ETC5512: Wild Caught Data

Australian election data

Lecturer: *Emi Tanaka*

Department of Econometrics and Business Statistics

✉ ETC5512.Clayton-x@monash.edu

📅 Week 5

Australian election data

- Much like the census, **election** attempts to collect the data from the population.

- In Australia, it is compulsory by law to vote in elections if you are an Australian citizen (or eligible British subject) aged 18 years old or over and have lived in your address for at least one month.

- The Australian Electoral Commission (AEC) is an independent federal agency in charge of federal Australian elections and provides the geographical boundaries of the electoral divisions.

Today you will:

- look at the 2019 election results

- visualise the 2019 election results spatially in a few ways

- look at reprojecting geographic data into different coordinate reference systems

2019 Australian Federal Election

- Parliament of Australia comprises two houses:
  - **Senate** (upper house) comprising 76 senators
  - **House of Representatives** (lower house) comprising 151 members
- Government is formed by the party or coalition with majority of the seats in the lower house
- The 2019 Australian Federal Election was held on Sat 18th May 2019
- The next federal election will be likely be held in 2022

- Major parties in Australia:
  - Coalition:
    - Liberal
    - National
  - Labor
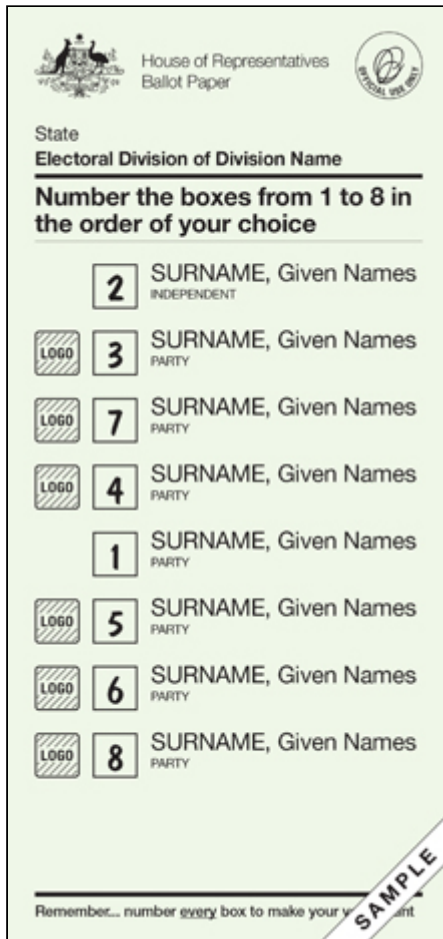- Some minor parties in Australia:
  - The Greens
  - One Nation

Ballots

- **House of Representatives** uses the instant-runoff voting system

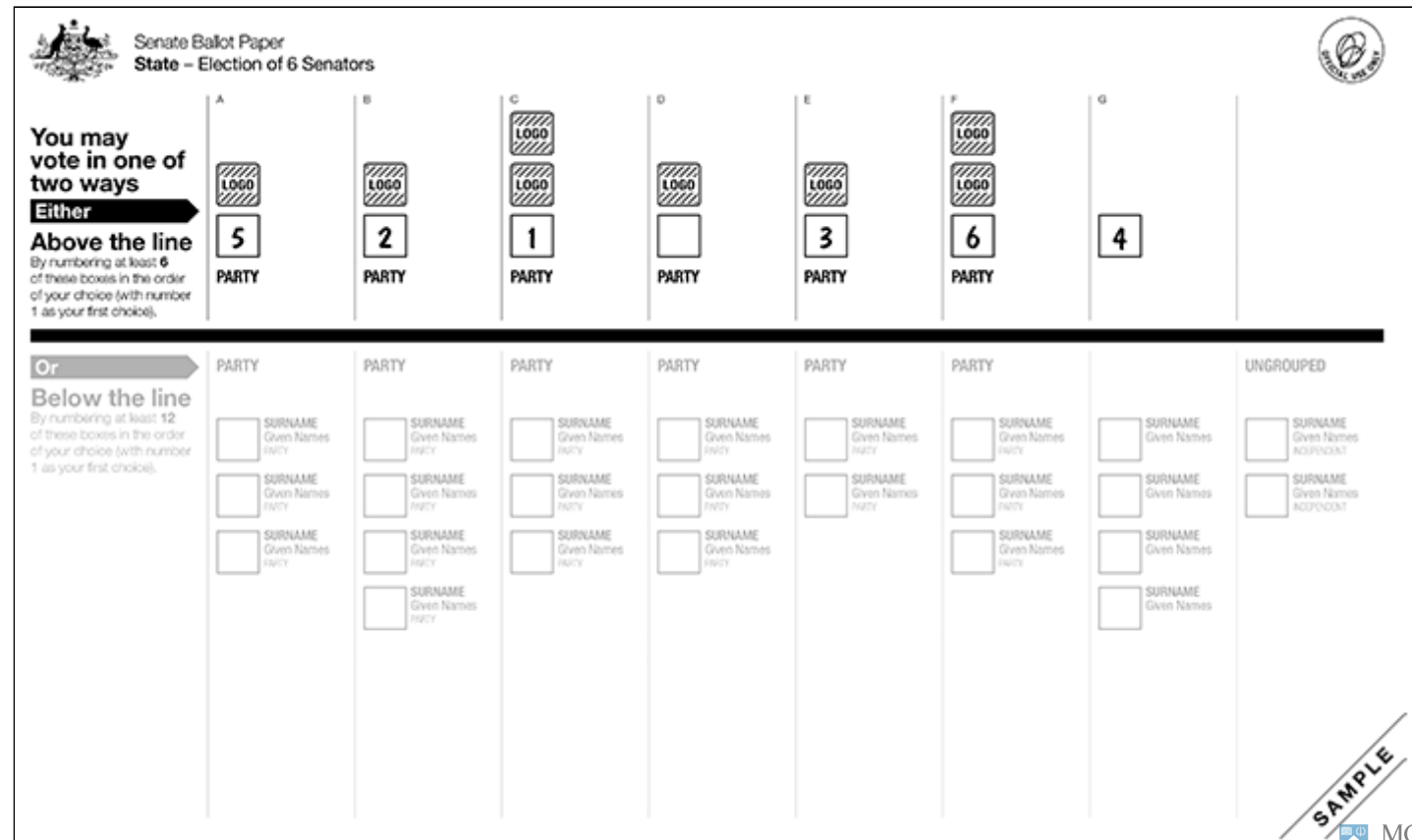- Senate uses the single transferable voting system

2019 Australian Federal Election Data

- Get the distribution of preferences by candidate by division for the 2019 Australian Federal Election

⬇ https://results.aec.gov.au

> ⓘ 1. 2019 federal election
>
>   2. Downloads
>
>   3. Distribution of preferences by candidate by division

- Or refer directly to the link:

  https://results.aec.gov.au/24310/Website/Downloads/HouseDopByDivisionDownload-24310.csv

House of Representative Voting Data

```
library(tidyverse)
votes <- read_csv("https://results.aec.gov.au/24310/Website/Downloads/HouseDopByDivisionDownload
glimpse(votes)
```

```
## Rows: 26,632
## Columns: 14
## $ StateAb          <chr> "ACT", "ACT", "ACT", "ACT", "ACT", "ACT", "ACT", "ACT…
## $ DivisionID       <dbl> 318, 318, 318, 318, 318, 318, 318, 318, 318, 318, 318…
## $ DivisionNm       <chr> "Bean", "Bean", "Bean", "Bean", "Bean", "Bean", "Bean…
## $ CountNumber      <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,…
## $ BallotPosition   <dbl> 1, 1, 1, 1, 2, 2, 2, 2, 3, 3, 3, 3, 4, 4, 4, 4, 5, 5,…
## $ CandidateID      <dbl> 33426, 33426, 33426, 33426, 32130, 32130, 32130, 3213…
## $ Surname          <chr> "FAULKNER", "FAULKNER", "FAULKNER", "FAULKNER", "CHRI…
## $ GivenNm          <chr> "Therese", "Therese", "Therese", "Therese", "Jamie", …
## $ PartyAb          <chr> "AUP", "AUP", "AUP", "AUP", "IND", "IND", "IND", "IND…
## $ PartyNm          <chr> "Australian Progressives", "Australian Progressives",…
## $ Elected          <chr> "N", "N", "N", "N", "N", "N", "N", "N", "N", "N", "N"…
## $ HistoricElected  <chr> "N", "N", "N", "N", "N", "N", "N", "N", "N", "N", "N"…
## $ CalculationType  <chr> "Preference Count", "Preference Percent", "Transfer C…
```

Electoral district of Monash

- Let's have a look at the electoral district named "Monash"

```
votes %>%
  filter(DivisionNm=="Monash") %>%
  DT::datatable(width = 1160, height = 400, options = list(pageLength = 400))
```

Show [400 ▼] entries   Search: [   ]

	StateAb	DivisionID	DivisionNm	CountNumber	BallotPosition	CandidateID	Surname	GivenNm
1	VIC	323	Monash	0	1	32690	VERHOEVEN	John
2	VIC	323	Monash	0	1	32690	VERHOEVEN	John
3	VIC	323	Monash	0	1	32690	VERHOEVEN	John
4	VIC	323	Monash	0	1	32690	VERHOEVEN	John
5	VIC	323	Monash	0	2	32137	FOZARD	Michael Stewart

District: Monash

```
votes_monash <- votes %>%
  # get the preference count only
  filter(CalculationType == "Preference Count") %>%
  # get the Monash division
  filter(DivisionNm == "Monash")
```


	StateAb	DivisionID	DivisionNm	CountNumber	BallotPosition	CandidateID
1	VIC	323	Monash	0	1	32
2	VIC	323	Monash	0	2	32
3	VIC	323	Monash	0	3	32
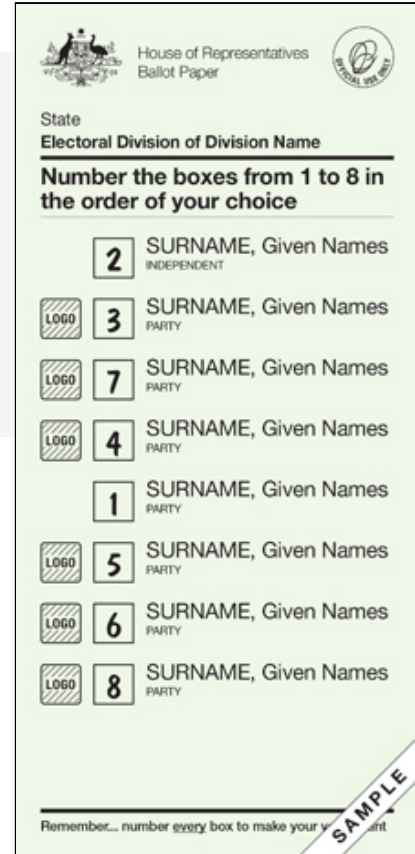
Visualising the counts

```r
ggplot(votes_monash) +
  geom_col(aes(x = CalculationValue, y = Surname)) +
  geom_text(aes(label = paste("Count", CountNumber)),
            x = 10000, y = 3, size = 16, color = "#ee64a4",
            alpha = 0.4, hjust = "left") +
  facet_wrap(~CountNumber)
```
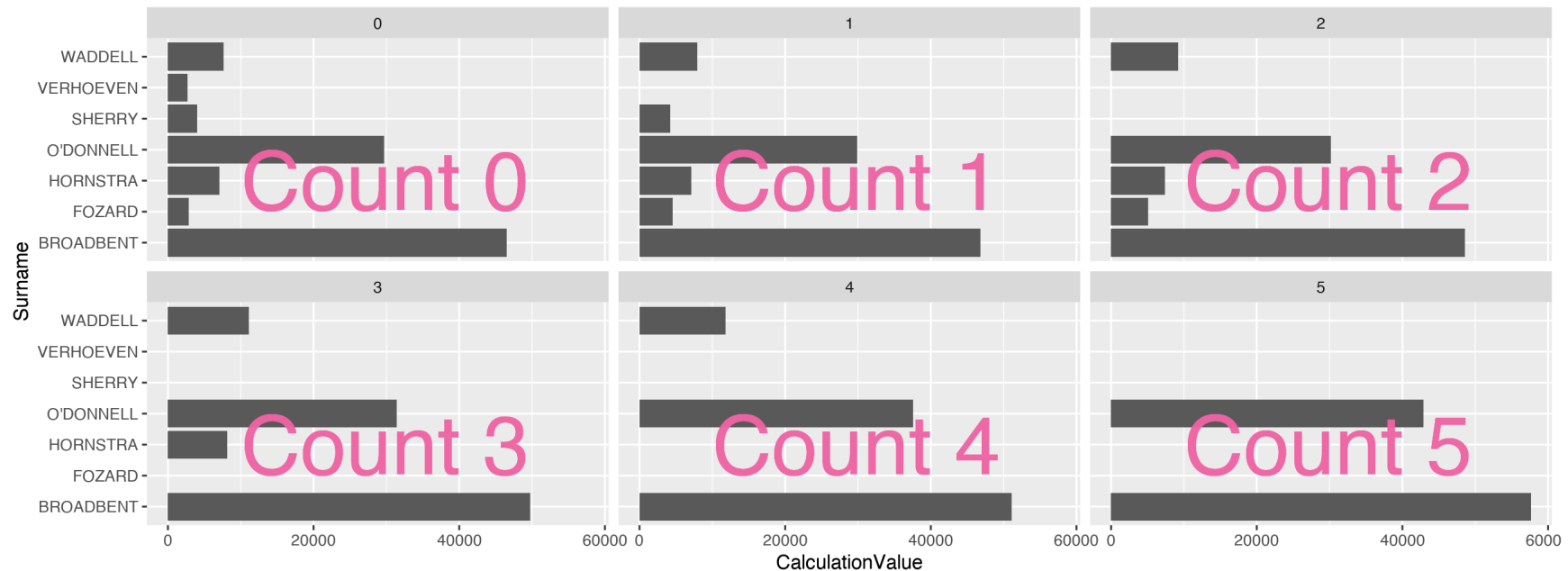
… but better to order candidates by counts

```
mutate(votes_monash, Surname = fct_reorder(Surname, CalculationValue, sum)) %>%
  ggplot() +
  geom_col(aes(x = CalculationValue, y = Surname)) +
  geom_text(aes(label = paste("Count", CountNumber + 1)),
            x = 10000, y = 3, size = 16, color = "#ee64a4", alpha = 0.4, hjust =
  facet_wrap(~CountNumber)
```

Winner:

Russel
Broadbent

Where is the electoral district of Monash?

- *...doesn't* include Monash Clayton campus
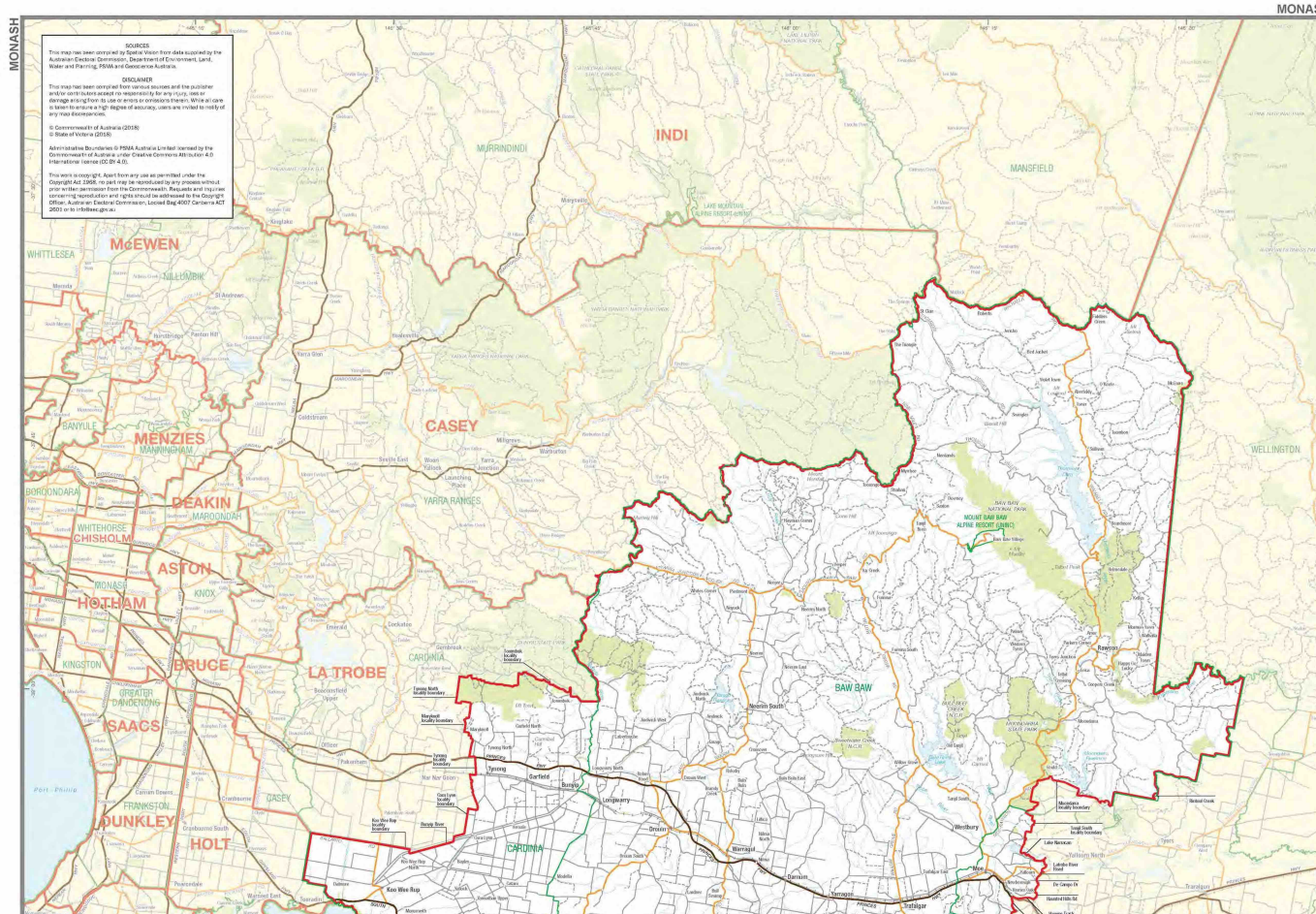
Electoral district of **Hotham**

- *Does* include Monash Clayton campus
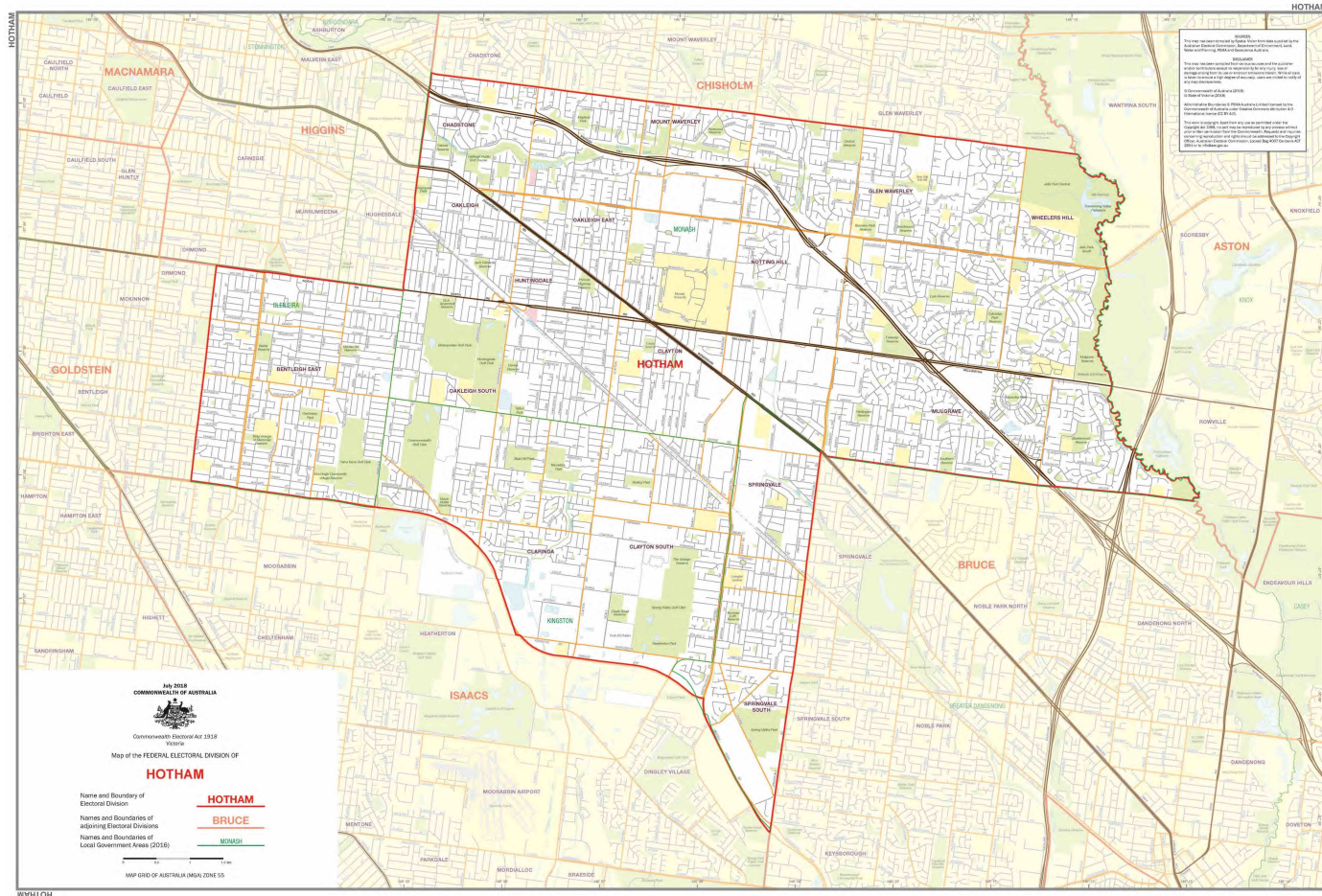
Australian Electorates Divsions

There are 151 electorates in 2019.
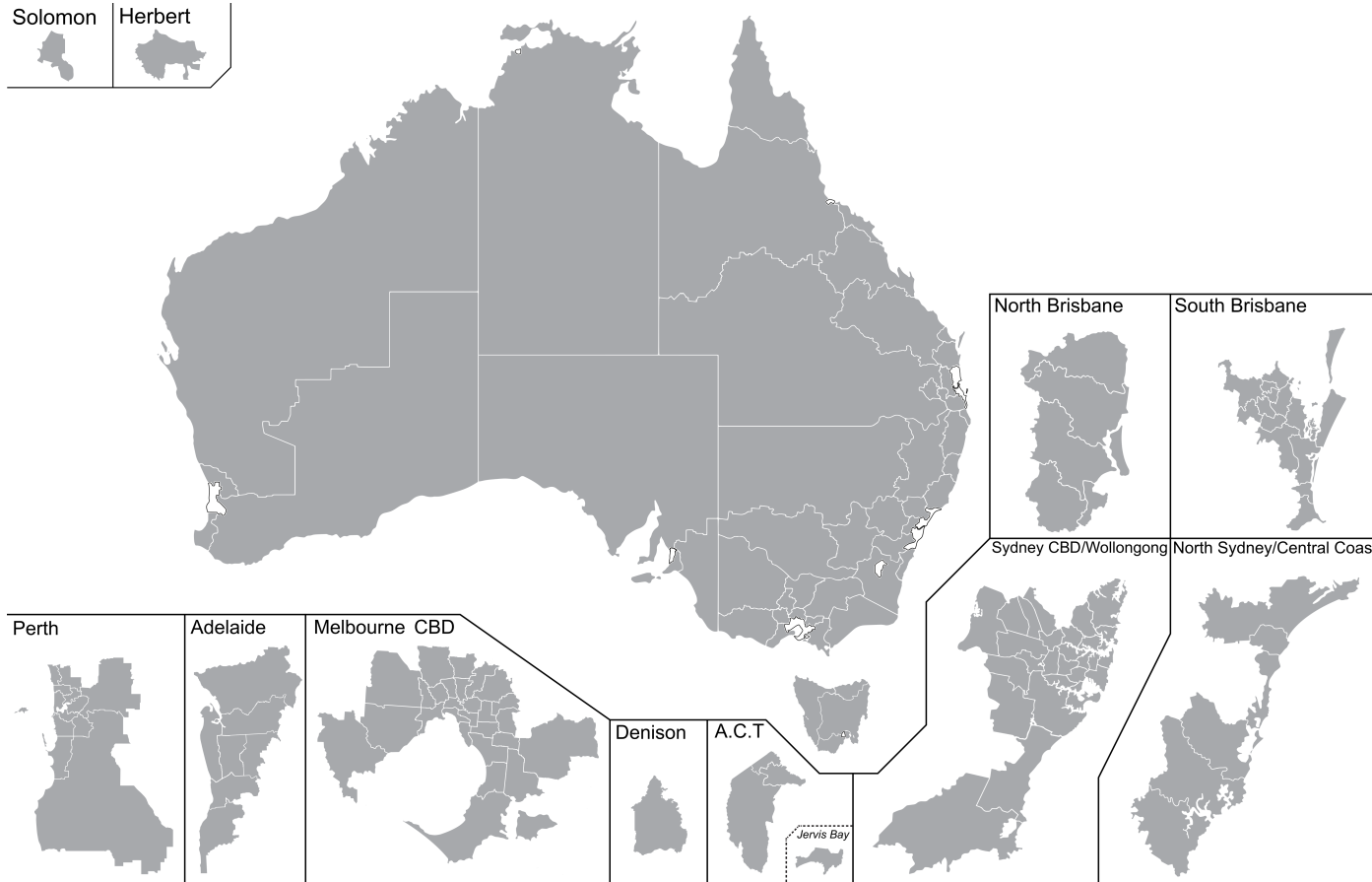
| | The **geographical boundaries of the electoral divisions** are determined by the Redistribution Committee and are redrawn every so often to ensure similar number of electors in each electoral division for a given state or territory. |

| | This means that the geographical boundaries could be different across years. |

Federal electoral boundary GIS data

- GIS (Geographic Information System) is a framework that capture and inspect geographical data.

- This data is found at

  ⬇ https://www.aec.gov.au/electorates/gis/licence.htm

- Agree to the license to get to the download page

> " *The Licensee must make End-users aware the data was sourced from the Australian Electoral Commission and is used under licence.*

Note: the federal electoral boundary is provided by Australian Electoral Commission

© Commonwealth of Australia (Australian Electoral Commission) 2021

- We download the ESRI zip file for Victoria.

- To work with spatial data, we use the `sf` R-package.

Working with spatial data

```r
library(sf)
aec_map <- read_sf(here::here("data/vic-july-2018-esri/E_AUGFN3_region.shp"))
aec_map

## Simple feature collection with 38 features and 9 fields
## Geometry type: MULTIPOLYGON
## Dimension:     XYZ
## Bounding box:  xmin: 140.9617 ymin: -39.15919 xmax: 149.9767 ymax: -33.9804
## z_range:       zmin: 0 zmax: 0
## Geodetic CRS:  GDA94
## # A tibble: 38 × 10
##    E_div_numb Elect_div Numccds Actual Projected Total_Popu Australian Area
##         <dbl> <chr>       <dbl>  <dbl>     <dbl>      <dbl>      <dbl>
## 1           1 Aston         377 109405    111014          0          0
```

Geometry object and visualisation in as `ggplot`

```
aec_map$geometry[[1]]

## MULTIPOLYGON Z (((145.3476 -37.85941 0, 145.3468 -37.8595 0, 145.3458 -37.859
```

```
ggplot(aec_map) +
    # or geom_sf(aes(geometry = geometry))
    geom_sf()
```
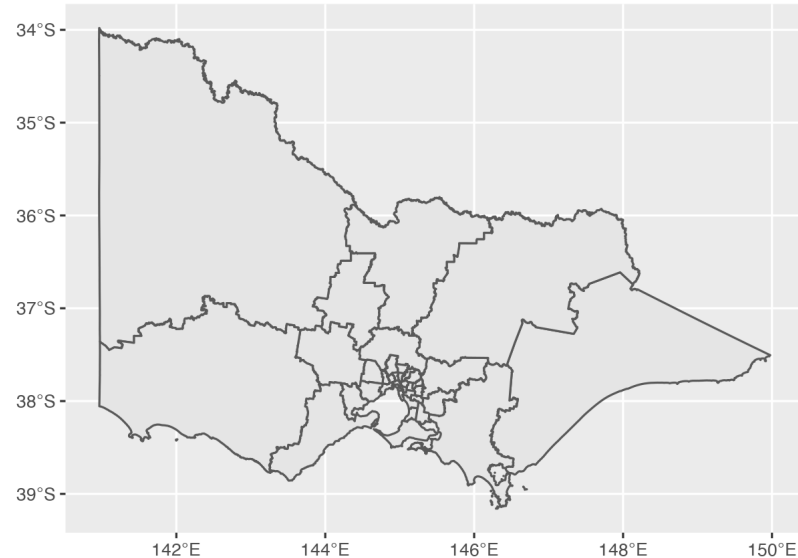
Integrating data of election winners

```
winners <- votes %>%
  # get the winner
  filter(Elected=="Y" & CountNumber==0 & CalculationType=="Preference Count") %>%
  # join the data
  right_join(aec_map, by = c("DivisionNm" = "Elect_div")) %>%
  select(DivisionNm, PartyAb, PartyNm, geometry)

ggplot(winners) + geom_sf(aes(fill = PartyAb, geometry = geometry))
```
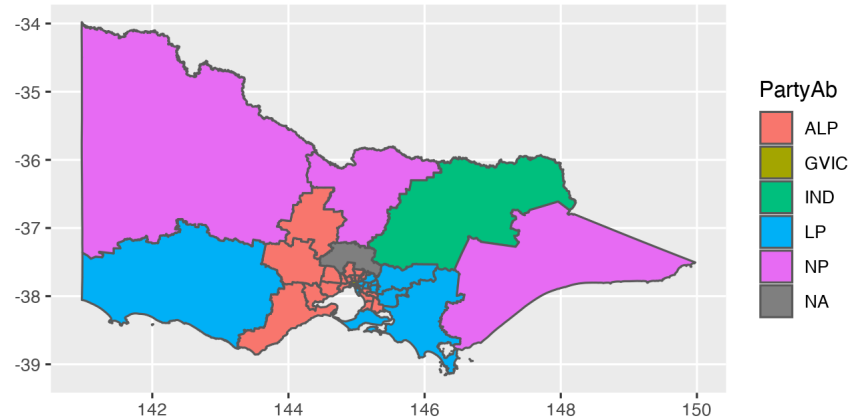

Is there something wrong here?

Investigating missing observation

```
winners %>%
  filter(is.na(PartyAb))

## # A tibble: 1 × 4
##   DivisionNm PartyAb PartyNm              geome
##   <chr>      <chr>   <chr>           <MULTIPOLYGON [
## 1 Mcewen     <NA>    <NA>    Z (((145.3664 -37.54513 0, 145.3663 -37.5459 0,
```

```
votes %>%
  # approximate string matching (or fuzzy matching)
  filter(agrepl("Mcewen", DivisionNm))

## # A tibble: 224 × 14
##   StateAb DivisionID DivisionNm CountNumber BallotPosi
##   <chr>        <dbl> <chr>            <dbl>
## 1 VIC            226 McEwen               0
## 2 VIC            226 McEwen               0                    1    32
```

So what went wrong here?

Victoria map of election winners

```r
winners_fix <- votes %>%
  mutate(DivisionNm = ifelse(DivisionNm=="McEwen", "Mcewen", DivisionNm)) %>%
  # get the winner
  filter(Elected=="Y" & CountNumber==0 & CalculationType=="Preference Count") %>
  # join the data
  right_join(aec_map, by = c("DivisionNm" = "Elect_div")) %>%
  select(DivisionNm, PartyAb, PartyNm, geometry)

ggplot(winners_fix) + geom_sf(aes(fill = PartyAb, geometry = geometry))
```

Maps visualisation

National map of election winners

```r
ausmap <- read_sf(here::here("data/national-esri-fe2019/COM_ELB_region.shp"))

all_winners <- votes %>%
  mutate(DivisionNm = case_when(DivisionNm=="McEwen" ~ "Mcewen",
                                DivisionNm=="McPherson" ~ "Mcpherson",
                                DivisionNm=="Eden-Monaro " ~ "Eden-monaro",
                                DivisionNm=="McMahon" ~ "Mcmahon",
                                DivisionNm=="O'Connor" ~ "O'connor",
                                TRUE ~ DivisionNm)) %>%
  # another way to select the winner
  filter(Elected=="Y") %>%
  group_by(DivisionID) %>%
  slice(1) %>%
  ungroup() %>%
  # then join the map data
  right_join(ausmap, by = c("DivisionNm" = "Elect_div"))
```
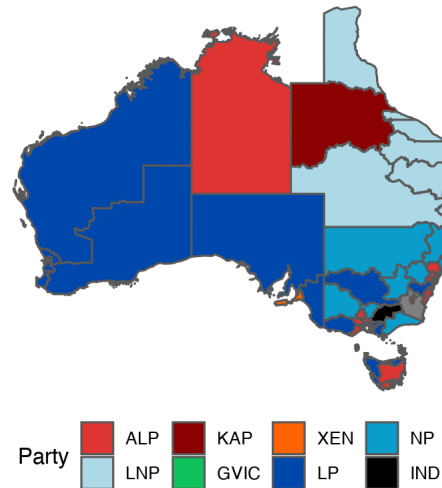
Using colors wisely

```r
auscolours <- c("ALP" = "#DE3533", "LNP" = "#ADD8E6", "KAP" = "#8B0000",
                "GVIC" = "#10C25B", "XEN" = "#ff6300", "LP" = "#0047AB",
                "NP" = "#0a9cca", "IND" = "#000000")

ggplot(all_winners) +
  geom_sf(aes(fill = PartyAb, geometry = geometry)) +
  scale_fill_manual(name = "Party", values = auscolours) +
  theme_void() +
  theme(legend.position="bottom")
```

Choropleth Map

Which party won from looking at this map and by how much?

Liberal/National Coalition: 77
Labor: 68
Greens: 1
Katter's Australian: 1
Centre Alliance: 1
Independents: 3

Mapping the centroids

```r
all_winners_centroid <- all_winners %>%
  # some issues with 122 and 137
  slice(-122, -137) %>%
  mutate(centroid = st_centroid(geometry))

ggplot(all_winners_centroid) +
  geom_sf(aes(geometry = geometry)) +
  geom_sf(aes(geometry = centroid, color = PartyAb)) +
  theme_void() +
  theme(legend.position="bottom")
```

Non-Contiguous, Dorling Cartogram

Party
ALP IND LP XEN
GVIC KAP NP
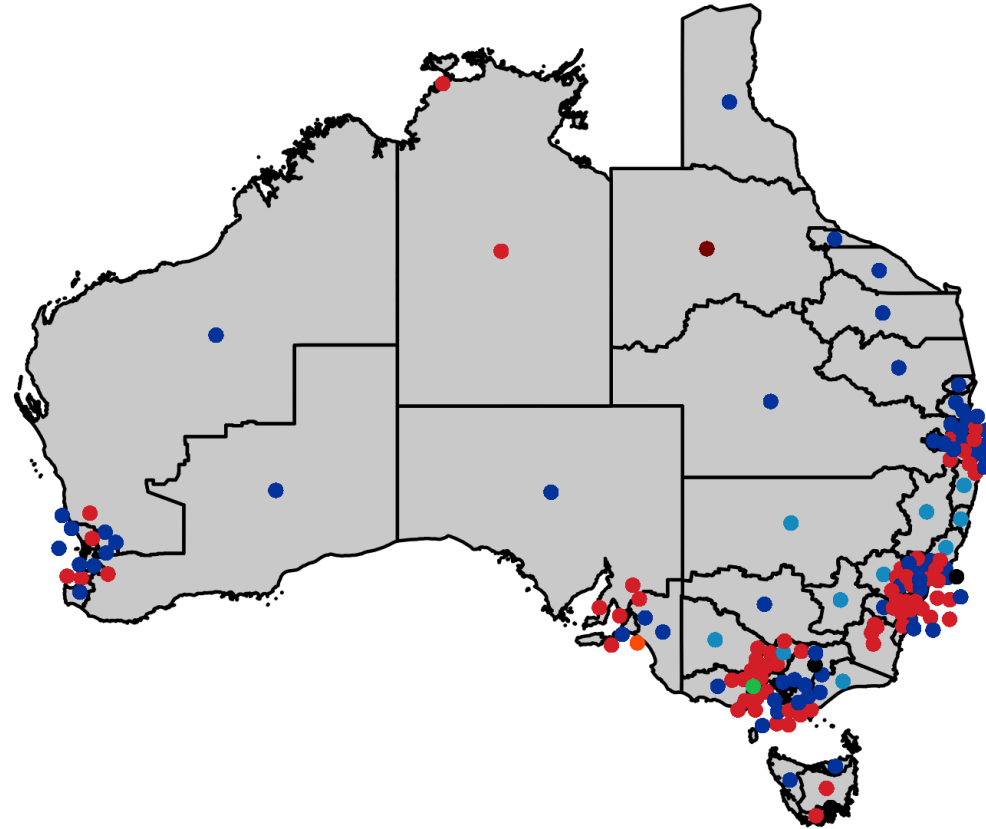
Improving the national map of election winners

```r
library(sugarbag)
centroids <- all_winners_centroid %>%
  select(DivisionNm, longitude = x, latitude = y)
grid <- create_grid(centroids = centroids, hex_size = 0.9, buffer_dist = 5)
hex_allocated <- allocate(centroids = centroids,
                          sf_id = "DivisionNm",
                          hex_grid = grid,
                          hex_size = 0.9, # same size used in create_grid
                          hex_filter = 10,
                          focal_points = capital_cities,
                          width = 30, verbose = TRUE)


hex_map <- hex_allocated %>%
  fortify_hexagon(hex_size = 0.9, sf_id = "DivisionNm") %>%
  left_join(all_winners_centroid, by = "DivisionNm")

ggplot(hex_map) +
  geom_sf(data = all_winners, aes(geometry = geometry)) +
  geom_polygon(aes(longitude, latitude, fill = PartyAb, group = DivisionNm), color = "black")
```

Tessellated Hexagon Map

Party

ALP	IND	LP	XEN
GVIC	KAP	NP	

MONASH University
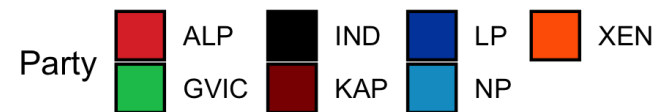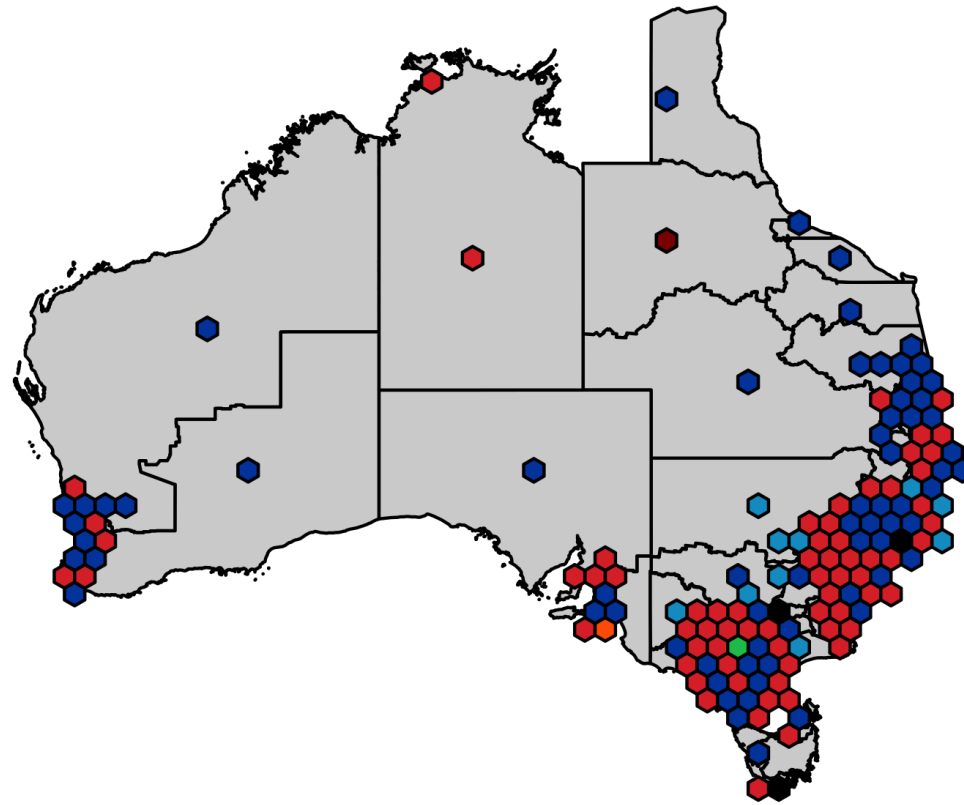
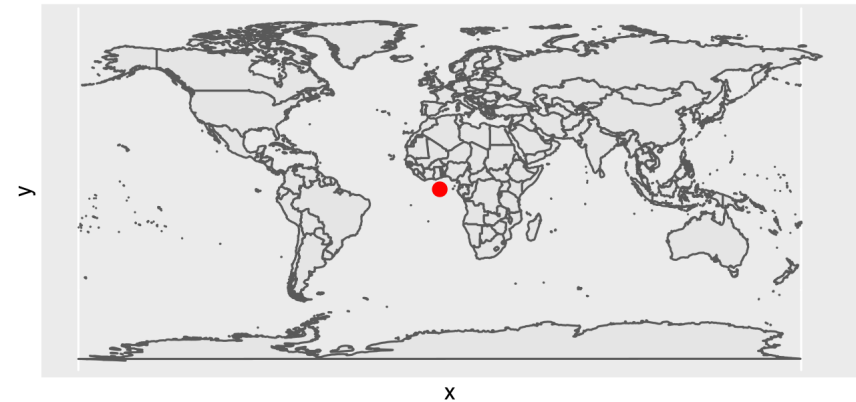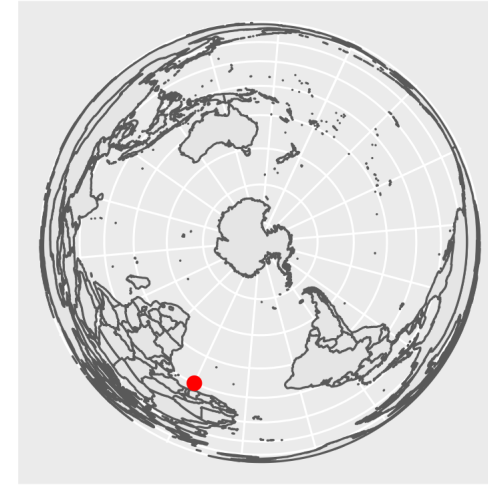# Coordinate reference system (CRS)

Geographic coordinate reference systems

- **Geographic CRSs** identify a location on the Earth's surface by *longitude* and *latitude*.

- **Longitude** is the East-West direction in angular distance from the Prime Meridian plane.

- **Latitude** is the angular distance North or South of the equatorial plane.
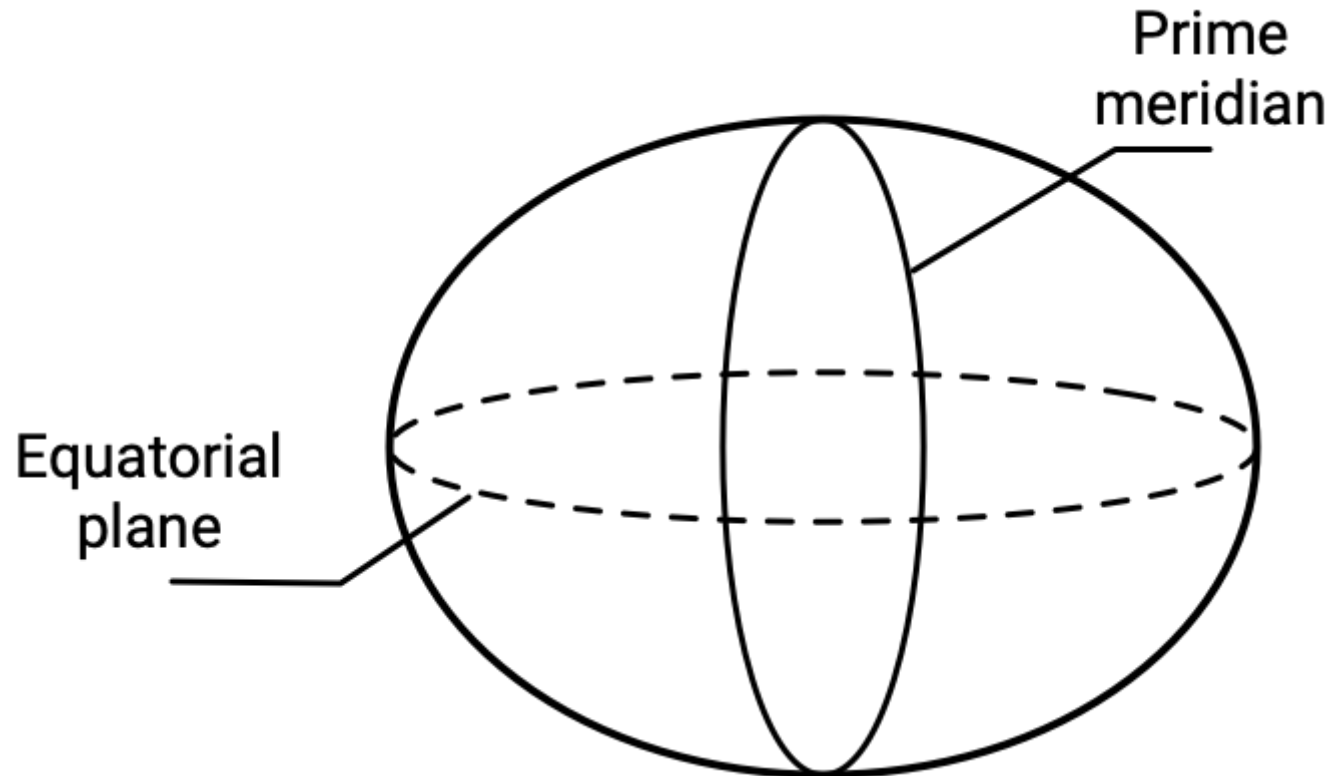
Projected coordinate reference systems

- All projected CRSs are based on a geographic CRS.

- Map projections convert the three-dimensional surface of the Earth into Easting and Northing (x and y) values (typically meters) in a projected CRS.

- These projected CRSs are based on Cartesian coordinates on a implicitly flat surface.

- Some deformations are introduced in the process, e.g. area, direction, distance or shape, while preserving one or two of these properties.
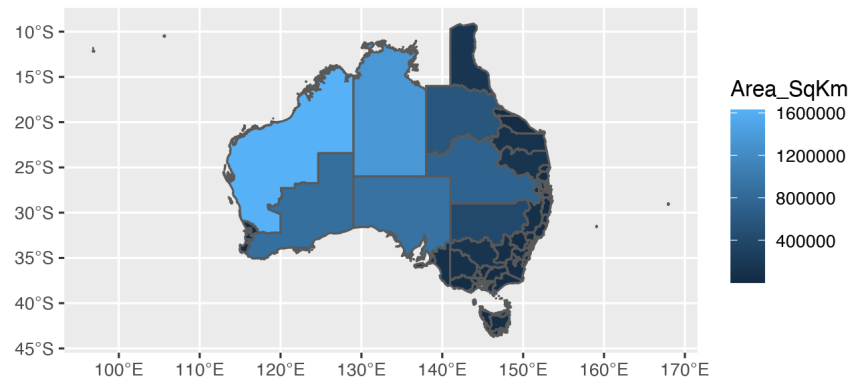
Well Known Text (WKT)

- Open Geospatial Consortium (OGC) developed an open standard format for describing CRSs called **WKT**

```
st_crs(ausmap)

## Coordinate Reference System:
##   User input: GDA94
##   wkt:
## GEOGCRS["GDA94",
##     DATUM["Geocentric Datum of Australia 1994",
##         ELLIPSOID["GRS 1980",6378137,298.257222101,
```

```
ggplot(ausmap) + geom_sf(aes(fill = Area_SqKm))
```

Changing map projections

- Map projections may be modified in multiple methods (it's beyond this unit to delve deep into this).

- Below uses the Lambert azimuthal equal-area projection centered on the longitude and latitude of (rough) Melbourne coordinates via `proj4string`:

```
ausmap %>%
  st_transform(crs = "+proj=laea +x_0=0 +y_0=0 +lon_0=145 +lat_0=-38") %>%
  ggplot() + geom_sf(aes(fill = Area_SqKm))
```
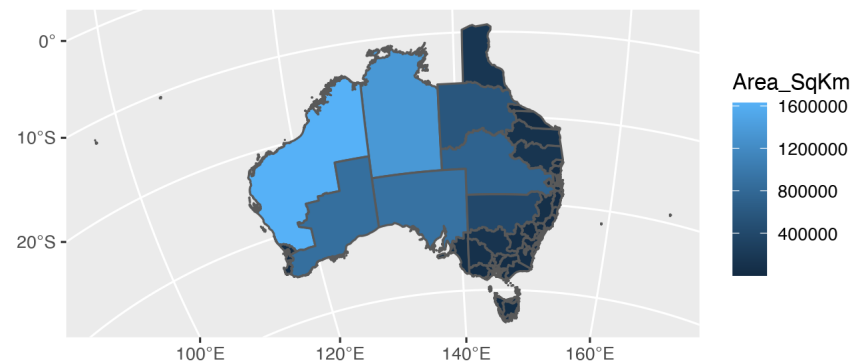
Summary

- We had a look a the 2019 federal election data

- We looked at visualising this data on a map in various ways

- We looked at reprojecting geographic data into different coordinate reference systems

Lecturer: *Emi Tanaka*

Department of Econometrics and Business Statistics

✉ ETC5512.Clayton-x@monash.edu

📅 Week 5