MATH2349 Semester 2, 2018

Code

Assignment 3 - Victorian Education Data by Local Government Area

Verity Miles(s3644459), Sam Holt (s3381728), Meg Cuddihy (s3608125)

20 October 2018

Required packages

Hide

```
library(readr)
library(readxl)
library(dplyr)
```

```

Attaching package: <U+393C><U+3E31>dplyr<U+393C><U+3E32>

The following objects are masked from <U+393C><U+3E31>package:stats<U+393C><U+3E32>:

    filter, lag

The following objects are masked from <U+393C><U+3E31>package:base<U+393C><U+3E32>:

    intersect, setdiff, setequal, union
```

Hide

```
library(tidyr)
library(editrules)
```

```
Loading required package: igraph

Attaching package: <U+393C><U+3E31>igraph<U+393C><U+3E32>

The following object is masked from <U+393C><U+3E31>package:tidyr<U+393C><U+3E32>:

    crossing

The following objects are masked from <U+393C><U+3E31>package:dplyr<U+393C><U+3E32>:

    as_data_frame, groups, union

The following objects are masked from <U+393C><U+3E31>package:stats<U+393C><U+3E32>:

    decompose, spectrum

The following object is masked from <U+393C><U+3E31>package:base<U+393C><U+3E32>:

    union


Attaching package: <U+393C><U+3E31>editrules<U+393C><U+3E32>

The following objects are masked from <U+393C><U+3E31>package:igraph<U+393C><U+3E32>:

    blocks, normalize

The following object is masked from <U+393C><U+3E31>package:tidyr<U+393C><U+3E32>:

    separate

The following object is masked from <U+393C><U+3E31>package:dplyr<U+393C><U+3E32>:

    contains
```

Hide

```
library(knitr)
library(mlr)
```

```
Loading required package: ParamHelpers

Attaching package: <U+393C><U+3E31>ParamHelpers<U+393C><U+3E32>

The following object is masked from <U+393C><U+3E31>package:editrules<U+393C><U+3E32>:

    isFeasible
```

Executive Summary

The purpose of this report is to demonstrate the concepts and techniques used to preprocess data, preparing it for analysis and modelling. We extracted two data sets: a 2016 Australian Census file containing information on education levels in different geographical areas in Victoria and a geographic concordance that matches Census statistical areas to Local Government Areas, which are commonly used. By merging these sets, the data becomes more usable and understandable to a wider range of people. We imported the data into R using appropriate import functions for each data file. We examined both data sets to understand the data and inform how we would preprocess it. This inspection led us to make some adjustments to the data sets, such as trimming rows. We compared the data sets to Hadley Wickham's Tidy Data Principles (2016) and found that the education data set was not tidy, so we converted the data from wide to long format. We then set one of the variables to an ordered factor. The next step was joining the two datasets. We inspected the results and generated a new variable to group and summarise the data. We inspected the joined data set for missing values and determined it would be appropriate to exclude them. We scanned for outliers, focusing on postgraduate and undergraduate data subsets. This was achieved by Tukey's Method of identifying outliers, using boxplots, on both total counts and proportional counts for each statistical area. These outliers were addressed through Windsorisation. We investigated the postgraduate and undergraduate data further by examining the distribution of each set on a histogram. The data was heavily positively skewed, so we applied a natural logarithm transformation to normalise the distribution. Finally, we tested how well machine learning could predict the relationship between undergraduate and postgraduate education.

Data

Data set 1: Education Levels

The first data set that we used comes from the 2016 Australian Census made available by the Australian Bureau of Statistics (ABS). We used a public TableBuilder account to extract educations level for each Statistical Area (SA1) within Victoria. Click here for information on TableBuilder (https://auth.censusdata.abs.gov.au/webapi/jsf/login.xhtml). SA1s are the smallest geographic unit available for the majority of census data. They generally have a population of between 200 and 800 people. Click here for more information (http://www.abs.gov.au/ausstats/abs@.nsf/Lookup/by%20Subject/1270.0.55.001~July%202016~Main%20Features~Statistical%20Area%20Level%201%20(SA1)~

The columns in this data set are:

- SA1 7 digit code
- Postgraduate Degree Level
    - Doctoral Degree Level
    - Masters Degree Level
- Graduate Diploma and Graduate Certificate Level
- Bachelor Degree Level
- Advanced Diploma and Diploma Level
- Certificate III & IV Level
- Secondary Education Years 10 and above
- Certificate I &II Level
- Secondary Education Years 9 and below
- Supplementary Codes
- Not stated
- Not applicable
- Total

The different education levels are coded as per the Australian Classification of Education (ASCED) 2001. For more information click here (http://www.abs.gov.au/ausstats/abs@.nsf/Latestproducts/1272.0Main%20Features12001?opendocument&tabname=Summary&prodno=1272.0&issue=2001&num=&view=).

When using data from the Australian Census, it is important to remember that all data is self-reported. This means that care needs to be taken when making conclusions from this data source. Additionally, cells with small numbers are randomly adjusted to protect confidentiality. This is something to be aware of, particularly when aggregating data to larger geographic areas (like we are in this assignment).

Data set 2: Geographic Look Up

The second data set that we used is a geographic lookup linking SA1s to local government areas (LGAs) and the Greater Capital City Statistical Area (GCCSA) of Greater Melbourne. We compiled this lookup using geographic files for SA1s and LGAs from the ABS and did a spatial join (within QGIS). Click here for more information about QGIS (https://www.qgis.org/en/site/).

The columns in this data set are:

- SA1_7DIG16
    - 7 digit unique identifier for SA1
- LGA_NAME17
    - Local council name
- MetroMelbourne
    - Indicator of whether the SA1 is within metropolitan Melbourne

Hide

```
education <- read_excel("allVic_education.xls", skip = 8, col_names = TRUE)
kable(head(education))
```

HEAP - 1 Digit Level	X__1	Postgraduate Degree Level	Graduate Diploma and Graduate Certificate Level	Bachelor Degree Level	Advanced Diploma and Diploma Level	Certificate III & IV Level	Secondary Education - Years 10 and above	Certificate I & II Level	Secondary Education - Years 9 and below	Supplementary Codes	Not stated	appli
NA	SA1 (UR)	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	
NA	2100101	22	12	44	31	61	111	0	25	10	32	
NA	2100102	0	0	24	16	23	58	0	52	3	21	
NA	2100105	9	5	30	31	46	86	0	19	11	31	
NA	2100106	13	16	69	31	75	154	0	37	8	48	

HEAP - 1 Digit Level	X__1	Postgraduate Degree Level	Graduate Diploma and Graduate Certificate Level	Bachelor Degree Level	Advanced Diploma and Diploma Level	Certificate III & IV Level	Secondary Education - Years 10 and above	Certificate I & II Level	Secondary Education - Years 9 and below	Supplementary Codes	Not stated	appli
NA	2100107	12	8	45	22	54	94	0	51	7	34	

Hide

```
geo_lookup <- read_csv("SA1_LGA_LookUp.csv")
```

```
Parsed with column specification:
cols(
  SA1_7DIG16 = col_integer(),
  LGA_NAME17 = col_character(),
  MetroMelbourne = col_character()
)
```

Hide

```
kable(head(geo_lookup))
```

SA1_7DIG16	LGA_NAME17	MetroMelbourne
2115615	Manningham (C)	Greater Melbourne
2115616	Manningham (C)	Greater Melbourne
2115617	Manningham (C)	Greater Melbourne
2115618	Manningham (C)	Greater Melbourne
2115619	Manningham (C)	Greater Melbourne
2115620	Manningham (C)	Greater Melbourne

Understand

The following techniques were used to inspect the data:

- head() - to inspect the first few rows for any extraneous rows or columns that need to be trimmed and to get an intial picture of the data
- tail() - to inspect the last few rows for any extraneous rows that need to be trimmed
- dim() - to check the size of the data sets
- str() - to check the internal structure of the data sets and identify the variable type of each attribute
- names() - to check the names of the attributes in the data sets and identify all the information each data set contains
- class() - to check the class of the overall data sets

This gave us a complete understanding of the data so we chose not to use additonal functions, such as attributes() or glimpse(), as we felt this would be redundant. At this stage, we considered the attribute data types and whether they would need to be changed in the next step of preprocessing.

Data set 1: Education Levels - Types of variables

Character:

- SA1 7 digit code

The SA1 7 digit code is a unique identifier and should not be summed or averaged. Though the SA1 observations are made up of numbers (e.g. 2115615), this is actually a categorical variable. The number strings have no arithmetical value. They are unique identifiers assigned to each geographical area. Therefore the class of character is correct.

Numeric:

- Postgraduate Degree Level
  - Doctoral Degree Level
  - Masters Degree Level
- Graduate Diploma and Graduate Certificate Level
- Bachelor Degree Level
- Advanced Diploma and Diploma Level
- Certificate III & IV Level
- Secondary Education Years 10 and above
- Certificate I &II Level
- Secondary Education Years 9 and below
- Supplementary Codes
- Not stated
- Not applicable
- Total

The values in these columns are a numeric count of people in each statistical area that have achieved a certain level of education. The levels of education are suitable for creating an ordered factor variable as there is a clear, sensible ordering of each level of education. Before this can be done, the data must be gathered together in a single column. This will be done in the next step.

Data set 2: Geographic Look Up - Types of variables

Integer:

- SA1_7DIG16

Though the SA1_7DIG16 (SA1) observations are made up of numbers (e.g. 2115615), this is actually a categorical variable. The number strings have no arithmetical value. They are unique identifiers assigned to each geographical area.

Character:

- LGA_NAME17
- MetroMelbourne

The values in this column are categorical data so character is the correct data type.

Please note that there is one LGA name that is 'Unincorporated Vic'. This is a legitimate value and should not be treated as a missing value.

Hide

```
#Education Levels
kable(head(education))
```

HEAP - 1 Digit Level	X__1	Postgraduate Degree Level	Graduate Diploma and Graduate Certificate Level	Bachelor Degree Level	Advanced Diploma and Diploma Level	Certificate III & IV Level	Secondary Education - Years 10 and above	Certificate I & II Level	Secondary Education - Years 9 and below	Supplementary Codes	Not stated	Not applicable
NA	SA1 (UR)	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
NA	2100101	22	12	44	31	61	111	0	25	10	32	82
NA	2100102	0	0	24	16	23	58	0	52	3	21	22
NA	2100105	9	5	30	31	46	86	0	19	11	31	36
NA	2100106	13	16	69	31	75	154	0	37	8	48	110
NA	2100107	12	8	45	22	54	94	0	51	7	34	51

Hide

```
kable(tail(education))
```

HEAP - 1 Digit Level	X__1	Postgraduate Degree Level	Graduate Diploma and Graduate Certificate Level	Bachelor Degree Level	Advanced Diploma and Diploma Level	Certificate III & IV Level	Secondary Education - Years 10 and above	Certificate I & II Level	Secondary Education - Years 9 and below	Supplementary Codes	Not stated	No applicabl
NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	N
NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	N
NA	Cells in this table have been randomly adjusted to avoid the release of confidential data. No reliance should be placed on small cells.	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	N
NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	N
NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	N
NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	N

Hide

```
dim(education)
```

```
[1] 14081    14
```

Hide

```
str(education)
```

```
Classes □tbl_df□, □tbl□ and 'data.frame':   14081 obs. of  14 variables:
 $ HEAP - 1 Digit Level                          : logi  NA NA NA NA NA NA ...
 $ X__1                                          : chr  "SA1 (UR)" "2100101" "2100102" "2100105" ...
 $ Postgraduate Degree Level                     : num  NA 22 0 9 13 12 20 12 14 10 ...
 $ Graduate Diploma and Graduate Certificate Level: num  NA 12 0 5 16 8 17 23 13 16 ...
 $ Bachelor Degree Level                         : num  NA 44 24 30 69 45 67 48 63 37 ...
 $ Advanced Diploma and Diploma Level            : num  NA 31 16 31 31 22 37 41 47 21 ...
 $ Certificate III & IV Level                    : num  NA 61 23 46 75 54 96 74 56 54 ...
 $ Secondary Education - Years 10 and above      : num  NA 111 58 86 154 94 185 135 138 103 ...
 $ Certificate I & II Level                      : num  NA 0 0 0 0 0 0 0 0 0 ...
 $ Secondary Education - Years 9 and below       : num  NA 25 52 19 37 51 54 45 29 23 ...
 $ Supplementary Codes                           : num  NA 10 3 11 8 7 17 16 9 4 ...
 $ Not stated                                    : num  NA 32 21 31 48 34 38 47 19 19 ...
 $ Not applicable                                : num  NA 82 22 36 110 51 155 120 114 93 ...
 $ Total                                         : num  NA 430 217 297 564 365 681 571 504 380 ...
```

Hide

```
names(education)
```

```
 [1] "HEAP - 1 Digit Level"                          "X__1"                                          "Postgraduate Degre
e Level"
 [4] "Graduate Diploma and Graduate Certificate Level" "Bachelor Degree Level"                         "Advanced Diploma a
nd Diploma Level"
 [7] "Certificate III & IV Level"                    "Secondary Education - Years 10 and above"      "Certificate I & II
Level"
[10] "Secondary Education - Years 9 and below"       "Supplementary Codes"                           "Not stated"

[13] "Not applicable"                                "Total"
```

Hide

```
class(education)
```

```
[1] "tbl_df"     "tbl"        "data.frame"
```

Brief summary of Education Data

- First column and first rows contain no data and need to be trimmed
- Extra rows at bottom of the data frame with no data and need to be trimmed
- There are 14 columns and 14,081 rows
    - One logical class column with no data
    - One character column with no header
    - 12 numeric columns, containing counts of people with varying levels of educational qualifications in each statistical area
- Column name #X__1 will need to be changed to SA1
- Levels of educational qualifications make this data a good candidate for gathering into a single column as they all pertain to the same type of information
- Class of object is a data frame as expected

Hide

```
#Geographical Data
kable(head(geo_lookup))
```

SA1_7DIG16	LGA_NAME17	MetroMelbourne
2115615	Manningham (C)	Greater Melbourne
2115616	Manningham (C)	Greater Melbourne
2115617	Manningham (C)	Greater Melbourne
2115618	Manningham (C)	Greater Melbourne
2115619	Manningham (C)	Greater Melbourne
2115620	Manningham (C)	Greater Melbourne

Hide

```
kable(tail(geo_lookup))
```

SA1_7DIG16	LGA_NAME17	MetroMelbourne
2110407	NA	Rest of Vic.
2110407	NA	Rest of Vic.
2110407	NA	Rest of Vic.
2148014	NA	Rest of Vic.
2140636	Murray River (A)	Rest of Vic.
2141338	Murray River (A)	Rest of Vic.

Hide

```
dim(geo_lookup)
```

```
[1] 14382     3
```

Hide

```
str(geo_lookup)
```

```
Classes □tbl_df□, □tbl□ and 'data.frame':   14382 obs. of  3 variables:
 $ SA1_7DIG16    : int  2115615 2115616 2115617 2115618 2115619 2115620 2115621 2115622 2115623 2115624 ...
 $ LGA_NAME17    : chr  "Manningham (C)" "Manningham (C)" "Manningham (C)" "Manningham (C)" ...
 $ MetroMelbourne: chr  "Greater Melbourne" "Greater Melbourne" "Greater Melbourne" "Greater Melbourne" ...
 - attr(*, "spec")=List of 2
  ..$ cols   :List of 3
  .. ..$ SA1_7DIG16    : list()
  .. .. ..- attr(*, "class")= chr  "collector_integer" "collector"
  .. ..$ LGA_NAME17    : list()
  .. .. ..- attr(*, "class")= chr  "collector_character" "collector"
  .. ..$ MetroMelbourne: list()
  .. .. ..- attr(*, "class")= chr  "collector_character" "collector"
  ..$ default: list()
  .. ..- attr(*, "class")= chr  "collector_guess" "collector"
  ..- attr(*, "class")= chr "col_spec"
```

Hide

```
names(geo_lookup)
```

```
[1] "SA1_7DIG16"     "LGA_NAME17"     "MetroMelbourne"
```

Hide

```
class(geo_lookup)
```

```
[1] "tbl_df"     "tbl"        "data.frame"
```

Brief summary of Geographical Data

- There are no extra columns or rows at the top of the data to trim
- There are no extra columns or rows at the end of the data to trim. A few missing values are visible
- There are 3 columns and 14,382 rows
    - More rows than the education data
    - Will consider this when choosing the type of join to use
- First column is integer SA1 values - this should be a character variable
- Second column is character LGA names
- Third column is character names for Melbourne Metropolitan Area
- Column names can be more neatly given for SA1 and LGA variables
- Class of object is a data frame as expected

Hide

```
#Convert the SA1 data to character variables
geo_lookup <- transform(geo_lookup, SA1_7DIG16 = as.character(SA1_7DIG16))
```

Tidy & Manipulate Data I

Firstly, some trimming of extraneous rows and columns, which did not contain any data, was undertaken to clean up the data set and make it more usable for analysis. Similarly, the column name for the education data was changed from the generic X__1 to SA1 to make it easier to understand. A totals row was specified so we could subset the data to ensure we excluded any of the extraneous rows and the totals row at the bottom of the data set.

Hide

```
#Removing first row and column
education <- education[-1,-1]
#The first column has no header. Setting column header to SA1
colnames(education)[1] <- "SA1"
#Add a totals row at the bottom of the data frame
totalrow <- which(education$SA1 == 'Total')
#Removing empty rows at end of data frame
education <- education[1:totalrow - 1,]
kable(tail(education))
```

SA1	Postgraduate Degree Level	Graduate Diploma and Graduate Certificate Level	Bachelor Degree Level	Advanced Diploma and Diploma Level	Certificate III & IV Level	Secondary Education - Years 10 and above	Certificate I & II Level	Secondary Education - Years 9 and below	Supplementary Codes	Not stated	Not applicable	Total
2148034	5	8	32	22	41	66	0	17	4	25	57	276
2148035	16	10	69	41	72	115	0	43	12	52	75	509

SA1	Postgraduate Degree Level	Graduate Diploma and Graduate Certificate Level	Bachelor Degree Level	Advanced Diploma and Diploma Level	Certificate III & IV Level	Secondary Education - Years 10 and above	Certificate I & II Level	Secondary Education - Years 9 and below	Supplementary Codes	Not stated	Not applicable	Total
2979991	0	0	0	0	0	0	0	0	0	0	0	0
2979992	0	0	4	4	9	3	0	0	0	5	4	24
2979993	0	0	0	0	6	0	0	0	0	0	0	4
2949999	237	93	792	403	778	1679	15	479	212	2133	747	7572

Tidy Data Principles:

1. Each variable must have its own column
2. Each observation must have its own row
3. Each value must have its own cell

Data set 1: Education Levels

The education data set does not conform to the the tidy data principles. Each variable should have its own column. Population count (or number of people) is a variable, but it is not contained within its own column and spread out across the data set. Though we have many columns, apart from SA1, they all refer to the variable education level. Therefore, we should have three columns for the three variables, SA1, Education Level and Number of People.

To address this, the gather function has been applied to all the education levels to bring them into a single column called Ed_level. Now the data set is comprised of three attributes, SA1, level of education and a count of people in each SA1 that meet a particular level of education.

As mentioned in the previous section, the education level variable is a good candidate for creating an ordered factor variable as there is a naturally ordering of education levels. Education levels were labelled and ordered. Note that some observations had “Not Stated” and “Not Applicable” for education level. We made the decision to include these, ordered last in the factor ordering. We didn’t want to exclude them as this may introduce bias to the data.

Data set 2: Geographic Look Up

This data set conforms to the tidy data principles.

Education Data

Hide

```
#Drop the totals column before gathering education level as this does not belong as an observation
edu_droptot <- education[,-13]
names(edu_droptot)
```

```
 [1] "SA1"                                       "Postgraduate Degree Level"                 "Graduate Diploma a
nd Graduate Certificate Level"
 [4] "Bachelor Degree Level"                     "Advanced Diploma and Diploma Level"        "Certificate III &
IV Level"
 [7] "Secondary Education - Years 10 and above"  "Certificate I & II Level"                  "Secondary Educatio
n - Years 9 and below"
[10] "Supplementary Codes"                       "Not stated"                                "Not applicable"
```

Hide

```
#Gathering column variables into a single row
edu_tidy <- gather(edu_droptot, "Ed_level", "People", 2:length(edu_droptot))
kable(head(edu_tidy))
```

SA1	Ed_level	People
2100101	Postgraduate Degree Level	22
2100102	Postgraduate Degree Level	0
2100105	Postgraduate Degree Level	9
2100106	Postgraduate Degree Level	13
2100107	Postgraduate Degree Level	12
2100108	Postgraduate Degree Level	20

Hide

```
#Set education level to an ordered factor
edu_tidy <- edu_tidy %>% mutate(Ed_level = factor(Ed_level, levels = c("Postgraduate Degree Level", "Graduate Diploma and Gr
aduate Certificate Level",
                                                                        "Bachelor Degree Level", "Advanced Diploma and Diplom
a Level", "Certificate III & IV Level",                                         "Secondary Education - Years 10 and above",
 "Certificate I & II Level", "Secondary Education - Years 9 and below",
                                                                        "Supplementary Codes", "Not stated", "Not applicable"
),
                                                    labels = c("Postgraduate Degree Level", "Graduate Diploma and Graduate Cer
tificate Level",
                                                               "Bachelor Degree Level", "Advanced Diploma and Diploma Level",
"Certificate III & IV Level",
                                                               "Secondary Education - Years 10 and above", "Certificate I & II
 Level", "Secondary Education - Years 9 and below",
                                                               "Supplementary Codes", "Not stated", "Not applicable"),
                                                    ordered = TRUE))
```

```
package <U+393C><U+3E31>bindrcpp<U+393C><U+3E32> was built under R version 3.4.4
```

Joining the datasets together

Now that the Education data is tidy, we have joined it with the Geographic Look Up data, which provides a concordance between LGA and SA1 geographic areas. Now we are able to access how many people in each LGA achieved each particular level of education. A left join was chosen, prioritising the education data on the left hand side as this is the variable of primary interest.

The LGA variable in the LGA Lookup data set is called “LGA_NAME17” which is the metadata label used by the source (ABS). The column name has been changed to LGA which is simpler for users to read.

Hide

```
#Left Join Education Data to geographical area lookup to match observations to their LGA.
edu_tidy_join <- edu_tidy %>% left_join(geo_lookup[,c("SA1_7DIG16", "LGA_NAME17", "MetroMelbourne")], by = c("SA1" = "SA1_7D
IG16"))
#Change name from LGA_NAME17 to LGA for neatness.
names(edu_tidy_join)[4] <- "LGA"
kable(head(edu_tidy_join))
```

SA1	Ed_level	People	LGA	MetroMelbourne
2100101	Postgraduate Degree Level	22	Ballarat (C)	Rest of Vic.
2100102	Postgraduate Degree Level	0	Ballarat (C)	Rest of Vic.
2100105	Postgraduate Degree Level	9	Ballarat (C)	Rest of Vic.
2100106	Postgraduate Degree Level	13	Ballarat (C)	Rest of Vic.
2100107	Postgraduate Degree Level	12	Ballarat (C)	Rest of Vic.
2100108	Postgraduate Degree Level	20	Ballarat (C)	Rest of Vic.

Hide

```
dim(edu_tidy_join)
```

```
[1] 158246      5
```

Hide

```
str(edu_tidy_join)
```

```
Classes □tbl_df□, □tbl□ and 'data.frame':   158246 obs. of  5 variables:
 $ SA1           : chr  "2100101" "2100102" "2100105" "2100106" ...
 $ Ed_level      : Ord.factor w/ 11 levels "Postgraduate Degree Level"<..: 1 1 1 1 1 1 1 1 1 1 ...
 $ People        : num  22 0 9 13 12 20 12 14 10 9 ...
 $ LGA           : chr  "Ballarat (C)" "Ballarat (C)" "Ballarat (C)" "Ballarat (C)" ...
 $ MetroMelbourne: chr  "Rest of Vic." "Rest of Vic." "Rest of Vic." "Rest of Vic." ...
```

Hide

```
kable(tail(edu_tidy_join))
```

SA1	Ed_level	People	LGA	MetroMelbourne
2148034	Not applicable	57	Warrnambool (C)	Rest of Vic.
2148035	Not applicable	75	Warrnambool (C)	Rest of Vic.
2979991	Not applicable	0	NA	NA
2979992	Not applicable	4	NA	NA
2979993	Not applicable	0	NA	NA
2949999	Not applicable	747	NA	NA

Hide

```
summary(edu_tidy_join)
```

```
     SA1                                                Ed_level         People           LGA            MetroMelbour
ne
 Length:158246      Postgraduate Degree Level                     :14386   Min.   :   0.0   Length:158246      Length:15824
6
 Class :character   Graduate Diploma and Graduate Certificate Level:14386   1st Qu.:   8.0   Class :character   Class :chara
cter
 Mode  :character   Bachelor Degree Level                         :14386   Median :  26.0   Mode  :character   Mode  :chara
cter
                    Advanced Diploma and Diploma Level            :14386   Mean   :  37.9
                    Certificate III & IV Level                    :14386   3rd Qu.:  54.0
                    Secondary Education - Years 10 and above      :14386   Max.   :2133.0
                    (Other)                                       :71930
```

Tidy & Manipulate Data II

Now that we have Education Levels by LGA from joining the two data sets, we have grouped observations by LGA first, then by Education Level and then calculated a summary of the number of people which each qualification for each LGA. Note that additonal demonstration of variable creation occur in later sections of the report.

Hide

```
#Grouping by LGA
edu_by_LGA <- edu_tidy_join %>% group_by(LGA, Ed_level) %>% summarise(PeopleSum = sum(People))
kable(head(edu_by_LGA))
```

LGA	Ed_level	PeopleSum
Alpine (S)	Postgraduate Degree Level	223
Alpine (S)	Graduate Diploma and Graduate Certificate Level	297
Alpine (S)	Bachelor Degree Level	1257
Alpine (S)	Advanced Diploma and Diploma Level	1092
Alpine (S)	Certificate III & IV Level	2230
Alpine (S)	Secondary Education - Years 10 and above	3405

Hide

```
glimpse(edu_by_LGA)
```

```
Observations: 902
Variables: 3
$ LGA       <chr> "Alpine (S)", "Alpine (S)", "Alpine (S)", "Alpine (S)", "Alpine (S)", "Alpine (S)", "Alpine (S)", "Alpine
(S)", "Alpine (S)", "Alpine (S)", "Alpine (S)", "Ararat (RC)"...
$ Ed_level  <ord> Postgraduate Degree Level, Graduate Diploma and Graduate Certificate Level, Bachelor Degree Level, Advance
d Diploma and Diploma Level, Certificate III & IV Level, Sec...
$ PeopleSum <dbl> 223, 297, 1257, 1092, 2230, 3405, 3, 1128, 336, 1573, 2115, 111, 157, 813, 801, 1681, 3027, 3, 1295, 300,
2024, 1930, 2714, 2268, 10556, 6940, 14614, 25723, 37, 8431,...
```

Scan I

The Education Levels by LGA data has been scanned for missing values (denoted by NA). 11 missing values where identfied in the LGA column using colSums. This makes sense as we have 11 attributes for education level so any counts of people where location data was not available would be recorded as NA. Since this is not a numeric variable, the NAs will not cause problems with computations but we may want to remove them anyway since they do not provide us with the information we need. 11 out of 902 records is very small but each record is actually a count of people so if we exclude NAs by using complete.cases, we could end up excluding a large number of people. To determine if this number signficant, we divided the total number of people with missing values by the total number of people for each level of education. For every level, the proportion of NAs was insignificant, with the largest proportion of NAs comprising only 1.1% of total people in the Certificate I & II Level bracket. Therefore, we decided there was a low risk of bias if NAs were removed. We removed the records with missing LGA data using complete.cases. Another method we could have used to reduce the number of NAs is to re-examine the original Geographic Lookup table and identified why there were missing values present initially. Because this is a categorical value, we could also impute missing values using the mode of the LGA data. We checked for any “Not a Number” values in the PeopleSum column as this is a numerical variable. None were found.

Hide

```
#Check the Education by LGA data frame for missing values using which(is.na())
dim(edu_by_LGA)
```

```
[1] 902   3
```

Hide

```
which(is.na(edu_by_LGA))
```

```
 [1] 892 893 894 895 896 897 898 899 900 901 902
```

Hide

```
colSums(is.na(edu_by_LGA))
```

```
      LGA  Ed_level PeopleSum 
       11         0         0 
```

Hide

```
#Calculating the ratio of NAs for each level of education
edu_by_LGA_NA <- edu_by_LGA[!complete.cases(edu_by_LGA),]
edu_by_LGA_Totals <- edu_tidy_join %>% group_by(Ed_level) %>% summarise(PeopleSum = sum(People))
edu_by_LGA_ratio <- edu_by_LGA %>% mutate(ratio = paste(round(edu_by_LGA_NA$PeopleSum/edu_by_LGA_Totals$PeopleSum*100, 2),
"%")) %>% select(LGA, Ed_level, ratio)
edu_by_LGA_ratio
```

LGA <chr>	Ed_level <ord>	ratio <chr>
Alpine (S)	Postgraduate Degree Level	0.1 %
Alpine (S)	Graduate Diploma and Graduate Certificate Level	0.1 %
Alpine (S)	Bachelor Degree Level	0.11 %
Alpine (S)	Advanced Diploma and Diploma Level	0.1 %
Alpine (S)	Certificate III & IV Level	0.12 %
Alpine (S)	Secondary Education - Years 10 and above	0.12 %
Alpine (S)	Certificate I & II Level	1.1 %
Alpine (S)	Secondary Education - Years 9 and below	0.11 %
Alpine (S)	Supplementary Codes	0.13 %
Alpine (S)	Not stated	0.45 %

Hide

```
#Remove NAs
edu_by_LGA_complete <- edu_by_LGA[complete.cases(edu_by_LGA), ]
which(is.na(edu_by_LGA_complete))
```

```
integer(0)
```

Hide

```
#Check for the special value
which(is.nan(edu_by_LGA$PeopleSum))
```

```
integer(0)
```

We checked for inconsistencies and obvious errors by checking the data against a validation rule that no population counts were negative or greater than the total population of Victorial (6.5 million) as this would inicate a signficant data error. We also inspected the data graphically and no inconsistencies in the data were detected.

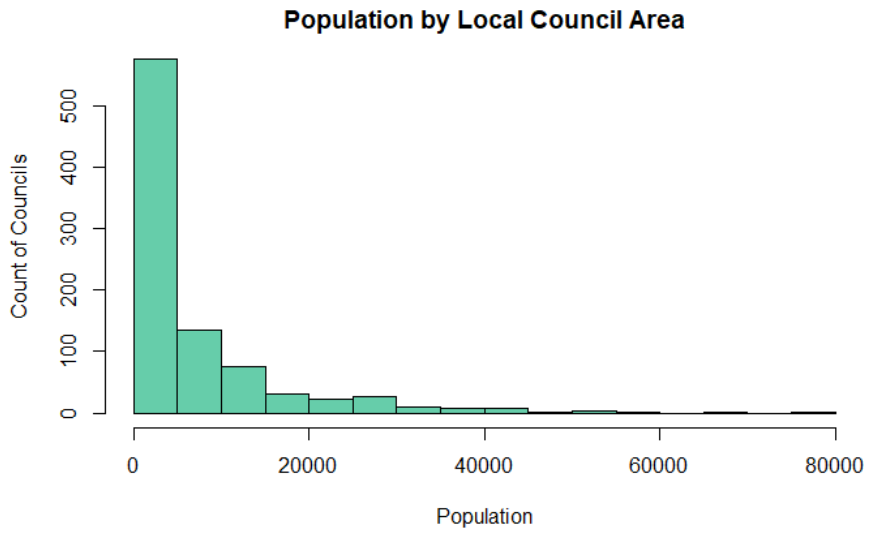
Hide

```
#check none of the sum totals are negative or greater than the population of victoria
rule1 <- editrules::editset(c('PeopleSum >= 0', 'PeopleSum < 6500000'))
any(violatedEdits(rule1, edu_by_LGA_complete)) # all FALSE
```

```
[1] FALSE
```

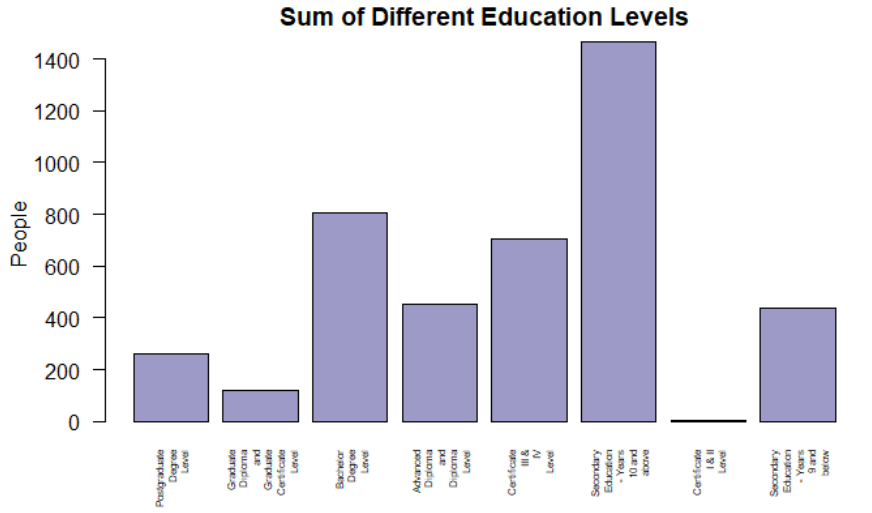
Hide

```
#Checking for inconsistencies
hist(edu_by_LGA$PeopleSum,
     main = "Population by Local Council Area",
     xlab = "Population",
     ylab = "Count of Councils",
     col = "#66CDAA",
     breaks = 15)
```



Hide

```
sum_edu <- edu_by_LGA %>% group_by(Ed_level) %>% summarise(PopK = sum(PeopleSum)/1000)
wrapped <- function(strings, width) vapply(strings, function(s)paste(collapse="\n", strwrap(s, width)), FUN.VALUE="", USE.NA
MES=FALSE) # from (http://r.789695.n4.nabble.com/Wrap-names-arg-text-in-barplot-td4593439.html)
barplot(sum_edu$PopK[1:8], names.arg = wrapped(sum_edu$Ed_level[1:8],8),
        las=2, cex.names = 0.5, col = "#9e9ac8",
        main = "Sum of Different Education Levels",
        ylab = "People")
```



Scan II

To scan for outliers, we decided to focus only on postgraduate (including graduate diploma and graduate certificate level) and bachelor degree level observations. We looked at two forms of the data, the first being the total count of postgraduate and undergraduate for each SA1 subsection. The second was the proportion of these counts against the totals found with their respective SA1.

We visualized the distribution of the data using the base R boxplot function to find the extent of the outliers. We then reassigned this function to a vector and observed the out attribute to find the number of outliers. Notably, the total count compared to the proportion outliers is the dramatic decrease when comparing the totals by SA1 region as opposed to the percentages.

Once the outliers were identified for each post and undergraduate education level, we used a capping function from Dr Anil Dolgun, to Windsorise and reassign to vectors in order to visualize using the base R boxplot function.

Hide

```
#Mutate postgraduate and undergradiate into one dataset
edu_uni_tot <- education %>%  mutate(Post = (education$`Postgraduate Degree Level` + education$`Graduate Diploma and Graduat
e Certificate Level`),
                                     UnderGrad = education$`Bachelor Degree Level`) %>%
                              select(SA1, Post, UnderGrad)
kable(head(edu_uni_tot))
```

SA1	Post	UnderGrad
2100101	34	44
2100102	0	24
2100105	14	30

SA1	Post	UnderGrad
2100106	29	69
2100107	20	45
2100108	37	67

Hide

```
edu_uni_tot2 <- edu_uni_tot %>% left_join(geo_lookup[, c('SA1_7DIG16','LGA_NAME17')], by = c('SA1' = 'SA1_7DIG16'))
kable(head(edu_uni_tot2))
```

SA1	Post	UnderGrad	LGA_NAME17
2100101	34	44	Ballarat (C)
2100102	0	24	Ballarat (C)
2100105	14	30	Ballarat (C)
2100106	29	69	Ballarat (C)
2100107	20	45	Ballarat (C)
2100108	37	67	Ballarat (C)

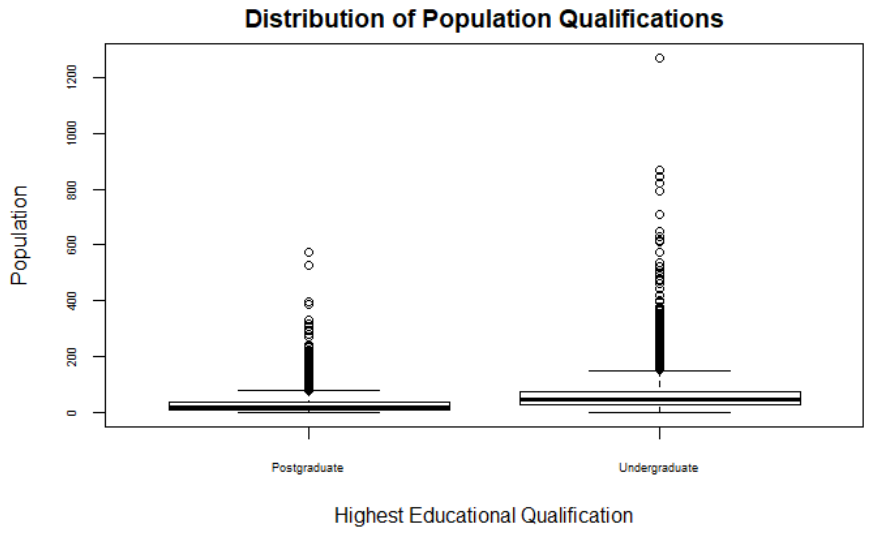
Hide

```
glimpse(edu_uni_tot2)
```

```
Observations: 14,386
Variables: 4
$ SA1        <chr> "2100101", "2100102", "2100105", "2100106", "2100107", "2100108", "2100109", "2100110", "2100111", "21001
12", "2100113", "2100114", "2100115", "2100117", "2100118", ...
$ Post       <dbl> 34, 0, 14, 29, 20, 37, 35, 27, 26, 19, 20, 29, 21, 64, 18, 19, 31, 44, 66, 51, 40, 60, 31, 27, 36, 32, 2
6, 41, 64, 0, 0, 46, 22, 35, 28, 42, 28, 31, 27, 16, 33, 13, ...
$ UnderGrad  <dbl> 44, 24, 30, 69, 45, 67, 48, 63, 37, 50, 42, 98, 42, 141, 39, 44, 77, 75, 179, 106, 54, 68, 42, 68, 51, 8
0, 60, 61, 104, 0, 0, 94, 46, 74, 61, 81, 53, 53, 70, 20, 44,...
$ LGA_NAME17 <chr> "Ballarat (C)", "Ballarat (C)", "Ballarat (C)", "Ballarat (C)", "Ballarat (C)", "Ballarat (C)", "Ballarat
(C)", "Ballarat (C)", "Ballarat (C)", "Ballarat (C)", "Ball...
```

Hide

```
#Visualise
box_tot <- boxplot(edu_uni_tot2[,2:3], main  = 'Distribution of Population Qualifications',
                        xlab = "Highest Educational Qualification", cex.axis = 0.55,
                        names = c("Postgraduate", "Undergraduate"),
                        ylab = "Population")
```



Hide

```
length(box_tot$out)
```

```
[1] 986
```

There are 986 outliers for both postgraduate and undergraduate qualifications

Hide

```
#Mutate post and grad into one dataset (using proportion of the total)
edu_uni_prop <- education %>%  mutate(Post = (education$`Postgraduate Degree Level` + education$`Graduate Diploma and Gradua
te Certificate Level`)/education$Total,
                                      UnderGrad = education$`Bachelor Degree Level`/education$Total) %>%
                                select(SA1, Post, UnderGrad)
kable(head(edu_uni_prop))
```

SA1	Post	UnderGrad
2100101	0.0790698	0.1023256
2100102	0.0000000	0.1105991
2100105	0.0471380	0.1010101
2100106	0.0514184	0.1223404
2100107	0.0547945	0.1232877
2100108	0.0543319	0.0983847

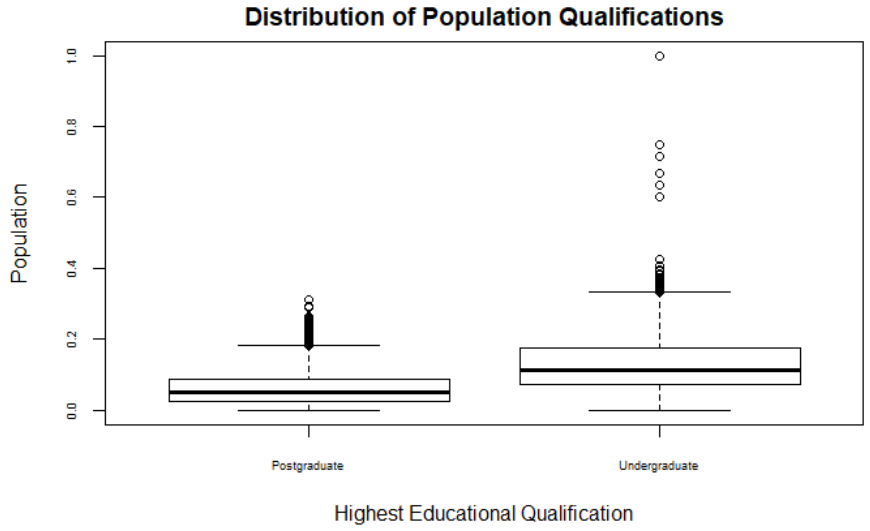
Hide

```
glimpse(edu_uni_prop)
```

```
Observations: 14,073
Variables: 3
$ SA1       <chr> "2100101", "2100102", "2100105", "2100106", "2100107", "2100108", "2100109", "2100110", "2100111", "210011
2", "2100113", "2100114", "2100115", "2100117", "2100118", "...
$ Post      <dbl> 0.07906977, 0.00000000, 0.04713805, 0.05141844, 0.05479452, 0.05433186, 0.06129597, 0.05357143, 0.0684210
5, 0.04203540, 0.05524862, 0.05301645, 0.03703704, 0.07609988...
$ UnderGrad <dbl> 0.10232558, 0.11059908, 0.10101010, 0.12234043, 0.12328767, 0.09838473, 0.08406305, 0.12500000, 0.0973684
2, 0.11061947, 0.11602210, 0.17915905, 0.07407407, 0.16765755...
```

Hide

```
#Spread and then mutate the data
box_tot_post <- boxplot(edu_uni_prop[,2:3], main  = 'Distribution of Population Qualifications',
                        xlab = "Highest Educational Qualification", cex.axis = 0.55,
                        names = c("Postgraduate", "Undergraduate"),
                        ylab = "Population")
```



Hide

```
length(box_tot_post$out)
```

```
[1] 296
```

Converting the data from absolute counts to proportions drastically reduces the count of outliers to 346

Handling LGA Outliers

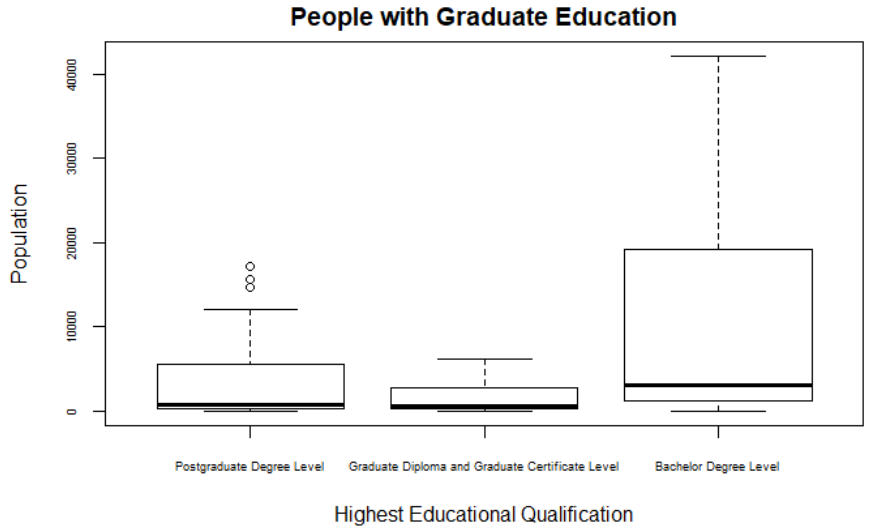
Hide

```
# Postgraduate and undergraduate count
edu_uni_lga <- edu_by_LGA %>% filter(Ed_level <= 'Bachelor Degree Level')
summary(edu_uni_lga)
```

```
     LGA                                                 Ed_level    PeopleSum
 Length:246         Postgraduate Degree Level                :82   Min.   :    0
 Class :character   Graduate Diploma and Graduate Certificate Level:82   1st Qu.:  385
 Mode  :character   Bachelor Degree Level                   :82   Median : 1506
                    Advanced Diploma and Diploma Level       : 0   Mean   : 4816
                    Certificate III & IV Level               : 0   3rd Qu.: 4515
                    Secondary Education - Years 10 and above : 0   Max.   :42145
                    (Other)                                  : 0
```

Hide

```
# Refactor levels of education
edu_uni_lga$Ed_level <- factor(edu_uni_lga$Ed_level,
                               labels = c("Postgraduate Degree Level", "Graduate Diploma and Graduate Certificate Level",
                                          "Bachelor Degree Level"),
                               levels = c("Postgraduate Degree Level", "Graduate Diploma and Graduate Certificate Level",
                                          "Bachelor Degree Level"))
boxplot(edu_uni_lga$PeopleSum~edu_uni_lga$Ed_level,
        main = "People with Graduate Education",
        xlab = "Highest Educational Qualification",
        cex.axis = 0.55, ylab = "Population")
```



Hide

```
#Mutation of the post and grad into one (total counts)
glimpse(education)
```

```
Observations: 14,073
Variables: 13
$ SA1                                          <chr> "2100101", "2100102", "2100105", "2100106", "2100107", "2100108",
"2100109", "2100110", "2100111", "2100112", "2100113", "2100...
$ `Postgraduate Degree Level`                  <dbl> 22, 0, 9, 13, 12, 20, 12, 14, 10, 9, 6, 11, 14, 44, 15, 3, 22, 28,
44, 38, 24, 33, 20, 16, 21, 20, 14, 23, 46, 0, 0, 17, 12, 1...
$ `Graduate Diploma and Graduate Certificate Level` <dbl> 12, 0, 5, 16, 8, 17, 23, 13, 16, 10, 14, 18, 7, 20, 3, 16, 9, 16,
22, 13, 16, 27, 11, 11, 15, 12, 12, 18, 18, 0, 0, 29, 10, 16...
$ `Bachelor Degree Level`                      <dbl> 44, 24, 30, 69, 45, 67, 48, 63, 37, 50, 42, 98, 42, 141, 39, 44, 7
7, 75, 179, 106, 54, 68, 42, 68, 51, 80, 60, 61, 104, 0, 0, ...
$ `Advanced Diploma and Diploma Level`         <dbl> 31, 16, 31, 31, 22, 37, 41, 47, 21, 37, 32, 44, 48, 68, 23, 25, 4
0, 54, 107, 58, 42, 59, 25, 39, 23, 33, 28, 35, 56, 0, 0, 40,...
$ `Certificate III & IV Level`                 <dbl> 61, 23, 46, 75, 54, 96, 74, 56, 54, 66, 46, 67, 93, 81, 33, 86, 8
2, 83, 206, 139, 59, 50, 29, 57, 35, 65, 20, 41, 39, 0, 0, 55...
$ `Secondary Education - Years 10 and above`   <dbl> 111, 58, 86, 154, 94, 185, 135, 138, 103, 147, 89, 140, 131, 213,
90, 147, 147, 166, 278, 204, 106, 125, 97, 108, 88, 141, 174...
$ `Certificate I & II Level`                   <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...
$ `Secondary Education - Years 9 and below`    <dbl> 25, 52, 19, 37, 51, 54, 45, 29, 23, 28, 21, 37, 32, 33, 23, 25, 1
9, 24, 61, 26, 20, 28, 19, 28, 21, 37, 70, 18, 28, 0, 0, 32, ...
$ `Supplementary Codes`                        <dbl> 10, 3, 11, 8, 7, 17, 16, 9, 4, 5, 7, 17, 16, 17, 4, 9, 7, 16, 18,
7, 12, 17, 9, 7, 14, 17, 3, 8, 7, 0, 0, 14, 10, 5, 8, 5, 5, ...
$ `Not stated`                                 <dbl> 32, 21, 31, 48, 34, 38, 47, 19, 19, 23, 19, 28, 19, 43, 8, 22, 28,
23, 67, 57, 22, 38, 29, 33, 27, 32, 16, 20, 54, 0, 0, 35, 3...
$ `Not applicable`                             <dbl> 82, 22, 36, 110, 51, 155, 120, 114, 93, 78, 87, 94, 162, 181, 73,
139, 202, 230, 360, 208, 145, 128, 59, 70, 48, 79, 96, 81, 8...
$ Total                                        <dbl> 430, 217, 297, 564, 365, 681, 571, 504, 380, 452, 362, 547, 567, 8
41, 315, 526, 629, 718, 1339, 859, 487, 575, 335, 436, 356, ...
```

Hide

```
edu_uni_tot_lga <- education %>%  mutate(Post = (education$`Postgraduate Degree Level` + education$`Graduate Diploma and Gra
duate Certificate Level`),
                                         UnderGrad = education$`Bachelor Degree Level`) %>% select(SA1, Post, UnderGrad)
kable(head(edu_uni_tot))
```

SA1	Post	UnderGrad
2100101	34	44
2100102	0	24
2100105	14	30
2100106	29	69
2100107	20	45
2100108	37	67

Hide

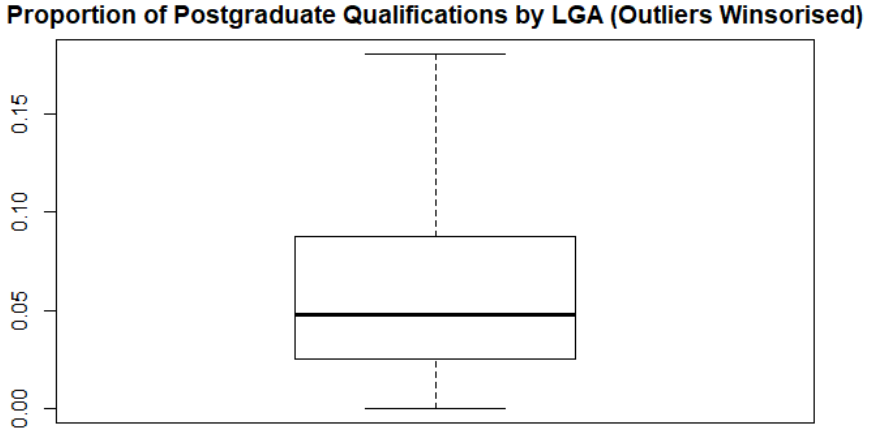
```
edu_uni_tot2 <- edu_uni_tot %>% left_join(geo_lookup[, c('SA1_7DIG16','LGA_NAME17')], by = c('SA1' = 'SA1_7DIG16'))
kable(head(edu_uni_tot2))
```

SA1	Post	UnderGrad	LGA_NAME17
2100101	34	44	Ballarat (C)
2100102	0	24	Ballarat (C)
2100105	14	30	Ballarat (C)
2100106	29	69	Ballarat (C)
2100107	20	45	Ballarat (C)
2100108	37	67	Ballarat (C)

Handling Outliers

Hide

```
# Replace NAs with 0 values in the proportions (NA's occured due to 0/0 mutate)
edu_uni_prop[which(is.na(edu_uni_prop$Post)),] <- 0
edu_uni_prop[which(is.na(edu_uni_prop$UnderGrad)),] <- 0
cap <- function(x){
  quantiles <- quantile( x, c(.05, 0.25, 0.75, .95 ) )
  x[ x < quantiles[2] - 1.5*IQR(x) ] <- quantiles[1]
  x[ x > quantiles[3] + 1.5*IQR(x) ] <- quantiles[4]
  x
} # from Module 6 of MATH 2349, credit to Dr Anil Dolgun
post_cap <- cap(edu_uni_prop$Post)
boxplot(post_cap, main = 'Proportion of Postgraduate Qualifications by LGA (Outliers Winsorised)')
```



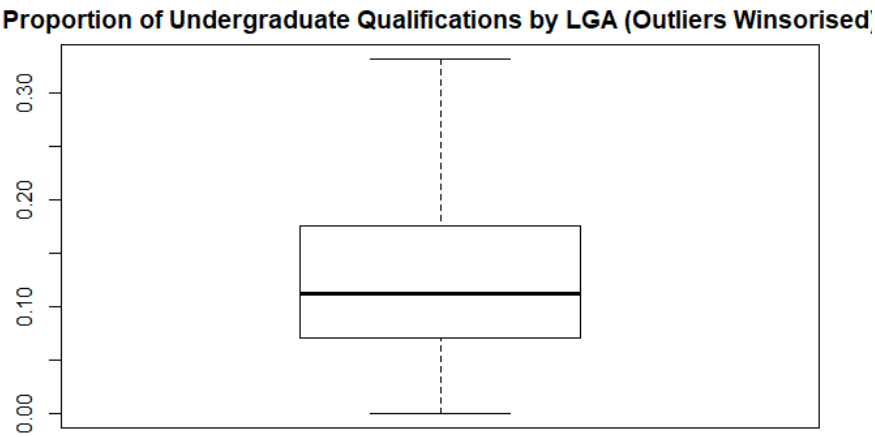
Hide

```
summary(post_cap)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
0.00000 0.02482 0.04781 0.05905 0.08728 0.18089 
```

Hide

```
under_cap <- cap(edu_uni_prop$UnderGrad)
boxplot(under_cap, main = 'Proportion of Undergraduate Qualifications by LGA (Outliers Winsorised)')
```



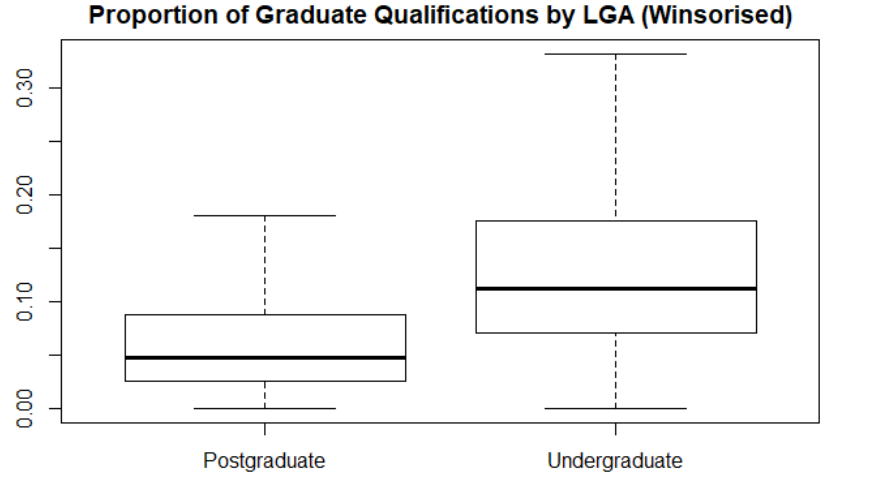
Hide

```
summary(under_cap)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
 0.0000  0.0709  0.1118  0.1269  0.1756  0.3325 
```

Hide

```
boxplot(post_cap, under_cap, main = 'Proportion of Graduate Qualifications by LGA (Winsorised)',
        names = c('Postgraduate', 'Undergraduate'))
```

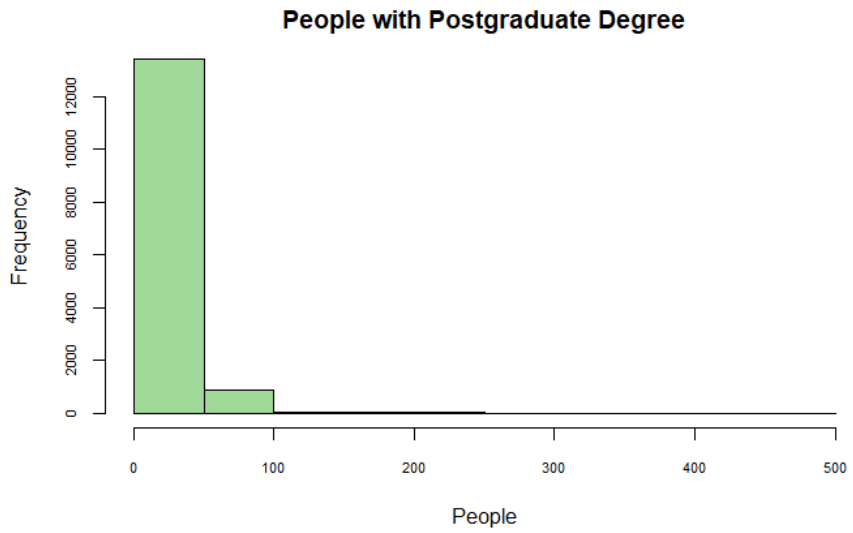


Transform

We further investigated the distributions of both the Post Graduate and Under Graduate education levels found within the tidied education dataframe. After filtering new dataframes and running histogram functions on the Post Graduate and Under Graduate education levels, we found that both distributions were heavily positively skewed (right skewed). In order to visualize a normal distribution for better understanding of the data spread, both Post and UnderGrad undertook a log transformation and reassigned to their own vectors. Running a histogram function over these new vectors visualises a clearer normal distribution.

Hide

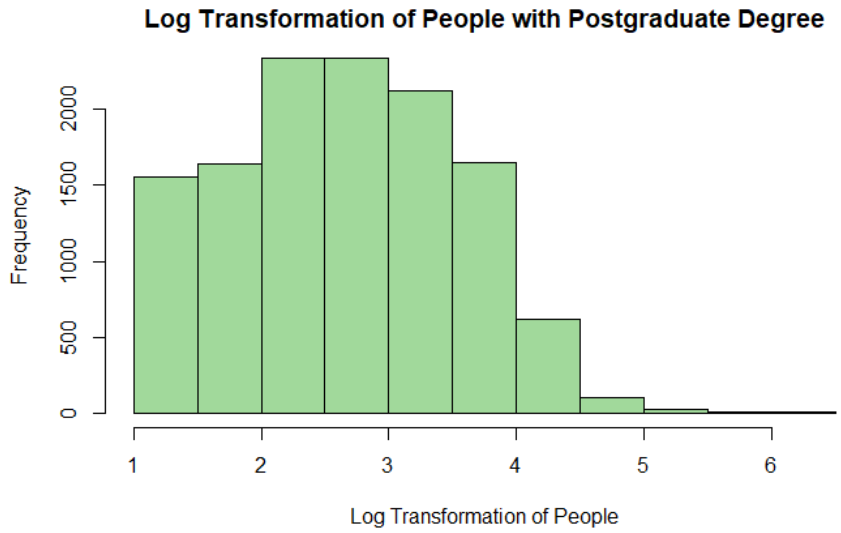
```
#Filter the data to only include counts of people who completed postgraduate degree level education.
edu_post <- edu_tidy_join %>%  filter(Ed_level == 'Postgraduate Degree Level')
#Plotting a frequency histogram
hist(edu_post$People, main = "People with Postgraduate Degree",
     xlab = "People", col = "#a1d99b", cex.axis = 0.7)
```



The above histogram shows that the distribution is heavily right skewed. Hence we will undertake transformation of the data to see if we can transform the data to become normally distributed.

Hide

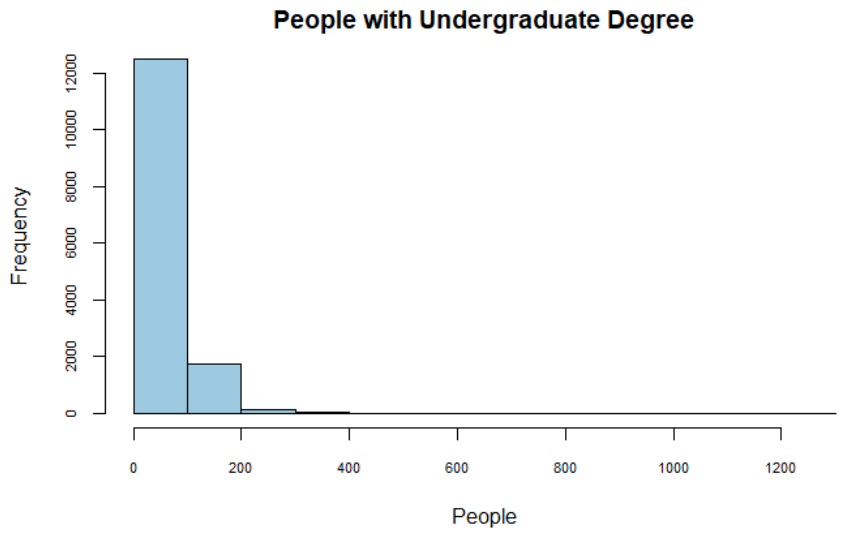
```
#Calculate the natural logarithm of the counts of people
edu_post_log <- log(edu_post$People)
hist(edu_post_log, main = "Log Transformation of People with Postgraduate Degree",
     xlab = "Log Transformation of People", col = "#a1d99b")
```



The log transformation results in a clear normal distribution

Hide

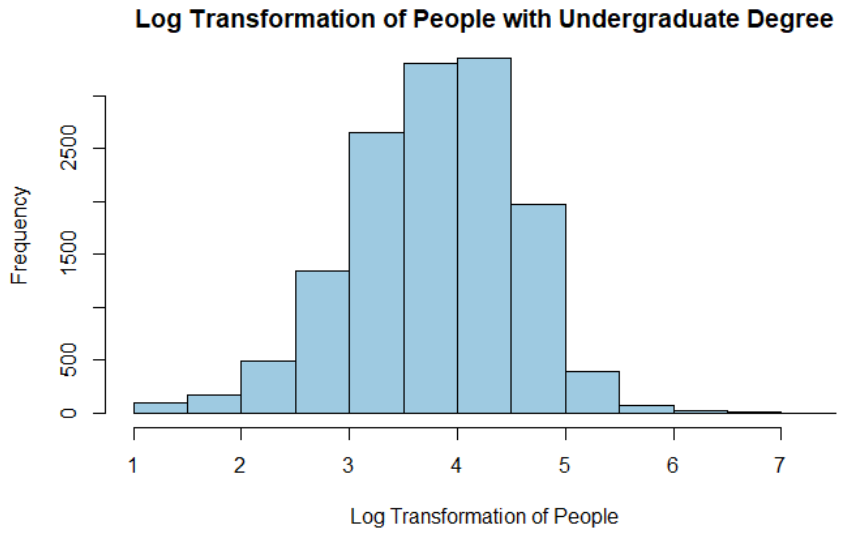
```
#Filter the data to only include counts of people who completed bachelor degree level education.
edu_under <- edu_tidy_join %>%  filter(Ed_level == 'Bachelor Degree Level')
#Plotting a frequency histogram
hist(edu_under$People, main = "People with Undergraduate Degree",
     xlab = "People", col = "#9ecae1", cex.axis = 0.7)
```



The above histogram shows that the distribution is heavily right skewed. Hence we will undertake transformation of the data to see if we can transform the data to become normally distributed.

Hide

```
#Applied the same treatment as above, taking the natural logarithm of the count
edu_under_log <- log(edu_under$People)
hist(edu_under_log, main = "Log Transformation of People with Undergraduate Degree",
     xlab = "Log Transformation of People", col = "#9ecae1")
```



Again, the data is now normally distributed as a result of the log transformation

Additional Step: Machine Learning

Here we wanted to investigate the linear relationship between undergraduate and post graduate education levels. In order to achieve a post graduate level of education, obviously an undergraduate level would have to be undertaken. The purpose of this analysis is to find if a consistent proportion of university attendees continue their studies to a post graduate level. For the machine learning aspect, we trained the model on the numeric data from the undergraduate variable to predict the postgraduate variable.

Hide

```
#Select data for machine learning
data <- edu_uni_tot2[2:3]
#Make task
task <- makeRegrTask(data = data, target = 'Post')
```

```
Provided data is not a pure data.frame but from class tbl_df, hence it will be converted.
```

Hide

```
#Make learner
learner <- makeLearner('regr.glm')
#Fit model
n <- nrow(data)
training.set <- sample(n, size = 2*n/3)
test.set <- setdiff(1:n, training.set)
model <- mlr::train(learner, task, subset = training.set)
#Predict
pred <- predict(model, task = task, subset = test.set)
#Evaluate
performance(pred, measures = list(mse, mae))
```

```
      mse       mae
97.067959  6.840149
```

Hide

```
x <- pred$data$truth
y <- pred$data$response
plot(x, y, xlab = 'Actual Value', ylab = 'Predicted Value', col = 'blue', main = 'Regression Machine Learning')
abline(1:500, 1:500, lwd = 2, col = 'red')
```