```
# Packages ------------------------------------------------------------

library(sf)
library(tidyverse)
library(readxl)
library(writexl)

# Load Data -----------------------------------------------------------

# 1. stations

stations <-
   read_csv('DataStory/alt_fuel_stations.csv')

# 2. EV stocks

# stocks

EV_registration_by_state <-
   read_excel('DataStory/ev-registration-counts-by-state.xlsx') %>%
   select(State, `Registration Count`) %>%
   filter(State != 'Total')

# registration

LDV_registration_2021 <-
   read_excel('DataStory/2021_LDV_registration.xlsx') %>%
   filter(State != 'United States')

# 3. Housing and other demography data

DMV_housing_temp <-
   read_xlsx('DataStory/DMV_housing.xlsx')

DMV_housing <-
   t(DMV_housing_temp[-1])

colnames(DMV_housing) <- select(DMV_housing_temp, `vehicles available`)

# basic states information

data(state)
```

```
state <-
   cbind(state.name, state.abb, state.area) %>%
   as_tibble() %>%
   rename(State = state.name,
          abb = state.abb,
          area = state.area)

# 4. laws and incentives

EV_law <-
   read_csv('DataStory/EV_law.csv') %>%
   filter(Recent == TRUE,
          State != 'US') %>%
   group_by(State) %>%
   summarise(EV_law = n()) %>%
   rename(state.abb = State)

# PHEV_law is useless since EV_law overlaps with it

PHEV_law <-
   read_csv('DataStory/PHEV_law.csv') %>%
   filter(`Recent?` == TRUE,
          State != 'US') %>%
   group_by(State) %>%
   summarise(PHEV_law = n()) %>%
   rename(state.abb = State)

# states that have incentives or not

state_w_incentive <-
   cbind(state.abb, state.name) %>%
   as_tibble() %>%
   full_join(EV_law) %>%
   full_join(PHEV_law) %>%
   rename(State = state.name,
          abb = state.abb) %>%
   mutate(EV_law_B = if_else(!is.na(EV_law),
                             T,
                             F)) %>%
   mutate(PHEV_law_B = if_else(!is.na(PHEV_law),
                               T,
                               F))

# Data preparation ----------------------------------------------------
```

```
# 1. charger number by state

charger_by_state <-
   stations %>%

   # the open date is ealier than 2022, so this is the existence of chargers in
   # 2021

   filter(year(`Open Date`) < 2022) %>%

   # 'ON' is in Canada

   filter(State != 'ON') %>%
   select(`Station Name`, `EV Level1 EVSE Num`, `EV Level2 EVSE Num`,
            `EV DC Fast Count`, State) %>%

   # 'NA' indicates that there is no this type of EVSE in the station, so there
   # is 0 this type of EVSE in the station

   replace(is.na(.), 0) %>%

   # the total number of charging points in a charging station is the sum of
   # 3 types of EVSEs

   mutate(EVSE = `EV Level1 EVSE Num` + `EV Level2 EVSE Num` +
            `EV DC Fast Count`) %>%

   # calculate the total number of EVSE in each state in 2021

   group_by(State) %>%
   summarise(EVSE_state = sum(EVSE))

# 2. increase of EV

# EV count

LDV_2021 <-
   LDV_registration_2021 %>%
   transmute(
      State = State,
      EV = `Electric (EV)`,
      PHEV = `Plug-In Hybrid Electric (PHEV)`) %>%
   mutate(count = EV + PHEV)
```

```
# percentage increase

EV_percentage <-
   LDV_2021 %>%
   full_join(EV_registration_by_state,
                  by = join_by(State)) %>%
   mutate(percent = `Registration Count` / count)

# Increase & charger count ----------------------------------------------

# An analysis based on 2021 data

increase_and_charger <-
   charger_by_state %>%
   rename(abb = State) %>%
   full_join(state,
                  by = join_by(abb)) %>%
   full_join(EV_percentage,
                  by = join_by(State)) %>%

   # combine the DC values into one row, then discard the previous rows
   # containing NA values

   rbind(c('DC', 798, 'District of Columbia', 68, 3700, 2500, 6200, 3700,
               0.596774193548387)) %>%
   na.omit()

# calculate the charger density in each state

increase_and_charger$EVSE_state <- as.numeric(increase_and_charger$EVSE_state)
increase_and_charger$area <- as.numeric(increase_and_charger$area)
increase_and_charger$percent <- as.numeric(increase_and_charger$percent)
increase_and_charger$count <- as.numeric(increase_and_charger$count)

# scatter plot with a fit line

increase_and_charger <-
   increase_and_charger %>%
   mutate(charger_density = EVSE_state / area)

# 1. charger density by area

p1 <-
```

```
    increase_and_charger %>%

    # exclude the outlier for a better visualization

    filter(abb != 'DC') %>%
    ggplot(aes(x = charger_density,
                       y = percent)) +
    geom_point(color = 'gray',
                       size = 4) +
    stat_smooth(method = 'lm',
                         formula = y ~ x,
                         se = F,
                         color = '#4dac26') +
    theme_bw() +
    theme(axis.line = element_line(colour = '#969696'),
              axis.ticks = element_blank(),
              panel.grid.minor = element_blank(),
              panel.grid.major = element_line(linetype = 'dashed'),
              panel.border = element_blank())

# 2. charger density by EV count

increase_and_charger <-
    increase_and_charger %>%
    mutate(charger_density = EVSE_state / count)

p2 <-
    increase_and_charger %>%

    # exclude the outlier for a better visualization

    filter(abb != 'DC') %>%
    ggplot(aes(x = charger_density,
                       y = percent)) +
    geom_point(color = 'gray',
                       size = 4) +
    stat_smooth(method = 'lm',
                         formula = y ~ x,
                         se = F,
                         color = '#4dac26') +
    theme_bw() +
    theme(axis.line = element_line(colour = '#969696'),
              axis.ticks = element_blank(),
              panel.grid.minor = element_blank(),
```

```
            panel.grid.major = element_line(linetype = 'dashed'),
            panel.border = element_blank())

# Scatter plot with 2 fit lines ----------------------------------------

# consider incentives for EVs and PHEVs

plot2_data <-
   increase_and_charger %>%
   left_join(state_w_incentive)

p3 <-
   plot2_data %>%
   filter(abb != 'DC') %>%
   ggplot() +
   geom_point(aes(x = charger_density,
                  y = percent,
                  color = EV_law_B),
              size = 4,
              alpha = 0.8) +
   scale_color_manual(values = c('#fc8d59', '#91cf60')) +
   geom_smooth(aes(x = charger_density,
                   y = percent,
                   fill = EV_law_B),
               method = 'lm',
               formula = y ~ x,
               se = F,
               color = '#969696') +
   theme_bw() +
   theme(axis.line = element_line(colour = '#969696'),
            axis.ticks = element_blank(),
            panel.grid.minor = element_blank(),
            panel.grid.major = element_line(linetype = 'dashed'),
            panel.border = element_blank())

# p4

p4 <-
   plot2_data %>%
   filter(abb != 'DC') %>%
   ggplot() +
   geom_point(aes(x = charger_density,
                  y = percent,
                  color = EV_law_B),
```

```
                    size = 4,
                    alpha = 0.8) +
  scale_color_manual(values = c('#fc8d59', '#91cf60')) +
  geom_smooth(aes(x = charger_density,
                  y = percent,
                  fill = EV_law_B),
              method = 'lm',
              formula = y ~ x,
              se = F,
              color = '#969696') +
  theme_bw() +
  theme(axis.line = element_line(colour = '#969696'),
        axis.ticks = element_blank(),
        panel.grid.minor = element_blank(),
        panel.grid.major = element_line(linetype = 'dashed'),
        panel.border = element_blank())

# Map ------------------------------------------------------------------

write_xlsx(plot2_data, 'DataStory/EV_map.xlsx')
```