Structured Dictionary Learning of Rating Migration Matrices for Credit Risk Modeling

Michaël Allouche, Emmanuel Gobet, Clara Lage, Edwin Mangin

▶ **To cite this version:**

Michaël Allouche, Emmanuel Gobet, Clara Lage, Edwin Mangin. Structured Dictionary Learning of Rating Migration Matrices for Credit Risk Modeling. 2022. hal-03715954

HAL Id: hal-03715954
https://hal.archives-ouvertes.fr/hal-03715954

Preprint submitted on 7 Jul 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Structured Dictionary Learning of Rating Migration Matrices for Credit Risk Modeling*

Michaël Allouche† Emmanuel Gobet‡ Clara Lage§ Edwin Mangin¶

July 7, 2022

Abstract

Rating Migration Matrix is a crux to assess credit risks. Modeling and predicting these matrices are then an issue of great importance for risk managers in any financial institution. As a challenger to usual parametric modeling approaches, we propose a new structured dictionary learning model with auto-regressive regularization that is able to meet key expectations and constraints: small amount of data, fast evolution in time of these matrices, economic interpretability of the calibrated model. To show the model applicability, we present a numerical test with real data. The source code and the data are available at https://github.com/michael-allouche/dictionary-learning-RMM.git for the sake of reproducibility of our research.

KEYWORDS: Rating Migration Matrix, Dictionary learning, auto-regressive modeling, interpretability

1 Introduction

1.1 Banking context

Credit risk refers to the risk of incurring losses due to unexpected changes in the credit quality of the counterparty. Such a risk is summarized in a structured rating migration matrix which captures all possible transition probabilities that an obligor will migrate from a credit state to another over a given time period (see Figure 1). According to the financial regulation guidelines (Basel II and III), banks can use internal ratings and risk exposure estimations in order to assess regulatory capital requirement and credit risk measures (VaR, ES, ...). See [1, 7, 21, 2] for extensive references on risk measures and credit risk. Rating migration matrices (RMM) are key indicators to assess credit risk portfolio through the estimation of the credit quality of the obligors. Rating allocation process includes models and expert systems taking into account obligors idiosyncratic features evolving over time given the economic situation. Observed migration frequencies are displayed into RMM that are the cornerstone of rating migration models upon which credit risk portfolio simulation relies. The most widely used method for modeling RMM is the one factor Gaussian copula model [13] which assumes that a single factor represents the underlying systemic credit quality in the economy and defines a stationary economic cycle. See [3] among others for estimating risk measures on the loss distribution of a large credit risk portfolio under this model. The popularity of the Gaussian copula model is due to the ease of use but it also suffers from too simple underlying hypothesis. These weak assumptions lead to miscapture the dependence structure in tails. However, and despite post subprime crisis criticisms [14], the one factor Gaussian copula model remains very popular in the banking industry because of its parsimony and of its ability to generate intuitive and interpretable results. The aim of this work is to derive from the data a non parametric representation of RMM, as an alternative and a challenger to the parametric Gaussian copula model. This work is devoted to the design of a new methodology with thorough tests. Full comparison with Gaussian copula model will be handled in a subsequent work.

1.2 Matrix Factorization for RMM

Let us start from the data. In practice, we observe at time t a one-year rating migration matrix $\mathbf{P}^t \in \mathbb{R}^{R-1} \otimes \mathbb{R}^R$, which encodes the probability of migrating from rating $i = 1, \dots, R-1$ to rating $j = 1, \dots, R$ within one year period starting at time $t-1$; in Figure 1 we have $R = 11$. The reconstruction of this matrix is made empirically by evaluating the frequencies of obligors going from the rating

*This action benefited from the support of the Chair Stress Test, Risk Management and Financial Steering, led by the French Ecole polytechnique and its foundation and sponsored by BNP Paribas.

†CMAP, CNRS, Ecole Polytechnique, Institut Polytechnique de Paris, Route de Saclay, Palaiseau, France. michael.allouche@polytechnique.edu

‡CMAP, CNRS, Ecole Polytechnique, Institut Polytechnique de Paris, Route de Saclay, Palaiseau, France. emmanuel.gobet@polytechnique.edu

§CMAP, CNRS, Ecole Polytechnique, Institut Polytechnique de Paris, Route de Saclay, Palaiseau, France. clara.lage@polytechnique.edu

¶BNP Paribas, 16 boulevard des Italiens, Paris, France. edwin.mangin@bnpparibas.com

	1	2	3	4	5	6	7	8	9	10	11
1	71.48	17.87	5.36	2.38	1.25	0.71	0.43	0.26	0.15	0.08	0.03
2	16.01	57.62	14.41	5.69	2.80	1.54	0.89	0.52	0.30	0.16	0.06
3	5.05	15.14	53.85	13.46	5.89	3.03	1.68	0.96	0.54	0.28	0.11
4	2.45	6.52	14.67	51.35	12.84	5.87	3.06	1.68	0.92	0.47	0.18
5	1.45	3.61	7.23	14.46	49.57	12.39	5.74	2.95	1.55	0.76	0.30
6	0.96	2.31	4.34	7.71	14.46	48.19	12.05	5.51	2.71	1.29	0.48
7	0.70	1.63	2.94	4.90	8.16	14.69	47.00	11.75	5.14	2.28	0.82
8	0.54	1.24	2.18	3.48	5.44	8.71	15.24	45.71	11.43	4.51	1.52
9	0.45	1.02	1.75	2.73	4.09	6.14	9.55	16.38	43.67	10.92	3.28
10	0.43	0.95	1.60	2.43	3.55	5.11	7.46	11.36	19.17	38.35	9.59

Figure 1: Representation of an idealized rating migration matrix of size 10×11. All values are in percentage. The credit quality goes from the highest (rating 1) to the lowest (rating 10), the default is 11.

i to rating j between times $t-1$ and t. It is important for risk management purposes to model the evolution of $\mathbf{P}^t$, by finding a representation of the type

$$\mathbf{P}^t \approx \sum_{k=1}^{K} \alpha_k^t \mathbf{d}_k, \quad \forall t \geq 1, \tag{1}$$

for some so-called (deterministic) basis vectors $\mathbf{d}_k$ and for some scalar random coefficients α_k^t which we should model the evolution. In a matrix form (using the vec() operator to simplify, see Section 1.5), we say that the collection of vectorized matrices $\mathbf{P} = \{\mathrm{vec}(\mathbf{P}^t) \in \mathbb{R}^d\}_{t=1}^T \in \mathbb{R}^d \otimes \mathbb{R}^T$, with

$$d := (R-1)R$$

for all $t = 1, \ldots, T$, admits a matrix factorization over a dictionary $\mathbf{D} \in \mathbb{R}^d \otimes \mathbb{R}^K$ composed by K elements (called atoms), if there exists a linear combination of atoms weighted by coefficients (called codings) $\mathbf{A} = \{\boldsymbol{\alpha}^t \in \mathbb{R}^K\}_{t=1}^T \in \mathbb{R}^K \otimes \mathbb{R}^T$ such that

$$\mathbf{P} \approx \mathbf{DA}. \tag{2}$$

1.3 Objective

In this work, the objective is to achieve (2) while requiring

- $\mathbf{D}$ to satisfy some linear constraints (see Section 1.5) in order to represent economically interpretable RMM,
- the time series of elements $\boldsymbol{\alpha}^t$ of $\mathbf{A}$ to be smooth enough in order to perform predictions through a time series modeling,
- consider a dimensionality reduction framework $K \ll d$ in order to work in a lower dimensional space with extracted meaningful information.

However, the RMM evolution may vary quickly over time and a limited data history is available (usually 10-20 years ≈ 200 observations) which is close to the dimension of the problem (usually $R = 11$ and $d = 110$). Thus, modeling constrained RMM in a data-based non-parametric way presents an important challenge, which has not been addressed so far to the best of our knowledge.

1.4 State of the art

Over the last years, a new paradigm of data-based models have emerged in the Machine Learning (ML) community in order to extract structured information from high-dimensional objects. A classical approach in ML is to use Matrix Factorization techniques in order to project the data in some relevant basis. It is well known that the optimal basis that minimizes the linear approximation error is the Karhunen-Loève basis [20, Theorem 9.8], also known as the principal components in the principal component analysis (PCA).

Introduced in [24], dictionary learning (DL), see [9] for an overview and [12] for theoretical results, is another matrix representation technique where the basis, called dictionary, is learned from the observations. Unlike in the PCA decomposition, neither the orthogonality nor the representation constraints of the basis vectors (atoms) are imposed, allowing more flexibility to adapt the desired representation to the data. Moreover, compared with a predefined dictionary like Gabor functions, wavelets or local cosine vectors [20], learning a dictionary adapted to the observations has shown better results in practice [10, 17].

In DL, the linear approximation (2) is usually coupled with a regularization criterion $\mathcal{R}(\mathbf{A})$ applied to the codings and yields to the general optimization problem

$$\min_{\mathbf{D},\mathbf{A}} \|\mathbf{P} - \mathbf{DA}\|_F^2 + \lambda \mathcal{R}(\mathbf{A}), \quad \lambda \geq 0, \tag{3}$$

where the regularization term shall reflect the expected codings representation, see [9, Chapter 4]. The most widely studied regularization is $\mathcal{R}(\mathbf{A}) = \|A\|_1$ referring to the so-called sparse coding (see [15] for an overview), where the optimization with respect to A is known as basis pursuit [6] or the Lasso [29]. DL with sparse representation was notably studied in image and video processing [16, 18, 19], in graph learning [30] and in clustering [28]. In the case of spatial data, and more precisely in image processing, Total Variation (TV) plays an important role. In one dimension we have $\mathcal{R}_{TV}(\mathbf{A}) = \sum_{t=1}^{T-1} \sum_{k=1}^{K} |\alpha_k^{t+1} - \alpha_k^t|$, which is the integral of the absolute value of the gradient [26]. The intuition of this type of regularization in images is to allow a smooth transition between close codings and can be understood as a prior in a Bayesian model, see [5].

Here, we rather focus on DL with a temporal structure. This application has been mainly studied in video denoising

[19, 25] where the temporal structure is exploited through an operator extracting patches of a fixed size in the objective function representing an energy minimization procedure. Another approach is to deal with an auto-regressive (AR) representation modeled either in the dictionary [8] or in the codings [32]. In the former, a mixed audio signal is decomposed into its constituent temporal sources (atoms of the dictionary) in order to detect the presence of a specific sound. In the latter, the authors present a framework which supports data-driven temporal learning and forecasting through an AR modelization of the codings represented as a regularization term. Our model described in Section 2 is inspired from this problem formulation.

Our main and original contributions are to

- propose a new RMM modelization technique using DL approach
- derive a DL solution with linear constraints and a temporal regularization term for both interpretable clustering and prediction of RMM
- retrieve an economic health indicator on real data.

The paper is organized as follows. We introduce our proposed model and the associated optimization procedure in Section 2. Then we provide a numerical study on real data in Section 3 with two applications: prediction and clustering of RMM with economic interpretations.

1.5 Notations and data constraints

Notations.

Let $\mathbf{M} \in \mathbb{R}^{d_1} \otimes \mathbb{R}^{d_2}$ be a matrix with d_1 rows and d_2 columns. We use the notation for the i-th row $\mathbf{M}_{i,:}$, for the j-th column $\mathbf{M}_{:,j} := \mathbf{m}_j$ and for the sum column-wise

$$M_{i,\geq j} := M_{i,j} + \cdots + M_{i,d_2}.$$

The vectorization operator $\mathrm{vec}\,(\cdot)$ and its inverse $\mathrm{vec}^{-1}(\cdot)$ are defined as column-major order, *i.e.*

$$\begin{aligned}\mathrm{vec}^{-1}\mathrm{vec}\,(\mathbf{M}) &:= \mathrm{vec}^{-1}\left([M_{1,1}, M_{2,1}, \ldots, M_{d_1,d_2}]^\top \in \mathbb{R}^{d_1 d_2}\right)\\ &= \mathbf{M}.\end{aligned}$$

The Frobenius norm is defined by:

$$\|\mathbf{M}\|_F = \sqrt{\mathrm{Tr}(\mathbf{M}\mathbf{M}^\top)} = \sqrt{\sum_{i\in[d_1], j\in[d_2]} M_{i,j}^2},$$

where $[d_i] := \{1, ..., d_i, \}$ for $i \in \{1, 2\}$. The orthogonal linear projection of a vector $\mathbf{u} \in \mathbb{R}^u$ onto the space generated by the columns of a matrix $\mathbf{V} \in \mathbb{R}^{v_1} \otimes \mathbb{R}^{v_2}$, with $u = v_1$, is denoted by $\mathrm{Proj}_{\mathbf{V}}(\mathbf{u})$. For a matrix $\mathbf{U} \in \mathbb{R}^{u_1} \otimes \mathbb{R}^{u_2}$, $(u_1 = v_1)$ with columns $\{\mathbf{U}_{:,1}, ..., \mathbf{U}_{:,u_2}\}$, the projection is defined by

$$\begin{aligned}&\mathrm{Proj}_{\mathbf{V}}(\mathbf{U}) \in \mathbb{R}^{v_2} \otimes \mathbb{R}^{u_2},\\ &\quad \text{with columns } \{\mathrm{Proj}_{\mathbf{V}}(\mathbf{U}_{:,1}), ..., \mathrm{Proj}_{\mathbf{V}}(\mathbf{U}_{:,u_2})\}.\end{aligned}$$

Data constraints

A rating migration matrix $\mathbf{M}$ must satisfy some constraints for both mathematical and economical reasons.

Mathematical constraints. Each row of $\mathbf{M}$ is a discrete probability, hence $\mathbf{M}$ is a stochastic matrix. The set of Stochastic Matrices is denoted by

$$\begin{aligned}\mathcal{M}^S := \Big\{\mathbf{M} \in \mathbb{R}^{(R-1)} \otimes \mathbb{R}^R : \sum_{j\in[R]} M_{i,j} = 1, \forall i \in [R-1],\\ M_{i,j} \geq 0,\ \forall (i,j) \in [R-1] \times [R]\Big\}.\end{aligned}$$

Economic constraints. Depending on their expertise, some risk managers may consider important to put additional constraints that are meaningful from economic point-of-view. For instance, the likelihood of default for higher-rated counterparties is lower than for the lower-quality ones. Then, the collection of rating matrices satisfying so-called economic constraints is denoted by

$$\begin{aligned}\mathcal{M}^E := \Big\{\mathbf{M} \in \mathbb{R}^{(R-1)} \otimes \mathbb{R}^R : M_{i,\geq j} \leq M_{i',\geq j},\\ \forall j \in [R], \quad 1 \leq i < i' \leq R-1\Big\}.\end{aligned}$$

A matrix satisfying such constraints is called an idealized matrix and is illustrated in Figure 1.

2 Dictionary learning: modeling and solving

2.1 Defining the regularization term

In the case of time-series DL, the expected time dependency can be encoded in the regularization part. We propose a regularization term with an extra parameter $\mathbf{w}$ that will be used to infer the behavior of the codings as a time-series.

Defining

$$\bar{\alpha}_k := \frac{1}{T}\sum_{t=1}^{T} \alpha_k^t,$$

the proposed DL problem is:

$$\min_{\substack{\mathbf{D},\mathbf{A},\mathbf{w}\\ \mathbf{D}\in\Omega,\ \alpha_k^t\geq 0,\ t\in[T],\ k\in[K]}} \|\mathbf{P} - \mathbf{D}\mathbf{A}\|_F^2 + \lambda \mathcal{R}_{AR}(\mathbf{A}, \mathbf{w}) \quad (4)$$

with regularization:

$$\mathcal{R}_{AR}(\mathbf{A}, \mathbf{w}) := \sum_{k=1}^{K}\sum_{t=1}^{T-1}\left(\alpha_k^{t+1} - \bar{\alpha}_k - w_k(\alpha_k^t - \bar{\alpha}_k)\right)^2, \quad (5)$$

where the extra parameter $\mathbf{w}$ allows us to estimate the AR parameters of the time-series $\boldsymbol{\alpha}_k$ for each $k \in [K]$, as it will

be detailed below. The available set Ω is the convex set of dictionaries verifying the idealized constraints (see Section 1.5)

$$\Omega := \left\{ \mathbf{D} \in \mathbb{R}^d \otimes \mathbb{R}^K : \mathrm{vec}^{-1}(\mathbf{d}_k) \in \mathcal{M}^E \cap \mathcal{M}^S, \ \forall k \in [K] \right\}.$$

Heuristics for the regularization strategy The AR model is an important time-series structure, largely applied in finance and other contexts (see [22]). For a fixed $k \in [K]$, we say that the time-series $\boldsymbol{\alpha}_k$ is auto-regressive of order 1, if

$$\alpha_k^{t+1} = \mu_k + w_k \alpha_k^t + \epsilon_k^t, \quad \text{for all } t > 1, \tag{6}$$

where μ_k is a constant called drift, w_k is the AR coefficient and $(\epsilon_k^t)_{t=1}^T$ are independent centered Gaussian variables with some variance parameter σ_k^2.

Starting from the DL model (3), we encourage the codings $\mathbf{A}$ to have an AR structure (6) through the regularization term. Thus, assuming an AR structure of $\boldsymbol{\alpha}_k$ for each $k \in [K]$, the log-likelihood with respect to parameters μ_k, w_k, and σ_k, up to a constant term, is:

$$\ell(\boldsymbol{\alpha}_k, \mu_k, w_k, \sigma_k) := -\frac{1}{2\sigma_k^2} \sum_{t=1}^{T-1} \left(\alpha_k^{t+1} - \mu_k - w_k \alpha_k^t \right)^2 - (T-1)\log(\sigma_k)$$

(see [27, Chapter 3.6]), with solutions:

$$\tilde{\mu}_k, \tilde{w}_k, \tilde{\sigma}_k = \underset{\mu_k, w_k, \sigma_k}{\arg\max}\, \ell(\boldsymbol{\alpha}_k, \mu_k, w_k, \sigma_k).$$

It readily follows that the optimal parameter μ_k is

$$\tilde{\mu}_k = \frac{1}{T-1} \sum_{t=1}^{T-1} (\alpha_k^{t+1} - \tilde{w}_k \alpha_k^t) \approx (1 - \tilde{w}_k)\bar{\boldsymbol{\alpha}}_k =: \widehat{\mu}_k$$

where the approximation holds for large values of T. Thus, the optimization of w_k boils down to (up to a small error)

$$\widehat{w}_k = \underset{w_k}{\arg\min} \sum_{t=1}^{T-1} \left(\alpha_k^{t+1} - \bar{\alpha}_k - w_k(\alpha_k^t - \bar{\alpha}_k) \right)^2. \tag{7}$$

Doing so, we obtain the regularization term of (5).

Later (in the hyper-parameter selection step in Section 3.1), we will need also to retrieve all AR coefficients from observations. Obtaining the optimal w_k is straightforward from (7):

$$\widehat{w}_k = \frac{\sum_{t=1}^{T-1} (\alpha_k^{t+1} - \bar{\boldsymbol{\alpha}}_k)(\alpha_k^t - \bar{\boldsymbol{\alpha}}_k)}{\sum_{t=1}^{T} (\alpha_k^t - \bar{\boldsymbol{\alpha}}_k)^2}. \tag{8}$$

Regarding σ_k, we proceed similarly and we get that $\tilde{\sigma}_k$ is close to

$$\widehat{\sigma}_k^2 = \frac{\sum_{t=1}^{T-1} \left(\alpha_k^{t+1} - \widehat{\mu}_k - \widehat{w}_k \alpha_k^t \right)^2}{T-1}.$$

We observe that, replacing $\widehat{w}_k$ in (7) (and then in (4)) would result in a non-convex function in terms of $\boldsymbol{\alpha}_k$, which would increase the difficulty of the optimization problem (4). Therefore, optimize this parameter w_k as an extra variable is the best choice regarding convexity purposes.

Remark 1. *The model presented in this work can be easily generalized to an AR model of order $p \in \mathbf{N}$. The choice to introduce it in order 1 simplifies our notation and is adequate to our case of application, see Section 3.1.1. The DL optimization strategy, Section 2.2, applies likewise to an AR model of order $p \in \mathbf{N}$.*

The importance of the parameter μ_k A similar AR regularization, inspired in graph theory, is proposed in [32]. The difference between the latter and our model is that their μ_k is considered to be zero. We discuss in this paragraph why in our case of application this choice would not work.

Indeed, because of Equation (1) and the fact that Ω is a convex set, we expect $\{\alpha_k^t\}_{k=1}^K$ to be coefficients of a linear combination of $\{\mathbf{d}_k\}_{k=1}^K$ that approximates $\mathbf{P}^t \in \mathcal{M}^S$ for each t fixed, as explained in Equation (1). Then:

$$\sum_{k=1}^{K} \alpha_k^t \approx 1, \quad \text{for all } t \in [T]. \tag{9}$$

On the other hand, if $\mu_k = 0$, and $\boldsymbol{\alpha}_k$ are AR time-series of order 1 with coefficients μ_k and w_k , the estimator $\widehat{\mu}_k$ gives that:

$$0 \approx \widehat{\mu}_k = (1 - w_k)\bar{\boldsymbol{\alpha}}_k.$$

Either $w_k = 1$, which restricts a lot the possible time-modeling of $\boldsymbol{\alpha}_k$. Or, if $w_k \neq 1$ for all $k \in [K]$, then $\frac{1}{T}\sum_{t=1}^T \alpha_k^t \approx 0$ and summing in k, we get $\sum_{k=1}^K \sum_{t=1}^T \alpha_k^t \approx 0$, which contradicts (9). This contradiction shows the difficulty of fitting an AR model with drift 0 in the case where dictionaries lie in a convex set and where the codings are expected to be a convex combination.

2.2 Dictionary learning optimization strategy

Problem (4) is not a convex optimization problem, as it is usually the case in DL problems. Nevertheless, the problem is convex in variables $\mathbf{D}$, $\mathbf{A}$ and $\mathbf{w}$, as we can observe in (4). This property encourages the use of a policy that consists in alternating the minimization in $\mathbf{A}$, $\mathbf{D}$ and $\mathbf{w}$. This largely applied strategy does not ensure a global solution of problem (4), but it is a straightforward way of finding a local minima of problem.

The quadratic problems presented in this section are solved by a Interior Point Method, see ([23, Section 16.8],[31]). Interior point methods (IPMs) are very well-suited to solving quadratic optimization problems, particularly when sizes of problems grow large, see [11].

2.2.1 Dictionary update.

We opt for a sequential update of each atom of the dictionary: $\mathbf{d}_k$ for $k \in [K]$. This choice is guided by two advantages: 1. The problem is strictly convex for each atom $\mathbf{d}_k$ (as stated in Proposition 2.1 below) which is not necessarily true for the whole matrix $\mathbf{D}$. 2. This strategy breaks the problem in smaller problems making the resolution less dependent on the amount of atoms K. Updating atoms separately is also the strategy of the widely used K-SVD (see [9, Section 3.5]), however, the purpose in that case is to find a closed form for the optimization problem, which is not true in our case of study because of the form of constraints.

Proposition 2.1. *Assume that $\{\alpha_k^t\}_{t=1}^T$ is non zero. The minimization of (4) over $\mathbf{d}_k \in \operatorname{vec}\left(\mathcal{M}^E \cap \mathcal{M}^S\right)$ is equivalent to minimizing a strictly convex quadratic problem with linear constraints*

$$\min_{\mathbf{d}_k} \left\| \operatorname{vec}\left(\widetilde{\mathbf{P}}_k\right) - \widetilde{\mathbf{A}}_k \mathbf{d}_k \right\|_F^2, \quad s.t.\ vec^{-1}\left(\mathbf{d}_k\right) \in \mathcal{M}^E \cap \mathcal{M}^S, \tag{10}$$

where $\widetilde{\mathbf{P}}_k$ and $\widetilde{\mathbf{A}}_k$ are explicitly defined in (12).

Observe that the condition on α_k is expected to be systematically satisfied since each element α_k^t is non-negative.

Proof. Start from the reconstruction error in (4) and write

$$\|\mathbf{P} - \mathbf{D}\mathbf{A}\|_F^2 = \left\| \mathbf{P} - \sum_{j \neq k}^{K} \mathbf{d}_j \mathbf{A}_{j,:} - \mathbf{d}_k \mathbf{A}_{k,:} \right\|_F^2. \tag{11}$$

From this, it is obvious that the function $\mathbf{d}_k \mapsto \|\mathbf{P} - \mathbf{D}\mathbf{A}\|_F^2$ to minimize is quadratic and convex. However, under this form, it is not yet clear it is strictly convex. To establish this property, define

$$\widetilde{\mathbf{P}}_k := \mathbf{P} - \sum_{j \neq k}^{K} \mathbf{d}_j \mathbf{A}_{j,:},$$

$$\widetilde{\mathbf{A}}_k := \begin{bmatrix} A_{k,1} & 0 & \dots & 0 \\ 0 & A_{k,1} & \dots & 0 \\ \vdots & \vdots & \ddots & 0 \\ 0 & 0 & \dots & A_{k,1} \\ \vdots & \vdots & \vdots & \vdots \\ A_{k,T} & 0 & \dots & 0 \\ 0 & A_{k,T} & \dots & 0 \\ \vdots & \vdots & \ddots & 0 \\ 0 & 0 & \dots & A_{k,T} \end{bmatrix} \in \mathbb{R}^{dT} \otimes \mathbb{R}^d. \tag{12}$$

The quantity in (11) is thus equal to

$$\left\| \operatorname{vec}\left(\widetilde{\mathbf{P}}_k\right) - \widetilde{\mathbf{A}}_k \mathbf{d}_k \right\|_F^2.$$

Note that $\widetilde{\mathbf{A}}_k^\top \widetilde{\mathbf{A}}_k$ is diagonal matrix equal to $\sum_{t=1}^T (\alpha_k^t)^2 \mathcal{I}_{\mathbb{R}^d}$. □

2.2.2 Codings update

Similarly to the dictionary update, we adopt a strategy based on the update of each $\mathbf{A}_{k,:}$ for $k \in [K]$. The reasons are the same: it is preferable to solve a smaller and strictly convex optimization problem. The fact that the optimization for each k is a strongly convex problem is not straightforward and is argued in the proposition below.

Proposition 2.2. *Let $k \in [K]$ be fixed. Consider the minimization of (4)-(5) over one coding $\mathbf{A}_{k,:}$, i.e.*

$$\min_{\substack{\mathbf{A}_{k,:},\\ A_{k,t} \geq 0}} \|\mathbf{P} - \mathbf{D}\mathbf{A}\|_F^2 + \lambda \sum_{k=1}^{K} \sum_{t=1}^{T-1} \left(A_{k,t+1} - \bar{\mathbf{A}}_{k,:} - w_k (A_{k,t} - \bar{\mathbf{A}}_{k,:}) \right)^2. \tag{13}$$

For any $\lambda \geq 0$, the above problem is a strongly convex quadratic optimization problem with linear constraints.

Proof. First, there is a symmetric non-negative matrix $\mathbf{H}^{w_k} \in \mathbb{R}^T \otimes \mathbb{R}^T$ such that

$$\begin{aligned} \mathcal{R}_{AR}^k(A_{k,:}, w_k) &= \sum_{t=1}^{T-1} \left(A_{k,t+1} - \bar{\mathbf{A}}_{k,:} - w_k (A_{k,t} - \bar{\mathbf{A}}_{k,:}) \right)^2 \\ &= \langle \mathbf{A}_{k,:}^\top, \mathbf{H}^{w_k} \mathbf{A}_{k,:}^\top \rangle \end{aligned}$$

since for w_k fixed, $\mathcal{R}_{AR}^k(\cdot, w_k)$ is a quadratic problem without linear term that can be represented by a symmetric matrix. Obviously, it is non-negative.

Similarly to the proof for the dictionary update, we define a matrix $\widetilde{\mathbf{D}}_k$:

$$\widetilde{\mathbf{D}}_k := \begin{bmatrix} \mathbf{D}_{1,k} & 0 & \dots & \dots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ \mathbf{D}_{d,k} & 0 & \dots & \dots & 0 \\ 0 & \mathbf{D}_{1,k} & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & 0 \\ 0 & \mathbf{D}_{d,k} & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \\ 0 & \dots & \dots & 0 & \mathbf{D}_{1,k} \\ \vdots & \ddots & \ddots & \vdots & \vdots \\ 0 & \dots & \dots & 0 & \mathbf{D}_{d,k} \end{bmatrix} \in \mathbb{R}^{dT} \otimes \mathbb{R}^T;$$

note that the minimization problem (13) is equivalent to

$$\min_{\substack{\mathbf{A}_{k,:},\\ A_{k,t} \geq 0}} \left\| \operatorname{vec}\left(\widetilde{\mathbf{P}}_k\right) - \widetilde{\mathbf{D}}_k A_{k,:}^\top \right\|_F^2 + \lambda \langle \mathbf{A}_{k,:}^\top, \mathbf{H}^{w_k} \mathbf{A}_{k,:}^\top \rangle, \tag{14}$$

which is a quadratic constrained optimization problem with quadratic term given by the matrix:

$$\mathbf{C}_k := \widetilde{\mathbf{D}}_k^\top \widetilde{\mathbf{D}}_k + \lambda \mathbf{H}^{w_k}.$$

Since $\widetilde{\mathbf{D}}_k^\top \widetilde{\mathbf{D}}_k = \|\mathbf{d}_k\|_2^2 \mathcal{I}_{\mathbb{R}^T}$ and that $\|\mathbf{d}_k\|_2$ is uniformly bounded from below on $\mathrm{vec}\left(\mathcal{M}^E \cap \mathcal{M}^S\right)$, and since $\mathbf{H}^{w_k}$ is symmetric non-negative, the matrix $\mathbf{C}_k$ is symmetric positive definite with a uniform lower bound for its eigenvalues. The announced statement is proved. □

2.3 Coefficient update

We note that, for each $k \in [K]$, the optimization problem with respect to $\mathbf{w}_k$ in equation (4) is a 1-dimensional quadratic problem with explicit solution given by Equation (8).

Remark 2. *For each $k \in [K]$, the solution of problem (10) and (13) decreases the objective value of the respective optimization problems. However, it is an open question to justify that this strategy provides a solution to problems (4) with respect to $\mathbf{D}$, $\mathbf{A}$ and $\mathbf{w}$.*

Algorithm 1: Dictionary Learning (DL)

Input: *matrix of vectorized RMM:* $\mathbf{P} \in \mathbb{R}^d \otimes \mathbb{R}^T$,
number of atoms: $K \in \{1, 2, ...\}$,
regularization parameter: $\lambda > 0$
number of iterations: $N \in \{1, 2, ...\}$
Output: *optimized dictionary, codings and drift:* $\mathbf{D}, \mathbf{A}, \mathbf{w}$

initialize $\mathbf{D} \in \mathbb{R}^d \otimes \mathbb{R}^K$ and $\mathbf{A} \in \mathbb{R}^K \otimes \mathbb{R}^T$
for $i = 1 : N$ **do**
  # Dictionary update
  for $k = 1 : K$ **do**
    update $\mathbf{d}_k$ with QP s.t. $\mathrm{vec}^{-1}(\mathbf{d}_k) \in \mathcal{M}^E \cap \mathcal{M}^S$
  # Codings update
  for $k = 1 : K$ **do**
    update $\boldsymbol{\alpha}_k$ with QP s.t. $\alpha_k^t \geq 0,\ t \in [T]$
  # Coefficient update
  for $k = 1 : K$ **do**
    update w_k with Equation (8)

3 Experiments

Our proposed DL method with temporal AR regularization will be evaluated on real RMM provided by BNP Paribas. The dataset contains $T = 192$ one-year observed transition frequency matrices with shape $R = 11$ issued monthly from 52 sectors composed by large European capitalization companies between January 2004 and December 2019. This period contains in particular the subprime crisis but not the COVID-19 pandemic.

3.1 Experimental design

Ideally, we should apply our DL method on each sector. However given the dataset, the observed matrices are very sparse which refers to another problem formulation (missing data). To overcome this issue we computed a (confidential) weighted sum among the 52 sectors to form a shareable (on git) set of matrices $\mathbf{P} = \{\mathrm{vec}\,(\mathbf{P}^t) \in \mathbb{R}^d\}_{t=1}^T$ with $d = 110$ and $T = 192$. Those matrices are still noisy and so they might not respect the economic constraints, *i.e.* $\mathbf{P}^t \in \mathcal{M}^S, \forall t \in [192]$. Based on this dataset, we performed a classical 80/20 non-random train-test split in time, stored respectively in $\mathbf{P}^{\mathrm{Train}} \in \mathbb{R}^d \otimes \mathbb{R}^{T^{\mathrm{Train}}}$ and $\mathbf{P}^{\mathrm{Test}} \in \mathbb{R}^d \otimes \mathbb{R}^{T^{\mathrm{Test}}}$. In all the following experiments we use $K = 3$ for ease of interpretation since we want to represent the RMM as a combination of three regimes, assuming that each one represents an economic state. Larger values of K perform similar results but lead to a more sophisticated economic analysis.

3.1.1 AR lag estimation

First let us check the relevance of the AR(1) model in (5). Starting from the optimization problem (4) with $\lambda = 0$, we trained the DL model on $\mathbf{P}^{\mathrm{Train}}$ and applied the well-known partial autocorrelation function (PACF) [4, Section 3.2.5] on the codings $\mathbf{A}_{k,:}^{\mathrm{Train}}$, for all $k \in [3]$. For each series, we identified just one statistically significant lag, suggesting a possible AR(1) model adapted to the data. See Figure 2 for $k = 1$, while the others behave similarly.

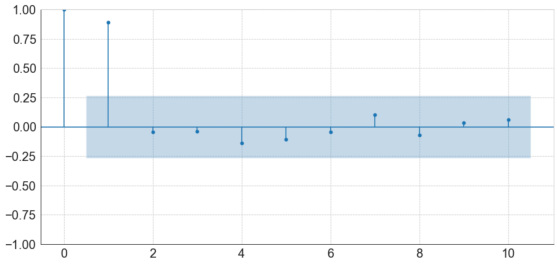

Figure 2: PACF of $\mathbf{A}_{1,:}^{\mathrm{Train}}$ for the first 10 lags. The shaded region represents the 95% confidence interval.

3.1.2 Hyper-parameter selection

The best hyperparmeter λ is chosen automatically through the procedure described in Algorithm 2. The latter allows to evaluate both the prediction and the reconstruction capacity of the model. We applied iteratively Algorithm 2 for $\lambda \in \{0.01, 0.1, 1, 3, 5, 6, 7, 10\}$ and stored the results in Table 1 which highlights the benefit of our proposed regularization

in the RMM predictions. The parameter associated with the smallest reconstruction error is $\lambda = 6$.

Algorithm 2: Hyper-parameter selection

Input: *data train:* $\mathbf{P}^{\text{Train}} \in \mathbb{R}^d \otimes \mathbb{R}^{T^{\text{Train}}}$,
data test: $\mathbf{P}^{\text{Test}} \in \mathbb{R}^d \otimes \mathbb{R}^{T^{\text{Test}}}$,
number of atoms: $K \in \{1, 2, ...\}$,
lambda: $\lambda > 0$

Output: *reconstruction error:* $\mathcal{E}$

$\mathbf{D}^{\text{Train}}, \mathbf{A}^{\text{Train}}, \mathbf{w}^{\text{Train}} \leftarrow \text{DL}(\mathbf{P}^{\text{Train}}, K, \lambda, 500)$, where the function DL refers to Algorithm 1

$\mathbf{A}^{\text{Test}} \leftarrow \text{Proj}_{\mathbf{D}^{\text{Train}}}\left(\mathbf{P}^{\text{Test}}\right)$

$A^{\text{Sim}}_{k,t} \leftarrow \widehat{\mu}_k + A^{\text{Test}}_{k,t} w^{\text{Train}}_k + \varepsilon^t_k$ with

$$\widehat{\mu}_k = \bar{\alpha}^{\text{Train}}_k (1 - w^{\text{Train}}_k),$$
$$\varepsilon^t_k \sim \mathcal{N}\left(0, \widehat{\sigma}^2_k\right),$$
$$\widehat{\sigma}^2_k \leftarrow \widehat{\text{Var}}[\boldsymbol{\alpha}^{\text{Train}}_k]\left(1 - (w^{\text{Train}}_k)^2\right),$$

for all $k \in [K]$ and $t \in [T^{\text{Test}} - 1]$

$\mathbf{P}^{\text{Reco}} \leftarrow \mathbf{D}^{\text{Train}} \mathbf{A}^{\text{Train}}$

$\mathbf{P}^{\text{Sim}} \leftarrow \mathbf{D}^{\text{Train}} \mathbf{A}^{\text{Sim}}$

$\mathcal{E} \leftarrow 0.8 \left\| \mathbf{P}^{\text{Test}}_{:,1:} - \mathbf{P}^{\text{Sim}} \right\|^2_F + 0.2 \left\| \mathbf{P}^{\text{Train}} - \mathbf{P}^{\text{Reco}} \right\|^2_F$ # without the first test value

λ	0.01	0.1	1	3	5	6	7	10
error	6.1	5.1	4.7	4.6	4.5848	**4.5846**	4.5847	4.593

Table 1: Reconstruction error from Algorithm 2 associated with various λ. The best result is emphasized in bold.

In the next section we illustrate the results of our DL model and propose two ML applications. First a time-series prediction of the RMM and second their unsupervised clustering in order to infer an estimation of the global economic sentiment.

3.2 Results

Computational aspects The numerical experiments have been conducted on a Macbook Pro (13-inch, M1, 2020), 512 Go SSD, 16 Go RAM. All the code was implemented in Python 3.10. It takes less than a minute to train the model with $K = 3$ during 500 iterations. Clearly from Algorithm 1 the training time is linear with respect to K.

3.2.1 Dictionary representation

Once learned, it appears that the dictionary managed to extract three candidates (atoms) to be good representatives for all RMM included in our dataset. In Figure 3 are represented these atoms in a matrix form generating a stable (strong diagonal, see Figure 3a), an upgrade (strong lower diagonal, see Figure 3b) and a downgrade (strong upper diagonal, see Figure 3c) risk configuration. Although automatically obtained by our algorithm, observe that these representatives make fully sense in terms of economic interpretation, and should reveal the underlying characteristics of RMM in our data set.

	1	2	3	4	5	6	7	8	9	10	11
1	91.17	6.23	0.00	1.33	0.10	0.78	0.30	0.09	0.01	0.00	0.00
2	0.31	93.87	3.22	0.91	0.51	0.70	0.39	0.09	0.01	0.00	0.00
3	0.31	5.07	79.24	10.93	3.04	0.93	0.38	0.09	0.01	0.00	0.00
4	0.01	1.34	6.67	83.00	5.37	3.14	0.38	0.09	0.01	0.00	0.00
5	0.01	0.25	0.89	6.27	80.05	9.86	2.05	0.61	0.01	0.00	0.00
6	0.00	0.02	0.55	0.71	6.46	85.48	5.86	0.92	0.01	0.00	0.00
7	0.00	0.02	0.06	0.23	0.34	5.52	88.85	4.98	0.01	0.00	0.00
8	0.00	0.00	0.00	0.00	0.00	0.04	4.66	95.30	0.01	0.00	0.00
9	0.00	0.00	0.00	0.00	0.00	0.00	4.69	17.71	77.59	0.00	0.00
10	0.00	0.00	0.00	0.00	0.00	0.00	0.40	11.48	0.00	87.85	0.26

(a) $\text{vec}^{-1}(\mathbf{d}_1)$

	1	2	3	4	5	6	7	8	9	10	11
1	100.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
2	9.23	61.82	24.71	2.10	0.73	0.00	0.00	0.89	0.22	0.29	0.01
3	1.50	14.93	79.33	0.00	0.57	1.82	0.06	0.93	0.34	0.34	0.18
4	1.12	0.49	23.07	63.87	7.09	1.42	1.05	0.91	0.37	0.39	0.23
5	1.12	0.49	4.02	31.24	53.29	6.85	0.76	0.34	0.59	1.00	0.31
6	0.59	0.99	0.70	4.38	31.29	45.97	9.31	3.16	0.94	1.55	1.13
7	0.58	0.80	0.88	1.24	6.48	33.83	31.56	16.50	4.30	1.31	2.52
8	0.58	0.72	0.85	0.76	2.67	9.67	39.09	24.71	9.94	5.44	5.58
9	0.58	0.72	0.78	0.78	2.73	4.11	10.93	0.00	67.80	5.91	5.67
10	0.00	0.00	0.00	0.00	1.15	1.22	18.25	0.00	48.79	0.00	30.59

(b) $\text{vec}^{-1}(\mathbf{d}_2)$

	1	2	3	4	5	6	7	8	9	10	11
1	67.86	21.46	4.94	3.51	0.44	1.14	0.00	0.00	0.00	0.00	0.64
2	0.00	81.81	11.37	4.59	0.44	0.60	0.16	0.01	0.21	0.15	0.64
3	0.00	0.00	91.67	6.10	0.44	0.60	0.16	0.01	0.20	0.16	0.64
4	0.00	0.00	0.00	81.92	16.29	0.60	0.16	0.01	0.19	0.18	0.64
5	0.00	0.00	0.00	2.90	85.00	7.09	2.22	0.97	0.54	0.64	0.64
6	0.00	0.00	0.00	1.24	5.01	72.67	14.51	1.58	1.56	0.99	2.44
7	0.00	0.00	0.00	0.54	1.64	6.03	75.40	9.56	3.33	0.75	2.75
8	0.00	0.00	0.00	0.22	0.93	0.97	20.89	56.45	10.38	3.67	6.48
9	0.00	0.00	0.00	0.21	0.15	1.76	0.00	15.43	0.00	32.09	50.36
10	0.00	0.00	0.00	0.00	0.00	0.00	2.12	1.88	0.00	43.07	52.92

(c) $\text{vec}^{-1}(\mathbf{d}_3)$

Figure 3: Matrix representation of the atoms in a trained dictionary with $K = 3, \lambda = 6$.

3.2.2 Codings.

The AR regularization (5) enforces an AR behavior of the codings making them more regular. The choice of the best prediction and analysis will then be a trade-off between the

reconstruction and the regularity of the time evolution of the codings. Figure 4 depicts this evolution. Note that for larger

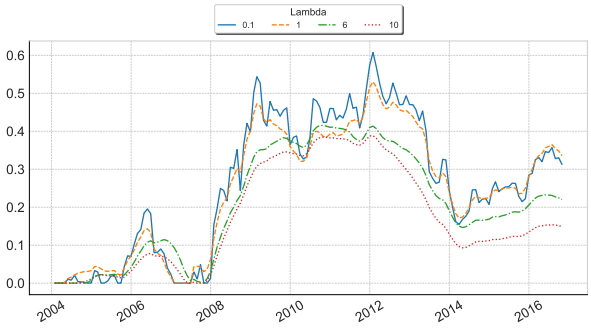

Figure 4: Time evolution of $\mathbf{A}^{\text{Train}}_{3,:}$ for $\lambda \in \{0.1, 1, 6, 10\}$.

values of λ the evolution in time of the coding is smoother. This property tends to advantage the prediction of future matrices.

3.2.3 Clustering

Let study now the benefit of our proposed regularized DL model in order to obtain an interpretable classification of the RMM. We fit a KMeans algorithm on the standardized $\mathbf{A}^{\text{Train}}$ in 3 clusters and predict the classes of the standardized $\mathbf{A}^{\text{Train}}$. Assuming that the atoms and the clusters represent different economic states, we obtain in Figure 5 a classification in time of the observed RMM. Additionally to the historical financial context, we present how to infer an economic sentiment indicator based on both the codings' classification and the dictionary. To do so, we store in Table 2 the weights of the atoms assigned to each cluster. Thus, combining with Figure 3, we can easily deduce that the labels associated to the clusters green, yellow, red are respectively a good, a stable and a bad economic sentiment indicator. Such an allocation can be confirmed graphically in Figure 5 which captures effectively the financial bubble between 2006-2008, as well as the subrprime crisis.

cluster \ k	1	2	3
green	81.7	11.7	6.6
yellow	60.3	23.6	16.2
red	57.0	6.6	36.4

Table 2: Centroid of $\mathbf{A}^{\text{Train}}$ in each cluster: green, yellow, red. The values are adjusted in percentage.

4 Conclusion

Modeling RMM is a challenging problem because it is necessary to find an interpretable representation, satisfying economic constraints, while the data are involving in time, not numerous and in a dimension close to the number of observations. We propose a new data-based method using a dictionary learning approach, which implementation boils down to solve small-dimension quadratic optimization problems with linear constraints, leading to a fast algorithm. On the modeling size, we overcome the challenge of a constrained dictionary learning problem and progress in temporal comprehension of the data through the AR regularization. When tested on a real data-set, the method enjoys good accuracy for reconstruction and includes the classification with respect to economic sentiment indicator.

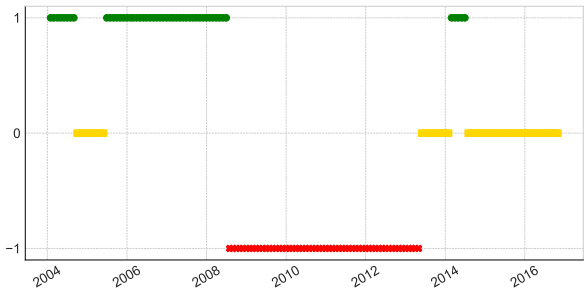

Figure 5: Unsupervised classification of the codings $\mathbf{A}^{\text{Train}}$ in 3 clusters: $\{-1$ (red cross), 0 (yellow square), 1 (green dot)$\}$.

As perspectives for further works, the clustering analysis suggests that it is possible to connect the temporal evolution of the codings with important macro-economical variables such as GDP or financial indices. The integration of these real indicators in the DL model, as well as its comparison with the Gaussian copula model are next steps for the consolidation of the model.

References

[1] P. Artzner, F. Delbaen, J.-M. Eber, and D. Heath. Coherent measures of risk. *Math. Finance*, 9(3):203–228, 1999.

[2] T. R. Bielecki and M. Rutkowski. *Credit risk: modelling, valuation and hedging.* Springer Finance. Springer-Verlag, Berlin, 2002.

[3] F. Bourgey, E. Gobet, and C. Rey. Metamodel of a large credit risk portfolio in the gaussian copula model. *SIAM Journal on Financial Mathematics*, 11(4):1098–1136, 2020.

[4] G. E. Box, G. M. Jenkins, and G. C. Reinsel. *Time Series Analysis: Forecasting and Control.* John Wiley and Sons, 2008.

[5] A. Chambolle, V. Caselles, M. Novaga, D. Cremers, and T. Pock. An introduction to total variation for image analysis, 2010.

[6] S. S. Chen, D. L. Donoho, and M. A. Saunders. Atomic decomposition by basis pursuit. *SIAM J. Sci. Comput.*, 20(1):33–61, 1998.

[7] U. Cherubini, E. Luciano, and W. Vecchiato. *Copula methods in finance*. Wiley Finance Series. John Wiley & Sons, Ltd., Chichester, 2004.

[8] Y. Cho and L. K. Saul. Learning dictionaries of stable autoregressive models for audio scene analysis. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 169–176, 2009.

[9] B. Dumitrescu and P. Irofti. *Dictionary learning algorithms and applications*. Springer, 2018.

[10] M. Elad and M. Aharon. Image denoising via sparse and redundant representations over learned dictionaries. *IEEE Trans. Image Process.*, 15(12):3736–3745, 2006.

[11] J. Gondzio. Interior point methods 25 years later. *European Journal of Operational Research*, 218(3):587–601, 2012.

[12] R. Gribonval, R. Jenatton, F. Bach, M. Kleinsteuber, and M. Seibert. Sample complexity of dictionary learning and other matrix factorizations. *IEEE Trans. Inform. Theory*, 61(6):3469–3486, 2015.

[13] D. X. Li. On default correlation: A copula function approach. *The Journal of Fixed Income*, 9(4):43–54, 2000.

[14] D. MacKenzie and T. Spears. the formula that killed wall street: The gaussian copula and modelling practices in investment banking. *Social Studies of Science*, 44(3):393–417, 2014.

[15] J. Mairal, F. Bach, and J. Ponce. Sparse modeling for image and vision processing. *arXiv preprint arXiv:1411.3230*, 2014.

[16] J. Mairal, F. Bach, J. Ponce, and G. Sapiro. Online dictionary learning for sparse coding. In *Proceedings of the 26th annual international conference on machine learning*, pages 689–696, 2009.

[17] J. Mairal, M. Elad, and G. Sapiro. Sparse representation for color image restoration. *IEEE Trans. Image Process.*, 17(1):53–69, 2008.

[18] J. Mairal, G. Sapiro, and M. Elad. Multiscale sparse image representationwith learned dictionaries. In *2007 IEEE International Conference on Image Processing*, volume 3, pages III–105. IEEE, 2007.

[19] J. Mairal, G. Sapiro, and M. Elad. Learning multiscale sparse representations for image and video restoration. *Multiscale Modeling & Simulation*, 7(1):214–241, 2008.

[20] S. Mallat. *A wavelet tour of signal processing*. Elsevier/Academic Press, Amsterdam, third edition, 2009.

[21] A. J. McNeil, R. Frey, and P. Embrechts. *Quantitative risk management*. Princeton Series in Finance. Princeton University Press, Princeton, NJ, revised edition, 2015. Concepts, techniques and tools.

[22] K. Neusser. *Time Series Econometrics*. Number 978-3-319-32862-1 in Springer Texts in Business and Economics. Springer, June 2016.

[23] J. Nocedal and S. Wright. *Numerical Optimization*. Springer New York, NY, 1999.

[24] B. A. Olshausen and D. J. Field. Sparse coding with an overcomplete basis set: A strategy employed by v1? *Vision research*, 37(23):3311–3325, 1997.

[25] M. Protter and M. Elad. Image sequence denoising via sparse and redundant representations. *IEEE Transactions on Image Processing*, 18(1):27–35, 2009.

[26] Z. Qiao, G. Redler, B. Epel, and H. Halpern. A balanced total-variation-chambolle-pock algorithm for epr imaging. *Journal of Magnetic Resonance*, 328:107009, 2021.

[27] R. Shumway and D. Stoffer. *Time Series and Its Applications*. Springer, New York, 2011.

[28] P. Sprechmann and G. Sapiro. Dictionary learning and sparse coding for unsupervised clustering. In *2010 IEEE international conference on acoustics, speech and signal processing*, pages 2042–2045. IEEE, 2010.

[29] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal Roy. Statist. Soc. Ser. B*, 58(1):267–288, 1996.

[30] C. Vincent-Cuaz, T. Vayer, R. Flamary, M. Corneli, and N. Courty. Online graph dictionary learning. In *International Conference on Machine Learning*, pages 10564–10574. PMLR, 2021.

[31] M. Wright. The interior-point revolution in optimization: History, recent developments, and lasting consequences. *Bulletin of The American Mathematical Society*, 42:39–57, 2004.

[32] H.-F. Yu, N. Rao, and I. S. Dhillon. Temporal regularized matrix factorization for high-dimensional time series prediction. *Advances in neural information processing systems*, 29, 2016.