CHARLES UNIVERSITY

FACULTY OF SOCIAL SCIENCES

Center for Economic Research and Graduate Education

Dissertation Thesis

2024 **Filip Staněk**

CHARLES UNIVERSITY
FACULTY OF SOCIAL SCIENCES
Center for Economic Research and Graduate Education

Filip Staněk

Essays in Time-Series Forecasting

Dissertation Thesis

Prague 2024

Author: **Filip Staněk**

Supervisor: **doc. Stanislav Anatolyev, Ph.D.**

Year of the defense: 2024

Dissertation Committee

Anatolyev Stanislav; Chair (CERGE-EI, Prague, Czech Republic)

Mittag Nikolas (CERGE-EI, Prague, Czech Republic)

Hanousek Jan (MENDELU, Brno, Czech Republic)

Referees

John Galbraith (McGill University, Montreal, Canada)

Andrey Vasnev (University of Sydney, Sydney, Australia)

References

STANĚK, Filip. *Essays in Time-Series Forecasting*. Praha, 2024. 117 pages. Dissertation thesis (PhD). Charles University, Faculty of Social Sciences, Center for Economic Research and Graduate Education – Economics Institute. Supervisor doc. Stanislav Anatolyev, Ph.D.

Abstract

The first chapter focuses on evaluation of time-series forecasts. It is a common practice to split a time series into in-sample and pseudo out-of-sample segments and estimate the out-of-sample loss for a given statistical model by evaluating forecasting performance over the pseudo out-of-sample segment. I propose an alternative estimator of the out-of-sample loss, which, contrary to conventional wisdom, utilizes criteria measured both in- and out-of-sample via a carefully constructed system of affine weights. I prove that, provided the time series is stationary, the proposed estimator is the best linear unbiased estimator of the out-of-sample loss, and outperforms the conventional estimator in terms of sampling variability. Application of the optimal estimator to Diebold-Mariano type tests of predictive ability leads to a substantial power gain without increasing finite sample size distortions. An extensive evaluation on real world time series from the M4 forecasting competition confirms the superiority of the proposed estimator, and also demonstrates substantial robustness to violations of the underlying assumption of stationarity.

In the second chapter we perform an extensive investigation of different specifications of the BEKK-type multivariate volatility models for a moderate number of assets, focusing on how the degree of parametrization affects forecasting performance. Because the unrestricted specification may be too generously parameterized, often one imposes restrictions on coefficient matrices constraining them to have a diagonal or even scalar structure. We frame all three model variations (full, diagonal, scalar) as special cases of a ridge-type regularized estimator, where the off-diagonal elements are shrunk towards zero and the diagonal elements are shrunk towards homogeneity. Our forecasting experiments with the BEKK-type Conditional Autoregressive Wishart model for realized volatility confirm the superiority of the more parsimonious scalar and diagonal model variations. Regularization of the diagonal and off-diagonal parameters does not regularly lead to tangible performance improvements irrespective of how precise the tuning of regularization intensity is. Additionally, our results highlight the crucial importance of frequent model re-estimation in improving the forecast precision, and, perhaps paradoxically, a slight advantage of shorter estimation windows compared to longer windows.

In the third chapter I propose a novel meta-learning model that utilizes hypernetworks to design a parametric model tailored to a specific family of forecasting tasks. The model's training can be directly performed with backpropagation, eliminating the need for reliance on higher-order derivatives, and is equivalent to a simultaneous search over the space of parametric functions and their optimal parameter values. This, in essence, provides a data-driven alternative to manually designing a parametric model for a group of similar prediction tasks, an endeavor that typically requires considerable statistical expertise and domain knowledge. I demonstrate the capabilities of the proposed meta-learning model on two applications. When applied to the sinusoidal regression task, the proposed model outperforms state-of-the-art meta-learning approaches and is capable of almost perfectly recovering the underlying parametric model. In the second application, the model is applied to the time-series from the M4 forecasting competition, where it outperforms conventional time-series forecasting models designed by human experts. As a third application, I use the model to make quintile predictions for asset returns in the M6

Financial Forecasting Competition. The model attained an RPS of 0.15689, securing the 4th place in the forecasting challenge and ultimately the 1st place in the overall duathlon ranking.

Abstrakt

První kapitola se zaměřuje na vyhodnocení přesnosti předpovědí časových řad. Je běžnou praxí rozdělit časovou řadu na in-sample a pseudo out-of-sample segmenty a odhadnout out-of-sample ztrátu daného statistického modelu vyhodnocením přesnosti předpovědí v pseudo out-of-sample segmentu. V této kapitole navrhuji alternativní estimátor out-of-sample ztráty, který, na rozdíl od konvenčního estimátoru, využívá kritéria měřená jak v in-sample, tak out-of-sample prostřednictvím pečlivě konstruovaného systému afinních vah. Za předpokladu, že časová řada je stacionární, navržený estimátor je nejlepším lineárně nezkresleným estimátorem out-of-sample ztráty a předčí konvenční estimátor z hlediska vzorkové variability. Použití tohoto optimálního estimátoru pro statistické testy prediktivní schopnosti typu Diebold-Mariano vede k podstatnému zvýšení statistické síly bez zvýšení zkreslení v malých vzorcích. Rozsáhlé vyhodnocení na reálných časových řadách ze soutěže M4 potvrzuje nižší vzorkovou variabilitu navrženého estimátoru a také prokazuje značnou odolnost vůči porušení základního předpokladu stacionarity.

Ve druhé kapitole zkoumáme různé specifikace BEKK multivariačních modelů volatility pro střední počet aktiv s důrazem na to, jak stupeň parametrizace ovlivňuje kvalitu předpovědí. Vzhledem k tomu, že neomezená specifikace může být příliš štědře parametrizována, často se ukládají omezení na koeficientové matice, omezující je na diagonální nebo dokonce skalární strukturu. Všechny tři varianty modelů (plný, diagonální, skalární) formulujeme jako speciální případy estimátoru s regulací typu ridge, kde jsou prvky mimo diagonálu penalizovány směrem k nule a prvky na diagonále penalizovány směrem k homogenitě. Naše experimenty s modely typu CAW (Conditional Autoregressive Wishart) pro realizovanou volatilitu potvrzují vhodnost restriktivnějších variant modelu v podobě skalární a diagonální specifikace. Dále naše výsledky zdůrazňují klíčový význam časté re-estimace modelu pro zvýšení přesnosti předpovědí a paradoxně také mírnou výhodu kratších estimačních oken ve srovnání s delšími okny.

Ve třetí kapitole navrhuji nový meta-learning model, který využívá neuronové hyper sítě k návrhu parametrického modelu přizpůsobeného konkrétní skupině predikčních úkolů. Trénink modelu lze provádět přímo pomocí zpětné propagace, což eliminuje potřebu spoléhat se na derivace vyšších řádů, a je ekvivalentní simultánnímu prohledávání prostoru parametrických funkcí a optimálních hodnot jejich parametrů. To poskytuje alternativu k ručnímu návrhu parametrického modelu pro skupinu podobných predikčních úkolů, což obvykle vyžaduje značné statistické a doménové znalosti. Schopnosti modelu jsou demonstrovány na třech aplikacích. Při aplikaci na problém sinusové regrese, navržený model předčí všechny alternativní přístupy a dokáže téměř dokonale obnovit původní parametrický model. Při aplikaci na časové řady z M4 Forecasting Competition model dosáhl vyšší přesnosti než konvenční modely běžně používané v praxi. V rámci třetí aplikace je model používán v soutěži M6 Financial Forecasting Competition k predikci kvintilů výnosů aktiv. Zde model dosáhl přesnosti 0.15689 RPS, což zajistilo 4. místo v předpovědní výzvě a 1. místo v celkovém pořadí duatlonu.

Keywords
Time-series Forecasting, Forecasts Evaluation, Predictive Ability Testing, Multivariate Volatility, Meta-learning

Klíčová slova
Predikce Časových Řad, Evaluace Predikcí Časových Řad, Testování Predikčních Schopností, Vícerozměrná Volatilita, Meta-učení

Length of the work
160604 characters with spaces, without abstracts and appendices

Declaration

1. I hereby declare that I have compiled this thesis using the listed literature and resources only.
2. I hereby declare that my thesis has not been used to gain any other academic title.
3. I fully agree to my work being used for study and scientific purposes.

In Prague on Filip Staněk

Acknowledgement

I am deeply grateful to my supervisor Stanislav Anatolyev for teaching me not to be satisfied with a surface-level understanding of statistical methods and for the steadfast support and guidance during both successes and setbacks. His open-mindedness and willingness to entertain any idea, no matter how unconventional, fostered intellectual curiosity and ultimately made this work possible.

I would like to extend my thanks to the CERGE-EI faculty, particularly Jan Hanousek, Nicolas Mittag, and Veronika Selezneva, for their invaluable feedback. Special thanks to Deborah Novakova, Andrea Downing, and Grayson Krueger from the Academic Skills Center for their assistance with language refinement. I am also thankful for having the opportunity to meet my classmates and for the countless discussions on topics both related and unrelated to our research that we had over the years.

My sincere appreciation extends to Spyros Makridakis and the M-competition team for their pivotal role in providing a fair and transparent platform for evaluating time-series forecasting methods and their enduring commitment to pushing the boundaries of the field for over 40 years. Data from these competitions are utilized in two out of three chapters of this dissertation.

Likewise, I would like to thank Prof. John Galbraith from McGill University and Prof. Andrey Vasnev from the University of Sydney for their numerous insightful comments, which significantly improved this manuscript.

Financial support from Grantová agentura UK under grant 264120, Czech Science Foundation support under grant 20-28055S, Czech Economic Society, Nadace Nadání Josefa, Marie a Zdenky Hlávkových, and Makridakis Open Forecasting Center is gratefully acknowledged.

Finally, and most importantly, I would like to express my heartfelt gratitude to my family for their unwavering support, and to my spouse, Tereza, for her endless patience and kindness. Without you, I would have never persevered.

All remaining errors are my own.

Czech Republic, Prague
9th August 2024

Filip Staněk

Contents

Introduction

This dissertation consists of three standalone articles on time-series forecasting stapled together, each forming a chapter.

The first chapter focuses on time-series forecast evaluation. Here, I demonstrate that the conventional estimator frequently used for out-of-sample loss is generally suboptimal for stationary time-series and derive the best linear unbiased estimator of out-of-sample loss. Furthermore, I propose a modification of the Diebold-Mariano type tests of equal predictive ability which utilize this proposed estimator, achieving higher power without increasing finite sample size distortions. Simulations and evaluation on real-world time-series from the M4 Forecasting Competition confirm the superiority of the proposed estimator and tests. This work was published in the *Journal of Forecasting.*[1] To facilitate broader use of the proposed optimal estimator, I introduce an `R` software package called ACV[2], which contains a ready-to-use implementation of the estimator and tests.

The second chapter is a joint work with Stanislav Anatolyev and focuses on the task of multivariate volatility forecasting. We propose a ridge-type regularized estimator of the full BEKK/CAW model, which conveniently nests all three model variations (full, diagonal, scalar) as special cases. We perform extensive evaluation on real-world data to assess the optimal degree of regularization and confirm that the most stringent regulariza-

[1] Staněk, F. (2023) "Optimal out-of-sample forecast evaluation under stationarity", *Journal of Forecasting*, 42(8), 2249-2279.

[2] Available at: `https://CRAN.R-project.org/package=ACV`.

tion of off-diagonal elements is typically preferred in terms of forecasting performance. Additionally, our results highlight the crucial importance of frequent model re-estimation and, paradoxically, an advantage of using shorter estimation windows as opposed to longer ones. This work was published in *Studies in Nonlinear Dynamics & Econometrics.*[3] A ready-to-use implementation of the proposed regularized estimator for both BEKK and CAW is publicly available as a `MATLAB` software package *RMV*.[4]

The third chapter is concerned with meta-learning, a field that is extremely relevant for time-series forecasting, as the task of finding the most suitable parametric model for a given family of prediction tasks can be framed as a meta-learning problem. I propose a meta-learning model based on hypernetworks, whose training is equivalent to a simultaneous search over the space of parametric functions and their optimal parameter values. This approach enables the creation of a parametric model specifically optimized for a particular family of prediction tasks, so that the degrees of freedom allotted to each task are fine-tuned to best capture any heterogeneity between the tasks. The model substantially outperforms state-of-the-art meta-learning approaches on the sinusoidal regression task, a synthetic problem often used to benchmark different meta-learning approaches. To demonstrate its applicability to real-life time-series forecasting problems, I participated with the model in the M6 Financial Forecasting Competition, where it secured the 4th place in the forecasting challenge and ultimately won the 1st place in the overall duathlon ranking. This work was submitted to the special issue of the *International Journal of Forecasting* dedicated to the M6 Competition and is currently *under review*, as of the writing of this thesis. To facilitate broader use of the proposed meta-learning model, a ready-to-use implementation is provided in the `R` language.[5]

[3] Anatolyev, S., Staněk, F. (2022) "Unrestricted, Restricted, and Regularized Models for Forecasting Multivariate Volatility", *Studies in Nonlinear Dynamics & Econometrics.*

[4] Available at: `https://github.com/stanek-fi/RMV`.

[5] Available at: `https://github.com/stanek-fi/MtMs_sinusoidal_task`

Chapter 1

Optimal Out-of-Sample Forecast Evaluation Under Stationarity

Originally published as:

Staněk, F. (2023) "Optimal out-of-sample forecast evaluation under stationarity", *Journal of Forecasting*, 42(8), 2249-2279.

1.1 Introduction

In the field of time-series forecasting, researchers are typically concerned with the expected performance of a particular statistical model on yet unseen data, the so called out-of-sample loss. It is used to assess whether a proposed model statistically significantly outperforms an already established benchmark model. Likewise, in practical forecasting tasks, the out-of-sample loss is frequently used to select a model that is likely to deliver the best forecasting performance from a set of competing models.

Out-of-sample loss is defined as the expected value of a contrast function that measures the discrepancy between the prediction and the observed value (e.g., the expected value of squared error). Thus, it is by definition unknown and needs to be estimated. This is typically achieved by excluding the most recent segment of the observed time-series from the estimation and performing a sequence of predictions for these observations instead, essentially mimicking the process of actual out-of-sample forecasting.[1] The estimate of the out-of-sample loss is then obtained simply by averaging the precision of individual predictions as measured by the contrast function, i.e., the so called empirical contrasts (e.g. squared errors). While there are many such pseudo out-of-sample evaluation schemes (for a survey, see Tashman, 2000), we restrict our attention to two prominent variants; the rolling scheme and the fixed scheme. When performing an evaluation under the rolling scheme, the model is repeatedly estimated on a rolling window of a fixed length and predictions are made for the subsequent observations. In the fixed scheme, the model is estimated only once on the first segment of the data and is then used to predict all remaining observations (see e.g. Clark and McCracken, 2013b).

A common drawback of all such pseudo out-of-sample evaluation schemes and corresponding estimators is the relatively high sampling variance, as the estimate is computed based on only a relatively few most recent observations reserved for the pseudo out-of-sample evaluation (Bergmeir and Benítez, 2012; Bergmeir et al., 2014; Schnaubelt, 2019; Cerqueira et al., 2020). Moreover, this issue of scarcity of pseudo out-of-sample observations and consequently of high sampling variance is not limited to situations with few observations, but also afflicts longer time-series. This is because there is an inevitable trade-off between the size of the data-sets designated to be in-sample and pseudo out-of-sample. The former allows for a more faithful approximation of the loss when the whole data-set is used for estimation, whilst the latter allows for more precise estimation of the loss (see Arlot and Celisse, 2010).

To alleviate this issue, we propose an alternative estimator of the out-of-sample loss that utilizes in-sample performance to aid the estimation of the out-of-sample loss, a practice often considered taboo in the forecasting community. In particular, we use in-sample empirical contrasts to partially eliminate the idiosyncratic noise present in observations designated for the out-of-sample evaluation, via a carefully constructed system of optimal affine weights. We prove that, under stationarity, the proposed estimator of the out-of-sample loss is optimal in terms of the sampling variance within the class of unbiased linear estimators, to which the conventional estimator also belongs. The proposed estimator hence offers a lower sampling variance relative to the conventional estimator, all without

[1] There is another class of evaluation schemes that do not respect the temporal ordering of the data and perform out-of-sample evaluation not dissimilar to the canonical cross-validation for independent processes, see e.g., Burman et al. (1994), Racine (2000), and Bergmeir et al. (2018). However, these are not as widely used in practice and hence are not considered in this chapter.

introducing any bias. In turn, this allows for a finer assessment of forecasting ability, more powerful inference about predictive ability, and more precise model selection.

The proposed optimal estimator is obtained by finding weights that minimize the sampling variance, subject to constraints that guarantee unbiasedness. Importantly, both in- and out-of-sample contrasts can be included with non-zero weights, and weights are allowed to be negative, unlike for the conventional estimator, which simply places equal positive weights only on out-of-sample contrasts. In practice, this translates to assigning negative weights to in-sample empirical contrasts that are positively correlated with out-of-sample empirical contrasts, and positive weights to in-sample empirical contrasts that are uncorrelated with out-of-sample empirical contrasts. At the same time, sums of weights of ex-ante identical in-sample contrasts are equal to zero, which ensures that the inclusion of in-sample contrasts does not alter the expected value of the estimator, and hence does not introduce bias. From a more general standpoint, the possibility to reduce the sampling variance arises because time-series out-of-sample evaluation schemes are inherently unbalanced in the sense of Shao (1993). That is, these schemes generally do not treat observations equally in terms of in-/out-of-sample usage. The proposed optimal weighting partially rectifies this unbalanced design.

Aside from the optimal estimator itself, we also propose modifications of the canonical Diebold-Mariano test (Diebold and Mariano, 1995) and of the sub-sampling test of equal predictive ability (Zhu and Timmermann, 2020; Ibragimov and Müller, 2010). Both modified tests leverage the proposed optimal weighting for estimation of the loss differential. We show that these tests are asymptotically valid and demonstrate that they exhibit a substantially higher power in detecting deviations from the null hypothesis of equal predictive ability relative to their respective benchmarks.

Finally, to assess the real-life applicability and the robustness of the proposed estimator, we perform an extensive evaluation on 100,000 time-series from the M4 forecasting competition (Makridakis et al., 2020) ranging from yearly to hourly frequency. The proposed estimator delivers more than a 10% reduction in the mean squared error relative to the conventional estimator when tasked with predicting the incurred loss on the test segments of time-series. Moreover, when selecting the model by comparing estimated losses, the proposed optimal estimator is more likely to select the best performing model and delivers a smaller overall incurred loss. Importantly, in this evaluation, we include time-series from the M4 competition without specifically selecting for stationarity. Many series in this dataset exhibit trends or seasonality, or both. The proposed estimator works well in this diverse setting, outperforming the conventional estimator across various types of time-series, though it should be noted that the gains are less sizable if the assumption of stationarity is violated. This demonstrates that the theoretical advantages of the proposed estimator translate effectively to practical forecasting applications, showcasing its ability to handle the complexities inherent in real-world time-series data.

Section 1.2 introduces the statistical framework and provides formal definitions of out-of-sample evaluation schemes and corresponding estimators. Section 1.3 introduces the proposed estimator of the out-of-sample loss, proves its optimality, and demonstrates its efficiency gains in a simulated environment. Section 1.4 introduces modified tests of equal out-of-sample predictive ability that utilize the optimal estimator, and demonstrates their power advantage relative to benchmarks. Section 1.5 compares the performance of the conventional estimator and the proposed optimal estimator on real world time-series from the M4 forecasting competition. Section 1.6 concludes. Sections 1.7, 1.8, 1.9, and Appendix 1.A contain proofs, estimators, algorithms, and supplementary results, respectively. A ready-to-use implementation of the estimator and tests is provided as an R software package *ACV*[2].

1.2 Conventional Estimator of the Loss

We follow the notation of Arlot and Celisse (2010). Consider a sequence $\{X_t\}_1^T \in \mathbb{R}^T$ from a stationary random process X_t for a given $T \in \mathbb{N}$. A statistical model $\mathcal{M} = \{s, \widehat{\theta}\}$ is composed of two functions. The estimator $\widehat{\theta} : \cup_{m\in\mathbb{N}}\mathbb{R}^m \to \Theta$, which takes sequence $\{X_t\}_1^m$ of length m and outputs model parameters θ belonging to the parameter space Θ, and the forecasting function $s : \{\mathbb{R}^k; \Theta\} \to \mathbb{R}$, which predicts the observation $X_{k+\tau}$ based on most recent observations $\{X_t\}_{t=1}^k$ where k is the memory of the model and τ is the forecast horizon.[3] While we refer to $\{s, \hat{\theta}\}$ simply as the *model*, it is important to note that this framework can also accommodate more complex procedures or algorithms. Notably, $\{s, \hat{\theta}\}$ can represent a combination of multiple models. In this case, s is a linear combination of

[2] Available at: `https://CRAN.R-project.org/package=ACV`.

[3] To facilitate the exposition, we take the liberty of representing the model as a prediction and estimation function pair $\mathcal{M} = \{s, \widehat{\theta}\}$ rather than a single function $\mathcal{A}$ representing a statistical algorithm as in Arlot and Celisse (2010), hence focusing on parametric models. All results can nonetheless be extended to non-parametric models by using the identity $\mathcal{A}\big(\{X_t\}_1^m\big)\big(\{X_t\}_{j-k-\tau+1}^{j-\tau}\big) = s\big(\{X_t\}_{j-k-\tau+1}^{j-\tau}; \widehat{\theta}\big(\{X_t\}_1^m\big)\big)$.

prediction functions, and the parameter vector $\hat{\theta}$ concatenates individual model parameters and forecast combination weights, if they are data-dependent. This flexibility is crucial, as forecast combinations have shown remarkable success in empirical applications, often outperforming their constituent models (for a comprehensive review, see Wang et al., 2023). To assess the quality of a model $\mathcal{M}$, we use a contrast function $\gamma : \{\mathbb{R},\, \mathbb{R}\} \to \mathbb{R}$ that measures the discrepancy between a prediction $\hat{X}_{k+\tau} = s(\{X_t\}_{t=1}^{k}\,,\, \theta)$ and the actual realization of the process $X_{k+\tau}$. For instance, a simple AR(k) model would correspond to $\widehat{X}_{k+1} = s(\{X_t\}_1^k\,;\, \widehat{\theta}) = \sum_1^k X_k \widehat{\theta}_k$ where $\widehat{\theta}$ is the corresponding OLS estimator. Contrast function is typically a squared error in which case $\gamma\Big(X_{k+1},\, \widehat{X}_{k+1}\Big) = \Big(X_{k+1} - \widehat{X}_{k+1}\Big)^2$.[4]

Finally, let us denote the loss[5] of model $\mathcal{M} = \{s,\, \widehat{\theta}\}$ when estimated on a sequence of length m and when faced with forecasting the period $j > m$ using observations $\{X_t\}_{j-k-\tau+1}^{j-\tau}$ as

$$\mathcal{L}_j^m\left(\mathcal{M}\right) = \mathbb{E}\left[\gamma\left(X_j, s\left(\{X_t\}_{j-k-\tau+1}^{j-\tau}\,; \widehat{\theta}\left(\{X_t\}_1^m\right)\right)\right)\right]. \tag{1.1}$$

Note that the expectation is taken over the whole segment $\{X_t\}_1^j$, i.e., both the forecasted observation X_j and its predecessors, including the estimation window $\{X_t\}_1^m$. We are therefore interested in the performance of model $\mathcal{M}$ rather than that of some particular forecasting function $s\Big(\{X_t\}_{j-k-\tau+1}^{j-\tau}\,; \theta_0\Big)$ with fixed $\theta_0 \in \Theta$ (i.e., Question 6 from Dietterich's (1998) taxonomy).

Further, for a "shifting" index $i : 0 \leq i \leq T - m$, we also denote the out-of-sample empirical contrast of model $\mathcal{M}$ when estimated on a sequence $\{X_t\}_{i+1}^{i+m}$ and evaluated at the $(i+j)$-th period with $j > m$ as

$$l_j^{m,i}\left(\mathcal{M}\right) = \gamma\left(X_{i+j}, s\left(\{X_t\}_{i+j-k-\tau+1}^{i+j-\tau}\,; \widehat{\theta}\left(\{X_t\}_{i+1}^{i+m}\right)\right)\right). \tag{1.2}$$

[4] Throughout the text, we focus exclusively on univariate point prediction for the sake of simplicity. The framework can be however readily extended to a general d-variate prediction problem by considering $s : \{\left(\mathbb{R}^d\right)^k;\, \Theta\} \to \Psi$ and $\gamma : \{\mathbb{R}^d,\, \Psi\} \to \mathbb{R}$ where Ψ represents the space of possible predictions. For instance, in the case of univariate conditional density forecasting, a model $\mathcal{M}$ is a class of densities with a corresponding estimator $\widehat{\theta}$ for its parameters, set Ψ is a space of density functions and $\psi(q) = s\big(\{X_t\}_1^k;\, \widehat{\theta}\big)(q) = \widehat{f}\big(q|\{X_t\}_1^k; \widehat{\theta}\big)$ is the predicted density at point q. One may take $\gamma\big(X_{k+\tau}, \psi\big) = -\ln\big(\psi\big(X_{k+\tau}\big)\big) = -\ln\big(\widehat{f}\big(X_{k+\tau}|\{X_t\}_1^k;\, \widehat{\theta}\big)\big)$ to obtain the Kullback-Leibler divergence (Kullback and Leibler, 1951) as a measure of precision.

[5] In certain research domains, the contrast function is known as the *loss function*, while loss is referred to as *risk* (see e.g., Chen and Liu, 2023). For consistency, we adhere to the former terminology in this chapter.

The assumption of stationarity then immediately implies

$$\mathbb{E}\left[l_j^{m,i}(\mathcal{M})\right] = \mathcal{L}_j^m(\mathcal{M}). \tag{1.3}$$

In this text, we focus on the pseudo out-of-sample evaluation with step-size v (see e.g., Callen et al. (1996) and Swanson and White (1997)). The procedure is as follows. The model is estimated on a segment of data of length m and forecasts are iteratively made on v consecutive periods for which empirical contrasts are recorded. After that, the estimation window is moved forward by v, and the process is repeated until the end of the sample is reached. The estimate of the out-of-sample loss is then computed simply by averaging all pseudo out-of-sample empirical contrasts incurred. Figure 1.1a provides a diagram of such a procedure. More formally, the estimator is expressed as[6]

$$\widehat{\mathcal{L}}_{CV} = \frac{1}{n}\sum_{i=1}^{n/v}\sum_{j=1}^{v} l_{m+j}^{m,\,(i-1)v} \tag{1.4}$$

where $n \equiv T - m$ is the number of observations designated for the pseudo out-of-sample evaluation.[7] This specification nests the two most common variants of pseudo out-of-sample evaluation. By setting $v = n$, we obtain the fixed scheme evaluation, which is popular because of its low computational requirements and simplicity. On the other hand, by setting $v = 1$, we obtain the rolling scheme evaluation, which requires repeated re-estimations, but is presumably more theoretically appealing (Swanson and White, 1997).

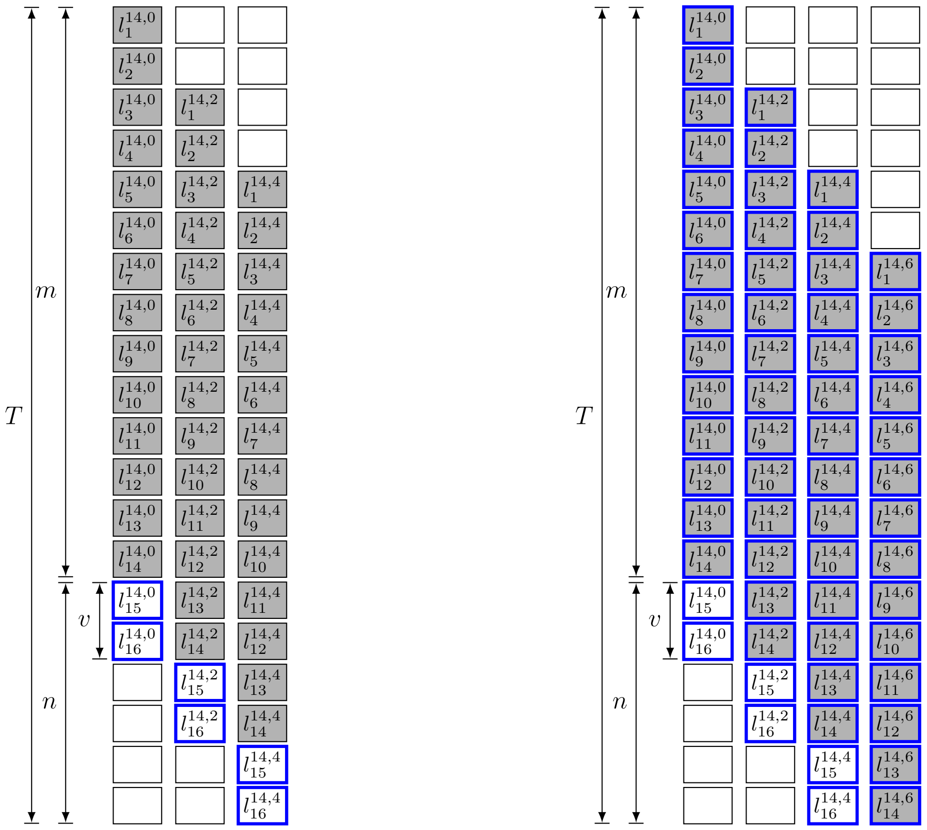
From Eq. 1.3, it follows that

$$\mathbb{E}\left[\widehat{\mathcal{L}}_{CV}\right] = \frac{1}{v}\sum_{j=1}^{v}\mathcal{L}_{m+j}^{m} \equiv \mathcal{L}_{CV} \tag{1.5}$$

where $\mathcal{L}_{CV}$ is the quantity of interest. Note that $\mathcal{L}_{CV}$ depends not only on model $\mathcal{M}$ but also τ, v, and m. Indeed, different losses $\mathcal{L}_{CV}$ might be relevant to different applications, depending on the desired horizon, the ability to update the model, and the length of the available data. However, irrespective of the particular $\mathcal{L}_{CV}$ to be estimated, we show that the conventional estimator $\widehat{\mathcal{L}}_{CV}$ is sub-optimal for that task. In the next section, we derive

[6] Due to space considerations, we omit $\mathcal{M}$ from the argument of empirical contrasts, losses, and estimators when it causes no confusion.

[7] Throughout this text, we assume that n is divisible by v, i.e. $n \bmod v = 0$.



T
m
n
v
$l_1^{14,0}$ … $l_{16}^{14,0}$
$l_1^{14,2}$ … $l_{16}^{14,2}$
$l_1^{14,4}$ … $l_{16}^{14,4}$
T
m
n
v
$l_1^{14,0}$ … $l_{16}^{14,0}$
$l_1^{14,2}$ … $l_{16}^{14,2}$
$l_1^{14,4}$ … $l_{16}^{14,4}$
$l_1^{14,6}$ … $l_{14}^{14,6}$

(a) Conventional estimator $\widehat{\mathcal{L}}_{CV}$.

(b) Optimal estimator $\widehat{\mathcal{L}}_{ACV}$.

Figure 1.1: A diagram illustrating estimators of the out-of-sample loss. The example is for $T = 20$ observations, length of the estimation window $m = 14$, and step size $v = 2$. The gray background indicates whether the observation X_t is used in the estimation of parameters θ. The blue outline indicates whether the empirical contrast $l_j^{m,i}$ is used when computing the estimate of the out-of-sample loss.

the optimal estimator of $\mathcal{L}_{CV}$ which, under the assumption of stationarity, outperforms the conventional estimator in terms of the sampling variance while retaining its unbiasedness.

1.3 Optimal Estimator of the Loss

Analogically to out-of-sample empirical contrasts, in-sample empirical contrasts can be expressed as

$$l_j^{m,i}\left(\mathcal{M}\right) = \gamma\left(X_{i+j}, s\left(\left\{X_t\right\}_{i+j-k-\tau+1}^{i+j-\tau}; \widehat{\theta}\left(\left\{X_t\right\}_{i+1}^{i+m}\right)\right)\right) \tag{1.6}$$

with the only difference being that $j \leq m$.[8] To construct the optimal estimator, we leverage two facts. First, the correlation between out-of-sample contrast $l_j^{m,\,i}$ and in-sample contrast $l_{j'}^{m,\,i'}$ varies, generally being the strongest when $j + i = j' + i'$, i.e. when the in-sample empirical contrast is computed from the same observation X_{i+j} as the out-of-sample contrast, and hence is influenced by the same idiosyncratic noise. Second, for any pair i and i' it holds that $\mathbb{E}[l_j^{m,\,i}] = \mathbb{E}[l_j^{m,\,i'}]$. Consequently, we can construct affine combinations of in-sample contrasts $l_j^{m,\,i}$, which are of zero mean, but are still negatively correlated with $\widehat{\mathcal{L}}_{CV}$, and whose inclusion hence reduces the sampling variance without introducing any bias. Figure 1.1b provides a diagram of such a procedure.

To provide a precise description of how such affine combinations should be obtained, we denote the vector of in-sample and out-of-sample contrasts of a model estimated on $\{X_t\}_{i+1}^{i+m}$ by $\boldsymbol{l}_{in}^{m,\,i}$ and $\boldsymbol{l}_{out}^{m,\,i}$ respectively, i.e.

$$\boldsymbol{l}_{in}^{m,\,i} = \left(l_1^{m,\,i},\, l_2^{m,\,i}, \ldots,\, l_m^{m,\,i}\right)^\top \tag{1.7}$$

$$\boldsymbol{l}_{out}^{m,\,i} = \left(l_{m+1}^{m,\,i},\, l_{m+2}^{m,\,i}, \ldots,\, l_{m+v}^{m,\,i}\right)^\top . \tag{1.8}$$

We can then collect all measured in-sample and out-of-sample contrasts across different window locations i to a single column vector ϕ, i.e.

$$\phi = \left(\begin{pmatrix} \boldsymbol{l}_{in}^{m,\,0v} \\ \boldsymbol{l}_{out}^{m,\,0v} \end{pmatrix}^\top, \begin{pmatrix} \boldsymbol{l}_{in}^{m,\,1v} \\ \boldsymbol{l}_{out}^{m,\,1v} \end{pmatrix}^\top, \ldots, \begin{pmatrix} \boldsymbol{l}_{in}^{m,\,(\frac{n}{v}-1)v} \\ \boldsymbol{l}_{out}^{m,\,(\frac{n}{v}-1)v} \end{pmatrix}^\top, \left(\boldsymbol{l}_{in}^{m,\,n}\right)^\top \right)^\top . \tag{1.9}$$

Throughout this chapter, we consider estimators linear in measured empirical contrasts, i.e.

$$\lambda^\top \phi \quad \text{with} \quad \lambda \in \mathbb{R}^{\text{card}(\phi)} \tag{1.10}$$

where, following the work of Lavancier and Rochet (2016) on optimal weighting of estimators, λ is a vector of weights for individual elements of ϕ. Note that the conventional

[8] All propositions bellow remain valid even if the definition in Eq. 1.6 is replaced with a measurable model-specific function $\kappa_j\big(\{X_t\}_{i+1}^{i+m}\big)$ proxying the in-sample contrasts as defined in Eq. 1.6. This allows us to also consider applications in which the forecasting function s uses all available observations up to $X_{j-\tau}$ in order to predict X_j, i.e., when $k = m$.

estimator $\widehat{\mathcal{L}}_{CV}$ can likewise be expressed as in Eq. 1.10; by defining[9]

$$\lambda_{CV\,q} = \begin{cases} \dfrac{1}{n} & \text{for } q \text{ coresponding to elements } l_j^{m,\,iv} \text{ with } 0 \leq i \leq \frac{n}{v} \text{ and } j > m \\ 0 & \text{otherwise} \end{cases} \tag{1.11}$$

it follows that

$$\widehat{\mathcal{L}}_{CV} = (\lambda_{CV})^\top \phi. \tag{1.12}$$

This automatically poses the question of whether the vector of weights λ_{CV} is optimal in terms of mean squared error

$$\mathbb{E}\left[\left(\lambda^\top \phi - \mathcal{L}_{CV}\right)^2\right] = \lambda^\top \Sigma_\phi \lambda \tag{1.13}$$

where

$$\Sigma_\phi = \mathbb{E}\left[(\phi - \mathcal{L}_{CV}\mathbf{1}_{\mathrm{card}(\phi)})(\phi - \mathcal{L}_{CV}\mathbf{1}_{\mathrm{card}(\phi)})^\top\right]. \tag{1.14}$$

In the following proposition, we derive the optimal linear unbiased estimator of $\mathcal{L}_{CV}$ (denoted by $\widehat{\mathcal{L}}_{ACV^*}$ where the "A" stands for affine) and show that the conventional estimator $\widehat{\mathcal{L}}_{CV}$ is generally not optimal.

Proposition 1. *Let $\{X_t\}$ be a stationary process and let V_ϕ be a positive definite covariance matrix of vector ϕ. It then holds that the set of all linear estimators of $\mathcal{L}_{CV}$ that are guaranteed to be unbiased is given as*

$$\mathbb{E}[\lambda^\top \phi] = \mathcal{L}_{CV} \qquad \Longleftrightarrow \qquad \lambda \in \Lambda_{ACV} \equiv \left\{ x \in \mathbb{R}^{\mathrm{card}(\phi)} \middle| Bx = b \right\} \tag{1.15}$$

with

$$B = \left(\mathbf{1}_{n/v}^\top \otimes I,\, I_{:,M}\right) \qquad b = \begin{pmatrix} \mathbf{0}_m \\ \frac{1}{v}\mathbf{1}_v \end{pmatrix} \tag{1.16}$$

where $M = (1,\, 2,\, \ldots,\, m)$. Furthermore, for estimator

$$\widehat{\mathcal{L}}_{ACV^*} = (\lambda_{ACV})^\top \phi \qquad \textit{with} \qquad \lambda_{ACV} = V_\phi^{-1} B^\top \left(B V_\phi^{-1} B^\top\right)^{-1} b \tag{1.17}$$

[9] We follow convention and denote q-th element of vector a by a_q and the row (resp. column) subset of matrix A by $A_{Q,:}$ (resp. $A_{:,Q}$) where Q is the set of indices to be kept. Furthermore, we denote the identity matrix by I and column vectors of ones (resp. zeroes) of length k by $\mathbf{1}_k$ (resp. $\mathbf{0}_k$).

it holds that

$$\mathbb{E}\left[\widehat{\mathcal{L}}_{ACV^*}\right] = \mathcal{L}_{CV}, \tag{1.18}$$

$$Var\left(\widehat{\mathcal{L}}_{ACV^*}\right) < Var\left(\lambda^\top \phi\right) \qquad \textit{with} \qquad \lambda \in \Lambda_{ACV},\, \lambda \neq \lambda_{ACV}, \tag{1.19}$$

and also

$$Var\left(\widehat{\mathcal{L}}_{ACV^*}\right) \leq Var\left(\widehat{\mathcal{L}}_{CV}\right). \tag{1.20}$$

In Proposition 1, we first show that, for all linear unbiased estimators, it holds that $\lambda \in \Lambda_{ACV}$. We then derive the variance minimizing weights λ_{ACV} within Λ_{ACV}. The corresponding optimal estimator $\widehat{\mathcal{L}}_{ACV^*} = (\lambda_{ACV})^\top \phi$ is preferred to the conventional estimator $\widehat{\mathcal{L}}_{CV}$ as it is also unbiased and $Var\Big(\widehat{\mathcal{L}}_{ACV^*}\Big) \leq Var\Big(\widehat{\mathcal{L}}_{CV}\Big)$.

The term *affine* is key here, as the inclusion of negative weights is fundamental to reducing sampling variance. Under non-negativity constraints, it can be shown that the variance-minimizing weights (given unbiasedness) would simply reduce to the conventional loss estimator, $\widehat{\mathcal{L}}_{CV}$. While, to our knowledge, there is no comprehensive analysis regulating the use of negative weights in optimal estimator weighting, we can draw insights from the closely related field of forecast combination (for a review, see Wang et al., 2023). Notably, Radchenko et al. (2023) show that negative weights primarily arise in situations where forecasts are strongly positively correlated and that the benefit of relaxing the non-negativity constraint lies in the ability of the resulting combined forecast to extend beyond the range of candidate forecasts. This same mechanism drives the benefit of optimal weighting in the case of $\widehat{\mathcal{L}}_{ACV^*}$ as well.

To illustrate this concept, consider the simplest possible example: a time series with only two observations. In this scenario, the model is estimated on the first observation X_1 and evaluated on the second observation X_2. We have three recorded contrasts: the in-sample contrast $l_1^{1,0}$, the out-of-sample contrast $l_2^{1,0}$, and the in-sample contrast computed with the shifted window, $l_1^{1,1}$. The condition of unbiasedness (the constraints in Eq 1.16) implies that $\lambda[2]$, corresponding to $l_2^{1,0}$, equals 1, and that for the remaining two weights, $\lambda[1]$ and $\lambda[3]$, corresponding to $l_2^{1,0}$ and $l_1^{1,1}$ respectively, we have $\lambda[1] + \lambda[3] = 0$. Consequently, all linear unbiased estimators $\widetilde{\mathcal{L}}_{CV}$ can be expressed, without loss of generality, as combinations of two components:

$$\widetilde{\mathcal{L}}_{CV} = 1 * l_2^{1,0} + \lambda[3](l_1^{1,1} - l_1^{1,0}) \qquad \text{with} \qquad \lambda[3] \in \mathbb{R}. \tag{1.21}$$

Typically, we observe that $cov(l_2^{1,0}, l_1^{1,1} - l_1^{1,0}) > 0$, as $l_2^{1,0}$ and $l_1^{1,1}$ are both computed from the same observation X_2. Note that $\mathbf{E}[l_2^{1,0}] = \mathcal{L}_{CV}$ and $\mathbf{E}[l_1^{1,1} - l_1^{1,0}] = 0$. To stabilize the estimator by exploiting this correlation, it hence becomes necessary to allow $\widetilde{\mathcal{L}}_{CV}$ to lie outside the range of both components $l_2^{1,0}$ and $l_1^{1,1} - l_1^{1,0}$ in approximately half the cases (assuming symmetric distributions), mirroring the case of forecast combination.

It is also worth noting that the efficiency gains do not necessarily stem from the stationarity per se, but rather from the existence of some partition (in addition to the partition of singletons) of vector ϕ where contrasts within components of that partition share a common mean. Consequently, analogous estimators can also be constructed for non-stationary series, provided that there is such a partition, i.e., as long as there is at least some degree of regularity. For example, by partitioning ϕ so $l_j^{m,\,iv}$ and $l_{j'}^{m,\,i'v}$ share a common component of the partition if and only if $j = j'$ and both contrasts are from the same day of the week, we can construct the optimal estimator for time-series with a day-of-the-week seasonality.

1.3.1 Feasible Approximate Optimal Estimator of the Loss

Obviously, the estimator $\widehat{\mathcal{L}}_{ACV^*}$ as presented in Eq. 1.17 is not feasible, as V_ϕ is not known and needs to be estimated. Given the large size of matrix V_ϕ relative to the amount of data available, some restrictions on its structure are necessary. Furthermore, computational resources needed for the storage of V_ϕ, and even more so for its inversion, grow very quickly, making the computation of optimal weights λ_{ACV} directly via Eq. 1.17 infeasible for even moderately sized applications.[10]

Consequently, to make the proposed estimator practical, it is essential to develop the estimator $\widehat{V}_\phi$ jointly with an algorithm for computation of weights $\widehat{\lambda}_{ACV}$, so it is not prohibitively computationally expensive. To achieve this, we assume the following covariance structure:

$$Cov(l_j^{m,\,iv}, l_{j'}^{m,\,i'v}) = \begin{cases} 0 & \text{for } j + iv \neq j' + i'v \\ \sigma^2 \rho^{|i-i'|} & \text{for } j + iv = j' + i'v \end{cases}, \tag{1.22}$$

i.e., only contrasts computed from the same period are mutually correlated, and the strength of that correlation increases in the overlap between respective estimation windows.

[10] For applications as small as $T = 600$, $m = 400$, and $v = 1$, approximately 109 GB of RAM would be needed merely for the storage of V_ϕ (assuming double precision). Inversion of such a matrix is practically impossible via regularly available CPUs, as it requires $O\big(\big((m+v)\frac{n}{v} + m\big)^3\big)$ floating-point operations.

We can then express $\widehat{V}_\phi$ as

$$\widehat{V}_\phi = \hat{\sigma}^2 \begin{pmatrix} I & A_L^1 & A_L^2 & \dots & A_L^{\frac{n}{v}-2} & A_L^{\frac{n}{v}-1} & (A_L^{\frac{n}{v}})_{:,M} \\ A_U^1 & I & A_L^1 & \ddots & & A_L^{\frac{n}{v}-2} & (A_L^{\frac{n}{v}-1})_{:,M} \\ A_U^2 & A_U^1 & I & \ddots & & & (A_L^{\frac{n}{v}-2})_{:,M} \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ A_U^{\frac{n}{v}-2} & & & \ddots & I & A_L^1 & (A_L^2)_{:,M} \\ A_U^{\frac{n}{v}-1} & A_U^{\frac{n}{v}-2} & & \ddots & A_U^1 & I & (A_L^1)_{:,M} \\ (A_U^{\frac{n}{v}})_{M,:} & (A_U^{\frac{n}{v}-1})_{M,:} & (A_U^{\frac{n}{v}-2})_{M,:} & \dots & (A_U^2)_{M,:} & (A_U^1)_{M,:} & (I)_{M,M} \end{pmatrix} \tag{1.23}$$

where

- $A_U^i = (\hat{\rho} U^v)^i$
- $A_L^i = (\hat{\rho} L^v)^i$

and $M = (1,\, 2,\, \dots,\, m)$. Matrices $U, L \in \mathbb{R}^{(m+v)^2}$ are upper and lower shift matrices, i.e., matrices with ones on the superdiagonal and subdiagonal, respectively:

$$U_{i,j} = \begin{cases} 0 & \text{for } i-j \neq -1 \\ 1 & \text{for } i-j = -1 \end{cases} \qquad L_{i,j} = \begin{cases} 0 & \text{for } i-j \neq 1 \\ 1 & \text{for } i-j = 1 \end{cases}. \tag{1.24}$$

Parameters ρ and σ^2 can be estimated via a generalized method of moments based on differenced contrasts $l_j^{m,\,iv}$ and $l_{j-xv}^{m,\,(i+x)v}$ with varying x as described in Section 1.8 in more detail. Combined with the convenient structure of $\widehat{V}_\phi$ from Eq. 1.23 which admits a closed-form inverse as shown in Lemma 2, we can compute a feasible and approximately optimal analog of $\widehat{\mathcal{L}}_{ACV^*}$; estimator $\widehat{\mathcal{L}}_{ACV}$ with weights

$$\widehat{\lambda}_{ACV} = \widehat{V}_\phi^{-1} B^\top \left(B \widehat{V}_\phi^{-1} B^\top \right)^{-1} b, \tag{1.25}$$

without the need to store or numerically invert $\widehat{V}_\phi$, as described in Algorithm 1 in Section 1.9.

Admittedly, the parametrization via ρ and σ^2 is rather restrictive and might not fully account for all complexities of the true V_ϕ. However, since the covariances of contrasts from the same period are generally larger than other entries of V_ϕ by an order of magnitude, and since they tend to decay approximately exponentially, $\widehat{V}_\phi$ as defined in Eq. 1.23

successfully captures the key properties relevant for optimal weighting. Consequently, it is able to reap a major share of the available reduction of sampling variance as demonstrated in Sub-section 1.3.2. This is in line with the observation of Lavancier and Rochet (2016) that the weighting of estimators is often beneficial, even when based on an imperfect variance estimator. Furthermore, the estimator $\widehat{\mathcal{L}}_{CV}$ retains unbiasedness irrespective of how well $\widehat{V}_\phi$ approximates the true V_ϕ, as by definition $\widehat{\lambda}_{ACV} \in \Lambda_{ACV}$. Therefore, only the magnitude of the reduction of sampling variance is at risk when V_ϕ is imprecisely estimated.

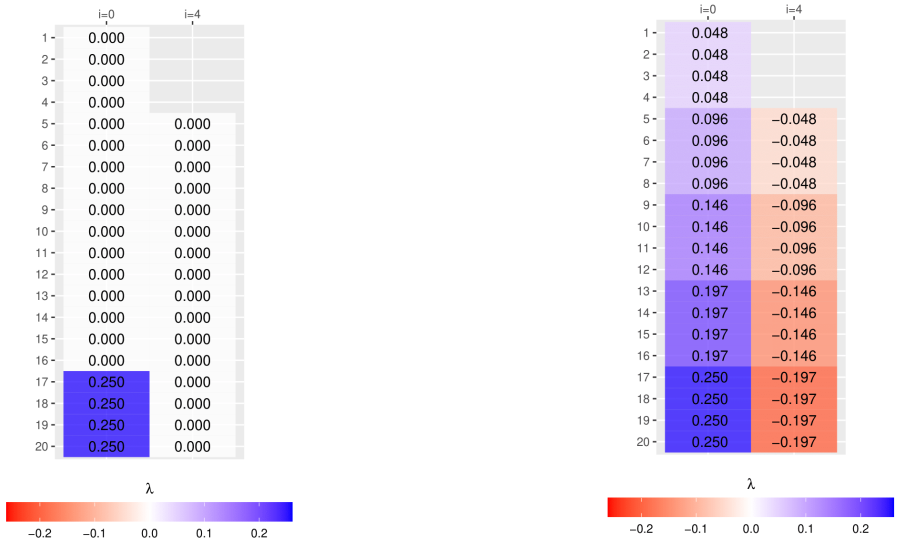
In summary, the process of computing the optimal (feasible) weights is as follows: First, the model is repeatedly estimated and evaluated using the rolling window pseudo out-of-sample evaluation scheme, recording both in-sample and out-of-sample contrasts to obtain ϕ. This step mirrors conventional loss estimation, with the addition of estimating the model at the final position where no out-of-sample observations exist (see Figure 1.1b). These additional in-sample contrasts $\boldsymbol{l}_{in}^{m,n}$ help to reduce sampling variance. Next, ϕ is used to estimate parameters $\hat{\rho}$ and $\hat{\sigma}^2$ of the assumed covariance structure using Estimator 1 detailed in Section 1.8. These estimates are then input into Algorithm 1 (Section 1.9) to compute the corresponding optimal weights $\widehat{\lambda}_{ACV}$ without explicitly constructing and inverting $\widehat{V}_\phi$. Finally, $\widehat{\mathcal{L}}_{ACV}$ is calculated as $\widehat{\mathcal{L}}_{ACV} = \widehat{\lambda}_{ACV}^\top \phi$. This entire process is automated in the `estimateL` function from the R package ACV[11], which takes the time-series $\{X_t\}_{t=1}^T$ and the model as inputs and performs all necessary steps.

1.3.2 Simulations

We first illustrate the core mechanism that leads to the reduction of sampling variance. Figures 1.2 and 1.3 display weights λ_{CV} and $\widehat{\lambda}_{ACV}$ for an illustrative simulated scenario with $T = 20$, $m = 16$, $n = 4$, and simple AR(1) process/model for the fixed and the rolling schemes, respectively. As is apparent from the figures, $\widehat{\lambda}_{ACV}$ includes in-sample empirical contrasts from periods $17-20$ with negative weights to eliminate a part of the idiosyncratic noise present in out-of-sample empirical contrasts. In turn, it is necessary to include other in-sample contrasts with positive weights to retain unbiasedness, creating a chain of positive and negative weights that gradually approach zero as we move towards the beginning of the sample. Obviously, such a small sample application is rarely encountered in practice, but it serves well for illustrative purposes, as the basic mechanics are the same

[11] Available at: `https://CRAN.R-project.org/package=ACV`

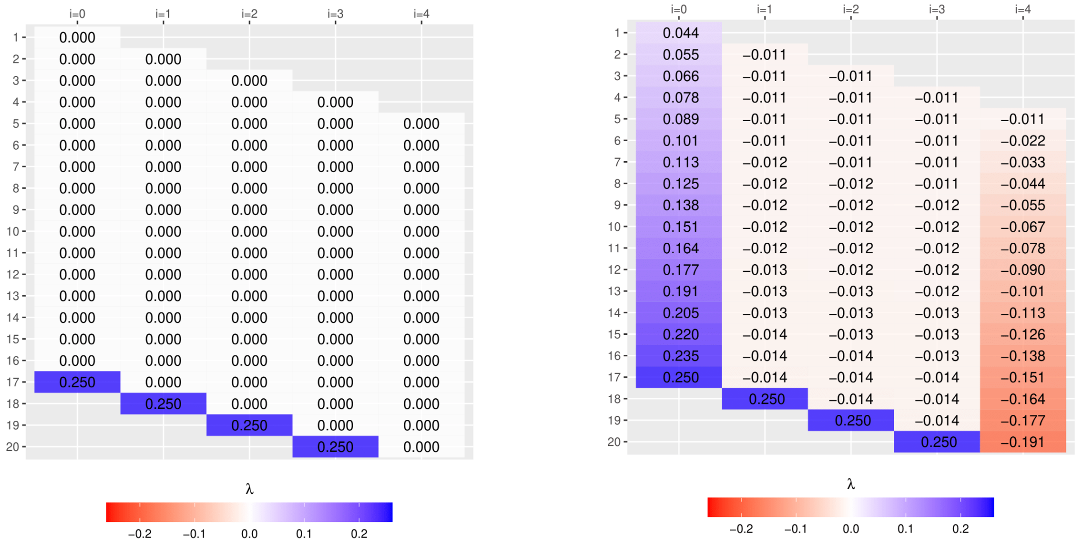
regardless of the sample size.



(a) Weights λ_{CV}

(b) Weights $\widehat{\lambda}_{ACV}$

Figure 1.2: A side by side comparison of weights λ_{CV} and $\widehat{\lambda}_{ACV}$ for the fixed scheme.



(a) Weights λ_{CV}

(b) Weights $\widehat{\lambda}_{ACV}$

Figure 1.3: A side by side comparison of weights λ_{CV} and $\widehat{\lambda}_{ACV}$ for the rolling scheme.

To assess the magnitude of the reduction of sampling variance, we perform a series of simulations with the AR(1) data generating process ($\varphi_1 = 0.9$) and an AR(1) model estimated via OLS. For varying m and n, we repeatedly (1,000 repetitions per combination) estimate the loss of the model by $\widehat{\mathcal{L}}_{CV}$ and $\widehat{\mathcal{L}}_{ACV}$ under a fixed scheme, and measure the variance of each estimator. Furthermore, to assess how well the feasible approximate estimator $\widehat{V}_\phi$ matches the true V_ϕ, we also compute the true V_ϕ by means of simulations, which then allows us to compute the unfeasible $\widehat{\mathcal{L}}_{ACV^*}$ and its variance as a reference point.

Figure 1.4 displays ratios $\frac{Var(\widehat{\mathcal{L}}_{ACV})}{Var(\widehat{\mathcal{L}}_{CV})}$ for different combinations of m and n. Clearly, the improvement brought by $\widehat{\mathcal{L}}_{ACV}$ relative to $\widehat{\mathcal{L}}_{CV}$ decreases in n and increases in m. This is because the larger the n, the more precise the $\widehat{\mathcal{L}}_{CV}$ and the lesser the potential of reducing the variance by optimal weighting. On the other hand, the larger the m, the stronger the correlation ρ, which in turn allows for better utilization of in-sample contrasts and larger reduction of sampling variance. Consequently, for commonly used in-/out-of-sample splitting rules that maintain a fixed ratio of n and m, $\widehat{\mathcal{L}}_{ACV}$ delivers a reduction of sampling variance that is approximately constant in the sample size T. Variance ratios range from ~ 0.4, when 1/3 of the sample is reserved for the out-of-sample evaluation, to ~ 0.1, when 1/10 of the sample is reserved for the out-of-sample evaluation. This clearly demonstrates that the gains are sizable and not limited to small sample applications.

Furthermore, the estimator $\widehat{V}_\phi$, despite its parsimonious parametrization, approximates the true matrix V_ϕ relatively well, as measured by the performance of $\widehat{\mathcal{L}}_{ACV}$ relative to $\widehat{\mathcal{L}}_{ACV^*}$. Indeed, the feasible estimator $\widehat{\mathcal{L}}_{ACV}$ is able to reap more than 90% of the available reduction of sampling variance relative to the optimal unfeasible estimator $\widehat{\mathcal{L}}_{ACV^*}$, as is apparent from the ratios $\frac{Var(\widehat{\mathcal{L}}_{CV})-Var(\widehat{\mathcal{L}}_{ACV})}{Var(\widehat{\mathcal{L}}_{CV})-Var(\widehat{\mathcal{L}}_{ACV^*})}$.

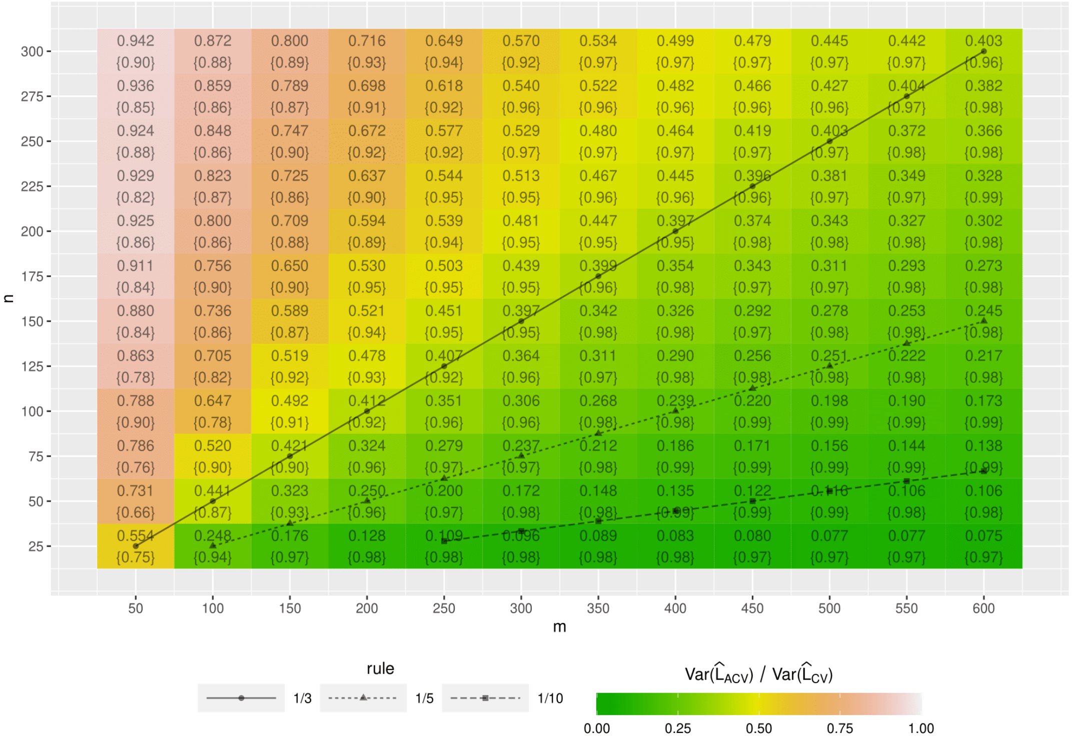


Figure 1.4: Ratios $Var(\widehat{\mathcal{L}}_{ACV})/Var(\widehat{\mathcal{L}}_{CV})$ for different combinations of m and n. Numbers in brackets measure the optimality of the feasible estimator relative to the true unfeasible optimal estimator, that is $(Var(\widehat{\mathcal{L}}_{CV}) - Var(\widehat{\mathcal{L}}_{ACV}))/(Var(\widehat{\mathcal{L}}_{CV}) - Var(\widehat{\mathcal{L}}_{ACV^*}))$. Common in-/out-of-sample splitting rules $\{1/3,\ 1/5,\ 1/10\}$ are highlighted.

1.4 Inference about Predictive Ability

The lower variance of the proposed estimator also translates to a substantial power advantage when performing inference about out-of-sample loss. Since Diebold and Mariano's (1995) pioneering work, many studies have been devoted to inference about predictive ability (see West (2006) or Clark and McCracken (2013b) for a comprehensive survey). Following the taxonomy of Clark and McCracken (2013b), these tests can be broadly divided into two families. First, there are the tests of population-level predictive ability (e.g. West, 1996; Clark and McCracken, 2001), which are concerned with the null hypothesis about prediction errors of models evaluated at the true, unknown parameters. Second, there are the tests of finite-sample predictive ability (e.g. Giacomini and White, 2006; Clark and McCracken, 2015), which are concerned with the null hypothesis about prediction errors of models with parameters that are themselves a function of a finitely sized window of observed data.

In this section, we apply the optimal estimator to an inference about finite-sample predictive ability, i.e., asymptotics $n \to \infty$ with m considered fixed. The reasons for adoption of this asymptotic framework are threefold. First, the null hypothesis addressed by the test of finite-sample predictive ability appeals to practitioners, as it takes into consideration the bias/variance trade-off inherent to comparing models of different complexity at a given sample size (Clark and McCracken, 2013b). Second, unlike for tests of population-level predictive ability, the null hypothesis cannot be addressed with full-sample methods, which tend to dominate pseudo out-of-sample methods in terms of power if applicable (Inoue and Kilian, 2005; Diebold, 2015). Furthermore, despite common belief, out-of-sample tests addressing the null hypothesis of population-level predictability seem to offer no advantage over purely in-sample tests in terms of robustness to data mining, dynamic misspecification, or structural changes, as demonstrated by Inoue and Kilian (2005). Lastly, the inference about finite-sample predictive ability is very general and can be used for both parametric/non-parametric and nested/non-nested models, which is in sharp contrast to tests of population-level predictive ability, where special care has to be taken to address individual cases (West, 2006).

We restrict our attention to the rolling window (i.e. $v = 1$) τ-step ahead unconditional test of equal predictive ability, i.e. the test of null hypothesis $H_0 : \mathcal{L}^m_{m+1}(\mathcal{M}_1) = \mathcal{L}^m_{m+1}(\mathcal{M}_2)$

for models $\mathcal{M}_1$ and $\mathcal{M}_2$.[12] This narrower scope is motivated by recent findings showing that the null hypothesis of equal conditional predictive ability can occur only under very specific data generating processes (Zhu and Timmermann, 2020) and findings that the inference under the fixed scheme (i.e. $v = n$) fails to address the desired null hypothesis about models $\mathcal{M}_1$ and $\mathcal{M}_2$ (McCracken, 2020).

Let $\Delta\widehat{\mathcal{L}}_{CV} \equiv \widehat{\mathcal{L}}_{CV}(\mathcal{M}_2) - \widehat{\mathcal{L}}_{CV}(\mathcal{M}_1)$ and let $\widehat{\sigma}^2_{CV}$ be a HAC estimator of its asymptotic variance; $\sigma^2_{CV} \equiv Var\Big(\sqrt{n}\Delta\widehat{\mathcal{L}}_{CV}\Big)$. As shown in Giacomini and White (2006), the following proposition applies.

Proposition 2. *Provided that:*

(i) $\{X_t\}$ *is mixing with* ϕ *of size* $-r/(2r-2)$, $r \geq 2$ *or* α *of size* $-r/(r-2)$, $r > 2$.

(ii) $\mathbb{E}\left[|\Delta l^{m,v}_{m+1}|^{2r}\right] < \infty$ *for all* v.

(iii) $\sigma^2_{CV} \equiv Var\Big(\sqrt{n}\Delta\widehat{\mathcal{L}}_{CV}\Big) > 0$ *for all* n *sufficiently large.*

Then under H_0

$$t_{DM} \equiv \frac{\Delta\widehat{\mathcal{L}}_{CV}}{\widehat{\sigma}_{CV}/\sqrt{n}} = \frac{(\lambda_{CV})^{\top}\Delta\phi}{\widehat{\sigma}_{CV}/\sqrt{n}} \xrightarrow{\text{d}} N(0,1) \tag{1.26}$$

where $\Delta\phi = \phi(\mathcal{M}_2) - \phi(\mathcal{M}_1)$ *and under* $H_A : |\mathbb{E}\Big[\Delta\widehat{\mathcal{L}}_{CV}\Big]| \geq \delta > 0$ *for all* n *sufficiently large*

$$P\left(|t_{DM}| > c\right) \longrightarrow 1. \tag{1.27}$$

We denote the test statistic by a subscript DM as it coincides exactly with the canonical Diebold and Mariano (1995) test (henceforth DM test).

Provided that $\{X_t\}$ is stationary, the third expression in Equation 1.26 motivates an alternative test statistic that utilizes the optimal weights $\widehat{\lambda}_{ACV}$ to gain more power. Note that, unlike in Section 1.3, here the weights are optimal for minimizing the variance of estimator of $\mathcal{L}^m_{m+1}(\mathcal{M}_2) - \mathcal{L}^m_{m+1}(\mathcal{M}_1)$ rather than that of individual estimators of $\mathcal{L}^m_{m+1}(\mathcal{M}_1)$ and $\mathcal{L}^m_{m+1}(\mathcal{M}_2)$, which is generally not the same task. We propose the following modification of the DM test, which uses the optimal affine weighting (ADM test henceforth).

[12] Note that in the generic definition of the rolling window estimator presented in Eq. 1.4, all observations up to m are utilized for estimation (but not as input to s) irrespective of the horizon τ. This is done for a notational convenience. To obtain the canonical rolling window estimator with $\tau > 1$, it suffices to define the $\widehat{\theta}$ associated with the given model so that it omits last $\tau - 1$ observations from the estimation (see e.g. Section 1.4.1).

Proposition 3. *Provided that $\{X_t\}$ is stationary, plim$(\hat{\rho}) \neq 1$, and (i)-(iii) holds, then*

$$t_{ADM} \equiv \frac{\Delta\widehat{\mathcal{L}}_{ACV}}{\widehat{\sigma}_{ACV}/\sqrt{n}} = \frac{(\widehat{\lambda}_{ACV})^{\top}\Delta\phi}{\widehat{\sigma}_{ACV}/\sqrt{n}} \xrightarrow{\text{d}} N(0,1) \tag{1.28}$$

where $\widehat{\sigma}_{ACV} = \widehat{\sigma}_{CV}\frac{\widehat{\lambda}_{ACV}^{\top}\widehat{V}_{\Delta\phi}\widehat{\lambda}_{ACV}}{\widehat{\lambda}_{CV}^{\top}\widehat{V}_{\Delta\phi}\widehat{\lambda}_{CV}}$ and under $H_A : |\mathbb{E}\Big[\Delta\widehat{\mathcal{L}}_{CV}\Big]| \geq \delta > 0$ for all n sufficiently large

$$P\left(|t_{ADM}| > c\right) \longrightarrow 1. \tag{1.29}$$

While widely adopted, the DM test is known to suffer from level distortions in small samples, stemming from the estimation of the long-run variance (see Clark and McCracken, 2013b). To mitigate this issue, Zhu and Timmermann (2020) propose to use Ibragimov and Müller's (2010) sub-sampling t-test (IM test henceforth), which does not require a variance estimation. In particular, Zhu and Timmermann (2020) prove the following proposition.

Proposition 4. *Suppose that $\{X_t\}$ is stationary and $E[\Delta l_{m+1}^{m,i}] = 0$. Assume that $E|\Delta l_{m+1}^{m,i}|^r = 0$ is bounded for some $r > 2$ and $\Delta l_{m+1}^{m,i}$ is strong mixing of size $-r/(r-2)$. Then, for fixed $K > 1$*

$$t_{IM} = \frac{\overline{\Delta\widehat{\mathcal{L}}_{CV}}}{\sqrt{(K-1)\sum_{k=1}^{K}\left(\widehat{\mathcal{L}}_{CV}^{(k)} - \overline{\Delta\widehat{\mathcal{L}}_{CV}}\right)^2/\sqrt{K}}} \xrightarrow{\text{d}} t_{K-1} \tag{1.30}$$

where $\widehat{\mathcal{L}}_{CV}^{(k)}$ is the loss estimate computed from the k-th block of data of size $\tilde{n} = n/K$, that is $\widehat{\mathcal{L}}_{CV}^{(k)} = \tilde{n}^{-1}\sum_{i=0}^{\tilde{n}-1}\Delta l_{m+1}^{m,i+\tilde{n}(k-1)} = \lambda_{CV}^{(k)}\Delta\phi^{(k)}$ where $\Delta\phi^{(k)} = \Delta\phi_M$ with $M = \{i\}_{i=1+\tilde{n}*(m+1)(k-1)}^{(\tilde{n}+1)*(m+1)-1+\tilde{n}*(m+1)(k-1)}$, and where $\overline{\Delta\widehat{\mathcal{L}}_{CV}} = K^{-1}\sum_{k=1}^{K}\widehat{\mathcal{L}}_{CV}^{(k)}$.

Similarly to the DM test, the IM test also immediately lends itself to a modified version that exploits the optimal weighting $\widehat{\lambda}_{ACV}$ (AIM test henceforth).

Proposition 5. *Suppose that $\{X_t\}$ is stationary, plim$(\hat{\rho}) \neq 1$, and $E[\Delta l_{m+1}^{m,i}] = 0$. Assume that $E|\Delta l_{m+1}^{m,i}|^r = 0$ is bounded for some $r > 2$ and $\Delta l_{m+1}^{m,i}$ is strong mixing of size $-r/(r-2)$. Then, for fixed $K > 1$*

$$t_{AIM} = \frac{\overline{\Delta\widehat{\mathcal{L}}_{ACV}}}{\sqrt{(K-1)\sum_{k=1}^{K}\left(\widehat{\mathcal{L}}_{ACV}^{(k)} - \overline{\Delta\widehat{\mathcal{L}}_{ACV}}\right)^2/\sqrt{K}}} \xrightarrow{\text{d}} t_{K-1} \tag{1.31}$$

where $\widehat{\mathcal{L}}^{(k)}_{ACV}$ is the loss estimate computed from the k-th block of data of size $\tilde{n} = n/K$, that is $\widehat{\mathcal{L}}^{(k)}_{ACV} = \widehat{\lambda}^{(k)}_{ACV}\Delta\phi^{(k)}$ where $\Delta\phi^{(k)} = \Delta\phi_M$ with $M = \{i\}_{i=1+\tilde{n}*(m+1)(k-1)}^{(\tilde{n}+1)*(m+1)-1+\tilde{n}*(m+1)(k-1)}$, and where $\overline{\Delta\widehat{\mathcal{L}}_{ACV}} = K^{-1}\sum_{k=1}^{K}\widehat{\mathcal{L}}^{(k)}_{ACV}$.

1.4.1 Power and Level Properties

To evaluate the power and level properties of the proposed tests, we adapt the simulation environment of McCracken (2019) that allows to generate series satisfying, or to a various degree violating, the null hypothesis of equal unconditional predictive ability under different forecast horizons τ. In particular, we consider a process

$$X_t = c + \eta_t \quad \text{with} \quad \eta_t = \varepsilon_t + \sum_{j=1}^{\tau-1} \varphi_j \varepsilon_{t-j}, \quad \varepsilon_t \sim N(0, \sigma^2), \tag{1.32}$$

and two models $\mathcal{M}_1 = \{s_1, \widehat{\theta}_1\}$ and $\mathcal{M}_2 = \{s_2, \widehat{\theta}_2\}$ producing point predictions of $X_{j+\tau}$:

$$s_1\left(\{X_t\}_{j-k-1}^{j}; \widehat{c}\right) = \widehat{c} \quad \text{with} \quad \widehat{c} = \widehat{\theta}_1\left(\{X_t\}_{j+\tau-m}^{j+\tau-1}\right) = 0, \tag{1.33}$$

$$s_2\left(\{X_t\}_{j-k-1}^{j}; \widehat{c}\right) = \widehat{c} \quad \text{with} \quad \widehat{c} = \widehat{\theta}_2\left(\{X_t\}_{j+\tau-m}^{j+\tau-1}\right) = \frac{1}{m-\tau+1}\sum_{t=j+\tau-m}^{j} X_t. \tag{1.34}$$

Model $\mathcal{M}_1$ is hence misspecified in that it omits the intercept c. Model $\mathcal{M}_2$, on the other hand, estimates c by averaging X_t over the estimation window. For $c \neq 0$ and $m \to \infty$, model $\mathcal{M}_2$ is always preferred over $\mathcal{M}_1$. In finite samples however, their relative performance is determined by m and c as expressed in the following proposition.

Proposition 6. *For the mean squared error contrast function $\gamma(X_t, \widehat{X_t}) = (X_t - \widehat{X_t})^2$, any $\varsigma \geq 1$, and*

$$c = \left(\varsigma\left(\alpha_0 + \frac{1}{\widetilde{m}}\alpha_0 + 2\sum_{i=1}^{\widetilde{m}-1}\frac{\widetilde{m}-i}{\widetilde{m}^2}\alpha_i\right) - \alpha_0\right)^{0.5}, \tag{1.35}$$

where $\widetilde{m} = m - \tau + 1$ and $\alpha_i = \mathbb{E}\left[\eta_t \eta_{t-i}\right]$, it holds that

$$\varsigma = \frac{\mathcal{L}^m_{m+1}(\mathcal{M}_1)}{\mathcal{L}^m_{m+1}(\mathcal{M}_2)}. \tag{1.36}$$

By setting $\varsigma = 1$, Proposition 6 allows to simulate series under H_0, that is with the constant c such that the loss stemming from the bias caused by its omission is exactly the same as the loss stemming from the noise introduced by its estimation. To explore the power of the proposed tests, we also consider values $\varsigma > 1$, in which the omission of the constant will result in worse predictions. In the exercise below, we follow the setup of McCracken (2019) and set $\sigma^2 = 1$ and $\varphi_j = (0.5)^j$. The truncation lag of Newey and West's (1987) HAC estimator in DM and ADM tests is chosen according to the commonly used rule $\lfloor \frac{3}{4} n^{\frac{1}{3}} \rfloor$ (see e.g. Lazarus et al., 2018). The number of groups K in IM and AIM tests is 2 as in Zhu and Timmermann (2020).

We repeatedly (2000 repetitions per combination of parameters) simulate the process from Eq. 1.32 with constant c corresponding to values of $\varsigma \in \{\ 1,\ \ 1.03125,\ \ 1.0625,\ \ 1.125,\ \ 1.25,\ \ 1.375,\ \ 1.5,\ \ 1.75,\ \ 2\ \}$ for $m = 100$, $n \in \{10, 20, 50, 100, 200, 300\}$, and $\tau \in \{1, 3, 6\}$. Figure 1.5 displays rejection rates for simulations with the forecast horizon $\tau = 1$. The proposed ADM and AIM tests exhibit substantially higher power relative to their conventional counterparts. In accordance with results from Section 1.3.2, the power gain is especially sizable in scenarios with small n relative to m. The power gain also appears to be more pronounced for IM type tests, which tend to sacrifice power in exchange for lesser finite sample level distortions, creating a greater opportunity for improvements. A similar power advantage of ADM and AIM tests relative to benchmarks is also observed when considering forecast horizons $\tau = 3$ and $\tau = 6$ as can be seen in Figures 1.6 and 1.7, respectively, in Appendix 1.A.

To better explore level properties, we repeat the exercise with $\varsigma = 1$, levels $p \in \{0.01, 0.05, 0.1\}$, and 10000 simulation repetitions. Inspecting Table 1.1, it is apparent that for all tests and forecast horizons, rejection probabilities approach the desired levels as $n \to \infty$. In small samples, we do observe the same level distortions for DM type tests as documented in the literature. The over-rejection is especially pronounced for higher τ as there, the data generating process exhibits a stronger temporal dependence which further complicates the estimation of the long run variance in small samples. Importantly however, the magnitude of these distortions is, in fact, smaller for the proposed ADM test. This shows that the power gain is achieved despite better level properties, not because of them. For IM type tests, rejection probabilities are generally closer to the desired levels even for small n, as expected. The AIM test exhibits larger level distortions in small samples relative to the conventional IM test. These distortions stem from stronger finite sample dependencies between individual estimators $\widehat{\mathcal{L}}_{ACV}^{(k)}$ introduced by the affine weighting. However, given the substantially higher power of the AIM test relative to the IM test, these finite sample distortions seem acceptable.

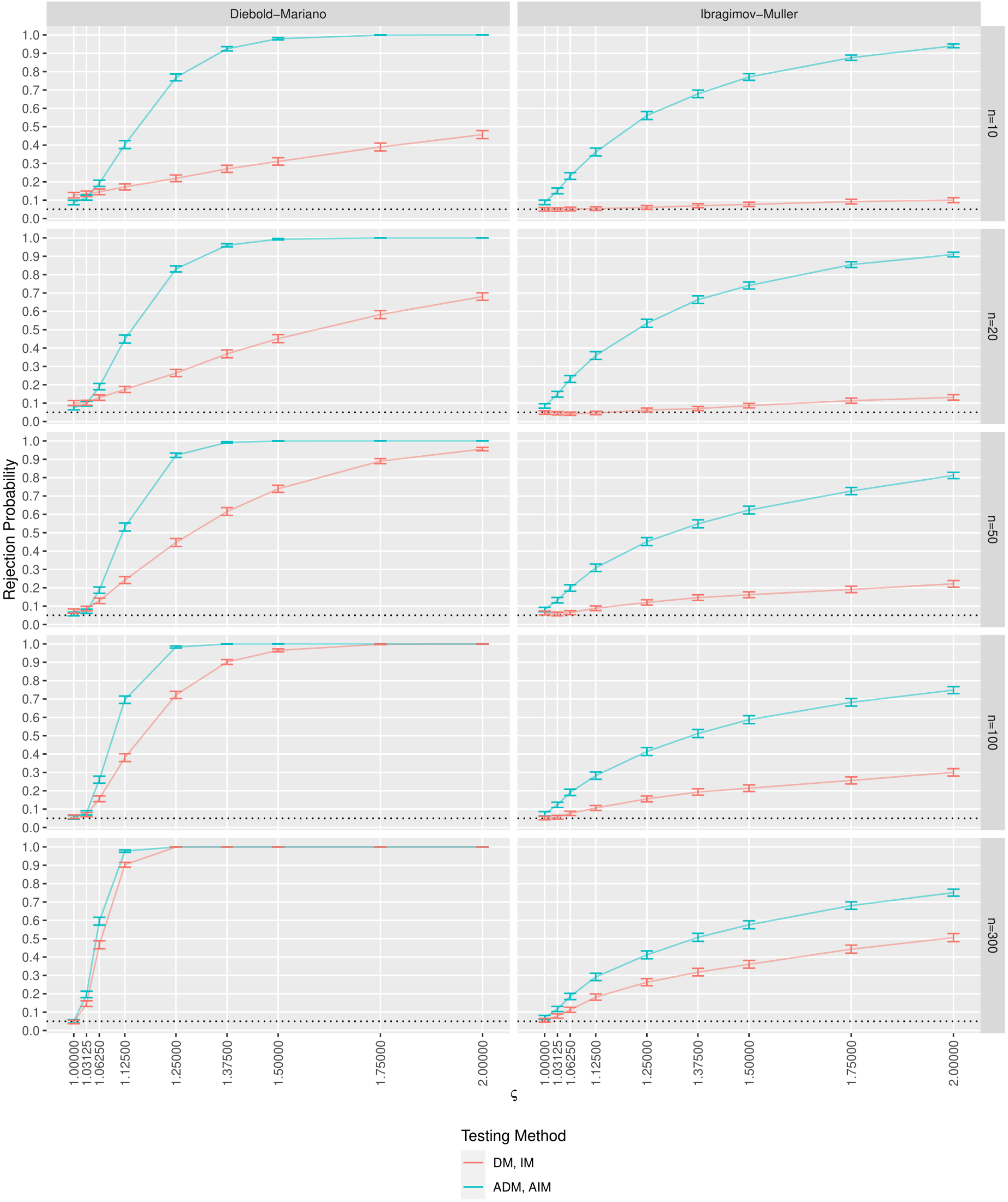


Figure 1.5: Plots of rejection probabilities for DM, IM, ADM, and AIM tests at level 0.05 for $\tau = 1$. Whiskers represent 95% confidence intervals.

		$p = 0.01$				$p = 0.05$				$p = 0.10$			
τ	n	DM	ADM	IM	AIM	DM	ADM	IM	AIM	DM	ADM	IM	AIM
1	10	0.048	0.035	0.009	0.019	0.120	0.084	0.047	0.090	0.188	0.132	0.095	0.174
	20	0.031	0.027	0.010	0.018	0.102	0.071	0.047	0.081	0.164	0.119	0.093	0.165
	50	0.016	0.018	0.012	0.016	0.072	0.055	0.052	0.075	0.126	0.100	0.097	0.151
	100	0.014	0.018	0.011	0.013	0.059	0.057	0.051	0.075	0.109	0.099	0.099	0.149
	300	0.011	0.011	0.012	0.015	0.046	0.045	0.057	0.072	0.087	0.082	0.115	0.141
3	10	0.130	0.098	0.010	0.020	0.235	0.183	0.054	0.098	0.318	0.254	0.101	0.188
	20	0.068	0.048	0.009	0.020	0.159	0.121	0.045	0.093	0.231	0.184	0.092	0.176
	50	0.035	0.028	0.010	0.015	0.109	0.085	0.049	0.076	0.181	0.144	0.097	0.150
	100	0.022	0.023	0.010	0.014	0.085	0.076	0.047	0.069	0.144	0.132	0.092	0.141
	300	0.012	0.012	0.010	0.012	0.054	0.055	0.051	0.068	0.103	0.101	0.106	0.134
6	10	0.171	0.139	0.010	0.024	0.284	0.240	0.055	0.109	0.365	0.313	0.108	0.208
	20	0.102	0.071	0.011	0.019	0.201	0.157	0.051	0.092	0.283	0.229	0.100	0.175
	50	0.053	0.041	0.008	0.016	0.149	0.112	0.043	0.077	0.226	0.182	0.089	0.150
	100	0.030	0.031	0.007	0.013	0.104	0.094	0.043	0.064	0.170	0.157	0.091	0.134
	300	0.014	0.014	0.011	0.014	0.064	0.063	0.053	0.062	0.115	0.111	0.108	0.132

Table 1.1: Rejection probabilities for DM, ADM, IM, and AIM tests under the null ($\varsigma = 1$) for different values of p and τ.

1.5 Empirical Evaluation

To demonstrate that the theoretical superiority of the proposed estimator also translates to real-life forecasting tasks, we perform an extensive evaluation on the M4 competition (Makridakis et al., 2020) data, consisting of 100,000 time-series ranging from yearly to hourly frequency. Participants in the M4 competition were asked to produce forecasts for each of the series for the upcoming 6/8/18/13/14/48 periods for yearly/quarterly/monthly/weekly/daily/hourly frequency, respectively. The organizers withheld the most recent segment of each series of corresponding length (test segments, henceforth). Submitted forecasts were then compared with test segments to evaluate their precision.

To assess the performance of $\widehat{\mathcal{L}}_{ACV}$ we consider two canonical models that were used as standards for comparison in the M4 competition; the ETS (Hyndman et al., 2002a), which automatically selects the optimal form of exponential smoothing via the information criterion, and the autoARIMA (Hyndman and Khandakar, 2008a), which selects the most appropriate ARIMA specification via the information criterion. Both these models are frequently used in practice and performed comparably well in the M4 competition, making them ideal candidates. Similarly to the competition, the performance of each model is assessed on the test segment of series using the sMAPE contrast function:[13]

$$\gamma\left(X_t,\, \widehat{X}_t\right) = \frac{|X_t - \widehat{X}_t|}{\frac{1}{2}|X_t| + \frac{1}{2}|\widehat{X}_t|}100. \tag{1.37}$$

Unlike in the M4 competition however, our interest is not in the performance of individual models per se, but rather in our ability to predict the out-of-sample performance $\widetilde{\mathcal{L}}_{CV,s}(\mathcal{M})$[14] on the test segment of a series s with the use of in-sample data only. To do so, we perform 1-step ahead pseudo out-of-sample evaluations under the rolling scheme (i.e. $\tau = 1$ and $v = 1$) with the same number of pseudo out-of-sample observations as in the test segment (i.e. $n \in \{6,\, 8,\, 18,\, 13,\, 14,\, 48\}$). For each series s, we compute the estimates $\widehat{\mathcal{L}}_{CV,s}(\mathcal{M})$ and $\widehat{\mathcal{L}}_{ACV,s}(\mathcal{M})$ and compare them with the actual 1-step ahead out-of-sample loss $\widetilde{\mathcal{L}}_{CV,s}(\mathcal{M})$ incurred on the test segment. The overall precision of the estimator is computed as

$$MSE_{CV}(\mathcal{M}) = \frac{1}{|S|}\sum_{s\in S}\left(\widetilde{\mathcal{L}}_{CV,s}(\mathcal{M}) - \widehat{\mathcal{L}}_{CV,s}(\mathcal{M})\right)^2 \tag{1.38}$$

and

$$MSE_{ACV}(\mathcal{M}) = \frac{1}{|S|}\sum_{s\in S}\left(\widetilde{\mathcal{L}}_{CV,s}(\mathcal{M}) - \widehat{\mathcal{L}}_{ACV,s}(\mathcal{M})\right)^2 \tag{1.39}$$

with S being a subset of time-series under consideration. To better assess the performance on different types of series, we also subject each series to a non-parametric CS test for the presence of a trend (Cox and Stuart, 1955) and a QS test for the presence of seasonality (Ljung and Box, 1978).

Table 1.2 depicts MSE_{CV} and MSE_{ACV} for both models across all frequencies, further broken down by the results of the CS and QS tests (for both, the threshold $p = 0.05$ is considered). For each model, percentage improvements of $\widehat{\mathcal{L}}_{ACV}$ over $\widehat{\mathcal{L}}_{CV}$ in terms of MSE are shown alongside their statistical significance. As is apparent, the use of $\widehat{\mathcal{L}}_{ACV}$ leads to a substantially more precise estimation of the incurred out-of-sample loss $\widetilde{\mathcal{L}}_{CV}$, in particular to a reduction of MSE by 13.0% and 10.6% on average for ETS and autoARIMA, respectively. It is worth highlighting that this reduction of MSE likely

[13] This contrast function was chosen by organizers so that losses of series on different scales are approximately comparable. As a robustness check, we also repeat the exercise with MAE and MSE contrast functions with prior normalization and obtain comparable results (available upon request).

[14] We use the notation $\widetilde{\mathcal{L}}_{CV}$ rather than $\widehat{\mathcal{L}}_{CV}$ to highlight that this is the loss incurred on the test segment (i.e., the true out-of-sample evaluation). However, as the test segment is of finite length, this is still only an estimate of the true theoretical loss $\mathcal{L}_{CV}$. The subscript CV indicates that the conventional estimator is used to compute the loss incurred on the test segment.

underestimates the true gains, as the comparison is made with respect to the estimate of loss $\widetilde{\mathcal{L}}_{CV}$ rather than the true theoretical loss $\mathcal{L}_{CV}$; hence, the corresponding part of the MSE in principle cannot be reduced. A back of the envelope calculation suggests that the theoretical reduction of MSE, if computed against the true loss rather than its estimate, is actually twice the size.

A significant portion of the series in the M4 competition display non-stationary characteristics. Specifically, 90% of the series exhibit a trend, 39% show seasonality, and 36% feature both trend and seasonal components. The fact that $\widehat{\mathcal{L}}_{ACV}$ exhibits superior performance relative to $\widehat{\mathcal{L}}_{CV}$, even when applied indiscriminately to a wide range of time-series without any regards for stationarity, clearly demonstrates its robustness and practical applicability. However, it is important to mention that the reduction in MSE is less pronounced for series with seasonal patterns.

To assess the robustness of these findings, we also repeat the exercise for forecast horizons τ up to 3 and 6 (Tables 1.5 and 1.6 in Appendix 1.A). In these cases, the optimal estimator $\widehat{\mathcal{L}}_{ACV}$ reduces MSE by 9.7% and 3.2%, respectively for ETS and 7.0% and 1.4%, respectively for autoARIMA. Although more modest than in the case with horizon $\tau = 1$, all these differences are statistically significant. The lower relative gains of $\widehat{\mathcal{L}}_{ACV}$ over $\widehat{\mathcal{L}}_{CV}$ for longer forecast horizons likely stems from the fact that the mean and the dispersion of out-of-sample contrasts tend to increase in the forecast horizon, reflecting the difficulty of forecasting far ahead to the future. Consequently, the gains achievable through optimal affine weighting are smaller in comparison to the higher inherent uncertainty present in the estimation of the out-of-sample loss.

To gauge the computational complexity of the proposed estimator, Table 1.4 in Appendix 1.A provides average run-times needed for the computation of the vector of contrasts ϕ as well as for the computation of $\widehat{\mathcal{L}}_{CV}$ and $\widehat{\mathcal{L}}_{ACV}$. Unsurprisingly, the computation of $\widehat{\mathcal{L}}_{ACV}$ is more demanding than that of the conventional estimator, averaging to approximately 5 seconds per series. Overall however, the usage of $\widehat{\mathcal{L}}_{ACV}$ results in only $<$ 20% longer run-time as the most demanding task is the computation of ϕ which is common to both $\widehat{\mathcal{L}}_{CV}$ and $\widehat{\mathcal{L}}_{ACV}$. For more complex forecasting models likely used in practice, the relative difference in run-times would be even smaller.

Lastly, we assess the performance of $\widehat{\mathcal{L}}_{ACV}$ in terms of model selection. In this exercise, the task is to use the loss estimate to select the model $\mathcal{M}$ that will perform best on the test segment of a given series, i.e., to identify the model with the smallest $\widetilde{\mathcal{L}}_{CV,s}(\mathcal{M})$. Table

	time-series			ETS			autoARIMA		
Period	Trending	Seasonal	N	MSE_{CV}	MSE_{ACV}	ΔMSE [%]	MSE_{CV}	MSE_{ACV}	ΔMSE [%]
Yearly			23000	48.68	41.47	-14.8***	57.05	51.37	-10.0***
				(1.65)	(1.49)		(2.42)	(2.38)	
	F	F	2214	139.93	126.41	-9.7***	194.26	187.93	-3.3
				(10.60)	(10.27)		(16.17)	(16.36)	
	F	T	267	20.22	19.86	-1.8	24.08	23.84	-1.0
				(5.49)	(5.43)		(6.22)	(5.76)	
	T	F	15076	49.62	41.06	-17.3***	54.29	46.57	-14.2***
				(1.92)	(1.65)		(2.74)	(2.63)	
	T	T	5443	10.35	9.12	-11.9**	10.51	10.44	-0.7
				(0.90)	(0.81)		(1.38)	(1.37)	
Quarterly			24000	28.70	24.18	-15.8***	33.87	29.30	-13.5***
				(1.16)	(0.97)		(1.41)	(1.22)	
	F	F	1561	92.17	78.02	-15.4**	101.50	81.68	-19.5***
				(8.96)	(7.75)		(9.51)	(7.78)	
	F	T	681	65.29	49.90	-23.6	90.95	81.41	-10.5
				(14.91)	(8.83)		(16.93)	(14.93)	
	T	F	14115	26.34	21.68	-17.7***	29.50	25.23	-14.5***
				(1.32)	(1.09)		(1.49)	(1.18)	
	T	T	7643	16.82	15.50	-7.9	23.02	21.50	-6.6
				(1.48)	(1.38)		(2.43)	(2.33)	
Monthly			48000	19.32	17.65	-8.6***	21.68	19.69	-9.2***
				(0.47)	(0.45)		(0.57)	(0.55)	
	F	F	2574	78.64	63.56	-19.2***	87.64	73.09	-16.6***
				(5.02)	(4.42)		(5.89)	(5.28)	
	F	T	1964	21.89	19.70	-10.0*	24.63	21.33	-13.4**
				(1.99)	(1.88)		(2.67)	(2.38)	
	T	F	21613	23.60	22.27	-5.6**	26.57	24.78	-6.8***
				(0.72)	(0.75)		(0.91)	(0.94)	
	T	T	21849	7.85	7.49	-4.7*	8.81	8.23	-6.6**
				(0.36)	(0.36)		(0.41)	(0.40)	
Weekly			359	8.81	5.55	-37.0***	6.47	5.95	-8.0
				(1.40)	(0.99)		(0.86)	(1.13)	
	F	F	54	13.18	10.50	-20.4	9.50	7.51	-21.0
				(3.05)	(4.09)		(2.27)	(1.67)	
	F	T	3	2.72	1.85	-32.1	0.81	0.67	-18.0
				(2.09)	(1.47)		(0.80)	(0.64)	
	T	F	257	8.81	5.15	-41.5***	6.39	6.35	-0.7
				(1.81)	(1.06)		(1.06)	(1.53)	
	T	T	45	4.01	2.14	-46.8	3.61	2.12	-41.2
				(1.99)	(0.91)		(1.65)	(0.84)	
Daily			4227	1.62	1.56	-3.5	2.11	2.15	1.7
				(0.33)	(0.37)		(0.51)	(0.54)	
	F	F	226	2.71	3.98	47.1	4.05	4.36	7.6
				(2.33)	(3.73)		(3.53)	(4.01)	
	F	T	19	0.39	0.43	10.8	0.33	0.42	26.5
				(0.19)	(0.24)		(0.18)	(0.21)	
	T	F	3535	0.89	0.71	-19.3*	0.89	0.77	-13.4*
				(0.22)	(0.19)		(0.23)	(0.22)	
	T	T	447	6.94	7.11	2.5	10.92	12.05	10.3**
				(2.29)	(2.45)		(4.07)	(4.31)	
Hourly			414	12.71	8.55	-32.7***	53.11	45.62	-14.1
				(2.27)	(1.47)		(11.36)	(12.16)	
	F	F	1	0.24	0.09	-60.4	0.06	0.02	-56.3
				(NA)	(NA)		(NA)	(NA)	
	F	T	125	29.67	17.97	-39.4***	90.33	59.04	-34.6***
				(6.65)	(3.89)		(21.18)	(15.35)	
	T	F	5	2.12	2.70	27.1	1.56	1.86	18.6
				(1.93)	(2.54)		(1.30)	(1.49)	
	T	T	283	5.45	4.53	-16.8	37.77	40.63	7.6
				(1.36)	(1.22)		(13.64)	(16.44)	
All			**100000**	**27.51**	**23.94**	**-13.0*****	**31.99**	**28.60**	**-10.6*****
				(0.52)	**(0.47)**		**(0.71)**	**(0.68)**	

Table 1.2: Comparison of $\widehat{\mathcal{L}}_{CV}$ and $\widehat{\mathcal{L}}_{ACV}$ in terms of the loss estimation. $\Delta MSE\,[\%] = \frac{MSE_{ACV} - MSE_{CV}}{MSE_{CV}} 100$. Standard errors in brackets, $***p < 0.001$, $**\ p < 0.01$, $*p < 0.05$.

1.3 shows the average incurred loss $\widetilde{\mathcal{L}}_{CV}$ and the probability of selecting the best model, for AIC (Akaike, 1998), $\widehat{\mathcal{L}}_{CV}$ and $\widehat{\mathcal{L}}_{ACV}$. The table also includes the average loss that would be incurred if we knew which model was the best-performing on the test segment.[15] Obviously, such a selection is not feasible in practice but it provides a useful benchmark, as it represents the best possible outcome that can be achieved via model selection alone. Compared to AIC, $\widehat{\mathcal{L}}_{ACV}$ achieves a 23.7% reduction of incurred loss relative to what is achievable and is more likely to select the best model by 4.9% points.[16] Compared to $\widehat{\mathcal{L}}_{CV}$, the relative reduction of loss is more modest, only 1.4%, but still statistically significant. The estimator $\widehat{\mathcal{L}}_{ACV}$ is 0.3% points more likely to select the best model than $\widehat{\mathcal{L}}_{CV}$.

time-series		ex-post opt.	AIC		CV		ACV		AIC vs ACV	CV vs ACV
Period	N	$\widetilde{\mathcal{L}}$	$P(best)$	$\widetilde{\mathcal{L}}$	$P(best)$	$\widetilde{\mathcal{L}}$	$P(best)$	$\widetilde{\mathcal{L}}$	$\Delta\widetilde{\mathcal{L}}\,[\%]$	$\Delta\widetilde{\mathcal{L}}\,[\%]$
Yearly	23000	6.489 (0.056)	0.513 (0.003)	7.186 (0.065)	0.528 (0.003)	7.096 (0.063)	0.526 (0.003)	7.089 (0.062)	-13.9***	-1.2
Quarterly	24000	5.602 (0.055)	0.484 (0.003)	6.198 (0.061)	0.548 (0.003)	6.007 (0.059)	0.551 (0.003)	6.002 (0.059)	-32.8***	-1.0
Monthly	48000	6.513 (0.043)	0.525 (0.002)	6.944 (0.046)	0.578 (0.002)	6.858 (0.045)	0.585 (0.002)	6.852 (0.045)	-21.3***	-1.7
Weekly	359	5.033 (0.298)	0.616 (0.026)	5.162 (0.303)	0.526 (0.026)	5.245 (0.316)	0.577 (0.026)	5.229 (0.316)	52.1	-7.3
Daily	4227	1.013 (0.027)	0.516 (0.008)	1.052 (0.031)	0.522 (0.008)	1.030 (0.028)	0.509 (0.008)	1.031 (0.028)	-53.6***	4.4*
Hourly	414	6.765 (0.443)	0.551 (0.024)	9.261 (0.655)	0.804 (0.020)	6.911 (0.452)	0.819 (0.019)	6.869 (0.450)	-95.9***	-28.9
All	**100000**	**6.052** **(0.028)**	**0.512** **(0.002)**	**6.575** **(0.031)**	**0.558** **(0.002)**	**6.456** **(0.030)**	**0.561** **(0.002)**	**6.451** **(0.030)**	**-23.7*** **	**-1.4***

Table 1.3: Comparison of AIC, $\widehat{\mathcal{L}}_{CV}$ and $\widehat{\mathcal{L}}_{ACV}$ in terms of model selection. For $x \in \{AIC, CV\}$, $\Delta\widetilde{\mathcal{L}}\,[\%] = \frac{\widetilde{\mathcal{L}}_{CV}(\mathcal{M}_{ACV}) - \widetilde{\mathcal{L}}_{CV}(\mathcal{M}_x)}{\widetilde{\mathcal{L}}_{CV}(\mathcal{M}_x) - \widetilde{\mathcal{L}}_{CV}(\mathcal{M}_{ex-post\,opt.})} 100$. Standard errors in brackets, $***p < 0.001$, $**\ p < 0.01$, $*p < 0.05$.

While the gains from more accurate model selection via $\widehat{\mathcal{L}}_{ACV}$ rather than $\widehat{\mathcal{L}}_{CV}$ are not as sizable, it should be noted that the variance minimizing weights of $\widehat{\mathcal{L}}_{ACV}$ are not necessarily optimal in terms of selecting a model so that its incurred loss is the lowest in expectation. By computing multiple sets of weights jointly, so that they are optimal in terms of model selection, we could presumably attain even better results. This promising research direction is, however, beyond the scope of this chapter.

[15] We denoted these incurred losses and probabilities of selecting the best model by $\widetilde{\mathcal{L}}_{CV}(\mathcal{M}_x)$ and $P(best)_x$, respectively, where $x \in \{AIC, CV, ACV, ex-post\,opt.\}$.

[16] The dominance of CV and ACV over AIC likely stems from violations of stationarity, which more heavily penalize the AIC than the ACV, and/or the fact that the sMAPE contrast function in Eq. 1.37 is not aligned with the MSE contrast function, for which the AIC is designed. A thorough theoretical comparison of the AIC and pseudo out-of-sample estimators such as $\widehat{\mathcal{L}}_{CV}$ or $\widehat{\mathcal{L}}_{ACV}$ is beyond the scope of this chapter. A detailed analysis can, however, be found in Inoue and Kilian (2006).

1.6 Conclusions

We propose an alternative estimator of the out-of-sample loss that optimally utilizes both in-sample and out-of-sample empirical contrasts via a system of affine weights. We prove that under stationarity, the proposed (unfeasible) estimator is the best unbiased linear estimator of the out-of-sample loss and that it dominates the conventional estimator in terms of the sampling variance. We also propose an approximate feasible variant of the estimator, which closely matches the performance of the unfeasible optimal estimator, and which exhibits a substantially smaller sampling variance relative to the conventional estimator, by a factor of ~ 0.4 to ~ 0.1 in our simulations. The reduction of sampling variance is most sizable in situations where few observations are designated for the out-of-sample evaluation relative to the number of in-sample observations.

The proposed optimal estimator can also be applied to the inference about predictive ability. We put forward modifications of Diebold and Mariano's (1995) test and of Ibragimov and Müller's (2010) test and show that utilization of the optimal estimator leads to a substantial power gain (often by a factor > 2) in detecting deviations from the null hypothesis of equal predictive ability. In addition, the finite sample level distortions of Diebold and Mariano's (1995) test frequently documented in the literature seem to be attenuated, rather than exacerbated, by the system of optimal affine weights.

Finally, to assess the real-life applicability of the estimator and its robustness, we perform an extensive evaluation on time-series from the M4 forecasting competition (Makridakis et al., 2020). In line with the theoretical derivations and the simulation evidence, the proposed estimator more precisely estimates the losses incurred on the test segments of series ($> 10\%$ reduction of MSE relative to the conventional estimator). Furthermore, selecting a model based on the proposed estimator leads to a higher probability of selecting the ex-post optimal model and also to an overall lower loss relative to that which would be incurred if the model were selected according to the conventional estimator. The performance gains of our proposed estimator are evident across the diverse range of time-series in the M4 competition. Many of these series display characteristics of non-stationarity, which provides an opportunity for us to assess the estimator's effectiveness beyond ideal theoretical conditions. Our results suggest a degree of robustness in the estimator's application to various real-world scenarios.

There are several natural extensions of our proposed estimator of loss. Throughout this chapter, we have focused predominantly on loss estimation and inference. However, in practice, loss estimation is frequently not an object of interest in its own right, but rather an intermediate objective to further improve forecasting performance. Time-series are often split into three segments: training, validation, and testing, with pseudo out-of-sample evaluation being repeatedly performed on the test set to select the most suitable model and fine-tune hyperparameters. A natural extension would be to consider leveraging in-sample empirical contrasts to make these techniques less susceptible to random noise and hence more effective when the most suitable models or hyperparameters are being selected. Likewise, numerous studies have demonstrated superior performance of forecast combinations (Wang et al., 2023). Many such forecast combination schemes rely on past performance (see, e.g., Bates and Granger (1969) and Winkler and Makridakis (1983)) when determining the weight of forecasts. A more efficient estimator of loss might therefore be used to improve the performance of such combination schemes.

There are two potential approaches to this enhancement. The first involves deriving variance-minimizing estimates for each hyperparameter combination or candidate model, with the expectation that reduced individual variances, even without accounting for their correlation structure, will yield performance improvements. The second approach would be to optimize directly for improved performance, akin to minimizing the variance of the loss differential as in Section 1.4 and in the model selection exercise in Section 1.5. However, generating such complex optimal weighting schemes in dimensions higher than two could prove challenging. We are grateful to Prof. Andrey Vasnev and Prof. John Galbraith for highlighting these possibilities.

Another line of possible extensions involves relaxing the underlying assumptions to make the estimator applicable to a wider array of practical applications. The estimator described in this chapter utilizes all in-sample contrasts up to the very beginning of the in-sample segment to reduce the sampling variance of the out-of-sample loss estimator. However, the benefit of including increasingly distant in-sample contrasts quickly diminishes, with the most recent in-sample observations being most useful. It might be possible to restrict weights for in-sample contrasts that are more distant than a certain multiple of the pseudo out-of-sample segment length to zero without significantly impairing the sampling variance reduction relative to the conventional estimator. This adjustment could make the estimator more robust to structural breaks or interruptions.

Furthermore, as formulated in this chapter, the estimator requires stationarity of the underlying time-series to guarantee unbiasedness. However, it might also be possible to prove unbiasedness under the weaker condition of stationarity of forecast errors, provided that the contrast function depends solely on the error. This would substantially widen the range of time-series to which the estimator might be applied. This possibility is also indirectly evidenced by the evaluation on the M4 dataset, where the proposed estimator outperformed the conventional estimator in terms of the sampling variance even when the assumptions of stationarity were violated. We are grateful to Prof. Andrey Vasnev for outlining these ways in which the estimator might be generalized.

1.7 Proofs

Lemma 1. *Let* $P = \left\{P_1, P_2, \ldots, P_{\text{card}(P)}\right\}$ *be a partition of* $\{1, 2, \ldots, \text{card}(\phi)\}$ *such that* $\forall j \in \{1, 2, \ldots, card(P)\}\ \forall i, i' \in P_j : \mathbb{E}[\phi_i] = \mathbb{E}[\phi_{i'}]$*. Then for* $\lambda \in \Lambda_{ACV}$ *where*

$$\Lambda_{ACV} = \left\{ \lambda_{CV} + x \middle| x \in \mathbb{R}^{\text{card}(\phi)} \wedge \forall j \in \{1, 2, \ldots, card(P)\} : \sum_{i \in P_j} x_i = 0 \right\}, \tag{1.40}$$

it holds that

$$\mathbb{E}[\lambda^\top \phi] = \mathcal{L}_{CV} \tag{1.41}$$

and

$$\lambda^\top \Sigma_\phi \lambda = \lambda^\top V_\phi \lambda \tag{1.42}$$

where $\Sigma_\phi = \mathbb{E}\left[(\phi - \mathcal{L}_{CV}\mathbf{1})(\phi - \mathcal{L}_{CV}\mathbf{1})^\top\right]$ *and* $V_\phi = Var(\phi)$*.*

Proof of Lemma 1 *To prove this lemma, consider*

$$\begin{aligned} \mathbb{E}[\lambda^\top \phi] &= \mathbb{E}[(\lambda_{CV} + x)^\top \phi] \\ &= \mathbb{E}[(\lambda_{CV})^\top \phi] + \mathbb{E}[x^\top \phi] \\ &= \mathcal{L}_{CV} + \sum_{j=1}^{\text{card}(P)} \underbrace{\sum_{i \in P_j} x_i \mathbb{E}[\phi_i]}_{=0} \\ &= \mathcal{L}_{CV}. \end{aligned} \tag{1.43}$$

Furthermore

$$\begin{aligned} \Sigma_\phi &= \mathbb{E}[(\phi - \mathcal{L}_{CV}\mathbf{1})\,(\phi - \mathcal{L}_{CV}\mathbf{1})^\top] \\ &= \mathbb{E}[((\phi - \mathbb{E}[\phi]) + (\mathbb{E}[\phi] - \mathcal{L}_{CV}\mathbf{1}))\,((\phi - \mathbb{E}[\phi]) + (\mathbb{E}[\phi] - \mathcal{L}_{CV}\mathbf{1}))^\top] \\ &= Var(\phi) + (\mathbb{E}[\phi] - \mathcal{L}_{CV}\mathbf{1})\,(\mathbb{E}[\phi] - \mathcal{L}_{CV}\mathbf{1})^\top \end{aligned} \tag{1.44}$$

and

$$\begin{aligned} \lambda^\top \Sigma_\phi \lambda &= \lambda^\top \left(Var(\phi) + (\mathbb{E}[\phi] - \mathcal{L}_{CV}\mathbf{1})\,(\mathbb{E}[\phi] - \mathcal{L}_{CV}\mathbf{1})^\top\right) \lambda \\ &= \lambda^\top Var(\phi) \lambda + \lambda^\top\,(\mathbb{E}[\phi] - \mathcal{L}_{CV}\mathbf{1})\,(\mathbb{E}[\phi] - \mathcal{L}_{CV}\mathbf{1})^\top \lambda \\ &= \lambda^\top Var(\phi) \lambda \end{aligned} \tag{1.45}$$

as

$$\begin{aligned}\lambda^\top\left(\mathbb{E}[\phi]-\mathcal{L}_{CV}\mathbf{1}\right) &= \left(\lambda_{CV}+x\right)^\top\left(\mathbb{E}[\phi]-\mathcal{L}_{CV}\mathbf{1}\right)\\ &= \left(\lambda_{CV}\right)^\top\mathbb{E}[\phi]-\left(\lambda_{CV}\right)^\top\mathcal{L}_{CV}\mathbf{1}+x^\top\mathbb{E}[\phi]-x^\top\mathcal{L}_{CV}\mathbf{1}\\ &= \mathcal{L}_{CV}-\mathcal{L}_{CV}+\sum_{j=1}^{\mathrm{card}(P)}\underbrace{\sum_{i\in P_j}x_i\mathbb{E}[\phi_i]}_{=0}-\mathcal{L}_{CV}\sum_{j=1}^{\mathrm{card}(P)}\underbrace{\sum_{i\in P_j}x_i\mathbf{1}_i}_{=0}\\ &= 0,\end{aligned}\tag{1.46}$$

which completes the proof.

Proof of Proposition 1. Let $P=\{P_1,\,P_2,\,\dots,\,P_{m+v}\}$ be a partition of $\{1,\;2,\;\dots,\;\mathrm{card}(\phi)\}$ such that $\forall j\in\{1,2,\dots,m+v\}\,\forall i\in\left\{0,\,1,\,\dots,\,\frac{n}{v}\right\}:\,l_j^{m,\,iv}\in P_j$. Due to stationarity, it holds that $\forall j\in\{1,2,\dots,\mathit{card}(P)\}\;\forall i,i'\in P_j:\mathbb{E}[\phi_i]=\mathbb{E}[\phi_{i'}]$ and hence Lemma 1 can be applied. Also note that the set Λ_{ACV} from Lemma 1 can be equivalently expressed as

$$\lambda\in\Lambda_{ACV}\qquad\Longleftrightarrow\qquad B\lambda=b\tag{1.47}$$

with

$$B=\left(\mathbf{1}_{n/v}^\top\otimes I,\,I_{:,M}\right)\qquad b=\begin{pmatrix}\mathbf{0}_m\\ \frac{1}{v}\mathbf{1}_v\end{pmatrix}\tag{1.48}$$

where $M=(1,\,2,\,\dots,\,m)$.

By virtue of Proposition 1, for any $\lambda\in\Lambda_{ACV}$, it holds that

$$\mathbb{E}[\lambda^\top\phi]=\mathcal{L}_{CV}\tag{1.49}$$

and

$$\lambda^\top\Sigma_\phi\lambda=\lambda^\top V_\phi\lambda,\tag{1.50}$$

i.e., all estimators with weights in Λ_{ACV} are unbiased estimators of $\mathcal{L}_{CV}$ and their mean squared error is equal to their variance. We are interested in the best possible estimator (in terms of mean squared error/variance) in the set Λ_{ACV}. Formally:

$$\underset{\lambda}{\mathrm{argmin}}\;\lambda^\top V_\phi\lambda\;\;\text{s.t}:B\lambda=b.\tag{1.51}$$

The Lagrangian associated with the problem is given by

$$L(\lambda,\, \alpha) = \lambda^\top V_\phi \lambda - \alpha^\top (B\lambda - b). \tag{1.52}$$

Necessary conditions for pair $\{\lambda,\, \alpha\}$ to be solution to Eq. 1.51 are

$$\frac{\partial L(\lambda,\, \alpha)}{\partial \lambda} = 2V_\phi \lambda - B^\top \alpha = 0, \tag{1.53}$$

$$\frac{\partial L(\lambda,\, \alpha)}{\partial \alpha} = B\lambda - b = 0. \tag{1.54}$$

From Eq. 1.53, it follows

$$\lambda = \frac{1}{2} V_\phi^{-1} B^\top \alpha, \tag{1.55}$$

combining that with Eq. 1.54 leads to

$$\alpha = 2\left(BV_\phi^{-1}B^\top\right)^{-1} b \tag{1.56}$$

and consequently

$$\lambda = V_\phi^{-1} B^\top \left(BV_\phi^{-1}B^\top\right)^{-1} b. \tag{1.57}$$

The invertibility of matrix V_ϕ and $\left(BV_\phi^{-1}B^\top\right)$ follows from positive-definiteness of V_ϕ and full rank of B. The sufficient conditions then follows from the fact that $\lambda^\top V_\phi \lambda$ is strictly convex function as V_ϕ is positive definite. We denote the optimum weights as λ_{ACV} and the corresponding estimator by $\widehat{\mathcal{L}}_{ACV^*}$, i.e.

$$\widehat{\mathcal{L}}_{ACV^*} = (\lambda_{ACV})^\top \phi \qquad \text{with} \qquad \lambda_{ACV} = V_\phi^{-1} B^\top \left(BV_\phi^{-1}B^\top\right)^{-1} b. \tag{1.58}$$

The statement

$$\mathbb{E}[\widehat{\mathcal{L}}_{ACV^*}] = \mathcal{L}_{CV} \tag{1.59}$$

stems directly from $\lambda_{ACV} \in \Lambda_{ACV}$ and Lemma 1. Statements

$$Var(\widehat{\mathcal{L}}_{ACV^*}) < Var(\lambda^\top \phi) \qquad \text{with} \qquad \lambda \in \Lambda_{ACV},\, \lambda \neq \lambda_{ACV} \tag{1.60}$$

and

$$Var(\widehat{\mathcal{L}}_{ACV^*}) \leq Var(\widehat{\mathcal{L}}_{CV}) \tag{1.61}$$

follows from strict convexity of function $\lambda^\top V_\phi \lambda$ and $\lambda_{CV} \in \Lambda_{ACV}$, respectively.

It remains to show that there is no $\lambda' \notin \Lambda_{ACV}$ such that it is guaranteed that $\mathbb{E}[(\lambda')^\top \phi] = \mathcal{L}_{CV}$. Suppose that there is such λ' and let $x = \lambda' - \lambda_{CV}$. From $\lambda' \notin \Lambda_{ACV}$ it follows that $\exists j' : \sum_{i \in P_{j'}} x_i = c \neq 0$. Suppose that $\forall j \in \{1,\, 2,\, ...,\, m+v\}\,, j \neq j' : \mathcal{L}_j^m = 0$ and $\mathcal{L}_{j'}^m \neq 0$. Then

$$\mathbb{E}[(\lambda')^\top \phi] = \mathbb{E}[\lambda_{CV}^\top \phi] + \mathbb{E}[x^\top \phi] = \mathcal{L}_{CV} + c\mathcal{L}_{j'}^m \neq \mathcal{L}_{CV}, \tag{1.62}$$

which is a contradiction.

Lemma 2. *Provided that $\hat{\rho} \neq 1$, matrix $\widehat{V}_\phi$ defined as:*

$$\widehat{V}_\phi = \hat{\sigma}^2 \begin{pmatrix} I & A_L^1 & A_L^2 & \dots & A_L^{\frac{n}{v}-2} & A_L^{\frac{n}{v}-1} & (A_L^{\frac{n}{v}})_{:,M} \\ A_U^1 & I & A_L^1 & \ddots & & A_L^{\frac{n}{v}-2} & (A_L^{\frac{n}{v}-1})_{:,M} \\ A_U^2 & A_U^1 & I & \ddots & & & (A_L^{\frac{n}{v}-2})_{:,M} \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ A_U^{\frac{n}{v}-2} & & & \ddots & I & A_L^1 & (A_L^2)_{:,M} \\ A_U^{\frac{n}{v}-1} & A_U^{\frac{n}{v}-2} & & \ddots & A_U^1 & I & (A_L^1)_{:,M} \\ (A_U^{\frac{n}{v}})_{M,:} & (A_U^{\frac{n}{v}-1})_{M,:} & (A_U^{\frac{n}{v}-2})_{M,:} & \dots & (A_U^2)_{M,:} & (A_U^1)_{M,:} & (I)_{M,M} \end{pmatrix} \tag{1.63}$$

with

- $A_U^i = (\hat{\rho} U^v)^i$
- $A_L^i = (\hat{\rho} L^v)^i$
- $M = (1,\, 2,\, \dots,\, m)$

is invertible and its inverse is given by:

$$\widehat{V}_\phi^{-1} = \frac{1}{\hat{\sigma}^2} \begin{pmatrix} Z_1 & Z_L & 0 & \dots & 0 & 0 & (0)_{:,M} \\ Z_U & Z_2 & Z_L & \ddots & & 0 & (0)_{:,M} \\ 0 & Z_U & Z_2 & \ddots & & & (0)_{:,M} \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & & & \ddots & Z_2 & Z_L & (0)_{:,M} \\ 0 & 0 & & \ddots & Z_U & Z_2 & (Z_L)_{:,M} \\ (0)_{M,:} & (0)_{M,:} & (0)_{M,:} & \dots & (0)_{M,:} & (Z_U)_{M,:} & (Z_3)_{M,M} \end{pmatrix} \tag{1.64}$$

with

- $Z_1 = I + \frac{\hat{\rho}^2}{1-\hat{\rho}^2} L^v U^v$
- $Z_2 = I + \frac{\hat{\rho}^2}{1-\hat{\rho}^2} \left(L^v U^v + U^v L^v\right)$
- $Z_3 = \frac{1}{1-\hat{\rho}^2} I$
- $Z_U = \frac{-\hat{\rho}}{1-\hat{\rho}^2} U^v$
- $Z_L = \frac{-\hat{\rho}}{1-\hat{\rho}^2} L^v$.

Proof of Lemma 2 *To prove this lemma, we check individual sub-matrices of $\widehat{V}_\phi \widehat{V}_\phi^{-1}$ to verify that, together, they indeed constitute an identity matrix:*

- $[i,i] : i = 1$

$$\begin{aligned} IZ_1 + A_L^1 Z_U &= I \left(I + \frac{\hat{\rho}^2}{1-\hat{\rho}^2} L^v U^v \right) + \hat{\rho} L^v \frac{-\hat{\rho}}{1-\hat{\rho}^2} U^v \\ &= I \end{aligned} \tag{1.65}$$

- $[i,i] : 1 < i \leq \frac{n}{v}$

$$\begin{aligned} A_U^1 Z_L + IZ_2 + A_L^1 Z_U &= \hat{\rho} U^v \frac{-\hat{\rho}}{1-\hat{\rho}^2} L^v + I \left(I + \frac{\hat{\rho}^2}{1-\hat{\rho}^2} \left(L^v U^v + U^v L^v\right) \right) \\ &\quad + \hat{\rho} L^v \frac{-\hat{\rho}}{1-\hat{\rho}^2} U^v \\ &= I \end{aligned} \tag{1.66}$$

- $[i,i] : i = \frac{n}{v} + 1$

$$\begin{aligned} (A_U^1)_{M,:}(Z_L)_{:,M} + (I)_{M,M}(Z_3)_{M,M} &= \hat{\rho}(U^v)_{M,:} \frac{-\hat{\rho}}{1-\hat{\rho}^2} (L^v)_{:,M} + (I)_{M,M} \frac{1}{1-\hat{\rho}^2} (I)_{M,M} \\ &= \frac{-\hat{\rho}^2}{1-\hat{\rho}^2} (I)_{M,M} + \frac{1}{1-\hat{\rho}^2} (I)_{M,M} \\ &= (I)_{M,M} \end{aligned} \tag{1.67}$$

- $[i,j] : 1 < i \leq \frac{n}{v}, j = 1$

$$\begin{aligned} A_U^{i-1}Z_1 + A_U^{i-2}Z_U &= (\hat{\rho}U^v)^{i-2}\left(\hat{\rho}U^v\left(I + \frac{\hat{\rho}^2}{1-\hat{\rho}^2}L^vU^v\right) + \frac{-\hat{\rho}}{1-\hat{\rho}^2}U^v\right) \\ &= (\hat{\rho}U^v)^{i-2}\frac{1}{1-\hat{\rho}^2}\left(\left(\hat{\rho}-\hat{\rho}^3\right)U^v + \hat{\rho}^3U^v - \hat{\rho}U^v\right) \\ &= 0 \end{aligned} \tag{1.68}$$

- $[i,j] : i = \frac{n}{v} + 1, j = 1$

$$\begin{aligned} (A_U^{i-1})_{M,:}Z_1 + (A_U^{i-2})_{M,:}Z_U &= (A_U^{i-1}Z_1 + A_U^{i-2}Z_U)_{M,:} \\ &= (0)_{M,:} \end{aligned} \tag{1.69}$$

- $[i,j] : j < i < \frac{n}{v}, 1 < j \leq \frac{n}{v}$

$$\begin{aligned} &A_U^{i-j+1}Z_L + A_U^{i-j}Z_2 + A_U^{i-j-1}Z_U = \\ &= (\hat{\rho}U^v)^{i-j-1}\left((\hat{\rho}U^v)^2\frac{-\hat{\rho}}{1-\hat{\rho}^2}L^v + \hat{\rho}U^v\left(I + \frac{\hat{\rho}^2}{1-\hat{\rho}^2}\left(L^vU^v + U^vL^v\right)\right) + \frac{-\hat{\rho}}{1-\hat{\rho}^2}U^v\right) \\ &= (\hat{\rho}U^v)^{i-2}\frac{1}{1-\hat{\rho}^2}\left(-\hat{\rho}^3U^{2v}L^v + \left(\hat{\rho}-\hat{\rho}^3\right)U^v + \hat{\rho}^3U^vL^vU^v + \hat{\rho}^3U^{2v}L^v - \hat{\rho}U^v\right) \\ &= 0 \end{aligned} \tag{1.70}$$

- $[i,j] : i = \frac{n}{v} + 1, 1 < j \leq \frac{n}{v}$

$$\begin{aligned} &(A_U^{i-j+1})_{M,:}Z_L + (A_U^{i-j})_{M,:}Z_2 + (A_U^{i-j-1})_{M,:}Z_U = \\ &= (A_U^{i-j+1}Z_L + A_U^{i-j}Z_2 + A_U^{i-j-1}Z_U)_{M,:} \\ &= (0)_{M,:}\,. \end{aligned} \tag{1.71}$$

The fact that remaining submatrices above the diagonal equal 0 *follows from the symmetry of* $\widehat{V}_\phi$*.*

Proof of Proposition 2*.* The proof is provided in Giacomini and White (2006, p. 1575).

Lemma 3. *Provided that* $\{X_t\}$ *is stationary,* $plim(\hat{\rho}) \neq 1$*, and* $v = 1$*, it holds that:*

$$\sqrt{n}(\widehat{\lambda}_{ACV} - \lambda_{CV})^\top\phi \xrightarrow{\text{p}} 0 \tag{1.72}$$

and

$$\frac{\widehat{\lambda}_{ACV}^{\top}\widehat{V}_{\phi}\widehat{\lambda}_{ACV}}{\lambda_{CV}^{\top}\widehat{V}_{\phi}\lambda_{CV}} \xrightarrow{\text{p}} 1. \tag{1.73}$$

Proof of Lemma 3 *To prove this lemma, we first express $\widehat{\lambda}_{ACV}$ as function of m, n and ρ. First let us recapitulate that*

$$\widehat{\lambda}_{ACV} = \widehat{V}_{\phi}^{-1}B^{\top}\left(B\widehat{V}_{\phi}^{-1}B^{\top}\right)^{-1}b \tag{1.74}$$

and note that for $v = 1$, the system of restriction B and b representing partition implied by stationarity is the following:

$$B = \left(\mathbf{1}_n^{\top} \otimes I,\, I_{:,M}\right) \qquad b = \begin{pmatrix} \mathbf{0}_m \\ 1 \end{pmatrix} \tag{1.75}$$

where $M = (1,\, 2,\, \ldots,\, m)$.

Consider any $\hat{\rho} \neq 1$, using the Lemma 2, we can express

$$\widehat{V}_{\phi}^{-1}B^{\top} = \frac{1}{\hat{\sigma}^2}\begin{pmatrix} Z_1 + Z_L \\ \mathbf{1}_{n-1} \otimes (Z_U + Z_2 + Z_L) \\ (Z_U)_{M,:} + (Z_3)_{M,M}\, I_{M,:} \end{pmatrix} \tag{1.76}$$

and furthermore

$$\begin{aligned} B\widehat{V}_{\phi}^{-1}B^{\top} &= \frac{1}{\hat{\sigma}^2}\Big(Z_1 + Z_L + (n-1)(Z_U + Z_2 + Z_L) + \underbrace{I_{:,M}\,(Z_U)_{M,:}}_{=Z_U} + \underbrace{I_{:,M}\,(Z_3)_{M,M}\,I_{M,:}}_{=\frac{1}{1-\hat{\rho}^2}U^vL^v}\Big) \\ &= \frac{1}{\hat{\sigma}^2}\left(n\,(Z_U + Z_2 + Z_L) + Z_1 - Z_2 + \frac{1}{1-\hat{\rho}^2}U^vL^v\right) \\ &= \frac{1}{\hat{\sigma}^2}\left(n\,(Z_U + Z_2 + Z_L) + U^vL^v\right) \\ &= \frac{1}{\hat{\sigma}^2}\frac{1}{1-\hat{\rho}^2}\left(n\left((1-\hat{\rho}^2)I + \hat{\rho}^2(L^vU^v + U^vL^v) - \hat{\rho}(U^v + L^v)\right) + (1-\hat{\rho}^2)U^vL^v\right). \end{aligned} \tag{1.77}$$

Under $v = 1$, the resulting matrix is tridiagonal, in particular:

$$B\widehat{V}_{\phi}^{-1}B^{\top} = \frac{1}{\hat{\sigma}^2}\frac{1}{1-\hat{\rho}^2}\underbrace{\begin{pmatrix} a_1 & c & 0 & \dots & 0 & 0 & 0 \\ c & a_2 & c & \ddots & & 0 & 0 \\ 0 & c & a_3 & \ddots & & & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & & & \ddots & a_{m-1} & c & 0 \\ 0 & 0 & & \ddots & c & a_m & c \\ 0 & 0 & 0 & \dots & 0 & c & a_{m+1} \end{pmatrix}}_{\equiv Y} \tag{1.78}$$

with

- $a_1 = n + 1 - \hat{\rho}^2$
- $a_j = (1 + \hat{\rho}^2)n + 1 - \hat{\rho}^2$, $1 < j < m + 1$
- $a_{m+1} = n$
- $c = -n\hat{\rho}$.

Using the results of Usmani (1994) on the inverse of tridiagonal matrices, we know that the left-most column of Y^{-1} can be expressed as

$$\begin{aligned} \left(Y^{-1}\right)_{j,m+1} &= (-1)^{j+(m+1)}\, c^{(m+1)-j}\,\frac{\theta_{j-1}}{\theta_{m+1}} * 1 \\ &= (n\hat{\rho})^{m+1-j}\,\frac{\theta_{j-1}}{\theta_{m+1}} \end{aligned} \tag{1.79}$$

with $\theta_0 = 1$, $\theta_1 = a_1$, and $\theta_j = a_j\theta_{j-1} + c^2\theta_{j-2}$ with $2 \le j \le m+1$. In our particular case it then follows that

$$\theta_j = \begin{cases} n^j + O(n^{j-1}) & 0 \le j \le m \\ (1-\hat{\rho}^2)n^j + O(n^{j-1}) & j = m+1, \end{cases} \tag{1.80}$$

which can be proven by induction as $\theta_0 = 1$ and $\theta_1 = n + 1 - \hat{\rho}^2$ and for $2 \le j \le m$ it *holds that*

$$\begin{aligned}\theta_j &= a_j\theta_{j-1} + c^2\theta_{j-2} \\ &= \left((1+\hat{\rho}^2)n + 1 - \hat{\rho}^2\right)\left(n^{j-1} + O(n^{j-2})\right) - (-n\hat{\rho})^2\left(n^{j-2} + O(n^{j-3})\right) \\ &= n^j + O(n^{j-1})\end{aligned} \tag{1.81}$$

and consequently for $j = m + 1$

$$\begin{aligned}\theta_j &= a_j\theta_{j-1} + c^2\theta_{j-2} \\ &= (n)\left(n^{j-1} + O(n^{j-2})\right) - (-n\hat{\rho})^2\left(n^{j-2} + O(n^{j-3})\right) \\ &= (1-\hat{\rho}^2)n^j + O(n^{j-1}).\end{aligned} \tag{1.82}$$

Therefore

$$\begin{aligned}\left(Y^{-1}\right)_{j,m+1} &= (n\hat{\rho})^{m+1-j}\,\frac{n^{j-1} + O(n^{j-2})}{(1-\hat{\rho}^2)n^{m+1} + O(n^m)} \\ &= \frac{\hat{\rho}^{m+1-j}n^m + O(n^{m-1})}{(1-\hat{\rho}^2)n^{m+1} + O(n^m)} \\ &= \frac{\hat{\rho}^{m+1-j}}{1-\hat{\rho}^2}\frac{1}{n} + O\left(\frac{1}{n^2}\right)\end{aligned} \tag{1.83}$$

and finally

$$\begin{aligned}\left(\left(B\widehat{V}_\phi^{-1}B^\top\right)^{-1} b\right)_j &= \hat{\sigma}^2\left(1-\hat{\rho}^2\right)\left(Y^{-1}\right)_{j,m+1} \\ &= \hat{\sigma}^2\hat{\rho}^{m+1-j}\frac{1}{n} + O\left(\frac{1}{n^2}\right)\end{aligned} \tag{1.84}$$

and furthemore using the definitions of Z_j, Z_U, and Z_L

$$\begin{aligned}\widehat{\lambda}_{ACV} &= \begin{pmatrix} Z_1 + Z_L \\ \mathbf{1}_{n-1} \otimes (Z_U + Z_2 + Z_L) \\ (Z_U)_{M,:} + (Z_3)_{M,M}\, I_{M,:} \end{pmatrix} \begin{pmatrix} \frac{1}{n}\hat{\rho}^m + O(\frac{1}{n^2}) \\ \frac{1}{n}\hat{\rho}^{m-1} + O(\frac{1}{n^2}) \\ \vdots \\ \frac{1}{n}\hat{\rho}^1 + O(\frac{1}{n^2}) \\ \frac{1}{n}\hat{\rho}^0 + O(\frac{1}{n^2}) \end{pmatrix} \\ &= \begin{pmatrix} \frac{1}{n}P + \epsilon_1(n) \\ \mathbf{1}_{n-1} \otimes \left(\frac{1}{n} \begin{pmatrix} \mathbf{0}_m \\ 1 \end{pmatrix} + \epsilon_2(n) \right) \\ \frac{1}{n}\mathbf{0}_m + \epsilon_3(n) \end{pmatrix}\end{aligned} \tag{1.85}$$

where $P = \left(\hat{\rho}^m, \hat{\rho}^{m-1}, ..., \hat{\rho}^1, \hat{\rho}^0\right)^\top$ and ϵ_k for $k \in \{1, 2, 3\}$ is a vector function that is element-wise $O(\frac{1}{n^2})$.

With the explicit, albeit approximate (up to $O(\frac{1}{n^2})$), expression for $\widehat{\lambda}_{ACV}$, we proceed with proving the individual claims. Let us denote

$$\begin{aligned}\lambda_\Delta \equiv \widehat{\lambda}_{ACV} - \lambda_{CV} &= \begin{pmatrix} \frac{1}{n}P + \epsilon_1(n) \\ \mathbf{1}_{n-1} \otimes \left(\frac{1}{n} \begin{pmatrix} \mathbf{0}_m \\ 1 \end{pmatrix} + \epsilon_2(n) \right) \\ \frac{1}{n}\mathbf{0}_m + \epsilon_3(n) \end{pmatrix} - \begin{pmatrix} \mathbf{1}_n \otimes \left(\frac{1}{n} \begin{pmatrix} \mathbf{0}_m \\ 1 \end{pmatrix} \right) \\ \frac{1}{n}\mathbf{0}_m \end{pmatrix} \\ &= \begin{pmatrix} \frac{1}{n}\left(P - \begin{pmatrix} \mathbf{0}_m \\ 1 \end{pmatrix} \right) + \epsilon_1(n) \\ \mathbf{1}_{n-1} \otimes \epsilon_2(n) \\ \epsilon_3(n) \end{pmatrix}\end{aligned} \tag{1.86}$$

and furthermore

$$\lambda_\Delta^\top \phi = \sum_{j=1}^{m+1} \underbrace{\left(\left(\frac{1}{n}\hat{\rho}^{m+1-j} + \epsilon_1(n)_j \right) l_j^{m,0} + \sum_{i=1}^{n-1} \epsilon_2(n)_j l_j^{m,i} + \epsilon_3(n)_j l_j^{m,n} \mathbf{1}(j \le m) \right)}_{\equiv Q_j}. \tag{1.87}$$

Consider any $j \in \{1, 2, ..., m+1\}$. From the definition of $\epsilon_k(n)$, $k \in \{1, 2, 3\}$ it follows *that* $\exists C, n_0 : \forall n \geq n_0$*:*

$$\begin{aligned}0 \leq |\sqrt{n}Q_j| \leq \sqrt{n}\Bigg(&\left(|\frac{1}{n}\hat{\rho}^{m+1-j}| + |\epsilon_1(n)_j|\right)|l_j^{m,0}| + \\ &\sum_{i=1}^{n-1}|\epsilon_2(n)_j||l_j^{m,i}| + |\epsilon_3(n)_j||l_j^{m,n}|\mathbf{1}(j \leq m)\Bigg) \\ &\leq \sqrt{n}\frac{1}{n}\hat{\rho}^{m+1-j}|l_j^{m,0}| + \sqrt{n}\sum_{i=1}^{n-1}C\frac{1}{n^2}|l_j^{m,i}| + \sqrt{n}C\frac{1}{n^2}|l_j^{m,n}|\mathbf{1}(j \leq m) \\ &= \underbrace{\frac{1}{\sqrt{n}}\hat{\rho}^{m+1-j}|l_j^{m,0}|}_{\xrightarrow{\text{p}}0} + \underbrace{\frac{1}{\sqrt{n}}C\frac{1}{n}\sum_{i=1}^{n-1}|l_j^{m,i}|}_{\xrightarrow{\text{p}}0} + \underbrace{\frac{1}{\sqrt{n}}C\frac{1}{n}|l_j^{m,n}|\mathbf{1}(j \leq m)}_{\xrightarrow{\text{p}}0} \xrightarrow{\text{p}} 0.\end{aligned} \tag{1.88}$$

Considering that

$$-\sum_{j=1}^{m+1}|\sqrt{n}Q_j| \leq -|\sqrt{n}\sum_{j=1}^{m+1}Q_j| \leq \sqrt{n}\lambda_\Delta^\top\phi \leq \sum_{j=1}^{m+1}|\sqrt{n}Q_j| \leq |\sqrt{n}\sum_{j=1}^{m+1}Q_j| \tag{1.89}$$

it follows that

$$\sqrt{n}(\lambda_{ACV} - \widehat{\lambda}_{CV})^\top\phi \xrightarrow{\text{p}} 0 \tag{1.90}$$

via Squeeze theorem. To prove the second claim, note that

$$\begin{aligned}\frac{\widehat{\lambda}_{ACV}^\top\widehat{V}_\phi\widehat{\lambda}_{ACV}}{\lambda_{CV}^\top\widehat{V}_\phi\lambda_{CV}} &= \frac{(\lambda_{CV} + \lambda_\Delta)^\top\widehat{V}_\phi(\lambda_{CV} + \lambda_\Delta)}{\lambda_{CV}^\top\widehat{V}_\phi\lambda_{CV}} \\ &= \frac{\lambda_{CV}^\top\widehat{V}_\phi\lambda_{CV} + \lambda_\Delta^\top\widehat{V}_\phi\lambda_{CV} + \lambda_{CV}^\top\widehat{V}_\phi\lambda_\Delta + \lambda_\Delta^\top\widehat{V}_\phi\lambda_\Delta}{\lambda_{CV}^\top\widehat{V}_\phi\lambda_{CV}}.\end{aligned} \tag{1.91}$$

Let us denote

$$\tilde{\epsilon}_1(n) = |\frac{1}{n}\left(P - \begin{pmatrix}\mathbf{0}_m \\ 1\end{pmatrix}\right) + \epsilon_1(n)| \tag{1.92}$$

$$e(n) = \frac{1}{n}\begin{pmatrix}\mathbf{0}_m \\ 1\end{pmatrix}. \tag{1.93}$$

From the definition of $\epsilon_k(n)$, $k \in \{1,\, 2,\, 3\}$ it follows that $\exists C, n_0 : \forall n \geq n_0$:

$$\begin{aligned}
n\lambda_\Delta^\top \widehat{V}_\phi \lambda_{CV} &\leq n|\lambda_\Delta^\top||\widehat{V}_\phi||\lambda_{CV}| \\
&\leq n\hat{\sigma}^2\Big((2m+1)|\tilde{\epsilon}_1(n)|^\top Je(n) + n(2m+1)|\epsilon_2(n)|^\top Je(n) \\
&\quad + (2m+1)|\epsilon_3(n)|^\top Je(n)\Big) \\
&\leq n\hat{\sigma}^2(m+1)\left((2m+1)C\frac{1}{n}\frac{1}{n} + n(2m+1)C\frac{1}{n^2}\frac{1}{n} + (2m+1)C\frac{1}{n^2}\frac{1}{n}\right) \xrightarrow{\text{p}} 0.
\end{aligned} \tag{1.94}$$

Where we utilized the fact that $\frac{1}{\hat{\sigma}^2}\widehat{V}_\phi$ can be bounded from above by a block-Toeplitz matrix with a matrix of ones (denoted by J) on the diagonal and first m sub/super-diagonals. Similarly for

$$\begin{aligned}
n\lambda_\Delta^\top \widehat{V}_\phi \lambda_\Delta &\leq n|\lambda_\Delta^\top||\widehat{V}_\phi||\lambda_\Delta| \\
&\leq n\hat{\sigma}^2(|\tilde{\epsilon}_1(n)|^\top J|\tilde{\epsilon}_1(n)| + 2(n-1)|\tilde{\epsilon}_1(n)|^\top J|\epsilon_2(n)| + (n-1)^2|\epsilon_2(n)|^\top J|\epsilon_2(n)| + \\
&+ 2|\tilde{\epsilon}_1(n)|^\top J|\epsilon_3(n)| + 2(n-1)|\epsilon_2(n)|^\top J|\epsilon_3(n)| + 2|\epsilon_3(n)|^\top J|\epsilon_3(n)|) \\
&\leq n\hat{\sigma}^2(m+1)^2(C\frac{1}{n}\frac{1}{n} + 2(n-1)C\frac{1}{n}\frac{1}{n^2} + (n-1)^2C\frac{1}{n^2}\frac{1}{n^2} + \\
&+ 2C\frac{1}{n}\frac{1}{n^2} + 2(n-1)C\frac{1}{n^2}\frac{1}{n^2} + C\frac{1}{n^2}\frac{1}{n^2}) \xrightarrow{\text{p}} 0.
\end{aligned} \tag{1.95}$$

Utilizing the Squeeze theorem, we obtain $n\lambda_\Delta^\top \widehat{V}_\phi \lambda_{CV} \xrightarrow{\text{p}} 0$ and $n\lambda_\Delta^\top \widehat{V}_\phi \lambda_\Delta \xrightarrow{\text{p}} 0$. By noting that $plim(n\lambda_{CV}^\top \widehat{V}_\phi \lambda_{CV}) = const$ we can invoke Slutsky's theorem to obtain

$$\frac{\widehat{\lambda}_{ACV}^\top \widehat{V}_\phi \widehat{\lambda}_{ACV}}{\lambda_{CV}^\top \widehat{V}_\phi \lambda_{CV}} = \frac{n\widehat{\lambda}_{ACV}^\top \widehat{V}_\phi \widehat{\lambda}_{ACV}}{n\lambda_{CV}^\top \widehat{V}_\phi \lambda_{CV}} \xrightarrow{\text{p}} 1. \tag{1.96}$$

Proof of Proposition 3. Applying lemma 3 to the contrasts differential $\Delta\phi$, it follows that

$$\sqrt{n}(\widehat{\lambda}_{ACV} - \lambda_{CV})^\top \Delta\phi \xrightarrow{\text{p}} 0 \tag{1.97}$$

$$\frac{\widehat{\lambda}_{ACV}^\top \widehat{V}_{\Delta\phi} \widehat{\lambda}_{ACV}}{\lambda_{CV}^\top \widehat{V}_{\Delta\phi} \lambda_{CV}} \xrightarrow{\text{p}} 1 \tag{1.98}$$

noting that

$$t_{ADM} \equiv \frac{(\widehat{\lambda}_{ACV})^\top \Delta\phi}{\widehat{\sigma}_{ACV}/\sqrt{n}} = \frac{\sqrt{n}(\lambda_{CV})^\top \Delta\phi + \sqrt{n}(\widehat{\lambda}_{ACV} - \lambda_{CV})^\top \Delta\phi}{\widehat{\sigma}_{CV}\frac{\widehat{\lambda}_{ACV}^\top \widehat{V}_{\Delta\phi} \widehat{\lambda}_{ACV}}{\lambda_{CV}^\top \widehat{V}_{\Delta\phi} \lambda_{CV}}} \tag{1.99}$$

and hence via Slutsky's theorem

$$plim(t_{ADM}) = plim(t_{DM}). \tag{1.100}$$

Combing this with already established results from Proposition 2, both

$$t_{ADM} \xrightarrow{\text{d}} N(0,1) \tag{1.101}$$

and

$$P\left(|t_{ADM}| > c\right) \longrightarrow 1 \tag{1.102}$$

immediately follow.

Proof of Proposition 4. The proof is provided in Zhu and Timmermann (2020). Just note that stationarity of $\left\{\Delta l_{m+1}^{m,i}\right\}$ follows from the stationarity of $\{X_t\}$.

Proof of Proposition 5. From Lemma 3 it follows that $\forall k \in \{1, ..., K\}$:

$$plim\left(\sqrt{\tilde{n}}\widehat{\mathcal{L}}_{CV}^{(k)}\right) = plim\left(\sqrt{\tilde{n}}\widehat{\mathcal{L}}_{ACV}^{(k)}\right). \tag{1.103}$$

As

$$\sqrt{\tilde{n}}\left(\widehat{\mathcal{L}}_{CV}^{(1)}, ..., \widehat{\mathcal{L}}_{CV}^{(K)}\right) \xrightarrow{\text{d}} N(0, c^2 I) \tag{1.104}$$

where $c^2 = E[\Delta l_{m+1}^{m,i}] + 2\sum_{s=1}^{\infty} E[\Delta l_{m+1}^{m,i} \Delta l_{m+1}^{m,i+s}]$ (see Zhu and Timmermann (2020)), it then immediately follows that also

$$\sqrt{\tilde{n}}\left(\widehat{\mathcal{L}}_{ACV}^{(1)}, ..., \widehat{\mathcal{L}}_{ACV}^{(K)}\right) \xrightarrow{\text{d}} N(0, c^2 I). \tag{1.105}$$

The rest of the proof coincides with Zhu and Timmermann (2020).

Proof of Proposition 6. Losses of both models are:

$$\mathcal{L}_{m+1}^{m}(\mathcal{M}_1) = \mathbb{E}\left[(X_{m+1} - 0)^2\right] = \mathbb{E}\left[(c + \eta_{m+1})^2\right] = c^2 + \alpha_0 \tag{1.106}$$

$$\begin{aligned}\mathcal{L}_{m+1}^{m}(\mathcal{M}_2) &= \mathbb{E}\left[\left(X_{m+1} - \frac{1}{\widetilde{m}}\sum_{t=1}^{\widetilde{m}} X_t\right)^2\right] \\ &= \mathbb{E}\left[\left(\eta_{m+1} - \frac{1}{\widetilde{m}}\sum_{t=1}^{\widetilde{m}} \eta_t\right)^2\right] \\ &= \alpha_0 + \frac{1}{\widetilde{m}}\alpha_0 + 2\sum_{i=1}^{\widetilde{m}-1}\frac{\widetilde{m}-i}{\widetilde{m}^2}\alpha_i.\end{aligned} \tag{1.107}$$

By setting

$$\varsigma = \frac{\mathcal{L}_{m+1}^{m}(\mathcal{M}_1)}{\mathcal{L}_{m+1}^{m}(\mathcal{M}_2)} \tag{1.108}$$

and solving for c, we obtain

$$c = \left(\varsigma\left(\alpha_0 + \frac{1}{\widetilde{m}}\alpha_0 + 2\sum_{i=1}^{\widetilde{m}-1}\frac{\widetilde{m}-i}{\widetilde{m}^2}\alpha_i\right) - \alpha_0\right)^{0.5}. \tag{1.109}$$

1.8 Estimators

Estimator 1. To estimate parameters ρ and σ^2, we utilize the following moment conditions which relates to variance of contrasts differenced across different shifts $x \in \{0,\, 1 \,...,\, nv^{-1}\}$ of the estimation window:[17]

$$g_x(l_j^{m,\,iv},\, l_{j-xv}^{m,\,(i+x)v};\, \sigma^2, \rho) = \left(l_j^{m,\,iv} - l_{j-xv}^{m,\,(i+x)v}\right)^2 - \left(2\sigma^2 - 2\sigma^2\rho^x\right). \tag{1.110}$$

We normalize individual moments by the number of pairs of contrasts

$$N_x = (m + v - xv)\left(nv^{-1} - x + 1\right) \tag{1.111}$$

available and collect them to a single vector function

$$g(\phi;\, \sigma^2, \rho) = \begin{pmatrix} \frac{1}{N_0}\sum_{i=0}^{nv^{-1}-0}\sum_{j=0v+1}^{m+v} g_0(l_j^{m,\,iv},\, l_{j-0v}^{m,\,(i+0)v};\, \sigma^2, \rho) \\ \frac{1}{N_1}\sum_{i=0}^{nv^{-1}-1}\sum_{j=1v+1}^{m+v} g_1(l_j^{m,\,iv},\, l_{j-1v}^{m,\,(i+1)v};\, \sigma^2, \rho) \\ \vdots \\ \frac{1}{N_{nv^{-1}}}\sum_{i=0}^{nv^{-1}-nv^{-1}}\sum_{j=nv^{-1}v+1}^{m+v} g_{nv^{-1}}(l_j^{m,\,iv},\, l_{j-nv^{-1}v}^{m,\,(i+nv^{-1})v};\, \sigma^2, \rho) \end{pmatrix}. \tag{1.112}$$

The estimates are solution to the following optimization problem:

$$\underset{\sigma^2,\rho}{\operatorname{argmin}} \quad g(\phi;\, \sigma^2, \rho)^\top W g(\phi;\, \sigma^2, \rho) \qquad \text{with} \qquad W = \operatorname{diag}(N_0,\, N_1,\, ...,\, N_{nv^{-1}}). \tag{1.113}$$

Instead of the two stage GMM, we weight directly by the precision of each moment to reduce computational costs. Since parameter σ^2 cancels out in the optimal weight computation (see Algorithm 1), it is possible to further simplify the estimation by normalizing contrasts beforehand and performing univariate search.

[17] We also tested moments based on products of contrasts but these tend to exhibit occasional erratic behavior.

1.9 Algorithms

Algorithm 1. Our goal is to express

$$\widehat{\lambda}_{ACV} = \widehat{V}_{\phi}^{-1} B^{\top} \left(B \widehat{V}_{\phi}^{-1} B^{\top} \right)^{-1} b \tag{1.114}$$

with

$$B = \left(\mathbf{1}_{n/v}^{\top} \otimes I, \; I_{:,M} \right) \qquad b = \begin{pmatrix} \mathbf{0}_m \\ \frac{1}{v} \mathbf{1}_v \end{pmatrix} \tag{1.115}$$

and V_{ϕ} as defined in Eq. 1.23 without the need to numerically invert nor store $\widehat{V}_{\phi}$, which is a square matrix of dimension $(m+v)\frac{n}{v} + m$.

Using lemma 2, we can express

$$\widehat{V}_{\phi}^{-1} B^{\top} = \frac{1}{\hat{\sigma}^2} \begin{pmatrix} Z_1 + Z_L \\ \mathbf{1}_{n-1} \otimes (Z_U + Z_2 + Z_L) \\ (Z_U)_{M,:} + (Z_3)_{M,M} \, I_{M,:} \end{pmatrix} \equiv \begin{pmatrix} F_{1,1} \\ \mathbf{1}_{n-1} \otimes F_{1,2} \\ F_{1,3} \end{pmatrix}, \tag{1.116}$$

$$\begin{aligned} B \widehat{V}_{\phi}^{-1} B^{\top} &= \frac{1}{\hat{\sigma}^2} \left(Z_1 + Z_L + (n-1)(Z_U + Z_2 + Z_L) + I_{:,M} (Z_U)_{M,:} + I_{:,M} (Z_3)_{M,M} I_{M,:} \right) \\ &\equiv F_2 \end{aligned} \tag{1.117}$$

with

- $Z_1 = I + \frac{\hat{\rho}^2}{1-\hat{\rho}^2} L^v U^v$
- $Z_2 = I + \frac{\hat{\rho}^2}{1-\hat{\rho}^2} (L^v U^v + U^v L^v)$
- $Z_3 = \frac{1}{1-\hat{\rho}^2} I$
- $Z_U = \frac{-\hat{\rho}}{1-\hat{\rho}^2} U^v$
- $Z_L = \frac{-\hat{\rho}}{1-\hat{\rho}^2} L^v.$

This in turn allows us to compute $\widehat{\lambda}_{ACV}$ as

$$\widehat{\lambda}_{ACV} = \begin{pmatrix} F_{1,1} F_3 \\ \mathbf{1}_{n-1} \otimes (F_{1,2} F_3) \\ F_{1,3} F_3 \end{pmatrix} \qquad \text{where} \qquad F_3 = (F_2)^{-1} b, \tag{1.118}$$

which involves inversion and multiplication of matrices of dimensions no greater than $m + v$.

1.A Supplementary Results

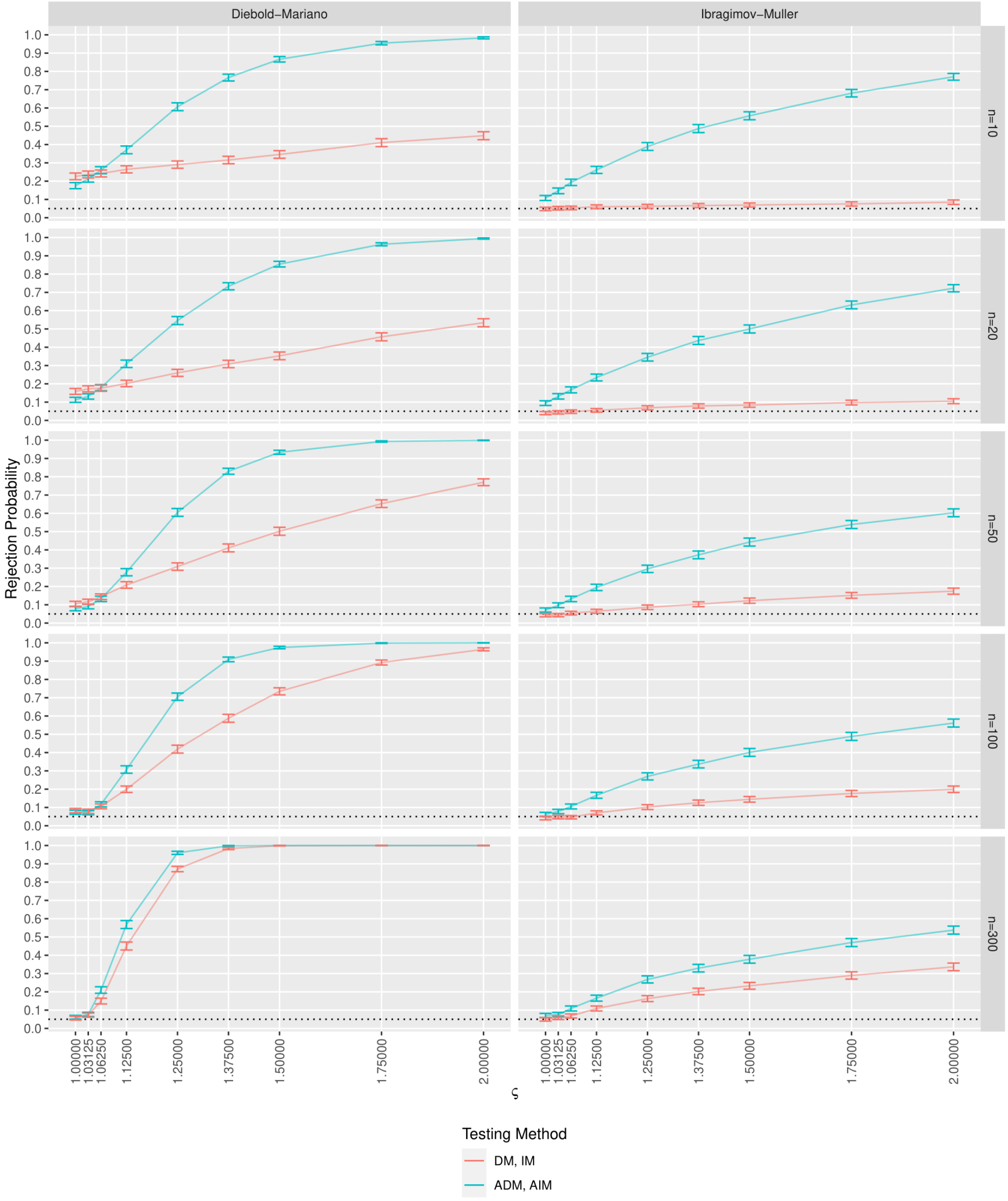


Figure 1.6: A plot of rejection probabilities for DM, IM, ADM, and AIM tests at level 0.05 for $\tau = 3$. Whiskers represent 95% confidence intervals.

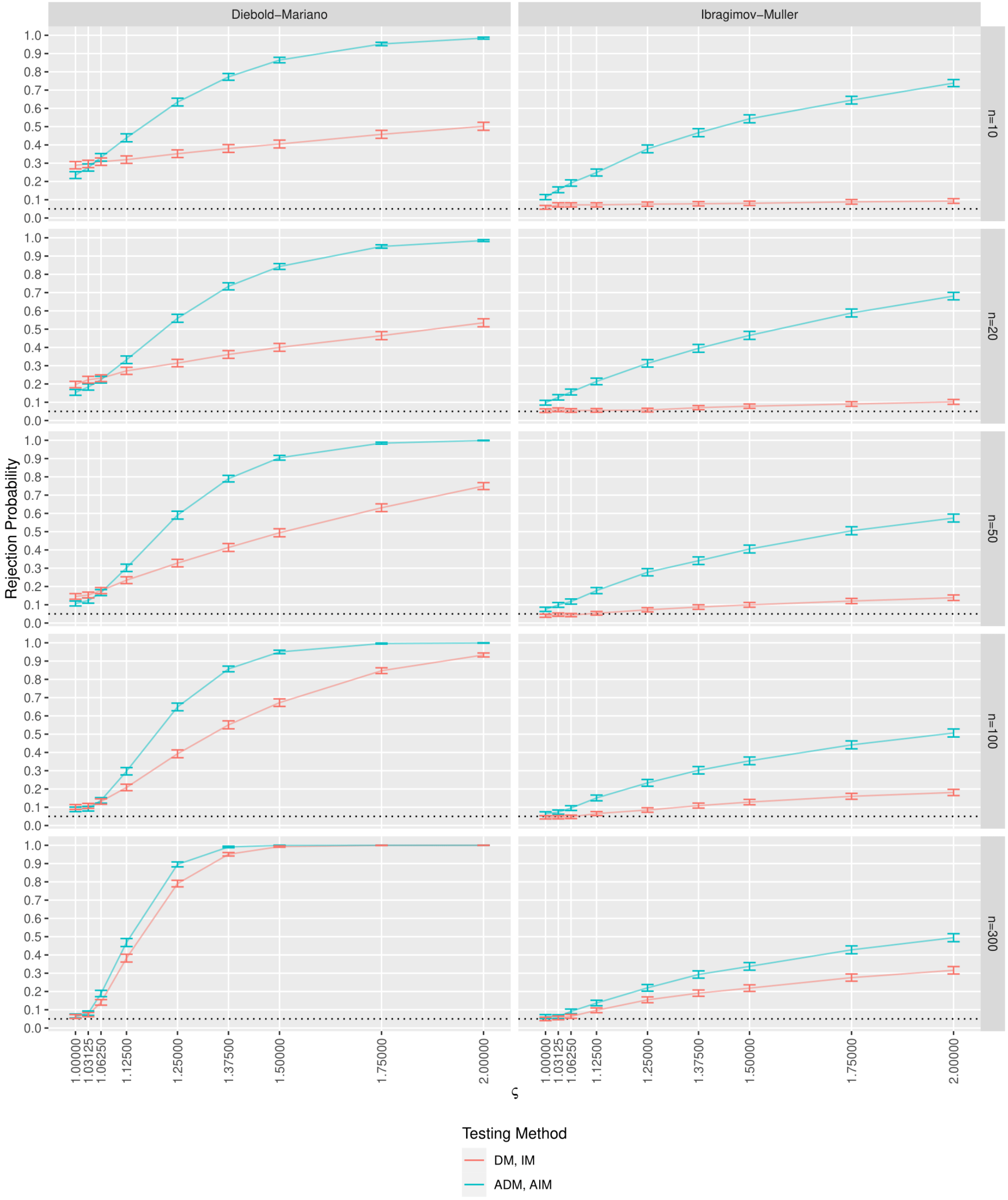


Figure 1.7: A plot of rejection probabilities for DM, IM, ADM, and AIM tests at level 0.05 for $\tau = 6$. Whiskers represent 95% confidence intervals.

time-series			ETS			autoARIMA		
Period	m	n	time(ϕ)	time($\widehat{\mathcal{L}}_{CV}$)	time($\widehat{\mathcal{L}}_{ACV}$)	time(ϕ)	time($\widehat{\mathcal{L}}_{CV}$)	time($\widehat{\mathcal{L}}_{ACV}$)
Yearly	31.324	6.000	0.201	0.001	0.143	0.559	0.001	0.143
Quarterly	92.254	8.000	4.474	0.001	0.178	1.811	0.001	0.177
Monthly	216.300	18.000	51.619	0.002	0.391	23.717	0.002	0.388
Weekly	1022.039	13.000	4.293	0.003	7.563	7.517	0.003	6.653
Daily	2357.383	14.000	10.986	0.006	109.778	8.475	0.005	104.449
Hourly	853.865	48.000	680.157	0.006	3.285	2747.418	0.006	3.294
All	240.020	12.777	29.193	0.002	4.944	23.707	0.002	4.714

Table 1.4: Comparison of run-times of $\widehat{\mathcal{L}}_{CV}$ and $\widehat{\mathcal{L}}_{ACV}$. The table displays the mean number of in-sample and out-of-sample observations m and n, and the mean run-times in seconds needed for the computation of ϕ, $\widehat{\mathcal{L}}_{CV}$, and $\widehat{\mathcal{L}}_{ACV}$.

	time-series				ETS			autoARIMA	
Period	Trending	Seasonal	N	MSE_{CV}	MSE_{ACV}	$\Delta MSE\,[\%]$	MSE_{CV}	MSE_{ACV}	$\Delta MSE\,[\%]$
Yearly			23000	97.77	88.81	-9.2***	103.20	97.54	-5.5***
				(2.84)	(2.62)		(3.38)	(3.33)	
	F	F	2214	229.60	215.83	-6.0**	305.21	301.61	-1.2
				(15.69)	(15.64)		(22.34)	(22.91)	
	F	T	267	56.98	49.67	-12.8*	57.95	57.13	-1.4
				(11.36)	(9.56)		(16.09)	(14.89)	
	T	F	15076	102.14	91.51	-10.4***	101.06	93.03	-8.0***
				(3.52)	(3.12)		(3.77)	(3.63)	
	T	T	5443	34.05	31.59	-7.2***	29.18	29.00	-0.6
				(2.45)	(2.29)		(2.73)	(2.48)	
Quarterly			24000	45.85	40.03	-12.7***	51.48	46.68	-9.3***
				(1.79)	(1.50)		(2.00)	(1.81)	
	F	F	1561	140.75	121.14	-13.9***	153.42	130.69	-14.8**
				(12.99)	(10.99)		(15.17)	(12.79)	
	F	T	681	92.94	75.12	-19.2	112.91	100.14	-11.3
				(23.19)	(15.49)		(21.06)	(18.33)	
	T	F	14115	45.73	39.64	-13.3***	48.18	43.56	-9.6***
				(2.24)	(1.93)		(2.13)	(1.81)	
	T	T	7643	22.49	21.04	-6.4	31.27	30.53	-2.4
				(1.67)	(1.52)		(3.22)	(3.38)	
Monthly			48000	27.98	25.77	-7.9***	29.93	27.61	-7.8***
				(0.63)	(0.62)		(0.70)	(0.68)	
	F	F	2574	98.69	81.02	-17.9***	107.14	91.97	-14.2***
				(6.26)	(5.45)		(6.92)	(6.43)	
	F	T	1964	29.62	26.98	-8.9*	33.68	30.22	-10.3**
				(2.70)	(2.60)		(4.15)	(3.65)	
	T	F	21613	35.80	33.43	-6.6***	37.27	34.67	-7.0***
				(1.00)	(1.04)		(1.10)	(1.09)	
	T	T	21849	11.78	11.58	-1.7	13.24	12.80	-3.3*
				(0.53)	(0.53)		(0.58)	(0.57)	
Weekly			359	13.17	9.86	-25.1**	9.90	10.60	7.1
				(2.07)	(2.04)		(1.46)	(2.15)	
	F	F	54	24.45	22.62	-7.5	13.05	11.52	-11.7
				(7.71)	(11.45)		(2.72)	(2.15)	
	F	T	3	2.44	1.48	-39.3	0.43	0.38	-11.7
				(1.41)	(0.73)		(0.28)	(0.20)	
	T	F	257	11.77	7.85	-33.3**	9.81	11.35	15.7
				(2.27)	(1.44)		(1.88)	(2.95)	
	T	T	45	8.31	6.58	-20.9	7.26	5.92	-18.4
				(3.87)	(2.65)		(3.11)	(2.26)	
Daily			4227	4.86	4.66	-4.0	7.75	7.98	3.0
				(1.22)	(1.26)		(2.30)	(2.40)	
	F	F	226	4.10	5.37	30.8	5.59	5.60	0.2
				(2.82)	(4.46)		(3.93)	(4.38)	
	F	T	19	1.19	1.23	3.7	1.16	1.27	9.8
				(0.79)	(0.90)		(0.83)	(0.92)	
	T	F	3535	1.92	1.63	-14.9**	1.89	1.65	-13.1**
				(0.52)	(0.52)		(0.53)	(0.54)	
	T	T	447	28.66	28.42	-0.8	55.42	59.59	7.5
				(10.64)	(10.88)		(21.12)	(22.03)	
Hourly			414	23.89	17.93	-24.9***	86.50	76.90	-11.1
				(3.57)	(2.50)		(18.51)	(19.59)	
	F	F	1	0.97	0.68	-29.7	0.01	0.01	-15.4
				(NA)	(NA)		(NA)	(NA)	
	F	T	125	51.78	36.26	-30.0***	149.95	110.10	-26.6***
				(9.40)	(5.98)		(37.47)	(31.12)	
	T	F	5	4.09	4.37	6.7	2.52	2.85	13.1
				(3.78)	(4.17)		(1.93)	(2.13)	
	T	T	283	12.00	10.13	-15.5	60.26	63.81	5.9
				(2.89)	(2.38)		(21.26)	(25.13)	
All			**100000**	**47.27**	**42.71**	**-9.7*** **	**51.18**	**47.58**	**-7.0*** **
				(0.84)	**(0.77)**		**(0.99)**	**(0.95)**	

Table 1.5: Comparison of $\widehat{\mathcal{L}}_{CV}$ and $\widehat{\mathcal{L}}_{ACV}$ in terms of the loss estimation for forecast horizon τ up to 3.
$\Delta MSE\,[\%] = \frac{MSE_{ACV}-MSE_{CV}}{MSE_{CV}}100$. Standard errors in brackets,
$***p < 0.001$, $**\ p < 0.01$, $*p < 0.05$.

	time-series				ETS			autoARIMA	
Period	Trending	Seasonal	N	MSE_{CV}	MSE_{ACV}	$\Delta MSE\,[\%]$	MSE_{CV}	MSE_{ACV}	$\Delta MSE\,[\%]$
Yearly			23000	151.91 (3.82)	150.46 (3.79)	-1.0	151.66 (4.20)	154.00 (4.33)	1.5
	F	F	2214	319.14 (20.30)	321.68 (21.45)	0.8	402.81 (27.49)	415.07 (28.76)	3.0
	F	T	267	124.49 (23.61)	117.87 (22.36)	-5.3	116.15 (26.05)	118.38 (24.57)	1.9
	T	F	15076	157.22 (4.72)	154.43 (4.53)	-1.8	149.01 (4.69)	149.57 (4.78)	0.4
	T	T	5443	70.53 (4.11)	71.44 (4.23)	1.3	58.58 (3.67)	61.82 (3.73)	5.5*
Quarterly			24000	71.74 (2.54)	67.99 (2.33)	-5.2***	77.85 (2.73)	75.27 (2.68)	-3.3*
	F	F	1561	208.50 (18.54)	194.95 (17.48)	-6.5	208.44 (19.66)	186.77 (17.54)	-10.4*
	F	T	681	115.34 (20.29)	104.09 (16.94)	-9.8	136.70 (23.21)	129.78 (23.36)	-5.1
	T	F	14115	75.65 (3.43)	70.89 (3.09)	-6.3***	79.94 (3.34)	77.01 (3.18)	-3.7*
	T	T	7643	32.72 (2.20)	33.50 (2.26)	2.4	42.07 (3.82)	44.42 (4.35)	5.6
Monthly			48000	42.91 (0.88)	40.68 (0.88)	-5.2***	45.15 (0.95)	43.06 (0.94)	-4.6***
	F	F	2574	131.07 (8.06)	112.19 (7.20)	-14.4***	139.13 (8.77)	123.49 (8.30)	-11.2***
	F	T	1964	37.86 (3.61)	36.00 (3.51)	-4.9	44.71 (5.63)	40.78 (5.10)	-8.8*
	T	F	21613	57.87 (1.43)	55.29 (1.51)	-4.5***	58.95 (1.51)	56.52 (1.52)	-4.1***
	T	T	21849	18.19 (0.79)	18.22 (0.81)	0.2	20.47 (0.86)	20.48 (0.87)	0.0
Weekly			359	18.05 (2.47)	15.17 (2.45)	-16.0*	16.28 (2.59)	18.24 (3.61)	12.0
	F	F	54	33.16 (8.83)	29.90 (11.71)	-9.8	18.80 (3.82)	16.86 (3.16)	-10.3
	F	T	3	4.25 (1.14)	3.27 (0.56)	-23.1	6.26 (3.36)	5.47 (3.17)	-12.6***
	T	F	257	14.99 (2.50)	11.80 (1.97)	-21.2	15.79 (3.32)	18.96 (4.87)	20.1
	T	T	45	18.36 (8.32)	17.47 (7.37)	-4.8	16.68 (6.97)	16.61 (6.52)	-0.5
Daily			4227	13.12 (3.35)	12.93 (3.43)	-1.5	22.83 (6.14)	23.60 (6.44)	3.4
	F	F	226	6.66 (3.65)	7.94 (5.49)	19.3	9.10 (4.97)	8.79 (5.29)	-3.5
	F	T	19	3.28 (2.56)	3.40 (2.87)	3.8	3.51 (2.46)	3.36 (2.50)	-4.2
	T	F	3535	4.02 (1.04)	3.54 (1.01)	-11.9*	4.08 (1.08)	3.67 (1.06)	-9.8
	T	T	447	88.81 (30.35)	90.08 (31.10)	1.4	178.95 (56.86)	189.55 (59.77)	5.9
Hourly			414	36.26 (5.00)	32.67 (4.10)	-9.9	106.11 (20.88)	97.04 (22.04)	-8.5
	F	F	1	1.83 (NA)	1.18 (NA)	-35.4	0.01 (NA)	0.00 (NA)	-93.3
	F	T	125	81.20 (13.45)	71.84 (10.86)	-11.5	193.90 (45.80)	152.58 (39.93)	-21.3**
	T	F	5	3.39 (2.73)	2.75 (2.19)	-19.0	3.43 (1.58)	3.63 (1.65)	5.9
	T	T	283	17.10 (3.72)	16.01 (3.10)	-6.4	69.53 (22.55)	74.50 (26.91)	7.2
All			**100000**	**73.53** **(1.17)**	**71.19** **(1.14)**	**-3.2*****	**76.70** **(1.29)**	**75.62** **(1.31)**	**-1.4***

Table 1.6: Comparison of $\widehat{\mathcal{L}}_{CV}$ and $\widehat{\mathcal{L}}_{ACV}$ in terms of the loss estimation for forecast horizon τ up to 6.
$\Delta MSE\,[\%] = \frac{MSE_{ACV} - MSE_{CV}}{MSE_{CV}} 100$. Standard errors in brackets,
$***p < 0.001$, $**\ p < 0.01$, $*p < 0.05$.

Chapter 2

Unrestricted, Restricted, and Regularized Models for Forecasting Multivariate Volatility

This chapter was coauthored with Stanislav Anatolyev (CERGE-EI).

Originally published as:

Anatolyev, S., Staněk, F. (2022) "Unrestricted, Restricted, and Regularized Models for Forecasting Multivariate Volatility", *Studies in Nonlinear Dynamics & Econometrics*, 2023, 27.2: 199-218.

2.1 Introduction

The knowledge of a covariance matrix of multivariate returns distribution is essential for many tasks such as portfolio allocation, risk management, derivative pricing, analysis of financial contagion, and so on. This has led to development of a variety of models for forecasting covariance matrices such as the constant conditional correlation model (CCC) of Bollerslev (1990), the dynamic conditional correlation model (DCC) of Engle (2002), the BEKK model (Engle and Kroner, 1995) and extensions thereof. More recently, with the machinery allowing the estimation of low frequency volatility from high frequency data (Andersen et al., 2003; Barndorff-Nielsen and Shephard, 2004), ideas behind these models were translated to realized covariance matrices. The Conditional Autoregressive Wishart model (CAW henceforth) of Golosnoy et al. (2012) utilizes directly realized volatilities and co-volatilities and models them via a BEKK-style dynamic equation. Among extensions of BEKK/CAW are asymmetric BEKK Caporin and McAleer (2014), proximity-based structured GARCH (Caporin and Paruolo, 2015), threshold CAW Anatolyev and Kobotaev (2018), and others.

The dynamic equation in these models is constructed to simultaneously attain two, somewhat conflicting objectives – to accurately capture both temporal and cross-sectional dependencies in return (co-)volatilities, and to maintain a reasonable model parsimony. In this regard, both the BEKK and CAW models are available in three variations of different complexity: a full BEKK/CAW where the law of motion is parameterized by unrestricted parameter matrices, a diagonal BEKK/CAW where these matrices are set to be diagonal, and a scalar BEKK/CAW where these matrices are identity matrices multiplied by scalars. While the full model offers highest flexibility in capturing the data generating process, it suffers from a curse of dimensionality as the number of parameters determining the law of motion increases at the rate $O(n^2)$ in the number of assets n, which frequently results in imprecise estimation of model parameters and, as a consequence, in poor out-of-sample predictive performance. Due to this demerit, researchers frequently opt for the diagonal or even the scalar model (see e.g.: Caporin and McAleer, 2012; Laurent et al., 2012; Zhipeng and Shenghong, 2018; Zolfaghari et al., 2020) rather than the full model.

In this chapter, we address the curse of dimensionality present in BEKK/CAW models by allowing for a smooth transition among differently parameterized models. Namely, we frame all three existing model variations – full, diagonal, scalar – as special cases of a regularized full model. The regularized estimator applies a combination of the standard ridge regularization towards zero (Hoerl and Kennard, 1970) that drives the full model towards the diagonal model, and the ridge regularization towards homogeneity (Anatolyev, 2020) that drives the diagonal model towards the scalar model. Thus, the regularized estimator naturally nests all three benchmarks – scalar, diagonal, full – and is hence capable of optimally selecting among them, or between any of their combinations. This allows us to assess the optimal degree of cross-sectional dependence or non-homogeneity that helps forecasting performance.

We perform an extensive battery of out-of-sample forecast evaluations on Noureldin et al. (2012) data-set of realized stock market co-volatilities of up to ten assets. We trace the influence of a number of assets n, as well as other factors affecting forecasting performance, such as a length of the estimation window and the recency of estimated coefficients relative to the forecasted period. Furthermore, we analyze the in-sample performance to assess the degree, to which the additional flexibility of the diagonal and the full model helps to account for the volatility dynamics.

Our experiments confirm the general superiority of more restricted models. The performance of the full model deteriorates for higher n, while among the scalar, diagonal and optimally regularized models, the diagonal model seems to be preferred, thought the evidence is noisy and the margin is small. The regularization does not seem to bring perceptible improvements indicating that cross-sectional dependencies are of limited relevance. This can be attributed to a need to tune the regularization intensities, but even in experiments with non-feasible optimal regularization, one can see that the maximal achievable gains from regularization tend to be below 1%. Additionally, we observe that increasing the length of the estimation window does not translate to more precise predictions and that forecasting performance rapidly deteriorates as we increase the distance between the forecasted period and the window on which parameters are estimated. The superiority of more parsimonious scalar and diagonal models is also confirmed by analysis of the in-sample performance. The additional flexibility of the full model delivers only a very modest reduction of the correlation of transformed residuals.

The remainder of the chapter is structured as follows. Section 2.2 describes the canonical models as well as the regularized estimator. Section 2.3 presents the empirical evaluation; Subsection 2.3.1 describes the design of the empirical evaluation, Subsection 2.3.2 presents results from a battery of out-of-sample forecasting evaluations, and Subsection 2.3.3 assess the in-sample performance. Section 2.4 concludes. The Appendix contains tables and figures with supplementary results.

2.2 Methodology

2.2.1 Canonical Models

Let us consider n assets living through time periods $t = 1, \ldots, T$, and let $\mathcal{F}_t$ denote observable information at t. The BEKK model (Engle and Kroner, 1995) describes evolution of conditional second moments of the n-vector of returns, while the CAW model (Golosnoy et al., 2012) describes evolution of its conditional first moments of the $n \times n$-matrix of realized co-volatilities. However, the structure of their dynamic equations for the object of interest is the same. This object of interest is a matrix $R_t \in \mathbb{R}^{n^2}$ that represents, in the case of BEKK, the outer product of demeaned returns, and in the case of CAW, the realized co-volatility matrix.

The BEKK/CAW model postulates the following law of motion for the conditional expectation of R_t denoted $S_t = \mathbb{E}\left[R_t|\mathcal{F}_{t-1}\right]$. The canonical BEKK$(p,q)$/CAW$(p,q)$ model reads

$$S_t = CC^\top + \sum_{i=1}^{q} A_i R_{t-i} {A_i}^\top + \sum_{i=1}^{p} B_i S_{t-i} {B_i}^\top. \tag{2.1}$$

Here, in the case of the BEKK model, $R_t = (r_t - \mu)(r_t - \mu)^\top$, and $r_t|\mathcal{F}_{t-1} \sim N(\mu, S_t)$ is a vector of n returns. In the case of the CAW model, R_t is a realized volatility matrix computed from high frequency data, $R_t|\mathcal{F}_{t-1} \sim W_n(v,\ S_t/v)$, where $W_n(v,\ S)$ represents an n-dimensional Wishart distribution with v degrees of freedom and a scale matrix S.

The lower-triangular parameter matrix $C \in \mathbb{R}^{n^2}$ and general parameter matrices $A_i \in \mathbb{R}^{n^2}$ and $B_i \in \mathbb{R}^{n^2}$ determine the law of motion for volatility and are to be estimated along with the other parameters.[1] This specification offers several advantages. First, it guarantees, by construction, the positive semidefinitness and symmetry of volatility predictions S_t without requiring parameter restrictions on matrices $\{C, \{A_i\}_{i=1}^q, \{B_i\}_{i=1}^p\}$. Second, in its most general form, it allows one to model various dependencies between the current volatility and past innovations or past volatility predictions across different assets through the off-diagonal elements of matrices A_i or B_i, respectively.

In practice, however, restrictions on parameters are frequently made in order to reduce the estimation noise at the cost of more probable misspecification. The most commonly used model variations are the following (the abbreviations corresponding to the CAW class):

1. fCAW: the full model with general parameter matrices: $A_i \in \mathbb{R}^{n^2}$ for $1 \leq i \leq q$ and $B_i \in \mathbb{R}^{n^2}$ for $1 \leq i \leq p$, resulting in $n^2(p+q) = O(n^2)$ parameters in matrices A and B.

2. dCAW: the diagonal model with zero restrictions on the off-diagonal elements: $A_i =$

[1] We maintain the standard assumption that the intercept matrix C is populated by $\frac{1}{2}n(n+1)$ parameters (Engle and Kroner, 1995; Golosnoy et al., 2012). For unique parameter identification, it is conventional and convenient to restrict all the diagonal elements of C and the first diagonal elements of all matrices A_i and B_i to be positive (Engle and Kroner, 1995).

dg $\{a_{i,1}, a_{i,2}, \ldots, a_{i,n}\}$ for $1 \leq i \leq q$ and $B_i = \text{dg}\{b_{i,1}, b_{i,2}, \ldots, b_{i,n}\}$ for $1 \leq i \leq p$, resulting in $n(p+q) = O(n)$ parameters in matrices A and B.

3. sCAW: the scalar model with zero restrictions on the off-diagonal elements and equality restrictions across the diagonal elements: $A_i = a_i I_n$ for $1 \leq i \leq q$ and $B_i = b_i I_n$ for $1 \leq i \leq p$, resulting in $p+q = O(1)$ parameters in matrices A and B.

While the full BEKK/CAW offers highest flexibility, it is rarely used in practice due to its excessive parametrization – for example, the full CAW$(1,1)$ model with 10 assets requires estimation of 256 parameters. Instead, researchers frequently opt for the diagonal or even scalar models.

2.2.2 Regularized Model

Maximum Likelihood Estimation

The parameters of equation (2.1) are estimated by the method of maximum likelihood. Let us denote the model parameters by θ, and the log-likelihood for one observation by $\ell\ell_t$. Then, $\theta = \{C, \{A_i\}_{i=1}^q, \{B_i\}_{i=1}^p, \mu\}$ for the BEKK, and $\theta = \{C, \{A_i\}_{i=1}^q, \{B_i\}_{i=1}^p, v\}$ for the CAW. The likelihood for observation t is

$$\ell\ell_t = \log\left((2\pi)^{-\frac{k}{2}} |S_t|^{-\frac{1}{2}}\right) - \frac{1}{2}\text{tr}\Big(S_t^{-1} R_t\Big)$$

for the BEKK, and

$$\ell\ell_t = \log\left(\frac{|S_t v^{-1}|^{-\frac{v}{2}} |R_t|^{\frac{v-n-1}{2}}}{2^{\frac{vn}{2}} \pi^{\frac{n(n-1)}{4}} \prod_{i=1}^n \Gamma\left(\frac{v+1-i}{2}\right)}\right) - \frac{1}{2}\text{tr}\Big(v S_t^{-1} R_t\Big)$$

for the CAW. The maximum likelihood estimator solves the optimization problem

$$\widehat{\theta}_{ML} = \arg\max_{\theta} \sum_{t=1}^{T} \ell\ell_t \tag{2.2}$$

subject to the evolution equation (2.1) and possibly additional parameter restrictions listed in subsection 2.2.1. The fCAW/fBEKK, dCAW/dBEKK and sCAW/sBEKK estimates emerge depending on which additional constraints are imposed.

Penalized Estimation

Regularization augments the log-likelihood in the optimization problem (2.2) by two ridge-type (i.e., relative to the L_2-norm) penalty terms.[2] The first penalty punishes for deviations of the off-diagonal elements from the zero value, providing regularization of fBEKK/fCAW towards dBEKK/dCAW corresponding to classical "ridging towards zero" (Hoerl and Kennard, 1970):

$$\tau_f = \sum_{i=1}^{q}\sum_{j=1}^{n}\sum_{k\neq j}^{n} A_{i,j,k}^2 + \sum_{i=1}^{p}\sum_{j=1}^{n}\sum_{k\neq j}^{n} B_{i,j,k}^2. \tag{2.3}$$

The second penalty punishes for deviations of the diagonal elements from the common value, providing regularization of dBEKK/dCAW towards sBEKK/sCAW corresponding to "ridging towards homogeneity" (Anatolyev, 2020):

$$\tau_d = \sum_{i=1}^{q}\sum_{j=1}^{n}\left(A_{i,j,j} - \frac{1}{n}\sum_{k=1}^{n} A_{i,k,k}\right)^2 + \sum_{i=1}^{p}\sum_{j=1}^{n}\left(B_{i,j,j} - \frac{1}{n}\sum_{k=1}^{n} B_{i,k,k}\right)^2. \tag{2.4}$$

This results in the following regularized maximum likelihood estimator:

$$\widehat{\theta}_{RML} = \arg\max_{\theta}\sum_{t=1}^{T}\left\{\ell\ell_t - \lambda_f\tau_f - \lambda_d\tau_d\right\} \tag{2.5}$$

subject to the evolution equation (2.1). The hyper-parameters λ_f and λ_d control the degree of regularization applied to the off-diagonal and diagonal matrix elements, respectively: λ_f is the intensity of ridging of fBEKK/fCAW towards dBEKK/dCAW, and λ_d is the intensity of ridging of dBEKK/dCAW towards sBEKK/sCAW.

This specification naturally nests the full BEKK/CAW ($\lambda_d = 0$, $\lambda_f = 0$), the diagonal BEKK/CAW ($\lambda_d = 0$, $\lambda_f \to \infty$), and the scalar BEKK/CAW ($\lambda_d \to \infty$, $\lambda_f \to \infty$). Apart from these three extreme cases, it also allows for intermediate states. We abbreviate this model and technique corresponding to the BEKK/CAW class by rBEKK/rCAW, where 'r' stands for 'regularized'.

[2] Another option would be LASSO (L1) regularization. While we did not experiment with this scheme, we anticipate that the results would be qualitatively similar to ridge regression, as LASSO and ridge often exhibit comparable behavior in terms of prediction accuracy in high-dimensional settings (see, e.g., Kim, 2014; Pereira et al., 2016).

Feasible Penalization

In practice, the hyper-parameters λ_d and λ_f are not known ex-ante and need to be tuned. We follow the usual tradition and select them a via fixed scheme time-series cross-validation (see Clark and McCracken, 2013a): given a pair of candidate values of hyper-parameters $\lambda_d \in \Lambda_d = \{\lambda_{d,1}, \lambda_{d,2}, \ldots, \lambda_{d,k_d}\}$ and $\lambda_f \in \Lambda_f = \{\lambda_{f,1}, \lambda_{f,2}, \ldots, \lambda_{f,k_f}\}$, the model is estimated on a window of data $\{1, 2, \ldots, T_0\}$, and then evaluated on the remaining $\{T_0 + 1, T_0 + 2, \ldots, T\}$ observations using a desired loss function; we use the Stein loss (see Section 2.3). The optimal values of hyper-parameters are selected so that they minimize the loss incurred in the validation sample $\{T_0 + 1, T_0 + 2, \ldots, T\}$.

As computational complexity of large scale numerical optimization unfortunately makes an exhaustive grid search for optimal values of λ_d and λ_f impractical even for moderately sparse sets Λ_d and Λ_f, we opt for a sequential search for the optimal hyper-parameters, starting with Λ_d and only then proceeding to Λ_f. Furthermore, one is able to further reduce the run-time of numerical optimizations by sorting the sets Λ_d and Λ_f in descending order and then proceeding so that the parameters estimated for $\lambda_{d,i}$ (respectively, $\lambda_{f,i}$) form a starting point of numerical optimization for $\lambda_{d,i+1}$ (respectively, $\lambda_{f,i+1}$), effectively tracing the optimal $\widehat{\theta}$ as an approximately continuous function of λ_d (resp. λ_f) rather than performing a search along a long optimization path from a common starting point. This procedure is analogous to the practice of using estimates of more restricted models of BEKK as starting values for more flexible models (Engle and Kroner, 1995) but applied to a gradually changing degree of regularization rather than the extreme models themselves. These measures make the regularized estimator not too much more computationally expensive than the standard full BEKK/CAW.

Moreover, the fact that optimization on a less constrained parameter space utilizes solutions found in a more restricted space as starting points reduces the possibility of converging to a sub-optimal local maximum, an understandable concern given the scale of our optimization problems. This is so because, by design, optimization in the less constrained parameter space cannot ever attain a worse solution than the best solution already found in the more restricted space including that of the most parsimonious sBEKK/sCAW, effectively ruling out a whole range of potential sub-optimal local maxima.[3]

[3] A MATLAB package building upon the MFE toolbox (Sheppard, 2013) implementing the regularized estimator for both BEKK and CAW is available at `https://github.com/stanek-fi/RMV`.

2.3 Empirical Evaluation

2.3.1 Evaluation Design

Throughout this section, we restrict our attention to volatility modeling via the CAW class of models for realized covariance matrices. The results for BEKK are qualitatively similar; these results are available upon request. The main reason of focusing on the CAW class is a wide availability of high frequency return data for modeling realized variances, whose use generally improves forecast precision relative to GARCH-type models (see, e.g., Andersen et al., 2003; Golosnoy et al., 2012). In addition, the use of models for realized volatility allows higher quality evaluation of forecasts as realized volatility is observable. Furthermore, we set $q = p = 1$ as that represents arguably the most commonly used model specification and because it also reduces computational burden.

Let $\widehat{R}_t$ be a forecast of R_t made in the previous period. The Stein loss (James and Stein, 1961) that we exploit here for forecast evaluation,

$$L_S\Big(R_r, \widehat{R}_t\Big) = \operatorname{tr}\Big(\widehat{R}_t R_r\Big) - \log\Big(\det\Big(\widehat{R}_t R_r\Big)\Big) - n,$$

is coherent with the Wishart likelihood and is robust to volatility measurement in the sense of Patton (2011). Along with the Stein loss, we exploit the Frobenius loss function

$$L_F\Big(R_r, \widehat{R}_t\Big) = \operatorname{tr}\left(\Big(R_r - \widehat{R}_t\Big)\Big(R_r - \widehat{R}_t\Big)^\top\right).$$

This is (along with a very similar Euclidean loss, see Laurent et al., 2012) a multivariate extension of the quadratic loss in the scalar case, and is also robust in the sense of Patton (2011). Among the two, the Stein loss generally exhibits less erratic behavior hence allowing us to draw finer conclusions.[4]

We use the popular realized stock market volatility data-set from Noureldin et al. (2012) covering 10 stocks (BAC, JPM, IBM, MSFT, XOM, AA, AXP, DD, GE, KO) from 2001-02-01 to 2009-12-31. In order to attain maximal external validity and to explore different factors which might affect forecasting performance, we employ the following experimental design, which also reflects on practitioners' decisions to employ training samples of arbitrary sizes. For each of $n \in \{2, 3, \ldots, 10\}$, 4 distinct combinations of stocks

[4] We report the results for the Stein loss in the main text, and those for the Frobenius loss in the Appendix.

of size n are randomly selected (except the case $n = 10$ when only one combination is available). For each combination, training windows of sizes $T \in \{2,\ 2.5,\ 3,\ 3.5,\ 4,\ 4.5,\ 5\}$ years[5] are rolled through the sample with increment 0.5 year. For each position of the window, a day ahead forecasts for the following 0.5 year are produced via estimated scalar, diagonal, full, and regularized CAW (recall the acronyms sCAW, dCAW, fCAW and rCAW, respectively). Finally, the quality of the forecasts is measured via the Stein and Frobenius losses, and significance of the observed differences is tested via model confidence sets (MCS) (Hansen et al., 2011). This setting assures a substantial variation in both the estimation and evaluation data as well as in the assets under consideration. In the case of rCAW, the regularization hyper-parameters λ_d and λ_f are selected from $\Lambda_d = \Lambda_f = \{\kappa_1, \cdots \kappa_8\}$, where $\kappa_i = \Big(\exp(2(i-1)) - 1\Big)/10$ for $i = 1, \ldots, 7$ and $\kappa_8 = \infty$,[6] by evaluation on the validation set – the last 0.5 year of data in the training window. The optimization itself is performed via a Newton-type optimizer with constraints accounted for with the use of an active-set method. The optimality tolerance is set to 10^{-6} to minimize the possibility of premature termination in almost flat regions of the objective function. To reduce computational requirements, the BFGS algorithm suitable for large scale problems such as this one is utilized in computation of Hessian updates.

2.3.2 Out-of-Sample Performance

As can be seen in Figure 2.1 displaying the average volatility alongside the average performance of individual models and the selected λ_d and λ_f, the data-set spans two volatile periods – the US stock market downturn of 2002 and the US housing crisis – as well a relatively calm period from 2003 to 2007. Clearly, the out-of-sample performance of all models as measured by the Stein loss deteriorates during the volatile housing crisis. This is especially true for the fCAW whose relative performance, as measured by the ratio of out-of-sample Stein loss (with the rCAW being the benchmark) and the average ranking among the models, deteriorates when trained or evaluated on highly volatile periods.

This fact is also mirrored by the optimal hyper-parameters λ_d and λ_f selected, as we can see that more stringent regularization is being chosen during the US housing crisis

[5] Due to space considerations, we choose not to display separate results for estimation windows of non-integer sizes $T \in \{1.5,\ 2.5,\ 3.5,\ 4.5\}$ years, thou they still do enter aggregate computations. Results for these intermediate window lengths are, as expected, intermediate, and are available upon request.

[6] This scale was purposefully chosen so that it allows for very mild up to stringent penalties. The cases $\lambda_d = \infty$ or $\lambda_f = \infty$ are implemented via corresponding restrictions on the likelihood maximization problem.

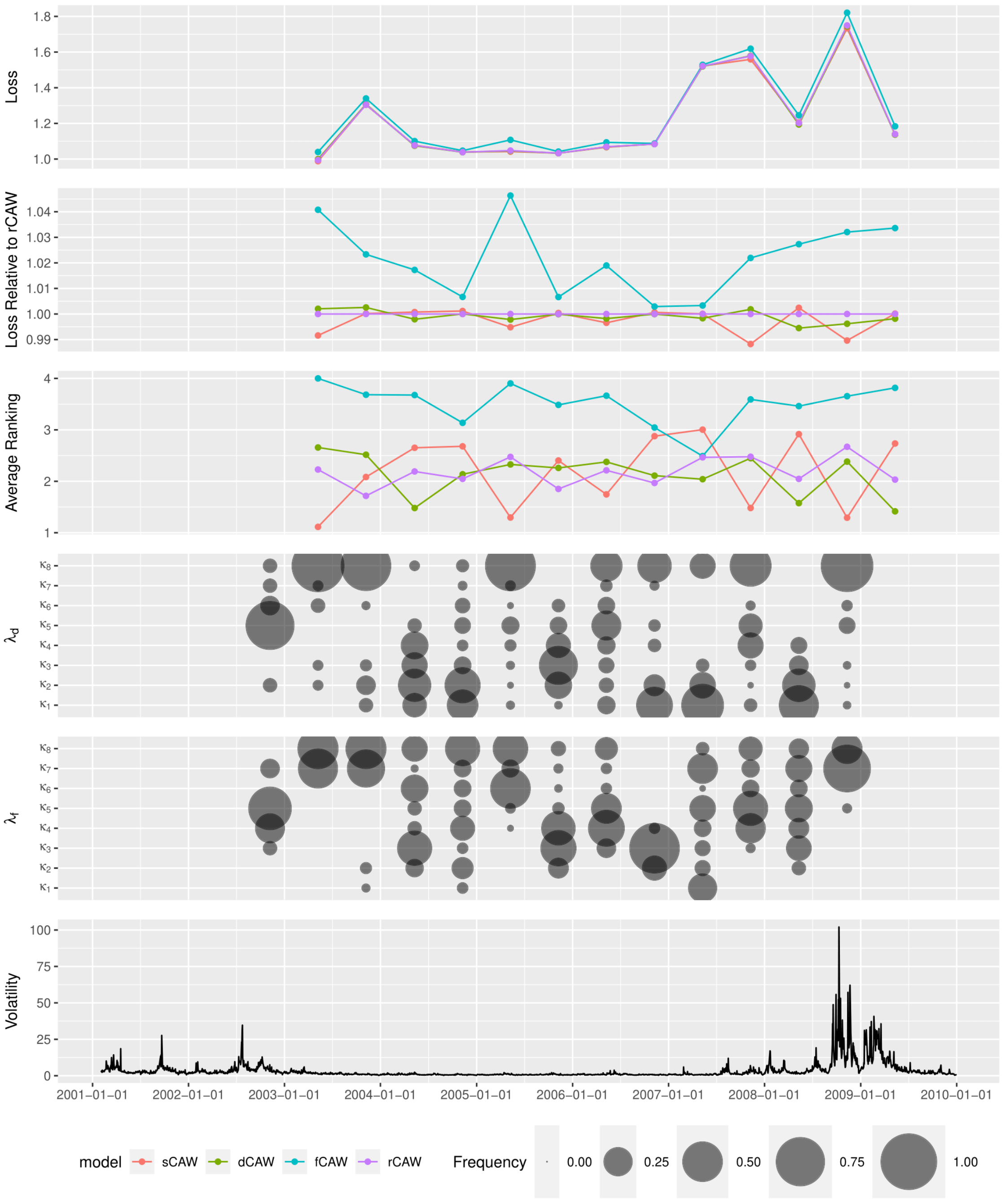


Figure 2.1: The three upper panels display the out-of-sample Stein loss, ratio of Stein losses (where rCAW is the benchmark), average ranking (with 1 indicating the best model and 4 the worst model), for individual models as a function of time. The three lower panels display the optimal regularization hyper-parameters λ_d and λ_f selected, alongside with the average realized volatility, as a function of time.

whereas less stringent regularization corresponding to lower λ_d (respectively, λ_f) is being chosen whenever dCAW (respectively, fCAW) performs notably well (e.g., during the 2003-2007 tranquil period). This variability over time indicates that possible cross-sectional dependencies might not be stable. Overall, however, we see that complete (κ_8) or very stringent regularization (κ_7) are most frequent for both the diagonal and off-diagonal elements.

Figure 2.2 displays in-sample and out-of-sample Stein losses for individual models relative to the loss of rCAW. As to be expected, more parameterized models achieve better in-sample performance, especially pronounced when the estimation window is short (which presumably reflects higher within-window stability). However, the magnitude of the observed differences between individual models is relatively small; for estimation windows larger than or equal to three years, it is generally within 1%. As for the rCAW, which optimally selects the degree of parameterization, the in-sample fit is comparable to that of the dCAW. With regard to the out-of-sample Stein losses, the sCAW, dCAW, and rCAW perform similarly (well within 1%), outperforming the much more parameterized fCAW on average by 1%–6%, depending on a length of the training window.

To be able to discern these relatively small differences between the sCAW, dCAW, and rCAW, Tables 2.1 and 2.2 provide a more detailed assessment of the out-of-sample performance broken down to individual combinations of the number of assets n and the training window length T. Table 2.1 depicts the average ranking of individual models as measured by the Stein loss. The average ranking of all the three models is in the neighborhood of 2.1, though slightly favoring the dCAW over the rCAW and sCAW, with the difference being somewhat more sizable for longer training windows. Table 2.2 depicts the frequency of rejection via the MCS at the 5% level for individual models, thus focusing on how often the models perform notably badly. According to this measure, the dCAW seems to be a slightly safer option relative to the sCAW or rCAW for situations with more assets and longer estimation windows, though it should be noted that the evidence is again not entirely conclusive. The most flexible fCAW model is, on average, rejected in more than half of the cases.

The fact that rCAW largely fails to outperform dCAW, merely matching its performance, can be attributed to two factors. First, it appears that sCAW and dCAW perform similarly without much gain obtained via optimal selection of λ_d. With respect to λ_f, the performance appears to quickly decline as we move towards less stringent regularization with no apparent intermediate region of improved performance relative to dCAW. In order to see this, we have performed an additional experiment. Figure 2.3a depicts the performance of sCAW and dCAW relative to the performance of *non-feasible* optimally diagonally regularized dCAW, with λ_d chosen to be *ex-post optimal* in terms of loss. The maximal achievable gains from diagonal regularization are generally below 1%. Furthermore, the potential gains from regularization of off-diagonal elements relative to dCAW are well bellow 0.3% as can be seen from Figure 2.3b depicting the performance of dCAW and fCAW relative to the performance of the unfeasible optimally off-diagonally regularized fCAW. Clearly, the off-diagonal regularization, no matter how stringent, fails to substantially improve dCAW even when we abstract from tuning the optimal λ_f.

Second, the quickly changing nature of the process (see Figure 2.1) makes it difficult to choose the optimal λ_d and λ_f and hence reap the already small improvements stemming from regularization as displayed in Figures 2.3a and 2.3b.

	n=2	n=3	n=4	n=5	n=6	n=7	n=8	n=9	n=10	**all**
2y	2.15	2.14	1.97	2.15	1.92	1.86	1.87	1.88	1.89	**1.98**
3y	2.43	2.33	2.41	2.26	2.13	2.00	2.05	2.15	2.05	**2.21**
4y	2.53	2.42	2.38	2.43	2.35	2.29	2.22	2.27	2.11	**2.35**
5y	2.70	2.52	2.43	2.75	2.71	2.55	2.39	2.28	2.36	**2.52**
all	**2.431**	**2.305**	**2.299**	**2.371**	**2.224**	**2.157**	**2.152**	**2.135**	**2.129**	**2.25**

(a) sCAW

	n=2	n=3	n=4	n=5	n=6	n=7	n=8	n=9	n=10	**all**
2y	2.26	2.14	2.28	1.94	2.02	2.19	2.10	2.05	2.00	**2.11**
3y	2.26	2.23	2.25	2.11	2.07	2.23	2.09	2.02	1.91	**2.14**
4y	2.28	2.22	2.22	2.03	2.03	1.97	1.94	2.07	2.00	**2.09**
5y	2.14	2.14	2.21	1.89	1.82	1.68	1.68	1.59	1.57	**1.87**
all	**2.283**	**2.249**	**2.291**	**2.032**	**1.95**	**2.071**	**1.993**	**1.955**	**1.943**	**2.09**

(b) dCAW

	n=2	n=3	n=4	n=5	n=6	n=7	n=8	n=9	n=10	**all**
2y	3.30	3.46	3.56	3.68	3.92	3.85	4.00	3.96	4.00	**3.75**
3y	2.76	3.18	3.30	3.46	3.84	3.73	3.89	3.88	3.91	**3.54**
4y	2.39	2.94	3.11	3.28	3.42	3.64	3.69	3.56	3.67	**3.28**
5y	2.50	2.89	2.96	3.14	3.18	3.36	3.57	3.59	3.57	**3.18**
all	**2.699**	**3.112**	**3.245**	**3.378**	**3.667**	**3.657**	**3.8**	**3.754**	**3.771**	**3.45**

(c) fCAW

	n=2	n=3	n=4	n=5	n=6	n=7	n=8	n=9	n=10	**all**
2y	2.30	2.26	2.19	2.23	2.14	2.11	2.04	2.11	2.12	**2.16**
3y	2.55	2.26	2.05	2.17	1.97	2.05	1.98	1.96	2.14	**2.11**
4y	2.81	2.42	2.29	2.26	2.21	2.10	2.14	2.10	2.22	**2.28**
5y	2.66	2.45	2.39	2.21	2.29	2.41	2.36	2.55	2.50	**2.42**
all	**2.587**	**2.334**	**2.165**	**2.219**	**2.159**	**2.114**	**2.055**	**2.156**	**2.157**	**2.22**

(d) rCAW

Table 2.1: Average ranking of individual models in terms of out-of-sample Stein loss (1 indicates the best model, 4 indicates the worst model) for combinations of a number of assets n and length of the estimation window T.

The factorial design of our experiments also allows us to address questions regarding the optimal length of the estimation window and frequency with which the model needs to be re-estimated. This is especially relevant because researchers, possibly due to computational limitations associated with high dimensional optimization, frequently opt for a fixed forecasting scheme (see eg. Lucheroni et al., 2019; Asai et al., 2020), in which case a single estimated model is used throughout the whole out-of-sample period.

With respect to the former question (the optimal length of the estimation window), the upper panel of Figure 2.4 depicts the out-of-sample Stein losses for different estimation window lengths relative to the loss, which would be achieved if the estimation window

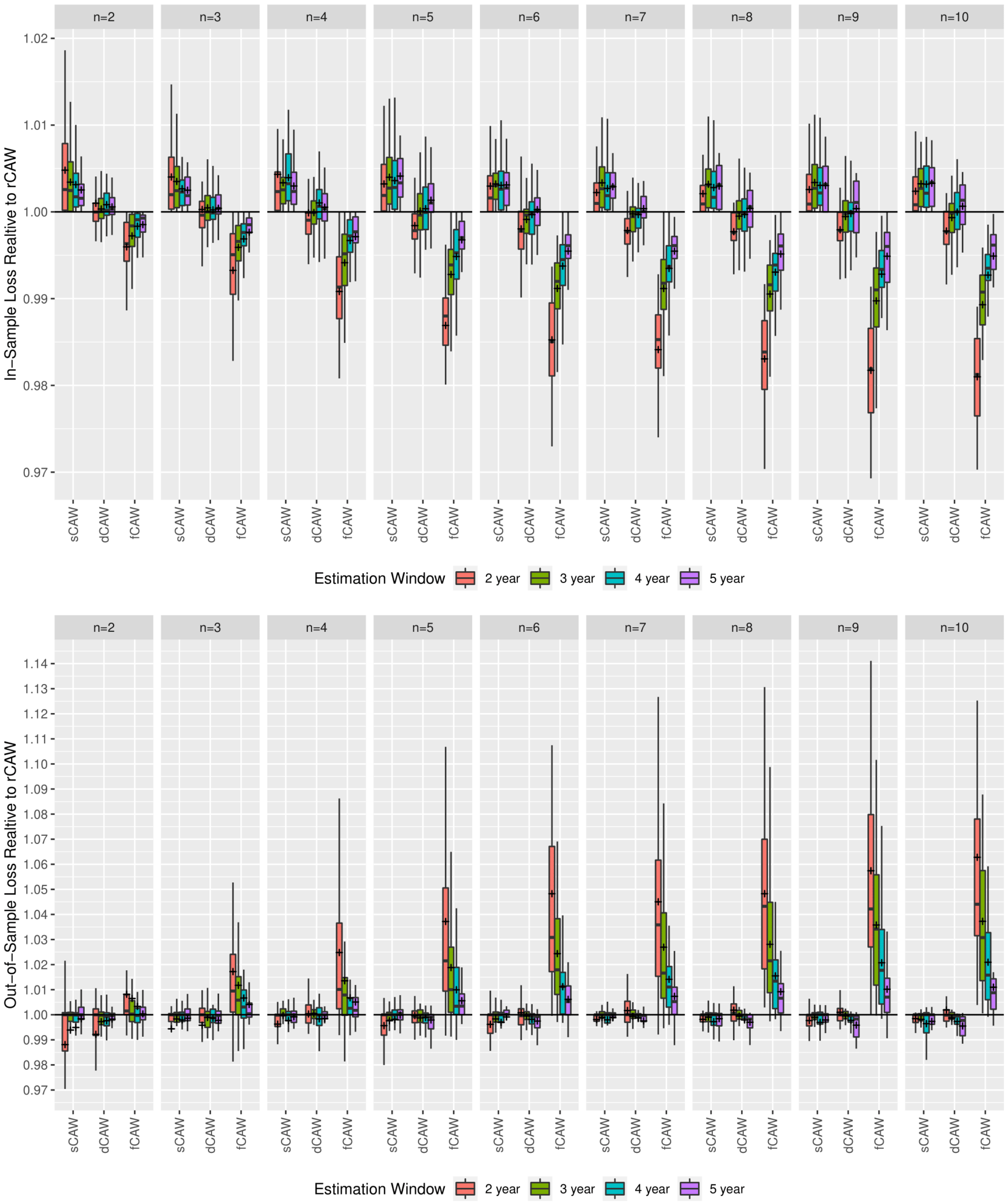
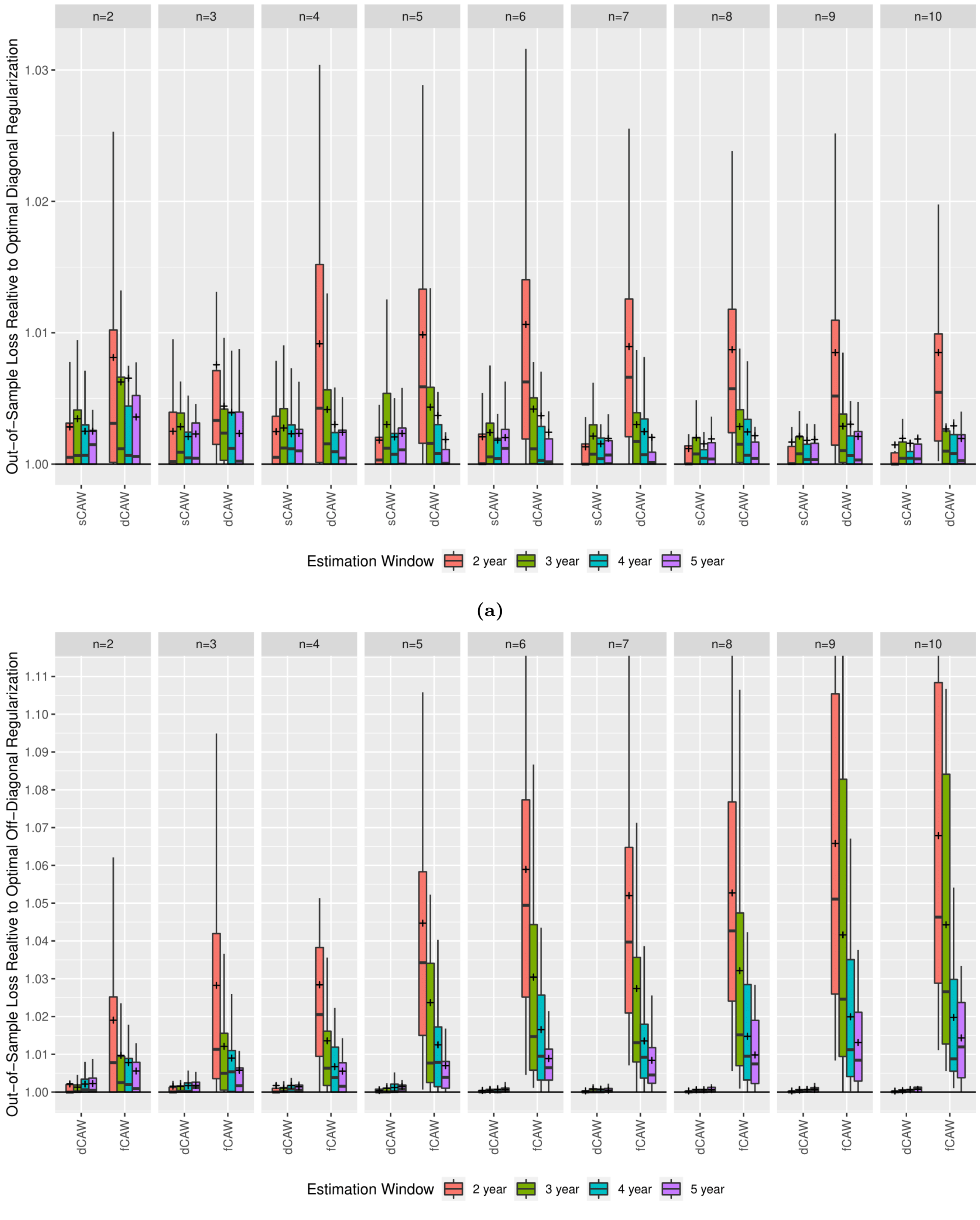


Figure 2.2: Ratios of in-sample (the upper panel) and out-of-sample (the lower panel) Stein losses of individual models (rCAW is the benchmark represented by the horizontal line) plotted for different combinations of a number of assets n and length of the estimation window T.

length was equal to a benchmark of 2 years for the given out-of-sample part of the data and the combination of assets. In line with expectations, for the fCAW, the ratios of losses are generally below 1 implying that increasing the estimation window length leads to a more precise forecast. However, for the dCAW and especially for the rCAW and



(a)

(b)

Figure 2.3: Ratios of Stein losses of individual models and the loss which would be achieved under the ex-post optimal regularization parameter λ_d (a) and λ_f (b) (benchmark, represented by the horizontal line) plotted for different combinations of a number of assets n and length of the estimation window T.

sCAW, increasing the estimation window length does not lead to a better forecasting performance; in fact, it somewhat worsens it (in the case of the sCAW, on average by

	n=2	n=3	n=4	n=5	n=6	n=7	n=8	n=9	n=10	**all**
2y	0.09	0.16	0.12	0.14	0.18	0.17	0.19	0.14	0.15	**0.15**
3y	0.12	0.18	0.16	0.23	0.18	0.25	0.25	0.27	0.27	**0.21**
4y	0.19	0.14	0.14	0.17	0.17	0.19	0.25	0.28	0.11	**0.19**
5y	0.14	0.11	0.14	0.18	0.39	0.25	0.21	0.27	0.29	**0.22**
all	**0.13**	**0.159**	**0.151**	**0.187**	**0.226**	**0.221**	**0.232**	**0.234**	**0.214**	0.20

(a) sCAW

	n=2	n=3	n=4	n=5	n=6	n=7	n=8	n=9	n=10	**all**
2y	0.23	0.16	0.36	0.26	0.29	0.37	0.39	0.28	0.31	**0.29**
3y	0.19	0.11	0.21	0.23	0.11	0.14	0.16	0.18	0.18	**0.17**
4y	0.31	0.14	0.17	0.17	0.14	0.19	0.19	0.22	0.22	**0.19**
5y	0.18	0.04	0.11	0.00	0.11	0.04	0.07	0.00	0.00	**0.06**
all	**0.249**	**0.137**	**0.212**	**0.183**	**0.14**	**0.186**	**0.2**	**0.191**	**0.186**	0.19

(b) dCAW

	n=2	n=3	n=4	n=5	n=6	n=7	n=8	n=9	n=10	**all**
2y	0.47	0.52	0.64	0.76	0.88	0.89	0.89	0.92	0.92	**0.77**
3y	0.24	0.41	0.52	0.68	0.73	0.73	0.77	0.85	0.91	**0.64**
4y	0.25	0.31	0.33	0.50	0.67	0.67	0.83	0.78	0.89	**0.57**
5y	0.11	0.25	0.43	0.43	0.57	0.46	0.64	0.63	0.57	**0.46**
all	**0.268**	**0.39**	**0.464**	**0.59**	**0.703**	**0.704**	**0.771**	**0.809**	**0.829**	0.61

(c) fCAW

	n=2	n=3	n=4	n=5	n=6	n=7	n=8	n=9	n=10	**all**
2y	0.34	0.28	0.36	0.32	0.35	0.29	0.33	0.27	0.23	**0.31**
3y	0.21	0.21	0.11	0.21	0.21	0.16	0.18	0.21	0.36	**0.19**
4y	0.28	0.22	0.14	0.19	0.22	0.25	0.28	0.37	0.44	**0.26**
5y	0.14	0.14	0.14	0.29	0.36	0.25	0.36	0.42	0.43	**0.28**
all	**0.249**	**0.22**	**0.187**	**0.216**	**0.262**	**0.225**	**0.25**	**0.282**	**0.314**	0.24

(d) rCAW

Table 2.2: Frequency of rejection of individual models via the MCS for the Stein loss at the 5% level for combinations of a number of assets n and length of the estimation window T.

0.27% per additional year of the estimation window). This counter-intuitive behavior may be attributed to changes of the data generating process, which penalize utilization of more distant data points during estimation.

With respect to the latter question (the importance of re-estimation), the lower panel of Figure 2.4 depicts the average Stein loss for individual out-of-sample observations with different distance from the end of the estimation window relative to the average Stein loss over the whole out-of-sample period.[7] Clearly, one-day-ahead forecasts made for the periods immediately following the estimation window are substantially better compared to day-ahead forecasts made for more distant periods. This effect is sizable averaging to 22% per year, completely dwarfing any differences that are observed among the sCAW, dCAW and rCAW. The effect is stronger for more parameterized models such as the fCAW and for larger numbers of assets n. Again, this indicates that the data generating process may be changing, and/or that CAW-type models are likely to merely locally approximate the process rather than to correctly describe it globally.

2.3.3 In-Sample Performance

In the previous subsection, we compare the models in terms of their out-of-sample predictive performance as it is admittedly the single most relevant factor for practitioners. The ranking of the models in terms of their in-sample performance, on the other hand, is evident given their nested nature. Nonetheless, the assessment of in-sample performance still serves a valuable role. While none of the models, not even the fCAW, can be reasonably

[7] The outlier period 2007-02-27 was removed from this analysis, as its relative value exceeds the sample standard deviation by approximately ten times, heavily skewing the results.

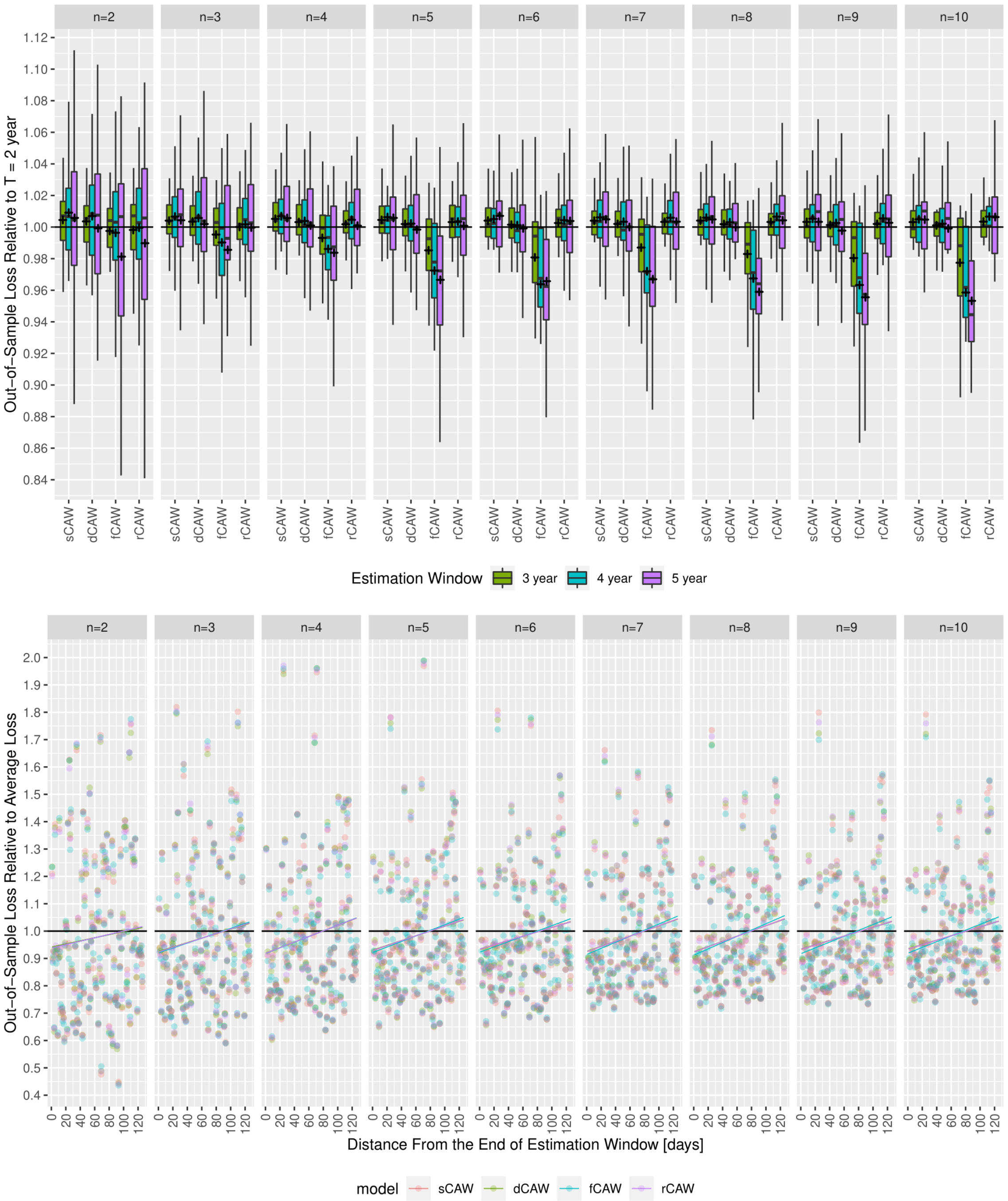


Figure 2.4: The upper panel displays ratios of out-of-sample Stein loss for lengths of estimation window {3, 4, 5} years and the loss for the estimation window of only 2 years (benchmark represented by the horizontal line) plotted for different combinations of a number of assets n and individual models. The lower panel displays ratios of the average Stein loss for individual out-of-sample observations with different distance from the end of the estimation window and the average loss over the whole out-of-sample period (benchmark represented by the horizontal line) plotted for different combinations of a number of assets n and individual models.

expected to describe the data generating process perfectly in all its complexity, comparing the in-sample performance of individual models offers valuable insights regarding the poor forecasting performance of fCAW and the incapability of rCAW to deliver a superior performance relative to sCAW and dCAW. More flexible models such as the fCAW should explain the data generating process notably better than the sCAW to justify hundreds of its additional parameters. Similarly, allowing for non-zero, albeit shrinked, off-diagonal parameters in the case of rCAW is beneficial only to the extent to which these parameters can actually genuinely help to explain the real volatility dynamics.

To assess the in-sample performance, we employ the extended Bartlett decomposition method for Wishart processes as proposed in Alfelt et al. (2020). In particular, we compute

$$U_t = \left(\underbrace{S_t^{-\frac{1}{2}} R_t \left(S_t^{-\frac{1}{2}} \right)^\top}_{=Q_t} \right)^{\frac{1}{2}}, \tag{2.6}$$

where Q_t is the re-scaled matrix of errors and U_t is its lower-triangular Cholesky root. We then compute transformed errors

$$e_{t,i,j} = \begin{cases} U_{t,i,j} \text{ for } j < i \\ \Phi^{-1} \left(F_{\Gamma(\frac{v-i+1}{2},2)} \left((U_{t,i,j})^2 \right) \right) \text{ for } j = i \end{cases} \tag{2.7}$$

and collect them to a single vector

$$e_t = \left(e_{t,1,1}, ..., e_{t,n,1}, e_{t,2,2}, ..., e_{t,n,n} \right)^\top . \tag{2.8}$$

As shown in Alfelt et al. (2020), if the model is correctly specified, the errors are IID standard normal, i.e., $e_t \sim \mathcal{N}(\mathbf{0}_k, I_k)$ with $k = n(n+1)/2$. This convenient decomposition allows us to separately assess how well CAW models account for different features of the data generating process; an auto-correlation of e_t signals misspecification of the law of motion for S_t, whereas a violation of $e_t \sim \mathcal{N}(\mathbf{0}_k, I_k)$ signals deviations from the assumed Wishart distribution and/or systematic prediction errors.

Table 2.3 displays average rejection rates across different window locations for selected tests performed on e_t. In this exercise, the estimation window is set to $T = 5$ years to obtain the largest possible power and the number of stocks is set to $n = 10$ to maximize the potential difference between the sCAW, dCAW and fCAW. Furthermore, we omit results for the rCAW as the regularization is not justified when only the in-sample fit is considered. For sCAW, dCAW and fCAW alike, all test (with the exception of t-test for $H_0 : \mathbb{E}[e_t] = \mathbf{0}_k$) reject the hypothesis that the data are consistent with the CAW$(1,1)$ model; the e_t exhibits an excess variance, non-normality, and temporal correlations. Considering the long estimation window and the power of these tests, these findings are not surprising. CAW model rarely perfectly explain the observed data; in-sample errors are often auto-correlated (Golosnoy et al., 2012; Alfelt et al., 2020), and inconsistent with the assumption of Wishart distribution (Alfelt et al., 2020), with only the test of $H_0 : \mathbb{E}[e_t] = \mathbf{0}_k$ usually exhibiting rejection rates close to 0 (see Alfelt et al., 2020). Moreover, similar results are likely to be observed even when considering higher order CAW models; in Golosnoy et al. (2012), increasing the model order from CAW$(1,1)$ to CAW$(3,3)$ leads to a non-rejection of only one additional error auto-correlation (out of 15).

$H_0 :$	$\mathbb{E}[e_t] = \mathbf{0}_k$	$Var(e_t) = I_k$	$e_t \sim \mathcal{N}(\mu, \Sigma)$	$e_t \perp\!\!\!\perp e_{t'}$
sCAW	0.571	1.000	1.000	1.000
dCAW	0.714	1.000	1.000	1.000
fCAW	0.000	1.000	1.000	1.000

Table 2.3: Average rejection rates at $p = 0.01$ across different window locations. For each window location, null hypotheses were tested via the t-test, chi-squared test, Anderson and Darling (1952) normality test, and multivariate Box and Pierce (1970) auto-correlation test (with $\lfloor log(5 * 252) \rfloor = 7$ lags), respectively.

To better assess the in-sample fit of different models, we also perform tests on individual elements $e_{t,i,j}$ rather than on the whole vector e_t, see Table 2.4. Despite its 198 additional parameter relative to the sCAW, the fCAW does not deliver a markedly better in-sample fit. It exhibits comparable rejection rates to the sCAW for all considered tests with the exception of the univariate Box and Pierce (1970) test, where it offers a modest 8.6% reduction of the rejection rate. This shows that non-zero off-diagonal elements of parameter matrices A and B generally fail to explain the residual error dependencies. Interestingly however, errors $e_{t,i,j}$ from the fCAW exhibit a smaller kurtosis relative those of the sCAW (35.8 vs 42.2), as can be seen in Figure 2.7 in Appendix. This is indicative of the fact that off-diagonal elements of parameter matrices A and B are used to better accommodate outliers in the estimation window rather than to genuinely model the law of motion for volatility. This also likely explains the poor performance of the fCAW, and by extension, also the incapability of the optimal regularized rCAW to outperform the sCAW and dCAW.

$H_0:$	$\mathbb{E}[e_{t,i,j}] = 0$	$Var(e_{t,i,j}) = 1$	$e_{t,i,j} \sim \mathcal{N}(\mu, \sigma^2)$	$e_{t,i,j} \perp\!\!\!\perp e_{t',i,j}$	$Cov(e_{t,i,j}, e_{t,i',j'}) = 0$
sCAW	0.125	0.987	0.699	0.847	0.176
dCAW	0.109	0.987	0.714	0.875	0.177
fCAW	0.117	0.987	0.688	0.761	0.168

Table 2.4: Average rejection rates at $p = 0.01$ across different window locations and i, j. For each window location and combination of i, j, null hypotheses were tested via the t-test, chi-squared test, Anderson and Darling (1952) normality test, univariate Box and Pierce (1970) auto-correlation test (with $\lfloor log(5 * 252) \rfloor = 7$ lags), and t-test for correlation, respectively.

2.4 Conclusions

We perform an extensive forecasting experiment examining the performance of the sCAW, dCAW, fCAW, and the regularized version thereof, rCAW, which nests all three via ridge-type regularization towards zero and towards homogeneity. The results confirm the poor predictive performance of the fCAW relative to more restricted models. The performance of the sCAW and dCAW is comparable, slightly favoring the dCAW. The optimal amount of regularization in the rCAW does not seem to bring any tangible improvements in terms of forecasting performance, irrespective of how precise is tuning of regularization intensity. This indicates that the cross-sectional volatility dependence is not a major factor, at least as far as the forecasting performance is concerned.

Further analysis shows that for the sCAW and dCAW, increasing the length of the estimation window typically does not lead to a better forecasting performance; oftentimes, the converse is true. Furthermore, we observe a very quick performance deterioration when one-day-ahead forecast are made using a model estimated on more distant segments of data, indicating possible model misspecification and/or changes in the data generating process. Overall, based on the results, we would recommend to perform multivariate volatility forecasting via diagonal variants of volatility models estimated on a short rolling window, to achieve the best forecasting performance.

While we have performed all experiments with the canonical CAW(1,1) model, we conjecture that the tendencies we have discovered also carry over to other models with a similar structure of the dynamic volatility equation – more general CAW and BEKK models, various extensions thereof (Caporin and Paruolo, 2015; Caporin and McAleer, 2014; Anatolyev and Kobotaev, 2018), and extensions with a block structure, these tendencies being applicable for blocks of assets.

There are several possible avenues for future research. A natural extension would be to validate these results using a wider universe of assets and more recent data. While our dataset from 2001-02 to 2009-12 covers the global financial crisis of 2007-2009, it would be valuable to examine whether the subpar performance of more complex models extends to the COVID-19 market crash in 2020 and the turbulent period in 2022. Another interesting direction would be to investigate the impact of regularization on the performance of BEKK-X type models with exogenous variables $x_t \in \mathbb{R}^k$, which are commonly appended to the modeling equation in the form $Dx_{t-1}x_{t-1}^\top D^\top$ (see Engle and Kroner, 1995; Thieu, 2016). These models also suffer from overparametrization, making them potential candidates for regularization. We can distinguish two cases in which this regularization could be implemented. In the first case, where the vector x_t contains the same type of univariate variable measured for all assets (i.e., $k = n$), regularization can be performed similarly to the method demonstrated in this chapter. Here, regularization of the diagonal elements of parameter matrix D would induce homogeneity, while regularization of off-diagonal elements would regulate cross-dependence. In the more common situation where x_t contains marketwise variables common to all assets, such as inflation or unemployment, a different approach is needed. In these cases, it would be sensible to perform regularization of individual columns of D towards homogeneity to unify the effects of these exogenous variables on individual assets.

The regularization scheme developed in this study, which shrinks diagonal elements towards homogeneity and off-diagonal elements towards zero, has potential applications beyond volatility forecasting. One promising area is the estimation of sample covariance matrices in Feasible Generalized Least Squares (FGLS). FGLS, while possessing desirable asymptotic properties, often perform worse than Ordinary Least Squares (OLS) with robust errors in small samples, primarily due to the additional degrees of freedom required for covariance structure estimation (see, e.g., Angrist and Pischke, 2009, Section 3.4.1.). By nesting OLS as a special case of such a regularized estimator and inferring the optimal degree of regularization via leave-one-out cross-validation, one might safeguard FGLS against poor performance relative to OLS, similarly as in DiCiccio et al. (2019), González-Coya and Perron (2024), and Chaudhuri and Renault (2023).

2.A Supplementary Results

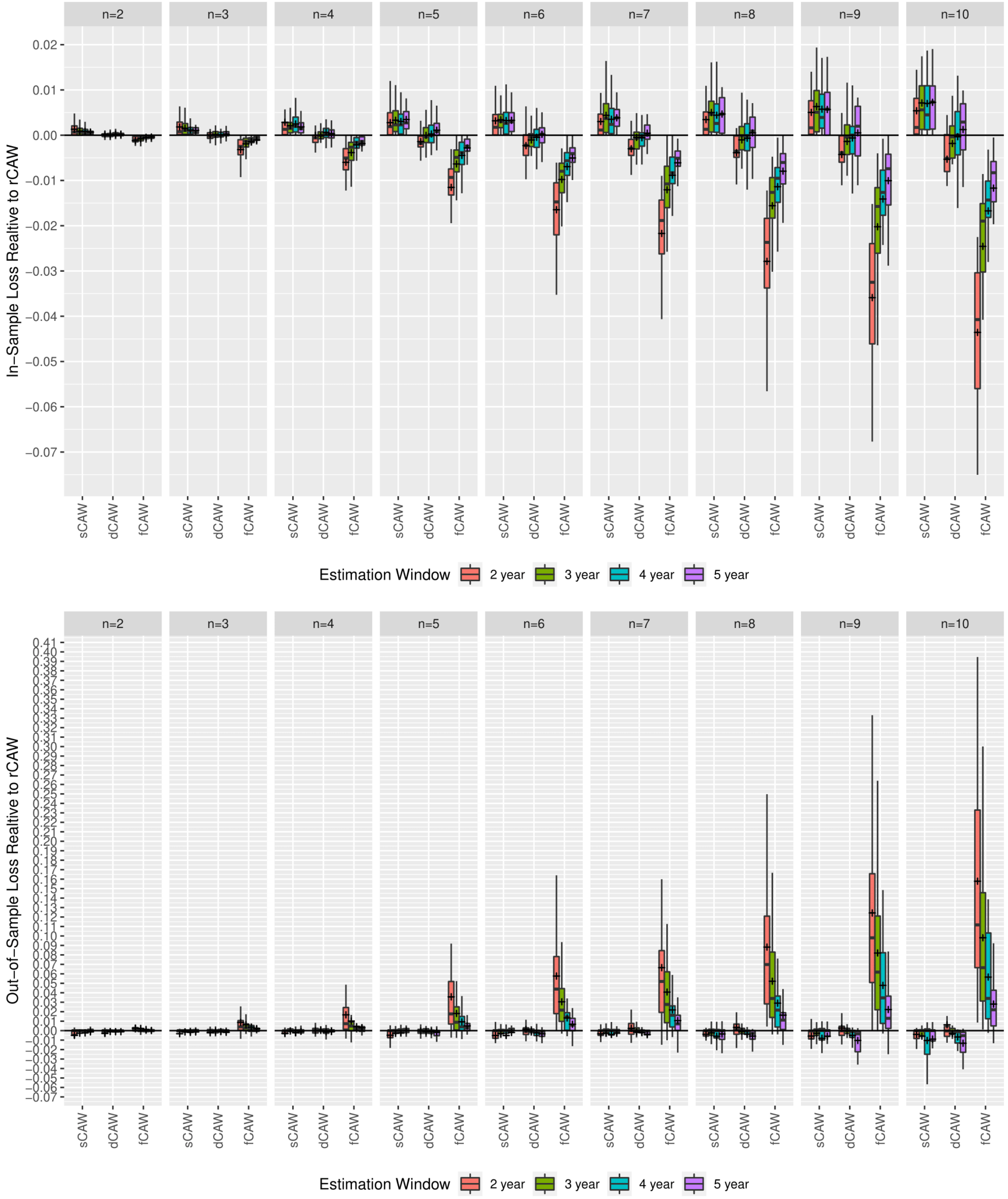


Figure 2.5: Differences of the in-sample (the upper panel) and out-of-sample (the lower panel) Stein losses of individual models and rCAW (benchmark represented by the horizontal line) plotted for different combinations of a number of assets n and length of the estimation window T.

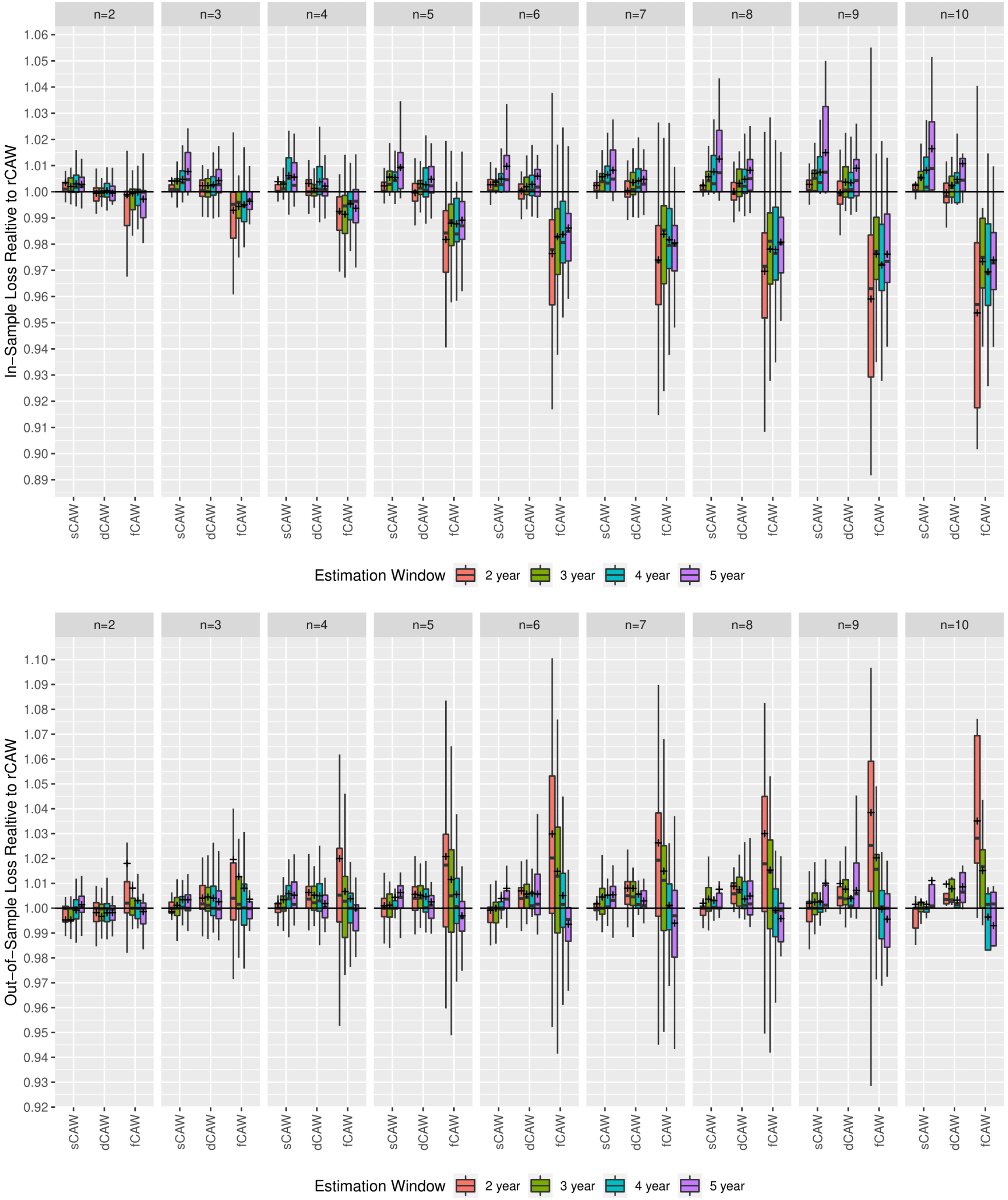


Figure 2.6: Ratios of in-sample (the upper panel) and out-of-sample (the lower panel) Frobenius losses of individual models and rCAW (benchmark represented by the horizontal line) plotted for different combinations of a number of assets n and length of the estimation window T.

	n=2	n=3	n=4	n=5	n=6	n=7	n=8	n=9	n=10	**all**
2y	2.11	2.24	2.23	2.17	1.96	1.97	2.00	1.94	1.89	**2.06**
3y	2.21	2.38	2.46	2.26	2.10	2.18	2.11	1.99	2.14	**2.20**
4y	2.61	2.61	2.88	2.60	2.40	2.54	2.47	2.14	1.78	**2.49**
5y	2.88	2.77	3.07	3.18	3.18	3.13	3.00	2.94	3.07	**3.01**
all	**2.42**	**2.471**	**2.597**	**2.489**	**2.346**	**2.389**	**2.345**	**2.192**	**2.157**	**2.39**

(a) sCAW

	n=2	n=3	n=4	n=5	n=6	n=7	n=8	n=9	n=10	**all**
2y	2.47	2.72	2.84	2.78	2.88	2.87	2.96	2.92	3.00	**2.82**
3y	2.36	2.50	2.80	2.89	2.80	2.96	3.05	3.08	3.09	**2.83**
4y	2.44	2.53	2.86	2.56	2.81	2.94	2.94	3.11	3.22	**2.81**
5y	2.39	2.32	2.54	2.57	2.57	2.71	2.50	2.68	2.71	**2.55**
all	**2.431**	**2.588**	**2.781**	**2.719**	**2.814**	**2.836**	**2.861**	**2.974**	**3.043**	**2.77**

(b) dCAW

	n=2	n=3	n=4	n=5	n=6	n=7	n=8	n=9	n=10	**all**
2y	2.98	2.84	2.76	2.84	3.04	3.17	3.12	3.36	3.39	**3.05**
3y	2.81	2.93	2.48	2.77	3.02	2.93	2.91	3.06	3.00	**2.88**
4y	2.36	2.47	2.08	2.53	2.67	2.50	2.44	2.48	2.89	**2.46**
5y	2.21	2.36	2.11	2.00	1.82	2.07	2.21	2.34	2.29	**2.16**
all	**2.628**	**2.643**	**2.41**	**2.594**	**2.692**	**2.736**	**2.761**	**2.831**	**2.914**	**2.68**

(c) fCAW

	n=2	n=3	n=4	n=5	n=6	n=7	n=8	n=9	n=10	**all**
2y	2.45	2.20	2.17	2.21	2.12	1.99	1.92	1.78	1.73	**2.07**
3y	2.62	2.19	2.27	2.08	2.08	1.93	1.93	1.88	1.77	**2.10**
4y	2.58	2.39	2.18	2.32	2.13	2.01	2.14	2.27	2.11	**2.25**
5y	2.52	2.55	2.29	2.25	2.43	2.09	2.29	2.04	1.93	**2.28**
all	**2.52**	**2.298**	**2.212**	**2.198**	**2.149**	**2.039**	**2.034**	**2.004**	**1.886**	**2.16**

(d) rCAW

Table 2.5: Average ranking of individual models in terms of out-of-sample Frobenius loss (1 indicating the best model and 4 indicating the worst model) for combinations of number of assets n and length of the estimation window T.

	n=2	n=3	n=4	n=5	n=6	n=7	n=8	n=9	n=10	**all**
2y	0.11	0.12	0.14	0.14	0.02	0.14	0.15	0.12	0.08	**0.12**
3y	0.02	0.21	0.14	0.14	0.09	0.07	0.14	0.06	0.09	**0.10**
4y	0.11	0.17	0.14	0.06	0.03	0.06	0.03	0.04	0.00	**0.07**
5y	0.07	0.07	0.11	0.18	0.25	0.21	0.18	0.22	0.29	**0.17**
all	**0.071**	**0.137**	**0.151**	**0.122**	**0.1**	**0.114**	**0.132**	**0.098**	**0.1**	**0.11**

(a) sCAW

	n=2	n=3	n=4	n=5	n=6	n=7	n=8	n=9	n=10	**all**
2y	0.21	0.30	0.26	0.18	0.22	0.37	0.33	0.33	0.46	**0.28**
3y	0.07	0.21	0.14	0.16	0.16	0.14	0.16	0.20	0.09	**0.15**
4y	0.17	0.11	0.08	0.06	0.03	0.06	0.08	0.07	0.11	**0.08**
5y	0.11	0.04	0.07	0.11	0.14	0.18	0.04	0.10	0.14	**0.10**
all	**0.141**	**0.188**	**0.155**	**0.126**	**0.147**	**0.193**	**0.175**	**0.189**	**0.2**	**0.17**

(b) dCAW

	n=2	n=3	n=4	n=5	n=6	n=7	n=8	n=9	n=10	**all**
2y	0.23	0.24	0.28	0.26	0.37	0.31	0.29	0.36	0.39	**0.30**
3y	0.17	0.23	0.11	0.21	0.23	0.30	0.23	0.30	0.18	**0.23**
4y	0.17	0.17	0.06	0.08	0.17	0.06	0.08	0.09	0.11	**0.11**
5y	0.11	0.14	0.07	0.04	0.07	0.11	0.11	0.15	0.00	**0.10**
all	**0.16**	**0.199**	**0.133**	**0.162**	**0.219**	**0.189**	**0.193**	**0.229**	**0.229**	**0.19**

(c) fCAW

	n=2	n=3	n=4	n=5	n=6	n=7	n=8	n=9	n=10	**all**
2y	0.17	0.14	0.18	0.20	0.20	0.19	0.19	0.18	0.23	**0.18**
3y	0.10	0.16	0.14	0.11	0.09	0.07	0.07	0.08	0.00	**0.10**
4y	0.14	0.11	0.06	0.00	0.00	0.00	0.00	0.00	0.00	**0.04**
5y	0.14	0.14	0.11	0.11	0.11	0.14	0.07	0.15	0.14	**0.12**
all	**0.141**	**0.152**	**0.126**	**0.112**	**0.097**	**0.107**	**0.096**	**0.1**	**0.1**	**0.12**

(d) rCAW

Table 2.6: Frequency of rejection of individual models via the MCS for the Frobenius loss at the 5% level for combinations of a number of assets n and length of the estimation window T.

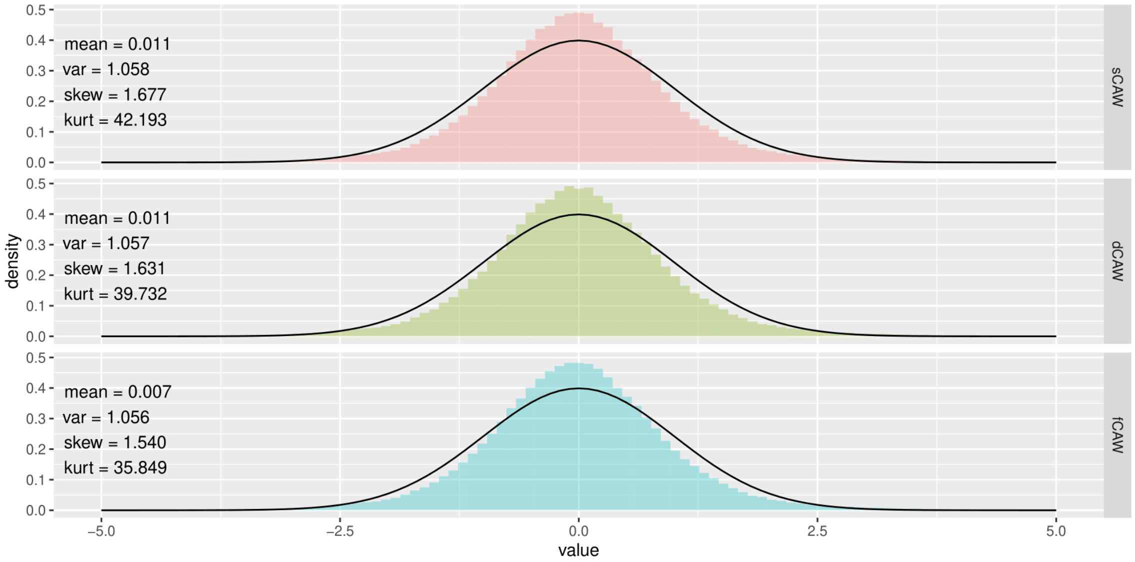


Figure 2.7: Distributions of $e_{t,i,j}$ for individual models plotted against $\mathcal{N}(0,1)$ distribution function.

Chapter 3

Designing Time-Series Models With Hypernetworks

An early version of this chapter appeared in a pre-print Staněk (2023a).

3.1 Introduction

According to the classification by Januschowski et al. (2020), time-series forecasting approaches can be broadly divided into two strains: The conventional approach, known as *local modeling*, involves selecting the most appropriate parametric model for a given family of forecasting tasks, often based on expert judgment. This model is then applied to each individual observed series independently. On the other hand, *global models* consider all observed time-series jointly. In extreme cases, this can be done via pooling, thus disregarding the information regarding which data belong to which series altogether and estimating a single global model (see e.g. Montero-Manso and Hyndman, 2021). However, it is often beneficial to utilize this information to help account for possible heterogeneity among the data generating processes (DGPs henceforth) underlying the series, an approach aptly dubbed the *localization of global models* (Godahewa et al., 2021). This is typically performed by grouping the series either with time-series clustering techniques based on time-series features (Bandara et al., 2020) or directly according to model performance

(Smyl and Kuber, 2016; Smyl, 2020) and estimating a specialized global model on each cluster. By adjusting the number of such clusters, one can then regulate the degree of globality/locality.

We present an alternative method that helps bridge the gap between these two extremes. A global model, which instead of deriving a single forecasting function for all time-series, outputs a function parameterized by a latent parameter vector specific to each series, thereby acknowledging the potential heterogeneity of DGPs. Accounting for heterogeneity through the latent parameter space, rather than clustering, offers the advantage of equally accommodating a mixture of several different types of DGPs as well as a family of continuously varying DGPs. Alternatively, this approach can also be viewed as a data-driven alternative to manually designing a parametric model for a group of related prediction tasks, an endeavor which typically requires considerable statistical expertise and domain knowledge.

Specifically, by connecting an encoder-decoder network that accepts a task identifier to the parameters of another network responsible for processing inputs and generating predictions for that task, we enable a simultaneous search across the space of parametric functions and their associated parameter values. Importantly, the resulting hyper-network allows for complete backpropagation and does not rely on the computation of higher-order derivatives for training, unlike alternative approaches (see e.g., Finn et al., 2017; Li et al., 2017). This allows, even with relatively limited computational resources, the design a parametric model that is finely tuned for a specific family of tasks, using the allotted degrees of freedom per task to capture the variability between tasks.

Abstracting from the time-series nature of the data, the method belongs to a broader category of meta-learning and/or multi-task learning methods, depending on how exactly it is deployed in practice. Meta-learning aims at designing/training a model based on multiple observed tasks so that it performs well when adapted with training data of a yet unseen task from the same family, and subsequently evaluated on the test data of that task. In contrast, multi-task learning aims to achieve optimal performance on new data from tasks that were used for the initial training. For an excellent review of these two closely related fields, please refer to Hospedales et al. (2021), Huisman et al. (2021), or Zhang and Yang (2022), respectively.

We demonstrate the performance of the model in two applications. First, we show that the proposed model substantially outperforms MAML (Finn et al., 2017) and other state-of-the-art meta-learning approaches on the sinusoidal regression task, a synthetic problem introduced in Finn et al. (2017) commonly used to benchmark various meta-learning approaches. In the second application, we apply the model to the time-series from the M4 forecasting competition, following the evaluation framework of Montero-Manso and Hyndman (2021). A simple linear model localized via MtMs outperforms both the corresponding global model applied on pooled data and models localized via clustering for the majority of series. In the third application, we apply the model to the forecasting challenge of the M6 Financial Forecasting Competition (see Makridakis et al., 2022). There, the model secured 4th place in the forecasting challenge, which, combined with the results of the investment challenge, ultimately resulted in the 1st place in the overall duathlon ranking.

The remainder of the chapter is structured as follows. Section 3.2 lays out the statistical framework and motivates the proposed model. Section 3.3 introduces the model. Section 3.4 demonstrates its superior performance on the sinusoidal regression task. Section 3.5 demonstrates the performance of the model on time-series from the M4 forecasting competition. Section 3.6 details its application to the M6 Financial Forecasting Competition. Section 3.7 concludes. Section 3.8 and Appendix 3.A contain proofs and supplementary materials, respectively.

3.2 Statistical Framework

In this section, we formulate the problem in terms of the meta-learning objective, as it is typically a more relevant paradigm for time-series forecasting. Following the notation of Hospedales et al. (2021), we denote a task as $\mathcal{T} = \{\mathcal{D}_{train}, \mathcal{D}_{val}\}$.[1] This task consists of data generated by some DGP split into a training set $\mathcal{D}_{train} = \{(x_t, y_t)\}_{t=1}^{K}$ used for estimating model parameters, and a validation set $\mathcal{D}_{val} = \{(x_t, y_t)\}_{t=K+1}^{N}$ for which we aim to make predictions. The vector $x_t \in \mathbb{R}^{d_x}$ typically contains lagged values of $y_t \in \mathbb{R}^{d_y}$ or some transformation of these values. Tasks are distributed according to an unknown distribution $p(\mathcal{T})$.

[1] Hospedales et al. (2021) allow for a slightly more general setup in which the loss function may also differ across tasks. However, this level of generality is not necessary for our purposes, so we suppress it for ease of exposition.

In the framework a model consists of two components: the prediction function

$$\hat{y}_t = f_\omega(x_t; \hat{\theta}) \tag{3.1}$$

which outputs predictions of y_t based on the predictors x_t, and the estimation function

$$\hat{\theta} = \kappa_\omega(\mathcal{D}_{train}) \tag{3.2}$$

which outputs the vector of task-specific parameters $\hat{\theta} \in \Theta$ given the observations $\mathcal{D}_{train}$. Both functions, $f_\omega(\cdot)$ and $\kappa_\omega(\cdot)$, are further parameterized by a vector of meta parameters $\omega \in \Omega$, which are not directly dependent on the task $\mathcal{T}$ and generally encompass any prior decisions regarding the model (e.g., the choice of an appropriate model and its particular specification, estimation procedures, regularization techniques applied when estimating $\hat{\theta}$ etc.). To clearly differentiate between the meta parameters ω and the task-specific parameters θ, we will refer to the latter as *mesa* parameters, following Hubinger et al. (2021).

The quality of the model is assessed by the loss incurred on the evaluation set, denoted by $\mathcal{L}(\mathcal{D}_{val}; \hat{\theta}, \omega)$, with

$$\mathcal{L}(\mathcal{D}; \hat{\theta}, \omega) = \frac{1}{|\mathcal{D}|} \sum_{(x_t, y_t) \in \mathcal{D}} \gamma\left(y_t, f_\omega(x_t; \hat{\theta})\right) \tag{3.3}$$

where the function γ measures the discrepancy between y_t and the prediction $\hat{y}_t$. Typically, to align the process of finding the optimal parameters θ, the estimation $\hat{\theta} = \kappa_\omega(\mathcal{D}_{train})$ is likewise performed by numerically minimizing the incurred loss over the training set:

$$\hat{\theta} = \kappa_\omega(\mathcal{D}_{train}) \approx \underset{\theta \in \Theta}{\arg\min}\, \mathcal{L}(\mathcal{D}_{train}; \theta, \omega). \tag{3.4}$$

Oftentimes, the information contained in ω regarding which forecasting function $f_\omega(\cdot)$ to use and the most appropriate estimation function $\kappa_\omega(\cdot)$ is determined through expert judgment, based on informal prior knowledge regarding the task and/or ad-hoc hyperparameter tuning. By considering a family of tasks distributed according to $p(\mathcal{T})$, we can formalize the problem of finding the most suitable model; ω such that, when observing $\mathcal{D}_{train}$ and adapting accordingly through $\hat{\theta}$, the expected performance on yet unobserved $\mathcal{D}_{val}$ will be minimized. Formally:

$$\begin{aligned} \omega^* &= \operatorname*{arg\,min}_{\omega\in\Omega} \mathop{\mathbb{E}}_{\mathcal{T}\sim p(\mathcal{T})}[\mathcal{L}(\mathcal{D}_{val};\hat{\theta},\omega)] \\ \text{s.t.: } \hat{\theta} &= \kappa_\omega(\mathcal{D}_{train}) \approx \operatorname*{arg\,min}_{\theta\in\Theta} \mathcal{L}(\mathcal{D}_{train};\theta,\omega). \end{aligned} \tag{3.5}$$

Solving this problem is not feasible as the distribution $p(\mathcal{D})$ is unknown. However, given a collection of M observed tasks $\{\mathcal{T}^{(m)}\}_{m=1}^M$, it is, at least in theory, possible to solve the finite sample equivalent of the problem instead:

$$\begin{aligned} \hat{\omega} &= \operatorname*{arg\,min}_{\omega\in\Omega} \frac{1}{M}\sum_{m=1}^{M} \mathcal{L}(\mathcal{D}_{val}^{(m)};\hat{\theta}^{(m)},\omega) \\ \text{s.t.: } \hat{\theta}^{(m)} &= \kappa_\omega(\mathcal{D}_{train}^{(m)}) \approx \operatorname*{arg\,min}_{\theta\in\Theta} \mathcal{L}(\mathcal{D}_{train}^{(m)};\theta,\omega). \end{aligned} \tag{3.6}$$

3.3 Model

The bi-level optimization problem presented in Eq. 3.6 is generally computationally demanding. It may be feasible to estimate $\{\theta^{(m)}\}_{m=1}^M$ for a limited set of different model specifications $\Omega = \{\omega_i\}_{i=1}^{d_\Omega}$, and choose the model $f_{\omega_i}(\cdot)$ that yields the best out-of-sample performance over $\{\mathcal{D}_{val}^{(m)}\}_{m=1}^M$. However, this approach quickly becomes untenable when the set Ω is large or even uncountable, for example, when considering a continuum of possible models rather than a limited set of predefined model specifications.

In addressing this problem, we adopt the following two simplifying assumptions:

A1: The estimation function $\kappa_\omega(\cdot)$ outputs the global minimizer of the in-sample loss:

$$\forall\omega\in\Omega\,\forall\mathcal{D}_{train}^{(m)} \in \left(\mathbb{R}^{d_x}\times\mathbb{R}^{d_y}\right)^K \exists!\theta^*\in\Theta: \tag{3.7}$$

$$\kappa_\omega(\mathcal{D}_{train}^{(m)}) = \theta^* = \operatorname*{arg\,min}_{\theta} \mathcal{L}(\mathcal{D}_{train}^{(m)};\theta,\omega). \tag{3.8}$$

A2: The training is conducted using a train-train split:

$$\begin{aligned} \hat{\omega} &= \operatorname*{arg\,min}_{\omega} \frac{1}{M}\sum_{m=1}^{M} \mathcal{L}(\mathcal{D}_{train}^{(m)};\hat{\theta}^{(m)},\omega) \\ \text{s.t.: } \hat{\theta}^{(m)} &= \kappa_\omega(\mathcal{D}_{train}^{(m)}). \end{aligned} \tag{3.9}$$

Assumption A1 is pragmatically motivated by our aim, which is finding optimal parametric models. This is in stark contrast to the widely popular family of meta-learning approaches derived from MAML (Finn et al., 2017) that primarily concentrate on estimation routines. There, ω typically represents the initial value of θ used in the estimation routine κ_ω or some additional information on how to adapt from θ (see, for example, Finn et al. (2017), Li et al. (2017), and Park and Oliva (2019)).

Assumption A2 implies that the training is not conducted with the train-val split (i.e., with $\mathcal{D}_{val}^{(m)}$ in the outer optimization problem and $\mathcal{D}_{train}^{(m)}$ in the inner optimization problem), which is typical for meta-learning. Instead, it is done with a train-train split (i.e., using $\mathcal{D}_{train}^{(m)}$ in both the outer and inner optimization problems), as is common in multi-task learning. In this setup the validation datasets $\mathcal{D}_{val}^{(m)}$ are still utilized, but typically for early stopping of the training process rather than being directly included in the objective function. This assumption is substantial because the training process, in this case, may not strictly correspond to how the model will be deployed in practice. That is, to the situation when observing a completely new task, $\mathcal{T}^{(M+1)}$, and being asked to adapt $\theta^{(M+1)}$ based on $\mathcal{D}_{train}^{(M+1)}$ to predict y in $\mathcal{D}_{val}^{(M+1)}$ while keeping ω fixed. Despite this, it appears justifiable in light of recent studies that demonstrate that for meta-learning, the commonly adopted train-val split might not always be preferable to a simpler train-train split (Bai et al., 2021) and that meta-learning and multi-task learning problems are closely connected (Wang et al., 2021)

The introduction of these assumptions substantially simplifies the optimization problem, as shown in the following proposition.

Proposition 1. *Under assumptions A1 and A2, there exist functions* $f(\cdot;\beta) : \mathbb{R}^{d_x} \to \mathbb{R}^{d_y}$ *parameterized by* $\beta \in B$ *and* $g(\cdot;\omega) : \Theta \to B$ *parameterized by* $\omega \in \Omega$*, such that the solution of*

$$\left\{\hat{\omega}, \left\{\hat{\theta}^{(m)}\right\}_{m=1}^{M}\right\} = \underset{\substack{\omega\in\Omega \\ \left\{\theta^{(m)}\right\}_{m=1}^{M}\in\Theta^M}}{\arg\min} \ \frac{1}{M}\sum_{m=1}^{M}\frac{1}{K}\sum_{i=1}^{K}\gamma(y_i^{(m)}, f(x_i^{(m)}; g(\theta^{(m)};\omega))) \tag{3.10}$$

coincides with the solution of the bilevel optimization problem introduced in Eq. 3.6.

The proposition demonstrates that under A1 and A2, the bilevel optimization problem in Eq. 3.6 collapses to a much simpler, single-level optimization problem. In this equivalent formulation, the model $f_\omega(\cdot, \theta^{(m)})$ is conveniently separated into two components: the *base* model $f(\cdot;\beta^{(m)})$, parameterized by $\beta^{(m)}$, which processes features to generate predictions, and a *meta* module $g(\theta^{(m)};\omega)$, which, based on the mesa parameter vector $\theta^{(m)}$, outputs the corresponding $\beta^{(m)}$. Thus, in effect, Proposition 1 allows for a simultaneous search over both parametric functions f_ω and their corresponding mesa parameters $\{\theta^{(m)}\}_{m=1}^{M}$.

Going back to the comparison with the clustering approach to the localization of global models, Proposition 1 allows one to deploy a distinct forecasting function for each time-series. These functions lie in the function space indexed by theta chosen to best describe the observed heterogeneity between individual series, instead of choosing a handful of functions in the unrestricted function space, one for each cluster, as is done in the case of clustering.

To allow for maximal flexibility, we express both the base model $f(\cdot;\beta)$ and the meta module $g(\cdot;\omega)$ as feedforward neural networks. The total size of the network $f(\cdot;\beta)$, represented by $d_\beta = \text{card}(\beta)$, controls the level of complexity with which the predicted values $\hat{y}_t$ depend on the input x_t. The size of the mesa parameters $d_\theta = \text{card}(\theta)$ corresponds to the number of degrees of freedom allotted to each task m and thus regulates the degree of globality/locality of the model.[2] Finally, the size of the network $g(\cdot;\omega)$, represented by $d_\omega = \text{card}(\omega)$, controls the nonlinearity of the model's response to mesa parameters $\theta^{(m)}$. Network g does not necessarily have to be fully connected. To reduce computational complexity, it is possible to leave some output nodes as orphaned constants, allowing the mesa parameters $\theta^{(m)}$ to affect only a part of the base model f, such as only its last layers.[3]

Importantly, given that the optimization problem in Eq. 3.10 is unconstrained and that both the meta parameters ω and the task-specific mesa parameters $\{\theta^{(m)}\}_{m=1}^{M}$ are optimized at the same level, the standard backpropagation techniques can be applied, considerably facilitating the training of the model.

When implementing the model, it is convenient to equivalently express the array of mesa parameters $\{\theta^{(m)}\}_{m=1}^{M}$ as a single neural network layer without any constants or nonlinearity. This layer takes, as input, the one-hot encoding of the task $q = e_m \in \{0,1\}^M$ and outputs the corresponding vector of mesa parameters $\theta^{(m)} = (\theta^{(1)}, \dots, \theta^{(M)})q$. The entire model can then be expressed as depicted in Figure 3.1. For brevity, we will refer

[2] If setting $d_\theta = 1$ would still yield too much flexibility, it is also possible to further regularize the mesa parameters. Allowing the regularization penalty to tend towards infinity renders the adaptation via θ ineffective, causing the model to collapse into a pure global model.

[3] This is motivated by the fact that adaptation predominantly occurs by altering the head of the network (Raghu et al., 2019; Lin et al., 2020).

to it simply as MtMs henceforth to emphasize the simultaneous training of both global **meta** parameters ω and task-specific **mesa** parameters $\{\theta^{(m)}\}_{m=1}^{M}$.

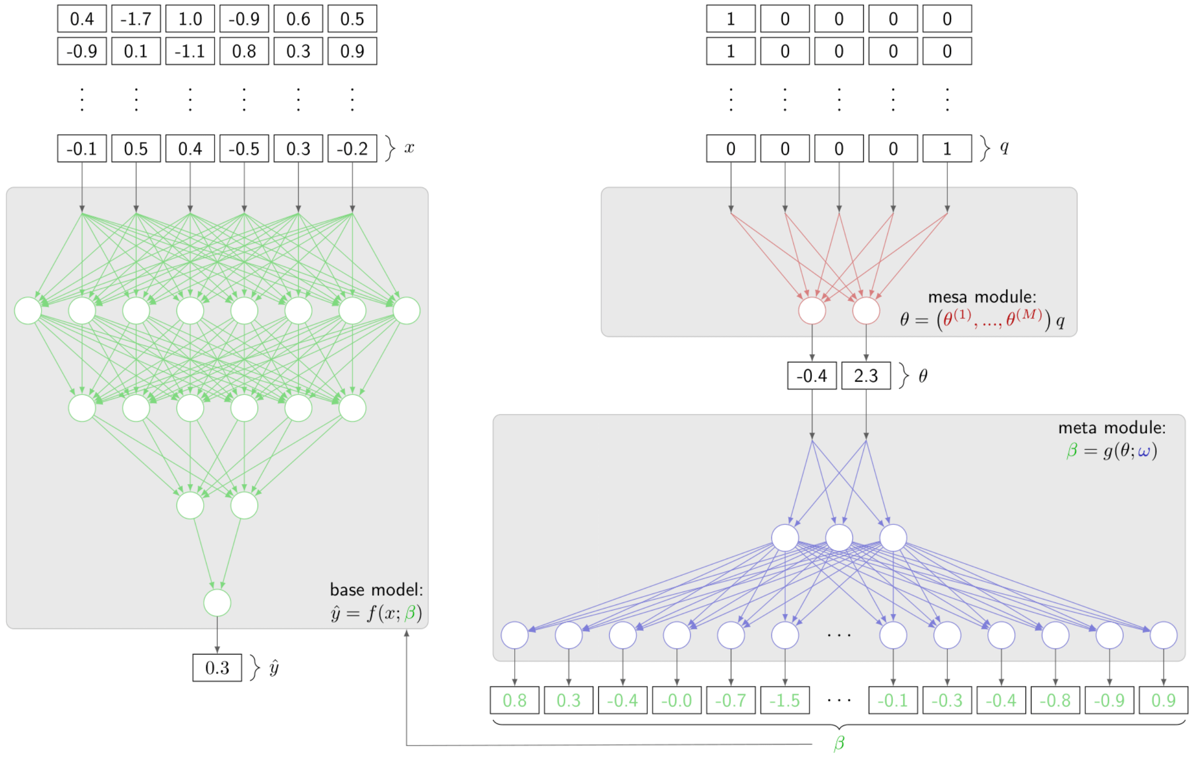


Figure 3.1: A diagram of the MtMs model for an illustrative example with 6 features and 5 tasks. The process of generating forecasts proceeds from the right to left. First, a one-hot encoded vector q, denoting to which task the observation belongs, is multiplied by a matrix of mesa parameters $(\theta^{(1)}, \ldots, \theta^{(M)})$ to extract the corresponding task-specific mesa parameter vector θ. This vector is then passed to the meta module $g(\theta; \omega)$ to generate task-specific parameters β of the base model $f(x; \beta)$. Lastly, the network $f(x; \beta)$ is used to process the corresponding feature vector x and generate the prediction $\hat{y}$.

Despite being trained under the multi-task learning paradigm, the model can be deployed for both multi-task and meta-learning problems. For multi-task learning, the model can be used as is without any further optimization. By providing more data from an already observed task m m, predictions can be made using $f_{\hat{\omega}}(\cdot; \hat{\theta}^{(m)}) = f(\cdot; g(\hat{\theta}^{(m)}; \hat{\omega}))$ with the corresponding estimated mesa parameter vector $\hat{\theta}^{(m)}$.

For meta-learning applications, we leverage Proposition 1, which states that the solution $\hat{\omega}$ from Eq. 3.10 can, under simplifying conditions A1 and A2, be interpreted as a parametric model $f_{\hat{\omega}}(\cdot; \theta) = f(\cdot; g(\theta; \hat{\omega}))$ that, out of all competing parametric models $\omega' \in \Omega$, delivers the smallest expected loss on a new task $\mathcal{T}^{(M+1)}$. To forecast $y_t^{(M+1)}$ of this previously unobserved task, it is therefore sufficient to perform optimization over the space of task-specific mesa parameters $\theta \in \Theta$:

$$\hat{\theta}^{(M+1)} = \arg\min_{\theta \in \Theta} \frac{1}{K} \sum_{t=1}^{K} \gamma(y_t^{(M+1)}, f(x_t^{(M+1)}; g(\theta; \hat{\omega}))), \tag{3.11}$$

while holding the model representation $\hat{\omega}$ fixed.

Note that this optimization is performed only in the low-dimensional space $\mathbb{R}^{d_\theta}$ and can be done using either backpropagation or conventional numerical optimization methods. In this sense, it is completely analogous to finding the parameters of any other parametric model. The only difference is that the functional form of the model $f_{\hat{\omega}}(\cdot; \theta)$, as represented by $\hat{\omega}$, is not presupposed by the researcher but instead is derived in a data-driven way specifically for the given family of prediction problems $p(\mathcal{T})$ in the initial meta-learning phase. Similar to a conventional parametric model manually crafted by a human expert, the parameter vector θ typically influences the prediction function $f_{\hat{\omega}}(\cdot; \theta) = f(\cdot; g(\theta; \hat{\omega}))$ in an interpretable way, as demonstrated in the applications presented later in this chapter (see Section 3.4 and Section 3.5). Further, though not addressed in this chapter, the fact that $\hat{\theta}^{(M+1)}$ is an extremum estimator allows inferences regarding model parameters, provided that regularity conditions are met.

This method belongs to the strain of meta-learning research where hypernetworks or embeddings are used to perform adaptation to individual tasks at a lower-dimensional manifold of the parameter space (see e.g. Lee and Choi, 2018; Zintgraf et al., 2019; Zhao et al., 2020; Flennerhag et al., 2020; von Oswald et al., 2022; Nava et al., 2023; Ramanarayanan et al., 2023). The main point of differentiation is that in these studies, hypernetworks are generally used to facilitate fine-tuning of network weights while retaining the MAML paradigm of bilevel optimization, where the inner optimization is restricted to a few gradient steps due to computational constraints. In contrast to this approach of fine-tuning network weights, MtMs sidestep the bilevel problem formulation by virtue of assumption A2, which, in turn, allows one to interpret mesa parameters $\{\theta^{(m)}\}_{m=1}^{M}$ as *global* optimizers of some underlying parametric model crafted specifically for the family of tasks $p(\mathcal{T})$. This is essential, as multistep task adaptation has been shown to be crucial in meta-learning (Lin et al., 2020). In this respect, the model is closely related to the seminal work of Shamsian et al. (2021), where a similar architecture with a custom training algorithm (pFedHN) is proposed for the task of personalized federated learning.

The MtMs model is also related to the fields of model selection and, by extension, model/forecast combination (for a comprehensive review, see Wang et al., 2023). Revisiting the example of a finite set of d_Ω possible model specifications $\Omega = \{\omega_i\}_{i=1}^{d_\Omega}$, as is common in model selection and forecast combination literature, a model selection algorithm can be represented as a function $\mathcal{A} : \mathcal{D}_{train}^{(m)} \rightarrow \{1,\, 2,\, ...,\, d_\Omega\}$. This function takes as input the training data $\mathcal{D}_{train}^{(m)}$ of a single series m and returns an index of the most appropriate model. The resulting prediction is then computed via:

$$\hat{y}_t^{(m)} = f_{w_{\mathcal{A}(\mathcal{D}_{train}^{(m)})}}(x_t^{(m)}; \hat{\theta}^{(m)}) \qquad \text{with} \qquad \hat{\theta}^{(m)} = \kappa_{\omega_{\mathcal{A}(\mathcal{D}_{train}^{(m)})}}(\mathcal{D}_{train}^{(m)}). \tag{3.12}$$

The algorithm $\mathcal{A}$ selecting the most suitable model can take various forms. These include comparing in-sample performance while controlling for model complexity (Wei, 1992) or evaluating pseudo out-of-sample performance (Zhang and Yang, 2015; Inoue et al., 2013). More closely aligned with our proposed model, $\mathcal{A}$ can also be trained in a prior meta-learning stage on a group of diverse time-series, leveraging cross-learning. A notable example of this approach is the FFORMS model by Talagala et al. (2023). This model trains a tree-based algorithm which, based on a vector of time-series features computed from $\mathcal{D}_{train}^{(m)}$, determines which parametric model $\{f_{\omega_i}, \kappa_{\omega_i}\}$ should be applied to deliver the best performance.

In this light, forecast combination can be viewed as an extension of model selection, that allows for choices both among the set of models $\{f_{\omega_i}, \kappa_{\omega_i}\}_{i=1}^{d_\Omega}$ and also among their linear combinations. Specifically, for $\mathcal{A} : \mathcal{D}_{train}^{(m)} \rightarrow \mathbb{R}^{d_\Omega}$, the resulting forecasts are expressed as:

$$\hat{y}_t^{(m)} = \sum_{i=1}^{d_\Omega} \mathcal{A}(\mathcal{D}_{train}^{(m)})[i] * f_{w_i}(x_t^{(m)}; \hat{\theta}^{(m)}) \qquad \text{with} \qquad \hat{\theta}^{(m)} = \kappa_{\omega_i}(\mathcal{D}_{train}^{(m)}). \tag{3.13}$$

These combinations can range from basic models that assign equal weight to each component model which, despite their simplicity, often exhibit surprisingly strong performance (Wang et al., 2023), to more complex models that use meta-learning to train $\mathcal{A}$. An example of the latter is FFORMA (Montero-Manso et al., 2020), an extension of the FFORMS model that employs a tree-based approach to determine the weights $\mathcal{A}(\mathcal{D}_{train}^{(m)})[i]$ of individual models based on time-series features.

Note that, compared to FFORMA, the MtMs model goes one step further in flexibility to accommodate a broader range of data generating processes. Rather than searching for the most suitable forecasting function among the d_Ω pre-specified families of functions (within each such family, the most suitable prediction function is found through optimization of

$\hat{\theta} \in \Theta$), MtMs aims to non-parametrically derive a single parametric model whose vector of parameters θ is best suited to accommodate the between-task variability observed in $\{\mathcal{T}^{(m)}\}_{m=1}^{M}$. This is especially beneficial in situations in which none of the pre-specified models (or their combinations) would accurately explain the DGPs sampled from $p(\mathcal{T})$. Furthermore, it is important to highlight that the optimization for $\{\hat{\omega}, \{\hat{\theta}^{(m)}\}_{m=1}^{M}\}$ is performed simultaneously, rather than in two steps in which, first, the most appropriate $\hat{\theta}^{(m)}$ is found for each model, and only then, with these parameters fixed, the auxiliary model for model weights $\mathcal{A}$ is estimated, as is done in FFORMA or forecast combination more generally. This is relevant, as the poor performance of more complex forecast combination schemes typically observed in empirical applications, a phenomenon dubbed the *forecast combination puzzle*, seems to be driven by the suboptimal two-step optimization with which the optimal weights are typically derived (Frazier et al., 2023). MtMs, by its very construction, does not suffer from this issue.[4]

3.4 Application: Sinusoidal Regression Task

To evaluate the potential of the MtMs to find the most appropriate parametric model for a given family of prediction problems, we first consider a simulation exercise originally proposed by Finn et al. (2017) to test the performance of MAML. Since then, this environment has frequently been used to compare competing meta-learning methods.

In particular, the tasks $\mathcal{T}^{(m)} = \{\mathcal{D}_{train}^{(m)}, \mathcal{D}_{val}^{(m)}\}$ are generated according to the following DGP:[5]

$$\begin{aligned} A^{(m)} &\sim U(0.1, 5) \\ b^{(m)} &\sim U(0, \pi) \\ x_i^{(m)} | A^{(m)}, b^{(m)} &\sim U(-5, 5) \\ y_i^{(m)} | x_i^{(m)}, A^{(m)}, b^{(m)} &= A^{(m)} * sin(x_i^{(m)} + b^{(m)}) \end{aligned} \tag{3.14}$$

The goal is to find the best model that can predict $y_i^{(m)}$ based on $x_i^{(m)}$ for $i > K$ after

[4] We are thankful to Prof. Andrey Vasnev for suggesting this connection.

[5] As the sinusoidal regression task is a cross-sectional exercise, we index individual observations by i rather than t to highlight that they are conditionally IID.

observing only $\mathcal{D}^{(m)}_{train}$, as measured by the mean squared error:

$$\begin{aligned} \mathcal{L}_m(\mathcal{D}^{(m)}_{val}; \hat{\theta}^{(m)}, \omega) &= \frac{1}{N-K} \sum_{i=K+1}^{N} (y_i^{(m)} - f_\omega(x_i^{(m)}; \hat{\theta}^{(m)}))^2 \\ \text{s.t.: } \hat{\theta}^{(m)} &= \kappa_\omega(\mathcal{D}^{(m)}_{train}) \end{aligned} \tag{3.15}$$

For fair comparison, we follow Finn et al. (2017) and set the base model to be a feedforward neural network with two hidden layers of size 40 and ReLU non-linearities. The number of mesa parameters, d_θ, is set to 2 and the meta module $g(\cdot; \omega)$ is a simple fully connected feedforward network with no hidden layers or non-linearities. For training of the MtMs, it is entirely sufficient to use only 1000 distinct tasks. This is far fewer then the 70000 task originally used in Finn et al. (2017) and in the followup studies. Likewise, the training is done with a fraction of the computational resources. It takes approximately 0.5 hour on a consumer grade mid-range CPU, which is in sharp contrast to the powerful GPU units used for training in other studies. Other simulation details follow Zhao et al. (2020) and are available in the replication repository[6].

Table 3.1 shows the mean squared error achieved by the MtMs for 5-shot learning and 10-shot learning. For comparison, we include the losses of commonly used meta-learning methods on this task (the performance of competing methods is taken from Park and Oliva (2019) and Zhao et al. (2020)). The proposed MtMs model outperforms all benchmark methods by an order of magnitude for both 5-shot learning and 10-shot learning of the sinusoidal task. In fact, the losses are in both cases very close to the theoretical minimum of 0, indicating that the MtMs is capable of recovering the data-generating process to such a degree that, when faced with only as few as 5 observations $\{x_i^{(m)}, y_i^{(m)}\}$ from task m, it is able to almost perfectly infer $y_i^{(m)}$ as a function of $x_i^{(m)}$ for the whole range $[-5, 5]$.

Figure 3.2 (resp. 3.3) shows predictions of the model $f_\omega(x; \theta)$ as a function of x for different values of mesa-parameters θ for $K = 5$ (resp. $K = 10$). As is apparent from Figure 3.2, plotted prediction functions closely resemble different sine waves, indicating the MtMs is indeed capable of correctly determining that each generated task follows a sine function with varying phase and amplitude. However, the mesa parameters $\theta = [\theta_1, \theta_2]$ explaining the variability between tasks do not directly correspond to the amplitude A and phase b. Instead, θ_1 regulates the amplitude (negatively), but to a lesser degree, it also regulates the phase (positively), while θ_2 primarily regulates the phase (positively) and, to a lesser

[6] `https://github.com/stanek-fi/MtMs_sinusoidal_task`

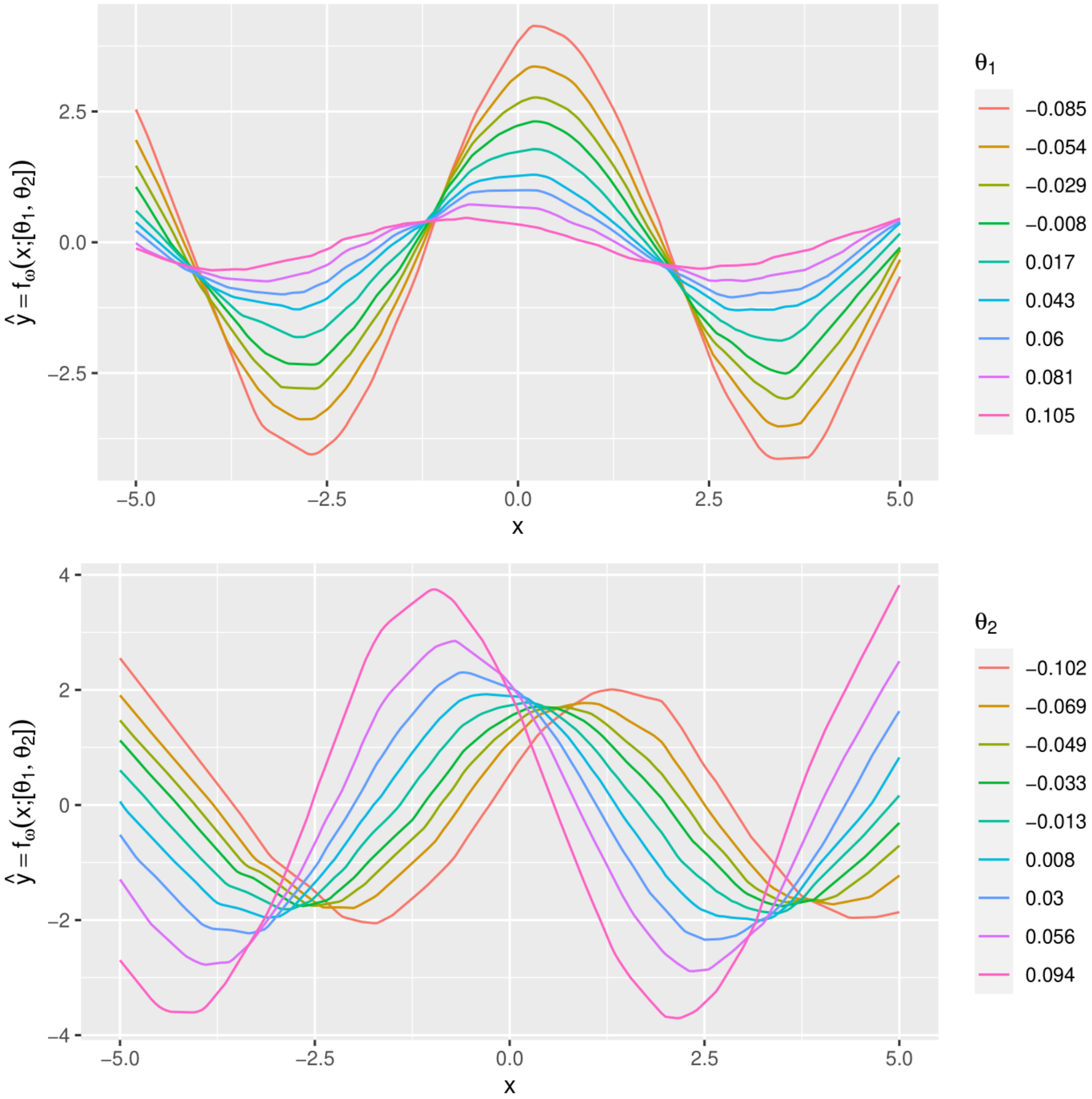


Figure 3.2: MtMs predictions for sinusoidal task ($K = 5$)
Plots of $f_\omega(x;\theta)$ as a function of x for different values of the mesa parameter vector θ. In the upper panel, the first mesa parameter θ_1 varies while θ_2 is fixed to its median value. In the lower panel, the second mesa parameter θ_2 varies while θ_1 is fixed to its median value.

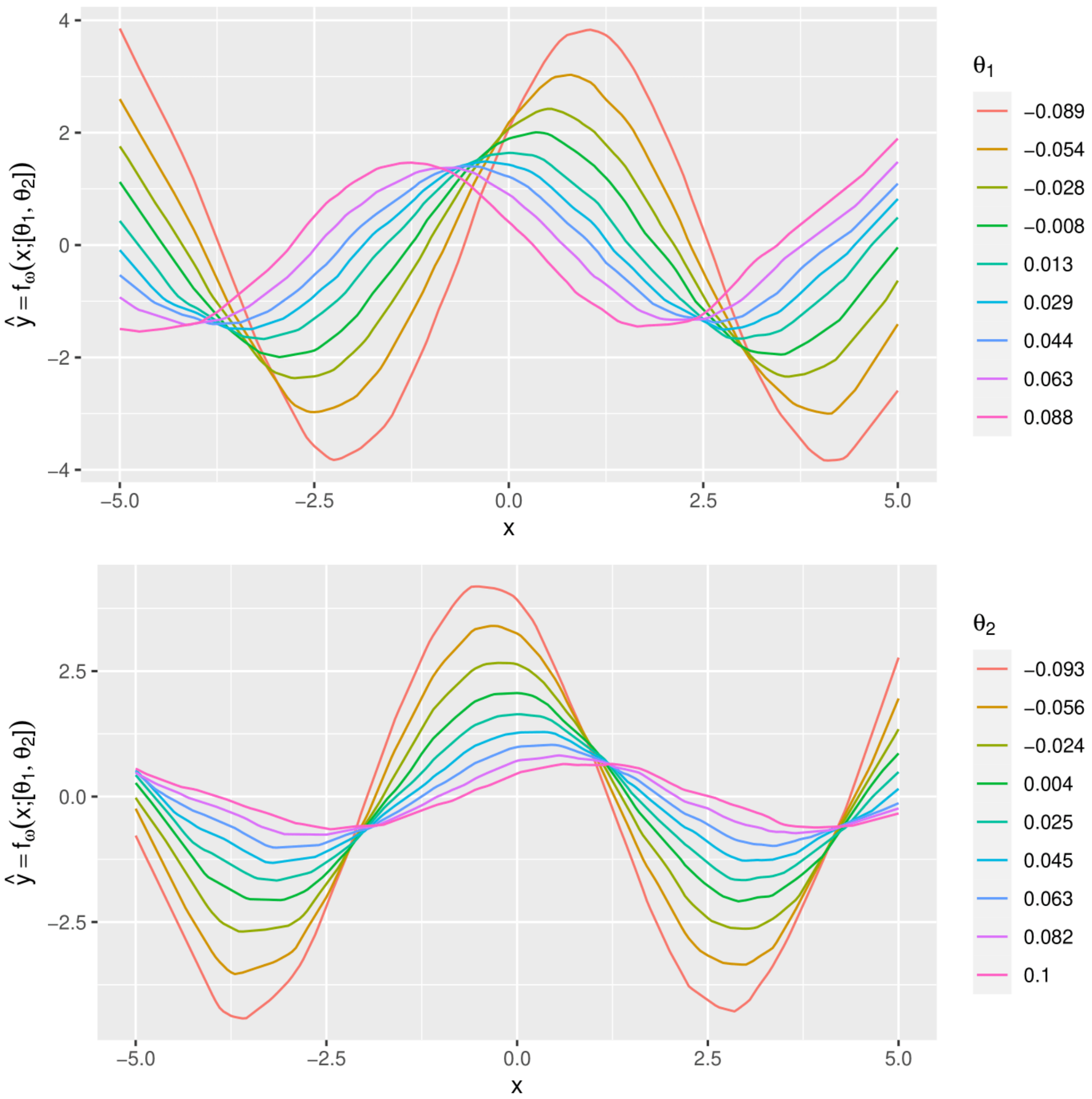


Figure 3.3: MtMs predictions for sinusoidal task ($K = 10$)
Plots of $f_\omega(x;\theta)$ as a function of x for different values of the mesa parameter vector θ. In the upper panel, the first mesa parameter θ_1 varies while θ_2 is fixed to its median value. In the lower panel, the second mesa parameter θ_2 varies while θ_1 is fixed to its median value.

degree, it also regulates the amplitude. This is not surprising, as there are infinitely many parametric models that are observationally equivalent to the DGP described in Eq. 3.14. In particular, any two vectors in $\mathbb{R}^2$ that are linearly independent are capable of spanning the whole space of $[b, A]$ just as well as the basis vectors used in Eq. 3.14. The MtMs hence generally converges to one of these equivalent parametrizations, not necessarily to the exact same parametrization used to simulate the data. In Figure 3.3, the prediction functions approximate sine waves even more closely, as $K = 10$ allows more accurate identification of the between-task variability.

Method	$K = 5$	$K = 10$
MAML (Finn et al., 2017)	$0.686^{\pm 0.070}$	$0.435^{\pm 0.039}$
LayerLR (Park and Oliva, 2019)	$0.528^{\pm 0.068}$	$0.269^{\pm 0.027}$
Meta-SGD (Li et al., 2017)	$0.482^{\pm 0.061}$	$0.258^{\pm 0.026}$
MC1 (Park and Oliva, 2019)	$0.426^{\pm 0.054}$	$0.239^{\pm 0.025}$
MC2 (Park and Oliva, 2019)	$0.405^{\pm 0.048}$	$0.201^{\pm 0.020}$
MH (Zhao et al., 2020)	$0.501^{\pm 0.082}$	$0.281^{\pm 0.072}$
MtMs (ours)	$0.022^{\pm 0.003}$	$0.014^{\pm 0.001}$

Table 3.1: Losses for sinusoidal task
Mean squared errors and corresponding 95% confidence intervals for different meta-learning methods.

Admittedly, the sinusoidal regression problem is relatively favorable to the MtMs because the data are generated using a clearly defined low-dimensional model, and MtMs is, at its core, a method for recovering unknown parametric models. To demonstrate that the good performance is not limited to artificial tasks like this one, in the next sections, we apply it to real-life forecasting problems posed in the M4 and M6 forecasting competition.

3.5 Application: M4 Forecasting Competition

To assess the ability of the MtMs model to localize global time-series forecasting models in more general settings than those encountered in M6, we also perform an extensive evaluation on the data from the M4 forecasting competition (Makridakis et al., 2020). We follow the evaluation framework of Montero-Manso and Hyndman (2021), who demonstrated the surprising performance of simple global models, including a simple pooled OLS with lagged values of the time series as regressors. We extend this forecasting exercise by exploring the extent to which the performance of the pooled OLS can be improved through localization via time-series clustering and MtMs.

Following Montero-Manso and Hyndman (2021), we focus on time-series with yearly, quarterly, monthly, and weekly frequencies (for forecast horizons of 6, 8, 18, and 13, respectively, using the recursive forecasting scheme) and use the MASE loss function with scaling applied as a preprocessing step. The feature vectors contain lagged values of the given time-series: $x_t^{(m)} = [y_{t-1}^{(m)}, y_{t-2}^{(m)}, \ldots, y_{t-d_x}^{(m)}]^\top$. For a time-series m of length d_m, the design matrix is defined as $X^{(m)} = [x_{d_x+1}^{(m)}, x_{d_x+2}^{(m)}, \ldots, x_{d_m}^{(m)}]^\top$, and the dependent variable vector is $y^{(m)} = [y_{d_x+1}^{(m)}, y_{d_x+2}^{(m)}, \ldots, y_{d_m}^{(m)}]^\top$. By stacking $\{y^{(m)}\}_{m=1}^M$ and $\{X^{(m)}\}_{m=1}^M$, one can obtain the design matrix and dependent variable vector for pooled regression to estimate the pooled β for all time-series of a given frequency. Likewise, after performing time-series clustering to account for heterogeneity across series, one can obtain β for each cluster of similar series.

To mimic the same settings with MtMs, we set $f(x; \beta^{(m)}) = x^\top \beta^{(m)}$, and define $g(\theta; \omega)$ as a simple neural network with no hidden layer or nonlinearity: $\beta^{(m)} = g(\theta^{(m)}; \omega) = \omega^b + \omega^w \theta^{(m)}$, where $\omega^b \in M(d_x, 1)$ and $\omega^w \in M(d_x, d_\theta)$. The predictions for time-series m can hence be expressed as

$$\hat{y}^{(m)} = X^{(m)} \underbrace{(\omega^b + \omega^w \theta^{(m)})}_{\beta^{(m)}}. \tag{3.16}$$

This expression shows that this special case of MtMs is analogous to performing PCA in the latent space of unobservable true regression coefficients $\{\beta^{*(m)}\}_{m=1}^M$. The bias vector ω^b captures the central tendency of $\{\beta^{*(m)}\}_{m=1}^M$ and corresponds to the action of demeaning variables prior to PCA. The column vectors of matrix ω^w are optimized to best explain the variability of the true unobserved $\{\beta^{*(m)}\}_{m=1}^M$, analogous to the loading vectors of individual principal components. The task-specific parameter vector $\theta^{(m)}$ measures the exposure to variance-explaining factors $\omega^w[:, i]$ for a given time-series, and corresponds to the row m of the score matrix from PCA.

Similarly to PCA, dimensionality reduction can be performed by choosing the number of factors (d_θ) used to explain the variability of $\{\beta^{*(m)}\}_{m=1}^M$. Choosing $d_\theta = 0$ is equivalent to estimating a pooled regression on the time-series, whereas choosing $d_\theta = d_x$ is equivalent to estimating a separate regression for each time-series m. In practice, given that the DGPs of many time-series are likely similar, only a handful of factors $\omega^w[:, i]$ are necessary to successfully explain most of the variability across time-series. Note that, unlike principal component regression (see, e.g., Hadi and Ling, 1998), where the dimensionality reduction is performed on the pooled design matrix as a preprocessing step, here the reduction is performed in the *latent* space of regression coefficients jointly with the estimation. In this sense, it is similar to reduced-rank regression (Izenman, 1975), with the exception that we are searching for a lower-dimensional representation of a set of regression coefficients across multiple tasks/time-series, rather than within a single dataset with multiple dependent variables.

Table 3.2 displays the average MASE for OLS localized via MtMs with $d_\theta = 2$ on the M4 datasets. To facilitate training, we leverage the fact that the optimal $\{\{\omega^b, \omega^w\}, \{\theta^{(m)}\}_{m=1}^M\}$ can be derived iteratively in closed form under the L2 loss[7], and we use these estimates to initialize MtMs. After initialization, the training is performed using backpropagation with the Adam optimizer, a learning rate of 0.001 and a minibatch size of 1,000 time-series under the MASE loss. For comparison with conventional localization techniques, we cluster the time-series into $\{2^i\}_{i=2}^{10}$ clusters using k-means on `stl_features` (seasonality & trend), `entropy` and `acf_features` (autocorrelation) from the `tsfeatures` package (Hyndman et al., 2023) and estimate regression coefficients for each cluster individually. For each frequency, we set the number of lags d_x to the maximum value according to the shortest series, following the setup of Montero-Manso and Hyndman (2021). For reference, we also include the performance of OLS on the pooled dataset and two widely used local models: `ETS` (Hyndman et al., 2002b) and `auto.arima` (Hyndman and Khandakar, 2008b).

With the exception of the yearly frequency, where localization provides only marginal improvements and where the two degrees of freedom per series likely lead to over-fitting, OLS localized via MtMs outperforms non-localized OLS and OLS localized via clustering across all cluster sizes. Furthermore, the simple linear parametric model derived in a data-driven way via MtMs;

$$\hat{y}_t^{(m)} = f_\omega(x_t^{(m)}; \theta^{(m)}) = x_t^{(m)\top}(\omega^b + \omega^w \theta^{(m)}) \tag{3.17}$$

[7] Eq. 3.16 can be expressed for all tasks m simultaneously as $\hat{\mathbf{y}} = \mathbf{X}(\tilde{\boldsymbol{\theta}} \otimes I_{d_x+1})\text{vec}(\omega)$ where $\hat{\mathbf{y}} = \left[\hat{y}^{(1)\top}, \dots, \hat{y}^{(M)\top}\right]^\top$, $\mathbf{X} = \text{blkdiag}(\{X^{(m)}\}_{m=1}^M)$, $\omega = [\omega^b, \omega^w]$, $\boldsymbol{\theta} = \left[\theta^{(1)}, \dots, \theta^{(M)}\right]^\top$, and $\tilde{\boldsymbol{\theta}} = [\mathbf{1}, \boldsymbol{\theta}]$. This results in to first-order conditions for vec(ω): $\text{vec}(\omega) = \left(H^\top H\right)^{-1} H^\top \mathbf{y}$ where $\mathbf{y} = \left[y^{(1)\top}, \dots, y^{(M)\top}\right]^\top$ and $H = \mathbf{X}(\tilde{\boldsymbol{\theta}} \otimes I_{d_x+1})$. First-order conditions for $\{\theta^{(m)}\}_{m=1}^M$ are $\{\theta^{(m)} = \left(Q^\top Q\right)^{-1} Q^\top (y^{(m)} - X^{(m)}\omega^b)\}_{m=1}^M$ where $Q = X^{(m)}\omega^w$. By iterating over these two sets of first-order conditions, $\{\{\omega^b, \omega^w\}, \{\theta^{(m)}\}_{m=1}^M\}$ converge to their joint optimal values.

with $\{\omega^b, \omega^w\}$ fixed, outperforms conventional local models crafted by human experts on all frequencies except yearly.

model	Yearly	Quarterly	Monthly	Weekly
ETS	3.478	1.164	0.948	2.513
auto.arima	3.407	1.161	0.929	2.542
OLS (pooled)	3.059	1.222	0.957	2.275
OLS (2 clusters)	3.011	1.218	0.950	2.246
OLS (4 clusters)	**2.990**	1.225	0.953	2.179
OLS (8 clusters)	3.020	1.229	0.949	2.194
OLS (16 clusters)	3.075	1.220	0.952	2.181
OLS (32 clusters)	3.132	1.214	0.950	
OLS (64 clusters)	3.188	1.210	0.949	
OLS (128 clusters)	3.258	1.205	0.943	
OLS (256 clusters)	3.304	1.201	0.939	
OLS (512 clusters)	3.402	1.197	0.936	
OLS (1024 clusters)	3.538	1.197	0.930	
OLS (localized via MtMs)	4.050	**1.133**	**0.911**	**2.104**

Table 3.2: Losses for the M4 datasets
Mean MASE losses for individual models on yearly, quarterly, monthly, and weekly datasets from the M4 competition. Bold text indicates the best-performing model for each frequency. Losses of OLS with more than 16 clusters for the weekly frequency are not available due to an insufficient number of observations to estimate OLS on all clusters.

Similarly to sinusoidal regression, the simple structure of the model allows us to visualize the heterogeneity across DGPs identified in the datasets. As an example, Figure 3.4 displays the column vectors $\omega^w[:,1]$ and $\omega^w[:,2]$ for the monthly frequency. Parameter $\theta[2]$ primarily regulates the persistence of the DGP (positively affecting the dependence on lag 1) and seasonality (negatively affecting the dependence on lags $\{12, 24, 36\}$). To a lesser extent, it also captures seasonality at lag 6, likely driven by time-series with bi-annual seasonal behavior. Parameter $\theta[1]$ also regulates persistence (negatively) and seasonality (negatively), but in addition appears to influence the decay of seasonal behavior, as evidenced by the gradually decreasing values of $\omega^w[:,1]$ corresponding to lags $\{13, 14, 25, 26, 37, 38\}$. By varying these two parameters $\theta[1]$ and $\theta[2]$, we can approximately span the space of regression coefficients corresponding to DGPs encountered in the M4 monthly dataset.

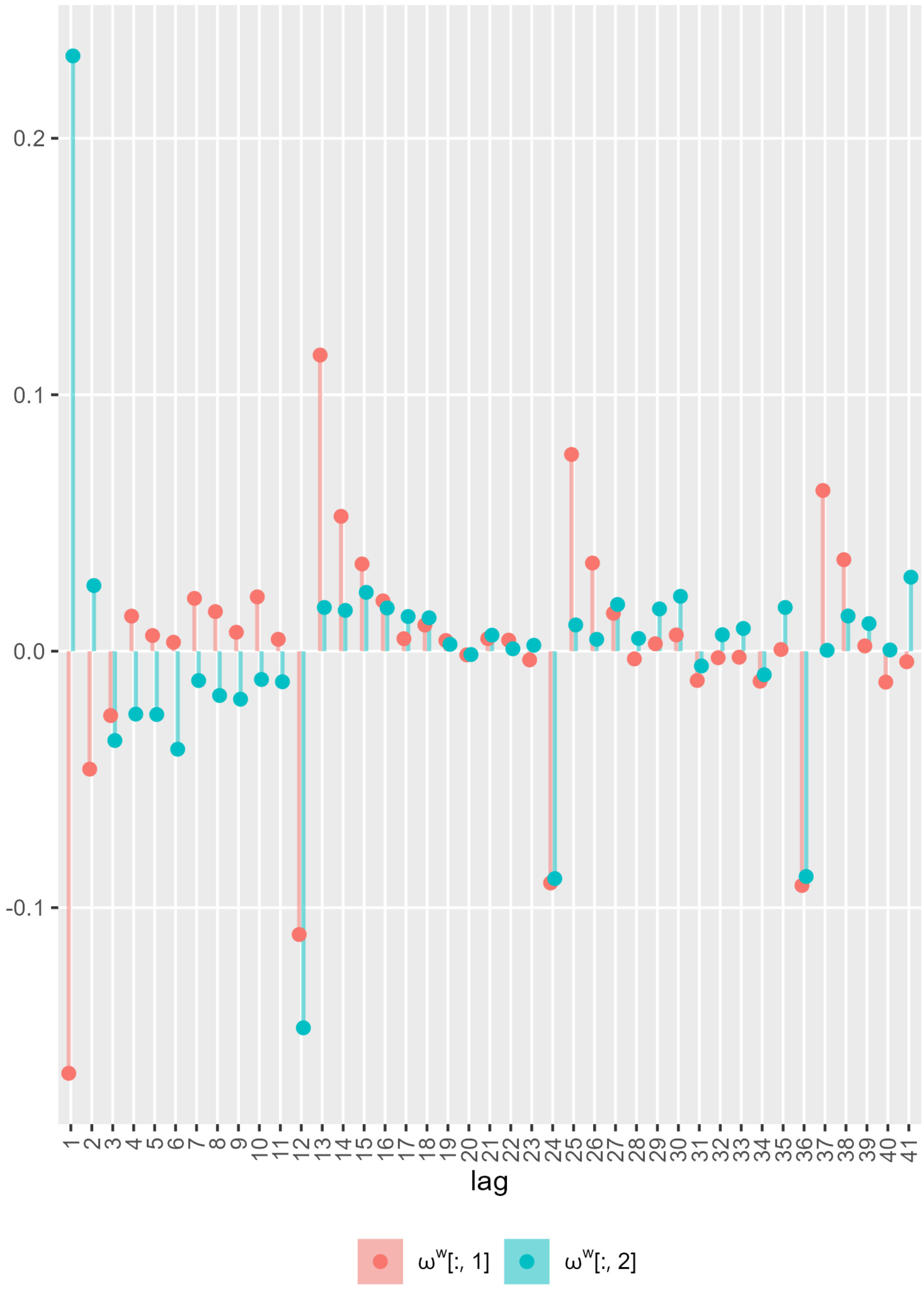


Figure 3.4: Estimated column vectors of ω^w for the M4 monthly dataset

3.6 Application: M6 Forecasting Competition[8]

The M6 Financial Forecasting Competition (see Makridakis et al., 2022) spanned from March 2022 to February 2023 and focused on a universe of 100 assets: 50 S&P 500 stocks and 50 international ETFs. In the forecasting challenge, participants were tasked with predicting probabilities for each asset's next 4-week returns falling into one of five quintiles relative to other assets in the universe. The accuracy of these predictions was assessed using the ranked probability score (RPS) loss after the 4-week period had passed. In the investment challenge, participants were required to submit portfolio weights for the upcoming 4-week interval. These portfolios were then evaluated based on risk-adjusted returns (IR). Additionally, participants competed in a duathlon, which combined both forecasting and investment challenges. The duathlon ranking was computed as an arithmetic mean of participants' ranks in the forecasting and investment challenges. This section describes the methods we employed for our submissions, which achieved 4th place in the forecasting challenge, 6th place in the investment challenge, and ultimately secured the 1st place in the duathlon.

MtMs, while broadly applicable, is especially well-suited for time-series forecasting, where the number of observations is typically insufficient to apply nonparametric methods on a per-series basis, but where multiple realizations of similar (but not necessarily ex-ante identical) time-series are available. In particular, in the case of M6, it allowed us to perform a search over the space of prediction functions parameterized by some latent parameter vector specific to each asset, as opposed to finding a single prediction function for all assets, as one would do when applying a conventional nonparametric model on pooled data. The latent parameter vector can absorb heterogeneity in DGPs across assets, hence improving the performance.

In the context of the forecasting challenge in the M6 competition, each task m represents a single asset. The variable $y_t^{(m)} \in \{0,1\}^5$ serves as an indicator for the quintile to which the returns of asset m belong within the 4-week interval t, and $x_t^{(m)}$ is a feature vector used for prediction.

[8] **The model specification evolved slightly during the competition. This section details the model's state as of the 12th and final submission. For the evolution of the model, please refer to the original repository `https://github.com/stanek-fi/M6` which contains unaltered scripts used for the submissions.**

3.6.1 Data augmention

To enhance training stability and performance, we augment the dataset with assets beyond the 100 specified in the M6 universe. Data augmentation is particularly advantageous for the MtMs model, as even if additional assets have substantially different DGPs from those in the M6 universe, these variations are likely to be absorbed by $\theta^{(m)}$.

We augment the original 50 stocks and 50 ETFs with an additional 450 stocks and 450 ETFs. These assets are selected from a pool of assets with sufficient trading activity[9] (must be at least 0.5 times the minimal trading activity observed in the M6 universe) and price history (must span from at least 2015 to the current date) to match the volatility observed in the M6 universe (the top 450 stocks/ETFs with the highest likelihood of their volatility being observed among the stocks/ETFs in the M6 universe are selected). Finally, the additional 450 stocks and 450 ETFs were randomly divided into 9 additional M6-like universes in order to compute quintiles $y_t^{(m)}$ of returns.[10]

In addition to augmentation across the dimension M, we calculate quintiles $y_t^{(m)}$ and features $x_t^{(m)}$ for 4-week intervals shifted by 1, 2, and 3 weeks relative to the actual start of the competition (2022-03-07). Assuming the time-series $y_t^{(m)}$ and $x_t^{(m)}$ are stationary, such augmentation does not alter the objective in any way and allows us to effortlessly quadruple the amount of data per asset m, further enhancing the stability of the training process.[11]

3.6.2 Features

As features $x_t^{(m)}$, we utilize an indicator for whether a given asset is an ETF, its own lagged 4-week returns and volatilities (up to lag 7), and an array of technical trading indicators from the TTR package (Ulrich, 2021), calculated based on historical prices. We opt for TTR because it offers a unified interface, allowing us to generate a diverse set of features programmatically without requiring extensive supervision or manual adjustments. A complete list of all 81 features is provided in Table 3.3 in Appendix 3.A.[12] Finally, we

[9] Measured by the product of the daily traded volume and the closing price.

[10] Note that computing quintiles based on all 900 additional assets at once does not generally align with the original objective.

[11] All the features in $x_t^{(m)}$ were either normalized by price or differenced to induce stationarity.

[12] Some indicators are multivariate and/or are computed with different lengths of the rolling window. The feature selection process involved initially training an XGBoost model (Chen et al., 2023) using all available technical trading indicators from TTR and subsequently pruning the least important features.

impute missing values with medians, and normalize the features to zero mean and unit variance. Missing observations, which comprise 0.947% of the dataset, are predominantly found at the beginning of each series. These gaps result from our having insufficient number of prior data points to calculate certain features and do not appear to substantially influence the results.

3.6.3 Model & training

The base model $f(\cdot;\beta)$ is a feedforward neural network comprising two hidden layers with 32 and 8 units, featuring leaky ReLU nonlinearity and a dropout rate of 0.2. The output layer has 5 units and utilizes a softmax transform. The meta module $g(\cdot;\omega)$ is a trivial feedforward network with no hidden layers or nonlinearity. One mesa parameter ($d_\theta = 1$) is allotted to each asset, influencing the weights and biases of the final layer in $f(\cdot;\beta)$. The architecture of the entire model is displayed in Figure 3.5.

To train the model, we utilize data from 2000 to 2022 for training, reserving the remaining data for testing. Given the high sensitivity of hypernetworks to their initialization (Beck et al., 2023), our training process consists of two steps. In the first step, the base model is trained on pooled data without taking into account which data belongs to which task. This training is conducted under the RPS loss using the Adam optimizer with a learning rate of 0.01, a minibatch size of 200, and early stopping.

In the second step, the trained weights from the first step serve as an initialization for the bias of the meta module $g(\cdot;\omega)$. Meanwhile, the weights of the meta module are initialized uniformly on the interval $[-1,1]$, and mesa parameters $\{\theta^{(m)}\}_{m=1}^{M}$ are set to 0. This means that the optimization begins from a point where the MtMs model is already proficient at predicting $y_t^{(m)}$, and the objective now is primarily to capture any systematic differences among the DGPs of individual assets through the mesa parameters $\{\theta^{(m)}\}_{m=1}^{M}$. The optimization is carried out iteratively using the Adam optimizer, with gradually decreasing learning rates, minibatches consisting of 100 randomly selected assets and early stopping. We employ this repeated training scheme because MtMs can be challenging to train, with the optimizer often struggling to adjust the model weights for improved test loss on the initial attempt. Multiple iterations are typically required. Finally, to make predictions, we can readily employ mesa parameters $\theta^{(m)}$ corresponding to the original asset universe without any further training (i.e., a multi-task learning scenario).

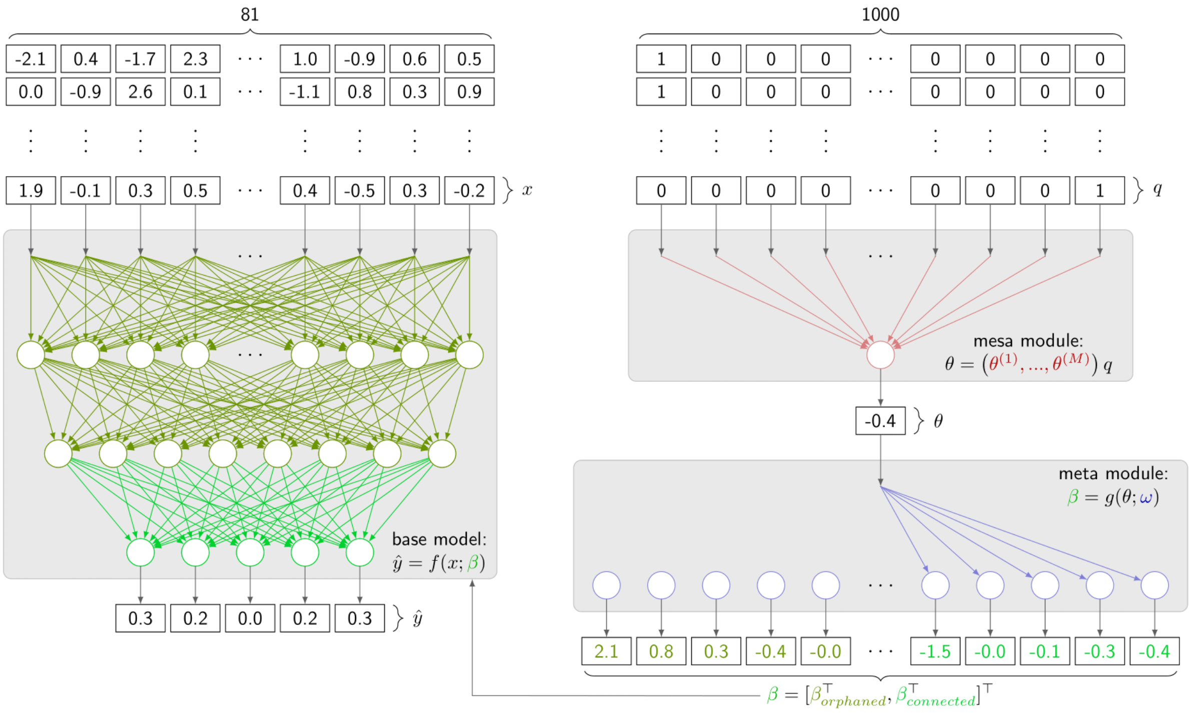


Figure 3.5: A diagram of the MtMs model applied to M6. In the case of M6, there are 1000 tasks/assets (100 specified by the organizers and 900 from the additional 9 auxiliary M6-like datasets). Each asset is allotted one univariate mesa parameter θ, which, through the meta module $g(\theta; \omega)$, determines the parameters β of the network $f(x; \beta)$. This network then processes the corresponding feature vector x to generate the prediction $\hat{y}$. The meta module $g(\theta; \omega)$ is a trivial single-layer neural network that connects θ to the weights and biases of the last layer of the network f; $\beta_{connected}$. The remaining nodes corresponding to parameters $\beta_{orphaned}$ are not influenced by θ and are hence constant across all tasks/assets.

Although the rankings of individual assets are intrinsically related (with exactly 20 assets belonging to each quintile within each universe), we choose to disregard this dependence and submit predictions $\hat{y}_t^{(m)} = f(x_t^{(m)}, g(\widehat{\theta}^{(m)}; \widehat{\omega}))$ without any post-processing or further adjustments.[13] While harmonizing the predictions could potentially yield performance improvements, we did not pursue this as the universe's size of 100 assets is adequate to ensure that $y_t^{(m)}$ is at least approximately unrelated in this regard.

3.6.4 Implications for Investment Decisions

The M6 competition organizers documented a significant disconnect between forecasting accuracy and investment performance. Analyses show almost zero correlation between the two, with only one team outperforming the investment benchmark over four quarters, and none across all twelve months. Notably, the top performing forecasting teams constructed inefficient portfolios, while the best investment teams submitted less accurate forecasts (Makridakis et al., 2023).

Our analysis of forecasts generated by the MtMs model seems to align with this finding and offers a possible explanation for this seemingly paradoxical disconnect between the two challenges. Despite the model's predictions performing well when measured by RPS loss[14], attaining an RPS of 0.15689, they contain surprisingly little information about expected returns. This severely limits their practical utility for forming investment portfolios, except for risk management purposes. Figure 3.6 shows the predicted probabilities of the 1st (resp. 2nd) quintile plotted against predicted probabilities of the 5th (resp 4th) quintile for individual assets throughout the competition. Predictions generally traverse along the diagonal line, implying that an increased probability of exceptionally good performance, relative to other assets, is accompanied by an increased probability of exceptionally poor performance, and vice versa, thus failing to provide any clear recommendations on which positions to take. This finding appears to align with the efficient market hypothesis (see, e.g., Malkiel, 2005), which posits that it is impossible to achieve abnormal returns based

[13] The only exceptions are the predictions for the DRE stock during submissions 10-12. After DRE stock was acquired by PLD, it exhibited zero price changes from that point forward. To address this, we overrode the predictions with observed frequencies with which a hypothetical asset with zero returns would belong to individual quintiles.

[14] Disentangling the precise causes of the model's relatively good performance is challenging. However, the training metrics suggest that the considered assets are relatively homogenous. The most significant improvements over naive predictions were achieved through joint training, with adaptation playing a secondary role. MtMs nonetheless still provided the advantage of using a much broader universe of assets for training without concerns about their dissimilarity to the assets specified by the organizers.

on information contained in the price history.

In general, the predictability of quintiles of rank does not necessarily imply predictability of expected returns; many DGPs for asset returns with identical means are compatible with non-uniform and predictable quintiles. Even the minor asymmetries in quintile predictions (i.e., a situations in which $\hat{P}(quintile_{m,t} = 1) \neq \hat{P}(quintile_{m,t} = 5)$ or $\hat{P}(quintile_{m,t} = 2) \neq \hat{P}(quintile_{m,t} = 4)$) occasionally observed in Figure 3.6 are not necessarily indicative of mean predictability. Instead, they are likely caused by a varying degree of asymmetry in the distribution of returns across different assets.

This posed a challenge regarding how to best approach the investment challenge in which we also participated. Given the lack of information in predictions regarding the expected returns, which would allow us to attain abnormal returns, we opted to use the investment challenge as ancillary to the prediction challenge and to strategically regulate the risk depending on the current ranking to improve the chances of securing a good enough rank in the duathlon challenge.[15] In particular, to maximize the probability of securing the top rank, it is desirable to take more risky positions when one ranks poorly in the public leaderboard, attempting to improve otherwise hopeless positions. Conversely, more conservative positions might be warranted if one already holds a sufficiently good rank and only wishes to maintain it.

A formalization of this type of approach as a dynamic programming problem and analysis of its performance is beyond the scope of this chapter, but can be found in (Staněk, 2023b). The simulations suggest that employing such an adversarial portfolio strategy can significantly improve the likelihood of achieving a favorable rank within the leaderboard. This effect is particularly notable for the highest rankings; the probability of securing the 1st place is approximately 3 times higher than expected by chance, comparable to that of a participant consistently generating double the market returns. The advantage for less extreme placements is less pronounced, with the probability of securing the 20th place or better being approximately 1.5 times higher than expected by chance.

[15] As the objective is defined in terms of risk-adjusted returns, it is challenging to directly control risk by forming portfolios with varying degrees of return variability. To circumvent this issue, we leveraged the competitive nature of the competition, where only relative performance matters, and the fact that participants were primarily taking long positions. By varying the proportion of short positions in the portfolio, one can control the extent to which returns of the submitted portfolio would be negatively or positively correlated with the returns of other participants at large.

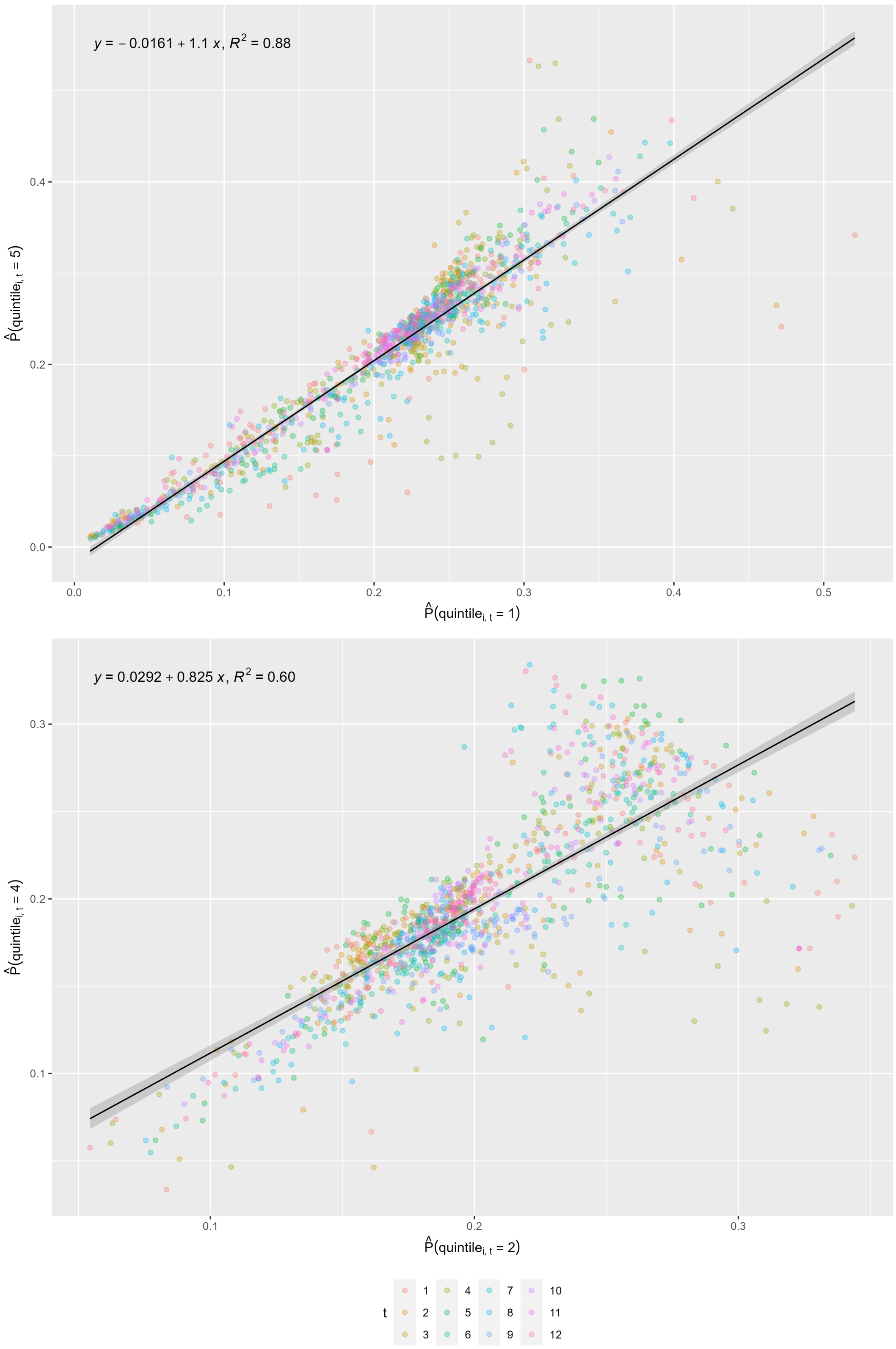


Figure 3.6: Predicted probabilities of the 1st quintile plotted against the probabilities of the 5th quintile (upper panel) and predicted probabilities of the 2nd quintile plotted against the probabilities of the 4th quintile (lower panel).

3.7 Conclusions

We propose a meta-learning/multi-task model based on hypernetworks. This approach enables the creation of a parametric model specifically optimized for a particular family of prediction tasks. In this way, the model's parameters are used to capture any between-task variability, while features of the DGP that are approximately invariant across tasks are learned from the pooled data. Unlike other meta-learning approaches, the training of the model does not rely on the computation of higher-order derivatives and can be done via standard backpropagation techniques, considerably reducing the computational resources required.

The model outperforms state-of-the-art meta-learning approaches on the sinusoidal regression task by an order of magnitude. It is capable of almost perfectly recovering the underlying parametric model (or one of its equivalent representations) and delivering near-oracle-level performance. In the second application, we apply MtMs under the multi-task learning paradigm to the time-series from the M4 forecasting competition, following the evaluation framework of Montero-Manso and Hyndman (2021). A simple linear model localized via MtMs outperforms both the corresponding global model applied on pooled data and models localized via clustering for the majority of series. Moreover, this very simple parametric linear model derived in a data-driven way through MtMs outperforms conventional widely used local models such as `ETS` (Hyndman et al., 2002b) and `auto.arima` (Hyndman and Khandakar, 2008b) on the majority of series. The third application in which we showcase the model is the M6 Financial Forecasting Competition. There, the MtMs model allowed us to leverage a much broader universe of assets for training while also acknowledging potential heterogeneity among the assets. The model attained an RPS of 0.15689, securing the 4th place in the forecasting challenge and ultimately the 1st place in the overall duathlon ranking.

These three applications clearly demonstrate the potential of MtMs in solving difficult prediction problems where neither pure global nor pure local models are completely appropriate. An exciting area for further research, which we plan to explore, is to test its applicability on a broader range of meta-learning problems, especially few-shot image recognition—a typical domain of meta-learning approaches—and general few-shot reasoning tasks.

3.8 Proofs

Proof of Proposition 1. Define $g(\theta^{(m)};\omega) = \{\theta^{(m)},\omega\}$ and $f(\cdot,\beta^{(m)}) = f_{\beta^{(m)}[2]}(\cdot;\beta^{(m)}[1])$ with $B = \Theta \times \Omega$. Under A1 and A2, the bilevel optimization problem in Eq. 3.6 reads as follows:

$$\begin{aligned}\hat{\omega} &= \arg\min_{\omega} \underbrace{\frac{1}{M}\sum_{m=1}^{M}\frac{1}{K}\sum_{i=1}^{K}\gamma(y_i^{(m)}, f(x_i^{(m)}; g(\theta^{(m)};\omega)))}_{\equiv Q(\omega,\{\kappa_\omega(\mathcal{D}_{train}^{(m)})\}_{m=1}^{M})} \\ \text{s.t.: } \hat{\theta}^{(m)} &= \kappa_\omega(\mathcal{D}_{train}^{(m)}) \\ &= \arg\min_{\theta} \frac{1}{K}\sum_{i=1}^{K}\gamma(y_i^{(m)}, f(x_i^{(m)}; g(\theta;\omega))).\end{aligned} \tag{3.18}$$

The assumption of existence and uniqueness of the inner optimization problems (A1) guarantees that the objective $Q(\omega,\{\kappa_\omega(\mathcal{D}_{train}^{(m)})\}_{m=1}^{M})$ is properly defined. Let us denote the set of solutions to the bilevel problem (Eq. 3.18) as $\Omega_B^* \subset \Omega$, and the ω component of the set of solutions to the single-level problem (Eq. 3.10) as Ω_S^*. The fact that $\Omega_B^* = \Omega_S^*$ directly stems from the fact that the individual components (m) of the outer optimization objective $Q(\cdot)$ coincide with the inner optimization objectives:

Let $\omega^* \in \Omega_B$. By virtue of optimality,

$$\forall \omega \in \Omega: \; Q(\omega^*, \{\kappa_{\omega^*}(\mathcal{D}_{train}^{(m)})\}_{m=1}^{M}) \leq Q(\omega, \{\kappa_\omega(\mathcal{D}_{train}^{(m)})\}_{m=1}^{M}). \tag{3.19}$$

From the definition of κ_ω and the additivity of $Q(\cdot)$, it also holds

$$\forall \omega \in \Omega \, \forall \{\theta^{(m)}\}_{m=1}^{M} \in \Theta^M: \; Q(\omega, \{\kappa_\omega(\mathcal{D}_{train}^{(m)})\}_{m=1}^{M}) \leq Q(\omega, \{\theta^{(m)}\}_{m=1}^{M}). \tag{3.20}$$

Combining these, we obtain

$$\forall \omega \in \Omega \, \forall \{\theta^{(m)}\}_{m=1}^{M} \in \Theta^M: \; Q(\omega^*, \{\kappa_{\omega^*}(\mathcal{D}_{train}^{(m)})\}_{m=1}^{M}) \leq Q(\omega, \{\theta^{(m)}\}_{m=1}^{M}) \tag{3.21}$$

which implies $\omega^* \in \Omega_S$.

Let $\omega^* \in \Omega_S$, and let $\{\theta^{*(m)}\}_{m=1}^{M} \in \Theta^M$ be the corresponding θ component of the solution. By virtue of optimality,

$$\forall \omega \in \Omega \, \forall \{\theta^{(m)}\}_{m=1}^{M} \in \Theta^M: \; Q(\omega^*, \{\theta^{*(m)}\}_{m=1}^{M}) \leq Q(\omega, \{\theta^{(m)}\}_{m=1}^{M}). \tag{3.22}$$

Since $\kappa_\omega(\mathcal{D}_{train}^{(m)}) \in \Theta$, it follows that

$$\forall \omega \in \Omega: \; Q(\omega^*, \{\theta^{*(m)}\}_{m=1}^M) \leq Q(\omega, \{\kappa_\omega(\mathcal{D}_{train}^{(m)})\}_{m=1}^M). \tag{3.23}$$

From the definition of κ_ω and additivity of $Q(\cdot)$, it also holds

$$Q(\omega^*, \{\kappa_{\omega^*}(\mathcal{D}_{train}^{(m)})\}_{m=1}^M) \leq Q(\omega^*, \{\theta^{*(m)}\}_{m=1}^M). \tag{3.24}$$

Combining these, we obtain

$$\forall \omega \in \Omega : Q(\omega^*, \{\kappa_{\omega^*}(\mathcal{D}_{train}^{(m)})\}_{m=1}^M) \leq Q(\omega, \{\kappa_\omega(\mathcal{D}_{train}^{(m)})\}_{m=1}^M), \tag{3.25}$$

which implies $\omega^* \in \Omega_B$.

3.A Supplementary Results

Source	Feature	Transformation
own	Volatility(lag = [1,2,3,4,5,6,7])	
own	Return(lag = [1,2,3,4,5,6,7])	
own	IsETF	
TTR	ADX	
TTR	aroon	
TTR	ATR(n=[7, 14, 28])	Norm.
TTR	BBands	Norm.
TTR	CCI	
TTR	chaikinAD	diff(1)
TTR	chaikinVolatility	
TTR	CLV	
TTR	CMF	
TTR	CMO	
TTR	CTI	
TTR	DEMA	Norm.
TTR	DonchianChannel	Norm.
TTR	EMA	Norm.
TTR	EVWMA	Norm.
TTR	GMMA(short=10, long=[30, 60])	Norm.
TTR	HMA	Norm.
TTR	KST	
TTR	MACD	
TTR	MFI	
TTR	OBV	diff(1)
TTR	PBands	Norm.
TTR	ROC	
TTR	RSI	
TTR	runPercentRank(n=100)	
TTR	SMI	
TTR	SNR(n=[20,60])	
TTR	TDI	Norm.
TTR	TRIX	
TTR	ultimateOscillator	
TTR	VHF	
TTR	volatility	
TTR	williamsAD	diff(1)
TTR	WPR	
TTR	ZLEMA	Norm.

Table 3.3: Features $x_t^{(m)}$ used as input to the model. The transformation “Norm.” indicates that the feature is normalized by the price of the asset while the transformation “diff(1)” denotes first differencing.

Bibliography

Akaike, H. (1998). Information Theory and an Extension of the Maximum Likelihood Principle. In Parzen, E., Tanabe, K., and Kitagawa, G., editors, *Selected Papers of Hirotugu Akaike*, Springer Series in Statistics, pages 199–213. Springer, New York, NY.

Alfelt, G., Bodnar, T., and Tyrcha, J. (2020). Goodness-of-fit tests for centralized Wishart processes. *Communications in Statistics-Theory and Methods*, 49(20):5060–5090.

Anatolyev, S. (2020). A ridge to homogeneity for linear models. *Journal of Statistical Computation and Simulation*, 90(13):2455–2472.

Anatolyev, S. and Kobotaev, N. (2018). Modeling and forecasting realized covariance matrices with accounting for leverage. *Econometric Reviews*, 37(2):114–139.

Andersen, T. G., Bollerslev, T., Diebold, F. X., and Labys, P. (2003). Modeling and forecasting realized volatility. *Econometrica*, 71(2):579–625.

Anderson, T. W. and Darling, D. A. (1952). Asymptotic Theory of Certain "Goodness of Fit" Criteria Based on Stochastic Processes. *The Annals of Mathematical Statistics*, 23(2):193–212.

Angrist, J. D. and Pischke, J.-S. (2009). *Mostly Harmless Econometrics: An Empiricist's Companion*. Princeton University Press.

Arlot, S. and Celisse, A. (2010). A survey of cross-validation procedures for model selection. *Statistics surveys*, 4:40–79.

Asai, M., Gupta, R., and McAleer, M. (2020). Forecasting volatility and co-volatility of crude oil and gold futures: Effects of leverage, jumps, spillovers, and geopolitical risks. *International Journal of Forecasting*, 36(3):933–948.

Bai, Y., Chen, M., Zhou, P., Zhao, T., Lee, J., Kakade, S., Wang, H., and Xiong, C. (2021). How Important is the Train-Validation Split in Meta-Learning? In *International Conference on Machine Learning*, pages 543–553. PMLR.

Bandara, K., Bergmeir, C., and Smyl, S. (2020). Forecasting across time series databases using recurrent neural networks on groups of similar series: A clustering approach. *Expert Systems with Applications*, 140:112896.

Barndorff-Nielsen, O. E. and Shephard, N. (2004). Econometric analysis of realized covariation: High frequency based covariance, regression, and correlation in financial economics. *Econometrica*, 72(3):885–925.

Bates, J. M. and Granger, C. W. J. (1969). The Combination of Forecasts. *Journal of the Operational Research Society*, 20(4):451–468.

Beck, J., Jackson, M. T., Vuorio, R., and Whiteson, S. (2023). Hypernetworks in Meta-Reinforcement Learning. In *Proceedings of The 6th Conference on Robot Learning*, pages 1478–1487. PMLR.

Bergmeir, C. and Benítez, J. M. (2012). On the use of cross-validation for time series predictor evaluation. *Information Sciences*, 191:192–213.

Bergmeir, C., Costantini, M., and Benítez, J. M. (2014). On the usefulness of cross-validation for directional forecast evaluation. *Computational Statistics & Data Analysis*, 76:132–143.

Bergmeir, C., Hyndman, R. J., and Koo, B. (2018). A note on the validity of cross-validation for evaluating autoregressive time series prediction. *Computational Statistics & Data Analysis*, 120:70–83.

Bollerslev, T. (1990). Modelling the coherence in short-run nominal exchange rates: A multivariate generalized ARCH model. *Review of Economics and Statistics*, 72(3):498.

Box, G. E. P. and Pierce, D. A. (1970). Distribution of Residual Autocorrelations in Autoregressive-Integrated Moving Average Time Series Models. *Journal of the American Statistical Association*, 65(332):1509–1526.

Burman, P., Chow, E., and Nolan, D. (1994). A cross-validatory method for dependent data. *Biometrika*, 81(2):351–358.

Callen, J. L., Kwan, C. C. Y., Yip, P. C. Y., and Yuan, Y. (1996). Neural network forecasting of quarterly accounting earnings. *International Journal of Forecasting*, 12(4):475–482.

Caporin, M. and McAleer, M. (2012). Do we really need both BEKK and DCC? A tale of two multivariate GARCH models. *Journal of Economic Surveys*, 26(4):736–751.

Caporin, M. and McAleer, M. (2014). Robust ranking of multivariate GARCH models by problem dimension. *Computational Statistics and Data Analysis*, 76:172–185.

Caporin, M. and Paruolo, P. (2015). Proximity-structured multivariate volatility models. *Econometric Reviews*, 34(5):559–593.

Cerqueira, V., Torgo, L., and Mozetič, I. (2020). Evaluating time series forecasting models: An empirical study on performance estimation methods. *Machine Learning*, 109(11):1997–2028.

Chaudhuri, S. and Renault, E. (2023). Efficient estimation of regression models with user-specified parametric model for heteroskedasticty. https://warwick.ac.uk/fac/soc/economics/research/workingpapers/2023/.

Chen, T., He, T., Benesty, M., Khotilovich, V., Tang, Y., Cho, H., Chen, K., Mitchell, R., Cano, I., Zhou, T., Li, M., Xie, J., Lin, M., Geng, Y., Li, Y., Yuan, J., and implementation), X. c. b. X. (2023). Xgboost: Extreme Gradient Boosting.

Chen, Y.-T. and Liu, C.-A. (2023). Model averaging for asymptotically optimal combined forecasts. *Journal of Econometrics*, 235(2):592–607.

Clark, T. and McCracken, M. (2013a). Advances in forecast evaluation. In *Handbook of Economic Forecasting*, volume 2, pages 1107–1201. Elsevier.

Clark, T. and McCracken, M. (2013b). Chapter 20 - Advances in Forecast Evaluation. In Elliott, G. and Timmermann, A., editors, *Handbook of Economic Forecasting*, volume 2 of *Handbook of Economic Forecasting*, pages 1107–1201. Elsevier.

Clark, T. E. and McCracken, M. W. (2001). Tests of equal forecast accuracy and encompassing for nested models. *Journal of Econometrics*, 105(1):85–110.

Clark, T. E. and McCracken, M. W. (2015). Nested forecast model comparisons: A new approach to testing equal accuracy. *Journal of Econometrics*, 186(1):160–177.

Cox, D. R. and Stuart, A. (1955). Some Quick Sign Tests for Trend in Location and Dispersion. *Biometrika*, 42(1/2):80–95.

DiCiccio, C. J., Romano, J. P., and Wolf, M. (2019). Improving weighted least squares inference. *Econometrics and Statistics*, 10:96–119.

Diebold, F. X. (2015). Comparing Predictive Accuracy, Twenty Years Later: A Personal Perspective on the Use and Abuse of Diebold–Mariano Tests. *Journal of Business & Economic Statistics*, 33(1):1–1.

Diebold, F. X. and Mariano, R. S. (1995). Comparing Predictive Accuracy. *Journal of Business & Economic Statistics*, 13(3).

Dietterich, T. G. (1998). Approximate Statistical Tests for Comparing Supervised Classification Learning Algorithms. *Neural Computation*, 10(7):1895–1923.

Engle, R. F. (2002). Dynamic conditional correlation: A simple class of multivariate generalized autoregressive conditional heteroskedasticity models. *Journal of Business & Economic Statistics*, 20(3):339–350.

Engle, R. F. and Kroner, K. F. (1995). Multivariate simultaneous generalized ARCH. *Econometric Theory*, 11(1):122–150.

Finn, C., Abbeel, P., and Levine, S. (2017). Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks. In *Proceedings of the 34th International Conference on Machine Learning*, pages 1126–1135. PMLR.

Flennerhag, S., Rusu, A. A., Pascanu, R., Visin, F., Yin, H., and Hadsell, R. (2020). Meta-Learning with Warped Gradient Descent.

Frazier, D. T., Covey, R., Martin, G. M., and Poskitt, D. (2023). Solving the Forecast Combination Puzzle.

Giacomini, R. and White, H. (2006). Tests of Conditional Predictive Ability. *Econometrica*, 74(6):1545–1578.

Godahewa, R., Bandara, K., Webb, G. I., Smyl, S., and Bergmeir, C. (2021). Ensembles of localised models for time series forecasting. *Knowledge-Based Systems*, 233:107518.

Golosnoy, V., Gribisch, B., and Liesenfeld, R. (2012). The conditional autoregressive Wishart model for multivariate stock market volatility. *Journal of Econometrics*, 167(1):211–223.

González-Coya, E. and Perron, P. (2024). Estimation in the Presence of Heteroskedasticity of Unknown Form: A Lasso-based Approach. *Journal of Econometric Methods*, 13(1):29–48.

Hadi, A. S. and Ling, R. F. (1998). Some Cautionary Notes on the Use of Principal Components Regression. *The American Statistician*, 52(1):15–19.

Hansen, P. R., Lunde, A., and Nason, J. M. (2011). The model confidence set. *Econometrica*, 79(2):453–497.

Hoerl, A. E. and Kennard, R. W. (1970). Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67.

Hospedales, T., Antoniou, A., Micaelli, P., and Storkey, A. (2021). Meta-learning in neural networks: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 44(9):5149–5169.

Hubinger, E., van Merwijk, C., Mikulik, V., Skalse, J., and Garrabrant, S. (2021). Risks from Learned Optimization in Advanced Machine Learning Systems.

Huisman, M., van Rijn, J. N., and Plaat, A. (2021). A survey of deep meta-learning. *Artificial Intelligence Review*, 54(6):4483–4541.

Hyndman, R., Montero-Manso, P., O'Hara-Wild, M., Talagala, T., Wang, E., Yang, Y., Taieb, S. B., Hanqing, C., Lake, D. K., Laptev, N., Moorman, J. R., and Zhang, B. (2023). Tsfeatures: Time Series Feature Extraction.

Hyndman, R. J. and Khandakar, Y. (2008a). Automatic Time Series Forecasting: The forecast Package for R. *Journal of Statistical Software*, 27(1):1–22.

Hyndman, R. J. and Khandakar, Y. (2008b). Automatic Time Series Forecasting: The forecast Package for R. *Journal of Statistical Software*, 27:1–22.

Hyndman, R. J., Koehler, A. B., Snyder, R. D., and Grose, S. (2002a). A state space framework for automatic forecasting using exponential smoothing methods. *International Journal of Forecasting*, 18(3):439–454.

Hyndman, R. J., Koehler, A. B., Snyder, R. D., and Grose, S. (2002b). A state space framework for automatic forecasting using exponential smoothing methods. *International Journal of Forecasting*, 18(3):439–454.

Ibragimov, R. and Müller, U. K. (2010). T-Statistic based correlation and heterogeneity robust inference. *Journal of Business & Economic Statistics*, 28(4):453–468.

Inoue, A. and Kilian, L. (2005). In-Sample or Out-of-Sample Tests of Predictability: Which One Should We Use? *Econometric Reviews*, 23(4):371–402.

Inoue, A. and Kilian, L. (2006). On the selection of forecasting models. *Journal of Econometrics*, 130(2):273–306.

Inoue, A., Rossi, B., and Jin, L. (2013). Consistent Model Selection: Over Rolling Windows. In Chen, X. and Swanson, N. R., editors, *Recent Advances and Future Directions in Causality, Prediction, and Specification Analysis: Essays in Honor of Halbert L. White Jr*, pages 299–330. Springer, New York, NY.

Izenman, A. J. (1975). Reduced-rank regression for the multivariate linear model. *Journal of Multivariate Analysis*, 5(2):248–264.

James, W. and Stein, C. (1961). Estimation with quadratic loss. *Proceedings of the Fourth Berkeley Symposium on Mathematical Statistics and Probability*, pages 443–460.

Januschowski, T., Gasthaus, J., Wang, Y., Salinas, D., Flunkert, V., Bohlke-Schneider, M., and Callot, L. (2020). Criteria for classifying forecasting methods. *International Journal of Forecasting*, 36(1):167–177.

Kim, J. (2014). Comparison of Lasso Type Estimators for High-Dimensional Data. *Communications for Statistical Applications and Methods*, 21(4):349–361.

Kullback, S. and Leibler, R. A. (1951). On Information and Sufficiency. *The Annals of Mathematical Statistics*, 22(1):79–86.

Laurent, S., Rombouts, J. V., and Violante, F. (2012). On the forecasting accuracy of multivariate GARCH models. *Journal of Applied Econometrics*, 27(6):934–955.

Lavancier, F. and Rochet, P. (2016). A general procedure to combine estimators. *Computational Statistics & Data Analysis*, 94:175–192.

Lazarus, E., Lewis, D. J., Stock, J. H., and Watson, M. W. (2018). HAR Inference: Recommendations for Practice. *Journal of Business & Economic Statistics*, 36(4):541–559.

Lee, Y. and Choi, S. (2018). Gradient-Based Meta-Learning with Learned Layerwise Metric and Subspace. In *Proceedings of the 35th International Conference on Machine Learning*, pages 2927–2936. PMLR.

Li, Z., Zhou, F., Chen, F., and Li, H. (2017). Meta-SGD: Learning to Learn Quickly for Few-Shot Learning.

Lin, Z., Zhao, Z., Zhang, Z., Baoxing, H., and Yuan, J. (2020). To Learn Effective Features: Understanding the Task-Specific Adaptation of MAML.

Ljung, G. M. and Box, G. E. P. (1978). On a measure of lack of fit in time series models. *Biometrika*, 65(2):297–303.

Lucheroni, C., Boland, J., and Ragno, C. (2019). Scenario generation and probabilistic forecasting analysis of spatio-temporal wind speed series with multivariate autoregressive volatility models. *Applied Energy*, 239:1226–1241.

Makridakis, S., Gaba, A., Hollyman, R., Petropoulos, F., Spiliotis, E., and Swanson, N. (2022). The M6 Financial Duathlon Competition Guidelines.

Makridakis, S., Spiliotis, E., and Assimakopoulos, V. (2020). The M4 Competition: 100,000 time series and 61 forecasting methods. *International Journal of Forecasting*, 36(1):54–74.

Makridakis, S., Spiliotis, E., Hollyman, R., Petropoulos, F., Swanson, N., and Gaba, A. (2023). The M6 forecasting competition: Bridging the gap between forecasting and investment decisions.

Malkiel, B. G. (2005). Reflections on the Efficient Market Hypothesis: 30 Years Later. *Financial Review*, 40(1):1–9.

McCracken, M. W. (2019). Tests of Conditional Predictive Ability: Some Simulation Evidence. SSRN Scholarly Paper ID 3368400, Social Science Research Network, Rochester, NY.

McCracken, M. W. (2020). Diverging Tests of Equal Predictive Ability. *Econometrica*, 88(4):1753–1754.

Montero-Manso, P., Athanasopoulos, G., Hyndman, R. J., and Talagala, T. S. (2020). FFORMA: Feature-based forecast model averaging. *International Journal of Forecasting*, 36(1):86–92.

Montero-Manso, P. and Hyndman, R. J. (2021). Principles and algorithms for forecasting groups of time series: Locality and globality. *International Journal of Forecasting*, 37(4):1632–1653.

Nava, E., Kobayashi, S., Yin, Y., Katzschmann, R. K., and Grewe, B. F. (2023). Meta-Learning via Classifier(-free) Diffusion Guidance.

Newey, W. K. and West, K. D. (1987). A Simple, Positive Semi-Definite, Heteroskedasticity and Autocorrelation Consistent Covariance Matrix. *Econometrica*, 55(3):703–708.

Noureldin, D., Shephard, N., and Sheppard, K. (2012). Multivariate high-frequency-based volatility (HEAVY) models. *Journal of Applied Econometrics*, 27(6):907–933.

Park, E. and Oliva, J. B. (2019). Meta-curvature. *Advances in Neural Information Processing Systems*, 32.

Patton, A. J. (2011). Volatility forecast comparison using imperfect volatility proxies. *Journal of Econometrics*, 160:246–256.

Pereira, J. M., Basto, M., and da Silva, A. F. (2016). The Logistic Lasso and Ridge Regression in Predicting Corporate Failure. *Procedia Economics and Finance*, 39:634–641.

Racine, J. (2000). Consistent cross-validatory model-selection for dependent data: Hv-block cross-validation. *Journal of Econometrics*, 99(1):39–61.

Radchenko, P., Vasnev, A. L., and Wang, W. (2023). Too similar to combine? On negative weights in forecast combination. *International Journal of Forecasting*, 39(1):18–38.

Raghu, A., Raghu, M., Bengio, S., and Vinyals, O. (2019). Rapid learning or feature reuse? towards understanding the effectiveness of maml. *arXiv preprint arXiv:1909.09157*.

Ramanarayanan, S., Palla, A., Ram, K., and Sivaprakasam, M. (2023). Generalizing supervised deep learning MRI reconstruction to multiple and unseen contrasts using meta-learning hypernetworks. *Applied Soft Computing*, 146:110633.

Schnaubelt, M. (2019). A comparison of machine learning model validation schemes for non-stationary time series data. Working Paper 11/2019, FAU Discussion Papers in Economics.

Shamsian, A., Navon, A., Fetaya, E., and Chechik, G. (2021). Personalized Federated Learning using Hypernetworks. In *Proceedings of the 38th International Conference on Machine Learning*, pages 9489–9502. PMLR.

Shao, J. (1993). Linear Model Selection by Cross-validation. *Journal of the American Statistical Association*, 88(422):486–494.

Sheppard, K. (2013). MFE Toolbox, https://www.kevinsheppard.com/code/matlab/mfe-toolbox/.

Smyl, S. (2020). A hybrid method of exponential smoothing and recurrent neural networks for time series forecasting. *International Journal of Forecasting*, 36(1):75–85.

Smyl, S. and Kuber, K. (2016). Data preprocessing and augmentation for multiple short time series forecasting with recurrent neural networks. In *36th International Symposium on Forecasting*.

Staněk, F. (2023a). A Note on the M6 Forecasting Competition: Designing Parametric Models with Hypernetworks.

Staněk, F. (2023b). A Note on the M6 Forecasting Competition: Rank Optimization.

Swanson, N. R. and White, H. (1997). Forecasting economic time series using flexible versus fixed specification and linear versus nonlinear econometric models. *International Journal of Forecasting*, 13(4):439–461.

Talagala, T. S., Hyndman, R. J., and Athanasopoulos, G. (2023). Meta-learning how to forecast time series. *Journal of Forecasting*, 42(6):1476–1501.

Tashman, L. J. (2000). Out-of-sample tests of forecasting accuracy: An analysis and review. *International Journal of Forecasting*, 16(4):437–450.

Thieu, L. Q. (2016). Variance targeting estimation of the BEKK-X model. https://mpra.ub.uni-muenchen.de/75572/.

Ulrich, J. (2021). TTR: Technical Trading Rules.

Usmani, R. A. (1994). Inversion of a tridiagonal Jacobi matrix. *Linear Algebra and its Applications*, 212(213):413–414.

von Oswald, J., Henning, C., Grewe, B. F., and Sacramento, J. (2022). Continual learning with hypernetworks.

Wang, H., Zhao, H., and Li, B. (2021). Bridging multi-task learning and meta-learning: Towards efficient training and effective adaptation. In *International Conference on Machine Learning*, pages 10991–11002. PMLR.

Wang, X., Hyndman, R. J., Li, F., and Kang, Y. (2023). Forecast combinations: An over 50-year review. *International Journal of Forecasting*, 39(4):1518–1547.

Wei, C. Z. (1992). On Predictive Least Squares Principles. *The Annals of Statistics*, 20(1):1–42.

West, K. D. (1996). Asymptotic Inference about Predictive Ability. *Econometrica*, 64(5):1067–1084.

West, K. D. (2006). Chapter 3 Forecast Evaluation. In Elliott, G., Granger, C. W. J., and Timmermann, A., editors, *Handbook of Economic Forecasting*, volume 1, pages 99–134. Elsevier.

Winkler, R. L. and Makridakis, S. (1983). The Combination of Forecasts. *Journal of the Royal Statistical Society. Series A (General)*, 146(2):150.

Zhang, Y. and Yang, Q. (2022). A Survey on Multi-Task Learning. *IEEE Transactions on Knowledge and Data Engineering*, 34(12):5586–5609.

Zhang, Y. and Yang, Y. (2015). Cross-validation for selecting a model selection procedure. *Journal of Econometrics*, 187(1):95–112.

Zhao, D., Kobayashi, S., Sacramento, J., and von Oswald, J. (2020). Meta-Learning via Hypernetworks. In *Zhao, Dominic; Kobayashi, Seijin; Sacramento, João; von Oswald, Johannes (2020). Meta-Learning via Hypernetworks. In: 4th Workshop on Meta-Learning at NeurIPS 2020 (MetaLearn 2020), Virtual Conference, 11 December 2020, IEEE.*, Virtual Conference. IEEE.

Zhipeng, Y. and Shenghong, L. (2018). Hedge ratio on Markov regime-switching diagonal Bekk–Garch model. *Finance Research Letters*, 24:49–55.

Zhu, Y. and Timmermann, A. (2020). Can Two Forecasts Have the Same Conditional Expected Accuracy? *arXiv:2006.03238 [stat]*.

Zintgraf, L., Shiarli, K., Kurin, V., Hofmann, K., and Whiteson, S. (2019). Fast context adaptation via meta-learning. In *International Conference on Machine Learning*, pages 7693–7702. PMLR.

Zolfaghari, M., Ghoddusi, H., and Faghihian, F. (2020). Volatility spillovers for energy prices: A diagonal BEKK approach. *Energy Economics*, 92:104965.