

Forecast combinations in R using the *ForecastCombinations* package A Manual

Eran Raviv
([link to website](#))

ABSTRACT. Model uncertainties and/or model instabilities are deeply-rooted in real-life forecasting challenges. More often than not and particularly in soft sciences, the underlying data-generating process is simply beyond us. When it comes to forecasting, this makes model-selection uncomfortably restrictive in dynamic environments. An obvious alternative to choosing single best forecasting method is to combine forecasts from different models. By now it is well established that combining different forecasts delivers results which are “better than the best”. There are numerous ways in which forecasts can be combined, with little theoretical underpinnings. This paper introduces the R package `ForecastCombinations`. The aim is to describe the arguably most common ways in which forecasts can be combined, and to demonstrate the implementation thereof. Among other combination schemes available, package `ForecastCombinations` allows for variance-based, constrained least-squares, information-criteria based or the recently advocated complete-subset-regression.

1. Introduction

In the business of forecasting, precision is key. The advent of “black-box” statistical learning methods such as *Artificial Neural Networks* testifies to the fact that one often prefers a more accurate forecast even over an explainable one. Especially in soft sciences, the admission that we simply don’t know what is the underlying data-generating-process (DGP) is making headway. Indeed, some would argue that such, one real underlying DGP, does not even exist. Our models, almost by definition, are simple and incomplete. In the field of Econometrics, [Hansen \(2005\)](#) soberly phrase this: “models should be viewed as approximations, and econometric theory should take this seriously”. If we now abandon the quest for a one *true* correctly specified model, then the door is open for us to use many models. In the context of forecasting, this means combining forecasts from different models. By now there is ample empirical evidence on the merits of combining forecasts. Put bluntly, a combined forecast is more accurate than any of it’s components. The reason for the improvement brought about by combining forecasts is not theoretically grounded. One explanation is that we simply “hedge” against model risk. In a constantly changing environment different forecasting models deliver different results at different time periods, and for different choices of data inputs. By combining different forecasts we reduce the risk of choosing a forecasting model which is fairly accurate when

evaluated using some validation sample, yet proves unreliable when applied using new data. Forecasts can be biased, partly or all through the sample and those biases are less likely to apply in full when they are combined.

The idea of “hedging” against model risk by means of combining different forecasts is intuitive and appealing. Mingled with accuracy gains, it 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. \(2009\)](#) and [Ravazzolo et al. \(2007\)](#) for forecasting stock returns, [Wright \(2008\)](#) for exchange rate forecasts, [Andrawis et al. \(2011\)](#) for forecasting inbound tourism figures, and [Nowotarski et al. \(2014\)](#) and [Raviv et al. \(2015\)](#) for electricity price 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 literature is already five decades old, going back to the seminal papers of [Crane and Crotty \(1967\)](#) and [Bates and Granger \(1969\)](#). Yet even in 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 \(2006\)](#).

With this motivation in mind, it is somewhat surprising to see the very few packages that are devoted to this subject in R ¹. The `ForecastCombinations` package aims to fill some of this void. The paper presents the functionalities made available and demonstrates how to implement them.

The `ForecastCombinations` package was created in version 3.1.0 of R , and depends on the following three packages: `quantreg`, `quadprog` and `utils`. It is available from the Comprehensive R Archive Network (CRAN) at <https://cran.r-project.org/web/packages/ForecastCombinations/index.html>.

The remainder of this paper is structured as follows. Section 2 describes the possible ways in which forecasts can be combined, and are implemented in the `ForecastCombinations` package. Section 3 provides a demonstration, an R implementation of each of those methods presented in the Section 2. We use the Boston MASS dataset as empirical example (the data is available from the `MASS` package). The article is concluded with a discussion in Section 4.

¹Notable exceptions are [Raftery and Painter \(2005\)](#) who created the `BMA` package (Bayesian Model Averaging), and the `hts` package written by [Hyndman et al. \(2015\)](#), which allows optimal combination weights for forecasts of hierarchical time series.

2. Forecast combinations

To fix notations, denote $F_{N \times P}$ matrix of forecast with dimension $N \times P$ where N is the number of rows and P is the number of columns (so we should have P forecasts at each point in time). Denote f_i as the forecast obtained using model i , $i \in \{1 \dots P\}$. Denote the weight given to that forecast in the overall combined forecast as w_i , and the combined forecast as f^c .

2.1. Frequently used schemes for forecast combinations.

- (A) *Simple average*. The most natural approach to combine forecasts is using the mean of all those forecasts. Over the years this innocent approach have been proven as an excellent benchmark, despite or perhaps because of its simplicity (Genre et al. 2013, for example). The combined forecast is given by

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

Of course other location measures, perhaps less sensitive to outliers such as the median could also be used.

- (B) *Ordinary Least Squares (OLS) regression*. An idea put up by Crane and Crotty (1967) and was successfully driven to the forefront by Granger and Ramanathan (1984). Using this approach, the combined forecast is not a function of P only as in the simple average, but is a linear function of the individual forecasts whereby the parameters are determined using a regression of the individual forecasts on the target itself:

$$y = \alpha + \sum_{i=1}^P \beta_i f_i + \varepsilon, \quad (2)$$

Note that we need to allocate a portion from of forecasts to be used as a learning period, a reserved portion of forecasts is needed for all subsequently presented combination schemes. Once we obtain our estimates, parameters are estimated, the combined forecast is then given by

$$f^c = \hat{\alpha} + \sum_{i=1}^P \hat{\beta}_i f_i, \quad (3)$$

An advantages of the OLS forecast combinations is that the combined forecast is unbiased, even if one of the individual forecasts is. A disadvantage is that the resulting weights can be hard to interpret. Especially if the coefficients are non-convex.²

²if the combination of two individual forecasts is not convex, the resulting combined forecast will 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>).

- (C) *Least Absolute Deviation (LAD) regression*. The coefficients in equation (3) are estimated by way of minimizing the sum of squared errors in equation ((B)). We may want to estimate those coefficients differently, minimizing different loss function, for example the absolute sum of squares. The reason is best explained using an example. Say 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, how costly is it to miss the target given the problem at hand.
- (D) *Constrained Least Squares (CLS) regression*. Naturally, since all forecasts aim at the same target, they will be correlated, sometimes very much so. High correlation can lead to weights which are unstable if equation (3) is estimated without constraints, a problem sometimes dubbed as "bouncing betas". CLS addresses this issue by minimizing the sum of squared errors under some additional constraints. Specifically, we constrain the estimated coefficients $\{\beta_i\}$, allowing only for positive solutions: $\beta_i \geq 0 \forall i$, and to sum up to one: $\sum_{i=1}^P \beta_i = 1$. The solution requires numerical minimization, but good optimization algorithms are at the ready. The package relies on the function `solve.QP` available from the `quadprog` package (Turlach and Weingesse 2013).

Theoretically, the additional constraints set CLS sub-optimally compared to the OLS. It lacks the good asymptotic properties admitted by the 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 swiftly interpreted for example as percentages devoted to each of the individual forecasts.

- (E) *Variance-based, or Inverse Mean Squared Error (Inverse MSE)*. An approach successfully applied in Stock and Watson (2004) for combining forecasts of output growth. Put simply, more accurate forecasting methods (lower MSE) are weighted more heavily. MSE here is computed based on out-of-sample forecasts, sometimes referred to as Mean Squared Prediction Error. The combined forecast is given by

$$f^c = \frac{\left(\frac{MSE_i}{\sum_{i=1}^P MSE_i}\right)^{-1}}{\sum_{i=1}^P \left(\frac{MSE_i}{\sum_{i=1}^P MSE_i}\right)^{-1}} f_i = \frac{\frac{1}{MSE_i}}{\sum_{i=1}^P \frac{1}{MSE_i}} f_i. \quad (4)$$

- (F) *Best Individual model (BI)*. Essentially, selecting one model out of the available pool based on some accuracy measure can be viewed as a specific case of forecast combinations, with strong constraints. To see this, simply write the combined

forecast as

$$f^c = w_i f_i, \quad \text{where} \quad \begin{cases} w_i = 1 & \text{if } MSE_i < MSE_{-i} \quad \forall i \in \{1, \dots, P\} \\ w_i = 0 & \text{otherwise} \end{cases} \quad (5)$$

As mentioned at the outset, this is a very strong form of regularization and so is somewhat restrictive. However, it is easy to calculate and it is easy to explain. Apart from those desirable properties, if the environment in which we are in is of a static nature, one individual forecasting method may be identified as decidedly superior. In such a situation restrictions as in equation (5) may be in order.

2.2. Complete subset regressions forecast combinations. The `ForecastCombinations` package allows the relatively new idea of complete subset regressions forecast combinations. In particular, for a fixed P we compute

$$\Psi = \sum_{i=1}^P \binom{P}{i} = \sum_{p=1}^P \frac{P!}{i!(P-i)!} \quad (6)$$

regressions and obtain Ψ forecasts. We use the mean of those Ψ forecasts as the final combined forecast. Referring to [Elliott et al. \(2013\)](#), the authors 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 propose a workaround based on random sampling from P). Additionally, since all $\psi = \{1, \dots, \Psi\}$ forecasts are returned, the user can freely refine the technique further. For example choosing to average not based on all Ψ forecasts but based on some partial subset of Ψ . For example the most accurate $g\%$ forecasts say, treating g as a hyper parameter. Using the median instead of the mean is another option that comes to mind.

2.3. Information-theoretic forecast combinations. Armed with Ψ , the package `ForecastCombinations` supports the frequentists approach to forecast combinations, also known as information-theoretic forecast combinations. Various information criteria are available, including the recently advocated Mallows's information criteria, or 'Mallows model Averaging' (MMA) as in the original paper of [Hansen \(2007\)](#).

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

$$w_j = \frac{\exp(-1/2\varrho_j)}{\sum_{j=1}^{\Psi} \exp(-1/2\varrho_j)}, \quad (7)$$

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

$$f^c = \sum_{j=1}^{\Psi} w_j \tilde{f}_j. \quad (8)$$

We denote \tilde{f} here so as to distinguish it from f , the original forecast.

There are numerous information criteria to choose from, each with its own merits and weaknesses. By far the most common are the AIC (Akaike's information criteria) and the BIC (Bayesian information criteria, 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 criteria (Hannan and Quinn 1979).

The Mallows information criteria is most directly given by

$$M_\psi = \frac{RSS + 2p(\psi)\hat{\sigma}_{p(\psi)}^2}{N}, \quad (9)$$

where RSS is the residual sum of squared errors based on some 'target' model, a model which is considered as an alternative. In our context it is the residual sum of squares based on linear regression which includes all forecasts. The effective number of parameters in a specific combo is given by $p(\psi)$, and $\hat{\sigma}^2$ is an estimate of the error variance in that combo, finally N is the number of observations. The combined forecast is given by

$$f^c = \sum_{j=1}^{\Psi} w_j \tilde{f}_j = \sum_{j=1}^{\Psi} \frac{M_j^{-1}}{\sum_{j=1}^{\Psi} M_j^{-1}} \tilde{f}_j \quad (10)$$

One advantage of this approach as well as of the complete subset regressions approach is that the amount of shrinkage enforced on the each individual forecast is data driven. The specification of shrinkage hyper-parameter, which is required in the corresponding Bayesian framework (Raftery and Painter 2005, for example) is spared from the user in this case.

3. Using the ForecastCombinations package

This section demonstrates the usage and usefulness of the `ForecastCombinations` package. We use the `Boston` dataset available as part of `MASS` package. The data contains information collected by the U.S Census Service concerning housing in the area of Boston Mass. This dataset is frequently used as a testing ground for evaluating various new algorithms. Short variable description is given in table 1.

3.1. Data preparation. We start by splitting the data into a training set and a test set. The training set will be used to train the individual models. The test set is again split into two parts, the first part out of the two is used to train the combination scheme, save for the simple average which does not need any training as the weights are a function of P only. The second portion of the test is reserved for the evaluation of the different combination schemes. The goal is to forecast with accuracy the variable *Median value of owner-occupied home* (column 14 in the data) based on all other explanatory variables.

Load the package (I assume you have it installed).

```
> library(ForecastCombinations)
```

Variable	Description
CRIM	per capita crime rate by town
ZN	proportion of residential land zoned for lots over 25,000 square feet
INDUS	proportion of non-retail business acres per town
CHAS	Charles River dummy variable (1 if tract bounds river; 0 otherwise)
NOX	nitric oxides concentration (parts per 10 million)
RM	average number of rooms per dwelling
AGE	proportion of owner-occupied units built prior to 1940
DIS	weighted distances to five Boston employment centres
RAD	index of accessibility to radial highways
TAX	full-value property-tax rate per \$10,000
PTRATIO	pupil-teacher ratio by town
BLACK	$1000(Bk - 0.63)^2$ where Bk is the proportion of blacks by town
LSTAT	% lower status of the population
MEDV	<i>Dependent variable:</i> Median value of owner-occupied homes in \$1000's

TABLE 1. Description of the Boston dataset

```

> library(MASS)
> nam <- names(Boston)
> set.seed(111)
> ind1 <- sample(NROW(Boston), size = round(NROW(Boston)*(1/2)))
> dtrain <- Boston[ind1, ]
> dtest_total <- Boston[-c(ind1), ]
> ind2 <- sample(NROW(dtest_total), size = round(NROW(dtest_total) * (1/2)))
> dtest_part1 <- dtest_total[ind2, ]
> dtest_part2 <- dtest_total[-ind2, ]
> ttest <- NROW(dtest_total)

```

3.2. Creating forecasts. The point of departure is that the forecaster already has her set of forecasts. Therefore, initially we create a set of forecasts to be combined later. Keeping our focus, where additional inputs are needed in creating the individual forecasts we dispense from any tweaking and/or model refinements and use the software default values or some ad-hoc choices. We don't elaborate here on the actual methods, the interested reader can inquire further using the references provided. The forecasts are based on the following forecasting methods:

- (1) Linear model (OLS)
- (2) Principal component regression (Mevik and Wehrens 2007)
- (3) Boosting regression trees (Ridgeway 2007)
- (4) Random forests (Liaw and Wiener 2002)
- (5) Support vector machine (Hornik et al. 2006)

(6) Neural Network (Venables and Ripley 2002)

A total of six forecasts.

```

> # -- Ordinary Least Squared -----
> lmtrain <- lm(medv~., data = dtrain)
> lm_p <- t(t(lmtrain$coef) %*% t(cbind(rep(1, ttest), dtest_total[, -14])))
> # -- Principal Component Regression-----
> pctr <- prcomp(dtrain[, -14])
> # We use 3 principal components
> num_factors <- 3
> pctrtrain <- lm(dtrain[, "medv"] ~ pctr$x[, 1:num_factors])
> pc_score_test <- t(t(pctr$rot[, 1:num_factors]) %*% t(dtest_total[, -14]))
> pcr_p <- t(t(pctrtrain$coef) %*% t(cbind(rep(1, ttest), pc_score_test)))
> # -- Boosting Regression Trees -----
> library(dismo)
> model_gbm <- gbm.step(data=dtrain, gbm.x = names(dtrain)[-14], gbm.y = "medv",
+ family = "gaussian", plot.main = F, silent = T)
> boost_p <- predict(model_gbm, newdata=dtest_total[, -14], n.trees=500)
> # -- Support Vector Machine -----
> library(e1071)
> svm_mod <- svm(medv~., data = dtrain, scale = T)
> svm_p <- predict(svm_mod, dtest_total[, nam!="medv"])
> # -- Random Forests-----
> library(randomForest)
> model_rf <- randomForest(medv~., data= dtrain)
> rf_p <- predict(model_rf, newdata= dtest_total[, -14], type= "response")
> # -- Neural Networks -----
> library("neuralnet")
> temp_f <- as.formula(paste("medv ~", paste(nam[!nam %in% "medv"],
+                                     collapse = " + ")))
> nn <- neuralnet(temp_f , data=scale(dtrain), linear.output=T)
> library(nnet)
> nn_p <- compute(nn, scale(dtest_total[,1:13]))
> # Scale back to the original values:
> nn_p <- nn_p$net.result*sd(dtrain$medv) + mean(dtrain$medv)
> # ----- Now we name and combine the forecast into a matrix:
> indiv_forecasts <- cbind(lm_p, pcr_p, boost_p, svm_p, rf_p, nn_p)
> ind_nam <- c("lm_p", "pcr_p", "boost_p", "svm_p", "rf_p", "nn_p")
> colnames(indiv_forecasts) <- ind_nam

```


3.3. Combining individual forecasts. We loop through all the options provided by the function, and print the results to the screen. We compare those results with those that can be reached by exclusively rely on a single model chosen beforehand. We will see that there are indeed accuracy gains, in line with the argumentation presented and previous literature.

```
> scheme= c("simple", "ols", "robust", "variance based", "cls", "best")
> combine_f <- list()
> for (i in scheme) {
+   combine_f[[i]] <- Forecast_comb(obs = dtest_total[ind2, "medv"],
+   fhat = indiv_forecasts[ind2, ],
+   fhat_new = indiv_forecasts[-ind2, ], Averaging_scheme = i)
+   tmp <- round(
+     sqrt(mean((combine_f[[i]]$pred - dtest_total[-ind2, 14])^2)),
+     3)
+   cat(i, ":", tmp, "\n")
+ }

simple : 3.859
ols : 3.12
robust : 3.065
variance based : 3.373
cls : 3.225
best : 3.3
```

We start by comparing those numbers with an evaluation of the individual forecasts themselves. What would be the results if we would simply choose the one forecasting model beforehand.

```
> rmse <- function(x,y){
+   sqrt(mean((x - y)^2))
+ }
> print(apply(indiv_forecasts[-ind2,], 2, rmse, y = dtest_total[-ind2, 14]),
+   digits= 3 )

  lm_p  pcr_p boost_p  svm_p  rf_p  nn_p
4.08  9.55  4.00  3.95  3.30  3.52
```

As is,³ *random forests* is the best performing individual model, with the lowest Root Mean Squared Error (RMSE). Out of the available combination schemes, only the variance-based and the simple average are worse than the best performing individual model. OLS combination or LAD combination are the best performers.

³Results may differ due to the randomness of data split between training and testing sets. Based on numerous runs, qualitatively, results stand firmly.

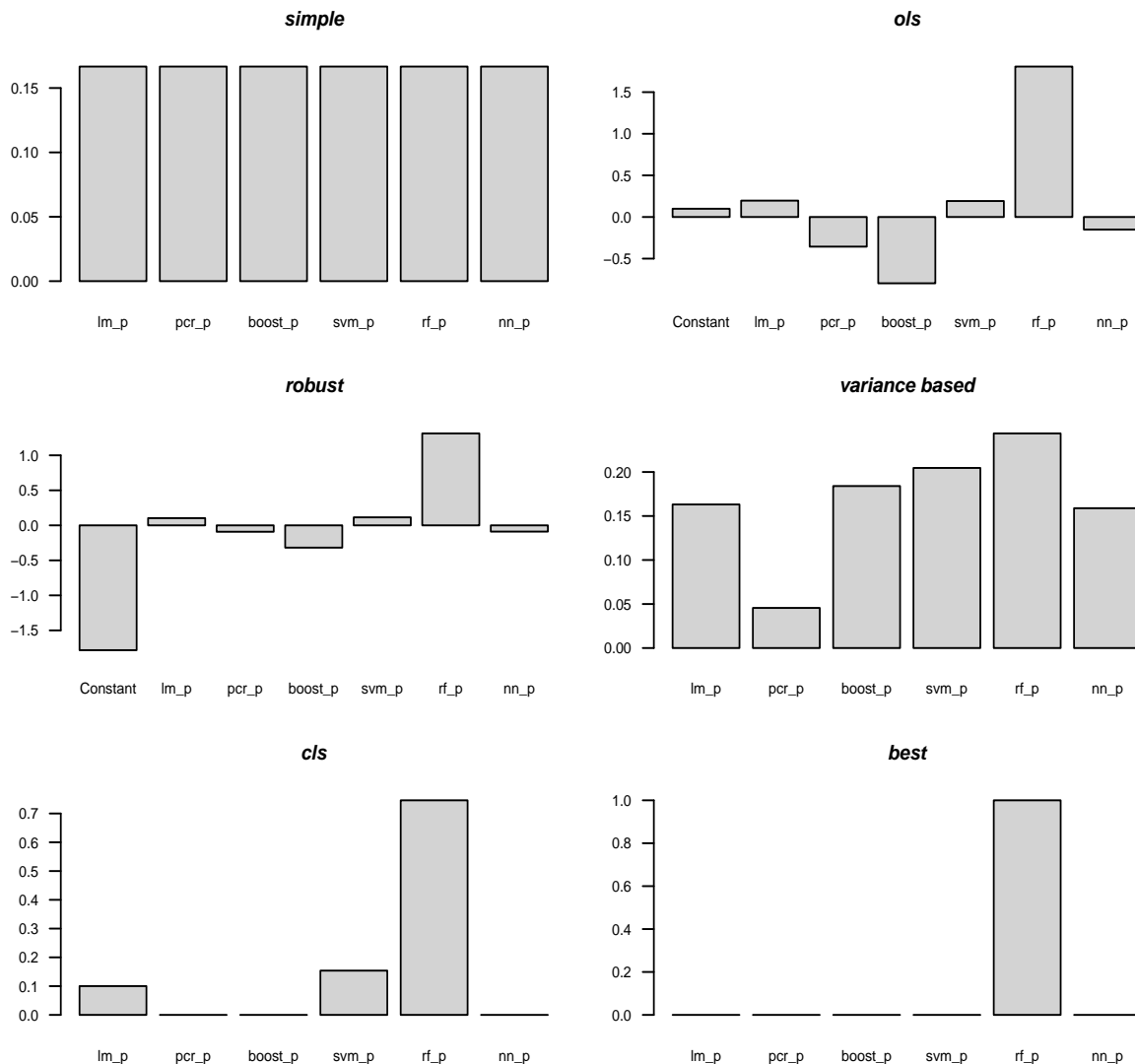


FIGURE 1. Weights of individual forecasts in the overall combined forecast

null device

1

To shed some more light, Figure (1) plots the resulting weights from each of the combination schemes. For the sake of completeness, the simple average and the constants in those regression-based schemes are plotted as well.

Several insights emerge. Ostensibly, given the good performance of the *random forests* model, it receives the highest weight in the overall combined forecast in all but the simple combination schemes. We also see that the constant in the regression-based forecast combinations is quite negative, suggesting a possible bias of all or few of the individual forecasts. Finally, we see that the CLS improves over the best individual model by allowing for a slight positive weight in the second best individual forecast, the SVM.

This is typical, and serves here as an important illustration of the counter intuitive nature of forecast combinations. We enjoy increased accuracy by allowing forecasts which

are *not* best performing in comparison, to enter computation and exert their pull over the final combined forecast. The strength in which those less accurate forecasts (again, in comparison) are given is strictly an empirical question, which is at least partially the reason for the colossal literature formed over the years.

As described above, another way to create a combined forecast is using a complete subset regressions. The following code demonstrates the use of the `Forecast_comb_all` function. Using the 6 forecasts constructed in the previous subsection,

```
> combine_f_all <- Forecast_comb_all(obs = dtest_total[ind2, "medv"],
+ fhat = indiv_forecasts[ind2, ], fhat_new = indiv_forecasts[-ind2, ])
|
|                                     | 0%
|
|=====                             | 20%
|
|=====                             | 40%
|
|=====                             | 60%
|
|=====                             | 80%
|
|=====                             | 100%
> # To get the combined forecasts from the all subset regression:
> Combined_f_all_simple <- apply(combine_f_all$pred, 1, mean)
> print(sqrt(mean((Combined_f_all_simple - dtest_total[-ind2, 14])^2)),
+       digits= 3 )
[1] 2.99
```

Finally, we can use the information-theoretic, combining all possible combos based on say the AIC criteria as in equation (7):

```
> # To get the combined forecasts based on aic criteria for example:
> Combined_f_all_aic <- t( combine_f_all$aic %*% t(combine_f_all$pred) )
> print(sqrt(mean((Combined_f_all_aic - dtest_total[-ind2, 14])^2)),
+       digits= 3)
[1] 3.1
```

4. Discussion and Conclusions

Some of the functions presented in this paper, despite being geared towards forecast combinations, can be used in other ways as

```
> CLS_reg <- Forecast_comb(obs = dtest_total[, 14],
+ fhat = as.matrix(dtest_total[, -14]), Averaging_scheme = "cls")
```

```
> coef_cls <- round(CLS_reg$weights, digits = 2)
> names(coef_cls) <- colnames(dtest_total[, -14])
> print(coef_cls[coef_cls!=0], digits = 3)
      zn      rm black
0.09  0.87  0.04
```

In the same fashion, the function `Forecast_comb_all` can be used to create a complete subset regressions object using all the original explanatory variables (the first 13 columns data).

```
> Complete_subset_reg <- Forecast_comb_all(obs = dtest_total[ind2, 14],
+ fhat = as.matrix(dtest_total[ind2, -14]),
+ fhat_new = dtest_total[-ind2, -14])
```

The object `Complete_subset_reg$pred` now contains forecasts from all possible combination of the original explanatory variables.

That said, the ecosystem of R is by now very rich, and so there are other packages that, although not forecasting oriented, are better designed to find the best subset of original explanatory variables, with more sophisticated methods such as forward or backward model selection ([Lumley et al. 2015](#)).

The best way to combine different forecasts has no theoretical underpinnings, a lot depends on the specifics of the data at hand. And so, we implemented some of the most popular ways in which forecasts can be combined, along with a demonstration of available functionalities in the `ForecastCombinations` package.

References

- Robert R Andrawis, Amir F Atiya, and Hisham 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.
- Doron Avramov. Stock return predictability and model uncertainty. *Journal of Financial Economics*, 64(3):423–458, 2002.
- John M. Bates and Clive W. J. Granger. The combination of forecasts. *Operations Research Quarterly*, 20:451–468, 1969.
- Gang Cheng and Yuhong Yang. Forecast combination with outlier protection. *International Journal of Forecasting*, 31(2):223–237, 2015.
- Charlotte Christiansen, Maik Schmeling, and Andreas Schrimpf. A comprehensive look at financial volatility prediction by economic variables. *Journal of Applied Econometrics*, 27(6):956–977, 2012.
- Robert T Clemen. Combining forecasts: A review and annotated bibliography. *International Journal of Forecasting*, 5(4):559–583, 1989.
- Dwight B. Crane and James R. Crotty. A two-stage forecasting model: Exponential smoothing and multiple regression. *Management Science*, 13(8):501–507, 1967.
- Graham Elliott, Antonio Gargano, and Allan Timmermann. Complete subset regressions. *Journal of Econometrics*, 2013.
- 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.
- Clive WJ Granger and Ramu Ramanathan. Improved methods of combining forecasts. *Journal of Forecasting*, 3(2):197–204, 1984.
- Edward J Hannan and Barry G Quinn. The determination of the order of an autoregression. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 190–195, 1979.
- B.E. Hansen. Challenges for econometric model selection. *Econometric Theory*, 21(1):60–68, 2005.
- Bruce E Hansen. Least squares model averaging. *Econometrica*, pages 1175–1189, 2007.
- Bruce E Hansen. Least-squares forecast averaging. *Journal of Econometrics*, 146(2):342–350, 2008.
- Bruce E Hansen and Jeffrey S Racine. Jackknife model averaging. *Journal of Econometrics*, 167(1):38–46, 2012.
- Kurt Hornik, David Meyer, and Alexandros Karatzoglou. Support vector machines in r. *Journal of statistical software*, 15(9):1–28, 2006.
- Clifford M Hurvich and Chih-Ling Tsai. Regression and time series model selection in small samples. *Biometrika*, 76(2):297–307, 1989.
- Rob J Hyndman, George Athanasopoulos, and Han Lin Shang. hts: An r package for forecasting hierarchical or grouped time series. 2015.

- George Kapetanios, Vincent Labhard, and Simon Price. Forecasting using bayesian and information-theoretic model averaging: An application to uk inflation. *Journal of Business & Economic Statistics*, 26(1):33–41, 2008.
- Andy Liaw and Matthew Wiener. Classification and regression by randomforest. *R news*, 2(3): 18–22, 2002.
- Thomas Lumley, Alan Miller, and Maintainer Thomas Lumley. Package 'leaps'. 2015.
- Jan R Magnus and Wendun Wang. Concept-based bayesian model averaging and growth empirics. *Oxford Bulletin of Economics and Statistics*, 76(6):874–897, 2014.
- Björn-Helge Mevik and Ron Wehrens. The pls package: principal component and partial least squares regression in r. *Journal of Statistical Software*, 18(2):1–24, 2007.
- Claudio Morana. Model averaging by stacking. *University of Milan Bicocca Department of Economics, Management and Statistics Working Paper*, (310), 2015.
- Jakub Nowotarski, Eran Raviv, Stefan Trück, and Rafał Weron. An empirical comparison of alternative schemes for combining electricity spot price forecasts. *Energy Economics*, 46: 395–412, 2014.
- Anne Opschoor, Dick JC Van Dijk, and Michel Van der Wel. Improving density forecasts and value-at-risk estimates by combining densities. 2014.
- Adrian E Raftery and Ian S Painter. Bma: an r package for bayesian model averaging. *R news*, 5(2):2–8, 2005.
- David E Rapach, Jack K Strauss, and Guofu Zhou. Out-of-sample equity premium prediction: Combination forecasts and links to the real economy. *Review of Financial Studies*, page hhp063, 2009.
- Francesco Ravazzolo, Marno Verbeek, and Herman K Van Dijk. Predictive gains from forecast combinations using time varying model weights. *Working paper*, 2007.
- Eran Raviv, Kees E. Bouwman, and Dick van Dijk. Forecasting day-ahead electricity prices: Utilizing hourly prices. *Energy Economics*, 50:227 – 239, 2015. ISSN 0140-9883.
- Greg Ridgeway. Generalized boosted models: A guide to the gbm package. *Update*, 1(1):2007, 2007.
- James H Stock and Mark W Watson. Combination forecasts of output growth in a seven-country data set. *Journal of Forecasting*, 23(6):405–430, 2004.
- Allan Timmermann. Forecast combinations. *Handbook of economic forecasting*, 1:135–196, 2006.
- Berwin A. Turlach and Andreas Weingesse. quadprog: Functions to solve Quadratic Programming Problems. 2013. URL <http://CRAN.R-project.org/package=quadprog>. R package version 1.5-5.
- W. N. Venables and B. D. Ripley. *Modern Applied Statistics with S*. Springer, New York, fourth edition, 2002. URL <http://www.stats.ox.ac.uk/pub/MASS4>. ISBN 0-387-95457-0.
- Jonathan H Wright. Bayesian model averaging and exchange rate forecasts. *Journal of Econometrics*, 146(2):329–341, 2008.
- Jonathan H Wright. Forecasting us inflation by bayesian model averaging. *Journal of Forecasting*, 28(2):131–144, 2009.