TI 2019-025/III
Tinbergen Institute Discussion Paper

Forecast Density Combinations with Dynamic Learning for Large Data Sets in Economics and Finance

Roberto Casarin[1]
Stefano Grassi[2]
Francesco Ravazzollo[3]
Herman K. van Dijk[4]

[1] University Ca' Foscari of Venice
[2] University of Rome `Tor Vergata'
[3] Free University of Bozen-Bolzano, CAMP, BI Norwegian Business School
[4] Erasmus University Rotterdam, Norges Bank, Tinbergen Institute

Forecast Density Combinations with Dynamic Learning for Large Data Sets in Economics and Finance*

Roberto Casarin[†] Stefano Grassi[‡]
Francesco Ravazzolo[§] Herman K. van Dijk[¶]

[†]University Ca' Foscari of Venice
[‡]University of Rome 'Tor Vergata'
[§]Free University of Bozen-Bolzano and CAMP, BI Norwegian Business School
[¶]Erasmus University Rotterdam, Norges Bank and Tinbergen Institute

March, 2019

Abstract

A flexible forecast density combination approach is introduced that can deal with large data sets. It extends the mixture of experts approach by allowing for model set incompleteness and dynamic learning of combination weights. A dimension reduction step is introduced using a sequential clustering mechanism that allocates the large set of forecast densities into a small number of subsets and the combination weights of the large set of densities are modelled as a dynamic factor model with a number of factors equal to the number of subsets. The forecast density combination is represented as a large finite mixture in nonlinear state space form. An efficient simulation-based Bayesian inferential procedure is proposed using parallel sequential clustering and filtering, implemented on graphics processing units. The approach is applied to track the Standard & Poor 500 index combining more than 7000 forecast densities based on 1856 US individual stocks that are are clustered in a relatively small subset. Substantial forecast and economic gains are obtained, in particular, in the tails using Value-at-Risk. Using a large macroeconomic data set of 142 series, similar forecast gains, including probabilities of recession, are obtained from multivariate forecast density combinations of US real GDP, Inflation, Treasury Bill yield and Employment. Evidence obtained on the dynamic patterns in the financial as well as macroeconomic clusters provide valuable signals useful for improved modelling and more effective economic and financial policies.

*The present paper should not be reported as representing the views of Norges Bank. The views expressed are those of the authors and do not necessarily reflect those of Norges Bank. This paper is a substantially revised and extended version from an earlier paper by the same authors, see Casarin et al. (2017). The authors are indebted to John Geweke, Lennart Hoogerheide and Frank Schorfheide for helpful comments.

1 Introduction

Forecasting with large sets of data is a topic of substantial interest to academic researchers as well as to professional and applied forecasters. It has been studied in several papers (e.g., see Stock and Watson, 1999, 2002, 2005, 2014, and Bańbura et al., 2010). The recent fast growth in (real-time) big data allows researchers to forecast variables of interest more accurately (e.g., see Choi and Varian, 2012; Varian, 2014; Varian and Scott, 2014; Einav and Levin, 2014). Stock and Watson (2005, 2014), Bańbura et al. (2010) and Koop and Korobilis (2013) suggest that there are also potential gains from forecasting using a large set of forecasts.

However, forecasting with large data sets, many forecasts and high-dimensional models requires new modelling strategies, efficient inference methods and extra computing power possibly resulting from parallel computing. We refer to Granger (1998) for an early discussion of these issues.

We propose a flexible parametric forecast density combination approach with dynamic learning that can deal with large data sets. It extends Billio et al. (2013) and McAlinn and West (2018) in several directions.

In terms of methodology we introduce three innovations. First, we use the mixture of experts and/or smoothly mixing regression approaches (Jacobs et al., 1991, Jordan and Jacobs, 1994, Jordan and Xu, 1995, Peng et al., 1996, Wood et al., 2002, Geweke and Keane, 2007, Villani et al., 2009, Norets, 2010) and extend these by allowing the combination weights to be dependent between models as well as to learn over time. Learning about model set incompleteness is also specified. In this context a diagnostic analysis is presented to signal

particular types of missing information.

Second, a dimension reduction step is introduced using a sequential clustering mechanism that allocates the large set of forecast densities into a small number of mutually exclusive subsets making use of such time-varying density features as past forecast accuracy, volatility and tail behaviour. The dimension reduction further involves modelling the combination weights of the large set of densities as a dynamic factor model with a number of factors equal to the number of subsets and these factors learn from past forecasting performance. The combination weights are mapped to the unit interval and interpreted as a convex set of probabilistic weights used for the construction of the large finite mixture of combination densities. Our approach contributes to the time series literature on a bounded domain, see, e.g., Aitchinson and Shen (1980) and Aitchinson (1982), and applies it to macroeconomic and finance problems extending the intuition in Stock and Watson (2014).

Third, an efficient simulation-based Bayesian inferential procedure is proposed. Given that the model can be represented as a nonlinear state space model where the measurement equation consists of a large finite mixture, parallel clustering and parallel sequential Monte Carlo filters are used for efficient numerical evaluation. Here, we follow the recent trend of using graphics processing units (GPU) for general, non-graphics, applications: the so-called general-purpose computing on GPU (GPGPU).

Using large data sets, the proposed approach is applied to two well-known problems in economics and finance. In the first example we use more than 7000 forecast densities based on 1856 US individual stock return series and four clusters to construct a combined forecast density of a replication of S&P 500

returns over the sample 2007-2009 and estimate several features of this density. We emphasise that our method allows for a time-varying composition of the four clusters letting individual stocks to switch across them or eventually exit the model set, for example, after a default as in the Lehman Brothers case. Compared to the no-forecast ability benchmark and forecasts from individual models estimated on the aggregate index, we find substantial accuracy gains in forecasting means, volatilities and tail events, in particular, with respect to the economic value of such events like Value-at-Risk. The observed dynamic patterns in the cluster-based weights provide valuable signals for improved economic and financial modelling and policy analysis.

In the macroeconomic example, we consider the extended Stock and Watson (2005) dataset, which includes 142 series sampled at a quarterly frequency from 1959Q1 to 2011Q2. Assuming the existence of 5-7 clusters, we identify two clusters related to real activities; one cluster related to prices; and one cluster related to financial variables. The other clusters contain the remaining series. As a result we find substantial gains in point and joint density forecasts of US real GDP, GDP deflator, Treasury Bill yield and Employment over the last 25 years for all horizons from one-quarter ahead to five-quarters ahead. The highest accuracy is achieved when the four series are forecasted simultaneously using our combination schemes with cluster weights based on log score learning. A dominant cluster does not exist but we note that the cluster that includes Exports, Imports and GDP deflator receives a relatively large weight. Using the complete forecast densities evidence is obtained on the probability of recession over time. Diagnostic analysis concerning model set incompleteness provides valuable signals that additional gains may be obtained with a more detailed

cluster grouping and different performance scoring rules for weights associated with models inside a cluster. This is left as a topic for further research.

The contents of this paper is structured as follows. Section 2 provides details of the methodological contributions of our approach. Section 3 contains novel empirical applications using a large set of US stocks and the Stock and Watson (2005) macroeconomic data set. Section 4 presents conclusions and suggestions for further research. The Supplementary Material contains details on a practical user guide, and more on data, derivations and results.

2 Forecast density combinations with model set incompleteness and dynamic learning for large data sets

Basic practice in macroeconomic and financial forecasting is to make use of a weighted combination of forecasts from many sources, say experts, models and/or large micro-data sets. More formally, let y_t be the variable of interest and assume that some form of forecast values $\tilde{y}_{1t}, \ldots, \tilde{y}_{nt}$ is available with a set of fixed weights $w_{1t}, \ldots, w_{nt}$. Basic practice is to make use of the linear combination

$$w_{1t}\tilde{y}_{1t} + \ldots + w_{nt}\tilde{y}_{nt} \tag{1}$$

and to assume it is a good forecast approximation to the variable of interest y_t. A major purpose of academic and professional forecasting is to give this practice a formal probabilistic foundation in order to quantify the uncertainty of such forecast density features as means, volatilities and tail behaviour. The literature on this topic is abundant, some basic references that are related to our approach

are: Billio et al. (2013), Aastveit et al. (2018), McAlinn and West (2018) and for a general survey on the field of forecast combinations we refer to Aastveit et al. (2019).

In this paper we give the practice, specified in equation (1), a stochastic interpretation using mixtures. Let $\tilde{\boldsymbol{y}}_t = (\tilde{y}_{1t}, \ldots, \tilde{y}_{nt})'$ denote the set of forecasts from n different models, we assume the forecast probability of the variable of interest y_t given $\tilde{\boldsymbol{y}}_t$, is a discrete mixture of conditional probabilities of y_t given $\tilde{y}_{it}$ coming from n different models. The mixture weights $\boldsymbol{w}_t = (w_{1t}, \ldots, w_{nt})'$ form a random partition of the unit interval and are now interpreted as probabilities. We specify such a probability model, in terms of densities, as

$$f(y_t|\tilde{\boldsymbol{y}}_t) = \sum_{i=1}^{n} w_{it} f(y_t|\tilde{y}_{it}). \tag{2}$$

We define $f(y_t|\tilde{\boldsymbol{y}}_t)$ as the *fundamental combination density*, see also Aastveit et al. (2019). The most simple form of this would be a degenerate one with fixed weights and a point mass at $\tilde{y}_{it}$ instead of a density $f(\cdot|\tilde{y}_{it})$. The purpose of this section is to make the approach operational to financial and macroeconomic models allowing for dynamic learning about mixture weights and model set incompleteness using large data sets.

2.1 Mixtures with model set incompleteness

Let the forecast densities from the n models be denoted as $f(\tilde{y}_{it}|\mathfrak{I}_{it}), i = 1, \ldots, n$, where $\mathfrak{I}_{it}$ is the information set of model i available at time $t-1$. We complement the theoretical analysis using as running example a case from finance where we consider four models: a Normal GARCH(1,1) model with a small and a large

variance and a $t-$GARCH(1,1) model with low and high degrees of freedom. As data we consider 1856 financial series with the aim to construct a combined forecast density of a replication of the S&P500 index and study its features like location, density shape and tail behaviour. For convenience, we do the analysis for one variable of interest but we emphasise that in the empirical analysis we also make use of a second case study which refers to a macroeconomic model with four joint variables of interest.

Given the combination model of equation (2) and the forecast densities from the n models, one can specify the marginal forecast density of y_t as a discrete/continuous mixture,

$$f(y_t|\mathfrak{I}_t) = \sum_{i=1}^{n} w_{it} \int f(y_t|\tilde{y}_{it}) f(\tilde{y}_{it}|\mathfrak{I}_{it}) d\tilde{y}_{it} \tag{3}$$

where $\mathfrak{I}_t$ is the joint set of information on all models. The numerical evaluation of (3) is relatively simple in case the forecast densities $f(\tilde{y}_{it}|\mathfrak{I}_{it})$ of the different models are known (say Normal GARCH(1,1) and Student $t-$GARCH(1,1)) and further the combination density $f(y_t|\tilde{y}_{it})$ and the weight density are normal. Using some well known MCMC method, one can generate forecast draws from the n different models which are inserted in the combination model. The densities are then combined by using draws from a normal weight density.

We make this approach operational to more realistic environments in finance and macroeconomics. A first step is to introduce time-varying model set incompleteness by specifying a Gaussian mixture model for the right hand side

of equation (2) as:

$$f(y_t|\tilde{\boldsymbol{y}}_t, \sigma_{1t}^2, \ldots, \sigma_{nt}^2) \;\; = \;\; \sum_{i=1}^{n} w_{it} \mathcal{N}(y_t|\tilde{y}_{it}, \sigma_{it}^2), \tag{4}$$

where σ_{it}^2 for $i = 1, \ldots, n$ and $t = 1, \ldots, T$ is specified to follow the stochastic volatility process

$$\log \sigma_{it}^2 \;\; \sim \;\; f(\log \sigma_{it}^2 | \log \sigma_{i,t-1}^2, \sigma_{\eta}^2). \tag{5}$$

The vector $(\sigma_{1t}^2, \ldots, \sigma_{nt}^2)'$ indicates the potential size of the misspecification in each of the combination models of the mixture. When the values of the vector $(\sigma_{1t}^2, \ldots, \sigma_{nt}^2)'$ are large, the overall uncertainty is substantial. When this uncertainty level tends to zero then the mixture of experts or the smoothly mixing regressions model is recovered as limiting case as shown in the following proposition.

Proposition 2.1 (***Mixture representation under model set incompleteness***)**.** *Under standard regularity conditions (integrals and summations exist) and given the information sets of all individual models, the marginal forecast density of y_t has the following discrete/continuous mixture representation*

$$f(y_t|\mathfrak{I}_t, \sigma_{1t}^2, \ldots, \sigma_{nt}^2) = \sum_{i=1}^{n} w_{it} \int \mathcal{N}(y_t|\tilde{y}_{it}, \sigma_{it}^2) f(\tilde{y}_{it}|\mathfrak{I}_{it}) d\tilde{y}_{it}. \tag{6}$$

If the uncertainty level, controlled by σ_{it}^2, $i = 1, \ldots, n$, *tends to zero, then*

$$f(y_t|\mathfrak{I}_t) \longrightarrow \sum_{i=1}^{n} w_{it} f(y_t|\mathfrak{I}_{it}). \tag{7}$$

A proof is presented in the Supplementary Material.

2.2 Dimension reduction and dynamic learning

In order to construct forecast density combinations for large data sets, say time series of several hundreds or thousand of observations, we introduce in this section a dimension reduction and learning process.

Clustering of forecasts. Without dimension reduction, the number of latent weights to estimate may be very large at every time period t which can be computationally demanding. As a first step in the dimension reduction process, the forecast densities of n series are clustered into m exclusive groups, using features of the forecast densities as discussed before. This allows to deal with model dependence, which is well documented in empirical studies but often ignored in density forecasting. Therefore, forecast densities with a similar level of dependence structure can be grouped together. Also, this grouping can change over time, following a learning mechanism which is defined by a sequential clustering rule. In such a way, even if the number of clusters is kept constant over time, the compositions of the clusters vary.[1] Details of the sequential clustering rule are given later and in the Supplementary Material, Section S.2. The clustering makes use of an allocation variable, ξ_{ijt}, which takes the value

[1]We note that the number of clusters could also be considered to vary over time, but their interpretation is then more difficult.

1 if the i-th forecast density is assigned to the j-th cluster of densities and 0 otherwise. This gives an $(n \times m)$ allocation matrix $\boldsymbol{\Xi}_t = (\boldsymbol{\xi}_{1t}, \ldots, \boldsymbol{\xi}_{jt}, \ldots, \boldsymbol{\xi}_{mt})$, with $\boldsymbol{\xi}_{jt} = (\xi_{1jt}, \ldots, \xi_{ijt}, \ldots, \xi_{njt})'$ with typical element $\xi_{ijt} \in \{0, 1\}$.

This clustering procedure also involves the construction of an $n \times m$ coefficient matrix $\boldsymbol{B}_t$, with the i-th row and j-th column element given by $b_{ijt} \in \mathbb{R}$, which is intended to show how each of the n forecasts contributes to the combination of forecasts. For the specification of specific values of the coefficients b_{ijt}, we propose two alternative strategies. In one strategy we assume that each model contributes to the combination with a specific weight driven by a model-specific forecasting performance with learning. Let $n_{jt} = \sum_{i=1}^{n} \xi_{ijt}$ be the number of forecast densities in the j-th cluster at time t and let g_{ijt} be the log score (see Mitchell and Hall, 2005 and the Supplementary Material) of the data series i at time t, then:

$$b_{ijt} = \begin{cases} \sum_{s=1}^{t} \exp\{g_{ijs}\} / \sum_{i=1}^{n} \sum_{s=1}^{t} \exp\{g_{ijs}\} & \text{if } \xi_{ijt} = 1 \\ 0 & \text{otherwise} \end{cases} \tag{8}$$

In order to compare the effect of learning with no-learning, we also consider the case where all coefficients in the cluster have the same weight, which corresponds to set:

$$b_{ijt} = \begin{cases} 1/n_{jt} & \text{if } \xi_{ijt} = 1 \\ 0 & \text{otherwise.} \end{cases} \tag{9}$$

Modelling large set of weights as a dynamic factor model. We start to specify latent cluster weights $\boldsymbol{v}_t = (v_{1t}, \ldots, v_{mt})'$ as a basic m-variate normal

random walk learning process

$$\boldsymbol{v}_t = \boldsymbol{v}_{t-1} + \boldsymbol{\eta}_t, \quad \boldsymbol{\eta}_t \overset{iid}{\sim} \mathcal{N}_m(\mathbf{0}_m, \boldsymbol{\Sigma}). \tag{10}$$

As a next step we specify a large set of n weights as linear combinations of the cluster weights

$$\boldsymbol{x}_t = \boldsymbol{B}_t \boldsymbol{v}_t \tag{11}$$

with the individual weights given as $x_{it} = \sum_{j=1}^{m} b_{ijt} v_{jt}, i = 1, 2, \ldots, n$. The model of (10) and (11) is a dynamic factor one, with perfect factors. That is, there is no noise in the connection between $\boldsymbol{x}_t$ and $\boldsymbol{v}_t$. The cluster weights $\boldsymbol{v}_t$ are now interpreted as latent factors with factor weights given as $\boldsymbol{B}_t$. In this way, both dimension reduction and dynamic learning is specified.

It is seen from equation (11) that the $n \times 1$ vector $\boldsymbol{x}_t$ is a linear combination of normally distributed random variables with the multivariate normal distribution: $\boldsymbol{x}_t \sim \mathcal{N}_n(\boldsymbol{B}_t \boldsymbol{v}_{t-1}, \boldsymbol{B}_t \boldsymbol{\Sigma} \boldsymbol{B}_t')$ which is degenerate in the case of an equal weight matrix $\boldsymbol{B}_t$.

Logistic transformation of the n-dimensional latent weights $\boldsymbol{x}_t$ to the $(n-1)$ simplex. We make use of an auxiliary $(n-1)$ vector $\boldsymbol{q_t}$ which is defined as $\boldsymbol{x}_t$ in deviation from its last value x_{nt}. That is, $\boldsymbol{q}_t = \boldsymbol{D}\boldsymbol{x}_t$ where the $(n-1) \times n$ matrix $\boldsymbol{D}$ is given by $\boldsymbol{D} = (\boldsymbol{I}_{n-1} | -\boldsymbol{\iota}_{n-1})$, with I_{n-1} equal to the $(n-1) \times (n-1)$ identity matrix, and ι_{n-1} is the $(n-1) \times 1$ vector containing only ones and there is singularity, see the Supplementary Material Section S.1 for details. Map the $(n-1)$ vector $\boldsymbol{q}_t$ to the $(n-1)$ dimensional simplex using a logistic transformation so that the resulting weights can be interpreted as a convex set of probabilistic

Elements in	Dimension of the latent weights and factors m	n	$n-1$
[0,1]		$\boldsymbol{w}_t$ $\xleftarrow[w_{nt}=1-\sum_{i=1}^{n-1}\tilde{w}_{it}]{w_{it}=\tilde{w}_{it}\ \ (i=1,2,\ldots,n-1)}$	$\tilde{\boldsymbol{w}}_t$
			$\uparrow\ \tilde{\boldsymbol{w}}_t = g(\boldsymbol{q}_t)$
$(-\infty,\infty)$	$\boldsymbol{v}_t$ $\xrightarrow{\boldsymbol{x}_t=\boldsymbol{B}_t\boldsymbol{v}_t}$	$\boldsymbol{x}_t$ $\xrightarrow{\boldsymbol{q}_t=D\boldsymbol{x}_t}$	$\boldsymbol{q}_t$

Table 1: Transformations diagram between latent weights and factors; $\boldsymbol{w}_t$ and $\tilde{\boldsymbol{w}}_t$ have logistic normal distributions; $\boldsymbol{v}_t$, $\boldsymbol{x}_t$ and $\boldsymbol{q}_t$ have multivariate normal distributions.

weights, denoted by the vector $\boldsymbol{w}_t$ and again there is singularity since once we have $w_{1t}, \ldots, w_{(n-1)t}$ we also known w_{nt}. Thus, this transformation, indicated by the function $g(.)$ goes from $(n-1)$ elements from $\boldsymbol{q}_t$ in $\mathbb{R}^{n-1}$ to the $(n-1)$ vector $\tilde{\boldsymbol{w}}_t = (w_{1t}, \ldots, w_{(n-1)t})'$, defined on the simplex $\mathbb{S}^{n-1}$.

We present the different steps in the transformation diagram of Table 1 going counterclockwise from the random walk cluster weights $\boldsymbol{v}_t$ shown at the bottom left to the large set of weights $\boldsymbol{x}_t$, next going to the auxiliary $(n-1)$-vector $\boldsymbol{q}_t$ and then taking the logistic transformation step $g(\boldsymbol{q}_t)$ on the vertical line which yields the $(n-1)$ vector $\tilde{\boldsymbol{w}}_t$ that is defined on the simplex of large dimension. Finally, the complete n-dimensional weight vector $\boldsymbol{w}_t$ is listed in the middle of the top line.

As a next result we present the distribution of $\tilde{\boldsymbol{w}}_t$ in the following proposition.

Proposition 2.2 (***Logistic normal distribution of weights*** $\tilde{\boldsymbol{w}}_t$). *Let the* $n \times 1$ *vector* $\boldsymbol{x}_t = \boldsymbol{B}_t\boldsymbol{v}_t$ *have a multivariate normal distribution:* $\boldsymbol{x}_t \sim \mathcal{N}_n(\boldsymbol{B}_t\boldsymbol{v}_{t-1}, \boldsymbol{B}_t\boldsymbol{\Sigma}\boldsymbol{B}_t')$. *Define the* $(n-1)$ *vector* $\tilde{\boldsymbol{w}}_t$ *as:*

$$w_{it} = \frac{\exp(x_{it} - x_{nt})}{\sum_{i=1}^{n}\exp(x_{it} - x_{nt})}, \qquad i = 1, 2, \ldots, n-1. \tag{12}$$

Then $\tilde{\boldsymbol{w}}_t$ *follows a logistic normal distribution:* $\tilde{\boldsymbol{w}}_t \sim \mathcal{L}_{n-1}(\boldsymbol{DB}_t\boldsymbol{v}_{t-1}, \boldsymbol{DB}_t\boldsymbol{\Sigma}\boldsymbol{B'}_t\boldsymbol{D'})$*, where* $w_{nt} = 1 - \sum_{i=1}^{n-1} w_{it}$.

A proof is presented in the Supplementary Material.

Probabilistic cluster weights in the m-dimensional space. Take the cluster weights $\boldsymbol{v}_t$ in deviation of their final value, that is, $\boldsymbol{D_{(m-1)}}\boldsymbol{v}_t$, where $\boldsymbol{D_{(m-1)}}$ has the same structure as the matrix $\boldsymbol{D}$ but it is now an $(m-1) \times m$ matrix. Use the logistic transformation $g(\boldsymbol{D_{(m-1)}}\boldsymbol{v}_t)$ in order to move $\boldsymbol{D_{(m-1)}}\boldsymbol{v}_t$ from $\mathbb{R}^{m-1}$ to the simplex $\mathbb{S}^{m-1}$ and label the resulting vector of probabilistic cluster weights as $\tilde{\boldsymbol{z}}_{\boldsymbol{t}} = g(\boldsymbol{D_{(m-1)}}\boldsymbol{v}_t)$ with elements $z_{jt}, j = 1, 2, \ldots, (m-1)$ with $z_{mt} = 1 - \sum_{j=1}^{m-1} \tilde{z}_{jt}$. In the empirical analysis we present results on the time series pattern of these weights. The density function is given as $\tilde{\boldsymbol{z}}_t \sim \mathcal{L}_{m-1}(\boldsymbol{D_{(m-1)}}\boldsymbol{v}_{t-1}, \boldsymbol{D_{(m-1)}}\boldsymbol{\Sigma}\boldsymbol{D'_{(m-1)}})$. There exists a nonlinear transformation from the weights $\tilde{\boldsymbol{z}}_t$ in the low dimensional space to the weights $\tilde{\boldsymbol{w}}_t$ in the high-dimensional space. One can use here the class preserving property of the logistic normal distribution. For a general treatment and more details, we refer to Casarin et al. (2017).

Remark on Diagnostic learning. A second way of learning is diagnostic by making use of the variances of the disturbances in the combination models. Here, the effect of misspecification or incompleteness of the model set is analysed. We show results in the empirical section.

2.3 State space representation and efficient filtering algorithms

Given the results in the preceding section, we can now write the forecast density combination model in nonlinear state space form. This representation allows us to make use of algorithms based on sequential Monte Carlo methods such as particle filters.

Proposition 2.3 (***Nonlinear state space representation***). *The forecast density combination model given in Section 2 has the following nonlinear state space representation where the density of the measurement equation is a large $n-$dimensional finite mixture of normals with time varying variances and the density of the mixture weights is the logistic normal one from Proposition 2.2:*

$$y_t \sim \sum_{i=1}^{n} w_{it} \mathcal{N}\left(\tilde{y}_{it}, \sigma_{it}^2\right) \tag{13}$$

$$\tilde{\boldsymbol{w}}_t \sim \mathcal{L}_{n-1}\left(\boldsymbol{D}\boldsymbol{B}_t \boldsymbol{v}_{t-1}, \boldsymbol{D}\boldsymbol{B}_t \boldsymbol{\Sigma} \boldsymbol{B}'_t \boldsymbol{D}'\right), \tag{14}$$

where $\tilde{\boldsymbol{w}}_t = (w_{1t}, \ldots, w_{n-1,t})'$, $w_{nt} = 1 - \tilde{\boldsymbol{w}}_t' \boldsymbol{\iota}_{n-1}$.

Distributions other than the logistic-normal can be used for weights such as the Dirichlet distribution, but as noted in Aitchinson and Shen (1980) this distribution may be too restrictive to be realistic in our analysis since the components of a Dirichlet composition have a correlation structure determined solely by the normalisation operation.

Next, we present a result that shows how this nonlinear state space model can be written, for computational purposes, as a generalised linear model with a nonlinear local level transition function when the real and simplex space of

the random measures are equipped with suitable operations and norms.[2] These properties enable us to state the following result.

Corollary 2.1. *Let $\boldsymbol{s}_t$ be a $n-$dimensional allocation vector, with $\boldsymbol{s}_t \sim \mathcal{M}_n(1, \boldsymbol{w}_t)$, where $\mathcal{M}_n(1, \boldsymbol{w}_t)$ denotes the multinomial distribution. Then, the state space model given in Proposition 2.3 can be written as:*

$$y_t = \sum_{i=1}^{n} (\tilde{y}_{it} + \varepsilon_{it}) s_{it}, \quad \varepsilon_{it} \sim \mathcal{N}(0, \sigma_{it}^2), \tag{15}$$

$$s_{it} = \begin{cases} 1 & \textit{with probability } w_{it} \\ 0 & \textit{otherwise} \end{cases}, \tag{16}$$

$$\boldsymbol{w}_t = g(\boldsymbol{D}\boldsymbol{x}_t) \tag{17}$$

$$\boldsymbol{x}_t = \boldsymbol{B}_t \boldsymbol{v}_t \tag{18}$$

$$\boldsymbol{v}_t = \boldsymbol{v}_{t-1} + \boldsymbol{\eta}_t, \quad \boldsymbol{\eta}_t \overset{iid}{\sim} \mathcal{N}_m(\boldsymbol{0}_m, \boldsymbol{\Sigma}), \tag{19}$$

where the function g is the logistic transformation given earlier and the matrix $\boldsymbol{\Sigma}$ is given as diagonal with elements σ_j^2, $j = 1, 2, \ldots, m$.

Algorithmic aspects: parallel sequential clustering and filtering.

The analytic solution of the filtering problem is generally not known, also the clustered-based mapping of the forecast densities requires the solution of an optimisation problem which is not available in closed form. Thus, we apply a sequential numerical approximation of the two problems and use an algorithm which, at time t, iterates over the following two steps:

1. Parallel sequential clustering computation of forecast densities.

[2]For details and background, see Aitchinson (1986) and Aitchinson (1992) and Billheimer et al. (2001).

2. Parallel sequential Monte Carlo approximation of weights and parameters of the combination models.

For details and background on the parallel sequential filtering we refer to the sequential Monte Carlo methods as in Casarin et al. (2015).

As regards the sequential clustering, we apply a parallel and sequential k-means method with a forgetting factor for the sequential learning of the group structure. K-means clustering is a method partitioning a set of n forecast densities into m disjoints sets defined clusters. Given a definition of dependence, the k-means will group forecast densities based on their distance. Moreover, the sequential k-means algorithm is easy to parallelise which has been executed on multi core CPU and GPU computing environments. Further details are given in the Supplementary Material.

3 Empirical applications

As a first application we focus on the financial case, briefly discussed in the previous section. We report results on several features of the combined forecast density of a replication of the daily Standard & Poor 500 (S&P500) index, including the economic value of tail events like Value-at-Risk. The second application considers the extended Stock and Watson (2005) dataset, which includes 142 series sampled at a quarterly frequency from 1959Q1 to 2011Q2. Here we focus on obtaining a set of relevant clusters and we provide evidence on forecast probabilities of a recession. In the financial and macroeconomic case, we study the weight patterns of the clusters over time which provide valuable signals that may lead to improved financial and macroeconomic modelling and

forecasting.

3.1 Forecast density combination features and S&P500 index tracking

The econometrician interested in forecasting the density of this index has, at least, two standard strategies. First, she can model the index with a parametric or non-parametric specification and produce a forecast of it. Second, she can forecast the price of each stock i and then aggregate them using an approximation of the unknown weighting scheme.

We propose an extension of the second strategy based on the fact that many investors, including mutual funds, hedge funds and exchange-traded funds, try to replicate the performance of the index by holding a set of stocks, which are not necessarily the exact same stocks included in the index. Apart from using the S&P500 index, we collected 1856 individual stock daily prices quoted in the NYSE and NASDAQ from Datastream over the sample March 18, 2002 to December 31, 2009, for a total of 2034 daily observations for each individual series. To control for liquidity we impose that each stock has been traded a number of days corresponding to at least 40% of the sample size. We compute log returns for all stocks. The S&P500 and the cross-section average statistics of all series are reported in Table S.2 in section S.4 of the Supplementary Material. We produce a density forecast for each of the stock returns and then apply our forecast density combination scheme in order to compute the time patterns of the weights of the different clusters and several other features of the combined density forecast. The cluster weights indicate their relative forecasting importance over

time. That is, a side output of our replication strategy is evidence of which sets of assets track more accurately the aggregate index. This may lead to improved investment policies, which is a topic for future research.

Model estimation.

To ease on the computational workload, we apply an optimisation method to estimate the posterior modes of the parameters from a Normal GARCH(1,1) model and a t-GARCH(1,1) model[3] using rolling samples of 1250 trading days (about five years) for each stock return:

$$y_{it} = c_i + \kappa_{it}\zeta_{it}, \tag{20}$$

$$\kappa_{it}^2 = \theta_{i0} + \theta_{i1}\zeta_{i,t-1}^2 + \theta_2\kappa_{i,t-1}^2, \qquad i = 1, 2, \ldots, n, \tag{21}$$

where y_{it} is the log return of stock i at day t, $\zeta_{it} \sim \mathcal{N}(0,1)$ and $\zeta_{it} \sim \mathcal{T}(\nu_i)$ for the Normal and t-Student cases, respectively. The number of degrees of freedom ν_i is estimated in the latter model. We produce 784 one day ahead forecast densities from January 1, 2007 to December 31, 2009. Our out of sample period is associated with high volatility driven by the US financial crisis and includes, among others, events such as the acquisitions of Bern Stearns, the default of Lehman Brothers and all events of the following week.

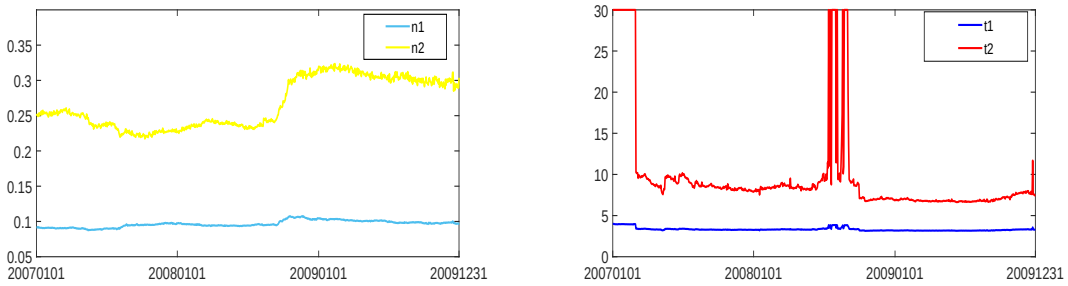
For further computational convenience, we specify for this case the parameter matrix $\boldsymbol{B}_t$ in equation (9) as equal weights.[4]

Four clusters.

As first step, we apply the sequential cluster analysis to our forecast densities.

[3]Given our flat prior and large sample, these estimates are equivalent to maximum likelihood estimates and also are approximate Bayes mean estimates

[4]See the macroeconomic case for a comparison with a different scoring rule.



Normal GARCH(1,1) t-GARCH(1,1)

Figure 1: The figures present the average variance of the forecasts from the two clusters for the Normal GARCH(1,1) models based on low (cluster 1, light blue) and high (cluster 2, yellow) volatility in the left panel; and the average degree of freedom of the forecasts from the two clusters for the t-GARCH(1,1) models based on low (cluster 3, dark blue) and high (cluster 4, red) degrees of freedom in the right panel. The degrees of freedom are bounded to 30.

We compute two clusters of forecast densities of the Normal GARCH(1,1) model and two clusters for the t-GARCH(1,1) model. The first two are characterised by low and high volatility; the third and the fourth ones are characterised by thick or no thick tails.[5] The cluster analysis is repeated every time a new forecast density is produced and therefore the cluster composition varies over time. Figure 1 presents results about these features. The clusters for the Normal GARCH(1,1) models differ substantially in terms of forecasted variance with cluster 1, with the light blue colour, having a rather low constant variance value over the entire period while cluster 2, with the yellow colour, has a variance more than double in size including a shock in the latter part of 2008. For the t-GARCH(1,1) model it is seen that cluster 3, with the dark blue colour, has a relatively constant thick tail over the entire period while cluster 4, with the red colour, has an average value of 10 for the degrees of freedom and in the crisis period the density collapses

[5]Low degrees of freedom occur jointly with a large scale and high degrees of freedom occur jointly with a low scale.

to a normal density with degrees of freedom higher than 30. The Lehman Brother effect is visible in the figure, with an increase of volatility in the normal cluster 2 and a decrease in the degrees of freedom in the t-cluster 4.

Time varying cluster weight patterns.

Plots of the estimated cluster weights z_{jt}, in the low dimensional simplex, which were defined in Section 2 are shown in Figure 2. Clearly there is an indication of a time varying pattern of the weights. One can distinguish three different subperiods. In the subperiod before the crisis, the Normal GARCH cluster with high volatility, cluster 2, and the t-GARCH cluster with low degrees of freedom, cluster 3, have almost equal high weights while clusters 1 and 4 play a much less important role. In the crisis period of 2008, cluster 3 receives almost all the weight with clusters 1 and 2 almost none. Some of the assets lead the large market decrease in that period. This results in very fat tailed densities and our combination scheme takes advantage of this information and assigns to cluster 3 more weight. In the period after the Lehman Brothers collapse cluster 3 receives again a substantial weight while the normal cluster 2, with large variance, is getting gradually more weight. Clearly time-varying fat tails are an important feature.

We also make use of canonical correlations, see Hotelling (1936), in order to show how the joint dependence among the weights has changed over time. The canonical correlations of the weights of each cluster versus the others are computed from the first one year of data, January 1, 2007 to December 31, 2007, and next we use an expanding window approach to the full sample until December 31, 2009. As one may expect from the time series behaviour of the individual cluster weights, the top-right panel in Figure 2 shows that

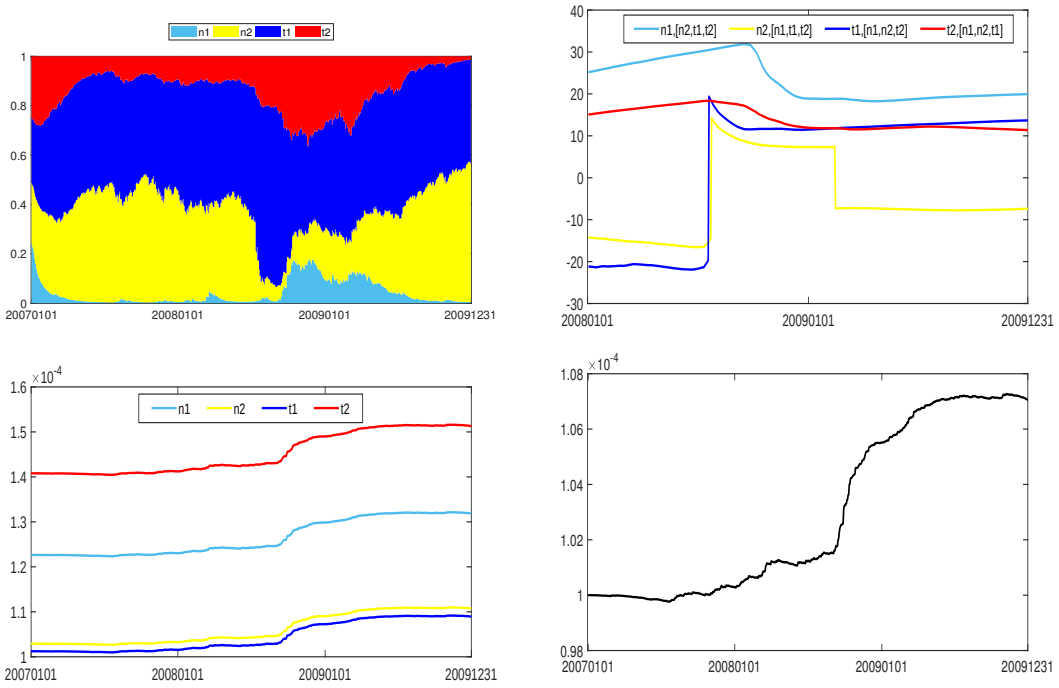


Figure 2: Top-left: the mean logistic-normal weights for the two Normal GARCH clusters, labeled in the graph "n1" and "n2", and for the two t-GARCH clusters, labeled in the graph "t1" and "t2". Top-right: 1-year canonical correlations of the weights for the clusters "n1", "n2", "t1" and "t2" respectively versus the other cluster weights (between square brackets). Bottom-left: posterior mean estimates of incompleteness measures in the four clusters in the scheme DCEW-SV. Bottom-right: average of the posterior mean estimates of all model incompleteness measures.

major changes occur onward from the Lehman Brother default, in particular, for clusters 2 and 3. We note that the correlation of cluster 2 with clusters 1 and 4 returns to the pre-crisis period; cluster 3 is more positively related to the other clusters than before the financial crisis.

Model set incompleteness.

We measure incompleteness for the model set Density Combination with Equal Weights and Stochastic Volatility, (DCEW-SV). Signals of model set incompleteness are shown in the bottom panel of Figure 2. We compute the

incompleteness contribution of each individual cluster as the average value of the squared posterior residuals, see equation (15). It is seen that the Normal GARCH cluster "n1" with low volatility and the $t-$Garch cluster "t2" with high degrees of freedom have the higher average incompleteness and the Normal GARCH cluster "n2" with high volatility and the $t-$Garch cluster "t1" with low degrees of freedom have lower average incompleteness. This diagnostic information confirms that clusters "n1" and "t2" give lower forecast accuracy. In terms of time series patterns, incompleteness for the four clusters follows a similar trend as the trend in the overall measure of incompleteness with a large increase after Lehman Brothers events.[6] We note that it is seen in Figure 2 that cluster "t2" has a larger weight than cluster "n1" in the combination but it has a worse fit. This result may be due to the misspecification feature.

We also plot an average estimate of the overall model incompleteness by computing the posterior mean estimates for σ_{it}^2 and taking their average, that is $\overline{\sigma}_t^2 = \sum_{i=1}^n \sigma_{it}^2/n$. The average variance estimate has a 7% increase in September 2008, which is due to the default of Lehman Brothers and related following events. Interestingly, the volatility does not reduce in 2009, a year with large positive returns opposite the large negative returns in 2008.

Forecast accuracy of center and shape of the distribution.

We compare the performance of our approach with five different basic models applied to the S&P500 log returns: a white noise model (or a random walk for prices), often used as a main benchmark in equity premium forecastability; the Normal GARCH(1,1) and the t-GARCH(1,1) models described above. In order

[6]We note that one may experiment with a larger set of individual models, see for example Geweke and Durham (2012).

to explore the sensitivity of our results for model set incompleteness in more detail, we include the GJR-GARCH(1,1) model in Glosten et al. (1993) that includes leverage effects in the model set. The GJR-GARCH is a richer model than the standard GARCH and should fit the data better. In fact, leverage effect is considered among the stylised facts of financial returns. So the added feature may become relevant in our analysis. Finally, since it might difficult to know which of the GARCH models perform better *ex-ante*, we apply also an equal weight combination of the three GARCH models, labeled EW-GARCH.

	RMSPE	LS	CRPS	avQS-T	avQS-L	Violation
WN	1.852	-9.045	1.017	0.429	0.425	3.57%
Normal GARCH	1.852	-4.164**	0.956**	0.139**	0.195**	2.93%
t-GARCH	1.852	-2.738**	0.937**	0.118**	0.154**	2.55%
GJR-GARCH	1.852	-4.068**	0.955**	0.125**	0.158**	2.75%
EW-GARCH	1.853	-3.145**	1.018	0.144**	0.171**	2.80%
DCEW	**1.812****	**2.249****	**0.911****	**0.114****	**0.149****	0.90%
DCEW-SV	1.816**	2.206**	0.913**	**0.114****	**0.149****	**1.02**%

Table 2: Forecasting results for next day S&P500 log returns. Bold numbers indicate the best statistic for each loss function. One or two asterisks indicate that differences in accuracy from the white noise (WN) benchmark are credibly different from zero at 5%, and 1%, respectively, using the Diebold-Mariano t-statistic for equal loss. The underlying p-values are based on t-statistics computed with a serial correlation-robust variance, using the pre-whitened quadratic spectral estimator of Andrews and Monahan (1992). The column "Violation" shows the number of times the realised value exceeds the 1% Value-at-Risk (VaR) forecasted by the different models over the sample.

Out-of-sample forecasting result are presented in Table 2. The first three columns deal with location and shape features of the forecast densities. It is seen that our combination schemes produce the lowest Root Mean Squared Prediction Error (RMSPE) and Cumulative Rank Probability Score (CRPS) and the highest Log Score (LS), see also Section S.3 of the Supplementary Material for

more details. The results indicate that the combination schemes are statistically superior to the no-forecastability WN benchmark. The Normal GARCH(1,1) model, the t-GARCH(1,1) model and the GJR-GARCH(1,1) model fitted on the index also provide more accurate density forecasts than the WN, but not on point forecasting. For all three score criteria, the statistics given by the three individual models are inferior to our combination schemes.

Tail estimates and their dynamic behaviour.

Apart from forecast accuracy in the center and of the complete shape of the distribution, we investigate whether the results also possess valuable signals about the tails. We consider two statistics that refer to left and right tails of the forecast densities. These refer to weighted averages of Gneiting and Raftery (2007) quantile scores that are based on quantile forecasts that correspond to the forecast densities from the different models. In the Supplementary Material it is shown that avQS-T emphasizes both tails and avQS-L the left tail of the forecast density relative to the realization 1-step ahead. To study how the models perform in the left tail forecasts over time, we consider the cumulative sum of avQS-L and the most accurate model at observation t produces the lowest cumavQS-L$_{i,h,t}$. The fourth and fifth columns of Table 2 show results for tail evaluation. Our schemes provides the lowest avQS-T and avQS-L statistics, confirming the accuracy of the method in the tails of the distribution.

Figure 3 shows for the time series of the full sample the cumulative avQS-L for the t-GARCH(1,1) model, the best ex-post GARCH model, the combination of GARCH models and DCEW model set. We note that our method requires some observations in the beginning to catch up with the other models. However, from August 2007 when stock markets start to experience large stress, it provides

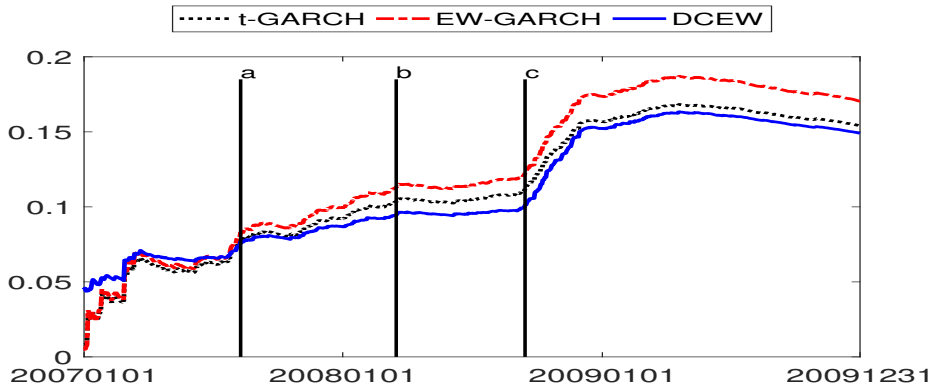


Figure 3: Cumulative left quantile scores described in formula S.10 (Appendix) of the t-GARCH model, EW-GARCH model and DCEW. Timeline legend: a - 8/9/2007, BNP Paribas redemptions on three investment funds; b - 3/17/2008, collapse of Bear Stearns; c - 9/15/2008, Lehman bankruptcy.

the most accurate tail forecasts. The gap between the three models increases steadily over time and it becomes substantially larger after the collapse of Bear Stearns. With the default of the Lehman brothers, the accuracy of all three schemes reduces sharply until November/December 2008 when central banks and governments from around the World started to take actions which reduced the volatility in financial markets. Our DCEW, however, provides the lowest statistic until the end of the sample.

Economic value of tail information.

As economic measure, we apply a Value-at-Risk (VaR) based measure, see Jorion (2006). We compare the accuracy of our models in terms of violations, that is the number of times that negative returns exceed the VaR forecast at time t, with the implication that actual losses on a portfolio are worse than had been forecasted. Higher accuracy results in numbers of violation close to nominal value of 1%. Moreover, to have a gauge of the severity of the violations we compute the total losses by summing the returns over the days of violation

for each model. When looking to VaR violations, reported in the final column of Table 2, the number for all individual models is high and above 1%, with the WN higher than 3%. The dramatic events in our sample, including the Lehman default and all the other features of the US financial crisis provide an explanation for the result. It is important to note that the two combination schemes provide the best statistics, with violations very close to the 1% theoretical value. The property of our combination schemes to assign higher weights to the fat tail cluster 3 helps to model more accurately the lower tail of the index returns and covers more adequately risks.

Remark on computation time.

Details are presented in the Supplementary Material, in particular, Table S.6 compares the execution time of the GPU parallel implementation of our density combination strategy and the CPU multi-core implementation. The results show substantial gains due to GPU parallelisation.

3.2 Dynamic cluster weights and recession probabilities in a large macroeconomic data set

We consider the extended Stock and Watson (2005) dataset, which includes 142 series sampled at a quarterly frequency from 1959Q1 to 2011Q2. A graphical description of the data is given in Figure S.2 in the Supplementary Material. The dataset includes only revised series and not vintages of real-time data.[7] In order to deal with stationary series, we apply the series-specific transformation suggested in Stock and Watson (2005). We also re-scale the series to have zero

[7]See Aastveit et al. (2018) for a real-time application, with fewer series, of combined density nowcasting and the role of model set incompleteness over vintages and time.

mean.

Set-up of the experiment.

We split the sample size 1959Q3-2011Q2 in two periods. The initial 102 observations from 1959Q3-1984Q1 are used as initial in-sample period; the remaining 106 observations from 1985Q1-2011Q2 are used as an out-of-sample period.

We evaluate combined forecast densities of four core variables often considered in monetary policy analysis: real GDP growth, Inflation measured as percentage change in the price deflator, 3-month Treasury Bill rate and total Employment for $h = 1, \ldots, 5$ step-ahead horizons but restrict the presentation to results for $h = 1$, 3, 5 horizons. For all variables we apply an AR(1) model and the Dynamic Factor Model (DFM) with 5 factors described in Stock and Watson (2012) as two benchmarks.

As described in Section 2, we consider two alternative strategies for the specification of the parameter matrices $\boldsymbol{B_t}$: equal weights and score recursive weights, where in the second case we fix the log scores for the various horizons h. We note that we keep the volatility of the incompleteness term constant, for convenience. In the present analysis, the number of components matters more.

We construct combinations of forecast densities for eight different specifications of the AR(1) model. That is, we make use of univariate versus multivariate models; equal cluster weights versus weights based on past log score performance, and 5 versus 7 clusters. Thus, we have eight cases, defined as UDCEW5 (univariate density combination based on 5 clusters with equal weights within clusters), MDCEW5 (multivariate density combination based on 5 clusters with equal weights within clusters), UDCLS5 (univariate density

combination based on 5 clusters with recursive log score weights within clusters), MDCLS5 (multivariate density combination based on 5 clusters with recursive log score weights within clusters), UDCEW7 (univariate density combination based on 7 clusters with equal weights within clusters), MDCEW7 (multivariate density combination based on 7 clusters with equal weights within clusters), UDCLS7 (univariate density combination based on 7 clusters with recursive log score weights within clusters), MDCLS7 (multivariate density combination based on 7 cluster with recursive log score weights within clusters).

Model estimation.

For each of the four variables we make use of a Gaussian autoregressive model of the first order, AR(1),

$$y_{it} = \alpha_i + \beta_i y_{it-1} + \zeta_{it}, \quad \zeta_{it} \sim \mathcal{N}(0, \sigma_i^2). \tag{22}$$

We estimated the model using Bayesian inference and use a rather diffuse informative Normal-Inverse-Gamma prior with means for α_i and the β equal to zero and variances equal to 100. For the variance σ_i^2 we use an Inverse-Gamma with degrees of freedom equal to the number of lags (one) and intercept, that is two. The AR models are estimated recursively and $h-$step ahead (Bayesian) $t-$Student forecast densities are constructed using a direct approach extending each vintage with the new available observation; see for example Koop (2003) for the exact formula of the mean, standard deviation and degrees of freedom.

We also consider as a benchmark the DFM with 5 factors described in Stock

and Watson (2012) as another benchmark. More precisely:

$$\boldsymbol{y}_t = \Lambda \boldsymbol{f}_t + \boldsymbol{\varepsilon}_{ft}, \quad \Phi(\mathrm{L})\boldsymbol{f}_t = \boldsymbol{\eta}_{ft}, \tag{23}$$

where the $\boldsymbol{y}_t = (y_{1t}, \ldots, y_{Kt})'$ is a $(K \times 1)$ vector of variables (in our case $K = 4$), $\boldsymbol{f}_t = (f_{1,t}, \ldots, f_{r,t})'$ is an r vector of latent factors, Λ is a $K \times r$ matrix of factors loadings, $\Phi(\mathrm{L})$ is an $(r \times r)$ matrix lag polynomial, $\boldsymbol{\varepsilon}_{ft}$ is a $(K \times 1)$ vector of idiosyncratic components and $\boldsymbol{\eta}_{rt}$ is an r vector of innovations. In this formulation the term $\Lambda \boldsymbol{f}_t$ is the common component of $\boldsymbol{y}_t$. Bayesian estimation of the model described in equation (23) is carried out using Gibbs Sampling given in Koop and Korobilis (2009).

Forecast accuracy of center and shape of the distributions

Table 3 reports the results to forecast real GDP growth, inflation measured as the price deflator of GDP growth, 3-month Treasury Bills and total employment for three different horizons and using three different scoring measures. For all variables, horizons and scoring measures our methodology provides more accurate forecasts than the AR(1) benchmark and the DFM benchmark. The DFM model provides in most cases more accurate forecasts than the AR(1) for real GDP and inflation at shorter horizons and gives mixed evidence for interest rates and employment, but several of our combination schemes outperform this benchmark. The combination that provides the largest gain is the multivariate one based on seven clusters and log score weights within clusters (MDCLS7), resulting in the best statistics 36 times out of 38 cases. In most of the cases, the difference is statistically credible at the 1% level. This finding extends evidence on the scope for multi-variable forecasting such as given in large Bayesian VAR,

see e.g. Bańbura et al. (2010) and Koop and Korobilis (2013). Fan charts in Figure S.4 of the Supplementary Material show that the forecasts are accurate even at our longest horizon, $h = 5$. The variable with low forecast gains is inflation, although our method provides credibly more accurate scores at the (at least) 5% credible level in several cases. Note that the multivariate combination based on 5 clusters and equal weights yields also some accurate forecasts for the 3-month Treasury Bill rate, see cluster MDCEW5.

The forecast gains are similar across different horizons for the four variables, that is around 10% relative to the AR benchmark in terms of RMSPE metrics and even larger for the log score and CRPS measures.[8] However, despite these consistent gains over horizons, the logistic-normal weights in Figure 4 differ across horizons. For example, when forecasting GDP growth (panel 1) cluster 4 has a weight around 20% at horizons 1 and 5, but half of this value at horizon 3, where clusters 2 and 5 have larger weights. The change is even larger for inflation, where cluster 2 has a 20% weight at horizon 1 and increases to 40-45% at horizon 5. The latter case also occurs when there is substantial instability over time. Changes over horizons are less relevant for the other two forecasted variables.

We conclude that combining joint model forecasts using multiple clusters with cluster-based weights provides substantial forecast gains in most cases. Of course, additional gains may be obtained by playing with a more detailed cluster grouping and different performance scoring rules for weights associated with models inside a cluster. This is left as a topic for further research.

[8]One would expect that RMSPE's are monotonic decreasing over longer horizons. This is not everywhere observed and is due to the fact of model misspecification.

Dynamic weight patterns.

We identify the clusters of forecast densities by applying our k-means clustering algorithm. Specifically, our forecast densities are grouped in clusters depending on mean, persistence and volatility properties. We are, in particular, interested in the interpretation and behaviour of the clusters over the full sample and consequently we impose that the cluster allocation of each model is fixed over the forecasting vintages. Note that in the finance exercise this assumption is relaxed. We assume alternatively 5 and 7 clusters.[9] In the grouping, we identify two clusters related to real activities; one cluster related to prices; and one cluster related to financial variables. The other clusters contains the remaining series. A detailed description of the 5 and 7 clusters is provided in Tables S.3-S.4 in the Supplementary Material. From the analysis of the time patterns of the weights in Figure 4 (see also Figure S.7 in the Supplementary Material for weights in the univariate combination), we note that the weights for the univariate combination are often less volatile than the weights in the multivariate approach. All figures show that the *sixth* cluster has a large weight, but several other clusters have also large positive weights, namely, clusters 2, 4, and 5 while clusters 1 and 7 do not receive much weight. Apparently, variables such as Exports, Imports and GDP deflator included in the sixth cluster play an important role in forecasting GDP growth, inflation, interest rate and employment.

Figure S.8 in the Supplementary Material shows a typical output of the model weights ($b_{k,ijt}$, with $k = 1, 2, 3, 4$ representing one of the four macroeconomic variables to be predicted) in the seven clusters. There are large differences across

[9]Interestingly, Stock and Watson (2012) find that a factor model with 5 factors provides superior forecasts to factor models with less factors. We also investigate combinations with a lower number of clusters, precisely 2 and 3 clusters, but forecasts are less accurate.

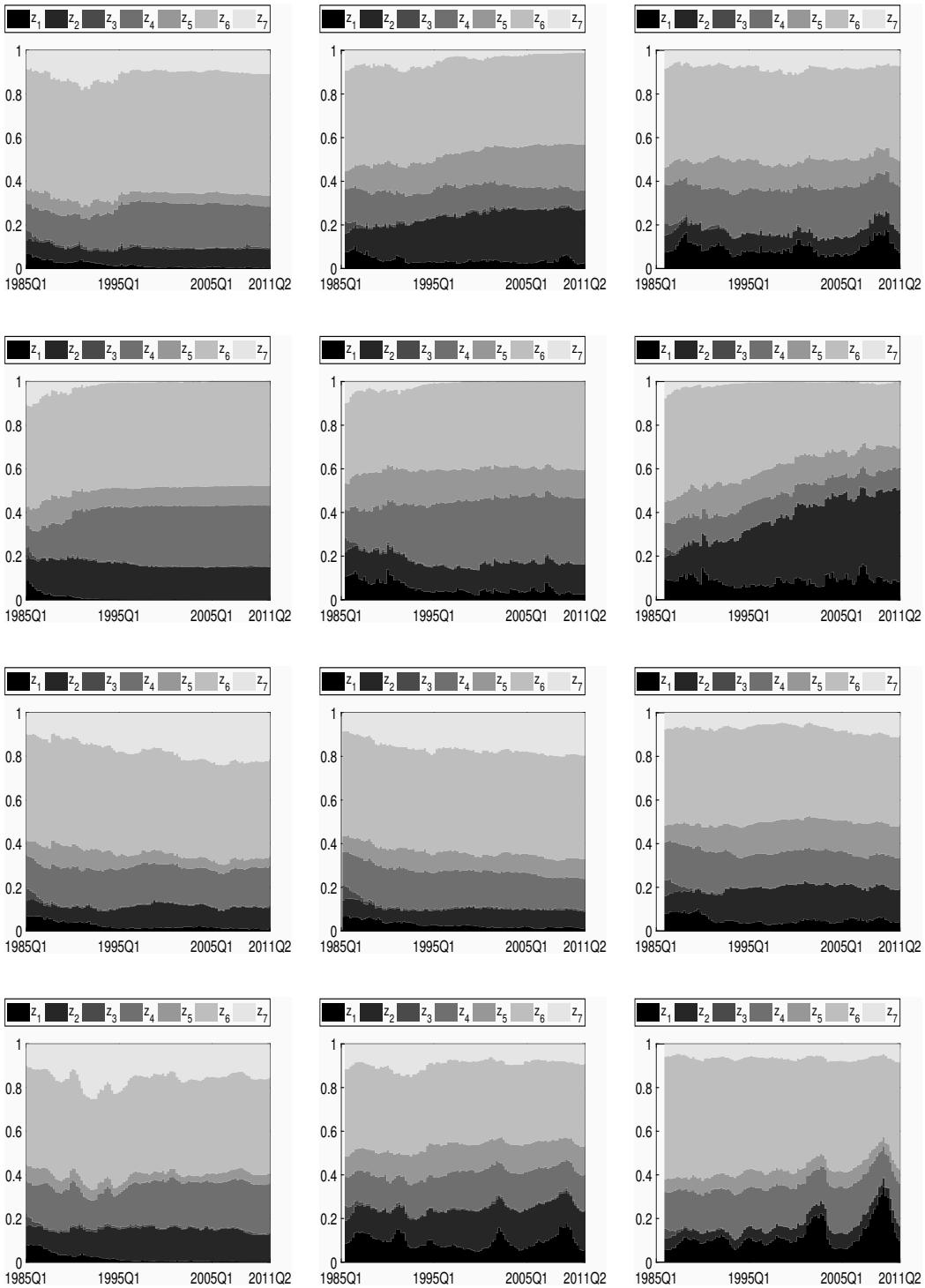


Figure 4: In each plot the logistic-normal weights (different lines) for the multivariate combination model are given. Rows: plot for the four series of interest (real GDP growth rate, GDP deflator, Treasury Bills, employment). Columns: forecast horizons (1, 3 and 5 quarters).

clusters: for clusters 2, 4, 5 and 6, only a few models have most of the weights; for the other clusters: 1, 3 and 7, similar weights occur across models. This finding associated with the evidence on the weights in Figure 4 for the clusters 2, 4, 5 and 6 indicates that using recursive time-varying $b_{k,ijt}$ weights within the clusters increases forecast accuracy for GDP growth relative to using equal weights. Figure S.8 also indicates that the weights within clusters are much more volatile than the cluster common component, indicating that individual model performances change over time even if information in a given clusters is stable.

Evidence is similar for the GDP deflator and employment, but this finding is less clear for bond returns. For this variable, MDCEW5 forecasts accurately. Also note that cluster 3, which includes the 3-month Treasury Bills, has the lowest weight in Figure 4. The explanation appears to be that the returns on the 3-month Treasury Bills are modeled with an AR model, which is probably less accurate for the series. Furthermore, the third cluster also contains stock prices and exchange rates that are different from other series with very low persistence and high volatility, making our combination to interpret this cluster more like a noisy component.

We conclude that the logistic-normal weights contain relevant signals about the importance of the forecasting performance of the models grouped in the clusters. Some clusters receive large weight while others have only little weight. Such a pattern may vary over long time periods. This may lead to the construction of alternative model combinations for more accurate out-of-sample forecasting and it is an interesting line of research to pursue.

Forecasting recession probabilities

As final exercise, we apply our combined forecasts to estimate turning

points and economic downturns. Following the idea in the Survey of Professional Forecasters (SPF) where individual economists are asked to report the probability of a decline in the level of real GDP in the current quarter and the following four quarters, we use our combination scheme to study the probability of negative growth (i.e., GDP growth forecast below 0) from the first to fifth ahead quarter. This measure appears to be a good estimate of recession probabilities. Figure 5 plots the recursive probabilities of negative growth in the first, third and fifth quarters from our combination scheme and compares it to ex-post NBER recession dates. The combination succeeds to forecast the 90's and the recent US Financial Crisis, even if the latter one is called with a small delay. In both cases, the recovery is well forecasted and our probabilities of negative growth substantially decrease in the last quarter of NBER recessions or at the maximum in the following one. But there is more uncertainty in the early 2000 recession, and one quarter ahead and five quarter ahead probabilities are always below 0.5. Apart from our model based forecasts, expert commentators also doubted on the definition of that recession.

Comparing the three horizons, the one quarter ahead seems the more timely and precise; the three and five quarters ahead never reach 100% probability of negative growth, confirming how difficult is to forecast a recession well ahead. But in the last year of data, the probabilities increase substantially, even if a NBER recession didn't realize at the end. Our longer forecasts gave a large probability of double-dip recession in 2011, supporting the debate on the fear of such an event at the time, see Shiller (2010). However, when new information came out and horizons shortened, forecasts also changed.

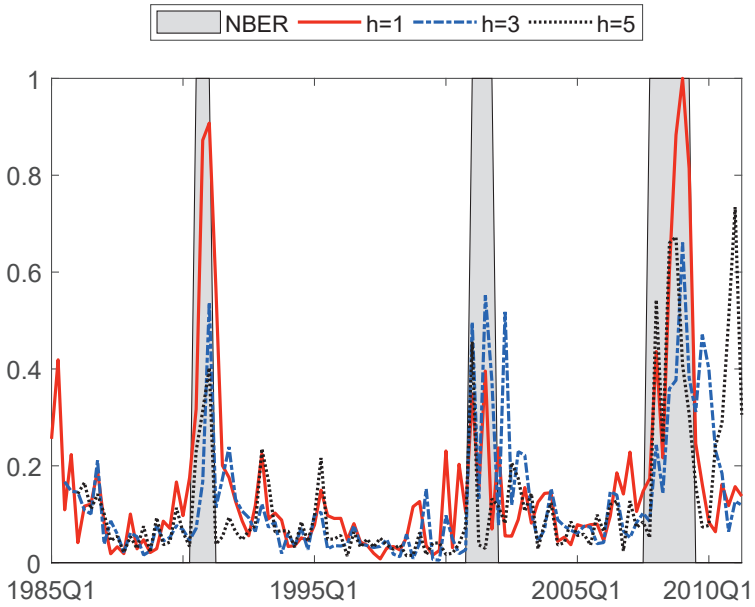


Figure 5: One quarter ahead, three quarters ahead and five quarter ahead probabilities over time of negative quarterly growth given by the the combination approach and ex-post NBER recession dates.

4 Conclusions

We propose in this paper a flexible Bayesian parametric modelling approach for the construction of combinations of forecast densities with dynamic learning that can deal with large data sets in economics and finance. The approach is based on clustering the set of forecast densities in mutually exclusive subsets and on a hierarchical specification of the combination weights. This modelling strategy reduces the dimension of the parameter and latent spaces and leads to a more parsimonious combination model. We provide several theoretical properties of the weights and propose the implementation of efficient and fast parallel clustering and sequential combination algorithms for the estimation of several features of the combined forecast densities.

We applied the methodology to large financial and macro data sets and find substantial gains in point and density forecasting for stock returns and four key macro variables. In the financial application, we show how 7000 forecast densities based on US individual stocks can be combined to replicate the daily Standard & Poor 500 (S&P500) index return accurately. Evidence obtained on the dynamic patterns of the cluster weights provide valuable signals which may be used for improved modelling and effective financial strategies. Forecasts of the economic value of tail events like Value-at-Risk are more accurate using combined forecast densities with dynamic learning than basic benchmarks like Random Walks.

In the macroeconomic exercise, we show that combining model forecasts for a set of joint variables with cluster-based weights increases forecast accuracy substantially; weights across clusters are very stable over time and horizons, with an important exception for inflation at longer horizons. Furthermore, weights within clusters are very volatile, indicating that individual model performances are more unstable, strengthening the use of density combinations. The combined forecast densities give also accurate estimates of recession probabilities over the data period considered.

The line of research presented in this paper can be extended in several directions. For example, the cluster-based weights contain relevant signals about the importance of the forecasting performance of each of the models used in the these clusters. Some clusters have a substantial weight while others have only little weight and such a pattern may vary over long time periods. This may lead to the construction of alternative model combinations for more accurate out-of-sample forecasting and improved policy analysis. Finally, we emphasise a potential fruitful connection between our approach and research in the field of

dynamic portfolio allocation, see Bãstürk et al. (2019).

References

Aastveit, K. A., Mitchell, J., Ravazzolo, F., and van Dijk, H. K. (2019). The evolution of forecast density combinations in economics. In *Oxford Research Encyclopedia of Economics and Finance*. Oxford University Press.

Aastveit, K. A., Ravazzolo, F., and van Dijk, H. K. (2018). Combined Density Nowcasting in an Uncertain Economic Environment. *Journal of Business Economics & Statistics*, 36:131–145.

Aitchinson, J. (1982). The Statistical Analysis of Compositional Data. *Journal of the Royal Statistical Society Series, Series B*, 44:139–177.

Aitchinson, J. (1986). *The Statistical Analysis of Compositional Data*. Chapman & Hall, London.

Aitchinson, J. (1992). On Criteria for Measures of Compositional Difference. *Mathematical Geology*, 24:365–379.

Aitchinson, J. and Shen, S. M. (1980). Logistic-normal Distributions: Some Properties and Uses. *Biometrika*, 67:261–272.

Bãstürk, N., Borowska, A., Grassi, S., Hoogerheide, L., and van Dijk, H. K. (2019). Forecast density combinations of dynamic models and data driven portfolio strategies. *Journal of Econometrics*, page In press.

Bańbura, M., Giannone, D., and Reichlin, L. (2010). Large Bayesian Vector Auto Regressions. *Journal of Applied Econometrics*, 25:71–92.

Billheimer, D., Guttorp, P., and Fagan, W. F. (2001). Statistical Interpretation of Species Composition. *Journal of the America Statistical Association*, 96:1205–1214.

Billio, M., Casarin, R., Ravazzolo, F., and van Dijk, H. K. (2013). Time-varying Combinations of Predictive Densities using Nonlinear Filtering. *Journal of Econometrics*, 177:213–232.

Casarin, R., Grassi, S., Ravazzolo, F., and van Dijk, H. K. (2015). Parallel Sequential Monte Carlo for Efficient Density Combination: the DeCo Matlab Toolbox. *Jounal of Statistical Software*, 68.

Casarin, R., Grassi, S., Ravazzolo, F., and Van Dijk, H. K. (2017). Dynamic Predictive Density Combinations for Large Data Sets in Economics and Finance. Technical Report 15–084/III, Tinbergen Institute. Working Paper.

Choi, H. and Varian, H. (2012). Predicting the Present with Google Trends. *Economic Record*, 88:2–9.

Einav, L. and Levin, J. (2014). Economics in The Age of Big Data. *Science*, 346:715–718.

Geweke, J. and Durham, G. (2012). Massively Parallel Sequential Monte Carlo for Bayesian Inference. Working papers, National Bureau of Economic Research, Inc.

Geweke, J. and Keane, M. (2007). Smoothly Mixing Regressions. *Journal of Econometrics*, 138:252–290.

Glosten, L. R., Jagannathan, R., and Runkle, D. E. (1993). On the Relation between the Expected Value and the Volatility of the Nominal Excess Return on Stocks. *Journal of Finance*, 48:1779–1801.

Gneiting, T. and Raftery, A. E. (2007). Strictly Proper Scoring Rules, Prediction, and Estimation. *Journal of the American Statistical Association*, 102:359–378.

Granger, C. W. J. (1998). Extracting Information from Mega-Panels and High-Frequency Data. *Statistica Neerlandica*, 52:258–272.

Hotelling, H. (1936). Relations Between Two Sets of Variates. *Biometrika*, 28:321–377.

Jacobs, R. A., Jordan, M. I., Nowlan, S. J., and Hinton, G. E. (1991). Adaptive Mixtures of Local Experts. *Journal of Neural Computation*, 3:79–87.

Jordan, M. I. and Jacobs, R. A. (1994). Hierarchical Mixtures of Experts and the EM Algorithm. *Journal Neural Computation*, 6:181–214.

Jordan, M. I. and Xu, L. (1995). Convergence Results for the EM Approach to Mixtures of Experts Architectures. *Neural Networks*, 8:1409–1431.

Jorion, P. (2006). *Value at Risk: The New Benchmark for Managing Financial Risk*. McGraw-Hill, New York.

Koop, G. (2003). *Bayesian Econometrics*. John Wiley and Sons.

Koop, G. and Korobilis, D. (2009). Bayesian Multivariate Time Series Methods for Empirical Macroeconomics. *Foundations and Trends in Econometrics*, 3:267–358.

Koop, G. and Korobilis, D. (2013). Large time-varying Parameter VARs. *Journal of Econometrics*, 177:185–198.

McAlinn, K. and West, M. (2018). Dynamic Bayesian Predictive Synthesis in Time Series Forecasting. *Journal of Econometrics*, forthcoming.

Mitchell, J. and Hall, S. G. (2005). Evaluating, Comparing and Combining Density Forecasts Using the KLIC with an Application to the Bank of England and NIESER "Fan" Charts of Inflation. *Oxford Bulletin of Economics and Statistics*, 67:995–1033.

Norets, A. (2010). Approximation of conditional densities by smooth mixtures of regressions. *Annals of statistics*, 38:1733–1766.

Peng, F., Jacobs, R. A., and Tanner, M. A. (1996). Bayesian inference in mixtures-of-experts and hierarchical mixtures-of-experts models with an application to speech recognition. *Journal of the American Statistical Association*, 91:953–960.

Shiller, R. (May 15, 2010). Fear of a double dip could cause one. *The New York Times*.

Stock, J. H. and Watson, W. M. (1999). Forecasting inflation. *Journal of Monetary Economics*, 44:293–335.

Stock, J. H. and Watson, W. M. (2002). Forecasting using principal components from a large number of predictors. *Journal of American Statistical Association*, 97:1167–1179.

Stock, J. H. and Watson, W. M. (2005). Implications of dynamic factor models for VAR analysis. Technical report, NBER Working Paper No. 11467.

Stock, J. H. and Watson, W. M. (2012). Disentangling the channels of the 2007-09 recession. *Brookings Papers on Economic Activity*, pages 81–156, Spring.

Stock, J. H. and Watson, W. M. (2014). Estimating turning points using large data sets. *Journal of Econometris*, 178:368–381.

Varian, H. (2014). Machine learning: New tricks for econometrics. *Journal of Economics Perspectives*, 28:3–28.

Varian, H. and Scott, S. (2014). Predicting the present with bayesian structural time series. *International Journal of Mathematical Modelling and Numerical Optimisation*, 5:4–23.

Villani, M., Kohn, R., and Giordani, P. (2009). Regression density estimation using smooth adaptive Gaussian mixtures. *Journal of Econometrics*, 153:155–173.

Wood, S. A., Jiang, W., and Tanner, M. (2002). Bayesian mixture of splines for spatially adaptive nonparametric regression. *Biometrika*, 89:513–528.

	h=1			h=3			h=5		
	PE	LS	CRPS	PE	LS	CRPS	PE	LS	CRPS
					RGDP				
AR	0.647	-1.002	0.492	0.671	-1.007	0.501	0.682	-1.009	0.506
BDFM	0.649	-1.091	0.382**	0.654	-1.138	0.388**	0.655	-1.099	0.388**
UDCEW5	0.644	-0.869	0.333**	0.657*	-0.900	0.341**	0.655*	-0.912	0.343**
MDCEW5	0.630	-0.928	0.326**	0.638*	-0.924	0.330**	0.636*	-0.844	0.324**
UDCLS5	0.773	-1.306	0.464	0.687	-1.339	0.446**	0.715	-1.380	0.481
MDCLS5	0.725	-1.145	0.505	0.581**	-1.041	0.340**	0.557*	-1.005	0.358**
UDCEW7	0.649	-0.875	0.334**	0.655	-0.889	0.337**	0.657*	-0.891	0.338**
MDCEW7	0.642	-0.979	0.334**	0.652*	-1.016	0.342*	0.654*	-1.009	0.342**
UDCLS7	0.646	-0.868*	0.332**	0.650*	-0.918	0.341**	0.657*	-0.914	0.342**
MDCLS7	**0.596***	**-0.586****	**0.275****	**0.607****	**-0.632****	**0.288****	**0.610****	**-0.634****	**0.286****
					GDP deflator				
AR	0.220	-0.933	0.356	0.206	-0.932	0.358	0.208	-0.932	0.361
BDFM	0.220	-0.584**	0.123*	0.206	-0.329**	0.115	0.208	-0.267	0.116
UDCEW5	0.230	-0.429	0.169	0.212	-0.422	0.165	0.213	-0.426	0.166
MDCEW5	0.204	-0.053	0.110*	0.203	-0.234	0.114	0.204	-0.194	0.113
UDCLS5	0.485	-1.085	0.354	0.259	-0.873	0.250	0.228	-0.892	0.252
MDCLS5	0.291	-0.280	0.309	0.143	0.031	0.125**	0.159	-0.226	0.147*
UDCEW7	0.223	-0.425**	0.166**	0.207	-0.416	0.163	0.210	-0.416	0.164
MDCEW7	0.208	-0.214**	0.115**	0.197*	-0.172**	0.109**	0.199	-0.200	0.111
UDCLS7	0.235	-0.507**	0.179**	0.224	-0.514	0.179	0.214	-0.475	0.171
MDCLS7	**0.197**	**0.436****	**0.098****	**0.165**	**0.571***	**0.083***	**0.175**	**0.495**	**0.088**
					3-month Treasury Bills				
AR	0.569	-1.058	0.363	0.518	-1.038	0.343	0.545	-1.041	0.358
BDFM	0.553*	-1.190	0.359	0.516	-1.092	0.392	0.517	-1.089	0.401
UDCEW5	0.519	-0.778**	0.288**	0.509	-0.772**	0.283	0.525	-0.791**	0.292*
MDCEW5	0.517**	-0.764**	0.285**	0.502*	**-0.749****	**0.276****	0.505**	-0.751**	0.278**
UDCLS5	0.740	-1.254	0.448	0.532	-1.210	0.381	0.584	-1.286	0.424
MDCLS5	0.710	-1.322	0.491	0.491**	-1.143	0.346	0.572**	-1.196	0.378
UDCEW7	0.525	-0.783**	0.289*	0.514	-0.768**	0.284*	0.522	-0.786**	0.289*
MDCEW7	0.526	-0.775**	0.289*	0.515	-0.761**	0.283*	0.513	-0.766**	0.283*
UDCLS7	0.512	-0.773**	0.284*	0.514	-0.770**	0.284*	0.521	-0.793**	0.289*
MDCLS7	**0.488****	**-0.725****	**0.270****	**0.515****	-0.755**	0.283	**0.496****	**-0.736****	**0.275****
					Employment				
AR	0.564	-0.995	0.447	0.597	-1.003	0.460	0.622	-1.009	0.468
BDFM	0.573	-1.064	0.336**	0.576	-1.192	0.333	0.582	-1.892	0.336
UDCEW5	0.585**	-0.906**	0.308**	0.579	-0.955**	0.305**	0.587	-0.951**	0.311**
MDCEW5	0.541**	-0.926**	0.277**	0.558	-0.917**	0.285**	0.571**	-0.790**	0.294**
UDCLS5	0.752	-1.301	0.456	0.565	-1.305	0.426	0.628	-1.335	0.438
MDCLS5	0.654	-1.180	0.568	0.487	-1.010	0.338	0.569	-1.076	0.360
UDCEW7	0.535**	-0.801**	0.283**	0.570	-0.854**	0.298**	0.583*	-0.881**	0.306**
MDCEW7	0.523**	-0.735**	0.266**	0.565	-0.827**	0.288**	0.578*	-0.885**	0.297**
UDCLS7	0.552**	-0.767**	0.289**	0.562	-0.849**	0.302**	0.588*	-0.895**	0.313**
MDCLS7	**0.516****	**-0.452****	**0.236****	**0.507**	**-0.479****	**0.237****	**0.560****	**-0.680****	**0.275****

Table 3: Forecasting results for $h = 1, 3, 5$ steps ahead. For all the series: root mean square forecast error (PE), logarithmic score (LS) and the continuous rank probability score (CRPS). Bold numbers indicate the best statistic for each horizon and loss function. One or two asterisks indicate that differences in accuracy versus the AR benchmark are credibly different from zero at 5%, and 1%, respectively, using the Diebold-Mariano t-statistic for equal loss. The underlying p-values are based on t-statistics computed with a serial correlation-robust variance, using the pre-whitened quadratic spectral estimator of Andrews and Monahan (1992).

Supplementary Material for "Forecast Density Combinations with Dynamic Learning for Large Data Sets in Economics and Finance"

Roberto Casarin[†] Stefano Grassi[‡]
Francesco Ravazzolo[§] Herman K. van Dijk[¶]

[†]University Ca' Foscari of Venice

[‡]University of Rome 'Tor Vergata'

[§]Free University of Bozen

[¶]Tinbergen Institute, Erasmus University Rotterdam and Norges Bank

March 2019

S.1 Proofs

S.1.1 Proof of Proposition 2.1

The marginal forecast density of the variable of interest y_t is obtained by integrating the joint density $f(y_t, \tilde{\boldsymbol{y}}_t|\mathfrak{I}_t, \sigma^2_{1t}, \ldots, \sigma^2_{nt})$ with respect to the n forecast random variables collected in the vector $\tilde{\boldsymbol{y}}_t$. This joint density is the product of the conditional density of y_t given $\tilde{\boldsymbol{y}}_t, \sigma^2_{1t}, \ldots, \sigma^2_{nt}$ and the marginal multivariate density of $\tilde{\boldsymbol{y}}_t$. For convenience, assume that this latter density is equal to the product of the individual densities. This gives:

$$f(y_t|\mathfrak{I}_t, \sigma^2_{1t}, \ldots, \sigma^2_{nt}) = \int_{\mathbb{R}} f(y_t|\tilde{\boldsymbol{y}}_t, \sigma^2_{1t}, \ldots, \sigma^2_{nt}) \prod_{j=1}^{n} f(\tilde{y}_{jt}|\mathfrak{I}_{jt}) d\tilde{y}_{jt} \tag{S.1}$$

Next, specify the conditional density as a finite mixture of normal combination densities which yields:

$$f(y_t|\mathfrak{I}_t, \sigma^2_{1t}, \ldots, \sigma^2_{nt}) = \int_{\mathbb{R}} \sum_{i=1}^{n} w_{it} \mathcal{N}(y_t|\tilde{y}_{it}, \sigma^2_{it}) \prod_{j=1}^{n} f(\tilde{y}_{jt}|\mathfrak{I}_{jt}) d\tilde{y}_{jt} \tag{S.2}$$

On the condition that integrals and summations exist, the order of integration is changed. Using the property that all cases where i is not equal to j can be ignored, one obtains

$$f(y_t|\mathfrak{I}_t, \sigma^2_{1t}, \ldots, \sigma^2_{nt}) = \sum_{i=1}^{n} w_{it} \int_{\mathbb{R}} \mathcal{N}(y_t|\tilde{y}_{it}, \sigma^2_{it}) f(\tilde{y}_{it}|\mathfrak{I}_{it}) d\tilde{y}_{it} \tag{S.3}$$

Now, by letting $\sigma_{it}^2 \to 0$ for all $i = 1, \ldots, n$, one has that $f(y_t|\mathfrak{I}_t, \sigma_{1t}^2, \ldots, \sigma_{nt}^2)$ converges to

$$f(y_t|\mathfrak{I}_t) = \sum_{i=1}^{n} w_{it} \int_{\mathbb{R}} \delta_{\tilde{y}_{it}}(y_t) f(\tilde{y}_{it}|\mathfrak{I}_{it}) d\tilde{y}_{it} = \sum_{i=1}^{n} w_{it} f(y_t|\mathfrak{I}_{it}). \tag{S.4}$$

S.1.2 Proof of Proposition 2.2

The proof consists of three main steps.

Start with the logistic transformation of x_{it} and define:

$$w_{it} = \frac{\exp(x_{it})}{\sum_{i=1}^{n} \exp(x_{it})} \qquad i = 1, 2, \ldots, n. \tag{S.5}$$

Next, divide the numerator and denominator of the right hand side of the equation above by $\exp(x_{nt})$, which yields:

$$w_{it} = \frac{\exp(x_{it} - x_{nt})}{\sum_{i=1}^{n} \exp(x_{it} - x_{nt})} \qquad i = 1, 2, \ldots, n-1, \tag{S.6}$$

where $w_{nt} = 1 - \sum_{i=1}^{n-1} w_{it}$, and define the *auxiliary* random vector $\boldsymbol{q_t}$ as $\boldsymbol{x_t}$ in deviation from it's last value x_{nt}. Then one has:

$$\boldsymbol{q_t} = \begin{pmatrix} x_{1t} - x_{nt} \\ \vdots \\ x_{(n-1)t} - x_{nt} \end{pmatrix} = \boldsymbol{D}\boldsymbol{x_t}. \tag{S.7}$$

The $(n-1) \times n$ matrix $\boldsymbol{D}$ is given by $\boldsymbol{D} = (\boldsymbol{I}_{n-1}| - \boldsymbol{\iota}_{n-1})$, with I_{n-1} equal to the $(n-1) \times (n-1)$ identity matrix, and ι_{n-1} is the $(n-1) \times 1$ vector containing only ones and there is singularity. It is seen that $\tilde{\boldsymbol{w}}_{\boldsymbol{t}} = g(\boldsymbol{q_t})$, where $g(\cdot)$ is a

one-to-one or bijective function.

Using $\boldsymbol{x}_t = \boldsymbol{B}_t \boldsymbol{v}_t$ and $\boldsymbol{v}_t = \boldsymbol{v}_{t-1} + \boldsymbol{\eta}_t, \quad \boldsymbol{\eta}_t \overset{iid}{\sim} \mathcal{N}_m(\mathbf{0}_m, \boldsymbol{\Sigma})$ it follows that

$$\boldsymbol{q_t} = \boldsymbol{D}\boldsymbol{B}_t \boldsymbol{v}_t \sim \mathcal{N}_{n-1}(\boldsymbol{D}\boldsymbol{B}_t \boldsymbol{v}_{t-1}, \boldsymbol{D}\boldsymbol{B}_t \boldsymbol{\Sigma} \boldsymbol{D}' \boldsymbol{B}_t'). \tag{S.8}$$

Second, the inverse transformation $\boldsymbol{q_t} = g^{-1}(\tilde{\mathbf{w}}_\mathbf{t})$ is given as;

$$q_{it} = \log\left(\frac{w_{it}}{w_{nt}}\right) = \log(w_{it}) - \log\left(1 - \sum_{i=1}^{n-1} w_{it}\right) \qquad i = 1, 2, \ldots, n-1, \tag{S.9}$$

with Jacobian matrix

$$\begin{aligned}
\frac{\partial \boldsymbol{q_t}}{\partial \tilde{\mathbf{w}}_\mathbf{t}} &= \begin{pmatrix} w_{1t}^{-1} & 0 & \cdots & 0 \\ 0 & w_{2t}^{-2} & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & \cdots & 0 & w_{(n-1)t}^{-1} \end{pmatrix} + \left(1 - \sum_{i=1}^{n-1} w_{it}\right)^{-1} \begin{pmatrix} 1 & 1 & \cdots & 1 \\ 1 & 1 & & \vdots \\ \vdots & & \ddots & 1 \\ 1 & \cdots & 1 & 1 \end{pmatrix} \\
&= \begin{pmatrix} w_{1t}^{-1} & 0 & \cdots & 0 \\ 0 & w_{2t}^{-2} & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & \cdots & 0 & w_{(n-1)t}^{-1} \end{pmatrix} + w_{nt}^{-1} \times \boldsymbol{\iota}_{(n-1)\times(n-1)},
\end{aligned} \tag{S.10}$$

where $\boldsymbol{\iota}_{(n-1)\times(n-1)}$ is the $(n-1) \times (n-1)$ matrix containing only ones.

The determinant of $\frac{\partial \boldsymbol{q_t}}{\partial \tilde{\mathbf{w}}_\mathbf{t}}$ is

$$\left|\frac{\partial \boldsymbol{q_t}}{\partial \tilde{\mathbf{w}}_\mathbf{t}}\right| = \prod_{i=1}^{n} w_{it}^{-1}, \tag{S.11}$$

where use is made the following determinant rule[1]

$$|\boldsymbol{A} + \boldsymbol{x}\boldsymbol{y}'| = |\boldsymbol{A}| \times (1 + \boldsymbol{y}'\boldsymbol{A}^{-1}\boldsymbol{x}),$$

with $\boldsymbol{x} = w_n^{-1} \times \boldsymbol{\iota}_{(n-1)}$, $\boldsymbol{y} = \boldsymbol{\iota}_{(n-1)}$ and

$$\boldsymbol{A} = \begin{pmatrix} w_1^{-1} & 0 & \cdots & 0 \\ 0 & w_2^{-2} & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & \cdots & 0 & w_{n-1}^{-1} \end{pmatrix}, \tag{S.12}$$

where

$$|\boldsymbol{A}| = \prod_{i=1}^{n-1} w_i^{-1} \tag{S.13}$$

(S.14)

$$\boldsymbol{A}^{-1} = \begin{pmatrix} w_1 & 0 & \cdots & 0 \\ 0 & w_2 & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & \cdots & 0 & w_{n-1} \end{pmatrix} \tag{S.15}$$

(S.16)

$$\boldsymbol{y}'\boldsymbol{A}^{-1}\boldsymbol{x} = w_n^{-1} \sum_{i=1}^{n-1} w_i = \frac{1 - w_n}{w_n} \tag{S.17}$$

(S.18)

$$1 + \boldsymbol{y}'\boldsymbol{A}^{-1}\boldsymbol{x} = 1 + \frac{1 - w_n}{w_n} = \frac{w_n + 1 - w_n}{w_n} = \frac{1}{w_n}, \tag{S.19}$$

where pre- and post-multiplying by $\boldsymbol{\iota}'_{n-1}$ and $\boldsymbol{\iota}_{n-1}$ obviously means that one can

[1] see **?** and for notatiaonal convenience the subindex t has been omitted

compute the sum of all elements of the matrix $\boldsymbol{A}^{-1}$ (where this sum is here equal to $\sum_{i=1}^{n-1} w_i$), and where $\sum_{i=1}^{n-1} w_i = 1 - w_n$, so that it follows that:

$$|\boldsymbol{A}| \times (1 + \boldsymbol{y}'\boldsymbol{A}^{-1}\boldsymbol{x}) = \prod_{i=1}^{n} w_i^{-1}.$$

Note: for $n = 2$ one has

$$\frac{\partial q}{\partial \tilde{z}} = w_1^{-1} + (1 - w_1)^{-1} = \frac{1}{w_1(1 - w_1)} = \frac{1}{w_1 w_2} = \prod_{i=1}^{2} w_i^{-1}.$$

For $n = 3$ one has

$$\frac{\partial \boldsymbol{q}}{\partial \tilde{\mathbf{z}}} = \begin{pmatrix} w_1^{-1} + w_3^{-1} & w_3^{-1} \\ \\ w_3^{-1} & w_2^{-1} + w_3^{-1} \end{pmatrix}$$

with

$$\begin{aligned} \left| \frac{\partial \boldsymbol{q}}{\partial \tilde{\mathbf{z}}} \right| &= (w_1^{-1} + w_3^{-1})(w_2^{-1} + w_3^{-1}) - w_3^{-2} = w_1^{-1}w_2^{-1} + w_1^{-1}w_3^{-1} + w_2^{-1}w_3^{-1} \\ &= \frac{w_3}{w_1 w_2 w_3} + \frac{w_2}{w_1 w_2 w_3} + \frac{w_1}{w_1 w_2 w_3} = \frac{w_1 + w_2 + w_3}{w_1 w_2 w_3} \\ \\ &= \frac{1}{w_1 w_2 w_3} = \prod_{i=1}^{3} w_i^{-1}, \end{aligned}$$

since $w_1 + w_2 + w_3 = 1$.

Third, given that $\boldsymbol{q_t}$ has the multivariate normal density function:

$$f(\boldsymbol{q_t}|\boldsymbol{DB}_t\boldsymbol{v}_{t-1}, \boldsymbol{DB}_t\boldsymbol{\Sigma}\boldsymbol{D}'\boldsymbol{B}'_t) = (2\pi)^{-(n-1)/2}|\boldsymbol{DB}_t\boldsymbol{\Sigma}\boldsymbol{D}'\boldsymbol{B}'_t|^{-1/2} \times \exp\left(-\frac{1}{2}\left(\boldsymbol{q_t} - \boldsymbol{DB}_t\boldsymbol{v}_{t-1}\right)'\left(\boldsymbol{DB}_t\boldsymbol{\Sigma}\boldsymbol{D}'\boldsymbol{B}'_t\right)^{-1}\left(\boldsymbol{q} - \boldsymbol{DB}_t\boldsymbol{v}_{t-1}\right)\right), \quad \text{(S.20)}$$

substitution of $\boldsymbol{q_t} = \log\left(\frac{\tilde{\mathbf{w}}_\mathbf{t}}{w_{nt}}\right)$ into (S.20) and multiplying with $\left|\frac{\partial \boldsymbol{q_t}}{\partial \tilde{\mathbf{w}}_\mathbf{t}}\right| = \prod_{i=1}^{n} w_{it}^{-1}$ yields:

$$f(\tilde{\mathbf{w}}_\mathbf{t}|\boldsymbol{DB}_t\boldsymbol{v}_{t-1}, \boldsymbol{DB}_t\boldsymbol{\Sigma}\boldsymbol{D}'\boldsymbol{B}'_t) = (2\pi)^{-(n-1)/2}|\boldsymbol{DB_t}\boldsymbol{\Sigma}\boldsymbol{D}'\boldsymbol{B}'_t|^{-1/2}\left(\prod_{i=1}^{n} w_{it}\right)^{-1} \times \exp\left(-\frac{1}{2}\left(\log\left(\frac{\tilde{\mathbf{w}}_\mathbf{t}}{w_{nt}}\right) - \boldsymbol{DB}_t\boldsymbol{v}_{t-1}\right)'\left(\boldsymbol{DB}_t\boldsymbol{\Sigma}\boldsymbol{D}'\boldsymbol{B}'_t\right)^{-1}\left(\log\left(\frac{\tilde{\mathbf{w}}_\mathbf{t}}{w_{nt}}\right) - \boldsymbol{DB}_t\boldsymbol{v}_{t-1}\right)\right) \quad \text{(S.21)}$$

Q.E.D.

S.2 Algorithmic details and practical user guide

The analytical solution of the optimal filtering problem is generally not known. Also, the cluster-based mapping requires the solution of an optimisation problem which is not available in analytical form. Thus, we apply a sequential numerical approximation of the two problems and use an algorithms that at time t iterates over the following two steps:

1) Parallel sequential clustering in order to determine the allocation matrix $\Xi_t = (\boldsymbol{\xi}_{1t}, \ldots, \boldsymbol{\xi}_{mt})$, with $\boldsymbol{\xi}_{jt} = (\xi_{j1t}, \ldots, \xi_{jnt})'$, $j = 1, \ldots, m$, the vector of

allocation variables $\xi_{jit} \in \{0, 1\}$, see the paper, Section 2.2.

2) Sequential Monte Carlo approximation that involve the parameters of the combination models and the latent weights. Let $\boldsymbol{\theta}_t \in \Theta$ be the parameter vector of the combination model, that is $\boldsymbol{\theta}_t = (\log \sigma^2_{c1t}, \ldots, \log \sigma^2_{cmt})$. Let $\mathbf{w}'_t = (\mathbf{w}'_{1t}, \ldots, \mathbf{w}_{nt})$ the vector of weights.

The details of the algorithms are given in the following subsections.

S.2.1 Sequential Clustering

The number of cluster in the K-means algorithm depends on the problem at hand and has to be set by the researcher. Practical details on how we proceeded in the finance and macroeconomic cases are given below in the practical user guide in S.1.4. We start here with a brief exposition on the algorithmic steps.

Let $\mathbf{c}_{j0}$, $j = 1, \ldots, m$, an initial set of random points and let $\mathbf{c}_{jt}$, $j = 1, \ldots, m$ be the centroids, defined as

$$\mathbf{c}_{jt} = \frac{1}{n_{jt}} \sum_{i \in N_{jt}} \boldsymbol{\psi}_{it},$$

where n_{jt} and N_{jt} have been define in Section 2.2 of the main text. At time $t+1$ a new set of observations $\boldsymbol{\psi}_{it+1} \in \mathbb{R}^d$, $i = 1, \ldots, n$ is assigned to the different m groups of observations based on the minimum distance, such as the Euclidean distance, $||\cdot||$, between the observations and the centroids $\mathbf{c}_{jt} \in \mathbb{R}^d$, $j = 1, \ldots, m$. Assume $j_i = \arg\min\{j = 1, \ldots, m| \, ||\boldsymbol{\psi}_{it} - \mathbf{c}_{jt}||\}$, $i = 1, \ldots, n$, then the allocation variable ξ_{ijt} is equal to 1 if $j = j_i$ and 0 otherwise and the centroids are updated

as follows:

$$\mathbf{c}_{jt+1} = \mathbf{c}_{jt} + \lambda_t(\mathbf{m}_{jt+1} - \mathbf{c}_{jt}) \tag{S.22}$$

where

$$\mathbf{m}_{jt+1} = \frac{1}{n_{jt+1}} \sum_{i \in N_{jt+1}} \boldsymbol{\psi}_{it} \tag{S.23}$$

and $\lambda_t \in [0, 1]$. Note that the choice $\lambda_t = n_{jt+1}/(n^c_{jt}+n_{jt+1})$, with $n^c_{jt} = \sum_{s=1}^{t} n_{js}$, implies a sequential clustering with forgetting driven by the processing of the blocks of observations. In the application we fix $\lambda_t = 0.99$.

S.2.2 Parallel sequential clustering

The parallel implementation of the k-means algorithm can be described as follows. Assume, for simplicity, the n data points can be split in P subsets, $N_p = \{(p-1)n_p + 1, \ldots, pn_p\}$, $p = 1 \ldots, P$, with the equal number of elements n_P. P is chosen according to the number of available cores.

1) Assign P sets of n_P data points to different cores.

2) For each core p, $p = 1, \ldots, P$

   2a) find $j_i = \arg\min\{j = 1, \ldots, m | \, ||\boldsymbol{\psi}_{it} - \mathbf{c}_{jt}||\}$, for each observation $i \in N_p$ assigned to the core p.

   2b) find the local centroid updates $\mathbf{m}_{p,jt+1}$, $j = 1, \ldots, m$

3) Find the global centroid updates $\mathbf{m}_{jt+1} = 1/P \sum_{p=1}^{P} \mathbf{m}_{p,jt+1}$, $j = 1, \ldots, m$

4) Update the centroids as in Eq. (S.22).

The k-means algorithm is fully parallel in point 2). In point 3) the parallelization is used to speed up the sum. This can be done with multiple CPU of in GPU context as we do in this paper.

S.2.3 Sequential Monte Carlo method using weights and model parameters of a measurement equation specified as mixture

As regards the sequential filtering we apply sequential Monte Carlo as introduced in Billio et al. (2013) and implemented in Casarin et al. (2015).
Following **?**, **?**, and **?**, we define the augmented state vector $\mathbf{w}_t^\theta = (\mathbf{w}_t, \boldsymbol{\theta}_t) \in \mathcal{Z}$, and the augmented state space $\mathcal{W} = \mathbb{S}^{n-1} \times \Theta$. Our model can be written in the augmented state space form where the measurement and transition densities are given in Section 2 of the paper and repeated here as

$$f(y_t|\mathbf{w}_t^\theta, \tilde{\mathbf{y}}_t) \propto \sum_{i=1}^{n} w_{it} \mathcal{N}\left(\tilde{y}_{it}, \sigma_{it}^2\right) \tag{S.24}$$

$$f(\tilde{\mathbf{w}}_t|\boldsymbol{\theta}_t, \mathbf{w}_{t-1}^\theta, \mathbf{y}_{1:t-1}, \tilde{\mathbf{y}}_{1:t-1}) \propto \mathcal{L}_{n-1}\left(\boldsymbol{D}\boldsymbol{B}_t\boldsymbol{v}_{t-1}, \boldsymbol{D}\boldsymbol{B}_t\boldsymbol{\Sigma}\boldsymbol{B'}_t\boldsymbol{D'}\right) \tag{S.25}$$

where $\tilde{\boldsymbol{w}}_t = (w_{1t}, \ldots, w_{n-1,t})'$, $w_{nt} = 1 - \tilde{\boldsymbol{w}}_t'\boldsymbol{\iota}_{n-1}$

Source code The mixture approach used in Section 2 in the paper is reported in the source MATLAB code below. In line 2 we draw one of the mixtures and in line 3 permute the associated prediction with our incompleteness. For comparison purposes, the probabilistic combination approach used in Casarin et al. (2015) is reported (commented) in line 1.

Listing 1: Source code for the mixture approach.

```
% S.ytilde(i, z) = logmul(S.mOmega(i, :, z)) *...
% mXTot(S.t, :, z)' + exp(0.5 * S.mSigma(i, z))
% * randn(1, 1);
I = EmpCPU(logmul(S.mOmega(i, :, z)), 1);
S.ytilde(i, z) = mXTot(S.t, I, z)' +...
exp(0.5 * S.mSigma(i, z)) * randn(1, 1);
```

S.2.4 Practical guide for practitioners

In this subsection we provide a description on how we settled prior values and the reasoning behind our choices. We describe separately priors for model weights, model incompleteness and cluster selection.

Incompleteness and weights parameters The algorithm prior parameters are described in Table S.1.

Prior	Value	Description
κ	0.7	ESS resampling threshold $\kappa > 0$.
λ_j	0.3	Variance prior for the cluster weight dynamics in eq. (23).
σ_i^2	0.01	Variance prior for the incompletness process σ_{it}^2.

Table S.1: Prior values.

The Table reports the prior parameters of our model. In particular, we set:

1) λ_j equal to 0.3 to allow for possible large variation and uncertainty in the weights. We remember weights assume values in the interval [0,1] and a similar variance allows in few days large switches in the weights.

2) σ_i^2 equal to 0.01 for a small level of incompletness corresponding to a one standard deviation of the series.

We also refer to Billio et al. (2013) and Casarin et al. (2015) for robustness of these values.

Number of clusters The number of cluster in the K-means algorithm depends on the problem at hand and has to be set by the researcher. A good rule is to set a number of clusters that brings to a reasonable number of series in each cluster and the economic theory that the researcher what to test, e.g. number of sectors. More clusters bring to lower number of series for each unit increasing uncertainty in the estimation. On the contrary, a small number of clusters can also bias the results because it can mix series that belong to different sectors. In our empirical application we use graphical evidence to choose the number of clusters. We compare variance and degree of freedom estimates for the financial application, see Figure 1, and persistence estimates for the macroeconomic example, see Figure S.3 and Table S.5.

In the financial applications, we believe the large differences across forecasts is in the higher moments and tails behaviour. Our cluster algorithm is then applied to predicted variance and predicted degree of freedom. In Figure 1, we find that when choosing two clusters for the Nornmal GARCH(1,1) models results in average cluster variances that are more than double in cluster n2 versus cluster n1. The same ratio applies to degree of freedom for the t-GARCH(1,1) models in clusters t2 versus t1. We think forecast densities should substantially differ to maximize gains from combinations and current evidence on average cluster variance and degree of freedom goes in that direction. Adding a third cluster

resulted in a less clear pattern, with values closer among the three clusters.

In the macroeconomic application, the cluster algorithm is applied to a persistent measure, precisely the autoregressive coefficient. Therefore, the source of discrimination among clusters is the persistence of the series. In Figure S.3 and Table S.5 we compare five clusters versus seven clusters. The cluster means span well all the individual series estimates. Reducing the cluster numbers lower than five and increasing higher than six resulted in a too sparse and too dense division. We also check six clusters, but the difference with five cluster was minor and we decided not to report it.

S.3 Forecast evaluation

To measure the forecast ability of our methodology, we consider several statistics for point and density forecasts previously proposed in the literature. Assume we have n different approaches to predict the variable y.

Point forecasts. We compare point forecasts in terms of Root Mean Square Prediction Errors (RMSPE)

$$RMSPE_{i,h} = \sqrt{\frac{1}{t^*}\sum_{t=\underline{t}}^{\bar{t}} e_{i,t+h}},$$

where $t^* = \bar{t} - \underline{t} + h$, $\bar{t}$ and $\underline{t}$ denote the beginning and end of the evaluation period, and $e_{i,t+h}$ is the h-step ahead square prediction error of model i.

Density forecasts. The complete predictive densities are evaluated as follows. Let $f(y_{t+h}|\mathfrak{I}_{it})$ be a candidate density obtained from the approach i. The

Logarithmic Score (LS) is then given as:

$$LS_{i,h} = -\frac{1}{t^*}\sum_{t=\underline{t}}^{\bar{t}} \ln f(y_{t+h}|\mathfrak{I}_{it}), \tag{S.26}$$

for all i and choose the model for which this score is minimal, or, as we report in our tables and use in the learning strategies, its opposite is maximal.

We also evaluate density forecasts based on the continuous rank probability score (CRPS); see, for example, Gneiting and Raftery (2007), **?**, **?** and **?**. The CRPS for the model i measures the average absolute distance between the empirical cumulative distribution function (CDF) of y_{t+h}, which is simply a step function in y_{t+h}, and the empirical CDF that is associated with model i's predictive density:

$$\begin{aligned} \text{CRPS}_{i,t+h} &= \int_{-\infty}^{+\infty} \Big(F(z|\mathfrak{I}_{it}) - \mathbb{I}_{[y_{t+h},+\infty)}(z)\Big)^2 \,\mathrm{d}z \\ &= \mathbb{E}_t|\tilde{y}_{i,t+h} - y_{t+h}| - \frac{1}{2}\mathbb{E}_t|\tilde{y}^*_{i,t+h} - \tilde{y}'_{i,t+h}|, \end{aligned} \tag{S.27}$$

where $F(\cdot|\mathfrak{I}_{it})$ is the CDF from the predictive density $f(y_{t+h}|\mathfrak{I}_{it})$ of model i and $\tilde{y}^*_{i,t+h}$ and $\tilde{y}'_{i,t+h}$ are independent random variables with common sampling density equal to the posterior predictive density $f(y_{t+h}|\mathfrak{I}_{it})$. We report the sample average CRPS:

$$\text{CRPS}_{i,h} = -\frac{1}{t^*}\sum_{t=\underline{t}}^{\bar{t}} \text{CRPS}_{i,t+h}. \tag{S.28}$$

Smaller CRPS values imply higher precisions and, as for the log score, we report the average $\text{CRPS}_{i,h}$ for each model i in all tables.

Tail forecasts. Given that our approach produces complete predictive densities for the variable of interest, it is particularly suitable to compute tail events. We consider two statistics and an economic measure for tail events. We compute weighted averages of Gneiting and Raftery (2007) quantile scores that are based on quantile forecasts that correspond to the predictive densities from the different models, i.e.,

$$\text{QS}(\alpha, i, t) = \left(\mathtt{I}\{y_{t+1} \leqq F^{-1}(\alpha, i)\} - \alpha\right)\left(F^{-1}(\alpha|\mathfrak{I}_{it}) - y_{t+1}\right), \tag{S.29}$$

with $F^{-1}(\alpha|\mathfrak{I}_{it})$ is the 1-step ahead quantile forecast using prediction i for level $\alpha \in (0,1)$. It can be shown that integrating (S.29) over $\alpha \in (0,1)$ will result in the CRPS measure (S.27), see **?**. **?**, **?** and **?** propose to integrate weighted versions of (S.29) over α, with these weights being fixed functions of α chosen such to emphasize in the forecast evaluation a certain area of the underlying forecast density. We use a discrete approximation to this integration and use weights that emphasize both tail and the left tail of the predictive density:

$$\begin{aligned}
\text{avQS-T}_i &= \frac{1}{T-t_0-1}\sum_{s=t_0-1}^{T-1}\left(\frac{1}{99}\sum_{j=1}^{99}(2\alpha_j-1)^2\text{QS}(\alpha_j, i, s+1)\right) \\
\text{avQS-L}_{i,h} &= \frac{1}{T-t_0-1}\sum_{s=t_0-1}^{T-1}\left(\frac{1}{99}\sum_{j=1}^{99}(1-\alpha_j)^2\text{QS}(\alpha_j, i, s+1)\right)
\end{aligned} \tag{S.30}$$

where $\alpha_j = j/100$ and $\text{QS}(\alpha_j, i, s+1)$ is defined in (S.29) for a quantile j. In (S.30), avQS-T emphasizes both tails and avQS-L the left tail of the predictive density relative to the realization 1-step ahead. To study how the models perform

in the left tail prediction over time, we consider the cumulative sum of avQS-L:

$$\text{cumavQS-L}_{i,h,t} = \sum_{s=t_0-1}^{t} \text{avQS-L}_{i,h,s} \tag{S.31}$$

The most accurate model at observation t produces the lowest $\text{cumavQS-L}_{i,h,t}$.

Finally, following **?**, we apply the **?** t-tests for equality of the average loss (with loss defined as squared error, log score, or CRPS). In our tables presented below, differences in accuracy that are statistically different from zero are denoted by one, two, or three asterisks, corresponding to significance levels of 10%, 5%, and 1%, respectively. The underlying p-values are based on t-statistics computed with a serial correlation-robust variance, using the pre-whitened quadratic spectral estimator of **?**. Monte Carlo evidence in **?** and **?** indicates that, with nested models, the Diebold-Mariano test compared against normal critical values can be viewed as a somewhat conservative (conservative in the sense of tending to have size modestly below nominal size) test for equal accuracy in the finite sample. Since the AR benchmark is always one of the model in the combination schemes, we treat each combination as nesting the baseline, and we report p-values based on one-sided tests, taking the AR as the null and the combination scheme in question as the alternative.

S.4 Additional details on empirical results

S.4.1 Additional details on the financial application

Table S.2 reports the cross-section average statistics, together with statistics for the S&P500. Some series have much lower average returns than the index and

	Subcomponents			S&P500
	Lower	Median	Upper	
Average	-0.002	0.000	0.001	0.000
St dev	0.016	0.035	0.139	0.019
Skewness	-1.185	0.033	1.060	-0.175
Kurtosis	8.558	16.327	65.380	9.410
Min	-1.322	-0.286	-0.121	-0.095
Max	0.122	0.264	1.386	0.110

Table S.2: Average cross-section statistics for the 1856 individual stock daily log returns in our dataset for the sample 18 March 2002 to 31 December 2009. The columns "Lower", "Median" and "Upper" refer to the cross-section 10% lower quantile, median and 90% upper quantile of the 3712 statistics in rows, respectively. The rows "Average", "St dev", "Skewness", "Kurtosis", "Min" and "Max" refers to sample average, sample standard deviation, sample skewness, sample kurtosis, sample minimum and sample maximum statistics, respectively. The column "S&P500" reports the sample statistics for the aggregate S&P500 log returns.

volatility higher than the index up to 400 times. Heterogeneity in skewness is also very evident with the series with lowest skewness equal to -42.5 and the one with highest skewness equal to 27.3 compared to a value equal to -0.18 for the index. Finally, maximum kurtosis is 200 times higher than the index value. The inclusion in our sample of the crisis period explains such differences, with some stocks that realized enormously negative returns in 2008 and impressive positive returns in 2009.

Details of the trajectories of the weights are given in Figure S.1 by using the De Finetti or ternary diagram (see **?** and **?**).

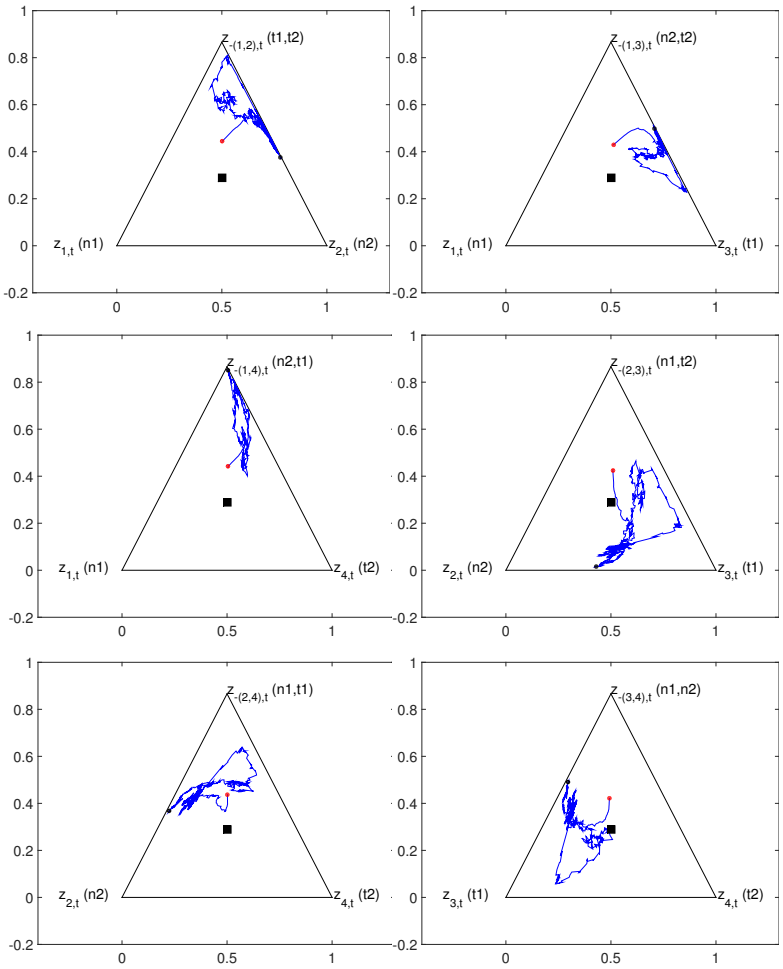


Figure S.1: De Finetti diagram for the pairwise subcomposition comparison between model weights over time. In each plot the trajectory of the ternary $(z_{it}, z_{jt}, z_{-(ij)t})$, $j > i$ (blue line), the starting point (red dot), the ending point (black dot) and the equal weight composition (square).

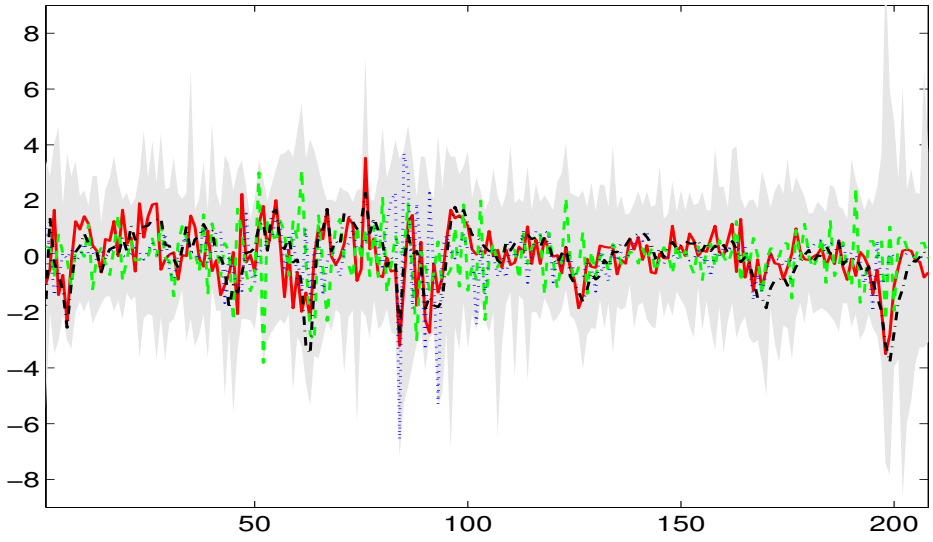


Figure S.2: Gray area: the set of series (standardised for a better graphical representation), at the monthly frequency, of the Stock and Watson dataset. Solid line: growth rate of real GDP (seasonally adjusted) for the US. Dashed line: inflation measured as the change in the GDP deflator index (seasonally adjusted). Dotted line: yields on US government 90-day T-Bills (secondary market). Dashed-dotted: total employment growth rate for private industries (seasonally adjusted).

S.4.2 Additional details on the macroeconomic application

We consider the extended Stock and Watson (2005) dataset, which includes 142 series sampled at a quarterly frequency from 1959Q1 to 2011Q2. A graphical description of the data is given in Figure S.2.

For each variable we estimate a Gaussian autoregressive model of the first order, AR(1),

$$y_{it} = \alpha_i + \beta_i y_{it-1} + \zeta_{it}, \quad \zeta_{it} \sim \mathcal{N}(0, \sigma_i^2), \tag{S.32}$$

using the first 60 observations from each series. Then we identify the clusters of parameters by applying our k-means clustering algorithm on the vectors,

$\hat{\boldsymbol{\theta}}_i = (\hat{\alpha}_i, \hat{\beta}_i, \hat{\sigma}_i^2)'$, of least square estimates of the AR(1) parameters. A detailed description of the 5 and 7 clusters is provided in Tables S.3-S.4.

The left and right columns in Fig.S.3 show the clusters of series in the parameter space. The results show substantial evidence of different time series characteristics in several groups of series. The groups are not well separated when looking at the intercept values (see Fig. S.3, first and second row). However, the groups are well separated along two directions of the parameter space, which are the one associated with the variance and the one associated with persistence parameters (Fig.S.3, last row). The differences in terms of persistence, in the different groups, is also evident from the heat maps given in Fig.S.5. Different gray levels in the two graphs show the value of the variables (horizontal axis) over time (vertical axis). The vertical red lines indicate the different clusters. One can see for example that the series in the 2nd and 4th cluster (of 5) are more persistent then the series in the clusters 1, 3 and 5 (see also Fig. S.3, bottom left). Series in cluster 1, 2 and 4 are less volatile than series in the cluster 3 and 5. This information is also summarised by the mean value of the parameter estimates for the series that belong to the same cluster. See the values in Table S.5. Looking at the composition of the predictor groups (see also Tables S.3-S.4), we find for the five clusters that:

1. The first cluster comprises capacity utilisation, employment variables, housing (building permits and new ownership started) and manufacturing variables (new orders, supplier deliveries index, inventories).

2. The second cluster contains exports, a large numbers of price indexes (e.g. prices indexes for personal consumption expenditures, and for gross

Table S.3: Predictors classification in 5 clusters (columns).

1	2	3	4	5
NAPMprodn	Exports	RGDP	Cons-Dur	Cons-Serv
CapacityUtil	PGDP	Cons	Imports	FixedInv
Emptotal	PCED	Cons-NonDur	GovFed	NonResInv
Empgdsprod	CPI-ALL	GPDInv	IPfuels	NonResInv-Struct
Empdblegds	PCED-Core	Gov	Ul5wks	NonResInv-Bequip
Empservices	CPI-Core	GovStateLoc	U5-14wks	Res.Inv
EmpTTU	PCED-DUR-HHEQ	IPconsgds	Orders(NDCapGoods)	IPtotal
Empwholesale	PCED-DUR-OTH	IPconsdble	PCED-DUR	IPproducts
EmpFIRE	PCED-NDUR	IP:consnondble	PCED-DUR-MOTOR	IPfinalprod
Avghrs	PCED-NDUR-FOOD	Empmining	PCED-NDUR-OTH	IP:buseqpt
HStartsTotal	PCED-NDUR-CLTH	EmpCPStotal	PFI-NRES	IPmatls
BuildPermits	PCED-NDUR-ENERGY	Overtimemfg	PFI-NRES-EQP	IPdblemats
HStartsNE	PCED-SERV	Umeanduration	Pimp	IP:nondblemats
HStartsMW	PCED-SERV-HOUS	U15-26wks	LaborProd	IPmfg
HStartsSouth	PCED-SERV-HOUSOP	Orders(ConsGoods)	RealCompHour	Empconst
HStartsWest	PCED-SERV-H0-ELGAS	Comspotprice(real)	3moT-bill	Empmfg
PMI	PCED-SERV-HO-OTH	OilPrice(Real)	6moT-bill	Empnondbles
NAPMnewordrs	PCED-SERV-TRAN	RealAHEgoods	5yrT-bond	Empretail
NAPMvendordel	PCED-SERV-MED	RealAHEmfg	10yrT-bond	EmpGovt
NAPMInvent	PCED-SERV-REC	UnitLaborCost	Reservesnonbor	Helpwantedindx
NAPMcomprice	PCED-SERV-OTH	Aaabond	ExrateSwitz	Helpwantedemp
Consumerexpect	PGPDI	Baabond	ExrateJapan	EmpCPSnonag
fygm10-fygm3	PFI	Exrateavg	DJIA	EmpHours
Fyaaac-fygt10	PFI-NRES-STRPrInd	ExrateUK		Uall
Fyaaac-fygt10	PFI-RES	EXrateCanada		U15pwks
	Pexp	S&P500		U27pwks
	Pgov	S&Pindust		RealAHEconst
	PgovFed	S&Pdivyield		Conscredit
	Pgovstatloc	S&PPEratio		fygm1-fygm3
	FedFunds	fygm6-fygm3		
	1yrT-bond			
	M1			
	MZM			
	M2			
	MB			
	Reservestot			
	BUSLOANS			

Table S.4: Predictors classification in 7 clusters (columns).

1	2	3	4	5	6	7
FixedInv	Cons-Serv	Empmining	IPfuels	RGDP	Exports	NAPMprodn
NonResInv	NonResInv-Bequip	CPI-ALL	PCED	Cons	Imports	CapacityUtil
NonResInv-Struct	Res.Inv	PCED-NDUR	CPI-Core	Cons-Dur	Ul5wks	Empwholesale
IPproducts	GovStateLoc	PCED-NDUR-CLTH	PCED-DUR-OTH	Cons-NonDur	Orders(NDCapGoods)	Helpwantedindx
IP:buseqpt	IPtotal	PCED-NDUR-ENERGY	PCED-SERV	GPDInv	PGDP	Avghrs
IP:nondblemats	IPfinalprod	PCED-SERV-H0-ELGAS	PCED-SERV-HOUS	Gov	PCED-NDUR-FOOD	HStartsTotal
Emptotal	IP:consnondble	FedFunds	PCED-SERV-HO-OTH	GovFed	PCED-SERV-HOUSOP	BuildPermits
Empgdsprod	IPmfg	3moT-bill	PCED-SERV-TRAN	IPconsgds	PCED-SERV-MED	HStartsNE
Empmfg	Empdblegds	6moT-bill	PCED-SERV-REC	IPconsdble	PGPDI	HStartsMW
Empnondbles	Helpwantedemp	1yrT-bond	PCED-SERV-OTH	IPmatls	PFI	HStartsSouth
Empservices	Overtimemfg	5yrT-bond	PFI-NRES-STRPrInd	IPdblemats	PFI-NRES	HStartsWest
EmpTTU	Orders(ConsGoods)	10yrT-bond	Pimp	Empconst	PFI-RES	PMI
Empretail	PCED-Core	M1	PgovFed	EmpCPStotal	Pexp	NAPMnewordrs
EmpFIRE	PFI-NRES-EQP	MZM	Pgovstatloc	U5-14wks	Pgov	NAPMvendordel
EmpGovt	Comspotprice(real)	MB	M2	U15-26wks	BUSLOANS	OilPrice(Real)
EmpCPSnonag	RealAHEconst	Reservestot		U27pwks		NAPMcomprice
EmpHours	RealCompHour	Reservesnonbor		PCED-DUR		Conscredit
Uall	UnitLaborCost	ExrateUK		PCED-DUR-MOTOR		Consumerexpect
Umeanduration	S&P500	EXrateCanada		RealAHEgoods		fygm10-fygm3
U15pwks	fygm6-fygm3	S&Pindust		RealAHEmfg		Fyaaac-fygt10
NAPMInvent		DJIA		LaborProd		Fyaaac-fygt10
PCED-DUR-HHEQ				S&Pdivyield		
PCED-NDUR-OTH						
Aaabond						
Baabond						
Exrateavg						
ExrateSwitz						
ExrateJapan						
S&PPEratio						
fygm1-fygm3						

domestic product) some money market variables (e.g. M1 and M2).

3. The third cluster includes real gross domestic product, consumption and consumption of non-durables, some industrial production indexes, and some financial market variables (e.g., S&P industrial, corporate bonds and USD - GBP exchange rate).

4. The fourth cluster includes imports, some price indexes and financials such as government debt (3- and 6-months T-bills and 5- and 10-years T-bonds), stocks and exchange rates.

5. The fifth cluster mainly includes investments, industrial production indexes (total and many sector indexes), and employment.

Evidence is similar for the seven clusters.

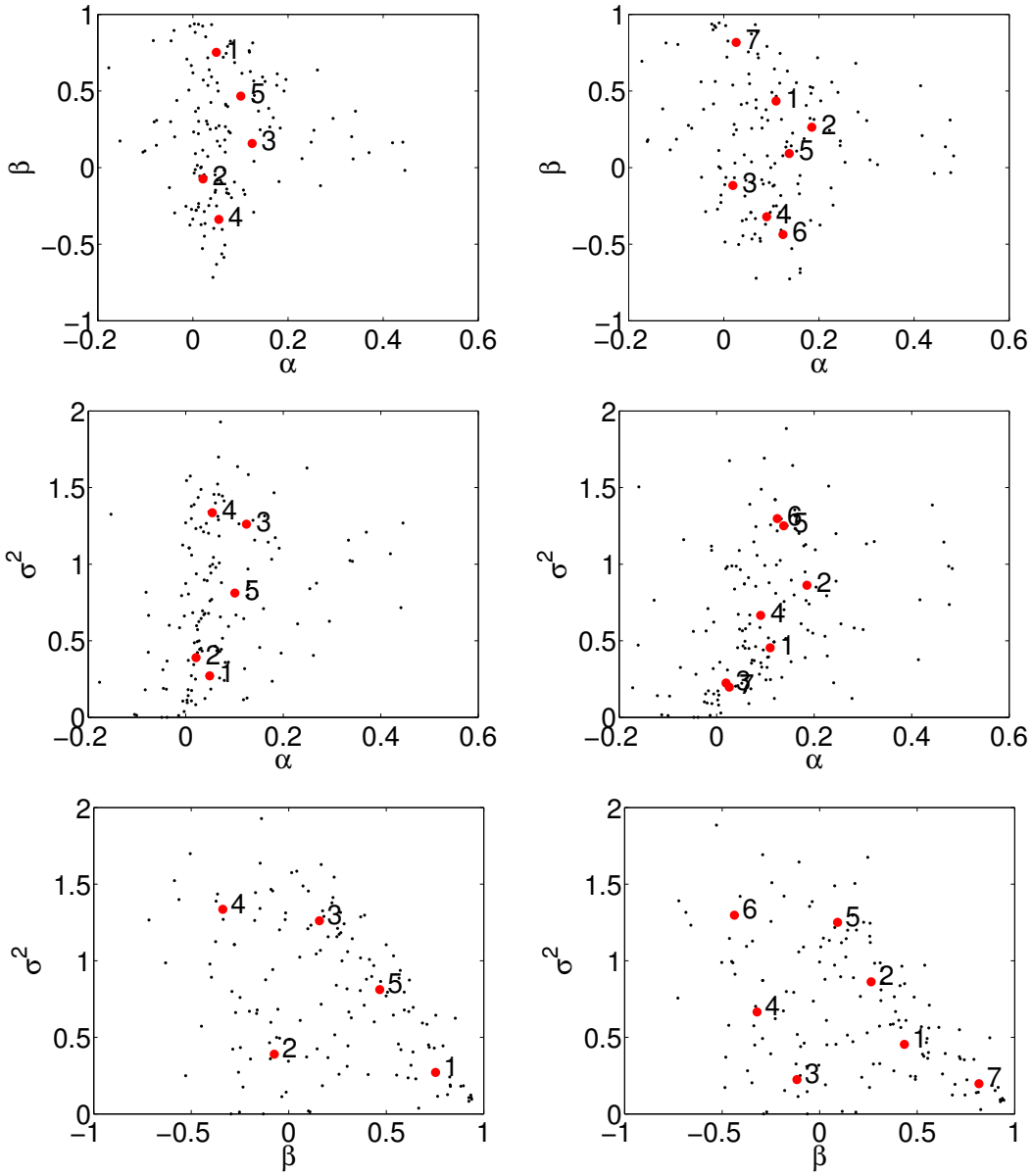


Figure S.3: Pairwise scatter plots of the series features: α_i and β_i (first row), α_i and σ_i^2 (second row) and β_i and σ_i^2 (last row). In each plot the red dots represent the cluster means. We assume alternatively 5 (left) and 7 (right) clusters.

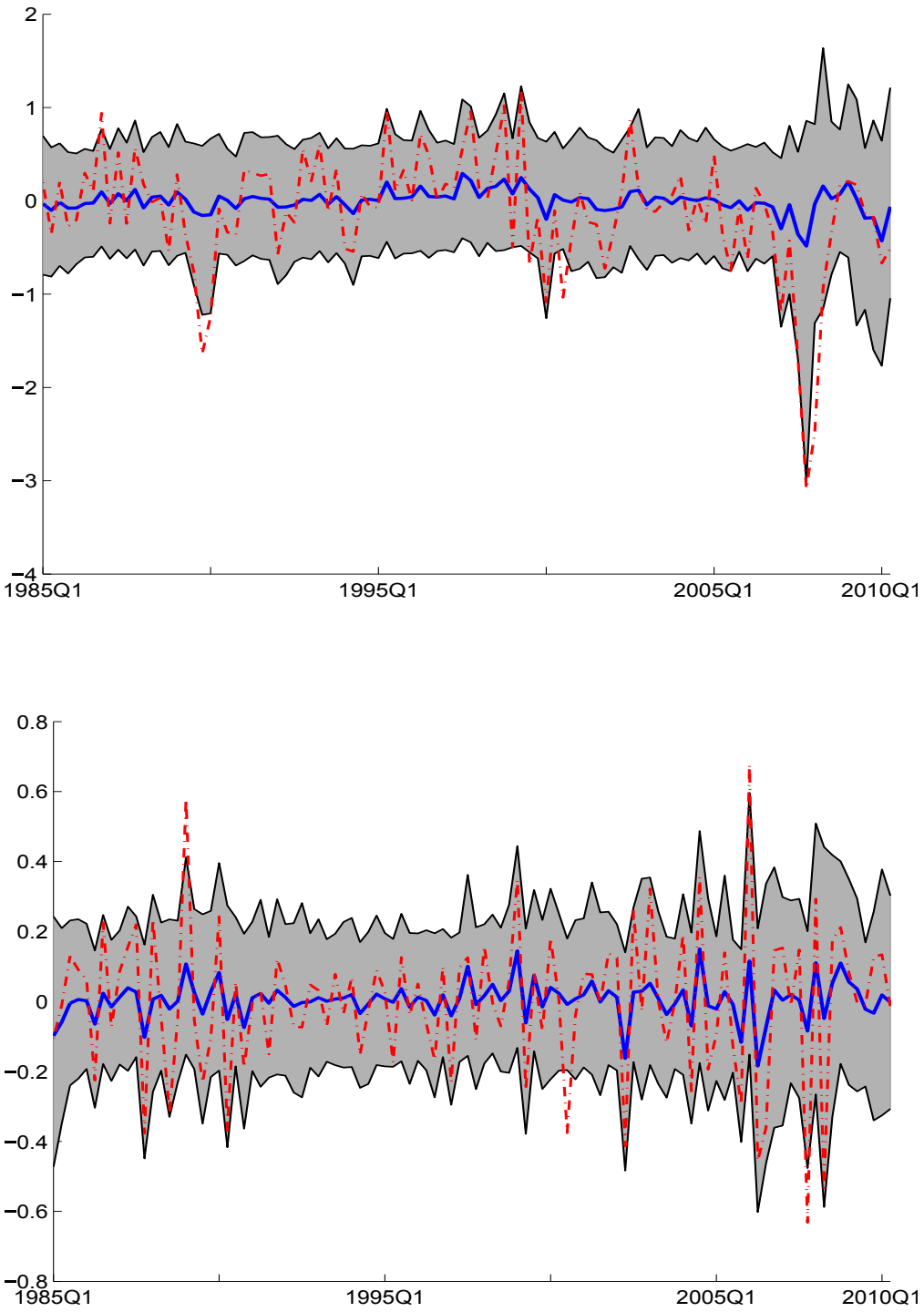


Figure S.4: 5-step ahead fan charts for demeaned GDP (top panel) and demeaned GDP deflator (bottom panel). Estimated mean (solid blue line) and 5% and 95% quantiles (gray area) of the marginal prediction density. (Demeaned) realizations in red dashed line

5 clusters			
k	α	β	σ^2
1	0.049	0.752	0.270
2	0.021	-0.074	0.390
3	0.124	0.157	1.260
4	0.054	-0.338	1.335
5	0.100	0.466	0.811

7 clusters			
k	α	β	σ^2
1	0.109	0.434	0.454
2	0.185	0.263	0.862
3	0.019	-0.116	0.224
4	0.090	-0.321	0.665
5	0.137	0.091	1.250
6	0.124	-0.437	1.297
7	0.026	0.817	0.197

Table S.5: Cluster means for the 5 (top table) and 7 (bottom table) cluster analysis. The first column, k, indicates the cluster number given in Fig. S.3 and the remaining three columns the cluster mean along the different directions of the parameter space.

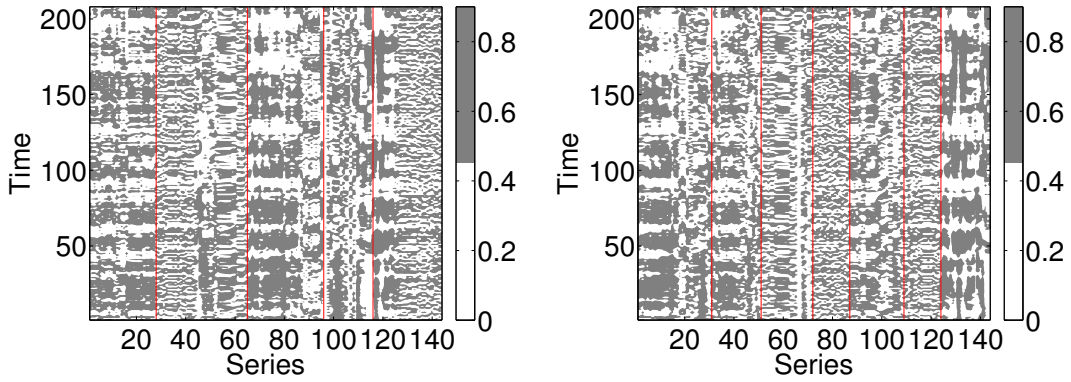


Figure S.5: Normal cumulative density function for the standardised series. The series are ordered by cluster label. We assume alternatively 5 (left) and 7 (right) clusters.

Figure S.6 shows the De Finetti's diagram of the two largest weights in the seven clusters for each of the variables to be predicted and a selection of horizons, $h = 1, 2, 5$, using multivariate combinations and assuming $b_{k,ij}$ equal to the recursive log score for model i in cluster j when predicting the series k.

Figure S.8 shows a typical output of the model weights ($b_{k,ij}$) in the seven clusters. There are large differences across clusters: the clusters 2, 4, 5 and 6 have few models with most of the weights; the other clusters, 1, 3 and 7, have more similar weights across models. This finding should be associated with the largest weights for the clusters 2, 4, 5 and 6 and indicates that using recursive time-varying $b_{k,ij}$ weights within the clusters increases forecast accuracy for GDP growth relative to using equal weights. Figure S.8 also indicates that the weights within clusters are much more volatile than the cluster common component, indicating that individual model performances may change much over time even if information in a given clusters is stable.

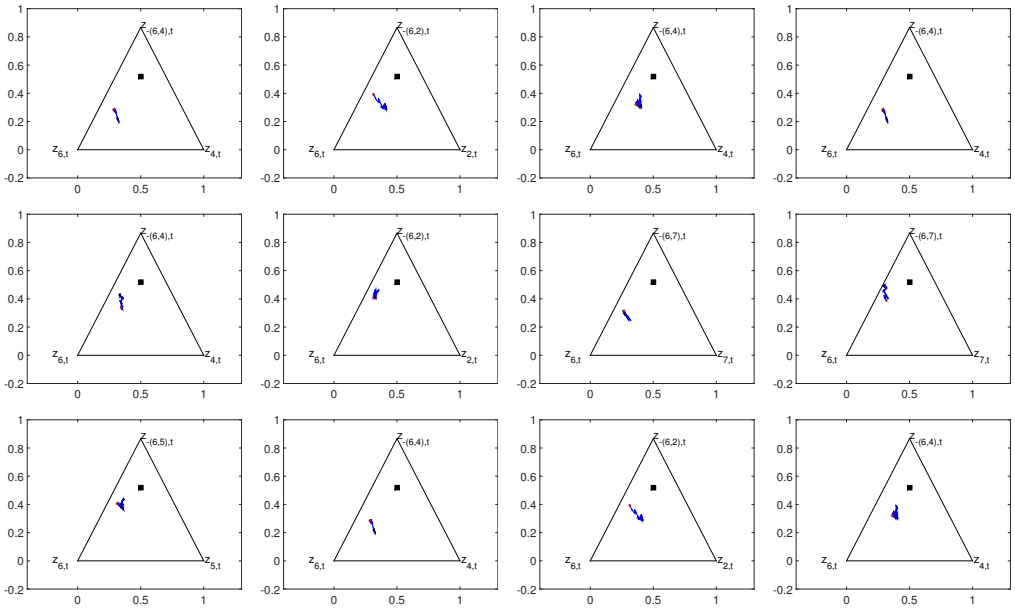


Figure S.6: De Finetti's diagrams for the dynamic comparison of the two largest weights. Rows: diagrams for the four series of interest (real GDP growth rate, GDP deflator, Treasury Bills, employment). Columns: forecast horizons (1, 3 and 5 quarters). In each plot the trajectory (blue line), the starting (red) and ending (black) points and the equal weight composition (square).

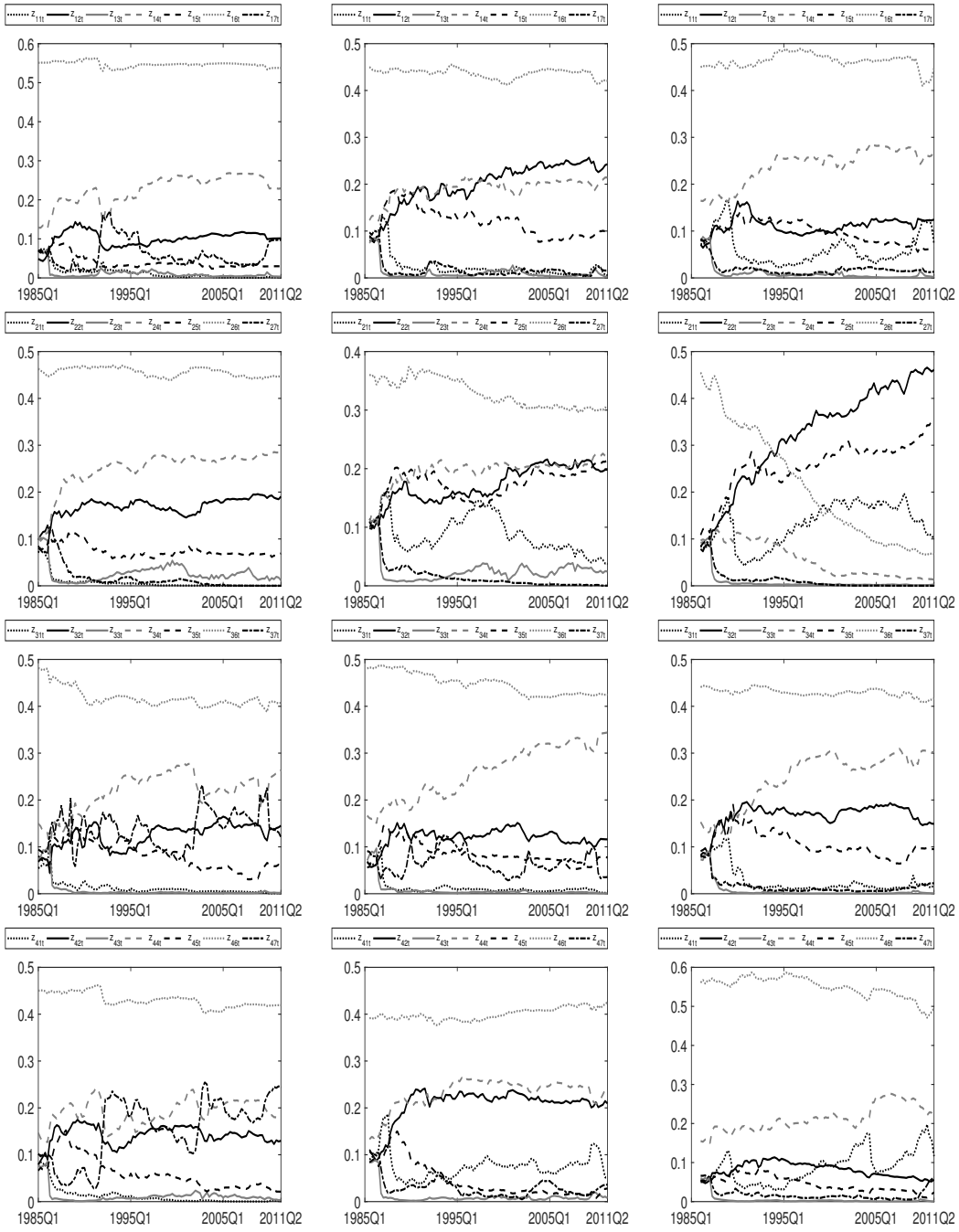


Figure S.7: In each plot the mean logistic-normal weights (different lines) for the univariate combination model are given. Rows: plot for the four series of interest (real GDP growth rate, GDP deflator, 3-month Treasury Bills, employment). Columns: forecast horizons (1, 3 and 5 quarters).

S.5 Computing time

In this section we compare the computational speed of CPU with GPU in the implementation of our combination algorithm for both the financial and macro application. Whether CPU computing is standard in econometrics, GPU approach to computing has been received large attention in economics only recently. See, for example, **?** for a review, Geweke and Durham (2012) and **?** for applications to Bayesian inference and **?**, **?** and **?** for solving DSGE models.

The CPU and the GPU versions of the computer program are written in MATLAB, as described in Casarin et al. (2015). In the CPU setting, our test machine is a server with two Intel Xeon CPU E5-2667 v2 processors and a total of 32 core. In the first GPU setting, our test machine is a NVIDIA Tesla K40c GPU. The Tesla K40c card is with 12GB memory and 2880 cores and it is installed in the CPU server. In the second GPU setting, our test machine is a NVIDIA GeForce GTX 660 GPU card, which is a middle-level video card, with a total of 960 cores. The test machine is a desktop Windows 8 machine, has 16 GB of Ram and only requires a MATLAB parallel toolbox license.

We compare two sets of combination experiments, the density combination based on 4 clusters with equal weights within clusters and time-varying volatility, DCEW-SV, and the density combination with univariate combination based on 7 clusters with recursive log score weights within clusters, UDCLS7[2], see Section S.4.2, for an increasing number of particles N. In both sets of experiments we calculated, in seconds, the overall average execution time reported in Table S.6.

[2]The case MCDCLS7 provide similar relative timing, in absolute terms a bit faster than the univariate ones.

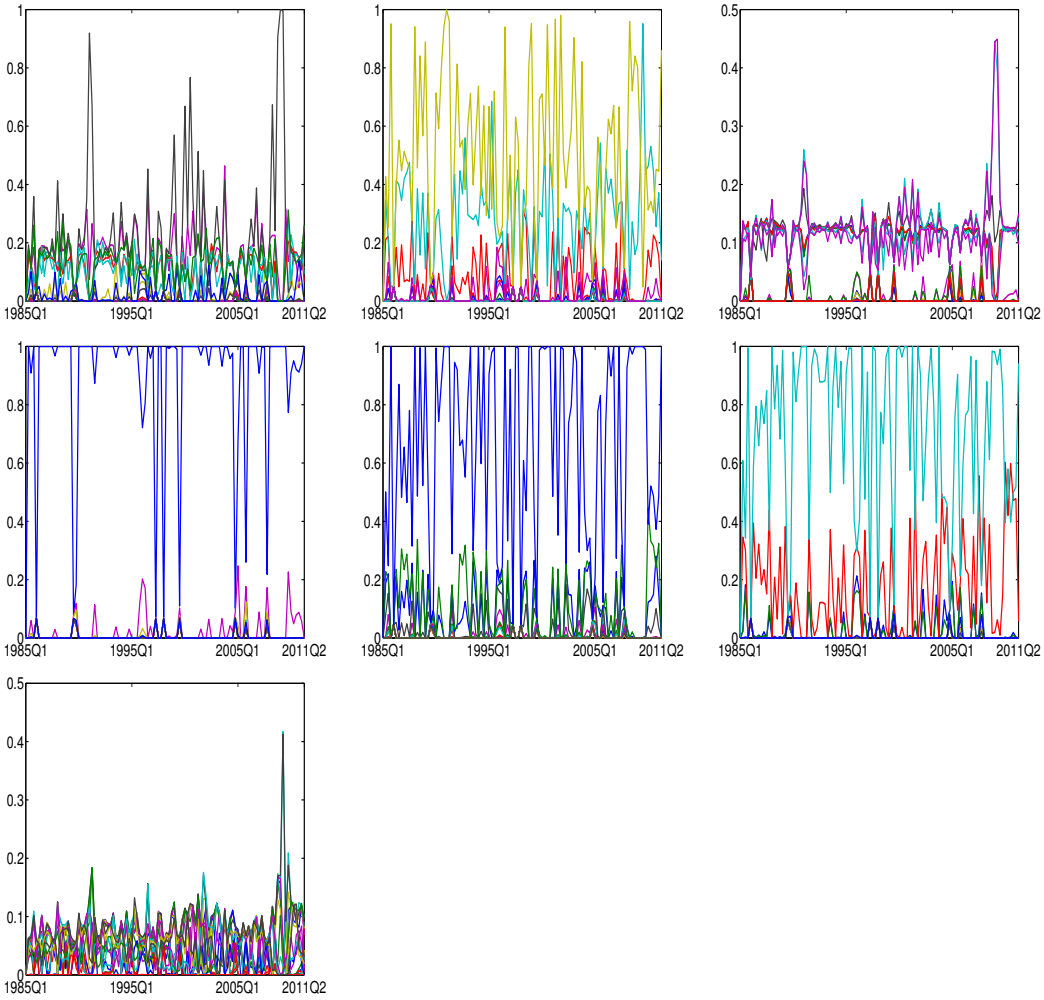


Figure S.8: The plots show the model weights ($b_{k,ij}$) in each cluster ($i = j$) when forecasting GDP growth ($k = 1$) at the 1-step ahead horizon. The first row refers to clusters 1, 2, and 3; the second row to clusters 4, 5, and 6; the last row to cluster 7.

As the table shows, the CPU implementation is slower then the first GPU set-up in all cases. The NVIDIA Tesla K40c GPU provides gains in the order of magnitude from 2 to 4 times than the CPU. Very interestingly, even the second GPU set-up, which can be installed in a desktop machine, provides execution times comparable to the CPU in the financial applications and large gains in the macro applications. Therefore, the GPU environment seems the preferred one for our density combination problems and when the number of predictive density becomes very large a GPU server card gives the highest gains.

	DCEW-SV			UDCLS7		
Draws	100	500	1000	100	500	1000
CPU	1032	5047	10192	5124	25683	51108
GPU 1	521	2107	4397	1613	6307	14017
GPU 2	1077	5577	13541	2789	13895	27691
Ratio 1	1.98	2.39	2.32	3.18	4.07	3.65
Ratio 2	0.96	0.90	0.75	1.84	1.85	1.85

Table S.6: Observed total time (in seconds) and CPU/GPU ratios for the algorithm on CPU and GPU on different machines and with different numbers of particles. The CPU is a 32 core Intel Xeon CPU E5-2667 v2 two processors and the GPU1 is a NVIDIA Tesla K40c GPU and the GPU2 is a NVIDIA GeForce GTX 660. "Ratio 1" refers to the CPU/GPU 1 ratio and "ratio 2" refers to the CPU/GPU 2 ratios. Number below 1 indicates the CPU is faster, number above one indicates that the GPU is faster.

References

Aldrich, E. M. (2014). GPU Computing in Economics. In L., K. J. and Schmedders, K., editors, *Handbook of Computational Economics, Vol. 3*. Elsevier.

Aldrich, E. M., Fernández-Villaverde, J., Gallant, A. R., and Rubio Ramırez, J. F. (2011). Tapping the Supercomputer Under Your Desk: Solving Dynamic Equilibrium Models with Graphics Processors. *Journal of Economic Dynamics and Control*, 35:386–393.

Anderson, T. W. (1984). *An Introduction to Multivariate Statistical Analysis, 3rd Edition*. Wiley.

Andrews, D. W. K. and Monahan, J. C. (1992). An Improved Heteroskedasticity and Autocorrelation Consistent Covariance Matrix Estimator. *Econometrica*, 60:953–966.

Billio, M., Casarin, R., Ravazzolo, F., and van Dijk, H. K. (2013). Time-varying Combinations of Predictive Densities using Nonlinear Filtering. *Journal of Econometrics*, 177:213–232.

Cannings, C. and Edwards, A. W. F. (1968). Natural Selection and the De Finetti Diagram. *Annals of Human Genetics*, 31:421–428.

Casarin, R., Grassi, S., Ravazzolo, F., and van Dijk, H. K. (2015). Parallel Sequential Monte Carlo for Efficient Density Combination: the DeCo Matlab Toolbox. *Jounal of Statistical Software*, 68.

Clark, T. E. and McCracken, M. W. (2011). Testing for Unconditional Predictive Ability. In *Oxford Handbook of Economic Forecasting*. Oxford University Press.

Clark, T. E. and McCracken, M. W. (2015). Nested Forecast Model Comparisons: A New Approach to Testing Equal Accuracy. *Journal of Econometrics*, 186:160–177.

Clark, T. E. and Ravazzolo, F. (2015). The Macroeconomic Forecasting Performance of Autoregressive Models with Alternative Specifications of Time-Varying Volatility. *Journal of Applied Econometrics*, 30:551–575.

Diebold, F. X. and Mariano, R. S. (1995). Comparing Predictive Accuracy. *Journal of Business & Economic Stastistics*, 13:253–263.

Dziubinski, M. P. and Grassi, S. (2013). Heterogeneous Computing In Economics: A Simplified Approach. *Computational Economics*, 43:485–495.

Geweke, J. and Durham, G. (2012). Massively Parallel Sequential Monte Carlo for Bayesian Inference. Working papers, National Bureau of Economic Research, Inc.

Gneiting, T. and Raftery, A. E. (2007). Strictly Proper Scoring Rules, Prediction, and Estimation. *Journal of the American Statistical Association*, 102:359–378.

Gneiting, T. and Ranjan, R. (2011). Comparing Density Forecasts Using Threshold and Quantile Weighted Scoring Rules. *Journal of Business and Economic Statistics*, 29:411–422.

Gneiting, T. and Ranjan, R. (2013). Combining Predicitve Distributions. *Electronic Journal of Statistics*, 7:1747–1782.

Groen, J. J. J., Paap, R., and Ravazzolo, F. (2013). Real-Time Inflation Forecasting in a Changing World. *Journal of Business & Economic Stastistics*, 31:29–44.

Kitagawa, G. (1998). Self-organizing State Space Model. *Journal of the American Statistical Association*, 93:1203–1215.

Kitagawa, G. and Sato, S. (2001). Monte Carlo Smoothing and Self-Organizing State-Space Model. In Doucet, A., de Freitas, N., and Gordon, N., editors, *Sequential Monte Carlo Methods in Practice*. Springer-Verlag.

Lee, A., Yau, C., Giles, M. B., Doucet, A., and Holmes, C. C. (2010). On the Utility of Graphic Cards to Perform Massively Parallel Simulation with Advanced Monte Carlo Methods. *Journal of Computational and Graphical Statistics*, 19:769–789.

Lerch, S., Thorarinsdottir, T., Ravazzolo, R., and Gneiting, T. (2017). Forecaster's Dilemma: Extreme Events and Forecast Evaluation. *Statistical Science*, 32:106–127.

Liu, J. and West, M. (2001). Combined Parameter and State Estimation in Simulation Based Filtering. In Doucet, A., de Freitas, N., and Gordon, N., editors, *Sequential Monte Carlo Methods in Practice*. Springer-Verlag.

Morozov, S. and Mathur, S. (2012). Massively Parallel Computation

Using Graphics Processors with Application to Optimal Experimentation in Dynamic Control. *Computational Economics*, 40:151–182.

Pawlowsky-Glahn, V., Egozcue, J. J., and Tolosana-Delgado, R. (2015). *Modeling and Analysis of Compositional Data*. Wiley.

Ravazzolo, F. and Vahey, S. V. (2014). Forecast densities for economic aggregates from disaggregate ensembles. *Studies in Nonlinear Dynamics and Econometrics*, 18:367–381.

Stock, J. H. and Watson, W. M. (2005). Implications of dynamic factor models for VAR analysis. Technical report, NBER Working Paper No. 11467.