Matlab Toolbox for Mixed Sampling Frequency Data Analysis using MIDAS Regression Models

Eric Ghysels

First Draft: December 2009
This Draft: December 21, 2014

© 2014 All rights reserved

Version 1.0

*The author benefited from funding by the Federal Reserve Bank of New York through the Resident Scholar Program. The code was written with the help of Xiafei Hu, Hang Qian, Arthur Sinko and Michael Sockin. Questions, comments and bug reports can be sent to matlabist@gmail.com.

†Department of Finance - Kenan-Flagler Business School and Department of Economics, University of North Carolina, McColl Building, Chapel Hill, NC 27599. e-mail: eghysels@unc.edu.
1 Introduction

Regression models involving data sampled at different frequencies are of general interest. In this document we describe Matlab code for running such regression based on a framework put forward in recent work by Ghysels, Santa-Clara, and Valkanov (2002), Ghysels, Santa-Clara, and Valkanov (2006) and Andreou, Ghysels, and Kourtellos (2010) using so called MIDAS, meaning Mi(xed) Da(ta) S(ampling), regressions.

Several recent surveys on the topic of MIDAS are worth mentioning at the outset. They are: Andreou, Ghysels, and Kourtellos (2011) who review more extensively some of the material summarized in this document, Armesto, Engemann, and Owyang (2010) who provide a very simple introduction to MIDAS regressions and finally Ghysels and Valkanov (2012) who discuss volatility models and mixed data sampling.


Recent work has used the regressions in the context of improving quarterly macro forecasts with monthly data (see e.g. Armesto, Hernandez-Murillo, Owyang, and Piger (2009), Clements and Galvão (2008a), Clements and Galvão (2008b), Frale and Monteforte (2011), Kuzin, Marcellino, and Schumacher (2011), Monteforte and Moretti (2012), Marcellino and Schumacher (2010), Schumacher and Breitung (2008)), or improving quarterly and monthly macroeconomic predictions with daily financial data (see e.g. Andreou, Ghysels, and Kourtellos (2013), Ghysels and Wright (2009), Hamilton (2008)).


MIDAS regression can also be viewed as a reduced form representation of the linear projection that emerges from a state space model approach - by reduced form we mean that the MIDAS regression does not require the specification of a full state space system of equations. Bai,
Ghysels, and Wright (2012) show that in some cases the MIDAS regression is an exact representation of the Kalman filter, in other cases it involves approximation errors that are typically small. The Kalman filter, while clearly optimal as far as linear projections goes, has several disadvantages (1) it is more prone to specification errors as a full system of measurement and state equations is required and as a consequence (2) requires a lot more parameters, which in turn results in (3) computational complexities that often limit the scope of applications. In contrast, MIDAS regressions - combined with forecast combination schemes if large data sets are involved (see Andreou, Ghysels, and Kourtellos (2013)) are computationally easy to implement and more prone to specification errors.

2 Intro to MIDAS regressions

For illustrative purpose we start with a combination of two sampling frequencies - and select a combination of one quarterly and one daily time series. In particular, suppose we are interested in forecasting a quarterly series $h$ horizons ahead ($h = 1, 2, \text{etc. quarters}$), denoted $Y_{t+h}$, using daily time series, denoted $X_{j,t}^{D}$, for the $j^{th}$ day in quarter $t$ (with $j = 1$ the first day of the quarter and $j = N_{D}$ the last, with $N_{D}$ the number of days in a quarter - assumed constant for simplicity). The conventional approach, in its simplest form, consists of aggregating the daily data to a quarterly frequency by computing for example averages to obtain $X_{t}^{Q} = (X_{N_{D},t}^{D} + X_{N_{D}-1,t}^{D} + \cdots + X_{1,t}^{D})/N_{D}$ and subsequently estimate a regression:

$$Y_{t+h}^{Q} = \mu + \beta X_{t}^{Q} + u_{t+h}$$

(2.1)

where $\mu$ and $\beta$ are unknown parameters and $u_{t+h}$ is the error term. In (2.1) uses implicitly an equal weighting scheme of the high frequency data since aggregation is based on an average of the daily data. An alternative approach would consist of estimating a model:

$$Y_{t+h}^{Q} = \mu + \sum_{j=0}^{N_{D}-1} \beta_{N_{D}-j} X_{N_{D}-j,t}^{D} + u_{t+h}.$$  

(2.2)

Such an approach is unappealing because of parameter proliferation: when $N_{D} = 66$, we have to estimate 68 slope coefficients$^1$

$^1$Typically we have about 66 observations for many daily financial data over a quarter since each month has 22 trading days.
The key feature of MIDAS regression models is the use of a parsimonious and data-driven weighting scheme. Let us proceed for the moment with the case $h = 1$. The discussion of $h > 1$ will covered in subsection 2.3. The parsimonious specification yields a linear projection of high frequency data $X_{D,t}^j$ onto $Y_t^Q$ using only a few parameters:

$$Y_{t+1}^Q = \mu + \beta \sum_{j=0}^{N_D-1} w_{N_D-j}(\theta^D) X_{N_D-j,t}^D + u_{t+1}.$$ (2.3)

Note that equation (2.3) nests the regression model in equation (2.1) under equal or flat weights. We assume that $\sum_{j=0}^{N_D-1} w_{N_D-j}(\theta^D) = 1$, which allows us to identify the slope coefficient $\beta$ in the MIDAS regression model. The parameters $(\mu, \beta, \theta^D)$ are estimated by Nonlinear Least Squares (NLS). Later we will discuss various parametric specifications for the weighting schemes.

Our understanding of MIDAS regression can be further enhanced by decomposing the conditional mean in equation as the sum of an aggregated term based on flat weights, $X_t^Q$, and a weighted sum of (higher order) differences of the high frequency variable. Following *, we can easily show that the MIDAS term in equation (2.3) can be written as

$$\sum_{j=0}^{N_D-1} w_{N_D-j}(\theta^D) X_{N_D-j,t}^D = \frac{1}{N_D} X_{N_D,t}^D + \frac{1}{N_D} X_{N_D-1,t}^D + ... + \frac{1}{N_D} X_{1,t}^D$$

where the last parenthesis uses the assumption that the weights sum to one. Substituting equation (2.4) into (2.3) we get

$$Y_{t+1}^Q = \mu + \beta X_t^Q + \beta \sum_{j=0}^{N_D-1} (w_{N_D-j}(\theta^D) - \frac{1}{N_D}) X_{N_D-j,t}^D + u_{t+1}.$$ (2.5)

Equation (2.5) shows that the traditional temporal aggregation approach, which imposes flat weights $w_j = 1/N_D$ and only accounts for $X_t^Q$, yields an omitted variable term in the regression model (2.1). This implies a host of econometric estimation issues - pertaining
to asymptotic inefficiencies at best or, as typical, asymptotic biases - that are discussed in

The remainder of this section is structured as follows. In subsection 2.1 we cover DL-MIDAS
regressions, followed by ADL-MIDAS in subsection 2.2. The parameterization of the various
MIDAS regressions is covered in subsection 2.4. Multiplicative ADL-MIDAS are discussed in
subsection 2.5. In the next subsection we introduce MIDAS regressions involving other low
frequency regressors followed by a subsection covering so called leads in MIDAS regressions.

### 2.1 DL-MIDAS regressions

MIDAS regressions share some features with distributed lag models, or DL models, and also
have unique novel features. A stylized DL model is of the following type:

$$
Y_{t+1}^Q = \mu + \sum_{j=0}^{q_Q-1} \beta_j(\theta^Q)X_{t-j}^Q + u_{t+1},
$$

where \(\sum_j \beta_j(\theta^Q)\) is some finite or infinite lag polynomial operator, usually parameterized
by a small set of hyperparameters (see e.g. Dhrymes (1971) for a survey on distributed lag
models). The same idea was used in the MIDAS regression we discussed so far, albeit with
series sampled at different frequencies.

By analogy with DL models, we can characterize a DL-MIDAS\((p_X^D)\) regression model as:

$$
Y_{t+1}^Q = \mu + \beta \sum_{j=0}^{q_X^D-1} \sum_{i=0}^{N_D-1} w_{N_D-i+j+N_D}(\theta^D)X_{N_D-i,t-j}^D + u_{t+1},
$$

where the second summation allows for daily lags to extend beyond the last day of quarter \(t\),
but to simplify notation, we will always take lags in blocks of quarterly sets of daily data, \(p_X^D\).

Note that equation (2.7) nests the simple DL model in equation (2.1) under flat-weights. We
assume again that \(\sum_{j=0}^{p_X^D-1} \sum_{i=0}^{N_D-1} w_{N_D-i+j+N_D}(\theta^D) = 1\), which allows for the identification of
the slope coefficient \(\beta\) in the DL-MIDAS regression model.

Note that in a general DL-MIDAS regression model the regressor is sampled \(m\) times more
frequently than the regressand where the latter has a sample of \(T\) observations. The
asymptotics are based on \(T\) for a given \(m\). Therefore one studies the asymptotic properties
of the estimators assuming that the span of the data set $T$ grows and the high frequency sample size of the regressors would be $mT$ such that when $T \to \infty$ then both the low and high frequency samples become large.

### 2.2 ADL-MIDAS regressions

When $Y^Q_t$ is serially correlated, as it is typically the case for time series variables, the simple model in equation (2.1) is extended to a dynamic linear regression or autoregressive distributed lag (ADL) model. Again the conventional approach, in its simplest form, aggregates the high frequency data at the low frequency by computing simple averages and estimates a simple linear regression of $Y^Q_{t+1}$ on $X^Q_t$. Take for instance the ADL(1,1) and let us set $h$ for the moment:

$$Y^Q_{t+1} = \mu + \mu Y^Q_t + \beta X^Q_t + u_{t+1},$$

(2.8)

where $\mu$ and $\beta$ are unknown parameters and $u_{t+1}$ is an error term. In a similar manner the ADL-MIDAS($p^Q_Y, p^D_X$) is:

$$Y^Q_{t+1} = \mu + \sum_{j=0}^{p^Q_Y-1} \mu_{j+1} Y^Q_{t-j} + \beta \sum_{j=0}^{p^Q_Y-1} \sum_{i=0}^{N_{D}-1} w_{N_{D}-i+j} N_{D} (\theta^D) X^D_{N_{D}-i,t-j} + u_{t+1}$$

(2.9)

Note that again the number of daily lags is a multiple of the number of trading days in a quarter, $N_{D}$. As above the slope coefficient $\beta$ in the MIDAS regression is identified via the scaling of the weights, such that they add up to one. The above model specification generates notation very similar to ARMA models, e.g. ADL-MIDAS(1,1) or ADL-MIDAS(AIC,AIC) (more on model selection later).

### 2.3 Some comments about multi-step horizon forecasts

The topic of mixing different sampling frequencies also emerges even when time series are available at the same frequency, but one is interested in multi-period forecasting. Take the example of an annual forecast with quarterly data. The first approach is to estimate a model with past annual data, and hence collapse the original multi-period setting into a single step forecast. The second approach is to estimate a quarterly forecasting model and then iterate forward the forecasts to a multi-period annual prediction. The forecasting literature...
refers to the first approach as *direct* and the second as *iterated*. (Marcellino, Stock, and Watson (2006)). Traditionally, the comparison has been made between direct and iterated forecasting, see e.g. Findley (1983), Findley (1985), Lin and Granger (1994), Clements and Hendry (1996), Bhansali (1999), and Chevillon and Hendry (2005). Multi-period forecasts can also be constructed using a mixed-data sampling approach. A MIDAS model can use past quarterly data to produce directly multi-period forecasts. The MIDAS approach can be viewed as a middle ground between the direct and the iterated approaches. Namely, one preserves the past high frequency data, to directly produce multi-period forecasts.

MIDAS is therefore a direct multi-step forecast device, in the sense that the left-hand side of the equation involves \( Y_{t+h}^Q \), refers to the period \( t + h \), directly. This is in contrast to many other forecasting approaches, such as autoregressive and state space models, where multi-step forecasts are done iteratively. To obtain projections for each forecast horizon \( h = 1, \ldots, H \), the left-hand side variable \( Y_{t+h}^Q \) the MIDAS equation to be estimated for each \( h \). Generalizing, say the ADL-MIDAS equation (2.9) to \( h \)-steps ahead forecasts, we get the ADL-MIDAS\((p_Y^Q, q_X^D)\) regression model given by:

\[
Y_{t+h}^Q = \mu(h) + \sum_{j=0}^{p_Y^Q-1} \mu_{j+1(h)} Y_{t-j}^Q + \sum_{j=0}^{p_Y^Q-1} \sum_{i=0}^{N_D-1} w_{N_D-i+j*N_D} \theta_{(h)}^{D} X_{N_D-i,t-j}^D + u_{t+h} \tag{2.10}
\]

where we index all the parameters by \( h \) to clarify that they change when one estimates the model for different multi-horizon forecasts.

### 2.4 Parameterizations the MIDAS polynomial weights

Various parameterizations for the polynomial lag structure appearing in equations such as (2.7), (2.9), (2.24), (2.22), (2.23), etc. have been used and discussed notably in Ghysels, Sinko, and Valkanov (2006). We will use \( N \) as the number of lags in the MIDAS polynomial - without specific reference to say daily lags. The specifications are as follows:

1. U-MIDAS (unrestricted MIDAS polynomial) approach suggested by Foroni, Marcellino, and Schumacher (2014) - where one estimates the individual coefficients unconstrained and therefore one can use a simple regression program. The U-MIDAS approach was shown to work for small values of \( N_D \). The prime example is
quarterly/monthly mixtures. U-MIDAS is a special case of MIDAS with step functions discussed below.

2. Normalized beta probability density function, unrestricted \((u)\) and restricted \((r)\) cases with non-zero and zero last lag. Please note that for specifications with a small number of MIDAS lags the zero-last-lag assumption may generate significant bias in the weighting scheme.

\[ w_{i}^{u,nz} = w_{i}(\theta_1, \theta_2, \theta_3) = \frac{x_i^{\theta_1-1}(1 - x_i)^{\theta_2-1}}{\sum_{i=1}^{N} x_i^{\theta_1-1}(1 - x_i)^{\theta_2-1}} + \theta_3 \]  

(2.11)

\[ w_{i}^{r,nz} = w_{i}(1, \theta_2, \theta_3) \]  

(2.12)

\[ w_{i}^{u,z} = w_{i}(\theta_1, \theta_2, 0) \]  

(2.13)

\[ w_{i}^{r,z} = w_{i}(1, \theta_2, 0) \]  

(2.14)

where \(x_i = (i - 1)/(N - 1)\)\(^2\)

3. Normalized exponential Almon lag polynomial

\[ w_{i}^{u} = w_{i}(\theta_1, \theta_2) = \frac{e^{\theta_1 i + \theta_2 i^2}}{\sum_{i=1}^{N} e^{\theta_1 i + \theta_2 i^2}} \]  

(2.15)

\[ w_{i}^{r} = w_{i}(\theta_1, 0) \]  

(2.16)

4. Almon lag polynomial specification of order \(P\) (not normalized, i.e. sum of individual weights is not equal to 1 and \(\beta w_{i}(\theta)\) from Eq. \(2.9\) are estimated jointly).

\[ \beta w_{i}(\theta_0, \ldots, \theta_P) = \sum_{p=0}^{P} \theta_p i^p \]  

(2.17)

\(^2\)To eliminate irregular behavior of the polynomial for some values of \(\theta\) at the ends of \([0,1]\) interval we use instead \(x_i = \text{eps} + (i - 1)/(N - 1)(1 - \text{eps})\), where \(\text{eps}\) is a machine 0 for MATLAB.
Note that this can also be written in matrix form:

\[
\begin{bmatrix}
w_0 \\
w_1 \\
w_2 \\
w_3 \\vdots \\
w_N \\
\end{bmatrix} =
\begin{bmatrix}
1 & 0 & 0 & \cdots & 0 \\
1 & 1 & 1 & \cdots & 1 \\
1 & 2 & 2^2 & \cdots & 2^P \\
1 & 3 & 3^2 & \cdots & 3^P \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
1 & N & N^2 & \cdots & N^P \\
\end{bmatrix}
\begin{bmatrix}
\theta_0 \\
\theta_1 \\
\vdots \\
\theta_P \\
\end{bmatrix}
\]  

(2.18)

Therefore the use of Almon lags in MIDAS models can be achieved via OLS estimation with properly transformed high frequency data regressors using the matrix representation appearing in the above equation. Once the weights are estimated via OLS, one can always rescale them to obtain a slope coefficient (assuming the weights do not sum up to zero).

5. Polynomial specification with step functions (not normalized)

\[
\beta w_i(\theta_1, \ldots, \theta_P) = \theta_1 I_{i \in [a_0,a_1]} + \sum_{p=2}^P \theta_p I_{i \in [a_{p-1},a_p]}
\]

\[a_0 = 1 < a_1 < \ldots < a_P = N \]  

(2.19)

\[
I_{i \in [a_{p-1},a_p]} = \begin{cases} 
1, & a_{p-1} \leq i \leq a_p \\
0, & otherwise 
\end{cases}
\]

where \(a_0 = 1 < a_1 < \ldots < a_P = N\).

We implicitly referred to model selection in the previous subsection when we mentioned ADL-MIDAS(AIC,AIC). Indeed, AIC/BIC criteria will be used to select optimal number of lags for the regressions, as discussed in detail Section 3.
2.5 Multiplicative MIDAS regressions

Recall that temporal aggregation imposes a fixed weighting scheme such as equal weights. What would happen if we instead consider a parameter-driven regressor:

\[
X_t^Q(\theta^D_X) \equiv \sum_{i=0}^{N_D-1} w_{N_D-i}(\theta^D_X)X_{N_D-i,t}^D
\]

where we scale weights again such that they add up to one. This will allow us to estimate a model we call multiplicative MIDAS or ADL-MIDAS-M\((p^Q_Y,p^Q_X)\):

\[
Y_{t+1}^Q = \mu + \sum_{k=0}^{p^Q_Y-1} \mu_k Y_{t-k}^Q + \sum_{k=0}^{p^Q_X-1} \beta_k X_{t-k}^Q(\theta^D_X) + u_{t+1}
\]

The above equation looks like a standard ADL model, except that it involves a parameter-driven regressor that mimics an aggregation scheme. There are pros and cons to the above approach. The pros are that the above multiplicative weighting scheme corresponds to the structure of a steady state Kalman filter with mixed data sampling, see Bai, Ghysels, and Wright (2012) for further details. Furthermore, the specification nests standard aggregation schemes and can also easily handle very complex aggregation schemes, as shown in Chen and Ghysels (2011b). The main disadvantage of the multiplicative MIDAS regression scheme is that it is not as parsimonious as the ADL-MIDAS in equation (2.9) which involves a single polynomial and therefore requires the estimation of a few parameters. Bai, Ghysels, and Wright (2012) in fact show that the differences in terms of RMSFE between standard ADL-MIDAS and the multiplicative specification are small, including cases where the multiplicative specification is an exact match of the steady state Kalman filter.

2.6 Factors and other regressors in ADL-MIDAS models

Recently, a large body of recent work has developed factor model techniques that are tailored to exploit a large cross-sectional dimension; see for instance, Bai and Ng (2002), Bai (2003), Forni, Hallin, Lippi, and Reichlin (2000), Forni, Hallin, Lippi, and Reichlin (2005), Stock and Watson (1989), Stock and Watson (2003), among many others. These factors are usually estimated at quarterly frequency using a large cross-section of time-series. Following this literature Andreou, Ghysels, and Kourtellos (2013) investigate whether
one can improve factor model forecasts by augmenting such models with high frequency information, especially daily financial data.

We therefore augment the aforementioned MIDAS models with factors, $F_t$, obtained by following dynamic factor model

$$
X_t = \Lambda_t F_t + u_t \tag{2.21}
$$

$$
F_t = \Phi F_{t-1} + \eta_t
$$

$$
u_{it} = a_{it}(L)u_{it-1} + \varepsilon_{it}, \quad i = 1, 2, ..., n
$$

where the number of factors is computed using criteria proposed by Bai and Ng (2002). The data used to implement the factor representation will be described in the next section. Suffice it here to say that we use series similar to those used by Stock and Watson (2008a).

Augmenting the MIDAS regression models from the previous subsection with the factors, we obtain a richer family of models that includes monthly frequency lagged dependent variable, quarterly factors, and a daily financial indicator. For instance, equation (2.9) generalizes to the FADL-MIDAS($p_F, p_Y, p_X^Q$) model:

$$
Y_{t+1}^Q = \mu + \sum_{i=0}^{p_F-1} \beta_i^Q F_{t-i}^Q + \sum_{j=0}^{p_Y^Q-1} \mu_j Y_{t-j}^Q + \beta \sum_{j=0}^{p_X^Q-1} \sum_{i=1}^{N_D} w_{N_D-i+j*N_D}(\theta^D)X_{N_D-i,t-j}^D + u_{t+1} \tag{2.22}
$$

where we use quarterly factors $F^Q$ to augment the ADL-MIDAS regression. Note that we can also formulate a FADL-MIDAS-M($p_F, p_Y^Q, p_X^Q$) model that involves a multiplicative MIDAS weighting scheme.

Equation (2.22) simplifies to the traditional factor model with additional regressors when the MIDAS features are turned off - i.e. say a flat aggregation scheme is used. When the lagged dependent variable is excluded then we have a projection on daily data, combined with aggregate factors.

It should finally be noted that we can add any low frequency regressor, not just factors. The software is written such that one can add any type of low frequency regressor.
2.7 MIDAS with leads

Nunes (2005) and Giannone, Reichlin, and Small (2008), among others, have formalized the process of updating the nowcast and forecasts as new releases of data become available. These studies typically use again state space setup. The process can be mimicked via MIDAS regression models with \textit{leads}. Say we are one or two months into quarter $t+1$. We consider MIDAS models with two months of daily lead data in order to incorporate real-time information available. Consider the Factor ADL model with MIDAS in equation \ref{eq:2.22}, which allows for $J^D_X$ daily leads for the daily predictor. Then we can specify the FADL-MIDAS($p^Q_Y, p^Q_F, p^Q_X, J^D_X$) model

$$
Y_{t+1}^Q = \mu + \sum_{i=0}^{p^F-1} \beta_i F_{t-1}^Q + \sum_{j=0}^{p^D_X-1} \mu_j Y_{t-j}^Q
$$

$$+eta \left[ \sum_{i=0}^{J^D_X-1} w_{i}^X \theta_X^{t-i} X_{t+1}^{D_X} + \sum_{i=0}^{J^D_X-1} w_{i}^X \theta_X^{t-i+j} \sum_{j=0}^{N^D_X-1} w_{j}^{N^D_X} \theta_X^{t-i+j} \right]
$$

$$+u_{t+1} \tag{2.23}$$

In the above equation, we use a single polynomial that carries the leads as well as the daily lags. In a multiplicative MIDAS scheme, we can write a model with leads as follows:

$$X_{t+1}^J(\theta_X) \equiv \sum_{i=0}^{J^D_X-1} w_{i}^X \theta_X^{t-i} X_{t+1}^{D_X}$$

where we scale weights again such that they add up to one. This will allow us to estimate a model we call multiplicative MIDAS or ADL-MIDAS-M($p^Q_Y, p^Q_X$):

$$Y_{t+1}^Q = \mu + \sum_{k=0}^{p^Q_X-1} \beta_k X_{t+1}^Q(\theta_X) + \sum_{k=0}^{p^Q_X-1} \beta_k X_{t-k}^Q(\theta_X) + u_{t+1} \tag{2.24}$$

Hence, the aggregation scheme is used via a partial sum for the lead days into month $t+1$.

To conclude it should be noted that two modes of forecasting can be used in the Matlab MIDAS Toolbox. The first is fixed in-sample estimation and fixed out-of-sample prediction and the second is a rolling window approach. For details, see Section 3.
2.8 Forecast combinations

There is a large literature on forecast combinations, see Timmermann (2006) for an excellent survey. Although there is a consensus that forecast combinations improve forecast accuracy there is no consensus concerning how to form the forecast weights.

Given the findings in Stock and Watson (2004), Stock and Watson (2008b) and Andreou, Ghysels, and Kourtellos (2013) we focus primarily on the Squared Discounted MSFE forecast combinations method, which delivers the highest forecast gains relative to other methods in many applications. The software also includes a BIC-based criterion as an option.

Let $\hat{y}_{i,t+h|t}$ denote the $i^{th}$ individual out-of-sample forecast of $y_{i,t+h|t}$ computed at date $t$. The forecast combination made at time $t$ is a (time-varying) weighted average of $n$ individual $h$-step ahead out-of-sample forecasts, $(\hat{y}_{1,t+h|t}, \ldots, \hat{y}_{n,t+h|t})$, given as:

$$f_{cM,t+h|t} = \sum_{i=1}^{n} w_{i,t}^h \hat{y}_{i,t+h|t}$$

(2.25)

where $(w_{1,t}^h, \ldots, w_{n,t}^h)$ is the vector of combination weights formed at time $t$ and $cM$ emphasizes the fact that the combined forecast depends on the class of models producing individual forecasts. A class of models is a collection of models involving either: (a) different high frequency series (the most common application) with each individual forecast $\hat{y}_{i,t+h|t}$ produced by an ADL-MIDAS regression involving the same type of polynomial and lag lengths for both the low and high frequency data, (b) different high frequency series with each individual forecast $\hat{y}_{i,t+h|t}$ produced by a ADL-MIDAS regression involving the different polynomial and lag lengths - for example selecting the best specification obtained with each individual series. In the latter case $\hat{y}_{i,t+h|t}$ and $\hat{y}_{j,t+h|t}$, for any $i$ and $j$, differ not only because of different high frequency series but also with regards to polynomial and/or lag lengths. In principle one could also consider forecast combinations involving the same high frequency series, but different polynomial and/or lag lengths. Finally, one could consider mega-combination simply combining all the series, all the polynomial specifications and with different lag lengths. Obviously the user has to define the class of models that are considered for the forecast combination exercise.

We consider four different weighting schemes:
- Equally weighted weights
  \[ w_{i,t} = \frac{1}{n} \]  

- BIC-weighted forecast
  \[ w_{i,t} = \frac{\exp(-BIC_i)}{\sum^n_{i=1} \exp(-BIC_i)} \]  

- MSFE-related model averaging:
  \[ w_{i,t} = \frac{m_{i,t}^{-1}}{\sum^n_{i=1} m_{i,t}^{-1}} \]
  \[ m_{i,t} = \sum_{t=T_0}^{T_0+h} \delta^{t-i}(y_{i,s+h} - \hat{y}_{i,s+h|s})^2 \]  

where \( T_0 \) is the first out-of-sample observation, \( \hat{y}_{i,s+h|s} \) – out-of-sample forecast, \( \delta \) – exponential averaging parameter.

1. MSFE averaging: \( \delta = 1 \)
2. DMSFE averaging: \( \delta = .9 \)

The BIC- and MSFE-based forecast combinations involve an estimation sample for all the models - involving either rolling windows or recursive window samples. In case of rolling windows, the user will have to specify the length of the window as well as the starting date. The BIC-weighted forecasts use the BIC from the latest available estimation sample. Hence, the forecast combination at time \( t \) for horizon \( h \) uses the BIC from the latest estimation sample - either rolling or recursive - with data up to time \( t \). For MSFE-related model averaging we need - in addition to the estimation sample - to define a forecast evaluation sample which is expressed in formula \( (2.28) \) as \( T_0 \) to \( T_0 + .t \). This means that the estimation sample ends in \( T_0 \). All the parameter estimates for the class of models are taken as given - they are produced by either the rolling or recursive sample with data until \( T_0 \) - and forecasts \( \hat{y}_{i,s+h|s} \) are produced over the sample starting \( \hat{y}_{i,T_0+h|T_0} \) until \( \hat{y}_{i,T_0+t+h|T_0+t} \). These h-step ahead forecasts yield a MFSE \( m_{i,t} \) for each member \( i \) of the class of models \( c_M \).

In a typical application, see e.g. Andreou, Ghysels, and Kourtellos (2013), involving quarterly data (low frequency) and either daily or monthly high frequency series, the estimation sample is usually 10 years (rolling sample) whereas the forecast evaluation sample is two years - or 8 quarters. This means that the first forecast combinations can be produced after 12 years.
(10 years for estimating the first models and 2 to appraise their out-of-sample performance). Then, for every additional quarter in the sample, one can update the estimates, produce new out-of-sample forecasts and finally generate additional forecast combinations.

2.9 Nuts and bolts issues

It is important to warn the user upfront that when creating data input files the dates need to be saved as text in Excel (American format). Any other format (even if it shows dates as mm/dd/yy) will not work. Other data formats will create errors which, on first sight, may appear unrelated to dates.

Different data providers have different data storing conventions. The approach we took is that the user is responsible for arranging the data in appropriate format. All that matters is that for each low frequency period there are $m$ high frequency data points and both high and low frequency data start and end at the same time.

We opted for the user to arrange the data properly rather than provide a general approach. Nevertheless, we briefly describe a typical situation encountered in MIDAS regression applications. Suppose quarterly data start in 1980Q1 and end in 2009Q2. Then the monthly data should start and 1980M01 and end at 2009M06. If there is insufficient data, i.e. some months at the beginning and/or the end are missing, NA values should be used. The 2009Q2 data should be aligned with 3 monthly observations 2009 M04, 2009 M05 and 2009 M06. Typically, the quarterly value of 2009Q2 becomes available after 2009M06. But this is a choice of the user. Ultimately, it is part of MIDAS regression models to specify which data is available at which time.

The historical data should be stored in a format compatible to the MIDAS toolbox. For instance the data input file of a quarterly sampled variable should look like the following:

<table>
<thead>
<tr>
<th>DATE</th>
<th>VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1947-03-01</td>
<td>237.2</td>
</tr>
<tr>
<td>1947-06-01</td>
<td>240.4</td>
</tr>
<tr>
<td>1947-09-01</td>
<td>244.5</td>
</tr>
<tr>
<td>1947-12-01</td>
<td>254.3</td>
</tr>
<tr>
<td>. . . . . .</td>
<td>. .</td>
</tr>
</tbody>
</table>
In this file, the field VALUE is the value of the input variable in the quarter starting with the month appearing in the DATE field. For example in the figure above, 14867.8 refers to the quarter 2011Q1. If you are using a different format of dating, you will need to align low frequency date to make sure the match with the high frequency data is correct. Similarly in a data input file of a monthly sampled variable, such as, date value

<table>
<thead>
<tr>
<th>DATE</th>
<th>VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1947-01-01</td>
<td>235.8</td>
</tr>
<tr>
<td>1947-02-01</td>
<td>250.3</td>
</tr>
<tr>
<td>1947-03-01</td>
<td>247.5</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>2010-02-01</td>
<td>456.0</td>
</tr>
<tr>
<td>2011-03-01</td>
<td>442.3</td>
</tr>
<tr>
<td>2011-04-01</td>
<td>473.6</td>
</tr>
</tbody>
</table>

the field VALUE is the value of the input variable in the month corresponding DATE field. Therefore in this table, 473.6 refers to 2011 M04.

In principle, you don’t need to have any other Matlab Toolbox to work with MIDAS Toolbox. There is only one simple m.file you may want to put into MIDAS Toolbox directory to be able to print plots. It is called suptitle.m, a function that puts a title above all subplots. If you receive a message stating that this m.file is missing, then please add it into your MIDAS Toolbox folder. It is available online.

Practical implementation of MIDAS involves issues that are typical for regression analysis, yet there are some not commonly encountered in standard regression problems and they pertain to the mixed sampling nature of the data.

Since the quarterly/daily combination has been used throughout this document, consider the situation of holidays occurring throughout a calendar year. This will create an unequal number of days on a quarter by quarter basis. While one can take different approaches towards this, we treat the holidays as missing values in the MIDAS polynomial. They will
be linearly interpolated using various schemes.

The algorithms can be grouped into (1) specifications with the same number of MIDAS lags each period and (2) specifications that cover the same time span each period.

Define a sequence of MIDAS polynomial weights \( w_{\tau_1}, w_{\tau_2}, \ldots \). Then we have the following:

1. Equally-spaced specification.
   
   (a) It is characterized by the fact that each observation point \( \{ y_t, X_{\text{factor}}_t, X_{\text{midas}}_t \} \) has the same number of MIDAS lags \( X_{\text{midas}}_t \). As a result, different periods may have different time span coverage but the same number of lags. The sequence of weights \( w_{\tau_1}, w_{\tau_{i+1}}, \ldots \) is defined in this case as \( w_i, w_{i+1}, \ldots \).

2. Real-time specifications. They are characterized by unequal number of MIDAS lags over time that cover the same time span.
   
   (a) Real time specification. The distance between \( w_{\tau_i} \) and \( w_{\tau_{i+1}} \) is proportionate to \( \tau_i - \tau_{i+1} \). No artificial observations are inserted in the MIDAS polynomial.
   
   (b) Real time specification with zeros at the end. Depending on the number of calendar days within a given time interval all missing days are added as zeros to the end of \( X_{\text{midas}} \) lag structure. MIDAS weights are constructed as in the equally-spaced case.\(^3\)

3 \ Software Usage

In this section we describe the implementation of data generation, estimation and forecast averaging algorithms for MIDAS regressions. The model of interest is

\[
Y_{t+s}^Q = \mu + \sum_{k=1}^{p_x-1} \sum_{i=0}^{p_x-1} \beta_{i,k} F_{t-i,k}^Q + \sum_{j=0}^{p_y-1} \mu_j Y_{t-j}^Q + \beta \sum_{j=0}^{p_y-1} \sum_{i=0}^{p_y-1} w_{N_D-i+j} N_D (\theta^D) X_{N_D-i,t-j}^D + u_{t+s} \tag{3.1}
\]

Currently multiplicative MIDAS is not implemented \( (p_x^D = 1) \). MIDAS with leads specification is implemented via defining proper values of \( s \).

\(^3\)Please note that normalization of the polynomial in this case is different from the equally-spaced specification.
To use the MIDAS package, first prepare the mixed frequency data: DataY, DataYdate, DataX, DataXdate. As the name suggests, DataY is the low frequency dependent variable data specified as a column vector. DataYdate indicates the dates corresponding to the low frequency observations. A variety of date formats are supported. For instance, '1985-01-01', '01/01/1985', 'January 1, 1985' are all legitimate dates. DataYdate is a cell array in which each element is a date string. Similarly, DataX and DataXdate are the high frequency data and dates.

The function MIDAS_ADL.m in the software package is the gateway to the MIDAS regression. The required input arguments are DataY, DataYdate, DataX, DataXdate. In addition, optional input arguments are specified as name-value pairs, which detail the mixed frequency model specifications. The options include:

- 'Xlag': the number of lagged the high frequency explanatory variables. It can be a scalar or descriptive string such as '3m','1q'. The default value is 9, which means that the explanatory variables include 9 lagged high frequency variables.

- 'Ylag': the number of lags of the autoregressive low frequency variables. It can be a scalar or descriptive string such as '3m','1q'. The default value is 1, which means that the predictors also include a lagged low frequency variable.

- 'Horizon': MIDAS lead/lag specification. It can be a scalar or descriptive string such as '3m','1q'. The default value is 1, which implies that dependent variables in period $t$ is accompanied by high frequency regressors in period $t - 1, t - 2$, etc. If 'Horizon' is reset to 2, dependent variables in period $t$ will be regressed on high frequency regressors in period $t - 2, t - 3$, etc. A negative integer value of 'Horizon' is also supported. In that case, it is a MIDAS with leads of high frequency regressors. Proper setting of 'Horizon' can offset the impact of different date styles of the low frequency data. For example, if the quarterly dates are coded as '01/01/1985', 'Horizon' = 1 implies that lagged high frequency monthly regressors start from '12/01/1984'. However, if the same quarterly data is recorded as '03/01/1985' instead, 'Horizon' can be set to 3 so that the lagged high frequency data still start from '12/01/1984'. In case of any confusion on the regression dates, refer to the time frame displayed on the screen.

- 'EstStart': start date of the estimation window, specified as a date string. By default, estimation starts from the beginning of the sample, adjusted by lagged values. It is
illegal to set the 'EstStart' out of the sample range. In that case, the program will explain the earliest date that can be supported.

- 'EstEnd': terminal date of the estimation window, specified as a date string. By default, estimation terminates at the end of the sample, adjusted by the 'Horizon' value. If 'EstEnd' is earlier than the (adjusted) last observation date, out-of-sample forecast will be performed and the forecast values will be compared with the unused observations. Best practice is to leave some observations for the out-of-sample forecast, which provides some assessment of the model performance.

- 'ExoReg': Exogenous low-frequency regressors specified as a T-by-k matrix, where T is the length of the data, k is the number of exogenous regressors. The frequency of exogenous regressors must be the same as the low frequency dependent variable DataY. The sample size must be at least as large as DataY. Do not include a constant, for it is automatically added to the regression. For instance, if the MIDAS is augmented by known factors, 'ExoReg' accommodates the factors data.

- 'ExoRegDate': Dates associated with exogenous regressors data specified as a T-by-1 cell array in which each element is a date string. All exogenous regressors share the same dates.

- 'Method': an option for estimation methods. Its value can be
  - 'FixedWindow' (default): Estimation window is defined as [estStart, estEnd]. Then the multi-step forecast values are compared with the unused observations.
  - 'RollingWindow': Multiple windows are defined as [estStart+i, estEnd+i]. Then the one-step forecast value is compared with the observations in estEnd+i+1.
  - 'Recursive': Multiple windows are defined as [estStart, estEnd+i]. Then the one-step forecast value is compared with the observations in estEnd+i+1.

- 'Polynomial': functional form of the MIDAS weights. Its value can be
  - 'Beta' (default): Normalized beta density with a zero last lag
  - 'BetaNN': Normalized beta density with a non-zero last lag
  - 'ExpAlmon': Normalized exponential Almon lag polynomial
  - 'UMIDAS': Unrestricted coefficients
- 'Step': Polynomial with step functions
- 'Almon': Almon lag polynomial of order p

- 'PolyStepFun': thresholds of the step function. This option is relevant only if 'Polynomial' is set to 'Step'.

- 'AlmonDegree': number of lags of the Almon lag. This option is relevant only if 'Polynomial' is set to 'Almon'.

- 'Discount': discount factor to compute the discounted mean squared error of forecast. The default value is 0.9.

- 'Display': the screen display style. Its value can be
  - 'full' (default): full display of the regression time frame, and the estimator summary
  - 'time': display of the regression time frame
  - 'estimate': display of the estimator summary
  - 'off': no display on the screen

- 'PlotWeights': logical value indicating whether to plot the MIDAS weights after parameter estimation. The default is true.

When the function MIDAS_ADL.m is called, it will first parse the mixed frequency data and model specifications. Intermediate results are stored in a struct array called 'MixedFreqData'. After that stage, a MIDAS regression is well defined and nonlinear least squares is employed to obtain the estimated model parameters. The estimation results are stored in a struct array called 'OutputEstimate'. Lastly, if the 'EstEnd' is earlier than the last observation, out-of-sample forecast is performed. The forecast values are compared with the realized values so as to evaluate the forecasting power of the model. The forecast results are stored in a struct array called 'OutputForecast'.

'OutputForecast' includes the following fields:

- Yf: point forecast of the low frequency data after 'EstEnd'
- RMSE: root mean squared error of forecast
• MSFE: mean squared error of forecast
• DMSFE: discounted mean squared error of forecast
• aic: Akaike information criteria of the regression (a copy from OutputEstimate)
• bic: Bayesian information criteria of the regression (a copy from OutputEstimate)

'OutputEstimate' includes the following fields:

• model: description of the MIDAS weight polynomial
• paramName: description of the model parameters
• estParams: estimated parameters
• EstParamsCov: covariance matrix of the estimated parameters
• se: standard errors of the estimated parameters
• tstat: t statistics of the estimated parameters
• sigma2: disturbance variance of the mixed frequency regression
• yfit: fitted low frequency data
• resid: residual of the mixed frequency regression
• estWeights: estimated coefficients of high frequency regressors (weights)
• logL: log likelihood of the low frequency data
• r2: R2 statistics of the regression
• aic: Akaike information criteria of the regression
• bic: Bayesian information criteria of the regression

'MixedFreqData' includes the following fields:

• EstY: low frequency data in the estimation periods, a T1-by-1 vector
• EstYdate: dates of low frequency data in the estimation periods, a T1-by-1 vector of MATLAB serial date numbers

• EstX: high frequency data in the estimation periods, a T1-by-Xlag matrix

• EstXdate: dates of high frequency data in the estimation periods, a T1-by-Xlag matrix of MATLAB serial date numbers

• EstLagY: low frequency lagged regressors in the estimation periods, a T1-by-Ylag matrix

• EstLagYdate: dates of low frequency lagged regressors in the estimation periods, a T1-by-Ylag matrix of MATLAB serial date numbers

• OutY: low frequency data in the forecasting periods, a T2-by-1 vector

• OutYdate: dates of low frequency data in the forecasting periods, a T2-by-1 vector of MATLAB serial date numbers

• OutX: high frequency data in the forecasting periods, a T2-by-Xlag matrix

• OutXdate: dates of high frequency data in the forecasting periods, a T2-by-Xlag matrix of MATLAB serial date numbers

• OutLagY: low frequency lagged regressors in the forecasting periods, a T2-by-Ylag matrix

• OutLagYdate: dates of low frequency lagged regressors in the forecasting periods, a T2-by-Ylag matrix of MATLAB serial date numbers

• Xlag: number of lagged the high frequency explanatory variables, in numerical format

• Ylag: number of lagged the low frequency explanatory variables, in numerical format

We revisit some of the examples in Armesto, Engemann, and Owyang (2010). In particular we run ADL-MIDAS regressions to forecast GDP growth with monthly employment growth. Seasonally adjusted real GDP quarterly data are taken from St. Louis FRED website and the real GDP growth is computed as log-quarterly first difference. Monthly total employment non-farm payrolls data are also taken from FRED and log-monthly first differences are computed.
The data are stored in the spreadsheet 'mydata.xlsx'. First, we load the data:

```matlab
[DataY, DataYdate] = xlsread('mydata.xlsx', 'sheet1');
DataYdate = DataYdate(2:end, 1);
[DataX, DataXdate] = xlsread('mydata.xlsx', 'sheet2');
DataXdate = DataXdate(2:end, 1);

DataXgrowth = log(DataX(2:end)./DataX(1:end-1))*100;
DataYgrowth = log(DataY(2:end)./DataY(1:end-1))*100;
DataX = DataXgrowth;
DataY = DataYgrowth;
DataYdate = DataYdate(2:end);
DataXdate = DataXdate(2:end);
```

Then we estimate the model with a variety of weight polynomials by calling the function `MIDAS_ADLM.m`. Note that all optional input arguments have default values. We use verbose syntax for illustration of those name-value pairs.

```matlab
Xlag = 9;
Ylag = 1;
Horizon = 3;
EstStart = '1985-01-01';
EstEnd = '2009-01-01';
Method = 'fixedWindow';

[OutputForecast1, OutputEstimate1, MixedFreqData] = MIDAS_ADLM(DataY, DataYdate, DataX, DataXdate,...
    'Xlag', Xlag, 'Ylag', Ylag, 'Horizon', Horizon, 'EstStart', EstStart, 'EstEnd',...%
    'EstEnd', 'Polynomial', 'beta', 'Method', Method, 'Display', 'full');
[OutputForecast2, OutputEstimate2] = MIDAS_ADLM(DataY, DataYdate, DataX, DataXdate,...
    'Xlag', Xlag, 'Ylag', Ylag, 'Horizon', Horizon, 'EstStart', EstStart, 'EstEnd',...%
    'Polynomial', 'betaNN', 'Method', Method, 'Display', 'estimate');
[OutputForecast3, OutputEstimate3] = MIDAS_ADLM(DataY, DataYdate, DataX, DataXdate,...
    'Xlag', Xlag, 'Ylag', Ylag, 'Horizon', Horizon, 'EstStart', EstStart, 'EstEnd',...%
    'Polynomial', 'expAlmon', 'Method', Method, 'Display', 'estimate');
[OutputForecast4, OutputEstimate4] = MIDAS_ADLM(DataY, DataYdate, DataX, DataXdate,...
    'Xlag', Xlag, 'Ylag', Ylag, 'Horizon', Horizon, 'EstStart', EstStart, 'EstEnd',...%
    'Polynomial', 'umidas', 'Method', Method, 'Display', 'estimate');
[OutputForecast5, OutputEstimate5] = MIDAS_ADLM(DataY, DataYdate, DataX, DataXdate,...
    'Xlag', Xlag, 'Ylag', Ylag, 'Horizon', Horizon, 'EstStart', EstStart, 'EstEnd',...%
```
In the full display mode, the time frame of the regression is shown on the screen, which helps to verify the mixed frequency date specification. The estimation results will also be reported on the screen. Occasionally, numerical optimization routine does not yield convergent results and it is possible that the returned estimator covariance matrix is not positive definite. In that case, model specification should be carefully reviewed. Diagnostics and new proposals might be in need.

| Frequency of Data Y: 3 month(s) |
| Frequency of Data X: 1 month(s) |
| Start Date: 01-Jan-1985 |
| Terminal Date: 01-Jan-2009 |

Mixed frequency regression time frame:
- Reg Y(01/01/85) on Y(10/01/84), X(10/01/84), X(09/01/84), ..., X(02/01/84)
- Reg Y(04/01/85) on Y(01/01/85), X(01/01/85), X(12/01/84), ..., X(05/01/84)
- ...
- Reg Y(01/01/09) on Y(10/01/08), X(10/01/08), X(09/01/08), ..., X(02/01/08)

MIDAS: Normalized beta density with a zero last lag

| 'Const' | [ 0.6656] | [0.1353] | [ 4.9184] |
| 'HighFreqSlope' | [ 1.9121] | [0.5592] | [ 3.4190] |
| 'Beta1' | [ 0.9904] | [0.0672] | [14.7435] |
| 'Beta2' | [ 6.6157] | [9.6620] | [ 0.6847] |
| 'Ylag1' | [ 0.2847] | [0.1156] | [ 2.4619] |

Since the estimation sample runs from 1985-01-01 to 2009-01-01 and the data for GDP growth in the example runs until the second quarter of 2011, there are nine quarters left for the out-of-sample evaluation. By extracting the RMSE of each model, we can compare their forecasting power:

```matlab
fprintf('RMSE Beta: %.4f
',OutputForecast1.RMSE);
```
In this example, the weight function of the normalized beta density with a non-zero last lag outperforms other models, though other weight specifications are not obviously inferior.

<table>
<thead>
<tr>
<th>Weight Specification</th>
<th>RMSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Beta</td>
<td>0.5650</td>
</tr>
<tr>
<td>Beta Non-Zero</td>
<td>0.5210</td>
</tr>
<tr>
<td>Exp Almon</td>
<td>0.5641</td>
</tr>
<tr>
<td>U-MIDAS</td>
<td>0.5424</td>
</tr>
<tr>
<td>Stepfun</td>
<td>0.5252</td>
</tr>
<tr>
<td>Almon</td>
<td>0.5329</td>
</tr>
</tbody>
</table>

Though the function MIDAS_ADL.m can plot the weights by setting the name-value pair 'PlotWeights', it is more desirable to have multiple curves in one figure for comparison. So we extract the weights from the estimation output and plot them manually.

```
Xlag = MixedFreqData.Xlag;
for m = 1:6
    weights = eval(['OutputEstimate',num2str(m),'.estWeights']);
    subplot(2,3,m);plot(1:Xlag,weights);title(['Model ',num2str(m)])
end
```

Users are encouraged to modify the model specification and see how the estimation/forecast results change accordingly. For example, we can slightly tweak the program to make it suitable for nowcasting. We estimate an ADL-MIDAS with two months of leads. If we reset 'Horizon' to 1, we will be forecasting with one month horizon rather than one quarter (we changed 1q to 1m).
Reg Y(01/01/09) on Y(10/01/08), X(12/01/08), X(11/01/08), ..., X(04/01/08)

| RMSE Beta: | 0.5214 |
| RMSE Beta Non-Zero: | 0.5176 |
| RMSE Exp Almon: | 0.5238 |
| RMSE U-MIDAS: | 0.5150 |
| RMSE Stepfun: | 0.5244 |
| RMSE Almon: | 0.5041 |

Note that we have made improvements in the RMSE across all polynomial specifications with the two extra months of information. The output structure allows one to appraise the new forecasts, parameter estimates, etc.

We turn our attention to the recursive estimation by setting the name-value pair 'Method', 'rollingwindow'. When either rolling or recursive estimation is chosen, the program re-estimates the model recursively. At each iteration, the program produces a rolling or recursive estimation/forecast of one step ahead. Substantial improvement are made in the recursive updates of the parameter estimates.

| RMSE Beta: | 0.3146 |
| RMSE Beta Non-Zero: | 0.3311 |
| RMSE Exp Almon: | 0.3280 |
| RMSE U-MIDAS: | 0.3272 |
| RMSE Stepfun: | 0.3245 |
| RMSE Almon: | 0.3376 |

Finally, we consider the model averaging by adding the industrial production as a second high frequency series. In the first model, we use the monthly total employment non-farm payrolls to predict GDP growth, while the second model uses the industrial production as the high frequency predictors. With two sets of forecast outputs, we use the function ForecastCombine.m to combine the forecast according to the MSFE, MSFE, aic/bic and flat weights respectively.

\[ YfMSFE = \text{ForecastCombine}(\text{OutputForecast1}, \text{OutputForecast2}); \]
\[ YfDMSFE = \text{ForecastCombine}(\text{OutputForecast1}, \text{OutputForecast2}, 'DMSFE'); \]
4 Other MIDAS Applications

Ghysels (2012) introduces a relatively simple mixed sampling frequency VAR model. By simple we mean, (1) a specification that does not involve latent shocks, (2) a specification that allows us to measure the impact of high frequency data onto low frequency ones and vice versa, (3) as far as VAR models go parsimonious, (4) a specification that can be estimated and analyzed with standard VAR analysis tools - such as impulse response analysis, and can be estimated with standard VAR estimation procedure (5) one that can track the proper timing of low and high frequency data - that may include releases of quarterly data in the middle of the next quarter along with the releases of monthly data or daily data.
The mixed frequency VAR provides an alternative to commonly used state space models involving mixed frequency data\footnote{See for example, Harvey and Pierse (1984), Bernanke, Gertler, and Watson (1997), Zadrozny (1990), Mariano and Murasawa (2003), Mittnik and Zadrozny (2004), and more recently Aruoba, Diebold, and Scotti (2009), Ghysels and Wright (2009), Kuzin, Marcellino, and Schumacher (2009), Marcellino and Schumacher (2010), among others.}. State space models involve latent processes, and therefore rely on filtering to extract hidden states that are used in order to predict future outcomes. State space models are, using the terminology of Cox (1981), parameter-driven models. The mixed frequency VAR models are, using again the same terminology, observation-driven models as they are formulated exclusively in terms of observable data. The fact we rely only on observable shocks has implications with respect to impulse response functions. Namely, we formulate impulse response functions in terms of observable data - high and low frequency - instead of shocks to some latent processes. Finally, mixed frequency VAR models, like MIDAS regressions, may be relatively frugal in terms of parameterization.

Technically speaking Ghysels (2012) adapts techniques typically used to study seasonal time series with hidden periodic structures, to multiple time series that have different sampling frequencies. The techniques we adapt relate to work by Gladyshev (1961), Pagano (1978), Tiao and Grupe (1980), Hansen and Sargent (1990, Chap. 17), Hansen and Sargent (1993), Ghysels (1994), Franses (1996), among others. In addition, the mixed frequency VAR model is a multivariate extension of MIDAS regressions proposed in recent work by Ghysels, Santa-Clara, and Valkanov (2006), Ghysels and Wright (2009), Andreou, Ghysels, and Kourtellos (2010) and Chen and Ghysels (2011b), among others.

Ghysels (2012) also characterizes the mapping between the mixed frequency VAR model and (1) a traditional VAR model where all the data are sampled at a common low frequency as well as (2) a hidden state high frequency VAR commonly used in a state space model setting. This mapping allows us to study the mis-specification of impulse response functions of traditional VAR models. The VAR models we propose can also handle time-varying mixed frequencies. Not all months have the same number of trading days, not all quarters have the same number of weeks, etc. Assuming a deterministic calendar effect, which makes all variation in changing mixed frequencies perfectly predictable, we are able to write a VAR with time-varying high frequency data structures.

Ghysels (2012) studies two classes of estimation procedures, classical and Bayesian, for mixed frequency VAR models. For the former Ghysels (2012) characterizes how the mis-specification of traditional VAR models translates into pseudo-true VAR parameter and
impulse response estimates. Parameter proliferation is an issue in both mixed frequency and traditional VAR models. A Bayesian approach which easily accommodates the potentially large set of parameters to be estimated is therefore also considered. While this software is not yet featured in the Tooblox, the programs are available upon request.

Econometric analysis of mixed frequency models is not limited to linear regressions. The purpose of this concluding section is to briefly discuss other applications of MIDAS beyond linear regression analysis.

Chen and Ghysels (2011b) introduce semi-parametric estimation of MIDAS regression models. They consider a regression model with a MIDAS polynomial (the parametric part) involving functional transformations of high frequency data, where the function is estimated via kernel methods.

The initial work on MIDAS and volatility involved a likelihood based approach on the risk return tradeoff. In particular the monthly variance is specified in Ghysels, Santa-Clara, and Valkanov (2005) as a weighted average of lagged daily squared returns and estimated via a QMLE similar a GARCH-in-mean approach. Hence, they estimate the coefficients of the conditional variance process jointly with the expected return equation. Hence, this approach is very different from the MIDAS regressions discussed in the previous sections. The similarity, however, is that in both MIDAS regressions and in the likelihood based MIDAS specification use the same type of parsimoniously specified lag polynomials.

The volatility specification in Ghysels, Santa-Clara, and Valkanov (2005) involves a single polynomial applied to daily data. Similar to the specification of the MIDAS regression one could think of introducing lagged volatilities. This approach would be similar to the specification of a GARCH model. This insight has recently been pursued by Chen, Ghysels, and Wang (2010). A key ingredient of conditional volatility models is that more weight is attached to the most recent returns (i.e. information). The model, however, involves returns sampled at different frequencies. For example, daily volatility can be predicted using intra-daily data. However, unlike realized volatility measures, intra-daily returns get different weights. Indeed, if volatility is a persistent process, it would be natural to weight intra-daily data differently. This is one example of the class of models Chen, Ghysels, and Wang (2010) called HYBRID GARCH models. They provide a unifying framework, based on a generic GARCH-type model, that addresses the issue of volatility forecasting involving forecast horizons of a different frequency than the information set. Hence, they propose a
class of GARCH models that can handle volatility forecasts over the next five business days and use past daily data, or tomorrows expected volatility while using intra- daily returns. The models are called HYBRID GARCH, which stands for High FrequencY Data Based PRojectIon Driven GARCH models.

Engle, Ghysels, and Sohn (2012) revisit modeling the economic sources of volatility. They consider a component model and suggest several new component model specifications with direct links to economic activity. Practically speaking, the research pursued is inspired by (1) Engle and Rangel (2008) who introduce a Spline-GARCH model where the daily equity volatility is a product of a slowly varying deterministic component and a mean reverting unit GARCH and (2) the use of MIDAS approach to link macroeconomic variables to the long term component of volatility. Hence, the new class of models is called GARCH-MIDAS, since it uses a mean reverting unit daily GARCH process, similar to Engle and Rangel (2008), and a MIDAS polynomial which applies to monthly, quarterly, or bi-annual macroeconomic or financial variables. Having introduced the GARCH-MIDAS model that allows us to extract two components of volatility, one pertaining to short term fluctuations, the other pertaining to a long run component, we are ready to revisit the relationship between stock market volatility and economic activity and volatility. The first specification we consider uses exclusively financial series. The GARCH component is based on daily (squared) returns, whereas the long term component is based on realized volatilities computed over a monthly, quarterly or bi-annual basis. The GARCH-MIDAS model also allows us to examine directly the macro-volatility links. Indeed, one can estimate GARCH-MIDAS models where macroeconomic variables enter directly the specification of the long term component. The fact that the macroeconomic series are sampled at a different frequency is not an obstacle, again due to the advantages of the MIDAS weighting scheme.

In addition, dynamic correlation models featuring mixed data sampling schemes based on MIDAS have been used by Colacito, Engle, and Ghysels (2011) and Baele, Bekaert, and Inghelbrecht (2010).
References


Bai, J., and S. Ng, 2002, Determining the number of factors in approximate factor models, Econometrica pp. 191–221.


Mittnik, Stefan, and Peter A. Zadrozny, 2004, *Forecasting quarterly German GDP at monthly intervals using monthly Ifo business conditions data* (Springer).


———, 2008a, Forecasting in Dynamic Factor Models Subject to Structural Instability, in Jennifer Castle, and Neil Shephard, ed.: *The Methodology and Practice of Econometrics, A Festschrift in Honour of Professor David F. Hendry*. Oxford University Press.


