Estimating Belgian sector-regional value added

A manual for ensemble predictions for the HERMREG model

Scientific team
Prof. Dr. Glenn Magerman (ULB, CEPR, CESIfo) - project leader
Prof. Dr. Jozef Konings (KUL, VIVES, CEPR)
Drs. Niccolò Consonni (ULB)
Drs. Federico Gallina (ULB)
Drs. Alberto Palazzolo (ULB)

Project manager
Dr. Palina Shauchuk (ULB)

February 2025
Solvay Public Policy House
Avenue Franklin Roosevelt 50, 1050 Brussels
contact: palina.shauchuk@ulb.be

This report was produced by a team under the responsibility of the Université Libre de Bruxelles (ULB) on behalf of the Federal Planning Bureau and its three regional partners: Statistiek Vlaanderen (VSA), the Institut wallon de l'évaluation, de la prospective et de la statistique (IWEPS), and the Brussels Instituut voor Statistiek en Analyse (BISA). The results, analyses, and conclusions presented in this document reflect the work and interpretation of the authors, who bear scientific responsibility for them. The commissioning parties retain full discretion regarding the use of the information contained in this report.

Executive summary

This report presents a comprehensive methodology for estimating sector-regional gross value added in Belgium, for the most recent year that is not yet available in the data. The framework integrates diverse datasets, multiple econometric and machine learning models, and an ensemble approach to ensure robust and accurate predictions consistent with national-level projections from the HERMES model. The methodology and main results are described in the current report. We also provide a full data and code toolbox in Python to recreate the results in this report, and to generate future updates of the HERMREG predictions.

Objectives and Context The project is part of the HERMREG model, which supports regional economic estimates for indicators such as gross value added, employment, and household income. The aim is to replace theoretical assumptions with a validated econometric framework for sector-regional gross value added estimation. The results contribute to improving regional economic outlooks while maintaining coherence with national projections.

Research Design The analysis uses detailed datasets, including regional accounts, VAT statistics, labor market indicators, and input-output linkages. These datasets serve as inputs to estimate a range of models:

1. Univariate time series models (ARIMA) capture temporal dynamics at the sector-region level.
2. Multivariate time series models (VAR/VEC) incorporate interdependencies across regions.
3. Panel fixed effects models account for observed and unobserved heterogeneity.
4. Spatial autocorrelation models leverage input-output relationships to capture sectoral spillovers.
5. Random forests exploit complex relationships with minimal assumptions.

Each model is estimated on a series of variable transforms (levels, logs, square root, inverse, and standardized), and then fitted and validated using standard performance metrics (Normalized Root Mean Squared Error), ensuring reliability across various data dimensions. We then aggregate all models' predictions into an ensemble model, combining the complementary strengths of all models. We provide two versions of the ensemble: a weighted average version, with weights based on out-of-sample validation performance, and a single best predictor model, based on the lowest out-of-sample performance. Finally, predictions are adjusted to align with HERMES national-level projections, ensuring consistency between regional and national estimates for each sector.

Toolbox evaluation Each model contributes to the final predictions by leveraging different dimensions of variation in the data, as well as model assumptions. The ARIMA models and the random forests turn out to be the best predictor for most sector-regions, while all five variable transforms contribute to the best predictors. These results underline the usefulness of the ensemble method to generate plausible predictions: while some models work relatively well for some sector-regions, other models perform better for others. We provide results for both the weighted average ensemble, as well as the single best predictor for each sector-region. It turns out that the single best predictor provides the best predictions on the current data. The average correction required to ensure consistency with the HERMES projections is very small at 0.8%. This implies that the predictions at the national level from

HERMES, and those obtained at the regional level from this toolbox, while using different methods and data, are close to each other on average. While the best-model approach currently outperforms the mean ensemble, its performance can improve as new data becomes available. We allow for this flexibility in the construction of the toolbox.

Predictions for gross value added Total gross value added for Belgium in 2023 is expected to be 525,599 million EUR in terms of current prices and 424,381 million EUR in chained prices. At the regional level, Flanders is expected to contribute 60% to Belgian GDP (313,833 million EUR), followed by Wallonia with 23% (120,359 million EUR) and Brussels with 17% (91,407 million EUR). These proportions remain similar in chained prices. These numbers are, by construction, the same as the predictions from the HERMES model.

The following numbers are predictions for sector-regions based on the current toolbox. In Brussels, sector KK (*Financial and insurance activities*) is the largest sector, with expected value added of 17,197 million EUR (current prices) and 13,271 million EUR (chained prices) in 2023, followed by OO (*Public administration and defence; compulsory social security*) with 12,945 million EUR (current prices) and 10,498 million EUR (chained prices), as well as MA (*Legal and accounting activities*) with 8,598 million EUR and 9,609 million EUR. Together, these results show the importance of Brussels as a financial and administrative hub. In Flanders, Sector GG (*Wholesale and retail trade*) dominates, reaching 43,275 million EUR (current prices) and 34,400 million EUR (chained prices), followed by MA (*Legal and accounting activities*) for 32,774 million EUR (current prices) and 26,906 million EUR (chained prices), and LL (*Real estate activities*) with 30,402 million EUR and 25,144 million EUR. In Wallonia, sector LL (*Real estate activities*) is the largest sector, with 13,227 million EUR (current prices) and 11,261 million EUR (chained prices). This is closely followed by sector GG (*Wholesale and retail trade*) for 12,592 million EUR and 9,879 million EUR, and sector PP (*Education*) with 11,602 million EUR and 8,549 million EUR.

Growth rates are predicted to be 5.53% in Brussels, 5.79% in Flanders, and 5.81% in Wallonia in current prices. In chained prices, growth is more moderate at 3.06% in Brussels, 1.85% in Flanders, and even a slight decline of 0.24% in Wallonia. In Brussels, the sectors that are expected to grow most in 2023 are sector II (*Accommodation and food service activities*) (57.43%), sector EE (*Water supply*) (51.19%), and sector CL (*Manufacture of transport equipment*) (21.98%). In Flanders, the sectors that are expected to grow most are sector AA (*Agriculture, forestry and fishing*) (27.46%), CL (*Manufacture of transport equipment*) (21.42%), and JC (*Computer programming, consultancy and related activities; information service activities*) (20.71%). For Wallonia, they are sector CJ (*Manufacture of electrical equipment*) (47.52%), AA (*Agriculture, forestry and fishing*) (26.64%), and CL (*Manufacture of transport equipment*) (24.14%).

Contributions and future directions The methodology developed in this report provides a flexible and scalable framework for sector-regional gross value added estimation. It allows for iterative improvements, integrating new data as it becomes available, while the performance and accuracy of the various models and the ensemble model are expected to further improve over time.

Contents

1 Introduction

1.1 Context: the HERMREG model

HERMREG is the macro-economic modeling project conducted in partnership with the Federal Planning Bureau (FPB), the Brussels Institute for Statistics and Analysis (BISA), the Walloon Institute for Evaluation, Foresight, and Statistics (IWEPS), and Statistics Flanders (SV). The objective of this project is to produce regional economic estimates for key economic indicators such as Gross Domestic Product (GDP), Gross Value Added (GVA), employment, and investments by sector, as well as other indicators related to the labor market (including commuting patterns) and components of household disposable income.

The HERMREG project encompasses both a *top-down* and a *bottom-up* approach. In the *top-down* approach, regional statistics are derived to ensure full consistency with national projections from the HERMES model. This is primarily achieved through endogenous regional allocation keys. These statistics are used, among other purposes, for the Regional Economic Outlook, which has been published since 2008 by the Federal Planning Bureau as an extension of the National Economic Outlook, maintaining coherence between the two.[1] The *bottom-up* approach produces regional statistics that are not necessarily consistent with the national projections of HERMES. Here, national projections are the sum of the regional projections. This approach enables the simulation of certain asymmetric shocks, which would not be possible within the constraints of the *top-down* methodology.

This report and its related toolbox fall within the framework of HERMREG's *top-down* approach. In particular, the goal is to provide econometric estimates of value added by region and by sector, for the most recent year that is not yet available in the data.

1.2 Objective: estimating sector-regional value added

The estimation of sectoral-regional value added was previously based on observations of hours worked by employees and several hypotheses related to the evolution of employees' productivity and to hours worked by self-employed and their productivity. However, this estimation has not been econometrically validated. The objective of this project is to develop a systematic econometric methodology to estimate gross value added in volumes for the most recent year that is not yet available in the data. Our estimation leverages additional variables at the national level for the same year, as well as variables from previous years at the regional level. The current report focuses on estimating regional value added for the year 2023. The method will subsequently serve as input for further updates in the regional projections developed by the HERMREG team.

1.3 Operationalization: data and research design

We operationalize the project objective as follows:

1. **Collection of relevant data:** We provide a collection of different datasets that are known pro-cyclical indicators and predictors for sector-regional value added. Datasets vary in terms of time coverage, reporting frequency, and units of observation.

[1] The most recent regional economic outlook for 2024-2029 is available here.

2. **Methodology development:** We develop a thorough econometric method to estimate gross value added by region and sector for the most recent year not yet available in the data. We estimate several econometric models. Each model exploits different sources of variation in the data and assumptions on the data generating process, providing a multi-dimensional approach to estimating sector-regional gross value added.

3. **Model Selection and Validation:** We select and validate econometric models based on multiple statistical tests, including goodness-of-fit measures, diagnostic tests, and measures for out-of-sample predictive power such as cross-validation techniques.

4. **Ensemble model construction**: We develop an ensemble of several econometric models to provide multiple estimates for a given sector-region gross value added. Different estimates are then weighted using the model validation and selection metrics to provide a final estimate for sector-region value added. Alternatively, users can select a subset of models for the final prediction, based on model performance and additional domain knowledge.

5. **Ensuring national consistency:** We ensure regional estimates are consistent with national-level projections of the HERMES model. In particular, we adjust the values in current prices proportionally using regional weights for each sector, and recalibrate values in chained prices. Our method is similar to, but different from the method of the HERMREG model explained in Hoorelbeke et al. (2007) and Bassilière et al. (2008).

6. **Toolbox development:** We provide a code and model toolbox, including data, codes, and documentation, allowing the entire process to be automated. The toolbox is written in the open source language Python, exploits well-developed packages for statistical and machine learning models, and allows the HERMREG team to independently perform the analys for recurring updates.

The rest of this report is structured as follows. Section 2 explains the econometric research design to estimate gross value added. Section 3 presents the various data sources and pre-processing. Section 4 describes various dimensions of the processed data used for analysis. Section 5 to Section 9 present the different models, their assumptions and estimation methods, as well as the main results on the model estimates. Section 10 provides the final predictions for gross value added for each sector-region, consistent with the national projections from HERMES. Section 11 discusses the final predicted values of gross value added for the next year. Section 12 concludes. We relegate the details on the working of the code toolbox to the README file delivered with the toolbox.

2 Econometric design

2.1 General formulation

The goal is to predict gross value added for each Belgian sector-region for the most recent year that is not yet available in the data, exploiting information on past values of value added for these sector-regions, as well as additional covariates at the sector, region, and/or sector-region levels. Formally, in its most general form, we aim to estimate

$$Y_{irt} = f(Y_{irt-k}, \mathbf{X}_{irt}) \tag{1}$$

where Y_{irt} is gross value added for sector $i = 1, ..., N$ in region $r = 1, ..., R$ and year $t = 1, ...T$. Current values Y_{irt} are modeled as a function $f(\cdot)$ of lagged values of gross value added Y_{irt-k} for lags $k = 1, ..., K$, and potential covariates $\mathbf{X}_{irt}$. The vector of covariates is defined as $\mathbf{X}_{irt} = [X_{irt,1}, ..., X_{irt,L}]'$ where L is the number of covariates that may vary at the sector, region, or sector-region level . The function $f(\cdot)$ is specified in a generic form and is further parameterized in each of the models that we construct and estimate in the toolbox.

While eq(1) is conceptually simple, the challenge is selecting the parametric form of $f(\cdot)$ and the combination of variables Y_{irt-k} and $\mathbf{X}_{irt}$ that best predict Y_{irt}. This objective defines a so-called 'y-hat' problem, where the focus is minimizing prediction error to forecast the dependent variable as accurately as possible. By contrast, a more classic 'β-hat' problem centers on inference – estimating coefficients to understand the relationship between independent variables and the dependent variable. This process often involves statistical significance testing, confidence interval estimation, and potential causal interpretation. These two objectives involve distinct trade-offs. Prediction (y-hat) may prioritize model flexibility and tolerate some bias to reduce error, especially for out-of-sample forecasting. Inference (β-hat), on the other hand, might require other assumptions to produce consistent and unbiased parameter estimates, while not optimizing for out-of-sample prediction.[2]

2.2 Model evaluation

We emphasize that the focus of this project is on prediction, rather than on causal inference. The ultimate test of a model is its ability to produce reliable predictions on unseen data, which is evaluated using the validation Normalized Root Mean Squared Error (NRMSE) explained below. We do implement multiple statistical tests for the model assumptions. These assumptions guarantee the mathematical properties of estimators (e.g., unbiasedness, consistency), and can help us understand when a model may fail to generalize on unseen data. Moreover, estimated coefficients are used for prediction, and thus do contribute to the predictive power of the model. However, not satisfying these assumptions does not necessarily invalidate or deteriorate the model's out of sample performance.

Moreover, statistical tests act as diagnostic tools rather than gate keepers of the truth. For example, when using a p-value threshold of 0.05, we expect 5% of tests to fail by chance even if the null hypothesis is true. I.e. these are false positives (Type I errors). In our setup with up to 555 estimated models, this means approximately 28 models might fail purely due to random variation, even if the

[2]For more information on the difference between the y-hat and β-hat problems, see e.g. "An Introduction to Statistical Learning" (James et al. (2021)), or a discussion here.

ground truth is that all models pass the test. Small samples can further reduce the power of the tests, failing to reject or not where needed. We therefore implement several models, each with different assumptions, capturing multiple dimensions of variation in the data. This setup is robust to changing conditions, as models that perform poorly on some aspects may be offset by others that generalize better, as well as over time as new data arrives.

2.3 Data dimensions and implied model restrictions

The dimensions in the data are N sectors in R regions for T years. We have information for 37 sectors at the A38 level (see the list in Table A1), 3 regions at the NUTS1 level (Brussels, Flanders, and Wallonia), and currently 20 years (for the period 2003-2022) to predict the most recent year that is not yet available in the data (2023).[3] While the data are relatively detailed and span a substantial time period, its dimensions do impose some important restrictions on the models we can estimate, and their predictive power.

First, each model must have sufficient degrees of freedom to estimate its parameters.[4] Let n be the number of observations and p the number of parameters to be estimated. If $p > n$, the model becomes under-identified (or over-parameterized): the underlying system of equations has infinitely many solutions, and parameter estimates cannot be uniquely determined.[5] Second, insufficient degrees of freedom also imply variance estimates for parameter coefficients are unreliable or undefined because they depend on the residual degrees of freedom, leading to larger standard errors, wider confidence intervals, lower statistical power, and an increased risk of Type II errors (i.e. failing to detect effects when they are in fact present). Important in the current setting, additional issues include overfitting and poor generalization out-of-sample. Third, although non-linear models may better capture relationships between variables, they consume more degrees of freedom than linear models, exacerbating the challenges posed by small sample sizes.

For example, a univariate time series $ARIMA$ model estimated using Maximum Likelihood has $n - p$ residual degrees of freedom, where n is the number of observations used in the regression (adjusted for differencing or lags in time-series models), and p is the number of parameters. For an $ARIMA(1, 1, 1)$ model estimated on the current data, the residual degrees of freedom amount to $n - p = 16$.[6] More complex models with additional parameters for the same number of observations further decrease the degrees of freedom. Conversely, panel fixed effects models pool information across all sector-regions and time periods, significantly increasing the degrees of freedom at the cost of estimating common parameters.[7]

[3] We predict gross value added for 37 detailed sectors (not the "TOT" aggregate) for each of the three regions (not the "BE" aggregate). For models with covariates, we use their information for 2023 to forecast gross value added in 2023. We consider the link between the sector-regions with the aggregates at the end when ensuring consistency with the BE aggregates for the HERMES model.

[4] Degrees of freedom in statistical modeling represent the amount of independent information available to estimate parameters, adjusted for constraints (e.g., the number of predictors in a regression).

[5] For example, in linear regression, the normal equations $X'X\beta = X'Y$ cannot be uniquely solved if $X'X$ is not full rank due to insufficient observations or multicollinearity.

[6] One sector-region time series contains 20 yearly observations. If the series is non-stationary and integrated of order $d = 1$, one observation is lost due to differencing the series, leaving $n = 19$. The $p = 3$ parameters include one autoregressive term $AR(1)$ coefficient, one moving average $MA(1)$ coefficient, and a constant.

[7] The current data vintage has $N \times R \times T = 37 \times 3 \times 20 = 2,220$ potential observations to be used in a panel regression. Suppose we include 5 covariates that are not collinear with the chosen fixed effects. A model with sector-region, sector-year, and region-year fixed effects would then have $2,220 - 5 - 37 \times 3 - 37 \times 20 - 3 \times 20 = 1,304$ residual degrees of freedom.

2.4 Toolbox overview

To address these challenges, we implement the following strategy. We construct, estimate, select, and validate several econometric models to predict sector-region gross value added for a next year. Each model exploits different sources of variation in the data and has its own assumptions to identify parameters and to predict sector-region outcomes. The final estimates for value added for the most recent year are derived from a combination of various models, or an ensemble model in machine learning. The intuition is that combining multiple models generally results in better predictive power than relying on the individual strength of a single model.[8] By combining these approaches, we aim to balance model complexity, statistical power, and predictive accuracy, ensuring reliable projections despite the relatively low number of observations in the data.

Figure 1 provides an overview of the methodology. In particular, we implement the following steps to generate a combined prediction of gross value added at the sector-region level for the most recent year that is not yet available in the data:

1. Choose a model to estimate.
2. Construct and estimate each model using information on Y_{irt-k} and/or $\mathbf{X}_{irt}$.
3. Evaluate in-sample goodness of fit.
4. Perform post-estimation diagnostics to evaluate model assumptions.
5. Predict the value of gross value added for the next year for each sector-region.
6. Validate performance of the model out-of-sample using cross-validation techniques.
7. Aggregate model predictions to provide an ensemble prediction for each sector-region.
8. Ensure the predictions are consistent with the national HERMES model.
9. Evaluate the method as new data arrives to evaluate the current predictions.

[8] For an introduction on ensemble models, see e.g. "An Introduction to Statistical Learning" James et al. (2021) For deeper theoretical aspects, see e.g. "The Elements of Statistical Learning" Hastie et al. (2009).

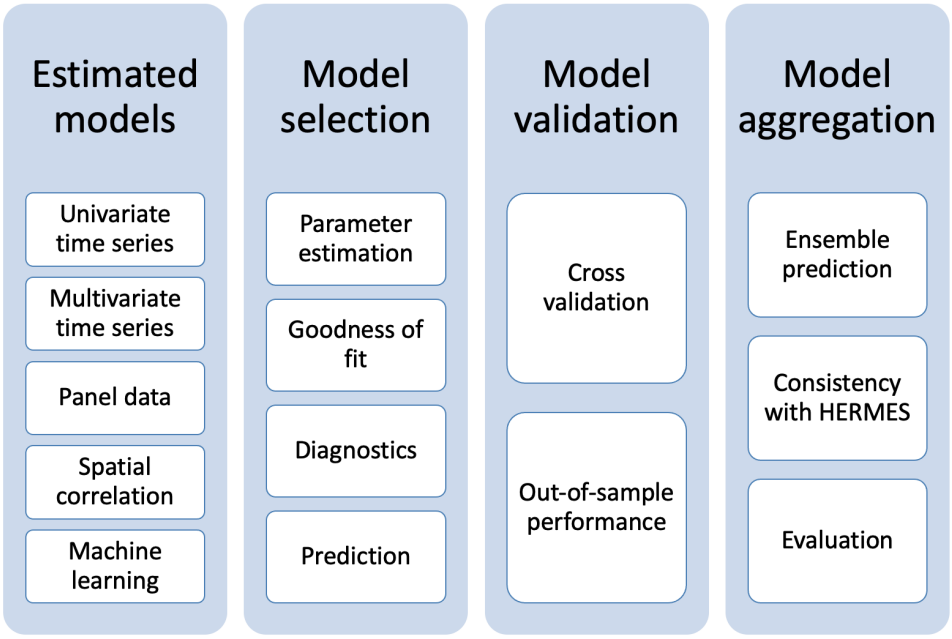


Figure 1: Toolbox overview.

Model estimation We construct and estimate the following models: (i) univariate time series models (ARIMA), (ii) Vector Auto-Regressive (VAR) and Vector Error Correction (VEC) models, (iii) panel data models (using fixed effects), (iv) Spatial Auto-Correlation models (with input-output linkages), and (v) machine learning classifiers (random forests). Each model captures different dimensions of the data, including temporal dependencies, cross-sectional heterogeneity, spatial interactions, and/or non-linearities.

Model selection While each model can generate predicted values for gross value added, some models are a better fit to the data than others. Model selection is done based on various tests, including for example finding optimal p, d, and q for $ARIMA(p, d, q)$ models using stationarity tests such as KPSS and selection criteria like BIC. Once a model variant is chosen, we calculate the in-sample (N)RMSE, which captures the difference between the data and the model fit. The lower this value, the better the model fits the data. This is an in-sample test for the model (i.e. which parameter values do we choose to optimize the model's fit to the data). We then predict the value of sector-region gross value added for the most recent year based on the lowest in-sample NRMSE value.

Model validation Next, each model is validated to evaluate its predictive power on unseen data. In particular, a model with many parameters might generate a very good fit on the data it is trained on (with a low in-sample NRMSE), but be terrible predicting out of sample (with a high validation NRMSE). To evaluate the model's out-of-sample prediction, we implement cross-validation techniques, catered to the particular model that is estimated. In particular, given an estimated model on the training set, we predict values of gross value added for the next year. We can compare these predictions to the observed data (up to 2022 for the report) in the validation set, which is withheld from the training set. A low validation NRMSE then implies the model does well in predicting 'unseen' data, at least up to the last year in the data. The cross-validation techniques we deploy include sliding window cross validation (SWCV) for time series models, leave-one-out cross validation (LOOCV) for panel data

models, and k-fold validation for the random forests. For each fold, we calculate its NRMSE and take the average as our validation NRMSE. We use the validation (out-of-sample) NMRSE as validation metric for the following reasons:

1. RMSE provides a direct measure of error magnitude in the same units as the target variable, making it easy to interpret. NRMSE normalizes RMSE, allowing comparisons across all sector-region-transforms that might have different ranges or units.
2. RMSE penalizes larger errors more heavily than smaller ones due to the squaring of residuals, making it sensitive to outliers. This is important when large deviations are critical to avoid.
3. For regressions, (N)RMSE or out-of-sample R^2 are typical validation metrics.
4. Many models optimize objectives (e.g., mean squared error) that are closely related to RMSE, ensuring consistency between training and validation evaluation.
5. (N)RMSE are standard metrics in the literature, making them comparable across studies.
6. Graphs provide qualitative insights but are less effective for quantitative and comparative evaluations.

Ensemble construction Next, each estimated model predicts a value for each sector-region-transform for the next year not yet available in the data. We combine these estimates across all models using an ensemble approach. We deploy a mean ensemble, which combines the different estimates for a given sector-region across all models by creating a weighted sum of predicted values, with weights derived from the out-of-sample NRMSE from the model validation part. I.e., a better model has a lower NRMSE and gets a higher weight to construct the final prediction. At this stage, users still have the full flexibility of selecting or preferring particular predictions over others. If users decide to implement additional criteria (such as dropping predictions for time series with structural breaks, those that fail to pass statistical tests etc.), this can be easily done by setting some weights to zero, allowing for the most flexible way to construct the final chosen results.

Ensuring consistency with the HERMES model The ensemble estimates must be rescaled to ensure consistency with the top-down version of HERMREG, meaning that the aggregation of regional gross value added aligns with the national values for that sector. These rescaled values are then the final prediction of gross value added for each sector-region.

Evaluation Finally, as the econometric framework will be applied to new data in future iterations of the HERMREG predictions, both historical data and predictions should be used to evaluate and re-optimize the combination of models. For example, previously out-of-sample predictions will become part of the sample, allowing to evaluate each model's predictive power. This can lead to updates of the particular weights chosen for each model in generating the final ensemble estimate. This iterative process ensures that the ensemble method adapts over time, further improving predictive accuracy as more data becomes available.

2.5 Data and coding pipeline

We provide the various datasets to predict the values for 2023, as well as the entire coding pipeline to estimate, select, validate, and aggregate all models as part of this project. The toolbox is written in the open source language Python, and is made available to the HERMREG team upon completion of the project. More information on the toolbox setup, the workflow, and its flexible, modular nature, is provided in the repository's Readme file. Figure 2 provides an overview of the toolbox folder structure.

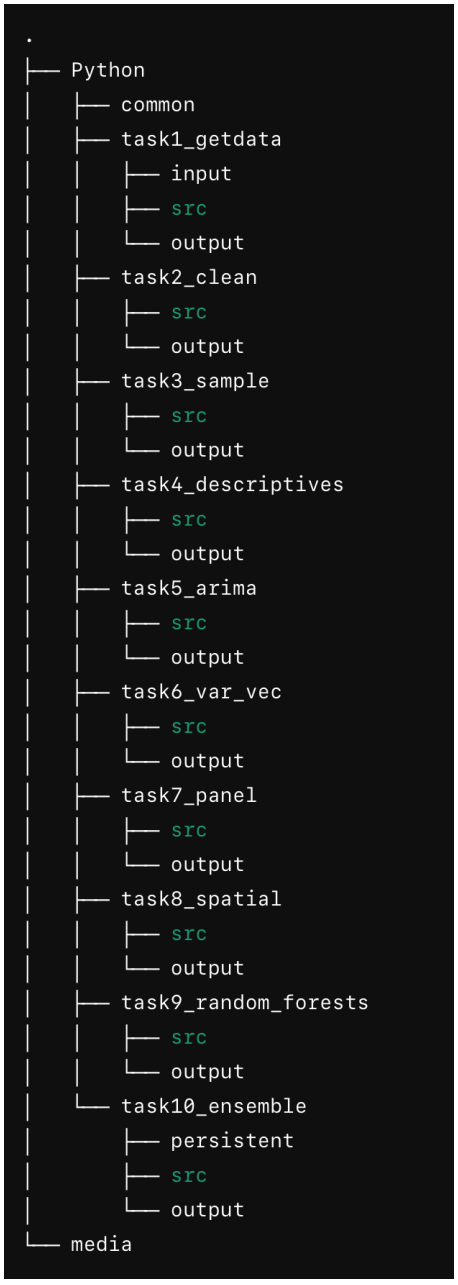


Figure 2: Toolbox overview.

3 Data sources and construction

In this section, we describe the datasets and variables used to predict gross value added for each sector-region. We discuss the dimensions, coverage, and cleaning of the variables, and how they are harmonized to construct a final dataset for analysis. We also discuss some additional data that might be used in a future iteration of the toolbox.

3.1 Dataset and key variables for analysis

We collect several datasets to estimate sector-region gross value added for the next year not yet available in the data. Table 1 provides an overview of these datasets, the data providers, the key variables used for analysis, and coverage of these variables.[9] The main dataset is the Annual Regional Accounts data, which contains information on gross value added in both current prices and chained prices. The other datasets are used to construct potential predictors for gross value added, including hours worked of employees and self-employed, labor compensation, VAT information, Multi-Regional Input-Output (MRIO) tables, and several typical pro-cyclical indicators such as the business confidence index, building permits, and labor market variables.

Some variables contain information at the sector-region level, while other are at a more aggregate or "macro" level. To match the units of observation in the gross value added data, we aggregate or disaggregate variables to sector-regions at the NACE A38 and NUTS1 levels where needed. If present, we drop the aggregate A38 sector ("TOT"), as well as the Belgian aggregate region ("BE" or "T") for the descriptive analysis and model predictions. Table A1 in Appendix A provides an overview of the A38 sectors, as well as the correspondences between sectors from the different datasets. Similarly, we aggregate some variables that are at a monthly or quarterly frequency to yearly values. Imported and cleaned datasets are merged into a single dataset for further analysis, containing all variables at the sector-region-year level. We describe these datasets and their construction for analysis in more detail below.

Dataset	Provider	Variables	Coverage	Missing
Regional Accounts	HERMREG	Gross value added (current prices, million EUR) Gross value added (chained prices, million EUR)	3 regions + Belgium, A38. Yearly, 2003-2022. 2023 only for Belgium A38.	2023, 3 regions (to be predicted).
		Hours worked (employees, thousands) Hours worked (self-employed, thousands) Compensation of employees (million EUR)	3 regions + Belgium, A38. Yearly, 2003-2023.	-
VAT Statistics	HERMREG	VAT turnover (million EUR) VAT purchases (million EUR) VAT investments (million EUR)	3 regions + Belgium, A38. Yearly, 2005-2023.	2003-2004, Not VAT-liable sectors: AA, KK, LL, OO, PP, QA, QB, TT.
MRIO Tables	Federal Planbureau	Direct requirements matrix Total requirements matrix	3 Regions, A38. Year 2015.	Only the year 2015 is available.
Business Survey	NBB	Business confidence index (synthetic curve)	3 Regions, 4 aggregate industries + total. Monthly, 2003-2023.	Belgium, A38 sectors.
Construction permits	NBB	Number of buildings (units) Number of dwellings (units) Number of buildings with one dwelling (units) Surface area (m2) Habitable surface area (m2) Volume (m3)	3 Regions + Belgium, no sectors. Monthly, 2003-2023.	A38 sectors.
Labor Market Data	Statbel	Employment rate (%) Unemployment rate (%) Activity rate (%)	3 Regions + Belgium, no sectors. Quarterly, 2003-2023.	A38 sectors.

Table 1: Main variables and their coverage.

[9] In the Python toolbox, task 1 "Get data" is about datasets. It imports the various datasets, provided in a spreadsheet format (.csv or .xlsx), and recasts them into .csv format for further use. Variables are given permissible names and unneeded information is trimmed (e.g. empty variables). Task 2 "clean" is about variables. It reshapes the data as required, renames variables, corrects mistakes, and collapses data into the proper sector-region-year format. Next, task 3 "sample" collects the various individual datasets and creates a single dataset for analysis, with information on various variables at the sector-region-year level. It selects the time dimension, and which variables are to be included in the final dataset.

Regional Accounts. The Annual Regional Accounts (ARA) dataset contains information on gross value added for the three Belgian regions and Belgium, for 38 NACE A38 sectors (37 sectors + 1 aggregate), over the years 2003-2022 (annual frequency), and is provided by the HERMREG team. The dataset contains information on 5 variables: QVU: gross value added (current prices, million EUR); QVO: gross value added (chained euros, base year 2015, million EUR); NFH: hours worked by employees (thousand units); NIH: hours worked by self-employed (thousand units); and WS: compensation of employees (current prices, million EUR). The following sectors have zero hours worked for self-employed for all years and all regions: *Cokes and refined petroleum products (CD)*, *Electricity, gas, steam and air-conditioning (DD)*, *Public administration and defense (OO)*, and *Activities of households as employers (TT)*. In 2023, data for gross value added is only available for each sector at the national level. This information will be used to ensure consistency with the national values at the end of the toolbox, so that the sum of gross value added for a given sector across the three regions is equal to the value for that sector for Belgium as a whole.

Value Added Tax Statistics. The Value Added Tax (VAT) dataset contains information on VAT turnover, purchases, and investments for each of the three regions and Belgium for 30 NACE A38 sectors (29 sectors + 1 aggregate), over the years 2005-2023 (annual frequency), and is provided by the HERMREG team. There are 3 variables: CA1: VAT turnover (current prices, million EUR); AC1: VAT purchases (current prices, million EUR); and IN1: VAT investments (current prices, million EUR). Data is not available for the years 2003-2004. Eight sectors are not VAT liable, and thus do not report VAT information: *Agriculture, forestry and fishing (AA)*, *Financial and insurance activities (KK)*, *Real estate activities (LL)*, *Public administration and defense; compulsory social security (OO)*, *Education (PP)*, *Human health activities (QA)*, *Social work activities (QB)*, and *Activities of households as employers (TT)*. Moreover, the sectors *Arts, Entertainment and Recreation (RR)* and *Other Service Activities (SS)* report data for each variable but are based on incomplete VAT statistics.

Multi-Regional Input-Output (MRIO) tables. We use the MRIO tables for Belgium for the year 2015 in the spatial autoregression model in Section 8. The MRIO tables are produced by the Federal Planbureau, but are not publicly available. They have been obtained under an agreement with the Federal Planbureau for this project. These tables contain information on intermediate goods sales and inputs, as well as the components of value added and final demand for the 3 regions and 124 NACE sectors. An observation in the intermediate goods matrix is the value of sales from one sector-region to another. From the intermediate goods matrix and total sales vectors, we construct the direct and total requirements matrices. We first aggregate the 124 NACE sectors to A38 sectors by summing over row and column values for intermediate goods and total sales. We then calculate the direct and total requirements matrices. Additional information on the construction is provided in Section 8. We also provide a heat map of the total requirements matrix across all sector-regions in Figure B3.

Business survey. This dataset contains information on how business managers perceive their current business environment, for the three regions and four broad sectors (Business-related services, Manufacturing, Structural building work, and Trade), over the years 2003-2023 (monthly frequency). The data is collected from the NBBStat website. Survey questions relate to assessing the current business situation and the expectations for the next three months, for production, order books, employment, and prices. An aggregate synthetic curve is constructed as a weighted average of

responses, with weights given by the importance of the firm within each activity covered by the survey. The value of the synthetic curve represents the balance between positive and negative responses: a negative value implies that more leaders evaluate the current and near future situation as deteriorating.[10] The four sectors are then mapped to NACE A38 sectors by using a correspondence constructed by the project team (see Table A2 in Appendix A). The index for Brussels is only available for the aggregate sector and starts in 2008. We include additional data for Brussels at the broad sector level, spanning the period 1980-2017, provided by the HERMREG team. This allows to add information for Brussels and four sectors for the period 2003-2017. Information on the four sectors for Brussels for the period 2018-2023 is obtained from the NBB.

Construction permits. Information on building permits is collected from the NBBStat website. This dataset contains information for the 3 Belgian regions and Belgium, over the years 2003-2023 (monthly frequency). There are six main variables: number of buildings (units); number of dwellings (units); number of buildings with one dwelling (units); surface area (m^2); habitable surface area (m^2); volume (m^3). The original variables are split across residential and non-residential buildings. We sum these into total values per variable. Construction is a classical pro-cyclical sector, which might be a good predictor of gross value added.

Labor market data. Finally, information on employment and unemployment is obtained from the Labour Force Survey and collected from the Statbel website for the 3 regions and Belgium, over the years 2003-2023 (quarterly frequency). Datasets are initially separate for periods 2003-2016 and 2017-2023 due to changes in the survey setup, and are merged to construct a continuous time series. There are three variables: Employment rate (%); Unemployment rate (%); and Activity rate (%). Similar to construction permits, employment rates are a classic pro-cyclical macro indicator for economic activity.

Additional datasets. We have explored several other datasets and variables that are typically considered as informative predictors of gross value added. These include industrial production, international trade, and consumer confidence indicators. Unfortunately, we cannot include these variables in the current toolbox. First, industrial production data from the Statbel website covers only manufacturing sectors. More problematic, within this subset of sectors, the time series coverage as well as the regional coverage fluctuates (Brussels is the most problematic). While we construct and estimate separate models for subgroups of sectors, forcing to include the industrial production data would drop many sector-region observations. We are convinced that a complete coverage for these 13 sectors exists for the entire span of the time series, as well as for the three regions. We have currently provided codes to extract and clean this industrial production data, such that, if a complete dataset becomes available, the HERMREG team can include these easily as additional covariates for the Manufacturing sector analysis. Second, international trade data at the regional level is inconsistent over time across different datasets. While there is detailed data available for exports and imports for both goods and services at the level of NACE A64 sectors through the Regional Accounts distribution of imports and exports at NUTS 1), the data is not available for the last year (for this report, 2023). Conversely, detailed and very recent data is available through the NBB's Foreign Trade for each region using the National Concept. But it does not go back in time up to 2003, only up to 2014. We have tried

[10] More information on the methodology can be found here.

to combine several datasets spanning both NACE and trade (Harmonized System) nomenclatures, to obtain a consistent dataset that spans 2003-2023. A final data request had been sent to the NBB, which has been unresolved at the time of the writing of this report. The ideal dataset would contain yearly import and export values for the three regions at the level of A38 (or A64) sectors, for the period 2003-2023. If such data becomes available, the coding pipeline is written flexibly to pre-process, transform, and clean the data for inclusion in the model estimations. We have also prepared the required correspondence table from NACE A64 to A38 (see Table A2 in Appendix A). Third, we have not included the consumer survey information. Again, data for Brussels only starts from 2009 onwards. Perhaps complete data exists somewhere. Finally, we have also explored DynamStat for more detailed data on (un)employment. Unfortunately, while this dataset is very detailed, data for 2023 was not yet available at the time of writing of the report, and also the data for 2022 was still provisional.

3.2 Aggregated industries and covariates

Several variables contain information only for a subset of sectors. For example, not all sectors are VAT liable. In order to maximize their use as covariates while avoiding sector-regions being dropped from missing observations in the covariates, we construct a taxonomy of aggregated industries that group sectors. For models with covariates, we then run the models separately on the sectors within these aggregate industries. See Table 2 for the broad classification and which variables can be used for sectors within these industries. We group A38 sectors into *"Primary and extraction"*, *"Manufacturing"*, *Services"*, and *"Non-market services"*. We have allocated *Utilities (DD and EE)* to *"Services"* as it pertains mostly to public infrastructure services, and it is the same allocation as in the Business Confidence dataset. We have allocated *Construction (FF)* to *"Manufacturing"*, as it mostly involves the creation of physical assets.

There is also an imperfect overlap in terms of VAT coverage. First, for models with VAT as a covariate, we run the models on data for the years 2005-2023. Second, to maintain the logical grouping of sectors into *"Primary and extraction"*, *"Manufacturing"*, *"Services"* and *"Non-market services"*, we set VAT values equal to zero for the sectors *Financial and insurance activities (KK)*, and *Real estate activities (LL)*. These services sectors do not charge VAT on their activities over the sample period, and are labeled "excluded from VAT statistics", as specified in the VAT dataset from the HERMREG team. Conversely, we do not use the VAT data for *Mining and quarrying (BB)*, which we allocate to *"Primary and extraction"*, and the sectors *Arts, entertainment and recreation (RR)* and *Other service activities (SS)*, which we allocate to *"Non-market services"*.

Broad Industry	**NACE A38 Sectors**	**Covariates come from datasets**
Primary and extraction	AA to BB	regional accounts, construction permits, labor markets.
Manufacturing	CA to CM and FF	regional accounts, construction permits, labor markets, VAT, business survey.
Services	DD, EE and GG to NN	regional accounts, construction permits, labor markets, VAT, business survey.
Non-market services	OO to TT	regional accounts, construction permits, labor markets.

Table 2: Broad industries, NACE A38 sectors, and covariates used for analysis.

4 Descriptive statistics

In this section, we describe the variables that are collected in the final dataset for analysis. We provide summary statistics and show the significant dispersion and skewness of some variables, suggesting the need to transform these for further analysis. We then turn to correlations. A larger covariance (potentially after transforms) between the dependent variables and covariates generally implies a better model fit. Next, we provide results on the evolution of sector-region gross value added, and the potential co-movement of the same sectors across regions. This co-movement can be exploited when predicting the multi-variate time series. Finally, we report results on the existence of structural breaks in the time series of sector-regions.[11]

4.1 Summary statistics

A first requirement after preparing the final dataset is to ensure that all variables have the expected number of observations. In particular, for N sectors, R regions, and T years, we generally expect $N \times R \times T$ observations. Hence, for the current vintage of the data (up to 2023), we expect $37 \times 3 \times 21 = 2,331$ sector-region-year observations. For gross value added, we have one year less, resulting in 2,220 observations. VAT is only available for 29 sectors across three regions and for the years 2005-2023, so we expect $29 \times 3 \times 19 = 1,653$ observations, and so on.

Next, Table 3 shows summary statistics for all numeric variables. Gross value added (measured in either current or chained prices) is on average 3.2 billion euro for a sector-region over the time period 2003-2022. There is sizable variation, with a sector-region-year observation at the 10th percentile generating only 120 million euro gross value added, while an observation at the 90th percentile generates almost 9 billion euro, or 75 times more. Similarly, while the average number of hours worked by employees is close to 50 million hours, a sector-region-year at the 10th percentile accounts for 2 million hours worked, while at the 90th percentile it is 147 million hours, again roughly a factor of 75 difference. Turning to hours worked by self-employed, we see that there are several sector-regions in which there are no self-employed hours reported: the 10th percentile value is zero. A similar significant skewness can be found for compensation of employees, VAT turnover, purchases, and investments, either measured in mean over median or the p90/p10 ratio. Other variables, such as the employment, unemployment, and activity rates, are much more centered around their mean value and with less variation.

4.2 Variable transforms

These numbers imply substantial variation across sector-regions for some variables, including the main variable of interest, gross value added. In Figure 3, we show such dispersion for selected variables, pooled over all years: while most sector-region-year observations are relatively small, a few observations are significantly larger, often spanning multiple magnitudes. Moreover, these variables are defined on the positive domain, and they exhibit both a high variance and substantial

[11] Task 4 "descriptives" in the Python toolbox generates all the results in this section, as well as many additional graphs and tables for individual variables etc., which are not included in the report. For example, we generate histograms for all numeric variables in levels, both pooled and for the last complete year in the data, currently 2022. The toolbox also includes a sanity check for the users that calculates the expected and observed number of observations for each variable for the current and each future vintage of the data, to ensure all data is available in the raw datasets and has been properly transformed and harmonized into the final dataset.

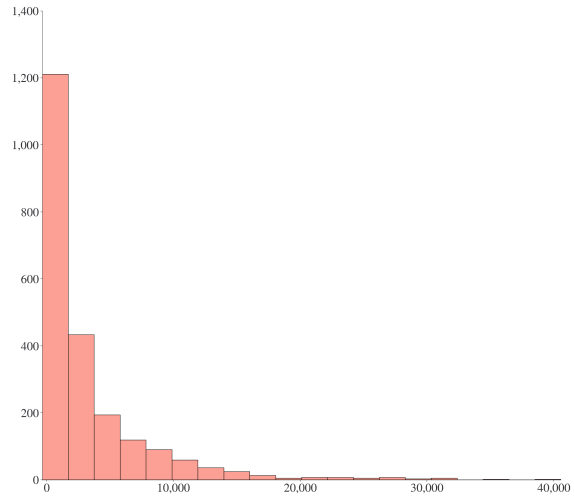
				percentiles				
Variable	N	mean	stdev	p10	p25	p50	p75	p90
GVA (current prices, million EUR)	2,220	3,192	4,618	120	437	1,455	3,963	8,686
GVA (chained prices, million EUR)	2,220	3,258	4,634	124	458	1,484	4,092	9,112
Hours worked by employees (th. units)	2,331	49,861	75,224	2,050	6,767	18,879	55,973	146,948
Hours worked by self-employed (th. units)	2,331	15,420	47,126	0	121	1,413	8,349	44,174
Compensation of employees (million EUR)	2,331	1,829	2,758	83	257	708	2,134	5,266
VAT purchases (million EUR)	1,653	10,446	30,200	231	671	2,440	7,419	24,586
VAT turnover (million EUR)	1,653	11,868	32,338	324	968	3,292	9,065	28,358
VAT investments (million EUR)	1,653	410	709	10	37	170	450	986
Business confidence indicator	1,764	-6.87	9.10	-17.99	-12.53	-7.43	-1.15	4.36
Building permits (th. units)	2,331	10.51	9.41	0.26	0.34	8.32	20.10	23.98
Building permits, one dwelling (th. units)	2,331	7.79	6.83	0.11	0.13	6.64	14.95	17.46
Dwellings (th. units)	2,331	18.02	15.48	2.45	3.36	11.83	36.00	42.28
Surface area, habitable (mil. m^2)	2,331	1.38	1.42	0.12	0.22	1.15	1.63	3.83
Surface area (mil. m^2)	2,331	5.45	5.09	0.52	0.77	3.22	11.53	12.96
Volume (mil. m^3)	2,331	23.12	22.74	1.77	2.60	12.58	50.22	57.98
Employment rate (%)	2,331	0.60	0.05	0.54	0.56	0.57	0.66	0.68
Unemployment rate (%)	2,331	0.10	0.05	0.04	0.05	0.11	0.13	0.17
Activity rate (%)	2,331	0.67	0.03	0.64	0.64	0.66	0.69	0.71

Table 3: Summary statistics of variables used for analysis.

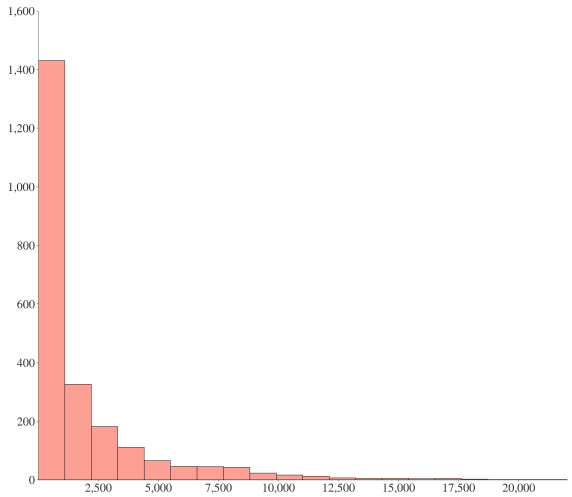
right-skewness.

These patterns suggest that transforming variables may be necessary for estimating models and forecasting values. First, all models except the machine learning models are linear in the dependent versus predictor variables. These models generally perform better when a strong linear relationship exists between the dependent variable and its predictors. Second, for highly skewed variables, the mean (predicted) value may not adequately represent any given unit in the population. Third, due to their rescaling and/or re-centering of variables, transformations can also help improve speed of convergence for iterative estimation procedures. We therefore estimate each model for the following five variable transforms: levels, natural logarithm, square root, inverse, and standardized. In particular, for gross value added Y_{irt}, we construct the following transforms:

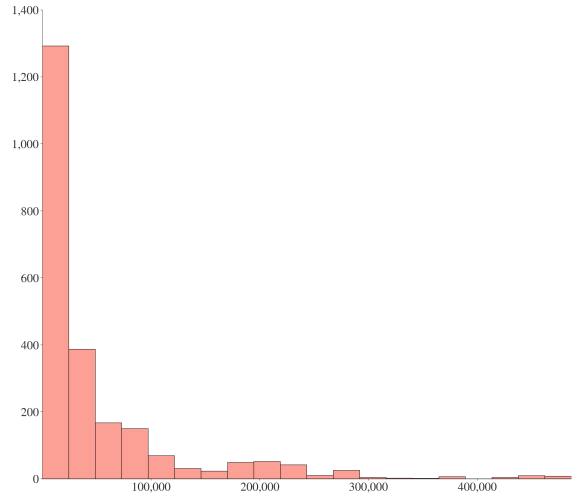
1. **Levels:** $\tilde{Y}_{irt} = Y_{irt}$ (untransformed),
2. **Natural logarithm:** $\tilde{Y}_{irt} = \ln(Y_{irt})$ (defined for $Y_{irt} > 0$),
3. **Square root:** $\tilde{Y}_{irt} = \sqrt{Y_{irt}}$ (defined for $Y_{irt} \geq 0$),
4. **Inverse:** $\tilde{Y}_{irt} = \frac{1}{Y_{irt}}$ (defined for $Y_{irt} \neq 0$),
5. **Standardized (z-score):** $\tilde{Y}_{irt} = \frac{Y_{irt} - \mu_{ir}}{\sigma_{ir}}$,



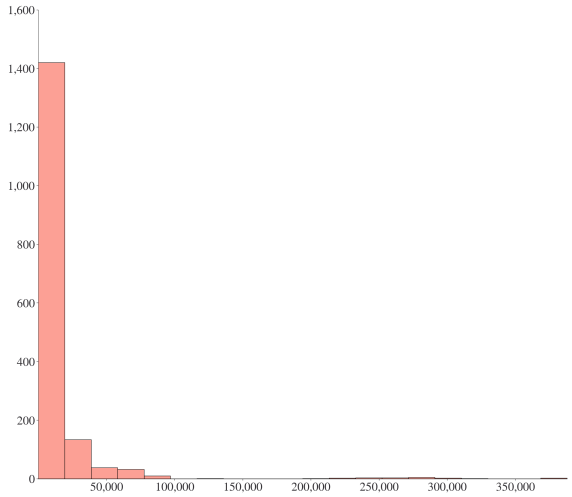
(a) GVA in current prices



(b) Compensation of employees



(c) Hours worked by employees



(d) VAT turnover

Figure 3: Skewed distributions of key variables (pooled).

where μ_{ir} is the time series mean, and σ_{ir} the time series standard deviation of Y_{irt}. Each transform has its own use. First, for models like linear regression, raw variables with high skewness or different scales can reduce model fit, making transformations necessary. Conversely, many machine learning models (e.g., random forests) work well with raw variables because they do not rely on assumptions of linearity or distribution. Second, natural logs are often used in economics, and are common for variables on the positive domain and with heavy right-skewness (e.g., GDP or population across countries or regions). The transform converts multiplicative relationships into additive ones, which many linear models (e.g., linear regression) handle better. One drawback is that it cannot handle zero or negative values without an ad hoc adjustment in many models. Third, the square root reduces the impact of large values while preserving relative orders, making it less aggressive than the logarithm. It is often used for count data (e.g., event occurrences). A drawback is that it is again only defined for values on the positive domain. Fourth, the inverse transform compresses large values and emphasizes differences in small values. It is often applied when large values dominate and obscure smaller patterns. The downside is that it is highly sensitive to very small values. Finally, standardization is

very often used in machine learning. It centers a variable so that its mean is zero and the variance is one. It is essential for many models that are sensitive to the scale of features (e.g., linear regression, support vector machines, principal component analysis, or neural networks).

Some sector-region-year observations contain negative values. For example, gross value added is negative for some years in the Brussels "*Coke and Refined Petroleum Products (CD)*" sector, and the business confidence index also exhibits negative values. Some transformations, like logarithms and square roots, cannot be applied to negative values. In that case, we do not estimate the models for these particular sector-regions and transforms. For these, we resort to other transforms like levels, inverse, and standardized variables, so that we still generate multiple predictions for each sector-region per estimated model.

4.3 Correlations

Turning to the relationship between gross value added and the potential predictors, Figure 4 shows the correlation matrix between all numeric variables in levels, i.e. without transformations. Red cells indicate a positive correlation, while blue cells a negative one. Gross value added, either in current or chained prices, is strongly positively correlated with hours worked by employees, compensation of employees, and the VAT variables. Gross value added also covaries positively with the construction permits and employment and activity rate. The correlation with the business confidence index is close to zero. Finally, it correlates negatively with the unemployment rate. We provide additional correlation graphs for each of the variable transforms in Appendix B, as well as full tables with correlation coefficients as part of the output of the toolbox.

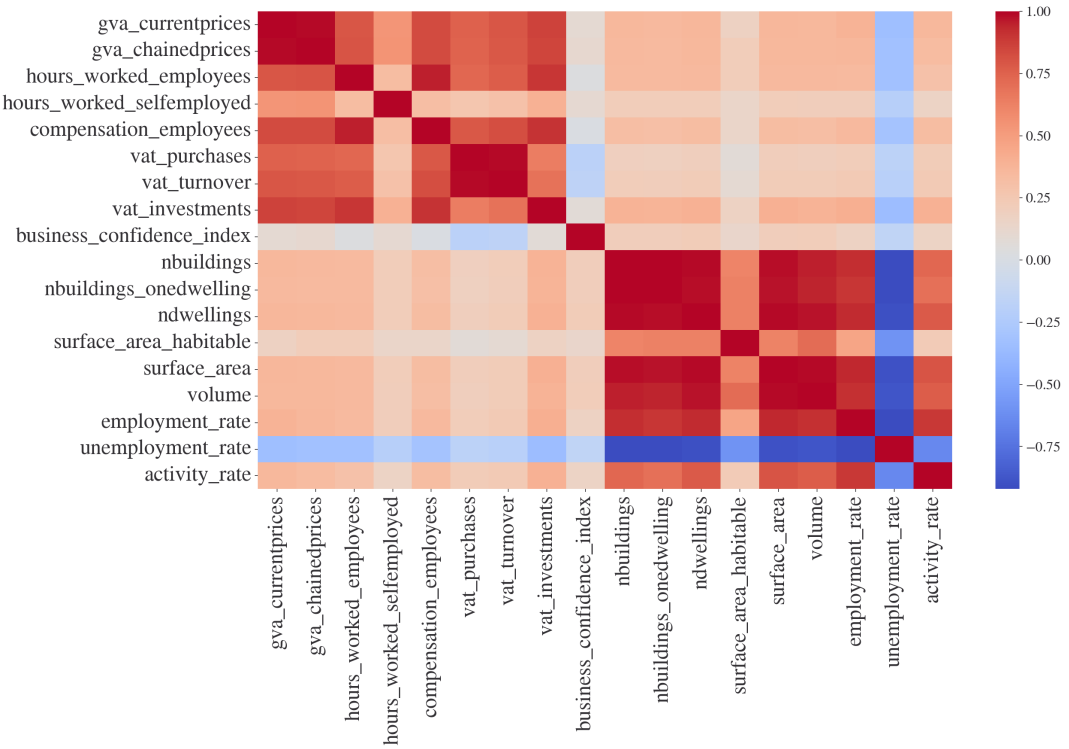


Figure 4: Correlation matrix (raw, levels).

4.4 Time series evolution and growth rates

We report the time series evolutions of gross value added. We start with aggregating gross value added in current prices by region in Figure 5. In 2019, Flanders reaches a gross value added of 249 billion euro, followed by 100 billion euro for Wallonia, and 78 billion euro for Brussels. These numbers naturally coincide with the gross value added in current prices as reported in the regional accounts on the NBBStat website. We then show gross value added for each sector, aggregated across regions, in Figure 6. By far the largest A38 sector in Belgium is the sector "*Wholesale and retail trade (GG)*", with value of almost 58 billion euro in 2022. This is followed by "*Legal and accounting activities; activities of head offices; management consultancy activities; architecture and engineering activities; technical testing and analysis (MA)*" with a value of 46 billion euro in 2022, and "*Real estate activities (LL)*" with 46 billion euro in 2022.

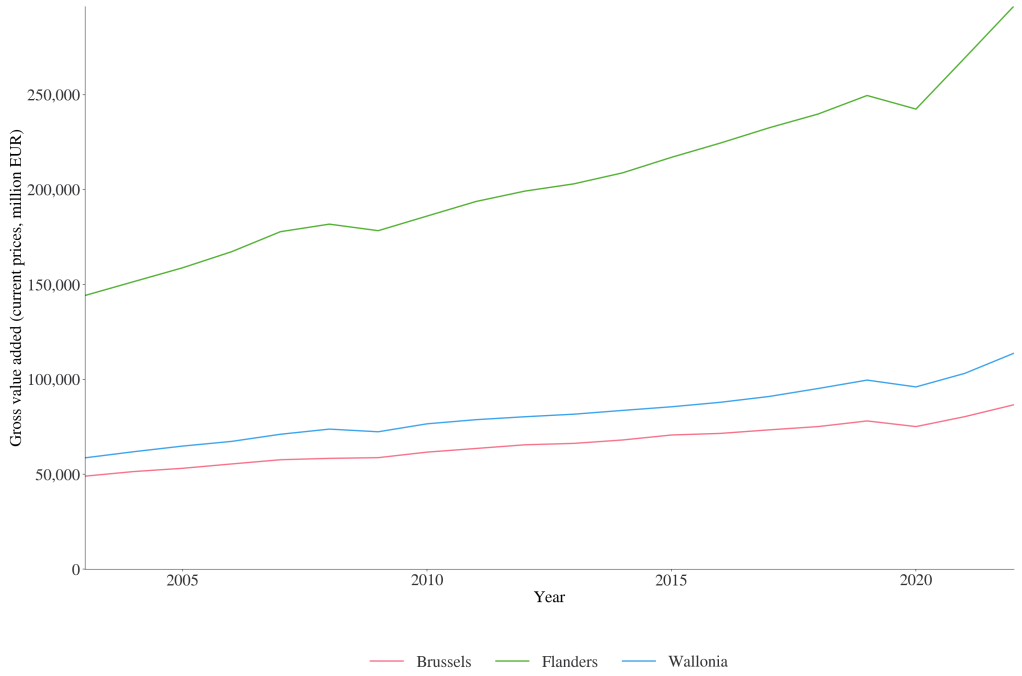


Figure 5: Gross value added across regions.

We then turn to the growth rates of sector-regions. We calculate the yearly percentage growth rate of each sector-region, and then plot the growth rates of the same sector across the three regions. We produce similar graphs for all sectors as output in the toolbox. In Figure 7, we focus on the three largest sectors for each region in 2022: "*Financial and insurance activities (KK)*" for Brussels, "*Wholesale and retail trade, repair of motor vehicles and motorcycles (GG)*" for Flanders, and "*Real estate activities (LL)*" for Wallonia. We use these sector-regions as a running example throughout the report. We then show the growth rates of these sector-regions across the three regions. We see that sectors tend to co-move across regions. For example, while real estate might be different in terms of size across the three Belgian regions, we would still expect similar growth rates of real estate activities across regions, due to economy-wide business cycles or sector-specific market evolutions. We can exploit this co-movement in the multivariate time series analysis in Section 6. Some sectors tend to co-move, and follow business cycles very well, including the 2008-2009 financial crisis and more recently the 2020-2021 Covid-19 pandemic. Other sectors tend to co-move less and/or vary less with the business

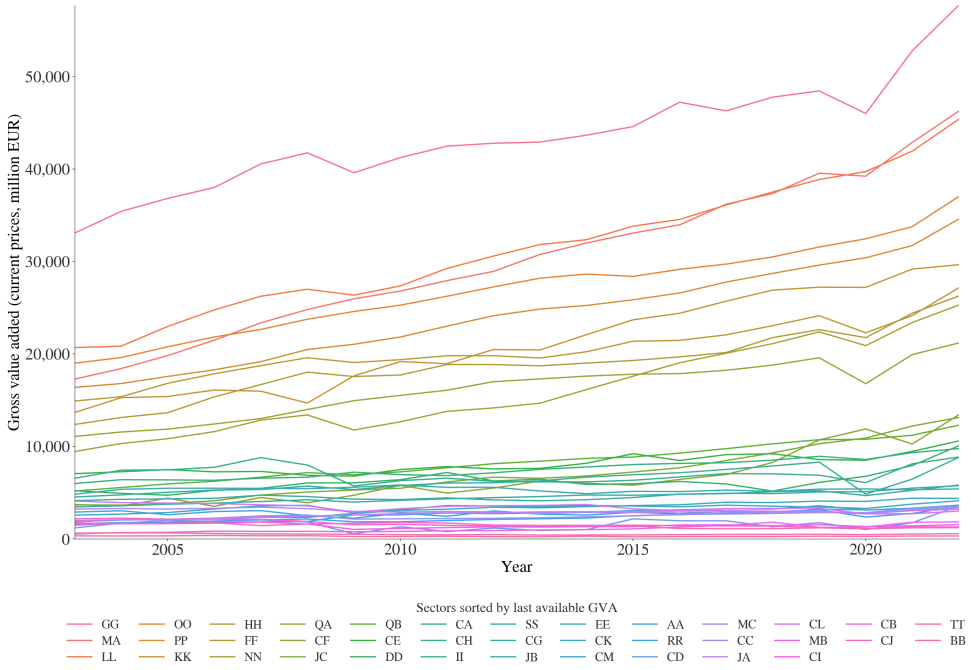


Figure 6: Gross value added across sectors.

cycle.[12]

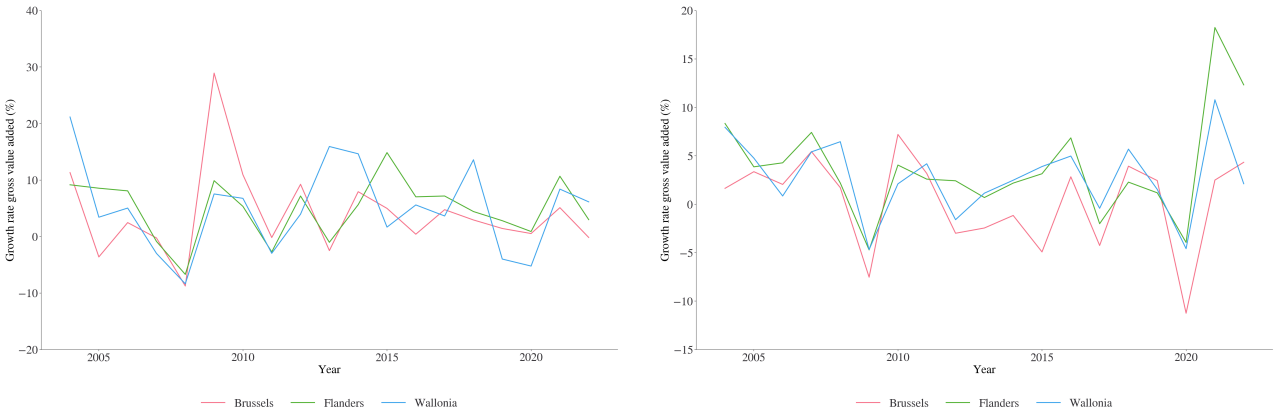
Finally, we show the correlation matrix for all numeric variables in terms of growth rates in Figure 8. While the correlation in levels is relatively high across variables, we see that this relationship mostly disappears when looking at growth rates. We exploit the strong correlation in levels by predicting gross value added in levels and recover predicted growth rates from these.

4.5 Structural breaks

We test for the existence of potential structural breaks in the time series of sector-region gross value added. Standard time series models (e.g., ARIMA and VAR/VEC models) assume that the underlying data generating process is stationary or at least stable over time. A structural break violates this assumption. A model will try to fit the entire dataset, but after the break, the relationship between variables or the structure of the data might change. This can result in biased parameter estimates as the model averages over pre- and post-break dynamics. For example, it is possible that some sector-regions have been particularly affected during the financial crisis in 2008-2009 or the Covid crisis in 2020-2021. Pre-break growth trends will heavily influence the model, leading to poor post-break forecasts.

In principle, one should then estimate a model separately for the pre- and post-break segments, or implement a regime switching model. However, given the relatively short time series in the current data and the low degrees of freedom in the time series models, we do not impose estimating any model on a subset of the data after the structural break. We therefore only flag time series that have an identified structural break in this report. Users can further investigate potential issues with model performance for time series with structural breaks. As longer time series will become available in the

[12]For the sectors with negative value added in some years, we impose the value in the denominator to be the absolute value to avoid negative growth just because the value for the previous year is negative when generating percentage growth rates.



(a) Financial and insurance activities (KK)

(b) Wholesale and retail trade (GG)

(c) Real estate activities (LL)

Figure 7: Sector-region growth rates across regions.

future, researchers can choose to estimate time series models on post-break periods.

We implement the structural break test as follows. First, for every transformation of gross value added, we perform a KPSS stationarity test and difference the series until stationary with a maximum of 2 differences.[13] Second, we perform a CUSUM (cumulative sum) test to test for a single break at an unspecified moment in the stationary time series. The CUSUM test is used to detect structural breaks in time series data by analyzing the cumulative sum of the residuals from an OLS regression model. The test helps identify points in time where the statistical properties of the time series change. Third, if the CUSUM test statistic is greater than the critical value at the 5% level, we indicate that there is a structural break in the time series.

Table 4 shows the result of these CUSUM tests. For each of the 111 sector-regions, we have 5 variable transforms, for a total of 555 sector-region-transforms.[14] Out of these, we find that 26 time series-transforms, or 4.68% of the time series, have a structural break over the panel period. When comparing the incidence of structural breaks across transforms in Table 5, we see that structural breaks are identified for roughly 2.7% of the time series in each transform, except for the inverse transform, which flags up to 12.6% of the time series as having a structural break. This might be due to the inverse

[13] We describe the KPSS test and all other tests in more detail in Appendix D.

[14] For sector-region-transforms with undefined values (e.g. logs for negative values of gross value added), we drop undefined observations, and perform the CUSUM test on the remaining observations within a time series.

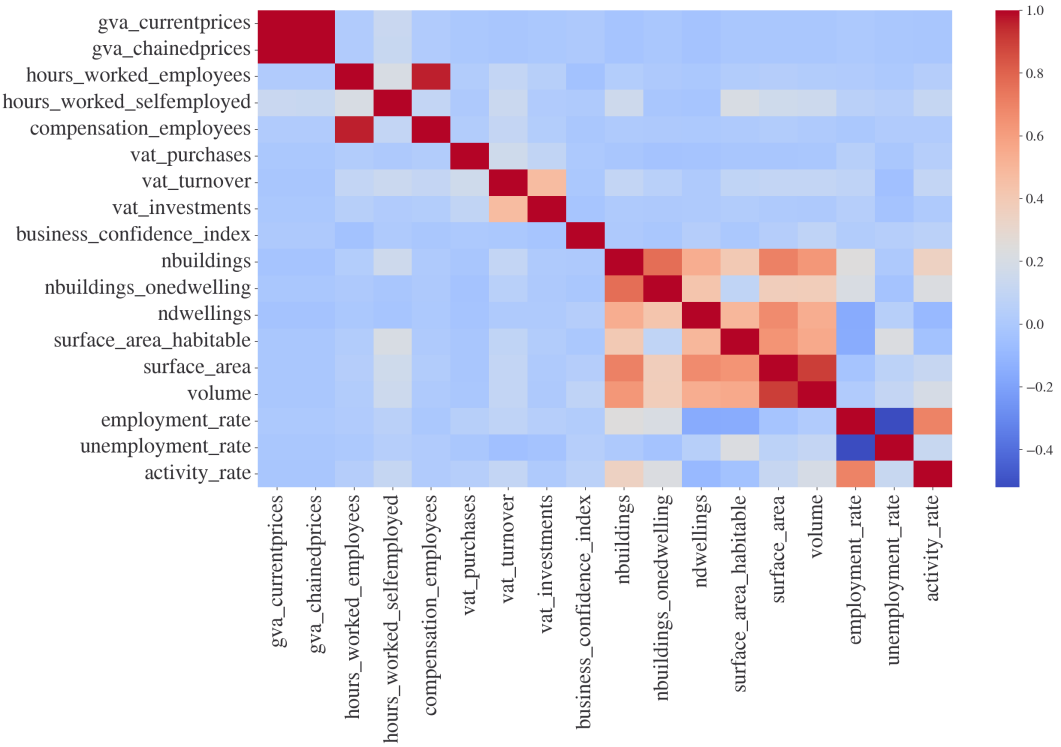


Figure 8: Correlation matrix (raw, growth).

transform magnifying small changes, leading to excessive structural breaks. We also provide a list of sector-region-transforms with structural breaks after differencing in Table B1.

Structural Break	Number of Time Series	Share (%)
No Break	529	95.32
Break	26	4.68
Total	555	100

Table 4: Time series structural breaks.

Structural Break	Raw	Log	Sqrt	Inv	Std	Total
No break	108	108	108	97	108	529
Break	3	3	3	14	3	26
Total	111	111	111	111	111	555

Table 5: Time series structural breaks, by transform.

5 Univariate time series: ARIMA

We now construct and estimate several models to predict values of gross value added in terms of both current prices and chained prices for each sector-region. In this section, we start by estimating univariate time series models. In particular, we estimate an Auto-Regressive Integrated Moving Average ($ARIMA$) model separately for each of the sector-region-transform time series. The goal is to predict the next value for gross value added for each sector-region, using information on past values of gross value added for that same sector-region. $ARIMA$ models are often used to model time series because they are parsimonious, have a good reputation for their predictive power, are often used as benchmark models, and only require time series of a single variable to estimate.

5.1 Setup

Specification An $ARIMA(p,d,q)$ model has three components. First, the auto-regressive (AR) component models the dependence of the current observation on past observations. It assumes that the value of the time series at a given point is linearly related to its previous values, and includes p lagged values of the dependent variable. Second, the integrated (I) component implies differencing up to order d to ensure a stationary time series. Third, the moving average (MA) part focuses on the relationship between the current error and the previous errors. It assumes that the error term at a given point is a linear combination of the error terms from the previous observations, and includes q lagged forecast errors. Formally, the $ARIMA(p,d,q)$ model is specified as

$$\underbrace{\Delta^d}_{I} Y_{irt} = c + \underbrace{\sum_{k=1}^{p} \phi_k \Delta^d Y_{irt-k}}_{AR} + \underbrace{\sum_{l=1}^{q} \theta_l \varepsilon_{irt-l}}_{MA} + \varepsilon_{irt} \tag{2}$$

where Y_{irt} is gross value added for sector i in region r in year t, c is a constant term (sometimes omitted in differenced models), Δ^d is the order d of differencing required to obtain a stationary time series of Y_{irt}, Y_{irt-k} are lagged values of the dependent variable up to lag p (the AR terms), ε_{irt-l} are lagged residuals up to lag q (the MA terms), and ε_{irt} is a white noise error term (the residual at time t). The estimated coefficients ϕ_k for $k = 1, ..., p$ model the auto-regressive component: how much do past values contribute to current values of Y_{irt}? Generally, in economic time series, estimated coefficients for ϕ_k are in the range $0 < \phi_k < 1$, i.e. positive, and not too large so to ensure a convergent series. Estimated coefficients θ_l for $l = 1, ..., q$ model the moving average component: the extent to which the forecast error at lag l (ε_{irt-l}), influences the current value of Y_{irt}. The total number of parameters is given by $p + d + q + 1$, where the 1 stands for estimating the variance of the residuals. If there is also a constant included, the number of parameters is $p + d + q + 2$. One more if we also include a trend. The degrees of freedom are $T - (p + d + q + 2)$. In our current setup, $T = 20$, so an $ARIMA(1,1,1)$ with a constant has $20 - 5 = 15$ degrees of freedom.

Assumptions The main assumptions for identification in the $ARIMA$ model are: (i) stationarity of the time series, (ii) linearity of the relationships, and (iii) white noise errors. First, stationarity means that statistical properties such as the mean, variance, and covariance, do not depend on the time at which the series is observed (i.e. that these remain constant over time). Formally, if Y_{irt} is a stationary time series, then for all s, the distribution of $(Y_{irt}, ..., Y_{irt+s})$ does not depend on t. Many statistical models require the series to be stationary to perform inference. Economic time series are

Hyper parameters		Estimation		Diagnostics	
max(*p*,*d*,*q*)	(5,2,5), grid search	**Stationarity**	KPSS	**White noise**	Ljung-Box
Convergence	LBFGS	**Parameter criterion**	BIC		
max iterations	50	**Final score**	BIC		

Table 6: ARIMA hyper parameters and settings.

often integrated of order 1, as they can include trends (e.g. inflation or economic growth). Several stationarity tests can be implemented, such as the KPSS, ADF or PP tests. Second, if the time series is non-linear, or if the components interact non-linearly (e.g. non-linear trends), variable transforms can be implemented. Third, white noise implies zero mean ($\mathbb{E}(\varepsilon_{irt}) = 0$), constant variance ($Var(\varepsilon_{irt}) = \sigma^2$), and no auto-correlation ($\text{Cov}(\varepsilon_{irt}, \varepsilon_{irt-k}) = 0$ for all k)). Standard tests include the Portmanteau or Ljung-Box test. Note that normality of the residuals is not required for white noise.

Strengths and limitations $ARIMA$ models offer several advantages for time series analysis. First, they are relatively simple and very parsimonious models to estimate, in the sense that $ARIMA$ relies only on a single time series to predict its own future values. This also implies that each sector-region can be estimated without constraints on its coefficients from jointly estimating multiple sector-regions in one model. Second, these models are quite flexible and can handle a wide range of time series patterns, including linear and non-linear trends (after transforms), seasonal patterns, and irregular fluctuations. Third, they have a legacy of being very good forecasters, particularly for short- to medium-term predictions. They are also often used as benchmark models against competing models. Finally, they allow for a clear interpretation of parameters. As with each model, there are also some limitations. First, we use fewer observations in the data (and have lower residual degrees of freedom) and we do not exploit potential explanatory variation from other covariates. Second, the performance can be dependent on selecting the appropriate values for p, d, and q. Inaccurate parameter selection can lead to poor forecasts. We therefore perform various model selection and validation tests to evaluate parameter fit and forecast errors. Finally, if the relationship remains non-linear after variable transforms, models like Long Short-Term Memory (LSTM) may be more suitable, albeit requiring significantly longer time series.

5.2 Estimation

We implement the $ARIMA(p, d, q)$ models using Python's autoARIMA package. This package searches automatically for the optimal order for an $ARIMA$ model and can be given a wide set of hyper parameters to fine-tune the models. Table 6 provides a summary of the hyper parameters and other settings for the implementation. We estimate eq(2) for each sector-region and for each of the transformations separately, for up to $37 \times 3 \times 5 = 555$ estimated models. We implement the estimation procedure following the classic Box-Jenkins methodology, which involves three main components: model identification, parameter estimation, and prediction. Table 7 provides a schematic overview of the $ARIMA$ implementation in the Python toolbox. We describe each step in detail below.

Steps to implement the ARIMA model.
1. Select a time series for a sector-region ir.
2. Choose a variable transform for Y_{irt}: levels, logs, standardized, inverse, square root.
3. Stationarity: Perform KPSS test. Difference and repeat until stationary, determining optimal d.
4. Estimation: Loop over p and q values for a given d. Select p and q given the lowest BIC.
5. Prediction: estimate with optimal parameters p, d, q and predict $\hat{Y}_{irt}$ for the transformed variable.
6. Diagnostics: perform post-estimation tests (residual analysis).
7. Obtain $\hat{Y}_{irt}$ in levels: reverse the transformation of $\hat{Y}_{irt}$.
8. In-sample goodness-of-fit: calculate NRMSE on the untransformed variable.
9. Out-of-sample performance: sliding window cross validation to calculate out-of-sample NRMSE for each fold.
10. Average out-of-sample NRMSE across folds for each model to obtain forecasting performance.
11. Repeat steps 1 to 10 and iterate over all sector-region-transforms.

Table 7: ARIMA steps.

Component 1: Model Identification. The first goals are to determine whether the time series are stationary and to identify appropriate values of p, d, and q. As a first step, a **stationarity test** is performed using the Kwiatkowski–Phillips–Schmidt–Shin (KPSS) test. In particular, the null hypothesis (H_0) states that an observable time series is stationary around a deterministic trend (i.e. trend-stationary) against the alternative (H_a) of a unit root.[15] If the test rejects the null at the 5% level for the values in levels, we difference the time series to make the time series stationary (i.e. we express eq(2) in differences $Y_{irt} - Y_{irt-1}$ instead of levels Y_{irt}). We run the KPSS test again on the differenced series. If we reject the H_0 again, we difference this series once more. We enforce a maximum of $d = 2$ to balance the degrees of freedom, overly complex models, and compute time. Users can choose different values of max d in the toolbox.

Once the time series has been made stationary, the next step is to **identify** p and q. In practice, we perform a full grid search to find optimal p and q, starting from a value of 0 for each up to some pre-deterimined order. We set the maxium values for p and q to 5. There are several iterative algorithms to **estimate** the model using Maximum Likelihood. The optimization algorithm we implement is LBFGS, for limited-memory Broyden-Fletcher-Goldfarb-Shanno. For each value of p and q, a Ljung-Box test is performed, which tests whether the residuals of the model exhibit significant autocorrelation at lags k

[15]We choose the KPSS test for stationarity since the Augmented Dickey-Fuller (ADF) test tends to have low power in small samples: with a short time series it can be difficult to reject the null even if the series is stationary, leading to excessively large values for d. The KPSS test is also the default option in the autoARIMA package. The ADF and/or Phillips-Perron (PP) tests can be chosen as alternative tests for the KPSS in the package. Please note that the null hypothesis of the KPSS test is the opposite of the classic ADF or PP tests: the null of the KPSS is stationarity, while the null in the ADF and PP tests is non-stationarity. One can also combine the ADF and KPSS tests to differentiate between trend-stationary and difference-stationary series. However, given the short time series and low power of the ADF, we prefer to only use the KPSS test. To streamline the process and avoid modeling trends individually, the toolbox detrends time series by applying first-differencing when required.

(H_0: there is no autocorrelation in the residuals up to lag k). If the test rejects the null, it suggests that additional terms (p or q) may be needed.

We estimate the model each time and record the BIC for each model. This criterion balances model complexity (number of parameters) with goodness-of-fit to prevent overfitting. The values for p and q that generate the lowest BIC over the grid search are selected to determine the final $ARIMA(p, d, q)$ model for a given sector-region time series and chosen transformation.[16]

Component 2: Parameter estimation. Once the optimal model structure is identified, the next step is to estimate the parameters of the ARIMA model: ϕ_k for $k = 1, ..., p$, ε_l for $l = 1, ..., q$, and σ^2.

Component 3: Prediction. Given the data and estimated model parameters, we next predict gross value added for each sector-region, $\hat{Y}_{irt}$. We do two things: (i) predict gross value added for all years in-sample, and (ii) forecast gross value added for the most recent year that is not yet available in the data. The first element provides in-sample goodness-of-fit. The second element is the main goal of the exercise. We also perform a post-estimation diagnostic test to ensure that the selected $ARIMA$ model adequately fits the data and that its assumptions are met. We implement another Ljung-Box test to see if predicted errors are white noise. Finally, we reverse the variable transformations to obtain predictions for gross value added in levels, and calculate the goodness-of-fit for each model on the untransformed data using RMSE and NRMSE. RMSE is expressed in units of the dependent variable. To compare across models, we also calculate the Normalized RMSE. We normalize based on the mean of the untransformed variables. All metrics are further described in Appendix D.

Component 4: Validation. Finally, we validate each model and its transforms using block cross validation (CV).[17] This step is implemented using the Sliding Window Forecast CV implementation in Python. See Figure 9 for a schematic overview. In particular, we split each time series into a training and a test set, and do this several times. We construct a training set of 10 years (from 2003 to 2012), and predict values for one year into the future (2013). We then construct a new training and test set by moving ahead one year, i.e. we train on the years 2004-2013 and test on the year 2014. We shift again one year and repeat until we reach prediction for 2022, totaling 10 sliding windows. For each of these training and test sets, we estimate the chosen $ARIMA(p, d, q)$ model on these 10 years, and predict 'pseudo out of sample' for the next year. Since we do have the realized data for these test years, we can evaluate how well the model actually performed on this unseen data. We evaluate performance by calculating the NRMSE of each run.[18] We do this for each window, and take the average selection metrics across the different runs for each sector-region-transform as our final validation metric. We can then use these average (N)RMSE values to compare model performance across transforms for a given sector-region model, as well as across models for a given sector-region.

[16] Alternative model selection criteria are the Corrected Akaike Information Criterion (AICc, with a correction for small samples), Hannan-Quinn Information Criterion (HQIC), or Out Of Bag (OOB).

[17] Standard cross validation techniques like K-fold or leave-one-out CV are not appropriate for time series models, as we cannot maintain the assumption of independent observations due to serial correlation. Also, we prefer block CV over rolling window CV, which can introduce leakage from future data to the model.

[18] The autoARIMA package only provides MSE and MAE. We calculate the NRMSE manually after inverting the variable transform again to levels if necessary so that the original mean (from the level variable) is used to normalize the values.

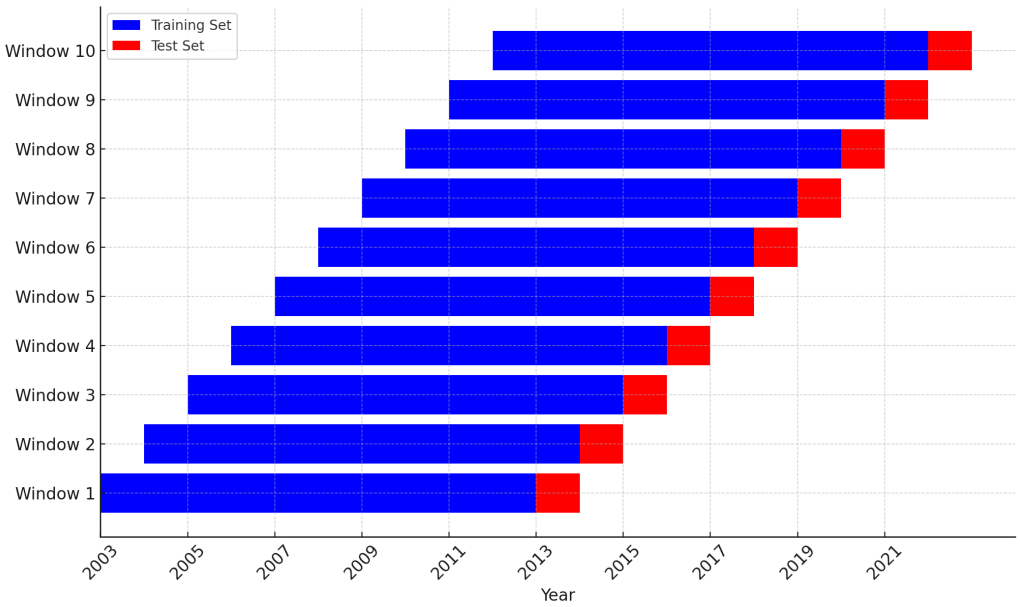


Figure 9: Sliding window cross validation.

5.3 Results

Results for the ARIMA models are available in the toolbox folder `/task5_arima/output`. There are two sub-folders: `/gva_currentprices` and `/gva_chainedprices`, each with their own full set of results. Each of these contain three subfolders: `/csv`, with model outputs in `arima_predictions.csv`, `arima_validation.csv` and `arima_parameters.csv`, as well as `/plots` with a series of plots with actual, fitted, and forecast values for each sector-region-transformation, together with other plots to further describe the model outputs. The third folder `/tex` contains the LaTeX version of the tables in this report. The graphs included in the report are part of `/plots`. For this report, throughout we discuss the results for the models in `/gva_currentprices`, but similar analysis can be done for gross value added in terms of chained prices.

The `arima_predictions.csv` file contains the following variables: `time`, `sector`, `region`, `sector_region`, `transform`, `gva_currentprices`, `gva_currentprices_pred`, `p`, `d`, `q`, `rmse`, `nrmse`, and `ljung-Box`. It contains the time series of gross value added for each sector-region-transform, as well as predicted values for each year: both in-sample predictions and out-of-sample forecasts. For each model, we report the optimal chosen parameters p, d, q, along with the in-sample goodness-of-fit measures RMSE and NRMSE values of the estimated models. In terms of diagnostics, the file also reports whether the residuals pass the white noise test using the Ljung-Box test (FALSE/TRUE, with TRUE indicating a p-value ≥ 0.05 for the test statistic under H_0: the residuals are white noise).

Figure 10 shows the distribution of the number of models per parameter combination. We have 551 estimated ARIMA models out of 555 possible sector-region-transform combinations. Non-estimated models are inverse, log, and square-root transforms for Brussels sector *Manufacture of coke and refined petroleum products (CD)*, and inverse transform for Brussels sector *Wholesale and retail trade, repair of motor vehicles and motorcycles (GG)*.[19] Out of these, the most common occurrence is of the form

[19]This is due to one or more values for gross value added being ≤ 0 over the time series for these sector-regions. We do

$ARIMA(0,1,0)$ (257 models), followed by $ARIMA(1,0,1)$ (63 models), $ARIMA(0,1,1)$ (46 models), and $ARIMA(1,0,0)$ (46 models). Together, these account for close to 75% of all estimated models.[20]

We also report the distribution of the in-sample prediction NRMSE in Figure 11. The median NRMSE is 8%, indicating that the model's prediction error is relatively small compared to the scale of the data: the median prediction errors are 8% of the mean gross value added value. The mean NRMSE is 14%. The Ljung-Box tests show that 5 out of 551 estimated models, or 0.9% of models, do not pass the white noise test for the residuals, much below the expected number of 27.5 of models to fail at a 5% significance level.[21] This can imply either overfitting of the model, or the residuals truly being white noise.

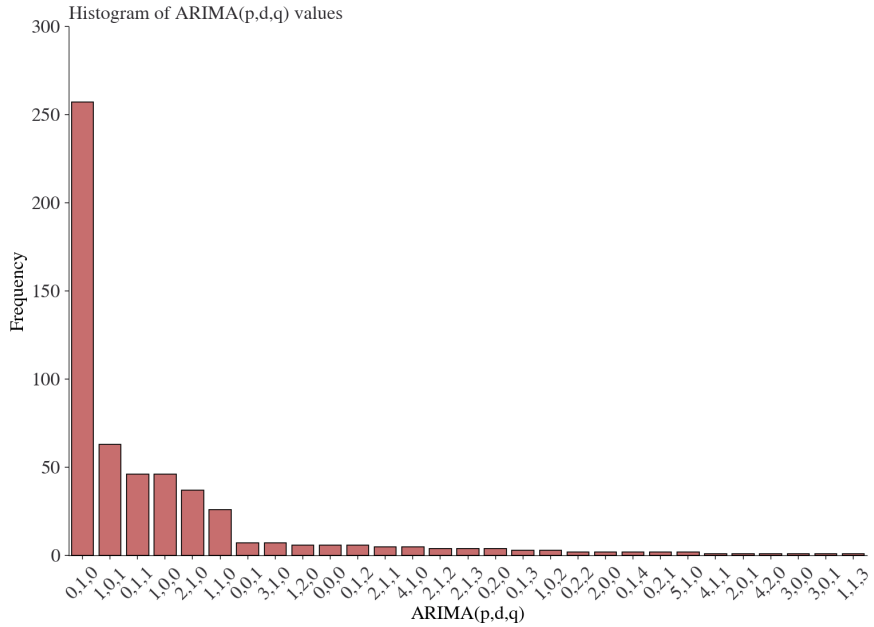


Figure 10: Distribution of estimated ARIMA model parameters.

Next, the `arima_validation.csv` file contains the pseudo-out-of-sample RMSE and NRMSE values for each sector-region-transformation, computed as the simple average of the metrics obtained from the sliding window cross-validation. The forecast value for the next year from the `arima_predictions.csv` file will be used as input to the ensemble prediction in Section 10 to generate the ensemble forecast: a weighted average forecast value for each sector-region, with weights constructed from the validation NRMSE. Better forecast models (lower NRMSE) will receive a higher weight in the ensemble prediction.

Table 8 reports the distribution of the average NRMSE from the validation stage across sliding

obtain results for raw and standardized transforms for Brussels CD, as well as the other four transforms for Brussels GG.

[20] Note that ARIMA models with few parameters are the outcome of the optimization routine that looks for the best ARIMA order using the BIC over the entire grid search for parameter values up to (5,2,5). The BIC penalizes complex models, specifically to avoid overfitting in the training stage that could otherwise lead to bad performance in the validation stage. Models with few parameters are thus optimally chosen to balance complexity with parsimony and overfitting.

[21] We expect 5% of tests to fail by chance even if the null hypothesis is true. With 551 estimated models, approximately 28 models might fail purely due to random variation, even if the ground truth is that all models pass the test. I.e. these are false positives (Type I errors). Out of the 5 models that fail the white noise test, only one has an in-sample NRMSE higher than 10%. Four of them have a validation NRMSE above 10%. Higher validation NRMSE values will imply lower weights of that model in the ensemble construction.

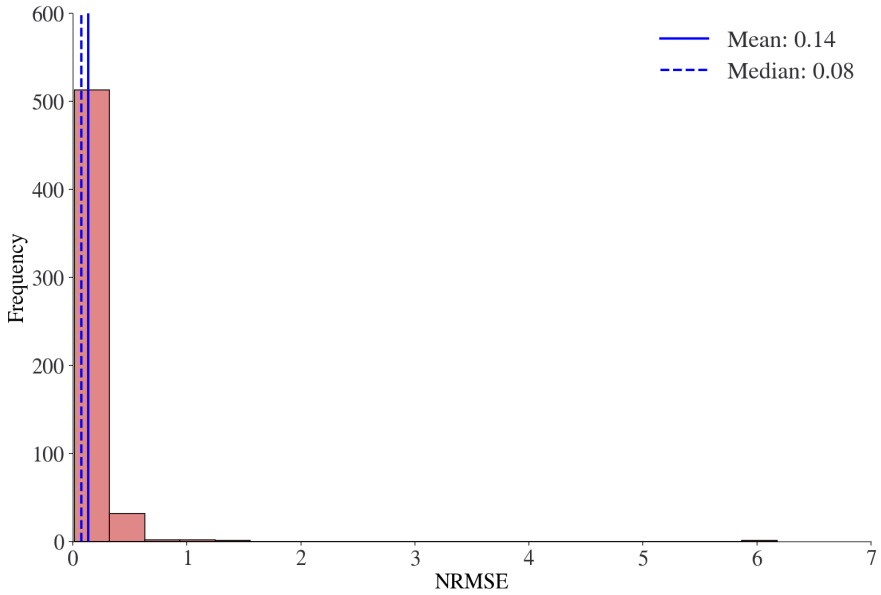


Figure 11: Distribution of in-sample NRMSE for ARIMA models.

window folds, for each variable transform.[22] All models except the inverse transform perform quite well in terms of out-of-sample validation: the median NRMSE is around 6%.[23] The inverse transform however, performs worse at the median, and also contains one or more outliers. Models with a very low NRMSE (e.g. around 2% for the best models) will ultimately obtain a large weight in the final prediction, while those with a very high NRMSE will get a weight close to zero.

Which sector-region-transforms perform best and worst? Table 9 shows the best 5 and worst 5 performing sector-region-transforms. The top performers do really well, with a validation NRMSE of around 2%, implying that the average error of the forecast is around 2% of the mean of the time series. These top performers include a mix of variable transformations and sectors. The worst performers for the *ARIMA* models are all inverse transforms with a bad forecasting performance.

Finally, the sub-folder `/plots` contains time series graphs for all sector-regions and each estimated transformation, as well as the summary statistics of the model outcomes presented above. For example, Figure 12 shows for each region the results for the sectors in our running example. In the figure, we select the best performing transform based on its validation (out-of-sample) NRMSE.

Some final remarks. First, visually, the models seem to perform relatively well in-sample, as actual and fitted values follow each other closely, including capturing both trends and changes in these trends. Yet, from a statistical perspective, the in-sample (N)RMSE is the relevant statistic to capture the deviations between the actual and fitted values and the overall goodness of fit. Second, these

[22] Notice that we retain 547 models for the validation stage out of 551 from the prediction stage. The Python package generates a "LU decomposition error" for CK and LL in Flanders, CJ in Brussels, and II in Wallonia. All errors appear in raw transformation except for LL in Flanders which is for the inverse transformation. There are several potential reasons for this to occur, including singular matrix and numerical instability.

[23] As a rule of thumb, NRMSE values below 10% are often considered as good for time series models. However, preferred values can vary across applications. For example, for stable time series like power consumption, even values below 10% might be preferred. Conversely, for volatile time series like weather prediction or stock markets, a higher value might be more realistic. The best performing models are around 2%, which is excellent. These models will get a larger weight at the ensemble stage of the toolbox.

Transform	Obs	Mean	SD	Min	p50	Max
Raw	108	0.100	0.133	0.018	0.064	1.221
Log	110	0.105	0.128	0.017	0.065	0.986
Square Root	110	0.091	0.079	0.017	0.065	0.446
Inverse	108	14,994.074	93,282.647	0.026	0.088	919,467.325
Standardized	111	0.097	0.119	0.015	0.064	1.082

Table 8: Summary statistics of validation NRMSE by transform type for ARIMA models.

Sector	Region	Transformation	NRMSE	Rank
Best 5				
PP	Flanders	standardized	0.0150	1
PP	Brussels	square root	0.0170	2
PP	Brussels	log	0.0172	3
PP	Flanders	raw	0.0176	4
PP	Flanders	square root	0.0180	5
Worst 5				
DD	Brussels	inverse	85,525.28	543
AA	Flanders	inverse	92,676.19	544
CG	Wallonia	inverse	182,180.00	545
CH	Wallonia	inverse	232,154.70	546
CH	Flanders	inverse	919,467.30	547

Table 9: Best 5 and worst 5 sector-region-transforms by validation NRMSE for ARIMA models.

results also show the added value of allowing for multiple transformations of each variable to forecast gross value added. The best in-sample performance for a given time series can vary substantially across time series. In this case, each sector has a different transform that performs best out of sample: standardized for sector KK in Brussels, raw for sector GG in Flanders, and square root for sector LL in Wallonia.

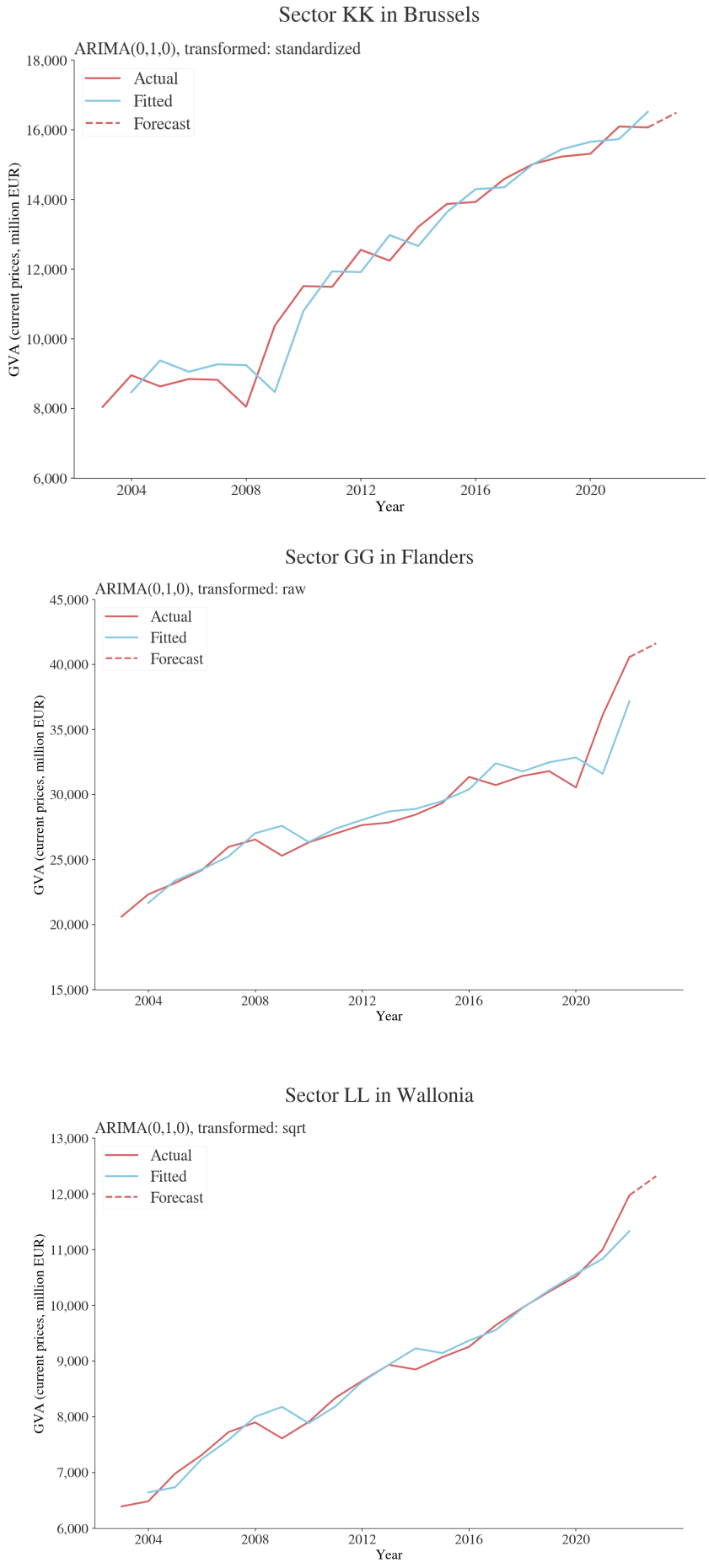


Figure 12: Actual, fitted, and forecast values for selected sector-regions in the ARIMA model.

6 Multivariate time series: VAR and VEC models

The next set of models we construct and estimate are multivariate time series models. In particular, we estimate Vector Auto-Regression (VAR) and Vector Error Correction (VEC) models. *VAR* and *VEC* models are suitable if we suspect interdependencies between the to be predicted variables. The goal is to predict the next value for gross value added for each sector-region, using information on both its own past values, as well as the past values of all other variables in the system. *VAR* models require stationarity of the time series in the system (potentially after differencing), while *VEC* models allow for cointegrated time series, i.e. series that move together over time in a way that preserves a long-run equilibrium.

6.1 Setup

Specification A $VAR(p)$ model of order p is a system of equations where each equation explains the dynamics of a variable based on its own lags, as well as the lags of other variables in the system, up to some lag p. A *VAR* model then models both (i) the dynamics of each variable over time, and (ii) the short-run relationship between those variables. Formally, a $VAR(p)$ model for L variables and p lags is specified as:

$$\mathbf{Y}_t = \mathbf{c} + \Phi_1 \mathbf{Y}_{t-1} + \Phi_2 \mathbf{Y}_{t-2} + \cdots + \Phi_p \mathbf{Y}_{t-p} + \varepsilon_t \tag{3}$$

where $\mathbf{Y}_t = [Y_{1t}, ..., Y_{irt}, ..., Y_{Lt}]'$ is a $L \times 1$ vector of variables at time t, $\mathbf{c}$ is a $L \times 1$ vector of constants, Φ_k is a $L \times L$ matrix of coefficients with elements ϕ_{ij} for each lag $k \leq p$, and $\varepsilon_t = [\varepsilon_{1t}, ..., \varepsilon_{Lt}]'$ is a $L \times 1$ vector of white noise error terms. Similar to the *ARIMA* model, the ϕ_{ij} coefficients reflect the auto-regressive component of the time series, in this case both on the own time series of Y_{irt}, as well as on the time series Y_{jst}, and both contemporaneously at time t as well as over time $t-k$ for $k = 1, ..., p$. For the series to converge, we expect $\phi_{ij} \in (-1, 1)$ for all i, j. It is possible that some time series co-move positively or negatively. The total number of parameters to be estimated is $L \times (p \times L + 1) = L^2 \times p + L$: there are $L \times L$ parameters to be estimated per lag plus one intercept (if included), and this for p lags. The number of parameters thus grows quadratically with L and linearly with p. The residual degrees of freedom are given by $T - p - (L^2 \times p + L) = T - pL^2 - L$, where $T - p$ is the number of effective observations for the time series T after subtracting p observations used for initialization of the model.

Consider the following example, for $L = 2$ variables (Y_{1t} and Y_{2t}) and one lag ($p = 1$). Suppose we consider the relationship between interest rates and GDP growth in a given region. Today's interest rate depends on past values of interest rates and past GDP growth, while today's GDP growth depends on past values of GDP growth and past interest rates. A $VAR(1)$ model is then defined as:

$$\begin{bmatrix} Y_{1t} \\ Y_{2t} \end{bmatrix} = \begin{bmatrix} c_1 \\ c_2 \end{bmatrix} + \begin{bmatrix} \phi_{11} & \phi_{12} \\ \phi_{21} & \phi_{22} \end{bmatrix} \begin{bmatrix} Y_{1,t-1} \\ Y_{2,t-1} \end{bmatrix} + \begin{bmatrix} \varepsilon_{1t} \\ \varepsilon_{2t} \end{bmatrix} \quad \text{and} \quad \text{Cov}(\varepsilon_{1t}, \varepsilon_{2s}) = \begin{cases} \sigma_{1,2} & \text{if } t = s, \\ 0 & \text{if } t \neq s. \end{cases}$$

where coefficients ϕ_{ij} indicate the effect of the j-th variable on the i-th variable, and ε_{1t} and ε_{2t} are error terms for Y_{1t} and Y_{2t}, respectively. Hence, current values of Y_{1t} can be affected through both its own lags Y_{1t-1} via ϕ_{11}, as well as through the lags of the other variable Y_{2t-1} via ϕ_{12}. Similarly for Y_{2t}, which can be affected by its own lags via ϕ_{22}, as well as by lags of the other variable via ϕ_{21}. More generally,

there is one matrix Φ_k for each included lag $k = 1, ..., p$ in the model. The number of parameters to estimate is $L^2 \times p + L = 4 \times 1 + 2 = 6$. The degrees of freedom is given by $T - p - 6 = 20 - 1 - 6 = 13$.

A VAR model requires that all time series $\mathbf{Y}_t = [Y_{1t}, ..., Y_{irt}, ..., Y_{Lt}]'$ are stationary, possibly after differencing. If one or more time series are not stationary but they are cointegrated, a VAR model does not capture such long-term relationships.[24] Instead, a $VEC(p, r)$ model is required to account for these relationships through the error correction term, which represents how the series adjust toward equilibrium in the long run. Parameter p captures the lags as in a VAR, while r captures the number of cointegrating equations. The VEC model is fit to the first differences of the non-stationary variables, and a lagged error-correction term is added to the relationship. In particular:

$$\Delta \mathbf{Y}_t = \mathbf{c} + \Pi \mathbf{Y}_{t-1} + \sum_{k=1}^{p-1} \Gamma_k \Delta \mathbf{Y}_{t-k} + \varepsilon_t \tag{4}$$

where $\mathbf{c}$ is a time series-specific deterministic trend or mean, $\Pi = \sum_{k=1}^{p} \Phi_k - I$ is a matrix capturing long-run (i.e. cointegration) relationships, and $\Gamma_k = -\sum_{j=k+1}^{p} \Phi_j$ is a matrix capturing short-run dynamics deviating from this long-run relationship. The $p - 1$ in the summation reflects the fact that the first difference formulation reduces the effective lag order of the short-run dynamics by 1. If the variables are cointegrated, the matrix Π has reduced rank $r < L$. It can then be decomposed as:

$$\Pi = \alpha \beta^\top$$

where α is a $L \times r$ loading matrix of adjustment coefficients and $\beta^\top$ is a $r \times L$ cointegration matrix of cointegrating relationships. If no cointegration is detected, the rank of the cointegration matrix is zero, meaning there are no long-term equilibrium relationships. For a VEC model, the number of parameters amounts to $L^2 \times (p-1) + L + (r \times L + L \times r)$, where $L^2 \times (p-1)$ is the total number of parameters for the short-run dynamics in Γ, L for the intercepts, $r \times L$ reflects the number of parameters in the cointegrating vectors β, and $L \times r$ those of the adjustment vectors α. The degrees of freedom are $T - p - [L^2 \times (p-1) + L + (r \times L + L \times r)]$. A $VEC(3, 2)$ for example, then has $9 \times 1 + 3 + (3 \times 2 + 2 \times 3) = 24$ parameters to estimate. The residual degrees of freedom are then 34. Finally, a VEC model with rank zero reduces to a VAR model on the first-differenced data. Essentially, there is then no long-run equilibrium to account for. Table 10 provides a schematic overview of the data generating process and which model to estimate.

Stationarity	**No Cointegration**	**Cointegration**
Stationary Data	VAR	Not Applicable
Non-Stationary Data	VAR in differences	VEC

Table 10: Classification of VAR and VEC models.

Assumptions The main assumptions for identification in the VAR model are: (i) stationarity of the time series, (ii) linearity of relationships between variables and their lags, and (iii) white noise errors.

[24] Cointegration implies that while individual time series might be non-stationary, a linear combination of them is stationary.

First, estimation of the parameters of the VAR requires that the variables in $\mathbf{Y}_t$ are covariance stationary. If the variables are not covariance stationary but their first differences are, they may be modeled using a VEC model if they are also cointegrated. Otherwise, we can estimate a VAR model in first differences. Second, the relationship between the variables is assumed to be linear. The model expresses each variable as a linear function of its own lags, the lags of other variables, and a stochastic error term. Third, the errors are assumed to be white noise, implying: zero mean ($\mathbb{E}[\varepsilon_t] = 0$), constant variance ($\mathbb{E}[\varepsilon_t \varepsilon_t'] = \Sigma_\varepsilon$), no auto-correlation ($\mathbb{E}[\varepsilon_t \varepsilon_k'] = 0$ for $t \neq k$), and no cross-correlation ($\mathbb{E}[\varepsilon_t \mathbf{Y}_{t-k}^\top] = 0$ for all $k > 0$).

Strengths and limitations $VAR(p)$ models have a few advantages compared to univariate time series analysis like $ARIMA$. First, they capture dynamic interactions by simultaneously modeling multiple time series and their interdependencies in a natural way, which can be useful for understanding how shocks comove and propagate through a system. Second, they are easy to estimate using OLS. Third, they perform well for forecasting systems with possibly correlated variables. Fourth, they allow for a simple way of introducing endogenous time series without the need to specify which series are endogenous or exogenous. As always, there are also some weaknesses. Most importantly, VAR and VEC models involve estimating more parameters due to multiple variables, lags, and interactions. A sufficiently large sample size is required. We currently have 20 observations per sector-region. For 37 sectors $\times$ 3 regions = 111 series, estimating a full VAR/VEC for all sectors and regions together is computationally intensive if at all possible.[25] We will introduce cross-series spillovers in an alternative way in the spatial models in Section 8. Second, there is the risk of overfitting if too many lags are added, especially with limited data. If the interdependencies are weak or if there is no evidence of cointegration, simpler approaches like univariate $ARIMA$ or panel data methods might be more robust and parsimonious.[26] Finally, it assumes that variables affect each other symmetrically and doesn't model causal relationships directly.[27]

6.2 Estimation

We implement the VAR(p) and $VEC(p, r)$ models using Python's Vector Autoregressions package from the Statsmodels module. This package allows to automatically identify and select a VAR or VEC model to estimate, have the model select an optimal lag order p and cointegration rank r based on some user-defined information criterion, predict the model, perform forecasting, and implement post-estimation diagnostics. Table 11 provides a summary of the hyper parameters and other settings for the implementation. To balance the number of time series to be jointly estimated with the constraints of the short time series and resulting degrees of freedom in the models, we choose to estimate eq(3) or eq(4) jointly for each sector across the three regions, i.e. we estimate 37 models, one for each sector,

[25] Formally, we require small N, large T. To estimate a VAR model, a rule of thumb is $T > 10 \times k$, where k is the number of parameters to estimate. In our setup, this is simply impossible, even in the very long: the number of parameters to estimate is in the order of $L^2 = 12,321$ for $p = 1$ and no cointegrating relationships, implying one should wait until roughly the year 125,000 to run a simple $VAR(1)$ specification.

[26] A valid compromise is the use of panel VAR models, that combine interdependence of time series with a much lower number of parameters to estimate from its panel structure. However, up to date, their implementation for estimation and validation are not fully developed yet in Python.

[27] For causal inference, structural VARs are often used. However, they require a stronger stance on the underlying theory, each restriction is another degree of freedom lost, and for the purpose of prediction, the focus is less on this component of the model evaluation.

Hyper parameters		Estimation		Diagnostics	
max(*d*)	2	**Stationarity**	KPSS	**White noise**	Ljung-Box
max(*p,r*)	Free	**Parameter selection**	BIC	**Normality**	Jarque-Bera
		Cointegration	Johansen (1995)		

Table 11: VAR/VEC hyper parameters and settings.

each with 3 time series for the three regions. We do this for each variable transform.[28] Similarly, the VAR and VEC models allow in principle for additional exogenous variables to be included in the model. However, this again increases the number of parameters to estimate while further reducing the degrees of freedom of the model beyond possibility of parameter estimation. We therefore mention this option in the report but do not include them in the current setup. We implement the estimation procedure similar to the $ARIMA$ procedure in Section 5. Table 12 provides a schematic overview of the VAR/VEC implementation in the Python toolbox.

Component 1: Model Identification. Like in the $ARIMA$ models, the first goal is to determine stationarity of the time series. We implement the KPSS test for each model (three regions for the same sector), by performing the test on each of the three time series separately, with a maximum order of differencing set at 2.

Component 2: Parameter estimation. If all time series are stationary in levels or after differencing, we estimate a VAR model on the stationary series. If the time series are non-stationary but cointegrated, we estimate a VEC model instead.[29] For the VEC model, the number of cointegrating equations r is determined using the Johansen (1995) test. In each case, the optimal number of lags is determined by choosing a selection criterion. We choose the BIC. In particular, for a given lag p, the test compares a VAR model with p lags with one with $p-1$ lags. The null hypothesis is that all the coefficients on the pth lags of the endogenous variables are zero. Starting from the most number of lags and going up, the first test that rejects the null hypothesis is the lag order selected by this process. See e.g. Lütkepohl (2005) for more details on this procedure. Introducing too many lags wastes degrees of freedom, while too few lags leave the equations potentially mis-specified and are likely to cause autocorrelation in the residuals.

[28] Alternative groupings might be to estimate all sectors for a given region, or to aggregate particular sectors. However, even this simple setup with three time series requires at least $3^2 \times 1 + 3 = 12$ parameters to estimate on 60 data points for the simplest $VAR(1)$ specification. A $VEC(3,2)$ requires 33 parameters to estimate. Moreover, the additional number of parameters imposes further restrictions on the windows for validation, reducing the number of folds to evaluate out-of-sample performance of the model.

[29] The Vector Autoregressions package selects a VAR model only if all regional time series are stationary without differencing. It runs a VEC model for optimal $d > 0$. However, a VEC model for which the cointegrating equations are zero is the same as a VAR model with optimal $d > 0$. In other words, a $VEC(2,0)$, that is a VEC that is not co-integrated, coincides with a $VAR(2)$. A VAR model with $p = 0$ (no lagged structure) and no cointegration, is specified as a $VEC(0,0)$. A series that is non-stationary and co-integrated runs as a $VEC(p,r)$ model. If the model fails to select any cointegration rank, it will result in an error. The model cannot be estimated and it will skip that sector-transform and move to the next one. This is most probably due to linear algebra: non-convergence of the cointegration equation that captures the long-term trends (i.e. the LT equation would not achieve equilibrium, which is operationalized by solving a system of equations).

Steps to implement the VAR/VEC model.

1. Select L time series for joint estimation: $Y_{ir,1}, Y_{ir,2}, ..., Y_{ir,L}$.
2. Choose a variable transform for Y_{irt}: levels, logs, standardized, inverse, square root.
3. Stationarity: perform KPSS test. Difference and repeat until stationary, determining optimal d.
4. Cointegration test: Johansen (1995) test to evaluate if variables are cointegrated of order r.
5. Estimation: if $r > 0$, estimate a VEC model. Otherwise estimate a VAR model. Select p given the lowest BIC.
6. Prediction: estimate with optimal parameters p, r and predict $\hat{Y}_{irt}$ for each of the L transformed variables.
7. Diagnostics: perform post-estimation tests (white noise, normality).
8. Obtain $\hat{Y}_{irt}$ in levels: reverse the transformation of $\hat{Y}_{irt}$.
9. In-sample goodness-of-fit: calculate NRMSE on the untransformed variable.
10. Out-of-sample performance: sliding window cross validation to calculate out-of-sample NRMSE for each fold.
11. Average out-of-sample NRMSE across folds for each model to obtain forecasting performance.
12. Repeat steps 1 to 11 and iterate over all sector-region-transforms.

Table 12: VAR/VEC steps.

Component 3: Prediction. We then estimate a VAR or VEC model with the optimal number of lags for both VAR/VEC and cointegration terms for VEC. We fit the VAR models using OLS and the VEC models using Maximum Likelihood. We also perform post-estimation diagnostics to validate the model assumptions. The first diagnostic is a Ljung-Box white noise test that checks for multiple lags. The null hypothesis (H_0) states that residuals are indeed white noise up to lag p. The second diagnostic is the Jarque-Bera test for the normality of the residuals, with the null hypothesis (H_0) stating that the error terms are normally distributed. Again, we reverse the transformations if applied to obtain gross value added in levels, and calculate the goodness-of-fit for each model on the untransformed data using RMSE and NRMSE. While the model is estimated jointly on three time series each time and produces a model (N)RMSE, we evaluate the (N)RMSE on each sector-region-transform separately for comparison across models. I.e. we calculate the RMSE for each individual time series as the difference between the data and the predicted values for that sector-region, even if the model jointly predicts several sector-regions, and normalize the RMSE by the average value for each sector region to obtain the NRMSE.

Component 4: Validation. Finally, we validate the performance of the model out of sample. We follow a similar procedure as in the $ARIMA$ models, using sliding window cross-validation. One important remark: since we need to estimate more parameters, the minimal time series to estimate the models are longer than the 10 years we used in the $ARIMA$ setup. We set the windows to 12 years, which seems to enable validation for all models. This also implies that the number of folds drops from ten to eight, inducing averaging over fewer out-of-sample predictions. Users are free to change the window size if more years become available.

6.3 Results

Results for the VAR/VEC models are available in the toolbox folder `/task6_var_vec/output`. Similar to the $ARIMA$ output, the toolbox generates again a full set for both `/gva_currentprices` and `/gva_chainedprieces`, each with sub-folders `/csv`, `/plots`, and `/tex`.

The `var_vec_predictions.csv` file contains the following variables: `time`, `model`, `sector`, `transform`, `lag_order`, `coint_rank`, `whiteness_test`, `jb_test`, `region`, `gva_currentprices`, `gva_currentprices_pred`, `rmse`, and `nrmse`. For each estimated model, it contains the time series of gross value added for each sector-region, as well as predicted values for each year (in-sample) and the forecast for the next missing year (out-of-sample). Again, up to five transformations of gross value added per sector-region are evaluated and predicted. For each model, the optimal chosen parameters are reported. For VAR models, it reports the optimal number of lags p. For VEC models, it reports the optimal number of lags p, as well as the number of cointegrating equations or rank r. Models are jointly estimated for each sector across the three regions, with sector-region specific predictions, as well as their (N)RMSE values. These sector-region specific values allow for comparison across models for a given sector-region at the ensemble stage. Post estimation results include the outcomes of the Ljung-Box test for white noise (with a boolean variable equal to "TRUE" if model residuals are white noise, i.e. the absence of autocorrelation), and for normality of the error terms following a Jarque-Bera test.

Figure 13 shows the distribution of the optimal model parameters across estimated VAR/VEC

models. There are 182 estimated models out of 185 potential models (37 sectors times 5 transforms). Sector *Manufacture of coke and refined petroleum products (CD)* is not estimated for the log, inverse, and square root transforms because of zero or negative values in the time series. We do have predictions for the raw and standardized transforms for this sector. Out of the estimated models, 175 are estimated as VEC models and 7 as VAR, respectively. The VAR models are all of the form $VAR(1)$. A $VEC(2,0)$ coincides with $VAR(2)$ model in differences without any cointegrating equation. For 160 out of 175 VEC models, the optimal p is given by 3. 85 models are estimated as a $VEC(3,3)$, 48 as $VEC(3,2)$, and 27 as $VEC(3,1)$.

We also report the distribution of the in-sample prediction NRMSE for each sector-region-transform in Figure 14. The median NRMSE is 3%, while the average is now at 6%. Both the mean and median NRMSE are less than half of those for the $ARIMA$ models, signaling a better fit of these models in-sample on average. Out of 182 models for which we have predictions, 67 models (37%) pass the white noise error tests, while 158 models or 87% pass the Jarque-Bera test.

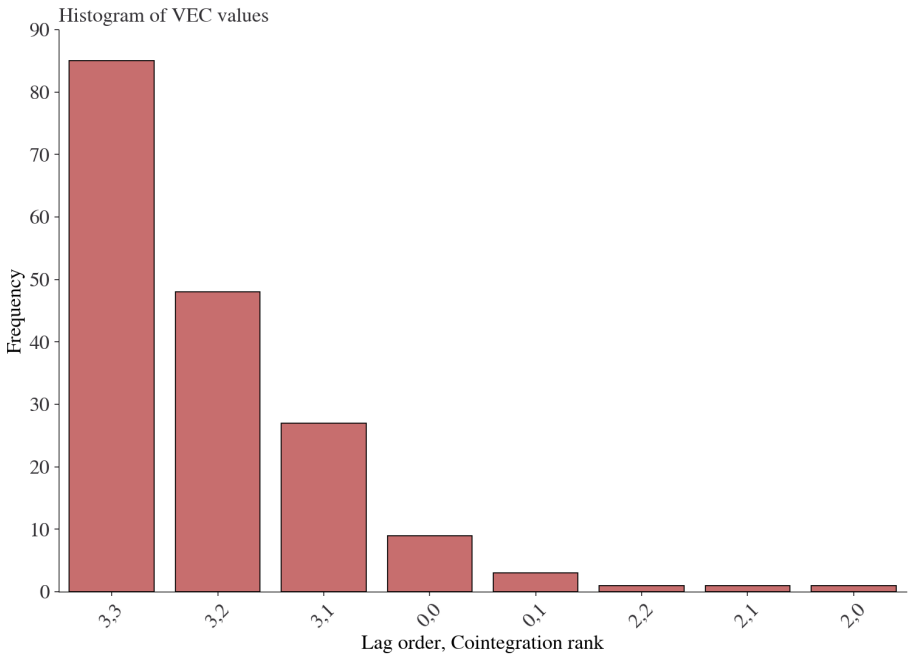


Figure 13: Distribution of estimated VEC model parameters.

Next, the `var_vec_validation.csv` file contains the pseudo-out-of-sample RMSE and NRMSE values for each cross-validation per sector-region-transform, as well as the start and end years of the training and test sets. Importantly, as we estimate more parameters than in the $ARIMA$ models, we must increase the minimal window size to 12 years instead of 10 years. This leaves us with eight cross validation folds to average across. The number of folds will increase by one for each year of additional data that will arrive in the future.

Table 13 reports the distribution of the average NRMSE from the validation stage across cross validation folds, for each sector-region-transform. This time, also the inverse transform performs within the same ball park as the other transforms. The median NRMSE is around 12%, while the mean is between 16.7% and 28.9%.[30]

[30]One additional remark. It turns out that the distribution of the validation NRMSE is identical for the raw and

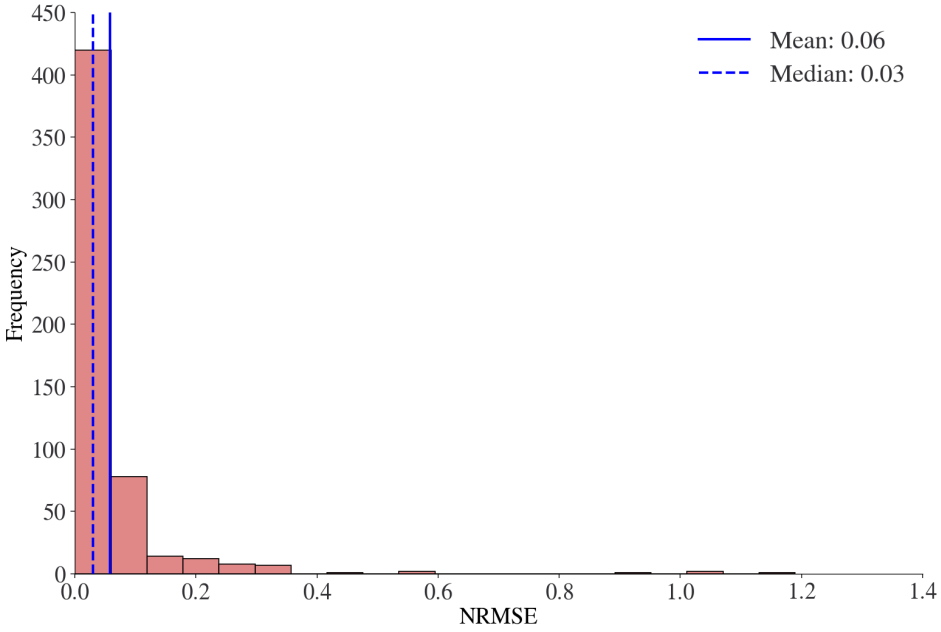


Figure 14: Distribution of in-sample NRMSE for VAR/VEC models.

Table 14 reports again the best and worst 5 sector-region-transforms in terms of out-of-sample validation NRMSE. Interestingly, the top performers all relate to the sector *Education (PP)* within Wallonia across transforms. Worst performers include sector *Scientific research and development (MB)*, as well as *Agriculture (AA)* and *Mining and quarrying (BB)* across the three regions and different transforms.

Transform	Obs	Mean	SD	Min	p50	Max
Raw	111	0.196	0.219	0.019	0.120	1.601
Log	108	0.173	0.200	0.020	0.113	1.607
Square Root	108	0.167	0.166	0.018	0.117	1.140
Inverse	108	0.289	0.677	0.017	0.115	5.476
Standardized	111	0.196	0.219	0.019	0.120	1.601

Table 13: Summary statistics of validation NRMSE by transform type for VAR/VEC models.

Finally, the sub-folder `/plots` time series graphs for all sector-regions and each estimated transformation. We continue our running example of the largest sectors for each region. We have now estimated these sectors jointly across the three regions, and show the results in Figure 15. Again,

standardized transforms in the table. This is possible in both VAR/VEC and $ARIMA$ models due to their invariance under linear transformations. In particular, standardization rescales the data but does not alter the underlying time series dynamics, such as trends, seasonality, or autocorrelation structures. The predictions from these models are made using the fitted coefficients and lagged values. If one fits the model on standardized data, the coefficients will adapt to the scale of the standardized series, but the structure of the predictions (when scaled back to the original units) will match those from the raw data. Hence, as long as the tests on raw and standardized data yield the same optimal parameters (this is not always the case) then we get the same inverted predictions.

Sector	Region	Transformation	NRMSE	Rank
Best 5				
PP	Wallonia	inverse	0.0166	1
PP	Wallonia	sqrt	0.0180	2
PP	Wallonia	standardized	0.0186	3
PP	Wallonia	raw	0.0186	4
PP	Wallonia	log	0.0196	5
Worst 5				
BB	Flanders	log	1.6069	542
II	Wallonia	inverse	2.2839	543
BB	Brussels	inverse	2.8021	544
MB	Wallonia	inverse	3.0026	545
AA	Brussels	inverse	5.4763	546

Table 14: Top 5 and Bottom 5 sectors by validation NRMSE.

we can see that there is variation in the optimal transform chosen for prediction, including log and inverse transforms. There is also variation in the number of optimal model parameters. While the models are estimated jointly across regions, there is substantial variation in the predicted time series for each individual region. Yet, because sectors might co-move across regions due to sectoral or macro-economic conditions, their cointegration can be important to model.

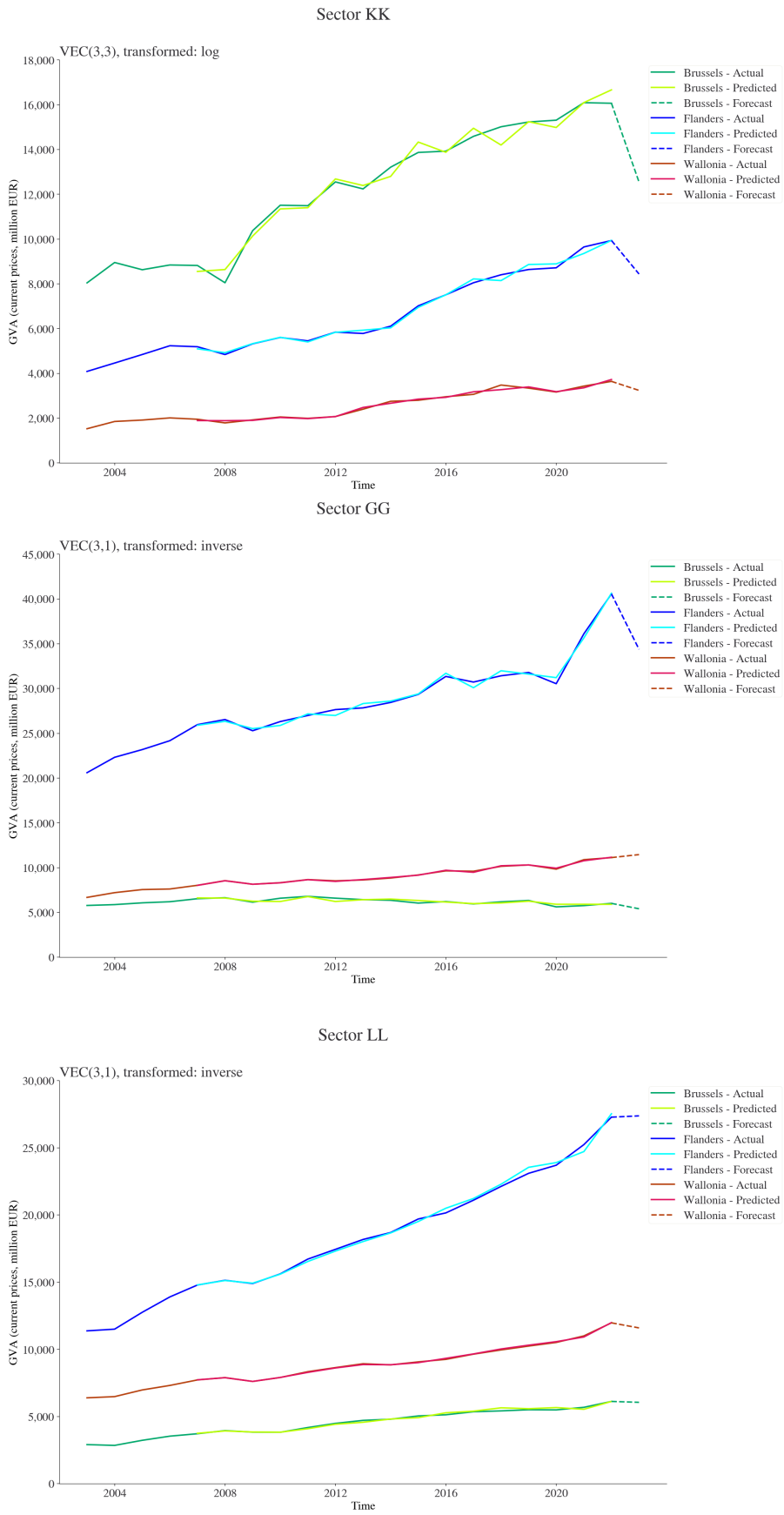


Figure 15: Actual, fitted, and forecast values for selected sectors in the VAR/VEC model.

7 Panel data: linear fixed effects models

Next, we estimate static linear panel data models. Fixed effects (FE) models are often used to analyze data that vary across both time and cross-sectional units, such as sectors and regions in our context. The goal is to predict the next value for gross value added for each sector-region, using information on both its own past values, as well as the values of all sector-regions, possibly together with additional covariates to control for observable characteristics. Fixed effects additionally help control for unobserved heterogeneity. The intuition is to exploit information on panel observations (sector-regions over time), while controlling for unobserved heterogeneity that is constant within groups (using fixed effects).

7.1 Setup

Specification The general formulation of the fixed effects model is

$$Y_{it} = \mathbf{X}_{it}\beta + \alpha_i + \varepsilon_{it} \tag{5}$$

where $\mathbf{X}_{it}$ is a vector of time-varying covariates, β is a vector of parameters to be estimated, α_i is the unobserved time-invariant individual effect, and ε_{it} is an error term, for $i = 1, ..., N$ and $t = 1, ..., T$. Generally, $N >> T$ in panel data settings. While $\mathbf{X}_{it}$ is observed in the data, α_i is not. The fixed effects model eliminates α_i by de-meaning the variables using the within transformation, i.e. differencing all variables with respect to their time series mean, with $Y_{it} - \bar{Y}_i$ for the dependent variable, where $\bar{Y}_i = \frac{1}{T}\sum_{t=1}^{T} Y_{it}$, and similarly for $\mathbf{X}_{it}$ and ε_{it}. Since $\alpha_i = \bar{\alpha}_i$, the unit fixed effect is eliminated without the need for it to be observed.

In our setup, cross-sectional units are given by $ir \in N \times R$ sector-region observations. We therefore estimate the following specifications with varying dimensions of fixed effects:

$$Y_{irt} = \mathbf{X}_{irt}\beta + \varepsilon_{irt} \tag{6}$$

$$Y_{irt} = \mathbf{X}_{irt}\beta + \alpha_i + \alpha_r + \varepsilon_{irt} \tag{7}$$

$$Y_{irt} = \mathbf{X}_{irt}\beta + \alpha_{ir} + \varepsilon_{irt} \tag{8}$$

where $\mathbf{X}_{irt} = [X_{irt,1}, ..., X_{irt,L}]'$ is a vector of L covariates. The first line regresses gross value added on the explanatory variables, the second line adds fixed effects at the sector i and region r levels, and the last line adds dyadic fixed effects at the sector-region (ir) levels. These specifications allow us to control for unobserved heterogeneity in two dimensions, either separately in eq(7) or in a dyadic way as in eq(8). We also estimate a pooled OLS without fixed effects following eq(6) as a benchmark model. Note that we do not include fixed effects in the time dimension: including these would be feasible for in-sample prediction, but it would be impossible to use them to forecast values for the next year, as the estimated coefficients on the time components would be missing.

Assumptions Estimating a fixed effects model implies that some assumptions must hold to ensure the model produces consistent and reliable estimates. These assumptions are: (i) linearity, (ii) constant unobserved heterogeneity, (iii) strict exogeneity, (iv) sufficient within-unit variation, (v) no perfect multi-collinearity, (vi) no serial correlation, and (vii) "large N, small T". First, linearity implies that

the relationship between the dependent and independent variables is linear. Hence why variable transforms can be useful to improve linearity. Second, within units ir, unobserved heterogeneity is constant over time. The fixed effects then absorb these unobserved characteristics of each unit. Third, strict exogeneity implies that the error term must be uncorrelated with the explanatory variables across all time periods. E.g. for eq(7): $\mathbb{E}[\varepsilon_{irt} \mid \mathbf{X}_{irt}, \alpha_i, \alpha_r] = 0$. Fourth, $\mathbf{X}_{irt}$ must vary over time within sectors i and regions r, and across sector-region combinations. Fifth, unit-invariant variables (e.g., region-wide constants) cannot be estimated as they are absorbed by α_i or α_r. Sixth, errors are to be serially uncorrelated for valid inference. Researchers often use robust or clustered standard errors. Finally, a sufficiently large N relative to T ensures that the fixed effects are reliably estimated.

Strengths and limitations Fixed effects models are commonplace in economics. Their strengths include the ability to control for unobserved within-unit differences across units, avoiding the need for including data on possibly many covariates that might co-move in the dimension of interest, including sector-region characteristics that are constant over time but hard to observe (e.g. institutions), or time trends. Another strength is that the model exploits a lot of variation in the data across all units to estimate gross value added, by pooling information to jointly estimate parameters (the slope of the regression line for each coefficient β), while the levels (intercepts) are allowed to differ. Conversely, especially for inference, one cannot estimate the effects of variables that do not change within units (e.g., constant policies). Moreover, saturating models with fixed effects does not imply causality, as inference relies on strict exogeneity. However, we are not after inference per se. More pungent in our setup is the fact that these models might have a misleadingly high R^2 due to the many fixed effects, and even a high adjusted R^2, potentially from autocorrelation within the time series of units. Hence why we also implement VAR/VEC models to explicitly account for these patterns.

7.2 Estimation

We implement the FE models using Python's Pyfixest package. We specify the different variants of fixed effects as in eq(6), eq(7), and eq(8). We also incorporate additional covariates to help predict gross value added. Two remarks. First, not all covariates are available in the data for all sector-regions and/or years. We therefore group all sector-regions into aggregate industries (see Table A2 in Appendix A). Covariates generally cover sector-regions within these industries (e.g. VAT and business survey data for manufacturing and services industries, see Table 2). This allows us to separately estimate the FE models for sector-regions for each of the four broad industries ("Primary and extraction", "Manufacturing", "Services", and "Non-market services"). This ensures units will not drop out of the regressions due to missing values for covariates, while adding additional flexibility by relaxing common slopes in the regressions for all sector-regions. The downside is that we use fewer observations for each model estimation, but the FE models do exhibit many degrees of freedom due to pooling observations to estimate common parameters.

Table 15 provides a schematic overview of the panel fixed effects implementation in the Python toolbox. Estimation proceeds as follows. First, we select a high-level industry. We then allocate the particular covariates to sector-regions that are part of this high-level industry (see Table 2). Next, we estimate a fixed effects models for each of the model specifications eq(6), eq(7) and eq(8). Standard errors are robust following MacKinnon and White (1985) heteroskedasticity robust standard errors. We do this for each variable transform. In particular, we transform both the dependent and the independent variables in the same way each time. E.g. $\ln(Y_{irt}) = \ln(\mathbf{X}_{irt})\beta + \alpha_i + \alpha_r + \varepsilon_{irt}$, etc. Each

Steps to implement the fixed effects model.
1. Select an aggregate industry: primary, manufacturing, services, non-market services.
2. Choose a variable transform for Y_{irt}: levels, logs, standardized, inverse, square root.
3. Choose the fixed effects structure: no FE, one-dimensional FE, dyadic FE.
4. Estimation: Estimate the model with potential covariates.
5. Prediction: predict Y_{irt} for the transformed variable.
6. Obtain $\hat{Y}_{irt}$ in levels: reverse the transformation of $\hat{Y}_{irt}$.
7. In-sample goodness-of-fit: calculate NRMSE on the untransformed variable.
8. Out-of-sample performance: leave-one-out cross validation to calculate out-of-sample NRMSE.
9. Repeat steps 1 to 8 and iterate over all aggregate industries, transforms, and FE structures.

Table 15: Panel fixed effects steps.

fixed effects specification runs for each aggregate industry and transform. For 5 transforms and 3 FE specifications, there will be 15 models per aggregate industry. If the dependent variable has incompatible values with the transform (such as negative values for logarithms and square roots) we skip estimating that transform. If we detect problematic values for a covariate, we include it in levels instead. For example, the business confidence index variable has mostly negative values. In that case for log and square root transformations, the index will be used in levels while the other variables are transformed, as long as they have no negative values.

Prediction Given parameter estimates for all β, and α_i and α_r, or α_{ir}, predicted values $\hat{Y}_{irt}$ are obtained. We calculate (N)RMSE for each sector-region separately by calculating the RMSE for the actual versus fitted values for each sector-region separately, and normalize the RMSE by the average value for each sector-region. As always, transforms are reversed before predicting $\hat{Y}_{irt}$ and calculating (N)RMSE values. The Pyfixest package currently does not include a Hausman specification test, which compares an estimator that is known to be consistent (fixed effects) with another estimator that is also efficient (random effects). We therefore do not provide the result of this test in the report.

Validation To evaluate the model's out of sample performance, we validate each estimated model using Leave-One-Out Cross Validation (LOOCV). Leave-one-out cross-validation uses all data except for one data point as the training set and then tests the model on that single data point. This process is repeated until every observation in the dataset has served exactly once as the test set. See Figure 16 for a schematic overview of the method. In particular, we use as a training set all *irt* observations in the data except one observation and validate the model on the excluded data point. We obtain a residual each time for the observation in the test sample. Once we have the residual for all data points (meaning that we went through all iterations to have each data point as test sample) we just group over the sector-region and compute the RMSE and NRMSE. This method is very efficient as it makes maximal use of the available data, and it is computationally inexpensive for smaller datasets. Another advantage is that it does not rely on sampling (with replacement) like in bootstrapping. This approach

has very low bias (i.e., systematic over- or underestimation of the true parameters) because almost all data is used to train the model. However, it can result in higher variance (variability of the estimate across different samples) in estimating the model's performance. Alternative methods are k-fold cross validation, which splits he dataset into k smaller subsets or "folds" of size $1/k$. The model is then trained on $k-1$ folds and tested on the fold that is left out. LOOCV is an extreme form of k-fold cross validation where each fold is the size of one observation.

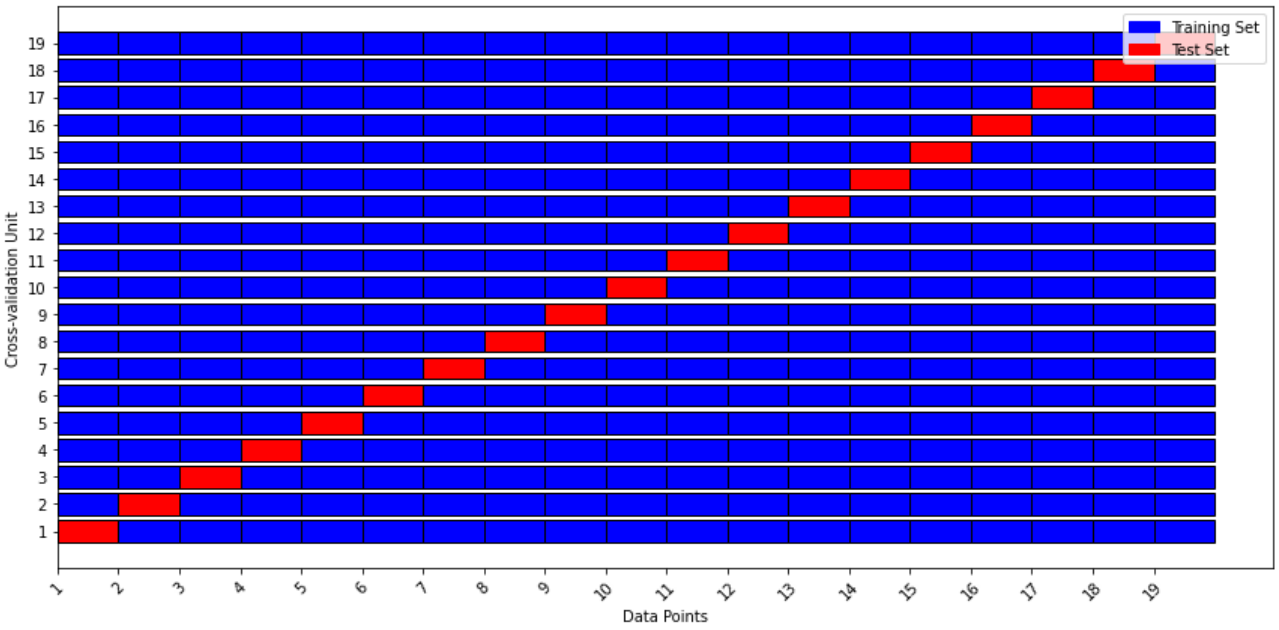


Figure 16: Leave-one-out cross validation.

7.3 Results

The toolbox generates the following outputs for both gross value added in current prices and chained prices in `/task7_panel/output`, in the sub-folders `/csv`: (i) `panel_predictions.csv`, (ii) `panel_validation.csv`, and (iii) `panel_coefficients.csv`, as well as a plots in the sub-folder `/plots` and tables in `/tex`.

The `panel_predictions.csv` file contains the following variables: `time`, `fe`, `transform`, `final_sector`, `region`, `sector`, `gva_currentprices`, `gva_currentprices_pred`, `rmse`, and `nrmse`. In particular, these variables reflect the time series of gross value added for each sector-region, predicted values (in-sample), the forecast for the next missing year (out-of-sample), and the RMSE and NRMSE for each estimated model. Up to five transformations of gross value added per sector-region are evaluated. We impose the same transform on the covariates as gross value added. If this would lead to dropping observations in the covariates, we use the covariates in levels ('raw') to estimate the model. We also report the model variant in terms of fixed effects: no FE, separate FE, and dyadic FE. Models are jointly estimated for each aggregate industry across the three regions, with sector-region specific predictions, as well as their (N)RMSE values.

We have 48 estimated models out of 60 potential models (5 transforms, 4 aggregate industries, and 3 FE specifications) for a total of 471 sector-region-transforms (out of 555). We cannot estimate models for the log and square root transforms for the aggregate industry "Manufacturing" due to

negative value added for some sector-regions. This implies dropping these models for each of the three FE specifications, totaling 12 models. We do estimate gross value added for sector-regions within Manufacturing using the other three transforms. The distribution of in-sample NRMSE for each sector-region-transform and FE choice is shown in Figure 17. The median NRMSE is now 21%, quite a bit higher than in the $ARIMA$ and VAR/VEC models. This can be the result of the trade-off to pool observations to increase the degrees of freedom while estimating fewer parameters, implying that there is quite some heterogeneity across sector-regions that is lost when imposing a more aggregate structure on the data, in this case the joint coefficients of covariates. The mean is very high at 239%, and is driven by some outliers.

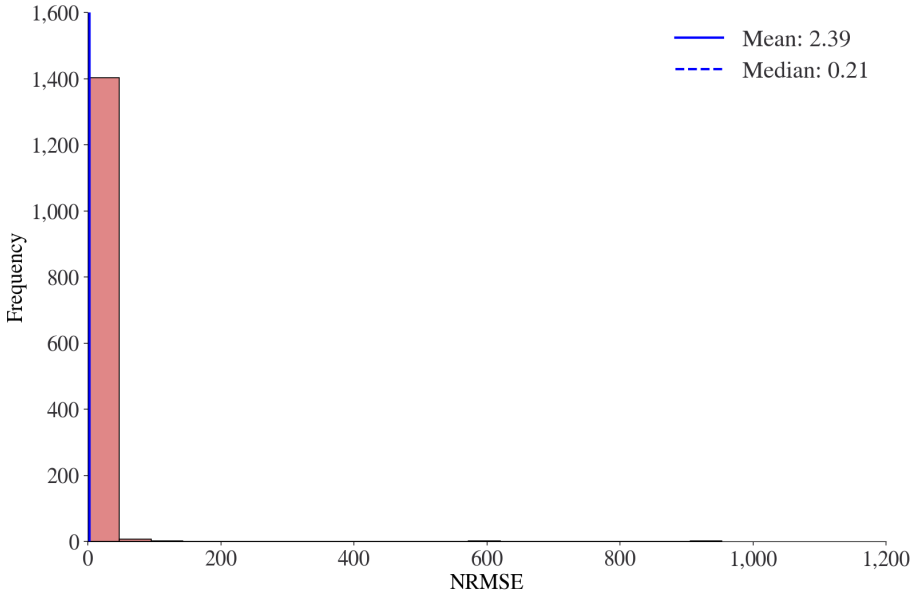


Figure 17: Distribution of in-sample NRMSE across panel FE models.

Next, the `panel_validation.csv` file contains the pseudo-out-of-sample RMSE and NRMSE values for each cross-validation per estimated model, to be used as inputs to the ensemble prediction. Table 16 reports the distribution of the average NRMSE from the validation stage across cross validation folds, for each variable transform. For each transform, there can be 333 sector-region specific observations: 37 sectors, 3 regions, and 3 FE specifications. We have fewer observation for the log and square root transforms. We see that both the median (around 13-21%) as well as the mean out-of-sample NRMSE are higher than in the $ARIMA$ and VAR/VEC models, suggesting that the FE models forecast worse and will generally obtain a smaller weight in the ensemble stage of the model. Turning to how well each fixed effects choice performs across forecasts in Table 17, we see that generally the specifications with dyadic FE perform better at the median, while the no FE models perform worst. We report the top 5 and bottom 5 sector-region-transforms in Table 18. We find very good NRMSE values for the best performers (around 0.7% forecast error), for public sectors PP (Education), TT (Activities of households as employers) and OO (Public administration). This might be partly due to public sector wages being much closer related to value added for these sectors, e.g. due to lower wage variability in these sectors relative to the market sectors. Conversely, the inverse

transforms seem to perform very poorly out of sample.

Transform	Obs	Mean	SD	Min	p50	Max
Raw	333	0.504	0.809	0.006	0.214	7.278
Log	207	0.221	0.280	0.023	0.132	2.150
Square Root	207	0.283	0.507	0.006	0.145	5.008
Inverse	333	5.982	35.462	0.007	0.774	578.045
Standardized	333	0.504	0.809	0.006	0.214	7.278

Table 16: Summary statistics of validation NRMSE by transform type for panel FE models.

FE type	N	Mean	SD	Min	p50	Max
No FE	471	2.852	27.743	0.015	0.348	578.045
Separate FE	471	1.082	7.241	0.006	0.223	152.897
Dyadic FE	471	1.229	9.067	0.006	0.144	148.769

Table 17: Summary statistics of validation NRMSE by fixed effects type for panel FE models.

Region	Sector	Transformation	NRMSE	Rank
Best 5				
Wallonia	PP	sqrt	0.0062	1
Wallonia	PP	standardized	0.0067	2
Wallonia	PP	raw	0.0067	3
Brussels	TT	inverse	0.0069	4
Wallonia	OO	standardized	0.0112	5
Worst 5				
Wallonia	CI	inverse	10.4453	467
Wallonia	CE	inverse	28.4334	468
Wallonia	CC	inverse	67.1012	469
Brussels	BB	inverse	106.2120	470
Brussels	LL	inverse	148.7693	471

Table 18: Top 5 and Bottom 5 sector-region-transforms by validation NRMSE for dyadic FE models.

We also include a `panel_coefficients.csv` file, reporting estimated coefficients for each

covariate per model, together with their p-value, as well as model R^2 and within R^2 (after partialling out the fixed effects), and a dummy if a variable is collinear with other regressors. While not the focus of the ensemble prediction, users can check this file for estimated coefficients for the covariates. Table 19 reports the distribution of coefficients across all estimated models, for those coefficients that are significant at the 5% level. It turns out that there are some estimated coefficients for which we expect a positive sign but see an estimated negative coefficient (e.g. hours worked for employees, or employment rate). We flag this observation in the report and leave it to the user to include this in her judgment of the models. Again, our measure of predictive performance is the validation NRMSE, rather than identification of parameters per se.

Variable	N	Mean	SD	Min	p50	Max
activity_rate	2	212.98	298.20	2.12	212.98	423.84
business_confidence_index	9	1.93	2.89	0.00	0.02	6.43
compensation_employees	46	1.14	0.69	-0.17	1.05	3.61
employment_rate	5	-1.53	3.41	-7.63	0.00	0.00
hours_worked_employees	30	-0.82	6.31	-21.64	-0.06	17.13
hours_worked_selfemployed	41	0.10	0.14	-0.00	0.04	0.57
nbuildings	1	0.00	.	0.00	0.00	0.00
nbuildings_onedwelling	1	10.31	.	10.31	10.31	10.31
ndwellings	4	11.88	22.36	0.12	1.00	45.41
surface_area	6	-34.63	51.50	-114.64	-3.71	-0.18
surface_area_habitable	3	0.01	0.02	-0.00	0.00	0.03
unemployment_rate	2	-45.59	64.47	-91.18	-45.59	-0.00
vat_investments	12	0.05	0.59	-1.09	0.00	1.25
vat_purchases	13	-0.65	1.47	-3.35	0.06	0.30
vat_turnover	14	1.01	2.26	-0.34	-0.05	5.46
volume	2	8.34	11.57	0.16	8.34	16.52

Table 19: Summary Statistics estimated coefficients for panel FE models.

Finally, the sub-folder `/plots` the time series graphs for all sector-regions and each estimated transformation as well as fixed effects choice. Sector-regions are now jointly estimated within an aggregate industry specification and allocated covariates. Figure 18 shows actual, fitted, and forecast values for our running example. We show the results for the choice of fixed effects and variable transform that generates the lowest out-of-sample NRMSE. It turns out that the dyadic FE setup provides the best NRMSE for all three sectors, while the transforms can vary again across sectors.

Figure 18: Actual, fitted, and forecast values for selected sectors in the panel FE model.

8 Input-output structures: spatial auto-correlation models

The next class of models we estimate are spatial auto-correlation models. These models account for potential cross-sectional dependence of outcomes across related units. Most often, this cross-sectional dependence is modeled in a spatial way: e.g. geographic units that are closer to each other might be correlated in terms of their outcomes due to geographic spillovers. Examples include housing values across districts, GDP and income levels across countries, pollution values across cities, etc. Instead, we model the spatial autocorrelation as dependence through value chains: sector-regions that rely on each other in terms of input use in their production structure, might be related in their outcomes. This class of models is complementary to the previous models, explicitly modeling cross-sectional dependence in a panel setup.

8.1 Setup

Specification We start from the classic Spatial Auto-Regressive (SAR) specification, which explicitly accounts for spatial dependence in the dependent variable. This specification incorporates a spatial lag of the dependent variable, allowing for the value of the dependent variable in one location to depend on its values in neighboring locations. Formally:

$$\mathbf{Y}_t = \rho \mathbf{W} \mathbf{Y}_t + \mathbf{X}_t \beta + \varepsilon_t \tag{9}$$

where $\rho \mathbf{W} \mathbf{Y}_t$ models the spatial lag, with spatial auto-regressive coefficient ρ measuring the strength and direction of spatial dependence, $\mathbf{W}$ a spatial weight matrix that defines the structure of spatial relationships between observations, and $\mathbf{X}_t$ a vector of covariates. Typically, $\mathbf{W}$ is a row-standardized matrix where each element w_{ij} represents the influence of unit j on unit i. A positive ρ then indicates that large values for $\mathbf{Y}_t$ in neighboring units are associated with high values of $\mathbf{Y}_t$ in the current unit. There are other specifications of the spatial auto-correlation model, including Spatial Error Models (SEM) that include spatial auto-correlation in the error terms, and Spatial Durbin Models (SDM) that also include spatial lags of independent variables. These specifications can also be combined in a richer spatial model.

Input-output relationships Common specifications for $\mathbf{W}$ include contiguity matrices from polygon shapefiles or geographical distance-based weights. However, we implement $\mathbf{W}$ using information on input-output linkages across sector-regions. The intuition is that gross value added of a given sector-region might be positively correlated with the gross value added of another sector-region through input-output linkages. For example, an increase in the steel industry in Wallonia might positively correlate with output in the car industry in Flanders. We start from the Multi-Regional Input-Output (MRIO) Tables for 143 NACE sectors across the three Belgian regions for the year 2015 (Avonds et al. (2021)). Flows are reported in terms of sales value from one sector-region to another. We aggregate these to the NACE A38 sectors across the three regions we use in the toolbox.

We operationalize the spatial dependence matrix $\mathbf{W}$ in two ways: as the direct requirements matrix, and as the total requirements matrix. In particular, the direct requirements elements are given by the technical coefficients $a_{ir,js} = \frac{z_{ir,js}}{x_{ir}}$ where $z_{ir,js}$ is the value of the output from sector-region js used as

input by sector-region ir, and x_{ir} is the total output value of sector-region ir.[31] Matrix $\mathbf{A}$ is then the direct requirements matrix with elements $a_{ir,js}$. Hence, each element shows the share of inputs that is directly sourced from js in total output or ir. We also construct the total requirements matrix as $\mathbf{L} = (\mathbf{I} - \mathbf{A})^{-1}$, where $\mathbf{L}$ is the total requirements matrix or Leontief inverse, and $\mathbf{I}$ is the identity matrix, of the same dimension as $\mathbf{A}$. Each element $l_{ir,js}$ in the matrix $\mathbf{L}$ represents the total (direct plus indirect) input required from sector-region js to produce one unit of output in sector-region ir.[32] The Leontief inverse can be expressed as an infinite series $L = (I - A)^{-1} = I + A + A^2 + \cdots$, which converges if $\rho(A) < 1$, where $\rho(A)$ is the spectral radius of A (Perron-Frobenius Theorem). This is always the case for IO tables, due to the existence of value added, ensuring the stability of the input-output system. We can also write this element-by-element: $l_{ir,js} = \delta_{ir,js} + a_{ir,js} + \sum_{k,t} a_{ir,kt}a_{kt,js} + \sum_{k,t,m,u} a_{ir,kt}a_{kt,mu}a_{mu,js} + \cdots$ where $_{ir,js}$ is the Kronecker delta, equal to 1 if $i = j$ and $r = s$, and 0 otherwise (from I), $a_{ir,js}$ is the direct requirements part, and all the other terms $\sum_{k,t} a_{ir,kt}a_{kt,js} + \sum_{k,t,m,u} a_{ir,kt}a_{kt,mu}a_{mu,js}+$ are the indirect requirements.

We provide a graphical version of the total requirements matrix across sector-regions in Figure B3 in Appendix B. The spatial models require the diagonal entries of a spatial matrix to be zero to avoid 'direct' spillovers. For both the direct and total requirements, we thus set the diagonal matrix elements equal to zero.

Operationalization We make one more adjustment to the standard specification in eq(9). In particular, in order to be able to use the model for forecasting, we lag the dependent variable not only spatially, but also in the time dimension. This has three key advantages in our setup. First, it allows to use data for the last year in the data to be used as a regressor to predict the next missing year. Second, it allows us to construct $\rho\mathbf{W}\mathbf{Y}_{t-1}$ by hand, and estimate eq(9) using OLS, bypassing the recursive nature of the setup. Because of $\mathbf{Y}_t$ showing up on both sides of the equation, standard estimation generally requires Maximum Likelihood or Generalized Method of Moments instead. Third, we can estimate a panel SAR model in a simple setup. Applied to our setting, the model can then be written in expanded form as:

$$Y_{irt} = \rho \sum_{j \in N} \sum_{s \in R} w_{ir,js} Y_{jst-1} + \sum_{l \in L} X_{irt,l}\beta_l + \varepsilon_{irt}, \tag{10}$$

where $w_{ir,js}$ is the input use of sector-region js goods or services in the production of ir output and Y_{jst-1} is the one-year lagged gross value added of direct or indirect supplying sector-region js.

Assumptions The SAR model builds on the assumptions of panel fixed effects models, with additional assumptions related to spatial dependence: (i) spatial dependence in the dependent variable, (ii) exogeneity of the spatial weight matrix, and (iii) invertibility of $\mathbf{I} - \rho\mathbf{W}$. The first states that the model assumes that the dependent variable in one spatial unit is influenced by the dependent variable in neighboring units. Second, the spatial weight matrix $\mathbf{W}$ is assumed to be exogenous and known. It is known from observing it through the direct and total requirements matrices. Exogeneity is harder to convey, especially since input use and gross value added are strongly related. We envision

[31] Note that we use the transpose version of the standard notation in national accounting to remain consistent with the notation of Y_{irt} as a dependent variable.

[32] It is important to note that the SAR model enforces diagonal entries of $\mathbf{W}$ to be zero, so to estimate the indirect impact of other sector-regions on the own sector-region. This implies that the diagonals of the direct and total requirements matrices are also set to zero for this analysis.

proving exogeneity beyond the scope of the current project.[33] Finally, invertibility is required for the SAR model to be well-defined. This requires ρ to satisfy certain bounds depending on the eigenvalues of **W**. When **W** is row-normalized (i.e., rows sum to 1) or column-normalized (i.e. columns sum to 1), the largest eigenvalue is typically $\lambda^{max} = 1$, and the bounds simplify to $\rho \in (-1, 1)$.

Strengths and limitations The SAR model is a powerful tool for analyzing spatial or other cross-sectional dependence across units. It captures spatial dependence in a relatively simple way, allowing the model to capture spillover effects, which is not accounted for in the other models in the toolbox. By including spatial dependence, the SAR model often improves explanatory power compared to non-spatial models, especially in datasets where spatial interconnections are significant. The SAR framework allows a simple decomposition of effects into direct, indirect, and total effects. In terms of downsides, estimating the SAR model can be computationally demanding. Moreover, canned routines in existing packages are currently not always as developed and standardized as for the other methodologies, in particular modeling panel SAR, allowing for more flexible weight matrices, etc.

8.2 Estimation

We implement the SAR models by using the Pyfixest package for fixed effects models and implementing the weighted sum in eq(10) as an additional variable constructed by hand. We implement the specification for no FE, single FE, and dyadic FE, as in the panel data models. We also incorporate additional covariates to help predict gross value added as in the panel data setup, and estimate the models separately for the subgroups within the aggregate industries of "Primary and extraction", "Manufacturing", "Services", and "Non-market services". Standard errors are robust following MacKinnon and White (1985) heteroskedasticity robust standard errors. Like before, we transform both the dependent and the independent variables in the same way each time. If the dependent variable has incompatible values with the transform (such as negative values for logarithms and square roots) we skip estimating that transform. If we detect problematic values for a covariate, we include it in levels instead. The estimation and prediction procedures follow the same steps as the panel data models in Section 7. We validate each model and its transforms using Leave-One-Out Cross Validation (LOOCV). The final validation value for the (N)RMSE is again the average of the individual (N)RMSEs across folds. Table 20 provides a summary the implementation in the Python toolbox.

8.3 Results

The toolbox generates the following outputs in `/task8_spatial/output`, again for both current and chained prices. Now, for each version of gross value added, there are sub-folders with full sets of results for both the `/direct_requirements` and `/total_requirements` as spatial weights matrices. For each of these, we have the following files: in `/csv` we have `spatial_prediction.csv`,

[33] A standard SAR estimates $Y_t = \rho W Y_t + X_t\beta + \varepsilon_t$. The exogeneity concern comes from having Y_t on both sides of the regression, inducing simultaneity bias. The standard SAR model sidesteps this issue by setting all diagonal entries equal to zero: i.e. Y_{irt} does not directly affect Y_{irt}. There are only indirect effects through Y_{jst} for j and s. The size of these effects is captured through the direct or total requirements matrices. We go another route. The reason is that we do not have information on Y_{irt} for the year 2023: we can estimate the model for 2003-2022, but we cannot use that model to predict values for 2023, since we need Y_{irt} for 2023 on the right-hand side. Therefore, we construct the WY_t part as WY_{t-1}. We still set the diagonal entries equal to zero to avoid within sector-region GVA from a previous period to affect that sector-region's current period GVA (autocorrelation). Then, the only effects of WY_{t-1} to Y_t on the left side go through the suppliers of that sector-region, either through the direct or the indirect requirements table. So, the effects are both spatially and temporally lagged. We have gained Granger causality, performing at least weakly better than the standard SAR.

Steps to implement the spatial model.
1. Select an aggregate industry: primary, manufacturing, services, non-market services.
2. Choose a variable transform for Y_{irt}: levels, logs, standardized, inverse, square root.
3. Choose the fixed effects structure: no FE, one-dimensional FE, dyadic FE.
4. Select the spatial dependence: direct requirements or total requirements.
5. Estimation: Estimate the model with covariates.
6. Prediction: predict Y_{irt} for the transformed variable.
7. Obtain $\hat{Y}_{irt}$ in levels: reverse the transformation of $\hat{Y}_{irt}$.
8. In-sample goodness-of-fit: calculate NRMSE on the untransformed variable.
9. Out-of-sample performance: leave-one-out cross validation to calculate out-of-sample NRMSE.
10. Repeat steps 1 to 9 and iterate over all aggregate industries.

Table 20: Spatial model steps.

(ii) `spatial_validation.csv`, and (iii) `spatial_coefficients.csv`. In sub-folders `/plots` we find the various graphs related to the estimated models, and `/tex` contains the LaTeX tables.

The `spatial_predictions.csv` file reports on the following variables: `time`, `fe`, `transform`, `final_sector`, `region`, `sector`, `gva_currentprices`, `gva_currentprices_pred`, `rmse`, and `nrmse`. As always, up to five transformations of gross value added per sector-region are evaluated. We impose the same transform on the covariates as gross value added. If this would lead to dropping observations in the covariates, we use the covariates in levels to estimate the model. We also report the model variant in terms of fixed effects: no FE, separate FE, and dyadic FE. As in the panel FE models, models are jointly estimated for each aggregate industry across the three regions, with sector-region specific predictions, as well as their (N)RMSE values.

The following results are reported for the direct requirements version. The total requirements results are documented in the toolbox. Identical to the panel FE models, we have 48 estimated models out of 60 potential models (5 transforms, 4 aggregate industries, and 3 FE specifications). Non-estimated models are for the log and square root transforms for the aggregate industry "Manufacturing" due to negative value added for some sector-regions. Figure 19 shows the distribution of in-sample NRMSE for each sector-region-transform and FE choice. The median NRMSE is 20%, while the mean is 741%, driven by the model without FE: the mean decreases to 136% when excluding the models without FEs.

Next, the `spatial_validation.csv` file contains the pseudo-out-of-sample RMSE and NRMSE values for each cross-validation per estimated model. Table 21 reports the distribution of the average out-of-sample NRMSE from the validation stage across cross validation folds, by variable transform. We have the same number of observations as in the panel FE models, and the median (13-21%) and mean NRMSEs are very similar to the panel FE models. The distributions across variable transforms

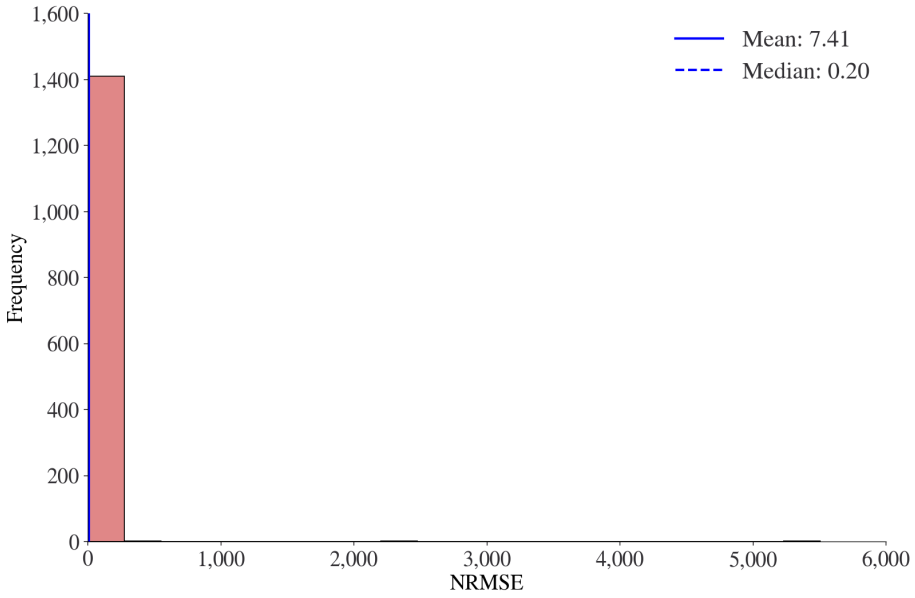


Figure 19: Distribution of in-sample NRMSE across spatial models.

are also very similar to the panel FE models, except for the inverse transform, which has a slightly lower median but much larger right tail than the panel FE models. Like before, we also report the distribution of validation NRMSE across FE specifications in Table 22. Here, all FEs specifications perform better than the panel model. We also report the top 5 and bottom 5 sector-region-transforms in Table 23. We find very good NRMSE values for the best performers (around 0.5% forecast error), for public sectors PP (Education), TT (Activities of households as employers) and OO (Public administration). These are similar sector-region-transforms as before, with a lower forecast error than under the panel FE models.

Transform	Obs	Mean	SD	Min	p50	Max
Raw	333	0.507	0.850	0.005	0.207	7.031
Log	207	0.213	0.261	0.009	0.132	2.087
Square Root	207	0.273	0.459	0.006	0.140	4.199
Inverse	333	5.344	30.150	0.004	0.674	386.731
Standardized	333	0.507	0.850	0.005	0.207	7.031

Table 21: Summary statistics of validation NRMSE by transform type for spatial models.

We also include a `spatial_coefficients.csv` file, reporting estimated coefficients for each covariate per model, together with their p-value, as well as model R^2 and within R^2 (after partialling out the fixed effects), and a dummy if a variable is collinear with other regressors. Compared to the panel FE models, there is one additional covariate, `index_gva`, which is computed as $\sum_{j \in N} \sum_{s \in R} w_{ir,js} Y_{jst-1}$, with weights given by $a_{ir,js}$ for the direct requirements version, and $l_{ir,js}$ for the total requirements version. Table 24 reports the distribution of coefficients across all estimated models,

FE type	N	Mean	SD	Min	p50	Max
No FE	471	2.607	24.522	0.011	0.321	386.731
Separate FE	471	1.061	4.169	0.007	0.220	67.813
Dyadic FE	471	1.041	6.012	0.004	0.135	109.523

Table 22: Summary statistics of validation NRMSE by fixed effects type for spatial models.

Region	Sector	Transformation	NRMSE	Rank
Best 5				
Brussels	TT	inverse	0.0042	1
Wallonia	PP	standardized	0.0051	2
Wallonia	PP	raw	0.0051	3
Brussels	OO	standardized	0.0059	4
Brussels	OO	raw	0.0059	5
Worst 5				
Brussels	MA	inverse	17.4670	467
Wallonia	CA	inverse	27.0512	468
Flanders	CA	inverse	36.8551	469
Wallonia	CK	inverse	43.7623	470
Wallonia	CI	inverse	109.5228	471

Table 23: Top 5 and Bottom 5 sector-region-transforms by validation NRMSE for spatial models with dyadic FEs.

for those coefficients that are significant at the 5% level.

Finally, the folder `/plots` contains the usual time series graphs for all sector-regions and each estimated transformation. Sector-regions are jointly estimated within an aggregate industry specification and allocated covariates. Figure 20 shows actual, fitted, and forecast values for each of the selected sector-regions in our running example. These sector-regions are jointly estimated within an aggregate industry specification and allocated covariates. Again, we show the results for the choice of fixed effects and variable transform that generates the lowest out-of-sample NRMSE.

Variable	N	Mean	SD	Min	p50	Max
activity_rate	2	11.59	1.35	10.64	11.59	12.55
business_confidence_index	9	1.91	2.86	0.00	0.02	6.38
compensation_employees	43	1.20	0.67	-0.14	1.10	3.90
employment_rate	3	0.00	0.00	-0.00	0.00	0.00
hours_worked_employees	38	-0.36	5.26	-19.95	-0.04	16.67
hours_worked_selfemployed	41	0.10	0.14	0.00	0.03	0.57
index_gva	29	-0.68	2.42	-12.93	-0.06	0.36
nbuildings	2	-0.05	0.00	-0.05	-0.05	-0.05
ndwellings	3	0.42	0.05	0.38	0.41	0.48
surface_area	5	-46.62	65.38	-137.07	-0.29	0.26
surface_area_habitable	5	0.02	0.03	0.00	0.00	0.08
unemployment_rate	3	-0.74	0.64	-1.13	-1.09	-0.00
vat_investments	13	-0.05	0.68	-1.50	0.00	1.30
vat_purchases	12	-0.70	1.51	-3.26	0.06	0.30
vat_turnover	12	1.17	2.38	-0.35	-0.07	5.24
volume	2	6.97	9.67	0.13	6.97	13.81

Table 24: Summary statistics estimated coefficients for spatial models.

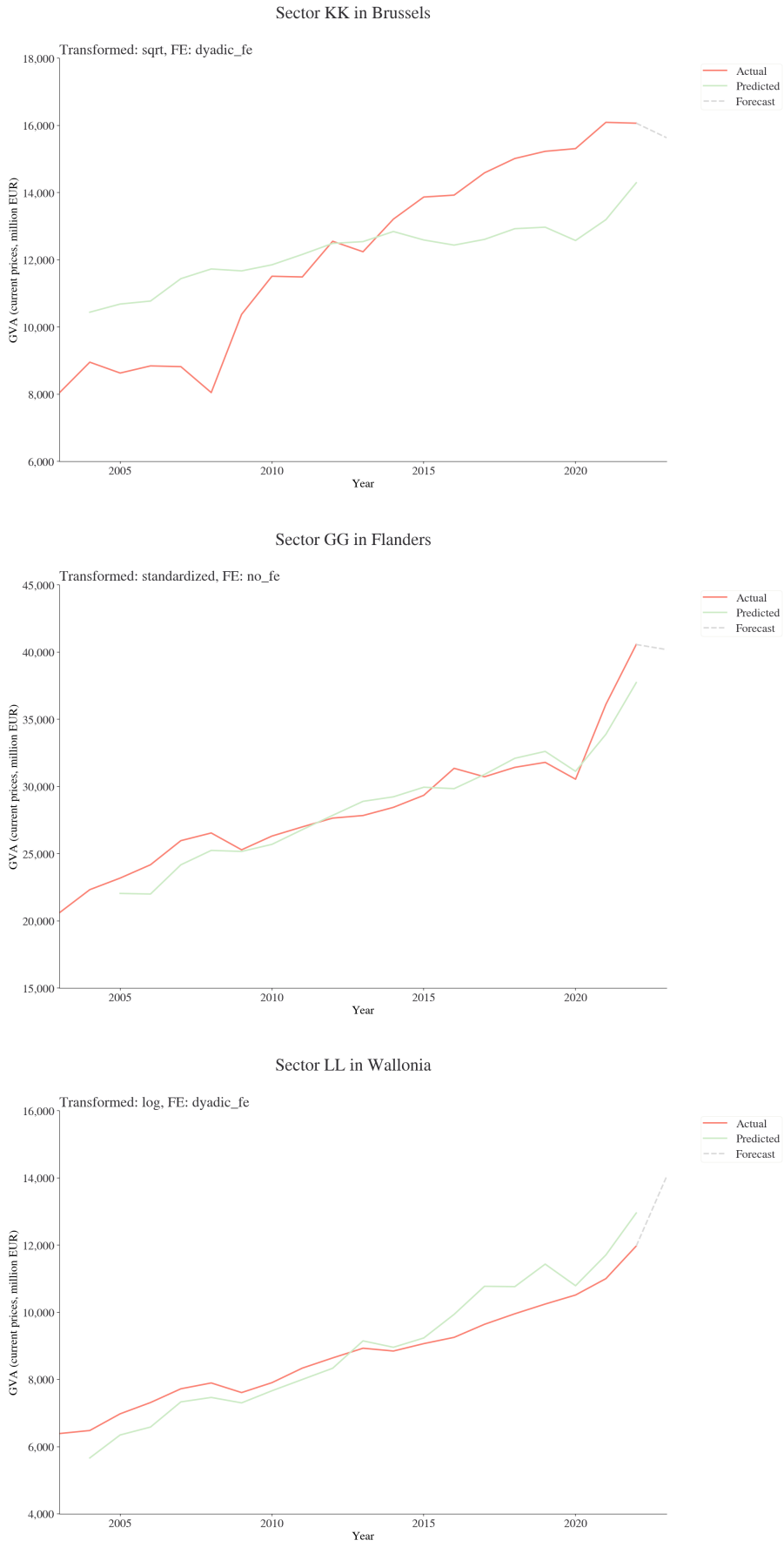


Figure 20: Actual, fitted, and forecast values for selected sectors in the spatial model.

9 Machine learning: random forests

The last class of models we estimate are random forests. Random forests are a powerful machine learning method used for classification and regression tasks. the method operates by constructing multiple decision trees and then aggregates their outputs to make predictions. The technique is based on the idea that combining multiple, potentially weak learners (decision trees) can create a strong learner (random forest) with better performance and robustness. In the context of the toolbox, random forests are highly complementary to the previous methods. Unlike traditional models where researchers specify which variables to include, a random forest automatically determines the most important predictors and interactions, making it a more automated and adaptable method.

9.1 Setup

Specification The building blocks of a random forest are decision trees, which recursively partition the covariate ('feature') space by splitting data at optimal thresholds to minimize an error metric, such as mean squared error. Each decision tree is trained on a random subsample of the data. A decision tree consists of nodes, where data is split based on a feature threshold. The top node is called the root, and it branches into child nodes. The process continues until a stopping criterion is met. The final nodes, which contain the grouped data, are called leaves. Each leaf defines a region in the feature space, and corresponds to a final decision or prediction for a given input. A random forest is then an ensemble of decision trees. By averaging their predictions, the forest reduces variability and makes more reliable forecasts than an individual tree.

Formally, let N bootstrap samples be drawn from the original dataset. For each sample, a decision tree, b, is trained by recursively partitioning the data into two parts, or regions R_L and R_R, over and over again. These regions represent the left and right child nodes of a tree. A decision tree is defined as the aggregation of all predictions across regions in the sample:

$$f_b(\mathbf{x}) = \sum_{m=1}^{M} c_m \mathbb{I}(\mathbf{x} \in R_m) \equiv \bar{y}_m, \tag{11}$$

where $\mathbf{x}$ is an input vector of features, R_m are regions of the feature space determined by optimal splits, c_m is the output value (e.g., the mean response for region m) for region R_m, $\mathbb{I}(\mathbf{x} \in R_m)$ is an indicator function that equals 1 if $\mathbf{x}$ belongs to region R_m, and 0 otherwise, and M is the total number of terminal nodes (leaves). Finally, $\bar{y}_m$ is the final prediction for a given decision tree. A region $R_m$consists of all data points that follow the same set of decision rules and receive the same final prediction. At each node of a tree, the optimal feature and threshold are selected to minimize a chosen split criterion. We use the mean squared error, defined as

$$MSE = \sum_{i \in R_L} (y_i - \bar{y}_L)^2 + \sum_{i \in R_R} (y_i - \bar{y}_R)^2 \tag{12}$$

where and R_R represent the left and right child regions after a split, and $\bar{y}_L$ and $\bar{y}_R$ are the mean target values in each region.

Building a decision tree involves the following steps:

1. Starting at the root node, which contains a bootstrapped sample of the data.

2. Selecting the best split at each node based on the feature and threshold that minimize MSE.

3. Recursively partitioning the dataset, where each split creates two child nodes containing subsets of the previous node's data.

4. Stopping when a criterion is met, such as: (i) a node reaches a minimum number of samples; (ii) further splits do not reduce MSE significantly, (iii) a pure leaf is formed (i.e., all target values in the node are identical).

Each resulting tree captures different patterns from the data due to randomness in both bootstrapping and feature selection.

Prediction Each decision tree in a random forest makes a separate prediction for an input. Instead of relying on just one tree, the random forest takes the average prediction across all trees. When making predictions, a single decision tree assigns an input $\mathbf{x}$ to a specific region R_m, based on the decision rules learned during training. The prediction is then simply the average of all observed values in that region:

$$f(\mathbf{x}) = \bar{y}_m, \quad \text{where } \mathbf{x} \in R_m.$$

Once a new data point $\mathbf{x}$ reaches a leaf node, it belongs to a specific region R_m. The prediction for $\mathbf{x}$ is the average (or majority vote) of all training samples that also fall into R_m. A random forest then takes the average prediction across many trees B, reducing errors and improving stability:

$$f_{\text{RF}}(\mathbf{x}) = \frac{1}{B} \sum_{b=1}^{B} f_b(\mathbf{x}), \tag{13}$$

where $f_b(\mathbf{x})$ is the prediction from the b-th tree and B is the total number of trees in the forest. By averaging predictions across multiple trees, the random forest reduces variance and improves generalization compared to individual decision trees.

Figure 21 provides an example of a decision tree applied to the toolbox. In particular, the example considers the aggregate industry of Manufacturing and the levels transform of gross value added, together with the set of potential covariates (or features) $\mathbf{x}$ for the Manufacturing sector. The figure shows the first nodes of one out of B =100 trees in the forest. The algorithm first samples some observations from the dataset through bootstrapping. It then builds a tree based on the covariates. The top node (root) splits the data into two parts (regions): below and above 2542.3 (million EUR) such that this split minimizes the MSE in eq(12). This split generates a predicted value c_m, in this case 1616.528. The algorithm then reruns on the 2 subgroups of the data, again generating an optimal split by minimizing the MSE for a given covariate. This process continues until the stopping criteria are met. The final splits and their predictions are then the leaves R_m. The final tree is then defined by eq(11), which provides the prediction for this tree. The process is repeated until we have 100 trees in the forest, each with a bootstrapped sample of the data and its covariates. The final prediction is then the average prediction across all trees as in eq(13).

Assumptions Random forests are flexible models. However, for them to perform well, a few conditions should generally be met. The main assumptions are: (i) independence of observations, (ii) additivity of features, (iii) sufficient data for bootstrapping, and (iv) meaningful feature relationships. First, each observation in the dataset is assumed to be independent of the others. This ensures that the bootstrap sampling process creates diverse training sets for each tree in the forest. Second,

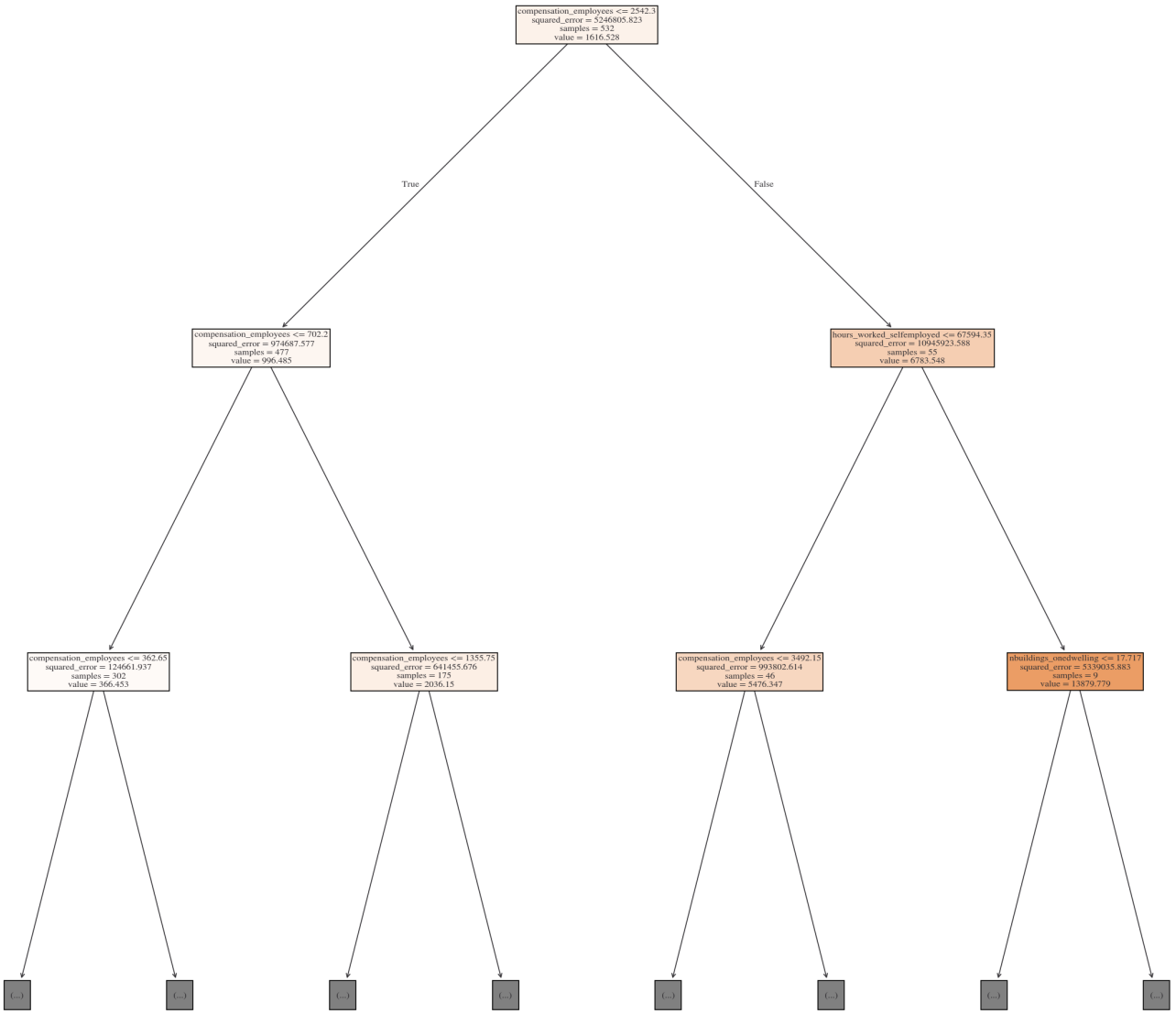


Figure 21: Example of a decision tree.

the underlying relationship between the predictors and the target is assumed to be additive (or approximately additive). While random forests can model complex, non-linear interactions, they are more efficient when the relationship is structured hierarchically or additively (e.g., predictors contribute independently or interact in specific combinations). Third, random forests rely on bootstrapping, meaning they repeatedly sample from the dataset. If the dataset is too small, many trees may end up seeing the same observations, reducing the diversity of the model. Finally, random forests assume that at least some features have a meaningful relationship with the target variable. Irrelevant or random features can degrade model performance, even though random forests have mechanisms to reduce their impact (e.g., feature selection via random splits).

Strengths and limitations Since random forests work with arguably minimal assumptions, they have a few key advantages. First, they can handle skewed distributions, non-linear relationships, and heteroskedastic data. Second, they handle high-dimensional data and large datasets effectively. Third, they are resistant to overfitting due to averaging across trees. Fourth, they are robust to multi-collinearity and can handle correlated features because each tree selects random subsets of features for splitting. Fifth, they can easily handle datasets with missing values by splitting nodes using only

Hyper parameters		Criteria	
Number of trees	100	**Minimization criterion**	MSE
Max depth of a tree	None	**Validation set**	20%
Min samples per leaf	2		
Features	1.0		

Table 25: Random forest hyper parameters and settings.

available data. Finally, random forests are less sensitive to outliers because outliers affect only specific bootstrap samples and not the entire ensemble. Flexibility comes at some costs. First, while individual decision trees are easy to interpret, a random forest, being an ensemble of trees, can be considered a ”black-box” model. Second, training and making predictions can be slower compared to simpler models, especially for large datasets or forests with many trees. Third, though reduced, overfitting can still occur, especially if the number of trees is small or hyper parameters are poorly tuned.

9.2 Estimation

We implement the random forest models using Python's Random Forest Regressor from the Scikit-learn Package. We implement the standard hyper parameters provided by the package, described in Table 25. We select mean squared error as criterion (in the code, ”criterion” = squared error). We use the default values for the number of trees (”n_estimators” = 100) and the maximum depth of a tree (”max_depth” = None). More trees generally improve performance but increase computational cost. Maximum depth limits the depth of each tree, controlling overfitting. For maximum depth None, nodes are expanded until all leaves are pure or until all leaves contain less than ”min_samples_split” samples, which is set at the default value of 2. Finally, the number of features is set at the default value of 1.0 meaning that all features are included at each split (”max_features” = 1). This value can be decreased to determine the size of the random subset of features considered at each split. For the validation part, we take 20% of the sample to construct the test set (”test_size” = 0.2). We also incorporate additional covariates ('features') for the models to choose from and estimate the models separately for the subgroups within the aggregate industries of ”Primary and extraction”, ”Manufacturing”, ”Services”, and ”Non-market services”. The model can be run at both the individual sector-region level, as well as under a panel structure. We estimate and report the version at the panel level. Table 26 reports the different steps to implement the random forest models.

9.3 Results

Results are in `/task9_random_forests/output`. Like before, the toolbox generates a full set of results for both the current and chained prices versions of gross value added. The `/csv` sub-folder contains `random_forest_predictions.csv`, `random_forest_validation.csv` and `random_forest_feature_importance.csv`. The sub-folder `/plots` contains the various plots.

The `random_forest_predictions.csv` file contains the following variables: `region`, `sector`, `time`, `gva_currentprices`, `gva_currentprices_pred`, `rmse`, `nrmse`, and `transform`.

Steps to implement the random forests.
1. Choose a variable transform for Y_{irt}: levels, logs, standardized, inverse, square root.
2. Bootstrap sampling: for each tree b, draw a bootstrap sample of size N (with replacement).
3. Feature selection: at each node, select a random subset of m features. Find the best split based on the selected features.
4. Grow the three: grow each tree to its maximum depth or until a stopping criterion is met.
5. Repeat steps 2–4 to grow B trees.
6. Prediction: average predictions over B trees.
7. Obtain $\hat{Y}_{irt}$ in levels: reverse the transformation of $\hat{Y}_{irt}$.
8. In-sample goodness-of-fit: calculate NRMSE on the untransformed variable.
9. Out-of-sample performance: leave-one-out cross validation to calculate out-of-sample NRMSE.
10. Repeat steps 1 to 9 and iterate over all aggregate industries.

Table 26: Random forest model steps.

Again, up to five transformations of gross value added per sector-region are evaluated. Models are estimated as in the panel data setup using subgroups within the aggregate industries of "Primary and extraction", "Manufacturing", "Services", and "Non-market services". We report in `random_forest_feature_importance.csv` the features included in the models and their importance. The importance of a feature is computed as the (normalized) total reduction of the criterion contributed by that feature. In Table 27 we report the features used for the "Manufacturing" aggregate sector in raw transform, and we see from the importance score that the most relevant feature is the compensation for employees, followed by the number of hours worked.

As in the panel data models, there are 471 estimated sector-region-transforms out of 555 potential combinations due to negative values for the aggregate industry "Manufacturing" that do not allow the log and square root tranforms to be applied for that industry. All other models are estimated. Figure 22 shows the distribution of the in-sample NRMSE across all estimated random forest models. The random forest models perform similarly to the VAR/VEC models and outperform the panel data models in terms of in-sample NRMSE: the median value is 4%, the mean is 6%, and also the value of outliers is much lower than in all other models.

Next, the `random_forest_validation.csv` file contains the forecast values, as well as average RMSE and NRMSE values for the cross-validation per sector-region-transform. Like before, Table 28 reports the distribution of the NRMSE from the validation stage. The median out-of-sample NRMSE is around 8%, which is better than most models, except for the $ARIMA$ models. The average values are comparable to its best competitor, again the $ARIMA$ models. In fact, the $ARIMA$ models perform better for the best performing models than the random forests. However, random forests have much fewer very badly performing models for the inverse transformation. This underlines again the usefulness of estimating several models to exploit different dimensions of variation in the data.

Aggregate sector	Transform	Feature	Importance
Manufacturing	raw	compensation_employees	0.80207
Manufacturing	raw	hours_worked_employees	0.06097
Manufacturing	raw	vat_investments	0.05171
Manufacturing	raw	hours_worked_selfemployed	0.04912
Manufacturing	raw	vat_purchases	0.01509
Manufacturing	raw	vat_turnover	0.00433
Manufacturing	raw	surface_area_habitable	0.00253
Manufacturing	raw	year	0.00242
Manufacturing	raw	nbuildings_onedwelling	0.00177
Manufacturing	raw	employment_rate	0.00153
Manufacturing	raw	sector_region_id	0.00143
Manufacturing	raw	nbuildings	0.00130
Manufacturing	raw	unemployment_rate	0.00122
Manufacturing	raw	activity_rate	0.00115
Manufacturing	raw	business_confidence_index	0.00112
Manufacturing	raw	volume	0.00093
Manufacturing	raw	surface_area	0.00087
Manufacturing	raw	ndwellings	0.00044

Table 27: Feature importance for Manufacturing with raw transform.

We also report the top and bottom 5 sector-region-transforms in terms of out-of-sample NRMSE in Table 29. The random forests perform slightly worse than the panel FE or spatial setup when comparing the top 5 best performing models. However, the worst performing models do up to two orders of magnitude better than the worst models in the other classes.

Transform	Obs	Mean	SD	Min	p50	Max
Raw	111	0.155	0.251	0.013	0.085	1.902
Log	69	0.111	0.114	0.014	0.077	0.575
Square Root	69	0.112	0.110	0.016	0.084	0.572
Inverse	111	0.187	0.405	0.004	0.090	3.817
Standardized	111	0.155	0.253	0.004	0.087	1.921

Table 28: Summary statistics of validation NRMSE by transform type for random forest models.

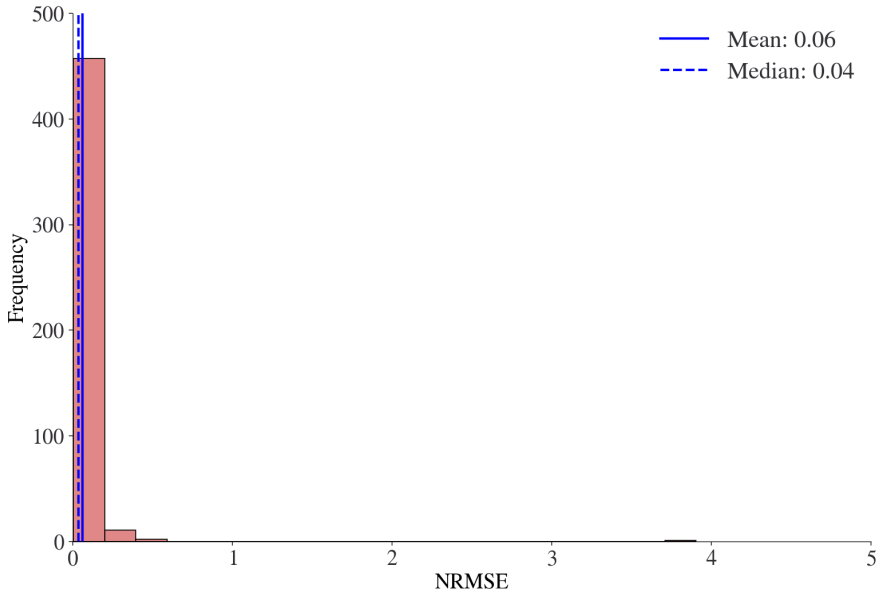


Figure 22: Distribution of in-sample NRMSE across random forest models.

Region	Sector	Transformation	NRMSE	Rank
Best 5				
Wallonia	OO	inverse	0.0040	1
Brussels	JC	standardized	0.0043	2
Wallonia	PP	standardized	0.0112	3
Wallonia	OO	standardized	0.0131	4
Wallonia	OO	raw	0.0134	5
Worst 5				
Brussels	CL	raw	1.5519	467
Brussels	CL	standardized	1.5656	468
Brussels	CD	raw	1.9017	469
Brussels	CD	standardized	1.9208	470
Brussels	CD	inverse	3.8170	471

Table 29: Best 5 and worst 5 sector-region-transforms by validation NRMSE for random forests.

Finally, the folder `/plots contains` all time series graphs for all sector-regions and each estimated transformation. Figure 23 shows actual, fitted, and forecast values for each of the selected sector-regions in our running example.

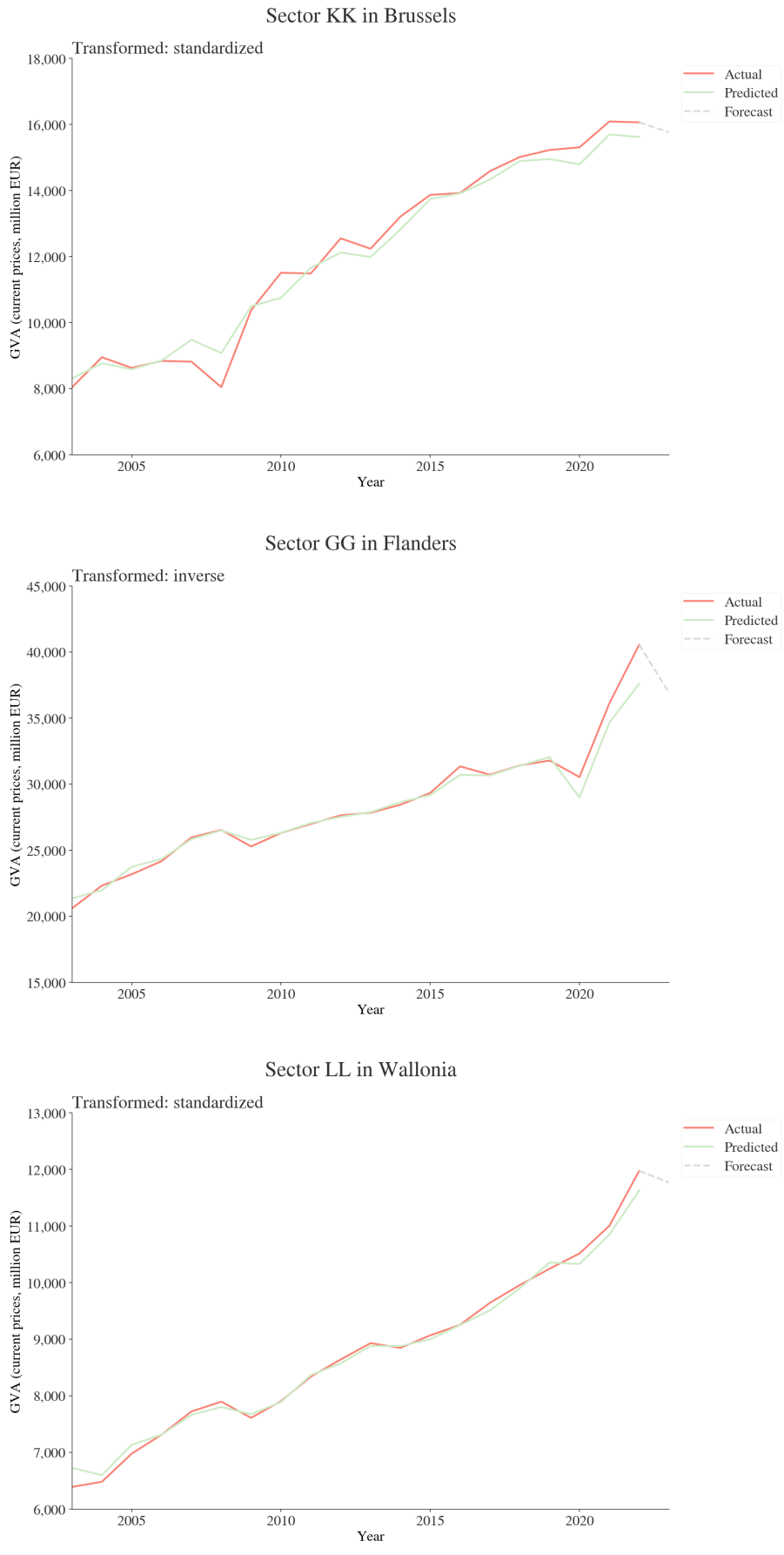


Figure 23: Actual, fitted, and forecast values for the random forest models.

10 Ensemble model and final predictions

We now use all models to construct the final predictions for each sector-region. To that end, we create an ensemble of models, a classic machine learning technique. The intuition is that combining different models generally provides better predictive power than the individual strength of a single model, leveraging the diversity and complementarity of multiple models and dimensions of variation in the data. We provide two versions of the ensemble model: the weighted average ensemble and the single best predictor. For each chosen prediction, we ensure that the final prediction is consistent with the gross value added for each sector at the Belgian level in either current or chained prices.

10.1 Setup

There are several types of ensemble models, including majority voting (used in classification, where the class that is predicted most often wins), average ensemble (taking the average prediction of models), as well as more advanced methods such as stacking, bagging, and boosting.[34] We construct two versions: a weighted average ensemble and a single best predictor.

Weighted average ensemble We first implement a weighted average ensemble, in which the final prediction for each sector-region is a weighted average of predictions across all models and transforms. Weights are given by how well the model performs pseudo out-of-sample. In particular, the ensemble prediction of gross value added $\hat{Y}_{irt}^{\text{ensemble}}$ for a sector i in region r for year t (both in-sample and out-of-sample), is given by:

$$\hat{Y}_{irt}^{\text{ensemble}} = \sum_{m=1}^{M} \omega_{irt}^{m} \hat{Y}_{irt}^{m} \tag{14}$$

where $\hat{Y}_{irt}^{m}$ is the predicted gross value added from a prediction m, and ω_{irt}^{m} is the weight allocated to each prediction m. Each m refers to a combination of a particular model ($ARIMA$, VAR/VEC, panel FE, spatial, random forest) and transform (raw, logs, square root, inverse, standardized), for a given sector-region in year t. Weights ω_{irt}^{m} are constructed as the inverse of the validation NRMSE (a lower NRMSE implies a larger weight), normalized to sum to one across all $m \in M$ for a given sector-region-year. Given that the NRMSE is sector-region-model specific, we correct for the fact that certain sector-region-models do not have predictions for all years using a time-varying dummy.[35] This modified NRMSE ensures that the normalization only includes positive weights for years for which the model makes a prediction. In practice, we compute:

$$\omega_{irt}^{m} \equiv \frac{1}{C_{irt}} \frac{1}{NRMSE_{ir}^{m}} \cdot Non\ Missing_{irt}^{m} \tag{15}$$

where $NRMSE_{ir}^{m}$ is the NRMSE for prediction m of gross value added for sector-region ir , $Non\ Missing_{irt}^{m}$ is a dummy variable equal to 1 if a sector-region-model has a non-missing prediction in year t and 0 otherwise, and $C_{irt} = \sum_{m=1}^{M} \left(\frac{1}{NRMSE_{ir}^{m}} \cdot Non\ Missing_{irt}^{m} \right)$ is a normalizing constant such that all weights across predictions m sum to one in a given year.

[34] See e.g. Chapter 16 in The Elements of Statistical Learning (Hastie et al. (2009)) for an introduction and overview.

[35] We make the weights time varying to ensure that models that contain lags ($ARIMA$,VAR/VEC, and Spatial Panel) contribute only to the ensemble prediction for observations in which they have a predicted value in a given year. E.g., due to the introduction of lagged values as controls, the 2003 prediction is missing (because it is the first year of the dataset so there is no 2002 value to use for the lagged control). With the modified NRMSE we make the normalization (and the resulting weight) year-specific so that the models with missing values receive a weight of 0.

These endogenous weights allow for the most flexibility in generating a final prediction for each sector-region. In particular, it is possible that the relative performance of individual models changes as new data arrives. For example, the predictions of the $ARIMA$ and VAR/VEC models might improve as the residual degrees of freedom will be less binding with longer time series. The weights adjust automatically when estimated on new data (e.g. future years that are added to the datasets). Additionally, users can choose to manually override weights, and select or give a higher weight to particular predictions based on their specific domain knowledge. For example, selecting one prediction only is equivalent to setting $\omega_{irt}^{m} = 1$ for one prediction and setting all others to zero. We implement this approach for the single best predictor method below. At this stage of the toolbox, users thus still have full flexibility to include which information they want for the final prediction.

Single best predictor Next, we construct the single best predictor. In particular, the ensemble prediction for sector i in region r in year t is the one that minimizes the out-of-sample NRMSE across all predictions:

$$\hat{Y}_{irt}^{\text{ensemble}} = \hat{Y}_{irt}^{m} \text{ for } m \in \min_{\text{OOS NRMSE}} \{\text{OOS NMRSE}_1, ..., \text{OOS NMRSE}_M\} \tag{16}$$

Note that we still exploit all the model estimations, and compare their predictive power before selecting one model that predicts best pseudo-out of-sample for this particular sector-region.

10.2 Ensuring consistency with national projections from HERMES

Finally, we rescale the ensemble predictions for the out-of-sample year, denoted as $T+1$, to $\hat{Y}_{irT+1}^{\text{final}}$,so that they are fully consistent with the national projections from HERMES. We rescale predicted values both for current prices and chained prices. Each version requires a separate methodology to ensure consistency.

Values in current prices The formula for predictions in current prices is:

$$\hat{Y}_{irT+1}^{\text{final}} = \hat{Y}_{irT+1}^{\text{ensemble}} \times \frac{Y_{iT+1}^{\text{HERMES}}}{\sum_r \hat{Y}_{irT+1}^{\text{ensemble}}} \tag{17}$$

where Y_{iT+1}^{HERMES} is the national value in current prices from HERMES for year $T+1$ (2023 in this case). The rescaling factor $\frac{Y_{iT+1}^{\text{HERMES}}}{\sum_r \hat{Y}_{irT+1}^{\text{ensemble}}}$ is constant across regions for sector i.[36] This can be rewritten as:

$$\hat{Y}_{irT+1}^{\text{final}} = \hat{w}_{irT+1}^{\text{ensemble}} \times Y_{iT+1}^{\text{HERMES}} \tag{18}$$

where $\hat{w}_{irT+1}^{\text{ensemble}} \equiv \hat{Y}_{irT+1}^{\text{ensemble}} / \sum_r \hat{Y}_{irT+1}^{\text{ensemble}}$ is the share of valued added for sector i in region r according to the ensemble predictions in the out-of-sample year $T+1$.

[36] For example, if we obtain $\sum_r \hat{Y}_{irT+1}^{\text{ensemble}} = 25 + 50 + 25 = 100$, while the Belgian aggregate from HERMES is 120, we rescale each region's predicted value added proportionally so that they sum to 120. The rescaling factor, which is common across regions for a given sector i, is equal to $120/100 = 1.2$. In other words, the observed aggregate value is 20% larger than the sum of our predictions so to match the aggregate we increase our regional predictions by 20%. In the example, we obtain $\sum_r \hat{Y}_{irT+1}^{\text{final}} = 25 \times 1.2 + 50 \times 1.2 + 25 \times 1.2 = 30 + 60 + 30 = 120$.

Values in chained prices We rescale the gross value added predictions in chained prices similar to the methodology in Bassilière et al. (2008).[37] We first compute growth rates that are consistent with observed national data from T to $T+1$, $\tilde{y}_{irT+1}$, for value added in chained prices of sector i in region r. Then, new values of the out-of-sample predictions are obtained applying the consistent growth rates to observed data for the last in-sample year T. We denote the out-of-sample (2023) ensemble prediction in chained prices for sector i and region r as $\hat{Y}_{irT+1}^{\text{ensemble-chained}}$. Then, growth in chained prices $\hat{y}_{irT+1}$ is equal to:

$$\hat{y}_{irT+1} = \frac{\hat{Y}_{irT+1}^{\text{ensemble-chained}} - Y_{irT}^{\text{chained}}}{Y_{irT}^{\text{chained}}}$$

where Y_{irT}^{chained} is the value of the gross valued added in chained prices for sector i in region r at the last year in-sample T (i.e., 2022 in this report). Similarly, we use data to compute the growth rate in chained prices for sector i at the national level as:

$$\bar{y}_{iT+1} = \frac{Y_{iT+1}^{\text{chained}} - Y_{iT}^{\text{chained}}}{Y_{iT}^{\text{chained}}}$$

where $Y_{iT+1}^{\text{chained}}$ and Y_{iT}^{chained} are observed national values in chained prices for the first out-of-sample and last in-sample years, respectively. Next, we define:

$$\gamma_{irT+1} \equiv \frac{(1+\hat{y}_{irT+1})}{(1+\bar{y}_{iT+1})}$$

and using observed values for gross value added in current prices, we also define the share of value added in sector i for region r in the last in-sample year T as:

$$\nu_{irT} \equiv \frac{Y_{irT}}{Y_{iT}}$$

Then, with 3 regions, we assign an index to the regions based on their alphabetical order, $r \in \{1,2,3\}$ (i.e., the region Wallonia is assigned a value of 3). We compute growth rates for gross value added that are consistent with national data using the following system of equations:

$$\begin{cases} \tilde{y}_{i1T+1} = \gamma_{i1T+1} - 1 + \gamma_{i1T+1}\tilde{y}_{i3T+1} \\ \tilde{y}_{i2T+1} = \gamma_{i2T+1} - 1 + \gamma_{i2T+1}\tilde{y}_{i3T+1} \\ \tilde{y}_{i3T+1} = \frac{\bar{y}_{iT+1} - (\gamma_{i1T+1}-1)\nu_{i1T} - (\gamma_{i2T+1}-1)\nu_{i2T}}{\gamma_{i1T+1}\nu_{i1T} + \gamma_{i2T+1}\nu_{i2T} + \gamma_{i3T+1}\nu_{i3T}} \end{cases}$$

This system of three equations in three unknowns can be solved numerically. Finally, we compute the levels of gross value aded in chained prices, $\hat{Y}_{irT+1}^{\text{final-chained}}$, as:

$$\hat{Y}_{irT+1}^{\text{final-chained}} = (1+\tilde{y}_{irT+1})Y_{irT}^{\text{chained}} \tag{19}$$

where $\tilde{y}_{irT+1}$ are the consistent growth rates and Y_{irT}^{chained} are the observed 2022 chained prices data

10.3 Results: ensemble model

Results for the ensemble model are available in the toolbox folder `/task10_ensemble/output`. As always, results are available for both `/gva_currentprices` and `/gva_chainedprices`. Within

[37] The method is different in that we predict levels of value added and then retrieve growth rates, while Bassilière et al. (2008) predict growth rates and then retrieve level variables.

each, there are sub-folders `/csv`, `/plots` and `/tex`. The `/csv` sub-folder contains the following files: (i) `weights.csv`, (ii) `full_models_file.csv`, (iii) `weighted_ensemble_end_year.csv`, (iv) `rescaled_weighted_ensemble_predictions.csv`, and (v) `rankings_based_on_nrmse.csv`. The sub-folder `/plots` contains all related plots and `/tex` the tables. Weights are merged with the full models predictions to compute the end year ensemble (in this case for the year 2023). These are then rescaled to match the Belgian aggregates. In the folder `/comparison_chained_and_current` we save files related to the comparison between forecasts obtained using the current and chained prices variables. The data are in `/csv` while the tables in `/tex`.

Ensemble predictions The `weights.csv` file contains the following variables: `model, region, sector, transform, fe, time, nrmse`, and `weights`. The file contains 139,860 observations: one for each sector-region-transform by year and by estimated model. There are six estimated models: $ARIMA$, VAR/VEC, panel FE, spatial (direct requirements), spatial (total requirements), and random forests. The file is exhaustive, in the sense that all combinations are reported as an observation. If a combination does not exist, the corresponding weight is set to zero.

Figure 24 shows a kernel density plot of the distribution of weights ω_{irt}^{m} across all models and years, while Table 30 reports their moments in more detail. On average across all models, year-specific weights are relatively small, at 1.7%. These averages are comparable across models. The median is lower for everey model, indicating right-skewness of the distribution, or the fact that some predictions are much more important than the average prediction, due to their good out-of-sample performance. The $ARIMA$ models provide the largest weights on average (4.1%), followed by the random forest and VAR/VEC models. All models have some predictions with zero weight, i.e. models that failed to provide forecasts for particular transforms, e.g. due to negative values in levels. However, each model also provides some large contributors to the ensemble prediction, with maximum weights between 13.1% for the spatial total requirements model, up to 57.7% for the random forests. Table 31 shows the top 10 weights for the out-of-sample year for which we want to compute the ensemble predictions (i.e., 2023). While the top is dominated by the $ARIMA$ models, almost all the weights for this model refer to the same sector-region (CL in Brussels) across transforms for which $ARIMA$ performs better than other models out-of-sample. Random forests also generate large weights, varying across sectors, regions, and transforms.

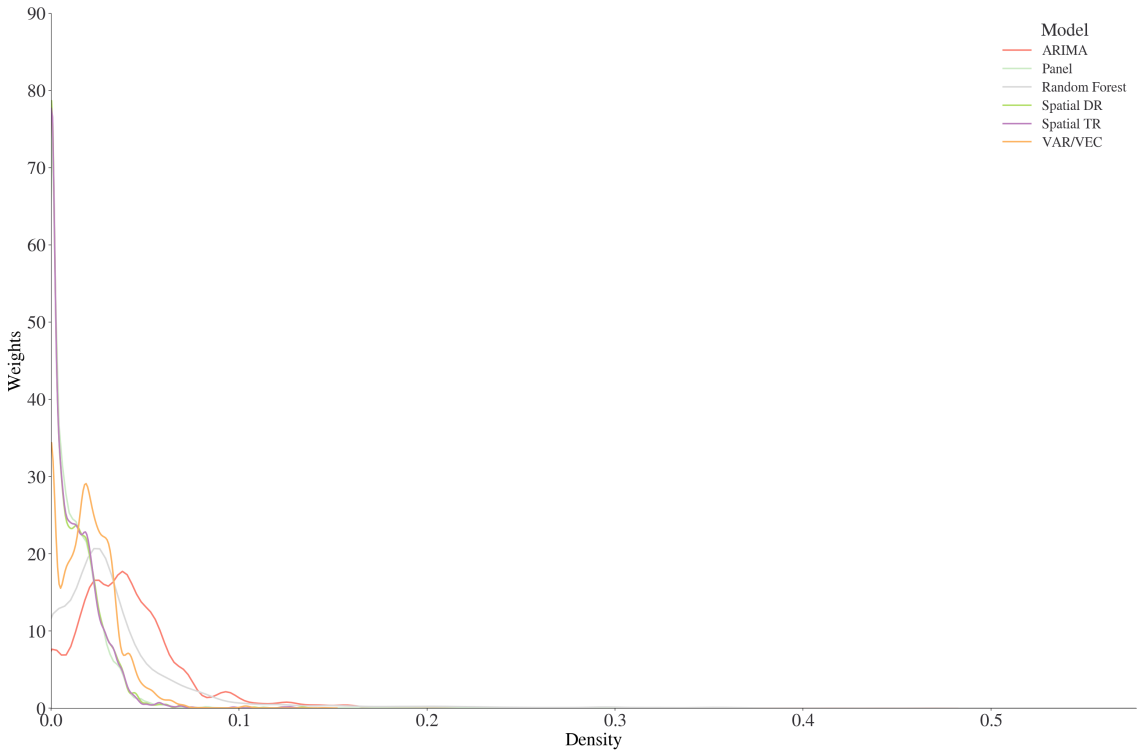


Figure 24: Distribution of weights across estimated models and years.

Model	**Mean**	**SD**	**Min**	**p25**	**p50**	**p75**	**Max**
ARIMA	0.041	0.031	0.000	0.022	0.037	0.053	0.468
Panel	0.012	0.013	0.000	0.001	0.008	0.019	0.351
Random Forest	0.035	0.045	0.000	0.013	0.025	0.040	0.577
Spatial DR	0.012	0.013	0.000	0.001	0.008	0.019	0.136
Spatial TR	0.012	0.013	0.000	0.001	0.008	0.019	0.130
VAR/VEC	0.019	0.015	0.000	0.007	0.018	0.028	0.145
Total	0.017	0.022	0.000	0.002	0.012	0.024	0.577

Table 30: Distribution of weights by estimated model.

Next, we also compare how 'easy' it is to forecast gross value added for a particular sector-region. The `ranking_based_on_nmrse.csv` file shows the best, worst, and average rank per sector-region in terms of validation NRMSE across all models. The ranking is constructed for a given model m across sector-region based on the out-of-sample NRMSE (i.e., the sector-region with the lowest NRMSE receives a rank of 1).[38] We then compute the simple average of rankings for a sector-region across all models. Table 32 shows the top 10 and bottom 10 average rankings for all sector-regions. We see that non-market services including PP (Education), OO (Public administration), and SS (Other

[38]We do not have to construct time-varying weights as in eq(150. We use the 'raw' validation NRMSE to calculate the rank of each model.

Model	Region	Sector	Transformation	FE	Weights
Random Forest	Brussels	JC	standardized	no_fe	0.294
Random Forest	Wallonia	CM	inverse	no_fe	0.157
ARIMA	Brussels	CL	raw	no_fe	0.157
Random Forest	Brussels	CI	inverse	no_fe	0.147
ARIMA	Brussels	CL	square root	no_fe	0.142
Spatial DR	Brussels	TT	inverse	dyadic_fe	0.133
Spatial TR	Brussels	TT	inverse	dyadic_fe	0.127
ARIMA	Brussels	CL	standardized	no_fe	0.127
ARIMA	Brussels	CL	log	no_fe	0.125
ARIMA	Wallonia	CD	standardized	no_fe	0.125

Table 31: Top 10 ensemble weights in 2023.

service activities) have on average the highest rank. This is quite surprising, given that generally, non-market service sectors are conceived harder to predict, as several variables such as VAT or business indicators are not available as predictors, and other often-used metrics like productivity are harder to construct for these sectors. Conversely, on the other side of the spectrum, we find mostly sectors in manufacturing: CF (Manufacture of basic pharmaceutical products) and CD (Manufacture of coke and refined petroleum products). Due to its particular nature in Belgium, sector CD is known to be hard to forecast (e.g. containing negative value added in some years, and potential transactions that are reallocated from Brussels and Wallonia to Flanders).

Top 10			Bottom 10		
Region	**Sector**	**Average Rank**	**Region**	**Sector**	**Average Rank**
Wallonia	PP	7.70	Brussels	CD	101.15
Wallonia	OO	12.55	Wallonia	CD	99.95
Flanders	PP	17.27	Brussels	CE	95.55
Brussels	OO	17.80	Brussels	CL	88.43
Flanders	OO	18.82	Brussels	CF	85.48
Wallonia	QB	19.37	Brussels	CJ	85.28
Brussels	PP	20.78	Brussels	AA	83.70
Wallonia	SS	23.52	Wallonia	CF	83.63
Wallonia	GG	24.64	Brussels	CK	83.18
Brussels	QA	26.07	Brussels	CI	82.18

Table 32: Top 10 and Bottom 10 regions and sectors by average rank.

We also provide a scatter plot for each sector-region, and its ranking per model in the sub-folder `/plots`. For example, Figure 25 shows the ranking across all models based on its validation NRMSE for the wholesale and retail sector (GG) in Flanders. Each model variant rank is calculated as that sector-region's rank relative to all other for the same model and transform, for a total of 60 models. The average rank of this sector-region across all models and variable transforms is 31.22 (the red vertical line). Some model-transform combinations perform much better than others for this sector-region, and there is quite some heterogeneity across models and variable transforms, ranging from the best ranked models at 7 for some of the spatial model specifications, and the worst performing models ranked at 103 for some of the other spatial model specifications. Moreover, none of the models performs consistently better across all sector-regions. These results underline again the usefulness of the ensemble method to generate plausible predictions: while some models work relatively well for some sector-regions, they do perform relatively worse for others.

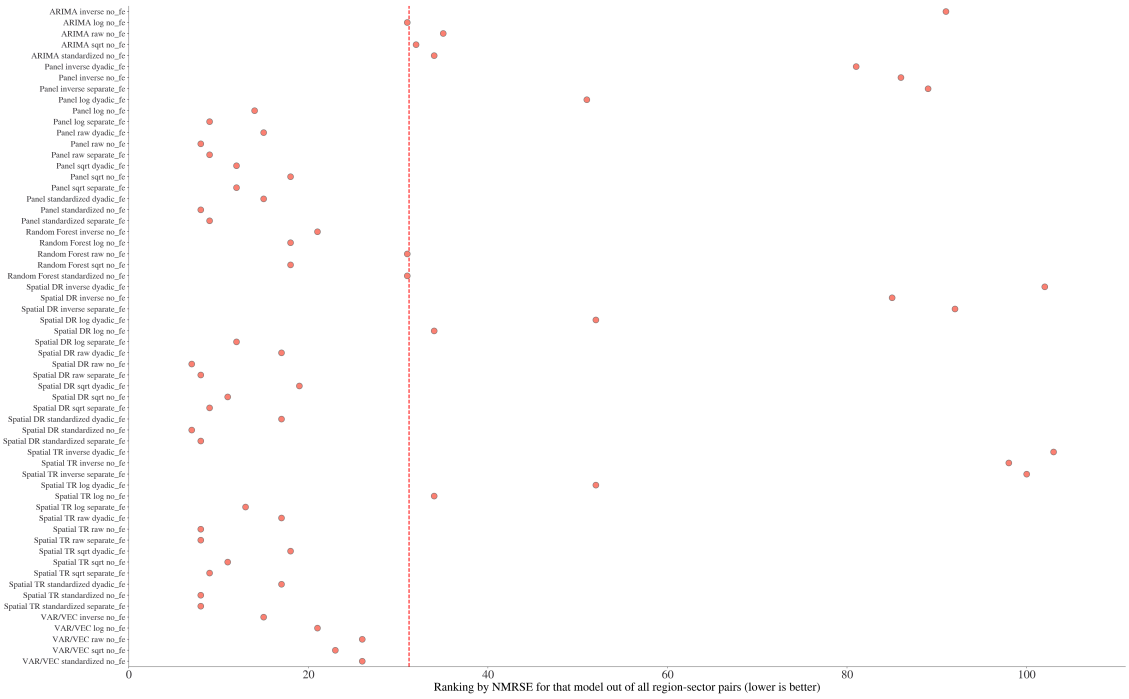


Figure 25: Ranking of NRMSE across predictions for GG - Flanders.

The following files are auxiliary to construct the final predictions: `full_models_file.csv` file contains the actual, predicted, and forecast values for each sector-region-transform prediction, as well as the weights and the ensemble prediction. The `weighted_ensemble_end_year.csv` contains the $\hat{Y}_{irT}^{\text{ensemble}}$ for each sector-region for the most recent year that is not yet available in the data (2023 in our case), aggregated across all predictions. There are also a series of bar plots in `/plots`, which show for every sector-region the predicted value added per model-transform, as well as the weights attached to each prediction. Continuing our running example, Figure 26 shows the bar plot for GG (wholesale and retail) in Flanders. The graph shows the predicted gross value added for this sector-region on

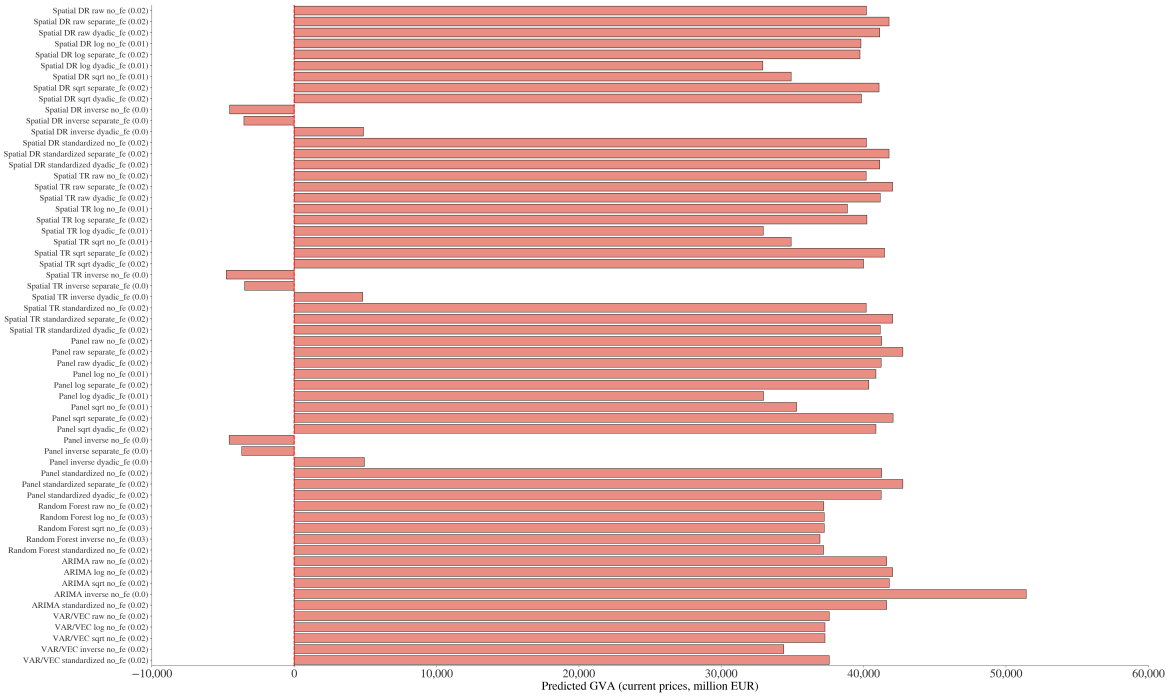


Figure 26: Predictions and weights for sector GG - Flanders.

the X axis, across all of the 60 predictions, and the models and their weights on the Y axis. Most predicted values are around the same values of 35,000-40,000 million EUR. There are also some small and negative predicted values, albeit for badly performing models, all for inverse transforms, and with tiny weights.

HERMES consistent predictions Next, `rescaled_weighted_ensemble_predictions.csv` contains the final predictions for each sector-region. It reports the ensemble predictions $\hat{Y}_{irT+1}^{\text{ensemble}}$ in the variable `gva_currentprices_ensemble`, as well as the final rescaled values $\hat{Y}_{irT+1}^{\text{final}}$ in the variable `gva_currentprices_ensemble_rescaled`. The following two variables serve as a sanity check. The `gva_currentprices_ensemble_rescaled_sum` variable sums over the sector-region values to the sector values for Belgium, and the `BE_aggregate` contains the Belgian aggregate value for that sector. Finally, `rescaling_difference` reports the difference of $\hat{Y}_{irT+1}^{\text{final}} - \hat{Y}_{irT+1}^{\text{ensemble}}$ in million EUR, while `rescaling_difference_percentage` transforms that difference to percentage terms. This variable serves as a benchmark on how close the ensemble predicted values are to the rescaled values. Table 33 shows the percent deviation between the ensemble and final predictions in current prices. Note that, across regions for a given sector, the scaling factor is constant (see eq(17), thus ending up with 37 observations, one for each sector-region. The average correction to ensure consistency with the HERMES projections is small at -2.2%, implying that the ensemble prediction is corrected downwards by 2.2% on average. The median correction is only -0.8%. This implies that the predictions at the national level from HERMES, and those obtained at the regional level in this toolbox, using different methods and data, are close to each other on average. Yet, we see some outliers with major corrections.

The largest downward correction is -117.5% for sector MA (Legal and accounting activities; activities of head offices; management consultancy activities; architecture and engineering activities; technical testing and analysis), while the largest upwards correction is 38.7% for sector AA (Agriculture, forestry and fishing).

Variable	N	Mean	SD	Min	p25	p50	p75	Max
Correction (%)	37	-2.2	24.4	-117.5	-8.3	-0.8	7.9	38.7

Table 33: Correction from ensemble to final prediction (%).

10.4 Results: single best predictor

We also exploit the ensemble in an alternative way. In particular, we select the single best forecasting model for each individual sector-region, based on its validation NRMSE. Results are in `best_model_by_nrmse.csv`.

We first report the best performing models and variable transforms in Table 34. Across 111 sector-regions, the $ARIMA$ model turns out to be the best performing model for 46 out of 111 sector-regions, followed by random forests with 38 sector-regions. Looking at the best performing transforms, we see that the inverse and standardized transforms count 27 sector-regions each while the least represented transform is the square root with 15 sector-regions. Again, this variety underlines the usefulness of estimating multiple models for each sector-region.

Model	Inverse	Log	Raw	Sqrt	Stdized	Total
ARIMA	9	9	9	5	14	46
Panel	1	0	0	0	2	3
Random Forest	13	2	9	4	10	38
Spatial DR	1	2	3	2	0	8
Spatial TR	2	4	1	2	1	10
VAR/VEC	1	2	1	2	0	6
Total	27	19	23	15	27	111

Table 34: Best performing model predictions.

Next, Figure 27 shows the predictions for the single best predictor model for our three example sector-regions. In Table 35, we report the distribution of the correction in percentage terms from the ensemble prediction to the final prediction. It turns out that the corrections are smaller on average than in the full ensemble model while the median is larger (3.0% instead of -0.8%). Notice that outliers, especially the minimum, are now smaller than in the full ensemble model.

As an additional exercise, we also compare the rescaling factors across the two ensemble models across all sectors in Figure 28. We see that there are a few sectors for which scaling factors are very

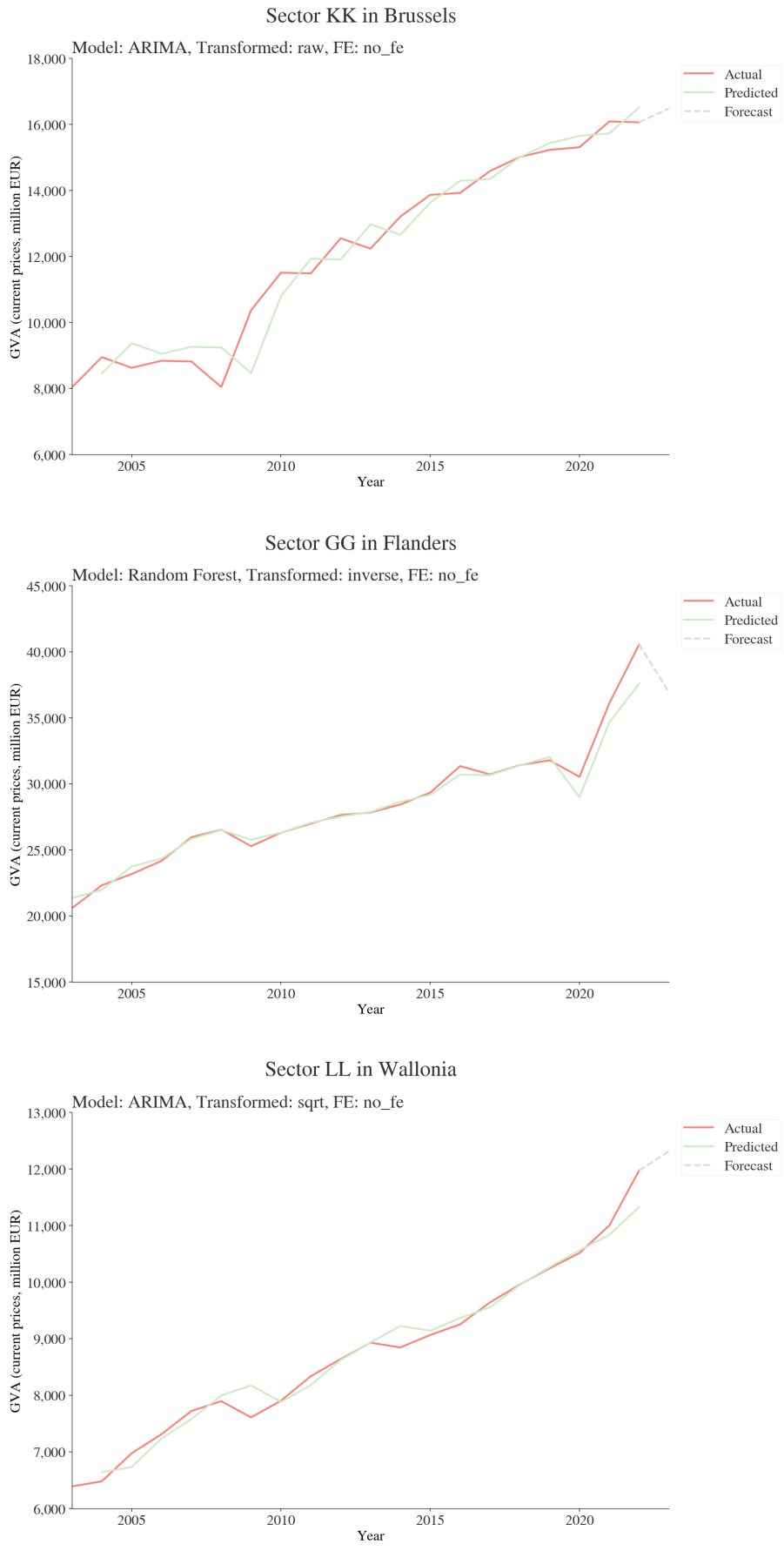


Figure 27: Actual, fitted, and forecast values for the best predicting model.

close to zero, and close to each other, such as OO (Public administration and defence; compulsory social security) and EE (Public administration and defence; compulsory social security). This implies that the weighted average ensemble and the single best predictor are close to each other, as well as close to the aggregate HERMES predictions. There are also sectors with significant differences, such as CD (Manufacture of coke and refined petroleum products) and MA (Legal and accounting activities; activities of head offices; management consultancy activities; architecture and engineering activities; technical testing and analysis). Note that, while the weighted average ensemble for MA requires a large correction, its correction in the single best predictor is much closer to zero.

Table 36 reports the non-rescaled and rescaled values for all sector-region predictions, the totals for each sector for Belgium, and the corresponding rescaling factor. In total across all sector-regions, we require a modest adjustment of 3.04% to align the ensemble prediction with the HERMES forecast. Sectors with a tiny rescaling factor are PP (Education) (0.21%), CM (Manufacture of furniture) (-0.77%), and OO (Public administration and defence) (0.95%). On the other hand, the worst performing prediction occurs for sector DD (Electricity and gas), for which predicted values are corrected downwards by -37.07%, followed by AA (Agriculture) (33.83%), and CL (Manufacture of transport equipment) (23.88%).

Variable	N	Mean	SD	Min	p25	p50	p75	Max
Correction (%)	37	0.8	12.4	-37.1	-6.2	3.0	7.3	33.8

Table 35: Correction from ensemble to final prediction (%).

Taken together, these results suggest that the single best predictor currently might be preferred over the weighted mean ensemble. Several factors might explain why the best-model approach currently performs better. First, individual models exhibit significant variation in performance. The mean ensemble, by averaging predictions, can be influenced by weaker models even if their weights are relatively low. In contrast, the best-model approach directly selects the model with the lowest validation error, avoiding this dilution effect. Second, RMSE-based weighting makes the mean ensemble sensitive to models with large errors in specific regions or sectors, which impact the overall ensemble error. The single best predictor approach avoids this by focusing only on the single best-performing model. However, it is possible that this approach increases variance compared to the weighted average ensemble. While the single best approach currently outperforms the mean ensemble, this does not imply that the mean ensemble is universally inferior; its performance could improve as new data becomes available. We allow for this flexibility in the construction of the toolbox.

10.5 Ensemble evaluation for future years

The last part of the ensemble involves creating persistent copies of files in the directory `/task10_ensemble/persistent` which include the forecasted values using the weighted and best model ensembles for each sector-region, both the non-rescaled and rescaled values are included in the files. An example of a file name used is `currentprices_best_model_ensemble_predictions_2023.csv`. When the actual data for 2023, the year T, become available and the pipeline is re-executed to obtain 2024 forecasts, this is $T + 1$ at the end of the ensemble, the code verifies the existence of

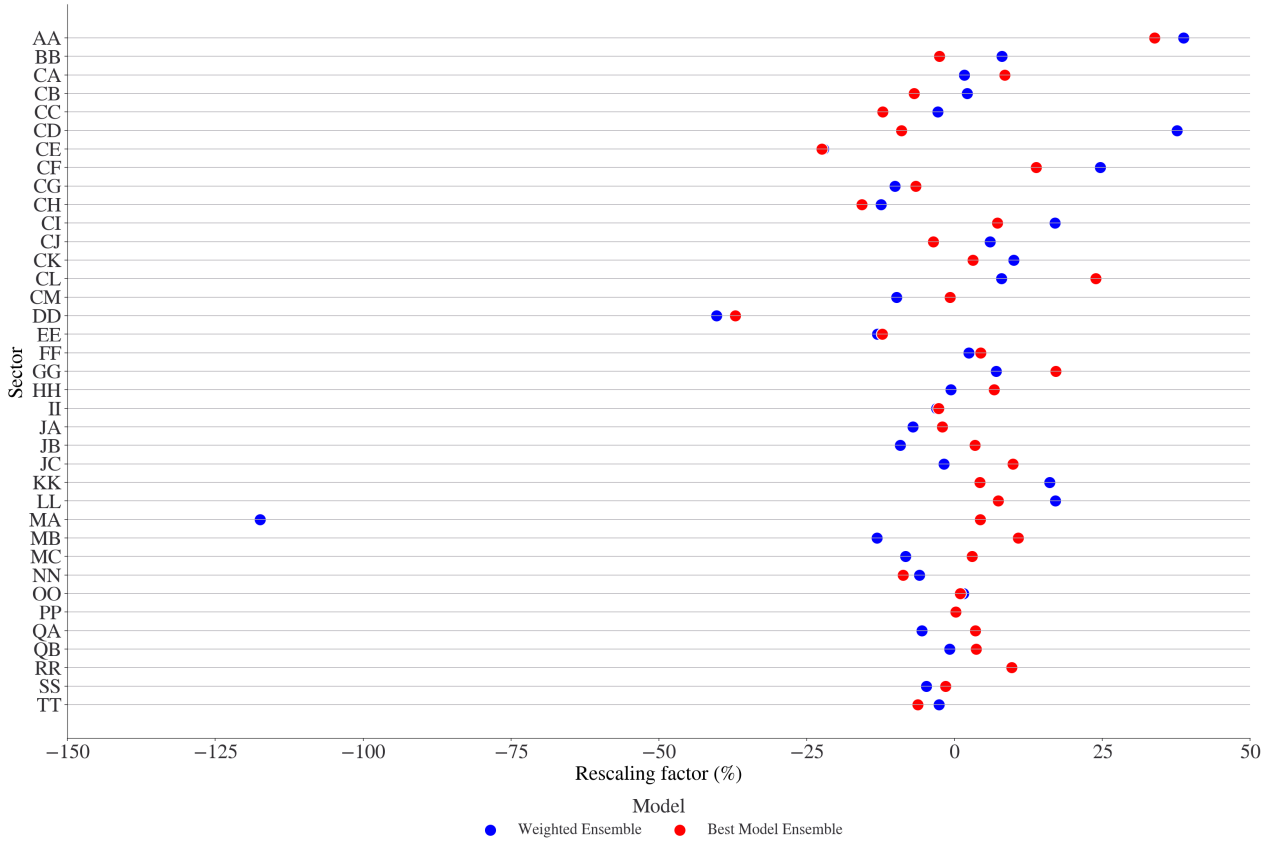


Figure 28: Rescaling factors by sector.

the corresponding persistent `.csv` file, both for T and $T+1$.[39] If the file exists, it will be loaded into memory to create a new file that contains for each sector-region the observed and ensemble values. For example, for the gross valued added in current prices, the file is saved to `/output/gva_-currentprices/csv/observed_vs_ensemble_comparison_2023.csv`. This allows the users to compare the performance of the ensemble forecasts to the realized values of gross value added.

[39]The code checks for both T and $T+1$ so that the file to compare observed to ensemble values is created even if a user executes the pipeline to forecast values for a year in which values are already observed (for example, 2022 with current data availability).

	Not Rescaled				Rescaled				
Sector	**Brussels**	**Flanders**	**Wallonia**	**Total**	**Brussels**	**Flanders**	**Wallonia**	**Total**	**Rescaling (%)**
AA	6.60	2,494.49	869.29	3,370.38	8.83	3,338.27	1,163.34	4,510.44	33.83
BB	10.20	86.30	201.71	298.21	9.94	84.09	196.54	290.56	-2.56
CA	405.02	7,057.03	2,143.62	9,605.67	439.40	7,656.11	2,325.59	10,421.10	8.49
CB	15.31	1,184.84	154.85	1,355.00	14.27	1,103.96	144.28	1,262.51	-6.83
CC	86.10	2,684.13	792.25	3,562.48	75.62	2,357.48	695.84	3,128.94	-12.17
CD	236.33	3,351.04	37.52	3,624.89	215.17	3,050.98	34.16	3,300.31	-8.95
CE	272.18	8,966.87	1,663.73	10,902.78	211.20	6,957.92	1,290.99	8,460.11	-22.40
CF	221.72	7,278.29	5,439.16	12,939.16	252.36	8,284.15	6,190.85	14,727.36	13.82
CG	40.67	3,703.18	1,963.61	5,707.46	38.01	3,460.95	1,835.16	5,334.12	-6.54
CH	126.25	7,115.47	1,927.57	9,169.29	106.50	6,002.29	1,626.01	7,734.80	-15.64
CI	18.68	1,141.13	404.64	1,564.44	20.03	1,224.01	434.02	1,678.07	7.26
CJ	9.67	800.10	604.77	1,414.54	9.33	771.48	583.14	1,363.95	-3.58
CK	60.13	3,981.65	423.25	4,465.03	62.02	4,106.78	436.55	4,605.35	3.14
CL	464.37	1,918.12	579.78	2,962.26	575.27	2,376.22	718.25	3,669.74	23.88
CM	184.78	2,915.45	691.11	3,791.34	183.35	2,892.94	685.77	3,762.06	-0.77
DD	1,921.22	4,480.45	2,478.32	8,879.99	1,208.94	2,819.36	1,559.50	5,587.80	-37.07
EE	934.28	2,598.45	1,582.00	5,114.73	820.38	2,281.68	1,389.14	4,491.20	-12.19
FF	1,892.87	19,112.33	5,907.87	26,913.07	1,976.80	19,959.78	6,169.83	28,106.41	4.43
GG	6,099.59	36,937.18	10,747.43	53,784.20	7,146.27	43,275.54	12,591.67	63,013.48	17.16
HH	5,055.17	15,399.77	5,583.28	26,038.23	5,394.60	16,433.79	5,958.17	27,786.56	6.71
II	2,441.59	5,831.74	2,495.25	10,768.58	2,375.50	5,673.89	2,427.71	10,477.11	-2.71
JA	1,159.67	1,807.92	443.76	3,411.35	1,135.83	1,770.75	434.63	3,341.21	-2.06
JB	2,617.47	2,268.20	518.02	5,403.68	2,707.93	2,346.59	535.92	5,590.44	3.46
JC	2,831.57	9,109.84	1,829.81	13,771.22	3,110.34	10,006.71	2,009.96	15,127.01	9.85
KK	16,489.02	10,244.44	4,226.86	30,960.32	17,197.21	10,684.43	4,408.40	32,290.04	4.29
LL	6,295.57	28,313.32	12,317.94	46,926.83	6,760.07	30,402.36	13,226.79	50,389.22	7.38
MA	8,236.26	31,396.77	8,174.17	47,807.20	8,597.64	32,774.35	8,532.82	49,904.81	4.39
MB	156.39	835.47	724.99	1,716.85	173.27	925.68	803.27	1,902.22	10.80
MC	732.48	2,394.90	541.38	3,668.76	754.23	2,466.00	557.46	3,777.69	2.97
NN	4,324.11	18,935.60	6,070.11	29,329.83	3,947.45	17,286.20	5,541.37	26,775.02	-8.71
OO	12,823.12	15,507.08	11,090.21	39,420.41	12,944.50	15,653.86	11,195.19	39,793.55	0.95
PP	5,578.88	19,904.02	11,578.59	37,061.49	5,590.32	19,944.85	11,602.34	37,137.51	0.21
QA	2,904.61	11,854.14	6,378.72	21,137.48	3,008.07	12,276.37	6,605.92	21,890.36	3.56
QB	1,377.40	7,775.64	3,781.62	12,934.65	1,427.57	8,058.90	3,919.38	13,405.86	3.64
RR	1,041.30	1,652.36	1,017.24	3,710.90	1,141.80	1,811.83	1,115.41	4,069.04	9.65
SS	1,714.83	2,917.38	1,374.87	6,007.09	1,689.33	2,873.99	1,354.42	5,917.74	-1.49
TT	82.69	468.03	63.14	613.86	77.53	438.82	59.20	575.54	-6.24
Total	88,868.10	304,423.15	116,822.42	510,113.66	91,406.89	313,833.35	120,358.98	525,599.22	3.04

Table 36: Best performing model predictions by sector.

11 Predicted gross value added for the next year

In this last section, we present and discuss the final and consistent predictions for gross value added across all sector-regions. We focus on the single best predictor results, and show results for predicted values in both levels and growth rates predictions, current and chained prices.

11.1 Predicted values in levels

We start with the predicted values of gross value added in levels in Table 37. The table reports the predicted values for each sector-region in both current prices and chained prices, as well as the totals for Belgium and by region. Total gross value added for Belgium in 2023 is expected to be 525,599 million EUR in terms of current prices and 424,381 million EUR in chained prices. At the regional level, Flanders is expected to contribute 60% to Belgian GDP (313,833 million EUR), followed by Wallonia with 23% (120,359 million EUR) and Brussels with 17% (91,407 million EUR). These proportions remain similar in chained prices.

The largest sector in Belgium is GG (*Wholesale and retail trade*), generating 63,013 million EUR in current prices and 49,284 million EUR in chained prices. In Brussels, sector KK (*Financial and insurance activities*) is the largest sector, with 17,197 million EUR (current prices) and 13,271 million EUR (chained prices). Other large sectors in Brussels are OO (*Public administration and defence; compulsory social security*) with 12,945 million EUR and 10,498 million EUR in current and chained prices, respectively, as well as MA (*Legal and accounting activities*) with 8,598 million EUR and 9,609 million EUR. Together, these results show the importance of Brussels as a financial and administrative hub. In Flanders, Sector GG (*Wholesale and retail trade*) dominates, reaching 43,275 million EUR (current prices) and 34,400 million EUR (chained prices). This is followed by MA (*Legal and accounting activities*) for 32,774 million EUR (current prices) and 26,906 million EUR (chained prices), and LL (*Real estate activities*) with 30,402 million EUR and 25,144 million EUR, respectively. In Wallonia, sector LL (*Real estate activities*) is the largest sector, with 13,227 million EUR (current prices) and 11,261 million EUR (chained prices). This is closely followed by sector GG (*Wholesale and retail trade*) for 12,592 million EUR and 9,879 million EUR, and sector PP (*Education*) with 11,602 million EUR and 8,549 million EUR. The sectors that are predicted to be the largest in each region are the same as in the last few years in the data. When comparing current and chained prices, the rankings are very similar across both, with some small changes for sectors that are close to each other in terms of gross value added. The largest relative declines are in *Agriculture* (AA), *Construction* (FF), *Transportation and storage* (HH), *Financial services* (KK), and *Public administration* (OO), suggesting that these sectors experience significant inflationary effects over time.

There are some sizable differences across regions for the same sector. For example, the *Agricultural sector* (AA) is almost three times larger in Flanders than it is in Wallonia, while it is almost non-existent in Brussels. Conversely, *Mining and quarrying* (BB) is larger in Wallonia than it is in Brussels or Flanders. All manufacturing sectors (CA to CM) are larger in Flanders than they are in Wallonia or Brussels. Brussels has the largest *Telecommunications* sector (JB), as well as for *Financial and insurance activities* (KK). All results are in terms of gross output, i.e. not corrected for population.

	Current prices				Chained prices			
Sector	**Brussels**	**Flanders**	**Wallonia**	**Total**	**Brussels**	**Flanders**	**Wallonia**	**Total**
AA	8.83	3,338.27	1,163.34	4,510.44	9.24	1,704.15	560.24	2,273.63
BB	9.94	84.09	196.54	290.56	8.41	79.65	139.90	227.95
CA	439.40	7,656.11	2,325.59	10,421.10	359.78	6,175.32	1,773.74	8,308.84
CB	14.27	1,103.96	144.28	1,262.51	11.76	773.79	104.59	890.14
CC	75.62	2,357.48	695.84	3,128.94	52.34	1,672.77	466.38	2,191.50
CD	215.17	3,050.98	34.16	3,300.31	252.70	3,369.32	50.55	3,672.56
CE	211.20	6,957.92	1,290.99	8,460.11	236.93	5,585.74	863.37	6,686.04
CF	252.36	8,284.15	6,190.85	14,727.36	189.64	6,656.96	6,261.90	13,108.50
CG	38.01	3,460.95	1,835.16	5,334.12	30.93	2,517.12	1,183.86	3,731.91
CH	106.50	6,002.29	1,626.01	7,734.80	67.17	3,856.71	1,046.49	4,970.37
CI	20.03	1,224.01	434.02	1,678.07	19.14	1,187.84	407.27	1,614.26
CJ	9.33	771.48	583.14	1,363.95	9.22	697.18	360.93	1,067.33
CK	62.02	4,106.78	436.55	4,605.35	62.84	3,467.61	539.85	4,070.30
CL	575.27	2,376.22	718.25	3,669.74	551.49	2,074.01	864.24	3,489.73
CM	183.35	2,892.94	685.77	3,762.06	169.25	2,420.02	547.35	3,136.62
DD	1,208.94	2,819.36	1,559.50	5,587.80	637.32	1,237.16	628.71	2,503.19
EE	820.38	2,281.68	1,389.14	4,491.20	464.32	2,476.97	1,247.51	4,188.80
FF	1,976.80	19,959.78	6,169.83	28,106.41	1,473.59	14,707.45	4,766.69	20,947.72
GG	7,146.27	43,275.54	12,591.67	63,013.48	5,005.77	34,399.57	9,878.92	49,284.26
HH	5,394.60	16,433.79	5,958.17	27,786.56	4,104.87	12,735.55	4,175.74	21,016.16
II	2,375.50	5,673.89	2,427.71	10,477.11	1,982.54	4,168.70	1,366.28	7,517.52
JA	1,135.83	1,770.75	434.63	3,341.21	1,040.75	1,323.55	351.12	2,715.42
JB	2,707.93	2,346.59	535.92	5,590.44	3,380.21	2,882.71	671.62	6,934.54
JC	3,110.34	10,006.71	2,009.96	15,127.01	2,529.36	8,456.01	1,802.77	12,788.15
KK	17,197.21	10,684.43	4,408.40	32,290.04	13,270.90	7,860.11	2,679.45	23,810.45
LL	6,760.07	30,402.36	13,226.79	50,389.22	5,623.99	25,143.80	11,260.53	42,028.32
MA	8,597.64	32,774.35	8,532.82	49,904.81	9,608.78	26,906.05	7,091.81	43,606.63
MB	173.27	925.68	803.27	1,902.22	140.46	720.70	670.85	1,532.01
MC	754.23	2,466.00	557.46	3,777.69	735.42	2,429.92	533.09	3,698.43
NN	3,947.45	17,286.20	5,541.37	26,775.02	3,666.30	14,587.17	4,093.53	22,347.01
OO	12,944.50	15,653.86	11,195.19	39,793.55	10,497.93	12,297.01	9,222.72	32,017.66
PP	5,590.32	19,944.85	11,602.34	37,137.51	4,231.61	14,925.06	8,548.82	27,705.49
QA	3,008.07	12,276.37	6,605.92	21,890.36	2,396.42	11,790.73	5,698.67	19,885.82
QB	1,427.57	8,058.90	3,919.38	13,405.86	1,089.39	7,353.96	2,979.50	11,422.86
RR	1,141.80	1,811.83	1,115.41	4,069.04	772.41	1,647.93	880.66	3,301.00
SS	1,689.33	2,873.99	1,354.42	5,917.74	1,774.60	2,269.09	1,144.83	5,188.51
TT	77.53	438.82	59.20	575.54	73.76	376.28	50.92	500.95
Total	91,406.89	313,833.35	120,358.98	525,599.22	76,531.53	252,933.66	94,915.40	424,380.59

Table 37: Predicted values across sector-regions, levels.

11.2 Predicted growth rates

Next, we turn to predicted growth rates. Table 38 reports the predicted annual percentage changes for each sector-region in both current and chained prices, as well as the total growth rates for each region. Growth rates are predicted to be 5.53% in Brussels, 5.79% in Flanders, and 5.81% in Wallonia in current prices. In chained prices, growth is more moderate at 3.06% in Brussels, 1.85% in Flanders, and even a slight decline of 0.24% in Wallonia.

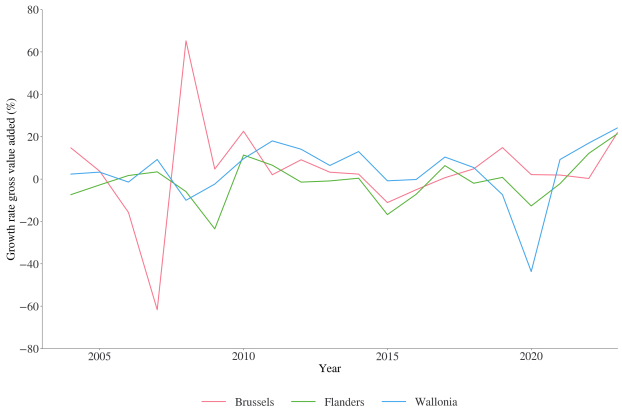
While aggregate growth rates are plausible, growth rates of individual sector-regions can be large, either positive or negative. We therefore also provide graphs for all sectors across the three regions in the `/plots` folder for growth rates in current prices. Each graph contains the full time series of both realized and predicted growth rates from 2004 up to 2023. This allows us to evaluate whether large swings are potential anomalies in prediction, or whether they are intrinsically part of the evolution of these sectors over time for some volatile sectors. In the report, we discuss the most growing and most contracting sector-regions, as well as heterogeneity in growth rates for the same sector across the three regions. We also relate the predicted growth rates for 2023 to these sector-region historical growth rates. Users can validate the results for all sector-regions, as well as chained prices, in the toolbox.

In Brussels, the sectors that are expected to grow most from 2022 to 2023 are sector II (*Accommodation and food service activities*) (57.43%), sector EE (*Water supply*) (51.19%), and sector CL (*Manufacture of transport equipment*) (21.98%). We also compare their previous and predicted growth rates in Figure 29 to evaluate whether these large growth numbers are plausible given their past evolutions. The *Accommodation and food service activities* (sector II) has seen a massive drop across all three regions during Covid-19, with drops up to 50% of their value added. Afterwards, this sector recovered dramatically across all regions in 2021 and 2022. The predicted growth rates for Brussels suggest that this sector will continue to grow rapidly, while the growth rates for the other two regions is returning to the long run average. Turning to sector EE (*Water supply*), the predicted growth is much higher in Brussels than in the other two regions. At the same time, it seems that this sector has a history of being quite volatile, especially in Brussels. Finally, growth of *Manufacture of transport equipment* (sector CL) is large in Brussels, albeit very much in line with expected growth rates for the other two regions. Here too, Brussels has a history of quite volatile growth rates. Note that both EE and CL are relatively small sectors in Brussels, which might explain large fluctuations in growth rates over time, as modest changes in EUR can have large effects on growth rates, inducing more volatility. Taken together, perhaps the large growth for sector EE can be labeled as excessive and much less correlated with expected growth rates of the same sector in the other regions.

At the other end of the spectrum, several sectors are expected to contract. The most significant declines are seen in sector CD (*Manufacture of coke and refined petroleum products*) (-61.48%), sector DD (*Electricity, gas, steam and air-conditioning supply*) (-44.54%), and sector AA (*Agriculture, forestry and fishing*) (-22.56%). Figure 30 shows the previous and predicted growth rates for these sectors across regions. Sector CD (*Manufacture of coke and refined petroleum products*) is well known to be difficult to predict, and even more so for Brussels, with massive realized fluctuations in the past. Again, this is a small sector, with additional particularities regarding allocating flows to particular regions in an accounting way. Next, sector DD (*Electricity, gas, steam and air-conditioning supply*) can fluctuate significantly over time, due to e.g. large swings in international energy prices, as recently witnessed during the Russian-Ukraine war in 2022. Moreover, historically, this sector is more volatile in Brussels, compared to the other regions. Nevertheless, all 3 regions expect a large drop in value added for this

(a) Sector II (Accommodation and food service activities).

(b) Sector EE (Water supply).



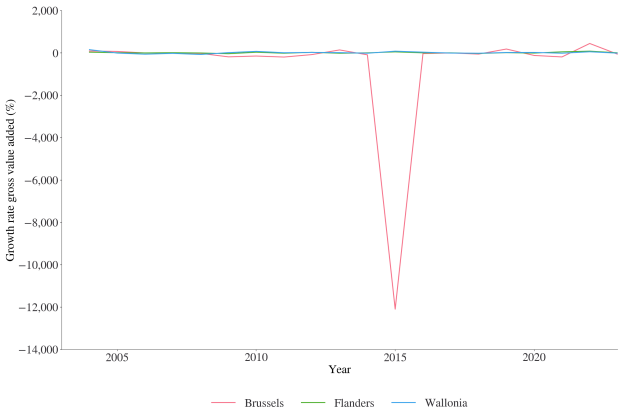
(c) Sector CL (Manufacture of transport equipment).

Figure 29: Growth rates fastest growing sectors in Brussels (2004-2023).

sector in 2023. Finally, sector AA (*Agriculture, forestry and fishing*) is again a tiny sector in Brussels, such that small nominal changes can generate large growth rate volatility for this sector. All in all, these large negative growth rates seem consistent with either tiny sectors in Brussels, or expected large macro-economic downturns across the three regions.

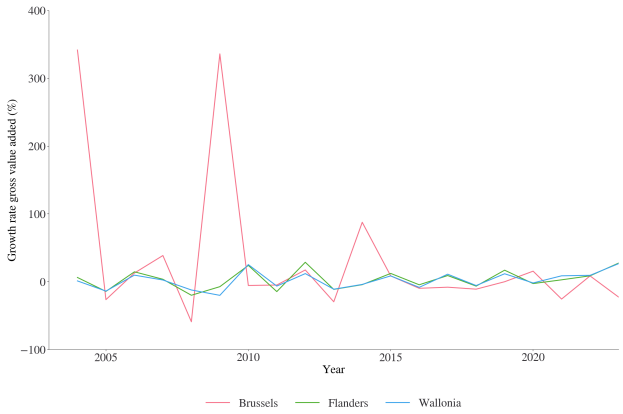
We can perform the same exercise for Flanders, and we show the top 3 sectors in Figure 31. The sectors that are expected to grow most are sector AA (*Agriculture, forestry and fishing*) (27.46%), CL (*Manufacture of transport equipment*) (21.42%), and JC (*Computer programming, consultancy and related activities; information service activities*) (20.71%). Interestingly, CL was also among the largest growth sectors for Brussels, showing significant growth for all three regions. Also AA was mentioned for Brussels, albeit as one of the largest decline sectors. In Flanders, *Agriculture* is expected to keep growing. New is JC (*Computer programming, consultancy and related activities; information service activities*, where we see a large increase for Flanders, which is still positive but more modest in the other two regions.

The top 3 largest decline sectors in Flanders are shown in Figure 32: DD (*Electricity, gas, steam and air-conditioning supply*) (-44.74%), CE (*Manufacture of chemicals and chemical products*) (-20.01%), and BB (*Mining and quarrying*) (-12.95%). We already saw *Electricity, gas, steam and air-conditioning supply* as



(a) Sector CD (Manufacture of coke and refined petroleum products).

(b) Sector DD (Electricity, gas, steam and air-conditioning supply).



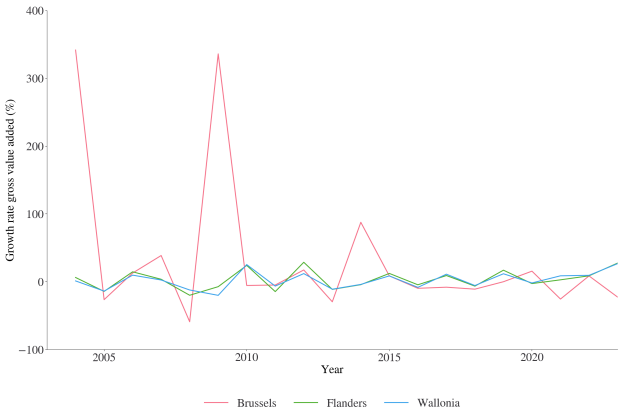
(c) Sector AA (Agriculture, forestry and fishing).

Figure 30: Growth rates fastest declining sectors in Brussels (2004-2023).

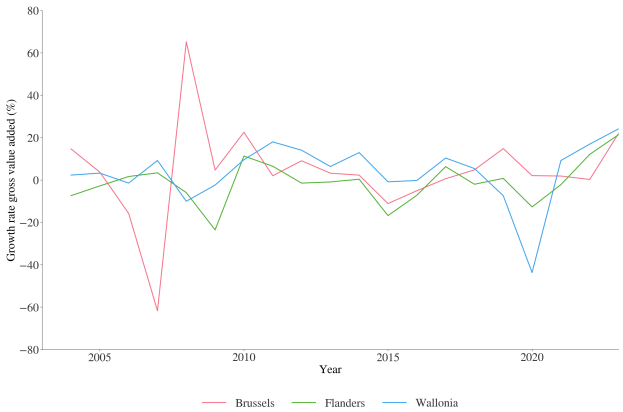
one of the large decline sectors for Brussels, driven by macro-economic influences. Also *Manufacture of chemicals and chemical products* is expected to decline across the board. Finally, *Mining and quarrying* is again very small and volatile in general, and even more so in Flanders.

We conclude with the same exercise for Wallonia. The top 3 growth sectors are shown in Figure 33. The sectors that are expected to grow most are sector CJ (*Manufacture of electrical equipment*) (47.52%), AA (*Agriculture, forestry and fishing*) (26.64%), and CL (*Manufacture of transport equipment*) (24.14%). While *Manufacture of electrical equipment* is expected to decline in the other two regions, there is a large uptick expected for Wallonia. Again AA (*Agriculture*) shows up in the top or bottom three sectors across the regions, while *Manufacture of transport equipment* was also among the expected top performers for Flanders.

On the negative side, the expected bottom three sectors for Wallonia are shown in Figure 34: DD (*Electricity, gas, steam and air-conditioning supply*) (-43.85%), CE (*Manufacture of chemicals and chemical products*) (-21.35%), and CH (*Manufacture of basic metals and fabricated metal products, except machinery and equipment*) (-20.38%). We already saw *Electricity, gas, steam and air-conditioning supply* as one of the large decline sectors the other two regions. Also *Manufacture of chemicals and chemical products* is expected to decline across the board. Finally, *Manufacture of basic metals and fabricated metal products,*



(a) Sector AA (Agriculture, forestry and fishing).



(b) Sector CL (Manufacture of transport equipment).

(c) Sector JC (Computer programming, consultancy and related activities; information service activities).

Figure 31: Growth rates fastest growing sectors in Flanders (2004-2023).

except machinery and equipment is expected to drop across the three regions, albeit most in Wallonia.

(a) Sector DD (Electricity, gas, steam and air-conditioning supply).

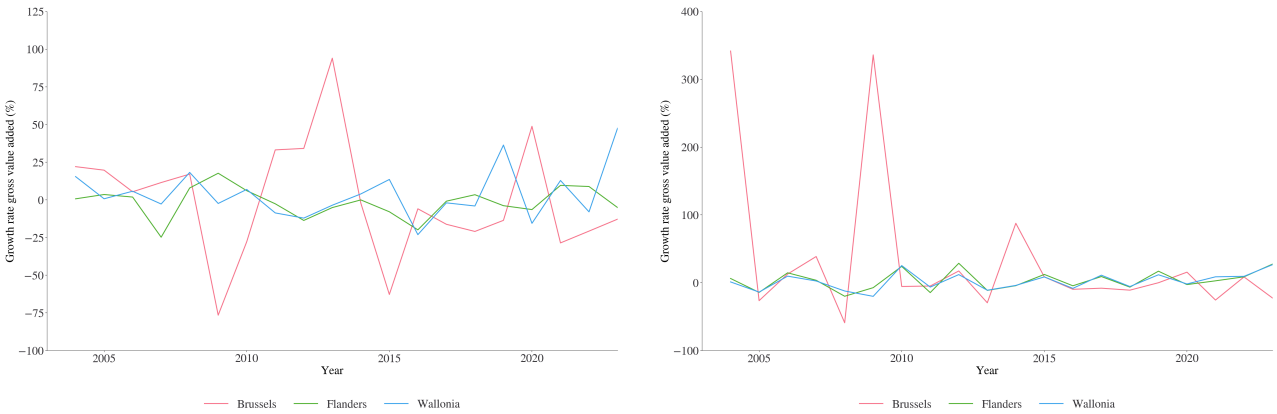
(b) Sector CE (Manufacture of chemicals and chemical products).

(c) Sector BB (Mining and quarrying).

Figure 32: Growth rates fastest declining sectors in Flanders (2004-2023).

	Current prices (%)			Chained prices (%)		
Sector	Brussels	Flanders	Wallonia	Brussels	Flanders	Wallonia
AA	-22.56	27.46	26.64	12.70	4.47	-7.23
BB	14.23	-12.95	-3.75	12.07	-4.96	-21.05
CA	4.94	6.74	7.24	-2.68	-2.51	-7.38
CB	-18.93	-0.45	-8.57	-11.58	-7.39	-12.03
CC	-16.07	-2.22	-19.93	-18.47	-1.00	-20.29
CD	-61.48	3.85	-12.40	-53.80	17.11	32.33
CE	-15.18	-20.01	-21.35	29.61	-12.54	-28.36
CF	11.46	15.01	3.37	-3.19	6.83	20.86
CG	2.73	-5.08	-11.29	8.54	-8.94	-25.39
CH	-17.63	-10.06	-20.38	-11.51	-0.80	-12.19
CI	4.89	6.61	18.94	0.75	3.83	12.01
CJ	-12.82	-5.10	47.52	0.26	-0.10	6.34
CK	-0.28	15.28	-11.16	6.88	2.95	16.20
CL	21.98	21.42	24.14	16.52	6.07	34.41
CM	16.49	2.21	5.05	22.73	0.40	-2.81
DD	-44.54	-44.74	-43.85	-19.70	-33.40	-37.83
EE	51.19	-9.49	5.32	-2.35	0.50	-0.76
FF	1.61	11.43	-3.64	-3.11	5.02	-4.77
GG	18.76	6.67	13.16	1.43	3.54	7.63
HH	5.60	-0.77	8.67	-3.54	-3.64	-8.17
II	57.43	7.35	20.31	69.49	1.75	-12.65
JA	-4.37	11.09	3.73	3.72	-2.04	-2.76
JB	2.35	5.41	1.85	2.25	3.64	2.15
JC	8.14	20.71	2.30	-1.41	14.35	2.86
KK	7.04	7.52	20.94	-0.57	-2.16	-2.94
LL	10.34	11.42	10.44	3.70	4.09	6.21
MA	-0.16	10.10	8.23	21.09	-1.87	-2.29
MB	9.25	3.45	2.92	3.13	-6.23	0.05
MC	3.01	8.46	5.86	-1.23	7.83	4.02
NN	-5.72	8.42	7.67	-1.53	4.53	-9.89
OO	9.41	7.42	5.36	2.96	1.60	0.63
PP	3.85	7.36	9.14	1.15	0.59	0.13
QA	7.36	0.19	7.75	-7.17	4.44	0.88
QB	9.62	8.28	10.39	-6.20	10.80	-5.90
RR	21.78	6.10	23.51	-4.57	12.93	13.57
SS	-12.00	9.33	6.31	2.42	-3.90	0.02
TT	2.96	2.53	4.41	10.41	-0.93	1.23
Total	5.53	5.79	5.81	3.06	1.85	-0.24

Table 38: Predicted values across sector-regions, growth rates.



(a) Sector CJ (Manufacture of electrical equipment).

(b) Sector AA (Agiculture, forestry and fishing).

(c) Sector CL (Manufacture of transport equipment).

Figure 33: Growth rates fastest growing sectors in Wallonia (2004-2023).

(a) Sector DD (Electricity, gas, steam and air-conditioning supply).

(b) Sector CE (Manufacture of chemicals and chemical products).

(c) Sector CH (Manufacture of basic metals and fabricated metal products, except machinery and equipment).

Figure 34: Growth rates fastest declining sectors in Wallonia (2004-2023).

12 Conclusion

This report develops a structured methodology to forecast sector-regional gross value added for Belgian sector-regions for the most recent year that is not yet available in the data. The framework integrates a diverse range of datasets and multiple econometric and machine learning models to maximize predictive accuracy while being consistent with national-level projections.

We first collect and describe several datasets, including regional accounts, VAT statistics, confidence indices, construction permits, labor market data, and input-output linkages. These datasets cover varying levels of granularity and reporting frequencies, providing a robust and multi-dimensional foundation for the analysis. The estimation process includes five classes of models: univariate time series (ARIMA), multivariate time series (VAR/VEC), panel fixed effects models, spatial autocorrelation models using input-output relationships, and machine learning classifiers (random forests). Each model captures specific dimensions of the data, such as temporal dynamics, cross-sectional heterogeneity, spatial dependencies, and/or non-linear relationships.

The validation results highlight important insights into the ensemble model performance. We provide two versions of an ensemble model: a weighted average prediction and a single best predictor model. The weighted ensemble approach, which combines predictions as a weighted sum, with weights derived from out-of-sample performance, should provide a robust and balanced prediction framework. This approach also accounts for model-specific strengths while mitigating potential overfitting or limitations of individual methods. However, single predictors often provide superior predictive accuracy for individual sector-regions, with often smaller corrections to match the national aggregates, making them essential components of the toolbox.

These findings emphasize the value of combining diverse methodologies. While ARIMA models excel in capturing time series dynamics and random forests perform well in exploiting complex, non-linear patterns, other models such as spatial autocorrelation and panel fixed effects add important perspectives by leveraging interdependencies across sector-regions. The ensemble integrates these contributions, providing adaptability to the existing data limitations.

This methodology is designed to be iterative, with future applications benefiting from expanded datasets and refinements in model weighting as new data becomes available. The flexible framework ensures that sector-regional estimates remain accurate, consistent, and aligned with national objectives, offering a reliable tool for economic planning and policy evaluation.

References

Avonds, L., Hertveldt, B., and Van den Cruyce, B. (2021). Opmaak van de interregionale input-outputtabel voor het jaar 2015: databronnen en methodologie. Working Paper 7-21, Federal Planning Bureau.

Bassilière, D., Bossier, F., Caruso, F., Hendrickx, K., Hoorelbeke, D., and Lohest, O. (2008). Uitwerking van een regionaal projectiemodel. Technical report.

Hastie, T., Tibshirani, R., and Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer, second edition edition.

Hoorelbeke, D., Lohest, O., Bassilière, D., Bossier, F., Bracke, I., and Caruso, F. (2007). Hermreg: A regionalisation model for belgium. Technical report.

James, G., Witten, D., Hastie, T., and Tibshirani, R. (2021). *An Introduction to Statistical Learning*. Springer, New York, NY, second edition edition. Available for free at the book's website.

Johansen, S. (1995). *Likelihood-Based Inference in Cointegrated Vector Autoregressive Models*. Oxford University Press, Oxford.

Lütkepohl, H. (2005). *New Introduction to Multiple Time Series Analysis*. Springer, Berlin, Heidelberg.

MacKinnon, J. G. and White, H. (1985). Some heteroskedasticity-consistent covariance matrix estimators with improved finite sample properties. *Journal of Econometrics*, 29(3):305–325.

A Sector classifications and correspondences

Table A1: A38 Sectors and their descriptions

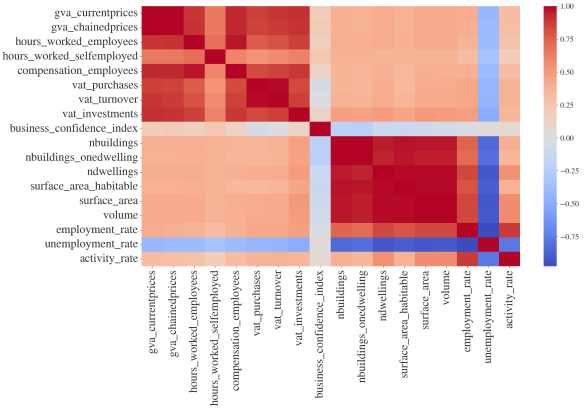
A38 Sector	A38 Description
AA	Agriculture, forestry and fishing.
BB	Mining and quarrying.
CA	Manufacture of food products, beverages and tobacco products.
CB	Manufacture of textiles, wearing apparel and leather products.
CC	Manufacture of wood and paper products, and printing.
CD	Manufacture of coke and refined petroleum products.
CE	Manufacture of chemicals and chemical products.
CF	Manufacture of basic pharmaceutical products and pharmaceutical preparations.
CG	Manufacture of rubber and plastics products, and other non-metallic mineral products.
CH	Manufacture of basic metals and fabricated metal products, except machinery and equipment.
CI	Manufacture of computer, electronic and optical products.
CJ	Manufacture of electrical equipment.
CK	Manufacture of machinery and equipment n.e.c.
CL	Manufacture of transport equipment.
CM	Manufacture of furniture; other manufacturing; repair and installation of machinery and equipment.
DD	Electricity, gas, steam and air-conditioning supply.
EE	Water supply; sewerage, waste management and remediation activities.
FF	Construction.
GG	Wholesale and retail trade, repair of motor vehicles and motorcycles.
HH	Transportation and storage.
II	Accommodation and food service activities.
JA	Publishing, audiovisual and broadcasting activities.
JB	Telecommunications.
JC	Computer programming, consultancy and related activities; information service activities.
KK	Financial and insurance activities.
LL	Real estate activities.
MA	Legal and accounting activities; activities of head offices; management consultancy activities; architecture and engineering activities; technical testing and analysis.
MB	Scientific research and development.
MC	Advertising and market research; other professional, scientific and technical activities; veterinary activities.
NN	Administrative and support service activities.
OO	Public administration and defence; compulsory social security.
PP	Education.
QA	Human health activities.
QB	Social work activities.
RR	Arts, entertainment and recreation.
SS	Other service activities.
TT	Activities of households as employers of domestic personnel and undifferentiated goods and services production of households for own use.

Table A2: Correspondences of sector classifications to NACE A38

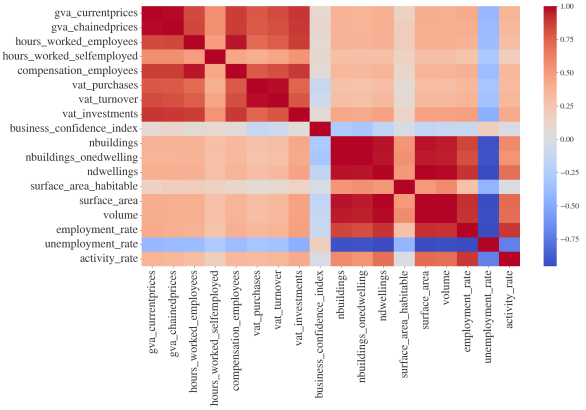
A38 Sector	Aggregate Industries	Business Confidence	NACE A64	Industrial Production
AA	Primary/extraction		01 to 03	
BB	Primary/extraction		05 to 09	B
CA	Manufacturing	Manufacturing	10 to 12	C10 to C12
CB	Manufacturing	Manufacturing	13 to 15	C13 to C15
CC	Manufacturing	Manufacturing	16 to 18	C16 to C18
CD	Manufacturing	Manufacturing	19	C19
CE	Manufacturing	Manufacturing	20	C20
CF	Manufacturing	Manufacturing	21	C21
CG	Manufacturing	Manufacturing	22 to 23	C22 to C23
CH	Manufacturing	Manufacturing	24 to 25	C24 to C25
CI	Manufacturing	Manufacturing	26	C26
CJ	Manufacturing	Manufacturing	27	C27
CK	Manufacturing	Manufacturing	28	C28
CL	Manufacturing	Manufacturing	29 to 30	C29 to C30
CM	Manufacturing	Manufacturing	31 to 33	
DD	Services	Services	35	D35
EE	Services	Services	36 to 39	
FF	Manufacturing	Construction	41 to 43	
GG	Services	wholesale/retail	45 to 47	
HH	Services	Services	49 to 53	
II	Services	Services	55 to 56	
JA	Services	Services	58 to 60	
JB	Services	Services	61	
JC	Services	Services	62 to 63	
KK	Services	Services	64 to 66	
LL	Services	Services	68	
MA	Services	Services	69 to 71	
MB	Services	Services	72	
MC	Services	Services	73 to 75	
NN	Services	Services	77 to 82	
OO	Non-market services		84	
PP	Non-market services		85	
QA	Non-market services		86	
QB	Non-market services		87 to 88	
RR	Non-market services		90 to 93	
SS	Non-market services		94 to 96	
TT	Non-market services		97 to 98	

B Additional descriptive statistics

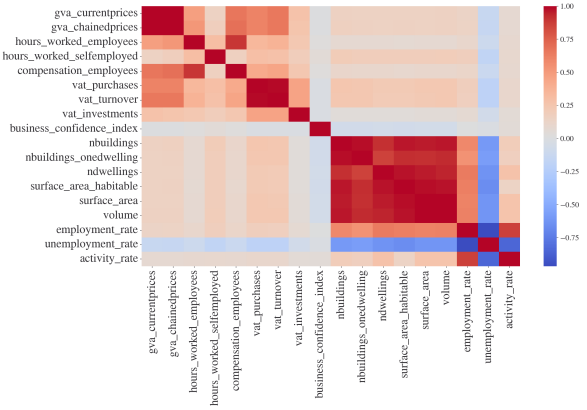
The following graphs show the correlation matrices for all numeric variables, both in levels (Figure B1) and growth rates (Figure B2), for each of the transforms described in Section 4. For the transforms in levels, the log transform provides a high covariance across all variables, while the inverse transform generates more low values. In terms of growth rates, there is much less co-movement across variables, except for similar variables within datasets (e.g. construction permits or (un)employment rates).



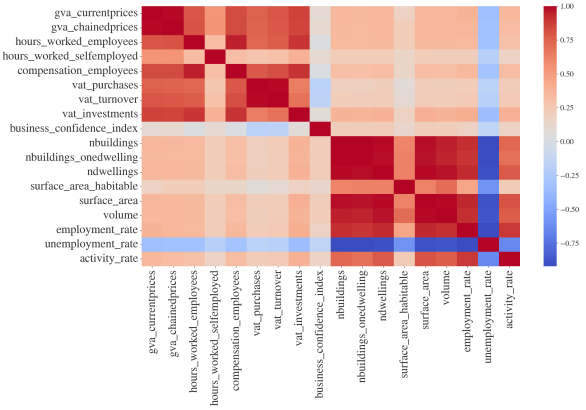
(a) Correlation matrix (levels, logs).



(b) Correlation matrix (levels, square root).

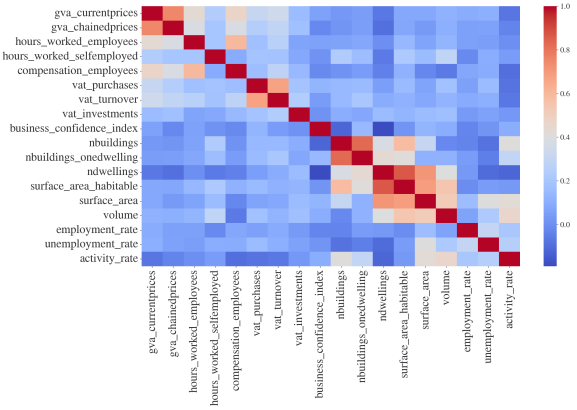


(c) Correlation matrix (levels, inverse).

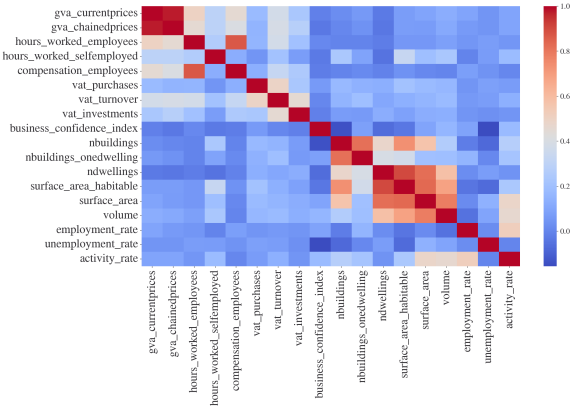


(d) Correlation matrix (levels, standardized).

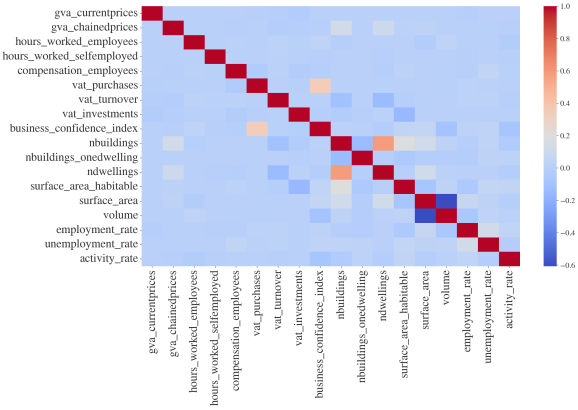
Figure B1: Correlation matrices for the different transformations (levels).



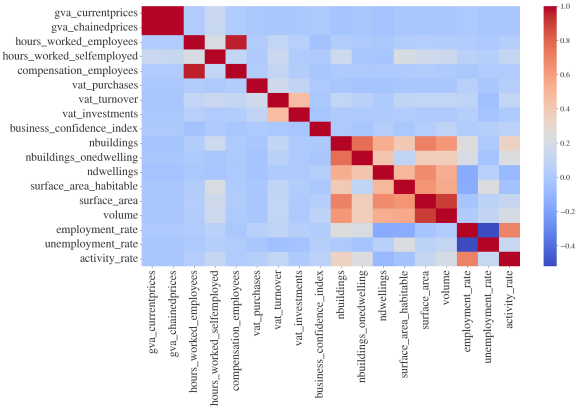
(a) Correlation matrix (growth, logs).



(b) Correlation matrix (growth, square root).



(c) Correlation matrix (growth, inverse).



(d) Correlation matrix (growth, standardized).

Figure B2: Correlation matrices for the different transformations (growth rates).

We also provide a graphical overview of the total requirements matrix across all sector-regions in Figure B3.

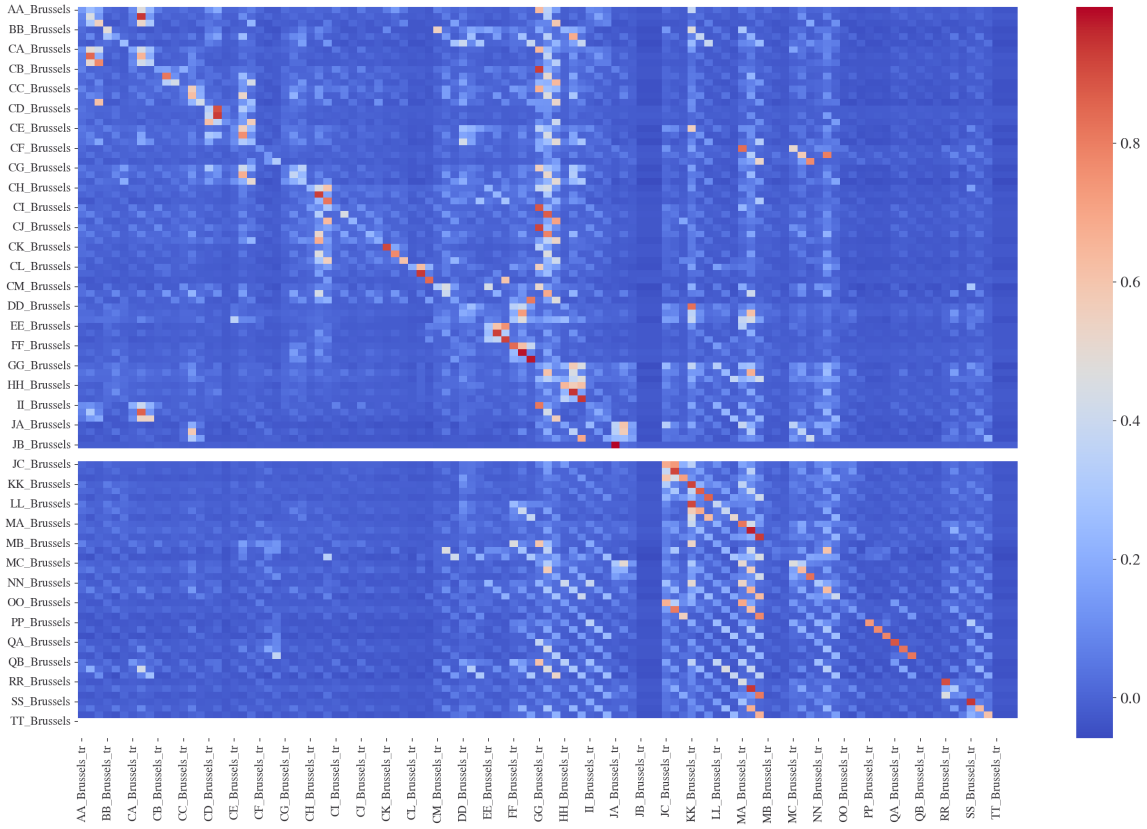


Figure B3: Total requirements matrix (heat map).

Sector-region	Transform
JC_Brussels	raw
JC_Flanders	raw
TT_Wallonia	raw
QB_Brussels	log
JB_Flanders	log
TT_Flanders	log
JC_Brussels	standardized
JC_Flanders	standardized
TT_Wallonia	standardized
JC_Flanders	sqrt
TT_Flanders	sqrt
TT_Wallonia	sqrt
MA_Brussels	inverse
OO_Brussels	inverse
QB_Brussels	inverse
EE_Flanders	inverse
FF_Flanders	inverse
JA_Flanders	inverse
JB_Flanders	inverse
MA_Flanders	inverse
PP_Flanders	inverse
TT_Flanders	inverse
EE_Wallonia	inverse
MA_Wallonia	inverse
PP_Wallonia	inverse
QB_Wallonia	inverse

Table B1: Sector-region-transforms with identified structural breaks.

C Additional results ensemble predictions

We compare the forecasts obtained for 2023 using variables for gross value added in current and chained prices. For the values to be comparable we use national deflators for 2023 to transform the current to chained prices and vice versa. In the report we discuss the results obtained using the ensemble based on the best performing model and the chained prices transformed to current prices. The other results (i.e., using the weighted version of the ensemble) are store in the `/comparison_-chained_and_current` directory.

In Table B2 we report the level of the variables across sectors and regions. We can see that the aggregated (across regions) values obtained using predictions in chained prices are close to the observed national values for 2023.[40] However, notice that there are sectors with small differences between the two variables, such as MC ("Advertising and market research; other professional, scientific and technical activities; veterinary activities"), but also sectors like DD ("Electricity, gas, steam and air-conditioning supply") for which the different rescaling methods lead to a significant divergence in the results. We also report growth rates in Table B3 computed using forecasted and last observed (i.e., 2022) values for each sector-region. For the last column, which includes growth rates at the national level, we use observed values for 2022 and 2023.

[40]The aggregated values across regions of the predictions in current prices exactly match the aggregated values for Belgium due to the rescaling factor used. Instead, for chained prices the rescaling is done before the values are transformed to current prices so the scaling method from Bassilière et al. (2008) is used.

	Current prices				Current prices from chained prices				
Sector	Brussels	Flanders	Wallonia	Total	Brussels	Flanders	Wallonia	Total	Belgium current prices
AA	8.83	3,338.27	1,163.34	4,510.44	18.31	3,376.91	1,110.15	4,505.38	4,510.44
BB	9.94	84.09	196.54	290.56	10.76	101.98	179.13	291.88	290.56
CA	439.40	7,656.11	2,325.59	10,421.10	450.90	7,739.39	2,222.99	10,413.28	10,421.10
CB	14.27	1,103.96	144.28	1,262.51	16.58	1,091.32	147.51	1,255.42	1,262.51
CC	75.62	2,357.48	695.84	3,128.94	75.67	2,418.39	674.26	3,168.32	3,128.94
CD	215.17	3,050.98	34.16	3,300.31	226.34	3,017.97	45.28	3,289.59	3,300.31
CE	211.20	6,957.92	1,290.99	8,460.11	297.14	7,005.15	1,082.77	8,385.06	8,460.11
CF	252.36	8,284.15	6,190.85	14,727.36	214.46	7,528.09	7,081.33	14,823.88	14,727.36
CG	38.01	3,460.95	1,835.16	5,334.12	44.27	3,601.98	1,694.09	5,340.34	5,334.12
CH	106.50	6,002.29	1,626.01	7,734.80	104.90	6,023.57	1,634.45	7,762.92	7,734.80
CI	20.03	1,224.01	434.02	1,678.07	19.80	1,228.44	421.19	1,669.43	1,678.07
CJ	9.33	771.48	583.14	1,363.95	11.62	878.42	454.76	1,344.80	1,363.95
CK	62.02	4,106.78	436.55	4,605.35	71.05	3,920.63	610.38	4,602.07	4,605.35
CL	575.27	2,376.22	718.25	3,669.74	575.52	2,164.39	901.90	3,641.81	3,669.74
CM	183.35	2,892.94	685.77	3,762.06	202.85	2,900.47	656.02	3,759.34	3,762.06
DD	1,208.94	2,819.36	1,559.50	5,587.80	1,210.18	2,349.18	1,193.84	4,753.20	5,587.80
EE	820.38	2,281.68	1,389.14	4,491.20	498.09	2,657.15	1,338.25	4,493.50	4,491.20
FF	1,976.80	19,959.78	6,169.83	28,106.41	1,979.18	19,753.62	6,402.16	28,134.96	28,106.41
GG	7,146.27	43,275.54	12,591.67	63,013.48	6,386.49	43,887.82	12,603.76	62,878.08	63,013.48
HH	5,394.60	16,433.79	5,958.17	27,786.56	5,418.45	16,811.01	5,512.00	27,741.46	27,786.56
II	2,375.50	5,673.89	2,427.71	10,477.11	2,722.52	5,724.64	1,876.23	10,323.39	10,477.11
JA	1,135.83	1,770.75	434.63	3,341.21	1,280.93	1,628.99	432.15	3,342.06	3,341.21
JB	2,707.93	2,346.59	535.92	5,590.44	2,722.73	2,322.00	540.98	5,585.72	5,590.44
JC	3,110.34	10,006.71	2,009.96	15,127.01	2,968.75	9,924.94	2,115.93	15,009.62	15,127.01
KK	17,197.21	10,684.43	4,408.40	32,290.04	17,999.21	10,660.60	3,634.11	32,293.91	32,290.04
LL	6,760.07	30,402.36	13,226.79	50,389.22	6,730.37	30,090.24	13,475.77	50,296.38	50,389.22
MA	8,597.64	32,774.35	8,532.82	49,904.81	10,988.32	30,768.98	8,109.99	49,867.29	49,904.81
MB	173.27	925.68	803.27	1,902.22	174.27	894.18	832.33	1,900.78	1,902.22
MC	754.23	2,466.00	557.46	3,777.69	748.72	2,473.87	542.73	3,765.32	3,777.69
NN	3,947.45	17,286.20	5,541.37	26,775.02	4,411.33	17,551.43	4,925.38	26,888.13	26,775.02
OO	12,944.50	15,653.86	11,195.19	39,793.55	13,046.44	15,282.27	11,461.66	39,790.37	39,793.55
PP	5,590.32	19,944.85	11,602.34	37,137.51	5,670.43	19,999.85	11,455.58	37,125.85	37,137.51
QA	3,008.07	12,276.37	6,605.92	21,890.36	2,637.48	12,976.77	6,271.91	21,886.16	21,890.36
QB	1,427.57	8,058.90	3,919.38	13,405.86	1,276.37	8,616.15	3,490.89	13,383.40	13,405.86
RR	1,141.80	1,811.83	1,115.41	4,069.04	948.97	2,024.60	1,081.95	4,055.51	4,069.04
SS	1,689.33	2,873.99	1,354.42	5,917.74	2,023.82	2,587.75	1,305.60	5,917.18	5,917.74
TT	77.53	438.82	59.20	575.54	84.72	432.24	58.49	575.45	575.54
Total	91,406.89	313,833.35	120,358.98	525,599.22	94,267.95	312,415.37	117,577.92	524,261.24	525,599.22

Table B2: Comparison predictions from current versus chained prices, levels.

	Current prices (%)			Current prices from chained prices (%)			
Sector	Brussels	Flanders	Wallonia	Brussels	Flanders	Wallonia	Belgium current prices (%)
AA	-22.56	27.46	26.64	60.64	28.93	20.85	27.09
BB	14.23	-12.95	-3.75	23.71	5.57	-12.28	-6.12
CA	4.94	6.74	7.24	7.69	7.90	2.51	6.77
CB	-18.93	-0.45	-8.57	-5.77	-1.59	-6.52	-1.70
CC	-16.07	-2.22	-19.93	-16.01	0.31	-22.41	-7.15
CD	-61.48	3.85	-12.40	-59.48	2.72	16.10	-6.65
CE	-15.18	-20.01	-21.35	19.33	-19.47	-34.03	-20.11
CF	11.46	15.01	3.37	-5.27	4.51	18.24	9.75
CG	2.73	-5.08	-11.29	19.64	-1.22	-18.11	-7.27
CH	-17.63	-10.06	-20.38	-18.87	-9.74	-19.97	-12.56
CI	4.89	6.61	18.94	3.65	7.00	15.43	9.53
CJ	-12.82	-5.10	47.52	8.61	8.06	15.04	11.90
CK	-0.28	15.28	-11.16	14.23	10.06	24.21	11.89
CL	21.98	21.42	24.14	22.04	10.59	55.88	22.03
CM	16.49	2.21	5.05	28.87	2.47	0.49	3.33
DD	-44.54	-44.74	-43.85	-44.49	-53.95	-57.02	-44.45
EE	51.19	-9.49	5.32	-8.20	5.40	1.46	2.48
FF	1.61	11.43	-3.64	1.74	10.27	-0.01	7.03
GG	18.76	6.67	13.16	6.13	8.18	13.27	9.18
HH	5.60	-0.77	8.67	6.07	1.51	0.53	2.34
II	57.43	7.35	20.31	80.43	8.31	-7.02	18.89
JA	-4.37	11.09	3.73	7.85	2.19	3.14	4.39
JB	2.35	5.41	1.85	2.91	4.30	2.81	3.56
JC	8.14	20.71	2.30	3.22	19.72	7.70	15.20
KK	7.04	7.52	20.94	12.03	7.29	-0.30	8.91
LL	10.34	11.42	10.44	9.86	10.28	12.52	11.02
MA	-0.16	10.10	8.23	27.60	3.37	2.86	7.87
MB	9.25	3.45	2.92	9.88	-0.07	6.64	3.73
MC	3.01	8.46	5.86	2.26	8.81	3.06	6.94
NN	-5.72	8.42	7.67	5.36	10.08	-4.30	5.92
OO	9.41	7.42	5.36	10.27	4.87	7.87	7.46
PP	3.85	7.36	9.14	5.34	7.66	7.76	7.36
QA	7.36	0.19	7.75	-5.87	5.91	2.30	3.33
QB	9.62	8.28	10.39	-1.99	15.77	-1.68	9.03
RR	21.78	6.10	23.51	1.21	18.56	19.80	14.68
SS	-12.00	9.33	6.31	5.42	-1.56	2.48	1.64
TT	2.96	2.53	4.41	12.52	0.99	3.16	2.78
Total	5.53	5.79	5.81	8.84	5.31	3.37	5.75

Table B3: Comparison predictions from current versus chained prices, growth rates.

D Metrics and statistical tests

This appendix presents and discusses the statistical tests and metrics used in the toolbox.

D.1 Goodness of fit and model selection metrics

Bayesian Information Criterion (BIC) The BIC is a model selection criterion used to compare statistical models, balancing model fit and complexity by penalizing the number of parameters. Lower BIC values indicate a better model. Formally:

$$BIC = -2\ln(\hat{L}) + k\ln(N) \tag{20}$$

where $\hat{L}$ is the maximized value of the likelihood function of the model (a higher likelihood indicates a better fit), k is the number of parameters in the model, and N is the number of observations. When fitting models, the likelihood can often be improved by adding parameters, at the risk of overfitting. BIC introduces a penalty term proportional to the number of parameters and the logarithm of the sample size, discouraging complexity. Compared to the Akaike Information Criterion (AIC), BIC imposes a stronger penalty for the number of parameters. Important for our setup, BIC enables the comparison of non-nested models. BIC values should not be interpreted as an absolute measure but rather used to rank competing models.

Root Mean Squared Error (RMSE) The RMSE is a goodness-of-fit metric used to evaluate the accuracy of predictions by measuring the average squared difference between observed and predicted values. Lower RMSE values indicate better model performance. Formally:

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(y_i - \hat{y}_i\right)^2} \tag{21}$$

where y_i is the observed value, $\hat{y}_i$is the predicted value, and N is the total number of observations. RMSE measures the average magnitude of residuals (errors) in the same units as the target variable. It is sensitive to large prediction errors due to squaring the residuals, which makes it particularly effective for detecting models that systematically over- or under-predict. However, it is scale-dependent, meaning it cannot be used to compare datasets with different units or scales directly.

Normalized Root Mean Squared Error (NRMSE) The Normalized Root Mean Squared Error (NRMSE) is a version of the RMSE that is normalized to allow comparison across datasets with different scales. Normalization is often done using the mean of the observed values. In our case, we normalize by the mean of the individual sector-region time series for the untransformed variables. Lower NRMSE values indicate better model performance. Formally:

$$NRMSE = \frac{\sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(y_i - \hat{y}_i\right)^2}}{\bar{y}} \tag{22}$$

where y_i is the observed value, $\hat{y}_i$is the predicted value, N is the total number of observations, and $\bar{y}$ is the mean of the observed values. NRMSE provides a scale-free metric, making it useful for comparing model performance across datasets or variables with different units. By normalizing with the mean, the NRMSE expresses error as a proportion of the mean, offering an intuitive interpretation:

an NRMSE of 0.1 indicates that the model's error is approximately 10% of the average gross value added for the sector-region.

In-sample (N)RMSE and validation (N)RMSE It is important to highlight the distinction between the in-sample (N)RMSE, which is calculated using the same data that was used to fit the model (the training data), and the pseudo out-of-sample or validation (N)RMSE that is calculated on a validation dataset that was not used during model training (the test data). The first measures how well the model fits the data it was trained on, while the second measures the model's ability to generalize to unseen data. Models with more regressors can fit the training data more closely, potentially reducing the in-sample RMSE on the training set but increasing the risk of overfitting, leading to poorer out-of-sample performance.

D.2 Statistical tests

CUSUM structural break test The Cumulative Sum (CUSUM) test is a statistical method used to detect structural breaks in a time series or regression model. A structural break represents an unexpected change over time in the parameters of regression models, which can lead to huge forecasting errors and general unreliability of the model. The CUSUM test examines the stability of parameters over time by analyzing the cumulative sum of recursive residuals. If the regression parameters are stable, the cumulative sum of these residuals should fluctuate randomly around zero. Significant deviations from this behavior indicate potential structural changes in the parameters. Deviations from zero in the CUSUM statistic indicate potential parameter instability or structural breaks. The test is based on the cumulative sum of standardized residuals:

$$W_t = \frac{1}{\sigma} \sum_{i=1}^{t} \hat{u}_i \tag{23}$$

where W_t is the cumulative sum of residuals up to observation (time period) $t \leq T$, $\hat{u}_i$ are the recursive residuals, and σ is the estimated standard deviation of the residuals. In particular: (i) compute the recursive residuals $\hat{u}_t$ from the regression model, (ii) calculate W_t for each t from $t_0 + 1$ to T, (iii) compare W_t to critical bounds. These bounds are typically derived from the expected behavior of a Brownian motion process under the null hypothesis of parameter stability. The critical bounds are given by $\pm c\sqrt{t - t_0}$ where c is a constant determined by the desired significance level. The intuition behind the CUSUM test is that if residuals change systematically from one period to the next, the one-step-ahead forecast will no longer be accurate, and the forecast error will deviate significantly from zero. Null hypothesis (H_0): the parameters are stable over time (no structural break). Alternative hypothesis (H_a): the parameters are not stable (structural break exists).

Jarque-Bera normality test The Jarque-Bera (JB) test is a goodness-of-fit test used to determine whether a model's residuals follow a normal distribution. It examines the skewness and kurtosis of the data compared to a normal distribution. The test is commonly applied in regression diagnostics to validate the assumption of normally distributed errors. The test statistic is defined as:

$$JB = \frac{N}{6} \left(S^2 + \frac{(K-3)^2}{4} \right) \tag{24}$$

where N is the number of observations, S is the sample skewness, and K is the sample kurtosis. The test statistic follows a chi-squared (χ^2) distribution with 2 degrees of freedom. Compare the JB statistic to the critical value from the chi-squared distribution or use the corresponding p-value. Null hypothesis (H_0): the data follows a normal distribution (skewness = 0 and kurtosis = 3). Alternative hypothesis (H_a): the data does not follow a normal distribution. The statistic is simple and easy to compute. However, in large sample sizes, even small deviations from normality can lead to rejection of the null.

Johansen cointegration test The Johansen cointegration test is a statistical method used to determine the presence and number of cointegrating relationships among multiple non-stationary time series. Cointegration indicates a long-term equilibrium relationship between the series, even if they are individually non-stationary. The test is performed within the framework of a Vector Error Correction (VEC) model. Null Hypothesis (H_0): there are at most r cointegrating relationships. Alternative Hypothesis (H_a): There are more than r cointegrating relationships. There are two test statistics:

1. Trace Statistic: $\lambda_{\text{trace}}(r) = -N \sum_{i=r+1}^{k} \ln(1 - \hat{\lambda}_i)$ where $\hat{\lambda}_i$ are the estimated eigenvalues of the Π matrix, and N is the sample size. It tests the null hypothesis that the number of cointegrating vectors is at most r against the alternative that it is greater than r.

2. Maximum Eigenvalue Statistic: $\lambda_{\max}(r, r+1) = -N \ln(1 - \hat{\lambda}_{r+1})$ tests the null hypothesis that the number of cointegrating vectors is r against the alternative of $r + 1$.

Compare the test statistics to critical values (often tabulated for specific confidence levels). If the statistic exceeds the critical value, reject the null hypothesis.

KPSS stationarity test The Kwiatkowski-Phillips-Schmidt-Shin (KPSS) test is a statistical test used to evaluate whether a time series is stationary around a deterministic trend or mean. Unlike many stationarity tests, the KPSS test's null hypothesis (H_0) assumes the data is stationary around a mean or deterministic trend, while the alternative hypothesis (H_a) assumes the presence of a unit root (non-stationarity). The KPSS test statistic is calculated as:

$$\eta = \frac{1}{N^2} \sum_{i=1}^{N} S_i^2 \Big/ \hat{\sigma}^2$$

where $S_i = \sum_{j=1}^{i} e_j$ is the partial sum of the residuals e_j, N is the number of observations, and $\hat{\sigma}^2$ is the long-run variance of the residuals. Compare η to critical values provided in KPSS tables. Reject the null hypothesis if η exceeds the critical value.

Ljung-Box or Portmanteau white noise test . The Ljung-Box test is used to determine whether a time series or residuals from a time series model are white noise. It evaluates whether a group of auto-correlations at different lags is significantly different from zero, helping to identify model mis-specification in time series models. If the test fails to reject the null hypothesis (H_0), the data are consistent with white noise, indicating no significant autocorrelation. The Ljung-Box test statistic is given by:

$$Q = T(T+2) \sum_{k=1}^{h} \frac{\hat{\rho}_k^2}{T-k}$$

where Q is the test statistic, T is the number of observations, h is the number of lags being tested, $\hat{\rho}_k$ is the sample autocorrelation at lag k. Null Hypothesis (H_0): the data are white noise (no autocorrelation at any lag up to h). Alternative Hypothesis (H_a): the data are not white noise (autocorrelation exists at one or more lags). The Q statistic follows a χ^2 distribution with h degrees of freedom. Reject H_0 if Q exceeds the critical value, indicating significant autocorrelation.