ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

DEPARTMENT OF ECONOMICS

SECOND CYCLE DEGREE
IN
ECONOMICS AND ECONOMETRICS

Dynamic Matrix Factor Models and the EM Algorithm: A Nowcasting Framework for Mixed-Frequency Data in the Euro Area

Dissertation in Macroeconometrics

Defended by:
Davide Delfino

ID number:
1126595

Supervisor:
Prof. Matteo Barigozzi

Session 1st
Academic Year 2024-2025

Abstract

The most recent macroeconometric literature has proposed Dynamic Matrix Factor Models (*DMFMs*) to address two major challenges in empirical analysis. First, this specification manages the high dimensionality of modern datasets through the use of low-rank latent structures; second, it improves model accuracy and forecasting performance by incorporating the dynamics of latent variables. The key innovation introduced by these models, compared to their vector counterparts, is the preservation of the natural matrix structure inherent in macroeconomic data.
This thesis presents a novel application of *DMFMs* for nowcasting matrix-variate time series, specifically focusing on the Gross Domestic Product (GDP) of major Euro Area (EA) countries. Nowcasts are produced by recursively estimating the DMFM parameters using the Expectation-Maximization (EM) algorithm, by simulating a pseudo real-time forecasting exercise.
This empirical framework allows for handling (i) the matrix structure of the data, (ii) mixed-frequency datastes, and (iii) general patterns of missing data, such as those introduced by the COVID-19 disruptions.
Standard Dynamic Factor Models (*DFMs*) in vector form are used as a benchmark to evaluate the nowcasting performance of *DMFMs*. Empirical results show that, under certain conditions, the additional dimensionality captured by *DMFMs* enhances forecasting accuracy. In particular, nowcasts for Germany and Italy significantly improve, suggesting that more interconnected and structurally central countries benefit the most from the matrix-based formulation.

Keywords: Dynamic Matrix Factor Models; Expectation-Maximization Algorithm; Kalman Smoother; Missing Values; Mixed Frequency; Nowcasting; Euro Area GDP;

Contents

1 Introduction **4**

2 Literature Review **9**

3 Methodology **13**

3.1 The Model 13

3.1.1 Dynamic Matrix Factor Model 14

3.1.2 Dynamic Factor Model 17

3.2 Estimation 19

3.2.1 Log-Likelihood 19

3.2.2 EM Algorithm 20

3.2.3 Parameters' Initialization 25

3.2.4 EM Convergence 31

3.3 Extension To missing Data 32

3.3.1 Imputation for the EM initialization 32

3.3.2 Modification of the EM-algorithm 34

3.4 Nowcasting 36

4 Empirics **38**

4.1 Data 39

4.1.1 Country Selection 40

4.1.2 Variables Selection 41

4.2 DMFM Factors: Selection and Interpretation 44

4.3 Euro Area Nowcasting 47

4.3.1 DMFM vs DFM 48

4.3.2 Contributions and Future Works 53

5 Conclusion **56**

Appendices **60**

A Additional Table **60**

B Additional Figures **60**

1 Introduction

Recent literature in macroeconometrics is currently facing two major, interconnected, and demanding challenges: managing the high dimensionality of modern datasets and developing more sophisticated techniques to improve time series analysis and forecasting accuracy. This Thesis contributes by providing an empirical application based on Euro Area (EA) macroeconomic data, referring to recent developments in factor models, estimation methods, and forecasting strategies. Specifically, it addresses these challenges by extending traditional factor models to the lowest-order representation of tensor data structures, namely matrix-variate time series, where observations are structured across three dimensions: units (rows), variables (columns), and time (slices). On this structure the Expectation-Maximization (EM) Algorithm is applied as an estimation technique, that is convenient also to provide nowcasts for EA country-specific GDP exploiting all the information that the model can retrieve from other countries and variables selected.

In recent years, matrix-variate time series have become increasingly popular in applied macroeconomic and financial research. In these fields, it is common to work with sequences of two-dimensional data arrays evolving over time. Consider, for instance, macroeconomic indicators observed across multiple countries, or asset returns and volatilities across portfolios. Several methodologies have been proposed to analyze such data structures, including matrix autoregressive models (e.g.,R. Chen, Xiao, and Yang 2021, Billio et al. 2023), matrix panel regression models (e.g., Kapetanios, Serlenga, and Shin 2021), and matrix factor models (e.g., Wang, Liu, and R. Chen 2019, L. Yu et al. 2022, or Matteo Barigozzi and Trapin 2025), that are the class of models explored in this Thesis.

Matrix Factor Models (*MFMs*) allow to preserve the original structure of the data as collected, and through factor analysis, provide a significant reduction in the dimensionality of the problem. Moreover, by introducing some dynamic in the latent factors, these models can also be convenient in terms of forecasting performances. Indeed, factors may evolve over time according to a matrix autoregressive (*MAR*)process, and from a static model for matrix-valued time series it is possible to implement a so-called Dynamic Matrix Factor Model (*DMFM*). This Thesis aims to show that *DMFMs* can outperform traditional vector-based Dynamic Factor Models (DFMs), which are typically applied to two-dimensional datasets within a multivariate time series framework, namely a set of variables observed over time. The comparison between these two class of models is provided in the context of nowcasting.

Before introducing the estimation methods and the nowcasting framework, it is worth stressing how preserving the matrix structure of the data, rather than vectorizing it, offers significant advantages. Indeed, even if it is a common practice to turn a matrix into a vector and apply standard vector time series analysis, it can be the case that the structural relationships embedded in the matrix get lost. In contrast, matrix factor models retain the original structure and exploit all the information that data collected in matrix form can convey. *MFM* can enhance the interpretability of latent factors, reduce the number of parameters to estimate, and more effectively capture dependencies and commonalities across rows and columns. This is particularly important when dealing with datasets in which rows and columns represent distinct but structurally connected entities, as for instance in our empirical example where economic indicators and EA countries can represent different layers of commonalities. To get convinced on how matrix data are the most general case possible, consider the following Matrix-valued time series dataset at a certain time t:

Country	**GDP Growth**	**Unemployment Rate**	**...**	**Industrial Production**
Germany	$X_{t,11}$	$X_{t,12}$	$\dots$	$X_{t,1p}$
France	$X_{t,21}$	$X_{t,22}$	$\dots$	$X_{t,2p}$
Italy	$X_{t,31}$	$X_{t,32}$	$\dots$	$X_{t,3p}$
Spain	$X_{t,41}$	$X_{t,42}$	$\dots$	$X_{t,4p}$

Table 1: Matrix-Valued Time Series Dataset

Table 1 shows just a slice of the time series $t = 1 \dots T$ and represents a matrix-valued time series. This structure generalizes classical univariate and multivariate time series models. For instance, a univariate time series describes the temporal evolution of a single variable (i.e., a single entry of the table repeated over time), while multivariate or panel consider vectors of variables observed jointly for a single unit (e.g., a row observed over time). However, in many empirical contexts, as in the economic and financial examples mentioned above, the data are more appropriately structured as matrices evolving over time. Whiting this framework, it is reasonable to assume that raws, such as countries within the same monetary union (for instance, in the European Monetary Union) share common features and can be convenient implement matrix factor models to provide a more general, flexible, and informative framework for modeling such data jointly.

Now that the advantages of the matrix-variate time series modeling are clear, it is worth introducing the Model considered in this Thesis. Recent advances in high-dimensional econometrics have led to the development of matrix-based factor models, namely the Matrix Factor Models (*MFM*), and the seminal contribution in this area was provided by Wang, Liu, and R. Chen 2019, who proposed modeling the data observed at each time t as a matrix $Y_t \in \mathbb{R}^{p_1 \times p_2}$, represented as:

$$Y_t = RF_tC^\top + E_t, \tag{1}$$

This specification preserves the bilinear structure in both the common and idiosyncratic components. Specifically, $F_t \in \mathbb{R}^{k_1 \times k_2}$ is a matrix of latent factors; $R \in \mathbb{R}^{p_1 \times k_1}$ and $C \in \mathbb{R}^{p_2 \times k_2}$ are the row and column loading matrices, respectively. finally E_t is the idiosyncratic error term. In this Thesis, E_t is allowed to exhibit both temporal and cross-sectional dependence across rows and columns. By definition, this class of models is referred to as approximate factor models, since it allows the idiosyncratic components to exhibit some weak dependence, rather than assuming they are entirely white noise with a diagonal structure.

Although Equation (1) defines a static factor structure, the model can be extended to a dynamic setting allowing latent factors to evolve over time. The problem can consequently be specified with a state-space representation where the measurement equation is given by Equation (1), and the transition equation for the latent factors, given by Equation (2), is assumed to follow a Matrix Autoregressive (*MAR*) process, as proposed by R. Chen, Xiao, and Yang 2021. *MAR* processes are a matrix-valued extension of the traditional *VAR* model and, for the latent variables, takes the form of:

$$F_t = \sum_{i=1}^{P} A_i F_{t-i} B_i^\top + U_t, \tag{2}$$

Equation (2) describes the temporal dynamics of the latent factors accounting for the bilinear structure for rows and columns in the autoregressive terms. For a generic order $i = \{1 \ldots P\}$, A_i and B_i are coefficient matrices respectively for rows and columns. In this Thesis, without loss of generality, the lag effect is assumed to last one period, namely $i = 1$.

Building on the original contribution of Wang, Liu, and R. Chen 2019, the literature has introduced several important and more general advancements, particularly regarding estimation methods, that are the main references for this Thesis. The first is the projected estimation method developed by L. Yu et al. 2022. This approach, based on Principal Component (*PC*) techniques, is employed for the initial estimation of the row and column loading matrices in the DMFM. As a PC-based method, it requires an imputation step in the presence of missing values. The imputation strategy adopted follows the methodology proposed by Cen and Lam 2025, which extends the “all-purpose estimator” of Xiong and Pelger 2023 to matrix-valued time series. The second and principal methodological contribution is the estimation framework by Matteo Barigozzi and Trapin 2025, which generalizes the EM algorithm, originally developed for vector factor models by Doz, Giannone, and Reichlin 2012, and later adapted to missing data settings by Bańbura and Modugno 2014, to the context of high-dimensional *DMFMs*. Within this framework, the projected estimator is used as the initialization step of the EM algorithm. It is also worth noting that the vector factor model, along with the estimation and forecasting strategy proposed by Bańbura and Modugno 2014, is used in this Thesis as a benchmark to assess the performance of the *DMFM* in terms of both estimation accuracy and nowcasting. The framework developed by Matteo Barigozzi and Trapin 2025 is in fact based on the same EM algorithmic structure as that of Bańbura and Modugno 2014, and it is consequently appealing to compare the performance of factor models in vector and matrix form by the results achieved using these two models for nowcasting.

Indeed, this Thesis adopts as a forecasting technique the nowcasting, a practical and increasingly popular method, adopted especially in Central Banks. Nowcasting is a real-time forecasting strategy that exploits the different frequency of variables and the staggered release of economic indicators. It basically allows to manage jagged edge datasets. Through this approach one can produces real-time estimates of low-frequency targets, such as GDP, by incorporating more timely, high-frequency variables. The method is particularly effective in this setting, because by reproducing a real-time flow of information, within a mixed-frequency framework, where monthly variables are released at different times within a quarter, it is possible to track the monthly contribution of monthly indicators on nowcasts updates through the Kalman Filter. The nowcasting framework was originally proposed by Giannone, Reichlin, and Small 2008, who built a large factor model estimated via principal components and updated through a Kalman smoother. While nowcasting is now standard practice, its integration with tensor, or matrix-structured data, remains relatively recent and underexplored. The Rolling-Nowcast proposed in this Thesis is a variant of the one in Giannone, Reichlin, and Small 2008 and extends the method by Bańbura and Modugno 2014 to the matrix case exploiting the EM algorithm. This estimation method is especially well-suited to this setting, as it naturally incorporates Kalman filtering techniques.

Based on the methodological framework described above, this Thesis proposes an empirical application using the "EA-MD-QD: Large Euro Area and Euro Member Countries Datasets for Macroeconomic Research" by Matteo Barigozzi, Lippi, and Luciani 2021, which includes both monthly and quarterly macroeconomic indicators from January 2000 to April 2025, for a total of $T = 300$ time periods. The empirical analysis focuses on Germany, France, Italy, and Spain, using 39 monthly variables and quarterly GDP. [1]. To implement the nowcasting exercise, the dataset is recursively truncated from the first quarter of 2017 (Q1:2017) to the first quarter of 2025 (Q1:2025), corresponding to the most recent available observation at the time of the implementation. At each iteration, the model is estimated by applying a masking procedure that replicates the actual release schedule of each variable, simulating a pseudo-real-time flow of information. GDP is then nowcasted for each country using: (i) all available information across countries in the matrix-based approach, or (ii) only country-specific information in the vector-based model. The model is re-estimated at each step, and the final variable values,computed using the Kalman smoother,are retained as nowcasts, since they correspond to the filtered predictions of the Kalman filter. The estimation of the *DMFM* at each step is performed using the Expectation-Maximization (EM) algorithm, which is selected over alternative methods (discussed in Chapter 2) due to its strong suitability for handling missing data and for enabling real-time nowcasting. This choice is further motivated by the need for flexibility in managing mixed-frequency structures and the presence of outliers during the COVID-19 period, when real variables, the most affected, are masked. The EM algorithm is particularly appropriate in this context not just because it accommodates general missing data patterns, but also because, as an iterative approach, it allows to find a way out to a quasi-likelihood with no closed-form solution.

To sum up, the application of nowcasting to GDP in Euro Area countries using *DFMs* in both vector and matrix form, addresses two central research questions:

- Does modeling data in matrix form improve forecasting accuracy at the country level?
- Does incorporating high-frequency information enhance GDP predictions?

Comparative results will show that, under certain conditions, nowcasting GDP for Euro Area countries using monthly variables through a *DMFM* can outperform the vector-based approach, especially for those countries that are structurally more central within the matrix. However, *DFMs* remain a desirable strategy for other countries, and are often preferable in terms of computational efficiency.

Finally, the contribution of this thesis lies in the empirical implementation of ricent advanced models, estimation techniques, and forecasting strategies, as it:

- accommodates structured macroeconomic datasets collected across comparable observational units through matrix-variate time series models;
- handles complex patterns of missing data arising from mixed frequencies, irregular release schedules, or disruptions such as the COVID-19 pandemic, through the implementation of the EM algorithm as an estimation technique;

[1]The accompanying R replication code, available at `https://github.com/dolpolo/Nowcasting_DMFM`, allows full reproducibility and flexible selection of countries and variables up to the limit of the reference dataset.

- supports nowcasting, one of the most widely used forecasting tools in modern central banking, representing one of the first applications of the nowcasting framework to matrix-variate time series.

Structure of the Thesis: The remainder of this work is structured as follows. Chapter 2 reviews the origins of factor models and their application to matrix-valued time series data, including estimation strategies proposed in the literature and motivating the methodological choices adopted in this thesis. Chapter 3 outlines the adopted methodological framework, presenting the dynamic matrix factor compared it to its vector counterpart. The discussion is strictly focused on the estimation procedure via the EM algorithm, its initialization and the adaptation to the nowcasting implementation. Chapter 4 introduces the empirical dataset, evaluates and compares the nowcasting performance of the models, and analyzes the results. Codes are available in `MATLAB`, for the Dynamic Factor Model in vector form and in `R`, for the DMFM. Finally, Chapter 5 summarizes the key findings and offers directions for future research.

2 Literature Review

This initial chapter aims to review the theoretical framework and the literature underlying this work, both in terms of model formalization and estimation methodology. I begin with a discussion on the evolution from traditional Dynamic Factor Models (DFMs) to their most recent matrix-valued extensions, namely Dynamic Matrix Factor Models (DMFMs). The main estimation approaches proposed in the literature for factor models in vector form can be extended also to the matrix case, and among them I particular focus on Quasi-Maximum Likelihood Estimation (QMLE) via the Expectation-Maximization (EM) algorithm, which is the method adopted in this work. This technique deserves special attention, especially in high-dimensional context and in presence of mixed-frequency datasets, because it allows to manage real world dataset with flexibility. Finally, the adoption of this estimation technique is appealing considered the empirical analysis this work provides. The final part of the chapter highlights the practical relevance of this framework in applied macroeconomics, with a focus on nowcasting applications and the related literature

To begin, it is useful to briefly outline the general framework studied in this thesis, namely factor analysis, one of the most common and widely used unsupervised statistical learning techniques. As a multivariate method, it is especially effective in high-dimensional settings where the number of time series n can be large relative to the number of time periods T. The central idea is to represent the n observed time series as a combination of $r \ll n$ latent factors, where factor loadings represent the influence of each factor on the n observed variables, and an idiosyncratic component, capturing the share of variability not explained by the common component.

This technique was originally introduced by Spearman (1904) in psychology, and borrowed by econometricians later on, during the final decades of the 20th century. The striking feature of this technique is the ability to achieve substantial dimensionality reduction, particularly in the high- and ultra-high-dimensional contexts, a feature that nowadays characterizes the majority of macroeconomic and financial datasets. Through factor analysis one can study co-movements in large datasets, estimate structural shocks, and forecast macroeconomic aggregates. This work contributes to that literature by applying a factor model and its corresponding estimation strategy for real-time GDP forecasting, a technique commonly known as nowcasting. The originality of the work is the extension of this forecasting technique to matrix-valued time series data, a field recently explored in economics where factor analysis is been successfully applied.

Indeed, the most recent literature in econometrics has proposed a historical transition from vector to matrix factor models, particularly suited for observations that are naturally structured in matrix form, such as panels of macroeconomic indicators collected across countries over time. A pioneering contribution to this line of research is provided by Wang, Liu, and R. Chen 2019, who introduced a matrix-valued factor model in which each observation is linked to both row and column latent factors. They propose an estimation via eigen-decomposition of the long-run covariance matrix. This method was then also applied in E. Y. Chen, Tsay, and R. Chen 2020, who further propose a general framework that incorporates prior knowledge or domain constraints into the model through linear restrictions.

Although the great intuition underlying this advancements in factor analysis, the original framework still poses two restrictive, but overcomable assumptions: independence of the idiosyncratic component both in time and cross-sectionally, and a static structure of latent factors. Subsequent research addressed these limitations. In particular, projected estimators were introduced by L. Yu et al. 2022 and E. Y. Chen and Fan 2023, allowing for autocorrelated idiosyncratic components. Their two-step methodology first estimates the number of factors and the loading matrices using principal components, and then projects the data accordingly to produce a significant dimensionality reduction. From this projection they recover through iterative algorithms the correct number of factors, based on the eigenvalue-ratio and a most effective estimates of loadings both for rows and columns. This approach is especially relevant here, as it is used to initialize the EM algorithm in our setup and provides a first level of generalization beyond the framework proposed by Wang, Liu, and R. Chen 2019. More recently Xu, Yuan, and Guo 2025 developed Q-MLE techniques specifically tailored for matrix factor models. These methods allow for heteroskedastic idiosyncratic components and avoid the need for a fully specified dynamic structure in the factors, thus enhancing robustness and feasibility in empirical applications.

Nevertheless, so far no dynamics on factors are yet considered. Indeed, introducing temporal dependence can provide valuable insights for forecasting purposes, enhancing the predictive power of factor models. According to the temporal line, advancements in the direction of dynamic factor models for matrix-variate time series data, have been preceded by an alternative approach: an adaptation of vector processes for time series to matrix data. Indeed, a first step in this direction is taken by R. Chen, Xiao, and Yang 2021, who generalize traditional vector autoregressive techniques to the matrix setting, maintaining the data's structure and preserving bilinearity. As in the MFM proposed by Wang, Liu, and R. Chen 2019, this approach achieves substantial dimensionality reduction and enhanced interpretability, although without involving latent factors. It should be noted, however, that matrix autoregressive (MAR) models are not always the most appropriate specification. As shown by Tsay 2024, while MAR models are convenient for matrix-variate time series analysis, matrix moving average (MMA) models may perform better in some contexts, such as modeling matrix-valued seasonal time series. In that context, Tsay 2024 extends traditional VARMA models to the matrix setting and proposes maximum likelihood estimation based on recursive innovations.

More recently, the theory of factor analysis for matrix-valued data has drawn inspiration from these advances and from the literature on dynamic factor models in vector form, which suggest that temporal dynamics can be incorporated via a state-space representation. As a result, the matrix factor models (MFMs) discussed above have been extended by allowing the latent factors to follow a dynamic process, typically specified as a matrix autoregressive (MAR) model. The resulting model is referred to as a Dynamic Matrix Factor Model (DMFM), and it has been formally studied in recent contributions by R. Yu et al. 2024 and Matteo Barigozzi and Trapin 2025. The latter provides the main reference for this thesis, as the estimation strategy employed here is directly inspired by their framework. They estimate DMFMs using an adapted EM algorithm for the matrix case. The initialization relies on the static projected estimator from L. Yu et al. 2022, while the EM algorithm itself generalizes the frameworks proposed by Doz, Giannone, and Reichlin 2012 and Bańbura and Modugno 2014. In particular, Doz, Giannone, and Reichlin 2012 demonstrate that the EM estimator for DFMs in vector form remains robust to mis-specification in the cross-sectional and serial correlation of idiosyncratic components using the Kalman smoother. Meanwhile, Bańbura and Modugno 2014 extend the EM algorithm—originally proposed by Watson and Engle 1983 for DFMs, to cases with arbitrary patterns of missing data and serially correlated idiosyncratic components. Their framework efficiently handles datasets with varying publication delays, mixed frequencies, and different sample lengths. It is well suited to tasks such as interpolation, backcasting (e.g., for emerging economies), and nowcasting. A further contribution of Bańbura and Modugno 2014 is the introduction of a model-based approach to quantify the impact of data releases (referred to as "news") on forecast revisions. This mechanism allows for interpreting nowcasting updates in terms of sign and magnitude, and helps to decompose the contribution of individual releases to changes in the forecast—especially useful when multiple indicators are published simultaneously.

Summing up, the approached followed by Matteo Barigozzi and Trapin 2025, that inspires this work is actually a generalization to the matrix case of Bańbura and Modugno 2014. Moreover, a distinctive feature of the approach by Matteo Barigozzi and Trapin 2025, in comparison to R. Yu et al. 2024, regardless the treatment of missing data patterns, is that their estimation technique integrates Kalman filtering within the EM framework, thereby enabling the model to accommodate unbalanced panels and mixed-frequency data. Handling missingness in high-dimensional time series is a recurring challenge, and addressing it adds a significant layer of generality to empirical analysis. The EM algorithm, being iterative, requires parameter initialization. This is provided via projected estimates, which are based on Principal Component (PC) methods. Since PC approaches are not robust to missing data, an initial imputation step is required. This thesis follows the imputation strategy proposed by Cen and Lam 2025, which extends the "all-purpose estimator" by Xiong and Pelger 2023 to the tensor setting.

If not mis-specified, the log-likelihood estimation via Maximum-Likelihood of an approximate DMFMs, as specified in this work, would be computationally infeasible. Chapter 3 discusses this issue in detail, drawing from the approach of Matteo Barigozzi and Trapin 2025. The high dimensionality and complexity of DMFMs make the full maximum likelihood estimation unfeasible. To address this issue, they adopt a quasi-likelihood framework inspired by Tipping and Bishop 1999 in the context of probabilistic PCA. As in Doz, Giannone, and Reichlin 2012, the idiosyncratic components are treated as approximately i.i.d., allowing for a tractable and robust estimation process via the EM algorithm.

Finally, since the core of this dissertation lies in the empirical comparison of forecasting performance between vector and matrix formulations of Dynamic Factor Models (DFMs), I briefly review the nowcasting methodology. Nowcasting—the forecasting of the present or very near future—has become increasingly relevant for central banks and policymakers. The foundational work by Giannone, Reichlin, and Small 2008 introduced a two-step procedure that combines principal components with Kalman filtering to manage unbalanced data releases in real time.

In this thesis, I adopt a modified version of their framework, adapted to matrix-valued data in which each country consistently reports a panel of indicators over time. Following the approach of Bańbura and Modugno 2014, who extend the nowcasting framework to account for general patterns of missing data and perform inference via the Kalman smoother, I reconstruct the full dataset at the end of each recursive step and extract the implied nowcasts directly from the smoothed estimates. This strategy enables a recursive update of GDP nowcasts month by month within each quarter, reflecting the real-time flow of incoming information.

While the majority of nowcasting applications to date have employed vector-valued models, the matrix formulation offers a novel advantage: it allows for the decomposition of nowcast revisions by both variable (columns) and cross-sectional unit (rows). This multidimensional structure opens the door to more granular analyses of informational content and shock transmission mechanisms across countries and indicators—an avenue of research that remains relatively underexplored and holds promise for future applications.

3 Methodology

This chapter describes the methodology underlying the emprical analysis on Euro Area data presented in Chapter 4. The main focus is on the EM algorithm and its capability to make this study realistic by handling general patterns of missing values, in particular the mixed-frequency dataset like the one addressed in the empirics. The key objective of this study is to provide a setup as general as possible, allowing for the management of real data and their inconvenient features, such as missing observations and outliers during crises, such as the pandemic.

In recent years, matrix-valued time series have been spreading rapidly, and factor analysis techniques applied to them are considered the new frontier in macroeconometrics. Their rapid diffusion among practitioners is largely justified by their strong empirical performance, often outperforming their vector or vectorized counterparts.

Dynamic Factor Models (DFMs) in vector form can be extended to the matrix case, as shown in Matteo Barigozzi and Trapin 2025. This structure is able to capture the bulk of the dynamics of the time series by accounting for both the cross-sectional and spatial correlation among entries over time. This new dimension makes the model and estimation strategy more sophisticated, and for this reason the following section will appropriately discuss the model, by comparing its structure to its vectorized counterpart in subsection 3.1.1, the loglikelihood mispecification and the estimation strategy for the its maximization through the EM-alghoritm in subsection 3.2.1. As an iterative procedure it allows tyo overcome the no closed form solution of the quasi log likelihood. EM algorithm described in 3.2.2 is then studied in all its steps starting from the initialization in subsection 3.2.3to the convergence in subsection 3.2.4. Finally also an extension to Missing value case odf the em algorithm is provided in subsection 3.3. Finally a methodological subsection on nowcasting 3.4is provided since it is the content of the empirical analysis and there is where the EM alghoritm is recursivelly implemented to estimate a pseudo real time data flow in a recursive nowcasting framework.

3.1 The Model

Nowadays, the high dimensionality of datasets represents a significant challenge for central banks and researchers. Factor models are identified as one of the main techniques to address this issue due to their ability to summarize the large amount of information through a reduced set of unobserved latent components, namely the factors.

In this work, I consider two different specifications of factor models, accounting also for their dynamics over time: the traditional Dynamic Factor Model (*DFM*) in vector form, and its generalization to the matrix case, namely the Dynamic Matrix Factor Model (*DMFM*). For practitioners, the way data are collected can strongly influence the choice between these two classes of dynamic factor models. In particular, as addressed in this work, even if these models share common features and can be estimated using analogous techniques, the matrix formulation may offer specific advantages. Indeed, it is becoming increasingly appealing in macroeconomic and financial contexts, where data are often collected as a sequence of variables for multiple units observed over time.

In this initial methodological section, I describe the DFM by comparing its vector and matrix formulations, emphasizing first the strengths of the more general matrix case, the *DMFM*, and then providing a brief review of the standard vector-based approach.

3.1.1 Dynamic Matrix Factor Model

Consider a Matrix Factor Model (*MFM*) for the $p_1 \times p_2$ centered and standardized matrix-valued stationary process $\{Y_t\}$. Without loss of generality, assume that the latent factors follow a Matrix Autoregressive (MAR) process of order $P = 1$. Formally, for any $t \in \mathbb{Z}$, the model is given by:

$$Y_t = RF_tC^\top + E_t, \tag{3}$$

$$F_t = AF_{t-1}B^\top + U_t, \tag{4}$$

According to this equation system at each time t, the data are collected in matrix form, resulting in a bi-dimensional observed data matrix $Y_t \in \mathbb{R}^{p_1 \times p_2}$. For now, assume that at each point in time the observed data matrix contains no missing observations. However, the framework, as presented, estimated and tested in this work, can be easily extended to account for missing values through appropriate methodological modifications, which are discussed later in Subsection 3.3.

The model is represented in a state-space form, where the measurement equation, namely Equation (3), corresponds to the formulation of the Matrix Factor Model by Wang, Liu, and R. Chen 2019, and the transition equation, namely Equation (4), describes the Matrix Autoregressive (*MAR*) process introduced by R. Chen, Xiao, and Yang 2021. The key idea is to retrieve the observed variables as a combination of unobserved factors, which are also allowed to evolve over time through a dynamic process. In high-dimensional settings, this strategy can definitely be a winning one, as the latent structure enables the condensation of a large amount of information into a few common components, directly mitigating the risk of incurring in the so-called curse of dimensionality. More formally, consider Equation (3) as a data-generating process consisting of a low-rank common component $\chi_t = RF_tC^\top$ and an idiosyncratic component $E_t \in \mathbb{R}^{p_1 \times p_2}$. On one hand, the common component χ_t is composed by $F_t \in \mathbb{R}^{k_1 \times k_2}$, the latent factor matrix, with $k_1, k_2 < \min(p_1, p_2)$, which drives the evolution of the matrix through row and column loading matrices $R \in \mathbb{R}^{p_1 \times k_1}$ and $C \in \mathbb{R}^{p_2 \times k_2}$, respectively. The loading matrices capture how each row and column is affected by the corresponding latent factors. This structure not only reduces dimensionality but also improves interpretability by allowing the identification the sources of variation through the loadings. In other words through this procedure it is straightforward to assign a name to the unobserved latent variable by analyzing their impact on loading matrices. On the other hand, the idiosyncratic component $E_t \in \mathbb{R}^{p_1 \times p_2}$ is characterized by a row covariance matrix $H \in \mathbb{R}^{p_1 \times p_1}$ and a column covariance matrix $K \in \mathbb{R}^{p_2 \times p_2}$. Note that even if no explicit dynamic model is assumed for the evolution of E_t, such extensions are possible. For consistent parameter estimation, it is assumed that $\{F_t\}$ and $\{E_t\}$ are uncorrelated at all leads and lags. It is worth noting that the idiosyncratic component will play a central role in estimation, particularly during the Log-Likelihood specification, due to the potential complexity introduced by the full structure of its covariance matrices. By now, let's consider the *DMFM* discussed so far as an *approximate* factor model, as it allows for full (non-diagonal) covariance matrices H and K. A similar structure applies also to Equation (4), which models the temporal dynamics of the latent factors. Indeed, the transition equation introduces time dependence among the factors F_t through the Matrix Autoregressive (*MAR*) structure. Also in this equation the bilinear form is preserved, with $A \in \mathbb{R}^{k_1 \times k_1}$ and $B \in \mathbb{R}^{k_2 \times k_2}$ denoting the autoregressive coefficient matrices. $U_t \in \mathbb{R}^{k_1 \times k_2}$ represents the innovation matrix, which has row covariance matrix $P \in \mathbb{R}^{k_1 \times k_1}$ and column covariance matrix $Q \in \mathbb{R}^{k_2 \times k_2}$.

The bilinear structure, expressed through the loading matrices R and C, and the covariance matrices H and K in the measurement equation, along with the autoregressive matrices A and B and the innovation covariance matrices P and Q in the transition equation, is the reason why the matrix representation of the Dynamic Factor Model can be seen as a generalization of the vector case. In fact, when the dimensionality is reduced to a single row or column, the matrix formulation simplifies to the vector version. Accounting for the matrix structure is especially appealing for macroeconomic and financial applications, as this methodology allows one to account for both time and cross-sectional dependence across rows and columns jointly, while also preserving the natural structure in which the data are collected.

The Vectorization: The *DMFM* presented above can be vectorized, as described in Section 7.2 in Durbin and Koopman 2012, to facilitate the estimation of the parameters via Quasi Maximum Likelihood (*QML*), by maximizing the prediction error decomposition of the Gaussian likelihood obtained from the Kalman filter. Unfortunately, the non trivial cost of vectorization of the DMFM is loosing the bilinear structure, that is exactly what this work aims to address, and increasing the number of parameters to estimate, that would make the estimation unfeasible. To show this let's consider the Vectorized form of the *DMFM* presented in Equations 3 and 4:

$$
\begin{aligned}
y_t &= (C \otimes R) f_t + e_t, \\
f_t &= (B_1 \otimes A) f_{t-1} + u_t,
\end{aligned}
$$

where $y_t = \mathrm{vec}(Y_t)$ is the vectorized observed data at time t, resulting from f_t, the vectorized matrix factor at time t. $C \otimes R$ is the Kronecker product of the column and row loading matrices, and $B_1 \otimes A$ is the transition matrix. Seemingly, $K \otimes H$ is the measurement error covariance matrix in e_t, and $Q \otimes P$ is the state error covariance matrix in u_t.

While the *DMFM* described above generates a total number of row and column loading parameters equal to $p_1 k_1 + p_2 k_2$, the corresponding vectorized *DMFM* requires estimating a full loading matrix of dimension $(p_1 p_2) \times (k_1 k_2)$. This loss of the bi-dimensional structure reduces interpretability and makes estimation more challenging, especially in high-dimensional settings. However, this vectorized form of the *DMFM*, as I will explain more formally later on, is particularly useful during some steps of the estimation procedure. In particular, during the Kalman Filter procedure in the E-step (Subsubsection 3.2.2), the vectorized form of the *MAR* parameters is used. This adaptation does not compromise the results, since the focus is not on the autoregressive parameters and the related innovation covariances. Nevertheless, it is useful for running the Kalman smoother without resorting to matrix-based versions of the Kalman filter for matrix state-space models, which offer only negligible computational advantages. Therefore, vectorization will be used solely to exploit the estimators obtained from the vectorized *MAR*, as in Matteo Barigozzi and Trapin 2025 and R. Yu et al. 2024.

DMFM Identification A critical aspect of the Matrix Factor Model, as introduced by Wang, Liu, and R. Chen 2019, concerns its identifiability, that is, the impossibility of uniquely estimating the single parameters diven the data without imposing additional restrictions. Identifiability is not an issue for forecasting purposes, because, as also noted by Bańbura and Modugno 2014, in such cases the primary interest is identifying the space spanned by the latent factors rather than uniquely estimating the individual factors themselves. In other words, the objective is estimate the product that regardless the single parameters produce the same combination. Nevertheless, addressing identifiability is essential to offer a rigorous theoretical understanding of the model and to correctly implement the estimation procedures proposed by R. Yu et al. 2024, used in this work as an initialization step for the EM-Algorithm.

The identifiability issue specifically affects the loading matrices R and C, as well as the factor matrix F_t, in the Matrix Factor Model (Equation (3)). These parameters are not separately identifiable because the product $RF_tC^\top$ remains unchanged under certain transformations involving invertible matrices $W_1 \in \mathbb{R}^{k_1 \times k_1}$ and $W_2 \in \mathbb{R}^{k_2 \times k_2}$, as shown below:

$$RF_tC^\top = (RW_1)(W_1^{-1}F_tW_2^{-1})(CW_2)^\top.$$

From this equation, it is clear that a complete ambiguity underlies the identifiability problem. Both the row and column loading matrices can be arbitrarily transformed using invertible matrices. In particular, for any invertible matrix W_1 the transformation on R does not change the product since $RW_1 \cdot W_1^{-1} = R$, and if W_2 is orthogonal, i.e., $W_2^\top = W_2^{-1}$ also the transformation of C leaves the product unchanged. This implies that further constraints are needed to uniquely identify the expression $RF_tC^\top$ from the observed data, as multiple combinations of R, F_t, and C can produce the same observed matrix.

A common solution to this issue in the *MFM* framework is to enforce identifiability by imposing orthonormality constraints on the loading matrices, as proposed by L. Yu et al. 2022:

$$\left\| \frac{1}{p_1} R^\top R - I_{k_1} \right\| \to 0, \quad \left\| \frac{1}{p_2} C^\top C - I_{k_2} \right\| \to 0. \tag{5}$$

By imposing these conditions, arbitrary invertible transformations are no longer permitted because, trough orthonormality constraints, the equivalence among multiple representations is broken, as any multiplication by a non-orthogonal matrix would violate the imposed conditions. This is why the model is said to be identifiable *up to orthogonal rotations*. This, in turn, allows for the estimation of a ”unique” and more interpretable set of parameters. It is worth noting that even under orthonormality constraints, the model remains identifiable also *up to the sign* of the factors and loadings. Inverting the sign of columns in R and C simultaneously leads to the same observed product $RF_tC^\top$.

To implement this constraint, the loading matrices can be decomposed as:

$$R = \sqrt{p_1} Q_1 W_1, \quad C = \sqrt{p_2} Q_2 W_2,$$

where $Q_1 \in \mathbb{R}^{p_1 \times k_1}$ and $Q_2 \in \mathbb{R}^{p_2 \times k_2}$ have orthonormal columns, meaning that each column has unit norm and the scalar product between different columns is zero [2]. The matrices W_1 and W_2 are full-rank, i.e., invertible square matrices. Finally, the scaling factors $\sqrt{p_1}$ and $\sqrt{p_2}$ are included to satisfy the normalization condition. When multiplying by any matrix W, it is now necessary to respect the constraint structure. The decomposition above implies:

$$\frac{1}{p_1} R^\top R = W_1^\top W_1, \quad \frac{1}{p_2} C^\top C = W_2^\top W_2.$$

If we additionally assume that W_1 and W_2 are orthogonal, an assumption now possible through the decomposition, then $W_i^\top W_i = I$, ensures that the orthonormality constraints hold exactly. These orthonormality assumptions are not only useful for identification purposes, but also help reduce the number of free parameters in the model and support the structural framework in which latent factors are assumed to be pervasive. In particular, orthonormality ensures that all columns of the loading matrices have unit norm and are mutually orthogonal, which facilitates both interpretation and estimation. When combined with the verifiable assumption that the loadings are distributed across many variables, this structure implies that each factor influences a relevant portion of the data matrix. This is especially useful in high-dimensional settings, where the goal is for the extracted common component to capture dynamics across many variables.

3.1.2 Dynamic Factor Model

Although this work builds upon the matrix formulation of the Dynamic Factor Model theory, namely the *DMFM*, the most widely used *DFM* are those in vector form, which can be viewed as a limiting case of the former. By reducing the number of rows to $p_1 = 1$ and the number of row factors to $k_1 = 1$, or alternatively the number of columns to $p_2 = 1$ and the number of column factors to $k_2 = 1$, the matrix structure collapses into a vector structure. The resulting model is a standard *DFM* that shares many of the properties described previously. Nothing prevents this model from being specified as an approximate dynamic factor model, but it lacks the bilinear formulation that is characteristic of the DMFM.

Let's consider a data-generating process analogous to the one in the DMFM, now applied to a vector context. The observed time series y_{nt} is an n-dimensional stationary vector process, standardized to have zero mean and unit variance. The latent factors are assumed to follow a Vector Autoregressive (*VAR*) process of order $P = 1$. The problem can be formulated through a state-space representation where the measurement equation links the observed data to the latent factors (Equation 6), and a transition equation drives the evolution of the factors through an autoregressive process (Equation 7):

[2] Unit norm means that in each column the weight is standardized across the observed variables, i.e., $\sum_i R_{i,j}^2 = 1$ allowing for column loadings comparison; orthogonality implies that each factor captures different information, that in other words means that is geometrically independent from the others.

$$y_{nt} = \Lambda_n F_t + \xi_{nt}, \qquad t = 1, \dots, T \tag{6}$$
$$F_t = AF_{t-1} + v_t, \qquad v_t \sim \mathcal{N}(0, \Gamma_v) \tag{7}$$

In Equation 6, $y_{nt} \in \mathbb{R}^n$ is modeled as the sum of the common component composed of the factor loading matrix $\Lambda_n \in \mathbb{R}^{n\times r}$ and the latent factor vector $F_t \in \mathbb{R}^r$ and the idiosyncratic noise, that is ξ_{nt}, which may exhibit weak cross-sectional correlation with other series. Equation 7 introduces dynamics by specifying a *VAR* process for the latent factors, with autoregressive coefficients captured by matrix $A \in \mathbb{R}^{r\times r}$ and innovation $v_t \sim \mathcal{N}(0, \Gamma_v)$.

DFM vs. DMFM: Model Comparison

Dynamic Factor Model (DFM)

Observation equation:

$$y_t = \Lambda F_t + \xi_t$$

Transition equation:

$$F_t = AF_{t-1} + v_t$$

Key characteristics:

- Vector-valued data: $y_t \in \mathbb{R}^n$
- Factor loadings: $\Lambda \in \mathbb{R}^{n\times r}$
- Approximate model (idiosyncratic errors)

Dynamic Matrix Factor Model (DMFM)

Observation equation:

$$Y_t = RF_tC^\top + E_t$$

Transition equation:

$$F_t = AF_{t-1}B^\top + U_t$$

Key characteristics:

- Matrix-valued data: $Y_t \in \mathbb{R}^{p_1\times p_2}$
- Bilinear structure with loadings: R, C
- Allows for row/column covariance: H, K

3.2 Estimation

In this subsection, I present the strategy followed for the estimation of the approximate Dynamic Matrix Factor Model (*DMFM*) discussed in Subsection 3.1.1. Borrowing the methodology by Matteo Barigozzi and Trapin 2025, the parameters of the *DMFM* are estimated jointly via Quasi Maximum Likelihood (*QML*) implementing the Expectation-Maximization (*EM*) algorithm, where The Kalman smoother provides preliminary forecasts and updated estimates of the latent factors and their covariances. As specifically addressed in Subsection 3.4, Kalman Filtering techniques will also plays a fundamental role for nowcasting, a recent and increasingly popular forecasting methodology.

Moreover, this framework is particularly appealing also for handling general patterns of missing data, characterizing the EM-algorithm as an approached able to address several layers of generality. In this case, the primary interest is on mixed frequency datasets, where the estimation strategy is already been successfully tested by Bańbura and Modugno 2014, who apply the EM-algorithm allowing for missing observations in the vector-valued case. Since, as already discussed, *DFM* in vector form can be considered a particular case of the matrix formulation, see Table 1, the matrix case is just a generalization of the approach. It comes without saying that the flexibility to accommodate mixed-frequency datasets and recursive nowcasting exercises for data collected in matrix form makes this approach particularly desirable for empirical forecasting applications, especially in central banks.

In order to estimate the parameters of the *DMFM* one needs to maximize the model's gaussian likelihood function. However, the first obstacle to face for the model as originally specified is it's approximate nature. The fully specified idiosyncratic covariance matrix makes the standard Maximum Likelihood estimation computationally infeasible because the number of parameters to estimate growth rapidly at rate of $\mathscr{O}((p_1^2+p_2^2)T)$. To reduces drastically the number of parameters the log-likelihood is intentionally mis-specified by imposing a diagonal structure on the idiosyncratic component, meaning restricting the cross-sectional and serial correlation.

Now that the problem is definitely more tractable, a second challenge has to be addressed, namely the no admitted closed-form solution of the resulting quasi-log-likelihood. In this step the EM algorithm plays a vital role since it is used to iteratively compute closed-form updates for all model parameters. From the estimates of the latent factors obtained through the Kalman smoother, the E-step reconstructs a "complete" dataset and the M-steps provides new estimates of the model's parameters. The iteration proceeds until the quasi-log-likelihood stabilizes around a local maximum given the model and observed data.

The following subsections are a rigorous discussion of this estimation strategy, with particular emphasis on the implementation of the EM algorithm, along with the initialization procedure and the convergence criteria adopted in the empirical analysis, as inspired by Matteo Barigozzi and Trapin 2025.

3.2.1 Log-Likelihood

The *DMFM* defined in Equations (3) and (4) implies the following covariance structure for the data:

$$\Omega_{Y_T} = (I_T \otimes C \otimes R)\,\Omega_{F_T}(A,B,P,Q)\,(I_T \otimes C \otimes R)^\top + \Omega_{E_T},$$

where the first term captures the systematic component, and the second term represents the idiosyncratic noise.

Estimating the full covariance matrix $\Omega_{E_T} = \text{Cov}(E_1, \dots, E_T)$ without restrictions is infeasible since the total number of observations available in the data are $T \cdot p_1 p_2$ while the full estimation of Ω_{E_T}, assuming cross-sectional and serial correlation, would require $(p_1 p_2)^2 T$ free parameters [3]. Indeed, assuming also autocorrelation of E_t the number of parameters in Ω_{E_T} would grow as

$$\frac{p_1^2 p_2^2 T (p_1^2 p_2^2 T + 1)}{2},$$

To overcome this issue, Matteo Barigozzi and Trapin 2025 exclude both cross-sectional and serial correlation of E_t, reducing the number of parameters to estimate from $\mathcal{O}((p_1 p_2 T)^2)$ to $\mathcal{O}(p_1 p_2 T)$. Through this assumption, the estimation becomes feasible since the idiosyncratic covariance structure reduces to:

$$\Omega_{E_T} \approx I_T \otimes \text{diag}(K) \otimes \text{diag}(H),$$

The quasi-log-likelihood function under this mis-specification of Ω_{E_T} becomes:

$$\begin{aligned} \ell(Y_T;\theta) = -\frac{p_1 p_2 T}{2}\log(2\pi) - \frac{1}{2}\log\left|(I_T \otimes C \otimes R)\Omega_{F_T}(I_T \otimes C \otimes R)^\top + I_T \otimes \text{diag}(K) \otimes \text{diag}(H)\right| \\ -\frac{1}{2} Y_T^\top \left[(I_T \otimes C \otimes R)\Omega_{F_T}(I_T \otimes C \otimes R)^\top + I_T \otimes \text{diag}(K) \otimes \text{diag}(H)\right]^{-1} Y_T. \end{aligned} \quad (8)$$

Where Ω_{F_T} is the covariance matrix of the vectorized latent factor process that depends on the matrices $A \in \mathbb{R}^{k_1 \times k_1}$, $B \in \mathbb{R}^{k_2 \times k_2}$, and the innovation covariances $P \in \mathbb{R}^{k_1 \times k_1}$ and $Q \in \mathbb{R}^{k_1 \times k_1}$.

The scope is to find the maximizer of this quasi-log-likelihood, defined as the Quasi-Maximum Likelihood (*QML*) estimator. Through the mis-specification of Ω_{E_T} it became computationally feasible even in high-dimensional settings. indeed, by imposing a diagonal structure, ruling out any form of correlation, the number of parameters to estimate decreased drastically.

3.2.2 EM Algorithm

So far, it is clear that the estimation of the DMFM parameters is not feasible using standard Maximum Likelihood Estimation. To overcome this issue, the first step to pursue a consistent estimation of the DMFM parameters is described the previous subsection. By imposing a diagonal structure on the idiosyncratic covariance matrix Ω_{FT} it is possible to restrict the its cross-sectional and serial correlation. Indeed, even if the model is assumed to be an approximate dynamic matrix factor model, to manage the high-dimensional structure, where parameters to estimate grows as $(p_1 \cdot p_2)^2 \cdot T$, this approach allows to estimate consistently just to the observed data $p_1 \cdot p_2 \cdot T$.

This step implies a mis-specification of the log-likelihood, represented by Equation 3.2.4. However, this more tractable Quasi-Log-Likelihood, poses another fundamental challenge to address, namely the absence of a closed-form solution. In general, a log-likelihood function for multivariate Gaussian data

[3] To compute Ω_{E_T}, vectorize E_t to obtain $\Omega^E := \text{Var}(\text{vec}(E_t)) \in \mathbb{R}^{(p_1 p_2) \times (p_1 p_2)}$. As a symmetric matrix, the number of free parameters amount to $(p_1 p_2)(p_1 p_2 + 1)/2 = \mathcal{O}(p_1^2 p_2^2)$, for each $t = 1, \dots, T$.

admits a closed-form expression only when it can be written explicitly with respect to the parameters and in this case neither the factor covariance matrix Ω_T^F nor the observation covariance matrix Ω_Y can be expressed in closed form. Indeed, as a first observation, let's say that the quasi-log-likelihood cannot be maximized analytically because the factor covariance matrix Ω_T^F depends implicitly on the unknown parameters θ, and since the latent factors F_t are unobserved, it is not possible to substitute this object into the likelihood directly. Moreover, even assuming the factors were known, the structure of the observation covariance matrix Ω_Y still prevents the derivation of closed-form expressions for its determinant and inverse. Ω_Y is indeed constructed as a sum of Kronecker products, that do not admit algebraic simplifications. In particular, $\log|\Omega_Y|$ and Ω_Y^{-1} cannot be computed analytically.

A common solution to address the absence of a closed-form quasi maximum likelihood estimator (QMLE), is implement an iterative estimation strategy such as the Expectation-Maximization (EM) algorithm. Through the EM-Algorithm it is possible first to compute partial estimates of latent variables and then optimizing conditionally on those estimates. The procedure involves two steps:

- In the **E-step** (Expectation), given a current estimate of the parameters $\theta^{(q)}$, the Kalman filter followed by the Kalman smoother provides estimates of the latent factors, specifically the conditional expectation $\hat{F}_t = \mathbb{E}[F_t \mid Y, \theta^{(q)}]$ and the conditional covariance $\mathbb{V}[F_t \mid Y, \theta^{(q)}]$. These quantities effectively reconstruct the missing information, allowing the algorithm to operate with a "complete" dataset that includes the factors.

- In the **M-step** (Maximization), using the quantities obtained from the E-step, the expected value of the complete-data log-likelihood is constructed and maximized with respect to the parameters θ. Although the factors are estimated conditionally on the observed data Y, the function can now be optimized with respect to all remaining parameters of the DMFM.

From any iteration one gains a new set of estimated parameters $\theta^{(q+1)}$ that have to be used as the benchmark for the next iteration. Repenting this process until convergence, namely until the observed log-likelihood stabilizes, means reaching a local maximum of the quasi-log-likelihood.

In the following paragraphs I describe rigorously these two steps along with a formal description of the Kalman Filtering techniques.

Kalman Filter and Smoother : Kalman filtering techniques, originally developed in physics and engineering, have been widely adopted in econometrics due to their usefulness in forecasting and state estimation. During the estimation procedure, for each EM iteration $n \geq 0$, and given the current parameter estimates $\hat{\Theta}^{(n)}$, the Kalman filter and smoother are employed to retrieve the key conditional expectations required in the E-step. Specifically, this procedure yields the expected value of the latent vectorized factor $f_t = \text{vec}(F_t)$, conditional on the full sample: $f_{t|T} = \mathbb{E}[f_t \mid Y_{1:T}; \hat{\Theta}^{(n)}]$, along with the corresponding conditional covariances.

As described in Section 3.1.1, the dynamic matrix factor model (DMFM) is vectorized while maintaining its state-space form for $y_t = \text{vec}(Y_t) \in \mathbb{R}^{p_1 p_2}$:

$$
\begin{aligned}
y_t &= Zf_t + e_t \\
f_t &= Tf_{t-1} + u_t
\end{aligned}
$$

where $Z = C \otimes R$ and $T = A \otimes B$.

Using some initial conditions $f_{0|0}$ and $P_{0|0}$, typically assumed to be zero and the identity matrix respectively, the Kalman filter proceeds forward in time accounting for some selection matrix to handle missing data at each time step.

For each time period $t = 1, \dots, T$, the filter performs a prediction step and an update step whenever the observations are available. These steps can be shown mathematically as follows:

1. *Prediction step*:

$$f_{t|t-1} = Tf_{t-1|t-1}, \qquad P_{t|t-1} = TP_{t-1|t-1}T^\top + Q$$

2. *Update step*:

$$S_t = Z_t P_{t|t-1} Z_t^\top + H_t, \qquad K_t = P_{t|t-1} Z_t^\top S_t^{-1}$$

$$f_{t|t} = f_{t|t-1} + K_t(y_t - Z_t f_{t|t-1}), \qquad P_{t|t} = P_{t|t-1} - K_t Z_t P_{t|t-1}$$

Subsequently, the Kalman smoother performs a backward recursion to refine the state estimates. The smoothed values are defined as:

$$
\begin{aligned}
f_{t|T} &= f_{t|t} + P_{t|t} r_t \\
P_{t|T} &= P_{t|t} - P_{t|t} N_t P_{t|t}
\end{aligned}
$$

where the smoothing terms are recursively determined as:

$$
\begin{aligned}
r_t &= Z_t^\top S_t^{-1}(y_t - Z_t f_{t|t-1}) + L_t^\top r_{t+1} \\
N_t &= Z_t^\top S_t^{-1} Z_t + L_t^\top N_{t+1} L_t
\end{aligned}
$$

The smoothed estimates $(f_{t|T}, P_{t|T})$ are then used in the E-step to compute the conditional expectations of the data and factors required to update the parameters in the M-step of the EM algorithm.

E-step: Once the Kalman smoother has provided the smoothed estimates of the conditional first and second moments of the latent factors, the E-step of the EM algorithm computes the expected Gaussian quasi-log-likelihood of the approximate DMFM. Using current parameter estimates $\hat{\theta}^{(n)}$, the observed data log-likelihood can be decomposed applying Bayes' rule as follows:

$$\ell(Y_T;\theta) = \underbrace{\mathbb{E}_{\hat{\theta}^{(n)}}\left[\ell(Y_T \mid F_T;\theta) \mid Y_T\right]}_{\substack{\text{Expected log-likelihood} \\ \text{of the observed data}}} + \underbrace{\mathbb{E}_{\hat{\theta}^{(n)}}\left[\ell(F_T;\theta) \mid Y_T\right]}_{\substack{\text{Expected log-likelihood} \\ \text{of the latent states}}} - \underbrace{\mathbb{E}_{\hat{\theta}^{(n)}}\left[\ell(F_T \mid Y_T;\theta) \mid Y_T\right]}_{\text{Conditional entropy}}$$

The first member captures the ability of the current parameters to explain the observed data if the factors were known, the second member describes the consistency of the current parameters with the temporal structure of the unobserved states, and the last member is practically a correction term. As shown by Dempster, Laird, and Rubin 1977 and, since this model belongs to the exponential family, the last term can be excluded from the maximization of the observed-data log-likelihood $\ell(Y_T;\theta)$.

For this reason, the E-step focuses on computing the so-called *Q-function*, defined as:

$$Q(\theta,\hat{\theta}^{(n)}) = \mathbb{E}_{\hat{\theta}^{(n)}}\left[\ell(Y_T \mid F_T;\theta) \mid Y_T\right] + \mathbb{E}_{\hat{\theta}^{(n)}}\left[\ell(F_T;\theta) \mid Y_T\right]$$

Therefore, the M-step of the EM algorithm focuses only on maximizing this expected complete-data log-likelihood with respect to θ.

Specifically, the Expected log-likelihood of the observed data has the following form:

$$\begin{aligned}\mathbb{E}_{\hat{\theta}^{(n)}}[\ell(Y_T|F_T;\theta) \mid Y_T] = &-\frac{T}{2}\left(p_1 \log|K| + p_2 \log|H|\right)\\ &-\frac{1}{2}\sum_{t=1}^{T}\mathbb{E}_{\hat{\theta}^{(n)}}\left[\operatorname{tr}\left(H^{-1}(Y_t - RF_tC^\top)K^{-1}(Y_t - RF_tC^\top)^\top\right)\Big| Y_T\right]\end{aligned} \tag{9}$$

while the Expected log-likelihood of the latent states is described by:

$$\begin{aligned}\mathbb{E}_{\hat{\theta}^{(n)}}[\ell(F_T;\theta) \mid Y_T] = &-\frac{T-1}{2}\left(k_1 \log|Q| + k_2 \log|P|\right)\\ &-\frac{1}{2}\sum_{t=1}^{T}\mathbb{E}_{\hat{\theta}^{(n)}}\left[\operatorname{tr}\left(P^{-1}(F_t - AF_{t-1}B^\top)Q^{-1}(F_t - AF_{t-1}B^\top)^\top\right)\Big| Y_T\right]\end{aligned} \tag{10}$$

It is worth noting that both these log-likelihoods depend directly on the data in its matrix form.

M-step: In the M-step, Equations (9) and (10), derived using the current parameter estimates $\hat{\theta}^{(n)}$, are maximized to obtain an updated set of parameters $\hat{\theta}^{(n+1)}$. At each iteration $n \geq 0$, new estimates of the DMFM parameters are computed, beginning with the row and column loading matrices, denoted by $\hat{R}^{(n+1)}$ and $\hat{C}^{(n+1)}$, respectively.

The row loadings are updated as follows:

$$\hat{R}^{(n+1)} = \left(\sum_{t=1}^{T} Y_t \hat{K}^{(n)-1}\hat{C}^{(n)}\hat{F}_{t|T}^{(n)\prime}\right)\left(\sum_{t=1}^{T}\left(\hat{C}^{(n)\prime}\hat{K}^{(n)-1}\hat{C}^{(n)}\right)\star\left(\hat{F}_{t|T}^{(n)}\hat{F}_{t|T}^{(n)\prime} + \Pi_{t|T}^{(n)}\right)\right)^{-1} \tag{11}$$

while the column loadings are updated via:

$$\hat{C}^{(n+1)} = \left(\sum_{t=1}^{T} Y_t^\top \hat{H}^{(n)-1}\hat{R}^{(n+1)}\hat{F}_{t|T}^{(n)}\right)\left(\sum_{t=1}^{T}\left(\hat{R}^{(n+1)\prime}\hat{H}^{(n)-1}\hat{R}^{(n+1)}\right)\star\left(K_{k_1k_2}\left(\hat{F}_{t|T}^{(n)}\hat{F}_{t|T}^{(n)\prime}+\Pi_{t|T}^{(n)}\right)K_{k_1k_2}^{\prime}\right)\right)^{-1} \tag{12}$$

Since the estimation of R and C depends on each other, the empirical strategy adopted is to first compute $\hat{R}^{(n+1)}$ conditionally on $\hat{C}^{(n)}$, and then update $\hat{C}^{(n+1)}$ based on the newly estimated $\hat{R}^{(n+1)}$. However, the reverse order would also be valid. As shown in the next subsection, these estimators are consistent at every iteration $n \geq 0$, thus preserving the bilinear structure of the DMFM.

Given $\hat{R}^{(n+1)}$ and $\hat{C}^{(n+1)}$, we can now estimate the idiosyncratic covariance matrices $\hat{H}^{(n+1)}$ and $\hat{K}^{(n+1)}$. As discussed in Section 3.2.1, these matrices are assumed to be diagonal, consistent with the quasi-log-likelihood specification in Equation 3.2.4. For $i = 1,\dots,p_1$, with $[\hat{H}^{(n+1)}]_{ij} = 0$ for $i \neq j$, $[\hat{H}^{(n+1)}]_{ij}$ is computed as:

$$\begin{aligned}
[\hat{H}^{(n+1)}]_{ii} = \frac{1}{Tp_2}\sum_{t=1}^{T}\Big[& Y_t\hat{K}^{(n)-1}Y_t^\top - Y_t\hat{K}^{(n)-1}\hat{C}^{(n+1)}\hat{F}_{t|T}^{(n)\prime}\hat{R}^{(n+1)\prime} - \hat{R}^{(n+1)}\hat{F}_{t|T}^{(n)}\hat{C}^{(n+1)\prime}\hat{K}^{(n)-1}Y_t^\top \\
& + \left(\hat{C}^{(n+1)\prime}\hat{K}^{(n)-1}\hat{C}^{(n+1)}\right)\star\left((I_{k_2}\otimes\hat{R}^{(n+1)})(\hat{F}_{t|T}^{(n)}\hat{F}_{t|T}^{(n)\prime}+\Pi_{t|T}^{(n)})(I_{k_2}\otimes\hat{R}^{(n+1)})^{\prime}\right)\Big]_{ii}
\end{aligned} \tag{13}$$

Similarly, for $i = 1,\dots,p_2$, with $[\hat{K}^{(n+1)}]_{ij} = 0$ for $i \neq j$, $[\hat{K}^{(n+1)}]_{ij}$ takes the form of :

$$\begin{aligned}
[\hat{K}^{(n+1)}]_{ii} = \frac{1}{Tp_1}\sum_{t=1}^{T}\Big[& Y_t^\top\hat{H}^{(n+1)-1}Y_t - Y_t^\top\hat{H}^{(n+1)-1}\hat{R}^{(n+1)}\hat{F}_{t|T}^{(n)}\hat{C}^{(n+1)\prime} - \hat{C}^{(n+1)}\hat{F}_{t|T}^{(n)\prime}\hat{R}^{(n+1)\prime}\hat{H}^{(n+1)-1}Y_t \\
& + \left(\hat{R}^{(n+1)\prime}\hat{H}^{(n+1)-1}\hat{R}^{(n+1)}\right)\star\left((I_{k_1}\otimes\hat{C}^{(n+1)})K_{k_1k_2}(\hat{F}_{t|T}^{(n)}\hat{F}_{t|T}^{(n)\prime}+\Pi_{t|T}^{(n)})K_{k_1k_2}^{\prime}(I_{k_1}\otimes\hat{C}^{(n+1)})^{\prime}\right)\Big]_{ii}
\end{aligned} \tag{14}$$

As discussed for the loadings, the estimation of $\hat{H}$ and $\hat{K}$ depend on each other. To maintain the bilinear structure of the model, $\hat{H}^{(n+1)}$ is computed first, conditional on $\hat{K}^{(n)}$.

Following Matteo Barigozzi and Trapin 2025, this work uses autoregressive matrices A and B and innovation covariance matrices P and Q solely for the implementation of the Kalman smoother on vectorized data. Therefore, the transition and innovation matrices are estimated via the vectorized MAR model as:

$$\widehat{B\otimes A}^{(n+1)} = \left(\sum_{t=2}^{T}\hat{F}_{t|T}^{(n+1)}\hat{F}_{t-1|T}^{(n)\prime}+\Delta_{t|T}^{(n)}\right)\left(\sum_{t=2}^{T}\hat{F}_{t-1|T}^{(n)}\hat{F}_{t-1|T}^{(n)\prime}+\Pi_{t-1|T}^{(n)}\right)^{-1} \tag{15}$$

$$\widehat{Q \otimes P}^{(n+1)} = \frac{1}{T} \sum_{t=2}^{T} \left(\hat{F}_{t|T}^{(n)} \hat{F}_{t|T}^{(n)\prime} + \Pi_{t|T}^{(n)} - \left(\hat{F}_{t|T}^{(n)} \hat{F}_{t-1|T}^{(n)\prime} + \Delta_{t|T}^{(n)} \right) \left(\widehat{B \otimes A}^{(n+1)} \right)^{\prime} \right) \tag{16}$$

In this case, the estimators do not enforce the bilinear structure of the MAR model directly. However, since singular estimates of the dynamic parameters are not required, and the estimated loadings and factors are asymptotically unaffected by this simplification, this approach is adopted without loss of generality.

3.2.3 Parameters' Initialization

The EM algorithm, as an iterative process used for the maximization, needs a starting point, namely initial estimates of the parameters. As already discussed in subsection 3.2, the quasi-log-likelihood considered is not concave and during the maximization process it is likely to end up in a local maximum. Since the E-step depends on initial estimates of model's parameters, the more close they are to the global maximum, the more fast and efficient will be the convergence.

In order to face the challenge of the parameters initialization the discussion treats separately the *MFM's* parameters in Equation3 and the *MAR's* once in Equation 4. For the former the methodology adopted are the projected estimates (*PE*) proposed by L. Yu et al. 2022, already sufficient to obtain consistent initial estimates of the row and column loading, and factor matrices, while, for the latter the pre-estimators are given by the OLS estimates of the vectorized transition and innovation covariance matrices.

MFM Parameters: L. Yu et al. 2022 describe the empirical implementation of the projected estimator for large-dimensional matrix factor models. Although their focus is on the static matrix factor model, as initially introduced by Wang, Liu, and R. Chen 2019, their methodology can be extended also in a dynamic setting. This approach allows for the estimation of the number of row and column factors and the corresponding loading matrices. The technique is based on the eigenvalue decomposition of specific covariance matrices constructed from projections of the data onto the row and column spaces where the dimension is drastically reduced. Since this projection procedure is based on a PCA approach, it requires data imputation in presence of missing. This extension is discussed in Subsection 3.3.2 and it is inspired by the imputation method proposed by Cen and Lam 2025.

The discussion of the L. Yu et al. 2022's methodology presented in this section describes first the projected estimation method and its consistency. Then, focuses on the initial projections when both row and column spaces are unknown, and finally describes how to extract matrix factor loadings using principal components and how to determine the number of row and column factors via the eigenvalue ratio algorithm proposed by Lam and Yao 2012. For these two algorithms I provide a supplementary concise boxed summary.

Let's start from a matrix factor model without explicit factor dynamics as proposed by L. Yu et al. 2022. This means the model can be considered static, but this does not limit the applicability of their method to dynamic matrix factor models, such as the one used in this work. Given a column loading matrix C satisfying the orthogonality condition

$$||\frac{1}{p_2}C^\top C - I_{k_2}|| = 0,$$

as broadly discussed in subsection 3.1.1 it is possible to project the data matrix onto the column space. The result is this projected data matrix:

$$Y_t = \frac{1}{p_2}X_t C = \frac{1}{p_2}RF_t C^\top C + \frac{1}{p_2}E_t C = RF_t + \tilde{E}_t,$$

where $\tilde{E}_t = \frac{1}{p_2}E_t C$ is the transformed noise term. The appealing feature of this projection procedure is that it immediately provides a dimensionality reduction, through a shrinkage of the number of columns from p_2 to k_2. Moreover, since the variance of $\tilde{E}_t$ is of order $\mathcal{O}(1/p_2)$, the noise level decreases significantly when p_2 is large. This makes the projected data Y_t behave like a near noise-free factor model:

$$Y_t = RF_t + \tilde{E}_t.$$

Thus, the projection turns the matrix factor model into a standard vector factor model, with the idiosyncratic component that asymptotically vanishes as $p_2 \to \infty$.

Given the projected data, to implement a Principal component analysis, the next step is to construct the sample covariance matrix as:

$$\tilde{M}_1 = \frac{1}{Tp_1}\sum_{t=1}^{T} Y_t Y_t^\top,$$

and then extract the eigenvectors corresponding to the leading k_1 eigenvalues of M_1 to obtain a consistent estimator of the row loading matrix R. It is worth noting that the number of leading eigenvalue relies on the Algorithm 2 discussed in paragraph 3.2.3. Finally, to ensure that the signal does not diminish as dimensionality increases, the matrix R must be rescaled by $\sqrt{p_1}$. The specular discussion is true for C as the box 3.2.3 shows. At this point it is possible to Update the loadings, namely $\tilde{R}$ and $\tilde{C}$ that can be used as new initial values for a sequent estimation. However, as shown by L. Yu et al. 2022, one iteration is often sufficient for accurate estimation.

Algorithm 1: Projected Estimation

1. **Initial Estimation:** Obtain preliminary estimates $\hat{R}$ and $\hat{C}$, for example via PCA.

2. **Projection:** Define projected data:

$$\hat{Y}_t = \frac{1}{p_2} X_t \hat{C}, \quad \hat{Z}_t = \frac{1}{p_1} X_t^\top \hat{R}.$$

3. **Covariance Matrices:** Extract the leading k_1 and k_2 eigenvectors $\tilde{Q}_1$ and $\tilde{Q}_2$ from the Covariance Matrices:

$$\tilde{M}_1 = \frac{1}{Tp_1} \sum_{t=1}^{T} \hat{Y}_t \hat{Y}_t^\top, \quad \tilde{M}_2 = \frac{1}{Tp_2} \sum_{t=1}^{T} \hat{Z}_t \hat{Z}_t^\top.$$

4. **Loadings:** Update the loadings as:

$$\tilde{R} = \sqrt{p_1} \tilde{Q}_1, \quad \tilde{C} = \sqrt{p_2} \tilde{Q}_2.$$

From the pre-estimators of R and C the factor matrix one is obtained via linear projection:

$$\hat{F}_t = \frac{\hat{R}^{(0)\top} Y_t \hat{C}^{(0)}}{p_1 p_2}.$$

Now it is possible to compute residuals as:

$$\hat{E}_t^{(0)} = Y_t - \hat{R}^{(0)} \hat{F}_t \hat{C}^{(0)\top},$$

end extract pre-estimators of the noise covariance matrices H and K are given by:

$$\hat{K}^{(0)} = \frac{1}{Tp_1} \sum_{t=1}^{T} \operatorname{tr}\left(\hat{E}_t^{(0)\top} \hat{E}_t^{(0)}\right), \quad \hat{H}^{(0)} = \frac{1}{Tp_1} \sum_{t=1}^{T} \operatorname{tr}\left(\hat{E}_t^{(0)} \hat{K}^{(0)-1} \hat{E}_t^{(0)\top}\right). \tag{17}$$

where only the the diagonal terms are required to run the EM algorithm.

Since initial estimates of the column or row loadings are unknown, another step to initialize this algorithm is needed, and the following paragraph provides the methodology.

Initial Estimation Challenge: Since column loading matrix C (or equally R) is unknown and must be estimated from the data a natural approach to obtain some initial estimators, namely $\hat{C}$ (or seemingly $\hat{R}$).

To estimate $\hat{C}$, first transpose the data matrices $X_t^\top$ and treat the rows of X_t as observations from a vector factor model. The next step is to compute the sample covariance matrix:

$$\hat{M}_2 = \frac{1}{Tp_1p_2} \sum_{t=1}^{T} X_t^\top X_t.$$

and from this $p_2 \times p_2$ matrix $\hat{M}_2$, extract the leading k_2 eigenvectors to obtain $\hat{Q}_2$. The initial estimator of the column loading matrix is then given by:

$$\hat{C} = \sqrt{p_2}\hat{Q}_2.$$

Similarly, the row loading matrix $\hat{R}$ is estimated by applying the same procedure to X_t:

$$\hat{M}_1 = \frac{1}{Tp_1p_2}\sum_{t=1}^{T} X_t X_t^\top, \quad \hat{R} = \sqrt{p_1}\hat{Q}_1,$$

where $\hat{Q}_1$ contains the leading k_1 eigenvectors of $\hat{M}_1$.

This estimates are noisier then the ones obtained after projection but are sufficient to initialize the iterative procedure summarized in 3.2.3. Indeed, even the first projection improves the estimates significantly, reducing the estimation variance, and enhancing convergence properties.

Estimation of the Number of Row and Column Factors As mentioned during the discussion of Algorithm 1 in subsection 3.2.3 to be practically efficient one needs to determine the correct number of row and column factors. This step is crucial both for or projected estimation and for the factors' number considered for the EM algorithm.

The number of row and column factors, respectively denoted by k_1 and k_2, provide the dimensions of the latent factor matrix $F_t \in \mathbb{R}^{k_1 \times k_2}$, and consequently the number of columns in the row and column loading matrices, respectively $R \in \mathbb{R}^{p_1 \times k_1}$ and $C \in \mathbb{R}^{p_2 \times k_2}$.

k_1 and k_2 are estimated using the eigenvalue ratio criterion, originally proposed by Lam and Yao 2012, which is based on detecting sudden jumps between variance explained by eigenvectors in the covariance matrix resulted from the projection process. Specifically, k_1 is selected as the index j that maximizes the ratio between the j-th and $(j+1)$-th largest eigenvalues of the projected covariance matrix $\tilde{M}_1$, formally:

$$\hat{k}_1 = \arg\max_{j \le k_{\max}} \frac{\lambda_j(\tilde{M}_1)}{\lambda_{j+1}(\tilde{M}_1)},$$

where $\lambda_j(\tilde{M}_1)$ is the j-th largest eigenvalue, and $k_{\max}$ is a pre-specified upper bound. Once the first k_1 eigenvalues significantly larger than the rest are detected, one can use them to explain the substantial part of the variance. The corresponding factor will be retained and the others discarded.

Since small eigenvalues could affect the process, the strategy to stabilize the ratio is adding a regularization term to the denominator and some small constant c as follows:

$$\lambda_{j+1} + c\delta, \quad \text{with } \delta = \max\left\{\frac{1}{\sqrt{Tp_2}}, \frac{1}{\sqrt{Tp_1}}, \frac{1}{p_1}\right\},$$

In practice, estimating $\tilde{M}_1$ requires a preliminary estimate of the column loading matrix $\hat{C}$, which in turn depends on k_2. Similarly, estimating $\tilde{M}_2$ requires knowledge of $\hat{R}$ and k_1. To address this interdependency, another iterative algorithm, Algorithm 2 is proposed by L. Yu et al. 2022 and it is breifly summarized in box 3.2.3.

Algorithm 2: Number of Factors Estimation

1. **Initialization:** Set initial guesses:

$$\hat{k}_1^{(0)} = k_{\max}, \quad \hat{k}_2^{(0)} = k_{\max}.$$

2. **Iterative Updates:** For iterations $t = 1, \dots, m$, where m is the maximum number of iterations:

   (a) Given $\hat{k}_2^{(t-1)}$, estimate the column loading matrix $\hat{C}^{(t)}$ via PCA.

   (b) Compute the projected covariance matrix:

$$\tilde{M}_1^{(t)} = \frac{1}{Tp_1} \sum_{t=1}^{T} \hat{Y}_t \hat{Y}_t^\top.$$

   (c) Update $\hat{k}_1^{(t)}$ as:

$$\hat{k}_1^{(t)} = \arg\max_j \frac{\lambda_j(\tilde{M}_1^{(t)})}{\lambda_{j+1}(\tilde{M}_1^{(t)})}.$$

   (d) Seemingly, given $\hat{k}_1^{(t)}$, estimate the row loading matrix $\hat{R}^{(t)}$, compute $\tilde{M}_2^{(t)}$ and update $\hat{k}_2^{(t)}$

3. **Stopping Criterion:** Stop if m or the convergence is reached:

$$\hat{k}_1^{(t)} = \hat{k}_1^{(t-1)} \quad \text{and} \quad \hat{k}_2^{(t)} = \hat{k}_2^{(t-1)},$$

MAR Parameters: Once the latent factor matrices are obtained, the next step is to estimate the parameters of the Matrix Autoregressive (*MAR*) process used to model factors' dynamics. The goal is to obtain initial estimators of the transition matrices A and B, as well as the innovation covariance matrices P and Q, to use them for the Kalman filter and smoother.

The pre-estimator of the vectorized factors have the following form:

$$\tilde{f}_t = \frac{(\hat{C}^{(0)} \otimes \hat{R}^{(0)})^\top y_t}{p_1 p_2},$$

where $y_t = \text{vec}(Y_t)$ is the vectorized projected data matrix at time t.

and the pre-estimators for the MAR transition parameters are then given by:

$$\hat{B} \otimes \hat{A}^{(0)} = \left(\sum_{t=2}^{T} \tilde{f}_t \tilde{f}_{t-1}^\top \right) \left(\sum_{t=2}^{T} \tilde{f}_{t-1} \tilde{f}_{t-1}^\top \right)^{-1},$$

$$\hat{Q} \otimes \hat{P}^{(0)} = \sum_{t=2}^{T} \left(\tilde{f}_t - (\hat{B} \otimes \hat{A}^{(0)}) \tilde{f}_{t-1} \right) \left(\tilde{f}_t - (\hat{B} \otimes \hat{A}^{(0)}) \tilde{f}_{t-1} \right)^\top.$$

Initialization Process for the EM Algorithm

Step 1:Number of Factors Estimation

Use the eigenvalue ration criterion form Lam and Yao 2012 to estimate $(\hat{k}_1, \hat{k}_2)$:

$$\hat{k}_1 = \arg\max_{j \le k_{\max}} \frac{\lambda_j(\tilde{M}_1)}{\lambda_{j+1}(\tilde{M}_1)}, \quad \hat{k}_2 = \arg\max_{j \le k_{\max}} \frac{\lambda_j(\tilde{M}_2)}{\lambda_{j+1}(\tilde{M}_2)}$$

Iterative procedure described in Algorithm 3.2.3.

Step 2: Initial Estimation of $\hat{R}$ and $\hat{C}$

Given $\hat{k}_1, \hat{k}_2$, estimate initial loading matrices using PCA:

$$\hat{R}^{(0)} = \sqrt{p_1} \cdot \text{eigvecs}(\hat{M}_1), \quad \hat{C}^{(0)} = \sqrt{p_2} \cdot \text{eigvecs}(\hat{M}_2)$$

with

$$\hat{M}_1 = \frac{1}{T p_1 p_2} \sum_{t=1}^{T} X_t X_t^\top, \quad \hat{M}_2 = \frac{1}{T p_1 p_2} \sum_{t=1}^{T} X_t^\top X_t.$$

Step 3: Projected Estimators $\tilde{R}, \tilde{C}$

Using $\hat{R}^{(0)}$, $\hat{C}^{(0)}$, project the data:

$$\hat{Y}_t = \frac{1}{p_2} X_t \hat{C}^{(0)}, \quad \hat{Z}_t = \frac{1}{p_1} X_t^\top \hat{R}^{(0)},$$

and construct

$$\tilde{M}_1 = \frac{1}{T p_1} \sum_{t=1}^{T} \hat{Y}_t \hat{Y}_t^\top, \quad \tilde{M}_2 = \frac{1}{T p_2} \sum_{t=1}^{T} \hat{Z}_t \hat{Z}_t^\top.$$

to extract the leading $\hat{k}_1, \hat{k}_2$ eigenvectors:

$$\tilde{R} = \sqrt{p_1} \cdot \tilde{Q}_1, \quad \tilde{C} = \sqrt{p_2} \cdot \tilde{Q}_2.$$

Step 4: Pre-Estimation of Factors and MAR Parameters

Use projected data to estimate:

$$\tilde{F}_t = \frac{\tilde{R}^\top Y_t \tilde{C}}{p_1 p_2},$$

and compute MAR parameters via least squares:

$$\widehat{B \otimes A}^{(0)}, \quad \widehat{Q \otimes P}^{(0)}.$$

Step 5: Kalman Filter Initialization

Initialize state and covariance:

$$f_{0|0}^{(0)} = 0_k, \quad \Pi_{0|0}^{(0)} = I_k.$$

3.2.4 EM Convergence

Any iterative procedure, especially within optimization frameworks, requires a well-defined convergence criterion to determine when the algorithm should stop. This also applies to the EM Algorithm. At each iteration n, the algorithm provides an updated set of estimated parameters, denoted by $\hat{\Theta}^{(n+1)}$. Based on these values, it is possible to compute the Prediction Error Log-Likelihood (*PE-LLK*).

The algorithm stops when the improvement in log-likelihood between two successive iterations becomes negligible. Specifically, convergence is reached when the relative difference in *PE-LLK* falls below an arbitrary and user-specified tolerance level ε. Formally this can be expressed as:

$$\Delta L_n = \frac{L(Y_T;\hat{\theta}^{(n+1)}) - L(Y_T;\hat{\theta}^{(n)})}{\frac{1}{2}\left(L(Y_T;\hat{\theta}^{(n+1)}) + L(Y_T;\hat{\theta}^{(n)})\right)}$$

for the seek of clarity, the log-likelihood $L(Y_T;\theta)$ is computed using the Kalman filter, which produces one-step-ahead predictions $f_{t|t-1}$ and associated prediction error covariances $\Pi_{t|t-1}$, as follows:

$$\begin{aligned} L(Y_T;\theta) = -\frac{1}{2}\sum_{t=1}^{T} \Bigg[& \log\det\Big((C\otimes R)\,\Pi_{t|t-1}\,(C\otimes R)^\top + K\otimes H\Big) + \\ & (y_t-(C\otimes R)f_{t|t-1})^\top \Big((C\otimes R)\,\Pi_{t|t-1}\,(C\otimes R)^\top + K\otimes H\Big)^{-1} (y_t-(C\otimes R)f_{t|t-1})\Bigg] \end{aligned} \tag{18}$$

The numerator in Equation 3.2.4 represents the absolute improvement in the *PE-LLK* between two consecutive iterations of the algorithm, while the denominator provides a normalization based on the mean value of the *PE-LLK* at these two iterations. This expression measures the relative percentage improvement of the log-likelihood with respect to the average log-likelihood value across the two iterations.

However, convergence may not occur within a finite number of steps, or the procedure may become computationally intensive in high or ultra-high dimensional settings such as in this case. A possible solution for practical implementation is setting an additional stopping rule based on a maximum number of iterations $n_{\max}$.

Then, the EM algorithm will stop the iterative process as soon as one of the two following conditions is met:

- The relative log-likelihood increment ΔL_n falls below ε,
- The number of iterations reaches $n_{\max}$.

When the stopping condition is satisfied at iteration n^*, the final EM estimate is defined as:

$$\hat{\Theta} = \hat{\Theta}^{(n^*+1)}.$$

and a last run of the Kalman smoother is performed to compute the smoothed estimates of the latent factor matrices.

3.3 Extension To missing Data

A fundamental challenge that any practitioner must face when working with real-world data is the presence of missing observations. This issue is often addressed by aggregating data at higher level or frequency (if in time series context), where all entries are available, or by applying simple imputation methods to fill in the gaps. However, such approaches are not well suited to practical applications, particularly in the context of factor models, since every observed entry can potentially carry relevant information. Aggregating data may lead to the exclusion of meaningful variation, while direct imputation methods risk to introduce some noise and bias into the dataset.

In this context, where the entire estimation strategy is designed to handle increasing layers of generality and structure, explicitly addressing the missingness problem becomes particularly important. The way this work addresses this challenge is the adoption of a methodology developed in the vector case, namely the "all-purpose estimator" introduced by Xiong and Pelger 2023, to the matrix setting, as proposed by Cen and Lam 2025. However, imputation alone is not sufficient. By implementing modifications to the EM algorithm that explicitly take into account the pattern of missing data, it is possible to fully exploit the available information without increasing the noise in the system. This allows the EM-based estimation procedure to retain its theoretical and practical advantages, even in the presence of incomplete data.

This subsection provides a discussion of this imputation process, with reference to its vector-based counterpart, and then presents the necessary adjustments to the EM algorithm as introduced by Matteo Barigozzi and Trapin 2025.

3.3.1 Imputation for the EM initialization

The initial parameters required by the EM algorithm for its first iteration can be obtained as discussed in Subsection 3.2.3. However, missing data can add a supplementary complexity layer in terms of the projected estimation. Indeed the methodology proposed by L. Yu et al. 2022 cannot be directly applied in presence of missing values, since it relies on the extraction of principal components from a sample covariance matrix obtained by projecting the observation matrix onto the row or column factor space. In general, PCA techniques cannot operate in the presence of missing values, and the solution is to impute the missing data beforehand. Once the missing observations have been imputed, Projected Estimators can work effectively.

A common practice, also adopted by Matteo Barigozzi and Trapin 2025, is to perform this initial imputation using the methodology proposed by Cen and Lam 2025. This approach extends to the tensor framework the so-called "all-purposes estimator", a method originally developed by Xiong and Pelger 2023 for vector time series. This Subsection focuses mainly on the extension to the tensor, specifically the matrix case, but briefly review also the standard vector approach as a simplifying introduction.

Imputation of Missing Values for the Vector Case Xiong and Pelger 2023 propose the so-called "all-purpose estimator" for handling missing values in high-dimensional vector time series within a latent factor model framework. This PCA-based approach consists in applying principal component analysis to an adjusted covariance matrix estimated from partially observed panel data. Specifically, given a data matrix $Y \in \mathbb{R}^{T\times N}$ with potentially random pattern of missing entries (as in our mixed-frequency case where quarterly variables are missing by design), an initial imputed matrix $\widetilde{Y}$ is constructed as:

$$\widetilde{Y}_{it} = Y_{it}W_{it}, \quad \text{for all} \quad i = 1,\dots,N \quad \text{and} \quad t = 1,\dots,T,$$

where $W_{it} = 1$ if Y_{it} is observed, and 0 otherwise.

Directly applying PCA to the sample covariance matrix $\frac{1}{T}YY^\top$ would result in biased estimates, because setting missing values to zero does not preserve the true covariance structure. And to correct for this bias, Xiong and Pelger 2023 propose estimating a weighted covariance matrix:

$$\widetilde{\Sigma}_{ij} = \frac{1}{|Q_{ij}|}\sum_{t\in Q_{ij}} Y_{it}Y_{jt},$$

where Q_{ij} is the set of time periods where both variables i and j are observed. Note that $|Q_{ij}|$ denotes its cardinality.

Thus, each entry of the covariance matrix is built using only available information, correcting for missing observations regardless of their pattern. This method is robust not only for mixed-frequency settings but also for datasets with missing-at-random entries or missing at block, and that's the reason why they call it ”all-purpose”.

After constructing $\widetilde{\Sigma}$, PCA extract the factor loadings. Specifically, the loadings $\widetilde{\Lambda}$ are estimated as the scaled eigenvectors associated with the largest eigenvalues:

$$\frac{1}{N}\widetilde{\Sigma}\widetilde{\Lambda} = \widetilde{\Lambda}\widetilde{V},$$

where $\widetilde{V}$ is a diagonal matrix containing the largest eigenvalues.

Once the loadings are estimated, the common factors are retrieved by regressing the observed entries of Y on the estimated loadings. For each time period t, the factors $\widetilde{F}_t$ are computed by solving a weighted least squares problem:

$$\widetilde{F}_t = \left(\sum_{i=1}^{N} W_{it}\widetilde{\Lambda}_i\widetilde{\Lambda}_i^\top\right)^{-1}\left(\sum_{i=1}^{N} W_{it}\widetilde{\Lambda}_i Y_{it}\right),$$

where $\widetilde{\Lambda}_i$ denotes the i-th row of $\widetilde{\Lambda}$.

Imputation Procedure for the Matrix Case Building upon the approach by Xiong and Pelger 2023, Cen and Lam 2025 extend the imputation methodology to the tensor setting. In the matrix setting, given a partially observed dataset $Y \in \mathbb{R}^{T\times N}$, the missing data problem is addressed by constructing an adjusted sample covariance matrix based only on available observations. Given any pair of variables (i, j), the adjusted covariance is computed as

$$\widetilde{\Sigma}_{ij} = \frac{1}{|Q_{ij}|}\sum_{t\in Q_{ij}} Y_{it}Y_{jt},$$

where Q_{ij} denotes the set of time periods where both series i and j are simultaneously observed. This estimator, weighted for observed data, corrects for the bias introduced by missing entries, and the appealing advantage is that it doesn't require any distributional assumption on the missingness.

PCA is performed on the adjusted covariance matrix $\widetilde{\Sigma}$, allowing the consistent estimation of the factor loading matrix $\widetilde{\Lambda}$ as the eigenvectors associated with the largest eigenvalues. Finally, the common factors are obtained at each time t by solving a weighted least squares problem:

$$\widetilde{F}_t = \left(\sum_{i=1}^{N} W_{it} \widetilde{\Lambda}_i \widetilde{\Lambda}_i^\top \right)^{-1} \left(\sum_{i=1}^{N} W_{it} \widetilde{\Lambda}_i Y_{it} \right),$$

where W_{it} is the observation indicator (1 if observed, 0 if missing).

This procedure is particularly appealing because it accommodates very general patterns of missing data, including non-random missingness and mixed-frequency structures, making it highly suitable for large-dimensional macroeconomic datasets used by institutions such as central banks.

3.3.2 Modification of the EM-algorithm

EM-algorithm can be considered a robust method to treat missing values, but the generalization comes at cost of slightly more demanding theoretical and computational consideration. As already mentioned, through the Kalman smoother implementation, factors and their associated *MSE* can be estimated consistently. Recalling the discussion in paragraph 3.2.2, one can decompose the expected log-likelihood in a part depending just on the factors, namely:

$$\mathbb{E}_{\theta^{(n)}}[\ell(F_{1:T}; \theta) \mid Y_{1:T}]$$

and a component depending on the data, that is:

$$\mathbb{E}_{\theta^{(n)}}[\ell(Y_{1:T} \mid F_{1:T}; \theta) \mid Y_{1:T}]$$

Since the first equation depends only on the latent factors and not directly on the observed data, missing occurs do not imply a change the analytical form and consequently parameters A, B, P, and Q remain untouched. On the other hand, the second component is affected by missingness since data change relatively to the existence and the pattern of missing values. Accordingly, *MFM* parameters, as loading matrices R and C and idiosyncratic variances H and K, change. Missing are detected practically by a selection matrix.

The M-step is then adjusted as in Bańbura and Modugno 2014 for the vector case, and Matteo Barigozzi and Trapin 2025 for the matrix case. Referring to the latter, thus in a matrix-variate time series framework, the selection matrix W_t, withe the same dimension $p_1 \times p_2$ of Y_t is the following:

$$(W_t)_{ij} = \begin{cases} 1 & \text{if } y_{t,ij} \text{ is observed} \\ 0 & \text{otherwise} \end{cases}$$

Sequently, *MFM* parameters are revised at each iteration. The explicit solutions for the loading matrices R and C for any current parameter estimates $\hat{\theta}^{(n)}$ are updated as follows during the estimation:

$$\begin{aligned}\operatorname{vec}\left(\hat{R}^{(n+1)}\right) = & \left(\sum_{t=1}^{T}\sum_{s=1}^{p_1}\sum_{q=1}^{p_1}\left[\left(\hat{C}^{(n)\top}D^{[s,q]}W_t\hat{K}^{(n)-1}\hat{C}^{(n)}\right)\star\left(\hat{f}_{t|T}^{(n)}\hat{f}_{t|T}^{(n)\top}+\Pi_{t|T}^{(n)}\right)\right]\otimes\left(E_{p_1,p_1}^{[s,q]}\hat{H}^{(n)-1}\right)\right)^{-1} \\ & \times\left(\sum_{t=1}^{T}\operatorname{vec}\left(\left[W_t\circ\hat{H}^{(n)-1}Y_t\hat{K}^{(n)-1}\right]\hat{C}^{(n)}\hat{f}_{t|T}^{(n)\top}\right)\right)\end{aligned}$$

$$\begin{aligned}\operatorname{vec}\left(\hat{C}^{(n+1)}\right) = & \left(\sum_{t=1}^{T}\sum_{k=1}^{p_2}\sum_{q=1}^{p_2}\left[\left(\hat{R}^{(n+1)\top}D^{[k,q]}W_t^{\top}\hat{H}^{(n)-1}\hat{R}^{(n+1)}\right)\star K_{k_1,k_2}\left(\hat{f}_{t|T}^{(n)}\hat{f}_{t|T}^{(n)\top}+\Pi_{t|T}^{(n)}\right)K_{k_1,k_2}^{\top}\right]\otimes\left(E_{p_2,p_2}^{[k,q]}\hat{K}^{(n)-1}\right)\right)^{-1} \\ & \times\left(\sum_{t=1}^{T}\operatorname{vec}\left(\left[W_t\circ\hat{H}^{(n)-1}Y_t\hat{K}^{(n)-1}\right]^{\top}\hat{R}^{(n+1)}\hat{f}_{t|T}^{(n)}\right)\right)\end{aligned}$$

for the same iteration $n \geq 0$, and the current parameter estimates $\hat{\theta}^{(n)}$, the idiosyncratic variances $\hat{H}$ and $\hat{K}$ in the presence of missing data are computed are:

$$\begin{aligned}\left[\hat{H}^{(n+1)}\right]_{ii} = \frac{1}{Tp_2}\sum_{t=1}^{T}\Bigg\{ & (W_t\circ Y_t)\hat{K}^{(n)-1}(W_t\circ Y_t)^{\top}-(W_t\circ Y_t)\hat{K}^{(n)-1}\left(W_t\circ\left[\hat{R}^{(n+1)}\hat{f}_{t|T}^{(n)}\hat{C}^{(n+1)\top}\right]\right)^{\top} \\ & -\left(W_t\circ\left[\hat{R}^{(n+1)}\hat{f}_{t|T}^{(n)}\hat{C}^{(n+1)\top}\right]\right)\hat{K}^{(n)-1}(W_t\circ Y_t)^{\top} \\ & +\hat{K}^{(n)-1}\star\left[D_{W_t}\left(\hat{C}^{(n+1)}\otimes\hat{R}^{(n+1)}\right)\left(\hat{f}_{t|T}^{(n)}\hat{f}_{t|T}^{(n)\top}+\Pi_{t|T}^{(n)}\right)\left(\hat{C}^{(n+1)}\otimes\hat{R}^{(n+1)}\right)^{\top}D_{W_t}^{\top}\right] \\ & +\left[\hat{H}^{(n)}\mathbf{1}_{p_1,p_2}\hat{K}^{(n)}\right]\hat{K}^{(n)-1}\left(\mathbf{1}_{p_1,p_2}-W_t\right)^{\top}\Bigg\}_{ii}\end{aligned}$$

$$\begin{aligned}\left[\hat{K}^{(n+1)}\right]_{ii} = \frac{1}{Tp_1}\sum_{t=1}^{T}\Bigg\{ & (W_t\circ Y_t)^{\top}\hat{H}^{(n+1)-1}(W_t\circ Y_t)-(W_t\circ Y_t)^{\top}\hat{H}^{(n+1)-1}\left(W_t\circ\left[\hat{R}^{(n+1)}\hat{f}_{t|T}^{(n)}\hat{C}^{(n+1)\top}\right]\right) \\ & -\left(W_t\circ\left[\hat{R}^{(n+1)}\hat{f}_{t|T}^{(n)}\hat{C}^{(n+1)\top}\right]\right)^{\top}\hat{H}^{(n+1)-1}(W_t\circ Y_t) \\ & +\hat{H}^{(n+1)-1}\star\left[D_{W_t}^{\top}\left(\hat{R}^{(n+1)}\otimes\hat{C}^{(n+1)}\right)K_{k_1,k_2}\left(\hat{f}_{t|T}^{(n)}\hat{f}_{t|T}^{(n)\top}+\Pi_{t|T}^{(n)}\right)K_{k_1,k_2}^{\top}\left(\hat{R}^{(n+1)}\otimes\hat{C}^{(n+1)}\right)^{\top}D_{W_t}^{\top}\right] \\ & +\left(\mathbf{1}_{p_1,p_2}-W_t\right)^{\top}\hat{H}^{(n+1)-1}\left(\hat{H}^{(n+1)}\mathbf{1}_{p_1,p_2}\hat{K}^{(n)}\right)\Bigg\}_{ii}\end{aligned}$$

3.4 Nowcasting

Forecasting in real-time is becoming increasingly populate over the last decade. This approach is called Nowcasting and it is particularly appealing when target variables exhibits less timely frequency with respect to other variables. Indeed, to extract a prediction of low-frequency target variables it is possible to exploit the information conveyed by high-frequency indicators. The original framework proposed by Giannone, Reichlin, and Small 2008 is in this paper extended to matrix-variate time series and nowcasts are retrived implicitly from a reconstructed data as in Bańbura and Modugno 2014. Nevertheless, the approach of Giannone, Reichlin, and Small 2008 was already able to outperform simple bridge equation models, by managing mixed-frequency time series and unbalanced release pattern at the end of the sample. The asynchronous data structure (jagged-edge) determined by different time of release of macroeconomic variables can be treated and provide regular update of the not-yet released variables of interest.

To introduce the empiric of this paper, consider quarterly variables treated as missing in the data matrix Y_t and whenever they are not observed can be predicted as part of the model, exploiting both the cross-sectional and time-series information from monthly and possibly other quarterly indicators, through the latent factor dynamics.

To Nowcast matrix-variate time series I provide an extension of the approach by Bańbura and Modugno 2014, who reconstruct the dataset using the factors in vector-form dynamic factor models. By vectorizing the Dynamic Matrix Factor Model in Equation 3, the equation for the data-generating process becomes:

$$\mathrm{vec}(Y_t) = (C \otimes R) \cdot \mathrm{vec}(F_t) + \varepsilon_t, \quad \varepsilon_t \sim \mathcal{N}(0, K \otimes H)$$

And the latent factors dynamics, assuming a matrix autoregressive process is:

$$\mathrm{vec}(F_t) = \Phi \cdot \mathrm{vec}(F_{t-1}) + u_t, \quad \Phi = B \otimes A$$

Once the model is estimated via the EM algorithm, accounting also for the presence of missing values, nowcasting can be performed using Kalman filtering techniques applied to the vectorized state.

As described in Paragraph 3.2.2, through the Kalman filter, one can compute the expected values of the latent factors, their second moments and eventually update the estimates by filtering the new data release. Then, the Kalman smoother performs a backward recursion to refine the state estimate. Now all that left is to reconstruct the data, as

$$\hat{Y}_{t|T} = R \cdot \mathrm{mat}(\hat{f}_{t|T}) \cdot C^{\top}$$

Since the last observations cannot be updated and corrected backward through the Kalman smoother since no other observations are collected, they correspond to the Kalman filter. Thus, these are the nowcast we are interested to. This procedure allows to extract a real-time estimate of the target quarterly variable.

The approach described can be recursively performed: at each vintage v_j, the model can be re-estimated and the Kalman filter is applied to update the nowcast of the target variable. At the next vintage v_{j+1}, the information set is expanded as new data are released. at this stage either new monthly or even quarterly observations are released or updates to previously released data can be accounted. This process is repeated in a rolling fashion to simulate real-time forecasting and revision, providing a time series of nowcast that can be compared to the true value of the variable of interest.

Recursive Nowcasting Procedure with *DMFM*

For any vintage t_v:

1. Build the observed tensor $Y^{(v)} = \{Y_1, \ldots, Y_{t_v}\}$;
2. Estimate model parameters using the EM algorithm;
3. Reconstruct the dataset using the Kalman Smoother
4. Provide the nowcasts:
   - If $y^q_{t_v+1}$ is missing: store $\hat{y}^q_{t_v+1|t_v}$ as the nowcast.
   - If $y^q_{t_v+1}$ is released: compute forecast error:

$$\text{Error}_{t_v+1} = y^q_{t_v+1} - \hat{y}^q_{t_v+1|t_v}$$

4 Empirics

This chapter represents the core contribution of this work, as it empirically assesses the validity of the methodology discussed in Chapter 3 by applying the estimation and nowcasting techniques discussed to Euro Area data modeled through a DMFM. The central research question is whether the DMFM, using factors extracted from a set of monthly indicators and GDP, can outperform its vector counterpart in terms of nowcasting performance.

In doing so, the approach offers several layers of flexibility, making the methodology well-suited for real-world applications, as it (i) preserves the original structure of the data, i.e., matrix-variate time series; (ii) accommodates missing values arising from the mixed-frequency nature of the dataset and the treatment of real variables during the COVID-19 period; (iii) exploits the asynchronous release of data, enabling sequential updates of nowcasts as more recent information becomes available over time.

The first step in the empirical analysis was to construct the matrix-variate time series, namely a tensor where two dimensions are observed over time. From the dataset provided by M. Barigozzi and Lissona 2024, I extracted a subset of countries and variables according to selection criteria detailed in Subsections 4.1.1 and 4.1.2. I then built a matrix where countries are arranged as rows and the macroeconomic variables specific to each country as columns. This bilinear structure is repeated over time, reflecting how data are originally collected within a Monetary Union. The empirical analysis presented here focuses on Germany, France, Italy, and Spain. For each of these countries, I considered the full set of monthly indicators along with GDP as the key national account variable, amounting to a total of 40 indicators[4].

The selected EA economies, as members of a Monetary Union, collect similar variables in a partially harmonized fashion. Being key drivers of Euro Area activity, they are plausibly tied to common latent factor, which interpretation is discussed in Subsection4.2. Refering first to some anecdotal evidence presented in Subsection 4.1 supporting this idea and through the rigorous analysis theoretically presented, the goal of this chapter is to assess the benefits of preserving the matrix structure for real-time forecasting. To do so performance of matrix-based DFMs are compared to their vector counterparts. The results are discussed in Subsection 4.3.1.

Practically, the exercises consists in recursively truncating the dataset month-by-month and applying a mask that accounts for the actual release calendar of each variable. then estimate, regardless of the model's vector or matrix form, the model parameters via the EM algorithm, and extracxt the corresponding nowcasts for quarterly target variables, namely GDP, through the Kalman Filter. Is the EM algorithm the key element in this approach as it enables consistent estimation in the presence of missing values stemming from both the different data frequencies and the treatment of real variables during the COVID period. Combined with the Kalman filter, it provides a robust strategy to handle the delayed release of data and provide nowcasts.

While this recursive nowcasting approach has become standard in vector-based DFM implementations, to the best of the author's knowledge, this work represents one of the first applications of a nowcasting

[4]From the replication code in R available at: `https://github.com/dolpolo/Nowcasting_DMFM`, this framework can be extended to include more variables and countries, up to the limits of the dataset in M. Barigozzi and Lissona 2024.

framework to tensor-structured data, even in its simplest three-dimensional form of matrix-variate time series. The empirical results suggest that, under certain conditions, the matrix-based approach can outperform traditional vector-based Dynamic Factor Models.

To summarize, the research questions that guide this empirical investigation are directed to understand whather co-movements among Euro Area countries can be exploited to improve the nowcast of an individual EA country's GDP, and if this can be achieved nowcasting high-frequency indicators, such as GDP, just extracting factors from monthly sequences. In other words what follows can be read keeping in mind the following question: can the inclusion of monthly indicators and cross-country information improve the accuracy of GDP nowcasts for individual EA countries?

4.1 Data

The dataset used for this analysis results from a country and variable selection performed on the "EA-MD-QD" dataset by M. Barigozzi and Lissona 2024. In this subsection, I discuss the selection criteria applied to extract the final set of countries and variables, along with the procedure used for data preparation.

The original dataset includes quarterly and monthly macroeconomic indicators from January 2000 to April 2025, recorded at monthly frequency, yielding a total of $T = 300$ observations for ten Euro Area member countries: Germany, France, Italy, Spain, the Netherlands, Belgium, Portugal, Austria, Ireland, and Greece. During this period, Europe experienced three major global shocks: the 2008 Global Financial Crisis, the 2012 Sovereign Debt Crisis, and the COVID-19 pandemic in 2020. Each of these shocks differs in origin and nature, with COVID-19 being a purely exogenous event that poses unique challenges for empirical modeling. To avoid the introduction of noise during this period, real variables have been masked and replaced with filtered forecasts obtained via the Kalman filter.

The core application presented in this work focuses on the first four countries, which are considered the primary drivers of the Euro Area business cycle. From the full set of available macroeconomic indicators, the analysis retains all monthly variables and includes GDP as the only quarterly variable, which also serves as the nowcasting target. The structure of the resulting dataset, in terms of variable selection, aims to replicate that used in Giannone, Reichlin, and Small 2008. Moreover, it is similar to the dataset adopted by Cascaldi-Garcia et al. 2024, although the model specification differs, as they construct a multi-country framework to nowcast economic conditions in the Euro Area and the same set of member countries—excluding Spain.

The time series are arranged in matrix format and stored in a three-dimensional tensor. The first dimension corresponds to countries (rows), the second to variables (columns), and the third to time (slices). The resulting matrix-valued time series contains $T = 300$ monthly observations for $p_1 = 4$ countries and $p_2 = 40$ variables. This dataset is inherently unbalanced, not only due to the mixed-frequency nature of the variables but also due to staggered release schedules.

For the implementation of the methodology, the variables are required to be stationary, with zero mean and unit variance. Stationarity can be automatically imposed during the data construction phase using tools provided by M. Barigozzi and Lissona 2024. Centering and standardization are initially applied, and then re-applied after the estimation procedure. A brief overview of this process is provided in the Appendix.

To summarize, after the collection and selection phase, the time series are arranged in matrix format and stored in a three-dimensional tensor. The first dimension refers to countries, the second to variables, and the third to time. The resulting structure consists of $T = 300$ monthly observations for $p_1 = 4$ countries and $p_2 = 40$ variables. Within this dataset, as expected, two main sources of missing data are observed: (i) blocks of missing values for real variables during the COVID-19 period, and (ii) a regular pattern of missing entries for lower-frequency variables (e.g., GDP), consistent with a mixed-frequency framework.

4.1.1 Country Selection

Countries considered in this analysis are: Germany, France, Italy, and Spain. These countries were chosen because they represent the largest economies within the Euro Area, and account for the majority of its aggregate output, as shown in Figure 1. Note that all the ten countries present in the "EA-MD-QD" dataset are marked in red,

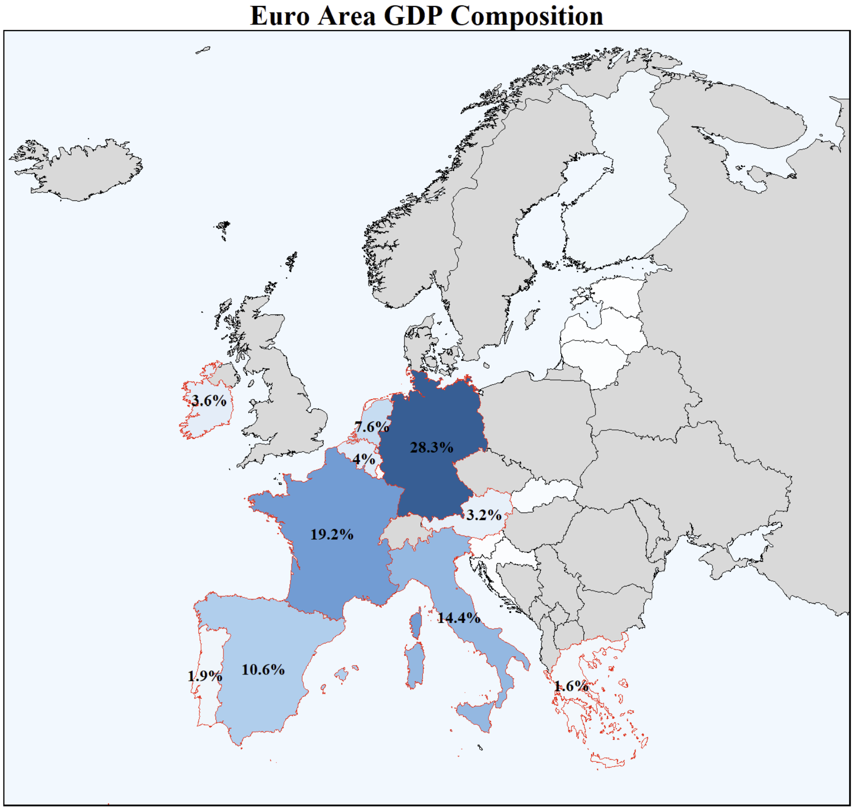
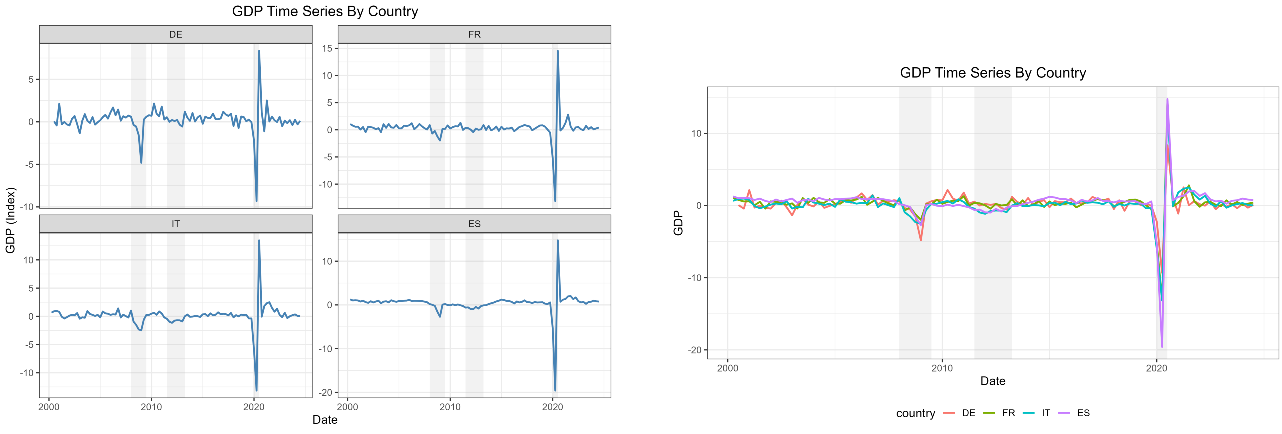


Figure 1: EA countrys' by GDP share relative to the total Euro Area GDP based on Eurostat data.

As shown in the figure, Germany is the largest contributor, followed by France, Italy, and Spain.

In addition to their economic relevance, these countries exhibit strong co-movements in their business cycles, providing anecdotal yet meaningful evidence of shared latent structures. This is confirmed by Figures 2a and 2b, which display the individual and overlapping quarterly GDP time series for the selected countries.

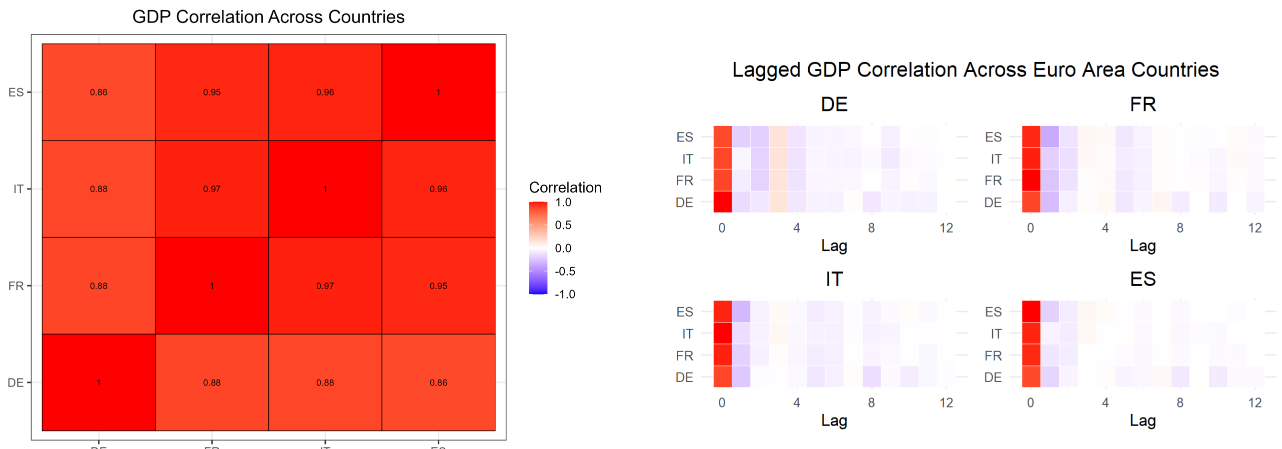


(a) Quarterly GDP Series

(b) Overlapping GDP Series

Figure 2: Visual comparison of GDP dynamics across selected Euro Area countries.

This co-movement is further supported by the high pairwise correlations among the GDP series, as reported in Figure 3a. For instance, the GDPs of Italy and France show a correlation of 0.97, with all other pairwise correlations exceeding 0.85. These strong contemporaneous relationships weaken notably when introducing time lags. As illustrated in Figure 3b, correlations decrease significantly with a one-month lag and become negligible thereafter. This empirical pattern underscores the need for models capable of capturing contemporaneous dependencies across countries, such as the DMFM framework adopted in this work.



(a) Contemporaneous GDP Correlation Matrix

(b) Lagged GDP Correlations (One-Year Lag)

Figure 3: Cross-country correlations in GDP series, contemporaneous and lagged.

4.1.2 Variables Selection

As metioned in the introduction of thi Chapter, from the large pool of monthly and quarterly variables in the dataset by M. Barigozzi and Lissona 2024, just a subset is been considered in the analysis and it is composed by 39 monthly indicators and GDP, the only quarterly variable, which also serves as the nowcasting target [5].

[5] In the R code, variables can be selected based on their correlation with GDP. For each country, a set of variables is extracted according to their correlation strength. These are then aggregated with the variables most correlated with the

Using such vaariable composition, namely a total of 40 variables, where just the GDP is quarterly, mirrors the empirical setup of Giannone, Reichlin, and Small 2008, where the authors rely solely on monthly indicators and GDP. This choice allows us to address the first core research question: whether monthly indicators alone are sufficient to predict GDP fluctuations. Furthermore, this design keeps the nowcasting exercise computationally feasible. Moreover, the decision to select a medium-sized set of variables is not a limitation. As shown in Bańbura and Modugno 2014 and applied in Cen and Lam 2025, a small or medium set can often be preferable to a larger one in terms of forecasting performance.

Once the variables have been chosen the challenge imposed by the COVID-19 disruption has to be faced. Pandemic crises affected primarly real variables and for this reason those were masked in the dataset in correspondence to the period during which they reported significant otliers. Considering missing values generated by the intrinsic mixed-frequency nature of the data and the treatment of real variables during COVID-19 the resulting dataset considers a total share of missing values equal to 4.7%.

Table 4.1.2 provides a compact overview of the main features of the dataset, with special attention to the asynchronous release schedule of variables. For each indicator, the column "Delay" reports the average lag, in terms of day, at which data are released relative to the last day of the frequency of that indicator. For instance, GDP is recorded quarterly but released 45 days after the last day of the reference period. In other words, the GDP of the first quarter is typically published in mid-April, namely on average 45 days after March 31^{st}. The most timely indicators are confidence once, which are generally published within the same month and provide early insights into economic conditions. These survey-based indicators are commonly referred to as *soft data*. While less precise than quantitative (*hard*) data, they are more timely and particularly useful for nowcasting purposes. In fact, central banks often rely on such signals to detect turning points in the business cycle, as noted by Bańbura and Modugno 2014 and Giannone, Lenza, and Primiceri 2021. Soft data reflect expectations and sentiment, whereas hard data (e.g., GDP, CPI, industrial production) are objective and quantitative but typically released with delay and subject to revision.

As mentioned, Table 4.1.2 is also relevant as it summarizes the key features of the variables used in the empirical analysis. From "Class" column one can check which variables are Real once and then masked during the COVID-19 period, specifically from the first quarter of 2020 to the second quarter of 2025. "Category" is binary indicator distinguishing soft data (most relevant for nowcasting) from hard data. "Transformation" is relevant when preparing the data to impose stationarity and for the final interpetation of results. Finaly "Frequency" which variables are monthly.

As for Countries, aanecdotal evidence supporte the presence of common latent factors among the selected variables. Figure 13 in Appendix displays the heatmap of pairwise correlations among the complete set of variables present in M. Barigozzi and Lissona 2024. This is a strong signal that can suggest a common factor structure.

GDP of other countries (e.g., selecting the top 5 variables per country and aggregating them across countries considered for each country also then 5 variables selected for the others if different).

Table 2: Macroeconomic variables selected and release delays

N	ID	Series Name	Class	Category	Transformation	Freq.	Delay
(1) National Accounts / Real Economy							
1	GDP	Real Gross Domestic Product	R	H	1	Q	45
(2) Labor Market							
2	UNETOT	Unemployment: Total	R	H	0	M	40
3	UNEU25	Unemployment: Under 25 years	R	H	0	M	40
4	UNEO25	Unemployment: Over 25 years	R	H	0	M	40
(4) Exchange and Interest Rates							
5	REER42	Real Exchange Rate (42 countries)	F	H	1	M	30
6	LTIRT	Long-Term Interest Rates (EMU)	F	H	2	M	30
(5) Industrial Production and Turnover							
7	IPMN	Industrial Production Index: Manufacturing	R	H	1	M	45
8	IPING	Industrial Production Index: Energy	R	H	1	M	45
9	IPCAG	Industrial Production Index: Capital Goods	R	H	1	M	45
10	IPDCOG	Industrial Production Index: Durable C. Goods	R	H	1	M	45
11	IPNDCOG	Industrial Production Index: Non-Durable C.G.	R	H	1	M	45
12	IPCOG	Industrial Production Index: Consumer Goods	R	H	1	M	45
13	IPNRG	Industrial Production Index: Energy	R	H	1	M	45
14	TRNING	Turnover Index: Intermediate Goods	R	H	1	M	45
15	TRNDCOG	Turnover Index: Durable Consumer Goods	R	H	1	M	45
16	TRNCAG	Turnover Index: Capital Goods	R	H	1	M	45
17	TRNNRG	Turnover Index: Energy	R	H	1	M	45
18	TRNCOG	Turnover Index: Consumer Goods	R	H	1	M	45
19	TRNNDCOG	Turnover Index: Non-Durable C. Goods	R	H	1	M	45
(6) Prices							
20	PPIING	Producer Price Index: Intermediate Goods	N	H	1	M	15
21	PPINRG	Producer Price Index: Energy	N	H	1	M	15
22	PPICAG	Producer Price Index: Capital Goods	N	H	1	M	15
23	PPICOG	Producer Price Index: Consumer Goods	N	H	1	M	15
24	PPINDCOG	Producer Price Index: Non-Durable C. Goods	N	H	1	M	15
25	PPIDCOG	Producer Price Index: Durable C. Goods	N	H	1	M	15
26	HICPNG	HICP: Energy	N	H	1	M	15
27	HICPNEF	HICP: All Items excl. Energy and Food	N	H	1	M	15
28	HICPIN	HICP: Industrial Goods	N	H	1	M	15
29	HICPOV	HICP: Overall Index	N	H	1	M	15
30	HICPG	HICP: Goods	N	H	1	M	15
31	HICPSV	HICP: Services	N	H	1	M	15
(3) Confidence Indicators							
32	ICONFIX	Industrial Confidence Indicator	C	S	0	M	0
33	CCONFIX	Consumer Confidence Indicator	C	S	0	M	0
34	KCONFIX	Construction Confidence Indicator	C	S	0	M	0
35	SCONFIX	Services Confidence Indicator	C	S	0	M	0
36	ESENTIX	Economic Sentiment Indicator	C	S	0	M	0
37	RTCONFIX	Retail Confidence Indicator	C	S	0	M	0
38	BCI	Composite Business Confidence Index	C	S	1	M	0
39	CCI	Composite Consumer Confidence Index	C	S	1	M	0
(7) Others							
40	SHIX	Share Price Index	F	H	1	M	30

4.2 DMFM Factors: Selection and Interpretation

This subsection introduces the empirical implementation of the methodology outlined in Subsubsection 3.2.3, focusing on the selection and interpretation of the number of common factors along rows and columns. Following the approach of L. Yu et al. 2022, the number of factors is determined using Algorithm 2, as presented in Box 3.2.3, and using this number of factors I provided the estimates of the row and column loadings as summarized in Algorithm 1 (Box 3.2.3). These estimates are subsequently used to initialize the EM algorithm.

As discussed in Subsubsections 4.1.2 and 4.1.1, the dataset exhibits strong co-movements across both countries and variables, suggesting the presence of a low-rank latent structure. To capture this, I apply the methods proposed in L. Yu et al. 2022, which rely on eigendecomposition of the covariance matrix and require a fully observed dataset. To address missing values, I employ the imputation strategy of Cen and Lam 2025, which begins with a high number of factors and iteratively reduces dimensionality as stability is achieved.

Finally, a brief justification is provided for the assumption of a single lag in the MAR process, along with a discussion of potential extensions for future research.

The first step involves selecting the correct number of factors by arbitrarily starting with a large number of row and column factors. Using Algorithm 2, I found immediate convergence to one row factor ($k_1 = 1$) and one column factor ($k_2 = 1$). Figure 4 graphically illustrates the eigenvalue spectrum, where a clear jump between the first and second components is evident for both rows and columns. This result is also supported by no data driven methods as the so-called elbow criterion.Cumulative variance is not a meaningful measure and it is always better to relay on the eigenvalue ratio.

Once the number of factors has been determined, the next step is the estimation of loadings using

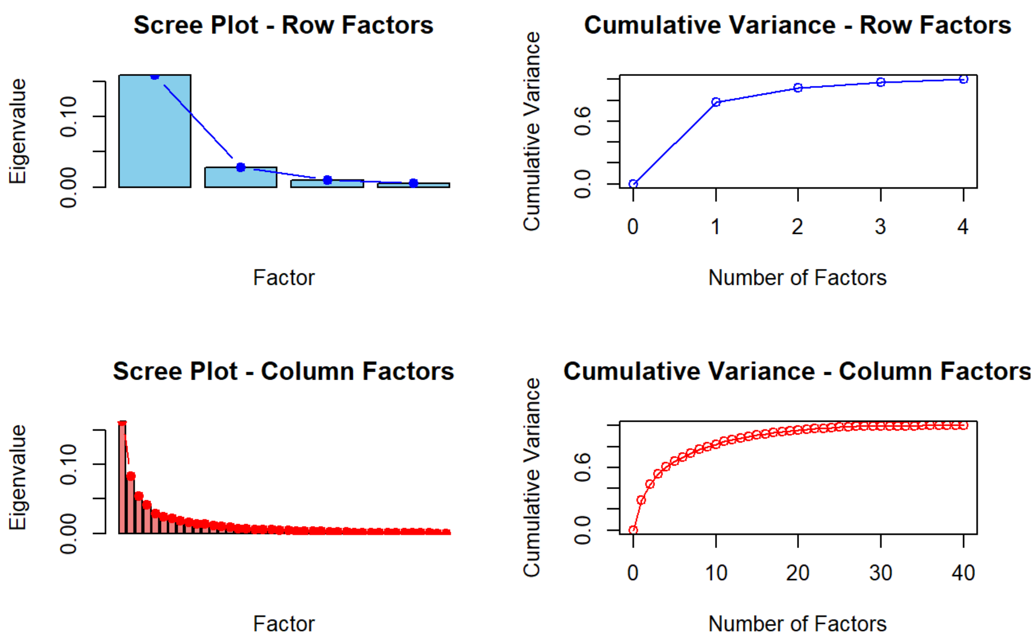


Figure 4: Scree plot and cumulative variance for row and column factor selection.

Algorithm 1 from L. Yu et al. 2022. According to the authors, a single iteration of the algorithm is often sufficient to obtain reliable estimates, which, in this case, provides a solid initialization for the EM algorithm and facilitates convergence to a maximum of the quasi log-likelihood.

Starting with the row loadings, Table 3 can be used to interpret the latent variable by evaluating the weights assigned to each country. Since the results show nearly equal loadings across all countries, representing almost an average, the row factor can be interpreted as capturing a shared feature: membership in the Euro Area. This factor is thus labeled the "Euro Area Membership Factor."

Table 3: Estimated Row Loadings

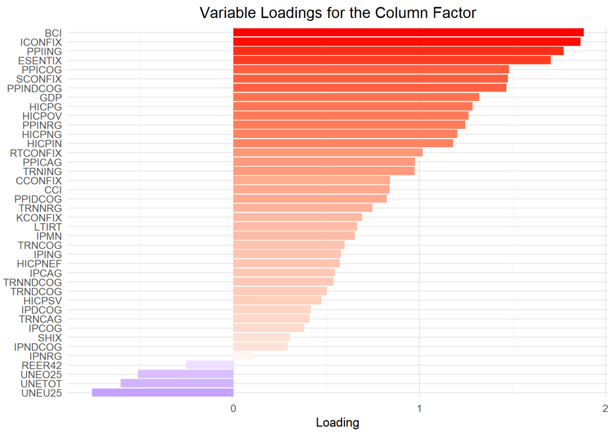
Country	**DE**	**FR**	**IT**	**ES**
Loading	0.899	1.078	1.024	0.990

In contrast, the column loadings reported in Table 4 offer a more insightful interpretation. Variables with high positive loadings include confidence indicators and GDP, while unemployment indicators exhibit negative loadings. This suggests that the factor reflects the state of the macroeconomic business cycle: when the factor is high, confidence is high and unemployment is low. Accordingly, this factor is labeled the "Recession Factor." To facilitate interpretation, Figure 5a displays variables ordered by their column factor loadings.

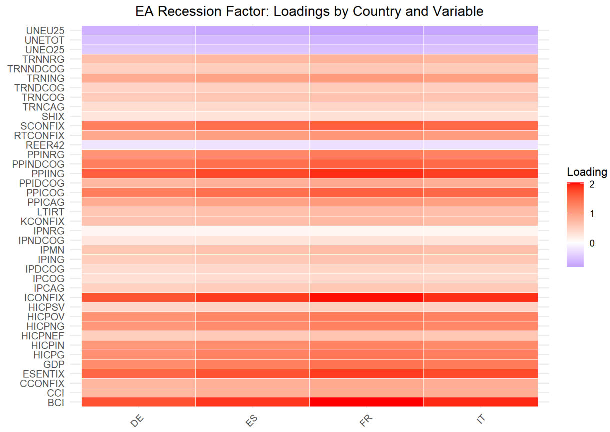
Table 4: Estimated Column Loadings

Variable	**Loading**	**Variable**	**Loading**
TRNING	0.975	BCI	1.885
IPMN	0.654	PPIING	1.776
IPING	0.578	ESENTIX	1.708
TRNDCOG	0.504	TRNNDCOG	0.538
IPCAG	0.546	IPNDCOG	0.291
IPDCOG	0.415	CCONFIX	0.842
TRNCAG	0.408	CCI	0.840
TRNNRG	0.747	PPINRG	1.246
IPCOG	0.379	HICPNG	1.204
TRNCOG	0.597	UNEU25	-0.760
SHIX	0.305	HICPNEF	0.571
RTCONFIX	1.017	SCONFIX	1.476
IPNRG	0.104	REER42	-0.253
ICONFIX	1.866	LTIRT	0.664
PPINDCOG	1.468	UNETOT	-0.608
PPICOG	1.481	UNEO25	-0.514
HICPSV	0.473	PPICAG	0.976
PPIDCOG	0.825	HICPIN	1.182
HICPOV	1.265	HICPG	1.287
KCONFIX	0.692	GDP	1.322

It is worth noting that the model is only identifiable up to orthogonal transformations. For interpretability, I restrict the row and column loadings to be positive, which improves consistency and aids in economic interpretation.



(a) Column Loadings Ordered by Variable



(b) Combined Loadings (Row × Column)

Together, the row and column factors can be interpreted as a single latent component capturing the Euro Area business cycle. Figure 5b visualizes the impact of this factor across Euro Area countries and variables. The result is not surprising: since the row loadings are nearly identical across countries, the impact of the column factor is approximately equal for Germany, France, Italy, and Spain.

Finally, the combination of the row and column factors is presented as a time series in Figure 6. The plot clearly shows that the dynamics of this factor align with major recessionary periods in the Euro Area, such as the crises in 2008, 2012, and 2020. This provides further support for interpreting it as the ”Euro Area Recession Factor,” which captures the majority of common variation in the data.

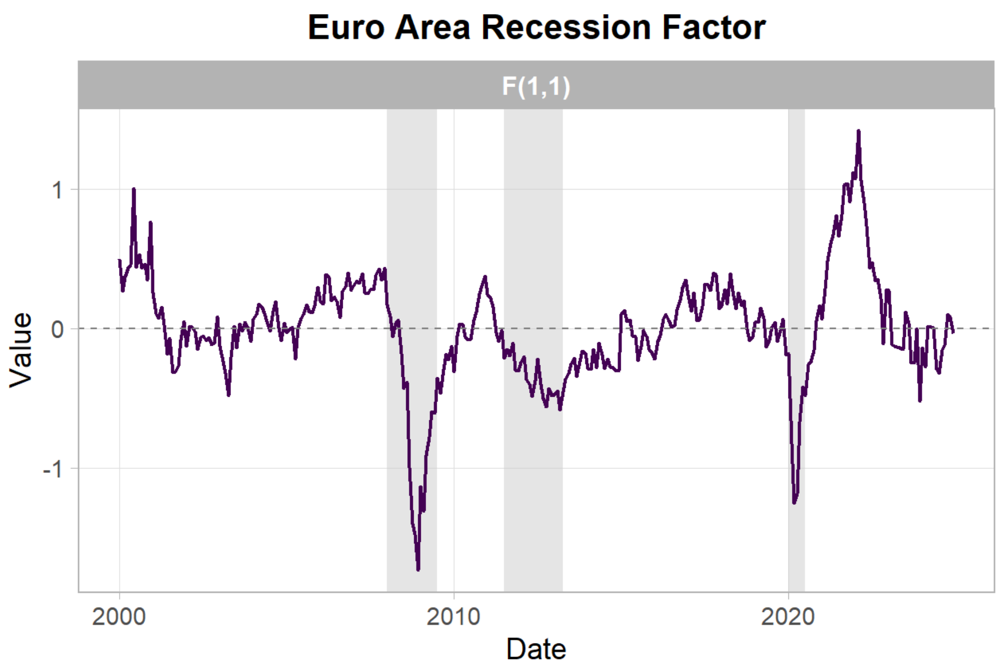


Figure 6: Estimated Euro Area Recession Factor. Grey rectangles indicate periods of recession.

4.3 Euro Area Nowcasting

This subsection presents the empirical application of the estimation and forecasting strategies discussed in Chapter 3, focusing on the Euro Area. The primary aim is to assess the benefits of introducing the Dynamic Matrix Factor Model (*DMFM*) compared to its vector-based counterpart, the Dynamic Factor Model (*DFM*), within a nowcasting framework. Enhancing forecasting performance is a key objective for central banks, as accurate real-time economic signals are crucial for timely and targeted monetary policy interventions. This empirical section also plays a central role in the thesis, as it represents, to the best of the author's knowledge, one of the first applications of nowcasting techniques to matrix-variate time series.

As described in Subsection 4.1, the analysis focuses on the four largest economies of the Euro Area: Germany, France, Italy, and Spain. Table 4.1.2 lists the macroeconomic indicators used in both the DFM and DMFM frameworks. These include monthly indicators and quarterly GDP, totaling 40 variables selected from the dataset compiled by Matteo Barigozzi, Lippi, and Luciani 2021. Previous studies—such as Bańbura and Modugno 2014, Chernis, Cheung, and Velasco 2020, and Giannone, Lenza, and Primiceri 2021—have shown that vector-based DFMs with a small or medium number of variables can outperform high-dimensional setups. This work extends that literature by exploring whether similar results hold under a matrix-factor structure. The model architecture follows the mixed-frequency nowcasting approach of Giannone, Reichlin, and Small 2008, where high-frequency indicators are used to predict a single low-frequency target: GDP.

Table 4.1.2 serves a dual purpose. Beyond listing the variables, it also reports their publication lags (column "Delays"), measured in days from the end of the reference period. This timing information is used to simulate a pseudo real-time nowcasting scenario. Specifically, for each nowcast date in the evaluation window (January 2017 to April 2025), the dataset is recursively truncated to reflect the availability of indicators according to their actual release schedules. For instance, GDP data for Q1 2025 is assumed to be released on April 1st, rather than in mid-April. Additionally, the table classifies variables (e.g., Real, Nominal, Financial, Confidence), which is crucial during the treatment of the COVID-19 period. Given the extreme disruptions, real variables—being the most affected—were excluded from the state-space during the pandemic, allowing the Kalman filter and smoother to operate without introducing excess noise.

Forecasting accuracy, measured by the Root Mean Squared Forecast Error (RMSFE), is evaluated separately for the pre- and post-COVID periods. The pandemic window is defined from March 2020 to June 2021. Although the formal recession in Europe lasted only a few months, its economic effects persisted well beyond that horizon, motivating an extended exclusion of real variables during this phase.

The objective of this analysis is to highlight the usefulness of the *DMFM* in nowcasting applications. Unlike the DFM, the matrix-based structure enables the joint modeling of country-specific time series, preserving the natural structure of data collection and allowing for the extraction of both row and column factors. This is particularly relevant for central banks and institutions monitoring the monetary union, where exploiting cross-country co-movements can yield valuable insights. Since the countries under analysis belong to a monetary union, modeling them jointly may help uncover common latent patterns and interdependencies that are otherwise missed.

The key hypothesis tested in this section is whether incorporating cross-country monthly indicators improves the forecasting performance for each individual country.

Thus, the two main goals of this empirical analysis are:

- To assess the nowcasting performance of the DMFM using the DFM as a benchmark, identifying in which countries the matrix formulation proves more effective;
- To verify whether a small set of monthly indicators is sufficient to produce reliable real-time forecasts of quarterly GDP.

The results suggest that monthly indicators alone can be sufficient for accurate nowcasting—whether in vector or matrix form—and that the DMFM offers tangible improvements by capturing cross-sectional linkages. However, these benefits come at a computational cost: matrix-based estimation, particularly with missing data, is significantly more demanding than vector-based approaches. Nevertheless, for some countries that share stronger co-movements, the DMFM outperforms the DFM in terms of nowcasting precision. The section concludes with a discussion of the contribution of this study and possible directions for future research.

4.3.1 DMFM vs DFM

Finally, this subsection presents the results of the nowcasting analysis applied to tensor-structured data, in comparison with the standard vector-based approach.[6]

Before proceeding, it is important to clarify that the comparison between the matrix and vector formulations of the DFM has been designed to ensure maximum fairness: both models are estimated on the same dataset, apply the same masking rules for real variables during the COVID-19 period, and follow the same variable release schedule at the end of the sample. This setup guarantees that any observed differences in performance stem from the model structures themselves rather than from differences in the data.

This comparison represents the core empirical contribution of the study, yielding several insights into the strengths and limitations of the two approaches. A first key result is that using a moderate set of monthly indicators—whether within a matrix or vector formulation—can lead to similar overall forecasting performance. However, the main finding is that the matrix formulation often offers superior interpretability and responsiveness to data updates, particularly when tracking the dynamics of individual countries. The sequential incorporation of new information within each month of a quarter is more clearly reflected in the tensor-based approach, although vector DFMs remain highly competitive and computationally efficient. Despite broadly comparable RMSFE values, the matrix model shows clear improvements for Germany, particularly in the post-COVID period. This suggests that leveraging country-specific information jointly can enhance predictions of Germany's business cycle. Italy exhibits a similar pattern after the pandemic, while France and Spain perform better under the vector model.

[6] Refer to Subsection 3.4 for details on the recursive nowcasting procedure.

Table 5: RMSFE Pre/Post COVID for DMFM and DFM Models for EA Countries

Country	**Month**	**DMFM**		**DFM**	
		Pre-COVID	*Post-COVID*	*Pre-COVID*	*Post-COVID*
Germany	M1	0.5349	0.7119	0.6493	1.0490
	M2	**0.5135**	**0.6858**	0.6512	0.9078
	M3	0.5286	0.6988	0.6294	0.8488
France	M1	0.4593	0.7727	0.4505	0.6992
	M2	*0.4434*	0.7519	0.4468	**0.6861**
	M3	0.4544	0.7670	0.4405	0.6895
Italy	M1	0.3222	0.8192	0.3031	1.0025
	M2	0.3084	**0.8045**	0.2964	1.4274
	M3	0.3186	0.8185	**0.2953**	1.3826
Spain	M1	0.5968	0.7332	0.3882	0.6004
	M2	0.5765	0.6994	**0.3782**	0.5828
	M3	0.5898	0.7161	0.3796	**0.5761**

Note: RMSFE = Root Mean Squared Forecast Error. Values are computed separately for DMFM and DFM models, pre- and post-COVID, and for each informational month (M1, M2, M3). Best RMSFE values are highlighted in bold.

Table 5 reports the RMSFE values for each country and month within the current quarter ($Q(0)M_1$, $Q(0)M_2$, and $Q(0)M_3$), evaluated against the true quarterly GDP value released approximately one month after quarter-end. This evaluation follows the framework of Bańbura and Modugno 2014, which originally proposed the recursive nowcasting setup for the DFM.

Interestingly, the DMFM consistently performs best in the second month (M2) of the quarter, when the data set expands to include new soft indicators and delayed releases from the first month—typically published within 30 days. This suggests that confidence and survey-based variables are reliable early signals of economic conditions, and that most of the predictive information for GDP is available by the end of M2. In contrast, the DFM model generally achieves its lowest RMSFE in the third month (M3), reflecting the cumulative benefit of increased information availability as the quarter concludes. That the DMFM does not improve in M3 as expected suggests the introduction of additional noise, possibly due to the model's sensitivity to late releases.

While RMSFE differences between the two models are generally small, Germany stands out as the country where the DMFM delivers consistently better results, both before and after the COVID-19 crisis. As shown in Figure 7, the DMFM nowcasts for Germany are not only more precise but also more responsive to economic updates. Notably, the DMFM responds immediately to the onset of the COVID-19 crisis, whereas the vector-based DFM shows a lagged reaction.

The reason why Germany benefits the most from the matrix formulation may be tied to its strong integration in the European economy. Germany's industrial output and macroeconomic trends are

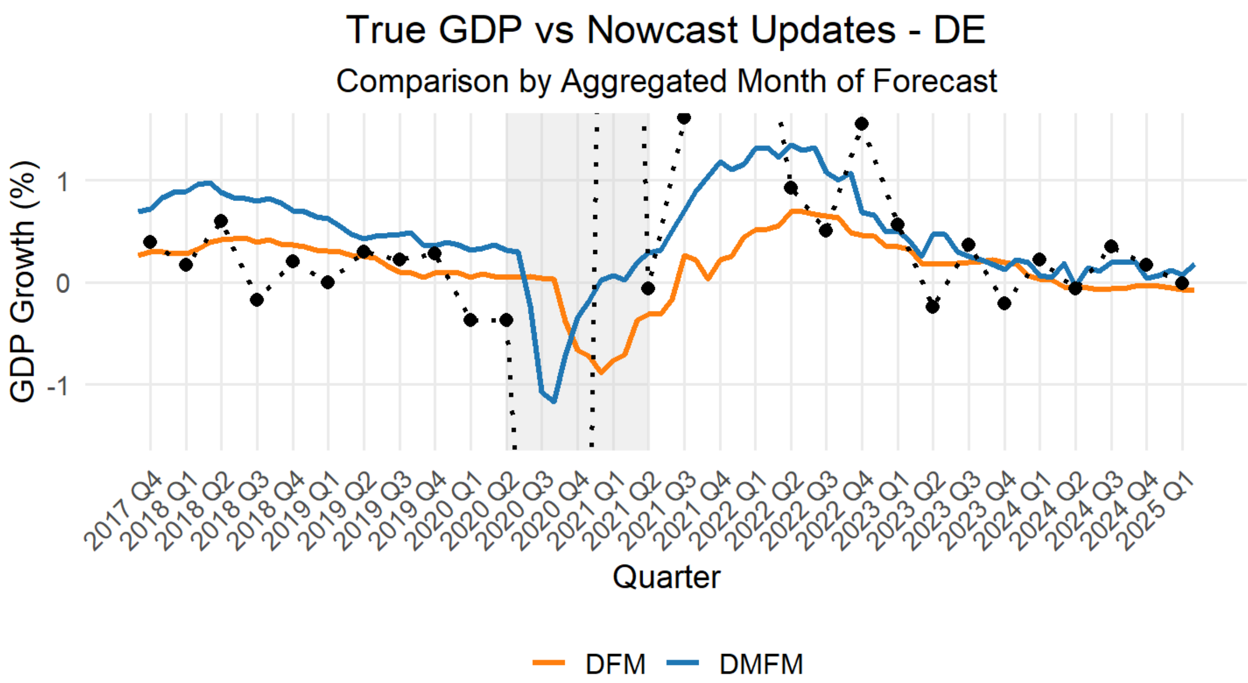


Figure 7: Nowcast Time Series for Germany: DMFM vs. DFM (2017Q4–2025Q1)

deeply linked to those of its neighbors, and pooling country-specific information from across the Euro Area improves the precision of its nowcasts. For other countries, results are more nuanced. France and Spain appear to be driven more by idiosyncratic national dynamics, making the vector DFM more suitable. Italy shows a hybrid behavior, with matrix-based advantages emerging primarily in the post-COVID period.

The general insight from these findings is the following: the more a country's economic performance is linked to broader Euro Area dynamics, the more it can benefit from a DMFM structure. This observation will be revisited in the final subsection, which outlines future research directions. Incorporating additional cross-country features such as trade flows, labor mobility, or supply chain integration could further enhance the effectiveness of matrix-based nowcasting models.

From Table 5, it is evident that post-COVID performance for Italy is also significantly improved, as can be also visualized in figure 8. This finding further reinforces the idea that the more interconnected a country is, the more beneficial it is to use a DMFM for analysis.

From Figures 7 and 8, and also those reported in the Appendix for France and Spain,Figures 14 and 15 respectively, one can observe the nowcast time series from 2017 to the last available quarter in 2025. These time series show the monthly updates of the nowcasts for the reference quarter, both for the DMFM and the DFM. The grey area represents the COVID-19 period, namely from March 2020 to June 2021, during which real variables were dropped due to the pandemic and were instead forecasted through the Kalman filter. From the time series, one can observe similar patterns in both the vector and matrix formulations of the DFM. However, some relevant differences emerge. Notably, the *DMFM* responds more rapidly to new information, particularly during the COVID-19 period, while the *DFM* in vector form exhibits a more delayed adjustment. Overall, the superior performance of Germany and Italy can be attributed to their alignment with the broader Euro Area business cycle, while for France

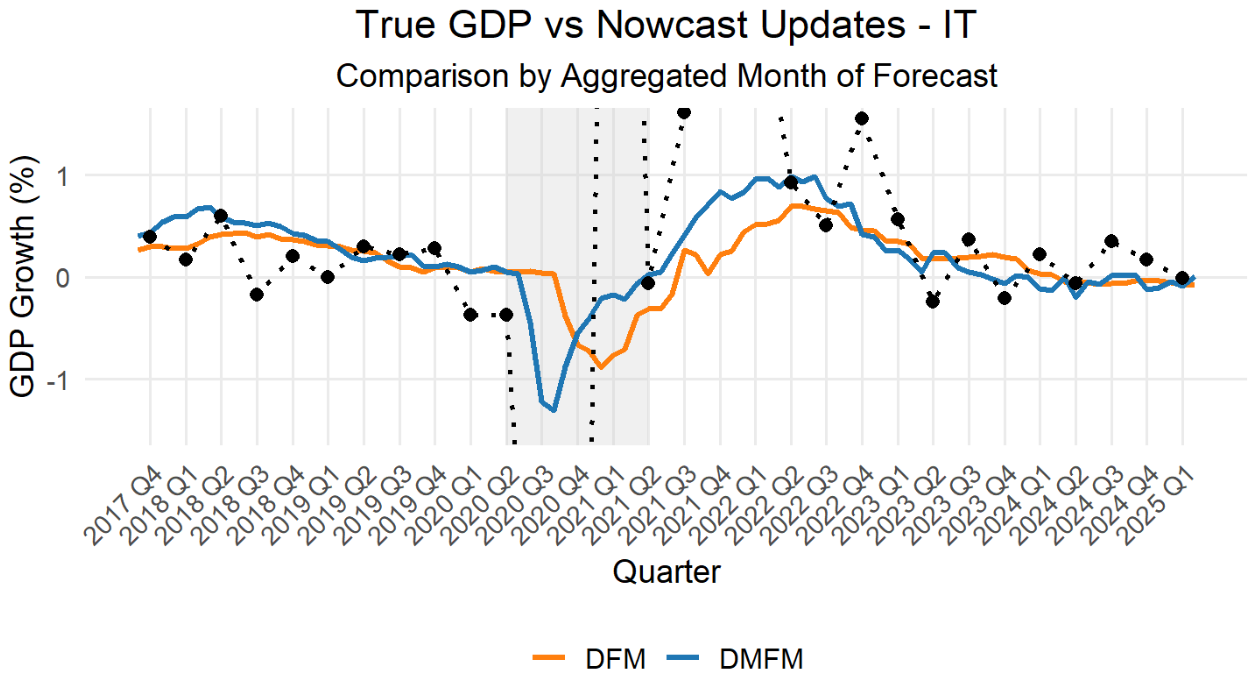


Figure 8: Nowcast comparison between DFM and DMFM (2017Q4–2025Q1)

and Spain, a vector DFM may suffice. The latter remains a valid choice, as it is less computationally demanding and faster to estimate.

Now, once the nowcast time series describing the evolution of the updates during the quarter has been discussed, it is also convenient to analyse separately the time series for each month of update. Namely, it is useful to visualize how much the nowcast changes from one month of the quarter to another. As the information set expands and we get closer to the end of the quarter, more information becomes available and we should observe an increase in forecast precision.

From Table 5, we observed that this is generally true for the DFM, but not for the DMFM, suggesting that some noise may confuse the matrix model. Nevertheless, we still expect some reaction to the information flow introduced as time passes. Figure 9, which shows the nowcasts for each month of the quarter separately for Italy, clearly illustrates that even if the *DFM* exhibits better performance in the pre-COVID period, its nowcast series is quite flat, and even after COVID, it does not respond significantly to new information. Moreover, the updates between months are not particularly pronounced.

On the other hand, the *DMFM* demonstrates greater responsiveness, particularly in the second and third months of the quarter, when additional news is expected to be released, thus enhancing the performance of the nowcast series obtained in the first month. This figure is particularly relevant for understanding how updates in each month affect changes in the nowcasts.

To gain insights into the distribution of the forecast errors and compare the two models, a boxplot can be useful. However, it does not reveal a clear preference for the DFM in matrix form with respect to the vector form, as shown in Figures 16 and 17 in the Appendix.

Overall, these nowcasting results highlight two key facts: first, the DMFM can indeed improve nowcasting performance under certain conditions—probably depending on the degree of interconnection

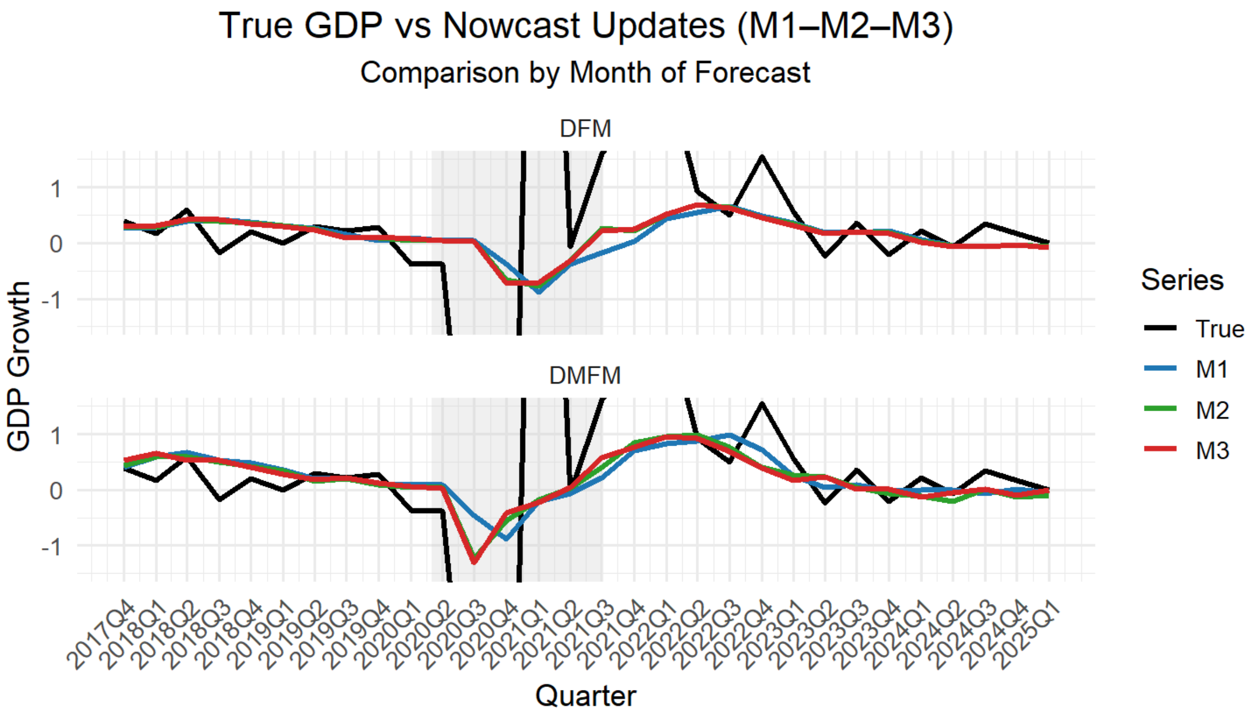


Figure 9: Nowcast responsiveness by each month of the quarter for Italy

between the individual observation and the whole matrix. In our case, this dependency is represented by the country-specific sensitivity to the matrix, driven by the Euro Area recession factor.

Second, monthly indicators alone are sufficient to successfully forecast GDP within a DMFM. However, nothing prevents the inclusion of additional quarterly variables. In that case, the matrix-form DFM may become significantly more computationally demanding, but it could also yield further improvements.

Using a medium-sized set of variables, consisting of approximately 40 monthly indicators, appears to be a sufficient approach for producing forecasts in Germany and Italy—especially in the post-COVID period. This is consistent with Giannone, Reichlin, and Small 2008 and Bańbura and Modugno 2014, who emphasize the superiority of small and medium-scale models over large high-dimensional settings. This conclusion still needs to be extended to the DMFM framework and represents an interesting direction for future work.

For France and Spain, the DFM remains the best strategy for obtaining nowcasts of the current quarter. Overall, the *DMFM* tends to perform better for Germany, but in the other countries there is no clear evidence of systematic improvement. Italy, in particular, shows similar results in the pre-COVID period but favorable outcomes for the DMFM after the pandemic. From this evidence, one could infer that Euro Area countries, when considered jointly, provide a valuable information set for forecasting the region's main economic drivers, especially those more closely aligned with the aggregate cycle.

The economic intuition behind this result is that smaller economies—typically considered followers—can collectively improve the forecast of a larger economy like Germany, which often acts as a leader. Periods of expansion or recession in Europe may bring both positive and negative signals even to leading economies, justifying the use of a matrix factor model to incorporate cross-country interactions more effectively.

Finally, since during nowcasting the masked dataset is recurivelly estimated period after period through the EM alghoritm, it is also possible to visualize the smoothed estimates for each country. Overall, this estimates are sufficiently reliable,as figure 10 shows, assessing the validity of this work [7].

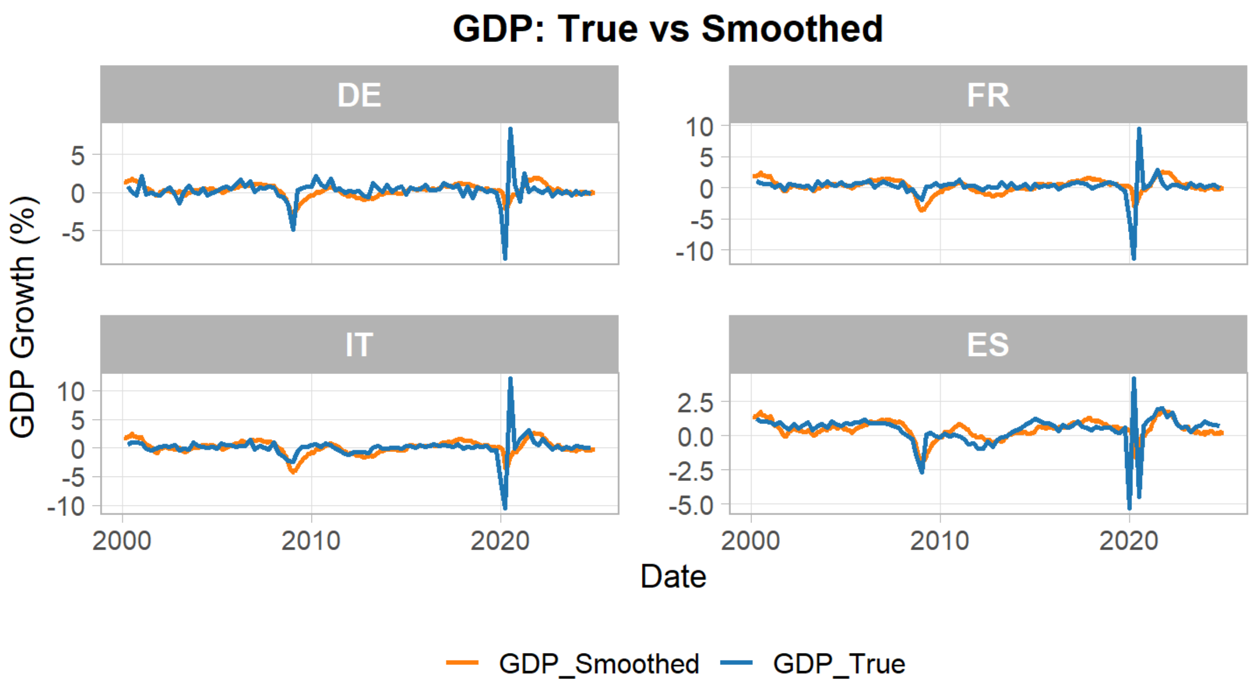


Figure 10: Smoothed estimates of GDP

For the last iteration over time, namely considering the full data available untill 2025, figure 18 in Appendix shows the convergence of the EM.

4.3.2 Contributions and Future Works

Throughout this work, several contributions have been made in the field of forecasting applications. Starting from the need to preserve the matrix structure of the data, I adopted the Dynamic Matrix Factor Model (DMFM), while also addressing the presence of missing observations due to mixed-frequency data and the irregular patterns caused by the economic disruptions of the COVID-19 pandemic. Simulating a pseudo real-time exercise, I estimated the quasi-log-likelihood using the EM algorithm and recursively produced nowcasts of the target quarterly variable—GDP—via the Kalman filter. The performance of the DMFM was then compared to that of the standard Dynamic Factor Model (DFM). These nowcasts were based on factors extracted from a matrix composed of four Euro Area (EA) countries and a set of monthly indicators, including the GDP target variable.

The primary contribution lies in applying nowcasting techniques to a small-scale tensor-structured dataset. A key finding is the substantial improvement in nowcasting accuracy for Germany when using the matrix formulation, compared to the vector approach. This suggests that applying the DMFM may be particularly beneficial for countries that act as economic leaders within the matrix, as they

[7] In the Appendix figure 19 provides a focus on Italy

are more interconnected with other economies. For these countries, incorporating information from other members enhances forecast quality. Conversely, for countries with weaker interconnections, the simpler vector-based DFM may still yield satisfactory results with lower computational costs.

Nonetheless, further research could strengthen and generalize the approach. Several assumptions were made during the empirical implementation, and this final section reflects on the sources of improvement and proposes avenues for future development.

First, the transition equation in Equation 3 could be extended to allow for higher-order lags. In this thesis, factors are assumed to follow a first-order matrix autoregressive (MAR(1)) process. However, model selection via BIC—shown in Figure 20 in the Appendix—suggests that two lags may provide a better fit. Extending the model to accommodate a MAR(2) structure would require adapting the DMFM to the companion form.

A second potential improvement involves expanding the analysis to include additional countries and variables. This would help verify the idea that small- and medium-scale models often outperform large-scale models in forecasting accuracy. Thanks to the modular nature of the R implementation, this extension can be easily implemented through the provided replication code.

The most promising avenue for future work, however, involves decomposing the forecast revisions not only by variable or group of variables, but also by country—identifying which member of the matrix is driving the nowcast revision for a given target country.

To illustrate this, we first clarify what is meant by forecast revision decomposition. Using the vector-form DFM applied to GDP nowcasting, I tracked updates from mid-June 2024 to mid-December 2024, following the methodology of Bańbura and Modugno 2014. Each monthly update reflects contributions

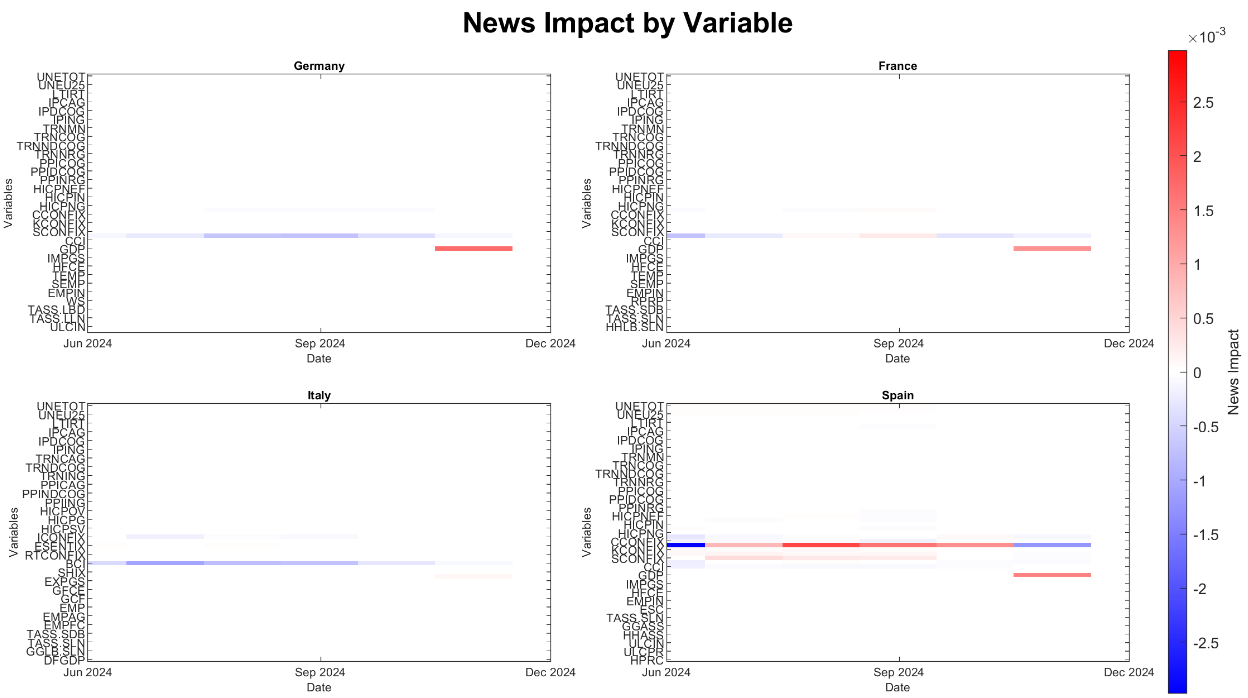


Figure 11: Decomposition of GDP Forecast Revisions by Variable Type

from groups of variables, as specified in Table 4.1.2. Forecast revisions are driven by both the magnitude of the news (unexpected data) and the variable's relevance to the model. For simplicity, I focused on the component of the revision due to the news itself. This aligns well with existing literature.

Figure 11 shows the decomposition of forecast updates by variable for each EA country. A consistent finding is that soft data, especially confidence indicators, contribute far more to updates than other variables. When grouping variables, confidence indicators (CIs) consistently emerge as the most impactful. News from national accounts or labor market indicators contributes more gradually as the quarter progresses, as shown in figure 12.

These findings confirm those of previous studies (e.g., Giannone, Reichlin, and Small 2008, Bańbura and Modugno 2014) and highlight the critical role of timely survey and confidence data, especially in the early stages of the quarter.

An innovative extension to this framework would be to decompose forecast revisions not only by variable or group, but also by country-level origin. In other words, for each country in the matrix, one could estimate how much of the forecast revision is driven by data updates from other countries. For instance, one could identify if Germany's nowcast contributes more significantly to improving Italy's forecast than data from France or Spain. This type of analysis would allow us to learn not only from national data releases but also from how information propagates across countries. Such a design would resemble neural network architectures used for modeling interdependencies in complex systems, where the direction and magnitude of information flow can be learned from data.

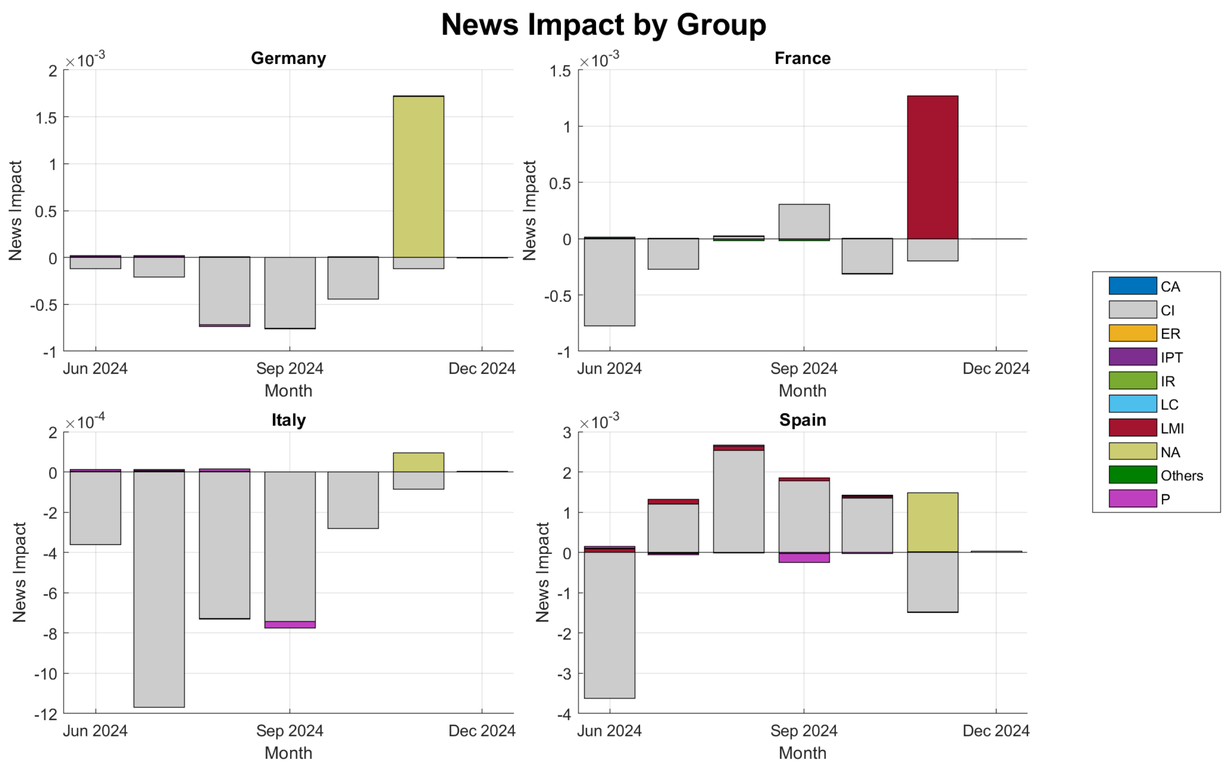


Figure 12: Decomposition of GDP Forecast Revisions by Variable Group

5 Conclusion

In the economic and financial fields, data are often collected in matrix form. Economists and econometricians have recently explored methodologies capable of preserving such structures for macroeconomic analysis. The taxonomy of matrix data analysis, viewed as a special case of tensor analysis, begins with the static Matrix Factor Model introduced by Wang, Liu, and R. Chen 2019. More recent contributions, such as Matteo Barigozzi and Trapin 2025, have extended this framework by incorporating a Matrix Autoregressive (MAR) structure, resulting in a state-space formulation known as the Dynamic Matrix Factor Model (*DMFM*). This model provides an appealing strategy for addressing high-dimensional datasets, as it extracts the main sources of variability through latent factors and models their dynamics, making it particularly well-suited for forecasting exercises.

Building on the estimation approach proposed by Matteo Barigozzi and Trapin 2025, this thesis applies the *DMFM* to a nowcasting framework using Euro Area data from Matteo Barigozzi, Lippi, and Luciani 2021. The adaptation of the EM algorithm to matrix-valued time series allows to provide a novel contribution by applying nowcasting techniques to matrix-variate datasets. Moreover, the use of the EM algorithm allows the model to handle multiple layers of complexity, including general patterns of missing data arising from mixed-frequency structures and the COVID-19 pandemic, which introduced systematic blocks of missing observations in real variables.

The core contribution of this thesis lies in the empirical implementation of nowcasting within the DMFM framework. A pseudo real-time forecasting exercise was conducted by updating the dataset monthly, re-estimating the model via the EM algorithm, and storing successive nowcast revisions. This setup also enabled a direct comparison between two competing formulations of the Dynamic Factor Model: the traditional *vector-based* DFM (estimated separately for each country) and the *matrix-based* DMFM.

To summarize, the methodological framework developed in this thesis addresses three key challenges that enhance its applicability:

1. It accommodates tensor-valued data through the DMFM structure;
2. It enables estimation via a misspecified Gaussian log-likelihood, solved through the EM algorithm in closed form, and handles general patterns of missing data;
3. It incorporates the Kalman filter within the EM algorithm to produce real-time nowcasts of quarterly GDP.

The complete framework is applied to four major Euro Area economies: Germany, France, Italy, and Spain. Using a dataset of 39 monthly indicators and quarterly GDP I simulate a realistic real-time nowcasting environment where the dataset is recursively updated month by month, with variables masked according to their actual release schedules.

The empirical findings demonstrate that incorporating monthly information is sufficient to generate reliable macroeconomic nowcasts. More importantly, they reveal that the GDP of a given Euro Area country can be more accurately predicted by leveraging information from other countries through a shared latent structure. In particular, the results show that, under certain conditions, the DMFM outperforms the country-specific vector-based DFM in terms of nowcasting accuracy. Notably, Germany and (post-COVID) Italy exhibit significant improvements, likely due to their greater interconnectedness within the European economic network. Conversely, France and Spain are better forecasted using the vector-based model, possibly because their economic dynamics are more driven by domestic factors.

These findings point to a promising direction for future research: decomposing nowcast revisions not only by variable or group of variables, but also by country of origin. Such decomposition would allow researchers to identify which countries and variables contribute most to each country's economic outlook, thereby offering a causal interpretation of interdependencies within a monetary union.

In conclusion, this thesis introduces a flexible structure capable of handling both COVID-related disruptions and mixed-frequency datasets. It constitutes an important first step toward applying nowcasting techniques to tensor-structured data, at least in the matrix-valued case.

References

Bańbura, Marta and Michele Modugno (2014). "Maximum likelihood estimation of factor models on datasets with arbitrary pattern of missing data". In: *Journal of applied econometrics* 29.1, pp. 133–160.

Barigozzi, M. and C. Lissona (2024). *EA-MD-QD: Large Euro Area and Euro Member Countries Datasets for Macroeconomic Research (Version 12.2023)*. Data set. URL: https://doi.org/10.5281/ZENODO.10514668.

Barigozzi, Matteo, Marco Lippi, and Matteo Luciani (2021). "Large-dimensional dynamic factor models: Estimation of impulse–response functions with I (1) cointegrated factors". In: *Journal of Econometrics* 221.2, pp. 455–482.

Barigozzi, Matteo and Luca Trapin (2025). "Quasi maximum likelihood estimation of high-dimensional approximate dynamic matrix factor models via the EM algorithm". In: *arXiv preprint arXiv:2502.04112*.

Billio, Monica et al. (2023). "Bayesian dynamic tensor regression". In: *Journal of Business & Economic Statistics* 41.2, pp. 429–439.

Cascaldi-Garcia, Danilo et al. (2024). "Back to the present: Learning about the euro area through a now-casting model". In: *International Journal of Forecasting* 40.2, pp. 661–686.

Cen, Zetai and Clifford Lam (2025). "Tensor time series imputation through tensor factor modelling". In: *Journal of Econometrics* 249, p. 105974.

Chen, Elynn Y and Jianqing Fan (2023). "Statistical inference for high-dimensional matrix-variate factor models". In: *Journal of the American Statistical Association* 118.542, pp. 1038–1055.

Chen, Elynn Y, Ruey S Tsay, and Rong Chen (2020). "Constrained factor models for high-dimensional matrix-variate time series". In: *Journal of the American Statistical Association*.

Chen, Rong, Han Xiao, and Dan Yang (2021). "Autoregressive models for matrix-valued time series". In: *Journal of Econometrics* 222.1, pp. 539–560.

Chernis, Tony, Calista Cheung, and Gabriella Velasco (2020). "A three-frequency dynamic factor model for nowcasting Canadian provincial GDP growth". In: *International Journal of Forecasting* 36.3, pp. 851–872.

Dempster, Arthur P, Nan M Laird, and Donald B Rubin (1977). "Maximum likelihood from incomplete data via the EM algorithm". In: *Journal of the royal statistical society: series B (methodological)* 39.1, pp. 1–22.

Doz, Catherine, Domenico Giannone, and Lucrezia Reichlin (2012). "A quasi–maximum likelihood approach for large, approximate dynamic factor models". In: *Review of economics and statistics* 94.4, pp. 1014–1024.

Durbin, James and Siem Jan Koopman (2012). *Time series analysis by state space methods*. Oxford University Press (UK).

Giannone, Domenico, Michele Lenza, and Giorgio E Primiceri (2021). "Economic predictions with big data: The illusion of sparsity". In: *Econometrica* 89.5, pp. 2409–2437.

Giannone, Domenico, Lucrezia Reichlin, and David Small (2008). "Nowcasting: The real-time informational content of macroeconomic data". In: *Journal of monetary economics* 55.4, pp. 665–676.

Kapetanios, George, Laura Serlenga, and Yongcheol Shin (2021). “Estimation and inference for multi-dimensional heterogeneous panel datasets with hierarchical multi-factor error structure”. In: *Journal of Econometrics* 220.2, pp. 504–531.

Lam, Clifford and Qiwei Yao (2012). “Factor modeling for high-dimensional time series: inference for the number of factors”. In: *The Annals of Statistics*, pp. 694–726.

Tipping, Michael E and Christopher M Bishop (1999). “Probabilistic principal component analysis”. In: *Journal of the Royal Statistical Society Series B: Statistical Methodology* 61.3, pp. 611–622.

Tsay, Ruey S (2024). “Matrix-Variate Time Series Analysis: A Brief Review and Some New Developments”. In: *International Statistical Review* 92.2, pp. 246–262.

Wang, Dong, Xialu Liu, and Rong Chen (2019). “Factor models for matrix-valued high-dimensional time series”. In: *Journal of econometrics* 208.1, pp. 231–248.

Watson, Mark W and Robert F Engle (1983). “Alternative algorithms for the estimation of dynamic factor, mimic and varying coefficient regression models”. In: *Journal of Econometrics* 23.3, pp. 385–400.

Xiong, Ruoxuan and Markus Pelger (2023). “Large dimensional latent factor modeling with missing observations and applications to causal inference”. In: *Journal of Econometrics* 233.1, pp. 271–301.

Xu, Sainan, Chaofeng Yuan, and Jianhua Guo (2025). “Quasi maximum likelihood estimation for large-dimensional matrix factor models”. In: *Journal of Business & Economic Statistics* 43.2, pp. 439–453.

Yu, Long et al. (2022). “Projected estimation for large-dimensional matrix factor models”. In: *Journal of Econometrics* 229.1, pp. 201–217.

Yu, Ruofan et al. (2024). “Dynamic matrix factor models for high dimensional time series”. In: *arXiv preprint arXiv:2407.05624*.

A Additional Table

Transformation

The EA-MD-QD dataset includes series that are already seasonally adjusted and appropriately transformed. Table 6 summarizes the transformations applied to the subset of variables used in this study and lists the codes identifying each variable's unit of measurement, category, frequency, and transformation. For instance, GDP is transformed using the logarithmic first difference scaled by 100, i.e., $100 \times \Delta \log(x_t)$, approximating its quarterly percentage change. This table also serves as a reference for interpreting Table 4.1.2, which reports the characteristics of the selected variables and adopts the same coding system.

Table 6: Summary of codes for variable unit, category, frequency, and transformation

Unit	**Category**	**Frequency**	**Transformation**
R = Real	S = Soft	Q = Quarterly	0 = No Transformation
N = Nominal	H = Hard	M = Monthly	$1 = 100 \times \Delta \log(x_t)$
F = Financial			$2 = \Delta x_t$
C = Confidence			

B Additional Figures

Anecdotal Evidence

Figure 13 reports the heatmap of pairwise correlations among the complete set of variables present in M. Barigozzi and Lissona 2024.

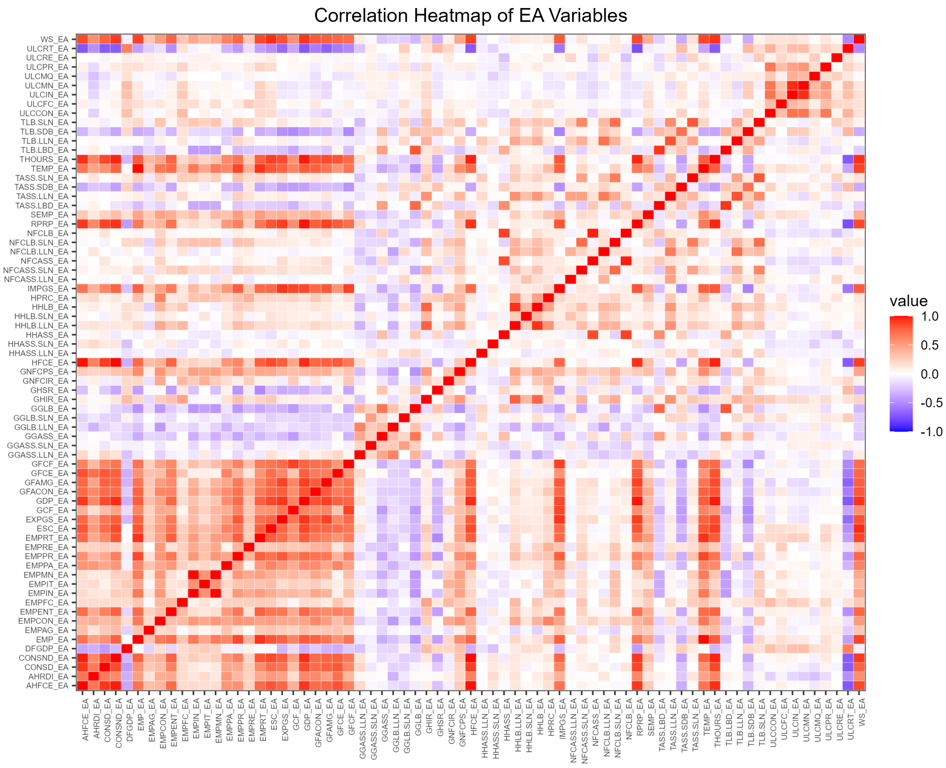


Figure 13: Heatmap of correlations among selected variables in the Euro Area

Nowcasting Performances

Figures 14 and 15 illustrate the quarterly nowcasts of the national GDP, month by month, for France and Spain over the period 2017Q4–2025Q1. Nowcasts of the Dynamic Matrix Factor Model (DMFM) are compared with the once of Dynamic Factor Model (DFM) in vector and the True GDP value.

For both countries, the DFM in vector form provides more accurate estimates of the actual GDP dynamics. However, the DMFM exhibits a quicker reaction during the COVID-19 pandemic and it is more responsive eto monthly updates generated by the releas of variables. This responsiveness suggests that the matrix-based specification may better capture macroeconomic downturns that is a desirable feature for real-time forecasting and policy-making.

However, these models yield comparable error distributions, as illustrated in the boxplots below (Figures 16 and 17) referred to Italy. After Covid-19 the immadiate response of DMFM provided a significantly less propagation of ouliers.

EM Algorithm – Full Sample

Figure 18 represents the Expectation-Maximization (EM) algorithm convergenge for the final iteration of the recursive nowcast procedure, when considering the whole sample until April 2025. After 6 iterations, the quasi log-likelihood stabilizes.

As a result, the smoothed estimate of the GDP series for Italy is presented in Figure 19. The smoother cosiders all available information up to the end of the sample, updating the estimates backward. In

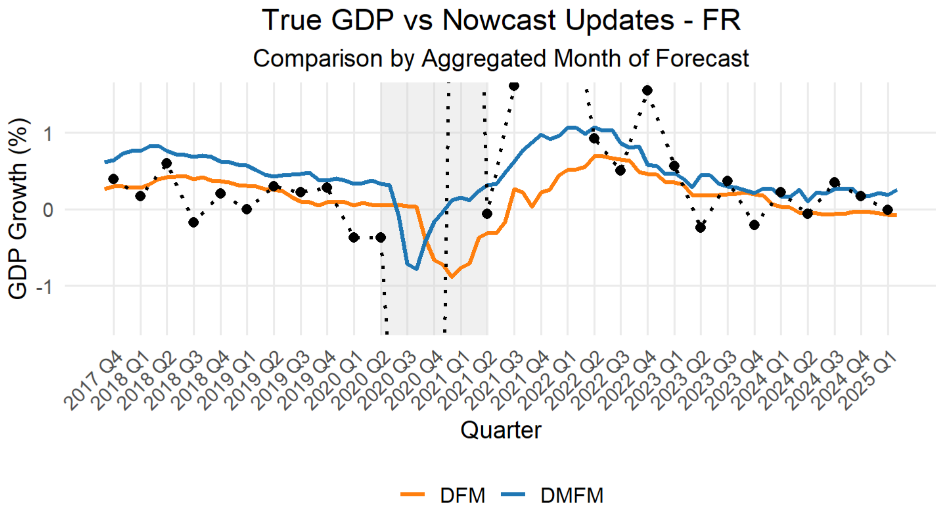


Figure 14: Nowcast comparison between DFM and DMFM (2017Q4–2025Q1) – France

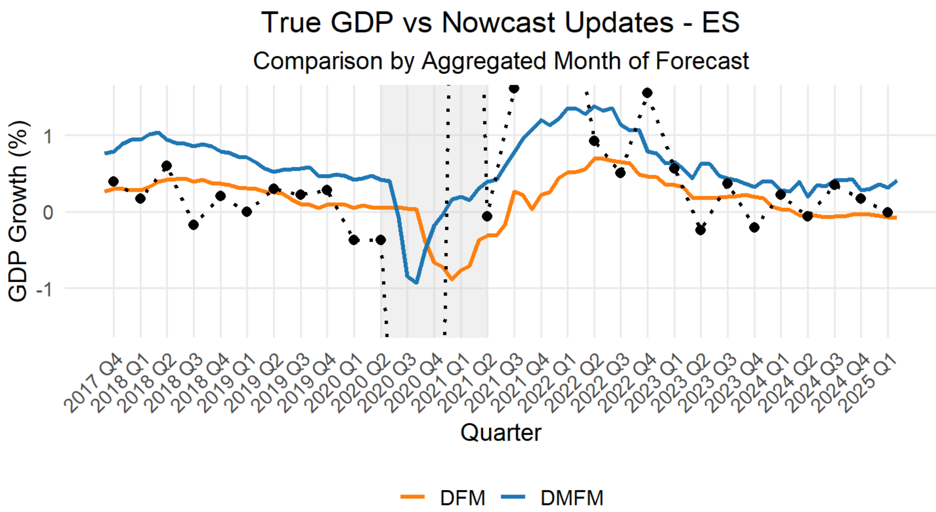


Figure 15: Nowcast comparison between DFM and DMFM (2017Q4–2025Q1) – Spain

other words it provides refined estimate of the latent factor associated with Italian economic activity.

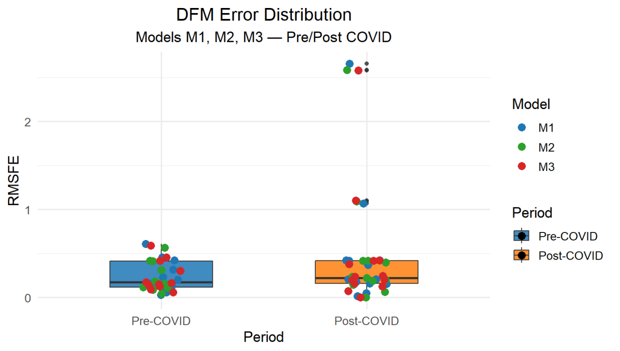


Figure 16: Distribution of forecasting errors – DFM

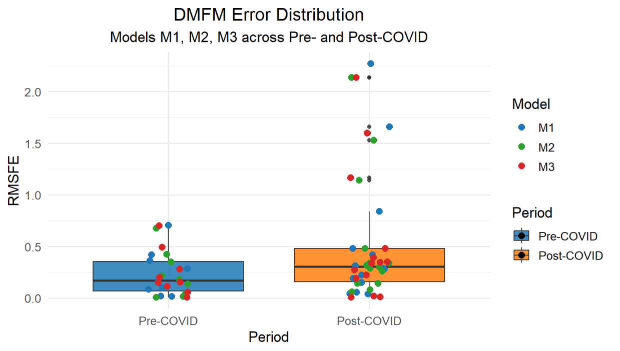


Figure 17: Distribution of forecasting errors – DMFM

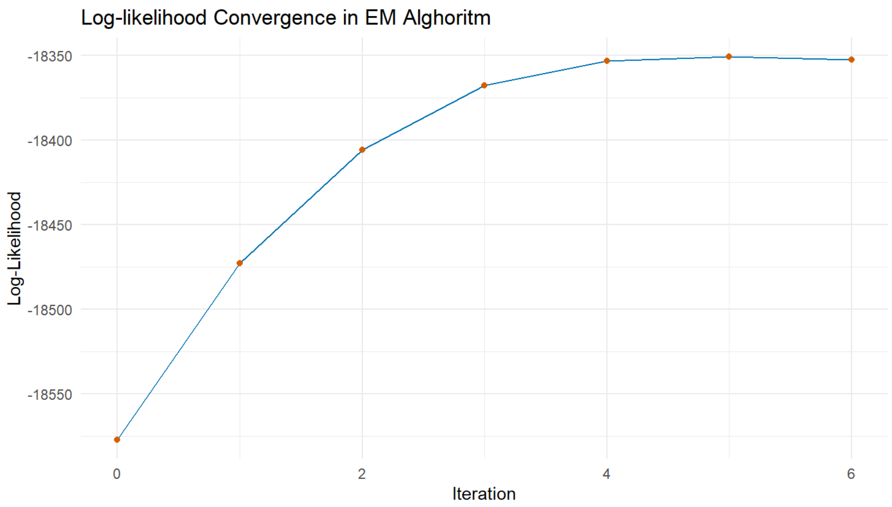


Figure 18: EM algorithm – Quasi log-likelihood convergence

Future Works

A potential refinement of this work involves estending the lag order of the transition equation in the DMFM (Equation 3). Using information criteria such as the Bayesian Information Criterion (BIC) and the Akaike Information Criterion (AIC) figure 20 suggests a MAR process of order 2.

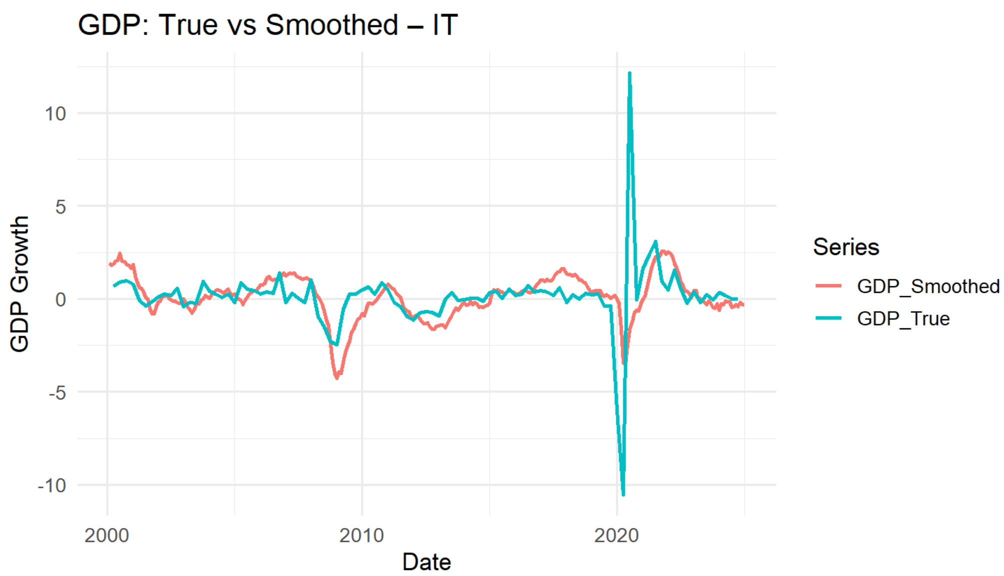


Figure 19: Smoothed estimate of Italian GDP

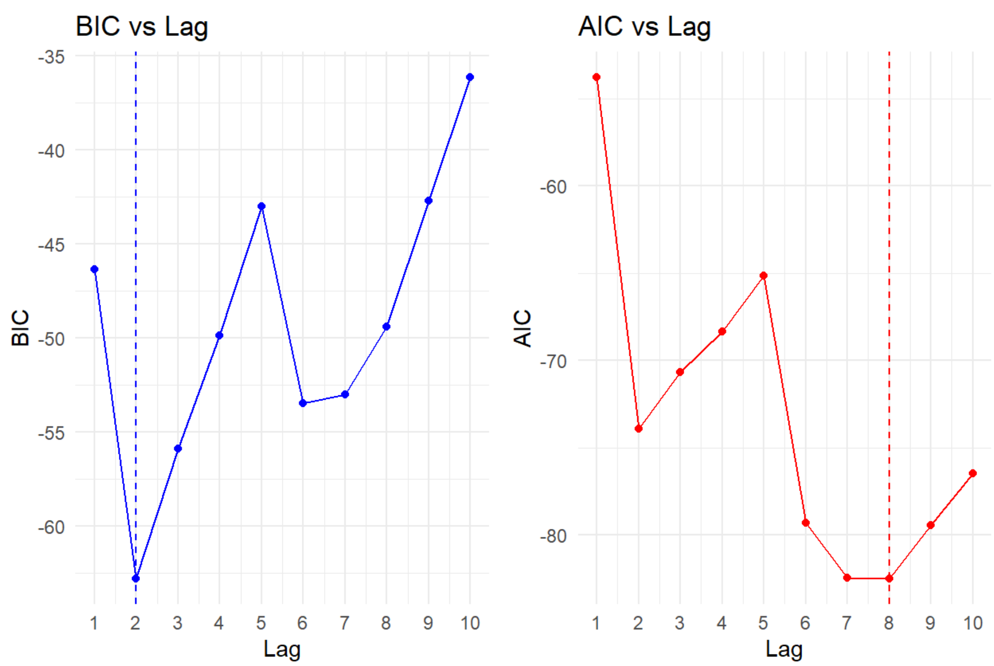


Figure 20: Model selection via BIC and AIC criteria