Lesson notes | Boxplots with {ggplot2}

Created by the GRAPH Courses team

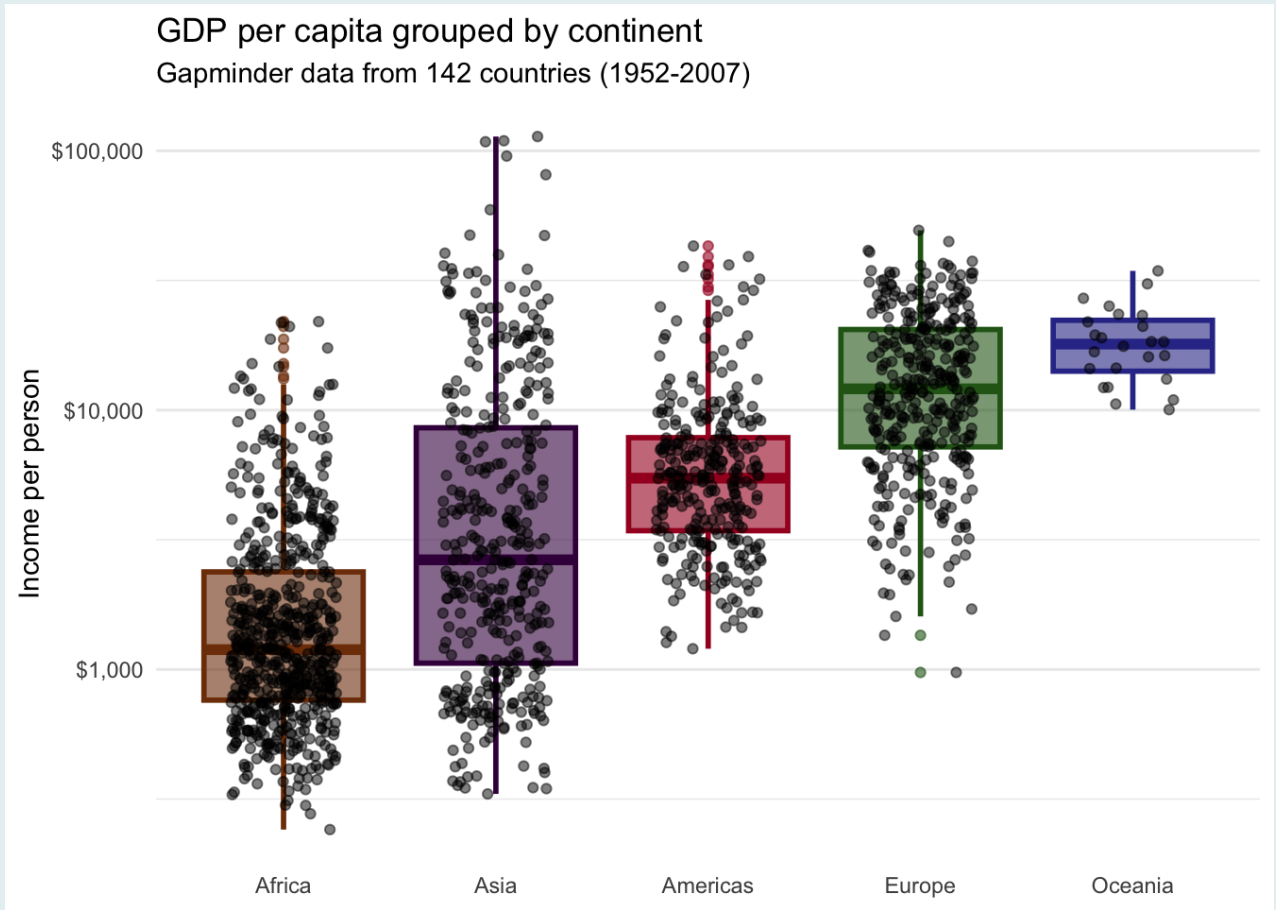
November 2023

This document serves as an accompaniment for a lesson found on https://thegraphcourses.org.

The GRAPH Courses is a project of the Global Research and Analyses for Public Health (GRAPH) Network, a non-profit headquartered at the University of Geneva Global Health Institute, and supported by the World Health Organization (WHO) and other partners

Boxplots with {ggplot2}

A side-by-side boxplot lets us compare the distribution of a numerical variable split by the values of another variable.



Learning Objectives

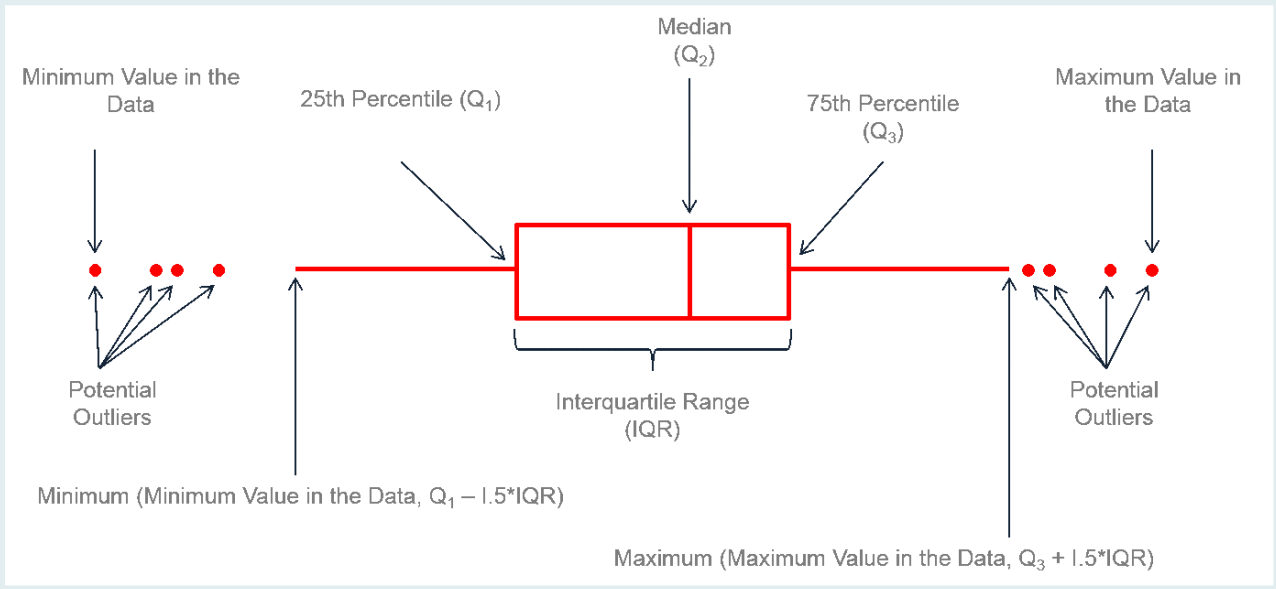
By the end of this lesson, you will be able to:

1. Plot a boxplot to visualize the distribution of continuous data using **`geom_boxplot()`**.
2. Reorder side-by-side boxplots with the **`reorder()`** function.
3. Add a layer of data points on a bloxplot using **`geom_jitter()`**.

Introduction

A boxplot is one of the simplest ways of representing a distribution of a continuous variable.

A boxplot allows us to visualize the distribution of one or more numeric variables.



Anatomy of a boxplot

It consists of two parts:

1. Box – Extends from the first to the third quartile (Q1 to Q3) with a line in the middle that represents the median. The range of values between Q1 and Q3 is also known as an Interquartile range (IQR).

2. Whiskers – Lines extending from both ends of the box indicate variability outside Q1 and Q3. The minimum/maximum whisker values are calculated as $Q1 - 1.5 \times IQR$ to $Q3 + 1.5 \times IQR$. Everything outside is represented as an outlier using dots or other markers.

Packages

```
        pacman::p_load(tidyverse,
                       gapminder,
                       here)
```

The gapminder dataset

For this lesson, we will be visualizing information from the `gapminder` data frame, which we've encountered in previous lessons. look at trends in world health and economics.

```
# View first few rows of the data
head(gapminder)
```

```
## # A tibble: 6 × 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.
```

RECAP

Gapminder is a country-year dataset with information on 142 countries, divided in to 5 "continents" or world regions.

```
# Data summary
summary(gapminder)
```

```
##         country        continent        year
##  Afghanistan:  12   Africa  :624   Min.   :1952
##  Albania    :  12   Americas:300   1st Qu.:1966
##  Algeria    :  12   Asia    :396   Median :1980
##  Angola     :  12   Europe  :360   Mean   :1980
##  Argentina  :  12   Oceania : 24   3rd Qu.:1993
##  Australia  :  12                  Max.   :2007
##  (Other)    :1632
##     lifeExp          pop             gdpPercap
##  Min.   :23.60   Min.   :6.001e+04   Min.   :   241.2
##  1st Qu.:48.20   1st Qu.:2.794e+06   1st Qu.:  1202.1
##  Median :60.71   Median :7.024e+06   Median :  3531.8
##  Mean   :59.47   Mean   :2.960e+07   Mean   :  7215.3
##  3rd Qu.:70.85   3rd Qu.:1.959e+07   3rd Qu.:  9325.5
##  Max.   :82.60   Max.   :1.319e+09   Max.   :113523.1
##
```

Data are recorded every 5 years from 1952 to 2007 (a total of 12 years).

Basic boxplots with `geom_boxplot()`

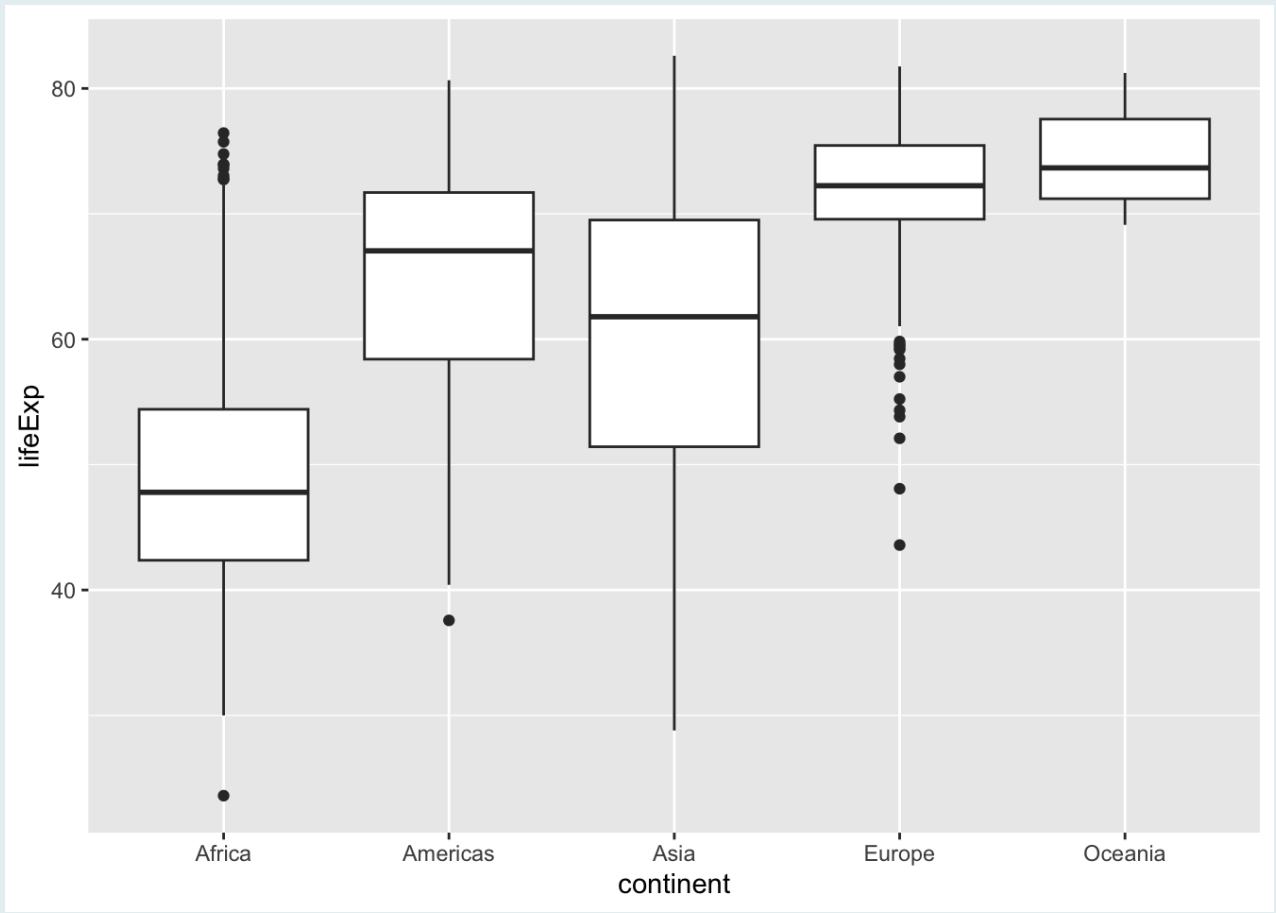
We will use boxplots display and compare *distributions* of variables across multiple groups.

The `gapminder` data frame gives us the life expectancy (`lifeExp`) for each country. Let's make a boxplot of life expectancy across continents.

Let's start with a base boxplot and then then add more aesthetics and layers from {ggplot2}.

We will first provide the `gapminder` data frame to `ggplot()` and then specify the aesthetics with `aes()` function. Inside `aes()`, we will specify *x*-axis and *y*-axis variables. To make the boxplot between `continent` vs `lifeExp`, we will use the `geom_boxplot()` layer

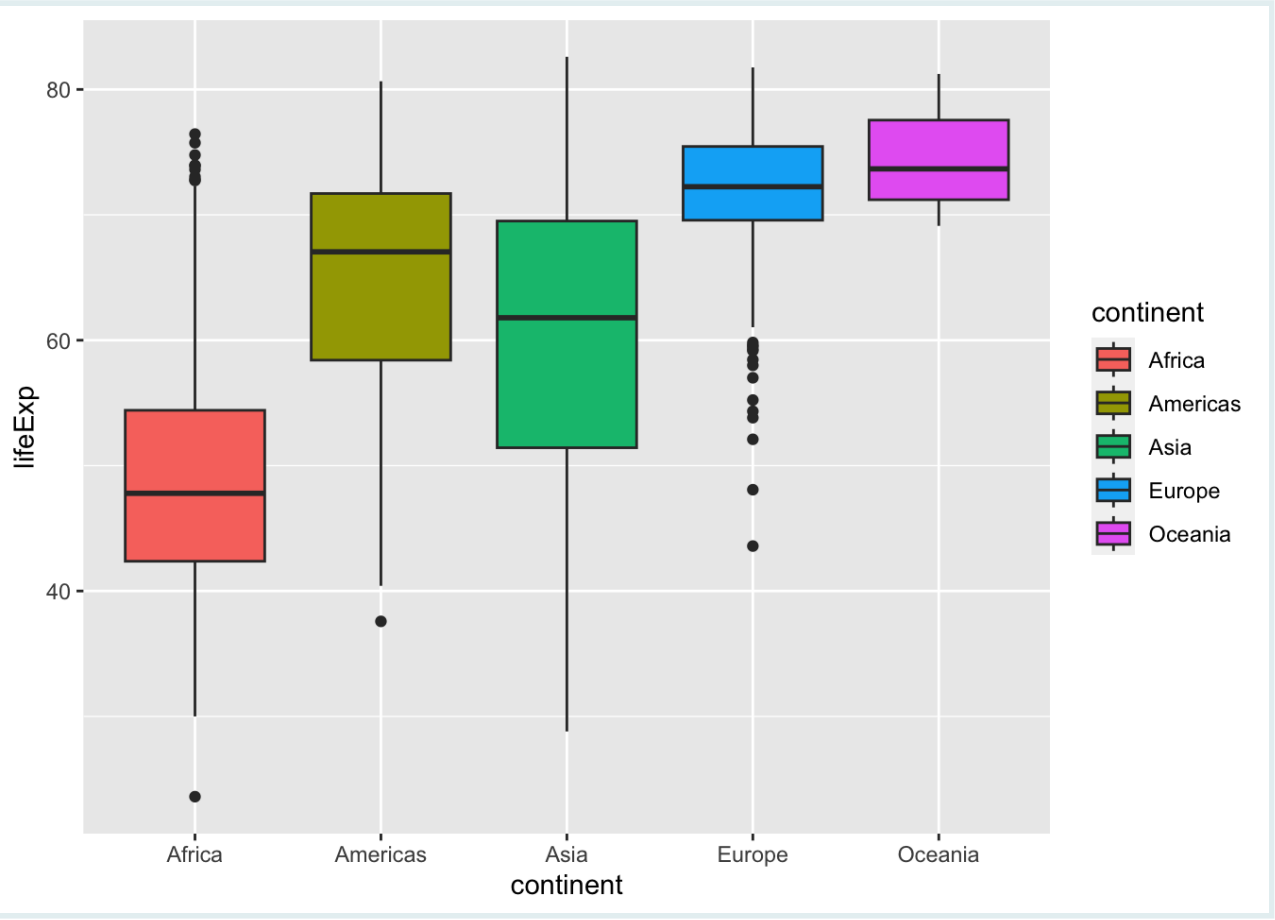
```
# Simple boxplot of lifeExp continent
ggplot(gapminder,
       aes(x = continent,
           y = lifeExp)) +
  geom_boxplot()
```



The result is a basic boxplot of `liefExp` for multiple continents.

Let us add colors to the basic boxplot. We can map the `continent` variable to fill color so that each box is colored according to which continent it represents.

```
ggplot(gapminder,
       aes(x = continent,
           y = lifeExp,
           fill = continent)) +
  geom_boxplot()
```



REMINDER

`ggplot2` allows you to color by specifying a variable. We can use `fill` argument inside the `aes()` function to specify which variable is mapped to fill color.

PRACTICE

(in RMD)

- Using the `gapminder` data frame create a boxplot comparing the distribution of GDP per capita (`gdpPercap`) across continents.

```
# Type and view your answer:
q1 <- "YOUR ANSWER HERE"

q1
```

- Similarly to how we changed the fill color of the boxes with the `fill` argument, change the outline color of the boxes with a `color` argument.

```
# Type and view your answer:
q2 <- "YOUR ANSWER HERE"

q2
```

MAKE THESE ONE QUESTION AND SPECIFY BORDER AND FILL COLORS

ADD SECTION TO RESCALE WITH SCALE_Y_LOG10()

The continents are a factor, and are ordered alphabetically by default. It might be more useful to order them by the mean or median life expectancy.

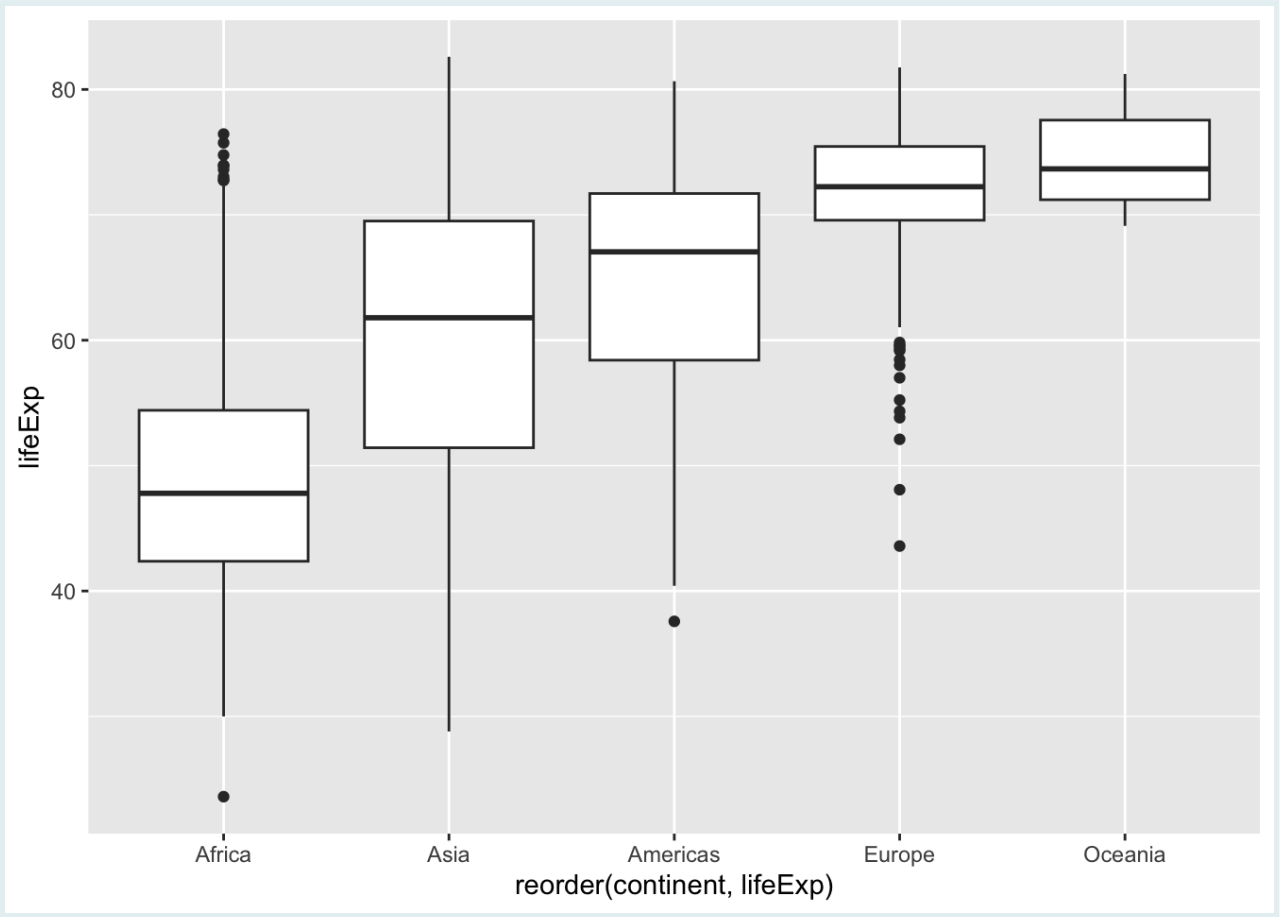
Reordering with `reorder()`

We can change the order of boxplots by using the `reorder()` function to rearrange the data being mapped on the *x*-axis.

`reorder()` treats its first argument as a categorical variable (usually a factor), and reorders its levels based on the values of a second variable (usually numeric). To reorder the levels of the `continent` variable based on `lifeExp`, we will write this: `reorder(continent, lifeExp)`.

Here we will edit the x argument and tell `ggplot` to plot the reordered variable.

```
ggplot(gapminder,
       aes(x = reorder(continent, lifeExp),
           y = lifeExp)) +
  geom_boxplot()
```



We can clearly see that there are notable differences in median life expectancy between continents. However, there is a lot of overlap between the range of values from each continent. For example, the median life expectancy for the continent of Africa is lower than that of Europe, but several African countries have life expectancy values higher than than the majority of European countries.

Reordering by function

mean, median, etc.

- Create the boxplot showing the distribution of GDP per capita for each continent, like you did in practice question 2. This time, change the x axis variable to reorder the boxes according to `gdpPercap`.

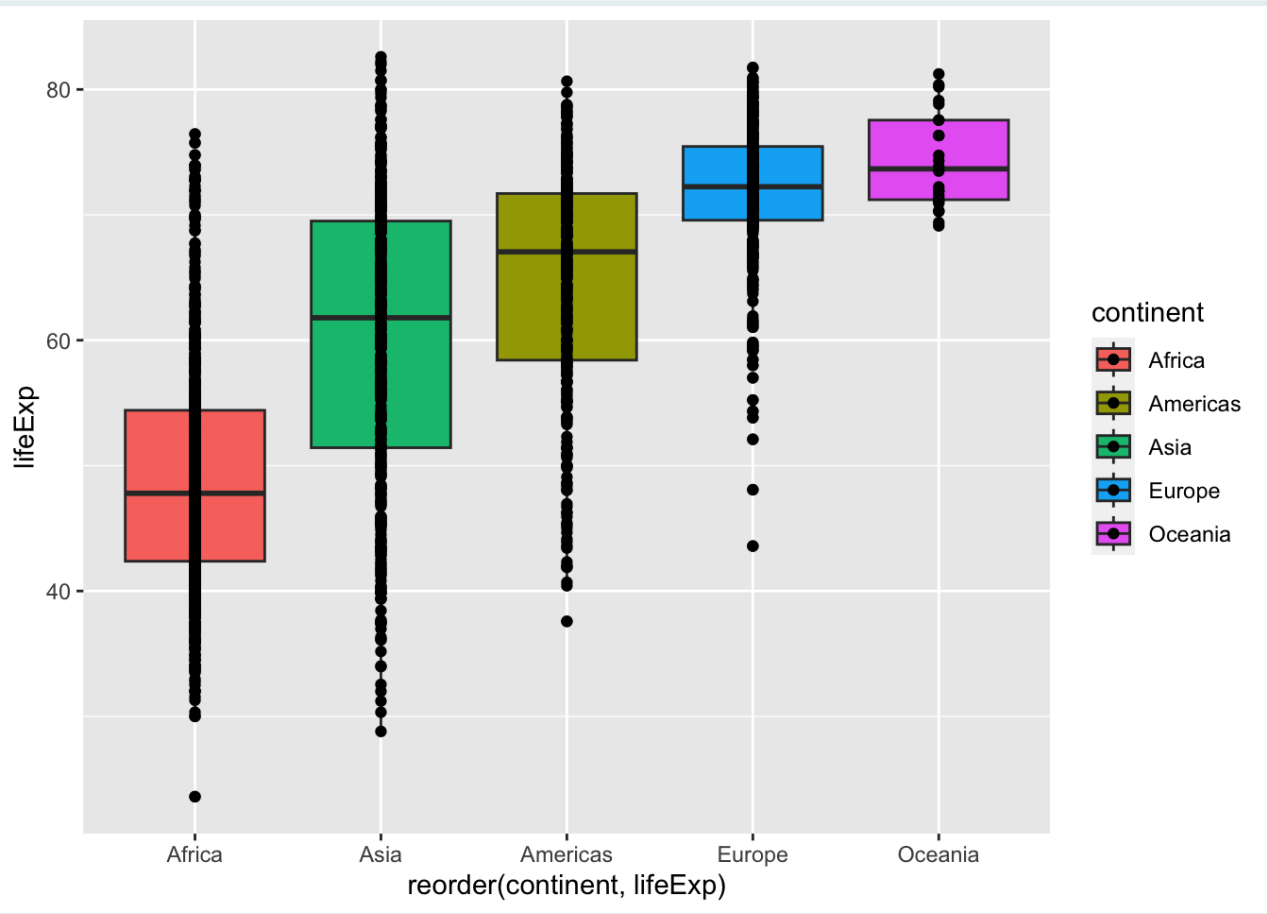
```
# Type and view your answer:
q3 <- "YOUR ANSWER HERE"

q3
```

ADD QUESTION ON LABS

Adding data points with `geom_jitter()`

Boxplots give us a very high-level summary of the distributions and do not show the actual life expectancy values for each country-year in the dataset. One way to display the distribution of individual data points is to plot an additional layer on top of the boxplot. We can do this by simply adding the `geom_point()` function.

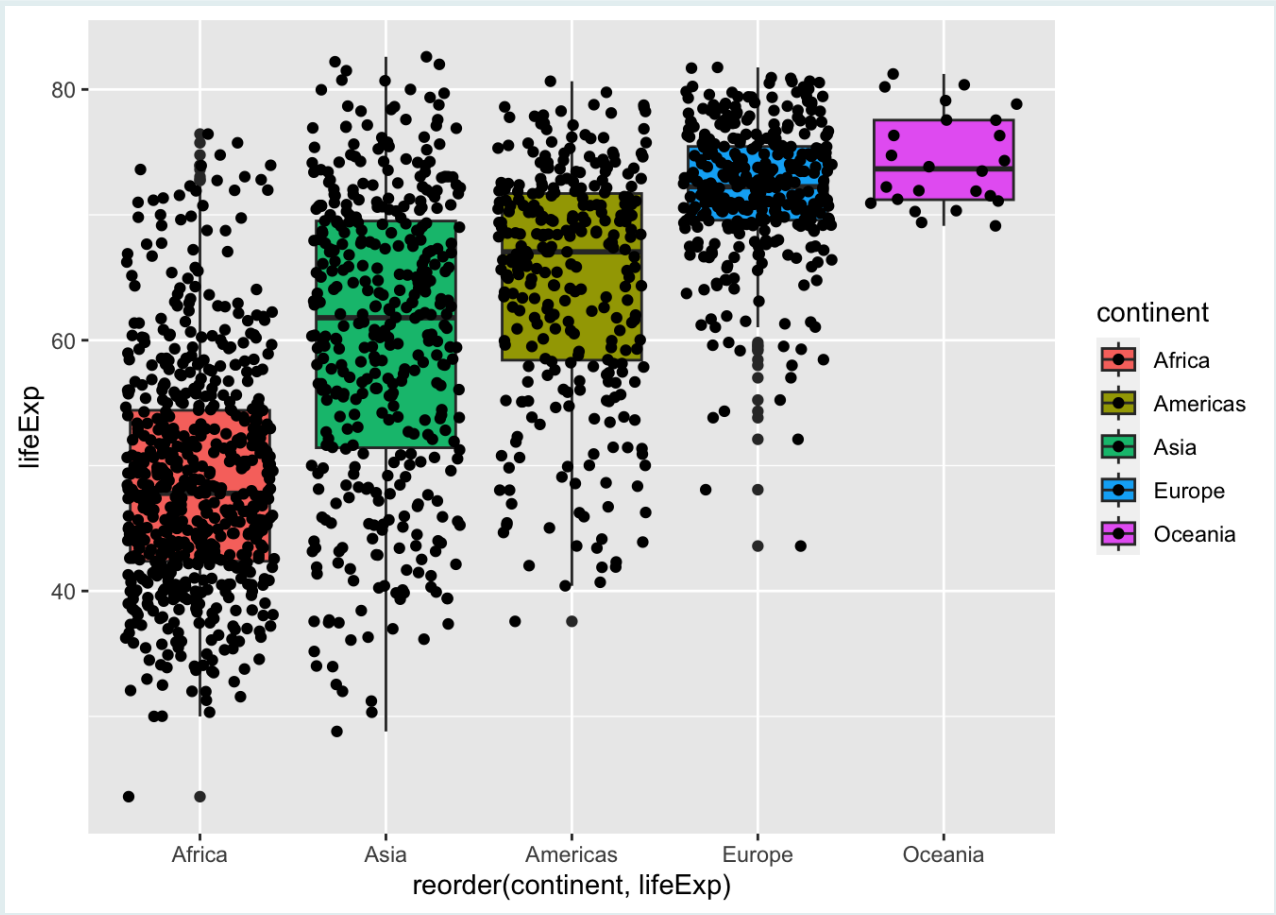
```
ggplot(gapminder,
       aes(x = reorder(continent, lifeExp),
           y = lifeExp,
           fill = continent)) +
      geom_boxplot()+
      geom_point()
```



Adding `geom_point()` as has plotted all the data points on a vertical line. That's not very useful since all the points with same life expectancy value directly overlap and are plotted on top of each other.

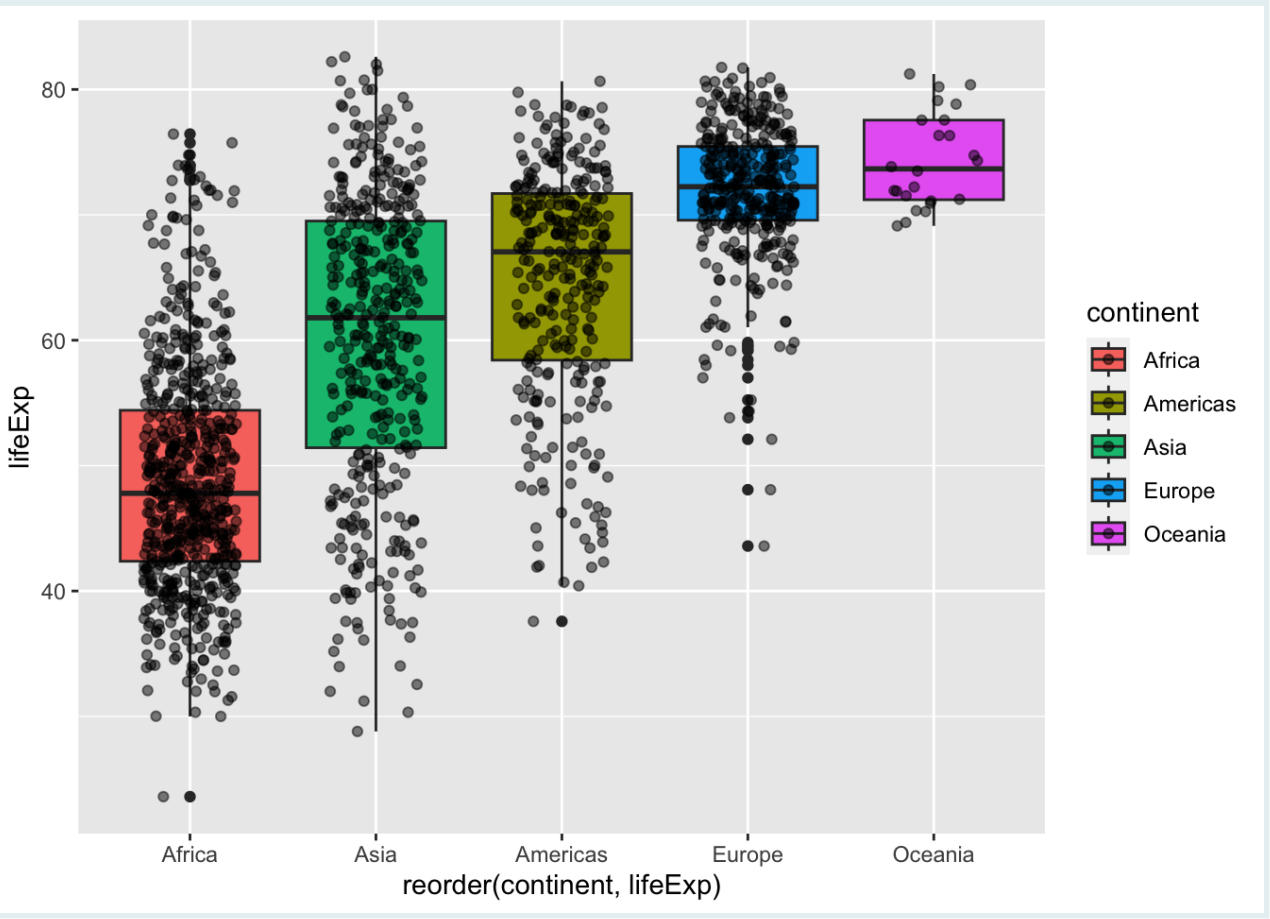
One solution for this is to randomly "jitter" data points horizontally. `ggplot` allows you to do that with the `geom_jitter()` function.

```
ggplot(gapminder,
       aes(x = reorder(continent, lifeExp),
           y = lifeExp,
           fill = continent)) +
  geom_boxplot() +
  geom_jitter()
```



You can also control the width of the jitter with `width` argument and specify transparency of data points with `alpha`.

```
ggplot(gapminder,
       aes(x = reorder(continent, lifeExp),
           y = lifeExp,
           fill = continent)) +
  geom_boxplot() +
  geom_jitter(width = 0.25,
              alpha = 0.5)
```



KEY POINT

Boxplots have the limitation that they summarize the data into five numbers: the 1st quartile, the median (the 2nd quartile), the 3rd quartile, and the upper and lower whiskers. By doing this, we might miss important characteristics of the data. One way to avoid this is by showing the data with points.

PRACTICE (in RMD)

- Create the boxplot showing the distribution of GDP per capita for each continent, like you did in practice question 3. Then add a layer of jittered points with `geom_jitter()`.

```
# Type and view your answer:
q4 <- "YOUR ANSWER HERE"

q4
```

- Adapt your answer to question 4 to make the points more transparent and change the width of the jitter to an appropriate

value.


```
# Type and view your answer:
q5 <- "YOUR ANSWER HERE"

q5
```

SPECIFY VALUES FOR WIDTH AND ALPHA

- Building on the boxplot of life expectancy per continent from the previous example, add the `labs()` function to edit text on your plot.
- set the plot title to "Variation in life expectancy across continents (1952-2007)"
- change the x axis label to "Continent", and
- change the y axis label to "Life expectancy (years)".
- Using the boxplot you made in question 5, use the `labs()` function to edit text on your plot. Set the plot title to "Variation in life expectancy across continents (1952-2007)", change the x axis label to "Continent", and the y axis label to "Life expectancy (years)".

```
# Type and view your answer:
q6 <- "YOUR ANSWER HERE"

q6
```

Wrap up

Side-by-side boxplots provide us with a way to compare the distribution of a continuous variable across multiple values of another variable. One can see where the median falls across the different groups by comparing the solid lines in the center of the boxes.

To study the spread of a continuous variable within one of the boxes, look at both the length of the box and also how far the whiskers extend from either end of the box. Outliers are even more easily identified when looking at a boxplot than when looking at a histogram as they are marked with distinct points.

Learning Outcomes

1. You can plot a boxplot to visualize the distribution of continuous data using **`geom_boxplot()`**.
2. You can reorder side-by-side boxplots with the **`reorder()`** function.

3. You can add a layer of individual data points on a bloxplot using **`geom_jitter()`**.

Contributors

The following team members contributed to this lesson:

JOY VAZ

R Developer and Instructor, the GRAPH Network
Loves doing science and teaching science

ADMIN TEAM

GRAPH Courses Administration Team
The GRAPH Courses team is building epidemiological training courses to enhance disease surveillance and data science for public health across the globe

References

Some material in this lesson was adapted from the following sources:

- Ismay, Chester, and Albert Y. Kim. 2022. *A ModernDive into R and the Tidyverse*. https://moderndive.com/.