CensusAssignment

April 11, 2021

Lab 4: Mini-Assignment

1 Census Lab Assignment

1.1 Task

Investigate a question of your choosing about a population from the 2016 Census data provided by the Australian Bureau of Statistics (ABS), and present your findings, as discussed in the accompanying assignment sheet.

1.1.1 Introduction

The aim of this analysis is to determine which types of educational institutions have the greatest number of students enrolled, and of the three of these with the greatest number of enrolments, how their students break down by gender. The analysis also aims to determine how the ratio of males to females changes between these institutions and why that may be. This is an area of interest as it can establish a framework to provide insight into how certain factors may influence a student's decision to attend a specific institution over the course of their life.

The data is sourced from the 2016 Census data provided by the Australian Bureau of Statistics (ABS). Specifically, table G15 - Type of Educational Institution Attending (Full/Part-Time Student Status by Age) by Sex. This table has specifically been selected as it provides counts of enrolment within educational institutions sorted by gender. This data and more information can be found at https://datapacks.censusdata.abs.gov.au/datapacks/ by selecting DataPack type - General Community Profile and Geography - Australia. For this analysis, the table is provided in the local directory.

1.1.2 Analysis

To analyse the data, it first needed to be cleaned and converted into an appropriate form. After importing the required Python modules, the data is read from it's .csv file and converted into two lists which contain (1) the type of institution and (2) the count of students in the institution.

[1]:
```
import matplotlib.pyplot as plt
from operator import itemgetter

# create variable to store file name
TYPE_EDU = '2016Census_G15_AUS.csv'
```

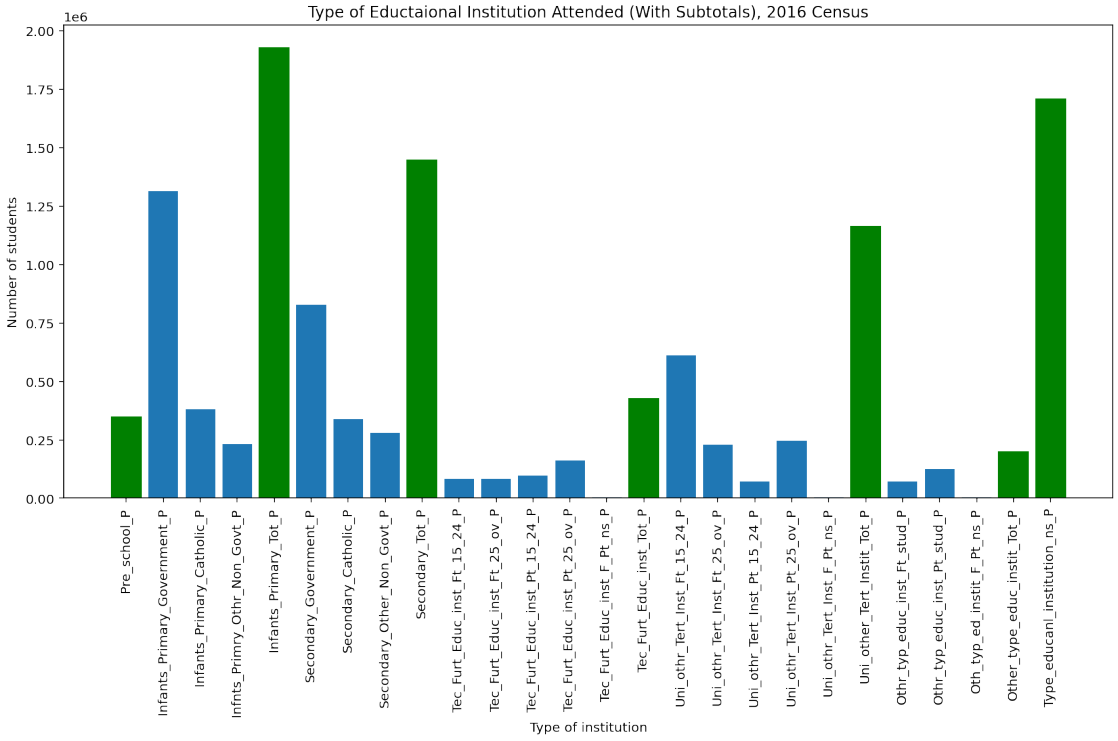
```
# open file and read its contents
with open(TYPE_EDU, 'r') as file:
    contents = file.read()

# split contents into two lists
lines = contents.splitlines()
edu_type = lines[0].split(',')[1:] # removes beginning ABS code cell heading
num_strings = lines[1].split(',')[1:] # removes beginning ABS code cell value

# numbers in list (2) are strings, convert to ints
num_people = []
for stringdata in num_strings:
    num_people.append(int(stringdata))

# remove grand total headings and values in both lists (Tot_M, Tot_F, Tot_P)
edu_type = edu_type[:-3]
num_people = num_people[:-3]
```

The two lists now contain values in three categories: male, female and people (total of male and female). Since we only want the total of both genders for now, the data is reduced to people only. Then, a high-level bar plot is created to show the total number of students enrolled in each institution.

[2]:
```
# slice to get grouped totals of two genders
grouped = slice(2, len(num_people), 3)
edu_type_grouped = edu_type[grouped]
num_people_grouped = num_people[grouped]

# bar plot of grouped totals with subtotals in green
plt.figure(figsize = (12, 8))
bar_grouped = plt.bar(edu_type_grouped, num_people_grouped)

# function to change color of bars with totals to green
def greenbar(bar_grouped, index):
    bar_grouped[index].set_color('g')

greenbar(bar_grouped, 0)
greenbar(bar_grouped, 4)
greenbar(bar_grouped, 8)
greenbar(bar_grouped, 14)
greenbar(bar_grouped, 20)
greenbar(bar_grouped, 24)
greenbar(bar_grouped, 25)

# plot formatting
plt.tick_params(axis='x', labelrotation=90)
```

```
plt.xlabel("Type of institution")
plt.ylabel("Number of students")
plt.title("Type of Eductaional Institution Attended (With Subtotals), 2016␣
 ↪Census")
plt.tight_layout()

plt.show()
```

[2]:



The plot displays the total number of students in each category, with each green bar representing a type of institution and each blue bar representing a specific group within that type. For example, the green bar 'Secondary_Tot_P' is the sum of Secondary Government, Secondary Catholic and Secondary Other/Non-government. This provides an overview of the data in terms of total people but it is not very clear in which three groups contain the highest number of students.

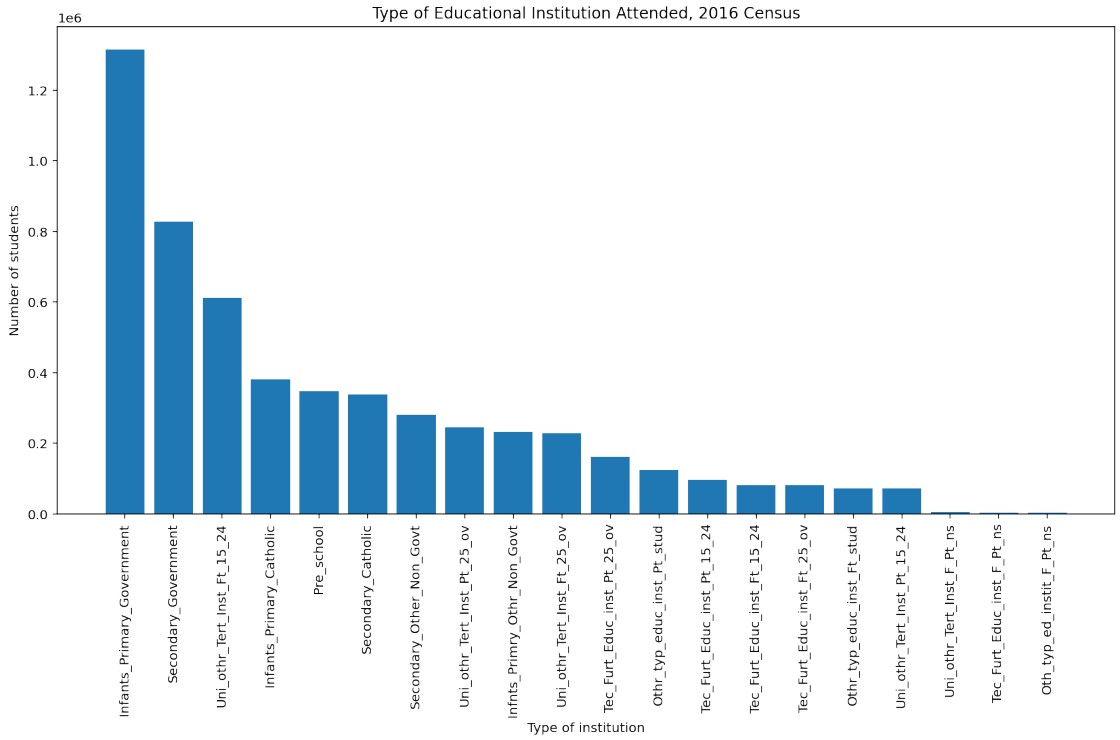
To find this, the subtotals (types) are removed from the data set. Since the category 'Type_educanl_institution_ns_P' refers to those who have not stated an institution, this is removed also as it does not provide any relevancy to the aim of the analysis. After this, each category and its corresponding count is sorted so that we can clearly see the institutions with the greatest number of students in descending order. A new bar plot is created from this reduced data.

[3]:

```
# function to pop subtotals
def popper(category):
    ix = edu_type_grouped.index(category)
    edu_type_grouped.pop(ix)
```

```
    num_people_grouped.pop(ix)

popper('Infants_Primary_Tot_P')
popper('Secondary_Tot_P')
popper('Tec_Furt_Educ_inst_Tot_P')
popper('Uni_other_Tert_Instit_Tot_P')
popper('Other_type_educ_instit_Tot_P')

# pop 'type of educational institution not stated'
ix = edu_type_grouped.index('Type_educanl_institution_ns_P')
edu_type_grouped.pop(ix)
num_people_grouped.pop(ix)

# sort pairs in descending order using zip
pairs = []
for i in range(len(num_people_grouped)):
    pairs.append((num_people_grouped[i], edu_type_grouped[i]))

pairs = zip(num_people_grouped, edu_type_grouped)
pairs_list = list(pairs)
pairs_list.sort(reverse = True)

# unzip pairs into new sorted lists
sorted_cat = []
sorted_num = []
for i in range(len(num_people_grouped)):
    sorted_cat.append(pairs_list[i][1][:-2]) # remove '_P' from end of each␣
 ↪category name
    sorted_num.append(pairs_list[i][0])

# new bar plot without subtotals, sorted from highest to lowest
plt.figure(figsize = (12, 8))
plt.bar(sorted_cat, sorted_num)
plt.tick_params(axis='x', labelrotation=90)
plt.xlabel("Type of institution")
plt.ylabel("Number of students")
plt.title("Type of Educational Institution Attended, 2016 Census")
plt.tight_layout()

plt.show()
```

[3]:

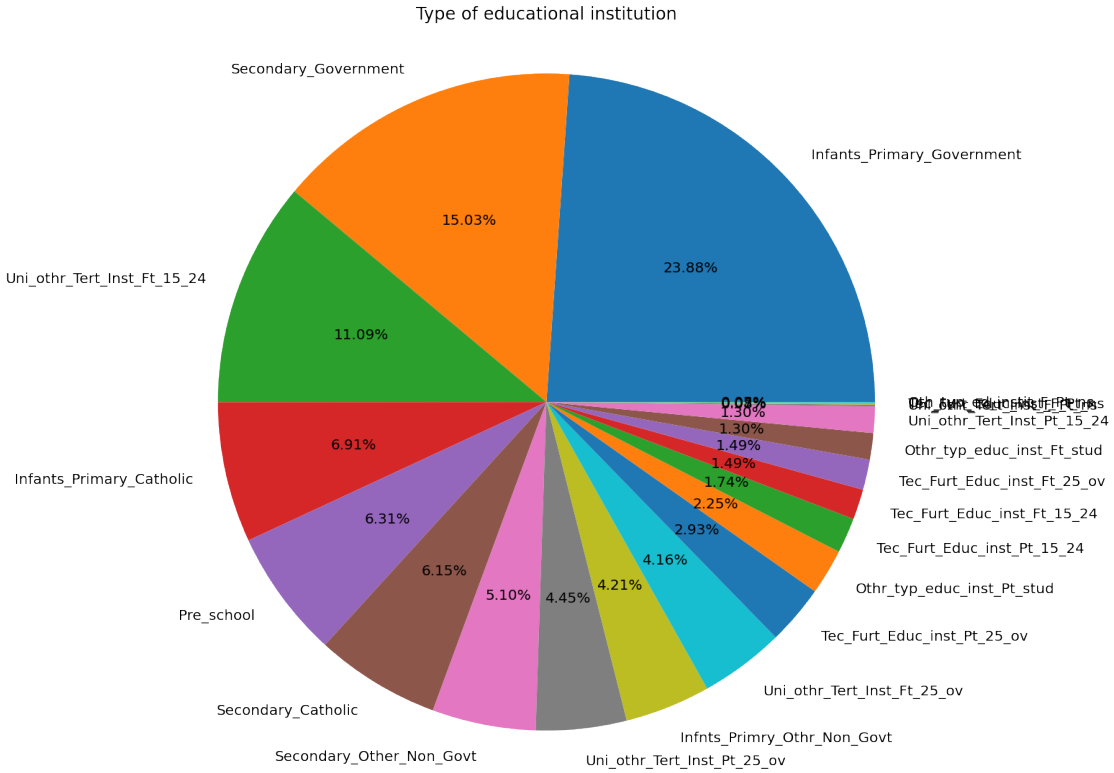


This plot is much more easily interpretable and clearly shows the institutions with the greatest number of students. From this, the top three categories are infants/primary school students in government institutions, secondary-age students in government institutions, and those who are full time university/tertiary students between the ages of 15 and 24.

Looking at the data in a simple pie chart, we observe the same thing.

[4]:
```
# pie chart of new sorted data
plt.pie(sorted_num, labels = sorted_cat, autopct='%1.2f%%', radius = 1.5)
plt.title("Type of educational institution\n\n\n\n\n") # newlines easily allow␣
 ↪title to show above plot

plt.show()
```

[4]:

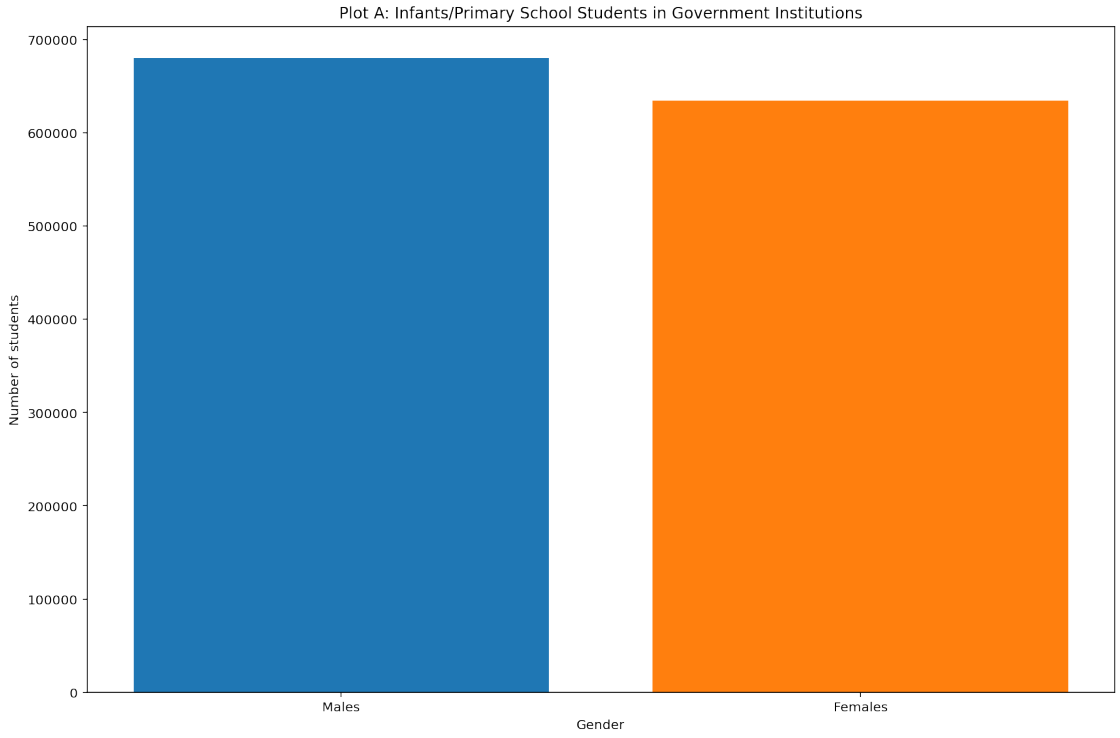


Now that we have the three groups, we can look into them further to get their distributions by gender, visualising how they compare with plots.

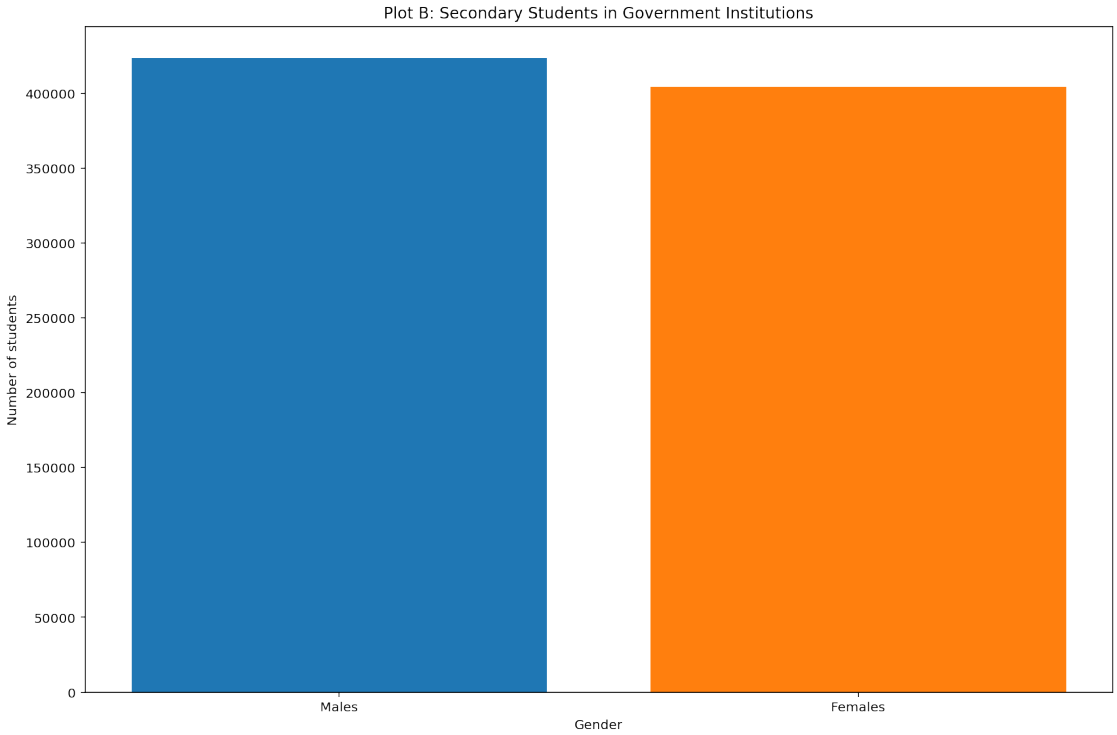
[5]:
```
# slice to get males
male = slice(0, len(num_people), 3)
edu_type_male = edu_type[male]
num_people_male = num_people[male]

# slice to get females
female = slice(1, len(num_people), 3)
edu_type_female = edu_type[female]
num_people_female = num_people[female]

# plot males and females for infants primary government
plt.figure(figsize = (12, 8))
plt.bar("Males", num_people_male[1])
plt.bar("Females", num_people_female[1])
plt.tick_params(axis='x')
plt.xlabel("Gender")
plt.ylabel("Number of students")
plt.title("Plot A: Infants/Primary School Students in Government Institutions")
plt.tight_layout()
```

```
plt.show()

# plot males and females for secondary government
plt.figure(figsize = (12, 8))
plt.bar("Males", num_people_male[5])
plt.bar("Females", num_people_female[5])
plt.tick_params(axis='x')
plt.xlabel("Gender")
plt.ylabel("Number of students")
plt.title("Plot B: Secondary Students in Government Institutions")
plt.tight_layout()

plt.show()

# plot males and females for uni other tert ft
plt.figure(figsize = (12, 8))
plt.bar("Males", num_people_male[15])
plt.bar("Females", num_people_female[15])
plt.tick_params(axis='x')
plt.xlabel("Gender")
plt.ylabel("Number of students")
plt.title("Plot C: Full-time Univeristy/Tertiary Students Between 15-24 Years␣
 ↪Old")
plt.tight_layout()

plt.show()
```

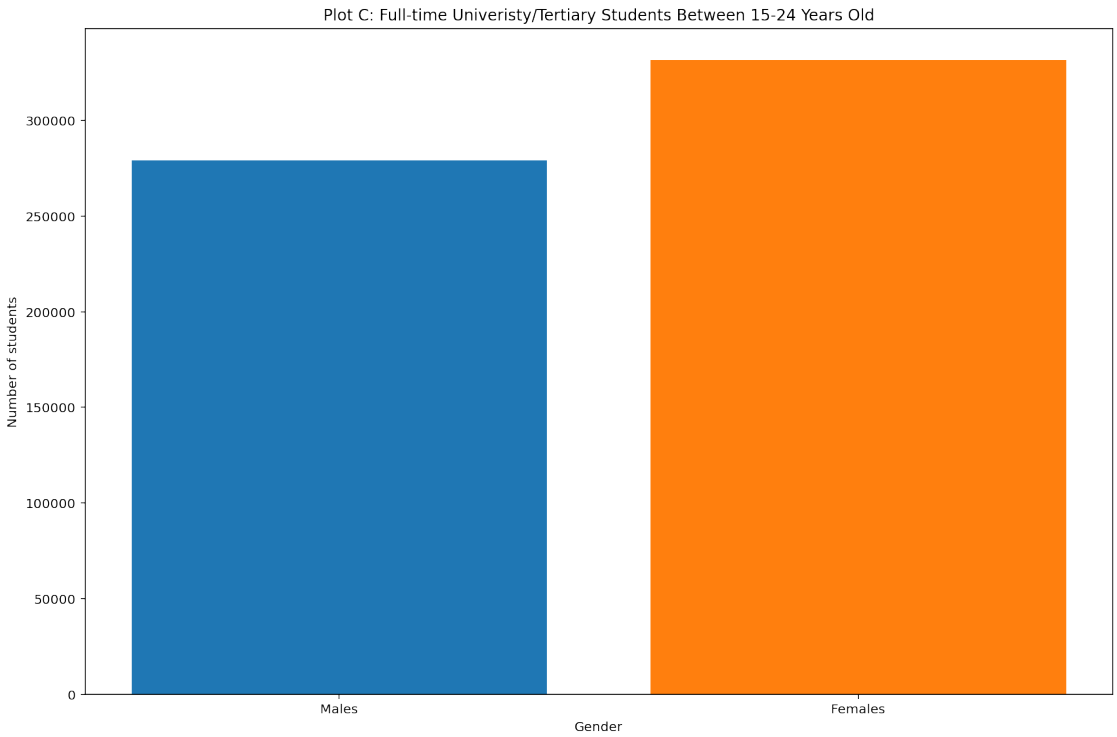
[5]:



[5]:


Plot B: Secondary Students in Government Institutions
Number of students
0
50000
100000
150000
200000
250000
300000
350000
400000
Males
Females
Gender

[5]:


Plot C: Full-time Univeristy/Tertiary Students Between 15-24 Years Old
Number of students
0
50000
100000
150000
200000
250000
300000
Males
Females
Gender

From plots A, B and C above we can determine that for infants/primary school students as well as secondary student in government institutions, there are slightly more males than females enrolled. For the third plot regarding university/tertiary students who are full-time and between the ages 15-24, there are more females than males enrolled. We can further calculate the percentage difference between the two for each type using a simple formula.

[6]:
```
# percent difference calcuation
percent_differenceA = round(((num_people_male[1] - num_people_female[1]) /␣
 ↪num_people_female[1]) * 100, 3)
percent_differenceB = round(((num_people_male[5] - num_people_female[5]) /␣
 ↪num_people_female[5]) * 100, 3)
percent_differenceC = round(((num_people_female[15] - num_people_male[15]) /␣
 ↪num_people_male[15]) * 100, 3)

# summary
print("Plot A: there are", percent_differenceA, "percent more males than␣
 ↪females enrolled")
print("Plot B: there are", percent_differenceB, "percent more males than␣
 ↪females enrolled")
print("Plot C: there are", percent_differenceC, "percent more females than␣
 ↪males enrolled")
```

```
Plot A: there are 7.225 percent more males than females enrolled
Plot B: there are 4.792 percent more males than females enrolled
Plot C: there are 18.842 percent more females than males enrolled
```

In Australia, schooling is compulsory for children between the ages of six and sixteen. The analysis shows that at primary-school age, there are approximately 7.2% more males than females enrolled in schooling. This difference decreases to around 4.8% as a student moves into secondary school age. One reason for this could be that in general, the number of males that move into the labouring workforce near the end of their secondary schooling heavily outweighs the number of females who do the same.

Since labouring jobs generally do not require educational qualifications for entry (other than completion of high-school in some cases) it follows that there would be a notable amount of male students who, when they turn sixteen and are no longer required to attend school, move straight into this workforce. This would partially account for the decrease in the gap between the enrolment numbers of males and females during these years as shown in the data.

After completing high-school, it's likely that an even greater number of males would move into the manual labour workforce, possibly accounting for the large amount and overtaking of the number of females enrolled in university/tertiary institutions compared to males. This is again supported by the data, which shows that for this age group there are around 18.8% more females enrolled in educational institutions.

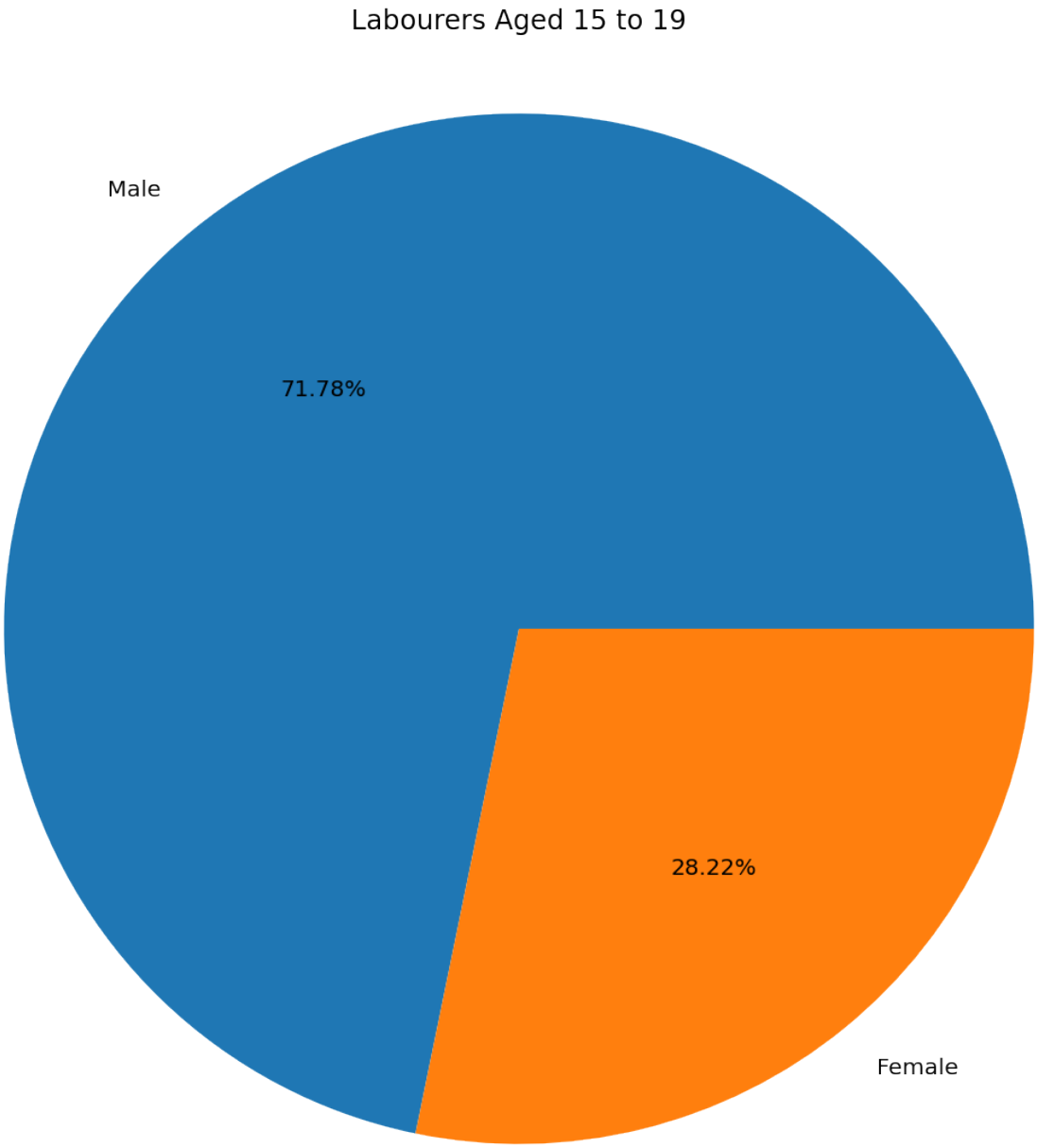
To see how this carries over into the real world, we can use a new dataset also from the 2016 Census data – Occupation by Age by Sex. As per the previous dataset, this data is read and converted

into to two lists of headings and counts. The aim is to compare the number of labourers aged 15 to 19 by gender and see if supports the hypothesis that there will be more male labourers by a large amount. A simple pie chart is created to show the findings.

[7]:
```
# read data and split into two lists as with previous dataset
OCCU = '2016Census_G57A_AUS.csv'
with open(OCCU, 'r') as file:
    contents = file.read()

lines = contents.splitlines()
work_type = lines[0].split(',')[1:]
num_strings = lines[1].split(',')[1:]

num_people = []
for stringdata in num_strings:
    num_people.append(int(stringdata))

# store variable with number of male labourers age 15-19
Male_15_to_19_Labourers = work_type[7]
Male_15_to_19_Labourers_Num = num_people[7]

# store variable with number of female labourers age 15-19
Female_15_to_19_Labourers = work_type[107]
Female_15_to_19_Labourers_Num = num_people[107]

# store combination of both groups in new lists
Male_Female_Labourer = []
Male_Female_Labourer_Num = []

Male_Female_Labourer.append("Male")
Male_Female_Labourer.append("Female")

Male_Female_Labourer_Num.append(Male_15_to_19_Labourers_Num)
Male_Female_Labourer_Num.append(Female_15_to_19_Labourers_Num)

# summary
print("Male Labourers Aged 15 to 19:", Male_15_to_19_Labourers_Num)
print("Female Labourers Aged 15 to 19:", Female_15_to_19_Labourers_Num)

# pie chart of male and female labourers age 15-19
plt.pie(Male_Female_Labourer_Num, labels = Male_Female_Labourer, autopct='%1.
 ↪2f%%', radius = 1.5)
plt.title("\n\nLabourers Aged 15 to 19\n\n\n\n\n")

plt.show()
```

```
Male Labourers Aged 15 to 19: 77370
Female Labourers Aged 15 to 19: 30420
```

[7]:



1.1.3 Conclusion

From the pie chart, it is clear that for this age group, the number of males employed in labouring jobs is much greater than the number of females employed. This supports the hypothesis for why the changes of educational enrolment may occur - because there are relatively less males seeking education during this period of their lives due to a large amount of them choosing to join the labouring workforce. The analysis has successfully determined which educational institutions have the greatest enrolment and proposed a strong argument for why the differences in enrolment

numbers between genders exist.

1.1.4 Declaration

This declaration should be completed and remain attached to the bottom of your submission.

I am aware of the University's policy on academic conduct and I declare that this assignment is my own work entirely and that suitable acknowledgement has been made for any sources of information used in preparing it. I have retained a hard copy for my own records.

Name: Gregory Edmonds

Date: 07/09/2020
