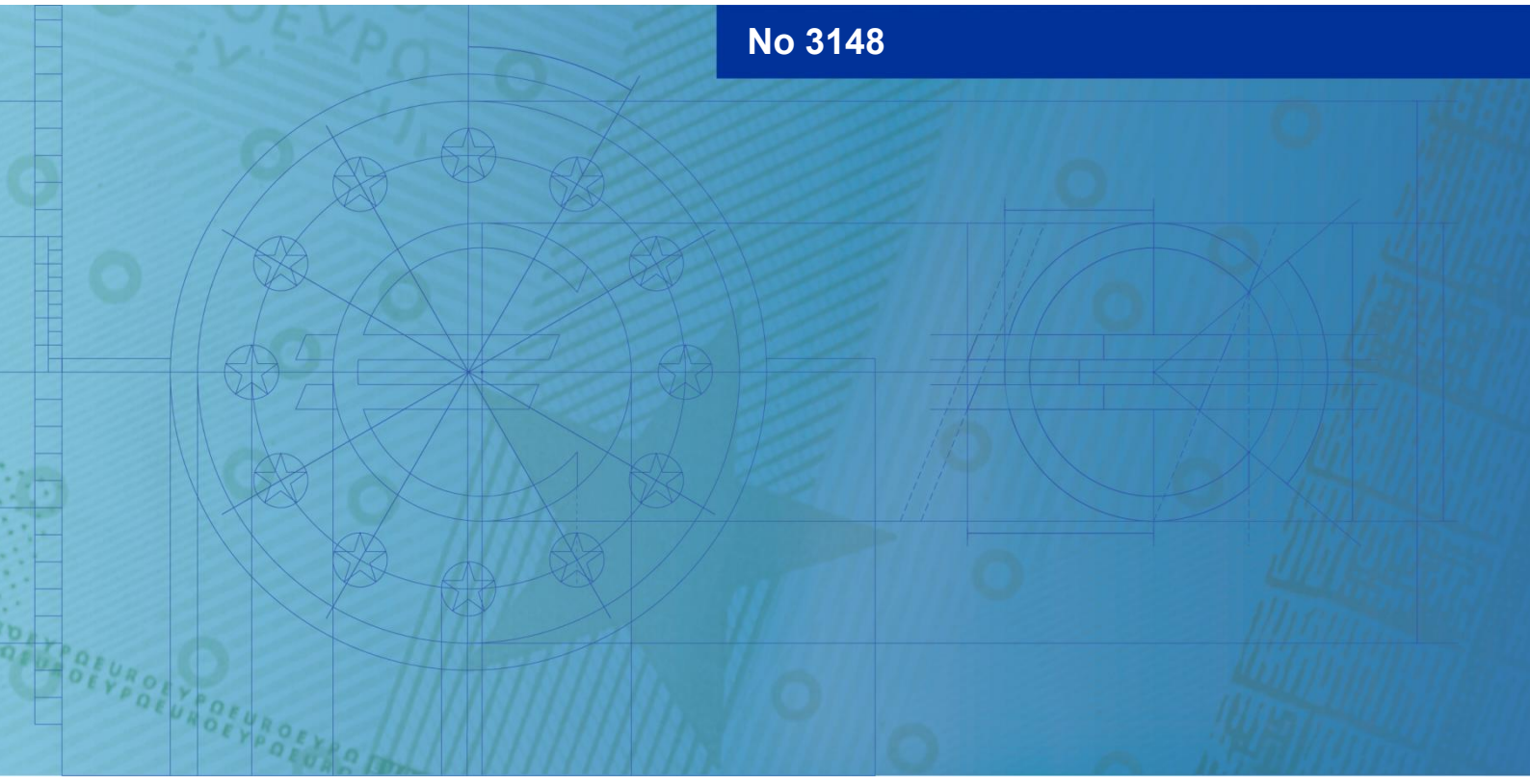
EUROPEAN CENTRAL BANK
EUROSYSTEM

Working Paper Series

Max Lampe, Ramón Adalid

A machine learning approach to real time identification of turning points in monetary aggregates M1 and M3

No 3148



Abstract

Monetary aggregates provide valuable information about the monetary policy transmission and the business cycle. This paper applies machine learning methods, namely Learning Vector Quantisation (LVQ) and its distinction-sensitive extension (DSLVQ), to identify turning points in euro area M1 and M3. We benchmark performance against the Bry–Boschan algorithm and standard classifiers. Our results show that LVQ detects M1 turning points with only a three-month delay, halving the six-month confirmation lag of Bry–Boschan dating. DSLVQ delivers comparable accuracy while offering interpretability: it assigns weights to the sources of broad money growth, showing that lending to households and firms, as well as Eurosystem asset purchases when present, are the main drivers of turning points in M3. The findings are robust across parameter choices, bootstrap designs, alternative performance metrics, and comparator models. These results demonstrate that machine learning can yield more timely and interpretable signals from monetary aggregates. For policymakers, this approach enhances the information set available for assessing near-term economic dynamics and understanding the transmission of monetary policy.

Keywords: Monetary aggregates, Turning points, machine learning

JEL Codes: E32, E51, C63

Non-technical summary

Monetary aggregates, such as M1 and M3, have long been monitored by economists and central banks. While historically this was mainly because money growth was regarded as a leading indicator of inflation, more recently attention has shifted towards their usefulness for understanding economic activity and the transmission of monetary policy. Turning points in real M1 have often been found to lead turning points in real GDP, while developments in M3 and its components help to trace the channels through which monetary policy affects households, firms and financial markets. **Therefore, for policymakers, timely recognition of turning points in monetary dynamics is highly valuable.** This study shows that modern machine learning methods can provide earlier signals of such turning points than traditional approaches. Traditional methods for identifying turning points, such as the Bry–Boschan dating algorithm, are reliable but slow. They typically require at least six months of data confirmation before a turning point can be dated, which limits their usefulness for real-time policy assessment. Moreover, conventional approaches do not explain which sources of money growth are most relevant when a shift occurs.

This study applies modern machine learning methods to address these shortcomings. We use Learning Vector Quantisation (LVQ) and a distinction-sensitive extension (DSLVQ) to identify turning points in euro area monetary aggregates. The analysis covers both M1, a narrow aggregate closely related to household money holdings, and M3, a broad aggregate that can be decomposed into its main counterparts: lending to households, lending to firms, government borrowing, external flows, banks' long-term liabilities, and Eurosystem asset purchases. By combining the classification capacity of machine learning with the ability of DSLVQ to assign weights to input variables, the approach provides not only more timely turning-point signals but also insights into their drivers.

The results show three main findings. First, LVQ identifies turning points in M1 with only a three-month delay, cutting in half the confirmation lag required by the Bry–Boschan method. This timelier detection is important because turning points in real M1 have long been regarded as leading signals for turning points in real economic activity. Second, DSLVQ delivers accuracy comparable to LVQ while offering additional interpretability. It shows that shifts in M3 are most strongly associated with lending to households and firms, with Eurosystem asset purchases also strengthening the signal when they are present. Other components, such as net external flows or banks' long-term liabilities, generally play a smaller role. Third, the findings are robust. They hold across different parameter choices, sampling designs and evaluation metrics, and they also outperform traditional statistical models.

These results matter for monetary policy analysis. They demonstrate that machine learning can provide more timely information about changes in monetary dynamics and, at the same time, shed light on the underlying drivers of broad money growth. By combining timeliness and interpretability, this approach enhances the toolkit available to central banks. It offers policymakers a richer information set when assessing short-term developments in economic activity and when evaluating the effectiveness of monetary policy transmission.

1 Introduction

Turnin gpoints in monetary aggregates have long attracted attention as potential signals of shifts in the business cycle and the transmission of monetary policy[1]. Real M1, in particular, has been shown to lead real GDP (Marcellino, 2006), while M3 and its counterparts shed light on the underlying sources of money creation and the state of monetary policy transmission (Fischer et al., 2009; Adalid et al., 2024). For policymakers, timely recognition of such turning points is valuable: it can expand the information set used in assessing near-term economic dynamics and provide insight into the channels through which monetary policy affects the economy.

Existing approaches, however, face limitations. The Bry–Boschan algorithm, widely used to date turning points (Bry and Boschan, 1971), requires at least six months of confirmation, which reduces its usefulness for real-time policy assessment. Moreover, standard classifications of monetary aggregates do not reveal which sources of money creation drive observed shifts in broad money.

This paper applies a machine-learning approach to overcome these shortcomings. We apply Learning Vector Quantisation (LVQ) and its distinction-sensitive extension (DSLVQ) to euro area data in order to identify turning points in M1 and M3. Our application yields three main findings. First, LVQ identifies M1 turning points with only a three-month delay, halving the confirmation lag of the Bry–Boschan algorithm. Second, DSLVQ not only matches LVQ's performance but also produces weights that highlight the relative importance of different sources of broad money growth. Lending to households and firms emerges as particularly influential, with Eurosystem asset purchases strengthening the signal when present. Third, the results are robust across parameter choices, bootstrap designs, alternative performance metrics, and comparator models.

These contributions are relevant for both research and policy. They demonstrate how machine learning can provide more timely signals of monetary turning points and, at the same time, offer interpretable insights into their drivers. In doing so, the paper adds to the literature on monetary aggregates and the analysis of monetary transmission, and it shows that methods

[1]Monetary aggregates were long used as early indicators of inflation, underpinning monetary targeting regimes in the 1970s–1990s. As the empirical link between money growth and inflation weakened in low-inflation environments, central banks shifted toward other indicators. Reflecting this, the ECB in its 2003 and 2021 strategy reviews revised the role of the 'monetary pillar,' progressively integrating money and credit into a broader analysis of monetary transmission (Holm-Hadulla et al., 2021; Lane, 2024). Other major central banks, such as the Federal Reserve, the Bank of England, and the Bank of Japan, also experimented with monetary targeting in the 1970s–1980s but subsequently abandoned it in favor of frameworks centered on interest rates and, later, inflation targeting. Nonetheless, the money–inflation relationship remains part of the debate.

commonly used in other fields can be fruitfully applied in central banking (Giusto and Piger, 2017).

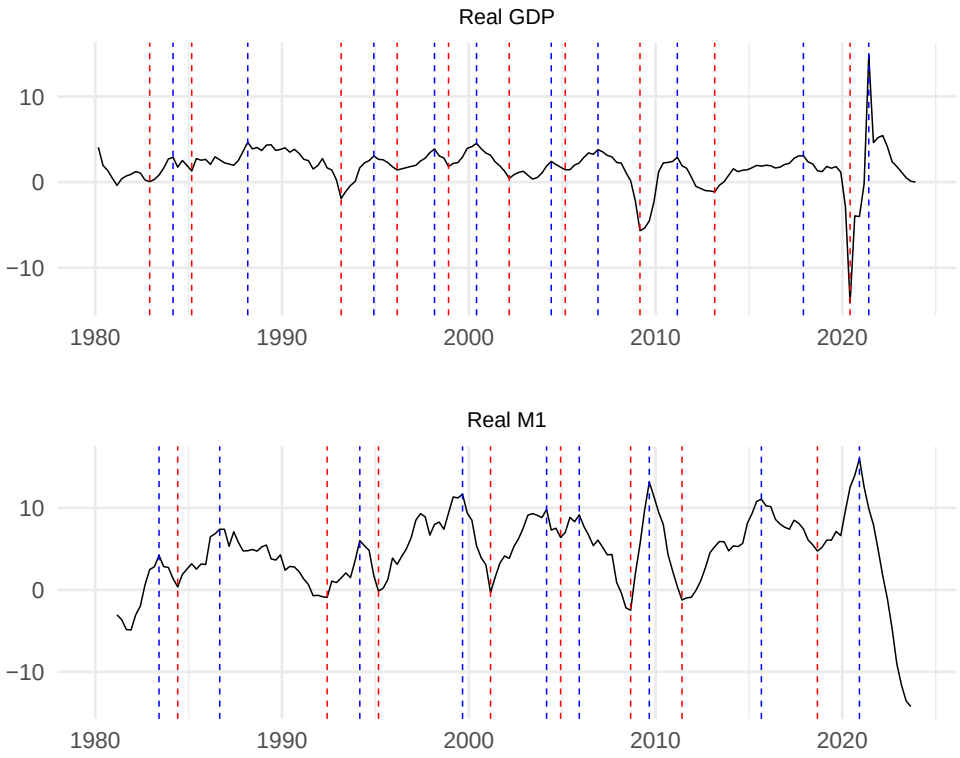
The paper is structured as follows. Section 2 introduces the concepts of turning points and the Bry–Boschan algorithm. Section 3 presents the LVQ and DSLVQ methodologies. Section 4 reports the application to M1 and M3, including results, robustness checks, and interpretation. Section 5 concludes with the main policy implications.

2 Turning Points in Monetary Aggregates and Business Cycles

A standard mathematical definition of a turning point, as in Clapham and Nicholson (2009), is a point on a curve where the derivative is zero and changes sign, indicating a local maximum or minimum. While precise, this definition is not well suited for analysing changes in macroeconomic variables that reflect different phases of the business cycle. If applied literally, it would identify an excessively high number of turning points, generating implausibly short cycles. This reflects the fact that macroeconomic data often display short-lived fluctuations that do not correspond to genuine shifts in the underlying macroeconomic environment. For this reason, business cycle analysis views macroeconomic variables as evolving across relatively persistent phases, such as expansion and contraction, with peaks and troughs marking the transitions between them. Consequently, statistical methods for dating turning points in macroeconomics must smooth out short-term noise.

A widely used approach is the algorithm developed by Bry and Boschan (1971). This non-parametric method identifies peaks and troughs in macroeconomic time series. When applied to GDP, the Bry–Boschan algorithm approximates well the turning points identified by expert committees, such as the National Bureau of Economic Research (NBER) for the United States or the Euro Area Business Cycle Network (EABCN) for the euro area (Marcellino, 2006). The method imposes restrictions on the identification of turning points.F or instance, it enforces minimum durations for phases (two quarters) and complete cycles (four quarters), reducing the risk that the identified turning points reflect short-lived fluctuations rather than substantive cyclical changes. The Bry–Boschan procedure has a key limitation for real-time analysis. Because of the two-quarter window, a turning point can only be confirmed once two additional quarters of data have become available. For policymaking purposes, this delay is considerable. For instance, it is widely accepted that real M1 typically leads GDP by only a few quarters.

Therefore, confirmation lags reduce its usefulness for forward-looking assessment.



Note: The dashed red vertical lines represent troughs and the dashed blue vertical lines represent peaks.

Figure 1: Time series of quarterly year-on-year real GDP and real M1 growth rate, and turning points identified using the Bry-Boschan algorithm

When applied to the relationship between real M1 and real GDP, the Bry–Boschan algorithm confirms that turning points in real M1 leads real GDP, as major turning points in M1 typically precede those in GDP. The correlation between real M1 and real GDP can also be quantified using the concordance index, which measures the proportion of time two binary series exhibit the same cyclical phase. Originally introduced for medical testing (Harrell et al., 1982) and later adapted to economics by Harding and Pagan (2003), the measure confirms that real M1 leads real GDP by around four quarters. This finding, consistent with Musso (2019), highlights the predictive value of M1 for identifying turning points in economic activity (Figure 2).

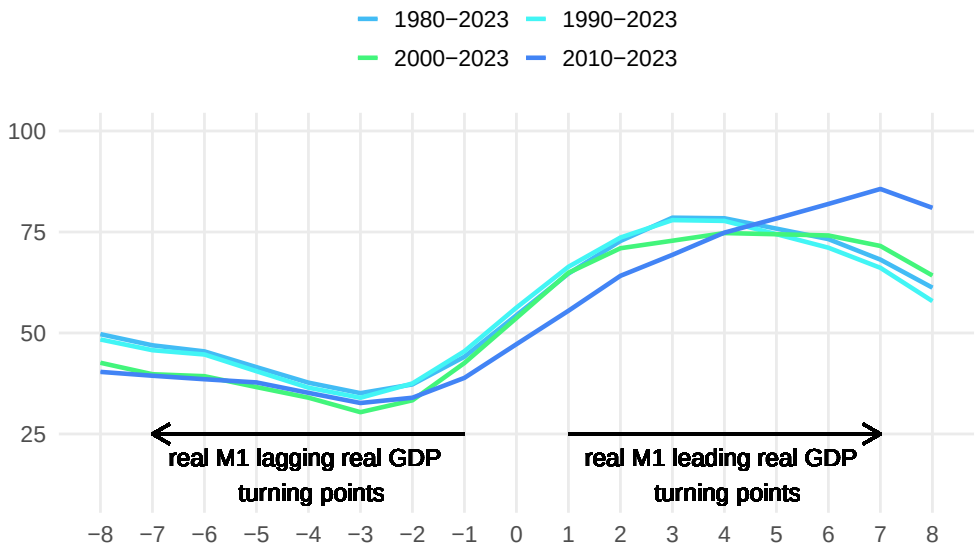


Figure 2: Concordance index of real GDP and real M1

3 Learning Vector Quantization (LVQ) and Distinction Sensitive Learning Vector Quantization (DSLVQ)

The usefulness of turning point identification for policymaking depends not only on eventual detection but also on the ability to recognise them with minimal delay. Traditional dating methods, such as the Bry–Boschan algorithm, involve confirmation lags of at least two quarters, which limits their value for forward-looking assessment. To reduce these lags, one requires techniques capable of classifying the state of the economy in near real time. Classification algorithms are well suited to do this task because they summarise complex data into a small number of discrete states, thereby reducing both dimensionality and the time required for identification. Their effectiveness has been demonstrated in fields as diverse as pattern recognition, medical diagnosis, and credit risk modelling, but they have been little used in economics.

One such algorithm is Learning Vector Quantisation (LVQ), developed by Kohonen (1982) as a supervised neural-network–based classifier. The basic idea of LVQ is to classify a new observation by comparing it with a small number of typical observations from the categories to be distinguished, for example, a period of expansion or a period of contraction. In the LVQ literature, these typical observations are called prototype vectors, as they serve as reference points that represent each class. A new observation is assigned to the class of the prototype vector to which it is closest. In this way, LVQ reduces complex data to a small number of meaningful

categories. LVQ has been applied successfully to tasks such as handwriting recognition (Takahashi and Nishiwaki, 2003), medical diagnostics (Jagric et al., 2011), and credit risk modelling (Moonasar, 2007). Its use in economics has been more limited, but Giusto and Piger (2017) showed that LVQ can be employed to identify business cycle turning points in the United States, demonstrating its relevance for macroeconomic applications.

Formally, LVQ assigns to an unclassified observation x_u the class c_u with the closest prototype vector in Euclidean distance:

$$c_u = arg\,min_{i \in \{1,...,\bar{N}\}} \{\|x_u - m_i\|\} \tag{1}$$

where $\bar{N}$ denotes the number of prototype vectors and $m_i \in \mathbb{R}^m, i = 1, ..., \bar{N}$ are the prototype vectors themselves, defined in the same space as the observations. In our application, the classes correspond to periods of increasing or decreasing growth in the monetary aggregate.

In order to use LVQ in practice, the prototype vectors must first be learned from the data. This process is known as training. In supervised learning models such as LVQ, training involves providing the algorithm with a set of historical observations for which the class is already known — for example, whether a period corresponded to an expansion or a contraction. The algorithm then adjusts the position of the prototypes so that they become good representatives of their respective classes. Without this training step, LVQ would not be able to classify new observations meaningfully, since the prototypes would not reflect the structure of the data [2]. A detailed description of the training algorithm lvq3 can be found in Kohonen (1990).

Training an LVQ model involves determining the location of the prototype vectors so that they represent their classes as accurately as possible. This is achieved through an iterative process in which the prototypes are repeatedly adjusted using labelled observations from a training set. For each observation the position of the two closest prototype are slightly moved. In case the prototype is of the same class it is moved closer is correctly classified, the corresponding prototype is moved slightly closer to it in case the prototype is of a different class it is moved slightly further away. Over many iterations, the prototypes converge to positions that provide good separation between the classes. The amount by which the prototypes are adjusted in each step is governed by a learning rate, which decreases gradually as training progresses to ensure

[2] A single iteration consists of adjusting each prototype based on each training point observation. Increasing the number of iterations improves the accuracy up to a specific point, beyond this point the accuracy decreases. The LVQ at such a point is overfitted to the training data and does not generalize well to unseen data. We set the number of iterations to 100 times the number of prototypes following the suggestion by Kohonen (1998).

the stability of the final solution.

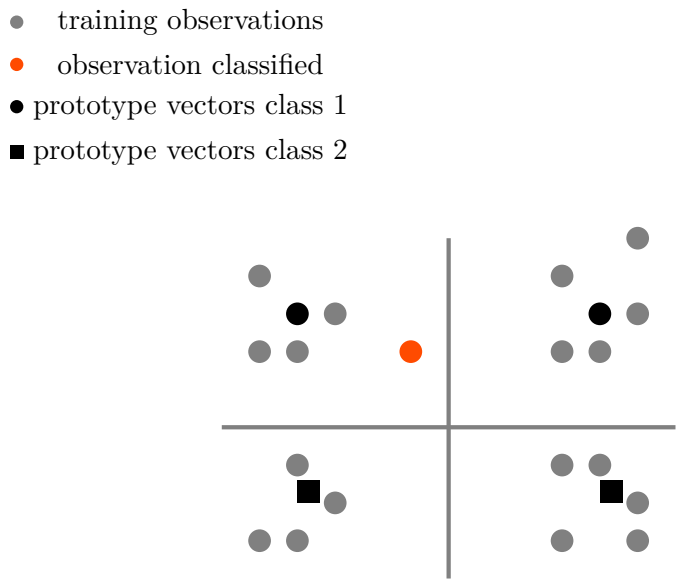
Formally, the adjustment of prototype vectors in iteration $t+1$ can be written as:

$$m_i\,(t+1) = m_i(t) - \alpha(t)\,[x\,(t) - m_i\,(t)] \tag{2}$$

$$m_j\,(t+1) = m_j(t) + \alpha(t)\,[x\,(t) - m_j\,(t)] \tag{3}$$

where $x(t)$ is the observation considered at iteration t, m_i is the closest prototype of the correct class, and m_j is the closest prototype of a different class. The parameter $\alpha(t)$ is the learning rate, which decreases over time. This winner-takes-all learning process ensures that prototypes are pulled towards observations of their own class and pushed away from those of other classes, thereby improving classification accuracy.

Figure 3 provides a two-dimensional illustration of an LVQ model with four prototype vectors (black points) and training observations (grey points). The lines indicate the regions of the space associated with each prototype. A new observation is classified by determining which region it falls into, that is, which prototype it is closest to. The figure illustrates how LVQ partitions the space into distinct areas corresponding to different classes.



Note: The grey dots represent observations used to train the prototype vectors colored black.

Figure 3: 2 dimensional illustration of a DSLVQ with 4 prototype vectors

Further to the geometric illustration in Figure 3, LVQ can also be represented as a simple neural network, shown in Figure 4. In this representation, the input vector x is compared with all prototype vectors, denoted by M, and the distances to each prototype are calculated. The

observation is then assigned to the class associated with the nearest prototype. This formulation highlights that LVQ can be viewed as a special case of a neural network with a classification rule. For readers unfamiliar with neural networks, this alternative representation is not required for understanding the methodology; its role is simply to connect LVQ to the broader machine-learning literature from which it originates.

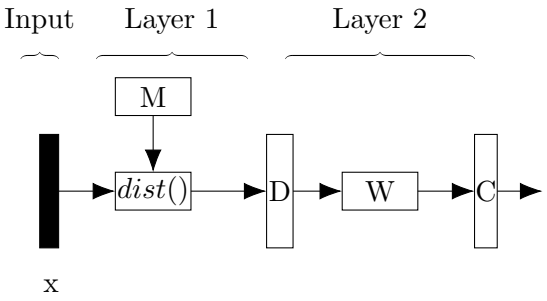


Figure 4: Illustration of a LVQ as a neural network.

While LVQ provides a flexible tool for classifying observations into discrete states, it has an important limitation: in its standard form, it treats all input features as equally relevant. In practice, some features contain more information than others. For example, in the context of monetary aggregates, some sources of money creation may provide stronger signals of turning points than others.

Distinction Sensitive Learning Vector Quantisation (DSLVQ), developed by Pregenzer et al. (1996), extends LVQ by introducing a weighted Euclidean distance. Instead of calculating distance on the basis of all features equally, DSLVQ learns a set of weights that reflect the relative importance of each feature for classification. Formally, the distance between an observation x and a prototype m is defined as:

$$DSdist\left(w_n, x_n, m_n\right) = \sqrt{\sum_{n=1}^{N} \left[\max\left(0, w_n\right)\left(x_n - m_n\right)\right]^2} \tag{4}$$

where w_n denotes the weight attached to feature n. During training, the weights are updated dynamically alongside the prototype positions. For feature n, the update rule is:

$$w\left(t+1\right) = \text{norm}\left(w\left(t\right) + \alpha\left(t\right)\left[nw\left(t\right) - w\left(t\right)\right]\right) \tag{5}$$

where $\alpha(t)$ is the learning rate and $nw(t)$ reflects the relative contribution of feature n to

separating classes at iteration t. A normalisation step ensures that the weights remain positive and sum to one:

$$nw_n(t) = \text{norm}\left(\frac{d_{i_n}(t) - d_{j_n}(t)}{\max\left(d_{i_n}(t), d_{j_n}(t)\right)}\right) \text{with } \text{norm}(y) = \frac{1}{\sum_{n=1}^{N}|y_n|}y \tag{6}$$

In this way, DSLVQ gradually shifts weight towards features that improve classification accuracy, while reducing weight on less informative features. In our application, each variable enters the model with several leads and lags. Estimating a separate weight for each lead-lag combination would introduce unnecessary complexity and reduce interpretability. We therefore compute a single aggregate weight for each variable by averaging across its leads and lags:

$$w_n = \frac{\sum_{l=1}^{L} w_{n,l}}{L} \tag{7}$$

where L denotes the number of leads and lags. These aggregate weights provide a clear measure of the relative importance of each variable in identifying turning points. The advantage of DSLVQ is that the weights themselves provide interpretable information about the classification process. Features that receive higher weights are those that contribute more strongly to distinguishing between classes. In our application, these weights reveal which counterparts of M3 are most relevant for identifying turning points. Thus, DSLVQ not only improves classification performance but can also offer insights into the drivers of turning points.

4 Application of (DS)LVQ to M1 and M3 Turning Points

4.1 Application setup

We now apply the methodology described in Section 3 to the case of monetary aggregates M1 and M3. The aim is to assess whether LVQ and DSLVQ can replicate and anticipate the chronology of turning points identified by the Bry–Boschan algorithm, and to compare their performance against standard benchmark classifiers.

We proceed in three steps which are repeated multiple times to gain distributions of the results. First, we sample a time series. Second, we construct a reference chronology of peaks and troughs in M1 and M3 using the Bry–Boschan algorithm, which we introduced in Section 2. Third, we

train LVQ and DSLVQ classifiers on half of the sampled data to reproduce these turning points, and use the second half for benchmarking their performance against a Gaussian Naive Bayes (GNB) model.

An important feature of the data is that turning points are very rare events. In the M1 sample of 297 observations, only 16 correspond to turning points, making this a highly unbalanced classification problem. Standard accuracy measures would be dominated by the large number of non–turning-point observations and thus provide a misleading picture of model performance. To address this, we use balanced accuracy, which gives equal weight to correctly classifying each class regardless of their frequency.

Our exercise is more extensive for M3 than for M1. For M1 (being a narrow aggregate), the information set is limited to the aggregate's own dynamics, so we include only leads and lags of its year-on-year growth rate. For M3, by contrast, the aggregate can be decomposed into its main counterparts or sources of money creation: loans to firms, loans to households, Eurosystem net purchases, net external monetary flows, bank credit to government, and bank long-term liabilities. Incorporating these additional series allows us not only to improve classification performance, but also, through the DSLVQ weights, to identify which sources of broad money growth contribute most to turning points. The sample covers April 1999 to June 2024 (295 observations).

4.2 Evaluation metric

As discussed in Section 4.1, we evaluate model performance using balanced accuracy, which serves as our loss function that penalises false positives and false negatives symmetrically. Formally, it is defined as the average of sensitivity and specificity. Figure 5 illustrates the confusion matrix underlying these measures. The results are consistent across metrics, showing that our conclusions do not depend on the choice of performance measure.

$$\text{Sensitivity} = \frac{TP}{TP + FN} \tag{8}$$

$$\text{Specificity} = \frac{TN}{TN + FP} \tag{9}$$

$$\text{Balanced Accuracy} = \frac{\text{Sensitivity} + \text{Specificity}}{2} \tag{10}$$

		prediction outcome	
		True positive (TP)	False negative (FP)
actual value		False positive (FP)	True negative (TN)

Figure 5: Illustration of the confusion matrix.

4.3 Training design

As with most machine learning models, training LVQ and DSLVQ requires selecting parameters that determine how much the model adapts to past data. If the model adapts too closely, it risks overfitting—capturing noise rather than true patterns. If it adapts too little, it may fail to capture relevant dynamics. To address this trade-off, we proceed as follows.

Prototypes are initialised randomly, and the classification is repeated 100 times. Following Giusto and Piger (2017), the final state is assigned by majority rule, using a threshold of 50%.

A key safeguard against overfitting is to evaluate performance out of sample. Each bootstrap resample is split into two halves of equal size: the first is used for training, the second for evaluation. This ensures that reported performance does not simply reflect memorisation of the training data.

Because business cycle data are serially correlated, we employ **block bootstrapping** rather than random resampling. This method preserves the dependence structure by drawing contiguous blocks of observations. We set the block length to 8 years, corresponding to the average length of euro area business cycles and close to the 6-year average monetary cycle in our sample (Politis and Romano, 1994). Each bootstrap resample contains 297 observations, and we generate 500 resamples in total.

The block bootstrap yields an empirical distribution of balanced accuracy for each model specification. This allows us to compare models not only by point estimates but also by the variability and robustness of their performance. Annex Tables A2–A5 show that results are stable with respect to modest changes in the learning rate, number of prototypes, classification threshold, and block length.

4.4 Results: M1

Table 1 reports the performance of LVQ relative to the GNB benchmark in classifying M1 growth states. The classification of M1 with LVQ models shows that LVQ generally outperforms GNB when using the same amount of data. From two leads/lags onwards, LVQ achieves consistently higher balanced accuracy, and the gains increase with additional leads and lags.

Mean balanced accuracy is around 0.86 with one lead/lag, rising to about 0.92 with three and to 0.95 with five. The additional information helps to avoid false classifications by better capturing the transition between states. However, these gains come at a cost: each added lead delays real-time assessment by one quarter. Beyond three leads/lags, the timeliness advantage over the Bry–Boschan algorithm (which requires six months of confirmation) is largely eroded.

Overall, LVQ identifies M1 turning points with a three-month lag, halving the delay inherent in Bry–Boschan dating. This improvement is policy relevant, since real M1 has long been recognised as a leading indicator of real GDP (see Section 2). A timelier assessment of M1 turning points therefore enhances the information set available to policymakers when anticipating shifts in economic activity.

We next turn to the results for M3, where the inclusion of counterpart information allows us to investigate not only the timing of turning points but also the drivers behind them.

Balanced accuracy			
	5th	mean	95th
GNB with 3 leads/lags	0.778	0.876	0.949
5 leads/lags	0.895	0.951	0.993
4 leads/lags	0.869	0.931	0.974
3 leads/lags	0.854	0.916	0.963
2 leads/lags	0.794	0.891	0.959
1 leads/lags	0.780	0.859	0.939

Table 1: Performance measures of the 5th and 95th percentiles, as well as the mean of the bootstrap distribution of balanced accuracy for GNB and LVQ M1 state classification from 1 up to 5 leads and legs

4.5 Results: M3

Table 2 presents the results for M3. The standalone specification using only M3's own leads and lags already achieves strong performance, with a mean balanced accuracy of about 0.89, clearly outperforming the GNB benchmark. Adding the individual sources of money creation does not substantially raise accuracy for LVQ, though it highlights the limitations of GNB, which performs worse when the dimensionality of the input increases. The main value of including counterparts therefore lies less in boosting accuracy than in enabling an interpretation of their relative importance.

DSLVQ, the distinction-sensitive extension of LVQ introduced in Section 3, provides exactly this interpretive advantage. First, it maintains accuracy comparable to the standard LVQ model even in the higher-dimensional setting. Second, it produces feature weights that quantify the relative importance of the different sources of broad money growth. These weights, shown in Figure 6, are not probabilities or elasticities but importance scores in the classifier's distance metric: they measure how much each counterpart contributes to distinguishing between turning points and non-turning points.

Across bootstrap replications, loans to households, loans to firms, and Eurosystem net purchases

Balanced accuracy

	5th	mean	95th
GNB 3 leads/lags only M3	0.785	0.845	0.903
GNB 3 leads/lags with all counterparts	0.570	0.723	0.868
3 leads/lags LVQ only M3	0.866	0.894	0.915
3 leads/lags with all counterparts LVQ	0.843	0.858	0.874
3 leads/lags with all counterparts DSLVQ	0.841	0.874	0.909
2 leads/lags with all counterparts DSLVQ	0.793	0.841	0.880
1 leads/lags with all counterparts DSLVQ	0.776	0.820	0.867

Table 2: Performance measures of the 5th and 95th percentiles, and mean of the bootstrap distribution of balanced accuracy for GNB, LVQ and DSLVQ M3 state classification from 1 up to 3 leads and legs

receive systematically higher weights — roughly 15% more, on average — than net external monetary flows, bank credit to government, and bank long-term liabilities. This pattern suggests that turning points in M3 are more strongly driven by lending to the private sector, with Eurosystem asset purchases also strengthening the signal when they are in place. Other sources of money creation generally provide a weaker signal for identifying turning points in M3 growth. Sensitivity measures the share of turning points correctly identified, while specificity measures

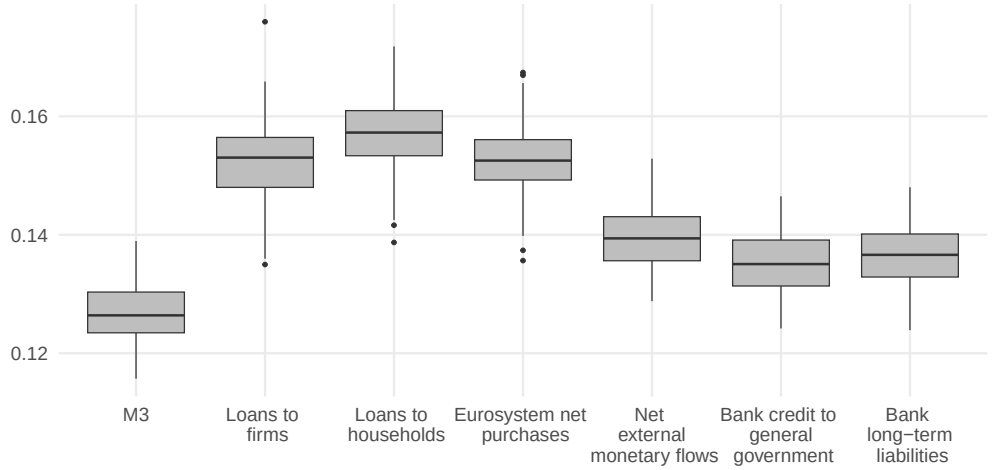


Figure 6: Boxplot of the DSLVQ inputs weights resulting from the bootstrap of the distinction-sensitive learning vector quantization for a M3 model with 3 leads/lags, weights are normalized to sum 1 for illustration purposes

the share of non–turning points correctly classified. To confirm robustness, we also report the F1 score, which provides a complementary measure of classification performance by considering both false positives and false negatives (see Annex Equation 11 for the definition)[3].

4.6 Robustness checks

Machine learning models can be sensitive to parameter choices, sample design, and the choice of a performance metric. To ensure that our results are not driven by such choices, we perform a series of robustness checks.

Annex Tables A2–A5 show that modest changes in the learning rate, number of prototypes, classification threshold, and block length have little effect on balanced accuracy. Annex Table A6 provides additional diagnostic evidence on the bootstrap procedure, including the distribution of resampled statistics. Together, these results indicate that model performance is not dependent on finetuned parameters or specific resampling choices.

Annex Tables A8 and A9 further demonstrate that LVQ outperforms not only the GNB benchmark but also logit regression by a modest margin. This suggests that the advantages of LVQ are not specific to the chosen comparator model but extend to other common classification approaches.

Finally, Annex Table A7 reports F1 scores, which confirm that the ranking of models is the

[3]The F1 score is the harmonic mean of precision and recall. Precision is the share of predicted turning points that were actually true turning points, while recall (equivalent to sensitivity) is the share of actual turning points that were correctly identified.

same as under balanced accuracy. This suggests that our conclusions are not sensitive to the choice of evaluation metric.

Taken together, these robustness checks provide reassurance that the findings reported in Sections 4.4 and 4.5 are stable across specifications and not driven by particular modelling choices.

5 Conclusion

Turning points in the monetary aggregates M1 and M3 can be identified with only a short delay using machine learning techniques. Applying LVQ and its distinction-sensitive extension (DSLVQ) to euro area data demonstrates that these methods provide both timely signals of turning points and insights into the sources of broad money growth.

Our findings have three main implications.

First, timeliness. LVQ identifies M1 turning points with a three-month lag, cutting in half the six-month confirmation delay of the Bry–Boschan algorithm. While not infallible, this improves the information set available to policymakers when assessing near-term shifts in economic activity.

Second, interpretability. DSLVQ not only matches LVQ's classification performance but also provides weights that indicate which sources of money creation matter most. Lending to households and firms emerges as a particularly strong driver, with Eurosystem asset purchases also playing an important role when they are present in the sample. This interpretability strengthens the link between turning point analysis and the transmission of monetary policy.

Third, robustness. The results are stable across specifications and parameter choices. Annex Tables A2–A5 show that modest variations in the learning rate, number of prototypes, classification threshold, and block length have little effect on balanced accuracy. Annex Tables A8 and A9 further demonstrate that LVQ outperforms not only GNB but also the logit regression. The overall conclusions are therefore not sensitive to the specific modelling choices or performance metrics employed.

These contributions matter for monetary analysis. Real M1 has long been recognised as a leading indicator of economic activity, while M3 and its counterparts provide information about the transmission of monetary policy. The ability to identify turning points more promptly, and to relate them to specific sources of money creation, enhance the analytical toolkit available to

central banks.

Looking ahead, further work could extend this framework by incorporating additional economic indicators, by distinguishing explicitly between peaks and troughs, or by testing the approach in different monetary regimes. As central banks continue to face rapidly changing environments, methods that combine machine learning flexibility with economic interpretability will be increasingly valuable for timely and robust policy assessment.

References

Ramón Adalid, Max Lampe, and Silvia Scopel. Monetary dynamics during the tightening cycle. *Economic Bulletin Boxes*, 8, 2024.

Gerhard Bry and Charlotte Boschan. Cyclical analysis of time series: Selected procedures and computer programs, 1971.

Christopher Clapham and James Nicholson. *The Concise Oxford Dictionary of Mathematics*. Oxford University Press, Oxford, 4th edition, 2009. ISBN 978-0199235940.

Björn Fischer, Michele Lenza, Huw Pill, and Lucrezia Reichlin. Monetary analysis and monetary policy in the euro area 1999–2006. *Journal of International Money and Finance*, 28(7):1138–1164, 2009.

Andrea Giusto and Jeremy Piger. Identifying business cycle turning points in real time with vector quantization. *International Journal of Forecasting*, 33(1):174–184, 2017.

Don Harding and Adrian Pagan. A comparison of two business cycle dating methods. *Journal of Economic Dynamics and Control*, 27(9):1681–1690, 2003.

Frank E Harrell, Robert M Califf, David B Pryor, Kerry L Lee, and Robert A Rosati. Evaluating the yield of medical tests. *Jama*, 247(18):2543–2546, 1982.

Fédéric Holm-Hadulla, Alberto Musso, Thomas Vlassopoulos, and Diego Rodriguez-Palenzuela. Evolution of the ECB's analytical framework. *ECB Occasional Paper*, (2021277), 2021.

Vita Jagric, Davorin Kracun, and Timotej Jagric. Does non-linearity matter in retail credit risk modeling? *Finance a Uver: Czech Journal of Economics & Finance*, 61(4), 2011.

Teuvo Kohonen. Self-organized formation of topologically correct feature maps. *Biological cybernetics*, 43(1):59–69, 1982.

Teuvo Kohonen. Improved versions of learning vector quantization. In *1990 ijcnn international joint conference on Neural networks*, pages 545–550. IEEE, 1990.

Teuvo Kohonen. The self-organizing map. *Neurocomputing*, 21(1-3):1–6, 1998.

Philip R. Lane. Modern monetary analysis, 6 2024. Speech at the 3rd Bank of Finland International Monetary Policy Conference "Monetary Policy in Low and High Inflation Environments".

Massimiliano Marcellino. Leading indicators. *Handbook of economic forecasting*, 1:879–960, 2006.

Viresh Moonasar. Credit risk analysis using artificial intelligence: evidence from a leading south african banking institution. Technical report, Research Report: Mbl3, 2007.

Alberto Musso. The predictive power of real M1 for real economic activity in the euro area. *Economic Bulletin Boxes*, 3, 2019.

Dimitris N Politis and Joseph P Romano. The stationary bootstrap. *Journal of the American Statistical association*, 89(428):1303–1313, 1994.

Martin Pregenzer, Gert Pfurtscheller, and Doris Flotzinger. Automated feature selection with a distinction sensitive learning vector quantizer. *Neurocomputing*, 11(1):19–29, 1996.

Katsuhiko Takahashi and Daisuke Nishiwaki. A class-modular GLVQ ensemble with outlier learning for handwritten digit recognition. In *Seventh International Conference on Document Analysis and Recognition, 2003. Proceedings.*, pages 268–272. IEEE, 2003.

A Annex

Data

The data sources for the time series and the details are summarized in Table A1. The M3 time series is corrected for a single one-off technical factor event which has no economical meaning and is also mentioned in the ECB of Press Release Monetary developments in the euro area: September [4] and October 2022[5].

[4]Press Release Monetary developments in the euro area: September 2022 footnote: "The September 2022 M3 figures include a large temporary position of the Eurosystem vis-à-vis a clearinghouse, classified within the "non-monetary financial corporations excluding insurance corporations and pension funds" sector. All the aggregates to which these deposits belong are inflated by this one-off technical factor." `https://www.ecb.europa.eu/press/stats/md/html/ecb.md2209~d7f36984da.en.html`

[5]Press Release Monetary developments in the euro area: October 2022 footnote: "The M3 annual growth rate in September 2022 would have stood at around 5.8% without a one-off technical factor related to a temporary deposit in the Eurosystem. That temporary deposit had been placed by a clearinghouse classified within the "non-monetary financial corporations excluding insurance corporations and pension funds" sector and was reversed in October 2022." `https://www.ecb.europa.eu/press/stats/md/html/ecb.md2209~d7f36984da.en.html`

Description	Source	Ticker
GDP - real gross domestic product	Eurostat	MNA.Q.Y.I9.W2.S1.S1.B.B1GQ._Z._Z._Z.EUR.LR.GY
M1 - narrow money, monetary aggregate, stock, flow	ECB	BSI.M.U2.Y.V.M10.X.1.U2.2300.Z01.E BSI.M.U2.Y.V.M10.X.4.U2.2300.Z01.E
M3 - broad money, monetary aggregate, stock, flow	ECB	BSI.M.U2.Y.V.M30.X.1.U2.2300.Z01.E BSI.M.U2.Y.V.M30.X.4.U2.2300.Z01.E
Loans to firms	ECB	BSI.M.U2.Y.U.A20T.A.4.U2.2240.Z01.E
Loans to households	ECB	BSI.M.U2.Y.U.A20T.A.4.U2.2250.Z01.E
Eurosystem net purchases	ECB	BSI.M.U2.N.C.A30.A.4.U2.2200.Z01.E BSI.M.U2.N.C.A30.A.4.U2.2100.Z01.E
Net external monetary flows	ECB	BSI.M.U2.Y.U.A80.A.4.U4.0000.Z01.E
Bank credit to general government	ECB	BSI.M.U2.Y.U.AT2.A.4.U2.2100.Z01.E BSI.M.U2.N.C.A30.A.4.U2.2100.Z01.E
Bank long-term liabilities	ECB	BSI.M.U2.Y.U.LT2.X.4.Z5.0000.Z01.E BSI.M.U2.N.A.L20.A.4.U2.2271.Z01.E BSI.M.U2.N.A.L20.L.4.U2.2271.Z01.E BSI.M.U2.n.a.L22.H.4.U2.2210.Z01.E

Table A1: Detailed description to the data series

Gaussian Naive Bayes classifier

The Gaussian Naive Bayes classification technique extends the Naive Bayes approach to continuous variables. It assumes each class follows a normal distribution and each parameter has an independent capacity of predicting the output variable. The combination of the prediction for all parameters is the final prediction that returns a probability of the dependent variable to be classified in each group. The final classification is assigned to the group with the higheest probability.

Following Bayes theorem, with y class variable and x_1 through x_n representing the features:

$$P\left(y|x_1,...,x_n\right) = \frac{P\left(y\right)P\left(x_1,...,x_n|y\right)}{P\left(x_1,...,x_n\right)}$$

with the conditional independence assumption

$$P\left(y|x_1,...,x_n\right) = \frac{P\left(y\right)\prod_{i=1}^{n}P\left(x_i|y\right)}{P\left(x_1,...,x_n\right)}$$

As $P(x_1, ..., x_n)$ is constant given the input, the following classification rule can be used:

$$P(y|x_1, ..., x_n) \propto P(y) \prod_{i=1}^{n} P(x_i|y)$$

$$\Rightarrow \hat{y} = \underset{y}{argmax} P(y) \prod_{i=1}^{n} P(x_i|y)$$

Robustness

The tables A2, A3, A4 and A5 show that variation initial learning rate α, the number of prototypes, the threshold deciding on the state and the block length for the boot-strapping have only a modest impact on the balance accuracy, thus proving that this methodology is fairly robust. The tables and show that LVQ is performing better than GNB and logit regression.
The second and third row in table A6 show that modest improvements in model accuracy are feasible in comparison to the first row by reducing the number variables in the training set. The information for the classification contained in the Eurosystem asset purchases improves the performance as illustrated by the fourth row further separating households and firms allows the classification to performance to improve. Table A7 shows the F1 scores of the same set of models as presented in 1. The results are comparable between the F1 and balance accuracy measure, hence the results are robust with regards to similar performance measures. The F1 score is defined as:

$$F1 = \frac{2TP}{2TP + FP + FN}. \tag{11}$$

Balanced accuracy			
	5th	mean	95th
Alpha 0.6	0.840	0.906	0.956
Alpha 0.5	0.844	0.913	0.966
Alpha 0.4	0.845	0.911	0.964
Alpha 0.3	0.852	0.915	0.966
Alpha 0.2	0.844	0.912	0.960
Alpha 0.1	0.847	0.917	0.966

Table A2: The performance measures balanced accuracy for different learning rates alpha for LVQ when considering M3 and 3 leads and lags

Balanced accuracy			
	5th	mean	95th
Number prototypes 118	0.825	0.912	0.978
Number prototypes 4	0.847	0.915	0.972
Number prototypes 3	0.852	0.914	0.972

Table A3: Performance measures of the 5th and 95th percentiles, and mean of the bootstrap distribution of balanced accuracy for LVQ M1 state classification with 3 leads and legs and different numbers of prototypes, 4 is the number used for the analysis

Balanced accuracy			
	5th	mean	95th
Threshold 0.9	0.808	0.894	0.954
Threshold 0.8	0.835	0.904	0.965
Threshold 0.7	0.850	0.908	0.959
Threshold 0.6	0.847	0.913	0.968
Threshold 0.5	0.854	0.916	0.960
Threshold 0.4	0.847	0.912	0.961

Table A4: Performance measures of the 5th and 95th percentiles, and mean of the bootstrap distribution of balanced accuracy for LVQ M1 state classification with 3 leads and legs and different thresholds

Balanced accuracy			
	5th	mean	95th
Years 6	0.843	0.912	0.964
Years 8	0.834	0.914	0.968
Years 10	0.836	0.904	0.955

Table A5: Performance measures of the 5th and 95th percentiles, and mean of the bootstrap distribution of balanced accuracy for LVQ M1 state classification with different block lengths

Balanced accuracy			
	5th	mean	95th
3 leads/lags DSLVQ with all counterparts	0.846	0.877	0.914
3 leads/lags LVQ with HH, Firms, Eurosystem	0.818	0.903	0.974
3 leads/lags DSLVQ with HH, Firms, Eurosystem	0.868	0.888	0.907
3 leads/lags DSLVQ with HH, Firms	0.826	0.847	0.873
3 leads/lags DSLVQ with Private Sector, Eurosystem	0.833	0.855	0.880

Note: HH abbreviates households, and Private sector combines firms and household to a single indicator.

Table A6: Performance measures of the 5th and 95th percentiles, as well as the mean of the bootstrap distribution of balanced accuracy for LVQ M3 state classification from 1 to up to 5 leads and legs

F1 score			
	5th	mean	95th
GNB with 3 leads/lags	0.772	0.890	0.955
5 leads/lags	0.912	0.958	0.993
4 leads/lags	0.879	0.941	0.980
3 leads/lags	0.858	0.924	0.967
2 leads/lags	0.823	0.904	0.959
1 leads/lags	0.780	0.859	0.939

Table A7: F1 performance measures of the 5th and 95th percentiles, as well as the mean of the bootstrap distribution of balanced accuracy for GNB and LVQ M1 state classification from 1 to up to 5 leads and legs

Balanced accuracy			
	5th	mean	95th
LVQ	0.854	0.916	0.963
GNB	0.778	0.876	0.949
LOGIT	0.841	0.894	0.979

Table A8: Performance measures of the 5th and 95th percentiles, and mean of the bootstrap distribution of balanced accuracy for LVQ, GNB and logit regression M1 state classification with 3 leads and legs

Balanced accuracy			
	5th	mean	95th
LVQ	0.866	0.894	0.915
GNB	0.785	0.845	0.903
LOGIT	0.811	0.863	0.907

Table A9: Performance measures of the 5th and 95th percentiles, and mean of the boot- strap distribution of balanced accuracy for LVQ, GNB and logit regression M3 state classification with 3 leads and legs

Acknowledgements

We would like to thank Miguel Boucinha and Jesus Fernández-Villaverde for their helpful comments provided in the context of the ECB AI in Economics Workshop.

Max Lampe (Corresponding author)
European Central Bank, Frankfurt am Main, Germany; email: Max.Lampe@ecb.europa.eu

Ramón Adalid (Corresponding author)
European Central Bank, Frankfurt am Main, Germany; email: Ramon.Adalid@ecb.europa.eu

PDF ISBN 978-92-899-7515-5 ISSN 1725-2806 doi: 10.2866/5068630 QB-01-25-263-EN-N