M4 Card forecasts

Jurgen A. Doornik
Economics Department, Nuffield college, and Institute for New Economic Thinking
at the Oxford Martin School, University of Oxford, UK.

June 8, 2018

1 Introduction

This codes replicates the submission of the *Card* forecasts for the M4 competition, see Doornik, Castle, Hendry *(reference coming later)*.

The code requires Ox 7 or newer, see Doornik (2013).

Running `code/forecast_card.ox` will create the submission files `M4_Doornik_Card.csv`, `M4_Doornik_CardHI.csv`, `M4_Doornik_CardLO.csv`, in the `code` folder. These should be identical to the files with `submitted` prefix.

NB1. The M4 data is the competition version: data have been withheld by the organizers and are not available at this stage. However, the M3 data is the full data set.

If the M4 data is extended, the following code would allow replication of the submitted results

```
decl ch = m4.GetH();
db.SetCutBack(ch);
db.SetHoldBack(ch);
```

assuming `m4` is a valid M4 object and `db` a ForecastAB object.

NB2. The focus is on replication at this stage. If there is more interest when the results of the competition have become available, we will provide formal documentation that can be cited, as well as Ox code.

2 `data` folder

2.1 Creating M4 data sets for analysis with OxMetrics

data_original The M4 data is supplied in csv files. However, these have the observations as a text string, which would need conversion first for handling in Ox. To fix this, load the files are in Excel, and save as xlsx files: `Daily-train.xlsx` ... `Yearly-train.xlsx`.

This folder and the original data is not included here.

data `convert_m4_data.ox` reads the xlsx files, and creates pairs of `in7/bn7` files. This format can be read very quickly by OxMetrics and Ox.

Because the variables are unbalanced, data files organized just by frequency would have many missing values, and so be slow to load. For this reason the files are somewhat arbitrarily split in chunks, after sorting the variables by sample size by name:

	# files	# series	%	H	T_{min}	T_{max}
Hourly	1	414	0.4	48	700	960
Daily	7	4227	4.2	14	93	9919
Weekly	1	359	0.4	13	80	2597
Monthly	11	48000	48.0	18	42	2794
Quarterly	5	24000	24.0	8	16	866
Yearly	6	23000	23.0	6	13	835

In contrast to the original files, the observations are aligned at the end, in such a way that the first forecast is always for 2000(1). The databases have H missing values at the end, reflecting the forecast horizon.

Because the variables are sorted by size, at the end, when the submission is generated, the code has to sort them back into the original numerical order.

2.2 Data handling

The data for a frequency are read into an array of databases. An analysis involves a loop over databases, and within databases over all variables. This processes the variables in increasing sample size (and, for the same sample size, by variable name).

2.3 Data sample

Data is given on an isolated time series y_t, and the objective is to forecast y_t. In the remainder, y_t is always the original series, while x_t is a (possibly) transformed version.

The sample may contain missing observations (but not in M4), so is decomposed in:

T_0 first valid observation,

T_2 last valid observation,

T_1 start of the trailing contiguous block,

H number of forecasts.

The data available as the basis for forecasting has $T = T_2 - T_1 + 1$ observations. Forecasting is from $T_2 + 1$ onwards. T_2 may be reduced if data is held back for forecasting, while T_1 is increased if the sample is deemed 'too long.' The $T_0, ..., T_1$ part of the sample has missing values, unless $T_1 = T_0$.

The primary frequency S is set in the database. If there is no seasonality at S, but constant seasonality is found at a lower frequency, then this can be adopted. A second factor S_2 may be used to introduce seasonality at SS_2. E.g. hourly data has $S = 24, S_2 = 7$ for a weekly frequency of 168.

3 `data_M3` folder

For comparison, the M3 data is supplied in the same format as M4. So it is easy to switch from one to the other.

M3 needs only one file each for annual, querterly and monthly data. The 'other' data is not included, because it was not clear to me what the appropriate frequency should be.

4 `code` folder

- `evaluate_insample.ox` Evaluates forecast methods by withholding H observations from the M4 data.

Switch to M3 by setting `ism3` in `main` to 1. Use `astypes` in the `foreach` loop to run over all frequencies. The current version prints (truncated):

```
loaded ../data/Yearly_01.in7 with 4556 series
loaded ../data/Yearly_02.in7 with 4997 series
loaded ../data/Yearly_03.in7 with 4387 series
loaded ../data/Yearly_04.in7 with 5084 series
loaded ../data/Yearly_05.in7 with 3621 series
loaded ../data/Yearly_06.in7 with 355 series
Run done in  3.92

        MAPE(naive2) MAPE(Delta) MAPE(Rho)MAPE(Delta+C) MAPE(Rho+C) MAPE(Card)
Yearly-median 13.923       8.212     8.882        8.162       8.650      8.183
Yearly-mean   18.462      15.152    15.485       15.112      15.245     14.886
Yearly-mean    1.000       0.821     0.839        0.819       0.826      0.806
```

- `experiment.ox` Creates forecasts for some specific series.

  This program is intended for experimenting or considering a specific series.

- `forecast_card.ox` Creates our M4 submission, producing out-of-sample *Card* forecasts.

- `ForecastAB.oxo` Forecast framework and forecasting methods.

- `M4.ox` Database management for M4 data.

5 code folder

Contains this document, as well as oxdoc documentation for the `ForecastAB` and `M4` classes.

References

Doornik, J. A. (2013). *Object-Oriented Matrix Programming using Ox* (7th ed.). London: Timberlake Consultants Press.