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")
   ```

   ```
   # DM 1: Load Data #
   ces2015 <- read_csv("~/GitHub/DataViz_2026/datasets/CES2015.csv")
   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
# DM 2:  Filter by answered question
unique(ces2015$p_voted) #unique answers incl NA/ "1000"
table(ces2015$p_voted) #see rough split to make sure its right
ces2015 <- ces2015 %>%
  filter(p_voted %in% c("Yes", "No")) #filter only to those who responded yes or no
```

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).

```r
# DM 3: Create Age Variable
str(ces2015$age) #character need to convert integer
table(ces2015$age) #check for missing values / potentially miscoded entries (i.e 1000)

ces2015 <- ces2015 %>%
  filter(!age %in% c("refused", "don't know", "1000")) %>% #remove nonvalid yobs
  mutate(
    age = as.integer(age), #convert to numeric
    age = 2015 - age, #years old as of 2015
    cat_age = cut( #create categorical variable
          age,
          breaks = c(0, 25, 35, 45, 55, 65, 75, 150), #range breaks
          labels= c("18-25", "26-35", "36-45", "46-55","56-65", "65-75", "<75")
        )
    )

table(ces2015$age) #check age range looks accurate
table(ces2015$cat_age) #view categories
```

Data Visualization

Create random palette to use for some of the graphs:

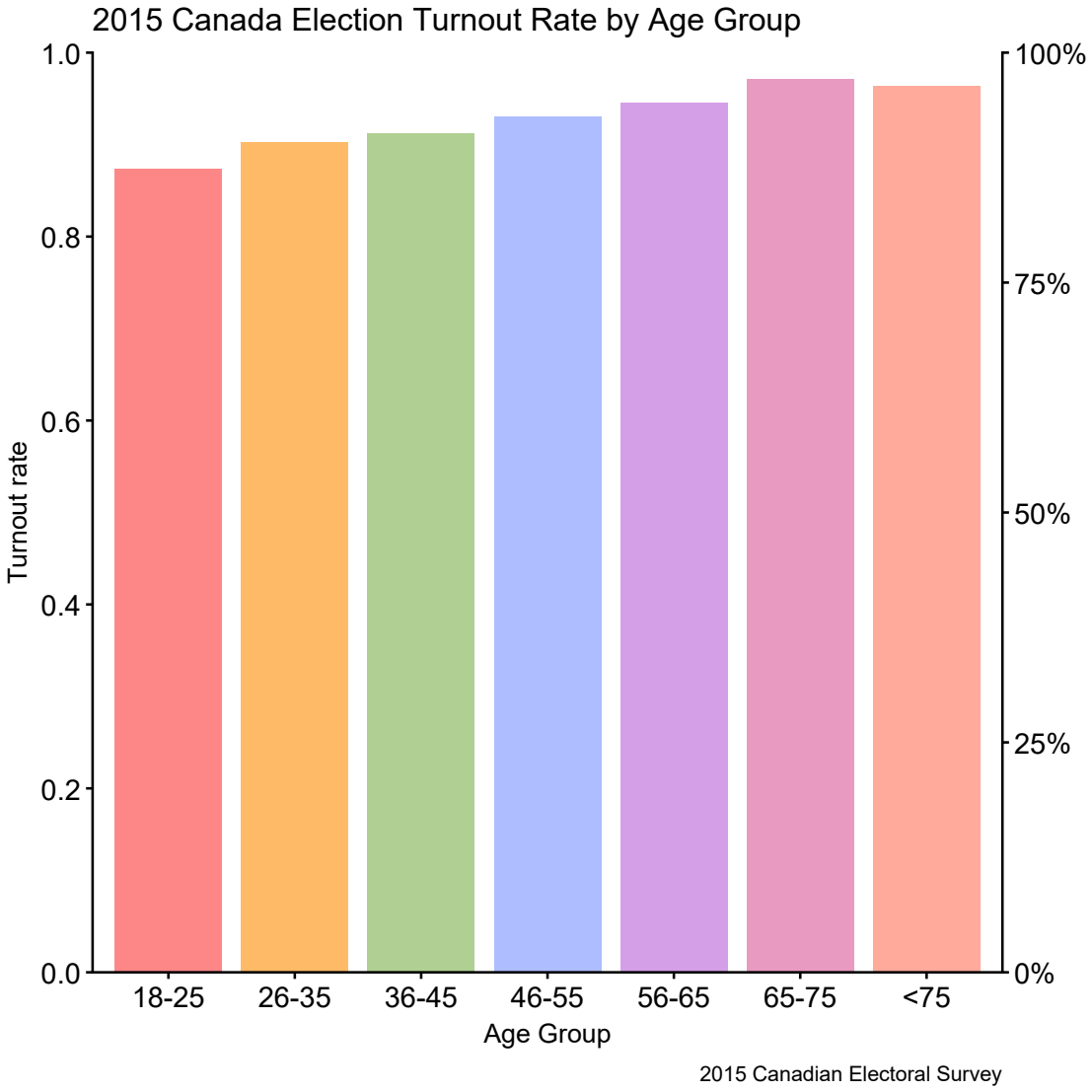
```r
scale_fill_eb <- function(...) {
  scale_fill_manual(
    values = c(
      "#fd3838","#ff8c02", "#7cb04b",  "#7792ff",  "#b960da", "#da5799","#ff7257","#c83131", "#e17421", "#399e72", "#5e72c5",  "#9b5ab2",  "#a72264",
    "#ff7257", "#ffb050", "#a4eac2", "#b6c3e8", "#e3aaf8","#ed90af","#fd3838","#ff8c02", "#7cb04b",  "#7792ff",  "#b960da"
    ),
    ...
  )
}
```

1. Plot turnout rate by age group. First need to create statistics summary for turnout for each age group created earlier.

```r
# DV 1: Turnout rate by age group
turnout_agegroup<- ces2015 %>%
  filter(!is.na(cat_age)) %>% #filter na age categories
  group_by(cat_age) %>% #group within age category to
  summarise(turnout = mean(p_voted == "Yes", na.rm = TRUE)) #turnout rate
    as yes voted vs no (filtered earlier)
```

Plot using a bar chart. As age categories were already in ascending order when plotted they remain so. Add a second y-axis for fun to see both percentage and proportion forms.

```r
plot1 <- ggplot(data = turnout_agegroup, aes(x = cat_age, y = turnout,
    fill = cat_age)) +
  geom_col(alpha = 0.6, width = 0.85) +
  scale_y_continuous(
    limits = c(0,1),
    breaks = seq(0, 1, by  = 0.2),
    expand = c(0,0),
    sec.axis = sec_axis(~. * 100,  #add second axis for fun transform to
    percentage
                        labels = function(x) paste0(x, "%")) #add % label
  ) +
  scale_fill_eb() +
  theme_classic(base_size = 12, base_family = "Arial") +
  labs(x = "Age Group", y = "Turnout rate",
       title = "2015 Canada Election Turnout Rate by Age Group", caption
    = "2015 Canadian Electoral Survey") +
  guides(fill = "none") +
  theme(
    axis.text.y = element_text(size = 13),
    axis.text.x = element_text(size = 13)
  )
ggsave("PS03_p1.pdf", plot1, device = cairo_pdf)
```


2015 Canada Election Turnout Rate by Age Group
1.0
0.8
0.6
0.4
0.2
0.0
Turnout rate
100%
75%
50%
25%
0%
18-25
26-35
36-45
46-55
56-65
65-75
<75
Age Group
2015 Canadian Electoral Survey

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).
Data exploration on the structure and values within key variables. Transform relevant variables into correct form before removing missing values and specifying only the main parties to be kept. Then ordering the party intentions by the mean value of the left-right placement so when plotted it is easier to see the pattern.

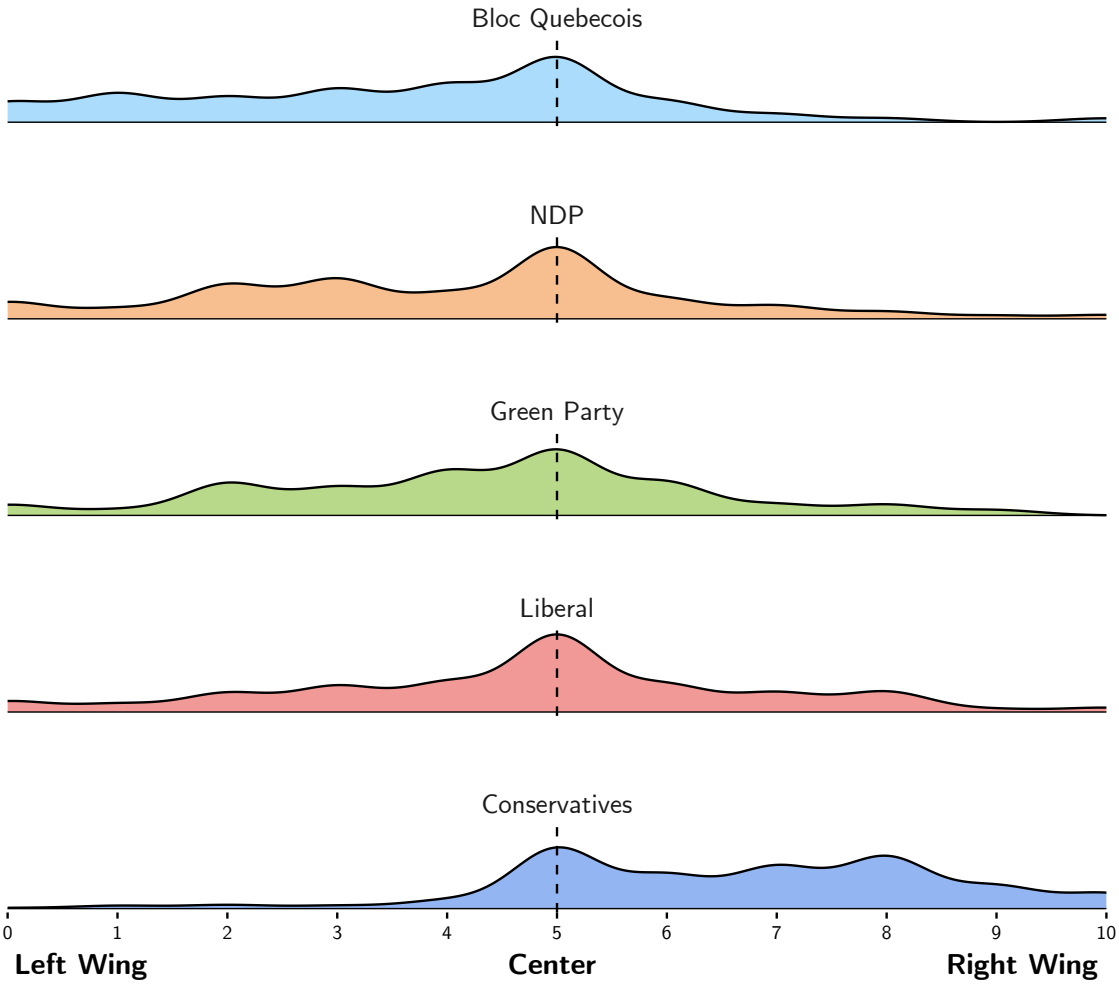
```
# DV 2: Density Plot of ideology by party
table(is.na(ces2015$p_selfplace)) #number missing values for selfplace
unique(ces2015$p_selfplace) #can see NA and 1000 out of range
unique(ces2015$vote_for) #party intend to vote for, see spellings/missing
    values
table(ces2015$vote_for) #baseline to check for after


ideo_by_party <- ces2015 %>%
  mutate(
    p_selfplace = as.integer(p_selfplace), #turn to integer
    vote_for = as.factor(vote_for) #transform to factor
  ) %>%
  filter(!is.na(p_selfplace), p_selfplace != "1000") %>% #remove na and
    1000 values
  filter(vote_for %in% c("Liberal", "ndp", "Green Party", "Conservatives"
    , "Bloc Quebecois")) %>%
  mutate(vote_for = recode(factor(vote_for), "ndp" = "NDP"))

table(ideo_by_party$vote_for) #check looks correct and parties
table(ideo_by_party$p_selfplace)
unique(ideo_by_party$vote_for)

#Order by mean left-right
ideo_by_party$vote_for <- reorder(ideo_by_party$vote_for, ideo_by_party$p
    _selfplace, FUN = mean)
```

Plot using faceted density plots with only one column to line up the x-axis for comparison. Looks similar to ggridge density plots but the facet title is in the center of each for a cleaner look. The hexcode colours for each party were extracted from the Wikipedia page. The bin-width was selected to see a semi-smooth density (due to integer scores) but enough to see the general trend. As adding annotations would appear for every facet, using spaces within the x-axis title (that appears once) to orient the left-right labels on the axis.

```
plot2 <- ggplot(ideo_by_party, aes(x= p_selfplace, fill= vote_for)) +
  geom_density(alpha = 0.7, bw = 0.4) + #scale height of factors
  geom_vline(xintercept = 5, linetype = "dashed") +
  geom_hline(yintercept = 0, linewidth = 0.3) +
  facet_wrap(vars(vote_for), nrow = 5)+
  coord_cartesian(xlim = c(0,10)) +
  scale_x_continuous(limits = c(0,10), #limit
```

```
                     breaks = 0:10, #add label for each integer
                     expand = c(0,0)) + #hard boundary with no excess
  scale_fill_manual(values = c( #manually set colours to hexcodes
    extracted from wiki page
     "Liberal" = "#EA6D6A", #correct party colours for clarity
     "NDP" = "#F4A460",
     "Green Party" = "#99C955",
     "Conservatives" = "#6495ED",
     "Bloc Quebecois" = "#87CEFA")) +
  labs(x = "Left Wing                                    Center
                          Right Wing",
       title = "Left-Right Self-Placement of Individuals Indending to
    Vote for the Main Parties\n",
       caption = "2015 Canadian Election Survey political orientation
    score of intended party voters") +
  guides(fill = "none") + #remove clutered legend
  theme_classic(base_family = "CMU Sans Serif") +
  theme(
    axis.text.y = element_blank(), #rmv y axis
    axis.title.y = element_blank(),
    axis.line.y = element_blank(),
    axis.line.x = element_blank(),
    axis.title.x = element_text(size = 13, face = "bold"), #bold the
    xaxis title
    strip.background = element_blank(), #remove facet border
    axis.ticks.y = element_blank(),
    plot.title = element_text(hjust = 0.5), #center plot title
    strip.text.x = element_text(size = 12, hjust = 0.5), #hjust = title
    in center of factet
    panel.spacing = unit(1, "cm")  #add space between facets to make
    clear title and plot
  )
ggsave("PS03_p2.pdf", plot2, device = cairo_pdf)
```

Left-Right Self-Placement of Individuals Indending to Vote for the Main Parties



2015 Canadian Election Survey political orientation score of intended party voters

3. Produce histogram counts of turnout by income (`income_full`), faceted by province. Key data manipulation here to clean data and get clear factors for income and province (labelled correctly). Removing missing values, before ordering income by income level (starting at lowest category) and province by size (i.e. by length when grouped) that will order the facets when plotting. Then creating a summary statistic of turnout (using yes from before) with the sum (for raw count rather than average)for income level per province to plot.

```
# DV3:  Hisogram Counts turnout by income facet province 172
table(ces2015$income_full) #see range of factors
table(ces2015$province) #see codes used for shorthand of provinces
ces2015 <- ces2015 %>%
  mutate(province = as.factor(province)) %>% #turn into factor
  mutate(province = recode_factor(province,#change based on codebook
                                  bc = "British Columbia",
                                  nb = "New Brunswick",
                                  ns = "Nova Scotia",
                                  nwt = "Northwest Territories",
                                  Nfld = "Newfoundland and \n Labrador",
                                  pei = "Prince Edward Island",
                                  Sask = "Saskatchewan")) %>%
  mutate(income_full = as.factor(income_full)) %>% #turn income into
    factor
  filter(!income_full %in% c(".r", ".d")) %>% droplevels() #remove the
    missing categories

table(ces2015$income_full)
ces2015$income_full <- relevel(ces2015$income_full,
                               ref = c("less than $29,999")) #specify
    levels of factor lowest
                  #to highest


ces2015<- ces2015 %>% #reorder for facet placement by size of province
  mutate(province = fct_reorder(province, province, .fun = length, .desc
    = TRUE))


turnout_income <- ces2015 %>%  #create turnout by income variable
  group_by(province, income_full) %>% #grouped by province
  filter(!is.na(income_full)) %>% #remove na
  summarise(turnout = sum(p_voted == "Yes", na.rm = TRUE))

class(turnout_income$province)
```

When plotting chose 5 columns as 10 categories for symmetry and ease of comparison, wrapping by province. Removing the messy elements like x-axis labels and relying on the legend for category interpretation was decided as the levels are already ordered. As

```r
plot3 <- ggplot(turnout_income, aes(x = income_full, y = turnout, fill = income_full)) +
  geom_col(alpha = 0.75)+
  facet_wrap(vars(province), ncol = 5) + #facet by proving
  scale_y_continuous(breaks = seq(0, 400, by = 50), expand = c(0,5)) + #specify axis amounts
  theme_classic(base_family = "CMU Sans Serif")+
  labs(y = "Turnout Count", fill = "Income Level",
       title = "Canadian Province Turnout Counts by Income Level",
       caption = "Canadian Electoral Survey 2015. Total number of respondants who said 'Yes' when asked if they voted") +
  scale_fill_eb() + #use colours from earlier
  theme(
    legend.position = "bottom", #move to bottom to have more horizontal space
    legend.text = element_text(size = 8),
    axis.text.x = element_blank(),
    axis.title.x = element_blank(),
    axis.line.x = element_line(colour = "black"),
    axis.ticks.x = element_blank(),
    strip.text = element_text(size = 6.5, colour = "white", face = "bold"), #facet title text white
    strip.background = element_rect(fill = "navy", colour = "navy"), #blue box white text
    panel.border = element_rect(color = "navy", fill = NA, linewidth = 0.4), #add distinguishing box around each plot
    panel.grid.major.y = element_line(colour = "#e6e9ff", linewidth = 0.2), #very light grid
    panel.grid.major.x = element_line(colour = "#e6e9ff", linewidth = 0.2),
    legend.background = element_rect(fill = "white",colour = "grey30", size = 0.5, linetype = "solid")
  ) +
  guides(fill = guide_legend(nrow = 2)) #2 rows for legend to condense long fill names
ggsave("PS03_p3.pdf", plot3, device = cairo_pdf)
```


Canadian Province Turnout Counts by Income Level
Ontario
Quebec
British Columbia
Alberta
Manitoba
Saskatchewan
Nova Scotia
New Brunswick
Newfoundland and Labrador
Prince Edward Island
Turnout Count
0
50
100
150
200
250
300
350
Income Level
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
Canadian Electoral Survey 2015. Total number of respondants who said 'Yes' when asked if they voted

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.

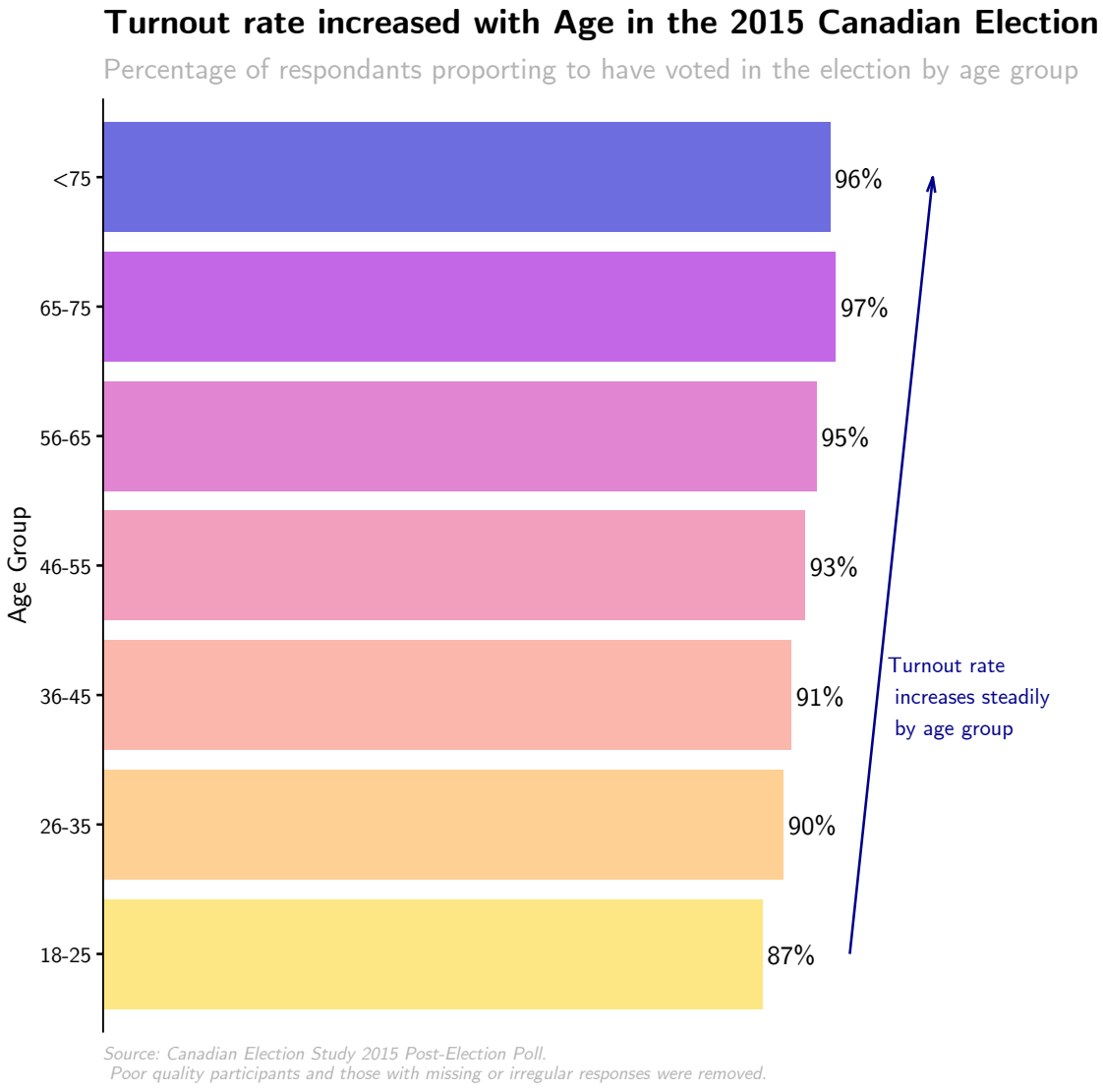
   Create work in progress theme.

```
# DV 4: Create customisable theme
theme_ellen <- function(...) {
  theme_classic(base_size = 12, base_family = "CMU Sans Serif") +
    theme(
      # Titles
      plot.title = element_text(face = "bold", size = rel(1.3)),
      plot.subtitle = element_text(face = "plain", size = rel(1.1), color
    = "grey70"),
      plot.caption = element_text(face = "italic", size = rel(0.7),
                                  color = "grey70", hjust = 0),
      #Backgrounds
      # Grid
      panel.background = element_rect(fill = "white", colour = NA), #
    clean base to work from
      panel.border = element_blank(),
      panel.grid = element_blank(),
      panel.grid.minor = element_blank(),

      #Axis
      axis.title = element_text(family = "CMU Sans Serif", size = 12), #
    title font
      axis.line.x = element_line(linewidth = 0.4), #axis lines tin
      axis.line.y = element_line(linewidth = 0.4),
      axis.text = element_text(colour = "black", size = 10), #axis text
    change
      axis.ticks.x = element_line(colour = "black", linewidth = 0.75),

      #Facets
      strip.background = element_blank(),
      strip.text = element_text(size = 11),

      #Legend
      legend.title = element_text(family = 'CMU Sans Serif', size = 12,
    colour = 'black'), #legend
      legend.text =  element_text(family = 'CMU Sans Serif', size = 10,
    colour = 'black'),# font change
      legend.key.height = unit(1, "lines"),#spacing
      legend.key.width = unit(1, "lines"),
      legend.spacing.y = unit(1, "lines"),
      legend.background = element_rect(
```

```
        fill = NA,
        colour = "grey60", #make box around legend to align it
        size = 0.5,
        linetype = "solid")
    )
}
```

Now add it to first graph and try to make a better visualisation. Flip the coordinates, added more informative titles and captions, selected a more cohesive and appealing palette (slightly gradiating colours is okay as categories are ordered). Rather than having an axis with percentages add the actual average turnout percentage to the right of the horizontal bar to make it clearer. Annotate with arrow and text additionally to explain trend.

```
#Apply to previos
plot4 <- plot1 + theme_ellen() + #ironically done most of this in first plt
  coord_flip() +
  labs(title = "Turnout rate increased with Age in the 2015 Canadian Election",
       subtitle = "Percentage of respondants proporting to have voted in the election by age group",
       caption = "Source: Canadian Election Study 2015 Post-Election Poll. \n Poor quality participants and those with missing or irregular responses were removed.") +
  theme(
    axis.text.x = element_blank(), #remove excess cluter
    axis.line.x = element_blank(),
    axis.title.x = element_blank(),
    axis.ticks.x = element_blank()
  ) +
  scale_fill_manual(values = c("#fdd735", "#ffb14e", "#fa8775", "#ea5f94", "#cd34b5", "#9d02d7", "#0d0dca")) +
  scale_y_continuous(limits = c(0,1.3), expand = c(0,0)) + #make space for text annotation
  geom_text(
    aes(label = scales::percent(turnout, accuracy = 1)), #turnout proportion as a percentage
    hjust = -0.1) + #slightly to the right of bar
  annotate(geom = "segment", x = 1, xend = 7, y = .99, yend = 1.1, colour = "darkblue",
           arrow = arrow(angle = 15, length = unit(0.5, "lines"))) + #add arrow angling trend
  annotate(geom = "text", x = 3, y = 1.04, label = "Turnout rate \n increases steadily \n by age group", hjust = 0, colour = "darkblue", size = 3.5)
  # add text describing key pattern
ggsave("PS03_p4.pdf", plot4, device = cairo_pdf)
```


Turnout rate increased with Age in the 2015 Canadian Election
Percentage of respondants proporting to have voted in the election by age group
Age Group
<75
65-75
56-65
46-55
36-45
26-35
18-25
96%
97%
95%
93%
91%
90%
87%
Turnout rate
increases steadily
by age group
Source: Canadian Election Study 2015 Post-Election Poll.
Poor quality participants and those with missing or irregular responses were removed.