# Using R in Seasonal Adjustment Production of Official Statistics October 2020

Brian C. Monsell[*]

**Abstract**

This paper details my work at the Bureau of Labor Statistics in two areas - developing R packages for use in seasonal adjustment production, and evaluating several methods of seasonal adjustment for weekly time series. The packages I'm developing extract seasonal adjustment diagnostics from the output of the seasonal package, duplicate the functionality of the TERROR utility produced by Agustin Maravall, and generate diagnostic plots used by analysts at BLS. In addition, I have examined the use of R routines developed at the Federal Reserve Board (implementing the Cleveland method) and the creators of JDemetra+ (implementing a canonical decomposition of a fractional airline model) and compared them to the current method at use at BLS.

**Key Words:** Diagnostics, high-frequency time series, signal extraction

## Disclaimer

This report is released to inform interested parties of research and to encourage discussion. The views expressed on statistical issues are those of the author and not necessarily those of the U. S. Bureau of Labor Statistics.

## 1. Introduction

This paper gives a short description of a group of R packages under developement at the Bureau of Labor Statistics designed to aid evaluation of official seasonal adjustments at the Bureau of Labor Statistics.

After that, I show an examination of two methods of seasonally adjusting weekly time series, both of which have been implented in R (see [17]):

- The Cleveland method, as programmed by the U. S. Federal Reserve and Bureau of Labor Statistics (see [6]);
- The Fractional Airline model, as programmed by the JDemetra+ team (see [10]).

I will give a brief summary of each method, and will look at output and graphs of adjustments generated by these methods for a preliminary evalutation of the strengths and weaknesses of these implementations as compared with the current method of seasonally adjusting weekly series at the Bureau of Labor Statistics.

## 2. R Packages Under Development

There are a number of R packages currently under development to aid the seasonal adjustment of official time series. The first is a number of utilites that generate seasonal adjustment diagnostic summaries for a number of series run with the `seasonal` R package developed by Christophe Sax ([18], [19]). This summary collects several commonly used seasonal adjustment diagnostics generated by X-13ARIMA-SEATS ([21]) and mirrors the diagnostic summary produced by the Win X-13 software ([14]). It shows various

[*] U. S. Bureau of Labor Statistics, 2 Massachusetts Avenue, Washington, DC 20212

seasonality and seasonal adjustment diagnostics produced by X-13ARIMA-SEATS with shading to indicate whether the diagnostic is acceptable or not. There are R functions that collect the diagnostics, and another set of R functions that evaluate whether the diagnostics are acceptable based on a user-defined criteria.

Another proposed package is a set of routines that generates graphs useful to seasonal adjustment analysts. It is based on an internal set of SAS graphics routines developed at the Bureau of Labor Statistics by Tom Evans. These R routines generate time series graphs of tables from the X-13ARMIA-SEATS output, either from an individual object generated from the `seasonal` package or a list of those objects. Special plots for forecasts, seasonal factors, and spectral plots of series can also be generated, and options to denote outliers in the series or recession periods can be specified.

There is also an omnibus function that produces a number of plots for a given series, and uses the `R.devices` package [1] in R to generate a PDF or PNG file with the requested graphs.

A number of other utilities are being developed as well, including

- routines that select the X-11 seasonal filter based on the magnitude of the estimate of the seasonal moving average coefficient from the airline model as in [4] and [7],
- a function that duplicates the functionality of the TERROR software ([3]) that performs quality control on time series based on one step ahead forecasts,
- a package of routines that generate reference week holiday regressors for modeling time series.

Almost all of the coding is finished. The work now concentrates on documentating some of the routines, package creation, and internal testing. This phase is expected to continue through the Fall of 2020.

## 3. Weekly Seasonal Adjustment

The rest of this paper deals with weekly seasonal adjustment, which poses problems not seen in monthly or quarterly seasonal adjustment.

Many of these problems have to do with the composition of the Gregorian calendar. One can think of weekly series being composed of mostly 52-week years, with a number of 53-week years occurring roughly every 5-7 years.

Another way of thinking about weekly series would be to consider them series with a fractional periodicity, that each year has 52 weeks and a fragment of another week, rather than a period of 12 or 4 as in monthly or quarterly series.

There are also more moving holidays to account for - monthly series are limited to holidays like Easter and Labor Day, but practically all official US holidays move from one week to another in a given year. Table 1 shows the weeks that specific US holidays occur; notice how many times holidays move from one week to another from one year to another.

There are other times when it is necessary to incorporate holidays for special situations that happen rarely but can have a large effect on the series; an example of this would be having an indicator variable when Independence Day falls on a Wednesday.

### 3.1 Cleveland Method

The Cleveland method, described in [6], estimates weighted regressions of trigonometric seasonal terms for each year of the series. This allows the seasonality to evolve from year to year, rather than a fixed representation of the seasonal component as in previous implementaitons. Impelementations of this method also allow for holiday and other external regressors, which can be used for outliers or other effects.

The seasonal model described in [6] is given in Equation 1 :

$$\begin{aligned} w_t = &\sum_{i=1}^{k}(a_i \sin(\frac{2\pi i Y_t}{N_y}) + b_i \cos(\frac{2\pi i Y_t}{N_y}))+ \\ &\sum_{i=1}^{l}(c_i \sin(\frac{2\pi i M_t}{N_m}) + d_i \cos(\frac{2\pi i M_t}{N_m})), \end{aligned} \qquad (1)$$

where $N_y$ and $N_m$ are the number of days in the year and the current month, $Y_t$ and $M_t$ are the day of the year and day of the month where week $t$ ends, and $l$ and $k$ are the number of terms.

This paper examines two implementations of this method: the MoveReg Fortran program used at the U. S. Bureau of Labor Statistics (BLS) to adjust for Unemployment Claims data from the Department of Labor (see [5]), and an R implementation used by the Federal Reserve Board to seasonally adjust weekly monetary supply.

### *3.1.1 MoveReg Fortran program*

The MoveReg program has recently been revised at BLS to Version 5.0 - this version can generate an additive seasonal adjustment as well as the default log-additive adjustment. The output has been updated to allow for additive factors, as well as facilitate the input of components into R for further analysis.

The program can model up to 12 US moving holidays. One of those holidays is New Year's Day, which is somewhat different from the other holidays as this holiday always occurs in Week 1 by definition. In practice, a holiday regressor is generated so that when New Year's Day falls on Sunday, corresponding to a late Week 1, its effects are split between Weeks 1 and 2; otherwise, its effects come in Week 2. This implies a regressor defined as follows:

$$NYD_t = \begin{cases} 2, & \text{if t is in Week 2 and New Year's Day does not fall on a Sunday;} \\ 1, & \text{if t is in Week 1 or 2 and New Year's Day falls on a Sunday;} \\ 0, & \text{otherwise.} \end{cases}$$

For more information on the treatment of official holidays in MoveReg, see [6].

Users can also include user-defined holidays by reading in regressors for that particular holiday. Examples of user-defined holidays used in practice is given in Table 2.

MoveReg also allows users to specify point and level change outliers (by year and week), though there is not an automatic identification option. Two years of forecasts are also produced for the adjustment factors and the holiday component by default.

### *3.1.2 Fedres R routines*

The R routines used by the Federal Reserve Board (which I will refer to here as Fedres) assumes that the series is an indexed time series as defined in the tis R package (see [11]). This is a package that defines a time series class of objects with utilities that duplicates the functionality of the FAME database. We use this package with the output of the other signal extraction procedures to simplify comparison between the methods.

For example, the following R commands would read a weekly series from an Excel file into an R data frame (using [20]), set the starting date of a weekly time series, and generate a weekly time series where the week is assumed to end on Saturday.

```
ic_df <- openxlsx::read.xlsx("uihis.xlsx", sheet=1)
uihis_start <- c(1988, 1)
ic_tis <- tis::tis(ic_df$ic, start = uihis_start,
                      tif = "wsaturday")
```

The `Fedres` routines include terms for many of the same US Holidays as does `MoveReg`. You can incorporate outliers into the `Fedres` procedure, but a matrix with the outlier regressors needs to generated externally. I have written R routines to generate the weekly point and level change outliers needed. There is no automatic outlier identification procedure within `Fedres`.

The Cleveland model estimated by the two implementation is slightly different - `MoveReg` does not estimate the within month effects embodied in the second term of Equation 1, while `Fedres` does. The two implementations have similar variables that need to be set before generating an adjustment.

Current practice at the Federal Reserve is to estimate the AICC over a number of settings for these arguments and pick the model with the lowest value of AICC. BLS analysts have empirically determined settings that work well for their series.

### 3.2 Fractional Airline Model

The second method an implementation of the fractional airline model in R by the Bank of Belgium as part of an R package that access a Java library under development for the JDemetra+ program. JDemetra+ is a tool for seasonal adjustment developed by the National Bank of Belgium in cooperation with the Deutsche Bundesbank and Eurostat in accordance with the Guidelines of the European Statistical System (ESS).

The `rjdhighfreq` package contains routines for the modelling and seasonal adjustment of high frequency series (see [16]). One of those methods is the fractional airline decomposition examined in this paper. A fractional airline model can be viewed as first order Taylor expansion of an ”airline model” ($ARIMA(011)(011)_s$, where $s$ is the seasonal period) with non-integer periodicity (in the weekly case, $s = 365.25/7 = 52.1726$).

In the case of a weekly model, the autoregressive polynomial of the model is

$$(1 - B)(1 - (.82B^{52}) - (.18B^{53})) \tag{2}$$

and the moving average polynomial is

$$(1 - \theta B)(1 - (.82\Theta B^{52}))(1 - (.18\Theta B^{53})). \tag{3}$$

A canonical decomposition is performed as in [2] and the different components are estimated by means of the Kalman smoother (with exact diffuse initialization). These adjustments are analogous to those of the SEATS software developed by Gómez and Maravall (see [8] and [9]).

The latest version of the high dimensional modeling R package includes several useful innovations, including:

- automatic outlier identification for level shift and point outliers, with the ability to specify a critical level for the outlier t$-$statistic;

- fixed regression term, which can be used to incorporate holiday and other outlier effects;
- optional standard error term for the components;
- ability to generate forecasts and backcasts;
- R routines that can be used to generate moving holiday regressors to include in the joint model.

In order to facilitate comparison with the other methods, I used holiday regressors as defined in the `MoveReg` Fortran program. These holiday regressors were generated from R functions developed by the author of this paper. Other R functions that allow for the generation of holiday and outlier effects and the assignment of outlier effects to different components of the signal extraction (point outliers to the irregular component, level change outliers to the trend component) are available in a package developed by the author.

## 4. Weekly Initial Claims For Unemployment

We will apply these methods to the weekly series of initial claims for unemployment insurance reported by the states and published by Department of Labor. This series is affected by the Covid-19 pandemic.

Figure 1 shows that the series is quite seasonal up until the pandemic effects starts in the third week of March. The series increases by an order of magnitude - the maximum before the pandemic period less than 750 thousand, whereby the beginning of April the number of claims was over 6 million.

Figure 2 concentrates on the pandemic period - note how the series dramatically changes in level and then starts to slowly decrease over the next few months.

### 4.1 Methodology

To compare these weekly adjustment methods the following model assumptions are used. The change in level from the pandemic is accounted for with a series of point outliers at the end of the series. This practice is noted in [15], a paper about seasonal adjustment practice at the US Census Bureau during the Great Recession.

An additive model for the seasonal adjustment is specified for all methods, as suggested in a blog post by Heerdt and Wright ([13]), as using a log transformation with the large changes in level at the end of the series might lead to over or under adjustments.

The same outliers and holiday regressors specified for the official adjustment of Initial Claims are used for all the methods. These regressors include:

- 10 official US moving holidays (New Years Day, Martin Luther King's Birthday, President's Day, Easter, Memorial Day, Independence Day, Labor Day, Columbus Day, Veterans Day, and Thanksgiving);
- 3 user-defined moving holiday regressors (detailed in Table 2);
- 23 point outliers before the pandemic (detailed in Table 3);
- 13 pandemic point outliers (weeks 12-24, 2020, detailed in Table 3).

In this empirical analysis, four different settings for our signal extraction methods are examined:

- **MoveReg**, run with Cleveland model settings used in official Department of Labor adjustments;
- **Fedres (Best)**, Cleveland model settings determined by the lowest AICC from the estimated model;
- **Fedres (BLS)**, Cleveland model settings used in official Department of Labor adjustments;
- **Fractional Airline Decomposition**.

Table 4 gives the estimated coefficients for the fractional airline model.

### 4.2 Plots of the Seasonal Adjustments

Figure 3 shows the four seasonal adjustments generated from the methods in the last section. There is not much difference visually between the methods.

The level of the fractional airline model decomposition is higher than the seasonal adjustment for `MoveReg` for most observations in the pandemic period. There also appears to be a difference in level for `Fedres` (BLS) during the pandemic period

Figure 4 shows just the seasonal adjustments for `MoveReg` and the fractional airline model for a brief span before the pandemic. Here we see that the red line of the airline decomposition is smoother than the `MoveReg` adjustment, with several divots in the `MoveReg` adjustment that the airline adjustment smooths over, and the level of the airline decomposition is a little higher than that of `MoveReg`.

### 4.3 Weekly Factor Subplots

Figures 5 to 7 are weekly subplots of the combined adjustment factors. The combined factors appear to be very spiky for most weeks. This can be attributed to the presence of moving holiday effects. These effects can come in and out of a given week depending on the year examined; refer to Table 1 for details for specific holidays.

`MoveReg` and the fractional airline model factors appear to be very similar and the factors generated by the `Fedres` method are quite erratic during the pandemic, especially for those factors generated by BLS options.

Figure 5 is the weekly subplots for `MoveReg`. Note the spikey nature of the combined factors for many of the weeks with holidays - New Year's, MLK day, and Veteran's Day.

Figure 6 is the weekly subplots for the fractional airline model decomposition. They are very similar in shape and position to those of `MoveReg`.

Figure 7 is the weekly subplots for `Fedres` using BLS options. Note how the combined factors trail up starting at the 12th week, with increasing magnitude until the factors start trailing down starting around the 18th week.

### 4.4 Pandemic Period Plots

To get an idea of why this is happening we'll examine the pandemic adjusted series. To generate the pandemic adjusted series we're generate factors for the point outliers specified during the pandemic period. Once those actors are generated, they are removed from the original series.

In general, `MoveReg` and the fractional airline model produces pandemic adjustments that look very similar. `Fedres` seem to over-adjust the series at the end of the pandemic.

Subsequent analysis showed that changing input parameters for the Cleveland model effects the degree of overadjustment in the estimated pandemic effects.

Figure 8 is the graph of the original series adjusted for pandemic effects. Note how `Fedres` first under-adjusts for the pandemic and then over-adjusts. This is most clear in the run done with BLS options, but it is also evident in the run done with the best AICC options. Note how the series generated using the BLS options generates a series that is less than zero.

The pandemic adjustments for `MoveReg` and the fractional airline model are quite similar.

## 5. Future Work

In future work, I plan to continue examining JDemetra+'s implementation of the Fractional Airline decomposition developed by the Bank of Belgium. Jean Palate's team continues to add useful features to the high dimension modeling package. We also want to compare these adjustments with those from a structural time series model based on the model of [12] estimated using Proc SSM in SAS, as we examine possible replacements for the Fortran implementation of `MoveReg`.

I will also continue working on R coding projects and hope to have some packages available for internal testing by the end of the year.

## 6. Acknowledgements

I want to acknowledge the contributions of several people:

- Tom Evans of the Bureau of Labor Statistics, for providing me with the claims data and the source code of `MoveReg`, as well as helpful discussions of results along the way and comments on drafts of this paper,
- Brian Preslopsky from the US Federal Reserve, for providing me with the Federal Reserve's R implementation of the Cleveland method used for their money supply series, and
- Jean Palate of the Bank of Belgium, for providing me with the preliminary version of their high frequency modeling package implemented in R.

### References

[1] H. Bengtsson. *R.devices: Unified Handling of Graphics Devices*, 2019. R package version 2.16.1, `https://CRAN.R-project.org/package=R.devices`.

[2] J. P. Burman. Seasonal Adjustment by Signal Extraction. *Journal of Royal Statistical Society A*, 143(3):321–337, 2004.

[3] G. Caporello and A. Maravall. A Tool for Quality Control of Time Series Data Program TERROR, 2004.

[4] Y.-J. Chu, G. C. Tiao, and W. R. Bell. A Mean Squared Error Criterion for Comparing X-12-ARIMA and Model-Based Seasonal Adjustment Filters. *Taiwan Economic Forecast and Policy*, 43(1):1–32, 2012. Available online at `https://www.census.gov/ts/papers/chutiaobelltefp.pdf`.

[5] W. P. Cleveland, J. Fields, S. Scott, T. Evans, and D. Byun. *MoveReg Guide Version 3.01 (DRAFT).* U.S. Bureau of Labor Statistics, U.S. Department of Labor, 2015.

[6] W. P. Cleveland and S. Scott. Seasonal Adjustment of Weekly Time Series with Application to Unemployment Claims and Steel Production. *Journal of Official Statistics*, 23(2):209–221, 2007.

[7] R. Depoutot and C. Planas. Comparing Seasonal Adjustment and Trend Extraction Filters with Application to a Model-Based Selection of X11 Linear Filters. Eurostat Working Papers, No. 9/1998/A/9, 1998.

[8] V. Gómez and A. Maravall. Programs TRAMO and SEATS : Instructions for the User (Beta Version: June 1997). Banco de España, Servicio de Estudios, DT 9628., 1996.

[9] V. Gómez and A. Maravall. Seasonal Adjustment and Signal Extraction in Economic Time Series. In D. Pena, G. C. Tiao, and R. S. Tsay, editors, *A Course in Time Series Analysis*. J. Wiley and Sons, New York, NY, 2001.

[10] S. Grudkowska. *JDemetra+ Reference Manual, Version 2.1*. Narodowy Bank Polski, Department of Statistics, Warsaw, Poland, 2016. `https://ec.europa.eu/eurostat/cros/system/files/jdemetra_reference_manual_version_2.1_0.pdf`.

[11] J. Hallman. *tis: Time Indexes and Time Indexed Series*, 2020. R package version 1.38, `https://CRAN.R-project.org/package=tis`.

[12] A. Harvey, S. J. Koopman, and M. Riani. The Modeling and Seasonal Adjustment of Weekly Observations. *Journal of Business and Economic Statistics*, 15(2):354–368, 1997. `http://links.jstor.org/sici?sici=0735-0015%28199707%2915%3A3%3C354%3ATMASAO%3E2.0.CO%3B2-6`.

[13] K. Heerdt and J. H. Wright. Beware of Seasonal Adjustment, April 2020. `https://cfe.econ.jhu.edu/beware-seasonal-adjustment/` [Online; posted 24-April-2020].

[14] D. Lytras. *Win X-13 Version 3.0: A Windows Interface for X-13ARIMA-SEATS*. U.S. Census Bureau, U.S. Department of Commerce, Washington, DC, 2020. `https://www.census.gov/data/software/x13as/winx13/documentation.html`.

[15] D. Lytras and W. R. Bell. Modeling Recession Effects and the Consequences on Seasonal Adjustment. *Proceeding of the American Statistical Association, Business and Economic Statistics Section*, 2013. Available online at `https://www.census.gov/ts/papers/jsm2013lytrasfinal.pdf`.

[16] J. Palate. *rjdhighfreq: Routines for modelling and seasonal adjustment of high frequency series, Version 0.0.4*. Bank of Belgium, Brussels, Belgium, August 2020. `https://github.com/palatej/rjdhighfreq`.

[17] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2019. `https://www.R-project.org/`.

[18] C. Sax. *seasonal: R Interface to X-13ARIMA-SEATS. R package version 1.7*, 2018. `https://CRAN.R-project.org/package=seasonal`.

[19] C. Sax and D. Eddelbuettel. Seasonal Adjustment by X-13ARIMA-SEATS in R. *Journal of Statistical Software, Articles*, 87(11):1–17, 2018. `https://www.jstatsoft.org/v087/i11`.

[20] P. Schauberger and A. Walker. *openxlsx: Read, Write and Edit xlsx Files*, September 2020. R package version 4.1.5.

[21] Time Series Software Group. *X-13ARIMA-SEATS Reference Manual, Version 1.1*. U.S. Census Bureau, U.S. Department of Commerce, 2020. `https://www2.census.gov/software/x-13arima-seats/x13as/windows/documentation/docx13ashtml.pdf`.

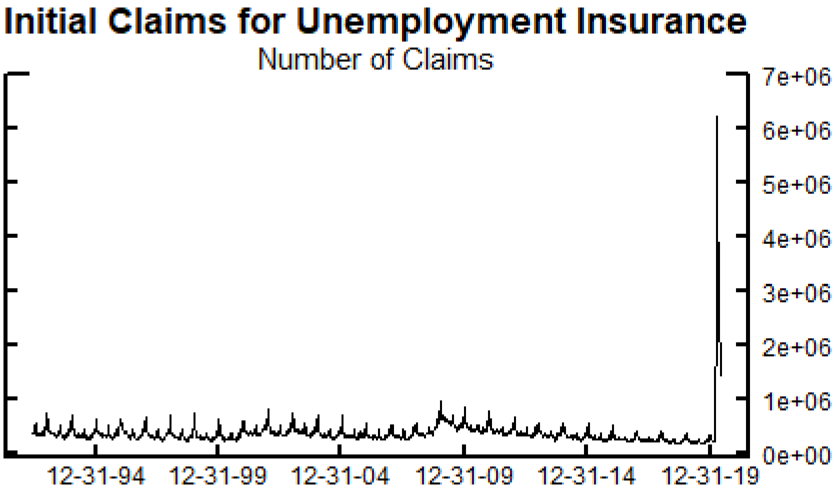

**Figure 1**: Initial Claims for Unemployment Insurance

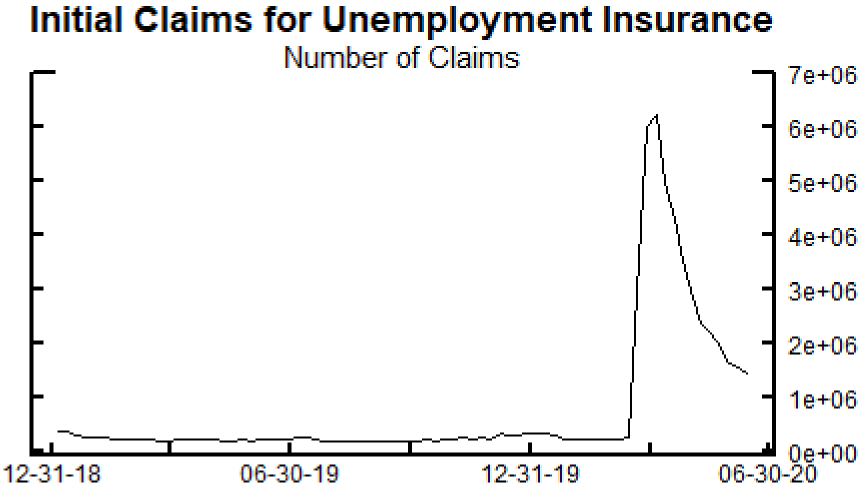

**Figure 2**: Initial Claims for Unemployment Insurance, Starting 2019

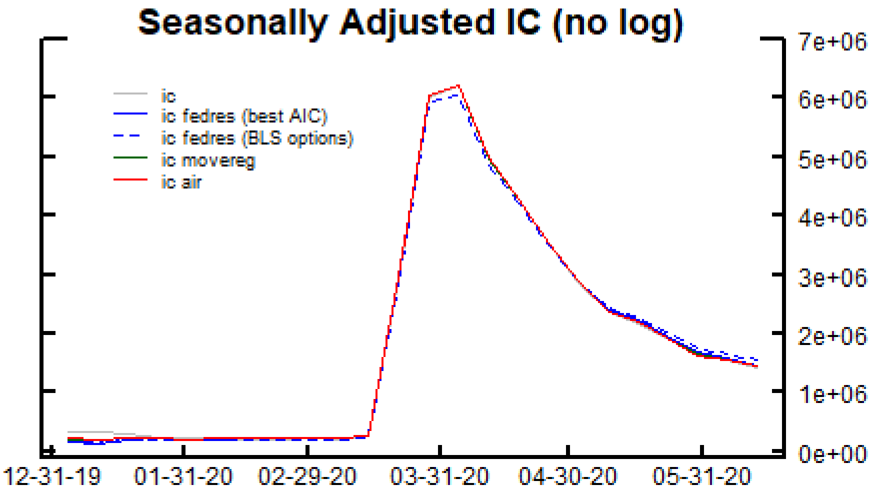

**Figure 3**: Seasonally Adjusted Initial Claims for Unemployment Insurance, from the start of 2020

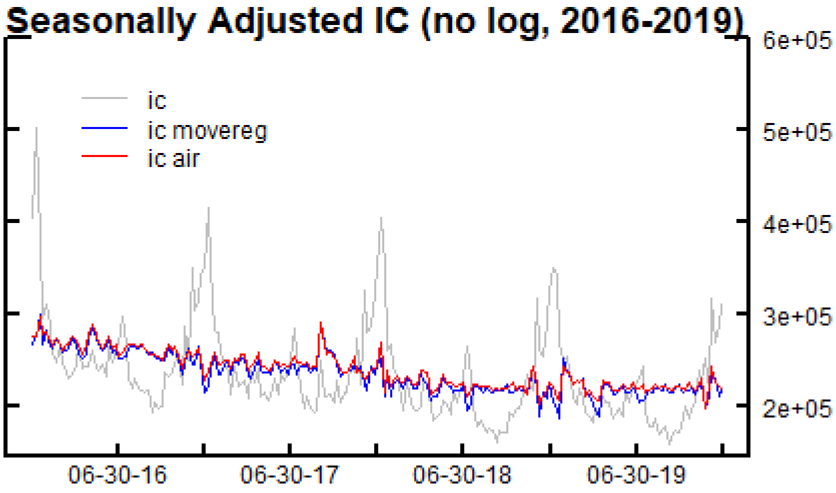

**Figure 4**: Seasonally Adjusted Initial Claims for Unemployment Insurance from 2016 to 2019, for `MoveReg` and the fractional airline model

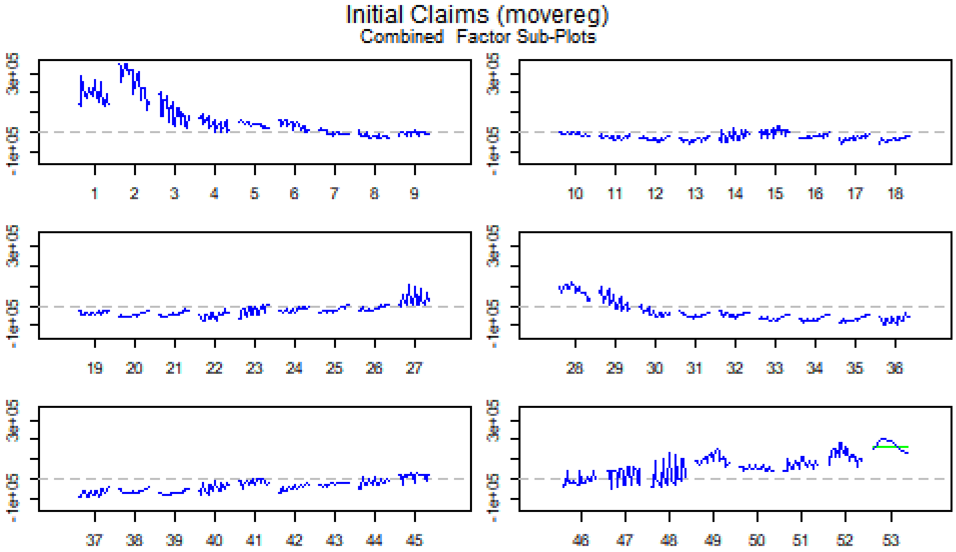

**Figure 5**: Weekly combined factor subplots for the MoveReg adjustments of Initial Claims for Unemployment Insurance

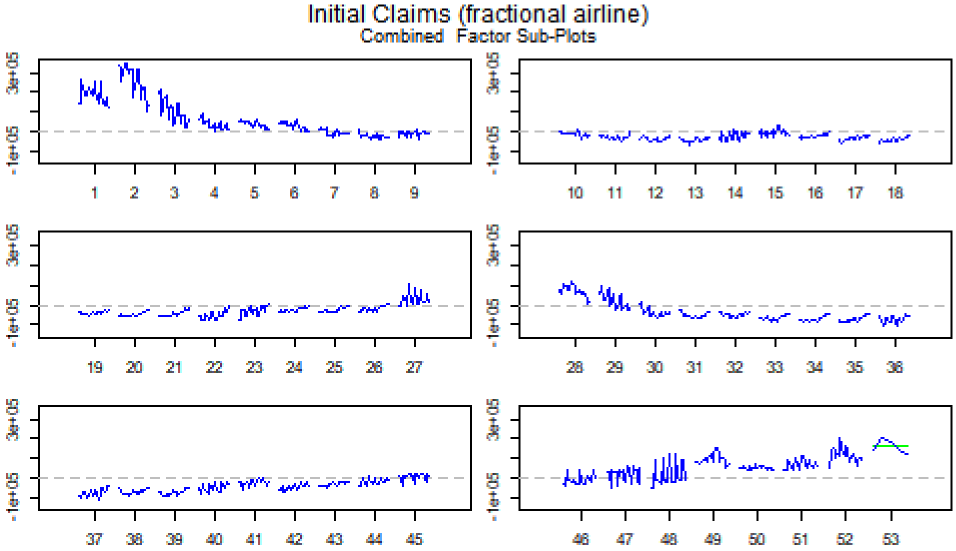

**Figure 6**: Weekly combined factor subplots for the Fractional Airline decomposition of Initial Claims for Unemployment Insurance

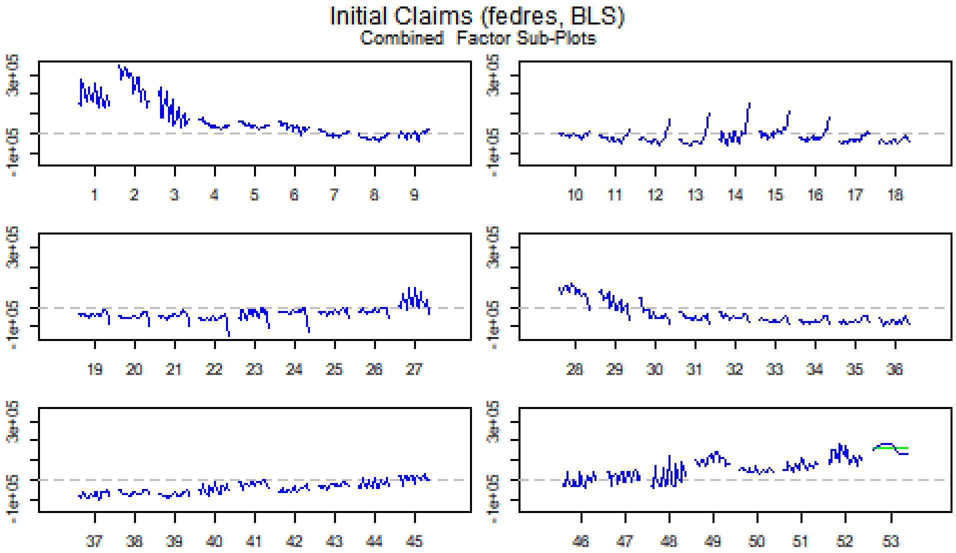

**Figure 7**: Weekly combined factor subplots for the Fedres (BLS) adjustments of Initial Claims for Unemployment Insurance

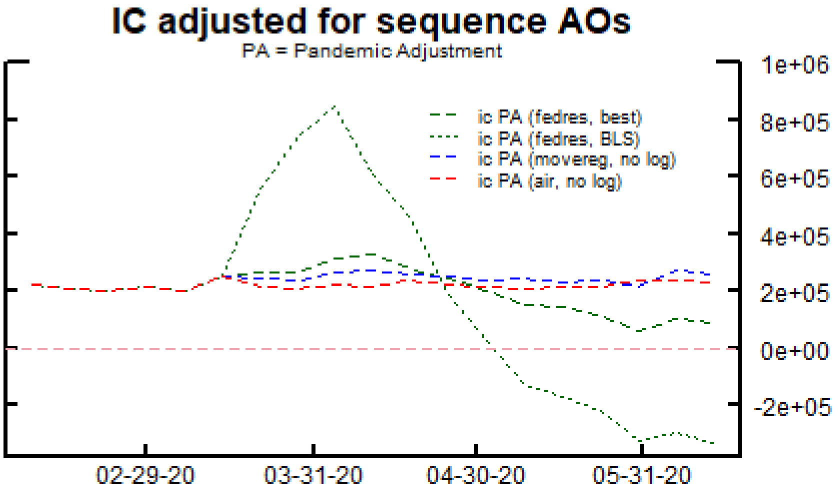

**Figure 8**: Initial Claims for Unemployment Insurance adjusted for sequence AOs during the pandemic period

**Table 1**: Weeks that Selected US Official Holidays Occur, 2000-2021

| Year | MLK Day | President’s Day | Good Friday and Easter | Memorial Day | Fourth of July | Labor Day | Columbus Day | Veteran’s Day | Thanksgiving |
|---|---|---|---|---|---|---|---|---|---|
| 2000 | 4 | 9 | 18 | 23 | 28 | 37 | 42 | 46 | 48 |
| 2001 | 3 | 8 | 16 | 22 | 27 | 36 | 41 | 46 | 47 |
| 2002 | 4 | 8 | 14 | 22 | 27 | 36 | 42 | 46 | 48 |
| 2003 | 4 | 8 | 17 | 22 | 27 | 36 | 42 | 46 | 48 |
| 2004 | 4 | 8 | 16 | 23 | 28 | 37 | 42 | 46 | 48 |
| 2005 | 4 | 9 | 14 | 23 | 28 | 37 | 42 | 46 | 48 |
| 2006 | 3 | 8 | 16 | 22 | 27 | 36 | 41 | 45 | 47 |
| 2007 | 3 | 8 | 15 | 22 | 27 | 36 | 41 | 46 | 47 |
| 2008 | 4 | 8 | 13 | 22 | 27 | 36 | 42 | 46 | 48 |
| 2009 | 4 | 8 | 16 | 22 | 27 | 37 | 42 | 46 | 48 |
| 2010 | 4 | 8 | 15 | 23 | 28 | 37 | 42 | 46 | 48 |
| 2011 | 4 | 9 | 18 | 23 | 28 | 37 | 42 | 46 | 48 |
| 2012 | 3 | 8 | 15 | 22 | 27 | 36 | 41 | 46 | 47 |
| 2013 | 4 | 8 | 14 | 22 | 27 | 36 | 42 | 46 | 48 |
| 2014 | 4 | 8 | 17 | 22 | 27 | 36 | 42 | 46 | 48 |
| 2015 | 4 | 8 | 15 | 22 | 27 | 37 | 42 | 46 | 48 |
| 2016 | 4 | 8 | 14 | 23 | 28 | 37 | 42 | 46 | 48 |
| 2017 | 3 | 8 | 16 | 22 | 27 | 36 | 41 | 45 | 47 |
| 2018 | 3 | 8 | 14 | 22 | 27 | 36 | 41 | 46 | 47 |
| 2019 | 4 | 8 | 17 | 22 | 27 | 36 | 42 | 46 | 48 |
| 2020 | 4 | 8 | 16 | 22 | 27 | 37 | 42 | 46 | 48 |
| 2021 | 4 | 8 | 15 | 23 | 28 | 37 | 42 | 46 | 48 |

**Table 2**: Special Holiday Indicator Variables Used

| User Holiday | Description |
|---|---|
| July 4 on Wednesday | Equal to one if July 4th occurs that week on a Wednesday, 0 otherwise |
| Xmas on Week 53 | Equal to one if Christmas occurs that week and is on the 53rd week, 0 otherwise |
| Xmas on Friday | Equal to one if Christmas occurs that week on a Friday, 0 otherwise |

**Table 3**: Outliers Included in Initial Claims Model (Pandemic Outliers in Bold)

| | | |
|---|---|---|
| Week 30, 1992 | Week 30, 1993 | Week 52, 1993 |
| Week 5, 1994 | Week 3, 1996 | Week 38, 2001 |
| Week 39, 2001 | Week 40, 2001 | Week 41, 2001 |
| Week 42, 2001 | Week 43, 2001 | Week 47, 2001 |
| Week 48, 2001 | Week 37, 2005 | Week 38, 2005 |
| Week 39, 2005 | Week 40, 2005 | Week 41, 2005 |
| Week 1, 2006 | Week 2, 2007 | Week 4, 2008 |
| Week 45, 2012 | Week 35, 2017 | **Week 12, 2020** |
| **Week 13, 2020** | **Week 14, 2020** | **Week 15, 2020** |
| **Week 16, 2020** | **Week 17, 2020** | **Week 18, 2020** |
| **Week 19, 2020** | **Week 20, 2020** | **Week 21, 2020** |
| **Week 22, 2020** | **Week 23, 2020** | **Week 24, 2020** |

**Table 4**: Estimated Coefficients for Fractional Airline Model

| | Coefficient | T-statistic |
|---|---|---|
| $\theta$ | 0.556 | 36.59 |
| $\Theta$ | 0.425 | 25.05 |