

# visualizations\_in\_ggplot.R

kfolelso

2021-10-13

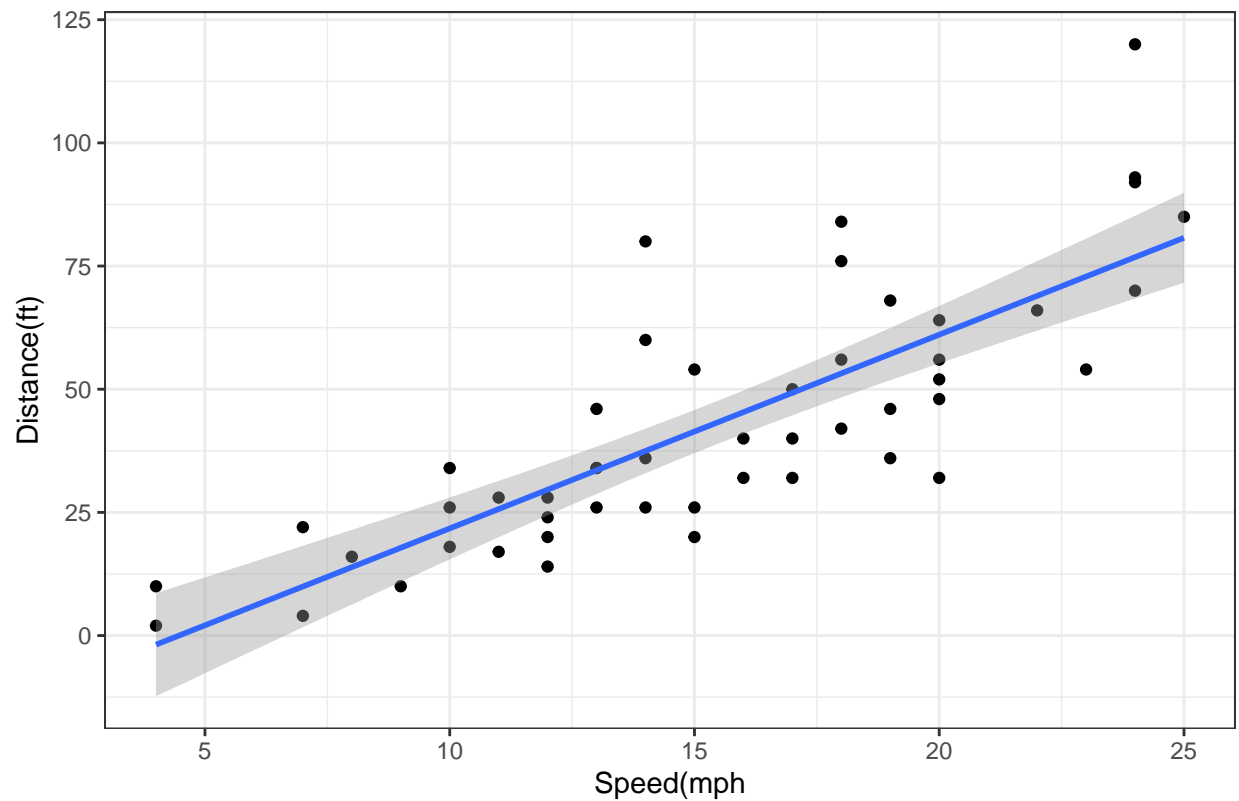
```
# Class 5: 10/13/21, Data visualization in R w/ ggplot2

library(ggplot2)

# will be using built-in ggplot dataset "cars"
# all ggplots have 3 layers:
# data + aes + geoms
ggplot(data=cars) +
  aes(x=speed, y=dist) +
  geom_point() +
  geom_smooth(method="lm")+
  labs(title="The Stopping Distance of Old Cars", x="Speed(mph)", y="Distance(ft)") +
  theme_bw()+
  theme(plot.title = element_text(hjust = 0.5))

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

## The Stopping Distance of Old Cars



```
#mapping additional graph aesthetics
#differential expression analysis w/ practice dataset
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
```

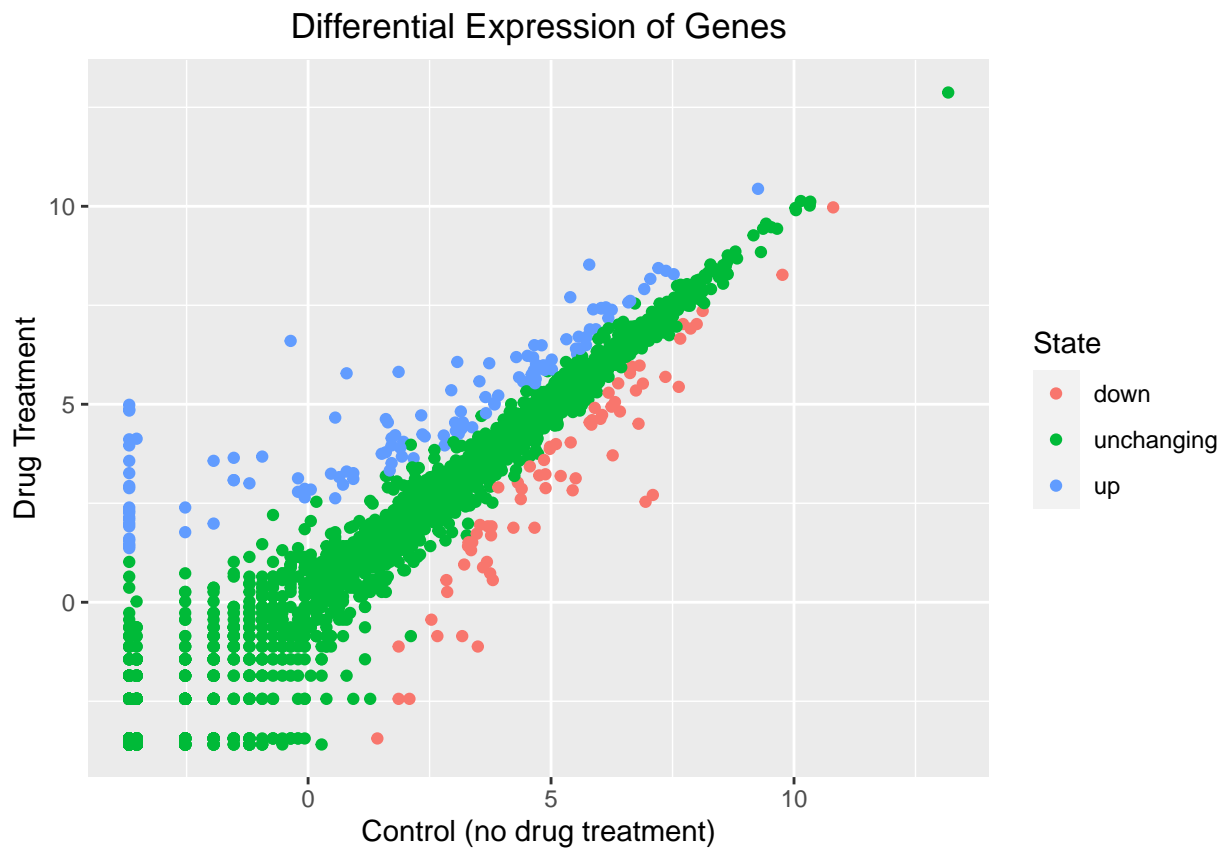
```
table(genes$State)
```

```
##
##      down  unchanged      up
##      72      4997      127
```

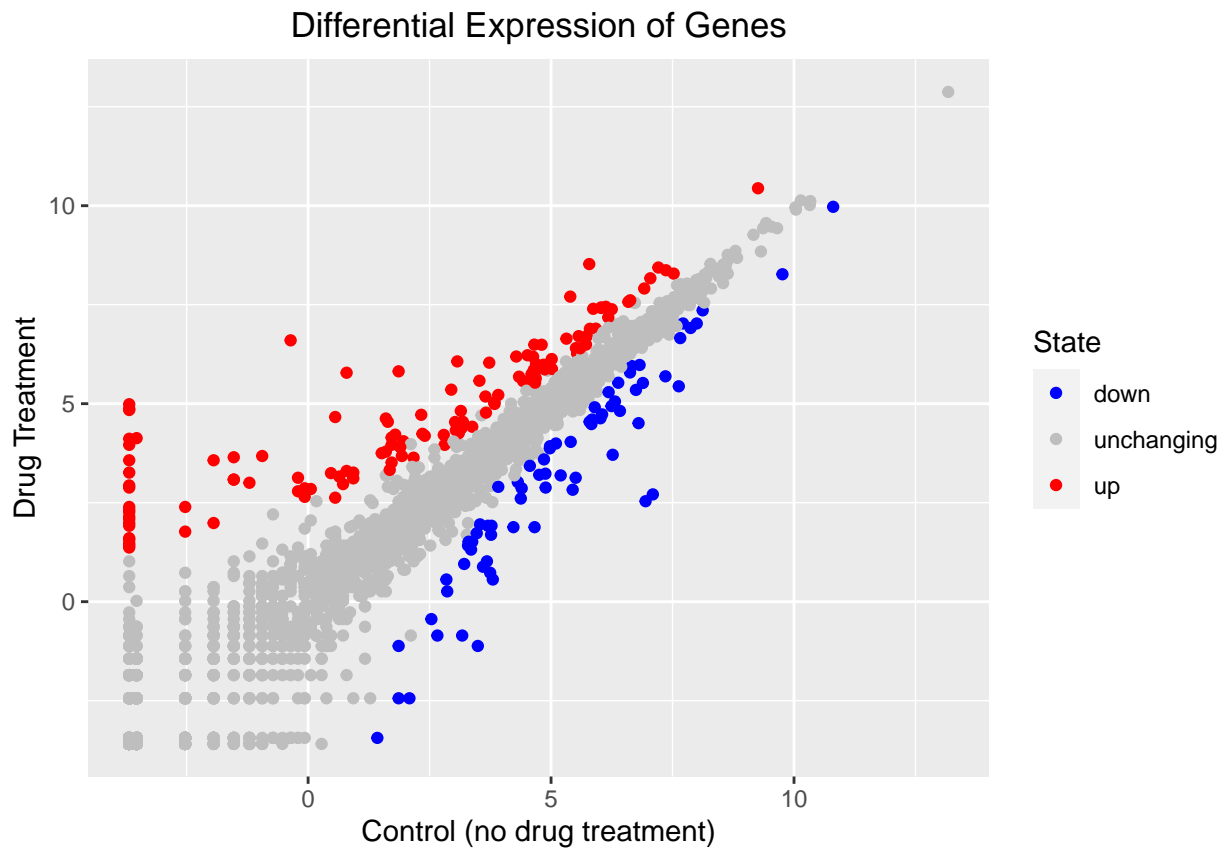
```
#calculate % of genes in each State category
table(genes$State)/nrow(genes) * 100
```

```
##
##      down  unchanged      up
## 1.385681 96.170131  2.444188
```

```
#plot genes df
p <- ggplot(genes) +
  aes(x=Condition1, y=Condition2, col=State)+
  geom_point() +
  labs(title="Differential Expression of Genes", x=
        "Control (no drug treatment)", y= "Drug Treatment") +
  theme(plot.title = element_text(hjust = 0.5))
p
```



```
# Change the colors of points based on State column (up & down-regulated)
p + scale_colour_manual( values=c("blue","gray","red") )
```



```
# Advanced plotting in ggplot2
# Example dataset uses economic & demographic data from various countries since 1952

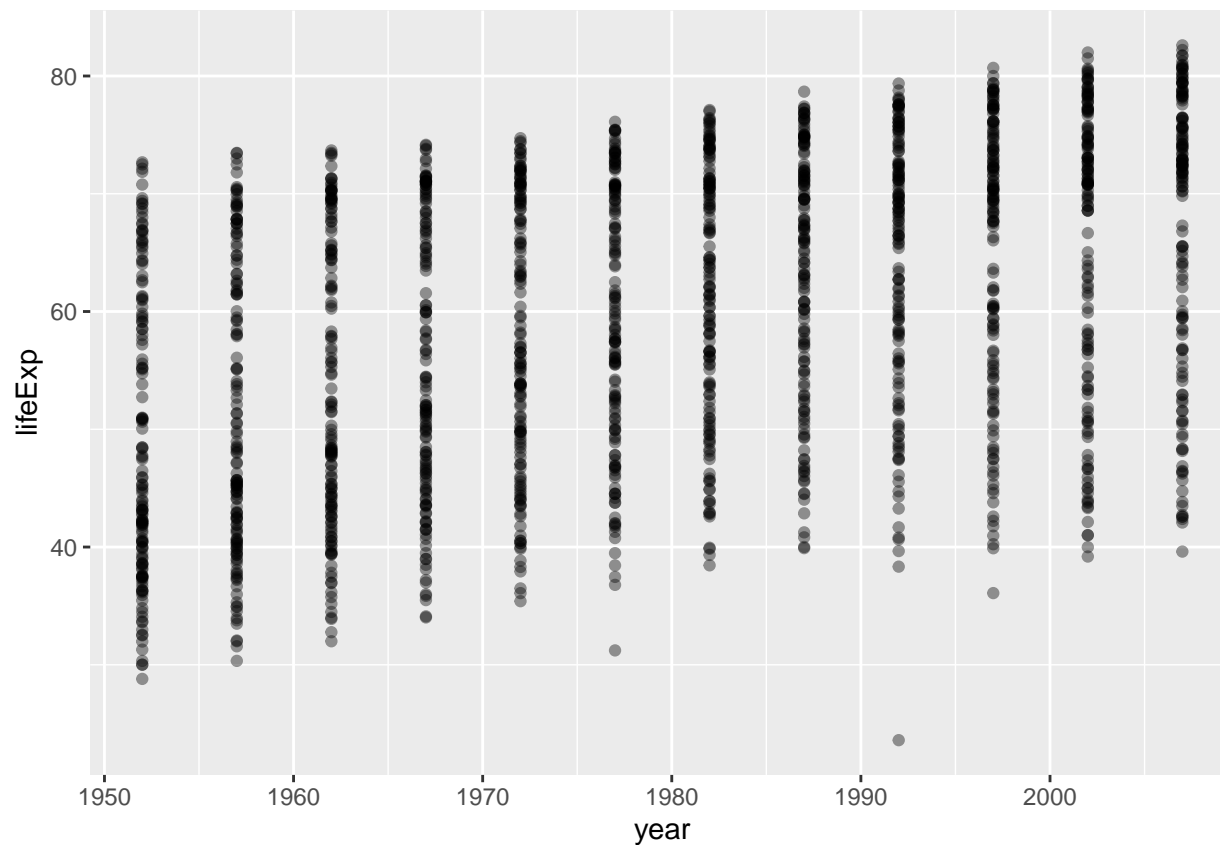
#install.packages("gapminder")
library(gapminder)
library(dplyr)
```

```
##
## 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
```

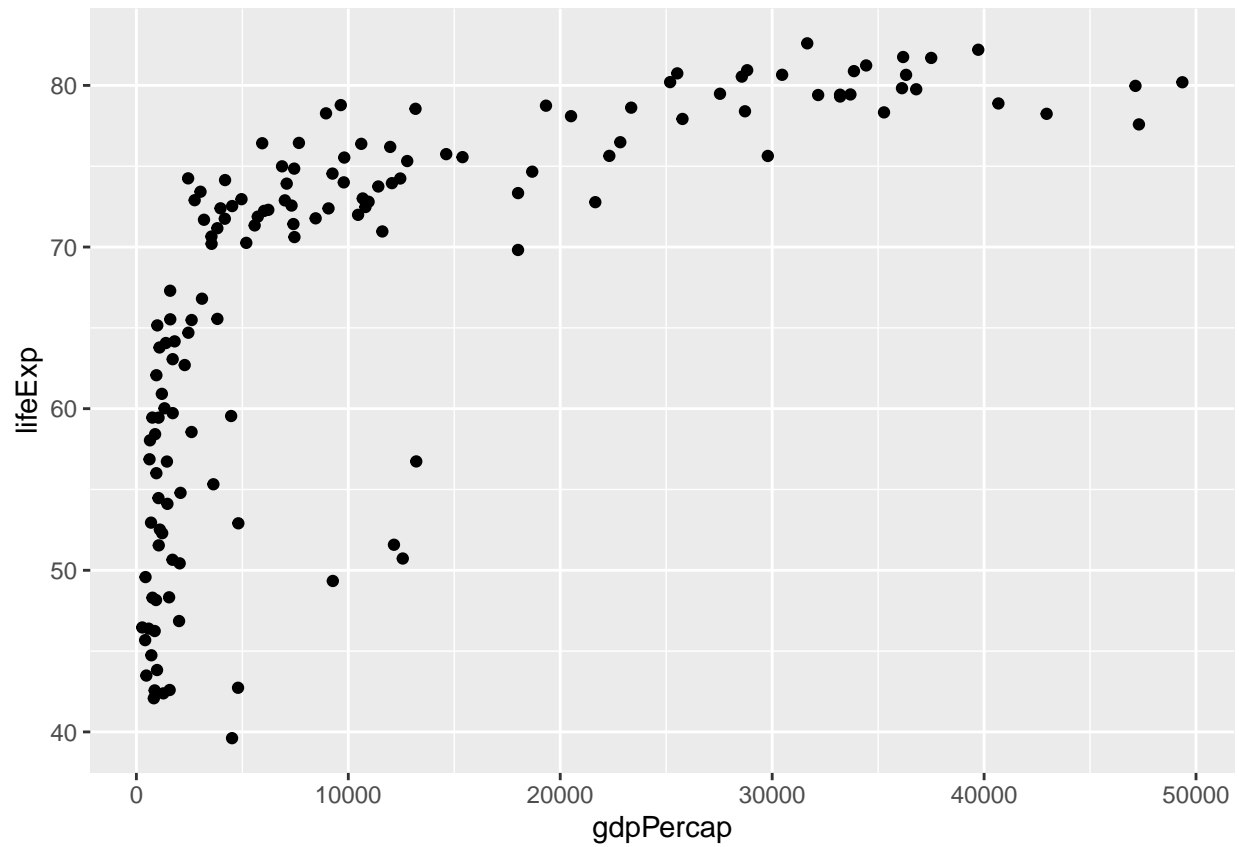
```
# Plot year vs Life expectancy
ggplot(gapminder) +
  aes(x=year, y=lifeExp) +
  geom_point(alpha=0.4) #decrease opacity of points
```



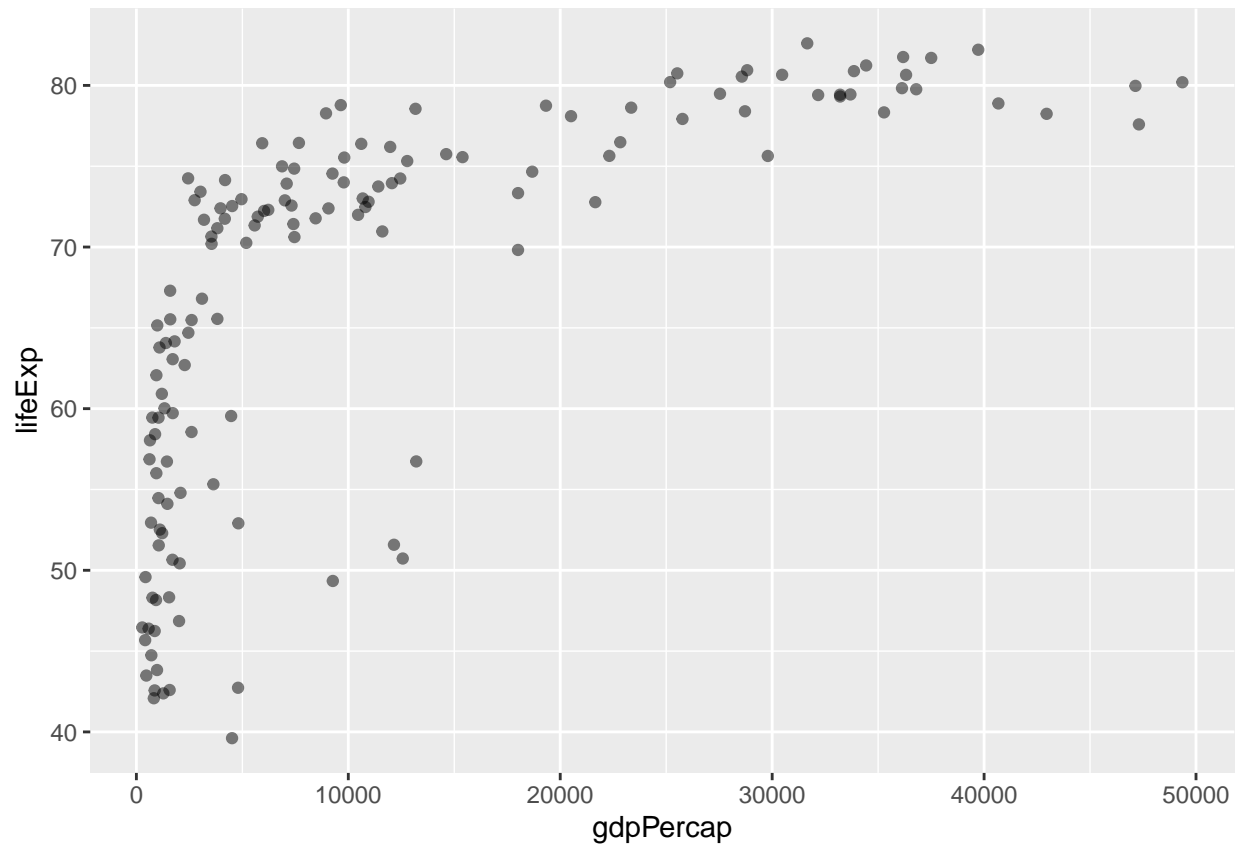
```
# Jitter points so they aren't stacked directly on top of each other
# Add boxplot to interpret distribution of points
life_exp_plot <- ggplot(gapminder) +
  aes(x=year, y=lifeExp, col=continent) +
  geom_boxplot(aes(group=year, alpha=0.2)) + #alternatively, can use other plots (e.g. violin)
  geom_jitter(width=0.3, alpha=0.4) #offset points

# Filter df to year 2007
gapminder_2007 <- gapminder %>% filter(year==2007)

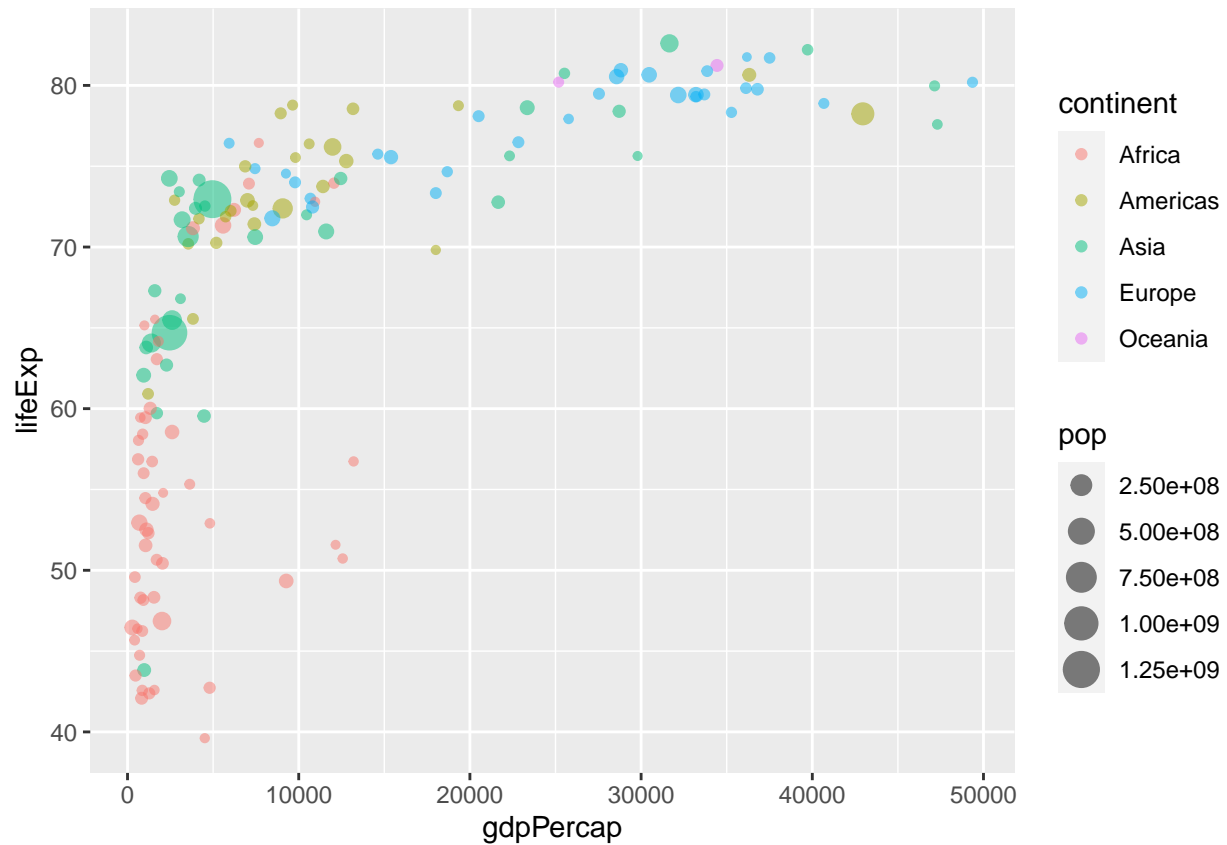
# For data from 2007, plot gdp per capita vs life expectancy
ggplot(gapminder_2007) +
  aes(x=gdpPercap, y=lifeExp) +
  geom_point()
```



```
# Decrease opacity of the points to visualize overlapping points  
ggplot(gapminder_2007) +  
  aes(x=gdpPercap, y=lifeExp) +  
  geom_point(alpha=0.5)
```



```
# Map continent to the color of the points and size to the population  
ggplot(gapminder_2007) +  
  aes(x=gdpPercap, y=lifeExp, col=continent, size=pop) +  
  geom_point(alpha=0.5)
```



```
#generate interactive plot, shareable by URL
#install.packages("plotly")
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
```

```
#ggplotly(life_exp_plot)
```