# Graduate Seminar

## Excel Import Demo

### Deaths Data set

| | A | B | C | D | E | F |
|---|---|---|---|---|---|---|
| 1 | For the sake | | | | | |
| 2 | | of consistency | | | in the | data layout, |
| 3 | which is really | | | | a | beautiful thing, |
| 4 | I will | keep making notes | | | | up here. |
| 5 | Name | Profession | Age | Has kids | Date of birth | Date of death |
| 6 | Vera Rubin | scientist | 88 | TRUE | 1928-07-23 | 2016-12-25 |
| 7 | Mohamed Ali | athlete | 74 | TRUE | 1942-01-17 | 2016-06-03 |
| 8 | Morley Safer | journalist | 84 | TRUE | 1931-11-08 | 2016-05-19 |
| 9 | Fidel Castro | politician | 90 | TRUE | 1926-08-13 | 2016-11-25 |
| 10 | Antonin Scalia | lawyer | 79 | TRUE | 1936-03-11 | 2016-02-13 |
| 11 | Jo Cox | politician | 41 | TRUE | 1974-06-22 | 2016-06-16 |
| 12 | Janet Reno | lawyer | 78 | FALSE | 1938-07-21 | 2016-11-07 |
| 13 | Gwen Ifill | journalist | 61 | FALSE | 1955-09-29 | 2016-11-14 |
| 14 | John Glenn | astronaut | 95 | TRUE | 1921-07-28 | 2016-12-08 |
| 15 | Pat Summit | coach | 64 | TRUE | 1952-06-14 | 2016-06-28 |
| 16 | This | | | | | |
| 17 | | has been really fun, but | | | | |
| 18 | we're signing | | | | | |
| 19 | | | off | | now! | |
| 20 | | | | | | |

Figure 1:

- Environment Tab –> Import Dataset

### First attempt

```
library(readxl)
deaths <- read_excel("~/Graduate_Seminar_Presentation/deaths.xlsx")

deaths
```

```
## # A tibble: 18 x 6
##                `Lots of people`                    X__1   X__2     X__3
##                           <chr>                   <chr>  <chr>    <chr>
##  1 simply cannot resist writing                     <NA>   <NA>     <NA>
##  2                           at                      the    top     <NA>
##  3                           or                  merging   <NA>     <NA>
##  4                         Name               Profession    Age Has kids
```

```
##  5                 David Bowie                musician     69     TRUE
##  6               Carrie Fisher                   actor     60     TRUE
##  7                 Chuck Berry                musician     90     TRUE
##  8                 Bill Paxton                   actor     61     TRUE
##  9                      Prince                musician     57     TRUE
## 10                Alan Rickman                   actor     69    FALSE
## 11           Florence Henderson                   actor     82     TRUE
## 12                  Harper Lee                  author     89    FALSE
## 13               Zsa Zsa Gábor                   actor     99     TRUE
## 14               George Michael                musician     53    FALSE
## 15                         Some                    <NA>   <NA>     <NA>
## 16                         <NA> also like to write stuff   <NA>     <NA>
## 17                         <NA>                    <NA> at the  bottom,
## 18                         <NA>                    <NA>   <NA>     <NA>
## # ... with 2 more variables: X__4 <chr>, X__5 <chr>
```

### Second (& successful) attempt

```
library(readxl)
deaths <- read_excel("~/Graduate_Seminar_Presentation/deaths.xlsx",
                     range = cell_rows(5:15))

deaths
```

```
## # A tibble: 10 x 6
##                  Name Profession   Age `Has kids` `Date of birth`
##                 <chr>      <chr> <dbl>      <lgl>          <dttm>
##  1        David Bowie   musician    69       TRUE      1947-01-08
##  2      Carrie Fisher      actor    60       TRUE      1956-10-21
##  3        Chuck Berry   musician    90       TRUE      1926-10-18
##  4        Bill Paxton      actor    61       TRUE      1955-05-17
##  5             Prince   musician    57       TRUE      1958-06-07
##  6       Alan Rickman      actor    69      FALSE      1946-02-21
##  7 Florence Henderson      actor    82       TRUE      1934-02-14
##  8         Harper Lee     author    89      FALSE      1926-04-28
##  9      Zsa Zsa Gábor      actor    99       TRUE      1917-02-06
## 10     George Michael   musician    53      FALSE      1963-06-25
## # ... with 1 more variables: `Date of death` <dttm>
```

## Data Munging Example

### Pipe Operator and mutate

Create a new column: `birthplace`

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
```

```
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
deaths %>%
  mutate(birthplace = c("UK", "US", "US", "US", "US",
                        "UK", "US", "US", "Hungary",
                        "UK"))
```

```
## # A tibble: 10 x 7
##                  Name Profession   Age `Has kids` `Date of birth`
##                 <chr>      <chr> <dbl>      <lgl>          <dttm>
##  1        David Bowie   musician    69       TRUE      1947-01-08
##  2      Carrie Fisher      actor    60       TRUE      1956-10-21
##  3        Chuck Berry   musician    90       TRUE      1926-10-18
##  4        Bill Paxton      actor    61       TRUE      1955-05-17
##  5             Prince   musician    57       TRUE      1958-06-07
##  6       Alan Rickman      actor    69      FALSE      1946-02-21
##  7 Florence Henderson      actor    82       TRUE      1934-02-14
##  8         Harper Lee     author    89      FALSE      1926-04-28
##  9      Zsa Zsa Gábor      actor    99       TRUE      1917-02-06
## 10     George Michael   musician    53      FALSE      1963-06-25
## # ... with 2 more variables: `Date of death` <dttm>, birthplace <chr>
```

# Data Exploration

## Gapminder Data

```
library(gapminder)

gapminder
```

```
## # A tibble: 1,704 x 6
##        country continent  year lifeExp      pop gdpPercap
##         <fctr>    <fctr> <int>   <dbl>    <int>     <dbl>
##  1 Afghanistan      Asia  1952  28.801  8425333  779.4453
##  2 Afghanistan      Asia  1957  30.332  9240934  820.8530
##  3 Afghanistan      Asia  1962  31.997 10267083  853.1007
##  4 Afghanistan      Asia  1967  34.020 11537966  836.1971
##  5 Afghanistan      Asia  1972  36.088 13079460  739.9811
##  6 Afghanistan      Asia  1977  38.438 14880372  786.1134
##  7 Afghanistan      Asia  1982  39.854 12881816  978.0114
##  8 Afghanistan      Asia  1987  40.822 13867957  852.3959
##  9 Afghanistan      Asia  1992  41.674 16317921  649.3414
## 10 Afghanistan      Asia  1997  41.763 22227415  635.3414
## # ... with 1,694 more rows
```

## Summary Statistics

### All countries

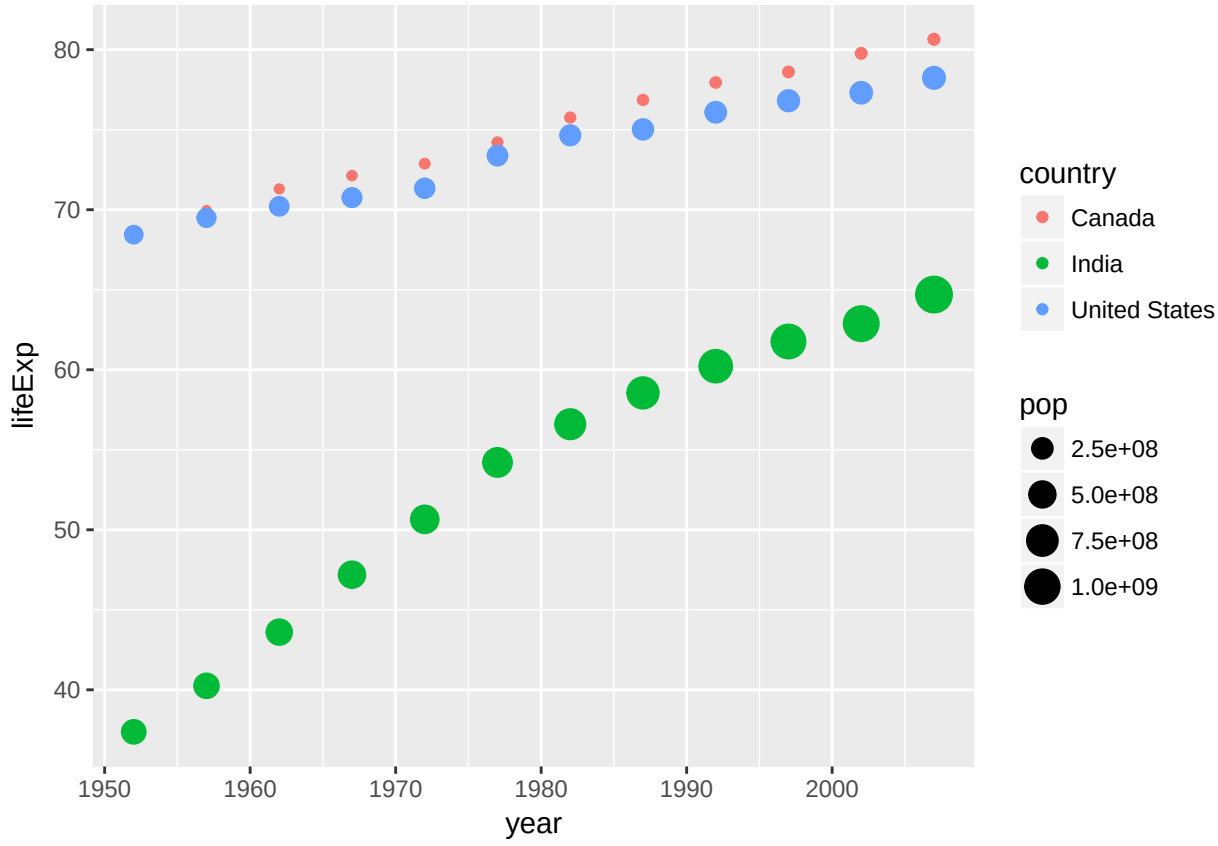
```
gapminder %>%
  group_by(country) %>%
  select(lifeExp, pop, gdpPercap) %>%
  summarize_all(funs(min, max, mean, sd, median))
```

```
## Adding missing grouping variables: `country`
```

```
## # A tibble: 142 x 16
##        country lifeExp_min  pop_min gdpPercap_min lifeExp_max   pop_max
##         <fctr>       <dbl>    <dbl>         <dbl>       <dbl>     <dbl>
##  1 Afghanistan      28.801  8425333      635.3414      43.828  31889923
##  2     Albania      55.230  1282697     1601.0561      76.423   3600523
##  3     Algeria      43.077  9279525     2449.0082      72.301  33333216
##  4      Angola      30.015  4232095     2277.1409      42.731  12420476
##  5   Argentina      62.485 17876956     5911.3151      75.320  40301927
##  6   Australia      69.120  8691212    10039.5956      81.235  20434176
##  7     Austria      66.800  6927772     6137.0765      79.829   8199783
##  8     Bahrain      50.939   120447     9867.0848      75.635    708573
##  9  Bangladesh      37.484 46886859      630.2336      64.062 150448339
## 10     Belgium      68.000  8730405     8343.1051      79.441  10392226
## # ... with 132 more rows, and 10 more variables: gdpPercap_max <dbl>,
## #   lifeExp_mean <dbl>, pop_mean <dbl>, gdpPercap_mean <dbl>,
## #   lifeExp_sd <dbl>, pop_sd <dbl>, gdpPercap_sd <dbl>,
## #   lifeExp_median <dbl>, pop_median <dbl>, gdpPercap_median <dbl>
```

### Plot (Interactive and non-interactive)

```
library(ggplot2)
gapminder %>%
  filter(country %in% c("Canada", "United States", "India")) %>%
  ggplot(data = .) +
  geom_point(aes(x = year, y = lifeExp, size = pop, color = country))
```




```
# gapminder %>%
#   filter(country %in% c("Canada", "United States", "India")) %>%
#   ggplot(data = .) +
#   geom_point(aes(x = year, y = lifeExp, size = pop)) +
#   facet_wrap(~country)
```

```
library(plotly)
```

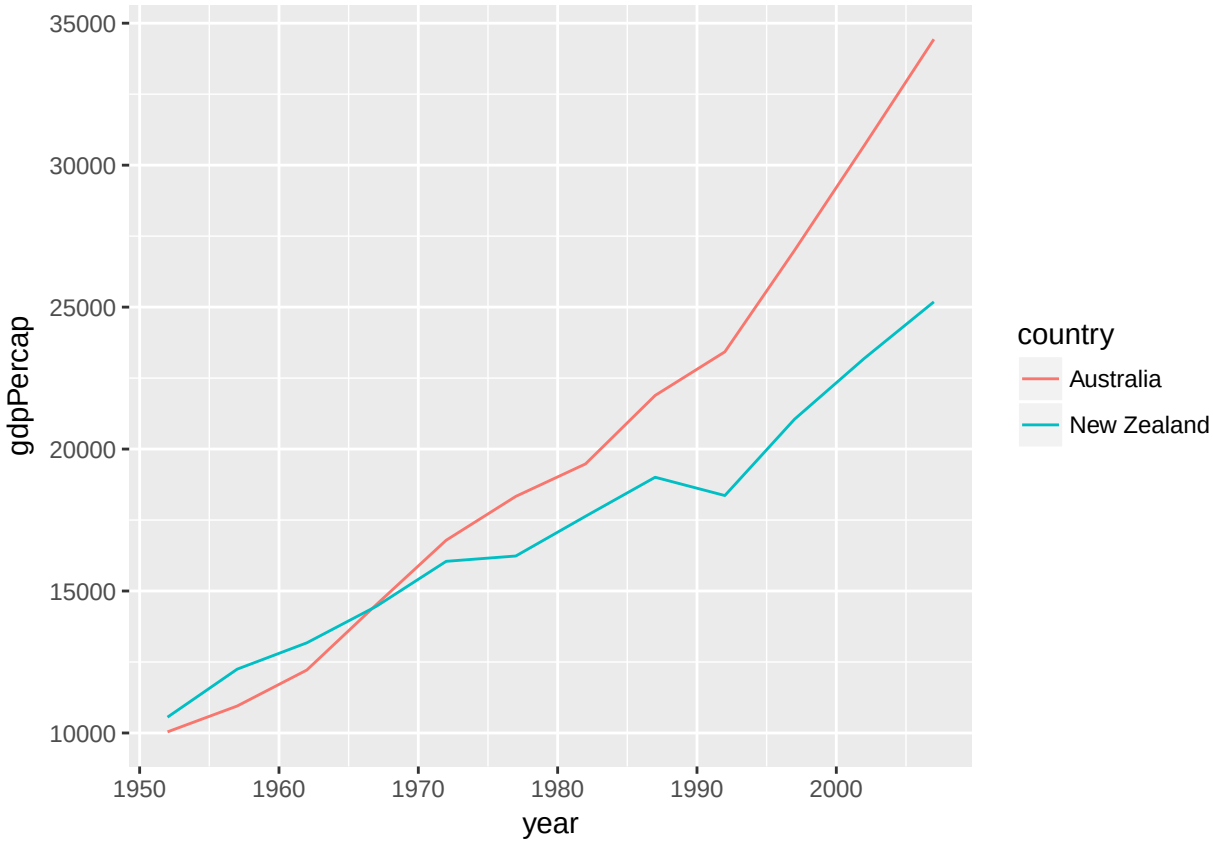
```
##
## Attaching package: 'plotly'
```

```
## The following object is masked from 'package:ggplot2':
##
##     last_plot
```

```
## The following object is masked from 'package:stats':
##
##     filter
```

```
## The following object is masked from 'package:graphics':
##
##     layout
```

```
  ggplot(data = gapminder %>%
  filter(continent=="Oceania")) + #Europe Africa
  geom_line(aes(x = year, y = gdpPercap, color = country))
```




```
  ggplotly()
```

```
## We recommend that you use the dev version of ggplot2 with `ggplotly()`
## Install it with: `devtools::install_github('hadley/ggplot2')`
```

# Modelling

## Gapminder data (Nested)

```
library(tidyr)
library(purrr)
```

```
##
## Attaching package: 'purrr'
```

```
## The following objects are masked from 'package:dplyr':
##
##     contains, order_by
```

```
by_country <- gapminder %>%
  group_by(country, continent) %>%
  nest()
by_country
```

```
## # A tibble: 142 x 3
##        country continent              data
```

```
##         <fctr>    <fctr>            <list>
##  1 Afghanistan      Asia <tibble [12 x 4]>
##  2     Albania    Europe <tibble [12 x 4]>
##  3     Algeria    Africa <tibble [12 x 4]>
##  4      Angola    Africa <tibble [12 x 4]>
##  5   Argentina  Americas <tibble [12 x 4]>
##  6   Australia   Oceania <tibble [12 x 4]>
##  7     Austria    Europe <tibble [12 x 4]>
##  8     Bahrain      Asia <tibble [12 x 4]>
##  9  Bangladesh      Asia <tibble [12 x 4]>
## 10     Belgium    Europe <tibble [12 x 4]>
## # ... with 132 more rows
```

### Fitting Models:

```
#Model
country_model <- function(df) {
  lm(lifeExp ~ year, data = df)
}

# Fitting model
by_country <- by_country %>%
  mutate(model = map(data, country_model))
by_country
```

```
## # A tibble: 142 x 4
##         country continent              data    model
##          <fctr>    <fctr>            <list>   <list>
##  1 Afghanistan      Asia <tibble [12 x 4]> <S3: lm>
##  2     Albania    Europe <tibble [12 x 4]> <S3: lm>
##  3     Algeria    Africa <tibble [12 x 4]> <S3: lm>
##  4      Angola    Africa <tibble [12 x 4]> <S3: lm>
##  5   Argentina  Americas <tibble [12 x 4]> <S3: lm>
##  6   Australia   Oceania <tibble [12 x 4]> <S3: lm>
##  7     Austria    Europe <tibble [12 x 4]> <S3: lm>
##  8     Bahrain      Asia <tibble [12 x 4]> <S3: lm>
##  9  Bangladesh      Asia <tibble [12 x 4]> <S3: lm>
## 10     Belgium    Europe <tibble [12 x 4]> <S3: lm>
## # ... with 132 more rows
```

## Getting Goodness of fit and Significance

```
library(broom)
by_country %>%
  mutate(glance = map(model, glance)) %>%
  unnest(glance, .drop = TRUE)
```

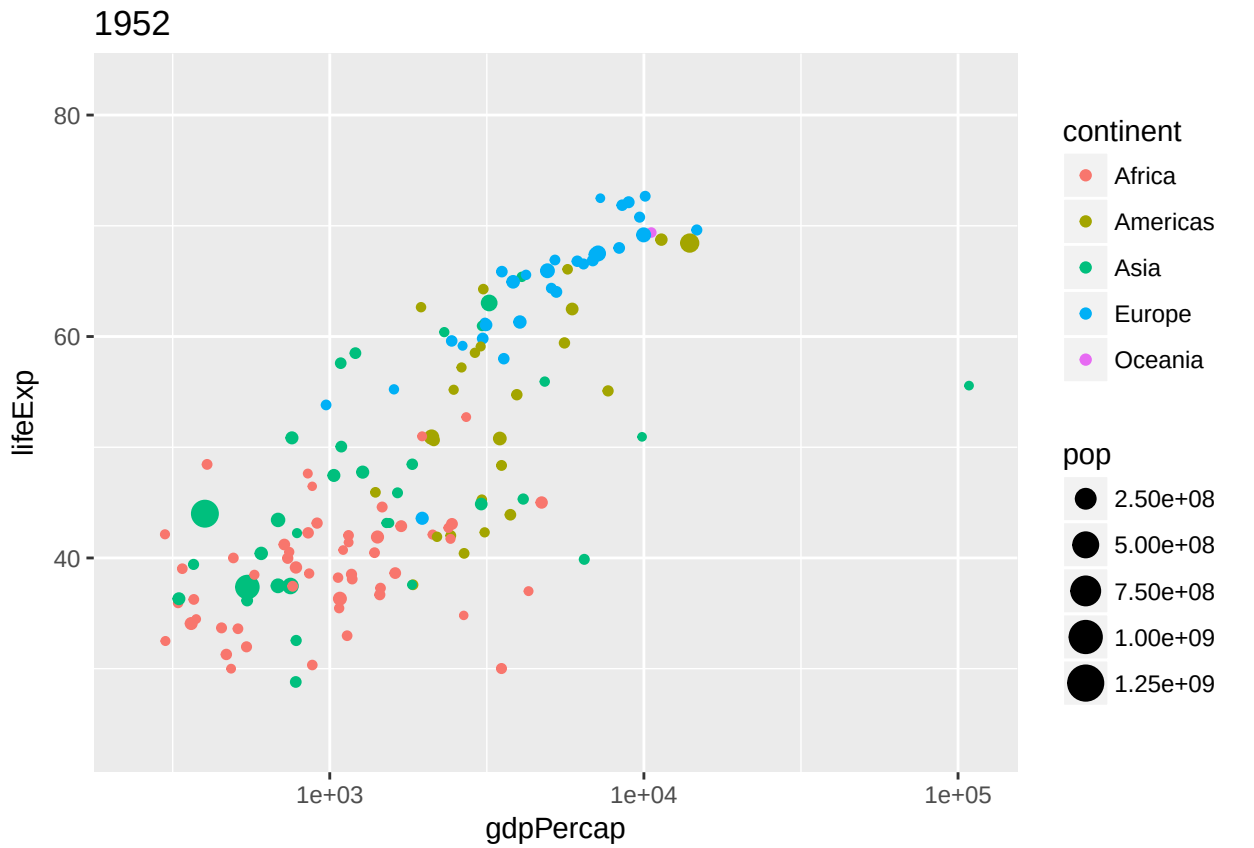
```
## # A tibble: 142 x 13
##         country continent r.squared adj.r.squared     sigma  statistic
##          <fctr>    <fctr>     <dbl>         <dbl>     <dbl>      <dbl>
##  1 Afghanistan      Asia 0.9477123     0.9424835 1.2227880  181.24941
```

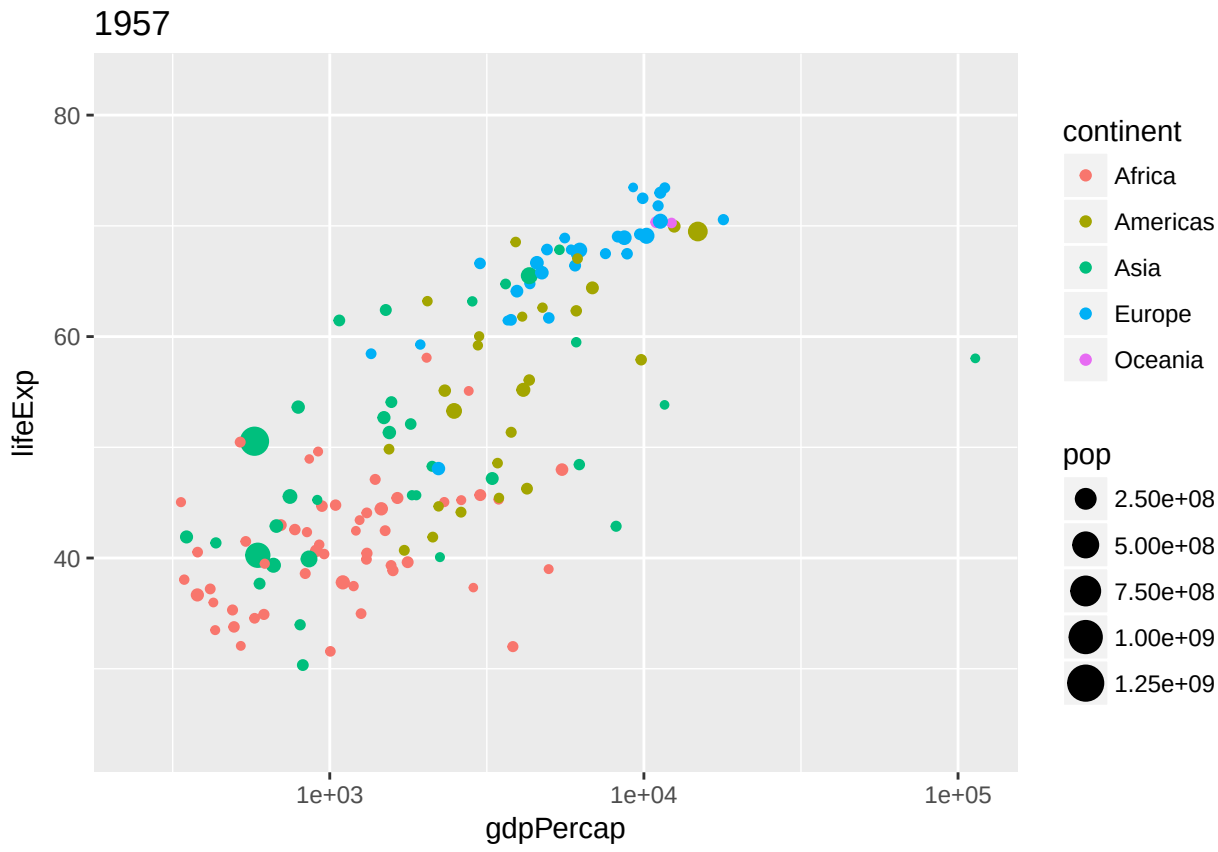
```
##  2     Albania    Europe 0.9105778     0.9016355 1.9830615  101.82901
##  3     Algeria    Africa 0.9851172     0.9836289 1.3230064  661.91709
##  4      Angola    Africa 0.8878146     0.8765961 1.4070091   79.13818
##  5   Argentina  Americas 0.9955681     0.9951249 0.2923072 2246.36635
##  6   Australia   Oceania 0.9796477     0.9776125 0.6206086  481.34586
##  7     Austria    Europe 0.9921340     0.9913474 0.4074094 1261.29629
##  8     Bahrain      Asia 0.9667398     0.9634138 1.6395865  290.65974
##  9  Bangladesh      Asia 0.9893609     0.9882970 0.9766908  929.92637
## 10     Belgium    Europe 0.9945406     0.9939946 0.2929025 1821.68840
## # ... with 132 more rows, and 7 more variables: p.value <dbl>, df <int>,
## #   logLik <dbl>, AIC <dbl>, BIC <dbl>, deviance <dbl>, df.residual <int>
```

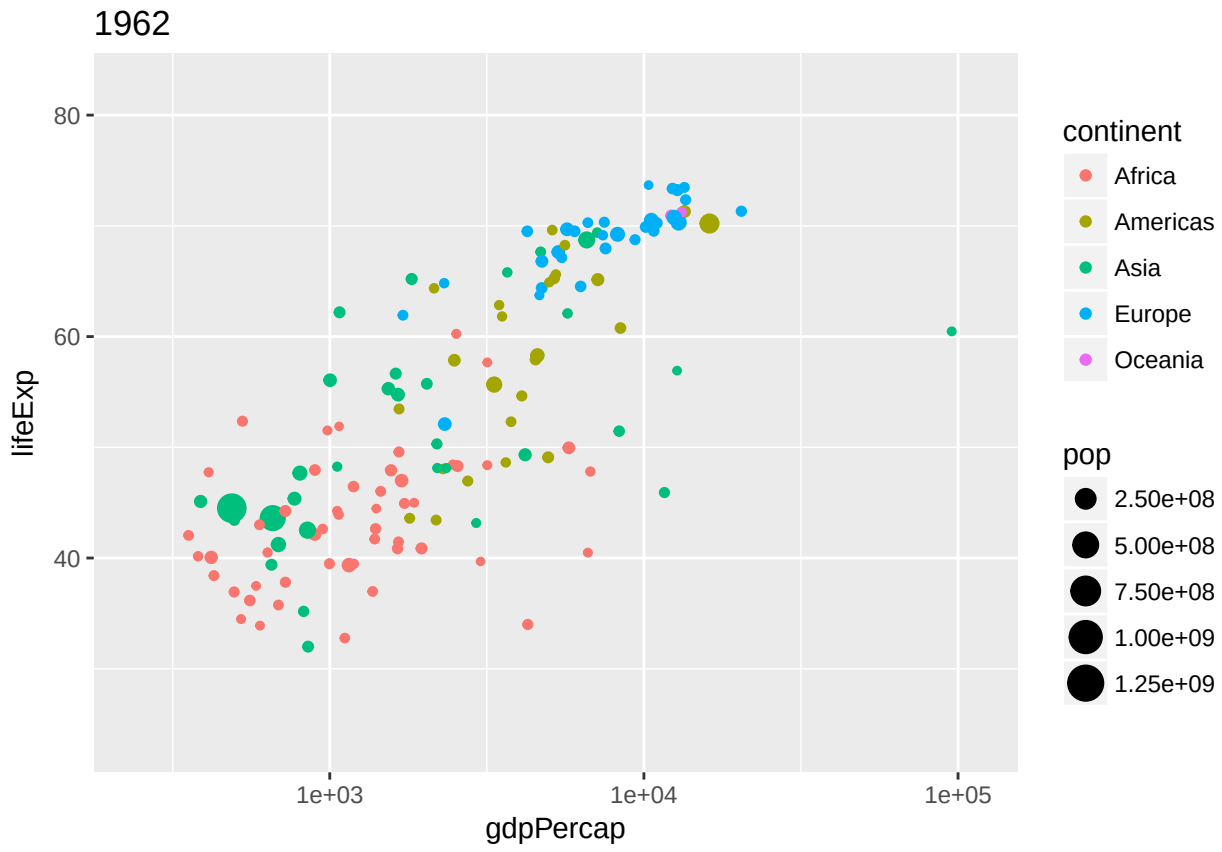
# Communication

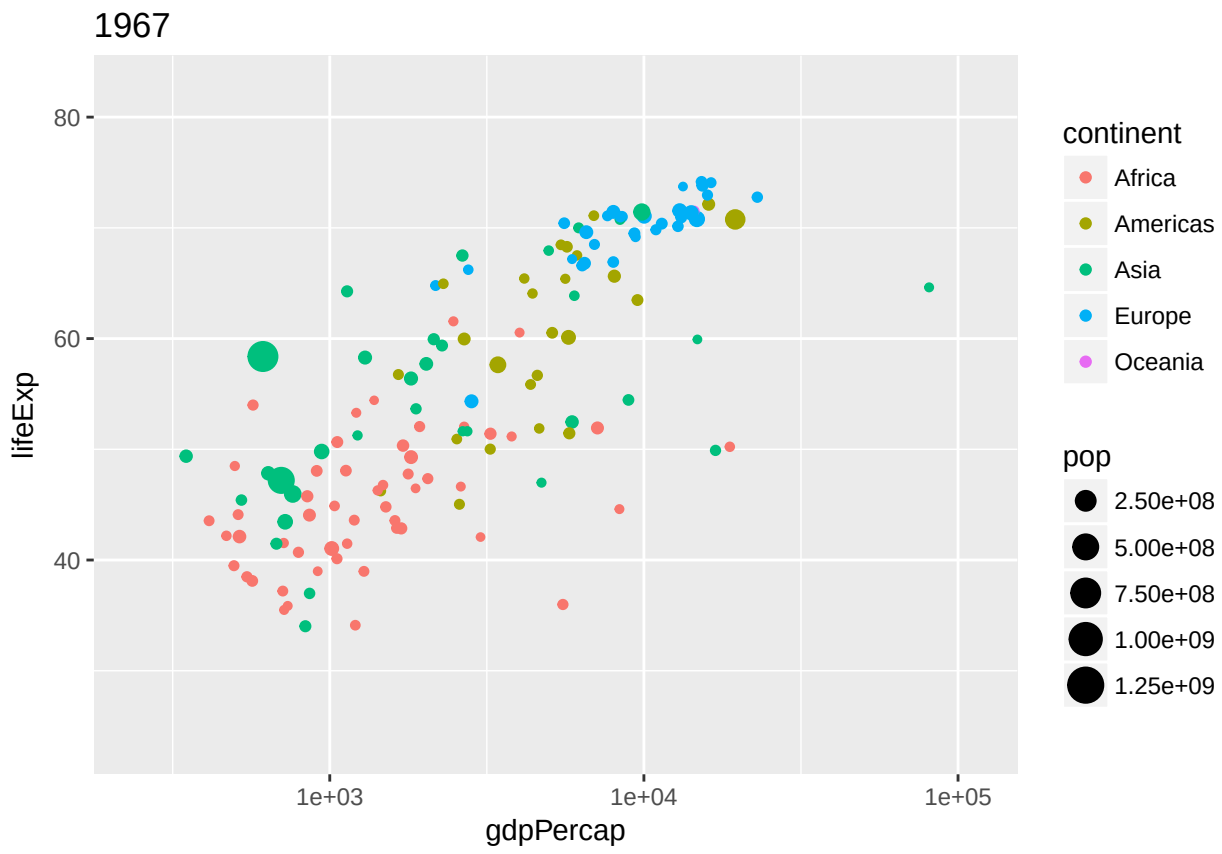
## Gapminder animation

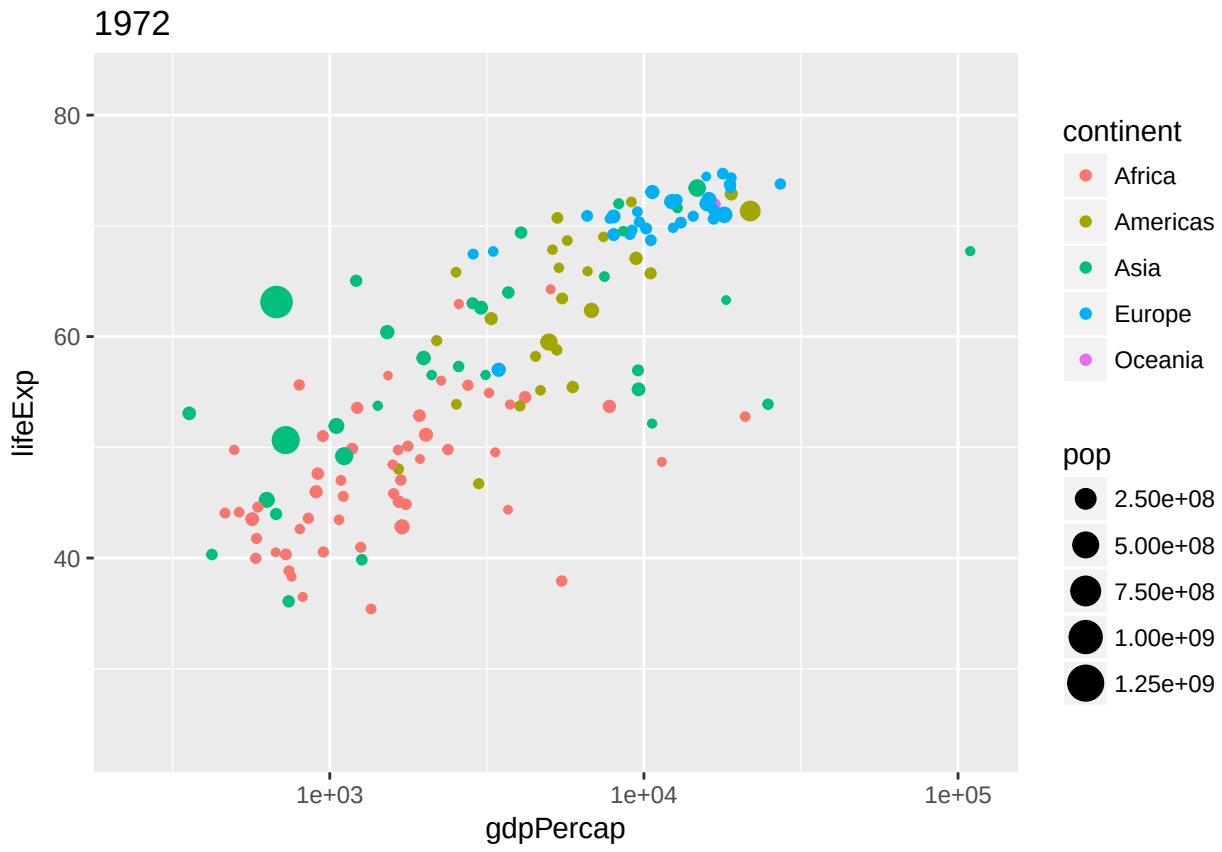
```
library(gganimate)

p <- ggplot(gapminder, aes(gdpPercap, lifeExp, size = pop, color = continent, frame = year)) +
  geom_point() +
  scale_x_log10()
gganimate(p)
```

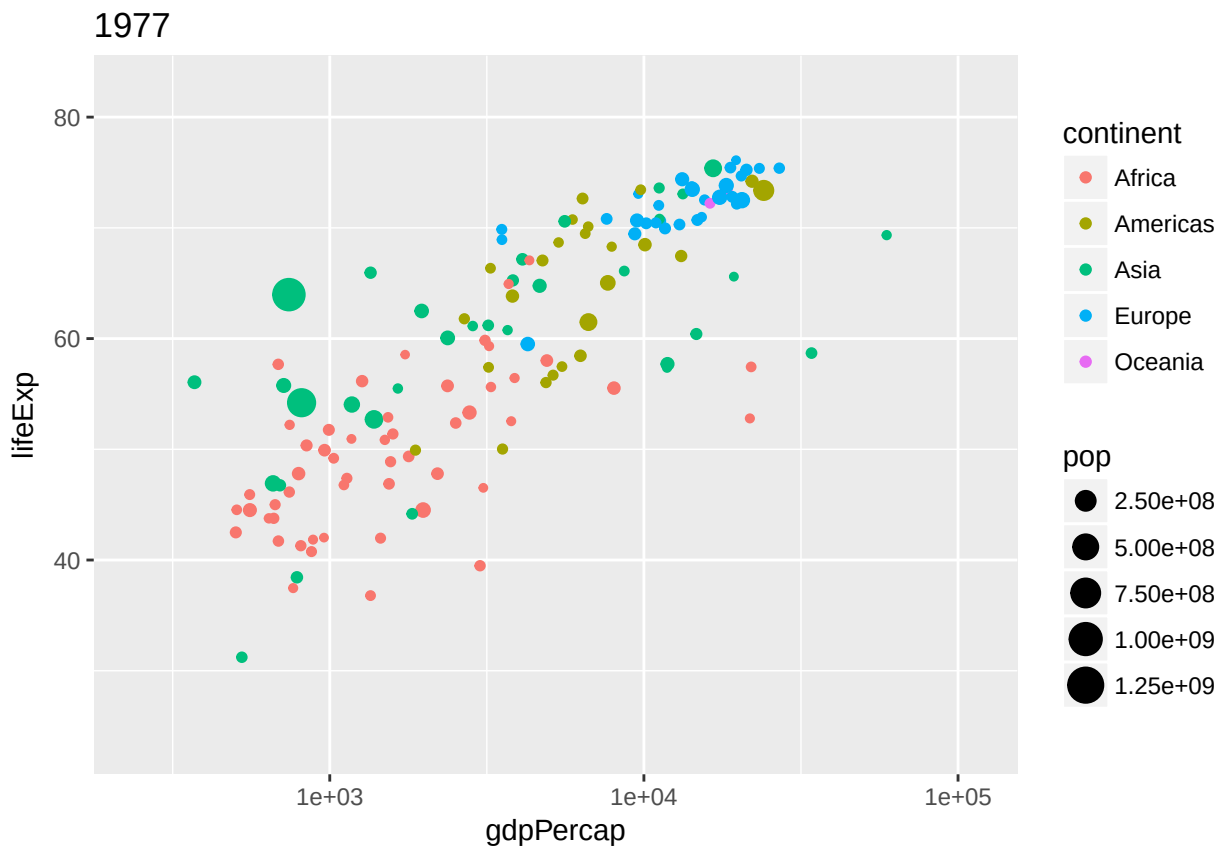


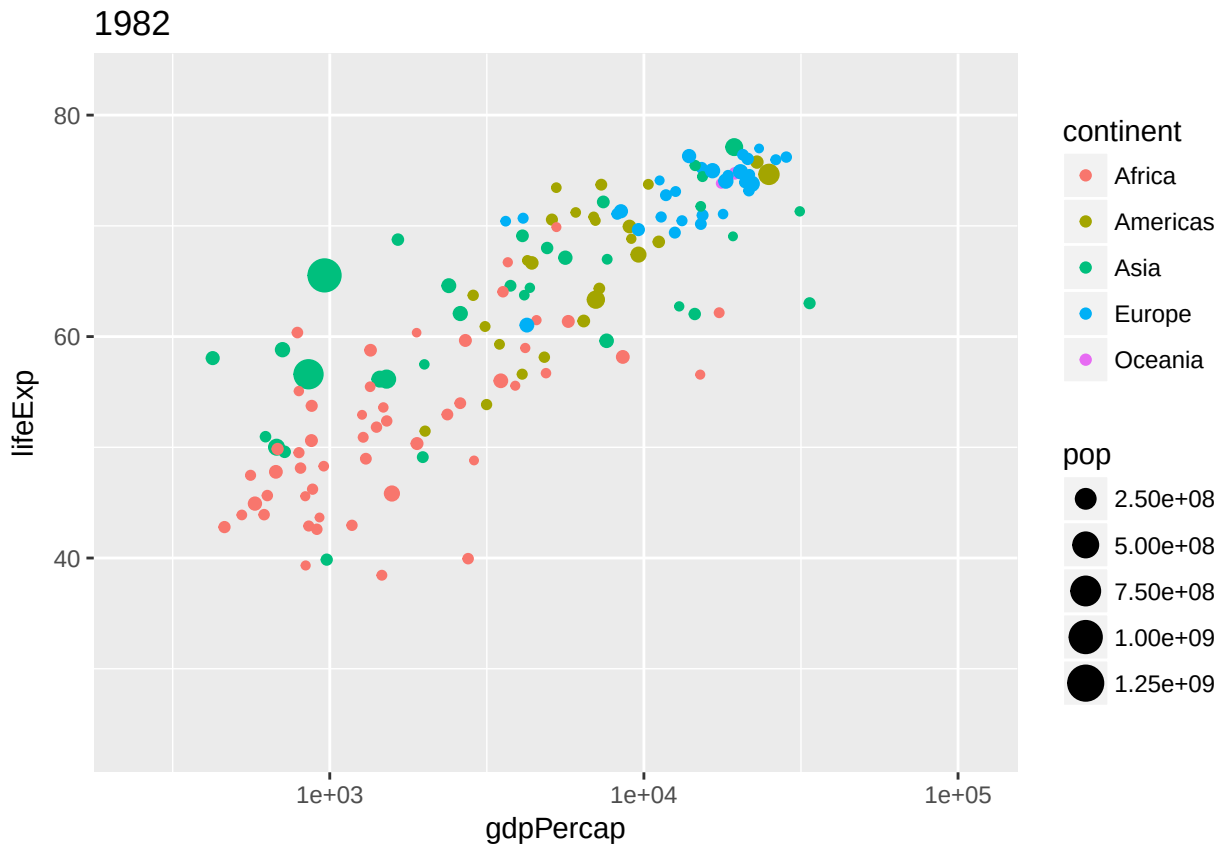

1957
80
60
40
lifeExp
1e+03
1e+04
1e+05
gdpPercap
continent
Africa
Americas
Asia
Europe
Oceania
pop
2.50e+08
5.00e+08
7.50e+08
1.00e+09
1.25e+09

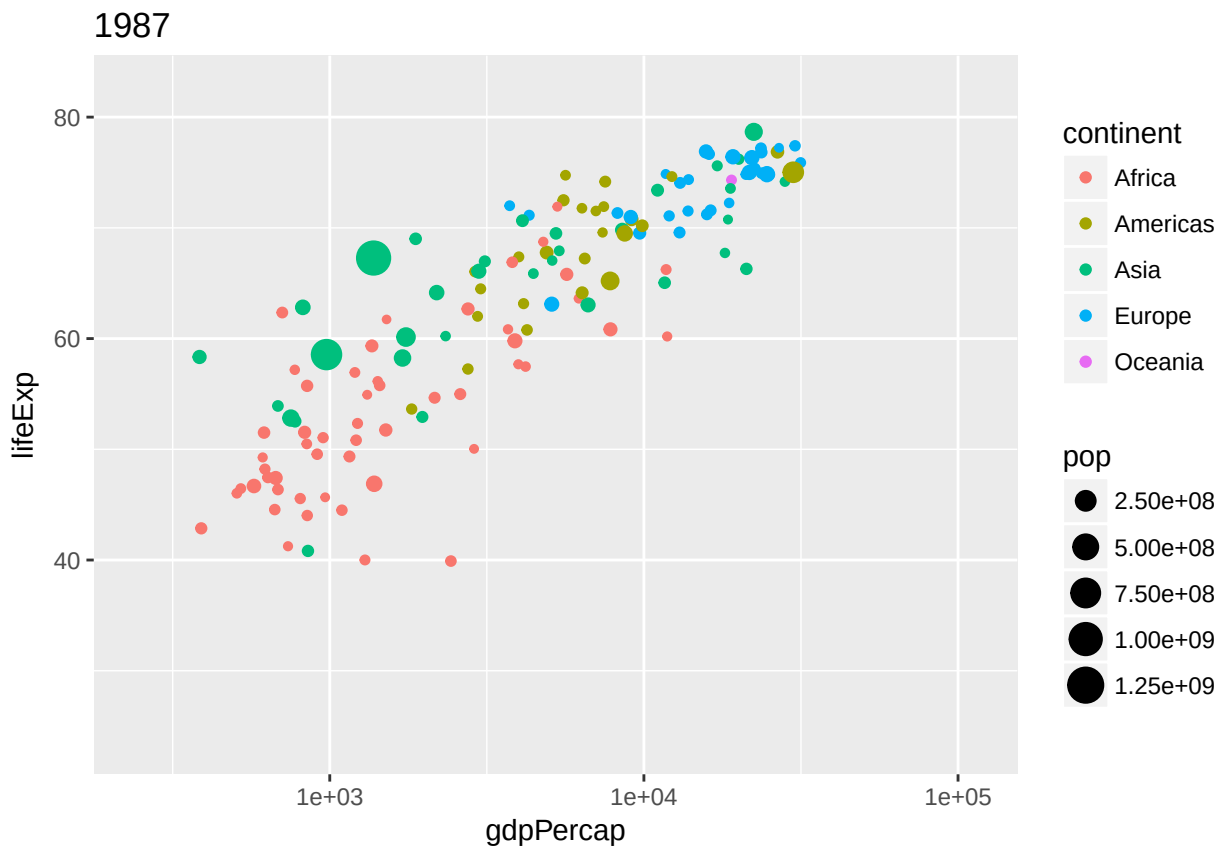

1962
80
60
40
lifeExp
1e+03
1e+04
1e+05
gdpPercap
continent
Africa
Americas
Asia
Europe
Oceania
pop
2.50e+08
5.00e+08
7.50e+08
1.00e+09
1.25e+09

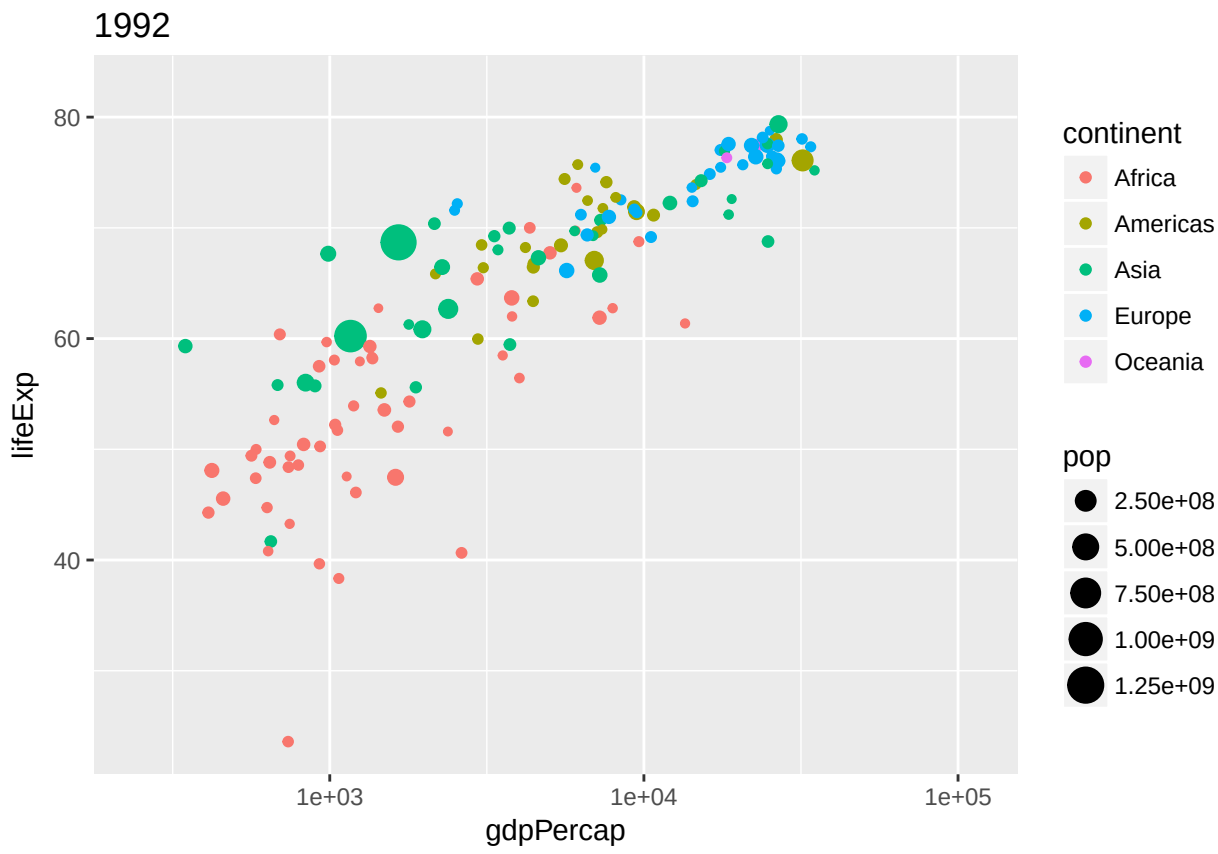

1967
80
60
40
lifeExp
1e+03
1e+04
1e+05
gdpPercap
continent
Africa
Americas
Asia
Europe
Oceania
pop
2.50e+08
5.00e+08
7.50e+08
1.00e+09
1.25e+09

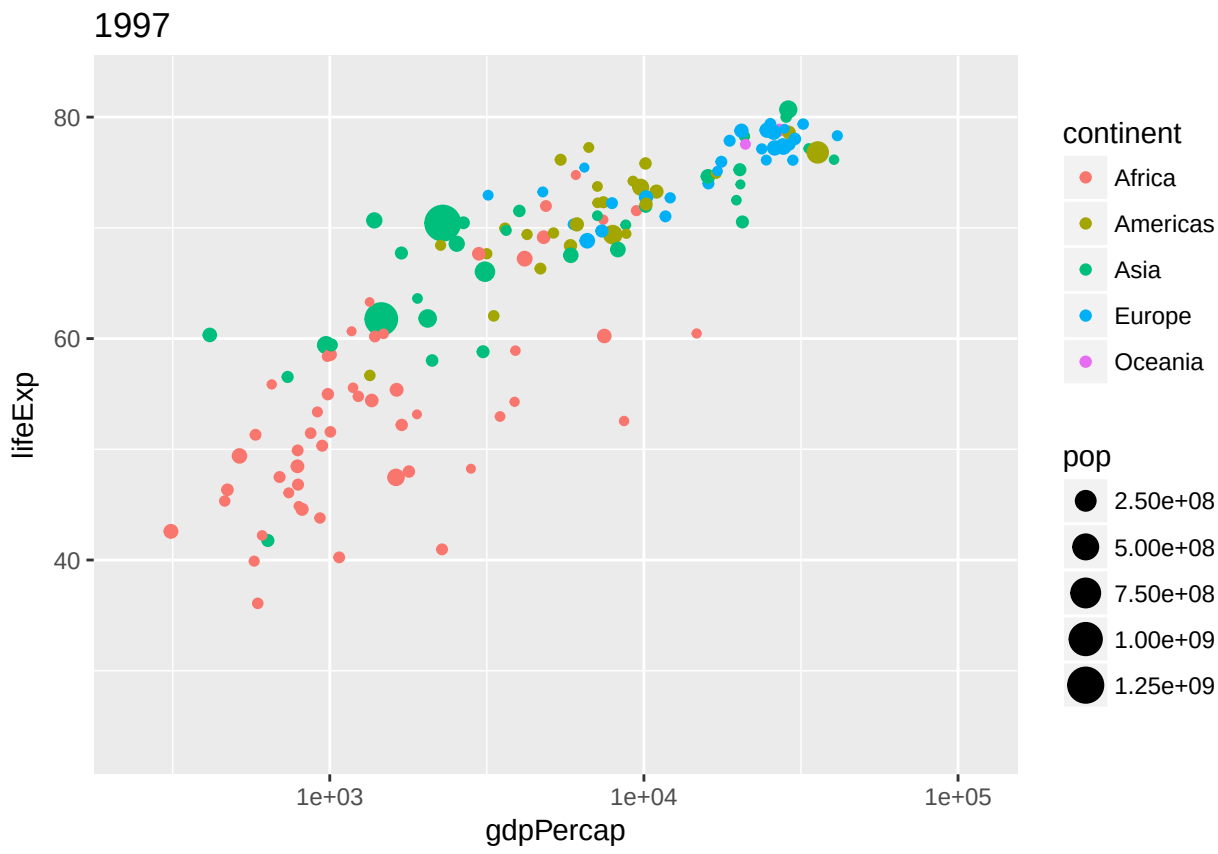

1972
80
60
40
lifeExp
1e+03
1e+04
1e+05
gdpPercap
continent
Africa
Americas
Asia
Europe
Oceania
pop
2.50e+08
5.00e+08
7.50e+08
1.00e+09
1.25e+09

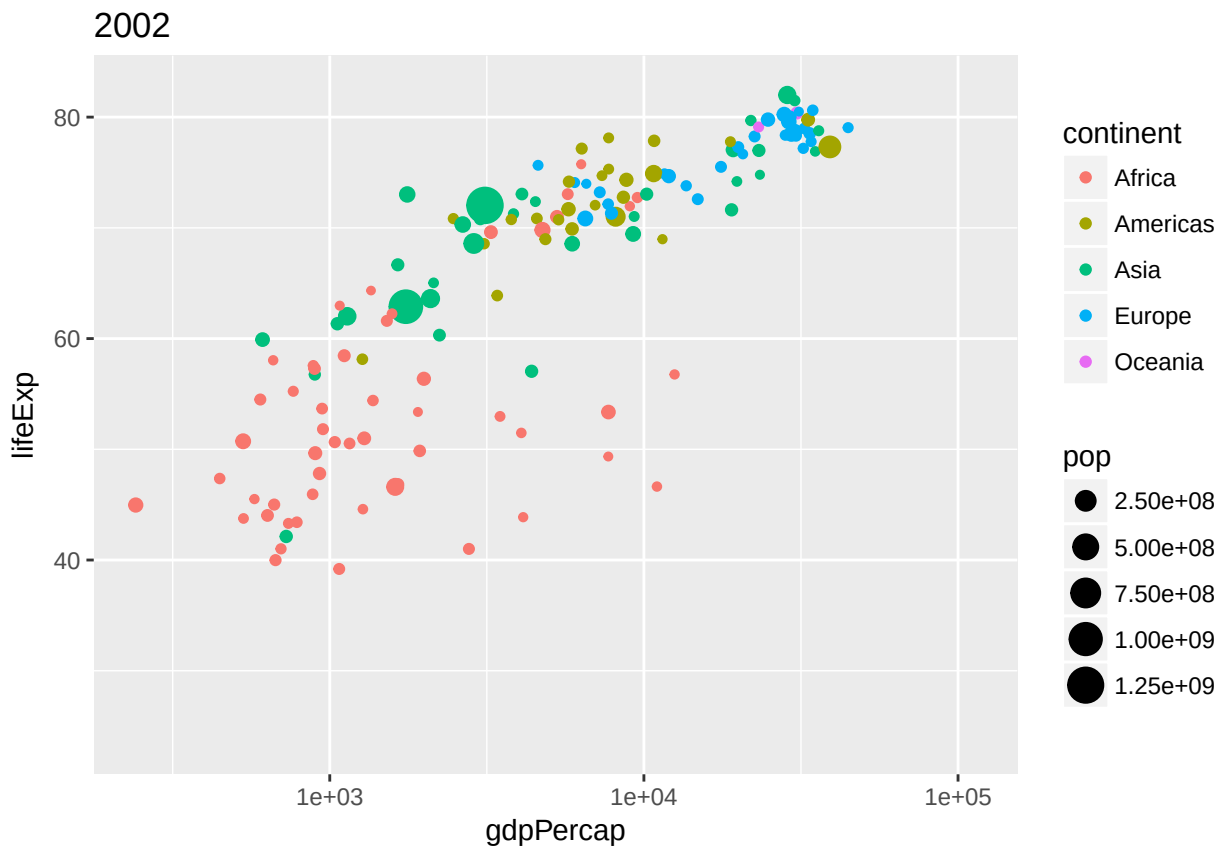

1977
80
60
40
lifeExp
1e+03
1e+04
1e+05
gdpPercap
continent
Africa
Americas
Asia
Europe
Oceania
pop
2.50e+08
5.00e+08
7.50e+08
1.00e+09
1.25e+09

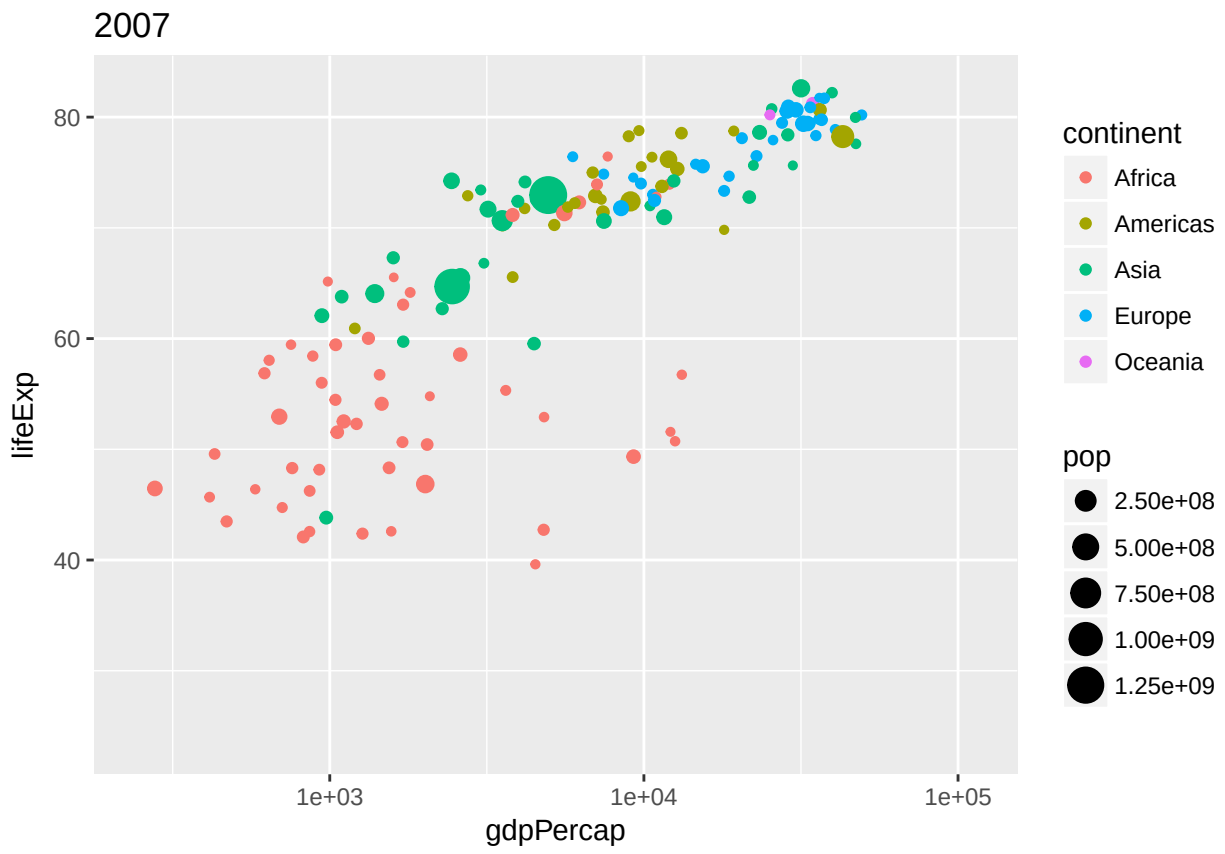

1982
80
60
40
lifeExp
1e+03
1e+04
1e+05
gdpPercap
continent
Africa
Americas
Asia
Europe
Oceania
pop
2.50e+08
5.00e+08
7.50e+08
1.00e+09
1.25e+09


1987
80
60
40
lifeExp
1e+03
1e+04
1e+05
gdpPercap
continent
Africa
Americas
Asia
Europe
Oceania
pop
2.50e+08
5.00e+08
7.50e+08
1.00e+09
1.25e+09


1992
80
60
40
lifeExp
1e+03
1e+04
1e+05
gdpPercap
continent
Africa
Americas
Asia
Europe
Oceania
pop
2.50e+08
5.00e+08
7.50e+08
1.00e+09
1.25e+09


1997
80
60
40
lifeExp
1e+03
1e+04
1e+05
gdpPercap
continent
Africa
Americas
Asia
Europe
Oceania
pop
2.50e+08
5.00e+08
7.50e+08
1.00e+09
1.25e+09


2002
80
60
40
lifeExp
1e+03
1e+04
1e+05
gdpPercap
continent
Africa
Americas
Asia
Europe
Oceania
pop
2.50e+08
5.00e+08
7.50e+08
1.00e+09
1.25e+09




```
sessionInfo()
```

```
## R version 3.4.1 (2017-06-30)
## Platform: x86_64-w64-mingw32/x64 (64-bit)
## Running under: Windows 8.1 x64 (build 9600)
##
## Matrix products: default
##
## locale:
## [1] LC_COLLATE=English_Canada.1252  LC_CTYPE=English_Canada.1252
## [3] LC_MONETARY=English_Canada.1252 LC_NUMERIC=C
## [5] LC_TIME=English_Canada.1252
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
##  [1] gganimate_0.1.0.9000 broom_0.4.2          purrr_0.2.2.2
##  [4] tidyr_0.6.3          plotly_4.7.0         ggplot2_2.2.1
##  [7] gapminder_0.2.0      bindrcpp_0.2         dplyr_0.7.2
## [10] readxl_1.0.0
##
## loaded via a namespace (and not attached):
##  [1] Rcpp_0.12.12      cellranger_1.1.0  compiler_3.4.1
##  [4] plyr_1.8.4        bindr_0.1         tools_3.4.1
##  [7] digest_0.6.12     lattice_0.20-35   nlme_3.1-131
```

```
## [10] jsonlite_1.5      evaluate_0.10.1   tibble_1.3.3
## [13] gtable_0.2.0      viridisLite_0.2.0 pkgconfig_2.0.1
## [16] rlang_0.1.1       psych_1.7.5       shiny_1.0.3
## [19] crosstalk_1.0.0   parallel_3.4.1    yaml_2.1.14
## [22] stringr_1.2.0     httr_1.2.1        knitr_1.16
## [25] htmlwidgets_0.9   rprojroot_1.2     grid_3.4.1
## [28] glue_1.1.1        data.table_1.10.4 R6_2.2.2
## [31] foreign_0.8-69    rmarkdown_1.6     reshape2_1.4.2
## [34] magrittr_1.5      backports_1.1.0   scales_0.4.1
## [37] htmltools_0.3.6   mnormt_1.5-5      assertthat_0.2.0
## [40] xtable_1.8-2      mime_0.5          colorspace_1.3-2
## [43] httpuv_1.3.5      labeling_0.3      stringi_1.1.5
## [46] lazyeval_0.2.0    munsell_0.4.3
```