

EXTENSIONS & INTERACTIVITY

Outline

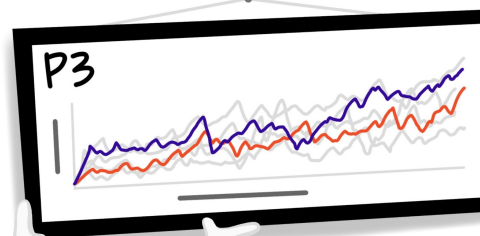
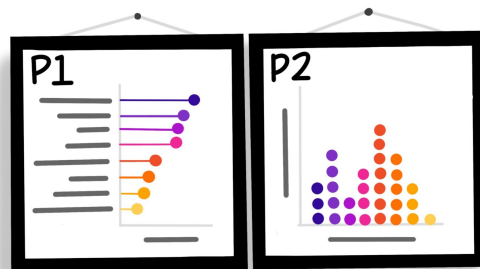
1. Composing multiple plots
2. A plethora of `{ggplot2}` extensions
3. Adding interactivity

Setup

```
## devtools::install_github("haleyjeppson/NCME23data")  
library(NCME23data)  
data(pisa_usa)  
data(pisa_small)  
data(pisa_wide)
```

Compose multiple plots

patchwork
Combine + arrange
your ggplots!



PLAN:
 $(P1 + P2) / P3$

P1	P2
P3	



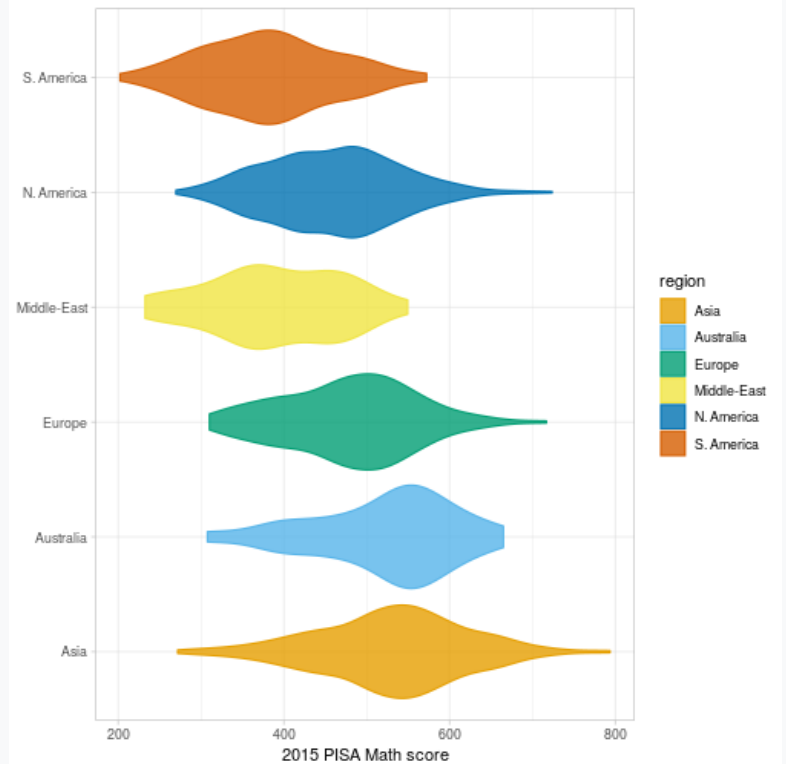
Horst 2.0

Patchwork

First we will build multiple related plots. Here's the first:

```
p1 <- ggplot(pisa_small) +  
  geom_violin(  
    aes(  
      x = math,  
      y = region,  
      color = region,  
      fill = region),  
    alpha = 0.8  
  ) +  
  theme_light() +  
  ggokabeito::scale_fill_okabe_ito() +  
  ggokabeito::scale_color_okabe_ito() +  
  labs(  
    x = "2015 PISA Math score",  
    y = NULL  
  )
```

p1

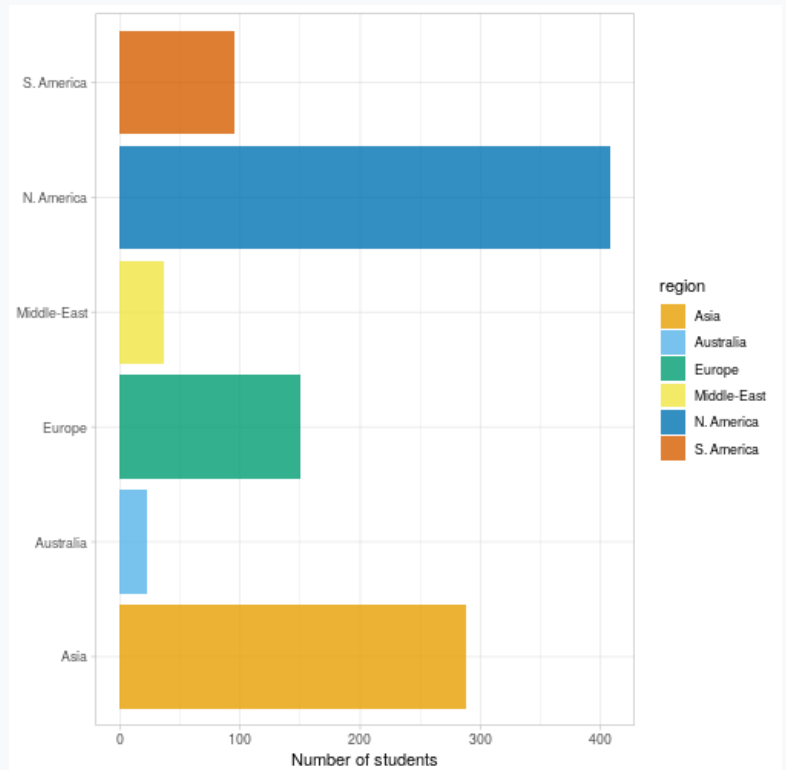


Patchwork

Here's the second

```
p2 <- ggplot(pisa_small) +  
  geom_bar(  
    aes(  
      y = region,  
      fill = region  
    ),  
    alpha = 0.8  
  ) +  
  theme_light() +  
  ggokabeito::scale_fill_okabe_ito() +  
  ggokabeito::scale_color_okabe_ito() +  
  labs(  
    x = "Number of students",  
    y = NULL  
  )
```

p2

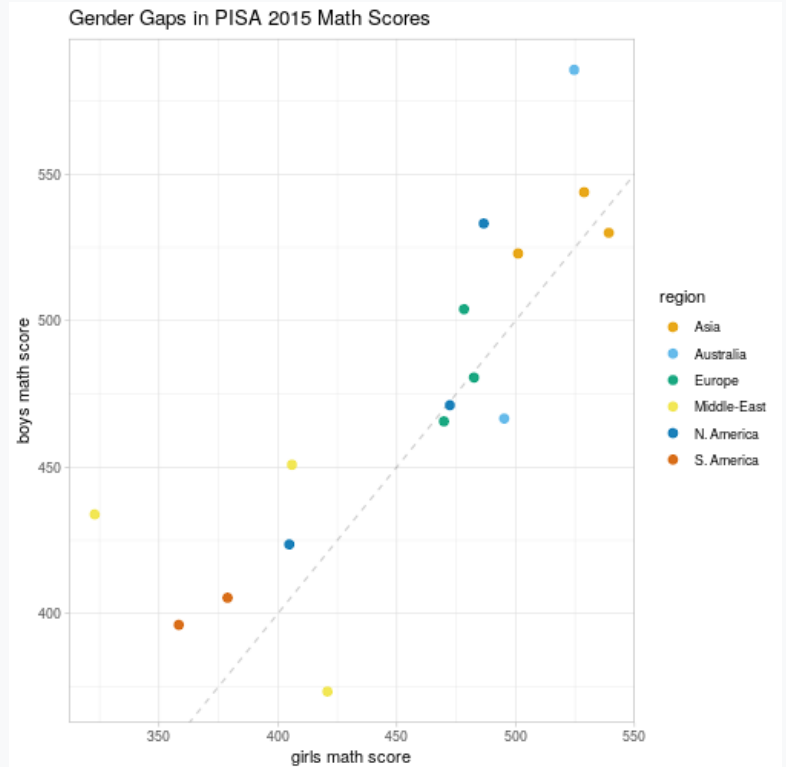


Patchwork

... and here's the third.

```
p3 <- ggplot(pisa_wide,
  aes(x = Female, y = Male)) +
  geom_abline(
    alpha = 0.2,
    linetype = "dashed"
  ) +
  geom_point(
    aes(color = region),
    alpha = 0.9,
    size = 2.5
  ) +
  theme_light() +
  ggokabeito::scale_fill_okabe_ito() +
  ggokabeito::scale_color_okabe_ito() +
  labs(title = "Gender Gaps in PISA 201",
    x = "girls math score",
    y = "boys math score")
```

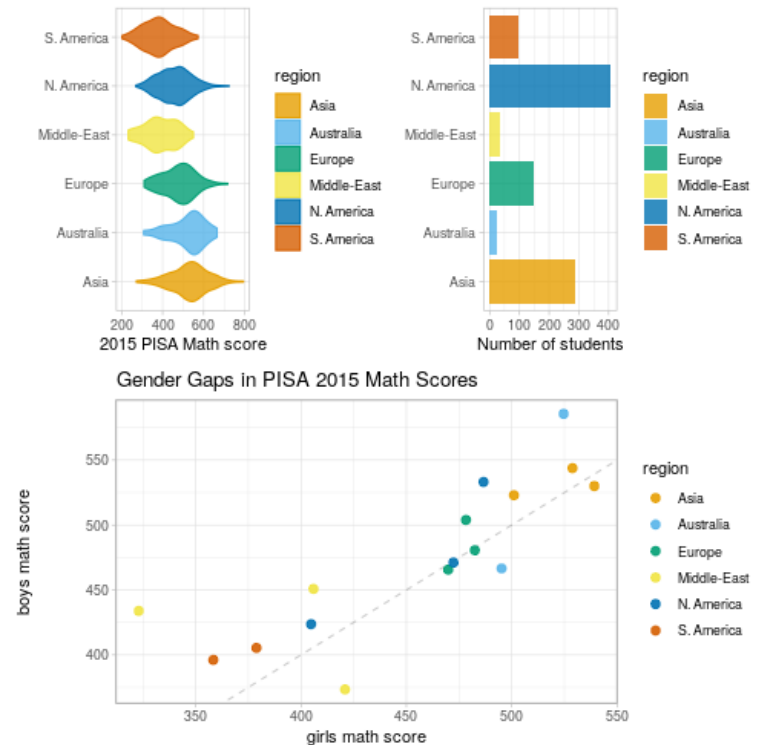
p3



Patchwork

Use **+** and **/** to put multiple figures together into a single graphic

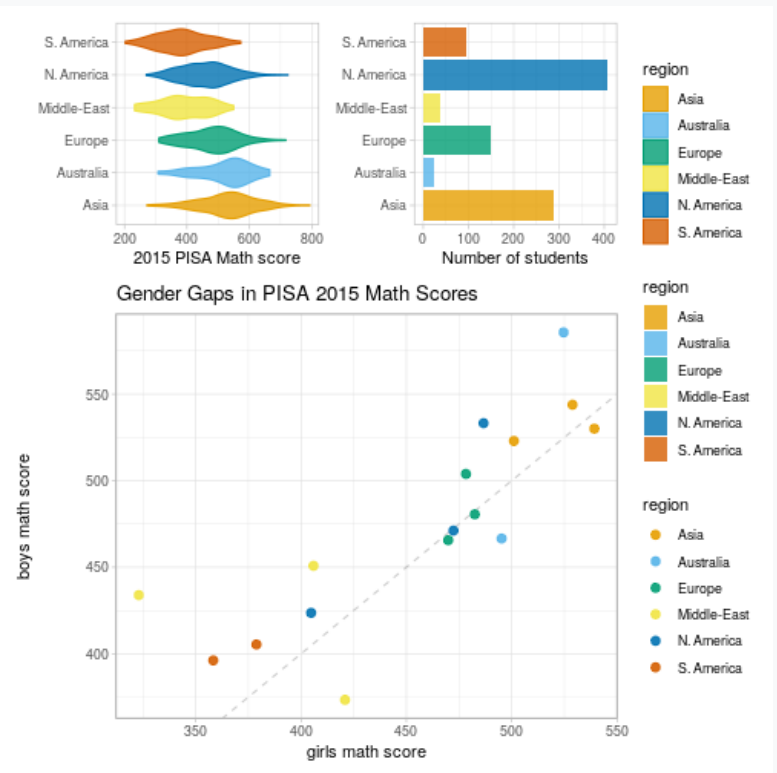
```
library(patchwork)
(p1 | p2) / p3
```



Patchwork

Use `plot_layout()` to control layout

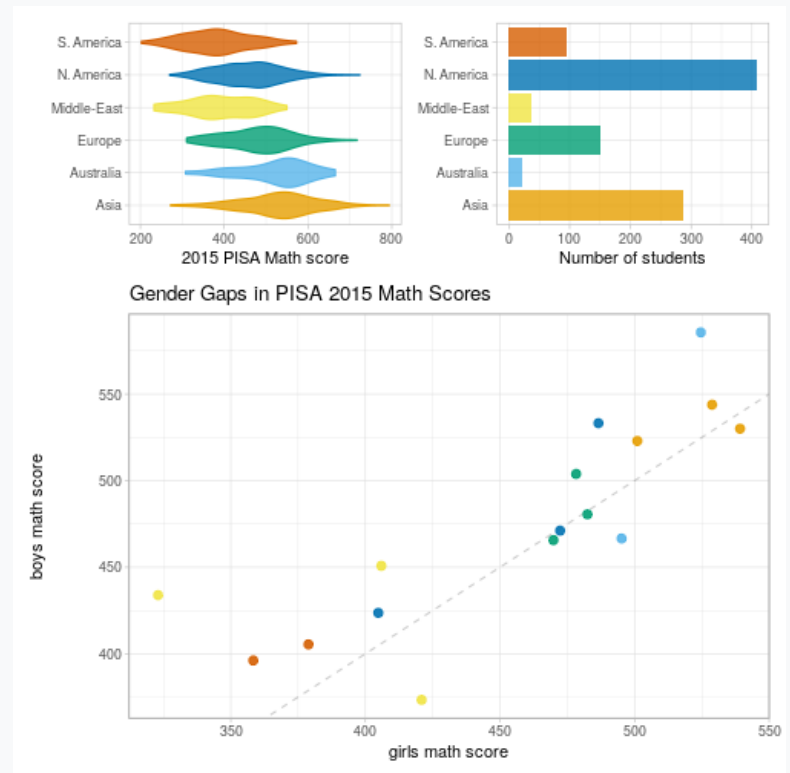
```
(p1 | p2) / p3 +  
  plot_layout(  
    heights = c(1,2),  
    guides = 'collect'  
  )
```



Patchwork

Use `&` to apply theme throughout

```
(p1 | p2) / p3 +  
  plot_layout(  
    heights = c(1,2)  
  ) &  
  theme(  
    legend.position = 'none'  
  )
```



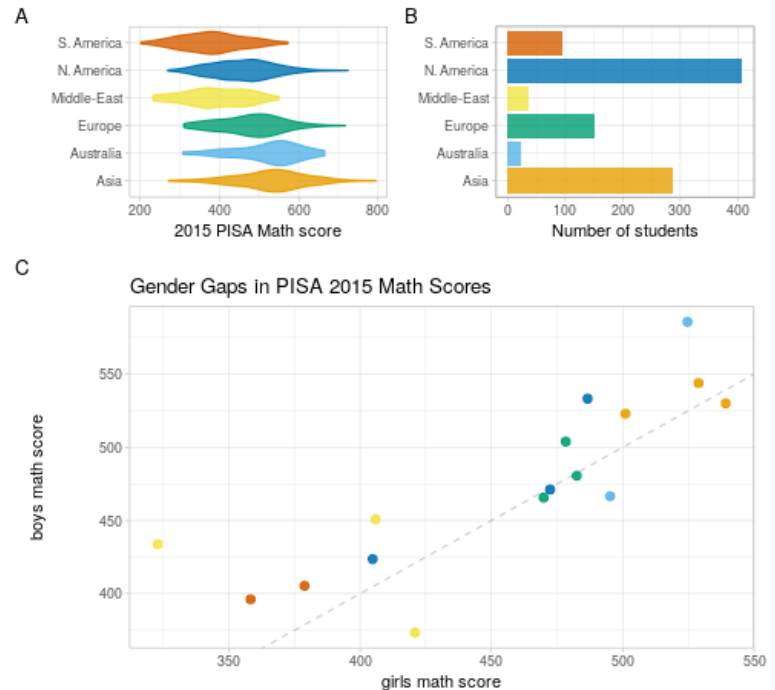
Patchwork

Use `plot_annotation()` to add titles and tags

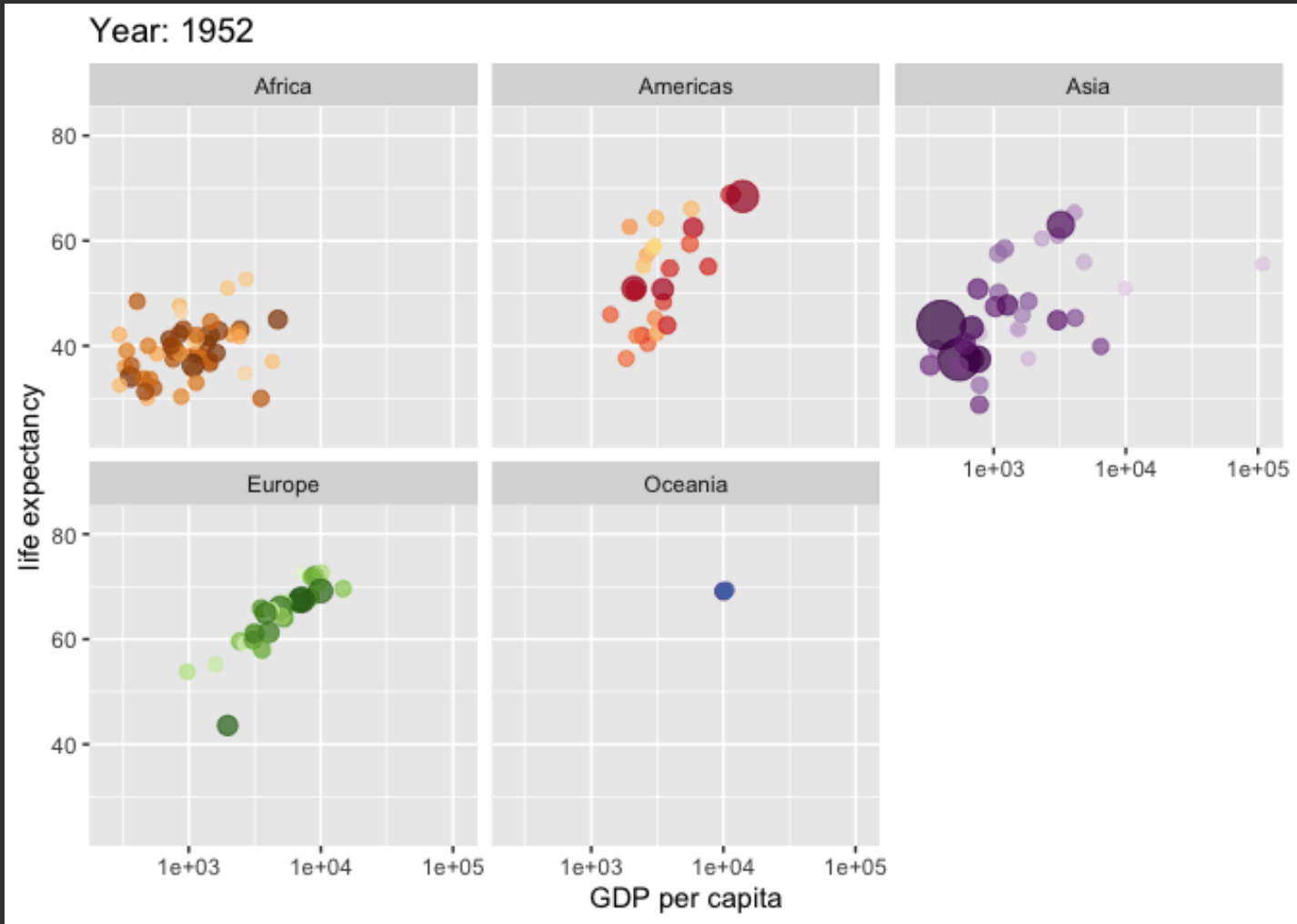
```
(p1 | p2) / p3 +  
  plot_layout(  
    heights = c(1,2)  
  ) +  
  plot_annotation(  
    title = 'Pisa Math Scores in 2015',  
    subtitle = 'A subset of the data shown by region and sex',  
    tag_levels = 'A'  
  ) &  
  theme(  
    legend.position = 'none'  
  )
```

Pisa Math Scores in 2015

A subset of the data shown by region and sex



{ggplot2} extensions



`{ggplot2}` extension packages

Theming & Compositions

- `{cowplot}`
- `{ggthemes}`
- `{ggrepel}`
- `{ggtext}`

Animation & Interactivity

- `{gganimate}`
- `{ggigraph}`
- `{plotly}`

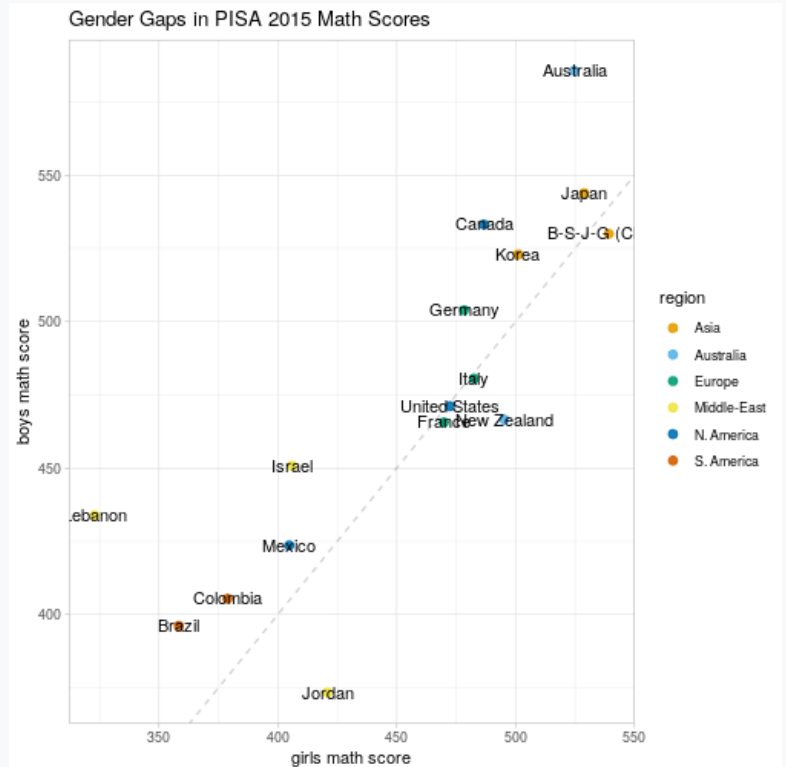
Additional plot types

- `{ggpubr}`
- `{GGally}`
- `{ggcorrplot}`
- `{ggstatsplot}`
- `{ggdag}`
- `{ggradar}`

See [gallery](#) for more!

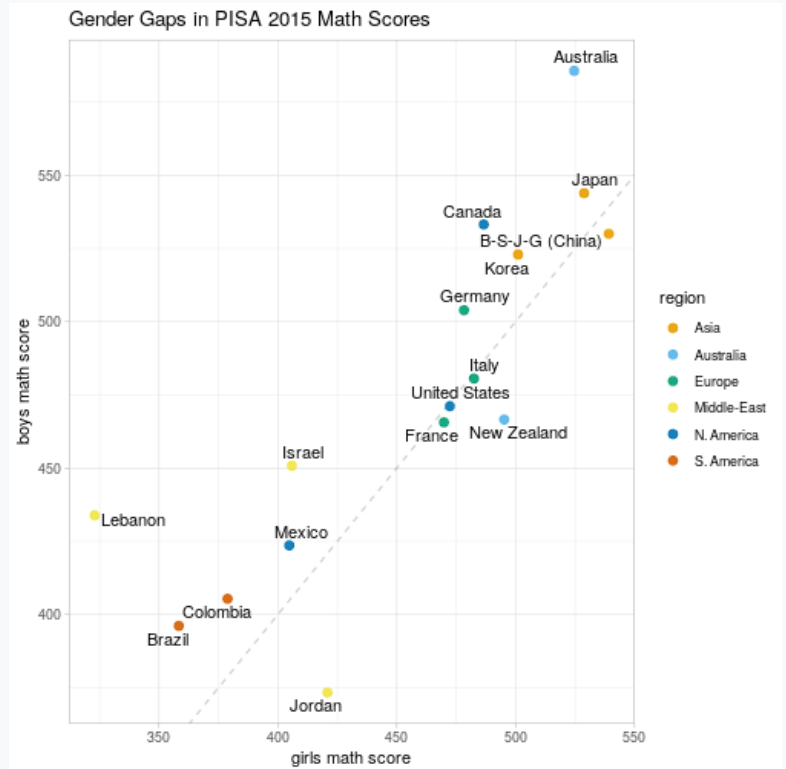
ggrepel

```
ggplot(pisa_wide,
       aes(x = Female, y = Male)) +
  geom_abline(
    alpha = 0.2,
    linetype = "dashed"
  ) +
  geom_point(
    aes(color = region),
    alpha = 0.9,
    size = 2.5
  ) +
  geom_text(
    aes(label = country)
  ) +
  theme_light() +
  ggkabeito::scale_fill_okabe_ito() +
  ggkabeito::scale_color_okabe_ito() +
  labs(
    title = "Gender Gaps in PISA 2015 M
    x = "girls math score",
    y = "boys math score"
  )
```



ggrepel

```
ggplot(pisa_wide,  
       aes(x = Female, y = Male)) +  
  geom_abline(  
    alpha = 0.2,  
    linetype = "dashed"  
  ) +  
  geom_point(  
    aes(color = region),  
    alpha = 0.9,  
    size = 2.5  
  ) +  
  ggrepel::geom_text_repel(  
    aes(label = country)  
  ) +  
  theme_light() +  
  ggkabeito::scale_fill_okabe_ito() +  
  ggkabeito::scale_color_okabe_ito() +  
  labs(  
    title = "Gender Gaps in PISA 2015 M  
    x = "girls math score",  
    y = "boys math score"  
  )
```

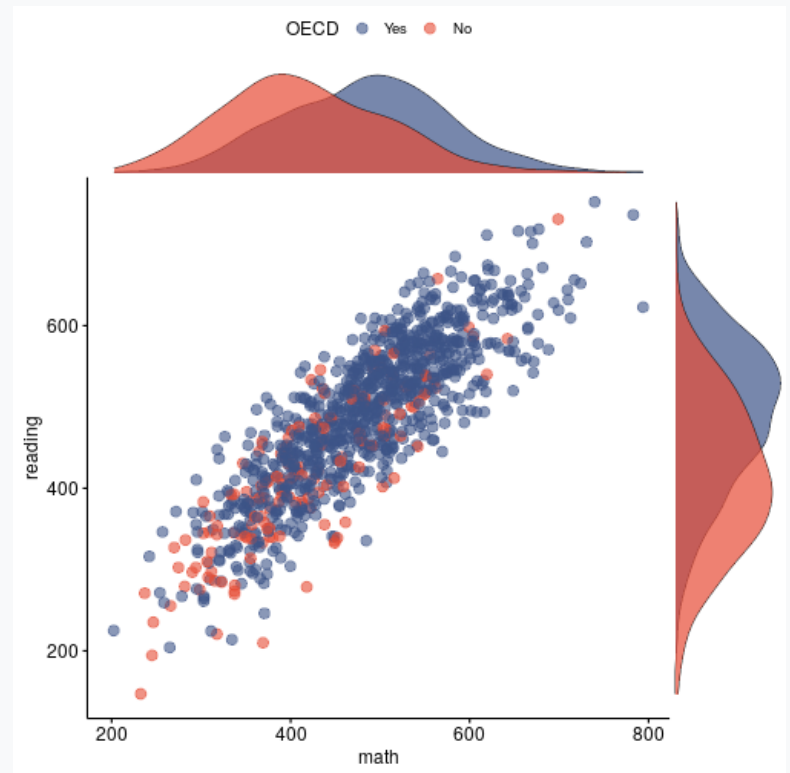


ggpubr

Grouped scatter plot with marginal density plots

```
library(ggpubr)

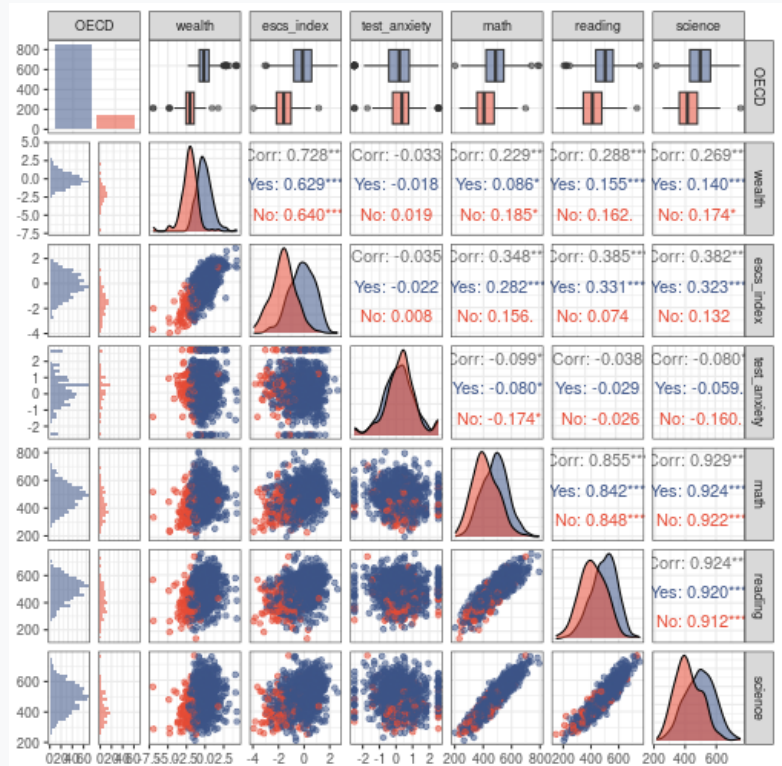
ggscatterhist(pisa_small,
  x = "math",
  y = "reading",
  color = "OECD",
  size = 3,
  alpha = 0.6,
  palette = c("#3C5488", "#E64B35"),
  margin.params = list(
    fill = "OECD",
    color = "black",
    size = 0.2)
)
```



GGally::ggpairs()

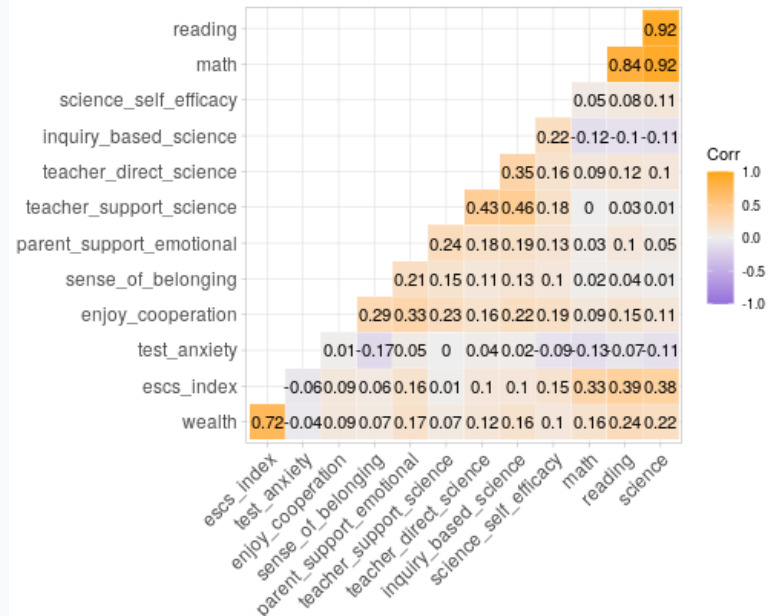
```
library(GGally)
pisa_small_subset <- pisa_small %>%
  select(OECD, wealth,
         escs_index, test_anxiety,
         math:science)
```

```
GGally::ggpairs(
  pisa_small_subset,
  aes(color = OECD,
       alpha = 0.5)
) +
theme_bw() +
scale_color_manual(
  values = c("#3C5488", "#E64B35")
) +
scale_fill_manual(
  values = c("#3C5488", "#E64B35")
)
```



ggcorrplot()

```
# devtools::install_github("kassambara/  
library(ggcorrplot)  
pisa_small_subset <- pisa_small %>%  
  select(  
    wealth, escs_index,  
    test_anxiety:parent_support_emotion  
    teacher_support_science:science)  
  
cor_pisa <- cor(pisa_small_subset,  
  use = "complete.obs")  
ggcorrplot(cor_pisa,  
  type = "lower",  
  outline.col = "white",  
  ggtheme = ggplot2::theme_light  
  colors = c(  
    "mediumpurple", "#EEEEEE",  
    "orange"),  
  lab = TRUE  
)
```



ggstatsplot

ggbetweenstats() creates either a violin plot, a box plot, or a mix of two for between-group or between-condition comparisons with results from statistical tests in the subtitle

```
# install.packages("ggstatsplot")
library(ggstatsplot)
set.seed(123)

ggbetweenstats(
  data = pisa_small,
  x = OECD,
  y = math,
  title = "Distribution of math scores
)
```



Your Turn

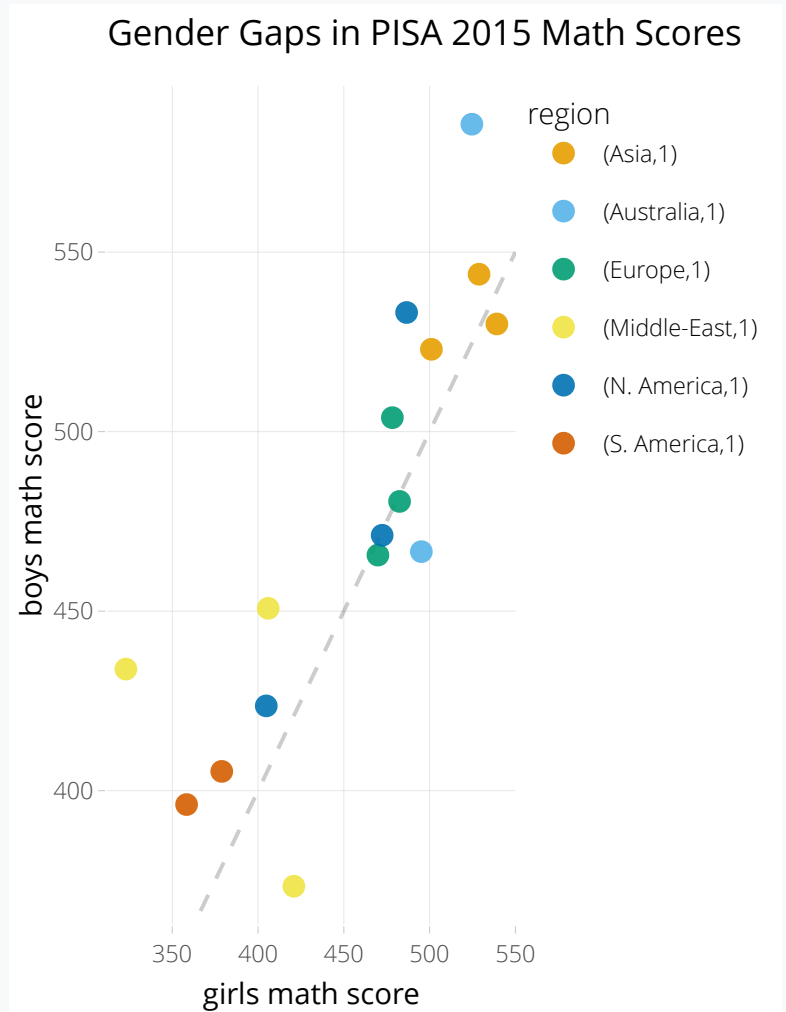
Look through the [extensions gallery](#). Pick out a package and install it.

Bonus: try to run the given example!

Interactivity

ggplotly

```
library(plotly)
ggplotly(p3)
```



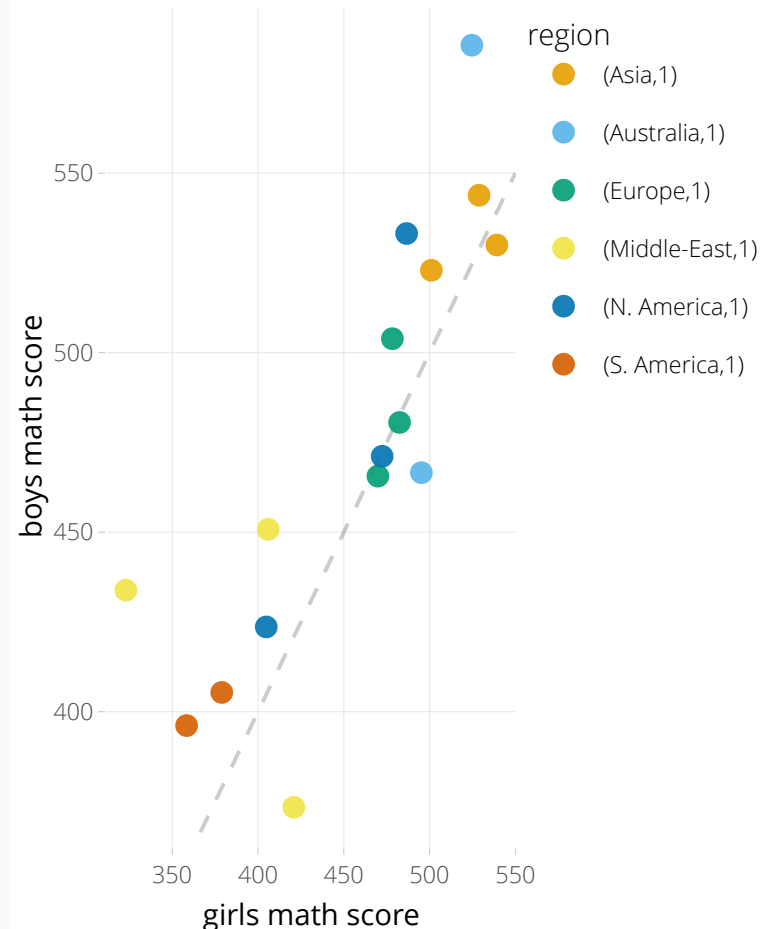
ggplotly

Modify the tooltip output

```
p3_updated <- ggplot(pisa_wide,
  aes(x = Female, y = Male,
      text = paste("country:", cou
    )) +
  geom_abline(alpha = 0.2, linetype = "
  geom_point(
    aes(color = region),
    alpha = 0.9,
    size = 2.5
  ) +
  theme_light() +
  ggokabeito::scale_fill_okabe_ito() +
  ggokabeito::scale_color_okabe_ito() +
  labs(title = "Gender Gaps in PISA 201
    x = "girls math score",
    y = "boys math score")

ggplotly(p3_updated)
```

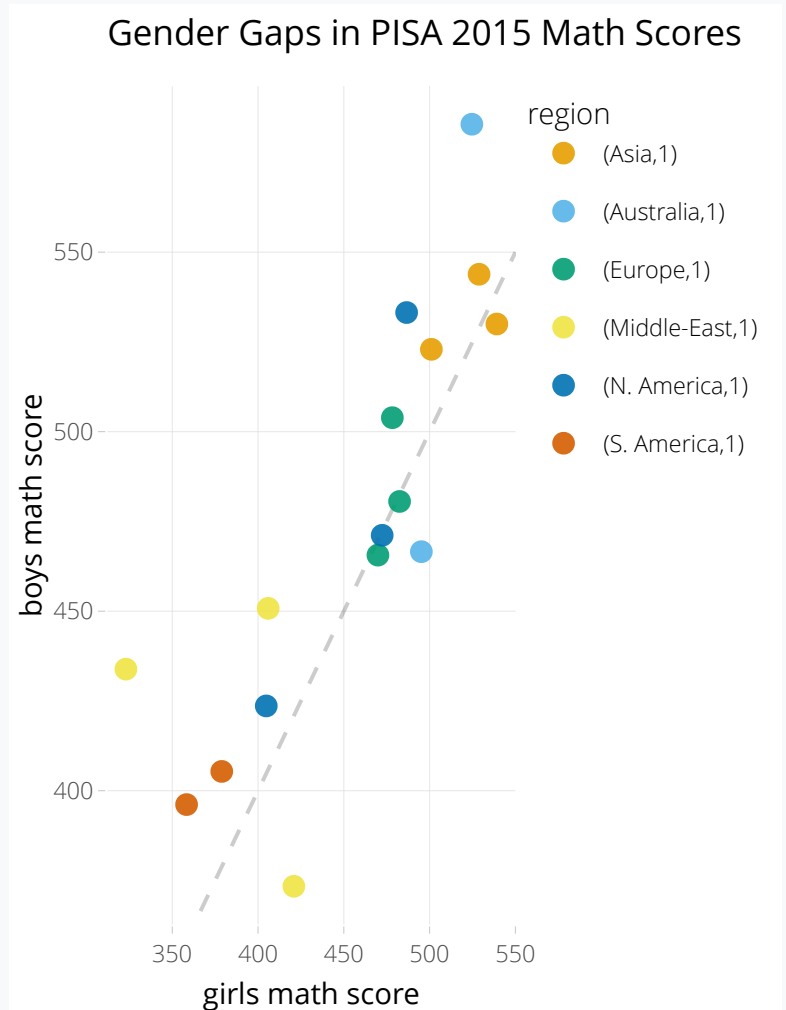
Gender Gaps in PISA 2015 Math Scores



ggplotly

Modify the tooltip output

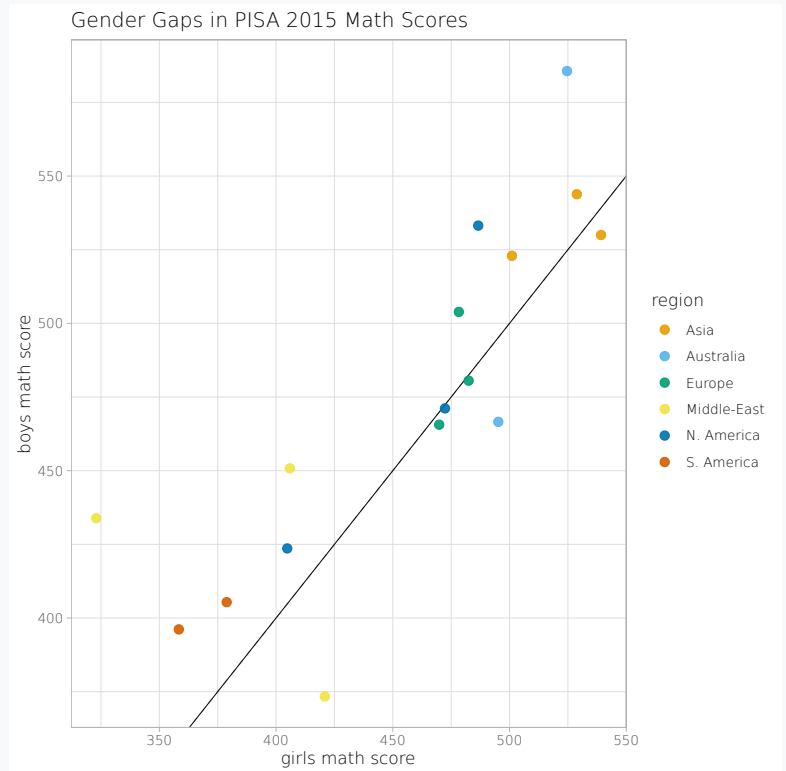
```
ggplotly(p3_updated,  
  tooltip = c("text", "x", "y")  
)
```



ggiraph

```
library(ggiraph)
p3_int <- ggplot(pisa_wide,
  aes(x = Female, y = Male)) +
  geom_abline(alpha = 0.2, linetype = "
  geom_point_interactive(
    aes(color = region,
      tooltip = country,
      data_id = country),
    alpha = 0.9,
    size = 2.5
  ) +
  theme_light() +
  ggokabeito::scale_fill_okabe_ito() +
  ggokabeito::scale_color_okabe_ito() +
  labs(title = "Gender Gaps in PISA 201
    x = "girls math score",
    y = "boys math score")

girafe(ggobj = p3_int)
```



class: yourturn

Your Turn

Take any plot from today's examples and add interactivity with `ggplotly()` (**Bonus:** modify the tooltip)

Resources

- Documentation: <http://ggplot2.tidyverse.org/reference/>
- RStudio cheat sheet for [ggplot2](#)
- Sam Tyner's [ggplot2 workshop](#)
- Thomas Lin Pedersen's ggplot2 webinar: [part 1](#) and [part 2](#)
- Cedric Scherer's "[A ggplot2 tutorial for beautiful plotting in R](#)"

