hbcu awards

September 13, 2023

1 The National Endowment for the Humanities's Grantmaking to Historically Black Colleges and Universities, 1965-Present

Erik Fredner

2023-09-13

2 Introduction

Of the awards granted by the National Endowment for the Humanities (NEH) to colleges and universities since its founding in 1965, what proportion have been allocated to historically Black colleges and universities (HBCUs)? And how has that allocation changed over time?

Answering these questions matters for our understanding of the past, present, and future of the NEH's grantmaking efforts. It has been frequently demonstrated by social scientists, economists, and humanists that actors both public and private have consistently underinvested in Black communities and institutions. Understanding the NEH's grantmaking history in relation to this widespread US pattern raises a broader question that we cannot answer with data alone. How should grantmakers and policymakers respond to these histories today?

2.1 Approach

By aligning the NEH's grant award data with the 2021 IPEDS Institutional Characteristics data (the most recent available), I determine whether an awarded institution is an HBCU.

I then filter the dataset to only consider other awards granted to Colleges and Universities.

Then, I output the following key numbers and figures, both in aggregate and over time:

- raw dollar amounts awarded to HBCUs and non-HBCUs
- 2022 dollar amounts adjusted using the Bureau of Labor Statistics' annualized Consumer Price Index averages
- total number of grants awarded to HBCUs and non-HBCUs
- odds ratios for HBCUs and non-HBCUs to receive a grant in a given decade, given prior awardees

Code can be found on the GitHub repository for this project and in the notebook below.

Please note that, due to the small values involved, I have chosen to visualize the data using tables, as they make it easier to see and interpret the fine gradations of change.

2.2 Key Findings

Since 1965, the NEH has awarded **$47 billion in 2022 dollars** to colleges and universities. Of that total, **$1.06 billion (2.3%)** has been granted to historically Black colleges and universities (HBCUs).

2.2.1 NEH awarded a smaller proportion of grant dollars to HBCUs in the 2020s than in 1970s

When he spoke at the National Cathedral on March 31, 1968, Martin Luther King, Jr. described a view that many Americans now hold when he said, "the arc of the moral universe is long but it bends toward justice." If bending toward justice in this case might suggest an absolute and relative increase in funding for HBCUs to compensate for structural disinvestment, the data counterintuively show almost the *opposite* pattern. In the 1970s, when the NEH's budget was larger than it is today, HBCUs received almost 4% of dollars awarded. So far in the 2020s, they have recieved about 1.6% of the funding. If patterns from the 2010s hold, the 2020s will have total funding of about 30% of the 1970s levels. HBCUs have been receiving a smaller portion of a shrinking pie, though their share has been fairly consistent since the 1990s.

Award Decade	HBCU Share of NEH Grant Dollars
1960	3.9%
1970	2.7%
1980	2.6%
1990	1.6%
2000	1.8%
2010	1.5%
2020	1.6%

2.2.2 NEH has granted a smaller proportion of its awards to HBCUs since the 1970s

Perhaps, even if awards today are smaller, we might expect that *more* projects from HBCUs would be funded. This is not the case either, although the proportion of awards allocated is higher than the proportion of dollars awarded. Awards to HBCUs peaked in the 1960s when they made up 3.3% of total awards, and reached their lowest point in the 1990s at about 2.1%. In the 2020s, they have received about 2.4% of total awards.

Award Decade	HBCU Share of NEH Awards
1960	3.3%
1970	2.2%
1980	3.1%
1990	2.1%
2000	2.8%
2010	2.2%
2020	2.4%

2.2.3 The odds of any HBCU receiving an award have often substantially higher than the odds of any non-HBCU receiving an award

Because there are a significantly smaller number of HBCUs than non-HBCUs, percentages can be misleading. Odds ratios account for such variation in absolute size among groups fairly robustly. They are also easy to calculate and reason about:

$$\frac{p_1/(1-p_1)}{p_2/(1-p_2)}$$

In our case, p_1 represents the total number of HBCUs that received an NEH award in a given decade divided by the number of HBCUs that had *ever* received an NEH award up to that date. p_2 is the same as p_1, but for non-HBCUs. The odds ratios tells us how much more likely it is for any given HBCU to receive a grant than any given non-HBCU. In cases where the odds ratio is greater than 1, it is more likely for a given HBCU to receive an award in a decade than a given non-HBCU.

Unlike the prior two key findings, which showed fairly consistent decline, odds ratios show a more complex picture. In the 1970s, a given HBCU had only about a 16% of receiving an award compared to non-HBCUs. Another way of saying this is that non-HBCUs were 1.84 times as likely to receive a grant as HBCUs. But in the 1980s, HBCUs had an astonishing 5.29 times greater chance of being selected than any given non-HBCU. Such huge fluctuations have stabilized more recently, with higher and lower odds ratios for these groups in alternate decades, and none so dramatic as the shift from the 1970s to the 1980s.

Award Decade	HBCU/CU Odds Ratio
1970	0.16
1980	5.29
1990	2.27
2000	1.62
2010	0.98
2020	1.06

2.2.4 Summary

It is surprising but true that HBCUs have received a consistently smaller relative share of the NEH's grants to colleges and universities since the 1970s. Less surprising but no less concerning is that they have also received a lower absolute number of dollars. A caveat to this point, however, is that, in some decades, HBCUs have had a significantly higher *chance* of receiving NEH awards than other colleges or universities that had previously won an NEH grant. Howeer, in the last two decades, those chances have become relatively even. This raises the question of whether even odds are *fair* odds.

As decades of reports by the National Endowment for the Arts have demonstrated, participation in artistic and humanitic activities (such as literary reading, which the NEA's reports also track) are strongly correlated with education, gender, income, and race (in roughly that order). One notion of equity policymakers could adopt would be reducing disparities in engagement with arts and humanities along these axes. NEH grantmaking could intervene in these areas, and increasing awards to HBCUs could be a part of a strategy of equity and repair.

2.3 Limitations

- This research demonstrates that, since the 1970s, NEH awards to HBCUs have been decreasing in absolute and relative terms as compared to other colleges and universities. However, it is a question of policy and institutional priorities as to what the *appropriate* level of funding would be.
- This studies *institutional* awards, not the races or ethnicities of awardees directly. This work does not imply that the NEH's awards to HBCUs represent the totality of its giving to Black scholars or Black institutions.
- IPEDs table does *not* include open date for colleges and universities, so odds ratios are calculated based on the proportion of all Colleges and Universities (HBCU or non-HBCU) that *had ever received an award from the NEH up to that decade* as compared to the the actual number of institutions awarded in a given year. This may erroneously assume that institutions that previously won awards did not cease operations (or rename themselves) in the interim.

3 Data Preparation

[ ]:
```
import os
import pandas as pd
from rapidfuzz import fuzz, process, utils
```

[ ]:
```
data_dir = "/Users/erik/code/neh-assessment/data"
```

[ ]:
```
# get decadal xml files
xml_files = []

for root, dirs, files in os.walk(data_dir):
    for file in files:
        if file.endswith(".xml"):
            xml_files.append(os.path.join(root, file))
```

[ ]:
```
# concatenate decadal xml files into one dataframe
l = list()

for file in xml_files:
    l.append(pd.read_xml(file))
```

[ ]:
```
df = pd.concat(l)
```

[ ]:
```
# subset df for needed columns
cols = [
    "AppNumber",
    "ApplicantType",
    "Institution",
    "OrganizationType",
    "InstPostalCode",
```

```
    "InstCountry",
    "Latitude",
    "Longitude",
    "YearAwarded",
    "Program",
    "Division",
    "AwardOutright",
    "AwardMatching",
    "SupplementAmount",
    "PrimaryDiscipline",
    "ParticipantCount",
]
```

[ ]:
```
df = df[cols].copy()
```

[ ]:
```
# calculate total awarded by NEH
df["total_award"] = df["AwardOutright"] + df["AwardMatching"] +␣
 ↪df["SupplementAmount"]
```

[ ]:
```
# calculate total award in 2022 dollars
cpi = pd.read_csv("/Users/erik/code/neh-assessment/data/cpi average.csv")
cpi_2022 = 292.655
awards = df[["AppNumber", "YearAwarded", "total_award"]].copy()
```

[ ]:
```
awards = pd.merge(awards, cpi, left_on="YearAwarded", right_on="cpi_year",␣
 ↪how="outer")
```

[ ]:
```
awards["total_award_2022"] = round(
    awards["total_award"] * (cpi_2022 / awards["cpi_annual_avg"]), 2
)
```

[ ]:
```
df = df.merge(awards[["AppNumber", "total_award_2022"]], on="AppNumber")
```

[ ]:
```
# add decade
df["award_decade"] = (df["YearAwarded"] // 10) * 10
```

[ ]:
```
# filter for US awardees and nonzero awards
df = df[(df["InstCountry"] == "USA") & (df["total_award"] > 0)]
```

[ ]:
```
# filter for colleges and universities
orgtypes = [
    "Four-Year College",
    "University",
    "Two-Year College",
]

df = df[df["OrganizationType"].isin(orgtypes)]
```

[ ]:
```
df.shape
```

[ ]:
```
(32538, 19)
```

[ ]:
```
# import the IPEDs data
ipeds = pd.read_csv(
    "/Users/erik/code/neh-assessment/data/hd2021.csv", encoding="latin-1"
)
```

[ ]:
```
# subset for needed cols
cols = [
    "UNITID",
    "INSTNM",
    "IALIAS",
    "ZIP",
    "SECTOR",
    "DEGGRANT",
    "HBCU",
    "TRIBAL",
    "LOCALE",
    "DEATHYR",
    "CARNEGIE",
    "LANDGRNT",
    "INSTSIZE",
    "LONGITUD",
    "LATITUDE",
]
```

[ ]:
```
ipeds = ipeds[cols].copy()
```

[ ]:
```
# tables with hbcu names + ipeds id
hbcus = ipeds[ipeds["HBCU"] == 1][["UNITID", "INSTNM"]].copy()
# table with df names + neh id
neh_insts = df[["AppNumber", "Institution"]].copy()
```

[ ]:
```
def get_inst_app_numbers(df, inst):
    """Get a list of app numbers associated with an institution."""
    return df[df["Institution"] == inst]["AppNumber"].tolist()
```

[ ]:
```
l = list()

for index, row in hbcus.iterrows():
    match = process.extractOne(
        row["INSTNM"],
        neh_insts["Institution"].unique(),
        scorer=fuzz.ratio,
        processor=utils.default_process,
```

```
    )
    score = match[1]
    if score >= 95:  # manually reviewed results and scores < 95 were inaccurate
        match_neh_AppNumbers = get_inst_app_numbers(neh_insts, match[0])
        for app in match_neh_AppNumbers:
            d = dict()
            d["hbcu"] = row["INSTNM"]
            d["hbcu_ipeds_UNITID"] = row["UNITID"]
            d["match"] = match[0]
            d["match_neh_AppNumber"] = app
            d["score"] = match[1]
            l.append(d)
```

[ ]:
```
matches = pd.DataFrame(l)
```

[ ]:
```
matches.shape
```

[ ]: (820, 5)

[ ]:
```
# apply matches to df
df["is_hbcu"] = df["AppNumber"].isin(matches["match_neh_AppNumber"])
```

[ ]:
```
df.shape
```

[ ]: (32538, 20)

4 Findings

4.1 Complete dataset, 1965-Present

[ ]:
```
total_awards = df.shape[0]
print(f"Total NEH awards granted to colleges and universities: {total_awards}")
```

```
Total NEH awards granted to colleges and universities: 32538
```

[ ]:
```
hbcu_awards = df[df["is_hbcu"] == True].shape[0]
print(
    f"Total NEH awards granted to historically Black colleges and universities␣
 ↪(HBCUs): {hbcu_awards}"
)
```

```
Total NEH awards granted to historically Black colleges and universities
(HBCUs): 820
```

[ ]:
```
print(
    f"Awards to HBCUs as a percentage of total awards: {round((hbcu_awards /␣
 ↪total_awards) * 100, 2)}%"
)
```

```
Awards to HBCUs as a percentage of total awards: 2.52%
```

[ ]:
```
total_2022 = round(df["total_award_2022"].sum() / 1000000, 2)
print(
    f"Total NEH dollars awarded to colleges and universities in millions of␣
 ↪2022 dollars: ${total_2022}M"
)
```

```
Total NEH dollars awarded to colleges and universities in millions of 2022
dollars: $4696.78M
```

[ ]:
```
total_hbcu_2022 = round(
    df[df["is_hbcu"] == True]["total_award_2022"].sum() / 1000000, 2
)
print(
    f"Total NEH dollars awarded to HBCUs in millions of 2022 dollars:␣
 ↪${total_hbcu_2022}M"
)
```

```
Total NEH dollars awarded to HBCUs in millions of 2022 dollars: $105.86M
```

[ ]:
```
hbcu_total_2022 = round((total_hbcu_2022 / total_2022) * 100, 2)
print(
    f"NEH dollars awarded to HBCUs as a percentage of total NEH dollars awarded␣
 ↪to colleges and universities: {hbcu_total_2022}%"
)
```

```
NEH dollars awarded to HBCUs as a percentage of total NEH dollars awarded to
colleges and universities: 2.25%
```

4.2 Decadal data

4.2.1 Total dollars awarded

[ ]:
```
cols = [
    "total_award_2022",
    "is_hbcu",
    "award_decade",
]
```

[ ]:
```
award_total_decade = (
    df[["award_decade", "total_award_2022"]].groupby("award_decade").sum()
)
```

[ ]:
```
# decadal NEH grants awarded relative to 1970s levels
award_total_decade / award_total_decade.loc[1970]
```

[ ]:
```
              total_award_2022
award_decade
1960                  0.037303
```

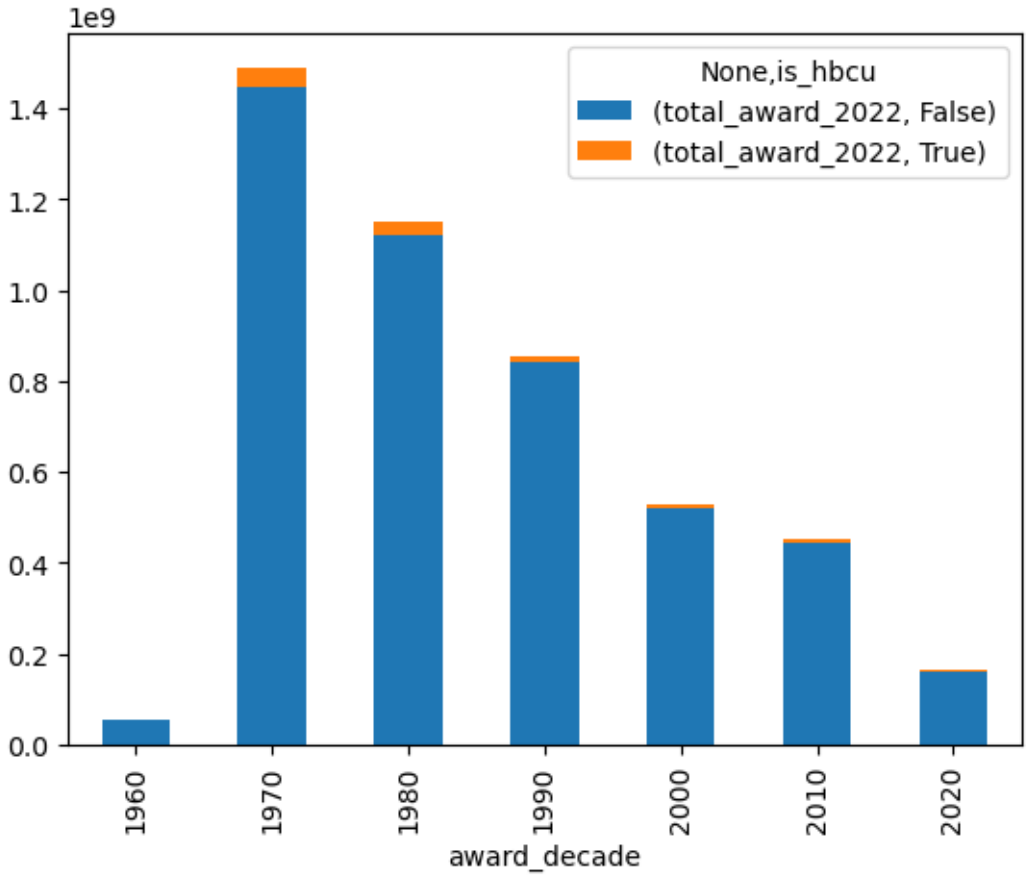
```
1970                 1.000000
1980                 0.773403
1990                 0.574490
2000                 0.354982
2010                 0.304168
2020                 0.110916
```

[ ]:
```
df[cols].groupby(["award_decade", "is_hbcu"]).sum()
```

[ ]:
```
                     total_award_2022
award_decade is_hbcu
1960         False        5.335124e+07
             True         2.175669e+06
1970         False        1.447683e+09
             True         4.087252e+07
1980         False        1.120850e+09
             True         3.040325e+07
1990         False        8.414105e+08
             True         1.374908e+07
2000         False        5.190240e+08
             True         9.386863e+06
2010         False        4.461081e+08
             True         6.662556e+06
2020         False        1.624918e+08
             True         2.613261e+06
```

[ ]:
```
df[cols].groupby(["award_decade", "is_hbcu"]).sum().unstack().plot(
    kind="bar", stacked=True
)
```

[ ]:
```
<Axes: xlabel='award_decade'>
```



[ ]:
```
df[cols].groupby(["award_decade", "is_hbcu"]).sum().divide(
    award_total_decade, level="award_decade"
)
```

[ ]:
```
                      total_award_2022
award_decade is_hbcu
1960         False            0.960818
             True             0.039182
1970         False            0.972542
             True             0.027458
1980         False            0.973591
             True             0.026409
1990         False            0.983922
             True             0.016078
2000         False            0.982236
             True             0.017764
2010         False            0.985285
             True             0.014715
2020         False            0.984172
```

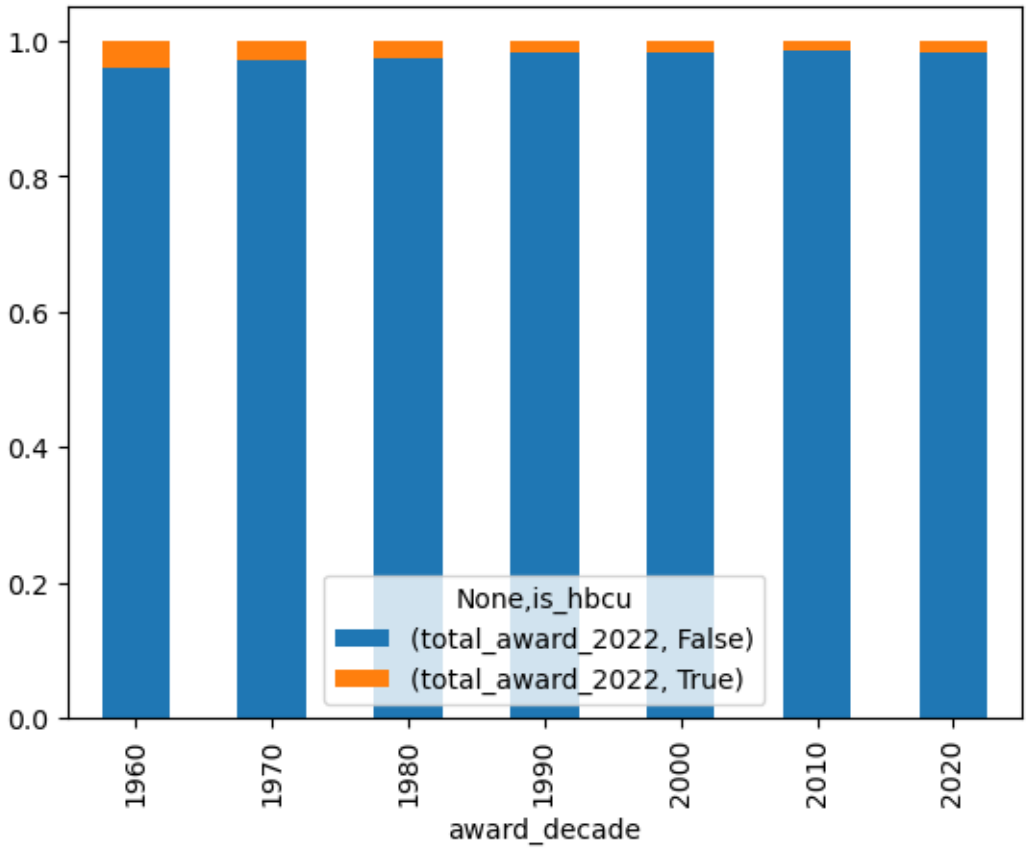
```
True                0.015828
```

[ ]:
```
# make table for the write-up
t = df[cols].groupby(["award_decade", "is_hbcu"]).sum().divide(
    award_total_decade, level="award_decade"
)
t.reset_index(inplace=True)
t = t[t['is_hbcu']==True][['award_decade', 'total_award_2022']]
t['total_award_2022'] = t['total_award_2022'].apply(lambda x: f"{round(x*100,␣
 ↪1)}%")
t.columns = ['Award Decade', 'HBCU Share of NEH Grant Dollars']
t.set_index('Award Decade', inplace=True)
print(t.to_markdown())
```

```
|   Award Decade | HBCU Share of NEH Grant Dollars   |
|---------------:|:----------------------------------|
|           1960 | 3.9%                              |
|           1970 | 2.7%                              |
|           1980 | 2.6%                              |
|           1990 | 1.6%                              |
|           2000 | 1.8%                              |
|           2010 | 1.5%                              |
|           2020 | 1.6%                              |
```

[ ]:
```
df[cols].groupby(["award_decade", "is_hbcu"]).sum().divide(
    award_total_decade, level="award_decade"
).unstack().plot(kind="bar", stacked=True)
```

[ ]:
```
<Axes: xlabel='award_decade'>
```



4.2.2 Total awards granted

[ ]:
```
# number of awards granted to colleges and universities by decade
awards_per_decade = df[["award_decade", "AppNumber"]].groupby("award_decade").
↪count()
awards_per_decade
```

[ ]:
```
              AppNumber
award_decade
1960                698
1970               8154
1980               7982
1990               5546
2000               4567
2010               3982
2020               1609
```

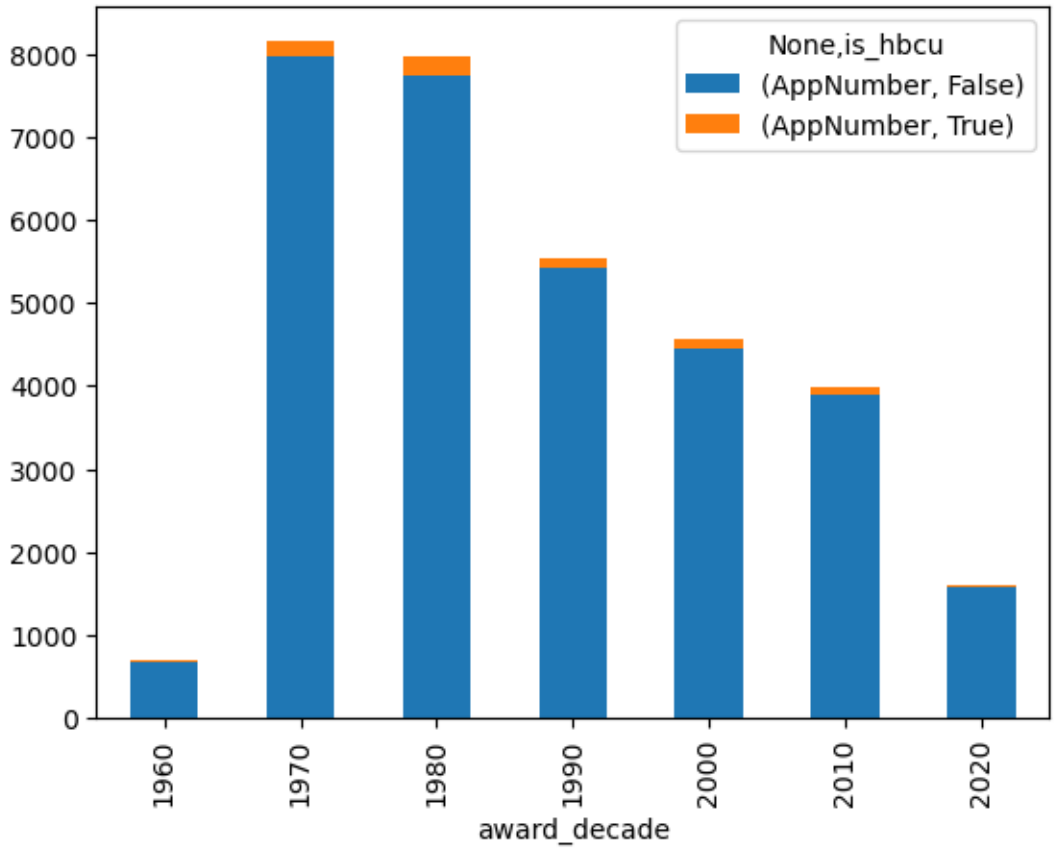
[ ]:
```
df[["award_decade", "AppNumber", "is_hbcu"]].groupby(
    ["award_decade", "is_hbcu"]
).count()
```

[ ]:

award_decade	is_hbcu	AppNumber
1960	False	675
	True	23
1970	False	7974
	True	180
1980	False	7734
	True	248
1990	False	5429
	True	117
2000	False	4440
	True	127
2010	False	3895
	True	87
2020	False	1571
	True	38

[ ]:
```
# number of awards granted to colleges and universities by decade
df[["award_decade", "AppNumber", "is_hbcu"]].groupby(
    ["award_decade", "is_hbcu"]
).count().unstack().plot(kind="bar", stacked=True)
```

[ ]: <Axes: xlabel='award_decade'>



[ ]:
```
awards_per_decade_hbcu = (
    df[["award_decade", "AppNumber", "is_hbcu"]]
    .groupby(["award_decade", "is_hbcu"])
    .count()
)
```

[ ]:
```
awards_per_decade_hbcu.divide(awards_per_decade, level="award_decade")
```

[ ]:
```
                          AppNumber
award_decade is_hbcu
1960         False     0.967049
             True      0.032951
1970         False     0.977925
             True      0.022075
1980         False     0.968930
             True      0.031070
1990         False     0.978904
             True      0.021096
2000         False     0.972192
             True      0.027808
```

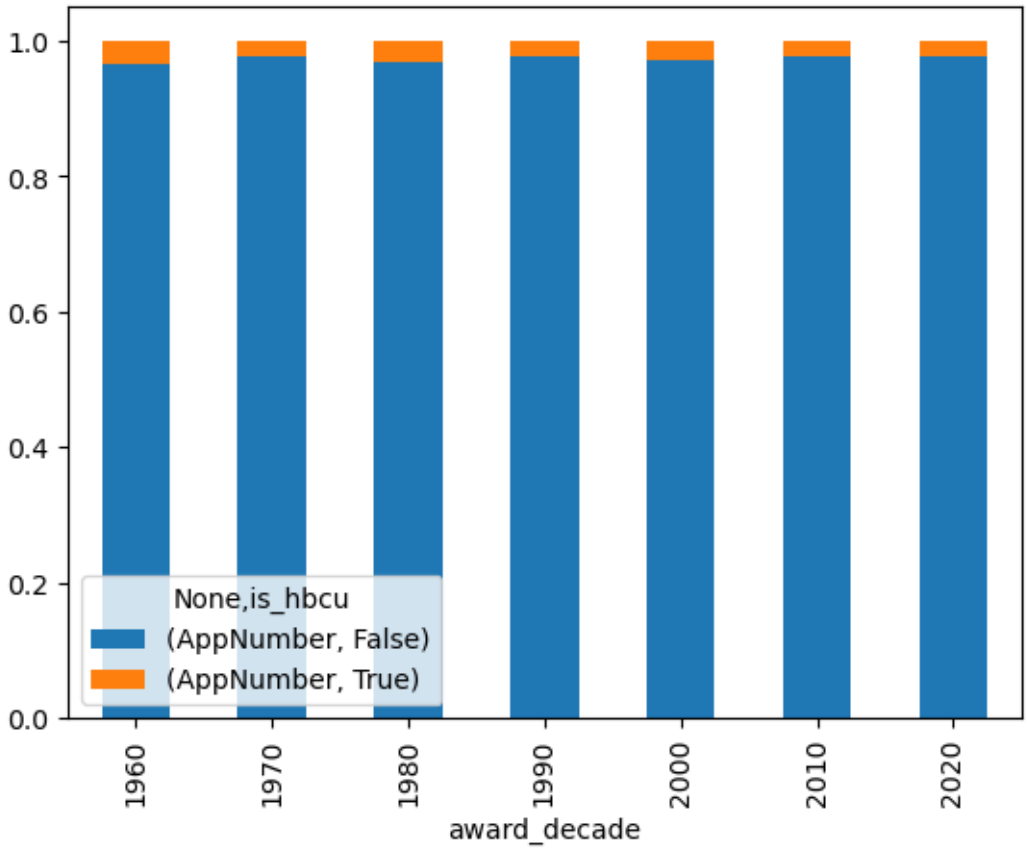
```
2010          False     0.978152
              True      0.021848
2020          False     0.976383
              True      0.023617
```

[ ]:
```
# table for write-up
t = awards_per_decade_hbcu.divide(awards_per_decade, level="award_decade")
t.reset_index(inplace=True)
t = t[t['is_hbcu']==True][['award_decade', 'AppNumber']]
t['AppNumber'] = t['AppNumber'].apply(lambda x: f"{round(x*100, 1)}%")
t.columns = ['Award Decade', 'HBCU Share of NEH Awards']
t.set_index('Award Decade', inplace=True)
print(t.to_markdown())
```

```
|   Award Decade | HBCU Share of NEH Awards   |
|---------------:|:---------------------------|
|           1960 | 3.3%                       |
|           1970 | 2.2%                       |
|           1980 | 3.1%                       |
|           1990 | 2.1%                       |
|           2000 | 2.8%                       |
|           2010 | 2.2%                       |
|           2020 | 2.4%                       |
```

[ ]:
```
awards_per_decade_hbcu.divide(awards_per_decade, level="award_decade").
 ↪unstack().plot(
    kind="bar", stacked=True
)
```

[ ]:
```
<Axes: xlabel='award_decade'>
```



4.3 Odds ratios

[ ]:
```
hbcu_unique = (
    df[df["is_hbcu"] == True]
    .groupby("award_decade")["Institution"]
    .unique()
    .reset_index()
)
cu_unique = (
    df[df["is_hbcu"] == False]
    .groupby("award_decade")["Institution"]
    .unique()
    .reset_index()
)
```

[ ]:
```
def get_institution_cumsum(df):
    """
    Get the number of unique institutions per decade from a dataframe.
    Used with dataframe filtered by HBCU.
    """
```

```
    awardees = set()
    l = list()

    for index, row in df.iterrows():
        awardees.update(row["Institution"])
        d = dict()
        d["award_decade"] = row["award_decade"]
        d["unique_institutions"] = len(awardees)
        l.append(d)

    return pd.DataFrame(l)
```

[ ]:
```
unique_institutions_cumsum = pd.merge(
    get_institution_cumsum(hbcu_unique),
    get_institution_cumsum(cu_unique),
    on="award_decade",
    suffixes=("_hbcu", "_cu"),
)
```

[ ]:
```
unique_institutions_cumsum
```

[ ]:

	award_decade	unique_institutions_hbcu	unique_institutions_cu
0	1960	18	286
1	1970	55	1392
2	1980	69	1672
3	1990	72	1785
4	2000	77	1961
5	2010	79	2173
6	2020	80	2231

[ ]:
```
unique_institutions = (
    df[["Institution", "is_hbcu", "award_decade"]]
    .groupby(["award_decade", "is_hbcu"])
    .nunique()
)
```

[ ]:
```
unique_institutions.reset_index(inplace=True)
```

[ ]:
```
unique_institutions = unique_institutions.pivot(
    index="award_decade", columns="is_hbcu", values="Institution"
)
```

[ ]:
```
unique_institutions
```

[ ]:

is_hbcu	False	True
award_decade		
1960	286	18

```
1970           1379     52
1980           1112     63
1990            808     47
2000            938     46
2010            974     35
2020            617     23
```

[ ]:
```
unique_institutions_cumsum.set_index("award_decade", inplace=True)
```

[ ]:
```
unique_institutions_cumsum
```

[ ]:
```
              unique_institutions_hbcu  unique_institutions_cu
award_decade
1960                                18                     286
1970                                55                    1392
1980                                69                    1672
1990                                72                    1785
2000                                77                    1961
2010                                79                    2173
2020                                80                    2231
```

[ ]:
```
l = list()
l.append(
    unique_institutions[True].divide(
        unique_institutions_cumsum["unique_institutions_hbcu"]
    )
)
l.append(
    unique_institutions[False].divide(
        unique_institutions_cumsum["unique_institutions_cu"]
    )
)
```

[ ]:
```
ratios = pd.concat(l, axis=1)
ratios.columns = ["hbcu", "cu"]
ratios
```

[ ]:
```
                 hbcu        cu
award_decade
1960         1.000000  1.000000
1970         0.945455  0.990661
1980         0.913043  0.665072
1990         0.652778  0.452661
2000         0.597403  0.478327
2010         0.443038  0.448228
2020         0.287500  0.276558
```

This uses the odds ratio for sequential observations as shown here:

Paul DiMaggio and Toqir Mukhtar, "Arts Participation as Cultural Capital in the United States, 1982–2002: Signs of Decline?," Poetics, Gender, networks, and cultural capital, 32, no. 2 (April 1, 2004): 169–94, https://doi.org/10.1016/j.poetic.2004.02.005.

[ ]:
```
l = list()

for index, row in ratios.iterrows():
    # div 0 warning in 1960s, where odds ratio is 1/(1-1)
    if row['award_decade'] == 1960:
        odds_ratio = 1
        l.append(odds_ratio)
        continue
    odds_ratio = (row["hbcu"] / (1 - row["hbcu"])) / (row["cu"] / (1 -␣
 ↪row["cu"]))
    l.append(odds_ratio)
```

[ ]:
```
ratios["hbcu/cu odds ratio"] = l
```

[ ]:
```
ratios.reset_index(inplace=True)
ratios
```

[ ]:

	index	award_decade	hbcu	cu	hbcu/cu odds ratio
0	0	1960	1.000000	1.000000	1.000000
1	1	1970	0.945455	0.990661	0.163403
2	2	1980	0.913043	0.665072	5.287770
3	3	1990	0.652778	0.452661	2.273218
4	4	2000	0.597403	0.478327	1.618337
5	5	2010	0.443038	0.448228	0.979209
6	6	2020	0.287500	0.276558	1.055532

[ ]:
```
# make table for report
t = ratios[['award_decade', 'hbcu/cu odds ratio']]
t = t[t['award_decade'] > 1960] # calculated relative to prior awards; 1960s␣
 ↪div0
t['hbcu/cu odds ratio'] = t['hbcu/cu odds ratio'].apply(lambda x: round(x, 2))
t.columns = ['Award Decade', 'HBCU/CU Odds Ratio']
t.set_index('Award Decade', inplace=True)
print(t.to_markdown())
```

Award Decade	HBCU/CU Odds Ratio
1970	0.16
1980	5.29
1990	2.27
2000	1.62
2010	0.98
2020	1.06

5 AI acknowledgment

This work was completed in Microsoft's Visual Studio Code with occasional assistance from GitHub Copilot.

Other references are linked in the document.