DTP assignment 2

Ma Lok Sum, Zoe

2022-08-17

Get the data

```
library(readr)
ashes <- read_csv("ashes.csv")
```

```
## Rows: 26 Columns: 13
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (13): batter, team, role, Test 1, Innings 1, Test 1, Innings 2, Test 2, ...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
ashes
```

```
## # A tibble: 26 x 13
##    batter   team   role  Test ~1 Test ~2 Test ~3 Test ~4 Test ~5 Test ~6 Test ~7
##    <chr>    <chr>  <chr> <chr>   <chr>   <chr>   <chr>   <chr>   <chr>   <chr>
##  1 Ali      Engla~ allr~ Battin~ Battin~ Battin~ Battin~ Battin~ Battin~ Battin~
##  2 Anderson Engli~ bowl  Battin~ Battin~ Battin~ Battin~ Battin~ Battin~ Battin~
##  3 Bairstow Engla~ wick~ Battin~ Battin~ Battin~ Battin~ Battin~ Battin~ Battin~
##  4 Ball     Engla~ bowl  Battin~ Battin~ Battin~ Battin~ Battin~ Battin~ Battin~
##  5 Bancroft Austr~ bat   Battin~ Battin~ Battin~ Battin~ Battin~ Battin~ Battin~
##  6 Bird     Austr~ bowl  Battin~ Battin~ Battin~ Battin~ Battin~ Battin~ Battin~
##  7 Broad    Engla~ bowl~ Battin~ Battin~ Battin~ Battin~ Battin~ Battin~ Battin~
##  8 Cook     Engla~ bat   Battin~ Battin~ Battin~ Battin~ Battin~ Battin~ Battin~
##  9 Crane    Engla~ bowl  Battin~ Battin~ Battin~ Battin~ Battin~ Battin~ Battin~
## 10 Cummins  Austr~ bowl  Battin~ Battin~ Battin~ Battin~ Battin~ Battin~ Battin~
## # ... with 16 more rows, 3 more variables: `Test 4, Innings 2` <chr>,
## #   `Test 5, Innings 1` <chr>, `Test 5, Innings 2` <chr>, and abbreviated
## #   variable names 1: `Test 1, Innings 1`, 2: `Test 1, Innings 2`,
## #   3: `Test 2, Innings 1`, 4: `Test 2, Innings 2`, 5: `Test 3, Innings 1`,
## #   6: `Test 3, Innings 2`, 7: `Test 4, Innings 1`
## # i Use `print(n = ...)` to see more rows, and `colnames()` to see all variable names
```

Question One: Reading and Cleaning

```
# 1(a) convert it into a long format
library(tidyr)
ashes_long <- gather(ashes, key = "innings", value = "scores", `Test 1, Innings 1`,
                     `Test 1, Innings 2`,`Test 2, Innings 1`,`Test 2, Innings 2`,
                     `Test 3, Innings 1`,`Test 3, Innings 2`,`Test 4, Innings 1`,
```

```
                    `Test 4, Innings 2`,`Test 5, Innings 1`,
                    `Test 5, Innings 2`)
# visualize it again
ashes_long
```

```
## # A tibble: 260 x 5
##    batter   team      role         innings           scores
##    <chr>    <chr>     <chr>        <chr>             <chr>
##  1 Ali      England   allrounder   Test 1, Innings 1 Batting at number 6, score~
##  2 Anderson English   bowl         Test 1, Innings 1 Batting at number 11, scor~
##  3 Bairstow England   wicketkeeper Test 1, Innings 1 Batting at number 7, score~
##  4 Ball     England   bowl         Test 1, Innings 1 Batting at number 10, scor~
##  5 Bancroft Australia bat          Test 1, Innings 1 Batting at number 1, score~
##  6 Bird     Australia bowl         Test 1, Innings 1 Batting at number NA, scor~
##  7 Broad    England   bowler       Test 1, Innings 1 Batting at number 9, score~
##  8 Cook     England   bat          Test 1, Innings 1 Batting at number 1, score~
##  9 Crane    England   bowl         Test 1, Innings 1 Batting at number NA, scor~
## 10 Cummins  Australia bowl         Test 1, Innings 1 Batting at number 9, score~
## # ... with 250 more rows
## # i Use `print(n = ...)` to see more rows
```

```
# create new columns for each of the following for each player innings:
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
```

```
library(stringr)

details <- str_match(ashes_long$scores,
'Batting at number (\\d+), scored (\\d+) runs from (\\d+) balls including (\\d+) fours and (\\d+) sixes

ashes_long$Batting <- details[,2]
ashes_long$Scores <-details[,3]
ashes_long$Balls <- details[,4]
ashes_long[5]<-NULL
ashes_long
```

```
## # A tibble: 260 x 7
##    batter   team      role         innings           Batting Scores Balls
##    <chr>    <chr>     <chr>        <chr>             <chr>   <chr>  <chr>
##  1 Ali      England   allrounder   Test 1, Innings 1 6       38     102
##  2 Anderson English   bowl         Test 1, Innings 1 11      5      9
##  3 Bairstow England   wicketkeeper Test 1, Innings 1 7       9      24
##  4 Ball     England   bowl         Test 1, Innings 1 10      14     11
##  5 Bancroft Australia bat          Test 1, Innings 1 1       5      19
##  6 Bird     Australia bowl         Test 1, Innings 1 <NA>    <NA>   <NA>
##  7 Broad    England   bowler       Test 1, Innings 1 9       20     32
```

```
##  8 Cook     England   bat          Test 1, Innings 1 1       2      10
##  9 Crane    England   bowl         Test 1, Innings 1 <NA>    <NA>   <NA>
## 10 Cummins  Australia bowl         Test 1, Innings 1 9       42     120
## # ... with 250 more rows
## # i Use `print(n = ...)` to see more rows
```

```
# 1(b) tame the data
library('tidyverse')
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --
## v ggplot2 3.3.6     v purrr   0.3.4
## v tibble  3.1.8     v forcats 0.5.1
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(forcats)
ashes_long$team <- as.factor(ashes_long$team)
ashes_long$role <- as.factor(ashes_long$role)
ashes_long$Batting <- as.integer(ashes_long$Batting)
ashes_long$Scores <- as.integer(ashes_long$Scores)
ashes_long$Balls <- as.integer(ashes_long$Balls)
ashes_long
```

```
## # A tibble: 260 x 7
##    batter   team      role         innings           Batting Scores Balls
##    <chr>    <fct>     <fct>        <chr>               <int>  <int> <int>
##  1 Ali      England   allrounder   Test 1, Innings 1       6     38   102
##  2 Anderson English   bowl         Test 1, Innings 1      11      5     9
##  3 Bairstow England   wicketkeeper Test 1, Innings 1       7      9    24
##  4 Ball     England   bowl         Test 1, Innings 1      10     14    11
##  5 Bancroft Australia bat          Test 1, Innings 1       1      5    19
##  6 Bird     Australia bowl         Test 1, Innings 1      NA     NA    NA
##  7 Broad    England   bowler       Test 1, Innings 1       9     20    32
##  8 Cook     England   bat          Test 1, Innings 1       1      2    10
##  9 Crane    England   bowl         Test 1, Innings 1      NA     NA    NA
## 10 Cummins  Australia bowl         Test 1, Innings 1       9     42   120
## # ... with 250 more rows
## # i Use `print(n = ...)` to see more rows
```

```
# 1(c) clean the data
ashes_long$role <- fct_recode(ashes_long$role,
           `all-rounder` = "all rounder",
           `all-rounder` = "allrounder",
           bowler = "bowl",
           batter = "bat",
           batter = "batsman",
           batter = "batting")

ashes_long$team <- fct_recode(ashes_long$team,
           England = "English")

ashes_long
```

```
## # A tibble: 260 x 7
##    batter   team      role         innings           Batting Scores Balls
```

```
##    <chr>    <fct>     <fct>        <chr>              <int>  <int> <int>
##  1 Ali      England   all-rounder  Test 1, Innings 1      6     38   102
##  2 Anderson England   bowler       Test 1, Innings 1     11      5     9
##  3 Bairstow England   wicketkeeper Test 1, Innings 1      7      9    24
##  4 Ball     England   bowler       Test 1, Innings 1     10     14    11
##  5 Bancroft Australia batter       Test 1, Innings 1      1      5    19
##  6 Bird     Australia bowler       Test 1, Innings 1     NA     NA    NA
##  7 Broad    England   bowler       Test 1, Innings 1      9     20    32
##  8 Cook     England   batter       Test 1, Innings 1      1      2    10
##  9 Crane    England   bowler       Test 1, Innings 1     NA     NA    NA
## 10 Cummins  Australia bowler       Test 1, Innings 1      9     42   120
## # ... with 250 more rows
## # i Use `print(n = ...)` to see more rows
```

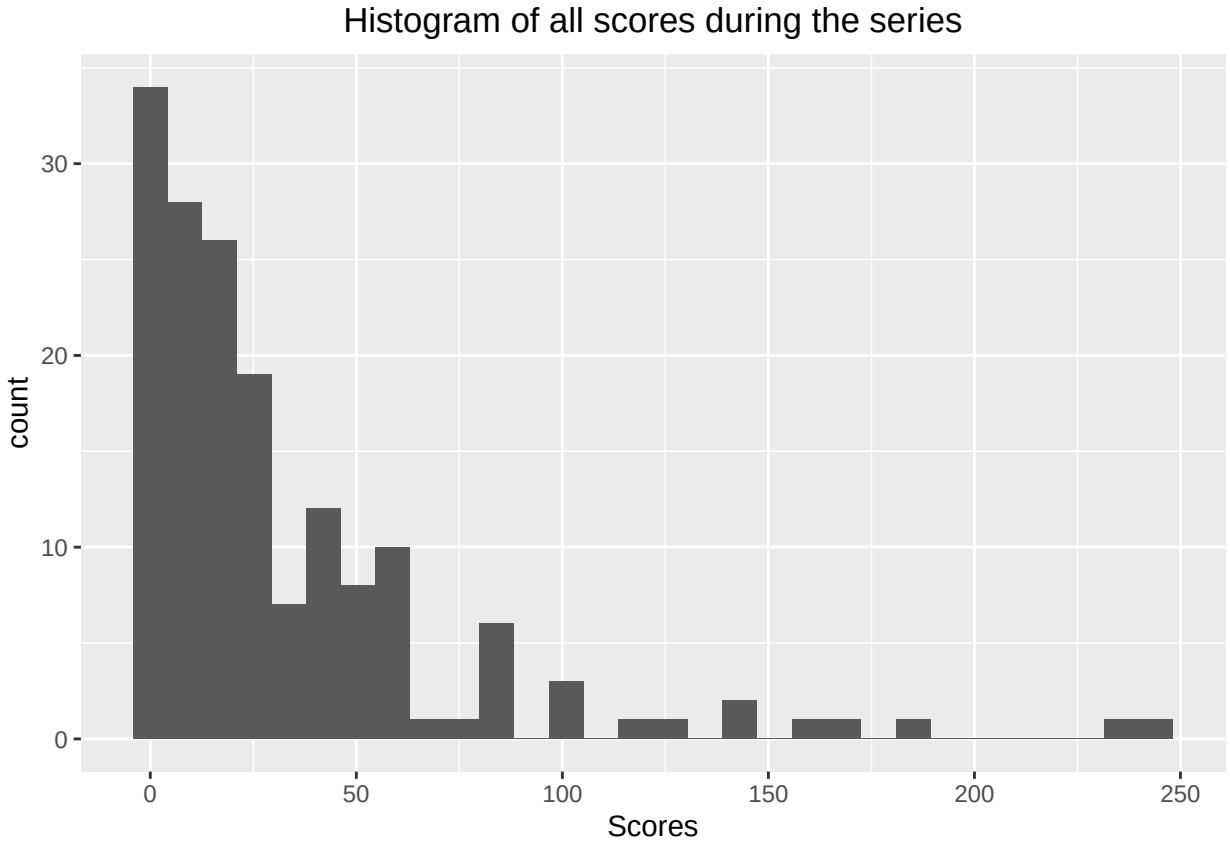
Question two: Univariate Analysis

```
# 2(a) produce a histogram

library(ggplot2)
ggplot(ashes_long,aes(x = Scores)) +
  geom_histogram() +
  ggtitle("Histogram of all scores during the series") +
  theme(plot.title = element_text(hjust = 0.5))
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

```
## Warning: Removed 96 rows containing non-finite values (stat_bin).
```



2(b)

It is a right-skewed distribution of scores since the peak of the histogram veers to the left. It has a tail on the right side, and the location spread is domain left. This histogram has outliers.

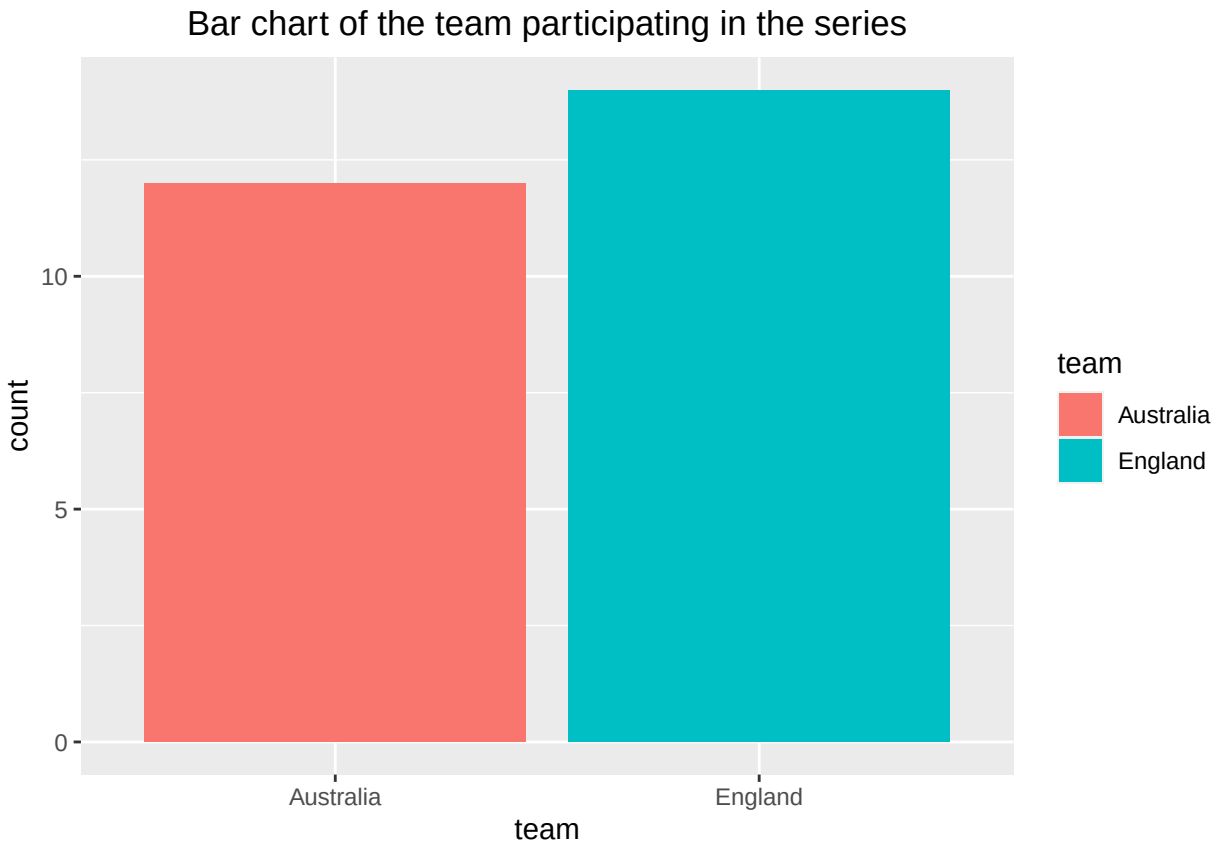
```
# 2(c) produce a bar chart
ashes_long_1 <- ashes_long%>%group_by(batter,team)%>%summarize(count=n())
```

```
## `summarise()` has grouped output by 'batter'. You can override using the
## `.groups` argument.
```

```
ashes_long_1
```

```
## # A tibble: 26 x 3
## # Groups:   batter [26]
##    batter   team      count
##    <chr>    <fct>     <int>
##  1 Ali      England      10
##  2 Anderson England      10
##  3 Bairstow England      10
##  4 Ball     England      10
##  5 Bancroft Australia    10
##  6 Bird     Australia    10
##  7 Broad    England      10
##  8 Cook     England      10
##  9 Crane    England      10
## 10 Cummins  Australia    10
## # ... with 16 more rows
## # i Use `print(n = ...)` to see more rows
```

```
ggplot(ashes_long_1, aes (x=team, fill = team)) + geom_bar() +
  ggtitle("Bar chart of the team participating in the series") +
  theme(plot.title = element_text(hjust = 0.5))
```

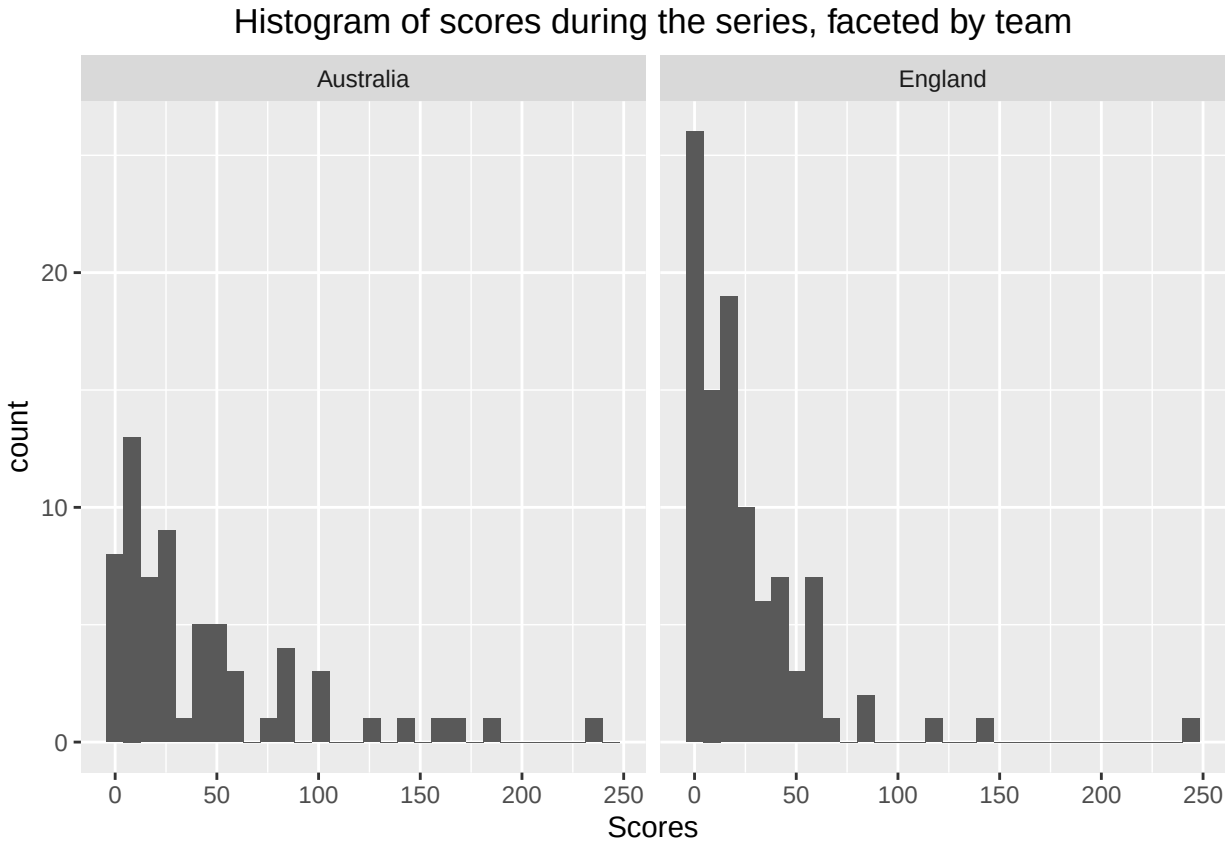


Question three: Scores for each team

```
# 3(a) produce histogram of scores during the series, faceted by team
ggplot(ashes_long,aes(x = Scores)) + geom_histogram() + facet_wrap(~team) +
  ggtitle("Histogram of scores during the series, faceted by team") +
  theme(plot.title = element_text(hjust = 0.5))
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

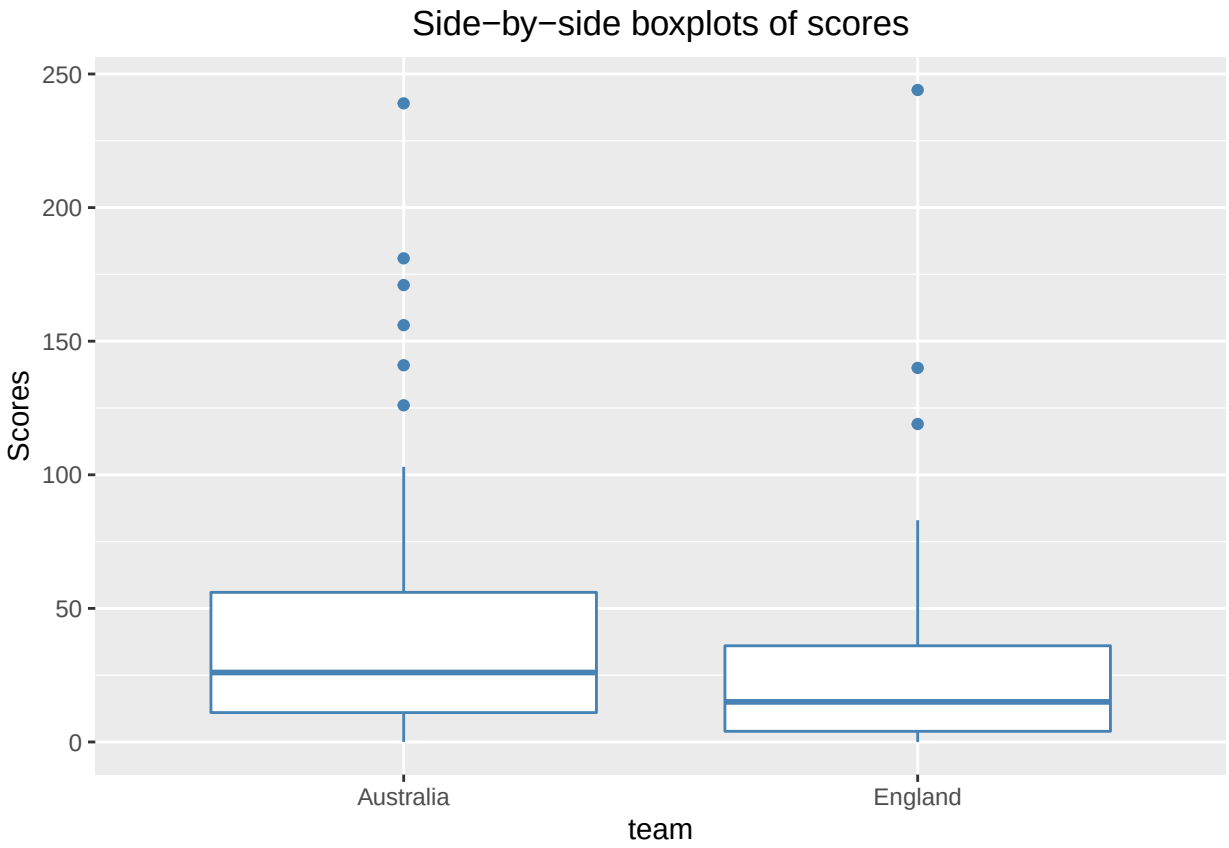
```
## Warning: Removed 96 rows containing non-finite values (stat_bin).
```




```
# 3(b) produce side-by-side boxplots of scores

ggplot(ashes_long,aes(x = team, y = Scores)) +
  geom_boxplot(col = "steelblue") +
  ggtitle("Side-by-side boxplots of scores") +
  theme(plot.title = element_text(hjust = 0.5))
```

```
## Warning: Removed 96 rows containing non-finite values (stat_boxplot).
```



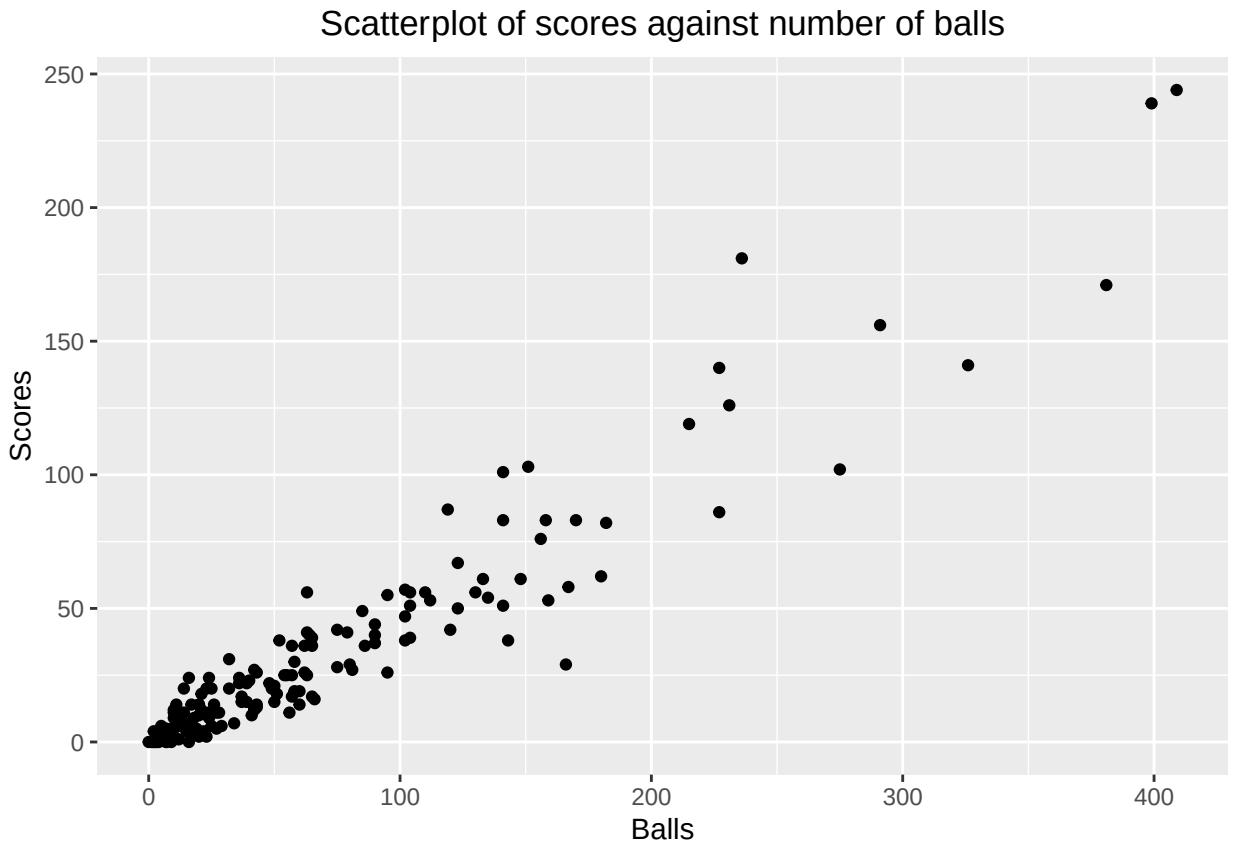
3(c)

Considering the shape and location of the boxplot in Australia, it is right-skew, with a long tail to the right (high values), as shown by the longer right whisker and the fact that the right part of the box (median to upper quartile) is longer than the left. For the England part, it has the same comments but seems only does not perform well as Australia. The more spread the boxplot graph is in Australia because the interquartile range is larger than in England. Both have potential outliers. Australia has had a higher average score because the median of the Australia team is higher than England.

Question four: Scoring rates

```
# 4(a) produce a scatterplot of scores against number of balls
ggplot(ashes_long,aes(x = Balls, y = Scores)) + geom_point()+
  ggtitle("Scatterplot of scores against number of balls") +
  theme(plot.title = element_text(hjust = 0.5))
```

```
## Warning: Removed 96 rows containing missing values (geom_point).
```

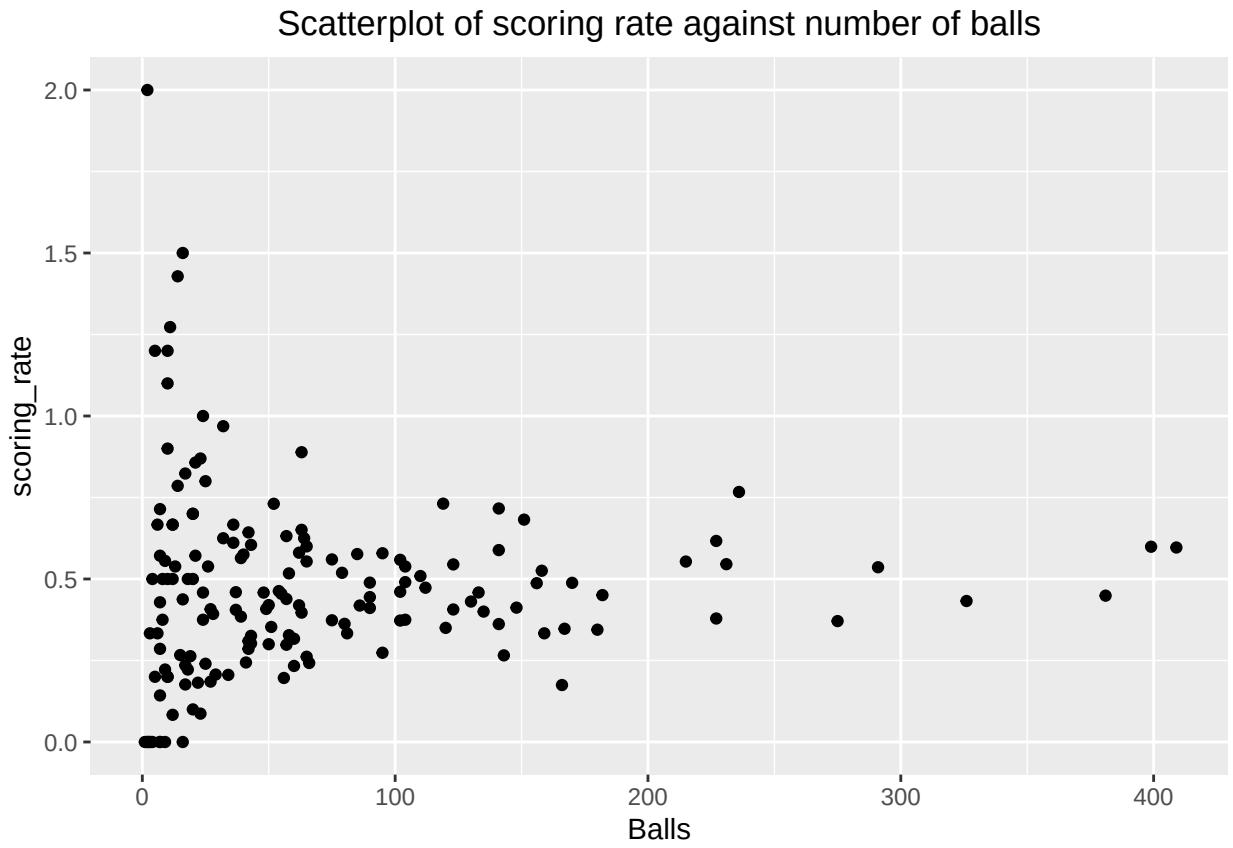


4(b)

The association between score and number of balls is linear. It is a moderate positive correlation, which has potential outliers. Therefore, players who face more balls are likely to score more.

```
# 4(c) compute a new variable and produce a scatterplot
ashes_long$scoring_rate <- ashes_long$Scores/ ashes_long$Balls
ggplot(ashes_long,aes(x = Balls, y = scoring_rate)) + geom_point() +
  ggtitle("Scatterplot of scoring rate against number of balls") +
  theme(plot.title = element_text(hjust = 0.5))
```

```
## Warning: Removed 97 rows containing missing values (geom_point).
```



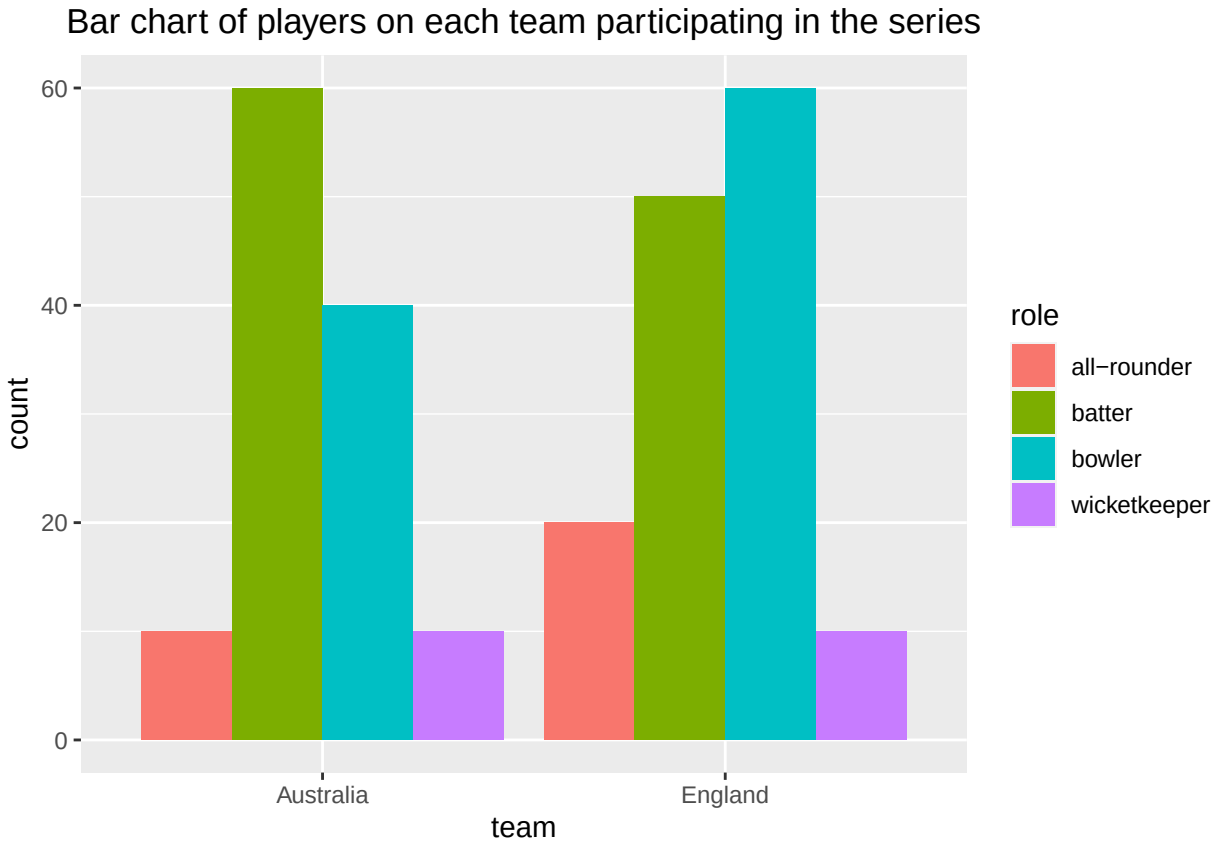
4(d)

There is a weak positive relationship between scoring rate and number of balls. Players who face more balls are likely to score runs more quickly.

Question five: Teams' role

```
# 5(a) produce a bar chart of the number of players on each team participating in the series

ggplot(ashes_long, aes(x = team, fill = role)) +
  geom_bar(position="dodge") +
  ggtitle("Bar chart of players on each team participating in the series") +
  theme(plot.title = element_text(hjust = 0.5))
```




```
# 5(b) produce a contingency table

table_3 <- prop.table(table(ashes_long$team, ashes_long$role), margin = 1)

table_3 = round(table_3, 3)

knitr::kable(
  table_3,
  caption = "a contingency table of proportion of players from each team who play in each particular ro
```

Table 1: a contingency table of proportion of players from each team who play in each particular role

	all-rounder	batter	bowler	wicketkeeper
Australia	0.083	0.500	0.333	0.083
England	0.143	0.357	0.429	0.071

5(c)

Australia had a larger proportion of batters. England had a larger proportion of bowlers.