WORKING PAPER

Dynamic predictive density combinations for large data sets in economics and finance

NORGES BANK RESEARCH

12 | 2015

AUTHORS:

ROBERTO CASARIN
STEFANO GRASSI
FRANCESCO RAVAZZOLO
HERMAN K. VAN DIJK

NORGES BANK

Working papers fra Norges Bank, fra 1992/1 til 2009/2 kan bestilles over e-post:
servicesenter@norges-bank.no

Fra 1999 og senere er publikasjonene tilgjengelige på www.norges-bank.no

Working papers inneholder forskningsarbeider og utredninger som vanligvis ikke har fått sin endelige form. Hensikten er blant annet at forfatteren kan motta kommentarer fra kolleger og andre interesserte. Synspunkter og konklusjoner i arbeidene står for forfatternes regning.

Working papers from Norges Bank, from 1992/1 to 2009/2 can be ordered by e-mail

servicesenter@norges-bank.no

Working papers from 1999 onwards are available on www.norges-bank.no

Norges Bank's working papers present research projects and reports (not usually in their final form) and are intended inter alia to enable the author to benefit from the comments of colleagues and other interested parties. Views and conclusions expressed in working papers are the responsibility of the authors alone.

ISSN 1502-8143 (online)
ISBN 978-82-7553-875-6 (online)

Dynamic Predictive Density Combinations for Large Data Sets in Economics and Finance*

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

[†]University Ca' Foscari of Venice
[‡]Norges Bank and Centre for Applied Macro and Petroleum economics
at BI Norwegian Business School
[§]University of Kent
[¶]Econometric Institute Erasmus University Rotterdam, Econometrics Department
VU University Amsterdam and Tinbergen Institute

July 2015

Abstract

A Bayesian nonparametric predictive model is introduced to construct time-varying weighted combinations of a large set of predictive densities. A clustering mechanism allocates these densities into a smaller number of mutually exclusive subsets. Using properties of the Aitchinson's geometry of the simplex, combination weights are defined with a probabilistic interpretation. The class-preserving property of the logistic-normal distribution is used to define a

*This working 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. We thank John Geweke, Jim Stock, Peter Schotman, Peter Hansen, Gael Martin, Michael Smith, Anastasios Panagiotalis, Barbara Rossi and conference and seminar participants at Erasmus University Rotterdam Workshop on "The Econometric Analysis of Recurrent Events in Macroeconomics and Finance", the 34th International Symposium on Forecasting, the 8th International CFE meeting in Pisa, the 25th EC^2 Conference on "Advances in Forecasting", the RCEA 9th Rimini Bayesian Workshop, the IAAE Conference in Thessaloniki, Institute for Advance Studies Vienna, Maastricht University, Monash University, Norges Bank, the Stevanovich Center at University of Chicago, UTS Sydney, and UPF Barcelona, for very useful comments. Roberto Casarin's research is supported by funding from the European Union, Seventh Framework Programme FP7/2007-2013 under grant agreement SYRTO-SSH-2012-320270, by the Institut Europlace of Finance, "Systemic Risk grant", the Global Risk Institute in Financial Services, the Louis Bachelier Institute, "Systemic Risk Research Initiative", and by the Italian Ministry of Education, University and Research (MIUR) PRIN 2010-11 grant MISURA.

compositional dynamic factor model for the weight dynamics with latent factors defined on a reduced dimension simplex. Groups of predictive models with combination weights are updated with parallel clustering and sequential Monte Carlo filters. The procedure is applied to predict Standard & Poor's 500 index using more than 7000 predictive densities based on US individual stocks and finds substantial forecast and economic gains. Similar forecast gains are obtained in point and density forecasting of US real GDP, Inflation, Treasury Bill yield and employment using a large data set.

JEL codes: C11, C15, C53, E37.
Keywords: Density Combination, Large Set of Predictive Densities, Compositional Factor Models, Nonlinear State Space, Bayesian Inference, GPU Computing.

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, 2004, 2005, 2014, and Bańbura et al., 2010). The recent fast growth in (real-time) big data allows researchers to predict 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, for instance, that there are potential gains from forecasting using a large set of predictors instead of a single predictor from a univariate time series. However, forecasting with many predictors and high-dimensional models requires new modeling strategies (to keep the number of parameters and latent variables relatively small), efficient inference methods and extra computing power like parallel computing. We refer to Granger (1998) for an early discussion of these issues.

We propose a Bayesian nonparametric model in order to deal with large set of predictive densities. The proposed model is still relatively parsimonious in the number of parameters and latent variables and has a representation in terms of a dependent sequence of random measures on the set of predictors of different models, with common atoms and component-specific random weights. Our model extends the mixture of the experts and the smoothly mixing regression models (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) by allowing for dependence between the random weights of the mixture and for model incompleteness. In this sense,

our combination model shares some similarities with the dependent random measures used in Bayesian nonparametric models (see Müller and Quintana, 2010 and Müller and Mitra, 2013).

The proposed approach introduces an information reduction step by making use of a clustering mechanism where allocation variables map the original set of predictive densities into a relatively small number of mutually exclusive subsets with combination weights driven by cluster specific latent processes specified as a compositional factor model, see Pawlowsky-Glahn and Buccianti (2011) for details on compositional data analysis. This structure of the latent space allows for a probabilistic interpretation of the weights as model probabilities in the combination scheme that are evolving over time. There exists an issue of analytic tractability of the probabilistic information in the information reduction step. Here the class-preserving property of the logistic-normal distribution (see Aitchinson and Shen, 1980, Aitchinson, 1982) is used. The complete model is represented in a nonlinear state space form where the measurement equation refers to the combination model and the transition function of the latent weights is a dynamic compositional factor model with a noise process that follows a multivariate logistic-normal distribution.[1] Given that the space of the random measures is equipped with suitable operations and norms, we also show that this nonlinear state space model may be interpreted as a generalized linear model with a local level component. Sequential prediction and filtering is applied in order to efficiently update the dynamic clustered weights of the combination model. In this sense the paper contributes to the literature on time series on a bounded domain (see, e.g., Aitchinson, 1982, Aitchinson, 1986 and Billheimer et al., 2001) and on state space models for compositional data analysis (see, e.g., Grunwald et al., 1993). In that literature the compositional data are usually observed, while in our model the weights are latent probabilities.

Our model extends Stock and Watson (2002) and Stock and Watson (2005) along two directions. First, we propose a joint prediction model for a group of variables of interest instead of a single variable; second, we combine large sets of predictive densities instead of large sets of point forecasts. We also extend Billio et al. (2013) and Casarin et al. (2015) substantially by making a connection with the mixture of experts literature and by allowing for a high dimensional combination model that is still parsimonious in the number of parameters and latent variables.

Another contribution of this paper refers to the literature on parallel computing.

[1] This distribution has arisen naturally in the reconciliation of subjective probabilities assessments, see Lindley et al. (1979) and also Pawlowsky-Glahn et al. (2015), chapter 6 for details.

We provide an estimate of the gain, in terms of computing time, of the GPU implementation of our density combination strategy with respect to CPU multi-core implementation. This approach to computing has been successfully applied in econometrics for Bayesian inference (Geweke and Durham, 2012 and Lee et al., 2010) and in economics for solving DSGE models (Aldrich et al., 2011 and Morozov and Mathur, 2012).

The proposed method is applied to two well-known problems in finance and economics: predicting stock returns and predicting macro-finance variables using the Stock and Watson (2005) dataset. In the first example, we use more than 7000 predictive densities based on 3712 US individual stock return series to replicate the daily aggregate S&P 500 returns over the sample 2007-2009 and predict the economic value of tail events like Value-at-Risk. We find large accuracy gains with respect to the no-predictability benchmark and predictions from individual models estimated on the aggregate index. In the second example, we find substantial gains in point and density forecasting of US real GDP, GDP deflator inflation, Treasury Bill yield and employment over the last 25 years for all horizons from one-quarter ahead to five-quarter ahead. The highest accuracy is achieved when the four series are predicted simultaneously using our combination schemes within and across cluster weights based on log score learning. We emphasize that the cluster-based weights contain relevant signals about the importance of the forecasting performance of each of the models used in the 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.

As far as computational gains using parallel computing is concerned, we find that the GPU algorithm reduces the computation time with respect to the CPU version of several multiples of CPU computing time.

The paper is structured as follows. Section 2 describes the Bayesian nonparametric predictive model and presents the strategy of the dimension reduction of the latent space. Section 3 provides details of the probabilistic information reduction and a representation of our model as a nonlinear compositional state space model. Section 4 presents the inference procedure. Section 5 applies our model to large set of US stocks are used to predict the aggregate index. Section 5.2 presents an analysis of the Stock and Watson (2005) macroeconomic data set. Section 6 concludes. The Appendices contain more details on data, derivations and results.

2 Density combination and clustering for large data sets

This paper builds on the combination of predictive densities with time-varying weights and on a information reduction technique based on sequential clustering.

2.1 Model uncertainty and model combination

Our combination approach is based on a convolution of predictive densities that consists of a model combination density, a time-varying weight density and a density of the predictors of many models (Billio et al., 2013, Casarin et al., 2015). See also Waggoner and Zha (2012) and Del Negro et al. (2014) who propose time-varying weights in the linear opinion framework and Fawcett et al. (2015) who introduce time-varying weights in the generalized linear pool. Conflitti et al. (2012) propose optimal combinations of large set of point and density survey forecasts; their weights are, however, not modeled with time-varying patterns. Finally, Raftery et al. (2010) develop Dynamic Model Averaging that allows the "correct" model to vary over time.

In this paper we provide a representation of the density combination approach in terms of a Bayesian nonparametric predictive model and show the relationship with the mixture of experts approach to construct predictive densities, elaborating on the model presented in Billio et al., 2013 Appendix B and in Del Negro et al. (2014). Let $\mathbf{y}_t = (y_{1t}, \ldots, y_{Kt})'$ be the K-dimensional vector of variables of interest, and $\tilde{\mathbf{y}}_t = (\tilde{y}_{1t}, \ldots, \tilde{y}_{nt})'$ a vector of n random predictors for the variables of interest with densities $f_{it}(\tilde{y}_{it})$, $i = 1, \ldots, n$, conditional on the information set available at time $t-1$. We introduce a sequence of discrete probability distributions over the set of predictors, which defines the probability, $w_{i,kt}$, of the i-th predictive model at time t to be used in forming the prediction for the variable of interest y_{kt}. Thus, we define the following sequence of possibly dependent random measures

$$\mathbb{P}_{kt}(d\vartheta_k) = \sum_{i=1}^{n} w_{i,kt}\delta_{\tilde{y}_{it}}(d\vartheta_k) \tag{1}$$

$t = 1, \ldots, T$, $k = 1, \ldots, K$. where δ_x is a point mass at x, ϑ_k is a parameter of interest of the predictive distribution of the variable y_{kt}, and $\mathbf{w}_{kt} = (w_{1,kt}, \ldots, w_{n,kt})'$ is a set of random weights defined by the following multivariate logistic construction

$$w_{i,kt} = \frac{\exp\{x_{i,kt}\}}{\sum_{i=1}^{n}\exp\{x_{i,kt}\}} \tag{2}$$

where $\mathbf{x}_{kt} = (x_{1,kt}, \ldots, x_{n,kt})' \in \mathbb{R}^n$ is a vector of latent variables. We denote

with $\mathbf{w}_{kt} = \phi^{-1}(\mathbf{x}_{kt})$ the multivariate logistic transform. The random measures $\mathbb{P}_{kt}$, $k = 1, \ldots, K$, contain extra-sample information about the variables of interest, and we assume that each random measure can be used as prior distribution for a parameter ϑ_k of a given predictive distribution for the variable of interest y_{kt}. The sequence of dependent random measures can be interpreted as an expert system and shares some similarities with the hierarchical mixtures of experts, the dependent Dirichlet processes and the random partition models as discussed in Müller and Quintana (2010). See also Müller and Mitra (2013) for a review. Finally, note that the random measures share the same atoms, but have different weights. See, e.g. Bassetti et al. (2014), for a different class of the random measures based on the stick-breaking construction of the weights and measure-specific atoms. Section 3 discusses some features of the space of the random weights used in this paper.

At time $t-1$, the sequence of random measure $\mathbb{P}_{kt}$, $k = 1, \ldots, K$ can be employed as a prior distribution for the following sequence of conditional predictive densities

$$y_{kt} \sim \mathcal{K}_{kt}(y_{kt}|\vartheta) \tag{3}$$

$k = 1, \ldots, K$, in order to obtain the following conditional predictive density

$$f_{kt}(y_{kt}|\tilde{\boldsymbol{y}}_t) = \int \mathcal{K}_{kt}(y_{kt}|\vartheta)\mathbb{P}_{kt}(d\vartheta) = \sum_{i=1}^{n} w_{i,kt}\mathcal{K}_{kt}(y_{kt}|\tilde{y}_{it}) \tag{4}$$

If one chooses $\mathcal{K}_{kt}(y_{kt}|\vartheta)$ to be the pdf of a normal distribution $\mathcal{N}(\mu, \sigma^2)$ and let μ be the parameter of interest, then y_{kt} follows a Gaussian mixture combination model (see Billio et al. (2013) for alternative specifications),

$$f_{kt}(y_{kt}|\mathbf{w}_{kt}, \sigma^2_{kt}, \tilde{\boldsymbol{y}}_t) \sim \sum_{i=1}^{n} w_{i,kt} f(y_{kt}|\tilde{y}_{it}, \sigma^2_{k,t}) \tag{5}$$

$$f_{kt}(\log \sigma^2_{kt}) \sim f(\log \sigma^2_{kt}|\log \sigma^2_{k,t-1}, \sigma^2_{\eta_k}) \tag{6}$$

$k = 1, \ldots, K$, $t = 1, \ldots, T$, where $f(y|\mu, \sigma^2)$ is the pdf of the normal distribution $\mathcal{N}(\mu, \sigma^2)$, and σ^2_{kt}, $t = 1, \ldots, T$, is a stochastic volatility process. As shown in the following, the process σ^2_{kt} controls the overall uncertainty level about the prediction models used in the combination. When the uncertainty level tends to zero then we recover as a limiting case the mixture of experts or the smoothly mixing regressions models (see Appendix B in Billio et al., 2013).

Proposition 2.1 (Mixture representation)**.** *Under standard regularity conditions,*

the marginal predictive density has the following discrete and continuous mixture representation

$$f_{kt}(y_{kt}|\mathbf{w}_{kt}) = \sum_{i=1}^{n} w_{k,it} \int_{\mathbb{R}} \mathcal{K}_{kt}(y_{kt}|\tilde{y}_{it}) f_{it}(\tilde{y}_{it}) d\tilde{y}_{it} \tag{7}$$

Under the assumption of a Gaussian predictive distribution one has $\mathcal{K}_{kt}(y|\tilde{y}_{it}) = f(y|\tilde{y}_{it}, \sigma^2_{kt})$ *and*

$$f_{kt}(y_{kt}|\mathbf{w}_{kt}) \longrightarrow \sum_{i=1}^{n} w_{i,kt} f_{it}(y_{kt}) \tag{8}$$

$k = 1, \ldots, K$, *for* $\sigma_{kt} \to 0$.

We emphasize that in our approach the overall level of uncertainty, controlled by σ^2_{kt} is a major indicator of incompleteness of the set of predictive models. The importance of measuring model incompleteness is shown in our empirical analyses.

2.2 Information reduction

In the specification of the combination model given in the previous section, the number of latent processes to estimate is nK at every time period t which can be computationally heavy, even when a small number of variables of interest, e.g. 4, and a moderate number of models, e.g. $K = 100$, are considered. The second contribution of the paper is to diminish the complexity of the combination exercise by reducing the dimension of the latent space.[2]

As a first step, the n predictors are clustered into m different groups, with $m < n$, following some (time-varying) features $\boldsymbol{\psi}_{it}$, $i = 1, \ldots, n$, of the predictive densities. We introduce $\xi_{j,it}$ as an allocation variable, which takes the value 1 if the i-th predictor is assigned to the j-th group of densities and 0 otherwise. We assume each predictor belongs to only one group, that $\sum_{j=1}^{m} \xi_{j,it} = 1$ for all i. Also, the grouping of the predictors can change over time, following a learning mechanism which is defined by a sequential clustering rule. Details of the sequential clustering rule are given in the following section.

Given the clustering of the predictors, we specify how to reduce the dimension of the latent weight space from nK to mK with $m < n$. To this aim, we specify the $(n \times m)$ allocation matrix $\Xi_t = (\boldsymbol{\xi}_{1t}, \ldots, \boldsymbol{\xi}_{mt})$, with $\boldsymbol{\xi}_{jt} = (\xi_{j,1t}, \ldots, \xi_{j,nt})'$, $j = 1, \ldots, m$, the vector of allocation variables $\xi_{j,it} \in \{0, 1\}$, and a $(m \times n)$ coefficient matrix B_{kt}

[2]We note that, although our aim is full Bayesian analysis, the very large scale of some problems and the implied heavy computations may lead to pragmatic decisions in this context in the sense that the very large set of predictive densities may be the result from applying either Bayesian or other inferential methods, see section 5.

with the i-th row and j-th column element given by $b_{ij,kt} \in \mathbb{R}$. The two matrices allow us to project the n-dimensional latent variable $\mathbf{x}_{kt}$ onto a reduced dimension latent space, through the following latent factor model

$$\mathbf{x}_{kt} = (\Xi_t \circ B_{kt})\mathbf{v}_{kt} \tag{9}$$

where $\circ$ denotes the element-by-element Hadamard's product, and $\boldsymbol{v}_{kt} = (v_{1,kt}, \ldots, v_{m,kt})'$ is a m-variate normal random walk process

$$\boldsymbol{v}_{kt} = \boldsymbol{v}_{k,t-1} + \boldsymbol{\chi}_{kt}, \quad \boldsymbol{\chi}_{kt} \overset{iid}{\sim} \mathcal{N}_m(\mathbf{0}_m, \Upsilon_k) \tag{10}$$

The process $\boldsymbol{v}_{kt}$, $t = 1, \ldots, T$, is latent and is driving the weights of the predictive densities which are used to forecast the k-th variable of interest. The set of all variable-specific latent processes, is associated width a latent space of dimension mK. The coefficients, $\xi_{j,it}$ and $b_{ij,kt}$, $j = 1, \ldots, m$, for each variable of interest k, predictor j and time t, are crucial in order to obtain a parsimonious latent variable model and consequently to reduce the computational complexity of the combination procedure.

For specific values of the coefficients $b_{ij,kt}$, we propose two alternative strategies. The first one is where all coefficients in the cluster have the same weights, which corresponds to set $b_{ij,kt}$ as:

$$b_{ij,kt} = \begin{cases} 1/n_{jt} & \text{if } \xi_{j,it} = 1 \\ 0 & \text{otherwise} \end{cases} \tag{11}$$

where

$$n_{jt} = \sum_{i=1}^{n} \xi_{j,it}$$

is the number of predictive densities in the j-th cluster at time t. Note that, following this specification of the coefficients, the weights of the n predictors for the k-th variable of interest are

$$w_{i,kt} = \frac{\exp\{v_{j_i,kt}/n_{j_i t}\}}{\sum_{j=1}^{m} \exp\{v_{j,kt}/n_{jt}\}}, \quad i = 1, \ldots, n$$

where $j_i = \sum_{j=1}^{m} j\xi_{j,it}$ indicates the group to which the i-th predictor belongs. The latent weights are driven by a set of m latent variables, with $m < n$, thus the dimensional reduction of the latent space is achieved. Moreover, let $N_{it} = \{j = 1, \ldots, n | \xi_{i,jt} = 1\}$ be the set of the indexes of all models in the cluster i, then one can see that this specifications may have the undesirable property that the weights are constant within a group, that is for all $j \in N_{it}$.

For this reason, we also propose the second specification strategy where we assume that each model contributes to the combination with a specific weight that is driven by a model-specific forecasting performance measure. If we assume g_{it} is the log score (see definition in (B.50)) of the model i at time t then

$$b_{ij,kt} = \begin{cases} \sum_{s=1}^{t} \exp\{g_{is}\}/\bar{g}_{it} & \text{if } \xi_{j,it} = 1 \\ 0 & \text{otherwise} \end{cases} \tag{12}$$

where $\bar{g}_{it} = \sum_{l \in N_{it}} \sum_{s=1}^{t} \exp\{g_{ls}\}$.

All the modeling assumptions discussed above allow us to reduce the complexity of the combination exercise because the set of time-varying combination weights to estimate is of dimension $mK < nK$.

3 Reduced-dimension state-space representation

The density combination model proposed in this paper can be written in terms of a nonlinear state space model defined on a reduced-dimension latent space. Moreover, thanks to the class-preserving property of the logistic-normal distribution, the proposed transition density can be represented as a compositional latent factor model. We also show that this nonlinear state space model may be written in the form of a generalized linear model with a local level component when the space of the random measures is equipped with suitable operations and norms.

3.1 Probabilistic information reduction

We start to introduce some useful results and definitions. Let $\mathbb{S}^n = \{\mathbf{u} \in \mathbb{R}^n_+ | u_1 + \ldots + u_n < 1\}$ be the n-dimensional standard simplex, where $\mathbb{R}^n_+$ denotes the positive orthant of $\mathbb{R}^n$. Proofs of results are presented in Appendix A.1.

Definition 3.1 (Composition function)**.** *The function* $C_m(\mathbf{u}) : \mathbb{R}^m_+ \to \mathbb{S}^{m-1}$, $\mathbf{u} \mapsto \mathbf{v} = C_m(\mathbf{u})$ *with the i-the element of* $\mathbf{v}$ *defined as* $v_i = u_i/v_m$, $i = 1, \ldots, m-1$, *with* $v_m = \mathbf{u}'\boldsymbol{\iota}_m$.

Proposition 3.1 (Logistic-normal distribution)**.** *Let* $\mathbf{v} \sim \mathcal{N}_m(\boldsymbol{\mu}, \Upsilon)$, *and define* $\mathbf{u} = \exp(\mathbf{v})$, *that is the component-wise exponential transform of* $\mathbf{v}$, *and* $\mathbf{z} = C_m(\mathbf{u})$, *that is the composition of* $\mathbf{u}$, *then* $\mathbf{u}$ *follows a m-variate log-normal distribution,* $\Lambda_m(\boldsymbol{\mu}, \Upsilon)$, *and* $\mathbf{z}$ *follows a logistic-normal distribution* $\mathcal{L}_{m-1}(D_m\boldsymbol{\mu}, D_m\Upsilon D_m')$ *with*

density function

$$p(\mathbf{z}|\boldsymbol{\mu},\Upsilon) = |2\pi D_m \Upsilon D_m'|^{-1/2} \left(\prod_{j=1}^{m-1} z_j\right)^{-1} \exp\Big(-\frac{1}{2}\left(\log(\mathbf{z}/z_m) - D_m\boldsymbol{\mu}\right) \quad (13)$$

$$\left(D_m \Upsilon D_m'\right)^{-1}\left(\log(\mathbf{z}/z_m) - D_m\boldsymbol{\mu}\right)'\Big) \quad (14)$$

where $\mathbf{z} \in \mathbb{S}^{m-1}$, $z_{m,kt} = 1 - \mathbf{z}'\boldsymbol{\iota}_{m-1}$, $D_m = (I_{m-1}, -\boldsymbol{\iota}_{m-1})$ *and* $\boldsymbol{\iota}_{m-1}$ *is the* $(m-1)$ *unit vector.*

Corollary 3.1. *Let* $\mathbf{v}_{kt} \sim \mathcal{N}_m(\mathbf{v}_{kt-1}, \Upsilon_k)$, *and* $\mathbf{z}_{kt} = C_m(\exp(\mathbf{v}_{kt}))$, *then* $\mathbf{z}_{kt} \in \mathbb{S}^{m-1}$ *follows the logistic-normal distribution* $\mathcal{L}_{m-1}(D_m\mathbf{v}_{kt-1}, D_m\Upsilon_k D_m')$.

The class-preserving property of the composition of the logistic-normal vectors (see Aitchinson and Shen, 1980) will be used in the proof of the main result of this section. We show how this property adapts to our state space model.

Proposition 3.2 (Class-preserving property)**.** *Let* $\mathbf{z}_{kt} \sim \mathcal{L}_{m-1}(D_m\mathbf{v}_{kt-1}, D_m\Upsilon_k D_m')$ *a logistic-normal vector, and* A *a* $(c \times m-1)$ *matrix. Define the following transform* $\mathbf{w} = \phi_A(\mathbf{z})$ *from* $\mathbb{S}^{m-1}$ *to* $\mathbb{S}^c$ *, with in our case* $m < c$,

$$w_{i,kt} = \prod_{j=1}^{m-1}\left(\frac{z_{j,kt}}{z_{m,kt}}\right)^{a_{ij}}\left(1 + \sum_{i=1}^{c}\prod_{j=1}^{m-1}\left(\frac{z_{j,kt}}{z_{m,kt}}\right)^{a_{ij}}\right)^{-1}, \quad i = 1,\ldots,c$$

then $\mathbf{w}_{kt} = (w_{1,kt},\ldots,w_{c,kt})$ *follows the logistic-normal* $\mathcal{L}_c(AD_m\mathbf{v}_{kt-1}, AD_m\Upsilon_k D_m' A')$.

3.2 A reduced-dimension state-space representation

Given the results in the preceding subsection, we can now state the main result.

Proposition 3.3 (State-space form)**.** *Let* $A_{kt} = \Xi_t \circ B_{kt}$, $k = 1,\ldots,K$, *be a matrix of coefficients, then the model given in equations 5-9 can be written in the following state space form*

$$\mathbf{y}_t \sim \prod_{k=1}^{K}\sum_{i=1}^{n} w_{i,kt}\mathcal{N}\left(\tilde{y}_{it}, \sigma_{kt}^2\right) \quad (15)$$

$$\tilde{\mathbf{w}}_{kt} \sim \mathcal{L}_{n-1}\left(\tilde{\tilde{A}}_{kt}D_m\mathbf{v}_{kt-1}, \tilde{\tilde{A}}_{kt}D_m\Upsilon_k D_m'\tilde{\tilde{A}}_{kt}'\right), \quad k = 1,\ldots,K \quad (16)$$

$\tilde{\mathbf{w}}_{kt} = (w_{1,kt},\ldots,w_{n-1,kt})'$ *and* $w_{n,kt} = 1 - \tilde{\mathbf{w}}_{kt}'\boldsymbol{\iota}_{n-1}$, $\otimes$ *denotes the Kronecker's product,* $\tilde{\tilde{A}}_{kt} = (\tilde{A}_{kt}', O_{(n-\tilde{n}_t)\times(m-1)}')'$, *with* $\tilde{n}_t = Card(\tilde{N}_t)$ *and* $\tilde{N}_t = \{i = 1,\ldots,n | \xi_{m,it} \neq 1\}$ *the set of indexes of the models allocated in the cluster* m.

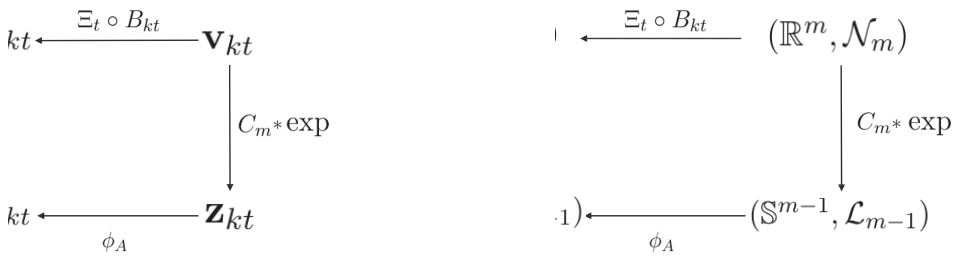

Figure 1: Relationships between the latent variables (left) and the latent probability spaces (right) involved in our compositional latent factor model. The origin of the directed edge indicates the transformed variable, the arrow indicates the results of the transformation, and the edge label defines the transform applied. The symbol $*$ indicates a composition of functions.

The previous proposition establishes a relationship between the set of latent weights $\mathbf{w}_{kt}$ and their projection, $\mathbf{z}_{kt}$, on the lower dimension latent space $\mathbb{S}^{m-1}$. The diagram on the left side of Figure 1 summarizes the relationships between the latent variables involved in our compositional latent factor model. The symbol $*$ indicates function composition. The diagram on the right shows the relationship between the probability latent spaces. In both diagrams, the chaining process given by the function composition $\phi_A * C_m * \exp$ indicates that the probabilistic interpretation of the n-dimensional weight vector $\mathbf{w}_{kt}$ naturally transfers to the m-dimensional vector $\mathbf{z}_{kt}$, with $m < n$.

In the same diagram an alternative chaining process is given by the function composition $C_n * \exp *(\Xi_t \circ B_{kt})$, which allows for the following alternative representation of the latent factor model as a logistic-normal factor model.

Corollary 3.2. *The transition density given in Proposition 3.3 can be written as* $\tilde{\mathbf{w}}_{kt} \sim \mathcal{L}_{n-1}\left(D_n A_{kt} \mathbf{v}_{kt-1}, D_n A_{kt} D_n \Upsilon_k D_n' A_{kt}'\right)$ *and* $w_{n,kt} = 1 - \tilde{\mathbf{w}}_{kt}' \boldsymbol{\iota}_{n-1}$.

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 simple to be realistic in the analysis of compositional data since the components of a Dirichlet composition have a correlation structure determined solely by the normalization operation in the composition. See, Aitchinson and Shen (1980) for a complete discussion of the advantages of the logistic-normal distribution compared to the Dirichlet.

We also present another result that shows how the state space model can be written as a generalized linear model with a local level transition function when the space of the random measures is equipped with suitable operations and norms. Moreover, we show that the probabilistic interpretation is preserved for the lower dimensional set of latent weights.

Define the observation real space $\mathbb{R}^K$ equipped with the inner product $<\mathbf{x},\mathbf{y}>=\sum_{i=1}^{K} x_i y_i$, $\mathbf{x},\mathbf{y}\in\mathbb{R}^K$ and scalar product $a\mathbf{x}=(ax_1,\ldots,ax_K)'$, $\mathbf{x}\in\mathbb{R}^K$, $a\in\mathbb{R}$ operations. Also, define the simplex (state) space, $\mathbb{S}^{n-1}$ equipped with a sum operation (also called perturbation operation), $\mathbf{u}\oplus\mathbf{v}=C(\mathbf{u}\circ\mathbf{v})$, $\mathbf{u},\mathbf{v}\in\mathbb{S}^{n-1}$ and a scalar product operation (also called power transform) $a\odot\mathbf{u}=C((u_1^a,\ldots,u_{n-1}^a)')$, $\mathbf{u}\in\mathbb{S}^{n-1}$, $a\in\mathbb{R}_+$. For details and background, see Aitchinson (1986) and Aitchinson (1992). Billheimer et al. (2001) showed that $\mathbb{S}^{n-1}$ equipped with the perturbation and powering operations is a vector space. Moreover $\mathbb{S}^{n-1}$ is an Hilbert space, i.e. a complete, inner product vector space, equipped with the inner product $<\mathbf{u},\mathbf{v}>_N=\mathbf{u},\mathbf{v}\in\mathbb{S}^{n-1}$ space. These properties enable us to state the following result.

Corollary 3.3. *Let $\mathbf{s}_t=(\mathbf{s}_{1t}',\ldots,\mathbf{s}_{Kt}')'$ be an allocation vector, with $\mathbf{s}_{kt}\sim\mathcal{M}_n(1,\mathbf{w}_{kt})$, $k=1,\ldots,K$, where $\mathcal{M}_n(1,\mathbf{w}_{kt})$ denotes the multinomial distribution, and $\Sigma_t=diag\{\sigma_{1t}^2,\ldots,\sigma_{Kt}^2\}$ a covariance matrix. Then, the state space model given in Proposition 3.3 can be written as*

$$\begin{aligned}
\mathbf{y}_t &= (I_K\otimes\tilde{\mathbf{y}}_t')\mathbf{s}_t+\boldsymbol{\varepsilon}_t,\quad \boldsymbol{\varepsilon}_t\sim\mathcal{N}_K(\mathbf{0},\Sigma_t) && (17)\\
s_{i,kt} &= \begin{cases} 1 & \text{with probability } w_{i,kt}\\ 0 & \text{otherwise}\end{cases} && (18)\\
\mathbf{w}_t &= \boldsymbol{\phi}(\mathbf{z}_t) && (19)\\
\mathbf{z}_{kt} &= \mathbf{z}_{kt-1}\oplus\boldsymbol{\eta}_{kt},\qquad \boldsymbol{\eta}_{kt}\sim\mathcal{L}_{m-1}\left(\mathbf{0},D_n\Upsilon_k D_m'\right) && (20)
\end{aligned}$$

where $\boldsymbol{\phi}(\mathbf{z}_t)=(\phi_{A_{1t}}(\mathbf{z}_{1t}),\ldots,\phi_{A_{Kt}}(\mathbf{z}_{Kt}))$ is a function from $\mathbb{S}^{m-1}$ to $\mathbb{S}^{n-1}$, where the function $\phi_A(\mathbf{z})$ has been defined in 3.2.

The representation in corollary 3.3 shows that the model is a conditionally linear model with link function defined by ϕ_A and a linear local level factor model on the simplex. Also, by extending the $\odot$ product operation to the case of a matrix of real numbers and exploiting the Euclidean vector space structure of $(\mathbb{S}^n,\oplus,\odot)$ allow us to write the transform ϕ_A, for special values of A, as a linear matrix operation between simplices of different dimensions as stated in the following remark. In the following we introduce the symbol $\boxplus$ and define the matrix multiplication operation.

Remark 1. *Let* $\mathbf{z} \in \mathbb{S}^{m-1}$ *be a composition,* A *a* $(n \times m)$ *real matrix and define the matrix multiplication* $A \boxplus \mathbf{z} = C_n \left(\prod_{j=1}^{m} z_j^{a_{1j}}, \ldots, \prod_{j=1}^{m} z_j^{a_{n-1j}} \right)$. *If* A *is such that* $A\boldsymbol{\iota}_m = \mathbf{0}_n$ *and* $a_{im} = -1$, $i = 1, \ldots, n-1$ *and* $a_{n,j} = 0$ $j = 1, \ldots, m$, *the transform defined in proposition 3.2 can be written as* $\phi_A(\mathbf{z}) = A \boxplus \mathbf{z}$.

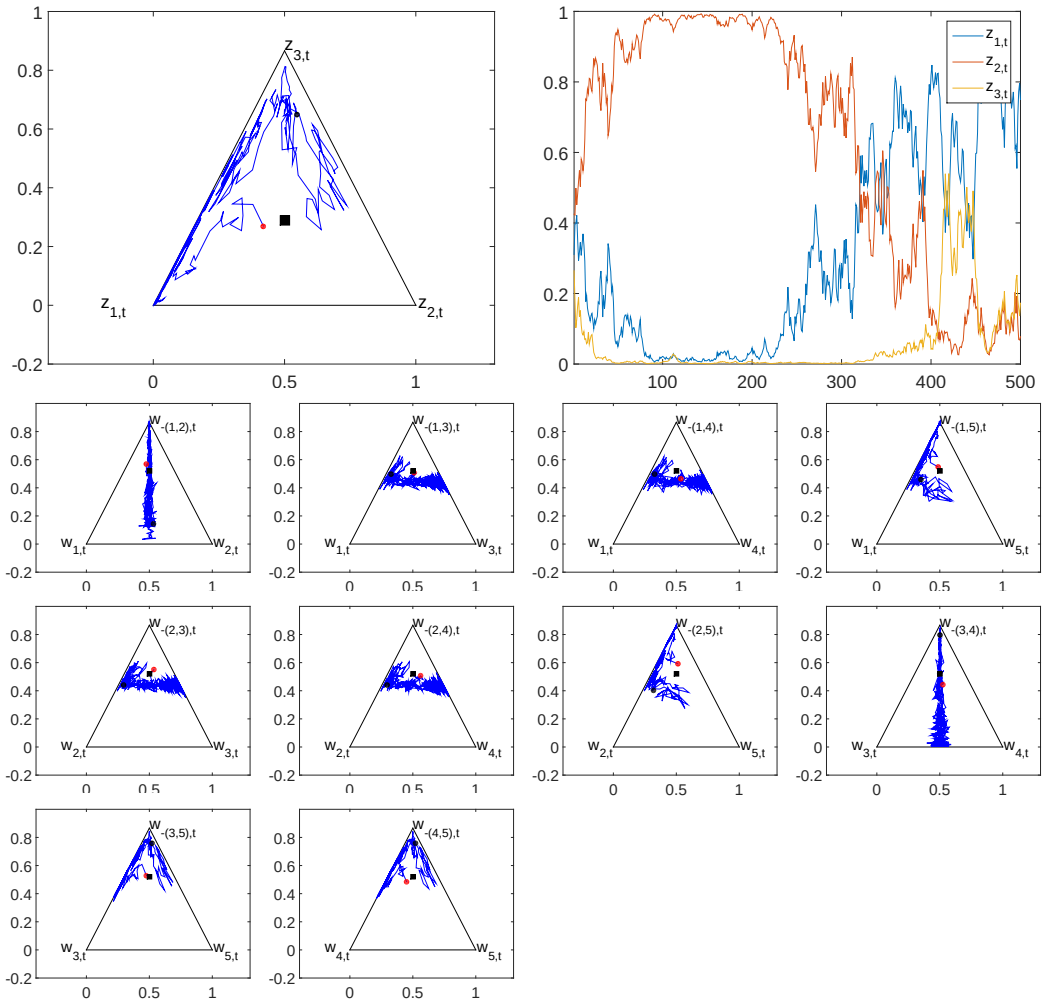

Figure 2: First row: De Finetti's diagram (left) and the time series plot (right) of the ternary $(z_{1,t}, z_{2,t}, z_{3,t})$. Other rows: De Finetti's diagram of the ternary $(w_{i,t}, w_{j,t}, w_{-(i,j)t})$, $j > i$. In each plot the trajectory (blue line), the starting (red) and ending (black) points and the equal weight composition (square).

A simulated example of compositional factor model is given in Fig. 2 by using the De Finetti or ternary diagram (see Cannings and Edwards (1968) and Pawlowsky-Glahn et al. (2015), Appendix A). The first row presents the evolution of three driving

factors $(z_{1,t}, z_{2,t}, z_{3,t})$ by using a De Finetti's diagram (left) and a time series plot (right). The other rows present the pairwise comparisons of the weight dynamics by the De Finetti's diagram of the trajectory (blue line) of the ternary $(w_{i,t}, w_{j,t}, w_{-(i,j),t})$ where $w_{-(i,j),t} = \sum_{l \neq i,j} w_{l,t}$ is the other model total weight. The red and black dots are the initial and final values. Further details of this example are given in section B.1 of the Online Appendix. We refer to the Billheimer et al. (2001) for further details on the algebraic structure of the simplex equipped with the perturbation and powering composition and for a Gibbs sampling scheme for compositional state space model. See also Egozcue et al. (2003), Egozcue and Pawlowskky-Glahn (2005) and Fiŝerová and Hron (2011) for further details on the isometric transforms from the real space to the simplex and and for further geometric aspects and property analysis of operations on the simplex, such as the amalgamation and subcomposition operations. See also Pawlowsky-Glahn and Buccianti (2011) and Pawlowsky-Glahn et al. (2015) for up-to-date and complete reviews on compositional data models.

4 Sequential inference

The analytical solution of the optimal filtering problem is generally not known, also the clustering-based mapping of the predictor weights onto the subset of latent variables requires the solution of an optimization 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 Ξ_t
2. Sequential Monte Carlo approximation of combination weights and predictive densities

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, see for an early treatment Hartigan and Wong (1979), is a method partitioning a set of n vectors of parameters or features of the predictors, $\boldsymbol{\psi}_{it}$, $i = 1, \ldots, n$, into m disjoints sets (clusters), in which each observation belongs to the cluster with the least distance. Moreover, the sequential k-means algorithm is easy to parallelize and it has been done on multi core CPU and GPU computing environments, see Favirar et al. (2008) and the reference therein. The details of the algorithm and its parallel implementation are given in Appendix A.2.

As regards the sequential filtering we apply sequential Monte Carlo as in Billio et al. (2013). Let $\boldsymbol{\theta}_t \in \Theta$ be the parameter vector of the combination model, that is $\boldsymbol{\theta}_t = (\log \sigma_{1t}^2, \ldots, \log \sigma_{Kt}^2, \text{vecd}(\Upsilon_{1t}), \ldots, \text{vecd}(\Upsilon_{Kt}))$. Let $\mathbf{w}_t' = (\mathbf{w}_{1t}', \ldots, \mathbf{w}_{kt})$ the vector of weights, and $\mathbf{u}_{1:t} = (\mathbf{u}_1, \ldots, \mathbf{u}_t)$ the collection of vectors $\mathbf{u}_t$ from time 1 to time t. Following Kitagawa (1998), Kitagawa and Sato (2001), and Liu and West (2001), 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 combination model writes in the state space form

$$\begin{aligned}
\mathbf{y}_t &\sim p(\mathbf{y}_t|\mathbf{w}_t^\theta, \tilde{\mathbf{y}}_t) && \text{(measurement density)} && (21)\\
\mathbf{w}_t^\theta &\sim p(\mathbf{w}_t^\theta|\mathbf{w}_{t-1}^\theta, \mathbf{y}_{1:t-1}, \tilde{\mathbf{y}}_{1:t-1}) && \text{(transition density)} && (22)\\
\mathbf{w}_0^\theta &\sim p(\mathbf{w}_0^\theta) && \text{(initial density)} && (23)
\end{aligned}$$

where the measurement density is

$$p(\mathbf{y}_t|\mathbf{w}_t^\theta, \tilde{\mathbf{y}}_t) \propto \prod_{k=1}^{K} \sum_{i=1}^{n} w_{i,kt} \mathcal{N}(\tilde{y}_{it}, \sigma_{kt}^2) \tag{24}$$

and the transition density is the probability density function of the distribution given in equation 16, that is

$$p(\mathbf{w}_t|\boldsymbol{\theta}_t, \mathbf{w}_{t-1}^\theta, \mathbf{y}_{1:t-1}, \tilde{\mathbf{y}}_{1:t-1}) \propto \tag{25}$$

$$\begin{aligned}
&\propto \prod_{k=1}^{K} \delta_{1-\boldsymbol{\iota}_{n-1}\tilde{\mathbf{w}}_{kt}}(w_{n,kt}) \left(\prod_{j=1}^{n-1} w_{j,kt}\right)^{-1} \prod_{j=1}^{n-1} \exp\Big(-\frac{1}{2}\Big(\log(w_{j,kt}/w_{n,kt}) \\
&-\tilde{\tilde{A}}_{kt} D_m \boldsymbol{\nu}_{kt-1}\Big)(\tilde{\tilde{A}}_{kt} D_m \Upsilon_t D_m' \tilde{\tilde{A}}_{kt}')^{-1} \Big(\log(w_{j,kt}/w_{n,kt}) - \tilde{\tilde{A}}_{kt} D_m \boldsymbol{\nu}_{kt-1}\Big)'\Big) \qquad (26)
\end{aligned}$$

The state predictive and filtering densities are

$$\begin{aligned}
p(\mathbf{w}_{t+1}^\theta|\mathbf{y}_{1:t}, \tilde{\mathbf{y}}_{1:t}) &= \int_{\mathcal{W}} p(\mathbf{w}_{t+1}^\theta|\mathbf{w}_t^\theta, \mathbf{y}_{1:t}, \tilde{\mathbf{y}}_{1:t}) p(\mathbf{w}_t^\theta|\mathbf{y}_{1:t}, \tilde{\mathbf{y}}_{1:t}) d\mathbf{w}_t^\theta && (27)\\
p(\mathbf{w}_{t+1}^\theta|\mathbf{y}_{1:t+1}, \tilde{\mathbf{y}}_{1:t+1}) &= \frac{p(\mathbf{y}_{t+1}|\mathbf{w}_{t+1}^\theta, \tilde{\mathbf{y}}_{t+1}) p(\mathbf{w}_{t+1}^\theta|\mathbf{y}_{1:t}, \tilde{\mathbf{y}}_{1:t})}{p(\mathbf{y}_{t+1}|\mathbf{y}_{1:t}, \tilde{\mathbf{y}}_{1:t})} && (28)
\end{aligned}$$

The marginal predictive density of the observable variables is

$$p(\mathbf{y}_{t+1}|\mathbf{y}_{1:t}) = \int_{\mathcal{Y}} p(\mathbf{y}_{t+1}|\mathbf{y}_{1:t}, \tilde{\mathbf{y}}_{t+1}) p(\tilde{\mathbf{y}}_{t+1}|\mathbf{y}_{1:t}) d\tilde{\mathbf{y}}_{t+1}$$

where $p(\mathbf{y}_{t+1}|\mathbf{y}_{1:t}, \tilde{\mathbf{y}}_{t+1})$ is defined as

$$\int_{\mathcal{W}\times\mathcal{Y}^t} p(\mathbf{y}_{t+1}|\mathbf{w}^{\theta}_{t+1}, \tilde{\mathbf{y}}_{t+1}) p(\mathbf{w}^{\theta}_{t+1}|\mathbf{y}_{1:t}, \tilde{\mathbf{y}}_{1:t}) p(\tilde{\mathbf{y}}_{1:t}|\mathbf{y}_{1:t-1}) d\mathbf{w}^{\theta}_{t+1} d\tilde{\mathbf{y}}_{1:t}$$

and represents the conditional predictive density of the observable given the past values of the observable and of the predictors. Further details of the algorithm is given in Appendices A.3, A.2 and B.2.

5 Results

The first example focuses on replicating the daily Standard & Poor 500 (S&P500) index return and predicting the economic value of tail events like Value-at-Risk. As a second example we consider the extended Stock and Watson (2005) dataset, which includes 142 series sampled at a quarterly frequency from 1959Q1 to 2011Q2. Finally, we compare the computational speed of CPU with GPU in the implementation of our combination algorithm for the financial and macro applications.

5.1 Predicting Standard & Poor 500 (S&P500)

The econometrician interested in predicting this index (or a transformation of it as the return) 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 predict the price of each stock i and then aggregate them using an approximation of the unknown weighting scheme.

We propose an alternative 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. We collect the S&P500 index and 3712 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 observation. 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. S&P500 and cross-section average statistics are reported in Table B.1 in section B.4 of the Online Appendix. We produce a density forecast for each of the stock prices and then apply our density combination scheme to compute clustered weights and a combined density forecast of the index. The output is a density forecast of the index with clustered weights that indicate the relative forecasting importance of these clusters. That is,

a side output of our method is that it produces a replication strategy of the index, providing evidence of which assets track more accurately the aggregate index. We leave a detailed analysis of this last topic for further research.

Individual model estimates

We estimate a Normal GARCH(1,1) model and a t-GARCH(1,1) model via maximum likelihood (ML) using rolling samples of 1250 trading days (about five years) for each stock return:

$$y_{it} = c_i + \kappa_{it}\zeta_{it} \tag{29}$$

$$\kappa_{it}^2 = \theta_{i0} + \theta_{i1}\zeta_{i,t-1}^2 + \theta_2\kappa_{i,t-1}^2 \tag{30}$$

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 density forecasts from January 1, 2007 to December 31, 2009 using the above equations and the first day ahead forecast refers to January 1, 2007. Our out-of-sample (OOS) 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 the following week events. The predictive densities are formed by substituting the ML estimates for the unknown parameters $(c_i, \theta_{i0}, \theta_{i1}, \theta_{i2}, \nu_i)$.

As first step, we apply a sequential cluster analysis to our forecasts. We compute two clusters for the Normal GARCH(1,1) model class and two clusters for the t-GARCH(1,1) model class. The first two are characterized by low and high volatility density predictions from Normal GARCH(1,1) models; the third and the fourth ones are characterized by thick or no thick tail density predictions from t-GARCH(1,1) models.[3] A detailed description of the cluster dynamics is given in section B.4 the Online Appendix.

Weight patterns, model incompleteness and signals of instability

For convenience, we specified the parameter matrices B_{kt} in equation (11), the cluster weights, as equal weights.[4] We also allow for model incompleteness to be modeled as a time-varying process and estimate σ_{kt}^2 in (5). We label it DCEW-SV and compare it with a combination scheme where $\sigma_{kt}^2 = \sigma_k^2$ is time-invariant and label

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

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

that as DCEW. We compare our two combination methods, DCEW and DCEW-SV described in section 5.1 to the standard no predictability white noise benchmark and also apply the Normal GARCH(1,1) model and the t-Student GARCH(1,1) model to the index log returns. The comparison is done by applying the predictive ability measures defined in Appendix B.3.

Plots of the estimated weights $z_{k,t}$ defined in Corollary 3.1 are shown in Figure 3. The same figure shows the De Finetti's diagrams for a pairwise comparison of the weight dynamics. In the diagrams the blue line represents the trajectory of the ternary $(z_{i,t}, z_{j,t}, z_{-(i,j),t})$ where $z_{-(i,j),t} = \sum_{l \neq i,j} z_{l,t}$ is the other model total weight. The red and black dots are the initial and final values.

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 tail densities and our combination scheme takes advantage of this information and assigns to cluster 3 more weight. RW and GARCH forecasts based on the index are less informative and before these models can show forecasts of negative returns they need evidence that the index is declining. 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. Summarizing, it is seen that the t-GARCH(1,1) cluster with small degrees of freedom has most of the period the largest weight. What implications this may have for constructing model combinations that forecast more accurately is a topic for further research.

Signals of model incompleteness and instability are shown in the top right panel of Figure 3 where plots of the posterior mean estimate for σ_{kt}^2 in the DCEW-SV scheme are presented. The estimates have 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.

From the results so far, we conclude that the combination of several time-varying volatility models with time-varying cluster weights copes with instability in our set of data. There is a clear signal of increased model incompleteness after the 2008 crisis. Individual flexible models that focus more on jumps in volatility and use data on realized volatility may be included in the analysis. This is an interesting topic of

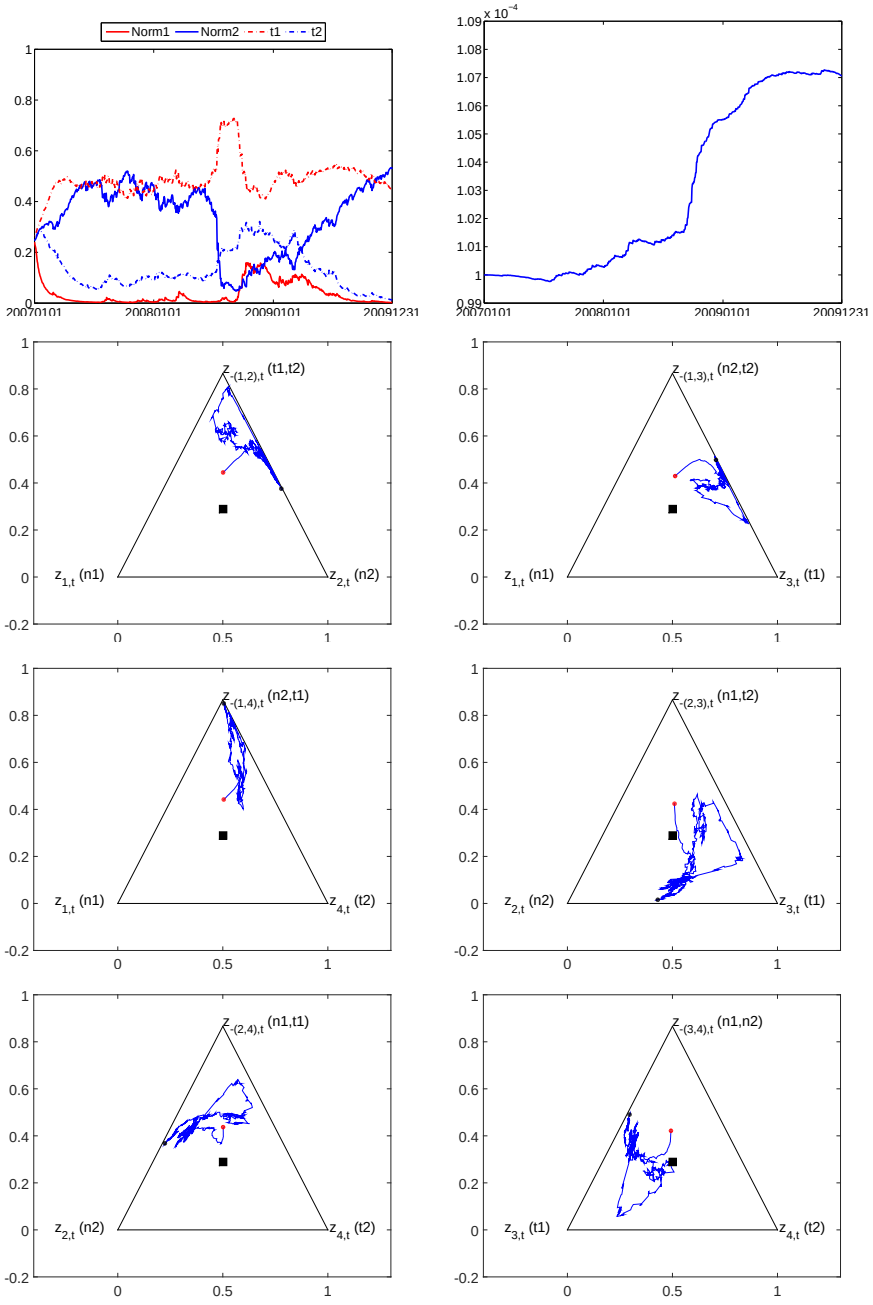

Figure 3: Top left: the mean logistic-normal weights for the two normal GARCH clusters, labeled in the graph "Norm1" and "Norm2", and for the two t-GARCH clusters, labeled in the graph "t3" and "t4". Top right: posterior mean estimate for σ_{kt} in the scheme DCEW-SV. Other rows: De Finetti's diagram for the pairwise subcomposition comparison between model weights over time. In each plot the trajectory of the ternary $(z_{it}, z_{jt}, z_{-(i,j)t})$, $j > i$ (blue line), the starting point (red dot), the ending point (black dot) and the equal weight composition (square).

further research.

Forecast accuracy and economic value

Out-of-sample forecasting result are presented in Table 1. Our combination schemes produce the lowest RMSPE and CRPS and the highest LS. The results indicate that the combination schemes are statistically superior to the no predictability benchmark. The Normal GARCH(1,1) model and t-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 two individual models are inferior to our combination schemes. Therefore, we conclude that our strategy to produce forecasts from a large set of assets, cluster them in groups and combine them to predict the S&P500 produces very accurate point and density forecasts that are superior to no predictability benchmark and classical strategies of modeling directly the index.

Apart from forecasting accuracy, we investigate whether the results documented in the previous paragraphs also possess some economic value. Given that our approach produces complete predictive densities for the variable of interest, it is particularly suitable to compute tail events and, therefore, Value-at-Risk (VaR) measures, 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 predicted. 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.

The last two columns of Table 1 show that the number of violations for all models is high and well above 1%, with the RW higher than 20%. The dramatic events in our sample, including the Lehman default and all the other features of the US financial crisis, explain the result. However, the two combination schemes provide the best statistics, with violations almost 50% lower than the best individual model and losses at least 15% lower than the best individual models. The DCEW-SV provides the most accurate results, but the difference with DCEW is marginal. 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.

Finally, Table B.6 in Appendix B.6 compares the execution time of the GPU parallel implementation of our density combination strategy and the CPU multi-core

implementation and show large gains from GPU parallelization.

	RMSPE	LS	CRPS	Violation	Loss
WN	1.8524	-9.0497	0.0102	20.3%	-50.1%
Normal GARCH	1.8522	-4.1636**	0.0096**	16.5%	-42.2%
t-GARCH	1.8524	-2.7383**	0.0094**	11.4%	-32.9%
DCEW	**1.8122****	**2.2490****	**0.0091****	6.6%	-28.1%
DCEW-SV	1.8165**	2.2060**	**0.0091****	**6.5**%	**-27.7**%

Table 1: Forecasting results for next day S&P500 log returns. For all the series are reported the: root mean square prediction error (RMSPE), logarithmic score (LS) and the continuous rank probability score (CRPS). 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 statistically 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 realized value exceeds the 1% Value-at-Risk (VaR) predicted by the different models over the sample and the column "Loss" reports the cumulative total loss associated to the violations.

5.2 A large macroeconomic dataset

As a second example, 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 B.3, in section B.5 of the Online Appendix. The dataset includes only revised series and not vintages of real-time data, when data are revised. See Aastveit et al. (2014) for a real-time application (with a dataset that includes fewer series) of density nowcasting and on the role of model incompleteness over vintages and time. In order to deal with stationary series, we apply the series-specific transformation suggested in Stock and Watson (2005). Let y_{it} with $i = 1, \dots, n$ and $t = 1, \dots, T$, be the set of transformed variables.

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

$$y_{it} = \alpha_i + \beta_i y_{it-1} + \zeta_{it}, \quad \zeta_{it} \sim \mathcal{N}(0, \sigma_i^2) \tag{31}$$

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. Since we are interested in an interpretation of the clusters over the full sample, differently than in the previous financial application, we impose that cluster allocation of each model is fixed over the

forecasting vintages, i.e. $\Xi_t = \Xi$, $t = 1, \ldots, T$. The first 102 observations, from 1959Q3 to 1984Q1, are used as initial in-sample (IS) period to fit AR(1) models to all the individual series and construct the clusters. We assume alternatively 5 and 7 clusters. A detailed description of the 7 clusters is provided in Table B.4 in section B.5 of the Online Appendix, together with further results.

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 (IS) period; the remaining 106 observations from 1985Q1-2011Q2 are used as an OOS period. The AR models are estimated recursively and $h-$step ahead (Bayesian) $t-$Student predictive 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. Clusters are, however, not updated and kept the same as the ones estimated in the IS period.

We predict four different series often considered core variables in monetary policy analysis: real GDP growth, inflation measured as price deflator growth, 3-month Treasury Bill rate and total employment. We consider $h = 1, 2, 3, 4, 5$ step-ahead horizons. For all the variables to be predicted, we apply an AR(1) as benchmark model.

As we described in Section 2, we consider two alternative strategies for the specification of the parameter matrices B_{kt}: equal weights and score recursive weights, where in the second case we fix $g_i = LS_{i,h}$ for the various horizons h presented in the following subsection. Further, the predictive densities can be combined with each of the four univariate series and/or with a multivariate approach. Following the evidence in Appendix B.5 we apply two clusters, $k = 5$ and 7. We note that we keep the volatility of the incompleteness term constant. To sum up, we have eight cases defined as UDCEW5 (univariate combination based on 5 clusters with equal weights within clusters), MDCEW5 (multivariate combination based on 5 clusters with equal weights within clusters), UDCLS5 (univariate combination based on 5 clusters with recursive log score weights within clusters), MDCLS5 (multivariate combination based on 5 clusters with recursive log score weights within clusters), UDCEW7 (univariate combination based on 7 clusters with equal weights within clusters), MDCEW7 (multivariate combination based on 7 clusters with equal weights within clusters), UDCLS7 (univariate combination based on 7 clusters with recursive log score weights within clusters), MDCLS7 (multivariate combination based on 7 cluster with recursive

log score weights within clusters).

Apart from the AR(1) benchmark we also compare our combinations to a benchmark that is specified as Dynamic Factor Model (DFM) with 5 factors described in Stock and Watson (2012). This DFM expresses each of the n time series as a component driven by the latent factors plus ad idiosyncratic disturbance. More precisely:

$$\mathbf{y}_t = \Lambda \mathbf{f}_t + \boldsymbol{\varepsilon}_t, \quad \Phi(\mathrm{L})\mathbf{f}_t = \boldsymbol{\eta}_t, \tag{32}$$

where the $\mathbf{y}_t = (y_{1,t}, \ldots, y_{n,t})'$ is an $n \times 1$ vector of observed series, $\mathbf{f}_t = (f_{1,t}, \ldots, f_{r,t})'$ is an r vector of latent factors, Λ is a $n \times r$ matrix of factors loadings, $\Phi(\mathrm{L})$ is an $r \times r$ matrix lag polynomial, $\boldsymbol{\varepsilon}_t$ is an n vector of idiosyncratic components and $\boldsymbol{\eta}_t$ is an r vector of innovations. In this formulation the term $\Lambda \mathbf{f}_t$ is the common component of $\mathbf{y}_t$. Bayesian estimation of the model described in equation (32) is carried out using Gibbs Sampling given in Koop and Korobilis (2009).

Weight patterns and forecasting results

Table 2 reports the results to predict real GDP growth, inflation measured by using the price deflator of GDP growth, 3-month Treasury Bills and total employment for five 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 Bayesian DFM. The Bayesian DFM model provides more accurate forecasts than the AR(1) for real GDP and inflation at shorter horizons and gives mixed evidence for interest rates and unemployment, 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 (MCDLS7), resulting in the best statistics 56 times over 60. 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 in large Bayesian VAR, see e.g. Bańbura et al. (2010) and Koop and Korobilis (2013). Fan charts in Figure B.8 in the Appendix B.5 show that the predictions are accurate even at our longest horizon, $h = 55$. The variable with low predictive gains is inflation, although our method provides credibly more accurate scores at (at least) 5% credible level in 8 cases over 15, but none in terms of point forecasting. The multivariate combination based on 5 clusters and equal weights yields accurate forecasts, see clusters MCDEW5. We conclude that combining models using multiple clusters with cluster-based weights provides substantial forecast gains in most cases. Additional gains may be obtained by playing with a more detailed cluster grouping and different performance scoring rules

	h=1			h=2			h=3			h=4			h=5		
	PE	LS	CRPS	PE	LS	CRPS	PE	LS	CRPS	PE	LS	CRPS	PE	LS	CRPS
RGDP															
AR	0.647	-1.002	0.492	0.658	-1.005	0.496	0.671	-1.007	0.501	0.676	-1.009	0.503	0.682	-1.009	0.506
BDFM	0.649	-1.091	0.382**	0.651	-1.066	0.385**	0.654	-1.138	0.388**	0.652	-1.060	0.384**	0.655	-1.099	0.388**
UDCEW5	0.644	-0.869	0.333**	0.655	-0.893	0.340**	0.657*	-0.900	0.341**	0.655*	-0.902	0.341**	0.658*	-0.912	0.343**
MDCEW5	0.630	-0.928	0.326**	0.645	-0.987	0.336**	0.638*	-0.924	0.330**	0.637*	-0.897	0.328**	0.636*	-0.844	0.324**
UDCLS5	0.773	-1.306	0.464	0.663	-1.275	0.433**	0.687	-1.339	0.446**	0.689	-1.327	0.448**	0.715	-1.380	0.481
MDCLS5	0.725	-1.145	0.505	0.591*	-1.071	0.365**	0.581**	-1.041	0.340**	0.591*	-1.079	0.354**	0.557*	-1.005	0.358**
UDCEW7	0.649	-0.875	0.334**	0.652	-0.880	0.335**	0.655	-0.889	0.337**	0.654	-0.886	0.336**	0.657*	-0.891	0.338**
MDCEW7	0.642	-0.979	0.334**	0.648	-1.012	0.338**	0.652*	-1.016	0.342*	0.651	-1.015	0.339**	0.654*	-1.009	0.342**
UDCLS7	0.646	-0.868*	0.332**	0.645	-0.905	0.338**	0.650*	-0.918	0.341**	0.655	-0.939	0.352**	0.657*	-0.914	0.342**
MDCLS7	**0.596***	**-0.586****	**0.275****	**0.586***	**-0.582****	**0.275****	**0.607****	**-0.632****	**0.288****	**0.588***	**-0.637****	**0.287****	**0.610****	**-0.634****	**0.286****
GDP deflator															
AR	0.220	-0.933	0.356	0.214	-0.932	0.357	0.206	-0.932	0.358	0.207	-0.932	0.359	0.208	-0.932	0.361
BDFM	0.220	-0.676**	0.123*	0.214	-0.225	0.441	0.221	-0.768**	0.373	0.223	-1.005	0.378	0.276	-1.072	0.382
UDCEW5	0.230	-0.429	0.169	0.220	-0.427	0.167	0.212	-0.422	0.165	0.214	-0.425	0.166	0.213	-0.426	0.166
MDCEW5	0.204	-0.053	0.110*	0.205	-0.285	0.115	0.203	-0.234	0.114	0.202	-0.167	0.112	0.204	-0.194	0.113
UDCLS5	0.485	-1.085	0.354	0.313	-1.001	0.294	0.259	-0.873	0.250	0.241	-0.875	0.248	0.228	-0.892	0.252
MDCLS5	0.291	-0.280	0.309	0.161	0.003	0.143**	0.143	0.031	0.125**	0.132	0.072	0.122*	0.159	-0.226	0.147*
UDCEW7	0.223	-0.425**	0.166**	0.214	-0.420	0.164**	0.207	-0.416	0.163	0.209	-0.416*	0.163*	0.210	-0.416	0.164
MDCEW7	0.208	-0.214**	0.115**	0.200*	-0.186*	0.111**	0.197*	-0.172**	0.109**	0.197	-0.175*	0.110*	0.199	-0.200	0.111
UDCLS7	0.235	-0.507**	0.179**	0.220	-0.519	0.180**	0.224	-0.514	0.179	0.221	-0.516	0.179	0.214	-0.475	0.171
MDCLS7	**0.197**	**0.436****	**0.098****	**0.183**	**0.462****	**0.092****	**0.165**	**0.571***	**0.083***	**0.160**	**0.570****	**0.082****	**0.175**	**0.495**	**0.088**
3-month Treasury Bills															
AR	0.569	-1.058	0.363	0.605	-1.074	0.374	0.518	-1.038	0.343	0.530	-1.037	0.353	0.545	-1.041	0.358
BDFM	0.522*	-1.190	0.359	0.694	-1.394	0.386	0.545	-1.092	0.392	0.552	-1.092	0.396	0.541	-1.089	0.401
UDCEW5	0.519	-0.778**	0.288**	0.521	-0.782**	0.288	0.509	-0.772**	0.283	0.517	-0.782**	0.288*	0.525	-0.791**	0.292*
MDCEW5	0.517**	-0.764**	0.285**	0.506	-0.752**	0.279**	0.502*	**-0.749****	**0.276****	**0.506****	**-0.755****	**0.278****	0.505**	-0.751**	0.278**
UDCLS5	0.740	-1.254	0.448	0.678	-1.301	0.453	0.532	-1.210	0.381	0.528	-1.216	0.385	0.584	-1.286	0.424
MDCLS5	0.710	-1.322	0.491	0.688	-1.297	0.454	0.491**	-1.143	0.346	0.487	-1.143	0.351	0.572**	-1.196	0.378
UDCEW7	0.525	-0.783**	0.289*	0.526	-0.784**	0.289*	0.514	-0.768**	0.284*	0.518	-0.774**	0.286*	0.522	-0.786**	0.289*
MDCEW7	0.526	-0.775**	0.289*	0.527	-0.777**	0.290*	0.515	-0.761**	0.283*	0.516	-0.765**	0.284*	0.513	-0.766**	0.283*
UDCLS7	0.512	-0.773**	0.284*	0.521	-0.799**	0.291*	0.514	-0.770**	0.284*	0.519	-0.783**	0.286*	0.521	-0.793**	0.289*
MDCLS7	**0.488****	**-0.725****	**0.270****	**0.484****	**-0.771****	**0.275***	**0.515****	-0.755**	0.283	0.513**	-0.771**	0.283	**0.496****	**-0.736****	**0.275****
Employment															
AR	0.564	-0.995	0.447	0.582	-0.999	0.454	0.597	-1.003	0.460	0.612	-1.007	0.464	0.622	-1.009	0.468
BDFM	0.571	-1.064	0.339**	0.565	-1.057	0.614	0.956	-1.192	0.907	0.724	-1.226	0.922	0.876	-1.892	0.998
UDCEW5	0.585**	-0.906**	0.308**	0.582**	-0.889**	0.307**	0.579	-0.955**	0.305**	0.584	-0.931**	0.308**	0.587	-0.951**	0.311**
MDCEW5	0.541**	-0.926**	0.277**	0.554**	-0.960**	0.284**	0.558	-0.917**	0.285**	0.560**	-0.740**	0.284**	0.571**	-0.790**	0.294**
UDCLS5	0.752	-1.301	0.456	0.548	-1.265	0.414	0.565	-1.305	0.426	0.648	-1.372	0.472	0.628	-1.335	0.438
MDCLS5	0.654	-1.180	0.568	0.416	-0.964	0.325	0.487	-1.010	0.338	0.478*	-0.976	0.340	0.569	-1.076	0.360
UDCEW7	0.535**	-0.801**	0.283**	0.555**	-0.828*	0.290**	0.570	-0.854**	0.298**	0.577	-0.867**	0.303**	0.583*	-0.881**	0.306**
MDCEW7	0.523**	-0.735**	0.266**	0.548**	-0.775**	0.278**	0.565	-0.827**	0.288**	0.571	-0.855**	0.293**	0.578*	-0.885**	0.297**
UDCLS7	0.552**	-0.767**	0.289**	0.535**	-0.805**	0.294**	0.562	-0.849**	0.302**	0.572	-0.878**	0.320**	0.588*	-0.895**	0.313**
MDCLS7	**0.516****	**-0.452****	**0.236****	**0.440****	**-0.437****	**0.219****	**0.507**	**-0.479****	**0.237****	**0.495***	**-0.488****	**0.241****	**0.560****	**-0.680****	**0.275****

Table 2: Forecasting results for h steps ahead. For all the series: root mean square prediction 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 statistically 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).

for weights associated with models inside a cluster. Figure 4 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. The diagrams show a substantial time stability of the two largest weights, a weight composition that is far from the equal weight case and a substantial relevance of the *sixth* cluster for all variables and horizons.

From the analysis of the weight time patterns in Figure 5 (see Figure B.6 in Appendix B.5 for weights in the univariate combination), we notice that the weights for the univariate are often less volatile than the weights in the multivariate approach.

All figures confirm he result that the *sixth* cluster has the largest weight, but several other clusters have large positive weights, like 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, although this role may differ across variables and horizons.

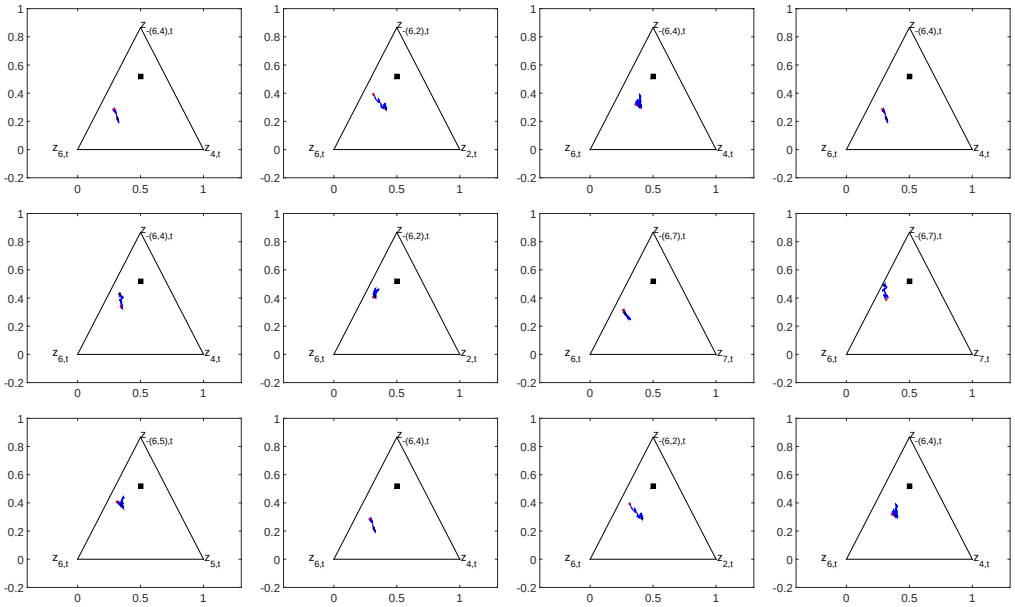

Figure 4: De Finetti's diagrams for the dynamic comparison of the two largest weights. Rows: diagrams for the four series of interest (real GDP growth rate, GDP deflator, Treasury Bills, employment). Columns: forecast horizons (1, 3 and 5 quarters). In each plot the trajectory (blue line), the starting (red) and ending (black) points and the equal weight composition (square).

The forecast gains are similar across horizons for the five 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. The lowest improvements are evident when predicting the 3-month Treasury Bills. Despite these consistent gains over horizons, the combination weights in Figure 5 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 more clear 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

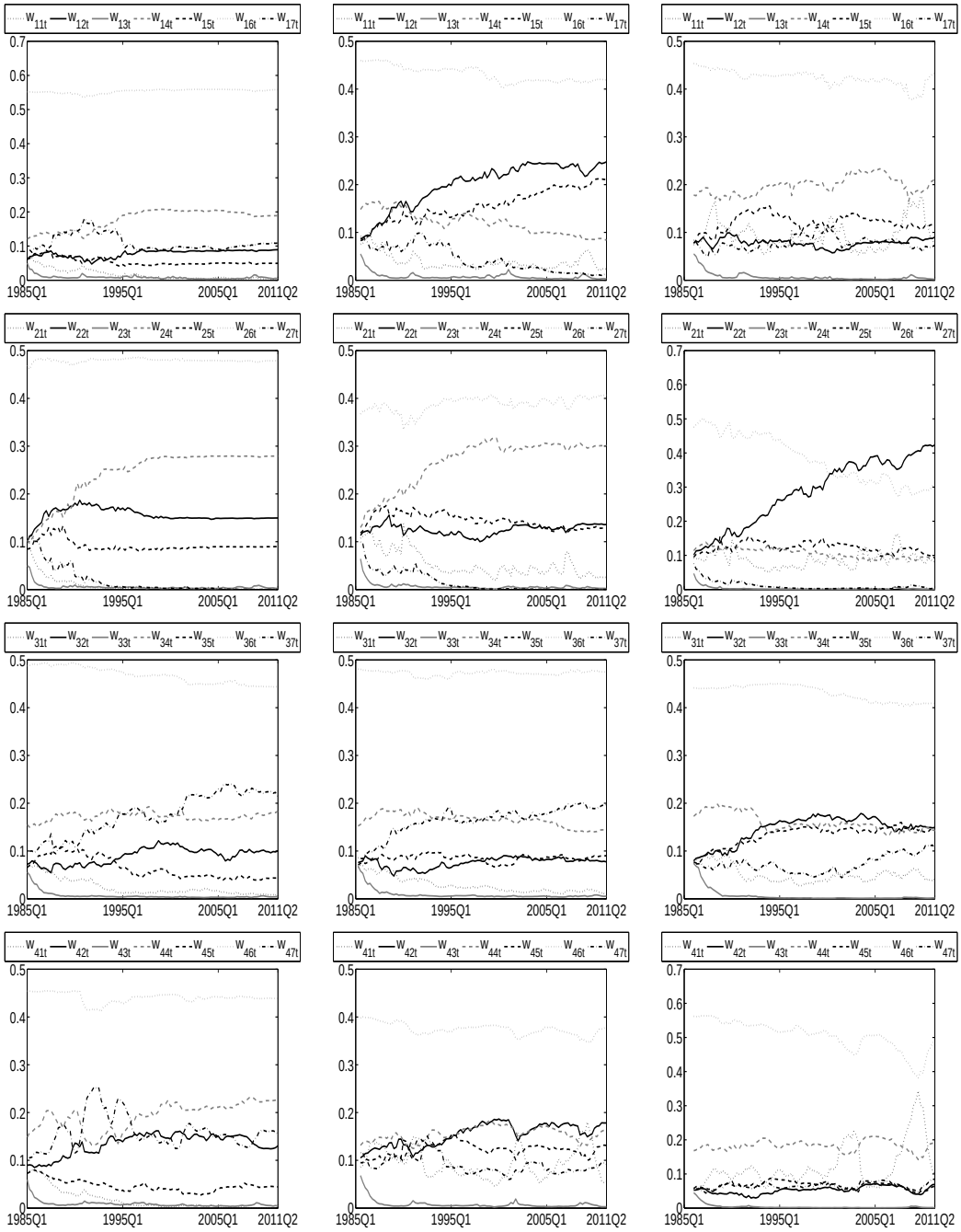

Figure 5: In each plot the logistic-normal weights (different lines) for the multivariate combination model are given. Rows: plot for the four series of interest (real GDP growth rate, GDP deflator, Treasury Bills, employment). Columns: forecast horizons (1, 3 and 5 quarters).

the other two predicted variables.

Figure B.7 in the Online Appendix 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 in Figure 5 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 B.7 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.

Evidence is similar for the GDP deflator and employment, but this finding is less clear for bond returns. For this variable, MDCEW5 also predicts accurately. Also notice that cluster 3, which includes the 3-month Treasury Bills, has the lowest weight in Figures 5. 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 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 is an interesting line of research to pursue.

6 Conclusions

We proposed in this paper a Bayesian nonparametric model to construct a time-varying weighted combination of many predictive densities that can deal with large data sets in economics and finance. The model is based on clustering the set of predictive densities in mutually exclusive subsets and on a hierarchical specification of the combination weights. This modeling 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.

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 applications, we show how 7000 predictive densities based on US individual stocks can be combined to replicate the daily Standard & Poor 500 (S&P500) index return and predict the economic value of tail events like Value-at-Risk. In the macroeconomic exercise, we show that combining models for multiple series 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 very unstable, strengthening the use of density combinations.

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. We notice also a potential fruitful connection between our approach and research in the field of dynamic portfolio allocation.

References

Aastveit, K. A., Ravazzolo, F., and van Dijk, H. K. (2014). Combined density nowcasting in an uncertain economic environment. Technical Report 14-152/II, Tinbergen Institute.

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.

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.

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

Bassetti, F., Casarin, R., and Leisen, F. (2014). Beta-product dependent Pitman-Yor processes for Bayesian inference. *Journal of Econometrics*, 180:49–72.

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.

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*, forthcoming.

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

Conflitti, C., De Mol, C., and Giannone, D. (2012). Optimal combination of survey forecasts. Technical report, ECARES Working Papers 2012-023, ULB – Universite Libre de Bruxelles.

Del Negro, M., Hasegawa, B. R., and Schorfheide, F. (2014). Dynamic prediction pools: An investigation of financial frictions and forecasting performance. Technical report, NBER Working Paper 20575, PIER WP 14-034.

Egozcue, J. J. and Pawlowskky-Glahn, V. (2005). Groups of parts and their balances in compositional data analysis. *Mathematical Geology*, 37:795–828.

Egozcue, J. J., Pawlowskky-Glahn, V., Mateu-Figueras, G., and Barcelo-Vidal, C. (2003). Isometric logratio transformations for compositional data analysis. *Mathematical Geology*, 35:279–300.

Einav, L. and Levin, J. (2014). Economics in the age of big data. *Science*, 346(6210):715–718.

Favirar, R., Rebolledo, D., Chan, E., and Campbell, R. (2008). A parallel implementation of K-Means clustering on GPUs. *Proceedings of 2008 International Conference on Parallel and Distributed Processing Techniques and Applications*, 2:14–17.

Fawcett, N., Kapetanios, G., Mitchell, J., and Price, S. (2015). Generalised density forecast combinations. *Journal of Econometrics*, 188:150–165.

Fîšerová, E. and Hron, K. (2011). On the interpretation of orthonormal coordinates for compositional data. *Mathematical Geosciences*, 43:455–468.

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.

Granger, C. W. J. (1998). Extracting information from mega-panels and high-frequency data. *Statistica Neerlandica*, 52:258–272.

Grunwald, G. K., Raftery, A. E., and Guttorp, P. (1993). Time series of continuous proportions. *Journal of the Royal Statistical Society B*, 55:103–116.

Hartigan, J. A. and Wong, M. A. (1979). Algorithm as 136: A K-Means clustering algorithm. *Journal of the Royal Statistical Society, Series C*, 28:100–108.

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.

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.

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.

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.

Lindley, D. V., Tversky, A., and Brown, R. V. (1979). On the reconciliation of probability assessments. *Journal of the Royal Statistical Society. Series A*, 142:146–180.

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.

Müller, P. and Mitra, R. (2013). Bayesian nonparametric inference - why and how. *Bayesian Analysis*, 8:269–302.

Müller, P. and Quintana, F. (2010). Random partition models with regression on covariates. *Journal of Statistical Planning and Inference*, 140:2801–2808.

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

Pawlowsky-Glahn, V. and Buccianti, A. (2011). *Compositional Data Analysis: Theory and Applications*. Wiley.

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

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.

Raftery, A. E., Kárný, M., and Ettler, P. (2010). Online prediction under model uncertainty via Dynamic Model Averaging: Application to a cold rolling mill. *Technometrics*, 52:52–66.

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. (2004). Combination forecasts of output growth in a seven - country data set. *Journal of Forecasting*, 23:405–430.

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.

Waggoner, D. F. and Zha, T. (2012). Confronting model misspecification in macroeconomics. *Journal of Econometrics*, 171:167–184.

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

A Appendix

A.1 Proofs of the results in sections 2 and 3

Proof of Proposition 2.1 The marginal predictive density is obtained by integrating out the predictors with respect to their distributions. Under regularity condition it is possible to exchange the order of integration and obtain

$$f_{kt}(y_{kt}|\mathbf{w}_{kt}) \stackrel{\text{def}}{=} \int_{\mathbb{R}^n} f_{kt}(y_{kt}|\tilde{\boldsymbol{y}}_t) \prod_{j=1}^{n} f_{jt}(\tilde{y}_{jt}) d\tilde{y}_{jt} \tag{A.33}$$

$$= \sum_{i=1}^{n} w_{k,it} \int_{\mathbb{R}^n} f(y_{kt}|\tilde{y}_{it}, \sigma^2_{kt}) \prod_{j=1}^{n} f_{jt}(\tilde{y}_{jt}) d\tilde{y}_{jt} \tag{A.34}$$

$$= \sum_{i=1}^{n} w_{k,it} \int_{\mathbb{R}} f(y_{kt}|\tilde{y}_{it}, \sigma^2_{kt}) f_{it}(\tilde{y}_{it}) d\tilde{y}_{it} \tag{A.35}$$

where $f(y|\vartheta, \sigma^2)$ is the pdf of the normal distribution $\mathcal{N}(\vartheta, \sigma^2)$. Now, by letting $\sigma^2_{kt} \to 0$ for all k, one has that $f_{kt}(y_{kt})$ converges to

$$\sum_{i=1}^{n} w_{k,it} \int_{\mathbb{R}} \delta_{\tilde{y}_{it}}(y_{kt}) f_{it}(\tilde{y}_{it}) d\tilde{y}_{it} = \sum_{i=1}^{n} w_{i,kt} f_{it}(y_{kt}) \tag{A.36}$$

$k = 1, \ldots, K$.

Proof of Proposition 3.1 See Aitchinson and Shen (1980), Section 2.

Proof of Corollary 3.1 It follows from 3.1 by taking $\mathbf{v} = \mathbf{v}_{kt}$ and $\mathbf{z} = \mathbf{z}_{kt}$.

Proof of Proposition 3.2 It follows from a direct application of the results in Aitchinson and Shen (1980), Section 2.

Proof of Proposition 3.3 From equations 5-9 it is easy to show that the measurement density for each variable of interest is $y_{kt} \sim \mathcal{N}\left(\tilde{\mathbf{y}}'_t \mathbf{s}_{kt}, \sigma^2_{kt}\right)$ with $\mathbf{s}_{kt} \sim \mathcal{M}_n(1, \mathbf{w}_{kt})$, $k = 1, \ldots, K$, where $\mathcal{M}_n(1, \mathbf{w}_{kt})$ denotes a multinomial distribution, and due to the conditional independence assumption one gets the joint measurement density as the product of the variable specific densities.

As regards the transition density, first observe that, thanks to proposition 3.1, $\mathbf{z}_{kt} = C_m(\exp(\mathbf{v}_{kt}))$ follows $\mathcal{L}_{m-1}(D_m \mathbf{v}_{kt-1}, D_m \Upsilon_k D'_m)$. Then note that the multivariate transform $x_{i,kt} = \sum_{j=1}^{m} \xi_{ij,kt} b_{ij,kt} v_{j,kt}$, $j = 1, \ldots, m$, $i = 1, \ldots, n$ implies that $\mathbf{x}_{kt} = A_{kt} \mathbf{v}_{kt}$, $\mathbf{x}_{kt} \sim \mathcal{N}_n(A_{kt} \mathbf{v}_{kt-1}, A_{kt} \Upsilon_k A'_{kt})$, with $A_{kt} = (\Xi_t \circ B_{kt})$

and that, from Proposition 3.1, $C_n(A_{kt}\mathbf{v}_{kt})$ follows $\mathcal{L}_{n-1}(D_nA_{kt}\mathbf{v}_{kt-1}, D_n\Upsilon_k D_n')$. Without loss of generality, we assume that $B_{kt} = \boldsymbol{\iota}_n\boldsymbol{\iota}_n'$ and that the $n - \tilde{n}_t$ elements in the cluster m correspond to the last $n - \tilde{n}_t$ element of $\tilde{\mathbf{y}}_t$. This implies the following partition of $\Xi_t' = ((\tilde{\Xi}_t, O_{\tilde{n}_t\times 1})', (O_{(n-\tilde{n}_t)\times(m-1)}, \boldsymbol{\iota}_{n-\tilde{n}_t})')$ and of $A_{kt}' = ((\tilde{A}_{kt}, O_{\tilde{n}_t\times 1})', (O_{(n-\tilde{n}_t)\times(m-1)}, \boldsymbol{\iota}_{n-\tilde{n}_t})')$, where $(\tilde{\Xi}_t, O_{\tilde{n}_t\times 1})$ and $(\tilde{A}_{kt}, O_{\tilde{n}_t\times 1})$ are a $(\tilde{n}_t \times m)$ matrices. Note that

$$\begin{aligned} D_nA_{kt} &= (I_{n-1}, -\iota_{n-1})((\tilde{A}_{kt}, (O_{\tilde{n}_t\times 1})', (O_{(n-\tilde{n}_t)\times(m-1)}, \boldsymbol{\iota}_{n-\tilde{n}_t})')' \\ &= ((\tilde{A}_{kt}, -\boldsymbol{\iota}_{\tilde{n}_t})', O'_{(n-\tilde{n}_t-1)\times m})' \\ &= (\tilde{A}_{kt}', O'_{(n-\tilde{n}_t-1)\times(m-1)})'D_m \end{aligned}$$

The result then follows by applying Proposition 3.2 to the set of weights $\mathbf{z}_{j,kt}$, $j = 1, \ldots, m-1$, with transform coefficients $A = (\tilde{A}_{kt}', O'_{(n-\tilde{n}_t)\times(m-1)})'$.

Proof of Corollary 3.2 The representation follows directly from the application of Proposition 3.1 to $\mathbf{x}_{kt} \sim \mathcal{N}_n(A_{kt}\mathbf{v}_{kt-1}, A_{kt}\Upsilon_k A_{kt}')$.

A.2 Sequential Clusering

The sequential clustering algorithm is summarized as follows. 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 the previous sections. 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 $\xi_{ij,t}$ is equal to 1 if $j = j_i$ and 0 otherwise and the centroids are updated as follows

$$\mathbf{c}_{jt+1} = \mathbf{c}_{jt} + \lambda_t(\mathbf{m}_{jt+1} - \mathbf{c}_{jt}) \tag{A.37}$$

where

$$\mathbf{m}_{jt+1} = \frac{1}{n_{jt+1}} \sum_{i \in N_{jt+1}} \boldsymbol{\psi}_{it} \tag{A.38}$$

and $\lambda_t \in [0, 1]$. Note that the choice $\lambda_t = n_{jt+1}/(n_{jt}^c + n_{jt+1})$, with $n_{jt}^c = \sum_{s=1}^t n_{js}$,

implies a sequential clustering with forgetting driven by the processing of the blocks of observations. In the application we fix $\lambda = 0.99$.

A.3 Nonlinear sequential filtering

Each particle set $\Phi_t^j = \{\mathbf{w}_t^{\theta\, ij}, \tilde{\gamma}_t^{ij}\}_{i=1}^N$, $j = 1, \ldots, M$, is updated through the following steps.

1. Conditional combination weights. The approximated state predictive density is

$$p_{N,j}(\mathbf{w}_{t+1}^{\theta}|\mathbf{y}_{1:t}, \tilde{\mathbf{y}}_{1:t}^j) = \sum_{i=1}^{N} p(\mathbf{w}_{t+1}^{\theta}|\mathbf{w}_t^{\theta}, \mathbf{y}_{1:t}, \tilde{\mathbf{y}}_{1:t}^j)\tilde{\gamma}_t^{ij}\delta_{\mathbf{w}_t^{\theta\, ij}}(\mathbf{w}_t^{\theta}) \tag{39}$$

2. Conditional prediction. The predictive density allows us to obtain the weight predictive density

$$p_{N,j}(\mathbf{z}_{t+1}|\mathbf{y}_{1:t+1}, \tilde{\mathbf{y}}_{1:t+1}^j) = \sum_{i=1}^{N} \gamma_{t+1}^{ij}\delta_{\mathbf{w}_{t+1}^{\theta\, ij}}(\mathbf{w}_{t+1}^{\theta}) \tag{40}$$

where $\gamma_{t+1}^{ij} \propto \tilde{\gamma}_t^{ij} p(y_{t+1}|\mathbf{w}_{t+1}^{\theta\, ij}, \tilde{\mathbf{y}}_{t+1}^j)$ is a set of normalized weights, and the observable predictive density

$$p_{N,j}(y_{t+1}|\mathbf{y}_{1:t}, \tilde{\mathbf{y}}_{1:t+1}^j) = \sum_{i=1}^{N} \gamma_{t+1}^{ij}\delta_{y_{t+1}^{ij}}(y_{t+1}) \tag{41}$$

where y_{t+1}^{ij} has been simulated from the combination model $p(y_{t+1}|\mathbf{w}_{t+1}^{\theta\, ij},\ \tilde{\mathbf{y}}_{t+1}^j)$ independently for $i = 1, \ldots, N$.

3. Resampling. Since the systematic resampling of the particles introduces extra Monte Carlo variations and reduces the efficiency of the importance sampling algorithm, we do resampling only when the effective sample size (ESS) is below a given threshold. See Casarin and Marin (2009) for ESS calculation. At the $t+1$-th iteration if $\text{ESS}_{t+1}^j < \kappa$, simulate $\Phi_{t+1}^j = \{\mathbf{w}_{t+1}^{\theta\, k_i j}, \tilde{\gamma}_{t+1}^{ij}\}_{i=1}^N$ from $\{\mathbf{w}_{t+1}^{\theta\, ij}, \gamma_{t+1}^{ij}\}_{i=1}^N$ (e.g., multinomial resampling) and set $\tilde{\gamma}_{t+1}^{ij} = 1/N$. We denote with k_i the index of the i-th re-sampled particle in the original set Φ_{t+1}^j. If $\text{ESS}_{t+1}^j \geq \kappa$ set $\Phi_{t+1}^j = \{\mathbf{w}_{t+1}^{\theta\, ij}, \tilde{\gamma}_{t+1}^{ij}\}_{i=1}^N$.

B Online Appendix

B.1 Simulation example

To provide a graphical illustration of our compositional factor model, a simulated example is presented. Let there be only one variable of interest $y_{1t} = y_t$, with values given by the combination of five predictors (i.e. $K = 1$ and $n = 5$)

$$y_t = \sum_{i=1}^{5} \tilde{y}_{it} s_{it} + \varepsilon_t, \quad \varepsilon_t \sim \mathcal{N}(0, 0.2) \tag{B.42}$$

$t = 1, \ldots, T$, where $\tilde{y}_{it} \sim \mathcal{N}(i, 0.1i)$ i.i.d. $i = 1, \ldots, 5$ are the predictive distributions, $(s_{1t}, \ldots, s_{5t})' \sim \mathcal{M}_n(1, (s_{1t}, \ldots, s_{5t}))$, and

$$\begin{pmatrix} w_{1t} \\ w_{2t} \\ w_{3t} \\ w_{4t} \\ w_{5t} \end{pmatrix} = \begin{pmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 0 & 1 & -1 \\ 0 & 1 & -1 \\ 0 & 0 & 0 \end{pmatrix} \odot \begin{pmatrix} z_{1t} \\ z_{2t} \\ z_{3t} \end{pmatrix} \oplus \begin{pmatrix} \varsigma_{1t} \\ \varsigma_{2t} \\ \varsigma_{3t} \\ \varsigma_{4t} \\ \varsigma_{5t} \end{pmatrix} \tag{B.43}$$

with $(\varsigma_{1t}, \varsigma_{2t}, \varsigma_{3t}, \varsigma_{4t})' \sim \mathcal{L}_4(\mathbf{0}_4, 0.1 D_5 D_5')$ i.i.d. and $\varsigma_{5t} = 1 - \varsigma_{1t} - \ldots - \varsigma_{4t}$.

For expository purposes, in order to show graphically the relationship between the components of $\mathbf{w}_t$, which are on the 4-dimension simplex, we assume $\mathbf{w}_t$ is a transform of $\mathbf{z}_t$ with some noise. The dynamics of the latent factors on the simplex of dimension 2 are given by

$$\begin{pmatrix} z_{1t} \\ z_{2t} \\ z_{3t} \end{pmatrix} = \frac{1}{3} \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} \oplus \begin{pmatrix} z_{1t-1} \\ z_{2t-1} \\ z_{3t-1} \end{pmatrix} \oplus \begin{pmatrix} \eta_{1t} \\ \eta_{2t} \\ \eta_{3t} \end{pmatrix} \tag{B.44}$$

with $(\eta_{1t}, \eta_{2t}) \sim \mathcal{L}_2(\mathbf{0}_2, 0.2 D_3 D_3')$ i.i.d. and $\eta_{3t} = 1 - \eta_{1t} - \eta_{2t}$. We generate a trajectory of $T = 500$ points from the latent factor process (blue line in the top-left chart of Fig. B.1) starting at $\mathbf{z}_0 = \boldsymbol{\iota}_3/3$ (black dot). The top-right chart of the same figure shows the scatter plot of w_{kt}, $k = 2, 3, 4$ against the first weight w_{1t}. One can easily see that w_{2t} moves along the same direction of w_{1t}, that is it lies on the 45-degree line, whereas w_{3t} and w_{4t} move together and their relationship with w_{1t} reflects the relationship between z_1 and z_{2t}. The bottom-left chart shows the trajectory of y_t which exhibits a change in mean and variance following the features of the largest combination weight at time t (see bottom-right chart).

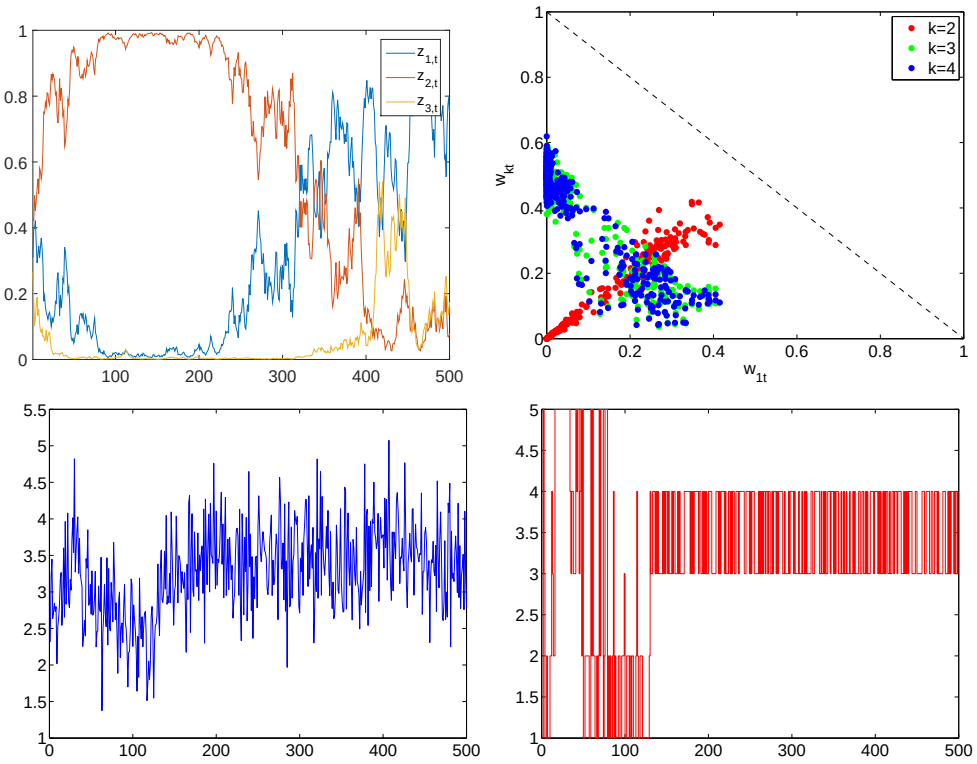

Figure B.1: Simplicial random walk trajectory $\mathbf{z}_t$ (top-left), scatter plot of elements of the latent weight vector $\mathbf{w}_t$ (top-right), observable process y_t (bottom-left) and the largest weight indicator $w_t^* = \max\{w_{kt}, k = 1, \ldots, 5\}$ (bottom-right).

B.2 Sequential approximation of combination weights and predictive densities

B.2.1 Parallel sequential filtering

With regard to the filtering part, we use M parallel conditional SMC filters, where each filter is conditioned on the predictor vector sequence $\tilde{\mathbf{y}}_s$, $s = 1, \ldots, t$. We initialise independently the M particle sets: $\Phi_0^j = \{\mathbf{w}_0^{\theta\, ij}, \tilde{\gamma}_0^{ij}\}_{i=1}^N$, $j = 1, \ldots, M$. Each particle set Φ_0^j contains N i.i.d. random variables $\mathbf{w}_0^{\theta\, ij}$ with random weights $\tilde{\gamma}_0^{ij}$. We initialise the set of predictors, by generating i.i.d. samples $\tilde{\mathbf{y}}_1^j$, $j = 1, \ldots, M$, from $p(\tilde{\mathbf{y}}_1|\mathbf{y}_0)$ where $\mathbf{y}_0$ is an initial set of observations for the variable of interest.

Then, at the iteration $t + 1$ of the combination algorithm, we approximate the

predictive density $p(\tilde{\mathbf{y}}_{t+1}|\mathbf{y}_{1:t})$ as follows

$$p_M(\tilde{\mathbf{y}}_{t+1}|\mathbf{y}_{1:t}) = \frac{1}{M}\sum_{j=1}^{M}\delta_{\tilde{\mathbf{y}}_{t+1}^j}(\tilde{\mathbf{y}}_{t+1})$$

where $\tilde{\mathbf{y}}_{t+1}^j$, $j = 1, \ldots, M$, are i.i.d. samples from the predictive densities and $\delta_x(y)$ denotes the Dirac mass at x.

We assume an independent sequence of particle sets $\Phi_t^j = \{\mathbf{w}_{1:t}^{\theta\, ij}, \tilde{\gamma}_t^{ij}\}_{i=1}^N$, $j = 1, \ldots, M$, is available at time $t+1$ and that each particle set provides the approximation

$$p_{N,j}(\mathbf{w}_t^\theta|\mathbf{y}_{1:t}, \tilde{\mathbf{y}}_{1:t}^j) = \sum_{i=1}^{N}\tilde{\gamma}_t^{ij}\delta_{\mathbf{w}^{\theta\, ij}}(\mathbf{w}_t^\theta) \tag{B.45}$$

of the filtering density, $p(\mathbf{w}_t^\theta|\mathbf{y}_{1:t}, \tilde{\mathbf{y}}_{1:t}^j)$, conditional on the j-th predictor realisation, $\tilde{\mathbf{y}}_{1:t}^j$. Then M independent conditional SMC algorithms are used to find a new sequence of M particle sets, which include the information available from the new observation and the new predictors. Each SMC algorithm iterates, for $j = 1, \ldots, M$, the steps given in Appendix A.

After collecting the results from the different particle sets, it is possible to obtain the following empirical predictive density

$$p_{M,N}(\mathbf{y}_{t+1}|\mathbf{y}_{1:t}) = \frac{1}{MN}\sum_{j=1}^{M}\sum_{i=1}^{N}\delta_{\mathbf{y}_{t+1}^{ij}}(\mathbf{y}_{t+1}) \tag{B.46}$$

For horizons $h > 1$, we apply a direct forecasting approach (see Massimiliano et al., 2006) and compute predictive densities $p_{M,N}(\mathbf{y}_{t+h}|\mathbf{y}_{1:t})$ following the steps previously described.

B.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.

2.b 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. A.37.

The k-means algorithm is parallel in point 2) and 3) and this can be used in the GPU context as we do in this paper.

B.3 Forecast evaluation

To shed light on the predictive ability of our methodology, we consider several evaluation statistics for point and density forecasts previously proposed in the literature. Suppose we have $i = 1, \ldots, n$ different approaches to predict the variable y. 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.

The complete predictive densities are evaluated using the Kullback Leibler Information Criterion (KLIC)-based measure, utilising the expected difference in the Logarithmic Scores of the candidate forecast densities; see, for example, Mitchell and Hall (2005), Hall and Mitchell (2007), Amisano and Giacomini (2007), Kascha and Ravazzolo (2010), Billio et al. (2013), and Aastveit et al. (2014).

The KLIC is the distance between the true density $p(y_{t+h}|\mathbf{y}_{1:t})$ of a random variable y_{t+h} and some candidate density $p_i(y_{t+h}|\mathbf{y}_{1:t})$ obtained from the approach i and chooses the model that on average gives the higher probability to events that actually occurred. An estimate of it can be obtained from the average of the sample information, $y_{\underline{t}+1}, \ldots, y_{\bar{t}+1}$, on $p(y_{t+h}|\mathbf{y}_{1:t})$ and $p_i(y_{t+h}|\mathbf{y}_{1:t})$:

$$\overline{KLIC}_{i,h} = \frac{1}{t^*} \sum_{t=\underline{t}}^{\bar{t}} [\ln p(y_{t+h}|\mathbf{y}_{1:t}) - \ln p_i(y_{t+h}|\mathbf{y}_{1:t})] \tag{B.47}$$

Although we do not know the true density, we can still compare different densities, $p_i(y_{t+h}|\mathbf{y}_{1:t})$, $i = 1, \ldots, n$. For the comparison of two competing models, it is sufficient

to consider the Logarithmic Score (LS), which corresponds to the latter term in the above sum,

$$LS_{i,h} = -\frac{1}{t^*}\sum_{t=\underline{t}}^{\bar{t}} \ln p_i(y_{t+h}|\mathbf{y}_{1:t}) \tag{B.48}$$

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

Secondly, we also evaluate density forecasts based on the continuous rank probability score (CRPS); see, for example, Gneiting and Raftery (2007), Gneiting and Roopesh (2013), Groen et al. (2013) and Ravazzolo and Vahey (2014). 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:

$$\begin{aligned} \text{CRPS}_{i,t+h} &= \int_{-\infty}^{+\infty} \left(F(z) - \mathbb{I}_{[y_{t+h},+\infty)}(z)\right)^2 \mathrm{d}z \\ &= \mathbb{E}_t|\tilde{y}_{i,t+h} - y_{t+h}| - \frac{1}{2}\mathbb{E}_t|\tilde{y}^*_{i,t+h} - \tilde{y}'_{i,t+h}| \end{aligned} \tag{B.49}$$

where F is the CDF from the predictive density $p_i(y_{t+h}|\mathbf{y}_{1:t})$ 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 $p_i(y_{t+h}|\mathbf{y}_{1:t})$. We report the sample average CRPS:

$$\text{CRPS}_{i,h} = -\frac{1}{t^*}\sum_{t=\underline{t}}^{\bar{t}} \text{CRPS}_{i,t+h} \tag{B.50}$$

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.

Finally, following Clark and Ravazzolo (2015), we apply the Diebold and Mariano (1995) 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 Andrews and Monahan (1992). Monte Carlo evidence in Clark and McCracken (2015) and Clark and McCracken (2011) indicates that, with nested models, the Diebold-Mariano test compared against

	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 B.1: Average cross-section statistics for the 3712 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.

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.

B.4 Additional details on the financial application

Table B.1 reports the cross-section average statistics, together with statistics for the S&P500. Some series have much lower average returns than the index and 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.

Figure B.2 presents the mean values of the predicted features $\boldsymbol{\psi}_{it}$ which belong to the $j-$th cluster at each of the 784 vintages, labeled as $\mathbf{m}_{jt+1}$. The clusters for the Normal GARCH(1,1) models differ substantially in terms of predicted variance with cluster 1 having a rather low constant variance value over the entire period while cluster 2 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 has a relatively

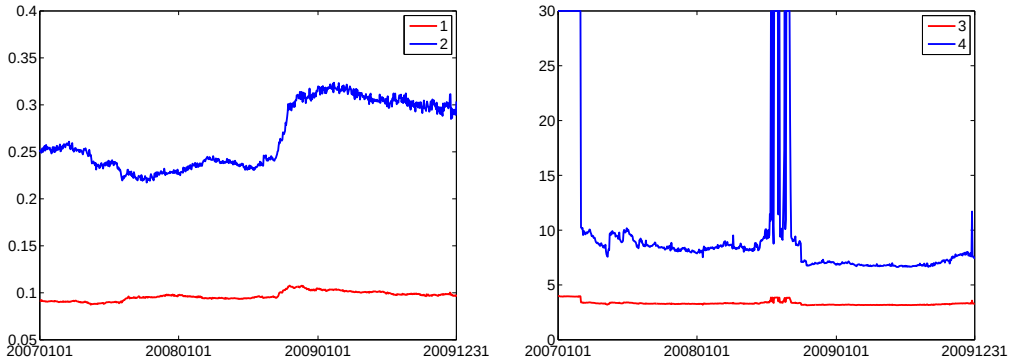

Figure B.2: The figures present the average variance of the predictions from the two clusters for the Normal GARCH(1,1) models based on low (cluster 1) and high (cluster 2) volatility in the left panel; and the average degree of freedom of the predictions from the two clusters for the t-GARCH(1,1) models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

constant thick tail over the entire period while cluster 4 has an average value of 10 for the degrees of freedom and in the crisis period the density collapses to a normal density with degrees of freedom higher than 30. In summary, The Lehman Brother effect is visible in the figure, with an increase of volatility in the normal cluster 2 and, interesting, an increase of the degrees of freedom in the t-cluster 4.

B.5 Additional details and on the macroeconomic application

This section reports a detailed description of the cluster composition, in terms of predictors, for the 5 and 7 clusters analysis of the series given in Fig. B.3, and additional figures and tables related to their analysis and forecasting results.

The left and right columns in Fig. B.4) 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. B.4, 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. B.4, last row). The differences in terms of persistence, in the different groups, is also evident from the heat maps given in Fig. B.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

Table B.2: 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						

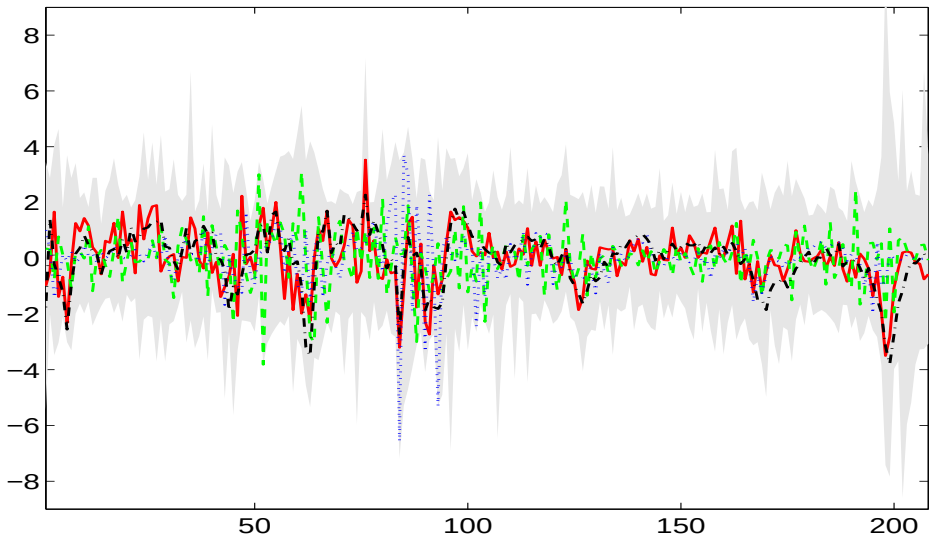

Figure B.3: Gray area: the set of series (standardised for a better graphical representation), at the monthly frequency, of the Stock and Watson dataset. Solid line: growth rate of real GDP (seasonally adjusted) for the US. Dashed line: inflation measured as the change in the GDP deflator index (seasonally adjusted). Dotted line: yields on US government 90-day T-Bills (secondary market). Dashed-dotted: total employment growth rate for private industries (seasonally adjusted).

2nd and 4th cluster (of 5) are more persistent then the series in the clusters 1, 3 and 5 (see also Fig. B.4, 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 B.5. Looking at the composition of the predictor groups (see also Tables B.3-B.4), we find 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 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).

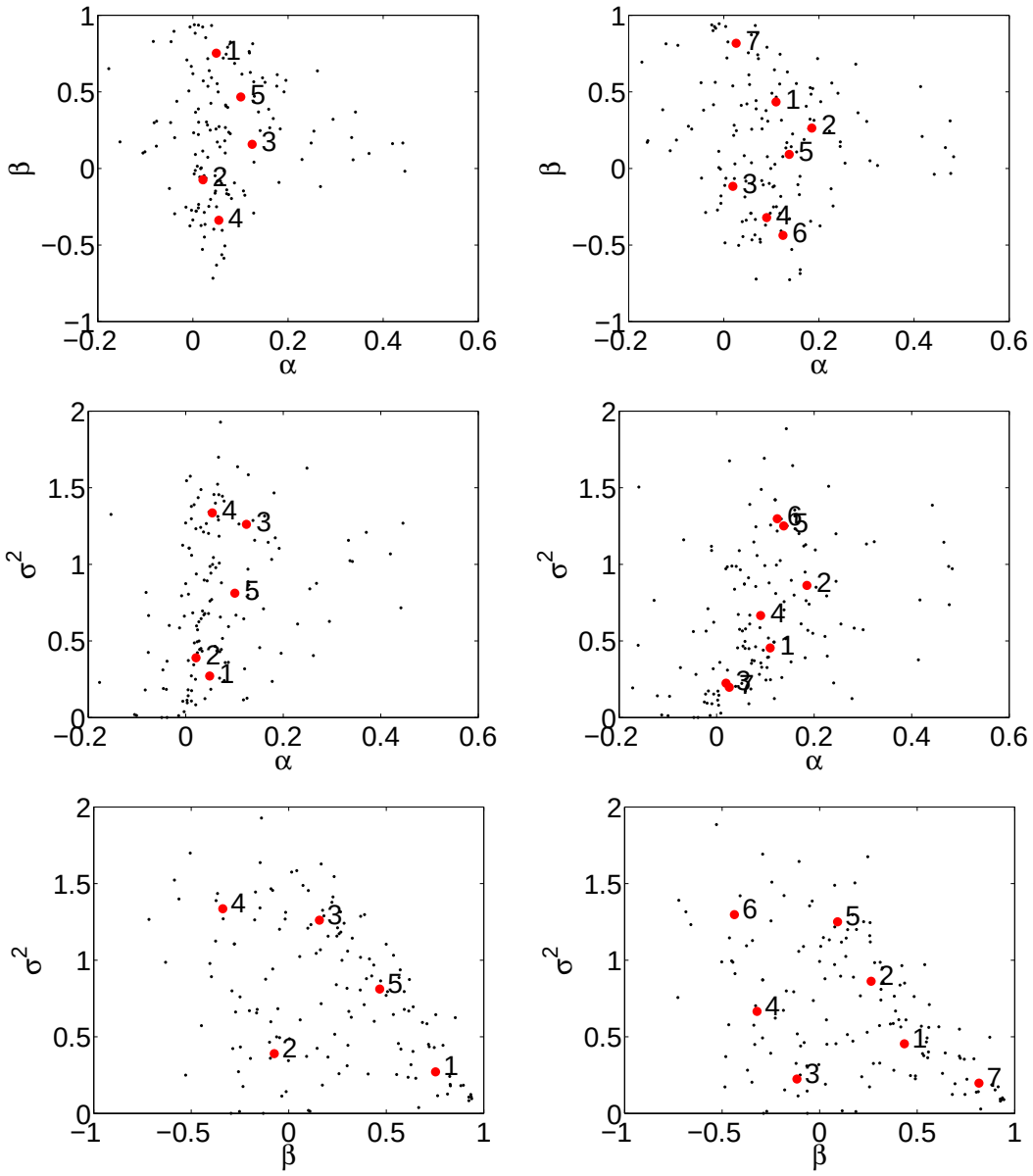

Figure B.4: Pairwise scatter plots of the series features: α_i and β_i (first row), α_i and σ_i^2 (second row) and β_i and σ_i^2 (last row). In each plot the red dots represent the cluster means. We assume alternatively 5 (left) and 7 (right) clusters.

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.

Table B.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 B.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						

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 B.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. B.4 and the remaining three columns the cluster mean along the different directions of the parameter space.

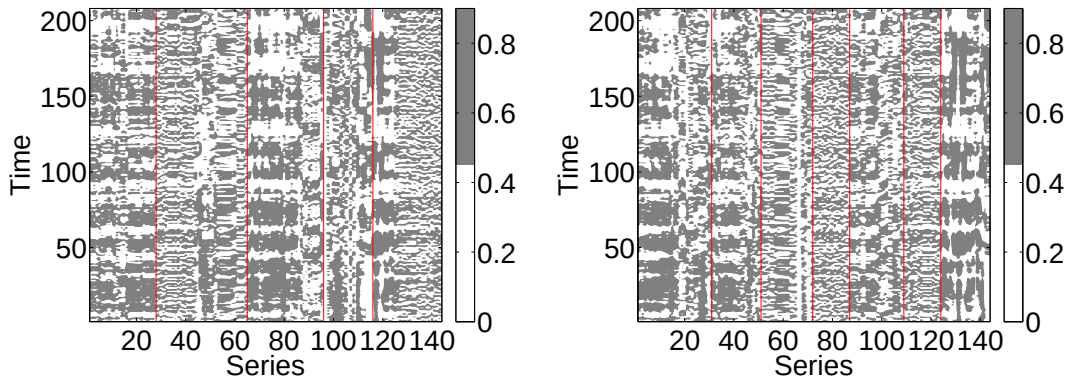

Figure B.5: Normal cumulative density function for the standardised series. The series are ordered by cluster label. We assume alternatively 5 (left) and 7 (right) clusters.

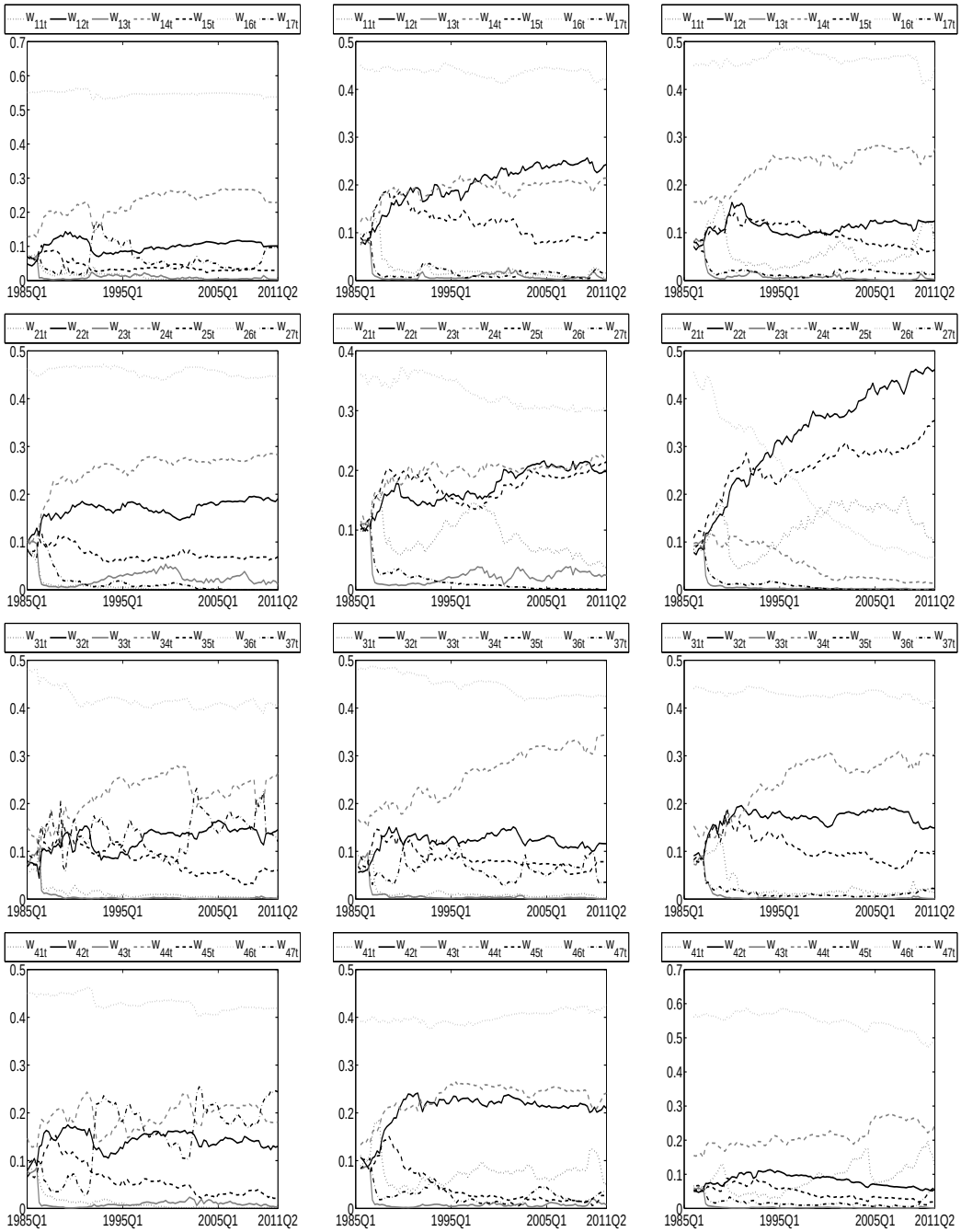

Figure B.6: In each plot the mean logistic-normal weights (different lines) for the univariate combination model are given. Rows: plot for the four series of interest (real GDP growth rate, GDP deflator, 3-month Treasury Bills, employment). Columns: forecast horizons (1, 3 and 5 quarters).

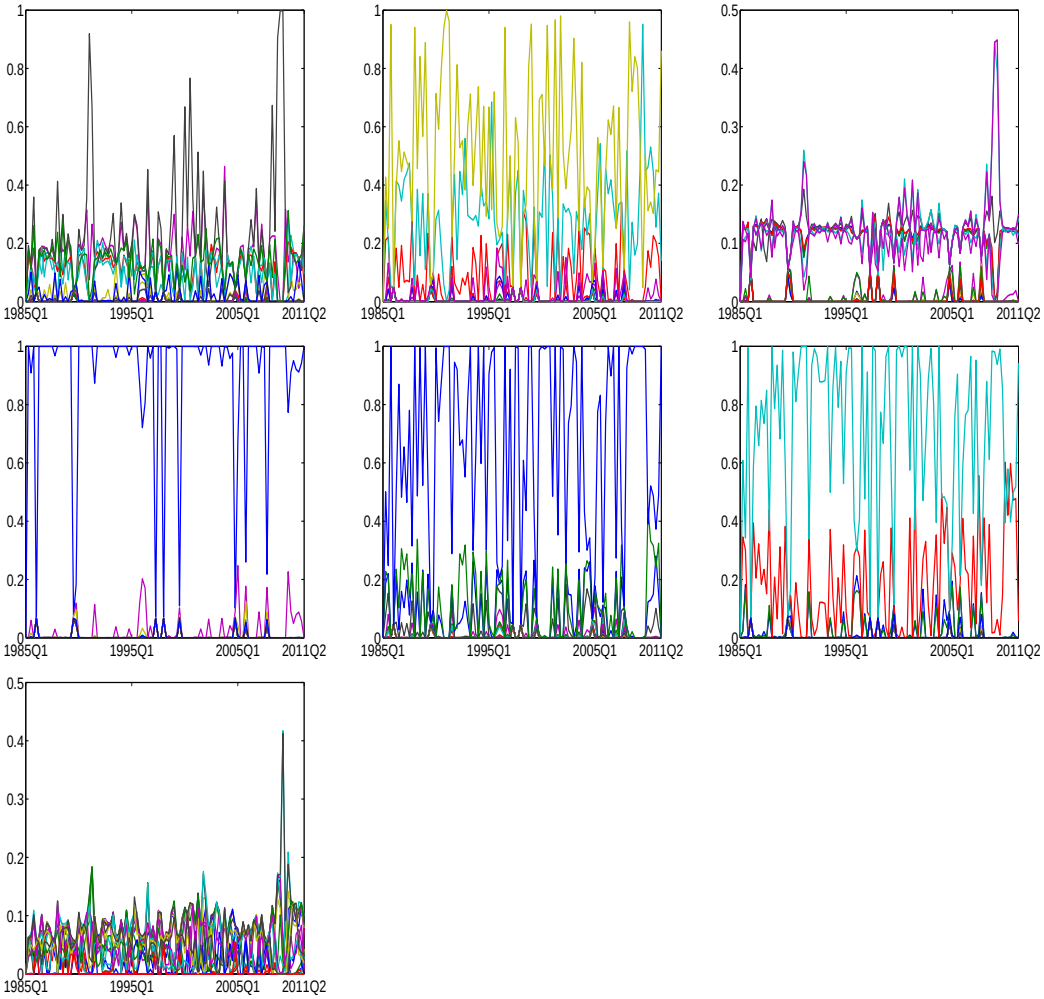

Figure B.7: The plots show the model weights ($b_{k,ij}$) in each cluster ($i = j$) when forecasting GDP growth ($k = 1$) at the 1-step ahead horizon. The first row refers to clusters 1, 2, and 3; the second row to clusters 4, 5, and 6; the last row to cluster 7.

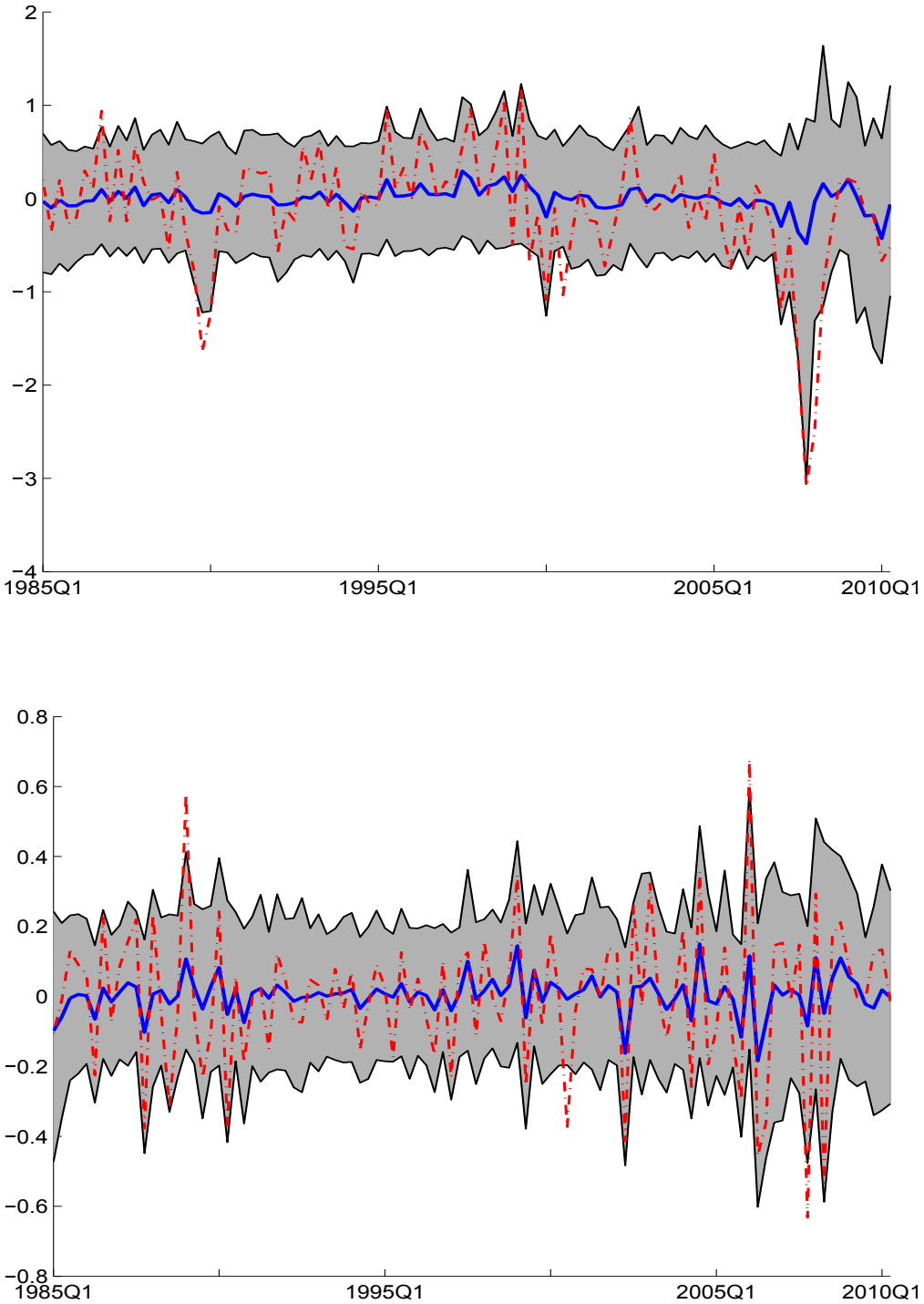

Figure B.8: 5-step ahead fan charts for demeaned GDP (top panel) and demeaned GDP deflator (bottom panel). Estimated mean (solid blue line) and 5% and 95% quantiles (gray area) of the marginal prediction density. (Demeaned) realizations in red dashed line

B.6 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, Aldrich (2014) for a review, Geweke and Durham (2012) and Lee et al. (2010) for applications to Bayesian inference and Aldrich et al. (2011), Morozov and Mathur (2012) and Dziubinski and Grassi (2013) 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, see Section 5, and the density combination with univariate combination based on 7 clusters with recursive log score weights within clusters, UDCLS7[5], see Section 5.2, for an increasing number of particles N. In both sets of experiments we calculated, in seconds, the overall average execution time reported in Table B.6.

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.

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

	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 B.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

Aastveit, K. A., Gerdrup, K. R., Jore, A. S., and Thorsrud, L. A. (2014). Nowcasting GDP in real-time:A density combination approach. *Journal of Business Economics & Statistics*, 32:48–68.

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.

Amisano, G. and Giacomini, R. (2007). Comparing density forecasts via weighted likelihood ratio tests. *Journal of Business & Economic Statistics*, 25:177–190.

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.

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*, forthcoming.

Casarin, R. and Marin, J. M. (2009). Online data processing: Comparison of Bayesian regularized particle filters. *Electronic Journal of Statistics*, 3:239–258.

Clark, T. E. and McCracken, M. W. (2011). Testing for unconditional predictive ability. In Clements, M. and Hendry, D., editors, *Oxford Handbook of Economic Forecasting*. Oxford University Press, Oxford.

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 Roopesh, 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.

Hall, S. G. and Mitchell, J. (2007). Combining density forecasts. *International Journal of Forecasting*, 23:1–13.

Kascha, C. and Ravazzolo, F. (2010). Combining inflation density forecasts. *Journal of Forecasting*, 29:231–250.

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.

Massimiliano, M., Stock, J. H., and Watson, M. W. (2006). A comparison of direct and iterated multistep AR methods for forecasting macroeconomic time series. *Journal of Econometrics*, 135:499–526.

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.

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.

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