EPIWEEK ARIMAX MODELS - without log-back transformations

Victor Felix

November 11, 2025

Here we run ensembles and single automatic ARIMAX models for forecasting weekly hospitalizations in 48 states on the contiguous U.S. These model use mean hospitalizations by epidemiological weeks over the last 2 years as exogenous variables. The function fits on a rolling window of previous 104 weeks for the state under analysis and a rolling windows with the same size with 1 week-lag for the exogenous variables to generate forecasts. These models do not include log-back transformations. It return some metrics that evaluate the performance of the models:target_end_date, abs_error, cases, forecast, 'N_of_models", weighted interval score (WIS), predictive quantiles. The user can choose a single best automatic ARIMAXs (auto=TRUE), or ensembles of 27 permutations of 0,1,2 pdq's (ES27=TRUE) or 64 permutations of 0,1,2,3 pdq's (ES64=TRUE). The user also chooses the number of weeks ahead for each forecast, and the size of the rolling window which is set as 2 years (104 weeks).

```
knitr::opts_chunk$set(echo = TRUE)
```

!!!!!!!!!!!!!!!!!!! LOADING THE PACKAGES !!!!!!!!!!!!!!!!!!!

```
library("tidyr")
library("MMWRweek")
library("data.table")
library("caret")
```

```
## Loading required package: ggplot2
```

```
## Loading required package: lattice
```

```
library("purrr")
```

```
##
## Attaching package: 'purrr'
```

```
## The following object is masked from 'package:caret':
##
##     lift
```

```
## The following object is masked from 'package:data.table':
##
##     transpose
```

```r
library("dplyr")
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:data.table':
##
##     between, first, last

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
library("tseries")
```

```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

```r
library("gtools")
library("forecast")
library("scoringutils")
```

```
## Note: scoringutils is currently undergoing major development changes (with an update planned for the
```

```r
library("covidHubUtils")
library("parallel")
library("future")#https://cran.r-project.org/web/packages/future/vignettes/future-4-issues.html
```

```
##
## Attaching package: 'future'

## The following object is masked from 'package:tseries':
##
##     value

## The following object is masked from 'package:caret':
##
##     cluster
```

```r
library("listenv")
```

```
##
## Attaching package: 'listenv'
```

```
## The following object is masked from 'package:purrr':
##
##     map
```

```
library("epitools")
```

!!!!!!!!!!!!!!!!!!!!!!!!!!!! LOADING DATASET AND FUNCTIONS !!!!!!!!!!!!!!!!!!!!!!!!!!!!

```
##########################################
#       LOADING AND CLEANING THE DATASET  #
##########################################

# Loads the ADJACENT states models
source("ES_EPIWEEK_nolog.R", local = TRUE, chdir = TRUE)

# Loads the ILI dataset
my_data = read.csv("treated_influenza_hosp_dataframe_v2.csv")
my_data$target_end_date<-as.Date(my_data$target_end_date) # set the dates as dates

list_of_states <- split(my_data, my_data$state_name)
```

AUTO EPIWEEK WEEK1

```
start_time <- Sys.time()

# RUN MODEL
AUTO_EPIWEEK_WEEK1_list <- mclapply(list_of_states, ES_EPIWEEK, auto=TRUE, n_weeks_ahead=1,week_lag=1, r
  setNames(names(list_of_states))

end_time <- Sys.time()
run_time <- end_time - start_time

print(run_time)
```

```
## Time difference of 12.79716 mins
```

```
# FINAL DATAFRAME
AUTO_EPIWEEK_WEEK1 <- bind_rows(AUTO_EPIWEEK_WEEK1_list, .id = "State")

# SAVE TO CORRECT FOLDER
#write.csv(EPIWEEK/AUTO/AUTO_EPIWEEK_WEEK1, file = "AUTO_EPIWEEK_WEEK1.csv", row.names = TRUE)
```

AUTO EPIWEEK WEEK2

```
start_time <- Sys.time()

# RUN MODEL
AUTO_EPIWEEK_WEEK2_list <- mclapply(list_of_states, ES_EPIWEEK, auto=TRUE, n_weeks_ahead=2,week_lag=1, r
  setNames(names(list_of_states))

end_time <- Sys.time()
run_time <- end_time - start_time

print(run_time)
```

```
## Time difference of 12.39938 mins
```

```
# FINAL DATAFRAME
AUTO_EPIWEEK_WEEK2 <- bind_rows(AUTO_EPIWEEK_WEEK2_list, .id = "State")
```

AUTO EPIWEEK WEEK3

```
start_time <- Sys.time()

AUTO_EPIWEEK_WEEK3_list <- mclapply(list_of_states, ES_EPIWEEK, auto=TRUE, n_weeks_ahead=3,week_lag=1, n
  setNames(names(list_of_states))

end_time <- Sys.time()
run_time <- end_time - start_time

print(run_time)
```

```
## Time difference of 12.05055 mins
```

```
# Combine the list of data frames into a single data frame with names as a column
AUTO_EPIWEEK_WEEK3 <- bind_rows(AUTO_EPIWEEK_WEEK3_list, .id = "State")
```

AUTO EPIWEEK WEEK4

```
start_time <- Sys.time()

AUTO_EPIWEEK_WEEK4_list <- mclapply(list_of_states, ES_EPIWEEK, auto=TRUE, n_weeks_ahead=4,week_lag=1 ,n
  setNames(names(list_of_states))

end_time <- Sys.time()
run_time <- end_time - start_time

print(run_time)
```

```
## Time difference of 12.04721 mins
```

```
# Combine the list of data frames into a single data frame with names as a column
AUTO_EPIWEEK_WEEK4 <- bind_rows(AUTO_EPIWEEK_WEEK4_list, .id = "State")
```

```
save.image("EPIWEEK_MODELS_influenza_hospitalization_nolog.Rdata")
```

ES27 EPIWEEK WEEK1

```
start_time <- Sys.time()

ES27_EPIWEEK_WEEK1_list <- mclapply(list_of_states, ES_EPIWEEK, ES27=TRUE, n_weeks_ahead=1,week_lag=1, n
  setNames(names(list_of_states))

end_time <- Sys.time()
run_time <- end_time - start_time

print(run_time)
```

```
## Time difference of 17.99846 mins
```

```
# Combine the list of data frames into a single data frame with names as a column
ES27_EPIWEEK_WEEK1 <- bind_rows(ES27_EPIWEEK_WEEK1_list, .id = "State")
```

ES27 EPIWEEK WEEK2

```
start_time <- Sys.time()

ES27_EPIWEEK_WEEK2_list <- mclapply(list_of_states, ES_EPIWEEK, ES27=TRUE, n_weeks_ahead=2,week_lag=1, n
  setNames(names(list_of_states))

end_time <- Sys.time()
run_time <- end_time - start_time

print(run_time)
```

```
## Time difference of 17.98568 mins
```

```
# Combine the list of data frames into a single data frame with names as a column
ES27_EPIWEEK_WEEK2 <- bind_rows(ES27_EPIWEEK_WEEK2_list, .id = "State")
```

ES27 EPIWEEK WEEK3

```
start_time <- Sys.time()

ES27_EPIWEEK_WEEK3_list <- mclapply(list_of_states, ES_EPIWEEK, ES27=TRUE, n_weeks_ahead=3,week_lag=1, n
  setNames(names(list_of_states))

end_time <- Sys.time()
run_time <- end_time - start_time

print(run_time)
```

```
## Time difference of 17.8958 mins
```

```
# Combine the list of data frames into a single data frame with names as a column
ES27_EPIWEEK_WEEK3 <- bind_rows(ES27_EPIWEEK_WEEK3_list, .id = "State")
```

ES27 EPIWEEK WEEK4

```
start_time <- Sys.time()

ES27_EPIWEEK_WEEK4_list <- mclapply(list_of_states, ES_EPIWEEK, ES27=TRUE, n_weeks_ahead=4,week_lag=1 ,n
  setNames(names(list_of_states))

end_time <- Sys.time()
run_time <- end_time - start_time

print(run_time)
```

```
## Time difference of 17.96919 mins
```

```r
# Combine the list of data frames into a single data frame with names as a column
ES27_EPIWEEK_WEEK4 <- bind_rows(ES27_EPIWEEK_WEEK4_list, .id = "State")
```

```r
save.image("EPIWEEK_MODELS_influenza_hospitalization_nolog.Rdata")
```

ES64 EPIWEEK WEEK1

```r
start_time <- Sys.time()

ES64_EPIWEEK_WEEK1_list <- mclapply(list_of_states, ES_EPIWEEK, ES64=TRUE, n_weeks_ahead=1,week_lag=1, n
  setNames(names(list_of_states))

end_time <- Sys.time()
run_time <- end_time - start_time

print(run_time)
```

```
## Time difference of 51.93265 mins
```

```r
# Combine the list of data frames into a single data frame with names as a column
ES64_EPIWEEK_WEEK1 <- bind_rows(ES64_EPIWEEK_WEEK1_list, .id = "State")
```

ES64 EPIWEEK WEEK2

```r
start_time <- Sys.time()

ES64_EPIWEEK_WEEK2_list <- mclapply(list_of_states, ES_EPIWEEK, ES64=TRUE, n_weeks_ahead=2,week_lag=1, n
  setNames(names(list_of_states))

end_time <- Sys.time()
run_time <- end_time - start_time

print(run_time)
```

```
## Time difference of 51.46322 mins
```

```r
# Combine the list of data frames into a single data frame with names as a column
ES64_EPIWEEK_WEEK2 <- bind_rows(ES64_EPIWEEK_WEEK2_list, .id = "State")
```

ES64 EPIWEEK WEEK3

```r
start_time <- Sys.time()

ES64_EPIWEEK_WEEK3_list <- mclapply(list_of_states, ES_EPIWEEK, ES64=TRUE, n_weeks_ahead=3,week_lag=1, n
  setNames(names(list_of_states))

end_time <- Sys.time()
run_time <- end_time - start_time

print(run_time)
```

```
## Time difference of 47.70147 mins
```

```
# Combine the list of data frames into a single data frame with names as a column
ES64_EPIWEEK_WEEK3 <- bind_rows(ES64_EPIWEEK_WEEK3_list, .id = "State")
```

ES64 EPIWEEK WEEK4

```
start_time <- Sys.time()

ES64_EPIWEEK_WEEK4_list <- mclapply(list_of_states, ES_EPIWEEK, ES64=TRUE, n_weeks_ahead=4,week_lag=1,m
  setNames(names(list_of_states))

end_time <- Sys.time()
run_time <- end_time - start_time

print(run_time)
```

```
## Time difference of 42.22869 mins
```

```
# Combine the list of data frames into a single data frame with names as a column
ES64_EPIWEEK_WEEK4 <- bind_rows(ES64_EPIWEEK_WEEK4_list, .id = "State")
```

```
save.image("EPIWEEK_MODELS_influenza_hospitalization_nolog.Rdata")
```