Meta-learning how to forecast time series

Thiyanga S. Talagala
Rob J Hyndman
George Athanasopoulos

Monash University, Australia

Large collections of time series

- Forecasting demand for thousands of products across multiple warehouses.

Time series features

Objective

Develop a framework that automates the selection of the most appropriate forecasting model for a given time series by using an array of features computed from the time series.

Time series features

Objective

Develop a framework that automates the selection of the most appropriate forecasting model for a given time series by using an array of features computed from the time series.

- **Basic idea:**
  Transform a given time series $y = \{y_1, y_2, \cdots, y_n\}$ to a feature vector $F = (f_1(y), f_2(y), \cdots, f_p(y))'$.

Time series features

Objective

Develop a framework that automates the selection of the most appropriate forecasting model for a given time series by using an array of features computed from the time series.

- **Basic idea:**
  Transform a given time series $y = \{y_1, y_2, \cdots, y_n\}$ to a feature vector $F = (f_1(y), f_2(y), \cdots, f_p(y))'$.
- Examples for time series features

Time series features

Objective

Develop a framework that automates the selection of the most appropriate forecasting model for a given time series by using an array of features computed from the time series.

- **Basic idea:**
  Transform a given time series $y = \{y_1, y_2, \cdots, y_n\}$ to a feature vector $F = (f_1(y), f_2(y), \cdots, f_p(y))'$.
- Examples for time series features
  - strength of trend

Time series features

Objective

Develop a framework that automates the selection of the most appropriate forecasting model for a given time series by using an array of features computed from the time series.

- **Basic idea:**
  Transform a given time series $y = \{y_1, y_2, \cdots, y_n\}$ to a feature vector $F = (f_1(y), f_2(y), \cdots, f_p(y))'$.
- Examples for time series features
  - strength of trend
  - strength of seasonality

Time series features

Objective

Develop a framework that automates the selection of the most appropriate forecasting model for a given time series by using an array of features computed from the time series.

- **Basic idea:**
  Transform a given time series $y = \{y_1, y_2, \cdots, y_n\}$ to a feature vector $F = (f_1(y), f_2(y), \cdots, f_p(y))'$.
- Examples for time series features
  - strength of trend
  - strength of seasonality
  - lag-1 autocorrelation

Time series features

Objective

Develop a framework that automates the selection of the most appropriate forecasting model for a given time series by using an array of features computed from the time series.

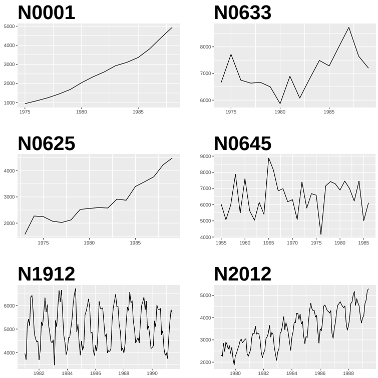
- **Basic idea:**
  Transform a given time series $y = \{y_1, y_2, \cdots, y_n\}$ to a feature vector $F = (f_1(y), f_2(y), \cdots, f_p(y))'$.
- Examples for time series features
  - strength of trend
  - strength of seasonality
  - lag-1 autocorrelation
  - spectral entropy

Feature-space of time series

STL-decomposition

$$Y_t = T_t + S_t + R_t$$

- strength of trend: $1 - \frac{Var(R_t)}{Var(Y_t - S_t)}$
- strength of seasonality: $1 - \frac{Var(R_t)}{Va(Y_t - T_t)}$

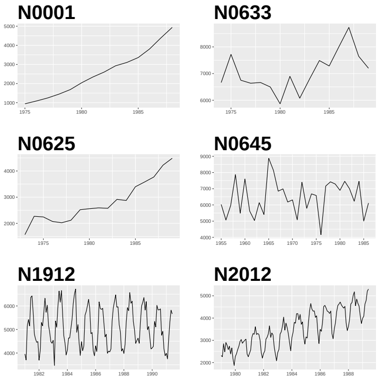


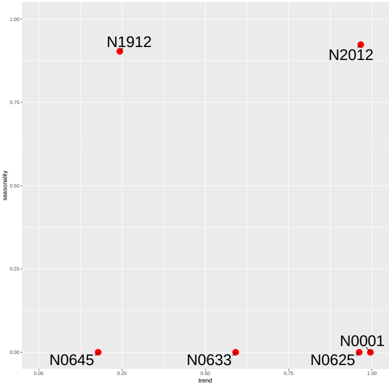
Feature-space of time series

STL-decomposition

$$Y_t = T_t + S_t + R_t$$

- strength of trend: $1 - \frac{Var(R_t)}{Var(Y_t - S_t)}$
- strength of seasonality: $1 - \frac{Var(R_t)}{Va(Y_t - T_t)}$





Time series features

- length
- strength of seasonality
- strength of trend
- linearity
- curvature
- spikiness
- stability
- lumpiness
- first ACF value of remainder series
- parameter estimates of Holt's linear trend method
- spectral entropy
- Hurst exponent
- nonlinearity
- parameter estimates of Holt-Winters' additive method
- unit root test statistics
- first ACF value of residual series of linear trend model
- ACF and PACF based features - calculated on both the raw and differenced series

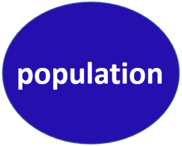
Methodology: FFORMS

FFORMS: Feature-based **FOR**ecast **M**odel **S**election

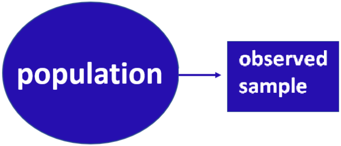
Offline: Classification algorithm is trained

Online: Use the classification algorithm to select appropriate forecast-models for new time series

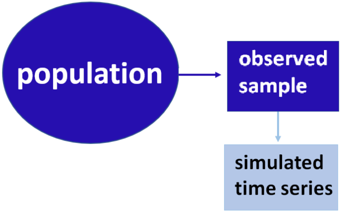
FFORMS: population



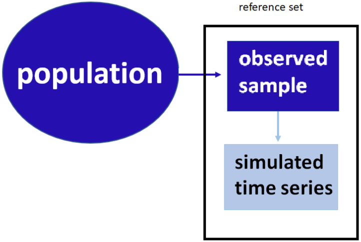
FFORMS: observed sample



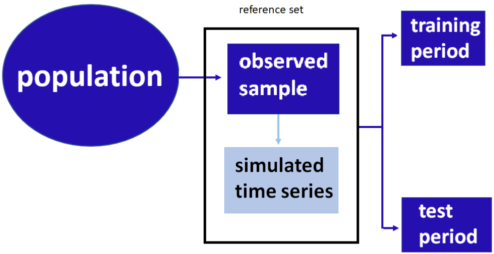
FFORMS: simulated time series



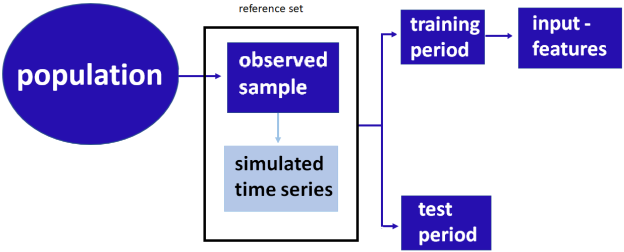
FFORMS: reference set



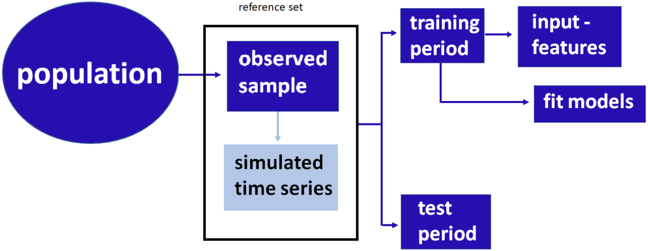
FFORMS: Meta-data



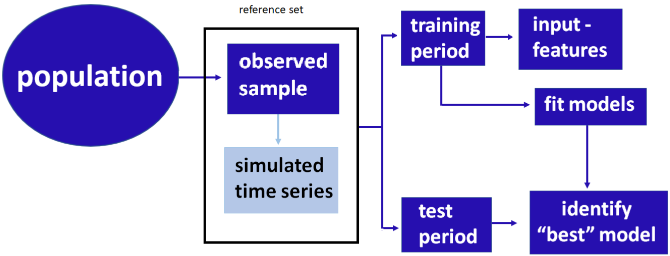
FFORMS: Meta-data



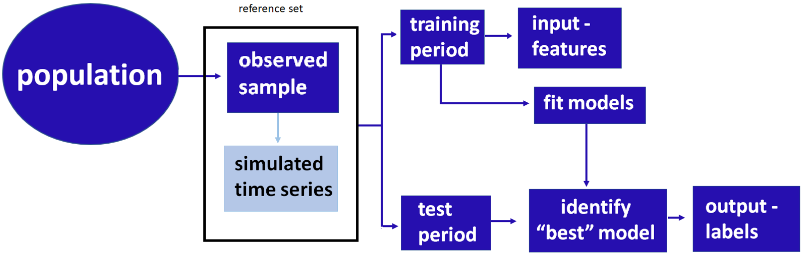
FFORMS: Meta-data



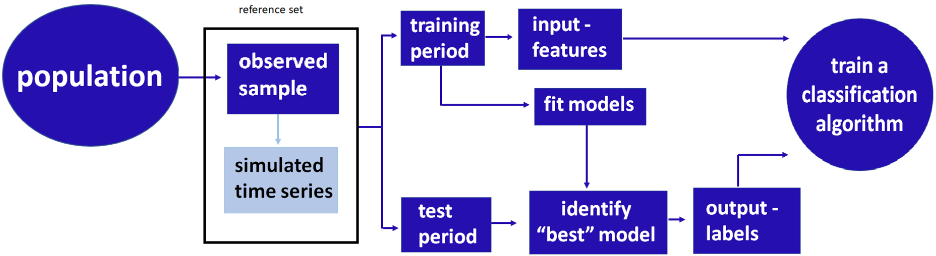
FFORMS: Meta-data



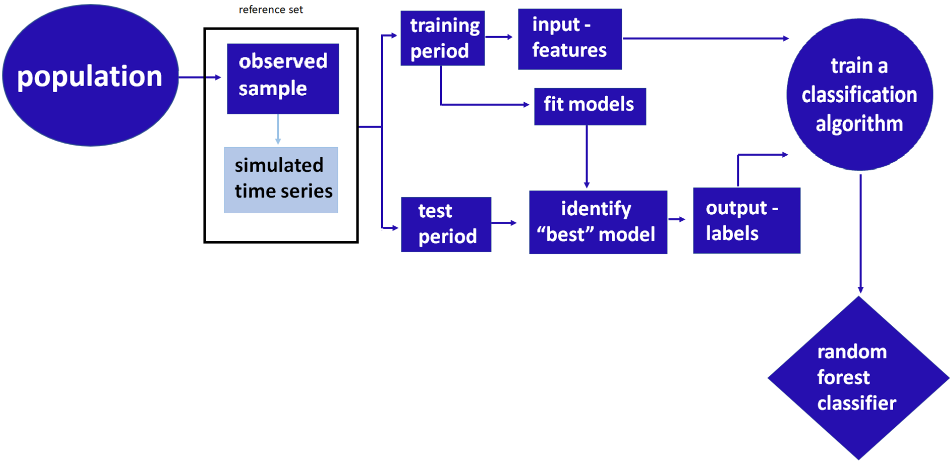
FFORMS: Meta-data



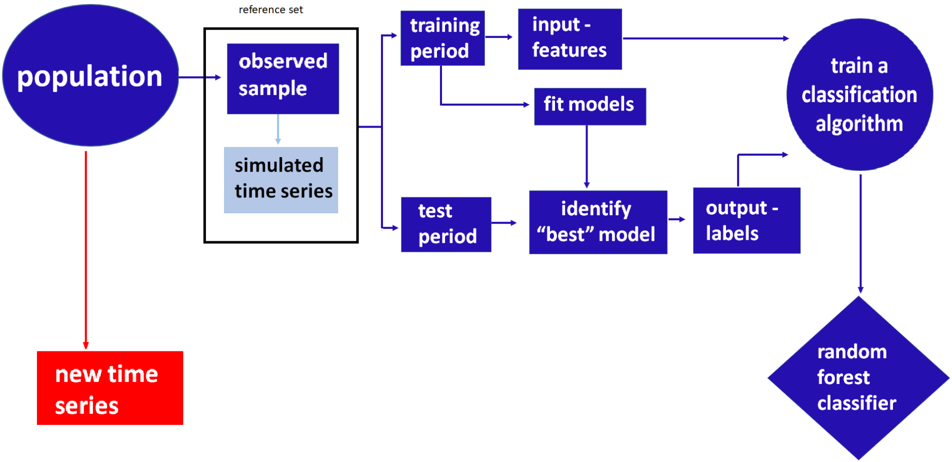
FFORMS: Meta-data



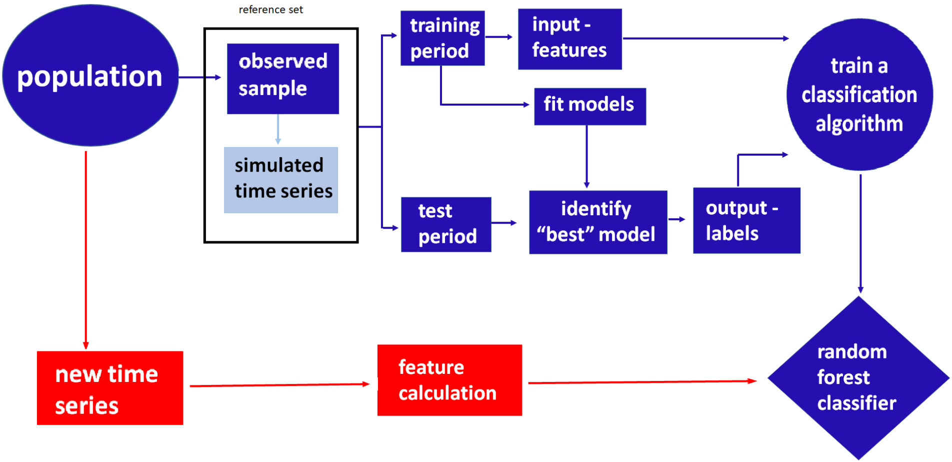
FFORMS: Random-forest classifier



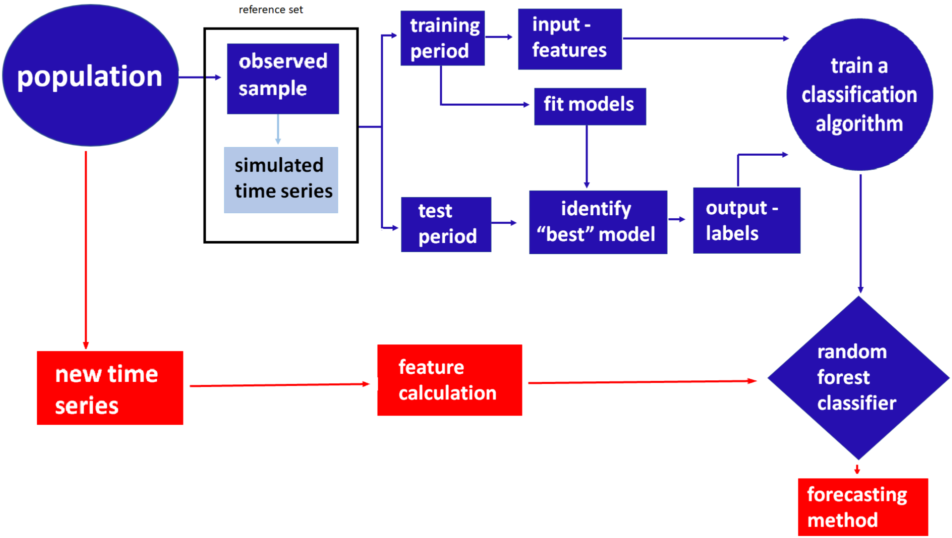
FFORMS: "online" part of the algorithm



FFORMS: “online” part of the algorithm



FFORMS: “online” part of the algorithm



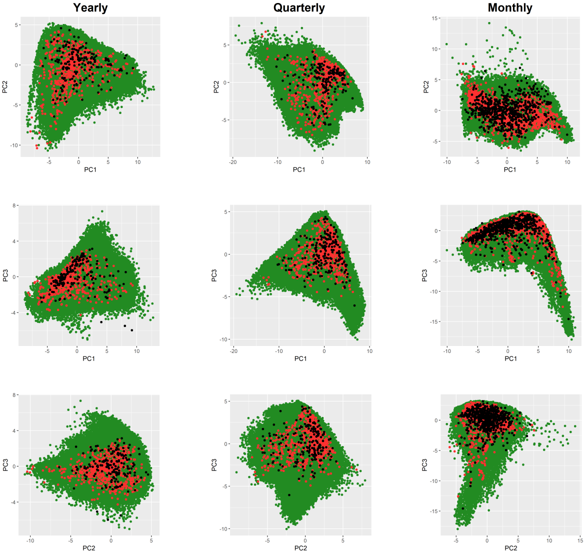
Application to M competition data

- Proposed algorithm is applied to yearly, quarterly and monthly series separately
- We run two experiments for each case.

	Experiment 1				Experiment 2			
	Source	Y	Q	M	Source	Y	Q	M
Observed series	M1	181	203	617	M3	645	756	1428
Simulated series		362000	406000	123400		1290000	1512000	285600
New series	M3	645	756	1428	M1	181	203	617

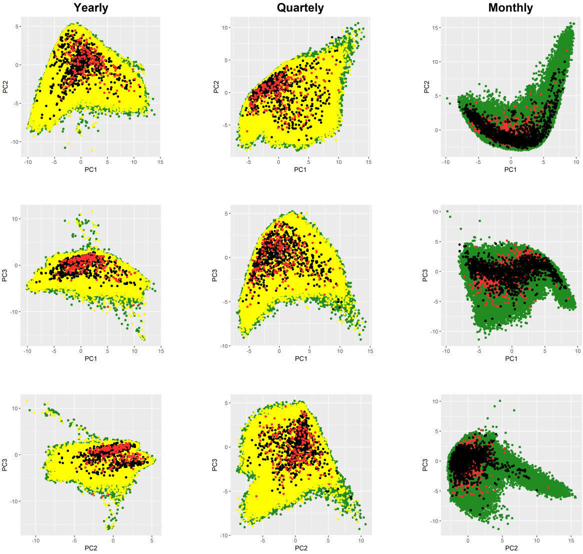
Experiment 1: Distribution of time series in the PCA space

observed - M1 simulated new - M3

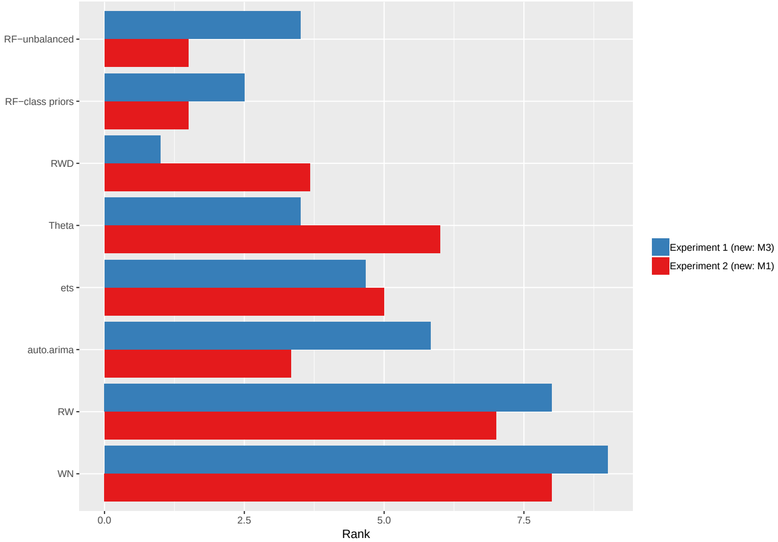


Experiment 2: Distribution of time series in the PCA space

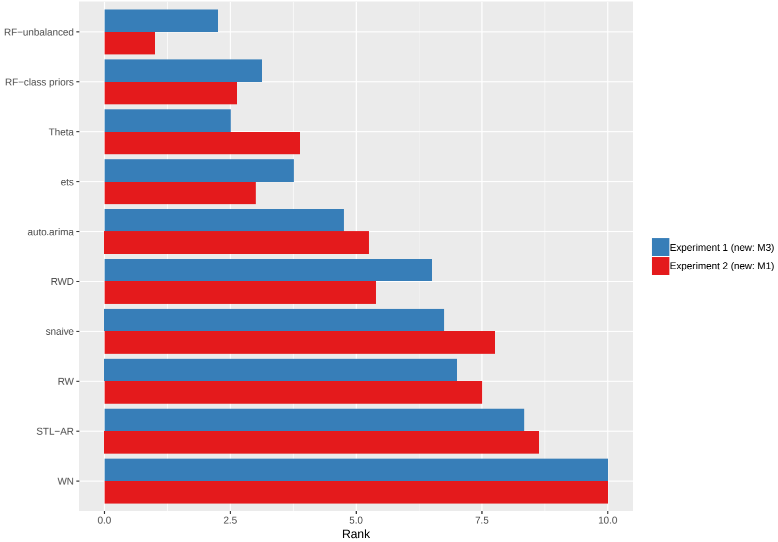
observed - M3 | simulated | subset | new - M1



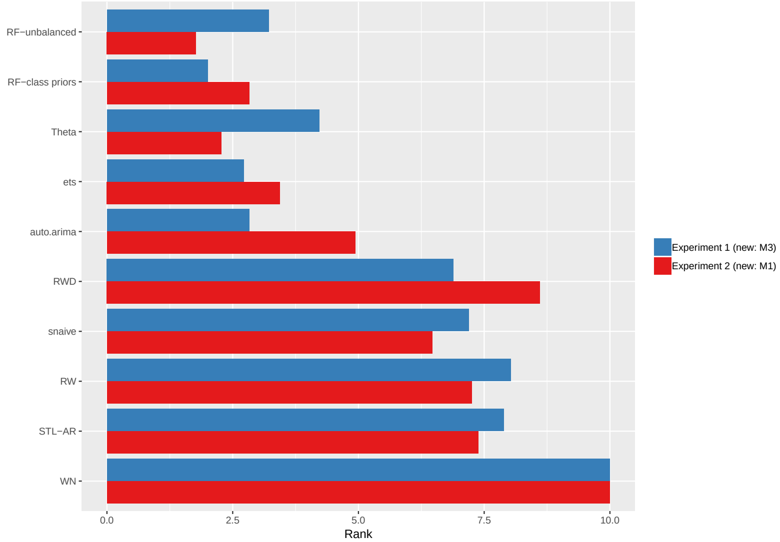
Results: Yearly



Results: Quarterly



Results: Monthly



Discussion and Conclusions

- Proposed a novel framework for forecast-model selection using meta-learning based on time series features.

Discussion and Conclusions

- Proposed a novel framework for forecast-model selection using meta-learning based on time series features.
- Our method almost always performs better than common benchmark methods, and better than the best-performing methods from the M3 competition.

Discussion and Conclusions

- Proposed a novel framework for forecast-model selection using meta-learning based on time series features.
- Our method almost always performs better than common benchmark methods, and better than the best-performing methods from the M3 competition.
- The framework is general and can be applied to any large collection of time series.

Discussion and Conclusions

- Proposed a novel framework for forecast-model selection using meta-learning based on time series features.
- Our method almost always performs better than common benchmark methods, and better than the best-performing methods from the M3 competition.
- The framework is general and can be applied to any large collection of time series.
- Advantage: Not necessary to estimate several different models for the data and undertake an empirical evaluation of their forecast accuracy on a given time series.

R package: seer

available at:

https://github.com/thiyangt/seer

R package: seer

seer

available at:

https://github.com/thiyangt/seer

Reference: Talagala, TS, RJ Hyndman & G Athanasopoulos (2018). Meta-learning how to forecast time series. Technical Report 6/18, Monash University.