

Project

November 2, 2019

1 Interesting Dataset of UNC Undergraduate Students

1.1 Experiment

Collect a sample 100 UNC undergraduate students to record the following data:

Gender

Residency

Full/Part-Time Status

Name and PID of the subjects will not be recorded to keep their identity private through the study. From the 100 undergraduate students in the sample 50 will be selected at random using the Simple Random Sample technique in order to reduce bias.

1.2 Objectives

Analyze the data using graphs, and come up with proportions, for example proportion of female/male undergraduate students at UNC. It follows to compare the experiments results with real statistics provided by The Office of Institutional Research & Assessment. Next, we will compare the result of our experiment with the statistics and assess the experiment conclusions based on how close are our results from the provided facts.

1.2.1 Importing recorded sample of UNC Undergraduate Students

```
[25]: import pandas as pd
      pd.read_csv("UNC Undergraduate Sample.csv")
```

```
[25]:   Gender  Residency Full/Part-Time Status
0    Male    In-State    Full-Time
1    Male    In-State    Full-Time
2  Female    In-State    Full-Time
3    Male    In-State    Part-Time
4    Male    In-State    Full-Time
..    ...      ...      ...
95   Male    In-State    Full-Time
96   Male    In-State    Full-Time
97  Female    In-State    Full-Time
98  Female  Out-State    Full-Time
99  Female    In-State    Full-Time
```

[100 rows x 3 columns]

1.2.2 Determining simple random sample

```
[26]: import random

random.seed(0)

simple_random_sample = random.sample(range(100), 50)
```

1.2.3 Selected Subjects

```
[27]: import data_gen as dt
data_dict = dt.srs
data_frame = pd.DataFrame(data_dict)
data_frame
```

```
[27]:
```

	Subject ID	Gender	Residency	Full/Part-Time	Status
0	0049	Male	In-State		Part-Time
1	0097	Female	In-State		Full-Time
2	0053	Female	In-State		Full-Time
3	0005	Male	In-State		Full-Time
4	0033	Female	In-State		Full-Time
5	0065	Male	In-State		Full-Time
6	0062	Female	In-State		Full-Time
7	0051	Female	In-State		Full-Time
8	0038	Female	Out-State		Full-Time
9	0061	Female	In-State		Full-Time
10	0045	Female	In-State		Full-Time
11	0074	Male	In-State		Full-Time
12	0027	Male	Out-State		Full-Time
13	0064	Female	In-State		Part-Time
14	0017	Female	Out-State		Full-Time
15	0036	Female	In-State		Full-Time
16	0085	Male	In-State		Full-Time
17	0012	Female	In-State		Full-Time
18	0079	Male	In-State		Full-Time
19	0032	Male	In-State		Full-Time
20	0068	Female	Out-State		Full-Time
21	0077	Male	In-State		Full-Time
22	0018	Female	In-State		Full-Time
23	0039	Female	In-State		Full-Time
24	0082	Female	In-State		Full-Time
25	0009	Female	In-State		Part-Time
26	0042	Female	In-State		Full-Time

27	0060	Male	In-State	Full-Time
28	0071	Female	In-State	Full-Time
29	0075	Female	In-State	Full-Time
30	0089	Female	Out-State	Full-Time
31	0055	Male	In-State	Full-Time
32	0040	Female	In-State	Full-Time
33	0026	Male	In-State	Full-Time
34	0090	Male	In-State	Full-Time
35	0056	Female	In-State	Full-Time
36	0095	Male	In-State	Full-Time
37	0003	Male	In-State	Part-Time
38	0092	Female	In-State	Full-Time
39	0058	Male	In-State	Full-Time
40	0035	Male	In-State	Full-Time
41	0072	Female	Out-State	Full-Time
42	0000	Male	In-State	Full-Time
43	0096	Male	In-State	Full-Time
44	0046	Male	In-State	Full-Time
45	0098	Female	Out-State	Full-Time
46	0025	Female	In-State	Full-Time
47	0069	Female	In-State	Full-Time
48	0050	Male	In-State	Full-Time
49	0073	Male	In-State	Full-Time

1.2.4 Gender Distribution

```
[28]: gender = ["Female", "Male"]
count_female = 0
count_male = 0
gen_counts = []
for x in dt.sub_gen:
    if x == "Female":
        count_female = count_female + 1
    elif x == "Male":
        count_male = count_male + 1
gen_counts.append(count_female)
gen_counts.append(count_male)

[29]: print("Out of the 50 undergrads selected at random: ")
print("Female count: " + str(count_female))
print("Male count: " + str(count_male))
```

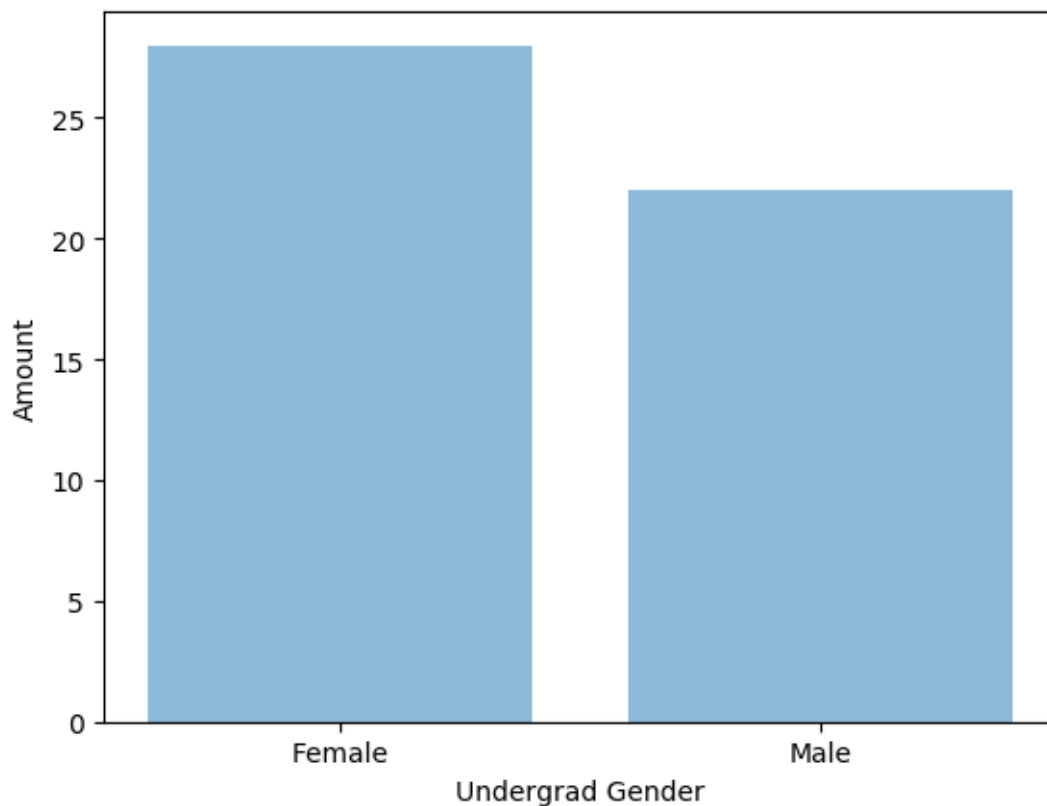
Out of the 50 undergrads selected at random:
 Female count: 28
 Male count: 22

```
[30]: import matplotlib.pyplot as plt; plt.rcdefaults()
import numpy as np
import matplotlib.pyplot as plt

y_pos = np.arange(len(gender))

plt.bar(y_pos, gen_counts, align='center', alpha=0.5)
plt.xticks(y_pos, gender)
plt.xlabel('Undergrad Gender')
plt.ylabel('Amount')

plt.show()
```



```
[31]: female_ratio = count_female / 50
male_ratio = count_male / 50

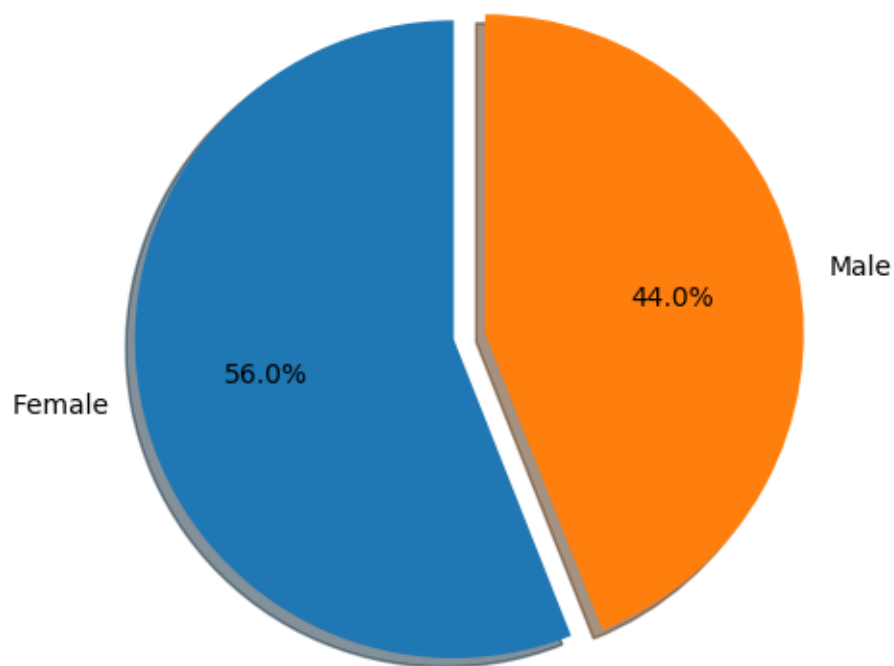
gen_ratios = []
gen_ratios.append(female_ratio)
gen_ratios.append(male_ratio)
```

```
[32]: import matplotlib.pyplot as plt

labels = 'Female', 'Male'
explode = (0, 0.1)

fig1, ax1 = plt.subplots()
ax1.pie(gen_ratios, explode=explode, labels=labels, autopct='%1.1f%%',
        shadow=True, startangle=90)
ax1.axis('equal')

plt.show()
```



1.2.5 Residency distribution

```
[33]: res_cat = ["In-State", "Out-State"]
count_in = 0
count_out = 0
res_counts = []
for x in dt.sub_res:
    if x == "In-State":
        count_in = count_in + 1
    elif x == "Out-State":
        count_out = count_out + 1
```

```
res_counts.append(count_in)
res_counts.append(count_out)
```

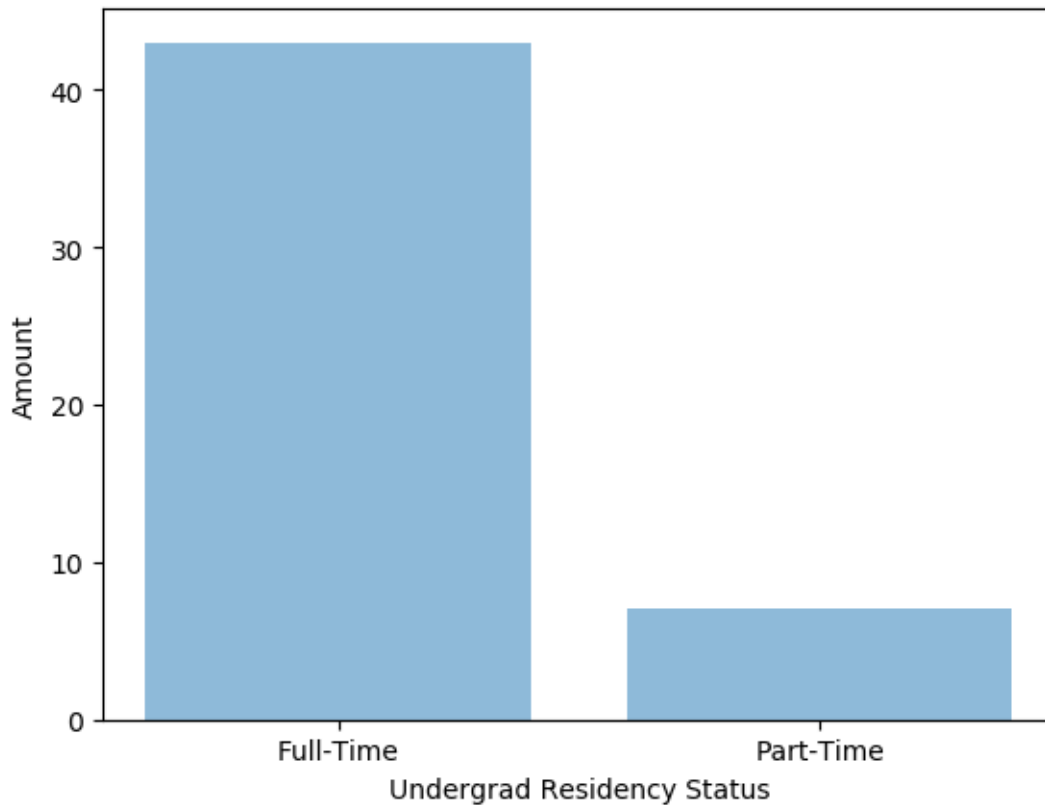
```
[34]: print("Out of the 50 undergrads selected at random: ")
      print("In-State count: " + str(count_in))
      print("Out-State count: " + str(count_out))
```

Out of the 50 undergrads selected at random:
In-State count: 43
Out-State count: 7

```
[42]: y_pos1 = np.arange(len(res_cat))

      plt.bar(y_pos1, res_counts, align='center', alpha=0.5)
      plt.xticks(y_pos1, res_cat)
      plt.xlabel('Undergrad Residency Status')
      plt.ylabel('Amount')

      plt.show()
```



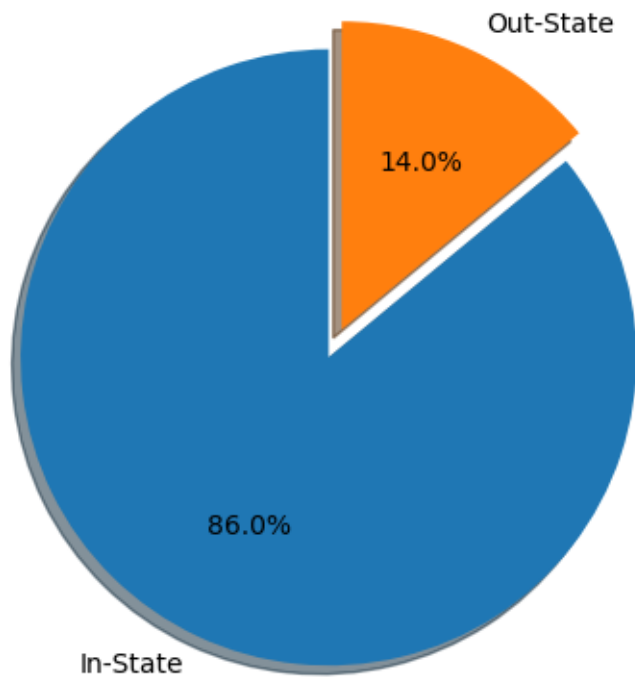
```
[43]: in_ratio = count_in / 50
      out_ratio = count_out / 50

      res_ratios = []
      res_ratios.append(in_ratio)
      res_ratios.append(out_ratio)

      labels = 'In-State', 'Out-State'
      explode = (0, 0.1)

      fig1, ax1 = plt.subplots()
      ax1.pie(res_ratios, explode=explode, labels=labels, autopct='%1.1f%%',
              shadow=True, startangle=90)
      ax1.axis('equal')

      plt.show()
```



1.2.6 Full/Part-Time Status distribution

```
[44]: sta_cat = ["Full-Time", "Part-Time"]
      count_full = 0
      count_part = 0
      sta_counts = []
```

```
for x in dt.sub_sta:
    if x == "Full-Time":
        count_full = count_full + 1
    elif x == "Part-Time":
        count_part = count_part + 1
sta_counts.append(count_full)
sta_counts.append(count_part)
```

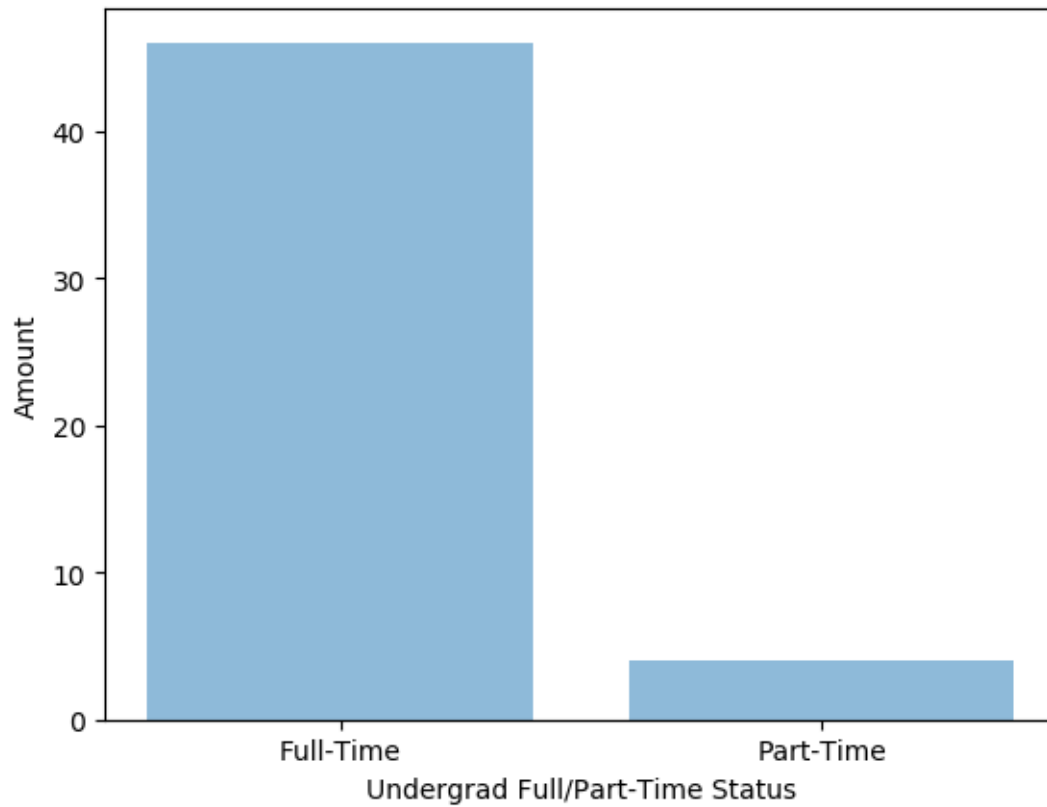
```
[45]: print("Out of the 50 undergrads selected at random: ")
      print("Full-Time count: " + str(count_full))
      print("Part-Time count: " + str(count_part))
```

```
Out of the 50 undergrads selected at random:
Full-Time count: 46
Part-Time count: 4
```

```
[47]: y_pos2 = np.arange(len(sta_cat))

      plt.bar(y_pos2, sta_counts, align='center', alpha=0.5)
      plt.xticks(y_pos2, sta_cat)
      plt.xlabel('Undergrad Full/Part-Time Status')
      plt.ylabel('Amount')

      plt.show()
```

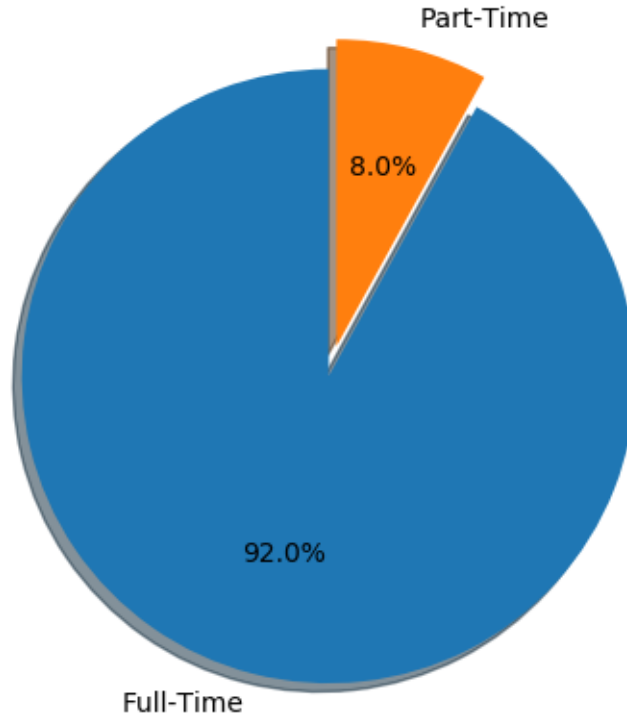
```
[48]: full_ratio = count_full / 50
      part_ratio = count_part / 50

      sta_ratios = []
      sta_ratios.append(full_ratio)
      sta_ratios.append(part_ratio)

      labels = 'Full-Time', 'Part-Time'
      explode = (0, 0.1)

      fig1, ax1 = plt.subplots()
      ax1.pie(sta_ratios, explode=explode, labels=labels, autopct='%1.1f%%',
              shadow=True, startangle=90)
      ax1.axis('equal')

      plt.show()
```



1.2.7 Conclusions

According to the experiment 56% of the UNC Undergraduate students are female, while 44 % are male. In conclusions there are more undergraduate females than males. In addition, 86% of the undergrads are In_State students, while 14% are Out-State. Finally, 92% of the undergrads are enrolled as Full-Time students, while only 8% are Part-Time.

1.2.8 Comparisons with ConnectCarolina Fall 2016 Census statistics

Variable/Category	ConnectCarolina Fall 2016 Census	Experiment Results
Gender: Female	58.4%	56%
Gender: Male	41.6%	44%
Residency: In-State	81.3%	86%
Residency: Out-State	18.7%	14%
Status: Full-Time	96.5%	92%
Status: Part-Time	3.5%	8%

In general our experiments results where very accurate. If more undergrads subjects are added to our sample, our experiment results will approach the statistics from the ConnectCarolina Fall 2016 Census. This is also possible if out of the 100 subjects we where to take a bigger simple random sample.

For access to the ConnectCarolina Fall 2016 Census visit: <https://oira.unc.edu/files/2018/06/Fact->

