Large-scale time series forecasting with applications: state-of-the-art

Feng Li

School of Statistics and Mathematics
Central University of Finance and Economics

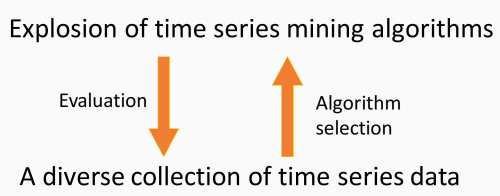
Outline

Elements of good forecasts: *state-of-the-art perspectives*

- Robust again a large collection of benchmarking data.
  - What if I do not have any benchmark data?
  - Build a model on machine-generated data and test on real data.
- Properly tackling model uncertainty and data uncertainty.
  - What shall we do when all forecasting model fail?
  - Let's forecast without data.
- Good speed performance with a large scale of time series.
  - Most forecast models could not scale up.
  - A need of a distributed forecasting framework.

GRATIS: GeneRAting TIme Series with diverse and controllable characteristics
↪ Motivation

- Train a time series model (*machine learning with dependent data*) is usually costly.
- New algorithms are developed every day.



- A well trained model with `my dataset` does not necessary work well for `your dataset`. Why?
- Is there a way to **forecast which algorithm works the best** for any time series *ex-ante* ?
  - Unrealistic because we could not collect all the time series in the world.
  - But we could work on the time series feature space.
  - Turns out it works equally well!

Time series features

Transform a given time series $\{x_1, x_2, \cdots, x_n\}$ to a feature vector $F = (F_1, F_2, \cdots, F_p)'$ (Kang et al., 2017)

A feature F_k can be any kind of function computed from a time series:

1. A simple mean
2. The parameter of a fitted model
3. Some statistic intended to highlight an attribute of the data
4. ...

GRATIS: GeneRAting TIme Series with diverse and controllable characteristics

↬ Time series features we use

Feature	Description	Feature	Description
F_1	Number of seasonal periods	F_{10}	Strength of trend
F_2	Vector of seasonal periods	F_{11}	Strength of seasonality
F_3	Number of differences for stationarity	F_{12}	Spikiness
F_4	Number of seasonal differences for stationarity	F_{13}	Autocorrelation coefficients of remainder
F_5	Autocorrelation coefficients	F_{14}	ARCH ACF statistic
F_6	Partial autocorrelation coefficients	F_{15}	GARCH ACF statistic
F_7	Spectral entropy	F_{16}	ARCH R^2 statistic
F_8	Nonlinearity coefficient	F_{17}	GARCH R^2 statistic
F_9	Long-memory coefficient		

- We have developed an R package: `tsfeatures` available on CRAN.

GRATIS: GeneRAting TIme Series with diverse and controllable characteristics ↛ with Gaussian Mixure Autoregressions

- Consist of multiple stationary or non-stationary autoregressive components.
- A K-component MAR model is defined as (Wong & Li, 2000) :

$$F(x_t|\mathcal{F}_{t-1}) = \sum_{k=1}^{K} \alpha_k \Phi\left(\frac{x_t - \phi_{k0} - \phi_{k1}x_{t-1} - \cdots - \phi_{kp_k}x_{t-p_k}}{\sigma_k}\right),$$

  where $F(x_t|\mathcal{F}_{t-1})$ is the conditional cumulative distribution of x_t give the past information $\mathcal{F}_{t-1}$. $\Phi(\cdot)$ is the cumulative distribution function of the standard normal distribution. $\sum_{k=1}^{K} \alpha_k = 1$, where $\alpha_k > 0$, $k = 1, 2, \cdots, K$.
- Mixtures of stationary and non-stationary components can yield a stationary process.
- To handle non-stationary time series, one can just include a unit root in each component.
- Possible to capture more (or any) time series features, since different specifications of finite mixtures have been shown to be able to approximate large nonparametric classes of conditional multivariate densities (Li et al., 2010; Norets, 2010).

↛ Investigating the coverage of MAR models

Dataset A	Dataset B					
	DGP	M4	M3	M1	Tourism	NNGC1
			Yearly			
DGP	0.00	0.02	0.01	0.00	0.00	0.00
M4	0.06	0.00	0.01	0.00	0.00	0.00
M3	0.35	0.31	0.00	0.04	0.05	0.00
M1	0.55	0.50	0.25	0.00	0.09	0.01
Tourism	0.51	0.47	0.22	0.05	0.00	0.01
NNGC1	0.66	0.61	0.34	0.13	0.20	0.00
			Quarterly			
DGP	0.00	0.04	0.01	0.00	0.00	0.00
M4	0.09	0.00	0.01	0.00	0.00	0.00
M3	0.42	0.34	0.00	0.04	0.08	0.01
M1	0.53	0.47	0.16	0.00	0.10	0.01
Tourism	0.53	0.46	0.20	0.10	0.00	0.01
NNGC1	0.65	0.58	0.26	0.13	0.14	0.00
			Monthly			
DGP	0.00	0.06	0.00	0.00	0.00	0.00
M4	0.07	0.00	0.00	0.01	0.00	0.00
M3	0.36	0.32	0.00	0.06	0.03	0.00
M1	0.45	0.42	0.16	0.00	0.06	0.00
Tourism	0.59	0.54	0.27	0.21	0.00	0.01
NNGC1	0.68	0.63	0.34	0.26	0.12	0.00
			Weekly			
DGP	0.00	0.00				0.00
M4	0.59	0.00				0.01
M3						

GRATIS: GeneRAting TIme Series with diverse and controllable characteristics

↛ Modelling features and forecasting performances with purely generated data

$$\mathbf{MASE_{N\times 6}} \Leftrightarrow \mathbf{F_{N\times p}}$$

$$\mathbf{MASE^{(i)}} = f_1^{(i)}(F_1) + f_2^{(i)}(F2) + \ldots + f_p^{(i)}(F_p) + \epsilon^{(i)}$$

- This relationship is obviously nonlinear. We use the Bayesian spline regressions to capture the nonlinearity (Li & Villani, 2013).
- R package: `movingknots` available on GitHub https://github.com/feng-li/movingknots

GRATIS: GeneRAting TIme Series with diverse and controllable characteristics

↛ Apply the model on the forecasts on M3 (out-of-sample)

Method	Yearly		Quarterly		Monthly		All	
	Mean	Median	Mean	Median	Mean	Median	Mean	Median
Naïve	3.172	2.267	1.464	1.044	1.175	0.927	1.707	1.135
Seasonal naïve	3.172	2.267	1.425	1.176	1.146	0.969	1.683	1.146
Theta	2.773	1.985	**1.114**	0.842	0.889	0.751	1.379	0.886
ETS	2.879	1.961	1.188	0.868	0.865	**0.716**	1.410	0.870
ARIMA	2.964	1.864	1.187	0.843	0.877	0.727	1.436	0.872
STL-AR	2.953	1.854	1.911	1.687	1.268	1.011	1.824	1.246
Method Selection	**2.746**	**1.782**	1.129	**0.813**	**0.855**	0.724	**1.360**	**0.857**

GRATIS: GeneRAting TIme Series with diverse and controllable characteristics

↪ Extensions

- Details available in Kang, Hyndman & Li (2020).
- Try our R package `gratis` available on CRAN.
- We also have an online APP at https://ebsmonash.shinyapps.io/tsgeneration/
- Density forecasting.
- Framework on non-time series.

Time series forecasting with cross-similarity

↬ "All models are wrong, but some are useful."– George Box

- Three sources of uncertainty exist in forecasting: **model**, **parameter**, and **data**.
  - Merely tackling the model uncertainty is sufficient to bring most of the performance benefits.
- "All models are wrong, but **some** are useful."
  - Researchers increasingly avoid using a single model, and opt for combinations of forecasts from multiple models.


GAME OVER
CONTINUE? 09
MemeCenter.com

Time series forecasting with cross-similarity
↪ Déjà Vu

- We argue that there is another way to avoid selecting a single model: **to select no models at all**.
- We provide a new way to forecasting that does not require the estimation of any forecasting models, while also exploiting the benefits of cross-learning.

Time series forecasting with cross-similarity
↪ The idea for déjà vu

1. A target series is compared against a set of reference series attempting to identify similar ones (déjà vu).
2. The point forecasts for the target series are the average of the future paths of the most similar reference series.
3. The prediction intervals are based on the distribution of the reference series, calibrated for low sampling variability. Note that no model extrapolations take place in our approach.
4. The proposed approach has several advantages compared to existing methods, namely
    - it tackles both model and parameter uncertainties
    - it does not use time series features or other statistics as a proxy for determining similarity, and
    - no explicit assumptions are made about the DGP as well as the distribution of the forecast errors.

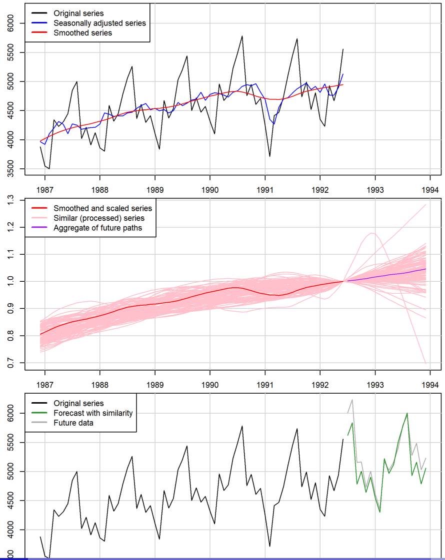
Time series forecasting with cross-similarity

↬ Methodology

- The objective of "forecasting with similarity" is to find the most similar ones to a target series, average their future paths, and use this average as the forecasts for the target series.
  1. **Removing seasonality**, if a series is identified as seasonal.
  2. **Smoothing** by estimating the trend component through time series decomposition.
  3. **Scaling** to render the target and possible similar series comparable.
  4. **Measuring similarity** by using a set of distance measures.
  5. **Forecasting** by aggregating the paths of the most similar series.
  6. **Inverse scaling** to bring the forecasts for the target series back to its original scale.
  7. **Recovering seasonality**, if the target series is found seasonal in Step 1.
- We use the yearly, quarterly, and monthly subsets of the M4 competition, which consist of 23000, 24000, and 48000 series, respectively.

Time series forecasting with cross-similarity

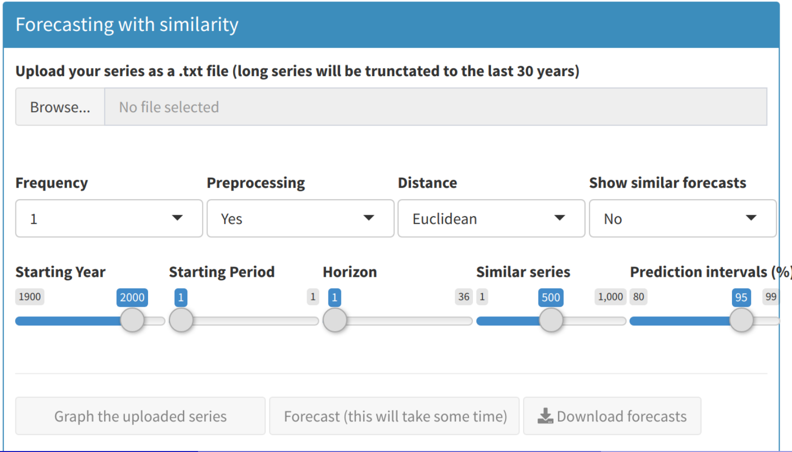
↛ Toy example



Time series forecasting with cross-similarity

Online APP

- Details available in Kang et al. (2021)
- Try our online App https://fotpetr.shinyapps.io/similarity/
- R package available at https://github.com/kl-lab/dejavu

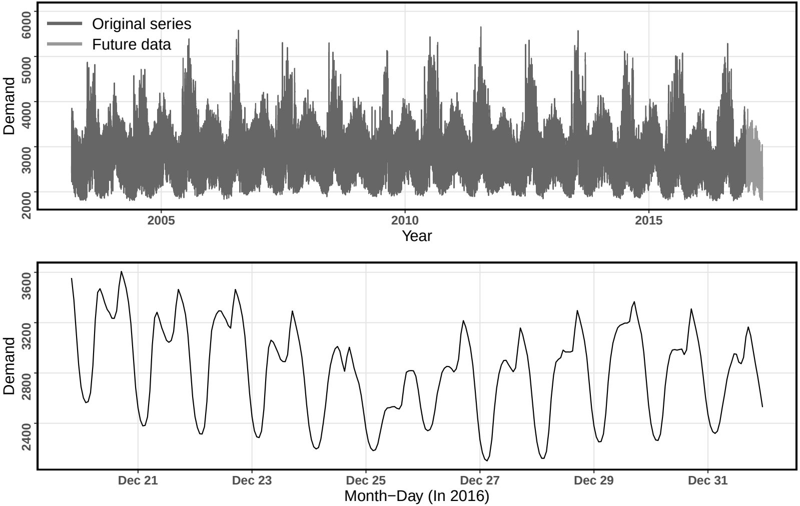


↪ Motivation

- Ultra-long time series are increasingly accumulated in many cases.
    - hourly electricity demands
    - daily maximum temperatures
    - streaming data generated in real-time
- Forecasting these time series is challenging.
    - time-consuming training process
    - hardware requirements
    - unrealistic assumption that the DGP remains invariant over a long time interval

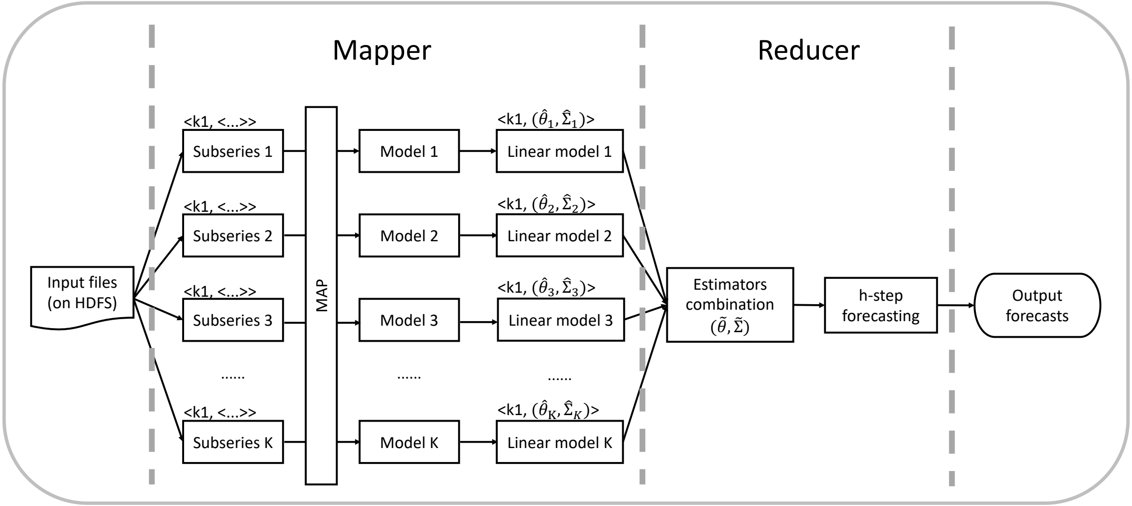
Distributed forecasting with ultra-long time series

↛ Electricity load data



Distributed forecasting with ultra-long time series

↪ The forecasting framework



Distributed forecasting with ultra-long time series

↪ Need for speed!

Max orders	Method	MASE	MSIS	Execution time (mins)
(5, 2, 5)	ARIMA	1.430	19.733	4.596
	DARIMA	**1.297**	**15.078**	**1.219**
(5, 2, 7)	ARIMA	1.410	18.695	14.189
	DARIMA	**1.297**	**15.078**	**1.211**
(6, 2, 7)	ARIMA	1.410	18.695	15.081
	DARIMA	**1.298**	**15.108**	**1.326**
(6, 3, 7)	ARIMA	1.413	15.444	21.072
	DARIMA	**1.324**	**12.590**	**1.709**
(6, 3, 10)	ARIMA	1.413	15.654	76.272
	DARIMA	**1.324**	**12.590**	**1.769**
(7, 3, 10)	ARIMA	1.413	15.654	83.077
	DARIMA	**1.327**	**12.561**	**1.829**
(7, 4, 10)	ARIMA	1.409	13.667	111.292
	DARIMA	**1.338**	**12.079**	**2.267**
(8, 4, 10)	ARIMA	1.409	13.667	117.875
	DARIMA	**1.335**	**12.076**	**2.224**

↳ Discussions

- Distributed forecasting not only speeds up the computation but also improves forecasting performance. **Why?**
- Details available in Wang, Kang, Hyndman & Li (2022).
- Try our software `https://github.com/feng-li/darima/` if you know distributed computation.

References

- Yanfei Kang, Rob J Hyndman & Feng Li (2020). GRATIS: GeneRAting TIme Series with diverse and controllable characteristics. *Statistical Analysis and Data Mining* **13** (4), 354–376.
- Yanfei Kang, Rob J Hyndman & Kate Smith-Miles (2017). Visualising forecasting algorithm performance using time series instance spaces. *International Journal of Forecasting* **33**(2), 345–358.
- Yanfei Kang, Evangelos Spiliotis, Fotios Petropoulos, Nikolaos Athiniotis, Feng Li & Vassilios Assimakopoulos (2021). Déjà vu: A data-centric forecasting approach through time series cross-similarity. *Journal of Business Research* **132**(2021), 719–731.
- Feng Li & Mattias Villani (2013). Efficient Bayesian Multivariate Surface Regression. *Scandinavian Journal of Statistics* **40**(4), 706–723.
- Feng Li, Mattias Villani & Robert Kohn (2010). Flexible modeling of conditional distributions using smooth mixtures of asymmetric student t densities. *Journal of Statistical Planning and Inference* **140**(12), 3638–3654.
- Andriy Norets (2010). Approximation of conditional densities by smooth mixtures of regressions. *The Annals of Statistics* **38**(3), 1733–1766.
- Xiaoqian Wang, Yanfei Kang, Rob J Hyndman & Feng Li (2022). Distributed ARIMA models for ultra-long time series. *International Journal of Forecasting* (In Press).
- Chun Shan Wong & Wai Keung Li (2000). On a mixture autoregressive model. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **62**(1), 95–115.

The best way to predict the future is to create it!

http://feng.li/
feng.li@cufe.edu.cn