CS 533 Assignment 1

Introduction

This assignment is about exploratory analysis and describing a data set. The basic outline of this assignment are:

- Obtaining a data set from a public source and use its documentation to understand it
- Setting up a Jupyter notebook and data set to begin a new analysis
- Carrying out an exploratory analysis to understand a data set’s contents and communicate them to others

Environment Setup

We will be using pandas to load and manipulate data, seaborn, and matlplotlib to plot various charts for data representation.

In [1]:

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

Data

The data that we will be using in this assignment is the dataset of **College ScoreCard** from the Department of Education's i.e. **Most recent cohorts all data element** under **Most Recent Institution-level Data**. The data set consists of a tabular CSV file named **Most-Recent-Cohorts-All-Data-Elements** that contains the data we will be exploring and analyzing. Additional files that we need to describe data will be a Data Documentation file named **FullDataDocumentation** and a data dictionary named **CollegeScorecardDataDictonary**.

The tabular file might contain any number of data instances and variables.

In [2]:

```
df = pd.read_csv("Most-Recent-Cohorts-All-Data-Elements.csv", low_memory=False)
df.info(memory_usage = "deep")
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 6694 entries, 0 to 6693
Columns: 2392 entries, UNITID to DOLPROVIDER
dtypes: float64(639), int64(15), object(1738)
memory usage: 780.7 MB
```

The memory usage for the above file is about 780MB which can be considered high. The number of variables used in the analysis will differ from each section. A good practice is to load only needed variables so that memory usage can be reduced. Section with analysis will only load data as required.

Analysis

1. Structural description of the data set

Every data set is unique. So it is necessary to know the basic structure of the dataset i.e. No of data instances, No of variables, type of variables. For the latter part of this section, only needed variables will be loaded into the memory. They are listed and described below.

1. **UNITID** (Integer): This is a unique identification number assigned to postsecondary institutions as surveyed through IPEDS
2. **INSTNM** (String): The institution's name as reported in IPEDS.
3. **STABBR**, (String): The institution’s location using the state abbreviation

a. How many schools and variables?

In [3]:

```
df.shape
```

Out[3]:

```
(6694, 2392)
```

There are 6694 entries and 2392 columns. Thus the number of schools is 6694. And there are 2392 variables.

In [4]:

```
df = pd.read_csv("Most-Recent-Cohorts-All-Data-Elements.csv", low_memory=False, usecols=["UNITI
df.info(memory_usage = "deep")
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 6694 entries, 0 to 6693
Data columns (total 3 columns):
 #   Column  Non-Null Count  Dtype 
---  ------  --------------  ----- 
 0   UNITID  6694 non-null   int64 
 1   INSTNM  6694 non-null   object
 2   STABBR  6694 non-null   object
dtypes: int64(1), object(2)
memory usage: 1008.9 KB
```

b. How many schools are there per state?

The schools per state can be calculated by grouping the data frame using column STABBR and counting the number of rows in that group. The code is as given below.

In [5]:

```
school_per_state = df.groupby(by= ["STABBR"])["UNITID"].agg(["count"])
school_per_state
```

Out[5]:

	count
STABBR	
AK	9
AL	89
AR	92
AS	1
AZ	114
CA	693
CO	101
CT	79
DC	25
DE	21
FL	378
FM	1
GA	175
GU	3

	count
STABBR	
HI	21
IA	80
ID	38
IL	258
IN	139
KS	80
KY	90
LA	121
MA	156
MD	85
ME	38
MH	1
MI	192
MN	112
MO	158
MP	1
MS	59
MT	32
NC	179
ND	26
NE	42
NH	37
NJ	168
NM	49
NV	38
NY	455
OH	289
OK	108
OR	78
PA	353
PR	144
PW	1
RI	23
SC	100
SD	28
TN	155
TX	433
UT	71

	count
STABBR	
VA	168
VI	2
VT	22
WA	105
WI	95
WV	73
WY	10

c. How are schools-per-state distributed? Compute a state-level variable "# of schools" and describe its distribution numerically and visually.

The distribution of the schools-per-state is described numerically as shown and can be visualized in the bar graph and histogram given below.

In [6]:

```
school_per_state = school_per_state.rename(columns={"count": "# of schools"})
school_per_state.describe()
```

Out[6]:

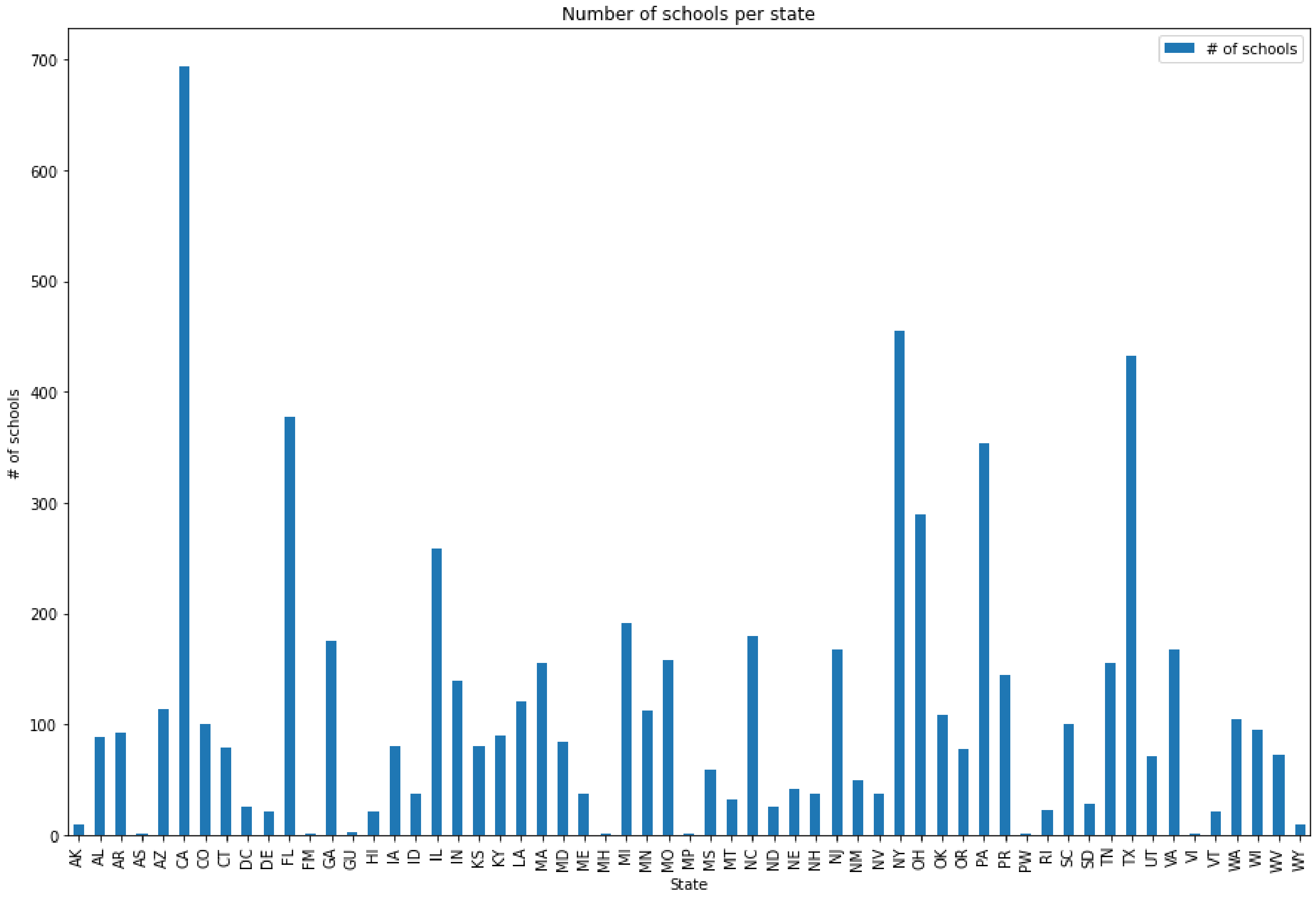
	# of schools
count	59.000000
mean	113.457627
std	129.991156
min	1.000000
25%	27.000000
50%	80.000000
75%	149.500000
max	693.000000

In [7]:

```
school_per_state.plot(kind="bar", figsize=(15,10), xlabel="State", ylabel="# of schools", title
```

Out[7]:

```
<AxesSubplot:title={'center':'Number of schools per state'}, xlabel='State', ylabel='# of schoo
ls'>
```

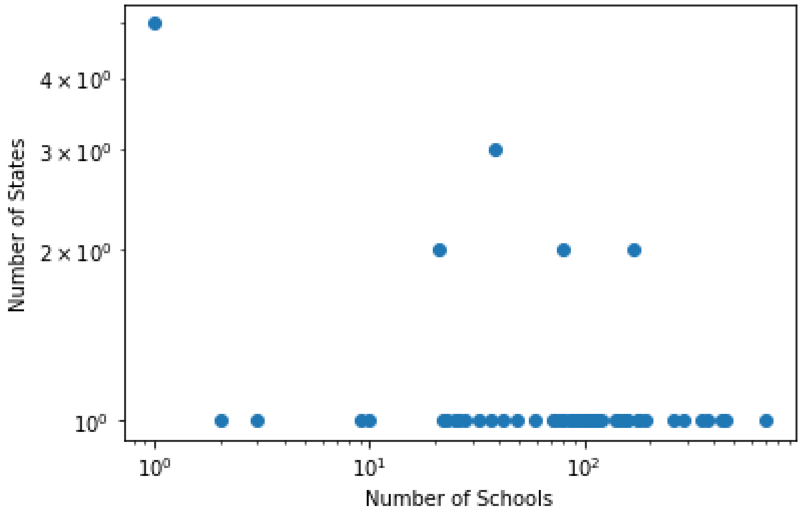


In [8]:

```
hist = school_per_state["# of schools"].value_counts()
plt.scatter(hist.index, hist)
plt.xscale("log")
plt.xlabel("Number of Schools")
plt.yscale("log")
plt.ylabel("Number of States")
```

Out[8]:

```
Text(0, 0.5, 'Number of States')
```

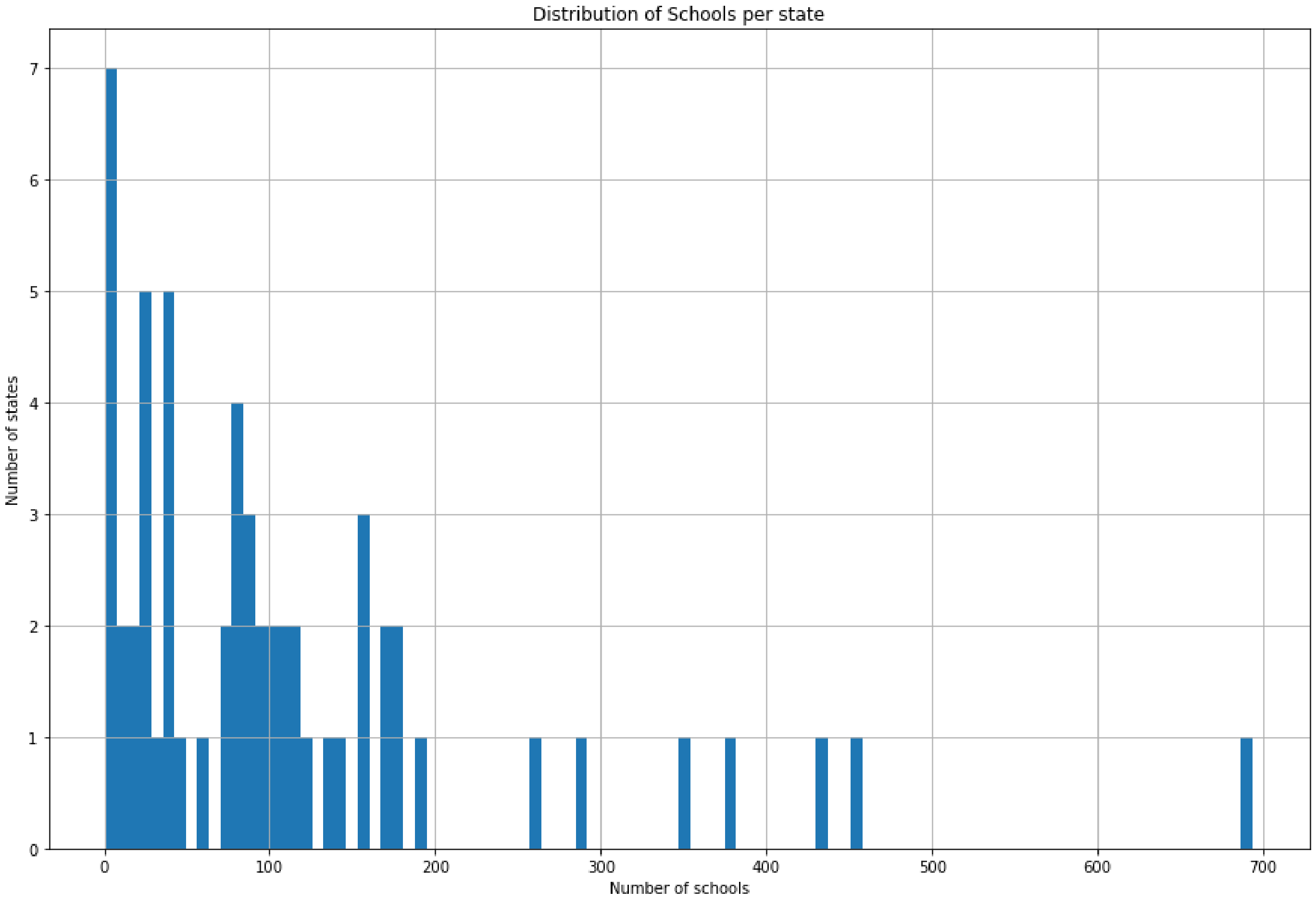


In [9]:

```
school_per_state.hist(figsize=(15,10), bins=100)
plt.title("Distribution of Schools per state")
plt.xlabel("Number of schools")
plt.ylabel("Number of states")
```

Out[9]:

```
Text(0, 0.5, 'Number of states')
```



The average number of school per state is much larger than the median, causing a right-skewed distribution. All states have at least one school, with some states having a large number of schools, causing the mean number of schools to be way higher.

2. Distribution of the overall completion rate

Completion rate denotes the portion of an admitted student graduating from college. Overall value might depend on various factors and can be calculated using various variables listed below. But for this assignment, We are going to choose a suitable variable.

1. **UNITID** (Integer): This is a unique identification number assigned to post-secondary institutions as surveyed through IPEDS
2. **INSTNM** (String): The institution's name as reported in IPEDS.
3. **C150_4** (Float): Completion rate for first-time, full-time students at four-year institutions (150% of expected time to completion)
4. **C150_L4** (Float): Completion rate for first-time, full-time students at less-than-four-year institutions (150% of expected time to completion)

In [10]:

```
df_comp_rate = pd.read_csv("Most-Recent-Cohorts-All-Data-Elements.csv", low_memory=False, usec
df_comp_rate["CMP_RATE"] = df_comp_rate["C150_4"].combine_first(df_comp_rate["C150_L4"])
```

a. Provide choice of completion rate variable with a justification for that choice.

The datasheet contains variables like **C100_4_POOLED**, **C100_L4_POOLED**, **C150_4_POOLED**, **C150_L4_POOLED**, **C200_4** and **C200_L4**. These completion rates are divided into Pooled and Un-pooled. Choosing Pooled one would be better because they were across two years on a rolling basis to reduce variability. But later part requires us to describe the completion rate based on race. Only **C150_4** (Un-Pooled) and **C150_L4** (Un-Pooled) are broken down into racial categories. So, to maintain uniformity, I chose **C150_4** and **C150_L4**. Looking at the data, schools either provide four years or less than four years programs. Thus, we can combine these two variables into **CMP_RATE**, denoting the overall completion rate.

b. Describe the distribution of that variable numerically and visually.

In [11]:

```
df_comp_rate["CMP_RATE"].describe()
```

Out[11]:

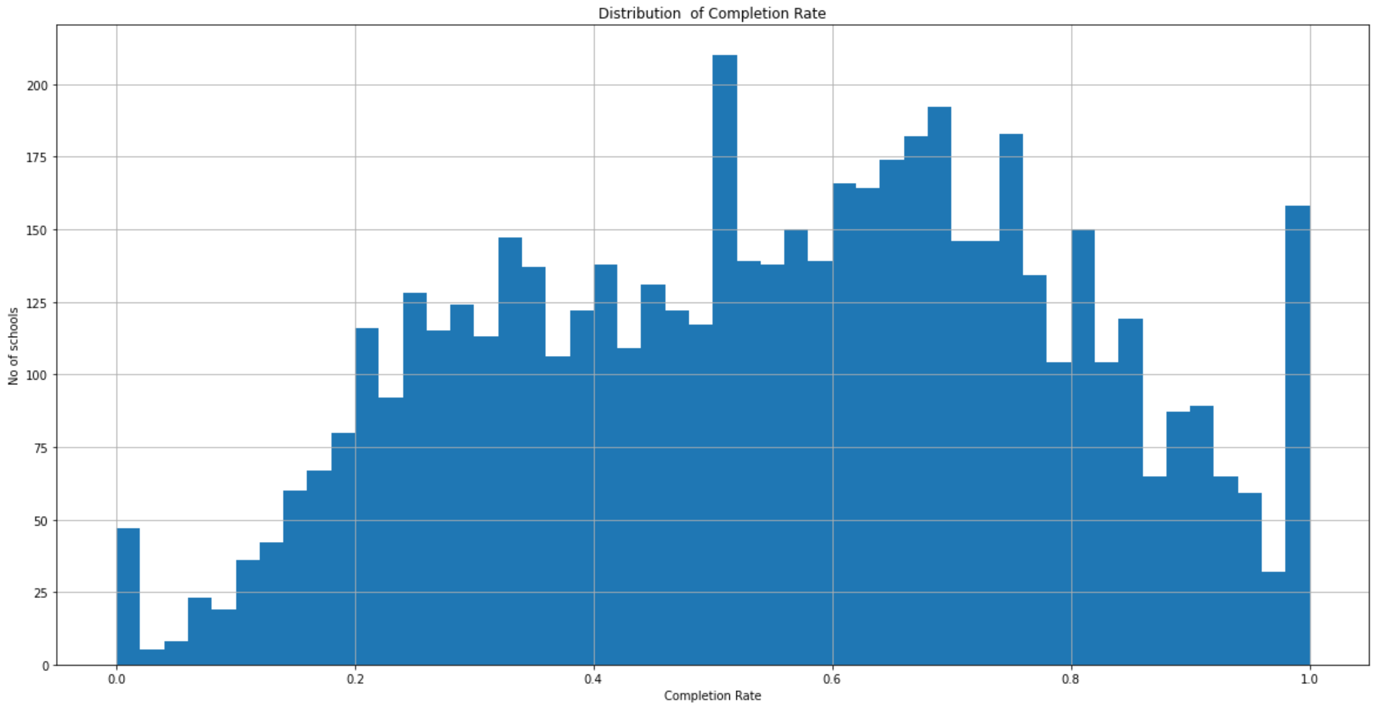
```
count    5499.000000
mean        0.553665
std         0.236376
min         0.000000
25%         0.363800
50%         0.568100
75%         0.735800
max         1.000000
Name: CMP_RATE, dtype: float64
```

In [12]:

```
df_comp_rate["CMP_RATE"].hist(bins=50, figsize=(20,10))
plt.title("Distribution  of Completion Rate")
plt.xlabel("Completion Rate")
plt.ylabel("No of schools")
```

Out[12]:

```
Text(0, 0.5, 'No of schools')
```



c. What is the mean? Is the distribution skewed?

The mean Completion Rate is 0.55. The distribution for Completion Rate is slightly left-skewed with a mean less than the median of 0.57. The reason for this slight skew might be due to a few schools with a completion rate of one or near one value.

3. Distribution of Admission Rate

The distribution of the admission rate, both numerically and graphically. After describing the continuous admission rate distribution, compute the admissions category (open, low-selectivity, or high-selectivity). Do not hard-code the median — compute the median, and use the calculated value (stored in a Python variable) to bucketize the admission rates. Show the distribution of admissions category (how many schools are in each class?).

Admission rate is the number of admitted undergraduates divided by the number of undergraduates who applied. For this section, only a few variables need to be loaded into the memory. They are listed and described below.

1. **UNITID** (Integer): This is a unique identification number assigned to post-secondary institutions as surveyed through IPEDS
2. **INSTNM** (String): The institution's name as reported in IPEDS.
3. **OPENADMP** (Integer): Is the admission policy Open or Not
4. **ADM_RATE** (Float): Admissions rate at each campus

In [13]:

```
df_adm_rate = pd.read_csv("Most-Recent-Cohorts-All-Data-Elements.csv", low_memory=False, usecol
df_adm_rate["ADM_RATE"].describe()
```

Out[13]:

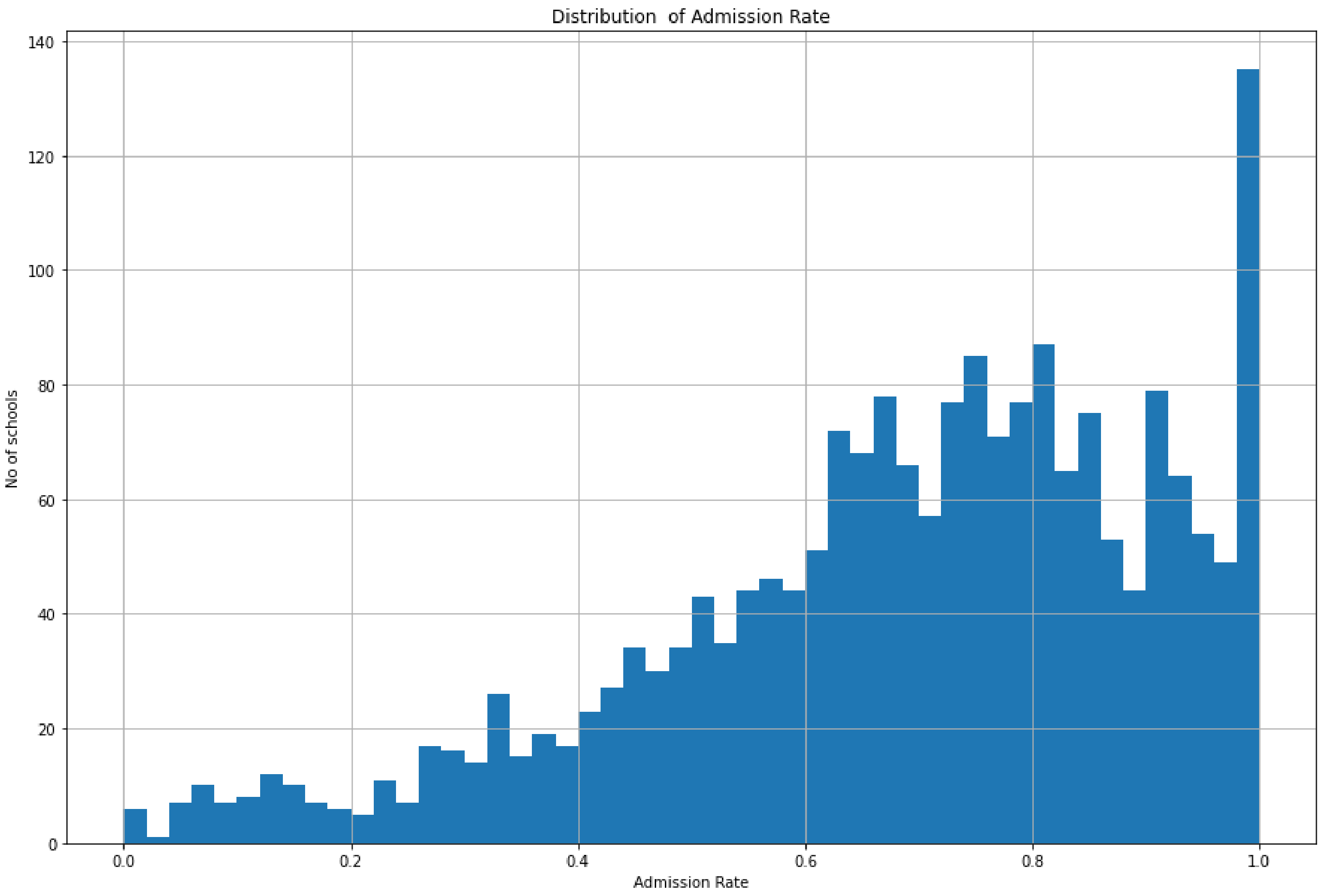
```
count    1988.000000
mean        0.689655
std         0.220990
min         0.000000
25%         0.562725
50%         0.726700
75%         0.852550
max         1.000000
Name: ADM_RATE, dtype: float64
```

In [14]:

```
df_adm_rate["ADM_RATE"].hist(figsize=(15,10), bins=50)
plt.title("Distribution  of Admission Rate")
plt.xlabel("Admission Rate")
plt.ylabel("No of schools")
```

Out[14]:

```
Text(0, 0.5, 'No of schools')
```



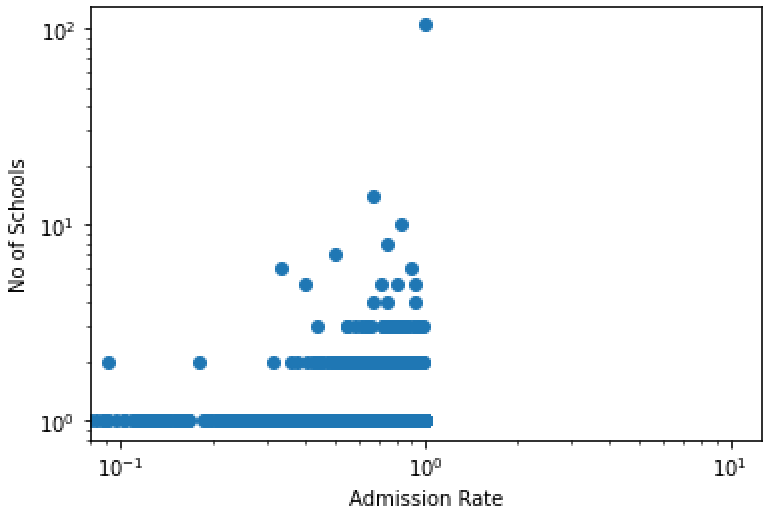
The distribution is left skewed. A lot of schools has admission rate closer to 1.

In [15]:

```
hist = df_adm_rate["ADM_RATE"].value_counts()
plt.scatter(hist.index, hist)
plt.xscale("log")
plt.xlabel("Admission Rate")
plt.yscale("log")
plt.ylabel("No of Schools")
```

```
Text(0, 0.5, 'No of Schools')
```

Out[15]:



In [16]:

```
adm_rate_med = df_adm_rate["ADM_RATE"].median()
adm_selectivity = pd.Series("No Data", index=df_adm_rate.index)
adm_selectivity[df_adm_rate.OPENADMP == 1] = "Open-admission"
adm_selectivity[(df_adm_rate.OPENADMP == 2) & (df_adm_rate.ADM_RATE > adm_rate_med)] = "Low-sel
adm_selectivity[(df_adm_rate.OPENADMP == 2) & (df_adm_rate.ADM_RATE < adm_rate_med)] = "High-se
adm_selectivity = adm_selectivity.astype("category")
df_adm_rate["ADMISSION_SELECTIVITY"] = adm_selectivity
df_adm_rate["ADMISSION_SELECTIVITY"].value_counts()
```

Out[16]:

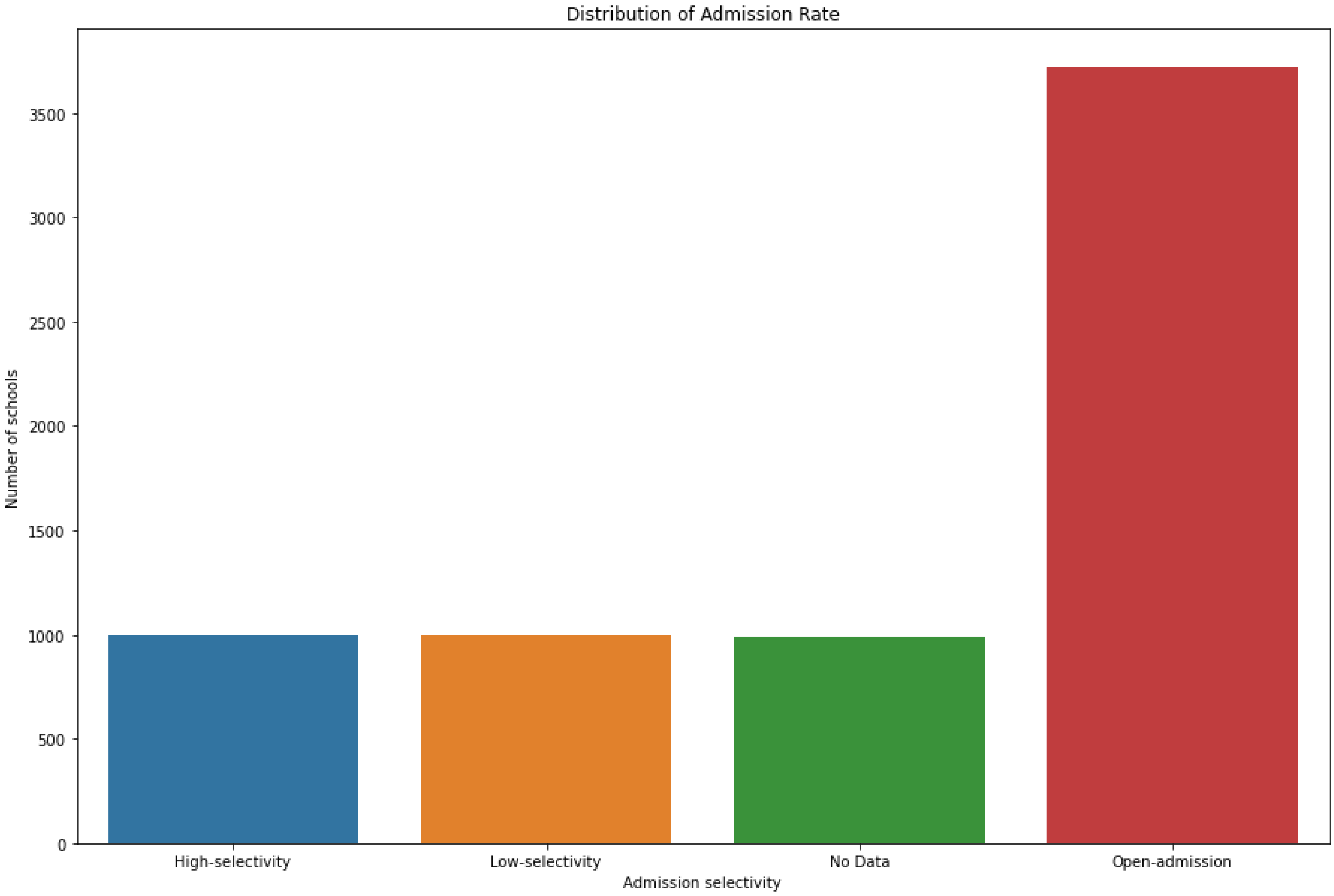
```
Open-admission      3719
High-selectivity     994
Low-selectivity      994
No Data              987
Name: ADMISSION_SELECTIVITY, dtype: int64
```

In [17]:

```
plt.figure(figsize=(15,10))
sns.countplot(x=df_adm_rate["ADMISSION_SELECTIVITY"])
plt.title("Distribution of Admission Rate")
plt.xlabel("Admission selectivity")
plt.ylabel("Number of schools")
```

Out[17]:

```
Text(0, 0.5, 'Number of schools')
```



Admission selectivity denotes the overall admission policy of the schools. Most of them are Open-admission types. The numbers of schools for Low-selectivity and High-selectivity are the same because we used the median to separate them.

4. Disaggregation of completion rate by race and some other characteristics

The breakdown (sometimes called a disaggregation) of completion rate by race, by the school characteristics described in "Question," and by one additional school characteristic, you select (30%). Give a justification for your choice of characteristic — why do you think it might be interesting? You need to show these breakdowns both numerically and graphically. Box plots are useful for this, as are bar charts.

- Race is a per-student characteristic; schools report completion rate separately for each racial category, in addition to the overall completion rate. The resulting chart should have one bar or box for each racial group.
- The other characteristics — selectivity, public/private status, and your chosen additional one — are per-school statistics. The resulting chart should have one box or bar for each value of the selected characteristic (e.g., for selectivity, these are open, low, and high). Describe differences you see, with references to specific features in the charts. What kinds of schools seem to be doing the best in terms of getting students to completion? The reason for choosing 150 for the completion rate is that only those are broken down by race. We use the same logic as before combining _*4 and* _L4 variables to a single completion rate.

For this section, only a few variables need to be loaded into the memory. They are listed and described below.

1. **UNITID** (Integer): This is a unique identification number assigned to post-secondary institutions as surveyed through IPEDS
2. **INSTNM** (String): The institution's name as reported in IPEDS.
3. **CONTROL** (Integer): Control of the institution
4. **LOCALE** (Float): Locality of the school

5. **C150_4** (Float): Completion rate for first-time, full-time students at four-year institutions (150% of the expected time to completion)
6. **C150_L4** (Float): Completion rate for first-time, full-time students at less-than-four-year institutions (150% of the expected time to completion)
7. **C150_4_WHITE** (Float): Completion rate for first-time, full-time students at four-year institutions (150% of the expected time to completion) for White students
8. **C150_4_BLACK** (Float): Completion rate for first-time, full-time students at four-year institutions (150% of the expected time to completion) for Black students
9. **C150_4_HISP** (Float): Completion rate for first-time, full-time students at four-year institutions (150% of the expected time to completion) for Hispanic students
10. **C150_4_ASIAN** (Float): Completion rate for first-time, full-time students at four-year institutions (150% of the expected time to completion) for Asian students
11. **C150_4_AIAN** (Float): Completion rate for first-time, full-time students at four-year institutions (150% of the expected time to completion) for American Indian/Alaska Native students
12. **C150_4_NHPI** (Float): Completion rate for first-time, full-time students at four-year institutions (150% of the expected time to completion) for Native Hawaiian/Pacific Islander students
13. **C150_4_2MOR** (Float): Completion rate for first-time, full-time students at four-year institutions (150% of expected time to completion) for students of two-or-more-races
14. **C150_4_NRA** (Float): Completion rate for first-time, full-time students at four-year institutions (150% of expected time to completion) for non-resident alien students
15. **C150_4_UNKN** (Float): Completion rate for first-time, full-time students at four-year institutions (150% of the expected time to completion) for students whose race is unknown
16. **C150_4_WHITENH** (Float): Completion rate for first-time, full-time students at four-year institutions (150% of the expected time to completion) for White non-Hispanic students
17. **C150_4_BLACKNH** (Float): Completion rate for first-time, full-time students at four-year institutions (150% of the expected time to completion) for Black non-Hispanic students
18. **C150_4_API** (Float): Completion rate for first-time, full-time students at four-year institutions (150% of expected time to completion) for Asian/Pacific Islander students
19. **C150_4_AIANOLD** (Float): Completion rate for first-time, full-time students at four-year institutions (150% of the expected time to completion) for American Indian/Alaska Native students
20. **C150_4_HISPOLD** (Float): Completion rate for first-time, full-time students at four-year institutions (150% of expected time to completion) for Hispanic students
21. **C150_L4_WHITE** (Float): Completion rate for first-time, full-time students at less-than-four-year institutions (150% of the expected time to completion) for White non-Hispanic students
22. **C150_L4_BLACK** (Float): Completion rate for first-time, full-time students at less-than-four-year institutions (150% of the expected time to completion) for Black non-Hispanic students
23. **C150_L4_HISP** (Float): Completion rate for first-time, full-time students at less-than-four-year institutions (150% of the expected time to completion) for Hispanic students
24. **C150_L4_ASIAN** (Float): Completion rate for first-time, full-time students at less-than-four-year institutions (150% of the expected time to completion) for Asian students
25. **C150_L4_AIAN** (Float): Completion rate for first-time, full-time students at less-than-four-year institutions (150% of the expected time to completion) for American Indian/Alaska Native students
26. **C150_L4_NHPI** (Float): Completion rate for first-time, full-time students at less-than-four-year institutions (150% of the expected time to completion) for Native Hawaiian/Pacific Islander students
27. **C150_L4_2MOR** (Float): Completion rate for first-time, full-time students at less-than-four-year institutions (150% of expected time to completion) for students of two-or-more-races
28. **C150_L4_NRA** (Float): Completion rate for first-time, full-time students at less-than-four-year institutions (150% of expected time to completion) for non-resident alien students
29. **C150_L4_UNKN** (Float): Completion rate for first-time, full-time students at less-than-four-year institutions (150% of the expected time to completion) for students whose race is unknown

30. **C150_L4_WHITENH** (Float): Completion rate for first-time, full-time students at less-than-four-year institutions (150% of the expected time to completion) for white non-Hispanic students
31. **C150_L4_BLACKNH** (Float): Completion rate for first-time, full-time students at less-than-four-year institutions (150% of the expected time to completion) for black non-Hispanic students
32. **C150_L4_API** (Float): Completion rate for first-time, full-time students at less-than-four-year institutions (150% of the expected time to completion) for Asian/Pacific Islander students
33. **C150_L4_AIANOLD** (Float): Completion rate for first-time, full-time students at less-than-four-year institutions (150% of the expected time to completion) for American Indian/Alaska Native students
34. **C150_L4_HISPOLD** (Float): Completion rate for first-time, full-time students at less-than-four-year institutions (150% of the expected time to completion) for Hispanic students

In [18]:

```python
usable_columns = ["UNITID", "INSTNM", "CONTROL","LOCALE", "C150_4", "C150_L4", "C150_4_WHITE",
df_char = pd.read_csv("Most-Recent-Cohorts-All-Data-Elements.csv", low_memory=False, usecols=us
combine_map = {
    "CMP_RATE": ["C150_4", "C150_L4"],
    "WHITE": ["C150_4_WHITE", "C150_L4_WHITE"],
    "BLACK": ["C150_4_BLACK", "C150_L4_BLACK"],
    "HISPANIC": ["C150_4_HISP", "C150_L4_HISP"],
    "ASIAN": ["C150_4_ASIAN", "C150_L4_ASIAN"],
    "AMERICAN INDIAN/ALASKA NATIVE": ["C150_4_AIAN", "C150_L4_AIAN"],
    "NATIVE HAWAIIAN/PACIFIC ISLANDER": ["C150_4_NHPI", "C150_L4_NHPI"],
    "TWO OR MORE RACES": ["C150_4_2MOR", "C150_L4_2MOR"],
    "NON RESIDENT ALIEN": ["C150_4_NRA", "C150_L4_NRA"],
    "UNKNOWN": ["C150_4_UNKN", "C150_L4_UNKN"],
    "WHITE NON HISPANIC": ["C150_4_WHITENH", "C150_L4_WHITENH"],
    "BLACK NON HISPANIC": ["C150_4_BLACKNH", "C150_L4_BLACKNH"],
    "ASIAN/PACIFIC ISLANDER": ["C150_4_API", "C150_L4_API"],
    "OLD AMERICAN INDIAN/ALASKA NATIVE": ["C150_4_AIANOLD", "C150_L4_AIANOLD"],
    "OLD HISPANIC": ["C150_4_HISPOLD", "C150_L4_HISPOLD"],
}
for key, value in combine_map.items():
    df_char[key] = df_char[value[0]].combine_first(df_char[value[1]])
df_char.drop(columns=usable_columns[4:], inplace=True)
df_char["ADMISSION_SELECTIVITY"] = adm_selectivity
df_char["CONTROL_NAME"] = pd.Categorical(df_char["CONTROL"]).rename_categories({1: "Public", 2:


df_by_race = df_char.melt(id_vars=["UNITID"], value_vars=list(combine_map.keys())[1:], var_name
df_by_race.groupby("RACE")["CMP_RATE_RACE"].describe()
```

Out[18]:

	count	mean	std	min	25%	50%	75%	max
RACE								
AMERICAN INDIAN/ALASKA NATIVE	2314.0	0.373609	0.363793	0.0	0.00000	0.33330	0.666700	1.0
ASIAN	2858.0	0.548124	0.316299	0.0	0.33330	0.55560	0.800000	1.0
ASIAN/PACIFIC ISLANDER	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN
BLACK	3420.0	0.384992	0.270301	0.0	0.16670	0.34100	0.559475	1.0
BLACK NON HISPANIC	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN
HISPANIC	3476.0	0.462634	0.270360	0.0	0.25000	0.44560	0.666700	1.0
NATIVE HAWAIIAN/PACIFIC ISLANDER	1517.0	0.447001	0.410721	0.0	0.00000	0.40000	1.000000	1.0
NON RESIDENT ALIEN	2156.0	0.529945	0.312375	0.0	0.30430	0.54550	0.775550	1.0
OLD AMERICAN INDIAN/ALASKA NATIVE	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN
OLD HISPANIC	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN
TWO OR MORE RACES	2890.0	0.436426	0.291992	0.0	0.21315	0.40935	0.650000	1.0
UNKNOWN	2673.0	0.458807	0.307632	0.0	0.20000	0.46430	0.666700	1.0

	count	mean	std	min	25%	50%	75%	max
RACE								
WHITE	3708.0	0.521261	0.238871	0.0	0.33975	0.51850	0.699525	1.0
WHITE NON HISPANIC	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN

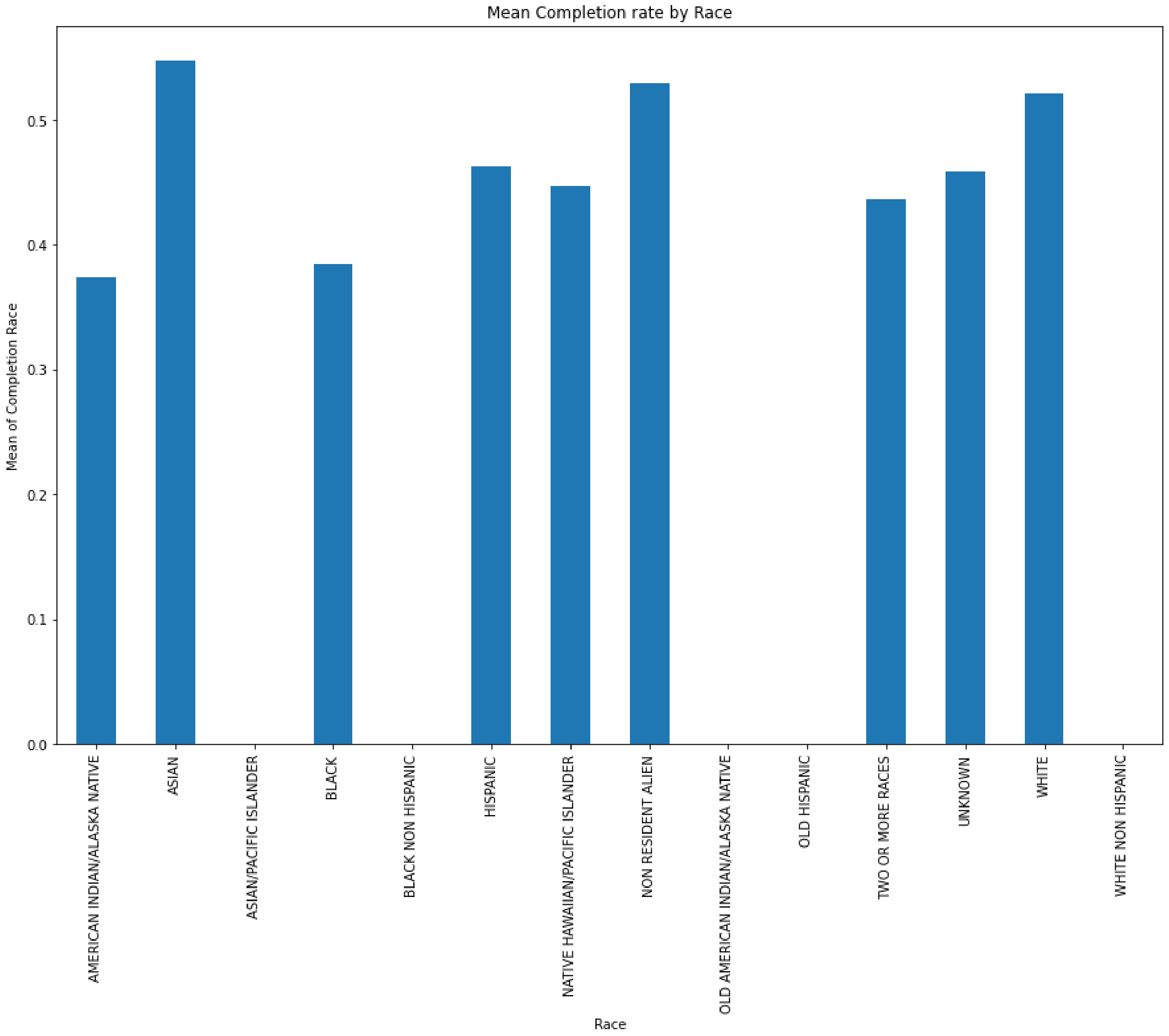
In [19]:

```
df_by_race.groupby("RACE")["CMP_RATE_RACE"].mean().plot(
    kind="bar",
    figsize=(15,10),
    title="Mean Completion rate by Race",
    xlabel="Race",
    ylabel="Mean of Completion Race"

)
```

Out[19]:

```
<AxesSubplot:title={'center':'Mean Completion rate by Race'}, xlabel='Race', ylabel='Mean of Co
mpletion Race'>
```

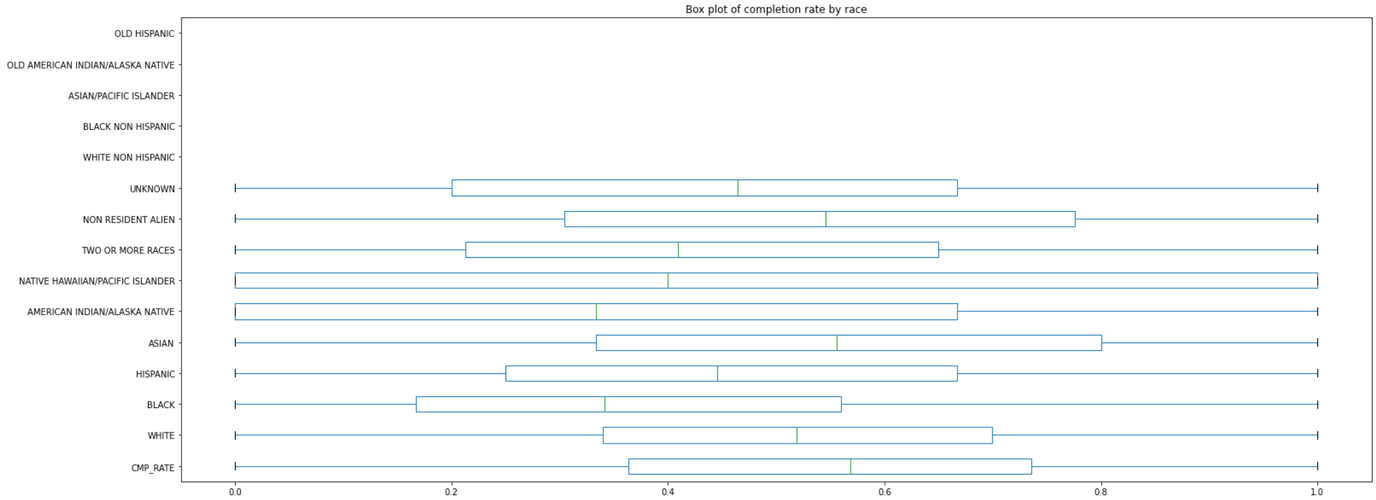


In [20]:

```
df_char[list(combine_map.keys())].plot(kind="box",figsize=(25,10), title="Box plot of completic
```

Out[20]:

```
<AxesSubplot:title={'center':'Box plot of completion rate by race'}>
```



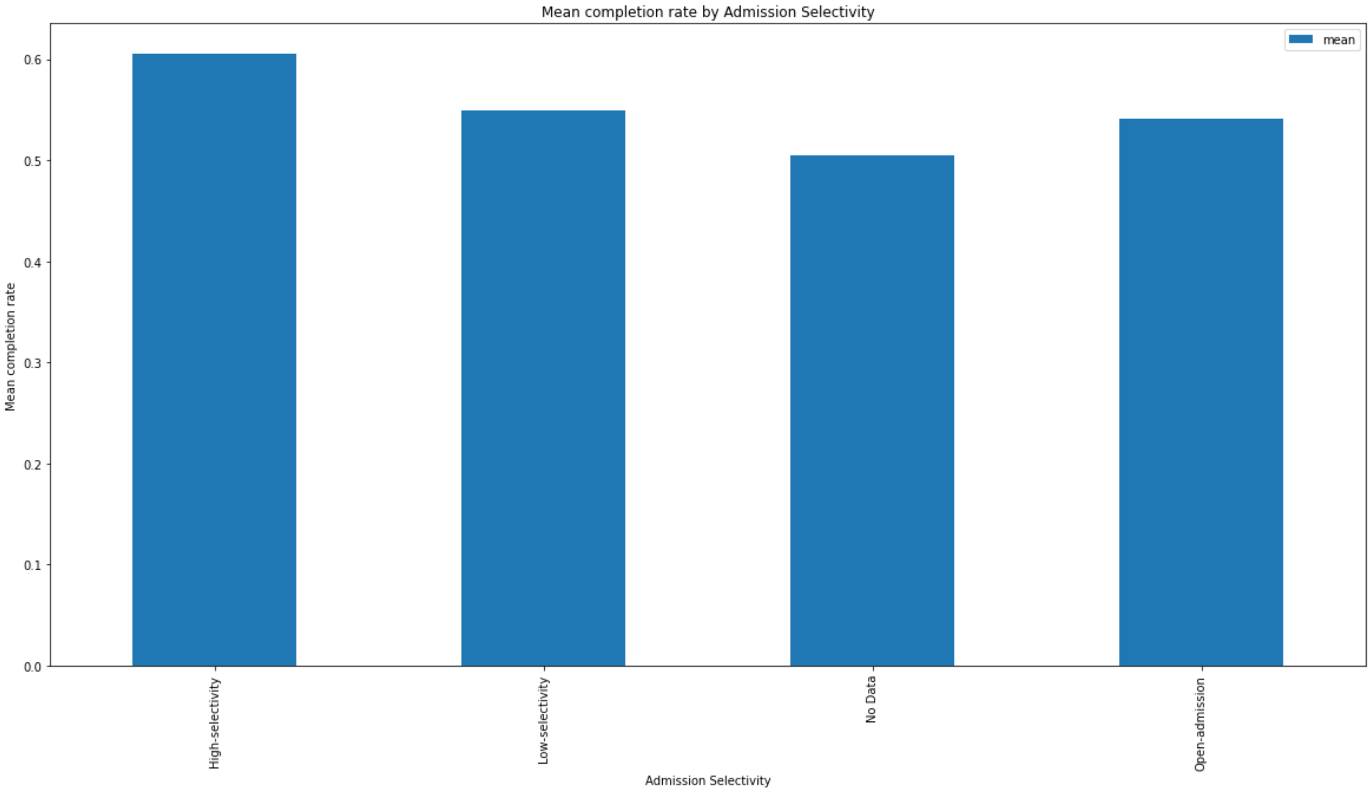
We can see that the completion rate is not even. Asian has the highest completion rate, followed by Non-Resident Alien. American Indian/Alaska Native has the lowest. Native Hawaiian/Pacific Islander has the same InterQuartile Range and the Rance, meaning it has a lot of data with Completion rate 0 and 1 with some data in between them.

In [21]:

```
df_char.groupby(by=["ADMISSION_SELECTIVITY"])["CMP_RATE"].agg(["mean"]).plot(
    kind="bar",
    figsize=(20,10),
    xlabel="Admission Selectivity",
    ylabel="Mean completion rate",
    title="Mean completion rate by Admission Selectivity",
)
```

Out[21]:

```
<AxesSubplot:title={'center':'Mean completion rate by Admission Selectivity'}, xlabel='Admissio
n Selectivity', ylabel='Mean completion rate'>
```

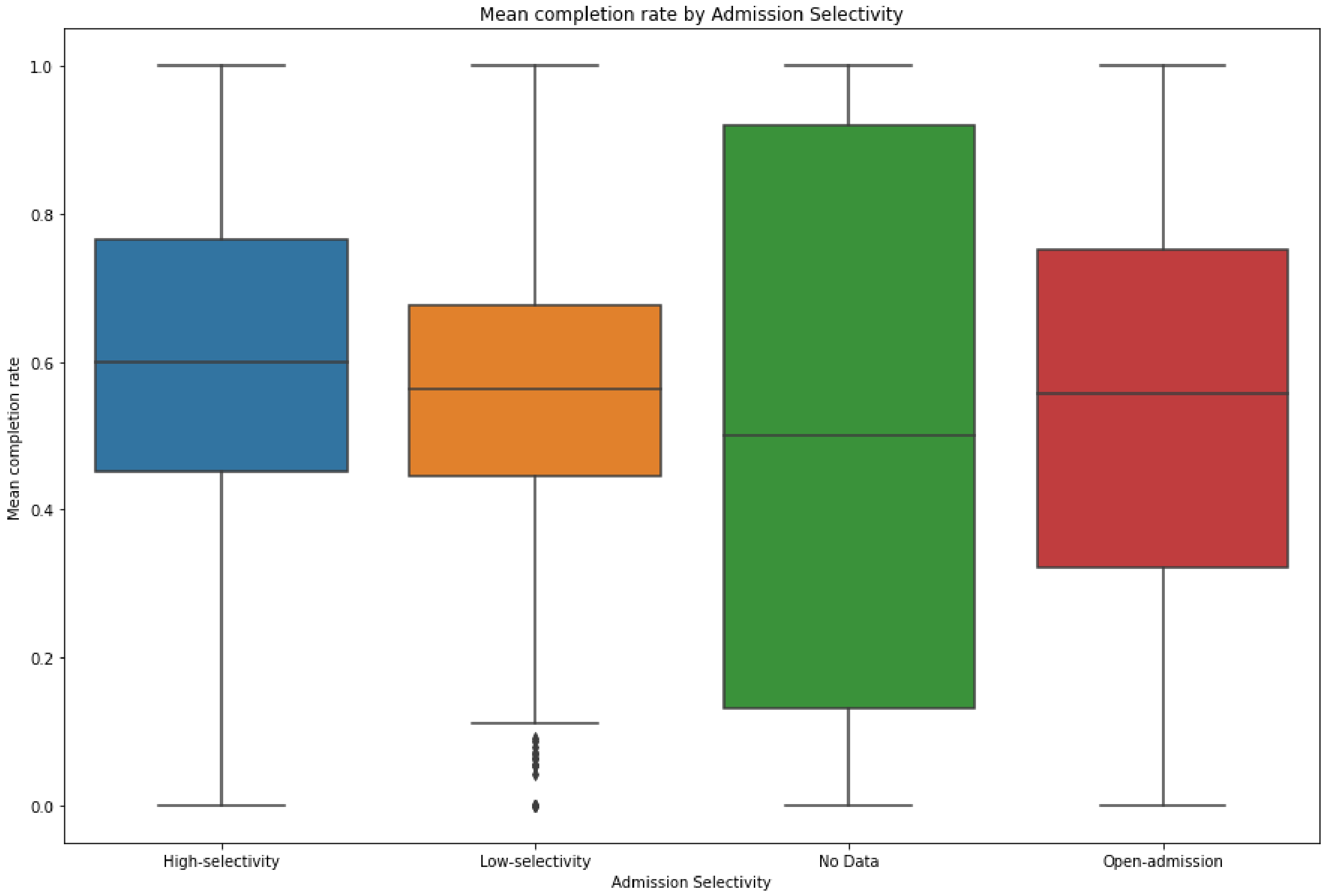


In [22]:

```
plt.figure(figsize=(15,10))
sns.boxplot(x="ADMISSION_SELECTIVITY", y="CMP_RATE", data=df_char)
plt.xlabel("Admission Selectivity")
plt.ylabel("Mean completion rate")
plt.title("Mean completion rate by Admission Selectivity")
```

Out[22]:

```
Text(0.5, 1.0, 'Mean completion rate by Admission Selectivity')
```



In [23]:

```
df_char.groupby(by=["ADMISSION_SELECTIVITY"])["CMP_RATE"].describe()
```

Out[23]:

ADMISSION_SELECTIVITY	count	mean	std	min	25%	50%	75%	max
High-selectivity	962.0	0.605136	0.209941	0.0	0.451025	0.59975	0.765650	1.0
Low-selectivity	952.0	0.549796	0.190944	0.0	0.445175	0.56275	0.675700	1.0
No Data	34.0	0.505174	0.373575	0.0	0.131625	0.50000	0.918725	1.0
Open-admission	3551.0	0.541223	0.250277	0.0	0.322100	0.55710	0.750000	1.0

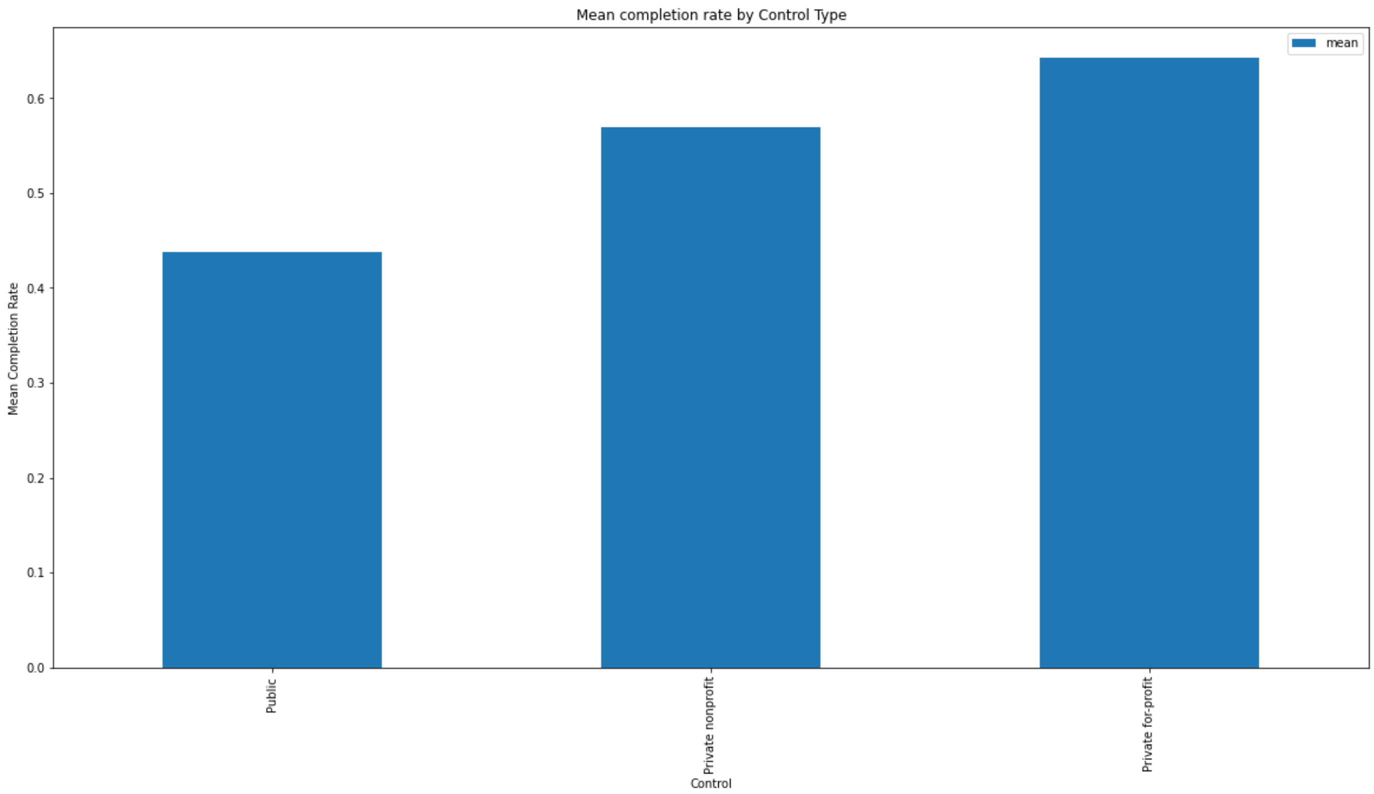
The above graphs show the completion rate based on Admission Policy. The Mean Completion rate for Highly selective Schools is higher than the others. (High > Low > Open). More selective the schools are better is the average completion rate.

In [24]:

```
df_char.groupby(by=["CONTROL_NAME"])["CMP_RATE"].agg(["mean"]).plot(
    kind="bar",
    figsize=(20,10),
    xlabel="Control",
    ylabel="Mean Completion Rate",
    title="Mean completion rate by Control Type"
)
```

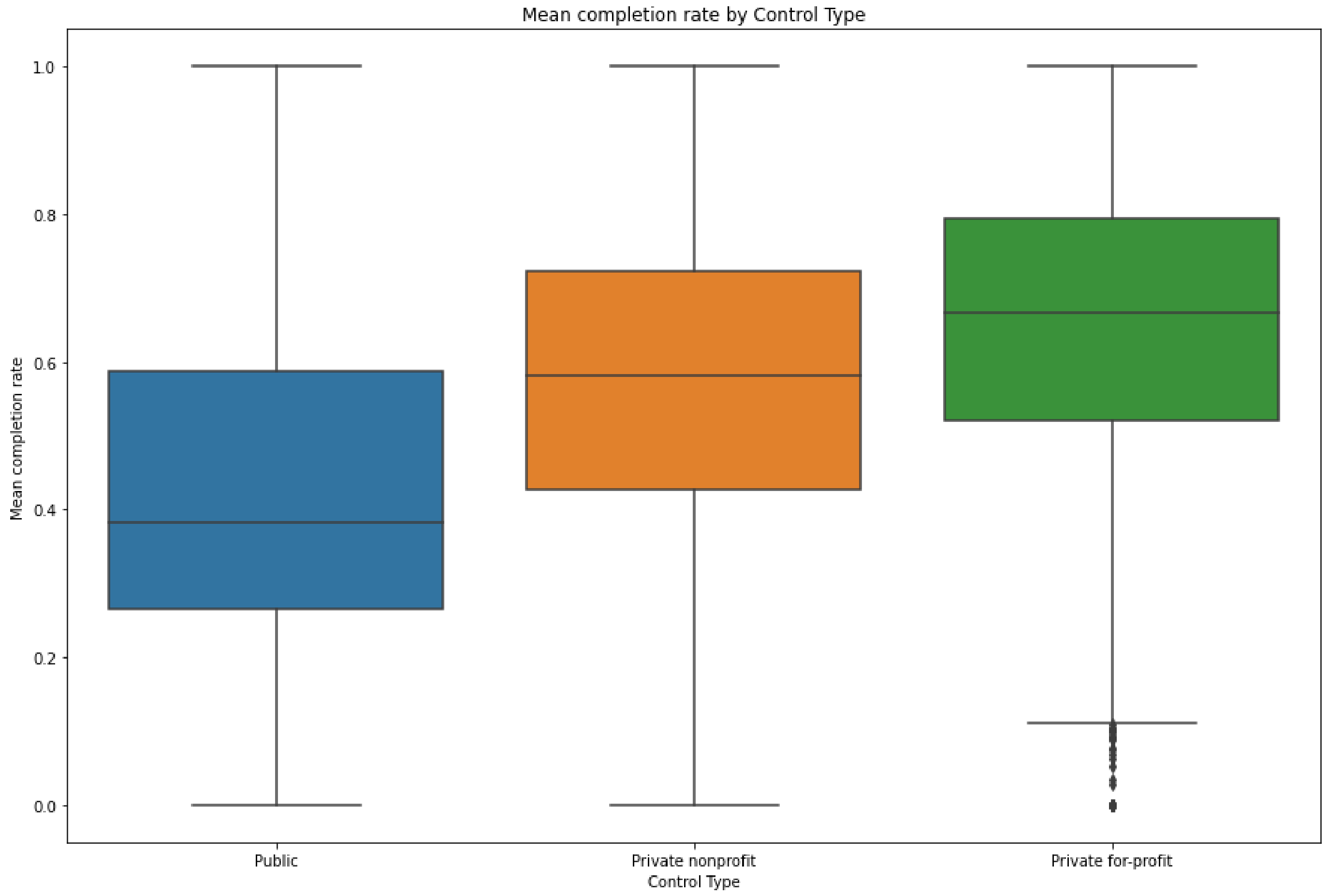
Out[24]:

```
<AxesSubplot:title={'center':'Mean completion rate by Control Type'}, xlabel='Control', ylabel
='Mean Completion Rate'>
```



In [25]:
```
plt.figure(figsize=(15,10))
sns.boxplot(x="CONTROL_NAME", y="CMP_RATE", data=df_char)
plt.xlabel("Control Type")
plt.ylabel("Mean completion rate")
plt.title("Mean completion rate by Control Type")
```

Out[25]:
```
Text(0.5, 1.0, 'Mean completion rate by Control Type')
```



In [26]:
```
df_char.groupby(by=["CONTROL_NAME"])["CMP_RATE"].describe()
```

Out[26]:

	count	mean	std	min	25%	50%	75%	max

CONTROL_NAME	count	mean	std	min	25%	50%	75%	max
CONTROL_NAME								
Public	1874.0	0.437789	0.220612	0.0	0.265525	0.38335	0.587025	1.0
Private nonprofit	1458.0	0.569826	0.224111	0.0	0.427825	0.58060	0.722500	1.0
Private for-profit	2167.0	0.643000	0.214887	0.0	0.520000	0.66670	0.793100	1.0

The above graphs show the completion rate based on School Control Type. The Mean Completion rate for Private for-profit schools is higher than the others. (Private Profit > Private Nonprofit > Public). More the schools are oriented toward profit are better is the average completion rate.

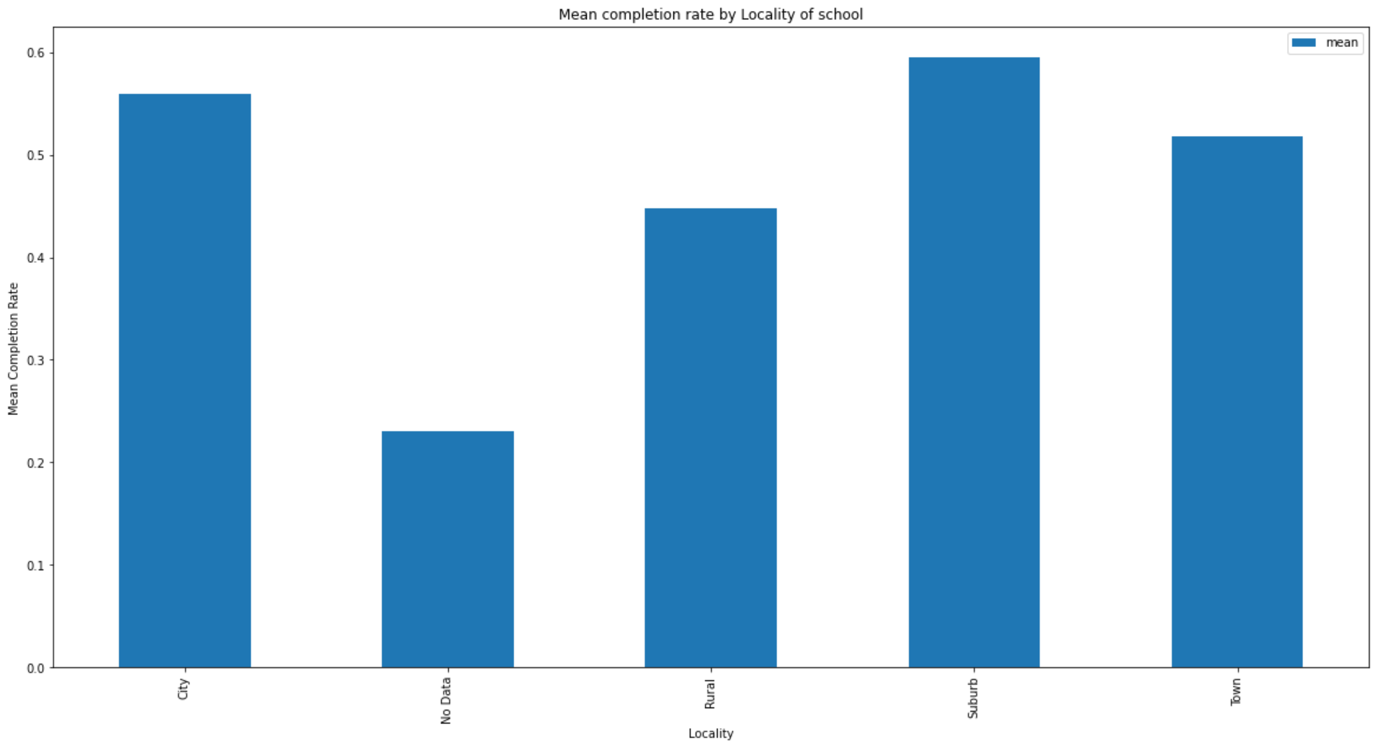
For my variable of choice, I decided to choose the locality of the school/college, i.e., City, Suburb, Town, or Rural. I think the locality of the school plays a critical role in the completion rate. Locality determines the Living Cost, Opportunities for jobs and various other factors that directly or indirectly impact the completion rate.

In [27]:

```
df_char["LOCATION_TYPE"] = pd.Categorical(df_char["LOCALE"]).rename_categories({
    11: "City:Large",
    12: "City:Midsize",
    13: "City:Small",
    21: "Suburb:Large",
    22: "Suburb:Midsize",
    23: "Suburb:Small",
    31: "Town:Large",
    32: "Town:Midsize",
    33: "Town:Small",
    41: "Rural:Large",
    42: "Rural:Midsize",
    43: "Rural:Small",
    -3: "No Data"
})
loc_type = pd.Series("No Data", index=df_char.index)
loc_type[(df_char.LOCALE == 11) | (df_char.LOCALE == 12) | (df_char.LOCALE == 13)] = "City"
loc_type[(df_char.LOCALE == 21) | (df_char.LOCALE == 22) | (df_char.LOCALE == 23)] = "Suburb"
loc_type[(df_char.LOCALE == 31) | (df_char.LOCALE == 32) | (df_char.LOCALE == 33)] = "Town"
loc_type[(df_char.LOCALE == 41) | (df_char.LOCALE == 42) | (df_char.LOCALE == 43)] = "Rural"
loc_type = loc_type.astype("category")
df_char["LOCALITY"] = loc_type
df_char.groupby(by=["LOCALITY"])["CMP_RATE"].agg(["mean"]).plot(
    kind="bar",
    figsize=(20,10),
    xlabel="Locality",
    ylabel="Mean Completion Rate",
    title="Mean completion rate by Locality of school"
)
```

Out[27]:

```
<AxesSubplot:title={'center':'Mean completion rate by Locality of school'}, xlabel='Locality',
ylabel='Mean Completion Rate'>
```

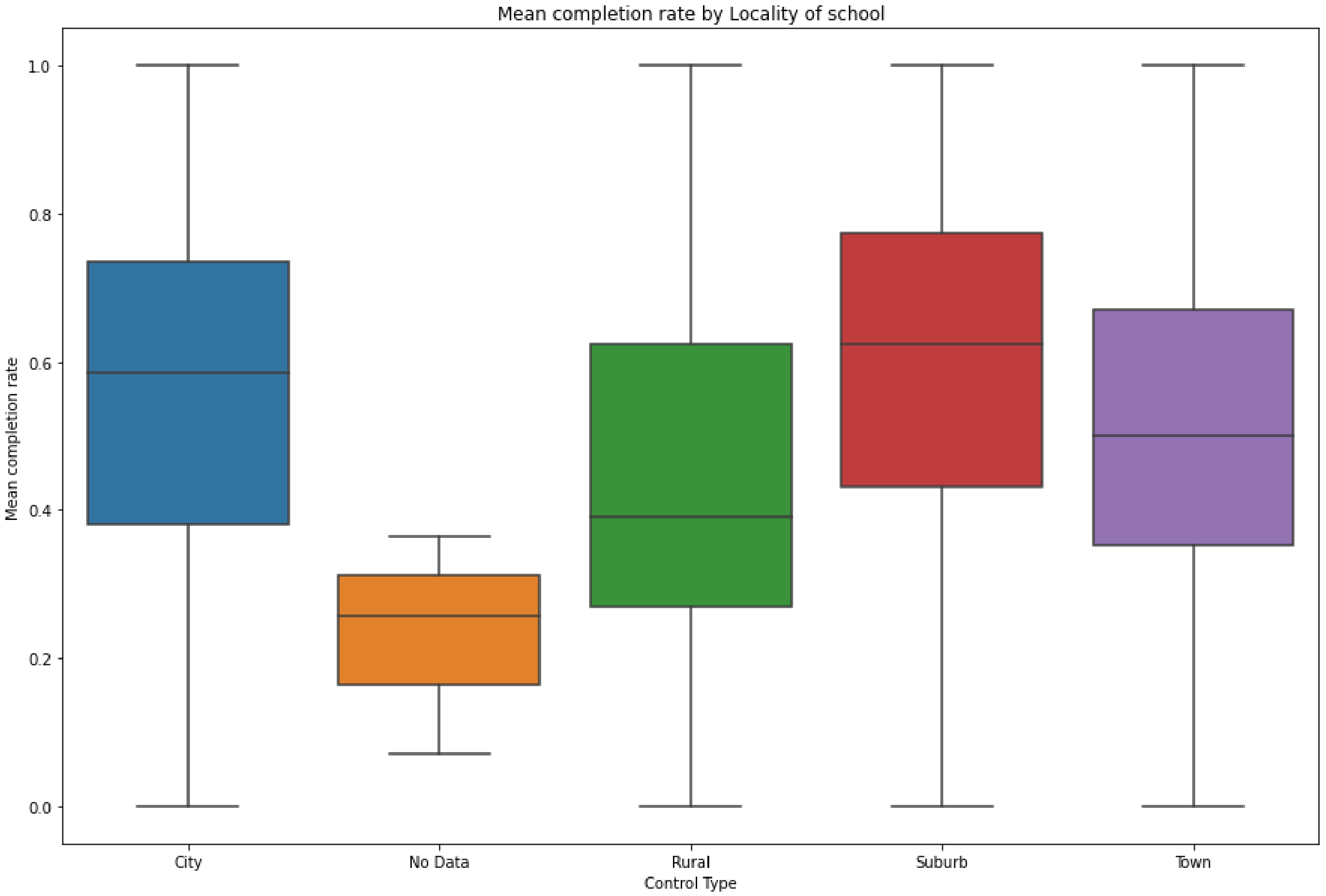


In [28]:

```python
plt.figure(figsize=(15,10))
sns.boxplot(x="LOCALITY", y="CMP_RATE", data=df_char)
plt.xlabel("Control Type")
plt.ylabel("Mean completion rate")
plt.title("Mean completion rate by Locality of school")
```

Out[28]:

```
Text(0.5, 1.0, 'Mean completion rate by Locality of school')
```



In [29]:

```python
df_char.groupby(by=["LOCALITY"])["CMP_RATE"].describe()
```

Out[29]:

	count	mean	std	min	25%	50%	75%	max
LOCALITY								

	count	mean	std	min	25%	50%	75%	max
LOCALITY								
City	2565.0	0.558878	0.235629	0.0000	0.379600	0.5844	0.735700	1.0000
No Data	3.0	0.230767	0.149002	0.0705	0.163600	0.2567	0.310900	0.3651
Rural	482.0	0.448279	0.239025	0.0000	0.268675	0.3903	0.624275	1.0000
Suburb	1621.0	0.595436	0.236065	0.0000	0.431800	0.6225	0.772700	1.0000
Town	828.0	0.518259	0.212753	0.0000	0.352475	0.5000	0.670125	1.0000

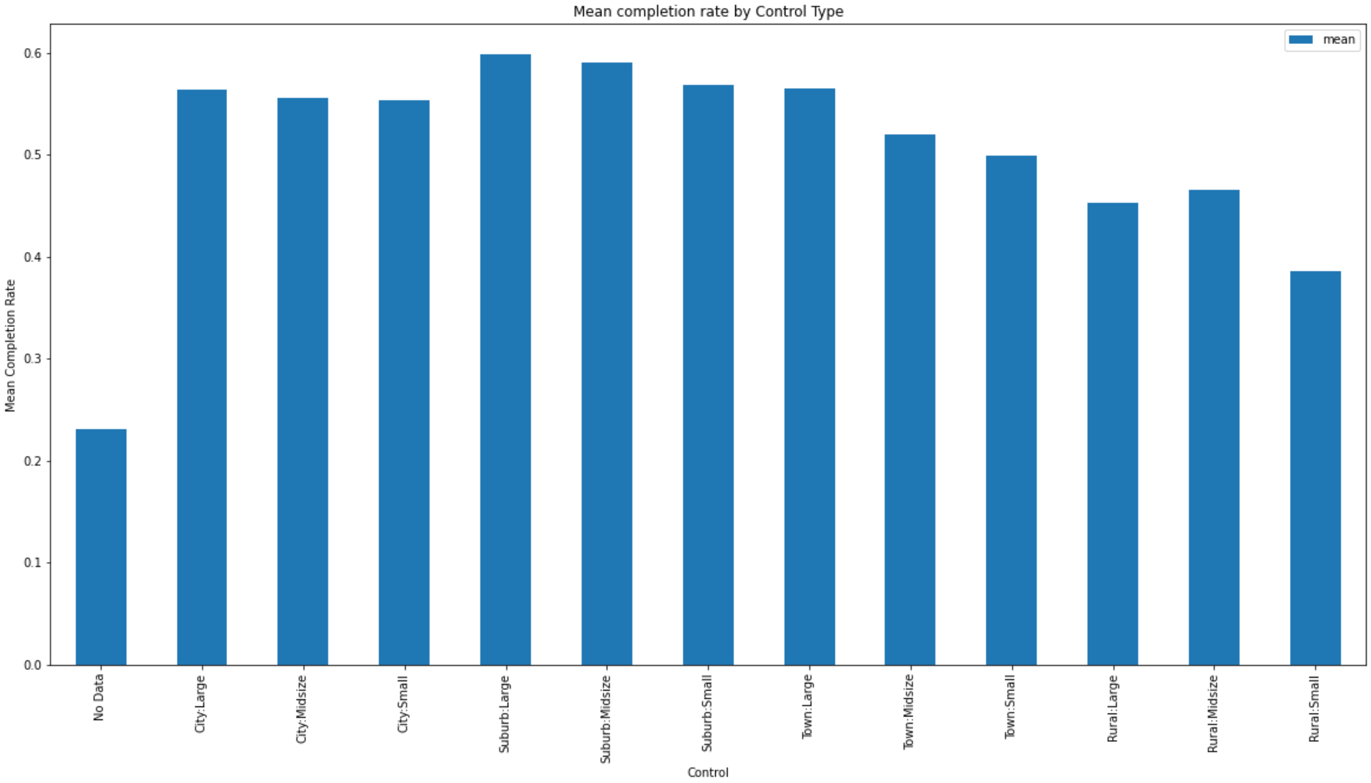
Suburb shows the highest mean completion rate followed by city with Rural being the lowest. Opportunities for Job will be most increased in a City or Town area, but the living cost will increase too. But for the suburb, Town or city will be at a drivable distance and have employment opportunities. But the cost of living might be lesser than in a city. If students cannot afford the cost of living plus their college expenses, they are likely to drop or change college.

In [30]:

```
df_char.groupby(by=["LOCATION_TYPE"])["CMP_RATE"].agg(["mean"]).plot(
    kind="bar",
    figsize=(20,10),
    xlabel="Control",
    ylabel="Mean Completion Rate",
    title="Mean completion rate by Control Type"
)
```

Out[30]:

```
<AxesSubplot:title={'center':'Mean completion rate by Control Type'}, xlabel='Control', ylabel
='Mean Completion Rate'>
```



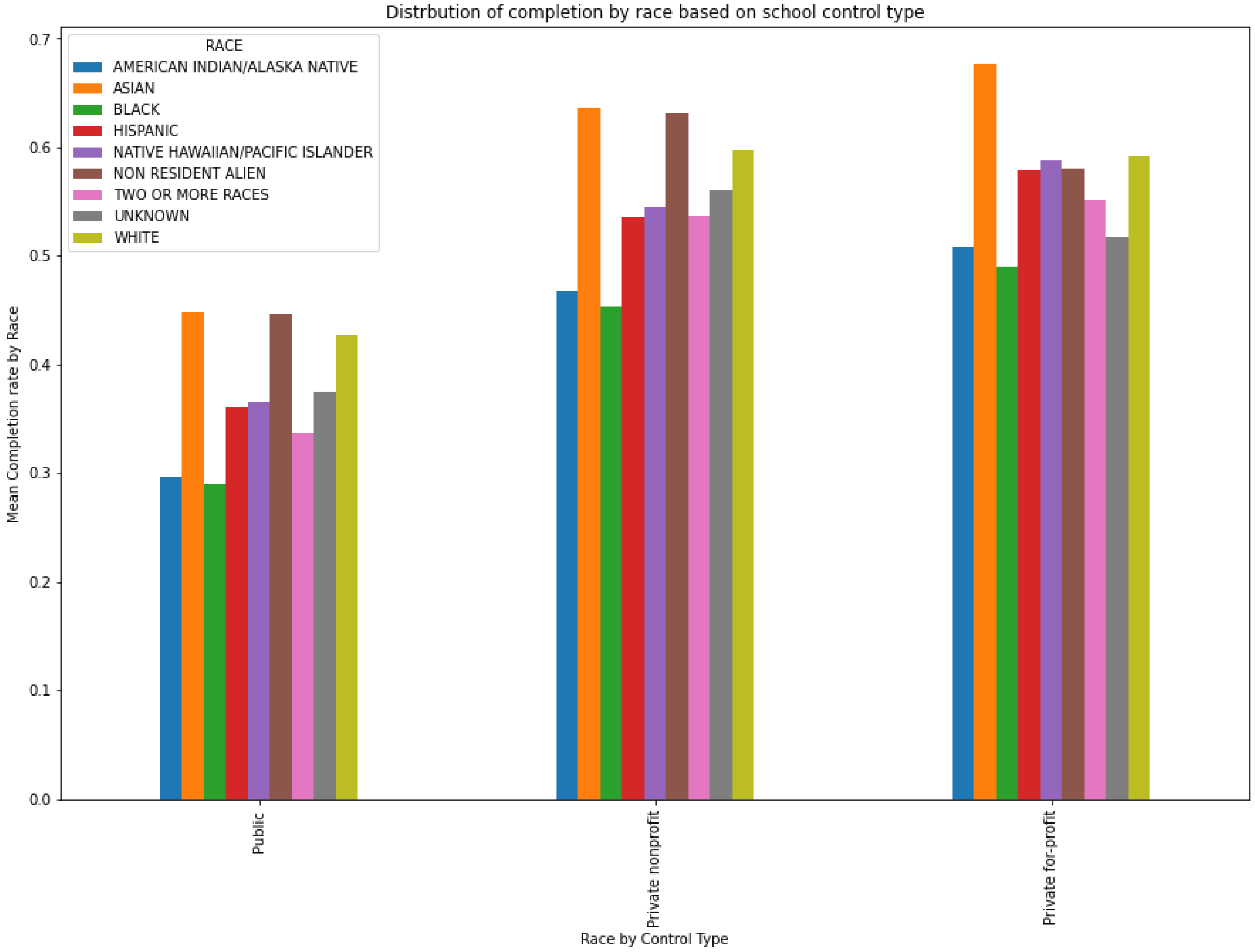
5. Extra credits: Difference in completion rate by race based on school characterstics

In [31]:

```
control_race_cmp = pd.merge(df_by_race, df_char[["CONTROL_NAME", "UNITID"]], on="UNITID")
pd.pivot_table(control_race_cmp, values="CMP_RATE_RACE", index=["CONTROL_NAME"], columns=["RACE
    kind="bar",
    figsize=(15,10),
    xlabel="Race by Control Type",
    ylabel="Mean Completion rate by Race",
    title="Distrbution of completion by race based on school control type"
    )
```

Out[31]:

```
<AxesSubplot:title={'center':'Distrbution of completion by race based on school control type'},
xlabel='Race by Control Type', ylabel='Mean Completion rate by Race'>
```



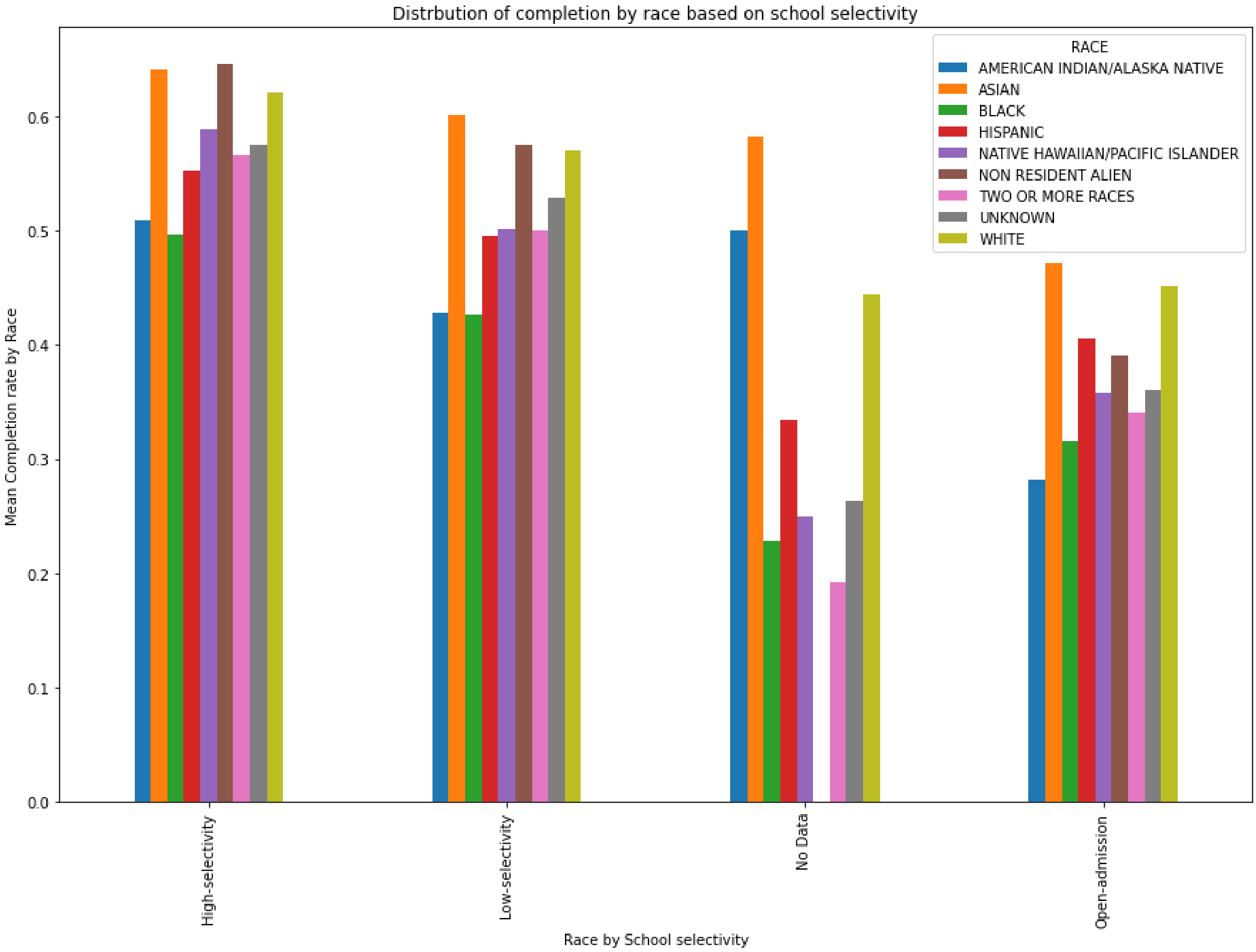
It is seen that Asian is the race with Highest completion rate across all school type with Non Resident Alien on second for 2 of them.

In [32]:

```
adm_selectivity_race_cmp = pd.merge(df_by_race, df_char[["ADMISSION_SELECTIVITY", "UNITID"]], c
pd.pivot_table(adm_selectivity_race_cmp, values="CMP_RATE_RACE", index=["ADMISSION_SELECTIVITY"
    kind="bar",
    figsize=(15,10),
    xlabel="Race by School selectivity",
    ylabel="Mean Completion rate by Race",
    title="Distrbution of completion by race based on school selectivity"
    )
```

Out[32]:

```
<AxesSubplot:title={'center':'Distrbution of completion by race based on school selectivity'},
xlabel='Race by School selectivity', ylabel='Mean Completion rate by Race'>
```



Non-Resident alien has the highest mean completion rate for highly selective school where as Asian for others.

6. Q/A of data according to Datasheets for Datasets

Answers to 5 questions of your choice from sections 3.1, 3.2, and 3.3 of Datasheets for Datasets, based on the documentation for the college scorecard data. Questions should come from at least two different sections of the paper.

a. For what purpose was the dataset created? Was there a specific task in mind? Was there a specific gap that needed to be filled? Please provide a description.

The College Scorecard project dataset is created to help students and families to compare how thriving individual postsecondary institutions are preparing their students to be successful. This data allows them to compare college costs and outcomes and weigh different colleges' tradeoffs, accounting for their own needs and educational goals.

b. Who created the dataset (e.g., which team, research group) and on behalf of which entity (e.g., company, institution, organization)?

Office of Planning, Evaluation and Policy Development (OPEPD) created the dataset on behalf of the U.S. Department of Education.

c. What do the instances that comprise the dataset represent (e.g., documents, photos, people, countries)? Are there multiple types of instances (e.g., movies, users, and ratings; people and interactions between them; nodes and edges)? Please provide a description.

The instances that comprise the data set represents the Institutions, available Academics, Admission rates and SAT scores, Cost of study, Elements to identify demographics about students, Financial Aids, Completion rates, i.e., finding a job, Earning, Loan Repayments. The dataset has a single instance that consists of data types such as string, integer, float, long.

d. What data does each instance consist of?

Each instance consists of data types such as string, integer, float, long representing various features such as institution names, states, coordinates, categorical variables, admission rates, completion rates, and more. One instance almost has 2392 variables.

e. How was the data associated with each instance acquired?

Data were collected from various sources such as Integrated Postsecondary Education Data System, National Students Loan Data System, Office of Post Secondary Education, Federal Student Aid, combining them to a single dataset.

Summary

Write two paragraphs reflecting on what you learned about this data, higher education, and data science through this assignment.

The primary purpose of this assignment was to learn to find data, study and understand data and apply data science techniques and methods to describe data to acquire various information and facts. The Most Recent Institution data is tabular that consists of different characteristics of schools and many other about different schools across the states. The objective of this data set was to enable students and families help to choose between colleges based on income and college costs, different characteristics, and trade-off comparisons.

Regarding Data Science, there were a lot of things to learn throughout the assignment. First and foremost, downloading the data, loading it, and exploring structural characteristics of data. We learned about data characteristics, their presentation, and the meaning they convey. We also learned about various Pandas techniques and functions such as grouping and aggregates used in Data Science to generate information. We learned about manipulating data using reshaping and selecting functions such as pivot and melt as data might not be correctly formatted. One of the significant learnings while working in this data set was choosing the variable being used. The data set had multiple variables for the completion rate, but we had to select a variable to maintain the uniformity of the results across the notebook. Choosing a completion rate for 150% was necessary because it was the only completion rate broken down into races. Data Science requires you to make these smaller choices, such as choosing suitable variables or graphs so that the result makes absolute sense. One of the main focuses of this assignment was the presentation of information. Data Science is not only about finding results. It is also about conveying those results to the respective audience in a meaningful way. Various graphs for different variables and distributions represent a different meaning. I learned that it is necessary to choose an appropriate graph to convey the proper interpretation of the data. More minor details like Legends, titles, labels play a more significant part in sharing the actual meaning of any graph. The formatting and organization of the actual Jupyter notebook play a vital role in Data Science. It is crucial to have proper headings and indentations mixing markdown and python cells to work on data and describe their meaning.