uc3m | Universidad Carlos III de Madrid

Master Degree in Statistics for Data Science
Academic Year (2019-2020)

Master Thesis

"Non-modellability under the Fundamental Review of the Trading Book (FRTB) framework."

Rafaela Berenice Becerra Robalino

André Alves Portela Santos
Carlos Abraham Catalán García

Madrid, 2020

AVOID PLAGIARISM
The University uses the **Turnitin Feedback Studio** program within the Aula Global for the delivery of student work. This program compares the originality of the work delivered by each student with millions of electronic resources and detects those parts of the text that are copied and pasted. Plagiarizing in a TFM is considered a **Serious Misconduct**, and may result in permanent expulsion from the University.

This work is licensed under Creative Commons **Attribution – Non Commercial – Non Derivatives**

SUMMARY

The Basel Committee on Banking Supervision (BCBS) as a response to the financial crisis of 2007-2008, proposed new banking regulations to strengthen the financial system. One of the most recent is the Fundamental Review of the Trading Book (FRTB) referred to as "Basel IV", which acts as a comprehensive suite of capital rules that will be applied to the banks' wholesale trading activities.

Among other changes, the FRTB includes new rules to measure market-risk, introducing the concept of "modellability". Then, if a risk factor does not meet the "modellability" criteria, it is deemed to be treated as a "Non-modellable Risk Factor" (NMRF) which drifts to a higher charge on the capital requirement for banks.

Considering that NMRFs can be expressed in terms of "Modellable Risk Factors" to reduce the capital requirements and that the number of risk factors managed by the institutions is large, diverse methodologies could be used to address a process of selecting the combination of most suitable MRFs to express each NMRF.

Under this framework, this study focuses on the exploration of diverse relevant feature selection techniques for Machine Learning to find the most adequate methodology. This will be the one that is able to return the largest quantity of proxies of NMRFs that surpass the test proposed by the regulator and move onwards to replace the corresponding NMRFs for the capital requirement calculation, avoiding extra charges.

To conduct this study, data of a trading desk of a financial institution has been used, and seventeen different methods have been analyzed to arrive at a methodology involving six possible techniques to determine the NMRFs of this desk by means of MRFs, highlighting in the process key insights for the expansion of this approach over more and new information.

Keywords: Fundamental Review of the Trading Book (FRTB), Basel IV, Feature Selection, Non-modellability, Modellability, NMRF, MRF, Risk Factor, Market Risk.

DEDICATION

I want to dedicate this work to God for letting me achieve one more goal. To Galo and Mireya, my mom and dad, who have always been there for me, for all the love and the sacrifices that have made for my education. To my sister Carolina, who is an outstanding role model. To Emilio for being there by my side at every milestone, for being the one that has shown me that we can reach more and greater goals together, for all the love and effort throughout this year.

I love you beyond words...

A special feeling of gratitude to my friends and professors that have made this experience a path of constant learning.

Also, I would like to express special thanks to my supervisors Carlos and André for the unconditional support and dedication that have invested since day one. Certainly, your effort has been key to the development of this work

CONTENTS

1. INTRODUCTION 1
2. CURRENT SITUATION 4
3. METHODS 5
3.1. Feature selection 5
3.2. Dataset 6
3.3. Assumptions and considerations 8
3.4. Partition 8
3.5. Notation 9
3.6. Preprocess. 9
3.7. Algorithms 10
3.7.1. Variable selection 14
3.7.2. Dimensionality reduction 22
3.7.3. Additional approaches 27
3.8. Performance measures 28
4. RESULTS 30
4.1. Partial Results 30
4.1.1. Count of maximum performance metrics per Model 31
4.1.2. Average MAE and RMSE 31
4.1.3. Selected methods 33
4.2. Total results: conservative partition. 35
4.3. Total results: regulator partition. 38
4.3.1. Characterization of NMRFs proxies by zones 38
4.4. Methodology results 41
4.4.1. Characterization of the selected predictors. 41
5. CONCLUSIONS 44
6. FUTURE LINES OF WORK. 45
BIBLIOGRAPHY. 46

7. ANNEXES . . . . . .
7.1. Annex 1: Assessment for Internal Model Approach . . . . . .
7.2. Annex 2: Project Timeline . . . . . .
7.3. Annex 3: Feature selection methods . . . . . .
7.4. Annex 4: Diagram of methodology for data preprocessing . . . . . .
7.5. Annex 5: Programming flow . . . . . .
7.6. Annex 6: Additional results of count of NMRFs . . . . . .
7.7. Annex 7: Additional results of average metrics . . . . . .
7.8. Annex 8: R packages . . . . . .
7.9. Annex 9: Glossary . . . . . .

LIST OF FIGURES

4.1 Characterization of the NMRF sample by term and type 30

4.2 Characterization of the NMRFs sample by currency 30

4.3 Missing values in partial results 31

4.4 Partial results based on maximum metrics 32

4.5 Partial results based on normalized average MAE and RMSE 34

4.6 Computational time for each method in partial results 35

4.7 Missing values in total results 36

4.8 Total results based on normalized average MAE and RMSE 37

4.9 Composition of proxies for NMRFs based on zones 38

4.10 Composition of proxies for NMRF based on zones, by term and type . . . 39

4.11 Computational time for methodologies 39

4.12 Zone composition of proxies for NMRF by methodology 39

4.13 Zone composition of proxies for NMRF based on zones, by methodology, term, and type 40

4.14 Comparison among final methodology and minimum RMSE 42

4.15 Characterization based on type of NMRF and type of predictors, considering term 42

4.16 Characterization based on the currency of NMRF and the currency of predictors 43

4.17 Characterization based on the type of NMRF and the type of predictors, considering currency 43

7.1 Assessment for Internal Model Approach . . . . . .

7.2 Project Timeline . . . . . .

7.3 Feature selection methods . . . . . .

7.4 Methodology diagram: Data Preprocessing . . . . . .

7.5 Programming flow . . . . . .

7.6 Total results for regulator partition based on counts of maximum performance . . . . . .

7.7 Total results for regulator partition based on average MAE and RMSE . .

LIST OF TABLES

3.1 Notation 10

3.2 Requirements Model 1. Correlation filter 15

3.3 Requirements Model 7. Stepwise 17

3.4 Requirements Model 8. Ridge Regression 19

3.5 Requirements Model 9. Lasso 19

3.6 Requirements Model 10. Elastic Net 21

3.7 Requirements Model 11. Reluctant Generalized Additive Model 22

3.8 Requirements Model 12. PCR 23

3.9 Requirements Model 13. PLS 24

3.10 Requirements Model 14. ICR 24

3.11 Requirements Model 15. SPLS 26

3.12 Requirements Model 17. XGBoost 28

3.13 PLA metrics criteria 29

4.1 Total results measure as counts of maximum performance 36

7.1 Graphical R packages . . . . . .

7.2 Statistical R packages . . . . . .

7.3 Other R packages . . . . . .

1. INTRODUCTION

After the high volume of losses and bailouts carried out by governments during the financial crisis of 2007-2008, it became clear that banking regulations were not strong enough to provide financial stability. Therefore, the Basel Committee on Banking Supervision (BCBS) decided to implement a series of regulatory updates to frameworks, especially on market risk issues, due to the shocking drop in trading portfolios. Part of this new regulatory framework is the Fundamental Review of the Trading Book (FRTB), which introduces the new concept of “Non-Modellable Risk Factors” (NMRFs) that are risk factors or market rates and prices that affect the trading position of a bank and whose capital requirement is greater than for the called “Modellable Risk Factors” (MRFs). In the light of the need for new methodologies to reduce the impact of the new regulations on the capital requirement, this paper explores, from a Machine Learning point of view, different methods to find quality proxies of MRFs to express NMRFs, decreasing capital charges.

The Fundamental Review of the Trading Book (FRTB), referred to as Basel IV, is prescribed with the aim to "contribute to a more resilient banking sector by strengthening capital standards for market risk" in response to the weakness of the framework for capitalization in trading activities exposed during the financial crisis [1]. The implementation of the FRTB, required by January 2023, will carry critical challenges to banks in provisioning, operations, and technology infrastructure. The increases in capital requirements estimated lay in an additional 40%, and the costs related to implementation programs could reach $250 millions for big institutions. There will also be implications in the cost of Business as Usual operations as more requirements will be set at desk level and a considerable investment in technology will be demanded to measure risk [2]. In consequence, banks will have to take diverse considerations to reduce the impact of the implementation of the new framework and develop strategies to diminish potential complications in the future.

Market risk is defined as "the risk of losses in on-and off-balance sheet risk positions arising from movements in market prices" [1, pg.122]. Moreover, a risk factor can be understood as a determinant of the variations in a transaction’s value that models a risk position. There are several broad categories of risk factors: interest rates, credit spread, equity, foreign exchange, and commodities, each involving subcategories and a specific treatment [3].

Under the FRTB, each bank will have to choose among two methodologies to determine its market risk: the Standardised Approach (SA) [1, pg.30-43] or the Internal Model Approach (IMA)[1, pg.23-30]. If the second option is followed, to prove the eligibility of its trading activities, a process of approval will be held supervised by the correspondent financial authority [1, pg.83-87]. This one starts with the assessment of the Bank’s

infrastructure and its internal risk capital model. Next, there has to be a decision of the trading desks that will follow this capital treatment. Trading desks are understood as independent units of business defined internally by banks, which englobe instruments that present common strategies and the same treatment for measuring risk. Once again, there has to be an assessment over each picked trading desk and should only remain in scope those that satisfy minimum criteria in PL attribution tests, daily backtesting, and model-independent assessment tools [1, pg.26-28]. The last step is the individual analysis of each risk factor in the remaining desk eligible for the Internal Model Approach regulatory capital treatment. This classification of factors involves the distinction of "modellable" and "non-modellable" risk factors, both involving a different calculation of capital requirements. These three steps are detailed graphically in **Annex 7.1**.

According to the regulation [4, pg.74-75], to define a risk factor as "modellable" it needs to present a "sufficient set of representative transactions" this sufficiency is fulfilled by two conditions: "real" prices and an acceptable frequency. A price is considered "real" only if the bank has transacted on an arms-length basis, represents a transaction with a third party, or is given by a committed quote. On the other hand, the acceptable frequency refers to at least 24 observations per year and more than four price observations in a 90-day period, or 100 observable observations, one per day.

The MRF under the IMA will be capitalized through an aggregate capital charge based on the weighted average of an unconstrained expected shortfall (ES) and an expected shortfall charge for the regulatory risk factor classes. The risk factors deemed to be treated as "non-modellable" (NMRF) will need to be capitalized at least as conservative as the expected shortfall of "modellable" risk factors. If the stress scenario passed by the bank for the unmodellable risk is not accepted, then the capitalization will correspond to the maximum loss. In a quantitative impact study QIS conducted by BCBS [5], some banks that followed a SA approach presented capital requirements 2 to 3 times grater than if they could follow IMA. Under these considerations, there can be certain considerable capital charges, then strategies for managing “non-modellable" risk factors are needed. [6] presents some straightforward methodologies, among them is the decomposition of the NMRF into a "modellable" benchmark and a residual basis deemed as unmodellable, and when the component of NMRF is large, contemplates the fallback to the use of Standardized Approach considering that the capital charge could be less.

The classification of risk factors and the following determination of capitalization for NMRF is seen as a critical component of the requirement for market risk [7] due to the severe capital charges that could imply that banks will be less motivated to the incursion on the diversification of their portfolio. Following the line of the strategy of a decomposition of the NMRF and considering the large amount of risk factors that banks face, this paper aims to explore diverse methodologies that could be used to generate an automated process. This will consist in determining the "modellable" risk factors that could be used to re-express the "non-modellable" ones, minimizing the residual basis, which will have a positive impact on the regulatory capital requirements decreasing the

punitive charges.

The presented methodology uses data from a trading desk that presents 23‘918,848 daily variations of interest rate risk factors that will be categorized into MRFs and NMRFs. Next, it proposes the exploration of seventeen possible feature selection methods in a sample of 100 NMRFs with a possible set of 800 MRFs, to finally select a set of methods that present the lower error measurements in the prediction for a total of 4260 NMRFs. In the last step, based on the regulation, a PLA test will be generated to determine the number of risk factors that can be treated in part as MRFs, decreasing the capital requirements, and proving the effectiveness of the selected methods.The project timeline can be seen in the **Annex 7.2**.

The structure of this document follows: **Section 2** offers a review of the current methodologies proposed by other authors to treat “non-modellability” under the FRTB. **Section 3** takes the reader through the phases of the project presenting the methods that will be explored for the decomposition of the NMRF; additionally, presents the dataset, its preprocessing, and describes the executed code programs. **Section 4** shows the results obtained in the selected methods. Finally, **Section 5** presents conclusions and future work left to discuss.

2. CURRENT SITUATION

FRTB has been catalogued as the biggest change in market risk regulations in the past 20 years [8], therefore since its release (October 2013) several research and consultancy papers have been published with the aim of exposing different approaches to overcome the difficulties that this overhaul could bring to banks.

Among all, we can highlight [8] which gives a short review of the challenges derived from the Minimum Capital Requirements referred as to FRTB; [9] which provides a full revision of modern regulatory frameworks for commercial banks and its evolution; [10] describes the macroeconomic effects derived by Basel III and former regulations by providing an exhaustive review of studies made by several entities and authors.

As mentioned before, one of the main concerns is the classification of risk factors to reduce the "non-modellability" portion and decrease the capital requirements. After the release of the regulation the BCBS conducted a quantitative impact study QIS with 78 banks which showed that the risk capital requirement derived from the non-modellabiality for 43 banks, in average, represented 7% of the total market risk capital requirements but for some banks it reached 60% [5], exposing the potential impact of the new regulation. About this specific topic, few papers can be found: [11] questions the conservative capital requirements considering that the probability of meeting the criterion to be modellable is low. [12] considers the machine learning tool of Gaussian Regression Process to provide a more conservative approach for calculating the Stressed Expected Shortfall (SES) which is the capital requirement for NMRF. Furthermore, recent studies like [13] propose different proxies for NMRF which contemplate simple alternatives like extrapolation, linear regression, PCA, and parametric methods, specifically SABR which expresses the implied volatility, to express NMRF with MRF.

Most of the previous work does not consider the amount of data that should be treated. In concordance to the literature, a Machine Learning feature selection approach could respond to the need of methods capable of improving underlying methodologies and be implemented to treat new quantitative rules [14][15].

3. METHODS

Banks present a large number of risk factors; then, there is a need to select which combination of "modellable" risk factors is a better suit for each "non-modellable" factor. This problem can be addressed with feature selection methods.

3.1. Feature selection

Feature selection can be understood as the process of identifying and discarding irrelevant and/or redundant information, which implies the co-presence of other features that could be remove and without affecting the learning process [16, pg.3].

Now, it is essential to define what “relevant” can imply. There are multiple definitions associated with it [17, pg.248-250] [18, pg.14-15]. According to [18], it can be defined as finding the minimal optimal problem referring to the smallest set of powerfully relevant variables so the remaining attributes are just containing repeated information. Another interesting approach refers to "all-relevant" features that involves finding all the strongly and weakly relevant variables. These weakly relevant refers to the ones that might or might not be crucial to keep depending on which other features are kept [17, pg.249]. Under these definitions, [19] proved that the "all-relevant" approach can only be fully achieved with small sets of data and could be addressed through algorithms like decision trees and best subset selection. On the other hand, "relevant" can be understood from a more natural statistical approach [17, pg.250] that refers to a measure of complexity that does not end with a subset of variables. This notation becomes pertinent when talking about dimensionality reduction algorithms like principal components.

According to [20], with the increasing dimensionality of data, the discovery of the relevant features has acquired great importance and has become essential to solve real problems. Several methods have been developed; in consequence, it is not an easy task to determine the most effective one for all cases. This is why a profound test of methods is needed to compare their effectiveness in a determined situation.

In literature, one can find numerous different classifications of feature selection methods [21, pg.204][22, pg.183][23, pg.484] [24] [16, pg.3][16, pg.362] [17, pg.246]. For this work, and considering the type of problem, we propose a scheme of categorization of feature selection methods and possible approaches that can be found in the **Annex 7.3**. The first approach, considered in this paper, contemplates the independence of the observations without taking into account the time series components.

Additionally, feature selection can be done considering a prediction for time series through feature engineering. We could start with a construction of date related features for identifying the day, month, year and unconventional days that mark specific behaviours,

followed by the creation of variables to express time delays among effects of variables and rolling or expanding windows for creating the training subsets. Moreover, Multivariate Bayesian Structural Time Series Model (MBSTS) can be a viable option considering that it performs model training and variable selection simultaneously, preventing overfitting, and has proven better accuracy performance than other methods like Bayesian Structural Time Series and traditional models like ARIMA [25]. In addition, feature selection methods could be addressed jointly with Neural Networks for time series prediction creating a combine approach as pointed by [26].

Another path could be optimization which can be used to address the “all-relevant” approach developing a process to search the best subset of features through Mixed Integer Optimization [27].

As stated by [28], the feature selection process, in general, will present two components: first, the selection algorithm that will imply the definition of the optimal set of variables or features, and second, the evaluation algorithm that will define the "goodness" of the performance of the subset in the algorithm.

Moreover, the theoretical meaning and statistical significance of the relationships among the "non-modellable" factor and the selected "modellable" risk factors acquire less importance, being the primary goal the predictive accuracy of the model derived by the associations between predictors and the outcome variable [29][30]. This association is related to the correlation among them, meaning how the values of the "non-modellable" factors co-occur with particular values of the "modellable" ones [31]. From this point of view, the model involves a predictive approach [32] with a clear bias towards the power of predictivity over interpretability; then, the accuracy in the prediction will be related to the "goodness" of the feature selection method.

In consonance with [23] [33, pg.1158] [34, pg.25] [35], feature selection can lead to many potential benefits like enhancing the inductive learner by increasing learning speed, generalization capacity or simplicity of the induced model, decreasing the measurement and storage requirements, and facilitating visualization and interpretation. As will be seen, the three first benefits mentioned are primordial for solving the problem and generating good performance models capable of minimizing the error rate.

3.2. Dataset

The data used to present this problem contains zero-coupon interest rates risk factors of a trading desk, including liquid currencies, domestic currency, and other currencies. Additionally, documentation about the methodology of the construction of the curves, a modellability map, and information for the granularity considerations have been handled by the entity, which is collaborating with this study.

The processed datasets are:

1. Dataset of 23‘918,848 observations that are daily values of risk factors, containing the specifications of the date, the name of the curve, the currency, the tenor expressed in days, the variation, and its type.

2. A dataset of the modellability map which contains the 1609 risk factors with the following characteristics: granularity in terms of periods, correspondent currency, and observed category among "modellable" and “non-modellable" defined by the compliance of the criteria previously described.

3. Tenor equivalence in terms of days obtained from information received about the granularity considered by the entity.

Behind the data of risk factors, there has been a methodology developed by the bank to convert Market Quotes that contain dependencies to other curves into zero-coupon instruments under the regulatory framework of the IMA, in concordance to the FRTB.

The data presents 113 curves, 3,414 dates, 15 different currencies, tenors that go from 1 day to 36,526 days, and considers three types of variations: additive, DHS (Hilbert-Schmidt distance), and multiplicative. Bear in mind that in this case, a risk factor will be understood as a particular tenor of a curve; under this consideration, the initial number of risk factors is 7,006.

Intending to simplify the variation data, it has been calculated as the difference between the observation of the risk factor at time $\mathbf{t}$ and the observation at time $\mathbf{t-1}$, replacing the variations given by the other methodologies. Then, the total remaining daily observations for each risk factor is 3,413.

Moreover, a distinction of the curves has been made based on the construction methodology. Considering this scope, the distinction shows 27 Single Currency curves, 78 Cross-Currency curves, and 8 Fx Ratio curves.

Fx Ratio curves correspond to instruments that are not observable in the market and which are entirely constructed based on other curves, becoming synthetic curves. These curves have been removed from the analysis, considering that the tenors of their dependent input curves are treated as "modellable" or transformed to "modellable" using proxies, consequently, their construction will be carried out on the fly. With respect to Cross-Currency curves, it is important to notice that these curves are treated as spread curves over another curve (e.g. the cross currency curve represents the difference with respect to another curve that is build using Single Currency methodologies). This distinction of the type of curves has been manually elaborated because no information was provided.

Other considerations have been taken into account; for example, all the risk factors not found in the modellability map have been deemed as "non-modellable" instruments. Additionally, buckets for which a future is passing through will be considered as "modellable". Furthermore, risk factors have been categorized based on the term, separating long term factors as those with tenors higher than two years, considering 365 days

as a measure of year, and the remaining as short-term factors. After these particulars, the remaining risk factors were 895 "modellable" and 5,615 deemed as "non-modellable".

Bearing in mind that the curves are constructed with their dependencies to other curves, there is the case where it could exist perfect correlations that can show that one risk factor moves in the same amount than other, and, if the sign is positive implicating that are also moving in the same direction[36]. In consequence, if a NMRF is perfectly correlated to one or more NMRFs, by capturing the MRFs that suit just one of them, we can extend the scope and replicate the solution for the rest. Likewise, if a MRF is perfectly correlated to other MRFs, it is not necessary to take into account all because just one of them will expose the behavior and effect of all, preventing overfitting. To identify the groups and the connections, a neural network was performed over both groups of risk factors, and only one random risk factor of each has been extracted. After this feature diminishing process, the remaining risk factors that will be analyzed are 4,260 "non-modellable" and 879 "modellable" series.

3.3. Assumptions and considerations

The risk factors considered are built under proven methodologies that have passed a validation process and which aim to generate a continuous series of curves and tenors. Consequently, there will not be missing values, and it can not be considered that the observations that present a distorted behavior from the average are outliers. These previously treated considerations motivate the omission of an imputation process or removal of missing values and the calculation of outliers. Additionally, there is not prior knowledge about feature importance nor the number of variables that define a correct solution.

Moreover, since the selection of variables to express NMRFs in terms of MRFs will be a continuous need, and the amount of data is expected to grow over time, changes will occur on the modellability map presented. Consequently, the methodology has to be constructed to involve automatization, reproducibility, scalability and traceability, which means that there is a need for a stable and general approach.

This study considers addressing this problem without taking into account the temporal dimension. We assume that there is no influence of the time basing the idea in the fact that a data forecast will not be required and that the main purpose is to express the "non-modellable" risk factors in terms of "modellable" ones.

3.4. Partition

To choose among the different methods, a validation set and cross-validation approach has been followed; this implies that a splitting of the observations is imperative to achieve an accurate estimate of the test error [21, pg.248]. Then, the training partition will be used to perform the fitting of the model, and the performance will be measured through the test

partition.

Regulation requirements imply testing of the proxy in the last 12 months to measure the actual performance of the method. Considering that the dataset contains 3,413 observations and that these 12 months represent only the 7.58% for testing, two different partitions are proposed: the first one used to determine the methods that capture the behavior of the 80% of the data for training and 20% for testing, this means that the 2,413 most distant days from the evaluation cut date are considered to develop the method and the rest to test the performance of it, and a second one will be used to reconfirm the performance of the methods by partitioning the data with a 92.42% for training and the remaining 7.58% or 259 days to test the methodology picked and return the error derived by the difference among the predicted variations and the real changes of the NMRFs, and to present the performance measure under the regulatory framework.

3.5. Notation

The notation that will be used all along this paper, is described in **Table 3.1**. The general scenario can be express as: $Y_{ki} = \beta_{0i} + \mathbf{X}_{ks}\beta_{\mathbf{i}} + \epsilon_i$ where the estimation can be view as: $\hat{Y_{ki}} = \hat{\beta_{0i}} + \mathbf{X}_{ks}\hat{\beta_{\mathbf{i}}}$ for $i \in \{1, \ldots, q\}$ and for $k \in \{1, \ldots, n\}$

The main objective will be to present a methodology that it is able to minimize $\epsilon_i = \sum_{k=1}^{n}(Y_{ki} - \hat{Y}_{ki})$ for $i \in \{1, \ldots, q\}$, and that through it more NMRFs can fulfill the criterion of the regulator [1] so they can be capitalized under the Internal Model Approach (IMA) avoiding an increasing in capital requirements.

3.6. Preprocess

Based on the workflow to develop feature selection proposed by [33, pg.1159], a preprocess treatment of the input data can be defined; this will consider a standardization, a Yeo-Johnson transformation, and a removal of dependencies. Standardization is required considering that the measurements of the variables differ, and the algorithms could be influenced by features with higher absolute scales. The standardization can be done using the methods "scale" and "center" in the `preprocess` function of the package *caret*, which is explained in the next section. The scale involves the calculation of the standard deviation of each feature and the division of the correspondent observations of each variable by this value. The centering comprises the calculation of the mean value of each feature and its subtraction of each of the observations. After both methods, the standardization will imply that every serie of a MRF will have a mean zero and a standard deviation of one.

Yeo-Jhonson [37] will allow to reduce skewness and to approximate to normality; this method proposes power transformations keeping the good properties of the Box-Cox

[1] This is presented in Section 3.6 Performance measures. For more detail refer to Minimum capital requirements for market risk released on January 2019 (rev. February 2019) [4]

Table 3.1. NOTATION

Notation	Description
n	Number of daily variations for each risk factor. These will be partitioned into trainning (ntrain) and test observations (ntest).
q	Total number of non-modellable risk factors
p	Total number of modellable risk factors
s	Number of modellable risk factors in a subset selected
S	Identifier of variables selected for subset
Y_i	Non-modellable risk factor, where $i \in \{1, \cdots, q\}$ and $Y_i \in \mathbb{R}^n$
$\mathbf{Y}$	Matrix of dimensions n rows and q columns. $\mathbf{Y} \in \mathbb{R}^{n \times q}$
Y_{ki}	Observation of a non-modellable risk factor where $i \in \{1, \cdots, q\}$ and $k \in \{1, \cdots, n\}$
$\mathbf{X}$	Matrix of dimensions n rows and p columns. $\mathbf{X} \in \mathbb{R}^{n \times p}$
X_j	Modellable risk factor, vector of n observations, where $j \in \{1, \cdots, p\}$ and $X_j \in \mathbb{R}^n$
$\mathbf{X}_s$	Matrix of dimensions n rows and s columns. $\mathbf{X} \in \mathbb{R}^{n \times s}$, where $s \leq p$
X_{kj}	Observation of a modellable risk factor where $j \in \{1, \cdots, p\}$ and $k \in \{1, \cdots, n\}$
ϵ	Error derived by the difference between the real variations and the predicted variations of the methods

transformations but allowing negative values for the input [38]. While the removal of the dependencies is necessary to remove the set of columns or variables that are linear combinations, this can be achieved with the function `findLinearCombos`, which uses a QR decomposition of the matrix to check if it is full rank and identifies the features that are part of the dependencies [39]. **Annex 7.4** presents a diagram with the essential steps described up to this section.

3.7. Algorithms

Considering that all the algorithms cannot be tested due to the vast amount of them, a background knowledge is needed to pick the supposed best ones for this particular problem. For that, a sample of algorithms of each class has been chosen based on previous tests with some random "non-modellable" variables, and its popularity, strengths, and limitations provided by literature. Those models have been tested in a larger sample of 100 "non-modellable" variables to determine the best algorithms that have been implemented in the whole set of NMRFs (4,260).

The main requirements for all the algorithms are the matrix $\mathbf{X}$ of the "modellable" series and the response variable, which correspond to a "non-modellable" risk factor Y_i where $i \in \{1, \cdots, q\}$. The training partition will be used to all the constructions of the

models and the testing observations for the prediction and posterior measure of the accuracy performance.

Some considerations have been taken into account concerning the programmed codes in R for the methodology [2].

In the first place, there is a partition of the codes based on the phases of the process; as explained before, there are three data sets that needed to be joined, involving tasks for unifying the information like checking the curves and separating the MRFs from the NMRFs, converting the tenors, creating identification variables, calculating the variations of the rates, adding more categories, creating the neural networks and distinguish the perfectly correlated factors, and changing format aspects of the dataset for them to be congruent. This is the first code that generates a new set of data frames that are saved in **datamods1.RData** which includes:

- Two data sets call "mod_n" and "nomod_n": a complete modellability map for MRFs and NMRFs which contains the name of the curve (curve_name), currency, an identification variable corresponding to the group equal to 1 if "modellable" and 0 if deemed as "non-modellable" (mod_code), tenor measure in days (tenor), type of curve based in the construction method applied (type), the term of the curve (term), and an ID variable constructed by joining the curve name, the tenor, the currency and the term (fr).

- Two more data sets which contain the daily observations for each risk factor. The one called “data_mods” will include the MRFs, which in this particular case presents 879 columns plus the first one corresponding to the date, and the “data_nomods” will contain the information of the NMRFs, which contains 4260 columns and, the same as before, one additional column for the date. The identification variable with the same structure as in 1 has also been set for both data sets.

Additionally, a second object is saved as **components.RData**, which contains the maps of groups perfectly correlated so that the results obtained could be expand to the whole set of "non-modellable" risk factors after the methodology is picked.

This script is expected to vary depending on the receiving information considering that there have been specific tasks developed based on documents of methodologies performed by the entity and expert criteria, which can vary and, consequently, adjustments will have to be made.

The second script will receive as input the four data sets **datamods1.RData** generated in the previous step and will comprise the preprocessing of the data and the development of diverse algorithms. This code has been created in a way that could be replicable no matter what is the number of rows or columns of the data; then it makes no distinction

[2] All the scripts detailed can be found in https://github.com/Rafaelab1227/NM-UNDER-FRTB.git

as to the number of "modellable" and "non-modellable" factors that will be analyzed, responding to the need of a general approach.

Other considerations that have been made are the use of lists that allow checking results. These lists include for every NMRF: the coefficients of the resulting selected variables, the tunned parameters, the predictions, and the corresponding performance for each model, responding to the requirement of traceability. Moreover, the script defines a seed for the random numbers generated in response to the reproducibility principle. After running the script, the results will show:the time elapsing for each method, the metrics of the performance for each model for every NMRF, and the list of all NMRFs with the corresponding results for each tested algorithm.

The exploration of the methods has been performed using only one computer to be able to generate a complete comparison among methods. It is important to note that the technical characteristics of the computer will affect the performance of the scripts [3]. As mentioned before, considering the computational limitations that have not allowed an exploration of all the methods, the initial approach considers a random sample of 100 NMRFs for training and testing the various algorithms. Then, the best methods will be tested on the 4,260 NMRFs; this subset of methods is presented in two different versions of the second script. The second version, contemplates the development of six methods with a 80% partition for training and 20% for testing, while the third version takes the same methods for a 92.42% for training and 7.58% for testing.

Furthermore, parallelization has been used in the critical points which lay in the running phase of the models for the three versions, and a second parallelization over the iterations in the second and third version to guaranty scalability. Parallelization can be understood as the simultaneous execution of large computations divided across multiple cores, with the aim to achieve a speed-up on the computational time [40]. Additionally, to improve the performance of the algorithms when a lot of features are contemplated, it is essential not to use the formula interface [41]; this is why this type of formulation is used only after the feature selection has been done on the algorithms.

The third script contains a program that receives as input the results from the previous step and performs a display of the rank of the metrics and performs some graphical outputs that simplify the visualization of the performance. There are two versions of it, one for the sample of 100 NMRFs and the other for the whole set of NMRFs.

Finally, the fourth script presents the calculation of the criterion for the regulator to create the new modellability map and determine which NMRF can be expressed as MRF. Also, includes some graphical outputs and the results of the methodology picked for each NMRF.

[3]Machine: "Intel(R) Core(TM) i3-6100U CPU @ 2.30GHz". Num cores:2, RAM:8.48 GB, R Version: 3.6.3. Through the package *benchmarkme* the computer characteristics were compare to 749 computers of other users as a benchmark, and it was located in the number 646, showing a performance below average in time consumption.

A more detail explanation of the structure of inputs and outputs of each script can be seen in **Annex 7.5**

Next, we will present the considered methods and the corresponding packages and requirements for the calculations. A broad division of feature selection methods has been used, dividing them into two major categories: variable selection and dimensionality reduction. In turn, the first category presents a division of the algorithms into three classes: filter, wrapper, and embedded. The main difference among these methods is that filter methods select features without optimizing the performance of the predictor. In contrast, wrapper methods utilize a learning machine or "black box" to categorize the variables based on their predictive power, and embedded methods involve a perturbation of the data [42, pg.5]. On the other side, dimensionality reduction methods are based on projecting the predictors into an M-dimensional subspace where the dimension is less than the number of predictors [21, pg.204].

Almost all the methods have been developed using the classification and regression training package *caret* [43] which provides a uniform interface to perform various algorithms in R. At the moment this document is written, the package presents 238 available models. The ones used will be described in detail in the section below, additionally in **Annex 7.8** it can be found a complete list of all the packages references. Moreover, the main functions used are `train` from *caret* and the generic function of `predict`.

According to the documentation [44], `train` can be used to find the best parameters, models and determine the performance of the training set (see Algorithm 1). In order to execute this function, there has to be a definition of the parameters required by the method that will be tuned; if not defined, they will act as a default. Also, a grid could be used with the function `expand.grid` [45], implemented in base R. Additionally, a resampling method could be defined using the function `trainControl`. In this case, it has been used as a non-repeated k-fold cross-validation method to find the value of the model hyperparameters. Moreover, there has to be a definition of the response and the input variables, and a call of the model that will be performed.

Algorithm 1: Train function

Result: Obtain the optimal hyperparameters and fit the model for the training partition

```
initialization;
for each parameter do
    for each resampling iteration do
        Pick specific samples;
        Fit the model on the rest;
        Predict with the samples;
    end
    Calculate the average performance;
end
```

After the model was fitted, the performance is measured based on the testing partition; this is achieved with the function `predict`.

3.7.1. Variable selection

Hybrid methods

Filter methods are considered faster than wrapper methods, in consequence, these are practical for higher dimensional data [34, pg.32]. However, advantages can be achieved by the mixture of both. These hybrid methods are said to be better than just considering a filter selector [46, pg.148][47, pg.2]. Even Hall in early 1999's thought that a reasonable approach could be a bidirectional search that adds and removes features more appropriately than either one of the methods, and that a computational expense could be reduced by a filter selector that has a starting subset [46, pg.148]. This is a scalable method commonly used for biological sequence data which uses domain knowledge to construct features, reviews the space of possible features, and identifies the most useful set of variables [16, pg.14, 365]. Then, with the set of features constructed, a feature selection is performed based on a wrapper method; this generation process allows a more space-efficient search. With these in consideration, the following methods, except Model.2, have been built using different filters to search for redundant information followed by a stepwise bidirectional search that considers both forward and backward selection used by the caret method "leapseq" [34, pg.29] which refers to the algorithm "seqrep" of the `regsubsets` from the package *leaps*[48].

"leapseq" performs a sequential replacement procedure. According to [49], this is a "cheap" method when it is not possible to do a profound and exhaustive search. It is based on adding predictors and replacing each one with the remaining part of those that are not in the subset in question if the residual sum of squares (RSS) is reduced until no replacements are possible. These subsets are defined to reach a maximum number of predictors.

Model 1. Correlation filter

The first method to be tested has been variable ranking by correlation [33, pg.1160][46][34] applying the function `findCorrelation` from the package *caret* to reduce the number of variables taking into consideration 0.75 as a cutoff for the allowed absolute pairwise correlation given that by expert criteria a correlation above 0.75 is consider high. In each step, the function will determine the variable with the highest mean absolute correlation coefficient concerning the rest of the variables, and this will be removed. In consequence, by taking into account the high correlation filter, the result will be a set of features where redundant and noisy features have been removed [46].

Procedure:

1. Filter of the matrix based on the pairwise correlation.

Table 3.2. REQUIREMENTS MODEL 1. CORRELATION FILTER

Package	Function	Parameter	Description	Value
caret	find Correlation	cutoff	Threashold for the pair-wise absolute correlation	0,75
	train	trControl	10-fold cross-validation	method = "cv", number=10
		tuneGrid	Maximum number of variables for the model (25)	nvmax = seq(1, 25, 1)
		method	definition of the algorithm	leapseq

2. Using the method “leapseq”, provided by `train` with 10-fold cross-validation without repetitions, to tune the hyperparameter “nvmax” which considers the maximum number of predictors to examine, in this case, set to 25 to improve computational time.

3. Fit the model and predict with the testing partition.

Model 2. Correlation-based Selection

Another approach used has been the Correlation-based Feature Selection or CFS referred by [47], but considering that the `findCorrelation` function gives a fast implementation of the pairwise correlation calculation, the whole approach has been developed. In consequence, by calculating: first, the pair-wise correlation among independent features with the same considerations as before, and second, the correlation among each "non-modellable" variable and the features considering a threshold of 50% , the results will show the “good” variables which are the ones catalogued as relevant and not redundant to any of the other relevant features.

Procedure:

1. Filtering of the matrix based on the pairwise correlation.

2. Find the correlation between the "non-modellable" factor and each "modellable", eliminate all which correlation is less than 0.5.

3. Perform a linear regression and predict algorithm with testing partition.

Additionally, other filters without considering statistical methods have been tested, given the fact that a pre-screen of the predictors could help to obtain a better subset of features. For that, based on domain knowledge can be said that curves that share similar characteristics might have similar behavior, and one could be explained by the others. For the filter step, it has been considered the distinction of:

Model 3: the type of curve considering the classes Single Currency Curve and Cross Currency curves.

Model 4: the currency of the risk factor based on its curve.

Model 5: curve to which the risk factor corresponds, bearing in mind that there might be curves that present only one tenor as "non-modellable", and the rest of tenors remain categorized as "modellable".

Model 6: term, by classifying the risk factors based on the long term (more than two years) and short term (less than two years).

After the construction of the feature set, the most relevant features have been identified by the same wrapper stepwise method presented in **Model.1**. Additionally, considering that the resulting subset returns a smaller quantity of predictor, the hyperparameter "nvmax" has not a maximum value, and a 10-fold cross-validation resampling is performed for the correspondent tuning.

Wrapper methods

Model 7. Stepwise

Feature Wrappers are said to achieve better results than filters because they can be adjusted to a particular interaction between the algorithm and the training data [34, pg.41]. In filter methods, the selection process is independent of the learning algorithm, while wrapper methods will iteratively use the learning performance to assess the quality of selected features in each step [35, pg.3], in consequence, and as exposed before, they will involve a high computational cost in large datasets.

In R, a good alternative is a stepwise procedure with the *bigstep* package, which will allow us to work with large datasets by converting them to a class *big* [50]. The package presents several functionalities that have been used. To begin the algorithm, the data is converted by `prepare_data`, next the `reduce_matrix` function has been used to calculate p-values corresponding to the Pearson correlation test among the "non-modellable" factor and every "modellable" factor. Then, a cutoff of 0.01 has been set to keep only the variables with higher p-values, there were cases where the cutoff was too low and all the variables were removed, for those the cutoff considered was 0.05. After this step, performed to reduce the complexity of the stepwise, a `fast_forward` function is done to add variables that reduce a specific criterion, followed by a `Multi-backward` step that removes variables based on the effect in the reduction of another define criterion, and finally, a `stepwise` function is introduced to the pipeline to do a bidirectional search of the variables that improve the third criterion. By this method, all the variables are taken into account, and given the reduction of the time performance achieved, it can be an excellent approximation to the best subset algorithm.

To choose the appropriate criteria, a realization of this method was done for each possible combination of the criterion 'mbic' and 'maic' [51], which correspond to a Bayesian modification of the original AIC and BIC with a higher penalty for the number of variables included [50]. After obtaining the results for all the combinations, the best one will

be the one which p-value for the F-test is lower than 0.05; if there is more than one option, then we chose the one that maximizes the adjusted R-squared; this measure is considered given that adding irrelevant features can cause an increase on the R-squared measure [52, pg.228]. If one or more combinations result in the same measure, the one where all variables are significant, taking into account a p-value of 0.05, will be the best combination. Given the case where there is not a final model with all its variables categorized as significant, the combination of the model which had the biggest number of significant variables has been considered. The procedure can be seen in detail in **Algorithm 2**.

Table 3.3. REQUIREMENTS MODEL 7. STEPWISE

Package	Function	Description	Value
bigstep	reduce_matrix	minpv value, p-value cutoff	0.01 or 0.05
	fast_forward	model selection criterion	mbic/maic
	multi_backward	model selection criterion	mbic/maic
	stepwise	model selection criterion	mbic/maic

Embedded methods

Shrinkage or embedded methods will make use of the intrinsic structure of a learning algorithm to incorporate feature selection into the underlying model [35, pg.3]; in this case, depending on the type of shrinkage performed, the selected methods were: Ridge Regression, Lasso, Elastic Net, and Reluctant Generalized Additive Models.

Model 8. Ridge regression

The Tikhonov–Phillips regularization [53] or Ridge regression will shrink the coefficients of correlated variables towards each other [54, pg.3]. The main objective with the algorithm will be to get for each NMRF the $\hat{\beta}^{\text{ridge}}$ that will minimize:

$$\sum_{k=1}^{n}\left(y_k - \beta_0 - \sum_{j=1}^{p}\beta_j x_{kj}\right)^2 + \lambda\sum_{j=1}^{p}\left|\beta_j^2\right| = \text{RSS} + \lambda\sum_{j=1}^{P}\left|\beta_j^2\right| \tag{3.1}$$

Then, if λ is equal to zero, the ridge estimator is equivalent to the Ordinary Least Squares estimator and as it grows the coefficients will shrink towards zero [55, pg.280]. In consequence, Ridge regression can be a good method to reduce multicollinearity that could be originated due to the fact that variables can be presented as interactions of an original set of variables [56].

Algorithm 2: Model 7 algorithm

```
Result: model with best combination criterion and predict
for each "non-modellable" risk factor do
    for each possible combination of criterion 1, criterion 2, criterion 3 of
      classes "maic" and "mbic" do
        Convertion of Y_i, Y_i ∈ ℝ^n and X;
        Reduction pf variables based in correlation; Add variables as long as they
          reduce criterion 1;
        Remove worst variable as long as they reduce criterion 2;
        Stepwise with criterion 3;
    end
    Results: 'Combination', 'R2', 'ADJR2','NVAR'(If all significant 1),
      'NVAR.P0.05' (Number of significant variables), 'P.VALUE.F'(F-test);
    for each possible result based on the combination of criteria do
        Being Ho: the fit of the model only with the intercept is the same as the
          current model and H1: the fit of the intercept model is significantly
          smaller than the current model, H_0 : β_2 = β_3 = ··· = β_s = 0 H_1 : H_0 not
          true And, F = (R^2/s) / ((1−R^2)/(n−s));
        if F bigger than threshold reject H_0 in favor of H_1 for only one
          combination then
            Combination is picked
        end
        else if in more than 1 combination F bigger than threshold then
            Check for the highest adjusted R2 R^2_adj = 1 − [((1−R^2)(n−1))/(n−s−1)];
            if more than one with same adjusted R2 then
                Check the significance of all the variables with threshold;
                if more than one then
                    Any combination is picked
                end
                else if not all are significant then
                    Check maximum number of significant variables
                end
                else
                    Combination is picked
                end
            end
            else
                Combination is picked
            end
        end
        else
            Any combination is picked
        end
    end
```

Table 3.4. REQUIREMENTS MODEL 8. RIDGE REGRESSION

Packag	Functic	Parameter	Description	Value
caret	train	trControl	10-fold cross-validation	method = "cv", number = 10
		tuneGrid	parameter of regularization λ	10^seq(-3, 3, length = 100)
			Elastic Net mixing parameter	0
		method	definition of the algorithm	glmnet

Procedure:

1. Using the `glmnet` [54][57] method for gradient descent by defining the elastic mixing parameter $\alpha = 0$ and possible λ grid for the penalty parameter, the minimization is achieved defining the optimal λ by fitting the model with a 10-fold cross-validation resampling.

2. With the partition for testing, the predict function is used to generate the estimated values for the "non-modellable" factor and are compared to the real variations to generate the performance measurements.

Model 9. Lasso

Lasso [58][59] is a method that performs both shrinkage, like ridge regression, and also a selection of features being the final goal to get the estimates $\hat{\beta}$, to minimize for each "non-modellable" risk factor:

$$\sum_{k=1}^{n}\left(y_k - \beta_0 - \sum_{j=1}^{p}\beta_j x_{kj}\right)^2 + \lambda\sum_{j=1}^{p}|\beta_j| = \text{RSS} + \lambda\sum_{j=1}^{P}|\beta_j| \tag{3.2}$$

Table 3.5. REQUIREMENTS MODEL 9. LASSO

Packag	Functic	Parameter	Description	Value
caret	train	trControl	10-fold cross-validation	method = "cv", number = 10
		tuneGrid	parameter of regularization λ	10^seq(-3, 3, length = 100)
			Elastic Net mixing parameter	1
		method	definition of the algorithm	glmnet

Procedure:

1. Through the `glmnet` [54] [57] method implemented in `caret` package, which uses the setting of the elastic net problem with an α equal to one, the minimization

problem described is solved. The resulting penalty corresponds to a Laplace prior that is expected to generate several coefficients close to zero and a few to be larger and different than zero [54, pg.3].

2. With the partition defined for the prediction, the function `predict` is passed, and the performance is measured.

Model 10. Elastic net or Enet

The Elastic net [60] acts as an alternative that tends to overcome lasso results. Lasso is indifferent to correlation among predictors [54, pg.3]; it can be derived into a limitation in some cases because it tends to extract just one variable of a group of correlated features. Even Tibshirani [58] empirically observed that the performance of the prediction is better with ridge regression if the pairwise correlation of the features is high. In contrast to lasso, Elastic Net can select a group of correlated variables, enhancing the prediction performance of the resulting models [60, pg.302].

The *glmnet* package presents an algorithm built under a focus on coordinate descent methods to implement the elastic net and, as seen in the previous methods, lasso and ridge regressions as particular cases of it. This perspective involves taking into account a sequence of values for λ and start with the smallest to get the estimates $\hat{\beta}$ to minimize for each "non-modellable" risk factor:

$$\frac{1}{2n}\sum_{k=1}^{n}(y_k - \beta_0 - X_k^{\top}\beta)^2 + \lambda P_\alpha(\beta) \tag{3.3}$$

$$\begin{aligned} P_\alpha(\beta) &= (1-\alpha)\frac{1}{2}||\beta||_2^2 + \alpha||\beta||_{\ell_1} \\ &= \sum_{j=1}^{P}\left[\frac{1}{2}(1-\alpha)\beta_j^2 + \alpha\left|\beta_j\right|\right] \end{aligned} \tag{3.4}$$

Where $0 \leq \alpha \leq 1$

This penalty is called the naive elastic net and has been shown that it can be useful when the number of predictors is larger than the number of observations, or when variables are correlated [54].

Procedure:

1. The train control for the election of the hyperparameters is done with 10-fold cross-validation; this means that the training set will be partition ten times, then *caret* will try ten models of optimal combinations for the parameters α and λ, and based on the RMSE there will be a retrieved of the optimal model.

2. Prediction and measurement of the performance with the testing data.

Table 3.6. REQUIREMENTS MODEL 10. ELASTIC NET

Packag	Functic	Parameter	Description	Value
caret	train	trControl	10-fold cross-validation	method = "cv", number = 10
		tuneGrid	parameter of regularization λ	program review 100 values
			Elastic Net mixing parameter	program review
		method	definition of the algorithm	glmnet

Model 11. Reluctant Generalized Additive Models

The Reluctant Generalized Additive Models or RGAM [61][62] consist of an extension of a sparse generalized linear model that enables the possibility of non-linear effects with the response variable allowing the construction of models with higher accuracies. The main advantage of RGAM models is that it implies a preference for linear effects, then fewer complex models are built [61]. It is possible a k-fold cross-validation, with a minimum number of 3 folds, in this case, a 5-fold CV has been picked to choose the lambda associated with the smallest mean squared error. Other parameters that have been set as default are the value of gamma, which is the scale factor for non-linear features defined as 0.6, avoiding models not to present non-linearities, according to [62]; the number of non-linear features described as all possible "modellable" risk factors; and for achieving a parallelization of the process, a change was made to the original function due to a package that was called which was not available in the version of R used in the present work. This changed function can be found in the GitHub repository with the scripts. [4].

Procedure:

1. Lasso of each "non-modellable" risk factor is performed on $\mathbf{X}$ to get the estimates $\hat{\beta}$; then, the error is calculated considering the λ picked in the CV process.

2. For every "modellable" risk factor considered, a smoothing spline is fitted. The new feature is expressed by $\hat{f}_j$. This will be rescaled considering the hyperparameter γ; in consequence, the mean standard deviation of each new feature over the mean standard deviation of X_j will be equal to γ.

3. Lasso of each "non-modellable" risk factor is performed on $\mathbf{X}$ and the resulting matrix of step 2 containing all the new features with the non-linear effects taking into account the sequence of the parameter λ.

4. The last three steps are performed by the function `relgam` in R. The prediction for the testing observations is developed with the use of `predict.cv.rgam` considering the penalization parameter associated to the minimum MSE.

[4] https://github.com/Rafaelab1227/NM-UNDER-FRTB.git

Table 3.7. REQUIREMENTS MODEL 11. RELUCTANT GENERALIZED ADDITIVE MODEL

Package	Function	Parameter	Description	Value
relgam	cv.rgam	λ	penalization parammeter $\lambda_1 > \cdots > \lambda_r \geq 0$,	sequence generated by the program
		γ	Scaling hyperparameter $0 \leq \gamma \leq 1$	0,6
		nfolds	Number of folds that will be performed in Cross-validation procedure	5
		parallel	parallelization option	TRUE
		init_nz	Modellable risk factors that are selected for constructing its correspondant non-linear features.	All
	predict.cv.rgam	s	Parameter pick for the penalty	lambda.min

3.7.2. Dimensionality reduction

The second category of algorithms is motivated by the reduction of a high dimensionality by the transformation of the predictors; this becomes handy considering that a projection of the data into a lower-dimensional space could be made while retaining most of the information [42, pg.3] [21, pg.229].

The general idea behind the next algorithms is to transform the predictors contained in **X** by generating:

$$Z_m = \sum_{j=1}^{p} \phi_{jm} X_j \tag{3.5}$$

Being:$\phi_{1m}, \phi_{2m} \ldots, \phi_{pm}$ constants where $m = 1, \ldots, M$ are the components,

Then, the initial regression problem for each "non-modellable" factor can be expressed as:

$$Y_k = \theta_0 + \sum_{m=1}^{M} \theta_m Z_{km} + \epsilon_k, \quad k = 1, \ldots, n \tag{3.6}$$

Model 12. PCR

The use of Principal Components in Regression was presented in the 50s and 60s by several authors who suggested that by replacing the predictors with its corresponding principal components, there will be a gaining in stability and computational time [63][64][65]. Since then, significant advances and new implementations can be found as well as a vast amount of theoretical documentation, converting it in a prevalent approach. However, several works illustrate how PCR is not the best method, criticizing its linearity, and the

way it diminishes the accuracy for new data [66]. Even in some simple examples, there can be seen that other methods of the wrapper feature selection category perform better [21, pg.235].

The approach considers an unsupervised method since the values of the "non-modellable" risk factor are not contemplated for the calculation of the linear combinations of the features corresponding to the "modellable" risk factors. A supervised approach could be explored considering the work of [67].

Table 3.8. REQUIREMENTS MODEL 12. PCR

Package	Function	Parameter	Description	Value
caret	train	trControl	10-fold cross-validation	method = "cv", number = 10
	train	tuneLength	Maximum of components	20
caret/pls	train	method	definition of the algorithm	pcr

Procedure:

1. The algorithm will use the `train` function in the *caret* package with a 10-fold cross-validation and an exploration of 1 to 20 for each resampling to find the number of principal components or 'ncomp' that will remain.

2. The method will find the $Z_1, \ldots, Z_M$ principal components, and next, it will fit a linear regression using least squares with 'ncomp' as predictors. If PCR holds, then the number of principal components will be lower than the number of original predictors, avoiding overfitting.

3. With the testing partition, the model is fitted, and the performance measurements are retrieved. The cumulative percentage of variance explained by the principal components is kept in the results list.

Model 13. PLS

Partial Least squares or PLS [68] is a well known method that overcomes the performance of PCR and is presented as a supervised alternative of a dimensional reduction method. This algorithm differs from PCR in the fact that it will consider the response variable, additionally to the predictors, for creating the new features based on linear combinations.

Procedure:

1. The idea behind the algorithm is that it will start with the setting of $\phi_{1m}, \phi_{2m} \ldots, \phi_{pm}$ as the coefficients of the simple regression of the "non-modellable" risk factors on to

Table 3.9. REQUIREMENTS MODEL 13. PLS

Package	Function	Parameter	Description	Value
caret	train	trControl	10-fold cross-validation	method = "cv", number = 10
	train	tuneLength	Maximum of components	20
caret/pls	train	method	definition of the algorithm	pls

X. The $Z_1, \ldots, Z_M$, will assign higher weights to the most correlated variables. This fitting of the model is done ten times considering the CV method for resampling, and in each iteration proves 1 to 20 principal components.

2. The final model is used to predict the values of the "non-modellable" factors, and the accuracy measures are retrieved.

Model 14. ICR

Independent Components Regression (ICR) is an analogue method to PCR that has its origins in the Independent Component Analysis. The main difference between ICR and PCR is that it is based on the assumption that the components or latent variables are statistically independent, meaning that the value of one of them will not give any information about the rest. Recall that uncorrelatedness is a weaker form of independence, in consequence, ICR can make use of higher-order statistical signals that are said to better capture the essential structure of the data [69][70][71].

The ICR has been implemented using the `fastICA` function of the package with the same name that will decompose the data into non-gaussian, statistically independent or as independent as possible variables, presenting a fast converge considering that in contrast to the original ICA based on gradient descent methods, the convergence can be cubic or at least quadratic and not only linear [72][73].

Table 3.10. REQUIREMENTS MODEL 14. ICR

Package	Function	Parameter	Description	Value
caret	train	trControl	10-fold cross-validation	method="cv", number=10
	train	tuneLength	Maximum of components	10
caret/fastICA	train	method	definition of the algorithm	icr

Procedure:

1. The implementation of the algorithm can be done through the package *caret* with the specification of the method “icr.” A 10-fold cross-validation procedure has been

followed to tune the hyperparameter correspondent to the number of components that have been set as a maximum of 10.

2. With the fitted model, the function `predict` is passed to test the performance of the model.

Model 15. SPLS

Sparse Partial Least Squares (SPLS) [74][75] is a relatively recent approach which takes into account that the asymptotic consistency of the PLS estimator could not hold with a large number of possible predictors and a comparatively small number of observations, presenting an alternative that selects variables with a sparse linear combination of the predictors. The main difference between PCR, PLS, and SPLS is that the first two will not perform a direct selection of variables, considering that the direction vectors or the linear combinations will be derived from almost all the original features.

There are two approaches of PLS, [76] and [68], the selected by this method is the NIPALS, which can be reformulated by introducing an elastic net penalty for variable selection. There will be M components and $\mathbf{Z} = \mathbf{X}^{\mathrm{T}}\mathbf{Y}$, considering $\mathbf{Z} = Z_m$ and $m = 1, \ldots, M$ the problem can be expressed as:

$$\min_{c,r} -\kappa c^T \mathbf{X}^{\mathrm{T}}\mathbf{Y}\mathbf{Y}^{\mathrm{T}}\mathbf{X} + (1-\kappa)(c-r)^T \mathbf{X}^{\mathrm{T}}\mathbf{Y}\mathbf{Y}^{\mathrm{T}}\mathbf{X}(c-r) + \lambda_1|r|_1 + \lambda_2|r|_2^2 \tag{3.7}$$

Where:

- κ is the parameter that affects the closeness of original data and the direction vectors. The default parameter for simplicity is 0.5.
- λ_1 and λ_2 are the elastic net penalization parameters.
- c and r are direction vectors.

The problem will be solved by iterating and solving for r with fixed c and c while fixing r. Then, two problems are generated. First, by given c fixed:

$$\min_{r} \left\{-\kappa r^{\mathrm{T}}\mathbf{X}^{\mathrm{T}}\mathbf{X}r + (1-\kappa)(c-r)^{\mathrm{T}}\mathbf{X}^{\mathrm{T}}\mathbf{X}(c-r)\right\} \tag{3.8}$$

s.t. $r^{\mathrm{T}}r = 1$

This can be obtained by the maximization of $c^T Z^T Z r$ s.t $c^{\mathrm{T}}c = 1$

And the second formulation is given by a fixed r :

$$\min_{c} \left(\left(\mathbf{Z}^{\mathrm{T}}c - \mathbf{Z}^{\mathrm{T}}w\right)^{\mathrm{T}}\left(\mathbf{Z}^{\mathrm{T}}c - \mathbf{Z}^{\mathrm{T}}w\right) + \lambda_1|c|_1 + \lambda_2|c|_2^2\right\} \tag{3.9}$$

This can be seen as the PLS problem for:

$$Zc = \sum_{j=1}^{p} Z_j r_j + e^* \tag{3.10}$$

First it considers the initial stage as $m = 1$, $Y_i = Y$, $\hat{\beta}^{\text{pLS}} = 0$, $\mathbf{X}_s = \mathbf{X}$, then $S = 1 \ldots q$

The algorithm will act as follows.

Algorithm 3: Sparse Partial Least Squares

Result: Obtain components and a subset of variables

initialization;
for *i in* $1, \ldots, q$ **do**
  for *each m in* $1, \ldots, M$ **do**
    Solve initial problem Eq.3.9 considering that $\mathbf{X}^{\mathrm{T}} Y Y^{\mathrm{T}} \mathbf{X} = \mathbf{X}_s^{\mathrm{T}} Y Y^{\mathrm{T}_i} \mathbf{X}_s$ to find $\hat{r}$;
    New S considering $\left\{ j : \hat{r}_j \neq 0 \right\} \cup \left\{ j : \hat{\beta}_j^{\text{PLS}} \neq 0 \right\}$;
    PLS with m components for $\mathbf{X}_s$;
    Adjust $\hat{\beta}$;
  end
end

In this setting, the parameter η will give the level of sparsity considering that $\lambda_2 = \infty$, then it will be set as a value between 0 and 1.

Table 3.11. REQUIREMENTS MODEL 15. SPLS

Package	Function	Parameter	Description	Value
caret	train	trControl	10-fold cross-validation	method= "cv", number= 10
		tuneLength	Maximum of components	10
		tuneGrid	Level of sparsity considering or η	seq(0.1, 0.9, 0.1)
			Number of hidden components m	5
			The effect on the closeness or κ	0.5
caret/spls		method	definition of the algorithm	spls

Procedure:

1. There is a direct implementation through the package *caret* by the method "spls" created by the authors of the methodology presented in [74]. For that, the hyper-parameters, η, and M as the number of hidden components have been set; additionally, κ is considered equal to 0.5. The algorithm will run a 10-fold cross-validation

procedure to determine the best η taking two extreme values, and the model will be fitted retrieving the selected variables and the corresponding coefficients.

2. The performance of the model will be measure over the testing partition.

3.7.3. Additional approaches

Two different approaches are presented next; these consider hybrid methods that were developed after getting the results of the models shown above for the sample of 100 "non-modellable" variables.

Model 16. Reluctant Generalized Additive Models with a non-statistical filter

[33, pg.159] suggests that a good starting point considers a linear approach, followed by a non-linear one to see if the subsets achieve a better estimation. Following this approach, for the second step of the feature selection after implementing a non-statistical filter, a Reluctant Generalized Additive Model has been used. According to the results obtained in the development of the methods above, one appealable filter was the type of currency since it divides the data into more categories and, subsequently, more computational benefits can be retrieved. The requirements for the algorithm have been considered the same as for **Model.11**.

Model 17. XGBoost with a non-statistical filter

XGBoost can be considered a wrapper method for variable selection. The computational time derived by this technique alone, was not attractive comparable to some of the algorithms presented above. However, when connecting it to a previous non-statistical filter, the results demonstrated improvement in terms of time consumption and performance. For this, a filter based on the term was picked, considering that in a previous test with a sample of 100 “non-modellable” risk factors, the performance of this filter method preceded by a stepwise was at the top of the rankings. Additionally, this filter is available for the whole set, and previous studies run by the entity pointed that there could be a typical behavior in risk factors with similar terms.

A Gradient Boosting Machine (GBM) is a model based on supervised machine learning that involves creating an ensemble of weak methods [77]. According to [78], in 2018, more than half of the machine learning competitions at Kaggle used XGBoost or Extreme Gradient Boosting [79] to solve the problems presented, which is a recent evolution of Gradient Boosting that learns higher-order interaction between features.

Computationally speaking, the advantage behind it is that it can automatically perform parallel computation and can reach a reduction in 10 times the traditional computational cost of GBM. Additionally, it also accepts the performance of a linear or sparse algorithm. In this case, the sparse approach appears to work better for the data.

This method involves a complex tuning of the parameters. By taking into account that there has to be a balance between the complexity of the algorithm and the gaining in

performance, the tuning has been kept as light as possible considering the large number of models required, but without sacrificing significant losses in the accuracy measure. The resulting requirements for the algorithm are the following:

Table 3.12. REQUIREMENTS MODEL 17. XGBOOST

Package	Function	Parameter	Description	Value
caret	train	trControl	5-fold cross-validation with parallelization	method= "cv", number= 5, allowParallel =T
		tuneGrid	The 'nrounds' or number of trees	200
			Maximum tree depth given by 'max_depth '	5 or 10
			Learning rate or the step size of each boosting given by η	0.1
			Parameter of regularization given by γ	0
			The column sampling or 'colsample_bytree'	seq(0.75, 0.9, length.out = 3)
			The minimum leaf weight or 'min_child_weight '	1
			The 'subsample' or row sampling	0.5
caret/xgboost		method	definition of the algorithm	xgbTree

The procedure for this algorithm will start with the performance of the method "xgTree" in *caret* with a 5-fold cross-validation to find the optimal parameters from the grid set. Next, the importance of the features is retrieved by two functions `xgb.importance` from the package *xgboost* and `varImp` implemented in *caret*, will be retrieved and the model will be fitted to finally continue to the prediction with the testing data to prove accuracy.

3.8. Performance measures

Three measures have been considered to compare the performance of the models described. These ones are:

1. RMSE or Root Mean Square Error express as: $\text{RMSE}_i = \sqrt{\frac{1}{n}\sum_{k=1}^{n}(y_{ik} - \hat{y}_{ik})^2}$for $i = \{1, \ldots, q\}$ and $k = \{1, \ldots, n\}$

2. MAE or Mean Absolute Error express as $\text{MAE}_i = \frac{1}{n}\sum_{k=1}^{n}\left|y_{ik} - \hat{y}_{ik}\right|$ for $i = \{1, \ldots, q\}$ and $k = \{1, \ldots, n\}$

3. R-squared which considers the squared coefficient of the correlation between the predicted values and the observed ones. This one can be a good metric to compare models that use a linear form because of its use for measuring goodness of fit, however, in non-linear models there could be misleading results [80].

Additionally, the regulation requires that a PLA test is performed to assess the modellability of a risk factor [4, pp.81-88]. Then, to determine the best methods it is necessary to calculate two other metrics over the 12 past months of the daily variations. According to the regulatory framework, the real observations can be called Risk-theoretical P&L (RTPL) and the variations generated by the methods are referred to as the Hipothetical P&L (HPL), considering these notations, the metrics are the following:

- Spearman correlation which will measure the correlation among the RTPL and the HPL, expressed as:

$$r_{Si} = \frac{\text{cov}\,(R_{HPLi}, R_{RTPLi})}{\sigma_{R_{HPLi}} \times \sigma_{R_{RTPLi}}} \tag{3.11}$$

  Where, $\sigma_{R_{HPLi}}$ and $\sigma_{R_{HPLi}}$ are the standard deviation of the HPL_i and $RTPI_i$ and R_{HPLi} and R_{RTPLi} the ranked values of the observations of HPL and RTPL for $i = \{1, \ldots, q\}$.

- Kolmogorov-Smirnov test (ks) which quantifies the distance between the cumulative empirical distribution function of the RTPL and the cumulative empirical distributions of HPL to review similarities.

$$D_i = \sup_i |F_{RTPLik} - F_{HPLik}| \tag{3.12}$$

  Where, $\sup_x$ is the supremum or the maximum difference, and F_{RTPLik} and F_{HPLik} are the cumulative distribution functions of RTPL and HPL for $i = \{1, \ldots, q\}$

The PLA test will retrieve a result based on the performance of the criterion by classifying the risk factors in three zones as seen in **Table 3.13**.

Table 3.13. PLA METRICS CRITERIA

	Zone	Spearman correlation	Criterion	Ks test
	Green	>0.80	and	<0.09 (p-value = 0.264)
	Amber	$\geq 0.7 and \leq 0.8$	or	$\geq 0.09(p - value = 0.264) and \leq 0.12(p - value = 0.055)$
	Red	<0.7	or	>0.12 (p-value = 0.055)

The fulfillment of the criterion by each proxy of the NMRF will determine if it is suitable or not for being treated as a MRF, in terms of capital requirements. The classes "Green" and "Amber", will mean that it will be eligible to use the IMA to calculate market risk capital requirements. In contrast, if is it cataloged as "Red", it will be require to consider it to follow a capitalization under the Standardise Approach, and as mentioned previously, it will carry additional capital charges.

4. RESULTS

4.1. Partial Results

The sample generated randomly retrieved 100 NMRFs which based on its characteristics can be categorized as 43% long-term (lt) risk factors and 57% short-term (st), and 27% correspondent to Cross-currency curves (CC) and 73% to Single Currency Curves (SC). These numbers are congruent with the behavior of the population, considering that there are more SC in the dataset provided and that the observations are partitioned almost in half when taking the term into account (**Figure 4.1**).

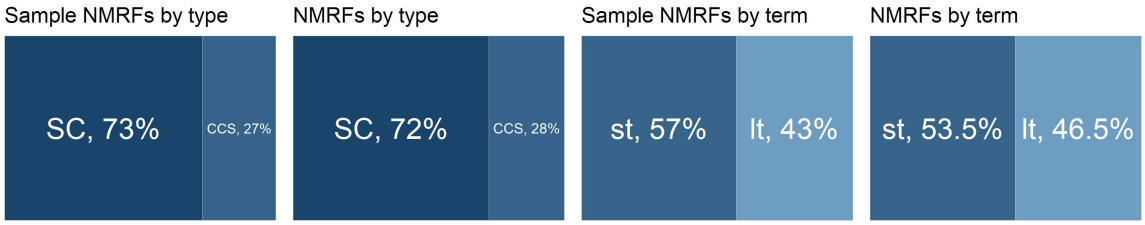

Fig. 4.1. Comparison of the characterization of the NMRF sample based on term and type

Additionally, concerning curves, the sample contemplates 64 distinct types of curves out of the total 113. Moreover, the composition of the sample distributed by currency shows that it takes into account NMRFs with all currencies from the population, but the portions differ. However, we can see similar behavior in terms of majority, given that the percentage of risk factors in the sample, as well as for the population, remains high in two specific currencies: USD and EUR (**Figure 4.2**).

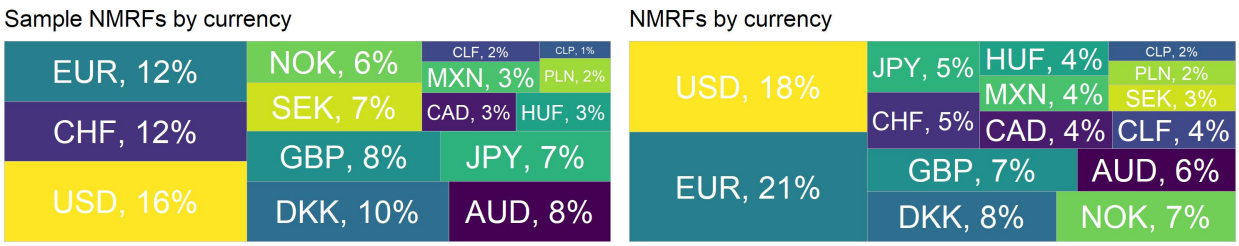

Fig. 4.2. Comparison of the characterization of the NMRFs sample based on the currency

Next, we present two approaches for analyzing the performance of the methods in the sample described. In the first place, we present a calculation based on the number of cases that each method achieves the best measurements, considered as the minimum MAE and RMSE, and maximum R-squared. Second, the performance will be displayed as a conglomerate average of the measurements by currency to review the overall error.

Considering that the R-squared is only a good measure for the models that contemplate linearity, it is not part of this last evaluation.

4.1.1. Count of maximum performance metrics per Model

As seen in **Figure 4.4**, the Model 3 corresponding to a non-statistical filter of type of curve and a subsequent stepwise, results in the model that achieved a lower RMSE for more variables. Additionally, when comparing it to other linear approaches, it can be seen that the measure of the R-squared also performs well.

Moreover, we can see that there are six models which present the best metrics in more "non-modellable" factors than others, these correspond to: Model 3, as said before, Model 4 which corresponds to a non-statistical filter of currency followed by a stepwise, Model 16 which is the variation of Reluctant Generalized Additive Model with a filter of currency, Model 7 or Stepwise for big data, Model 6 that implements a filter of term and a stepwise, and Model 5 that corresponds to a stepwise with a filter based on the curves. In contrast, the methods that perform worst, under these criteria, are the ones that use dimensionality reduction techniques and the linear penalizations of Ridge and Lasso.

4.1.2. Average MAE and RMSE

The second approach takes into consideration that a methodology could perform better for one NMRF and worse for others, then there might be a compensating effect that could give a better global result. For this, a calculation for the average of the MAE and RMSE was performed. The results are shown by currency considering that these are the units of measurement of the response variables which correspond to the NMRFs.

To show the difference in the metrics, the results have been normalized, being 1 the minimum value and 0 the maximum one, the methodologies that present missing values are left as "NA" to not affect the metrics considering that there are methodologies that are not suitable for certain variables. Specifically, Model 5, the stepwise with a filter of the curve will only be valid for curves that have tenors "non-modellable" and "modellable". Additionally, we can see that the methodology for Model 2 is not suitable for NMRFs that are poorly correlated with the candidate predictors. The number of missing values in each method can be seen in **Figure 4.3**.

Fig. 4.3. Missing values in each method for partial results

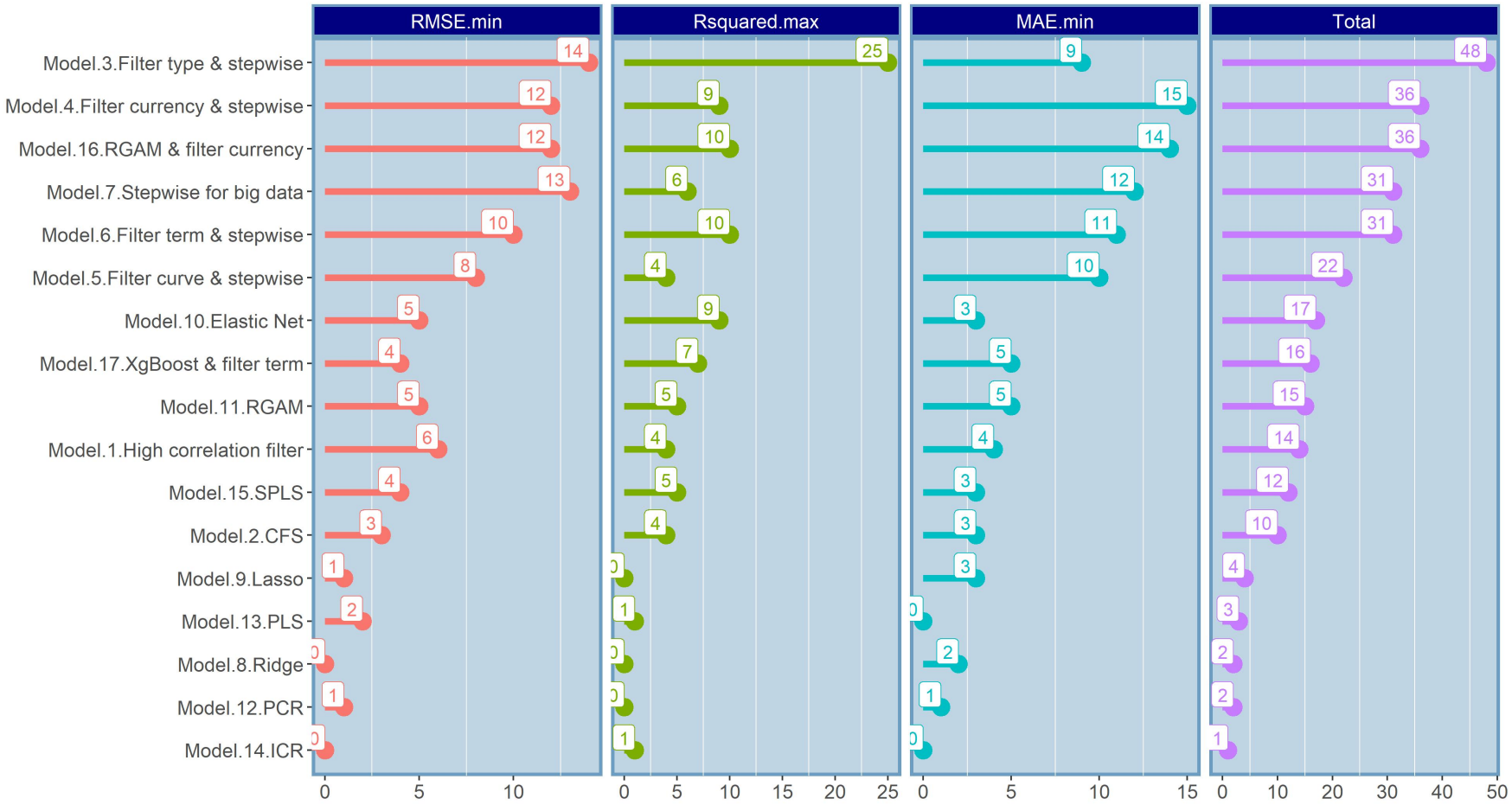

Fig. 4.4. Partial results for models metrics as counts of cases of maximum performance

As seen in **Figure 4.5**, the normalized average RMSE presents, in general terms, lower values and greater difference among methods, showing a more conservative approach. The RMSE and the MAE, both do not consider the direction of the errors but the RMSE differs by giving high weight to larger errors, denoting that there is a higher difference across the methods. In consequence, we will focus in the results given by this measure.

In concordance to the normalized average RMSE, the methodologies that have performed better are Model 3, Model 4, Model 5, Model 6, which recall are filter-based. Additionally, results showed that the wrapper method of the stepwise for big data, the Elastic Net, and the RGAM in both of its variations (Model 11 and Model 16 with a filter) performed well in almost all currencies. In contrast, Model 9, which corresponds to Lasso, factor base models 12 and 14, PCR, and ICR, correspondingly, presented lower values showing higher metrics for the error rate.

Viewing the results across currencies, we can see that for some of them the metrics are stable among the models that performed better. For example, for CAD, CLP, EUR, PLN, and USD, we can see that the normalized average RMSE presents measures over 0.7 for models 3, 4, 5, 6, 7, 10, and 16. Additionally, for a few currencies, the metric shows higher differences among methodologies, highlighting the importance of one of the methods. Particularly, in the cases of AUD, CHF, JPY, and NOK, we can see that Model 5, the filter method for curves, presents the highest average performance with a marked margin. For CLF, we see the same behavior, but with Model 7.

4.1.3. Selected methods

The results from both approaches showed that the methods that performed better are filter-based and wrapper. In contrast, the lower metrics correspond to dimensionality reduction techniques that are unsupervised algorithms.

Moreover, in terms of linearity by comparing Model 4, which is a filter-based method followed by a stepwise, and Model 16 that uses the same filter but followed by an RGAM that contemplates non-linear features, we observed similar performance. In consequence, we can say the non-linear effects are reduced, at least for this sample of NMRFs.

The time derived by the calculation of each method is presented in **Figure 4.6**. The computational time is not related to the performance achieved, as seen and in concordance to literature, the filter-based methods which are the less time-consuming methods, presented also better metrics than more complex algorithms. Additionally, by comparing Model 11, the RGAM, with Model 16, the RGAM with a currency filter, we can see a major improvement in time, illustrating the gaining derived by a pre-screening methodology.

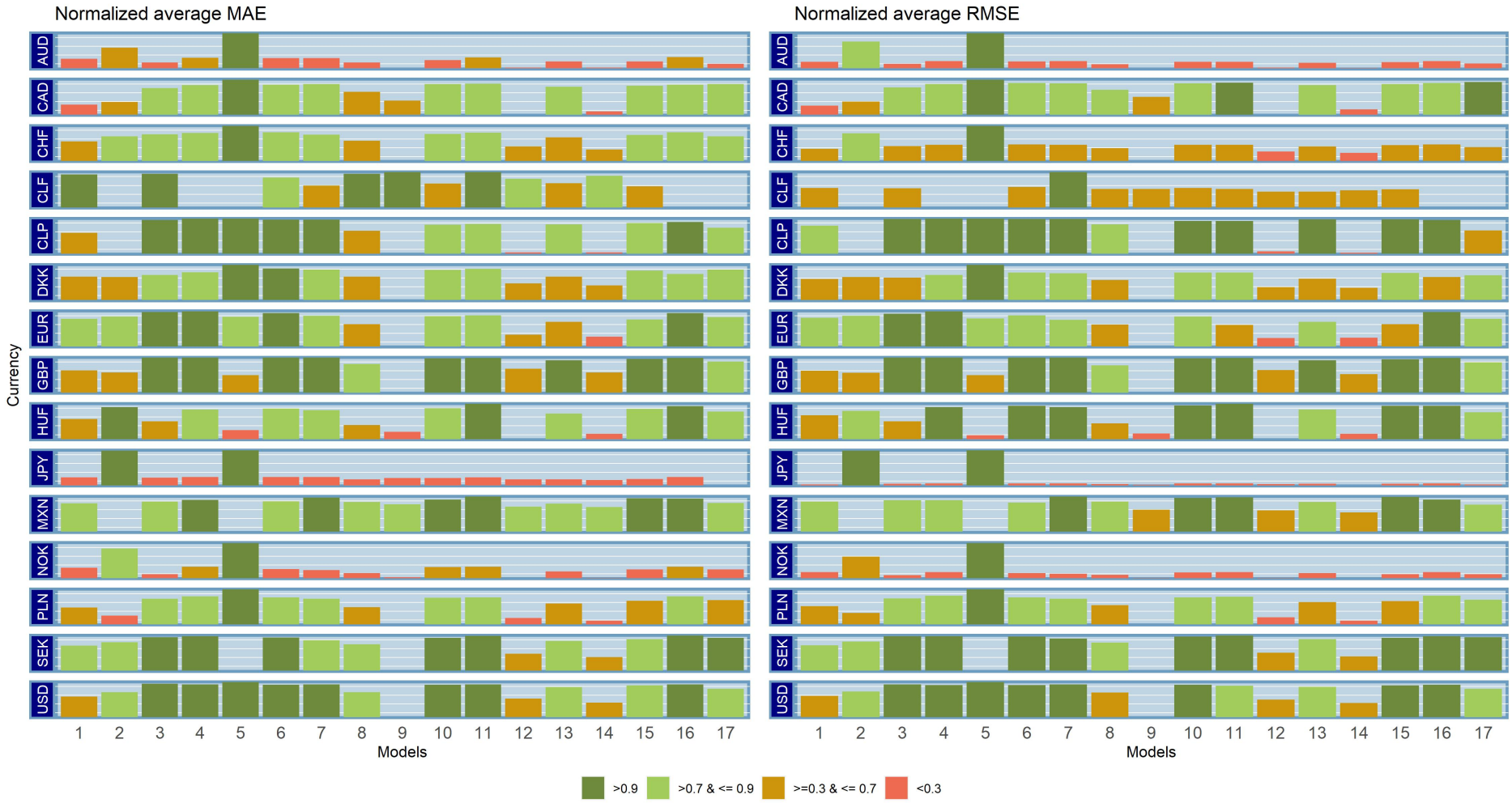

Fig. 4.5. Partial results for models measure as the normalized average MAE and RMSE per currency, where 1 is the minimum and 0 the maximum RMSE

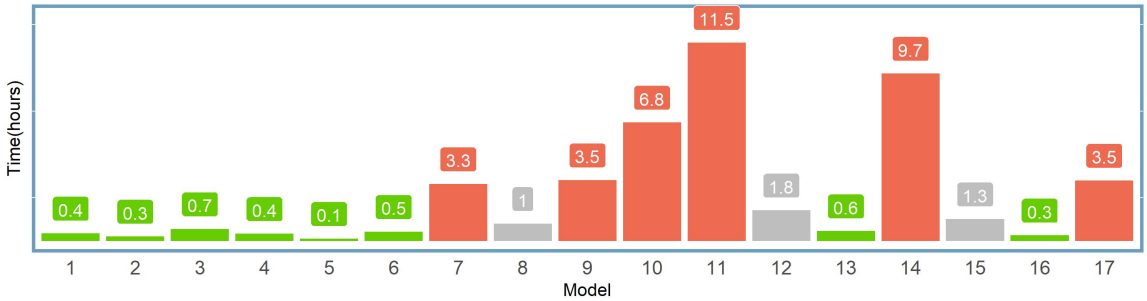

Fig. 4.6. Computational time for each method measure in hours

Based on the two approaches for measuring the performance metrics, the selected methods to be executed over the complete set of NMRFs have been the filter-based methods: Model 3, Filter by type & stepwise; Model 4, a filter by currency & stepwise; Model 5, filter by curve name & stepwise; and Model 6, a filter by term & stepwise. Also, the wrapper methods: Model 7, the stepwise for big data, and Model 16, RGAM & currency filter. Contemplating, the computational time of Model 11, and considering that Model 16 is presented as a variation of it, it was not considered. Furthermore, Model 10, was also deemed out of the scope of this paper for its computational time and given that only very few risk factors lead to obtaining the best metrics, as shown by the first approach developed to measure performance.

The next section will present the total results obtained for the methods described, contemplating two different partitions, a more conservative procedure that uses 80% of the NMRF for training and the rest 20% for testing, and the one required by the regulator that uses only the last 12 months to measure the performance and the effectiveness of the methods.

4.2. Total results: conservative partition

To measure the performance of the methods and achieve a comparison among them, the same approaches of the last section were used.

First, **Table 4.1** provides the results in terms of counts of the risk factors that resulted in the minimum metric with each method. As seen, almost 50% of the NMRFs achieved their minimum measure of RMSE with the stepwise for big data and the RGAM. The other half is distributed in the remaining methodologies, being Model 5 the one that presents the lowest score with just 222 NMRFs, over the total 4,260. Recall, that precisely this last method cannot be generated for all risk factors. Moreover, by reviewing the number of missing values in the method (3064)(**Figure 4.7**), we can say that if there were more NMRFs suitable to be treated by Model 5, this could perform better than other models. This finding is derived from the hypothetical case in which 4,260 factors belong to curves that also present other factors in the "modellable" partition and that the ratio of the number of factors that reached the minimum RMSE with this methodology, equivalent to 18.56%

Table 4.1. TOTAL RESULTS MEASURE AS COUNTS OF MAXIMUM PERFORMANCE

Model	Minimum RMSE	Minimum RMSE (%)	Minimum MAE	Minimum MAE (%)
Model.7.Stepwise for big data	1053	24,7%	906	21,3%
Model.16.RGAM & filter by currency	950	22,3%	1104	25,9%
Model.6.Filter term & stepwise	733	17,2%	700	16,4%
Model.4.Filter currency & stepwise	710	16,7%	718	16,9%
Model.3.Filter type & stepwise	592	13,9%	611	14,3%
Model.5.Filter curve & stepwise	222	5,2%	221	5,2%
Total NMRF	**4260**	**100%**	**4260**	**100%**

(222/1196), is maintained. This would mean that this methodology could present 853 with minimum metrics, surpassing methods 3, 4, and 6.

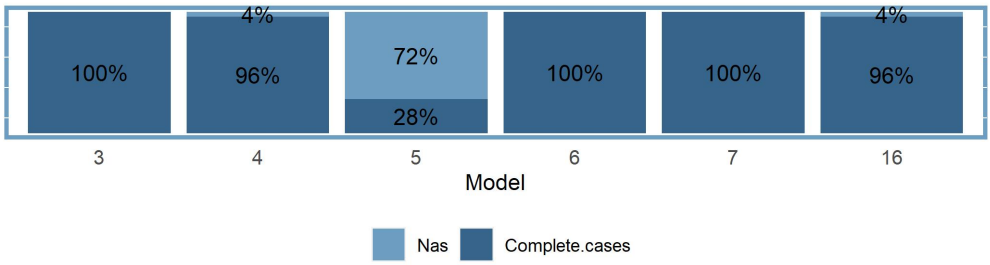

Fig. 4.7. Missing values in each method for total results

Second, we present the normalized average error metrics of MAE and RMSE per currency, considering that, as mentioned, there are methods that could not be calculated due to the nature of the data that does not allow comparison over the sum of the metrics. As **Figure 4.8** shows, there are more similarities among the results of the metric of MAE and RMSE under this scope, however, we see cases like for the USD, EUR, and CLF, for which both metrics differ.

As previously stated, the RMSE could act as a more conservative metric considering that captures better the larger effects. Then, according to the normalized average RMSE measured across the currencies, we can see that Model 5 and Model 7 performed better. Model 5 presents a metric over 0.9 for eight currencies and denotes larger differences with the results of the other methods, indicating an improvement in the predictions. Moreover, Model 7 presents, in six of the remaining seven currencies, a normalized average RMSE higher than 0.9.

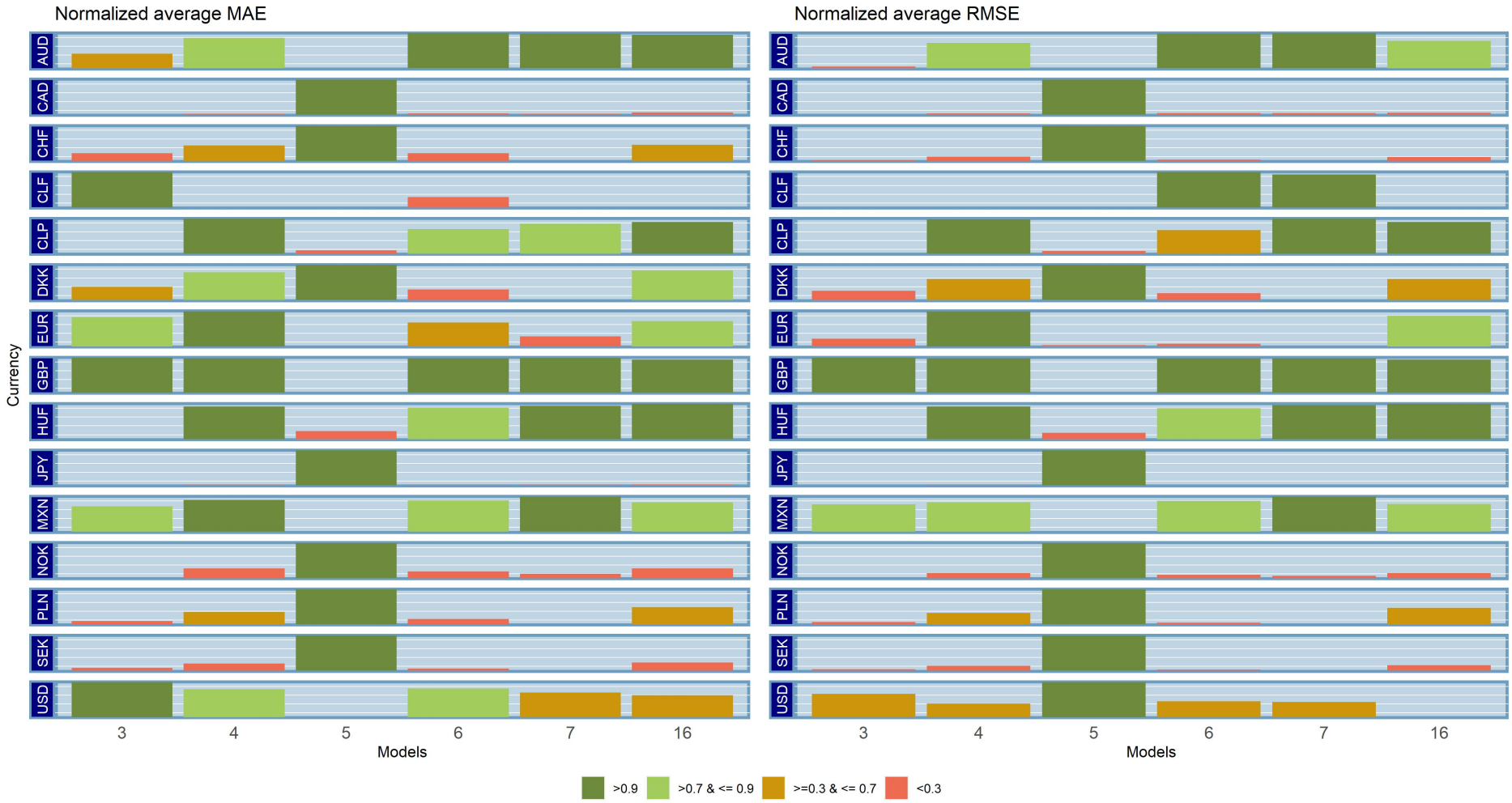

Fig. 4.8. Total results for models measure as the normalized average MAE and RMSE per currency, where 1 is the minimum and 0 the maximum RMSE

In the latter, we can see more stability in the rest of the metrics, in consequence, the difference generated by this methodology is not as large as for the last case. The remaining currency, EUR, presents Model 4 as the one with a higher metric of normalized average of the RMSE presenting a close metric to Model 16 and a big difference countered with the rest of the methods.

By taking a look at the results of the metrics under both methodologies acros currencies, it is clear that there is not one path solution for the data. In consequence, for the partition required for the regulator, the five methods will remain in the scope of this study.

4.3. Total results: regulator partition

To measure the effectiveness of the methods used to generate the proxies for the NMRFs, the regulator requires a measure of their performance over the last 12 months, specifically, using two metrics: the Spearman Correlation and the Kolmogorov-Smirnov test. As stated in **Section 3.8**, the results of both tests combined will return the zone of each factor, being the "Green" and "Amber" the ones of greater interest because both allow the capitalization of the NMRFs under IMA. In contrast, if the proxy does not reach the minimum requirements in the metrics, then it will be definitively deemed to be treated as an NMRF.

According to the regulator test and by taking into account the best zone achieved by each NMRF from the results of all of the six methods, it can be noticed that 68.2% of the NMRFs could be expressed in terms of MRFs. Additionally, a 5% , corresponding to the "Amber" zone, could also be subject to be capitalized under IMA. This means, that only 27% of the NMRFs will be definitively deemed as NMRFs after the proposed methodology (**Figure 4.9**). Next, we present some characterizations of the risk factors based on the zone to gain a global understanding of the NMRFs in terms of the results of the final proxies.

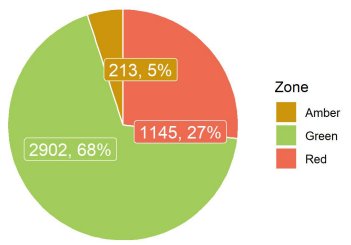

Fig. 4.9. Composition of proxies for NMRFs

4.3.1. Characterization of NMRFs proxies by zones

As **Figure 4.10** shows, this 68.12% of "Green" proxies is composed by a 37% of long-term risk factors and a smaller portion of 31% of short-term risk factors, even though the total number of short-term risk factors in the trading desk is higher (see **Section 4**, **Fig. 4.1**). Consequently, it can be said that better proxies were achieved for long-term NMRF tenors.

Concerning the type of curve, the NMRFs that could be treated as MRFs through

proxies, correspond mostly to Single Currency curves, this could be explained by the fact that Cross-Currency curves are constructed by a combination of risk factors which increases its complexity and hinders its subsequent decomposition.

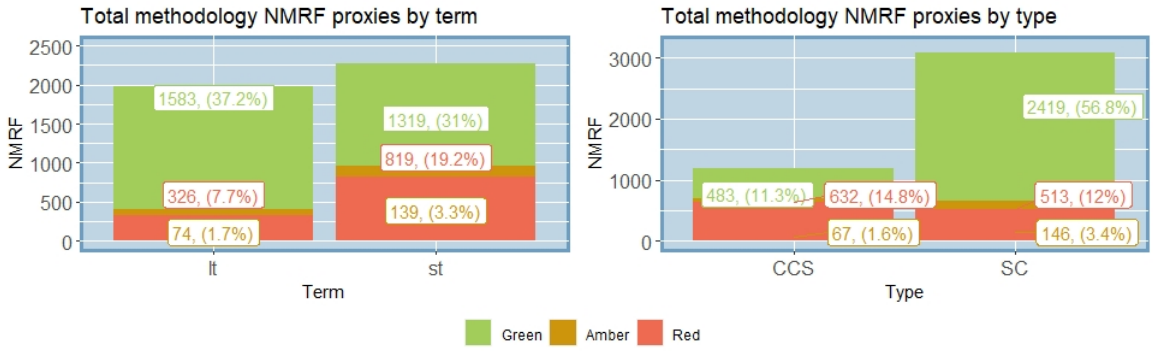

Fig. 4.10. Composition of proxies generated for NMRF based on zones type and term

By looking in detail the results for each method, it can be stated that there are similar results (**Figure 4.12**). Model 7, presented the highest number of NMRFs categorized as "Green", in consequence, could act as the best single option. However, considering the computational time derived from its calculation process (**Figure 4.11**), other options with very close results could be considered, such as Model 16 based on an RGAM. About this method, we can highlight that it could achieve greater stability when replicating it with new input data with different behavior, considering that it can capture both linear and non-linear effects. Additionally, as shown in the section above there is a direct benefit derived from the currency filter applied at the beginning of the pipeline.

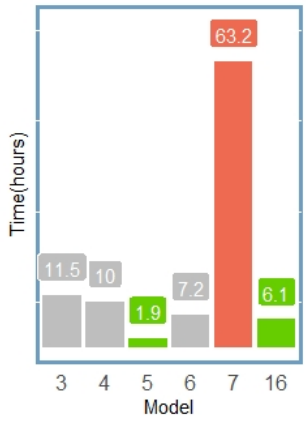

Fig. 4.11. Computational time of parallelized methods

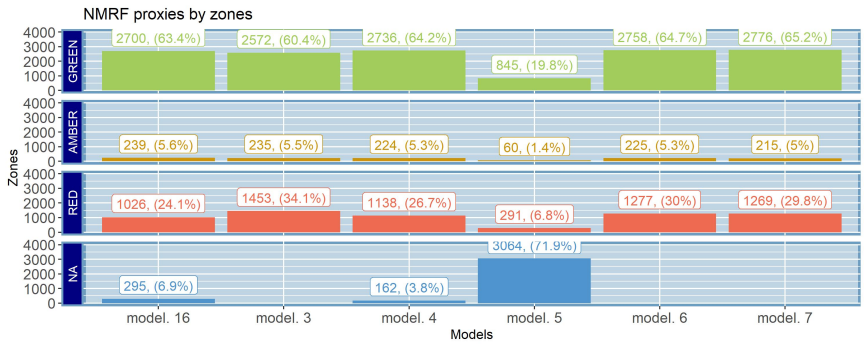

Fig. 4.12. Zone composition of proxies for NMRF by methodology, expressed as counts and percentage over the total NMRF

Nevertheless, it is strongly suggested that the methodology does not depend only on one method, taking into account that there are methods that work well like Model 5 but, as mentioned, are not suitable for all variables. Then, there will be a general gaining if the use of more than one method is implemented. As presented above, by considering the approach of multiple methods, the total number of "Green" NMRFs that are suitable for IMA and can be treated with proxies, resulted in 2902, exceeding the maximum number of any method at its own, as seen in the figure above.

Likewise, the importance of risk factors corresponding to the currencies of EUR, USD, and GBD is evidenced in this trading desk. **Figure 4.13** shows that the amount of NMRF that are categorized as "Green", reached up to 88%, 92%, and 93%, respectively, out of the total of NMRFs for each currency. The stability in the results, denoted by these currencies, implies that the methods picked are delivering predictions that are capable of decomposing the risk factors in an effective way.

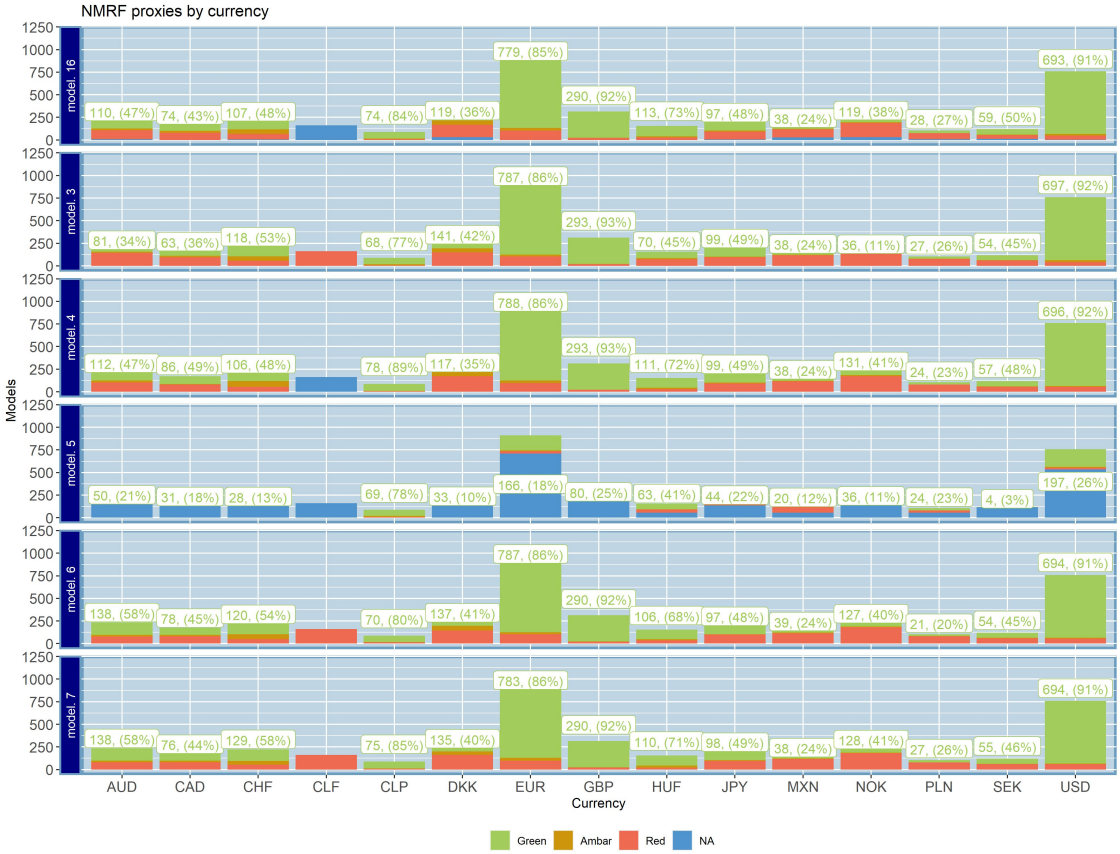

Fig. 4.13. Zone composition of proxies for NMRF based on zones, by methodology, term, and type, expressed as counts and percentage over the total NMRF by currency

Moreover, in less common currencies (CAD, NOK, AUD, and HUF) we see larger differences among methods. For those, it can be seen that Model 3 performs poorly. This has to do with the filter that was used, considering that takes into account the type of instrument to pre-select the candidates to act as predictors. These less common currencies are commonly represented by Cross-Currency curves, which as stated before, are less

suitable to be easily decompose. However, they do show better results with other methods like Model 4, which uses a currency filter, and Model 6 that presents a term filter.

Additionally, we have measured the average metric for this partition, as for the other cases. The results are similar to the ones presented in the previous section. For the first approach (**Annex 7.6**), we can see that more NMRFs achieved better RMSE metrics through Model 7, followed by Model 16. Nonetheless, there is an exchange between Model 16 and the remaining, considering that more than 110 NMRFs pointed to other methods. With respect to the second approach (**Annex 7.7**), it can be observed that when testing the methods with the last 12 months of the series, the type of curve gains greater importance over the name of the curve for the USD NMRFs. However, there is clear stability of the pointed methods of the rest currencies across the two partitions, denoting a congruent method selection.

4.4. Methodology results

The final methodology to obtain the most suitable method for each NMRF involved: in the first place, the criterion granted by the regulator; in the second place, the statistical accuracy from the conservative partition, which takes into account a longer period for testing; and in the third place, the performance in terms of computational time.

The procedure starts with the "best zone" criterion, understood as picking the method that retrieves the zone that it is more beneficial ("Green" over "Amber, and both over "Red"). Moreover, if two methods pointed to the same "best zone", then we returned to the results of the conservative partition to check which methods of this subset with the same zone reflected the lowest RMSE. In the case that there was more than one with the same RMSE, we took as reference the computational time retrieved in the calculation of the methods for the set of the 4,260 NMRFs to pick the method that involved less time.

By comparing the final results of the detailed methodology and the results considering the minimum RMSE for each NMRF, we can see that both are coherent with a percentage of stability in the selection of the method of more than 96%. As **Figure 4.14** shows, the 98% of NMRFs that use Model.7 to calculate its proxies under this methodology, were determined to use the same model via minimum RMSE. Additionally, we can see that this model is the one that retrieves the best results as more NMRFs were set to use this method, followed next by Model 16 the RGAM, and Model 6 a stepwise with a term filter. In contrast, Model 4 presents the lowest stability, as 8.3% of these NMRFs were pointed to use Model 16, if taking into account only the minimum RMSE. However, recall that both methods use the same currency filter, consequently, results could be related.

4.4.1. Characterization of the selected predictors

Finally, with the aim of understanding the relationships among the characteristics of NMRF and MRF, an analysis has been conducted over the selected predictors for the chosen method for each NMRF.

Among the important insights that are word mention, is the relevance of the type of curve, especially in the case of Single Currency curves (SC), considering that 87.6% of the predictors of SC NMRF correspond to SC MRF. In contrast, for CC NMRF curves we see that the percentage of the selected CC MRF is 53.7%, which is considerably lower.

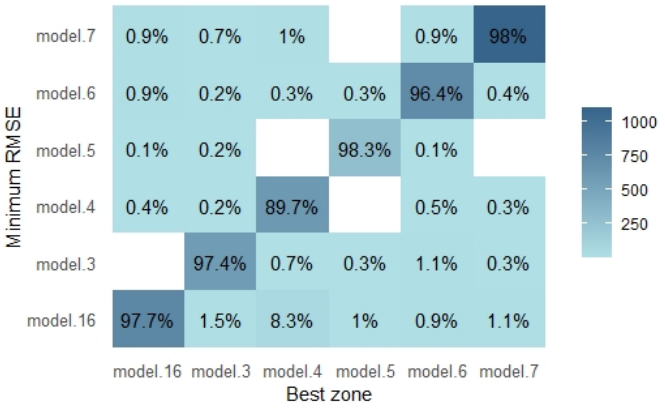

Fig. 4.14. Comparison among final methodology and minimum RMSE in terms of percentage and gradient by number of NMRF

Additionally, the filter of the term could result less informative taking into account that short-term NMRFs present 66.9% of short-term MRF predictors, while long-term NMRFs present 66.3% of long-term MRF predictors, presenting in both cases an almost divided partition among the distinct terms.

Moreover, if we cross both, term and type filters **Figure 4.15**, we can see that 61.7% of the short-term SC predictors and 53% of long-term SC predictors show the same term and type of their correspondent NMRFs. However, the predictors which share similar characteristics with the corresponding NMRFs for short-term CC and long-term CC are only 37% and 44.2%, respectively. This means that a combined filter of this nature could not act as a general viable option for treating a large number of variables.

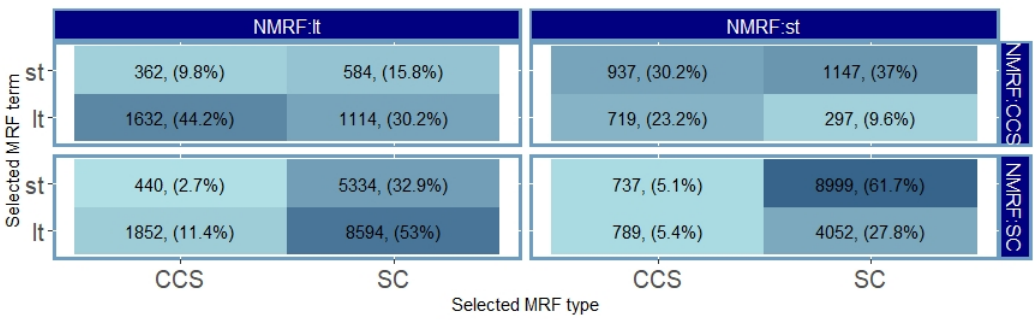

Fig. 4.15. Characterization based on the type of NMRF and the type of predictors, considering term

Nevertheless, when exploring, under the same methodology, a combined filter of currency and type of curve, we can see that this could become a better suitable option.

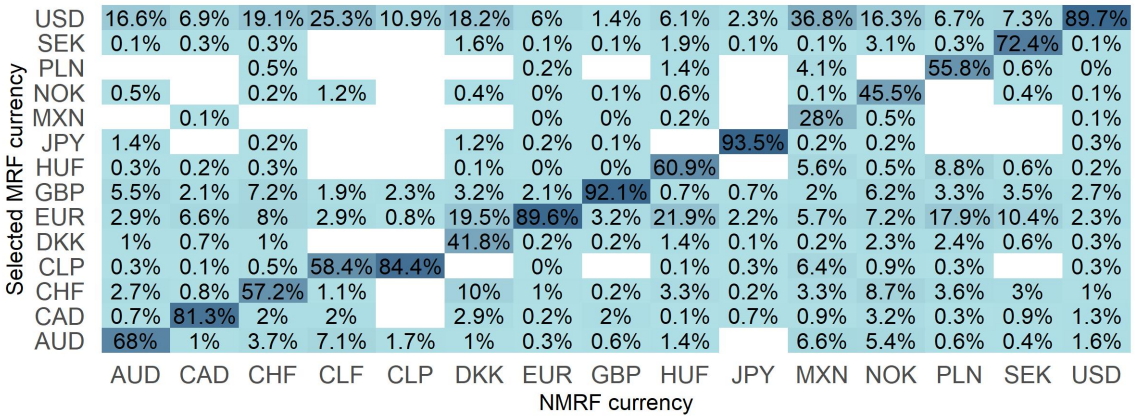

Fig. 4.16. Characterization based on the currency of NMRF and the currency of predictors

As **Figure 4.16** shows, more than 80% of the predictors present the same currency than for the NMRFs in the cases of CAD, CLP, EUR, GBP, JPY, and USD. Moreover, the cross matrix of the characterization of the predictors and the NMRFs shows that more than 80% of the predictors for the SC NMRFs of the currencies EUR, GBP, HUF, PLN, SEK, and USD, present the same currency and type (**Figure 4.17**). Recall that almost half of the NMRFs of this trading desk are expressed in USD, EUR, and GBP (**Figure 4.2**). Likewise, it is important to mention that the NMRFs related to these currencies represent the 41.2% of the total of NMRFs, where a 40.2% correspond to SC NMRFs. Also they constitute a majority of the 61.4% of the NMRFs that could be treated as MRFs given that were deemed to belong to the "Green" zone. In consequence, there is clear importance of the curves related to these currencies in the analysis, and considering that this combined filter will directly select the more adequate candidates for these curves, it is a suitable solution to be explored in these risk factors and could be tested combined to the methodology developed in other type of data to test its scope.

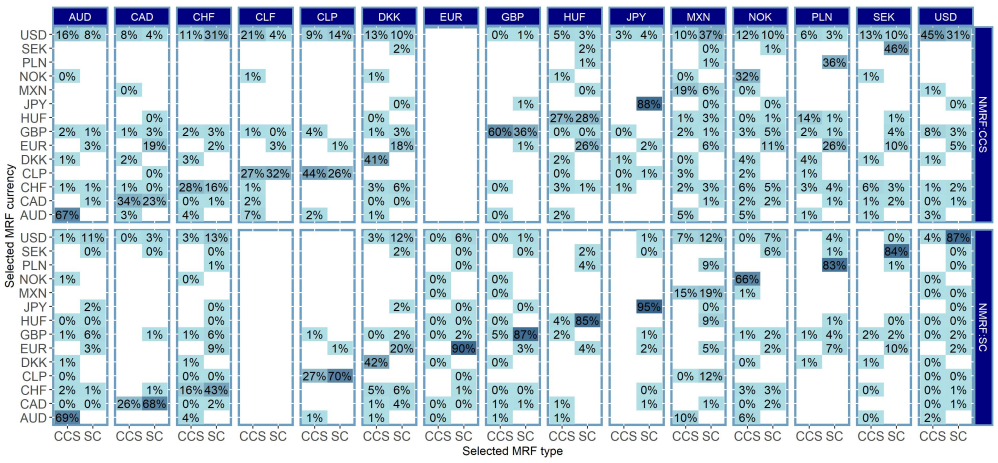

Fig. 4.17. Characterization based on the type of NMRF and the type of predictors, considering currency

5. CONCLUSIONS

The implementation of the Fundamental Review of the Trading Book (FRTB) will bring great challenges for the financial institutions considering that it has led to radical reforms for a more conservative market risk framework. The concept of "modellability", particularly, will involve the distinction of risk factors and a posterior type of capital requirement associated with each one. More NMRFs will imply higher capital charges that could affect the profitability of banks. Throughout this study, we have explored distinct feature selection methods to create a methodology that, while responding to the requirements of scalability, reproducibility, and traceability can express effectively the NMRFs given in terms of MRFs.

The dataset studied showed risk factors with different characteristics, and consequently, the best methodology has resulted in a determination of a specific method for each NMRF so it will lead to an improving performance derived by the inclusion of methods that, given the nature of the data, are not suitable to all variables but perform exceptionally for some NMRFs. Under this methodology, out of 4260 NMRFs, 27% have been sent to the set of factors that could not be treated under the Internal Model Approach, while for the remaining 73% we could avoid the additional associated capital requirement.

For this data, we can highlight that the methods with less predictive power are the ones that search for dimensionality reduction via projection. In turn, the most accurate resulted to be filter methods, a stepwise for big data, and a Reluctant Generalized Additive Model.

There has been a proven gain in computational time derived from the use of filters. Therefore, it is highly recommended that these are included in new methodologies because they diminish the candidates of MRFs to become predictors of NMRFs, simplifying the algorithms. In this set, the curve name and the currency filter have shown to be good pre-screening methods, that could be adjusted before other methods as a previous step. Moreover, based on the results, we saw that a combined filter of currency and type of curve should be an option to be explored.

As per the curves, based on the data of this trading desk, it can be concluded that NMRFs of Single Currency curves have shown to be more easily decomposed than Cross-Currency curves and that there are highly relevant currencies like EUR and USD that are conditioning the capital charges because they represent a high portion of the NMRFs that could be treated under IMA.

6. FUTURE LINES OF WORK

There is not a consensus on which method of feature selection is better, so each situation requires a review of distinct methods. We have provided an extensive analysis of some well-known feature selection methodologies that can be applied to new data including more recent information, or different data from other trading desks that involve other types of risk factors. In the latter, it is proposed that the presented methodology is followed but that new methods, based on the criteria of the expert, are also included (e.g. linear methods could bias the results when there are underlying non-linearities, then for different risk factors it could be recommended to prove more models that take into account both effects).

We have analyzed the interest rate asset class, but according to the BCBS, the higher proportion of NMRF can be seen in Equity risk factors. Therefore, it would be appropriate to test the same methodology in this type of data considering that the benefits could be higher due to the larger capital charges that could be associated. Ideally, all the proposed methodologies (Model 1 to 17) could be applied to the new set of NMRFs to review in detail the performance of the methods.

Finally, two other approaches are left for further discussion, one is to consider Mix Integer Optimization to solve the best subset problem [27], and another one to consider temporal characteristics by: first, including feature engineering and adding date-related features or rolling widows; second, with the use of Multivariate Bayesian Structural Time Series Model [25]; and third, through the implementation of feature selection methods addressed jointly with Neural Networks [26].

BIBLIOGRAPHY

[1] BCBS, "Fundamental review of the trading book: A revised market risk framework", *Bank for International Settlements*, 2013. [Online]. Available: https://www.bis.org/publ/bcbs265.pdf.

[2] "From theory to action: Fundamental Review of the Trading Book", Accenture, Tech. Rep., 2017. [Online]. Available: https://accntu.re/32geLpY.

[3] EBA, "Instructions for filling the NMRF SSRM data template", Jun. 2019, [Online]. Available: https://bit.ly/2FffKxF.

[4] BCBS, "Minimum capital requirements for market risk", *Bank for International Settlements*, 2019. [Online]. Available: https://bit.ly/3bOCall.

[5] BCBS', "Fundamental review of the trading book – interim impact analysis", *Bank for International Settlements*, 2015. [Online]. Available: https://bit.ly/3k52J90.

[6] S. Sharma, "Managing "Non-modellable" risk factors under FRTB", Aug. 2020, [Online]. Available: https://bit.ly/3bLwYPk.

[7] EBA, "Consultation Paper: Draft Regulatory Technical Standards on the calculation of the stress scenario risk measure under Article 325bk(3) of Regulation (EU) No 575/2013 (Capital Requirements Regulation 2 - CRR2)", Jun. 2020.

[8] H. Farag, "Bracing for the FRTB: Capital, business and operational impact", *Journal of Securities Operations Custody*, vol. 9, no. 2, pp. 160–177, 2017.

[9] H. Zhang, "Market Risk Modeling Framework under Basel", *Commercial Banking Risk Management*, 2017. DOI: https://bit.ly/2RcDMf7.

[10] J. Fidrmuc and R. Lind, "Macroeconomic impact of Basel III: Evidence from a meta-analysis", *Journal of Banking and Finance*, vol. 112, no. 1, pp. 105–359, 2020. DOI: 10.1016/j.jbankfin.2018.05.017..

[11] T. Obitz, "Estimation of Risk Factor Modellability as a Function of Arrival Rate and Seasonality", Dec. 2016. [Online]. Available: https://bit.ly/3bPg2HG.

[12] B. Slime, "Non-Modellable Risk Factor (NMRF) Measurement Using Gaussian Process Regression (GPR)", *SSRN Electronic Journal*, Nov. 2018.

[13] A. Lloret, "Interest Rate Volatility Risk under FRTB", *Universidad de Valencia*, Jul. 2019. [Online]. Available: https://bit.ly/32bOEjL.

[14] N. Lehdili, P. Oswald, and H.Guéneau, *Market Risk Assessment of a trading book using Statistical and Machine Learning*, Nov. 2019.

[15] M. Leo, S. Sharma, and K. Maddulety, "Machine Learning in Banking Risk Management: A Literature Review", *Risks*, vol. 7, p. 29, Mar. 2019.

[16] H. Liu and H. Motoda, *Computational methods of feature selection*. Chapman Hall, 2008.

[17] A. Blum and P. Langley, *Selection of relevant features and examples in machine learning. Artificial intelligence*, ser. 1-2. 1997, vol. 97.

[18] U. Stańczyk and L. Jain, "Feature Selection for Data and Pattern Recognition", vol. 584, 2014.

[19] R. Nilsson, J. Peña, J. Björkegren, and J. Tegnér, "Detecting multivariate differentially expressed genes", *BMC Bioinform*, vol. 8, no. 150, 2007.

[20] A.Kalousis, J.Prados, and M. Hilario, "Stability of feature selection algorithms: A study on high-dimensional spaces", *Knowledge and Information Systems*, vol. 12, no. 1, pp. 95–116, 2007. DOI: 10.1007/s10115-006-0040-8.

[21] G. James, D. Witten, T. Hastie, R.Tibshirani, and R. Gazzo, *Introduction to Statistical Learning: with Applications in R*, 6th, S. F. G. Casella and I. Olkin, Eds. Springer Science Business Media, 2013.

[22] K. Ramasubramanian and A. Singh, *Machine Learning Using R. A Comprehensive Guide to Machine Learning*. New Delhi, Delhi, India: Springer, 2017.

[23] V. Bolón-Canedo, N. Sánchez-Maroño, and A. Alonso-Betanzos, "A review of feature selection methods on synthetic data", *Knowledge and Information Systems*, vol. 34, no. 3, pp. 483–519, DOI: 10.1007/s10115-012-0487-8.

[24] J. George, R. Kohavi, and K. Pfleger, "Irrelevant Features and the Subset Selection Problem", *Elsevier Inc*, pp. 121–129, 1994.

[25] S. Jammalamadaka, J. Qiu, and N. Ning, "Multivariate Bayesian Structural Time Series Model", *Journal of Machine Learning Research*, vol. 19, Jan. 2018.

[26] S. Crone and N. Kourentzes, "Feature selection for time series prediction – a combined filter and wrapper approach for neural networks", *Neurocomputing*, pp. 1923–1936, Jun. 2010. DOI: 10.1016/j.neucom.2010.01.017.

[27] D. Bertsimas, A. King, and R. Mazumder, "Best Subset Selection via a Modern Optimization Lens", *The Annals of Statistics*, vol. 44, Jul. 2015.

[28] M. Karagiannopoulos, D. Anyfantis, S. Kotsiantis, and P. Pintelas, "A feature selection for regression problems", *Proceedings of the 8th Hellenic European Conference on Computer Mathematics and Its Applications (HERCMA '07)*, 2007.

[29] S. Sweet and K. Grace-Martin, "Data Analysis with SPSS: A First Course in Applied Statistics", *American Sociological Association*, vol. 36, no. 3, pp. 285–287, 2008.

[30] J. Meyer, "Differences in Model Building Between Explanatory and Predictive Models", *The Analysis Factor.*, Jul. 2020. [Online]. Available: https://bit.ly/3hfb3B5.

[31] K. Grace-Martin, "The Difference Between Interaction and Association", *The Analysis Factor*, Jul. 2020. [Online]. Available: https://bit.ly/3bLX2tp.

[32] G. Shmueli, "To explain or to predict?", *Statistical science*, vol. 25, no. 3, pp. 289–310, 2010. DOI: 10.1214/10-sts330.

[33] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection", *JMach Learn Res*, vol. 3, pp. 1157–1182,

[34] G. Piatetsky-Shapiro and W. J. Frawley, "Knowledge discovery in databases", *MIT Press*, 1991.

[35] J. Li *et al.*, "Feature selection", *ACM Computing Surveys (CSUR)*, vol. 50, no. 6, pp. 1–45, 2018. DOI: 10.1145/3136625.

[36] K. Randolph and L. Myers, "Basic statistics in multivariate analysis", *Oxford University*,

[37] I.-K. Yeo and R. Johnson, "A new family of power transformations to improve normality or symmetry", *Biometrika*, vol. 87, no. 4, pp. 954–959, 2000.

[38] S. Weisberg, "Yeo-johnson power transformations", *Department of Applied Statistics, University of Minnesota*, 2001. [Online]. Available: https://bit.ly/2GFTldf.

[39] M. Kuhn, "Findlinearcombos", *R Documentation*, Aug. 2020. [Online]. Available: https://bit.ly/2F52qMx.

[40] M. Khun, "Caret package", *Cran*, Aug. 2020. [Online]. Available: https://bit.ly/3jX4tRi.

[41] M. Kursa, "Boruta for those in a hurry", *Cran*, 2020. [Online]. Available: https://bit.ly/3hheHKJ.

[42] J. Kacprzyk, "Feature extraction: Foundations and applications", vol. 207, 2006.

[43] M. Khun, "The caret package", *Github*, Jul. 2020. [Online]. Available: https://topepo.github.io/caret/index.html.

[44] Y. Bengio and J.Bergstra, "Random Search for Hyper-Parameter Optimization", *Journal of Machine Learning Research*, vol. 13, 2012. [Online]. Available: https://bit.ly/2F52Wdr.

[45] R. documentation, "Expand.grid", Tech. Rep., Aug. 2020. [Online]. Available: https://bit.ly/3k031Oe.

[46] M. Hall, "Correlation-based Feature Selection for Machine Learning", PhD thesis, Dpto. Of Computer Science, Hamilton, 1999.

[47] L. Yu and H. Liu., "Feature Selection for High-Dimensional Data: A Fast Correlation-Based Filter Solution", *Department of Computer Science Engineering, Arizona State University, Tempe, USA*, 2003.

[48] A. Miler, "Package 'leaps'.", *Cran*, Aug. 2020. [Online]. Available: https://bit.ly/2R9VLmz.

[49] A. J. Miller, "Selection of subsets of regression variables", *Journal of the Royal Statistical Society. Series A. General*, no. 3, pp. 389–425,

[50] P. SzulcG, "The stepwise procedure for big data", *Cran*, Jul. 2020. [Online]. Available: https://bit.ly/3m9tSsS.

[51] M. Bogdan, J. Ghosh, and R. Doerge, "Modifying the schwarz bayesian information criterion to locate multiple interacting quantitative trait loci. genetics", vol. 167, pp. 989–999, 2004.

[52] S. J. Sheather, *A Modern Approach to Regression with R*. Springer, Jul. 2009.

[53] D. Phillips, "A technique for the numerical solution of certain integral equations of the first kind", *Journal of the ACM (JACM)*, vol. 9, no. 1, pp. 84–97, 1962. DOI: 10.1145/321105.321114.

[54] J. Friedman, T. Hastie, and R. Tibshirani, "Glmnet: Lasso and elastic-net regularized generalized linear models", *Cran*, [Online]. Available: https://bit.ly/2RaMGd7.

[55] P. Kenedy, *A Guide to Econometrics*, 4th. Cambridge: MIT Press, 1998.

[56] D. Friedman and D. Montgomery, "Evaluation of the predictive performance of biased regression estimators", *Journal of Forecasting*, vol. 4, no. 2, pp. 153–163, 1985. DOI: 10.1002/for.3980040205.

[57] J. Friedman, T. Hastie, and R. Tibshirani, "Regularization paths for generalized linear models via coordinate descent", *Journal of statistical software*, vol. 33, no. 1, pp. 1–22, 2010. DOI: 10.18637/jss.v033.i01.

[58] R. Tibshirani, "Regression Shrinkage and Selection Via the Lasso", *Journal of the Royal Statistical Society. Series B, Methodological*, vol. 58, no. 1, pp. 267–288, 1996. DOI: 10.1111/j.2517-6161.1996.tb02080.x.

[59] R. Tibshirani, "Regression shrinkage and selection via the lasso: A retrospective", *Journal of the Royal Statistical Society. Series B, Statistical methodology*, vol. 73, no. 3, pp. 273–282, 2011. DOI: 10.1111/j.1467-9868.2011.00771.x.

[60] H. Zou and T. Hastie, "Regression and variable election via the elastic net", *Journal of the Royal Statistical Society, Series B*, vol. 67, pp. 301–320, 2005.

[61] J. K. Tay and R. Tibshirani, "Reluctant generalized additive modeling", *Department of Statistics, and Department of Biomedical Data Science Stanford University*, 2020.

[62] R. Tibshirani and J. Tay, "Reluctant generalized additive modeling", *Cran*, Aug. 2020. [Online]. Available: https://bit.ly/2DNc45K.

[63] J. N. Jeffers, "Two case studies in the application of principal component analysis", *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, vol. 3, pp. 225–236, DOI: 10.2307/2985919.

[64] H. Hotelling, "The Relations of the Newer Multivariate Statistical Methods to Factor Analysis", *British Journal of Statistical Psychology*, vol. 10, pp. 69–79,

[65] M. G. Kendall, *A course in multivariate analysis*, 2nd. Charles Griffin, 1961.

[66] H. Artigue and G. Smith, "The principal problem with principal components regression", *Cogent Mathematics Statistics*, 2019.

[67] J.Taylor and R. Tibshirani, "Statistical learning and selective inference", *Proceedings of the National Academy of Sciences*, vol. 112, pp. 7629–7634, 2015.

[68] H. Wold, "Estimation of principal components and related models by iterative least squares", *Academic Press*, 1966.

[69] X. Shao, H. Wang, H. Wei, Z. Hou, and W. Cai, "A new regression method based on independent component analysis", *Talanta*, vol. 69, no. 3, pp. 676–680, 2006. DOI: 10.1016/j.talanta.2005.10.039.

[70] A. Hyvärinen and E. Oja, "Independent component analysis: Algorithms and applications", *Neural Networks*, vol. 13, no. 4, pp. 411–430, 2000.

[71] A. Hyvarinen, J. Karhunen, and E. Oja, "Independent component analysis", *John Wiley Sons*, 2001.

[72] A. Hyvärinen, "Fast and robust fixed-point algorithms for independent component analysis", *IEEE Transactions on Neural Networks*, vol. 10, no. 3, pp. 626–634, 1999.

[73] A. Hyvärinen and E. Oja, "A Fast Fixed-Point Algorithm for Independent Component Analysis", *Neural computation*, vol. 9, no. 7, pp. 1483–1492, 1997.

[74] H. Chun and K. Sündüz, "Sparse partial least squares regression for simultaneous dimension reduction and variable selection", *Journal of the Royal Statistical Society. Series B, Statistical methodology*, vol. 72, no. 1, pp. 3–25, 2010. DOI: 10.1111/j.1467-9868.2009.00723.x.

[75] D. Lee, W. Lee, and Y. Pawitan, "Sparse partial least-squares regression and its applications to high-throughput data analysis", *Chemometrics and Intelligent Laboratory Systems*, vol. 109, no. 1, pp. 1–8, 2011.

[76] S. Jong, "Simpls: An alternative approach to partial least squares regression", *Chemometrics and Intelligent Laboratory Systems*, vol. 18, pp. 251–263, 1993.

[77] J. H. Friedman, "Greedy Function Approximation: A Gradient Boosting Machine", *The Annals of Statistics*, vol. 29, no. 5, pp. 1189–1232, 2001.

[78] D. Gutierrez, "Gradient Boosting and XGBoost", *OpenDataScience*, Jul. 2020. [Online]. Available: https://bit.ly/2GM2X6s.

[79] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system", 2016. DOI: 10.1145/2939672.2939785.

[80] T. O. Kvålseth, "Cautionary Note about R2", *The American statistician*, vol. 39, no. 4, pp. 279–285, 1985. DOI: 10.1080/00031305.1985.10479448.

7. ANNEXES

7.1. Annex 1: Assessment for Internal Model Approach

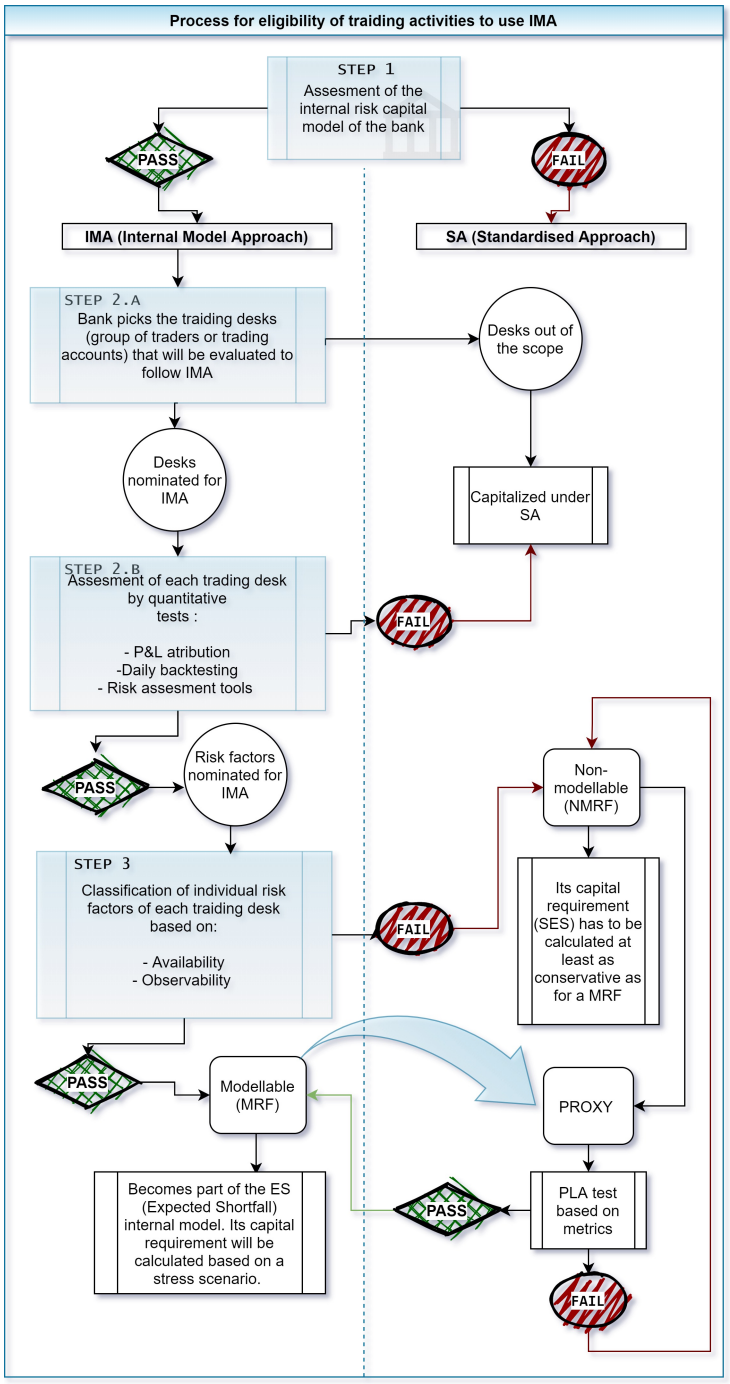

Fig. 7.1. Assessment for Internal Model Approach

7.2. Annex 2: Project Timeline

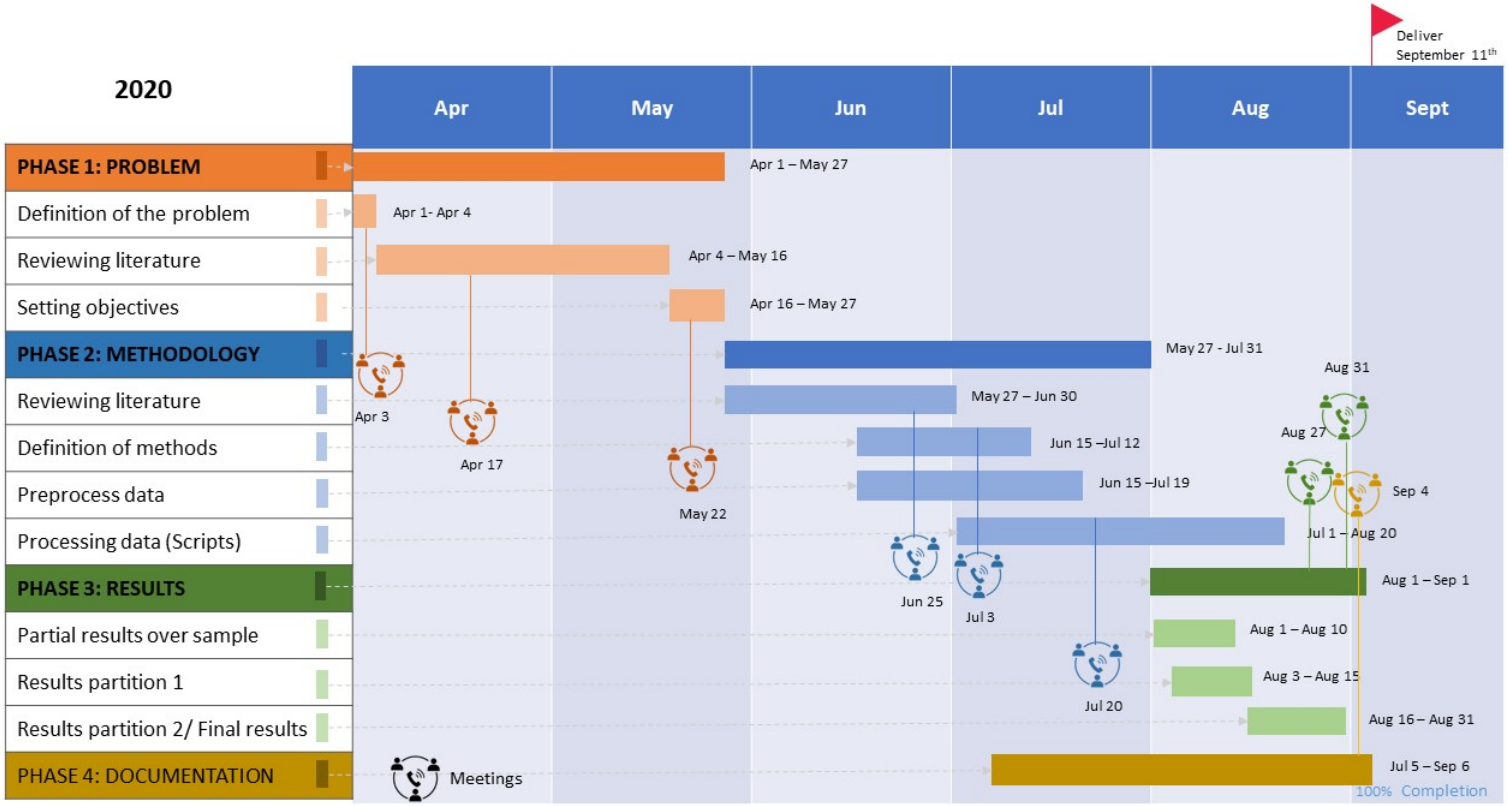

Fig. 7.2. Project Timeline

7.3. Annex 3: Feature selection methods

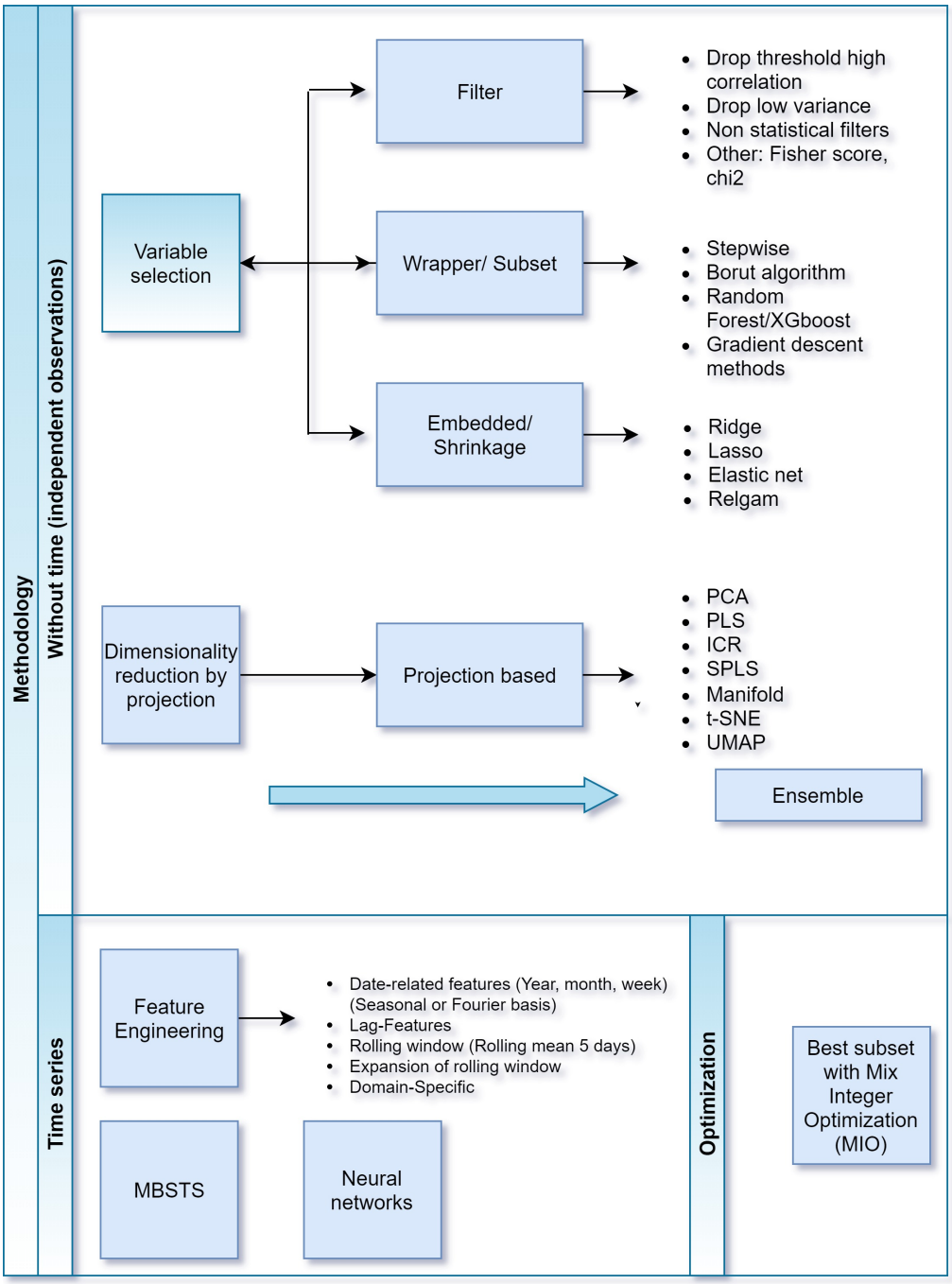

Fig. 7.3. Methods

7.4. Annex 4: Diagram of methodology for data preprocessing

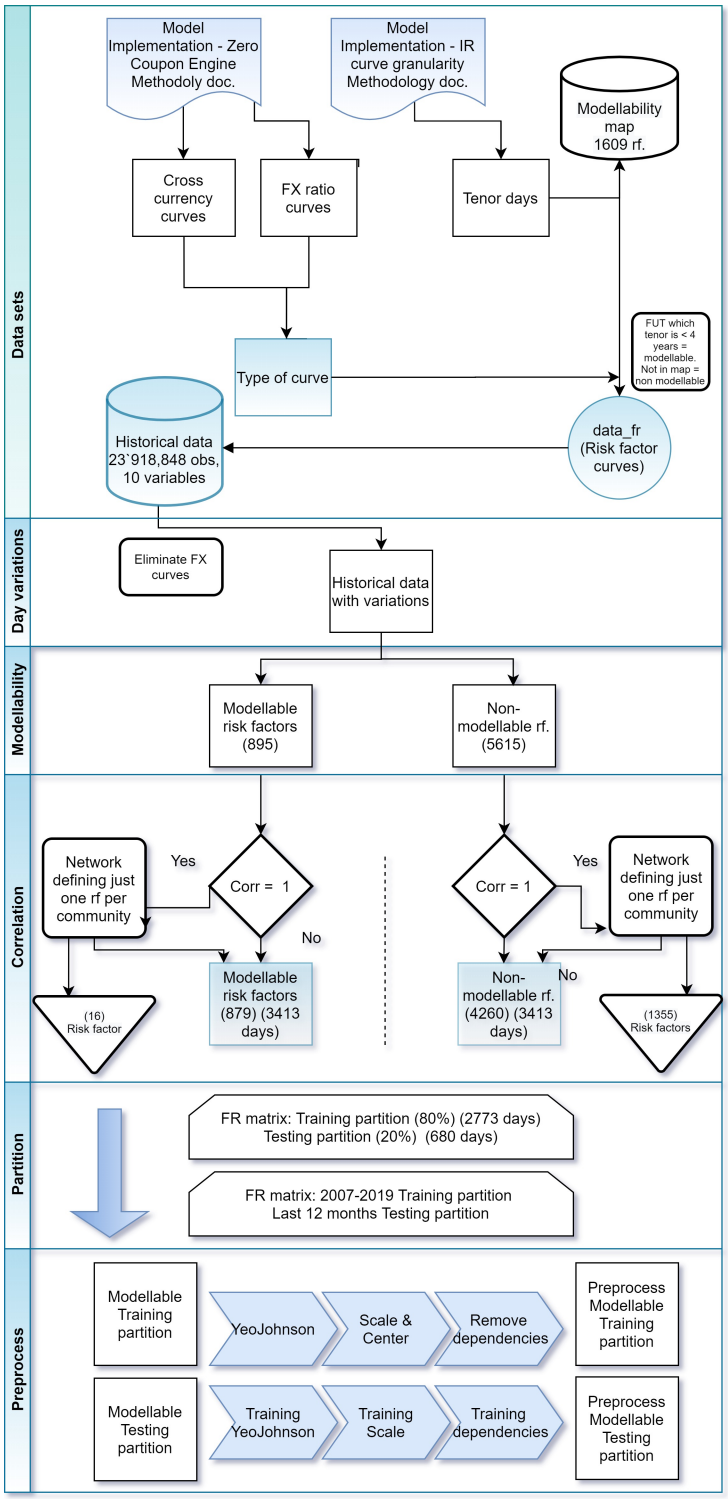

Fig. 7.4. Methodology diagram for data preprocessing

7.5. Annex 5: Programming flow

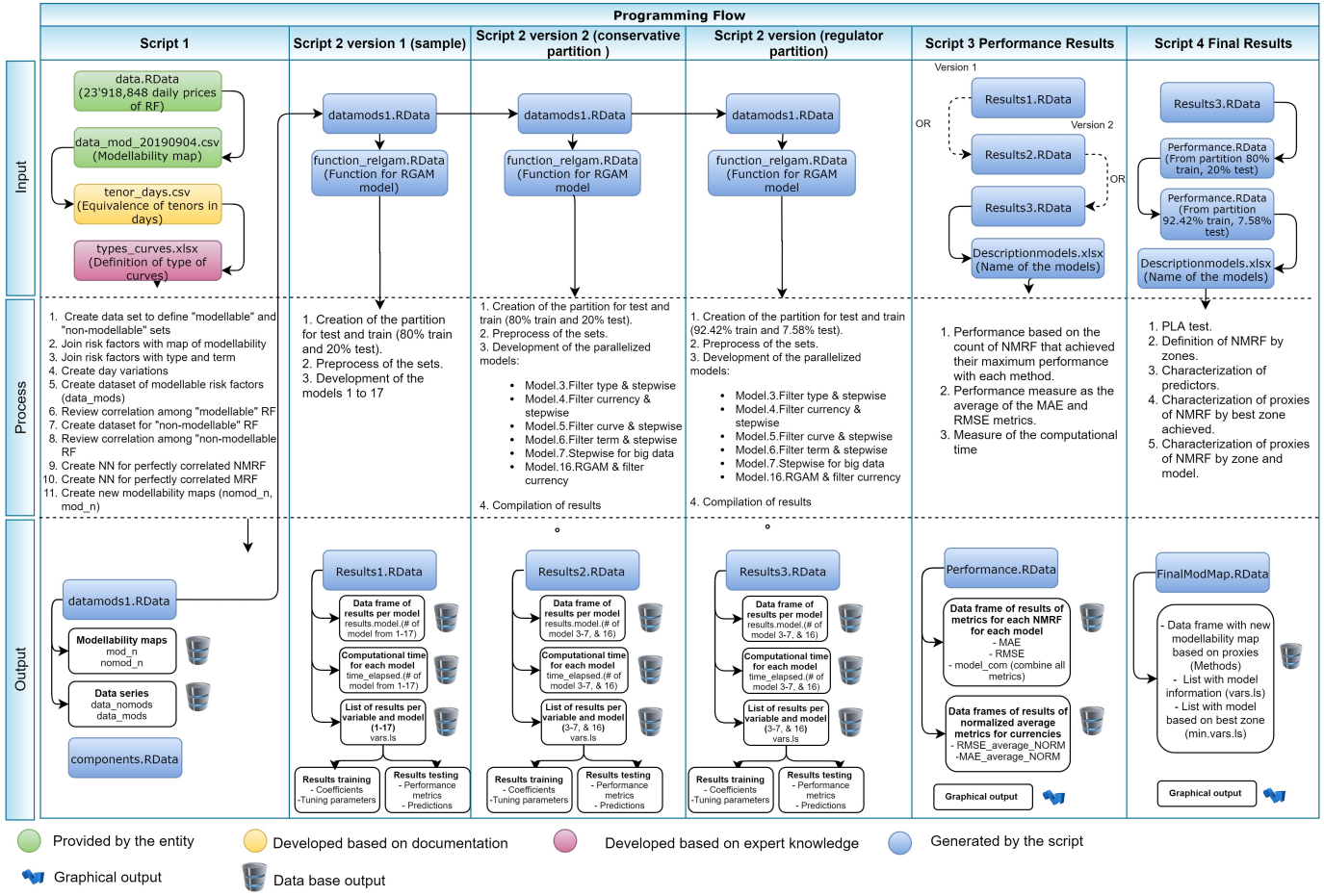

Fig. 7.5. Programming flow

7.6. Annex 6: Additional results of count of NMRFs

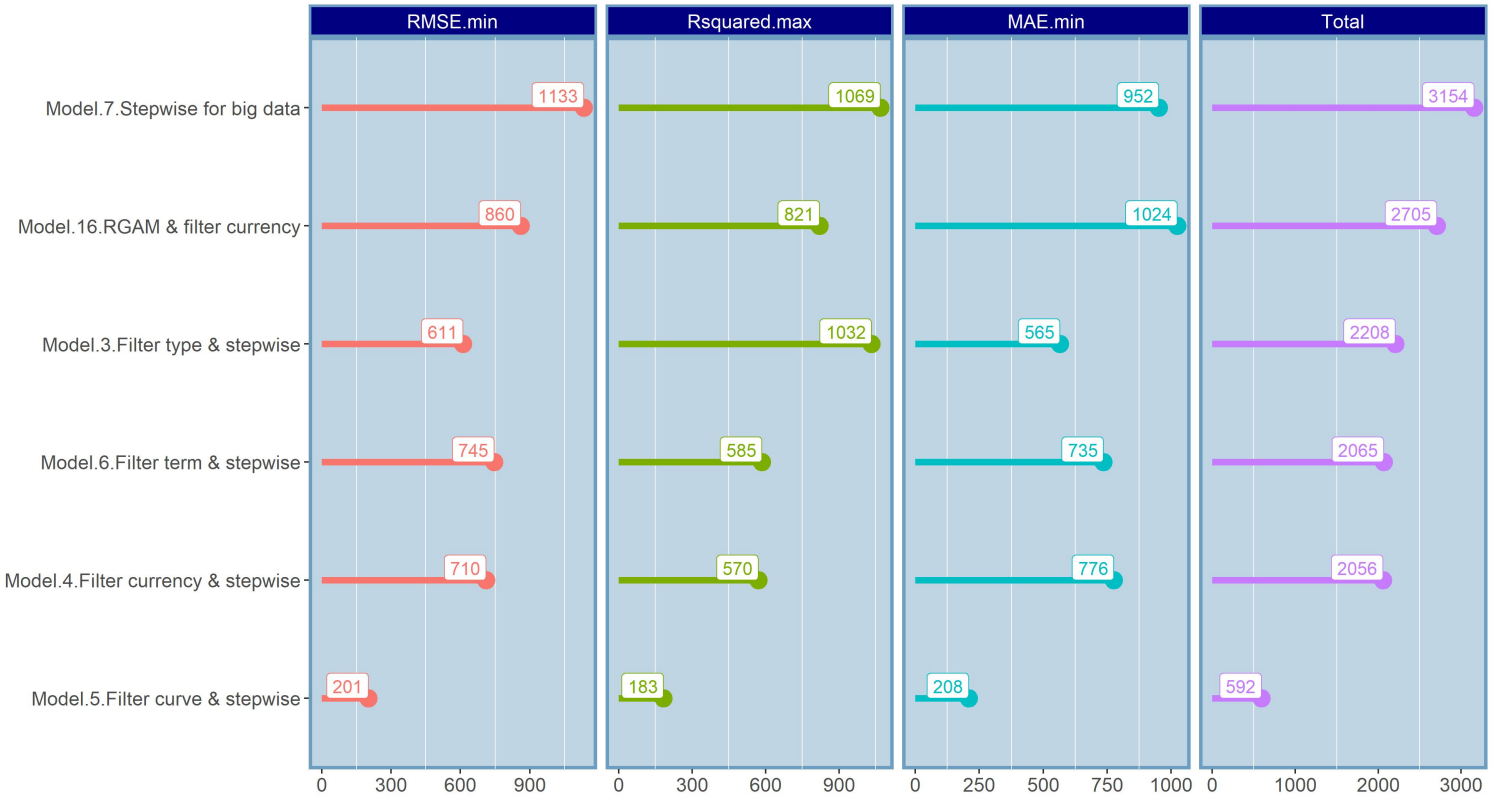

Fig. 7.6. Total results for regulator partition based on counts of NMRFs that achieved maximum performance by each method

7.7. Annex 7: Additional results of average metrics

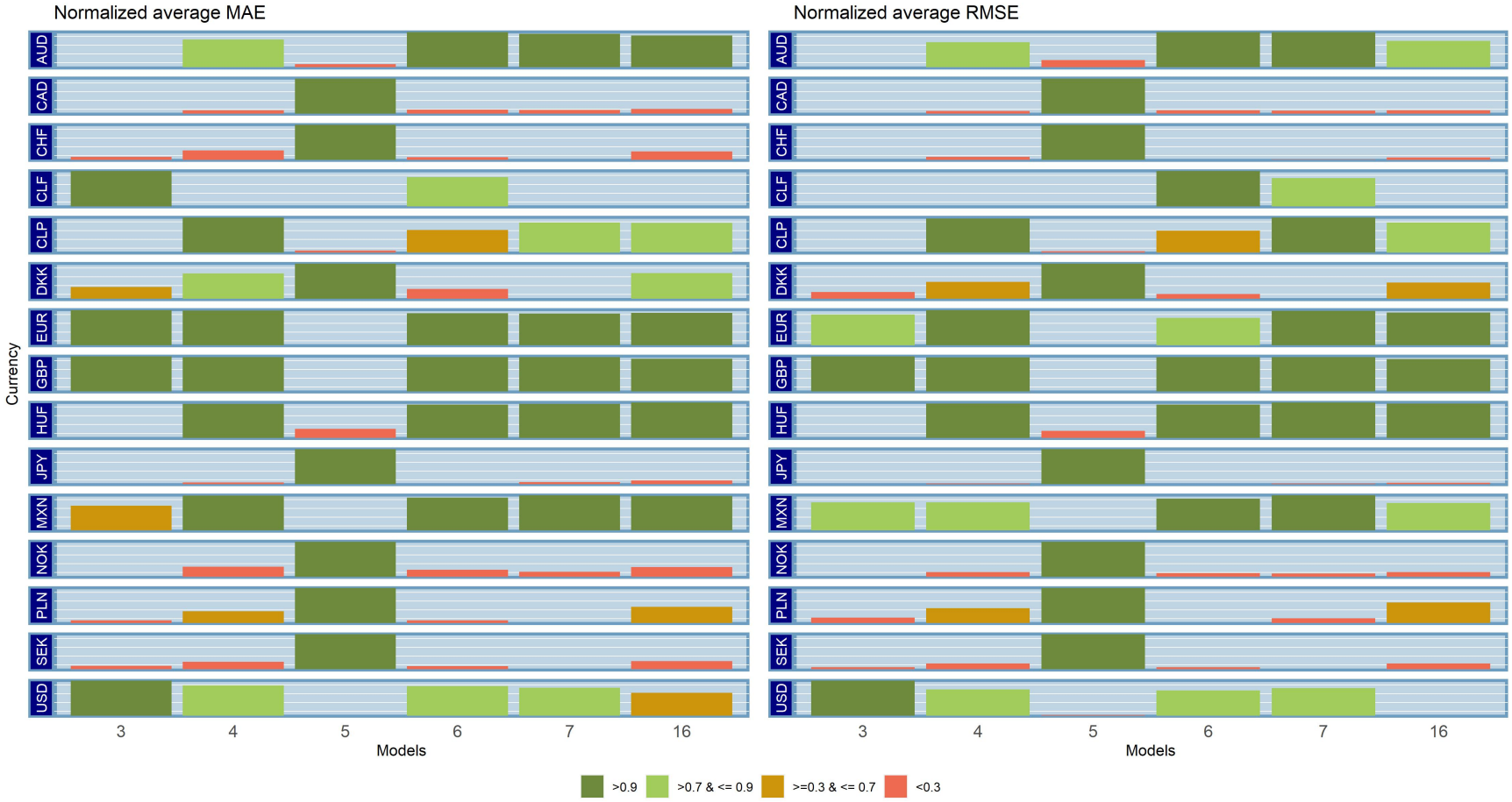

Fig. 7.7. Total results for regulator partition for models measure as normalized average MAE and RMSE per currency

7.8. Annex 8: R packages

Table 7.1. GRAPHICAL R PACKAGES

Package	Description	Documentation	Github
ggplot2	Elegant Data Visualisations Using the Grammar of Graphics	https://cran.r-project.org/web/packages/ggplot2/ggplot2.pdf	https://github.com/tidyverse/ggplot2
ggpubr	ggplot2' Based Publication Ready Plots	https://cran.r-project.org/web/packages/ggpubr/ggpubr.pdf	https://github.com/kassambara/ggpubr
ggrepel	Automatically Position Non-Overlapping Text Labels with 'ggplot2'	https://cran.r-project.org/web/packages/ggrepel/ggrepel.pdf	http://github.com/slowkow/ggrepel
ggthemes	Extra Themes, Scales and Geoms for 'ggplot2'	https://cran.r-project.org/web/packages/ggthemes/ggthemes.pdf	http://github.com/jrnold/ggthemes
treemapify	Draw Treemaps in 'ggplot2'	https://cran.r-project.org/web/packages/treemapify/treemapify.pdf	https://github.com/wilkox/treemapify

Table 7.2. STATISTICAL R PACKAGES

Package	Description	Documentation	Github
bigstep	Stepwise Selection for Large Data Sets	https://cran.r-project.org/web/packages/bigstep/bigstep.pdf	https://github.com/cran/bigstep
caret/ glmnet	Lasso and Elastic-Net Regularized Generalized Linear Models	https://cran.r-project.org/web/packages/glmnet/glmnet.pdf	https://github.com/cran/glmnet
caret/fastICA	Algorithms to Perform ICA and Projection Pursuit	https://cran.r-project.org/web/packages/fastICA/fastICA.pdf	https://github.com/cran/fastICA
caret/spls	Sparse Partial Least Squares	https://cran.r-project.org/web/packages/spls/spls.pdf	https://github.com/cran/spls
caret/xgboost	Extreme Gradient Boosting.	https://cran.r-project.org/web/packages/xgboost/xgboost.pdf	https://github.com/dmlc/xgboost
leaps	Regression Subset Selection	https://cran.r-project.org/web/packages/leaps/leaps.pdf	https://github.com/cran/leaps
relgam	Reluctant Generalized Additive Models	https://cran.r-project.org/web/packages/relgam/relgam.pdf	https://github.com/kjytay/relgam
igraph	Network Analysis and Visualization	https://cran.r-project.org/web/packages/igraph/igraph.pdf	https://github.com/igraph

Table 7.3. OTHER R PACKAGES

Package	Description	Documentation	Github
doParallel	Foreach Parallel Adaptor for the 'parallel' Package	https://cran.r-project.org/web/packages/doParallel/doParallel.pdf	https://github.com/cran/doParallel
dplyr	Grammar of Data Manipulation	https://cloud.r-project.org/web/packages/dplyr/dplyr.pdf	https://github.com/tidyverse/dplyr
Hmisc	Harrell Miscellaneous	https://cran.r-project.org/web/packages/Hmisc/Hmisc.pdf	https://github.com/harrelfe/Hmisc/
microbenchmark	Import excel files into R	https://cloud.r-project.org/web/packages/readxl/readxl.pdf	https://github.com/libxls/libxls
reshape2	Flexibly Reshape Data	https://cran.r-project.org/web/packages/reshape2/reshape2.pdf	https://github.com/hadley/reshape
scales	Scale Functions for Visualization	https://cran.r-project.org/web/packages/scales/scales.pdf	https://github.com/r-lib/scales
stringr	Simple, Consistent Wrappers for Common String	https://cloud.r-project.org/web/packages/stringr/stringr.pdf	https://github.com/tidyverse/stringr
tidyverse	Set of packages	https://cran.r-project.org/web/packages/tidyverse/tidyverse.pdf	https://github.com/tidyverse/tidyverse

7.9. Annex 9: Glossary

BCBS	Basel Committee on Banking Supervision
CC	Cross-currency curves
DHS	Hilbert-Schmidt distance
FRTB	Fundamental Review of the Trading Book
GBM	Gradient Boosting Machine
HPL	Hipothetical P&L
ICR	Independent Components Regression
IMA	Internal Model Approach
LT	Long-term
MAE	Mean Absolute Error
MBSTS	Multivariate Bayesian Structural Time Series Model
MRF	Modellable Risk Factor
PCA	Principal Components Analysis
PCR	Principal Components in Regression
PLS	Partial Least squares
RGAM	Reluctant Generalized Additive Models
RMSE	Root Mean Square Error
RTPL	Risk-theoretical P&L
SA	Standardised Approach
SC	Single Currency Curves
SPLS	Sparse Partial Least Squares
ST	Short-term