



Discussion Paper

Deutsche Bundesbank
No 39/2024

Benchmarking short term forecasts of regional banknote lodgements and withdrawals

Benedikt Sonnleitner

(Fraunhofer IIS, Nürnberg, and
Otto-Friedrich-Universität Bamberg)

Jelena Stapf

(Deutsche Bundesbank)

Kai Wulff

(Deutsche Bundesbank)

Editorial Board:

Daniel Foos
Stephan Jank
Thomas Kick
Martin Kliem
Malte Knüppel
Christoph Memmel
Hannah Paule-Paludkiewicz

Deutsche Bundesbank, Wilhelm-Epstein-Straße 14, 60431 Frankfurt am Main,
Postfach 10 06 02, 60006 Frankfurt am Main

Tel +49 69 9566-0

Please address all orders in writing to: Deutsche Bundesbank,
Press and Public Relations Division, at the above address or via fax +49 69 9566-3077

Internet <http://www.bundesbank.de>

Reproduction permitted only if source is stated.

ISBN 978-3-98848-014-9

ISSN 2941-7503

Non-technical summary

Research Question

Cash is the most frequently used means of payment at the point of sale in Germany. The Bundesbank operates a nationwide branch network to provide credit institutions and retailers with high-quality cash at all times. These customers' lodgements and withdrawals constitute the stock of banknotes of the regional branches. A more accurate forecast of lodgements and withdrawals might allow for more efficient inventory management and transportation planning, thus ensuring cash availability at all times. The performance of different data driven methods is thereby dependent on the unique characteristics of the dataset. To provide guidelines on the choice of methods in application, we benchmark different forecasting methods that represent the state of research in forecasting and are used in industry practice.

Contribution

Up to now, the Deutsche Bundesbank does not employ formal data driven cash demand forecasting on branch and denomination granularity. We use the daily transactions by six regional branches to explore the accuracy and inventory performance of three statistical, and two machine learning based forecasting methods. We benchmark each against a seasonal naive forecast. For accuracy, we evaluate point forecasts and uncertainty forecasts. For inventory performance, we measure service level (i.e., out of stock periods) in relation to held stock for varying target service levels.

Results

DeepAR, a neural network, delivers the best results in terms of accuracy and inventory performance and ETS, a statistical method, ranks second. Employing one these methods in a formalized forecasting process should allow for efficiency gains in transport and inventory planning, as it allows for fewer held inventory at the same service level. Although the central bank's objective function in the storage of cash differs from that of industrial enterprises due to a clear focus on the ability of the branch network to pay out at all times, this study offers an interesting starting point for improving processes in inventories and logistics.

Nichttechnische Zusammenfassung

Fragestellung

Bargeld ist das meistverwendete Zahlungsmittel an der Ladenkasse in Deutschland. Dies erfordert seitens der Bundesbank ein flächendeckendes Filialnetz, um Kreditinstitute und Einzelhändler jederzeit mit hochwertigem Bargeld zu versorgen. Die Einzahlungen und Abhebungen der Kunden bilden dabei den Banknotenbestand der regionalen Filialen. Eine genauere Vorhersage dieser Beträge kann eine effizientere Lagerhaltungsplanung und Transportplanung ermöglichen. Die Leistung verschiedener Prognosemethoden hängt dabei von den Merkmalen des jeweiligen Datensatzes ab. Um Leitlinien für die Wahl der Methoden in der Anwendung zu geben, vergleichen wir verschiedene Prognosemethoden.

Beitrag

Bisher verwendet die Deutsche Bundesbank keine formale datengestützte Bargeldbedarfsprognose auf Filial- und Stückelungsgranularität. Wir benutzen die täglichen Transaktionen von sechs regionalen Filialen, um die Prognosegenauigkeit und die Auswirkungen auf das Lagerhaltungsmanagement von drei statistischen und zwei auf maschinellem Lernen basierenden Prognosemethoden zu untersuchen. Wir vergleichen jede Methode mit einer saisonalen naiven Prognose in Bezug auf Genauigkeit und Bestandsleistung. Für die Genauigkeit bewerten wir Punkt- und Unsicherheitsprognosen. Für die Bestandsleistung messen wir das Serviceniveau (d. h. die Zeiten, in denen keine Ware vorrätig ist) im Verhältnis zum gehaltenen Bestand bei unterschiedlichen Ziel-Serviceniveaus.

Ergebnisse

DeepAR, ein neuronales Netz, liefert das beste Ergebnis in Bezug auf Genauigkeit und Auswirkungen auf das Lagerhaltungsmanagement, ETS, eine statistische Methode, liefert die zweitbesten Ergebnisse. Außerdem verbessern alle berücksichtigten Prognosemethoden die naive Prognose erheblich. Der Einsatz einer dieser Methoden in einem formalisierten Prognoseprozess sollte Effizienzgewinne bei der Transport- und Bestandsplanung ermöglichen, da bei gleicher Auszahlungsbereitschaft weniger Bestände vorgehalten werden können. Obwohl sich die Ziele von Zentralbanken bei der Lagerung von Bargeld von den Zielen von Industrieunternehmen unterscheiden, da der Fokus auf der Fähigkeit liegt, jederzeit Auszahlungen vorzunehmen, bietet diese Studie einen interessanten Ansatzpunkt für die Verbesserung der Prozesse in Lagerhaltung und Logistik.

Benchmarking short term forecasts of regional banknote lodgements and withdrawals *

Benedikt Sonnleitner

Fraunhofer IIS, Nürnberg, Otto-Friedrich-Universität Bamberg

Jelena Stapf

Deutsche Bundesbank, Frankfurt

Kai Wulff

Deutsche Bundesbank, Frankfurt

Abstract

Among the most important tasks of central banks is to ensure the availability of cash to credit institutions and retailers. Forecasting the demand for cash on a granular level is crucial in the process to keep logistics costs low, while being resilient to demand or supply shocks. Whereas to date, cash forecasts with central banks mostly comprise structural models to define banknote production for the coming years, our contribution is to combine features of macro level forecasting with more granular and short term regional forecasts methods. We show in an inventory simulation, that elaborate forecasting methods on granular level can substantially improve inventory performance for this use-case. To guide the implementation of a forecasting process at the Bundesbank, we benchmark statistical and machine learning methods on demand and supply of cash, using anonymized data on transactions of six regional branches of Deutsche Bundesbank. We use a pseudo out of sample predictive performance framework to evaluate the accuracy of our forecasts and perform an inventory cost simulation. We find that (i) DeepAR outperforms the other benchmarks substantially on all data sets. (ii) ETS, ARIMA, and DeepAR clearly outperform the naive benchmark in terms of accuracy across all data sets, and inventory performance.

Keywords: Global learning, Forecasting, Machine Learning

JEL classification: E31, G21.

*Contact address: Benedikt Sonnleitner E-Mail: benedikt.sonnleitner@iis.fraunhofer.de. We thank Nikolaus Bartsch, Daniel Goll and Ursula Neumann for helpful comments. Any remaining errors are the responsibility of the authors. The views expressed in this paper are those of the authors and do not necessarily coincide with the views of the Deutsche Bundesbank or the Eurosystem.

1 Introduction

Cash payments at the point of sale are still the most important means of payments in Germany ([Bundesbank, 2022](#)). Customers value cash payments owing to their simplicity, finality, inclusiveness, independence of technical and data infrastructure, privacy and the control over their expenses. The Bundesbank runs a nationwide network of 31 branches to provide credit institutions and retailers with high quality banknotes at all times. Accordingly, in all of the Bundesbank branches, a substantial amount of inventory is held. Together with customers' cash demand, i.e. lodgements and processing capacities reduced by withdrawals, these constitute the stock of banknotes at the regional branches. To ensure smooth and stable supply of banknotes for each denomination, transport and inventory planning relies on forecasted regional cash demand.

The current planning process uses a weekly forecast of withdrawals and lodgements of individual branches without a standardized forecasting process, where forecasts are done implicitly manually. Based on these, the logistics centre for banknotes of the Bundesbank plans the transports for filling or disposal. Thereby it takes into account various constraints such as deliveries from printing works, cross border and international transports and the schedule of the accompanying regional police forces.

Improving the current forecast process by using formalized and accurate forecasting models might therefore help to improve on the ensurance of availability of cash, reduce inventory, allocate resources, gain advance time for planning or save transports. Our central contribution is to introduce this new forecasting use-case that is addressed by central banks across the globe, but that is not dealt with in academic forecasting literature yet. We offer insights on the value that state of the art forecasting methods can bring to it. An apparent difference to forecasting in logistics for central banks compared to most industries is the absolute crucial availability of banknotes - i.e. a 100 percent service level is required at the branches: "Most obviously guardians of monetary stability must not run out of their most trusted instrument" ([Hinge, 2022](#)).

There are two other use-cases for cash demand forecasting that relate to the introduced one: First, the issuers of legal tender currency make yearly or longer-term projections for production requirements. These encompass break-downs for the denominational split of the banknotes to be produced and incorporates replacement ratios for so called unfit banknotes, i.e. torn and soiled banknotes which cannot be paid out any longer. For example [Miller \(2017\)](#) and [Bartzsch, Brandi, de Pastor, Devigne, Maddaloni, Restrepo, and Sene \(2023\)](#) use a broad range of structural time series models. Our use-case differs since we forecast for cash management on operative level, which requires shorter forecast horizons (21 days vs. several months), and spatially more granular forecasts (local branches vs. whole countries).

Second, a strand of literature evolves around replenishment optimization problems for ATMs. This has been the foundation of the NN5 competition ([Crone, 2008](#)), whose data set is since used as a benchmark data set in several works (for example [Ben Taieb, Bontempo, Atiya, and Sorjamaa, 2012](#); [Venkatesh, Ravi, Prinzie, and den van Poel, 2014](#)).

Also beyond the competition, the use-case has attracted forecasting researchers, for example [Riabykh, Suleimanov, Surzhko, Konovalikhin, and Ryazanov \(2022\)](#) benchmarked a Machine Learning pipeline against several statistical approaches on ATM data from a large Russian bank, and [Fallah-Tafti, Aghaaminiha, Akbarghanadian, and Weckman \(2022\)](#) analyze how the Covid19 pandemic influenced the ranking on prediction accuracy for several methods on ATM data by a bank in Tehran. In our study we consider cash demand for local central bank branches that deliver cash to commercial banks instead of consumers. This introduces potentially different demand patterns and adds an additional layer of forecast complexity compared to ATM-forecasting by incorporating features from long term demand forecasting such as recycling and replacement ratio issues. Bundesbank branches might generate out-payable banknotes by processing paid-in banknotes on the other day. The difference between processed banknotes and banknotes that must be destroyed because of soil and stain issues, i. e. the replacement ratio, influences the stock of out-payable banknotes and therefore delays replenishment.

To date in demand forecasting software applications, statistical and machine learning approaches are most commonly used. Models of both types can be trained by fitting one model per time series, or a single model across multiple series. The former is referred to as local or per series training and allows to focus on the very specific patterns of the respective time series. The latter is commonly referred to as cross learning or global learning and requires that a single model has to learn the patterns of multiple time series ([Januschowski, Gasthaus, Wang, Salinas, Flunkert, Bohlke-Schneider, and Callot, 2020](#); [Smyl, 2020](#); [Montero-Manso and Hyndman, 2021](#)). Traditionally, the most prevalent statistical models, i.e. ETS and ARIMA, are trained locally ([Hyndman and Athanasopoulos, 2018](#); [O’Hara-Wild, Hyndman, and Wang, 2021](#)). In contrast, Machine Learning models, such as LightGbm and DeepAR blossomed specifically when trained globally, as these require more available data to avoid excessive overfitting ([Makridakis, Spiliotis, and Assimakopoulos, 2020](#); [Salinas, Flunkert, Gasthaus, and Januschowski, 2020](#); [Montero-Manso and Hyndman, 2021](#); [Kunz, Birr, Raslan, Ma, and Januschowski, 2023](#)).

The data that we use in cash demand forecasting is small compared to large data sets as they were for example used in the M5 competition ([Makridakis et al., 2020](#)), yet it is relatively large in comparison to many macroeconomic studies. Therefore we benchmark the accuracy of the local and global approaches on the data set of cash demand for six regional branches with the Deutsche Bundesbank. As forecasting accuracy is only opaquely connected with inventory performance (see for example [Kourentzes, Trapero, and Barrow, 2020](#)), we further investigate the difference in inventory cost induced by the various methods by a simulation approach.

In [Section 2](#) we present an overview of the evaluated forecasting methods with a focus on statistical versus machine learning methods. We then provide further insights on the specific data challenges for cash distributions in [Section 3](#). In [Section 4.1](#) we outline our evaluation framework for the accuracy benchmark, thereby pointing out the need to evaluate pseudo out of sample predictive performance in comparison to in sample evaluations which is the dominant evaluation scheme for many macroeconomic causal studies. We present our results subsequently. In [Section 4.2](#) we describe the inventory performance evaluation framework, again followed by the results. We conclude with a

summary of our main findings, limitations of the study and suggestions of future research in Section 5.

2 Forecasting methods

The demand for cash on branch and denomination level varies strongly over time, as illustrated in Section 3. In line with most forecasting tasks, the true data generating process is unknown, i.e., we do not know the actual conditional distribution of cash demand (Chatfield, 2000). Accordingly, we fit the forecast methods based on past observed data. The evaluated forecasting methods thereby vary concerning their underlying assumptions, ultimately leading to different forecasts. In the following, we describe the evaluated forecasting methods.

2.1 Statistical versus Machine Learning forecasting

In statistical forecasting, there is an underlying assumption about the data generating process, i.e., it is imposed that the time series follows some structure, including randomness, whose parameters are estimated based on data. Empirically for example exponential smoothing and ARIMA models have been shown to be good local approximations of many demand processes (Hyndman, 2008; Hyndman and Athanasopoulos, 2018). In practice statistical models are usually combined with some model selection procedure. As we typically see a large amount of time series in business forecasting, doing this manually often becomes cumbersome. Instead, the most used time series forecasting packages fit multiple models from a model family and afterwards pick the one which minimizes an information criteria, mostly AIC or BIC (see for example Hyndman and Khandakar, 2008).

By contrast, machine learning methods impose no, or few assumptions about the data generating process (Hornik, Stinchcombe, and White, 1989; Schäfer and Zimmermann, 2006; Barker, 2020). This leads to challenges in the estimation efficiency, as the hypotheses space becomes large (i.e., it can be chosen from a large amount of functions that fit the data similar well), and hence the estimation can be more prone to overfitting than in statistical forecasting. However, training in a global fashion, i.e., across many time series, offers a remedy, as the method is forced to learn patterns that apply to the whole data set. The most prevalent machine learning approaches are on one hand decision tree based methods, such as Random Forests, XgBoost, and LightGBM (Makridakis et al., 2020; Januschowski, Wang, Torkkola, Erkkilä, Hasson, and Gasthaus, 2022), on the other hand various neural networks, with a dynamic development of new architectures are used (see for example Smyl, 2020; Salinas et al., 2020).

Empirically, we see mixed result concerning the benefit of machine learning approaches over statistical approaches: historically there was a broad consensus that simple (statistical) forecasting methods should be preferred over complex ones (Makridakis and Hibon, 2000). However especially in recent years evidence has intensified that machine learning

models can improve accuracy over statistical approaches. Especially if machine learning models are trained in a global fashion, i.e., over multiple time series simultaneously, thereby reducing overfitting (Makridakis et al., 2020; Makridakis, Spiliotis, Assimakopoulos, Chen, Gaba, Tsetlin, and Winkler, 2020). Nevertheless, also if machine learning models are globally applied, the benefit seems to lie in specific architectures. For example Hewamalage, Bergmeir, and Bandara (2021) applied six commonly used recurrent neural network architectures in combination with different preprocessing and training procedures (in total 90 experiments), and found that none of the models significantly outperformed the statistical benchmarks (ETS and ARIMA) on five out of six large datasets. See Godahewa, Bergmeir, Webb, Hyndman, and Montero-Manso (2021) for a dynamically updated overview of the performance of various models on multiple data sets.

The discussion shows that there is no forecasting model nor training-scheme that clearly outperforms all other models/ training-schemes. Accordingly for our study, we select some representative statistical and machine learning models that reflect upon the general state of the art in forecasting research. These allow us to quantify the value of modern forecasting approaches for the use-case at hand. Our goal is thereby the application and evaluation on the specific use-case, not to draw conclusions about the general performance of statistical versus machine learning models.

2.2 Benchmarked methods

In general, the performance of forecasting methods is highly dependent on the characteristics of the data set on which they are evaluated, encouraging a “horses for courses” approach, where the forecast method is chosen based on the specific data set (Petropoulos, Makridakis, Assimakopoulos, and Nikolopoulos, 2014). Accordingly, in our study, we evaluate various models on four data sets for physical cash distribution, to provide insights which of these should be employed in practice and whether they can provide value over simple benchmarks. Our data sets consist of 6 to 42 time series with smooth demand on a daily frequency. That means they are small in comparison to most data sets where global machine learning models dominated, and it does not exhibit mostly intermittent time series which was the case for example in the M5 competition (Makridakis et al., 2020). In the following, we provide a brief overview of the evaluated forecast methods. For a detailed introduction we refer to the referenced literature.

Seasonal Naive: the forecasted value is the last observed value on the respective seasonal period. For example the forecasted value for the next H Tuesdays would be the last observed value of a Tuesday. We include this as benchmark, as it imposes no model assumptions, besides daily seasonality, which is very apparent in our datasets.

ETS encompasses a family of models, which relies on decomposing a time series in a level, trend, and seasonality component. Depending on the data, only some of the components are included, requiring model selection beforehand, which is in our case done by selecting the model that minimizes the in sample AIC (Hyndman and Khandakar, 2008). If only the level component is included, ETS reduces for example to simple exponential smoothing,

given by $\hat{y}_t = \alpha y_t + (1-\alpha)\hat{y}_{t-1}$, with the smoothing parameter α , the actual observations y_t , with $t \in 1, \dots, T$, and its forecast $\hat{y}$. α is thereby estimated for a specific time series, using only data of this time series. The trend and/ or the seasonality component are smoothed in a similar fashion if included, and accordingly the relation between the parameters and the forecast remains linear in all cases, facilitating quick estimations that are proven optimal, for the DGP assumption imposed by the model.

ARIMA as well encompasses a family of models, consisting of autoregressive terms (i.e., the lagged actual observations of the time series), smoothing terms (i.e., the lagged previous forecast errors by the model), and n -th order differencing of the original time series, to obtain stationarity (Hyndman and Athanasopoulos, 2018). Similar to ETS, not all components are necessarily included, and model selection has to be done as well in similar fashion. An ARMA forecast method with a lag-one autoregressive term, a lag-one moving average term, and an intercept is then for example given as $\hat{y}_t = c + \phi_1 y_{t-1} + \theta_1 (y_{t-1} - \hat{y}_{t-1})$, where the parameters c , ϕ_1 , and θ_1 need to be estimated. Again, the models remain linear in their coefficients in all cases and are trained per single time series.

As both, ARIMA and ETS, rely on an assumption about the data generating process, prediction intervals can be calculated based on analytic formulae and the one step ahead prediction errors for the additive ETS models and most ARIMA models. The seasonal naive forecast can be seen as a special case of ARIMA model, where as well an analytic formula for prediction intervals is available. In our experiments, we rely on the *fable* package by O’Hara-Wild et al. (2021), which uses these when available for the chosen model and otherwise relies on simulated future paths (Hyndman and Athanasopoulos, 2018). If not otherwise noted we use the standard settings of the package. Svetunkov (2023, chapter 18.1) describes how simulation paths for uncertainty estimates with ETS can be obtained. For a more extensive introduction into ETS and ARIMA, see Hyndman (2008); Hyndman and Athanasopoulos (2018); Svetunkov (2023). Specifically Hyndman and Athanasopoulos (2018, chapters 8.4 and 9) provide the equations, model selection and parameter estimation methods that we use for the ETS, and ARIMA models.

DeepAR combines distributional assumptions with a neural network, usually trained in a global fashion (Salinas et al., 2020). Instead of forecasting the expectation of a time series, as ETS and ARIMA do, the parameters of an assumed distribution are forecasted. Accordingly, the loss function corresponds to the likelihood of the forecasted distributional parameters, given the training data. For an assumed normal distribution, the loss would be for example calculated as $l(y_t|\mu_t, \sigma_t) = \mathcal{N}_{\mu_t, \sigma_t}(y_t) = \frac{1}{\sigma_t \sqrt{2\pi}} \exp\left(\frac{-(y_t - \mu_t)^2}{2\sigma_t^2}\right)$, with the conditional expectation μ_t and the conditional variance σ^2 , both output by the final nodes. Accordingly, by default, we can obtain prediction intervals, quantiles or any other form of uncertainty estimation from the models trained with this loss and final node. We combine it with a recurrent network, as originally proposed by Salinas et al. (2020) and a simple feed forward network, as implemented in the gluonts package (Alexandrov, Benidis, Bohlke-Schneider, Flunkert, Gasthaus, Januschowski, Maddix, Rangapuram, Salinas, Schulz, Stella, Türkmen, and Wang, 2020). In the following we refer to the former as DeepAR, and the latter as Multi Layer Perceptron (MLP).

MLPs are the simplest form of neural networks. They consist of stacked linear and non-linear functions that transform a feature vector (the independent variables) into an output vector (the dependent variable). Thereby the output of one layer serves as input for the next layer and finally the prediction. A layer usually consists of several nodes. If we consider a single layer perceptron with a single node, the input of the node is given by a weighted sum of the features, in our case lagged variables y_{t-i} , $i \in 1, \dots, K$ and some constant: $Z = w_{1,0} + \sum_{i=1}^K w_{1,i} y_{t-i}$. This is then transformed by a non-linear activation function, for example a sigmoid function, and another linear weight to form the output: $y_t = w_{2,0} \frac{\exp(Z)}{1 + \exp(Z)}$. The weights $w_{j,i}$ are fitted based on data during training. As the weights are embedded in a non-linear function, the model has non-linear coefficients, preventing an optimal closed form solution (Hyndman and Athanasopoulos, 2018).

Recurrent neural networks also embed linear and non-linear functions, but in addition they use a state vector which models the dynamics of the time series over time. Such, they can be seen as a non-linear adaptation of state space models such as the introduced ETS models. In our study we consider a sequence-to-sequence recurrent neural network with LSTM cells, as originally proposed by Salinas et al. (2020). See as well Hewamalage et al. (2021) for an introduction. In our experiments, we rely on the mxnet implementation in Alexandrov et al. (2020) for the neural networks. For both neural networks, we form an ensemble of ten independently initialized models, as this can improve accuracy and desensitizes the results against performance differences due to random initialization. We train DeepAR and the MLP globally per data set. The other two methods, ARIMA and ETS are trained locally. Table 5 in the Appendix summarizes the hyper parameters for the two machine learning models.

2.3 Feature engineering

Many of the considered time series exhibit apparent seasonality on a daily level, around holidays and during christmas season. Further, domain experts suggested yearly and monthly seasonality. Accordingly, we encode the following features: (i) dummy variables per weekday, (ii) per month, (iii) for public holidays at which no cash is distributed, (iv) for the week before Christmas, the days between 26th of December, the first of January, and the first 5 workdays in a year, and (v) triangular variables (sinus and cosinus) for the day per month, and the week per year. We add the features for each forecasting model that can by default use these (MLP, ARIMAX, DeepAR). We did not include the features for ETS, as in its standard implementation it is univariate. Additionally, we add a univariate ARIMA as benchmark.

3 Data

To facilitate inventory management and transportation planning, a forecast of the net demand for banknotes per denomination and branch is required. We cannot observe this directly, instead it results from the difference between withdrawals and reissuable

lodgements per denomination and branch:

$$\text{net demand} = \text{withdrawals} - \text{reissuable lodgements}.$$

For withdrawals, historical data of cash orders is readily available as professional cash handlers are required to submit their orders per branch and in number of notes per denomination, and we forecast it with an horizon of 7 days. There is no required lead time on cash orders. Reissuable lodgements on the other hand need to be calculated based on different forecasts, as we first observe the overall value of lodged notes, and only after these are processed, the denomination and whether they are reissuable becomes known. However, between lodgement and processing there is a time lag of up to two weeks, depending on the current local workload. Accordingly, in line with the domain experts, we forecast per branch: (i) the value of deposits with an horizon of 7 days; (ii) the share of the respective denomination with an horizon of 21 days; (iii) the share of reissuable banknotes per denomination with an horizon of 21 days; and (iv) the cash demand per denomination with an horizon of 7 days. From the first three forecasts, a forecast for the lodged reissuable number of notes per denomination and branch can be calculated, and then from the cash demand forecast also the net demand per denomination and branch. In the following we focus our experiments on the four mentioned forecasts, and evaluate them independently. Table 1 provides an overview of the used data.

Table 1: Data characteristics

dataset	granularity	interval	from - to	# time series	points per series
cash orders	daily	$[0, \infty)$	02.01.2017 - 19.05.2022	36	1964
lodgements overall value	daily	$[0, \infty)$	02.01.2017 - 19.05.2022	6	1964
lodgements share per denomination	daily	$[0, 1]$ (ratio)	02.01.2017 - 19.05.2022	42	1964
lodgements share reissuable	daily	$[0, 1]$ (ratio)	02.01.2017 - 19.05.2022	36	1964

The time series in all considered data sets vary over time and exhibit apparent daily seasonality. Figure 1 shows exemplary the distribution of daily cash orders per weekday for Villing-Schwenningen and the 20 Euro denomination. The demand for cash correlates highly between different branches and different banknotes. Tables 2 and 3 shows exemplary the correlation matrix for the different notes in a single branch, and the correlation matrix between branches for the five Euro denomination, respectively. This high correlation suggests that learning across series might be beneficial, as the demand patterns seem to be very similar across the different time series.

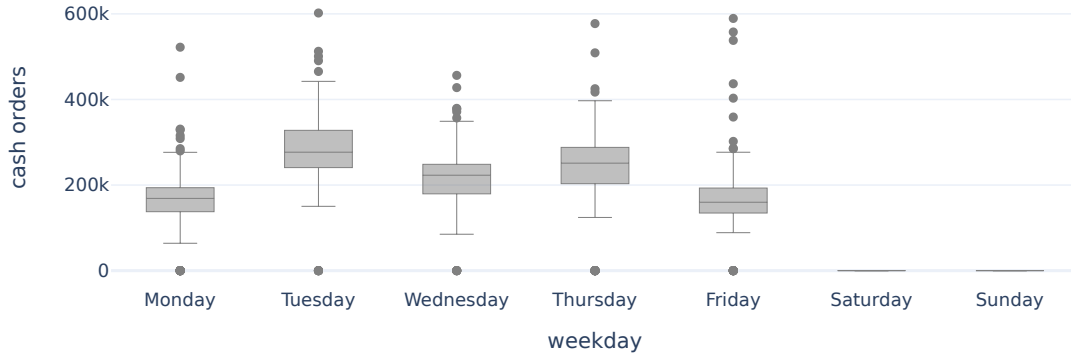


Figure 1: Distribution of cash orders for 20 Euro notes per weekday in Villing-Schwenningen

Table 2: Pearson correlation between cash demand of different denominations in one branch

denomination	five	ten	twenty	fifty	hundred	two hundred
five	1.00	0.97	0.95	0.96	0.87	0.40
ten	0.97	1.00	0.99	0.99	0.92	0.46
twenty	0.95	0.99	1.00	0.99	0.91	0.46
fifty	0.96	0.99	0.99	1.00	0.93	0.50
hundred	0.87	0.92	0.91	0.93	1.00	0.62
two hundred	0.40	0.46	0.46	0.50	0.62	1.00

Table 3: Pearson correlation between cash demand of five euro notes across branches

	Branch 1	Branch 2	Branch 3	Branch 4	Branch 5	Branch 6
Branch 1	1.00	0.76	0.84	0.83	0.77	0.84
Branch 2	0.76	1.00	0.79	0.79	0.73	0.75
Branch 3	0.84	0.79	1.00	0.93	0.93	0.91
Branch 4	0.83	0.79	0.93	1.00	0.91	0.91
Branch 5	0.77	0.73	0.93	0.91	1.00	0.91
Branch 6	0.84	0.75	0.91	0.91	0.91	1.00

Note that it is also common practice to send unprocessed banknotes to other branches to make better use of spare processing capacity, which can further extend the mentioned time lag, and more importantly, blur the traceability of the deposits to their original branches. We do not consider this in our forecast or evaluation scheme. Further, 500

Euro notes are only considered for the share of of notes per lodgement, as these are not issued any more, and are thus not relevant for the other data sets.

4 Evaluation set up and empirical findings

We are interested in finding the forecast method that minimizes the cost for physical cash distribution while allowing for a high service level, when planned upon it. Accordingly, we are not primarily interested in finding causal relationships but instead focus on forecast accuracy and cost of decisions, given the forecast of the respective method. A full end to end simulation of forecasts and overall planning (including transportation scheduling, routing, and inventory management) is not possible due to data confidentiality reasons. Instead we provide (i) an accuracy benchmark over various metrics and (ii) an inventory simulation assuming a simplified inventory policy, as forecast accuracy is only loosely connected with inventory performance and we believe this to be a better proxy for the overall process than only accuracy (Kourentzes et al., 2020). For both approaches, we evaluate over rolling origins, i.e., we split the data set iteratively in training and test set, thereby desensitizing the evaluation of special time periods (Tashman, 2000). Further we evaluate over all time series in the four data sets, to obtain a representative result. The required horizons per data set are dictated by the business processes, (see Section 3) and we average across them for the accuracy metrics. For the inventory simulation the respective cumulative lead time demand quantile is the input for the inventory policy. In the following, we describe both evaluation procedures.

4.1 Accuracy evaluation

Table 4 gives an overview over the dimensions of evaluation. We evaluate each time series across thirty equally spaced rolling origins: $O = \{01.03.2021, 13.03.2021, 26.03.2021, \dots, 28.02.2021\}$. Together with the different forecast horizon and number of time series, we accordingly evaluate $6 * 30 * 7 = 1.260$ errors on the smallest, and $42 * 30 * 21 = 26.460$ errors on the biggest data set, which we assume statistically suffice for evaluating the six models under consideration. The different error metrics account for different central tendencies of the predictive distribution: The mean squared error (MSE) accounts for the mean, the mean absolute error (MAE) accounts for the median, and the pinball loss (PIN) for the respective quantile (Gneiting and Raftery, 2007; Kolassa, 2020). Further we evaluate forecast bias, i.e. systematic over- or underforecasting with the mean error (ME). All the evaluated forecast methods assume a normal distribution, where the mean estimate equals the median estimate, and accordingly we use the mean forecast for both, the MAE and the MSE evaluation. To account for the different scales of time series, and facilitate comparison across time series, we scale each metric by the respective accuracy of the seasonal naive forecast per series. In the following we provide the equation for each of the used error metrics. We denote the expectation forecast at origin t with forecast horizon h for y_{t+h} by $\hat{y}_{t,h}$, and the γ -quantile forecast as $\hat{q}_{t,h}^{[\gamma]}$. H is the set of evaluated

origins and horizons respectively:

$$\begin{aligned}
\text{AME} &= \left| \frac{1}{|O|} \frac{1}{|H|} \sum_{t \in O, h \in H} y_{t+h} - \hat{y}_{t,h} \right| & \text{relAME} &= \frac{\text{AME}_m}{\text{AME}_{\text{seasonal naive}}} \\
\text{MSE} &= \frac{1}{|O|} \frac{1}{|H|} \sum_{t \in O, h \in H} (y_{t+h} - \hat{y}_{t,h})^2 & \text{relMSE} &= \frac{\text{MSE}_m}{\text{MSE}_{\text{seasonal naive}}} \\
\text{MAE} &= \frac{1}{|O|} \frac{1}{|H|} \sum_{t \in O, h \in H} |y_{t+h} - \hat{y}_{t,h}| & \text{relMAE} &= \frac{\text{MAE}_m}{\text{MAE}_{\text{seasonal naive}}} \\
\text{mPIN}^{[\gamma]} &= \frac{1}{|O|} \frac{1}{|H|} \sum_{t \in O, h \in H} \text{PIN}(y_{t+h}, \hat{q}_{t,h}^{[\gamma]}) & \text{relPIN} &= \frac{\text{PIN}_m^{[\gamma]}}{\text{PIN}_{\text{seasonal naive}}^{[\gamma]}} \\
\text{PIN} &= \begin{cases} (y_{t+h} - \hat{q}_{t,h}^{[\gamma]})\gamma, & \text{if } y_{t+h} \geq \hat{q}_{t,h}^{[\gamma]} \\ (\hat{q}_{t,h}^{[\gamma]} - y_{t+h})(1 - \gamma), & \text{if } y_{t+h} < \hat{q}_{t,h}^{[\gamma]} \end{cases}
\end{aligned}$$

Table 4: Summary of evaluation setup

dataset	origins	horizons	metrics	# time series
cash orders	30 origins, uniformly between 01.03.2021 and 28.02.2022	1-7	rMSE, rMAE, rAME, rPIN	36
lodgements overall value	30 origins, uniformly between 01.03.2021 and 28.02.2022	1-7	rMSE, rMAE, rAME, rPIN	6
lodgements share per note	30 origins, uniformly between 01.03.2021 and 28.02.2022	1-21	rMSE, rMAE, rAME, rPIN	42
logdements share reissuable	30 origins, uniformly between 01.03.2021 and 28.02.2022	1-21	rMSE, rMAE, rAME, rPIN	36

Figures 2, 3, 4, and 5 summarize the results of the accuracy benchmark study. Further, we provide the respective number in Tables 6, 7, 8, and 9 in the Appendix. We find that in most cases, most elaborated forecast methods outperform the naive benchmark. Further we find:

- DeepAR is across all four datasets the most accurate method in terms of bias, mean forecasts, and the 90 % quantile forecast. ETS performs similar to ARIMA across data sets.
- Counter intuitively, even though a logistic distribution should fit the lodgements, share reissuable and lodgements, share denominations datasets better, DeepAR with

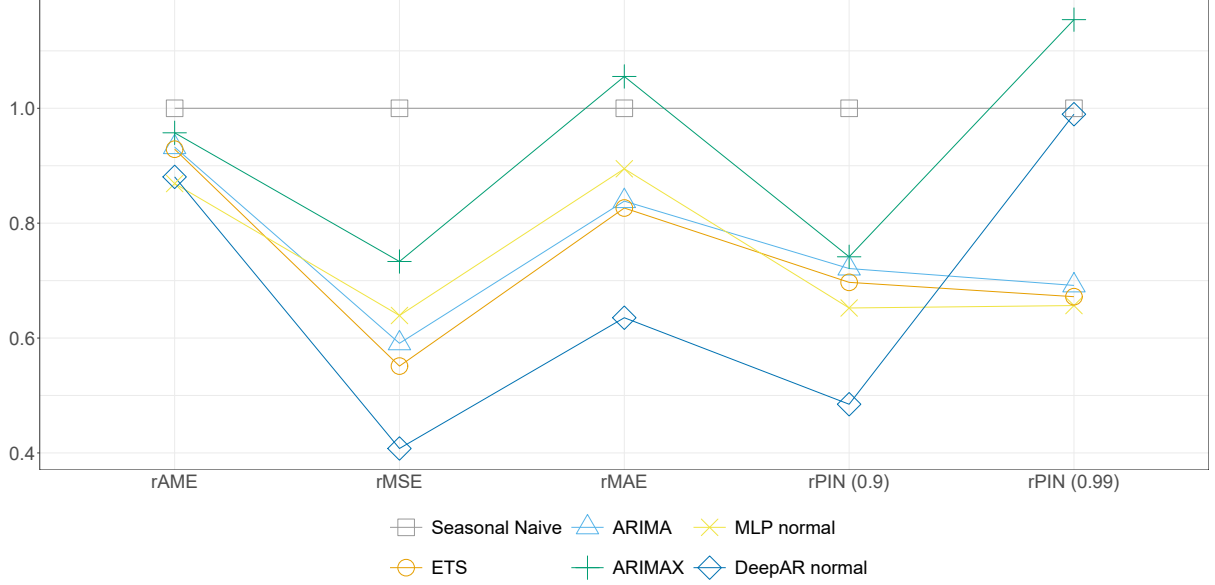


Figure 2: Accuracy per method and metric for withdrawals

an assumed normal distribution outperforms DeepAR with an assumed logistic distribution on lodgements, share denominations.

- ARIMAX performs worse than the naive forecast on lodgements, overall value, and substantially worse than the other methods on withdrawals. It performs similar to ETS and ARIMA on lodgements, share denominations and outperforms these on lodgements, share reissuable.

4.2 Inventory simulation

Forecast accuracy and inventory performance of forecast methods are not linearly related with each other (Kourentzes et al., 2020), and accordingly the evaluated accuracy metrics only provide a limited proxy for the evaluation of forecast methods. Therefore, we additionally benchmark the value of the different forecasts with an inventory simulation study. Thereby we predict in a rolling fashion along a test set and simulate the inventory decisions, given that forecast. We do so for each denomination and branch. This allows us to calculate the achieved service level, along with the overall inventory, and compare these across methods. We assume an order up to policy with backorders and continuous review. See Axsäter (2015) for an introduction.

In the considered policy, in each period the forecasted cumulative lead time demand is calculated. For each of the considered methods, we obtain N samples of the predictive distributions per period $\hat{y}_{t,s}$, and we calculate sample paths of the lead time cumulative distribution for lead time T by $\hat{L}_{s,t} = \sum_{t=1}^T \hat{y}_{t,s}$. To achieve the required service level γ , we need to plan on the respective quantile of this distribution $S_t = Q(\hat{L}_s, s = \{1, \dots, N\}, \gamma)$, with $Q(\cdot)$ being the quantile function (Axsäter, 2015, pages 77 and 81). As estimation

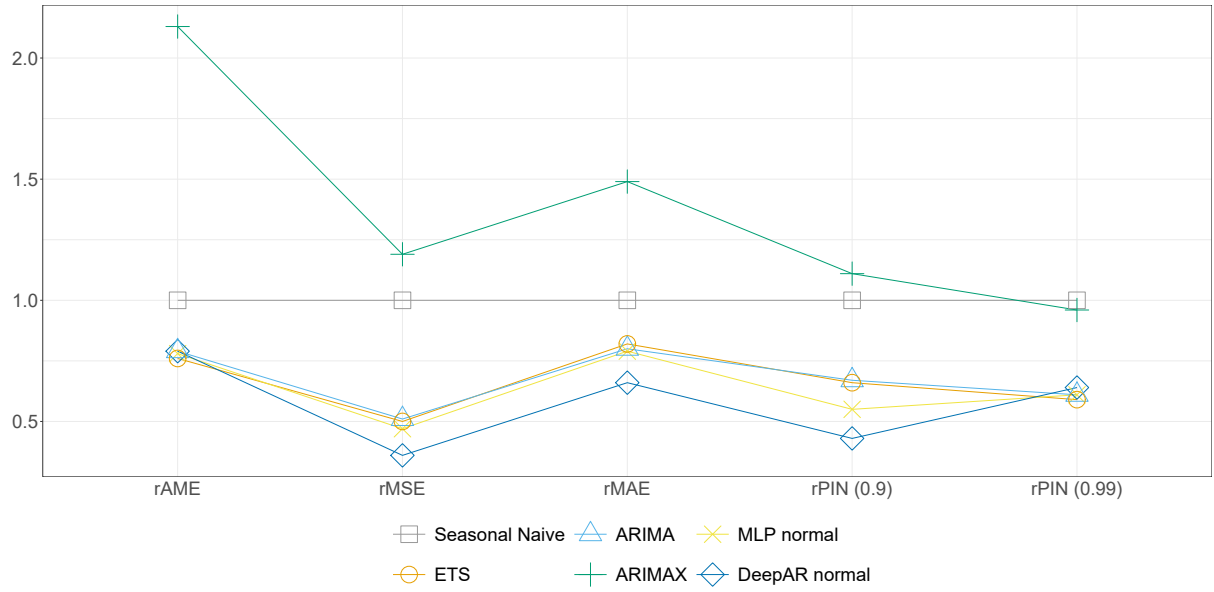


Figure 3: Accuracy per method and metric for lodgements, overall value

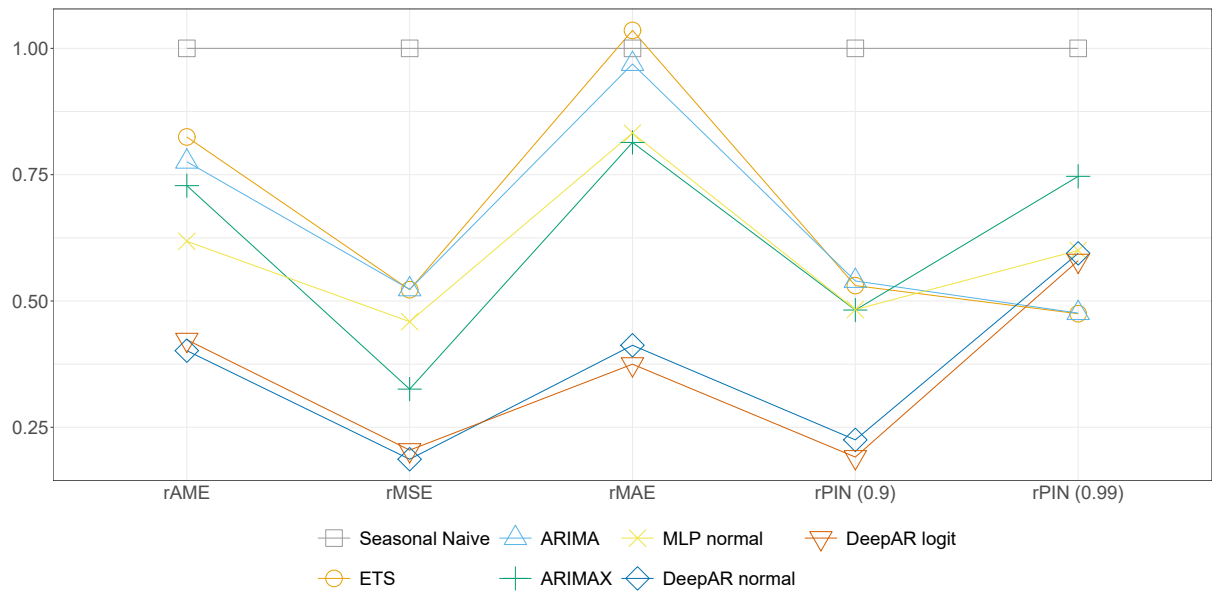


Figure 4: Accuracy per method and metric for lodgements, share reissuable

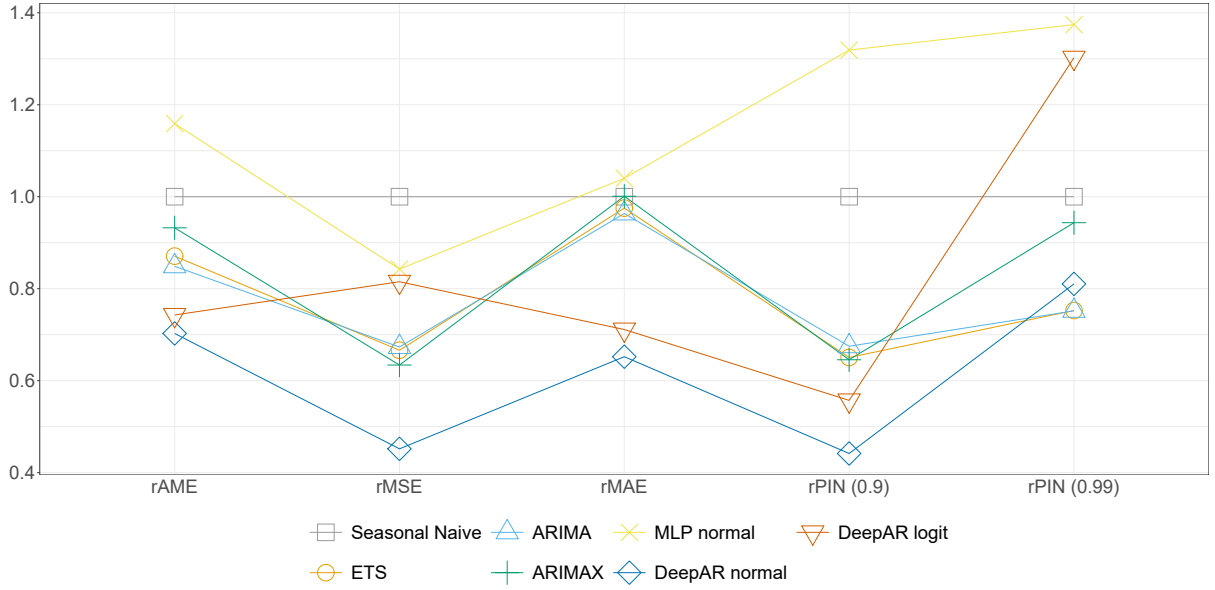


Figure 5: Accuracy per method and metric for lodgements, share denominations

of quantiles becomes inefficient on high probabilities (Taylor, 2021), we assume a conditionally normal distributed demand and estimate it by the empirical mean and standard deviation of the cumulative lead time demand. Then the quantiles can be calculated analytically.

We account further for the current stock on hand l_i in period i and the previous orders which arrive during the lead time $\sum_{t=i-T}^{i-1} o_t$. The ordered quantity in i , which will arrive in $i + T$ is then calculated as:

$$o_t = \max(S_t - l_t - \sum_{t=i-T}^{i-1} o_t, 0),$$

We simulate the placed order for all types of currently used notes, all six considered branches, and for each day from 30.07.2021 to 30.04.2022. To facilitate evaluation, we only consider cash orders and omit cash deposits. We evaluate the achieved service level and the on average held stock. The achieved service level is calculated by the share of out of stock periods, i.e., the periods where the available stock was smaller than the actual demand. Our aim is to match or exceed the target service level, while holding few average stock as this induces inventory holding cost. We can rank the different forecasting approaches accordingly. For each forecast method, we evaluate six target service levels (0.5, 0.7, 0.8, 0.9, 0.95, 0.99) and we consider three different lead times (7 days, 14 days, 21 days).

Figures 6, 7, and 8 summarize the findings of the inventory study. Each solid coloured line corresponds to one forecasting method and shows the relation between held average stock and the achieved service level. The farther left top a method is, the better is the respective inventory performance, as with the same held inventory a higher service level is reached. The black, dotted vertical lines show the deviation between achieved service

level and target service level. If they are below the respective line, the target service level is exceeded, if they are above, the method fails to reach the target service level.

Similar to the accuracy ranking, DeepAR outperforms the the other methods. Notably, ARIMAX ranks second across lead times at inventory performance, even though it ranked relatively bad with respect to forecast accuracy. Further we find:

- Only DeepAR does not reach the target service level, as it seems to underestimate the forecast uncertainty. This might be due to over-fitting, as the whole forecast distribution is learned on the training-set. Such, if DeepAR is used for the discussed use-case, some recalibration of the uncertainty estimate based on a hold-out set, or choosing a broader distribution might be helpful. In its standard form it is not suitable for our use-case, as central banks require reliably very high service levels. All other methods almost always exceed the required service level, apart from the 50 % service level.
- All elaborate methods improve substantially over the naive benchmark.
- With increasing lead time, the required inventory to satisfy the target service level increases substantially.

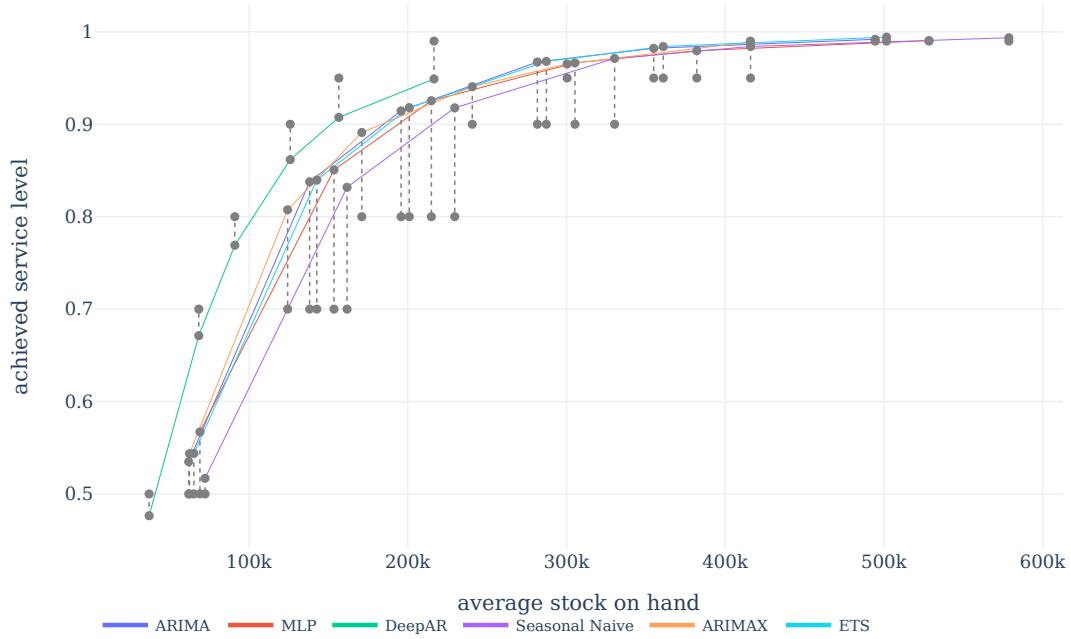


Figure 6: Inventory performance curves for lead time 7

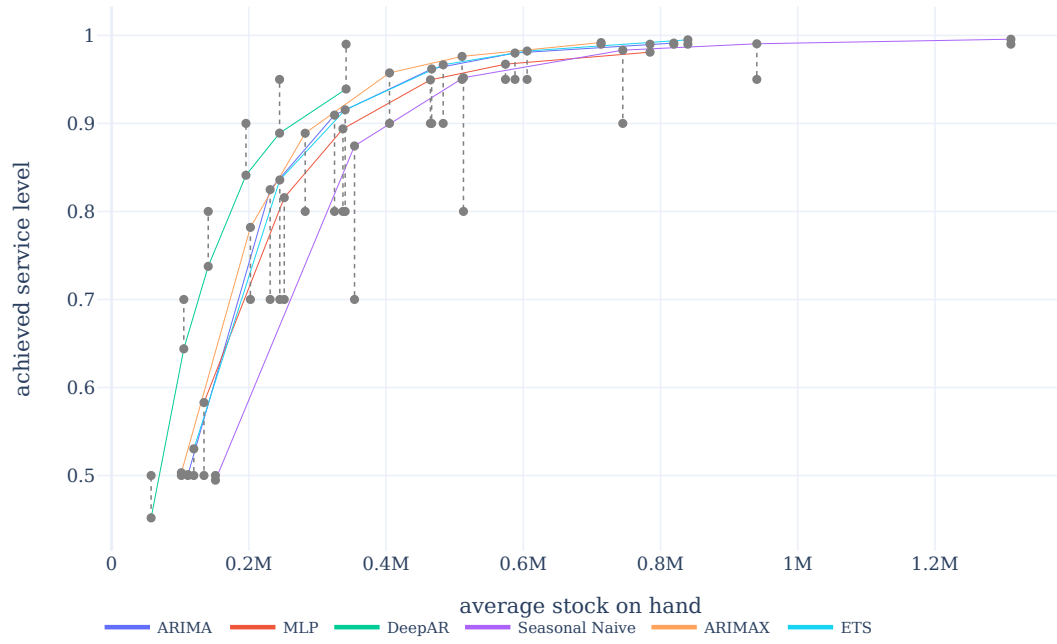


Figure 7: Inventory performance curves for lead time 14

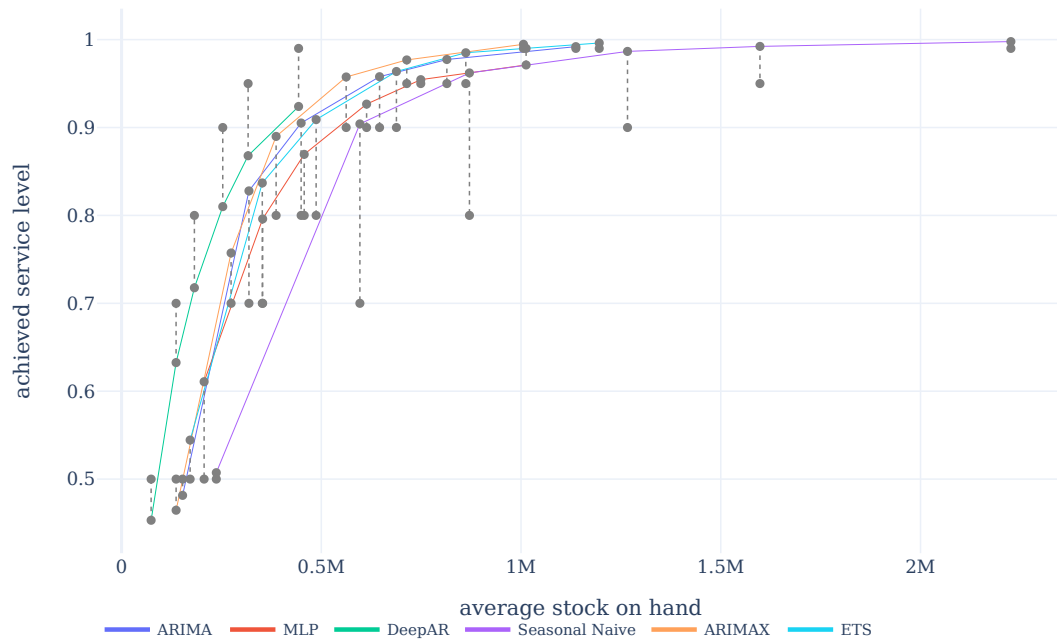


Figure 8: Inventory performance curves for lead time 21

5 Conclusion

We conducted a benchmark study on forecasting cash demand across all denominations in six regional branches of the Bundesbank using four daily data sets on withdrawals, lodgements and processed fit banknotes running from January 2017 to May 2022. We evaluated ARIMA, ARIMAX, ETS, MLP, and DeepAR against the seasonal naive benchmark in terms of forecast accuracy and inventory performance. We find that DeepAR outperforms the other methods substantially with regard to accuracy of the mean forecast, the 90 % quantile forecast, and out of sample bias. ARIMA and ETS perform similar across data sets, ARIMAX performs similar or worse than these in terms of accuracy on three out of four data sets. All benchmarked methods, apart from ARIMAX and the MLP outperform the seasonal naive benchmark across all data sets and metrics. The improvement in forecast accuracy translates also into better inventory performance, where DeepAR shows the best inventory performance, however systematically does not reach the target service level. In contrast to the accuracy study, ARIMAX ranks second best. All benchmarked methods thereby exceed the required service level, apart from DeepAR, which falls short of it, however at a substantially lower amount of held inventory. Accordingly we argue that data driven forecasting, using neural networks can benefit the planning processes of physical cash distribution by ensuring a higher service level at the same held inventory, or vice versa.

Future work should focus on robust estimation of very high cumulative lead time demand quantiles, as these are required to ensure the availability of cash even under demand or supply shocks. Further, we suggest to evaluate the different forecasting methods in combination with the inventory routing problem, that includes not only the inventory management in the branches but also the required transports, as costs induced by these are substantial due to the high security requirements with further restrictions like availability of accompanying regional police forces. To improve acceptance of the evaluated Machine Learning methods, we further suggest to investigate explainable versions of these.

We hope that this study can amalgamate features of longer term forecasting for banknote production within central banks and replenishment optimization problems for ATM within the industry and deliver a starting point for a more formalized and data driven forecast for physical cash distribution at the Deutsche Bundesbank and other central banks.

A Time series visualizations

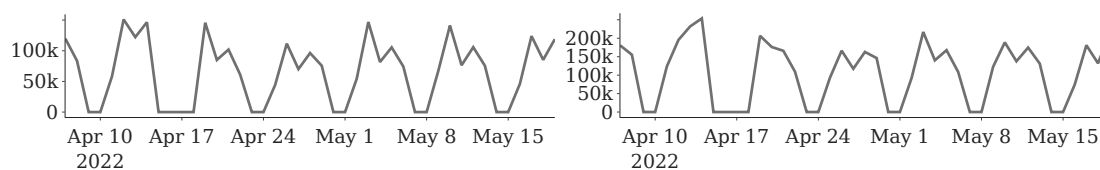


Figure 9: Time series for cash orders five (left) and ten (right) euro denominations in Villing-Schwenningen

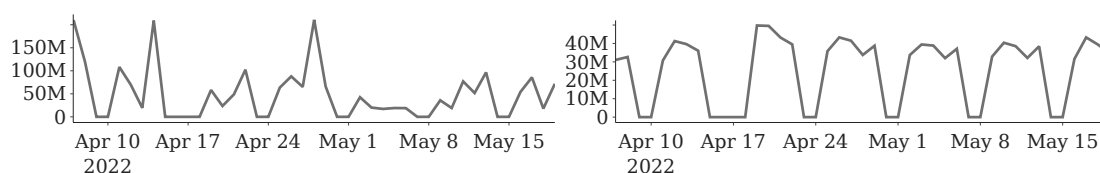


Figure 10: Time series for logdements, overall value in the branches Villing-Schwenningen (left) and Reutlingen (right)

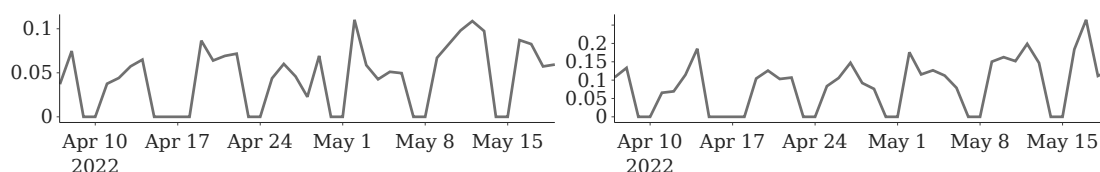


Figure 11: Time series for logdements, share per note, in the branches Villing-Schwenningen (left) and Reutlingen (right)

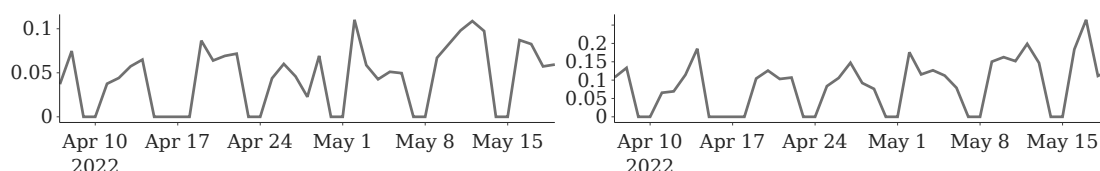


Figure 12: Time series for logdements, share per note, in the branches Villing-Schwenningen and Reutlingen

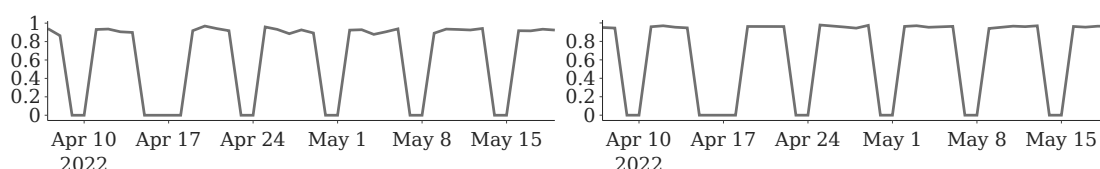


Figure 13: Time series for logdements, share reissuable, 5 Euro notes and 10 Euro notes in the branch Villing-Schwenningen

B Hyper parameter settings DeepAR and MLP

Table 5: Hyper parameters

parameter	MLP	DeepAR
no. of lags (context length)	2 * prediction length	2 * prediction length
hidden layers	[40, 40, 40]	[40, 40]
learning rate	1e-3	1e-3
maximum number of epochs	500	500
batches per epoch	100	100
batch size	32	32
weight decay	1e ⁻⁸	1e ⁻⁸
weight initialization	xavier	xavier
clip gradient	10.0	10.0
cell type	-	LSTM

C Tables with accuracy benchmark results

Table 6: Accuracy benchmark for withdrawals

Method	rAME	rMSE	rMAE	rPIN (0.9)	rPIN (0.99)
seasonal naive	1.00	1.00	1.00	1.00	1.00
MLP	0.87	0.64	0.89	0.65	0.66
DeepAR	0.88	0.41	0.64	0.48	0.99
ARIMA	0.93	0.59	0.84	0.72	0.69
ARIMAX	0.96	0.73	1.06	0.74	1.15
ETS	0.93	0.55	0.83	0.70	0.67

Table 7: Accuracy benchmark for lodgements, overall value

Method	rAME	rMSE	rMAE	rPIN (0.9)	rPIN (0.99)
seasonal naive	1	1	1	1	1
MLP	0.78	0.47	0.79	0.55	0.61
DeepAR	0.79	0.36	0.66	0.43	0.64
ARIMA	0.79	0.51	0.8	0.67	0.61
ARIMAX	2.13	1.19	1.49	1.11	0.96
ETS	0.76	0.5	0.82	0.66	0.59

Table 8: Accuracy benchmark for lodgements, share reissuable

Method	rAME	rMSE	rMAE	rPIN (0.9)	rPIN (0.99)
seasonal naive	1.00	1.00	1.00	1.00	1.00
MLP normal	0.62	0.46	0.83	0.48	0.60
DeepAR normal	0.40	0.19	0.41	0.23	0.59
DeepAR logit	0.42	0.20	0.37	0.19	0.58
ARIMA	0.78	0.52	0.97	0.54	0.48
ARIMAX	0.73	0.33	0.81	0.48	0.75
ETS	0.82	0.52	1.04	0.53	0.47

Table 9: Accuracy benchmark for lodgements, share denominations

Method	rAME	rMSE	rMAE	rPIN (0.9)	rPIN (0.99)
seasonal naive	1.00	1.00	1.00	1.00	1.00
MLP normal	1.16	0.84	1.04	1.32	1.37
DeepAR normal	0.70	0.45	0.65	0.44	0.81
DeepAR logit	0.74	0.82	0.71	0.56	1.30
ARIMA	0.85	0.67	0.96	0.67	0.75
ARIMAX	0.93	0.63	1.00	0.65	0.94
ETS	0.87	0.67	0.98	0.65	0.75

References

- Alexandrov, A., K. Benidis, M. Bohlke-Schneider, V. Flunkert, J. Gasthaus, T. Januschowski, D. C. Maddix, S. Rangapuram, D. Salinas, J. Schulz, L. Stella, A. C. Türkmen, and Y. Wang (2020). Gluonts: Probabilistic and neural time series modeling in python. *Journal of Machine Learning Research* 21(116), 1–6.
- Axsäter, S. (2015). *Inventory Control*, Volume 225. Cham: Springer International Publishing.
- Barker, J. (2020). Machine learning in m4: What makes a good unstructured model? *International Journal of Forecasting* 36(1), 150–155.
- Bartzsch, N., M. Brandi, R. de Pastor, L. Devigne, G. Maddaloni, D. P. Restrepo, and G. Sene (2023). Forecasting banknote circulation during the covid-19 pandemic using structural time series models. *Deutsche Bundesbank, Discussion Paper* (20).
- Ben Taieb, S., G. Bontempi, A. F. Atiya, and A. Sorjamaa (2012). A review and comparison of strategies for multi-step ahead time series forecasting based on the nn5 forecasting competition. *Expert Systems with Applications* 39(8), 7067–7083.
- Bundesbank (2022). Zahlungsverhalten in deutschland 2021.

- Chatfield, C. (2000). *Time-Series Forecasting*. Chapman and Hall/CRC.
- Crone, S. F. (2008). Nn5 competition.
- Fallahtafti, A., M. Aghaaminiha, S. Akbarghanadian, and G. R. Weckman (2022). Forecasting atm cash demand before and during the covid-19 pandemic using an extensive evaluation of statistical and machine learning models. *SN computer science* 3(2), 164.
- Gneiting, T. and A. E. Raftery (2007). Strictly proper scoring rules, prediction, and estimation. *Journal of the American Statistical Association* 102(477), 359–378.
- Godaheewa, R., C. Bergmeir, G. I. Webb, R. J. Hyndman, and P. Montero-Manso (2021). Monash time series forecasting archive. In *Neural Information Processing Systems Track on Datasets and Benchmarks*.
- Hewamalage, H., C. Bergmeir, and K. Bandara (2021). Recurrent neural networks for time series forecasting: Current status and future directions. *International Journal of Forecasting* 37(1), 388–427.
- Hinge, D. (2022). The signal and the noise: cash forecasting in uncertain times. *Central banking* 32(4), 95–100.
- Hornik, K., M. Stinchcombe, and H. White (1989). Multilayer feedforward networks are universal approximators. *Neural Networks* 2(5), 359–366.
- Hyndman, R. J. (2008). *Forecasting with exponential smoothing: The state space approach*. Springer series in statistics. Berlin and Heidelberg: Springer.
- Hyndman, R. J. and G. Athanasopoulos (2018). *Forecasting: Principles and practice* (2nd edition ed.). Lexington, Ky.: Otexts online open-access textbook.
- Hyndman, R. J. and Y. Khandakar (2008). Automatic time series forecasting: The forecast package for r. *Journal of Statistical Software* 27(3).
- Januschowski, T., J. Gasthaus, Y. Wang, D. Salinas, V. Flunkert, M. Bohlke-Schneider, and L. Callot (2020). Criteria for classifying forecasting methods. *International Journal of Forecasting* 36(1), 167–177.
- Januschowski, T., Y. Wang, K. Torkkola, T. Erkkilä, H. Hasson, and J. Gasthaus (2022). Forecasting with trees. *International Journal of Forecasting* 38(4), 1473–1481.
- Kolassa, S. (2020). Why the “best” point forecast depends on the error or accuracy measure. *International Journal of Forecasting* 36(1), 208–211.
- Kourentzes, N., J. R. Trapero, and D. K. Barrow (2020). Optimising forecasting models for inventory planning. *International Journal of production economics* 225, 107597.
- Kunz, M., S. Birr, M. Raslan, L. Ma, and T. Januschowski (2023). Deep learning based forecasting: A case study from the online fashion industry. In M. Hamoudia, S. G. Makridakis, and E. Spiliotis (Eds.), *Forecasting with Artificial Intelligence*, Palgrave Advances in the Economics of Innovation and Technology, pp. 279–311. Cham, Switzerland: Palgrave Macmillan.

- Makridakis, S. and M. Hibon (2000). The m3-competition: results, conclusions and implications. *International Journal of Forecasting* 16(4), 451–476.
- Makridakis, S., E. Spiliotis, and V. Assimakopoulos (2020). The m5 accuracy competition: Results, findings and conclusions. *International Journal of Forecasting*, 1346–1364.
- Makridakis, S., E. Spiliotis, V. Assimakopoulos, Z. Chen, A. Gaba, I. Tsetlin, and R. L. Winkler (2020). The m5 uncertainty competition: Results, findings and conclusions. *International Journal of Forecasting*, 1365–1385.
- Miller, C. (2017). Addressing the limitations of forecasting banknote demand. In Deutsche Bundesbank (Ed.), *International Cash Conference 2017*.
- Montero-Manso, P. and R. J. Hyndman (2021). Principles and algorithms for forecasting groups of time series: Locality and globality. *International Journal of Forecasting* 37(4), 1632–1653.
- O’Hara-Wild, M., R. J. Hyndman, and E. Wang (2021). fable.
- Petropoulos, F., S. Makridakis, V. Assimakopoulos, and K. Nikolopoulos (2014). ‘horses for courses’ in demand forecasting. *European Journal of Operational Research* 237(1), 152–163.
- Riabykh, A., I. Suleimanov, D. Surzhko, M. Konovalikhin, and V. Ryazanov (2022). Atm cash flow prediction using local and global model approaches in cash management optimization. *Pattern Recognition and Image Analysis* 32(4), 803–820.
- Salinas, D., V. Flunkert, J. Gasthaus, and T. Januschowski (2020). Deepar: Probabilistic forecasting with autoregressive recurrent networks. *International Journal of Forecasting* 36(3), 1181–1191.
- Schäfer, A. M. and H. G. Zimmermann (2006). Recurrent neural networks are universal approximators. In D. Hutchison, T. Kanade, J. Kittler, J. M. Kleinberg, F. Mattern, J. C. Mitchell, M. Naor, O. Nierstrasz, C. Pandu Rangan, B. Steffen, M. Sudan, D. Terzopoulos, D. Tygar, M. Y. Vardi, G. Weikum, S. D. Kollias, A. Stafylopatis, W. Duch, and E. Oja (Eds.), *Artificial Neural Networks – ICANN 2006*, Volume 4131 of *Lecture Notes in Computer Science*, pp. 632–640. Berlin, Heidelberg: Springer Berlin Heidelberg.
- Smyl, S. (2020). A hybrid method of exponential smoothing and recurrent neural networks for time series forecasting. *International Journal of Forecasting* 36(1), 75–85.
- Svetunkov, I. (2023). *Forecasting and Analytics with the Augmented Dynamic Adaptive Model (ADAM)*. Chapman and Hall/CRC.
- Tashman, L. J. (2000). Out-of-sample tests of forecasting accuracy: an analysis and review. *International Journal of Forecasting* 16(4), 437–450.
- Taylor, J. W. (2021). Evaluating quantile-bounded and expectile-bounded interval forecasts. *International Journal of Forecasting* 37(2), 800–811.

Venkatesh, K., V. Ravi, A. Prinzie, and D. den van Poel (2014). Cash demand forecasting in atms by clustering and neural networks. *European Journal of Operational Research* 232(2), 383–392.