Rank-Based Selection Strategies for Forecast Combinations: An Evaluation Study

Magnus Svensson

Master's Thesis in Statistics
15 ECTS credits

April 2019

Supervised by
Peter Gustafsson

Department of Statistics
Lund University

Abstract

This thesis evaluates four of the most popular methods for combining time series forecasts. One aspect that is often overlooked in the literature is the choice of which forecasts to include in a forecast combination. The focus here, is to investigate the variability in forecast accuracy that occurs between all distinct subsets from a fixed set of eleven individual forecasting models that a combination method can be fed with. Six rank-based strategies for selecting these subsets are also evaluated.

The methods are evaluated across more than 1000 monthly time series. The accuracy of one-period-ahead forecasts is analyzed. More than 66 million forecasts are evaluated. The forecasts are assessed by the Mean Absolute Scaled Error metric, and via a Model Confidence Set approach. The latter to be able to generalize the results beyond the evaluation sample.

When selecting number of forecasts to include in a combination, then it is often a matter of balancing risk and reward. The variance of the forecast accuracy between the different subsets of input forecasts is greatest when number of forecasts included is small and decreases as number of forecasts included increases. The results suggest that the mean combination method is especially fragile if poor performing subsets are selected. The three methods that uses training data handles this situation much better. If the performance of some of the input forecasts is lagging behind the rest, then it is recommended not to include these forecasts in a forecast combination.

If there exists a large dataset with similar time series comparable to the one that is being studied, then using this data together with one of the recommended selection strategies may improve the forecast accuracy of a combination method. If this is not feasible, then it is recommended to select input forecasts based on past accuracy.

Keywords: *time series forecasting, combining forecasts, forecast combination, M3-Competition, forecast accuracy, evaluation study, model confidence set*

Contents

1 Introduction

1.1 Background

Combining univariate time series forecasts from multiple models is a popular approach dating back to the seminar paper by Bates and Granger (1969). This approach has been shown to be advantageous in many situations by reducing the forecasting error compared to using a single forecast, as seen by results in meta-analysis conducted by for example Clemen (1989). Most studies in the field focus on evaluating and comparing the average forecast error between different methods for combining forecasts. The performance of a forecast combination computed by a simple arithmetic mean might for instance be compared to a median or a trimmed mean forecast combination. More advanced combination methods formed by weighting the input forecasts by past forecast errors are also often considered.

Forecast combination methods can be seen as a two-step process, the first is about determining which forecasts that should be included, while the second is about how to weight them. One aspect that is often overlooked in the literature is the choice of which forecasts to include in a forecast combination, since most studies only choose to focus on the second step of assigning weights to the forecasting models (Aiolfi et al., 2011). In a typical study, a forecast combination method is only evaluated when all input forecasts from a fixed set of input forecasts are included. Evaluation of the performance of forecast combinations when only a subset of the available input forecasts is utilized is rarely in focus. If a total of r forecasts are available, then selecting a subset k of them can be done in $\binom{r}{k} = r!/(k!\,(r-k)!)$ unique ways. With for example $r = 11$ and $k = 5$ then there are $11!/(5!\,(11-5)!) = 462$ unique ways to pick five forecasts from the total of eleven available forecasts. Makridakis and Winkler (1983), and Hibon and Evgeniou (2005) are some of the few exceptions where this is studied. In Makridakis and Winkler (1983) the mean forecast combination method is studied under two situations, one where $r = 14$ and $k = 2, \ldots, 14$, and another similar with $r = 10$. In Hibon and Evgeniou (2005), the mean forecast combination method is again examined across all distinct subsets when $r = 14$. The gap in performance between the worst and the best subset at a given k is, among other things, measured in these studies, and it seems not to be negligible. Since the focus of Hibon and Evgeniou (2005) is mainly to examine other hypotheses, not much information is given on this subject. Similar analysis but for other popular combination methods, especially those using training data to estimate the weights, are largely lacking in the literature. One reason to this could be that it has been computationally expensive to compute forecast combinations for all distinct subsets of available forecasts. With the sharp increase in number of CPU cores and threads lately, this kind of analysis is becoming more and more accessible. The task at hand requires very little effort to separate the problem into several parallel, and concurrent, tasks.

The conclusion to be made here is that a pre-screening strategy that focus on getting rid of some forecasts before combining the rest might be more powerful than letting a forecast combination method handling this by the inherent nature of the method itself. This may especially be the case if forecasts from poor forecasting models, for some reason, have been included in the set of available input forecasts to begin with. A situation like this is not unlikely in practice. In Svensson (2018), nineteen combination methods were evaluated across more than 800 time series. The evaluation study showed that selecting only $k = 5$ forecasts as input to a forecast combination, instead of all $r = 11$ available, reduced the forecasting error in the sample across all nineteen combination methods studied. The way in which the five forecasts were selected in Svensson (2018) was by computing the average forecast error for each of the eleven forecast models individually, based on an independent, but similar, dataset with around 200 time series. The five models with the lowest average forecasting error were then selected and used as input to a forecast combination. This strategy, however, results in only one of the 462 possible combinations of picking five forecasting models from the eleven. It might be possible to reduce the forecast error even further if one of the other 461 combinations of five input forecast is selected instead. Different strategies could therefore also be evaluated to select k forecasts from r. These strategies might be based on ranking the input forecasting models in various ways, or something else. In the study described, the same five forecasting models were selected and applied to all the time series in the evaluation data set in a static fashion. A more dynamic and individual adapted selection of r forecasting models, unique to a particular time series in question, or even a single period within a time series, might also be part of an improved pre-screening strategy.

A related topic is the selection of how many forecasts that should be utilized in a forecast combination. Armstrong (2001) is one of the few that mention a guiding rule. He argues that, when feasible, five or more forecasts should be combined. More supporting evidence is needed on this topic as well.

1.2 Objectives

The main goal with this thesis is to examine three objectives and they are presented below.

Objective I: For $k = 2, \ldots, r$, respectively, investigate the differences in forecasting error among the $\binom{r}{k}$ distinct subsets of input forecasts that a forecast combination method can be fed with. This is of interest to provide more guiding knowledge about the importance of selecting forecasts to include in a forecast combination.

Objective II: Investigate how simple strategies for selecting a subset of k from r input forecasts to include in a forecast combination perform from a forecasting error standpoint.

Objective III: Provide more supporting guidance about selecting the value of k input forecasts to include in a forecast combination. This is mainly done by comparing the results from the selection strategies in Objective II across $k = 2, \ldots, 11$. Through this analysis it might be possible to understand if forecast combinations when $k < r$ reduces the forecast error compared to when $k = r$. A comparison between the forecast error of the input forecasts, i.e. when $k = 1$, and forecasts from the forecast combinations can also be done. This objective is also loosely related to an analysis of Armstrong's (2001) guiding rule of including five or more forecasts in a combination.

The objectives are evaluated for the top three performing combination methods from Svensson (2018) together with the often-recommended mean forecast combination method (Clemen, 1989; Armstrong, 2001; Genre et al., 2013). These methods are described in the next section. The focus here is to mainly analyze the situation when $r = 11$, and with $k = 2, \ldots, 11$. The input forecast models involved in the analysis is the same as in Svensson (2018) and includes a broad span in forecast accuracy. This is on purpose since it is of interest how well a forecast combination method can handle a situation like this. A brief overview of the models is also provided in the next section. The evaluation includes forecasts for one-period-ahead only. The study is limited to an analysis that is based on real-life time series from the M3-Competition (Makridakis and Hibon, 2000) dataset.

2 Methods

Since the focus of this thesis is time series forecasting, a brief explanation of what here is meant by a time series is presented. A joint annotation is also provided. A time series consists of an ordered sequence of values of a variable at equally, or almost equally, spaced time intervals, thus it is a sequence of discrete time data occurring at separate points in time. An example might be number of bottles of wine sold by a store measured per month. Actual values from a time series is denoted as y_t for the time periods $t = 1, \ldots, T$. The corresponding one-period-ahead point forecast is often represented by $\hat{y}_{t|t-h}$ with $h = 1$ but since all forecasts in this thesis are of this type they are hereafter denoted as just $\hat{y}_t$. Since forecasts from multiple models are utilized as input to a forecast combination method, a more distinct denotation of an input forecast is $f_{t(j)}$, which emphasize from which of the $j = 1, \ldots, k$ input forecast it is referring to. A forecast from a forecast combination method is represented by f_t^C where C indicates the name of the combination method.

The layout of the Method section is divided into six parts. First, the methods used for combining forecasts in this study are presented. This is followed by a short description of the forecasting models used as input to the combination methods. The mean absolute scaled error (MASE) is used for two purposes in the study; in the selection strategies themselves, as well as in the final evaluation of all forecasts. Because of this, the entire section about evaluating forecasts is presented next. Thereafter, an introduction to selecting input forecast to a forecast combination is given, followed by the part where the selection strategies are defined. The last part describes the ex-post analysis across all distinct subsets.

2.1 Methods for combining forecasts

A forecast combination method uses two or more forecasts from various models as input and combines them according to a certain logic in order to produce the output forecast, the forecast combination. Forecast combination methods can be divided into two different types. The first type utilizes only information at time period t from the k input forecasts $f_{t(j)}$ together with a deterministic weighting scheme that tells us how the forecasts should be combined. This is here referred to as a forecast combination with no training, hence past input forecasts are not being used. Computing the mean or median across the k input forecasts are examples of this type of forecast combination method.

The second type uses training data from the q periods prior to period t as well. A forecast combination at period t therefore contains a two-step procedure. Step one consists of training, or fitting, a combination model using $f_{i(j)}$'s from periods $i = t - q, \ldots, t - 1$ where $0 < q < t$ is an arbitrary integer representing the training length. Step two consists of inserting the $f_{t(j)}$ forecasts at time t into the trained model from step one to receive f_t^C. As pointed out earlier, four forecast combination methods are utilized in this evaluation study. The mean forecast combination method is the only one of the four that does not use training data. The methods are described below.

2.1.1 Mean forecast combination method

Computing an arithmetic mean is likely the most obvious choice when it comes to combining k forecasts at time period t. The mean forecast f_t^{MEAN} is obtained as

$$f_t^{MEAN} = \frac{1}{k}\sum_{j=1}^{k} f_{t(j)}. \tag{1}$$

This method is simply referred to as MEAN henceforth. It should be noted that MEAN might be highly affected if the set of input forecasts contains outliers. The method has previously shown to perform well in evaluation studies (Clemen, 1989; Armstrong, 2001; Genre et al., 2013) but too often evaluation studies disregard the part of designing the universe of input forecasts (Aiolfi et al., 2011). Hence, the situation where poor input forecasting models are found in the mix of available models is usually not studied, and the reported performance of MEAN might not be generalizable under these conditions. More potential disadvantages and advantages with the method is not explained further. The reason for this is that the main subject under the microscope is not the combination method itself but the evaluation of the spread in forecasting error between subsets of input forecasts. The same scarce treatment is given for the other combination methods as well.

2.1.2 Optimal trimmed mean forecast combination method

The optimal trimmed mean combination method is a method that utilizes training data. Let $f^*_{t(j)}$ represent the j:th smallest value of the input forecasts $f_{t(1)}, f_{t(2)}, \ldots, f_{t(k)}$ where $j = 1, 2, \ldots, k$. For the ordered input forecasts, we now have that $f^*_{t(1)} \leq f^*_{t(2)} \leq \cdots \leq f^*_{t(k)}$. A trimmed mean 'trims' off a certain proportion $0 \leq p \leq 0.5$ of the ordered forecasts in each tail and thereafter computes a mean on the remaining ones. When $p = 0$, a trimmed mean simplifies to the simple arithmetic mean while $p = 0.5$ gives the median. The trimmed mean can be viewed as an interpolation between the mean and the median and has been proposed by, for example, Armstrong (2001) and Jose and Winkler (2008). A, non-optimal, trimmed mean forecast combination, that only uses forecasts at period t as input, is obtained as

$$f_t^{TM(p)} = \frac{1}{k - 2\lfloor kp \rfloor}\sum_{j=\lfloor kp \rfloor + 1}^{k-\lfloor kp \rfloor} f^*_{t(j)} \tag{2}$$

where $\lfloor \cdot \rfloor$ is the floor function that gives as output the greatest integer less than or equal to the expression stated in it. The p in $f_t^{TM(p)}$ refers to the trimmed proportion selected. By using the q periods prior to t, an exhaustive brute-force search through an equidistant spaced subset of $p \in [0, 0.5]$ can be used in order to find the p that minimizes the sum of the squared errors in the training set. Hence, an optimal trimmed mean forecast combination is obtained by finding the p that minimizes the criterion

Q^{OTM},

$$Q^{OTM} = \min_{p} \sum_{i=t-q}^{t-1} (y_i - f_i^{TM(p)})^2 \tag{3}$$

where $f_i^{TM(p)}$ corresponds to the formula in (2). The optimal trimmed mean forecast combination, f_t^{OTM}, is then computed with the optimal p from (3) as input to (2). This method is simply referred to as OTM from this point forward.

2.1.3 MSE-weighted forecast combination method

Using training data, the mean squared error (MSE) for each of the k input forecast models, respectively, can be computed based on the q periods prior to t. MSE for an input forecast model is computed as

$$\text{MSE}_j = \frac{1}{q} \sum_{i=t-q}^{t-1} (y_i - f_{i(j)})^2. \tag{4}$$

Weights, ω_j, for each of the input forecast models are then obtained by

$$\omega_j = \frac{1/\text{MSE}_j}{\sum\limits_{j=1}^{k} 1/\text{MSE}_j} \tag{5}$$

where $\sum_{j=1}^{k} \omega_j = 1$. The MSE weighted forecast combination at period t is then calculated as

$$f_t^{MSEW} = \sum_{j=1}^{k} \omega_j f_{t(j)} \tag{6}$$

and henceforth referred to as MSEW. This method is one of the methods mentioned in Bates and Granger (1969).

2.1.4 Constrained least squares forecast combination method

The forth forecast combination method is based on an equality and non-negative constrained least squares regression approach. Using the training data, estimates of the weights ω_j are obtained by minimizing the error sum of squares,

$$\min_{\omega_1,\ldots,\omega_k} \sum_{i=t-q}^{t-1} \Big(y_i - \sum_{j=1}^{k} \omega_j f_{i(j)}\Big)^2 \tag{7}$$

subject to the constraints $\omega_j \geq 0$ and $\sum_{j=1}^{k} \omega_j = 1$ with respect to the regression parameters $\omega_1, \ldots, \omega_k$. The weights can be estimated by formulating it as a quadratic optimization programming problem and solve it numerically. Once the estimated weights, $\hat{\omega}_j$, are retrieved, the forecast combination is obtained as

$$f_t^{CLS} = \sum_{j=1}^{k} \hat{\omega}_j f_{t(j)} \tag{8}$$

and it is simply referred to as CLS from now on. The CLS method has a good track record of outperforming the less restrictive OLS forecast (Aksu and Gunter, 1992; Gunter, 1992; Genre et al., 2013; Svensson, 2018).

2.2 Input forecast models

The set of forecasting models, or procedures, utilized as input to the forecast combination methods are the same as in Svensson (2018). The eleven forecast models are readily available for automatic, or at least semi-automatic, utilization through the R programming language (R Core Team, 2018). The input forecast models are listed in Table 2.1 where abbreviations of the models are provided, together with a brief explanation of each model. Since the primary focus in this thesis is the selection of input forecasts to the forecast combination methods, not much details about the input forecast models are provided here. A more detailed, but still very simplified, explanation of each model is provided in Svensson (2018). For a more in-depth explanation, the reader is advised to read the R package's descriptions and the references given there. The input forecast models AUTOARIMA, ETS, FFNN, SES, TBATS, THETA and TSLM are found in the R package `forecast` (Hyndman, 2017). The MAPA model is available through the `MAPA` package (Kourentzes and Petropoulos, 2017), PROPHET is available via `prophet` (Taylor and Letham, 2017), THIEF via the `thief` package (Hyndman and Kourentzes, 2018) and PSF is available via the `PSF` package (Bokde et al., 2017). The default setting of each R function is utilized but with a few exceptions as discussed in Svensson (2018).

Table 2.1: Overview of the forecasting models utilized as input to the forecast combination methods in the evaluation study.

Input forecast	**Brief explanation**
AUTOARIMA	Fits a 'best' autoregressive integrated moving average model.
ETS	Exponential smoothing state space model.
FFNN	Feed-forward neural network with single hidden layer.
MAPA	Multiple Aggregation Prediction Algorithm, where multiple time series are constructed through temporal aggregation and ETS forecasts are generated for each, and the result is then combined.
PROPHET	Facebook-made algorithm involving additive regression with non-linear trends, seasonal adjustments and automatic change-point detection.
PSF	Fits a model based on the 'pattern sequence-based forecasting algorithm'.
SES	Simple exponential smoothing.
TBATS	Fits a model centered around Box-Cox transformation, ARMA errors, trend and trigonometric seasonal components.
THETA	Simple exponential smoothing with drift.
THIEF	Temporal Hierarchical Forecasting, with ETS forecasts computed on temporal aggregated time series combined with a hierarchical reconciliation methodology.
TSLM	Fits a linear regression with a time trend and a seasonal component.

2.3 Evaluating forecast accuracy

An important part in any evaluation study of this kind is measuring the forecast accuracy. An often utilized approach in forecast evaluation is through so called out-of-sample evaluation (Hyndman and Koehler, 2006; Elliott and Timmermann, 2016), and this is used here as well. More precisely, an expanding window approach is used to compute the one-step-ahead forecasts from the input forecasting models for period t by using data up to and including period $t-1$ in the model fitting stage. This is done in an iterative fashion. With $t = 110$ then $t = 1, \ldots, 109$ are used in the model fitting stage, and with $t = 111$ then $t = 1, \ldots, 110$ are used in the model fitting stage, and so on. For the forecast combination methods, a rolling window with the q periods prior to t is used in the model training stage, when needed.

This subsection introduces methods for computing and evaluating forecasting errors, and how to compare the forecasting performance across forecasts generated from different models. The most viable way of measuring the forecasting error from one model is through calculating an average forecast error across multiple forecasts. This average forecast error can however be computed in many ways. Here, one method is explained. A statistical test for comparing forecasts from one forecast model against another is also described.

2.3.1 Mean Absolute Scaled Error (MASE)

Comparing the forecast error across multiple time series where the scales of the values might be rather different between time series requires that an evaluation measure takes this into account. In Hyndman and Koehler (2006), the author presents the mean absolute scaled error (MASE), which is a scale invariant forecast error measure. Another popular scale invariant measure is the mean absolute percentage error (MAPE) (Tofallis, 2015) but it is stained with many problems, as briefly discussed in Hyndman and Koehler (2006). Measures that are based on the adjusted (or symmetric) mean absolute percentage error, that was first introduced by Armstrong (1985, p. 348), correct for some of the issues but are not without criticism (Ord, 2001). Using a root mean squared error (RMSE) that is scale invariant is appealing because of the theoretical relevance of RMSE in statistical modelling. However, a squared error is more sensitive to outliers compared to an absolute error and authors like Armstrong (2001) has recommend against their use in forecast evaluation (Hyndman and Koehler, 2006). The MASE measure is the one utilized in this evaluation study.

Let e_t denote the forecast error for a certain forecast f_t at time t. It can be obtained as

$$e_t = y_t - f_t. \tag{9}$$

The MASE computation is a two-step process. First, a scaled error is computed and then the mean across all scaled errors is calculated. To take scale invariance into account, the scaled error utilized by MASE scales the $|e_t|$ by dividing it with the in-sample mean absolute error from a naïve, random walk, forecast. In this case, 'in-sample' refers to observations up to but not including the period t for which f_t is computed. This naïve forecast might consider seasonality but here it is set equal to the previous observation in the time series. The scaled error utilized by MASE is computed as

$$\text{ScE}_t = \frac{e_t}{\frac{1}{t-2}\sum_{i=2}^{t-1}|y_i - y_{i-1}|}. \tag{10}$$

With $|\text{ScE}_t| < 1$ then a single observed f_t performs better than the mean naïve in-sample forecast. The opposite, $|\text{ScE}_t| > 1$, occurs when a single observed f_t performs worse compared to the mean naïve in-sample forecast. To simplify how MASE is expressed, let us consider a single sequence with $t = 1, \dots, n_\gamma$ observations containing both y_t and f_t. The MASE score is then obtained as

$$\text{MASE}_\gamma = \frac{1}{n_\gamma} \sum_{t=1}^{n_\gamma} |\text{ScE}_t|. \tag{11}$$

The computation can easily be generalized to more than one time series by computing the mean in (11) not only across n_γ observations belonging to a single time series but across multiple time series as well. With MASE defined, then MASE is non-negative with a value closer to zero represents a lower forecast error, i.e. a better accuracy. A MASE $= 1$ indicates that a forecast model is performing equal to the mean naïve in-sample forecast. If MASE < 1 then this indicates a lower forecast error for the forecast model in question compared to the mean naïve in-sample forecast. The reverse appears when MASE > 1. The only situation when MASE is undefined is if all in-sample observations are held equal for one or more of the time series.

2.3.2 Testing for equality of forecast accuracy

The MASE score for one forecasting model, as presented above, is a sample average across observations in the evaluation dataset. Even in recent work in many disciplines, it is of common practice to report these sample estimates without uncertainty (Elliott and Timmermann, 2016). As a rule of thumb, this approach might be good enough to detect large differences in effect size between competing forecasts when the evaluation sample size is large. However, even a small difference in forecasting performance between two forecasting models can potentially be worth examining. The gains from even a small increase in forecast accuracy may be beneficial when considering the cost aspect, for example monetary or time-wise. Another, related, issue when uncertainty is not considered, is the fact that in forecasting 'horse races' there is usually one forecasting model that is declared the winner in the evaluation sample. Still, the result may not be generalizable to the population from which the sample belongs, because the win is not statistically significant. Because of these aspects, statistical tests are also applied to detect statistically significant differences in forecasting accuracy. The Diebold-Mariano (DM) test (Diebold and Mariano, 1995) is one of the most common used in the forecast evaluation literature (Diebold, 2015). The DM test is utilized for pairwise comparison of forecasts from competing models. However, when number of competing models m is large then $m(m-1)/2$ pairwise DM tests must be performed, and this is related to the multiple comparison problem. One way to take this problem into account is through a family-wise error rate correction, like a Šidák or Bonferroni correction, that basically means stricter significance thresholds for each of the individual pairwise comparisons (see for example Abdi, 2007). White's Reality Check (White, 2000) is another popular approach that limits the number of comparisons by only comparing to a predefined benchmark model. However, a choice of a benchmark model is not always obvious, and this is the situation in this study.

One thing that complicates the evaluation procedure in this thesis even more is that competing models, or strategies, can be compared at different values of k, causing the multiple comparison problem to get even worse. A paired comparison design, like an ANOVA with block design as a noise reduction technique, as a first step in the analysis, is usually not feasible when number of blocks, here n_γ, is large. In memory-bound programming environments, as R, this means that the sheer size of the model to be estimated might exceed the available amount of random access memory (RAM). An attractive approach when the number of competing forecasting alternatives is large, or needs to be performed in many scenarios, is through the Model Confidence Set (MCS) described in Hansen et al. (2011). An MCS is a set that contains one or more models and is built recursively in such a way that it contains the best model with a given level of confidence. The MCS approach uses a test for equal predictive ability to compare models and an elimination rule to cut off inferior models. The MCS approach is the main tool used in this study to test for differences in forecast accuracy and is described rather briefly in the next subsection. For more details, the reader is referred to Hansen et al. (2011).

2.3.2.1 Model Confidence Set

The goal of the MCS procedure is to determine the set of models, $\mathcal{M}^*$, that contains the best model, or models, from an initial set of models, $\mathcal{M}^0$, based on a loss function decided by the user. The term 'models' is used here but it may refer to objects in general, as for example competing forecasts, forecasting models or a set of alternative strategies. More formally, consider a set, $\mathcal{M}^0$, that consists of a finite number of models under evaluation that are indexed by $i = 1, \ldots, m_0$. Further, the loss function, L, when evaluating point forecasts for model i in period t is denoted as $L_{i,t} = L(y_t, f_{i,t})$ where $t = 1, \ldots, n_\gamma$. The exact form of the loss function $L_{i,t}$ can be the squared error, $(y_t - f_{i,t})^2$, the absolute error, $|y_t - f_{i,t}|$, or even a scaled absolute error like $|\text{ScE}_{i,t}|$. The latter is used in this study. Also, let $d_{ij,t}$ denote the relative loss differential between models i and j and define it as

$$d_{ij,t} = L_{i,t} - L_{j,t} \tag{12}$$

for all $i, j, \in \mathcal{M}^0$. The MCS approach assumes that $\mu_{ij} = \mathrm{E}(d_{ij,t})$ does not depend on t and is finite for all $i, j \in \mathcal{M}^0$. The set of superior models is defined as

$$\mathcal{M}^* = \{i \in \mathcal{M}^0 \colon \mu_{ij} \leq 0 \text{ for all } j \in \mathcal{M}^0\}. \tag{13}$$

The MCS approach uses a sequential testing procedure where models that are significant inferior to other models in $\mathcal{M}^0$ are eliminated stepwise to determine $\mathcal{M}^*$. The null hypothesis that is being tested is

$$H_{0,\mathcal{M}} \colon \mu_{ij} = 0 \text{ for all } i, j \in \mathcal{M} \tag{14}$$

where $\mathcal{M} \subset \mathcal{M}^0$, and with the alternative hypothesis being

$$H_{1,\mathcal{M}} \colon \mu_{ij} \neq 0 \text{ for some } i, j \in \mathcal{M}. \tag{15}$$

When $H_{0,\mathcal{M}}$ is rejected this indicates that at least one of the models in $\mathcal{M}$ is inferior and the model that contributes most to rejecting $H_{0,\mathcal{M}}$ should in a next step be eliminated from $\mathcal{M}$. There are many different test statistics that can be used in the sequential MCS testing procedure. The range statistic is the one preferred in Hansen et al. (2014), and this is utilized here as well. First define the relative loss differential statistic as $\bar{d}_{ij} = \frac{1}{n_\gamma} \sum_{t=1}^{n_\gamma} d_{ij,t}$ where $\bar{d}_{ij}$ represents the relative sample loss between models i and j. With the help of $\bar{d}_{ij}$, then form the t-statistic as,

$$t_{ij} = \frac{\bar{d}_{ij}}{\sqrt{\widehat{\mathrm{var}}(\bar{d}_{ij})}} \tag{16}$$

for $i, j \in \mathcal{M}$ where $\widehat{\mathrm{var}}(\bar{d}_{ij})$ represents the estimate of $\mathrm{var}(\bar{d}_{ij})$. The null hypothesis $H_{0,\mathcal{M}}$ can then be tested with the test statistic

$$T_{R,\mathcal{M}} = \max_{i,j \in \mathcal{M}} |t_{ij}| \tag{17}$$

and this is related to the equivalence test $\delta_{R,\mathcal{M}}$ mentioned by Hansen et al. (2011). If $H_{0,\mathcal{M}}$ is rejected, then the elimination rule $e_{R,\mathcal{M}} = \arg\max_{i \in \mathcal{M}} \sup_{j \in \mathcal{M}} t_{ij}$ is used to eliminate a model. Because the asymptotic distribution of the test statistic is non-standard and does depend on nuisance parameters a block bootstrap procedure is utilized to estimate the distribution. Another important assumption to point out is that the MCS procedure does allow for nonstationarity in $L_{i,t}$ when all models in $\mathcal{M}^0$

are influenced similarly in order to protect the stationarity of $d_{ij,t}$. The fixed rolling window out-of-sample evaluation approach applied to the forecast combinations, as described in a previous section, is also preferred to deal with the potential problem, even if a recursive expanding evaluation scheme produces very similar results according to Hansen et al. (2011). The algorithm of MCS can basically be described in three steps,

1. Set $\mathcal{M} = \mathcal{M}^0$.
2. Test the null hypothesis $H_{0,\mathcal{M}}$ using the equivalence test $\delta_{R,\mathcal{M}}$ at significance level α.
3. If $H_{0,\mathcal{M}}$ is not rejected then define the model confidence set with the coverage probability $1 - \alpha$ as $\widehat{\mathcal{M}}^*_{1-\alpha} = \mathcal{M}$, else utilize the elimination rule $e_{R,\mathcal{M}}$ to remove a model from $\mathcal{M}$ and go to step 2.

The result from the testing procedure can easily be summarized through the MCS p-values, p_{MCS}. With p_{MCS} it is straightforward to judge if a certain model is in $\widehat{\mathcal{M}}^*_{1-\alpha}$ for any given α. This means that for model i that is a member of $\widehat{\mathcal{M}}^*_{1-\alpha}$ then $\alpha \leq p_{\text{MCS}}$ holds, and the opposite occur when $\alpha > p_{\text{MCS}}$, i.e. when i is not a member of $\widehat{\mathcal{M}}^*_{1-\alpha}$. The MCS p-values corresponds well with the meaning of the p-values in classical statistics since the model confidence set contains the best, or 'true', model with a probability of at least $1 - \alpha$. The MCS procedure is available in R via the `rugarch` package (Ghalanos, 2019).

2.4 Selecting input forecasts to include in a forecast combination

The earlier described forecast combination methods say nothing about which input forecasts that should be included, by default all available are included. Results from the evaluation study in Svensson (2018) suggests that selecting only a subset of all available input forecasts may improve the forecast accuracy of a forecast combination considerably. The trick is to define a repeatable and consistent strategy that, in advance, selects forecast models that improve the accuracy of a forecast combination while ignoring forecast models that reduce it. There are of course many ways to define such a pre-screening strategy. Note that one distinct subset of input forecasts might not always perform optimal across different forecast combination methods, i.e. the optimum selected subset is dependent on the inherent nature of the combination method.

Let R and K be finite sets of input forecast models and let K be a subset of R, i.e. $K \subsetneq R$. Also let $r = |R|$ and $k = |K|$ be the cardinality of the sets. Thus R is a

r-element set of forecast models, and K is a k-element subset of R, with $k < r$. For an arbitrary and valid value of k, there are $C(r,k) = \binom{r}{k} = r!/(k!\,(r-k)!)$ distinct subsets, i.e. combinations, of k forecasts that can be drawn from the set R, without replacement. Let K_δ represent one of the distinct subsets with $\delta = 1, \ldots, C(r,k)$. The order of the drawn k forecasts is not of importance here since a forecast combination method will give the same result independent of the order.

An example illustrates the scenario. Let $R = \{f_{(1)}, f_{(2)}, f_{(3)}\}$, i.e. $r = |R| = 3$. If two elements are drawn from R, $k = 2$, then there are $\binom{3}{2} = 3!/(2!\,(3-2)!) = 3$ distinct 2-element subsets. The three subsets, let them be denoted K_1, K_2 and K_3, are $K_1 = \{f_{(1)}, f_{(2)}\}$, $K_2 = \{f_{(1)}, f_{(3)}\}$ and $K_3 = \{f_{(2)}, f_{(3)}\}$. Given $k = 2$, then either K_1, K_2 or K_3 could be utilized as input to a certain forecast combination method. Depending on which of the subsets that are being fed to a combination method, different forecasts will likely be produced. A certain pre-screening strategy A might select K_1 while a strategy B might select K_2, with the end-result being potentially different forecast combinations even if k is the same.

The basic idea behind combining forecasts is based on the fact that there exists variation between forecasts from different input models. Without any variation at all, there is no reason to combine whatsoever. In the case without any variation, just pick any of the input forecast models; this will lead to the most accurate forecast. However, the variation between forecasts from different input forecast models, that we strive for, might lead to the problem which can potentially be reduced by selecting a subset of R. For all four forecast combination methods described, the value of the forecast combination itself is an interpolation ranging from the minima to the maxima of the values of the input forecasts, once they are available. If all r input forecasts are utilized then $f_t^{C|R} \in [f^*_{t(1)}, f^*_{t(r)}]$ and if a subset K of the input forecasts is used then $f_t^{C|K} \in [f^*_{t(1)}, f^*_{t(k)}]$. It is in advance difficult to say which of $f_t^{C|R}$ and $f_t^{C|K}$ that will be most accurate if no additional information is available.

The mean combination, f_t^{MEAN}, is especially influenced by outliers, as pointed out earlier. As an example, let $r = 5$ so that $R = \{f_{(1)}, f_{(2)}, f_{(3)}, f_{(4)}, f_{(5)}\}$. Further, let us imaging a situation where we with certainty have information that tells us that at period t the forecast error of $f_{t(4)}$ and $f_{t(5)}$ is much larger, and in the same direction, compared to the corresponding errors for $f_{t(1)}$, $f_{t(2)}$ and $f_{t(3)}$. The weights per input forecast in f_t^{MEAN} are $1/r = 1/5$ when all five forecasts are utilized. The less accurate $f_{t(4)}$ and $f_{t(5)}$ get the same weights as the three more accurate forecasts. By removing $f_{t(4)}$ and $f_{t(5)}$ the weights for $f_{t(4)}$ and $f_{t(5)}$ could be seen as forced to being equal to zero while the weights for the remaining three forecasts increases from $1/5$ to $1/r(1 + (r-k)/k) = 1/3$. By redistributing the weights, a more accurate forecast combination is provided. The forecast combination methods with training can, in some sense, redistribute the weights automatically. However, the weights for $f_{t(4)}$ and $f_{t(5)}$ might still, via this automatic weight redistribution, be greater than zero, and therefore lead to a less optimal forecast combination compared to removing $f_{t(4)}$ and $f_{t(5)}$ entirely. The scenario described is of course an imaginary product that never exists. However, it might be possible to at least imitate it via different selection strategies.

The evaluation study in this thesis utilizes a set with $r = 11$ while k varies from 2 to 10. With $k = 1$, then the combination methods are not combination methods by definition; they are simplified to the input forecast chosen. When $k = 11$, then no real subset is selected since all elements are chosen. However, both situations, with $k = 1$ and $k = 11$, are reported as well for comparison purposes. Number of distinct subsets per k given $r = 11$ is given by Table 2.2. When $k = 5$, then 462 distinct 5-element subsets of R exist. When $k = 8$, then there exist 165, and so on. There are a total $\sum_{k=2}^{10} C(11, k) = 2035$ distinct and relevant subsets. If the cases where $k = 1$ and $k = 11$ are added to the total, it sums up to 2047. This means that with $k = 5$ selected in advance, then there is a maximum of 462 distinct outcomes from a certain forecast combination method, ceteris paribus. One subset, or more if they generate equal outcome, will produce the most accurate forecast. An optimal strategy will therefore always select the subset that leads to the most accurate forecast.

Table 2.2: With $r = 11$, number of distinct k-element subsets at each value of k, i.e. $C(r, k)$.

k	$C(11, k)$
1	11
2	55
3	165
4	330
5	462
6	462
7	330
8	165
9	55
10	11
11	1
$\sum_{k=1}^{11} C(11, k)$	2047

Strategies for choosing an optimal subset K, or close to it, might be built on different foundations. To most naive is to just select one of the $C(r, k)$ subsets at random, and then compute a forecast combination in the next step. The focus here is to utilize training data, from past actual observations and forecasts, in order to select the subset

K. The training data might belong to the time series in question, or from other time series that can mirror the behavior needed to select a legitim subset. Aspects to also consider is the computational time to produce the final forecast combination. A training dataset that involves a longer time horizon of past periods will require that input forecasts are available on these time periods, which in turn will take a certain time to produce. Strategies that utilize data from a large set of time series to select K will of course demand more computation time than a strategy that does not. The most straightforward strategy is probably to rank the input forecast models. The top k ranked forecast models could then be included into the subset K. This ranking procedure could involve evaluating the accuracy of the input forecasts on the past q periods on the time series itself. It could also involve ranking the input forecast models across a relevant set of time series. A strategy might even involve computing forecast combinations for each of the distinct $C(r, k)$ subsets on past training periods and then rank them, this is one of the most computational expensive strategies. A similar strategy might also involve computing the average across forecast combinations from all distinct subsets; an idea much related to the complete subset regression approach described in Elliott et al. (2013). This latter idea, when multiple forecast combinations are averaged, is outside the scope of this paper. Furthermore, a subset K can be selected in a statically fashion, meaning that a certain K is always applied to a certain time series. The opposite can involve a strategy with a dynamic selection of K that may change as new information is available on newly added past periods as we move forward in a certain time series. The strategies evaluated in this study are presented next, and they are all based on simple rank-based selection principles.

2.5 Simple strategies for selecting a subset of input forecasts

To more easily introduce the strategies that follow, a homogeneous annotation is needed. Let Y represent the time series for which a forecast combination is to be obtained. Let U denote a set of different time series, with $Y \notin U$. The set U consists of $\zeta = 1, 2, \ldots, z$ subsets, each defined by their domain type so that $U = \{D_1, D_2, \ldots, D_z\}$. A subset D_1 might for example contain all time series classified as belonging to the "demographic" domain, while D_z contains time series that belongs to the "microeconomic" domain, and so on. The time series Y is in advance labeled to one, and one only, of the defined z domain types in U, when needed.

Six selection strategies are presented next. The first three, S1, S2 and S3, are based on ranking the input forecasts with MASE. The last three, S4, S5 and S6, are based on ranking forecast combinations from all distinct subsets, with MASE. Strategy S1 and S2 uses an independent dataset to rank input forecasts in order to select the ones with the lowest MASE, for a given value of k. This is what is here referred to as a 'static' ranking. Strategy S3 uses past periods from the studied time series to rank the input forecast. This is here referred to as 'dynamic' ranking. Strategy S4 is similar to S1 but ranks forecast combinations from all subsets. The same relationship exists between S5 and S2, and between S6 and S3. Strategy S6 is the naivest, since forecast combinations from all distinct subsets of input forecasts are ranked using only a short window of past observations.

2.5.1 Strategy S1 - Static MASE ranking

1. For all input forecasts models in R, compute forecasts on the s last observations for all time series in U.
2. Compute MASE for each of the r forecast models, respectively, based on data from the previous step.
3. Select subset K_δ based on the k models with the lowest MASE in the previous step. If ties occur, then these are separated by a random draw; the same principle apply to all strategies described.
4. For time series Y, compute f_t^C based on K_δ, with training length q when needed.

2.5.2 Strategy S2 - Static MASE ranking per domain

1. Label time series Y to one of the z domain types ζ in U.
2. For all input forecasts models in R, compute forecasts on the s last observations for all time series in D_ζ.
3. Continue as described in Steps 2 to 4 from Strategy S1.

2.5.3 Strategy S3 - Dynamic MASE ranking

1. For all input forecasts models in R, compute forecasts on the q last observations prior to t in Y.
2. Continue as described in Steps 2 to 4 from Strategy S1.

2.5.4 Strategy S4 - Static MASE ranking across all distinct forecast combinations

1. For all input forecasts models in R, compute forecasts on the $s+q$ last observations for all time series in U. If the forecast combination method requires no training data, then set $q = 0$.
2. For a given k, list all $C(r, k)$ distinct subsets K_δ of input forecasts.

3. Given a forecast combination method, compute f^C based on each K_δ, respectively, from the previous step. Do this for the s last observations from the time series in the initial step. Utilize training length equal to q for a forecast combination method, if needed.

4. For each of the subset K_δ, respectively, compute MASE for forecast combinations based on data from the previous step.

5. Select the subset K_δ that generates the lowest MASE in the previous step.

6. For time series Y, compute f_t^C based on the selected K_δ, with training length q when needed.

2.5.5 Strategy S5 - Static MASE ranking per domain across all distinct forecast combinations

1. Label time series Y to one of the z domain types ζ in U.

2. For all input forecasts models in R, compute forecasts on the $s+q$ last observations for all time series in D_ζ. If the forecast combination method requires no training data, then set $q = 0$.

3. Continue as described in Steps 2 to 6 from Strategy S4.

2.5.6 Strategy S6 - Dynamic MASE ranking across all distinct forecast combinations

1. For all input forecasts models in R, compute forecasts on the $s+q$ last observations prior to t in Y. If the forecast combination method requires no training data, then set $q = 0$.

2. Continue as described in Steps 2 to 6 from Strategy S4.

2.6 Ex-post investigation of the forecasting error across subsets

Since R is a finite set of input forecast models to choose from, this also implies that for a given period t, and r, k, a forecast combination method and q defined in advance if needed, there is a fixed maximum number of outcomes. With $r = 11$ and $k = 5$, then there are potentially 462 different forecasted values; one from each subset. One of the forecasted values, or more if ties occur, will be closest to the actual outcome, and this information will be available afterwards, i.e. ex-post. If forecasts from all subsets are computed and analyzed ex-post, then it is possible to rank the subsets based on

MASE. A subset that produces the smallest forecast error receives the lowest rank and the subset with the largest error receives the highest rank. Three types of ex-post investigations, R1, R2 and R3, are included in the study, and they are presented here.

R1: Rank subsets across all time series and periods

Given a k and a combination method, rank all distinct subsets of input forecasts across all time series and periods in the evaluation dataset according to MASE. Rank equal to one corresponds to selecting the single subset that performs best across all observations in the evaluation dataset. Similarly, the single subset that performs worst corresponds to the last position in the ranking. Two or more subsets with equal observed forecasting errors are ranked in random order to avoid ties in the ranking; the same principle is utilized on all three types of ex-post investigation.

R2: Rank subsets per time series individually

Given a k and a combination method, rank all distinct subsets of input forecasts for each time series individually according to MASE. Then summarize the result across all time series by adding together the errors from a certain position in the per-time series rankings. This means that rank one across all time series consists of always selecting the subset that performs best on each time series individually. Rank one across all time series may therefore not correspond to a single distinct subset, but the best performing subset on each time series individually in the evaluation dataset.

R3: Rank subsets per period within each time series individually

Given a k and a combination method, rank all distinct subsets of input forecasts per time period within each time series individually, according to MASE. This is similar to R2 but with individual rankings of the subsets per period within each time series that are then summarized across all observations in the evaluation dataset. Rank one in this case corresponds to always selecting the subset with the smallest error for each observation, individually, in the evaluation dataset, i.e. for each period within each time series.

R3 possesses a potentially higher degree of variability compared to R2, which in turn possesses the same over R1. This leads to that R3 could potentially obtain a lower forecasting error compared to R2 and R1 given the same position in the rankings. The results from the strategies in the previous section can also be compared to the optimal results obtained ex-post. Through such a comparison, it can be assessed how close to the theoretical limit a certain strategy performs. It should be noted that from the strategies presented in the previous section, some strategies are restricted to using the same subset K_δ across all time series (S1 and S4) while others allow a difference between domains (S2 and S5). The dynamic strategies (S3 and S6) are the least restrictive in that sense and they allow the selected subset, K_δ, to be varied between periods within a time series.

3 Data

3.1 Data generating procedure

The M3-Competition dataset described in Makridakis and Hibon (2000) is utilized to evaluate forecast combinations from the various selection strategies that were presented in the previous chapter. The dataset contains 3003 real-life time series from different domains such as demographic, microeconomic and finance. The M3-Competition dataset is available via the `Mcomp` package (Hyndman et al., 2017) in R. The reason why the dataset was put together to begin with was to evaluate how different forecasting models performs in a real-life scenario across many time series. Only a subset of the original dataset is used here.

In the end, forecast combinations are evaluated using the last ten observations from all time series included in the evaluation dataset. The training length of a forecast combination method is set to $q = 36$ unless otherwise stated. The parameter s, described in the strategy section, is set to $s = 10$. The data needed for the ex-post investigation requires that forecast combinations from each of the four methods, respectively, are computed from all $\sum_{k=2}^{11} C(11, k) = 2036$ distinct subset of R. Once this computational step is made, forecast combinations from all strategies are easily available as well; it is just a matter of selecting them according the principles described in the strategy section. A detailed description of the data generating steps are provided below.

1. From the M3-Competition dataset, select time series reported on monthly level that contain 110 or more observations.

2. Remove time series from domains containing less than 50 time series. Otherwise strategies that utilize only domain-specific data may be too volatile.

3. Randomly assign the time series that remains after Step 2 into two disjoint subsets based on a 20/80 percentage split. The subset that contains 20 percent of the time series is what is earlier denoted as U. The subset that contains 80 percent of the time series is utilized for final evaluation.

4. Compute one-step-ahead forecasts from all eleven input forecast models in R for,

i. the last $10 + 36 = 46$ periods for all time series in U,

ii. and the last $10 + 10 + 36 = 56$ periods for all time series not a member of U (i.e. for the 80 percent subset). The extra ten forecasts are required by Strategy S6.

5. For $k = 2, \dots, 11$, respectively, list all $\sum_{k=2}^{11} C(11, k) = 2036$ distinct subsets, K_δ, of input forecasts.
6. For each subset listed in Step 5, compute one-step-ahead forecast combinations, from each of the four methods, respectively, on the

   i. last 10 periods of each time series in U,

   ii. and the last $10 + 10 = 20$ periods of each time series not a member of U.
7. Apply the strategies described in the Method section, to generate strategy-based forecasts for the last ten observations for all time series $\notin U$. Note that all forecast combinations can be retrieved directly from Step 6.
8. Evaluate the strategy-based forecast combinations from Step 7. If a forecast combination method fails to produce one or more forecasts in Step 6 for a certain time series, then this time series is removed from the whole evaluation dataset. A comparison with the performance of forecast combinations from all subsets, for the last ten periods of each time series not in U, is also readily available by including forecasts from Step 6. Forecasts from the input forecast models from Step 4 are also available for a comparison.

3.2 Data generated

All data is generated in R version 3.5.1 (R Core Team, 2018). There are 1029 time series from the M3-Competition dataset that meet the requirements of Steps 1 and 2 from the data generating procedure. The criterion of 110 observations in a time series as a minimum for inclusion, is to ensure that a relevant amount of observations is available for when both the input forecasts and the forecast combinations are generated. The requirement in Step 2 removes the domain “Other” that contains two time series. Forecasts from the input forecast models are produced without any missingness. However, the CLS forecast combination method fails to generate forecasts from eleven time series in Step 6.[1] A detailed summary of the missing CLS forecasts is provided in Table 3.1. There are two time series with missing forecasts in the set U while nine time series lack forecasts in the final evaluation data set, i.e. the set labelled as $\notin U$. It should be noted

[1] The time series from M3-Competition with missing CLS forecasts are N1943, N2039, N2552, N2553, N2580, N2609, N2623, N2630, N2631, N2635 and N2639.

Table 3.1: Number of unique time series, per k and divided into sets $\in U$ and $\notin U$, for which CLS forecasts could not be computed across one or more of the $C(11, k)$ distinct subsets of input forecasts.

	Number of unique time series with missing forecasts		
k	$\in U$	$\notin U$	Total
2	1	8	9
3	1	9	10
4	2	9	11
5	2	9	11
6	2	9	11
7	2	9	11
8	2	9	11
9	2	9	11
10	2	9	11
11	1	9	10
2 to 11	2	9	11

that this means that CLS forecasts from one or more of the 2036 subsets of R fails to produce forecasts on these time series. The number of time series with CLS forecasts missing is mainly independent of the value of k selected.

The final evaluation data set, from Step 8, is described in Table 3.2. The table contains a summary of the number of time series per domain and how they are divided between set U and the final evaluation set, according to Step 3. Number of time series before and after removal due to missingness is also provided. A total of 1018 time series are used in the evaluation study, of which 203 are used as set U by the static strategies and 815 are for the final evaluation of all strategies. Nine out of the eleven time series with missing forecasts originates from the Finance domain. Each strategy, and each ranked subset in the ex-post analysis, for each combination method and k, is therefore evaluated across 8150 observations, ten from each of the 815 time series, respectively. A total of 1 760 400 forecast combinations from the 6 strategies are evaluated primarily.[2] The ex-post investigation of the optimal selection, as described in an earlier section, requires that forecast combinations from all 2036 distinct subsets of input forecasts from Step 5 are analyzed. For this reason, a total of 66 373 600 forecast combinations for the 8150 observations are generated and evaluated.[3] Since strategy S6 is based on ranking forecast combinations from the different subsets of input forecasts, even more

[2] 6 strategies × 4 combination methods × 9 values of k × 8150 observations = 1 760 400 forecast combinations.

[3] 4 combination methods × 2036 subsets × 815 time series × 10 periods = 66 373 600.

Table 3.2: Number of time series included in the evaluation study, divided per domain, before and after removal of time series due to missing forecasts. A breakdown per domain into time series $\in U$ and $\notin U$, respectively, is also available.

Time series	**Number of time series before removal**	**Number of time series with missing forecasts**	**Number of time series after removal**
Demographic			
$\in U$	22	0	22
$\notin U$	68	0	68
Total	90	0	90
Finance			
$\in U$	31	2	29
$\notin U$	87	7	80
Total	118	9	109
Industry			
$\in U$	69	0	69
$\notin U$	264	2	262
Total	333	2	331
Macro			
$\in U$	49	0	49
$\notin U$	242	0	242
Total	291	0	291
Micro			
$\in U$	34	0	34
$\notin U$	163	0	163
Total	197	0	197
Total			
$\in U$	205	2	203
$\notin U$	824	9	815
Total	1029	11	1018

forecasts are needed to be computed. For this reason, a total of 149 279 520 forecast combinations and 604 758 input forecasts are computed altogether.[4] The number of periods observed per time series across the set of time series included in the study varies from 110 to 144 with a mean and median of 144 periods, respectively. All observed, actual, values are non-negative.

[4] 4 combination methods × 2036 subsets (203 time series × 10 periods + 815 time series × 20 periods) = 149 279 520 forecast combinations; 11 × (203 × 46 + 815 × 56) = 604 758 input forecasts.

4 Results

The Results section is centered around the three objectives presented in the Introduction section. Input forecasts from the eleven forecasting models and the four forecast combination methods, as defined in the Methods section, are applied to the M3-Competition dataset, as explained in the Data section. The results are based on evaluating the one-period-ahead forecasts that are generated for the 8150 observations, ten forecasts from each of the 815 time series, in the evaluation dataset.

4.1 Objective I - Forecast accuracy between subsets

Visualizations of the results from the three ex-post investigations, R1, R2 and R3, are available in Fig. 4.1 and 4.2. From Fig. 4.1 it is seen that the spread in MASE is largest at $k = 2$ for all combination methods, and the spread decreases as k increases. At $k = 11$, when all input forecasts are utilized, only one MASE score is available and therefore the spread shrinks to a single value. The same pattern is seen between R1, R2 and R3, although it is more profound at R3 since the variability is greater. The highest decile (the darkest colored ribbon at the top) covers a large part of the highest scores of MASE, this is especially the case for CLS and MSEW. For R1, note that there are three input forecast models, seen as red dots, that are performing much worse than the rest. Avoiding subsets with input forecasts from the highest decile lowers the maximum MASE score substantially in the evaluation sample. The span between lowest and highest MASE, for a given value of k and for a selected ex-post scenario, is in general highest for MEAN and lowest for CLS and MSEW. The OTM method is placed in-between. The minimum and maximum of MASE from Fig. 4.1 are highlighted in Fig. 4.2 together with the mean and standard deviation of MASE. The figure accentuates the comparison between combination methods, and input forecasts, as well.

The minimum of MASE in R1, i.e. when the best distinct subset of input forecasts is selected in the evaluation, illustrates the best possible forecasting accuracy in this type of scenario in the sample. Forecasts from the best ranked input forecast model generates a higher MASE compared to when $2 \leq k \leq 8$ for all four combination methods. However, the best possible outcome for MEAN is much worse when $9 \leq k \leq 11$. In R2 and R3, as k increases, the increase of the minimum of MASE is more profound. When the best ranked subset is selected for each observation, respectively, in the sample, i.e. in R3, then a deviation is seen. The MEAN method is here able to reach a lower MASE at $2 \leq k \leq 8$ compared to the combination methods that uses training. When the best input forecast always is selected then this beats all possible outcomes from the combination methods that uses training at $k > 2$; for MEAN it is achieved when $k > 3$.

The maximum of MASE, i.e. representing the worst possible forecasting accuracy in the evaluation sample, shows a similar pattern between R1, R2 and R3; even though at different levels of MASE. The input forecasts alone reach a maximum MASE score that is between 0.8 to 1.4 worse than the worst outcomes from the combination methods. The maximum of MASE for the combination methods decreases as k increases. The MEAN method exhibits a higher maximum of MASE compared with the methods that uses training; this applies across all values of k, besides at $k = 2$ when it coincides with OTM.

When it comes to the mean of MASE, then the MEAN method generates a higher MASE in comparison to the methods with training; again, besides at $k = 2$ when it coincides with OTM. At $k \geq 5$, the mean of MASE for CLS and MSEW doesn't decrease much. The results at R1, R2 and R3 are identical and represented by a single plot at R1. The, population, standard deviation of MASE usually decreases as k increases. At $k = 11$ it shrinks to zero, since only one set of input forecasts is available. In R1, the standard deviation for CLS reaches below 0.01 at $k = 6$ and then decreases further as k increases. The behavior for MSEW, and to some extent for OTM as well, is similar but the standard deviation is somewhat higher in the evaluation dataset. This is also seen in Fig. 4.1 when the span between maximum and minimum MASE get less as k increases. The standard deviation for the input forecasts, at $k = 1$, is higher in all scenarios compared with the combination methods. At R1, the standard deviation at $k = 1$ is more than 4 times as high as CLS at $k = 2$ (0.87/0.21).

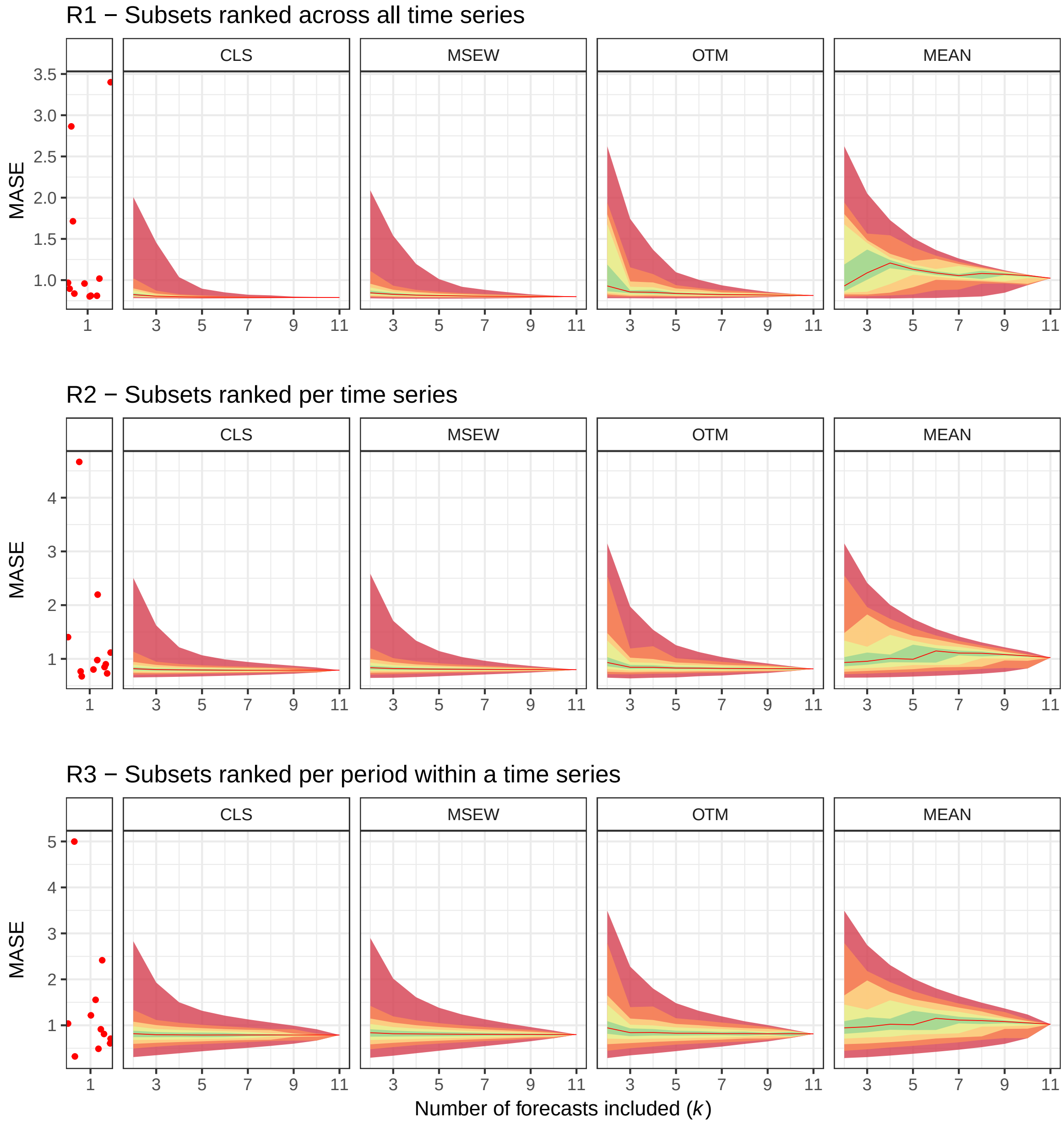

Figure 4.1: This figure shows the distribution of the one-period-ahead MASE scores in three ex-post scenarios (R1, R2 and R3) for each of the four forecast combination methods (CLS, MSEW, OTM and MEAN) when number of input forecasts varies from $k = 2, \ldots, 11$. The left-hand plots on each row shows the input forecasts ranked according to ex-post scenarios (as a jitter of red dots at $k = 1$). The top row (R1) of plots represents when the distinct subsets of input forecasts are ranked across all observations in the evaluation dataset (8150 observations; 815 time series with 10 forecasted periods each). Second and third row of plots shows corresponding information but for R2 and R3, respectively. For each combination method, deciles 1 to 10 of the MASE distribution, for a given k, are shown as colored ribbons around the median (indicated by a red line).

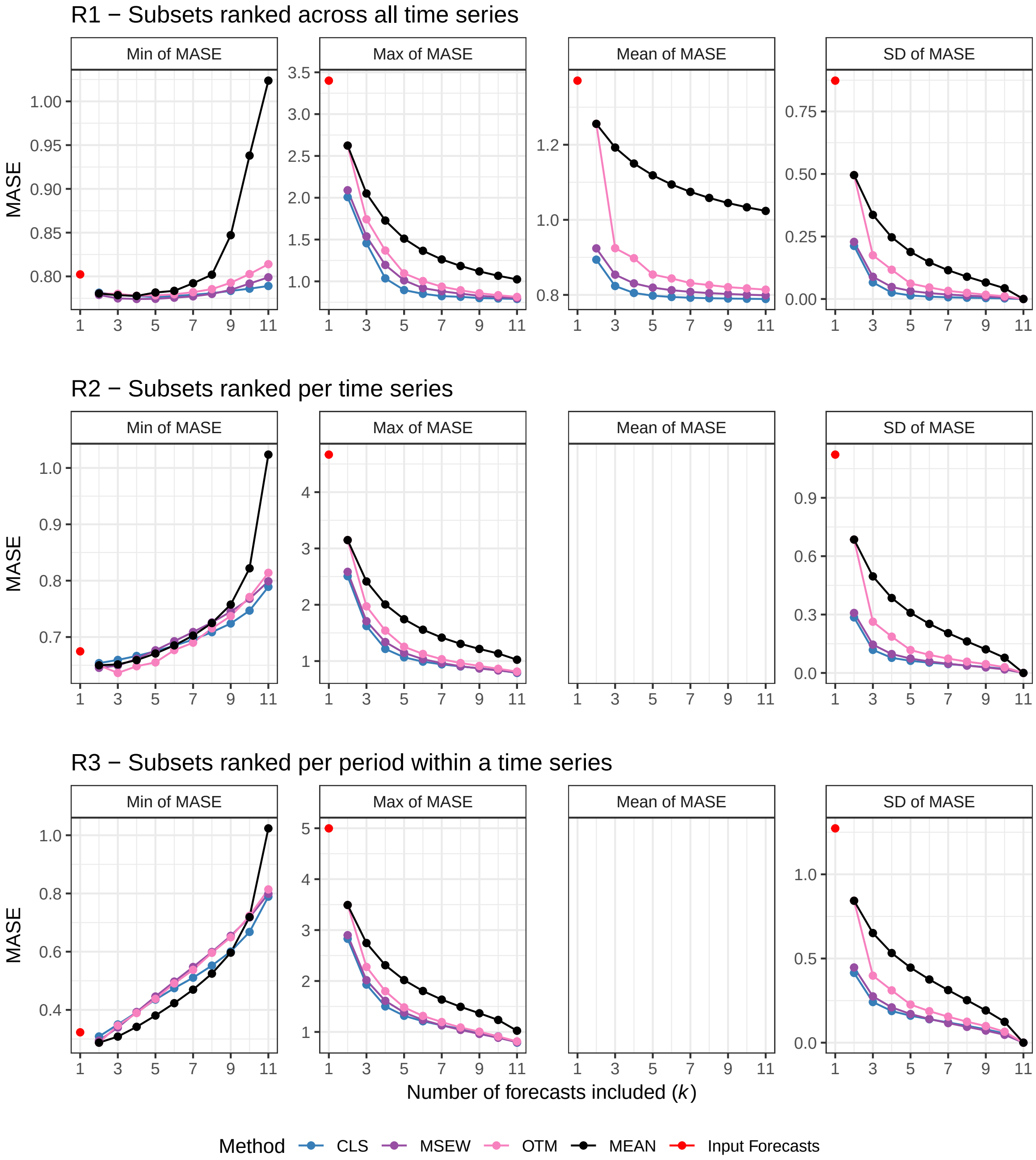

Figure 4.2: This figure shows max, min and mean of the one-period-ahead MASE, as well as the (population) standard deviation (SD) of MASE, from the evaluation dataset (8150 observations; 815 time series with 10 forecasted periods each), for each of the four forecast combination methods (CLS, MSEW, OTM and MEAN) when number of input forecasts varies from $k = 2, \ldots, 11$. Corresponding values for input forecasts are available at $k = 1$. The four metrics are presented columnwise. Each row of plots shows results from one of the three ex-post scenarios (R1, R2 and R3). The Mean of MASE at R1, R2 and R3 are identical and therefore represented by a single plot at R1. Note that all plots have individual y-axes.

4.2 Objective II - Forecast accuracy for selection strategies

The MASE from the evaluation of the six selection strategies are visualized in Fig. 4.3 when applied to each of the four combination methods. The results show that strategies S1, S2, S3, S4 and S5 are often covarying in the sample. Strategy S6 generates a worse MASE compared with the other strategies at lower values of k. This is visible in the sample for all four combination methods. When the strategies are applied to MEAN, then a sharp increase in MASE is seen at the higher values of k. This sharp increase in magnitude of MASE is not as evident for CLS, MSEW and OTM, as seen by the upper row of plots. However, an increase in MASE still exists for the three combination methods that uses training data, but the size of the increase in MASE is much less, as seen in the second row of plots where individual y-axes are utilized. For a comparison, MASE related to the input forecasts is seen on the left-hand side of the upper row of plots but note that the three worst performing models are removed from the plot. This is done on purpose to easier compare the results between the combination methods, as well as with the input forecasts. Plots without the three models removed are available in Fig. A.1 in the appendix.

How the strategies perform in the evaluation sample relative to the full range of possible outcomes of MASE, as defined by the ex-post analyses R1, R2 and R3, is seen in Fig. 4.4. The figure contains the colored ribbon distributions of MASE from Fig. 4.1, but zoomed in, with the strategies from Fig. 4.3 overlaid. In R1, when distinct subsets are ranked across all observations in the evaluation sample, then the strategies are managing to perform close to the best decile of subsets. Compared to R2, then the top five strategies often perform better than 60 to 70 percent of the ranked subsets at lower values of k, when it comes to the combination methods with training; for MEAN it is even better. In ex-post analysis R3, then the performance of the strategies S1 to S5 is closer to the median of MASE for CLS, MSEW and OTM while still better for MEAN.

So far, only results from the sample are reported. To generalize the results beyond just the evaluation sample, a set of MCS test are conducted. The generalization is towards a, hypothetical, population from which the observations in the evaluation sample are, randomly, drawn from. The sequentially tested hypotheses are about equal predictive ability between the strategies, in order to define the model confidence set that contains the ‘best’ strategy with a nominal coverage probability of $1 - \alpha$. The MCS tests are performed independently at $k = 2, \ldots, 10$ for a given combination method. Since nine MCS tests are performed for a combination method, a family-wise Šidák correction is performed to guard against false discoveries, i.e. an adjustment of α as $\alpha_{\text{adj}} = 1 - (1 - \alpha)^{1/m}$ where $m = 9$ is made. The MCS tests uses $|\text{ScE}_t|$ as loss function. Each test uses a block bootstrap with 5000 samples and with a block length of 15. This choice of block length is motivated since the evaluation dataset contains 815 time series, each with ten observations. The selected block length is to secure that the cluster of ten observations from a certain time series usually are drawn together.[5] Since the strategies are designed to select a subset of input forecasts, two or more strategies might select identical subsets across all observation in the evaluation. If this occurs, only one of the strategies are utilized in the MCS test while the other identical strategies obtain the same result. These occurrences are easily spotted through MASE, and the rank of MASE, since the rank will be identical between strategies. Tables 4.1, 4.2, 4.3 and 4.4 shows the results from the MCS tests for CLS, MSEW, OTM and MEAN based strategies, respectively.

For the strategy-based CLS forecasts, strategies S1, S2, S3, S4 and S5 all belongs to the nine model confidence sets with a 95% confidence level, i.e. to the nine $\widehat{\mathcal{M}}^*_{95\%(\text{adj})}$. Strategy S6 is included in them five times. The model confidence sets with 75% confidence level, which are not as 'wide' as the corresponding 95% confidence level counterparts, are more difficult to belong to but at the price of a lower confidence level. Strategies S3, S4 and S6 are here rejected 2, 1 and 5 times, while S1, S2 and S5 are still not rejected from any.

The results from the MCS tests for the strategy-based MSEW forecasts are relatively similar to the results reported for CLS. Here, S4 and S5 are in $\widehat{\mathcal{M}}^*_{75\%(\text{adj})}$ all nine times while S1, S2 and S3 are included 7, 8 and 6 times, respectively. Strategy S6 is lagging behind the rest. Corresponding results for OTM reveals a similar situation. Strategy S6 is rejected 7 times from the model confidence sets with 75% confidence level. The other five strategies are rejected between 0 and 2 times.

The strategy-based MEAN forecasts generate similar results in the MCS tests compared to the forecasts from the combination methods with training. The main difference is that S5 is rejected from the model confidence sets with 75% confidence level three times; this is the first time S5 is rejected across all four combination methods. Two of the times occur at $k = 9$ and $k = 10$ where S6 is the only strategy in the model confidence sets. This pattern is visible in Fig. 4.3 as well. At low values of k, S6 is rejected at $p_{\text{MCS}} < 0.001$. Strategies S1, S3 and S4 are included in $\widehat{\mathcal{M}}^*_{75\%(\text{adj})}$ 7 times.

To summarize the results, strategy S6 is often inferior to the other five strategies. This is true across all four combination methods. Strategies S1, S2, S3, S4 and S5 perform somewhat similar according to the MCS tests. Strategy S5, where forecast combinations from all distinct subsets of input forecasts are ranked per domain in an individual dataset and the subset with the lowest MASE is selected, is included in all 27 model confidence sets with 75% confidence level for the combination methods with training.

[5] Block length of 2, 10 and 20 provides very similar results.

Strategy S3, where input forecasts are ranked and selected based on the performance on past periods for the time series in question, is in many situations not inferior. S3 is however ranked in fifth place, in the evaluation sample according to MASE, on all combination methods that uses training data. Strategy S3 is also tied for first place when it comes to the MEAN method.

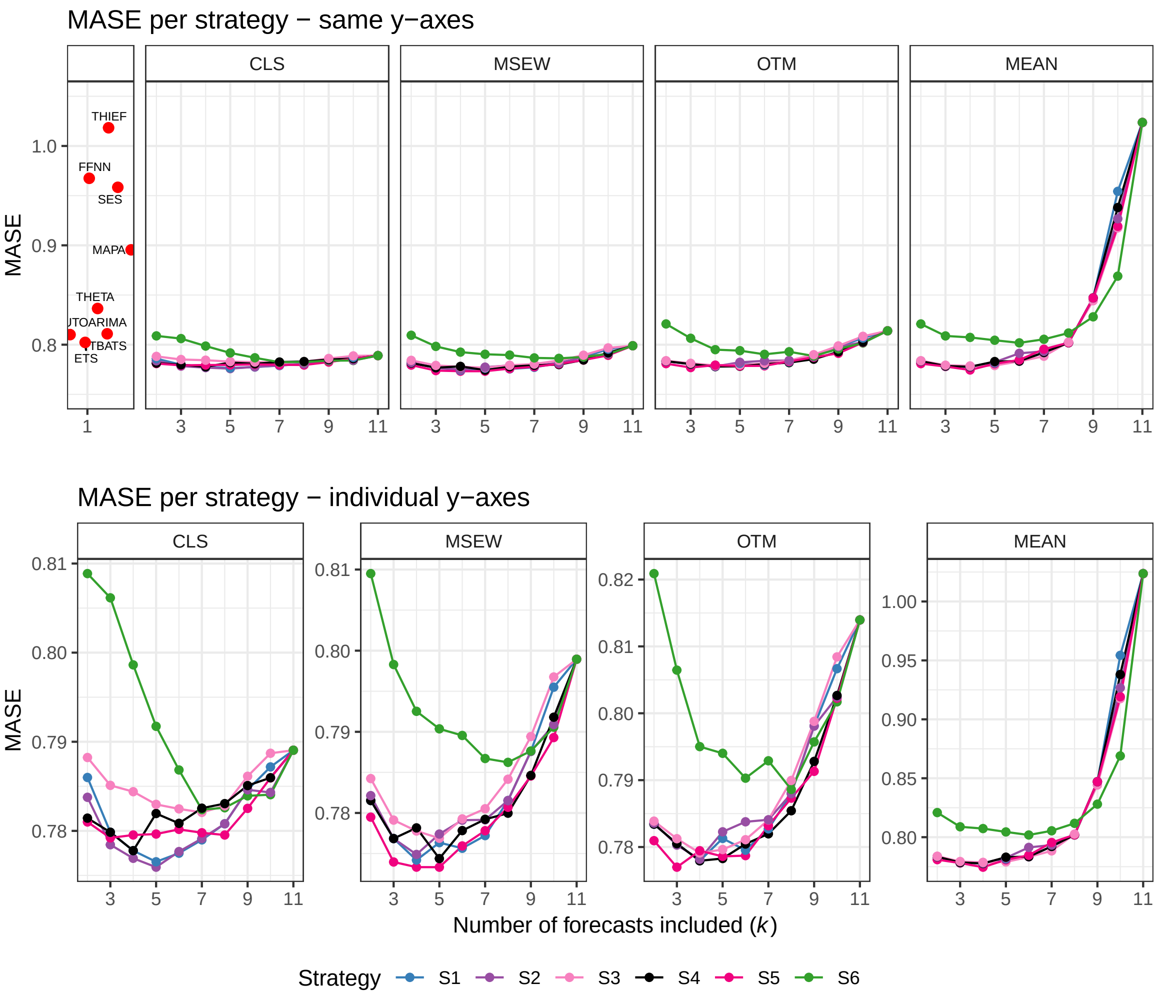

Figure 4.3: This figure shows the one-period-ahead MASE for the six selection strategies (S1, S2, S3, S4, S5 and S6), from the evaluation dataset (8150 observations; 815 time series with 10 forecasted periods each), when applied to each of the four combination methods (CLS, MSEW, OTM and MEAN) when number of input forecasts varies from $k = 2, \ldots, 11$. The upper row of plots also includes the corresponding MASE score for eight of the eleven input forecasts (as a jitter of red dots at $k = 1$, to the left). The three worst input forecasts (PROPHET, MASE = 1.71; PSF, MASE = 2.86; TSLM, MASE = 3.40) are removed for a better comparison. The second row of plots contains the same information as the upper but gives a close-up view; Note that the plots have individual y-axes.

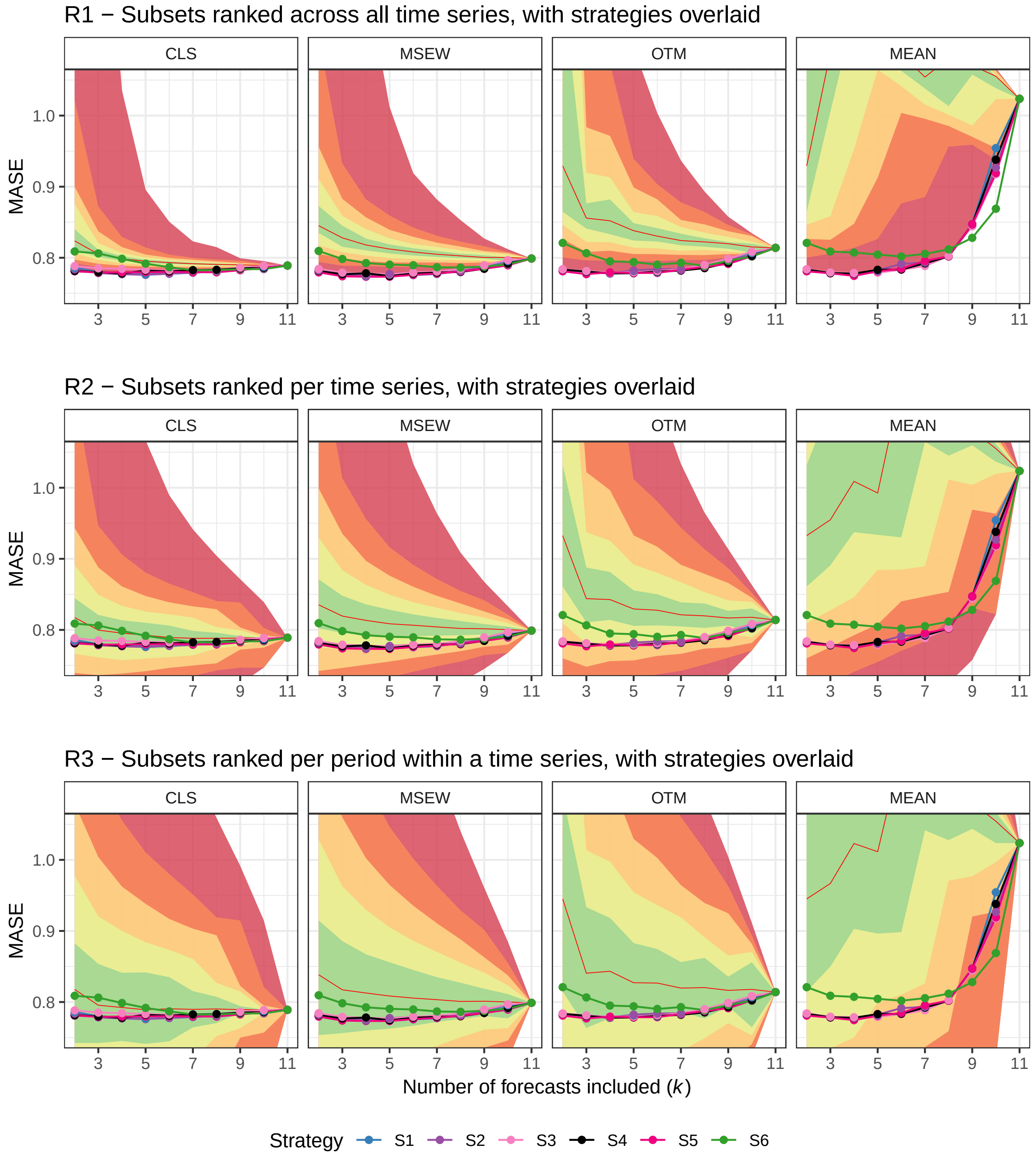

Figure 4.4: This figure shows a zoomed in version of Fig. 4.1, with input forecasts removed, and with MASE from the six selection strategies (S1, S2, S3, S4, S5 and S6), from Fig. 4.4, overlaid as dotted lines in each of R1, R2 and R3. The selection strategies, represented by the dotted lines, are the same in R1, R2 and R3; it is only the background (the colored ribbon distribution of MASE) that is replaced. See Fig. 4.1 and 4.4 for more information.

Table 4.1: MCS for strategy-based CLS forecasts.

		CLS					
		Strategy					
k		**S1**	**S2**	**S3**	**S4**	**S5**	**S6**
2	MASE	0.78599	0.78377	0.78823	0.78143	0.78100	0.80887
	Rank	4	3	5	2	1	6
	p_{MCS}	0.239^{**}	0.412^{**}	0.183^{**}	0.844^{**}	1.000^{**}	<0.001
3	MASE	0.77984	0.77844	0.78511	0.77984	0.77921	0.80615
	Rank	3.5	1	5	3.5	2	6
	p_{MCS}	0.435^{**}	1.000^{**}	0.064^{**}	0.435^{**}	0.775^{**}	<0.001
4	MASE	0.77778	0.77692	0.78440	0.77778	0.77954	0.79863
	Rank	2.5	1	5	2.5	4	6
	p_{MCS}	0.510^{**}	1.000^{**}	0.012^{*}	0.510^{**}	0.510^{**}	<0.001
5	MASE	0.77653	0.77593	0.78297	0.78195	0.77964	0.79173
	Rank	2	1	5	4	3	6
	p_{MCS}	0.405^{**}	1.000^{**}	0.028^{*}	0.028^{*}	0.246^{**}	<0.001
6	MASE	0.77750	0.77768	0.78247	0.78084	0.78017	0.78683
	Rank	1	2	5	4	3	6
	p_{MCS}	1.000^{**}	0.918^{**}	0.103^{**}	0.386^{**}	0.436^{**}	0.030^{*}
7	MASE	0.77899	0.77914	0.78207	0.78255	0.77979	0.78239
	Rank	1	2	4	6	3	5
	p_{MCS}	1.000^{**}	0.925^{**}	0.565^{**}	0.565^{**}	0.925^{**}	0.640^{**}
8	MASE	0.78079	0.78079	0.78281	0.78306	0.77954	0.78259
	Rank	2.5	2.5	5	6	1	4
	p_{MCS}	0.586^{**}	0.586^{**}	0.448^{**}	0.448^{**}	1.000^{**}	0.448^{**}
9	MASE	0.78461	0.78461	0.78613	0.78509	0.78252	0.78393
	Rank	3.5	3.5	6	5	1	2
	p_{MCS}	0.657^{**}	0.657^{**}	0.301^{**}	0.657^{**}	1.000^{**}	0.657^{**}
10	MASE	0.78718	0.78431	0.78871	0.78596	0.78595	0.78406
	Rank	5	2	6	4	3	1
	p_{MCS}	0.261^{**}	0.885^{**}	0.041^{**}	0.699^{**}	0.699^{**}	1.000^{**}
2-10	MASE	0.78102	0.78018	0.78477	0.78206	0.78082	0.79169
	Rank	3	1	5	4	2	6
	Median Rank	2.5	2	5	4	3	6
	Times in $\widehat{\mathcal{M}}^*_{75\%(\text{adj})}$	9	9	7	8	9	4
	Times in $\widehat{\mathcal{M}}^*_{95\%(\text{adj})}$	9	9	9	9	9	5

Note:
The table shows the results from nine MCS tests, computed at $k = 2, \ldots, 10$, respectively. The CLS forecasts from the six strategies (S1, S2, S3, S4, S5 and S6) are tested using the evaluation dataset (8150 observations). Results from a test should be read horizontally across the strategies. MASE, and the rank of MASE, is also available. The strategies in the Šidák family-wise adjusted $\widehat{\mathcal{M}}^*_{95\%(\text{adj})}$ and $\widehat{\mathcal{M}}^*_{75\%(\text{adj})}$ are identified by one and two asterisks, respectively. With family-wise $\alpha = 0.05$ and $\alpha = 0.25$, respectively, across the $m = 9$ tests, these corresponds to p_{MCS} greater than 0.0057 and 0.0315, respectively. Each MCS test (with loss function $|\text{ScE}_t|$) is based on a block bootstrap with 5000 samples, with a block length of 15. The last 5 rows contain summary statistics. Median Rank, as well as number of times a strategy occurs in $\widehat{\mathcal{M}}^*_{95\%(\text{adj})}$ and $\widehat{\mathcal{M}}^*_{75\%(\text{adj})}$, should be read columnwise across k.

Table 4.2: MCS for strategy-based MSEW forecasts.

		MSEW					
		Strategy					
k		**S1**	**S2**	**S3**	**S4**	**S5**	**S6**
2	MASE	0.78153	0.78213	0.78424	0.78153	0.77949	0.80950
	Rank	2.5	4	5	2.5	1	6
	p_{MCS}	0.355^{**}	0.355^{**}	0.355^{**}	0.355^{**}	1.000^{**}	<0.001
3	MASE	0.77685	0.77685	0.77912	0.77685	0.77396	0.79829
	Rank	3.5	2	5	3.5	1	6
	p_{MCS}	0.415^{**}	0.415^{**}	0.269^{**}	0.415^{**}	1.000^{**}	<0.001
4	MASE	0.77419	0.77489	0.77778	0.77816	0.77332	0.79253
	Rank	2	3	4	5	1	6
	p_{MCS}	0.665^{**}	0.577^{**}	0.269^{**}	0.077^{**}	1.000^{**}	<0.001
5	MASE	0.77634	0.77740	0.77688	0.77438	0.77332	0.79037
	Rank	3	5	4	2	1	6
	p_{MCS}	0.382^{**}	0.166^{**}	0.382^{**}	0.567^{**}	1.000^{**}	<0.001
6	MASE	0.77565	0.77911	0.77927	0.77783	0.77594	0.78955
	Rank	1	4	5	3	2	6
	p_{MCS}	1.000^{**}	0.060^{**}	0.060^{**}	0.212^{**}	0.827^{**}	<0.001
7	MASE	0.77721	0.77915	0.78052	0.77922	0.77781	0.78671
	Rank	1	3	5	4	2	6
	p_{MCS}	1.000^{**}	0.102^{**}	0.102^{**}	0.307^{**}	0.569^{**}	<0.001
8	MASE	0.78153	0.78153	0.78416	0.77997	0.78070	0.78624
	Rank	3.5	3.5	5	1	2	6
	p_{MCS}	0.259^{**}	0.259^{**}	0.013^{*}	1.000^{**}	0.329^{**}	0.003
9	MASE	0.78759	0.78759	0.78942	0.78462	0.78455	0.78764
	Rank	3.5	3.5	6	2	1	5
	p_{MCS}	0.012^{*}	0.012^{*}	0.001	0.859^{**}	1.000^{**}	0.055^{**}
10	MASE	0.79550	0.79089	0.79676	0.79180	0.78929	0.79057
	Rank	5	3	6	4	1	2
	p_{MCS}	<0.001	0.135^{**}	<0.001	0.039^{**}	1.000^{**}	0.176^{**}
2-10	MASE	0.78071	0.78106	0.78313	0.78049	0.77871	0.79238
	Rank	3	4	5	2	1	6
	Median Rank	3	3.5	5	3	1	6
	Times in $\widehat{\mathcal{M}}^*_{75\%(\text{adj})}$	7	8	6	9	9	2
	Times in $\widehat{\mathcal{M}}^*_{95\%(\text{adj})}$	8	9	7	9	9	2

Note:
The table shows the results from nine MCS tests, computed at $k = 2, \ldots, 10$, respectively. The MSEW forecasts from the six strategies (S1, S2, S3, S4, S5 and S6) are tested using the evaluation dataset (8150 observations). Results from a test should be read horizontally across the strategies. MASE, and the rank of MASE, is also available. The strategies in the Šidák family-wise adjusted $\widehat{\mathcal{M}}^*_{95\%(\text{adj})}$ and $\widehat{\mathcal{M}}^*_{75\%(\text{adj})}$ are identified by one and two asterisks, respectively. With family-wise $\alpha = 0.05$ and $\alpha = 0.25$, respectively, across the $m = 9$ tests, these corresponds to p_{MCS} greater than 0.0057 and 0.0315, respectively. Each MCS test (with loss function $|\text{ScE}_t|$) is based on a block bootstrap with 5000 samples, with a block length of 15. The last 5 rows contain summary statistics. Median Rank, as well as number of times a strategy occurs in $\widehat{\mathcal{M}}^*_{95\%(\text{adj})}$ and $\widehat{\mathcal{M}}^*_{75\%(\text{adj})}$, should be read columnwise across k.

Table 4.3: MCS for strategy-based OTM forecasts.

OTM							
		Strategy					
k		**S1**	**S2**	**S3**	**S4**	**S5**	**S6**
2	MASE	0.78342	0.78358	0.78387	0.78342	0.78095	0.82089
	Rank	2.5	4	5	2.5	1	6
	p_{MCS}	0.363**	0.363**	0.363**	0.363**	1.000**	$<$0.001
3	MASE	0.78051	0.78028	0.78125	0.78051	0.77699	0.80645
	Rank	3.5	2	5	3.5	1	6
	p_{MCS}	0.471**	0.471**	0.471**	0.471**	1.000**	$<$0.001
4	MASE	0.77794	0.77818	0.77911	0.77794	0.77945	0.79502
	Rank	1.5	3	4	1.5	5	6
	p_{MCS}	1.000**	0.911**	0.911**	1.000**	0.911**	$<$0.001
5	MASE	0.78129	0.78227	0.77962	0.77828	0.77860	0.79401
	Rank	4	5	3	1	2	6
	p_{MCS}	0.439**	0.182**	0.819**	1.000**	0.867**	$<$0.001
6	MASE	0.77953	0.78378	0.78109	0.78045	0.77872	0.79030
	Rank	2	5	4	3	1	6
	p_{MCS}	0.693**	0.039**	0.676**	0.676**	1.000**	0.001
7	MASE	0.78271	0.78408	0.78405	0.78197	0.78318	0.79290
	Rank	2	5	4	1	3	6
	p_{MCS}	0.778**	0.623**	0.778**	1.000**	0.778**	$<$0.001
8	MASE	0.78800	0.78800	0.78995	0.78543	0.78731	0.78861
	Rank	3.5	3.5	6	1	2	5
	p_{MCS}	0.479**	0.479**	0.179**	1.000**	0.479**	0.479**
9	MASE	0.79810	0.79810	0.79879	0.79280	0.79132	0.79572
	Rank	4.5	4.5	6	2	1	3
	p_{MCS}	0.003	0.003	0.003	0.017*	1.000**	0.017*
10	MASE	0.80667	0.80244	0.80843	0.80266	0.80226	0.80171
	Rank	5	3	6	4	2	1
	p_{MCS}	0.002	0.914**	$<$0.001	0.914**	0.914**	1.000**
2-10	MASE	0.78646	0.78674	0.78735	0.78483	0.78431	0.79840
	Rank	3	4	5	2	1	6
	Median Rank	3.5	4	5	2	2	6
	Times in $\widehat{\mathcal{M}}^*_{75\%(\text{adj})}$	7	8	7	8	9	2
	Times in $\widehat{\mathcal{M}}^*_{95\%(\text{adj})}$	7	8	7	9	9	3

Note:
The table shows the results from nine MCS tests, computed at $k = 2, \dots, 10$, respectively. The OTM forecasts from the six strategies (S1, S2, S3, S4, S5 and S6) are tested using the evaluation dataset (8150 observations). Results from a test should be read horizontally across the strategies. MASE, and the rank of MASE, is also available. The strategies in the Šidák family-wise adjusted $\widehat{\mathcal{M}}^*_{95\%(\text{adj})}$ and $\widehat{\mathcal{M}}^*_{75\%(\text{adj})}$ are identified by one and two asterisks, respectively. With family-wise $\alpha = 0.05$ and $\alpha = 0.25$, respectively, across the $m = 9$ tests, these corresponds to p_{MCS} greater than 0.0057 and 0.0315, respectively. Each MCS test (with loss function $|\text{ScE}_t|$) is based on a block bootstrap with 5000 samples, with a block length of 15. The last 5 rows contain summary statistics. Median Rank, as well as number of times a strategy occurs in $\widehat{\mathcal{M}}^*_{95\%(\text{adj})}$ and $\widehat{\mathcal{M}}^*_{75\%(\text{adj})}$, should be read columnwise across k.

Table 4.4: MCS for strategy-based MEAN forecasts.

		MEAN					
		Strategy					
k		**S1**	**S2**	**S3**	**S4**	**S5**	**S6**
2	MASE	0.78342	0.78358	0.78387	0.78342	0.78095	0.82089
	Rank	2.5	4	5	2.5	1	6
	p_{MCS}	0.363^{**}	0.363^{**}	0.363^{**}	0.363^{**}	1.000^{**}	<0.001
3	MASE	0.77857	0.77829	0.77936	0.77857	0.77805	0.80883
	Rank	3.5	2	5	3.5	1	6
	p_{MCS}	0.970^{**}	0.970^{**}	0.970^{**}	0.970^{**}	1.000^{**}	<0.001
4	MASE	0.77770	0.77805	0.77863	0.77770	0.77456	0.80732
	Rank	2.5	4	5	2.5	1	6
	p_{MCS}	0.153^{**}	0.153^{**}	0.153^{**}	0.153^{**}	1.000^{**}	<0.001
5	MASE	0.78162	0.78242	0.77902	0.78303	0.78084	0.80447
	Rank	3	4	1	5	2	6
	p_{MCS}	0.495^{**}	0.371^{**}	1.000^{**}	0.371^{**}	0.495^{**}	<0.001
6	MASE	0.78349	0.79133	0.78323	0.78349	0.78467	0.80186
	Rank	2.5	5	1	2.5	4	6
	p_{MCS}	0.897^{**}	<0.001	1.000^{**}	0.897^{**}	0.225^{**}	<0.001
7	MASE	0.79216	0.79331	0.78840	0.79216	0.79550	0.80544
	Rank	2.5	4	1	2.5	5	6
	p_{MCS}	0.053^{**}	0.028^{*}	1.000^{**}	0.053^{**}	0.006	<0.001
8	MASE	0.80194	0.80194	0.80264	0.80194	0.80204	0.81176
	Rank	2	2	5	2	4	6
	p_{MCS}	1.000^{**}	1.000^{**}	0.886^{**}	1.000^{**}	0.886^{**}	<0.001
9	MASE	0.84710	0.84710	0.84427	0.84710	0.84710	0.82804
	Rank	4.5	4.5	2	4.5	4.5	1
	p_{MCS}	0.002	0.002	0.002	0.002	0.002	1.000^{**}
10	MASE	0.95431	0.92674	0.91756	0.93803	0.91918	0.86897
	Rank	6	4	2	5	3	1
	p_{MCS}	<0.001	<0.001	<0.001	<0.001	<0.001	1.000^{**}
2-10	MASE	0.81115	0.80919	0.80633	0.80949	0.80699	0.81751
	Rank	5	3	1	4	2	6
	Median Rank	2.5	4	2	2.5	3	6
	Times in $\widehat{\mathcal{M}}^*_{75\%(\text{adj})}$	7	5	7	7	6	2
	Times in $\widehat{\mathcal{M}}^*_{95\%(\text{adj})}$	7	6	7	7	6	2

Note:
The table shows the results from nine MCS tests, computed at $k = 2, \dots, 10$, respectively. The MEAN forecasts from the six strategies (S1, S2, S3, S4, S5 and S6) are tested using the evaluation dataset (8150 observations). Results from a test should be read horizontally across the strategies. MASE, and the rank of MASE, is also available. The strategies in the Šidák family-wise adjusted $\widehat{\mathcal{M}}^*_{95\%(\text{adj})}$ and $\widehat{\mathcal{M}}^*_{75\%(\text{adj})}$ are identified by one and two asterisks, respectively. With family-wise $\alpha = 0.05$ and $\alpha = 0.25$, respectively, across the $m = 9$ tests, these corresponds to p_{MCS} greater than 0.0057 and 0.0315, respectively. Each MCS test (with loss function $|\text{ScE}_t|$) is based on a block bootstrap with 5000 samples, with a block length of 15. The last 5 rows contain summary statistics. Median Rank, as well as number of times a strategy occurs in $\widehat{\mathcal{M}}^*_{95\%(\text{adj})}$ and $\widehat{\mathcal{M}}^*_{75\%(\text{adj})}$, should be read columnwise across k.

4.3 Objective III - Forecast accuracy between values of k

As stated in the Introduction section, Objective III is about providing more supporting guidance when it comes to selecting number of input forecasts, k, to include in a forecast combination. Only forecasts from the earlier mentioned strategies S3 and S5 are utilized for this purpose, so the results are mainly valid for these rank-based strategies. Results from MCS tests where forecasts from S5 are compared between $k = 2, \dots, 11$ as well as with all the eleven individual input forecasts, are available in Table 4.5. The table contains results from three separate MCS tests, one individually for each of the CLS, MSEW and OTM forecasts. The MEAN forecasts are not included since S5 is not the recommended strategy for this method. A family-wise Šidák correction with $m = 3$ tests is made. The MCS test with CLS rejects all eleven input forecasts from the model confidence set with a $p_{\text{MCS}} < 0.001$. The set $\widehat{\mathcal{M}}^*_{75\%\text{(adj)}}$ contains S5 forecasts with $k = 2, \dots, 9$ while $k = 10, 11$ are not rejected from the $\widehat{\mathcal{M}}^*_{95\%\text{(adj)}}$ set. The results from the MCS test with MSEW also rejects all input forecasts from the model confidence set. The set $\widehat{\mathcal{M}}^*_{75\%\text{(adj)}}$ only includes S5 forecasts at $k = 3, 4, 5$ while $k = 2, 6, 7$ are also included in the model confidence set with a 95% confidence level, although $k = 2, 7$ are almost rejected. The S5 forecasts at $k > 7$ are rejected. Results for OTM are similar. Here, S5 forecasts at $2 \leq k \leq 6$ are members of $\widehat{\mathcal{M}}^*_{75\%\text{(adj)}}$ while $k = 7$ is almost included. Also note that the S5-based OTM forecasts at $k = 11$ perform worse than AUTOARIMA and ETS, in the sample. However, these differences in accuracy, in the population, are not tested here.

In Table 4.6 corresponding MCS tests but for strategy S3 are reported. An MCS test for S3-based MEAN forecasts is also reported. The input forecasts are rejected from all model confidence sets computed. However, note that S3-based MEAN forecasts with $k > 7$ perform worse than some input forecasts in the sample as seen by MASE. All S3-based CLS forecasts are members of $\widehat{\mathcal{M}}^*_{95\%\text{(adj)}}$, and where $k = 7$ contains the lowest MASE in the evaluation sample. Again, the results from MSEW and OTM are similar. Forecasts with $k > 7$ are rejected from $\widehat{\mathcal{M}}^*_{95\%\text{(adj)}}$ for both; for MSEW $k = 2$ is rejected as well. For S3-based MEAN forecasts, only $k = 3, 4, 5$ are included in $\widehat{\mathcal{M}}^*_{95\%\text{(adj)}}$.

Table 4.5: MCS for S5-based forecasts and input forecasts.

	CLS			**MSEW**			**OTM**		
	MASE	**Rank**	p_{MCS}	**MASE**	**Rank**	p_{MCS}	**MASE**	**Rank**	p_{MCS}
S5 with $k =$									
2	0.78100	7	0.979^{**}	0.77949	6	0.019^{*}	0.78095	5	0.453^{**}
3	0.77921	1	1.000^{**}	0.77396	3	0.901^{**}	0.77699	1	1.000^{**}
4	0.77954	3	1.000^{**}	0.77332	2	0.996^{**}	0.77945	4	0.565^{**}
5	0.77964	4	1.000^{**}	0.77332	1	1.000^{**}	0.77860	2	0.800^{**}
6	0.78017	6	0.999^{**}	0.77594	4	0.091^{*}	0.77872	3	0.800^{**}
7	0.77979	5	1.000^{**}	0.77781	5	0.019^{*}	0.78318	6	0.074^{*}
8	0.77954	2	1.000^{**}	0.78070	7	<0.001	0.78731	7	<0.001
9	0.78252	8	0.428^{**}	0.78455	8	<0.001	0.79132	8	<0.001
10	0.78595	9	0.064^{*}	0.78929	9	<0.001	0.80226	9	<0.001
11	0.78905	10	0.023^{*}	0.79895	10	<0.001	0.81396	13	<0.001
Input forecast									
AUTOARIMA	0.81012	12	<0.001		12	<0.001		11	<0.001
ETS	0.80229	11	<0.001		11	<0.001		10	<0.001
FFNN	0.96745	17	<0.001		17	<0.001		17	<0.001
MAPA	0.89546	15	<0.001		15	<0.001		15	<0.001
PROPHET	1.71347	19	<0.001		19	<0.001		19	<0.001
PSF	2.86437	20	<0.001		20	<0.001		20	<0.001
SES	0.95840	16	<0.001		16	<0.001		16	<0.001
TBATS	0.81096	13	<0.001		13	<0.001		12	<0.001
THETA	0.83642	14	<0.001		14	<0.001		14	<0.001
THIEF	1.01831	18	<0.001		18	<0.001		18	<0.001
TSLM	3.40033	21	<0.001		21	<0.001		21	<0.001

Note:
The table shows the results from three MCS tests related to strategy S5; one for each of CLS, MSEW and OTM. Results from a MCS test that compares S5-based CLS forecasts across $k = 2, \ldots, 11$ as well as with the eleven individual input forecasts are reported, i.e. 21 models are included at start of the MCS procedure. MASE, and the rank of MASE, is also available. Same test but with MSEW and OTM, respectively, is also reported. Results from a test should be read vertically. The evaluation dataset contains 8150 observations. The models in the Šidák family-wise adjusted $\widehat{\mathcal{M}}^{*}_{95\%(\text{adj})}$ and $\widehat{\mathcal{M}}^{*}_{75\%(\text{adj})}$ are identified by one and two asterisks, respectively. With family-wise $\alpha = 0.05$ and $\alpha = 0.25$, respectively, across the $m = 3$ tests, these corresponds to p_{MCS} greater than 0.017 and 0.091, respectively. Each MCS test (with loss function $|\text{ScE}_t|$) is based on a block bootstrap with 5000 samples, with a block length of 15. MASE scores from the input forecasts are only reported once.

Table 4.6: MCS for S3-based forecasts and input forecasts.

	CLS			**MSEW**			**OTM**			**MEAN**		
	MASE	**Rank**	p_{MCS}	**MASE**	**Rank**	p_{MCS}	**MASE**	**Rank**	p_{MCS}	**MASE**	**Rank**	p_{MCS}
S3 with $k =$												
2	0.78823	8	0.068*	0.78424	7	0.004	0.78387	5	0.114**	0.78387	5	0.008
3	0.78511	6	0.581**	0.77912	3	0.318**	0.78125	4	0.374**	0.77936	3	0.825**
4	0.78440	5	0.581**	0.77778	2	0.346**	0.77911	1	1.000**	0.77863	1	1.000**
5	0.78297	4	0.854**	0.77688	1	1.000**	0.77962	2	0.648**	0.77902	2	0.825**
6	0.78247	2	0.854**	0.77927	4	0.053*	0.78109	3	0.428**	0.78323	4	0.008
7	0.78207	1	1.000**	0.78052	5	0.053*	0.78405	6	0.082**	0.78840	6	<0.001
8	0.78281	3	0.854**	0.78416	6	<0.001	0.78995	7	<0.001	0.80264	8	<0.001
9	0.78613	7	0.068*	0.78942	8	<0.001	0.79879	8	<0.001	0.84427	12	<0.001
10	0.78871	9	0.023*	0.79676	9	<0.001	0.80843	10	<0.001	0.91756	14	<0.001
11	0.78905	10	0.068*	0.79895	10	<0.001	0.81396	13	<0.001	1.02365	18	<0.001
Input forecast												
AUTOARIMA	0.81012	12	<0.001		12	<0.001		11	<0.001		9	<0.001
ETS	0.80229	11	<0.001		11	<0.001		9	<0.001		7	<0.001
FFNN	0.96745	17	<0.001		17	<0.001		17	<0.001		16	<0.001
MAPA	0.89546	15	<0.001		15	<0.001		15	<0.001		13	<0.001
PROPHET	1.71347	19	<0.001		19	<0.001		19	<0.001		19	<0.001
PSF	2.86437	20	<0.001		20	<0.001		20	<0.001		20	<0.001
SES	0.95840	16	<0.001		16	<0.001		16	<0.001		15	<0.001
TBATS	0.81096	13	<0.001		13	<0.001		12	<0.001		10	<0.001
THETA	0.83642	14	<0.001		14	<0.001		14	<0.001		11	<0.001
THIEF	1.01831	18	<0.001		18	<0.001		18	<0.001		17	<0.001
TSLM	3.40033	21	<0.001		21	<0.001		21	<0.001		21	<0.001

Note:

The table shows the results from four MCS tests related to strategy S3; one for each of CLS, MSEW, OTM and MEAN. Results from a MCS test that compares S3-based CLS forecasts across $k = 2, \ldots, 11$ as well as with the eleven individual input forecasts are reported, i.e. 21 models are included at start of the MCS procedure. MASE, and the rank of MASE, is also available. Same test but with MSEW, OTM and MEAN, respectively, are also reported. Results from a test should be read vertically. The evaluation dataset contains 8150 observations. The models in the Šidák family-wise adjusted $\widehat{\mathcal{M}}^{*}_{95\%(\text{adj})}$ and $\widehat{\mathcal{M}}^{*}_{75\%(\text{adj})}$ are identified by one and two asterisks, respectively. With family-wise $\alpha = 0.05$ and $\alpha = 0.25$, respectively, across the $m = 4$ tests, these corresponds to p_{MCS} greater than 0.013 and 0.069, respectively. Each MCS test (with loss function $|\text{ScE}_t|$) is based on a block bootstrap with 5000 samples, with a block length of 15. MASE scores from the input forecasts are only reported once.

5 Discussion

The centerpiece of this thesis is the investigation of the variability in forecast accuracy for four methods of combining time series forecasts. With a fixed set of eleven potential forecasting models to include in a combination method, the forecast accuracy over all distinct subsets of them has been studied. A set of 8150 observations from 815 time series from the M3-Competition dataset is utilized to draw conclusions. From the reported results, it is clear that it is often a matter of balancing risk and reward when it comes to select the number of forecasts to include in a combination method. The spread between the best and worst forecast accuracy from the different subsets of input forecasts is greatest when number of forecasts included is small, and then decreases as number of forecasts included increases. This pattern goes hand in hand with the behavior of the variance of the mean forecast accuracy as well. This applies to all four combination methods analyzed. The conclusion also confirms the findings in Makridakis and Winkler (1983) and Hibon and Evgeniou (2005) where the MEAN combination method exclusively was studied, but here this is seen for the CLS, MSEW and OTM methods as well. Since the lowest possible mean forecast error, as measured by MASE, in the studied sample often is achieved when $k = 2$, at the same time as the highest forecast error also is possible here, it is a risky move to just combine two forecasts. You might get a good forecast accuracy if you select the 'right' subset of input forecasts, but you might as well get a much worse if you select the 'wrong' subset. The MEAN forecast method is especially prone to this reality since the method is not using any training data to downweight poor forecasts that are included in a set of input forecasts. The much higher variance in mean forecast accuracy for MEAN, compared with the other three methods, is most likely related to this. This behavior is expected. The effect is probably exaggerated in this study because there are three input forecast models that on average perform much worse than the rest. However, it is this situation, when poor performing forecasting models are included in the mix, that is also important to understand better. As explained in Aiolfi et al. (2011), designing the 'universe' of input forecasting models is often ignored in the literature, and conclusions about the performance of combination methods from these studies might be misrepresented in cases where this situation happens. Any analyst who uses the MEAN method to combine forecasts, because many studies recommend an equal weighted combination as described in Clemen (1989), should therefore be aware of this fact. The CLS, MSEW and, to some extent, OTM makes the choice of the selected input forecasts somewhat less important, as seen by the lower variance of MASE.

For all four combination methods in the evaluation sample, the mean of MASE, across all distinct subsets, decreases as k increases. It could be tempting to suggest to always combine as many forecasts as possible. However, the price that you pay is that you sacrifice the potential gain in forecast accuracy that is available when fewer forecasts are selected. The advantage is that you do not need to carefully pick an optimal subset since you do not risk as much in lost accuracy.

The simple rank-based selection strategies suggested in this study are successful in lowering the mean forecasting error when a subset of available forecasts is selected; as seen by the MCS tests for strategies S3 and S5. Here, a significant gain in accuracy is seen compared to the situation where all forecasts are included, if the pruning is not too aggressive and only leaves two forecasts to combine. Forecast combinations from the selection strategies are also significantly better than forecasts from all the individual forecasting models, respectively.

To give a general advice about how many forecasts that should be removed is difficult. It will likely depend on the situation in front of you. Some hints are however given. With the S5 and S3 based CLS forecasts, the best result is attained when 2 to 9 respectively 3 to 8 forecasts are included. With MSEW the corresponding figures are 3 to 5 for both. With OTM then k is 2 to 6 and 2 to 7, respectively. For the S3-based MEAN forecasts the best result is achieved when number of forecasts included varies between 3 and 5. This suggests that least pruning is needed for CLS while a more aggressive removal of forecasts is recommended for the other three methods; especially for MEAN as already discussed. It should also be noted that the importance of the pruning is most likely less beneficial when the ‘universe’ of input forecasts does not contain bad apples to begin with. As a comparison, the argument made by Armstrong (2001), who suggest that five or more forecasts should be combined, could therefore be somewhat contested. To combine more forecasts is not always a better idea.

Five of the six selection strategies analyzed in the study perform similarly. If a large set of time series is available that mimics a studied time series, then this set is useful to select a subset of forecasts. Strategies S1, S2, S4 and the mentioned S5 do this successfully. The more advance S5, where all distinct subsets of forecast combinations are ranked per domain in the large set of time series, performs best for the combination methods that utilize training data. However, the computational cost of this strategy cannot be disregarded. If this is not motivated then strategy S3, where only the input forecasts are ranked based on past periods for the time series in question, almost performs as good and at a much lower cost.

Some limitations need to be addressed. All results from the evaluation are based on using the, mean, absolute scaled error as a loss function. If the evaluation was based on a squared scaled error, or a symmetric absolute error, the results might have been slightly different. The choice of a loss function is a common problem in many studies. Another thing to address is the selection of the eleven input forecasting models to include in the study. The results, like the performance of the strategy-based forecasts, might be different if another set of forecasting models was analyzed. However, the size of the study might mitigate some of these issues since the strategies are exposed to many different situations. The utilized subset of the M3-Competition dataset, that contains only monthly time series, is another restriction from making the conclusions generally viable. The assumption about stationarity of the relative loss between models in the MCS tests is difficult to test in reality since many comparisons between models are made. One thing that complicates this further is also the design of the evaluation which contains many time series, each with ten forecasts. This makes it more difficult to test but at the same time it also limits the potential negative impact since the stationarity issue is not an issue between time series, only within. Utilizing a fixed rolling window forecast scheme for the input forecasts models would help here, but it comes at a computational cost. The evaluation design here, where a recursive expanding evaluation scheme is used for the input forecasts together with a fixed rolling window forecast scheme for the combination methods, is from an MCS test standpoint probably not too problematic. Hansen et al. (2011) argues that recursive expanding and fixed rolling window forecast schemes, respectively, gives almost identical results in the tests. Another limitation of the study is the design of the static strategies. For example, the domain-based strategies utilize the manmade and predefined ‘domain’ label from the M3-Competition dataset. This might not be optimal because the similarities between the time series within a domain may be irrelevant. Furthermore, a relevant label, to connect time series that belongs together, might not be available at all in some situations.

To summarize, some advice is given. If the performance of the input forecast models is unknown, then use the CLS or MSEW methods to combine the forecasts. This will likely avoid many of the worst situations. If the performance of some of the input forecasts are lagging the rest, then it is a good idea to not include them in a forecast combination; especially if the MEAN or OTM methods are used. Selecting how many to include is a case-by-case choice and is not primarily in focus here. If there exists a large dataset with similar time series comparable to the one that is being studied, then using this data together with one of the recommended selection strategies might improve the forecast accuracy of a combination method. If this is not available, or too costly, then a dynamic selection strategy, where the input forecasts are ranked, is recommended.

5.1 Future research

Comments on interesting future research topics are also made here. A direct thought is to try to replicate the results using a different dataset and with a new set of input forecast models. The M3-Competition dataset is starting to get somewhat overused. A simulation study, with controlled conditions, would also be interesting to provide more in-depth knowledge. An investigation where common factors between subsets with the highest accuracy are studied is also of interest in order to understand more. Another intriguing area is related to the static selection strategies. In this study, just the manmade ‘domain’ label was used for mapping the relationship between a static ranking of input forecasts and a time series. This relationship could instead be analyzed through a similarity measure, like for example a clustering technique. Another exciting research area is related to testing whether the average across several subsets improves the forecast accuracy. Strategies S4, S5 and S6, respectively, computes a forecast combination based on only the single subset of input forecasts that ranks first. However, if forecast combinations are computed from each of the subsets that are ranked in, for example, the top 20 percent, then a mean across all of them can be computed. This mean of multiple forecast combinations can then be utilized as a forecast. Further, development and evaluation of prediction intervals for the forecast combination methods are also interesting topics. Since selecting number of forecasts to include in a forecast combination is not directly evaluated, this is also a very compelling next step. If the pieces presented here are added together in a different way, then it would be fascinating to see how well the MCS test by itself performs at selecting the appropriate number of forecasts to include in a forecast combination.

A Appendix

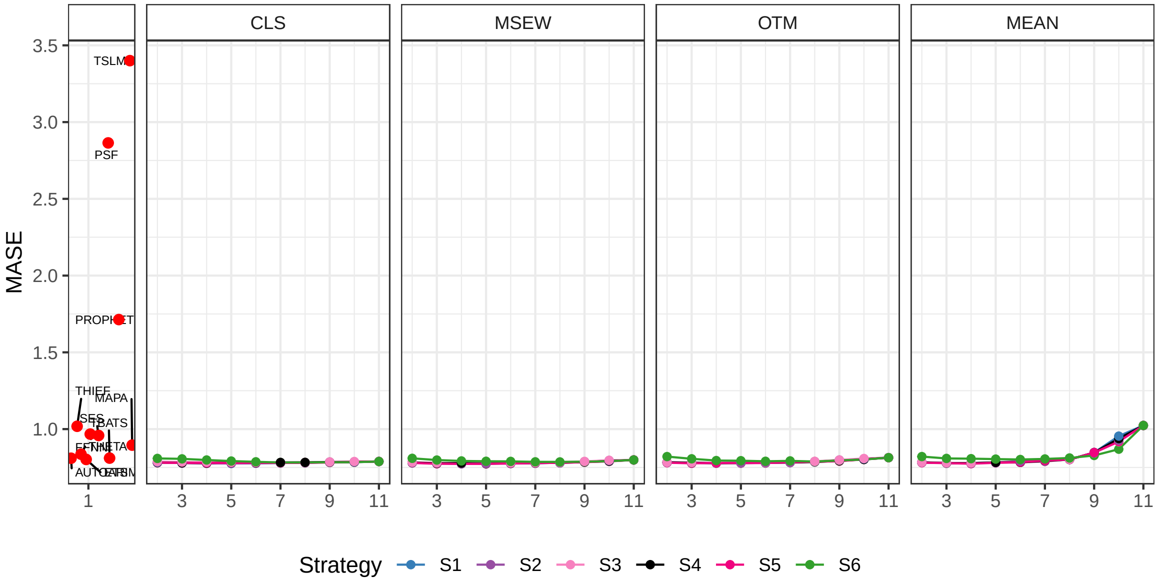

Figure A.1: This figure shows a zoomed-out version of the upper rows of plots from Fig. 4.3. The three worst input forecasts (PROPHET, MASE = 1.71; PSF, MASE = 2.86; TSLM, MASE = 3.40) are here not removed. See Fig. 4.3 for more information.

References

Abdi, H. (2007). *The Bonferonni and Šidák Corrections for Multiple Comparisons. In: Encyclopedia of Measurement and Statistics.* Sage, Thousand Oaks (CA).

Aiolfi, M., Capistrán, C., and Timmermann, A. (2011). *Forecast Combinations. In: The Oxford Handbook of Economic Forecasting.* Oxford University Press, Oxford.

Aksu, C. and Gunter, S. I. (1992). An empirical analysis of the accuracy of SA, OLS, ERLS and NRLS combination forecasts. *International Journal of Forecasting*, 8:27–43.

Armstrong, J. S. (1985). *Long-Range Forecasting: From Crystal Ball to Computer.* John Wiley & Sons, New York, 2nd edition.

Armstrong, J. S. (2001). *Combining Forecasts. In: Principles of Forecasting.* Springer, Boston.

Bates, J. M. and Granger, C. W. J. (1969). The Combination of Forecasts. *Operations Research Quarterly*, (20):451–469.

Bokde, N., Asencio-Cortes, G., and Martinez-Alvarez, F. (2017). *PSF: Forecasting of Univariate Time Series Using the Pattern Sequence-Based Forecasting (PSF) Algorithm.* R package version 0.4, https://CRAN.R-project.org/package=PSF.

Clemen, R. T. (1989). Combining forecasts: A review and annotated bibliography. *International Journal of Forecasting*, 5(4):559–583.

Diebold, F. X. (2015). Comparing Predictive Accuracy, Twenty Years Later: A Personal Perspective on the Use and Abuse of Diebold–Mariano Tests. *Journal of Business & Economic Statistics*, 33(1):1–1.

Diebold, F. X. and Mariano, R. S. (1995). Comparing Predictive Accuracy. *Journal of Business & Economic Statistics*, 13(3):55.

Elliott, G., Gargano, A., and Timmermann, A. (2013). Complete subset regressions. *Journal of Econometrics*, 177(2):357–373.

Elliott, G. and Timmermann, A. (2016). Forecasting in Economics and Finance. *Annual Review of Economics*, 8(1):81–110.

Genre, V., Kenny, G., Meyler, A., and Timmermann, A. (2013). Combining expert forecasts: Can anything beat the simple average? *International Journal of Forecasting*, 29(1):108–121.

Ghalanos, A. (2019). *rugarch: Univariate GARCH models.* R package version 1.4-1, https://CRAN.R-project.org/package=rugarch.

Gunter, S. I. (1992). Nonnegativity restricted least squares combinations. *International Journal of Forecasting*, 8:45–59.

Hansen, P. R., Lunde, A., and Nason, J. M. (2011). The Model Confidence Set. *Econometrica*, 79(2):453–497.

Hansen, P. R., Lunde, A., and Nason, J. M. (2014). *Corrigendum to "The Model Confidence Set".* Retrieved February 1, 2019, from https://docs.google.com/viewer?a=v&pid=sites&srcid= ZGVmYXVsdGRvbWFpbnxwZXRlcnJlaW5oYXJkaGFuc2VufGd4OjY1ZjQxODUxYmM1ZThhOWE.

Hibon, M. and Evgeniou, T. (2005). To combine or not to combine: Selecting among forecasts and their combinations. *International Journal of Forecasting*, 21:15–24.

Hyndman, R. J. (2017). *forecast: Forecasting functions for time series and linear models.* R package version 8.1, https://CRAN.R-project.org/package=forecast.

Hyndman, R. J., Akram, M., Bergmeir, C., and O'Hara-Wild, M. (2017). *Mcomp: Data from the M-Competitions.* R package version 2.6, https://CRAN.R-project.org/package=Mcomp.

Hyndman, R. J. and Koehler, A. B. (2006). Another look at measures of forecast accuracy. *International Journal of Forecasting*, 22(4):679–688.

Hyndman, R. J. and Kourentzes, N. (2018). *thief: Temporal HIErarchical Forecasting.* R package version 0.3, https://CRAN.R-project.org/package=thief.

Jose, V. R. R. and Winkler, R. L. (2008). Simple robust averages of forecasts: Some empirical results. *International Journal of Forecasting*, 24(1):163–169.

Kourentzes, N. and Petropoulos, F. (2017). *MAPA: Multiple Aggregation Prediction Algorithm.* R package version 2.0.2, https://CRAN.R-project.org/package=MAPA.

Makridakis, S. and Hibon, M. (2000). The M3-Competition: results, conclusions and implications. *International Journal of Forecasting*, 16(4):451–476.

Makridakis, S. and Winkler, R. L. (1983). Averages of Forecasts: Some Empirical Results. *Management Science*, 29(9):987–996.

Ord, K. (2001). Commentaries on the M3-Competition: An introduction, some comments and a scorecard. *International Journal of Forecasting*, 17(4):537–584.

R Core Team (2018). *R: A Language and Environment for Statistical Computing.* R Foundation for Statistical Computing, Vienna, Austria. https://www.R-project.org/.

Svensson, M. (2018). An Evaluation of Methods for Combining Univariate Time Series Forecasts. (Bachelor's Thesis, Lund University, Sweden). Retrieved from https://lup.lub.lu.se/student-papers/search/publication/8939148.

Taylor, S. J. and Letham, B. (2017). *prophet: Automatic Forecasting Procedure.* R package version 0.2, https://CRAN.R-project.org/package=prophet.

Tofallis, C. (2015). A better measure of relative prediction accuracy for model selection and model estimation. *Journal of the Operational Research Society*, 66(8):1352–1362.

White, H. (2000). A Reality Check for Data Snooping. *Econometrica*, 68(5):1097–1126.