class05.R

sherrera

2021-10-13

```r
# Class 05: Data Visualization

# Today we are going to use ggplot2 package
# First we need to load the package!
# install.packages("ggplot2")
library(ggplot2)

# We will use this inbuilt "cars" dataset first
head(cars)
```

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

```r
# All ggplots have at least 3 layers,
# data + aes + geoms
ggplot(data=cars) +
  aes(x=speed, y=dist) +
  geom_point() +
  geom_smooth(method="lm") +
  labs(title="Stopping Distance of Old Cars",
       x="Speed (MPH)",
       y="Stopping Distance (ft)")
```
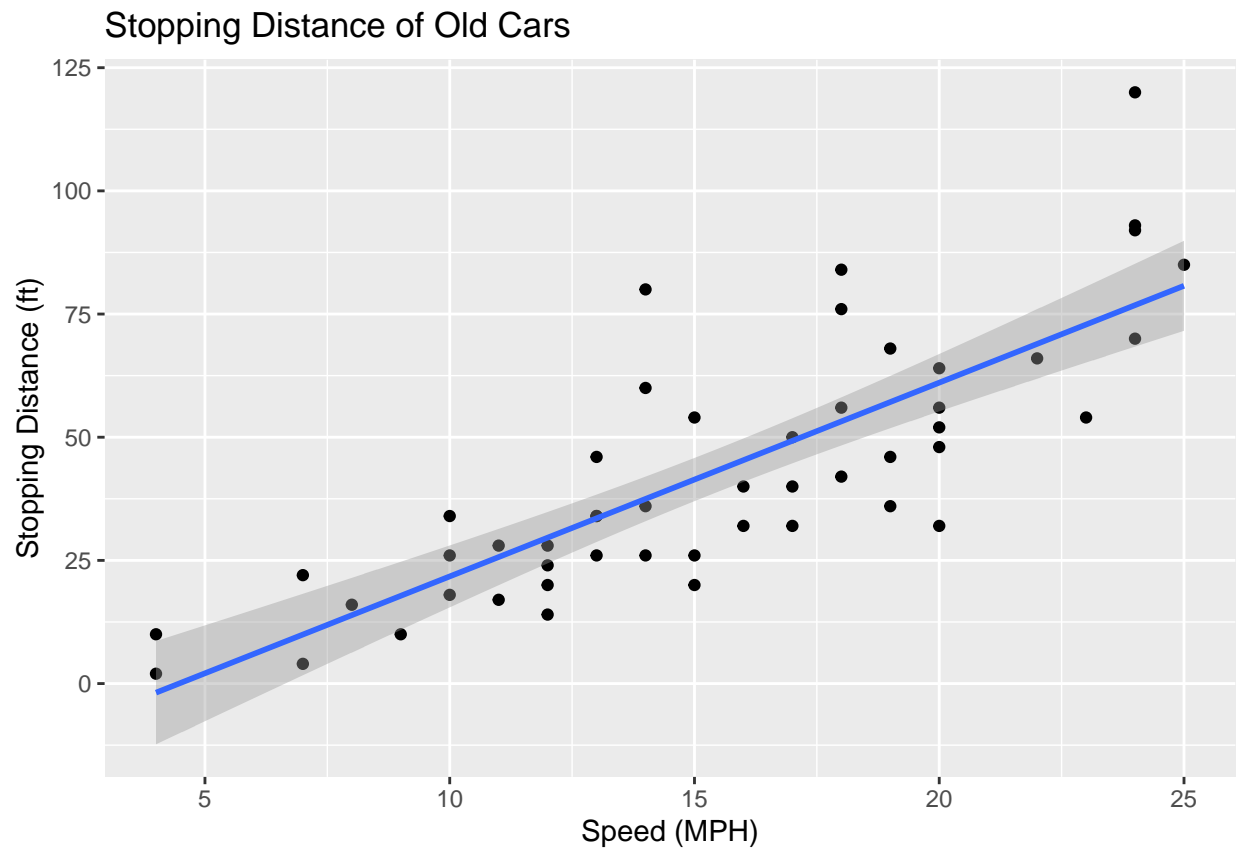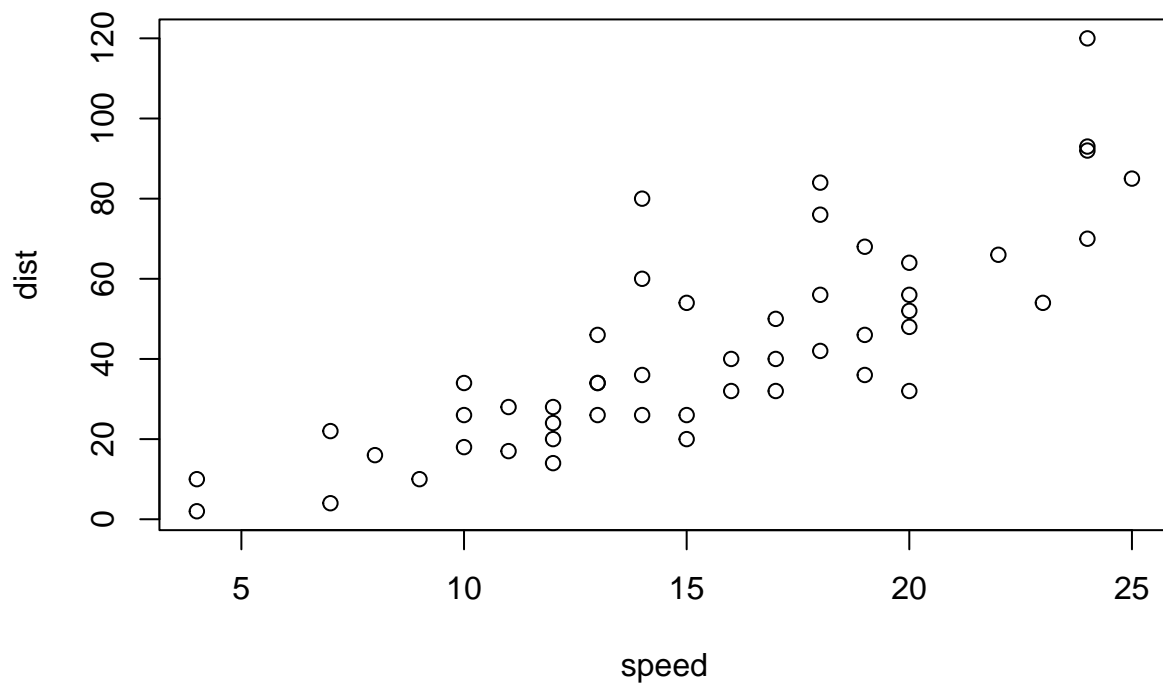
```
## `geom_smooth()` using formula 'y ~ x'
```

Stopping Distance of Old Cars


```
# Side note: ggplot is not the only graphics system
# A very popular one is good old "base" R graphics
plot(cars)
```

```r
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
```

```r
#Q How many genes in this dataset
nrow(genes)
```

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

```r
colnames(genes)
```

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

```r
ncol(genes)
```

```
## [1] 4
```
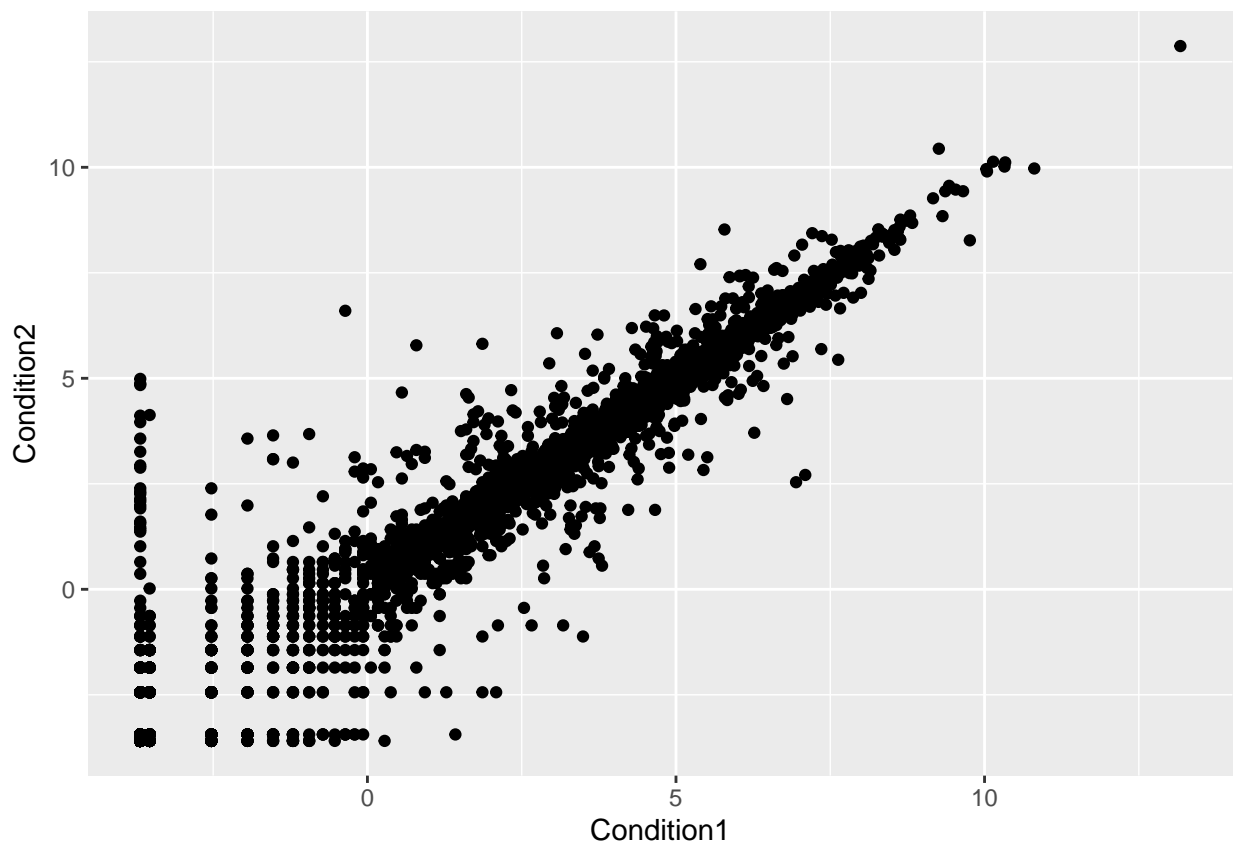
```
#Q How many genes are up
table(genes$State)
```

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

```
# To obtain the % of up genes compared to total genes:
round( table(genes$State)/nrow(genes) * 100, 2 )
```
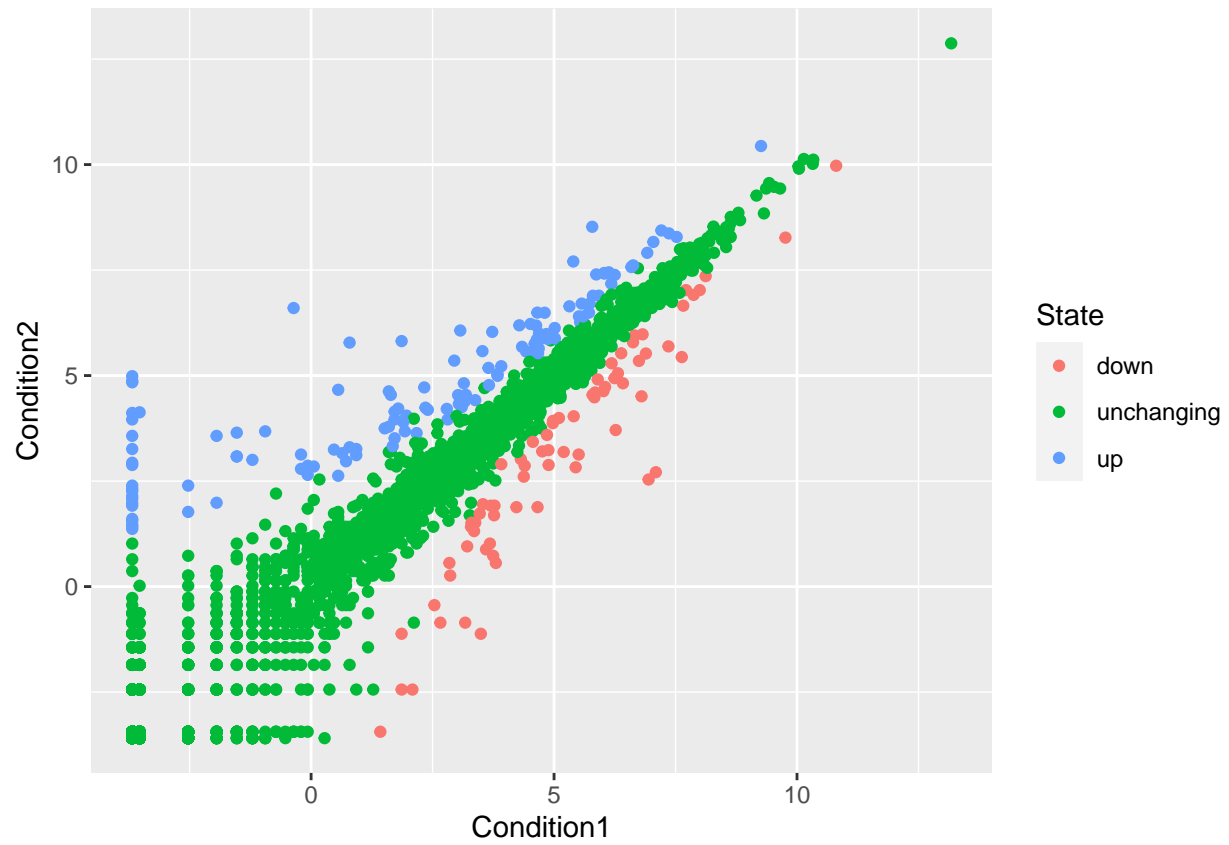
```
##
##       down unchanging         up
##       1.39      96.17       2.44
```
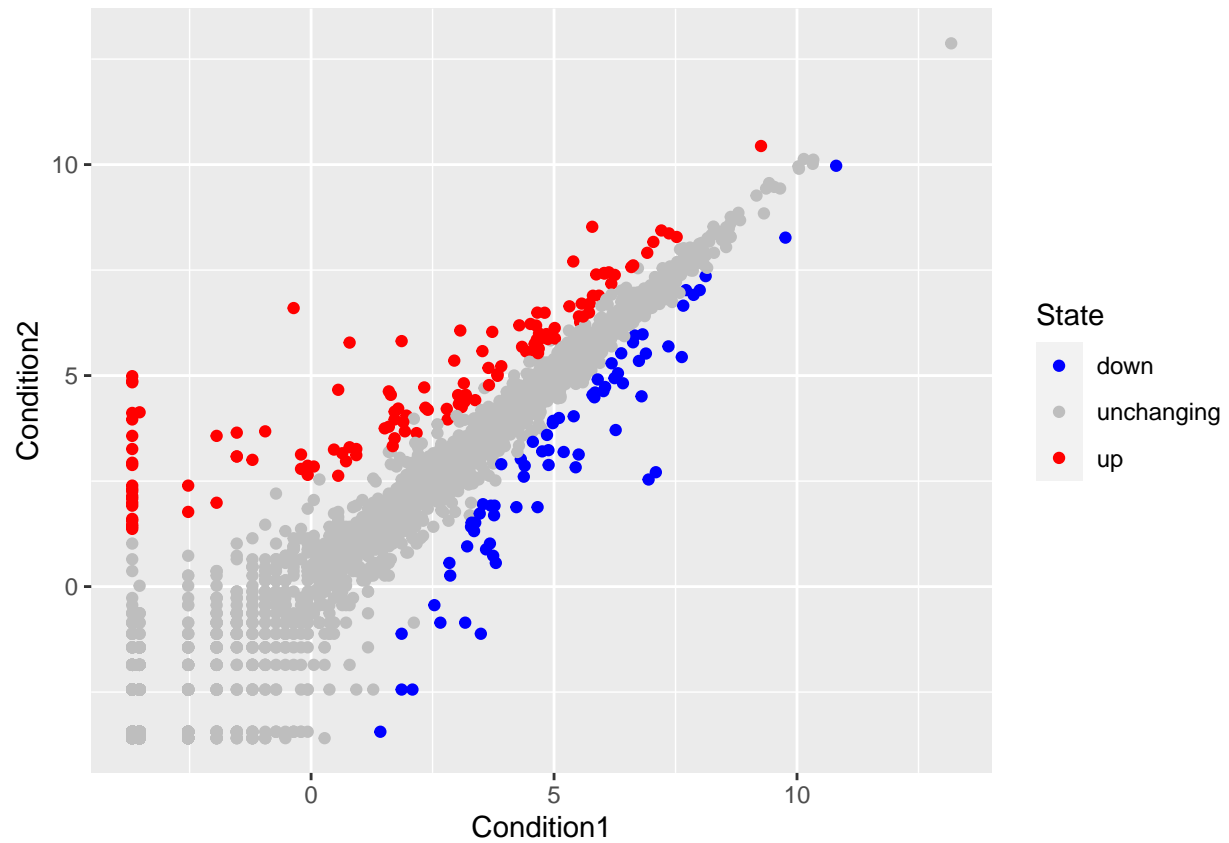
```
# Make first basic scatter plot

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



```
# Adding a third object, State (genes up or down) and saving it as an object, "p":
p <- ggplot(genes) +
  aes(x=Condition1, y=Condition2, col=State) +
  geom_point()
p
```
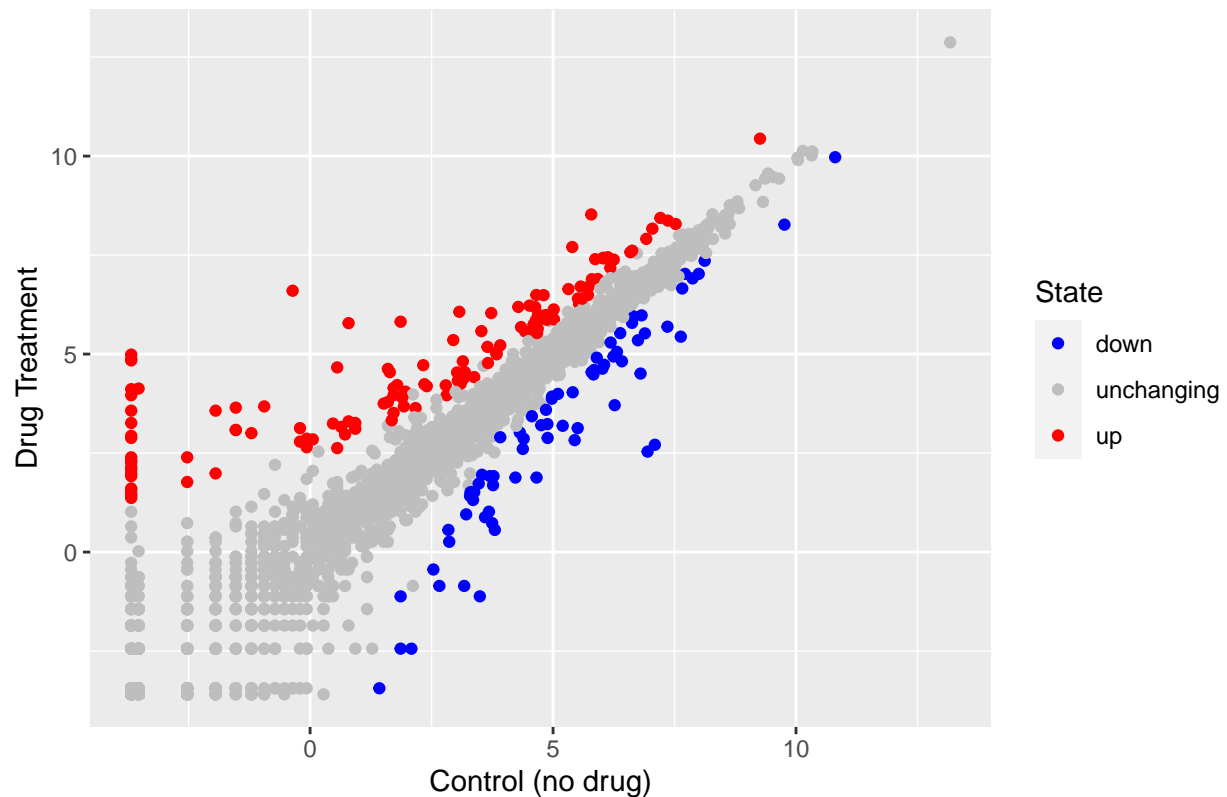
```
# Changing colors:
p + scale_colour_manual( values=c("blue", "gray", "red") )
```

```
# Changing labels:
p + scale_colour_manual( values=c("blue", "gray", "red") ) +
  labs(title="Gene Expression Changes Upon Drug Treatment",
       x="Control (no drug)",
       y="Drug Treatment")
```

Gene Expression Changes Upon Drug Treatment


```r
# Let's explore the gapminder dataset
# install.packages("gapminder")
library(gapminder)
head(gapminder)
```

```
## # A tibble: 6 x 6
##   country     continent  year lifeExp      pop gdpPercap
##   <fct>       <fct>     <int>   <dbl>    <int>     <dbl>
## 1 Afghanistan Asia       1952    28.8  8425333      779.
## 2 Afghanistan Asia       1957    30.3  9240934      821.
## 3 Afghanistan Asia       1962    32.0 10267083      853.
## 4 Afghanistan Asia       1967    34.0 11537966      836.
## 5 Afghanistan Asia       1972    36.1 13079460      740.
## 6 Afghanistan Asia       1977    38.4 14880372      786.
```

```r
# Let's make a new plot of year vs lifeExp (we can use boxplot/violin)
ggplot(gapminder) +
  aes(x=year, y=lifeExp, color=continent) +
  geom_jitter(width=0.3,alpha=0.4) +
  geom_violin( aes(group=year), alpha=0.2, draw_quantiles = c(0.5))
```
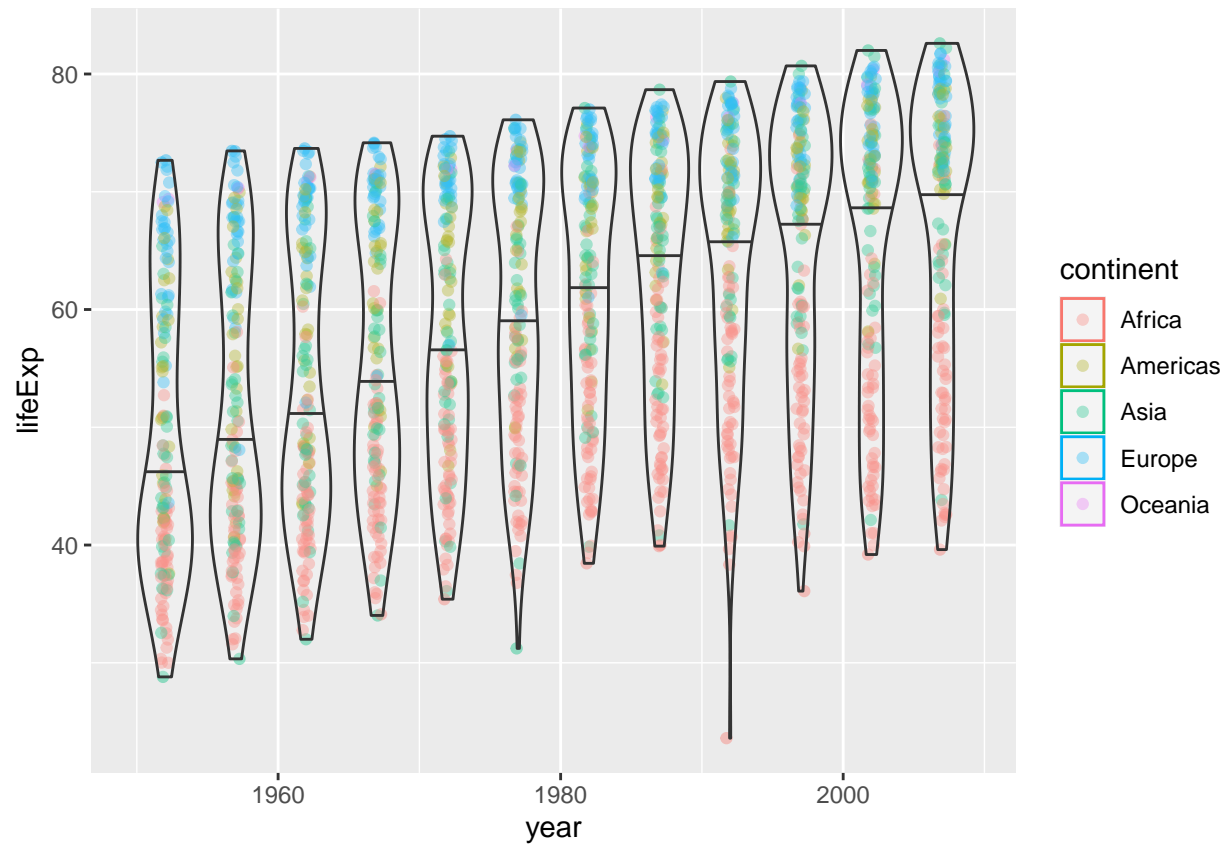
```r
# Let's turn it interactive
#Install the plotly package
# library(plotly)
# ggplotly()
# ggplotly(p)
```