

```
library(ggplot2)
library(dplyr)
library(ggtext)

plot_data <- ToothGrowth %>%
  mutate(dose = factor(dose)) %>%
  group_by(dose, supp) %>%
  summarise(len = mean(len)) %>%
  ungroup()

# Unstyled plot
ggplot(
  data = plot_data,
  mapping = aes(x = len, y = dose, fill = supp)
) +
  geom_col(position = "dodge")

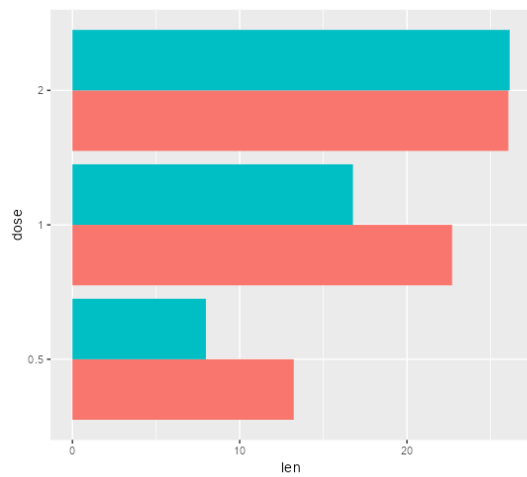
# Styled plot
ggplot(
  data = plot_data,
  mapping = aes(x = len, y = dose, fill = supp)
) +
  geom_col(
    position = position_dodge(width = 0.7),
    width = 0.7
  ) +
```

1

```

scale_x_continuous(
  limits = c(0, 30),
  name = "    "
) +
geom_text(
  mapping = aes(label = round(len, 0)),
  position = position_dodge(width = 0.7),
  hjust = 1.5,
  size = 6,
  fontface = "bold",
  colour = "white"
) +
scale_fill_manual(values = c("#9B1D20", "#3D5A80")) +
labs(
  title = "    ",
  subtitle = "60          C 3          (0.5, 1, 2 mg/ )
          :
          <span style='color: #9B1D20'>**      **</span>
          <span style='color: #3D5A80'>**      **</span>.",
  y = "    (mg/ )"
) +
theme_minimal(base_size = 14) +
theme(
  legend.position = "none",
  plot.title = element_textbox_simple(face = "bold"),
  plot.subtitle = element_textbox_simple(
    margin = margin(t = 10),
    lineheight = 1.5
  ),
  plot.title.position = "plot",
  plot.margin = margin(15, 10, 10, 15),
  panel.grid = element_blank(),
  axis.text.x = element_blank()
)

```



### Tooth Growth

Each of 60 guinea pigs received one of three dose levels of vitamin C (0.5, 1, and 2 mg/day) by one of two delivery methods: **orange juice** or **ascorbic acid**.

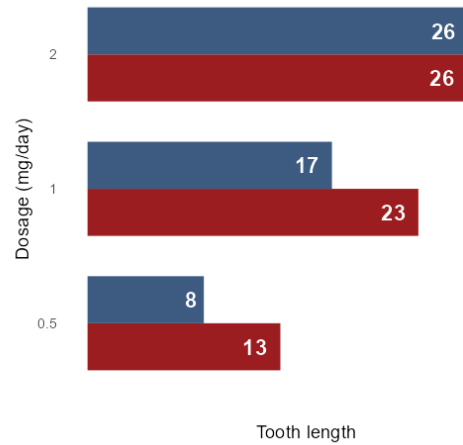


Figure 1:

? Beecham et al. (2021)

```
library(dplyr)
library(ggplot2)
plot_data <- mtcars %>%
  mutate(car = rownames(mtcars))

# Colour all bars
ggplot(
  data = plot_data,
  mapping = aes(
    y = reorder(car, disp),
    x = disp,
    fill = car
  )
) +
  geom_col() +
  labs(
```

```

    x = "  1",
    y = ""
  ) +
  coord_cartesian(expand = FALSE) +
  theme_minimal(base_size = 14) +
  theme(
    legend.position = "none",
    legend.title = element_blank(),
    plot.title = element_text(
      face = "bold",
      margin = margin(b = 10)
    ),
    plot.title.position = "plot",
    plot.margin = margin(15, 10, 10, 15)
  )

# Highlight one bar
ggplot(
  data = plot_data,
  mapping = aes(
    y = reorder(car, disp),
    x = disp,
    fill = (car == "Maserati Bora")
  )
) +
  geom_col() +
  scale_fill_manual(values = c("#AFE1AF", "#7a9d7a")) +
  labs(
    x = "  1",
    y = ""
  ) +
  coord_cartesian(expand = FALSE) +
  theme_minimal(base_size = 14) +
  theme(
    legend.position = "none",
    legend.title = element_blank(),
    plot.title = element_text(
      face = "bold",
      margin = margin(b = 10)
    ),
    plot.title.position = "plot",
    plot.margin = margin(15, 10, 10, 15)
  )

```

)

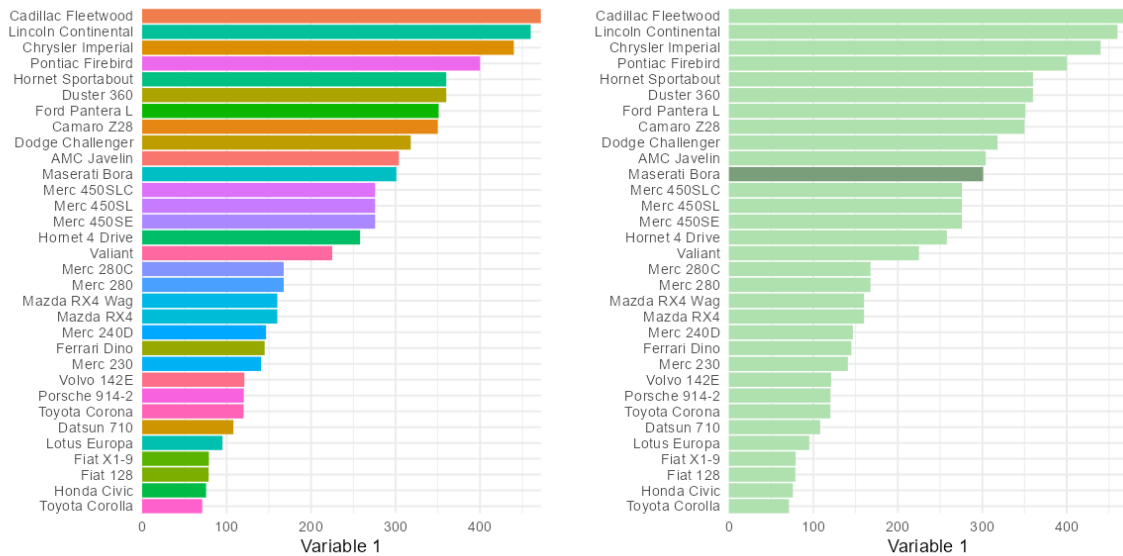


Figure 2:

y

,

( : )

.

```
library(readr)
library(dplyr)
library(tidyr)
library(ggplot2)
wheels <- read_csv(
  "https://raw.githubusercontent.com/rfordatascience/tidytuesday/master/data/2022/2022-08-09,
)
plot_data <- wheels %>%
  select(country, height, diameter) %>%
  drop_na() %>%
  filter(country %in% c("USA", "Japan"))

# Colour only
ggplot(
  data = plot_data,
  mapping = aes(
```

```

    x = diameter,
    y = height,
    colour = country
  )
) +
geom_point(size = 3, alpha = 0.8) +
scale_x_continuous(limits = c(0, 800)) +
scale_y_continuous(limits = c(0, 800)) +
scale_colour_brewer(palette = "Dark2") +
coord_cartesian(expand = FALSE) +
labs(
  title = " ",
  x = " (ft)",
  y = " (ft)"
) +
theme_minimal(base_size = 14) +
theme(
  legend.position = "top",
  legend.title = element_blank(),
  plot.title = element_text(
    face = "bold",
    margin = margin(b = 10)
  ),
  plot.title.position = "plot",
  plot.margin = margin(15, 10, 10, 15)
)

# Shapes and colours
ggplot(
  data = plot_data,
  mapping = aes(
    x = diameter,
    y = height,
    colour = country,
    shape = country
  )
) +
geom_point(size = 3, alpha = 0.8) +
scale_x_continuous(limits = c(0, 800)) +
scale_y_continuous(limits = c(0, 800)) +
scale_colour_brewer(palette = "Dark2") +
coord_cartesian(expand = FALSE) +

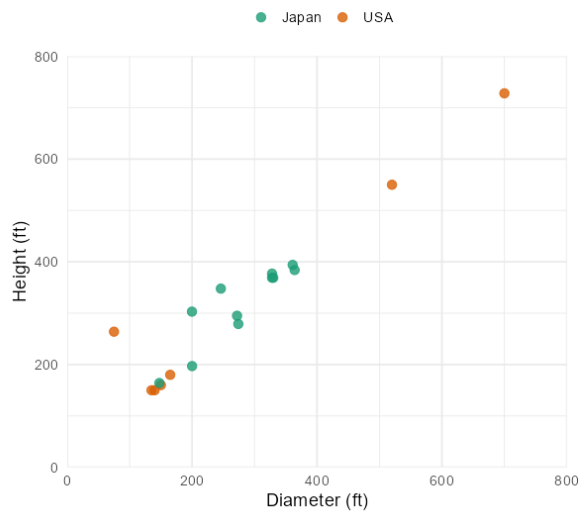
```

```

labs(
  title = " ",
  x = " (ft)",
  y = " (ft)"
) +
theme_minimal(base_size = 14) +
theme(
  legend.position = "top",
  legend.title = element_blank(),
  plot.title = element_text(
    face = "bold",
    margin = margin(b = 10)
  ),
  plot.title.position = "plot",
  plot.margin = margin(15, 10, 10, 15)
)

```

**Ferris wheels**



**Ferris wheels**

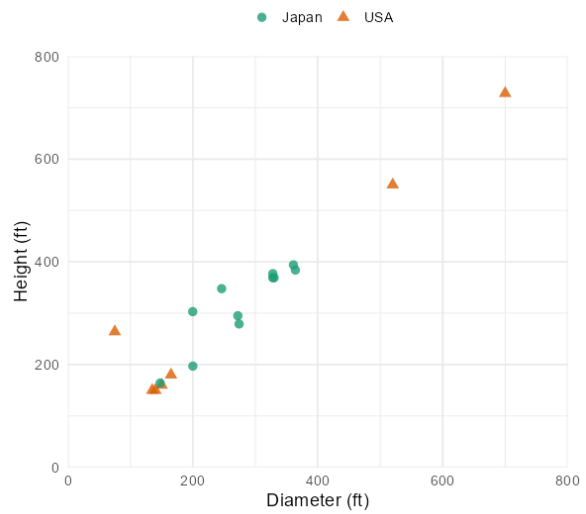


Figure 3:

- (Sequential): ( ) ( : ).
- (Diverging): ( : ).
- (Qualitative): ( : ).

```
library(ggplot2)
library(PrettyCols)

# Sequential
ggplot(
  data = data.frame(x = 1:7, y = 1),
  mapping = aes(x = x, y = y, fill = x)
) +
  geom_tile() +
  labs(title = " (Sequential)") +
  scale_fill_pretty_c("Teals") +
  theme_void() +
  theme(
    legend.position = "none",
    plot.title = element_text(size = 20, face = "bold"),
    plot.margin = margin(15, 10, 10, 15)
  )

# Diverging
ggplot(
  data = data.frame(x = 1:7, y = 1),
  mapping = aes(x = x, y = y, fill = x - mean(x))
) +
  geom_tile() +
  labs(title = " (Diverging)") +
  scale_fill_gradient2(low = "#f1a340", high = "#998ec3") +
  theme_void() +
  theme(
    legend.position = "none",
    plot.title = element_text(size = 20, face = "bold"),
    plot.margin = margin(15, 10, 10, 15)
  )

# Qualitative
ggplot(
  data = data.frame(x = 1:7, y = 1),
  mapping = aes(x = x, y = y, fill = factor(x))
) +
  geom_tile() +
```



```
labs(title = " (Qualitative)") +  
scale_fill_brewer(palette = "Dark2") +  
theme_void() +  
theme(  
  legend.position = "none",  
  plot.title = element_text(size = 20, face = "bold"),  
  plot.margin = margin(15, 10, 10, 15)  
)
```

### Sequential



### Diverging



### Qualitative



Figure 4: .

. (“Coblis — Color Blindness Simulator” n.d.)  
(luminosity) . Tennekes and Puts (2023) .

```

library(ggplot2)
library(dplyr)
library(ggtext)
library(colorblindr)
plot_data <- ToothGrowth %>%
  mutate(dose = factor(dose)) %>%
  group_by(dose, supp) %>%
  summarise(len = mean(len)) %>%
  ungroup()

g <- ggplot(
  data = plot_data,
  mapping = aes(x = len, y = dose, fill = supp)
) +
  geom_col(
    position = position_dodge(width = 0.7),
    width = 0.7
  ) +
  scale_x_continuous(
    limits = c(0, 30),
    name = " "
  ) +
  geom_text(
    mapping = aes(label = round(len, 0)),
    position = position_dodge(width = 0.7),
    hjust = 1.5,
    size = 6,
    fontface = "bold",
    colour = "white"
  ) +
  scale_fill_manual(values = c("#9B1D20", "#3D5A80")) +
  labs(
    title = " ",
    subtitle = "60 C 3 (0.5, 1, 2 mg/ )
      :
      <span style='color: #9B1D20'>** **</span>
      <span style='color: #3D5A80'>** **</span>.",
    y = " (mg/ )"
  ) +
  theme_minimal(base_size = 14) +

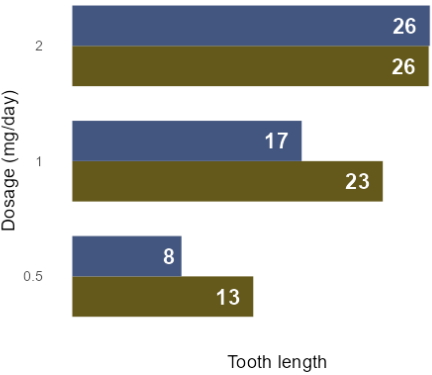
```

```
theme(  
  legend.position = "none",  
  plot.title = element_textbox_simple(face = "bold"),  
  plot.subtitle = element_textbox_simple(  
    margin = margin(t = 10),  
    lineheight = 1.5  
  ),  
  plot.title.position = "plot",  
  plot.margin = margin(15, 10, 10, 15),  
  panel.grid = element_blank(),  
  axis.text.x = element_blank()  
)  
  
cvd_grid(g)
```

Deutanomaly

Tooth Growth

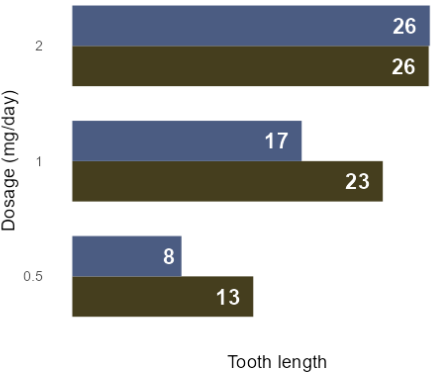
Each of 60 guinea pigs received one of three dose levels of vitamin C (0.5, 1, and 2 mg/day) by one of two delivery methods: orange juice or ascorbic acid.



Protanomaly

Tooth Growth

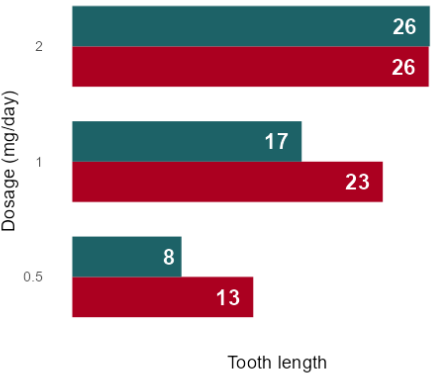
Each of 60 guinea pigs received one of three dose levels of vitamin C (0.5, 1, and 2 mg/day) by one of two delivery methods: orange juice or ascorbic acid.



Tritanomaly

Tooth Growth

Each of 60 guinea pigs received one of three dose levels of vitamin C (0.5, 1, and 2 mg/day) by one of two delivery methods: orange juice or ascorbic acid.



Desaturated

Tooth Growth

Each of 60 guinea pigs received one of three dose levels of vitamin C (0.5, 1, and 2 mg/day) by one of two delivery methods: orange juice or ascorbic acid.

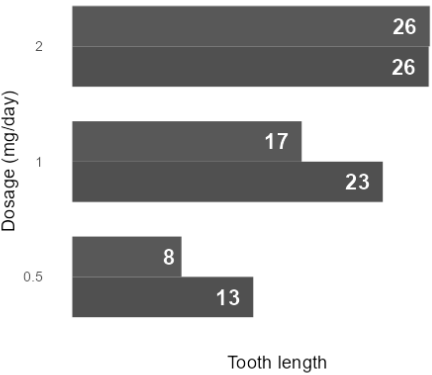


Figure 5:

Muth (2018)

- 
- 
- 
- 
- 

y ( )

```
library(ggplot2)
library(dplyr)
plot_data <- ToothGrowth %>%
  mutate(dose = factor(dose)) %>%
  group_by(dose, supp) %>%
  summarise(len = mean(len)) %>%
  ungroup()

# Not annotated
ggplot(
  data = plot_data,
  mapping = aes(
    x = len,
    y = dose,
    fill = supp
  )
) +
  geom_col(
    position = position_dodge(width = 0.7),
    width = 0.7
  ) +
  scale_x_continuous(
    limits = c(0, 30),
    name = " "
  ) +
```

```

scale_fill_manual(
  name = " : ",
  values = c("#9B1D20", "#3D5A80")
) +
labs(
  title = " ",
  y = " "
) +
theme_minimal(base_size = 14) +
theme(
  legend.position = "top",
  plot.title = element_text(face = "bold"),
  plot.title.position = "plot",
  plot.margin = margin(15, 10, 10, 15)
)

# Annotated
ggplot(
  data = plot_data,
  mapping = aes(
    x = len,
    y = dose,
    fill = supp
  )
) +
geom_col(
  position = position_dodge(width = 0.7),
  width = 0.7
) +
scale_x_continuous(
  limits = c(0, 30),
  name = " "
) +
geom_text(
  mapping = aes(label = round(len, 0)),
  position = position_dodge(width = 0.7),
  hjust = 1.5,
  size = 6,
  fontface = "bold",
  colour = "white"
) +
scale_fill_manual(

```

```

name = " : ",
values = c("#9B1D20", "#3D5A80")
) +
labs(
  title = " ",
  y = " "
) +
theme_minimal(base_size = 14) +
theme(
  legend.position = "top",
  plot.title = element_text(face = "bold"),
  plot.title.position = "plot",
  plot.margin = margin(15, 10, 10, 15)
)

```

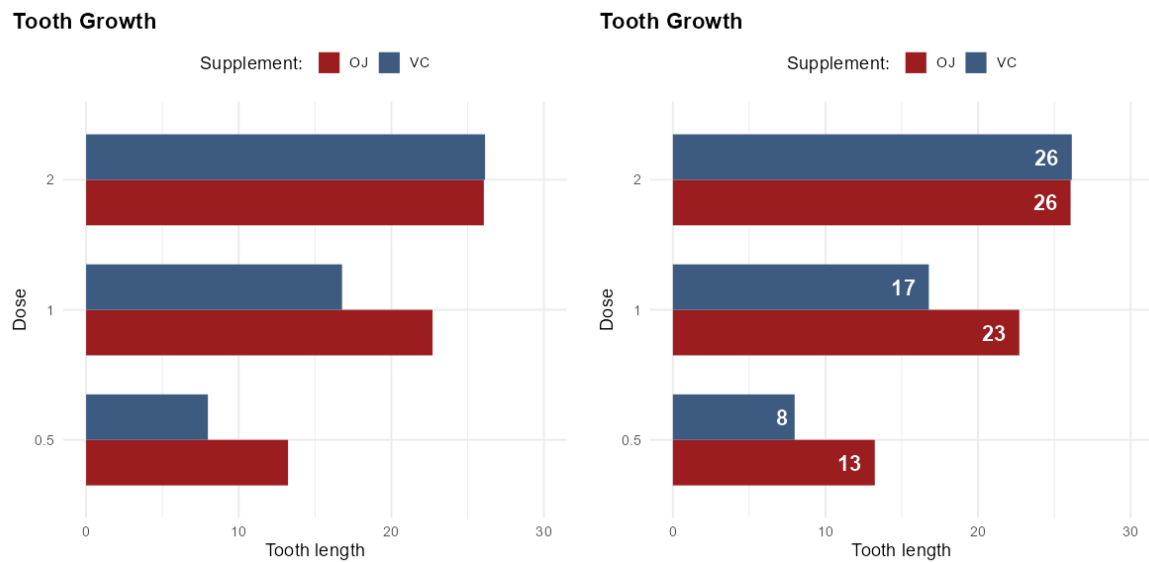


Figure 6:

- - 
  -
- 2008
- 2008
- “Chart Titles and Text” (n.d.)
- 3~4



- •  
•

•

.

,

.



,

.

12pt

36pt

?

- (Serif):  . ( )
- (Sans serif):  . ( )
- (Monospace):  .

$$\left( \begin{array}{c} \text{ } \\ \text{ } \end{array} \right)$$

,

( : Times New Roman)

•

. Arial, Calibri, Verdana

Dyslexie    OpenDyslexic

Hyperlegible Braille Institute of America

Institute      Google Fonts

Wery and Diliberto

(2017).

,

$$\mathbb{I}, \quad \mathbb{I}, \quad 1$$

. Source

Sans Pro Verdana

(Rello and Baeza-Yates 2016).

- 12pt
- I, 1, 1
- (2 )
- ( )

(Alternative text )

Green (2023) Mine Dogucu

. Cesal (2020)

- Beecham, Roger, Jason Dykes, Layik Hama, and Nik Lomax. 2021. “On the Use of ‘Glyphmaps’ for Analysing the Scale and Temporal Spread of COVID-19 Reported Cases.” *ISPRS International Journal of Geo-Information* 10 (4). <https://doi.org/10.3390/ijgi10040213>.
- Cesal, Amy. 2020. “Writing Alt Text for Data Visualization.” Nightingale. 2020. <https://medium.com/nightingale/writing-alt-text-for-data-visualization-2a218ef43f81>.
- “Chart Titles and Text.” n.d. Office for National Statistics. Accessed July 10, 2023. <https://style.ons.gov.uk/data-visualisation/titles-and-text/annotation-and-footnotes/>.
- “Coblis — Color Blindness Simulator.” n.d. Colblindor. Accessed July 10, 2023. <https://www.color-blindness.com/coblis-color-blindness-simulator/>.
- Green, Nathan. 2023. “Why your data viz needs alt text.” *Significance* 20 (1): 38–39. <https://doi.org/10.1093/jrssig/qmad011>.
- Muth, Lisa. 2018. “An Alternative to Pink & Blue: Colors for Gender Data.” Datawrapper. 2018. <https://blog.datawrapper.de/gendercolor/>.
- Rello, Luz, and Ricardo Baeza-Yates. 2016. “The Effect of Font Type on Screen Readability by People with Dyslexia.” *ACM Trans. Access. Comput.* 8 (4). <https://doi.org/10.1145/2897736>.
- Tennekes, Martijn, and Marco J. H. Puts. 2023. “cols4all: a Color Palette Analysis Tool.” In *EuroVis 2023 - Short Papers*, edited by Thomas Hoellt, Wolfgang Aigner, and Bei Wang. The Eurographics Association. <https://doi.org/10.2312/evs.20231040>.
- Tol, Paul. 2021. “Introduction to Colour Schemes.” 2021. <https://personal.sron.nl/~pault/>.
- Wery, J. J., and J. A. Diliberto. 2017. “The Effect of a Specialized Dyslexia Font, OpenDyslexic, on Reading Rate and Accuracy.” *Ann. Of Dyslexia*. 67: 114–27. <https://doi.org/10.1007/s11881-016-0127-1>.