Reading performance analysis

Report By: Prakhar Gurawa

Introduction:
The Organization for Economic Co-operation and Development (OECD) is an intergovernmental economic organization to stimulate economic progress and world trade has also provided the 2018 PISA results (Programme for International Student Assessment) which will help us simulate and visualize the reading performance of girls and boys through different countries over a period of time.

About the Dataset:
The dataset contains the reading performance values for boys and girls separately, differentiated by their countries over a period of 2000 to 2018. The columns named are as follows: Location (Country code), Indicator (Performance measure name), Subject (Gender), Measure, Frequency, Time(Year), Value(Score) and Flag Code.

Part 1: Analysis of provided graphic
The provided graphic helps the viewers visualize the progress of reading performance ability which is judge using the reading ability score represented on the y-axis for different countries represented on the x-axis, separately viewed for both the genders.

- **Aesthetic mappings:** aesthetics means how the data values are mapped to the features of visual space, here different shapes are used to represent different genders (dot to represent boys and a rhombus for girls), with emphasis on mean and Ireland using a separate color mapping for both (as the visual tries to create an emphasis on these two), the plot used is here dot plot which gives a much clear and less crowded look as compared to bar graphs or stacked bar charts, etc.
- **Axes:** The x-axis represents the different countries with a total of 40 countries and 1 value for the average values, on the other side the y-axis represents the reading performance measure. The X-axis is categorical and Y-axis is continuous ranging from 340 to 560, both lead to an ordered increasing data.
- **Gridlines:** Vertical gridlines are limited by values of the boys, connector gridlines are used between girl and boy for a clear and better visual which helps the viewer to compare the performance of girls and boys of a country better, the vertical grid lines are separated by different countries, horizontal grid line are separated by a constant reading measure value difference of 20. A white grid line is used over the light blue background which gives a non contrasting visual with a smooth and elegant touch.
- **Legend:** Present on the bottom left corner of visual, informs critical information of shape used for both gender and needs to be at a position where it can be viewed before the actual visual, I feel it would be better to keep it on the top center
- **Background:** The background color used is #e2edf3 which is not a loud color and thus a good choice for background color. Also, it blends well with the color of points and the white gridlines to give a smooth touch
- **Colour:** #e2edf3 (Catskill White) for background color, #406d89 (Ming) is used for the points (boys and girls), separate color #000000 (Black) is used to represent the average values with #ea1f25 (Alizarin Crimson) to represent Ireland's values. Overall the colors used are too bold and thus provides an elegant visual.
- **Other Visual Details:** Orientation of x-axis is tilted, the y axis labels exist over the gridlines with spaces between labels. The y axis starts from 340 and not from 0 and the values range from 340 to 560. the x-axis is not sorted on the basis of countries name but countries reading performance value: data aspects

Part 2: Reproducing the provided visual

In this part, we will replicate the visual provided by OECD on the performance of reading ability of boys and girls of 40 distinct countries for the year 2018. The steps followed are as follows:

- **Importing relevant libraries:** libraries like ggplot, dplyr, grid, and repr are extensively used for this part. An additional new library named country code is used to convert country codes to country names, which reduces our manual work.
- **Ignoring irrelevant rows and columns:** All other columns except location, subject, time, and value are dropped as are not required in this task. After this, an additional filter to select only 2018 data of only boys and girls is applied which leads to our final data frames which will be used to create visuals.
- **Finding the country ordering for visual:** The data frame is bifurcated in a data frame with only boys and one with girls, after which both are sorted based on the value (reading parameter). The ordering of countries from the boy's data frame is used as the final ordering in the visualization. This leads to an ordered data as shown in the original visual.
- **Generating country names:** An external library country code is used to generate countries' names from country codes, which gives a warning for the "OAVG" which is manually written as "OECD - Average" from the null value generated by the library.
- **Creating the visual:** To replicate the original visual the following steps are taken care off:
    1. **Dimensions:** The generated figure has been fixed height of 5.5 and a width of 11 to give a similar dimensional look.
    2. **Aesthetics:** The top sections contain the heading, subheading, and caption adjusted accordingly by fixed hjust and vjust values. A thin blue line is generated just above the graph using annotate(). The boys are represented by the dots and girls using the rhombus shape which are manually specified shapes using scale_shape_manual().
    3. **Axis:** The x-axis contains all the countries with the average in between as ordered using the list generated by the boy's data frame. The text at the x-axis is tilted at an angle of 45 degrees with a fixed hjust and vjust. The colors for special cases of Ireland and average are changed accordingly. The y-axis represents the reading performance ranging from 340 to 560 having breaks every 20 units. The y-axis is pushed inside the graph by extending the x-axis by a unit of 1.5.
    4. **Panel:** The panel filled with color #e2edf3, which also has some free space at the bottom generated by adding a line at the x-axis bottom of the same color as that of the background.
    5. **Gridlines:** The horizontal gridlines are of white color exists every 20 using gap, whereas vertical gridlines are created using a hack by creating white lines for every country from bottom to boys coordinate using geom_segment() and a #bbc9d4 colored lines from boys coordinate to girls using geom_line().
    6. **Colors:** The colors used are as follows #e2edf3 for background, white for gridlines,#406d89 for coordinates of boys and girls, black to represent average value coordinate, and red for Ireland. Whereas for special cases of average and Ireland's girls coordinate the shape used is different from that of normal girls coordinates.
    7. **Legend:** The legend is kept horizontal using legend.direction = "horizontal", with title manually removed. As the legend was creating some size and orientation issues in visual I have removed it for now.
    8. **Other aspects:** The colors are replicated using hex code generators from the image to give a similar color feel as that of the original image. The colors are the same for heading, coordinates, background, etc. A number of hacks are used in code just to give as accurate a visual as possible like filling the background color inside rhombus for normal girl's coordinates, otherwise, the gridlines were visible inside that area.
    9. **Code**: The code for this part is present at the end of this sheet, after code for part 3.

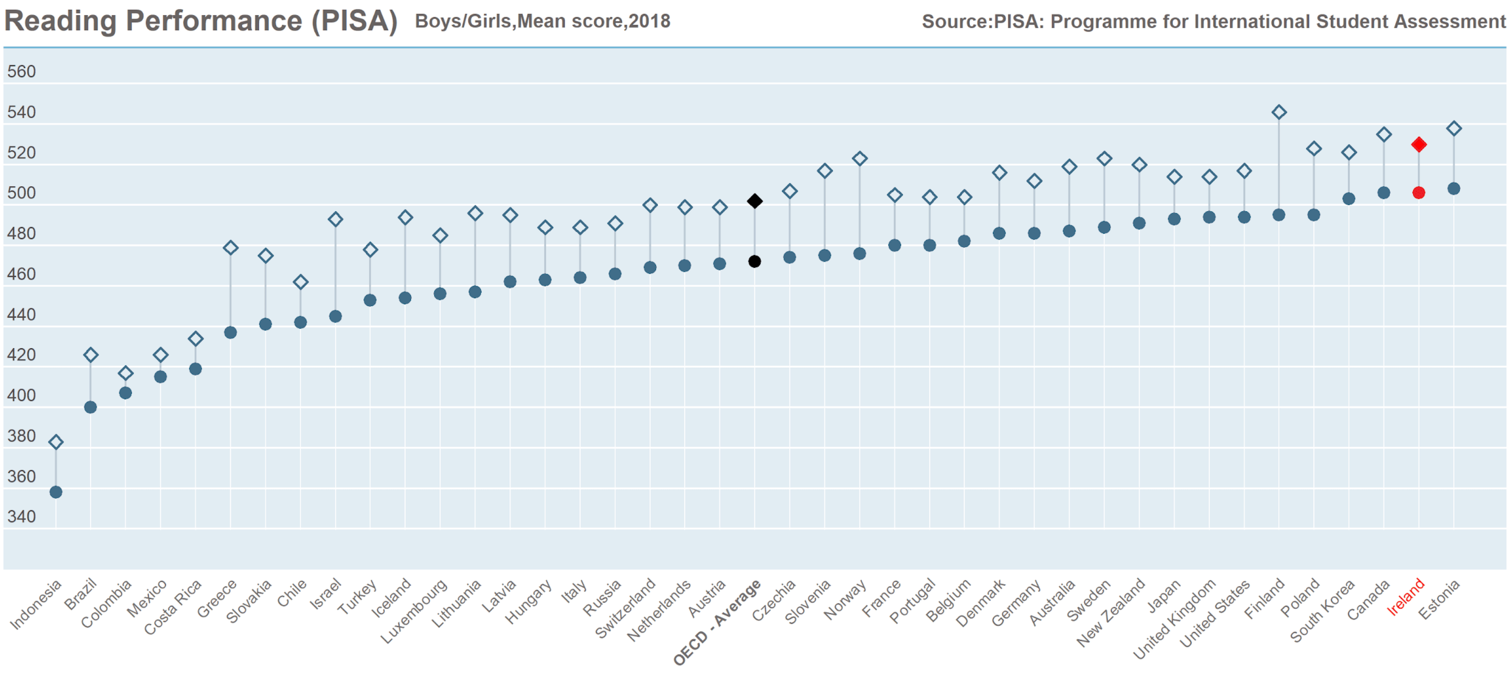


Fig 1. The generated figure for Part 2. The image is also available as Part2.png

Part3: Visualisation of reading performance over time

In this part we have considered 3 countries Ireland,Austria,Indonesia and Average scores to represent the reading perforce of boys and girls separately over time. The countries chosen to show the difference better as Ireland's scores are on the higher side, whereas Indonesia comes on lower with Austria in mid.

- The plots for boys and girls are separated for better understanding and less confusion among the viewers. Variation of both on the same plot could create a lot of confusion and vagueness.
- The x-axis represents the time in the form of year with a grid line for every 5 year gap.
- The y-axis represents the reading scores ranging between 340 to 540.
- The look for this visual is kept similar to the previous plot with the same background,heading type, heading line and y axis text above the gridline.
- It can be inferred the Indonesia scores are much lower than that of other countries and median scores.
- Austria's score almost goes along with that of global median scores.
- The performance of girls for all the countries as well as on average is better than that of boys.
- The legend is kept on top right corner for better visibility representing the four sections and their colours.
- The two plots for boys and colour are arrange side by side using grid.arrange().
- The legend for the left plot is removed as the legend for both the plot is similar and will be redundant otherwise. Legend's background is kept as white to blend well with background and orientation is kept horizontal.
- Ticks and titles for the axis are removed for both the sections.Also minor gridlines are removed.
- A major drop in the performance of Ireland's boys can be seen between the period of 2005 and 2010, which can be seen for the girls too.

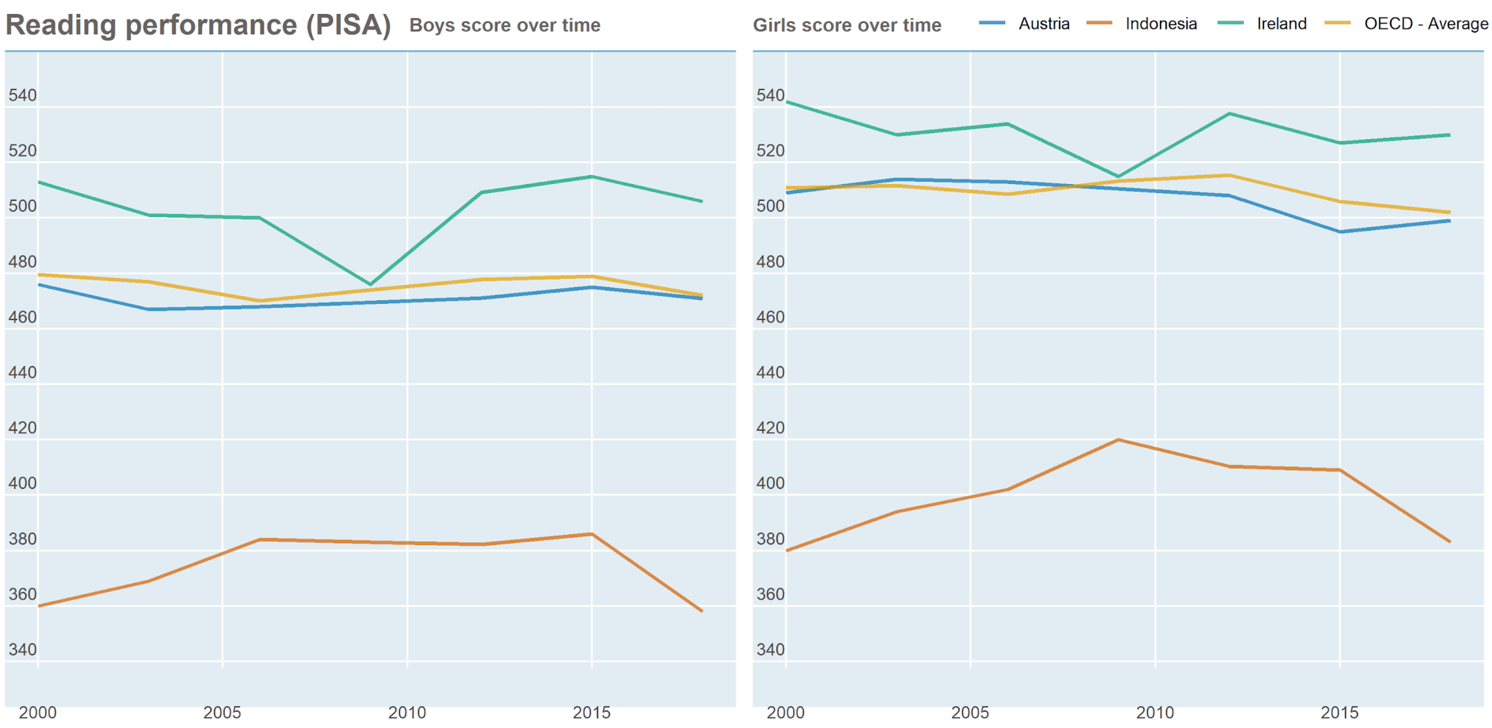


Fig 1. The generated figure for Part 3. The image is also available as Part3.png

Code for Part 3:

Part 3: Performance over years

Importing relevant libraries

```
library(readr)

## Warning: package 'readr' was built under R version 4.0.4

library(ggplot2)

## Warning: package 'ggplot2' was built under R version 4.0.4

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(tidyr)
library(lubridate)
```

```
##
## Attaching package: 'lubridate'

## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union

library(scales)

##
## Attaching package: 'scales'

## The following object is masked from 'package:readr':
##
##     col_factor

library(kableExtra)

## Warning: package 'kableExtra' was built under R version 4.0.4

##
## Attaching package: 'kableExtra'

## The following object is masked from 'package:dplyr':
##
##     group_rows

library(countrycode)

## Warning: package 'countrycode' was built under R version 4.0.5

library(grid)
library(repr)

## Warning: package 'repr' was built under R version 4.0.5

library(gridExtra)

## Warning: package 'gridExtra' was built under R version 4.0.4

##
## Attaching package: 'gridExtra'

## The following object is masked from 'package:dplyr':
##
##     combine
```

Importing dataset

```
df <- read.csv("OECD_PISA.csv") %>% select("ï..LOCATION","SUBJECT","TIME","Value")
colnames(df)<-c("Location","Subject","Time","Value")
head(df)

##   Location Subject Time  Value
## 1      AUS     BOY 2000 513.00
## 2      AUS     BOY 2003 506.00
## 3      AUS     BOY 2006 495.00
## 4      AUS     BOY 2009 496.00
## 5      AUS     BOY 2012 495.09
## 6      AUS     BOY 2015 487.00
```

Removing unnecessary rows and columsn

```
df2 <- df %>% filter(Location=="IDN"|Location=="IRL"|Location=="AUT"|Location=="OAVG")
df_boys <- df2 %>% filter(Subject=="BOY") %>% select("Location","Time","Subject","Value")
%>% arrange(Location)
df_girls <- df2 %>% filter(Subject=="GIRL") %>%
select("Location","Time","Subject","Value")%>% arrange(Location)
head(df_boys)
```

```
##   Location Time Subject   Value
## 1      AUT 2000     BOY 476.000
## 2      AUT 2003     BOY 467.000
## 3      AUT 2006     BOY 468.000
## 4      AUT 2012     BOY 471.093
## 5      AUT 2015     BOY 475.000
## 6      AUT 2018     BOY 471.000
```

```
head(df_girls)
```

```
##   Location Time Subject   Value
## 1      AUT 2000    GIRL 509.000
## 2      AUT 2003    GIRL 514.000
## 3      AUT 2006    GIRL 513.000
## 4      AUT 2012    GIRL 508.021
## 5      AUT 2015    GIRL 495.000
## 6      AUT 2018    GIRL 499.000
```

Plot

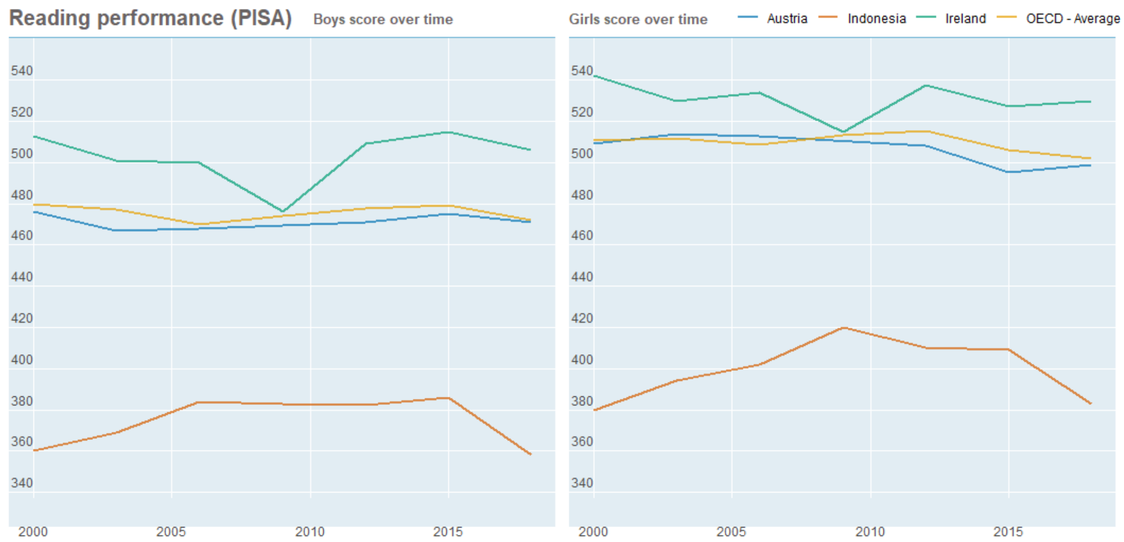
```
# plot 1 : trends of boys reading score over time for the specific countries
gboys<-ggplot(df_boys,aes(Time,Value,color=Location)) +
  geom_line(size = 1, alpha = 0.7) +
  scale_color_manual(values = c("#0072b2", "#D55E00", "#009e73", "#E69F00"),name = NULL)
+
  annotate(geom = 'segment', y = Inf, yend = Inf, color = '#6eb4d5', x = -Inf, xend =
Inf, size = 1)+ # creates a blue top margin
  labs(title = "Reading performance (PISA)",subtitle = "Boys score over time") +
  theme(
      axis.text.y = element_text(vjust = -0.5,margin = margin(r = -20)), # The text of y
axis exists above the gridlines
      axis.title.x = element_blank(),
      axis.title.y = element_blank(),
      axis.ticks.y = element_blank(),
      axis.ticks.x = element_blank(),
      panel.background = element_rect(fill = "#e2edf3"),
      axis.line.x.bottom = element_line("#e2edf3", size= 9.7), # Hack : a background
colour line at bottom to hide lower gridlines
      plot.title = element_text(size=15,color = "#696363",vjust = -4,face="bold"),
      plot.subtitle = element_text(size=10,hjust = 0.75,vjust = 2,color =
"#696363",face="bold"),
      legend.position = "none", # removing legend for left visual
      panel.grid.minor.x = element_blank(),
      axis.text.x = element_text(vjust = -2)
  ) +
  scale_y_continuous(breaks = seq(340, 540, by = 20),minor_breaks = seq(0, 20, 10),limits
= c(341, 550))

# plot 2 : trends of girls reading score over time for the specific countries
```

```
ggirls<-ggplot(df_girls,aes(Time,Value,color=Location)) +
  geom_line(size = 1, alpha = 0.7) +
  scale_color_manual(values = c("#0072b2", "#D55E00", "#009e73", "#E69F00"),labels =
c("Austria", "Indonesia","Ireland","OECD - Average")) +
  annotate(geom = 'segment', y = Inf, yend = Inf, color = '#6eb4d5', x = -Inf, xend =
Inf, size = 1)+ # creates a blue top margin
  labs(title = " ",subtitle = "Girls score over time") +
  theme(
      axis.text.y = element_text(vjust = -0.5,margin = margin(r = -20)), # The text of y
axis exists above the gridlines
      axis.title.x = element_blank(),
      axis.title.y = element_blank(),
      axis.ticks.y = element_blank(),
      axis.ticks.x = element_blank(),
      panel.background = element_rect(fill = "#e2edf3"),
      axis.line.x.bottom = element_line("#e2edf3", size= 9.7), # Hack : a background
colour line at bottom to hide lower gridlines
      plot.title = element_text(size=15,color = "#696363",vjust = -4,face="bold"),
      plot.subtitle = element_text(size=10,hjust = 0,vjust = 2,color =
"#696363",face="bold"),
      legend.direction = "horizontal",
      legend.key = element_rect(fill = "white", color = NA),
      legend.position = c(0.65, 1.043), # manually specifying legend location
      legend.title = element_blank(),
      panel.grid.minor.x = element_blank(),
      axis.text.x = element_text(vjust = -2)
  ) +
  scale_y_continuous(breaks = seq(340, 540, by = 20),minor_breaks = seq(0, 20, 10),limits
= c(341, 550))



g<-grid.arrange(gboys, ggirls, nrow = 1)
```



```
ggsave(plot = g, filename = "Part3.png")
```

Code for Part 2:

Part 2 : Replicating the Visual

Importing relevant libraries

```
library(readr)

## Warning: package 'readr' was built under R version 4.0.4

library(ggplot2)

## Warning: package 'ggplot2' was built under R version 4.0.4

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(tidyr)
library(lubridate)

##
## Attaching package: 'lubridate'

## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union

library(scales)

##
## Attaching package: 'scales'

## The following object is masked from 'package:readr':
##
##     col_factor

library(kableExtra)

## Warning: package 'kableExtra' was built under R version 4.0.4

##
## Attaching package: 'kableExtra'

## The following object is masked from 'package:dplyr':
##
##     group_rows

library(countrycode)
```

```
## Warning: package 'countrycode' was built under R version 4.0.5
```

```r
library(grid)
library(repr)
```

```
## Warning: package 'repr' was built under R version 4.0.5
```

Reading the original dataset

```r
df <- read.csv("OECD_PISA.csv") %>% select("ï..LOCATION","SUBJECT","TIME","Value")
colnames(df)<-c("Location","Subject","Time","Value")
head(df)
```

```
##   Location Subject Time  Value
## 1      AUS     BOY 2000 513.00
## 2      AUS     BOY 2003 506.00
## 3      AUS     BOY 2006 495.00
## 4      AUS     BOY 2009 496.00
## 5      AUS     BOY 2012 495.09
## 6      AUS     BOY 2015 487.00
```

Filtering the relevant rows for this part

```r
df2 <- df %>% filter(Time==2018) %>% filter(Subject=="BOY"|Subject=="GIRL")
head(df2)
```

```
##   Location Subject Time Value
## 1      AUS     BOY 2018   487
## 2      AUS    GIRL 2018   519
## 3      AUT     BOY 2018   471
## 4      AUT    GIRL 2018   499
## 5      BEL     BOY 2018   482
## 6      BEL    GIRL 2018   504
```

Dividing dataset into boys and girls to find ordering of data

```r
df_boys <- df2 %>% filter(Subject=="BOY") %>% select("Location","Subject","Value") %>%
arrange(Value)
df_girls <- df2 %>% filter(Subject=="GIRL") %>% select("Location","Subject","Value")%>%
arrange(Value)
head(df_boys)
```

```
##   Location Subject Value
## 1      IDN     BOY   358
## 2      BRA     BOY   400
## 3      COL     BOY   407
## 4      MEX     BOY   415
## 5      CRI     BOY   419
## 6      GRC     BOY   437
```

```r
head(df_girls)
```

```
##   Location Subject Value
## 1      IDN    GIRL   383
## 2      COL    GIRL   417
## 3      MEX    GIRL   426
## 4      BRA    GIRL   426
## 5      CRI    GIRL   434
## 6      CHL    GIRL   462
```

```
# The plot orders in the flow of boys dataset
```

Finding the ordered country codes and country names

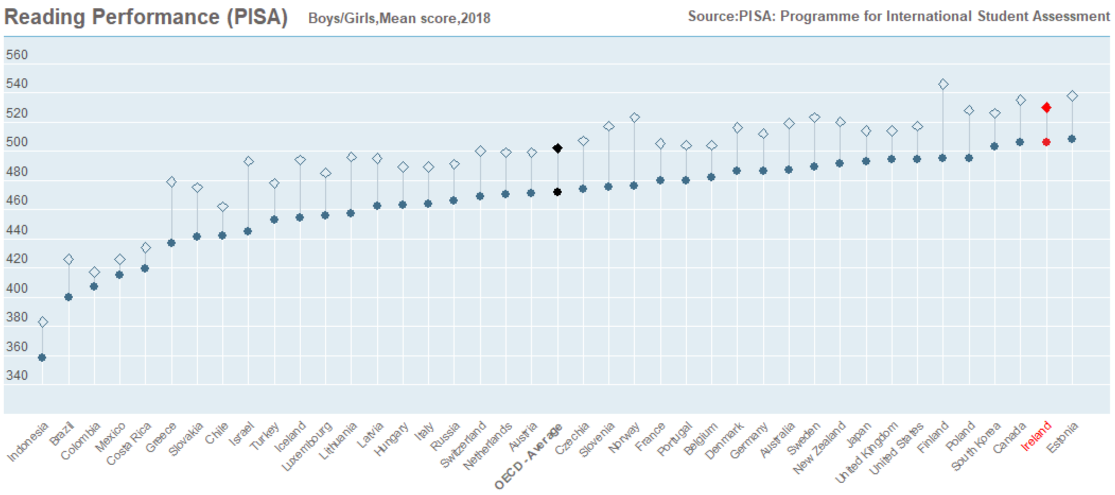
```
# the level of boys will be used in visual as it creates an increasing order
countrycodes <- df_boys$Location
countryfullnames <- countrycode(df_boys$Location,origin = 'iso3c', destination =
'country.name')
# warning: Some values were not matched unambiguously: OAVG
countryfullnames[is.na(countryfullnames)]='OECD - Average'
countryfullnames
```

```
##  [1] "Indonesia"      "Brazil"         "Colombia"       "Mexico"
##  [5] "Costa Rica"     "Greece"         "Slovakia"       "Chile"
##  [9] "Israel"         "Turkey"         "Iceland"        "Luxembourg"
## [13] "Lithuania"      "Latvia"         "Hungary"        "Italy"
## [17] "Russia"         "Switzerland"    "Netherlands"    "Austria"
## [21] "OECD - Average" "Czechia"        "Slovenia"       "Norway"
## [25] "France"         "Portugal"       "Belgium"        "Denmark"
## [29] "Germany"        "Australia"      "Sweden"         "New Zealand"
## [33] "Japan"          "United Kingdom" "United States"  "Finland"
## [37] "Poland"         "South Korea"    "Canada"         "Ireland"
## [41] "Estonia"
```

```
# creating a separate dataframe for this exercise
df3 <-df2
df3$Color <- ifelse(df3$Location == "IRL"|df3$Location == "OAVG",ifelse(df3$Location ==
"IRL","red","black"),"blue")
df3$shape <-ifelse(df3$Subject=="BOY","A",ifelse(df3$Location == "IRL"|df3$Location ==
"OAVG","B","C"))
df3$fill <- ifelse(df3$Subject=="BOY","A",ifelse(df3$Location ==
"IRL","B",ifelse(df3$Location == "OAVG","C","D")))

# Define plot title,subtile and caption
plot.title = "Reading Performance (PISA)"
plot.subtitle = "Boys/Girls,Mean score,2018"
plot.caption = "Source:PISA: Programme for International Student Assessment"

g <-ggplot(df3,aes(x=factor(Location,levels = countrycodes),
y=Value,colour=Color,shape=shape,fill=fill)) + #countrycodes represent country code
orders of boys dataset
  geom_segment(aes(xend = Location), yend = 0, colour="white", size=0.25) + # create a
white line till boys coordinate
  geom_line(aes(group = Location),colour = "#bbc9d4", size=0.5,aplha=0.7) + #creates a
coloured line from boys to girls coordinate
  geom_point(size=2,stroke = 1) +
  annotate(geom = 'segment', y = Inf, yend = Inf, color = '#6eb4d5', x = -Inf, xend =
Inf, size = 1)+ # creates a blue top margin
  labs(title = plot.title,subtitle = plot.subtitle,caption = plot.caption)+
    theme(
      axis.text.x = element_text(angle = 45, vjust = 0.8, hjust = 1,color =
ifelse(countryfullnames=="Ireland","Red","#716d6d"),face = ifelse(countryfullnames=="OECD
- Average","bold","plain"),size = 8), # specific colours for special cases like Ireland
and OECD Average
      axis.text.y = element_text(vjust = -0.5,margin = margin(r = -20)), # The text of y
axis exists above the gridlines
      axis.title.x = element_blank(),
      axis.title.y = element_blank(),
      axis.ticks.y = element_blank(),
      axis.ticks.x = element_blank(),
      axis.line.x.bottom = element_line("#e2edf3", size= 9.7), # Hack : a background
colour line at bottom to hide lower gridlines
      panel.background = element_rect(fill = "#e2edf3"),
      panel.grid.major.x = element_blank(),
      #legend.position = c(0.077, 1),
      legend.position = "none",
      legend.title = element_blank(),
      legend.direction = "horizontal",
      legend.key = element_rect(fill = "white", color = NA),
      plot.title = element_text(size=15,color = "#696363",vjust = -4,face="bold"),
      plot.subtitle = element_text(size=10,hjust = 0.33,vjust = 2,color =
"#696363",face="bold"),
      plot.caption  = element_text(size=10,hjust = 1,vjust = 151,color =
"#696363",face="bold"),
      plot.margin = unit(c(5.5,5.5,26,5.5), "points") # Default margins
:theme_get()$plot.margin -> 5.5points 5.5points 5.5points 5.5points
    ) +
  scale_colour_manual(values = c("#000000","#406d89","#ea1f25")) +
  scale_shape_manual(values = c(19, 23,23)) +
  scale_fill_manual(values = c("yellow","red","black","#e2edf3"))+
  scale_y_continuous(breaks = seq(340, 560, by = 20),minor_breaks = seq(0, 20, 10),limits
= c(341, 567)) +
  scale_x_discrete(labels= countryfullnames,expand = expansion(add = 1.5))
g
```



```
ggsave(plot = g, filename = "Part2.png")

# References:
# https://www.datanovia.com/en/blog/ggplot-legend-title-position-and-labels/
#
https://stackoverflow.com/questions/48214915/how-to-increase-size-of-ggplot-squeezed-hori
zontal-bar-chart
# https://www.listendata.com/2017/03/if-else-in-r.html
# https://viz-ggplot2.rsquaredacademy.com/labels.html
# https://stackoverflow.com/questions/55406829/ggplot-put-axis-text-inside-plot
# https://stackoverflow.com/questions/56097381/adding-some-space-between-the-x-axis-and-the-bars-in-ggplot/56097971
# https://stackoverflow.com/questions/46256851/how-to-add-line-at-top-panel-border-of-ggplot2
# https://stackoverflow.com/questions/10836843/ggplot2-plot-area-margins
```