THEORETICAL FRAMEWORK FOR INDICATOR CONSTRUCTION

4.1 Introduction to Linear Dynamic Harmonic Regression

The methodology of the Dynamic Harmonic Regression used in this study has proven to be especially useful for the tasks related to the manipulation of time series data, such as adaptive seasonal adjustment, signal extraction, and time series forecasting. The Linear Dynamic Harmonic Regression algorithm consists in the identification and the estimation of the unobserved components that form the time series, assigning the correct model for each of them. This regression model is based on a spectral approach that decomposes the series into various DHR components who have different variances that are concentrated around certain frequencies. Therefore, this algorithm makes an assumption that the time series at hand has well-defined spectral peaks.

Considering a univariate time series and applying the LDHR algorithm, the series can be decomposed into the trend, the seasonal, or a periodic component, and the irregular component or error term, which is assumed to be normally distributed and with zero mean. Each of those components have different variances that can be scaled at low or high frequencies, much like the musical harmonics of sound waves. The trend of the series is the component of most interest for this study, and its variance, which is infinite in the theoretical model, is located at a low frequency. The trend is also known as a low frequency component; the objective of this study is to estimate it for each time series chosen and then utilise the weighted aggregation of trends in order to construct a synthetic composite indicator.

Before diving into how LDHR is applied in this specific study, it is essential to first discuss how this algorithm would work on a theoretical model.
There are two types of theoretical models of interest: mean non-stationary and mean stationary models. A mean non-stationary model is a time series model where the average value of the series, defined as the summation of all data points divided by the number of data points, is non-constant over time. This means that the average value of the series might increase or/and decrease over time. A mean stationary time series is the opposite: the average value of the series is unchanged over time, and even though data points fluctuate around this mean, the average value remains constant around different time periods. It is crucial to mention that stationary time series are much easier to analyse and work with because they exhibit a well defined variance, autocovariance and mean, which are the statistical properties necessary to build a model over which the time series can be interpolated and extrapolated.
It is here where it must be noted that the time series subject to analysis and identification in this particular study are non stationary: this type of data is most commonly encountered in economics due to the dynamic and non-constant growth of most economic and financial variables over time. GDP components, inflation rates, unemployment rates and stock prices manifest systematic patterns with non - constant mean and variance over time, affected by complex consumer behaviour choices, external shocks and the climate of market dynamics at a certain point in time, all of which is the reason why non stationary time series are challenging to model and forecast.

4.2 The theoretical functioning of Linear Dynamic Harmonic Regression

In a theoretical framework with a mean stationary time series, LDHR is set in motion with the concept of the autocovariance generating function.
A time series inherently displays a degree of correlation between data points across time. This correlation can be quantified by calculating covariances between a data point at time 't' and previous data points at 't-1', 't-2', 't-3', and so forth. The computed autocovariances can be recorded into a vector which forms the autocovariance generating function, where the first element of this vector is the variance of the time series, or the covariance of a data point with itself.

A visual representation of the autocovariance generating function can be viewed in a correlogram, which is a depiction of autocovariances that are uniformly graphed in bar format, and each bar is commonly referred to as a lag. This is a widely used tool in ARMA model specification where the speed with which these lags decrease with time or the statistical significance of each lag is crucial in identifying whether the model has significant MA (moving average) or/and AR (autoregressive) components.

A stationary time series that may be identified with an ARMA (Autoregressive Moving Average) model may be expressed in polynomial form, where time series data multiplied by the autoregressive component (AR) polynomial are equal to a white noise process multiplied by a moving average (MA) polynomial. For a series to be mean stationary, the AR polynomial has to be invertible (it has to possess roots outside of the unit circle), which means it would be possible to multiply this polynomial by its own inverse and the result would yield ‘one’. This result is known as a unique stationary solution for an infinite stationary sequence of time series data. It is also important to mention that there is an infinite number of inverses for this polynomial, but only one of those inverses is applicable to this study.
To expand, every inverse of this AR polynomial is an infinite sequence of numbers, but only one of those inverses has an infinite sequence of numbers such that these numbers are summable to a finite quantity. So, in order to combat this problem successfully, the AR polynomial must be multiplied by this special inverse and the product would yield ‘one’.

Adopting the assumption that the above requirements are indeed satisfied, a theoretical ARMA model is the one for which the original data is equal to an MA polynomial multiplied by the inverse of an AR polynomial, multiplied by white noise process. This way of disintegrating the original series at hand is known as the infinite moving average representation of time series data.
The autocovariance generating function is related to the notion of an ARMA process because for an Autoregressive process, this function is the inverse of the AR polynomial, while for an MA process, the autocovariance generating function is a Moving Average polynomial itself.

Once the autocovariance generating function is well defined for a stationary time series, it may be passed through the Fourier Transform and yield the spectrum of this stationary time series process. The spectrum of a univariate time series can be defined as a representation of the frequency content of the variances of this series. The analysis and the characteristics of the spectrum will reveal the distribution of the variance frequencies in the data, much like the frequential energy content distribution of a stationary signal. So, by employing the Fourier transform to the autocovariance

generating function, the spectrum can be expressed as a summation of cosine functions of variance frequencies multiplied by the autocovariances of the series.
Once the spectrum is well defined, it is possible to depict it between pi and -pi. This depicted function evaluated at 0 would possess a maximum (with the maximum variance of the series), which is the point of the lowest frequency for the data.

Despite of how well the autocovariance generating function is able to transform into the spectrum of a stationary time series, non - stationary stochastic processes present a huge dilemma inside this theoretical framework. Since non-stationary series of data possess no explicitly defined mean and have infinite variance, it is not mathematically possible to compute the autocovariance generating function, nor does it really exist. Therefore, there is a level of abstraction regarding the statistical methods applied to analyse and model non stationary data. There is no unique stationary solution, nor is there an inverse of an AR polynomial which would multiply this polynomial and yield ‘1’. So, whenever the AR polynomial has roots on the unit circle, the spectrum is not well defined.

However, used in modern literature, the notion of a pseudospectrum will be utilised, which would describe the distribution of variances over their frequencies for a non stationary stochastic process. For a time series with no defined mean or variance, it is possible to determine certain ‘pseudo covariances’ and therefore have a ‘pseudo autocovariance generating function’, which is a fictitious function that cannot yet be explained through coherent mathematical findings. Nevertheless, non-stationary data are still assumed to be equal to the product between an MA polynomial, the inverse of an AR polynomial and white noise process. However, in this particular case, the inverse of an Autoregressive polynomial would be an infinite sequence of numbers which is not summable into a finite quantity, unlike in the stationary case.

Despite the scientific challenges described above, the pseudo covariance generating function is still employed to further investigate non - stationary stochastic processes. By passing the pseudo covariance generating function through the Extended Fourier Transform, it is possible to obtain the pseudospectrum of non - stationary data. The pseudospectrum describes the distribution of variance over frequency, much like the spectrum for stationary time series data, but it is rather obtained through pseudo covariances that are not explicitly defined in mathematical literature.

The depiction of the pseudospectrum is equivalent to the one of the stationary time series spectrum, except that each root of the unit circle of non - stationary data results in a pole in the pseudospectrum with an infinite variance.

Another way of tackling the problem of the pseudo covariance generating function and the frequency distribution of variances of non stationary processes is through the Ordinary Least Square minimisation of the distance between an estimation of the spectrum of non stationary data and its real spectrum. The problem is that the real spectrum for non-stationary series is the pseudospectrum with spectral peaks of infinite variance. Therefore, the distance to minimise is not well defined and is infinite in some points.

Linear Dynamic Harmonic Regression addresses this issue by detecting a certain function (for each of the time series components) that will multiply the minimization problem explained above and transform the spectral peaks with infinite variance into points with zero variance. To rephrase, LDHR focuses on determining the appropriate function that will convert the originally undefined infinite distance into a minimised distance through the method of Ordinary Least Squares for each of the components involved of time series data.

PRACTICAL DEVELOPMENT OF SYNTHETIC INDICATOR

5.1 Criticism of Moving Average smoothing techniques over the LDHR method

Prior to applying the Linear Dynamic Harmonic Regression method to extract the smoothed trend for each of the chosen datasets, a more common technique is the Moving Average smoothing that has been applied to this study through built in Octave functions. Moving Average smoothing is a well known algorithm used in signal processing and time series analysis to reduce noise and reveal the underlying patterns within the data. This method is applied by taking an average of a sample of close- by data points and replacing each data point in this sample with its average. In Octave, the function ‘movmean’ allows the user to input a window size for the number of neighbouring data points to include in each mean calculation. And so, the original data is replaced by a set of moving average values.

However, moving average filters are often criticised and alternative methods are employed to smooth the data. Firstly, moving average filters introduce a delay in the data, which would lead to inaccuracies in predicting future trends. Secondly, the MA filter exhibits poor performance when it comes to handling anomalies in the data, and this could be particularly harmful when these anomalies provide valuable insights into the underlying patterns of the data. The filter smoothes out important spikes or dips, potentially discarding vital information about the behaviour of the time series.

The third, and probably the most important notion for this study is the fact that Moving Average filters manifest a poor frequency response in time series data smoothing.
Time series data exhibits various frequency components, including the low-frequency trend and the high-frequency noise. Seasonal components display a range of frequencies between the maximum and the minimum frequencies.
The Moving Average filter aims to diminish the high frequency noise in the data and preserve the low frequency components, such as the trend, or any other gradual variations in time series data. However, the issue is that this filter uniformly weakens all frequencies, resulting in insufficient reduction of high frequencies during the smoothing process. Consequently, the trend pattern does not appear as smooth as desired.

This issue persists when applying the first difference to the moving average trend. Taking the first difference is a widely used technique to put emphasis on short-term fluctuations to better capture the rate of change of the data. That said, the first difference filter amplifies high-frequencies once again, and since the Moving Average filter did not reduce the high frequencies sufficiently, once they are amplified, the differenced trend exhibits noisy and unclear behaviour.
In contrast, the Linear Dynamic Harmonic Regression algorithm effectively reduces high frequencies of the original data to zero in the smoothed trend. Consequently, when applying the first difference to this trend and higher frequencies are once again amplified to highlight minor fluctuations, the outcome is once again a smooth, but differenced trend. This is because the annulled high noise frequencies remain unaffected

during the amplification process of applying the first difference, resulting in a smooth differenced trend.

5.2 Programming Implementation for Linear Dynamic Harmonic Regression

In order to put theory to practise with real world data for the United States of America, the Octave programming interface has been utilised in this study. The objective is to dissect the economic datasets through the algorithm of Linear Dynamic Harmonic Regression. Initially, monthly time series data that have not been seasonally adjusted prior to the analysis, have been imported in comma separated value format into the Octave interface in Jupyter notebook. It is vital for this study to employ raw data that has not been modified or filtered using statistical smoothing and pattern adjustment techniques since Linear Dynamic Harmonic Regression will focus on the application of its unique filtering algorithm so meticulously narrated above.

It is important to remember that the data used in this analysis is mean non - stationary, but the datasets are composed of a finite set of observations, as a sample from an infinite non-stationary stochastic process is taken. Therefore, the samples from the datasets chosen do have a defined mean and variance, and so it is possible to define the autocovariance generating function and the corresponding spectrum. The spectral peaks in this study would take very large values (however, they are not infinite), and these peaks are located at the specific frequencies where there exists a significant contribution to the variance of the dataset.

Once the data has been collected and imported, then, for each separate time series at hand, the LDHR algorithm in Octave is the following. Firstly, it is essential to define the inputs for the ‘autodhr’ function, which is a custom made programming construction in Octave that will identify and estimate a model for each one of the components of the time series data, completing the Linear Dynamic Harmonic Regression algorithm.

The first input for the ‘autodhr’ function is a vector called PaP and it is designed to outline the periods, or the harmonics corresponding to seasonality. This vector indicates the ‘autodhr’ program to look for the models with a trend, a seasonal component of period 12 and all of its corresponding harmonics:

```
PaP=12./(0:6)
PaP=[Inf, 12, 6, 4, 3, 2.4, 2]
```

For monthly observations, the seasonal component follows a harmonic pattern identical to music harmonics, oscillating at periods of 12, 6, 4, 3, 2.4, and 2. The spectral peaks for the seasonal component of monthly data are accordingly found at frequencies such as '2pi/12', '2pi/6', '2pi/4', '2pi/3', '2pi/2.4' and '2pi/2'.

Afterwards, a matrix denominated TVPaP is created as a second input for the ‘autodhr’ function. This matrix possesses two rows and as many columns as the LDHR components in the time series. With monthly data employed for this analysis, there are 7 LDHR components, the first one being the trend, followed by 6 seasonal components. The elements of this matrix indicate the modulus of the roots of the Autoregressive processes of order 1 and 2, which are able to model the amplitude of the oscillations of each component of the series:

```
TVPaP = [1 1 1 1 1 1 1 ; 1 0 0 0 0 0 0]
TVPaP =
  [ 1   1   1   1   1   1   1
    1   0   0   0   0   0   0]
```

Once these two elements have been defined, the ‘autodhr’ function is designed to identify and estimate the best model for each time series component present in the original series. Additional inputs for this programming algorithm are the original, not modified data and the periodicity of the observations, which is 12 for monthly data and 4 for quarterly data. The third and the fourth input in this function are optional, and the square brackets indicate that the default parameters for these inputs ought to be used:

(**what did 1 at the end stand for again)**

```
[VAR, P, TVP, oar]=autodhr(data, 12, [], [], PaP, TVPaP, 1)
```

The outputs, or the exit arguments of this function are of most interest for this study. The output ‘VAR’ is a vector of 8 elements that stores the variances of innovations for the LDHR components. There are 8 variances stored in this vector, the first corresponding to the trend variance, the last corresponding to the irregular component variance, and the remaining 6 being the variances for each of the seasonal components. The second output, P, corresponds to the periodicity of LDHR components, and it is composed of the same values as the PaP input vector of this programming construct. The explanation behind the TVP matrix of output is identical to that of the P vector output: it is the same matrix that indicates the modulus of the roots of the AR processes used in this regression. Last but not least, the ‘oar’ output of the ‘autodhr’ instruction indicates the autoregressive order employed in the estimation of the spectrum of the analysed dataset. For most of the time series utilised in this study, the value of this autoregressive order exceeded 20.

Consequently, but prior to the extraction of each component from the model estimated, it is necessary to define a vector denominated NVR, which computes the ratios of the variance of innovations of each component (except for the trend component) over the variance of innovations of the trend of the series. This is imperative to scale the importance of every component’s variance over the maximum variance of the series, which is the trend variance:

```
NVR =VAR (2:8)./VAR (1)
```

Finally, in order to dissect the original data into its trend, seasonal component, cyclical component and its irregular component, the ‘dhrfilt’ function is applied. This is another custom made instruction within the Octave framework that is able to extract the necessary components listed above from the data. The ‘dhrfilt’ function by itself is not part of the Linear Dynamic Harmonic Regression algorithm, but much rather is a Kalman Filter application to the LDHR framework. The Kalman filter is a useful mathematical algorithm in time series analysis as it is able to estimate the true underlying state of a dynamic system observed through noisy measurements over time.

Now, the inputs of the ‘dhrfilt’ program are such as the original dataset, the parameters serving as outputs of the ‘autodhr’ function, specifically the ‘P’ vector, the ‘TVP’

matrix, the ‘VAR’ vector, the periodicity of the data and the type of filter applied. In this particular analysis, the ‘filt’ input is equal to 0, which indicates to the ‘dhrfilt’ program to refer to the ‘e4trend’ function in a custom made E4 toolbox in Octave. The final input equal to 0 specifies to the ‘dhrfilt’ program to omit and graphical representations of the estimated components:
(what does the e4trend function do)

```
[trend, season, cycle, irreg] = dhrfilt (data, P, TVP, VAR, 12, filt, 0)
```

The output arguments of this programming construct are the estimated LDHR components. The ‘trend’ consists of a matrix in which the first column is the trend vector of interest for this study. ‘Season’ is another matrix with its first column corresponding to the complete seasonal component: the remaining columns display the other estimated seasonal components that mirror the harmonics explained above. ‘Cycle’ is the third output of the ‘dhrfilt’ function and it is a matrix where the first column is the complete estimated cyclical component of the series, but in general this is a matrix full of zeros as no cyclical LDHR components have been defined previously in this study. The last output corresponds to the irregular component or the noise of the observed data.