ALMA MATER STUDIORUM – UNIVERSITÀ DI BOLOGNA

DEPARTMENT OF STATISTICAL SCIENCES

"PAOLO FORTUNATI"

First Cycle Degree / Bachelor in Statistical Sciences

Curriculum Stats&Maths

TITLE

Unemployment Rate Analysis on 30 European Nations

Presented by:

Tasdid Mahmud
1023917

Supervisor:

Prof. Christian Martin Hennig

SESSION July

ACADEMIC YEAR 2024/2025

Unemployment Rate Analysis on 30 European Nations

Tasdid Mahmud

Abstract

This thesis analyses how unemployment rates have changed across 30 European countries from 2013 to 2023, with a focus on the impact of major economic events such as the global financial post-crisis and the COVID-19 pandemic period. This study treats unemployment rates as continuous trends rather than discrete data points, which allows for the general comparison between countries and helps to gain a deeper understanding of how labour markets change over time. This research identifies distinct groups of countries that share similar unemployment patterns, using advanced clustering techniques, and compares the mean trends of individual groups. The main finding is that unemployment rates were very high in some countries at the beginning of the observation period, such as Greece and Spain; nonetheless, most European countries experienced a steady decline in unemployment over the decade, indicating a general economic recovery. Furthermore, it also demonstrates that the differences in unemployment rates between countries have decreased, suggesting a more unified European labour market. These insights could help policymakers and economists better understand the dynamics of the regional labour market and design more effective employment strategies.

1. Introduction

Understanding how unemployment rates change over time is crucial for assessing a country's economic health and evaluating the effectiveness of its policies. While many studies focus on fixed points of time, such an approach can overlook important trends and patterns that develop gradually. This study aims to explore how unemployment rates have changed across 30 European countries from 2013 to 2023, with a focus on how these rates evolved during times of economic recovery and in major events, such as the COVID-19 pandemic.

The study applies a statistical approach known as Functional Data Analysis (FDA) that allows for the treatment of unemployment rates as continuous curves over time rather than just as isolated data points. This approach helps us to understand how unemployment trends shift, identify underlying patterns, and differences among countries. FDA offers a powerful tool for studying time-based phenomena in a more informative way.

In addition to describing general trends, this thesis also seeks to identify clusters of countries that exhibit similar unemployment behaviours. This is done by using the Discriminative Functional Mixture (DFM) model, which groups countries based on the shape and structure of their unemployment curves.

The motivation behind this research lies in its potential applications. By uncovering how unemployment trends have evolved, this study can help policymakers, economists, and international institutions better understand regional labour markets. For example, identifying the countries that experienced strong recovery after economic crises can offer insights into effective employment strategies.

Ultimately, this thesis contributes a detailed, time-aware perspective on unemployment across Europe using modern statistical techniques to go beyond static descriptions and explore the temporal dynamics that shape the labour market.

2. Data Overview

The raw dataset provided by *Eurostat (2025)* contains unemployment rates over an 11-year period from the start of 2013 to the end of 2023 for 30 European countries. Each observation is done with a monthly time frequency on the total age class in the labour force of males and females together. The unit of measure is the percentage of the population in the labour force. No missing values are presented in the dataset.

3. Visualisations

To initiate our analysis, we begin by visualising the average unemployment rate over the observational time period, thereby establishing a foundational understanding of the trend dynamics. This will provide insights into long-term trends and structural changes in the labour market, smoothing out short-term fluctuations caused by seasonal or cyclical factors.

Figure 1 illustrates the increasing average unemployment rates calculated over monthly data, with a range of one positive and negative standard deviation indicated by the red line. Here, Czechia shows the lowest unemployment; on the other hand, Spain and Greece are the highest. The mean unemployment rates for Spain and Greece are around 20%, which is relatively high compared to other European countries. While Czechia has the lowest unemployment rate, it has a higher standard deviation compared to other countries like Germany, Norway and Switzerland. Notably, Switzerland has the lowest standard deviation among all countries in the dataset. For the mean value of each year, I have plotted a heatmap in the following, which will help to understand how the mean values change over the years in observation.

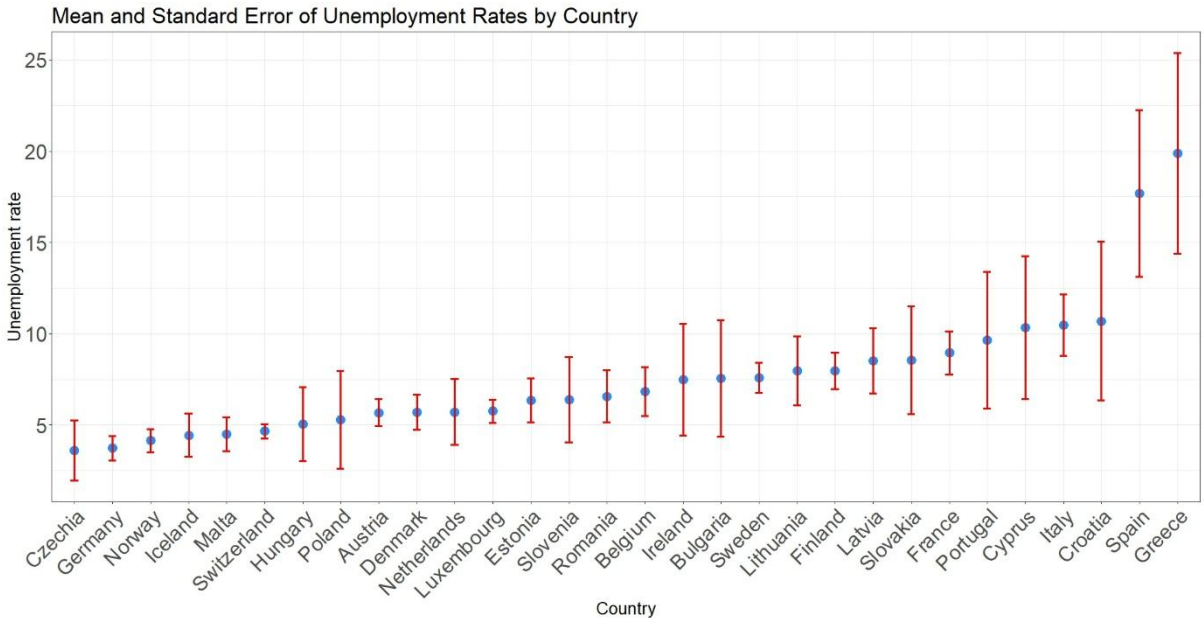

Figure 1: Mean and standard error of unemployment rate of the 11 years with increasing order of mean.

Figure 2 represents a heatmap of the mean values for each country over the years of observation. The colour gradient indicates the magnitude of the unemployment rate, with blue representing lower rates and yellow indicating higher rates. Specifically, values at or below 10% are depicted in blue, rates around 10% to 20% are represented by a blue-yellow transition, and values exceeding 20% are shown in yellow.

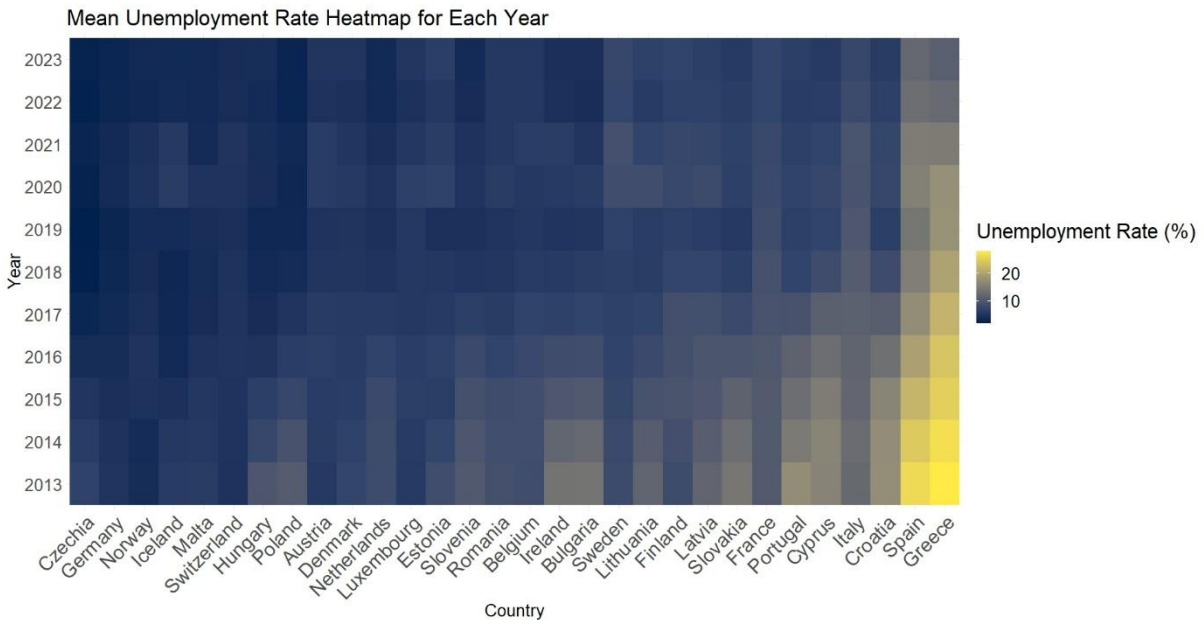

Figure 2: Mean Unemployment rate heat map over observed time period (yellow - higher unemployment rate, blue - lower unemployment rate; see text for details).

In the heatmap, Greece and Spain had extremely high unemployment rates in the initial years of observation among the other countries, as indicated by the noticeable yellow squares in the heatmap. This trend was especially pronounced in the earlier years (2013-2016), indicating

potential long-term economic challenges in these regions. High unemployment signals economic distress, reduces national output and consumption, and increases social and fiscal difficulties. As for Greece, unemployment is severe than in the rest of European countries. It had a profound impact on the severe GDP contraction, collapse in investment, social and demographic effects, and many other consequences. Although unemployment declined over time, it remained higher than in the rest of the European countries, keeping economic growth sluggish.

In contrast, countries such as Germany and Switzerland maintained relatively low unemployment rates throughout the period, as indicated by the consistent blue shading. This reflects a strong economy and improved living standards. Additionally, too low unemployment can create inflationary pressures as employers compete for scarce workers by raising wages, which can lead to higher prices for goods and services.

Some countries, such as Croatia, Cyprus, Portugal, and Slovakia, among others, experienced a moderate increase in unemployment during certain years, which then gradually decreased over time. Furthermore, we can expand the patterns from the heatmap and observe a linear trend in the following plot.

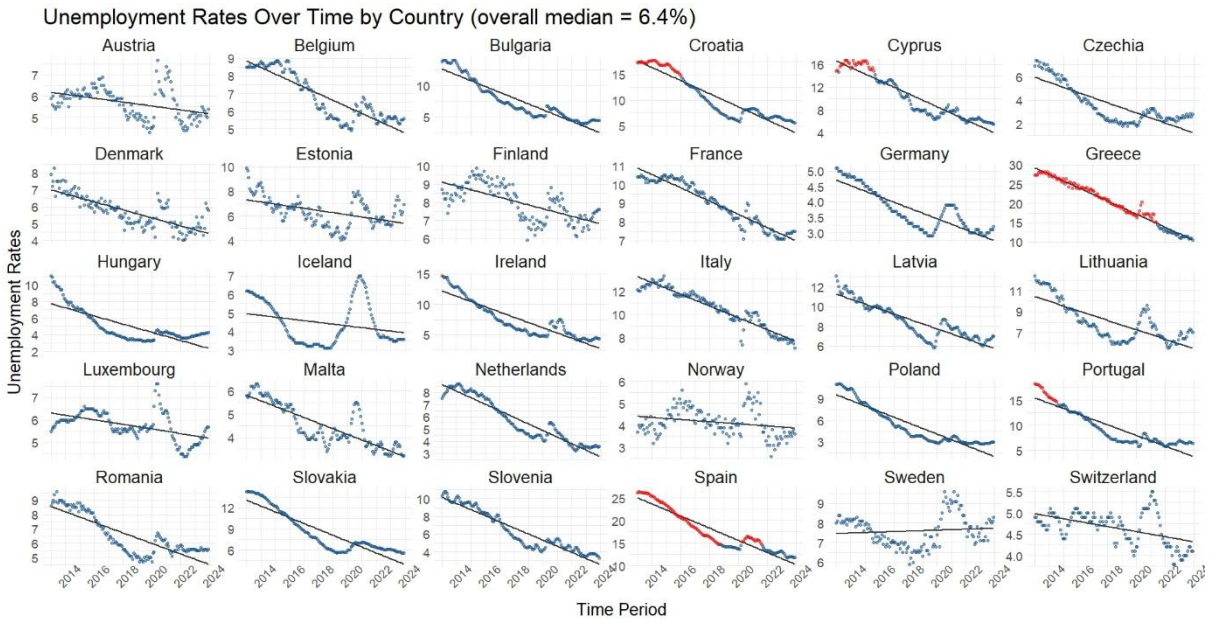

Figure 3: Line graph of unemployment rates over time by individual countries (Red - points above the upper fence, Blue - points below the upper fence; see text for more details).

Figure 3 illustrates linear trends of unemployment rates over time for each country, highlighting that they are significantly different from the overall median value of 6.4%. Points that exceed the overall median by more than 1.5 times the Inter Quartile Range (IQR) of overall data points, known as the upper fence, are flagged in red. It indicates points that are much higher than most data points. Data points that are less than the upper fence are in colour blue. The lower fence is not defined here since the lower fence becomes a negative value, and the unemployment rate is always greater than 0%.

While some countries show extreme unemployment at the beginning of the observation period, the intensity gradually declines over time. The majority of countries exhibit a downward trend in unemployment rates over the observation period, suggesting a broad recovery from the economic challenges of the early 2010s. Many countries show a trend of convergence around the overall median unemployment rate, suggesting disparities in unemployment are declining across the European landscape.

The linear model used in Figure 3 is Ordinary Least Squares (OLS) linear regression, which estimates model parameters by minimising the sum of squared residuals (the difference between the observed and predicted outcomes). This has been used solely to illustrate the linear trend around the data points.

As we have seen, Greece and Spain have the highest unemployment rates among European countries; here we observe a sharp downward trend in these two countries. The red points emphasise their prolonged struggle with high unemployment, especially in earlier years. While both countries exhibit a downward trend, their rates remain substantially higher than most other European nations.

Czechia, Switzerland, Germany, and Norway maintained unemployment rates well below the median throughout the observed period. Their stability signifies efficient labour markets. Croatia, Cyprus, and Portugal have experienced relatively high unemployment in the initial years but exhibited a noticeable decline towards the median over time. Red points in the early segments of their graphs highlight the severity of their initial challenges, while the blue points in later years reflect their improved labour conditions.

Iceland, Sweden, Norway and some other countries exhibit a sharp spike in unemployment, and according to the linear trend, Sweden shows a straight line over time. In these countries, the linear trend line doesn't explain much about the trend. Since their trend is more sinusoidal than linear, a higher-degree polynomial would better describe the trend of these countries than a linear model.

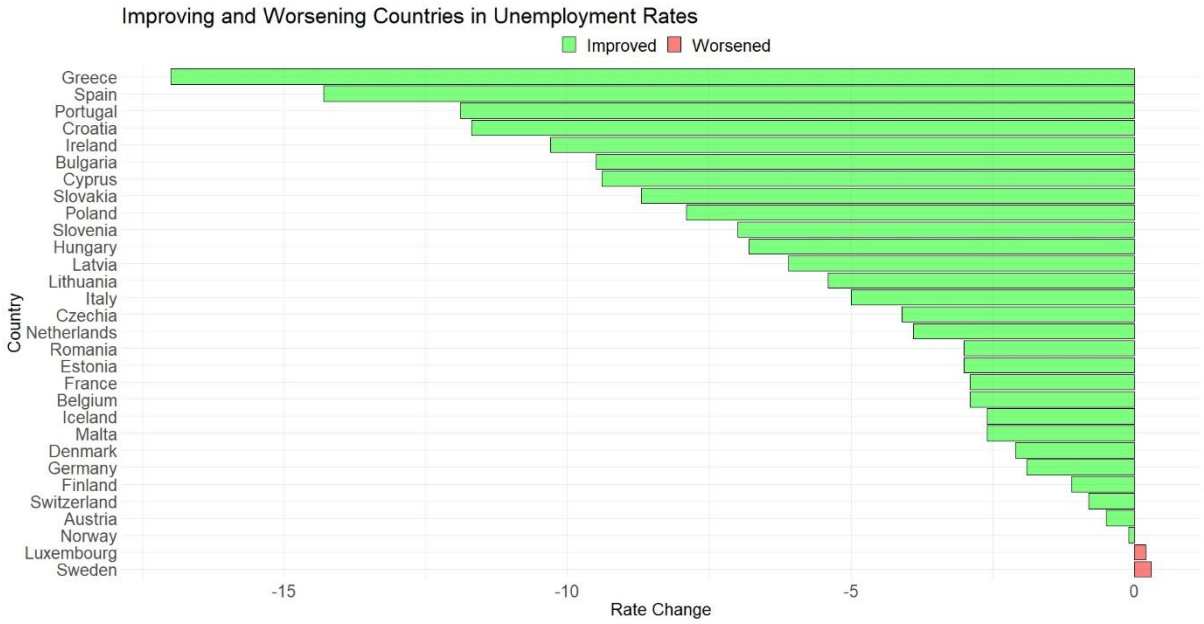

Figure 4: Unemployment rate of change between the starting of 2013 and the end of 2023.

Figure 4 illustrates a horizontal bar plot showing the change in unemployment rate between the start and end of the observation period, with green bars indicating improvement (decrease in unemployment) and red bars indicating worsening (increase in unemployment). The overwhelming majority of countries experienced a reduction in unemployment rates from the start of 2013 to the end of 2023. This reinforces the earlier observation of a general trend towards declining unemployment across Europe during this period. Greece displays the most significant reduction in its unemployment rate during the observational period, recovering from the severe economic crisis of the early 2010s. Only a few countries show worsening or no significant change in unemployment rates.

While the preceding visualisations provide a valuable overview of unemployment trends, they represent only a preliminary step in this analysis. To provide a more rigorous understanding of countries with similar unemployment dynamics, the subsequent sections will give details about the analytical methodology.

4. Methodology

4.1. Functional Data Analysis

Functional Data Analysis (FDA) is a statistical method for data, where observations are functions over a continuous space. These functions are the basic unit of analysis. The most common types of functions have a one-dimensional continuous argument, usually time, although they can also have a multidimensional argument, such as position in space. Some basic theoretical aspects of functional data analysis are presented in the following; more detailed information can be found in *Ramsay, J. O., & Silverman, B. W. (2005)*.

Functional data are observed as a discrete set of n pairs $(t_j, y(t_j))$, where $y(t_j)$ represents a sampled value of the underlying function at time t_j. Observed functional data are single entities, rather than a sequence of individual observations. Furthermore, we assume the existence of a function x that generates the observed data. It is generally desirable to impose a smoothness constraint on x, so that pair of near values $y(t_j)$ and $y(t_{j+1})$ are linked together in such a way that they are unlikely to differ from each other. As noted by *Ramsay, J. O., & Silverman, B. W. (2005)*, if the assumption of smoothness does not hold, there is limited justification for treating the data as functional rather than merely multivariate.

A basis function system consists of a set of mathematically independent known functions denoted as $\psi(t)$, possessing the property that any arbitrary function $x(t)$ can be approximated by a linear combination of a sufficiently large number $p = m + L - 1$ of these functions such that

$$\mathrm{x(t)} = \sum_{\mathrm{i=1}}^{\mathrm{p}} c_i \psi_i(t) = \boldsymbol{c}'\boldsymbol{\psi} = \boldsymbol{\psi}'\boldsymbol{c}.$$

Here, c_i is a constant which can be estimated, $\boldsymbol{\psi}$ is a vector-valued function, $\boldsymbol{c}$ is the vector of constants, and m, L are explained in the next section. Assuming that, $y(t_j) = x(t_j) + \epsilon_j, j = 1, \dots, n$ such that $\epsilon_j \sim N(0, \sigma^2)$ and also Independent and Identically Distributed (I.I.D). Let us

define the $n\ by\ p$ matrix $\boldsymbol{\Psi}$ as containing the values $\psi_i(t_j)$. Then, the estimation of $\boldsymbol{y}$ can be rewritten in matrix form as,

$$\hat{\boldsymbol{y}} = \boldsymbol{\Psi c}.$$

B-spline basis system developed by *de Boor, C, (2001)* has been used to transform discrete values into a functional data.

4.2. B-spline Basis Functions

B-spline basis system by *de Boor, C. (2001)* is most popular, where splines are piecewise polynomials of order m (degree $\mathrm{m}-1$). The spline function $\psi(t)$ for B-spline curve is defined by the Cox-de Boor recursion formula

$$\psi_{i,0}(t) = \begin{cases} 1 & if\ t_i \le t \le t_{i+1} \\ 0 & otherwise \end{cases},$$

$$\psi_{i,j}(t) = \frac{t - t_i}{t_{i+j} - t_i}\psi_{i,j-1}(t) + \frac{t_{i+j+1} - t}{t_{i+j+1} - t_{i+1}}\psi_{i+1,j-1}(t)\ .$$

Here, $j = 0, \dots, m$ and m indicates the order of the polynomial of the spline in each sub-interval and $i = 0, \dots, p-1$. The interval over which a function is to be approximated is divided into L sub-intervals separated by values $\mathrm{t_l}, \mathrm{l} = 1, \dots, \mathrm{L}-1$ that are called internal knots. For example, three internal knots divide the interval into four sub-intervals, including endpoints, and we may number them $\mathrm{t}_0, \dots, \mathrm{t}_L$, where $\mathrm{L} = 4$. The knots could be any non-decreasing vector, but for the B-spline implementation open uniform knot vector is used. This is done by replicating $m-1$ times of first and last knots at the boundaries of the vector with equally spaced knots. One way to define a knot vector would be:

$$\boldsymbol{T} = \left(t_0, t_1, \dots, t_{p-1}\right) with\ t_0, \dots, t_{m-1} = 0;\ t_L, \dots, t_{p-1} = L-1;$$

$$and\ t_i = i - m + 1\ for\ m-1 < i < L\ .$$

For example, with $m = 3$ and $L = 4$, the knot vector is:

$$\boldsymbol{T} = (t_0, t_1, t_2, t_3, t_4, t_5, t_6) = (0,0,0,1,2,3,3,3)$$

In many applications, equal spacing between intervals is the default choice. However, this approach is appropriate only when the data are evenly distributed across the observation period. Since our unemployment rates are based on monthly data, this approach is fine enough for our analysis.

4.3. Smoothing with a Roughness Penalty

Roughness penalty methods are based on optimising a fitting criterion that defines what a smooth of the data is trying to achieve. The concept of roughness penalty in FDA was introduced in the earlier edition of *Ramsay, J. O., & Silverman, B. W. (2005).*

The square of the second derivative of a function $\mathrm{x}(t)$, known as curvature at t, is a popular way to quantify the notion "roughness" of a function. A measure of a function's roughness is

$$\mathrm{PE}N_2(x) = \int [D^2 x(t)]^2 \, \mathrm{dt}.$$

Penalised residual sum of squares for a function, given the discrete data to be smoothed, is defined as

$$\mathrm{PENSS}E_2(x|\boldsymbol{y}) = [\boldsymbol{y} - x(\boldsymbol{t})]^T \boldsymbol{W} [\boldsymbol{y} - x(\boldsymbol{t})] + \lambda \times PEN_2(x)\,.$$

Here, the first part of the summation is known as the Sum of Squared Error or SSE with differential weighting of residuals. The vector $\mathrm{x}(\boldsymbol{t})$ is resulting from function x being evaluated at the vector $\boldsymbol{t}$. The discrete observable value is $\boldsymbol{y}$ vector. $\boldsymbol{W}$ is a symmetric positive definite $n \times n$ matrix, which is the inverse of the variance-covariance matrix of the residuals of the model $\boldsymbol{y}_i = \mathrm{x}_i(\boldsymbol{t}) + \boldsymbol{\varepsilon}_i, i = 1, \ldots, N$ such that N is indicating the number of curve replicates. Variance-covariance matrix can be estimated by the following:

$$\hat{\Sigma}_e = (N-1)^{-1} \boldsymbol{E}^T \boldsymbol{E}\,,$$

Where, $\boldsymbol{E}$ is a N by n residual matrix and i-th column of $\boldsymbol{E}$ is the $\boldsymbol{\varepsilon}_i$ vector. Since residuals are assumed to be I.I.D., therefore, covariances between errors are assumed to be zero, i.e. $\boldsymbol{W}$ is a diagonal with reciprocals of the error variance associated with the $\boldsymbol{y}_i$'s.

The parameter λ is called the smoothing parameter. As λ becomes larger and larger, the fitted curve x approaches the standard linear regression to the observed data, where $\mathrm{PE}N_2(x) = 0$. On the other hand, for small λ, the curve x approaches as an interpolation of the data, satisfying $\mathrm{x}(t_j) = y_j$ for all $j = 1, \ldots, n$. Following the finding of optimal lambda, the expression for the estimated coefficient vector is given by,

$$\hat{\boldsymbol{c}} = (\boldsymbol{\Psi}' \boldsymbol{W} \boldsymbol{\Psi} + \lambda \boldsymbol{R})^{-1} \boldsymbol{\Psi}' \boldsymbol{W} \boldsymbol{y}\,.$$

Here, $\boldsymbol{R}$ contains integrals of outer products of the second derivative of the basis functions vector, i.e. $\boldsymbol{R} = \int D^2 \boldsymbol{\psi}(s) D^2 \boldsymbol{\psi}'(s) ds$. Then the vector of fits to the data is

$$\hat{\boldsymbol{y}} = \boldsymbol{\Psi} \hat{\boldsymbol{c}} = \boldsymbol{\Psi} (\boldsymbol{\Psi}' \boldsymbol{W} \boldsymbol{\Psi} + \lambda \boldsymbol{R})^{-1} \boldsymbol{\Psi}' \boldsymbol{W} \boldsymbol{y} = \boldsymbol{S}_{\boldsymbol{\Psi},\lambda} \boldsymbol{y}\,.$$

Furthermore, $\boldsymbol{S}_{\boldsymbol{\Psi},\lambda}$ is an order n symmetric hat matrix that linear transforms the observable value to the smoothed curve, i.e. $\hat{\boldsymbol{y}} = \mathrm{S}_{\boldsymbol{\Psi},\lambda} \boldsymbol{y}$. The hat matrix is also known as a sub-projection operator, since it does not satisfy the idempotency relation of a projection matrix.

Our estimation is obtained by finding the function $x(t)$ that minimizes $\mathrm{PENSS}E_2(x)$ over the space of $\mathrm{x}(t)$ for which $\mathrm{PE}N_2(x)$ is defined. *De Boor, C. (2001)* and many other advanced texts on smoothing states that the function $\mathrm{x}(t)$ that minimizes $\mathrm{PENSS}E_2(x)$ is a cubic spline with knots at each data point t_j. In spline smoothing, an order four B-spline basis function with knots at sampling points is known as *cubic spline smoothing*. This solves the problem of where to place knots that minimises the penalised sum of squares. The process of selecting the optimal lambda will be discussed in the next subsection.

4.4.Generalised Cross-validation Method

A technique for choosing a smoothing parameter involves leave-one-out cross-validation. This procedure is repeated for each n observations in turn, and the resulting error sums of

squares are summed over all values. This criterion is computed over a range of values of λ, and chooses the value that yields the minimum of cross-validated SSE. However, as stated by *Ramsay, J. O., & Silverman, B. W. (2005),* it is usually computationally intensive and minimising Cross-Validation or CV can lead to under-smoothing the data. Hence, a simpler version of the CV procedure, which avoids the need to re-smooth n times, known as Generalised Cross-Validation or GCV, developed by *Craven, P. and Wahba, G. (1979),* is used for smoothing parameter selection.

$$\mathrm{GCV}(\lambda) = \frac{n * SSE}{\left(n - df(\lambda)\right)^2},$$

where, $\mathrm{df}(\lambda)$ is the degree of freedom of the smoothing parameter, defined as $\mathrm{df}(\lambda) = \mathrm{trace} S_{\boldsymbol{\Psi},\lambda}$, which is the number of basis functions that should be used for that similar level of smoothing. As stated in *Ramsay, J. O., Hooker, G., & Graves, S. (2009)*, GCV values often change slowly with $\log_{10}\lambda$ near the minimum value, so that a relatively wide range of λ values may give roughly the same GCV value. Therefore, it is not worth finding precisely the minimising value and simply plotting GCV over a mesh of $\log_{10}\lambda$ might be sufficient.

Higher values of the smoothing parameter (λ) lead to lower degrees of freedom, requiring fewer basis functions and smoothing out noise, known as underfitting. On the other hand, lower λ values return higher degrees of freedom, requiring more basis functions, leading the model to fit data points closely and allowing the model to capture more noise, i.e. random variation not explained by the true underlying model in the data. Real-world data contains noise, and a flexible model with many parameters or basis functions can fit these random fluctuations by mistakenly identifying them as meaningful patterns, known as overfitting. This can lead to bias in the statistical analysis of the data. Therefore, finding λ that doesn't underfit or overfit the model is reliable for further analysis.

4.5. Functional Box Plot

Functional box plot is a method to identify outliers by ordering them according to depth values. This method is similar to the classical boxplot, which is a graphical method for displaying the median, the first and third quartiles, and the non-outlying minimum and maximum observations. The functional boxplot is a natural extension of the classical box plot. *López-Pintado, S., & Romo, J. (2009)* introduced the notion of Band Depth (BD) and Modified Band Depth (MBD), which allows for ordering a sample of curves from the centre to outwards. BD on 2 sample curves of a function $x(t)$ is defined as follows:

$$BD^{(2)}(x) = \frac{\sum_{1 \le i_1 \le i_2 \le \cdots \le n} I\{G(y) \subseteq B(x_{i_1}, x_{i_2})\}}{\binom{n}{2}},$$

i.e. the frequency of containing the whole graph of the curve $x(t)$, whose coordinates defined as $G(x) = \{(t, x(t)) : t \in I\}$ and $B(x_{i_1}, x_{i_2})$ indicates the band that is delimited by 2 different random sample curves. Here, $I\{.\}$ is an indicator function and n is the number of sample curves in the observation. The bigger the value of BD, the more central position the curve has. Furthermore, the idea behind MBD is the same as BD but in the case of MBD we also take into consideration the proportion of times that a curve is in the band into account. Here,

Modified refers to the improvement of the indicator function, which returns 1 if the whole curve is delimited by sample curves, but rather than returning 0 otherwise, it will return the proportion of times that the curve was in between the delimited area. After depths are assigned to each curve, they are ordered from the highest to the lowest depth value. The curve which has the highest depth value is known as the median curve.

The 50% central region of a functional boxplot is defined by curves that have the maximum and minimum depth value of the first $\lceil n/2 \rceil$, the smallest integer not less than $n/2$, curves from the ordered curves. The fences, same as "whiskers" in a boxplot, are obtained by inflating the envelope, the maximum and minimum curve of the 50% central region, by 1.5 times the range of the central region. Any curves outside the fences are flagged as potential outliers. There are two types of outliers: magnitude outliers are distant from the median, and shape outliers have a different pattern from the other curves. Further details can be found in *Sun, Y., & Genton, M. G. (2010)*.

4.6. Functional Principal Components Analysis

Functional Principal Components Analysis (FPCA) offers a more informative approach to examine the covariance structure. The motivation behind FPCA is to find a set of orthonormal functions ξ over N observations, so that each curve in terms of these basis functions approximates the curve as closely as possible.

The first step in functional PCA is to find the weight function $\xi_1(s)$ by maximising,

$$N^{-1}\sum_{i=1}^{N}(f_{i1}^2) = N^{-1}\sum_i\left(\int \xi_1 x_i\right)^2,$$

with the constraint of unit sum of square constraint, i.e., $\int \xi_1(s)^2 ds = 1$. Also, the notations $\|\xi_1^2\| = 1$ is used for the squared norm of the function ξ_1. Here, f_i is known as principal components score and is defined as follows,

$$f_i = \int \xi\, x_i = \int \xi(s)\, x_i(s) ds.$$

We want to find orthonormal functions that maximise the square mean of principal component scores, and by subtracting the mean from functional values, i.e., cross-sectional means $N^{-1}\sum_i x_i(t)$ are zero, before doing functional PCA, allows us to solve the maximisation problem using their sample variances.

To find the following orthonormal functions that explain sample variances less than the previous orthonormal functions, we resort to the orthogonality constraint $\int \xi_k \xi_m = 0, k < m$ on subsequent steps.

The variance-covariance function is defined as follows:

$$v(s,t) = N^{-1}\sum_{i=1}^{N} x_i(s)x_i(t).$$

The weight functions should satisfy the equation,

$$\int v(s,t)\xi(t)dt = \rho\xi(s).$$

Suppose that each weight function has basis expansion,

$$\mathrm{x_i(t)} = \sum_{\mathrm{k=1}}^{\mathrm{p}} c_{ik}\psi_k(t).$$

We may write this more compactly as

$$\boldsymbol{x} = \boldsymbol{C\psi},$$

where the coefficient matrix $\boldsymbol{C}$ is $N \times p$. Furthermore, the vector-valued function $\boldsymbol{x}$ have components $x_1, \ldots, x_N$ and the vector-valued function $\boldsymbol{\psi}$ have components $\psi_1, \ldots, \psi_p$.

In matrix terms, the variance-covariance function is

$$v(s,t) = N^{-1}\boldsymbol{\psi}(s)'\boldsymbol{C}'\boldsymbol{C\psi}(t).$$

Let's define the order p symmetric matrix $\boldsymbol{M}$ to have entries

$$m_{p_1,p_2} = \int \psi_{p_1}\psi_{p_2} \; or \; \boldsymbol{M} = \int \boldsymbol{\psi\psi}'.$$

Suppose that an eigenfunction ξ has an expansion

$$\xi(s) = \sum_{k=1}^{p} b_k\psi_k(s)$$

or, in matrix notation $\xi(s) = \boldsymbol{\psi}(s)'\boldsymbol{b}$. These yields

$$\int v(s,t)\xi(t)dt = \int N^{-1}\boldsymbol{\psi}(s)'\boldsymbol{C}'\boldsymbol{C\psi}(t)\boldsymbol{\psi}(t)'\boldsymbol{b}dt = \boldsymbol{\psi}(s)'N^{-1}\boldsymbol{C}'\boldsymbol{CMb}.$$

Therefore, this can be expressed as,

$$\boldsymbol{\psi}(s)'N^{-1}\boldsymbol{C}'\boldsymbol{CMb} = \rho\boldsymbol{\psi}(s)'\boldsymbol{b}.$$

Since this equation must hold for all s, this implies the purely matrix equation

$$N^{-1}\boldsymbol{C}'\boldsymbol{CMb} = \rho\boldsymbol{b}.$$

But $\|\xi\| = 1$, implies that $\boldsymbol{b}'\boldsymbol{Mb} = 1$ and similarly two eigenfunctions will be orthogonal if and only if $\boldsymbol{b_1Mb_2} = 0$. To get the required principal components, we define $\boldsymbol{u} = \boldsymbol{M}^{1/2}\boldsymbol{b}$ and solve the equivalent symmetric eigenvalue problem

$$N^{-1}\boldsymbol{M}^{1/2}\boldsymbol{C}'\boldsymbol{C}\boldsymbol{M}^{1/2}\boldsymbol{u} = \rho\boldsymbol{u}$$

and compute $\boldsymbol{b} = \boldsymbol{M}^{-1/2}\boldsymbol{u}$ for each eigen vector. Further detailed information can be found in *Ramsay, J. O., & Silverman, B. W. (2005)*.

4.7. Discriminative Functional Mixture Model

The Discriminative Functional Mixture (DFM) model is a clustering method based on probability distributions for functional data, introduced by *Bouveyron, C., Come, E., & Jacques, J. (2014)*.

The aim is to cluster a set of observed curves $\{x_1, \dots, x_n\}$ that are independent realisations of a $L_2[0,T]$ continuous stochastic process $X = \{X(t)_{t\in[0,T]}\}$ into K homogenous groups. Let's assume there exists an unobserved random variable $Z = (Z_1, \dots, Z_K) \in \{0,1\}^K$ indicating the group membership of X: $Z_k = 1$ if X belongs to k-th group and 0 otherwise. The clustering task is therefore to predict the value $z_i = (z_{i1}, \dots, z_{iK})$ of Z for each observed curve x_i, for $i = 1, \dots, n$. Furthermore, let $F[0,T]$ be a latent subspace of $L_2[0,T]$ assumed to be the most discriminative subspace that easily separates the K groups of curves spanned by d basis functions in $L_2[0,T]$.

It is necessary first to reconstruct the functional form of discrete observations, as discussed in the Functional Data Analysis section. Suppose $\{\psi_j\}_{j=1,\dots,p}$ are the basis functions in the observed space, and $\{\phi_j\}_{j=1,\dots,d}$ are the basis functions in the discriminative subspace with $d < K < p$. The latter basis functions are obtained through a linear transformation $\phi_j = \sum_{l=1}^{p} u_{jl}\,\psi_l$ such that $p \times d$ matrix $U = (u_{jl})$ is orthogonal. Let, $\{\eta_1, \dots, \eta_n\}$ be the latent expansion coefficients of the observed curves in the discriminative subspace, assumed to be independent realisations of a latent random vector $\mathbf{H} \in \mathbb{R}^d$ and $\{\gamma_1, \dots, \gamma_n\}$ are independent realisations of a random vector $\mathbf{\Gamma} \in \mathbb{R}^p$ such that,

$$\mathrm{x_i(t)} = \sum_{\mathrm{j=1}}^{\mathrm{p}} \gamma_{ij}\psi_j(t) \approx \sum_{j=1}^{d} \eta_{ij}\phi_j(t).$$

The relationship between both bases suggests the following linear transformation:

$$\mathbf{\Gamma} = \boldsymbol{U}\mathbf{H} + \boldsymbol{\epsilon}\,,$$

where $\boldsymbol{\epsilon} \in \mathbb{R}^p$ is an independent and random noise term.

Furthermore, assume

$$\mathbf{\Lambda}_{|\mathrm{Z=k}} \sim N(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)\,,$$

where $\boldsymbol{\mu}_k$ and $\mathbf{\Sigma}_k$ are the mean vector and the covariance matrix of the k-th group respectively.

Secondly, the error term is also assumed to be distributed according to a multivariate Gaussian density:

$$\boldsymbol{\epsilon} \sim N(\mathbf{0}, \Xi)\,.$$

With these distributional assumptions, the marginal distribution of $\mathbf{\Gamma}$ is:

$$p(\gamma) = \sum_{k=1}^{K} \pi_k \phi\left(\gamma; \boldsymbol{U}_{\mu_k}, \boldsymbol{U}^t \boldsymbol{\Sigma}_k \boldsymbol{U} + \Xi\right),$$

where ϕ is the standard Gaussian density function, and the prior probability of the k-th group is $\pi_k = P(Z = k)$.

Finally, assume $\boldsymbol{\Xi}$ is such that:

$$\boldsymbol{\Delta}_k = cov(\boldsymbol{\Omega}^t \boldsymbol{\Gamma} \mid Z = k) = \boldsymbol{\Omega}^t (\boldsymbol{U}^t \boldsymbol{\Sigma}_k \boldsymbol{U} + \Xi) \boldsymbol{\Omega},$$

with $\boldsymbol{\Omega} = [\boldsymbol{U}, \boldsymbol{V}]$, $\boldsymbol{V}$ orthogonal complement of $\boldsymbol{U}$, has the following form:

$$\boldsymbol{\Delta}_k = \begin{pmatrix} \boldsymbol{\Sigma}_k & \mathbf{0} \\ \mathbf{0} & \beta_k \boldsymbol{I}_{p-d} \end{pmatrix}.$$

This model is referred to as $DFM_{[\boldsymbol{\Sigma}_k \beta_k]}$, which is a more general model in the sense that the variance of the actual data is modelled by $\boldsymbol{\Sigma}$ and the parameter β models the variance of the noise outside the functional subspace, that are different for each k-th cluster. Here, the variance of the actual data of the k-th group is modelled by $\boldsymbol{\Sigma}_k$ and the parameter β_k models the variance of the noise outside the functional subspace F. Following the strategy of *Bouveyron, C., & Brunet, C. (2012)*, 12 different Discriminative Functional Mixture (DFM) models can be generated by applying constraints on the parameters of the $\boldsymbol{\Delta}_k$ matrix. All the possible models are shown in Table 1 of *Bouveyron, C., Come, E., & Jacques, J. (2014)*.

Since the group memberships $\{z_1, \ldots, z_n\}$ for the curves are unknown and functional subspace F is assumed to be the most discriminative subspace, therefore $\boldsymbol{U}$ must be estimated separately. This can be done by an iterative 3-step algorithm, FunFEM, proposed by *Bouveyron, C., Come, E., & Jacques, J. (2014)*. In the first step, the aim is to find a matrix $\boldsymbol{U}$ such that the variances within the groups are minimum while the variances between the groups are maximum in the functional subspace F, hence known as Fisher's step. Next is the M-step, this step aims at maximising the log-likelihood conditionally on the matrix $\boldsymbol{U}^{(q)}$ obtained from the previous step. Last is the E-step, where the posterior probability is evaluated using all the estimates previously obtained. Furthermore, the Bayesian Information Criterion (BIC) can be used for model selection between the 12 proposed models.

5. Data Analysis

Unemployment data is fundamentally functional as it reflects continuous temporal processes characterised by smooth trends and structured variability (as in Figure 3), consistent with the basic principles of Functional Data Analysis.

5.1. Smoothing Parameter Selection

Smoothing functional data is the crucial beginning step in FDA so that we don't overfit our discrete data into a function. Here, the roughness penalty approach has been used for the smoothing method. I have used *de Boor, C. (2001)* theorem to minimize roughness penalty criterion by choosing *cubic spline smoothing* with 132 knots, which are more than enough to

fit n = 132 data points exactly if λ = 0; i.e. no smoothing is done, rather data are interpolated by the curve. Since the relation between the number of basis functions and knots for cubic spline smoothing is given by p equals n plus 2, in our case, which becomes 132 plus 2 equals 134 basis functions. GCV method is used to choose the smoothing parameter and the GCV values were computed at given $\log_{10}$, since a fairly wide range of λ may give roughly the same GCV value.

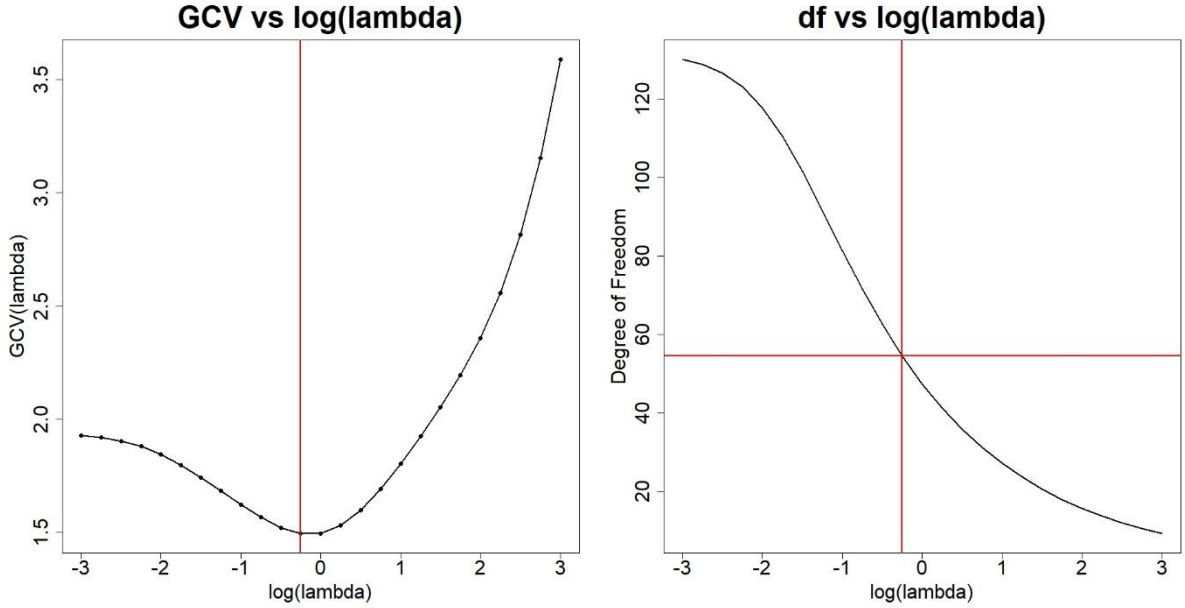

Figure 5: (Left) Optimal lambda selection by minimising Generalised Cross-Validation (red line – optimal point). (Right) Degree of freedom associated with each lambda (red lines point out the degree of freedom of the optimal lambda)

In Figure 5 each point in the left panel is 0.25 unit further away from the previous point on the x-axis. The minimum value of GCV attended at −0.25 indicated with a vertical red line, which is equal to 0.56. The right plot is the degree of freedom curve for $\log_{10}$ and for our chosen lambda, the degree of freedom for our curves is 55, i.e. we need 55 of B-spline basis functions for the same level of smoothing of our unemployment data. So, either we use the selected smoothing parameter on our 134 B-spline basis functions to smooth our functional data curves, or we can fit 55 B-spline basis functions to achieve the same level of smoothness. I have implemented the former approach on unemployment rates for further analysis.

5.2. Functional Data Analysis

Following the smoothing process, we can proceed with our subsequent analysis of unemployment data across 30 European nations. Figure 6 represents smoothed curves for all the observations from January 2013 to December 2023. Here, the blue line represents the overall mean value, which is 11%, at the start of the observational time frame, and the red line represents the smoothed mean curve for all the observations. It is evident that the overall mean clearly decreased over the year of observation, with a slight increase around 2020, which may be caused by the pandemic around that period. Afterwards, the mean value stays

stable for the rest of the time frame. Notably, the mean value decreases by approximately 50% over the observational period, reaching half of its initial magnitude at the study's endpoint.

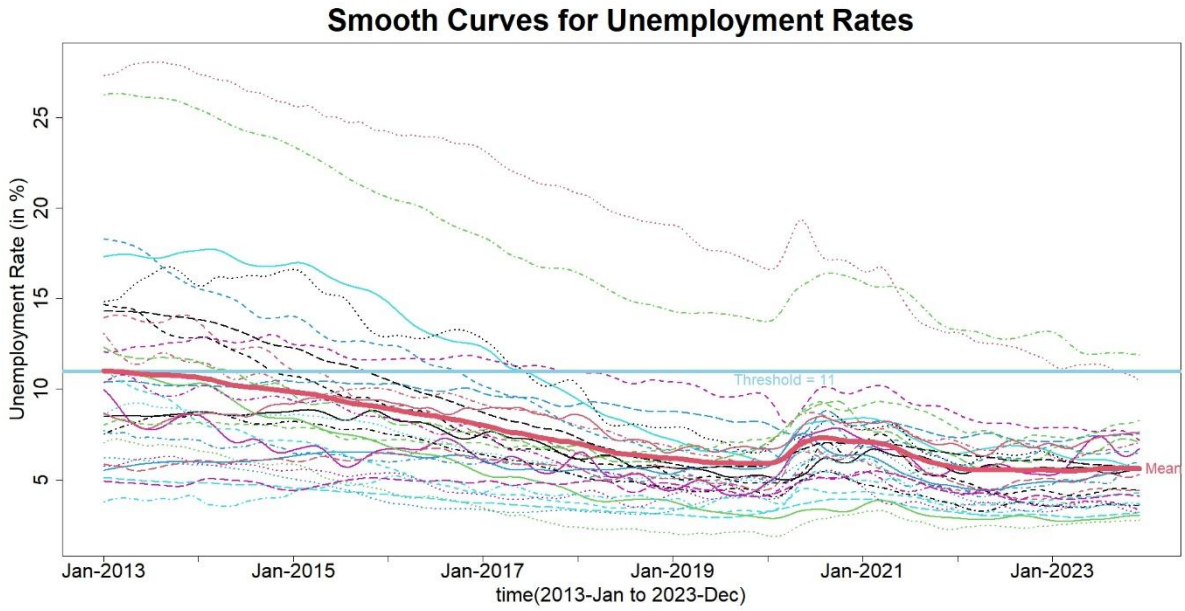

Figure 6: Smooth curves of unemployment rates with the threshold (blue line) being equal to the overall mean at the start of the observation period. The red curve indicates the overall mean unemployment rate of the curves.

5.3. Functional Boxplot

In Figure 6, some curves are further away from the mean curve, which can be identified as magnitude outliers. *Sun, Y., & Genton, M. G. (2010)* discussed the magnitude outliers in their literature and proposed a functional boxplot to identify these outliers.

Figure 7 shows a functional box plot of our data curves with the Modified Band Depth (MBD) criterion. Here, the purple coloured region is the 50% central region, analogous to the "inter-quartile range" (IQR) of a classical box plot. This region consists of curves whose coordinates are between the maximum and minimum coordinate values of the first half of the statistically ordered curves, depending on depth values. The black curve is similar to the median of the boxplot; here, it is the most central curve with the largest band depth value, known as the median curve. The blue lines are called fences, similar to whiskers of a boxplot, are obtained by extending the central region by 1.5 times the range of the 50% central region. Any curves outside the fences are flagged as potential outliers.

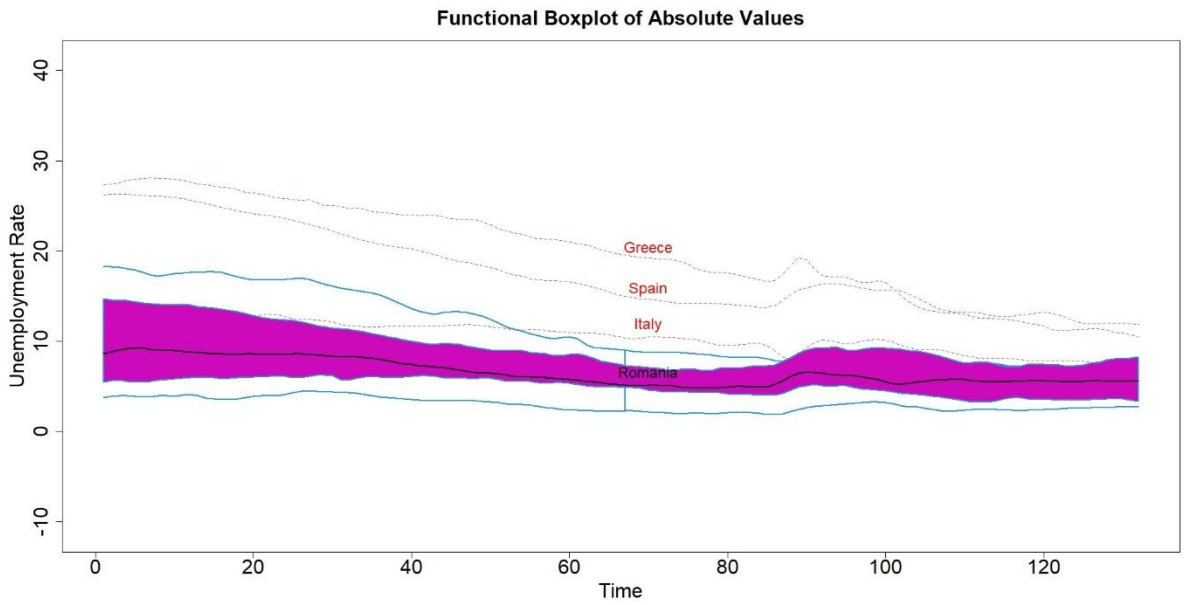

Figure 7: Functional box plot of unemployment curves with outliers and median functional curve.

Here, the functional curve of Romania has the highest band depth, serving as the functional median, which reflects the common pattern shared by the majority of the data. This curve appears quite stable, indicating that the typical pattern of unemployment in our dataset would be stable. While the central region at the start of the observational period is wide, it starts shrinking as time goes on. It is reasonable since in Figure 6 we see that at the start the unemployment rates were high for countries, but they eventually dropped. But there are magnitude outlier curves that show higher unemployment than the rest of the nations. As we know, Greece and Spain had higher unemployment, which is easily identifiable in Figure 6 and also in Figure 7. There is another curve that is identified as an outlier, which is the curve of Italy. It wasn't easy to identify it without a functional boxplot. While Italy has higher unemployment than the majority of countries, it is not as extreme as Greece and Spain. Unfortunately, unemployment in Italy didn't drop as similarly as the majority of countries in Europe. The curve stays outside the fence for the last half of the time, which implies less improvement in the employment of workers than in other countries.

5.4. Functional Principal Components Analysis

As we have explored central patterns and extreme curves in unemployment rates through the functional boxplot, we now turn to a complementary technique that provides a deeper understanding of the underlying modes of variation in the data. FPCA helps us to decompose the complex temporal unemployment trajectories into a set of principal components that capture the main sources of variation across countries. This approach not only reduces the dimensionality of the functional data but also facilitates the interpretation of dominant trends and differences among countries, providing insights beyond what can be visually understood from the boxplot alone.

The left plot in Figure 8 shows the cumulative variance that is explained by each FPC. This plot shows that 2 FPCs can explain more than 95% variance of the data; the black horizontal line in the plot shows 95% threshold.

The right plot in Figure 8 shows the first two functional principal component curves. The first FPC explains 93% of the total variance of unemployment rates data, which means that almost all the variability across countries' unemployment trajectories can be summarised by this single principal component. Furthermore, FPC1 shows a pattern where it assigns weights to curves with a slight downward trend over time. In practical terms, FPC1 captures whether a country's unemployment rate is consistently higher or lower than the mean unemployment trajectory. The FPC1 captures a general level shift that distinguishes countries based on their overall unemployment levels. That is, countries with higher unemployment rates with respect to the overall mean curve will be associated with positive scores, while those with lower rates will have negative scores and countries that are close to the mean curve will be close to the score 0 on the axis of the first FPC. Furthermore, the magnitude of the scores indicates how far above or below the country's unemployment curve is than the mean curve.

The second FPC explains only 5% of the total variance of the data, which is substantially less than FPC1. It shows a roughly linear increase up to around 2018 then roughly stabilise afterwards, and the sign of weights changes in mid-2016. This indicates that FPC2 captures a contrast between earlier and later periods in unemployment trajectories. FPC2 reflects differences in the temporal dynamics of unemployment rates among countries. A positive FPC2 score implies a country's unemployment curve tends to be lower than average in early years (2013 - 2016) and higher than average in the later years, and the contrary would be for negative FPC2 scores. That is, countries with a strong decline in unemployment will have negative FPC2 scores, countries where unemployment increased or declined less will have positive FPC2 scores and countries with little change over time will have FPC2 scores near zero.

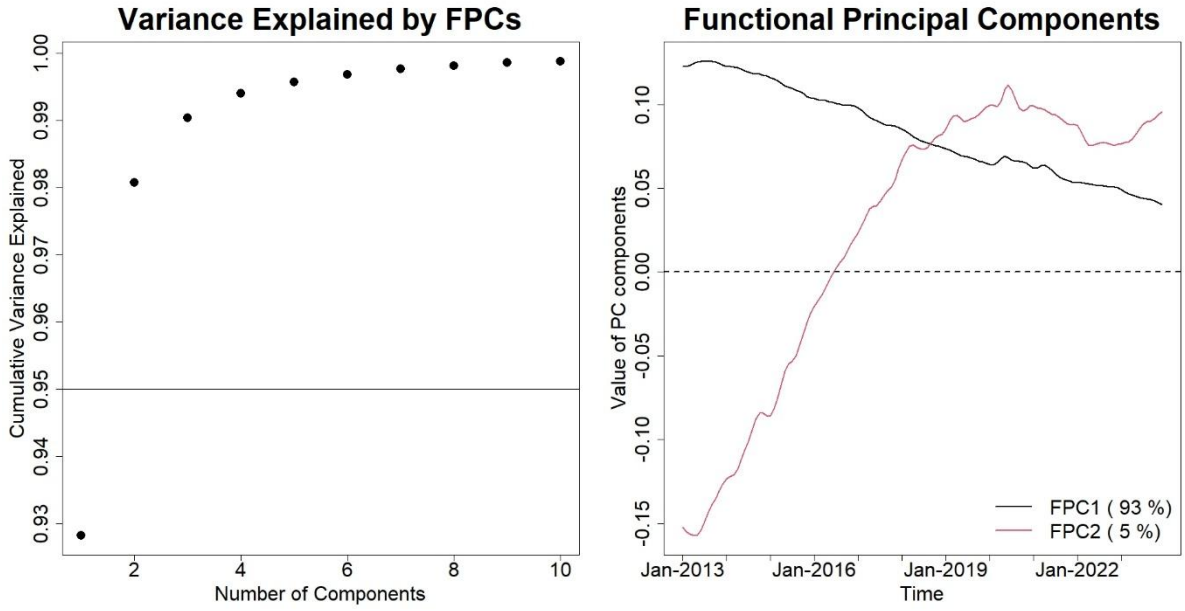

Figure 8: (Left) Cumulative variance explained by each FPC. (Right) First two FPC curves.

The plot of scores of each country in the first two FPCs, which explains the cumulative variance of 98% of the total variance, has been plotted in Figure 9. The widespread along the FPC1 axis shows that countries differ primarily in their overall unemployment levels, and a smaller spread along FPC2 indicates that most countries followed almost similar unemployment trajectories, with some exceptions, over time.

In the scores plot, Figure 9, Spain and Greece have higher scores in the FPC1 axis, indicating that they had persistently high unemployment rates throughout the entire period, well above the European average. This reflects the severe impact of the 2008 financial crisis on these two countries. Spain's crisis was primarily caused by a housing bubble burst (*European Central Bank. (2019)*), i.e. quick decrease in value, and Greece's crisis was triggered by high government debt and chronic fiscal mismanagement (*IMF. (2015)*). This led to a higher unemployment rate overall than the European average rate for over a decade.

Italy, Cyprus, Croatia, and Portugal show moderately elevated unemployment compared to the average. These countries are likely to face a slower economic recovery that prevents convergence to European norms. The moderate positive score indicates chronic but manageable unemployment challenges.

Sweden, Finland, Ireland, and many more show unemployment rates close to the European average. These countries experienced typical European unemployment dynamics, with no extreme deviations. They serve as a baseline for comparison – what "normal" European unemployment looks like during this period.

Germany, Norway, Czechia and some other countries show consistently lower unemployment than the average unemployment rates in Europe, according to the collected data. They reflect a successful economic transition and robust labour market performance.

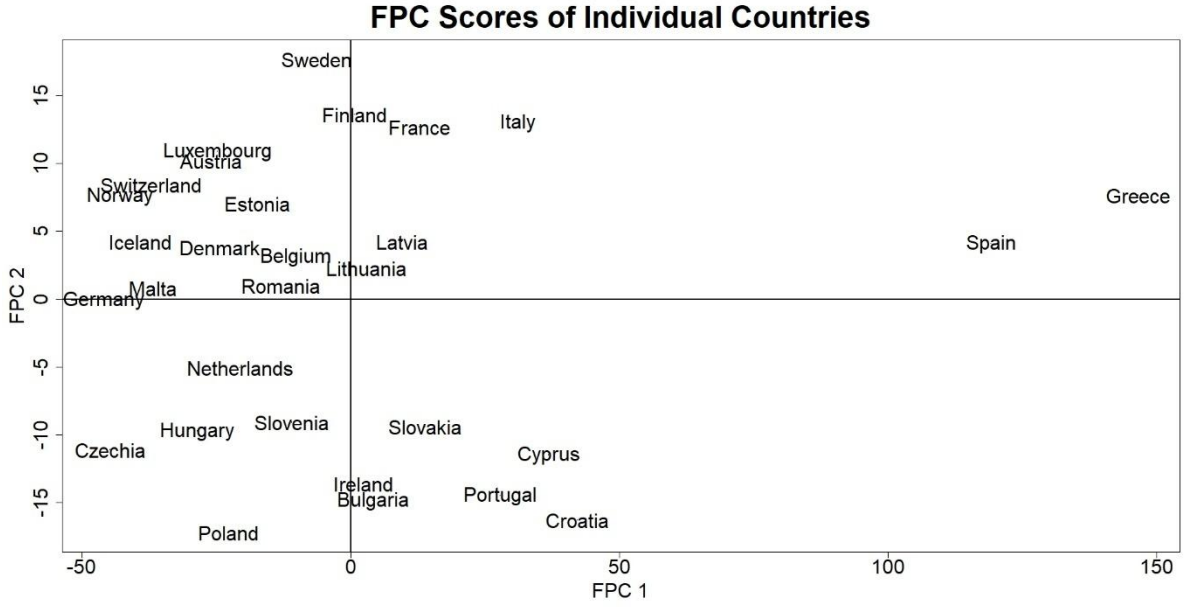

Figure 9: Scores of each country in the first two functional principal components.

The FPC2 scores capture early vs late period contrast of unemployment. Sweden, Finland, France and many more have a high positive FPC2 score, indicating a relatively low

unemployment rate in early years but higher unemployment in later years compared to the average. If we analyse the temporal patterns of these countries in Figure 3, it is easy to identify that they had relatively higher unemployment around 2020, which was around the global pandemic period. It is possible that FPC2 shows countries that had an effect of the pandemic having positive values and negative values if the pandemic didn't cause much unemployment in that period, relative to the average.

Poland, Croatia, Bulgaria, and others that have negative FPC2 scores have experienced stronger-than-average declines in unemployment over time. Germany, Malta, and Romania have FPC2 scores near zero, i.e. their unemployment trajectory closely follows the average temporal pattern.

5.5. Clustering

FPCA transformed complex functional data into a set of principal component scores, which facilitated comparisons between countries' unemployment rates in Europe. Now it would be a good time to identify countries that show similar patterns. The DFM model with a higher BIC index will be used for the clustering model. Each model can be generated by applying constraints on the parameter of the matrix Δ_k in 4.7. Each model has two capital letters, the first is either D or A to indicate if variance across clusters is general or diagonal, respectively, and the second letter is B for β_k. Furthermore, there are k, j, or no letters in between that indicate, respectively, the difference between clusters, the difference between variables and the same over variables and clusters. For example, the model "AkjBk" is where variance Σ_k is assumed to be diagonal which differ between clusters and variables, and β_k is different for each cluster.

The left plot of Figure 10 illustrates the BIC index of each DFM model with the number of clusters up to 10. "AjBk" with 8 clusters is the model with the highest BIC index. "AjBk" implies that Σ_k is diagonal and common in each group, but has different values for each variable, and β_k is different in each group. Therefore, depending on the BIC index "AjBk" with 8 clusters are a better option for DFM clustering on the unemployment rates.The following table shows the cluster number and the countries that belong to each cluster:

Cluster no.	Country name
1	Bulgaria, Ireland, Latvia, Lithuania, Slovakia
2	Greece, Spain
3	Denmark, Estonia, Luxembourg, Austria
4	Czechia, Germany, Malta, Iceland, Norway, Switzerland
5	Hungary, Netherlands, Poland, Slovenia
6	Belgium, Romania, Finland, Sweden
7	France, Italy
8	Croatia, Cyprus, Portugal

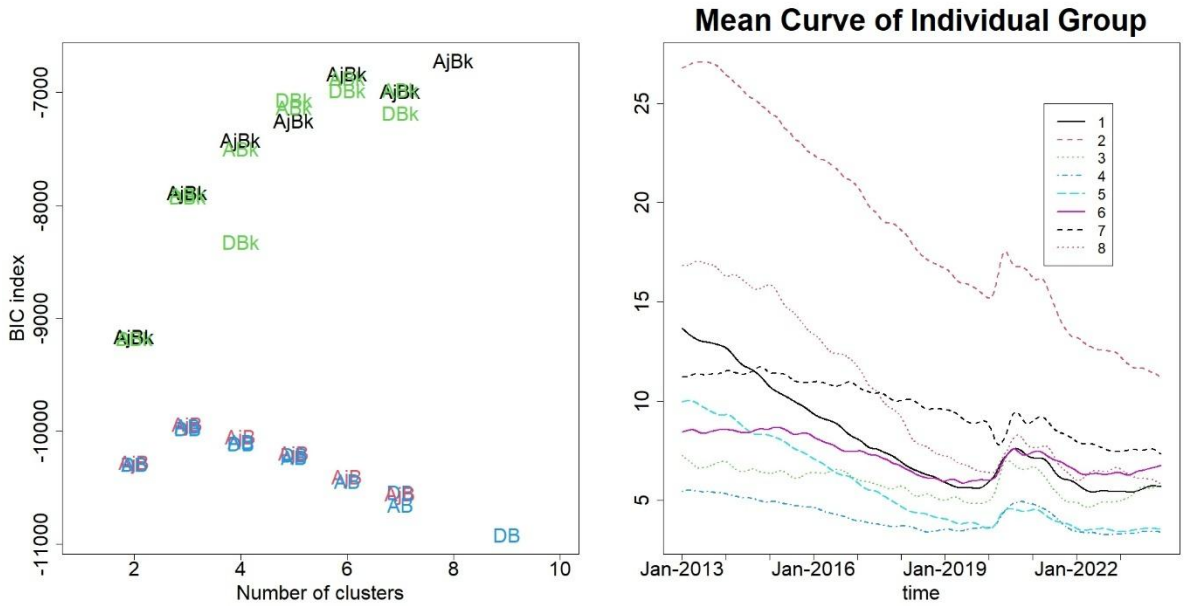

Figure 10:(Left) BIC index of the DFM models, (Right) Mean curve of individual clusters.

The right plot of Figure 10 shows the mean curve of the individual cluster done by the "AjBk" model. It is evident that the mean of each group has experienced a slight increase around 2020 and 2021, which possibly happened because of the world pandemic.

The mean curve of cluster 1 starts around 14% in 2013 and it shows a steady decline to around 6%, with an uprise around 2020. This represents a post-crisis economic recovery that experienced significant improvement. This cluster shows approximately a similar rate of decline in the curve with the 2nd and 5th clusters until 2020. The improvement of cluster 1 could be because of Ireland's recovery from its banking crisis (*IMF (2019)*), and EU integration and economic modernisation of other countries in the cluster.

Cluster 2's mean curve starts at the highest level around 27%, this cluster shows the steepest decline and remains elevated around 12% by the end of 2023. These countries were most severely impacted by the house bubble burst and European debt crisis (*European Central Bank. (2019), IMF. (2015)*), experiencing very high unemployment that gradually improved but remained above EU averages.

The mean of cluster 3 shows a stable curve in between $6\% - 8\%$. These countries maintained relatively strong institutions and economic fundamentals, showing consistent improvement throughout the period.

Cluster 4 shows the best performance, starting around 6% and declining to $3\% - 4\%$. These represent strong and robust labour markets. Germany's economic strength (*Marin, D. (2018)*), Switzerland's strong labour market (*OECD (2024)*) and Norway's robust social welfare (*OECD (2014)*) characterise this high-performing group.

Cluster 5 shows moderate starting levels around 10% and declining to $3\% - 4\%$. While having different starting points, the curve gets similar as cluster 4's mean curve in 2020, and both follow the same trend afterwards. The reduction in the gap between the two curves

reflects a process called unemployment convergence. This could reflect better implementation of policies, strong economic growth and labour mobility that reduce disparities in unemployment.

Cluster 6 demonstrates relatively stable performance around 8% up to mid-2015 and then shows a slow declining phase until the pandemic, and after the pandemic recovery, the curve seems to return to the same unemployment rate as the beginning. If we observe Figure 1, the Nordic welfare states (Finland, Sweden) show an unemployment rebound at the end of 2023, which could be the possible reason that made the mean curve rebound to 8%, while Romania and Belgium show a stable rate.

Cluster 7 shows a steady and slow decline in the post-economic crisis period and moderate improvement. There is little to no impact of the COVID-19 pandemic. The countries in this cluster follow a linear decline, as shown in Figure 1. These large European economies face structural labour market rigidities in both countries (*European Commission. (2019), OECD (2019)*) and demographic challenges in Italy (*IMF (2024)*), resulting in slower improvement compared to other clusters despite their economic size.

Cluster 8 shows a notable decline from 17% to 7% around the end of 2020, with a greater decline rate in some years. These tourism-dependent economies were possibly affected by the COVID-19 restrictions, which may explain the spike followed by recovery as travel resumed. This mean curve follows almost similar trend as cluster 2.

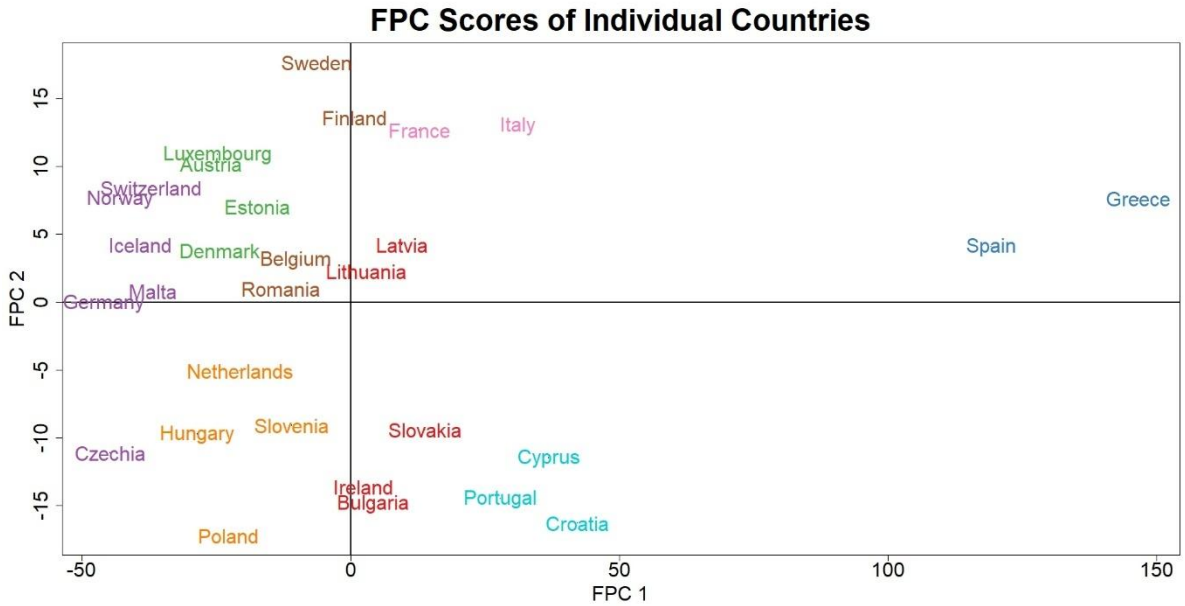

Figure 11: Clustering of countries (colour labelled) in the FPC score plot.

Figure 11 shows how the scores of countries are grouped according to the DFM modelling on the smoothed functional observations. Figure 11 and Figure 12 both have the same colour scheme for the visualisation of each group. Policy makers can use these clusters to design region-specific strategies. If a group of countries faces similar economic cycles or shocks, coordinated policies might be more effective. Countries can benchmark themselves against their peers within a cluster to set realistic targets or learn from successful ones. Furthermore,

international organisations or the EU can allocate resources more efficiently by targeting clusters rather than individual countries, ensuring that interventions address common needs.

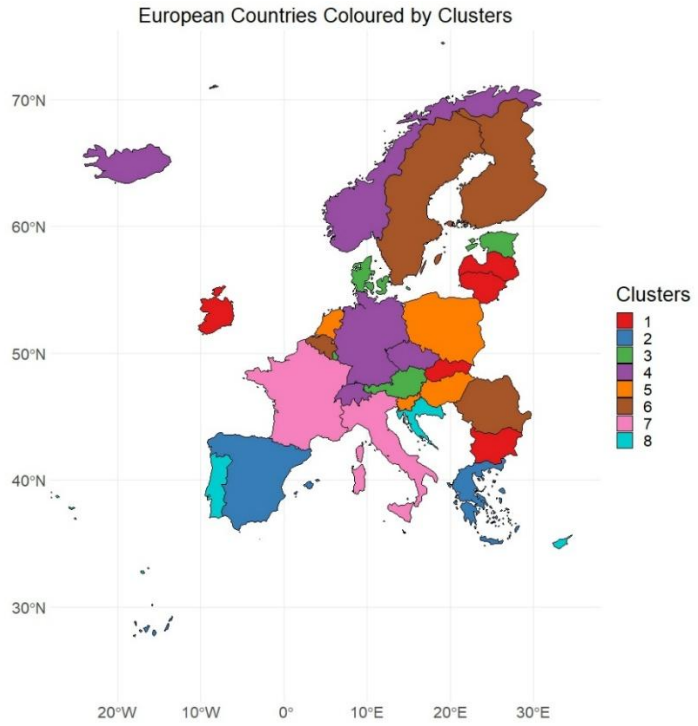

Figure 12: Clustering of European countries based on unemployment trends.

Figure 12 shows the choropleth map of different clusters. These clusters are geographically close to each other, showing geographic impact on unemployment, and some countries in the cluster are further away.

6. Conclusions

Most European countries showed a downward trend in unemployment rates over the decade, indicating a broad recovery from the economic impacts of the early 2010s and adaptation to the COVID-19 pandemic.

Greece and Spain constantly exhibit the highest unemployment rates (~20%), reflecting deep economic distress. However, both countries showed notable downward trends, with Greece achieving the largest reduction in unemployment among all 30 countries, signalling a substantial, though incomplete, recovery. In contrast, nations like Germany, Switzerland, and the Czech Republic maintained consistently low rates (~5%), reflecting stable labour markets. Southern European nations (e.g., Croatia, Portugal, Cyprus) had high initial unemployment but converged toward the overall median (6.4%) by 2022.

The first principal component accounted for most of the variation and represented the overall unemployment level. The second captured differences in early vs. late period performance, highlighting pandemic-related shifts in countries.

Eight distinct clusters of countries were identified based on the shape of their unemployment trajectories. These clusters shared similar dynamics on unemployment and can inform group-level labour policy responses.

By focusing on the time-based evolution of unemployment, this study offers a clearer understanding of how European labour markets diverged and converged over the past decade. The results could be valuable for economists and policymakers who seek to design regionally tailored, evidence-based interventions. Clustering similar countries can help coordinate employment strategies and target support where it's needed most.

Future work could extend this framework by incorporating additional indicators such as GDP growth, education levels, or labour mobility to explore how various factors interact with unemployment trends in a multivariate functional setting.

7. References

Bouveyron, C., & Brunet, C. (2012). Simultaneous model-based clustering and visualization in the Fisher discriminative subspace. Statistics and Computing, 22(1), 301–324.

Bouveyron, C., Come, E., & Jacques, J. (2014). The discriminative functional mixture model for the analysis of bike sharing systems. HAL Archives-Ouvertes.

Craven, P., & Wahba, G. (1979). Smoothing noisy data with spline functions: Estimating the correct degree of smoothing by the method of generalized cross-validation. Numerische Mathematik, 31, 377–403.

de Boor, C. (2001). A practical guide to splines (Revised ed.). New York, NY: Springer.

European Central Bank. (2019). The financial transmission of housing bubbles: Evidence from Spain (ECB Working Paper No. 2245).

European Commission. (2019). Productivity and competitiveness: Where does France stand in the euro zone?

Eurostat. (2025). Unemployment by sex and age - monthly data. https://doi.org/10.2908/UNE_RT_M

International Monetary Fund (IMF). (2015). Greece: Staff report for the 2015 Article IV consultation.

International Monetary Fund (IMF). (2019). Ireland: Recovery from financial crisis. IMF lending case study. https://www.imf.org/en/Countries/IRL/ireland-lending-case-study. Accessed 23 July 2025.

International Monetary Fund (IMF). (2024). Italy: IMF country report No. 24/241.

López-Pintado, S., & Romo, J. (2009). On the concept of depth for functional data. Journal of the American Statistical Association, 104(486), 718–734.

Marin, D. (2018). Explaining Germany's exceptional recovery. CEPR Press.

OECD. (2014). OECD economic surveys: Norway 2014. OECD Publishing. https://dx.doi.org/10.1787/eco_surveys-nor-2014-en

OECD. (2019). Strengthening active labour market policies in Italy: Connecting people with jobs. OECD Publishing. https://doi.org/10.1787/160a3c28-en

OECD. (2024). OECD economic surveys: Switzerland 2024. OECD Publishing. https://doi.org/10.1787/070d119b-en

Ramsay, J. O., & Silverman, B. W. (2005). Functional data analysis (2nd ed.). Springer.

Ramsay, J. O., Hooker, G., & Graves, S. (2009). Functional data analysis with R and MATLAB. Springer.

Sun, Y., & Genton, M. (2010). Functional boxplots. Journal of Computational and Graphical Statistics, 20(2), 316–334.

8. Appendix

```
library(tidyverse)
library(ggplot2)
library(zoo)
library(fda)
library(funFEM)
library(giscoR)
library(countrycode)
library(sf)

# read data (collected from eurostat)
unemploy_rate <- read.csv("unemployment_in_europe.csv")
head(unemploy_rate)
nrow(unemploy_rate)
# 30 countries under observation
ncol(unemploy_rate)
# 133 - 1(first column) = 132 data points for individual
observations

# missing data check
anyNA(unemploy_rate) # no data are missing

# changing time format
# removes x in front of dates
time_clean <- sub("^X", "", colnames(unemploy_rate[1,-1]))
# transform dates into month-year format
ym_dates <- as.yearmon(time_clean, "%Y.%m")
# transform dates into year-month-date format
formatted_dates <- format(ym_dates, "%Y-%m-%d")
# changes column names to formatted date
cols <- c(colnames(unemploy_rate[1]), formatted_dates)
colnames(unemploy_rate) <- cols
head(unemploy_rate)
```

```
### data visualisation

# convert wide data to long data
long_data <- unemploy_rate %>%
  pivot_longer(cols = -Reference_area,
               names_to = "Time_Period",
               values_to = "Unemployment")
# character dates to date type
long_data$Time_Period <- as.Date(long_data$Time_Period)

# Figure 1
# highest and lowest average unemployment value over the year 2013
to 2023

# average rates over 11 years observation period
avg_rate <- aggregate(Unemployment ~ Reference_area, long_data,
mean)
# standard deviation of unemployment rates over 11 years observation
period
sd_error <- aggregate(Unemployment ~ Reference_area, long_data, sd)
# create a data frame with reference name, mean value and standard
error columns
data_avg_sd <- data.frame(Reference_area = avg_rate$Reference_area,
                          Average = avg_rate$Unemployment,
                          Sderror = sd_error$Unemployment)
# plot point graph with addition and subtraction of 1 standard error
wicks
windows()
ggplot(data_avg_sd, aes(x = reorder(Reference_area, Average),
                        y = Average)) +
  geom_point(size = 5, color = "dodgerblue") +
  geom_errorbar(aes(ymin = Average-Sderror, ymax = Average+Sderror),
                width = 0.2, size = 1.2, color = "red")+
  labs(title = "Mean and Standard Error of Unemployment Rates by
Country",
       x = "Country", y = "Unemployment rate") +
  theme_bw()+
  theme(axis.text.x = element_text(angle = 45, hjust = 1, size =
25),
        legend.position = "none",
        axis.title = element_text(size = 20),
        axis.text = element_text(size = 14),
        axis.text.y = element_text(size = 25),
        plot.title = element_text(size = 26))

# higher average unemployment, a potential indicator of economic
challenges in
# those regions.
# lower average unemployment, possibly reflecting stronger or more
stable labor markets.
```

```
#figure 2
#heatmap on rates over each year
# extracting years
year_data <- long_data %>%
  mutate(Year = format(Time_Period, "%Y"))

# mean rates over each years
mean_year_data <- aggregate(Unemployment ~ Reference_area + Year,
year_data, mean) %>%
  arrange(Reference_area)

# plotting heat map with each square representing each year
windows()
ggplot(mean_year_data, aes(x = reorder(Reference_area,
Unemployment),
                           y = Year, fill = Unemployment)) +
  geom_tile() +
  scale_fill_viridis_c(option = "cividis")+
  labs(title = "Mean Unemployment Rate Heatmap for Each Year",
       x = "Country", y = "Year", fill = "Unemployment Rate (%)") +
  theme_minimal()+
  theme(
    axis.title = element_text(size = 20),
    axis.text = element_text(size = 20),
    axis.text.x = element_text(size = 23, angle = 45, hjust =1),
    plot.title = element_text(size = 26),
    text = element_text(size = 25))


# Figure 3
# identify values that are significantly greater than others
overall_median <- median(long_data$Unemployment)
Q1 <- quantile(long_data$Unemployment, 0.25)
Q3 <- quantile(long_data$Unemployment, 0.75)
IQR <- Q3 - Q1
upper_fence <- Q3 + 1.5 * IQR
long_data$sig_high <- long_data$Unemployment > upper_fence

# unemployment rate over time line plot for each country
windows()
ggplot(long_data, aes(x = Time_Period, y = Unemployment,
                      color = sig_high, group = Reference_area)) +
  labs(title = paste0("Unemployment Rates Over Time by Country
(overall median = ",
                      overall_median, "%)"),
       x = "Time Period", y = "Unemployment Rates") +
  geom_smooth(method = lm, se = F, color = "gray20")+
  theme_minimal() +
  geom_point(aes(color = sig_high), alpha = 0.5) +
```

```
  scale_colour_manual(values = c("FALSE" = "dodgerblue4", "TRUE" =
"red")) +
  theme(
    axis.title = element_text(size = 20),
    axis.text = element_text(size = 14),
    axis.text.x = element_text(size = 14, angle = 45),
    plot.title = element_text(size = 26),
    text = element_text(size = 25))+
  facet_wrap(~Reference_area, scales = "free_y") +
  theme(legend.position = "none")


#figure 4
#rough unemployment rate over at starting and ending of the time
frame
unemploy_trends <- long_data %>%
  group_by(Reference_area) %>%
  summarize(
    first_rate = first(Unemployment),
    last_rate = last(Unemployment),
    rate_change = last(Unemployment) - first(Unemployment)
  )
# Sort by rate change
unemploy_trends <- unemploy_trends %>%
  arrange(rate_change)

#improving and worsening countries
windows()
ggplot(unemploy_trends, aes(x = reorder(Reference_area, rate_change,
decreasing = T),
                            y = rate_change, fill = rate_change > 0)) +
  geom_bar(stat = "identity", color = "black", alpha = 0.5) +
  scale_fill_manual(values = c("TRUE" = "red", "FALSE" = "green"),
                    labels = c("Improved", "Worsened")) +
  coord_flip() +
  labs(
    title = "Improving and Worsening Countries in Unemployment
Rates",
    x = "Country",
    y = "Rate Change"
  ) +
  theme_minimal() +
  theme(legend.position = "top", legend.title = element_blank())+
  theme(
    axis.title = element_text(size = 20),
    axis.text = element_text(size = 20),
    plot.title = element_text(size = 26),
    text = element_text(size = 25))
```

```
# Functional Data Analysis
ncol(unemploy_rate)
basis <- create.bspline.basis(1:132, nbasis = 134)
data_matrix <- unemploy_rate[,-1]

# function to select lambda value using GCV method
gcv_lambda_select <- function(data_matrix){
    #GCV to choose lambda for absolute data
    loglam <- seq(-3, 3, 0.25)
    nlam <- length(loglam)
    dfsave <- rep(NA, nlam)
    gcvsave <- rep(NA, nlam)

    # create linear differential operator of Lfd class
    lfd_unemploy <- int2Lfd(2) # second derivative as penalty term

    #function to check gcv for all provided log lambda
    for(ilam in 1:nlam){
      cat(paste("log10 Lambda =", loglam[ilam], "\n"))
      lambda = 10^loglam[ilam]
      fdParobj = fdPar(basis, lfd_unemploy, lambda)
      smoothlist = smooth.basis(1:132, data_matrix,
                                fdParobj)
      dfsave[ilam] = smoothlist$df
      gcvsave[ilam] = sum(smoothlist$gcv)
    }

    # store lambda which has minimum gcv value
    log_lambda <- loglam[which.min(gcvsave)]
    lambda <- 10^log_lambda
    df_lambda <- dfsave[which.min(gcvsave)]

    return(list(
      lambda = lambda,
      df_lambda = df_lambda,
      gcvsave = gcvsave,
      loglam = loglam,
      dfsave = dfsave,
      log_lambda = log_lambda,
      df_lambda = df_lambda
    ))
}

# plot of gcv and df curve over selected lambdas
gcv_lambda <- gcv_lambda_select(t(data_matrix))
windows()
par(mar = c(5,5,4,2),mfrow=c(1,2))
plot(gcv_lambda$loglam, gcv_lambda$gcvsave, type = "l", lwd = 2,
     ylab = "GCV(lambda)", xlab = "log(lambda)",
     main = "GCV vs log(lambda)",
```

```
     cex.axis = 2, cex.lab = 2,cex.main = 3)
points(gcv_lambda$loglam, gcv_lambda$gcvsave, cex = 1, pch = 16)
abline(v = gcv_lambda$log_lambda, col = "red", lwd = 2)
plot(gcv_lambda$loglam, gcv_lambda$dfsave, type = "l", lwd = 2,
     ylab = "Degree of Freedom", xlab = "log(lambda)",
     main = "df vs log(lambda)",
     cex.axis = 2, cex.lab = 2,cex.main = 3)
abline(v =gcv_lambda$log_lambda, h = gcv_lambda$df_lambda, col =
"red", lwd = 2)
par(mfrow = c(1,1))

# start and end date for the given data
start_date <- as.Date("2013-01-01")
end_date <- as.Date("2023-12-01")
date_seq <- seq(start_date, end_date, by = "month")
form_date <- format(date_seq, "%b-%Y")

# smooth splines with the selected lambda (all curves)
windows()
par(mar = c(5,5,4,2))
lfd_unemploy <- int2Lfd(2)
fdpar_unemploy <- fdPar(basis, lfd_unemploy, gcv_lambda$lambda)
fd_unemploy <- smooth.basis(1:132, t(data_matrix),
fdpar_unemploy)$fd
plot(fd_unemploy, ylab = "Unemployment Rate (in %)",
     xlab = "time(2013-Jan to 2023-Dec)",
     xaxt = "n", lwd = 2,
     main = "Smooth Curves for Unemployment Rates",
     cex.axis = 2, cex.lab = 2, cex.main = 3)
abline(h = 11, lwd = 5, col = "skyblue")
text(80, 10.5, "Threshold = 11", col = "skyblue", cex = 1.5, pos =
4)
m_unemploy <- mean.fd(fd_unemploy)
lines(m_unemploy, col = 2, lwd = 8)
text(132, eval.fd(132, m_unemploy), "Mean", col = 2, cex = 1.5, pos
= 4)
axis(1, cex.axis= 2, at = seq(1,132, by = 12), labels =
form_date[c(1, seq(13,132, by = 12))])

# magnitude outliers detection
windows()
par(mar = c(5,5,4,2))
smooth_data_matrix <- eval.fd(1:132, fd_unemploy)
boxplot_fdata <- fbplot(smooth_data_matrix, 1:132, method = "MBD",
xlab = "Time",
                        ylab = "Unemployment Rate",
                        main = "Functional Boxplot of Absolute
Values",
                        cex.axis = 2, cex.lab = 2, cex.main = 2)
#this plot shows curves that are further away from central region
outliers <- boxplot_fdata$outpoint
```

```
for(i in outliers){
  text(x = 70, y = data_matrix[i, 70],
       labels = unemploy_rate[i, 1], pos = 3, col = "red", cex =
1.5)
}
med_curve <- boxplot_fdata$medcurve
text(x = 70, y = data_matrix[med_curve, 70],
     labels = unemploy_rate[med_curve, 1], pos = 3, col = "black",
cex = 1.5)


# Functional PCA
fpca <- pca.fd(fd_unemploy, nharm = 4, centerfns = TRUE)


# plot cumulative variance explained
windows()
par(mar = c(5,5,4,2), mfrow = c(1,2))
# plot the explained cumulative percentage of total variations
plot(cumsum(fpca$values[1:10])/sum(fpca$values),
     xlab = "Number of Components",
     ylab = "Cumulative Variance Explained",
     main = "Variance Explained by FPCs",
     cex.axis = 2, cex.lab = 2,
     cex.main = 3, pch = 16, cex = 2)
abline(h = 0.95)
# two pcs are enough to explain more than 95% variances

# plot FPC curves
harm <- fpca$harmonics
harmvals <- eval.fd(1:132, harm)
matplot(1:132, harmvals[,1:2], type = "l", col = 1:3, lty = 1, lwd =
2.5,
        ylab = "Value of PC components", xlab = "Time", xaxt = "n",
        main = "Functional Principal Components",
        cex.axis = 2, cex.lab = 2, cex.main = 3)
axis(1, cex.axis= 2, at = seq(1,132, by = 12), labels =
form_date[c(1, seq(13,132, by = 12))])
abline(h=0, lty = 2, col = 1, lwd = 2)
legend("bottomright", c(paste("FPC1 (",round(fpca$varprop[1],2)*100,
"%)"),
                        paste("FPC2 (",round(fpca$varprop[2],2)*100,
"%)")),
       col = 1:2, lty = 1, cex = 2, bty = "n", lwd = 2)
par(mfrow=c(1,1))
# fpc 1 explains countries that had higher unemployment rate but
decreased over time
# fpc 2 positive value explains rates were high between 2016 and
2020 respect to other years in observation
# and negative value explains higher unemployment rates at  2013 to
2016
```

```
#plot fpc scores
windows()
par(mar = c(5,5,4,2))
plot(fpca$scores[,1:2], cex = 0.01, xlab = "FPC 1",
     ylab = "FPC 2", cex.axis = 2, cex.lab = 2,
     cex.main = 3, main = "FPC Scores of Individual Countries")
abline(h = 0, v = 0, lwd = 2)
text(fpca$scores[,1], fpca$scores[,2],
     labels = unemploy_rate$Reference_area, cex = 2)


# clustering using DFM model
set.seed(420)
femmodels <- c("DkBk", "DkB", "DBk",
               "DB", "AkjBk", "AkjB", "AkB", "AkBk", "AjBk", "AjB",
"ABk",
               "AB")
nmodels <- length(femmodels)
femresults <- list()
bestk <- bestbic <- numeric(0)
K=2:10
fembic <- matrix(NA,nrow=nmodels,ncol=max(K))
for (i in 1:nmodels){
  try({print(femmodels[i])
      femresults[[i]] <- funFEM(fd_unemploy,model=femmodels[i],K=K)
      fembic[i,K] <- femresults[[i]]$allCriterions$bic
      bestk[i] <- which(fembic[i,]==max(fembic[i,K],na.rm=TRUE))
      bestbic[i] <- max(fembic[i,K],na.rm=TRUE)})
}
besti <- which(bestbic==max(bestbic,na.rm=TRUE))
# DFM model selected using BIC index
femmodels[besti] #AjBk
# number of clusters that returns higher BIC index for the chosen
DFM model
bestk[besti] #8
femresult <- femresults[[besti]]

# print countries in clusters
for(i in 1:femresult$K){
  print(i)
  print(unemploy_rate[femresult$cls==i, 1])
}

# BIC plot for all models and K
windows()
par(mar = c(5,5,4,2), mfrow = c(1,2))
i <- 1
plot(1:max(K), fembic[i,], col = i, pch = i,
```

```
     ylim = c(min(fembic, na.rm = T), max(fembic, na.rm = T)), type
= "n",
     xlab = "Number of clusters", ylab = "BIC index",
     cex.axis = 2, cex.lab = 2, cex.main = 3)
for(i in 1:nmodels){
  text(1:max(K), fembic[i,], femmodels[i], col = i, cex = 2)
}

#Curve plot for each individual clusters
clmeans <- fd_unemploy
clmeans$coefs <- t(femresult$prms$my)
plot(clmeans, lwd = 2, xaxt="n", ylab = "",
     cex.axis = 2, cex.lab = 2, cex.main = 3,
     main = "Mean Curve of Individual Group")
axis(1, cex.axis= 2, at = seq(1,132, by = 12),
     labels = form_date[c(1, seq(13,132, by = 12))])
legend(100,25,legend=1:8,col=c(1:6,1:2),lty=c(1:5,1:3),
       lwd = 2, cex = 1.5)


# Colour scheme for the clusters
cluster_colors <- c("#E41A1C", "#377EB8", "#4DAF4A", "#984EA3",
                    "#FF7F00", "#A65628", "#F781BF", "#00CED1")

# cluster on functional principal components
windows()
par(mar = c(5,5,4,2))
plot(fpca$scores[,1:2], cex = 0.01, xlab = "FPC 1",
     ylab = "FPC 2", cex.axis = 2, cex.lab = 2,
     cex.main = 3, main = "FPC Scores of Individual Countries")
abline(h = 0, v = 0, lwd = 2)
text(fpca$scores[,1], fpca$scores[,2],
     labels = unemploy_rate$Reference_area, cex = 2,
     col = cluster_colors[femresult$cls])


# Mapping of clusters

# getting map polygon values
europe <- gisco_get_nuts(year = 2024, nuts_level = 0)
# extract country code that are relevant to analysis
eu_countries <- countrycode(unemploy_rate[,1], origin =
"country.name",
                            destination = 'iso2c',
                            custom_match = c("Greece" = "EL"))

# Data frame with Country code and the cluster that individual
country belongs to
cluster_data <- data.frame(CNTR_CODE = eu_countries,
                           cluster = femresult$cls)
```

```
# Join cluster info to spatial data
europe_clusters <- europe %>%
  filter(CNTR_CODE %in% cluster_data$CNTR_CODE) %>%
  left_join(cluster_data, by = "CNTR_CODE")

# Plot map colored by cluster
windows()
ggplot(europe_clusters) +
  geom_sf(aes(fill = factor(cluster)), color = "black") +
  scale_fill_manual(values = cluster_colors, name = "Clusters") +
  coord_sf(xlim = c(-25,35), ylim = c(25, 73), expand = T) +
  theme_minimal() +
  labs(title = "European Countries Coloured by Clusters") +
  theme(plot.title = element_text(size = 20, hjust = 0.5),
        text = element_text(size = 20))
```