

**FOR
INFORMATION**

FO/DIS/22/50

March 29, 2022

To: Members of the Executive Board

From: The Secretary

Subject: **Jordan—Technical Assistance Report—Forecasting Framework for Currency
in Circulation**

Board Action:	Executive Directors' information
Publication:	Yes, after Tuesday, April 5, 2022
Questions:	Mr. Lafarguette, MCM (ext. 37545)



TECHNICAL ASSISTANCE REPORT

JORDAN

Forecasting Framework for Currency in Circulation

MARCH 2022

Prepared By

*Romain Lafarguette (Mission Chief), Arthur Charpentier (External Expert),
Nikolaos Kourentzes (External Expert), and Kei Moriya (ITD)*

Authoring Departments:

**Monetary and Capital Markets Department
Information Technology Department**

Contents	Page
Glossary	3
Preface.....	4
Executive Summary	5
I. Introduction	8
II. Forecasting Framework.....	10
III. Data Pre-Processing	12
IV. Modeling Seasonality and Special Events	14
V. First-Tier: Individual Forecasting Models	17
VI. Second-Tier: Model Evaluation, Selection, and Combination	20
VII. Results, Recommendations, and Conclusion	21
 Figures	
1. Currency Issued, with Imputed Values Highlighted.....	13
2. Currency in Circulation, with Imputed Values Highlighted.....	13
3. Seasonal Plots for the Three Seasonalities Present in the Time Series.....	16
4. Example of Full and Sparse Seasonal Encoding Using Binary and Trigonometric Indicators	17
5. Nemenyi Test Results for the Currency Issued Forecasts	24
6. Nemenyi Test Results for the Currency in Circulation Forecasts.....	25
7. A Forecast with its 80 percent, 90 percent, and 95 percent Prediction Intervals.....	26
 Tables	
1. Key Recommendations	7
2. Example of Two Lags and Two Leads From a Binary Indicator Variable.....	14
3. Summary Results for the Forecasting of the Currency Issued.....	22
4. Summary Results for the Forecasting of the Currency in Circulation.....	23
5. Summary Results for the Forecasting of Currency Issued Using Temporal Hierarchies	25
 Annexes	
I. Software Documentation	28
II. User Guide for R Code of Currency in Circulation	34

GLOSSARY

AIC	Akaike's Information Criterion
ARIMA	Auto Regressive Integrated Moving Average
ARMA	Auto Regressive Moving Average
CBJ	Central Bank of Jordan
CiC	Currency in Circulation
CrI	Currency Issued
ELM	Extreme Learning Machine
ETS	Exponential Smoothing Model (E: Error Term, T: Trend, S: Seasonal)
IMF	International Monetary Fund
KPSS	Kwiatkowski–Phillips–Schmidt–Shin Test of Stationarity
MCM	Monetary and Capital Markets Department
MLE	Maximum Likelihood Estimator
MoU	Memorandum of Understanding
NN	Neural Network
RMSE	Root Mean Squared Error
SARIMA	Seasonal Auto Regressive Integrated Moving Average
SLA	Staff Level Agreement
TBATS	Trigonometric Seasonal, Box-Cox Transformation, ARMA Residuals, Trend and Seasonality
THieF	Temporal Hierarchy Forecasting

PREFACE

At the request of the Central Bank of Jordan (CBJ), a Monetary and Capital Markets (MCM) Department remote mission took place from April to June 2021 to assist the authorities in forecasting currency in circulation for Jordan.

The mission met with Nedal Azzam (Executive Manager, Research Department); Mohammad A. Khreisat, Rajeh A. Alkhdour and Rami A. Alhadid (all Executive Manager Assistants, Research Department); Mohammed H. Shinnar (Executive Manager Assistant, Open Market Operations and Public Debt Department); Mohammed N. Arar (Executive Manager Assistant, Investments and Foreign Operations Department); Ibrahim Q. Naser (Head of Division, Monetary Policy Division, Research Department); Tariq A. Almuhaissen, Mahmoud W. Qasem, Farah M. Khamash, Muna M. Alkurdi, Osama Alsayeh (Senior Economists and Senior Specialists at the Research Department and the Open Market Operations and Public Debt Department respectively). The mission wishes to thank the colleagues of the Central Bank of Jordan for their cooperation and productive discussions.

EXECUTIVE SUMMARY

The currency in circulation forecasting model presently used by the Central Bank of Jordan is aligned with international practices and provides a solid basis for liquidity management. The central bank uses an Auto Regressive Integrated Moving Average (ARIMA) model with many indicator variables to model binary seasonality and to capture special events. The ARIMA model is fitted on daily currency in circulation data using a standard maximum likelihood estimator. This ARIMA approach is aligned with the models traditionally used by central banks in emerging and middle-income countries.

However, there is substantial room to improve the forecasting approach of the Central Bank of Jordan to provide more accurate and robust forecasts of currency in circulation. Modeling seasonality patterns with many indicator variables implies estimating many parameters, hence amplifying parametric noise and reducing forecasting accuracy. This approach also increases the risk of overfitting and is very sensitive to structural breaks.

The IMF team has developed a full-fledged forecasting framework, relying on the most recent advances in the statistical literature, to model both currency in circulation and currency issued. Over the last 20 years, the statistical literature has made large progress over the standard ARIMA model. It has proposed a wide range of different modeling approaches to model seasonality more accurately. Drawing on these new methodologies, the International Monetary Fund (IMF) team provided a complete forecasting framework. This framework tests different families of forecasting models, with an integrated selection of the best performer based on information criteria and out-of-sample performance. It also allows to automatically reparametrize the models to account for structural breaks and new data developments.

The mission's objective is to provide a methodology that will automatically detect the best modeling strategy and reparametrize the models in real-time on the currency in circulation (CiC) time series. This unified forecasting approach can handle different time series—for instance, currency in circulation, currency issued, and individual banknotes denomination—at different frequencies. The framework also adapts to the idiosyncrasies of each time series while remaining resilient to changes in the conditions that the central bank faces.

The forecasting framework is organized in a two-tier system. This system encompasses in its first-tier an estimation of six different families of forecasting individual models. As second-tier, both dynamic model selection and model aggregation are employed to maximize accuracy. The software infrastructure automatically carries out the estimation of the first-tier individual models. The main statistics of interest—projections, parameters, in-sample fit, out-of-sample performance metrics, etc.—are generated immediately. The second-tier applies both model aggregation and model selection techniques to improve the individual models' performances while also delivering on the fly the main statistics of interest.

The two-tier forecasting framework allows the authorities to choose the best approach to arbitrage between forecasting accuracy and modeling complexity. While the second-tier forecasts outperform the results of the first-tier individual models, they come at the cost of extra complexity. From an operational perspective, the gain in accuracy might not be sufficient for the authorities to use the most accurate methods. The central bank might prefer to use the best individual model performers, as single models are more parsimonious and faster to estimate.

The IMF team found that dynamic model selection is the best overall performer, while an ARIMA with trigonometric seasonality delivers the best performance/complexity compromise. Based on out-of-sample performance metrics, the IMF team has assessed the different models on both tiers at different horizons. Overall, the ARIMA with trigonometric seasonality performs the best among the first-tier individual models, and the gain of using the dynamic selection of models is significant but not substantial. Therefore, the authorities might prefer using the ARIMA with trigonometric seasonality specification. This model is parsimonious, relies on a straightforward specification and , performs quite well. In particular, it overperforms the ARIMA with binary seasonality substantially. Therefore, it represents an improvement over the current ARIMA with binary seasonality used by the central bank.

The mission also provided alternative modeling venues, such as the aggregation of banknotes denomination and temporal aggregation, and hence explored a wide range of modeling strategies. The mission explored alternative frontier strategies for further improving the accuracy of the currency in circulation forecasts. These frontier strategies include: (i) aggregating forecasts at different frequencies (daily, weekly, monthly, quarterly) to capture short-term and long-term dynamics differently; and (ii) forecasting banknotes by denomination individually and aggregating back to infer the whole currency in circulation time series. While conceptually appealing, the IMF team has found that these approaches do not provide significant forecasting gains and come at the cost of high complexity. Hence, the mission does not recommend using them.

The mission provided a full-fledged software infrastructure to the authorities using a state-of-the-art, open-source programming language. The two-tier forecasting framework has been entirely programmed under R, the leading free and open-source statistical software used in the forecasting industry. The IMF experts have designed a user-friendly infrastructure to help the authorities produce their forecasts smoothly and quickly. The programming infrastructure is entirely automatized and produces charts and tables on the fly for easy inspection and interpretation. The code is fully documented, using best programming practices to limit errors and bugs. Finally, the IMF experts have also taught the authorities how to use the software, ensuring that the central bank can readily use the forecasts for operational purposes.

Table 1. Jordan: Key Recommendations

Recommendations and Authority Responsible for Implementation	Priority	Timeframe¹
Use a trigonometric ARIMA instead of a binary ARIMA (CBJ)	High	Near-term
Implement the two-tier forecasting framework and obtain the set of forecasts, leveraging on the full range of models (CBJ)	Medium	Medium-term
Produce monitoring forecasting reports and analyze the performance of the forecasts at a regular interval (CBJ)	Medium	Medium-term
Publish the methodology to strengthen the transparency and the credibility of the central bank (CBJ)	Medium	Medium-term

¹ Near term: < 12 months; Medium term: 12 to 24 months.

I. INTRODUCTION

- 1. The Central Bank of Jordan has operated a fixed exchange rate arrangement against the US dollar since 1995 and manages domestic liquidity through a corridor system.** The operational framework of the CBJ relies on a comprehensive set of monetary instruments, including various types of open market operations, in the context of an interest rate corridor, supplemented with long-term (up to one year) collateralized lending and reserve requirements. While a rediscount facility is part of the operational framework, it has not been utilized recently. The CBJ publishes its liquidity forecast daily, but this forecast doesn't include a formal model.
- 2. Accurate forecasts of the autonomous factors are crucial to calibrate the CBJ operations adequately.** The central bank aims to maintain a certain level of structural liquidity surplus to guarantee a smooth transmission of monetary policy while preserving financial stability. An excessive level of structural liquidity surplus would dampen monetary policy transmission, while an insufficient level would spur interbank rate volatility and endanger financial stability. Therefore, it is critical for the central bank to forecast the autonomous factors accurately to be able to anticipate and maintain an adequate level over time.
- 3. At the request of the authorities, the mission has focused on forecasting currency in circulation (CiC).** There are three types of autonomous factors: the currency in circulation, the state account at the central bank, and the net foreign assets. Central banks usually forecast only the currency in circulation, as it is a straightforward item to forecast. The state account at the central bank is often forecasted by the Treasury, based on its expected income and expenses flows and debt issuance. However, in the case of Jordan, the Treasury doesn't provide forecasts of its expected cash flow movements to the CBJ. Finally, net foreign assets can be very volatile in fixed exchange rate arrangements and are often not forecasted by the central bank, beyond the two-day settlement date. Therefore, the CBJ has asked the IMF technical assistance team to focus on the CIC forecasts, as a step towards liquidity forecast. However, to improve further liquidity management, the authorities should forecast all the autonomous factors, including the government position and central bank net foreign assets. The forecast of the government position at the central bank would greatly benefit from an institutional arrangement between the central bank and the Treasury, formalized, for instance, by a Staff Level Agreement (SLA) or a Memorandum of Understanding (MoU). The authorities could also request a follow-up mission with the IMF to deploy a full-fledged forecasting framework for the autonomous factors.
- 4. The current liquidity forecasting framework of the central bank is solid and in line with international standards.** The central bank uses an ARIMA model with many indicator variables to model binary seasonality and to capture special events. The ARIMA model is fitted on daily currency in circulation data using a standard maximum likelihood

estimator. This ARIMA approach is aligned with the models traditionally used by central banks in emerging and middle-income countries.

5. However, there is room to substantially improve the accuracy and the robustness of the estimator currently used by the CBJ. While the ARIMA with binary seasonality is used by many central banks worldwide, it has many drawbacks. Modeling seasonality patterns with many indicator variables implies estimating many parameters, hence amplifying parametric noise and reducing forecasting accuracy. This approach also increases the risk of overfitting and is very sensitive to structural breaks. The results suggest that the performance of ARIMA with binary seasonality can be improved substantially by using more appropriate models of seasonality.

6. The recent forecasting literature has developed more appropriate models to deal with time series seasonality in a parsimonious yet accurate way. Over the last twenty years, the forecasting literature has developed more robust, parsimonious, and accurate methods to model seasonal patterns than using indicator variables. These approaches include methods derived from the Fourier transform (trigonometric seasonality), advanced smoothing techniques, machine learning and neural network, etc.

7. The mission has tested many different models to select the most appropriate one for the CBJ. Rather than relying on a single model, the IMF team has tested many different forecasting models, following the most updated statistical literature. The models are then benchmarked using out-of-sample performance metrics, which is considered as best practice in forecast evaluation. Based on this comprehensive statistical approach, the mission provides the CBJ with two options: (i) a simple, yet accurate ARIMA model with trigonometric seasonality; or (ii) a more advanced, most accurate model selection approach which automatically chooses the most appropriate model across time and re-estimates the parameters based on recent data developments. Depending on the level of complexity the CBJ wants to implement, the central bank can either: (i) use a single robust model, which uses the same class of model as they currently use but improves the modeling of seasonality; or (ii) decide to implement a more complex statistical architecture which consistently outperforms other models. The IMF team has provided the authorities with all the necessary codes and documentation to implement these different approaches.

8. The rest of the technical report is organized as follows. Section II presents the forecasting framework designed by the IMF teams. Section III discusses the preliminary data pre-processing steps necessary to the estimation of the model. Section IV discusses in detail the modeling of seasonality and special events for currency in circulation. Section V presents the first-tier of the forecasting framework (i.e., the single models). Section VI provides an in-depth view of the second-tier modeling framework, the advanced aggregation and selection models. Section VII discusses the results and formulates the recommendations and concludes. Annex I provides the documentation of the software codes shared by the IMF

team with the authorities. Annex II is a short user guide on how to use the codes for daily operational work.

II. FORECASTING FRAMEWORK

9. The mission proposes to improve the current CBJ liquidity forecasting framework by implementing a flexible, multi-model forecasting framework. The two-tier modeling framework offers the possibility to fit both simple and complex models and to test their performance. The objective is to rely on more complex models only if there is sufficient evidence of benefits to forecasting. More complex models are typically better at fitting to training data due to their increased flexibility in parametric estimation. However, this extra flexibility can lead to overfitting and substantially lower quality of forecasts. Another side effect of the increased complexity/number of parameters is a potentially increased volatility of the forecasts over time, over different sample origins, as new data become available. Therefore, the authorities should choose more complex models only if they substantially outperform simpler benchmark models.

10. The first-tier modeling framework tests a series of individual models, considered as standard in the forecasting literature. The mission first tests single “standard” models on the data and assesses their performances. These models include: (i) basic models, such as the random walk (the “naïve model”), exponential smoothing, and ARIMA with binary seasonality; and (ii) an augmented version of these models where seasonality is modeled trigonometrically and holidays/special events are added. The team also tested neural networks and TBATS models, but these models have been excluded from the analysis given their poor performances. The software infrastructure automatically carries out the estimation of the first-tier individual models. The main statistics of interest (i.e., projections, parameters, in-sample fit, out-of-sample performance metrics, etc.) are provided on the fly.

11. The second-tier modeling framework uses dynamic selection to optimally select the best model and parametrization based on recent performances. The second-tier builds on the first-tier to improve the forecasting accuracy of the single models, either by combining them or selecting them. A combination of models averages the forecasts of different models (potentially with different weights) to improve the overall accuracy; the idea is that some models might be performing better than others over certain periods, and aggregating them reduces the modeling risk (i.e., trusting a single model for the forecasts). On the other hand, model selection consists of selecting the most appropriate model based on its recent performance. For instance, if last week a given model outperformed all the alternatives, it will be chosen for this week, etc. The model selection method quickly adjusts to structural breaks by shifting towards the most appropriate model when necessary.

12. The framework uses a battery of forecasting performance metrics and information criteria to select the most appropriate model. The team has used both in-sample information criteria and out-of-sample performance metrics to benchmark the models

and choose the most appropriate one. Using out-of-sample metrics is crucial to reduce the risk of overfitting and provide an accurate picture of the model's actual performance. In a nutshell, out-of-sample metrics consist of splitting historical data into two groups: a training group from 2010 to 2018 and a testing group from 2018–2020. The model is only fitted on the training group—2010 to 2018—and then forecasts 2018–2020 for which the realized values are known to the researcher. Then, the central bank can readily assess the model's performance between the forecasts in 2018–2020 (based on data before 2018) and the observed realized values.

13. The team also investigated alternative forecasting frameworks without performance gains. The IMF mission has explored further the data patterns in currency in circulation by exploiting cross-sectional aggregation across banknotes denomination and temporal aggregation of different frequencies. In the case of banknotes denomination, the idea is to model the banknotes patterns differently. For instance, large denominations might be used during special events, while small denominations are used for daily, basic transactions such as buying groceries. However, the gain to use highly granular data is counterbalanced by the volatility of each sub-series. In the case of Jordan, the granular approach provides worse quality forecasts than forecasting the total currency in circulation. Likewise, decomposing by frequency (i.e., yearly, quarterly, monthly, etc.) is expected to capture short-term and long-term dynamics differently. However, the increase in granularity comes at the cost of increased volatility and does not outperform forecasts on the total CiC time series in the case of Jordan.

14. The results for Jordan suggest that ARIMA with trigonometric seasonality is the best performing individual model. The results of the first-tier suggest that the ARIMA with trigonometric seasonality is the best performer among single models. This model improves over the ARIMA with binary seasonality used by the CBJ by modeling seasonality through sinusoidal functions rather than by many indicator variables. Using a trigonometric approach is a parsimonious yet accurate way to capture complex seasonality patterns while reducing parametric noise. This model represents an incremental improvement over the current CBJ approach rather than a radical change.

15. The dynamic model selection is the most accurate approach overall, as it embeds automatic re-parametrization to adjust to structural breaks and new developments. Dynamic model selection is the best-performing approach over all alternatives, both in the first and second-tier modeling. The possibility to change both the type of models and the parameters at each period maximizes the accuracy. It makes sure that the forecasting framework reacts fast to structural breaks and changes in seasonal patterns. It performs better than the aggregation approach, which averages both “good” and “bad” models at every point in time, sacrificing performance for stability. Dynamic model selection is preferable as it always picks the best performer at each point in time, rather than aggregating many models.

16. However, the ARIMA model with trigonometric seasonality offers the best compromise between accuracy and parsimony and is therefore more appealing to the CBJ. The gain in forecasting accuracy from the dynamic model selection is relatively small compared to the ARIMA with trigonometric seasonality. Given that the ARIMA with trigonometric seasonality is a relatively simple, parsimonious approach, and well-aligned with the authorities' current framework, it might be preferred for daily operations. That being said, should the CBJ wish to use the best approach available, then the practitioner should pick the dynamic model selection. The IMF team provided the open-source software infrastructure to the authorities to replicate the entire forecasting framework and results.

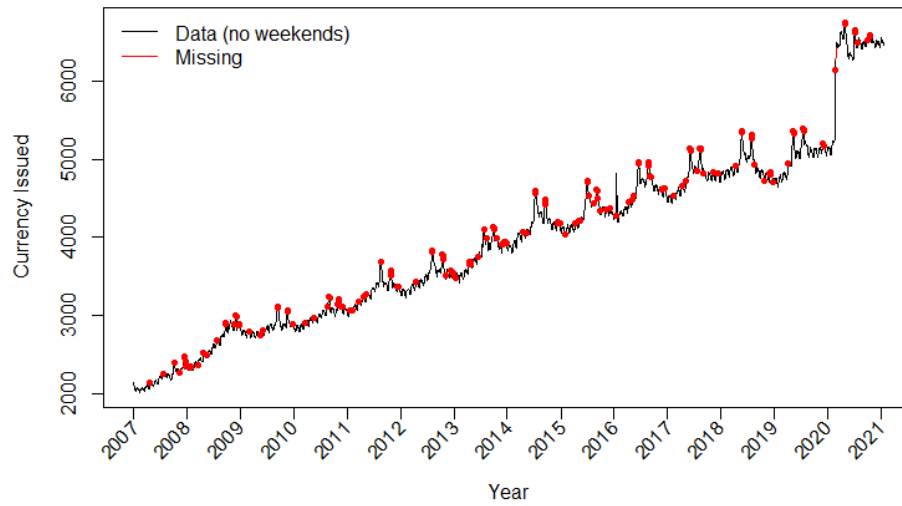
III. DATA PRE-PROCESSING

17. Data pre-treatment is a crucial element to increase model performance. The CBJ has provided data spanning from January 6, 2007, to February 15, 2021. The dataset contains daily records of Currency Issued (CrI) and Currency in Circulation (CiC). There are observations for the complete span of data for CrI, while the CiC observations started on November 1, 2018. The authorities also provided a calendar of special events across the years of the sample.

18. The treatment of missing values impacts matters to model seasonality properly. It is important to ensure that the time series is continuous in terms of dates, with no omissions. Continuity is essential for the modeling of any potential seasonality in the time series. With omitted dates, the periodicity between observations is broken, making the estimation of the seasonal terms very difficult. Therefore, when encountering missing values, the IMF team imputed them. There are multiple ways to impute missing values. The mission relies on linearly interpolating from neighboring values. Although this is not necessarily the best approach, the scarcity in missing values suggests that the method's robustness and simplicity counterbalance any minor losses in accurately imputing the missing values. Another alternative would be to provide indicator variables for each of the missing values to the models. Mathematically this is equivalent to excluding the observations.

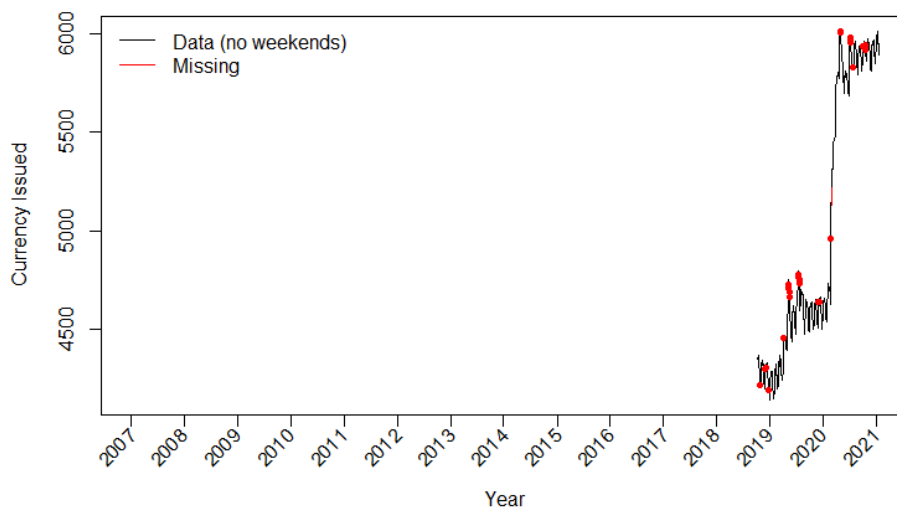
19. The model is fitted on five day weeks to tackle the problem of weekends. Therefore, the model excludes all non-working days (Fridays and Saturdays) from the time series and further inputs. This approach is not detrimental to the estimation of the seasonal patterns, as the periodicity is retained. Figure 1 provides an example of the imputations in the Currency Issued time series. Figure 2 provides the same view for the Currency in Circulation.

Figure 1. Currency Issued, with Imputed Values Highlighted



Source: IMF staff

Figure 2. Currency in Circulation, with Imputed Values Highlighted



Source: IMF staff

For both CiC and CrI, there is a substantial shift in the time series, which appears to be in response to the COVID-19 pandemic. This shift is modeled using an indicator variable, where all periods prior will be set equal to zero and afterwards equal to one. Furthermore, there is strong evidence of non-stationarity for both time series.

20. The team used indicator variables to capture holidays with up to five days lead and lag effects. Binary indicator variables are capturing holidays. The indicator variable has the value of 1 when the specific special event occurs across the years and 0 otherwise. For each indicator variable, the team constructed five lagged and five leading versions by shifting the indicator by the appropriate number of periods. Table 1 provides an example of this operation. These indicators are constructed for future periods as well. A different class of indicator variables marks any long-term or policy effects. These variables have the value of 1 after the policy/effect starts and 0 before or once it has ended. There are no leads and lags for long-term and policy effects but could be constructed if needed. These policy effects model level shifts in the time series and can be identified by exploration or judgmentally by relying on contextual information.

Table 2. Jordan: Example of Two Lags and Two Leads From a Binary Indicator Variable

Period	Indicator	Lag 1	Lag 2	Lead 1	Lead 2
1	0	0	0	0	0
2	0	0	0	0	1
3	1	0	0	1	0
4	0	1	0	0	0
5	0	0	1	0	0
6	0	0	0	0	0

Source: IMF staff

IV. MODELING SEASONALITY AND SPECIAL EVENTS

21. The IMF team incorporates the multiple seasonal cycles into the estimation (day in the week, day in the month, and day in the year). Multiple seasonal cycles substantially complicate the creation of forecasts, as many models typically incorporate a single seasonal periodicity. Three elements are of interest in modeling multiple seasonalities. First, the practitioner should prefer the length of the seasonal cycles, second their encoding, and third the encoding efficiency, as parsimonious models. To deal with the length of the seasonal cycles, the team counts how many days are in each periodicity. For example, there are five days in the week (weekends are removed). However, the day in the month seasonality is more challenging, as months have different days. To overcome this difficulty, the seasonal periodicity is locked to a quarter, as a quarter contains a fixed number of weeks, and by extension, days. Figure 3 provides seasonal plots of the Currency Issued time series. The top

three plots visualize the information as line plots, while the bottom three plots offer the same information in boxplots, which are more akin to the view that models will have. Any substantial differences in the location of the boxplots suggest that the estimation of the seasonal variability is more straightforward. The plots for the CiC series are similar.

22. Each seasonal cycle is encoded using trigonometric indicator variables.² Given the length of a season of s periods, the team constructed $s/2$ pairs of trigonometric variables, with $i = 1, \dots, s/2$:

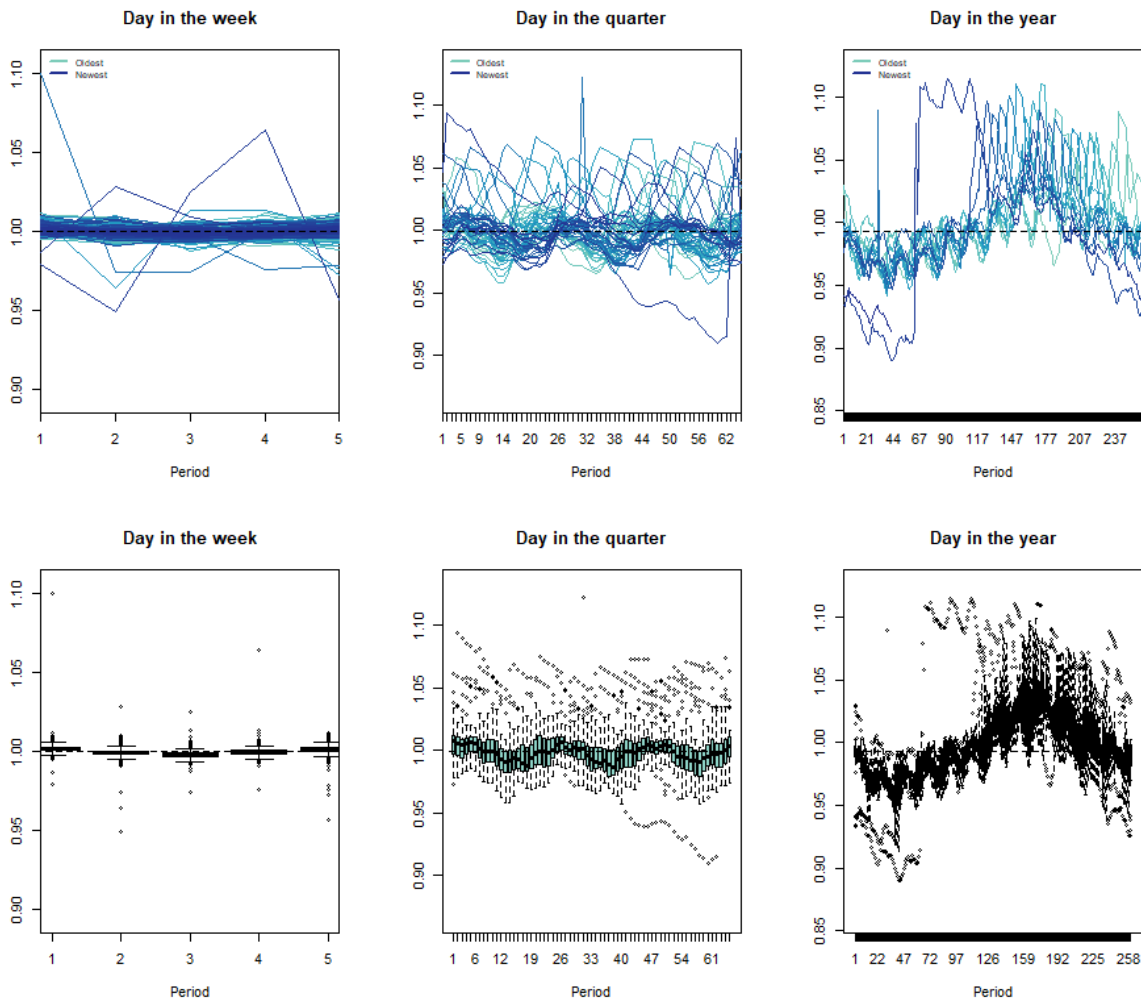
$$d_i = \cos\left(\frac{2i\pi t}{s}\right),$$

$$d_{i+s/2} = \sin\left(\frac{2i\pi t}{s}\right),$$

where $t = 1, \dots, n$ with n being the sample size. When s is an odd number, it is round up $s/2$ to the closest integer. This encoding is mathematically equivalent to using s binary indicator variables, in which case each binary indicator would encode the level of a particular day in the season. In some cases, one of the indicators will correspond to a constant, resulting in $s - 1$ informative indicator variables. One major advantage of the trigonometric representation is that it can encode complex seasonal effects, such as leap years. For example, instead of setting $s_{annual} = 365$ it can be set to $s_{annual} = 365.25$ to account for the one additional day every four years. This is not possible with the binary indicator encoding. In this case, the number of the trigonometric pairs is computed as the floor of the true s , for example, for the day in the year we will have $365/2$ cosines and $365/2$ sines.

² For simplicity, the code provided will not be exactly as detailed in this paragraph and the next. Instead of selecting the dummies through a combination of AIC and Lasso regression from a large pool, a smaller pool is generated with no selection. The difference in outcome is negligible.

Figure 3. Seasonal Plots for the Three Seasonalities Present in the Time Series (The Bottom Three Plots Correspond to the Ones in the Top Row but Expressed as Boxplots)



Source: IMF staff

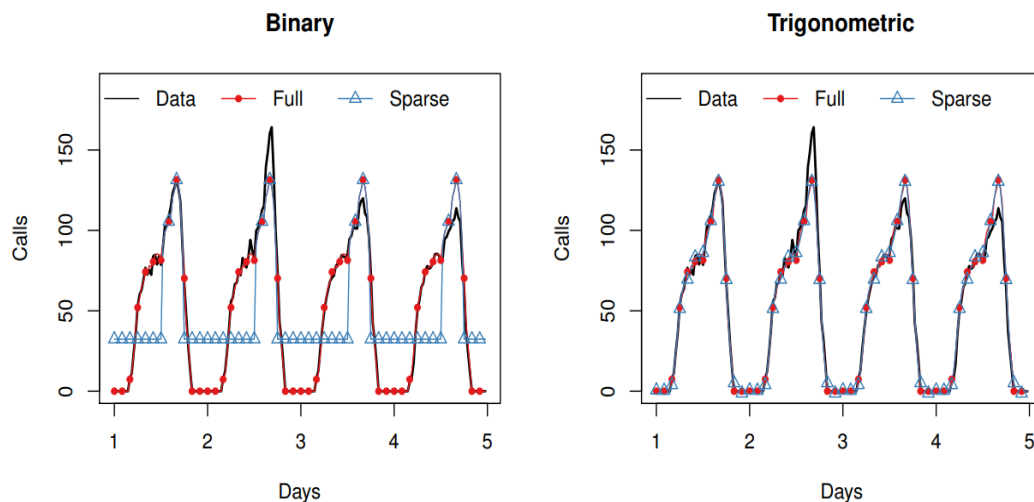
23. The team filters redundant seasonal indicators to make the model more parsimonious. The filtering is based on a two-step approach. First, the time series is a simple ARIMA model is fitted on the time series, accounting for automatically identified autoregressive and moving average terms, together with any indicator variables that relate to policy effects. Using the residuals of that ARIMA as a target variable, the team jointly estimates the inclusion of the day in the week and day in the quarter trigonometric dummies with a stepwise regression based on Akaike’s Information Criterion (AIC).³ The team uses a Lasso regression to eliminate superfluous terms because the number of indicator variables used for capturing the annual seasonality is high. Once the two groups of seasonal

³ The procedure is robust to different choices of information criterion.

trigonometric dummies have been filtered, they are pooled together and refiltered through an AIC based stepwise regression. Some of the trigonometric indicators between the seasonalities will be identical, as some of the seasonalities are encompassed. The last step ensures that all redundant variables are excluded.

24. Trigonometric indicator variables provide an efficient sparse approximation of seasonal patterns. Figure 4 exemplifies this point. Using all indicators, binary and trigonometric variables (for integer s) give the same output. However, when superfluous terms are eliminated, binary encoding omits all seasonal information for that period, while the trigonometric merely provides a smoother approximation of the seasonal profile.

Figure 4. Example of Full and Sparse Seasonal Encoding Using Binary and Trigonometric Indicators



Source: IMF staff

25. Special events indicators are chosen using an AIC stepwise regression. The process is similar to the one described above for the seasonality. Each indicator with its leads and lags together is potentially filtered, and once the first step of filtering is done, then all indicators are pooled together and the team performs one last iteration of stepwise selection. The target for this process is a time series of the residuals of an ARIMA, with automatically specified orders, including indicators for policy effects and the above trigonometric indicators. The seasonal indicators are selected first, and then the ones for the special events, as the former are typically associated with more variance in the time series.

V. FIRST-TIER: INDIVIDUAL FORECASTING MODELS

26. The team uses three benchmark models, the Naïve model (a random walk), an automatically specified Exponential Smoothing (ETS), and an automatically specified ARIMA. For the automatic specification of both models the team uses the AIC. Time series are differenced as needed to become level stationary, but no seasonal differences are used. To

identify the appropriate level differencing the IMF experts use the KPSS test. Furthermore, the term ARIMA is used in this report to also include Seasonal ARIMA, which may result from the automatic specification (the automatic specification potentially fits SARIMA if needed). For the exponential smoothing, the team relies on the state-space formulation of the family of models, which enables maximum likelihood estimation and automatic model selection. For both ETS and ARIMA only the day in the week seasonality is modeled. Ord et al. (2017)⁴ and Hyndman et al. (2008)⁵ provide a textbook presentation of these models.

27. The team also provides advanced forecasting models, augmented versions of the ETS and ARIMA models, with multiple seasonalities, and inputs for special and policy events. The benchmark models are extended to include any of the resulting indicators from the preselection of the indicators for the seasonalities and events. Improved seasonality modeling translates into an augmented model for exponential smoothing (ETStrig), and two augmented models for ARIMA (ARIMATrigA, and ARIMATrigB). The two ARIMA models differ in the automatic order specification strategies and, more importantly, in their estimation. ARIMATrigA (based on the forecast package for R) makes a sequential estimation of the parameters for the indicators and the ARMA orders of the model. In contrast, ARIMATrigB (based on the smooth package for R) makes simultaneous estimation. The former is expected to face a more straightforward specification problem but has potentially biased parameters. The latter avoids any biases due to the simultaneous estimation but faces a more complex specification problem.

28. The IMF experts also tested a neural network extreme learning machine (NN-ELM) and a TBATS model that performed poorly and were removed from the analysis. The neural network (NN) is a feedforward extreme learning machine. Its inputs are similar to the rest of the advanced models. Extreme learning machines are a modification of classical multilayer perceptrons to improve training times (see Barrow and Kourentzes, 2018,⁶ for specification details). The provided forecast is an ensemble of 20 networks to ensure reliable outputs, using the median operator (implementation using the nnfor package for R). The TBATS model has many similarities with the state-space exponential smoothing. It differs in that it encodes seasonality internally using trigonometric representation, and further allows for ARMA errors and pre-transformation of the time series to normalize its variance. Although the TBATS model doesn't allow for introducing external variables, such as indicators for special events, it has been shown to perform well on data with multiple seasonalities, and therefore it is useful as an advanced benchmark. More details can be found

⁴ Ord, K., Fildes, R., & Kourentzes, N. (2017). Principles of Business Forecasting. 2nd ed. Wessex Publishing.

⁵ Hyndman, R., Koehler, A. B., Ord, J. K., & Snyder, R. D. (2008). Forecasting with Exponential Smoothing: The State Space Approach. Springer Science & Business Media.

⁶ Barrow, D., & Kourentzes, N. (2018). The Impact of Special Days in Call Arrivals Forecasting: A Neural Network Approach to Modelling Special Days. European Journal of Operational Research, 264(3), 967-977.

in de Livera et al. (2011).⁷ In the case of CrI and CiC time series, ELM provides unreliable forecasts, while the TBATS faces substantial difficulties with the non-stationarity of the data and the shift due to the COVID-19 pandemic. Both models are consistently outperformed by the rest of the advanced models. Given their substantial computational cost, they are excluded from any further analysis.

29. IMF experts also consider Temporal Hierarchy Forecasting (THieF) to augment the various model forecasts further and capture the different seasonal components appropriately. Temporal hierarchies have two main advantages. First, it is often difficult to identify all the features of a given time series. When identifying best model parameters, model specification methodologies and optimizers focus on time series components that explain most of the variance. This feature is often desirable, as components that may have minimal effect in explaining the variability in the data are excluded, making forecasts resilient to overfitting. However, it can also be the case that these components carry valuable information. It is possible to extract these components through temporal aggregation. For example, the seasonal part dominates in a time series sampled at a monthly frequency. On the other hand, in annually aggregated data, the seasonality is filtered, and any long-term dynamics and trends become visible and easier to model.

30. THieF also reduces modeling risks associated with the use of single models. Using a single model entails modeling risks, which stems both from the specification and the selection of the single model. The specification and selection may be incorrect, as the underlying data generating process is unknown. One approach to reducing this modeling risk is to combine alternative models, in which case model diversity, i.e., capturing the information in the time series differently is found to be beneficial. Temporal aggregation enforces this diversity. THieF attempts to take advantage of the different views obtained from temporally aggregating a time series, modeling each different level with an appropriate model, and then combining these models to provide a holistic representation of the time series. This way, THieF forecasts use more available information and reduce the modelling risk, as they don't rely on a single model. The details of THieF can be found in Athanasopoulos et al. (2017),⁸ and a simplified discussion can be found at Kourentzes and Athanasopoulos (2021).⁹ Specifically for the application at hand, the following aggregation levels are considered: 1, 2, 4, 5, 10, 13, 20, 26, 52, 65, 130, and 260, to determine the number of days for aggregation. Due to the five-day weeks, the aggregation over 260 days corresponds approximately to complete years, while 65 corresponds to quarters. The other

⁷ De Livera, A. M., Hyndman, R. J., & Snyder, R. D. (2011). Forecasting Time Series with Complex Seasonal Patterns using Exponential Smoothing. *Journal of the American Statistical Association*, 106(496), 1513-1527.

⁸ Athanasopoulos, G., Hyndman, R. J., Kourentzes, N., & Petropoulos, F. (2017). Forecasting with Temporal Hierarchies. *European Journal of Operational Research*, 262(1), 60-74.

⁹ Kourentzes, N., & Athanasopoulos, G. (2021). Elucidate Structure in Intermittent Demand Series. *European Journal of Operational Research*, 288(1), 141-152.

aggregation levels can be mapped into standard calendar frequency similarly. The team uses structural scaling to obtain the combination weights for the forecasts produced at the different levels. Details of the estimation can be found in the references mentioned above. Finally, for the CiC time series, due to the limited sample size, only the first eight aggregation levels are. Since the time series starts at the end of 2018, there are very limited observations to build models at the very aggregate levels. The sample size is not a restriction for the CrI.

VI. SECOND-TIER: MODEL EVALUATION, SELECTION, AND COMBINATION

31. Alternative forecasts performance are benchmarked via out-of-sample rolling origin forecasts. The team produced five day ahead forecasts and retained the last 100 days as a test set. The rolling origin evaluation is implemented as follows. The training set specifies and trains all models and produces forecasts for the first five days of the test set. First, the out-of-sample forecast accuracy is recorded. Next, the training set is increased by one observation so that the models are re-specified and re-trained. The process is repeated until all the observations in the test set are exhausted.

32. The mean and median Root Mean Squared Error (RMSE) across forecast origins is the metric to evaluate forecast accuracy. Quadratic errors track the mean of the distribution of the forecast target, preferring unbiased forecasts, but are sensitive to extreme values. For this reason, the mean and median RMSE across origins are preferred. If the two metrics deliver similar results, it implies that the forecasts are stable across forecast origins. On the contrary, when there is a large difference between the mean and the median RMSE, particularly with substantially higher mean RMSE over the median RMSE, this implies that the forecasts are unreliable across periods, and there are events of extreme errors.

33. The rolling origin setup supports dynamic model selection across forecast periods. The user can identify forecasts that perform better locally by tracking the accuracy of past forecast origins. The performance tracking uses the mean RMSE of the $t+1$ to $t+5$ forecasts over the last five forecast origins. To increase the reliability of model selection, the IMF experts have used a window of five-period origins. The random part of the actuals heavily influences errors from a single origin in the evaluation period. As the evaluation window expands to include more periods, some of this randomness is canceled out. The selection of five periods is a compromise of some mitigation of randomness while still retaining the recent observations. Increasing the window to more periods expands the focus of the selection to older observations. The advantages of using more than a single period are discussed further in the forecasting literature when the merits of rolling origin evaluation are explained in details (Ord et al., 2017; Tashman, 2000).¹⁰

¹⁰ Tashman, L. J. (2000). Out-of-sample Tests of Forecasting Accuracy: An Analysis and Review. *International Journal of Forecasting*, 16(4), 437-450.

34. Forecast combination is used as an alternative selection. The forecasting literature has shown that forecast combination is generally beneficial over the use of a single forecast. Forecast combination does not require selecting a single model and therefore minimizes the risk of choosing and relying on a single potentially inappropriate forecast. The team uses equal weights in combining the forecasts. Equal weights limit the estimation errors in the combination weights that can affect negatively the performance of the combined forecast. Forecast combination builds both on the substantial weight of empirical evidence in the forecasting literature, and the more recent theoretical understanding on the effect of the estimation uncertainty of the combination weights to the final forecast. Kourentzes et al. (2019)¹¹ discuss forecast combination extensively.

35. Forecast pooling produces a single forecast from a set of projections, both for dynamic model selection and combination. The pooling selects only well-performing forecasts. Forecast pooling is used to offset the variability of the final forecast both from the selection and the combination described above. The notion of pooling suggests that instead of using all forecasts considered, the pool is restricted to the top-performing models. Forecast pooling is somewhat at odds with the theoretical understanding that forecast combination performs best when there is substantial diversity in the base forecasts to be combined. However, the team reconciles these two approaches by limiting the pool of models to those that capture partial or complete seasonal information, excluding the Naïve (Random Walk) from the pool of models. These models capture the information in the time series in different ways, which introduces diversity. Kourentzes et al. (2019) provide an overview of the ideas of forecast pooling.

VII. RESULTS, RECOMMENDATIONS, AND CONCLUSION

36. Table 3 summarises the results from the empirical evaluation for the currency issued, modeling at the daily level. Table 2 presents the mean and median RMSE and the mean rank of the RMSE across origins. The latter is calculated by first finding the ranking of the forecasts from a single forecast origin and then calculating the average across all origins. It differs from the mean and median RMSE. It only considers the ranking of the forecasts and not the size of the errors, providing a non-parametric comparison between the alternatives. The results for forecasting 5 and 24 days ahead are presented separately. For each error metric, the best-performing forecast is highlighted in boldface.

¹¹ Kourentzes, N., Barrow, D., & Petropoulos, F. (2019). Another Look at Forecast Selection and Combination: Evidence from Forecast Pooling. *International Journal of Production Economics*, 209, 226-235.

Table 3. Jordan: Summary Results for the Forecasting of the Currency Issued

Forecast	5 days			24 days		
	Mean RMSE	Median RMSE	Mean Rank	Mean RMSE	Median RMSE	Mean Rank
Naïve	29.62	19.35	4.70	49.27	35.89	4.34
ETS	30.31	21.94	4.70	56.93	40.11	4.94
ARIMA	30.77	23.14	5.01	54.44	39.87	4.73
ETStrig	28.30	18.15	4.54	64.45	54.31	5.54
ARIMATrigA	26.37	18.27	4.36	48.21	36.90	4.31
ARIMATrigB	26.49	19.58	4.44	46.58	35.52	4.27
Select	24.79	16.26	4.03	42.13	30.91	3.72
Combine	25.96	19.07	4.22	46.61	34.88	4.16

Source: IMF staff

37. The results for currency issued suggest that the ARIMA-based advanced models ARIMATrigA, ARIMATrigB, and the dynamic selection perform best. The naïve benchmark is very competitive, implying that the other benchmarks do not manage to add value. The same is true for ETStrig for the longer-term forecasts. This result is suggestive of increased randomness in the series that hinders many of the models. The differences between the mean and the median RMSE suggest that multiple outlying forecast errors are at play. The non-normality of the error distribution indicates that the mean rank is an excellent metric to rank the alternative forecasts. For the five-days ahead forecast, from the single models, ARIMATrigA is best. Both selection and combination perform better, with selection being best. The dynamic selection remains the best performing option for the long-term forecasts, followed by the combination and ARIMATrigB, which is now marginally better than ARIMATrigA. For all eligible forecasts, including indicators for the various special days improves accuracy.

38. Table 4 provides the results for the currency in circulation, with dynamic selection providing the best results. The table is structured as Tables 1 and 2. The advanced ARIMA-based models perform well compared to the alternative single model forecasts. For both horizons, the best mean rank of a single forecast is obtained by the ARIMATrigB. Nonetheless, both selection and combination perform better. Including indicators for the various calendar events is beneficial over the short term but does not add significant value for longer-term forecasts. In some rare cases, at longer horizons, including indicators for special events make the forecasts marginally less accurate. This pattern differs from the currency

issued because of the different sample sizes between currency in circulation and currency issued.

Table 4. Jordan: Summary Results for the Forecasting of the Currency in Circulation

Forecast	5 days			24 days		
	Mean RMSE	Median RMSE	Mean Rank	Mean RMSE	Median RMSE	Mean Rank
Naïve	39.82	29.49	4.82	54.82	42.79	4.17
ETS	42.09	29.78	4.81	66.44	51.65	4.91
ARIMA	42.21	34.43	5.49	65.71	49.56	5.08
ETStrig	38.96	28.40	4.97	97.98	87.97	6.37
ARIMATrigA	31.50	22.30	4.54	50.45	40.51	4.34
ARIMATrigB	27.92	18.42	3.83	47.83	33.00	3.79
Select	26.01	16.92	3.64	44.49	29.36	3.63
Combine	29.11	20.01	3.90	45.07	30.56	3.70

Source: IMF staff

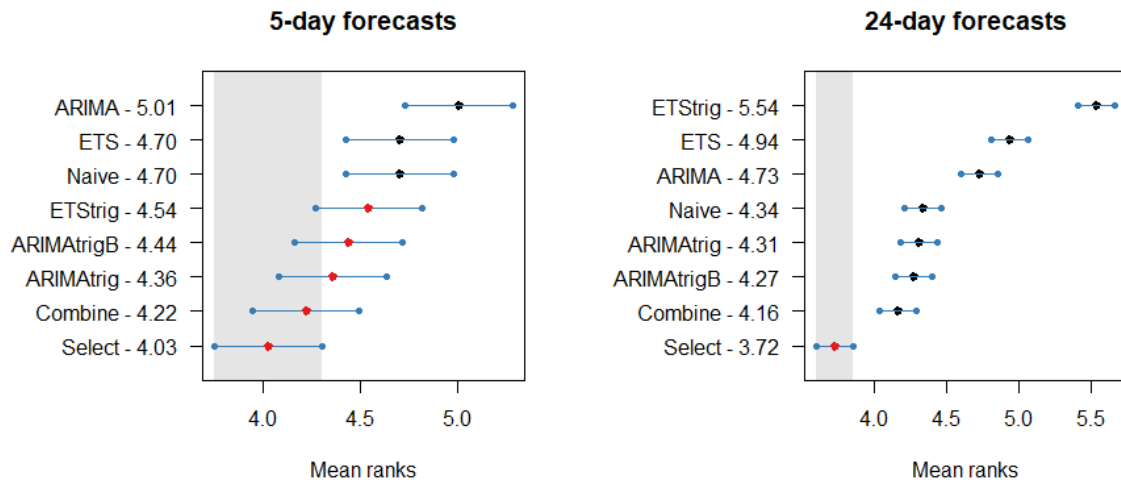
39. The dynamic model selection approach is the best performing model at a daily frequency. Dynamic model selection performs best for both time series and consistently across horizons and metrics. Nonetheless, this approach is more complex computationally as it requires calculating all forecasts in the model pool. ARIMATrigA or ARIMATrigB are parsimonious and are the top-performing single model forecasts. Finally, including indicators for the special calendar events is overall beneficial to the forecast accuracy.

40. Tests for statistical significance of the differences of the reported results confirm the results.¹² The team uses non-parametric testing, given the non-normality of the error distributions. Furthermore, because of the multiple forecasts, the IMF experts have used a combination of Friedman and Nemenyi post-hoc tests to avoid multiple testing that may inflate corresponding p-values. The Friedman test is the non-parametric equivalent to the ANOVA test, helping to find evidence that at least one set of forecasts is significantly different from the rest. Once the null hypothesis is rejected, the team then proceed with the

¹² The discussion here is for illustration purposes only, and this is not included in the code provided.

Nemenyi test.¹³ For each forecast, its mean rank is identified and a corresponding critical distance that depends on the significance level (we use 5 percent), sample size, and competing models. Forecasts whose mean ranks are within the margins set by the critical distance can be grouped together, as there is no evidence of statistically significant differences. Figure 5 provides the results for the currency in circulation forecasts. The horizontal axis reports the mean ranks, with the lowest being the best, also reported next to the name of each forecast. A dot indicates the mean rank. The lines around each dot correspond to the critical distance. The area within the critical distance of the best-performing forecast is highlighted in grey. For any forecasts within that area there is no evidence of significant differences. For the five day forecast, however, there is evidence of significant differences for the benchmarks, while all advanced models perform well. For the longer-term forecast the dynamic model selection exhibits significantly different accuracy than the rest. Figure 6 summarizes the results for the currency in circulation. The results suggest that all dynamic selection, combination, and ARIMAtrigB perform significantly better than the rest of the forecasts for both the short and long-term forecasts.

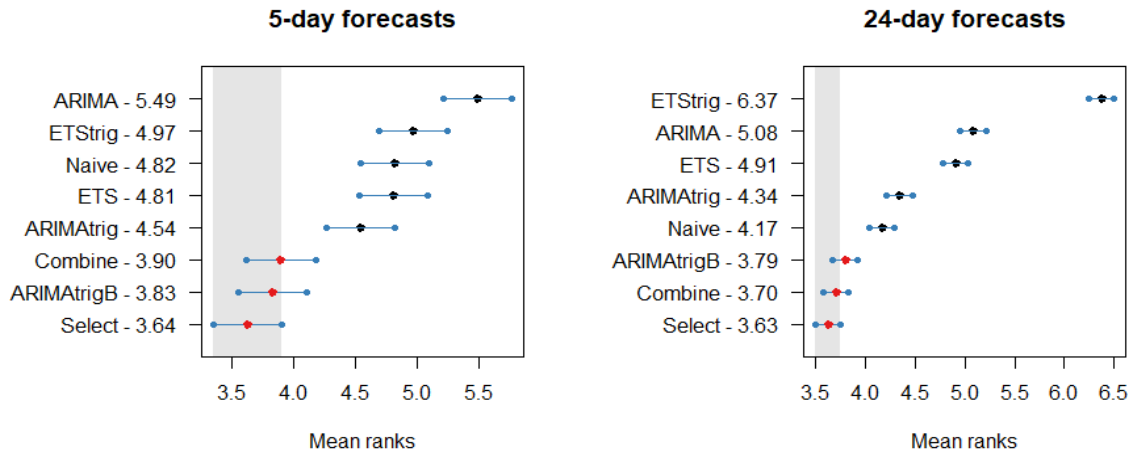
Figure 5. Nemenyi Test results for the Currency Issued Forecasts



Source: IMF staff

¹³ Additional details about the tests can be found at: Hollander, M., Wolfe, D. A., & Chicken, E. (2013). Nonparametric Statistical Methods (Vol. 751). John Wiley & Sons.

Figure 6. Nemenyi Test Results for the Currency in Circulation Forecasts



Source: IMF staff

41. Table 5 provides the difference in performance by using THieF for the case of the currency in circulation.¹⁴ The table provides the differences in the results for the best three options as identified before, the ARIMAtrigB, dynamic selection, and combination. There are marginal gains in Mean RMSE in the short-term but losses in the median. For the long-term, THieF exhibits constant gains in performance. This result is to be expected, as THieF is well documented to provide most gains for longer-term forecasts. However, the team recommends **against** the usage of temporal hierarchies due to the increased complexity that this entails for only modest gains in accuracy.

Table 5. Jordan: Summary Results for the Forecasting of Currency Issued Using Temporal Hierarchies

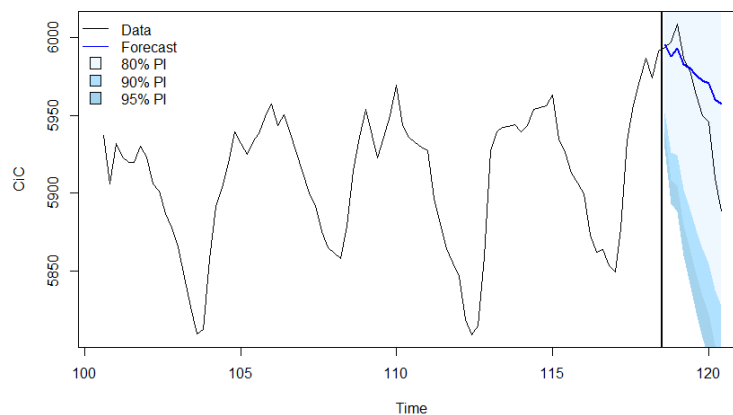
Accuracy difference	5-days		24 days	
	Mean RMSE	Median RMSE	Mean RMSE	Median RMSE
ARIMAtrigB	0.53	-0.30	4.14	2.12
Select	-3.04	-4.08	5.20	3.42
Combine	1.76	-0.21	7.10	2.17

Source: IMF staff

¹⁴ As there is little gain from the use of THieF, it is not included in the code provided.

42. The recommended forecasting approaches, ARIMA with trigonometric seasonality and dynamic model selection, have the advantage of providing prediction intervals. The forecasts are based on statistical models for which there are analytical formulas for asymptotic behavior. Therefore, it is straightforward to obtain probabilistic forecasts. Probabilistic forecasts provide an intuitive view of the uncertainty of the forecasts. Figure 7 exemplifies this by providing a forecast with its 80 percent, 90 percent, and 95 percent prediction intervals. Given its past (in-sample) performance and its specification, the model anticipates that 80 percent, 90 percent, and 95 percent of the future observations will fall within these intervals, respectively. A miscalibrated model may provide unrealistically narrow intervals, which can be a symptom of multiple modeling issues. A common problem can be overfitting the in-sample data. In our collection of forecasts, the forecast combination does not readily provide prediction intervals. Nonetheless, these could be empirically calculated.¹⁵

Figure 7. A Forecast with its 80 percent, 90 percent, and 95 percent Prediction Intervals



Source: IMF staff

43. The authorities should regularly operate the ARIMA with trigonometric seasonality model but update the parameters of all the models once a month—or when a structural break occurs. The IMF has provided the authorities with a full-fledged software infrastructure, described in Appendix I and II. The entire script is re-estimating all the models and assessing the benchmarking comparison performance, and it is pretty time-consuming. For this reason, the central bank should execute the whole code once a month—or when there is a major structural break—to update the parameters of all models. However, for daily operations, the authorities should directly run the model they prefer (e.g., ARIMA

¹⁵ For example, of empirical prediction intervals, see: Trapero, J. R., Cardós, M., & Kourentzes, N. (2019). Empirical Safety Stock Estimation Based on Kernel and GARCH Models. *Omega*, 84, 199-211.

with trigonometric seasonality) with the parameters estimated previously. A single model forecast runs very fast and could be executed every day within a few seconds.

44. The central bank could enhance the forecasts by incorporating further the expertise and granular knowledge of the CBJ experts. This expertise can improve further the construction and selection of indicators for special events and policy events in our models. Furthermore, the IMF team has followed an agnostic approach, building on information that practitioners could extract solely from the data. Contextual data and expertise can further refine modeling choices to improve forecast further.

ANNEX I. SOFTWARE DOCUMENTATION

The documentation describes the R code used during Jordan's mission on forecasting currency in circulation, what it aims to do, how to run it, and the underlying models and outputs.

The Language

The code is written in the R language, and information is available at <https://www.r-project.org/>. R is an open-source language, meaning that all source code is available and is made of contributions by many people worldwide. It is also freely available, so the language and other libraries are also available to anybody with an internet connection free of charge.

The team chose the R language for this project because R has many statistical and forecasting packages that are easy to use “out-of-the-box.” Abundant pre-coded packages allow implementation easily under R, such as ARIMA, exponential smoothing, and other models. R can also write out into familiar formats like csv and Excel and create visualizations that can be exported to PDF format, making the output readable to a large audience of non-specialists.

The Environment

For most users of the R language, RStudio <https://www.rstudio.com/> is the de facto environment for reading, writing and executing R code. RStudio is also free.

and is used by statisticians, economists, and other professionals all over the world. To install RStudio, please visit <https://www.rstudio.com/> and install the version corresponding to the operating system.

To run the code provided by the IMF team, it is recommended to use R version 3.6.3 so that the library version is the same. For a Windows environment, the IMF provides a folder containing the necessary libraries to run the code. The user should place this code in the same folder as the R source code, and this will ensure that the version of all libraries also matches.

Files

The IMF provides the following files:

- R scripts
- config file
- documentation (this file)

R scripts are files that have the extension .R, and contain commands that can be interpreted by the R language and be executed. These files contain all of the instructions to forecast currency in circulation and can be used for general forecasting of time series. All R scripts use a config file to set up parameters for the instructions. These include input file names, days of the week that are weekends, models to run, output files, and others.

To execute an R script it is easiest to open the file in RStudio, and execute either one line at a time or all at once. A separate session can be set up with the IMF to provide training on how to use RStudio and run R scripts.

Scripts

There are four R scripts provided:

- step_000_dataclean.R
- step_001_features.R
- step_002_model.R
- step_003_summary.R

These files should be run in order, as the output of each step is then used as the input for the next step. All of these files should read in the same config file, which is specified with the line

```
source("config.R")
```

The advantage of this approach is that by editing the config file, which is a reasonably easy-to-understand file, the user can have some flexibility when running the R code even if they are not too familiar with the R language.

The config file and R scripts have been heavily commented for clarity.

Step 0 - Cleaning Data

The file step_000_dataclean.R does the following:

1. Read in the specified Excel or csv files from the `rawdatafile` parameter in the config file.
2. Set the start and end dates and remove bad dates based on the config file parameters.
3. Add a column `datatype` with values of data, holiday, and imputed.

4. Remove weekends based on the `exclude_days` parameter in the config file.
5. Linearly interpolate any dates that are missing in the data.
6. Save the output as specified by the `dataclean` parameter in the config file.
7. Create a plot of the target variable and save as PDF file.

This script should take only a few seconds to run. The original input file(s) should have dates in the first column, and the target variable column should be specified by the `targetcol` parameter in the config file. In the simplest case, the file will have two columns, one for the date and another for the target variable, with `targetcol` being 2.

Step 1 - Adding Features

Once the original data is read in and cleaned, the next step is to add additional features for each date to help with the modeling. The file `step_001_features.R` does the following:

1. Read in the cleaned csv file from step 0.
2. Generate daily biases for each day of the week.
3. Generate trigonometric biases for each day of the month.
4. Generate trigonometric biases for each day of the year.
5. Add structural breaks based on the `structural_breaks` parameter in the config file.
6. Add bias terms around the 20th of each month.
7. Add bias terms for the quarter.
8. Combine all of the bias terms with the data and save as specified by the `dataFeat` parameter in the config file.
9. Plot all features and save as PDF files.

Each of the biases are background terms that are used to model different seasonalities. For example if the data showed a very strong tendency for Mondays to have higher values, the bias corresponding to Mondays would be modeled with a higher value to reflect this. In general if there are N weekdays that are used (typically with $N=5$), then there will be $N-1$ biases so that each weekday gets a unique bias.

Similar constructions are created to model how much within the month, quarter or year a particular date is in. In the case of CiC there can be structural breaks where the value changes rapidly for given dates, the `structural_breaks` parameter allows modeling of each structural break as different biases.

All of these biases are combined with the original data and saved as a csv file.

Step 2 – Fitting the Model

The R script `step_002_model.R` does all of the modeling based on the output csv file from step 1. The script does the following:

1. Read in the csv output from step 1 containing the data and all features (biases)
2. Create a matrix of features
3. Run the function `forecastLoop` that runs all models specified in the config file over a range of dates.
4. Save the forecasted results as an Excel file.

The function `forecastLoop` takes in a list of index values to run on, the data `Y`, the length of the validation set `testLength`, the forecast horizon `horizon`, the dates in the data `dates`, a matrix of features for each date `X`, and the models given by `modelLabels`. The function will iterate over the list of index values and for each one will generate a forecast for that date using all models given by `modelLabels`. The benefit of this approach is that each forecast date is treated independently. By editing the `modelLabels` parameter in the config file, the user can specify the needed models. The models available are:

1. **Naive** This model assumes a random walk with no parameters and forecasts future values with the most recent value. It can be considered a benchmark against which any model should perform better.
2. **SNaive** Short for “Seasonal Naïve,” this model uses the base frequency of the data (typically five days per week) to forecast future values using the five most recent values repeatedly. If the data has a weekly periodicity, the expectation is that this model will perform better than the Naive model.
3. **ETS** Short for “ExponenTial Smoothing”, this model is an autoregression where the coefficients are weighted more heavily to recent dates. Typically one parameter will control how much recent dates are weighted compared to past dates.

4. **ARIMA** The ARIMA model is a well-known model which allows autoregression of a time series. The `auto.arima` function used can select the best ARIMA model within the specified range of the parameters `max.d`, `max.p`, `max.q` which correspond to the usual `d`, `p`, `q` parameters of an ARIMA model.
5. **ETSfeat** This is a model using externally specified features. In this case, the external features are the seasonal biases that were generated in step 1.
6. **ARIMAfeat** This is the version of ARIMA models with external features, again using the seasonal biases from step 1.
7. **ARIMAtrigB** An alternate version of ARIMAfeat, it uses the same external features, but uses a state-space ARIMA model.
8. **TBATS** Based on the model initially developed by De Livera, Hyndman, and Snyder, this is an exponential smoothing state-space model with Box-Cox transformation to make the seasonal variation similar across the whole series.
9. **TBATSfeat** A two-step approach that first uses an exponential smoothing model to fit the overall shape, then fits the residuals of the initial fit using a TBATS model.

Once all models are fit for all periods, the forecast results are saved as an Excel file.

Step 3 – Summarizing the Results

The file `step_003_summary.R` will do the following:

1. Read in the Excel file of forecasts.
2. Calculate the errors between the true values and the forecasts.
3. Calculate the RMSE (root mean square error) for each model.
4. Calculate further models that either select the “best” models or the mean of several.
5. Write out the results of the RMSE calculation as a csv file.
6. Plot the RMSE values and save as PDF.
7. Plot the forecast values together with the original data.

Once the Excel file from step 2 for the forecast values is generated, these can be compared to the true values and errors can be calculated. Two further composite models are then introduced. One is to use the model with the best rolling error of each model for several dates leading up to the forecast date, and choosing the model with the smallest errors. Another is to

take the mean of several models. Finally, these models are calculated and RMSE values are calculated for all models.

The forecast values and data plots are merged together to form one large PDF file, with each page showing different forecast origins. By flipping through these files, one can visually see what each model is doing and its comparison to the data.

For help with the code, please contact Kei Moriya (kmoriya@imf.org) and Romain Lafarguette (rlafarguette@imf.org).

ANNEX II. USER GUIDE FOR R CODE OF CURRENCY IN CIRCULATION

IMF R Programs/Libraries for Forecasting Autonomous Factors

Kei Moriya, Adeleke Adeyemi (IMF, IT Department)¹

This document describes how to set up and run the R programs and libraries that were used to forecast the Jordan currency in circulation. R is an open source language with many third-party libraries that are constantly evolving, the main difficulty is ensuring that the versions of all libraries match. In cases where they do not, this can lead to warnings, errors or in some cases insidious changes that go unnoticed. To mitigate these risks, below we describe how to set up an R environment containing the same libraries as was used at the IMF.

1. Installation of R/RStudio

The easiest interactive environment to read/write/run R code is RStudio. Download the latest version at <https://www.rstudio.com/>.

For versions of R, use 4.1.0 or 4.1.1, there should not be any significant differences.

2. Zipped Library Files

The R community has many third-party packages such as the tidyverse package <https://www.tidyverse.org/> for data manipulation and graphical output, the forecast package <https://pkg.robjhyndman.com/forecast/> for forecasting time series, and many others for various operations such as reading/writing Excel files, running machine learning algorithms, etc. As these packages are constantly evolving, it is necessary that the code match the version of each library, otherwise this may lead to various failures.

For this reason we have zipped together all third-party libraries that were used to write and test the code for the project. The code can be downloaded via the One Drive link that was sent to you.

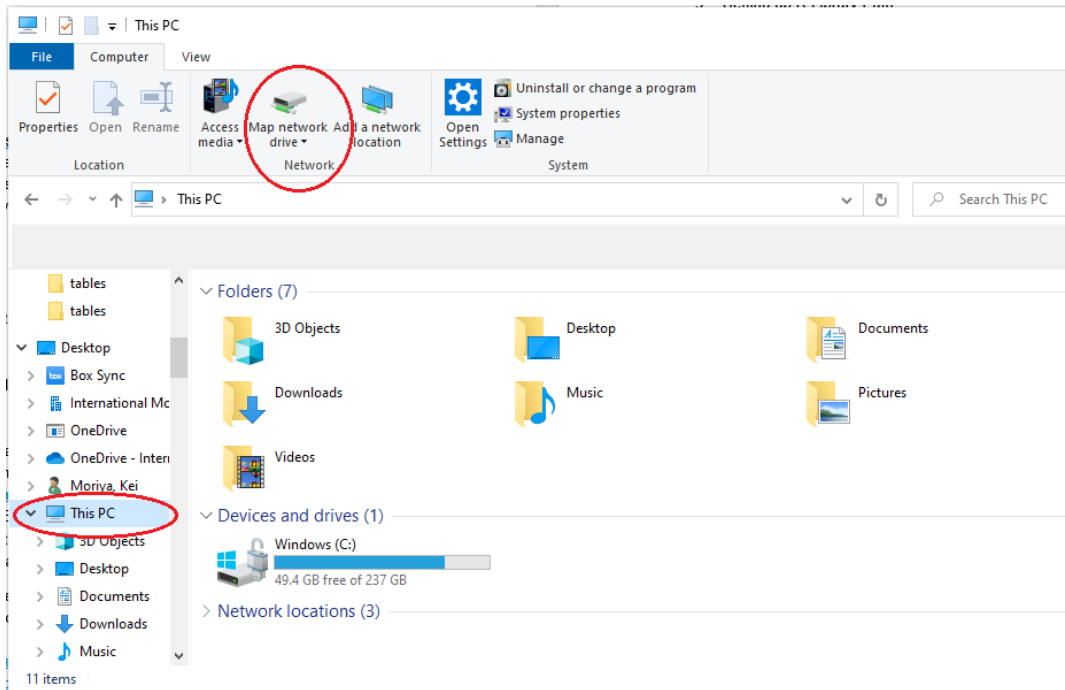
The file library_JOR.zip (approximately 197 MB) should be downloaded and unzipped.

Next, it is recommended that this folder be put on a common network location so everybody can access it. Having it on the network will slow down reading in the libraries somewhat, but the benefit is that if updates are necessary, there is only one

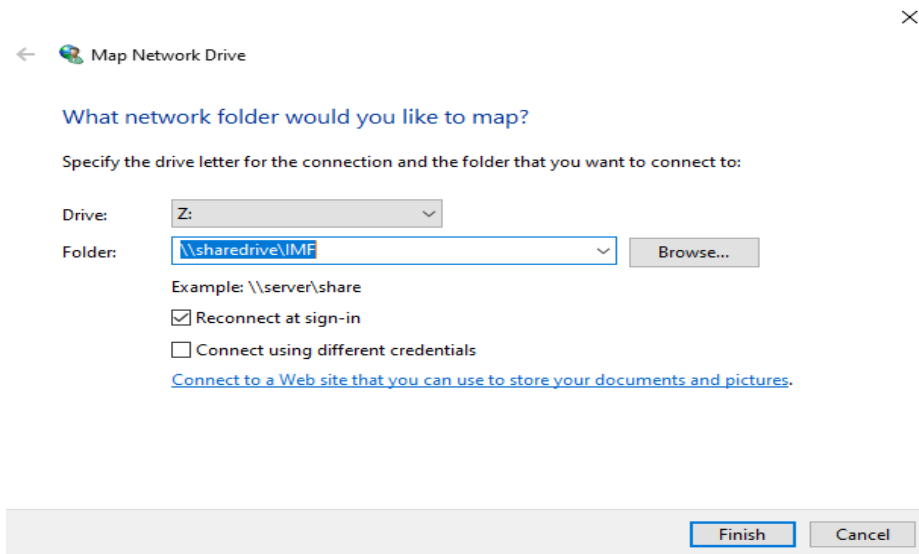
¹ Contact: kmoriya@imf.org, aadeyemi@imf.org.

location to change. Of course, if users can create their own local copies if necessary, but these will need to be synced to any future changes.

Note that **the shared drive must be mapped as a drive on the computer that wants to run the code**. For example, if the unzipped files are placed under [\\sharedrive\IMF\library](https://sharedrive\IMF\library), open a Windows Explorer window and navigate to “This PC” and select “Map network drive” at the top.



In the window that opens select an available character for the drive name and map the folder:

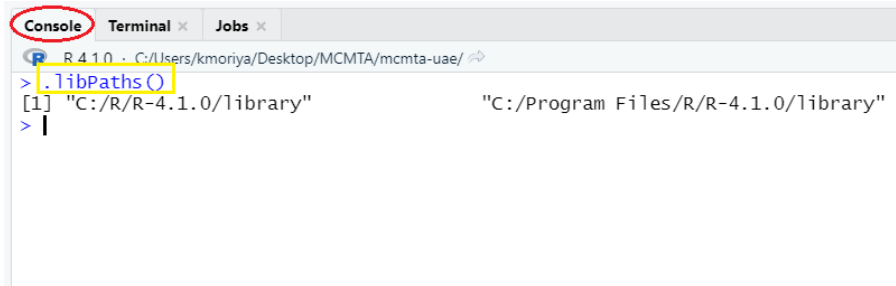


This procedure must be done for all computers that want to run the code.

3. Setting up R Library Path

Once the unzipped library folder's location is determined, the path that R uses to find libraries must be modified.

Within RStudio, in the bottom left window, find the tab Console and type `.libPaths()` (note the period at the beginning):



```
R 4.1.0 · C:/Users/kmoriya/Desktop/MCMTA/mcmta-uae/ ↗
> .libPaths()
[1] "C:/R/R-4.1.0/library"          "C:/Program Files/R/R-4.1.0/library"
> |
```

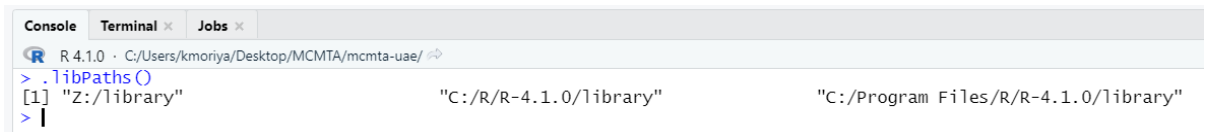
This will show where R will search for third-party libraries. We will modify this with the following commands:

```
.Library.site <- file.path("Z:/library")
```

```
.libPaths(.Library.site)
```

where `Z:/library` is the location of the unzipped files. These two lines should be added to the beginning of each R script.

Note that once this command is executed, the new list of paths can be checked, and the mapped path should appear first:



```
R 4.1.0 · C:/Users/kmoriya/Desktop/MCMTA/mcmta-uae/ ↗
> .libPaths()
[1] "Z:/library"                   "C:/R/R-4.1.0/library"          "C:/Program Files/R/R-4.1.0/library"
> |
```

Since this operation does not remove the original places that R looks for libraries, it is possible to install additional libraries at these locations, and if they are not found in the mapped location these will be used. In this way we set up an environment where the libraries under the mapped drive are guaranteed to match those used at the IMF while still allowing the flexibility to install further packages.

4. Running the R Scripts

After adding the two lines above in each R script, they can be run with the Source button:



```
1 # Change path to R libraries
2 .Library.site <- file.path("Z:/library")
3 .libPaths(.Library.site)
4 |
5 library(tidyverse)
6 library(readxl)
7 library(zoo)
8 library(ConvCalendar)
9 library(lubridate)
10 library(forecast)
11 rm(list=ls())
12
13 #####
14 ### Currency in circulation ###
15 #####
16
17 ###Load and clean cjc data
18
19 #Base of filename (change depending on directory structure)
20 filebase<-"data/raw/autonomous_factors/"
21
```

For help with the code, please contact Kei Moriya (kmoriya@imf.org) and Romain Lafarguette (rlafarguette@imf.org).