ggplotline

telling a story with labels, colors, and layout

Malcolm Barrett

10/08/2018

Slides: malco.io/slides/ggplotline

Don't use too many aesthetics and labels. Be selective.

Don't use too many aesthetics and labels. Be selective.

Use color to focus the reader's attention.

Don't use too many aesthetics and labels. Be selective.

Use color to focus the reader's attention.

Combine plots from simpler to more complex. Be consistent but not boring.

```r
library(tidyverse)

scatterplot_extras <- function(legend.position = "none") {
  list(
    theme_minimal(base_size = 14),
    theme(
      legend.position = legend.position,
      panel.grid.minor.x = element_blank(),
      panel.grid.minor.y = element_blank()
     ),
    labs(
      x = "log(GDP per capita)",
      y = "life expectancy"
     ),
    scale_color_manual(values = country_colors)
  )
}
```

```r
library(gapminder)

gapminder_2007 <- gapminder %>%
  filter(year == 2007)

gapminder_2007 %>%
  ggplot(aes(log(gdpPercap), lifeExp, col = country)) +
    geom_point(size = 3.5, alpha = .9) +
    scatterplot_extras("right")
```

an Republic

ı. Rep.

.

:

blic

Direct labeling

1. Label data directly
2. Ditch the legend
3. Use proximity and similarity (e.g. same color)

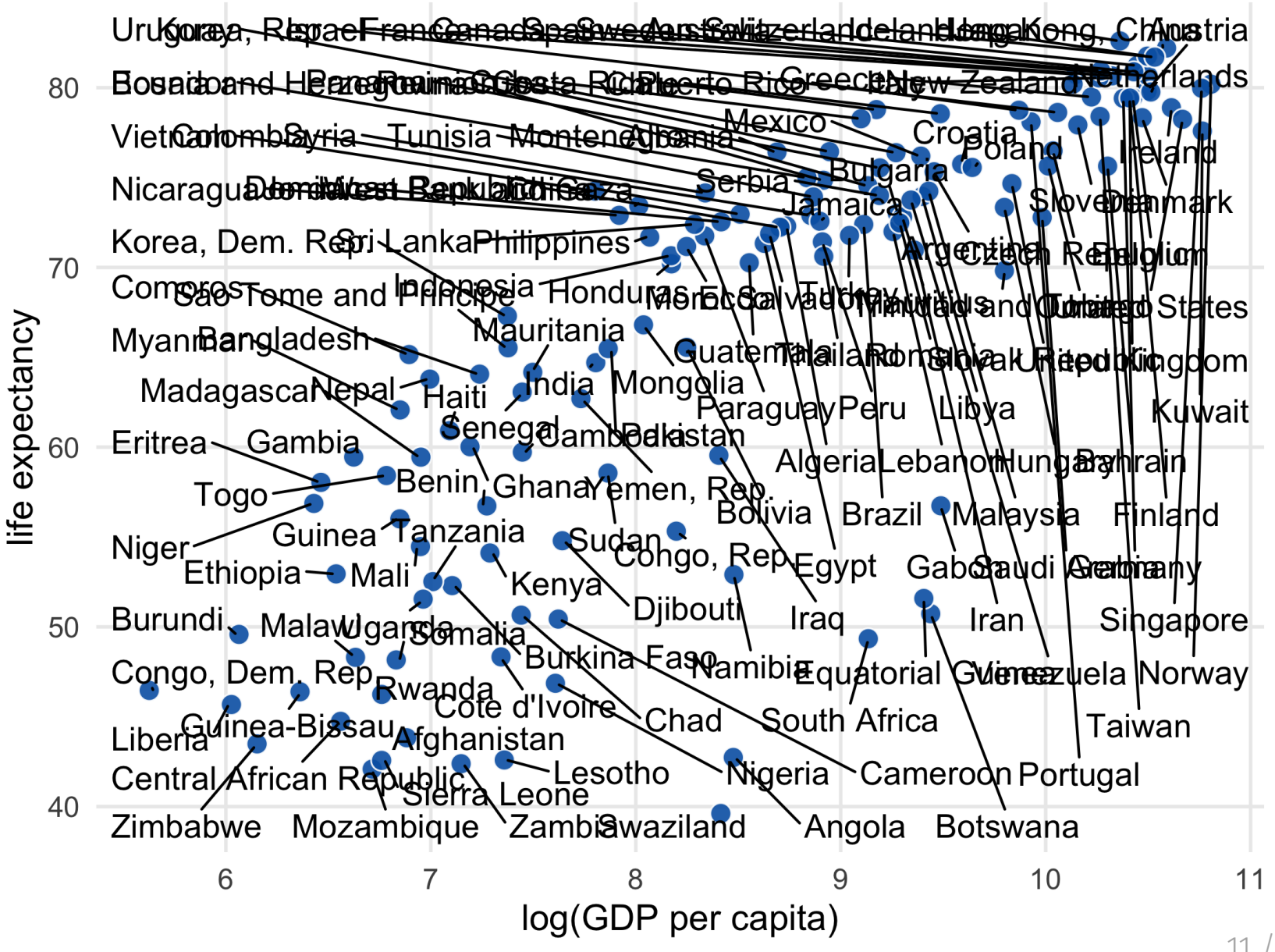
ggrepel: Repel overlapping text

```
library(ggrepel)
```

geom_text_repel()

geom_label_repel()

```
gapminder_2007 %>%
  ggplot(aes(log(gdpPercap), lifeExp)) +
    geom_point(
       size = 3.5,
       alpha = .9,
       shape = 21,
       col = "white",
       fill = "#0162B2"
     ) +
    geom_text_repel(
      aes(label = country),
      size = 4.5,
      point.padding = .2,
      box.padding = .4,
      min.segment.length = 0
      ) +
    scatterplot_extras()
```

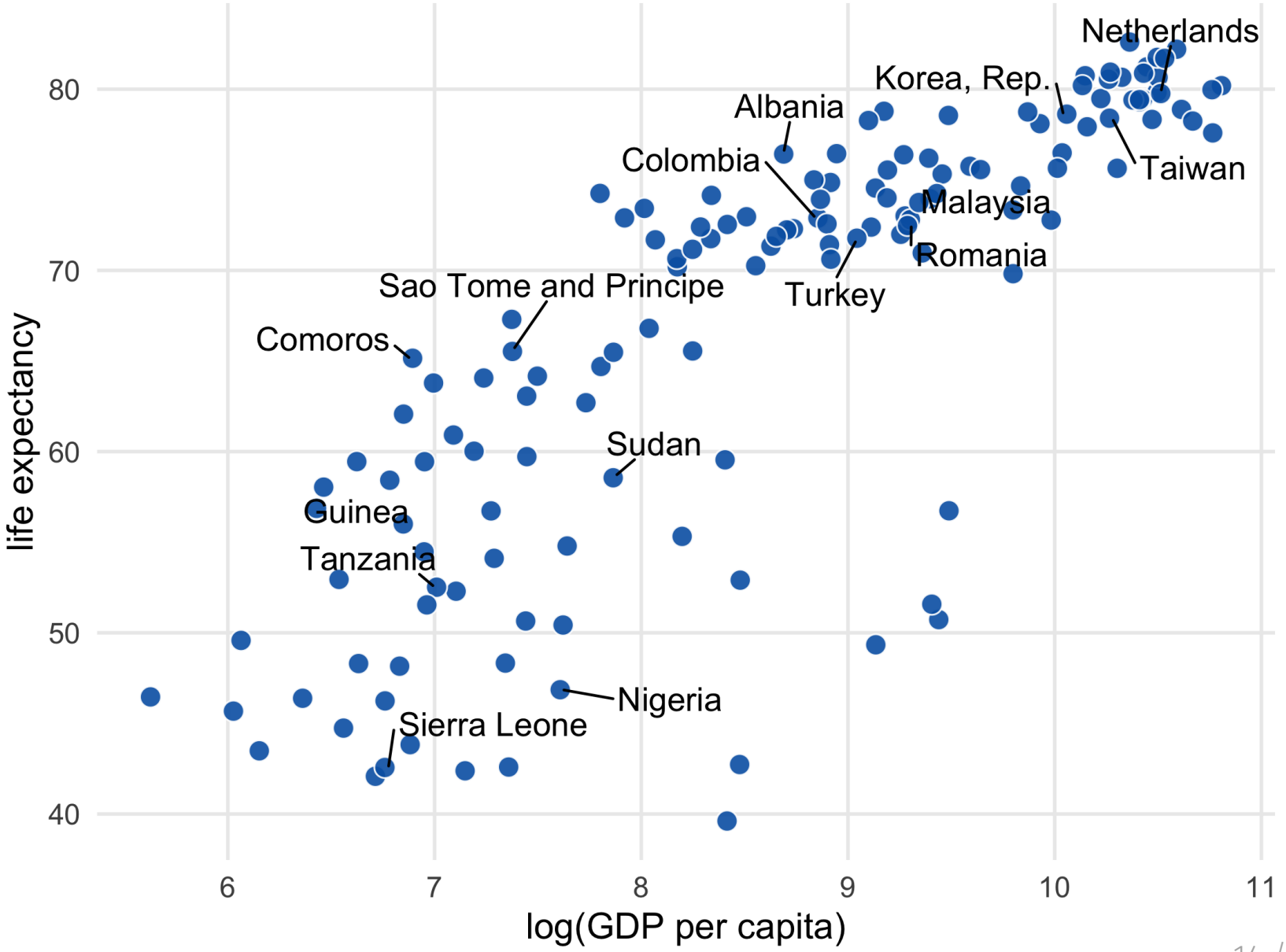

life expectancy
log(GDP per capita)
80
70
60
50
40
6
7
8
9
10
11
Uruguay
Korea, Rep.
Israel
France
Canada
Spain
Sweden
Australia
Switzerland
Iceland
Hong Kong, China
Austria
Bosnia and Herzegovina
Puerto Rico
Greece
New Zealand
Netherlands
Vietnam
Colombia
Syria
Tunisia
Montenegro
Albania
Mexico
Croatia
Poland
Ireland
Nicaragua
Serbia
Bulgaria
Jamaica
Slovenia
Denmark
Korea, Dem. Rep.
Sri Lanka
Philippines
Argentina
Czech Republic
Belgium
Comoros
Sao Tome and Principe
Indonesia
Honduras
Morocco
El Salvador
Turkey
Trinidad and Tobago
United States
Myanmar
Mauritania
Bangladesh
Guatemala
Thailand
Romania
Slovak Republic
United Kingdom
Madagascar
Nepal
Haiti
India
Mongolia
Paraguay
Peru
Libya
Kuwait
Eritrea
Gambia
Senegal
Cambodia
Pakistan
Algeria
Lebanon
Hungary
Bahrain
Togo
Benin
Ghana
Yemen, Rep.
Bolivia
Brazil
Malaysia
Finland
Niger
Guinea
Tanzania
Sudan
Congo, Rep.
Egypt
Gabon
Saudi Arabia
Germany
Ethiopia
Mali
Kenya
Djibouti
Iraq
Iran
Singapore
Burundi
Malawi
Uganda
Somalia
Burkina Faso
Namibia
Equatorial Guinea
Venezuela
Norway
Congo, Dem. Rep.
Rwanda
Cote d'Ivoire
Chad
South Africa
Taiwan
Guinea-Bissau
Liberia
Afghanistan
Central African Republic
Lesotho
Nigeria
Cameroon
Portugal
Sierra Leone
Zimbabwe
Mozambique
Zambia
Swaziland
Angola
Botswana

Sample labels rather than display them all

```
set.seed(1010)

countries <- gapminder_2007 %>%
    sample_n(15) %>%
    pull(country)

countries
```

```
##  [1] Malaysia                 Comoros                  Colombia
##  [4] Nigeria                  Taiwan                   Sierra Leone
##  [7] Netherlands              Turkey                   Guinea
## [10] Romania                  Sudan                    Sao Tome and Princip
## [13] Tanzania                 Korea, Rep.              Albania
## 142 Levels: Afghanistan Albania Algeria Angola Argentina ... Zimbabwe
```

```r
gapminder_2007 %>%
  mutate(label = ifelse(
    country %in% countries,
    as.character(country), ""
   )) %>%
  ggplot(aes(log(gdpPercap), lifeExp)) +
    geom_point(
      size = 3.5,
      alpha = .9,
      shape = 21,
      col = "white",
      fill = "#0162B2"
    ) +
    geom_text_repel(
      aes(label = label),
      size = 4.5,
      point.padding = .2,
      box.padding = .4,
      min.segment.length = 0) +
    scatterplot_extras()
```


Netherlands
Korea, Rep.
Albania
Colombia
Taiwan
Malaysia
Romania
Turkey
Sao Tome and Principe
Comoros
Sudan
Guinea
Tanzania
Nigeria
Sierra Leone
life expectancy
80
70
60
50
40
6
7
8
9
10
11
log(GDP per capita)

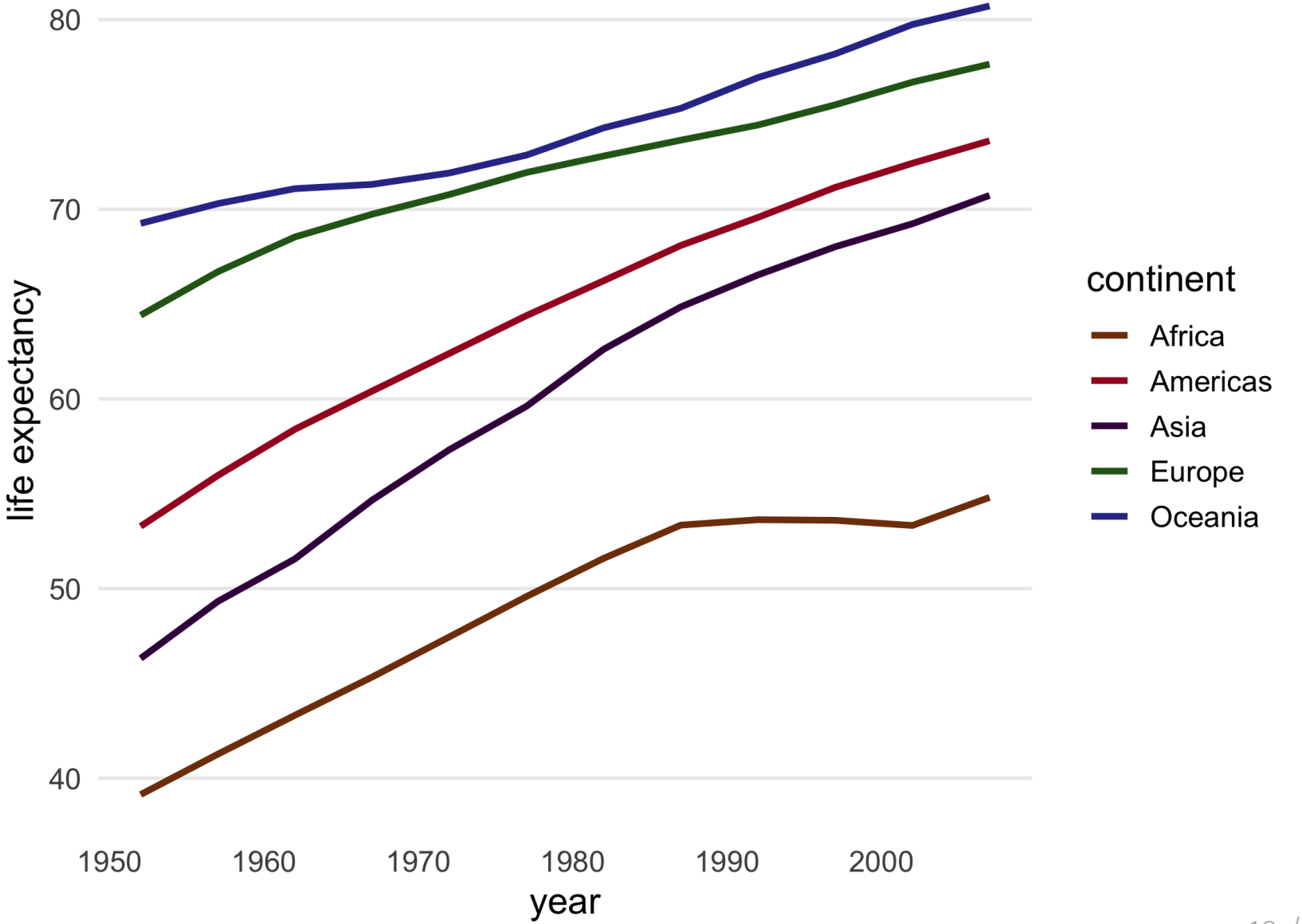
Direct labeling

1. Label data directly
2. Ditch the legend
3. Use proximity and similarity (e.g. same color)

```r
continent_data <- gapminder %>%
   group_by(continent, year) %>%
   summarise(lifeExp = mean(lifeExp))

line_plot_extras <- function(legend.position = "none",
                             values = continent_colors) {
  list(
    theme_minimal(base_size = 14),
    theme(
      legend.position = legend.position,
      panel.grid.major.x = element_blank(),
      panel.grid.minor.x = element_blank(),
      panel.grid.minor.y = element_blank()
     ),
    scale_color_manual(values = values),
    labs(y = "life expectancy")
  )
}
```

Change in average life expectancy by continent

```
continent_data %>%
  ggplot(aes(year, lifeExp, col = continent)) +
    geom_line(size = 1.2) +
    line_plot_extras("right")
```


80
70
60
50
40
life expectancy
1950
1960
1970
1980
1990
2000
year
continent
Africa
Americas
Asia
Europe
Oceania

Change in average life expectancy by continent

```
direct_labels <- continent_data %>%
  group_by(continent) %>%
  summarize(
    x = max(year),
    y = max(lifeExp)
   )

direct_labels
```

```
## # A tibble: 5 x 3
##   continent     x     y
##   <fct>     <dbl> <dbl>
## 1 Africa     2007  54.8
## 2 Americas   2007  73.6
## 3 Asia       2007  70.7
## 4 Europe     2007  77.6
## 5 Oceania    2007  80.7
```

cowplot: Manipulate ggplots

```
library(cowplot)
```

Themes

plot_grid()

Manipulating ggplots

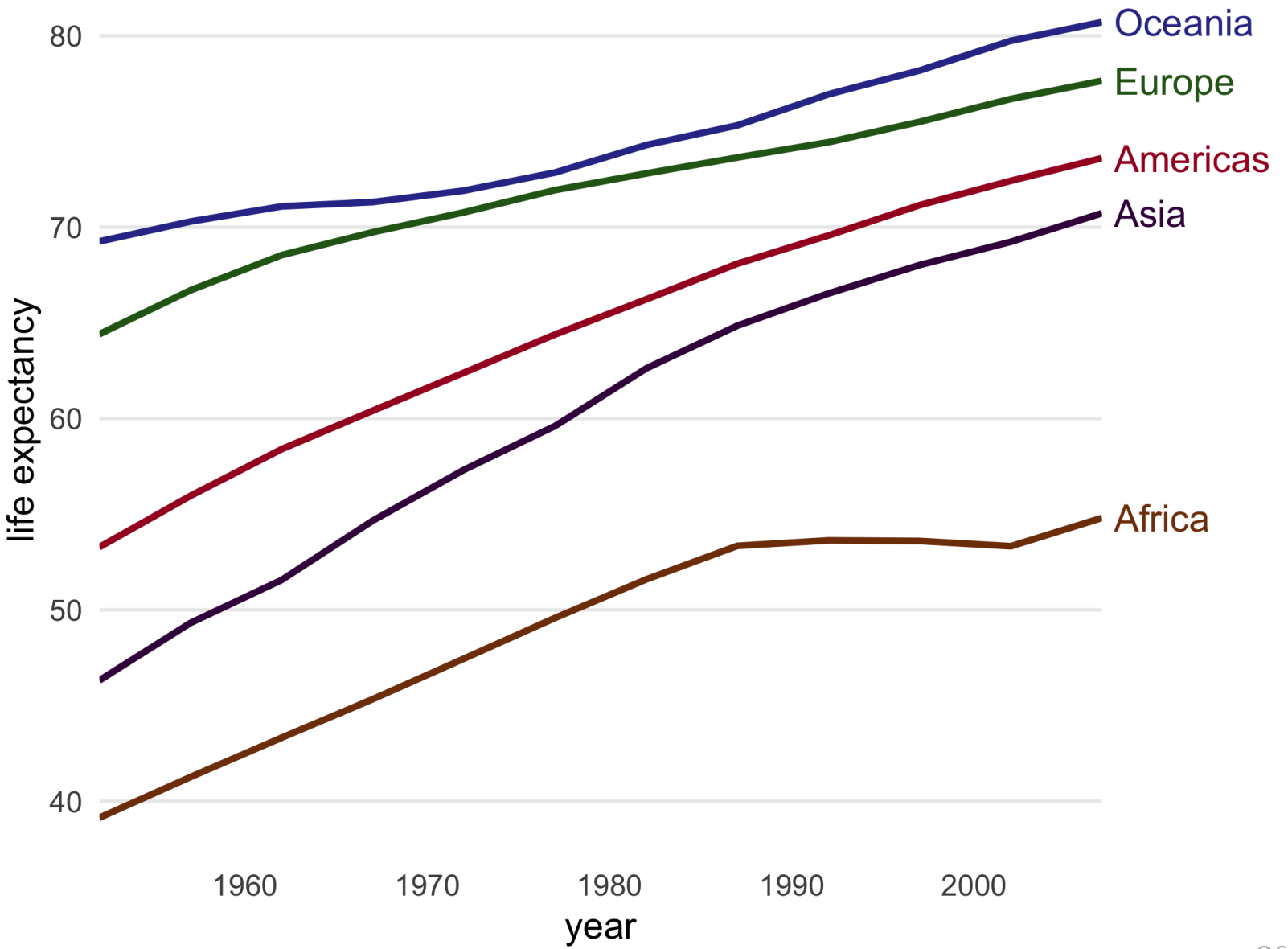
```r
p <- continent_data %>%
  ggplot(aes(year, lifeExp, col = continent)) +
    geom_line(size = 1.2) +
    line_plot_extras() +
    scale_x_continuous(expand = expand_scale(0))

direct_labels_axis <- axis_canvas(p, axis = "y") +
     geom_text(
       data = direct_labels,
       aes(y = y, label = continent),
       x = 0.06,
       hjust = 0,
       size = 5,
       col = continent_colors
      )

p_direct_labels <- insert_yaxis_grob(p, direct_labels_axis)

ggdraw(p_direct_labels)
```

```r
p <- continent_data %>%
  ggplot(aes(year, lifeExp, col = continent)) +
    geom_line(size = 1.2) +
    line_plot_extras() +
    scale_x_continuous(expand = expand_scale(0))

direct_labels_axis <- axis_canvas(p, axis = "y") +
     geom_text(
       data = direct_labels,
       aes(y = y, label = continent),
       x = 0.06,
       hjust = 0,
       size = 5,
       col = continent_colors
      )

p_direct_labels <- insert_yaxis_grob(p, direct_labels_axis)

ggdraw(p_direct_labels)
```

```r
p <- continent_data %>%
  ggplot(aes(year, lifeExp, col = continent)) +
    geom_line(size = 1.2) +
    line_plot_extras() +
    scale_x_continuous(expand = expand_scale(0))

direct_labels_axis <- axis_canvas(p, axis = "y") +
     geom_text(
       data = direct_labels,
       aes(y = y, label = continent),
       x = 0.06,
       hjust = 0,
       size = 5,
       col = continent_colors
      )

p_direct_labels <- insert_yaxis_grob(p, direct_labels_axis)

ggdraw(p_direct_labels)
```

```r
p <- continent_data %>%
  ggplot(aes(year, lifeExp, col = continent)) +
    geom_line(size = 1.2) +
    line_plot_extras() +
    scale_x_continuous(expand = expand_scale(0))

direct_labels_axis <- axis_canvas(p, axis = "y") +
     geom_text(
       data = direct_labels,
       aes(y = y, label = continent),
       x = 0.06,
       hjust = 0,
       size = 5,
       col = continent_colors
      )

p_direct_labels <- insert_yaxis_grob(p, direct_labels_axis)

ggdraw(p_direct_labels)
```

```
p <- continent_data %>%
  ggplot(aes(year, lifeExp, col = continent)) +
    geom_line(size = 1.2) +
    line_plot_extras() +
    scale_x_continuous(expand = expand_scale(0))

direct_labels_axis <- axis_canvas(p, axis = "y") +
     geom_text(
       data = direct_labels,
       aes(y = y, label = continent),
       x = 0.06,
       hjust = 0,
       size = 5,
       col = continent_colors
      )

p_direct_labels <- insert_yaxis_grob(p, direct_labels_axis)

ggdraw(p_direct_labels)
```


80
70
60
50
40
life expectancy
1960
1970
1980
1990
2000
year
Oceania
Europe
Americas
Asia
Africa

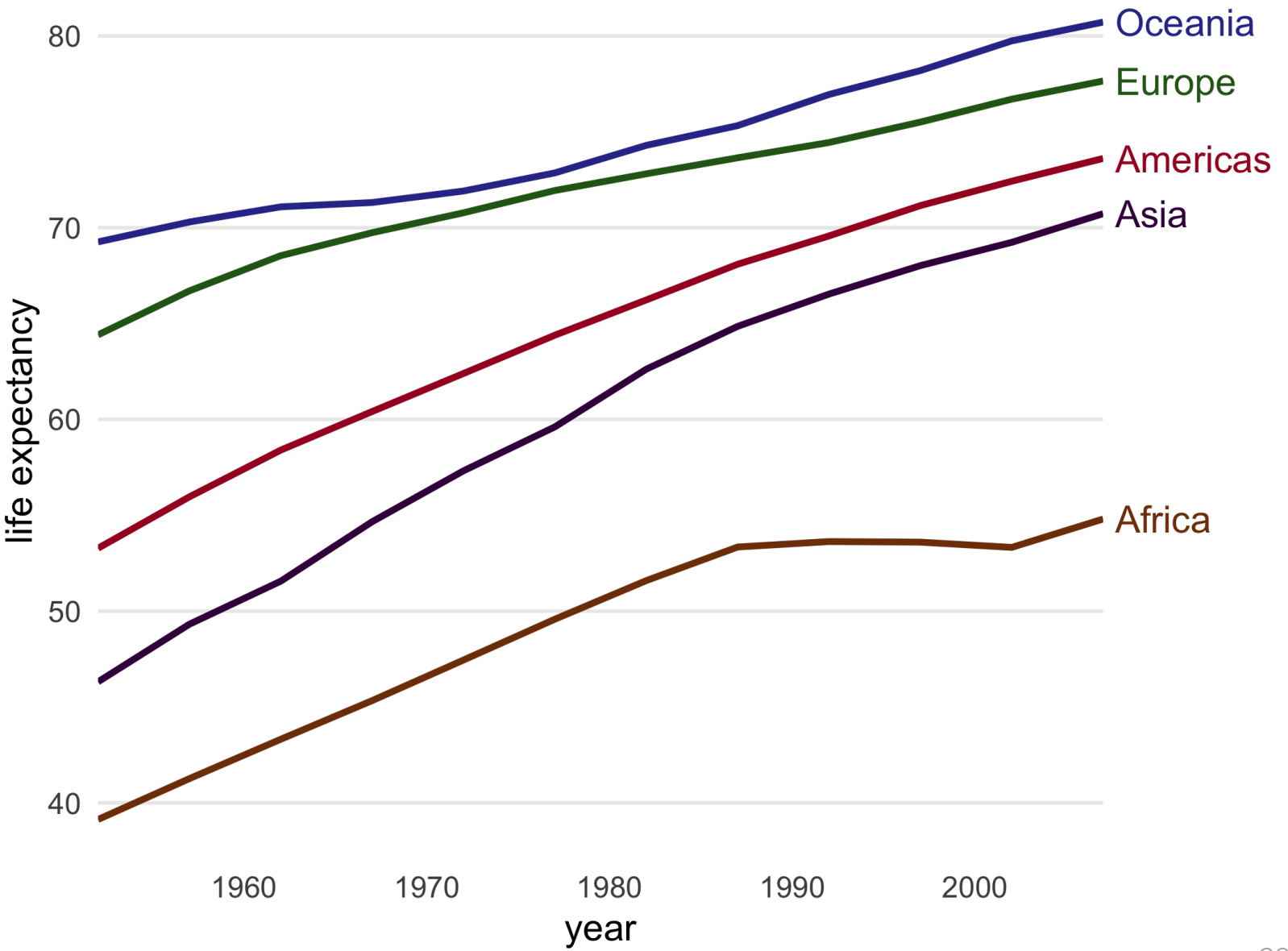
Use color to focus attention

1 2 3 4 5 6 7 8 9

Use color to focus attention

1 2 3 4 5 6 7 8 9

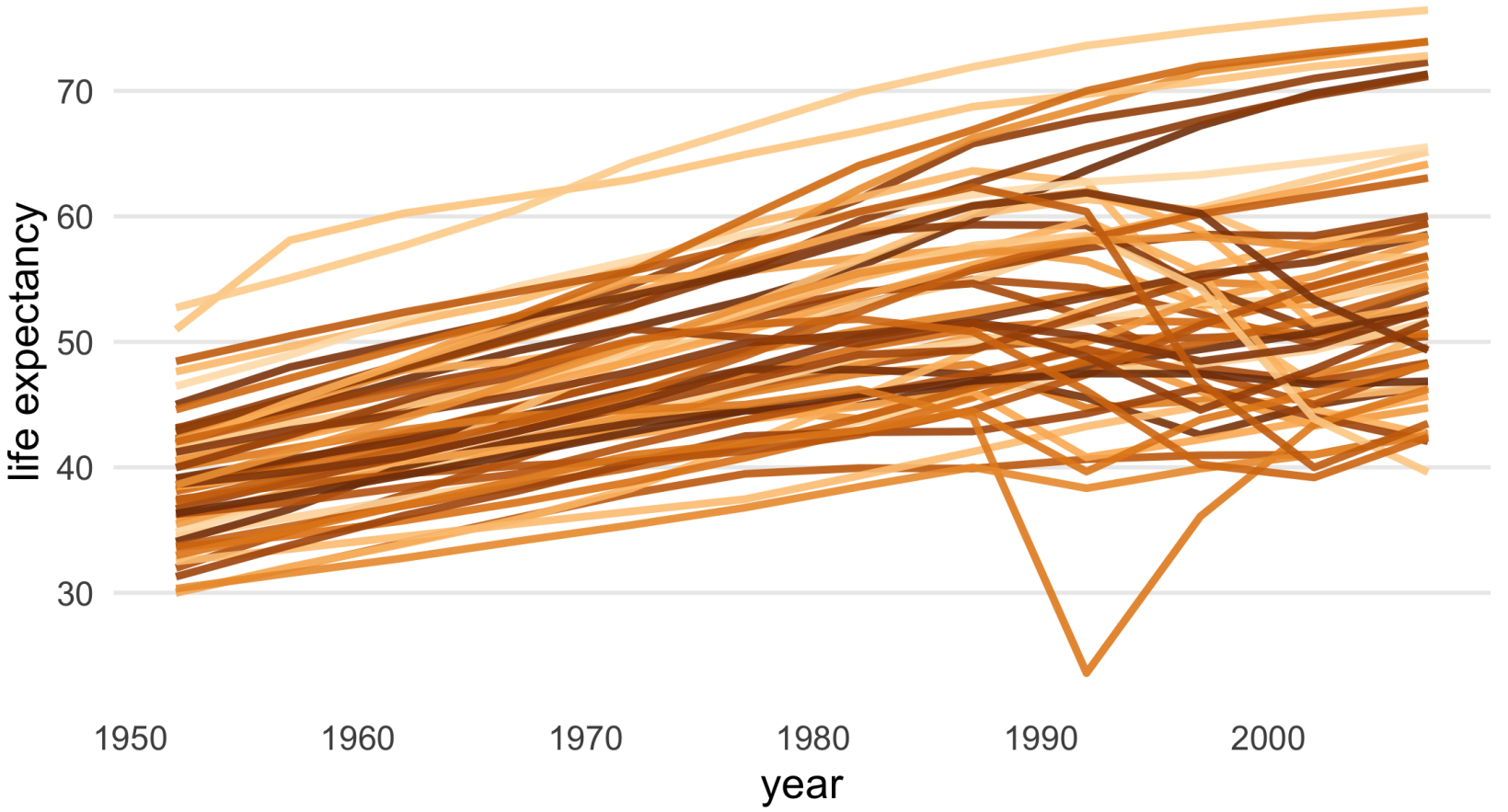
1 2 3 4 5 6 7 8 9


80
70
60
50
40
life expectancy
1960
1970
1980
1990
2000
year
Oceania
Europe
Americas
Asia
Africa

```r
africa <- gapminder %>%
  filter(continent == "Africa")

africa
```

```
## # A tibble: 624 x 6
##    country continent  year lifeExp    pop
##    <fct>   <fct>     <int>   <dbl>  <int>
##  1 Algeria Africa     1952    43.1 9.28e6
##  2 Algeria Africa     1957    45.7 1.03e7
##  3 Algeria Africa     1962    48.3 1.10e7
##  4 Algeria Africa     1967    51.4 1.28e7
##  5 Algeria Africa     1972    54.5 1.48e7
##  6 Algeria Africa     1977    58.0 1.72e7
##  7 Algeria Africa     1982    61.4 2.00e7
##  8 Algeria Africa     1987    65.8 2.33e7
##  9 Algeria Africa     1992    67.7 2.63e7
## 10 Algeria Africa     1997    69.2 2.91e7
## # ... with 614 more rows, and 1 more variable:
## #   gdpPercap <dbl>
```

```
africa %>%
  ggplot(aes(year, lifeExp, col = country)) +
    geom_line(size = 1.2, alpha = .9) +
    line_plot_extras(values = country_colors)
```



gghighlight: Highlight geoms

```
library(gghighlight)
```

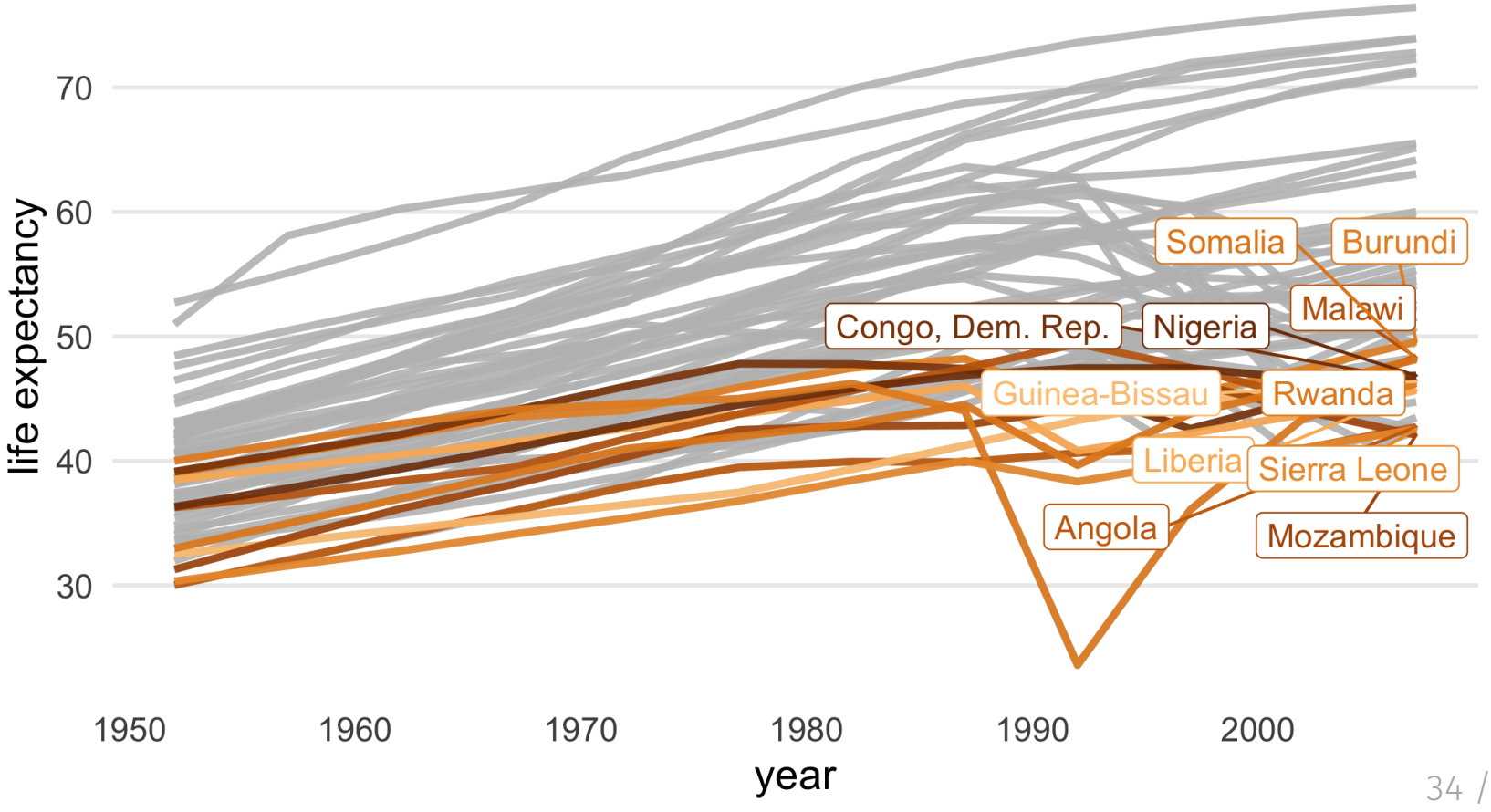
gghighlight(predicate)

Works with points, lines, and histograms

Facets well

```
africa %>%
  ggplot(aes(year, lifeExp, col = country)) +
    geom_line(size = 1.2, alpha = .9) +
    gghighlight(max(lifeExp) < 50, label_key = country) +
    line_plot_extras(values = country_colors)
```

```
africa %>%
  ggplot(aes(year, lifeExp, col = country)) +
    geom_line(size = 1.2, alpha = .9) +
    gghighlight(max(lifeExp) < 50, label_key = country) +
    line_plot_extras(values = country_colors)
```

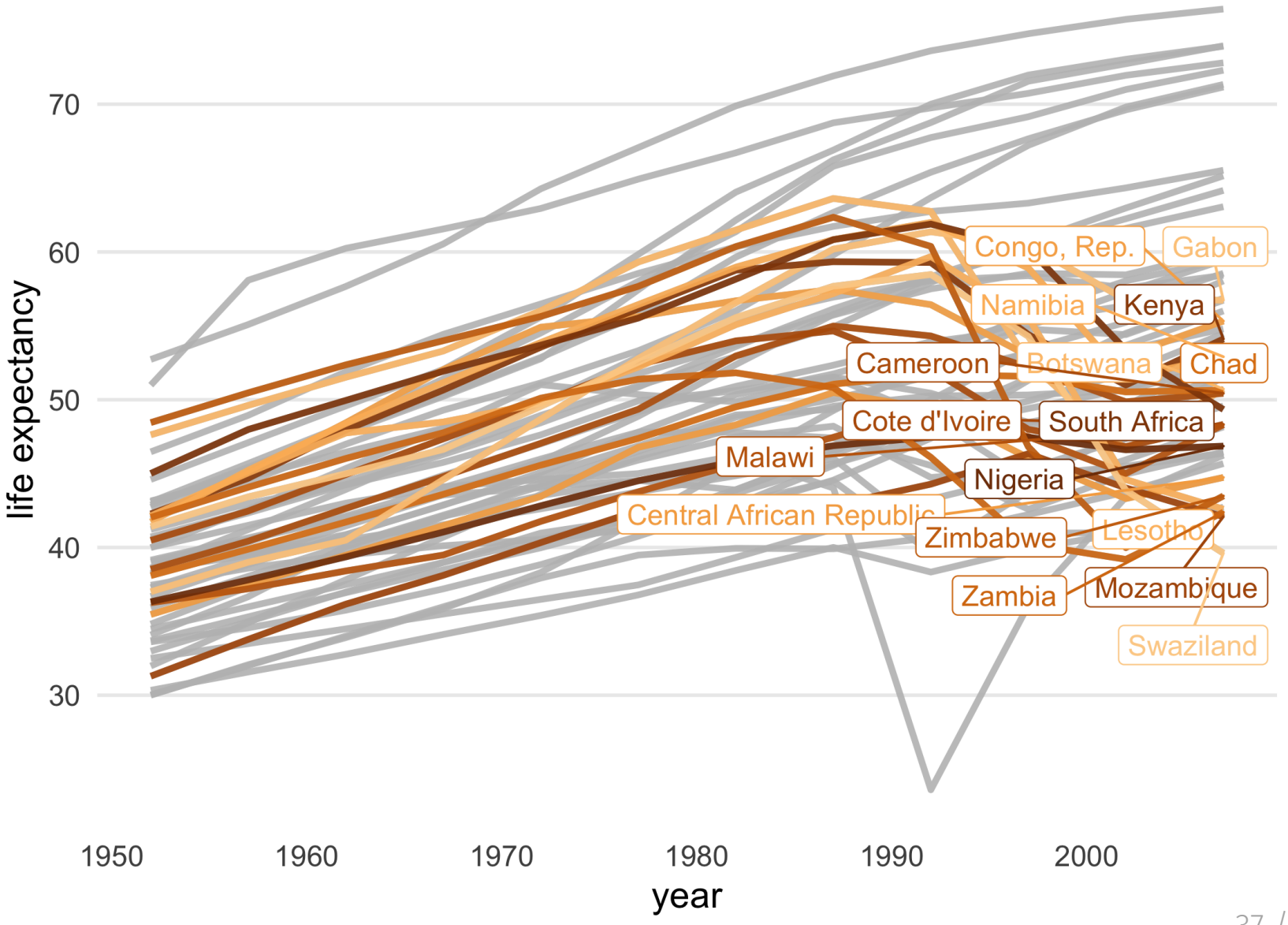


Which countries had higher life expectancy in 1992 than 2007?

```
africa <- africa %>%
  select(country, year, lifeExp) %>%
  spread(year, lifeExp) %>%
  mutate(le_dropped = `1992` > `2007`) %>%
  select(country, le_dropped) %>%
  left_join(africa, by = "country")
```

Which countries had higher life expectancy in 1992 than 2007?

```
africa %>%
  ggplot(aes(year, lifeExp, col = country)) +
    geom_line(
      size = 1.2,
      alpha = .9
     ) +
    gghighlight(
      le_dropped,
      use_group_by = FALSE,
      label_key = country
    ) +
    line_plot_extras(values = country_colors)
```


70
60
50
40
30
life expectancy
1950
1960
1970
1980
1990
2000
year
Congo, Rep.
Gabon
Namibia
Kenya
Cameroon
Botswana
Chad
Cote d'Ivoire
South Africa
Malawi
Nigeria
Central African Republic
Zimbabwe
Lesotho
Zambia
Mozambique
Swaziland

```
africa %>%
  ggplot(aes(year, lifeExp, col = country)) +
    geom_line(
      size = 1.2,
      alpha = .9,
      col = "#E58C23"
    ) +
    gghighlight(
      le_dropped,
      use_group_by = FALSE,
      label_key = labels,
      unhighlighted_colour = "grey90"
     ) +
    line_plot_extras(values = country_colors) +
    xlim(1950, 2015) +
    facet_wrap(~country)
```

Combine plots to tell a story

1. Build plots up from simpler to more complex
2. Don't use the same type of plot in each
3. Use consistent color

patchwork: Compose ggplots

```
library(patchwork)
```

combine plots horizontally: +

combine plots vertically: /

group plots: ()

control layout: plot_layout()

```
diabetes
```

```
## # A tibble: 403 x 19
##       id  chol stab.glu   hdl ratio glyhb location
##    <int> <int>    <int> <int> <dbl> <dbl> <chr>
##  1  1000   203       82    56  3.60  4.31 Bucking…
##  2  1001   165       97    24  6.90  4.44 Bucking…
##  3  1002   228       92    37  6.20  4.64 Bucking…
##  4  1003    78       93    12  6.5   4.63 Bucking…
##  5  1005   249       90    28  8.90  7.72 Bucking…
##  6  1008   248       94    69  3.60  4.81 Bucking…
##  7  1011   195       92    41  4.80  4.84 Bucking…
##  8  1015   227       75    44  5.20  3.94 Bucking…
##  9  1016   177       87    49  3.60  4.84 Bucking…
## 10  1022   263       89    40  6.60  5.78 Bucking…
## # ... with 393 more rows, and 12 more variables:
## #   age <int>, sex <chr>, height <int>,
## #   weight <int>, frame <chr>, bp.1s <int>, …
```

```
diabetes
```

```
## # A tibble: 403 x 19
##       id  chol stab.glu   hdl ratio glyhb location
##    <int> <int>    <int> <int> <dbl> <dbl> <chr>
##  1  1000   203       82    56  3.60  4.31 Bucking…
##  2  1001   165       97    24  6.90  4.44 Bucking…
##  3  1002   228       92    37  6.20  4.64 Bucking…
##  4  1003    78       93    12  6.5   4.63 Bucking…
##  5  1005   249       90    28  8.90  7.72 Bucking…
##  6  1008   248       94    69  3.60  4.81 Bucking…
##  7  1011   195       92    41  4.80  4.84 Bucking…
##  8  1015   227       75    44  5.20  3.94 Bucking…
##  9  1016   177       87    49  3.60  4.84 Bucking…
## 10  1022   263       89    40  6.60  5.78 Bucking…
## # ... with 393 more rows, and 12 more variables:
## #   age <int>, sex <chr>, height <int>,
## #   weight <int>, frame <chr>, bp.1s <int>, …
```

Assess the relationship between sex, a1c, waist to hip ratio, and body frame

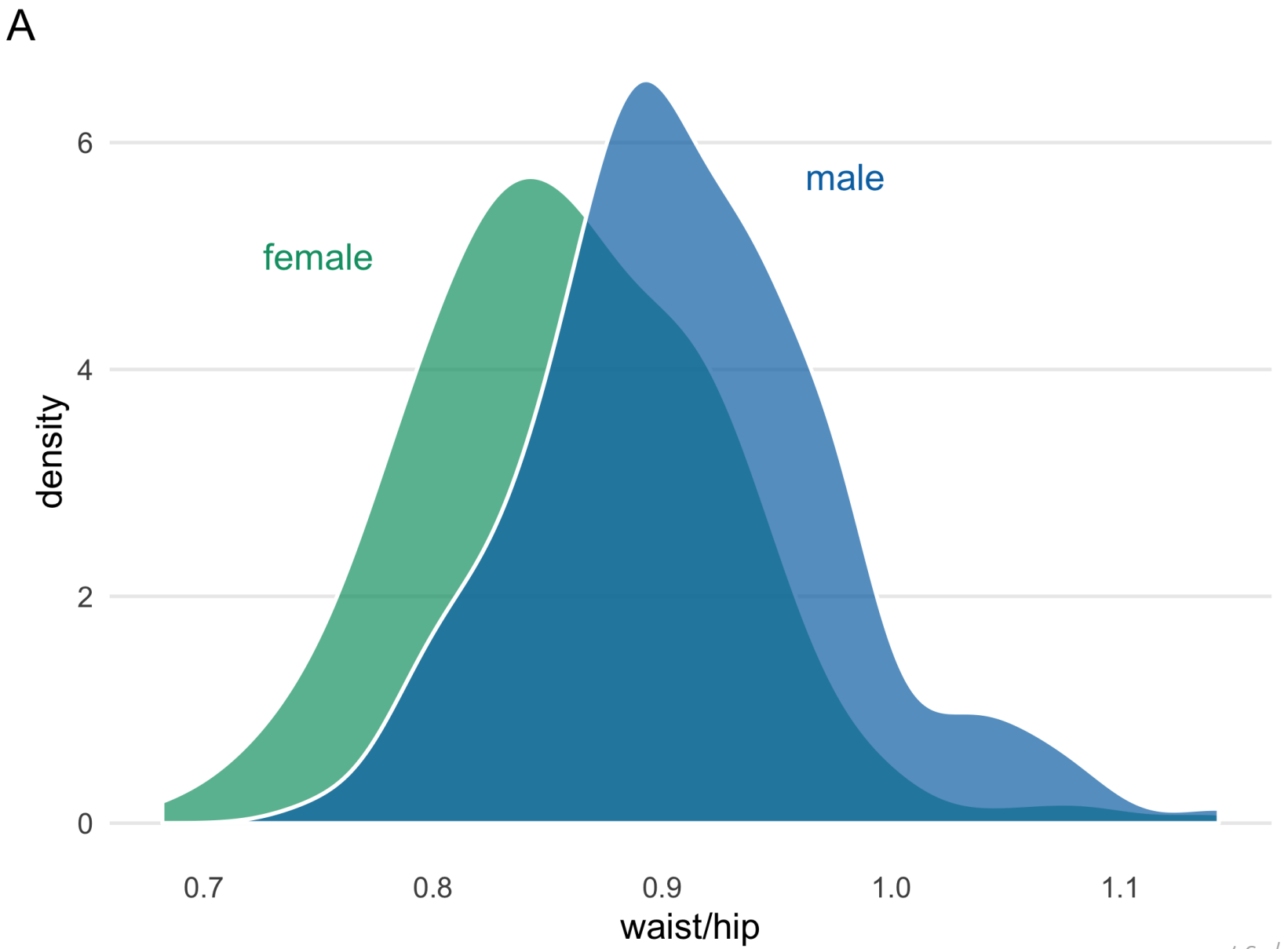
```r
label_frames <- function(lbl) paste(lbl, "\nframe")

theme_multiplot <- function(base_size = 14, ...) {
  theme_minimal(base_size = base_size, ...) %+replace%
    theme(
      panel.grid.major.x = element_blank(),
      panel.grid.minor.x = element_blank(),
      panel.grid.minor.y = element_blank(),
      legend.position = "none"
    )
}
```

Combine plots to tell a story

1. Build plots up from simpler to more complex
2. Don't use the same type of plot in each
3. Use consistent color

density plot

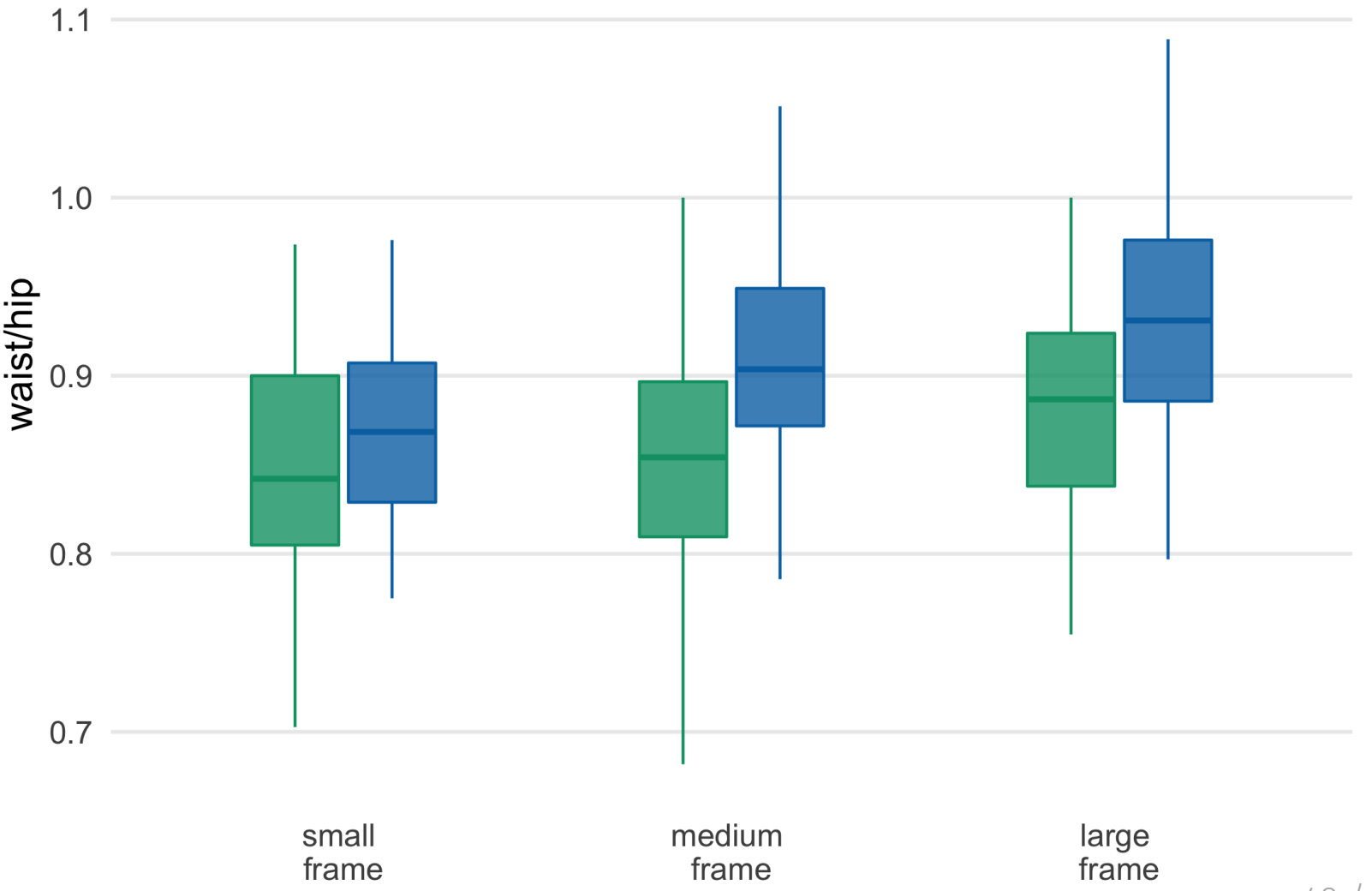
```
plot_a <- diabetes %>%
  ggplot(aes(waist/hip, fill = sex)) +
    geom_density(
      col = "white",
      alpha = .7,
      size = .75
     ) +
    theme_multiplot() +
    scale_fill_manual(values = c("#009E73", "#0072B2")) +
    annotate(
       "text",
       x = c(.75, .98),
       y = c(5, 5.70),
       label = c("female", "male"),
       color = c("#009E73", "#0072B2"),
       size = 5
      ) +
    labs(tag = "A")
```


A
6
4
2
0
density
female
male
0.7
0.8
0.9
1.0
1.1
waist/hip

boxplot

```
plot_b <- diabetes %>%
  drop_na(frame) %>%
  ggplot(aes(fct_rev(frame), waist/hip, fill = sex, col = sex)) +
    geom_boxplot(
      outlier.color = NA,
      alpha = .8,
      width = .5
     ) +
    theme_multiplot() %+replace%
    theme(axis.title.x = element_blank()) +
    scale_x_discrete(labels = label_frames) +
    scale_color_manual(values = c("#009E73", "#0072B2")) +
    scale_fill_manual(values = c("#009E7370", "#0072B270")) +
    labs(tag = "B")
```

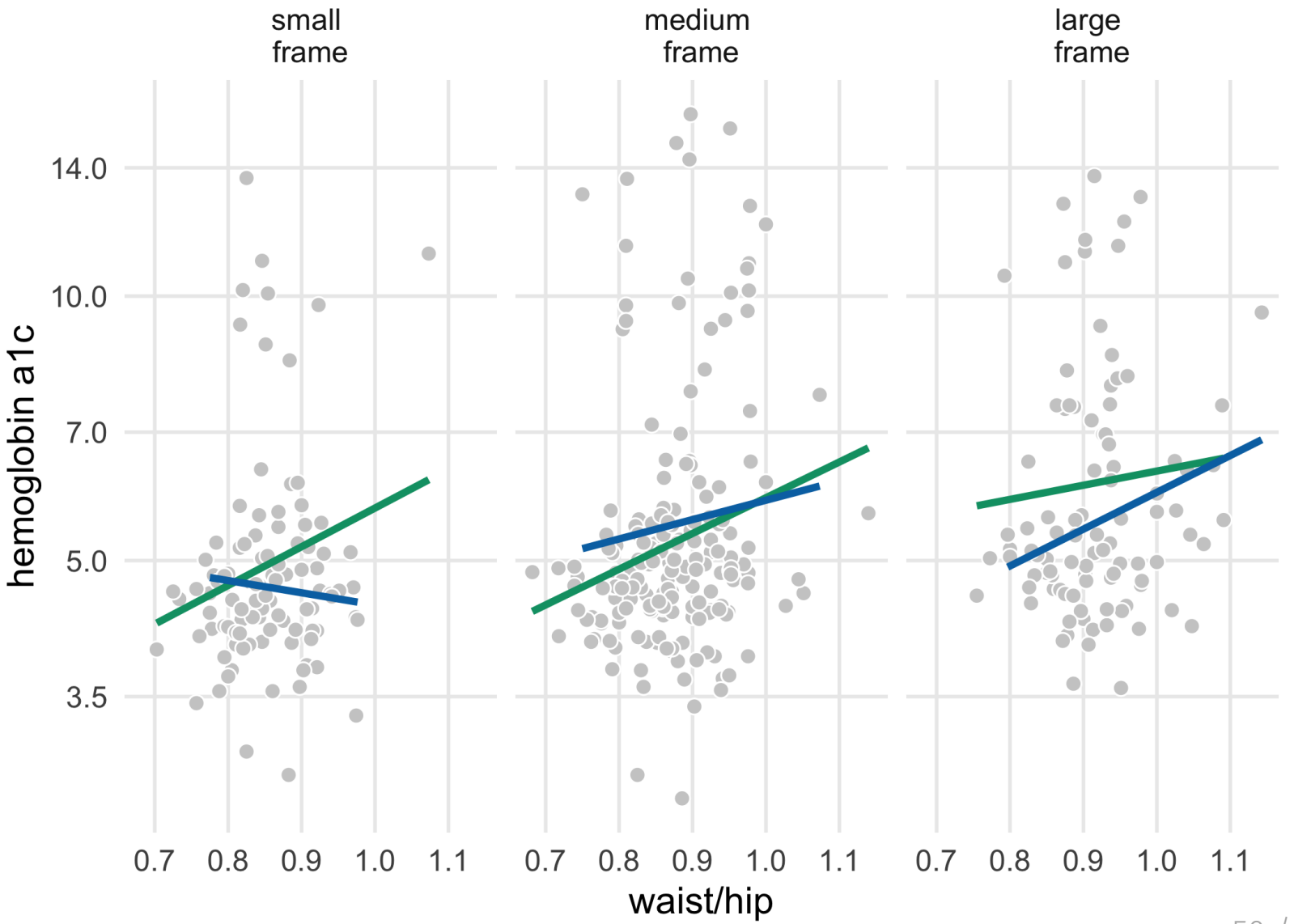
B



scatter plot with regression lines

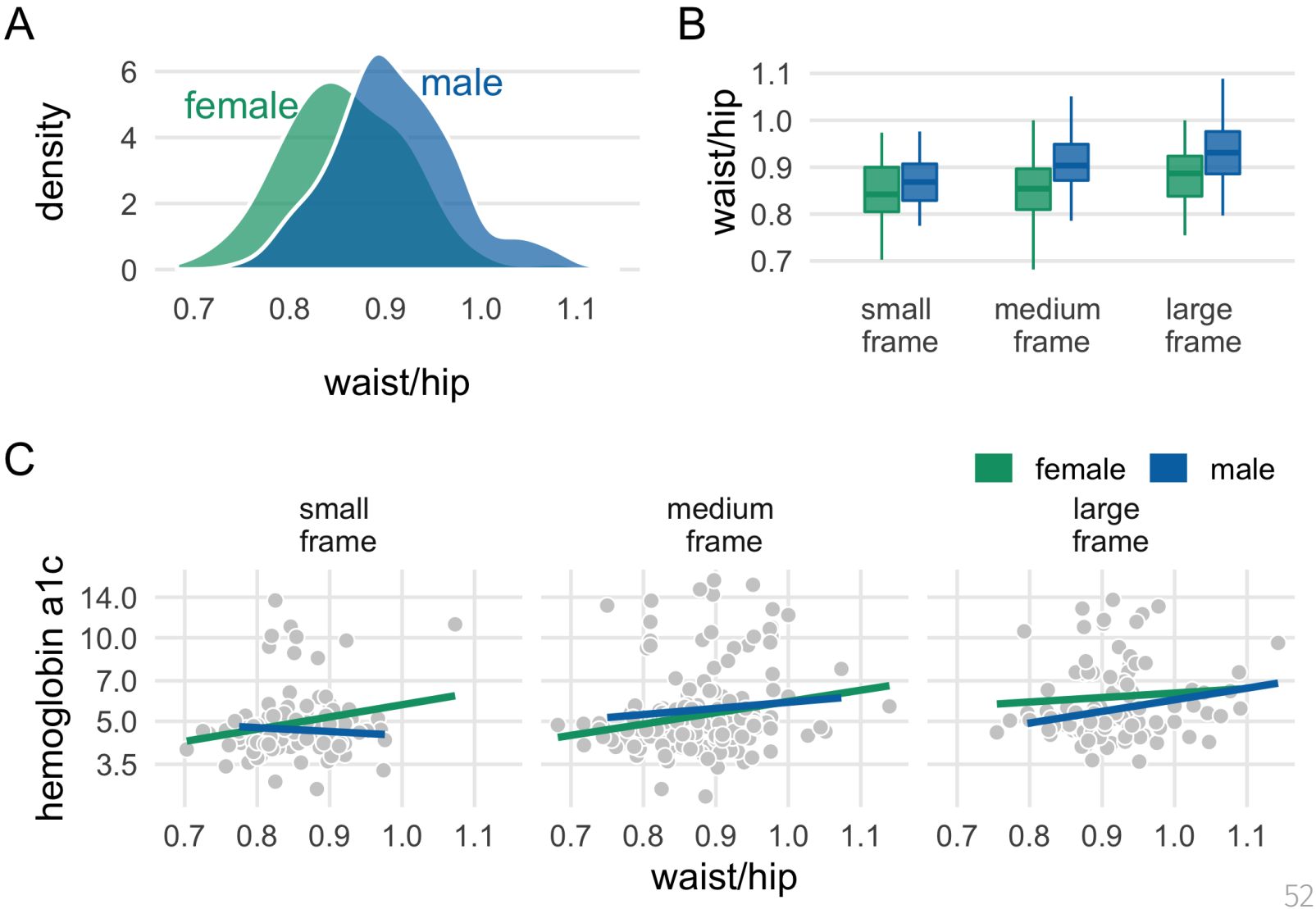
```
plot_c <- diabetes %>%
  drop_na(frame) %>%
  ggplot(aes(waist/hip, glyhb,  col = sex)) +
    geom_point(
      shape = 21,
      col = "white",
      fill = "grey80",
      size = 2.5
     ) +
    geom_smooth(
      method = "lm",
      se = FALSE,
      size = 1.3
    ) +
    theme_minimal(base_size = 14) +
    theme(
      legend.position = c(1, 1.25),
      legend.justification = c(1, 0),
      legend.direction = "horizontal",
      panel.grid.minor.x = element_blank(),
      panel.grid.minor.y = element_blank()
     ) +
    facet_wrap(~fct_rev(frame), labeller = as_labeller(label_frames)) +
    labs(tag = "C", y = "hemoglobin a1c") +
    scale_y_log10(breaks = c(3.5, 5.0, 7.0, 10.0, 14.0)) +
    scale_color_manual(name = "", values = c("#009E73FF", "#0072B2FF")) +
    guides(color = guide_legend(override.aes = list(size = 5)))
```

C


small frame
medium frame
large frame
hemoglobin a1c
14.0
10.0
7.0
5.0
3.5
0.7
0.8
0.9
1.0
1.1
waist/hip

```
(plot_a + plot_b) / plot_c
```

```
(plot_a + plot_b) / plot_c
```



Combining patchwork and cowplot

```
legend <- ggdraw() +
  get_legend(plot_c + theme(legend.position = "bottom"))

(plot_a + plot_b) /
(plot_c + theme(legend.position = "none")) /
legend +
plot_layout(heights = c(10, 10, 1)) +
plot_annotation(
  "The relationship between waist/hip ratio in males and females by
    frame size",
  theme = theme(plot.title = element_text(size = 16, face = "bold"))
)
```

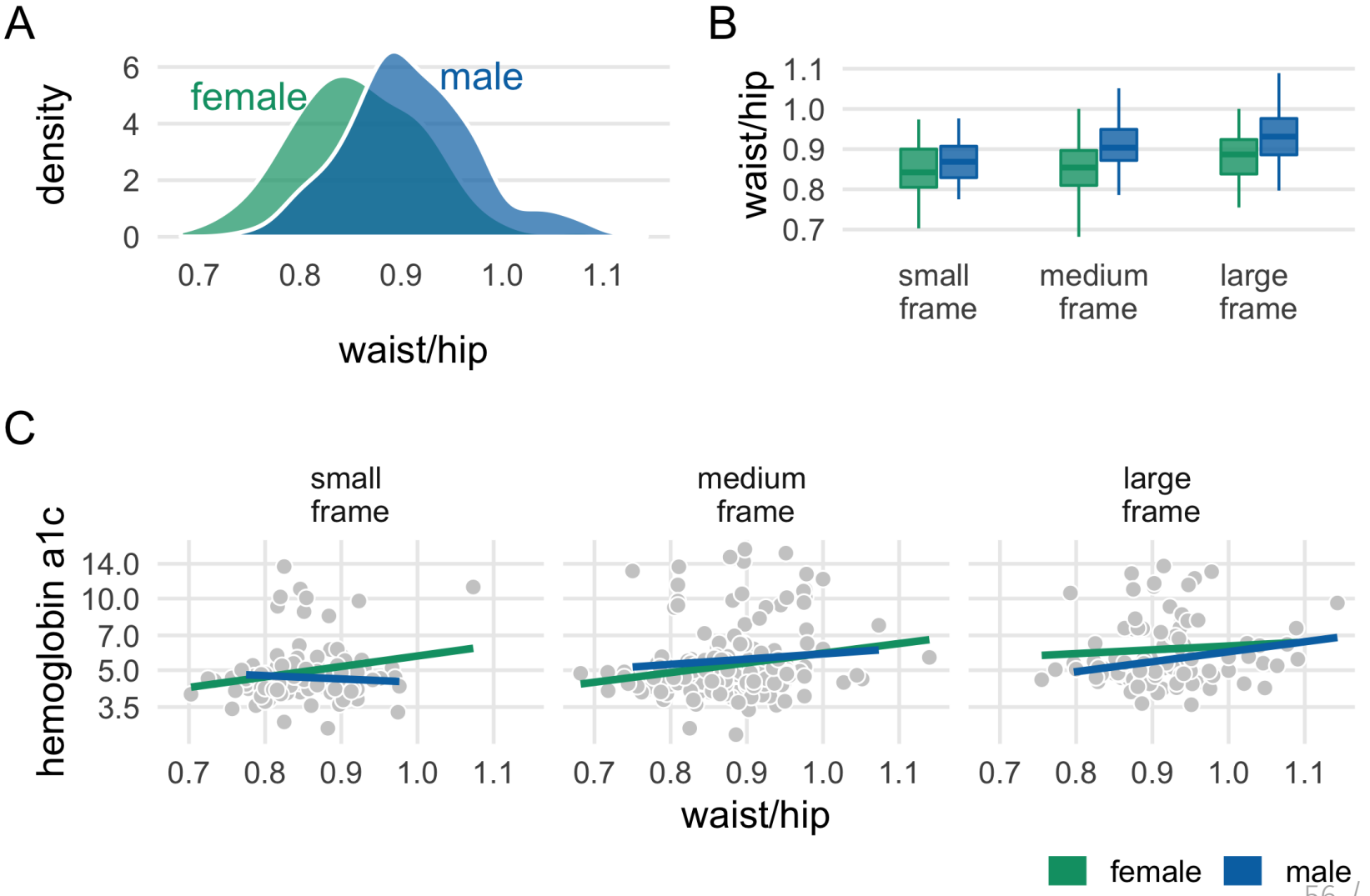
Combining patchwork and cowplot

```
legend <- ggdraw() +
  get_legend(plot_c + theme(legend.position = "bottom"))

(plot_a + plot_b) /
(plot_c + theme(legend.position = "none")) /
legend +
plot_layout(heights = c(10, 10, 1)) +
plot_annotation(
  "The relationship between waist/hip ratio in males and females by
    frame size",
  theme = theme(plot.title = element_text(size = 16, face = "bold"))
)
```

Combining patchwork and cowplot

```
legend <- ggdraw() +
  get_legend(plot_c + theme(legend.position = "bottom"))

(plot_a + plot_b) /
(plot_c + theme(legend.position = "none")) /
legend +
plot_layout(heights = c(10, 10, 1)) +
plot_annotation(
  "The relationship between waist/hip ratio in males and females by
    frame size",
  theme = theme(plot.title = element_text(size = 16, face = "bold"))
)
```

The relationship between waist/hip ratio in males and females by frame size



Don't use too many aesthetics and labels. Be selective.

Don't use too many aesthetics and labels. Be selective.

Use color to focus the reader's attention.

Don't use too many aesthetics and labels. Be selective.

Use color to focus the reader's attention.

Combine plots from simpler to more complex. Be consistent but not boring.

Book Recommendations

Fundamentals of Data Visualization by Claus O. Wilke

Storytelling with Data by Cole Nussbaumer Knaflic

Better Presentations by Jonathan Schwabish

malcolmbarrett

@malco_barrett

Slides created via the R package xaringan.

Slides: malco.io/slides/ggplotline