Class 05 Data Visualization

Anita Wang

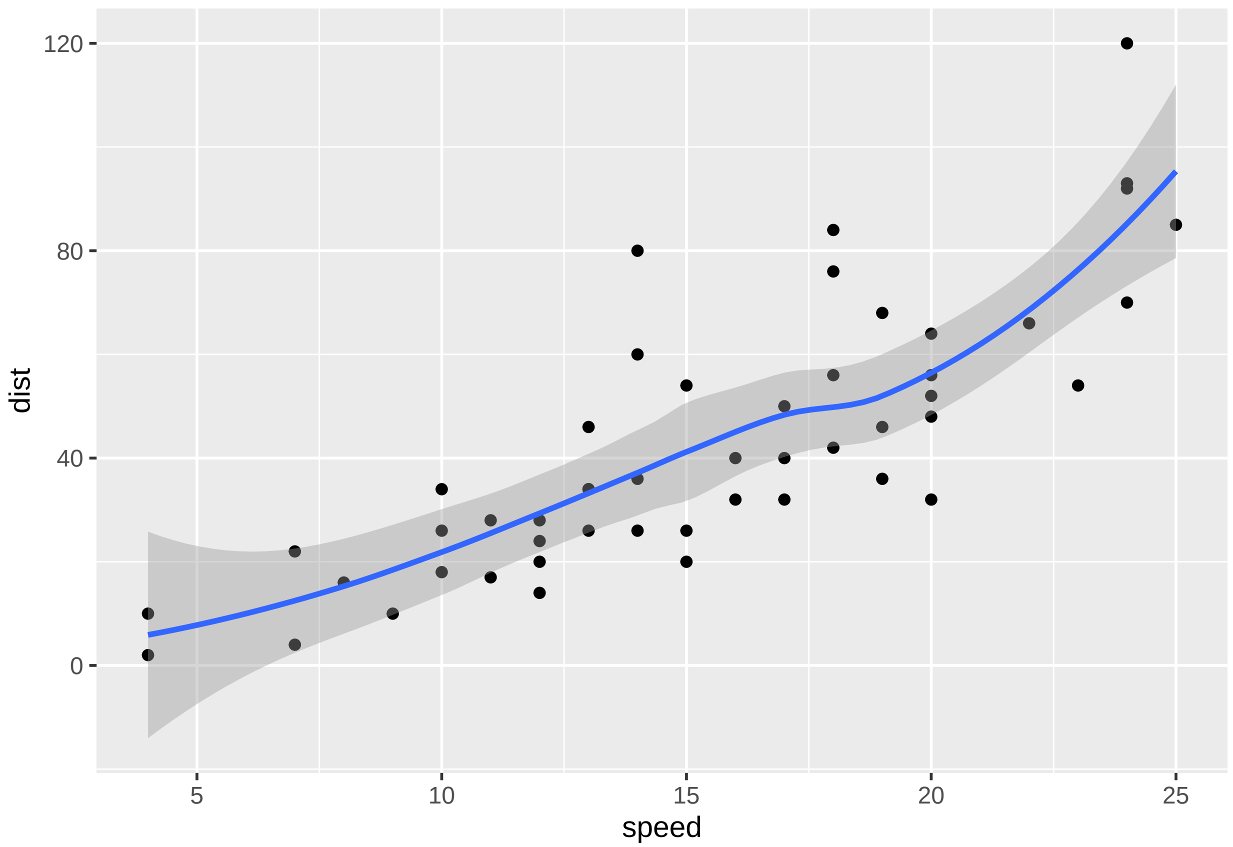
10.12.21

```
#5. Creating Scatter Plots
```

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

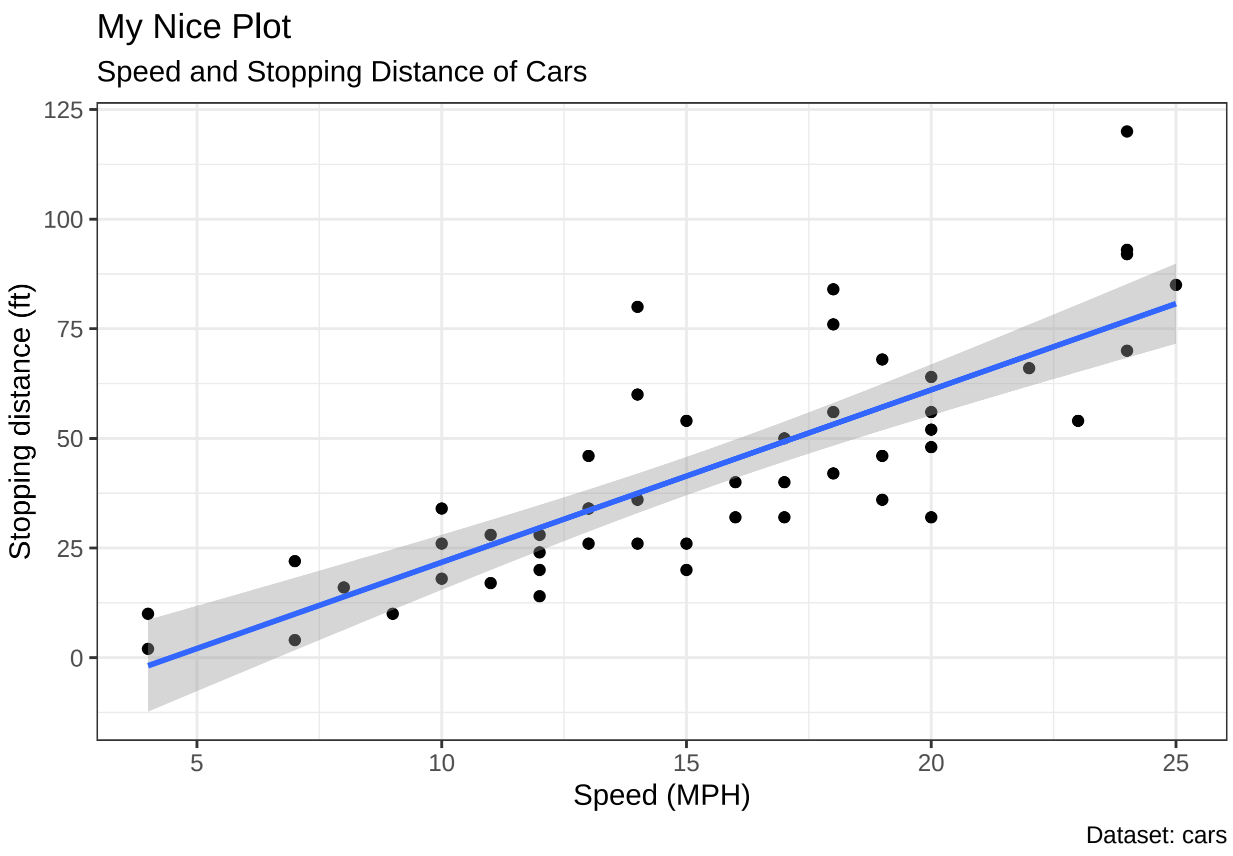
```
#Start with a scatterplot
#Before we can use ggplot, we need to load it up
library(ggplot2)

#Every ggplot has a data + aesthetics + geometry layers -- specify the dataset + aesthetic mappings + g
ggplot(data=cars) +
  aes(x=speed, y=dist) +
  geom_point() +
  geom_smooth()
```

```
## ‘geom_smooth()‘ using method = ’loess’ and formula ’y ~ x’
```

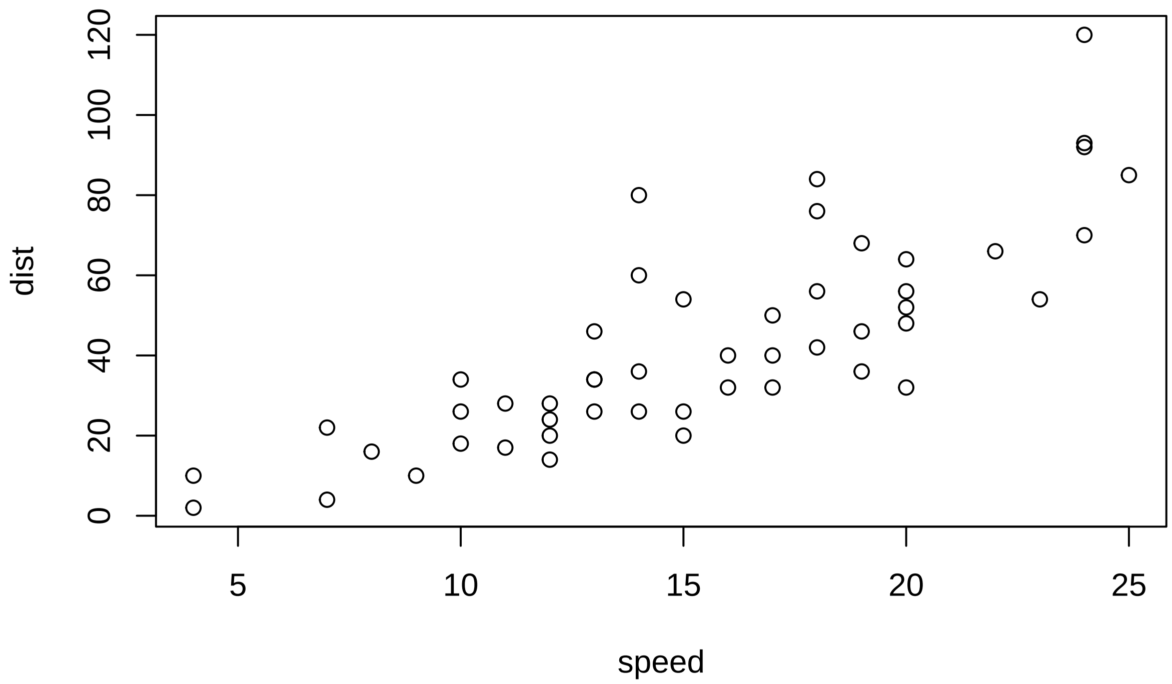


```
#Change to a linear model
p <- ggplot(data=cars) +
  aes(x=speed, y=dist) +
  geom_point() +
  geom_smooth(method="lm") +
  theme_bw()

p + labs(title="My Nice Plot",
            x="Speed (MPH)",
            y="Stopping distance (ft)",
            subtitle="Speed and Stopping Distance of Cars",
            caption="Dataset: cars")
```

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



```r
#Base graphics is shorter
plot(cars)
```



```
#Adding more plot aesthetics through aes()
#Plotting a more relevant data set(results of a differential expression analysis where a new anti-viral
url <- "https://bioboot.github.io/bimm143_S20/class-material/up_down_expression.txt"
genes <- read.delim(url)

#Display top portion of data set
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
```

```
#Q: How many genes in data set?
nrow(genes)
```

```
## [1] 5196
```

```
#Q: What are the column names?
colnames(genes)
```

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

```r
#Q: How many columns are there?
ncol(genes)
```

```
## [1] 4
```

```r
#Q: How to access "state" column --  How many up-regulated genes are there?
table(genes$State)
```

```
## 
##       down unchanging         up 
##         72       4997        127
```

```r
#Q: What fraction of total genes is up-regulated in this dataset?
table(genes$State) / nrow(genes)
```

```
## 
##       down unchanging         up 
## 0.01385681 0.96170131 0.02444188
```

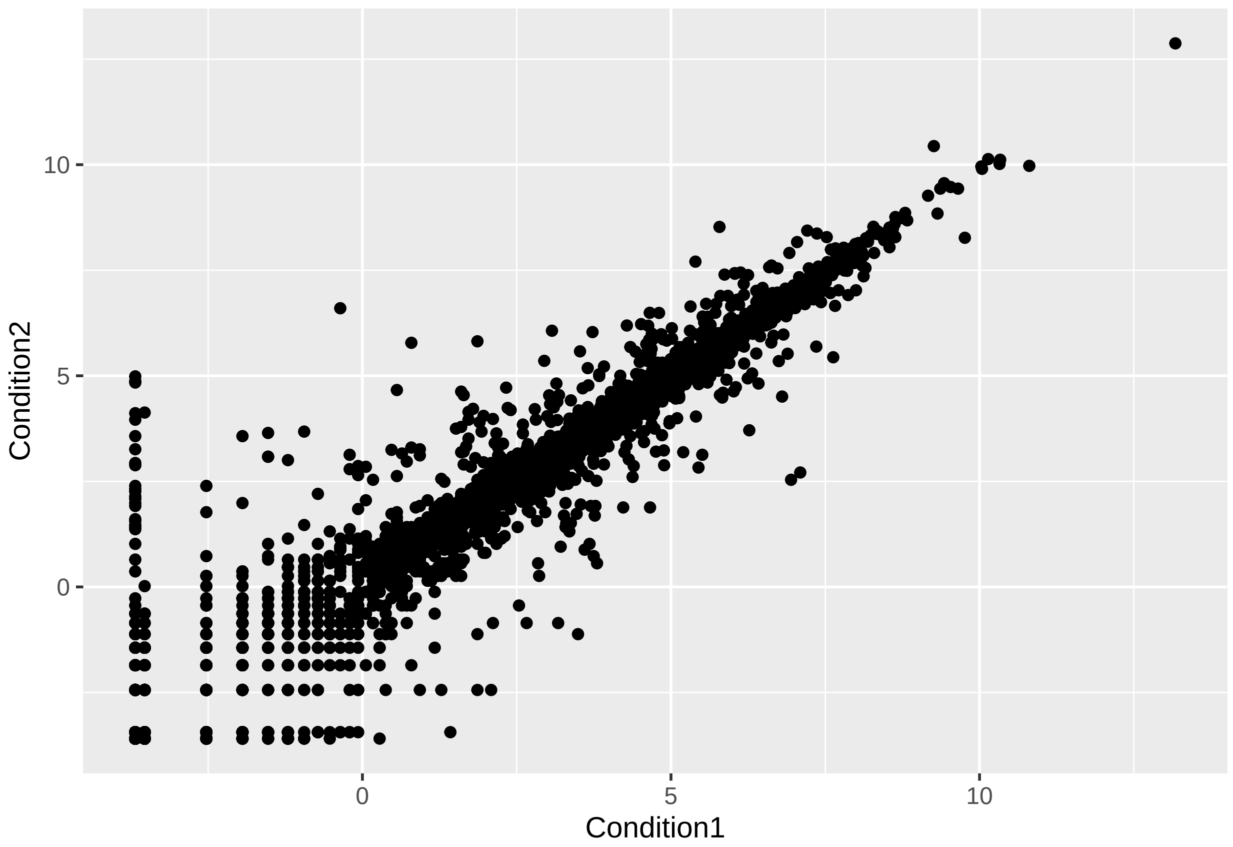
```r
#Q: Round answer
round(table(genes$State) / nrow(genes))
```

```
## 
##       down unchanging         up 
##          0          1          0
```

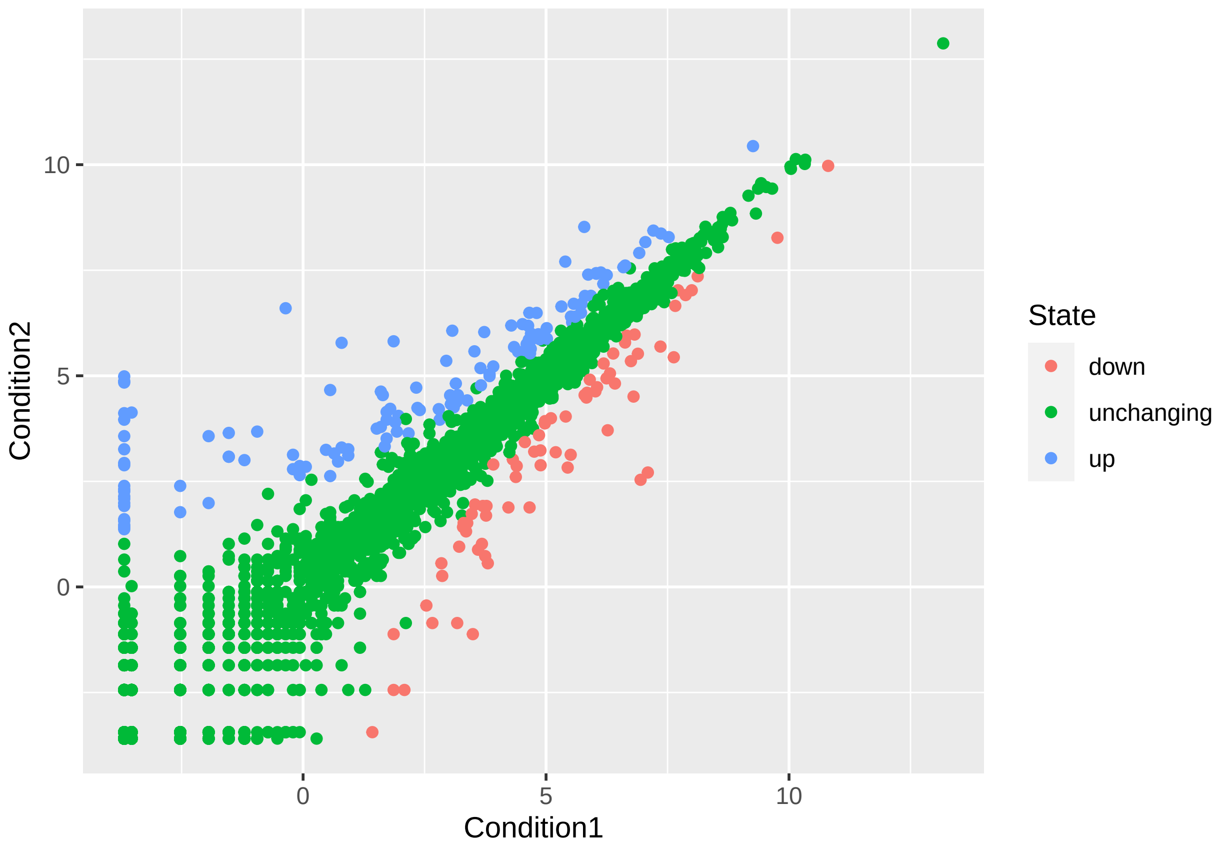
```r
#Q: Convert to percentage and round answer to 2 sig figs
round(table(genes$State) / nrow(genes)*100,2)
```

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

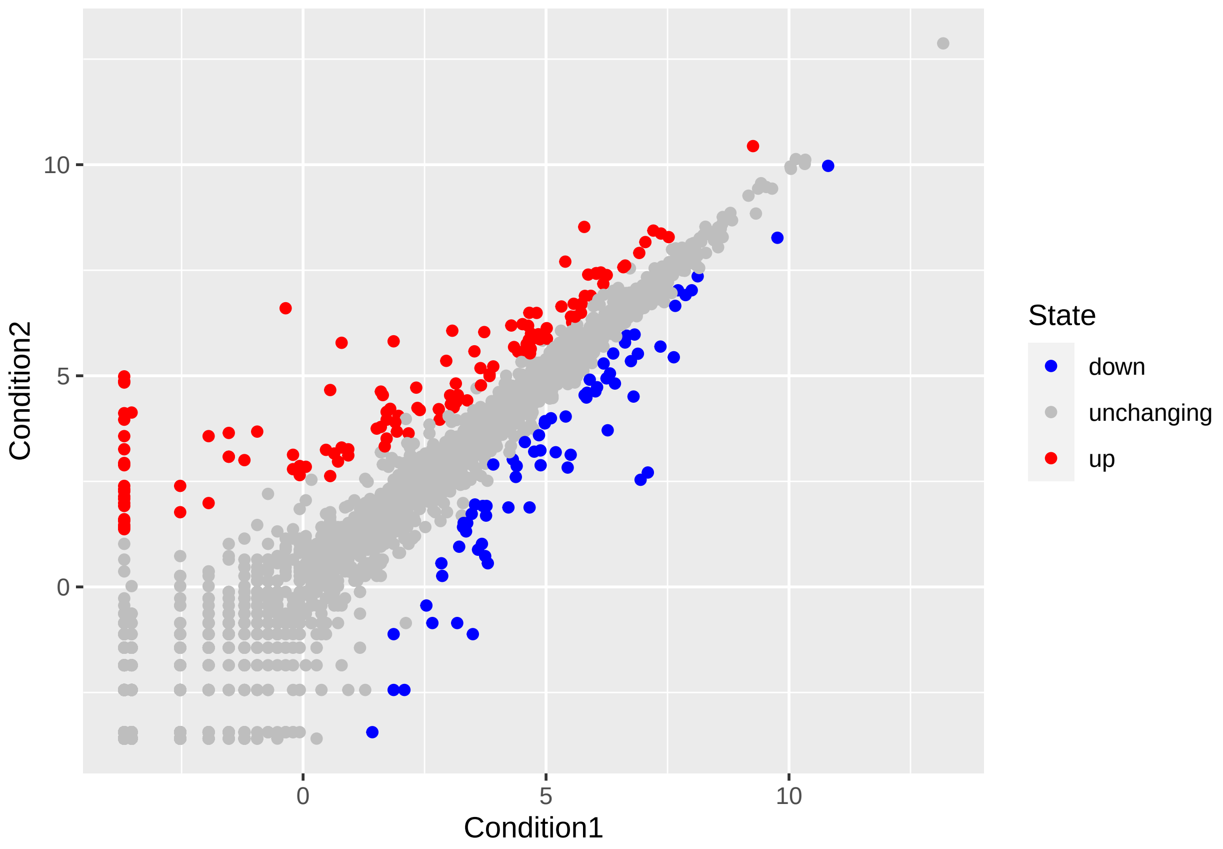
```r
#Q5 last pt: Complete the code
library(ggplot2)
ggplot(genes) +
  aes(x=Condition1, y=Condition2) +
  geom_point()
```



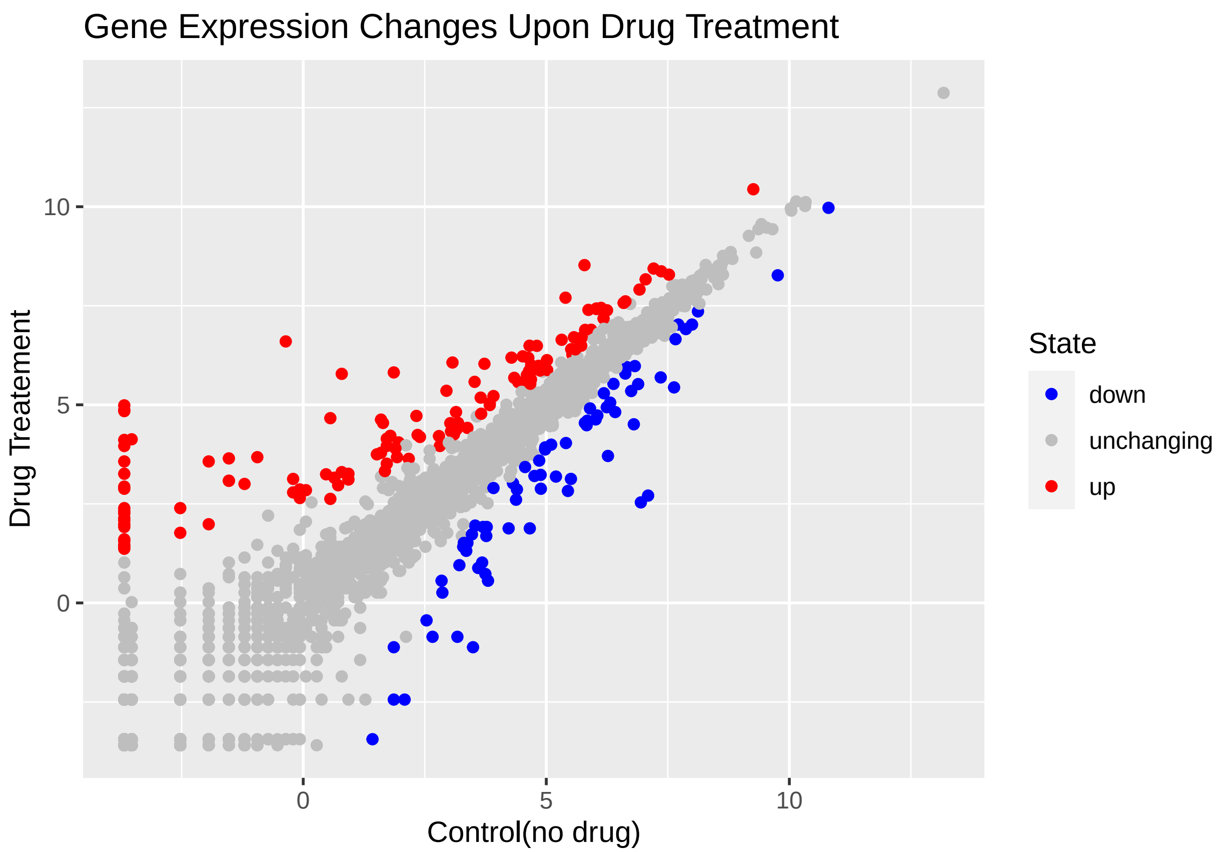
```
# Color code plot by State
p <- ggplot(genes) +
  aes(x=Condition1, y=Condition2, col=State) +
  geom_point()
p
```



```
#Change default colors
p + scale_colour_manual(values=c("blue","gray","red"))
```



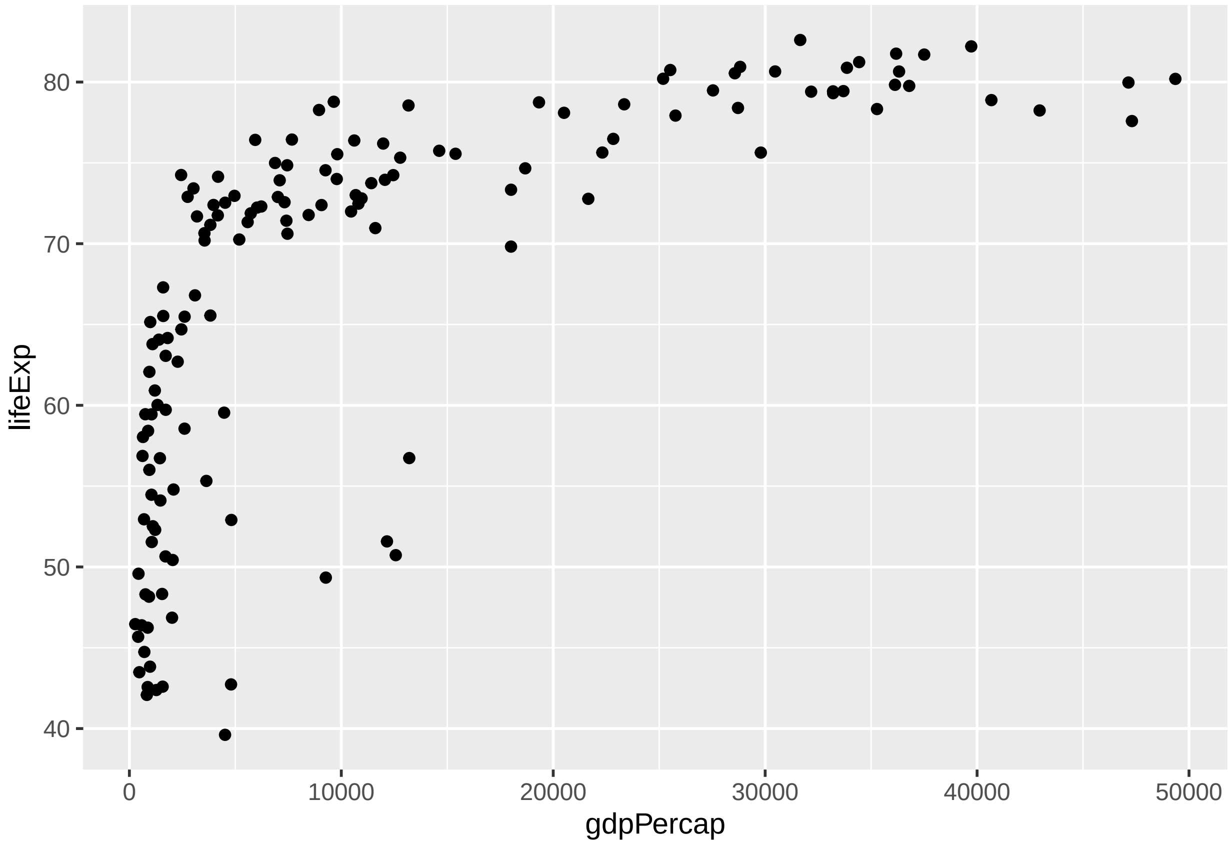
```
#Q: Add some plot annotations to the p object with the labs() function
p + scale_colour_manual(values=c("blue","gray","red")) +
  labs(title="Gene Expression Changes Upon Drug Treatment",
       x="Control(no drug)",
       y="Drug Treatement")
```



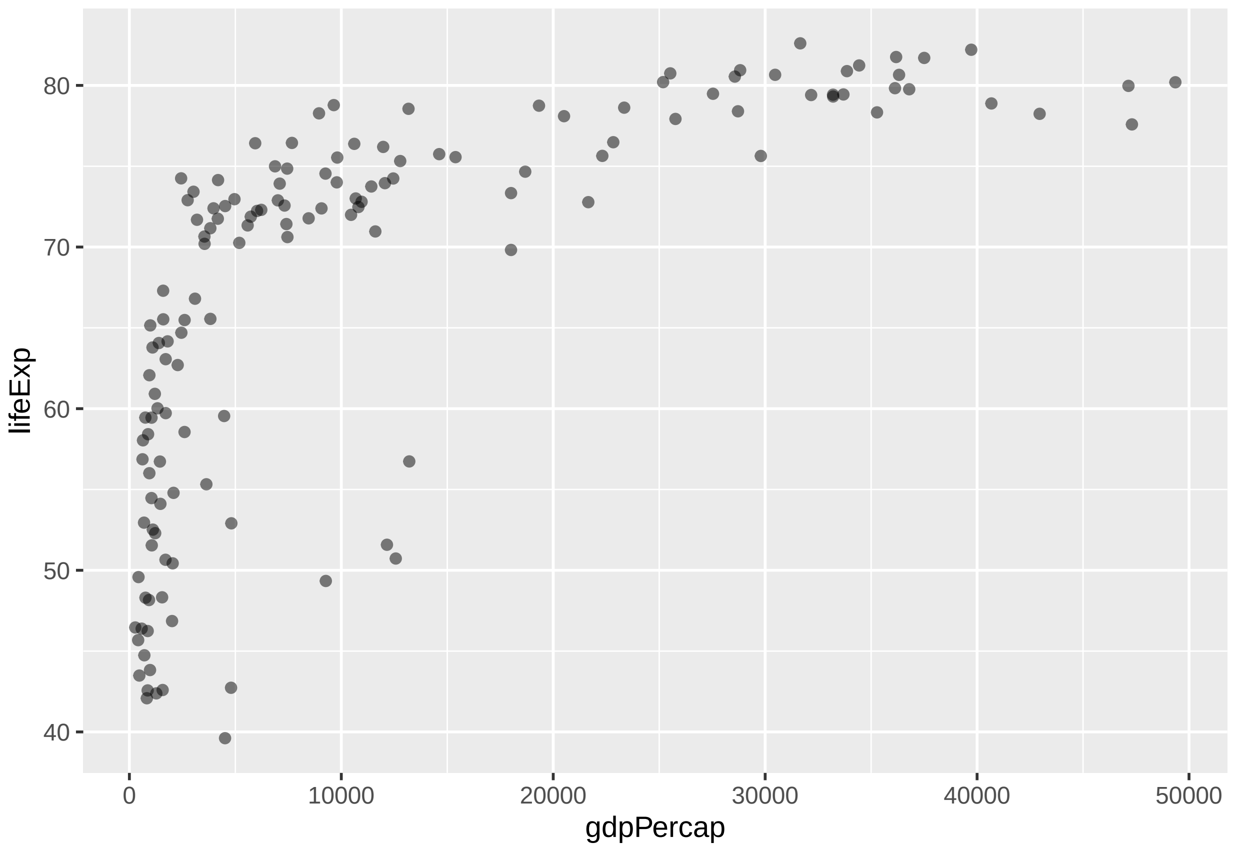
```
#Q: OPTIONAL: Going Further
#Visualizing the gapminder dataset
#install.packages("gapminder")
library(gapminder)

#install dplyr package to focus in on a single year in gapminder dataset
#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
```

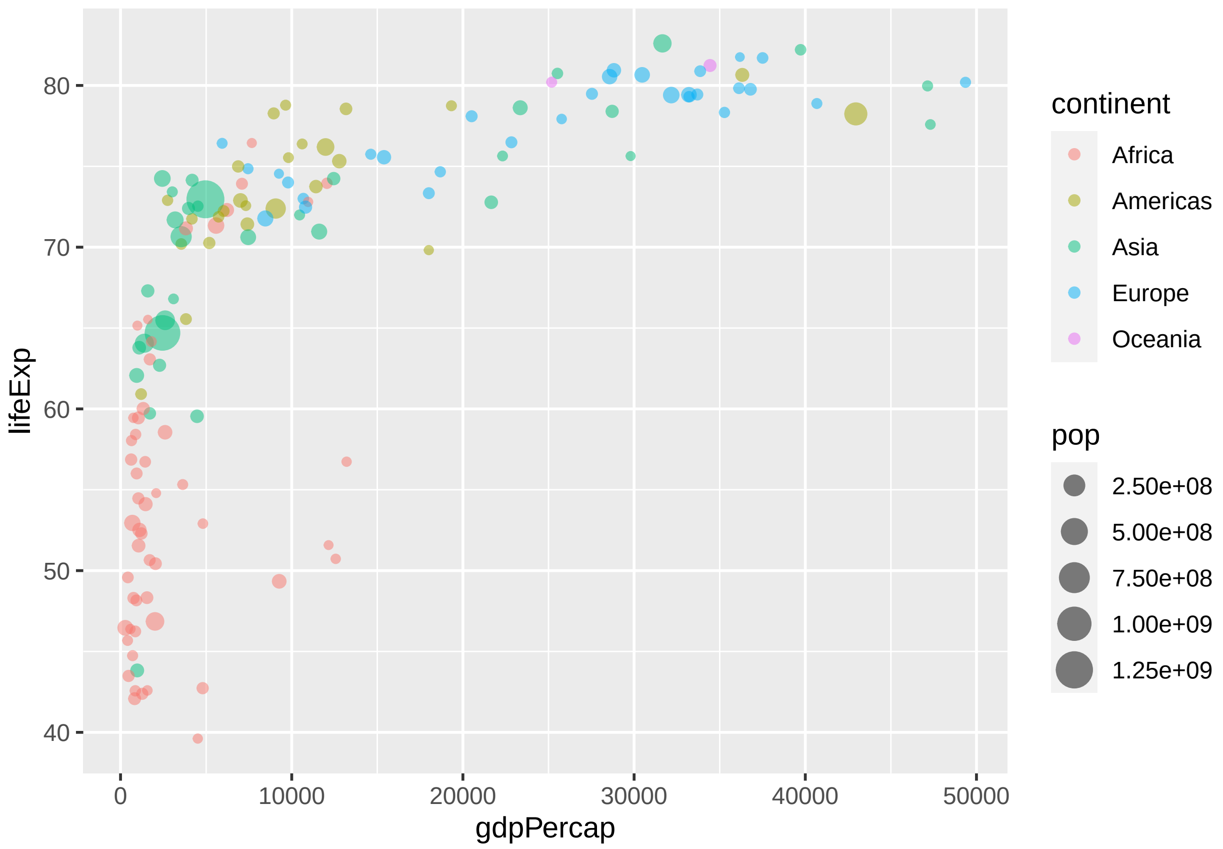
```
gapminder_2007 <-gapminder %>% filter(year==2007)
#Q: Produce first basica scatter plot of gapminder_2007 dataset
library(ggplot2)
ggplot(gapminder_2007) +
  aes(x=gdpPercap, y=lifeExp) +
  geom_point()
```



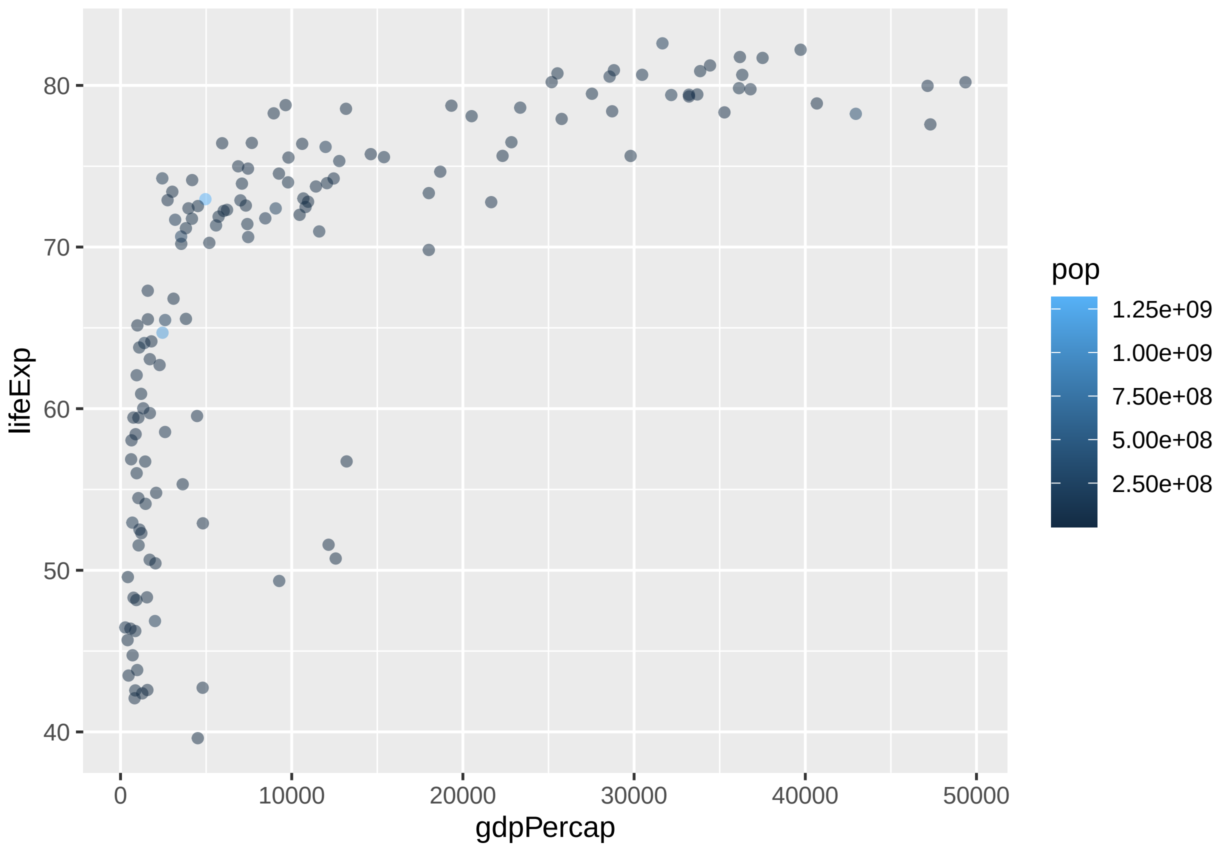
```
#Add alpha argument to make points slightly transparent
ggplot(gapminder_2007) +
  aes(x=gdpPercap, y=lifeExp) +
  geom_point(alpha=0.5)
```



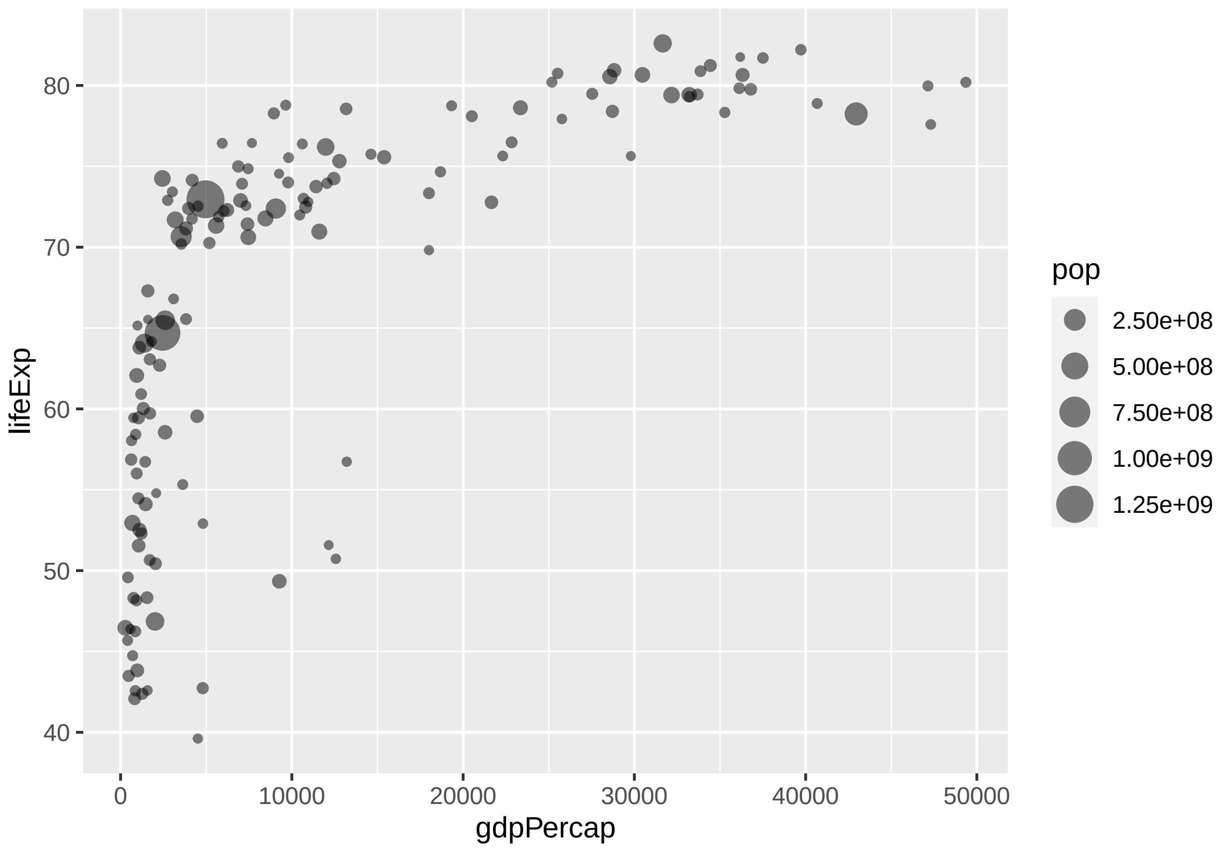
```
#Adding more variables to aes()
#Coloring scheme based on the categorical data type of the variable continent
ggplot(gapminder_2007)+
  aes(x=gdpPercap, y=lifeExp, color=continent, size=pop)+
  geom_point(alpha=0.5)
```



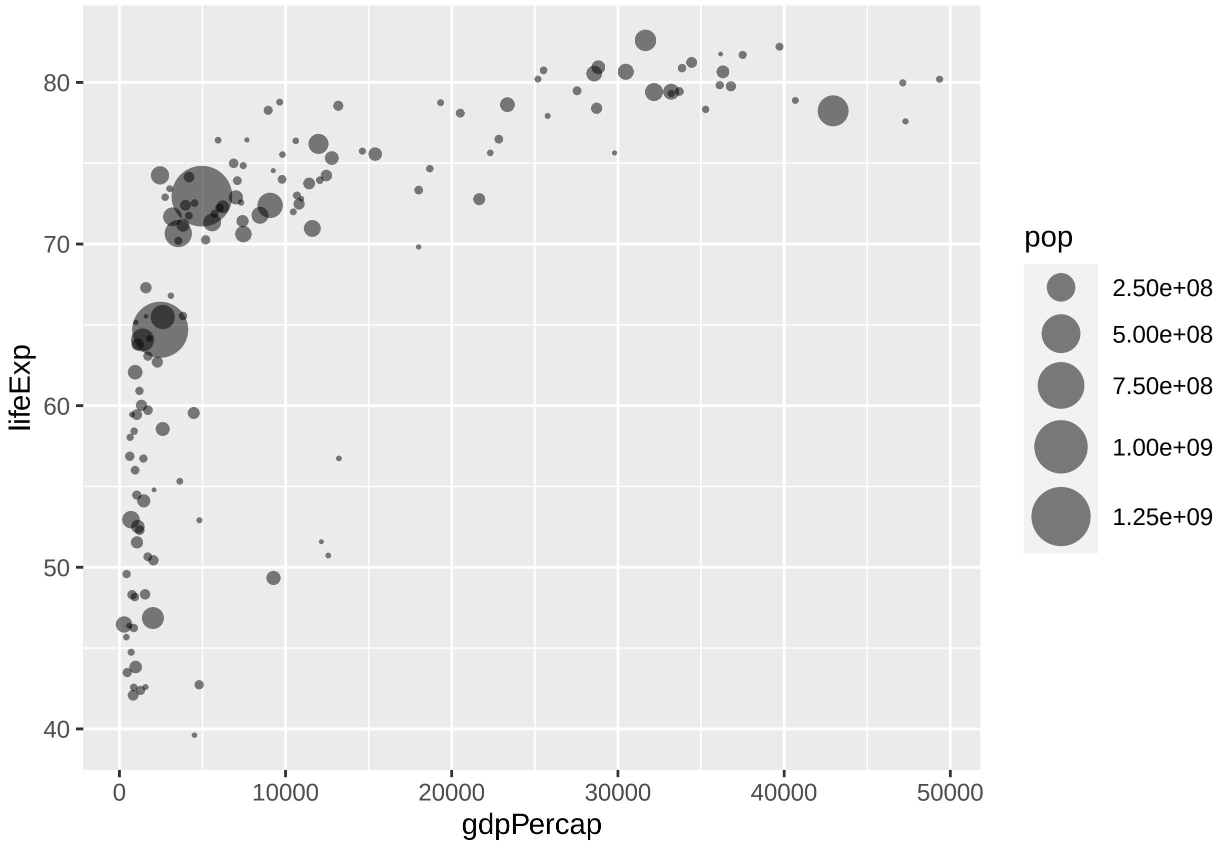
```
#Coloring scheme based on the numeric variable population pop
ggplot(gapminder_2007)+
  aes(x=gdpPercap, y=lifeExp, color=pop)+
  geom_point(alpha=0.5)
```



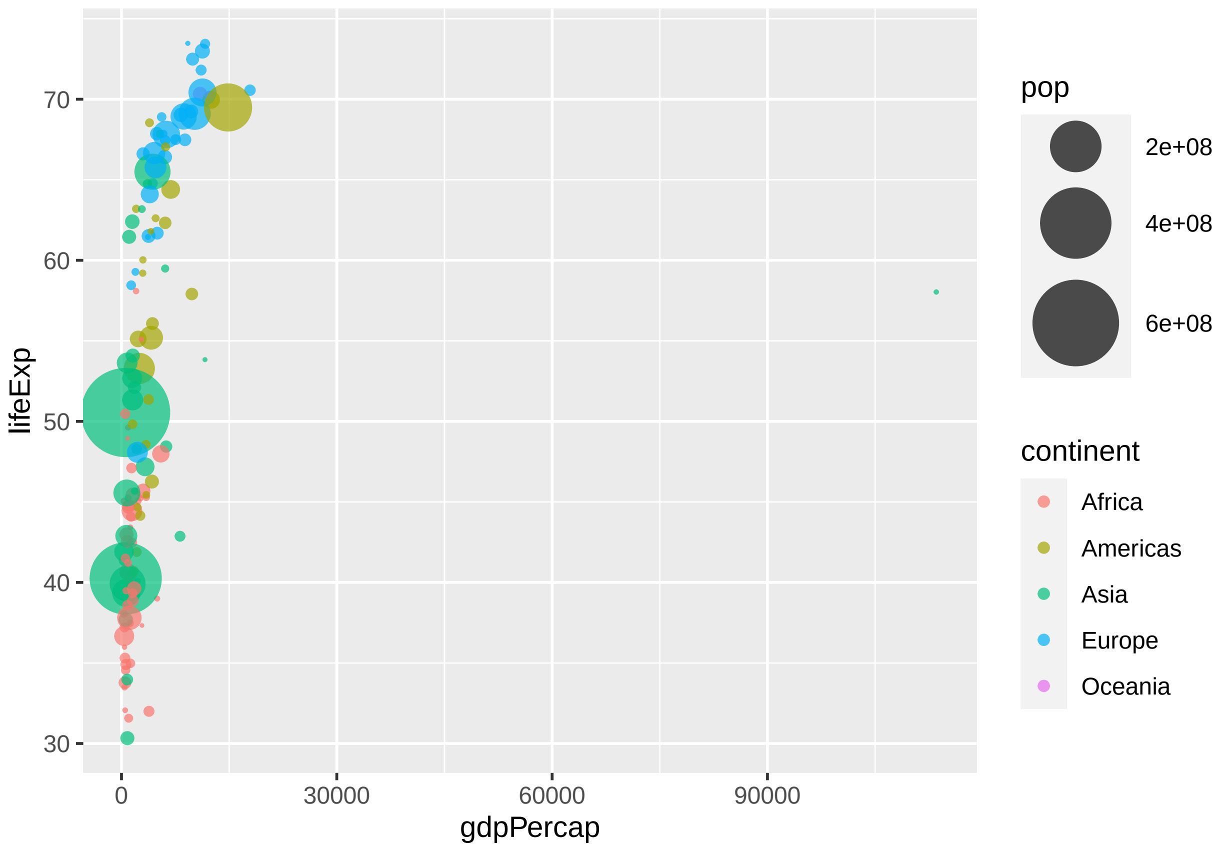
```
#Adjusting point size -- binned by default
ggplot(gapminder_2007)+
  aes(x=gdpPercap, y=lifeExp, size=pop)+
  geom_point(alpha=0.5)
```



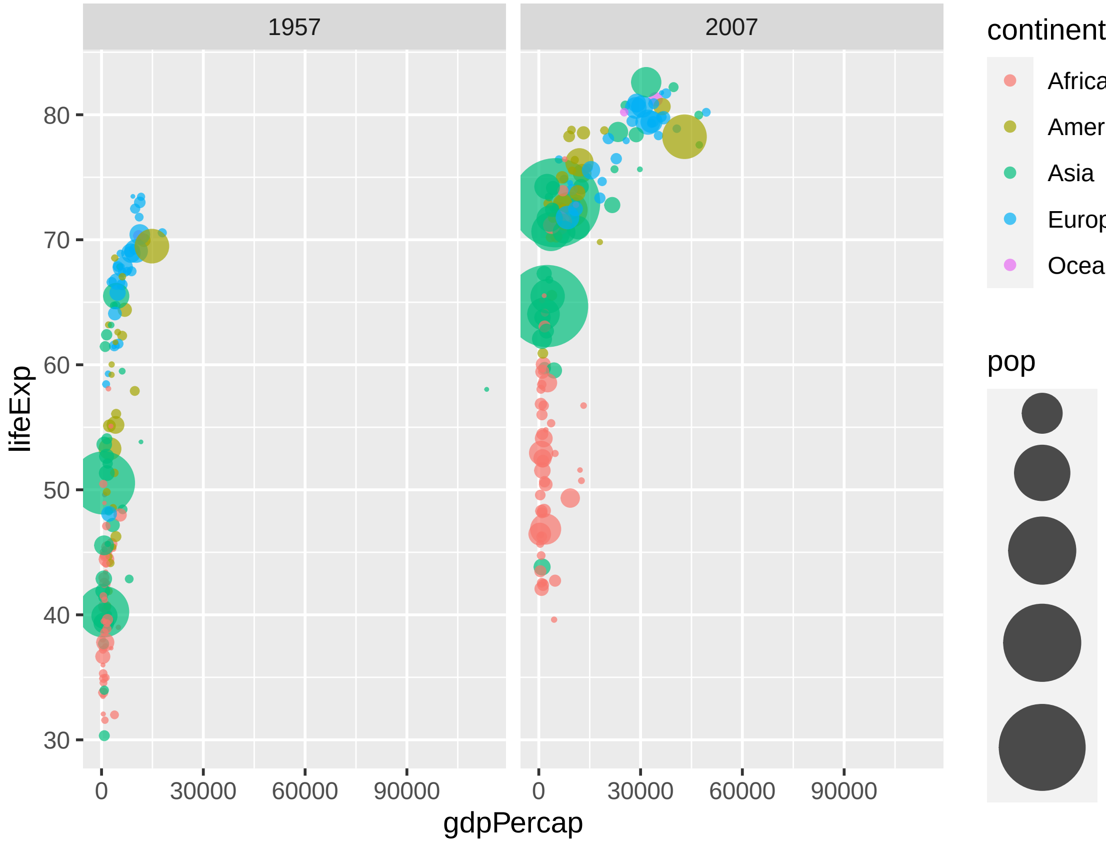
```
#Adjusting point size -- to reflect actual population differences, we adjust the point's max_size, resu
ggplot(gapminder_2007)+
  geom_point(aes(x=gdpPercap, y=lifeExp, size=pop), alpha=0.5)+
  scale_size_area(max_size=10)
```



```
#Producing gapminder scatter plot for the year 1957
library(dplyr)
gapminder_1957 <-gapminder %>% filter(year==1957)
library(ggplot2)
ggplot(gapminder_1957)+
  geom_point(aes(x=gdpPercap, y=lifeExp, color=continent,size=pop), alpha=0.7)+
  scale_size_area(max_size=15)
```



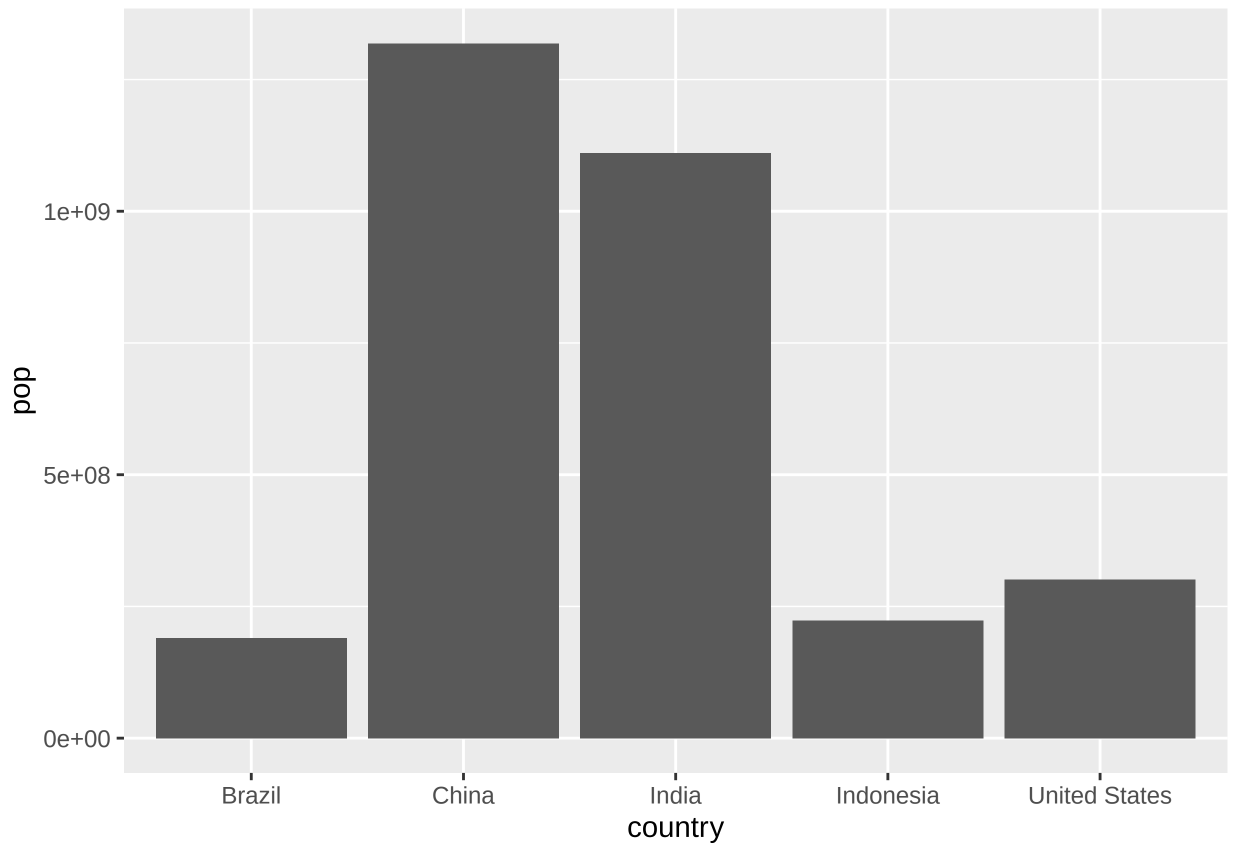
```
#Q: Visualizing years 1957 + 2007 together
library(dplyr)
gapminder_both <-gapminder %>% filter(year==1957 | year==2007)
library(ggplot2)
ggplot(gapminder_both)+
  geom_point(aes(x=gdpPercap, y=lifeExp, color=continent,size=pop), alpha=0.7)+
  scale_size_area(max_size=15)+
  facet_wrap(~year)
```



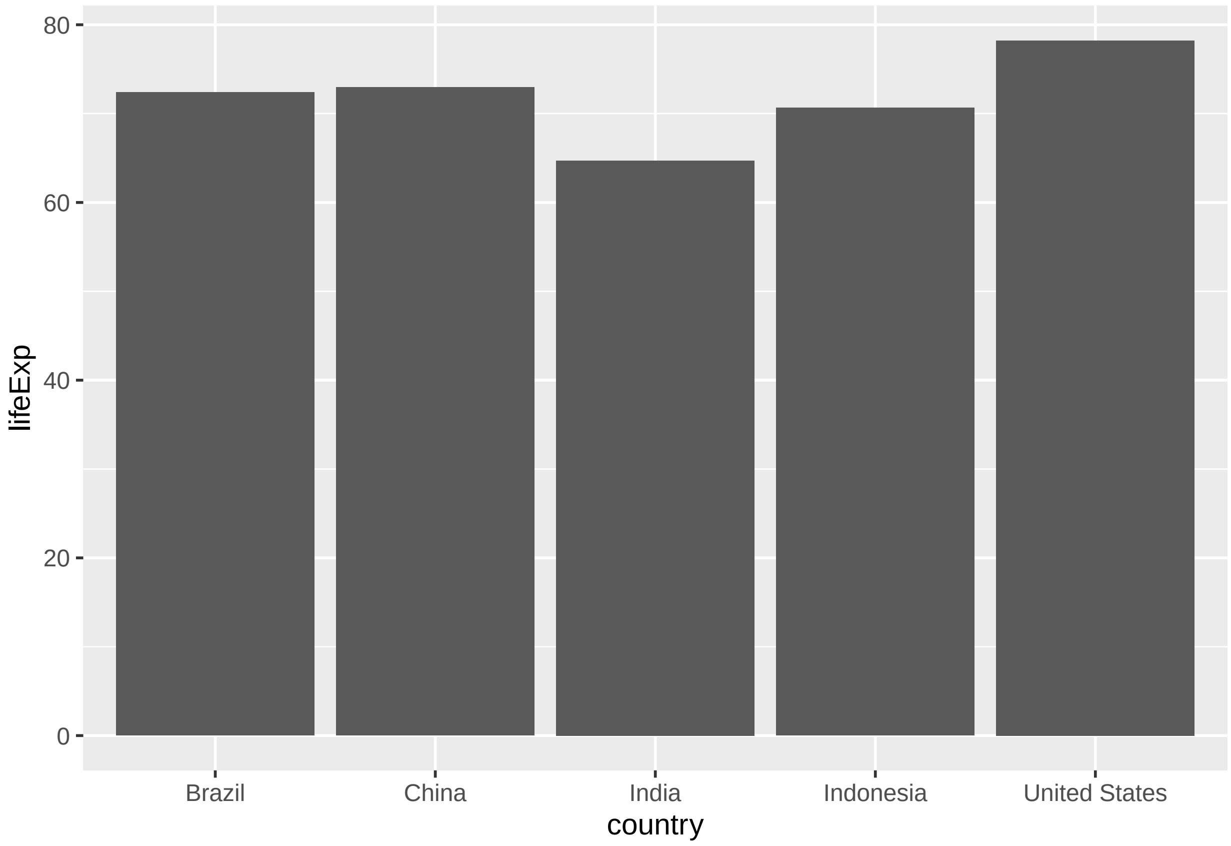
```
#7.OPTIONAL: Bar Charts
#Intro
library(dplyr)
gapminder_top5 <- gapminder %>%
  filter(year==2007) %>%
  arrange(desc(pop))%>%
  top_n(5,pop)
gapminder_top5
```

```
## # 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.
```

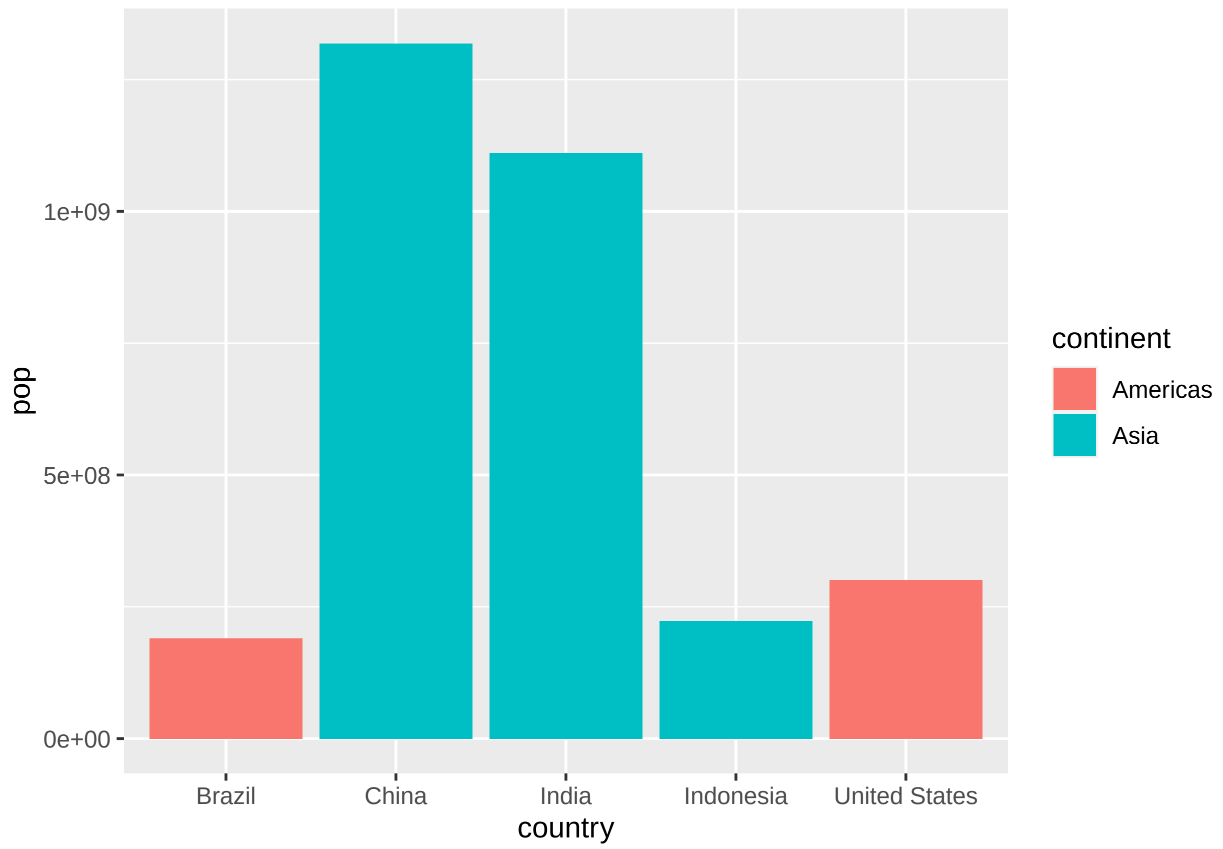
```
#Creating a simple bar chart
library(ggplot2)
ggplot(gapminder_top5)+
  geom_col(aes(x=country,y=pop))
```



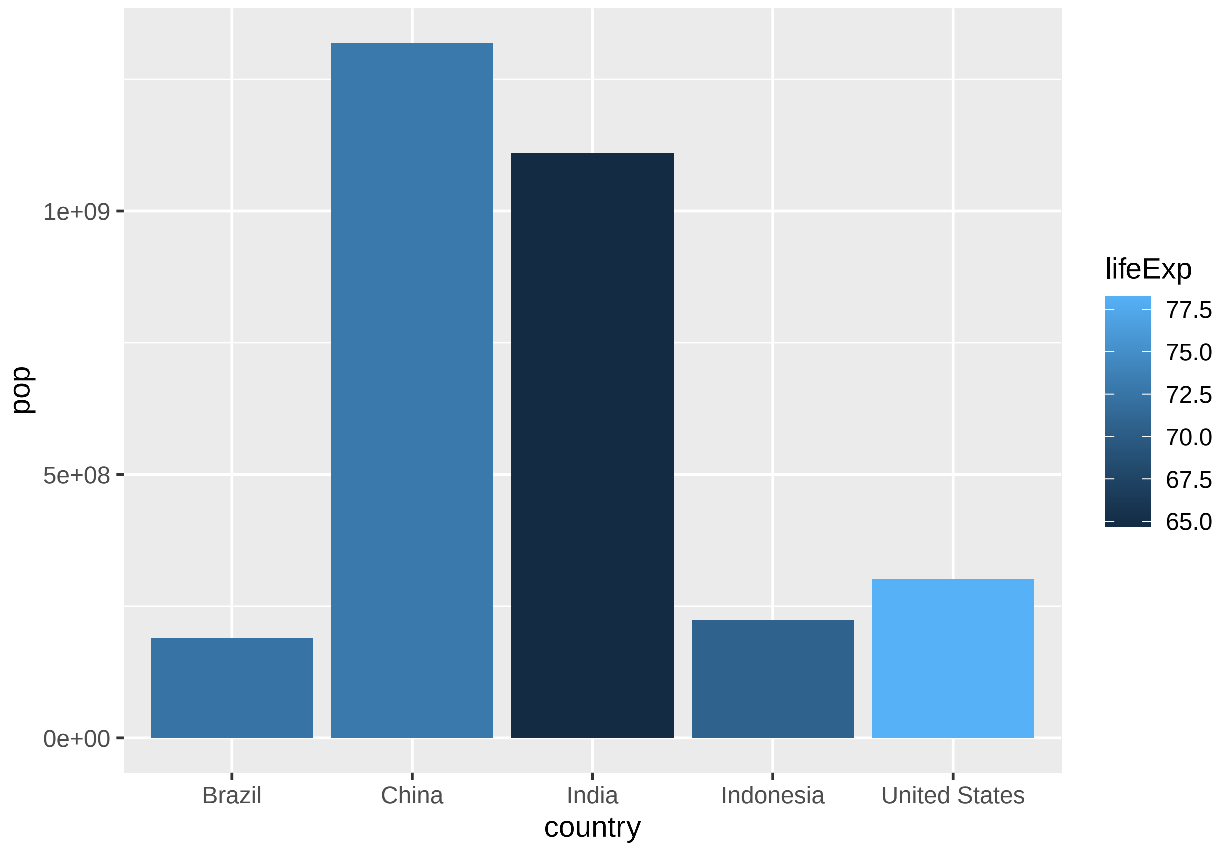
```
#Plot life expectancy by country
ggplot(gapminder_top5)+
  geom_col(aes(x=country, y=lifeExp))
```



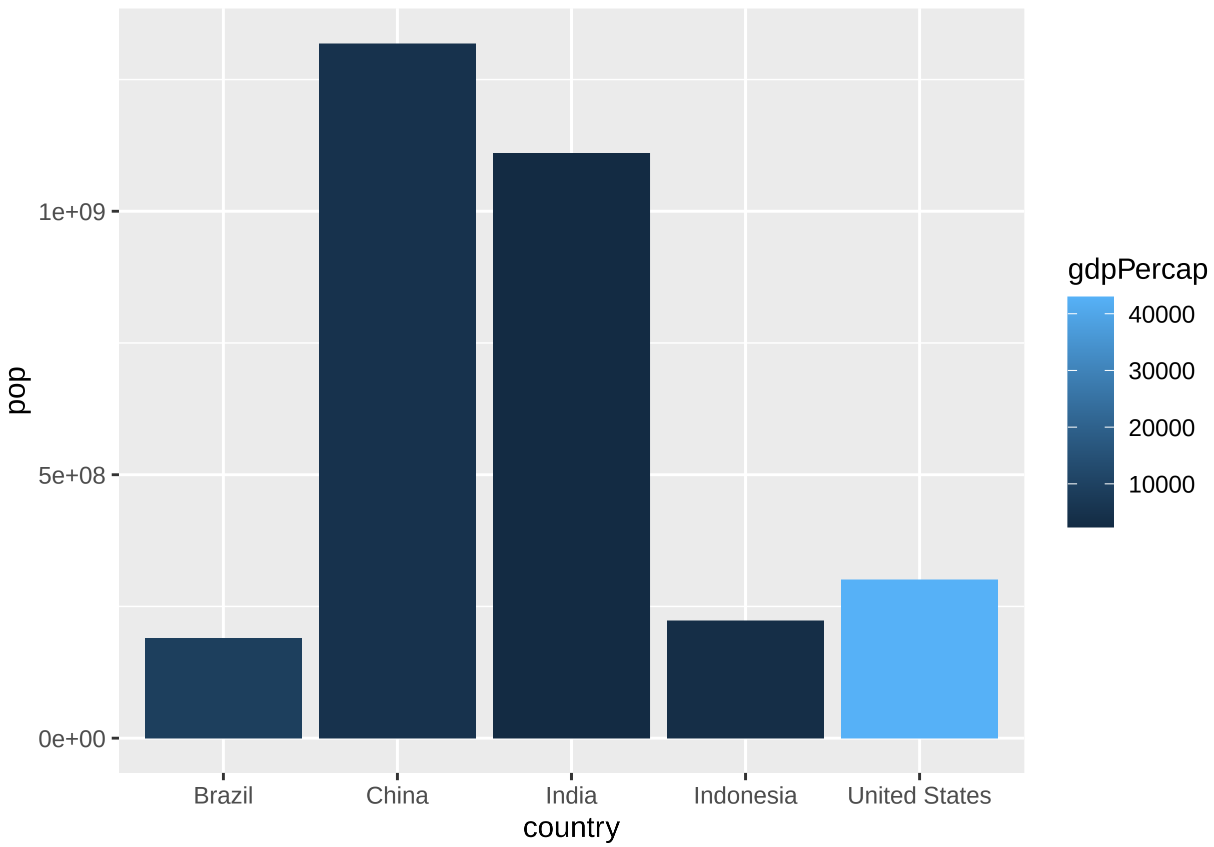
```
#Filling bars with color --
#with categorical variable ("continent") == clear color scheme
ggplot(gapminder_top5)+
  geom_col(aes(x=country,y=pop,fill=continent))
```



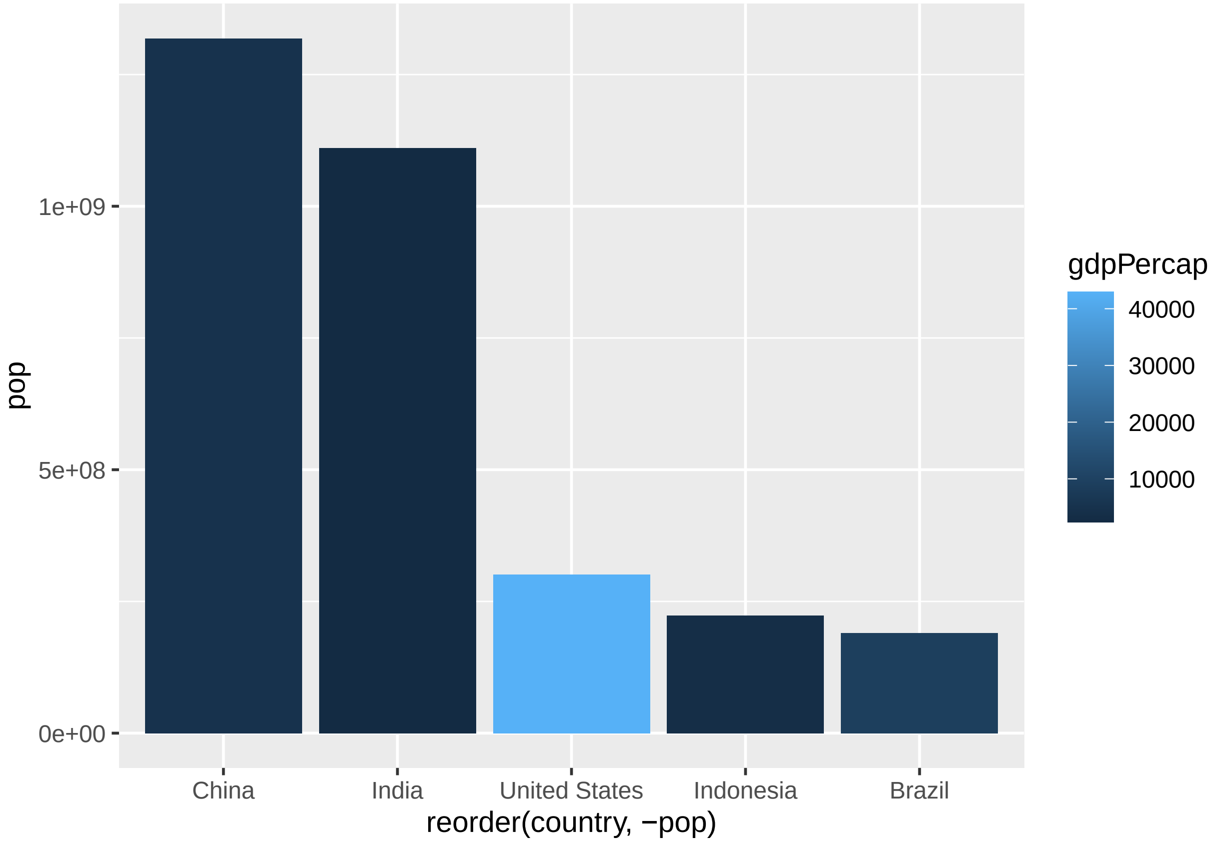
```
#Filling bars with color --
#with numberic variable (lifeExp) == continuous color scheme
#Numeric variables can be used to fill bars!
ggplot(gapminder_top5)+
  geom_col(aes(x=country,y=pop,fill=lifeExp))
```



```
#Plot population size by country
ggplot(gapminder_top5)+
  geom_col(aes(x=country, y=pop,fill=gdpPercap))
```

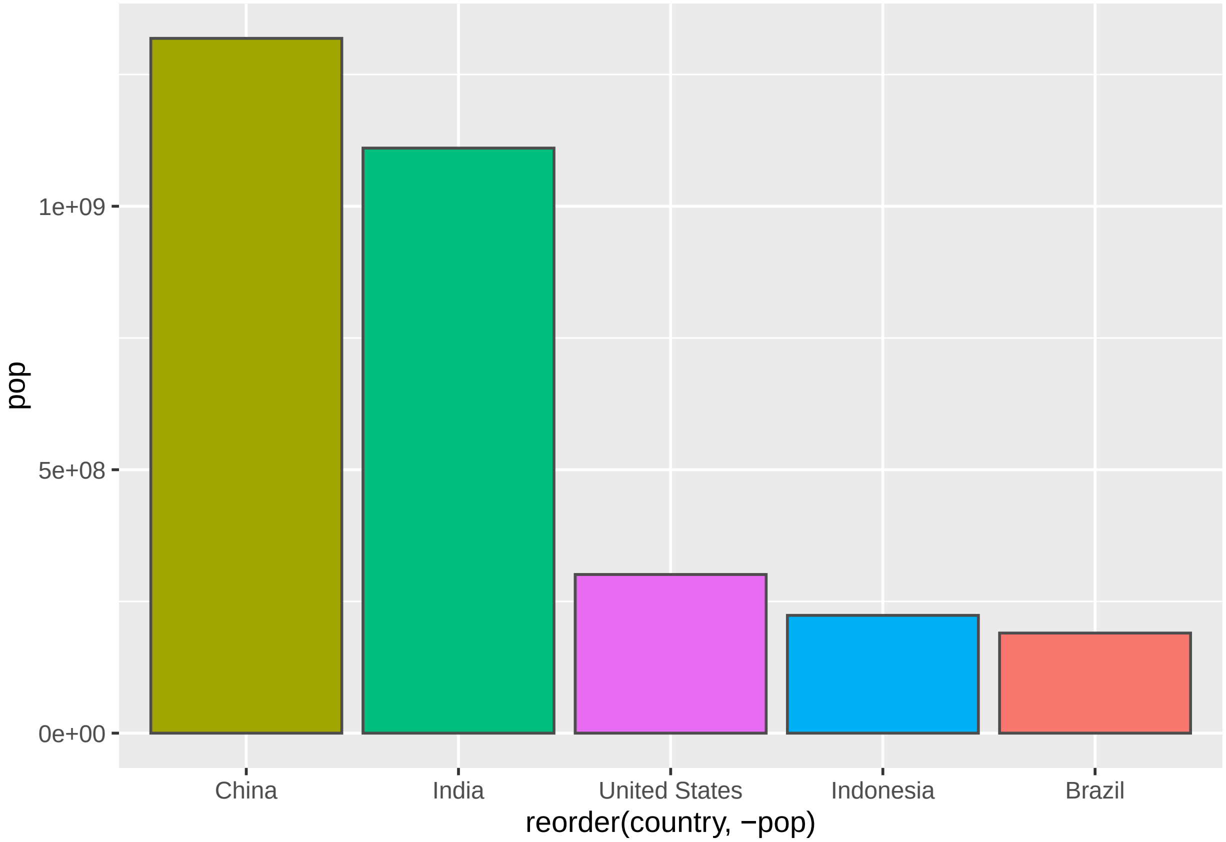


```
#Changing the order of the bars
ggplot(gapminder_top5)+
  aes(x=reorder(country,-pop),y=pop,fill=gdpPercap)+
  geom_col()
```



```
#Fill by country
ggplot(gapminder_top5)+
  aes(x=reorder(country,-pop),y=pop,fill=country)+
  geom_col(col="gray30")+
  guides(fill=FALSE)
```

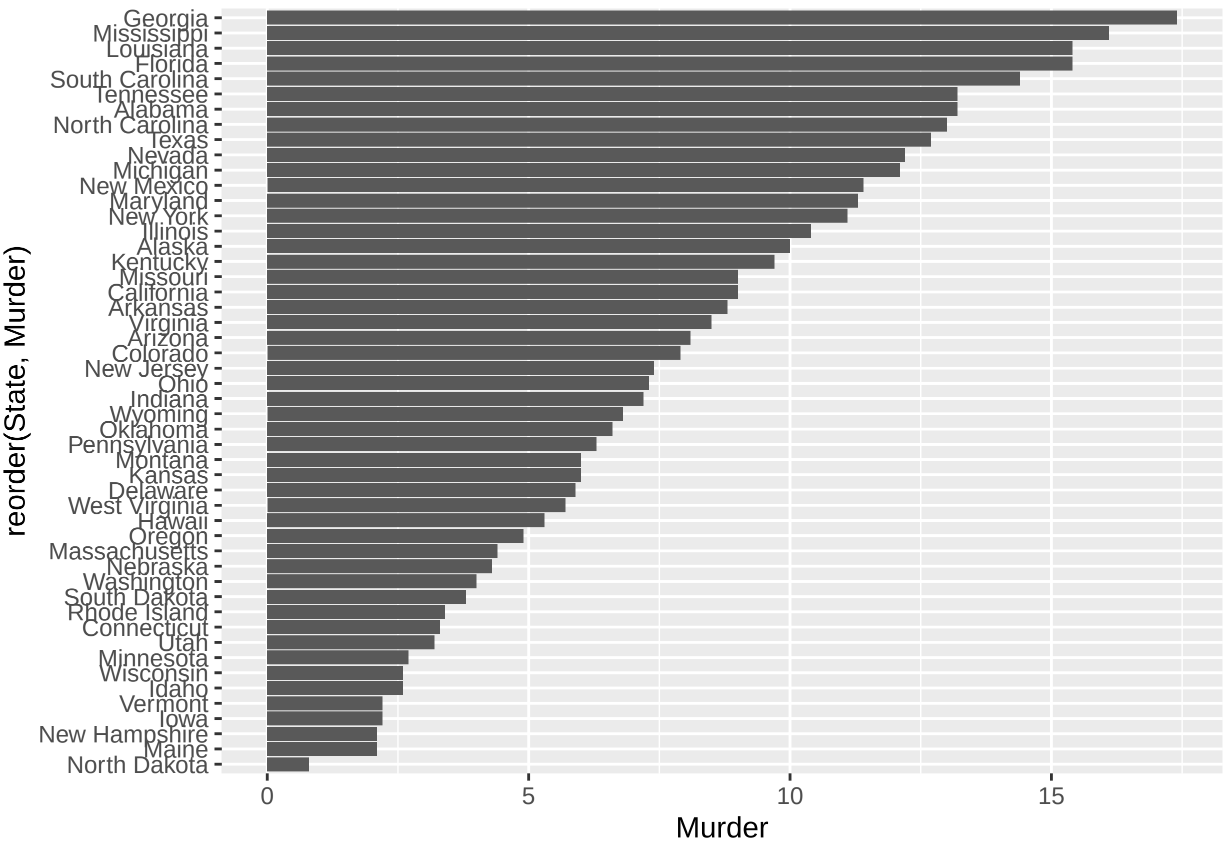
```
## Warning: ‘guides(<scale> = FALSE)‘ is deprecated. Please use ‘guides(<scale> =
## "none")‘ instead.
```



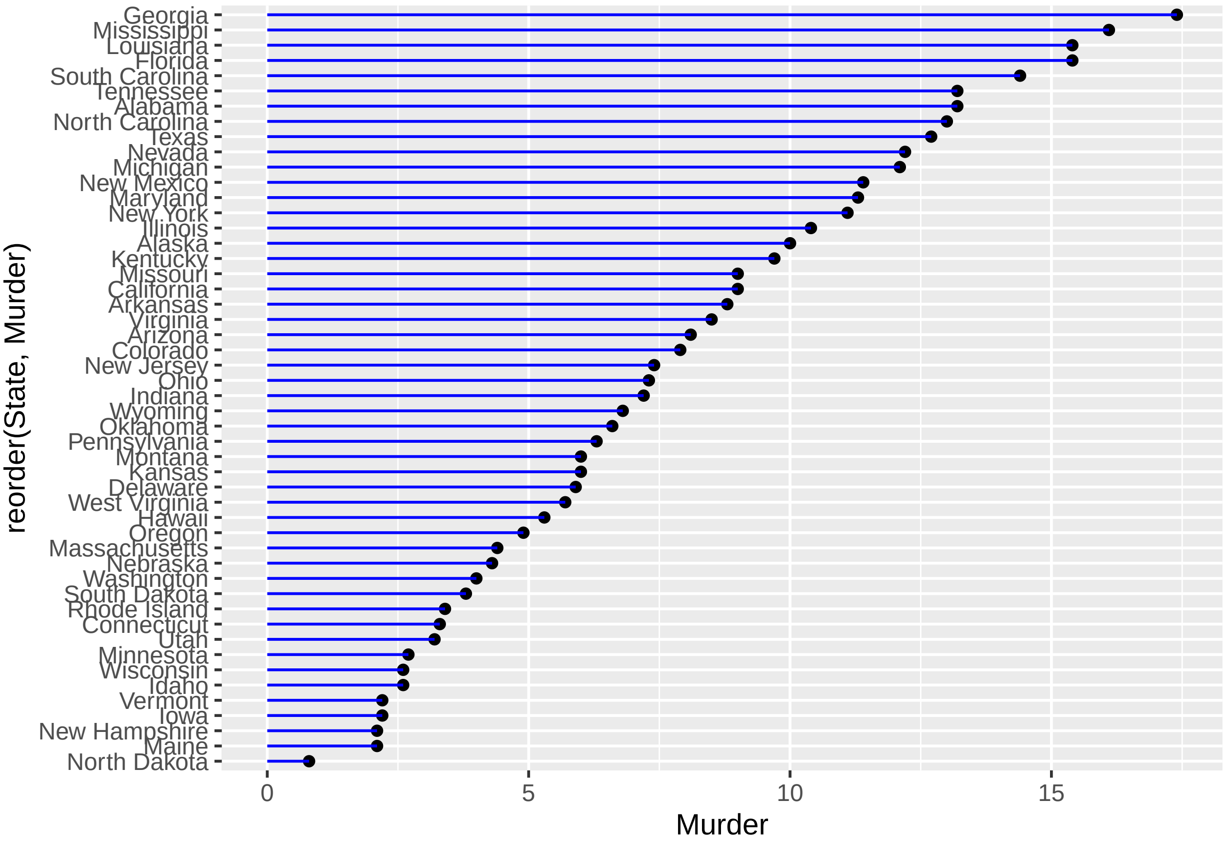
```
#Flipping bar charts
head(USArrests)
```

```
##            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)
library(ggplot2)
ggplot(USArrests)+
  aes(x=reorder(State,Murder),y=Murder)+
  geom_col()+
  coord_flip()
```



```
#Flipped bar graph too crowded? -> Try alternative custom visualization
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()
```



```r
#8.ADVANCED: Plot Animation
#Install gganimate and gifski
#install.packages("gganimate")
#install.packages("gifski")
library(gapminder)
library(gganimate)
```

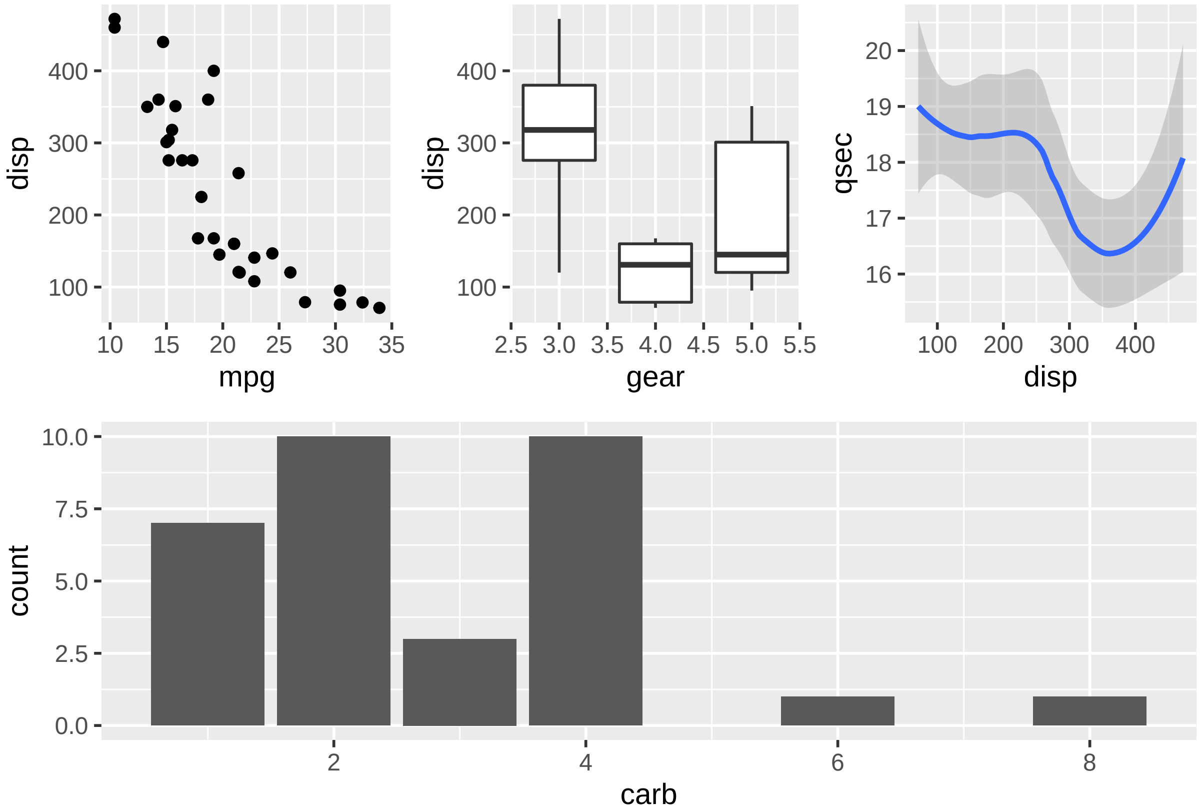
```
## No renderer backend detected. gganimate will default to writing frames to separate files
## Consider installing:
## - the ‘gifski‘ package for gif output
## - the ‘av‘ package for video output
## and restarting the R session
```

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

```
## Warning: No renderer available. Please install the gifski, av, or magick package
## to create animated output
```

```
#9.Combining plots
#Install patchwork
#install.packages("patchwork")
library(patchwork)
library(ggplot2)
#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 p1, p2, p3, p4
(p1|p2|p3)/
    p4
```

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


disp
mpg
disp
gear
qsec
disp
count
carb