Boosting multiplicative model combination

Boosting multiplicative model combination

PAOLO VIDONI

Department of Economics and Statistics, University of Udine

RUNNING HEADLINE: Boosting model combination

Abstract

In this paper we define a new boosting-type algorithm for multiplicative model combination using as loss function the Hyvärinen scoring rule. In particular, we focus on density estimation problems and the aim is to define a suitable estimator, using a multiplicative combination of elementary density functions, which correspond to simplified or partially specified probability models for the interest random phenomenon. The boosting algorithm provides a simple sequential procedure for updating the weights of the component density functions, until an optimality criterion is satisfied. An extension of this procedure can be useful for composite likelihood inference, in order to specify the weights of the component likelihood objects and, simultaneously, implement parameter estimation. Finally, three applications are presented. The first one regards prediction and inference for autoregressive models, the second one the use of model pools for prediction in a time series framework and the third one the estimation of the covariance and the precision matrices of a multivariate Gaussian distribution. Empirical results on real-world financial data are presented in challenging contexts where we have to deal with a large dataset or with sparse matrices and a large number of unknown parameters.

Keywords: boosting, composite likelihood, density estimation, Hyvärinen's divergence, multiplicative mixture model, precision matrix

1 Introduction

This paper concerns a new boosting-type procedure for estimating density functions. Boosting algorithms (see, for example, Schapire & Freund, 2012) were originally introduced in the classification framework with the aim of combining simple, weak classifiers in order to obtain a final weighted classifier having a strong discriminating ability. These techniques are also applied in a more general regression framework, for model calibration and prediction purposes. However, the basic idea is the same: to combine weak learners (for example, classification rules or predictors) or, more generally, basic models in order to define a pooled model with an improved predictive performance. This final model is usually defined as a linear mixture model, where the components and the associated weights are sequentially specified using the available data, according to a specific optimality criterion.

The focus here is on density estimation and, in particular, the aim is to estimate the unknown density $f(z)$ of the random vector Z, using a suitable combination of basic density functions $p_j(z)$, $j = 1, \ldots, J$. The problem of combining density functions, also termed (prediction) pooling, is considered quite often in the econometric and in the quantitative finance literature (see, for example, Kascha & Ravazzolo, 2010, Geweke & Amisano, 2011, and Casarin *et al.*, 2016, for a more general combination approach in a Bayesian inferential framework). Moreover, the component functions might correspond to partially specified, and often misspecified, probability models for Z, which focus on particular features of the random phenomenon or represent computationally tractable alternatives to the true density. In particular, they could also specify suitable marginal or conditional distributions associated to the components of vector Z. This happens, for example, in composite likelihood inferential methods, where the full likelihood is substituted by a combination of a number of low-dimensional likelihood objects (see for example, Lindsay, 1988, and Varin *et al.*, 2011).

The original contribution of the paper is twofold. First, we consider a multiplicative model combination in order to define an estimate for $f(z)$ and we employ, as optimality criterion for the inferential procedure, a divergence introduced by Hyvärinen (2005), instead of the usual Kullback-Liebler divergence. One reason behind this choice is that the calculation of the normalizing constant, usually required for defining a multiplicative mixture model, is not necessary and this reduces the computational burden of the numerical procedure. Second, we define a new boosting-type algorithm which gives a sequential update for the weights of the component density functions, until a suitable stopping criterion is satisfied. Although the estimation of the model corresponds to a simple optimization of quadratic form, this procedure turns out to be useful whenever the mixture model has a large number of components. That is, a regularization procedure is implicitly introduced with the aim of considering only the most informative components terms. Furthermore, this approach can be readily extended to the case where the component model parameters are unknown. Some potential applications of

these results concern univariate and multivariate density estimation, the combination of density forecasts, the estimation of large covariance and precision matrices and the specification of the weights in composite likelihood inferential methods.

The paper is organized as follows. Section 2 introduces the problem, focusing on the notion of multiplicative mixture models and on the definition and the properties of the Hyvärinen's divergence. The new boosting algorithm for multiplicative model combination is presented in Section 3, considering the situation in which the model parameters of the component density functions are known. In Section 4, the optimization procedure is extended to the case of unknown model parameters. Finally, Section 5 shows applications to prediction and inference for simple autoregressive models with additive observation noise, to the use of model pools for prediction in a time series framework and to the estimation of the covariance and the precision matrices of a multivariate Gaussian distribution. In this context a first real-world data analysis regards the definition of an optimal model pool for predicting daily S&P500 returns and a second one concerns the analysis of monthly returns of US industry portfolios, where we find that the new boosting-type procedure provides promising results for handling large-dimension, sparse precision matrices.

2 Multiplicative model combination

2.1 Definition and properties

Let us consider a continuous K-dimensional random vector $Z = (Z_1, \ldots, Z_K)^T$, with $K \geq 1$, having an unknown density function $f(z)$, $z = (z_1, \ldots, z_K)^T \in \mathbb{R}^K$, and a finite-dimensional set $\mathcal{P} = \{p_j(z),\, j = 1, \ldots, J\}$ containing $J \geq 1$ plausible density functions for Z. Functions $p_j(z)$, $j = 1, \ldots, J$, are supposed to be known; in the following we consider also the case where the densities are specified up to an unknown finite dimensional parameter. Let us assume that an n-dimensional sample $z^{(1)}, \ldots, z^{(n)}$, with $n \geq 1$, is available from Z under $f(z)$. The aim is to use the information given by the observed

sample in order to define a combination of the models in $\mathcal{P}$ as a useful surrogate for the true density $f(z)$. In particular, the combined model is expected to provide accurate prediction statements for Z and it will be specified according to a suitable optimality criterion.

In this paper the focus is on multiplicative combinations of densities, called also logarithmic pools, defined as

$$f_p(z; w) = c(w)^{-1} \prod_{j=1}^{J} p_j(z)^{w_j}, \quad z \in \mathbb{R}^K, \tag{1}$$

with $w = (w_1, \ldots, w_J)^T$ a J-dimensional vector of non-negative weights and $c(w) = \int_{\mathbb{R}^K} \prod_{j=1}^{J} p_j(z)^{w_j} dz$ the normalizing constant, supposed to be finite. We implicitly assume that the product of densities in (1) is integrable, so that it gives a valid density function after normalization. Under this respect, a sufficient condition is that functions $p_j(z)$, $j = 1, \ldots, J$, are bounded probability density functions. Note that density (1) can be viewed as a multiparameter exponential family obtained as the tilting of the component densities $p_j(z)$, $j = 1, \ldots, J$, and its logarithmic transformation can be expressed as

$$\log f_p(z; w) = w^T P_0(z) - \log c(w),$$

with $P_0(z) = (\log p_1(z), \ldots, \log p_J(z))^T$.

A well-known alternative procedure of density aggregation is the linear pool, which defines a (linear) mixture density

$$\tilde{f}_p(z; \tilde{w}) = \sum_{j=1}^{J} \tilde{w}_j p_j(z), \quad z \in \mathbb{R}^K,$$

with $\tilde{w} = (\tilde{w}_1, \ldots, \tilde{w}_J)^T$ a vector of non-negative weights such that $\sum_{=1}^{J} \tilde{w}_j = 1$. Compared to the linear pool, the logarithmic one gives densities which are typically uni-modal and less dispersed. Moreover, if the weights w are normalized to sum up to one, the logarithmic combination method is invariant under rescaling and it verifies the property of external Bayesianity (Genest, 1984, and Allard *et al.*, 2012). This last property characterizes the logarithmic pool and it essentially means that, given

a new observation, the operation of updating the component distributions with a common likelihood commutes with the pooling operator.

Although the interest here is on the whole probability distribution of Z, and the aim is to define a suitable multiplicative density combination as an estimator for the unknown density function $f(z)$, it can be interesting to discuss briefly also the statistical properties of the combined density, with particular concern to the associated moments. With regard to the linear pool, Pauwels *et al.* (2018) emphasize that the moments of the combination may have an unexpected behaviour, often not predictable considering the moments of the component densities. A similar situation occurs for the logarithmic pool since an upper bound for its k-th moment, with $k \in \mathbb{N}^+$, is proportional to the k-th moment of the linear pool. Using Jensen's inequality, it is quite easy to prove that

$$E_{f_p}(Z^k) = \int_{\mathbb{R}} z^k f_p(z; w) dz \leq c(w)^{-1} \int_{\mathbb{R}} z^k \tilde{f}_p(z; w) dz = c(w)^{-1} E_{\tilde{f}_p}(Z^k),$$

provided that the weights sum up to one and the integrals are finite. Hereafter, $E_g(\cdot)$ corresponds to the expectation with respect to the density g.

The following numerical examples show that, also for logarithmic pools, the indices of skewness and kurtosis may vary considerably even when the component models present the same higher order moments. In the first example, we consider the multiplicative combination of two skew-normal distributions with the same scale parameter 1 and shape parameter 5 (so that the common skewness is 0.85); the location parameters are different: 0.1 and 1 in the first case and -1 and 1 in the second case. The behaviour of the skewness as a function of w_1 is described in Figure 1. In the second example, we combine two generalized Student t distributions with the same scale parameter 1 and degrees of freedom 5 (so that the common kurtosis is 9); the location parameters are different: -1 and 1 in the first case and -5 and 1 in the second case. The behaviour of the kurtosis as a function of w_1 is also described in Figure 1.

Figure 1 here

These empirical findings, although related to specific situations, highlight that the logarithmic pool defines a flexible parametric family of density functions. Even though this fact could be an advantage from the inferential point of view, for some application, such as in finance, the potential reduction of skewness and kurtosis, produced by the model combination procedure, could be problematic. In this framework, Pauwels *et al.* (2018) propose a suitable weights calibration procedure in order to face this problem.

2.2 Weights selection

The aim is to find the multiplicative density combination (1), or equivalently the vector of weights w, which minimizes a suitable notion of distance between $f(z)$ and $f_p(z; w)$. In order to evaluate the distance between these two densities, it is quite common to consider the Kullback-Liebler divergence, and the associated log score, and then to adopt a likelihood-based inferential procedure for estimating the unknown weights w. However, this approach requires the specification of the normalising constant $c(w)$, which could not have an explicit form and also the use of numerical computation methods could be too computational demanding.

For a continuous random vector Z taking values in $\mathbb{R}^K$, an alternative divergence, based on the gradient of the log densities, is introduced by Hyvärinen (2005) as

$$d_H(f, f_p; w) = \int_{\mathbb{R}^K} ||\nabla \log f(z) - \nabla \log f_p(z; w)||^2 f(z)\, dz,$$

where ∇ denotes the gradient operator, that is $\nabla\, g(x) = (\partial g(x)/\partial x_1, \cdots, \partial g(x)/\partial x_K)^T$, with $x = (x_1, \ldots, x_K)^T$ and g a differentiable, real-valued function defined on $\mathbb{R}^K$, and $||x|| = (x_1^2 + \cdots + x_K^2)^{1/2}$ is the Euclidean norm. This divergence is non-negative, it vanishes only when $f = f_p$ and, since it involves the gradient of the log-densities, it can be computed without the knowledge of the normalizing

constants of $f(z)$ and $f_p(z;w)$. Since it matches the scores, with respect to the vector z, it is also referred to as the score matching loss. Under suitable regularity assumptions, Hyvärinen (2005) shows that minimizing $d_H(f, f_p; w)$ is equivalent to minimizing the expected Hyvärinen score

$$\begin{aligned} S_H(f_p; w) &= \int_{\mathbb{R}^K} \left\{ 2 \triangle \log f_p(z;w) + ||\nabla \log f_p(z;w)||^2 \right\} f(z)\, dz \\ &= E_f \left[2 \triangle \log f_p(Z;w) + ||\nabla \log f_p(Z;w)||^2 \right], \end{aligned} \tag{2}$$

where $\triangle$ indicates the Laplacian operator, so that $\triangle g(x) = \sum_{k=1}^{K} \partial^2 g(x)/\partial x_k^2$. This result holds for continuous random vectors with support $\mathbb{R}^K$ and it can be proved using essentially the integration by parts rule, provided that suitable conditions, assuring that the boundary terms vanish and that the densities are twice continuously differentiable, are satisfied. Extensions to the case of continuous, non-negative random vectors and to some particular discrete random vectors may be found in Hyvärinen (2007).

The main argument of the integral (2), namely $S_H(z, g) = 2 \triangle \log g(z) + ||\nabla \log g(z)||^2$ with $g = f_p$, is called the Hyvärinen score of f_p, when the observed value of Z is z. As emphasized by Parry *et al.* (2012), the Hyvärinen score $S_H(z, g)$, with $g(z)$ a suitable candidate density for Z, satisfies some interesting properties. In particular, it is strictly proper since the unique minimum, with respect to g, of the associate expected score $S_H(g) = E_f[S_H(Z, g)]$ defined by (2) with g substituted for f_p, is achieved when g equals the true density f. In addition, g enters in the expression for $S_H(z, g)$ only through its first and second derivatives and then, even if it is not local in a strict sense, it is termed local of order 2. Furthermore, the Hyvärinen scoring rule is homogeneous in the density g since, if the aim is to minimize the expected score, the objective function is formally unchanged if g is multiplied by a positive constant. This last property turns out to be extremely relevant with respect to the problem considered in this paper, since the loss function which measures the quality of f_p for estimating f, does not require the calculation of the normalizing constant $c(w)$, which may be hard to obtain.

In order to get an estimate for the unknown weights parameter w, we consider the sample version of the objective function (2), obtained by replacing the integration with the sample average based on the observed sample $z^{(1)}, \ldots, z^{(n)}$ and given by

$$\widehat{S}_H(f_p; w) = \frac{1}{n} \sum_{i=1}^{n} \left[2 \triangle \log f_p(z^{(i)}; w) + ||\nabla \log f_p(z^{(i)}; w)||^2 \right]. \tag{3}$$

Since

$$\nabla \log f_p(z; w) = \nabla w^T P_0(z) = \begin{pmatrix} w^T P_1(z) \\ \vdots \\ w^T P_K(z) \end{pmatrix} = w^T \mathbf{P}(z)$$

with $P_k(z) = (\partial \log p_1(z)/\partial z_k, \ldots, \partial \log p_J(z)/\partial z_k)^T$, $k = 1, \ldots, K$, and $\mathbf{P}(z) = (P_1(z), \ldots, P_K(z))$ a matrix with dimension $J \times K$, and

$$\triangle \log f_p(z; w) = \sum_{k=1}^{K} w^T P_{kk}(z) = w^T \sum_{k=1}^{K} P_{kk}(z),$$

with $P_{kk}(z) = (\partial^2 \log p_1(z)/\partial z_k^2, \ldots, \partial^2 \log p_J(z)/\partial z_k^2)^T$, $k = 1, \ldots, K$, the objective function (3) can be rewritten as

$$\widehat{S}_H(f_p; w) = w^T \left\{ \frac{2}{n} \sum_{i=1}^{n} \sum_{k=1}^{K} P_{kk}(z^{(i)}) \right\} + w^T \left\{ \frac{1}{n} \sum_{i=1}^{n} \mathbf{P}(z^{(i)}) \mathbf{P}(z^{(i)})^T \right\} w. \tag{4}$$

Moreover, it is immediate to see that in the univariate case, namely with $K = 1$, we have that

$$\begin{aligned} \widehat{S}_H(f_p; w) &= \frac{1}{n} \sum_{i=1}^{n} \left[2 \frac{f_p''(z^{(i)}; w)}{f_p(z^{(i)}; w)} - \left\{ \frac{f_p'(z^{(i)}; w)}{f_p(z^{(i)}; w)} \right\}^2 \right] \\ &= \frac{2}{n} \sum_{i=1}^{n} \sum_{j=1}^{J} w_j \left[\frac{p_j''(z^{(i)})}{p_j(z^{(i)})} - \left\{ \frac{p_j'(z^{(i)})}{p_j(z^{(i)})} \right\}^2 \right] + \frac{1}{n} \sum_{i=1}^{n} \left\{ \sum_{j=1}^{J} w_j \frac{p_j'(z^{(i)})}{p_j(z^{(i)})} \right\}^2, \end{aligned} \tag{5}$$

where $g'(x)$ and $g''(x)$ indicate, respectively, the first and the second derivatives of a real-valued function $g(x)$ defined on $\mathbb{R}$.

An estimate for w, based on the the Hyvärinen score, is thus obtained by solving the following constraint optimization problem

$$\widehat{w} = \min_{w \in \mathbb{R}_+^J} \widehat{S}_H(f_p; w), \tag{6}$$

with $\mathbb{R}_+^J = \{w \in \mathbb{R} : w_j \geq 0, j = 1, \ldots, J\}$. To this end we take advantage of the following explicit expressions for the gradient vector, with respect to the interest parameter w,

$$\nabla \widehat{S}_H(f_p; w) = \frac{2}{n} \sum_{i=1}^{n} \sum_{k=1}^{K} P_{kk}(z^{(i)}) + 2 \left\{ \frac{1}{n} \sum_{i=1}^{n} \mathbf{P}(z^{(i)}) \mathbf{P}(z^{(i)})^T \right\} w, \tag{7}$$

and for the associated Hessian matrix

$$\nabla^2 \widehat{S}_H(f_p; w) = 2 \left\{ \frac{1}{n} \sum_{i=1}^{n} \mathbf{P}(z^{(i)}) \mathbf{P}(z^{(i)})^T \right\}. \tag{8}$$

Note that the objective function as described in equation (4) is a simple quadratic function of w and then the optimization problem (6) corresponds to a quadratic form, subject to non-negativity constraints (see, for example, Boyd & Vandenberghe, 2004), for which many solvers are available. However, in the following section, this problem will be faced by using a new boosting-type algorithm which gives a simple sequential updating procedure for specifying the weights of the component density functions, until a suitable stopping criterion is satisfied.

For a review on the theoretical properties of the estimators based on proper scoring rules, and in particular on the Hyvärinen score, we may consider Dawid *et al.* (2016) and Yu *et al.* (2016). In this context a topic that receives particular attention is whether the true model belongs or not to the set of model combinations, namely if there exists a weights vector $w_0 \in \mathbb{R}_+^J$ such that $f(z) \equiv f_p(z; w_0)$. In this case, since the scoring rule is strictly proper, it is expected that

$$w_0 = \min_{w \in \mathbb{R}_+^J} S_H(f_p; w)$$

and, if $\widehat{S}_H(f_p; w)$ converges in probability to $S_H(f_p; w)$, as $n \to +\infty$, and some regularity assumptions are satisfied, then we may prove that $\widehat{w}$ is a consistent estimator for w_0. Note that, as a special case, we may consider the situation where one of the component models in $\mathcal{P}$ is the true one; that is, $f = p_h$, with $h \in \{1, \ldots, J\}$, and then, as proved in the Appendix A.1, $w_0 = (w_{01}, \ldots, w_{0J})$, with $w_{0j} = 1$ for $j = h$ and $w_{0j} = 0$ for $j \neq h$. More general results, similar to those presented in Geweke & Amisano (2011), can also be proved in this context.

With regard to the challenging situation where the true model does not belong to the set of model combinations, since the model set is misspecified or incomplete, using a common terminology in econometrics, the results concerning parametric inference under misspecification could be applied; that is, we seek appropriate conditions ensuring that, as n increases, the selected model combination attains the best approximation for the true model in the model set. A new alternative approach allows that the component models in $\mathcal{P}$ vary over time, for example considering time-varying weights, in order to improve the flexibility of the combined model in approximating the true one (see, for example, Aastveit *et al.*, 2018, and Billio *et al.*, 2013). The application of this approach to misspecified logarithmic model pools, using the Hyvärinen score, could be a matter of future research.

3 A boosting-type algorithm for multiplicative model combination

Boosting algorithms share the goal of pooling weak or base learning algorithms, classification rules, predictors, or, more generally, base models in order to incrementally define a combined model with an improved predictive performance. This improved model is usually defined as a linear combination of the elementary models, where the components and the associated weights are sequentially specified using the available data, according to a specific optimality criterion. Thus, boosting algorithms, such as AdaBoost, may be alternatively viewed as procedures for optimizing a suitable objective function, and then they can be analysed as well in the framework of optimization procedures (see, for example, Friedman, 2001, and Schapire & Freund, 2012, Chapter 7). Boosting procedures for unsupervised learning problem of density estimation have been considered in Rosset & Segal (2003) and Welling *et al.* (2003), using a likelihood-based loss criterion.

In this section we shall define a new boosting algorithm for multiplicative model combination using as loss function the Hyvärinen scoring rule. More precisely, the goal is to solve an optimization problem of the form (6) and to this end we use the Gauss-Southwell procedure, which is a simple variant of

the coordinate descent methods (see, for example, Luenberger & Ye, 2008, Section 8.9). We focus on these methods since they specify simple, easy to implement optimization algorithms, though their convergence properties are usually poorer than those of steepest descent procedures. The conditions assuring the identification of a global minimum $\widehat{w}$ for the problem (6) and the convergence to the true value, as n increases, are discussed in the Appendix A.2.

Let us consider a real-valued function $\widehat{S}$, defined over a convex set $\Omega \subseteq \mathbb{R}^J$, $J > 1$, which is, in particular, a convex function, twice continuously differentiable. We want to solve the minimization problem

$$\min_{w \in \Omega} \widehat{S}(w),$$

that is to find the global minimum $\widehat{w} \in \Omega$ by means of an iterative procedure where, chosen an initial value $\widehat{w}^{(0)} \in \Omega$, we repeat the updating step

$$\widehat{w}^{(r)} = \widehat{w}^{(r-1)} + \alpha^{(r)}\, d^{(r)}, \quad r = 1, 2, \ldots,$$

until a stopping criterion is satisfied. The vector $d^{(r)} \in \mathbb{R}^J$ indicates the search direction and, for coordinate descent methods, it corresponds to a vector e_h with a one in position $h \in \{1, \ldots, J\}$ and zero in all other positions, while $\alpha^{(r)}$ is a scaling factor specifying the step size. Thus, at the r-th step, only changes in the single component $\widehat{w}_h^{(r-1)}$ of vector $\widehat{w}^{(r-1)}$ are allowed in seeking a new vector $\widehat{w}^{(r)} \in \Omega$. The objective is to ensure that $\widehat{S}(\widehat{w}^{(r)}) < \widehat{S}(\widehat{w}^{(r-1)})$ and, under this respect, a safe choice for the search direction is to take a descendent coordinate h and a step size $\alpha^{(r)}$ such that

$$\nabla_h \widehat{S}(\widehat{w}^{(r-1)})\alpha^{(r)} < 0, \tag{9}$$

where $\nabla_h \widehat{S}(\widehat{w}^{(r-1)})$ corresponds to the partial derivative $\nabla_h \widehat{S}(w) = \partial S(w)/\partial w_h$, evaluated at $w = \widehat{w}^{(r-1)}$. In this context, the Gauss-Southwell rule consists in selecting, as the coordinate h for descent, that one corresponding to the largest component of the gradient vector in absolute value, that is

$$h = \text{argmax}_{u \in \{1,\ldots,J\}} |\nabla_u \widehat{S}(\widehat{w}^{(r-1)})|.$$

Furthermore, the computation of the step size $\alpha^{(r)}$ may be performed by line search, so that

$$\alpha^{(r)} = \text{argmin}_{\alpha \in A} \widehat{S}(\widehat{w}^{(r-1)} + \alpha\, d^{(r)}), \tag{10}$$

with $A = \{\alpha \in \mathbb{R} : \widehat{w}^{(r-1)} + \alpha d^{(r)} \in \Omega\}$ and $d^{(r)} = e_h$. As an alternative, a suitable constant step size may be defined: usually a fixed quantity with a small absolute value and the sign chosen in order to satisfy the descent condition (9).

We apply the coordinate descent algorithm to the objective function $\widehat{S}_H(f_p; w)$ defined by (4), which is, as showed in the Appendix A.2, a convex real-valued function, twice continuously differentiable, defined on $\Omega = \mathbb{R}^J_+$. Using the line search procedure specified by (10), we have that

$$\begin{aligned} \widehat{S}_H(f_p; \widehat{w}^{(r-1)} + \alpha\, d^{(r)}) &= \widehat{S}_H(f_p; \widehat{w}^{(r-1)}) + \alpha^2 \frac{1}{n} \sum_{i=1}^{n} \sum_{k=1}^{K} \left\{ \frac{\partial \log p_h(z^{(i)})}{\partial z_k} \right\}^2 \\ &+ \alpha \frac{2}{n} \sum_{i=1}^{n} \sum_{k=1}^{K} \left\{ \frac{\partial^2 \log p_h(z^{(i)})}{\partial z_k^2} + w^T P_k(z^{(i)}) \frac{\partial \log p_h(z^{(i)})}{\partial z_k} \right\}, \end{aligned}$$

where $d^{(r)} = e_h$, with h a fixed value in $\{1, \ldots, J\}$, and $\alpha \in \mathbb{R}$. Since this function is quadratic in the argument α, its global minimum satisfies the descent condition (9) and it is

$$\tilde{\alpha} = -\frac{\nabla_h \widehat{S}_H(f_p; \widehat{w}^{(r-1)})}{\nabla^2_{hh} \widehat{S}_H(f_p; \widehat{w}^{(r-1)})}, \tag{11}$$

provided that $\alpha \in A$, with $A = \{\alpha \in \mathbb{R} : \widehat{w}_h^{(r-1)} + \alpha \geq 0\}$. Here, $\nabla_h \widehat{S}_H(f_p; \widehat{w}^{(r-1)})$ is the h-th element of vector (7) and $\nabla^2_{hh} \widehat{S}_H(f_p; \widehat{w}^{(r-1)})$ is the (h, h) entry of matrix (8). In the univariate case, the u-th component of the gradient vector is simply specified as

$$\nabla_u \widehat{S}_H(f_p; w) = \frac{\partial \widehat{S}_H(f_p; w)}{\partial w_u} = \frac{2}{n} \sum_{i=1}^{n} \left[\frac{p''_u(z^{(i)})}{p_u(z^{(i)})} - \left\{ \frac{p'_u(z^{(i)})}{p_u(z^{(i)})} \right\}^2 \right] + \frac{2}{n} \sum_{i=1}^{n} \frac{p'_u(z^{(i)})}{p_u(z^{(i)})} \frac{f'_p(z^{(i)}; w)}{f_p(z^{(i)}; w)}, \tag{12}$$

with $u = 1, \ldots, J$, and the (u, v) element of the Hessian matrix corresponds to

$$\nabla^2_{uv} \widehat{S}_H(f_p; w) = \frac{\partial^2 \widehat{S}_H(f_p; w)}{\partial w_u \partial w_v} = \frac{2}{n} \sum_{i=1}^{n} \frac{p'_u(z^{(i)})}{p_u(z^{(i)})} \frac{p'_v(z^{(i)})}{p_v(z^{(i)})}, \quad u, v = 1, \ldots, J. \tag{13}$$

Note that equation (11) provides an analytical solution to the current optimization problem and this fact is not so common in the optimization algorithm research context.

Considering these findings, the algorithm which minimizes the empirical Hyvärinen loss (3) is defined as follows:

A.1 Inizialize: $\widehat{f}_p^{(0)}(z) \equiv f_p(z; \widehat{w}^{(0)})$, defined by (1) with $\widehat{w}^{(0)} \in \mathbb{R}_+^J$.

A.2 For $r = 1, \ldots, r_{\max}$:

(a) Chose $h = \text{argmax}_{u \in \{1,\ldots,J\}} |\nabla_u \widehat{S}_H(f_p; \widehat{w}^{(r-1)})|$, where $\nabla_u \widehat{S}_H(f_p; \widehat{w}^{(r-1)})$ is the u-th element of (7) with $w = \widehat{w}^{(r-1)}$.

(b) Find $\alpha^{(r)} = \text{argmin}_{\alpha \in A} \widehat{S}_H(f_p; \widehat{w}^{(r-1)} + \alpha\, d^{(r)})$, with $d^{(r)} = e_h$; thus, $\alpha^{(r)} = \max\{\tilde{\alpha}, -\widehat{w}_h^{(r-1)}\}$, with $\tilde{\alpha}$ defined by (11).

(c) Update: $\widehat{f}_p^{(r)}(z) \equiv f_p(z; \widehat{w}^{(r)}) \propto f_p(z; \widehat{w}^{(r-1)}) p_h(z)^{\alpha^{(r)}}$, with $\widehat{w}^{(r)} = \widehat{w}^{(r-1)} + \alpha^{(r)} e_h$.

A.3 Output: $\widehat{f}_p^{(r_{\max})}(z) \equiv f_p(z; \widehat{w}^{(r_{\max})})$.

This boosting-type algorithm, chosen for solving the optimization problem (6), is very simple and it is true that more advanced algorithms could be considered in order to achieve better convergence results. However, in this case, the application of the Gauss-Southwell coordinate descent procedure is usually quite feasible and not excessively demanding from the computational perspective. Furthermore, a regularization procedure is implicitly introduced, with the aim of considering only the most informative terms of the set $\mathcal{P}$. This can be particularly useful when the multiplicative mixture model (1) has a large number of components and it can be viewed as an alternative to other regularization methods such as those based on the lasso approach.

Regarding the structure of the algorithm, some variants and developments may be introduced. First, the step length α could be also considered as constant. More precisely, we may fix a small value for $|\alpha|$ and specify the sign for α as the opposite to that of the selected gradient, in order to satisfy the descent condition (9). Clearly, a small value for $|\alpha|$ requires an increased number of iterations,

and thus more computing time. However, empirical evidence supports the fact that, in this case, the predictive accuracy of f_p is usually more stable than that obtained choosing α using the line search procedure. Second, the algorithm stops after $r_{\max}$ iterations, where $r_{\max}$ may be defined according to a suitable stopping criterion, related to the stability of the value achieved by the objective function or to the closeness to zero of the associated gradient. Nevertheless, $r_{\max}$ may be considered as a tuning parameter to be defined in order to prevent overfitting, using cross-validation or other suitable predictive criteria. This topic is surely relevant for the effective implementation of the method and it will be considered in future research. In the applications presented in Section 5, the stopping criterion is simply based on the stability of the estimated objective function, by fixing a suitable value for the (relative) convergence tolerance.

Finally, an interesting interpretation of the rationale behind the algorithm is readily available and it turns out to be in accordance with the general idea behind boosting-type procedures. More precisely, if we add the negative quantity $-n^{-1}\sum_{i=1}^{n}\sum_{k=1}^{K}\{w^T P_k(z^{(i)})\}^2$ to the components of the gradient vector (7), we do not modify the output of the Step A.2(a) of the algorithm but we consider, instead of $\nabla_u \widehat{S}_H(f_p; \widehat{w}^{(r-1)})$, an estimate for

$$-d_H(f_p, p_u; w) + S_H(p_u; w),$$

based on the sample $z^{(1)}, \ldots, z^{(n)}$. Then, when all the quantities $\nabla_u \widehat{S}_H(f_p; \widehat{w}^{(r-1)})$, $u = 1, \ldots, J$, are negative, Step A.2(a) selects, as coordinate for descent, that one associated to a density p_h which is close to the true density f and distant from the actual estimate $\widehat{f}_p^{(r-1)}$. In this case, the corresponding weight will be increased by a positive quantity α. On the other hand, if the gradient components are all positive, the chosen coordinate indicates a density p_h which is distant from the true density f and close to the actual estimate $\widehat{f}_p^{(r-1)}$, and then it turns out to be penalized by reducing the associate weight. In the remaining situations, the procedure selects a suitable component density p_h, to be

penalized or upgraded according to the sign of the corresponding gradient component.

4 Inference for multiplicative model combinations

We have considered so far the simplified scenario where the component density functions $p_j(z)$, $j = 1, \ldots, J$, are supposed to be known and the interest has been focussed on the determination of a suitable estimate for the weights vector w. However, in most applications, the component densities depend on an unknown d-dimensional parameter $\theta = (\theta_1, \ldots, \theta_d) \in \Theta \subseteq \mathbb{R}^d$, $d \geq 1$, that needs to be inferred from the data. Sometimes, we have component models where the estimated parameters are obtained before combining them (see, for example, Geweke & Amisano, 2011), so that the optimization procedure is greatly simplified and it corresponds to the weights calibration discussed in the previous section. In general, when a joint optimization with respect to both θ and w is required, it is possible, as done in Section 3 for estimating w, to develop a tractable inferential procedure for the model parameter θ using the score matching divergence defined by Hyvärinen.

In this more general framework the empirical objective function, involving the Hyvärinen scoring rule, corresponds to $\widehat{S}_H(f_p; \theta, w)$, as given by (4) with $f_p(z; \theta, w) = c(\theta, w)^{-1} \prod_{j=1}^{J} p_j(z; \theta)^{w_j}$. Hereafter, we modify the notation in order to make explicit the dependence on the parameter θ. Thus, the aim is to specify the corresponding estimators for both θ and w, defined as $(\widehat{\theta}, \widehat{w}) = \text{argmin}_{\theta,w} \widehat{S}_H(f_p; \theta, w)$. However, since this expression is usually hard to minimize, we adopt a different objective function where the parameter w is profiled out by considering a vector of functions $\widehat{w}_\theta = (\widehat{w}_1(\theta), \ldots, \widehat{w}_J(\theta))$ such that $\widehat{w}_\theta = \text{argmin}_w \widehat{S}_H(f_p; \theta, w)$, for a fixed $\theta \in \Theta$. Thus, the estimator for θ corresponds to

$$\widehat{\theta} = \text{argmin}_\theta \widehat{S}_H(f_p; \theta, \widehat{w}_\theta) \tag{14}$$

and then the estimator for w may be defined as $\widehat{w} = \widehat{w}_{\widehat{\theta}}$. Even if $\widehat{w}_\theta$, for a given value of θ, can be

obtained at least approximatively using the boosting-type algorithm outlined in the previous section, the calculation of the optimizer $\widehat{\theta}$ could be difficult. Furthermore, since an explicit solution for this inferential problem is usually not available, the estimate has to be computed using iterative methods such as Newton-Raphson, and the associated variants, or derivative-free methods. Concerning the first class of algorithms, the implementation can be problematic because the J-dimensional function $\widehat{w}_\theta$ is difficult to evaluate and then to differentiate. Derivative-free procedures may be a feasible alternative, provided that the computational burden does not make the search of the optimizer inefficient and too slow.

In this section, we focus on the case where $\widehat{\theta}$ is defined as the unique solution of the score-type equation $\nabla_\theta \widehat{S}_H(f_p; \theta, \widehat{w}_\theta) = 0$, which can be expressed in the following equivalent form

$$\nabla_\theta \widehat{S}_H(f_p; \theta, w)|_{w=\widehat{w}_\theta} + (\nabla_\theta \widehat{w}_\theta)^T \nabla_w \widehat{S}_H(f_p; \theta, w)|_{w=\widehat{w}_\theta} = 0, \tag{15}$$

where $\nabla_\theta \widehat{S}_H(f_p; \theta, w)$ and $\nabla_w \widehat{S}_H(f_p; \theta, w)$ are the gradient vectors, expressed as column vectors, associated to the components θ and w, respectively, and $\nabla_\theta \widehat{w}_\theta$ is a matrix with (u, r) element $\partial \widehat{w}_u(\theta)/\partial \theta_r$, $u = 1, \ldots, J$, $r = 1, \ldots, d$. Instead of trying to solve directly equation (15) with respect to θ, we employ the iterative backfitting algorithm introduced by Fan *et al.* (2015). This scheme can be adapted to the present problem as follows:

B.1 Inizialize: $\widehat{\theta}^{(0)} \in \Theta$.

B.2 For $s = 1, \ldots, s_{\max}$:

(a) Find $\widehat{w}_{\widehat{\theta}^{(s-1)}} = \text{argmin}_w \widehat{S}_H(f_p; \widehat{\theta}^{(s-1)}, w)$, using, for example, the algorithm defined in Section 3.

(b) Define $\widehat{\theta}^{(s)}$ as the solution, with respect to θ, of

$$\nabla_\theta \widehat{S}_H(f_p; \theta, \widehat{w}_{\widehat{\theta}^{(s-1)}}) + (\nabla_\theta \widehat{w}_{\widehat{\theta}^{(s-1)}})^T \nabla_w \widehat{S}_H(f_p; \widehat{\theta}^{(s-1)}, \widehat{w}_{\widehat{\theta}^{(s-1)}}) = 0. \tag{16}$$

B.3 Find $\widehat{w}_{\widehat{\theta}^{(s_{\max})}} = \text{argmin}_w \widehat{S}_H(f_p; \widehat{\theta}^{(s_{\max})}, w)$

B.4 Output: $\widehat{\theta} = \widehat{\theta}^{(s_{\max})}$, $\widehat{w} = \widehat{w}_{\widehat{\theta}^{(s_{\max})}}$.

The algorithm stops after $s_{\max}$ iterations, with $s_{\max}$ specified by means of a convenient stopping criterion. With regard to the conditions assuring the consistency of the estimators defined by the above algorithm, we recall some standard result in the Appendix A.3.

Note that, whenever the minimizer $\widehat{w}_{\widehat{\theta}^{(s-1)}}$ is such that $\nabla_w \widehat{S}_H(f_p; \widehat{\theta}^{(s-1)}, \widehat{w}_{\widehat{\theta}^{(s-1)}}) = 0$, for each iteration $s = 1, \ldots, s_{\max}$, Step B.2(b) turns out to be greatly simplified, since equation (16) becomes

$$\nabla_\theta \widehat{S}_H(f_p; \theta, \widehat{w}_{\widehat{\theta}^{(s-1)}}) = 0.$$

In this case, the differentiation of $\widehat{w}_\theta$ with respect to θ, which is usually problematic, becomes unnecessary and the optimization algorithm corresponds to a basic backfitting procedure, which, in this particular situation, gives an efficient estimator for θ.

5 Applications

5.1 Autoregressive models with additive observation noise

In this first application, we consider a simple situation where the model function is available in a closed form. Then, we can show how the boosting-type algorithm works and we can compare the multiplicative mixture density $f_p(z; w)$ with the true density function $f(z)$, analytically available in this case. Indeed, since the likelihood function is also available, we can check the inferential procedure based on the Hyvärinen score against the benchmark corresponding to the likelihood approach.

We consider the following first-order autoregressive model with additive observation noise,

$$\begin{aligned} Y_t &= \beta + X_t + V_t, \quad t \geq 1, \\ X_t &= \gamma X_{t-1} + W_t, \quad t \geq 1, \end{aligned}$$

with $V_t \sim N(0, \sigma^2)$, $W_t \sim N(0, \tau^2)$, $t \geq 1$, mutually independent Gaussian random variables. We assume that $X_0 \sim N(0, \tau^2/(1-\gamma^2))$ and that the latent autoregressive process X_t, $t \geq 0$, is stationary, being $|\gamma| < 1$. We observe $Y = (Y_1, \ldots, Y_m)$, $m \geq 1$, and we aim at predicting the future random variable $Z = Y_{m+1}$. The parameter $\theta = (\beta, \sigma^2, \gamma, \tau^2)$ is unknown. In this elementary example the likelihood function and the conditional density of Z given $Y = y$, which corresponds to the interest density function $f(z; \theta)$, are available in a closed form and they can be efficiently computed by means of Kalman filter recursions. Moreover, as component density function $p_j(z; \theta)$, $j = 1, \ldots, J$, with a fixed value for $J \in \{1, \ldots, m\}$, we consider the density of $Z|Y_{m+1-j} = y_{m+1-j}$, namely the pairwise conditional density of Z given the observation at a lag distance j.

It is easy to see that $Y_i|Y_{i-j} = y_{i-j} \sim N(\mu_{i,j}, \sigma^2_{i,j})$, $i = J+1, \ldots, m+1$, where $\mu_{i,j} = \beta + \rho_{i,j}(y_{i-j} - \beta)$ and $\sigma^2_{i,j} = (1 - \rho^2_{i,j})\{\sigma^2 + \tau^2/(1-\gamma^2)\}$, with $\rho_{i,j} = \{\tau^2\gamma^j/(1-\gamma^2)\}/\{\sigma^2 + \tau^2/(1-\gamma^2)\}$ the correlation coefficient of Y_i and Y_{i-j}. Therefore, $p_j(z; \theta) = \phi(z; \mu_{m+1,j}, \sigma^2_{m+1,j})$, namely a Gaussian density with mean $\mu_{m+1,j}$ and variance $\sigma^2_{m+1,j}$, and we can conclude that the multiplicative mixture density (1) corresponds to $f_p(z; \theta, w) = \phi(z; b_{m+1}, s^2_{m+1})$ with

$$b_{m+1} = \frac{\sum_{j=1}^{J} \mu_{m+1,j} w_j/(1 - \rho^2_{m+1,j})}{\sum_{j=1}^{J} w_j/(1 - \rho^2_{m+1,j})}, \quad s^2_{m+1} = \frac{\sigma^2 + \tau^2/(1-\gamma^2)}{\sum_{j=1}^{J} w_j/(1 - \rho^2_{m+1,j})}. \tag{17}$$

We present the results of two simulation studies. In the first one, given the observations $y = (y_1, \ldots, y_m)$, we compute the maximum likelihood estimate $\widehat{\theta}_{ML}$ for θ and we compare the target conditional density $f(z; \theta)$, evaluated at $\theta = \widehat{\theta}_{ML}$, with the multiplicative mixture density $f_p(z; \widehat{\theta}_{ML}, w)$, where the weights are specified by the boosting-type algorithm introduced in Section 3. We consider also, for a further comparison, the additive mixture density $\tilde{f}_p(z; \widehat{\theta}_{ML}, w)$, where the weights are estimated using the log score. In the second one, given a set of simulated observations for $Y = (Y_1, \ldots, Y_m)$, we analyse the behaviour of the estimator for θ introduced in Section 4 using the Hyvärinen score function. In both applications, since we have observations from the random

vector Y, we consider optimization procedures where the objective function is given by the following empirical average

$$\widehat{S}_H(f_p;\theta,w)=\frac{1}{m-J}\sum_{i=J+1}^{m}\left[2\frac{f_p''(y_i;\theta,w)}{f_p(y_i;\theta,w)}-\left\{\frac{f_p'(y_i;\theta,w)}{f_p(y_i;\theta,w)}\right\}^2\right], \tag{18}$$

obtained as (5) with $f_p(z^{(i)};\theta,w)$ substituted by $f_p(y_i;\theta,w)$, namely the multiplicative mixture density of Y_i given $(Y_1,\ldots,Y_{i-1})=(y_1,\ldots,y_{i-1})$. Notice that, since $f_p(y_i;\theta,w)$ is a Gaussian density, we get

$$\frac{f_p'(y_i;\theta,w)}{f_p(y_i;\theta,w)}=-\frac{y_i-b_i}{s_i^2},\quad \frac{f_p''(y_i;\theta,w)}{f_p(y_i;\theta,w)}=\frac{1}{s_i^2}\left\{\frac{(y_i-b_i)^2}{s_i^2}-1\right\},$$

where b_i and s_i^2 are given by (17) with $m+1$ substituted by i.

Let us first consider time series of dimension $m=100,200,500$ simulated from a first-order autoregressive model plus additive observation noise, with $\beta=0.2$, $\sigma=1$, $\tau=1$, (a) $\gamma=0.5$, (b) $\gamma=0.85$ and (c) $\gamma=0.95$ as true parameter values. Similar results are obtained with alternative choices for θ. The unknown parameter θ is estimated with the maximum likelihood estimator $\widehat{\theta}_{ML}$, obtained by considering the likelihood function calculated with the Kalman filter recursions. The weights w are specified using the boosting-type algorithm defined at the end of Section 3, considering as objective function the empirical average $\widehat{S}_H(f_p;\widehat{\theta}_{ML},w)$, given by (18) evaluated at $\theta=\widehat{\theta}_{ML}$. We emphasize that, since in this case a sample from the interest random variable Z is not available, this objective function can be interpreted as a convenient estimate for the expected Hyvärinen score (2) for $K=1$, under stationary assumption. As a consequence, the u-th component of the gradient vector and the (u,v) entry of the Hessian matrix are given by (12) and (13), respectively, with $f_p(y_i;\widehat{\theta}_{ML},w)$ substituted for $f_p(z^{(i)};w)$ and $p_j(y_i;\widehat{\theta}_{ML})$ substituted for $p_j(z^{(i)})$. Since $p_j(y_i;\widehat{\theta}_{ML})$ is a Gaussian density, we have that

$$\frac{p_j'(y_i;\widehat{\theta}_{ML})}{p_j(y_i;\widehat{\theta}_{ML})}=-\frac{y_i-\widehat{\mu}_{i,j}}{\widehat{\sigma}_{i,j}^2},\quad \frac{p_j''(y_i;\widehat{\theta}_{ML})}{p_j(y_i;\widehat{\theta}_{ML})}=\frac{1}{\widehat{\sigma}_{i,j}^2}\left\{\frac{(y_i-\widehat{\mu}_{i,j})^2}{\widehat{\sigma}_{i,j}^2}-1\right\},$$

where a hat means evaluation at $\theta=\widehat{\theta}_{ML}$.

In the following Table 1 we show the optimal weights given by the boosting algorithm. We consider the variant of the algorithm with a fixed step length 0.001, which gives more stable results, and we assume a maximum lag distance $J = 10$, since the model is supposed to present a relatively short range dependence.

Table 1 here

Furthermore, in Figure 2 we compare the target conditional density of Z given $Y = y$, evaluated at $\theta = \widehat{\theta}_{ML}$, with the multiplicative mixture density $f_p(z; \widehat{\theta}_{ML}, w)$, considering both the optimal weights given in Table 1 and equal weights $w_1 = w_2 = w_3 = 1/3$, with a maximum lag distance $J = 3$. We show also the additive mixture density $\tilde{f}_p(z; \widehat{\theta}_{ML}, w)$, considering both the optimal weights given by maximum likelihood procedure and equal weights as before. We find that, as expected, the multiplicative mixture density, with weights given by the boosting procedure, provides a satisfactory approximation for the true density almost everywhere, and in particular for large simulated samples. The optimal additive mixture density usually gives a good approximation, even if it seems less stable than the multiplicative one.

Figure 2 here

In the second study, we consider $1,000$ simulated time series of dimension $m = 250,500$ from a first-order autoregressive model plus additive observation noise, with $\beta = 0.2$, $\sigma = 1$, $\tau = 1$, (a) $\gamma = 0.5$ and (b) $\gamma = 0.75$ as true parameter values. In order to simplify the optimization procedure, and to guarantee identifiability in the objective function, we assume $\tau = 1$ as known and $\theta = (\beta, \sigma, \gamma)$ as the unknown parameter. We aim at comparing the empirical properties of the maximum likelihood estimator $\widehat{\theta}_{ML}$, available in this case, and of the estimator $\widehat{\theta}_H$ based on the Hyvärinen score function. We consider also two estimators, $\widehat{\theta}_{PW1}$ and $\widehat{\theta}_{PW2}$, based on the conditional pairwise likelihood with a maximum lag distance $J = 3$ (see for example, Varin & Vidoni, 2009, and Varin *et al.*, 2011). The

first one is obtained by considering the optimal weights given by the boosting-type procedure, while the second one is simply based on equal weights $w_1 = w_2 = w_3 = 1/3$.

We emphasize that $\widehat{\theta}_H$ is computed using the algorithm outlined in Section 4, where the objective function corresponds to the empirical average $\widehat{S}_H(f_p; \theta, w)$ as given by (18). Note that this function, with w fixed, can be viewed as an alternative to the likelihood function, obtained using the Hyvärinen's divergence. Thus, similarly to the pairwise likelihood approach and, more generally to the composite likelihood approach, we base our inferential procedure on a function which can be interpreted as a surrogate of the likelihood function, whenever unknown or not available.

The sample means and standard errors for the four estimators are summarized in Table 2.

Table 2 here

We find that, whereas the location parameter β is always well estimated, for the remaining parameters, the quality of the estimator based on the conditional pairwise likelihood with equal weights is considerably reduced, since there is a substantial bias and an overall low accuracy. Notice that, when the weights are calculated with the boosting-type procedure, the performance of the conditional pairwise likelihood estimator increases and this confirms that a crucial point in the composite likelihood inferential approach is the choice of the likelihood objects and, more generally, the specification of the system of weights (see, for example, the discussion in Varin & Vidoni, 2009). Finally, we emphasize that the accuracy of the estimator based on the Hyvärinen score function is similar to that of the benchmark estimator, namely the maximum likelihood estimator.

5.2 Model pools for time series prediction

In this second application, we consider a real-valued, univariate time series $Y = (Y_h, \ldots, Y_T)$, with $h \leq 0$ and $T \geq 1$ the starting and the ending dates of the series, respectively. The aim is to predict the random variable $Z = Y_{m+1}$, for $m = 0, \ldots, T$, given the observations $y_h^m = (y_h, \ldots, y_m)$ of the random

variables $Y_h^m = (Y_h, \ldots, Y_m)$, available at time m. Let us suppose that there are $J > 1$ alternative parametric statistical models for the time series Y, giving J different conditional density functions for Y_{m+1} given $Y_h^m = y_h^m$, namely $p_j(y_{m+1}|y_h^m; \theta_j)$, with θ_j the unknown model parameter. Moreover, using the observations y_h^m available at time m, we specify a suitable estimate for the parameter θ_j, usually the maximum likelihood estimate $\widehat{\theta}_j^m = \widehat{\theta}_j(y_h^m)$, and then we consider, as prediction models for Y_{m+1}, the estimative predictive densities $p_j(y_{m+1}|y_h^m; \widehat{\theta}_j^m)$, $j = 1, \ldots, J$.

In this study we assume that there is an ergodic data generating process underlying the time series Y and we focus on the evaluation of the predictive accuracy of the J different prediction models and of the associated additive and multiplicative pools given, respectively, by

$$f_p(y_{m+1}; \widehat{\theta}^m, w) = c(\widehat{\theta}^m, w)^{-1} \prod_{j=1}^{J} p_j(y_{m+1}|y_h^m; \widehat{\theta}_j^m)^{w_j}, \quad \tilde{f}_p(y_{m+1}; \widehat{\theta}^m, \tilde{w}) = \sum_{j=1}^{J} \tilde{w}_j p_j(y_{m+1}|y_h^m; \widehat{\theta}_j^m),$$

with $\widehat{\theta}^m = (\widehat{\theta}_1^m, \ldots, \widehat{\theta}_J^m)$, $w = (w_1, \ldots, w_J)^T$ and $\tilde{w} = (\tilde{w}_1, \ldots, \tilde{w}_J)^T$ the vectors of weights and $c(\widehat{\theta}^m, w)$ the normalizing constant, supposed to be finite. Furthermore, we evaluate these densities using both the sample Hyvärinen score, specified by extending equation (3) to the time series framework,

$$\widehat{S}_H(g) = \frac{1}{T} \sum_{t=1}^{T} \left[2 \frac{d^2 \log g(y_t|y_h^{t-1}; \widehat{\theta}^{t-1})}{dy_t^2} + \left\{ \frac{d \log g(y_t|y_h^{t-1}; \widehat{\theta}^{t-1})}{dy_t} \right\}^2 \right],$$

and the sample log score (see, for example, Geweke & Amisano, 2011)

$$\widehat{S}_L(g) = \sum_{t=1}^{T} \log g(y_t|y_h^{t-1}; \widehat{\theta}^{t-1}),$$

with $g(\cdot)$ a generic prediction model. As usual for temporal data, we consider a rolling sample y_h^0 and the first predictive density refers to time $t = 1$. In particular, we use the sample Hyvärinen score in order to find the optimal weights w for the multiplicative pool, defined as those ones minimizing the objective function

$$\widehat{S}_H(f_p; w) = \frac{1}{T} \sum_{t=1}^{T} \left[2 \sum_{j=1}^{J} w_j \frac{d^2 \log p_j(y_t|y_h^{t-1}; \widehat{\theta}_j^{t-1})}{dy_t^2} + \left\{ \sum_{j=1}^{J} w_j \frac{d \log p_j(y_t|y_h^{t-1}; \widehat{\theta}_j^{t-1})}{dy_t} \right\}^2 \right],$$

and the sample log score in order to find the optimal weights $\tilde{w}$ for the additive pool, defined as those ones maximizing the objective function

$$\widehat{S}_L(\tilde{f}_p; \tilde{w}) = \sum_{t=1}^{T} \log \left\{ \sum_{j=1}^{J} \tilde{w}_j p_j(y_t | y_h^{t-1}; \widehat{\theta}_j^{t-1}) \right\}.$$

We discuss and compare the properties of these two alternative model pooling procedures by considering the same data set analized by Geweke & Amisano (2011, Sections 3 and 5) for illustrating the usefulness of a linear model pool strategy. More precisely, we use the daily percent log returns of the Standard and Poors 500 index (S&P500) from January 3, 1972 (which corresponds to $h = -1249$) to December 16, 2005 ($T = 7324$), with December 14, 1976 ($t = 1$) as the first date for prediction. We consider six alternative models for the returns, namely a Gaussian independent, identically distributed (i.i.d.) model (abbreviated as Gaussian), a Student t i.i.d. model (Student), a Gaussian GARCH(1,1) model (GARCH), a GARCH(1,1) model with Student t errors (t-GARCH), a Gaussian exponential GARCH(1,1) model (EGARCH) and an exponential GARCH(1,1) model with Student t errors (t-EGARCH). Geweke & Amisano (2011) consider the same models except the second and the last ones and they take into account two further models, giving Bayesian predictive densities using MCMC simulations. Here we follow a non-Bayesian approach and all of the models are sequentially estimated by maximum likelihood using the R package rugarch (Ghalanos, 2014), with a rolling sample of 1250 trading days.

Table 3 provides the sample Hyvärinen score and the sample log score for each model. We notice that both criteria strongly favour the t-EGARCH and the t-GARCH models, since they present the lowest values for the Hyvärinen score and the highest values for the log score. The remaining models have different ratings, with the Gaussian one always in the last position.

Table 3 here

Table 4 shows the weights for the optimal multiplicative pool, and the associated sample Hyvärinen

score, and the weights for the optimal linear pool, and the associated sample log score. The optimization procedures are trivial and they require less than 1 second, provided that the component estimative predictive densities are available. In the optimal multiplicative pool the t-GARCH and the t-EGARCH models are jointly dominant; these models present the highest weights also in the optimal linear pool. Notice that t-GARCH and t-EGARCH are the best models highlighted in Table 3. By considering the impact on the sample score of excluding one of these models, we state that in both cases the weights describe the relevance of the model's contribution to the sample score. However, we have to declare that this correspondence does not have a general validity, as noticed by Geweke & Amisano (2011, Section 5). It is easy to specify a multiplicative pool and a linear pool with only the two dominant models, assuming equal weights. For the first pool, the sample Hyvärinen score is -1.729 and, for the second one, the sample log score is -9282.93; these values are quite close to those of the optimal pools. Moreover, as noted by Geweke & Amisano (2011), the objective function for the linear pool, based on the sample log score, tends to be rather flat near its mode, so that if we move away from the optimal solution only a small relative reduction in the function value is produced. On the other hand, if we consider the objective function for the multiplicative pool, based on the sample Hyvärinen score, we usually obtain values with an higher relative difference.

The optimization procedures considered so far are based on the entire observed sample and, for this reason, they are not useful for day-by-day prediction. Thus, taking this into account, we compute the sequence of optimal weights obtained each day by using only the current available data. The associated sample scores provide more effective measures for the predictive accuracy of the model pools. In Table 4 we report also the sample Hyvärinen score for a multiplicative pool and the sample log score for a linear pool with weights reoptimized each day. Although the value for the linear pool is close to the value obtained for the pool optimized on the entire sample, with regard to the multiplicative pool we observe a substantial relative improvement. Furthermore, we represent in Table 4 the mean

value of the sequentially optimized weights (excluding the first 1000 values which exhibit remarkable oscillations) and we notice that these values are significantly different from those computed using the entire sample. Finally, the temporal evolution of the optimal weights for the multiplicative pool is described in Figure 3. We notice that there are periods characterized by high variability and that, as the time increases, the weight of the t-EGARCH model tends to increase and that of the t-GARCH model shows a consequent reduction. A similar pattern is observed for the optimal weights of the linear pool, although with less intense oscillations.

Figure 3 here

5.3 Estimation of covariance and precision matrices

In this third application, we draw attention to the usefulness of the boosting-type algorithm based on the Hyvärinen's divergence for estimating covariance and precision matrices associated to a multivariate Gaussian distribution. The precision matrix is defined as the inverse of the covariance matrix. When the dimension of the interest matrix is large, with respect to the sample size, this estimation problem could be challenging. It is well-known that the sample covariance matrix is singular when its dimension is larger than the sample size and that, also for a moderate dimension, it is a rather unstable estimator. Moreover, with regard to the precision matrix, it is of particular interest the identification of zero entries, indicating that the corresponding component random variables are conditional independent given all the others. The inverse of the sample covariance matrix usually fails to identify zero entries and then it is not useful for describing the potential sparse structure (namely, with a large number of zero entries) of the precision matrix. The estimation of the precision matrix is relevant for the specification of Gaussian graph models (Cox & Wermuth, 1966) and of Gaussian Markov random fields (Rue & Held, 2005), where the conditional dependence structure is commonly represented by an undirected graph; the edge between two component random variables is absent if

and only if they are conditional independent given the other variables, therefore the corresponding entry of the precision matrix is zero.

Let Z be a K-dimensional random vector following a multivariate Gaussian distribution $N_K(\mu, \Sigma)$, with unknown mean vector μ and non-singular covariance matrix $\Sigma = (\sigma_{rs})$; then, the target density $f(z) = \phi(z; \mu, \Sigma)$ is a multivariate Gaussian density. Given a sample $z^{(1)}, \ldots, z^{(n)}$, with $z^{(i)} = (z_1^{(i)}, \ldots, z_K^{(i)})^T$, $i = 1, \ldots, n$, from $f(z)$, the maximum likelihood estimates of μ and Σ are, respectively,

$$\bar{z} = \left(\frac{1}{n} \sum_{i=1}^{n} z_1^{(i)}, \ldots, \frac{1}{n} \sum_{i=1}^{n} z_K^{(i)} \right)^T, \quad \widehat{\Sigma} = \frac{1}{n} \sum_{i=1}^{n} (z^{(i)} - \bar{z})(z^{(i)} - \bar{z})^T,$$

while the commonly used sample covariance matrix is $S = n\widehat{\Sigma}/(n-1)$. Therefore, it is almost immediate to estimate the precision matrix $Q = \Sigma^{-1} = (q_{rs})$ using $\widehat{\Sigma}^{-1}$ or S^{-1}. Since these estimators, as mentioned before, are usually non satisfactory, alternative inferential procedures have been introduced, focusing on various regularization techniques such as lasso (see, for example, Friedman *et al.*, 2007, and Fan *et al.*, 2016). Given that our focus is on the covariance and the precision matrices, we assume that the observations are suitably centred, so that we can consider a null mean vector $\mu = 0$.

In the following, we propose a new method for estimating the precision matrix of a multivariate Gaussian distribution, based on the multiplicative combination of a number of multivariate Gaussian distributions having a simple covariance structure; the optimization procedure makes use of the boosting-type algorithms presented in the preceding sections. More precisely, as component density function $p_j(z; \theta)$, $j = 1, \ldots, J$, we consider $\phi(z; 0, Q_j^{-1})$, namely a K-variate Gaussian density with a null mean vector and a suitable symmetric precision matrix $Q_j = Q_j(\theta) = (q_{j,rs}(\theta))$. Thus, using well-known properties of the Gaussian distribution, we can conclude that the multiplicative mixture density (1) corresponds to $f_p(z; \theta, w) = \phi(z; 0, (\sum_{j=1}^{J} w_j Q_j)^{-1})$. Using the algorithm presented in Section 4, we find the values $\widehat{w}$ and $\widehat{\theta}$ assuring that the distance between the unknown density $f(z)$ and the mixture density $f_p(z; \theta, w)$, in terms of Hyvärinen's divergence, reaches its minimum and then

$\widehat{Q}_H = \sum_{j=1}^{J} \widehat{w}_j Q_j(\widehat{\theta})$ can be viewed as a convenient estimator for the unknown precision matrix Q, provided that it is symmetric and positive-definite. The use of the Hyvärinen's divergence is essential for reducing the computational burden of the optimization procedure, since the calculation of the Gaussian normalizing constant is not required.

The estimator $\widehat{Q}_H$ is defined as a linear combination of a set of simple precision matrices $\widehat{Q}_j$, $j = 1, \ldots, J$, giving a partial description of the conditional covariance structure of the random vector Z. The choice of this system of matrices is crucial for the effectiveness of the inferential procedure. For example, if we know that Q is a band matrix, namely it is a sparse matrix with non-zero entries on a diagonal band with unknown dimension, we may consider

$$Q_1 = \begin{pmatrix} q_{1,11} & 0 & 0 & \cdots & 0 \\ 0 & q_{1,22} & 0 & \ddots & 0 \\ 0 & 0 & q_{1,33} & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & q_{1,KK} \end{pmatrix}, Q_2 = \begin{pmatrix} q_{2,11} & q_{2,12} & 0 & \cdots & 0 \\ q_{2,21} & q_{2,22} & q_{2,23} & \ddots & 0 \\ 0 & q_{2,32} & q_{2,33} & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & q_{2,(K-1)K} \\ 0 & 0 & \cdots & q_{2,K(K-1)} & q_{2,KK} \end{pmatrix},$$

$$Q_3 = \begin{pmatrix} q_{3,11} & 0 & q_{3,13} & \cdots & 0 \\ 0 & q_{3,22} & 0 & \ddots & q_{3,3K} \\ q_{3,31} & 0 & q_{3,33} & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & q_{3,K(K-1)} & \cdots & q_{3,KK} \end{pmatrix}, \cdots, Q_J = \begin{pmatrix} q_{J,11} & 0 & 0 & \cdots & q_{J,1K} \\ 0 & q_{J,22} & 0 & \ddots & 0 \\ 0 & 0 & q_{J,33} & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ q_{J,K1} & 0 & 0 & \cdots & q_{J,KK} \end{pmatrix},$$

with $J = K$. In this case, matrices Q_j, $j = 2, \ldots, J$ present the main diagonal and only two equal non-null, symmetric diagonals and, with a suitable choice for the weights w, they may be combined in order to obtain an estimate for the true unknown band matrix Q. Clearly, if the bandwidth of Q (namely, the number of diagonals with non-null elements) is $M \geq 1$, we expect that all the matrices Q_j, with $j > (M+1)/2$, have null off-diagonal entries.

More broadly, whenever the conditional covariance structure is completely unknown, we have to define an extremely general system of component precision matrices such as

$$Q_1 = \begin{pmatrix} q_{1,11} & 0 & 0 & \cdots & 0 \\ 0 & q_{1,22} & 0 & \ddots & 0 \\ 0 & 0 & q_{1,33} & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & q_{1,KK} \end{pmatrix}, Q_2 = \begin{pmatrix} q_{2,11} & q_{2,12} & 0 & \cdots & 0 \\ q_{2,21} & q_{2,22} & 0 & \ddots & 0 \\ 0 & 0 & q_{2,33} & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & q_{2,KK} \end{pmatrix},$$

$$Q_3 = \begin{pmatrix} q_{3,11} & 0 & q_{3,13} & \cdots & 0 \\ 0 & q_{3,22} & 0 & \ddots & 0 \\ q_{3,31} & 0 & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & q_{3,KK} \end{pmatrix}, \cdots, Q_J = \begin{pmatrix} q_{J,11} & 0 & 0 & \cdots & 0 \\ 0 & q_{J,22} & 0 & \ddots & 0 \\ 0 & 0 & q_{J,33} & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & q_{J,(K-1)K} \\ 0 & 0 & \cdots & q_{J,K(K-1)} & q_{J,KK} \end{pmatrix},$$

with $J = 1 + K(K-1)/2$. Here, each matrix Q_j, $j = 2, \ldots, J$, presents the main diagonal and only two equal, non-null symmetric values and it is obvious that the computational cost for estimating the true unknown precision matrix Q could be substantial when K is large.

In this framework, given the sample $z^{(1)}, \ldots, z^{(n)}$, we consider as objective function the empirical average (4) given by

$$\widehat{S}_H(f_p; \theta, w) = \frac{1}{n} \sum_{i=1}^{n} \sum_{k=1}^{K} \left[\left\{ \sum_{j=1}^{J} w_j \sum_{s=1}^{K} z_s^{(i)} q_{j,ks}(\theta) \right\}^2 - 2 \sum_{j=1}^{J} w_j q_{j,kk}(\theta) \right].$$

Moreover, it is easy to see that the u-th component of the gradient vector and the (u, v) entry of the Hessian matrix are given, respectively, by

$$\nabla_u \widehat{S}_H(f_p; \theta, w) = \frac{2}{n} \sum_{i=1}^{n} \sum_{k=1}^{K} \left[\left\{ \sum_{j=1}^{J} w_j \sum_{s=1}^{K} z_s^{(i)} q_{j,ks}(\theta) \right\} \left\{ \sum_{s=1}^{K} z_s^{(i)} q_{u,ks}(\theta) \right\} \right] - 2 \sum_{k=1}^{K} q_{u,kk}(\theta),$$

with $u = 1, \ldots, J$, and

$$\nabla^2_{u,v} \widehat{S}_H(f_p; \theta, w) = \frac{2}{n} \sum_{i=1}^{n} \sum_{k=1}^{K} \left[\left\{ \sum_{s=1}^{K} z_s^{(i)} q_{u,ks}(\theta) \right\} \left\{ \sum_{s=1}^{K} z_s^{(i)} q_{v,ks}(\theta) \right\} \right], \quad u, v = 1, \ldots, J.$$

We emphasize that the optimization procedure has to deal with many unknown quantities, namely the weights $w = (w_1, \ldots, w_J)^T$, specifying the multiplicative mixture model, and the non-null entries

of the component precision matrices Q_j, $j = 1, \ldots, J$. In order to guarantee identifiability in the objective function, we assume $q_{1,rr} = \theta_r$ and $q_{j,rr} = 0$, $j = 2, \ldots, J$, $r = 1, \ldots, K$; furthermore, all the non-null, off-diagonal elements of the matrices are considered as equal to 1. Thus, matrix Q_1 (with a fixed weight $w_1 = 1$) is a diagonal matrix defining the conditional precision of each marginal component of vector Z, whereas the remaining matrices Q_j, $j = 2, \ldots, J$, specify the presence of some specific non-null conditional correlations, whose values are defined by the corresponding weights w_j, $j = 2, \ldots, J$. For example, the precision matrix $\sum_{j=1}^{J} w_j Q_j$ obtained from the two systems of component precision matrices outlined before corresponds, respectively, to

$$\begin{pmatrix} \theta_1 & w_2 & w_3 & \cdots & w_K \\ w_2 & \theta_2 & w_2 & \ddots & w_{K-1} \\ w_3 & w_2 & \theta_3 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & w_2 \\ w_K & w_{K-1} & \cdots & w_2 & \theta_K \end{pmatrix}, \quad \begin{pmatrix} \theta_1 & w_2 & w_3 & \cdots & w_K \\ w_2 & \theta_2 & w_{K+1} & \ddots & w_{2K-2} \\ w_3 & w_{K+1} & \theta_3 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & w_{1+K(K-1)/2} \\ w_K & w_{2K-2} & \cdots & w_{1+K(K-1)/2} & \theta_K \end{pmatrix}.$$

Notice that, even if the component matrices Q_j, $j = 2, \ldots, J$, present a null main diagonal, and are expected to be singular, the optimization procedure can be applied in the same way, giving a useful estimated precision matrix $\widehat{Q}_H$. Moreover, in the following simulation study, we use the optimization scheme introduced in Section 4, where the weights $w_2, \ldots, w_J$ and the parameter $\theta = (\theta_1, \ldots, \theta_K)$ are recursively estimated using the boosting algorithm and a direct minimization of the Hyvärinen's divergence, respectively.

We consider 100 simulated samples of dimension $n = 100, 200, 500$ from a K-variate Gaussian distribution with $K = 10, 20$, having a null mean vector and a band precision matrix Q with non-null entries $q_{kk} = 2$, $k = 1, \ldots, K$, $q_{k(k-1)} = q_{(k-1)k} = -0.5$, $k = 2, \ldots, K$, $q_{k(k-2)} = q_{(k-2)k} = 0.4$, $k = 3, \ldots, K$, $q_{k(k-3)} = q_{(k-3)k} = -0.3$, $k = 4, \ldots, K$, and $q_{k(k-4)} = q_{(k-4)k} = 0.2$, $k = 5, \ldots, K$. We aim at comparing the empirical properties of the following estimators for Q: $\widehat{Q}_{H1}$, based on the boosting-type algorithm assuming a band structure for the component precision matrices (the first case

considered above), $\widehat{Q}_{H2}$, based on the boosting-type algorithm without assuming a specific structure for the component precision matrices (the second case considered above), $\widehat{Q}_{GL}$, based on the graphical lasso algorithm proposed by Mazumder & Hastie (2012) and S^{-1}, corresponding to the inverse of the sample covariance matrix. With regard to the boosting algorithm, we consider the variant with a fixed step length 0.005, which gives more stable results, and we assume as initial value for θ the sample estimates of the conditional precision of each marginal component of vector Z.

We compare the alternative methods in terms of the Kullback-Leibler loss between the true $\phi(z; 0, Q^{-1})$ and the estimated $\phi(z; 0, \widehat{Q}^{-1})$ Gaussian densities

$$\mathrm{KL} = \mathrm{tr}(Q^{-1}\widehat{Q}) - \log(|Q^{-1}\widehat{Q}|) - K,$$

where $\mathrm{tr}(\cdot)$ and $|\cdot|$ indicate, respectively, the trace and the determinant of a matrix. Quantity KL measures how close the estimated $\widehat{Q}$ is to the true Q and lower values indicate a better estimate, with $\mathrm{KL} = 0$ if $\widehat{Q} = Q$. The sample estimates of KL, with the associate standard errors, are presented in Table 5. We emphasize that the estimators based on the boosting-type algorithm and the Hyvärinen's

Table 5 here

divergence exhibit a good performance. In particular, $\widehat{Q}_{H1}$ achieves definitely the best results, whereas the behaviour of $\widehat{Q}_{H2}$ is quite similar to that of the lasso-type estimator $\widehat{Q}_{GL}$ in all the experimental situations, excluding the case with $n = 100$. This can be explained by recalling that $\widehat{Q}_{H2}$ is defined without assuming a particular structure for the system of component precision matrices and then it corresponds to the most general and less powerful estimator of this family. Finally, the inverse of the sample covariance matrix performs, as expected, very poorly, in particular for $K = 20$ and $n = 100, 200$.

Moreover, we evaluate the observed proportion of false positives (FP), namely null elements in Q incorrectly estimated as non-zero, and of false negatives (FN), namely non-null elements in Q

incorrectly estimated as zero. We compute also the FP and the FN by defining a suitable threshold $\epsilon > 0$, so that the estimated off-diagonal entries of $\widehat{Q}$ belonging to $(-\epsilon, \epsilon)$ are considered as null. The results are given in Table 6, and we have to state that the comparison of the various estimated

Table 6 here

proportions of FP and FN, with or without the threshold $\epsilon = 0.02$, is not easy. We note that the graphical lasso estimator $\widehat{Q}_{GL}$ presents lower values for FP, excluding the case with $n = 500$. On the other hand, the estimators based on the boosting-type algorithm and the Hyvärinen's divergence show lower values for FN almost everywhere. With regard to the inverse of the sample covariance matrix, the values for FP and FN are 1.00 and 0.00, when we do not consider the threshold. This means that S^{-1} does not present null entries and then it is not suitable for estimating sparse matrices. Finally, we emphasize that, although the values for FP and FN are both desired to be small, the impact of a large proportion of FN can be more problematic, since many conditional dependence relations are missed and relevant information for the model specification are lost. On the contrary, a large proportion of FP means that unnecessary non-null entries are considered in the precision matrix, giving a more complicated model structure than required. Under this respect, the performance of $\widehat{Q}_{H1}$ and $\widehat{Q}_{H2}$, which present a low FN rate, is uniformly better than that of $\widehat{Q}_{GL}$.

Finally, we test the performance of the boosting-type estimator for the precision matrix, compared to that of the estimator based on the graphical lasso approach, using empirical data. More precisely, we consider the monthly returns of 48 US industries portfolios from August 2009 to July 2019 (120 observations for each portfolio); the dataset is provided by Kenneth French and it is publicly available on his website (http : //mba.tuck.dartmouth.edu/pages/faculty/ken.french/). The aim is to get a suitable estimate for the associated precision matrix having dimension 48×48, since this can be useful for defining optimal minimum risk strategies and for detecting the conditional dependence structure

among the portfolios returns.

The observations associated to the 48 US industries portfolios are centred and, as a working assumption, we assume that the data generating process follows a multivariate Gaussian distribution with null mean vector and unknown precision matrix. We apply both the lasso-type estimator and that one based on the boosting approach, considering the more general formulation where no specific structure for the component precision matrices is assumed. The inverse of the sample covariance matrix is also obtained. The first two estimates are shown in Figure 4, using a heatmap representation. We may conclude that both the estimators have been successful in detecting the sparsity of the precision matrix and they highlight almost the same conditional dependence structure. The only difference concerns the conditional correlation coefficients, which tend to be slightly underestimated by the glasso procedure. Thus, at least in this particular application, the boosting-type approach produces a lither regularization and this is in some sense confirmed by the fact that the KL distance between the boosting estimate and the inverse of the sample covariance matrix is considerably lower than that computed for glasso estimate.

Figure 4 here

References

[1] Aastveit, K. A., Ravazzolo, F. & van Dijk, H. K. (2018). Combined density nowcasting in an uncertain economic environment. *J. Bus. Econom. Statist.* **36**, 131-145.

[2] Allard, D., Comunian, A. & Renard, P. (2012). Probability aggregation methods in geoscience. *Math. Geosci.* **44**, 545-581.

[3] Bazaraa, M. S., Sherali, H. D. & Shetty, C. M. (2006). *Nonlinear programming*, 3rd edn. Wiley, Hoboken, New Jersey.

[4] Billio M., Casarin, R., Ravazzolo, F. & van Dijk, H. K. (2013). Time-varying combinations of predictive densities using nonlinear filtering. *J. Econometrics* **177**, 213-232.

[5] Boyd, S. & Vandenberghe, L. (2004). *Convex optimization*. Cambridge University Press, New York.

[6] Casarin, R., Mantovan, G. & Ravazzolo, F. (2016). Bayesian calibration of generalized pools of predictive distributions. *Econometrics* **4**, 1-24.

[7] Cox, D. R. & Wermuth, N. (1996). *Multivariate dependencies: models, analysis and interpretation*. Chapman & Hall, London.

[8] Dawid, A. P., Musio, M. & Ventura, L. (2016). Minimum scoring rule inference. *Scand. J. Stat.* **43**, 123-138.

[9] Dominitz, J. & Sherman, R. P. (2005). Convergence theory for iterative estimation procedures with an application to semiparametric estimation. *Econometric Theory* **21**, 838-863.

[10] Fan, J., Liao, Y. & Liu, H. (2016). An overview of the estimation of large covariance and precision matrices. *Econom. J.* **19**, C1-C32.

[11] Fan, Y., Pastorello, S. & Renault, E. (2015). Maximization by parts in extremum estimation. *Econom. J.* **18**, 147-171.

[12] Friedman, J. H. (2001). Greedy function approximation: a gradient boosting machine. *Ann. Statist.* **29**, 1189-1232.

[13] Friedman, J., Hastie, T. & Tibshirani, R. (2008). Sparse inverse covariance estimation with the graphical lasso. *Biostatistics* **9**, 432-441.

[14] Genest, C. (1984). A characterization theorem for externally Bayesian groups. *Ann. Statist.* **12**, 1100-1105.

[15] Geweke, J. & Amisano, G. (2011). Optimal prediction pools. *J. Econometrics* **164**, 130-141.

[16] Ghalanos, A. (2014). rugarch: univariate GARCH models. *R package version 1.4-0.*

[17] Hyvärinen, A. (2005). Estimation of non-normalized statistical models by score matching. *J. Mach. Learn. Res.* **6**, 695-709.

[18] Hyvärinen, A. (2007). Some extensions of score matching. *Comput. Statist. Data Anal.* **51**, 2499-2512.

[19] Kascha, C. & Ravazzolo, F. (2010). Combining inflation density forecasts. *J. Forecast.* **29**, 231-250.

[20] Kim, S., Pasupathy, R. & Henderson, S. G. (2015). A guide to sample average approximation. In *Handbook of simulation optimization* (ed M. C. Fu), 207-243. Springer, New York.

[21] Lindsay, B. G. (1988). Composite likelihood methods. *Contemp. Math.* **80**, 221-239

[22] Luenberger, D. G. & Ye, Y. (2008). *Linear and nonlinear programming*, 3rd edn. Springer, New York.

[23] Mazumder, R. & Hastie, T. (2012). The graphical lasso: new insights and alternatives. *Electron. J. Stat.* **6**, 2125-2149.

[24] Parry, M., Dawid, A. P. & Lauritzen, S. (2012). Proper local scoring rules. *Ann. Statist.* **40**, 561-592.

[25] Pauwels, L., Radchenko, P. & Vasnev, A. L. (2018). Higher moment constraints for predictive density combinations. *BA Working Paper*, BAWP-2019-01, University of Sidney.

[26] Rosset, S. & Segal, E. (2003). Boosting density estimation. In *Advances in neural information processing systems 15* (eds S. Becker, S. Thrun & K. Obermayer), 657-664. MIT Press.

[27] Rue, H. & Held, L. (2005). *Gaussian markov random fields*. Chapman & Hall, Boca Raton.

[28] Schapire, R. E. & Freund, Y. (2012). *Boosting. Foundations and algorithms*. MIT Press.

[29] Shapiro, A., Dentcheva, D. & Ruszczyński, A. (2009). *Lectures on stochastic programming: modeling and theory*. SIAM-MPS, Philadelphia.

[30] Varin, C., Reid, N. & Firth. D. (2011). An overview of composite likelihood methods. *Statist. Sinica* **21**, 5-42.

[31] Varin, C. & Vidoni, P. (2009). Pairwise likelihood inference for general state space models. *Econometric Rev.* **28**, 170-185.

[32] Welling, M., Zemel, R. S. & Hinton, G. E. (2003). Self supervised boosting. In *Advances in neural information processing systems 15* (eds S. Becker, S. Thrun & K. Obermayer), 681-688. MIT Press.

[33] Yu, M., Kolar, M. & Gupta, V. (2016). Statistical inference for pairwise graphical models using score matching. In *Advances in neural information processing systems 29* (eds D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon & R. Garnett), 2829-2837. Curran Associates Inc.

Acknowledgments

The author thanks the anonymous referees for suggestions and comments which served to improve the paper. The research was partially supported by the Italian Ministry for University and Research under the PRIN2015 Grant No. 2015EASZFS_003.

Corresponding author's address

Paolo Vidoni

Department of Economics and Statistics, University of Udine

via Tomadini 30/a, I-33100 Udine, Italy

E-mail: paolo.vidoni@uniud.it

A Appendix

A.1 The expected Hyvärinen score for pools of models

We assume that one of the models in $\mathcal{P}$ is the true one and we prove that this model minimizes the expected Hyvärinen score $S_H(f_p; w)$. Thus, under suitable regularity assumptions, the estimator given by (6) would attain the true model specification as $n \to +\infty$. We consider, with no loss of generality, that $f = p_1$, and we verify that the weights vector minimizing $S_H(f_p; w)$ is $w_0 = (1, 0, \ldots, 0)$.

Since the gradient vector, with respect to the interest parameter w, and the associated Hessian matrix of $S_H(f_p; w)$ are, respectively,

$$\nabla S_H(f_p; w) = 2E_f\left[\sum_{k=1}^{K} P_{kk}(Z)\right] + 2E_f\left[\mathbf{P}(Z)\mathbf{P}(Z)^T\right] w, \quad \nabla^2 S_H(f_p; w) = 2E_f\left[\mathbf{P}(Z)\mathbf{P}(Z)^T\right],$$

we conclude that

$$\nabla S_H(f_p; w_0) = 2E_f\left[\sum_{k=1}^{K} P_{kk}(Z)\right] + 2E_f\left[\mathbf{P}(Z)Q(Z)\right] = 0,$$

with $Q(z) = (\partial \log p_1(z)/\partial z_1, \ldots, \partial \log p_1(z)/\partial z_K)^T$. The last equality is easily obtained by considering that $f = p_1$ and that, following Hyvärinen (2005, pag. 707),

$$E_f\left[w^T P_{kk}(Z)\right] = -E_f\left[\partial \log f(z)/\partial z_k \, w^T P_k(Z)\right], \quad k = 1, \ldots, K,$$

for each $w \in \mathbb{R}^J_+$. The proof is completed by noticing that, if we exclude pathological situations, the Hessian matrix $\nabla^2 S_H(f_p; w)$ is positive definite.

A.2 Convergence of the boosting-type algorithm

The minimization problem (6) may be solved using several alternative optimization procedures. Note that this is not a completely unconstrained optimization problem, since the region Ω is a proper subset of $\mathbb{R}^J$. The solution takes the form of an iterative algorithm that generates a sequence of points in Ω, according to a prescribed rule until a termination criterion is verified. The focus here is on coordinate descent methods, and in particular on the well-known Gauss-Southwell procedure, where, at each step, the descent direction corresponds to a single component of the J-dimensional vector w. More precisely, the coordinate selected for descent corresponds to the largest, in absolute value, component of the gradient vector of the objective function (see, for example, Luenberger & Ye, 2008, Section 8.9).

The algorithms belonging to the class of coordinate descent methods are attractive because they are easy to implement, although their convergence properties are usually poorer than those of the procedures focusing on the steepest descent direction. In passing, we note that the Gauss-Southwell rule identifies the steepest descent, whenever the L_1-norm is taken into account (Boyd & Vandenberghe, 2004, Section 9.4). Nevertheless, for the optimization problem considered in this paper, this rule seems attractive since the objective function can be easily differentiated and, even if the dimension J of the domain set could be substantial, the stationary point is usually expected to have many null elements.

In order to discuss the convergence of the optimization algorithm, we emphasize that Ω is a convex set and that the objective function (4) is twice continuously differentiable. Indeed, it is a convex function and this can be easily proved, at least for the case with a univariate random variable Z, by verifying that it is convex whenever restricted to any line intersecting its domain (see, for example, Boyd & Vandenberghe, 2004, Section 3.1). Furthermore, it easy to see that strict convexity holds provided that we can find $i \in \{1, \ldots, n\}$ and $j \in \{1, \ldots, J\}$ such that $p'_j(z^{(i)}) \neq 0$. Thus, local minima are not present but this is not sufficient to conclude that a global minimum will be attained at a

finite value for w. The assumptions required for the convergence of the Gauss-Southwell method to a stationary point are specified, for example, in Bazaraa *et al.* (2006, Section 8.5).

Finally, we point out that the optimization problem (6) may be interpreted as a surrogate of the original one specified by considering the theoretical objective function $S_H(f_p; w)$ instead of its sample average estimate $\widehat{S}_H(f_p; w)$. The optimization problem (6) involves a deterministic objective function and it can be solved by applying the optimization algorithms mentioned before. This procedure, called sample average approximation in the stochastic programming framework, is effective when the sample objective function converges in some sense to the theoretical one, and both functions share the same structural properties, such as continuity and differentiability. Moreover, we expect that the optimizer of the sample average approximation problem converges to that of the true problem, as $n \to \infty$. A deep and extensive discussion on these relevant topics, from the optimization perspective, can be found, for example, in Shapiro *et al.* (2009, Chapter 5) and Kim *et al.* (2015).

A.3 Consistency of the estimators based on the backfitting procedure

In this subsection, we recall the assumptions required for the consistency of the estimators for θ defined by the algorithm presented in Section 4. To clarify the exposition, we make explicit the indication of the sample size n, so that $\widehat{\theta}_n^{(s)}$ is the estimator obtained from the s-th iteration of Step B.2(b) of the algorithm and $\widehat{\theta}_n$ is the estimator for θ as defined by (14). Let us consider the simplified situation where the multivariate function $\widehat{w}_n(\theta) = \widehat{w}_\theta = \text{argmin}_w \widehat{S}_H(f_p; \theta, w)$ is known and its partial derivatives, with respect to the components of vector θ, exist and are finite. Indeed, the score function defined in equation (16) is here rewritten adopting an alternative parametrization so that, with a slight abuse of notation, it corresponds to

$$\nabla_\theta S_n(\theta, \theta') = \nabla_\theta \widehat{S}_H(f_p; \theta, \widehat{w}_n(\theta')) + (\nabla_\theta \widehat{w}_n(\theta'))^T \nabla_w \widehat{S}_H(f_p; \theta', \widehat{w}_n(\theta')).$$

Thus, according to iterative algorithm, given $\widehat{\theta}_n^{(s-1)}$, the next value $\widehat{\theta}_n^{(s)}$ is obtained as the solution, with respect to θ, of the score-type equation $\nabla_\theta S_n(\theta, \widehat{\theta}_n^{(s-1)}) = 0$. If there exists a function $\bar{\theta}_n(\cdot)$ such that $\widehat{\theta}_n^{(s)} = \bar{\theta}_n(\widehat{\theta}_n^{(s-1)})$, for $s \geq 1$, and it is an asymptotic contraction mapping, then it admits a fixed point corresponding to the estimator $\widehat{\theta}_n$ and the sequence $\widehat{\theta}_n^{(s)}$ converges to $\widehat{\theta}_n$ as $s \to +\infty$ (see Dominitz & Sherman, 2005).

In order to prove the consistency of $\widehat{\theta}_n^{(s)}$, when $n \to +\infty$, as an estimator for the true unknown value θ_0, the following assumptions have to be considered. First, as usual, the set Θ is supposed to be a compact subset of $\mathbb{R}^d$ and, for any $n \geq 1$, $\nabla_\theta S_n(\theta, \theta')$ is a measurable function of the observations $z^{(1)}, \ldots, z^{(n)}$ and it is continuous, as a function of θ and θ'. Second, there exists a limit differentiable function $w(\theta)$ and a limit function $\nabla_\theta S(\theta, \theta')$ such that $\sup_\theta |\widehat{w}_n(\theta) - w(\theta)| = o_p(1)$ and $\sup_{\theta,\theta'} |\nabla_\theta S_n(\theta, \theta') - \nabla_\theta S(\theta, \theta')| = o_p(1)$, as $n \to +\infty$. Finally, for any fixed $\theta' \in \Theta$, the function $|\nabla_\theta S(\theta, \theta')|$ admits a unique minimizer $\bar{\theta}(\theta')$, with respect to θ, and the map $\bar{\theta}(\cdot)$ is continuous on Θ, admitting θ_0 as a fixed point.

Under the above mentioned assumptions, if the starting point $\widehat{\theta}_n^{(0)}$ of the algorithm is a weakly consistent estimator of θ_0, then $\widehat{\theta}_n^{(s)}$ is weakly consistent too, for each $s \geq 1$. On the other hand, if $\widehat{\theta}_n^{(0)}$ is not consistent, $\widehat{\theta}_n^{(s)}$ is weakly consistent only if the number s of iterations tends to infinity with the sample size n, provided that $\bar{\theta}(\cdot)$ is a contracting map on Θ. An additional, in-depth analysis of the asymptotic properties of the estimators may be found in Fan *et al.* (2015) and references therein.

Table 1: Optimal weights given by the boosting algorithm (values $w_j = 0$, $j = 4, \ldots, 10$, are not reported in the table). Simulated samples of dimension $m = 100, 200, 500$ from a first-order autoregressive model plus additive observation noise, with $\beta = 0.2$, $\sigma = 1$, $\tau = 1$, (a) $\gamma = 0.5$, (b) $\gamma = 0.85$ and (c) $\gamma = 0.95$.

	(a)			(b)			(c)		
	w_1	w_2	w_3	w_1	w_2	w_3	w_1	w_2	w_3
$m = 100$	0.836	0	0.135	0.678	0.404	0	0.742	0.191	0.065
$m = 200$	0.795	0.252	0	0.746	0.374	0	0.707	0.361	0
$m = 500$	0.869	0.136	0.009	0.721	0.348	0.016	0.773	0.260	0.040

Table 2: Sample means and standard errors (in brackets) for the maximum likelihood estimator ($\widehat{\theta}_{ML}$), the estimator based on the Hyvärinen score function ($\widehat{\theta}_H$) and the estimators based on the conditional pairwise likelihood with a maximum lag distance $J = 3$ and optimal weights ($\widehat{\theta}_{PW1}$) or equal weights ($\widehat{\theta}_{PW2}$). Simulated samples of dimension $m = 250, 500$ from a first-order autoregressive model plus additive observation noise, with $\beta = 0.2$, $\sigma = 1$, $\tau = 1$, (a) $\gamma = 0.5$, (b) $\gamma = 0.75$.

(a)

		$m = 250$				$m = 500$			
	True	$\widehat{\theta}_{ML}$	$\widehat{\theta}_H$	$\widehat{\theta}_{PW1}$	$\widehat{\theta}_{PW2}$	$\widehat{\theta}_{ML}$	$\widehat{\theta}_H$	$\widehat{\theta}_{PW1}$	$\widehat{\theta}_{PW2}$
β	0.2	0.201	0.201	0.201	0.201	0.194	0.194	0.194	0.194
		(0.140)	(0.140)	(0.140)	(0.140)	(0.099)	(0.099)	(0.099)	(0.099)
σ	1	0.993	0.992	1.009	0.813	0.997	0.996	1.012	0.812
		(0.088)	(0.094)	(0.097)	(0.168)	(0.061)	(0.062)	(0.058)	(0.202)
γ	0.5	0.477	0.479	0.497	0.652	0.487	0.488	0.504	0.663
		(0.088)	(0.094)	(0.097)	(0.168)	(0.059)	(0.063)	(0.067)	(0.169)

(b)

		$m = 250$				$m = 500$			
	True	$\widehat{\theta}_{ML}$	$\widehat{\theta}_H$	$\widehat{\theta}_{PW1}$	$\widehat{\theta}_{PW2}$	$\widehat{\theta}_{ML}$	$\widehat{\theta}_H$	$\widehat{\theta}_{PW1}$	$\widehat{\theta}_{PW2}$
β	0.2	0.209	0.209	0.209	0.208	0.193	0.193	0.193	0.193
		(0.248)	(0.252)	(0.252)	(0.252)	(0.181)	(0.182)	(0.182)	(0.182)
σ	1	0.998	1.010	1.031	0.721	0.998	1.007	1.024	0.717
		(0.080)	(0.078)	(0.075)	(0.296)	(0.056)	(0.055)	(0.055)	(0.291)
γ	0.75	0.733	0.764	0.784	0.832	0.740	0.774	0.793	0.837
		(0.055)	(0.071)	(0.074)	(0.089)	(0.038)	(0.059)	(0.064)	(0.090)

Table 3: Sample Hyvärinen score and sample log score for the six alternative models.

Sample Hyvärinen score					
Gaussian	Student	GARCH	t-GARCH	EGARCH	t-EGARCH
-0.985	-1.465	-1.461	-1.705	-1.417	-1.714
Sample log score					
Gaussian	Student	GARCH	t-GARCH	EGARCH	t-EGARCH
-10479.84	-9749.63	-9607.19	-9320.03	-9578.09	-9297.55

Table 4: Weights for the optimal multiplicative pool and the associated score, according to sample Hyvärinen score, and weights for the optimal linear pool and the associated score, according to the sample log score. The X symbol indicates that the model is not considered in the pool. The last row of the two parts shows the mean value of the weights reoptimized at each day (excluding the first 1000 values) and the associated score.

Weights and sample Hyvärinen score for multiplicative pools						
Gaussian	Student	GARCH	t-GARCH	EGARCH	t-EGARCH	score
0.000	0.000	0.000	0.411	0.000	0.568	-1.730
0.000	0.060	0.054	X	0.000	0.863	-1.720
0.000	0.008	0.000	0.928	0.061	X	-1.706
Sample Hyvärinen score with weights reoptimized						
0.000	0.116	0.000	0.637	0.000	0.253	-1.776
Weights and sample log score for linear pools						
Gaussian	Student	GARCH	t-GARCH	EGARCH	t-EGARCH	score
0.010	0.000	0.007	0.328	0.055	0.599	-9280.04
0.000	0.035	0.173	X	0.000	0.792	-9284.17
0.000	0.000	0.000	0.671	0.329	X	-9297.65
Sample log score with weights reoptimized						
0.009	0.031	0.002	0.569	0.013	0.376	-9278.44

Table 5: Estimated Kullback-Leibler loss and standard errors (in brackets) for the boosting-type estimator with band component precision matrices $\widehat{Q}_{H1}$, the boosting-type estimator based on general precision matrices $\widehat{Q}_{H2}$, the graphical lasso estimator $\widehat{Q}_{GL}$ and the inverse of the sample covariance matrix S^{-1}. Simulated samples of dimension $m = 100, 200, 500$ from a K-variate Gaussian distribution with $K = 10, 20$, having a null mean vector and a band precision matrix.

	$K = 10$				$K = 20$			
n	$\widehat{Q}_{H1}$	$\widehat{Q}_{H2}$	$\widehat{Q}_{GL}$	S^{-1}	$\widehat{Q}_{H1}$	$\widehat{Q}_{H2}$	$\widehat{Q}_{GL}$	S^{-1}
100	0.194	0.566	0.495	0.657	0.405	1.775	1.303	3.067
	(0.006)	(0.012)	(0.008)	(0.015)	(0.009)	(0.025)	(0.010)	(0.043)
200	0.096	0.268	0.274	0.303	0.228	0.791	0.837	1.238
	(0.003)	(0.005)	(0.005)	(0.006)	(0.006)	(0.010)	(0.008)	(0.014)
500	0.038	0.106	0.107	0.113	0.123	0.363	0.360	0.447
	(0.001)	(0.002)	(0.002)	(0.002)	(0.003)	(0.004)	(0.004)	(0.005)

Table 6: Observed proportion of false positives (FP) and false negatives (FN) associated to the estimated precision matrices $\widehat{Q}_{H1}$, $\widehat{Q}_{H2}$, $\widehat{Q}_{GL}$ and S^{-1}. In the second line, a threshold $\epsilon = 0.02$ is considered. Simulated samples of dimension $m = 100, 200, 500$ from a K-variate Gaussian distribution with $K = 10, 20$, having a null mean vector and a band precision matrix.

	$K = 10$				$K = 20$			
n	$\widehat{Q}_{H1}$	$\widehat{Q}_{H2}$	$\widehat{Q}_{GL}$	S^{-1}	$\widehat{Q}_{H1}$	$\widehat{Q}_{H2}$	$\widehat{Q}_{GL}$	S^{-1}
	FP FN	FP FN	FP FN	FP FN	FP FN	FP FN	FP FN	FP FN
100	0.94 0.00	0.84 0.04	0.51 0.24	1.00 0.00	0.86 0.00	0.66 0.12	0.27 0.44	1.00 0.00
	0.79 0.00	0.77 0.05	0.43 0.28	0.92 0.02	0.62 0.00	0.59 0.14	0.22 0.48	0.94 0.02
200	0.90 0.00	0.79 0.03	0.73 0.06	1.00 0.00	0.83 0.00	0.55 0.08	0.48 0.16	1.00 0.00
	0.70 0.00	0.68 0.05	0.62 0.09	0.90 0.01	0.48 0.00	0.45 0.10	0.38 0.20	0.90 0.01
500	0.81 0.00	0.64 0.01	0.83 0.00	1.00 0.00	0.71 0.00	0.38 0.04	0.68 0.16	1.00 0.00
	0.49 0.00	0.48 0.02	0.65 0.01	0.81 0.00	0.26 0.00	0.26 0.05	0.51 0.03	0.83 0.00

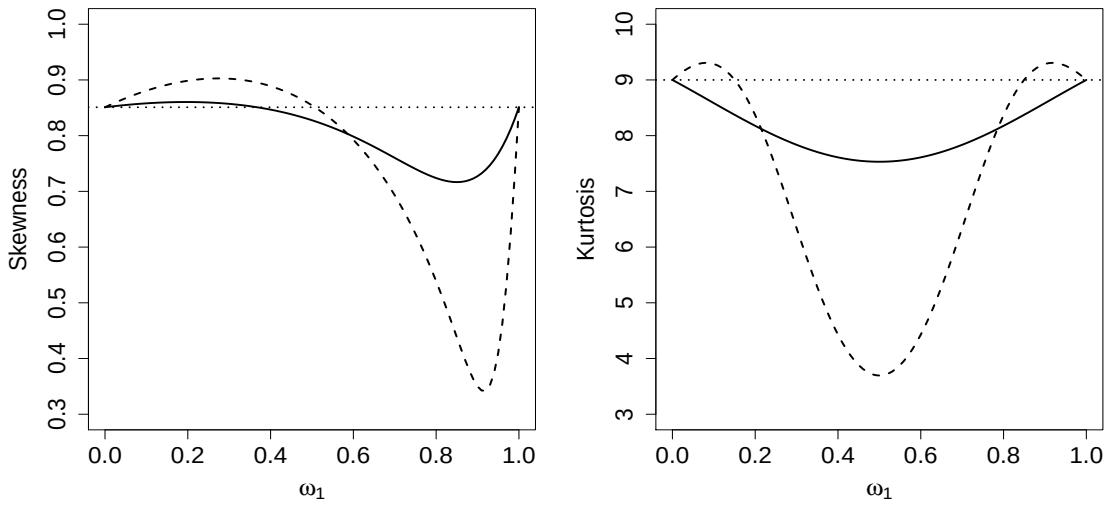

Figure 1: Left: skewness as a function of w_1 for a multiplicative combination of two skew-normal distributions with the same skewness 0.85 (dotted line) and location parameters 0.1 and 1 (solid line), -1 and 1 (dashed line). Right: kurtosis as a function of w_1 for a multiplicative combination of two Student t distributions with the same kurtosis 9 (dotted line) and location parameters -1 and 1 (solid line), -5 and 1 (dashed line)

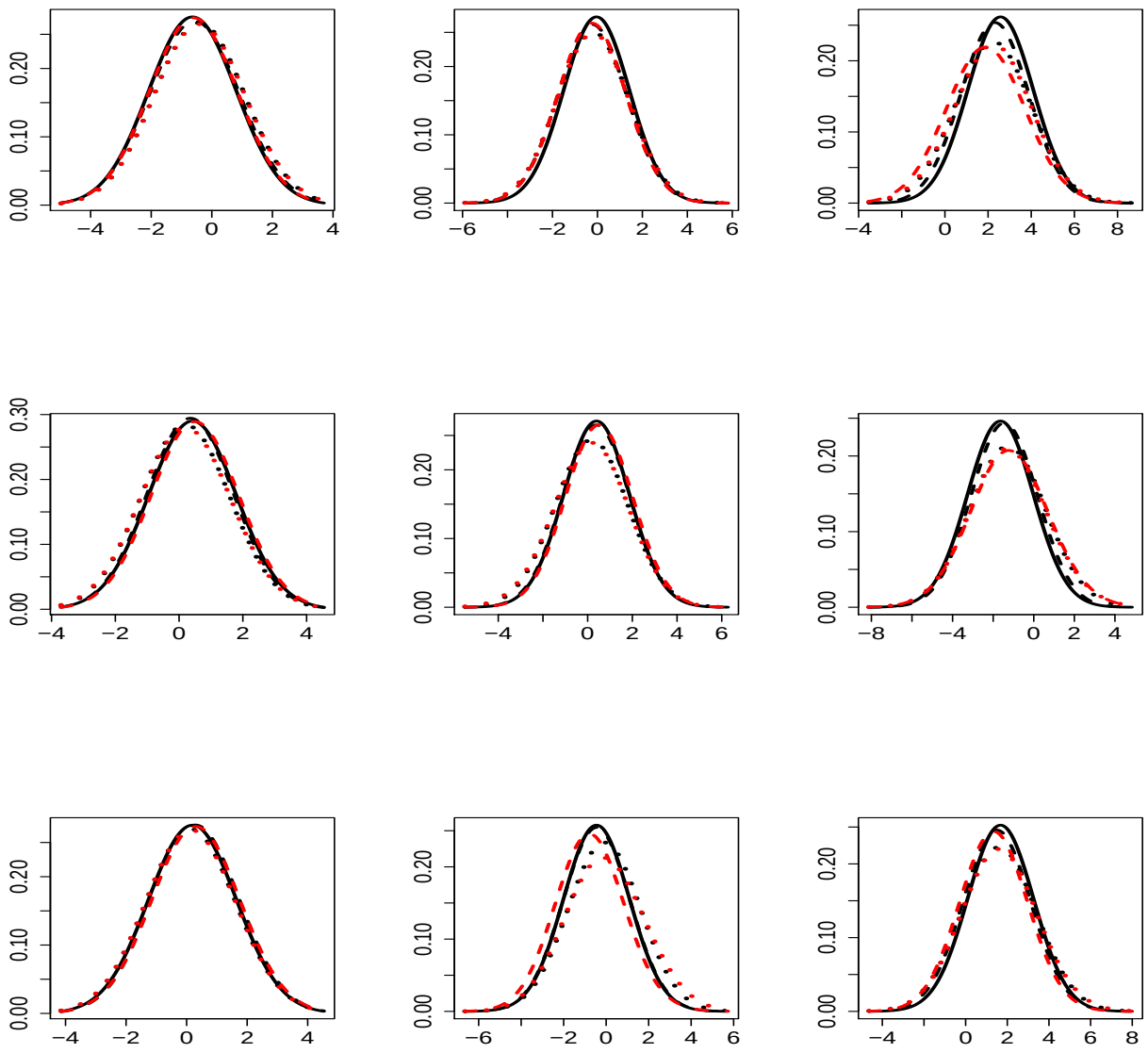

Figure 2: Target conditional density (solid line), multiplicative mixture density with the optimal weights given in Table 1 (dashed line) and equal weights $w_1 = w_2 = w_3 = 1/3$, with $J = 3$ (dotted line), additive mixture density with optimal weights (red dashed line) and equal weights $w_1 = w_2 = w_3 = 1/3$, with $J = 3$ (red dotted line). Simulated samples of dimension $m = 100$ (top line), $m = 200$ (central line), $m = 500$ (bottom line) from a first-order autoregressive model plus additive observation noise, with $\beta = 0.2$, $\sigma = 1$, $\tau = 1$, (a) $\gamma = 0.5$ (left), (b) $\gamma = 0.85$ (center) and (c) $\gamma = 0.95$ (right).

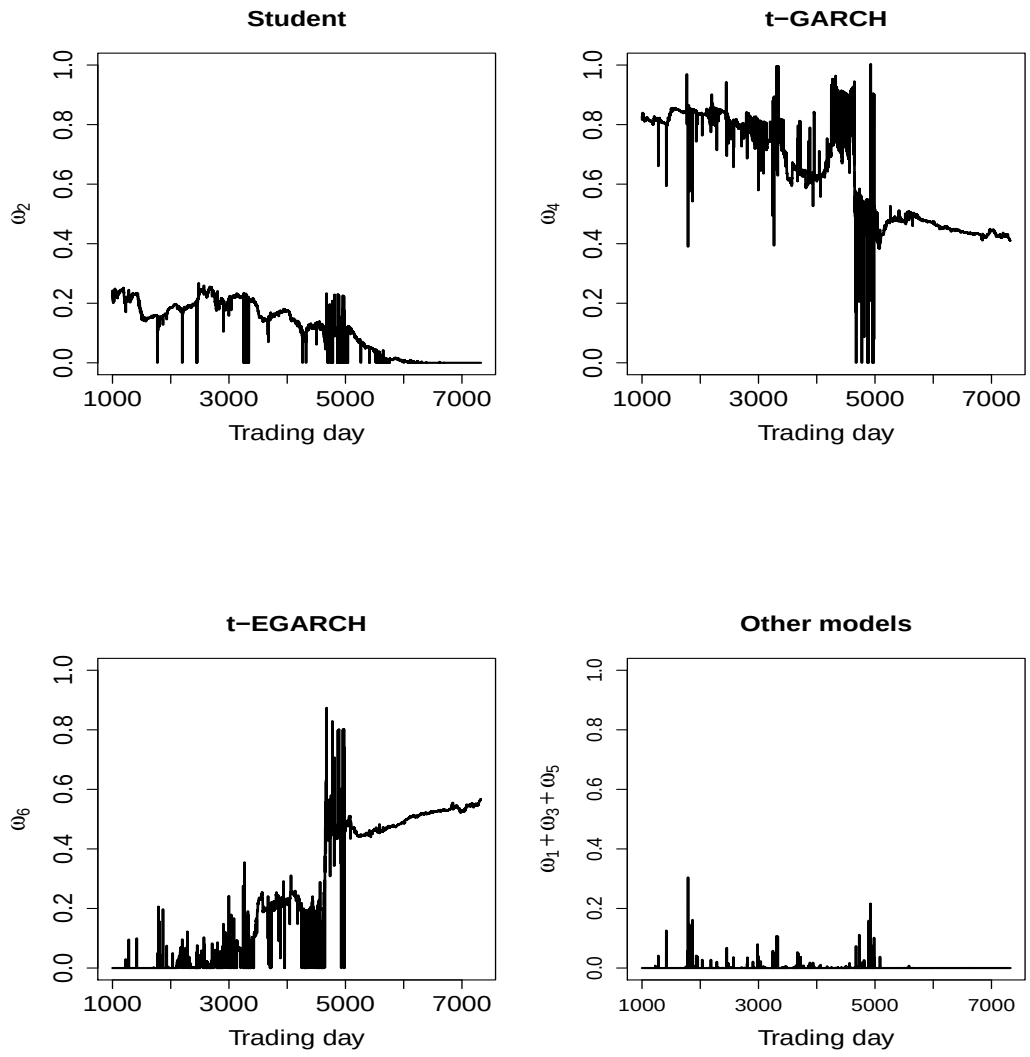

Figure 3: Temporal evolution of the optimal weights for the multiplicative pool according to the sample Hyvärinen score (excluding the first 1000 values); the models with the highest values are considered, namely Student (ω_2), t-GARCH (ω_4) and t-EGARCH (ω_6), together with the sum of the weights of the remaining models ($\omega_1 + \omega_3 + \omega_5$).

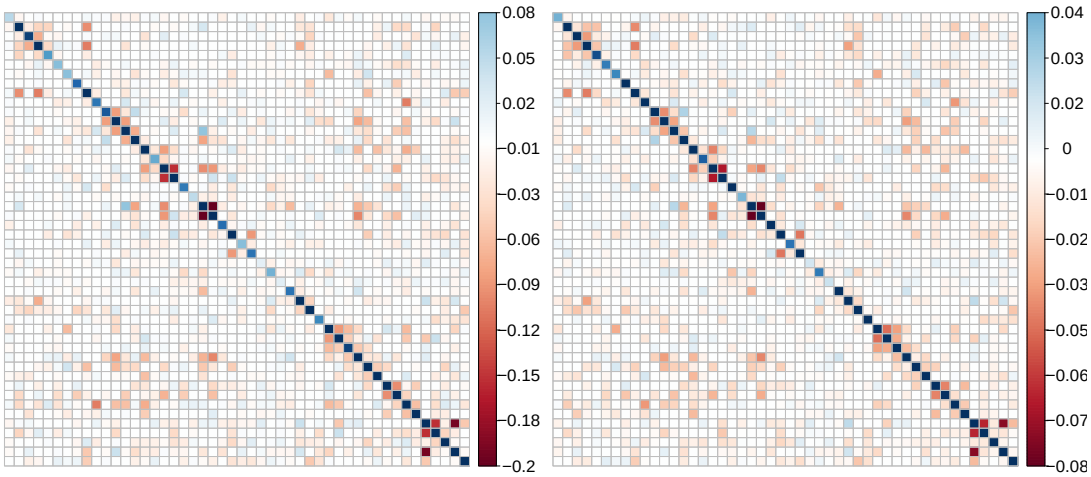

Figure 4: Heatmap of the estimated precision matrix based on the boosting procedure (left) and on the lasso-type approach (right); dataset with the monthly returns of 48 US industries portfolios from August 2009 to July 2019.