Weekly RT-LBCI to Monthly

Vimaljeet Singh

2023-05-16

Load the file with raw RT-LBCI values for all cities

```
# load the file with raw RT-LBCI values
index <- read.csv("RT-LBCI.csv")

# Converting to date type
index$Date <- as.Date(index$Date)

# Extracting year and and month
index$Year <- year(index$Date)
index$Month <- month(index$Date)

head(index)
```

```
##         Date Montreal Ottawa Toronto Calgary Vancouver Year Month
## 1 2020-08-10   100.00 100.00  100.00  100.00    100.00 2020     8
## 2 2020-08-17    89.20  97.79  104.41  102.18    103.20 2020     8
## 3 2020-08-24    87.22 102.30  106.39  114.89    101.99 2020     8
## 4 2020-08-31    98.07  99.30  109.65  109.88    101.32 2020     8
## 5 2020-09-07   102.95 107.32  112.93  119.03    109.63 2020     9
## 6 2020-09-14    97.83 103.15  109.35  120.31    108.88 2020     9
```

Required functions

- Function 1 works on the output of the first function and aggregates the lagged values by calculating the mean value by grouping them by year then month in 'cities' and rounds the mean values to 2 decimal places.
- Function 2 creates a date column by clubbing the month and year columns and making the date to be the first day of that month and drops the year and month column in the input df (which is the output from function 1)

```
# Function 1
aggregate_lagged_values <- function(data) {
  cities = c("Montreal", "Ottawa", "Toronto", "Calgary", "Vancouver")
  result <- data %>%
    group_by(Year, Month) %>%
    summarize(across(all_of(cities), ~ round(mean(., na.rm = TRUE), 2)))
```

```
  result <- as.data.frame(result)  # Convert to data frame

  return(result)
}

# Function 2
add_date <- function(data) {
  # Convert Year and Month columns to numeric if they are stored as characters
  data$Year <- as.numeric(data$Year)
  data$Month <- as.numeric(data$Month)

  # Create a new Date column using the Year and Month values
  data$Date <- as.Date(paste(data$Year, data$Month, "01", sep = "-"))

  # Move the Date column to the first position
  data <- data[, c("Date", names(data)[-which(names(data) %in% c("Year", "Month"))])]

  # Remove the duplicate Date column
  data <- data[, -which(names(data) == "Date.1")]

  return(data)
}
```

Aggregate the values to monthly from weekly

```
aggregated_by_month <- add_date(aggregate_lagged_values(index))
```

```
## `summarise()` has grouped output by 'Year'. You can override using the
## `.groups` argument.
```

```
head(aggregated_by_month)
```

```
##         Date Montreal Ottawa Toronto Calgary Vancouver
## 1 2020-08-01    93.62  99.85  105.11  106.74    101.63
## 2 2020-09-01   107.23 103.39  111.25  117.86    108.22
## 3 2020-10-01   120.58 115.71  126.67  136.03    131.34
## 4 2020-11-01   143.80 132.55  129.75  147.33    139.94
## 5 2020-12-01   155.14 143.09  134.18  139.74    156.87
## 6 2021-01-01   163.44 119.11  139.36  143.25    173.70
```

Save as CSV

```
# uncomment the code to run
# Change the file path if you want to save the dataframes as csv in your machine
# write.csv(aggregated_by_month, "RT-LBCI_Monthly.csv", row.names = FALSE)
```