Problem Set 3

Data Visualisation for Social Scientists

Due: February 18, 2026

Instructions

- Please show your work! You may lose points by simply writing in the answer. If the problem requires you to execute commands in `R`, please include the code you used to get your answers. Please also include the `.R` file that contains your code. If you are not sure if work needs to be shown for a particular problem, please ask.
- Your homework should be submitted electronically on GitHub.
- This problem set is due before 23:59 on Wednesday February 18, 2026. No late assignments will be accepted.

Canadian Election Study

The data for this problem set come from the Canadian Election Study (CES) in 2015. The main purpose of the study is to give a comprehensive picture of the Canadian election: why people vote as they do, what changes during campaigns and across elections, and how Canadian voting compares with that in other democracies.

Data Manipulation

1. Load the CES .csv file from GitHub into your global environment. Filter respondents to only include "high quality" participants:

   ```
   ces2015 <- ces2015 |> filter(discard == "Good quality")
   ```

2. Filter the dataset to those participants that answered the question about voting for the past election using `p_voted`. Consider respondents who gave a "Yes" answer as having voted, while "No" as not having voted. Treat "Don't know" and "Refused" as missing.

```r
data$p_voted[data$p_voted == "Yes"] <- 1
data$p_voted[data$p_voted == "No"] <- 0
data$p_voted <- as.numeric(ifelse(data$p_voted == 1 | data$p_voted == 0,
                                  data$p_voted, NA))
```

3. Create an age variable and group into categories (e.g., <30, 30-44, 45-64, 65+). Year of birth is in age (four-digit year).

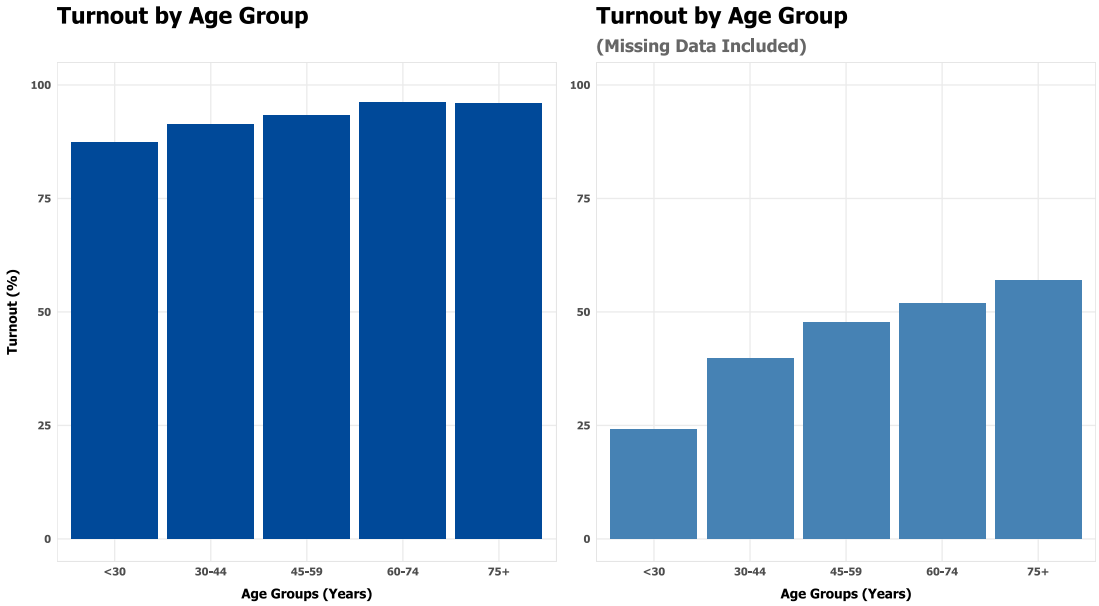
```
data$date <- dmy(data$date)
data$age_groups <- as.numeric(format(data$date, "%Y")) - as.numeric(data$
    age)
data <- data %>% mutate(age_groups = case_when(
  age_groups < 30 ~ "<30",
  age_groups >= 30 & age_groups < 45 ~ "30-44",
  age_groups >= 45 & age_groups < 60 ~ "45-59",
  age_groups >= 60 & age_groups < 75 ~ "60-74",
  age_groups >= 75 ~ "75+"
))
```

Data Visualization

1. Plot turnout rate by age group.

```
turnout_data <- data %>%
  drop_na(age_groups) %>%
  group_by(age_groups) %>%
  summarize(turnout = mean(p_voted, na.rm = TRUE))

no_nas <- ggplot(turnout_data, aes(x = age_groups, y = turnout*100)) +
  geom_col(fill = "#004999") +
  ylim(0, 100) +
  theme(plot.title = element_text(hjust = 0.5)) +
  labs(x = "Age Groups (Years)",
       y = "Turnout (%)",
       title = "Turnout by Age Group")

turnout_data_nas <- data %>%
  mutate(p_voted = case_when(
    is.na(p_voted) ~ 0,
    !is.na(p_voted) ~ p_voted
  )) %>%
  drop_na(age_groups) %>%
  group_by(age_groups) %>%
  summarize(turnout = mean(p_voted))

w_nas <- ggplot(turnout_data_nas, aes(x = age_groups, y = turnout*100)) +
  geom_col(fill = "steelblue") +
  ylim(0, 100) +
  theme(plot.title = element_text(hjust = 0.5)) +
  labs(x = "Age Groups (Years)",
       y = element_blank(),
       title = "Turnout by Age Group",
       subtitle = "(Missing Data Included)")

turnout_graphs <- no_nas + SVU + w_nas + SVU
```

Figure 1: Turnout Percentages

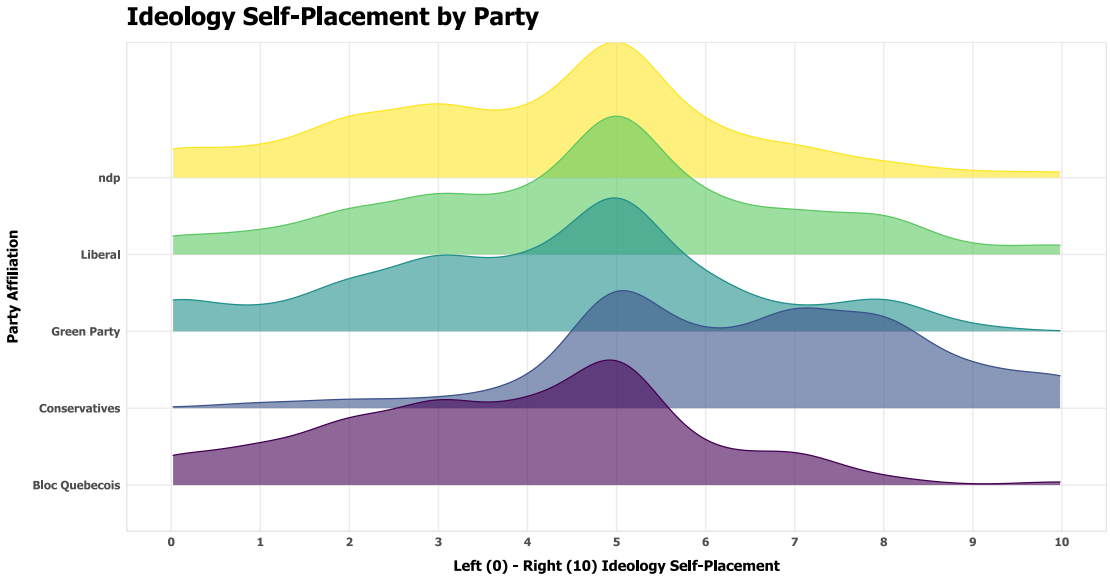


2. Create a density plot of ideology by party, restricting your sample to respondents with non-missing left–right self-placement (0–10 scale) and those that intended to vote for a main party (e.g., Liberal, Conservative, NDP, Bloc in Quebec, and Green).

```r
q2_data <- data %>% mutate(p_selfplace = as.numeric(p_selfplace))
q2_data <- q2_data %>%
  filter(!is.na(p_selfplace) & !is.na(vote_for)) %>%
  filter(p_selfplace != 1000) %>%
  filter(vote_for == "Liberal" | vote_for == "Conservatives" |
           vote_for == "Bloc Quebecois" | vote_for == "Green Party" |
           vote_for == "ndp")

plot_2.1 <- ggplot(q2_data %>% group_by(vote_for), aes(x = p_selfplace,
                                                       y = vote_for,
                                                       fill = vote_for,
                                                       color = vote_for)) +
  geom_density_ridges(alpha = 0.6) +
  theme(legend.position = "none",
        plot.title = element_text(hjust = 0.5)) +
  labs(x = "Left (0) - Right (10) Ideology Self-Placement",
       y = "Party Affiliation",
       title = "Ideology Self-Placement by Party") +
  scale_x_continuous(
    breaks = seq(0, 10, by = 1),
    limits = c(0, 10)) +
  scale_fill_viridis_d() +
  scale_color_viridis_d()

```

```
25 plot_2.2 <- plot_2.1 + SVU + theme(legend.position = "none")
```

Figure 2: Self-Placement on Left-Right Political Scale by Region



3. Produce histogram counts of turnout by income (`income_full`), faceted by province.

   Data wrangling:

```
prov_data <- data %>%
  filter(income_full != ".d" & income_full != ".r" &
         !is.na(income_full) & province != 1000 &
         province != "Yukon" & province != "nunavut" &
         province != "Quebec" & province != "Ontario")

ont_que_data <- data %>% filter(income_full != ".d" & income_full != ".r" &
                                  !is.na(income_full))
ont_que_data <- ont_que_data %>%
  filter(province == "Quebec" | province == "Ontario")
prov_data$income_full <- factor(prov_data$income_full,
                                ordered = TRUE, levels = c(
  "less than $29,999", "between $30,000 and $59,999",
  "between $60,000 and $89,999", "between $90,000 and $109,999",
  "more than $110,000"))
ont_que_data$income_full <- factor(ont_que_data$income_full,
                                   ordered = TRUE, levels = c(
  "less than $29,999", "between $30,000 and $59,999",
  "between $60,000 and $89,999", "between $90,000 and $109,999",
  "more than $110,000"))
```

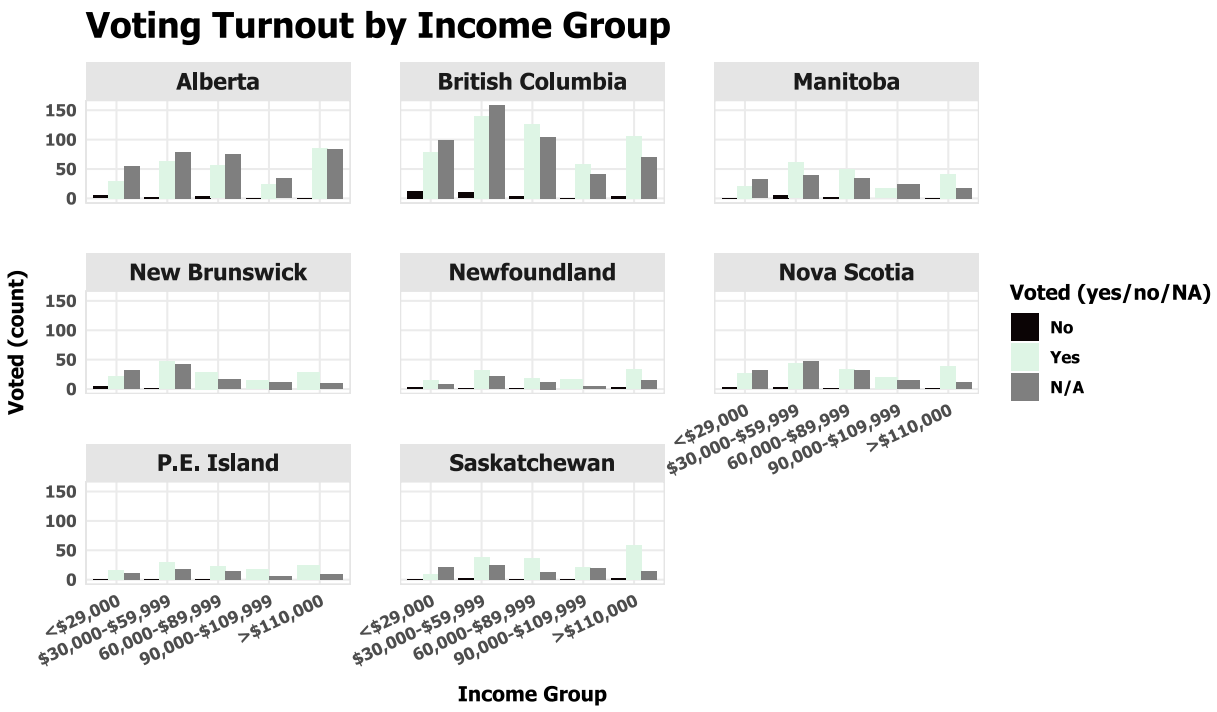
   Plot 1:

```
province_labs <- c("Alberta", "British Columbia", "Manitoba",
                   "New Brunswick", "Newfoundland", "Nova Scotia",
                   "P.E. Island", "Saskatchewan")
names(province_labs) <- c("Alberta", "bc", "Manitoba", "nb", "Nfld",
                          "ns", "pei", "Sask")

prov_na <- ggplot(prov_data, aes(x = income_full, fill = factor(p_voted))) +
  geom_histogram(stat = "count", position = "dodge") +
  scale_x_discrete(labels = c("less than $29,999" = "<$29,000",
                              "between $30,000 and $59,999" = "$30,000-$59,999",
                              "between $60,000 and $89,999" = "60,000-$89,999",
                              "between $90,000 and $109,999" = "90,000-$109,999",
                              "more than $110,000" = ">$110,000")) +
  facet_wrap(~province, labeller = labeller(province = province_labs)) +
  theme(axis.title.x = element_blank(),
        plot.title = element_text(hjust = 0.5),
        axis.text.x = element_text(angle = 45, hjust = 1, vjust = 1)) +
  labs(
    x = "Income Group",
    title = "Voting Turnout by Income Group",
    y = "Voted (count)",
```

```
    fill = "Voted (yes/no/NA)"
  ) +
  scale_fill_discrete(labels = c("No", "Yes", "N/A"))
```

```
plot_3.2 <- prov_na + SVU + theme(axis.text.x = element_text(
  angle = 25, hjust = 1, vjust = 1),
  panel.spacing = unit(2, "lines"))
```

Figure 3: Voting Turnout Lower Population Provinces, NA's included

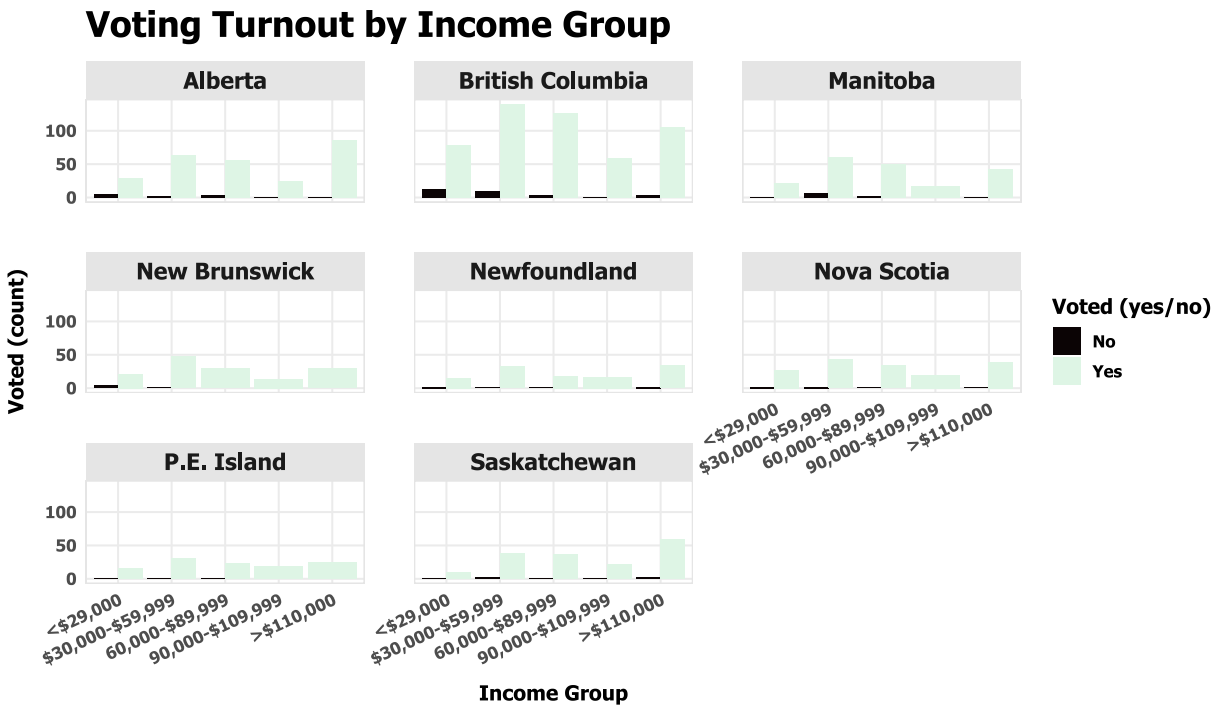


Plot 2:

```
prov_base <- ggplot(prov_data %>% filter(!is.na(p_voted)),
       aes(x = income_full, fill = factor(p_voted))) +
  geom_histogram(stat = "count", position = "dodge") +
  scale_x_discrete(labels = c("less than $29,999" = "<$29,000",
                              "between $30,000 and $59,999" = "$30,000-$
    59,999",
                              "between $60,000 and $89,999" = "60,000-$
    89,999",
                              "between $90,000 and $109,999" = "90,000-$
    109,999",
                              "more than $110,000" = ">$110,000")) +
  facet_wrap(~province, labeller = labeller(province = province_labs)) +
  theme(axis.title.x = element_blank(),
        plot.title = element_text(hjust = 0.5),
        axis.text.x = element_text(angle = 45, hjust = 1, vjust = 1)) +
  labs(
    x = "Income Group",
    title = "Voting Turnout by Income Group",
    y = "Voted (count)",
    fill = "Voted (yes/no)"
  ) +
  scale_fill_discrete(labels = c("No", "Yes"))
```

```
plot_3.1 <- prov_base + SVU + theme(axis.text.x = element_text(
  angle = 25, hjust = 1, vjust = 1),
  panel.spacing = unit(2, "lines"))
```

Figure 4: Voting Turnout Lower Population Provinces, NA's **not included**

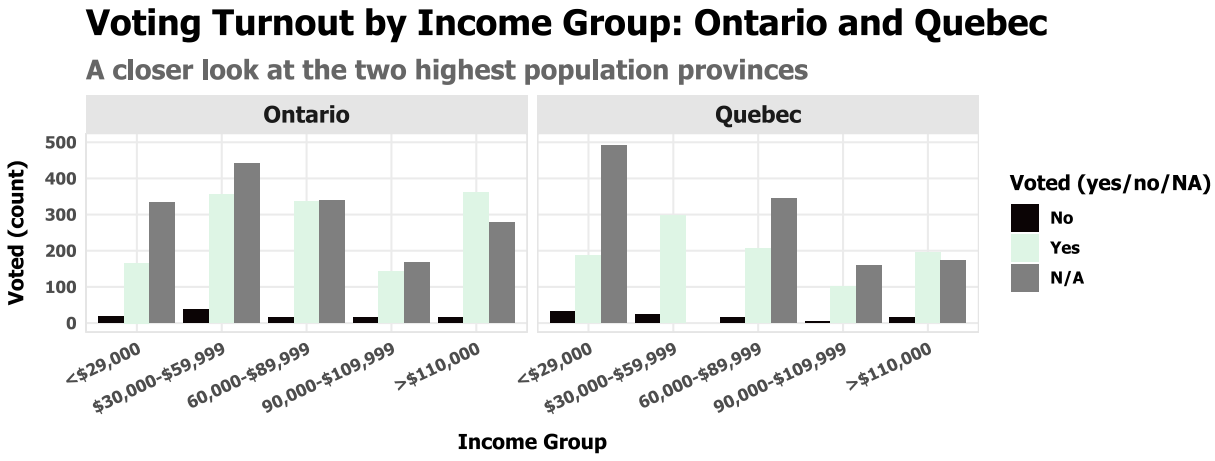


Plot 3:

```
ont_que_na <- ggplot(ont_que_data, aes(x = income_full, fill = factor(p_
    voted))) +
  geom_bar(position = "dodge") +
  scale_x_discrete(labels = c("less than $29,999" = "<$29,000",
                              "between $30,000 and $59,999" = "$30,000-$
    59,999",
                              "between $60,000 and $89,999" = "60,000-$
    89,999",
                              "between $90,000 and $109,999" = "90,000-$
    109,999",
                              "more than $110,000" = ">$110,000")) +
  facet_wrap(~province) +
  theme(axis.title.x = element_blank(),
        plot.title = element_text(hjust = 0.5),
        axis.text.x = element_text(angle = 45, hjust = 1, vjust = 1)) +
  ylim(0, 500) +
  labs(
    x = "Income Group",
    title = "Voting Turnout by Income Group: Ontario and Quebec",
    subtitle = "A closer look at the two highest population provinces",
    y = "Voted (count)",
    fill = "Voted (yes/no/NA)"
  ) +
```

```
  scale_fill_discrete(labels = c("No", "Yes", "N/A"))

plot_3.4 <- ont_que_na + SVU + theme(axis.text.x = element_text(
  angle = 25, hjust = 1, vjust = 1))
```

Figure 5: Voting Turnout Higher Population Provinces, NA's included

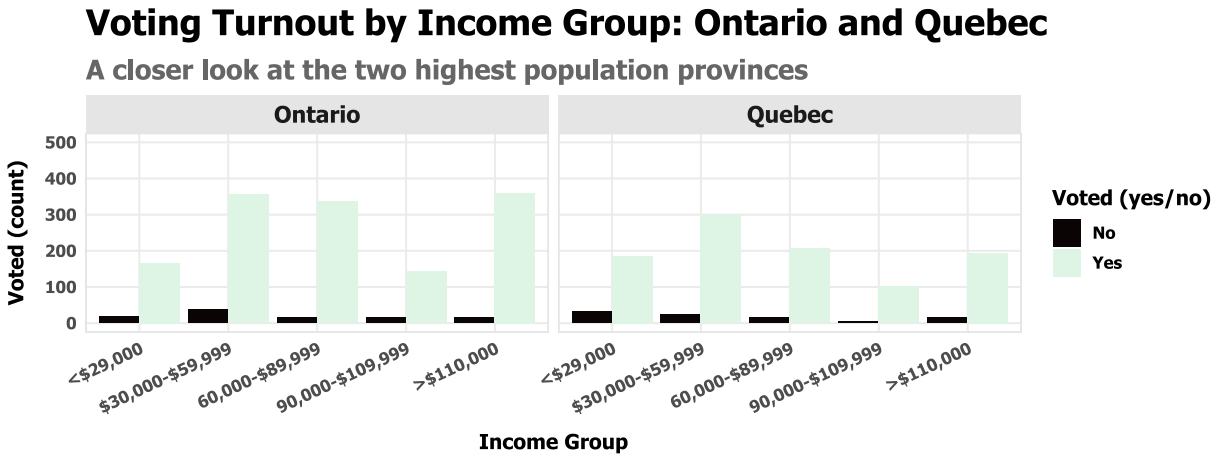


Plot 4:

```
ont_que <- ggplot(ont_que_data %>% filter(!is.na(p_voted)),
  aes(x = income_full, fill = factor(p_voted))) +
  geom_histogram(stat = "count", position = "dodge") +
  scale_x_discrete(labels = c("less than $29,999" = "<$29,000",
                              "between $30,000 and $59,999" = "$30,000-$
    59,999",
                              "between $60,000 and $89,999" = "60,000-$
    89,999",
                              "between $90,000 and $109,999" = "90,000-$
    109,999",
                              "more than $110,000" = ">$110,000")) +
  facet_wrap(~province) +
  theme(axis.title.x = element_blank(),
        plot.title = element_text(hjust = 0.5),
        axis.text.x = element_text(angle = 45, hjust = 1, vjust = 1)) +
  ylim(0, 500) +
  labs(
    x = "Income Group",
    title = "Voting Turnout by Income Group: Ontario and Quebec",
    subtitle = "A closer look at the two highest population provinces",
    y = "Voted (count)",
    fill = "Voted (yes/no)"
  ) +
  scale_fill_discrete(labels = c("No", "Yes"))
```

```
plot_3.3 <- ont_que + SVU + theme(axis.text.x = element_text(
  angle = 25, hjust = 1, vjust = 1))
```

Figure 6: Voting Turnout Higher Population Provinces, NA's **not included**



4. Create your own reusable custom theme. Apply your theme to one of the previous plots and add:

   (a) An improved title summarizing the main substantive takeaway.

   (b) A more informative subtitle describing the sample and variables.

   (c) A caption noting data source, weighting, and key coding decisions.

   (d) At least one direct annotation using `ggrepel` that calls out a key pattern.

```r
SVU <- theme_minimal(base_family = "Tahoma", base_size = 12) +
  theme(panel.grid.minor = element_blank(),
        # Bold, bigger title
        plot.title = element_text(margin = margin(t = 10, b = 10),
                                  face = "bold", size = rel(1.7)),
        # Plain, slightly bigger subtitle that is grey
        plot.subtitle = element_text(face = "plain", size = rel(1.3),
                                     color = "grey40"),
        # Italic, smaller, grey caption that is left-aligned
        plot.caption = element_text(face = "italic", size = rel(0.7),
                                    color = "grey70", hjust = 0),
        # Bold legend titles
        legend.title = element_text(face = "bold"),
        # Bold, slightly larger facet titles that are centered
        strip.text = element_text(face = "bold",
                                  size = rel(1.1), hjust = 0.5),
        # Bold axis titles
        axis.title = element_text(face = "bold"),
        # Add some space above the x-axis title
        axis.title.x = element_text(margin = margin(t = 10,
                                                    b = 10), hjust = 0.5)
    ,
        # Add some space to the right of the y-axis title
        axis.title.y = element_text(margin = margin(r = 10,
                                                    l = 8), hjust = 0.5),
        # Facet titles grey background
        strip.background = element_rect(fill = "grey90", color = NA),
        # Facet panel border
        panel.border = element_rect(color = "grey90", fill = NA),
        # color / fill for discrete scales
        palette.colour.discrete = function(n)
          scales::pal_viridis(option = "G")(n),
        palette.fill.discrete = function(n)
          scales::pal_viridis(option = "G")(n),
        # color / fill for continuous scales
        palette.colour.continuous = scales::pal_viridis(option = "G"),
        palette.fill.continuous = scales::pal_viridis(option = "G"))
```