Appendix

Appendix 1 - Example of Australian Bureau of Statistics Census 2016 data extract R code

```
library(rsdmx)
library(tidyverse)

# getwd()
setwd(dirname(rstudioapi::getActiveDocumentContext()$path))

# Check to make sure the ABS folder is available
# and, if not, create it. Saving file to right
# location will fail without the required folder
if (!dir.exists("../Data Files/ABS")) {
    create.dir("../Data Files/ABS")
}

# Get the ABS Census 2016 Data on Dwelling Type
dwelling_data <- as.data.frame(readSDMX(providerId = "ABS",
    resource = "data", flowRef = "ABS_C16_T24_SA",
    key = "TOT.TOT+11+21+22+31+32+33+34+91+92+93+94+Z+NA.0+1+2+3+4+5+6+7+8+9.SA2",
    key.mode = "SDMX", start = 2016, end = 2016))
summary(dwelling_data)
head(dwelling_data)
str(dwelling_data)

# MISSING 9 SA2 Codes
dwelling_data %>% distinct(ASGS_2016)

# Distinct dimension values
dwelling_data %>% distinct(DWTD_2016)

## Retrieve Metadata to help with decoding values.
ds_url = "http://stat.data.abs.gov.au/restsdmx/sdmx.ashx/GetDataStructure/ABS_C16_T24_SA"
dataStructure <- readSDMX(ds_url)
codeList <- slot(dataStructure, "codelists")

# Dwelling Type
dwelling_type <- as.data.frame(codeList, codelistId = "CL_ABS_C16_T24_SA_STRD_2016")

# Get Required Data and put in meaningful
# descriptions
dwelling_data_final <- dwelling_data %>% inner_join(dwelling_type,
    by = c(STRD_2016 = "id")) %>% select(SA2_CODE = ASGS_2016,
    DWELLING_TYPE = label.en, obsValue)

# getwd()
write_csv(dwelling_data_final, "../Data Files/ABS/Dwelling_Type_SA2_2016.csv")
```

Appendix 2 - Example of NSW Government Air Quality data download using R

```
# getwd()
setwd(dirname(rstudioapi::getActiveDocumentContext()$path))

if (!dir.exists("../Data Files/NSWGovt/")) {
    dir.create("../Data Files/NSWGovt/")
}

## Download NSW Air Quality File if it doesn't
## already exist
if (!file.exists("../Data Files/NSWGovt/AirQuality_Data.xls")) {
    aq = "https://airquality.environment.nsw.gov.au/aquisnetnswphp/tmp/tmp_table_21553_1555911469.xls"
    download.file(aq, destfile = "../Data Files/NSWGovt/AirQuality_Data.xls",
        mode = "wb")
}

## Download NSW Air Quality Stations if it doesn't
## already exist
if (!file.exists("../Data Files/NSWGovt/AirQuality_Station_Data.xlsx")) {
    stations = paste0("https://datasets.seed.nsw.gov.au/dataset/",
        "ee5fd225-ab54-49c4-8c91-930219018cd0/resource/",
        "e09a1918-af2b-4375-ad04-00fabce72a10/download/",
        "air-quality-monitoring-sites-summary.xlsx")
    download.file(stations, destfile = "../Data Files/NSWGovt/AirQuality_Stations_Data.xlsx",
        mode = "wb")
}
```

Appendix 3 - Example of Australian Bureau of Statistics Socio-Economic Indexs for Areas data extract using R

```
library(rsdmx)

# getwd()
setwd(dirname(rstudioapi::getActiveDocumentContext()$path))

if (!dir.exists("../Data Files/ABS/")) {
    dir.create("../Data Files/ABS/")
}

data <- as.data.frame(readSDMX(providerId = "ABS",
    resource = "data", flowRef = "ABS_SEIFA2016_SA2",
    key.mode = "SDMX", start = 2016, end = 2016))

write.csv(data, "../Data Files/ABS/SEIFA_2016_Data.csv")
```

Appendix 4 - Example of Australian Bureau of Statistics Census 2016 data cleaning using R (Dwelling Type)

```
library(tidyverse)
library(data.table)

# getwd()
setwd(dirname(rstudioapi::getActiveDocumentContext()$path))

## Read the raw data csv
raw_data <- read.csv("../../Raw Data/Data Files/ABS/Dwelling_Type_SA2_2016.csv",
    quote = "\"")

head(raw_data)
str(raw_data)

# Clean the data - Band Dwelling Type and
# create percentages - Note Total is not
# always the sum of the breakdown
clean_data <- raw_data %>% filter(DWELLING_TYPE !=
    "Total") %>% mutate(DWELLING_BAND = case_when(DWELLING_TYPE ==
    "Separate house " ~ "DWELLING_HOUSE",
    DWELLING_TYPE %like% "Semi-detached, row or terrace house" ~
        "DWELLING_SEMI", DWELLING_TYPE %like%
        "Flat or apartment" ~ "DWELLING_FLAT",
    DWELLING_TYPE %like% "House or flat attached to a shop" ~
        "DWELLING_FLAT", TRUE ~ "DWELLING_OTHER")) %>%
    select(SA2_CODE, DWELLING_BAND, obsValue) %>%
    group_by(SA2_CODE, DWELLING_BAND) %>%
    summarise(Total_Value = sum(obsValue)) %>%
    spread(DWELLING_BAND, Total_Value) %>%
    mutate(PERC_DWELLING_HOUSE = DWELLING_HOUSE/(DWELLING_HOUSE +
        DWELLING_FLAT + DWELLING_SEMI + DWELLING_OTHER),
        PERC_DWELLING_FLAT = DWELLING_FLAT/(DWELLING_HOUSE +
            DWELLING_FLAT + DWELLING_SEMI +
            DWELLING_OTHER), PERC_DWELLING_SEMI = DWELLING_SEMI/(DWELLING_HOUSE +
            DWELLING_FLAT + DWELLING_SEMI +
            DWELLING_OTHER), PERC_DWELLING_OTHER = DWELLING_OTHER/(DWELLING_HOUSE +
            DWELLING_FLAT + DWELLING_SEMI +
            DWELLING_OTHER))

# Write cleaned data set to csv getwd()
write_csv(clean_data, "../Data Files/ABS/Dwelling_Type_SA2.csv")
```

Appendix 5 - Example of Australian Bureau of Statistics Census 2016 data cleaning using R (Demographics)

```
library(tidyverse)
library(janitor)

setwd(dirname(rstudioapi::getActiveDocumentContext()$path))

erp <- read_csv("../../Raw Data/Data Files/ABS/ERP/ABS_ERP_ASGS2016_25042019132433480.csv")

names(erp)
erp <- erp %>% select(-c("MEASURE", "Measure",
    "SEX_ABS", "AGE", "FREQUENCY", "Frequency",
    "TIME", "Flag Codes", "Flags", "REGIONTYPE",
    "Geography Level"))

erp <- erp %>% rename(sa2_code = ASGS_2016) %>%
    clean_names()

erp_by_sex <- erp %>% group_by(sa2_code,
    sex) %>% summarise(total_value = sum(value)) %>%
    spread(sex, total_value) %>% clean_names()

erp_by_age <- erp %>% group_by(sa2_code,
    age) %>% summarise(total_value = sum(value)) %>%
    spread(age, total_value) %>% clean_names()

erp <- erp_by_sex %>% left_join(erp_by_age)

write_csv(erp, "../Data Files/ABS/ERP_SA2_2016.csv")
```

Appendix 6 - Example of finding percent of SA2 area covered by parkland using R

```
library(tidyverse)

# getwd()
setwd(dirname(rstudioapi::getActiveDocumentContext()$path))

# Create destination folder if it doesn't
# already exist
if (!dir.exists("../Data Files/ABS/")) {
    dir.create("../Data Files/ABS/")
}

# Read the raw data csv
mesh_blocks <- read.csv("../../Raw Data/Data Files/ABS/Mesh_Blocks/MB_2016_NSW.csv")

str(mesh_blocks)
mesh_blocks %>% distinct(MB_CATEGORY_NAME_2016)

# Find % of space allocated to Parkland
# for each mesh block
open_space <- mesh_blocks %>% filter(STATE_NAME_2016 ==
    "New South Wales") %>% select(MB_CODE_2016,
    MB_CATEGORY_NAME_2016, SA2_CODE = SA2_MAINCODE_2016,
    AREA_SQKM = AREA_ALBERS_SQKM) %>% group_by(SA2_CODE,
    MB_CATEGORY_NAME_2016) %>% summarise(SUM_AREA_SQKM = sum(AREA_SQKM)) %>%
    spread(MB_CATEGORY_NAME_2016, SUM_AREA_SQKM,
        fill = 0) %>% mutate(PERC_OPEN_SPACE = Parkland/(Commercial +
    Education + `Hospital/Medical` + Industrial +
    MIGRATORY + NOUSUALRESIDENCE + OFFSHORE +
    Other + Parkland + `Primary Production` +
    Residential + SHIPPING + Transport +
    Water))

# Write data to csv
write_csv(open_space, "../Data Files/ABS/Open_Space_SA2.csv")
```

Appendix 7 - Example of Australian Bureau of Statistics Socio-Economic Indexs for Areas data cleaning using R

```
library(tidyverse)

# Set directory to my the location where
# this file is
setwd(dirname(rstudioapi::getActiveDocumentContext()$path))
# getwd()

seifa_data <- read_csv("../../Raw Data/Data Files/ABS/SEIFA_2016_Data.csv")

# Review data - AsGS_2016 field is an INT
head(seifa_data)

# Summarise the data - obsValue has 48
# NA's but non of these have a
# SEIFA_MEASURE == SCORE
summary(seifa_data)
seifa_nas <- seifa_data %>% filter(is.na(obsValue) ==
    TRUE)

# Create a clean data set for use - only
# want high level scores and remove
# records with an obsValue of NA
clean_seifa_data <- seifa_data %>% filter(SEIFA_MEASURE ==
    "SCORE") %>% select(SA2_CODE = ASGS_2016,
    SEIFAINDEXTYPE, obsValue) %>% spread(SEIFAINDEXTYPE,
    obsValue) %>% select(SA2_CODE, SEIFA_Edu_Occ_Index = IEO,
    SEIFA_Economic_Res_Index = IER, SEIFA_Rel_SocioEco_Adv_Disadv_Index = IRSAD,
    SEIFA_Rel_SocioEco_Disadv_Index = IRSD)

# Write Clean Data to disk
write_csv(clean_seifa_data, "../Data Files/ABS/SEIFA_2016_Data.csv")

# Check for duplicates - nope none, only
# 1 record per SA2
clean_seifa_data %>% group_by(SA2_CODE) %>%
    mutate(total = n()) %>% filter(total >
    1)
```

Appendix 8 - Example of Exploratory Data Analysis of Australian Bureau of Statistics Census 2016 data

```
library(tidyverse)
library(Hmisc)
library(corrplot)

# getwd()
setwd(dirname(rstudioapi::getActiveDocumentContext()$path))

# read cleaned data set
dwelling_type <- read_csv("../Data Files/ABS/Dwelling_Type_SA2.csv")
mesh_blocks <- read_csv("../../Raw Data/Data Files/ABS/Mesh_Blocks/MB_2016_NSW.csv")
str(mesh_blocks)

# Get mesh block data at SA2 level
sa2_data <- mesh_blocks %>% distinct(SA2_MAINCODE_2016,
    SA2_NAME_2016, STATE_CODE_2016, STATE_NAME_2016)

# No duplicate SA2 Codes
sa2_data %>% group_by(SA2_MAINCODE_2016) %>% summarise(cnt = n()) %>%
    filter(cnt > 1)

# Some SA2's don't have any dwellings - positive
# skew
dwelling_type %>% mutate(TOTAL = DWELLING_HOUSE + DWELLING_FLAT +
    DWELLING_SEMI + DWELLING_OTHER) %>% ggplot() +
    geom_histogram(aes(x = TOTAL), bins = 50)

# 62 SA2's have no dwellings - 7 in NSW, a military
# base, centennial park, a NP, a cemetry, and
# Industrial area, Banksmeadow is whaves and
# industry
dwelling_type %>% mutate(TOTAL = DWELLING_HOUSE + DWELLING_FLAT +
    DWELLING_SEMI + DWELLING_OTHER) %>% filter(TOTAL ==
    0) %>% left_join(sa2_data, by = c(SA2_CODE = "SA2_MAINCODE_2016")) %>%
    select(SA2_CODE, TOTAL, SA2_NAME_2016, STATE_NAME_2016) %>%
    filter(between(SA2_CODE, 1e+08, 2e+08))

# There are a couple of areas with high numbers of
# dwellings - Waterloo/Beaconsfield in NSW is high
# density
dwelling_type %>% mutate(TOTAL = DWELLING_HOUSE + DWELLING_FLAT +
    DWELLING_SEMI + DWELLING_OTHER) %>% filter(TOTAL >
    15000) %>% left_join(sa2_data, by = c(SA2_CODE = "SA2_MAINCODE_2016")) %>%
    select(SA2_CODE, TOTAL, SA2_NAME_2016, STATE_NAME_2016) %>%
    filter(between(SA2_CODE, 1e+08, 2e+08))

# Remove SA2's with no dwellings and only show NSW
# SA2's
dwelling_type_filtered <- dwelling_type %>% mutate(TOTAL = DWELLING_HOUSE +
    DWELLING_FLAT + DWELLING_SEMI + DWELLING_OTHER) %>%
    filter(TOTAL != 0) %>% inner_join(sa2_data, by = c(SA2_CODE = "SA2_MAINCODE_2016"))
```

```r
## DWELLING HOUSE Some areas in NSW have no houses -
## data may be slightly skewed
dwelling_type_filtered %>% ggplot() + geom_histogram(aes(x = DWELLING_HOUSE),
    bins = 50)

# Standardise data and confirm data has a long tail
house_std <- scale(dwelling_type_filtered$DWELLING_HOUSE)
qqnorm(house_std)
abline(a = 0, b = 1, col = "grey")

## DWELLING FLAT There are 21 no flat SA2's in NSW -
## industrial areas, offshore shipping, Rural areas
dwelling_type_filtered %>% filter(DWELLING_FLAT ==
    0) %>% select(SA2_CODE, SA2_NAME_2016, PERC_DWELLING_HOUSE,
    PERC_DWELLING_FLAT, PERC_DWELLING_SEMI, PERC_DWELLING_OTHER)

# Some areas in NSW have no houses - data skewed
dwelling_type_filtered %>% ggplot() + geom_histogram(aes(x = DWELLING_FLAT),
    bins = 100)

# Standardise data and confirm data is not normally
# distributed
flat_std <- scale(dwelling_type_filtered$DWELLING_FLAT)
qqnorm(flat_std)
abline(a = 0, b = 1, col = "grey")

## DWELLING SEMI There are 19 no semi SA2's in NSW -
## industrial areas, airport, offshore shipping,
## rural areas
dwelling_type_filtered %>% filter(DWELLING_SEMI ==
    0) %>% select(SA2_CODE, SA2_NAME_2016, PERC_DWELLING_HOUSE,
    PERC_DWELLING_FLAT, PERC_DWELLING_SEMI, PERC_DWELLING_OTHER)

# Some areas in NSW have no houses - data skewed
dwelling_type_filtered %>% ggplot() + geom_histogram(aes(x = DWELLING_SEMI),
    bins = 100)

# Standardise data and confirm data is not normally
# distributed
semi_std <- scale(dwelling_type_filtered$DWELLING_SEMI)
qqnorm(semi_std)
abline(a = 0, b = 1, col = "grey")

# Check correlation between variables
dwelling_matrix <- dwelling_type_filtered %>% select(DWELLING_FLAT,
    DWELLING_HOUSE, DWELLING_OTHER, DWELLING_SEMI) %>%
    as.matrix()

# Show values - nothing really high
rcorr(dwelling_matrix, type = "pearson")

# And a plot for good measure
corrplot(cor(dwelling_matrix), method = "ellipse")
```

Appendix 9 - Example of merging cleaned datasets for modelling using R

```
library(tidyverse)

# getwd()
setwd(dirname(rstudioapi::getActiveDocumentContext()$path))

# Read csv's
dwelling_type <- read_csv("../../Clean Data/Data Files/ABS/Dwelling_Type_SA2.csv")
hh_composition <- read_csv("../../Clean Data/Data Files/ABS/HouseHold_Composition_SA2.csv")
place_of_birth <- read_csv("../../Clean Data/Data Files/ABS/Place_Of_Birth_SA2.csv")
seifa <- read_csv("../../Clean Data/Data Files/ABS/SEIFA_2016_Data.csv")
mesh_blocks <- read_csv("../../Raw Data/Data Files/ABS/Mesh_Blocks/MB_2016_NSW.csv")

# Get mesh block data at SA2 level
sa2_data <- mesh_blocks %>% distinct(SA2_MAINCODE_2016,
    SA2_NAME_2016, STATE_CODE_2016, STATE_NAME_2016)

# Join Datasets together
model_data <- dwelling_type %>% inner_join(hh_composition,
    by = c("SA2_CODE")) %>% inner_join(place_of_birth,
    by = c("SA2_CODE")) %>% inner_join(seifa, by = c("SA2_CODE")) %>%
    semi_join(sa2_data, by = c(SA2_CODE = "SA2_MAINCODE_2016")) %>%
    select(-starts_with("PERC_"))
```