gretl/MIDAS: a supplement

Allin Cottrell

May 23, 2017

1 Introduction

The document titled "MIDAS in gretl" (hereinafter referred to as *MiG*) gives an account of the support offered by gretl for MIDAS (Mixed Data Sampling) modeling.[1] This supplement explores some questions regarding the practicalities and effectiveness of MIDAS regression as a forecasting tool. We address two main sets of questions, with particular focus on forecasting US GDP.

- Given the timing with which relevant data become available, what are the options for forecasting at different horizons, and what are the implications for the lag structure of the models one may use?
- What choices of high-frequency data and MIDAS parameterization give the best forecasting performance? And how do MIDAS-based forecasts compare with simpler methods that use data of a single frequency?

These are big questions and we cannot explore them fully here, let alone offer definitive answers. Nonetheless we hope that the arguments and findings herein may be of some use to people interested in making practical use of MIDAS.

2 Data timing and lag structure

Let us define the "data lag" for a given series as the lag between the end of a period and the first publication of data pertaining to that period.

From inspection of the release schedules of the Bureau of Labor Statistics (BLS), Bureau of Economic Analysis (BEA) and Board of Governors of the Federal Reserve System (Fed), the following table of approximate data lags can be drawn up for some commonly referenced US macro time series (PAYEMS = payroll employment, INDPRO = industrial production):

series	source	frequency	approx lag
CPI	BLS	monthly	2 weeks
PAYEMS	BLS	monthly	1 week
INDPRO	Fed	monthly	2 weeks
GDP	BEA	quarterly	4 weeks

[1] See http://gretl.sourceforge.net/midas/midas_gretl.pdf.

In addition, according to the BEA the second estimate of quarterly GDP follows the first ("advance") estimate by about 4 weeks, and the third estimate follows the second by around another 4 weeks.

We focus below on forecasting GDP and so we drop the CPI from current consideration. A full analysis of real-time forecasting would require looking into the revision schedule for the monthly data as well as for GDP, but we omit that for simplicity. Table 1 shows a stylized data schedule for a given quarter, Q_t. We take this quarter as "the present" (lag 0); an "h" against a lag indicates a high-frequency (HF) lag (refer to *MiG* for details).

Month	end week	latest data	lag
1	1	PAYEMS Q_{t-1} month 3	1h
	2	INDPRO Q_{t-1} month 3	1h
	4	est. 1, GDP Q_{t-1}	1
2	1	PAYEMS Q_t, month 1	0h
	2	INDPRO Q_t, month 1	0h
	4	est. 2, GDP Q_{t-1}	1
3	1	PAYEMS Q_t, month 2	-1h
	2	INDPRO Q_t, month 2	-1h
	4	est. 3, GDP Q_{t-1}	1

Table 1: Data arrival schedule for a given quarter

Suppose we have in mind a simple ADL-MIDAS model (in log differences) for GDP, which includes the first lag of GDP and HF lags 1 to p of INDPRO. We may write this as

$$y_t = \alpha + \beta y_{t-1} + \gamma W(x_{\tau-1}, x_{\tau-2}, \ldots, x_{\tau-p}; \theta) + \varepsilon_t \qquad (1)$$

where $W(\cdot)$ is the weighting function corresponding to the chosen MIDAS parameterization, θ is a vector of hyperparameters, and τ indicates "high frequency time." The forecast from this model is then

$$\hat{y}_t = \hat{\alpha} + \hat{\beta} y_{t-1} + \hat{\gamma} W(x_{\tau-1}, x_{\tau-2}, \ldots, x_{\tau-p}; \hat{\theta})$$

Suppose for the moment that in order to generate a forecast we require actual published values (even if these be "advance" data) for all the regressors. Then the schedule shows that we will not be able to generate a forecast of quarter Q_t's GDP until the end of week 4 of the quarter, when the advance estimate of the prior quarter's GDP becomes available. We may call this a "nowcast," but note that it will precede the end of the period by 2 months (and precede the advance estimate for Q_t from the BEA by 3 months).

We can also see from the data schedule that an updated nowcast could be produced at various points during the quarter. Most obviously, a fitted value based on estimation of (1) could be recalculated using the revised GDP figures for Q_{t-1} available towards the end of months 2 and 3. But in addition a second model whose HF lags start at 0 could produce a nowcast incorporating the new INDPRO information that arrives in month 2:

$$y_t = \alpha_1 + \beta_1 y_{t-1} + \gamma_1 W(x_\tau, x_{\tau-1}, \ldots, x_{\tau-p+1}; \theta_1) + \eta_t \qquad (2)$$

Similarly, a model whose HF lags start at -1 could use INDPRO from month 3 to produce a further update.

We now consider relaxing the requirement that our forecasting equation take nothing but "actual data" as input. Since the equation forecasts GDP using lagged GDP, in principle the forecast could be chained: based on observed GDP for Q_{t-1} we could produce an estimate for Q_t and then use this in the equation for Q_{t+1}, before the Q_t datum is published. However, expanding the forecast horizon in this way would require that either

- we somehow obtain forecasts for the high frequency data too, or
- we revise the model to employ less recent lags of the high frequency data.

We leave the first of these possibilities aside, since we have no model to forecast the high frequency series in view at present. Taking up the second possibility, let us look back at Table 1. In week 4 of Q_t we can form a first nowcast of Q_t's GDP, but at this point the most recent INDPRO datum is from month 3 of Q_{t-1}, that is, at HF lag 4 relative to the equation for Q_{t+1}. So the model for forecasting GDP "truly ahead" would have to be estimated with a minimum HF lag of 4:

$$y_t = \alpha_2 + \beta_2 y_{t-1} + \gamma_2 W(x_{\tau-4}, x_{\tau-5}, \ldots, x_{\tau-q}; \theta_2) + \nu_t \quad (3)$$

We could then calculate at the end of month 1 of Q_t

$$\hat{y}_{t+1} = \hat{\alpha}_2 + \hat{\beta}_2 \tilde{y}_t + \hat{\gamma}_2 W(x_{\tau-1}, x_{\tau-2}, \ldots, x_{\tau-q+3}; \hat{\theta}_2)$$

with $\tilde{y}_t$ obtained as fitted value from (1).

ADL examples in the MIDAS Matlab Toolbox

In light of the above, consider the ADL-MIDAS examples given in Eric Ghysels' MIDAS Matlab Toolbox. These forecast the log difference of GDP (y) using the first lag of the dependent variable and HF lags 3 to 11 of the log difference of monthly payroll employment (x). As discussed in *MiG*, HF lag 3 of a monthly series in a quarterly context means (for both Matlab and gretl) the third-to-last month of the previous quarter. For example, if we're in Q1 of some year, HF lag 3 corresponds to October of the previous year.

The Toolbox forecasts are static (no chaining). So if a datum for y_{t-1} is to be used, then, as argued above, forecasts (in fact, nowcasts) cannot be produced until about 4 weeks into the quarter. It therefore seems a bit odd to be using $x_{\tau-3}$ as the most recent monthly lag, since a published value for $x_{\tau-1}$ will surely be available at that point. One could perhaps argue that more recent x data are too noisy to be useful (being early estimates). Or we could think of the lag scheme as a set-up for a dynamic forecast (though the forecasts are not in fact dynamic).

The examples in question are clearly intended as illustrations of the capability of the software rather than full-blown realistic exercises in forecasting, so there is no call to be pedantic in assessing the specification. Nonetheless, there's good reason to consider alternative lag schemes in the experiments that follow rather than simply following the precedent set by Ghysels.

3 Forecast comparisons

In this section we present forecast comparisons along 5 dimensions. This invites combinatorial explosion so we limit ourselves to relatively few "tics" in most of the dimensions, as follows.

1. *Choice of high-frequency regressor*. We consider two candidate monthly series, the Fed's Index of Industrial Production (INDPRO) and non-form Payroll Employment (PAYEMS) from the BLS.

2. *Parameterization*. We start with four MIDAS parameterizations and two single-frequency alternatives. The MIDAS variants are:

Beta 2	Two-parameter normalized beta
Beta 1	As Beta 2 but with θ_1 clamped at 1.0
NEAlmon	Normalized exponential Almon with 2 parameters
Almon poly	Almon polynomial of order 4

   The single-frequency alternatives are:

AR(1)	OLS with regressors constant and y_{t-1}
ARMA(1,1)	Exact ML, including a constant

3. *MIDAS lags*. In the MIDAS specifications we fix on 10 lags of the high-frequency variable, but we compare results between lags 1 to 10 and lags 0 to 9. (We also mention but do not report in detail comparisons with the scheme of lags 3 to 11 as used in the MIDAS Matlab Toolbox examples.)

4. *Estimation sample size (T)*. In general, of course, the more data the better. But if there are structural breaks—or less dramatically, structural drift—it may be that a shorter sample yields more accurate forecasts. We allow T to vary between 60 and 120 quarters.

5. *Forecast start date*. Results are liable to differ depending on the particular historical stretch of data under consideration. We initially employ four different starting points for forecasts: 2000Q1, 2005Q1, 2010Q1 and 2015Q1.

The data: realtime or hindsight?

Our forecasting exercises involve selecting some past period as the target for forecasts, and estimating the competing models on data that pertain to periods prior to the start of the forecast period. One possibility here would be to use "realtime" data for estimation—that is, the data that would have been available to an econometrician at the specified point in time. The alternative is to use "hindsight"—that is, hypothetically to provide the econometrician in, say, 1999Q4 with the latest revision of the relevant data as of the time of performing the exercise (in this case, May 2017). A similar choice arises with respect to the calculation of forecast errors: should these be assessed relative to the first publication of the relevant GDP data, or relative to our current best estimate of GDP at the time?

While a case can certainly be made for the realtime approach (and we talk about this briefly in the final section of this document), we go with the hindsight approach below. Partly this is just for convenience—it means that we don't have to construct a different archival dataset for each forecast period we consider—but also partly because we suspect that we *may* be able to resolve more sharply the differential effectiveness of forecasting methods if we "net out" the noise associated with early data releases. However, we recognize that insofar as we use high-frequency data with a slightly more recent observation date in some of our MIDAS formulations, this approach risks biasing the comparison in favor of MIDAS over the single-frequency alternatives. This is therefore a point which should be revisited in further research.

Forecast assessment plots

We start the comparison by examining plots (Figures 1, 2 and 3) which are constructed as follows. We choose a particular high-frequency predictor, a particular MIDAS lag set and a particular forecast start date. We then run an iteration across sample size: for each T we estimate each of the six models mentioned above, generate 8 static forecasts, and record the RMSE. Each plot shows RMSE against sample size.

Figure 1 shows a single plot of the sort just described. The HF regressor is `INDPRO`, the MIDAS lags are 1 to 10, and the forecast start date is 2000Q1 (so the estimation samples end in 1999Q4). In this example there is some turbulence at the lower end of the sample-size range, but once that resolves the three most parsimonious MIDAS specifications perform with roughly equal effectiveness and their RMSEs are distinctly smaller than the competition. On the basis of this example we might in addition hazard the guess that T in the range of something like 90 to 100 quarters is optimal for MIDAS forecasting.

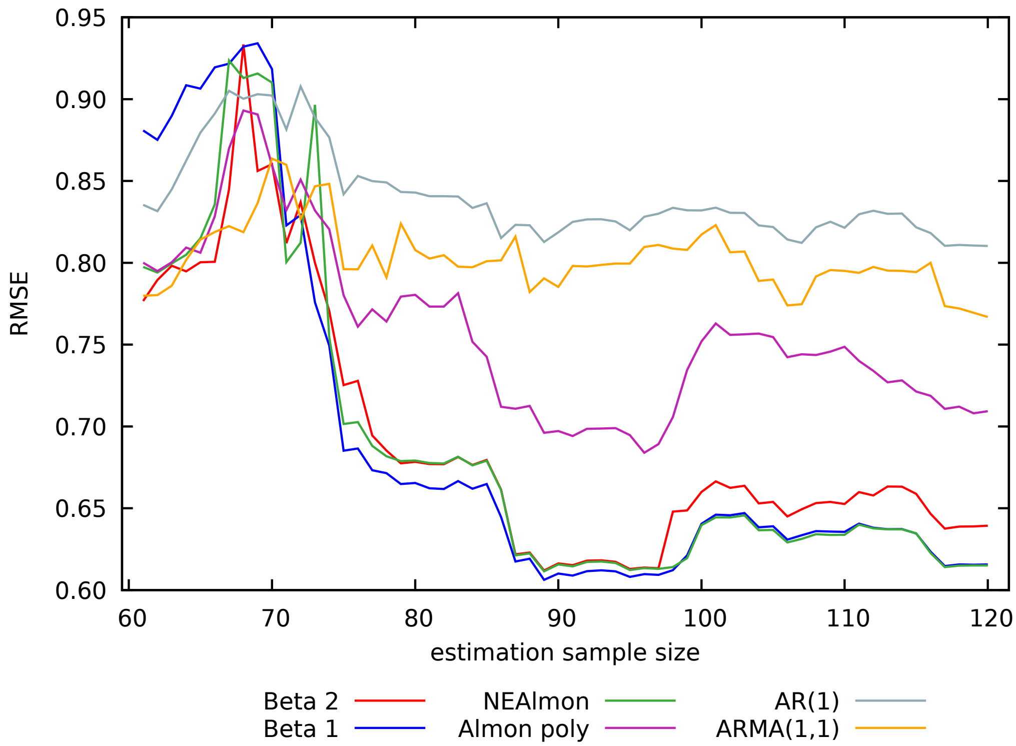


Figure 1: Comparison of GDP growth forecast performance: RMSE of 8 successive quarterly forecasts starting in 2000Q1. The MIDAS specifications use high-frequency lags 1 to 10 of industrial production.

It turns out that matters are somewhat murkier when we consider the other dimensions of variation, in particular choice of high-frequency regressor and forecast target date. The relevant experiments are shown in Figures 2 and 3. In these Figures we vary the high-frequency regressor (by column) and target date (by row).

Relative to Figure 1, the notable "reversals" in Figure 2 (which maintains the MIDAS lag set of the former) include:

- `INDPRO`, target 2010Q1: ARMA(1,1) beats MIDAS except for a small set of sample sizes where NEAlmon wins.
- `INDPRO`, target 2015Q1: MIDAS does quite well, but only at sample sizes greater than appeared optimal from Figure 1.

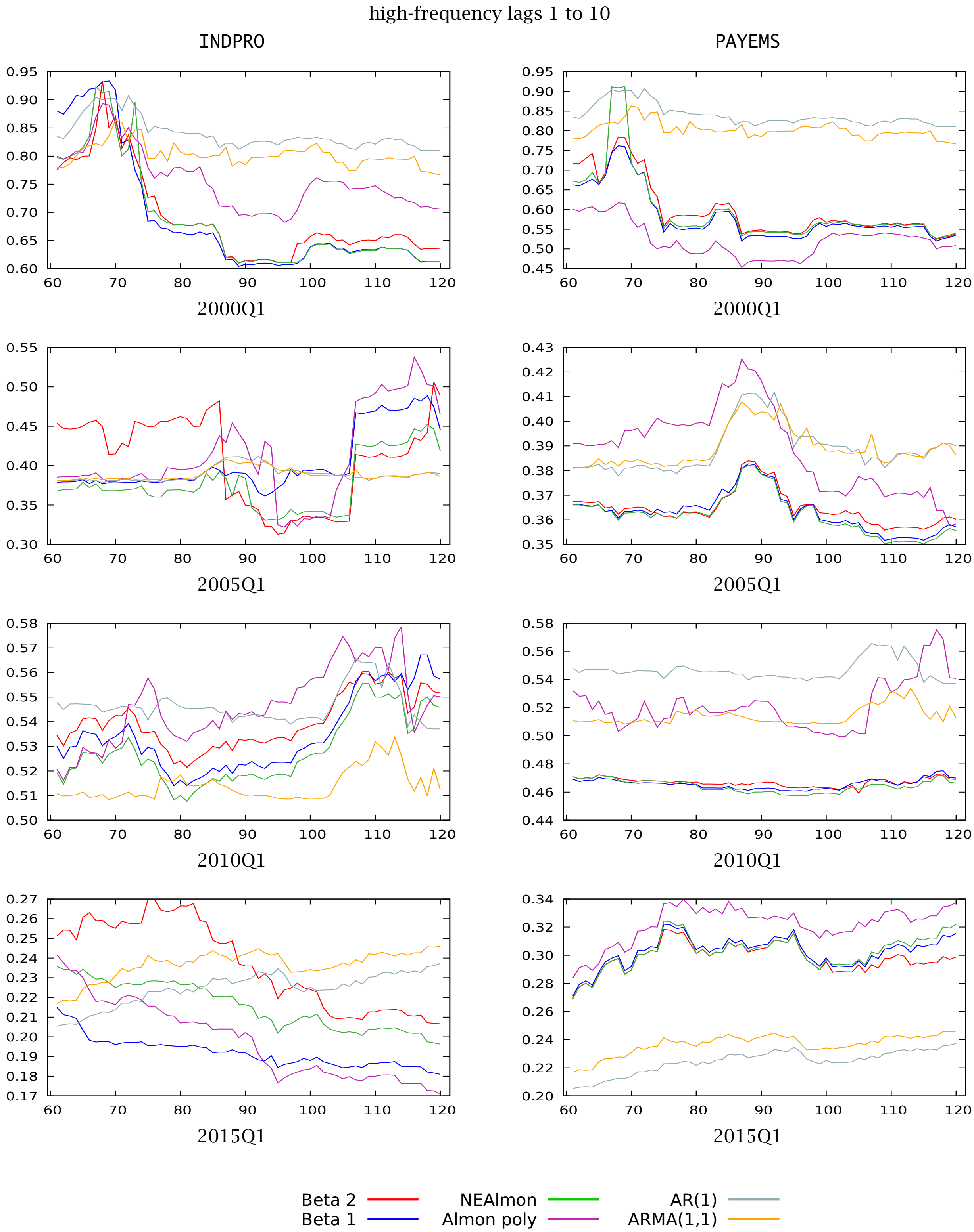


Figure 2: Comparison of GDP growth forecast performance: estimation sample size on x-axis, RMSE of 8 successive quarterly forecasts on y-axis. Forecasts start on the date shown under each panel; high-frequency regressor shown at top of column.

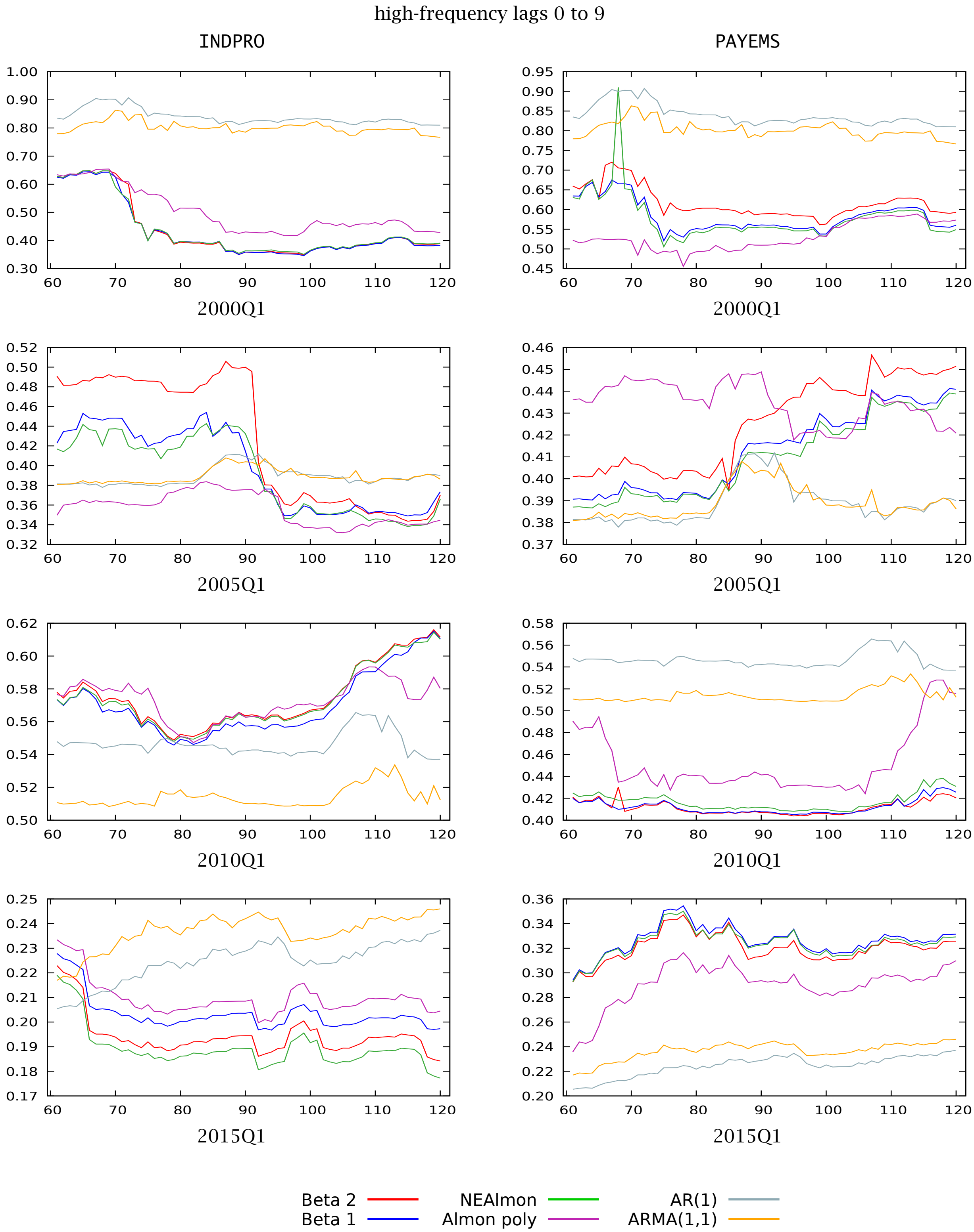


Figure 3: Comparison of GDP growth forecast performance: estimation sample size on x-axis, RMSE of 8 successive quarterly forecasts on y-axis. Forecasts start on the date shown under each panel; high-frequency regressor shown at top of column.

- PAYEMS, target 2015Q1: AR(1) and ARMA(1,1) perform distinctly better than any MIDAS formulation.

However, looking on the bright side for MIDAS, note that in 6 out of the 8 panels in Figure 2, the more parsimonious MIDAS specifications beat plain AR and ARMA, for T around 85–100 if not more generally. In only one case (PAYEMS, target 2015Q1) do we see a clear and fairly substantial advantage for the non-MIDAS methods.

The information in Figure 2 also suggests a tentative conclusion with regard to the Beta 2 MIDAS variant. It seems that this may be "over-fitted" relative to Beta 1: it arguably exhibits excessively erratic forecast performance. The fact that it produces the least RMSE in a few instances (forecast target $\times$ sample size) is not inconsistent with this judgment. We might say that same of the Almon polynomial specification: sometimes it's the least-RMSE MIDAS variant but it's clearly risky.

As regards the relative forecasting power of the two high-frequency regressors, the evidence in Figure 2 is mixed. For the target dates 2000Q1 and 2010Q1, use of PAYEMS produces sharper estimates; for 2015Q1 INDPRO works better; and for 2005Q1 "it depends" (on both the estimation sample size and the MIDAS parameterization).

We now turn to Figure 3, which differs from Figure 2 only in respect of the MIDAS lag set: here we specify HF lags 0 to 9, which implies use of high-frequency data obtained a week (PAYEMS) or two (INDPRO) after the prior period's GDP data, or in other words a nowcast at a slightly later point in the given quarter (see Table 1). We might suppose that use of such data ("new information") would enhance the performance of MIDAS forecasts relative to the single-frequency methods. But this seems to be somewhat chancy: on visual inspection it looks as if three cases are better with the MIDAS lags starting at 0 (INDPRO 2001Q1, 2005Q1; PAYEMS 2010Q1), the remainder being either debatable or worse.[2]

However, in asking ourselves to compare two sets of 8 plots—each showing 6 series that may or may not be moving in tandem—we're pushing the capacity of visual cortex. We need to try for some simplification.

A different cut through the dimensions

We now try slicing the dimensions of the forecasting problem differently, and reformulating the experiment such that it's easier to produce numerical figures of merit. We fix on $T = 90$ observations for estimation, and drop the arguably "less successful" MIDAS variants, Beta 2 and Almon poly. We advance the forecast target quarter-by-quarter from 2000Q1 to 2016Q4, in each case re-estimating the models and generating a single forecast.[3] We then calculate the Mean Absolute Error and Mean Square Error for the series of 68 forecasts, per model.

Results are shown in Table 2, for each of the two high-frequency series under consideration and for three different choices of HF lags. Take the INDPRO results first. The MIDAS forecasting advantage seems reasonably clear when HF lags 0 to 9 are used (whether we take the absolute or squared forecast error as the criterion). Comparing the best MIDAS parameterization with the best non-MIDAS variant, we calculate the paired-difference t-statistics shown in the rightmost column.

MIDAS also "wins" when using lags 1 to 10 of INDPRO, but in this case the difference from non-MIDAS performance is not so convincing. When we go to lags 3 to 11, the MIDAS forecast

[2]For completeness, we also generated a set of plots on the pattern of Figures 2 and 3 using HF lags 3 to 11, as per Ghysels. It was clear that this produced worse results than either of the Figures shown.

[3]Note that this is unlike the first experiments, in which we followed Ghysels' example by generating 8 successive forecasts from each equation, without re-estimation.

performance is inferior to both AR(1) and ARMA(1,1), although the difference is not significant at conventional levels.

HF series `INDPRO`

HF lags		Beta 1	NEAlmon	AR(1)	ARMA(1,1)	$t(67)$
0 to 9	MAE	0.3788	0.3758	0.4399	0.4445	−1.805
	MSE	0.2648	0.2648	0.3713	0.3590	−2.023
1 to 10	MAE	0.3935	0.3977	0.4399	0.4445	−1.581
	MSE	0.2874	0.2946	0.3713	0.3590	−1.953
3 to 11	MAE	0.4493	0.4594	0.4399	0.4445	0.697
	MSE	0.3867	0.3812	0.3713	0.3590	1.345

HF series `PAYEMS`

HF lags		Beta 1	NEAlmon	AR(1)	ARMA(1,1)	$t(67)$
0 to 9	MAE	0.4404	0.4400	0.4399	0.4445	0.004
	MSE	0.3124	0.3117	0.3713	0.3590	−0.131
1 to 10	MAE	0.4291	0.4303	0.4399	0.4445	−0.305
	MSE	0.3243	0.3286	0.3713	0.3590	−0.537
3 to 11	MAE	0.4502	0.4560	0.4399	0.4445	0.515
	MSE	0.3689	0.3803	0.3713	0.3590	0.439

Table 2: Mean absolute and mean square forecast errors for 68 successive quarterly US GDP forecasts, 2000Q1 to 2016Q4. Highlighted: row minima in red, comparison values for paired-difference test in blue.

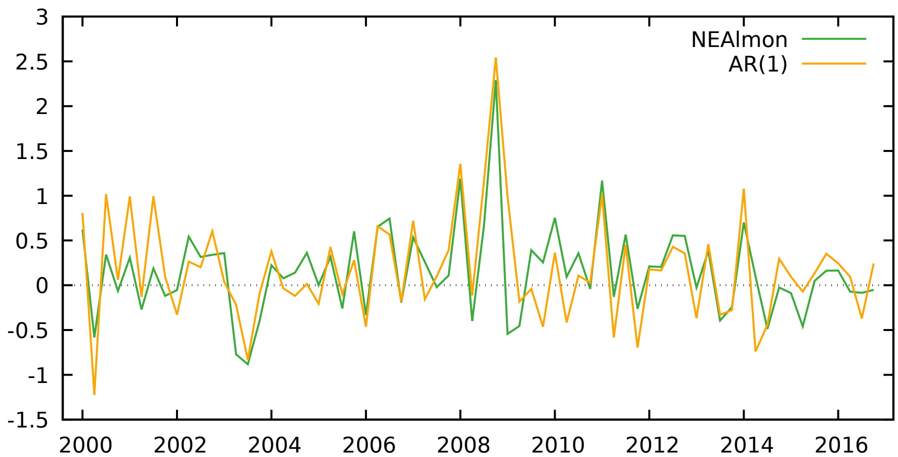
Payroll employment does not do as well as industrial production. While MIDAS shows a marginal advantage in some cases (when lags 1 to 10 are used, in particular), in no case is there a difference from the best non-MIDAS method that approaches statistical significance.

Figure 4 shows the most favorable case for MIDAS from the present experiment: where we use lags 0 to 9 of `INDPRO`. The upper panel simply shows forecast errors from the best representative of MIDAS—which we take to be normalized exponential Almon, `NEAlmon`—and the best representative of non-MIDAS, namely AR(1). The lower panel aims to make the comparison a little clearer, by showing the cumulated difference between the absolute forecast error for `NEAlmon` and that for AR(1). That this series migrates quite far into negative territory arguably indicates an endorsement of MIDAS.

What have we learned?

Suppose we were trying to persuade a skeptic who doubts whether MIDAS forecasting methods offer any worthwhile advantage over simple, traditional single-frequency methods. Well, it's up to the skeptic to judge, but it seems that the relative performance of monthly industrial production in forecasting US GDP in the 21st century should give some grounds for considering MIDAS as a live alternative. Whether this point generalizes to other countries, we are not in a position to say. It would be interesting to see if our findings are confirmed or thrown in doubt by examination of European macroeconomic data.

(a) Forecast errors



(b) Cumulation of difference in absolute errors

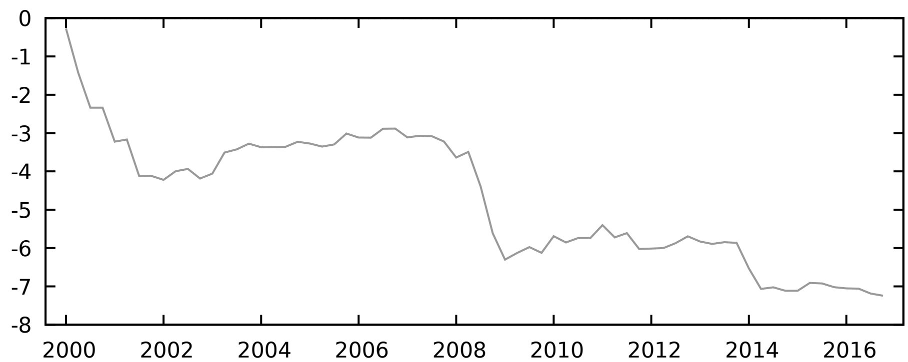


Figure 4: GDP growth forecasts, 2000Q1 to 2016Q4. The MIDAS specification is normalized exponential Almon using HF lags 0 to 9 of `INDPRO`. Panel (b) shows the cumulated value of the difference in absolute errors, MIDAS minus AR(1).

4 Using realtime data

As mentioned in section 2, we decided not to use realtime data in the present study. Nonetheless, we wish to show that this is quite feasible, albeit more complicated, in gretl. Let us take, for example, the nowcast for the log difference of GDP in 2000Q3 (one of the points shown in Figure 4).

To produce a counterpart realtime nowcast and error (the error being assessed relative to the first published GDP data), we can draw from the ALFRED archival data system at the Federal Reserve Bank of St Louis. To formulate a suitable query we need to figure out three dates; in the terminology of the ALFRED API these are `realtime_start` and `realtime_end` (the earliest and most recent "vintages" of data to retrieve), plus `observation_start` (the starting point of the observations).

If we're nowcasting GDP for 2000Q3 (months 7 to 9) using lags 0 to 9 of industrial production, we need to "place ourselves" about six weeks into the quarter—say, 2000-08-16. This will be our `realtime_start`; we'll confine ourselves to data released by that point in estimating our forecasting models, and in supplying right-hand side values for the forecasting equations.

If we want to compare the nowcast with an actual outcome, however, we'll need some later data, namely the first published value of GDP for 2000Q3, which will not be available until a month or so into 2000Q4. So we'll set a suitable `realtime_end`, 2000-10-31.

To allow for $T = 90$ observations (or more) for estimation, we set `observation_start` to 1975-01-01.

Listing 1 presents a hansl script that assembles the required dataset. This script calls a function named `alfred_download` which is shown in Listing 2. The script in Listing 3 opens the data file created by the first script and performs the estimation and forecasting. Table 3 presents output from the realtime script, compared with results based on current data for the period in question.

	actual	beta 1	NEAlmon	AR(1)	ARMA(1,1)
current, data and nowcast	0.1206	0.4646	0.4637	1.1369	1.0879
nowcast error, $\hat{y}_t - y_t$		0.3439	0.3430	1.0163	0.9673
realtime, data and nowcast	0.7854	0.9713	0.9824	0.9401	0.9947
nowcast error, $\hat{y}_t - y_t$		0.1859	0.1970	0.1547	0.2092

Table 3: Published values and nowcasts for log difference of US GDP in 2000Q3 (y_t); current data as of May 2017, realtime data as of August–October 2000.

On current data it appears that real GDP grew by 0.12 percent in 2000Q3, and the normalized exponential Almon MIDAS method came closest to forecasting this result. On the data available at the time, however, GDP grew by 0.79 percent, and the AR(1) forecast was closest. (In all cases the forecasts overestimated growth.)

Listing 1: Assembling an archival MIDAS dataset

```
set verbose off
include alfred_get_json.inp

string rt1 = "2000-08-16" # real-time start
string rt2 = "2000-10-31" # real-time end
string ot1 = "1975-01-01" # observation date start
string key = "YOUR-FRED-API-KEY-HERE"

# get monthly Industrial Production
string CSV = alfred_download("INDPRO", rt1, rt2, ot1, key)
outfile "tmp.csv" --write
print CSV
outfile --close

T = smplspan("1975:01", "2000:09", 12)
nulldata T --preserve
setobs 12 1975:01
delete index
join tmp.csv INDPRO rtstart --tkey=date --aggr=min(rtstart)

# compact to quarterly and get GDP
dataset compact 4 spread
string CSV = alfred_download("GDPC1", rt1, rt2, ot1, key)
outfile "tmp.csv" --write
print CSV
outfile --close
join tmp.csv GDPC1 rtstart --tkey=date --aggr=min(rtstart)

# move quarterly data up front
dataset renumber GDPC1 1
dataset renumber rtstart 2

# save data and clean up
store alfred-20000816.gdt
remove("tmp.csv")
```

Listing 2: Function to download specified archival data from ALFRED

```
function string alfred_download (const string name,
                                 const string rt1,
                                 const string rt2,
                                 const string ot1,
                                 const string key)
  string URL = "https://api.stlouisfed.org/fred/series/observations"
  URL += sprintf("?series_id=%s", name)
  URL += sprintf("&realtime_start=%s", rt1)
  URL += sprintf("&realtime_end=%s", rt2)
  URL += sprintf("&observation_start=%s", ot1)
  URL += sprintf("&api_key=%s", key)
  URL += "&file_type=json"
  string s = readfile(URL)
  string obscount = jsonget(s, "$.count")
  printf "observations count = %s\n", obscount
  strings js = array(4)
  js[1] = jsonget(s, "$.observations[*].realtime_start")
  js[2] = jsonget(s, "$.observations[*].realtime_end")
  js[3] = jsonget(s, "$.observations[*].date")
  js[4] = jsonget(s, "$.observations[*].value")
  string s1, s2, s3, s4
  outfile CSV --buffer --write --quiet
  printf "rtstart,rtend,date,%s\n", name
  loop while getline(js[1], s1) --quiet
    getline(js[2], s2)
    getline(js[3], s3)
    getline(js[4], s4)
    # convert real-time start/end to numeric
    printf "%s,%s,%s,%s\n", strsub(s1,"-",""), strsub(s2,"-",""), \
      s3, s4
  endloop
  outfile --close
  # ensure that we clean up
  loop i=1..4 -q
    getline(js[i], null)
  endloop
  return CSV
end function
```

Listing 3: Estimation and calculation of a single nowcast and error, using four models.

```
set verbose off
open alfred-20000816.gdt
list INDPRO = INDPRO_m*
series dy = 100 * ldiff(GDPC1)

### set simulation parameters ###
scalar T = 90   # sample size
scalar pmin = 0 # min MIDAS lag
scalar pmax = 9 # max MIDAS lag
fcp = obsnum(2000:3) # forecast period
et2 = fcp - 1        # estimation end
et1 = fcp - T        # estimation start
### end simulation parameters ###

list dXL = hflags(pmin, pmax, hfldiff(INDPRO, 100))
matrix FC = {}
smpl et1 et2

# (1) beta, first parameter clamped at 1.0
midasreg dy 0 dy(-1) ; mdsl(dXL, 2, {1,5}) --clamp-beta
fcast fcp fcp --static --quiet
FC ~= $fcast

# (2) Normalized exponential Almon
midasreg dy 0 dy(-1) ; mdsl(dXL, 1, 2)
fcast fcp fcp --static --quiet
FC ~= $fcast

# (3) AR(1)
ols dy 0 dy(-1)
fcast fcp fcp --static --quiet
FC ~= $fcast

# (4) ARMA(1,1)
arma 1 1 ; dy
fcast fcp fcp --static --quiet
FC ~= $fcast

printf "actual y = %.5g\n", dy[fcp]

# add prediction errors
FC |= FC .- dy[fcp]
colnames(FC, "Beta1 NEAlmon AR(1) ARMA(1,1)")
rownames(FC, "nowcast error")
print FC
```