

Class 5 Data Visualization with ggplot2

A17576411

Using GGPLOT

The ggplot2 package needs to be installed as it does not come with R “out of the box”.

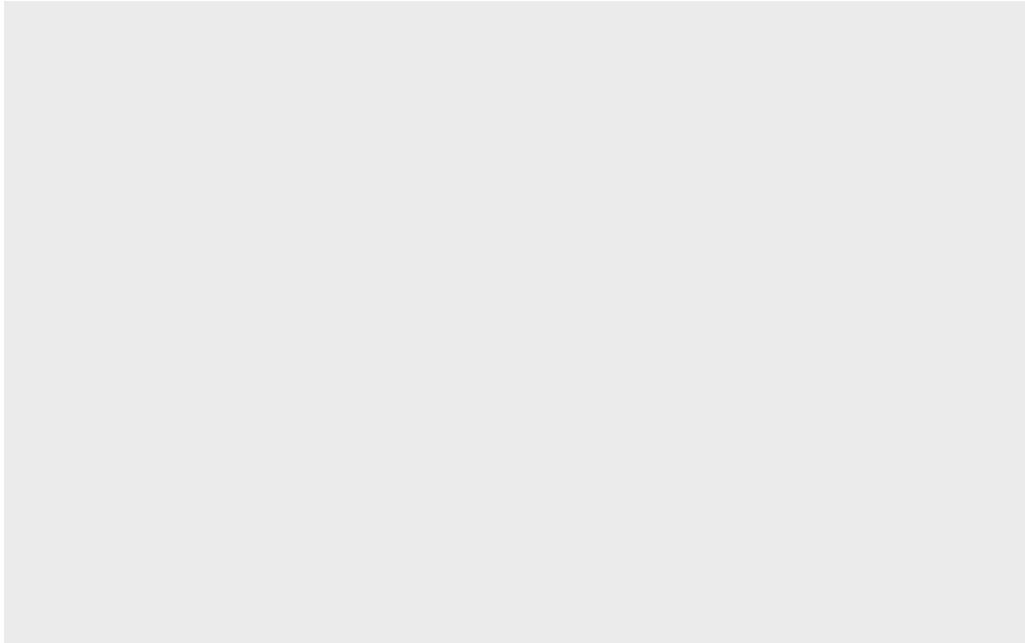
We use the ‘install.packages()’ function to do this.

```
head(cars)
```

	speed	dist
1	4	2
2	4	10
3	7	4
4	7	22
5	8	16
6	9	10

To use ggplot I need to load it up before I can call any of the functions in the package. I do this with the `library()` function.

```
library(ggplot2)  
ggplot()
```



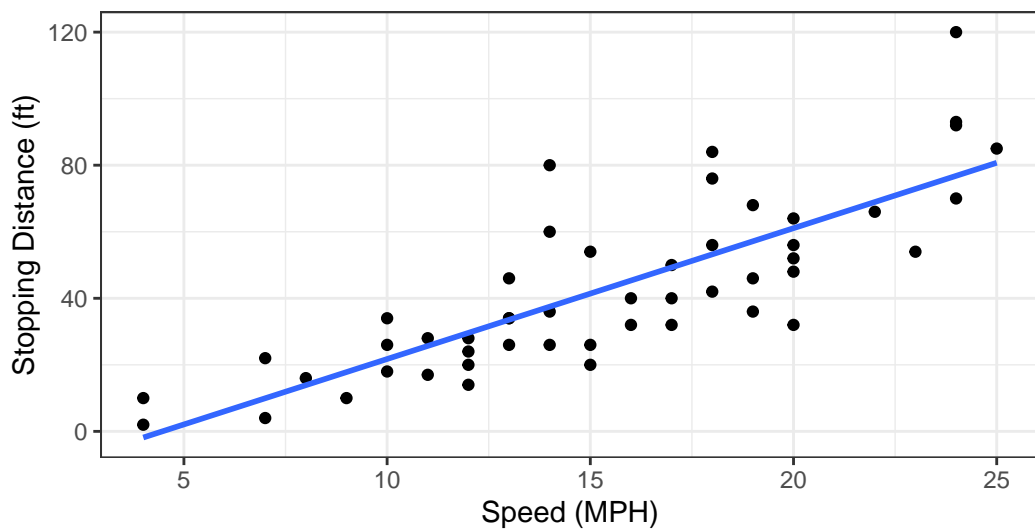
All ggplot figures have at least 3 things: - data (the stuff we want to plot) - aesthetic mapping (aes values) - geoms

```
ggplot(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'

ggplot is not the only graphing system in R there are lots of others. There is even “base R” graphics.

```
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

```
nrow(genes)
```

```
[1] 5196
```

```
ncol(genes)
```

```
[1] 4
```

```
colnames(genes)
```

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

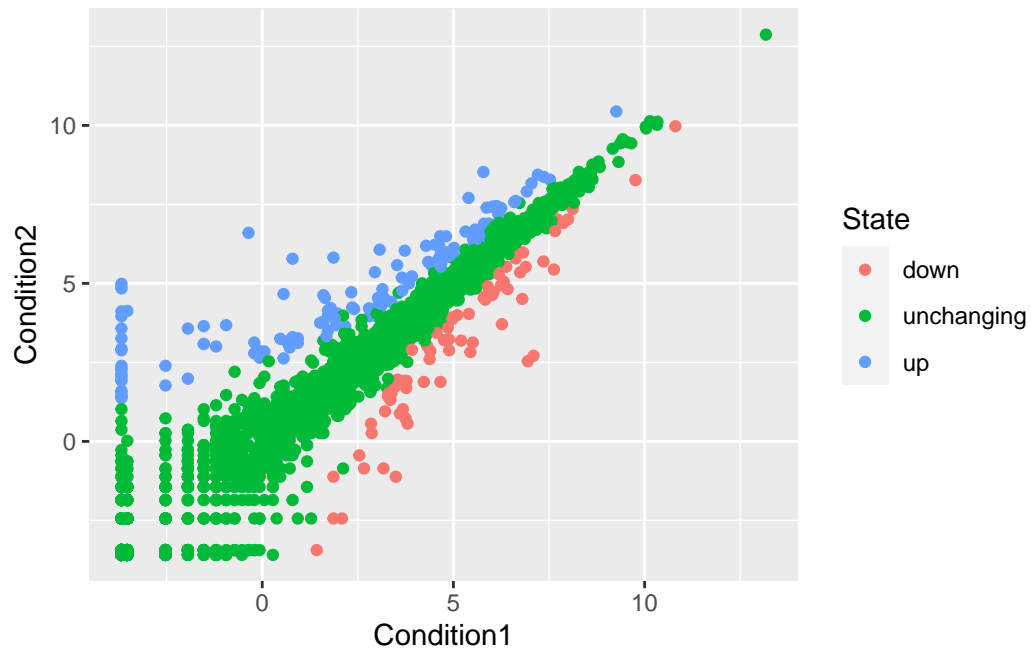
```
table(genes$State)
```

down	unchanging	up
72	4997	127

```
round( table(genes$State)/nrow(genes)* 100,2)
```

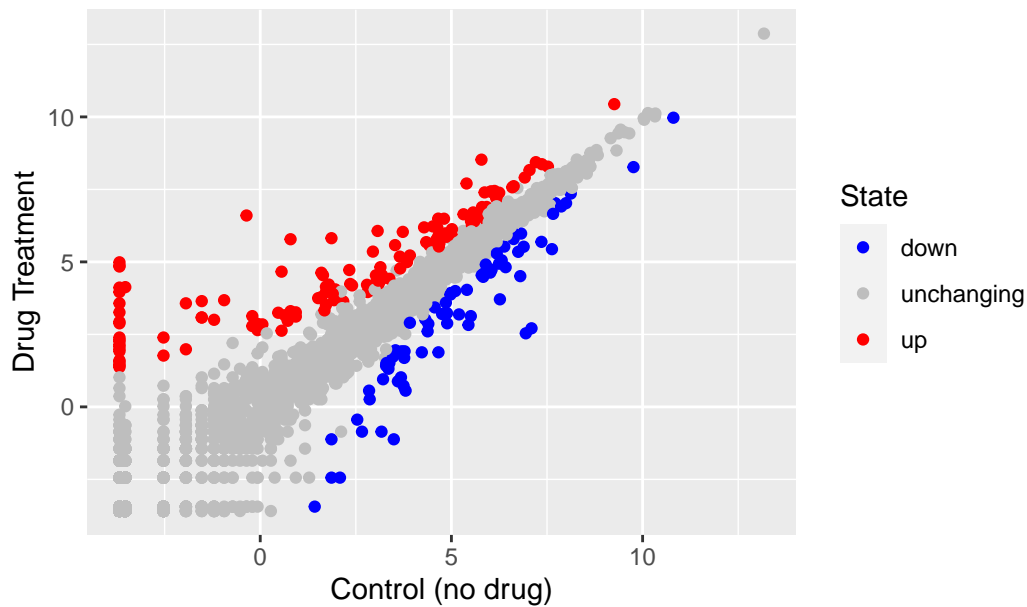
down	unchanging	up
1.39	96.17	2.44

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



```
p <- ggplot (genes) +
  aes(x=Condition1, y=Condition2, col=State) +
  geom_point()
p + scale_colour_manual(values=c("blue","gray","red")) +
  labs(title = "Gene Expression Changes Upon Drug Treatment", y="Drug Treatment", x="Contr
```

Gene Expression Changes Upon Drug Treatment



```
url <- "https://raw.githubusercontent.com/jennybc/gapminder/master/inst/extdata/gapminder."
gapminder <- read.delim(url)
#install.packages("dplyr")
library(dplyr)
```

Attaching package: 'dplyr'

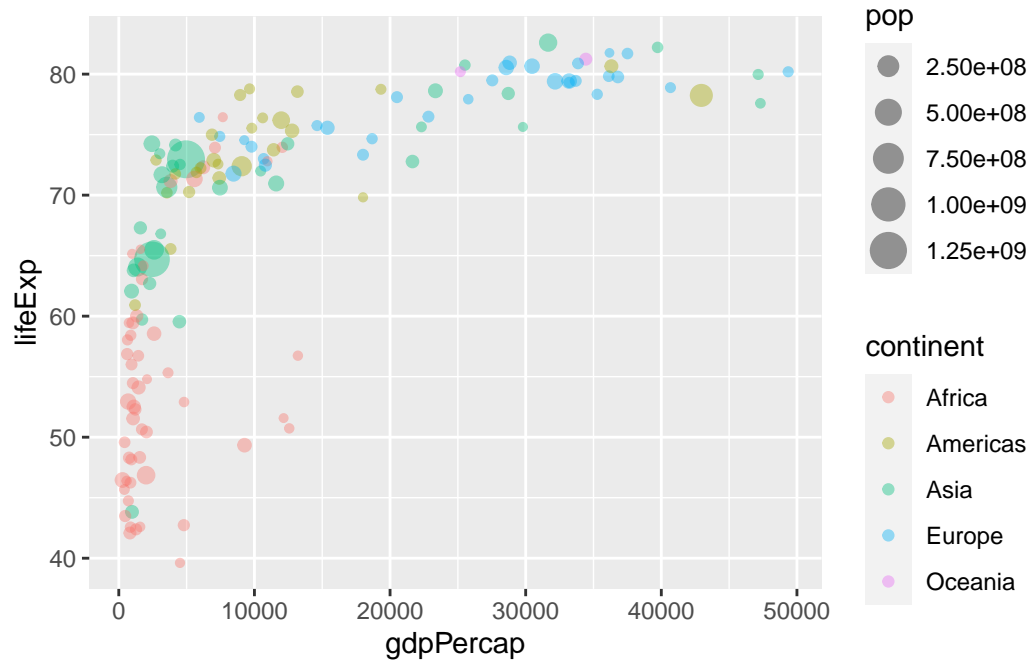
The following objects are masked from 'package:stats':

filter, lag

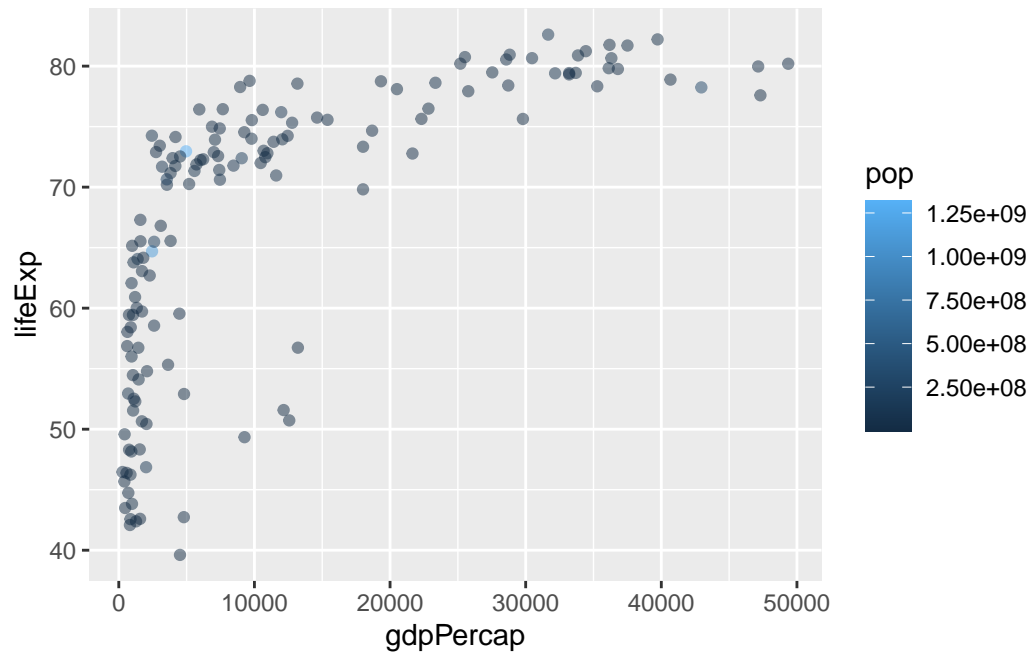
The following objects are masked from 'package:base':

intersect, setdiff, setequal, union

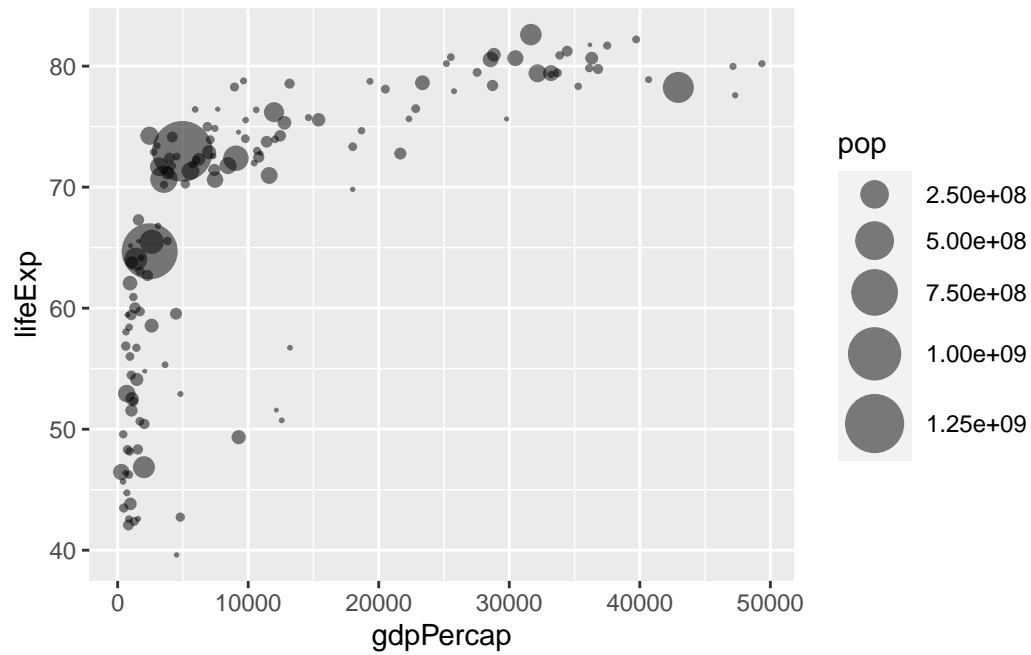
```
gapminder_2007 <- gapminder %>% filter(year==2007)
ggplot(gapminder_2007) +
  aes(x=gdpPercap,y=lifeExp,color=continent, size=pop) +
  geom_point(alpha=0.4)
```



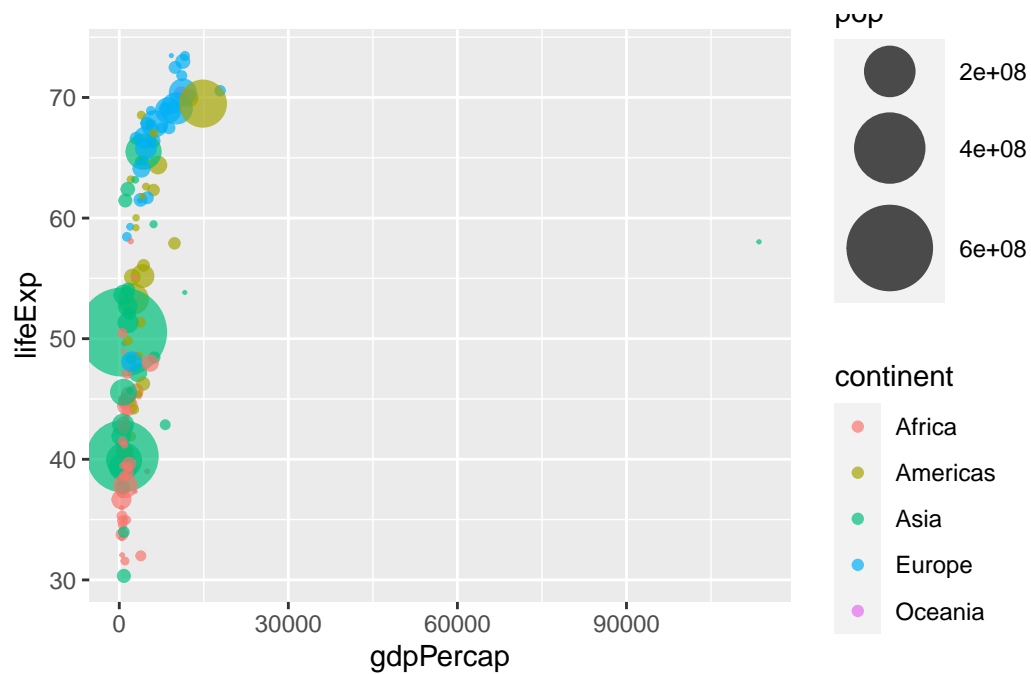
```
ggplot(gapminder_2007) +  
  aes(x=gdpPercap, y=lifeExp, color=pop) +  
  geom_point(alpha=0.5)
```



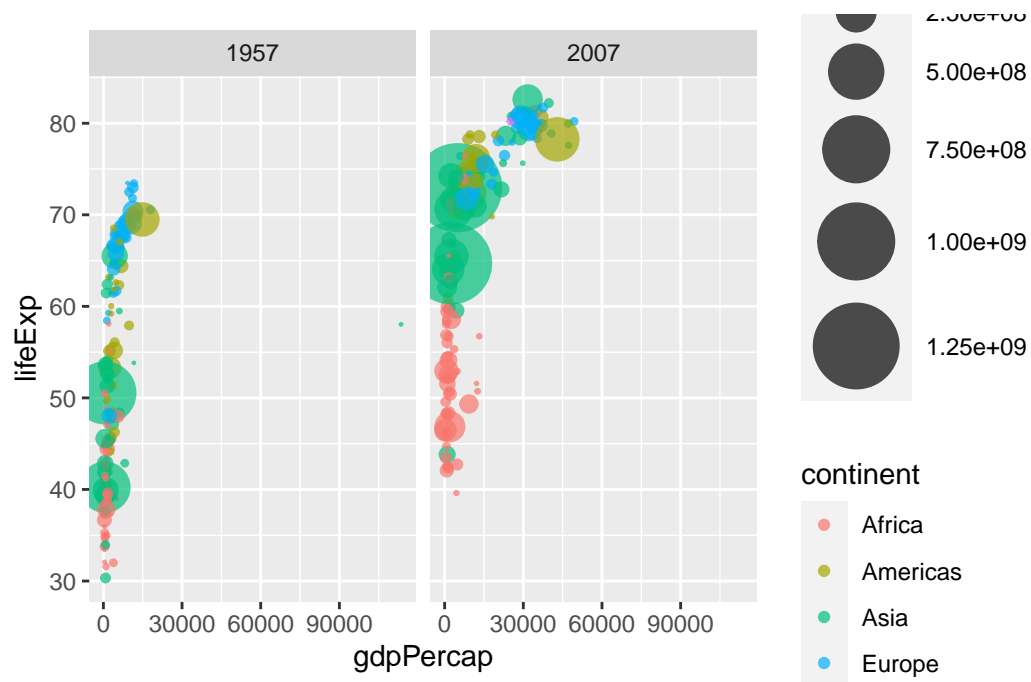
```
ggplot(gapminder_2007) +  
  geom_point(aes(x = gdpPercap, y = lifeExp,  
                 size = pop), alpha=0.5) +  
  scale_size_area(max_size = 10)
```

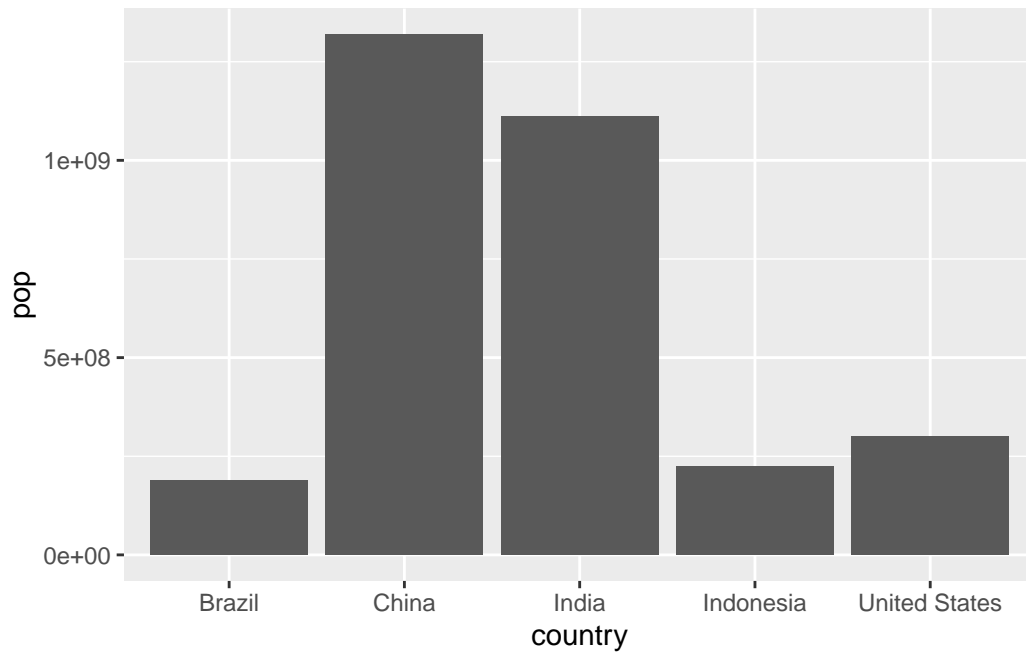
```
gapminder_1957 <- gapminder %>% filter(year==1957)
ggplot(gapminder_1957) +
  geom_point(alpha=0.7) +
  aes(x=gdpPercap, y=lifeExp, color=continent, size=pop) +
  scale_size_area(max_size = 15)
```



```
gapminder_combo <- gapminder %>% filter(year==1957 | year==2007)
ggplot(gapminder_combo) +
  geom_point(alpha=0.7) +
  aes(x=gdpPercap, y=lifeExp, color=continent, size=pop) +
  scale_size_area(max_size = 15) +
  facet_wrap(~year)
```



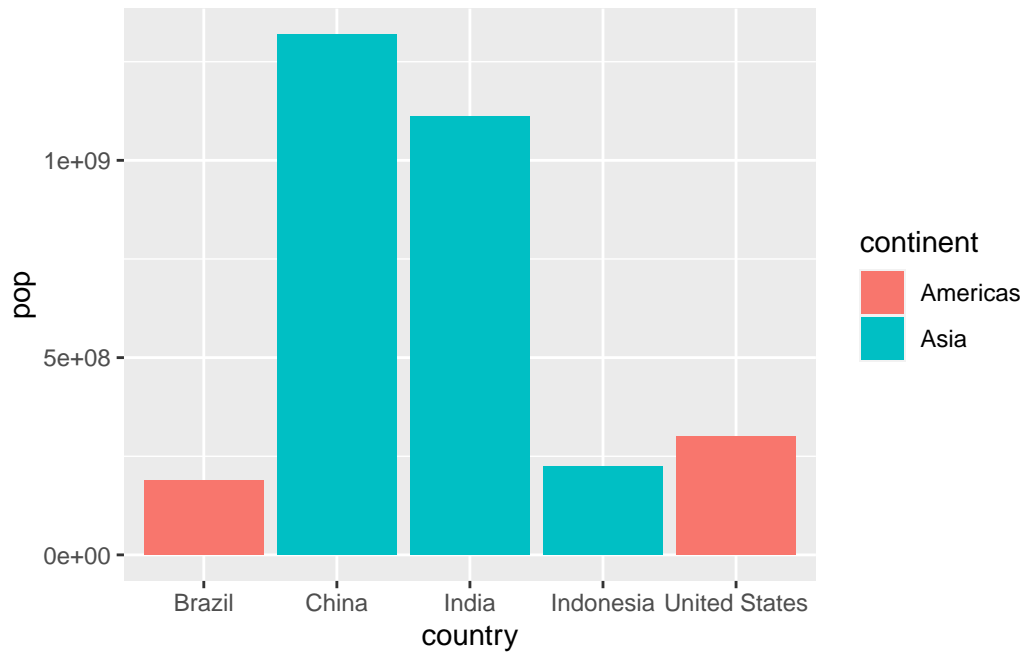
```
gapminder_top5 <- gapminder %>%
  filter(year==2007) %>%
  arrange(desc(pop)) %>%
  top_n(5, pop)
ggplot(gapminder_top5) +
  geom_col(aes(x = country, y = pop))
```



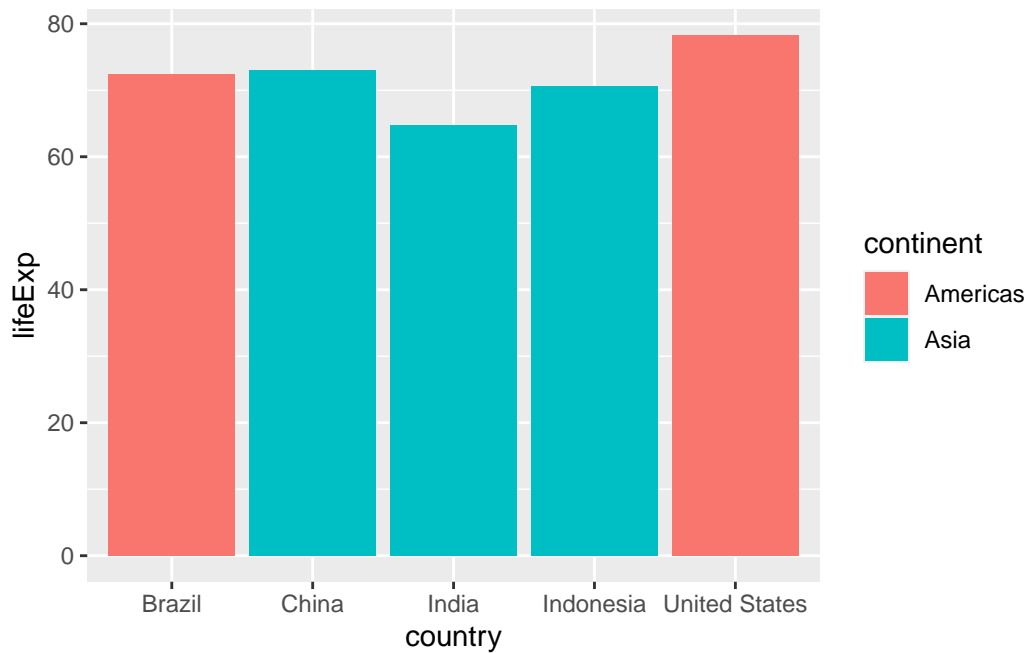
```
gapminder_top5
```

	country	continent	year	lifeExp	pop	gdpPercap
1	China	Asia	2007	72.961	1318683096	4959.115
2	India	Asia	2007	64.698	1110396331	2452.210
3	United States	Americas	2007	78.242	301139947	42951.653
4	Indonesia	Asia	2007	70.650	223547000	3540.652
5	Brazil	Americas	2007	72.390	190010647	9065.801

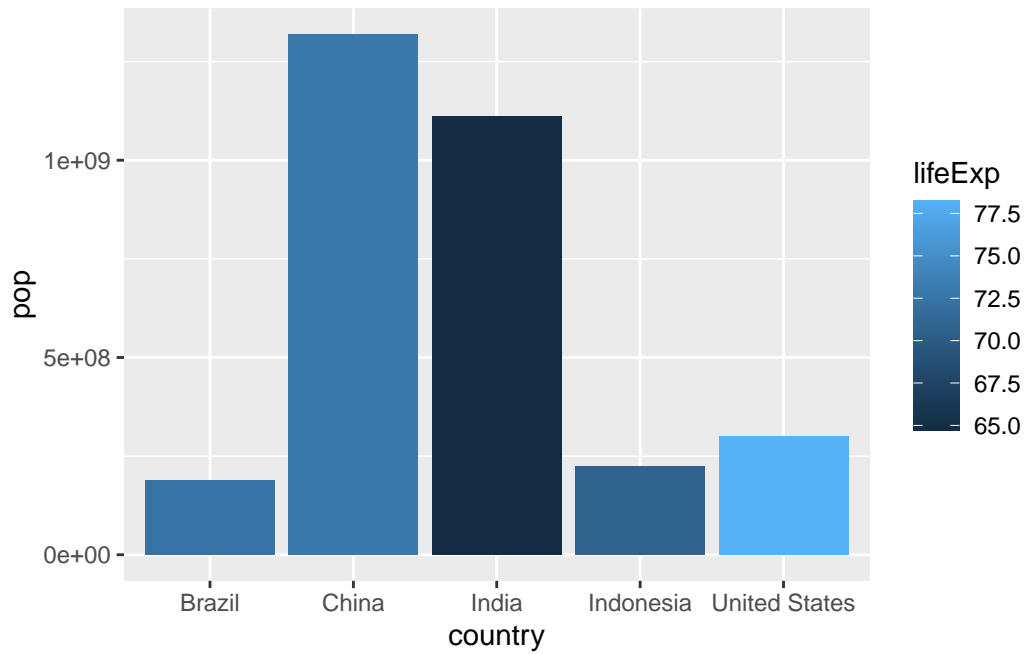
```
ggplot(gapminder_top5) +
  geom_col(aes(x = country, y = pop, fill = continent))
```



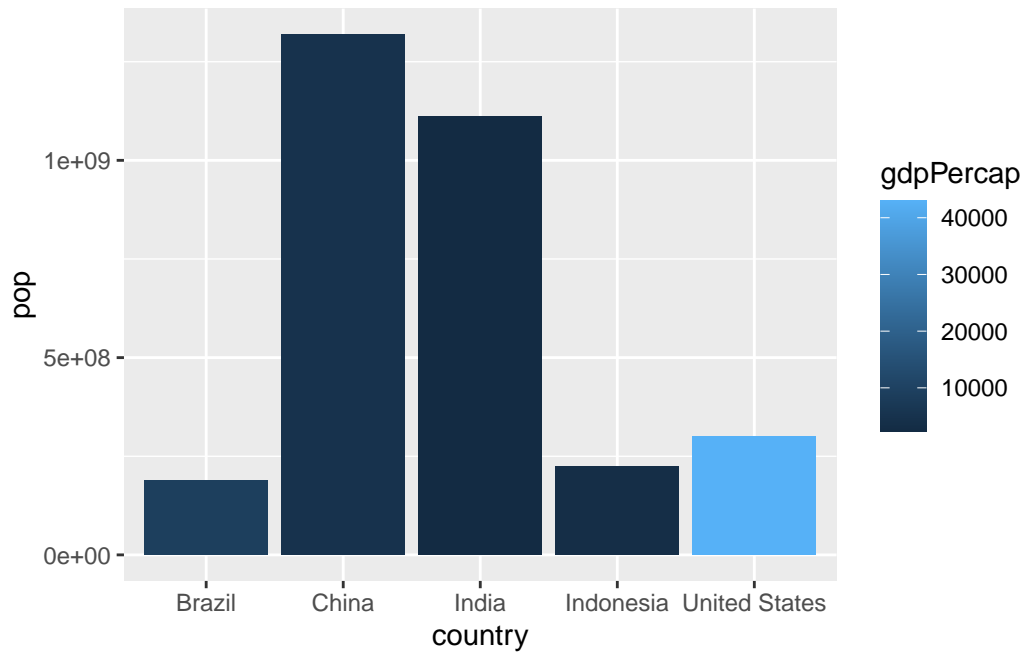
```
ggplot(gapminder_top5) +  
  geom_col(aes(x = country, y = lifeExp, fill = continent))
```



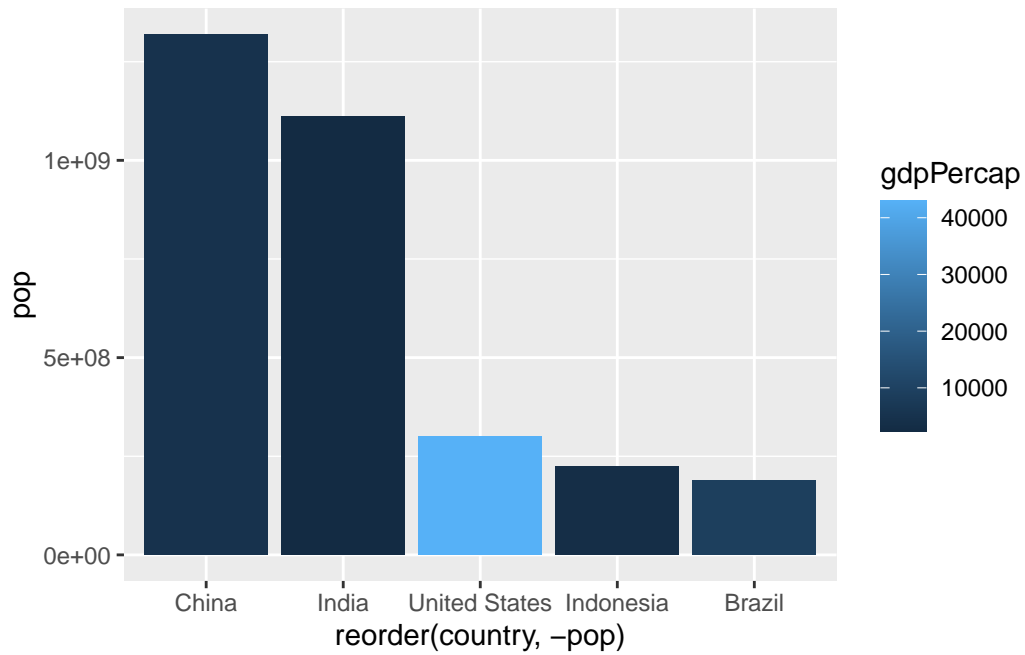
```
ggplot(gapminder_top5) +  
  geom_col(aes(x = country, y = pop, fill = lifeExp))
```



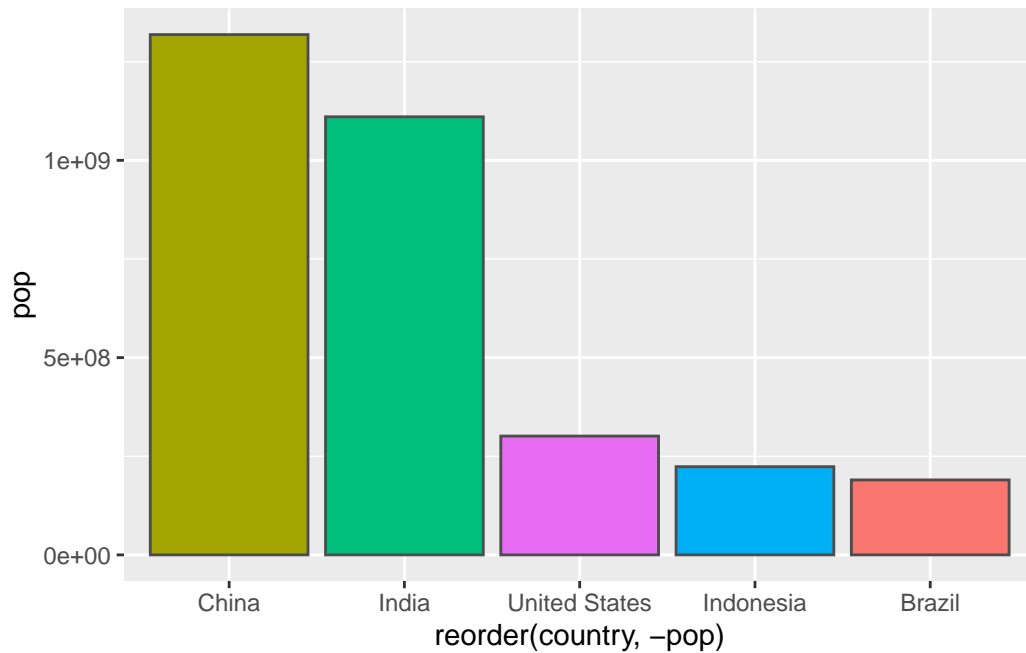
```
ggplot(gapminder_top5) +  
  aes(x=country, y=pop, fill=gdpPercap) +  
  geom_col()
```



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



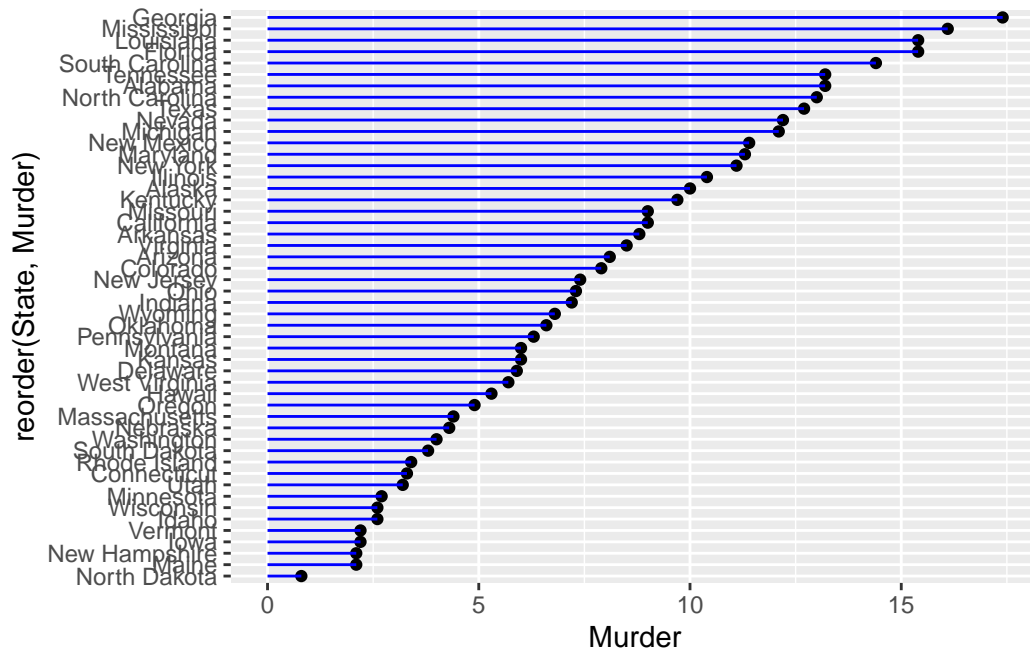
```
ggplot(gapminder_top5) +  
  aes(x=reorder(country, -pop), y=pop, fill=country) +  
  geom_col(col="gray30") +  
  guides(fill="none")
```

```
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()
```



```
#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_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(displ, 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'

