Class 05: Data Visualization with GGPLOT

Madeleine Duquette
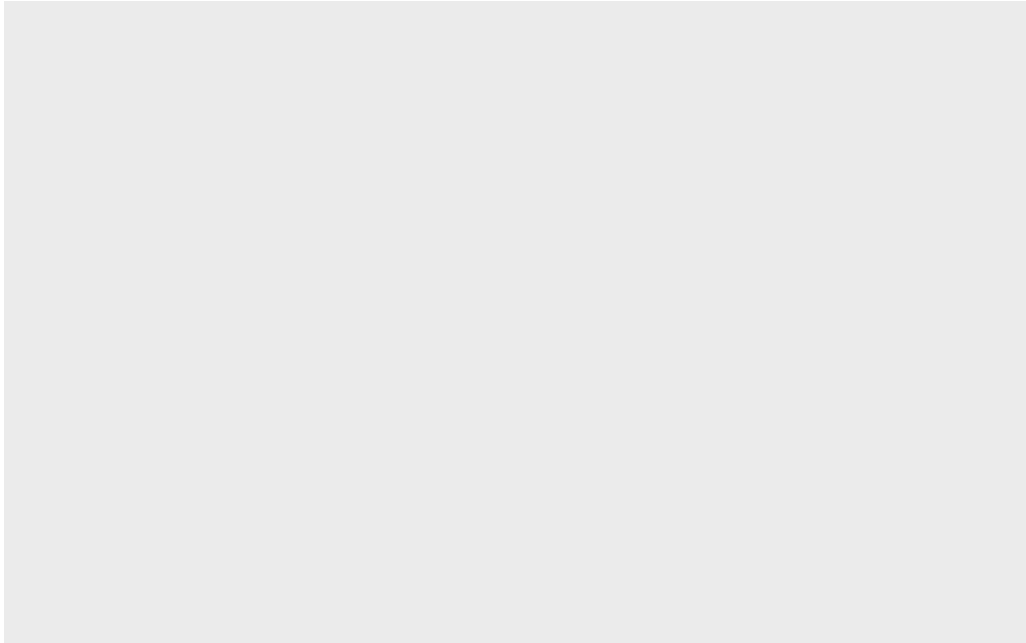
Table of contents

Our First Plot

R has base graphics

```
plot(cars)
```

How would I plot this with 'ggplot2'?

We need to install and load the ggplot2 package first. To install any package in R we use the 'install.packages()' function. To load the package 'library()'

```
# install.packages("ggplot2")

library(ggplot2)
ggplot(cars)
```

Every ggplot needs at least 3 layers: -**Data** (i.e. the data.frame) -**Aes** (the aesthetic mapping of what we want to plot(data–>aesthetics)) -**Geoms** (how we want to plot it!)

```
ggplot(data=cars) +
  aes(x=speed, y=dist) +
  geom_point() +
  geom_line()
```
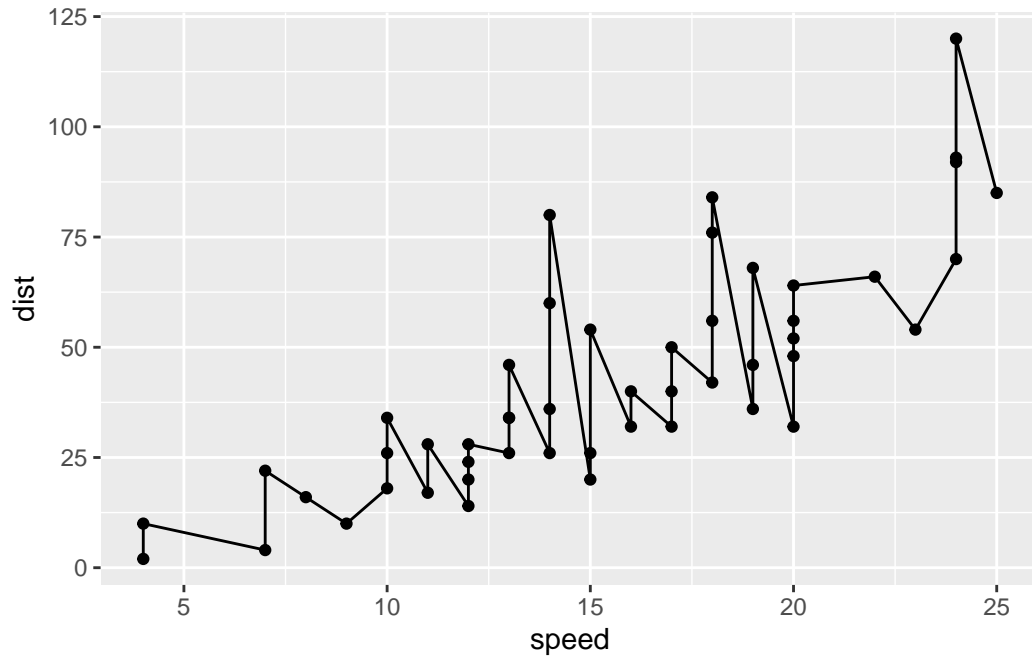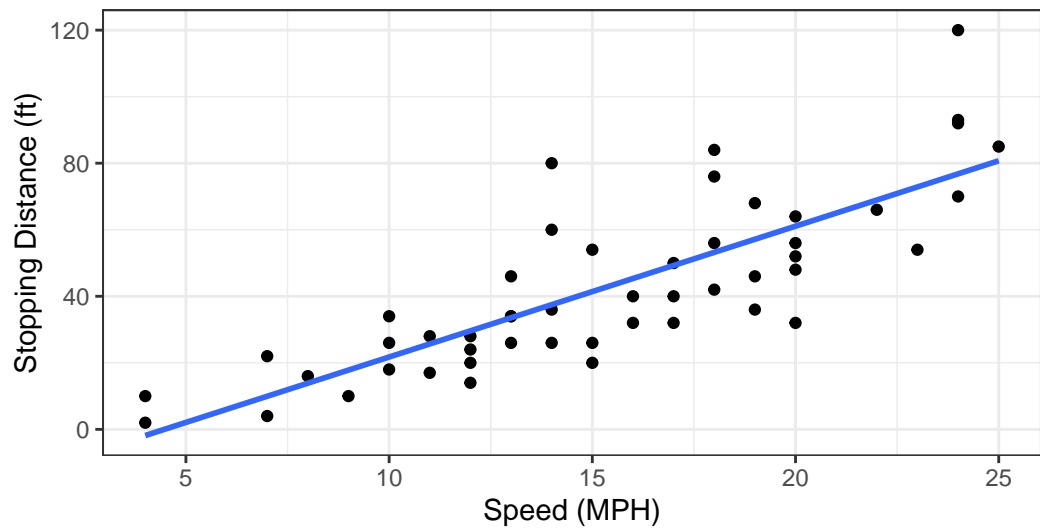
Add another geom

```r
ggplot(data=cars) +
  aes(x=speed, y=dist) +
  geom_point() +
  geom_smooth(method = lm, se=FALSE) +
  labs(title="Speed and Stopping Distances of Cars",
       x="Speed (MPH)",
       y="Stopping Distance (ft)",
       subtitle = "Your informative subtitle text here",
       caption="Dataset: 'cars'") +
  theme_bw()
```

`geom_smooth()` using formula 'y ~ x'

Speed and Stopping Distances of Cars
Your informative subtitle text here

Dataset: 'cars'

LAB ASSIGNMENT

```
url <- "https://bioboot.github.io/bimm143_S20/class-material/up_down_expression.txt"
genes <- read.delim(url)
head(genes)
```

```
       Gene Condition1 Condition2      State
1      A4GNT -3.6808610 -3.4401355 unchanging
2       AAAS  4.5479580  4.3864126 unchanging
3      AASDH  3.7190695  3.4787276 unchanging
4       AATF  5.0784720  5.0151916 unchanging
5       AATK  0.4711421  0.5598642 unchanging
6 AB015752.4 -3.6808610 -3.5921390 unchanging
```

```
ggplot(genes) +
  aes(x=Condition1, y=Condition2) +
  geom_point()
```
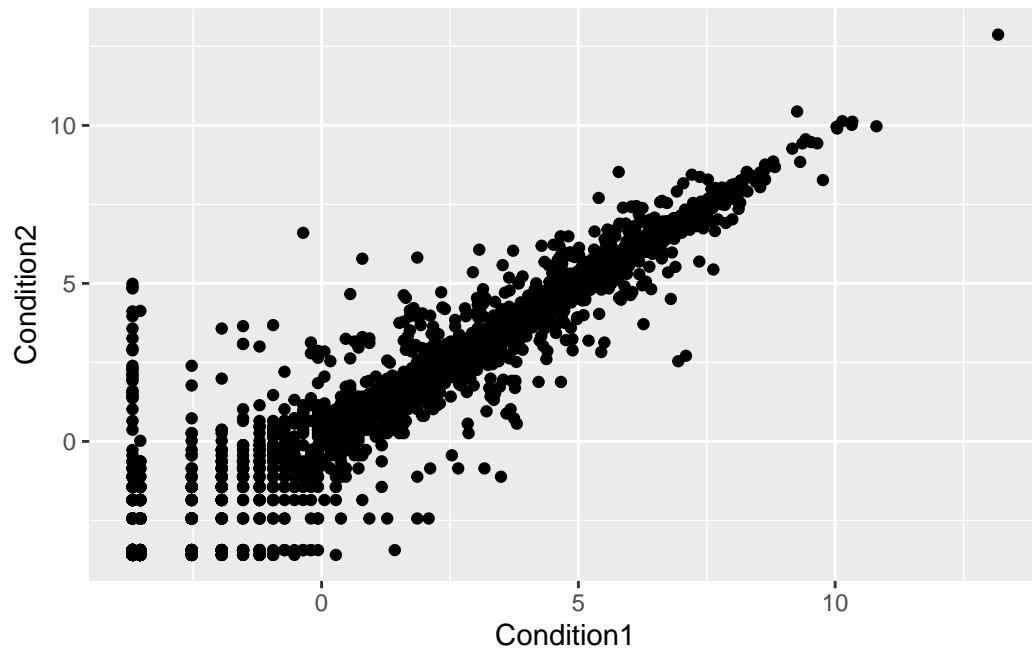
```
# How many genes in the data
Total_genes <- nrow(genes)
#Column names
colnames(genes)
```

```
[1] "Gene"       "Condition1" "Condition2" "State"
```

```
# How many columns
ncol(genes)
```

```
[1] 4
```

```
#Find out how many 'up' regulated genes there are
table(genes$State)
```

```
    down unchanging         up
      72       4997        127
```
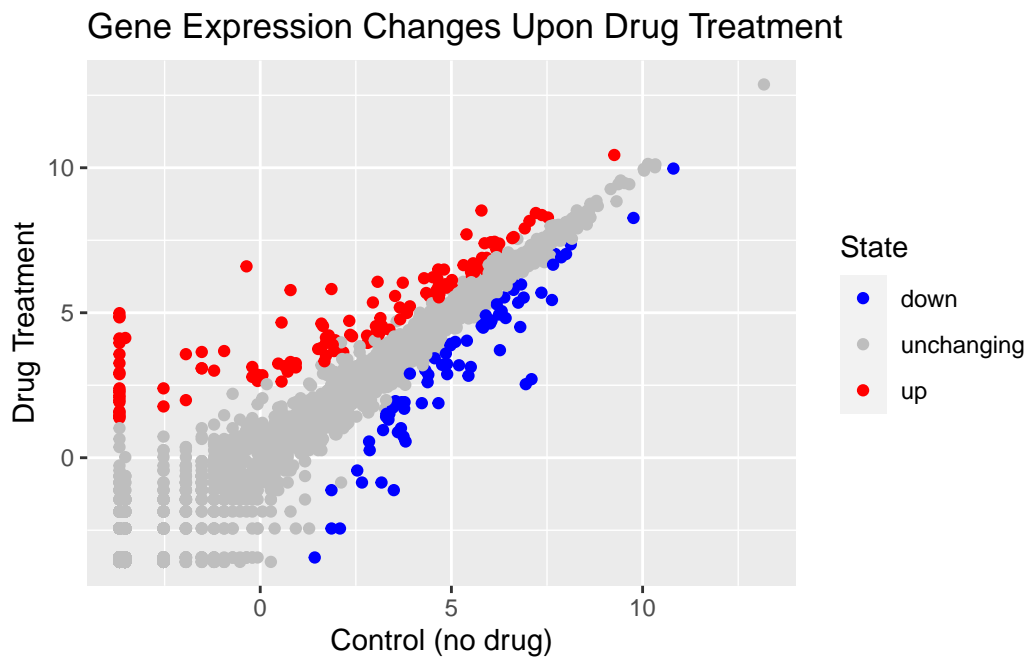
```
#Fraction of upregulated genes?
round( table(genes$State)/Total_genes * 100, 2)
```

```
    down unchanging         up
    1.39      96.17       2.44
```

There are 5196 in the dataset

```
p <- ggplot(genes) +
  aes(x=Condition1, y=Condition2, col=State) +
  geom_point()
```

```
p + scale_color_manual(values = c("blue","gray","red")) +
  labs(x = "Control (no drug)", y = "Drug Treatment", title = "Gene Expression Changes Upo
```

Bonus Things

```r
#install.packages("gapminder")
library(gapminder)
#install.packages("dplyr")
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
```
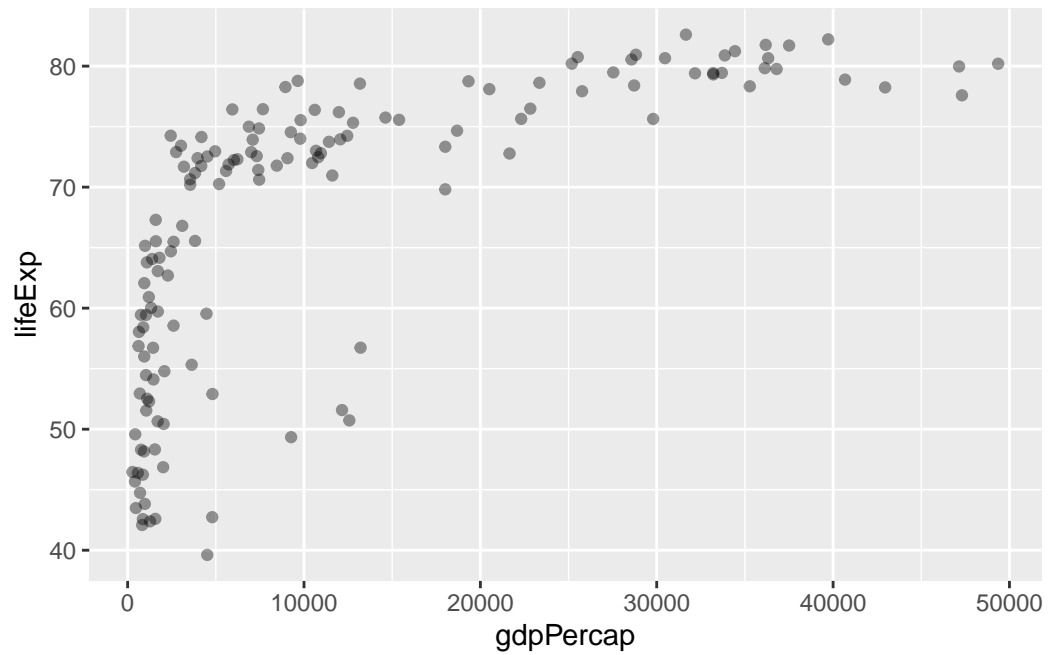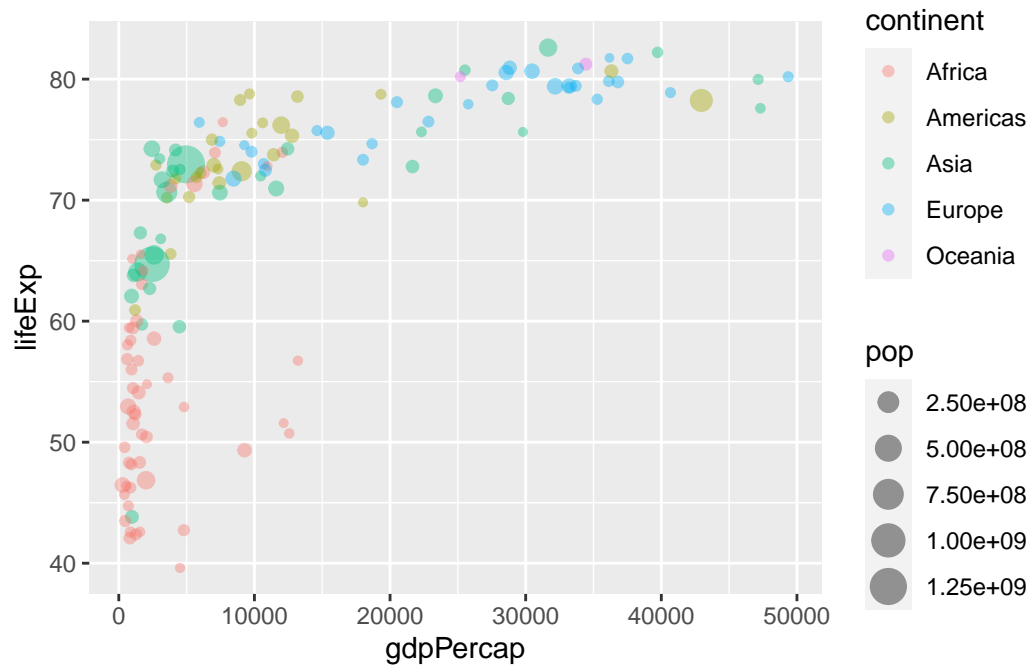
```r
gapminder_2007 <- gapminder %>% filter(year==2007)

ggplot(gapminder_2007) +
  aes(x=gdpPercap, y=lifeExp) +
  geom_point(alpha = 0.4)
```
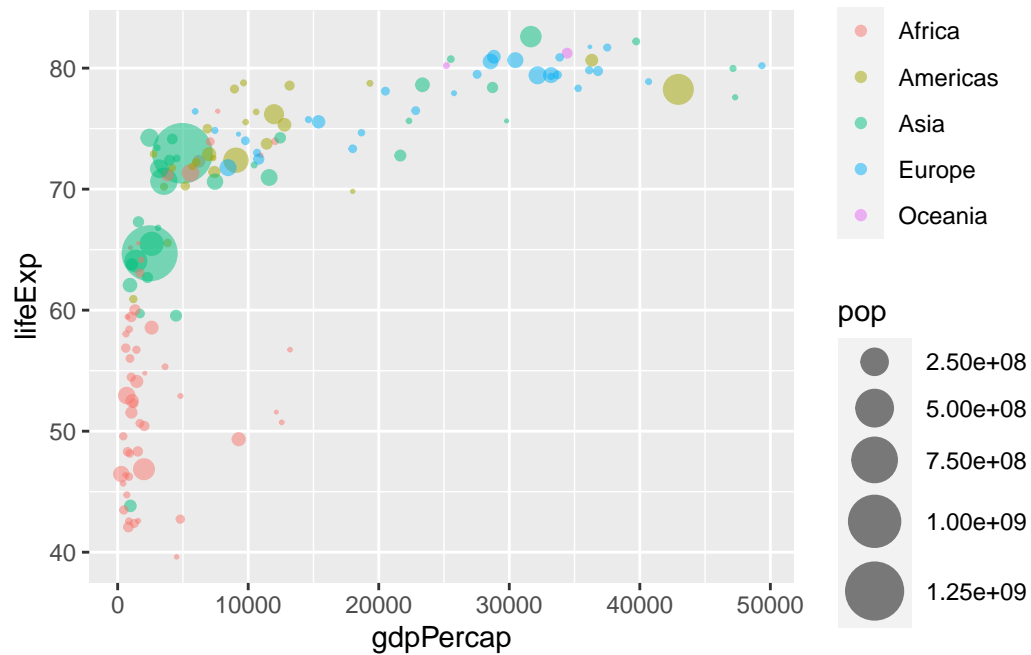
```
ggplot(gapminder_2007) +
  aes(x=gdpPercap, y=lifeExp, color=continent, size=pop) +
  geom_point(alpha = 0.4)
```
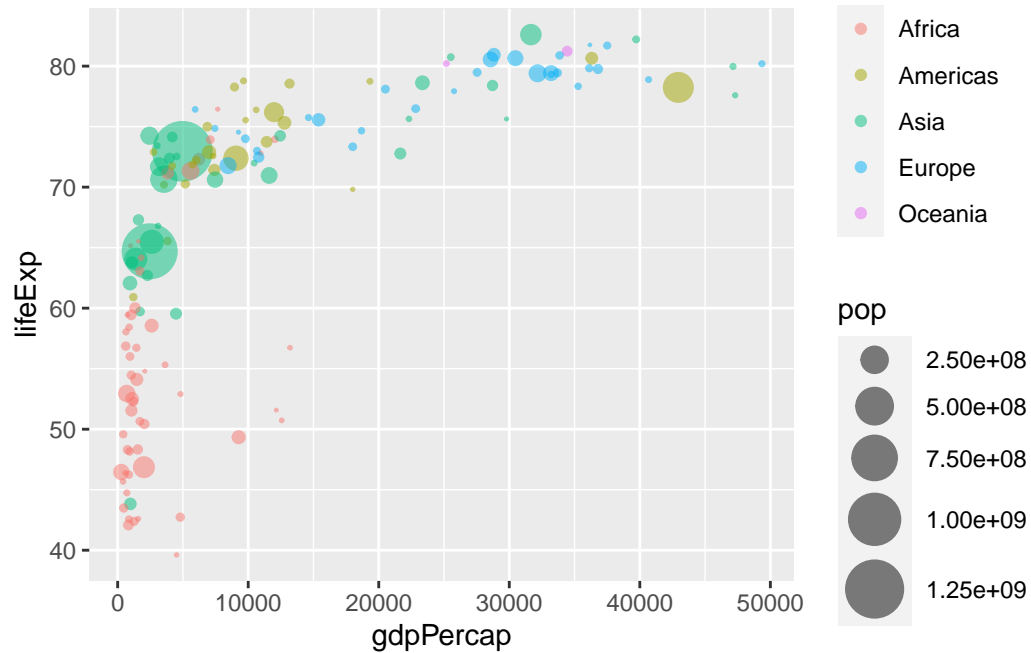
```
ggplot(gapminder_2007) +
  geom_point(aes(x=gdpPercap, y=lifeExp, size=pop, color = continent), alpha = 0.5) +
  scale_size_area(max_size = 10)
```

Checking to see if there's a difference between passing aes into geom_point vs. having it as a separate function...
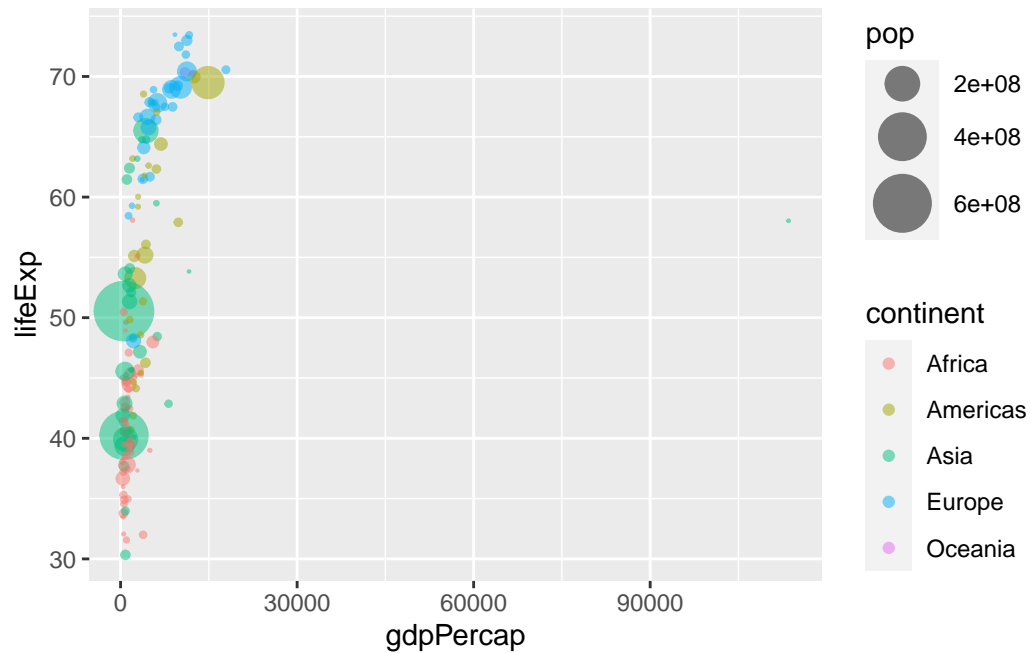
```
ggplot(gapminder_2007) +
  aes(x=gdpPercap, y=lifeExp, size=pop, color = continent) +
geom_point(alpha = 0.5) +
  scale_size_area(max_size = 10)
```

Repeat GapMinder for 1957

```r
gapminder_1957 <- gapminder %>% filter(year==1957)

ggplot(gapminder_1957) +
  aes(x=gdpPercap, y=lifeExp, size=pop, color = continent) +
geom_point(alpha = 0.5) +
  scale_size_area(max_size = 10)
```
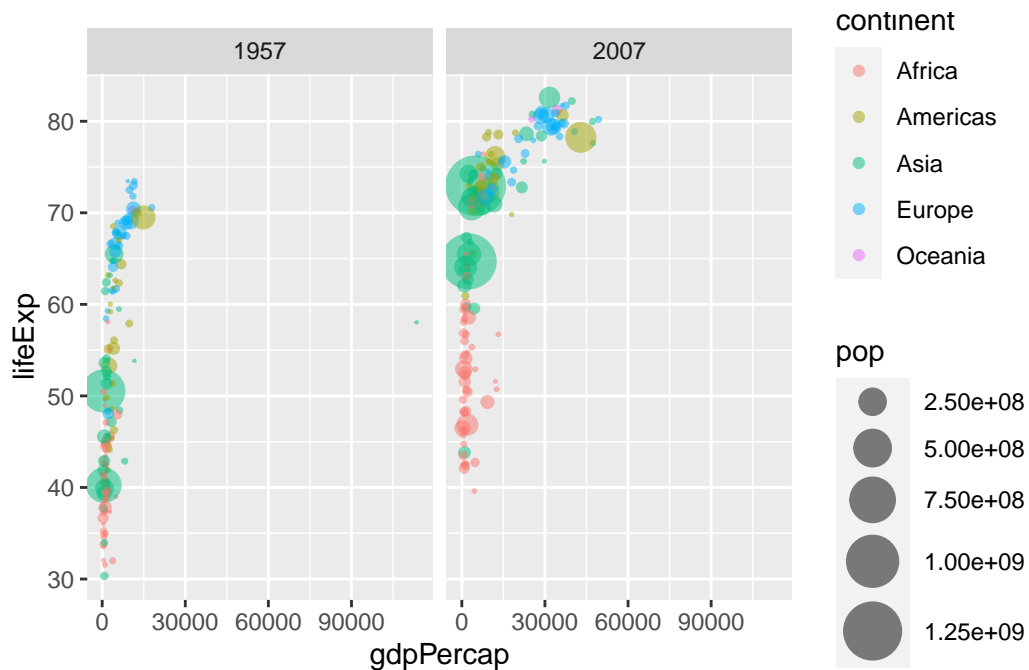
Compare 1957 and 2007 Side-by-Side

```r
gapminder_1957 <- gapminder %>% filter(year==1957 | year==2007)

ggplot(gapminder_1957) +
  aes(x=gdpPercap, y=lifeExp, size=pop, color = continent) +
geom_point(alpha = 0.5) +
  scale_size_area(max_size = 10) +
  facet_wrap(~year)
```

```
gapminder_top5 <- gapminder %>%
  filter(year==2007) %>%
  arrange(desc(pop)) %>%
  top_n(5, pop)

gapminder_top5
```

```
# A tibble: 5 x 6
  country       continent  year lifeExp        pop gdpPercap
  <fct>         <fct>     <int> <dbl>      <int>     <dbl>
1 China         Asia       2007  73.0 1318683096      4959.
2 India         Asia       2007  64.7 1110396331      2452.
3 United States Americas   2007  78.2  301139947     42952.
4 Indonesia     Asia       2007  70.6  223547000      3541.
5 Brazil        Americas   2007  72.4  190010647      9066.
```
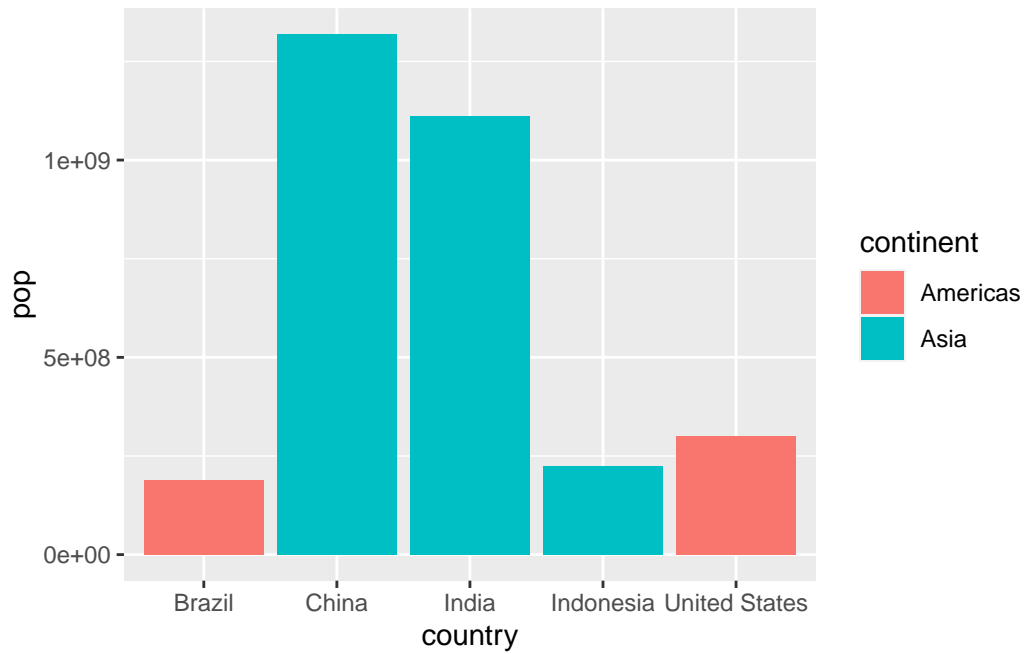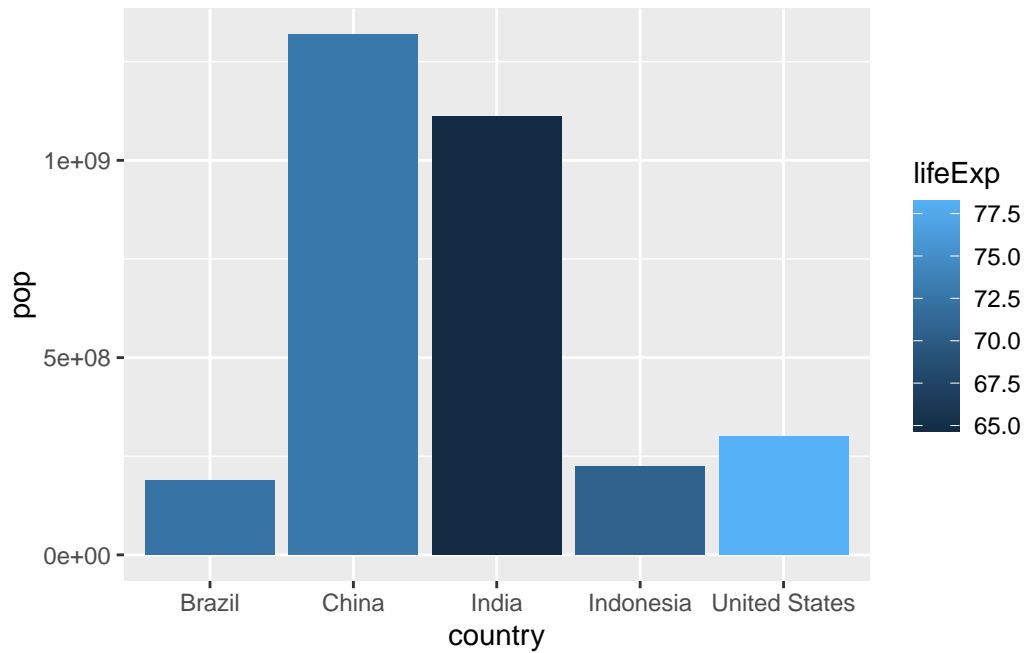
Various Representations of the Top 5

```
ggplot(gapminder_top5) +
  geom_col(aes(x=country, y=pop, fill=continent))
```
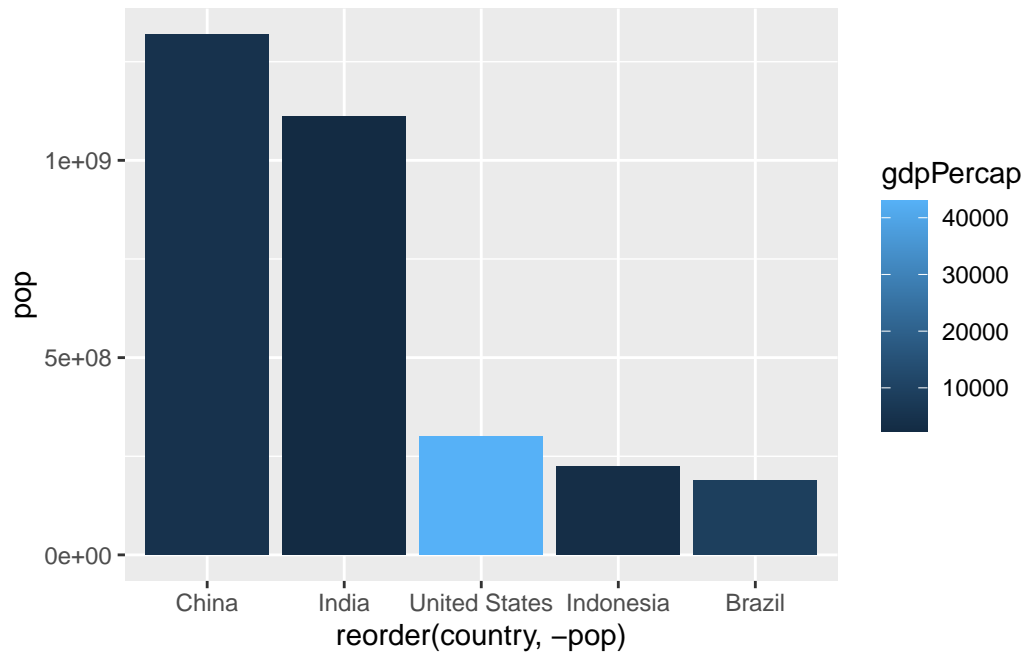


```
ggplot(gapminder_top5) +
  geom_col(aes(x=country, y=pop, fill=lifeExp))
```
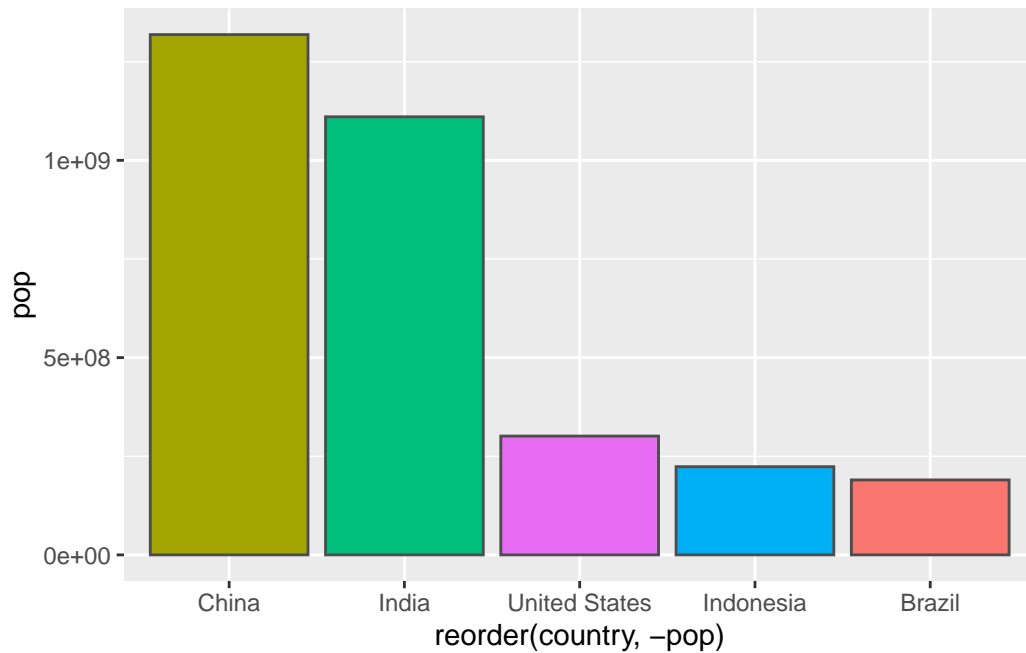
Reordering by Population

```
ggplot(gapminder_top5) +
  aes(x=reorder(country,-pop), y=pop, fill=gdpPercap) +
  geom_col()
```

Changing colors :)

```
ggplot(gapminder_top5) +
  aes(x=reorder(country,-pop), y=pop, fill=country) +
  geom_col(color="gray30") +
  guides(fill="none")
```

Flipping Bar Charts

```
head(USArrests)
```

```
          Murder Assault UrbanPop Rape
Alabama     13.2     236       58 21.2
Alaska      10.0     263       48 44.5
Arizona      8.1     294       80 31.0
Arkansas     8.8     190       50 19.5
California    9.0     276       91 40.6
Colorado     7.9     204       78 38.7
```

```
USArrests$State <- rownames(USArrests)
ggplot(USArrests) +
  aes(x=reorder(State,Murder), y=Murder) +
  geom_point() +
  geom_segment(aes(x=State,
                   xend=State,
                   y=0,
                   yend=Murder), color="blue") +
```

```
coord_flip()
```
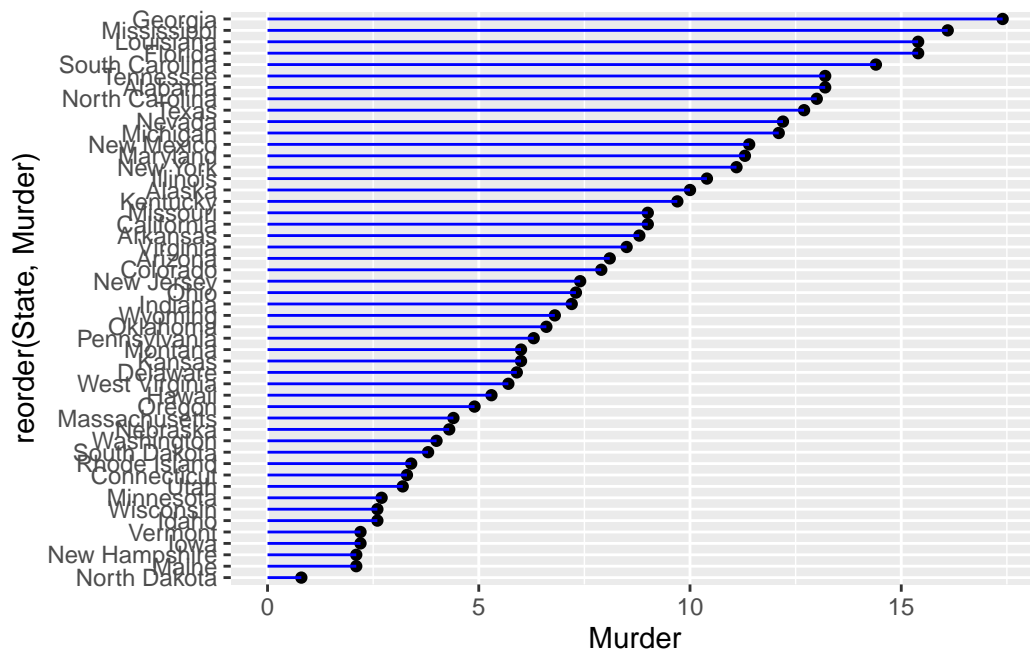


```
#install.packages("gganimate")
library(gapminder)
library(gganimate)

# Setup nice regular ggplot of the gapminder data
ggplot(gapminder, aes(gdpPercap, lifeExp, size = pop, colour = country)) +
  geom_point(alpha = 0.7, show.legend = FALSE) +
  scale_colour_manual(values = country_colors) +
  scale_size(range = c(2, 12)) +
  scale_x_log10() +
  # Facet by continent
  facet_wrap(~continent) +
  # Here comes the gganimate specific bits
  labs(title = 'Year: {frame_time}', x = 'GDP per capita', y = 'life expectancy') +
  transition_time(year) +
  shadow_wake(wake_length = 0.1, alpha = FALSE)
```

```
library(patchwork)

# Setup some example plots
p1 <- ggplot(mtcars) + geom_point(aes(mpg, disp))
p2 <- ggplot(mtcars) + geom_boxplot(aes(gear, disp, group = gear))
p3 <- ggplot(mtcars) + geom_smooth(aes(disp, qsec))
p4 <- ggplot(mtcars) + geom_bar(aes(carb))

# Use patchwork to combine them here:
(p1 | p2 | p3) /
      p4
```

`geom_smooth()` using method = 'loess' and formula 'y ~ x'