Bagging-Clustering Methods to Forecast Time Series

Tiago Dantas
Fernando Cyrino

Pontifical Catholic University of Rio de Janeiro,Brazil

ISF 2017, Cairns-Australia, June 25-28

June 28, 2017

Introduction

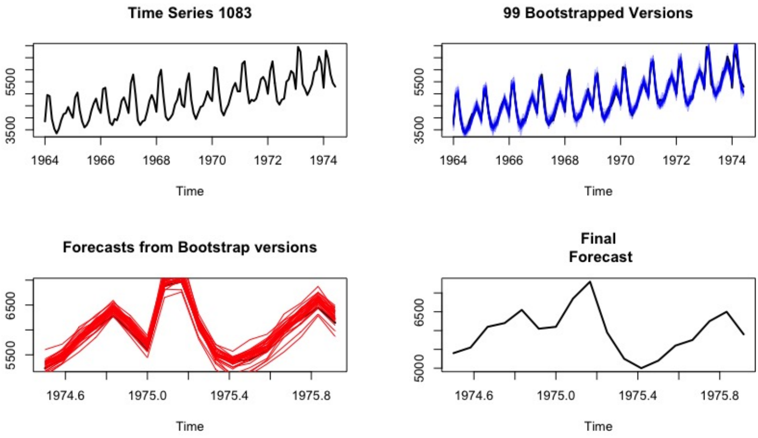
- Cordeiro and Neves (2009) and Bergmeir et al. (2016), have proposed new ways to generate forecasts using a very popular Machine Learning technique, called **Bagging** (Bootstrap Aggregating), proposed by Breiman (1996), in combination with **Exponential Smoothing** methods to improve forecast accuracy
- The main idea is to use Bootstrap to generate an ensemble of forecasts that is combined into one single output

Bagged.BLD.MBB.ETS - Bergmeir et al.(2016)

Best model using Bagging and Exponential Smoothing methods

- Box-Cox transformation (stabilizes variance)
- STL decomposition (decompose time series into seasonal, trend and remainder)
- Moving Block Bootstrap (generate new versions of the remainder)
- Forecasts are obtained selecting one ETS model for each time series (original and bootstrap versions)
- Final forecast is obtained using the median (other possibilities are mean, trimmed mean, among others)

Bagged.BLD.MBB.ETS - Bergmeir et al.(2016)



Why Bagging tends to work

The Mean Squared Forecast Error (MSFE) can be decomposed into three terms:

$$MSFE = Var(y_{t+1|t}) + bias(\hat{y}_{t+1|t})^2 + Var(\hat{y}_{t+1|t})$$

Why Bagging tends to work

The average forecast over the Bootstrap samples can be written as:

$$\tilde{y}_{t+1|t} = \frac{1}{B} \sum_{i=1}^{B} \hat{y}^{*}_{(i)t+1|t}$$

where the tilde indicates Bagging forecast and B is the total number of Bootstrap samples.

Why Bagging tends to work

$$bias(\tilde{y}_{t+1|t}) = \frac{1}{B}\sum_{i=1}^{B} bias(\hat{y}^{*}_{(i)t+1|t})$$

- Note that unbiased Bootstrapped versions lead to a **relatively unbiased ensemble**

Why Bagging tends to work

$$Var(\tilde{y}_{t+1|t}) = \frac{1}{B^2}\sum_{i=1}^{B} Var(\hat{y}^*_{(i)t+1|t}) + \frac{1}{B^2}\sum_{i \neq i'} Cov[\hat{y}^*_{(i)t+1|t}, \hat{y}^*_{(i')t+1|t}]$$

- Variance tends to be reduced

Why Bagging tends to work

- When applying Bagging and Exponential Smoothing what happens is variance reduction
- If the variances are approximately equal **and there is no correlation**:

$$Var(\tilde{y}_{t+1|t}) \approx \frac{1}{B} Var(\hat{y}^{*}_{(1)t+1|t})$$

Why Bagging tends to work

- **Reducing covariance seems to be a good idea**

$$Var(\tilde{y}_{t+1|t}) = \frac{1}{B^2}\sum_{i=1}^{B} Var(\hat{y}^*_{(i)t+1|t}) + \frac{1}{B^2}\sum_{i \neq i'} Cov[\hat{y}^*_{(i)t+1|t}, \hat{y}^*_{(i')t+1|t}]$$

- The proposed approach tries to use this idea in order to reduce forecast error

Proposed Approach

The proposed approach can be divided in two parts:

1. Generation of Bootstrapped series - Algorithm developed by Bergmeir et al. (2016)
2. The procedure to forecast and aggregate the series - **New developments**

Proposed Approach

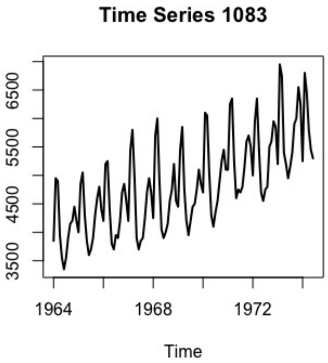
Generating bootstrapped series

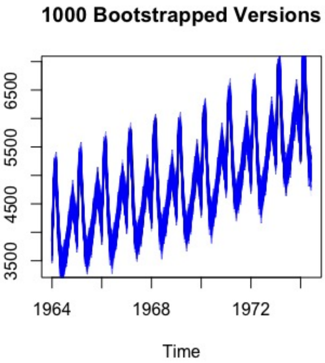
- Bootstrapped series are generated in the same way as Bagged.BLD.MBB.ETS

Algorithm 1 Generating bootstrapped series

```
procedure BOOTSTRAP(ts,num.boot)
    λ ← BoxCox.lambda(ts,min=0,max=1)
    ts.bc← BoxCox(ts,λ = 1)
    if ts is seasonal then
        [trend seasonal remainder] ← stl(ts.bc)
    else
        seasonal ← 0
        [trend,remainder] ← loess(ts.bc)
    end if
    recon.series[1] ← ts
    for i in 2 to num.boot do
        boot.sample[i] ← MBB(remainder)
        recon.series.bc[i] ← trend + seasonal +boot.sample[i]
        recon.series[i] ← InvBoxCox(recon.series.bc[i],λ)
    end for
    return recon.series
end procedure
```

Proposed Approach





Proposed Approach

The procedure to forecast and aggregate the series

- the Proposed approach and Bagged.BLD.MBB.ETS differ in the way the ensemble is constructed
- Bagged.BLD.MBB.ETS consider all of the Bootstrapped versions to make forecasts
- The proposed approach considers a less correlated group of time series to make forecasts

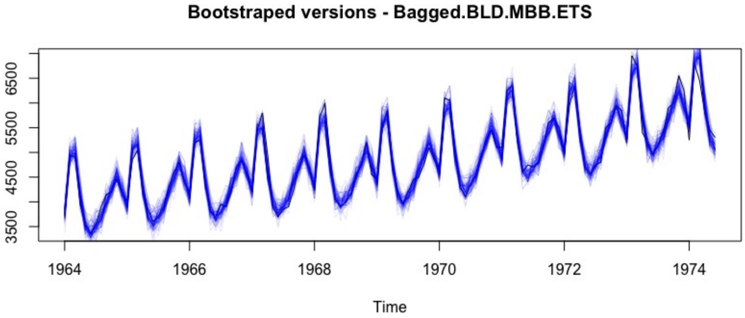
Proposed Approach

- To create a less correlated ensemble, the proposal is to generate clusters from the Bootstrapped versions
- Cluster procedures maximize similarity within the group and minimize it between them
- The expectation is that selecting series from different clusters would lead to an ensemble less correlated and, therefore, less correlated forecasts to be aggregated

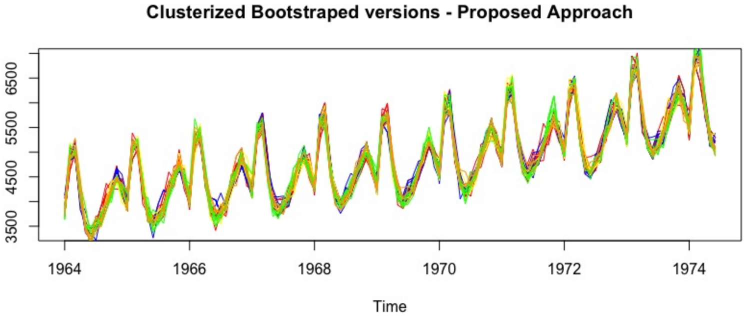
Proposed Approach

- Partitioning Around Medoids Algorithm (PAM) and euclidean distance are used to create the clusters (fast algorithm and less sensible to outliers)
- The number of cluster can be defined using cross-validation or any other method (e.g. Silhouette Information)

Proposed Approach



Proposed Approach



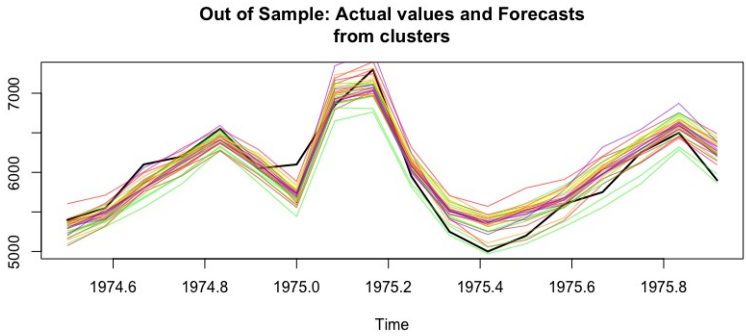
Proposed Approach

- The user has to define the total number of series to aggregate (100 was the choice made by Bergmeir and colleagues)
- The number of series to be selected in each cluster is defined as proportionally equal to the size of each cluster

Example: B =1000 and the total number of series to be aggregated is 100. If cluster 1 has 20 series, therefore 2 series would be selected (10%). But, Which 2 time series?

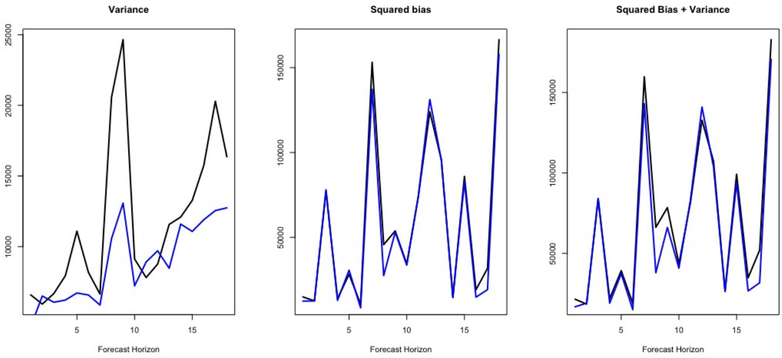
- A validation set is defined
- The time series with best results (sMAPE) in the validation set are the ones selected in each cluster
- The final forecast is obtained taking the median of the forecasts (other possibilities are the mean, trimmed mean)

Proposed Approach



MSE decomposition

Bagged.BLD.MBB.ETS (black) and the Proposed Approach (blue) - Forecast (up to 18 steps ahead) - Time series 1083 - M3 competition



Empirical Results

- The proposed approach was validated on public available time series from the M3 competition (1428 monthly, 756 quarterly and 645 yearly time series)
- The experiment was conducted using R and the majorly the forecast package (version 8.0)
- The results for Bagged.BLD.MBB.ETS were obtained using **baggedETS()**

Monthly data

Methods	Rank sMAPE	Mean sMAPE	Median sMAPE
Proposed Approach	**11.15**	**13.62**	**8.74**
Bagged.BLD.MBB.ETS	11.30	13.65	8.85
THETA	11.53	13.89	8.92
ForecastPro	11.56	13.90	8.81
COMB S-H-D	12.54	14.47	9.37
ForcX	12.76	14.47	9.21
HOLT	12.78	15.79	9.28
WINTER	13.06	15.93	9.30
RBF	13.27	14.76	9.21
AAM1	13.48	15.67	9.67
DAMPEN	13.48	14.58	9.44
AutoBox2	13.60	15.73	9.28
B-J auto	13.69	14.80	9.32
AutoBox1	13.69	15.81	9.27
SMARTFCS	13.82	15.01	9.52
Flors-Pearc2	13.84	15.19	9.61
AAM2	13.85	15.94	9.62
Auto-ANN	13.91	15.03	9.62
PP-Autocast	14.13	15.33	9.90
ARARMA	14.20	15.83	9.80
AutoBox3	14.21	16.59	9.40
Flors-Pearc1	14.54	15.99	9.96
THETAsm	14.58	15.38	9.65
ROBUST-Trend	14.79	18.93	9.73
SINGLE	15.22	15.30	10.03
NAIVE2	16.04	16.89	10.12

Quarterly data

Methods	Rank sMAPE	Mean sMAPE	Median sMAPE
THETA	**11.39**	**8.96**	5.37
COMB S-H-D	12.18	9.22	5.32
ROBUST-Trend	12.44	9.79	**5.00**
DAMPEN	12.66	9.36	5.59
PP-Autocast	12.81	9.39	5.26
ForcX	12.86	9.54	5.62
Bagged.BLD.MBB.ETS	12.96	9.80	5.81
B-J auto	13.16	10.26	5.69
ForecastPro	13.20	9.82	5.84
Proposed Approach	13.24	9.89	5.82
HOLT	13.27	10.94	5.71
RBF	13.30	9.57	5.67
AutoBox2	13.38	10.00	5.59
WINTER	13.38	10.84	5.71
Flors-Pearc1	13.48	9.95	5.61
ARARMA	13.49	10.19	6.11
Auto-ANN	13.89	10.20	6.28
THETAsm	14.18	9.82	5.65
AAM1	14.25	10.16	6.36
SMARTFCS	14.27	10.15	5.71
Flors-Pearc2	14.30	10.43	6.22
AutoBox3	14.38	11.19	6.15
AAM2	14.41	10.26	6.44
SINGLE	14.66	9.72	6.18
AutoBox1	14.69	10.96	6.14
NAIVE2	14.80	9.95	6.18

Yearly data

Methods	Rank sMAPE	Mean sMAPE	Median sMAPE
ForcX	**11.15**	16.48	11.34
RBF	11.46	**16.42**	10.74
AutoBox2	11.48	16.59	11.31
Flors-Pearc1	11.57	17.21	**10.72**
THETA	11.58	16.97	11.25
ForecastPro	11.73	17.27	11.05
ROBUST-Trend	11.81	17.03	11.30
PP-Autocast	11.87	17.13	10.83
Bagged.BLD.MBB.ETS	11.89	17.40	11.20
DAMPEN	11.92	17.36	10.95
COMB S-H-D	11.99	17.07	11.68
Proposed Approach	12.21	17.56	11.42
SMARTFCS	12.38	17.71	11.83
HOLT	12.64	20.02	11.77
WINTER	12.64	20.02	11.77
Flors-Pearc2	13.02	17.84	12.55
ARARMA	13.03	18.36	11.35
B-J auto	13.04	17.73	11.70
Auto-ANN	13.32	18.57	13.08
AutoBox3	13.52	20.88	12.89
THETAsm	13.55	17.92	12.21
AutoBox1	13.82	21.59	12.75
NAIVE2	14.16	17.88	12.37
SINGLE	14.21	17.82	12.44

Concluding Remarks

- This proposed approach make forecasts combining Bagging, Exponential Smoothing and Cluster methods
- The empirical results demonstrate the approach was capable of generating highly accurate forecasts for monthly time series
- The so far, not explicitly addressed, covariance effect on the combination of Bagging and Exponential Smoothing, is probably resposible for reducing the forecast error
- The method doesn't seem to work well on short time series (such as the case of yearly and quarterly time series from the M3 competition)

Concluding Remarks

Future work

- Other weighting schemes for selected series
- Other decomposition and forecasting methods

Thank You!

t.mendesdantas@gmail.com

References

Bergmeir, C., Hyndman, R.J., Benitez,J.M. (2016). Bagging exponential smoothing methods using STL decomposition and Box-Cox transformation.*International Journal of Forecasting*, 32(2), 303-312.

Breiman, L. (1996). Bagging Predictors.*Machine Learning*, 24, 123-140.

Cordeiro, C.,Neves,M. (2009). Forecasting time series with BOOT.EXPOS procedure. *REVSTAT-Statistical Journal*, 7(2), 135-149.

Liao, T.W. (2005). Clustering of time series data - a survey.Journal of Pattern Recognition, 38(11), 1857-1874.