# Lecture 9: Finalizing graphs

Michael Lopez, Skidmore College

# Data viz's in the news

http://www.spiegel.de/wissenschaft/natur/bild-1119424-1067142.html

http://www.spiegel.de/wissenschaft/natur/bild-1119539-1067290.html

# Finishing graphs

Today's goal: building a final plot

1. Axis transformation
2. Labeling
3. Annotation & other geometric shapes
4. Animation & Interactivity
5. Saving

# Gapminder data

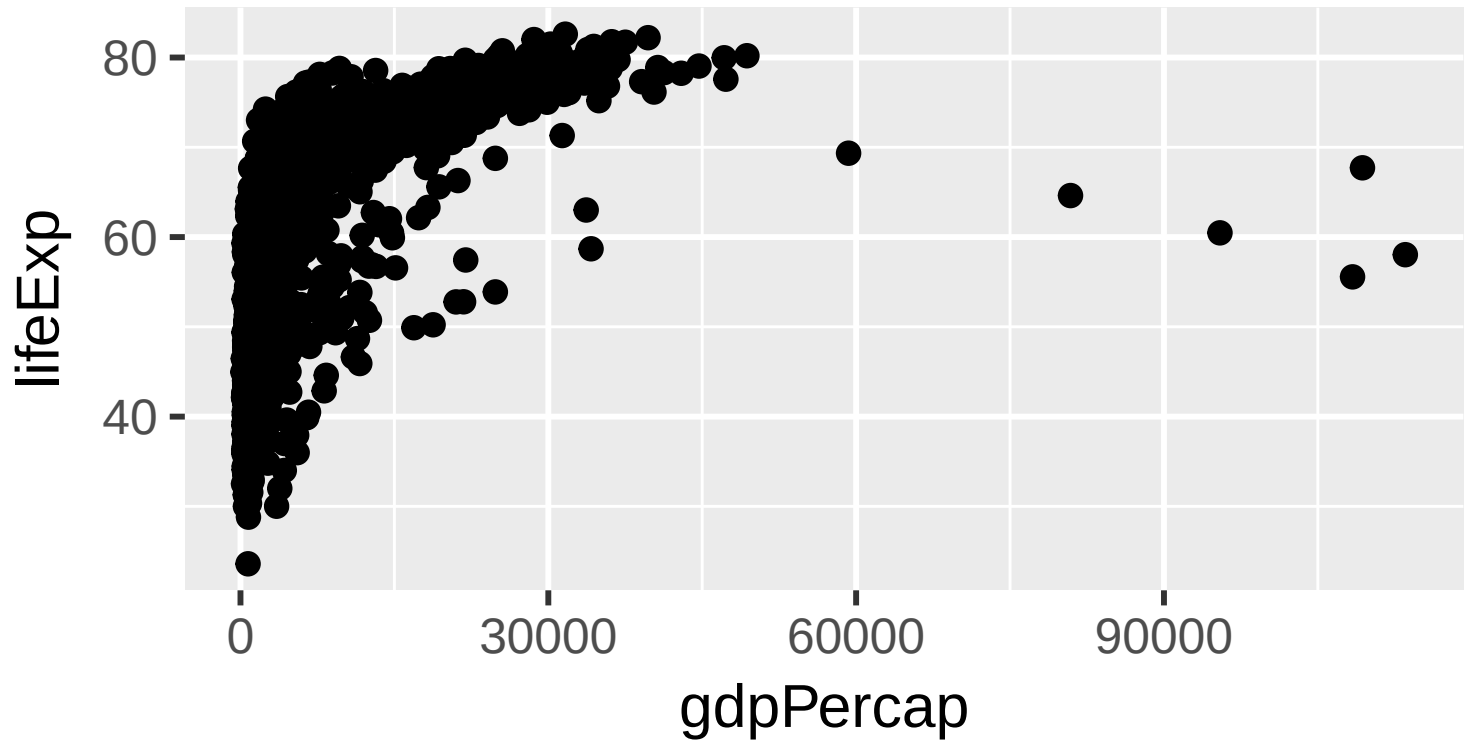
```
library(gapminder); library(dplyr); library(ggplot2)
gapminder %>%
  head(3)
```

```
## # A tibble: 3 × 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
```

Ultimate goal: How does life expectency vary based on gdp per capita?

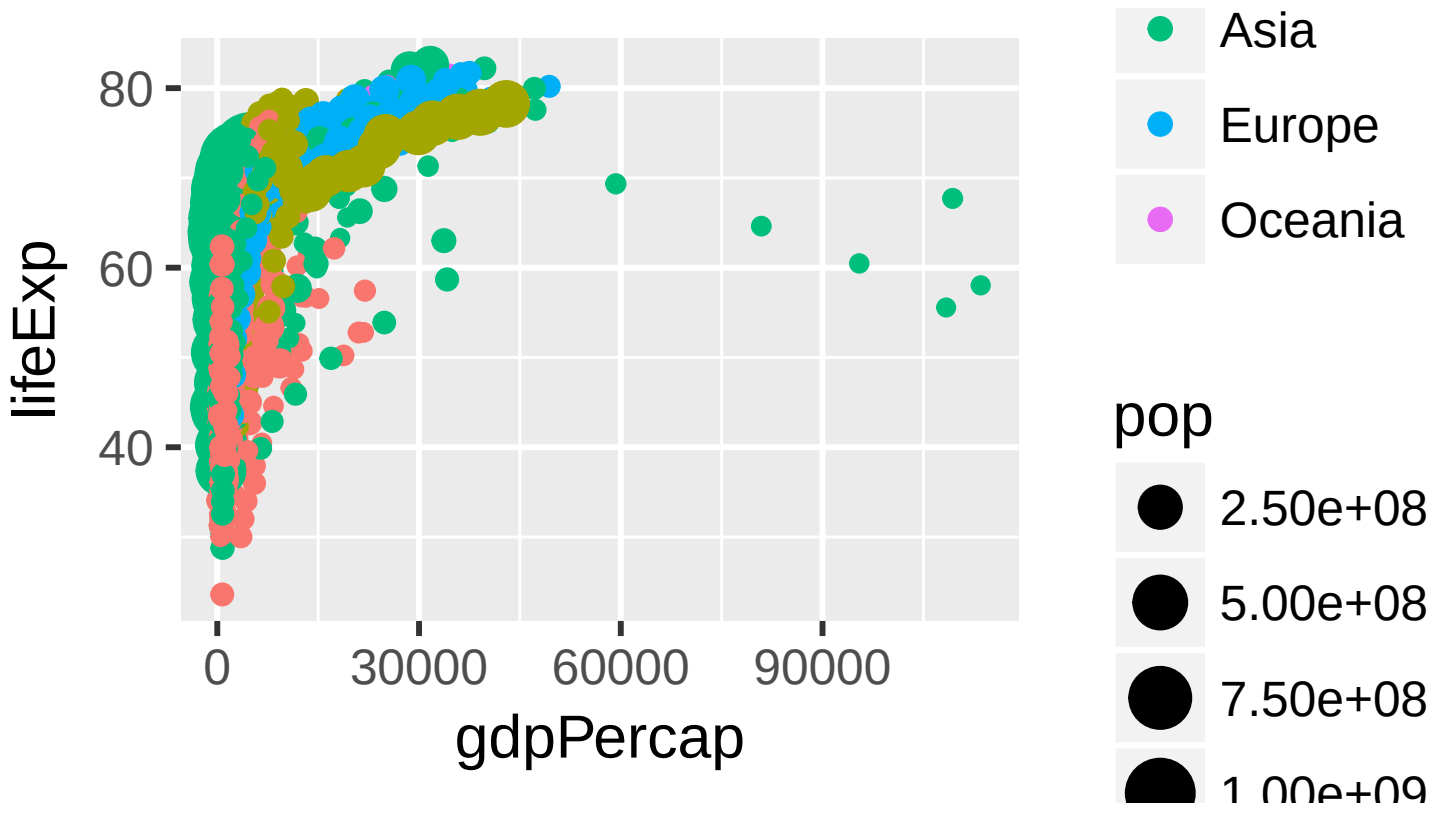
# Gapminder data: initial look

```
p <- ggplot(gapminder, aes(gdpPercap, lifeExp)) +
  geom_point()
p
```



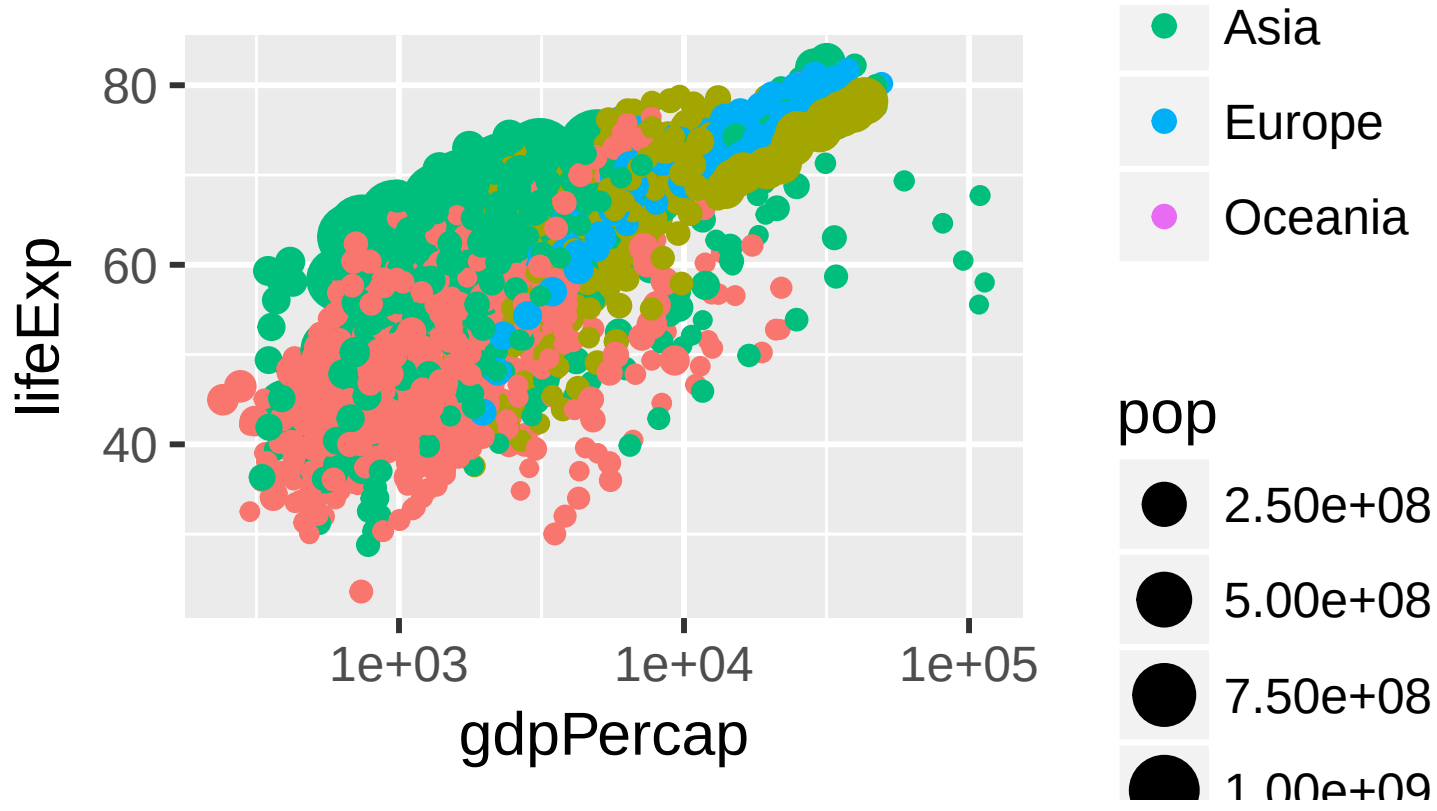
# Gapminder data: grouping

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



# Gapminder data: changing the axis

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



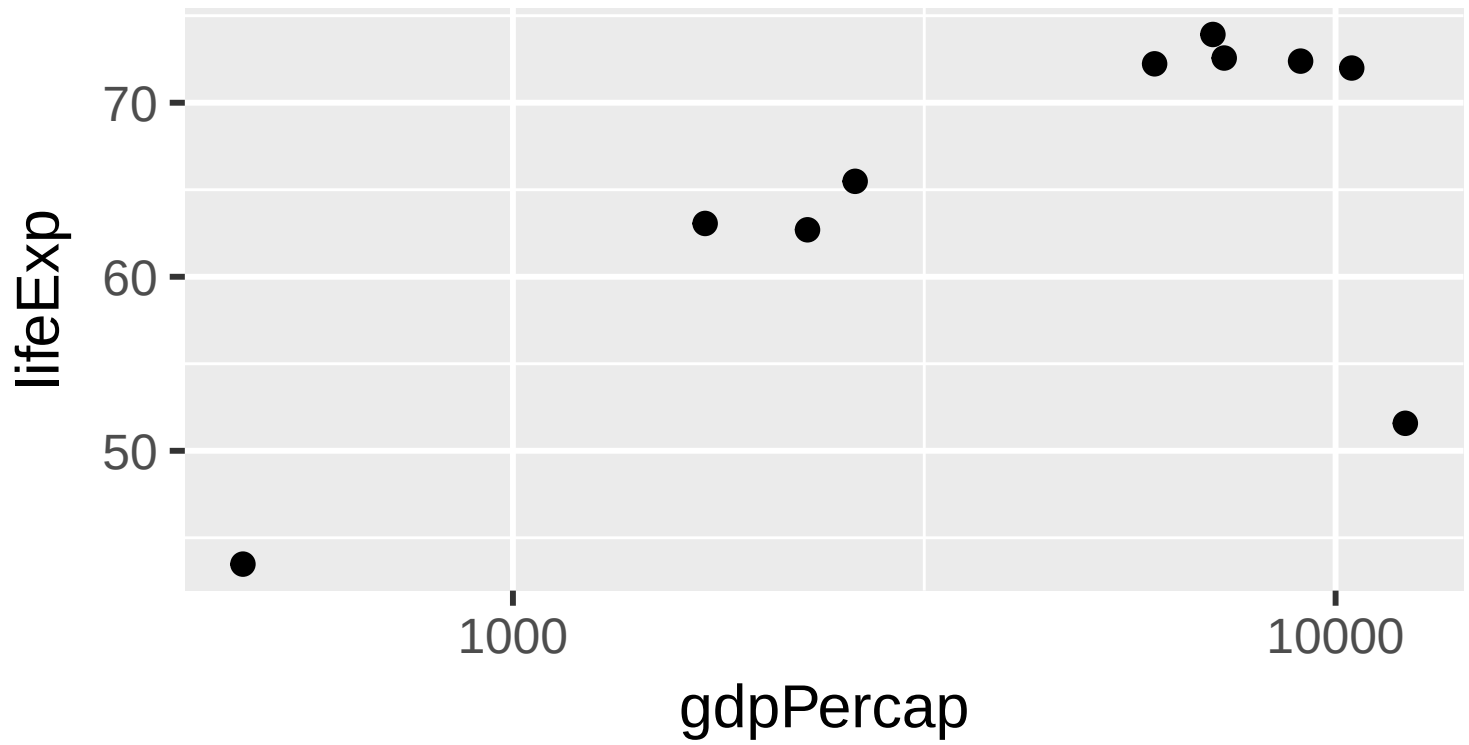
# Tranformed scale

```
gapminder1 <- gapminder %>%
   mutate(gdp.log10 = log(gdpPercap, 10))
set.seed(5)
gapminder1 %>% arrange(gdpPercap) %>% sample_n(2)
```

```
## # A tibble: 2 × 7
##   country continent  year lifeExp       pop gdpPercap gdp.log10
##    <fctr>    <fctr> <int>   <dbl>     <int>     <dbl>     <dbl>
## 1   India      Asia  1987  58.553 788000000  976.5127  2.989678
## 2 Jamaica  Americas  1997  72.262   2531311 7121.9247  3.852597
```

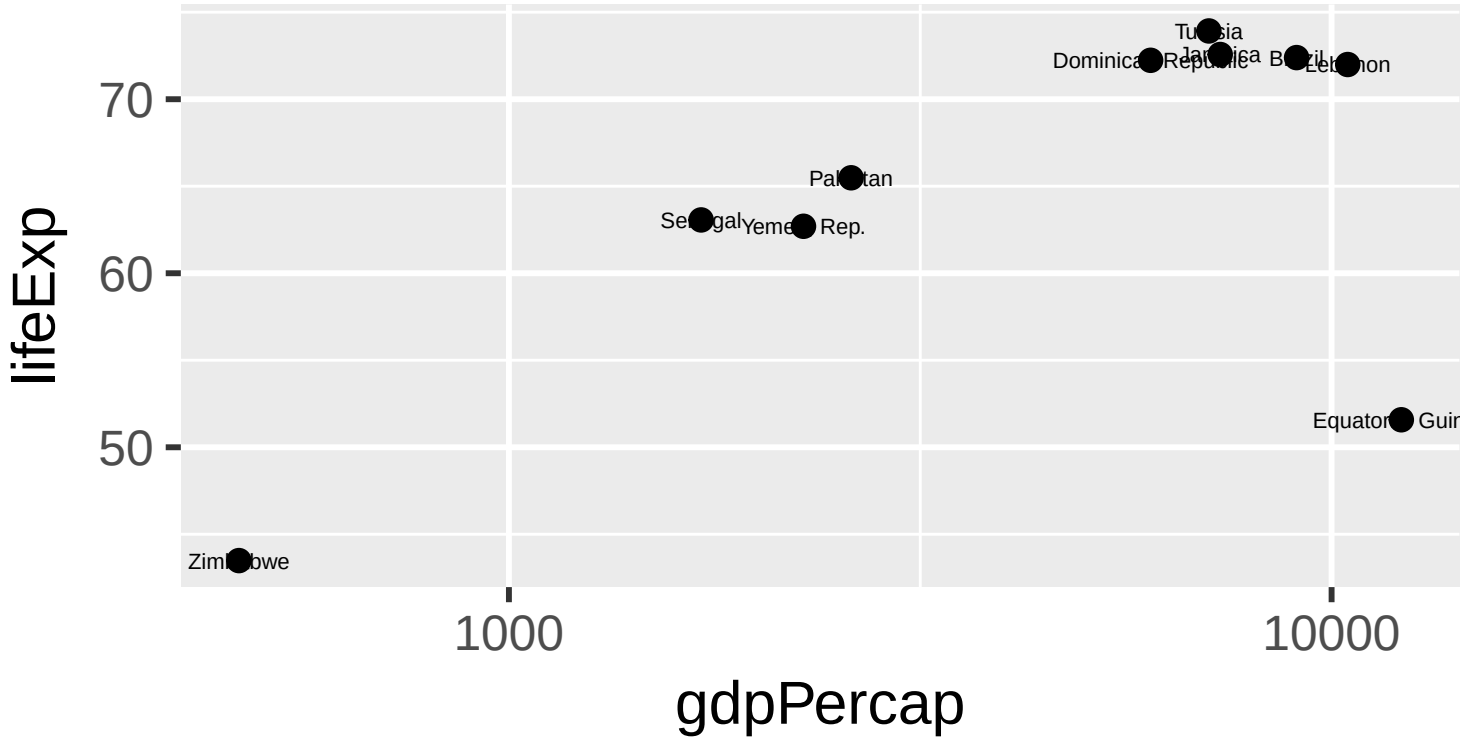
# How to improve?

```
gapminder2 <- gapminder1 %>%
     filter(year == 2007) %>% sample_n(10)
p <- ggplot(gapminder2, aes(gdpPercap, lifeExp)) +
  geom_point() +
  scale_x_log10()
p
```



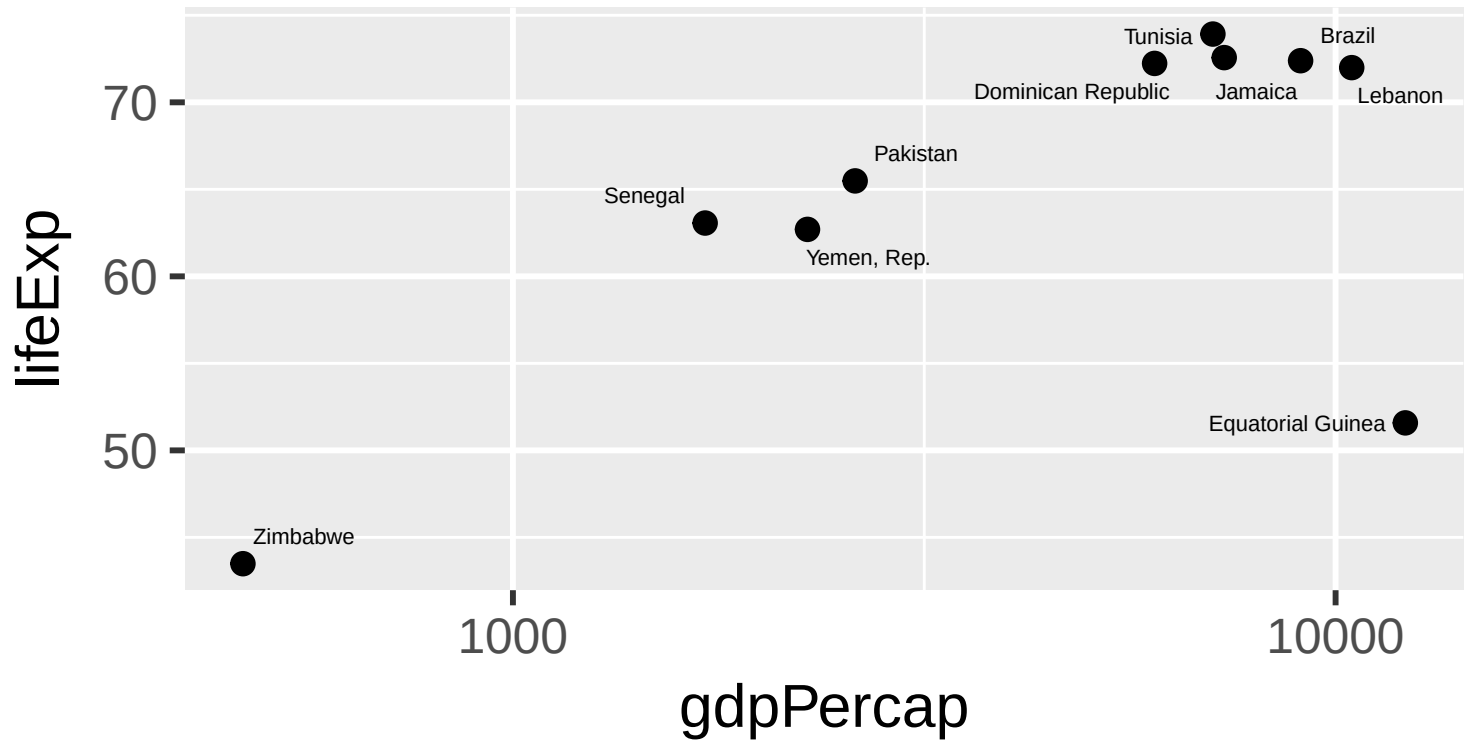
# Adding labels

```
p + geom_text(aes(label = country), size = 1.5)
```



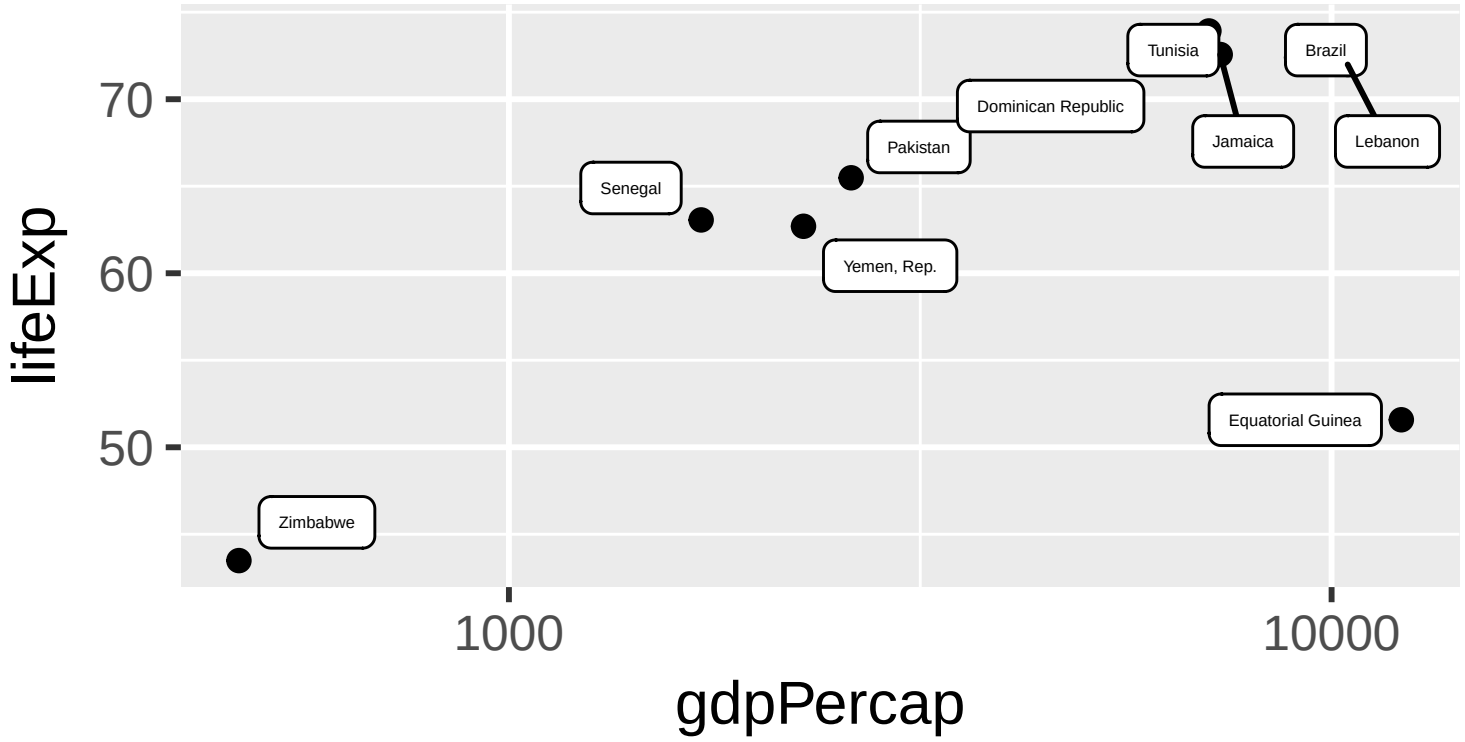
# Adding labels

```
library(ggrepel)
p + geom_text_repel(aes(label = country), size = 1.5)
```



# Adding labels

```
p + geom_label_repel(aes(label = country), size = 1.2)
```

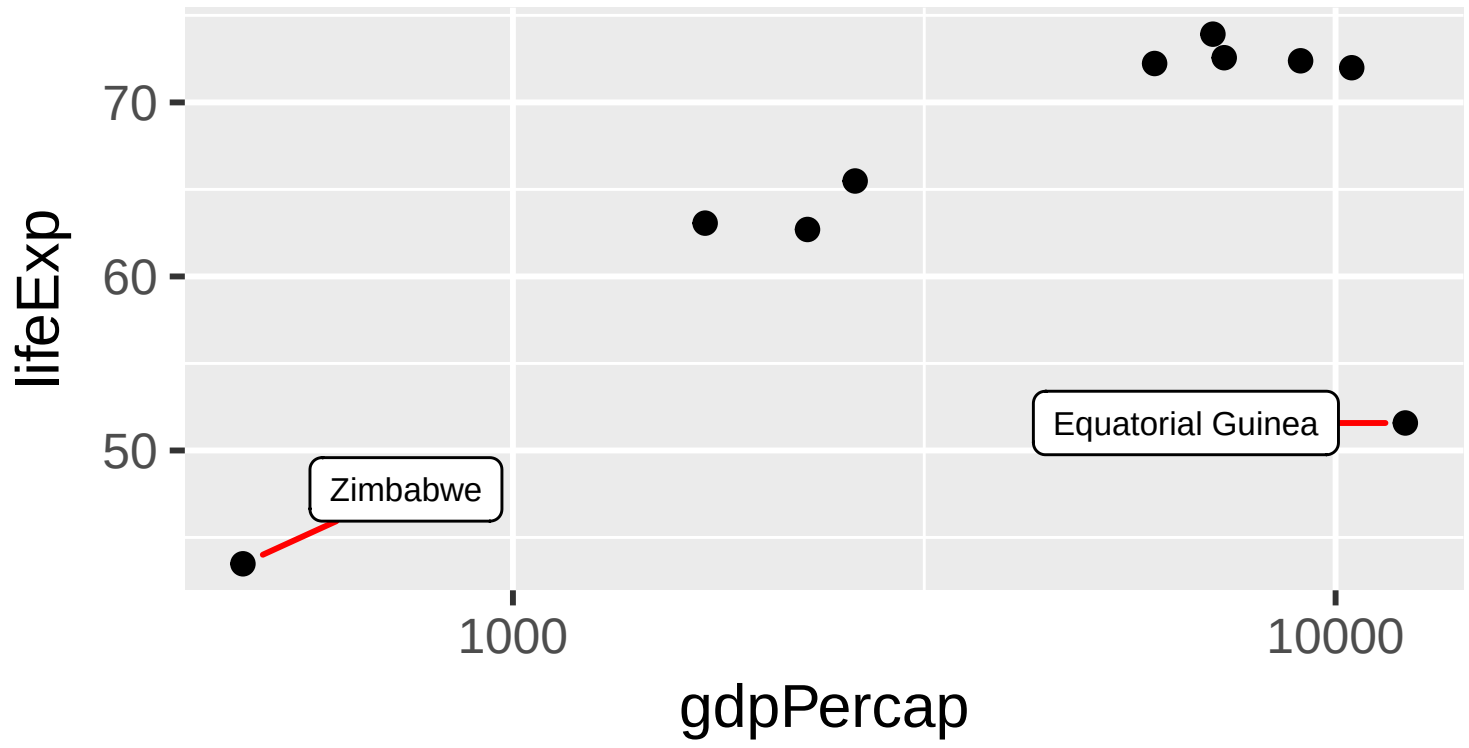


# Adding better labels

```
p1 <- p + geom_label_repel(data = filter(gapminder2,
          lifeExp < 60),
          aes(label = country),
    box.padding = unit(0.35, "lines"),
    point.padding = unit(0.5, "lines"),
    size = 2,
    segment.color = 'red')
```

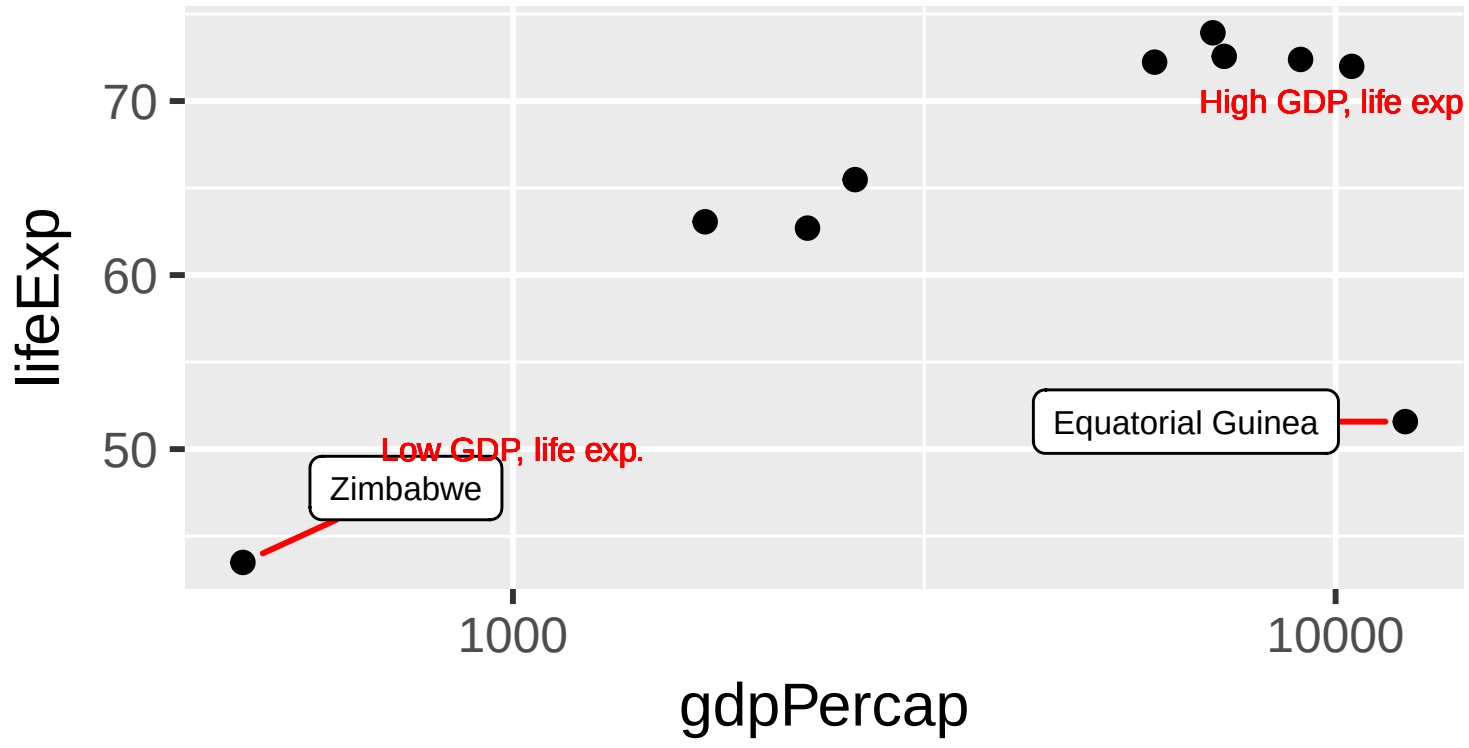
# Adding better labels

```
p1
```



# Adding manual objects

```
p1 + geom_text(aes(x = 1000, y = 50), label = "Low GDP, life exp.",
                 colour = "red", size = 2) +
  geom_text(aes(x = 10000, y = 70), label = "High GDP, life exp.",
              colour = "red", size = 2)
```



# Adding manual objects

1. Segments
2. Arrows
3. Pictures
4. What else?

# Animation

Steps to animation (Mac.. PC?)

1. Install macports at macports.org
2. Command line: `sudo port install ImageMagick`
3. Install `gganimate` package

# Animation

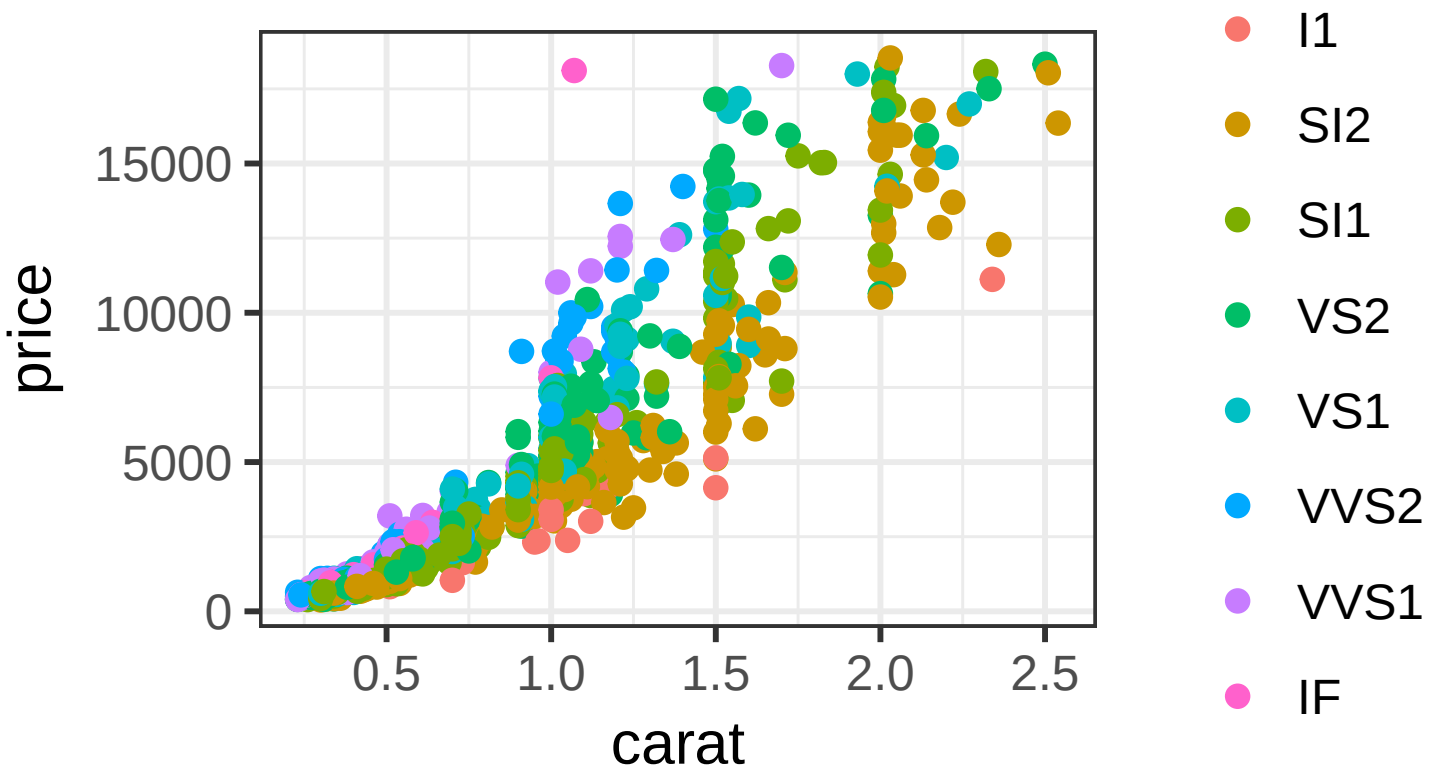
```
library(gganimate)
library(gapminder)
library(ggplot2)
theme_set(theme_bw())

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

# Animation (show ups in knit .Rmd's)

```
#gg_animate(p)
```

# Interactivity (shows up in knit .Rmd's)

```
library(plotly)
dsamp <- sample_n(diamonds, 1000)
ggplot(dsamp, aes(carat, price, colour=clarity)) +
  geom_point()
```



# Interactivity

Shootouts in the NHL

# Saving images

```
#ggsave(p1, file = "gapminder.pdf", width = 6, height = 5)
#ggsave(p1, file = "gapminder.png", width = 6, height = 5)
```