Seasonal Trend And Holiday Decomposition with Loess (STAHL): A Real-Time Approach to Analyzing High-Frequency Alternative Data

Sebastien Daniel[†], Vincent Haller[‡], Benoit Bellone[§]
This version[¶]: December 21, 2023

Abstract

This research note exhibits a new seasonal adjustment methodology developed at QuantCube Technology: Seasonal Trend And Holiday decomposition with Loess (STAHL). Derived from the STL procedure introduced by R. B. Cleveland et al. (1990), STAHL aims to be applied to alternative data series at multiple frequencies. It is able to proceed to Working Day Adjustment coping with time-varying or periodic calendars such as the Chinese New Year, to handle structural or periodic missing values, and to provide a point-in-time procedure to generate unrevised trend and seasonal components. In a first part, we describe the STAHL framework and its different key innovative steps: spectral identification of multiple seasonal frequencies, industrial preprocessing including resampling, seasonal adjustment with missing value handling, specific holiday adjustment and point in time seasonal trend extraction. In the second part, we provide multiple empirical illustrations based on high frequency data. We notably focus on the US Weekly Initial claims series during the outstanding periods of Covid outbreaks which raised critical issues for real-time seasonality extraction. As such we discuss the consequences of our point-in-time (ie. no revision) principle compared to the adjusted series produced by the US Department of Labour. Second, we focus on many different high frequency alternative data series to explore how STAHL deals with multiple seasonality and holidays. We notably focus on periodic missing value treatments and measure the Chinese Lunar Calendar impact on human activity captured through daily measures of NO2 Air pollution.

Key words : Seasonal adjustment, Calendar adjustment, Loess, High-frequency, Alternative data, Point-in-time estimations, Missing values

---

[†]Lead Data Scientist, QuantCube Technology

[‡]Data Scientist, QuantCube Technology

[§]Head of Research, QuantCube Technology, b [dot] bellone [at] quant-cube.com

[¶]Acknowledgements: We are grateful to the members of the third Workshop on Time Series Analysis and Statistical Disclosure Control Methods for Official Statistics where a first version of this paper was presented at the OECD in Paris on December 14 2023. We notably thank Dominique Ladiray, Gian Luigi Mazzi, Jean Palate, Annabelle Mourougane for their constructive remarks, and Nicolas De Roux and Amelie Sciberras for their help and suggestions.

1 Introduction

Macroeconomic nowcasting has increasingly involved the use of various types of massive data over the past decade. Alternative data refer to non-traditional sources of information that can provide unique insights. They can be clustered into four major categories: text data, geospatial data, geolocation data, and structured data. Text data can be recovered via multiple sources, mainly through the internet, such as social media data, professional blogs, news articles, job ads, web searches, or hotel and restaurant reviews, for instance.

The availability of high-resolution satellite imagery and the development of deep learning models have led to numerous applications allowing the recovery of various geospatial data from earth observation satellite images, atmospheric data, or radar data. Geolocation data can take the form of shipping traffic, flights, mobility, or vehicle transit numbers. Structured data can encompass prices of goods and services, real estate prices, internet queries, and web traffic.

QuantCube Technology aims to provide a competitive edge by uncovering hidden patterns, detecting emerging trends, and enhancing predictive models by leveraging all those alternative data. Most of these data are produced by QuantCube Technology at a daily frequency, seven days a week, without exception. These time series exhibit a rather short history, oscillating between five and just over ten years. As such, their statistical analysis must address many specific issues. A primary problem to tackle is the complexity of the multiplicity and potential interactions of seasonal patterns (notably annual, monthly, weekly, and daily cycles) and calendar effects (such as holidays, and time-varying non-Gregorian calendars like the Chinese or Hijri calendars).

Furthermore, some alternative data may suffer from high sensitivity to outliers at high frequencies, such as mobility bans during the COVID periods, and may be prone to periodic or structural missing values, like clouds disturbing the quality of satellite images at regular intervals. In addition, most financial and macro practitioners expect these alternative data sources to be point-in-time and un-revised, which presents a specific practical challenge from a seasonality extraction perspective. This article aims to precisely tackle all these reported issues by introducing a new seasonal adjustment framework developed at QuantCube Technology: Seasonal Trend and Holiday Decomposition based on LOESS (STAHL).

In the first section, this research note provides a brief literature review of the current state of the art in high-frequency seasonal adjustment procedures, along with their pros and cons from a practitioner's and industrial perspective. We particularly focus on the STL (Seasonal Trend Decomposition based on LOESS) approach and discuss why we consider an extended version of this procedure as the best compromise for point-in-time seasonal adjustment of alternative data. In the second section, we introduce the STAHL framework, describing the five steps of the algorithm and emphasizing its specific innovations: handling of missing values, adjustment for working days, and the inclusion of constraints for point-in-time calculation. While the third section details the treatment of periodic or structural missing values as one innovative advantage of such a procedure, the fourth section focuses on a completely new approach in STL to deal with working day adjustment. The fifth section details how point-in-time seasonal adjustment can be conducted using asymmetric filters and hints at potential improvements. The final section is devoted to empirical illustrations, where different types of high-frequency data are studied with STAHL: weekly US initial unemployment claims and daily China air pollution data based on satellite images. Both series provide interesting illustrations of the STAHL procedures, whether it is to produce point-in-time rolling seasonally adjusted series, manage outliers and extreme periods such as the COVID-19 episode, handle structural missing values, or address the impact of the Chinese Lunar Calendar.

2 Seasonal Adjustment Methods for High Frequency Alternative Data: a Review

The digital transformation process of our modern economies and the outbreak of the COVID-19 pandemic have enhanced an interest for the statistical treatment of alternative and high frequency data. Infra-monthly economic data are in strong demand to provide more timely early warning economic nowcasting indicators which require to tackle specific statistical treatment. Webel (2022) has provided the most recent, to our knowledge, general and very exhaustive review of such methods.

Let's note y_t a discrete time series with a seasonal periodicity of τ. For example, $\tau = 12$ for monthly data, $\tau = 52.18$ for weekly series and $\tau = 365.2425$ for yearly seasonality measured at daily frequency. The consensus in the statistical literature qualifies as a higher frequency (HF) time series if data is observed at infra-monthly intervals ($\tau > 12$) (resp. lower frequency time series (LF) if data is observed at monthly or lower periodicities ($\tau \leq 12$)).

While different parts of the literature focused on minute-by-minute, hourly, 4-hourly data, we will focus on daily and weekly data in this exercise, as QuantCube Technology produces a large variety of those kind of HF series in an industrial way seven days a week. At such frequencies, there are already multiple difficulties arising due to interactions of calendar and seasonal effects which are almost absent at low frequencies. For example, as described in Webel (2022), *fixed-holiday and end-of-period effects may depend on the particular days of the week which the corresponding events fall onto. Christmas effects may be noticeably different for 24 to 26 December falling onto Tuesday through Thursday Friday through Sunday. The same applies to the short-lived end-of-Q3 elevation in level if the final day of that quarter had not been a Monday.*

Calendar-related dynamics may become even more complicated when secular and religious activities mainly follow different calendars with HF data. Campante and Yanagizawa-Drott (2015) provide evidence that the Hijri (ie. muslim) calendar impacts economic time series such as growth, mobility and private consumption.[1] In this research, we focus on the impact of the Chinese

[1]Webel (2022) also hints at the lunar Hijri calendar as deviations from the solar Gregorian calendar, which is approximately 11 days shorter.

lunar calendar in the empirical sections to hint at a key improvement to STL related to STAHL.

As illustrated in Webel (2022) or in Proietti and Pedregal (2022), the seasonal profile of HF time series is highly complex due to the coexistence of multiple and super-imposed seasonal patterns with integer versus non-integer periodicities.[2]

The issue of data accessibility presents another quandary. Currently, high-frequency (HF) observations typically provide only a few years of history, which is often inadequate for reliably predicting all aspects of HF dynamics, including possible interaction effects. Conducting seasonal and calendar extraction on alternative data with an 'industrial', i.e., a strongly standardized approach, is a challenge that must be addressed with as parsimonious a model as possible. Alternative data can also suffer from a significant amount of missing values, due either to measurement issues or holiday unavailability. This adds another constraint to the seasonal adjustment method, which must be able to deal with both regular and irregular patterns of missing values.

2.1 Extending Conventional Methods from the X11 Family

Numerous statistical organizations worldwide use one of several recognized methods for seasonally-adjusting low frequency (LF) time series. These methods include the X-11 approach, the AutoRegressive Integrated Moving Average (ARIMA) model-based strategy, and the Structural Time Series (STS) models.

Ladiray, Palate, et al. (2018) provide some guidelines to adapt the 'X11 family' seasonal adjustment procedure first introduced in Ladiray and Quenneville (2012) to HF data[3], but also the STL (Seasonal Trend Decomposition using LOESS) introduced by R. B. Cleveland et al. (1990). They show that the main seasonal adjustment methods used for monthly and quarterly series, such as TRAMO-SEATS and X12-ARIMA, as well as STL, can be adapted to high frequency data which present multiple and non integer periodicities. For instance, TRAMO-SEATS can be modified using fractional ARIMA models and more efficient numerical algorithms. The non-parametric and iterative processes of X11 and STL can also be adapted after imputation of the missing values induced by the different lengths of months and year. But the authors argue that *the tuning of the multiple parameters of the methods might be cumbersome.*

Indeed, according to the authors, the tuning of X11 and STL parameters, namely the length of the filters used in the decomposition, needs to be improved. In the genuine X11 algorithm, and for monthly and quarterly series, the order of the moving averages is selected according to a "signal to noise" ratio. Thresholds have been defined by simulations and practice, which may be an obstacle for an industrial generalization of this procedure for series dealing with multiple high frequencies. For Ladiray, Palate, et al. (2018) large scale

[2] Seasonal periodicities of infra-weekly and shorter patterns will be integers but those of the monthly, quarterly and yearly patterns will be non-integers when consider at daily frequency.

[3] i.e. methods based on moving averages like X11, X11-ARIMA, X12-ARIMA and X-13 ARIMA-SEATS and methods based on Arima models like TRAMO-SEATS.

simulations have still to be done to understand the behavior of these ratios and to elaborate a decision rule for high frequency data.

2.2 Structural Time Series Models and its Variants

Proietti and Pedregal (2022) advocate for using Structural Time Series (STS) models on HF times series following the seminal modelling of Harvey and Koopman (1993) and Harvey, Koopman, and Riani (1997). Unobserved Component Models (UCM) can, by design, handle any kind of high-frequency data, but the models tend to be very "series specific," according to the authors. In particular, the selection of harmonics can be rather arbitrary and not that easy to conduct at scale. As a consequence, UCM models lack the desired flexibility we are looking for to implement an industrial seasonal adjustment procedure that could be applied to a wide range of different alternative data and frequencies. When testing UCM approaches, we notably experienced the same difficulties as indicated by Ollech (2021). We also faced severe identification and convergence issues in the maximum likelihood estimation of parameters. The optimization method may indeed exhibit convergence issues with high frequency data, leading to different local minima depending on the initialization, as pointed in Ladiray, Palate, et al. (2018). This leads to issues with the identification of the signal-to-noise ratios of the irregular component, and the separation between the trend component and lower frequency seasonal components when considering STS and UCM models as an industrial solution for high-frequency seasonal and calendar adjustment.

In the community of alternative data and machine learning players, Prophet, a Bayesian approach developed by Facebook's Core Data Science team (Taylor and Letham (2018)) is an often popular algorithm to conduct seasonal adjustment extraction. It has been particularly designed to provide a flexible and reliable forecasting tool that can be configured, interpreted and evaluated by subject-matter experts and analysts without great expertise in time series modelling. The general idea is to specify relatively sparse Unobserved Component models and impose priors on the unknown parameters. There are many pros in favor of Prophet: it has been designed to detect yearly, weekly, and daily seasonality in data and to be robust to outliers using L1 regularization in the modeling process. It offers an intuitive way to include holidays, provided theirs effects are known and well identified. However, Prophet was more designed as a forecasting package rather than offering a general statistical decomposition framework. As such it often works as a black-box making it harder to understand the model intricacies. Furthermore, the Prophet framework does not currently allow the seasonal pattern to evolve with time, making it too restrictive for our macroeconomics daily series. Besides, trends extraction modelling is based on a non linear saturating growth and trend change detection approach which may also seem to be too restrictive, and not adapted to a point in time procedure.

2.3 Toward an Extended STL

According to Ollech (2021), the semi-parametric STL approach introduced in R. B. Cleveland et al. (1990) offers an alternative technique to extract HF series seasonality. Rooted in a locally weighted regression smoother (LOESS), it stands out from X-11 due to its heightened flexibility concerning the frequency of the time series: STL can handle any type of frequency, not just monthly or quarterly.

A key strength of the STL algorithm, relative to STS, is its speedy computation, which enables the adjustment of varying seasonal frequencies within a unified iterative process, as well as being robust to outliers, making it suitable for data with anomalies. The primary purpose of STL is the decomposition of a time series into its trend, seasonal, and remainder components, making it straightforward to provide an interpretable decomposition. While it helps if the seasonal period is known, STL can be used without this knowledge, which can be a key strength when designing an industrial statistical decomposition framework. Given the simplicity of STL, it is somewhat easier to diagnose issues when they arise and to scale the approach. We will propose STAHL, an extended framework, to automatically assess holiday impacts in an integrated manner.

Ollech (2021) contributes to the existing literature by devising a seasonal adjustment routine for daily data, but also hints at STL's limitations to single seasonal frequencies as it cannot deal with calendar effects such as the influences of moving holidays. Ollech (2021) also hints at convergence issues of the RegARIMA model for holiday adjustment of daily time series. Ollech (2021) also points at needed developments of STL to enhance its computational speed and reliability of the outlier detection and estimation. Webel (2022) takes also stock of needed automatic procedures to produce HF seasonal adjustments. The extended STL and X-11 currently lack automatic selection rules for trend and seasonal filters given HF data. We found many studies about point-in-time vs in-sample seasonal estimates, and implied biases related to trend extraction, but to our knowledge there is no procedure fully dedicated to point in time seasonal adjustment, addressing notably asymmetric filtering problems. Our goal is notably to tackle those issues.

3 The STAHL Framework

Based on the well-known additive model STAHL decomposes a time series (Y_t) into a trend-cycle (T_t) component, a seasonal (S_t) component, a calendar-holiday component (H_t) and an irregular component (I_t) as in STL using LOESS regressions and moving averages introduced in R. B. Cleveland et al. (1990):

$$Y_t = T_t + S_t + H_t + I_t, t = 1..N$$

Each value is regressed in LOESS regressions on a local neighbourhood of a linear or quadratic fitted polynomials[4]. For any point in time, the weight of the observation x_i, is given by[5] :

$$v_i(x) = \left[1 - \left(\frac{|x_i - x|}{\lambda_q(x)}\right)^3\right]^3$$

with $\lambda_q(x) = |x_i - x_q|$ the distance between the q^{th} farthest x_i and x. The parameter q is key to disentangle the trend from the seasonal component : by determining the number of neighbouring observations in the local regression, the higher q, the smoother the identified seasonal factor. The x_i close to x have the largest weights which ultimately decrease to zero at the q^{th} farthest point.

As illustrated in Figure 1, STAHL consists of a modified STL inner loop following the k^{th} iteration made of seven stages focusing on extracting a first version of trend and seasonal components :

1. Trend and Holiday adjustment[6] $(S + I = Y - T - 0.5H)$
2. Preliminary subseries smoothing by LOESS of bandwidth n_s to yield a preliminary factor C_t^k including both irregular and seasonal components
3. Low-pass filter composed of moving average (Section 6) and LOESS asymmetric filters of bandwidth n_t to capture a low frequency L_t^k from C_t^k
4. First Seasonal Component : $S_t^k = C_t^k - L_t^k$
5. First Seasonally adjusted Series : $SA_t^k = Y_t^k - S_t^k$
6. First Trend Series T_t^k extracted from the SA_t^k series with a LOESS filter
7. First Irregular component : $I_t^k = Y_t - SA_t^k - T_t^k$

At this stage, the irregular component may contain outliers[6] and non-seasonal but regular holiday pattern effects. One of the key innovation with the STAHL framework is to add a second inner loop that will disentangle outliers from holiday estimations, and leads to a systemic construction of seasonally and holiday adjusted series.

[4]In the following STAHL implementation, we ignored the possibility of using quadratic fitted polynomials to estimate LOESS regressions and restricted it to the linear case function of the (weighted) observations x

[5]To minimize outliers contamination, LOESS regressions combine those weights of observations with Tukey biweight function

[6]Only half the holiday effect is removed for two reasons. First to give the seasonal component priority over the holiday component. Second so that holidays still show up as outliers in the outlier detection, for this reason the outlier detection takes in $+0.5H$.

As introduced in Figure 1, the HTL (Holiday Trend Decomposition Loop) works as follows :

8. Extraction of potential Holiday dates from SA_t^k
9. Holiday estimation, H_t^k, through LOESS regression of bandwidth n_h
10. Holiday selection with adjusted Confidence Interval (Appendix A.3.2)
11. Working Day Adjustment $SA_t^k = SA_t^k - H_t^k$
12. Trend Reestimation T_t^k based on SA_t^k with LOESS Regression of bandwidth n_t.
13. Reconstruction of the Irregular : $I_t^k = Y_t - SA_t^k - T_t^k$

Similarly to the STL framework, an outer loop is performed in order to prevent outlier contamination in the seasonal adjustment. The loop consists in extracting robustness weights ω_t from the idiosyncratic component I_t using Tukey's (Tukey (1960)) biweight function :

$$\omega_t = \begin{Bmatrix} \left(1 - \left[\frac{I_t}{h}\right]^2\right)^2 \text{ if } I_t < h \\ 0 \text{ otherwise} \end{Bmatrix}$$

where $h = 6 \times median(I_t)$.

The robustness weights are then used as multipliers for the LOESS regression weights in the STL and HTL inner loops, significantly reducing the weights of outlier observations.

With this framework, the parameters of the procedure are the following:

- n_p the size of the seasonal period
- n_s the bandwidth of the seasonal LOESS
- n_t the bandwidth of the low-pass LOESS for trend extraction
- n_h the bandwidth of the holiday LOESS
- n_o the number of outer loops

4 STAHL Preprocessing Industrial Approach

4.1 Identification of seasonality

Before setting any other parameter for the STAHL procedure, the first parameter to identify is n_p, which is the number of points in the seasonal pattern. For this, in the context of high-frequency data, spectral density analysis, as used, for instance, in T. S. McElroy, Monsell, and Hutchinson (2018), is very useful to visually identifying the peaks in the spectrum of the series, and thus the seasonal frequency to correct. Furthermore, considering that our adjustment procedure

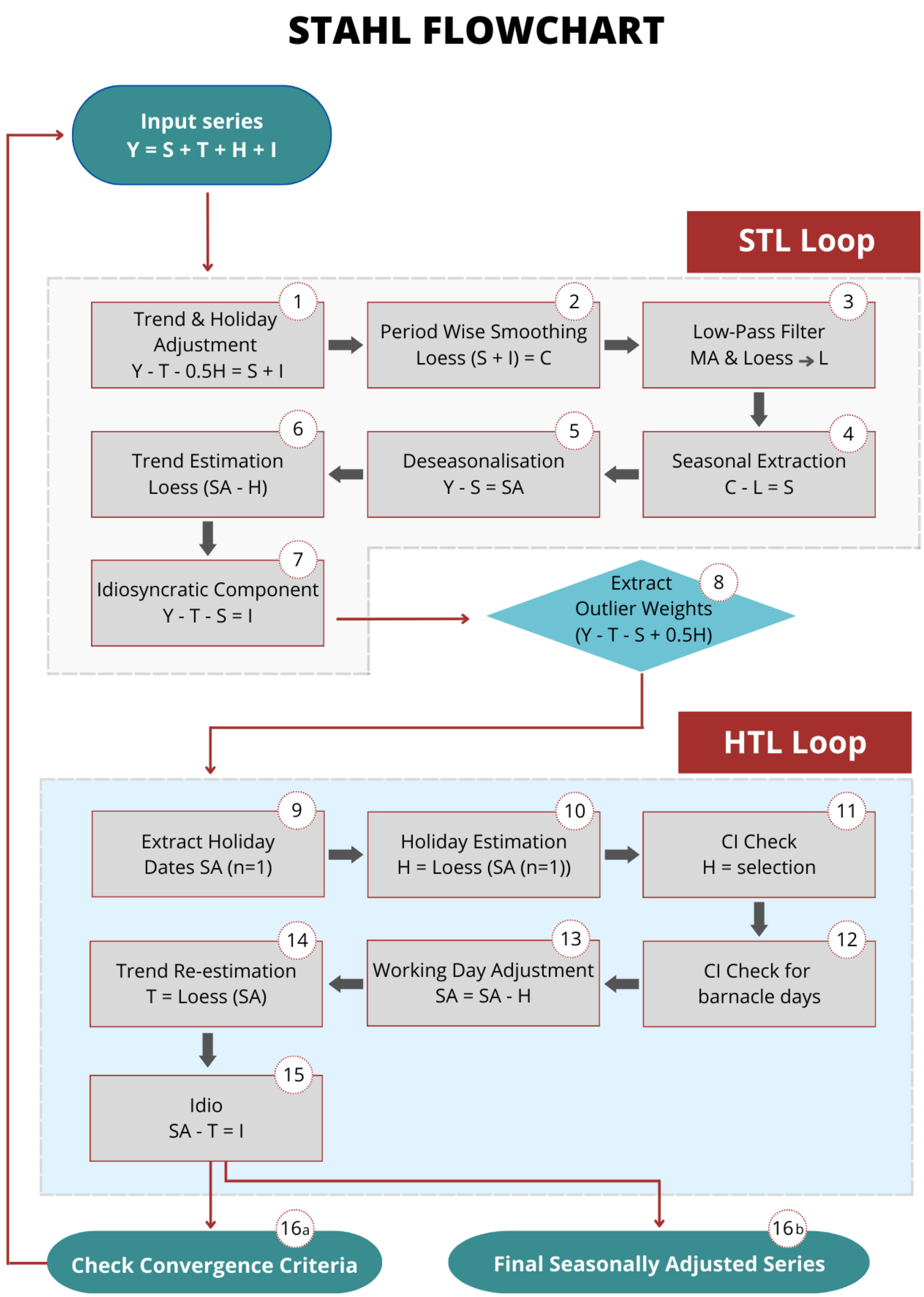

Figure 1: STAHL Framework Decomposition

is intended for use by people who may not be familiar with Fourier analysis, we also include an automatic seasonality detection method in the procedure, as described in Puech et al. (2020) and Li, Wang, and Han (2012).

The methodology consists in creating n random permutations $(\tilde{y}_i)_{i\in[0,n]}$ of a time series y. The permutations will remove any seasonal pattern, and thus the spectral power of y will be spread across all frequencies as noise. By taking the maximum of the spectral power of each permuted series $\tilde{y}_i$ we have a threshold for the maximum power of the noise, which is smaller than the power associated to a seasonal pattern in y. Using $n = 100$ permutations, we take the $99th$ largest value across the series as the threshold above which the original series spectral power indicates seasonality.

In practice, there can be issues of spectral leakage when attempting to correctly identify peaks in the spectral density. However, since there is only a small set of possible seasonal frequencies (yearly, quarterly, monthly, weekly), we simply need to check if the spectral power around those frequencies is above a certain threshold.

4.2 Resampling of data

By nature, the STAHL procedure can only be used for integer periodicities. While this is not a problem for monthly and quarterly data, in practice, the periodicities for weekly and daily data are mostly non-integers. To apply the procedure to high-frequency data, we need to resample our data.

For daily data, we use the methodology described in Ollech (2021). To adjust for seasonality, the length of each month is extended to 31 days, either by time-warping the months over 31 points or by adding extra points at the end of each month using cubic spline interpolation, depending on the nature of the data. For annual seasonality, February 29th is removed to ensure a consistent 365 points per year before adjusting. Once the STAHL procedure has been applied to the resampled data, the resampling procedure is reversed: the added dates are removed, and the removed dates are added back using interpolation.

In the case of weekly data, the issue is slightly different because of the phasing effect of weeks in the year: the year does not always start on a specific day of the week, meaning the underlying annual seasonality is observed with a progressive shift, with 52 or 53 weeks per year. [7]

Therefore, for weekly data, we use a time-warping procedure to ensure that we have 53 equally spaced point per year, by zero-padding the Fourier transform of the series (see Appendix A.1 for more details).

This method of upsampling the data comes with *caveats*:

- With the time-warped axis, the holiday effects become less identifiable due to the upsampling creating a spillover effect. However, since the before and after effects of the holidays are also corrected by the STAHL

[7] Furthermore for economic data, the seasonal pattern usually takes place exactly between January 1st and December 31st and simply be very slightly distorted for leap years, justifying the removal of February 29th. It is not the case for weekly data, where the span of days covered over 52 weeks shifts from year to year.

procedure via the Barnacle Day procedure, the spillover is, in practice, limited.

- The upsampling procedure is not point-in-time, the value of a point is influenced by the future. Therefore to reproduce the pseudo real-time conditions, you need to upsample, seasonally adjust, and then downsample vintage by vintage in an iterative way.

5 STAHL Parameter Estimation

In order to estimate the parameters of the STAHL procedure, our approach provides a method to select them:

- For n_s and n_h, the selection is based on a leave-one-out cross-validation.
- Regarding n_t the default value is the one suggested by R. B. Cleveland et al. (1990) : $n_t = \frac{1.5 \times n_p}{1 - n_s^{-1}}$. However, for some series we need a more rigid low-pass filter to separate the trend from the seasonal and holiday component, with the bandwidth of the filter equal to the size of the burn-in period, *i.e* $n_t = n_p \times n_s$.
- The number of outer loops n_o is determined by convergence criteria on the trend, the seasonal and the holiday components, as suggested in R. B. Cleveland et al. (1990).

6 STAHL Seasonal Adjustment with Missing Values

The treatment of periodic or structural missing values (Northern Regions of China Air pollution Data impacted by clouds every winter) is one innovative advantage of our procedure: every instance of a season can have missing-value, the normal STL cannot handle such structural or periodic missing values.

The STL methodology excels at dealing with a few, randomly occurring missing values. As the smoothed subseries is constructed, the LOESS regression can easily fill in singular missing values. When the subseries are then combined in the low-pass filter, there are no more missing values. However, this process breaks down when the missing values are structural. By this, we mean that a specific subseries consists entirely of missing values. If this occurs, the moving average in the low-pass filter will encounter issues.

We address this problem by creating a missing-value-robust moving averages procedure. Normally, if the moving average is of length 15, you would add the last 15 values and divide by 15. However, if there is a missing value, it is replaced by zero. Then, the last 15 values (including the zero) are added and divided by 14. If the point at which the value is being estimated is missing, the output value will also be missing. Finally, to ensure that there are no extreme edge effects, each point is assigned a weight. The weight is 1 if there is no missing value in the moving average, for the previous example the weight of that point would be $\frac{14}{15}$. These weights are multiplied with the outlier weights before the LOESS part of the low-pass filter.

7 Holiday Adjustment within STAHL

The Holiday procedure is one of the main contributions of the STAHL methodology. It allows for the removal of the effects of holidays that are non-seasonal by nature. For instance, Christmas always occurs at the same point in time in each seasonal cycle. The 25th of December is always the 360th day of the year. As such, when yearly seasonal adjustment is applied to a daily series, Christmas will have its own subseries. However, some other holidays, such as Easter, Ramadan, or the Chinese New Year, do not occur on the same day each year in the Gregorian Calendar cycle. This means that they will not automatically be addressed during a regular STL procedure. STAHL offers an innovative procedure to apply the same logic used in STL for removing these specific time-varying holiday effects.

Several steps are required to perform a successful holiday extraction. The first step is to remove any seasonal holidays from the list of holidays, as the STL procedure can handle these by itself, as mentioned earlier. The next step is to test the statistical significance of holiday dates. Some holidays might not have any effect, and adding them can introduce unnecessary complexity to the procedure. Including holidays that are statistically insignificant can adversely affect the seasonal extraction.

While some holidays, such as Ramadan, last several days, others, like Easter, do not. However, this does not mean that their effect is limited only to the 'official' holiday dates. To address this spillover issue, we developed the Barnacle procedure [8]. The Barnacle procedure allows us to handle days that are significantly affected by their proximity to a major holiday. We conduct the Barnacle procedure by checking the days adjacent to the holiday for significance. If these days are significant, the subsequent day is checked, and so on, up to a hard limit of 46 days before and after each holiday. This approach differs from the impact models (constant impact, linear ramp-up and down), used, for example, in Ladiray, Palate, et al. (2018), as here the shape of the impact is more flexible.

8 Point-in-Time Seasonal Trend Decomposition with STAHL

At QuantCube, all series are produced and delivered on a daily basis and cannot be revised. To ensure that the structure of each series remains consistent, the historical parts of each series must be calculated in pseudo-real-time. This means that each series is only allowed to use historical data for a single estimation. Moreover, we do not want past values to change when further data is added to the series.

In the classical STL procedure, there are three different stages where forward-looking bias can occur. The first and most obvious is during any step of the LOESS regressions. The second occurs during the low-pass filtering stage, when

[8] The barnacle is a crustacean that attaches itself to other animals or the ocean substrate. We named the procedure after it because Barnacle days are significant due to their 'attachment' to the holiday events.

a moving average is applied in both directions. Finally, during the outlier detection, the median value that is normally calculated using the entire dataset can introduce bias. To industrialize and create replicable series, these issues need to be addressed.

8.1 Asymmetric LOESS

As a kernel regression, the LOESS applies a weight to the points on each side next to the estimation point. It also applies a non-zero weight to the points in the future. We get around this problem by applying an asymmetric kernel-regression.

The LOESS regression differs from a more classical kernel regression in the way the kernel width is set. In a classical kernel regression the bandwidth is set to a fixed width and the number of points used to make an estimate changes depending on how many data points are present around the estimation point. In the LOESS regression the number of points are fixed and the width of the kernel is set so that the same number of points are always present.

Starting with a kernel width of zero, in regular symmetric LOESS the kernel width is slowly expanded in either direction until the required number of data points is in the bandwidth. In the case of an asymmetric kernel, only the left hand side bandwidth is expanded while the right hand side of the estimation point has a bandwidth of zero.

In the edge-case, at the beginning or end of the dataset, the bandwidth can only expand in one direction. Therefore both the symmetric and asymmetric LOESS have identical kernels at the start and at the end. At the start this means that there is a slight phase shift comparing the kernels at the end in Figure 2 makes it clear how the asymmetric kernel replicates the point in time estimation of a symmetric LOESS.

The LOESS asymmetric filter may not be the most optimal filter in terms of gain and phase shift. Grun-Rehomme, Guggemos, and Ladiray (2018) provide a comprehensive survey of optimal asymmetric filters and propose an approach that balances *the moving average properties in terms of accuracy, revisions and timeliness and allows constructing asymmetric moving averages that present almost no phase-shift*. Their work also suggests exploring the non-parametric approach based on spectral theory, as promoted by Wildi (2007), Wildi and T. McElroy (2016), Wildi and T. S. McElroy (2019) with the Dynamic Asymmetric Filters family. This procedure, designed in the frequency domain, would allow the splitting of the revision criterion into two distinct effects, one related to the gain and the other to the phase of the transfer functions. We leave the exploration of introducing alternative asymmetric filters into STAHL for further research.

8.2 Extrapolating Moving Averages Procedure

In the STL procedure, the smoothed subseries for the days is extended to be two values longer than the original subseries. The outermost values are determined using linear extrapolation. This ensures that when the bidirectional moving

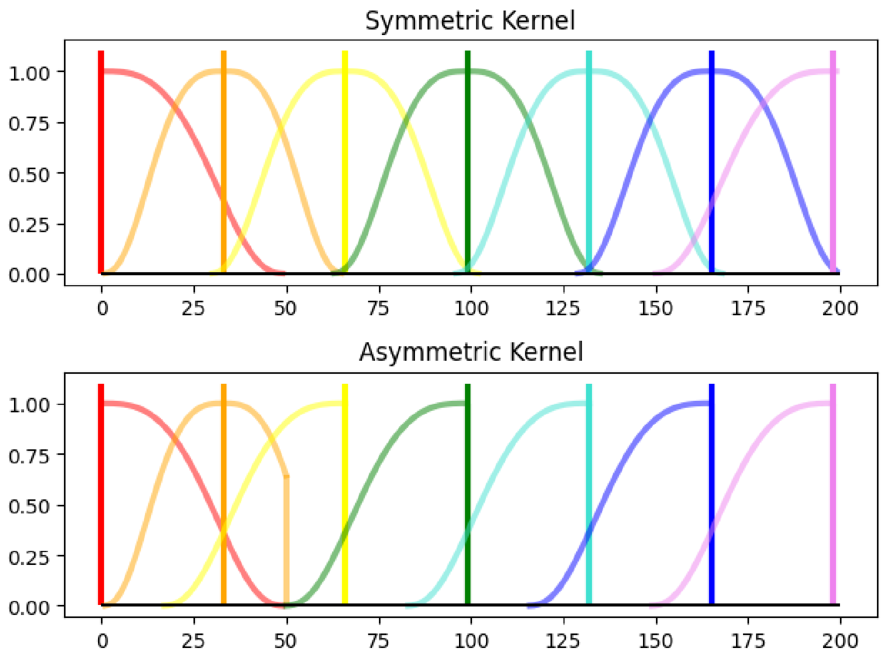

Figure 2: Kernel Comparison

averages procedure is applied, the final series has a length equal to that of the original series. However, an issue arises in that the reverse moving average (MA) procedure uses values from the future. To circumvent this issue, instead of using the smoothed subseries values directly, at each point in time, the future values are linearly extrapolated. Then, the moving average procedure is applied to these extrapolated values.

8.3 Outlier Weighting

During the outlier weighting, all values are divided by six times the median value of the series. If the entire series is taken into account, this can be considered forward-looking behavior. To prevent this, we set a validation date, so that only the median of the series up to the validation date is considered. This approach ensures that the values of the series do not change when more values are added in the future.

8.4 Critical Burn-In period

The critical burn-in period corresponds to the time span at the beginning of the estimation, during which we are not yet able to produce data in pseudo-real time. Each of the three sources of forward-looking bias has a different burn-in period and affects the structure of the data-generating process in a unique way.

During the early stages of the asymmetric LOESS the kernel start of with a right-tail, then it becomes symmetric before finally reaching its desired left-tailed shape. The LOESS is used in the subseries, the low-pass and the trend.

The final LOESS to reach the left tailed shape is the LOESS subseries. This occurs at the point $n_s \times n_p$. Before this burn-in is completed the data will have significant forward-looking, and as the weighted center of the kernel if further forward it will also have a slight phase shift relative to the data past the burn in.

The extrapolating moving average needs at least two values before it can start extrapolating. As it extrapolated from the same point in each cycle it requires $2 \times n_p$ points before it completes its burn-in. The change in the seasonal extraction process comes from the fact that the extrapolated data tends to be slightly more volatile than the original values. As they are averaged out again so this does not significantly impact the quality of the estimation.

Finally, the validation date can be set freely and does not significantly impact the seasonal extraction process. It therefore makes sense to set it to be at least $n_s \times n_p$, but it does not cause significant forward looking issues if it is set further into the future. Once a series goes into production this should not be changed anymore in order to ensure a stable history.

8.5 Multiple Seasonality

The STAHL procedure handles the seasonality in an iterative manner, similar to Ollech (2021), and MSTL in Bandara, Hyndman, and Bergmeir (2021), starting by removing the seasonality of highest frequency first and then moving on to lower frequencies.

9 Empirical Illustrations

9.1 Traditional Weekly Data: US Initial Unemployment Insurance Claims

9.1.1 Data Description

As a first example, we applied the STAHL procedure to the US Initial Unemployment Claims series. This weekly series, published by the US Employment and Training Administration, covers the number of initial unemployment claims made by unemployed US individuals to be eligible for employment benefits. It is published both as a seasonally adjusted and a non-seasonally adjusted series every Wednesday. This series is relevant to illustrate our seasonal adjustment procedure for the following reasons:

- Its weekly frequency allows us to illustrate the resampling procedure.
- Available since 1968, its long history helps to illustrate the influence of the business cycle on the trend. We also have access to every data vintage from 2009 onward, which allows us to gauge the impact of revisions in both the SA and NSA data.
- It exhibits outliers that can be due to catastrophic external events (hurricanes, the Covid pandemic outbreak) or labour-related events (strikes).

- It is a relevant series for tracking the macroeconomic situation and the business cycle, used, for instance, in high-frequency composite indicators such as the Aruoba-Diebold-Scotti Business Conditions Index[9].

9.1.2 Official Seasonal Adjustment Methodology

The official initial claims seasonal adjustment[10] is performed by the Bureau of Labor Statistics (BLS) and has relied on the methodology described in W. P. Cleveland, Evans, and Scott (2014) since 2002. The series exhibits yearly multiplicative and the seasonal factors are computed using a locally weighted regression on sine and cosine terms, with the weights extracted from a state-space modeling of the series. There is no trend extraction *per se*, as the series are detrended by differentiating before the seasonal adjustment. The holiday procedure is based on the X-13-ARIMA-SEATS program, with holiday weights constant over time. The outliers and intervention adjustment also uses the X-13-ARIMA-SEATS program.

The official data both SA and NSA is subject to revisions with each publications. The main source of changes is the revisions of the seasonal factors, as showed in Figure 3. We focus in this illustration on the claims values for the first publication and the last available vintage (end of October 2023).

The Covid-19 pandemic has also led to further changes in the methodology. Due to its extreme impact on unemployment claims, the BLS changed the seasonal factors from multiplicative to additive for the period from March 2020 to June 2021, then switched back to multiplicative after that date. The extreme nature of the event and its duration also necessitated an *a posteriori* update (in April 2023) of the outliers' specification during the pandemic period, leading to major revisions of the seasonal factors from the end of 2021.

9.1.3 STAHL set-up

We will now apply the STAHL procedure to produce our own version of the seasonally adjusted claims data. In doing so, we are operating under real-time conditions, meaning that there are no revisions of the data, and we use real-time data vintages from May 2009 to November 2023.

First, following the official procedure, we assume a multiplicative seasonality scheme, meaning a log transformation is applied. Given the weekly frequency of the data, we apply the resampling procedures described in section 4.2 to have 53 equally spaced datapoints per year. The resampled holidays given to the HTL procedure are the following: New Year's Day, Washington's Birthday, Memorial Day, Independence Day, Labor Day, Columbus Day, Veteran's Day,

[9] The Aruoba-Diebold-Scotti Business Conditions Index, published by the Federal Reserve of Philadelphia, is designed to track real business conditions at a high observation frequency. Its underlying economic indicators, which are seasonally adjusted, include weekly initial jobless claims, monthly payroll employment, monthly industrial production, monthly real personal income less transfer payments, monthly real manufacturing and trade sales, and quarterly real GDP. These blend high-frequency and low-frequency data.

[10] `https://www.bls.gov/lau/seasonal-adjustment-for-weekly-unemployment-insurance-claims.htm`

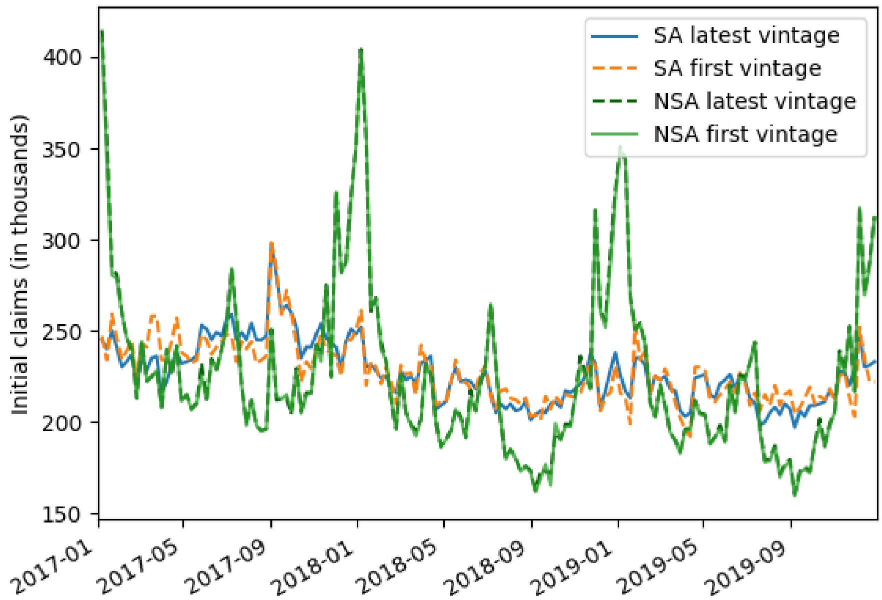

Figure 3: Revisions of the initial claims data, first to last vintage (October 2023)

Thanksgiving, Christmas and Easter. The Barnacle procedure detects Barnacle days effect before and after Thanksgiving and Independence day.

During the Covid period, we do not make the switch to additive, and keep the multiplicative scheme. However, even with the built-in robustness to outlier of the STAHL procedure, the unprecedented extreme behavior of the series during that period contaminates the seasonal adjustment for the years 2022 and 2023. Therefore on that front we follow the BLS choice, and remove the period from March 2020 to June 2021 from the data when performing STAHL starting January 2022.

9.1.4 Point in Time Seasonal Adjustment with STAHL

We observe in Figure 4 that the seasonal adjustment obtained via STAHL follows closely the official seasonally adjusted signal, both in terms of trend and idiosyncratic behaviour. The procedure is robust to outliers. For instance, the amplitude of the peak in August 2017 due to hurricane Harvey matches the one measured in official data. However, the volatility around the Thanksgiving period is higher for our model than for the official data, a consequence of the upsampling.

Regarding the Covid period, as observed in Figure 5, STAHL is sufficiently robust to outliers, such that maintaining a multiplicative scheme yields results similar to those of the BLS methodology for the period from April 2020 to December 2021. However, it is noteworthy that STAHL output diverges from the official data briefly in January 2021, before recovering. For the post-Covid period, the real-time output of STAHL is less volatile than the first estimates

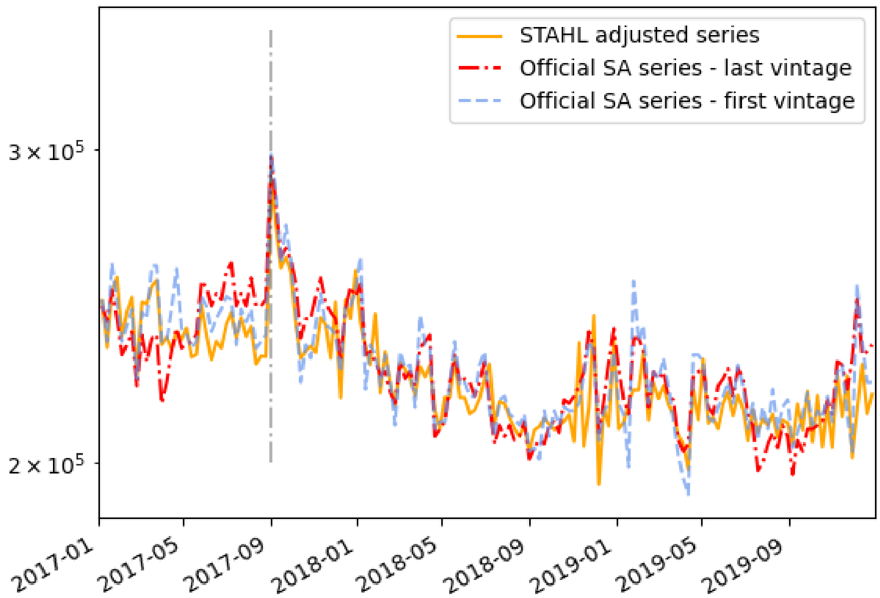

Figure 4: Comparison of STAHL vs first and last vintage SA (October 2023)

and closer to the latest vintage. This illustrates both the stability of our model over time and the benefit of hindsight gained from excluding the Covid period starting in 2022.

A visual analysis of the spectral density for the non-seasonally adjusted original series and the STAHL seasonally adjusted series (see Figure 29 in Appendix A.4) shows that the procedure removes all the seasonal peaks from the spectrum while preserving the lower frequency components.

The metrics shown in Table 1 confirm that, in the post-Covid period, the STAHL model is closer to the latest estimates than to the first vintage in terms of Mean Absolute Log Error. In the pre-Covid period, the Mean Log Error of STAHL compared to both the first and last vintages is close to 0.02, which is significant compared to the revised MLE. This suggests that the STAHL output tends to underestimate the underlying trend compared to the official data. This discrepancy could be attributed to differences in the methods of trend separation. This underestimation also results in a Median Absolute Log Error of around 0.03. This issue could potentially be addressed by forcing the seasonal factor over each year to sum to zero.

	vs 1st estimate		**vs last estimate**		**Official revisions**	
Metrics	**MLE**	**MALE**	**MLE**	**MALE**	**MLE**	**MALE**
2009-2019	0.016	0.026	0.022	0.028	0.003	0.015
2020-2021	0.006	0.05	-0.025	0.061	-0.009	0.04
2022-2023	0.03	0.065	0.008	0.045	0.0	0.054

Table 1: Median Log Error and Median Absolute Log Error of STAHL compared to official data

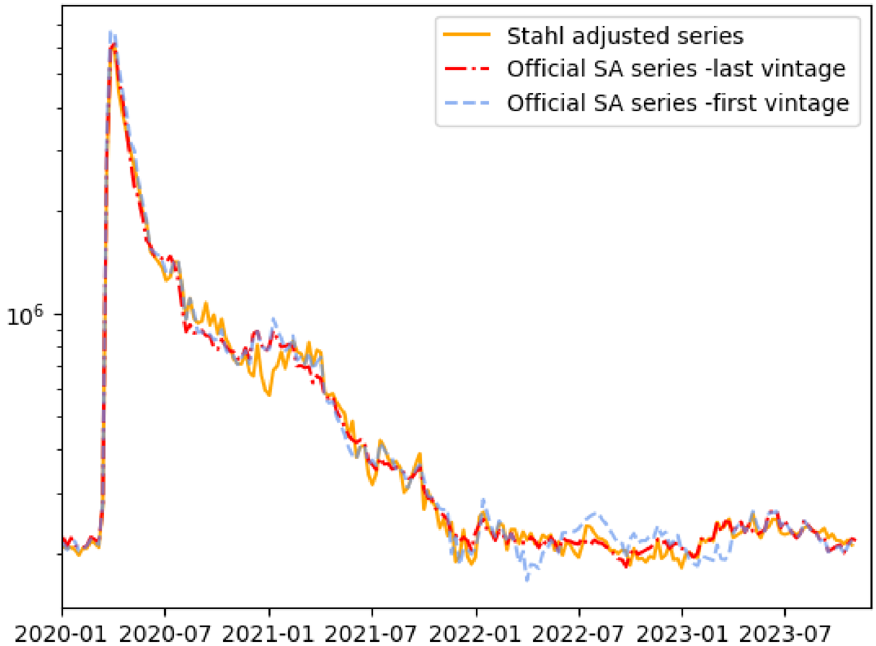

Figure 5: Comparison of STAHL vs first and last vintage SA - Covid and post-Covid period

Through this illustration, we demonstrate the capacity of STAHL to reproduce the seasonal adjustment of the official methodology for high-frequency data using a robust, more parsimonious, and purely point-in-time method.

9.2 Daily Electricity Data featuring Multiple Seasonality

9.2.1 Data Description

In this section, we analyze an empirical application of the STL procedure with multiple seasonality. We seasonally adjust electricity consumption data in MWh from the US power grid, published by the Energy Information Agency [11], with a history from mid-2015 to the beginning of 2023. The data is daily and exhibits both a weekly and a yearly seasonal pattern. While electricity data contains useful economic information about both consumption and production, its low signal-to-seasonality ratio can make it tricky to work with. This series appears to be a perfect candidate to demonstrate the effectiveness of the STL procedure. Figure 6 shows the raw, non-seasonally adjusted electricity data.

When dealing with multiple seasonalities, the proper procedure is to seasonally adjust the shorter cycle first and then move on to the longer cycle. Since we are dealing with weekly (7) and yearly (365) seasonality, we start of by removing the weekly seasonality.

[11] `https://www.eia.gov/electricity/gridmonitor/dashboard/electric_overview/US48/US48`

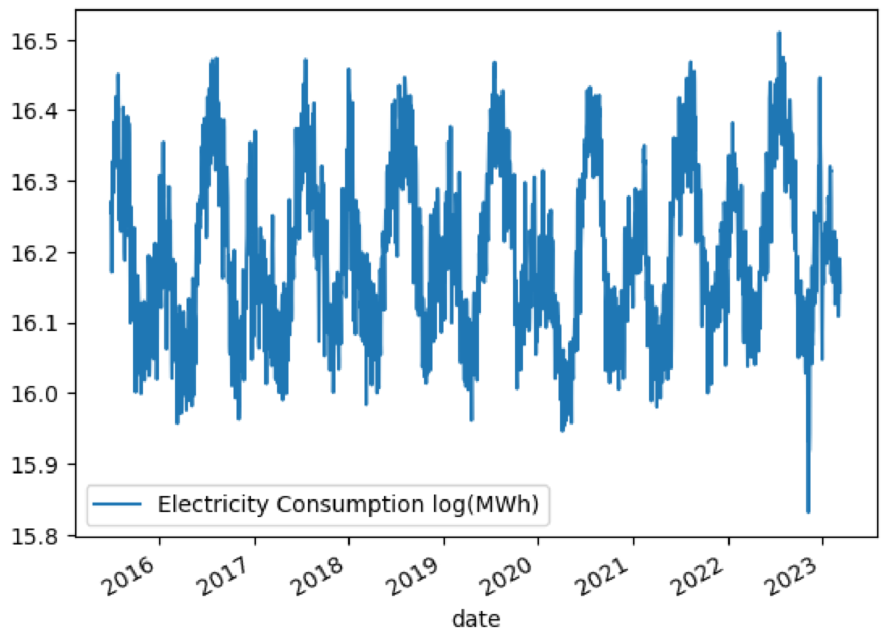

Figure 6: Electricity Consumption

9.2.2 Weekly Seasonal Adjustment

The first step of the STL procedure is to create subseries. Since we have 7 days in our weekly seasonal cycle, we create 7 subseries, one for each day of the week. Each subseries consists of only Mondays, Tuesdays, and so forth. An example of the subseries for Monday and Sunday can be seen in Figure 7. Here, we can clearly see that the electricity consumption is, on average, lower on Sunday than on Monday. To capture this effect, a LOESS regression is fitted to the data. This creates the smoothed subseries, which is conceptually similar to a seasonal dummy for days like Monday or Sunday. The major advantage over a simple seasonal dummy is that it is much more flexible and allows for changing seasonal dynamics.

Figure 8 exhibits the first year of the data with the weekly seasonality removed. The series looks much smoother and is now missing the weekly oscillations. However, a lot of yearly seasonality is still visible, as there is a spike during summer related to air conditioning and a second smaller spike in winter for heating demand.

9.2.3 Yearly Seasonal Adjustment

Following the same procedure , to get the week-yearly seasonally adjusted series, a second STL procedure is subsequently applied. Figure 9 exemplifies how the different components of the series play out. It illustrates how much variation in the series comes from the yearly seasonal component any why it is so important to seasonally adjust such series before attempting to use them for econometric procedures.

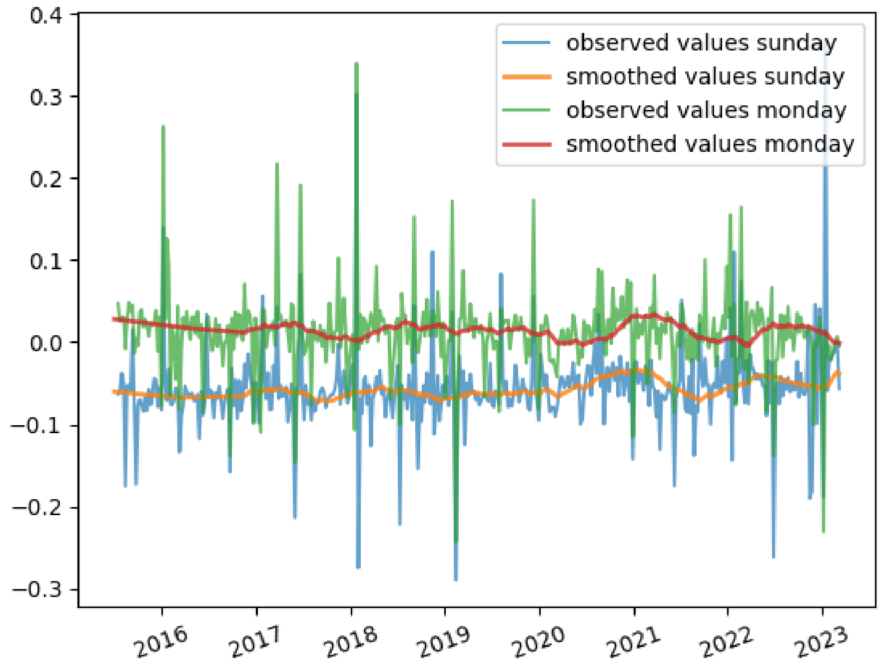

Figure 7: Monday and Sunday Electricity Subseries

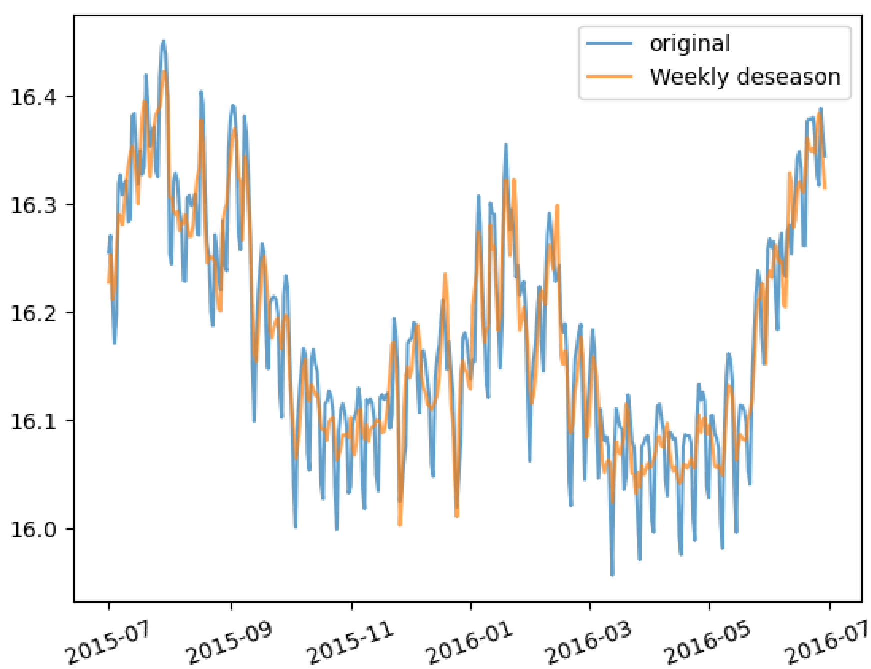

Figure 8: Electricity Consumption Weekly Seasonal Adjustment

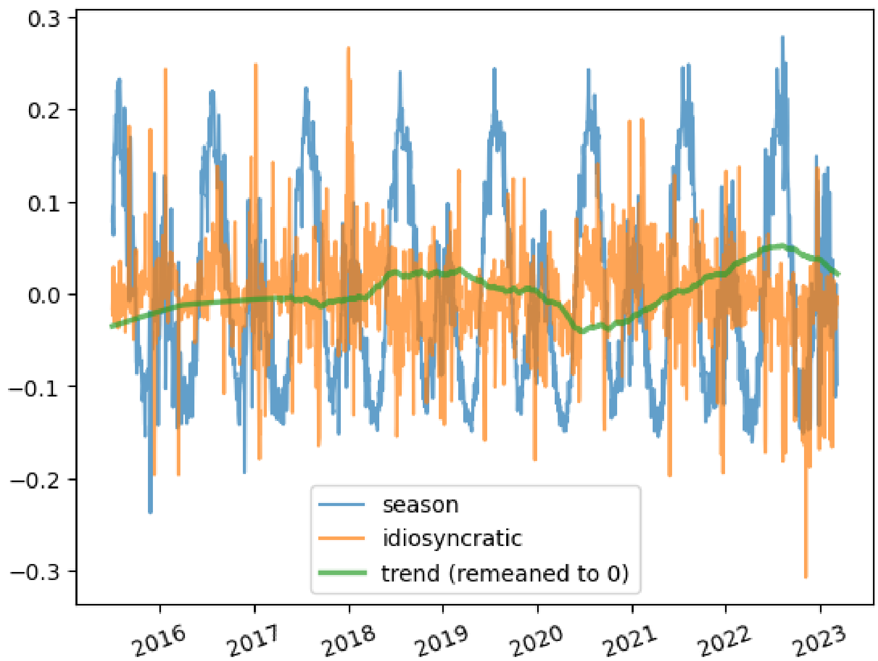

Figure 9: Electricity Consumption Components

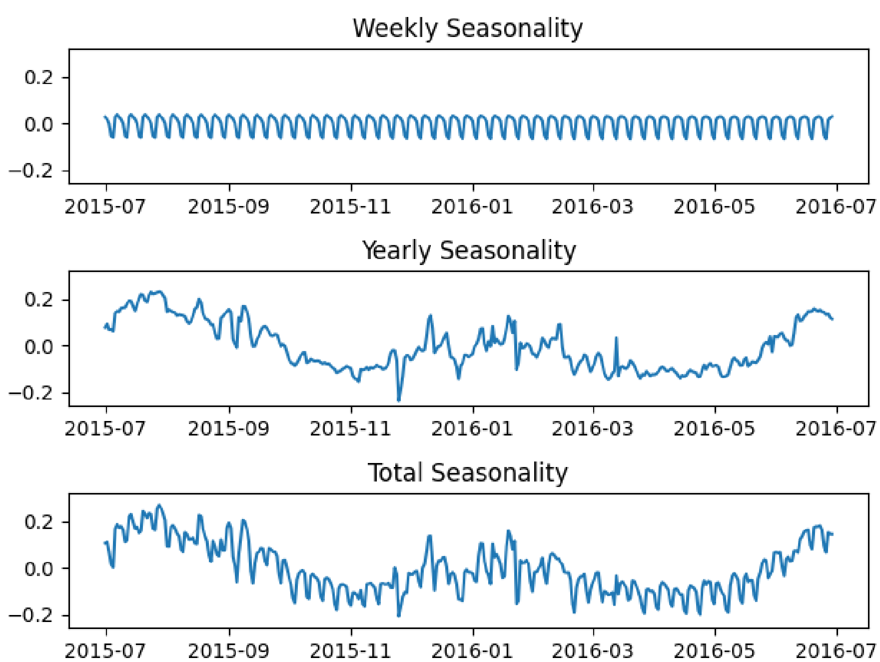

Figure 10: Electricity Consumption Seasonal Patterns

9.2.4 Seasonally Adjusted Series

In Figure 10, the weekly and yearly seasonal components, as well as the total seasonality extracted, are provided. Figure 11 demonstrates the difference between the original series and the fully seasonally-adjusted series.

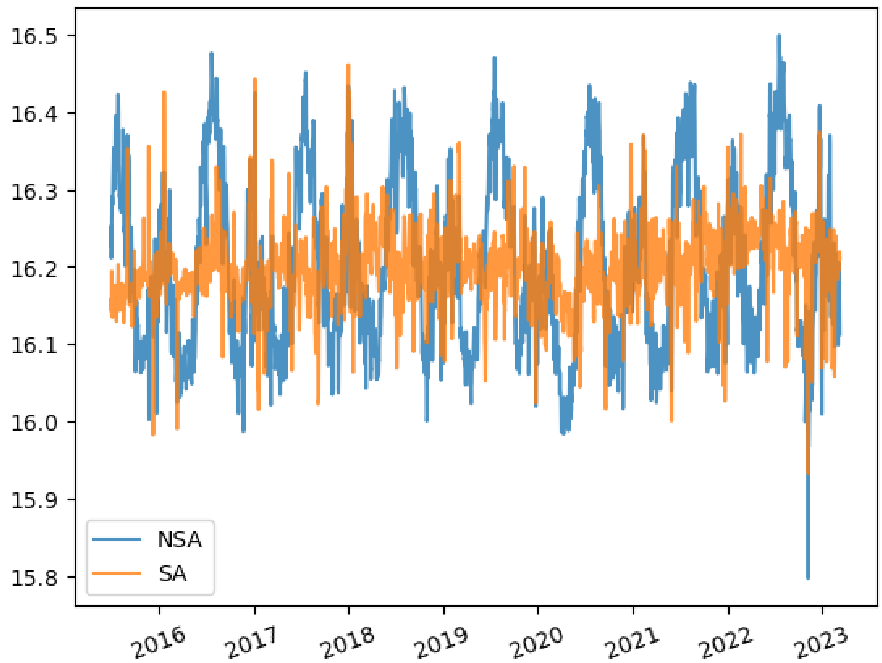

Figure 11: Electricity Consumption Seasonally Adjusted

9.3 Holiday Adjustment: Google Searches for "Easter Bunny"

9.3.1 Data Description

Easter is a tricky holiday to capture in a seasonal adjustment procedure, as it does not have a fixed place in the Gregorian calendar. To illustrate its effect, we chose the Google Search Volume series for the keywords "Easter Bunny". While most series have both seasonal and holiday effects, this series has been specifically selected because all movements can clearly and easily be attributed solely to the holiday effect. It therefore serves as an excellent test bed to demonstrate the issues that can arise when trying to adjust for holidays using the standard STL procedure, which is conceptualized for regular seasonal effects. The Google Search Volume series starts in 2010 and goes all the way to August of 2023, comprising 4981 daily values. They range between 0 and 100, with the values representing the relative search volume for the period.

9.3.2 Applying Classical STL

When we look at the first 500 days of the Easter Bunny Google searches in Figure 12, two of the Easter events can be easily isolated. The uptick in searches starts quite some time before Easter and then quickly drops once Easter is over. Examining the same time period after having seasonally adjusted the series using the classical STL approach (Figure 13), the results appear rather disappointing. For each of the two Easter events, STL assigns some negative values: for the first day before Easter, and for the second day after Easter. Clearly, the fact that Easter moves around confuses the STL procedure, which tries to find some kind of 'average' Easter effect over a longer time period.

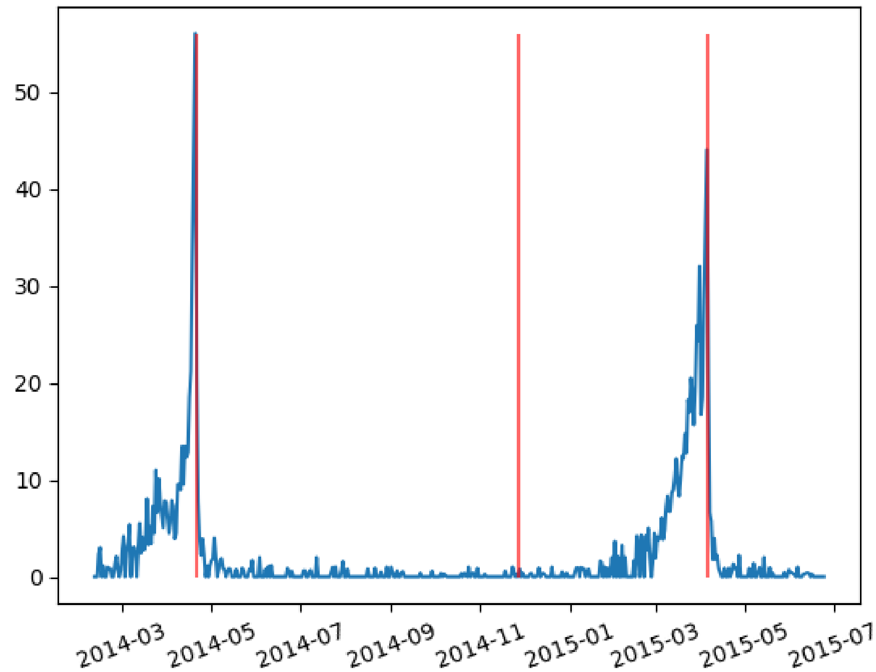

Figure 12: Easter Bunny searches

9.3.3 Applying HTL

As it is obvious that the classical STL procedure is not well-suited to handle moving holiday effects, we developed the HTL procedure. Conceptually, HTL works in a very similar way to the STL procedure. The main difference lies in the fact that the subseries consist only of the holiday events, rather than having a subseries for every period in a seasonal cycle.

Figure 14 shows the estimated holiday effect for Easter alongside the original series. While Easter is deemed significant (and thus an 'Easter effect' is estimated), the results are not very impressive. The estimated Easter effect appears much smaller than the actual effect and is confined to only one day per Easter event. To correct for this, we developed the Barnacle procedure (see appendix). In the Barnacle procedure, we begin by testing the statistical significance of the holiday effect. If the holiday is found to be significant, we repeat the process with the days on either side of the holiday, i.e., the Barnacle days. We continue this process until we reach a day that is not significant or we reach the 46-day limit.

In Figure 15 We can see that once we include the Barnacle day effects our estimations are much closer to the original series than before. Figure 16

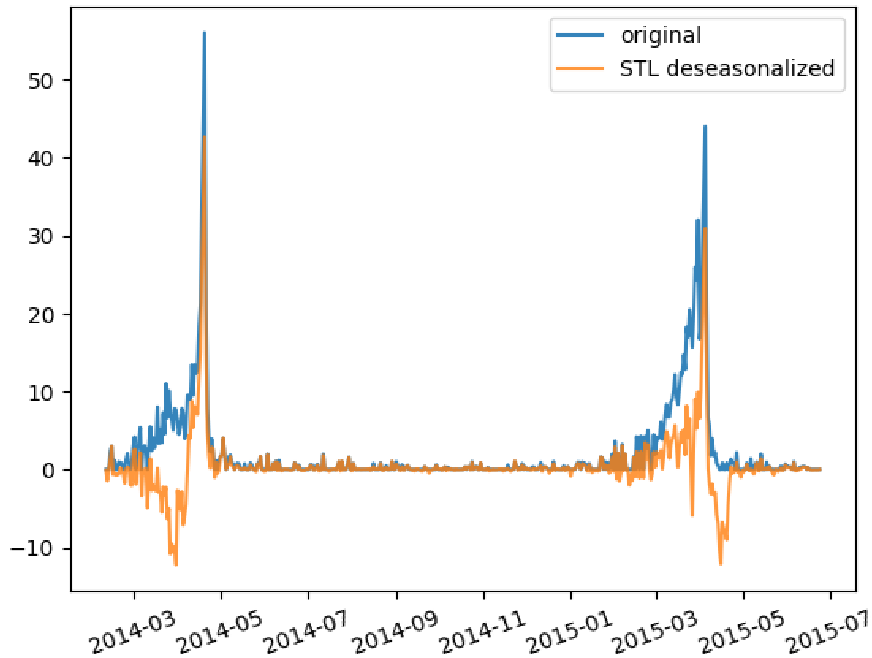

Figure 13: STL seasonal adjustment

illustrates the dramatic improvement that comes from switching from STL to HTL with the Barnacle procedure. As such, we are able to reduce both the positive and the negative extremes of the series.

9.3.4 STAHL Application of Easter Bunny Searches

This series of searches for the Easter Bunny is simpler than many other series that QuantCube uses daily. The main difference is that this series exhibits purely holiday effects, whereas most series have a mix of holiday and seasonal effects. We will examine this in more detail in the next section. Figure 17 demonstrates the seasonally adjusted series using HTL and STAHL. We can observe that the STAHL seasonal adjustment slightly struggles with the negative effects, similar to what we observed in the STL procedure. However, it is also evident that STAHL can handle the dates close to the Easter event quite effectively. The efficiency of the HTL adjustment is visually confirmed by the spectral density graph (see Appendix A.4, Figure 31), where we observe that the HTL procedure removes the quick succession of peaks present in the density of the non-seasonally adjusted (NSA) data. These peaks correspond to the Fourier transform of a Dirac comb-like holiday pattern[12].

[12] A Dirac comb is a pattern of regularly spaced peaks in the time domain.

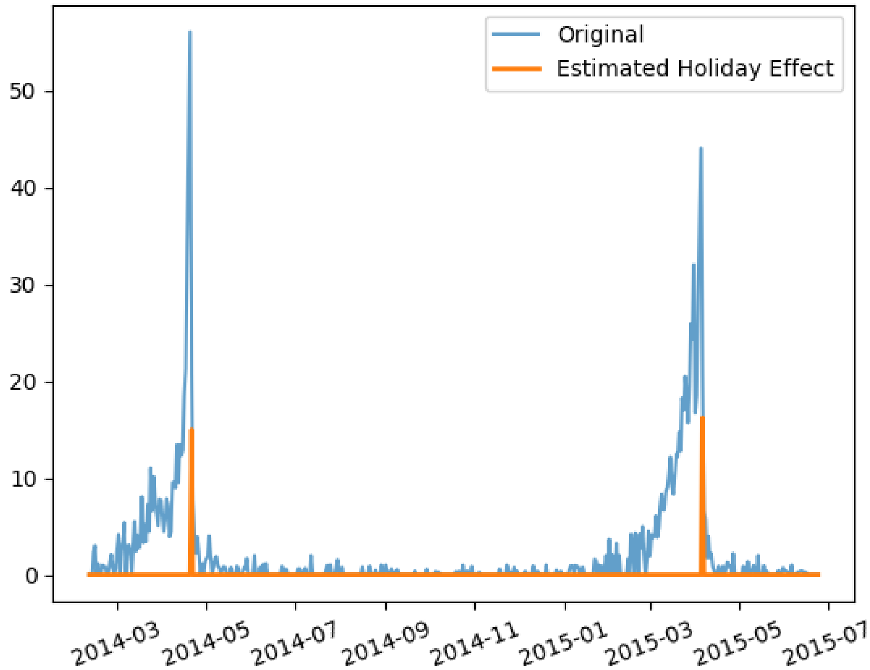

Figure 14: Easter Bunny searches and estimated Holiday Effect

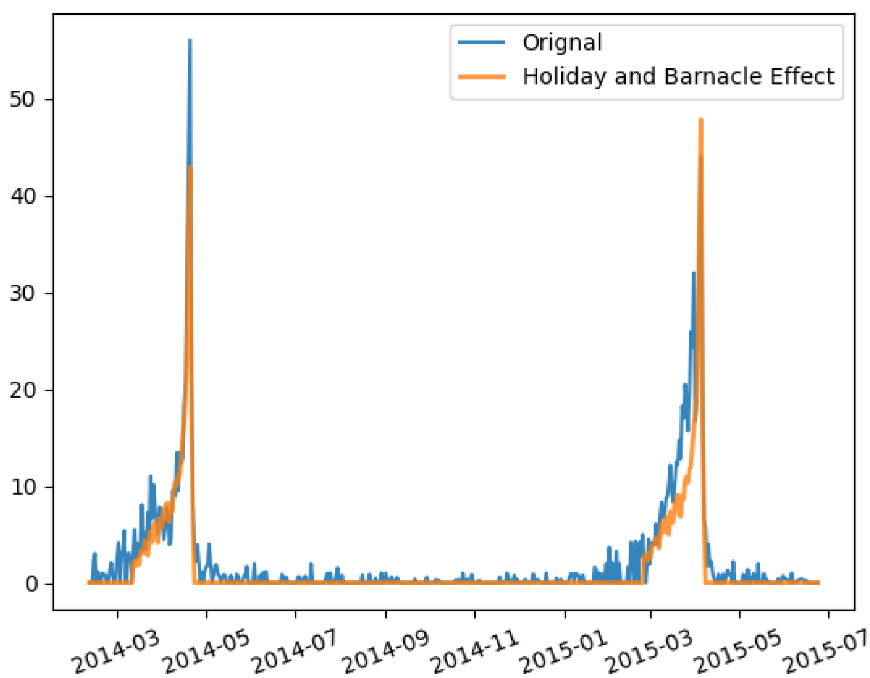

Figure 15: Easter Bunny searches and estimated Holiday + Barnacle Effect

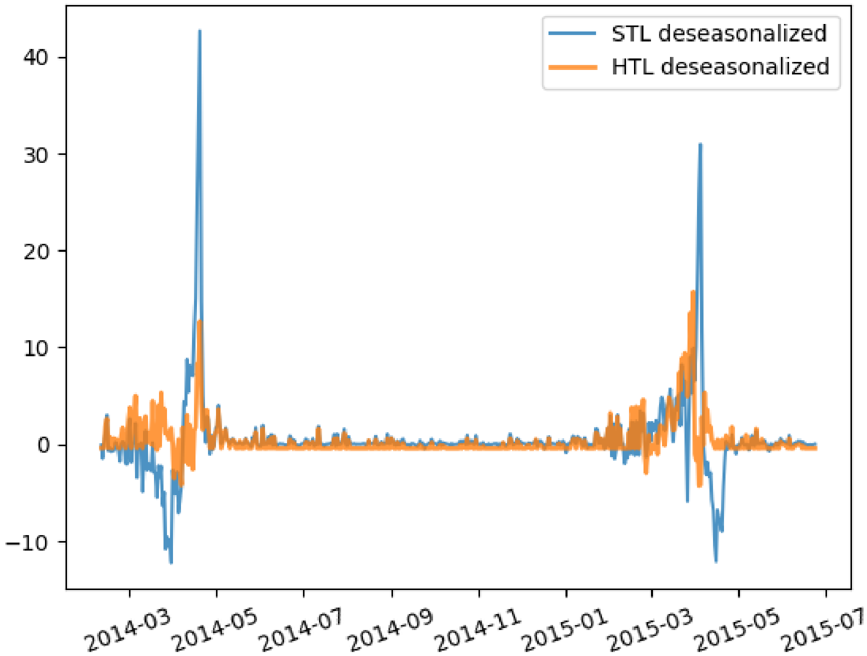

Figure 16: Easter Bunny searches STL vs HTL

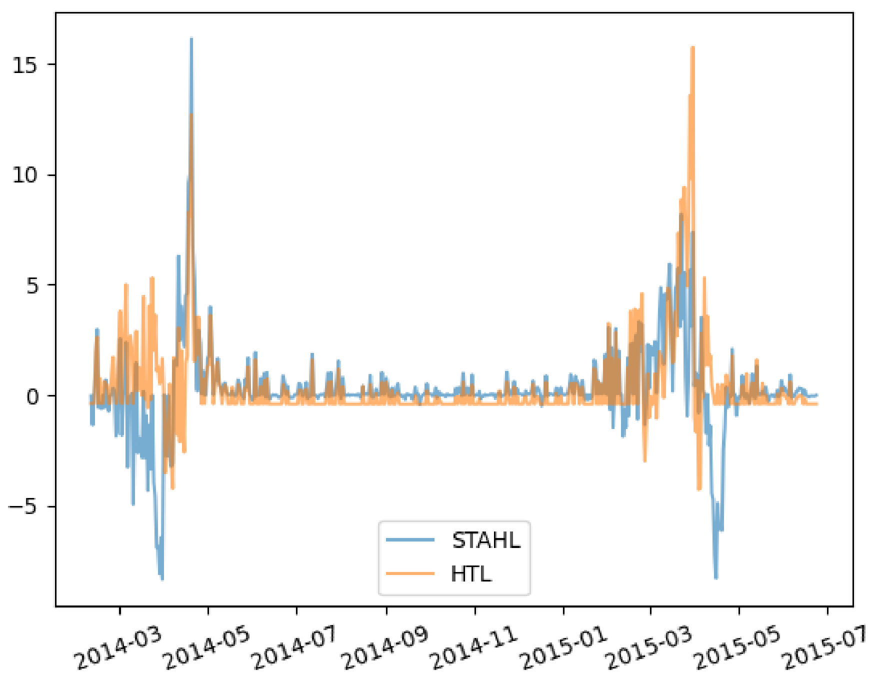

Figure 17: Easter Bunny searches HTL vs STAHL

9.4 STAHL Application on Alternative Data

9.4.1 Data Description

This section will cover the application of STAHL to alternative data series as it is implemented at QuantCube. To illustrate the empirical uses of STAHL, we present two different series. The first is a series on the daily trucking mileage on German highways, and the second is air pollution (NO_2) levels over the city of Dongguan in China. Both of these series exhibit very high levels of seasonality and significant moving holiday effects. In Figure 18, we can see the raw series, both of which clearly show yearly seasonality.

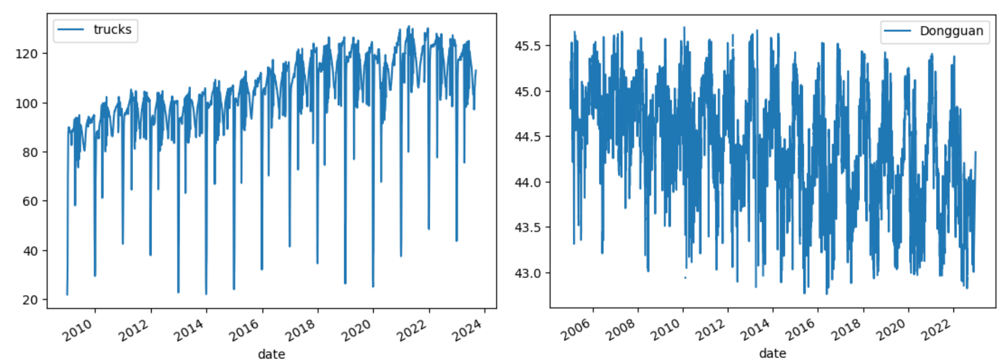

Figure 18: Raw Series German Trucking and Chinese Pollution

Dongguan is an important industrial hub, particularly for manufacturing and export, in the south of China. The pollution series for this city is captured by the Ozone Monitoring Instrument (OMI) aboard NASA's Aura satellite, which was launched in 2004. The series starts on January 1st, 2005, and goes until December 31st, 2022, providing a total of 6503 data points.

The German daily trucking mileage data are available online on the Destatis website[13]. In Germany, there is a general driving ban for vehicles over 7.5 tonnes in weight on public holidays. This means that, in addition to the seasonal effects of the series, they also exhibit significant holiday effects. Ignoring these holiday patterns can have very detrimental effects when trying to interpret the series.

9.4.2 Holiday Effects

The holiday adjustment of the Chinese series handles only the Chinese New Year. Chinese New Year is an important family holiday in China, during which many workers take time off to visit their families. As a result, industrial production, particularly in human capital-intensive industries, drops during this period. This, in turn, *should* lead to a predictable decrease in pollution during the Chinese New Year event. Since the Chinese New Year marks the start of a new year in the Chinese lunisolar calendar, it does not align perfectly with any specific day in the Gregorian calendar. This makes it a moving holiday from the perspective of the Gregorian calendar.

[13]see `https://www.destatis.de/EN/Service/EXSTAT/Datensaetze/truck-toll-mileage.html`

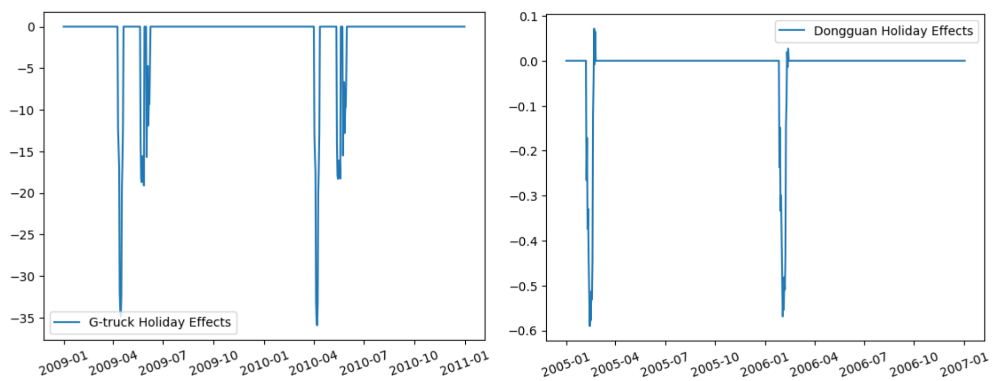

Figure 19: Holiday Effects German Trucking and Chinese Pollution

For the German holiday adjustment, there are three moving holidays that need to be taken into account: Easter, Ascension Day, and Whit Monday. These are all public holidays in Germany and, as such, do not permit heavy vehicles to be driven on these days. For both series, these holidays lead to significant drops in activity.

Figure 19 shows the holiday effects of both the series for the first two years. The Barnacle effects for Chinese New Year go from -2 to $+14$ days. The Barnacle days for Easter, Ascension day and Whit Monday are $(-3, +6)$, $(0, +6)$ and $(0, +6)$ respectively.

9.4.3 Seasonally Adjusted Series

Figures 20 and 21 exhibit the first two years of the series after seasonal adjustment, both with and without the holiday effect removed. The missing holiday effect in the German trucking series is much more apparent than in the Chinese pollution series. However, both effects are significant in the Barnacle procedure, indicating that they are clearly predictable. Just as with predictable seasonal patterns, predictable holiday effects are not informative for nowcasting economic time series and should thus be expunged from the series.

We can also examine the calendar day variance. While the effect is smaller in the Chinese pollution series than in the German trucking series, we can clearly see a drop in the variance of the series during periods of potential holiday events in Figure 22, and to a lesser extent in Figure 23 in the Dongguan pollution series as well. In the German series, we can also observe that even regular holidays (such as Christmas) exhibit higher volatility than 'regular' periods.

Looking at the spectral density of the Dongguan series, the STAHL procedure removes the peak corresponding to the yearly periodicity while preserving the shape of the spectrum for other frequencies (see Figure 30 in Appendix A.4).

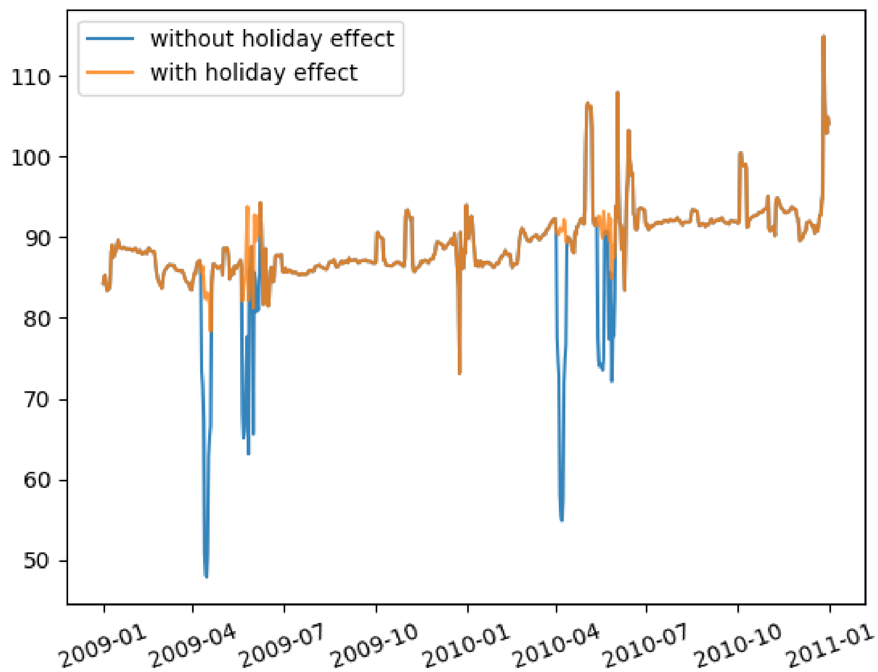

Figure 20: German Trucks seasonally adjusted with/without holiday removal

Figure 21: Dongguan Pollution seasonally adjusted with/without holiday removal

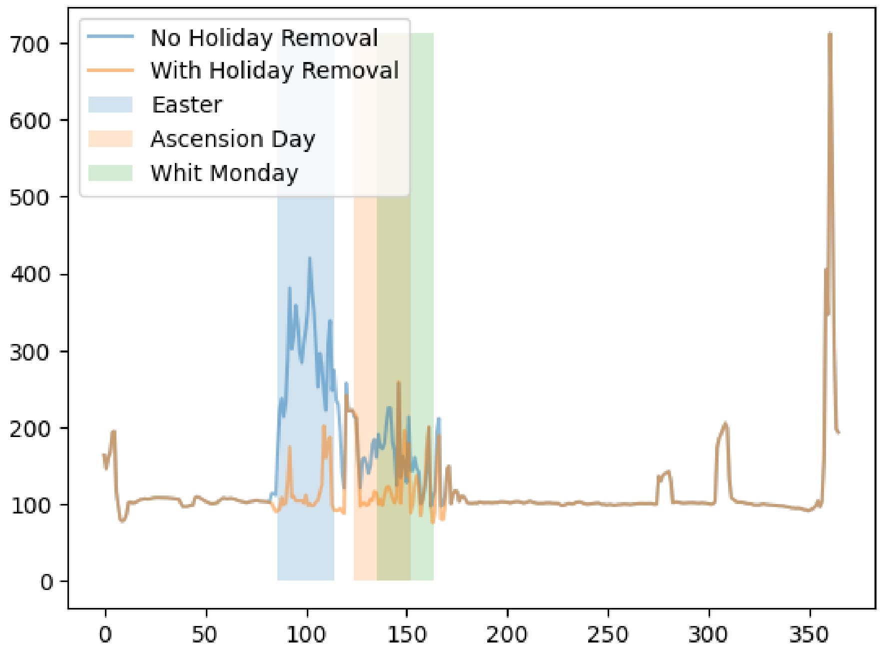

Figure 22: German Trucks Subseries Variance

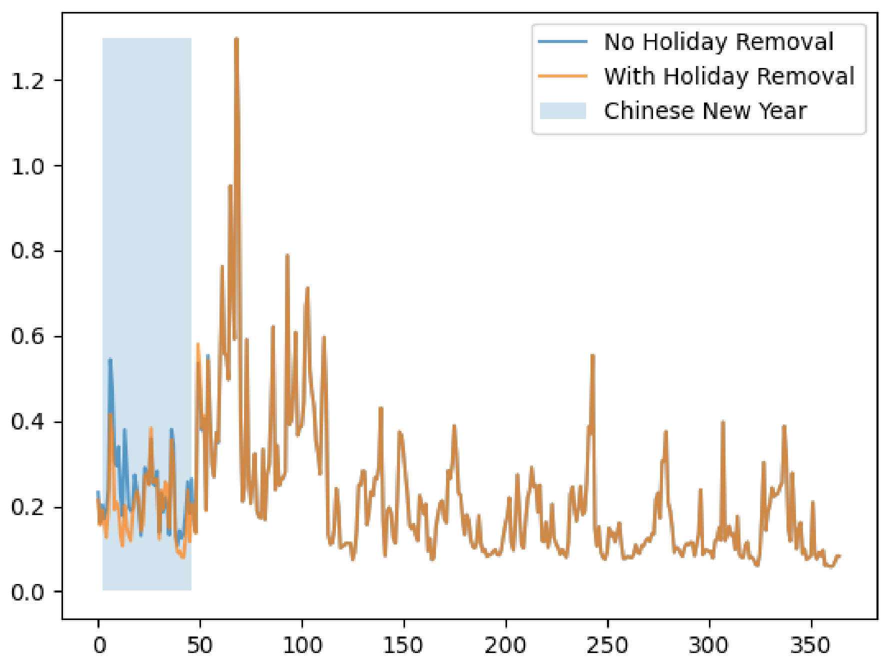

Figure 23: Dongguan Pollution Subseries Variance

The spectral density of the STAHL-adjusted series of German trucks shows that the peaks at yearly periodicity are suppressed, along with the multiple peaks that follow. These correspond in the frequency domain to the seasonal pattern and the Dirac comb-like holiday pattern, which are expunged by the procedure (see Figure 32 in Appendix A.4).

10 Conclusion

In this research note, we unveil STAHL (Seasonal Trend And Holiday decomposition based on LOESS), an innovative framework for seasonal adjustment developed by QuantCube Technology. Drawing on the STL method pioneered by R. B. Cleveland et al. (1990), STAHL has been specifically crafted to accommodate the intricate seasonal patterns observed in diverse daily datasets, such as satellite imagery, textual analytics, daily pricing records, or web searches. STAHL's utility is manifold, but its three primary enhancements stand out.

Foremost is the procedure's adeptness at point-in-time estimation, which significantly enriches the 'industrial production' of seasonally adjusted series. This advancement paves the way for subsequent exploratory research into various asymmetric filters to potentially improve the point-in-time trend extraction procedure. Additionally, we introduce a 'holiday-loop' feature that proficiently accounts for holiday influences, adeptly adjusting for calendar effects that vary over time, including those as diverse as the Chinese New Year and the Hijri calendar. Lastly, STAHL's enhanced capability for handling missing values represents a leap forward in robustness, facilitating the management of gaps in seasonal data.

The practical applications of STAHL have been thoroughly tested, showcasing its ability to proficiently produce data that is adjusted for both seasonal and working-day effects across a spectrum of frequencies and series types, including both traditional and alternative ones. This establishes a new benchmark for industrial-scale seasonal adjustment of high-frequency datasets and may broaden the horizon for future applications in this domain.

A Appendix

A.1 Upsampling Procedure

The upsampling of the weekly data consists in transforming our signal from a sampling frequency of $365.25/7 \approx 52.18$ to a sampling frequency of exactly 53. In practice, given y a seasonal, weekly time series and $\hat{y}$ its discrete Fourier transform, we obtain the modified Fourier transform by extracting $\hat{z}$:

$$\hat{z}(n) = \begin{Bmatrix} \hat{y}(n/M) \text{ if } n/M \in \mathcal{Z} \\ 0 \text{ otherwise} \end{Bmatrix}$$

where $M = 52.18/53$ and $\mathcal{Z}$ is the support of $\hat{y}$.

By taking the inverse Fourier transform or $\hat{z}$, we obtain a time series z that has the exact same spectral signature as y but sampled at a frequency that suits the STAHL procedure. After the seasonal adjustment of the upsampled data, with a n_p parameter set to 53, we can downsample back the data to its weekly frequency by applying the same methodology with $M' = 53/52.18$.

A.2 LOESS Confidence Interval

In this part we need to specify some notation. (x_i, y_i) refers to the set of (x, y) values that are observed in the kernel K_i at point i. The kernel simply applies the set of weights (as can be seen in Figure 2) to the points. A WLS (Weighted Least Square) regression is then applied to get the vector of estimates $\widehat{y_i}$. N is the total number of data points (including missing values). q refers to the kernel width in the LOESS. α is the significance level. We finish off with SE_i which is the standard error for each estimation point in the LOESS. Once this has been calculated it is trivial to use a t-statistic (q) to find the Confidence Interval (CI) around the estimated regression line. This procedure is loosely based on the CI procedure for weighted regressions exposed in Chatterjee and Hadi (2013). The validity of the LOESS confidence procedure is confirmed via Monte-Carlo simulations.

Algorithm 1 LOESS confidence interval

for $i <= N$ **do**
    for $j <= n$ **do**
        $\widehat{y_i}[j] \leftarrow WLS(y, x, W[i], j)$
        $\varepsilon_i^2[j] \leftarrow (y_i[j] - \widehat{y_i[j]})^2$
    $SSTx_i \leftarrow \sum_{j=1}^{q} (Wx_i[j] - \overline{Wx_i})^2$
    $var(\widehat{y_i}) \leftarrow \frac{1}{q-2} \sum_{j=1}^{q} \varepsilon_i^2[j]$
    $var(x_i) \leftarrow \frac{1}{q} \sum_{j=1}^{q} x_i^2[j]$
    $SE_i \leftarrow \sqrt{\frac{var(\widehat{y_i}) \cdot var(x_i)}{SSTx_i}}$
    $CI_i \leftarrow \widehat{y_i[0]} \pm (t(\alpha/2, q) \cdot SE_i)$

A.3 HTL Barnacle Procedure

The Barnacle procedure that we developed here works particularly well in capturing flexible holiday effects. One of the particular challenges was ensuring that the number of false positives was kept to a minimum. To illustrate this point, we show how the Easter Bunny series reacts when a Thanksgiving holiday dummy is included.

A.3.1 The Barnacle Algorithm

Notation: L is a $N \times h$ matrix where N is the number of observations and h is the number of holidays that are evaluated. L_b is a shifted version of L to mark the Barnacle days. In this example, h is 2, Easter and Thanksgiving. L contains 0 on days that are not a holiday event and 1 on days that are holiday events. H is a matrix with $n \times h$ values. n is the number of holiday events that are observed in the series, for Easter or Thanksgiving this means 1 holiday event per year so $n << N$. $\widehat{y_H}$ is a vector of length N that all estimated holiday effects.

Algorithm 2 Barnacle Procedure

$H \leftarrow y[L = 1]$
$\widehat{H} \leftarrow LOESS(H)$
if $\widehat{H}$ is significant **then**
    $\widehat{y_H}[L = 1] \leftarrow \widehat{H}$
$i \leftarrow 0$
while $\widehat{H}$ is significant **do**
    $i \leftarrow i + 1$
    $L_b \leftarrow shift(L, i)$
    $H \leftarrow y[L = 1]$
    $\widehat{H} \leftarrow LOESS(H)$
    if $\widehat{H}$ is significant **then**
        $\widehat{y_H}[L_b = 1] \leftarrow \widehat{H}$
$i \leftarrow 0$
while $\widehat{H}$ is significant **do**
    $i \leftarrow i - 1$
    $L_b \leftarrow shift(L, i)$
    $H \leftarrow y[L = 1]$
    $\widehat{H} \leftarrow LOESS(H)$
    if $\widehat{H}$ is significant **then**
        $\widehat{y_H}[L_b = 1] \leftarrow \widehat{H}$

A.3.2 Testing Significance

Normal significance tests are conducted by examining how much of the confidence interval (CI) contains the x-axis. When testing the significance of holidays, the main issue is that holidays tend to introduce a lot of additional volatility into the series. Figure 12 clearly provides evidence that the series is very volatile around Easter events, whereas during the rest of the year it is quite stable. This creates an issue with false positives when testing for the significance of other holidays occurring during 'stable' periods. We plot the variance of each day of the year for the Easter Bunny series in Figure 24. It is clearly evident that during the period of possible Easter events, there is a much higher variance in searches than during the rest of the year. During the period of possible Thanksgiving dates, the variance is close to zero. This presents an issue because the CI heavily depends on the variance of the true series.

Figure 25 shows the subseries and the smoothed regression with the CI of Easter and Thanksgiving next to each other. Both pass the 80% significance test. However, when looking at the scale and plotting both series (Figure 26), Easter has clearly a much more important effect on the series than Thanksgiving.

In order to counteract these effects we use an *adjusted-CI* (aCI) to evaluate its significance. The aCI uses a multiplier to increase or decrease the CI of a series. In order to calculate the aCI we take the average variance of the series and we divide this by the variance of the holiday subseries.

$$\text{aCI} = \text{CI} \cdot \frac{mean(var(\text{daily subseries})}{var(\text{holiday subseries})}$$

Once we apply the *aCI* we can see in Figure 27 that the significance values have changed. The Easter subseries now has a significance of 90% while the significance of the Thanksgiving series has fallen to just 10%. In Figure 28 we can see that the width of the aCI of the two series is now very similar. The difference comes form the magnitude of the effect.

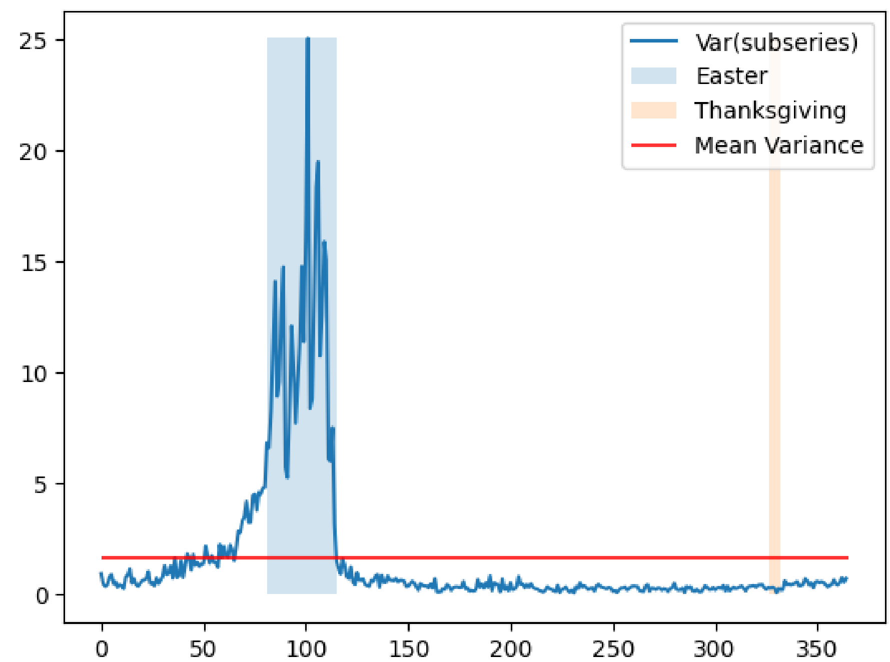

Figure 24: Variance of Easter Bunny searches

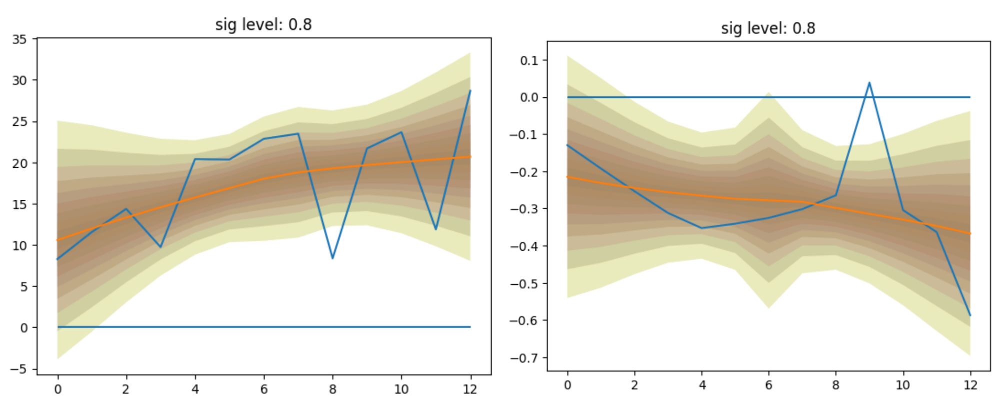

Figure 25: Easter and Thanksgiving Significance with Normal CI

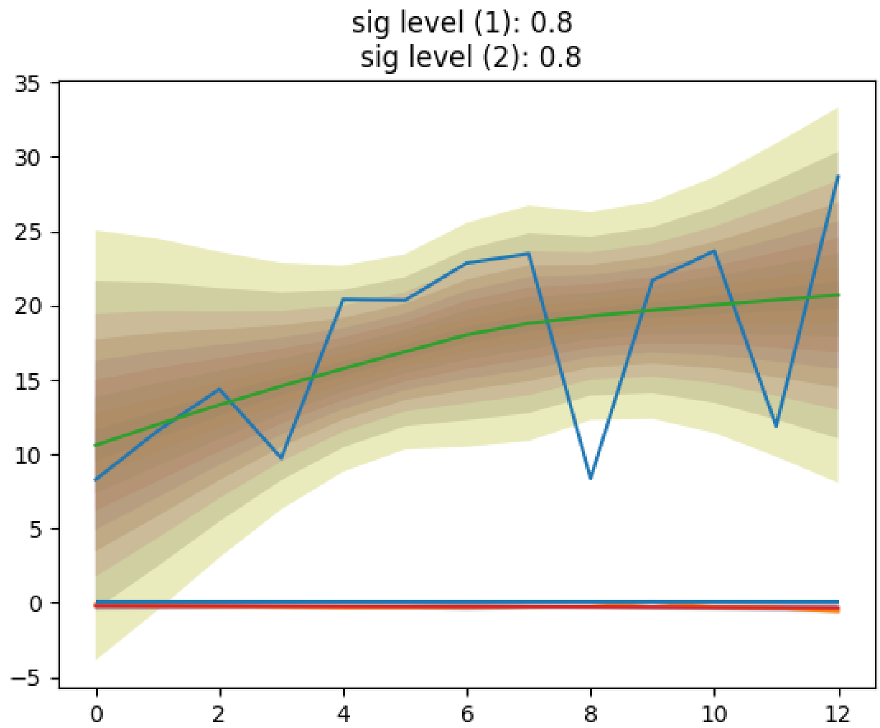

Figure 26: Easter and Thanksgiving with Normal CI

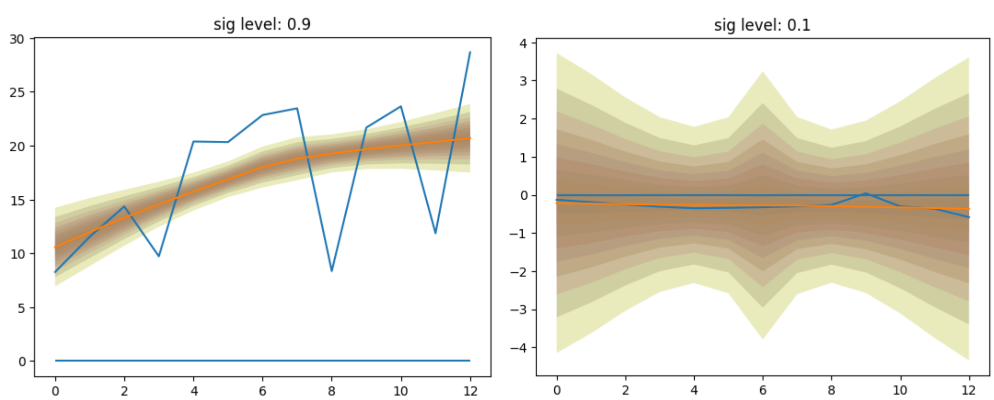

Figure 27: Easter and Thanksgiving Significance with Adjusted CI

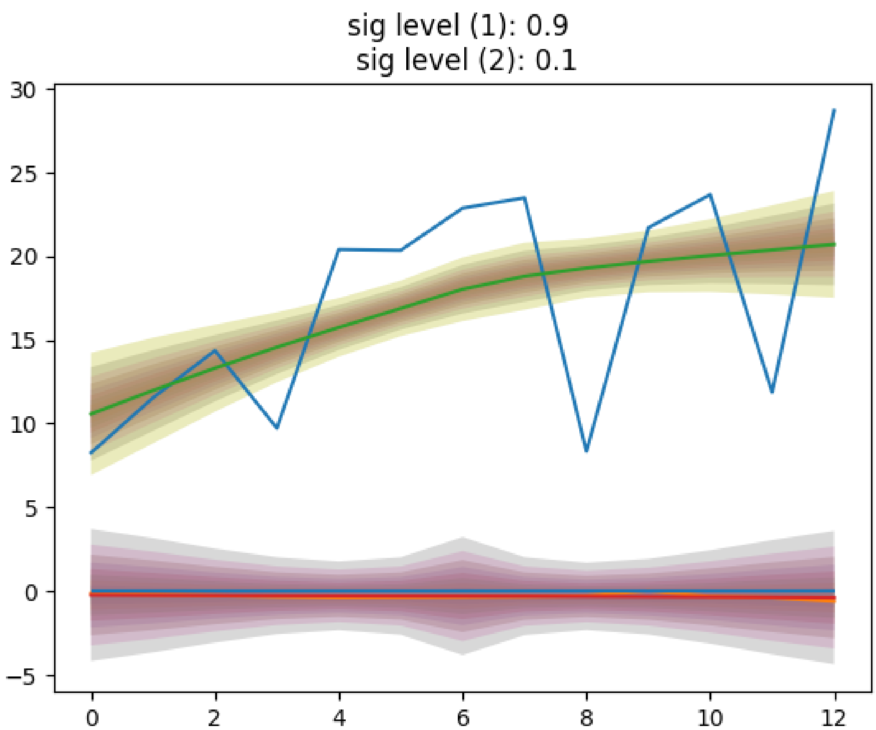

Figure 28: Easter and Thanksgiving with Adjusted CI

A.4 Spectral density plots

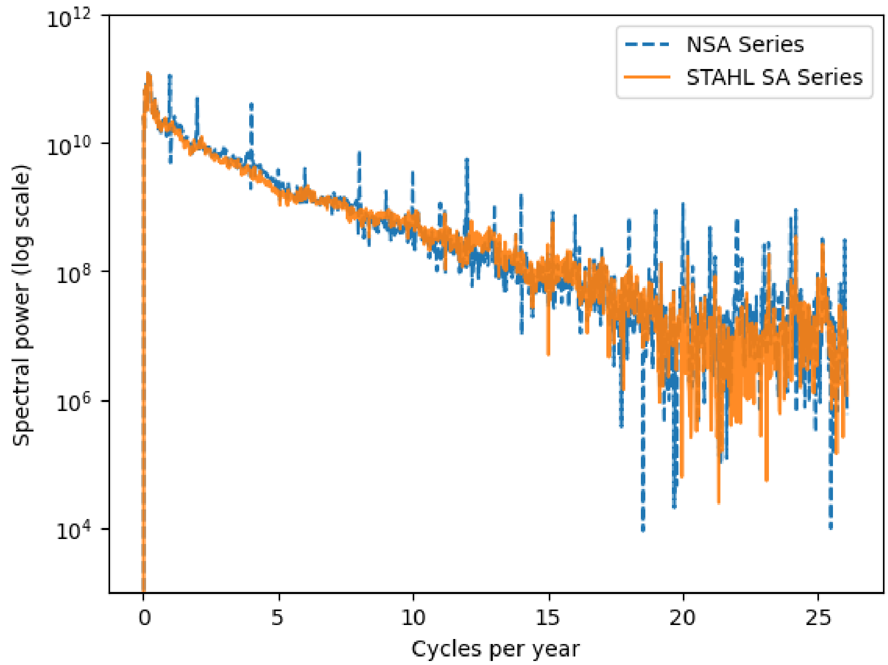

Figure 29: Spectral density for STAHL SA and NSA Initial claims data

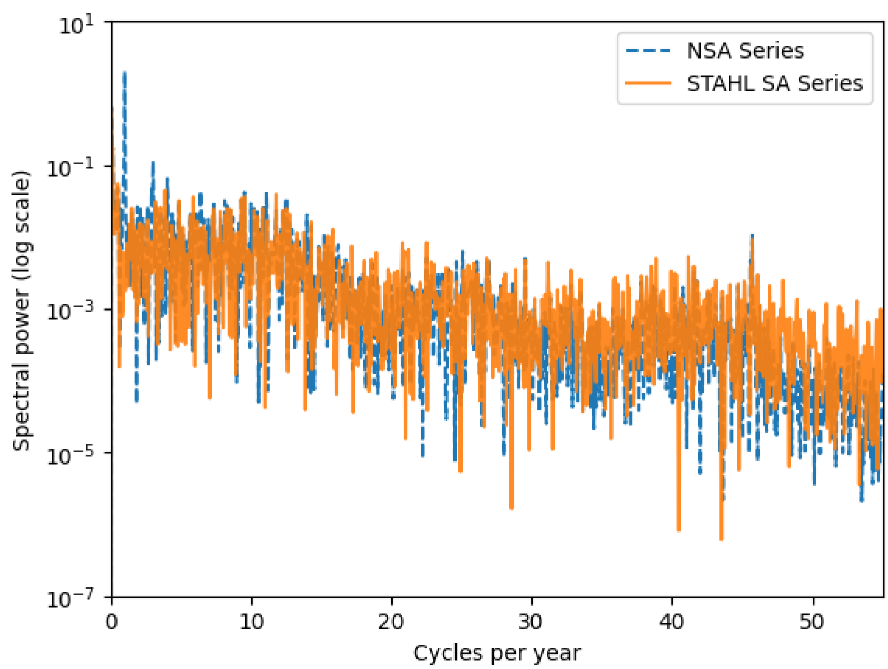

Figure 30: Spectral density for STAHL SA and NSA Pollution over Dongguan, China

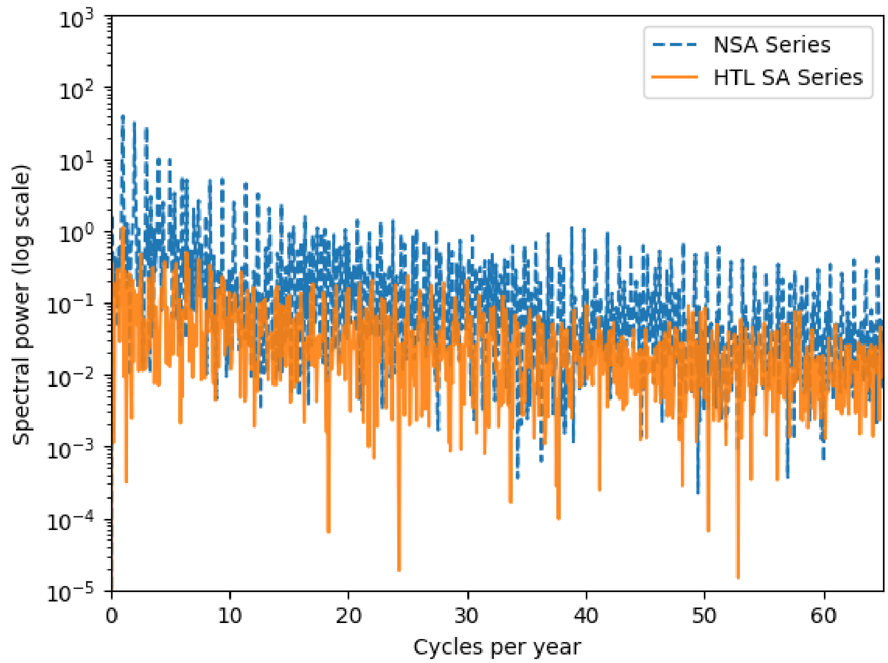

Figure 31: Spectral density for HTL SA and NSA Google search volume for "Easter Bunny"

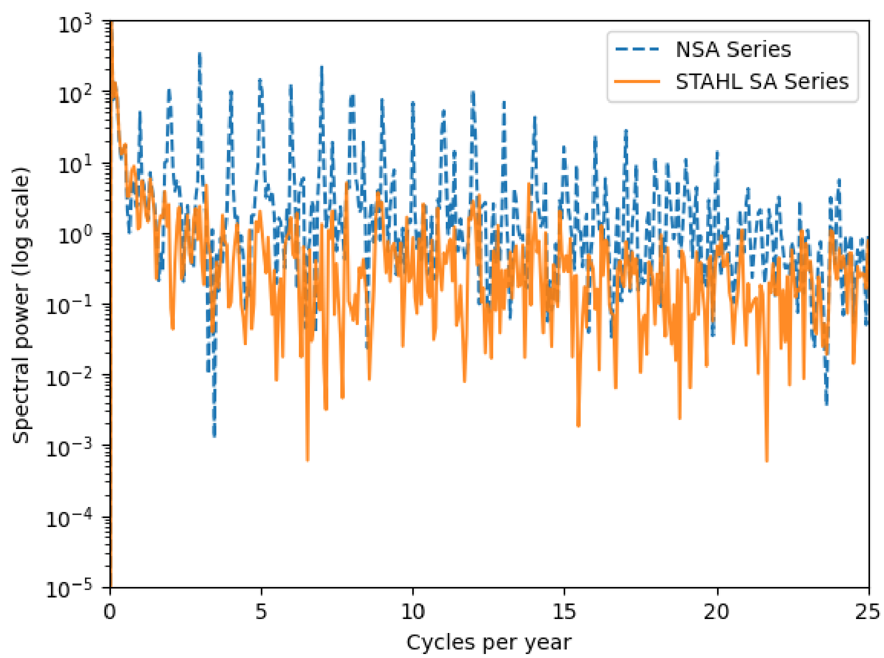

Figure 32: Spectral density for STAHL SA and NSA German Trucks Mileage Index

References

Bandara, Kasun, Rob J Hyndman, and Christoph Bergmeir (2021). "MSTL: A seasonal-trend decomposition algorithm for time series with multiple seasonal patterns." In: *arXiv preprint arXiv:2107.13462*.

Campante, Filipe and David Yanagizawa-Drott (2015). "Does religion affect economic growth and happiness? Evidence from Ramadan." In: *The Quarterly Journal of Economics* 130(2), pp. 615–658.

Chatterjee, Samprit and Ali S Hadi (2013). *Regression analysis by example.* John Wiley & Sons.

Cleveland, Robert B et al. (1990). "STL: A Seasonal-Trend Decomposition Procedure Based on Loess." In: *Journal of Official Statistics* 6(1), pp. 3–73.

Cleveland, William P, Thomas D Evans, and Stuart Scott (2014). *Weekly Seasonal Adjustment-A Locally-weighted Regression Approach.* Tech. rep. Bureau of Labor Statistics.

Grun-Rehomme, Michel, Fabien Guggemos, and Dominique Ladiray (2018). "Asymmetric Moving Averages Minimizing Phase Shift." In: *Handbook on Seasonal Adjustment*, pp. 391–413.

Harvey, Andrew and Siem Jan Koopman (1993). "Forecasting hourly electricity demand using time-varying splines." In: *Journal of the American Statistical Association* 88(424), pp. 1228–1236.

Harvey, Andrew, Siem Jan Koopman, and Marco Riani (1997). "The modeling and seasonal adjustment of weekly observations." In: *Journal of Business & Economic Statistics* 15(3), pp. 354–368.

Ladiray, Dominique, Jean Palate, et al. (2018). "Seasonal adjustment of daily data." In: *Proceedings of the 16th Conference of IAOS, Paris, France*, pp. 19–21.

Ladiray, Dominique and Benoit Quenneville (2012). *Seasonal adjustment with the X-11 method.* Vol. 158. Springer Science & Business Media.

Li, Zhenhui, Jingjing Wang, and Jiawei Han (2012). "Mining event periodicity from incomplete observations." In: *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 444–452.

McElroy, Tucker S, Brian C Monsell, and Rebecca J Hutchinson (2018). "Modeling of holiday effects and seasonality in daily time series." In: *Statistics* 1.

Ollech, Daniel (2021). "Seasonal adjustment of daily time series." In: *Journal of Time Series Econometrics* 13(2), pp. 235–264.

Proietti, T and DJ Pedregal (2022). *Seasonality in High Frequency Time Series. Econometrics & Statistics.*

Puech, Tom et al. (2020). "A fully automated periodicity detection in time series." In: *Advanced Analytics and Learning on Temporal Data: 4th ECML PKDD Workshop, AALTD 2019, Würzburg, Germany, September 20, 2019, Revised Selected Papers 4.* Springer, pp. 43–54.

Taylor, Sean J and Benjamin Letham (2018). "Forecasting at scale." In: *The American Statistician* 72(1), pp. 37–45.

Tukey, John Wilder (1960). “A survey of sampling from contaminated distributions.” In: *Contributions to probability and statistics*, pp. 448–485.

Webel, Karsten (2022). “A review of some recent developments in the modelling and seasonal adjustment of infra-monthly time series.” In: *Deutsche Bundesbank Discussion Paper*.

Wildi, Marc (2007). “Real-Time Signal Extraction.” In: *Lecture Notes in Economics and Mathematical Systems* 547.

Wildi, Marc and Tucker McElroy (2016). “Optimal real-time filters for linear prediction problems.” In: *Journal of Time Series Econometrics* 8(2), pp. 155–192.

Wildi, Marc and Tucker S McElroy (2019). “The trilemma between accuracy, timeliness and smoothness in real-time signal extraction.” In: *International Journal of Forecasting* 35(3), pp. 1072–1084.