## Assignment 2

For this assignment I'd like for you to explore the power of ggplot while also applying your data visualization critique knowledge to your new visualizations

### Step 1 (1pt)

Load tidyverse package. Load gapminder package Assign the data in the gapminder package to an object "data"

```
#how to check to see if a package is installed
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.3     v dplyr   1.0.7
## v tidyr   1.1.3     v stringr 1.4.0
## v readr   2.0.1     v forcats 0.5.1

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(gapminder)

data <- tibble(gapminder)
```

### Step 2 (2pts)

Explore the variables and dimensions of your newly created object using the head() and dim() functions

```
head(data)
```

```
## # 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.
```

```
dim(data)
```

```
## [1] 1704    6
```

**Step 3 (2pts)**

Create a data visualization that explores the relationship between year and life expectancy in the gapminder data

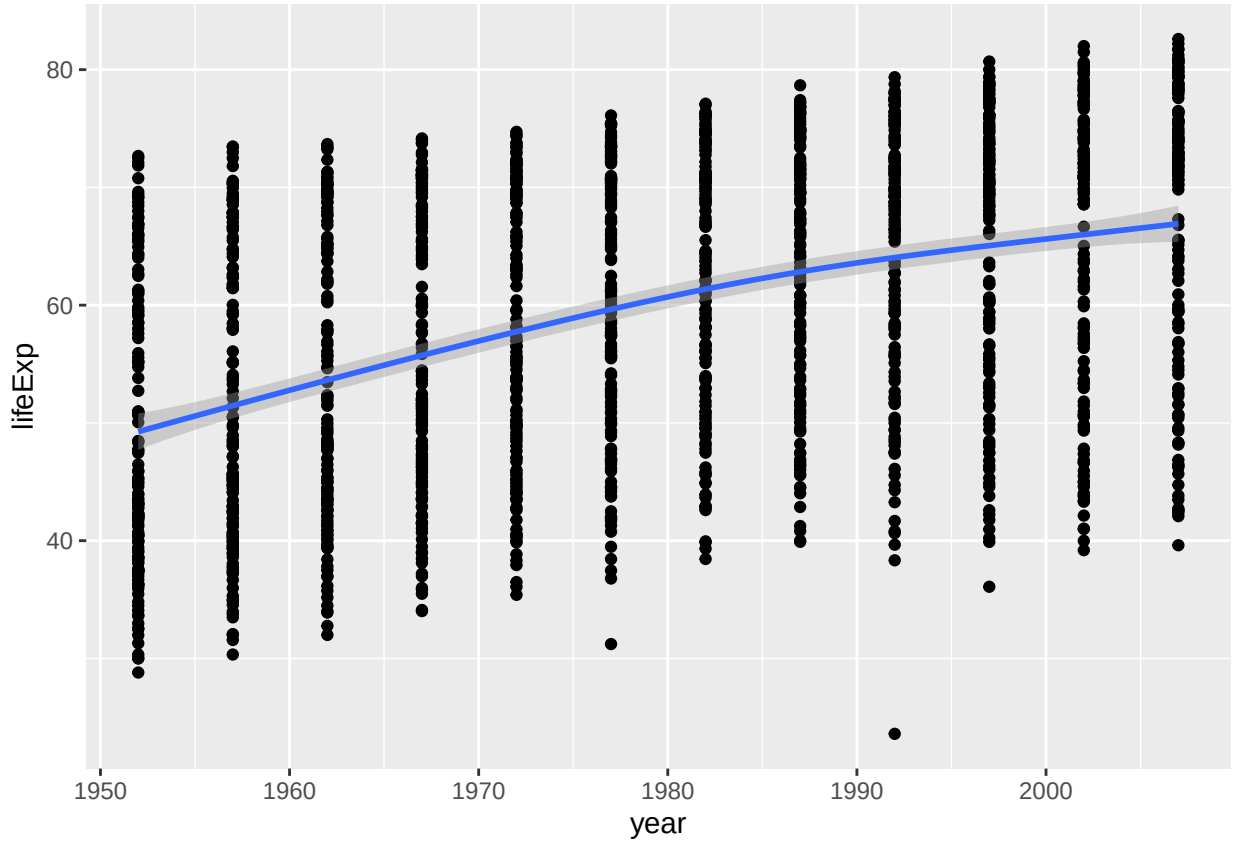
```
year_lifeexp <- ggplot(data = data, mapping = aes(x = year, y = lifeExp))

year_lifeexp <- year_lifeexp + geom_point()

year_lifeexp <- year_lifeexp + geom_smooth()

year_lifeexp
```

```
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
```




**Step 4 (2pts)**

Evaluate your data visualization based on the 5 principles of an effective data visualizations we talked about in class on 9/4. Make sure to use *markdown* effectively.

The **objective** of this visualization was to explore the relatoinship between year and lifeexp. With the smooth fitting line, it is clear there is a general increase in life expectancy across the last 60 years.

There is a good data to ink ratio but the **aesthetics** of this visualization is rather bland and unmemorable. The **data** is also not effectively represented here because the points overlap with each other making it hard

to evaluate the trends over time. The use of color helps the viewer quickly ascertain that their is a positive correlation here but the significance of that correlation is not clear.

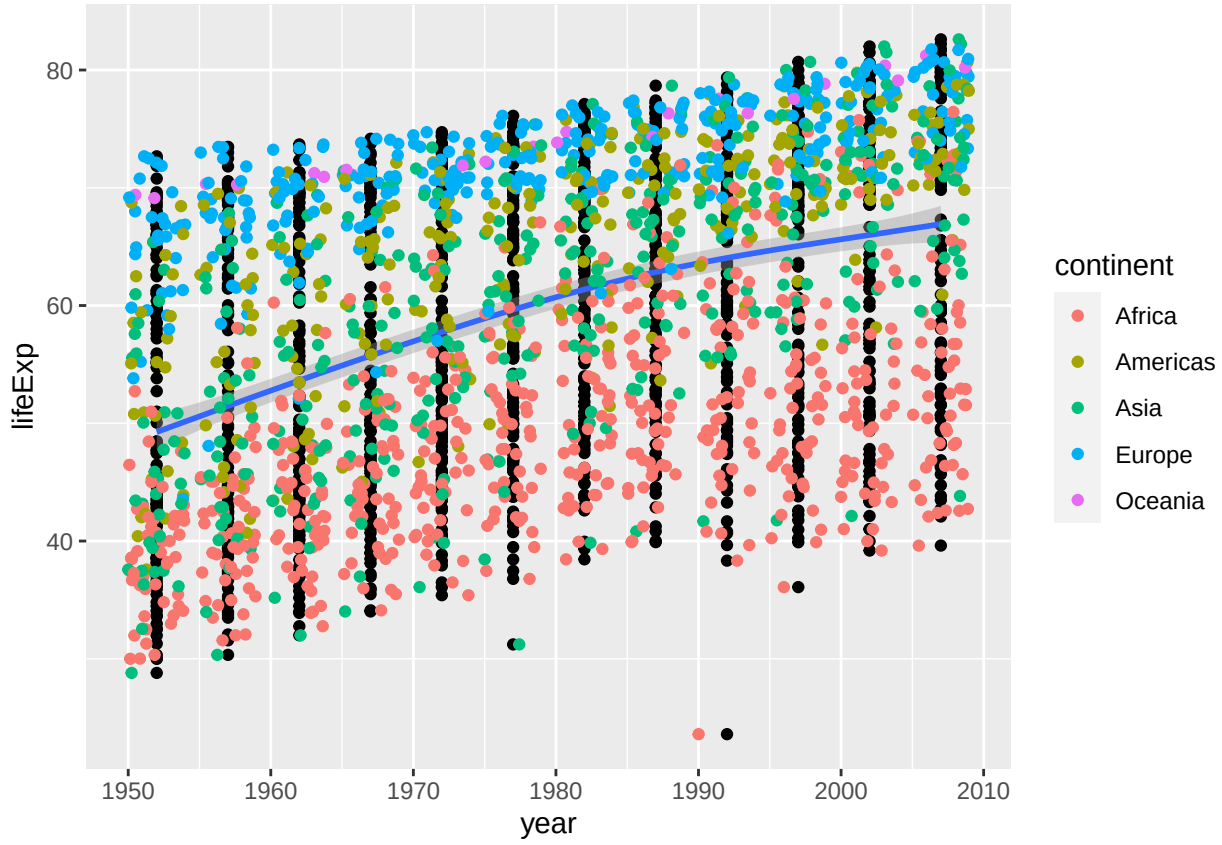
### Step 5 (3pts)

Create a new data visualization that adds an additional variable to your visualization (maps a new variable to a new aesthetic). Describe the resulting visualization (3 sentences or less)

```
year_lifeexp_2 <- year_lifeexp + geom_jitter(mapping = aes(color = continent))

year_lifeexp_2
```

```
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
```




### Next steps

Let's say you want to look at just countries in africa and the americas. What code would we run to filter the data?

```
AA <- c("Americas","Africa")

#option 1
just_AA <- filter(data, continent %in% AA)
```

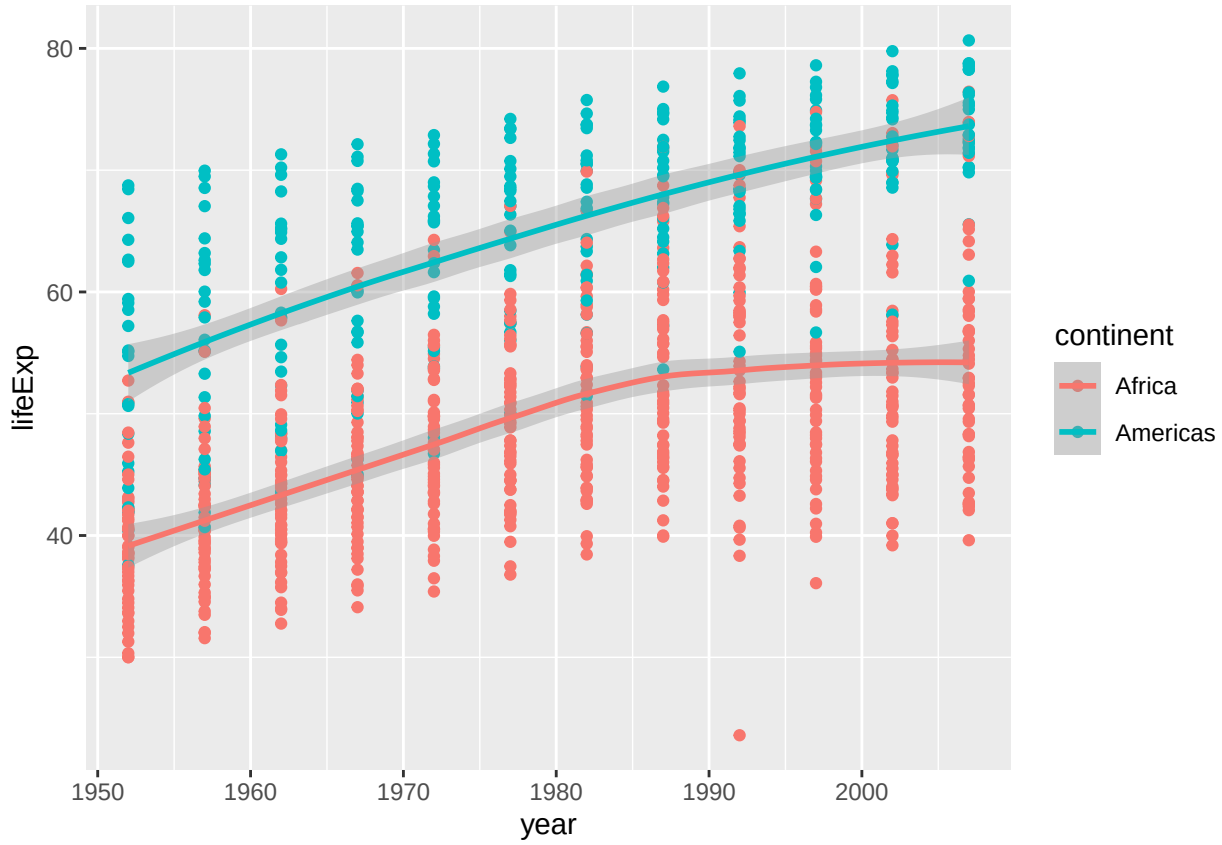
```
#option 2
just_AA <- filter(data, continent == "Africa" | continent == "Americas")
```

Now we want to cut down on the code a bit by reducing the number of times we modify our ggplot object

```
just_AA_viz <- ggplot(data = just_AA, mapping = aes(x = year, y = lifeExp, color = continent)) +
                        geom_point()+
                        geom_smooth()


just_AA_viz
```

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




Next we want address another exploratory question. Does population have a meaningful impact on this relationship between year and life expectancy?

```
life_AA_withpop_viz <- ggplot(data = just_AA, mapping = aes(x = year,
                                                            y = lifeExp,
                                                            color = continent,
                                                            size = pop)) +
                        geom_point()+
                        geom_smooth()

life_AA_withpop_viz
```

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