Forecast Combinations in R using the ForecastComb Package

by Christoph E. Weiss and Eran Raviv and Gernot Roetzer

Abstract This paper introduces the R package **ForecastComb**. The aim is to provide researchers and practitioners with a comprehensive implementation of the most common ways in which forecasts can be combined. The package in its current version covers 15 popular estimation methods for creating a combined forecasts – including simple methods, regression-based methods, and eigenvector-based methods. It also includes useful tools to deal with common challenges of forecast combination (e.g., missing values in component forecasts, or multicollinearity), and to rationalize and visualize the combination results.

Introduction

Model uncertainties and/or model instabilities are deeply-rooted in real-life forecasting challenges. More often than not, and particularly in soft sciences including econometrics, the underlying data-generating process is not well understood.

In the business of time series forecasting, accuracy is key. The advent of "black-box" statistical learning methods (such as *Artificial Neural Networks*) testifies to the fact that forecasting practitioners and researchers alike tend to favor accuracy over explainability of a forecast model. There is a strong consensus in the social sciences that the observed processes are too complex to be modelled perfectly. Excluding some natural sciences, it is generally undisputed that the underlying data-generating process is unknown[1]. Even in the unrealistic case where the structural form of a data-generating process is more or less known, except for the parameter values that has to be estimated, a misspecified model can produce better forecasts than a correctly specified model due to parameters' estimation uncertainty[2].

Hence, statistical models are habitually too simple, misspecified, and\or incomplete; a fact that is widely accepted in theory, but less widely applied in practice, in the field of econometrics, in which researchers often still hang on to the conceptual error of assuming one true data-generating process and putting too much focus on model selection to find the one true model. Hansen (2005) takes note of this and related misconceptions of econometric forecasting practice in his essay on the challenges for econometric model selection: "models should be viewed as approximations, and econometric theory should take this seriously". An obvious alternative to choosing a single 'best' forecasting method is to combine forecasts from different models. A combined forecast can also be thought of as one which originates from a very complex model, an overarching model with different underlying simpler models in order to achieve a better data-generating process approximation. By now it is well established, empirically but in many different settings, that combining different forecasts delivers results which are "better than the best".

Following the advice in Hansen (2005), we abandon the quest for the one *true* correctly specified model, so that we are free to include information from different models. While this freedom, as seen empirically, improve forecast accuracy it is generally more intricate to interpret a forecast combined from different models compared to a forecast generated from a single model. In the context of forecasting, this means combining forecasts from different sources (models, experts). This emphasizes the aforementioned shift in approaching practical econometric forecasting problems: model misspecification cannot be fully rectified, but strategies can be found to mitigate its adverse effects on forecast quality. Selecting a single "best" forecasting model bears the risk of ending up with a model which is only accurate when evaluated using some validation sample, yet might prove unreliable when applied to new data. In the past decades, ample empirical evidence on the merits of combining forecasts has piled up; it is generally accepted that the (mostly linear) combination of forecasts from different models is an appealing strategy to hedge against forecast risk. Even though reduction of forecast risk is the main argument for using combined forecasts, it should also be noted that there are cases when combined forecasts are more accurate than even its best component (e.g. Graefe et al., 2014). While this is not theoretically sound, it is somewhat intuitive that in a continuously changing environment different forecasting models deliver different results at different time periods. For example, it is reasonable for one method to perform well in a low-volatility regime, and for another method to perform poorly in that regime, but perform well in a high-volatility regime. Strong empirical support for this argument of change in relative performance of different methods over time is found among

---

[1] Referring for example to George's Box well rehearsed aphorism: "All models are wrong but some are useful".
[2] We thank an anonymous referee for pointing this out.

others by Elliott and Timmermann (2005).

"Hedging" against model risk by combining different forecasts is intuitive and appealing. Joined with accuracy gains, the idea is widely adopted in macroeconomics and finance, with application abound. Magnus and Wang (2014) explore growth determinants, Stock and Watson (2004) use forecast combination for output forecasting, Wright (2009) and Kapetanios et al. (2008) for inflation forecasting, Avramov (2002), Rapach et al. (2010) and Ravazzolo et al. (2007) for forecasting stock returns, Wright (2008) for exchange rate forecasts, Andrawis et al. (2011) for forecasting inbound tourism figures, Nowotarski et al. (2014) and Raviv et al. (2015) for electricity price forecasting, and Weiss (2017) for healthcare demand forecasting. More recently, forecast combinations have been applied not only in "first moment" applications, but for higher moments as well. For example in Christiansen et al. (2012) for volatility forecasting, and Opschoor et al. (2014) for Value-at-Risk forecasting.

The theoretical foundations of forecast combination date back five decades to the seminal papers of Crane and Crotty (1967) and Bates and Granger (1969). Yet even in the recent past, papers discussing new combination techniques are published in reputable journals and stimulate further research still (Hansen, 2007, 2008; Hansen and Racine, 2012; Elliott et al., 2013; Morana, 2015; Cheng and Yang, 2015). Two extensive reviews of the literature, techniques and applications of forecast combinations are Clemen (1989) and Timmermann (2006).

Given the topic's popularity in both theoretical and empirical research, it is somewhat surprising that very few combination techniques are readily available in R: There are some packages that cover specialized specific topics related to combination methods, e.g. the **BMA** package by Raftery and Painter (2005) for Bayesian model averaging, as well as the **opera** package by Gaillard and Goude (2016) and the **forecastHybrid** package by Shaub and Ellis (2017). However, as of now the only two R packages that are entirely devoted to forecast combination are the **ForecastCombinations** package by Raviv (2015), which focuses on regression-based combination methods, and the **GeomComb** package by Weiss and Roetzer (2016), which focuses on geometric, eigenvector-based combination methods.

Aiming to improve user experience, we have merged these last two aforementioned packages, providing one unified package for the widest range of forecast combination approaches available today in R. The package is flexible and provides enough guidance for users familiar or unfamiliar with the world of forecasting. We have made both regression-based and eigenvector-based combination methods available to users in a single standardized framework based on S3 classes and methods. The logic behind this choice is that comparing regression-based and eigenvector-based combination methods is often insightful – as pointed out by Hsiao and Wan (2014), the conditions under which these two approaches perform well differ from each other: regression-based methods tend to produce more accurate forecasts when one or a few of the individual forecasts are considerably better than the rest, while eigenvector-based methods perform better when the individual forecasts are in the same ballpark. This paper presents the functionalities made available in the package and demonstrates how to implement them in an empirical exercise.

In the following section we review the forecast combination methods that are available in the **ForecastComb** package. We then provides a detailed implementation description using the package, with a specific empirical example – we combine univariate time series forecasts for UK energy supply.

Forecast combinations

Since the seminal paper by Bates and Granger (1969), myriad combination strategies have been put forward in theoretical and empirical literature. The best way to combine different forecasts has no theoretical underpinnings, a lot depends on the specifics of the data at hand. Andrawis et al. (2011) even suggest to use hierarchical forecast combinations, i.e. combining combined forecasts. The purpose here is not to investigate which combination method is suitable in which context. Rather, we provide a broad range of options for the user to explore and experiment with. Again, owing to a lack of theoretical foundation or the empirical lack of evidence for a one dominant way in which forecasts should be combined.

To fix notations, denote $F_{T \times P}$ as the matrix of forecast with dimension $T \times P$ where T is the number of rows and P is the number of columns (so we have P forecasts at each point in time). Denote f_i as the forecast obtained using model i, $i \in \{1 \dots P\}$. When there is no danger of confusion, we omit the additional subscript t which denotes the time dimension of the forecast. Some combination methods require an ordering of the component forecasts. When this is the case, $f_{(i)}$ denotes the ith order statistic of the cross-section of component forecasts. Finally, the weight given to that forecast in the overall combined forecast is denoted as w_i, and the combined forecast as f^c.

Frequently used schemes for forecast combinations

Simple Combination Methods

We now present few simple ways in which forecasts can be combined. They are simple in that there is no need to exactly estimate the weight each forecast should be given in the overall combination. We just use some location measure (for example the average) of the cross-sectional distribution of the individual forecasts. In theory some location measures are the same if the cross-sectional distribution is symmetric, but if the distribution is asymmetric, and\or if there is one method which produces extreme forecasts then the following few combination methods become pertinent.

1. *Simple Average*. The most intuitive approach to combine forecasts is using the average of all those forecasts. Over the years this innocent approach has established itself as an excellent benchmark, despite or perhaps because of its simplicity (e.g. Genre et al., 2013). The combined forecast is straightforwardly given by

$$f^c = \frac{1}{P} \sum_{i=1}^{P} f_i.$$ (1)

   Clemen (1989) argues that this equal weighting of component forecasts is often the best strategy in this context. This is still true almost thirty years later and called the "forecast combination puzzle", a term coined by Stock and Watson (2004). A rigorous attempt to explain why simple average weights often outperform more sophisticated forecast combination techniques is provided in a simulation study by Smith and Wallis (2009), who ascribe this surprising empirical finding to the effect of finite-sample error in estimating the combination weights. Recently, Claeskens et al. (2016) provide a theoretical argumentation to these empirical findings. The authors make the case that lower estimation noise, when the weights are determined rather than estimated, goes a long way in explaning the puzzle. A more detailed overview of the empirical support for the "forecast combination puzzle" can be found in Graefe et al. (2014).

2. *Median*. Another fairly simple and appealing combination method is using the median of the component forecasts. The median is insensitive to outliers, which can be relevant for some applications. Palm and Zellner (1992) suggest that simple averaging may not be a suitable combination method when some of the component forecasts are biased. This calls for the use of another location measures which is robust to outliers. The median method is an appealing, rank-based combination method that has been used in a wide range of empirical studies (e.g. Armstrong, 1989; McNees, 1992; Hendry and Clements, 2004; Stock and Watson, 2004; Timmermann, 2006).
   For the median method, the combined forecast is given by:

   • For odd P:

   $$f^c = f_{\left(\frac{P}{2}+0.5\right)}$$ (2)

   • For even P:

   $$f^c = \frac{1}{2}\left(f_{\left(\frac{P}{2}\right)} + f_{\left(\frac{P}{2}+1\right)}\right)$$ (3)

3. *Trimmed Mean*. Another outlier-robust location measure that is commonly used is the trimmed mean (e.g. Armstrong, 2001; Stock and Watson, 2004; Jose and Winkler, 2008).
   Using a trim factor λ (i.e. the top/bottom $100 \times \lambda$% are trimmed) the combined forecast is calculated as:

   $$f^c = \frac{1}{P(1-2\lambda)} \sum_{i=\lambda P+1}^{(1-\lambda)P} f_{(i)}$$ (4)

   Typically, we use $\lambda = 0.1$ indicating we trim the top and bottom 10% of the most extreme component forecasts, excluding those from the computation of the combined forecast. The trimmed mean is an interpolation between the simple average ($\lambda = 0$) and the median ($\lambda = 0.5$).

4. *Winsorized Mean*. Like the trimmed mean, the winsorized mean is a robust statistic that is less sensitive to outliers than the simple average. It takes a softer line when handling outliers: Instead of altogether removing them as in the trimmed mean approach, it caps outliers at a

certain level. By capping outliers rather than removing them, we allow for at least some degree of influence. For this reason, the measure is sometimes preferred, for example by Jose and Winkler (2008).

Let λ be the trim factor (i.e. the top/bottom $100 \times \lambda\%$ are winsorized) and $K = \lambda P$. The combined forecast is then calculated as:

$$f^c = \frac{1}{P} \left[K f_{(K+1)} + \sum_{i=K+1}^{P-K} K f_{(P-K)} \right] \tag{5}$$

5. *Bates/Granger (1969)*. In their seminal paper, Bates and Granger (1969) introduced the idea of combining forecasts. Their approach builds on portfolio diversification theory and uses the diagonal elements of the estimated mean squared prediction error matrix in order to compute combination weights:

$$f^c = \sum_{i=1}^{P} f_i' \times \frac{\hat{\sigma}^{-2}(i)}{\sum_{j=1}^{P} \hat{\sigma}^{-2}(j)} \tag{6}$$

where $\hat{\sigma}^2(i)$ is the estimated mean squared prediction error of model i.

The approach ignores correlation between component forecasts due to difficulties in precisely estimating the covariance matrix.

6. *Newbold/Granger (1974)*. Building on the earlier research by Bates and Granger (1969), the methodology of Newbold and Granger (1974) also extracts the combination weights from the estimated mean squared prediction error matrix.

Let Σ be the mean squared prediction error matrix of $F_{N \times P}$ and e be a $P \times 1$ vector of $(1, 1, ..., 1)'$. Newbold and Granger (1974)'s method is a constrained minimization of the mean squared prediction error using the normalization condition $e'w = 1$. This yields the following combined forecast:

$$f^c = F_{N \times P} \times \frac{\Sigma^{-1} e}{e' \Sigma^{-1} e} \tag{7}$$

While the method dates back to Newbold and Granger (1974), the variant of the method we use in the **ForecastComb** package does not impose the prior restriction that Σ is diagonal. This approach, used by Hsiao and Wan (2014), is a generalization of the original method.

7. *Inverse Rank*. The inverse rank approach, suggested by Aiolfi and Timmermann (2006), ranks the forecast models based on their performance up to time N. The model with the lowest mean squared prediction error is assigned the rank 1, the model with the second lowest mean squared prediction error is assigned the rank 2, etc. The combined forecast is then calculated as follows:

$$f^c = \sum_{i=1}^{P} f_i' \times \frac{Rank_i^{-1}}{\sum_{j=1}^{P} Rank_j^{-1}} \tag{8}$$

Timmermann (2006) points out that this method, just like Bates and Granger (1969), also ignores correlations across forecast errors. However, the method is more robust to outliers, since total rankings are not likely to change dramatically by the presence of extreme forecasts.

Regression-based Combination Methods

8. *Ordinary Least Squares (OLS) regression*. The idea to use regression for combining forecasts was put forward by Crane and Crotty (1967) and successfully driven to the forefront by Granger and Ramanathan (1984). Using this approach, the combined forecast is a linear function of the individual forecasts where the weights are determined using a regression of the individual forecasts on the target itself:

$$y = \alpha + \sum_{i=1}^{P} w_i f_i + \varepsilon, \tag{9}$$

Using a portion of the forecasts to train the regression model, the OLS coefficients can be estimated by way of minimizing the sum of squared errors in equation(8). The combined

forecast is then given by:

$$f^c = \widehat{\alpha} + \sum_{i=1}^{P} \widehat{w}_i f_i,\tag{10}$$

An advantage of the OLS forecast combinations is that the combined forecast is unbiased due to the intercept in the equation, even if one of the individual forecasts is biased. A disadvantage is that the method places no restriction on the combination weights (i.e. they do not add up to 1 and can be negative), which complicatesinterpretation, especially if the coefficients are non-convex.[3]

9. *Least Absolute Deviation (LAD) regression.* While the OLS regression estimates the coefficients in equation ([10]) by minimizing the sum of squared errors, we may want to estimate those coefficients differently, minimizing a different loss function[4], for example the absolute sum of squares. The reason is best explained using an example: Assume we have a model that performs well in general, yet every now and then misses the target by a very large margin. Such a model would be weighted more heavily under the LAD scheme than under the OLS scheme since those large but infrequent errors will be more heavily penalized using OLS. Whether this is beneficial depends on the user's preference and/or the cost of missing the target given the problem at hand. It should be noted that this lower sensitivity to outliers has another advantage: OLS weights can be unstable when predictors are highly correlated, which is the norm in forecast combination. Minor fluctuations in the sample can cause major shifts in the coefficient vector ('bouncing betas'), often leading to poor out-of-sample performance. This suggests that LAD combination should be favored in the presence of highly correlated component forecasts (Nowotarski et al., 2014).

10. *Constrained Least Squares (CLS) regression.* Like the LAD approach, CLS addresses the issue of 'bounding betas'. It does so by minimizing the sum of squared errors under some additional constraints. Specifically, we constrain the estimated coefficients $\{w_i\}$, allowing only for positive solutions: $w_i \geq 0 \ \forall i$, and to sum up to one: $\sum_{i=1}^{P} w_i = 1$. The solution requires numerical minimization, but good optimization algorithms are readily available: The **ForecastComb** package relies on the function solve.QP available from the **quadprog** package (Turlach and Weingessel, 2013). To tackle problems with high (but imperfect) collinearity that can cause errors in the CLS estimation, we also implement a revised Cholesky decomposition based on Ridge regression which has been propsed by Babaie-Kafaki and Roozbeh (2017) and can mitigate issues with multicollinearity.

    Theoretically, the additional constraints set CLS sub-optimally compared to the OLS. It lacks the good asymptotic properties admitted by OLS. However, in practice it is often found to perform better, especially so when the individual forecasts are highly correlated. In addition, the CLS weights are more easily interpretable. It is hard to justify a non-convex linear combination of two forecasts, while CLS weights can be conveniently interpreted for example as percentages devoted to each of the individual forecasts.

11. *Complete subset regression.* The **ForecastComb** package allows the relatively new idea of computing forecast combination weights using complete subset regression. The underlying idea is relatively straightforward: With P component forecasts that can serve as predictors in the regression model, we can form n regression models, each with a unique subset of predictors. n, the total number of combined forecasts from regression models is given by

$$n = \sum_{i=1}^{P} \binom{P}{i} = \sum_{i=1}^{P} \frac{P!}{i!(P-i)!}\tag{11}$$

    In the most basic variant, the final combined forecast is obtained by taking the simple average over the cross-section of these n combined forecasts from complete subset regressions. The method is proposed by Elliott et al. (2013) who develop the theory behind this estimator and present favourable results from simulations and empirical application for US stock returns. Admittedly, the scheme is computationally expensive, and thus additional computational resources may be required if P is in the dozens (Elliott et al. (2013), Section 2.5.1 proposes a workaround based on random sampling from P). Additionally, since all n forecasts are returned,

---

[3]if the combination of two individual forecasts is not convex, the resulting combined forecast will not necessarily lie between those individual forecasts. As an example, we can look at the first quarter (2014 or 2015) GDPplus series estimate, published by the Federal Reserve Bank of Philadelphia. The combined prediction of two individual estimates of the US GDP; one based on the expenditure-side and one based on the income-side, lied above the sum of those two estimate. Intuitively, this is hard to communicate (https://www.philadelphiafed.org/research-and-data/real-time-center/gdpplus).

[4]For a discussion of optimal forecast combinations under general loss functions, see Elliott and Timmermann (2004)

the user can freely refine the technique further – for example by choosing not to average over all n forecasts, but some partial subset. Using the median instead of the mean is another option that comes to mind.

Obtaining n combined forecasts from the complete subset regression also allows us to use a frequentist approach to forecast combination, also known as information-theoretic forecast combinations. In the **ForecastComb** package, several information criteria are available in the complete subset regression method, each with its own merits and weaknesses: By far the most common are the AIC (Akaike's information criterion) and the BIC (Bayesian information criterion, also known as the Schwarz information criterion). Both are supplied in addition to the corrected AIC (Hurvich and Tsai, 1989) and the Hannan Quinn information criterion (Hannan and Quinn, 1979).

Formally, the weight given to each forecast based on the information-theoretic forecast combinations is the following:

$$w_i = \frac{\exp(-1/2\varrho_i)}{\sum_{j=1}^n \exp(-1/2\varrho_j)}, \qquad (12)$$

where ϱ_i is the information criterion for forecast i obtained using a regression with a specific combination of forecasts. The value of n is fixed as the number of possible combinations, and the combined forecast is given by:

$$f^c = \sum_{i=1}^n w_i \tilde{f}_i. \qquad (13)$$

It is worth noting that this is a two-step combination method. The first step is the computation of n combined forecasts $\tilde{f}_i$ using the complete subset regression method with the original P forecasts as predictors; the second step is the combination of these combined forecasts using the weights based on information criteria.

One advantage of this frequentist approach to model averaging is that the amount of shrinkage enforced on each individual forecast is data driven. The specification of a shrinkage hyper-parameter, which is required in the corresponding Bayesian framework (e.g. Raftery and Painter, 2005) is spared from the user in this case.

Eigenvector-based Combination Methods

The eigenvector-based forecast combination methods, proposed by Hsiao and Wan (2014), are based on the idea of minimizing the mean squared prediction error subject to a normalization condition.

The most commonly used normalization condition for this purpose is to require the combination weights to add up to one, i.e. $\sum_{i=1}^P w_i = 1$ (e.g. Newbold and Granger, 1974; Timmermann, 2006). Hsiao and Wan (2014) show that this normalization condition leads to a constrained minimum of the mean squared prediction error (MSPE), and propose a normalization condition that leads to an unconstrained minimum of the MSPE:

$$\sum_{i=1}^P w_i^2 = 1 \qquad (14)$$

This unconstrained minimum of the MSPE is the basis of the four eigenvector-based approaches in the **ForecastComb** package.

12. *Standard Eigenvector Approach*. The standard eigenvector approach retrieves combination weights from the estimated MSPE matrix as follows: The P positive eigenvalues of the MSPE matrix are arranged in increasing order ($\Phi_1 = \Phi_{min}, \Phi_2, ..., \Phi_P$), and κ_i denotes the eigenvector corresponding to Φ_i. Let $d_i = e'\kappa_i$ with e being a $P \times 1$ vector of $(1, 1, ...1)'$. The combination weights w are then chosen corresponding to the minimum of $(\frac{\Phi_1}{d_1^2}, \frac{\Phi_2}{d_2^2}, ..., \frac{\Phi_P}{d_P^2})$, denoted as κ_l, as:

$$w = \frac{1}{d_l}\kappa_l \qquad (15)$$

The combined forecast is then obtained as usual:

$$f^c = \sum_{i=1}^P f_i' w_i \qquad (16)$$

13. *Bias-Corrected Eigenvector Approach*. The bias-corrected eigenvector approach builds on the idea that if one or more of the component models yield biased forecasts, the accuracy of the

standard eigenvector approach can be improved by eliminating the bias. It modifies the standard approach by decomposing forecast errors into three parts: model-specific bias, omitted common factors of all component models, as well as an idiosyncratic part that is uncorrelated across the component models.

The optimization procedure to obtain combination weights coincides with the standard approach, except that we use as an input the centered MSPE matrix, i.e. after extracting the bias by subtracting the column means of the MSPE:

$$w = \frac{1}{\tilde{d}_l}\tilde{\kappa}_l \tag{17}$$

where $\tilde{d}_i$ and $\tilde{\kappa}_i$ are defined analogously to d_i and κ_i in the standard eigenvector approach with the only difference that they correspond to the spectral decomposition of the centered MSPE matrix rather than the original (uncentered) MSPE matrix.

The combined forecast is then obtained by:

$$f^c = \alpha + \sum_{i=1}^{P} f_i' w_i \tag{18}$$

where the intercept α corrects for the potential bias.

14. *Trimmed Eigenvector Approach.*

The standard approach is highly sensitive to the disparities in performance of different predictive models, i.e. the standard eigenvector approach's performance could be severely impaired by one or more component models that produce poor forecasts. This is due to treating uncertainties in the actual series, y, and the uncertainties of the component models, $F_{N \times P}$, symmetrically. For a detailed discussion of this so-called orthogonality principle, see Section 3 in Hsiao and Wan (2014). The trimmed eigenvector approach takes note of this issue.

The idea of trimming the pool of input forecasts has been used by Aiolfi and Timmermann (2006) and is picked up by Hsiao and Wan (2014) using the eigenvector framework – the weights are computed exactly as in the standard eigenvector approach, but based on the MSPE matrix of the trimmed forecasts, after discarding particularly bad component models.

15. *Trimmed Bias-Corrected Eigenvector Approach.* The underlying methodology of the trimmed bias-corrected eigenvector approach is the same as the bias-corrected eigenvector approach: The weights are retrieved through the spectral decomposition of the centered MSPE matrix.

The only difference to the bias-corrected eigenvector approach is that this method, like the trimmed eigenvector approach, pre-selects component models that serve as input for the forecast combination; only a subset of the available forecast models is retained, while the models with the worst performance are discarded, thereby combining the favourable modifications of the previous two methods.

There are three more related functions. The function `auto_combine` computes the fit for all the available forecast combination methods and is based on a grid-search optimization. It returns the combined forecast with the best in-sample accuracy. The default accuracy metric is the RMSE and MAE or MAPE are also available. The function `rolling_combine` is simply a dynamic variant, where the computation is done in an expanding window fashion rather than a single in- and out-of-sample split. Finally, the function `predict.foreccomb_res` allows to obtain the forecasts using previously trained forecast combination model. It is a simple utility function to quickly get the forecasts in an uncomplicated manner.

Implementation

The main functions provided in the **ForecastComb** package can be classified in 3 categories:

- Data Preparation
- Estimation of Forecast Combination
- Post-Fit Presentation of Results

In addition, some auxiliary functions are provided. Table 1 gives the list of the main functions.

Function	Description
Data Preparation Functions	
foreccomb	Transform raw input data for forecast combination
cs_dispersion	Compute cross-sectional dispersion
Forecast Combination Functions	
Simple Forecast Combination Functions	
comb_BG	Bates/Granger (1969) forecast combination
comb_InvW	Inverse Rank forecast combination
comb_MED	Median Forecast Combination
comb_NG	Newbold/Granger (1974) forecast combination
comb_SA	Simple Average forecast combination
comb_TA	Trimmed Mean forecast combination
comb_WA	Winsorized Mean forecast combination
Regression-Based Forecast Combination Functions	
comb_CLS	Constrained Least Squares (CLS) forecast combination
comb_CSR	Complete Subset Regression forecast combination
comb_LAD	Least Absolute Deviation (LAD) forecast combination
comb_OLS	Ordinary Least Squares (OLS) forecast combination
Eigenvector-Based Forecast Combination Functions	
comb_EIG1	Standard Eigenvector forecast combination
comb_EIG2	Bias-Corrected Eigenvector forecast combination
comb_EIG3	Trimmed Eigenvector forecast combination
comb_EIG4	Trimmed Bias-Corrected Eigenvector forecast combination
Other Forecast Combination Functions	
auto_combine	Automated grid-search forecast combination
rolling_combine	Rolling forecast combination (time-varying combination weights)
Post-Fit Functions	
plot.foreccomb_res	S3 method to plot results from a forecast combination model
summary.foreccomb_res	S3 method – summary of the forecast combination estimation
predict.foreccomb_res	S3 method to create forecasts using weights from previously trained forecast combination model

Table 1: Brief description of main functions available in the **ForecastComb** package

Data Preparation

The **ForecastComb** package considers as a starting point that the user has already obtained a set of component forecasts, either from survey data or using statistical techniques, and now seeks to improve accuracy by combining those component forecasts into one. If the user only has the actual time series data, other packages in R can be used to create a set of component models. For instance, the **forecastHybrid** package by Shaub and Ellis (2017) which creates several univariate forecasts using methods available in the mature and popular **forecast** package (Hyndman, 2017).

The method `foreccomb` is the workhorse in the data preparation step. It supports the user with transforming the raw input data to make sure that the estimation of the forecast combinations will run smoothly.

The call of `foreccomb` is:

```
foreccomb(observed_vector, prediction_matrix, newobs = NULL,
  newpreds = NULL, byrow = FALSE, na.impute = TRUE, criterion = "RMSE")
```

The function requires user input for the parameters `observed_vector` (a vector, the actual data) and `prediction_matrix` (a matrix, the set of component forecasts to be combined). The format of the input matrix is as follows: Each column contains the forecasts from one of the P component models. Each row corresponds to the cross-section of component forecasts at a specific point in time. A situation where the format of the data is reversed, meaning that rows correspond to forecast models and columns correspond to the time index, is handled by setting the argument `byrow = TRUE`.

The `foreccomb` function includes some convenient features that take note of the fact that in many cases combination methods are applied to survey forecasts and the challenges that come along with this:

- *Split into Training Set and Test Set.* The function allows the user to specify a training set (`observed_vector` and `prediction_matrix`)and a test set (`newobs` and `newpreds`) separately. This is useful since most combination functions have to estimate the weights (requiring part of the sample to be dedicated to that task), while it is recommended that a test set is available to evaluate the model's performance on "new" data.

- *Missing Value Imputation.* Survey forecasts usually include missing values. This can be either because some of the survey participants did not respond or because the set of survey participants is changed. The `foreccomb` function provides two alternatives to deal with missing values: The default option (`na.impute = TRUE`) uses the missing value imputation algorithm from the **mtsdi** by Junger and de Leon (2012). It is a modified version of the EM algorithm for imputation that is specifically adjusted for multivariate time series data, accounting for correlation between the forecasting models and the time structure of the series itself. A smoothing spline is fitted to each of the time series at every iteration and the degrees of freedom of each spline are chosen by cross-validation. Alternatively, the argument can be set to `na.impute = FALSE`, which means the component forecast models that include any missing values are dropped prior to estimating forecast combinations, and the user is notified in the console if so relevant.

- *Handling Multicollinearity.* More often than not component forecasts that are used in the forecast combination are highly correlated. This can trouble the estimation process which does not handle well perfect collinearity. The `foreccomb` function has an inherent algorithm that checks the set of component forecasts for perfect multicollinearity, and if detected, drops one of the component models from the input data. The algorithm is designed to minimize the cost of dropping one or more models from the input data, in the sense that out of the models that cause perfect multicollinearity, it drops the least accurate forecast model. Only perfect multicollinearity is handled. By default, Root Mean Squared Error (RMSE) is used as the accuracy metric, but alternatively the user may choose the Mean Absolute Error (MAE) or the Mean Absolute Percentage Error (MAPE) by changing the argument `criterion`.

The output of the `foreccomb` function is an object of S3 class `foreccomb` that can be passed on to the estimation functions or the other auxiliary functions, for instance the function `cs_dispersion` which computes the cross-sectional dispersion of the set of component forecasts.

This is often helpful for selecting a suitable combination method: One of the main findings of Hsiao and Wan (2014) is that regression-based methods produce more accurate forecasts when one or a few of the component forecasts are much better than the rest, while eigenvector-based methods perform better when there is low dispersion among the component forecasts. The `cs_dispersion` function can be used to compute and plot this cross-sectional dispersion using standard deviation (default), interquartile range, or range.

Estimation of Forecast Combination

The package provides the user with functions for the 15 estimation techniques for combined forecasts, which were described in previous Section. The estimation functions require an object of S3 class `foreccomb` as input, which is obtained using the methods from the previous subsection.

Four of the methods include trimming, i.e. a pre-selected subset of the full set of component models that should be used in the estimation of combination weights. These are:

- Trimmed Mean (`comb_TA`)
- Winsorized Mean (`comb_WA`)
- Trimmed Eigenvector Approach (`comb_EIG3`)
- Trimmed Bias-Corrected Eigenvector Approach (`comb_EIG4`)

For these methods, the user has some flexibility. The package provides the option to set the trimming factor (or, for the eigenvector methods, the number of retained component models) manually. Otherwise, an inbuilt optimization algorithm is used for choosing the trimming factor such that the combined forecast has the best possible fit. Again, this optimization can be based on either RMSE, MAE, or MAPE, which are controlled by the argument `criterion`.

A simple simulation example:

```
R> actual <- rnorm(100)
R> forecasts <- matrix(rnorm(1000, 1), 100, 10)

R> input_data <- foreccomb(actual, forecasts)

R> # Manual Selection of Trimming Factor:
R> model1 <- comb_TA(input_data, trim_factor = 0.3)

R> # Assess accuracy of the combined forecast:
R> model1$AccuracyTrain


                  ME     RMSE     MAE     MPE     MAPE      ACF1 Theil's U
Training Set -1.07312 1.520668 1.274116 146.411 486.3501 -0.0480562 1.698456

R> # Algorithm-Optimized Selection of Trimming Factor:
R> model2 <- comb_TA(input_data, criterion = "RMSE")

Optimization algorithm chooses trim factor for trimmed mean approach...
Algorithm finished. Optimized trim factor: 0.1

R> # Assess accuracy of the combined forecast:
R> model2$AccuracyTrain


                  ME     RMSE     MAE     MPE     MAPE      ACF1 Theil's U
Training Set -1.063363 1.515091 1.27304 134.7211 489.4353 -0.03678255 1.692617
```

As can be seen, the automated selection of the trimming factor leads to an improved accuracy of the combined forecast.

The 15 methods included in the package all produce static combination weights, i.e. the models use the training set data to estimate combination weights, which will in turn be applied to all periods of the test set. The research community in the forecasting field is strongly divided in the assessment of the value of time-varying combination weights, since putting higher weights on more recent data tends to increase the parameter variance. Section 4.1 in Timmermann (2006) reviews the advantages and challenges of allowing for time-varying weights.

While the **ForecastComb** mainly uses time-invariant combination weights, the user is provided with some flexibility. The `rolling_combine` function allows for the estimation of each of the methods with time-varying weights. The approach builds on the idea of time series cross-validation (Bergmeir et al., 2015), using the provided training set as a departure point to estimate starting weights, and then increasing the training set one step at a time and re-estimating the weights for the remaining test set. However, this approach requires that the user provides a full test set, i.e. also providing observed values for the test set.

The function `auto_combine` provides a quick and painless alternative. The function is based on a grid-search optimization that returns the combined forecast with the best in-sample accuracy (using

RMSE as accuracy metric in the default setting).

All of the estimation methods return an object of S3 class `foreccomb_res` with the components presented in Table 2. The object can subsequently be passed on to post-fit functions.

Return Values	Description
For all methods	
Method	Returns the used forecast combination method
Models	Returns the individual input models that were used for the forecast combinations
Weights	Returns the combination weights obtained by applying the combination method to the training set
Fitted	Returns the fitted values of the combination method for the training set
Accuracy_Train	Returns a range of accuracy measures for the training set
Forecasts_Test	Returns forecasts produced by the combination method for the test set. Only returned if input included a forecast matrix for the test set
Accuracy_Test	Returs a range of accuracy measures for the test set. Only returned if input included a forecast matrix and a vector of actual values for the test set
Input_Data	Returns the data forwarded to the method
Only for `comb_TA` *and* `comb_WA`	
Trim Factor	Returns the trim factor, λ
Only for `comb_OLS`, `comb_LAD`, `comb_EIG3` *and* `comb_EIG4`	
Intercept	Returns the intercept (bias correction)
Only for `comb_EIG3` *and* `comb_EIG4`	
Top_Predictors	Number of retained predictors
Ranking	Ranking of predictors that determines which models are removed in the trimming step

Table 2: Output Components of the Forecast Combination Estimation Metods

Post-Fit Functions

Results from the estimation of combined forecasts can be passed on to two post-fit convenience functions: `summary` and `plot`, which are S3 methods specific to the class `foreccomb_res`.

The summary function displays the output of the respective forecast combination in concise form, it displays the estimated combination weights (and the intercept, if the combination method includes one), as well as accuracy statistics for the training set and the test set.

The plot function will produce different plots based on the input data. If only a training set was provided, it plots the actual versus the fitted values; if a test set was also provided, it plots the combined forecasts as well. Another option for the user is a plot of the combination weights[5], obtained by setting `which = 2` in the function call.

---

[5]In case the combined forecast is produced using time-varying combination weights, the weights plot displays only the average weight of the respective component model over the test set period.

UK Electricity Supply: An Empirical Example

The **ForecastComb** package includes the dataset `electricity`, which is a multivariate time series of monthly UK electricity supply (in GWh) from January 2007 to March 2017, and 5 univariate time series forecasts for the same series and period. The observed data series is sourced from the International Energy Agency (IEA, 2017). The component models to be combined are the following cross-validated one-month univariate forecasts in the dataset:

- ARIMA (produced using the `auto.arima` function in the **forecast** package),
- ETS (produced using the `ets` function in the **forecast** package),
- Neural Network (produced using the `nnetar` function in the **forecast** package),
- Damped Trend (produced using the `ets` function in the **forecast** package),
- Dynamic Optimized Theta Model (produced using the `dotm` function in the **forecTheta** package by Fiorucci et al. (2016)).

To illustrate the functionalities of the package, we apply 4 combination techniques: the simple average (`comb_SA`), the OLS regression (`comb_OLS`), the standard eigenvector approach (`comb_EIG1`) and the trimmed bias-corrected eigenvector approach (`comb_EIG4`). The selected methods span all three categories of combination techniques (statistics-based, regression-based, and eigenvector-based) and includes trimmed and bias-corrected methods and are therefore suitable to show the full functionality of the **ForecastComb** package in this empirical context. using both the static and dynamic version of each. For the selected combination methods, we produce both static and dynamic forecasts, which gives us a total of 7 different time series of combined forecasts (not 8, since the static and dynamic versions of the simple average combination are identical).

UK Electricity Supply, 2007 – 2017
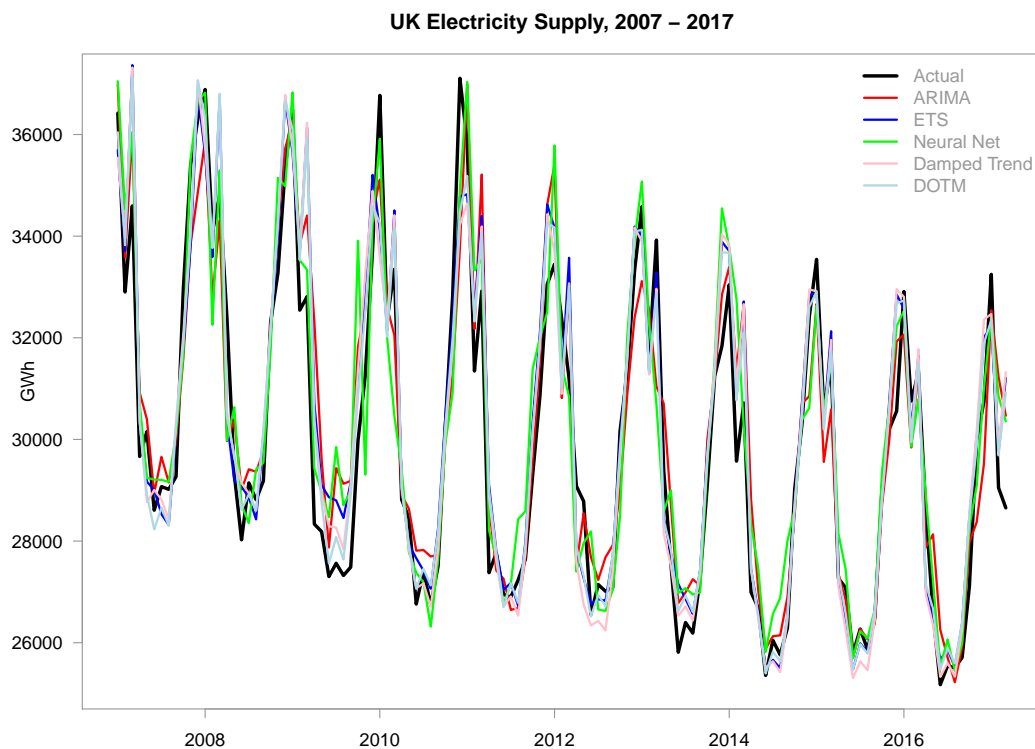
Figure 1: UK Electricity Supply, 2007 - 2017: Actual Value and Forecasts.

For the purpose of this exercise, we use the first 84 months as training set, which leaves us with a test set size of 39. Figure 1 plots the actual series and the univariate forecasts. The forecasts (which are 1-month forecasts obtained via time series cross-validation) track the actual series very well. None of them performs exceptionally poorly compared with the rest, which are conditions that tend to favour eigenvector approaches (Hsiao and Wan, 2014). As it is with most electricity data, the main difference between the individual models is in their ability to quickly recognize and adjust for turning points.

For example, the neural nets model handles turning points well, but sometimes also overshoots, while the ARIMA model has a smoother behaviour around turning points.

First, we format the data correctly for the estimation of combination weights. This step ensures later operations would proceed smoothly. Since there are no missing values and no perfectly collinear columns in our dataset, this is relatively straightforward:

```
R> data(electricity)
R> train.obs <- electricity[1:84, 6]
R> train.pred <- electricity[1:84, 1:5]
R> test.obs <- electricity[85:123, 6]
R> test.pred <- electricity[85:123, 1:5]
R> input_data <- foreccomb(train.obs, train.pred, test.obs, test.pred)
```

Once the object of S3 class foreccomb is created, it can be fed into the estimation functions. We can look at the cross-sectional dispersion, to get a better idea of variability in the univariate forecasts.

```
R> cs_dispersion(input_data, measure = "SD", plot = TRUE)
```
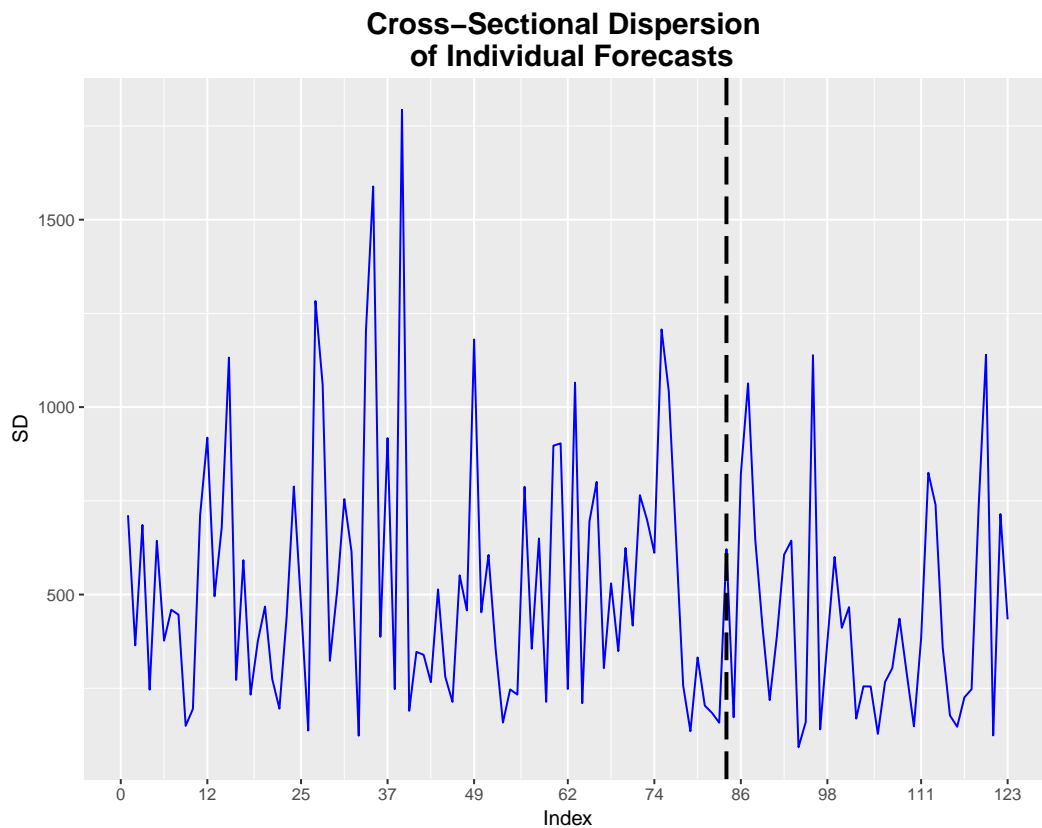

Figure 2: Cross-Sectional Standard Deviation of the Component Forecasts.

Figure 2 shows that apart from a brief period of increased dispersion around the end of 2009, the cross-sectional standard deviation of the component forecasts is rather stable and low given the level of around 25,000 to 35,000 GWh. This begs the question whether conditions have fluctuated enough during the test set so that estimation of time-varying weights is beneficial, yet we proceed with it for this demonstration.

```
R> ####### ESTIMATION OF STATIC FORECAST COMBINATIONS ########
```

```
R> SA <- comb_SA(input_data)
R> OLS_static <- comb_OLS(input_data)
R> EIG1_static <- comb_EIG1(input_data)
R> EIG4_static <- comb_EIG4(input_data, criterion = "MAE")

R> ###### ESTIMATION OF DYNAMIC FORECAST COMBINATIONS
R> OLS_dyn <- rolling_combine(input_data, "comb_OLS")
R> EIG1_dyn <- rolling_combine(input_data, "comb_EIG1")
R> EIG4_dyn <- rolling_combine(input_data, "comb_EIG4", criterion = "MAE")
```

The 7 combined forecasts can be evaluated separately by looking at their summary measures, which we present here for the static Ordinary Least Squares approach:

```
R> summary(OLS_static)

Summary of Forecast Combination
-------------------------------
Method:  Ordinary Least Squares Regression
Individual Forecasts & Combination Weights:
        Combination Weight
arima           0.02152869
ets            -0.20646266
nnet            0.20992792
dampedt        -1.04349858
dotm            1.97991049


Intercept (Bias-Correction):  962.3229


Accuracy of Combined Forecast:

                     ME      RMSE      MAE        MPE     MAPE
Training Set -4.417560e-12 888.1433 697.8645 -0.08262472 2.254470
Test Set     -4.007742e+01 671.5214 536.0331 -0.24705122 1.841961


Additional information can be extracted from the combination object:
For fitted values (training set):  OLS_static$Fitted
For forecasts (test set):  OLS_static$Forecasts_Test
See  str(OLS_static)  for full list.
```

The output shows that the OLS combination puts an extremely high relative weight on the forecast from the Dynamic Optimized Theta model, which seems to be the best component forecast, which is rather surprising given that seasonality is an important feature in the analyzed series and Theta models cannot incorporate seasonality into the estimation so far, relying on pre-estimation deseasonalizing and post-estimation reseasonalizing. Table 3 shows a comparison of the accuracies achieved by the combined forecasts. Since all forecasts are for the same series, it is reasonable to use MAE as accuracy metric.

Method	MAE Training Set	MAE Test Set
Simple Average	819.28	573.39
Ordinary Least Squares (static)	697.86	536.03
Ordinary Least Squares (dynamic)	697.86	533.47
Standard Eigenvector (static)	821.60	573.84
Standard Eigenvector (dynamic)	821.60	572.99
Trimmed Bias-Corrected Eigenvector (static)	785.30	540.18
Trimmed Bias-Corrected Eigenvector (dynamic)	785.30	541.98

Table 3: Mean Absolute Errors of Combined Forecasts

The evaluation of accuracy delivers some interesting insight: All of the combination models perform better in the test set than in the training set, which is counter-intuitive, but is likely due to the increased dispersion among the component forecasts in the early period. The results also clearly suggest that one or more of the component forecasts were biased. The two methods with an intercept

(i.e. bias correction) perform best. Finally, allowing for time-varying combination weights does not seem to change test-set accuracy much compared with the models' static counterparts, suggesting that the estimated combination weights did not fluctuate a lot over time. There are some potential explanations why the OLS method performed extremely well in this case:

- With such stable conditions the risk of 'bouncing betas' (described in the previous section) is low,

- The OLS method produces unbiased forecasts even if one or more of the component forecasts are biased (which is why the trimmed bias-corrected eigenvector approach performed reasonably well too),

- One of the component forecasts is much better than the rest, a situation that is favorable for regression-based approaches, as pointed out by Hsiao and Wan (2014). These are also conditions under which sophisticated methods actually can largely improve upon a simple average combination.

Now that we learned that some of the combination models produced more accurate forecasts than the simple average, we address the next, very natural question: How well did the combination methods perform compared to the univariate component forecasts themselves? To shed more light on this question, Table 4 shows the MAE values for the univariate models during the test set period.

Method	MAE Test Set
ARIMA	770.32
ETS	615.88
Neural Network	730.35
Damped Trend	660.08
DOTM	540.24

Table 4: Mean Absolute Errors of Univariate Component Forecasts

The results of the accuracy evaluation speak for themselves. Not only did two of the combined forecasts perform better or equally well as the best univariate forecast over the test set period, but also the forecast risk is considerably lower indeed. In the test set the range of MAEs of the different combined forecast methods is only 40, while the corresponding value for the univariate forecasting methods is 230. It is worth noting that all of the combined forecasts perform considerably better than even the second-best univariate forecast, emphasizing the appeal of forecast combination: In the ideal case, it is possible to end up with a forecast that is better than the best univariate forecast. Even if this is not the case, using forecast combination considerably decreases the risk of ending up with a poorly performing model.

Finally, we take a closer look at the results from the best combined forecast, which is the dynamic OLS combination method in this exercise.

```
R> ##### ACTUAL VS FITTED PLOT #####
R> plot(OLS_dyn)



R> ##### COMBINATION WEIGHTS #####
R> OLS_static$Weights
 fhatarima     fhatets    fhatnnet fhatdampedt     fhatdotm
 0.02152869 -0.20646266  0.20992792 -1.04349858  1.97991049
R> colMeans(OLS_dyn$Weights)


   fhatarima       fhatets     fhatnnet fhatdampedt      fhatdotm
 0.002750948 -0.123811218  0.184846358 -1.098381593  2.001988985
```

Figures 3 shows how well the combined forecast obtained from the dynamic OLS method predicts the monthly electricity supply series. Results confirm the conjecture that the stable conditions (low cross-sectional dispersion and one very dominant univariate component forecast) do not cause a lot of fluctuation in the combination weights even when allowing for time-varying weights.

The weights presented also confirms another thing: that weights obtained using the OLS combination methods can be hard to interpret. It seems obvious that the method should put a high weight on the DOTM forecast, since it is the best univariate forecast by far. However, the reason why it assigns
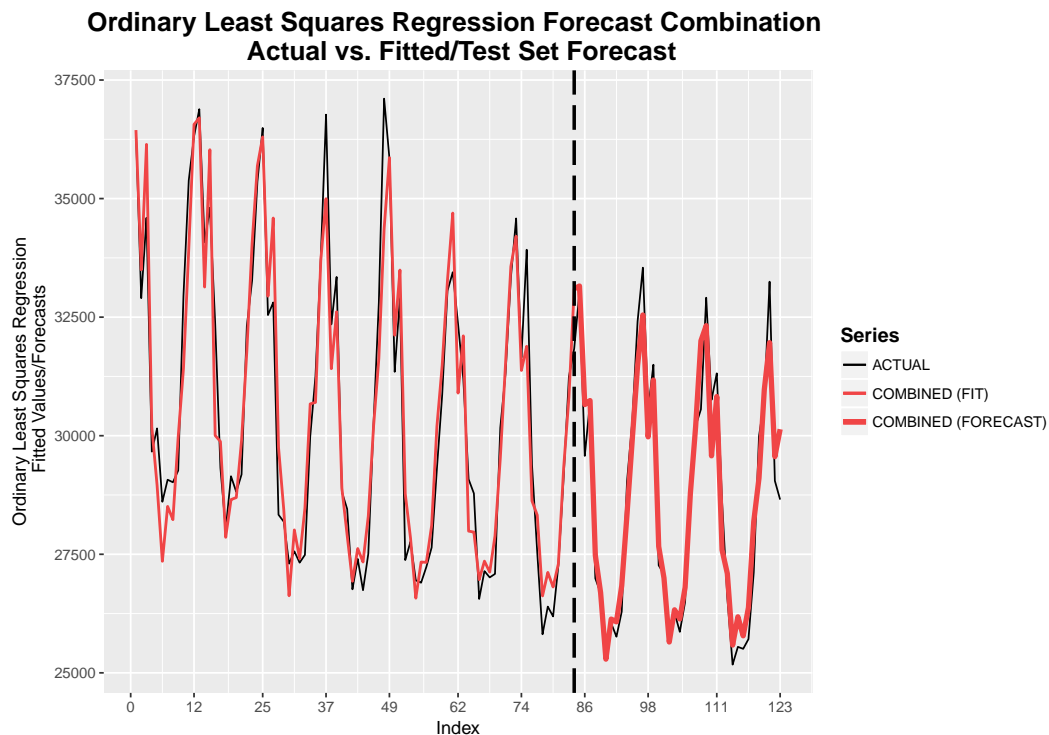
Figure 3: Static OLS Forecast Combination: Actual vs Fitted Plot.

negative weights to the ETS and Damped Trend forecasts (the second- and third-best univariate forecasts) is not very intuitive. A possible explanation might be that all three of these are exponential smoothing-type models, suggesting that the information obtained from the ETS and Damped Trend models is better captured by the DOTM model, while ARIMA and Neural Networks are not closely related modelling approaches to the Theta model, so that even though these models perform far worse on average, they capture information differently and might outperform the DOTM forecast in some periods for that reason, justifying their positive weights (however small) and explaining how the combined forecast can slightly outperform the best component forecast.

Discussion and Conclusions

Forecast combination is a useful strategy to hedge against model risk. Even if combined forecasts do not win over the most accurate component forecast, they generally avoid poor performance by circumventing the choice between individual methods (Timmermann, 2006). Instead of putting all eggs into one basket using model selection, these model averaging techniques are motivated by portfolio theory and diversify across component forecasts.

The **ForecastComb** package categorizes some of the most popular approaches into 3 groups: (a) simple statistics-based methods, (b) regression-based methods, and (c) eigenvector-based methods. Providing both regression-based combinations and eigenvector-based combinations to the users is considered useful, since the former tend to perform relatively better when one or a few component forecasts are much better than the rest, while the latter perform relatively better when all forecasts are in the same ballpark (Hsiao and Wan, 2014).

The package is designed to support users along the entire modelling process: data preparation, model estimation, and interpretation of results using summaries and plotting functionalities. It includes tools for data transformation that are designed to deal with two common issues in forecast combination prior to estimation – missing values and multicollinearity. The 15 combination methods are available in both static and dynamic variants, and users have the option to automate the selection algorithm so that a good combination method is found based on training set fit. Finally, the package offers specialized functions for summarizing and visualizing the combination results.

While the current version of the package already provides a comprehensive set of tools for forecast combination, there is scope for further extensions in future updates. First, additional combination methods that showed promising results recently can be added, for instance the factor-augmented regression approach by Cheng and Hansen (2015) and the AdaBoost algorithms reviewed by Barrow and Crone (2016). Second, additional algorithms for adaptive combination weights (cf. Timmermann, 2006) can be implemented to provide even more flexibility with dynamic estimation. There are few other possible extensions which are not handled here. For example the adaptation for a forecast combination context of the mean absolute scaled error (Hyndman and Koehler, 2006) – the current gold standard for accuracy evaluation – using the in-sample MAE of the best component forecast as scaling factor. Another extension is of course the combination of forecast densities, while in this paper the scope was limited to point-forecasts only.

Acknowledgements

Bibliography

M. Aiolfi and A. Timmermann. Persistence in forecasting performance and conditional combination strategies. *Journal of Econometrics*, 135(1):31–53, 2006. URL https://doi.org/10.1016/j.jeconom.2005.07.015. [p4, 7]

R. R. Andrawis, A. F. Atiya, and H. El-Shishiny. Combination of long term and short term forecasts, with application to tourism demand forecasting. *International Journal of Forecasting*, 27(3):870–886, 2011. URL https://doi.org/10.1016/j.ijforecast.2010.05.019. [p2]

J. S. Armstrong. Combining forecasts: The end of the beginning or the beginning of the end? *International Journal of Forecasting*, 5(4):585–588, 1989. URL https://doi.org/10.1016/0024-6301(90)90317-w. [p3]

J. S. Armstrong. Combining forecasts. In J. S. Armstrong, editor, *Principles of Forecasting: A Handbook for Researchers and Practitioners*, pages 417–439. Kluwer Academic Publisher, 2001. URL https://doi.org/10.1016/s0040-1625(02)00180-4. [p3]

D. Avramov. Stock return predictability and model uncertainty. *Journal of Financial Economics*, 64(3): 423–458, 2002. URL https://doi.org/10.2139/ssrn.260591. [p2]

S. Babaie-Kafaki and M. Roozbeh. A revised cholesky decomposition to combat multicollinearity in multiple regression models. *Journal of Statistical Computation and Simulation*, 87(12):2298–2308, 2017. URL https://doi.org/10.1080/00949655.2017.1328599. [p5]

D. K. Barrow and S. F. Crone. A comparison of adaboost algorithms for time series forecast combination. *International Journal of Forecasting*, 32(4):1103–1119, 2016. URL https://doi.org/10.1016/j.ijforecast.2016.01.006. [p17]

J. M. Bates and C. W. J. Granger. The combination of forecasts. *Operations Research Quarterly*, 20: 451–468, 1969. URL https://doi.org/10.1016/b978-0-12-295180-0.50027-6. [p2, 4]

C. Bergmeir, R. J. Hyndman, and B. Koo. A note on the validity of cross-validation for evaluating time series prediction. Working Paper No. 10/15, Monash University, Department of Econometrics and Business Statistics, 2015. URL https://doi.org/10.1016/j.csda.2017.11.003. [p10]

G. Cheng and Y. Yang. Forecast combination with outlier protection. *International Journal of Forecasting*, 31(2):223–237, 2015. URL https://doi.org/10.1016/j.ijforecast.2014.06.004. [p2]

X. Cheng and B. E. Hansen. Forecasting with factor-augmented regression: A frequentist model averaging approach. *Journal of Econometrics*, 186(2):280–293, 2015. URL https://doi.org/10.2139/ssrn.2343103. [p17]

C. Christiansen, M. Schmeling, and A. Schrimpf. A comprehensive look at financial volatility prediction by economic variables. *Journal of Applied Econometrics*, 27(6):956–977, 2012. URL https://doi.org/10.1002/jae.2298. [p2]

G. Claeskens, J. R. Magnus, A. L. Vasnev, and W. Wang. The forecast combination puzzle: A simple theoretical explanation. *International Journal of Forecasting*, 32(3):754–762, 2016. URL https://doi.org/10.1016/j.ijforecast.2015.12.005. [p3]

R. T. Clemen. Combining forecasts: A review and annotated bibliography. *International Journal of Forecasting*, 5(4):559–583, 1989. URL https://doi.org/10.1016/0169-2070(89)90012-5. [p2, 3]

D. B. Crane and J. R. Crotty. A two-stage forecasting model: Exponential smoothing and multiple regression. *Management Science*, 13(8):501–507, 1967. URL https://doi.org/10.1287/mnsc.13.8.b501. [p2, 4]

G. Elliott and A. Timmermann. Optimal forecast combinations under general loss functions and forecast error distributions. *Journal of Econometrics*, 122(1):47–79, 2004. URL https://doi.org/10.1016/j.jeconom.2003.10.019. [p5]

G. Elliott and A. Timmermann. Optimal forecast combinations under regime switching. *International Economic Review*, 46(4):1081–1102, 2005. URL https://doi.org/10.1111/j.1468-2354.2005.00361.x. [p2]

G. Elliott, A. Gargano, and A. Timmermann. Complete subset regressions. *Journal of Econometrics*, 2013. URL https://doi.org/10.1016/j.jeconom.2013.04.017. [p2, 5]

J. A. Fiorucci, F. Louzada, and B. Yiqi. *forecTheta: Forecasting Time Series by Theta Models*, 2016. URL https://CRAN.R-project.org/package=forecTheta. R package version 2.2. [p12]

P. Gaillard and Y. Goude. *Opera: Online Prediction by Expert Aggregation*, 2016. URL https://CRAN.R-project.org/package=opera. R package version 1.0. [p2]

V. Genre, G. Kenny, A. Meyler, and A. Timmermann. Combining expert forecasts: Can anything beat the simple average? *International Journal of Forecasting*, 29(1):108–121, 2013. URL https://doi.org/10.1016/j.ijforecast.2012.06.004. [p3]

A. Graefe, J. S. Armstrong, R. J. Jones, and A. G. Cuzán. Combining forecasts: An application to elections. *International Journal of Forecasting*, 30(1):43–55, 2014. URL https://doi.org/10.1016/j.ijforecast.2013.02.005. [p1, 3]

C. W. Granger and R. Ramanathan. Improved methods of combining forecasts. *Journal of Forecasting*, 3(2):197–204, 1984. URL https://doi.org/10.1002/for.3980030207. [p4]

E. J. Hannan and B. G. Quinn. The determination of the order of an autoregression. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 190–195, 1979. URL http://www.jstor.org/stable/2985032. [p6]

B. E. Hansen. Challenges for econometric model selection. *Econometric Theory*, 21(1):60–68, 2005. URL https://doi.org/10.1017/s0266466605050048. [p1]

B. E. Hansen. Least squares model averaging. *Econometrica*, pages 1175–1189, 2007. URL https://doi.org/10.1111/j.1468-0262.2007.00785.x. [p2]

B. E. Hansen. Least-squares forecast averaging. *Journal of Econometrics*, 146(2):342–350, 2008. URL https://doi.org/10.1016/j.jeconom.2008.08.022. [p2]

B. E. Hansen and J. S. Racine. Jackknife model averaging. *Journal of Econometrics*, 167(1):38–46, 2012. URL https://doi.org/10.1016/j.jeconom.2011.06.019. [p2]

D. F. Hendry and M. P. Clements. Pooling of forecasts. *The Econometrics Journal*, 7(1):1–31, 2004. URL ttps://doi.org/10.1111/j.1368-423x.2004.00119.x. [p3]

C. Hsiao and S. K. Wan. Is there an optimal forecast combination? *Journal of Econometrics*, 178(2):294–309, 2014. URL https://doi.org/10.1016/j.jeconom.2013.11.003. [p2, 4, 6, 7, 9, 12, 15, 16]

C. M. Hurvich and C.-L. Tsai. Regression and time series model selection in small samples. *Biometrika*, 76(2):297–307, 1989. URL https://doi.org/10.2307/2336663. [p6]

R. J. Hyndman. *forecast: Forecasting Functions for Time Series and Linear Models*, 2017. URL http://github.com/robjhyndman/forecast. R package version 8.1. [p9]

R. J. Hyndman and A. B. Koehler. Another look at measures of forecast accuracy. *International Journal of Forecasting*, 22(4):679–688, 2006. URL https://doi.org/10.1016/j.ijforecast.2006.03.001. [p17]

IEA. *Monthly Electricity Statistics*, 2017. International Energy Agency. [p12]

V. R. R. Jose and R. L. Winkler. Simple robust averages of forecasts: Some empirical results. *International Journal of Forecasting*, 24(1):163–169, 2008. URL https://doi.org/10.1016/j.ijforecast.2007.06.001. [p3, 4]

W. Junger and A. P. de Leon. *Mtsdi: Multivariate Time Series Data Imputation*, 2012. URL https://CRAN.R-project.org/package=mtsdi. R package version 0.3.3. [p9]

G. Kapetanios, V. Labhard, and S. Price. Forecasting using bayesian and information-theoretic model averaging: An application to uk inflation. *Journal of Business & Economic Statistics*, 26(1):33–41, 2008. URL https://doi.org/10.1198/073500107000000232. [p2]

J. R. Magnus and W. Wang. Concept-based bayesian model averaging and growth empirics. *Oxford Bulletin of Economics and Statistics*, 76(6):874–897, 2014. URL https://doi.org/10.1111/obes.12068. [p2]

S. K. McNees. The uses and abuses of 'consensus' forecasts. *Journal of Forecasting*, 11(8):703–710, 1992. URL https://doi.org/10.1016/0169-2070(93)90039-p. [p3]

C. Morana. Model averaging by stacking. Working Paper 310, University of Milan Bicocca, Department of Economics, Management and Statistics, 2015. URL https://doi.org/10.2139/ssrn.2665704. [p2]

P. Newbold and C. W. J. Granger. Experience with forecasting univariate time series and the combination of forecasts. *Journal of the Royal Statistical Society A*, 137(2):131–165, 1974. URL https://doi.org/10.2307/2344546. [p4, 6]

J. Nowotarski, E. Raviv, S. Trück, and R. Weron. An empirical comparison of alternative schemes for combining electricity spot price forecasts. *Energy Economics*, 46:395–412, 2014. URL https://doi.org/10.2139/ssrn.2313553. [p2, 5]

A. Opschoor, D. J. Van Dijk, and M. Van der Wel. Improving density forecasts and value-at-risk estimates by combining densities. Discussion Paper 14-090/III, Tinbergen Institute, 2014. URL https://doi.org/10.2139/ssrn.2469024. [p2]

F. C. Palm and A. Zellner. To combine or not to combine? issues of combining forecasts. *Journal of Forecasting*, 11(8):687–701, 1992. URL https://doi.org/10.1002/for.3980110806. [p3]

A. E. Raftery and I. S. Painter. Bma: An r package for bayesian model averaging. *R news*, 5(2):2–8, 2005. [p2, 6]

D. E. Rapach, J. K. Strauss, and G. Zhou. Out-of-sample equity premium prediction: Combination forecasts and links to the real economy. *Review of Financial Studies*, 23(2):821–862, 2010. [p2]

F. Ravazzolo, M. Verbeek, and H. K. Van Dijk. Predictive gains from forecast combinations using time varying model weights. Report 2007-26, Econometric Institute, 2007. URL https://doi.org/10.2139/ssrn.1012574. [p2]

E. Raviv. *ForecastCombinations: Forecast Combinations in R*, 2015. URL https://CRAN.R-project.org/package=ForecastCombinations. R package version 1.1. [p2]

E. Raviv, K. E. Bouwman, and D. van Dijk. Forecasting day-ahead electricity prices: Utilizing hourly prices. *Energy Economics*, 50:227 – 239, 2015. ISSN 0140-9883. URL https://doi.org/10.1016/j.eneco.2015.05.014. [p2]

D. Shaub and P. Ellis. *forecastHybrid: Convenient Functions for Ensemble Time Series Forecasts*, 2017. URL https://CRAN.R-project.org/package=forecastHybrid. R package version 0.4.1. [p2, 9]

J. Smith and K. F. Wallis. A simple explanation of the forecast combination puzzle. *Oxford Bulletin of Economics and Statistics*, 71(3):331–355, 2009. URL https://doi.org/10.1111/j.1468-0084.2008.00541.x. [p3]

J. H. Stock and M. W. Watson. Combination forecasts of output growth in a seven-country data set. *Journal of Forecasting*, 23(6):405–430, 2004. URL https://doi.org/10.1002/for.928. [p2, 3]

A. Timmermann. Forecast combinations. *Handbook of economic forecasting*, 1:135–196, 2006. URL https://doi.org/10.1016/s1574-0706(05)01004-9. [p2, 3, 4, 6, 10, 16, 17]

B. A. Turlach and A. Weingessel. *Quadprog: Functions to Solve Quadratic Programming Problems.*, 2013. URL https://CRAN.R-project.org/package=quadprog. R package version 1.5-5. [p5]

C. E. Weiss. Hierarchical time series and forecast combination in healthcare demand forecasting. Working Paper, Judge Business School, University of Cambridge, 2017. [p2]

C. E. Weiss and G. R. Roetzer. *GeomComb: (Geometric) Forecast Combination Methods*, 2016. URL https://CRAN.R-project.org/package=GeomComb. R package version 1.0. [p2]

J. H. Wright. Bayesian model averaging and exchange rate forecasts. *Journal of Econometrics*, 146(2): 329–341, 2008. URL https://doi.org/10.2139/ssrn.457345. [p2]

J. H. Wright. Forecasting us inflation by bayesian model averaging. *Journal of Forecasting*, 28(2):131–144, 2009. URL https://doi.org/10.1002/for.1088. [p2]

Christoph E. Weiss
Judge Business School
University of Cambridge
Cambridge CB2 1AG
United Kingdom
christoph.weiss@cantab.net

Eran Raviv
APG - Asset Management
Gustav Mahlerplein 3, 1082 Amsterdam
The Netherlands
eran.raviv@apg-am.nl

Gernot Roetzer
Trinity College Dublin
College Green, Dublin 2
Ireland
roetzeg@tcd.ie