Data Visualization with ggplot2 (Bivariate)

Learning Objectives

1. Compare a quantitative variable across a categorical variable using side by side boxplots
2. Summarize two categorical variables using tables
3. Summarize two categorical variables using bar charts
4. Summarize two quantitative variables using scatterplots

We will be using another dataset, `gapminder`, from the `gapminder` package. Install and load the `gapminder` package. Also load the `tidyverse` package (which automatically loads the `ggplot2` package).

```
library(tidyverse)
library(gapminder)
```

We can take a look at the `gapminder` dataset

```
gapminder[1:15,]
```

```
## # A tibble: 15 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.
##  7 Afghanistan Asia       1982    39.9 12881816      978.
##  8 Afghanistan Asia       1987    40.8 13867957      852.
##  9 Afghanistan Asia       1992    41.7 16317921      649.
## 10 Afghanistan Asia       1997    41.8 22227415      635.
## 11 Afghanistan Asia       2002    42.1 25268405      727.
## 12 Afghanistan Asia       2007    43.8 31889923      975.
## 13 Albania     Europe     1952    55.2  1282697     1601.
## 14 Albania     Europe     1957    59.3  1476505     1942.
## 15 Albania     Europe     1962    64.8  1728137     2313.
```

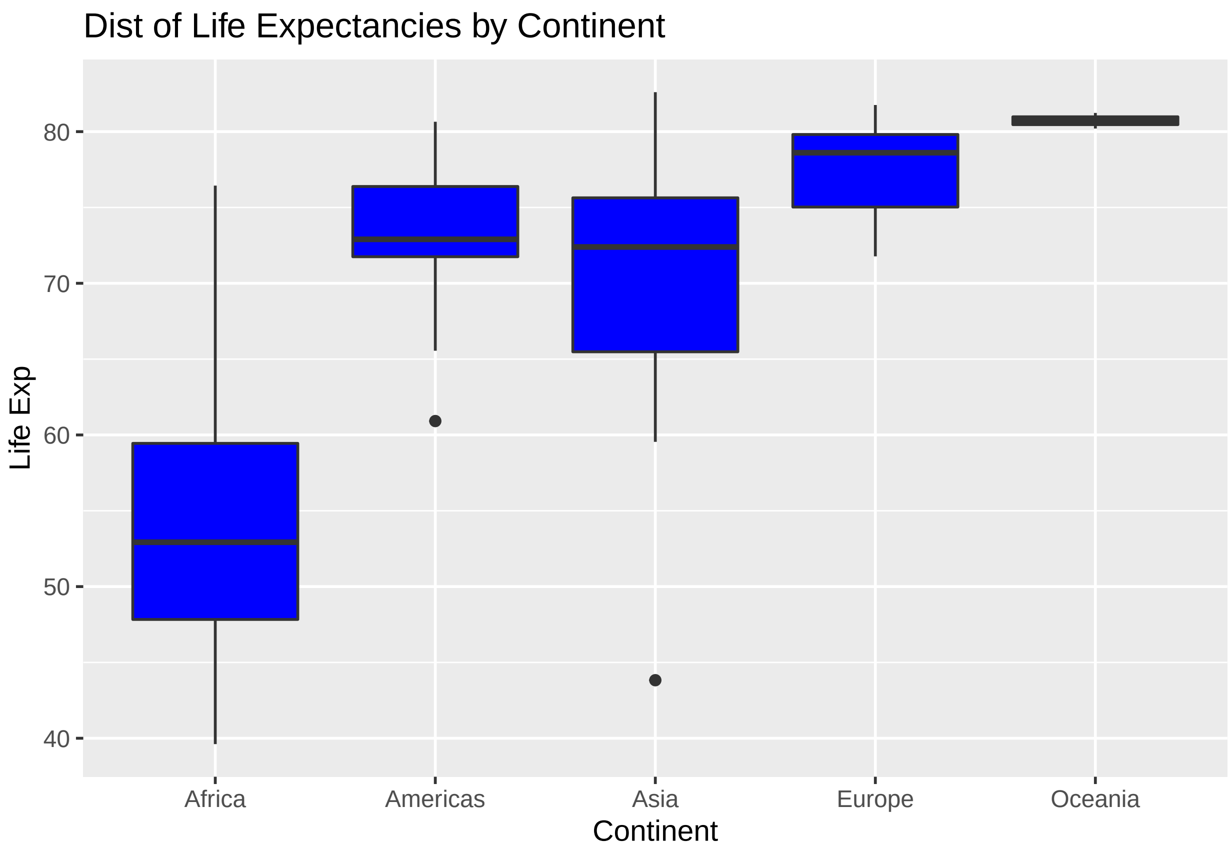
We notice that data are collected from each country across a number of different years: 1952 to 2007 in increments of five years. For this example, we will mainly focus on the data for the most recent year, 2007.

```
Data<-gapminder%>%
  filter(year==2007)
```

1. Compare a quantitative variable across a categorical variable using side by side boxplots

As mentioned previously, side by side boxplots are useful to compare the distribution of a quantitative variable across a categorical variable. For example, to view the distribution of life expectancies across the continents in 2007

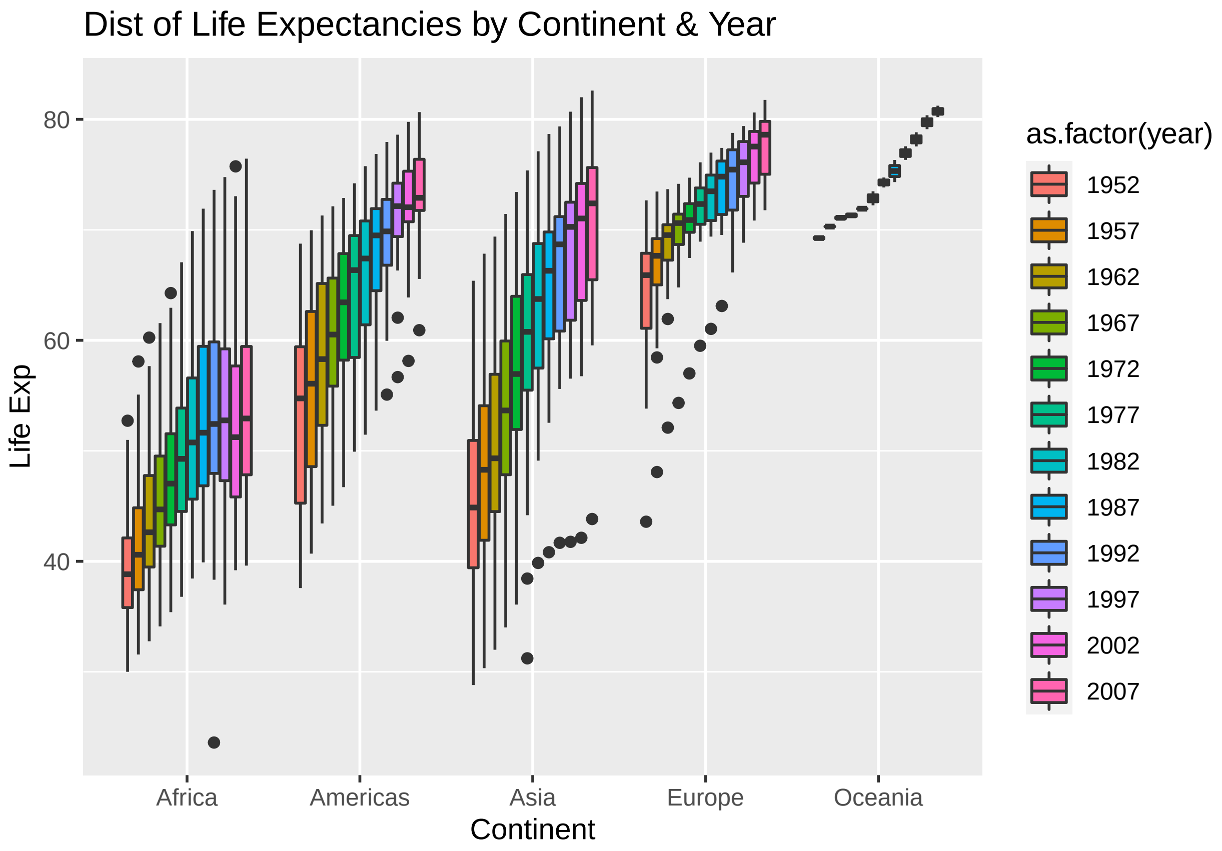
```
ggplot(Data, aes(x=continent, y=lifeExp))+
  geom_boxplot(fill="Blue")+
  labs(x="Continent", y="Life Exp",
       title="Dist of Life Expectancies by Continent")
```



Countries in the Oceania region have long life expectancies with little variation. Comparing the Americas and Asia, the median life expectancies are similar, but the spread is larger for Asia.

Since the data were collected over a number of years, we can compare boxplots of life expectancies for the continents over the years

```
ggplot(gapminder, aes(x=continent, y=lifeExp, fill=as.factor(year)))+
  geom_boxplot()+
  labs(x="Continent", y="Life Exp",
       title="Dist of Life Expectancies by Continent & Year")
```



2. Summarize two categorical variables using tables

For this example, we create a new binary variable called `expectancy`, which will be denoted as `low` if the life expectancy in the country is less than 70 years, and `high` otherwise.

```
Data<-Data%>%
  mutate(expectancy=ifelse(lifeExp<70,"Low","High"))
```

Suppose we want to see how `expectancy` varies across the continents. A two-way table can be created for produce counts

```
mytab2<-table(Data$continent, Data$expectancy)
##continent in rows, expectancy in columns
mytab2
```

```
##
##              High Low
##   Africa       7  45
##   Americas    22   3
##   Asia        22  11
##   Europe      30   0
##   Oceania      2   0
```

The first variable in `table()` will be placed in the rows, the second variable will be placed in the columns.

To convert this table to proportions, we can use `prop.table()`

```
prop.table(mytab2, 1)
```

```
##
##                   High       Low
##   Africa   0.1346154 0.8653846
##   Americas 0.8800000 0.1200000
##   Asia     0.6666667 0.3333333
##   Europe   1.0000000 0.0000000
##   Oceania  1.0000000 0.0000000
```

In this example, it makes sense to want proportions for each continent, so we want proportions in each row to add up to 1. Therefore, the second argument in `prop.table()` is 1. Enter 2 for this argument if we want the proportions in each column to add up to 1.

As before, to convert to percentages and round to 2 decimal places

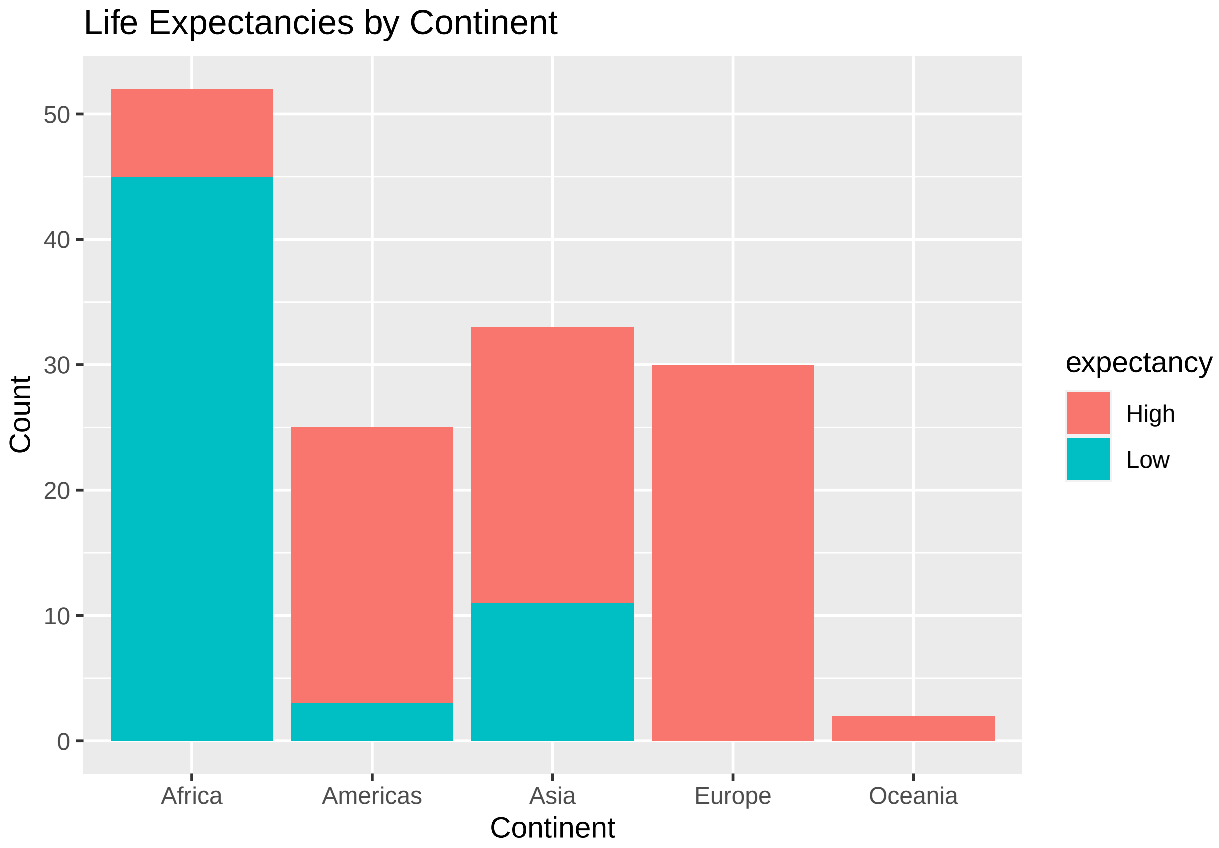
```
round(prop.table(mytab2, 1) * 100, 2)
```

```
##
##              High    Low
##   Africa    13.46  86.54
##   Americas  88.00  12.00
##   Asia      66.67  33.33
##   Europe   100.00   0.00
##   Oceania  100.00   0.00
```

3. Summarize two categorical variables using bar charts

A stacked bar chart can be used to display the relationship between the binary variable `expectancy` across continents.

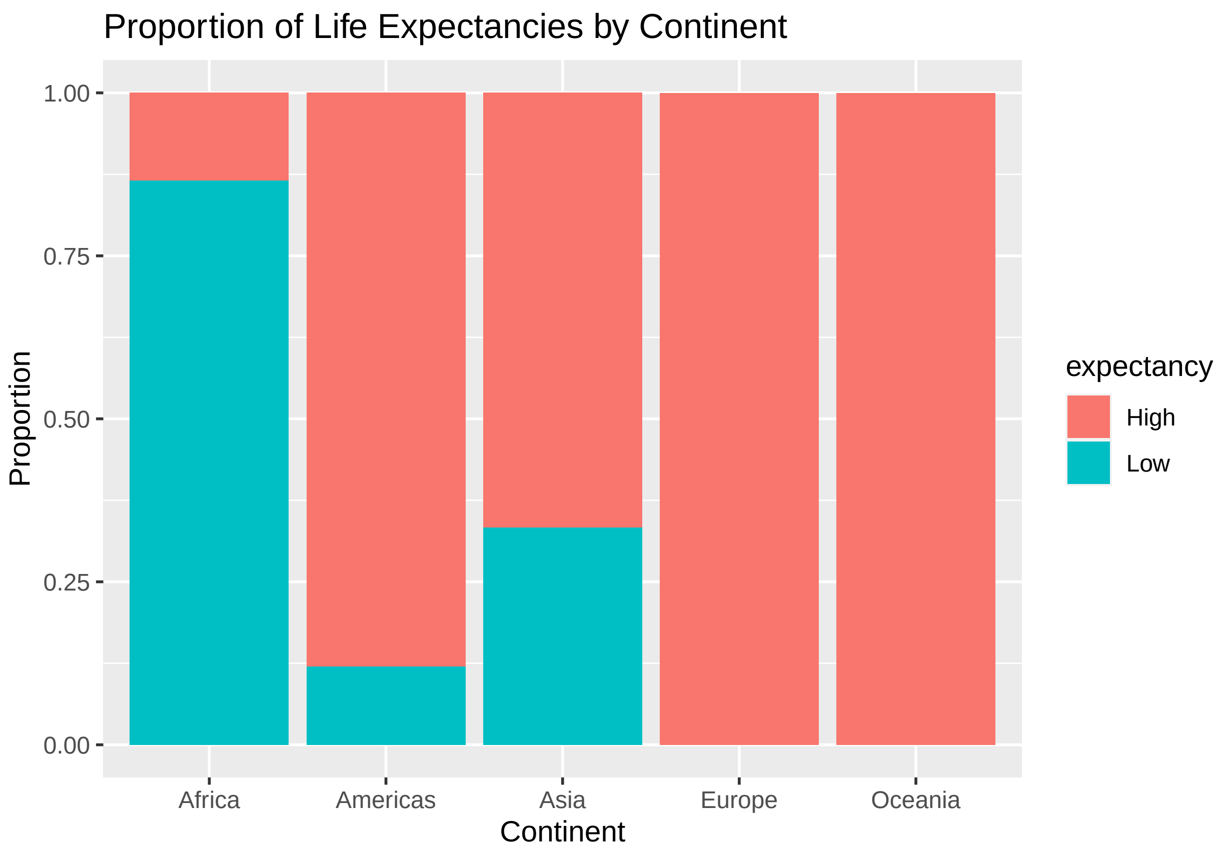
```
ggplot(Data, aes(x=continent, fill=expectancy))+
  geom_bar(position = "stack")+
  labs(x="Continent", y="Count", title="Life Expectancies by Continent")
```



We can see how many countries exist in each continent, and how many of these countries in each continent have high or low life expectancies. We can change the way the bar chart is displayed by changing `position` in `geom_bar()` to `position = "dodge"` or `position = "fill"`, the latter being more useful for proportions instead of counts.

```
ggplot(Data, aes(x=continent, fill=expectancy))+
  geom_bar(position = "dodge")
```

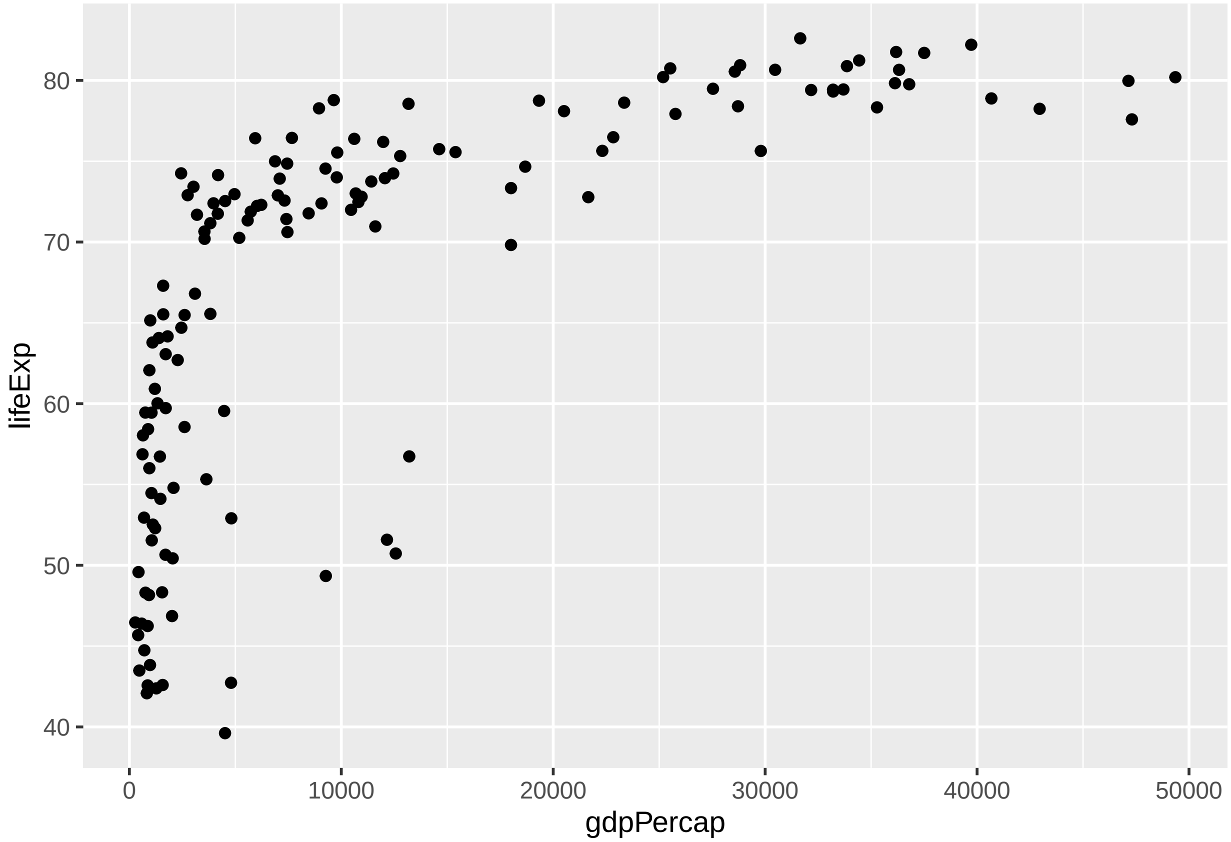
```
ggplot(Data, aes(x=continent, fill=expectancy))+
  geom_bar(position = "fill")+
  labs(x="Continent", y="Proportion",
       title="Proportion of Life Expectancies by Continent")
```



4. Summarize two quantitative variables using scatterplots

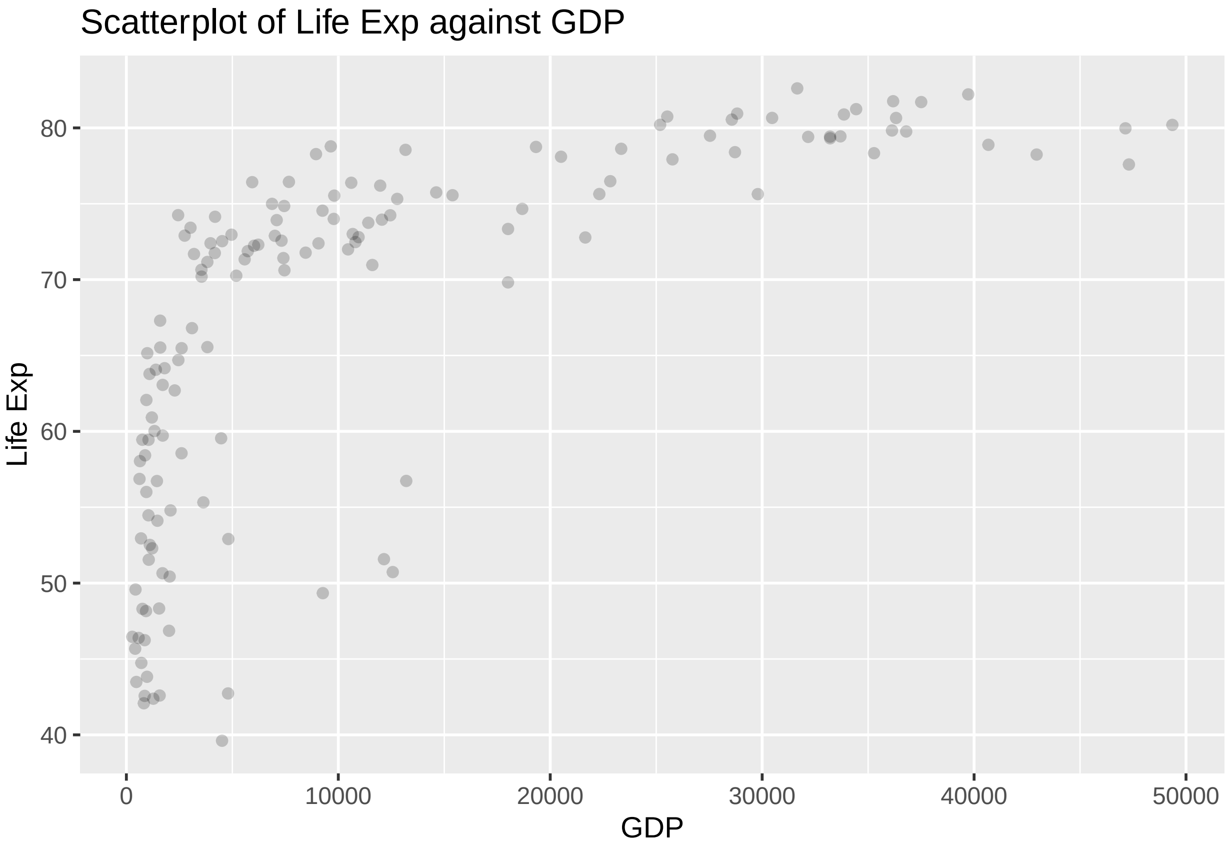
Scatterplots are the standard visualization when two quantitative variables are involved. To create a scatterplot for life expectancy against GDP per capita

```
ggplot(Data, aes(x=gdpPercap,y=lifeExp))+
  geom_point()
```



When there are many observations, plots on the scatterplot may actually overlap each other. To have a sense of how many of these exist, we can add a transparency scale called `alpha=0.2` inside ‘geom_point()

```
ggplot(Data, aes(x=gdpPercap,y=lifeExp))+
  geom_point(alpha=0.2)+
  labs(x="GDP", y="Life Exp",
       title="Scatterplot of Life Exp against GDP")
```



The default value for `alpha` is 1, which means the points are not at all transparent. The closer this value is to 0, the more transparent the points are. A darker point indicates more observations with those specific values on both variables.