From pandemic to endemic: summer 2022 divergence of COVID-19 case numbers and SARS-CoV-2 RNA detection in wastewaters of Rochester, Minnesota

Dr Biruhalem Taye

04/07/2023

#Supplementary data

```
#loading R libraries
library(dplyr)
library(tidyverse)
library(reticulate)
library(rstatix)
library(ggpubr)
library(ggplot2)
```

loading and formatting data

```
#loading data

alldata <- read.csv("C:/Users/biruh/Desktop/Wastwater/Figuresversion2/2023Alldatarefined.csv")

#head(alldata)

#Converting character data to date to be recognized by R

alldata$Days <- as.Date(alldata$Days,"%m/%d/%Y")

alldata$Weekday <- format(alldata$Days, '%A') # annotating weekdays

#head(alldata) #inspecting the data
```

Normality test

Normality test (Testing for normal/Gaussian distribution of the data)

Shapiro-Wilk normality test:

If the p-values > 0.05 the data is normally distributed. $P < 0.05$ the data doesn't assume normal distribution.

```
library(rstatix)

shapiro_test(alldata, Rochester_N1, Rochester_N2, Rochester_SARS, Rochester_Cases, Stewartville_SARS, S
```

```
## # A tibble: 8 x 3
##   variable           statistic        p
##   <chr>                  <dbl>    <dbl>
## 1 Byron_Cases            0.608 1.14e-36
## 2 Byron_SARS             0.628 1.15e-12
## 3 Rochester_Cases        0.505 6.75e-40
## 4 Rochester_N1           0.706 5.76e-24
## 5 Rochester_N2           0.704 5.10e-24
## 6 Rochester_SARS         0.704 4.87e-24
## 7 Stewartville_Cases     0.598 5.09e-37
## 8 Stewartville_SARS      0.730 7.91e-11
```

Based on Shapiro-Wilk normality test all the data are not normally distributed, hence Spearman correlation is used for correlation analysis.

Modeling

Loading libraries (R-packages)

```
#Loading libraries

library(dLagM)
library(tictoc)
library(lmtest)
library(tseries)
library(forecast)
library(pracma)
library(dlnm)
```

#loading interpolated Rochester data

```
roch <- read.csv("C:/Users/biruh/Desktop/Wastwater/2023rochesterdata2.csv")

#Converting character data to date to be recognized by R

roch$Days <- as.Date(roch$Days,"%m/%d/%Y")

#Creating column that corresponds the days from Monday to Sunday.
roch$nameoftheday <- format(roch$Days, '%A')

wdata <- roch %>% select(1,9,10, 11) # selection of column 1, 9, 10 and 11 (relevant data)
```

Extracting data before 2022-04-25

```
#pcr = extracted data before 2022-04-25 (440 days)

pcr <- wdata %>% filter(Days <= '2022-04-25')
#head(pcr)
```

Modeling

For all modeling analysis we used the data before changes in guideline in COVID-19 case report (data before 2022-04-25 (440 days))

Distributed lag models and autoregressive distributed lag (ARDL)

Distributed lag model was selected based on previous publication.

https://pubmed.ncbi.nlm.nih.gov/36380770/

An R package "dLagM" and its dependencies were used for analysis

(https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0228812)

A distributed lag model can be an appropriate choice when there is a lagged relationship between the independent variable and the dependent variable.

independent variable = N1_N2gflowadj,

dependent variable = Daily_cases

MODEL 1

Finite distributed lag model

```
dlmFit1 <- dlm(x = pcr$N1_N2gflowadj, y = pcr$Daily_cases, q = 15)
summary(dlmFit1)
```

```
##
## Call:
## lm(formula = model.formula, data = design)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -221.817   -8.758    4.120   10.752  240.894
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -7.082541   2.699319  -2.624 0.009021 **
## x.t          0.622749   0.121249   5.136 4.35e-07 ***
## x.1          0.125655   0.160495   0.783 0.434127
## x.2          0.027478   0.161666   0.170 0.865117
## x.3          0.306917   0.161792   1.897 0.058537 .
## x.4         -0.079872   0.161790  -0.494 0.621800
## x.5          0.018551   0.162396   0.114 0.909111
## x.6          0.300763   0.162424   1.852 0.064789 .
## x.7          0.541775   0.162243   3.339 0.000917 ***
## x.8          0.330828   0.162259   2.039 0.042105 *
## x.9          0.071310   0.162565   0.439 0.661141
## x.10        -0.040336   0.162545  -0.248 0.804142
## x.11        -0.027693   0.161951  -0.171 0.864313
## x.12        -0.157267   0.161983  -0.971 0.332180
## x.13        -0.006496   0.161867  -0.040 0.968010
## x.14         0.515533   0.160755   3.207 0.001447 **
## x.15        -0.247211   0.121424  -2.036 0.042402 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 43.37 on 408 degrees of freedom
## Multiple R-squared:  0.7981, Adjusted R-squared:  0.7902
## F-statistic: 100.8 on 16 and 408 DF,  p-value: < 2.2e-16
##
## AIC and BIC values for the model:
##        AIC      BIC
```

```
## 1 4428.997 4501.934
```

MODEL 2

Autoregressive Distributed Lag (ARDL) model (with p = 8)

```
ARDLfit2 <- ardlDlm(x = pcr$N1_N2gflowadj, y = pcr$Daily_cases, p=8, q=15)
summary(ARDLfit2)
```

```
##
## Time series regression with "ts" data:
## Start = 16, End = 440
##
## Call:
## dynlm(formula = as.formula(model.text), data = data, start = 1)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -129.517   -6.410    0.397    6.857  114.869
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.446875   1.389351  -0.322 0.747891
## X.t          0.424863   0.065210   6.515 2.18e-10 ***
## X.1         -0.260722   0.087889  -2.967 0.003193 **
## X.2          0.019430   0.088740   0.219 0.826795
## X.3          0.257401   0.089825   2.866 0.004382 **
## X.4         -0.144860   0.089852  -1.612 0.107708
## X.5          0.077966   0.089838   0.868 0.386001
## X.6          0.313126   0.089841   3.485 0.000546 ***
## X.7         -0.039107   0.089763  -0.436 0.663313
## X.8         -0.051374   0.072299  -0.711 0.477760
## Y.1          0.776190   0.050428  15.392  < 2e-16 ***
## Y.2         -0.009379   0.064181  -0.146 0.883887
## Y.3         -0.125554   0.062971  -1.994 0.046849 *
## Y.4          0.097438   0.063222   1.541 0.124058
## Y.5         -0.197490   0.062877  -3.141 0.001809 **
## Y.6          0.158395   0.063428   2.497 0.012918 *
## Y.7          0.573427   0.064058   8.952  < 2e-16 ***
## Y.8         -0.441209   0.065737  -6.712 6.60e-11 ***
## Y.9         -0.039062   0.061375  -0.636 0.524846
## Y.10        -0.035003   0.057907  -0.604 0.545882
## Y.11         0.042609   0.057937   0.735 0.462512
## Y.12         0.035334   0.058468   0.604 0.545970
## Y.13        -0.151335   0.058410  -2.591 0.009922 **
## Y.14         0.118740   0.059358   2.000 0.046133 *
## Y.15        -0.085872   0.045550  -1.885 0.060125 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 22.11 on 400 degrees of freedom
## Multiple R-squared:  0.9486, Adjusted R-squared:  0.9455
## F-statistic: 307.3 on 24 and 400 DF,  p-value: < 2.2e-16
```

GOODNESS-OF-FIT MEASURES

```
sortScore(x = MASE(dlmFit1, ARDLfit2), score = c("mase"))
```

```
##            n      MASE
## ARDLfit2 425 0.6545076
## dlmFit1  425 1.2379335
```

```
bestfitted_model <- GoF(dlmFit1, ARDLfit2)
print(bestfitted_model)
```

```
##            n      MAE MPE MAPE     sMAPE      MASE       MSE       MRAE
## dlmFit1  425 24.38203 Inf  Inf 0.7550239 1.2379335 1805.4876 4654220424
## ARDLfit2 425 12.89102 NaN  Inf 0.4607592 0.6545076  460.1304 1010625076
##             GMRAE       MBRAE      UMBRAE
## dlmFit1  4.329524 -0.05132741 -0.04882153
## ARDLfit2 2.417013 -0.35745543 -0.26332756
```