Demand forecasting by temporal aggregation: using optimal or multiple aggregation levels?

Nikolaos Kourentzes\textsuperscript{a}
Bahman R. Tabar\textsuperscript{b}
Devon K. Barrow\textsuperscript{b}

\textsuperscript{a}Lancaster University; \textsuperscript{b}Coventry University

ISIR 2016
Agenda

1. Demand forecasting by temporal aggregation
2. Optimal aggregation level
3. Multiple aggregation levels
4. Evaluation
How do we build models now?

• This is by no means a resolved question, but there are some reliable approaches. Key questions: model form & estimation.

• Take the example of exponential smoothing family:
  • Considered one of the most reliable and robust methods for automatic univariate forecasting [Gardner, 2006].
  • It is a family of methods: **ETS (error type, trend type, seasonality type)** [Hyndman et al., 2002, 2008]
    • Error: **Additive** or **Multiplicative**
    • Trend: **None** or **Additive** or **Multiplicative**, Linear or Damped/Exponential
    • Seasonality: **None** or **Additive** or **Multiplicative**
  • Adequate for a most types of time series.
  • Within the state space framework we can select and fit model parameters automatically and reliably.
Issues with modelling:

- Model selection \(\rightarrow\) How good is the best fit model? How reliable?
- Sampling uncertainty \(\rightarrow\) Identified model/parameters stable as new data appear?
- Model uncertainty \(\rightarrow\) Appropriate model structure and parameters?
- Transparency/Trust \(\rightarrow\) Practitioners do not trust systems that change substantially

A single additional observation changed the selected model and forecast!
Temporal aggregation and forecasting

- Temporal aggregation has been explored as a way to help us deal with these issues.
- It is not new, but the question has been at which single level to model the time series. Econometrics have investigate the question for decades → inconclusive
- Supply chain applications: ADIDA [Nikolopoulos et al., 2011] → beneficial to slow and fast moving items forecast accuracy (like everything... not always!):
  - **Step 1**: Temporally aggregate time series to the appropriate level
  - **Step 2**: Forecast
  - **Step 3**: Disaggregate forecast and use
  - Selection of aggregation level → No theoretical grounding for general case, but good understanding for AR(1)/MA(1)/ARMA(1,1) cases [Rostami-Tabar et al., 2013, 2014].
Temporal aggregation and forecasting

Recently there has been a resurgence in using temporal aggregation for forecasting.

- Non-overlapping temporal aggregation is a moving average filter.
- Filters high frequency components: noise, seasonality, etc.
- Reduces intermittency [Nikolopoulos et al, 2011; Petropoulos & Kourentzes, 2014].
- But can increase complexity [Wei, 1978; Rossana & Seater, 1995; Silvestrini & Veredas, 2008]:
  - Loss of estimation efficiency;
  - Complicates dynamics of underlying (ARIMA) process;
  - Identifiable process converge to IMA - often IMA(1,1);
  - What is the best temporal aggregation to work on?

Two school of thoughts. **How do they compare?**

i. **Traditional**: Identify a single optimal temporal aggregation level to model [Rossana & Seater, 1995; Nikolopoulos et al., 2011; Rostami-Tabar et al., 2013, 2014].

ii. **Use multiple temporal aggregation levels** [Kourentzes et al., 2014; Kourentzes & Petropoulos, 2015].
How temporal aggregation changes the series

Seasonal diagrams
Identifying the optimal aggregation level

Rostami-Tabar et al. 2014 evaluate analytically the impact of temporal aggregation for ARMA(1,1), AR(1) and MA(1) and derive formulas to find the optimal aggregation level in terms of MSE when forecasting with **Single Exponential Smoothing**.

\[
\text{MSE}_{\text{ARMA}} = 2\sigma^2 \frac{\left( k (1 - 2\phi \theta + \theta^2) + (\phi - \theta)(1 - \phi \theta) \left( \sum_{i=1}^{k-1} 2(k-i)\phi^{i-1} \right) \right)}{(2 - \alpha)(1 - \phi^2)} + 2\sigma^2\alpha \frac{\left( \sum_{i=1}^{k} (i\phi^{i-1}) + \sum_{i=2}^{k} (i-1)\phi^{2(k-i)} \right) (\phi - \theta)(1 - \phi \theta)}{(2 - \alpha)(1 - \phi^k + \alpha\phi^k)(1 - \phi^2)}
\]

\[
\text{MSE}_{\text{AR}} = 2\sigma^2 \frac{\left( k + \sum_{i=1}^{k-1} 2(k-i)\phi^i \right)}{(1 - \phi^2)(2 - \alpha)} + 2\sigma^2\alpha \frac{\left( \sum_{i=1}^{k} (i\phi^{i-1}) + \sum_{i=2}^{k} (i-1)\phi^{2(k-i)} \right)}{(2 - \alpha)(1 - \phi^k + \alpha\phi^k)(1 - \phi^2)}
\]

\[
\text{MSE}_{\text{MA}} = \frac{\sigma^2 (2k(1 + \theta^2) - 2(k - 1)\theta + 2\alpha \theta)}{2 - \alpha}
\]

We need to know **parameters of original process** \((\phi, \theta)\) and optimal smoothing parameter \(\alpha\) for SES at **aggregate level** \(k\).

Calculate MSE for various \(k\) and pick the best to find the optimum temporal aggregation level.
Using multiple aggregation levels

What if we do not select an aggregation level? \(\rightarrow\) use multiple [Kourentzes et al., 2014]

### Issues:
- Different model
- Different length
- Combination
\[ y_t^{[1]} \]

**Aggregate**

**Fit state space ETS**

**Save states**

- Level
- Trend
- Season

Aggregate data showing time series with a trend and seasonality.

Fit state space ETS showing the model fit with trend and seasonality components.

Save states showing the state evolution over time for level, trend, and season components.
Transform states to additive and to original sampling frequency

Combine states (components)

Produce forecasts

\[ \hat{y}_{t+h[1]} = \bar{l}_{t+h[1]} + \bar{b}_{t+h[1]} + \bar{s}_{t-S+h[1]} \]
Multiple Aggregation Prediction Algorithm (MAPA)

**Step 1: Aggregation**
- $Y^{[i]}$
- $k = 2$  $\rightarrow Y^{[2]}$
- $k = 3$  $\rightarrow Y^{[3]}$
- $\ldots$
- $k = K$  $\rightarrow Y^{[K]}$

Strengthens and attenuates components

**Step 2: Forecasting**
- ETS Model Selection
- $K^{-1}\sum$

Estimation of parameters at multiple levels

**Step 3: Combination**
- $\hat{Y}^{[i]}$

Robustness on model selection and parameterisation
Empirical evaluation

- **Forecast the next 13 periods**
  - **Simulated**: known processes can be used to assess the optimal selection
    - ARIMA($p,d,q$), with $p = (0,1,2)$, $d = (0,1)$ and $q = (0,1,2)$
    - 500 series each process, 60 fit set & 40 test set.
  - **Real**: 229 series of 173 weekly observations – non-seasonal
    - ADF test suggests that 90% is ARIMA($p,0,q$) and 10% ARIMA($p,1,q$).
    - 130 fit set & 43 test set
  - Accuracy metric: ARMAE; < 1 better than benchmark!

\[
MAE = m^{-1} \sum_{t=1}^{m} |y_t - \hat{y}_t| \\
ARMAE = \sqrt[n]{\prod \left( \frac{MAE_i}{MAE_b} \right)}
\]
Methods

• Original sampling frequency (no aggregation)
  • Benchmark: Single exponential smoothing – Orig-SES
  • ETS model family – Orig-ETS

• Single temporal aggregation
  • Heuristic based level (13 periods) – Heur-SES
  • Optimal level – Opt-SES

• Multiple temporal aggregation
  • Restricted to SES only – MAPA-SES
  • Unrestricted – MAPA
### Accuracy - ARMAE

<table>
<thead>
<tr>
<th>Demand</th>
<th>No aggregation</th>
<th>Single level</th>
<th>Multiple levels</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Orig-SES</td>
<td>Orig-ETS</td>
<td>Heur-SES</td>
</tr>
<tr>
<td>ARIMA(1,0,0)</td>
<td>1.000</td>
<td>0.979</td>
<td>0.974</td>
</tr>
<tr>
<td>ARIMA(0,0,1)</td>
<td>1.000</td>
<td>1.002</td>
<td><strong>0.960</strong></td>
</tr>
<tr>
<td>ARIMA(2,0,0)</td>
<td>1.000</td>
<td>0.971</td>
<td>0.986</td>
</tr>
<tr>
<td>ARIMA(0,0,2)</td>
<td>1.000</td>
<td>1.002</td>
<td><strong>0.969</strong> = <strong>0.969</strong></td>
</tr>
<tr>
<td>ARIMA(1,0,1)</td>
<td>1.000</td>
<td>1.001</td>
<td>0.966</td>
</tr>
<tr>
<td>ARIMA(2,0,2)</td>
<td>1.000</td>
<td>0.983</td>
<td>0.990</td>
</tr>
<tr>
<td>ARIMA(1,1,0)</td>
<td><strong>1.000</strong></td>
<td><strong>1.000</strong></td>
<td>1.439</td>
</tr>
<tr>
<td>ARIMA(0,1,1)</td>
<td><strong>1.000</strong></td>
<td>1.051</td>
<td>1.290</td>
</tr>
<tr>
<td>ARIMA(2,1,0)</td>
<td>1.000</td>
<td><strong>0.891</strong></td>
<td>1.444</td>
</tr>
<tr>
<td>ARIMA(0,1,2)</td>
<td><strong>1.000</strong></td>
<td>1.048</td>
<td>1.278</td>
</tr>
<tr>
<td>ARIMA(1,1,1)</td>
<td>1.000</td>
<td><strong>0.975</strong></td>
<td>1.349</td>
</tr>
<tr>
<td>ARIMA(2,1,2)</td>
<td>1.000</td>
<td>0.927</td>
<td>1.327</td>
</tr>
<tr>
<td>ARIMA(<em>,0,</em>)</td>
<td>1.000</td>
<td>0.989</td>
<td>0.974 = 0.974</td>
</tr>
<tr>
<td>ARIMA(<em>,1,</em>)</td>
<td>1.000</td>
<td>0.980</td>
<td>1.353</td>
</tr>
<tr>
<td>ARIMA(<em>,</em>,*)</td>
<td>1.000</td>
<td>0.985</td>
<td>1.148</td>
</tr>
<tr>
<td>Real Data</td>
<td>1.000</td>
<td>1.011</td>
<td>0.999 = 0.999</td>
</tr>
</tbody>
</table>
Conclusions

• Temporal aggregation **improves accuracy → use it!**

• Identifying **optimal aggregation level** does not always work as expected, but **overall equal if not better than heuristic** for selecting level.

• Why? **Optimal selection** of level is **not robust enough to model uncertainty** at the original and aggregate level → we simply do not know the true process and optimal selection assumes knowledge of it.

• **MAPA** by construction is **suboptimal** for any process, **but it is very robust** and reliable → consistently resulted in better accuracy (matches the literature).

• Future research should focus on:
  • How can we make optimal more robust?
  • How can we make MAPA “more optimal”?
To temporally aggregate a series use the function `tsaggr` from the **MAPA** package:
http://cran.r-project.org/web/packages/MAPA/index.html

Code for finding the **optimal temporal aggregation level** is available for R, in the **TStools** package, which is available at GitHub (not in CRAN yet):
https://github.com/trnnick/TStools - function: `get.opt.k`

The **Multiple Aggregation Prediction Algorithm** is available for R, in the **MAPA** package:
http://cran.r-project.org/web/packages/MAPA/index.html
Its intermittent demand counterpart is available in the **tsintermittent** package:
http://cran.r-project.org/web/packages/tsintermittent/index.html
Examples and interactive demos for both are available at my blog:
http://nikolaos.kourentzes.com
Thank you for your attention!

Questions?

Published, working papers and code available at my blog!

Nikolaos Kourentzes

date: n.kourentzes@lancaster.ac.uk
date: http://nikolaos.kourentzes.com