Temperature `temp_7` S3 Data Class

Miranda Johnson

Temperature Data

Loading temperature data for November 21, 2021 to November 26, 2022 to a data frame can be done as follows:

```
library(tidyverse)
library(ggplot2)
library(patchwork)

temperature <- read.csv(url("https://raw.githubusercontent.com/aftermiranda/temp_7/main/average-tempera
```

This data, in it's current form, represents the daily average temperatures for each of 7 locations over the course of 53 weeks. The goal of the `temp_7` data type is to reduce the data to weekly averages, and provide summary information for local and country wide values.

Average Weekly Temperature - `temp_7` class

To do start, create a new function called `avg_week_temp` to calculate the temperature over each 7 day period.

```
avg_week_temp <- function(obj) {
  # function to calculate the average weekly temperature for a vector of data
  # takes in a vector with minimum length of 7, expected length to be a multiple
  # of 7, but will not raise errors for other lengths
  # returns a vector 1/7 the length of the original vector
  i <- 7
  loop_count <- 1
  avt <- vector()
  while(i < length(obj) + 1) {
    # define the beginning of the week, and calculate the week average
    j = i - 6
    next_avg <- mean(obj[j:i])

    # add the average value to the vector
    avt[[loop_count]] <- next_avg

    # increment looping variables
    i <- i + 7
    loop_count <- loop_count + 1
  }
  # return the vector of weekly averages
  return(avt)
}
```

Each column from the data frame can be passed in to the `avg_week_temp` function, but this requires us to evaluate this function 7 times to get the output for all of the columns. To clean up the output, we will create a second function called `local_avg` to call the `avg_week_temp` function in a loop.

```
local_avg <- function(obj){
  # function to calculate the Canada wide average weekly temperature for 1 year
  # takes in a data frame with 371 observations (53 weeks worth of daily averages)
  # for 7 cities along the Trans Canada Railway
  # returns a vector with 53 weekly averages
  obj_col_names <- colnames(obj)
  week_averages <- data.frame(
    matrix(vector(), 53, 10,
           dimnames=list(c(), c(obj_col_names))))
  # Get dates for the end of week
  for (i in 1:3){
    a <- obj[[i]][seq(7, length(obj[[i]]), 7)]
    week_averages[[i]] <- a
  }
  # Get avg_week_temp for each location
  for (i in 4:10){
    # the first 3 cols in obj are dates, so we start at column 4
    week_averages[[i]] <- avg_week_temp(obj[[i]])
  }
  week_averages$date <- as.Date(with(week_averages, paste(Year, Month, Day, sep = "-")),
                                "%Y-%m-%d")
  ave <- select(week_averages, date, Vancouver:Charlotte)
  class(ave) <- 'temp_7'
  return(ave)
}

local_weekly_avg <- local_avg(temperature)
class(local_weekly_avg)
```

```
## [1] "temp_7"
```

Due to the fact that I want the *temp_7* class data to be formatted as a data frame with 53 observations of 10 variables, I have the *local_avg* function return the item of that class.

To Data Frame

Since the data needs to be turned to a dataframe to manipulate it, we will make a function to return it to a dataframe object. This will be called from other functions that manipulate the `temp_7` data.

```
to_df <- function(obj){
  # change temp_7 data to a dataframe

  n <- names(obj[1:8])
  f <- data.frame(
    matrix(vector(), 53, 8, dimnames=list(c(), c(n))))
  f[[1]] <- obj$date
  f[[2]] <- obj$Vancouver
  f[[3]] <- obj$Calgary
  f[[4]] <- obj$Saskatoon
  f[[5]] <- obj$Winnipeg
  f[[6]] <- obj$Ottawa
  f[[7]] <- obj$Quebec.City
```

```
  f[[8]] <- obj$Charlotte

  return(f)
}
```

Summary for `temp_7`

The intended use of this data is to have a weekly country wide average to compare to weekly country wide rail data. The summary method should therefore return the row average for each week. I have elected to add the overall minimum and maximum average for each of the weeks as well.

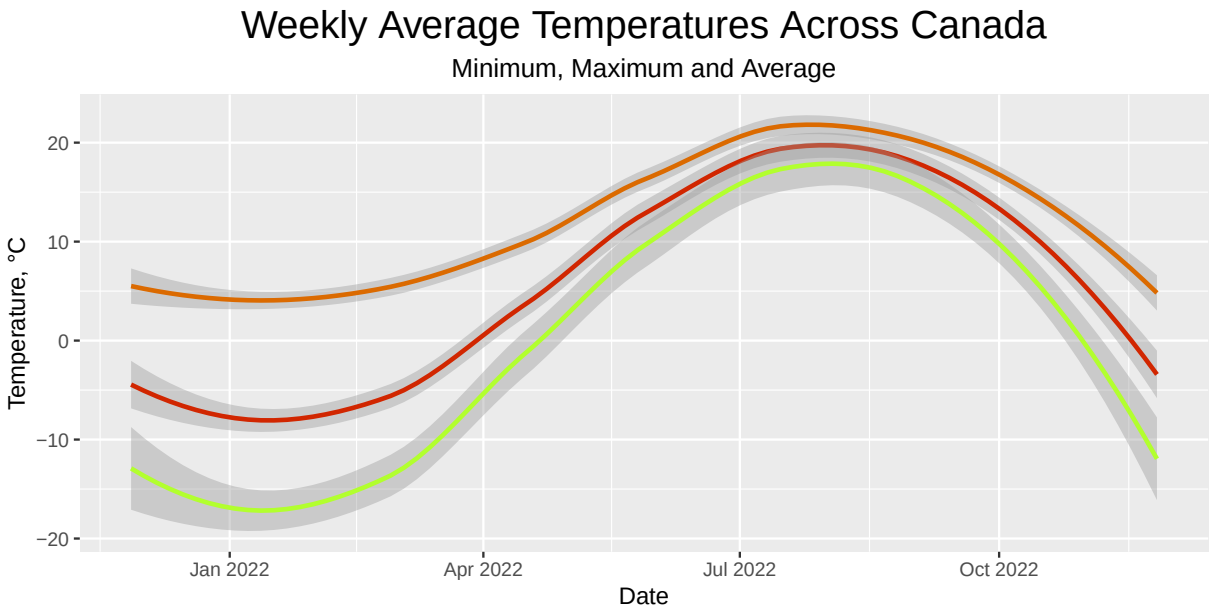
```
summary.temp_7 <- function(obj){
  ## Accepts a temp_7 class object containing 53 observations of 8 variables
  ## Calculates a Canada wide average, max and min

  s <- to_df(obj)

  s$average <- round(rowMeans(s[2:8]), digits=2)
  s$max <- round(apply(X=s[2:8], MARGIN = 1, FUN = max), digits=2)
  s$min <- round(apply(X=s[2:8], MARGIN = 1, FUN = min), digits=2)

  new_data <- select(s, date, min, average, max)
  return(new_data)
}

canada_sum <- summary(local_weekly_avg)
str(canada_sum)
```

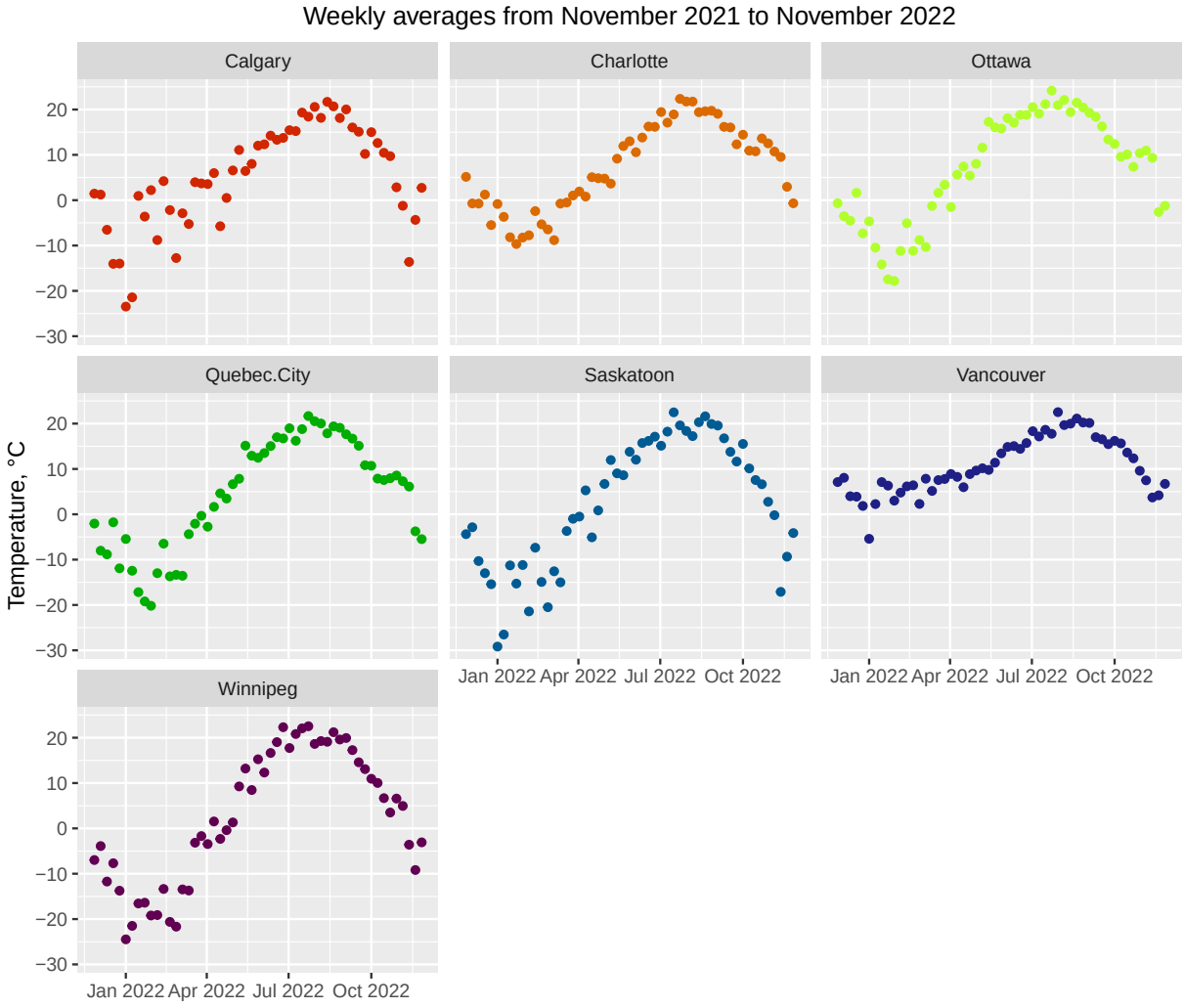
```
## 'data.frame':    53 obs. of  4 variables:
##  $ date   : Date, format: "2021-11-27" "2021-12-04" ...
##  $ min    : num  -6.98 -8.02 -11.72 -14.04 -15.41 ...
##  $ average: num  -0.05 -1.39 -5.53 -4.24 -9.43 ...
##  $ max    : num  7.11 8.05 3.98 3.89 1.84 -0.84 2.26 7.11 6.32 3.01 ...
```

Print for `temp_7`

The print function for the `temp_7` class is going to provide images rather than numbers. In order to show the differences in temperature by location, the data for each location is given its own plot. I have elected to also show the overall average, maximum and minimum using a local regression to fit a curve to the data (loess regression).

```
print.temp_7 <- function(obj){
  suppressMessages(library(tidyverse))
  suppressMessages(library(ggplot2))
  suppressMessages(library(patchwork))

  # Override print function to allow desired output for temp_7 class data

  rainbow <- c("#D12600", "#DB6A00", "#B2FF2E",
               "#00AD00", "#005B94", "#1E2085", "#610052")

  # update ggplot and patchwork theme to center title, subtitle, and caption text
  theme_update(plot.title = element_text(hjust = 0.5, size = 18),
               plot.subtitle = element_text(hjust = 0.5, size = 12),
               plot.caption = element_text(hjust = 0.5))

  s <- to_df(obj)
  s$Average <- rowMeans(s[2:8])
  s$Min <- apply(X=s[2:8], MARGIN = 1, FUN = min)
  s$Max <- apply(X=s[2:8], MARGIN = 1, FUN = max)

  longer_data <- s %>%
    pivot_longer(Vancouver:Charlotte, names_to = "Location", values_to = "Temperature")
  sum_data <- s %>%
    pivot_longer(Average:Max, names_to = "Stat", values_to = "Temperature")

  p1 <- ggplot(longer_data,  aes(x=date, y=Temperature, color=Location)) +
    geom_point() + scale_color_manual(values = rainbow) +
    guides(colour="none") + facet_wrap(vars(Location)) +
    labs(x="", y="Temperature, °C",
         subtitle = "Weekly averages from November 2021 to November 2022")

  p2 <- ggplot(sum_data, aes(x=date, y=Temperature, color=Stat)) +
    geom_smooth(method = 'loess', formula = 'y~x') + guides(colour="none") +
    labs(title="Weekly Average Temperatures Across Canada",
         subtitle="Minimum, Maximum and Average", x="Date", y="Temperature, °C") +
    scale_color_manual(values = rainbow)

  layout <- c(area(1,1,1,2), area(2,1,3,2))

  p2 + p1 + plot_layout(design = layout)
}

print(local_weekly_avg)
```

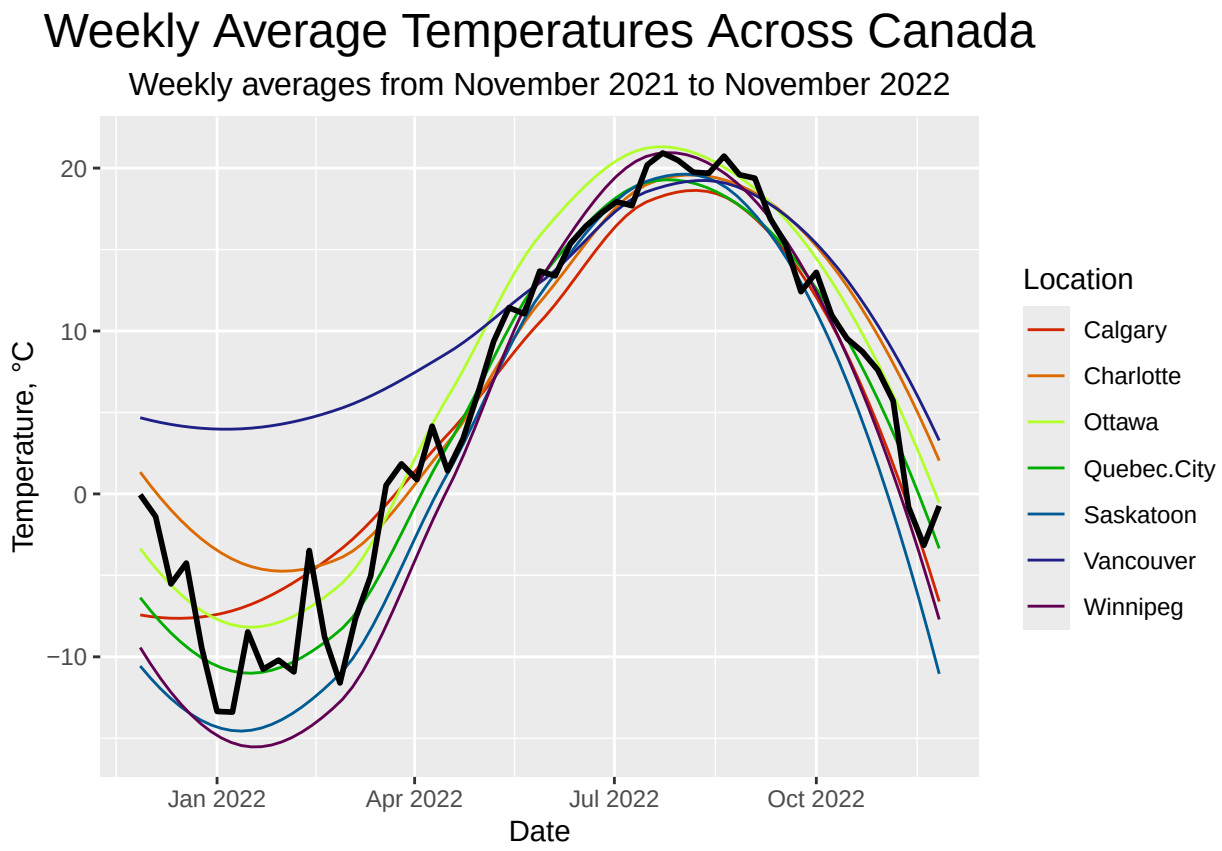

Weekly Average Temperatures Across Canada
Minimum, Maximum and Average
Temperature, °C
20
10
0
−10
−20
Jan 2022
Apr 2022
Jul 2022
Oct 2022
Date


Weekly averages from November 2021 to November 2022
Calgary
Charlotte
Ottawa
Quebec.City
Saskatoon
Vancouver
Winnipeg
Temperature, °C
20
10
0
−10
−20
−30
Jan 2022
Apr 2022
Jul 2022
Oct 2022

Plot for `temp_7`

To plot the data, I have elected to include data for all locations on a single scatter plot. I have added a line to map the country wide average. While the use of the 'jcolors' library is not strictly necessary, I found the default color mapping did not provide enough contrast with 7 data sets on one image.

```
plot.temp_7 <- function(obj){
  # Accepts a temp_7 class item containing 53 observations of 8 variables
  # Creates a plot showing the average temperature for each location each week

  suppressMessages(library(tidyverse))
  suppressMessages(library(ggplot2))
  suppressMessages(library(patchwork))

  rainbow <- c("#D12600", "#DB6A00", "#B2FF2E",
               "#00AD00", "#005B94", "#1E2085", "#610052")

  # update ggplot and patchwork theme to center title, subtitle, and caption text
  theme_update(plot.title = element_text(hjust = 0.5, size = 18),
               plot.subtitle = element_text(hjust = 0.5, size = 12),
               plot.caption = element_text(hjust = 0.5))

  plot_1 <- to_df(obj)

  plot_1$overall <- rowMeans(plot_1[,2:8])

  longer_data <- plot_1 %>%
    pivot_longer(Vancouver:Charlotte, names_to = "Location", values_to = "Temperature")

  p1 <- ggplot(longer_data,  aes(x=date, y=Temperature, color=factor(Location))) +
    geom_smooth(method = "loess", se = FALSE, linewidth = 0.5) +
    guides(color = guide_legend("Location"), line = guide_legend("Overall")) +
    labs(title="Weekly Average Temperatures Across Canada", x="Date", y="Temperature, °C",
         subtitle = "Weekly averages from November 2021 to November 2022") +
    geom_line(mapping = aes(x=date, y=overall), color="black", linewidth = 1)


  p1 + scale_color_manual(values = rainbow)

}

plot(local_weekly_avg)
```


Weekly Average Temperatures Across Canada
Weekly averages from November 2021 to November 2022
Temperature, °C
20
10
0
−10
Jan 2022
Apr 2022
Jul 2022
Oct 2022
Date
Location
Calgary
Charlotte
Ottawa
Quebec.City
Saskatoon
Vancouver
Winnipeg