

ADJACENT STATES ARIMAX MODEL - Influenza hospitalization data

Victor Felix

March 19, 2025

This function utilizes ensembles and single automatic ARIMAX models which have mean cases by adjacent states 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. 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 DATASET !!!!!!!!!!!!!!!!!!!!!

```
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
```

```
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
```

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

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

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

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

```
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
```

```
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_ADJACENT_48.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 ADJACENT WEEK1 - ALL STATES

IF YOU WANT A FAST RESULT RUN ONLY THE AUTO_example, the others take longer.

```
start_time <- Sys.time()

AUTO_ADJACENT_WEEK1_list <- mclapply(list_of_states, ES_ADJACENT, auto=TRUE, n_weeks_ahead=1,list_of_st=
  setNames(names(list_of_states))

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

print(run_time)
```

Time difference of 9.920608 mins

```
# FINAL DATAFRAME
AUTO_ADJACENT_WEEK1 <- bind_rows(AUTO_ADJACENT_WEEK1_list, .id = "State")
```

AUTO ADJACENT WEEK2 - ALL STATES

```
start_time <- Sys.time()

# RUN MODEL
AUTO_ADJACENT_WEEK2_list <- mclapply(list_of_states, ES_ADJACENT, auto=TRUE, n_weeks_ahead=2,list_of_st=
  setNames(names(list_of_states))

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

print(run_time)
```

```
## Time difference of 9.963827 mins
```

```
# FINAL DATAFRAME
```

```
AUTO_ADJACENT_WEEK2 <- bind_rows(AUTO_ADJACENT_WEEK2_list, .id = "State")
```

AUTO ADJACENT WEEK3 - ALL STATES

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

```
AUTO_ADJACENT_WEEK3_list <- mclapply(list_of_states, ES_ADJACENT, auto=TRUE, n_weeks_ahead=3, list_of_states=list_of_states,
  setNames(names(list_of_states)))
```

```
end_time <- Sys.time()
```

```
run_time <- end_time - start_time
```

```
print(run_time)
```

```
## Time difference of 9.643929 mins
```

```
# Combine the list of data frames into a single data frame with names as a column
```

```
AUTO_ADJACENT_WEEK3 <- bind_rows(AUTO_ADJACENT_WEEK3_list, .id = "State")
```

AUTO ADJACENT WEEK4 - ALL STATES

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

```
AUTO_ADJACENT_WEEK4_list <- mclapply(list_of_states, ES_ADJACENT, auto=TRUE, n_weeks_ahead=4, list_of_states=list_of_states,
  setNames(names(list_of_states)))
```

```
end_time <- Sys.time()
```

```
run_time <- end_time - start_time
```

```
print(run_time)
```

```
## Time difference of 9.676791 mins
```

```
# Combine the list of data frames into a single data frame with names as a column
```

```
AUTO_ADJACENT_WEEK4 <- bind_rows(AUTO_ADJACENT_WEEK4_list, .id = "State")
```

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

ES27 ADJACENT WEEK1 - ALL STATES

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

```
ES27_ADJACENT_WEEK1_list <- mclapply(list_of_states, ES_ADJACENT, ES27=TRUE, n_weeks_ahead=1, list_of_states=list_of_states,
  setNames(names(list_of_states)))
```

```
end_time <- Sys.time()
```

```
run_time <- end_time - start_time
```

```
print(run_time)
```

```
## Time difference of 16.45789 mins
```

```
# FINAL DATAFRAME
```

```
ES27_ADJACENT_WEEK1 <- bind_rows(ES27_ADJACENT_WEEK1_list, .id = "State")
```

ES27 ADJACENT WEEK2 - ALL STATES

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

```
ES27_ADJACENT_WEEK2_list <- mclapply(list_of_states, ES_ADJACENT, ES27=TRUE, n_weeks_ahead=2, list_of_states = list_of_states,
  setNames(names(list_of_states)))
```

```
end_time <- Sys.time()
```

```
run_time <- end_time - start_time
```

```
print(run_time)
```

```
## Time difference of 16.69681 mins
```

```
# FINAL DATAFRAME
```

```
ES27_ADJACENT_WEEK2 <- bind_rows(ES27_ADJACENT_WEEK2_list, .id = "State")
```

ES27 ADJACENT WEEK3 - ALL STATES

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

```
ES27_ADJACENT_WEEK3_list <- mclapply(list_of_states, ES_ADJACENT, ES27=TRUE, n_weeks_ahead=3, list_of_states = list_of_states,
  setNames(names(list_of_states)))
```

```
end_time <- Sys.time()
```

```
run_time <- end_time - start_time
```

```
print(run_time)
```

```
## Time difference of 17.04776 mins
```

```
# FINAL DATAFRAME
```

```
ES27_ADJACENT_WEEK3 <- bind_rows(ES27_ADJACENT_WEEK3_list, .id = "State")
```

ES27 ADJACENT WEEK4 - ALL STATES

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

```
ES27_ADJACENT_WEEK4_list <- mclapply(list_of_states, ES_ADJACENT, ES27=TRUE, n_weeks_ahead=4, list_of_states = list_of_states,
  setNames(names(list_of_states)))
```

```
end_time <- Sys.time()
```

```
run_time <- end_time - start_time
```

```
print(run_time)
```

```
## Time difference of 16.60477 mins
```

```
# FINAL DATAFRAME
ES27_ADJACENT_WEEK4 <- bind_rows(ES27_ADJACENT_WEEK4_list, .id = "State")
```

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

ES64 ADJACENT WEEK1 - ALL STATES

```
start_time <- Sys.time()

ES64_ADJACENT_WEEK1_list <- mclapply(list_of_states, ES_ADJACENT, ES64=TRUE, n_weeks_ahead=1, list_of_states=list_of_states,
  setNames(names(list_of_states)))

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

print(run_time)
```

Time difference of 48.15236 mins

```
# FINAL DATAFRAME
ES64_ADJACENT_WEEK1 <- bind_rows(ES64_ADJACENT_WEEK1_list, .id = "State")
```

ES64 ADJACENT WEEK2 - ALL STATES

```
start_time <- Sys.time()

ES64_ADJACENT_WEEK2_list <- mclapply(list_of_states, ES_ADJACENT, ES64=TRUE, n_weeks_ahead=2, list_of_states=list_of_states,
  setNames(names(list_of_states)))

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

print(run_time)
```

Time difference of 47.05213 mins

```
# Combine the list of data frames into a single data frame with names as a column
ES64_ADJACENT_WEEK2 <- bind_rows(ES64_ADJACENT_WEEK2_list, .id = "State")
```

ES64 ADJACENT WEEK3 - ALL STATES

```
start_time <- Sys.time()

ES64_ADJACENT_WEEK3_list <- mclapply(list_of_states, ES_ADJACENT, ES64=TRUE, n_weeks_ahead=3, list_of_states=list_of_states,
  setNames(names(list_of_states)))

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

print(run_time)
```

```
## Time difference of 45.34943 mins
```

```
# Combine the list of data frames into a single data frame with names as a column  
ES64_ADJACENT_WEEK3 <- bind_rows(ES64_ADJACENT_WEEK3_list, .id = "State")
```

ES64 ADJACENT WEEK4 - ALL STATES

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

```
ES64_ADJACENT_WEEK4_list <- mclapply(list_of_states, ES_ADJACENT, ES64=TRUE, n_weeks_ahead=4, list_of_st  
  setNames(names(list_of_states))
```

```
end_time <- Sys.time()
```

```
run_time <- end_time - start_time
```

```
print(run_time)
```

```
## Time difference of 41.45717 mins
```

```
# Combine the list of data frames into a single data frame with names as a column  
ES64_ADJACENT_WEEK4 <- bind_rows(ES64_ADJACENT_WEEK4_list, .id = "State")
```

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