates_foreign_countries)
+   )
> dim(clean_z)
[1] 1314  18
```

```
72 #Confirm that there are no NA values now
73 new.row.has.na <- apply(clean_z, 1, function(x){any(is.na(x))}) #checks each row for atleast one NA value
74 length(row.has.na[new.row.has.na==TRUE]) #no. of rows containing at least one NA value
75
```

74:90 (Top Level) ▾

R Script ▾

Console Terminal ✎ Background Jobs ✎

R 4.2.1 · ~/

```
> #Confirm that there are no NA values now
> new.row.has.na <- apply(clean_z, 1, function(x){any(is.na(x))}) #checks each row for atleast one NA value
> length(row.has.na[new.row.has.na==TRUE]) #no. of rows containing at least one NA value
```

[1] 0

```
76 #So,our final dataset 2 is clean_z
77 final.dataset.2=clean_z
78 #Finally save the dataset into a new file
79 write.csv(final.dataset.2,file = "D:/ds/Tableau/Book3b_na.csv")
80
```

79:64 (Top Level) ▾

R Script ▾

Console Terminal ✎ Background Jobs ✎

R 4.2.1 · ~/

```
> #So,our final dataset 2 is clean_z
> final.dataset.2=clean_z
> #Finally save the dataset into a new file
> write.csv(final.dataset.2,file = "D:/ds/Tableau/Book3b_na.csv")
>
```

## Cleaning and refining the third dataset:

```
81 ##### 3rd Dataset
82 ds3=read.csv("D:/ds/Tableau/Book3c_es.csv") #reading the second dataset
83 dim(ds3)
84 colnames(ds3)
85 
86 clean<-clean_names(ds3) # replaces '.' in column names with an underscore '_'
87 colnames(clean)
88 
89 clean_x<-clean %>% remove_empty(which=c("rows")) #deletes the rows which contain NA values in all of their cells
90 dim(clean_x)
91 
```

84:14 (Top Level) R Script

Console Terminal Background Jobs R 4.2.1 · ~/

```
> ##### 3rd Dataset
> ds3=read.csv("D:/ds/Tableau/Book3c_es.csv") #reading the second dataset
> dim(ds3)
[1] 1534     8
> colnames(ds3)
[1] "ID.number"                               "Tuition.and.fees..2010.11"
[3] "Tuition.and.fees..2011.12"                "Tuition.and.fees..2012.13"
[5] "Tuition.and.fees..2013.14"                "Total.price.for.in.state.students.living.on.campus.
2013.14"
[7] "Total.price.for.out.of.state.students.living.on.campus.2013.14" "Percent.of.freshmen.receiving.any.financial.aid"
[9] 
```

90:13 (Top Level) R Script

Console Terminal Background Jobs R 4.2.1 · ~/

```
> clean<-clean_names(ds3) # replaces '.' in column names with an underscore '_'
> colnames(clean)
[1] "id_number"
[2] "tuition_and_fees_2010_11"
[3] "tuition_and_fees_2011_12"
[4] "tuition_and_fees_2012_13"
[5] "tuition_and_fees_2013_14"
[6] "total_price_for_in_state_students_living_on_campus_2013_14"
[7] "total_price_for_out_of_state_students_living_on_campus_2013_14"
[8] "percent_of_freshmen_receiving_any_financial_aid"
>
> clean_x<-clean %>% remove_empty(which=c("rows")) #deletes the rows which contain NA values in all of their cells
> dim(clean_x)
[1] 1534     8
```

```
92 row.has.na <- apply(clean_x, 1, function(x){any(is.na(x))}) #checks each row for atleast one NA value
93 length(row.has.na[row.has.na==TRUE]) #no. of rows containing at least one NA value
94
```

93:86 (Top Level) ▾

R Script

Console Terminal X Background Jobs X

R 4.2.1 · ~/

```
> row.has.na <- apply(clean_x, 1, function(x){any(is.na(x))}) #checks each row for atleast one NA value
> length(row.has.na[row.has.na==TRUE]) #no. of rows containing at least one NA value
```

[1] 137

▀

```
95 #since there are 137 rows containing at least 1 NA value, we can't delete all of them
96 row.has.no.na <- apply(clean_x, 1, function(x){sum(is.na(x))}) #gives the no. of NA values in each row
97 un.no.of.na<-unique(row.has.no.na) #gives the unique values for count of NA values of every row
98 un.no.of.na
99
```

98:12 (Top Level) ▾

R Script

Console Terminal X Background Jobs X

R 4.2.1 · ~/

```
> #Since there are 137 rows containing at least 1 NA value, we can't delete all of them
> row.has.no.na <- apply(clean_x, 1, function(x){sum(is.na(x))}) #gives the no. of NA values in each row
> un.no.of.na<-unique(row.has.no.na) #gives the unique values for count of NA values of every row
> un.no.of.na
[1] 0 2 7 5 3 6
```

```
100 #So,we keep the rows with NA count<3 and delete the rest of the rows
101 clean_y=clean_x[row.has.no.na<3,]
102 dim(clean_y) #new dimensions
103
```

102:42 (Top Level) ▾

R Script

Console Terminal X Background Jobs X

R 4.2.1 · ~/

```
> #So,we keep the rows with NA count<3 and delete the rest of the rows
> clean_y=clean_x[row.has.no.na<3,]
> dim(clean_y) #new dimensions
[1] 1489 8
```

```
104 #Now, we will fill the cells with NA values with the mean of their respective columns
105
106 col.has.na<-apply(clean_y, 2, anyNA)      #checks each column for atleast one NA value
107 list_na <- colnames(clean_y)[col.has.na]    #gives the names of columns with NA values
108 list_na
109
```

108:8 (Top Level) ▾

R Script ▾

Console Terminal X Background Jobs X

R 4.2.1 · ~/

```
> #Now, we will fill the cells with NA values with the mean of their respective columns
>
> col.has.na<-apply(clean_y, 2, anyNA)      #checks each column for atleast one NA value
> list_na <- colnames(clean_y)[col.has.na]    #gives the names of columns with NA values
> list_na
[1] "tuition_and_fees_2010_11"
[2] "tuition_and_fees_2011_12"
[3] "total_price_for_in_state_students_living_on_campus_2013_14"
[4] "total_price_for_out_of_state_students_living_on_campus_2013_14"
```

```
110 # Creates mean for each of the columns with NA values
111 mean_missing <- apply(clean_y[,colnames(clean_y) %in% list_na],2,mean,na.rm = TRUE)
112 mean_missing
113
```

110:1 (Top Level) ▾

R Script ▾

Console Terminal X Background Jobs X

R 4.2.1 · ~/

```
> # Creates mean for each of the columns with NA values
> mean_missing <- apply(clean_y[,colnames(clean_y) %in% list_na],2,mean,na.rm = TRUE)
> mean_missing
          tuition_and_fees_2010_11
          18373.91
          tuition_and_fees_2011_12
          19262.43
          total_price_for_in_state_students_living_on_campus_2013_14
          34946.90
          total_price_for_out_of_state_students_living_on_campus_2013_14
          38784.69
```

```

114 # Fill the NA values with the mean of the columns
115 clean_z <- clean_y %>%
116   mutate(tuition_and_fees_2010_11 = ifelse(is.na(tuition_and_fees_2010_11),mean_missing[1], tuition_and_fees_2010_11),
117         tuition_and_fees_2011_12 = ifelse(is.na(tuition_and_fees_2011_12),mean_missing[2],tuition_and_fees_2011_12),
118         total_price_for_in_state_students_living_on_campus_2013_14=ifelse(is.na(total_price_for_in_state_students_living_on_campus_2013_14),
119                         mean_missing[3], total_price_for_in_state_students_living_on_campus_2013_14),
120         total_price_for_out_of_state_students_living_on_campus_2013_14=ifelse(is.na(total_price_for_out_of_state_students_living_on_campus_2013_14),
121                         mean_missing[4],total_price_for_out_of_state_students_living_on_campus_2013_14 )
122       )
123 dim(clean_z)
124

```

123:13 (Top Level) ▾

R Script ▾

Console Terminal Background Jobs

R 4.2.1 · ~/

```

> # Fill the NA values with the mean of the columns
> clean_z <- clean_y %>%
+   mutate(tuition_and_fees_2010_11 = ifelse(is.na(tuition_and_fees_2010_11),mean_missing[1], tuition_and_fees_2010_11),
+         tuition_and_fees_2011_12 = ifelse(is.na(tuition_and_fees_2011_12),mean_missing[2],tuition_and_fees_2011_12),
+         total_price_for_in_state_students_living_on_campus_2013_14=ifelse(is.na(total_price_for_in_state_students_living_on_campus_2013_14),
+                         mean_missing[3], total_price_for_in_state_students_living_on_campus_2013_14),
+         total_price_for_out_of_state_students_living_on_campus_2013_14 = ifelse(is.na(total_price_for_out_of_state_students_living_on_campus_2013_14),
+                         mean_missing[4],total_price_for_out_of_state_students_living_on_campus_2013_14 )
+       )
> dim(clean_z)
[1] 1489  8

```

```

125 #Confirm that there are no NA values now
126 new.row.has.na <- apply(clean_z, 1, function(x){any(is.na(x))}) #checks each row for atleast one NA value
127 length(new.row.has.na[new.row.has.na==TRUE]) #no. of rows containing at least one NA value
128

```

128:90 (Top Level) ▾

R Script ▾

Console Terminal Background Jobs

R 4.2.1 · ~/

```

> #Confirm that there are no NA values now
> new.row.has.na <- apply(clean_z, 1, function(x){any(is.na(x))}) #checks each row for atleast one NA value
> length(new.row.has.na[new.row.has.na==TRUE]) #no. of rows containing at least one NA value
[1] 0

```

```

129 #So,our final dataset 2 is clean_z
130 final.dataset.3=clean_z
131 #Finally save the dataset into a new file
132 write.csv(final.dataset.3,file = "D:/ds/Tableau/Book3c_na.csv")
133

```

133:64 (Top Level) ▾

R Script ▾

Console Terminal Background Jobs

R 4.2.1 · ~/

```

> #So,our final dataset 2 is clean_z
> final.dataset.3=clean_z
> #Finally save the dataset into a new file
> write.csv(final.dataset.3,file = "D:/ds/Tableau/Book3c_na.csv")

```

# PROCESSED DATASETS:

## Cleaned and refined datasets:

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
1	id_number	name	year	highest_de	county_na	religious_e	longitude	latitude	lc offers	ass offers	bac offers	pos offers	ma offers	pos offers	do offers	docs offers	total_enrc	full_time_p
2	100654	Alabama A	2013	Doctor's d	Madison C	Not applic	-86.5685	34.78337	Implied no	Yes	Implied no	Yes	Implied no	Yes	Implied no	Implied no	5020	4439
3	100663	University	2013	Doctor's d	Jefferson (	Not applic	-86.8092	33.50223	Implied no	Yes	18568	11961						
4	100690	Amridge U	2013	Doctor's d	Montgom	Churches c	-86.174	32.36261	Yes	Yes	Implied no	Yes	Implied no	Yes	Yes	Implied no	631	323
5	100706	University	2013	Doctor's d	Madison C	Not applic	-86.6384	34.72282	Implied no	Yes	Yes	Yes	Yes	Yes	Yes	Implied no	7376	4802
6	100724	Alabama S	2013	Doctor's d	Montgom	Not applic	-86.2957	32.36432	Implied no	Yes	Implied no	Yes	Yes	Yes	Yes	Implied no	6075	5182
7	100751	The Univei	2013	Doctor's d	Tuscaloosa	Not applic	-87.5458	33.2144	Implied no	Yes	Implied no	Yes	Yes	Yes	Yes	Implied no	34752	29498
8	100812	Athens Sta	2013	Bachelor's	Limestone	Not applic	-86.9651	34.80563	Implied no	Yes	Implied no	Yes	Yes	Yes	Yes	Implied no	3170	1350
9	100830	Auburn Un	2013	Doctor's d	Montgom	Not applic	-86.1774	32.36994	Implied no	Yes	Implied no	Yes	Yes	Yes	Yes	Implied no	5084	3223
10	100858	Auburn Un	2013	Doctor's d	Lee Count	Not applic	-85.4924	32.6002	Implied no	Yes	24864	21093						
11	100937	Birmingham	2013	Bachelor's	Jefferson (	United Me	-86.8536	33.51545	Implied no	Yes	Implied no	Yes	Implied no	Yes	Implied no	Implied no	1188	1167
12	101073	Concordia	2013	Bachelor's	Dallas Cou	Lutheran C	-87.0235	32.42443	Yes	Yes	Implied no	Yes	Implied no	Yes	Implied no	Implied no	600	543
13	101189	Faulkner U	2013	Doctor's d	Montgom	Churches c	-86.2164	32.38418	Yes	Yes	Implied no	Yes	Implied no	Yes	Implied no	Implied no	3193	2321
14	101435	Huntingdo	2013	Bachelor's	Montgom	United Me	-86.2853	32.35094	Implied no	Yes	Implied no	Yes	Implied no	Yes	Implied no	Implied no	1110	902
15	101480	Jacksonville	2013	Doctor's d	Calhoun C	Not applic	-85.7666	33.82213	Implied no	Yes	Yes	Yes	Yes	Yes	Yes	Implied no	8693	5982
16	101541	Judson Col	2013	Bachelor's	Perry Cour	Baptist	-87.3161	32.63053	Yes	Yes	Implied no	Yes	Implied no	Yes	Implied no	Implied no	347	262
17	101587	University	2013	Master's d	Sumter Co	Not applic	-88.1861	32.59244	Yes	Yes	Implied no	Yes	Implied no	Yes	Implied no	Implied no	4427	3890
18	101675	Miles Colle	2013	Bachelor's	Jefferson (	Christian N	-86.9086	33.48131	Implied no	Yes	Implied no	Yes	Implied no	Yes	Implied no	Implied no	1666	1577
19	101693	University	2013	Master's d	Mobile Co	Southern E	-88.1289	30.79325	Yes	Yes	Implied no	Yes	Implied no	Yes	Implied no	Implied no	1610	1288
20	101709	University	2013	Master's d	Shelby Cou	Not applic	-86.8651	33.10625	Implied no	Yes	Implied no	Yes	Implied no	Yes	Implied no	Implied no	3062	2526
21	101879	University	2013	Master's d	Lauderdal	Not applic	-87.681	34.80658	Implied no	Yes	Implied no	Yes	Implied no	Yes	Implied no	Implied no	6931	5133
22	101912	Oakwood	2013	Master's d	Madison C	Seventh D	-86.6616	34.75663	Yes	Yes	Yes	Yes	Implied no	Yes	Implied no	Implied no	1903	1810
23	102049	Samford U	2013	Doctor's d	Jefferson (	Baptist	-86.7909	33.46458	Implied no	Yes	Implied no	Yes	Yes	Yes	Yes	Implied no	4833	4361
24	102094	University	2013	Doctor's d	Mobile Co	Not applic	-88.1819	30.69508	Implied no	Yes	Yes	Yes	Yes	Yes	Yes	Implied no	15065	12028
25	102234	Spring Hill	2013	Master's d	Mobile Co	Roman Ca	-88.1382	30.69475	Implied no	Yes	Yes	Yes	Yes	Yes	Yes	Implied no	1422	1210
26	102270	Stillman Cr	2013	Bachelor's	Tuscaloosa	Presbyteri	-87.5852	33.19923	Implied no	Yes	Implied no	Yes	Implied no	Yes	Implied no	Implied no	863	808
27	102298	Talladega I	2013	Bachelor's	Talladega I	United Chu	-86.1137	33.43426	Implied no	Yes	Implied no	Yes	Implied no	Yes	Implied no	Implied no	932	849
28	102368	Troy Unive	2013	Doctor's d	Pike Count	Not applic	-85.9502	31.80013	Yes	Yes	Implied no	Yes	Implied no	Yes	Implied no	Implied no	20573	10334

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
1	id_number	applicants	admission	enrolled	t_percent_o	percent_o	number_o	number_o	number_o	number_o	number_o	percent_o						
2	100654	6142	5521	1104	15	88	0	604	213	8	9	0	1	92	55	68.46333	27.82072	3.074505
3	100663	5689	4934	1773	6	93	0	2159	1533	542	70	0	5	21	61	86	13	1
4	100706	2054	1656	651	34	94	0	1091	444	38	28	1	4	12	44	79	14	4
5	100724	10245	5251	1479	18	87	0	599	147	48	27	0	0	91	61	58	37	4
6	100751	30975	17515	6454	23	76	0	4963	1649	493	38	0	1	11	55	40	57	3
7	100830	1958	1639	579	0	54	0	551	239	0	14	0	2	30	63	95	2	0
8	100858	15745	13027	3726	17	83	0	4192	1068	483	58	1	2	7	49	63	37	1
9	100937	1931	1240	356	32	82	0	299	0	0	0	1	4	9	46	68.46333	27.82072	3.074505
10	101435	1470	923	261	16	93	0	217	0	0	0	1	1	19	50	72	27	0
11	101480	3083	2567	1158	18	89	0	1365	339	0	71	1	1	25	58	76	24	1
12	101541	268	198	68	10	94	20	38	0	0	0	0	1	14	95	88	12	0
13	101587	462	460	380	0	100	53	242	660	0	205	1	1	49	72	68.46333	27.82072	3.074505
14	101693	866	617	259	13	87	13	286	57	0	0	2	1	24	68	68.46333	27.82072	3.074505
15	101709	1385	1209	531	3	96	0	325	165	0	33	1	1	15	67	94	5	0
16	101879	2542	2057	970	3	96	0	952	334	0	4	1	1	13	58	85	14	0
17	101912	2728	937	409	47	58	2	350	14	0	18	0	0	84	58	68.46333	27.82072	3.074505
18	102049	3447	2653	765	36	84	3	659	338	335	19	0	2	7	62	68.46333	27.82072	3.074505
19	102094	4814	4142	1878	8	92	0	1789	713	168	51	1	3	20	61	79	20	1
20	102234	6596	3052	431	20	92	0	212	45	0	14	1	1	16	60	68.46333	27.82072	3.074505
21	102270	4121	1772	257	6	59	0	137	0	0	0	0	0	92	49	68.46333	27.82072	3.074505
22	102368	5946	4056	2087	7	38	494	2821	1507	9	23	1	1	38	64	72	26	2
23	102377	10022	3519	650	43	75	0	344	59	49	0	0	1	76	60	33	64	0
24	102669	494	185	33	74	26	6	68	45	0	14	13	1	2	65	68.46333	27.82072	3.074505
25	104151	21770	17465	7171	64	51	0	8606	2772	723	90	1	6	4	43	58	32	10
26	104179	26329	20251	7401	66	52	0	6494	1663	854	135	1	6	3	52	63	34	0
27	104586	1689	1334	448	71	53	0	319	17	0	0	1	8	2	21	16	77	6
28	105330	33989	31057	4772	64	50	0	4427	1423	100	149	3	2	3	59	59	39	1

A1	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	id_number	tuition_an	tuition_an	tuition_an	tuition_an	total_price	total_price	percent_of_freshmen_receiving_any_financial_aid						
2	100654	5800	6828	7182	7182	21849	27441	97						
3	100663	5806	6264	6798	7206	22495	31687	90						
4	100690	8360	8720	6800	6870	34946.9	38784.69	100						
5	100706	7492	8094	8794	9192	23466	35780	87						
6	100724	7164	8082	7932	8720	18286	25222	93						
7	100751	7900	8600	9200	9450	27000	41500	75						
8	100830	6620	7580	8150	8750	17820	34020	90						
9	100858	7900	8698	9446	9852	25282	41794	76						
10	100937	27890	29290	30690	30690	45470	45470	98						
11	101073	7370	8090	8090	8090	18330	18330	100						
12	101189	14810	15880	17380	18230	31010	31010	99						
13	101435	20990	21990	22500	23500	33885	33885	100						
14	101480	5424	6120	6360	7092	21237	28029	93						
15	101541	13900	14460	15100	15630	28840	28840	100						
16	101587	6524	6918	7320	7660	18162	24672	95						
17	101675	9896	10490	11014	11454	22318	22318	99						
18	101693	16120	17220	18080	18690	32570	32570	93						
19	101709	7750	8520	9280	10000	23340	33030	88						
20	101879	5426	6084	6528	7078	16933	22525	82						
21	101912	14250	14678	15414	16234	36295	36295	94						
22	102049	21942	23963	25150	26328	40900	40900	98						
23	102094	5748	6204	6660	6948	19980	26628	91						
24	102234	26730	28060	29450	30924	46170	46170	100						
25	102270	14464	15062	15665	15901	26877	26877	100						
26	102298	11111	11143	11492	11492	21296	21296	98						
27	102368	5860	6412	6844	7276	18228	24564	78						
28	102377	16750	17870	18900	18900	32853	32853	96						

# TABLEAU DATA SOURCES :

## Book3a\_na

Filters  
0 | Add

Book3a\_na.csv is made of 1 table. ⓘ

Book3a\_na.csv

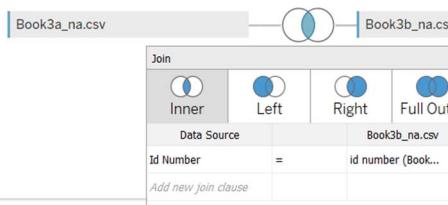
X

Book3a_na.csv		24 fields 1532 rows											
										100	→ rows	⚙️	▼
Table Details		#	Abc Book3a_na.csv	#	Abc Book3a_na.csv	#	Abc Book3a_na.csv	#	Abc Book3a_na.csv	#	Abc Book3a_na.csv	#	Abc Book3a_na.csv
Id Number	Name	Year	Highest Degree Offered	County Name	Religious Affiliation	Longitude Location Of I...	Latitude Locatio						
100654	Alabama A & M University	2013	Doctor's degree - research/s...	Madison County	Not applicable	-86.5685							
100663	University of Alabama at Bir...	2013	Doctor's degree - research/s...	Jefferson County	Not applicable	-86.8092							
100690	Amridge University	2013	Doctor's degree - research/s...	Montgomery County	Churches of Christ	-86.1740							
100706	University of Alabama in Hun...	2013	Doctor's degree - research/s...	Madison County	Not applicable	-86.6384							
100724	Alabama State University	2013	Doctor's degree - research/s...	Montgomery County	Not applicable	-86.2957							
100751	The University of Alabama	2013	Doctor's degree - research/s...	Tuscaloosa County	Not applicable	-87.5458							
100812	Athens State University	2013	Bachelor's degree	Limestone County	Not applicable	-86.9651							
100830	Auburn University at Montgo...	2013	Doctor's degree - research/s...	Montgomery County	Not applicable	-86.1774							
100858	Auburn University	2013	Doctor's degree - research/s...	Lee County	Not applicable	-85.4924							
100937	Birmingham Southern College	2013	Bachelor's degree	Jefferson County	United Methodist	-86.8536							
101073	Concordia College Alabama	2013	Bachelor's degree	Dallas County	Lutheran Church - Mis...	-87.0235							

## Book3a\_na+

Filters  
0 | Add

Book3a\_na.csv is made of 2 tables. ⓘ



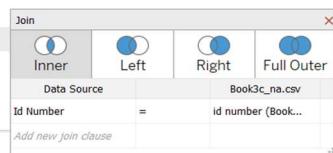
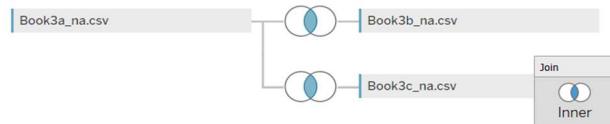
X

Book3a_na.csv		42 fields 1314 rows											
										100	→ rows	⚙️	▼
Table Details		#	Abc Book3a_na.csv	#	Abc Book3a_na.csv	#	Abc Book3a_na.csv	#	Abc Book3b_na.csv	#	Abc Book3b_na.csv	#	Abc Book3b_na.csv
Graduate Enrollment	Level Of Institution	Control Of Institution	Degree Of Urbanization ...	Id number (Book3b_na...	Applicants Total 2013	Admissions Total 2013							
969	Four or more years	Public	City: Midsize		100654		6,142						
7,066	Four or more years	Public	City: Midsize		100663		5,689						
1,680	Four or more years	Public	City: Midsize		100706		2,054						
719	Four or more years	Public	City: Midsize		100724		10,245						
5,312	Four or more years	Public	City: Small		100751		30,975						
762	Four or more years	Public	City: Midsize		100830		1,958						
5,065	Four or more years	Public	City: Small		100858		15,745						
0	Four or more years	Private not-for-profit	City: Midsize		100937		1,931						
0	Four or more years	Private not-for-profit	City: Midsize		101073		1,470						

## ⊖ Cmplt\_Dataset

Filters  
0 | Add

Cmplt\_Dataset is made of 3 tables. ①



Cmplt_Dataset		50 fields 1311 rows							
Table Details	#	Graduate Enrollment	Abc	Abc	Degree Of Urbanization ...	#	Applicants Total 2013	Ac	
	4,051	969	Four or more years	Public	City: Midsize	100654	6,142		
	11,502	7,066	Four or more years	Public	City: Midsize	100663	5,689		
	5,696	1,680	Four or more years	Public	City: Midsize	100706	2,054		
	5,356	719	Four or more years	Public	City: Midsize	100724	10,245		
	29,440	5,312	Four or more years	Public	City: Small	100751	30,975		
	4,322	762	Four or more years	Public	City: Midsize	100830	1,958		
	19,799	5,065	Four or more years	Public	City: Small	100858	15,745		
	1,188	0	Four or more years	Private not-for-profit	City: Midsize	100937	1,931		
	1110	0	Four or more years	Private not-for-profit	City: Midsize	101435	1,470		

⋮ Sehajparkash S... ▾ [4 5 6 7] [grid] [refresh]

# CALCULATED FIELDS IN TABLEAU

# id number (Book3c\_na.csv)

=T|F Associate degree(bool)

- =T|F No. of Bachelor's degree
- =T|F No. of Doctor S Degree Other
- =T|F No. of Doctor's degree
- =T|F No. of Doctor's degree -professional
- =T|F No. of Master's degree
- =T|F No. of Post Master's degree
- =T|F No. of Postbaccalaureate Certificate
- ∅ Religious Affiliation (group)

Abc Measure Names

Book3a\_na.csv

- # Full Time Enrollment
- # Graduate Enrollment
- " id number (Book3c\_na.csv)
- =T|F Associate degree(bool)
- =T|F No. of Bachelor's degree
- =T|F No. of Doctor S Degree Other
- =T|F No. of Doctor's degree
- =T|F No. of Doctor's degree -professional
- =T|F No. of Master's degree
- =T|F No. of Post Master's degree
- =T|F No. of Postbaccalaureate Certificate
- ∅ Religious Affiliation (group)

Abc Measure Names

Book3a\_na.csv

- # Full Time Enrollment
- # Graduate Enrollment
- =T|F Associate degree(bool)
- =T|F Bachelor's degree(bool)
- =T|F Doctor S Degree Other(bool)
- =T|F No. of Doctor's degree
- =T|F No. of Doctor's degree -professional
- =T|F No. of Master's degree
- =T|F No. of Post Master's degree
- =T|F No. of Postbaccalaureate Certificate
- ∅ Religious Affiliation (group)

Abc Measure Names

Book3a\_na.csv

- # Full Time Enrollment
- # Graduate Enrollment

**Associate degree(bool)**

[Offers Associate S Degree]== "Yes"

The calculation is valid. 2 Dependencies ▾ Apply OK

**Bachelor's degree(bool)**

[Offers Bachelor S Degree]== "Yes"

The calculation is valid. Apply OK

**Doctor S Degree Other(b**

[Offers Doctor S Degree Other]== "Yes"

The calculation is valid. Apply OK

Screenshot of a software interface showing three validation dialogs for calculated fields.

**Left Panel:** A list of calculated fields and measures. The calculated fields are grouped under "Book3a\_na.csv".

- Bachelor's degree(bool)
- Doctor S Degree Other(bool)
- Doctor's degree(bool)**
- No. of Doctor's degree -professional
- No. of Master's degree
- No. of Post Master's degree
- No. of Postbaccalaureate Certificate
- Religious Affiliation (group)
- Measure Names**
- Book3a\_na.csv**
  - # Full Time Enrollment
  - # Graduate Enrollment
  - Latitude Location Of Institution
  - Bachelor's degree(bool)
  - Doctor S Degree Other(bool)
  - Doctor's degree -professional(bool)**
  - Doctor's degree(bool)
  - No. of Master's degree
  - No. of Post Master's degree
  - No. of Postbaccalaureate Certificate
  - Religious Affiliation (group)
  - Measure Names**
- Book3a\_na.csv**
  - # Full Time Enrollment
  - # Graduate Enrollment
  - Latitude Location Of Institution
  - Doctor S Degree Other(bool)
  - Doctor's degree -professional(bool)
  - Doctor's degree(bool)**
  - Master's degree(bool)
  - No. of Post Master's degree
  - No. of Postbaccalaureate Certificate
  - Religious Affiliation (group)
  - Measure Names**
- Book3a\_na.csv**
  - # Full Time Enrollment
  - # Graduate Enrollment
  - Latitude Location Of Institution
  - Longitude Location Of Institution

**Top Dialog:** Validation for "Doctor's degree(bool)"

Automatic

Doctor's degree(bool)

[Offers Doctor S Degree Other]=="Yes"

The calculation is valid. 5 Dependencies

**Middle Dialog:** Validation for "Doctor's degree -profess"

Automatic

Doctor's degree -profess

[Offers Doctor S Degree Professional Practice]=="Yes"

The calculation is valid. 2 Dependencies

**Bottom Dialog:** Validation for "Master's degree(bool)"

Automatic

Master's degree(bool)

[Offers Master S Degree]=="Yes"

The calculation is valid. 2 Dependencies

No. of Postbaccalaureate Certificate

**Post Master's degree(bool)**

Religious Affiliation (group)

Measure Names

**Book3a\_na.csv**

- # Full Time Enrollment
- # Graduate Enrollment
- @ Latitude Location Of Institution
- @ Longitude Location Of Institution
- # Part Time Enrollment
- # Total Enrollment
- # Undergraduate Enrollment

**Book3b\_na.csv**

Post Master's degree(bool)

**Postbaccalaureate Certificate(bool)**

Religious Affiliation (group)

Measure Names

**Book3a\_na.csv**

- # Full Time Enrollment
- # Graduate Enrollment
- @ Latitude Location Of Institution
- @ Longitude Location Of Institution
- # Part Time Enrollment
- # Total Enrollment
- # Undergraduate Enrollment

**Doctor's Degree Professional Practice T**

- # Doctor's degree T
- # Post Master S Certificate T
- # Postbaccalaureate Certificate T
- # Associate's degree T
- # Avg\_Tuition\_fees
- # Bachelor's Degree T
- # Calculation1
- # Doctor's Degree Other Total
- # Master S Degree Total
- # Sum Tution fee
- # Total degrees
- # Total degrees distributed

Detail | Tooltip |

Post Master's degree(bo)

[Offers Post Master S Certificate]=="Yes"

The calculation is valid. 2 Dependencies ▾

**Apply** **OK**

Detail | Tooltip |

Postbaccalaureate Certif

[Offers Postbaccalaureate Certificate]=="Yes"

The calculation is valid. 2 Dependencies ▾

**Apply** **OK**

Detail | Tooltip |

Professional Practice T

SUM(if  
[Offers Doctor S Degree Professional Practice]  
== 'Yes' then 1 else 0 end)

The calculation is valid. 4 Dependencies ▾

**Apply** **OK**

The screenshot shows a software interface with a sidebar containing a list of items, each preceded by a '#'. The items are:

- # Doctor's Degree Professional Practice T
- # **Doctor's degree T**
- # Post Master S Certificate T
- # Postbaccalaureate Certificat
- # Associate's degree T
- # Avg\_Tuition\_fees
- # Bachelor's Degree T
- # Calculation1
- # Doctor's Degree Other Total
- # Master S Degree Total
- # Sum Tution fee
- # Total degrees
- # Total degrees distributed
- # zero field

Three floating dialog boxes are displayed over the sidebar:

- Doctor's degree T**:  
Calculation: `SUM(if[Doctor's degree(bool)] == TRUE then 1 else 0 end)`  
Status: The calculation is valid.  
Dependencies: 3 Dependencies  
Buttons: Apply, OK
- Post Master S Certificate T**:  
Calculation: `SUM(if[Offers Post Master S Certificate] == 'Yes' then 1 else 0 end)`  
Buttons: SUM(expression), Apply, OK
- Postbaccalaureate Certificate T**:  
Calculation: `SUM(if[Offers Postbaccalaureate Certificate] == 'Yes' then 1 else 0 end)`  
Buttons: Apply, OK

```
# Doctor's degree T
# Post Master S Certificate T
# Postbaccalaureate Certificate T
# Associate's degree T
# Avg_Tuition_fees
# Bachelor's Degree T
# Calculation1
# Doctor's Degree Other Total
# Master S Degree Total
# Sum Tution fee
# Total degrees
# Total degrees distributed
# zero field
# Doctor's degree T
# Post Master S Certificate T
# Postbaccalaureate Certificate T
# Associate's degree T
# Avg_Tuition_fees
# Bachelor's Degree T
# Calculation1
# Doctor's Degree Other Total
# Master S Degree Total
# Sum Tution fee
# Total degrees
# Total degrees distributed
# zero field
# Avg_Tuition_fees
# Bachelor's Degree T
# Calculation1
# Doctor's Degree Other Total
# Master S Degree Total
# Sum Tution fee
# Total degrees
# Total degrees distributed
# zero field
# Cmplt_Dataset (Count)
# Latitude (generated)
# Longitude (generated)
# Measure Values
```

Associate's degree T

```
SUM(if [Offers Associate S Degree] == 'Yes'  
then 1 else 0 end)
```

The calculation is valid. 4 Dependencies ▾ Apply OK

Avg\_Tuition\_fees

```
([Tuition And Fees 2010 11]+[Tuition And Fees 2011 12]+[
```

The calculation is valid. 2 Dependencies ▾ Apply OK

Bachelor's Degree T

```
SUM(if [Offers Bachelor S Degree] == 'Yes'  
then 1 else 0 end)
```

The calculation is valid. 3 Dependencies ▾ Apply OK

# Calculation1

# Doctor's Degree Other Total

# Master S Degree Total

# Sum Tution fee

# Total degrees

# Total degrees distributed

# zero field

# Cmplt\_Dataset (Count)

# Latitude (generated)

# Longitude (generated)

# Measure Values

The calculation is valid. 3 Dependencies ▾ Apply OK

# Doctor's Degree Other Total

# Master S Degree Total

# Sum Tution fee

# Total degrees

# Total degrees distributed

# zero field

# Cmplt\_Dataset (Count)

# Latitude (generated)

# Longitude (generated)

# Measure Values

The calculation is valid. 3 Dependencies ▾ Apply OK

# Doctor's Degree Other Total

# Master S Degree Total

# Sum Tution fee

# Total degrees

# Total degrees distributed

# zero field

# Cmplt\_Dataset (Count)

# Latitude (generated)

# Longitude (generated)

# Measure Values

The calculation is valid. 3 Dependencies ▾ Apply OK

- # Doctor's Degree Other Total
- # Master S Degree Total
- # Sum Tution fee
- # **Total degrees**
- # Total degrees distributed
- # zero field
- # Cmplt\_Dataset(Count)
- ⌚ Latitude (generated)
- ⌚ Longitude (generated)
- # Measure Values

Total degrees

```
[ Doctor's Degree Professional Practice T]+  
[Doctor's Degree Other Total]+  
[ Doctor's degree T]+[Master S Degree Total]+  
[Bachelor's Degree T]+  
[ Postbaccalaureate Certificate T]+  
[ Post Master S Certificate T]+[Associate's degree T]
```

X

Apply

OK

- Data Source    Sheet 1    Sheet 2
- # Doctor's Degree Other Total
  - # Master S Degree Total
  - # Sum Tution fee
  - # Total degrees
  - # **Total degrees distributed**
  - # zero field
  - # Cmplt\_Dataset(Count)
  - ⌚ Latitude (generated)
  - ⌚ Longitude (generated)
  - # Measure Values

Total degrees distributed

```
[ Bachelor S Degree receiving- Number Of Students ]+  
[ Doctor S Degree receiving-Number Of Students ]+  
[ Master S Degree receiving-Number Of Students ]+  
[Postbaccalaureate Or Post Master S Certificate receivin  
[ Associate S Degree receiving-Number Of Students ]
```

X

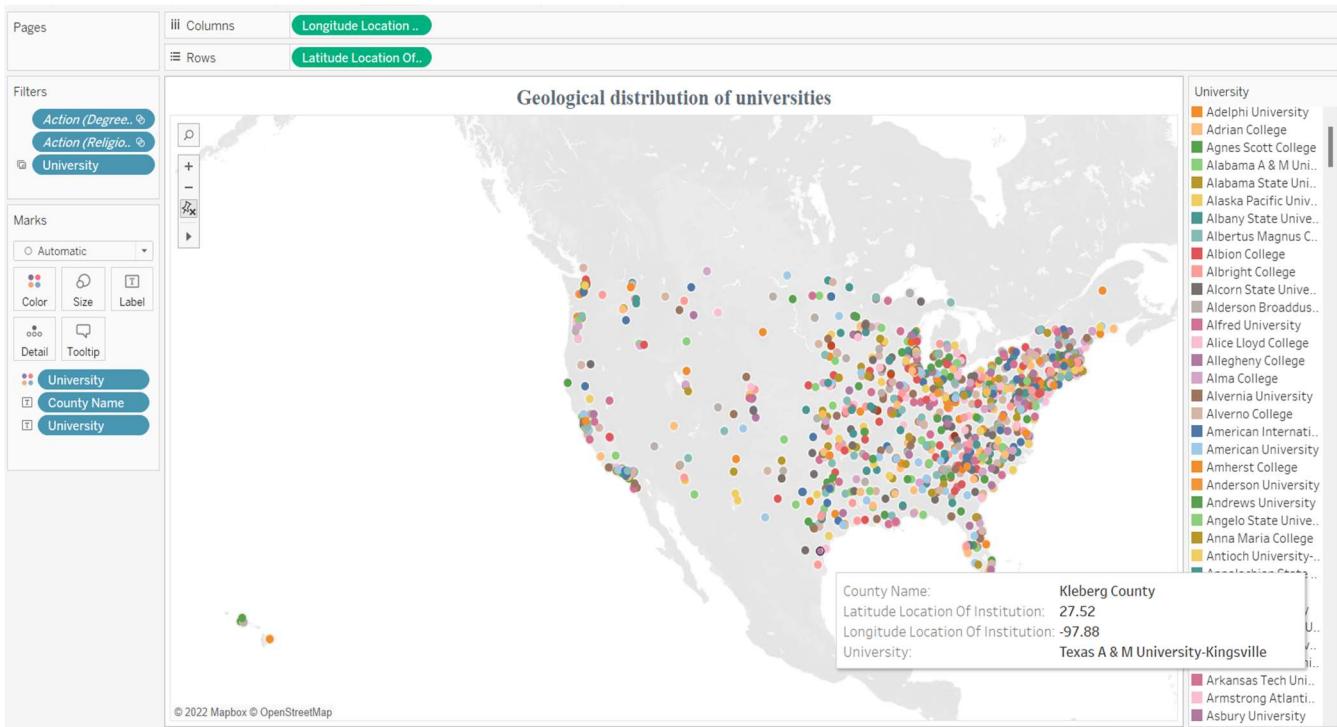
Apply

OK

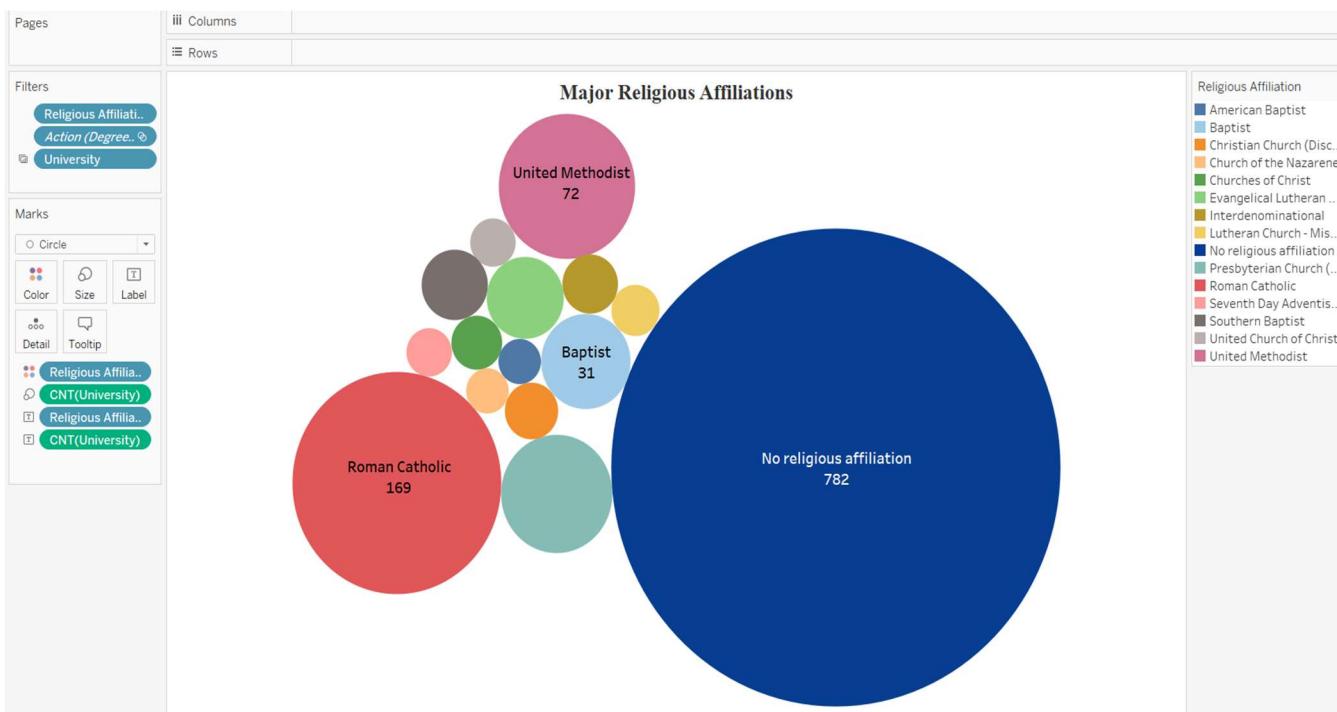
- Data Source    Sheet 1    Sheet 2

# DATA VISUALIZATIONS

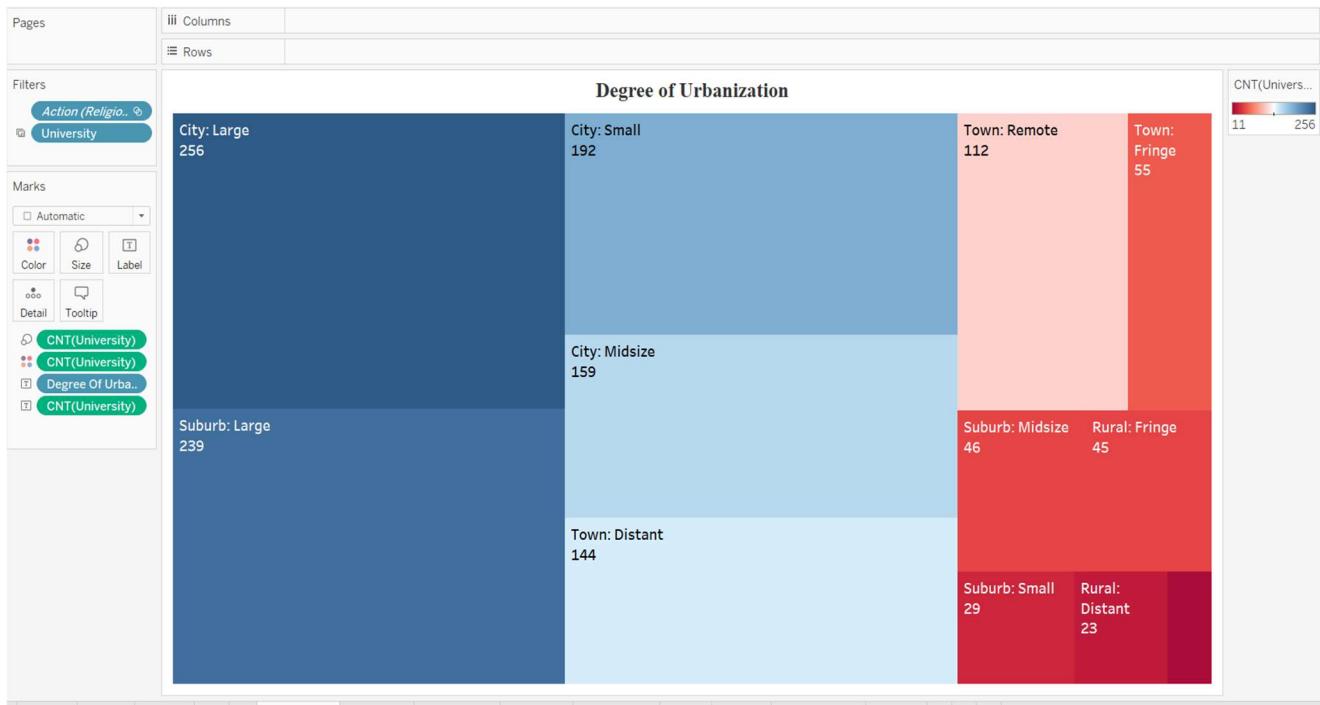
## Geological distribution of universities :



## Major Religious Affiliations :



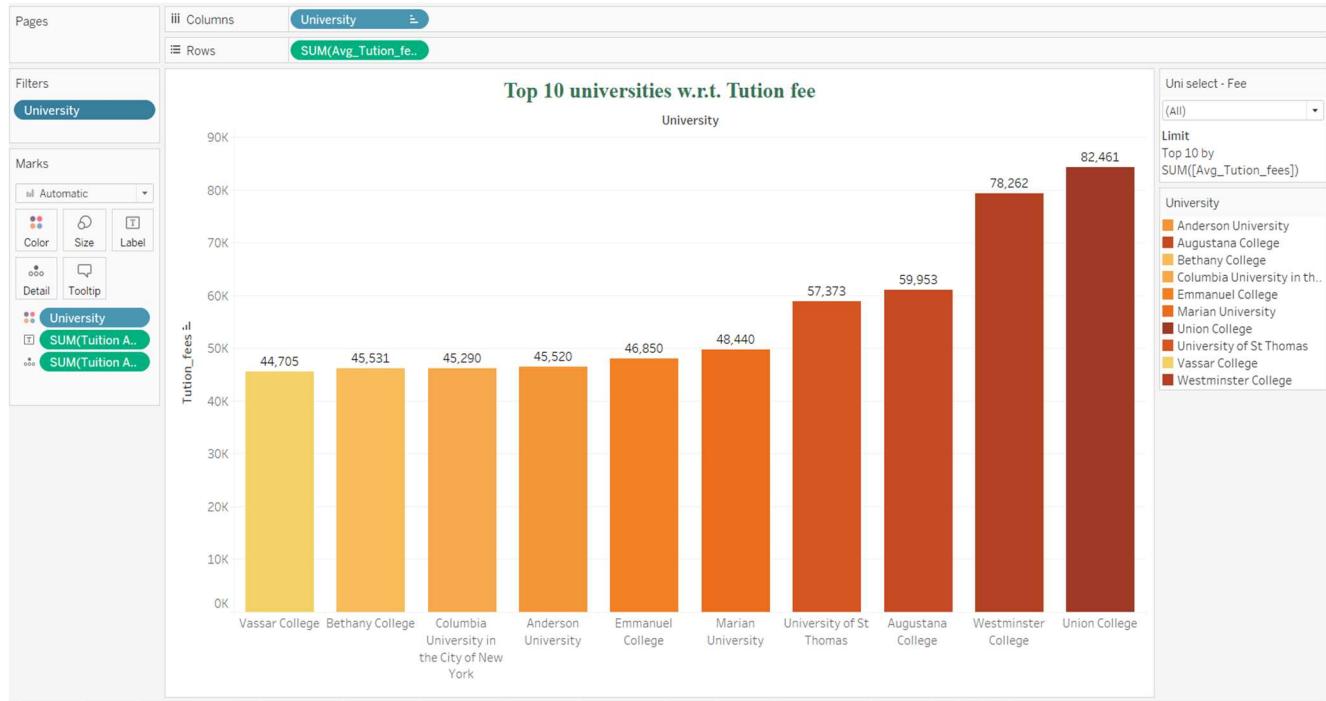
## Degree of Urbanization :



## Top 10 universities w.r.t. Enrollment :



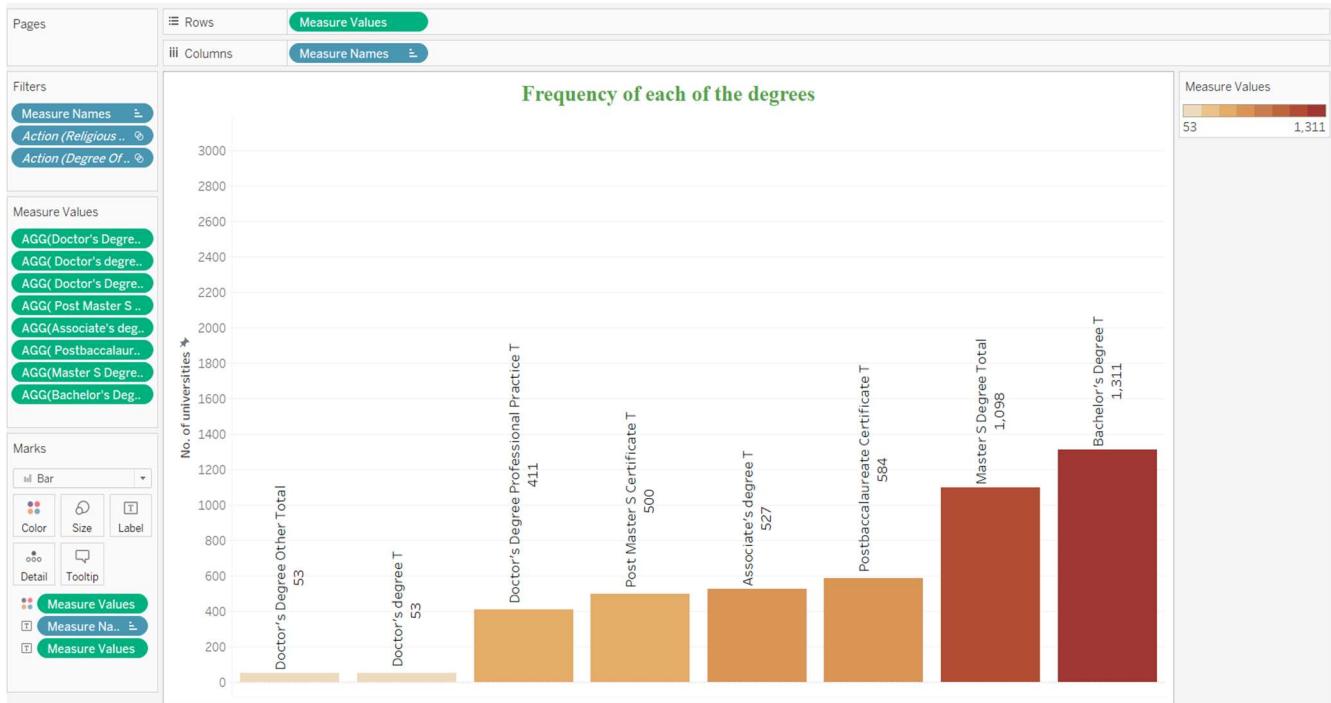
## Top 10 universities w.r.t. Tuition fee :



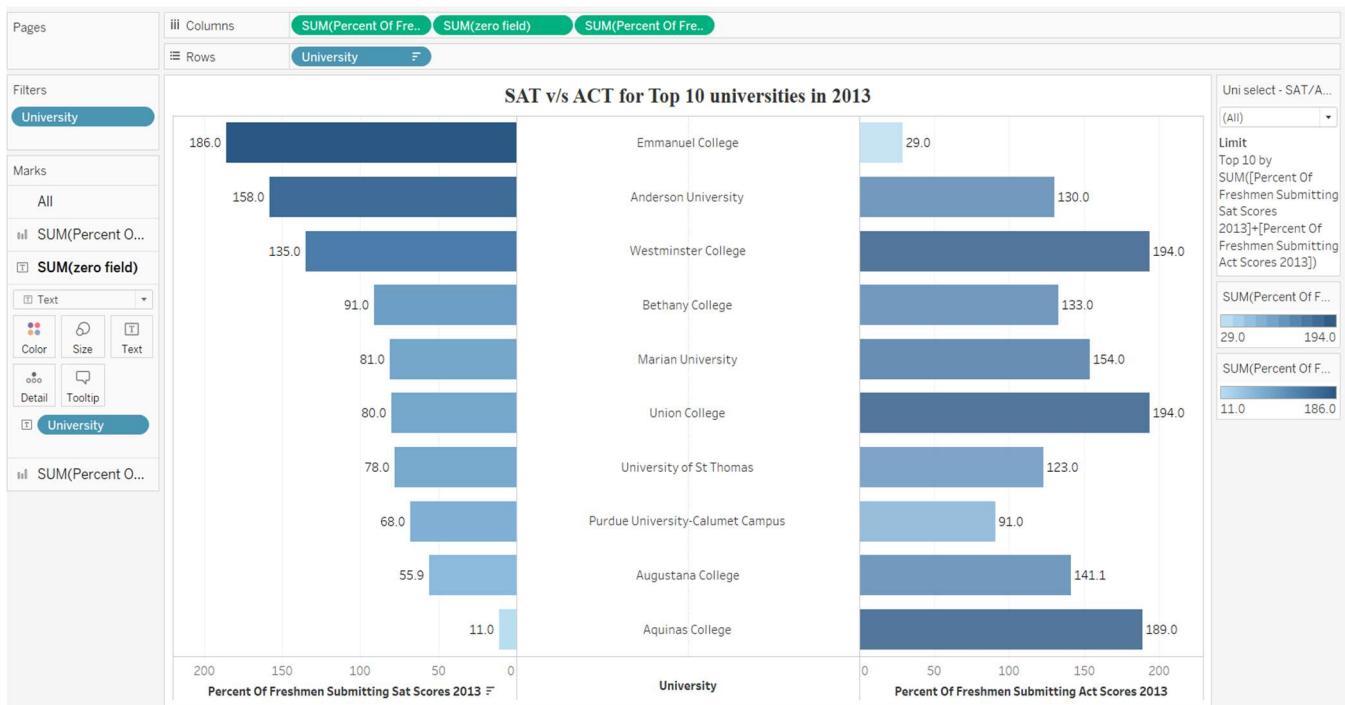
## Distribution of degrees awarded by top 5 universities :



## Frequency of each of the degrees :



## SAT v/s ACT for Top 10 universities in 2013 :



# Universities offering all the 8 degrees :

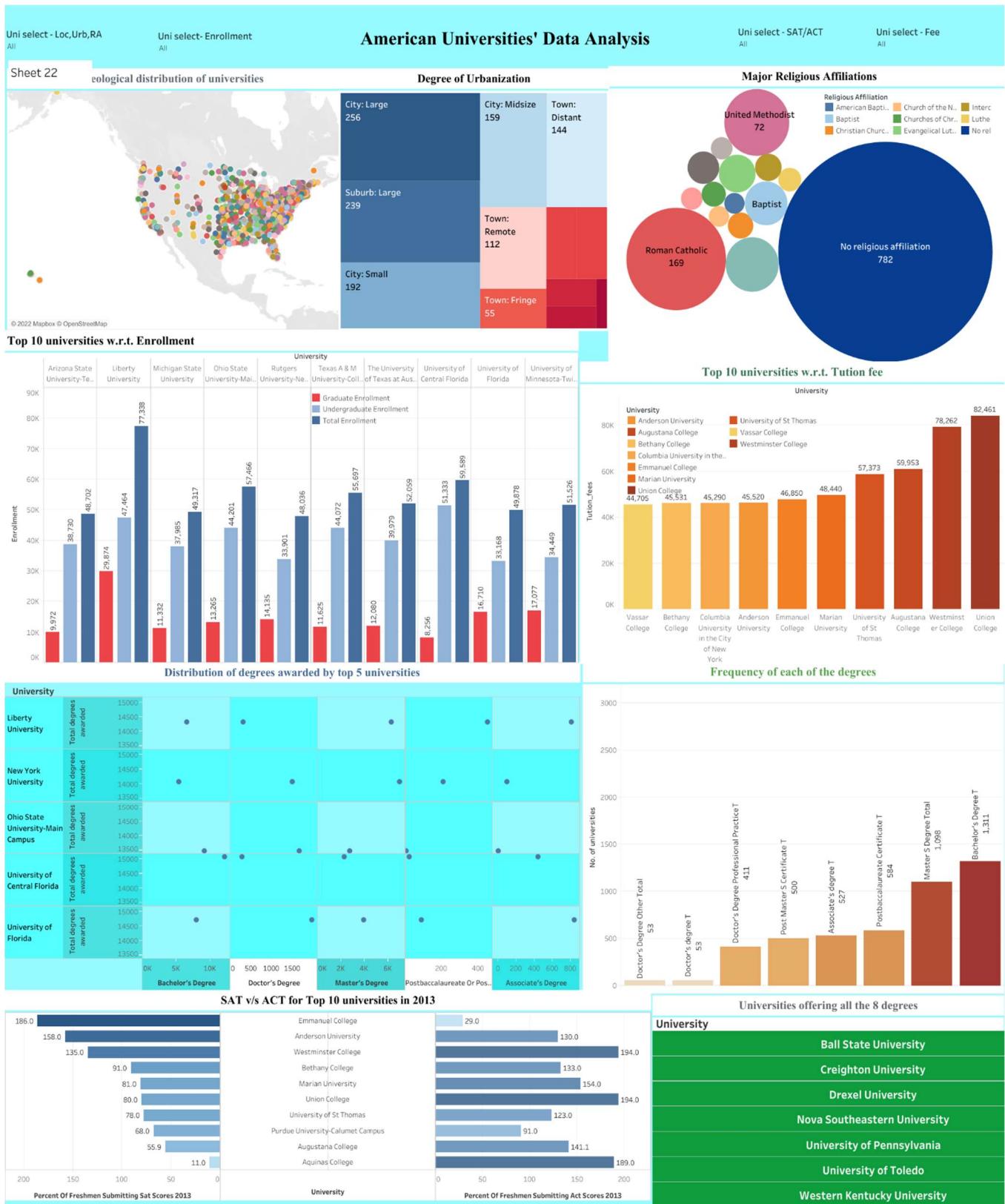
Pages iii Columns Rows University County Name

Filters Associate degree(bool..) Doctor's degree(bool..) Doctor's degree -pro.. Master's degree(bool..) Post Master's degree.. Postbaccalaureate C.. Action (Religious ..) Action (Degree Of..)

Marks Automatic Color Size Text Detail Tooltip

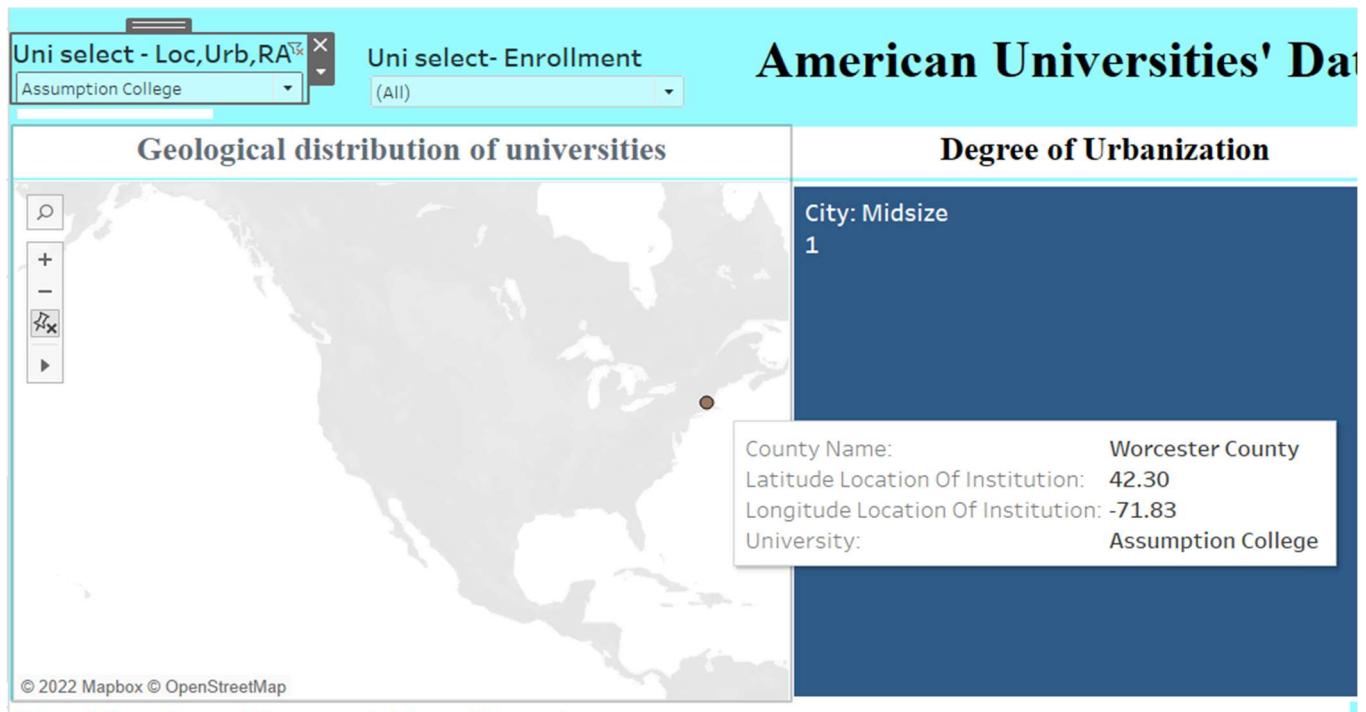
Universities offering all the 8 degrees	
University	County Name
Ball State University	Delaware County
Creighton University	Douglas County
Drexel University	Philadelphia County
Nova Southeastern University	Broward County
University of Pennsylvania	Philadelphia County
University of Toledo	Lucas County
Western Kentucky University	Warren County

# DASHBOARD

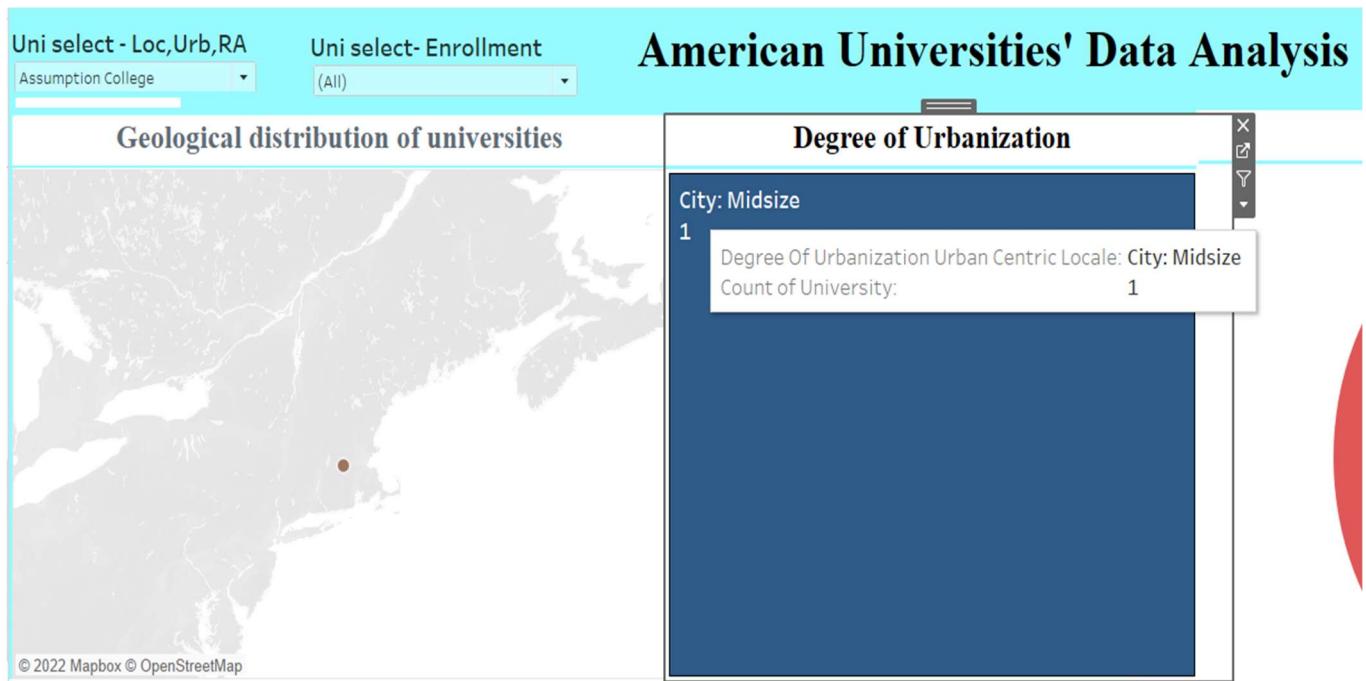


# QUERIES (DATA RETRIEVAL)

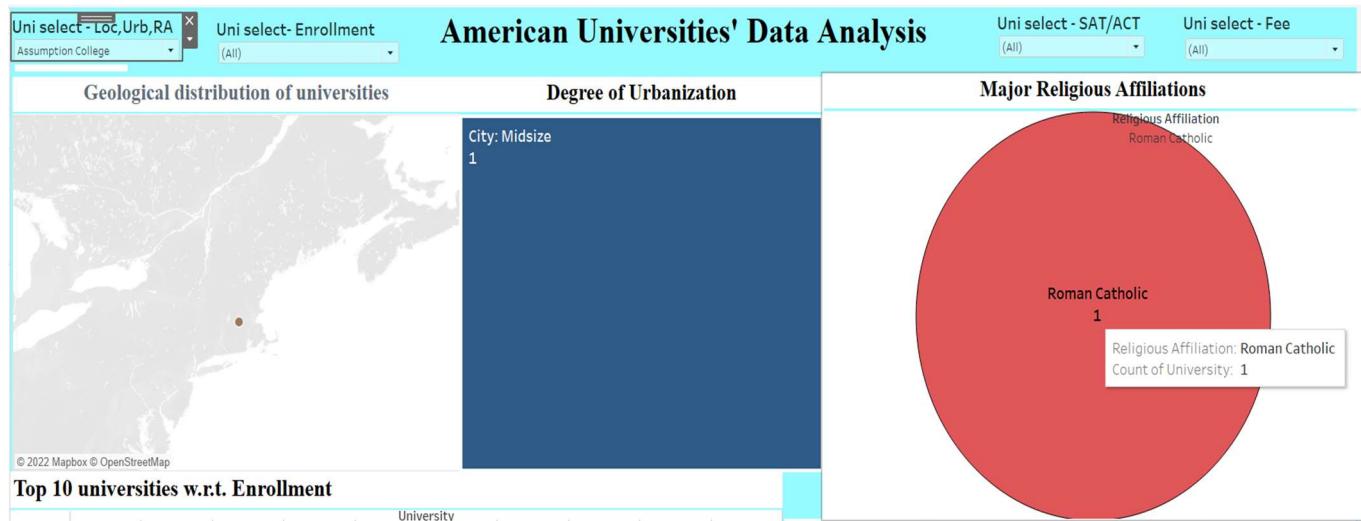
## Geological location of one/many university :



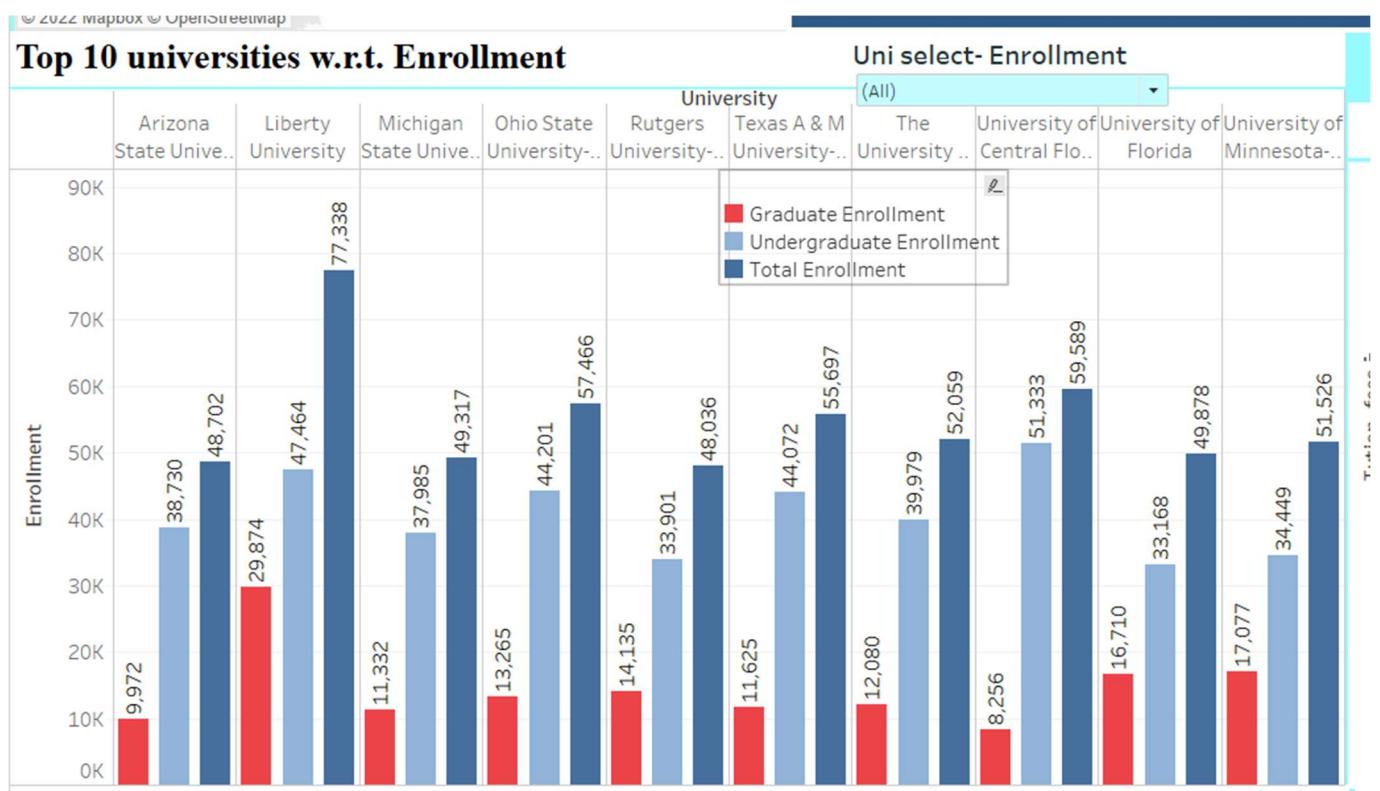
## Degree of urbanization of a university:



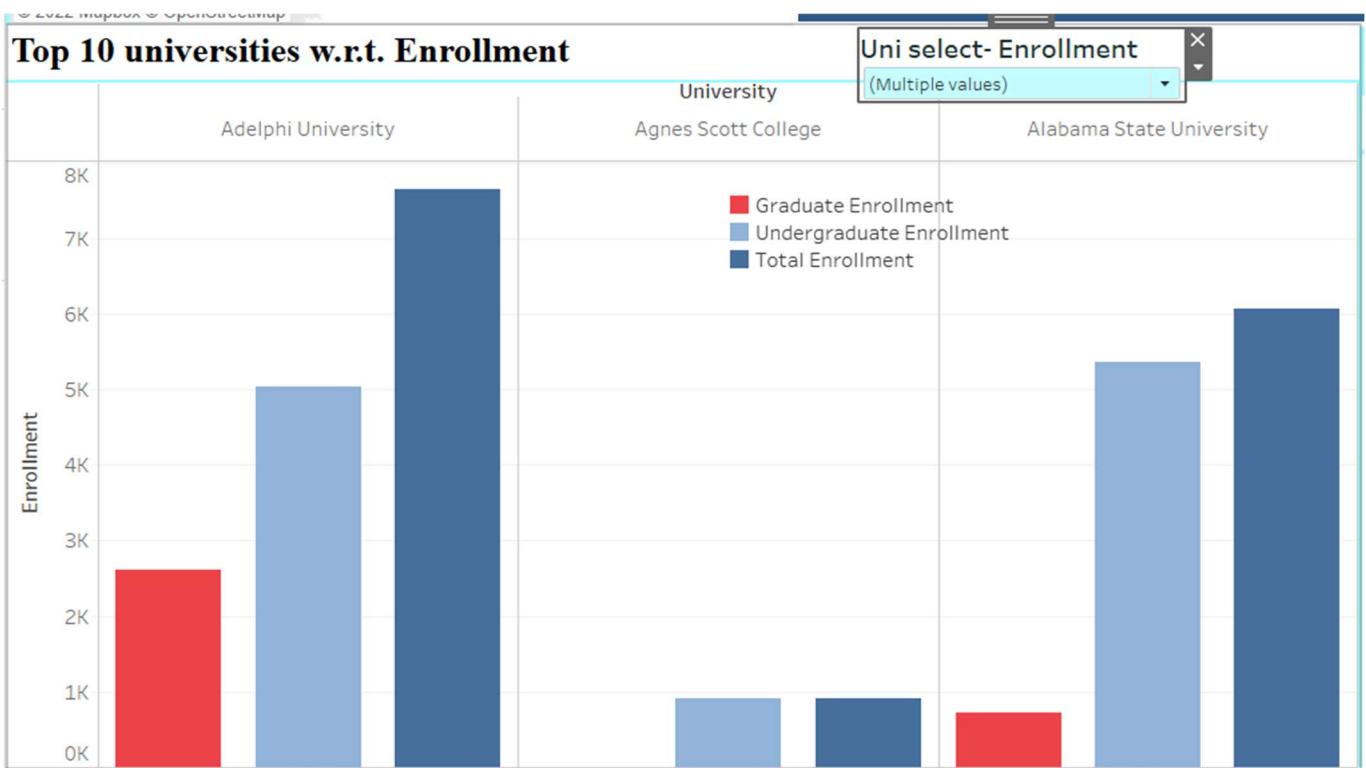
## Religious affiliation of a university :



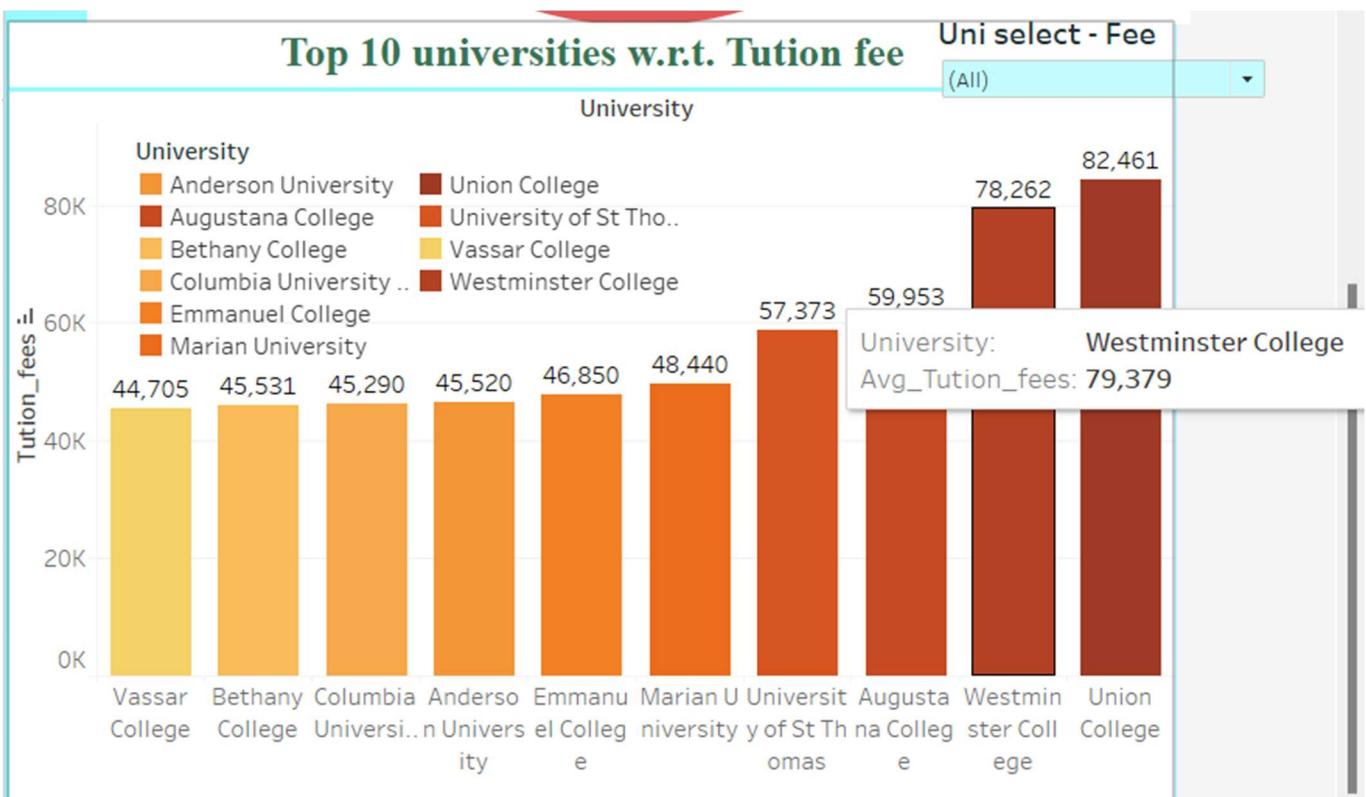
## Top 10 universities in enrolment:



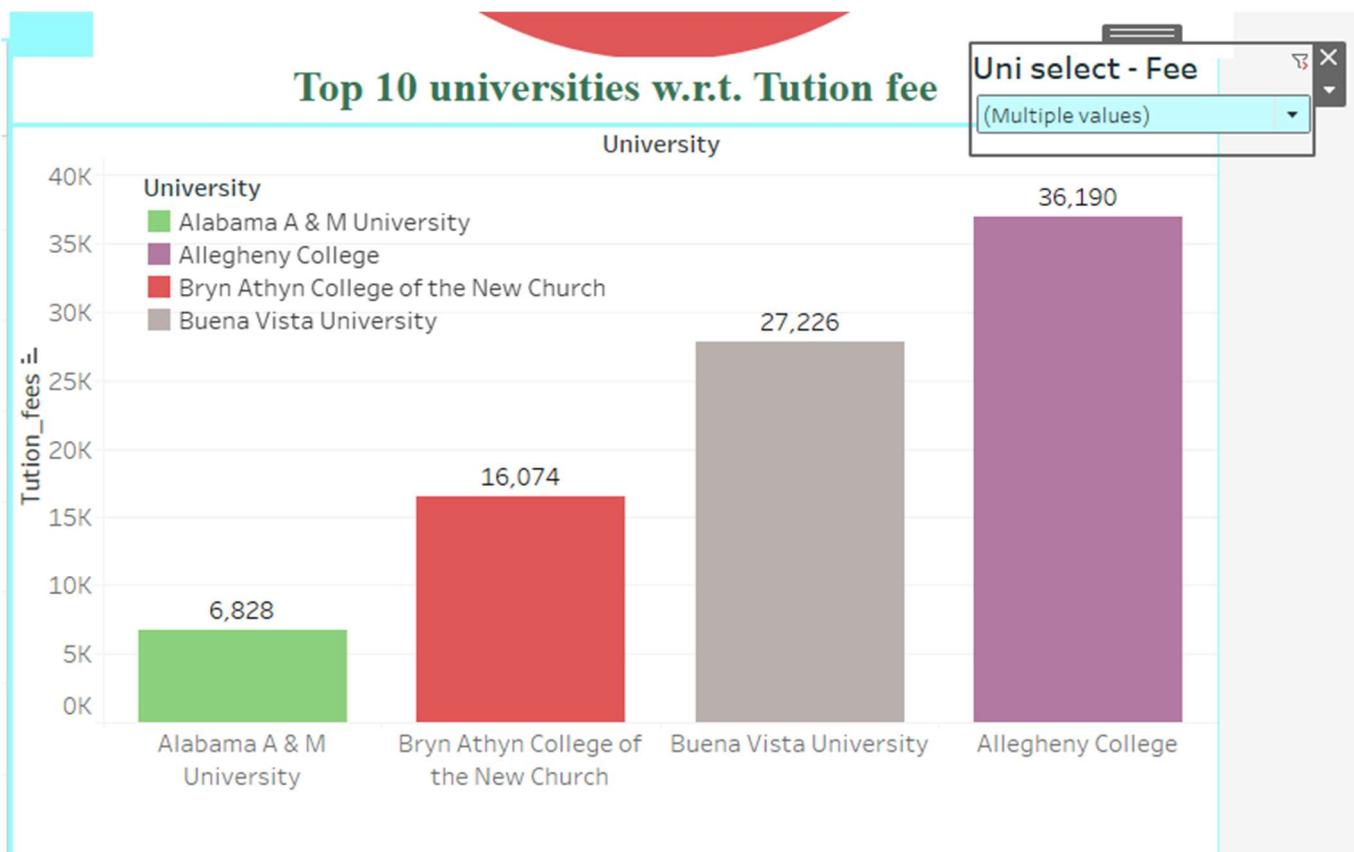
## Enrolment in any of the universities:



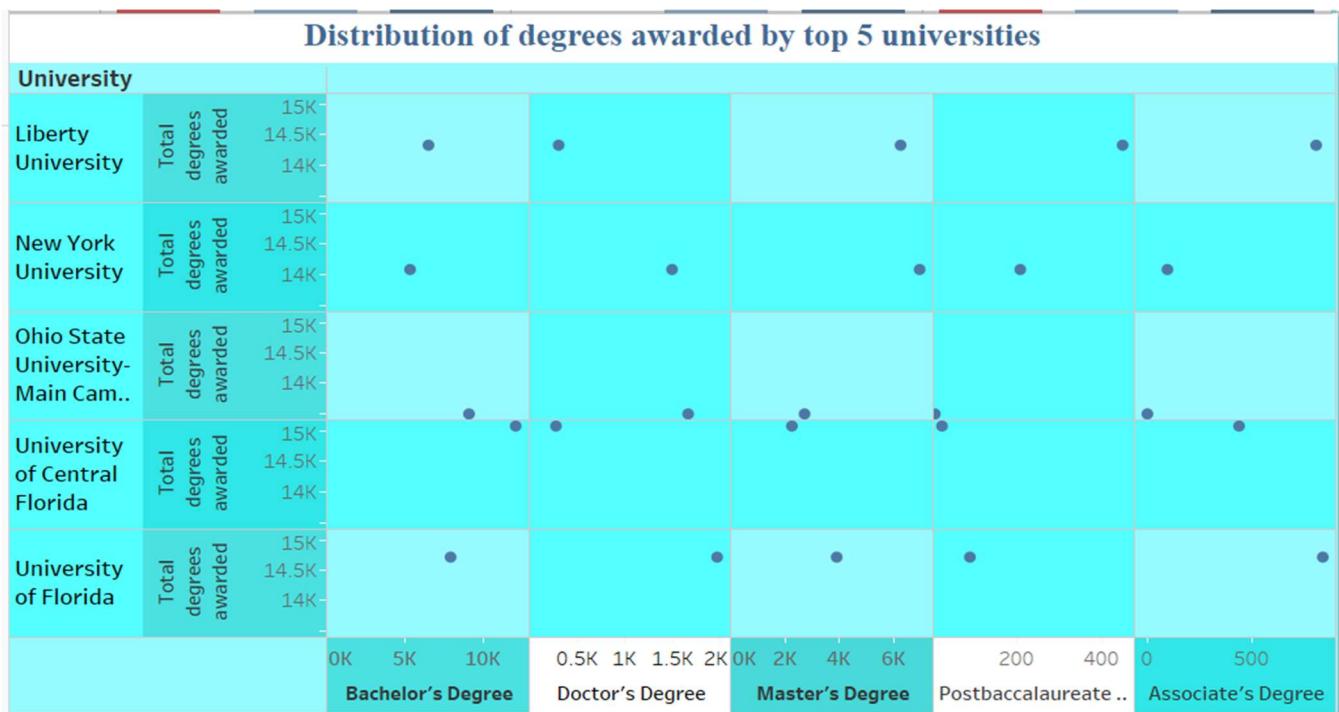
## Top 10 universities with regards to tuition fee:



## Tuition fee of any of the universities:

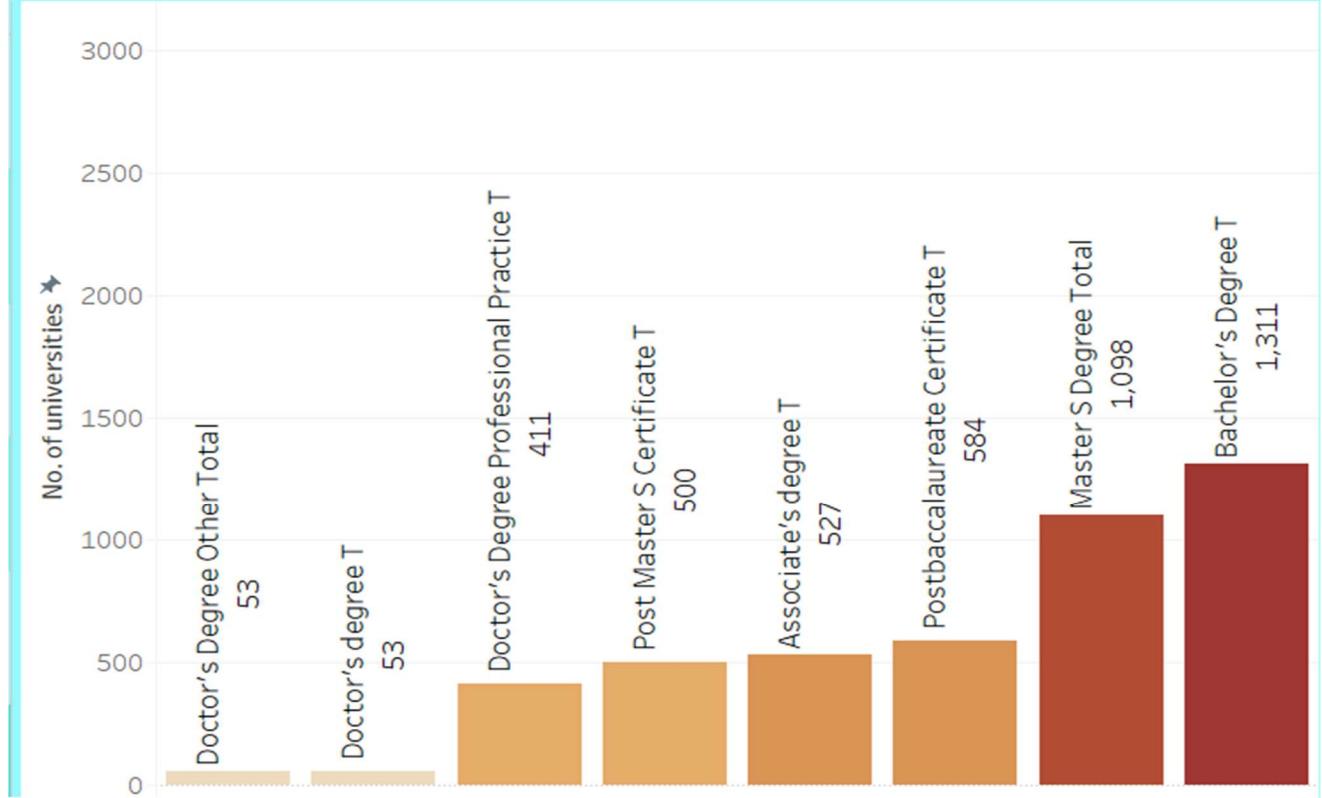


## Various degrees awarded by top 5 universities :



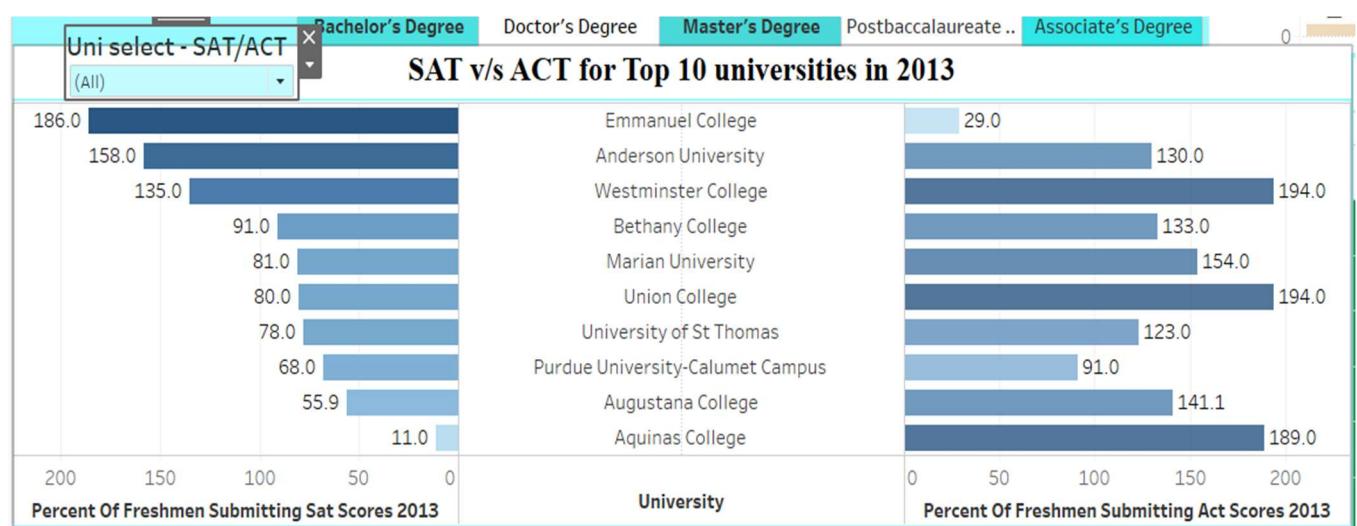
## Weightage of all the 8 degrees in the USA :

**Frequency of each of the degrees**

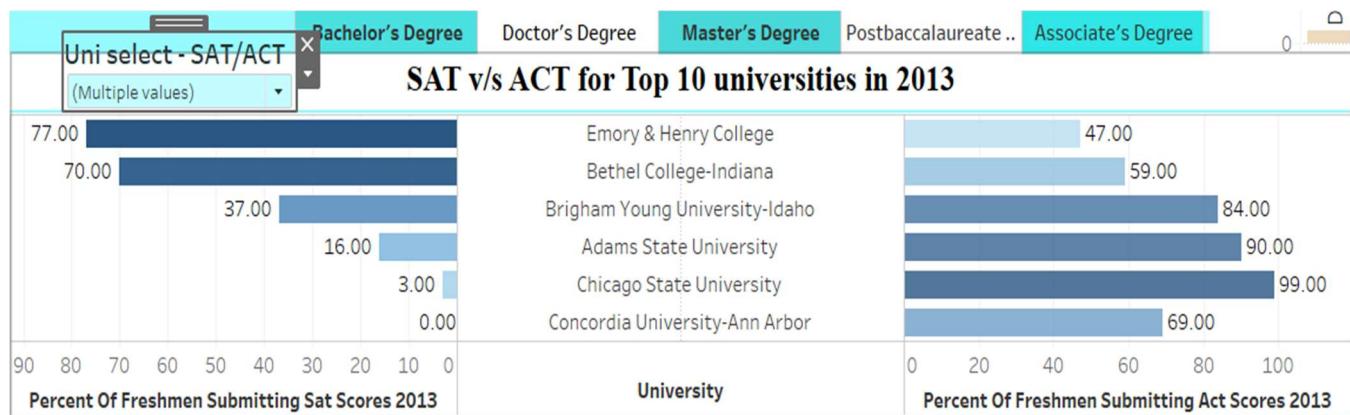


**Bachelor' s degree is pursued the most in the USA**

## SAT v/s ACT as the admission criteria in top 10 universities:



## SAT v/s ACT as the admission criteria for any of the universities :



## Universities which offer all the types of degrees :

### Universities offering all the 8 degrees

University	County Name
Ball State University	Delaware County
Creighton University	Douglas County
Drexel University	Philadelphia County
Nova Southeastern University	Broward County
University of Pennsylvania	Philadelphia County
University of Toledo	Lucas County
Western Kentucky University	Warren County