Data Visualisation

Chapter 6: Multivariate Strategies

Dr James Baglin

How to use these slides

Viewing slides…

- Press ‘f’ enable fullscreen mode
- Press ‘o’ or ‘Esc’ to enable overview mode
- Pressing ‘Esc’ exits all of these modes.
- Hold down ‘alt’ and click on any element to zoom in. ‘Alt’ + click anywhere to zoom back out.
- Use the Search box (top right) to search keywords in presentation

Printing slides…

- Click here to open a printable version of these slides.
- Right click and print from browser or save as PDF (e.g. Chrome)

Multivariate Thinking

- The following demonstration will highlight the importance of multivariate thinking and data visualisation
- We will explore a number of multivariate data visualisation strategies/methods:
  - Mapping additional aesthetics
  - Faceting
  - 3D scatter

FEV Data

- The `FEV.csv` dataset contains the forced expiratory volume (FEV), smoking status, age, height and sex of a sample of 654 children and adolescents aged between 3 and 19 (M = 9.93, SD = 2.95)

- Sample consisted of 589 nonsmokers (90%) and 65 smokers (10%)
- Do children or adolescent smokers tend to have lower FEV readings than non-smokers?

Bivariate Thinking

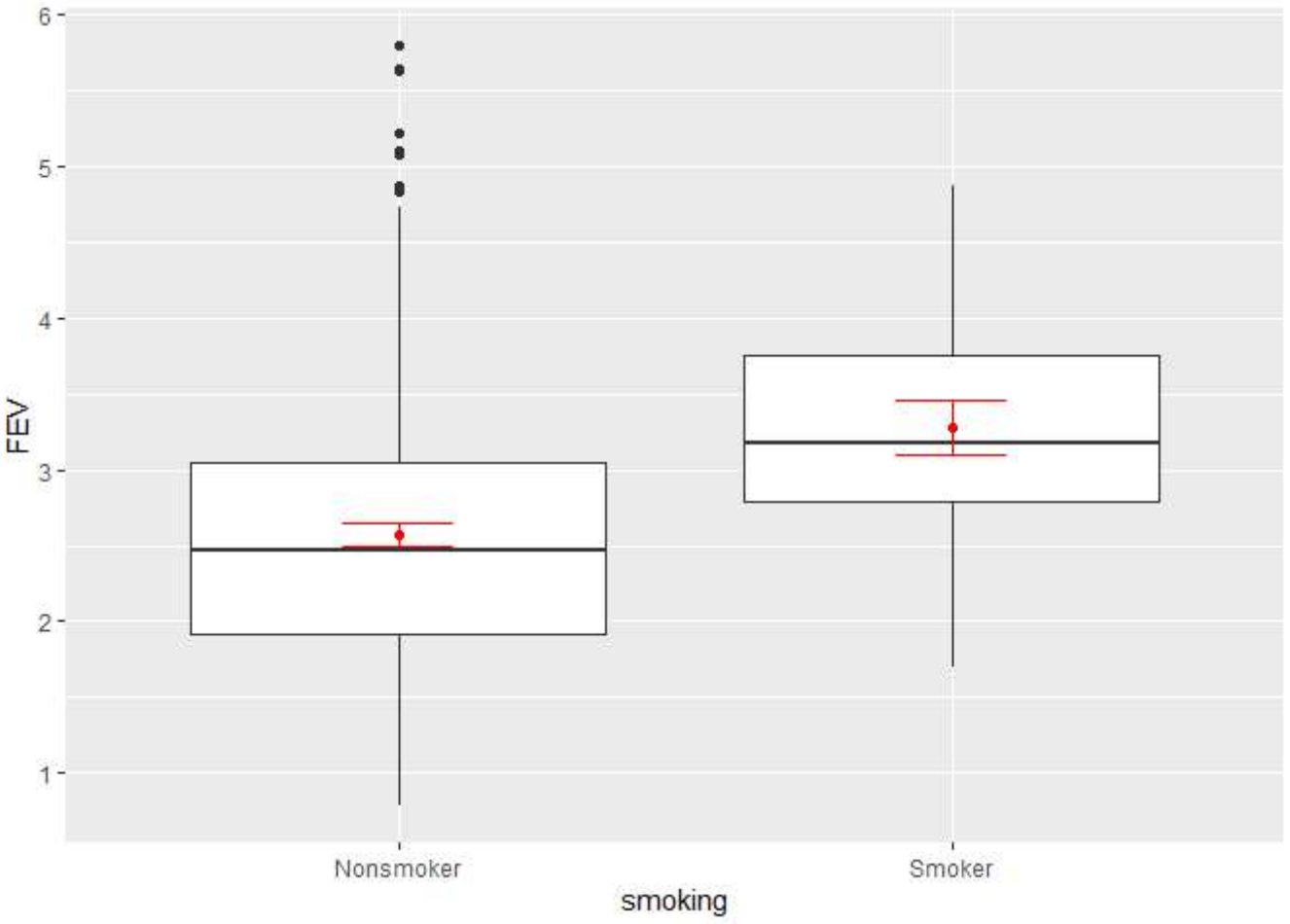
- Let's start with a side-by-side box plot with mean and error bars (95% CI)

```
FEV = read.csv("../data/FEV.csv")
p1 <- ggplot(data = FEV, aes(x = smoking, y = FEV))
p1 + geom_boxplot() + stat_summary(fun.y = "mean", geom = "point",
                                   colour = "red") +
  stat_summary(fun.data = "mean_cl_boot", colour = "red",
               geom = "errorbar", width = .2)
```

- Plot on next slide…

Bivariate Thinking 2

- Huh?



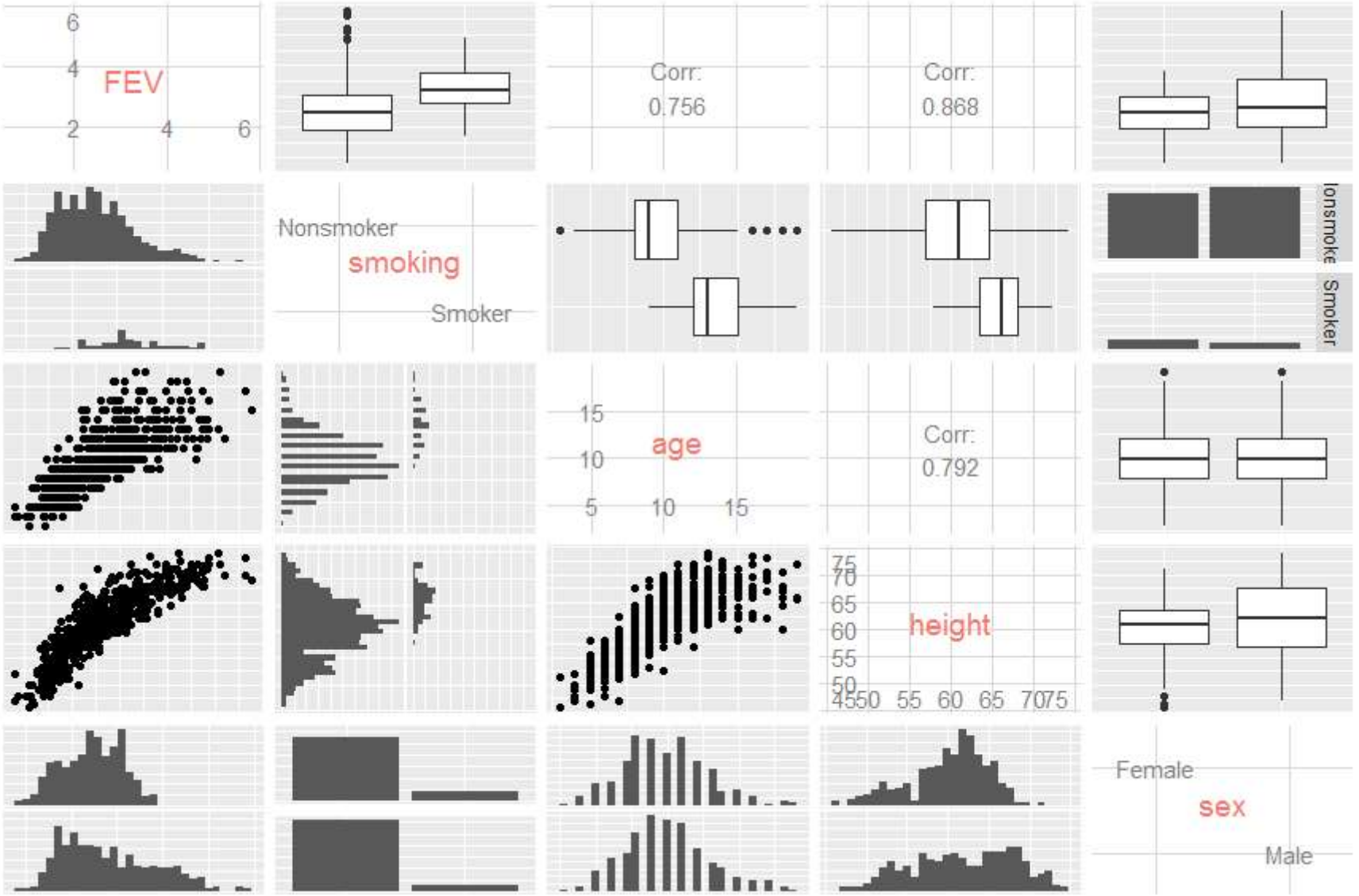
Exploring Covariation

- We need to consider other variables that might be at play…

```
library(GGally)
ggpairs(FEV, columns = 1:5,axisLabels = "internal")
```

- Plot on next slide…

Exploring Covariation 2



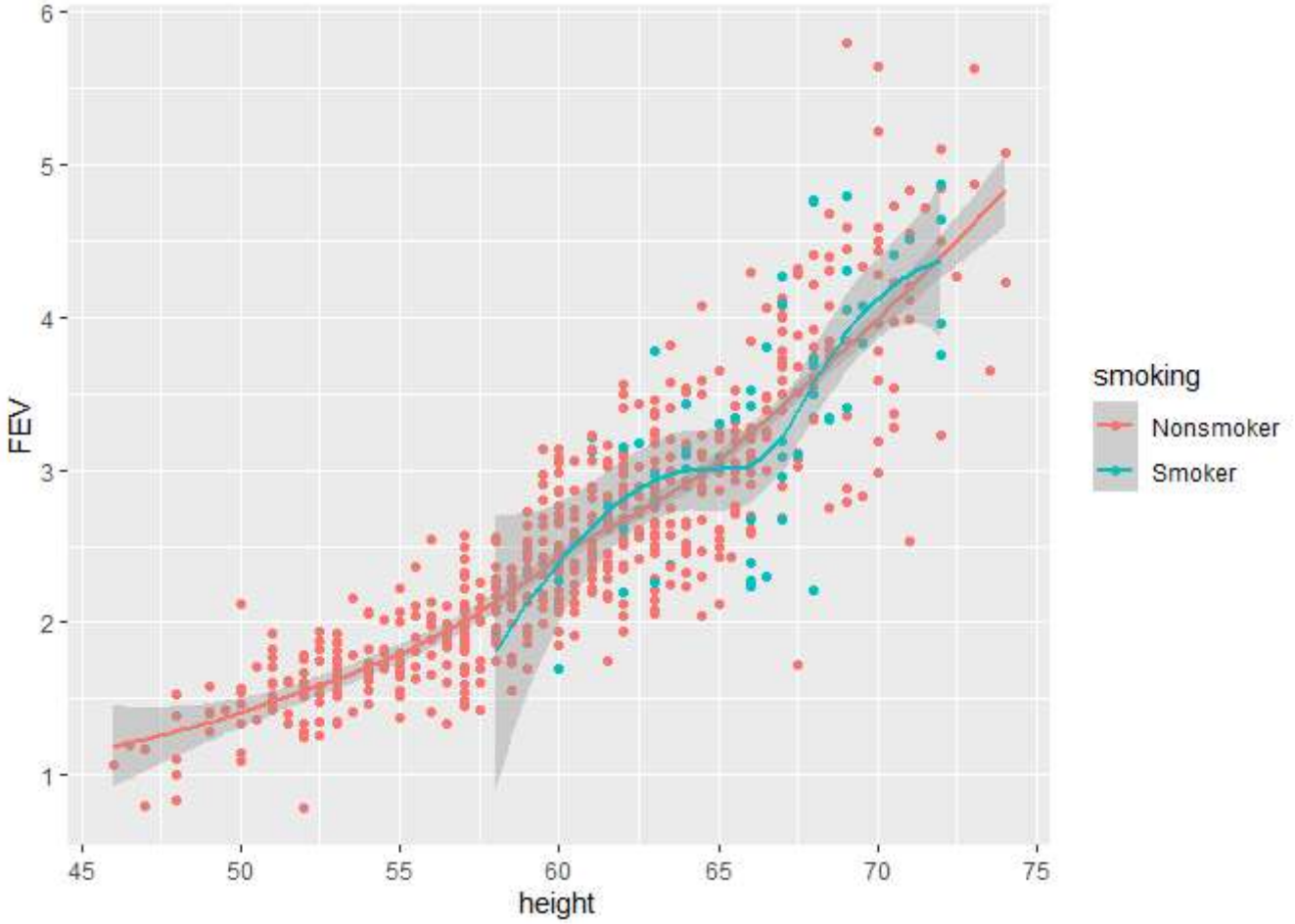
Multivariate Data Visualisation

- Does height explain the difference between FEV for smokers and nonsmokers?

```
p2 <- ggplot(data = FEV, aes(x = height, y = FEV, colour = smoking))
p2 + geom_point() + geom_smooth()
```

- Plot on next slide…

Multivariate Data Visualisation 2



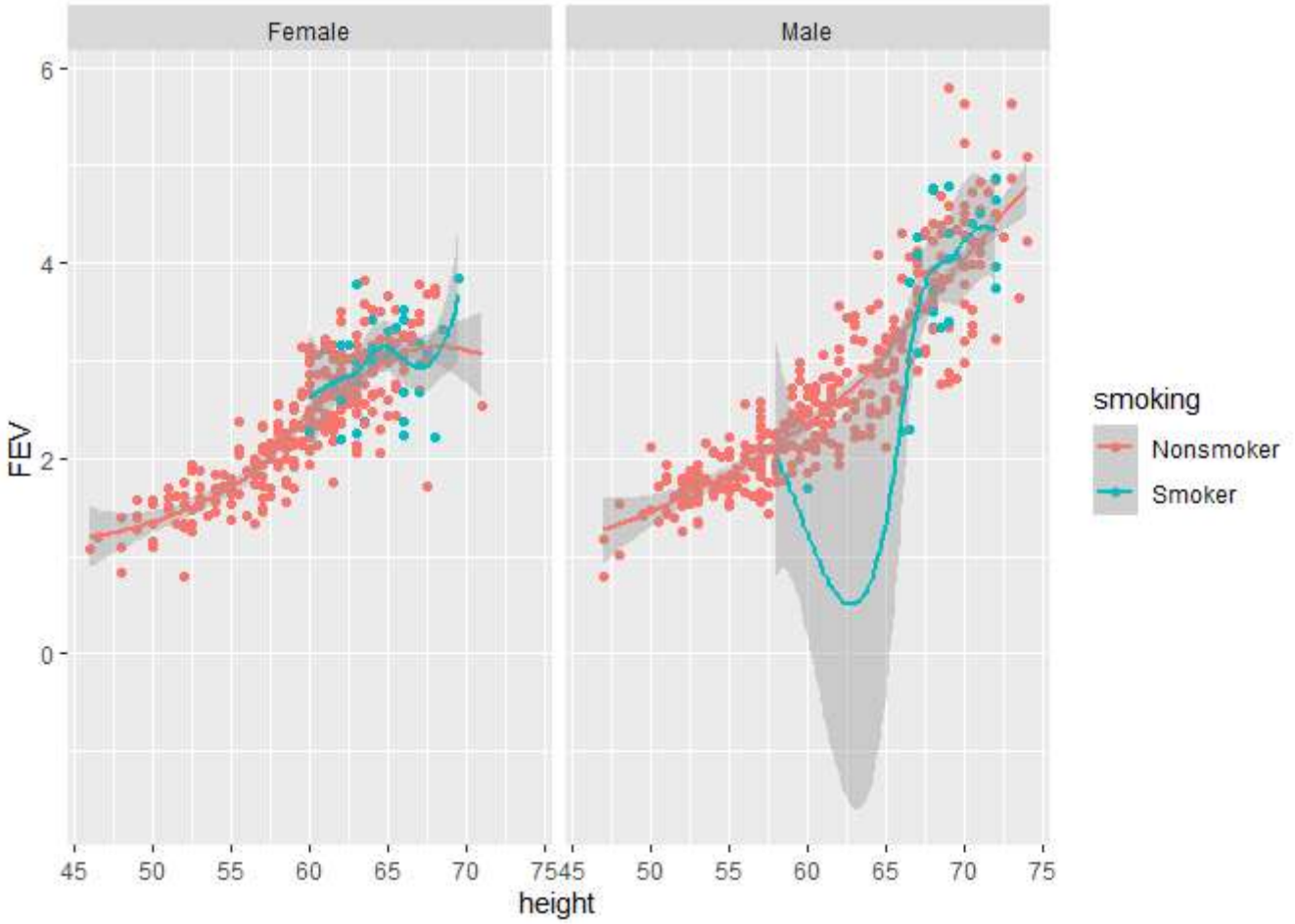
Multivariate Data Visualisation 3

- Does it also depend on gender?

```
p3 <- ggplot(data = FEV, aes(x = height, y = FEV, colour = smoking))
p3 + geom_point() + geom_smooth() + facet_grid(. ~ sex)
```

- Plot on next slide…

Multivariate Data Visualisation 4



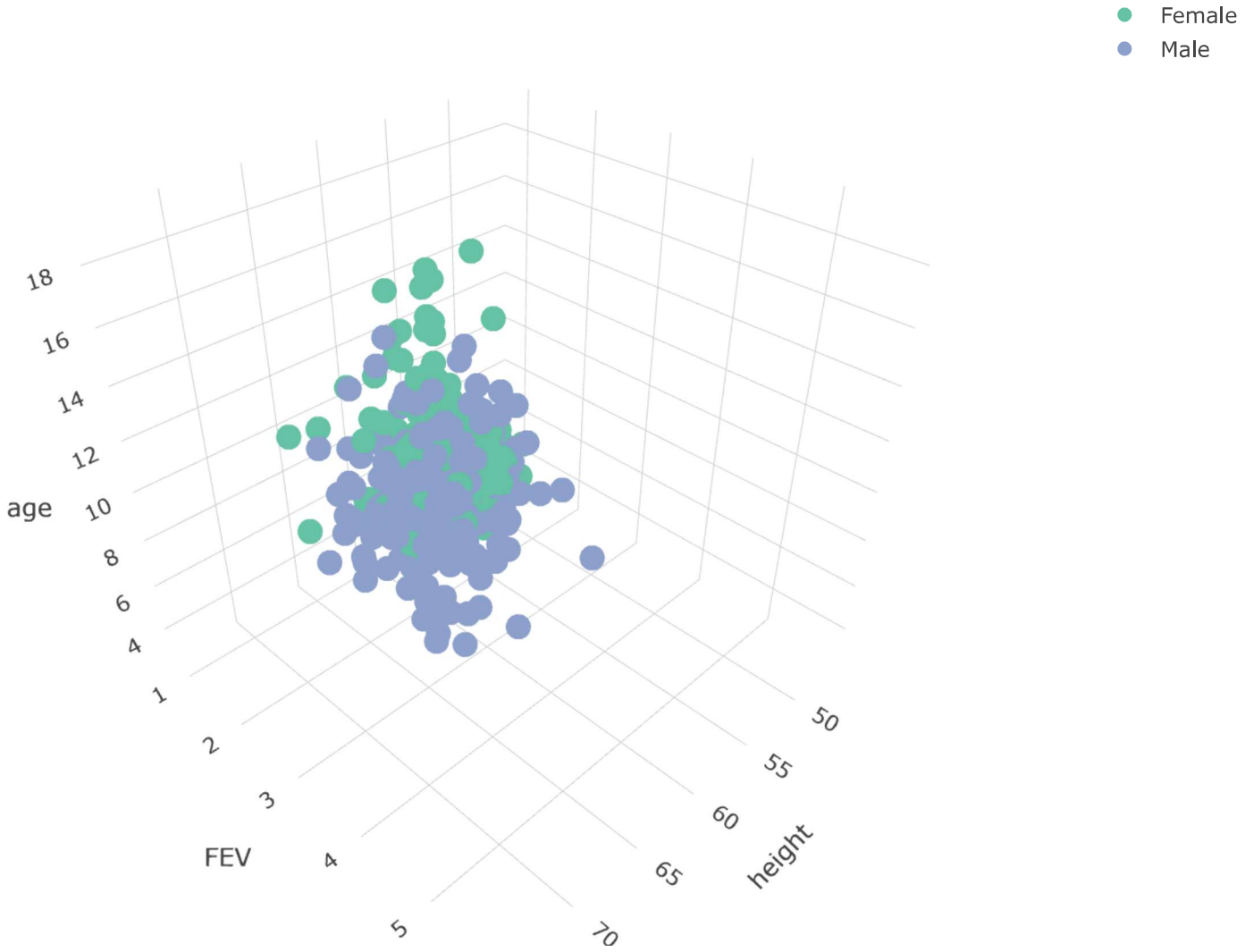
Multivariate Data Visualisation 5

- `p1` suggested that smokers had better FEV than non-smokers!
- However, it was apparent that many other variables were at play.
- Age, height and gender are also related to FEV, however, all these variables are interrelated.
- Can we visualise all these relationships in one plot?

```
library(plotly)
plot_ly(data = FEV, x = ~height, y = ~FEV, z = ~age, color = ~sex,
        width = 800*1.5, height = 600*1.5) %>%
  add_markers()
```

- Plot on next slide...

Multivariate Data Visualisation 6



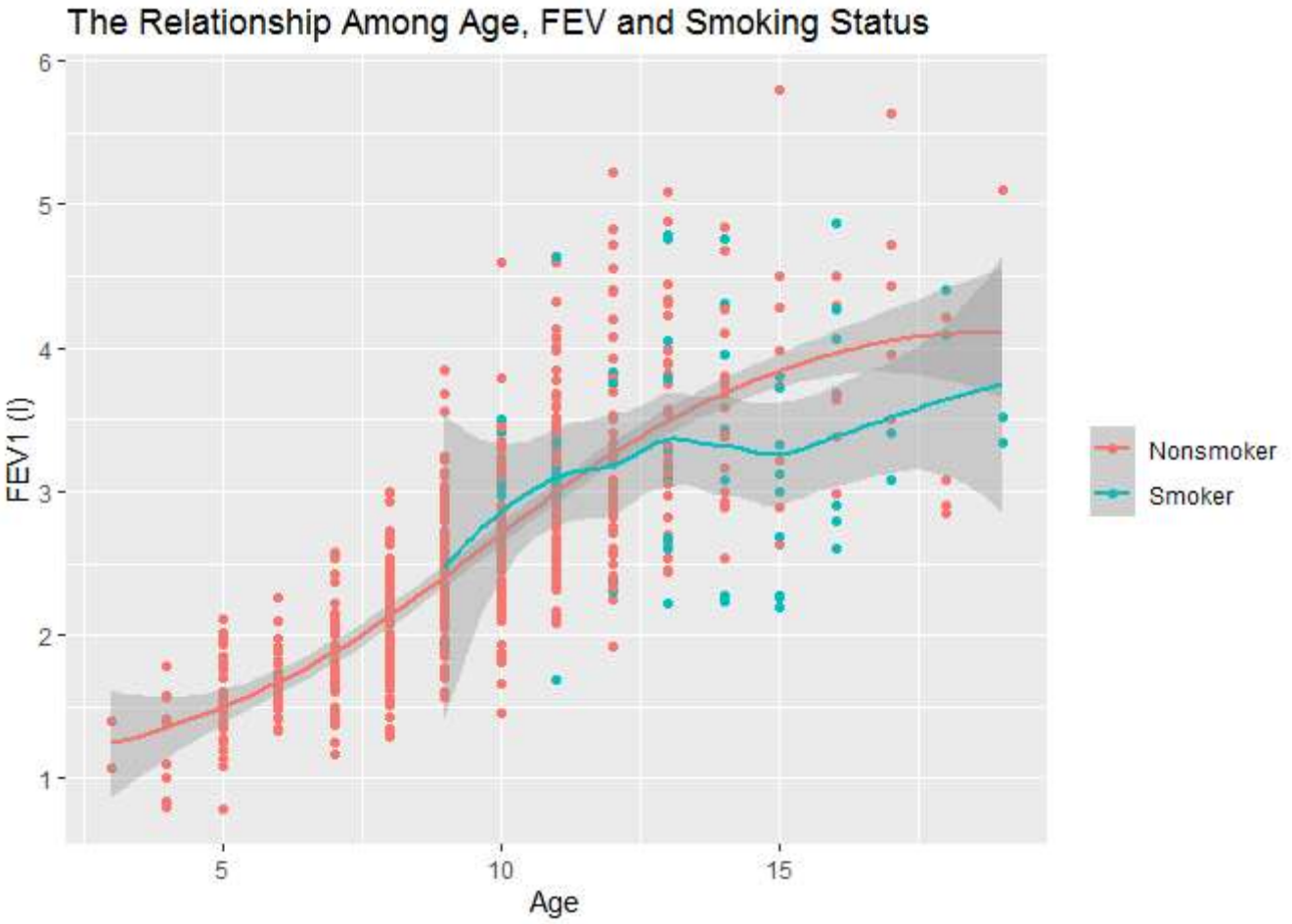
Multivariate Thinking 2

- The world is not bivariate.
- Our world is far more interesting and complex.
- The aim of multivariate data visualisation is to see through this complexity

```
p4 <- ggplot(data = FEV, aes(x = age, y = FEV, colour = smoking))
p4 + geom_point() + geom_smooth() +
  labs(x = "Age", y = "FEV1 (l)",
       title = "The Relationship Among Age, FEV and Smoking Status") +
  scale_colour_discrete(name = "")
```

- Plot on next slide...

Multivariate Thinking 3

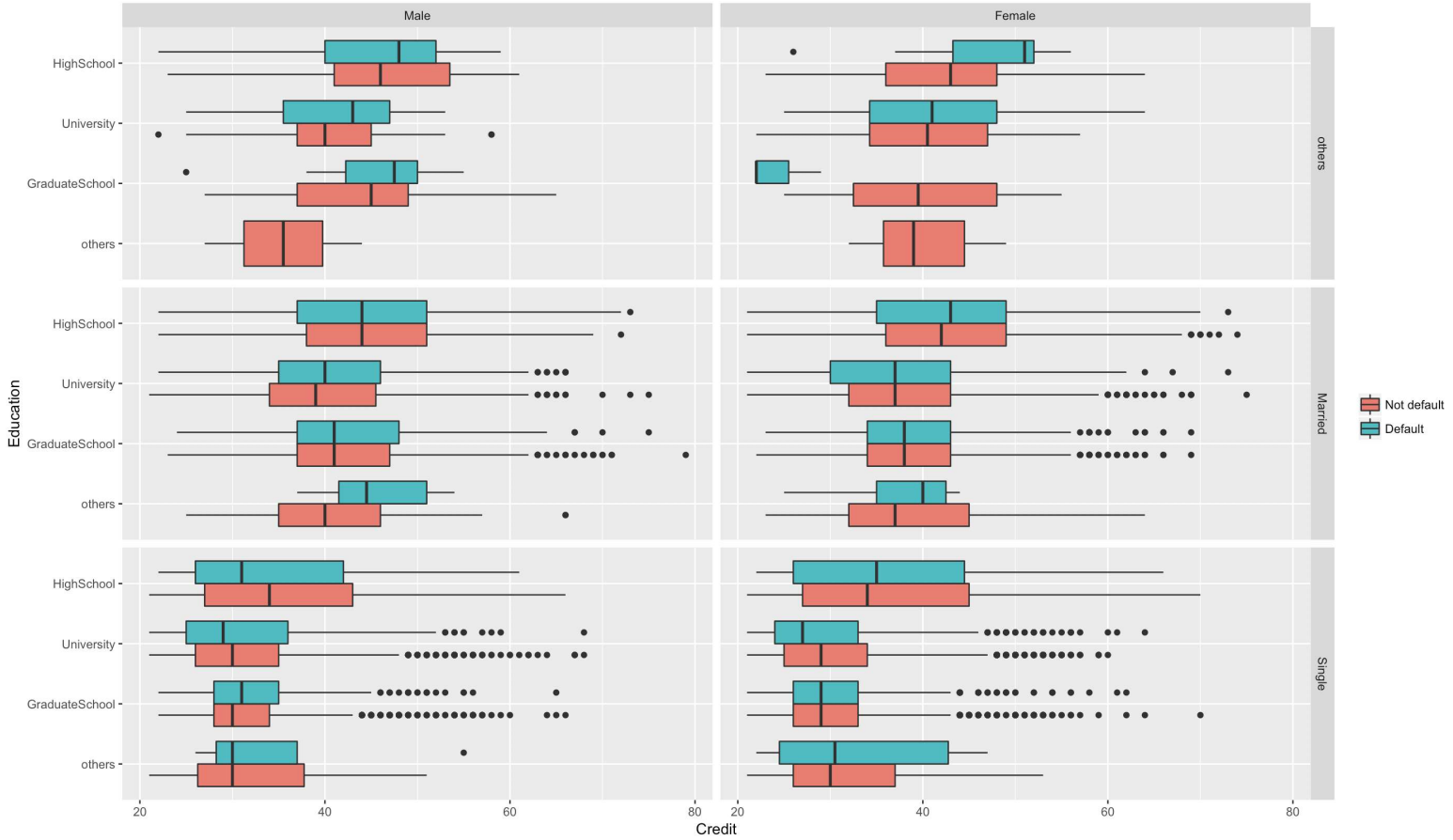


Multivariate Data Visualisations

- There are several strategies for visualising multivariate data, which can be grouped into the following major categories:
  - **Mapping additional aesthetics**: `ggplot2`: `x`, `y` (position), `colour`, `fill`, `shape`, `size`, `weight`, `linetype`, `alpha`… No more than three in one plot (Wickham 2010).
  - **Faceting**: Breaking visualisation into small multiples. No more than two variables.
  - **Purpose built**: Many examples: Sankey diagrams, parallel coordinates, 3D scatter plots and multivariate mosaic plots…
  - **Animation**: Change or transition mapped to a variable, e.g. time (Chapters 8 & 9).

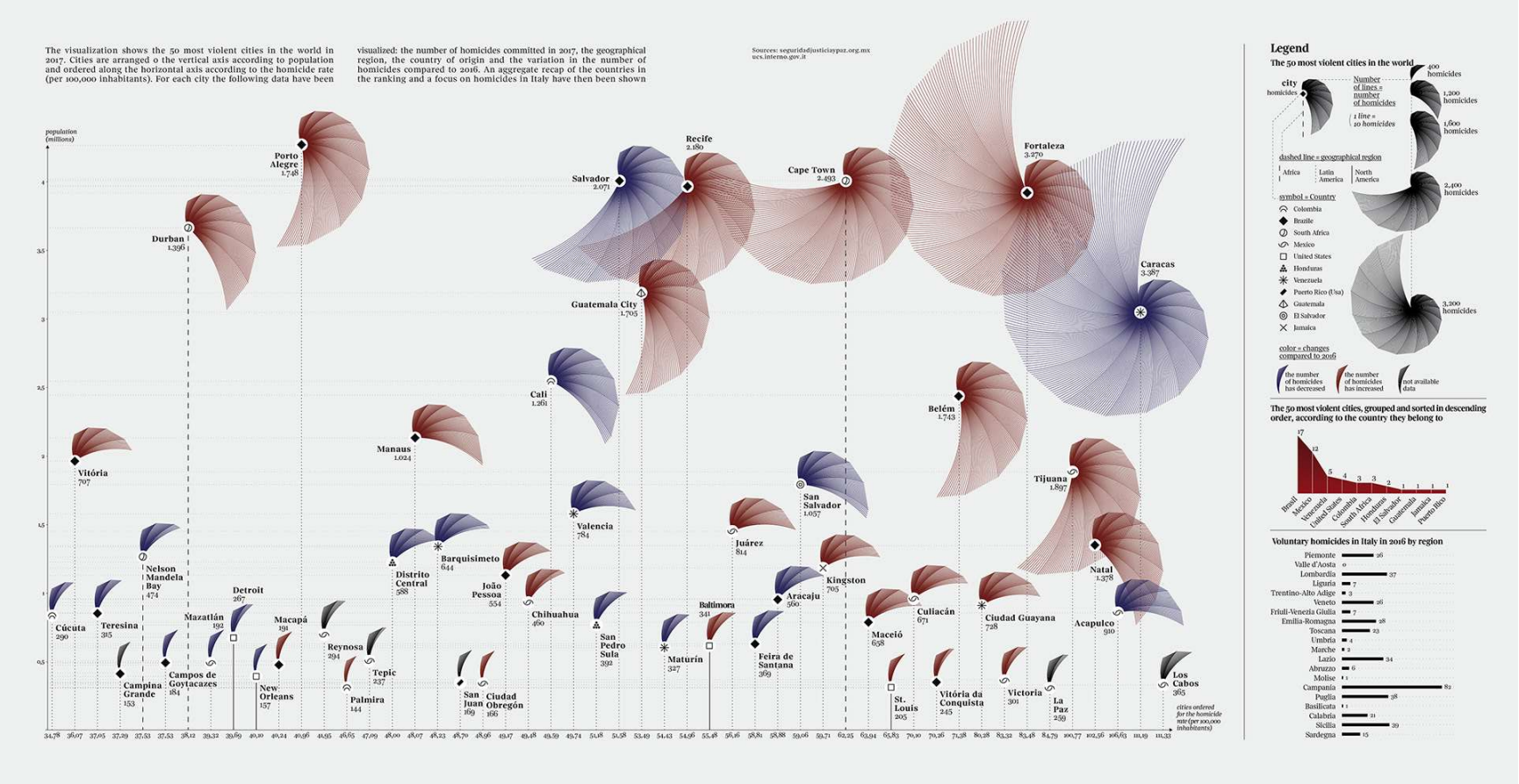
Rule of Thumb

- Count how many variables are contained in the following visualisation



The Multivariate Dilemma

- Violent Cities (Fragapane 2018)



Clarify the Objective

- To educate the audience about the cities with the highest rates of homicide.
  - Rank cities by homicide rate
  - Show population size
  - Show total homicides
  - Show country/region
  - Change (Increased or decreased from 2016) - no data*

Data

- Data was sourced from Citizen Council for Public Security and Criminal Justice AC (2018).
- Download a clean copy here.

```
library(readxl)
top50 <- read_excel("../data/top_50_cities_by_homicide_2018.xlsx")
```

- Original ranks were incorrect

```
top50$Rank = rank(top50$Rate)
```

- Reorder ranks into ascending order

```
library(forcats)
top50$City <- fct_reorder(top50$City, top50$Rank, desc = FALSE)
```

Data - Collapse Countries

- Convert population unit to millions - easier to plot

```
top50$Population <- top50$Population/1000000
```

- Collapse low tally countries into “other” category

```
table(top50$Country)
```

```
##
##         Brazil       Colombia    El Salvador      Guatemala       Honduras
##             14              2              1              1              2
##        Jamaica         Mexico   South Africa  United States      Venezuela
##              1             15              3              5              6
```

Data - Collapse Countries Cont.

```
top50$Country_2<- fct_collapse(top50$Country,
  Brazil = "Brazil",
  Mexico = "Mexico",
  United_States = "United States",
  Venezuela = "Venezuela",
  South_Africa = "South Africa",
  Other = c("Colombia", "El Salvador","Guatemala","Honduras","Jamaica")
)

table(top50$Country_2)
```

```
##
##        Brazil         Other        Mexico  South_Africa United_States
##            14             7            15             3             5
##     Venezuela
##             6
```

Data - Collapse Countries Cont.2

- Order and label levels.

```
top50$Country_2 <- factor(top50$Country_2,
                          levels = c("Mexico",
                                     "Brazil",
                                     "Venezuela",
                                     "Other",
                                     "United_States",
                                     "South_Africa"
                                     ),
                          labels = c("Mexico",
                                     "Brazil",
                                     "Venezuela",
                                     "Other*",
                                     "United States",
                                     "South Africa"
                                     ))
table(top50$Country_2)
```

```
##
##         Mexico         Brazil      Venezuela         Other* United States
##             15             14              6              7              5
##   South Africa
##              3
```

Data - Wide to Long Data Format

- Data is in a wide format

```
library(data.table)
top50
```

```
## # A tibble: 50 x 8
##     Rank City       Country  Region   Homicides Population  Rate Count
##    <dbl> <fct>      <chr>    <chr>        <dbl>      <dbl> <dbl> <fct>
##  1    50 Tijuana    Mexico   Latin A~      2640      1.91  138.  Mexic
##  2    49 Acapulco   Mexico   Latin A~       948      0.858 111.  Mexic
##  3    48 Caracas    Venezue~ Latin A~      2980      2.98  100.0 Venez
##  4    47 Ciudad Vi~ Mexico   Latin A~       314      0.365  86.0 Mexic
##  5    46 Ciudad Ju~ Mexico   Latin A~      1251      1.46   85.6 Mexic
##  6    45 Irapuato   Mexico   Latin A~       473      0.581  81.4 Mexic
##  7    44 Ciudad Gu~ Venezue~ Latin A~       645      0.824  78.3 Venez
##  8    43 Natal      Brazil   Latin A~      1185      1.59   74.7 Brazi
##  9    42 Fortaleza  Brazil   Latin A~      2724      3.94   69.1 Brazi
## 10    41 Ciudad Bo~ Venezue~ Latin A~       264      0.382  69.1 Venez
## # ... with 40 more rows
```

Data - Wide to Long Data Format Cont.

- Restructure data to long format

Wide Data Format

ID	Var1	Var2	Var3
1	2	5	5
2	4	5	6
3	4	4	5
4	2	1	9
5	1	8	3
6	3	4	5
7	5	5	4
8	7	2	3
9	2	3	1

```
> gather(df, Variable,
  Values, Var1:Var3)
```

Long Data Format

ID	Variable	Values
1	Var1	2
2	Var1	4
3	Var1	4
4	Var1	2
5	Var1	1
6	Var1	3
7	Var1	5
8	Var1	7
9	Var1	2
1	Var2	5
2	Var2	5
3	Var2	4
4	Var2	1
5	Var2	8
6	Var2	4
7	Var2	5
8	Var2	2
9	Var2	3
1	Var3	5
2	Var3	6
3	Var3	5
4	Var3	9
5	Var3	3
6	Var3	5
7	Var3	4
8	Var3	3
9	Var3	1

Data - Wide to Long Data Format Cont. 2

- Easy to do using the `tidyr::gather()` function.

```
library(tidyr)
top50_long <- gather(top50, key = "Variable", value = "Value", Homicides
```

```
## # A tibble: 150 x 7
##     Rank City            Country   Region        Country_2 Variable  V
##    <dbl> <fct>           <chr>     <chr>         <fct>     <chr>     <
##  1    50 Tijuana        Mexico    Latin America Mexico    Homicides
##  2    49 Acapulco       Mexico    Latin America Mexico    Homicides
##  3    48 Caracas        Venezuela Latin America Venezuela Homicides
##  4    47 Ciudad Victoria Mexico   Latin America Mexico    Homicides
##  5    46 Ciudad Juárez  Mexico    Latin America Mexico    Homicides
##  6    45 Irapuato       Mexico    Latin America Mexico    Homicides
##  7    44 Ciudad Guayana Venezuela Latin America Venezuela Homicides
##  8    43 Natal          Brazil    Latin America Brazil    Homicides
##  9    42 Fortaleza      Brazil    Latin America Brazil    Homicides
## 10    41 Ciudad Bolívar Venezuela Latin America Venezuela Homicides
## # ... with 140 more rows
```

Data - Add Rank Labels and Variable Labels

- Add a new rank label variable in order to label a county's rank on each variable.

```
top50_long$Rank_labels <- c(rank(-top50$Homicides,ties.method = "min"),
                            rank(-top50$Population,,ties.method = "min"),
                            rank(-top50$Rate,ties.method = "min"))
```

- Clean up factor labels for visualisation.

```
top50_long$Variable <- factor(top50_long$Variable,
                              levels = c("Rate","Homicides","Population"),
                              labels = c("Homicides per 100,000",
                                         "Total Homicides",
                                         "Population Size (Millions)"))
```

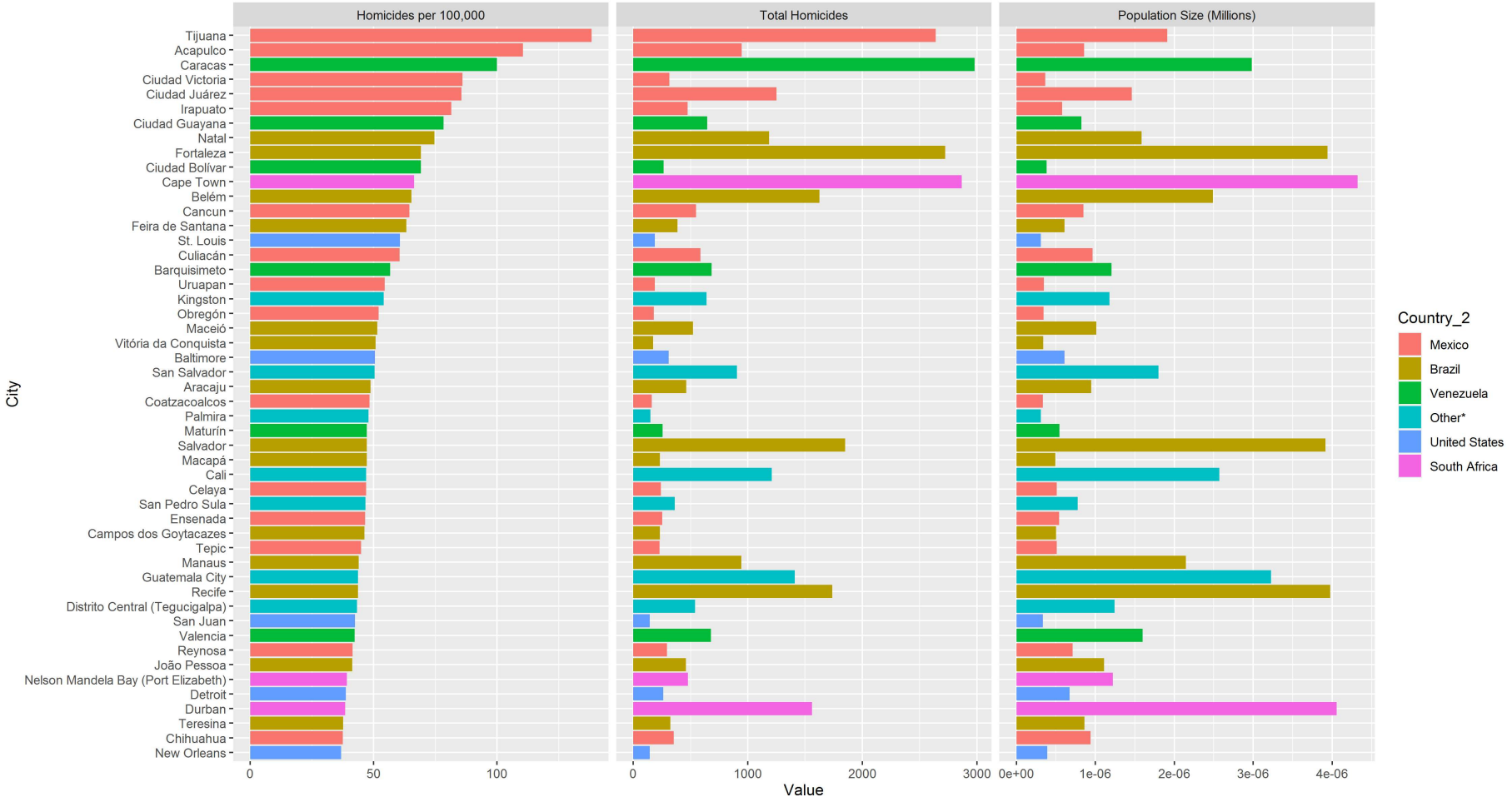
Basic Visualisation

- Make use of facets to align, but isolate each variable.

```
p <- ggplot(data = top50_long,
            aes(x = City, y = Value, fill = Country_2)) +
  geom_bar(stat = "identity") + coord_flip() +
  facet_grid(.~Variable, scales = "free")
```

- Result on next slide…

Basic Visualisation Cont.

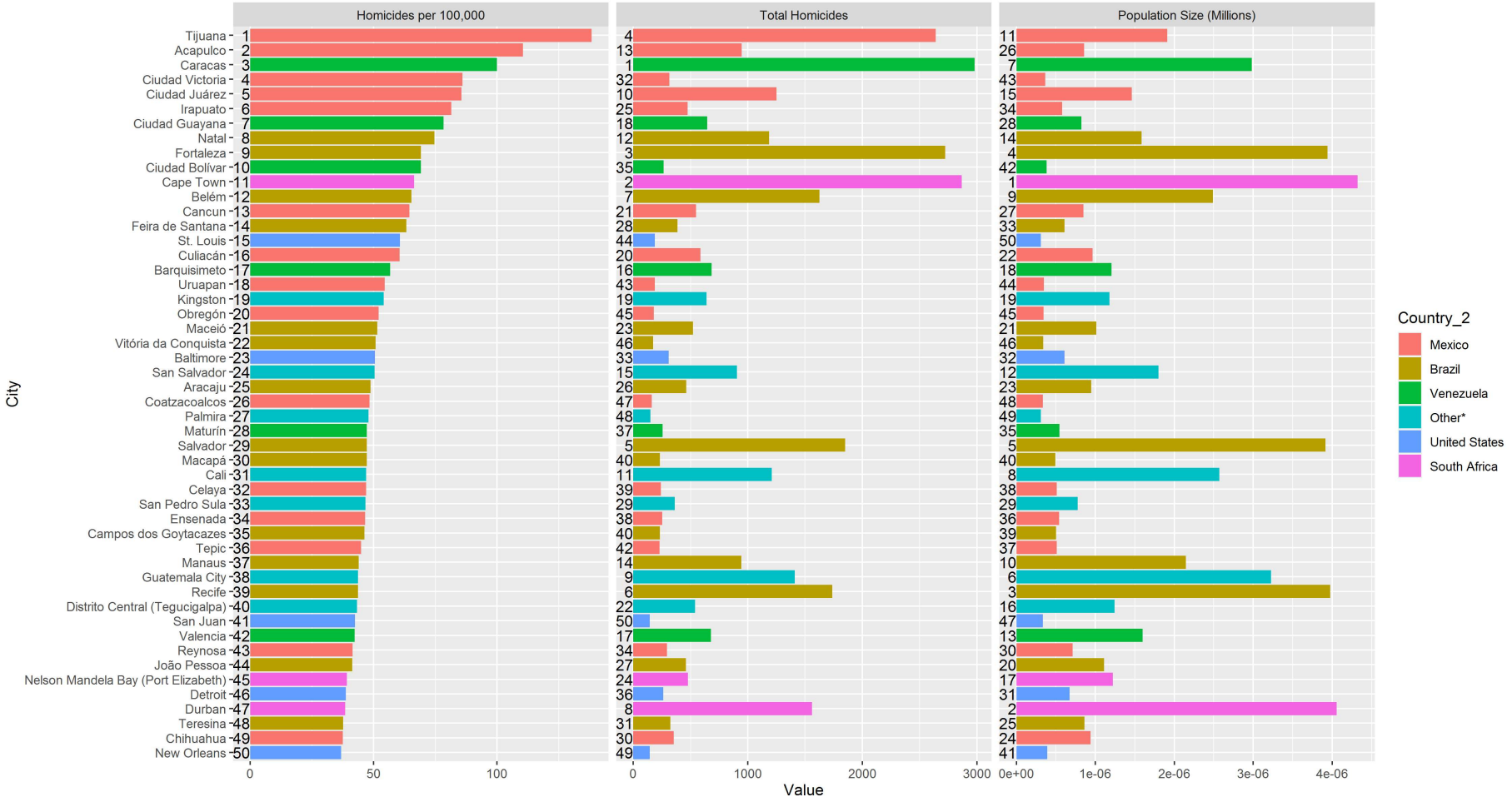


Add rank labels

```
p <- p +
  geom_text(aes(label = Rank_labels, y = 0),
            fill = "gray", hjust = "top", family="Georgia")
```

- Result on next slide…

Add rank labels Cont.

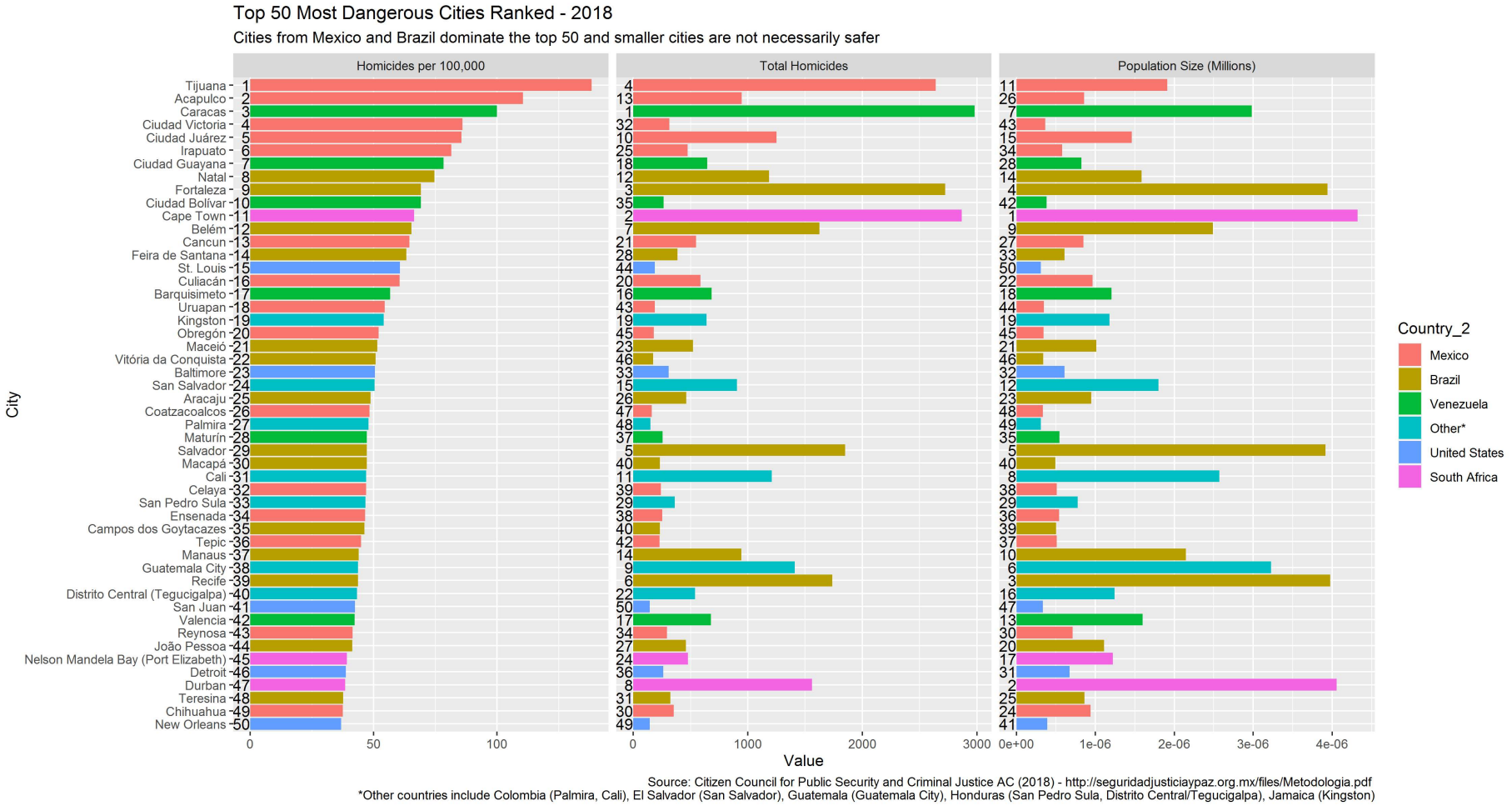


Add Details

```
p <- p +
  labs(title = "Top 50 Most Dangerous Cities Ranked - 2018",
       subtitle  = "Cities from Mexico and Brazil dominate the top 50 and
       caption = "Source: Citizen Council for Public Security and Crimina
       *Other countries include Colombia (Palmira, Cali), El Salvador (Sa
```

- Result on next slide…

Add Details Cont.



Source: Citizen Council for Public Security and Criminal Justice AC (2018) - http://seguridadjusticiaypaz.org.mx/files/Metodologia.pdf
*Other countries include Colombia (Palmira, Cali), El Salvador (San Salvador), Guatemala (Guatemala City), Honduras (San Pedro Sula, Distrito Central/Tegucigalpa), Jamaica (Kingston)

Adjust Colour, Font and Clean-up

- We can closely replicate a colour theme by obtaining colours from the original
  - background - #EFF1F0
  - red - #89141C
  - blue - #29265B
- Use coolors website to create a colour palette.

```
background <- "#EFF1F0"

pal <- c("#89141cff",
         "#b16264ff",
         "#d8b0acff",
         "#d8d0c1ff",
         "#3a506bff",
         "#29265bff"
)
```

Adjust Colour, Font and Clean-up Cont.

- The original data visualisation used a font similar to Georgia.
- To access a greater range of fonts for `ggplot2` use the `extrafont` package.

```
install.packages("extrafont")
library(extrafont)
font_import()
```

Adjust Colour, Font and Clean-up Cont. 2

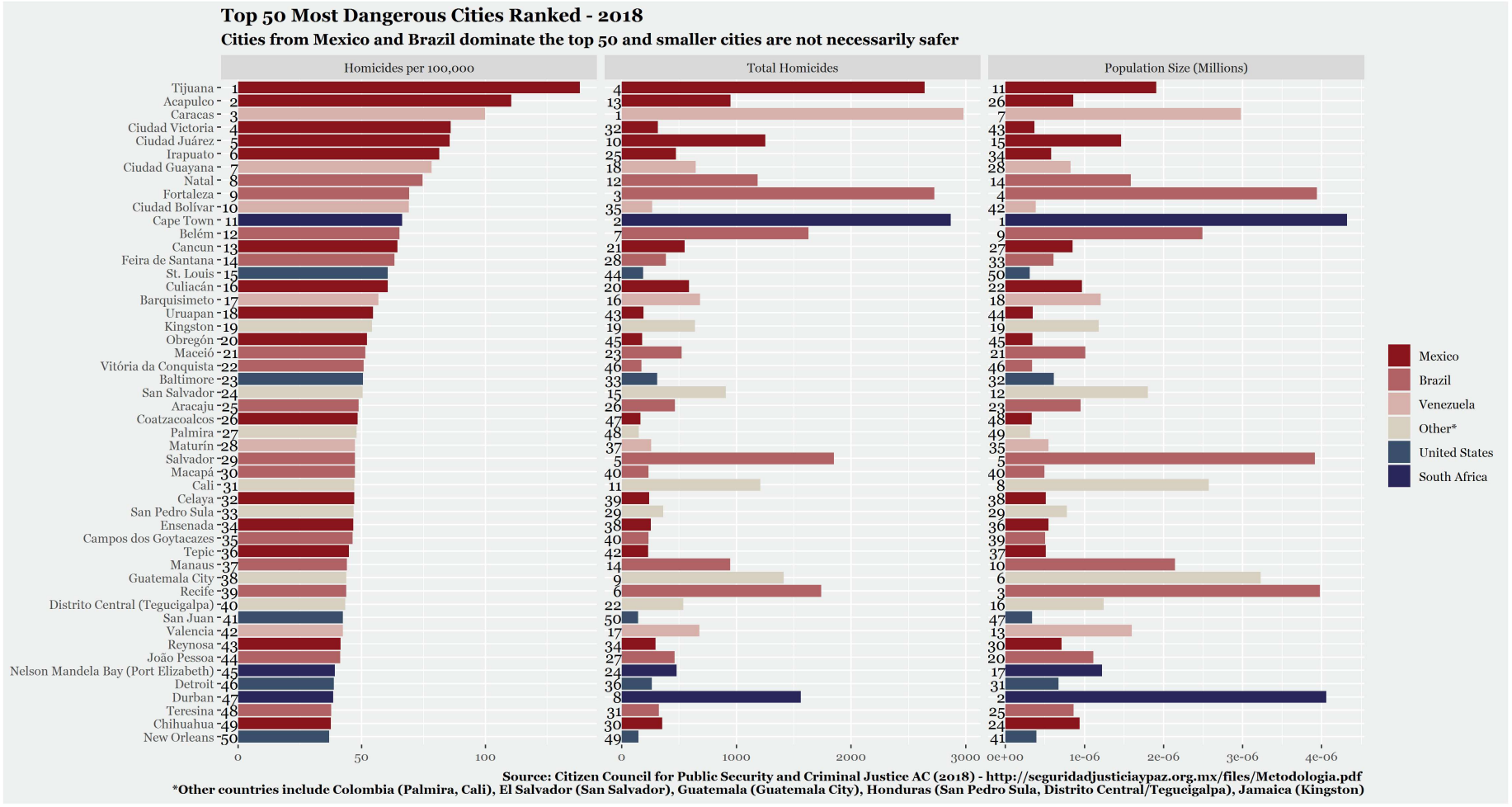
- Putting this all together…

```
p <- p +
  theme_gray() +
  scale_fill_manual(values = pal) +
  theme(plot.background = element_rect(fill = background),
        panel.background = element_rect(fill = background),
        legend.background = element_rect(fill = background),
        text=element_text(family="Georgia"),
        title = element_text(face = "bold"),
        axis.title.x = element_blank(),
        axis.title.y = element_blank(),
        legend.title = element_blank())
```

- Result on next slide…

Adjust Colour, Font and Clean-up Cont. 3

- Best viewed full-screen



Multivariate Data Visualisation Topics

- Chapter 6 contains many other case studies of multivariate data visualisations. Work through these in your own time.
  - Using the size and colour aesthetics in `ggplot2`
  - Using faceting
  - Purpose built
    - Sankey diagrams
    - Parallel coordinates
    - Multivariate mosaic plots
- These techniques may come in handy for your assignments.
- Enjoy :)

References

Citizen Council for Public Security and Criminal Justice AC. 2018. “Ranking methodology (2018) of the 50 most violent cities in the world.” http://seguridadjusticiaypaz.org.mx/files/Metodologia.pdf.

Fragapane, Federica. 2018. “The most violent cities in the world.” https://www.behance.net/gallery/70033395/The-Most-Violent-Cities/ .

Wickham, H. 2010. “A layered grammar of graphics.” *Journal of Computational and Graphical Statistics* 19 (1): 3–28. doi:10.1198/jcgs.2009.07098.