Inference in Predictive Regression Models with Persistent Regressors

Inaugural-Dissertation
zur Erlangung des akademischen Grades eines Doktors
der Wirtschafts- und Sozialwissenschaften
der Wirtschafts- und Sozialwissenschaftlichen Fakultät
der Christian-Albrechts-Universität zu Kiel

vorgelegt von
Benjamin Hillmann, M.Sc.
aus Hamburg

Kiel, 2021

Erstbegutachtung: Prof. Dr. Matei Demetrescu
Zweitbegutachtung: Prof. Dr. Kai Carstensen

Tag der mündlichen Prüfung: 23. September 2021

Acknowledgments

I wrote this dissertation while being employed at the Institute for Statistics and Econometrics at Kiel University. I would like to express my gratitude and appreciation to my thesis supervisor Prof. Dr. Matei Demetrescu for continuous support while writing this thesis.

In addition, I would like to thank my second supervisor Prof. Dr. Kai Carstensen helpful comments and Prof. Dr. Uwe Jensen for guiding me during the entirety of my studies.

I am also very grateful to Dr. Anna Titova, Dr. Jan Roestel and Angelika Reinmüller for collegial, statistical and organizational advices.

My thesis would not have been possible without the support of Julian Schröder and Albrecht Mengel from the IT department. Special thanks go out to number 156 for working day and night on my computations.

I am very thankful to my family for their encouragement and loving care.

Finally, I could not have completed this dissertation without the emotional, professional, caring and loving support of Nathalie Stelzer.

Contents

List of Abbreviations

ADA adaptive test

AR(p) autoregression of order p

BM book to market ratio

CAPM capital asset pricing model

COMB combination test

DCFSE differences in cumulated squared forecast errors

DFY default yield spread

DGP data generating process

DY dividend yield

GARCH generalized autoregressive conditional heteroskedasticity

INFL inflation

IS in-sample

IV instrumental variable

IVX extended instrumental variable

logDP log dividend price ratio

logEP log earnings price ratio

LTR long term rate of returns

LTY long term yield

MC monte carlo

MSE mean squared error

OLS ordinary least squares

OOS out-of-sample

RSS residual sum of square

SVAR stock variance

TBL treasury-bill rates

TBM tree based method

TMS term spread

VAR(m) vector autoregression of order m

w.r.t. with respect to

List of Tables

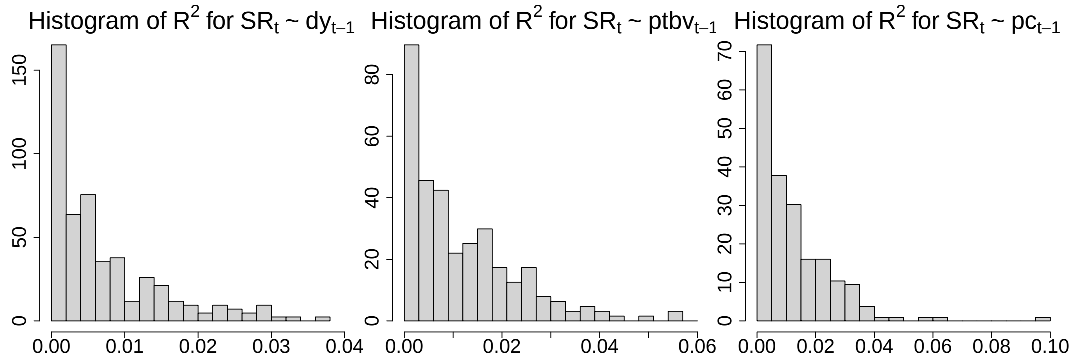
List of Figures

Chapter 1

Introduction

Stock return predictions and tests for predictability have received increasing attention in recent years. However, despite a well-developed theory, predictability of stock returns using fundamental variables is difficult to establish in practice. This dissertation presents three essays regarding predictive regressions and describes how to cope with potential pitfalls. One typical issue of predictive regressions is that the parameters to be estimated are often close to zero and the signal-to-noise ratio is fairly low (c.f. Campbell (2008) and Phillips (2015)). Thus, predictive regressions often exhibit a low R^2. The distributions of resulting R^2 values for 212 univariate predictive regressions with three different predictor variables are illustrated in Figure 1.1. As expected, the R^2 values are in fact quite low in all three cases. Data for this example is taken from Chapter 4 of this dissertation.

Figure 1.1: Weak Signals - Distributions of R^2



Notes: Distribution of R^2 for 212 firm-specific univariate least squares estimations with stock returns as dependent variable to illustrate weak signals for predictions. Regressors from left to right: lagged dividend yield, price to book ratio and price to cash flow ratio. Data for these estimations is taken from Chapter 4.4.3.

Tests for predictability need to detect local deviations from the null, since parameter values in predictive regressions are close to the null. Thus, these tests need a reasonable local power (i.e. local in the neighborhood of the null).

We are going to examine the local power for various tests for predictability in Monte Carlo simulations in Chapter 2 and 3.

Furthermore, all three papers present different approaches for dealing with predictive regressions when facing persistent as well as endogenous regressor variables. We illustrate this issue using the following predictive regression model (c.f. Stambaugh (1999)):

$$y_t = \alpha + \beta x_{t-1} + u_t \tag{1.1}$$

$$x_t = \theta + \rho x_{t-1} + v_t \tag{1.2}$$

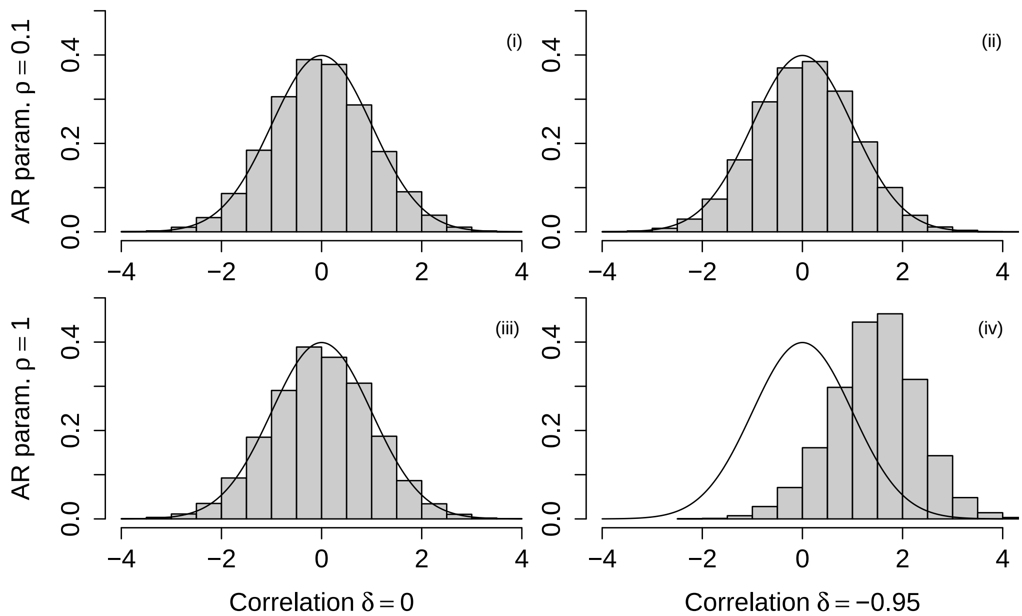
with y_t being the variable of interest, e.g. stock returns, and x_{t-1} a lagged predictor variable, e.g. the lagged log dividend price ratio. We additionally assume:

$$\begin{pmatrix} u_t \\ v_t \end{pmatrix} \overset{iid}{\sim} N\left(0, \begin{pmatrix} 1 & \delta \\ \delta & 1 \end{pmatrix}\right). \tag{1.3}$$

The regressor is endogenous for correlated innovation terms, i.e. $\delta \neq 0$. When applying least squares estimation and testing the null hypothesis of no predictability ($H_0 : \beta = 0$), the test statistic does not depend on α, β and θ but on ρ and δ.

We perform Monte Carlo simulations under the null of $\beta = 0$ with $\alpha, \theta = 0$, $\rho \in \{0.1, 1\}$ and $\delta \in \{0, -0.95\}$ with sample size $T = 240$ and 10000 Monte Carlo replications. The distributions of the corresponding test statistics for the four arising cases are illustrated in panels (i) - (iv) of Figure 1.2.

Figure 1.2: Distribution of OLS test statistics for different parameter values of ρ and δ



The regressor variables exhibit low degrees of persistence in both top panels ($\rho = 0.1$), while they are highly persistent in the lower ones ($\rho = 1$). They are endogenous in the right panels ($\delta = -0.95$) but not in the left panels ($\delta = 0$). Panels (i) to (iii) depict (close to) standard normal distributions of the test statistics. Two-sided tests on a significance level of $\alpha = 0.05$ yield rejection rates of 0.053, 0.050 and 0.053, respectively. However, the distribution of the test statistic in panel (iv) is not standard normal. It yields a two-sided rejection rate of 0.278 (right-sided 0.422), i.e. the null of no predictability is rejected too often when dealing with a persistent and endogenous regressor.

Least squares estimation performs well in the first three cases. However, for persistent and endogenous regressor variables, the least squares estimator for β in Equation (1.1) suffers from second-order bias. Also, its test statistic has a nonnormal limiting distribution characterised by ρ and δ. This case is not only relevant in theory but also for practitioners, e.g. when predicting or testing for predictability of stock returns.

We discuss three different approaches to deal with the issue of persistent and endogenous regressors as well as further pitfalls such as nonlinearity and model uncertainty in the remainder of this thesis.

Essay 1 is joint work with Matei Demetrescu and entitled "Gaussian Inference in Predictive Regressions for Stock Returns". We take the test statistic of an OLS estimator of Equation (1.1) with its nonnormal limiting distribution for persistent and endogenous regressors. We then exploit the fact that the null distribution of this test statistic can be rewritten as a weighted sum of a normal and a nonstandard part. Limiting distributions of corresponding test statistics for M estimations have similar properties in these predictive regressions. Furthermore, the distributions of the test statistics for different M estimations can each be written as weighted sums of a normal and a nonstandard part, whereby the nonstandard parts are identical. We exploit this by combining the test statistics of two different M estimations to cancel out the nonstandard part. Thus, we obtain a standard normal test statistic for the null of no predictability. We deduce suitable combinations of different M-based test statistics and test for predictability of US stock returns. Critical values are derived from a fixed-regressor bootstrap to account for multiple testing in rolling windows.

Essay 2 is entitled "Nonlinear Predictability of Stock Returns? Parametric Versus Nonparametric Inference in Predictive Regressions". It is joint work with Matei Demetrescu and is published in the *Journal of Business and Economic Statistics*. We argue that the relation between stock returns and predictor variables might not be linear. Therefore, we extend the previous linear model (1.1) to an additive nonlinear one:

$$y_t = \alpha + \beta f(x_{t-1}) + u_t. \tag{1.4}$$

There are basically two options for testing the predictive power of a potentially nonlinear model. One could either use a nonparametric test or apply a test based on a linear model, since Equation (1.4) is linear under the null. However, with a nonparametric test one would lose power due to leaving the parametric framework leading to lower convergence rates. On the other hand, one would also lose power when applying a test based on a linear model due to a misspecification under the alternative. We argue that the loss in power for misspecified linear models is lower than for nonparametric tests when deviations from the null are small, i.e. when β is close to zero (as in the case of stock return predictions). This case of a local deviation from the null is of particular interest, since predictive regressions (for stock returns) exhibit weak signals, i.e. small coefficients.

We employ the IVX test by Kostakis et al. (2015) and an additional two-stage least squares test including the IVX instrument as (misspecified) parametric approaches. In addition, we employ the nonparametric tests by Juhl (2014) and Kasparis et al. (2015). All four tests possess standard limiting distributions irrespective of degree of persistence and endogeneity of the regressors.

We then compare these tests in Monte Carlo simulations with respect to their local power to mimic the presence of weak signals. Afterwards, they compete in an empirical application for testing the predictability of S&P 500 stock returns. Moreover, we employ the different parametric and nonparametric tests as pretests when constructing out-of-sample stock return predictions. Based on the pretest, predictions are constructed by employing either the historic mean or a (non)linear forecasting procedure. We apply a linear model and a generalized additive model with smoothing splines in the linear and nonlinear forecasting procedures, respectively. We find best results for combinations of linear pretests with nonlinear forecasting procedures. However, in order to obtain competitive results selection of regressor variables is essential. This procedure outperforms historic mean predictions (w.r.t RSS) for the regressor logEP but not for logDP.

Finally, we expand the different tests to multivariate versions and estimate the stock return predictions. Though, results are not as promising as in the univariate case.

The third essay is titled “Predicting Stock Returns with Regression Trees: Nonlinearities and Predictor Selection”. The first two essays mainly focus on testing for (stock return) predictability, while this paper discusses (stock return) predictions from different tree based methods and assesses their accuracy. We pay special attention to the performance of local linear tree based methods.

Trees are constructed by recursively partitioning the data space and performing predictions for each emerging region. We either employ regional means of the dependent variable or perform linear regressions within each region to obtain prediction values. Thus, tree based methods are more flexible than parametric approaches, since they perform separate predictions for each region.

We estimate univariate as well as multivariate predictions for S&P 500 stock returns applying the same regressor variables as in the second essay. Trees have the advantage of a built-in parameter selection. Only the most informative regressor variables in multivariate tasks are employed to grow regression trees and perform predictions. The importance of parameter selection becomes eminent from the previous essay. In the last section of that essay we find that a “kitchen sink” regression, employing all available regressor variables, performs worse than any univariate regression. However, regression trees should perform better in the presence of multiple predictor variables, as they have a built-in regressor selector. Including all available and potentially uninformative regressor variables should not harm predictions from tree based methods as much, since regression trees perform a predictor selection. Region averages as predictions are not affected by all regressor variables, only by those which are responsible for defining the corresponding region. Nevertheless, we also perform stock return predictions employing different subsets of all available regressor variables for comparison.

Chapter 2

Gaussian Inference in Predictive Regressions for Stock Returns

Coauthored by: Matei Demetrescu

2.1 Introduction

Predictive regressions play an important role in applied work. They are for instance widely used to assess whether future stock returns can be predicted by current information. Among others, predictive regressions have been used to study the performance of mutual funds, to test the (conditional) CAPM, and in studies of optimal asset allocation; see Paye and Timmermann (2006, pp. 274-275) and the references therein. Based on economic theory, various putative predictors have been considered for predictive regressions, for instance financial ratios like the dividend yield, default premia, or the term structure of interest rates; e.g. Welch and Goyal (2008) evaluate the predictive performance of 11 different variables.

From a technical perspective, inference in such predictive regressions is however challenging. Concretely, any inferential procedure needs to take into account the stylized features of the data, which, for predictive regressions for stock returns, include the uncertain persistence of the putative predictors and the contemporaneous correlation of the predictors' innovations with those of the stock returns.

For this reason, we consider here the popular predictive regression model

$$y_t = \delta + \beta x_{t-1} + u_t, \qquad t = 2, \dots, T, \tag{2.1}$$

where the disturbances u_t are taken to be unpredictable, such that predictability of y_t translates into $\beta \neq 0$. The regressor is taken to have an autoregressive structure,

$$x_t = \mu_x + \xi_t \quad \text{and} \quad \xi_t = \rho \xi_{t-1} + v_t \tag{2.2}$$

with ξ_1 bounded in probability. Various values of ρ then capture various degrees of persistence of the putative predictor x_t.

We do not impose a particular type of persistence for x_t; rather, we let the regressor be either stationary (and thus of low persistence), i.e. ρ is fixed and bounded away from unity in absolute value, or near integrated (and thus of high persistence), $\rho = \rho_T = 1 - c/T$. Both modelling approaches have been considered in the literature; see among others Amihud and Hurvich (2004) and Campbell and Yogo (2006). By allowing ρ to be either fixed and away from unity, or near to unity, we cover a wide range of degrees of persistence for the predictor x_t. Importantly, our results are valid under both low and high persistence, and are thus robust to the persistence present in the data, which is typically plagued by uncertainty.

Under low persistence as defined above, inference is standard with asymptotically normal slope coefficient estimator and standard normal null distributions of the associated t statistic. High persistence in the form of near-integrated variables has, on the other hand, direct implications for estimation and inference whenever u_t and v_t correlate contemporaneously; see Stambaugh (1999) and Elliott and Stock (1994). For instance, the OLS estimator of β in (2.1) exhibits second-order bias and the corresponding t-statistic has a nonnormal limiting distribution characterized by c and the correlation between u_t and v_t. In specific, regularity conditions assumed, the limiting null distribution of the usual OLS-based t statistic is given by

$$t_{ls} \xrightarrow{d} \frac{\int_0^1 \left(J_c(s) - \bar{J}_c\right) \mathrm{d}U(s)}{\sqrt{\int_0^1 \left(J_c(s) - \bar{J}_c\right)^2 \mathrm{d}s}},$$

where $\bar{J}_c = \int_0^1 J_c(s)\mathrm{d}s$, $J_c(s) = V(s) - c\int_0^s e^{-c(s-r)}V(r)\mathrm{d}r$ is an Ornstein-Uhlenbeck process, and $(U(s), V(s))'$ is a bivariate Brownian motion with unit-variance elements, possibly correlated; see Elliott and Stock (1994). (Formally, the low-persistence case may be obtained as a sequential limit with $T \to \infty$ followed by $c \to 0$.)

Since the parameter c cannot be consistently estimated (Phillips, 1987), it is not known which critical values should be used for testing hypotheses on β with the OLS t-statistic.[1] A number of solutions to this problem have been proposed; see e.g. the bias-correction approach of Amihud and Hurvich (2004), the Bonferroni procedure of Campbell and Yogo (2006), or more recently the near-optimal tests of Elliott et al. (2015), the extended IV approach of Kostakis et al. (2015) and the variable addition and IV based inference of Breitung and Demetrescu (2015). All these methods refer in one way or another to predictability in the conditional mean.

A number of contributions have however addressed the issue of predictability from a different perspective. For instance, Cenesizoglu and Timmermann (2008) discuss predictability of stock returns at various quantiles (see also Lee, 2016; Meligkotsidou et al., 2014, 2019), while Demetrescu and Roling (2021) test for predictability under a general loss function when examining the forward premium puzzle. These approaches may be cast (as is also the case with OLS) in an M estimation and testing framework ; see Huber (1981). Interestingly, the limiting distributions of the corresponding t statistics in the case of M estimation have a similar structure. In fact, Lee (2016) and Demetrescu and Roling (2021) obtain the same expression for the limiting

[1] A conservative approach using critical values for the worst-case scenario (typically $c = 0$) leads to large power losses; see e.g. Campbell and Yogo (2006).

distribution of such M based t statistics,

$$t \xrightarrow{d} \frac{\int_0^1 \left(J_c(s) - \bar{J}_c\right) \mathrm{d}\tilde{U}(s)}{\sqrt{\int_0^1 \left(J_c(s) - \bar{J}_c\right)^2 \mathrm{d}s}}, \tag{2.3}$$

where $\tilde{U}(s)$ is the limit process of the partial sums of a *transformation* of u_t specific to the particular M estimator employed.[2]

Here, we exploit the fact that the limiting null distribution in (2.3) can be expressed as a weighted sum of two components, one of which is nonstandard and depends on $V(s)$, but not on $\tilde{U}(s)$, whereas the second is Gaussian, and provide an asymptotically Gaussian test procedure for the null of no predictability in predictive regressions with possibly near-integrated regressors. To this end, we construct in Section 2.2 a weighted average of two different M tests of the null of no predictability, $\beta = 0$, where the weights are picked in such a way that the nonstandard component cancels out and one works with the Gaussian components only. We address the issue of power, and explore the finite-sample properties of the proposed test in Section 2.3. Section 2.4 re-examines US stock return predictability. Section 2.5 concludes, and technical proofs have been gathered in the appendix, which also contains supplementary simulation results.

In terms of notation, we denote by $\Rightarrow$ weak convergence on the space of càdlàg functions defined on $[0, 1]$. For weak convergence on the real line we use equivalently $\xrightarrow{d}$, while $\xrightarrow{p}$ signifies convergence in probability.

2.2 Aggregating M-based tests

For the predictive regression model (2.1), we consider the following class of M estimators,

$$\hat{\delta}, \hat{\beta} = \underset{\delta^*, \beta^*}{\arg\min} \sum_{t=2}^{T} \mathcal{L}\left(y_t - \delta^* - \beta^* x_{t-1}\right), \tag{2.4}$$

where $\mathcal{L}$ belongs to a suitable family of quasi-convex functions. We focus on asymmetric quadratic loss functions,

Assumption 2.1 *The loss function $\mathcal{L}(u) \mapsto \mathbb{R}^+$ is given by*

$$\mathcal{L}(u) = \left((1 - 2\alpha)\mathbf{1}(u < 0) + \alpha\right) |u|^2$$

where $\alpha \in (0, 1)$.

Asymmetric quadratic losses are a convenient choice here. While the more general class of asymmetric power loss functions proposed by Elliott et al. (2005) allows e.g. for asymmetric

[2] For quantile regression, the transformation is the generalized sign function, while, for estimation under a general loss function, the transformation is the derivative of the loss function; see Section 2 for details. For OLS, the transformation is then simply the identity function.

linear or asymmetric cubic losses, asymmetric linear losses (in effect quantile regression) is computationally more demanding,[3] while e.g. asymmetric cubic losses would require stricter moment conditions on the regression errors u_t; see Demetrescu and Roling (2021).

The corresponding t-type statistic of no predictability, $t = \hat{\beta}/s.e.(\hat{\beta})$, is computed with Huber-Eicker standard errors,

$$s.e.\left(\hat{\beta}\right) = \sqrt{\left[B_T^{-1} M_T B_T^{-1}\right]_{2,2}},$$

where

$$\begin{aligned} B_T &= \begin{pmatrix} \sum_{t=2}^{T} \mathcal{L}''\left(\hat{u}_t\right) & \sum_{t=2}^{T} x_{t-1}\mathcal{L}''\left(\hat{u}_t\right) \\ \sum_{t=2}^{T} x_{t-1}\mathcal{L}''\left(\hat{u}_t\right) & \sum_{t=2}^{T} x_{t-1}^2\mathcal{L}''\left(\hat{u}_t\right) \end{pmatrix} \\ M_T &= \begin{pmatrix} \sum_{t=2}^{T} \left(\mathcal{L}'\left(\hat{u}_t\right)\right)^2 & \sum_{t=2}^{T} x_{t-1}\left(\mathcal{L}'\left(\hat{u}_t\right)\right)^2 \\ \sum_{t=2}^{T} x_{t-1}\left(\mathcal{L}'\left(\hat{u}_t\right)\right)^2 & \sum_{t=2}^{T} x_{t-1}^2\left(\mathcal{L}'\left(\hat{u}_t\right)\right)^2 \end{pmatrix} \end{aligned}$$

and $\hat{u}_t = y_t - \hat{\delta} - \hat{\beta}x_{t-1}$.

To rigorously introduce our new test, we make the follwing assumptions about the data generating process.

Assumption 2.2 *The autoregressive coefficient ρ is either fixed and bounded away from unity, $|\rho| < 1$, or local to unity, $\rho = 1 - c/T$.*

Assumption 2.3 *Let $v_t = \sum_{j\geq 0} b_j \nu_{t-j}$ where $\sum_{j\geq 0} j\,|b_j| < \infty$, $\lambda = \sum_{j\geq 0} b_j \neq 0$, and $(u_t, \nu_t)'$ is a serially independent, identically distributed series with finite kurtosis and no atoms, where ν_t has zero mean.*

The zero mean condition on ν_t ensures that the stochastic component of the predictor, ξ_t, has zero mean and no drift, and any deterministic components of x_t are captured by μ_x. We impose this because deterministic trend components are not plausible for the typical predictor say in regressions of stock returns on dividend yields. The serial independence assumption is somewhat restrictive given that y_t usually stand for stock returns in this framework, but we note that the predictive regressions literature using financial valuation ratios as putative predictors tends to use low-frequency data such as monthly, quarterly and even yearly returns, so GARCH effects should not be of serious concern.

For later reference, we define

$$\delta_{\mathcal{L}} = \arg\min_{\delta^*} \mathrm{E}\left(\mathcal{L}\left(u_t - \delta^*\right)\right) \quad \text{and} \quad \tilde{u}_t = \mathcal{L}'\left(u_t - \delta_{\mathcal{L}}\right).$$

Note that $\tilde{u}_t$ are so-called generalized forecast errors (see e.g. Granger, 1999) and, under Assumptions 2.1 and 2.3, can be shown to satisfy

$$\mathrm{E}\left(\tilde{u}_t | x_{t-1}, x_{t-2}, \ldots\right) = 0,$$

[3] Newey and Powell (1987) provide for instance an iteratively reweighted least squares algorithm to solve the minimum problem in (2.4).

such that the disturbances are not predictable under $\mathcal{L}$.

Importantly, for different loss functions (i.e. different α under Assumption 2.1), one has different generalized forecast errors $\tilde{u}_t$, and, correspondingly, different limit processes $\tilde{U}(s)$ for the partial sums of $\tilde{u}_t$ as follows.

Under our assumptions, the sequence $(\tilde{u}_t, \nu_t)'$ is zero-mean iid with finite variance, and we have the weak convergence

$$\frac{1}{\sqrt{T}} \sum_{t=2}^{[sT]} \begin{pmatrix} \tilde{u}_t \\ v_t \end{pmatrix} \Rightarrow \begin{pmatrix} \sigma_{\tilde{u}} \tilde{U}(s) \\ \lambda \sigma_\nu V(s) \end{pmatrix}$$

jointly with $T^{-1/2}\xi_{[sT]} \Rightarrow \lambda\sigma_\nu J_c(s)$ where, recall, $J_c(s) = V(s) - c\int_0^s e^{-c(s-r)} V(r)\,\mathrm{d}r$ is an Ornstein-Uhlenbeck process, and the covariance matrix of $(\tilde{u}_t, \nu_t)'$ is

$$\mathbf{\Sigma} = \begin{pmatrix} \sigma_{\tilde{u}}^2 & \sigma_{\tilde{u}}\gamma \\ \sigma_{\tilde{u}}\sigma_\nu\gamma & \sigma_\nu^2 \end{pmatrix},$$

and γ gives the correlation of $\tilde{u}_t = \mathcal{L}'(u_t - \delta_{\mathcal{L}})$ and ν_t. The correlation γ depends on the choice α pinning down the loss function.

Define now the projection

$$\tilde{W}(s) = \frac{1}{\sqrt{1-\gamma^2}} \left(\tilde{U}(s) - \gamma V(s) \right),$$

and note that $\tilde{W}$ is independent of V (since the processes are jointly Gaussian and orthogonal by construction). Therefore, $\tilde{W}$ is independent of J_c as well. We then have the following representation for the limiting null distribution (2.3) of the M based t statistic in the high persistence case,

$$\frac{\int_0^1 \left(J_c(s) - \bar{J}_c\right) \mathrm{d}\tilde{U}(s)}{\sqrt{\int_0^1 \left(J_c(s) - \bar{J}_c\right)^2 \mathrm{d}s}} \equiv \gamma \frac{\int_0^1 \left(J_c(s) - \bar{J}_c\right) \mathrm{d}V(s)}{\sqrt{\int_0^1 \left(J_c(s) - \bar{J}_c\right)^2 \mathrm{d}s}} + \sqrt{1-\gamma^2} \frac{\int_0^1 \left(J_c(s) - \bar{J}_c\right) \mathrm{d}\tilde{W}(s)}{\sqrt{\int_0^1 \left(J_c(s) - \bar{J}_c\right)^2 \mathrm{d}s}}, \tag{2.5}$$

where the independence of $\tilde{W}$ and J_c implies the numerator of the second summand on the r.h.s. to be mixed Gaussian, such that the second summand is Gaussian and independent of V – and thus of the first summand on the r.h.s. of (2.5).

The decomposition shows how the limiting distribution depends on the two parameters of relevance, the strength of mean-reversion c and the correlation γ of the two involved Wiener processes V and $\tilde{U}$. Note that the correlation γ could be estimated from residuals $\hat{\nu}_t$ and $\hat{u}_t$, in contrast to the mean-reversion parameter c.

The key observation enabling our proposal is that, for *different* loss functions $\mathcal{L}$, implying *different* processes $\tilde{W}(s)$ and *different* γ, the resulting M estimators and tests need not be perfectly correlated, although the nonstandard component of their distribution is *the same.* Hence a suitable linear combination of two *different* M-based statistics will be Gaussian, irrespective of c. Contrary to the mean reversion parameter c, the involved weights can then be estimated consistently so a feasible version is at hand. Closest to this idea is the proposal of Lucas (1995) and Herce (1996) to combine Dickey-Fuller and M-estimation based unit rot tests to obtain a standard normal unit root test. However, we're primarily interested in pivotal inference, i.e. in removing the dependence on c of the limiting null distribution of the relevant test statistics; Gaussianity is here just a bonus.

So consider now two M estimators of β, $\hat{\beta}_1$ and $\hat{\beta}_2$, resulting from two different loss functions $\mathcal{L}_1$ and $\mathcal{L}_2$ (i.e. two different values for the asymmetry parameter, α_1 and α_2) and let t_1 and t_2 be the resulting t-type statistics. Then, the linear combination

$$\gamma_2 t_1 - \gamma_1 t_2$$

would be Gaussian since the nonstandard component cancels out. At the same time, this linear combination of t ratios, while following a Gaussian limiting distribution, is not standardized. Concretely, the correlation ω of the Gaussian terms in (2.5) needs to be taken into account. The correlation ω is easily seen to equal the correlation of $\tilde{W}_1$ and $\tilde{W}_2$, and therefore that of $\tilde{u}_{t1} - (\gamma_1 \sigma_{\tilde{u}_1}/\sigma_\nu)\,\nu_t$ and $\tilde{u}_{t2} - (\gamma_2 \sigma_{\tilde{u}_2}/\sigma_\nu)\,\nu_t$. Also, these unknown parameters need to be estimated to obtain a feasible test statistic.

Summing up, the proposed test statistic is given by

$$\bar{t} = \frac{\hat{\gamma}_2 t_1 - \hat{\gamma}_1 t_2}{\sqrt{\hat{\gamma}_2^2\left(1-\hat{\gamma}_1^2\right) + \hat{\gamma}_1^2\left(1-\hat{\gamma}_2^2\right) - 2\hat{\omega}\hat{\gamma}_1\hat{\gamma}_2\sqrt{\left(1-\hat{\gamma}_1^2\right)\left(1-\hat{\gamma}_2^2\right)}}}, \tag{2.6}$$

where, for $i = 1, 2$,

$$\hat{\gamma}_i = \frac{1}{\hat{\sigma}_{\tilde{u}_i}\hat{\sigma}_\nu T}\sum_{t=2}^{T}\hat{\nu}_t \mathcal{L}_i'\left(\hat{u}_{ti}\right) \quad \text{and} \quad \hat{\sigma}_{\tilde{u}_i} = \frac{1}{T}\sum_{t=2}^{T}\left(\mathcal{L}_i'\left(\hat{u}_{ti}\right)\right)^2$$

with $\hat{\nu}_t$ being the residuals of a so-called long autoregression of x_t (i.e. of order going to infinity as $T \to \infty$), and

$$\hat{\omega} = \frac{\sum_{t=2}^{T}\hat{w}_{t1}\hat{w}_{t2}}{\sqrt{\sum_{t=2}^{T}\hat{w}_{t1}^2\sum_{t=2}^{T}\hat{w}_{t2}^2}}$$

with $\hat{w}_{ti}$ residuals from a regression of $\mathcal{L}_i'\left(\hat{u}_{ti}\right)$ on $\hat{\nu}_t$ for $i = 1, 2$. The residuals

$$\hat{u}_{ti} = y_t - \hat{\delta}_i - \hat{\beta}_i x_{t-1}$$

come from the respective M estimation of (2.1), and the order of the long autoregression may be picked using the usual model selection tools.[4]

Some power reduction may be expected compared to the OLS estimator. In order to assess the price paid for Gaussianity, we examine the limiting behavior of $\bar{t}$ under a sequence of local alternatives of the form $\beta_1 = {}^b/_T$. The following result shows that the test has nontrivial power

[4] We resort to the Akaike IC in the following, but any asymptotically conservative order selection procedure could be employed.

against such sequences of local alternatives.

Proposition 2.1 *Under local alternatives of the form $\beta = \beta_T = {}^b/_T$ and high persistence, we have under Assumptions 2.1 – 2.3 that*

$$\bar{t} \xrightarrow{d} \mathcal{Z} + b\zeta\sqrt{\int_0^1 \left(J_c(s) - \int_0^1 J_c(s)\,\mathrm{d}s\right)^2 \mathrm{d}s}$$

as $T \to \infty$, where $\mathcal{Z}$ is a standard normal variate independent of $J_c(s)$ and

$$\zeta = \frac{\sigma_v\left(\frac{\kappa_1\gamma_2}{\sigma_{\tilde{u}_1}} - \frac{\kappa_2\gamma_1}{\sigma_{\tilde{u}_2}}\right)}{\sqrt{\gamma_2^2\left(1-\gamma_1^2\right) + \gamma_1^2\left(1-\gamma_2^2\right) - 2\omega\gamma_1\gamma_2\sqrt{\left(1-\gamma_1^2\right)\left(1-\gamma_2^2\right)}}} \tag{2.7}$$

with $\kappa = \mathrm{E}\left(\mathcal{L}''\left(u_t - \delta_{\mathcal{L}}\right)\right)$.

Proof: *See the Appendix.*

We notice that the IVX approach of Kostakis et al. (2015) implies some loss of asymptotic power, as it only has nontrivial power against local alternatives of the form $b/T^{1/2+\eta/2}$ for some $\eta \in (0,1)$ under high persistence. In finite samples, however, the rate advantage may not suffice to detect alternatives with small but nonzero coefficients β, so one would choose the two loss functions such that the coefficient ζ is as large as possible; see Section 2.3 for details.

In the stationary case, we obtain the following

Proposition 2.2 *Under local alternatives of the form $\beta = \beta_T = {}^b/\sqrt{T}$ and low persistence, we have under Assumptions 2.1 – 2.3 that*

$$\bar{t} \xrightarrow{d} \mathcal{Z} + b\zeta\sqrt{\mathrm{Var}\left(x_t\right)}$$

as $T \to \infty$, where $\mathcal{Z}$ is a standard normal variate and ζ is defined in Proposition 2.1.

Proof: *Follows with standard arguments and we omit the details.*

Therefore, the limiting null distribution of $\bar{t}$ is the standard normal under both types of persistence, without requiring any user input as to which type of persistence is relevant for the concrete data set. Moreover, the power against sequences of local alternatives depends apart from the localization parameter b, on the same quantity ζ specific to our procedure.

Although the bulk of the predictive regressions literature focuses on simple regression models with predictability of various putative predictors being tested one at the time (see e.g. Campbell and Yogo, 2006), the natural question arises, how to deal with multiple regressors in this framework. Dealing with the multiple regression case is however more demanding; see the Appendix for an explanation. The solution we favor here is to run individual regressions to detect individual predictability, paired with multiple testing techniques to maintain the overall significance level. We leave a detailed discussion of this approach for further research.

2.3 Finite-sample analysis

2.3.1 Setup

We now compare the proposed test $\bar{t}$ procedure with the IVX procedure of Kostakis et al. (2015), which is popular in the literature (see e.g. Gonzalo and Pitarakis, 2012; Phillips and Lee, 2013; Lee, 2016; Demetrescu and Hillmann, 2020; Demetrescu et al., 2020). In order to do this, we perform Monte Carlo Simulations and display the rejection rates for different combinations of α_1 and α_2. Our DGP follows Demetrescu and Hillmann (2020) with $t = 1, \ldots, T$ for $T = 250$:

$$y_t = \beta x_{t-1} + u_t,$$
$$x_t = \rho x_{t-1} + v_t$$

and

$$\begin{pmatrix} u_t \\ v_t \end{pmatrix} \overset{iid}{\sim} N\left(0,\ \sigma_t^2 \begin{pmatrix} 1 & \delta \\ \delta & 1 \end{pmatrix}\right)$$

with $\rho = 1 - c/T$, $c \in \{0, 10, 25\}$ and $\beta = b/T$ with $b \in \{-20, -5, 0, 5, 20\}$. We allow for contemporaneous correlation of the innovation terms with $\sigma_t^2 = 1$ and $\delta = -0.95$ and compute 5000 Monte Carlo replications for each setup. The variables u_t and v_t are obtained by generating two independent standard normal variables ψ_{1t} and ψ_{2t} and multiplying the vector $(\psi_{1t}, \psi_{2t})'$ with the Cholesky factor resulting from the decomposition of the desired covariance matrix of $(u_t, v_t)'$.

2.3.2 Preliminary simulations

We study first the role the choice of the loss functions (i.e. of the asymmetry parameters $\alpha_{1,2}$) plays in the behavior of the proposed test statistic. Size and power for different combinations of values for $\alpha_{1,2}$ of the two sided combination test are displayed in the heatmaps of Figure 2.1. The test statistic $\bar{t}$ from 2.6 is computed for $\alpha_{1,2} \in \{0.1, 0.15, \ldots, 0.9\}$ with $\alpha_1 \neq \alpha_2$. Please note, that the color scaling is different for each heatmap, spanning from lowest to highest rejection rate within each heatmap.

The middle panels in Figure 2.1 with $b = 0$ indicate the sizes for different combinations of values of α_1 and α_2. Crucial oversizedness is highlighted by blue (> 0.07) and red (> 0.09) tiles and especially occurs for extreme but similar values of α_1 and α_2 close to 0.1 or 0.9. The lowest power can be found in the center of each heatmap with α_1 and α_2 both being close to 0.5. Heatmaps above and below the third row in Figure 2.1 with $b \neq 0$ display the power of each combination. Not surprisingly, we observe higher rejection frequencies for those combinations of values of $\alpha_{1,2}$ which also lead to oversizedness under the null. We also provide results for left- and right-tailed tests as well as for a sample size of $T = 1000$ in the Appendix; see Figures 2.5, 2.6 and 2.7. We observe qualitatively similar behavior for the larger sample size.

Choosing α_1 and α_2 too close to one another leads to oversizedness, but at the same time to power tends to be higher when α_1 and α_2 are closer. We also observe more power for higher asymmetry of the loss functions. Furthermore, the distributional features of the errors also influence the power levels, and, particularly interesting, the combinations of α_1 and α_2 for which power is higher under the alternative; see Figure 2.2, where the shocks follow asymmetric distributions generated by replacing the standard normal distribution for ψ_{1t} and $\psi_{2,t}$ with standardized $\chi^2(3)$ distributions prior to the multiplication with the Cholesky factor. We also provide results for left- and right-tailed tests in the Appendix; see Figures 2.8 and 2.9, who confirm the influence of the asymmetry on the optimal choice of α_1 and α_2.

Therefore, it pays to look in a data-driven manner for a suitable combination of α_1 and α_2, and, to this end, we maximize an estimate of ζ subject to the restriction that $|\alpha_1 - \alpha_2| \geq \Delta$ for a suitable choice of Δ. This results in an adaptive test which we examine in the following.

2.3.3 Comparison

After having derived a rule to find a suitable combination for the loss function parameters, we move on to compare the adaptive test in terms of size and power to other tests. As described above, we perform a grid search to maximize ζ from equation (2.7) to find suitable values for the tuning parameters α_1 and α_2. We set the following restrictions for these tuning parameters: (i) $\alpha_1, \alpha_2 \in [0.1, 0.9]$ and (ii) $|\alpha_1 - \alpha_2| \geq \Delta$ with Δ depending on the sample sizes. We choose $\Delta = 0.2$ for $T < 500$, $\Delta = 0.1$ for $500 \leq T < 1000$ and $\Delta = 0.05$ for $1000 \leq T$. We therefore use Δ as increment in our grid search for αs. (We tested finer grids, but they did not perform better.) The resulting combination of optimal values for $\alpha_{1,2}$ is then used to compute the combination test, resulting in the test statistic $\bar{t}_{ADA}$.

Within the remainder of this section, we perform a series of Monte Carlo Simulations to compare the adaptive test with the IVX test of Kostakis et al. (2015) in terms of size and power. Further attention is paid on weighted combinations of both procedures as

$$t_{comb} = \frac{\gamma \bar{t}_{ADA} + (1-\gamma) t_{IVX}}{\sqrt{\gamma^2 + (1-\gamma)^2}} \quad \text{with} \quad \gamma = |M_3|/(1 + |M_3|) \tag{2.8}$$

and M_3 being the third standardized moment of the residuals from a linear regression of y_t on x_{t-1}. We do this because power and size are typically better for $\bar{t}_{ADA}$ for asymmetric error distributions, and typically worse than IVX for symmetric error distributions; see Figures 2.1 and 2.2 again. The expression of t_{comb} is due to the fact that $\bar{t}_{ADA}$ and t_{IVX} can be show to be asymptotically independent under our assumptions; we do not provide the details to save space.

We test the null hypothesis of no predictability in one-sided as well as in two-sided tests and apply the DGP from Section 2.3.2. We modify the DGP in two respects: (i) we allow for an explosive regressor with $\rho = 1 - c/T$ and $c \in \{-5, 0, 10, 25\}$ and (ii) we include more values for $\beta = b/T$ with $b \in \{\pm 20, \pm 10, \pm 5, \pm 2, 0\}$.

The rejection rates for the IVX test, the adaptive test and a weighted combination of both procedures in a two-sided setup are summarized in the first three columns of Table 2.1. The adaptive test turns out to be oversized, regardless of the regressors degree of persistence. Despite its oversizedness, it has the lowest power in our comparison. Although the adaptive test does

Figure 2.1: Size and power for two-sided $\bar{t}$ tests for different combinations of α_1 and α_2; $T = 250$, u_t and v_t jointly normally distributed.

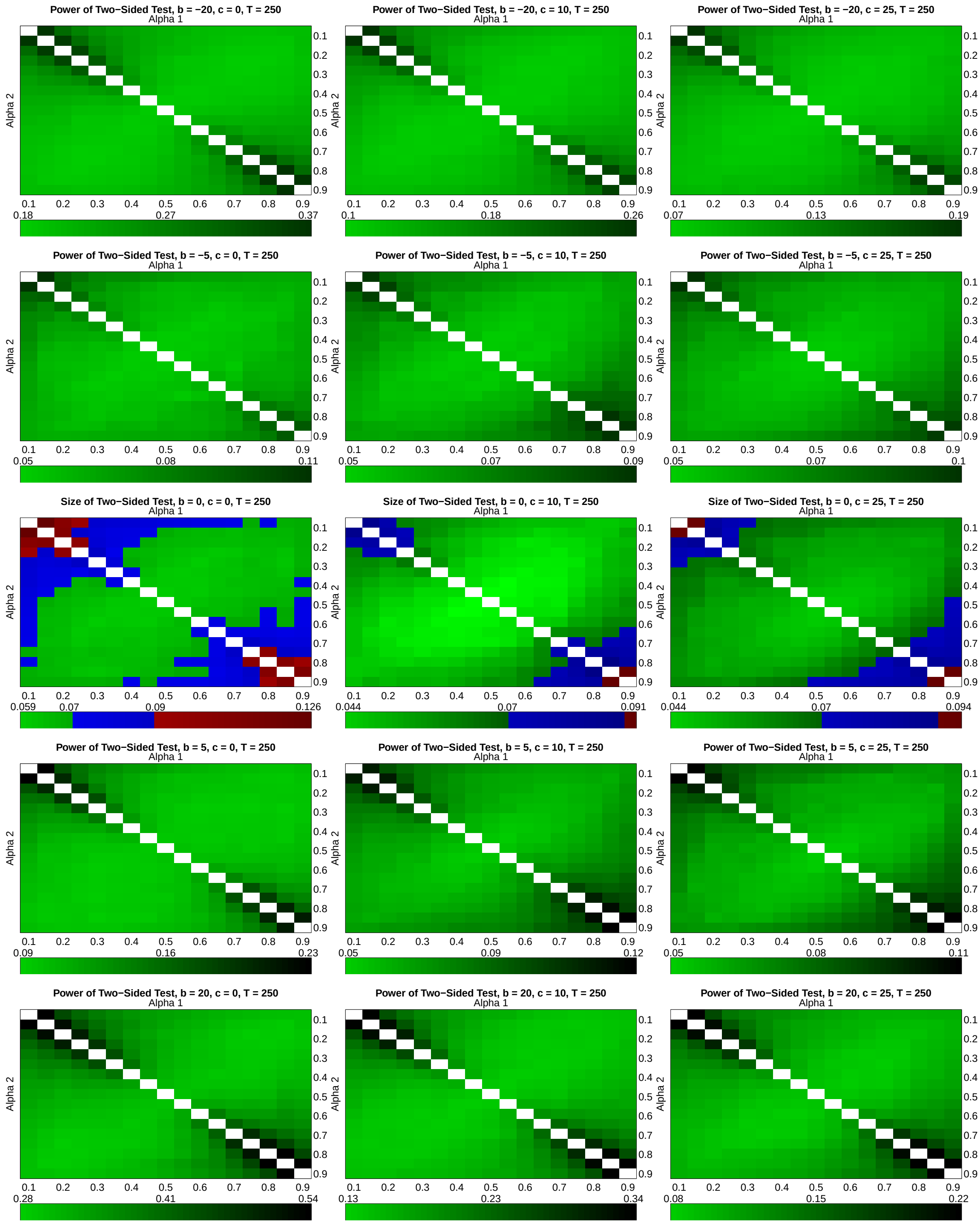
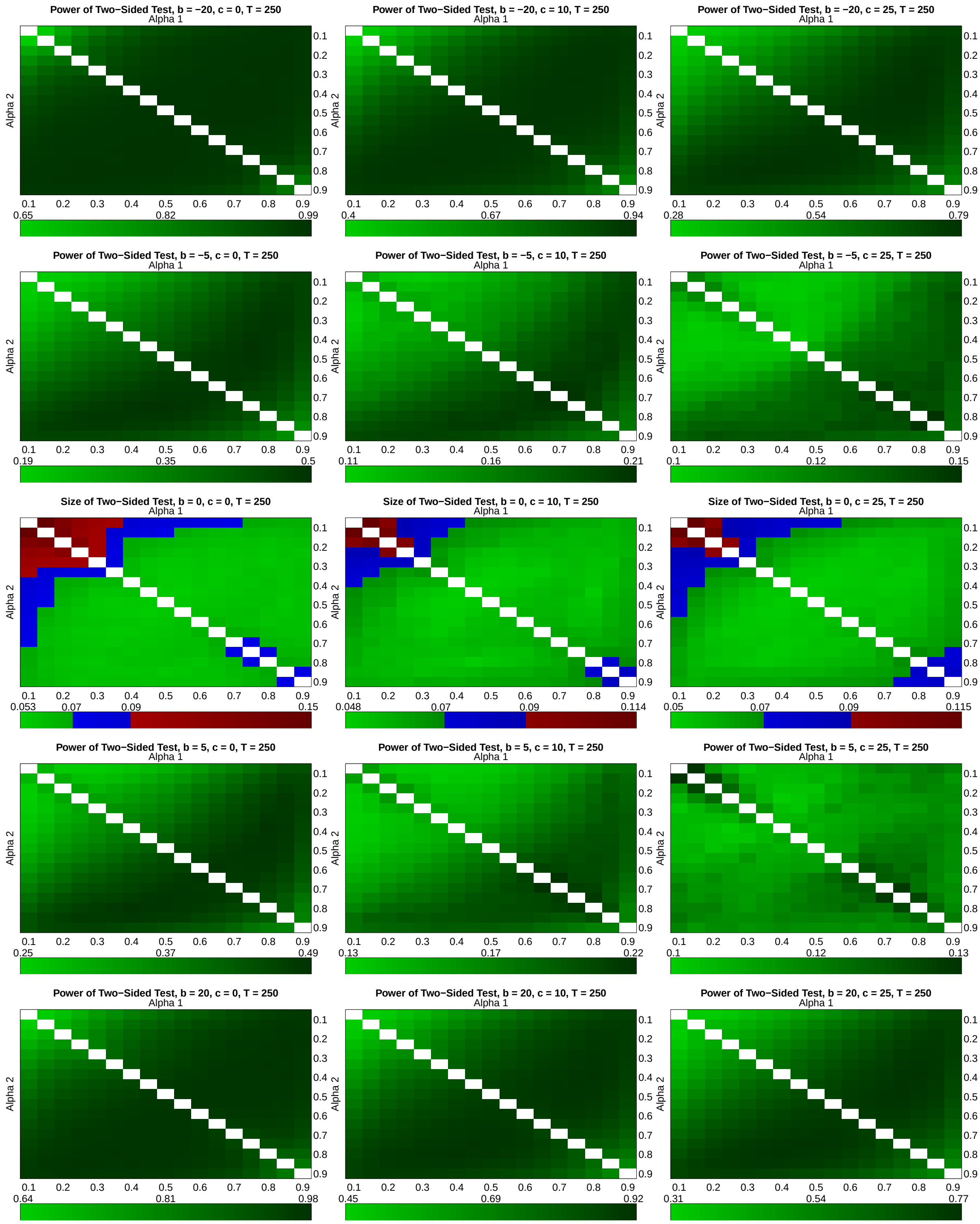


Figure 2.2: Size and power for two-sided $\bar{t}$ tests for different combinations of α_1 and α_2; $T = 250$, u_t and v_t follow skewed distributions.



not perform on point, a weighted combination with IVX is able to outperform IVX in terms of size and power.

Corresponding rejection rates for the left-tailed testing setup are depicted in the middle columns of Table 2.1. IVX is dominated by the combination approach in this task, but both are severely undersized. The adaptive test on the other hand shows a viable size while it is outperformed by the other tests in terms of power. We therefore construct an additional combination approach with weights shifted towards the adaptive test to increase the combination tests size. We replace the weights γ in equation (2.8) by $\gamma^{(a)} = \theta + (1-\theta)\gamma$ with $\theta \in [0,1]$; we use $\theta = 0.5$. The size of this test increases compared to the previous combination test, while its power decreases. No approach dominates in this setup, but we tend to favour the second combination approach with weights shifted towards the adaptive test due to its decent performance in terms of size and power.

The behaviour of the tests in a right-tailed setup is opposed to the behaviour in a left-tailed setup, the tests now tend to be oversized. We again construct an additional combination test which shifts the weights towards IVX, since it is less oversized than the adaptive test. Similar to the left-tailed case, we now replace γ in equation (2.8) by $\gamma^{(b)} = (1-\tilde{\theta})\gamma$ with $\tilde{\theta} \in [0,1]$; we use $\tilde{\theta} = 0.5$. The combination test performs better after adjusting its weights. Nevertheless, it is still inferior to IVX in terms of both, size and power. IVX is the best performing procedure in this right-sided testing setup, even when we further increase $\tilde{\theta}$.

Note that we employed a negative covariance $\delta = -0.95$ for the innovations u_t and v_t in the previous Monte Carlo Simulations. The Monte Carlo results change when we use a positive δ, as can be seen in Table 2.5 for $\delta = +0.95$ (in the Appendix). Roughly speaking, when the sign of δ switches, the behaviour for positive and negative betas as well as the behaviour of left-tailed and right-tailed tests flips over (qualitatively).

Summing up, the combination of adaptive test and IVX performs best in our setup when dealing with a two-tailed testing scheme. For one-sided tasks we cannot tell which test performs best a priori, it depends on both, the null hypothesis and the direction of the innovations correlation. One drawback of one-sided adaptive tests are too large rejection rates when testing in the wrong direction in presence of highly persistent or even explosive regressors.

Additionally, we performed Monte Carlo Simulations for $T = 1000$ (Table 2.6), the results are (qualitatively) identical to those for $T = 250$. We furthermore simulated with innovations from a leptokurtic distribution (i.e. standardized $t(3)$, Table 2.7), without much effect on the results. Innovations from an asymmetric distribution (i.e. standardized $\chi^2(3)$, Table 2.8) on the other hand have a considerable impact. In all test situations, left-, right- and two-tailed, the adaptive test performs best when the innovation terms are distributed asymmetrically. Hence, the adaptive test might be a suitable choice if one faces skewed data.

Table 2.1: Rejection rates: $T = 250$ and $\delta = -0.95$

	$H_0: \beta = 0$ vs. $H_1: \beta \neq 0$			$H_0: \beta \geq 0$ vs. $H_1: \beta < 0$				$H_0: \beta \leq 0$ vs. $H_1: \beta > 0$			
b	IVX	ADA	Comb	IVX	ADA	Comb	$\theta = 0.5$	IVX	ADA	Comb	$\tilde{\theta} = 0.5$
					c = -5 ($\rho = 1.02$)						
-20	0.9910	0.9178	0.9710	0.9928	0.7444	0.9574	0.8140	0.0000	0.1852	0.0192	0.0036
-10	0.9714	0.8604	0.9614	0.9754	0.7072	0.9650	0.8132	0.0000	0.1732	0.0032	0.0006
-5	0.9486	0.7920	0.9456	0.9554	0.6600	0.9530	0.8064	0.0042	0.1564	0.0046	0.0042
-2	0.8918	0.6190	0.8936	0.8996	0.5462	0.9002	0.7822	0.0096	0.1184	0.0104	0.0104
0	0.0322	0.1004	0.0390	0.0012	0.0588	0.0022	0.0288	0.0666	0.0976	0.0756	0.0714
2	0.9596	0.6324	0.9602	0.0000	0.1142	0.0000	0.0240	0.9780	0.5676	0.9792	0.9788
5	0.9980	0.7984	0.9948	0.0000	0.1550	0.0008	0.0682	0.9984	0.6704	0.9956	0.9986
10	1.0000	0.8766	0.9864	0.0000	0.1692	0.0046	0.1098	1.0000	0.7230	0.9844	0.9968
20	1.0000	0.9324	0.9782	0.0000	0.1836	0.0170	0.1390	1.0000	0.7586	0.9654	0.9876
					c = 0 ($\rho = 1$)						
-20	0.7896	0.3028	0.8012	0.8422	0.3478	0.8578	0.7712	0.0000	0.0320	0.0000	0.0000
-10	0.3716	0.1300	0.3840	0.4520	0.1584	0.4670	0.3978	0.0018	0.0402	0.0020	0.0018
-5	0.0944	0.0778	0.1028	0.1298	0.0794	0.1340	0.1322	0.0130	0.0568	0.0132	0.0124
-2	0.0284	0.0812	0.0322	0.0158	0.0488	0.0172	0.0380	0.0462	0.0906	0.0552	0.0500
0	0.0412	0.0876	0.0514	0.0008	0.0366	0.0012	0.0070	0.0920	0.1128	0.1068	0.0990
2	0.1272	0.1098	0.1502	0.0000	0.0370	0.0000	0.0012	0.2480	0.1460	0.2856	0.2646
5	0.5692	0.1540	0.6116	0.0000	0.0288	0.0000	0.0002	0.7846	0.2050	0.8118	0.8032
10	0.9372	0.2670	0.9498	0.0000	0.0252	0.0000	0.0000	0.9790	0.3268	0.9854	0.9862
20	0.9940	0.4708	0.9950	0.0000	0.0400	0.0000	0.0006	0.9988	0.5110	0.9988	0.9986
					c = 10 ($\rho = 0.96$)						
-20	0.7084	0.1786	0.7176	0.7770	0.2230	0.7836	0.6576	0.0000	0.0300	0.0000	0.0000
-10	0.2900	0.1014	0.2994	0.3778	0.1258	0.3832	0.3126	0.0028	0.0456	0.0030	0.0030
-5	0.0852	0.0760	0.0950	0.1316	0.0758	0.1372	0.1384	0.0126	0.0584	0.0140	0.0136
-2	0.0366	0.0748	0.0438	0.0428	0.0586	0.0478	0.0650	0.0346	0.0706	0.0386	0.0350
0	0.0350	0.0736	0.0422	0.0134	0.0480	0.0154	0.0350	0.0634	0.0814	0.0690	0.0660
2	0.0510	0.0740	0.0614	0.0028	0.0460	0.0044	0.0150	0.1058	0.0900	0.1220	0.1144
5	0.1176	0.0994	0.1400	0.0000	0.0400	0.0000	0.0048	0.2354	0.1262	0.2580	0.2438
10	0.4886	0.1306	0.5170	0.0000	0.0326	0.0000	0.0002	0.6948	0.1726	0.7180	0.7092
20	0.9884	0.2432	0.9906	0.0000	0.0320	0.0000	0.0000	0.9984	0.2924	0.9986	0.9988
					c = 25 ($\rho = 0.9$)						
-20	0.5372	0.1286	0.5504	0.6426	0.1656	0.6502	0.5040	0.0002	0.0374	0.0006	0.0004
-10	0.1810	0.0920	0.1920	0.2638	0.1102	0.2720	0.2304	0.0042	0.0484	0.0048	0.0048
-5	0.0640	0.0772	0.0706	0.1020	0.0800	0.1092	0.1250	0.0146	0.0534	0.0182	0.0166
-2	0.0358	0.0672	0.0452	0.0452	0.0632	0.0490	0.0696	0.0300	0.0564	0.0338	0.0322
0	0.0310	0.0728	0.0398	0.0212	0.0572	0.0244	0.0490	0.0528	0.0754	0.0596	0.0560
2	0.0422	0.0754	0.0494	0.0090	0.0516	0.0094	0.0306	0.0780	0.0834	0.0890	0.0846
5	0.0800	0.0880	0.0944	0.0022	0.0430	0.0020	0.0096	0.1610	0.1008	0.1764	0.1668
10	0.2304	0.0986	0.2470	0.0002	0.0330	0.0004	0.0034	0.3708	0.1266	0.3944	0.3836
20	0.7982	0.1624	0.8100	0.0000	0.0342	0.0000	0.0002	0.9188	0.2026	0.9248	0.9240

Notes: Rejection rates for IVX-, Adaptive- and Combination-Tests for Sample Size $T = 250$ and $\delta = -0.95$.

2.4 Predictability of S&P 500 Stock Returns

In this section, we focus on the issue of stock return predictability and apply the previously discussed tests to S&P 500. We employ log Dividend Price Ratio [logDP], log Earnings Price Ratio [logEP], Dividend Yield [DY], Book to Market Ratio [BM], Treasury-Bill Rates [TBL], Default Yield Spread [DFY], Stock Variance [SVAR], Long Term Yield [LTY], Long Term Rate of Returns [LTR], Inflation [INFL] and Term Spread [TMS] as (lagged) predictor variables. The data were obtained from the webpage of Amit Goyal. We make use of the most recent update of this data set (as to December 2020) and use demeaned monthly data from 1926:M01 to 2019:M12 in this process. We drop Net Equity Expansion and the Cross Sectional Premium, since they are not available for the whole testing period. As dependent variable we use Stock Returns including dividends. See Welch and Goyal (2008) for further details regarding the construction of these variables.

We perform a moving window analysis with windows of length 5, 10, 20 and 40 years which leaves us with 1069, 1009, 889 and 649 shifted windows, respectively. Note that TBL is constant for 60 consecutive observations in the 1940s and therefore neglected for the analysis of windows with a length of 5 years.

The two-sided rejection rates for IVX, the adaptive test and the previously discussed combination thereof are summarized in Table 2.2. We furthermore added OLS to the comparison, though keeping in mind that it tends to lead to spurious findings due to overrejections (c.f. Stambaugh (1999)). In addition to the rejection rates we display the estimated AR(1) parameters for each regressor as indicator for its degree of persistence.

Rejection rates increase for most combinations of test procedure and predictor variable when the window length increases. The adaptive test tends to indicate predictability more often than all other procedures in this comparison, even more frequent than the oversized OLS test. IVX, on the other hand, yields the lowest rejection rates and thus shows the least predictability. The adaptive test yields its lowest rejection rate for LTR while all other procedures reach (relatively) high rejection rates for this predictor. Note that LTR is by far the least persistent regressor in our setup with an estimated AR(1)-parameter of 0.039 (which we did not cover in our previous Monte Carlo Simulations).

We additionally examine time series plots of the test statistics evaluated over rolling windows to gain a better insight into stock return predictability. The plots in Figure 2.3 show exemplarily whether and when stock returns are predictable for logDP and logEP as regressor variables; plots for further regressor variables can be found in Figure 2.4. We display the evolution of the test statistic development over time for OLS-, IVX-, adaptive- and combination-test with rolling windows of length 5, 10 and 20 years. The dashed horizontal lines indicate bootstraped critical values for the combination test accounting for multiple testing for each regressor and window length separately (in red for one-tailed tests and blue for two-tailed tests). See the Appendix for the exact bootstrap scheme and its justification. In contrast, the OLS statistic for instance is essentially more difficult to bootstrap in our setup with uncertain persistence.

Table 2.2: Two-sided S&P 500 rejection rates

length	x_t	logDP	logEP	DY	BM	TBL	DFY	SVAR	LTY	LTR	INFL	TMS
	AR(1)	0.994	0.987	0.988	0.987	0.993	0.975	0.632	0.997	0.039	0.482	0.962
5	OLS	25.82	13.94	17.40	23.57	-	13.19	11.60	9.64	10.76	8.61	12.54
	IVX	3.56	3.93	4.58	0.75	-	4.96	4.30	3.65	5.61	1.68	7.86
	ADA	43.87	36.76	24.14	40.88	-	31.62	28.72	23.67	23.29	18.43	20.49
	COMB	22.64	10.95	17.77	20.58	-	14.69	21.61	14.78	15.06	17.59	14.41
10	OLS	24.58	12.98	17.25	31.22	16.95	18.34	16.75	10.11	17.64	14.57	11.00
	IVX	1.29	5.15	6.84	0.69	9.51	9.02	0.00	0.69	12.59	2.48	11.30
	ADA	51.04	34.49	18.93	40.04	37.86	31.91	36.27	28.54	14.07	10.51	20.71
	COMB	28.15	8.33	15.96	18.14	28.94	23.89	19.62	18.04	19.92	11.79	23.09
20	OLS	44.99	16.31	23.62	37.35	19.69	28.68	17.77	12.49	27.56	25.76	28.57
	IVX	0.00	1.91	4.61	3.04	6.52	16.31	0.00	0.00	20.81	3.15	19.57
	ADA	42.30	57.59	33.18	38.13	24.86	36.45	46.12	38.58	12.94	24.86	14.85
	COMB	15.19	15.19	19.12	16.09	12.94	29.36	26.32	4.95	27.11	23.51	30.03
40	OLS	58.24	31.90	35.44	39.14	8.94	51.93	24.50	2.16	63.79	63.95	43.30
	IVX	19.57	6.47	10.32	3.39	0.31	8.94	0.00	0.46	49.62	35.75	27.74
	ADA	34.05	28.81	41.91	39.91	40.22	50.54	71.96	22.80	0.31	36.36	12.33
	COMB	50.69	19.11	18.80	18.18	6.16	52.08	36.06	10.79	49.62	23.88	37.60

Notes: Two-tailed rejection rates for rolling windows of size 5, 10, 20 and 40 years. Estimated autoregressive coefficients AR(1) indicate the degree of persistence for each regressor.

All regressor variables and window sizes have in common that predictability is found most often by the adaptive test. Overall, predictability in subsequent periods appears more often for larger window sizes. This is an indication of a weak predictive signal. We find stock return predictability in the volatile 1970s with its Oil Shock for logDP and logEP which is in line with the findings of Welch and Goyal (2008). We furthermore find some evidence for a period of stock return predictability beginning in the Great Recession around 2008/2009 and lasting for some years, depending on the predictor variable and its moving window lengths. If predictability by a certain predictor is indicated for a few consecutive months during the Great Moderation (between Oil Shock and Great Recession), it is not confirmed by other predictor variables or moving window sizes. Moreover, we find evidence for stock return predictability during World War II for several predictor variables. Overall, we manage to show stock return predictability for volatile business cycle periods, while we are not able to do so for calmer times.

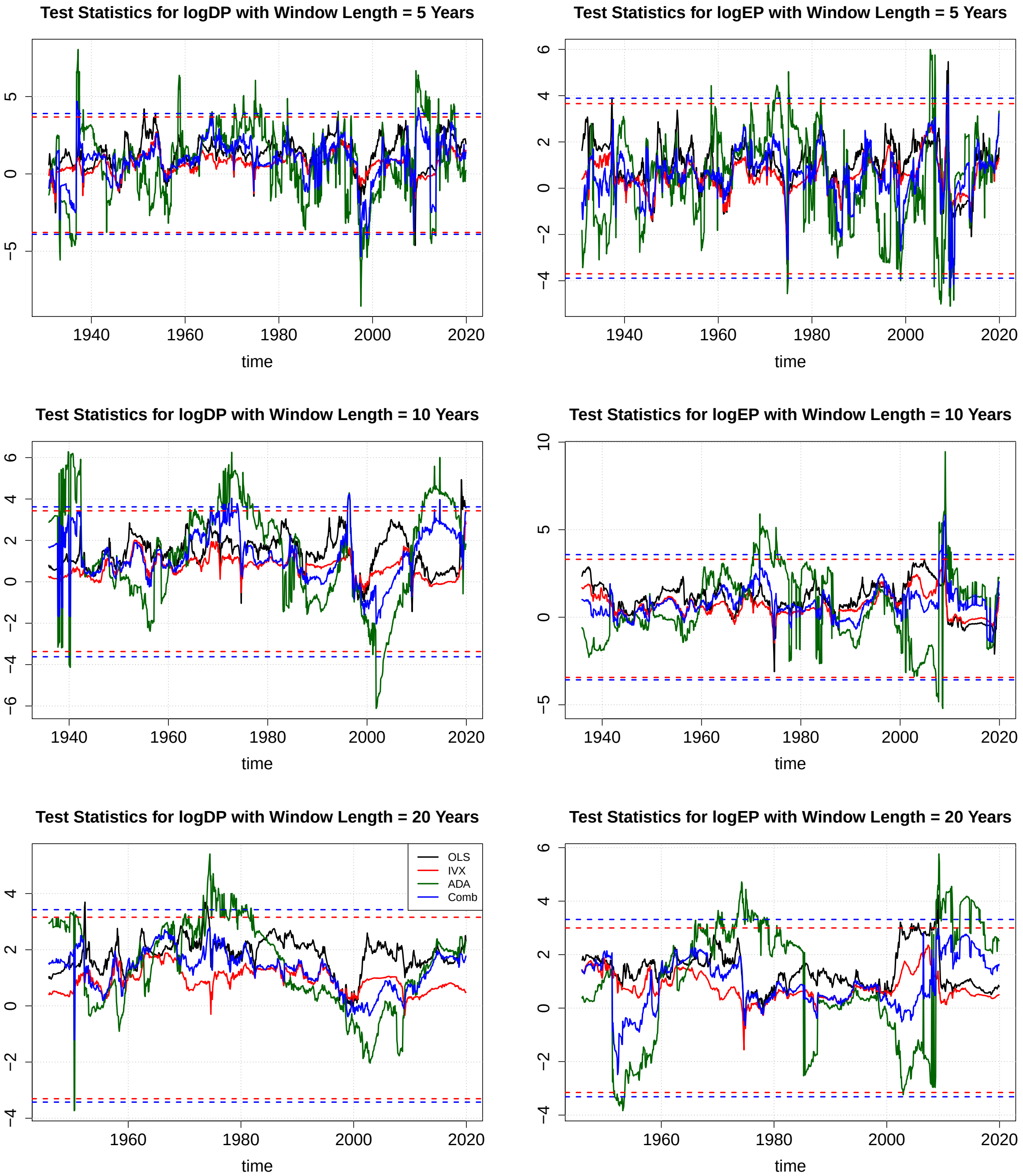
Rejection rates for left- and right-tailed tests are displayed in Tables 2.3 and 2.4. It might be possible that a predictors influence changes its direction over time. However, we do not consider this to be very likely in the long run. From Tables 2.3 and 2.4 we conclude that (lagged) TBL, LTY and INFL tend to have a negative effect on stock returns while logDP, logEP, DFY, BM, DFY, LTR and TMS have a positive effect. The results for one-tailed tests look qualitatively similar to those of two-tailed tests after matching each regressor either to a left- or right-sided test. IVX tends to indicate predictability the least often, while the adaptive test signalises predictability the most frequently. Finally, we have to point out that the adaptive test tends to have too large rejection rates when testing in the wrong direction for one-sided tests, as previously seen in the Monte Carlo Simulations

2.5 Concluding remarks

We discussed the construction of asymptotically Gaussian tests of no predictability under uncertain predictor persistence and possible endogeneity. The new test is obtained by linearly combining 2 different t statistics in an M estimation framework, with weights chosen such that non-Gaussian components cancel out while at the same time asymptotic local power is maximal in a certain class of M estimators. A fixed regressor bootstrap is employed to deal with subsample-based testing to detect windows of predictability.

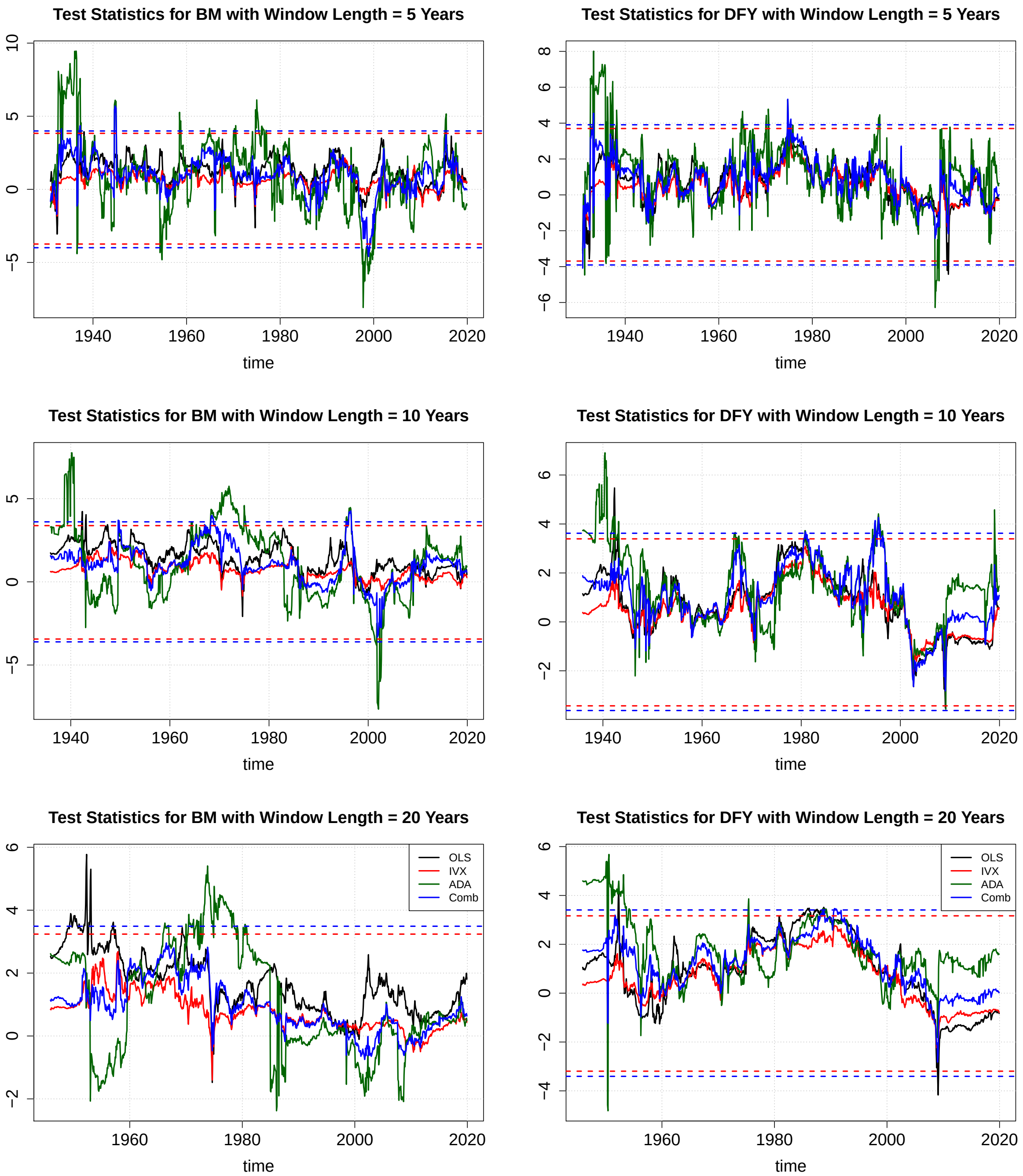
We analyzed the adaptive tests power when predicting S&P 500 stock returns and compared it to the power of an IVX based test. Computations were performed separately for 11 regressor variables in a rolling window setup with various window lengths. The null hypothesis of no predictability is rejected more often for the adaptive test than for IVX, for one-tailed as well as for two-tailed tests. We are able to find evidence for stock return predictability in certain periods of time when applying critical values from the fixed regressor bootstrap. These periods coincide with the volatile business cycle periods of the investigated data, namely World War II, Oil Crisis and Great Recession. Beyond these volatile periods, we find no significant predictability of stock returns.

Figure 2.3: Time Series Plot of Test Statistic for logDP and logEP.



***Notes*:** Time Series Plot of Test Statistic for logDP in left and logEP in right panels for moving windows of length 5, 10 and 20 years from top to bottom. Dashed horizontal lines indicate bootstrapped critical values for one-tailed (red) and two-tailed tests (blue).

Figure 2.4: Time Series Plot of Test Statistic for BM and DFY.



***Notes*:** Time Series Plot of Test Statistic for BM in left and DFY in right panels for moving windows of length 5, 10 and 20 years from top to bottom. Dashed horizontal lines indicate bootstrapped critical values for one-tailed (red) and two-tailed tests (blue).

Table 2.3: Left-sided S&P 500 rejection rates

length	x_t	logDP	logEP	DY	BM	TBL	DFY	SVAR	LTY	LTR	INFL	TMS
	AR(1)	0.994	0.987	0.988	0.987	0.993	0.975	0.632	0.997	0.039	0.482	0.962
5	OLS	0.94	0.468	0.374	0.374	-	2.152	11.038	13.377	1.216	14.219	6.642
	IVX	0.19	0.28	0.28	0.19	-	0.56	0.37	6.08	1.68	7.02	2.99
	ADA	17.03	16.18	9.64	12.35	-	4.12	11.13	22.83	9.07	18.80	6.55
	COMB	6.08	3.37	2.53	3.27	-	1.40	11.97	18.05	7.02	18.90	5.05
10	OLS	0.00	0.40	0.40	0.10	23.69	2.48	23.69	17.84	0.00	22.60	0.00
	IVX	0.00	0.00	0.00	0.00	11.70	0.69	0.30	2.38	0.30	9.71	0.30
	ADA	10.41	14.57	9.81	5.95	33.99	0.60	7.04	28.54	8.92	11.40	13.08
	COMB	1.09	0.00	0.00	0.99	30.23	2.87	15.46	23.98	7.63	20.91	8.33
20	OLS	0.00	0.00	0.00	0.00	23.40	1.01	20.59	15.75	0.00	45.67	0.00
	IVX	0.00	0.00	0.00	0.00	12.71	0.68	0.00	1.13	0.00	22.39	0.00
	ADA	2.25	18.90	5.06	2.93	27.78	0.45	16.31	25.65	5.17	19.91	0.56
	COMB	0.00	0.68	0.23	0.00	18.00	0.56	16.54	10.91	1.24	35.10	4.61
40	OLS	0.00	0.00	0.00	0.00	18.80	0.00	37.13	5.09	0.00	67.64	0.00
	IVX	0.00	0.00	0.00	0.00	2.47	0.00	0.15	1.70	0.00	56.24	0.00
	ADA	0.00	18.64	0.00	0.00	36.67	0.31	43.30	24.19	0.00	25.42	0.15
	COMB	0.00	0.00	0.00	0.00	21.88	0.31	52.39	14.48	0.00	39.14	0.00

Notes: Left-tailed rejection rates for rolling windows of size 5, 10, 20 and 40 years. Estimated autoregressive coefficients AR(1) indicate the degree of persistence for each regressor.

Table 2.4: Right-sided S&P 500 rejection rates

length	x_t	logDP	logEP	DY	BM	TBL	DFY	SVAR	LTY	LTR	INFL	TMS
	AR(1)	0.994	0.987	0.988	0.987	0.993	0.975	0.632	0.997	0.039	0.482	0.962
5	OLS	37.42	25.73	27.41	34.89	-	18.90	11.13	3.09	18.24	0.47	15.62
	IVX	7.48	5.61	7.86	3.65	-	8.61	5.43	1.59	10.01	0.47	12.16
	ADA	36.30	27.88	23.76	37.70	-	36.95	30.59	7.30	20.67	7.67	23.20
	COMB	26.47	14.59	20.39	26.01	-	20.39	18.43	2.99	17.03	5.33	17.12
10	OLS	40.14	24.68	27.75	43.21	2.28	23.69	6.05	2.38	21.41	0.50	18.34
	IVX	4.96	9.61	10.80	5.95	1.09	13.68	2.28	1.09	17.94	0.00	17.74
	ADA	46.68	32.01	18.93	41.53	9.71	40.24	37.56	11.20	15.76	8.23	20.61
	COMB	37.17	12.39	19.82	23.79	3.87	31.02	20.61	2.38	25.77	2.68	24.18
20	OLS	64.34	33.86	33.97	48.26	1.35	31.16	3.71	1.35	36.33	0.00	38.13
	IVX	6.75	5.40	8.89	10.24	0.00	26.32	0.23	0.00	29.81	0.00	30.71
	ADA	47.92	44.88	41.28	41.06	3.60	44.88	34.42	19.57	17.21	17.89	19.46
	COMB	31.16	24.63	29.25	20.59	1.80	45.22	14.29	4.39	41.28	0.00	39.93
40	OLS	82.13	67.03	69.49	47.92	0.00	55.16	7.86	0.31	73.65	0.00	49.77
	IVX	33.90	11.09	15.72	9.71	0.00	38.98	0.00	0.00	63.48	0.00	38.68
	ADA	43.76	29.74	61.79	40.06	9.25	75.81	36.36	5.86	3.24	19.11	24.19
	COMB	55.93	29.89	54.70	24.19	0.00	60.25	8.48	0.15	58.71	0.00	43.76

Notes: Right-tailed rejection rates for rolling windows of size 5, 10, 20 and 40 years. Estimated autoregressive coefficients AR(1) indicate the degree of persistence for each regressor.

Appendix

Supplementary finite sample evidence

Figure 2.5: Size and power for left-sided $\bar{t}$ tests for different combinations of α_1 and α_2; $T = 250$, u_t and v_t jointly normal.

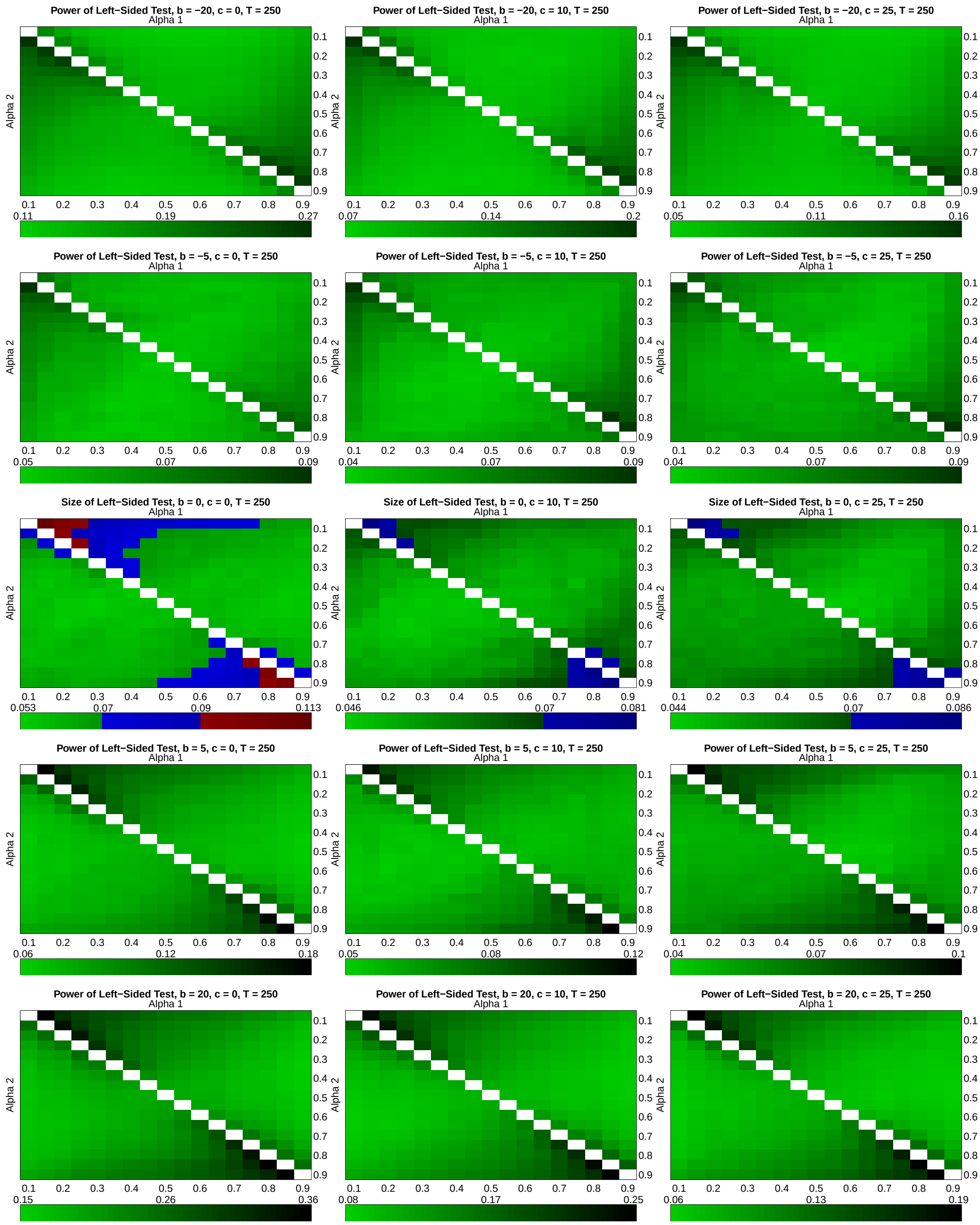


Figure 2.6: Size and power for right-sided $\bar{t}$ tests for different combinations of α_1 and α_2; $T = 250$, u_t and v_t jointly normal.

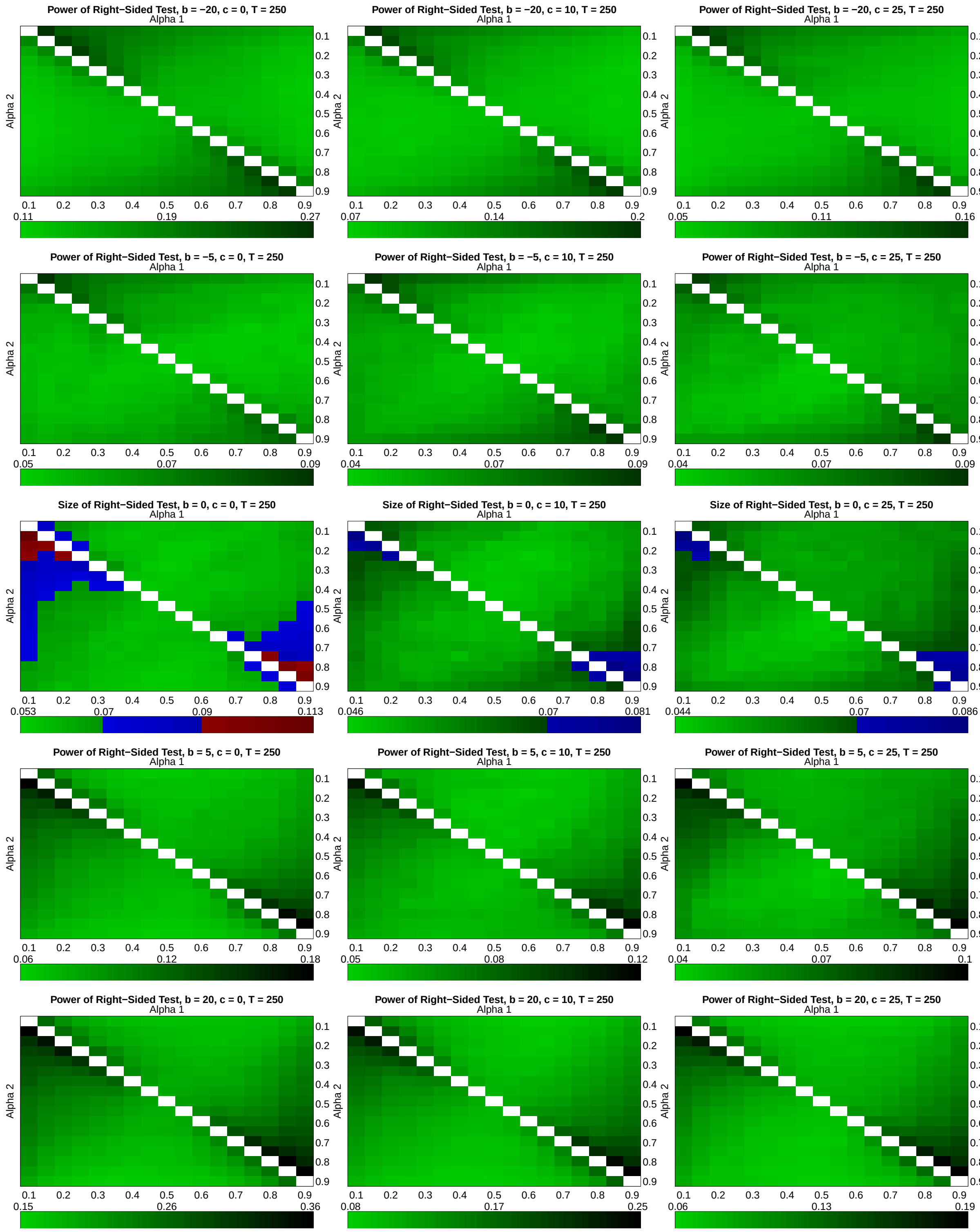


Figure 2.7: Size and power for two-sided $\bar{t}$ tests for different combinations of α_1 and α_2; $T = 1000$, u_t and v_t jointly normal.

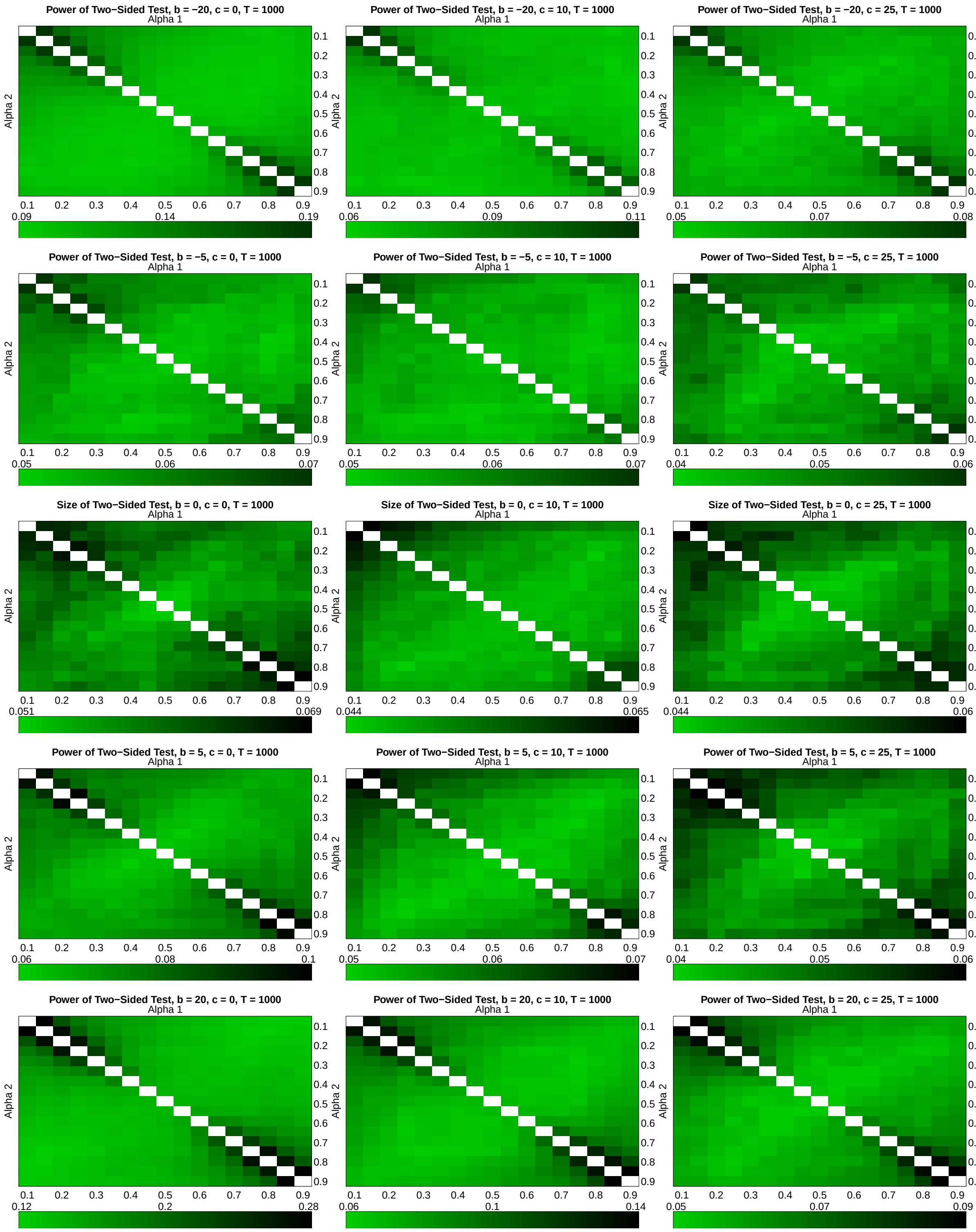


Figure 2.8: Size and power for left-sided $\bar{t}$ tests for different combinations of α_1 and α_2; $T = 250$, u_t and v_t follow an asymmetric distribution.

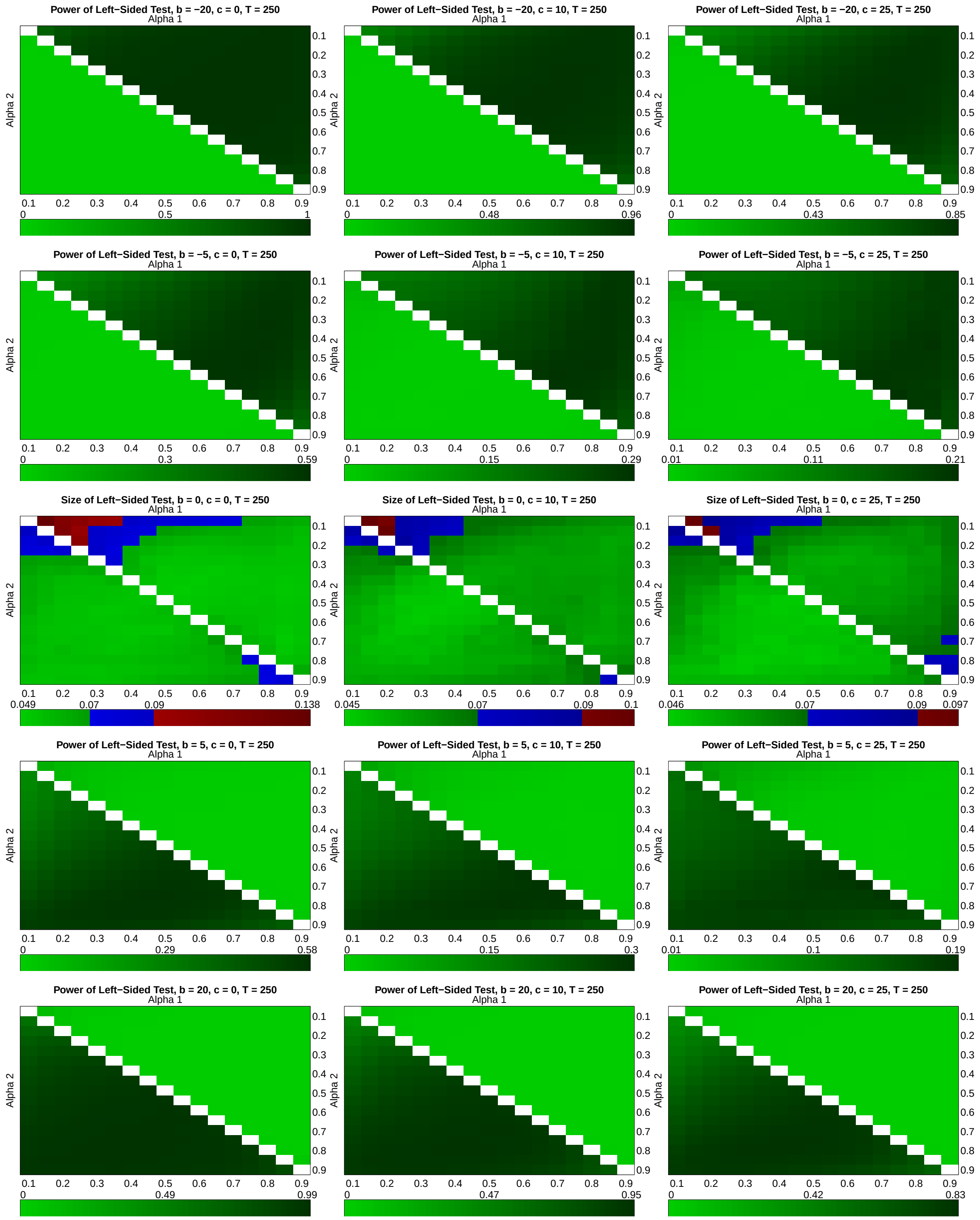


Figure 2.9: Size and power for right-sided $\bar{t}$ tests for different combinations of α_1 and α_2; $T = 250$, u_t and v_t follow an asymmetric distribution.

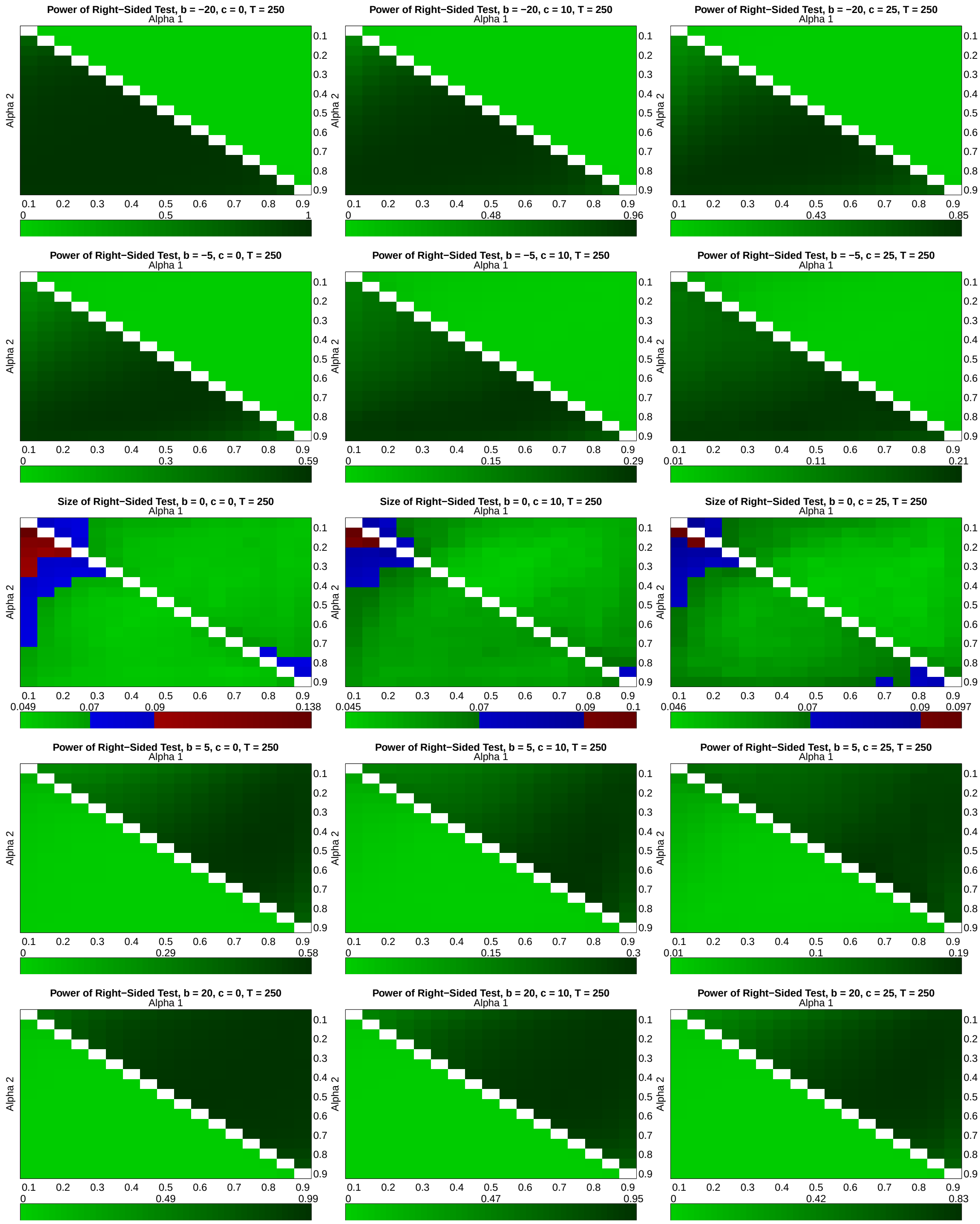


Table 2.5: Rejection rates: $T = 250$ and $\delta = +0.95$

	$H_0: \beta = 0$ vs. $H_1: \beta \neq 0$			$H_0: \beta \geq 0$ vs. $H_1: \beta < 0$				$H_0: \beta \leq 0$ vs. $H_1: \beta > 0$			
b	IVX	ADA	Comb	IVX	ADA	Comb	$\theta = 0.5$	IVX	ADA	Comb	$\tilde{\theta} = 0.5$
					c = -5 ($\rho = 1.02$)						
-20	1.0000	0.9336	0.9766	1.0000	0.7496	0.9660	0.9886	0.0000	0.1958	0.0150	0.1450
-10	1.0000	0.8812	0.9880	1.0000	0.7254	0.9862	0.9972	0.0000	0.1740	0.0040	0.1034
-5	0.9984	0.8058	0.9934	0.9990	0.6714	0.9946	0.9990	0.0000	0.1638	0.0008	0.0762
-2	0.9620	0.6262	0.9630	0.9770	0.5716	0.9796	0.9786	0.0000	0.1052	0.0000	0.0198
0	0.0310	0.0874	0.0382	0.0626	0.0858	0.0722	0.0676	0.0016	0.0596	0.0026	0.0290
2	0.8954	0.6078	0.8968	0.0120	0.1120	0.0128	0.0126	0.9016	0.5402	0.9046	0.7896
5	0.9516	0.7796	0.9494	0.0032	0.1510	0.0054	0.0036	0.9538	0.6614	0.9510	0.8154
10	0.9730	0.8746	0.9608	0.0006	0.1718	0.0080	0.0016	0.9760	0.7204	0.9588	0.8168
20	0.9900	0.9236	0.9688	0.0000	0.1936	0.0180	0.0030	0.9918	0.7420	0.9574	0.8072
					c = 0 ($\rho = 1$)						
-20	0.9954	0.4594	0.9958	0.9990	0.4988	0.9988	0.9994	0.0000	0.0424	0.0000	0.0004
-10	0.9382	0.2728	0.9516	0.9804	0.3288	0.9862	0.9852	0.0000	0.0224	0.0000	0.0000
-5	0.5642	0.1570	0.6116	0.7832	0.2078	0.8156	0.8062	0.0000	0.0248	0.0000	0.0000
-2	0.1178	0.1088	0.1464	0.2482	0.1418	0.2874	0.2668	0.0000	0.0338	0.0000	0.0012
0	0.0452	0.0908	0.0544	0.0952	0.1116	0.1130	0.1040	0.0000	0.0400	0.0004	0.0082
2	0.0306	0.0794	0.0358	0.0472	0.0814	0.0556	0.0520	0.0160	0.0518	0.0158	0.0330
5	0.0996	0.0868	0.1086	0.0142	0.0578	0.0176	0.0152	0.1358	0.0922	0.1426	0.1430
10	0.3842	0.1412	0.3964	0.0022	0.0426	0.0024	0.0024	0.4576	0.1698	0.4698	0.4114
20	0.7972	0.3186	0.8072	0.0000	0.0352	0.0000	0.0000	0.8512	0.3708	0.8598	0.7750
					c = 10 ($\rho = 0.96$)						
-20	0.9886	0.2360	0.9898	0.9972	0.2932	0.9982	0.9980	0.0000	0.0278	0.0000	0.0000
-10	0.4836	0.1244	0.5172	0.7034	0.1594	0.7234	0.7194	0.0000	0.0322	0.0000	0.0004
-5	0.1208	0.0930	0.1394	0.2398	0.1174	0.2616	0.2488	0.0000	0.0404	0.0000	0.0040
-2	0.0492	0.0830	0.0590	0.0996	0.0914	0.1086	0.1040	0.0018	0.0486	0.0022	0.0138
0	0.0338	0.0698	0.0404	0.0582	0.0828	0.0672	0.0604	0.0142	0.0536	0.0166	0.0354
2	0.0410	0.0764	0.0482	0.0310	0.0686	0.0384	0.0342	0.0464	0.0608	0.0496	0.0620
5	0.0912	0.0802	0.1014	0.0118	0.0554	0.0144	0.0128	0.1376	0.0848	0.1462	0.1458
10	0.2842	0.0944	0.2988	0.0014	0.0448	0.0034	0.0018	0.3786	0.1192	0.3908	0.3198
20	0.7064	0.1900	0.7182	0.0002	0.0314	0.0004	0.0002	0.7716	0.2306	0.7848	0.6662
					c = 25 ($\rho = 0.9$)						
-20	0.8088	0.1524	0.8262	0.9222	0.1936	0.9276	0.9274	0.0000	0.0246	0.0000	0.0000
-10	0.2248	0.0958	0.2384	0.3600	0.1212	0.3778	0.3678	0.0000	0.0414	0.0000	0.0040
-5	0.0768	0.0854	0.0900	0.1518	0.0936	0.1694	0.1604	0.0018	0.0462	0.0020	0.0136
-2	0.0386	0.0760	0.0444	0.0800	0.0770	0.0864	0.0822	0.0096	0.0500	0.0132	0.0320
0	0.0316	0.0782	0.0366	0.0502	0.0770	0.0568	0.0528	0.0228	0.0594	0.0242	0.0468
2	0.0408	0.0732	0.0444	0.0322	0.0682	0.0368	0.0344	0.0446	0.0650	0.0498	0.0706
5	0.0594	0.0752	0.0648	0.0134	0.0634	0.0170	0.0146	0.0948	0.0750	0.1032	0.1116
10	0.1708	0.0916	0.1862	0.0032	0.0518	0.0034	0.0034	0.2530	0.1026	0.2602	0.2270
20	0.5362	0.1260	0.5486	0.0000	0.0378	0.0000	0.0000	0.6458	0.1560	0.6522	0.4872

Notes: Rejection rates for IVX-, Adaptive- and Combination-Tests for Sample Size $T = 250$ and $\delta = +0.95$.

Table 2.6: Rejection rates: $T = 1000$ and $\delta = -0.95$

	$H_0: \beta = 0$ vs. $H_1: \beta \neq 0$			$H_0: \beta \geq 0$ vs. $H_1: \beta < 0$				$H_0: \beta \leq 0$ vs. $H_1: \beta > 0$			
b	IVX	ADA	Comb	IVX	ADA	Comb	$\theta = 0.5$	IVX	ADA	Comb	$\tilde{\theta} = 0.5$
c = -5 ($\rho = 1.005$)											
-20	0.9938	0.9264	0.9928	0.9948	0.7976	0.9936	0.9016	0.0000	0.1384	0.0006	0.0002
-10	0.9762	0.8712	0.9758	0.9784	0.7736	0.9786	0.9232	0.0004	0.1124	0.0008	0.0008
-5	0.9558	0.7586	0.9568	0.9576	0.6804	0.9578	0.9212	0.0044	0.1088	0.0044	0.0044
-2	0.9092	0.5960	0.9104	0.9146	0.5792	0.9154	0.8864	0.0096	0.0684	0.0102	0.0102
0	0.0354	0.0670	0.0370	0.0024	0.0430	0.0026	0.0198	0.0698	0.0752	0.0726	0.0712
2	0.9660	0.6060	0.9672	0.0000	0.0680	0.0000	0.0018	0.9802	0.5910	0.9804	0.9800
5	0.9992	0.7870	0.9994	0.0000	0.1032	0.0000	0.0170	1.0000	0.7134	0.9998	1.0000
10	1.0000	0.8752	1.0000	0.0000	0.1246	0.0002	0.0426	1.0000	0.7684	0.9998	1.0000
20	1.0000	0.9274	0.9996	0.0000	0.1326	0.0004	0.0676	1.0000	0.8038	0.9992	1.0000
c = 0 ($\rho = 1$)											
-20	0.7936	0.2782	0.8020	0.8392	0.3442	0.8428	0.7920	0.0002	0.0176	0.0002	0.0002
-10	0.3692	0.1184	0.3744	0.4436	0.1472	0.4492	0.4102	0.0038	0.0336	0.0032	0.0036
-5	0.0996	0.0632	0.1038	0.1364	0.0702	0.1384	0.1390	0.0176	0.0510	0.0188	0.0180
-2	0.0288	0.0626	0.0306	0.0146	0.0454	0.0152	0.0266	0.0452	0.0716	0.0502	0.0466
0	0.0552	0.0794	0.0582	0.0006	0.0344	0.0008	0.0066	0.1040	0.0940	0.1116	0.1072
2	0.1252	0.0794	0.1350	0.0000	0.0258	0.0000	0.0008	0.2616	0.1228	0.2820	0.2710
5	0.6114	0.1252	0.6324	0.0000	0.0208	0.0000	0.0000	0.8042	0.1790	0.8188	0.8134
10	0.9490	0.2190	0.9578	0.0000	0.0144	0.0000	0.0002	0.9842	0.2928	0.9888	0.9862
20	0.9976	0.4308	0.9982	0.0000	0.0166	0.0000	0.0000	0.9990	0.5038	0.9992	0.9992
c = 10 ($\rho = 0.99$)											
-20	0.7054	0.1390	0.7088	0.7758	0.1914	0.7802	0.6808	0.0002	0.0234	0.0002	0.0002
-10	0.2772	0.0772	0.2834	0.3604	0.0974	0.3630	0.3038	0.0014	0.0358	0.0020	0.0018
-5	0.0944	0.0672	0.0974	0.1384	0.0730	0.1416	0.1396	0.0128	0.0532	0.0132	0.0130
-2	0.0414	0.0654	0.0460	0.0460	0.0538	0.0472	0.0610	0.0346	0.0630	0.0358	0.0350
0	0.0394	0.0626	0.0422	0.0114	0.0474	0.0118	0.0306	0.0694	0.0760	0.0738	0.0714
2	0.0584	0.0628	0.0624	0.0018	0.0374	0.0026	0.0128	0.1130	0.0808	0.1170	0.1144
5	0.1272	0.0738	0.1352	0.0000	0.0322	0.0000	0.0036	0.2392	0.0974	0.2514	0.2448
10	0.5004	0.1098	0.5138	0.0000	0.0248	0.0000	0.0002	0.7014	0.1538	0.7126	0.7072
20	0.9892	0.1850	0.9906	0.0000	0.0208	0.0000	0.0000	0.9980	0.2564	0.9982	0.9982
c = 25 ($\rho = 0.975$)											
-20	0.5296	0.1044	0.5356	0.6296	0.1414	0.6384	0.5126	0.0002	0.0286	0.0004	0.0002
-10	0.1780	0.0676	0.1828	0.2584	0.0760	0.2618	0.2182	0.0038	0.0416	0.0038	0.0040
-5	0.0696	0.0614	0.0736	0.1050	0.0656	0.1084	0.1138	0.0194	0.0536	0.0196	0.0196
-2	0.0382	0.0662	0.0412	0.0450	0.0580	0.0446	0.0598	0.0360	0.0592	0.0388	0.0376
0	0.0332	0.0682	0.0356	0.0250	0.0496	0.0262	0.0396	0.0494	0.0678	0.0522	0.0500
2	0.0418	0.0648	0.0460	0.0086	0.0436	0.0090	0.0236	0.0782	0.0740	0.0836	0.0802
5	0.0760	0.0714	0.0812	0.0018	0.0378	0.0020	0.0112	0.1520	0.0924	0.1578	0.1538
10	0.2160	0.0786	0.2250	0.0000	0.0348	0.0000	0.0026	0.3588	0.1084	0.3688	0.3648
20	0.8118	0.1212	0.8210	0.0000	0.0250	0.0000	0.0000	0.9268	0.1708	0.9310	0.9288

Notes: Rejection rates for IVX-, Adaptive- and Combination-Tests for Sample Size $T = 1000$ and $\delta = -0.95$.

Table 2.7: Rejection rates: $T = 250$, $\delta = -0.95$ and $t(3)$-distribution

	$H_0:\ \beta = 0$ vs. $H_1:\ \beta \neq 0$			$H_0:\ \beta \geq 0$ vs. $H_1:\ \beta < 0$				$H_0:\ \beta \leq 0$ vs. $H_1:\ \beta > 0$			
b	IVX	ADA	Comb	IVX	ADA	Comb	$\theta = 0.5$	IVX	ADA	Comb	$\tilde{\theta} = 0.5$
c = -5 ($\rho = 1.02$)											
-20	0.9890	0.9786	0.9840	0.9910	0.9390	0.9726	0.9466	0.0000	0.0428	0.0146	0.0040
-10	0.9748	0.9490	0.9716	0.9762	0.9224	0.9684	0.9380	0.0002	0.0364	0.0064	0.0010
-5	0.9408	0.9024	0.9458	0.9454	0.8838	0.9464	0.9070	0.0034	0.0338	0.0074	0.0042
-2	0.8826	0.8010	0.8952	0.8916	0.8026	0.9038	0.8474	0.0098	0.0268	0.0108	0.0096
0	0.0316	0.0960	0.0602	0.0020	0.0818	0.0256	0.0642	0.0630	0.0766	0.0774	0.0672
2	0.9534	0.8040	0.9478	0.0000	0.0282	0.0024	0.0152	0.9714	0.8048	0.9630	0.9768
5	0.9948	0.9136	0.9878	0.0000	0.0308	0.0040	0.0228	0.9980	0.8960	0.9868	0.9984
10	0.9996	0.9550	0.9908	0.0000	0.0330	0.0058	0.0264	0.9998	0.9300	0.9872	0.9972
20	1.0000	0.9772	0.9926	0.0000	0.0418	0.0142	0.0370	1.0000	0.9394	0.9792	0.9916
c = 0 ($\rho = 1$)											
-20	0.7988	0.7794	0.8954	0.8450	0.8238	0.9222	0.8978	0.0000	0.0078	0.0012	0.0000
-10	0.4048	0.4868	0.5736	0.4712	0.5704	0.6328	0.6398	0.0018	0.0060	0.0022	0.0016
-5	0.1098	0.2196	0.2200	0.1472	0.2888	0.2834	0.3144	0.0122	0.0140	0.0100	0.0108
-2	0.0248	0.0814	0.0576	0.0148	0.1128	0.0592	0.1018	0.0382	0.0306	0.0450	0.0398
0	0.0480	0.0634	0.0594	0.0014	0.0478	0.0086	0.0318	0.1008	0.0644	0.1136	0.1066
2	0.1306	0.1114	0.1958	0.0000	0.0262	0.0018	0.0148	0.2692	0.1518	0.3246	0.3210
5	0.5892	0.2590	0.6372	0.0000	0.0174	0.0006	0.0046	0.7884	0.3330	0.7768	0.8394
10	0.9268	0.5172	0.9272	0.0000	0.0148	0.0012	0.0074	0.9778	0.5810	0.9570	0.9840
20	0.9950	0.7828	0.9868	0.0000	0.0100	0.0016	0.0050	0.9990	0.8174	0.9880	0.9982
c = 10 ($\rho = 0.96$)											
-20	0.7062	0.5590	0.8062	0.7732	0.6402	0.8554	0.7662	0.0002	0.0078	0.0006	0.0002
-10	0.3118	0.2554	0.4210	0.4024	0.3446	0.5014	0.4278	0.0022	0.0106	0.0018	0.0018
-5	0.0944	0.0988	0.1360	0.1440	0.1416	0.1898	0.1672	0.0104	0.0262	0.0124	0.0098
-2	0.0396	0.0640	0.0666	0.0526	0.0842	0.0810	0.0868	0.0318	0.0380	0.0356	0.0324
0	0.0342	0.0670	0.0532	0.0154	0.0540	0.0280	0.0460	0.0608	0.0676	0.0688	0.0658
2	0.0548	0.0778	0.0780	0.0016	0.0406	0.0066	0.0272	0.1098	0.0988	0.1390	0.1254
5	0.1450	0.1268	0.1932	0.0002	0.0264	0.0010	0.0154	0.2566	0.1746	0.3040	0.2946
10	0.4984	0.2712	0.5906	0.0000	0.0182	0.0002	0.0076	0.7060	0.3452	0.7350	0.7712
20	0.9720	0.5670	0.9528	0.0000	0.0132	0.0006	0.0052	0.9916	0.6310	0.9732	0.9940
c = 25 ($\rho = 0.9$)											
-20	0.5528	0.3760	0.6582	0.6540	0.4592	0.7426	0.5830	0.0000	0.0124	0.0014	0.0002
-10	0.2026	0.1676	0.2796	0.2856	0.2388	0.3722	0.3000	0.0016	0.0200	0.0040	0.0024
-5	0.0636	0.1002	0.1114	0.1030	0.1406	0.1590	0.1566	0.0118	0.0292	0.0176	0.0124
-2	0.0376	0.0684	0.0632	0.0474	0.0780	0.0734	0.0816	0.0306	0.0532	0.0404	0.0302
0	0.0330	0.0680	0.0548	0.0212	0.0654	0.0430	0.0606	0.0548	0.0638	0.0648	0.0562
2	0.0398	0.0730	0.0620	0.0082	0.0496	0.0164	0.0376	0.0834	0.0856	0.0972	0.0924
5	0.0828	0.0976	0.1192	0.0014	0.0350	0.0066	0.0248	0.1580	0.1212	0.1934	0.1770
10	0.2306	0.1668	0.3030	0.0000	0.0230	0.0016	0.0120	0.3758	0.2230	0.4314	0.4266
20	0.7906	0.3954	0.8214	0.0000	0.0178	0.0026	0.0080	0.9054	0.4708	0.8912	0.9394

Notes: Rejection rates for IVX-, Adaptive- and Combination-Tests for Sample Size$T = 250$, $\delta = -0.95$, u_t and v_t from standardized $t(3)$-distribution.

Table 2.8: Rejection rates: $T = 250$, $\delta = -0.95$ and $\chi^2(3)$-distribution

	$H_0: \beta = 0$ vs. $H_1: \beta \neq 0$			$H_0: \beta \geq 0$ vs. $H_1: \beta < 0$				$H_0: \beta \leq 0$ vs. $H_1: \beta > 0$			
b	IVX	ADA	Comb	IVX	ADA	Comb	$\theta = 0.5$	IVX	ADA	Comb	$\tilde{\theta} = 0.5$
					c = -5 ($\rho = 1.02$)						
-20	0.9896	0.9994	0.9996	0.9910	0.9992	0.9994	0.9992	0.0000	0.0002	0.0002	0.0002
-10	0.9742	0.9926	0.9880	0.9762	0.9948	0.9904	0.9944	0.0008	0.0000	0.0000	0.0002
-5	0.9492	0.9746	0.9726	0.9538	0.9806	0.9754	0.9810	0.0056	0.0002	0.0008	0.0038
-2	0.8906	0.9340	0.9388	0.8970	0.9438	0.9438	0.9462	0.0122	0.0008	0.0052	0.0116
0	0.0346	0.0744	0.0650	0.0040	0.0626	0.0330	0.0578	0.0616	0.0656	0.0872	0.0654
2	0.9564	0.9290	0.9688	0.0000	0.0008	0.0000	0.0004	0.9758	0.9408	0.9786	0.9862
5	0.9972	0.9766	0.9958	0.0000	0.0002	0.0000	0.0000	0.9996	0.9796	0.9978	0.9998
10	1.0000	0.9920	0.9996	0.0000	0.0000	0.0000	0.0000	1.0000	0.9934	0.9998	1.0000
20	1.0000	0.9984	1.0000	0.0000	0.0004	0.0004	0.0004	1.0000	0.9990	0.9996	0.9998
					c = 0 ($\rho = 1$)						
-20	0.7946	0.9912	0.9886	0.8478	0.9942	0.9946	0.9956	0.0004	0.0000	0.0000	0.0000
-10	0.3720	0.8616	0.8340	0.4500	0.9068	0.8736	0.9144	0.0022	0.0002	0.0004	0.0012
-5	0.0974	0.4914	0.4396	0.1306	0.5896	0.5230	0.5934	0.0146	0.0020	0.0048	0.0122
-2	0.0248	0.1466	0.1058	0.0138	0.2184	0.1366	0.2062	0.0446	0.0156	0.0336	0.0400
0	0.0494	0.0640	0.0838	0.0008	0.0486	0.0116	0.0368	0.1048	0.0678	0.1384	0.1176
2	0.1220	0.1742	0.3222	0.0000	0.0110	0.0002	0.0058	0.2434	0.2502	0.4470	0.3692
5	0.5748	0.5132	0.7806	0.0000	0.0014	0.0000	0.0004	0.7776	0.5962	0.8604	0.8966
10	0.9314	0.8482	0.9828	0.0000	0.0000	0.0000	0.0000	0.9780	0.8916	0.9902	0.9972
20	0.9962	0.9838	0.9996	0.0000	0.0000	0.0000	0.0000	0.9992	0.9894	0.9998	1.0000
					c = 10 ($\rho = 0.96$)						
-20	0.7156	0.9348	0.9614	0.7866	0.9592	0.9774	0.9746	0.0002	0.0000	0.0000	0.0000
-10	0.2948	0.5628	0.6502	0.3840	0.6656	0.7298	0.7124	0.0016	0.0004	0.0004	0.0018
-5	0.0872	0.2190	0.2560	0.1326	0.3108	0.3364	0.3352	0.0134	0.0066	0.0104	0.0124
-2	0.0420	0.0852	0.0930	0.0456	0.1252	0.1226	0.1344	0.0338	0.0220	0.0368	0.0336
0	0.0366	0.0568	0.0616	0.0170	0.0564	0.0390	0.0502	0.0644	0.0522	0.0778	0.0656
2	0.0514	0.0896	0.1240	0.0018	0.0240	0.0086	0.0198	0.1118	0.1270	0.1984	0.1392
5	0.1368	0.2286	0.3630	0.0000	0.0050	0.0006	0.0036	0.2534	0.3188	0.4892	0.3588
10	0.4906	0.5766	0.8434	0.0000	0.0002	0.0000	0.0000	0.7086	0.6748	0.9148	0.8896
20	0.9862	0.9316	0.9978	0.0000	0.0000	0.0000	0.0000	0.9964	0.9534	0.9990	0.9998
					c = 25 ($\rho = 0.9$)						
-20	0.5558	0.7946	0.8908	0.6572	0.8584	0.9250	0.8970	0.0000	0.0000	0.0000	0.0000
-10	0.1752	0.3548	0.4418	0.2632	0.4556	0.5394	0.4984	0.0048	0.0016	0.0014	0.0034
-5	0.0668	0.1330	0.1690	0.1014	0.1998	0.2326	0.2232	0.0170	0.0116	0.0126	0.0140
-2	0.0354	0.0752	0.0814	0.0420	0.1082	0.1070	0.1120	0.0316	0.0304	0.0386	0.0352
0	0.0318	0.0590	0.0678	0.0190	0.0608	0.0466	0.0634	0.0536	0.0526	0.0746	0.0614
2	0.0462	0.0734	0.0866	0.0084	0.0334	0.0214	0.0314	0.0846	0.0958	0.1278	0.1014
5	0.0850	0.1346	0.2004	0.0014	0.0118	0.0030	0.0094	0.1570	0.1962	0.2906	0.2090
10	0.2270	0.3500	0.5394	0.0000	0.0024	0.0000	0.0006	0.3638	0.4512	0.6602	0.5104
20	0.8020	0.7850	0.9742	0.0000	0.0004	0.0000	0.0000	0.9216	0.8524	0.9882	0.9880

Notes: Rejection rates for IVX-, Adaptive- and Combination-Tests for Sample Size$T = 250$, $\delta = -0.95$, u_t and v_t from standardized $\chi^2(3)$-distribution.

Multiple predictors

In order to deal with the situation where there is more than one persistent regressor, we extend our assumptions as follows. The regression model is

$$y_t = \delta + \boldsymbol{\beta}'\boldsymbol{x}_{t-1} + u_t, \qquad t = 2, \dots, T,$$

where we estimate the regression coefficients by

$$\hat{\delta}, \hat{\boldsymbol{\beta}} = \underset{\delta^*, \boldsymbol{\beta}^*}{\arg\min} \sum_{t=2}^{T} \mathcal{L}\left(y_t - \delta^* - \boldsymbol{\beta}^{*\prime}\boldsymbol{x}_{t-1}\right).$$

The regressors are taken for simplicity to have a restricted vector autoregressive structure,

$$\boldsymbol{x}_t = \boldsymbol{\mu}_x + \boldsymbol{\xi}_t \quad \text{and} \quad \boldsymbol{\xi}_t = \rho\boldsymbol{\xi}_{t-1} + \boldsymbol{v}_t,$$

with $\boldsymbol{v}_t = \sum_{j\geq 0} \mathbf{B}_j \boldsymbol{\nu}_{t-j}$ where $\sum_{j\geq 0} j \left\|\mathbf{B}_j\right\| < \infty$, $\sum_{j\geq 0} \mathbf{B}_j \neq 0$, and $(u_t, \boldsymbol{\nu}_t')'$ is serially independent, identically distributed series with finite kurtosis and $\boldsymbol{\nu}_t$ has zero mean.

A straightforward multivariate generalization of Proposition 2.1 then indicates that the usual Wald statistic, $\mathcal{W} = \hat{\boldsymbol{\beta}}'\widehat{\operatorname{Cov}\left(\hat{\boldsymbol{\beta}}\right)}\hat{\boldsymbol{\beta}}$, has the following limiting null distribution

$$\mathcal{W} \xrightarrow{d} \int_0^1 \bar{\boldsymbol{J}}_C'(s)\, \mathrm{d}\tilde{U}(s) \left(\int_0^1 \bar{\boldsymbol{J}}_C'(s)\, \bar{\boldsymbol{J}}_C(s)\, \mathrm{d}s\right)^{-1} \int_0^1 \bar{\boldsymbol{J}}_C(s)\, \mathrm{d}\tilde{U}(s)$$

where $\bar{\boldsymbol{J}}_C = \boldsymbol{J}_C(s) - \int_0^1 \boldsymbol{J}_C(r)\, \mathrm{d}r$ is the demeaned multivariate OU process driven by $\boldsymbol{V}$ with mean reversion matrix C. Project now $\tilde{U}$ on $\boldsymbol{V}$ to obtain with $\boldsymbol{\gamma} = \operatorname{Cov}(\tilde{u}_t, \boldsymbol{v}_t) = \operatorname{Cov}\left(\tilde{U}(1), \boldsymbol{V}(1)\right)$ and $\boldsymbol{\Xi}_\nu = \operatorname{Cov}(\boldsymbol{V}(1)) = \operatorname{Cov}(\boldsymbol{v}_t)$

$$\tilde{W} = \frac{1}{\sqrt{1 - \boldsymbol{\gamma}'\Xi_v^{-1}\boldsymbol{\gamma}}}\left(\tilde{U} - \boldsymbol{V}'\boldsymbol{\Xi}_\nu^{-1}\boldsymbol{\gamma}\right).$$

We then obtain the limiting distribution of $\mathcal{W}$ as the sum of three terms,

$$\frac{1}{1 - \boldsymbol{\gamma}'\boldsymbol{\Xi}_\nu^{-1}\boldsymbol{\gamma}} \int_0^1 \bar{\boldsymbol{J}}_C'(s)\, \mathrm{d}\tilde{W}(s) \left(\int_0^1 \bar{\boldsymbol{J}}_C'(s)\, \bar{\boldsymbol{J}}_C(s)\, \mathrm{d}s\right)^{-1} \int_0^1 \bar{\boldsymbol{J}}_C(s)\, \mathrm{d}\tilde{W}(s),$$

$$\frac{2}{\sqrt{1 - \boldsymbol{\gamma}'\boldsymbol{\Xi}_\nu^{-1}\boldsymbol{\gamma}}} \int_0^1 \bar{\boldsymbol{J}}_C'(s)\, \mathrm{d}\tilde{W}(s) \left(\int_0^1 \bar{\boldsymbol{J}}_C'(s)\, \bar{\boldsymbol{J}}_C(s)\, \mathrm{d}s\right)^{-1} \int_0^1 \bar{\boldsymbol{J}}_C(s)\, \mathrm{d}\boldsymbol{V}'(s)\, \boldsymbol{\Xi}_\nu^{-1}\boldsymbol{\gamma}$$

and

$$\boldsymbol{\gamma}'\boldsymbol{\Xi}_\nu^{-1} \int_0^1 \mathrm{d}\boldsymbol{V}(s)\, \bar{\boldsymbol{J}}_C'(s) \left(\int_0^1 \bar{\boldsymbol{J}}_C'(s)\, \bar{\boldsymbol{J}}_C(s)\, \mathrm{d}s\right)^{-1} \int_0^1 \bar{\boldsymbol{J}}_C(s)\, \mathrm{d}\boldsymbol{V}'(s)\, \boldsymbol{\Xi}_\nu^{-1}\boldsymbol{\gamma}.$$

Because of this structure it is difficult to correct for the second order bias in the same manner. Not even when $\boldsymbol{\gamma}_1$ and $\boldsymbol{\gamma}_2$ are proportional would the nonstandard parts vanish. One would have to correct the estimators prior to building the Wald statistic. We do not further pursue the topic here.

Bootstrapping subsample-based statistics

For subsamples of the form $[(s-\tau)T], \ldots, [sT]$, $s \in [\tau, 1]$ for some $\tau \in (0,1)$, consider the t statistics $t(s)$ computed subsample-wise, i.e.

$$t(s) = \frac{\hat{\beta}(s)}{s.e.\left(\hat{\beta}(s)\right)}$$

where

$$\hat{\delta}(s), \hat{\beta}(s) = \arg\min_{\delta^*, \beta^*} \sum_{t=[(s-\tau)T]+1}^{[sT]} \mathcal{L}\left(y_t - \delta^* - \beta^* x_{t-1}\right)$$

and

$$s.e.\left(\hat{\beta}(s)\right) = \sqrt{\left[B_T^{-1}(s) M_T(s) B_T^{-1}(s)\right]_{2,2}},$$

where

$$\begin{aligned} B_T(s) &= \begin{pmatrix} \sum_{t=[(s-\tau)T]+1}^{[sT]} \mathcal{L}''\left(\hat{u}_t\right) & \sum_{t=[(s-\tau)T]+1}^{[sT]} x_{t-1} \mathcal{L}''\left(\hat{u}_t\right) \\ \sum_{t=[(s-\tau)T]+1}^{[sT]} \mathcal{L}''\left(\hat{u}_t\right) & \sum_{t=[(s-\tau)T]+1}^{[sT]} x_{t-1}^2 \mathcal{L}''\left(\hat{u}_t\right) \end{pmatrix} \\ M_T(s) &= \begin{pmatrix} \sum_{t=[(s-\tau)T]+1}^{[sT]} \left(\mathcal{L}'\left(\hat{u}_t\right)\right)^2 & \sum_{t=[(s-\tau)T]+1}^{[sT]} x_{t-1} \left(\mathcal{L}'\left(\hat{u}_t\right)\right)^2 \\ \sum_{t=[(s-\tau)T]+1}^{[sT]} \left(\mathcal{L}'\left(\hat{u}_t\right)\right)^2 & \sum_{t=[(s-\tau)T]+1}^{[sT]} x_{t-1}^2 \left(\mathcal{L}'\left(\hat{u}_t\right)\right)^2 \end{pmatrix}. \end{aligned}$$

To motivate our proposed bootstrap, we discuss the high persistence case first. It is straightforward to show that, along the lines of Proposition 2.1,

$$t(s) \Rightarrow \frac{\int_{s-\tau}^{s} \left(J_c(r) - \frac{1}{\tau}\int_{s-\tau}^{s} J_c(r)\mathrm{d}r\right) \mathrm{d}\tilde{U}(r)}{\sqrt{\int_{s-\tau}^{s} \left(J_c(r) - \frac{1}{\tau}\int_{s-\tau}^{s} J_c(r)\mathrm{d}r\right)^2 \mathrm{d}r}}$$

under the null, such that we obtain the following limiting null distribution for the subsample-specific statistic combining two such t ratios computed under different loss functions:

$$\bar{t}(s) \Rightarrow \frac{\int_{s-\tau}^{s} \left(J_c(r) - \frac{1}{\tau}\int_{s-\tau}^{s} J_c(r)\mathrm{d}r\right) \mathrm{d}\tilde{W}(r)}{\sqrt{\int_{s-\tau}^{s} \left(J_c(r) - \frac{1}{\tau}\int_{s-\tau}^{s} J_c(r)\mathrm{d}r\right)^2 \mathrm{d}r}}.$$

(The required estimates e.g. of $\hat{\gamma}_{1,2}$ are also computed subsample-wise and can be shown to be uniformly consistent over all considered subsamples.) The continuous mapping theorem then implies

$$\max_{s\in[\tau,1]} \bar{t}(s) \Rightarrow \sup_{s\in[\tau,1]} \frac{\int_{s-\tau}^{s} J_c(r)\mathrm{d}\tilde{W}(r) - \left(\frac{1}{\tau}\int_{s-\tau}^{s} J_c(r)\mathrm{d}r\right)\int_{s-\tau}^{s} \mathrm{d}\tilde{W}(r)}{\sqrt{\int_{s-\tau}^{s} \left(J_c(r) - \frac{1}{\tau}\int_{s-\tau}^{s} J_c(r)\mathrm{d}r\right)^2 \mathrm{d}r}}. \tag{2.9}$$

While, for each s, the combined statistic $\bar{t}(s)$ is standard normal (see Proposition 2.1 again) and does not depend on c, the invariance argument cannot be made for the sequence of statistics $\bar{t}(s)$.

To see why, note that the numerator of the r.h.s. of (2.9) may be written as $\left(1\,;\,-\frac{1}{\tau}\int_{s-\tau}^{s} J_c(r)\mathrm{d}r\right)' \boldsymbol{X}(s)$ where

$$\boldsymbol{X}(s) = \begin{pmatrix} \int_0^s J_c(r)\mathrm{d}\tilde{W}(r) \\ \int_0^s \mathrm{d}\tilde{W}(r) \end{pmatrix}$$

is a stochastic volatility continuous-time process. Intuitively, since $\tilde{W}$ is independent of J_c, $\int_0^s J_c(r)\mathrm{d}\tilde{W}(r)$ will behave, conditional on J_c, like a *time-transformed* Brownian motion Davidson (see e.g. 1994, Section 29.4). Then, for any fixed s, the denominator correctly standardizes the ratio on the r.h.s. of (2.9). At the same time, we know, e.g. from Demetrescu et al. (2020, Remark 15), that deterministic time-varying volatility affects the limiting distributions of maxima of statistics computed over various subsamples even upon individual standardization of the numerators. Therefore, $\max_{s\in[\tau,1]} \bar{t}(s)$ will have a conditional distribution depending on J_c in general – and therefore a nonpivotal marginal distribution.

To obtain correctly sized inference, at least asymptotically, we resort to bootstrap critical values. It should be emphasized that the dependence of the conditional limiting null distributions of $\max_{s\in[\tau,1]} \bar{t}(s)$ on J_c prevents the use of standard bootstrap validity arguments where the limit of the bootstrap distribution is the same as the actual limiting distribution of interest. We may resort however to the conditional approach of Boswijk et al. (2019), who argue that, for a large variety of stochastic volatility processes, it is possible to bootstrap the correct limiting *conditional* distribution of specific statistics, such that the conditional probability of a type-I error converges to the desired significance level, and therefore the marginal size is controlled in the limit. In a nutshell, it can be shown that

$$\max_{s\in[\tau,1]} \bar{t}(s)\Big|\, x_1,\dots x_{T-1} \Rightarrow \sup_{s\in[\tau,1]} \frac{\int_{s-\tau}^{s} J_c(r)\mathrm{d}\tilde{W}(r) - \left(\frac{1}{\tau}\int_{s-\tau}^{s} J_c(r)\mathrm{d}r\right)\int_{s-\tau}^{s}\mathrm{d}\tilde{W}(r)}{\sqrt{\int_{s-\tau}^{s}\left(J_c(r) - \frac{1}{\tau}\int_{s-\tau}^{s} J_c(r)\mathrm{d}r\right)^2 \mathrm{d}r}}\Bigg|\, J_c,$$

while a suitable bootstrap delivers the same limiting conditional distribution,

$$\max_{s\in[\tau,1]} \bar{t}^*(s)\Big|\, x_1,\dots x_{T-1} \overset{p}{\Rightarrow} \sup_{s\in[\tau,1]} \frac{\int_{s-\tau}^{s} J_c(r)\mathrm{d}\tilde{W}(r) - \left(\frac{1}{\tau}\int_{s-\tau}^{s} J_c(r)\mathrm{d}r\right)\int_{s-\tau}^{s}\mathrm{d}\tilde{W}(r)}{\sqrt{\int_{s-\tau}^{s}\left(J_c(r) - \frac{1}{\tau}\int_{s-\tau}^{s} J_c(r)\mathrm{d}r\right)^2 \mathrm{d}r}}\Bigg|\, J_c,$$

with $\overset{p}{\Rightarrow}$ denoting weak convergence in probability. See also Georgiev et al. (2018, 2019) for applications to bootstrap specification tests in predictive regressions and Cavaliere and Georgiev (2020) for a more general treatment of validity of conditional bootstrap schemes.

To exploit this conditional validity argument, we resort to the following resampling scheme:

Algorithm 1 (Fixed-regressor bootstrap) *For each bootstrap replication $b = 1, \dots, B$, iterate:*

1. *For $t = 2, \dots, T$, generate $v_{t,b}^* \sim iid\big(0, \sigma^2\big)$ independent of x_{t-1}.*[5]

[5] We use a simplified scheme (and, in our application, we resort to the standard normal), but note that

2. Set $y^*_{t,b} = v^*_{t,b}$ *and* $x^*_{t-1,b} = x_{t-1,b}$.

3. Compute the subsample statistics $\bar{t}^*_b(s)$ *and the max,* $\mathcal{T}^*_b = \max_{s\in[\tau,1]} \bar{t}^*(s)$.

Use the $(1-\alpha)$*-quantile of the empirical distribution of* $\mathcal{T}^*_1, \ldots, \mathcal{T}^*_B$ *as critical value for* $\max_{s\in[\tau,1]} \bar{t}(s)$.

The asymptotic size of the bootstrap test will be controlled in the high persistence case as outlined above. In the low persistence case, the bootstrap is a standard fixed-regressor bootstrap which poses no difficulties and we take its validity for granted. Importantly, the above bootstrap scheme does not require any user-input regarding the type of persistence of the data generating process.

We note furthermore that Step 2 of the algorithm imposes the null hypothesis of no predictability, so we expect power to be nontrivial.

Finally, one may bootstrap other functionals of $\bar{t}(s)$ such as $\min_{s\in[\tau,1]} \bar{t}(s)$ or $\max_{s\in[\tau,1]} |\bar{t}(s)|$ in an entirely analogous fashion. To extend the argument to the combination test, which relies on a linear combination of the $\bar{t}_{ADA}$ statistic with the IVX statistic, we note that a) the optimal weights for $\bar{t}_{ADA}$ are consistently estimated, and b) the IVX test statistic can be bootstrapped subsample-wise using a fixed-regressor bootstrap too; see Demetrescu et al. (2021).

Boswijk et al. (2019) resort to a wild bootstrap due to the fact that the volatility process is latent in their case, while x_{t-1}, whose weak limit J_c plays the role of the time-varying volatility in $\boldsymbol{X}$, is observed.

Auxiliary results and proofs

Lemma 2.1 *Under the assumptions of Proposition 2.1, the following properties hold true as* $T \to \infty$.

1. *For* $\tilde{u}_{ti} = \mathcal{L}_i\left(u_t - \delta_{\mathcal{L}_i}\right)$, $i = 1, 2$, *and two different loss functions* $\mathcal{L}_1$ *and* $\mathcal{L}_2$ *obeying Assumption 2.1, we have*

$$\frac{1}{\sqrt{T}} \sum_{t=1}^{[sT]} \begin{pmatrix} \tilde{u}_{t1} \\ \tilde{u}_{t2} \\ \nu_t \end{pmatrix} \Rightarrow \begin{pmatrix} \sigma_{\tilde{u}_1} \tilde{U}_1(s) \\ \sigma_{\tilde{u}_2} \tilde{U}_2(s) \\ \sigma_v V(s) \end{pmatrix},$$

where the covariance matrix of the standard Wiener processes $\tilde{U}_1$, $\tilde{U}_2$ *and* V *is*

$$\operatorname{Cov} \begin{pmatrix} \tilde{U}_1(1) \\ \tilde{U}_2(1) \\ V(1) \end{pmatrix} = \operatorname{corr}\left(\begin{pmatrix} \tilde{u}_{t1} \\ \tilde{u}_{t2} \\ v_t \end{pmatrix}\right) = \begin{pmatrix} 1 & \varpi & \gamma_1 \\ \varpi & 1 & \gamma_2 \\ \gamma_1 & \gamma_2 & 1 \end{pmatrix}$$

with $\varpi = \omega\sqrt{\left(1-\gamma_1^2\right)\left(1-\gamma_2^2\right)} + \gamma_1\gamma_2$.

2. *Furthermore,*

$$\frac{1}{T}\sum_{t=2}^{T} x_{t-1}\tilde{u}_t \Rightarrow \lambda\sigma_\nu\sigma_{\tilde{u}} \int_0^1 J_c(s)\,\mathrm{d}\tilde{U}(s).$$

3. *For* $k = 1, 2$,

$$\frac{1}{T^{1+k/2}}\sum_{t=2}^{T} x_{t-1}^k \mathcal{L}''\left(u_t - \delta_{\mathcal{L}}\right) \Rightarrow \kappa\lambda^k\sigma_\nu^k \int_0^1 J_c^k(s)\,\mathrm{d}s.$$

4. *For* $k = 1, 2$,

$$\frac{1}{T^{1+k/2}}\sum_{t=2}^{T} x_{t-1}^k \tilde{u}_t^2 \Rightarrow \sigma_{\tilde{u}}^2\lambda^k\sigma_v^k \int_0^1 J_c^k(s)\,\mathrm{d}s.$$

5. *For any* $\tilde{\delta} = \delta + \delta_{\mathcal{L}} + o_p(1)$, $\tilde{\beta} = \beta + o_p\left(T^{-1/2}\right)$, *and* $k = 0, 1, 2$, *it holds that*

$$\frac{1}{T^{1+k/2}}\sum_{t=2}^{T} x_{t-1}^k \mathcal{L}''\left(y_t - \tilde{\delta} - \tilde{\beta}x_{t-1}\right) = \frac{1}{T^{1+k/2}}\sum_{t=2}^{T} x_{t-1}^k \mathcal{L}''\left(u_t - \delta_{\mathcal{L}}\right) + o_p(1).$$

6. *For any* $\tilde{\delta} = \delta + \delta_{\mathcal{L}} + o_p(1)$, $\tilde{\beta} = \beta + o_p\left(T^{-1/2}\right)$, *and* $k = 0, 1, 2$, *it holds that*

$$\frac{1}{T^{1+k/2}}\sum_{t=2}^{T} x_{t-1}^k \left(\mathcal{L}'\left(y_t - \tilde{\delta} - \tilde{\beta}x_{t-1}\right)\right)^2 = \frac{1}{T^{1+k/2}}\sum_{t=2}^{T} x_{t-1}^k \tilde{u}_t^2 + o_p(1).$$

7. *For* $\hat{\delta}$ *and* $\hat{\beta}$ *from* (2.4), *it holds that*

$$\left(\hat{\delta}, \hat{\beta}\right)' \xrightarrow{p} \left(\delta + \delta_{\mathcal{L}}, \beta\right)',$$

as $T \to \infty$ *such that*

$$\hat{\beta} - \beta = o_p\left(T^{-1/2}\right).$$

Proof of Lemma 2.1

1. The correlation ϖ of $\tilde{u}_{t1}$ and $\tilde{u}_{t2}$ is easily related to the correlation ω of $\tilde{u}_{t1} - (\gamma_1 \sigma_{\tilde{u}_1}/\sigma_\nu)\,\nu_t$ and $\tilde{u}_{t2} - (\gamma_2 \sigma_{\tilde{u}_2}/\sigma_\nu)\,\nu_t$, where $\mathrm{Var}\left(\tilde{u}_{ti} - (\gamma_i \sigma_{\tilde{u}_i}/\sigma_\nu)\,\nu_t\right) = \sigma^2_{\tilde{u}_i}\left(1-\gamma_i^2\right)$ as required. The weak convergence follows immediately and we omit the details.

2. Given the independence of $\tilde{u}_t$ of $u_{t-1}, v_{t-1}, \ldots$, the result follows from the weak convergence of $T^{-1/2}\xi_{[sT]}$ (and thus of $T^{-1/2}x_{[sT]}$) jointly with part 1 of this Lemma; see e.g. Kurtz and Protter (1991).

3. Write with $\kappa = \mathrm{E}\left(\mathcal{L}''\left(u_t - \delta_{\mathcal{L}}\right)\right)$

$$\frac{1}{T^{1+k/2}}\sum_{t=2}^{T} x_{t-1}^k \mathcal{L}''\left(u_t - \delta_{\mathcal{L}}\right) = \kappa \frac{1}{T^{1+k/2}}\sum_{t=2}^{T} x_{t-1}^k - \frac{1}{T^{1+k/2}}\sum_{t=2}^{T} x_{t-1}^k \left(\mathcal{L}''\left(u_t - \delta_{\mathcal{L}}\right) - \kappa\right),$$

and the result follows if the second summand on the r.h.s. vanishes as $T \to \infty$. To prove this, let $\tilde{z}_t = z_t - \mathrm{E}\left(z_t\right)$ and note that, since z_t is iid, $\mathrm{E}\left(\tilde{z}_t | \tilde{z}_{t-m}, \tilde{z}_{t-m-1}, \ldots\right) \xrightarrow{p} 0$ as $m \to \infty$. Furthermore, since $\mathcal{L}''$ is piecewise constant, $\tilde{z}_t$ is uniformly integrable and thus $\mathrm{E}\left(\left|\mathrm{E}\left(\tilde{z}_t | \tilde{z}_{t-m}, \tilde{z}_{t-m-1}, \ldots\right)\right|\right) \to 0$. Then, we obtain the desired

$$\frac{1}{T^{1+k/2}}\sum_{t=2}^{T} x_{t-1}^k \tilde{z}_t \xrightarrow{p} 0$$

by applying Theorem 3.3 of Hansen (1992).

4. Analogous to the proof of part 3 and omitted.

5. The proof is straightforward when $\alpha = 0.5$, so we focus on $\alpha \neq 0.5$, where we recall that $\mathcal{L}''$ is piecewise constant but discontinuous at 0. Therefore, $\mathcal{L}''\left(y_t - \tilde{\delta} - \tilde{\beta}x_{t-1}\right) = \mathcal{L}''\left(u_t - \delta_{\mathcal{L}}\right)$ if the two arguments have the same sign.
Let $\zeta_{t,T} = \tilde{\delta} - \left(\delta + \delta_{\mathcal{L}}\right) + \left(\tilde{\beta} - \beta\right) x_{t-1}$ and note that $\xi_{t,T} \xrightarrow{p} 0$ uniformly in t (this is because $\sup_t |x_{t-1}| = O_p\left(\sqrt{T}\right)$ thanks to the weak convergence of $T^{-1/2}x_{[sT]}$ to the pathwise continuous J_c). Now, $y_t - \tilde{\delta} - \tilde{\beta}x_{t-1}$ and $u_t - \delta_{\mathcal{L}}$ can only have a different sign when $|\zeta_{t,T}| \geq |u_t - \delta_{\mathcal{L}}|$, we have that $\mathcal{L}''\left(y_t - \tilde{\delta} - \tilde{\beta}x_{t-1}\right) = \mathcal{L}''\left(u_t - \beta_0^{\mathcal{L}}\right)$ whenever $|\zeta_{t,T}| < |u_t - \delta_{\mathcal{L}}|$. Write therefore

$$\begin{aligned} \mathcal{L}''\left(y_t - \tilde{\delta} - \tilde{\beta}x_{t-1}\right) &= \mathcal{L}''\left(u_t - \delta_{\mathcal{L}} - \zeta_{t,T}\right)\left(\mathbf{1}\left(|\zeta_{t,T}| \geq |u_t - \delta_{\mathcal{L}}|\right) + \mathbf{1}\left(|\xi_T| < |u_t - \delta_{\mathcal{L}}|\right)\right), \\ &= \mathcal{L}''\left(u_t - \delta_{\mathcal{L}}\right) + \left(\mathcal{L}''\left(y_t - \tilde{\delta} - \tilde{\beta}x_{t-1}\right) - \mathcal{L}''\left(u_t - \delta_{\mathcal{L}}\right)\right)\mathbf{1}\left(|\zeta_{t,T}| \geq |u_t - \delta_{\mathcal{L}}|\right) \end{aligned}$$

and, to establish the desired result, it suffices to show that

$$\begin{aligned} \frac{1}{T^{1+k/2}}\sum_{t=2}^{T} x_{t-1}^k \mathcal{L}''\left(y_t - \tilde{\delta} - \tilde{\beta}x_{t-1}\right)\mathbf{1}\left(|\zeta_{t,T}| \geq \left|u_t - \beta_0^{\mathcal{L}}\right|\right) &\xrightarrow{p} 0, \\ \frac{1}{T^{1+k/2}}\sum_{t=2}^{T} x_{t-1}^k \mathcal{L}''\left(u_t - \delta_{\mathcal{L}}\right)\mathbf{1}\left(|\zeta_{t,T}| \geq \left|u_t - \beta_0^{\mathcal{L}}\right|\right) &\xrightarrow{p} 0. \end{aligned}$$

Since $\mathcal{L}''$ is bounded, we have as upper bound for the respective absolute values

$$\sup_t \frac{|x_{t-1}^k|}{T^{k/2}} \sup_u \left|\mathcal{L}''(u)\right| \cdot \frac{1}{T}\sum_{t=2}^{T} \mathbf{1}\left(|\zeta_{t,T}| \geq \left|u_t - \beta_0^{\mathcal{L}}\right|\right),$$

where $\sup_t |x_{t-1}| = O_p\left(\sqrt{T}\right)$. Given that $\mathrm{E}\left(\mathbf{1}\left(|\zeta_{t,T}| \geq \left|u_t - \beta_0^{\mathcal{L}}\right|\right)\right) = \Pr\left(\left|u_t - \beta_0^{\mathcal{L}}\right| \leq |\zeta_{t,T}|\right)$ vanishes when u_t does not have an atom at $\delta_{\mathcal{L}}$, it follows as required that $\frac{1}{T}\sum_{t=2}^{T} \mathbf{1}\left(|\zeta_{t,T}| \geq \left|u_t - \beta_0^{\mathcal{L}}\right|\right) \xrightarrow{p} 0$ thanks to Markov's inequality.

6. Analogous to the proof of part 5 and omitted.

7. We establish consistency using a theorem of the type "if the target function converges uniformly in probability to deterministic function, minimized at the true values of the parameters, then argmin estimators are consistent" (see Chapter 4 of Amemyia, 1985). Assume for simplicity that δ is known to belong to a compact set Δ;
Let first $\beta^* = \beta$. Then,

$$\frac{1}{T}\sum_{t=2}^{T} \mathcal{L}\left(y_t - \delta^* - \beta^* x_{t-1}\right) = \frac{1}{T}\sum_{t=2}^{T} \mathcal{L}\left(u_t + \delta - \delta^*\right) \xrightarrow{p} \mathrm{E}\left(\mathcal{L}\left(u_t + \delta - \delta^*\right)\right),$$

pointwise in δ^*, due to the iid assumption on u_t and the finiteness of the expected loss. Consider then the case $\beta^* \neq \beta$, where we obtain

$$\frac{1}{T}\sum_{t=2}^{T} \mathcal{L}\left(y_t - \delta^* - \beta^* x_{t-1}\right) = \frac{1}{T}\sum_{t=2}^{T} \mathcal{L}\left(u_t + \delta - \delta^* + \left(\beta - \beta^*\right) x_{t-1}\right).$$

Since the loss function $\mathcal{L}$ is continuous and homogeneous of order 2, $\sup_t |u_t| = o_p\left(\sqrt{T}\right)$, and $\delta \in \Delta$, the CMT leads to

$$\begin{aligned} \frac{1}{T^2}\sum_{t=2}^{T} \mathcal{L}\left(y_t - \delta^* - \beta^* x_{t-1}\right) &= \frac{1}{T}\sum_{t=2}^{T} \mathcal{L}\left(\frac{u_t + \delta - \delta^*}{\sqrt{T}} + \left(\beta - \beta^*\right)\frac{x_{t-1}}{\sqrt{T}}\right) \\ &\Rightarrow \int_0^1 \mathcal{L}\left(\left(\beta - \beta^*\right)\lambda\sigma_\nu J_c(s)\right) \mathrm{d}s \end{aligned}$$

which, given homogeneity of $\mathcal{L}$, satisfies

$$\int_0^1 \mathcal{L}\left(\left(\beta - \beta^*\right)\lambda\sigma_\nu J_c(s)\right) \mathrm{d}s \geq \left(\beta - \beta^*\right)^2 \lambda^2\sigma_\nu^2 \int_0^1 \mathcal{L}\left(J_c(s)\right) \mathrm{d}s.$$

Therefore,

$$\frac{1}{T}\sum_{t=2}^{T} \mathcal{L}\left(y_t - \delta^* - \beta^* x_{t-1}\right) \xrightarrow{p} \infty \quad \forall\, \beta^* \neq \beta,$$

and, since $\mathrm{E}\left(\mathcal{L}\left(u_t + \delta - \delta^*\right)\right)$ is finite, the target function is minimized with probability approaching 1 at β as $T \to \infty$. Therefore, $\hat{\beta} \xrightarrow{p} \beta$ irrespective of the behavior of $\hat{\delta}$.
To analyze $\hat{\delta}$, pointwise convergence on the compact Δ and convexity of the target function imply uniform convergence Andersen and Gill (1982, Lemma II.1) to the argmin of

$$\mathrm{E}\left(\mathcal{L}\left(u_t + \delta - \delta^*\right)\right) = \mathrm{E}\left(\mathcal{L}\left(u_t - \delta_{\mathcal{L}} + \delta + \delta_{\mathcal{L}} - \delta^*\right)\right).$$

But the expectation is minimized for $\delta+\delta_{\mathcal{L}}-\delta^*=0$ given the definition of $\delta_{\mathcal{L}}$, so $\widehat{\delta}\xrightarrow{p}\delta+\delta_{\mathcal{L}}$ as required.
To establish the desired bound for the convergence rate of $\widehat{\beta}$, consider the sequence $\beta^*=\beta+b/\sqrt{T}$ and let w.l.o.g. $\delta^*=\delta-\delta_{\mathcal{L}}$.[6] Using a Taylor expansion around β, it follows that

$$\begin{aligned}\frac{1}{T}\sum_{t=2}^{T}\mathcal{L}\left(y_t-\delta^*-\beta^*x_{t-1}\right) &= \frac{1}{T}\sum_{t=2}^{T}\mathcal{L}\left(u_t-\delta_{\mathcal{L}}\right)+\frac{b}{T}\sum_{t=2}^{T}\mathcal{L}'\left(u_t-\delta_{\mathcal{L}}\right)\frac{x_{t-1}}{\sqrt{T}}\\ &\quad+\frac{b^2}{T}\sum_{t=2}^{T}\mathcal{L}''\left(u_t-\delta_{\mathcal{L}}-\frac{b^*}{\sqrt{T}}x_{t-1}\right)\left(\frac{x_{t-1}}{\sqrt{T}}\right)^2,\end{aligned}$$

where $0\le b^*\le b$. The first term on the r.h.s. converges to $\mathrm{E}\left(\mathcal{L}\left(u_t-\delta_{\mathcal{L}}\right)\right)$ which is the minimum of the target function; the second converges to zero in probability thanks to part 2 of this Lemma. For the third, note that, due to the convexity of $\mathcal{L}$, $\mathcal{L}''$ is bounded away from zero, so there exists $C>0$ such that

$$\frac{b^2}{T}\sum_{t=2}^{T}\mathcal{L}''\left(u_t-\delta_{\mathcal{L}}-\frac{b^*}{\sqrt{T}}x_{t-1}\right)\left(\frac{x_{t-1}}{\sqrt{T}}\right)^2\ge\frac{Cb^2}{T}\sum_{t=2}^{T}\left(\frac{x_{t-1}}{\sqrt{T}}\right)^2,$$

where $T^{-1}\sum_{t=2}^{T}\left(\frac{x_{t-1}}{\sqrt{T}}\right)^2\Rightarrow\int_0^1 J_c^2(s)\,\mathrm{d}s$ which is positive w.p.1. Hence, unless $b=0$, the minimum of the target function is not achieved under $\beta_1^*=\beta_1+b/\sqrt{T}$ and $\widehat{\beta}_1$ must converge at a rate faster than $T^{-1/2}$, as required.

Proof of Proposition 2.1

We shall derive the limiting distribution of the M-based t-type statistic under the relevant local alternative. The result for $\bar{t}$ then follows thanks to the joint convergence in Lemma 2.1.1.

Take the Taylor expansion with rest term in differential form of the two first-order conditions of the minimum problem in (2.4) around $(\delta+\delta_{\mathcal{L}},0)'$ and evaluate at $\left(\hat{\delta},\hat{\beta}\right)'$; with $\hat{u}_t=y_t-\hat{\delta}-\hat{\beta}x_{t-1}$ we have

$$\begin{aligned}-\begin{pmatrix}\sum\mathcal{L}'\left(\hat{u}_t\right)\\ \sum x_{t-1}\mathcal{L}'\left(\hat{u}_t\right)\end{pmatrix} &= -\begin{pmatrix}\sum\mathcal{L}'\left(y_t-\delta-\delta_{\mathcal{L}}\right)\\ \sum x_{t-1}\mathcal{L}'\left(y_t-\delta-\delta_{\mathcal{L}}\right)\end{pmatrix}\\ &\quad+\begin{pmatrix}\sum\mathcal{L}''\left(y_t-\tilde{\delta}_1-\tilde{\beta}_1x_{t-1}\right) & \sum x_{t-1}\mathcal{L}''\left(y_t-\tilde{\delta}_1-\tilde{\beta}_1x_{t-1}\right)\\ \sum x_{t-1}\mathcal{L}''\left(y_t-\tilde{\delta}_2-\tilde{\beta}_2x_{t-1}\right) & \sum x_{t-1}^2\mathcal{L}''\left(y_t-\tilde{\delta}_2-\tilde{\beta}_2x_{t-1}\right)\end{pmatrix}\begin{pmatrix}\hat{\delta}-\delta-\delta_{\mathcal{L}}\\ \hat{\beta}\end{pmatrix}\end{aligned}$$

where $\tilde{\delta}_{1,2}$ lie between δ and $\hat{\delta}$, and and $\tilde{\beta}_{1,2}$ lie between 0 and $\hat{\beta}$, respectively. (We use $\sum$ as shorthand for $\sum_{t=2}^{T}$.) Evaluated at $\left(\hat{\delta},\hat{\beta}\right)'$, the gradient is 0, so

$$\begin{aligned}\begin{pmatrix}\hat{\delta}-\delta-\delta_{\mathcal{L}}\\ \hat{\beta}\end{pmatrix} &= \begin{pmatrix}\sum\mathcal{L}''\left(y_t-\tilde{\delta}_1-\tilde{\beta}_1x_{t-1}\right) & \sum x_{t-1}\mathcal{L}''\left(y_t-\tilde{\delta}_1-\tilde{\beta}_1x_{t-1}\right)\\ \sum x_{t-1}\mathcal{L}''\left(y_t-\tilde{\delta}_2-\tilde{\beta}_2x_{t-1}\right) & \sum x_{t-1}^2\mathcal{L}''\left(y_t-\tilde{\delta}_2-\tilde{\beta}_2x_{t-1}\right)\end{pmatrix}^{-1}\\ &\quad\times\begin{pmatrix}\sum\mathcal{L}'\left(y_t-\delta-\delta_{\mathcal{L}}\right)\\ \sum x_{t-1}\mathcal{L}'\left(y_t-\delta-\delta_{\mathcal{L}}\right)\end{pmatrix}.\end{aligned}$$

[6]The argument can easily be seen to apply for any other value of δ^*, which essentially works for the same reason $\hat{\beta}\xrightarrow{p}\beta$ irrespective of the behavior of $\hat{\delta}$.

Since $\hat{\beta}-\beta$ is $o_p\left(T^{-1/2}\right)$ from Lemma 2.1.7 and $\beta = {}^b\!/_T$, it must hold that $\tilde{\beta}_{1,2} = o_p\left(T^{-1/2}\right)$; also, $\tilde{\delta}-\delta-\delta_{\mathcal{L}} = o_p(1)$. Using Lemma 2.1.5 we may then conclude that

$$\begin{pmatrix} \frac{1}{T}\sum \mathcal{L}''\left(y_t-\tilde{\delta}_1-\tilde{\beta}_1 x_{t-1}\right) & \frac{1}{T^{3/2}}\sum x_{t-1}\mathcal{L}''\left(y_t-\tilde{\delta}_1-\tilde{\beta}_1 x_{t-1}\right) \\ \frac{1}{T^{3/2}}\sum x_{t-1}\mathcal{L}''\left(y_t-\tilde{\delta}_2-\tilde{\beta}_2 x_{t-1}\right) & \frac{1}{T^2}\sum x_{t-1}^2\mathcal{L}''\left(y_t-\tilde{\delta}_2-\tilde{\beta}_2 x_{t-1}\right) \end{pmatrix}^{-1}$$
$$= \begin{pmatrix} \frac{1}{T}\sum \mathcal{L}''\left(u_t-\delta_{\mathcal{L}}\right) & \frac{1}{T^{3/2}}\sum x_{t-1}\mathcal{L}''\left(u_t-\delta_{\mathcal{L}}\right) \\ \frac{1}{T^{3/2}}\sum x_{t-1}\mathcal{L}''\left(u_t-\delta_{\mathcal{L}}\right) & \frac{1}{T^2}\sum x_{t-1}^2\mathcal{L}''\left(u_t-\delta_{\mathcal{L}}\right) \end{pmatrix}^{-1} + o_p(1)$$

where the inverse on the r.h.s. exists with probability approaching 1. Standard matrix algebra further indicates that

$$T\hat{\beta} = \frac{\frac{1}{T^2}A_{1T}}{\frac{1}{T^3}B_{1T}} + o_p(1),$$

with

$$\begin{aligned} A_{1T} &= \left(\sum \mathcal{L}''\left(u_t-\delta_{\mathcal{L}}\right)\right)\left(\sum x_{t-1}\mathcal{L}'\left(u_t-\delta_{\mathcal{L}}-\frac{b}{T}x_{t-1}\right)\right) \\ &\quad - \left(\sum x_{t-1}\mathcal{L}''\left(u_t-\delta_{\mathcal{L}}\right)\right)\left(\sum \mathcal{L}'\left(u_t-\delta_{\mathcal{L}}-\frac{b}{T}x_{t-1}\right)\right) \\ B_{1T} &= \left(\sum \mathcal{L}''\left(u_t-\delta_{\mathcal{L}}\right)\right)\left(\sum x_{t-1}^2\mathcal{L}''\left(u_t-\delta_{\mathcal{L}}\right)\right) - \left(\sum x_{t-1}\mathcal{L}''\left(u_t-\delta_{\mathcal{L}}\right)\right)^2. \end{aligned}$$

Now, resorting again to a Taylor series expansion,

$$\frac{1}{\sqrt{T}}\sum \mathcal{L}'\left(u_t-\delta_{\mathcal{L}}-\frac{b}{T}x_{t-1}\right) = \frac{1}{\sqrt{T}}\sum \mathcal{L}'\left(u_t-\delta_{\mathcal{L}}\right) - \frac{1}{\sqrt{T}}\sum \frac{b}{T}x_{t-1}\mathcal{L}''\left(u_t-\delta_{\mathcal{L}}-\tilde{\delta}\right)$$

where $\tilde{\delta}$ lies between 0 and $\frac{b}{T}x_{t-1}$. With $\sup_t |x_{t-1}| = O_p\left(\sqrt{T}\right)$, following from the weak convergence on $[0,1]$ of the levels of $x_{[sT]}$ to the pathwise continuous process J_c, we obtain from Lemma 2.1.5 that

$$\frac{1}{\sqrt{T}}\sum \mathcal{L}'\left(u_t-\delta_{\mathcal{L}}-\frac{b}{T}x_{t-1}\right) = \frac{1}{\sqrt{T}}\sum \tilde{u}_t - \frac{b}{T^{3/2}}\sum x_{t-1}\mathcal{L}''\left(u_t-\delta_{\mathcal{L}}\right) + o_p(1)$$

and analogously

$$\frac{1}{T}\sum x_{t-1}\mathcal{L}'\left(u_t-\delta_{\mathcal{L}}-\frac{b}{T}x_{t-1}\right) = \frac{1}{T}\sum x_{t-1}\tilde{u}_t - \frac{b}{T^2}\sum x_{t-1}^2\mathcal{L}''\left(u_t-\delta_{\mathcal{L}}\right) + o_p(1).$$

The standard error of $\hat{\beta}$ is easily checked to be

$$s.e.\left(\hat{\beta}\right) = \sqrt{M_{1T}B_{1T}^{-2}} \qquad \text{s.t.} \qquad t = \frac{\frac{1}{T^2}A_{1T}}{\sqrt{\frac{1}{T^4}M_{1T}}}$$

with B_{1T} given above and

$$\begin{aligned} M_{1T} &= \left(\sum \mathcal{L}''\left(\hat{u}_t\right)\right)^2 \sum x_{t-1}^2\left(\mathcal{L}'\left(\hat{u}_t\right)\right)^2 + \left(\sum x_{t-1}\mathcal{L}''\left(\hat{u}_t\right)\right)^2 \sum \left(\mathcal{L}'\left(\hat{u}_t\right)\right)^2 \\ &\quad - 2\sum \mathcal{L}''\left(\hat{u}_t\right)\sum x_{t-1}\mathcal{L}''\left(\hat{u}_t\right)\sum x_{t-1}\left(\mathcal{L}'\left(\hat{u}_t\right)\right)^2. \end{aligned}$$

It is not difficult to show that $\hat{u}_t = u_t - \delta_{\mathcal{L}} + o_p(1)$, such that Lemma 2.1 parts 5 and 6 imply

$$\begin{aligned} M_{1T} &= \left(\sum \mathcal{L}''\left(u_t-\delta_{\mathcal{L}}\right)\right)^2 \sum x_{t-1}^2 \left(\mathcal{L}'\left(u_t-\delta_{\mathcal{L}}-\frac{b}{T}x_{t-1}\right)\right)^2 \\ &\quad + \left(\sum x_{t-1}\mathcal{L}''\left(u_t-\delta_{\mathcal{L}}\right)\right)^2 \sum \left(\mathcal{L}'\left(u_t-\delta_{\mathcal{L}}-\frac{b}{T}x_{t-1}\right)\right)^2 \\ &\quad -2\sum \mathcal{L}''\left(u_t-\delta_{\mathcal{L}}\right)\sum x_{t-1}\mathcal{L}''\left(u_t-\delta_{\mathcal{L}}\right)\sum x_{t-1}\left(\mathcal{L}'\left(u_t-\delta_{\mathcal{L}}-\frac{b}{T}x_{t-1}\right)\right)^2 + o_p\left(T^4\right) \\ &= \left(\sum \mathcal{L}''\left(u_t-\delta_{\mathcal{L}}\right)\right)^2 \sum x_{t-1}^2\tilde{u}_t^2 + \left(\sum x_{t-1}\mathcal{L}''\left(u_t-\delta_{\mathcal{L}}\right)\right)^2\sum \tilde{u}_t^2 \\ &\quad -2\sum \mathcal{L}''\left(u_t-\delta_{\mathcal{L}}\right)\sum x_{t-1}\mathcal{L}''\left(u_t-\delta_{\mathcal{L}}\right)\sum x_{t-1}\tilde{u}_t^2 + o_p\left(T^4\right). \end{aligned}$$

Thus,

$$\begin{aligned} \frac{1}{T^2}A_{1T} &= \left(\frac{1}{T}\sum \mathcal{L}''\left(u_t-\delta_{\mathcal{L}}\right)\right)\left(\frac{1}{T}\sum x_{t-1}\tilde{u}_t - \frac{b}{T^2}\sum x_{t-1}^2\mathcal{L}''\left(u_t-\delta_{\mathcal{L}}\right)\right) \\ &\quad -\left(\frac{1}{T^{3/2}}\sum x_{t-1}\mathcal{L}''\left(u_t-\delta_{\mathcal{L}}\right)\right)\left(\frac{1}{\sqrt{T}}\sum \tilde{u}_t - \frac{b}{T^{3/2}}\sum x_{t-1}\mathcal{L}''\left(u_t-\delta_{\mathcal{L}}\right)\right) + o_p\left(1\right) \end{aligned}$$

and

$$\begin{aligned} \frac{1}{T^4}M_{1T} &= \left(\frac{1}{T}\sum \mathcal{L}''\left(u_t-\delta_{\mathcal{L}}\right)\right)^2 \frac{1}{T^2}\sum x_{t-1}^2\tilde{u}_t^2 + \left(\frac{1}{T^{3/2}}\sum x_{t-1}\mathcal{L}''\left(u_t-\delta_{\mathcal{L}}\right)\right)^2\frac{1}{T}\sum \tilde{u}_t^2 \\ &\quad -2\frac{1}{T}\sum \mathcal{L}''\left(u_t-\delta_{\mathcal{L}}\right)\frac{1}{T^{3/2}}\sum x_{t-1}\mathcal{L}''\left(u_t-\delta_{\mathcal{L}}\right)\frac{1}{T^{3/2}}\sum x_{t-1}\tilde{u}_t^2 + o_p\left(1\right), \end{aligned}$$

and the result follows with Lemma 2.1 items 2, 3 and 4.

Chapter 3

Nonlinear Predictability of Stock Returns? Parametric Versus Nonparametric Inference in Predictive Regressions

Coauthored by: Matei Demetrescu

ABSTRACT:
Nonparametric test procedures in predictive regressions have χ^2 limiting null distributions under both low and high regressor persistence, but low local power compared to misspecified linear predictive regressions. We argue that IV inference is better suited (in terms of local power) for analyzing additive predictive models with uncertain predictor persistence. Then, a two-step procedure is proposed for out-of-sample predictions. For the current estimation window, one first tests for predictability; in case of a rejection, one predicts using a nonlinear regression model, otherwise the historic average of the stock returns is used. This two-step approach performs better than competitors (though not by a large margin) in a pseudo-out-of-sample prediction exercise for the S&P 500.

Key words: Chi-square distribution, Endogeneity, Nonlinear regression function, Predictive Regression, Time-varying variance, Unknown persistence

JEL classification: C12 (Hypothesis Testing), C22 (Time-Series Models), G17 (Financial Forecasting and Simulation) https://doi.org/10.1080/07350015.2020.1819821

Chapter 4

Predicting Stock Returns with Regression Trees: Nonlinearities and Predictor Selection

4.1 Introduction

Predicting stock returns has been a central issue in financial economics for many years. For a long time, the prevailing belief was that stock returns follow a random walk and are therefore unpredictable (c.f Fama (1995)). Although many people now believe that stock returns are predictable to a certain extent, predictions are still a major challenge. Stock market data mostly exhibits a low signal-to-noise ratio with estimated (slope) parameters often being close to zero. Furthermore, one has to cope with nonlinear and nonstationary market dynamics.

As the computational power of computers progresses, machine learning is becoming an increasingly growing area in the prediction of market data. Specifically regression tree models are a prominent machine learning technique to analyze high-dimensional data sets. Furthermore, they are able to cope with nonlinear and non-continuous DGPs and can compute precise predictions for these cases. Also, no strong parametric assumptions such as monotonicity or linearity are required to employ trees and methods thereof.

In this article, we explore the potential of tree based methods in predicting stock returns. We conduct a comparative analysis of various methods. One advantage of trees is a built-in model selection through their splits. Model averaging is performed by applying tree ensembles.

While tree based methods in stock return predictions have mainly been employed in a classification approach, predicting whether a stock price increases or decreases (c.f. Basak et al. (2019)), we choose to apply regression trees and predict actual values of stock prices.

In addition to the application of standard regression trees, we also introduce the concept of local linear trees. Regression trees implicitly employ constant piecewise functions, while local linear trees apply non-continuous piecewise linear functions enabling them to capture smooth signals (c.f. Greene (2003)). Piecewise linear functions nest piecewise constant functions, i.e. local linear trees nest regression trees. Thus, we argue that local linear trees are more flexible than regression trees by being able to capture linear trends within certain areas.

Although there exists a vast literature on machine learning and tree based methods, there are comparatively few publications on linear trees. The selection of publications with an econometric background is even smaller. The idea of trees with linear regressions within the nodes was employed relatively early in 1992 by Karalic (1992). Instead of linear regressions, Menze et al. (2011) apply ridge regressions when growing trees. The more recent papers by Friedberg et al. (2020) and Athey et al. (2019) apply local linear forests and generalized random forests, respectively. Both employ the R-function grf for their computations which we also examine in our comparison.

The paper is structured as follows. We illustrate the difference of regression trees and local linear trees in Chapter 4.2. Variations of these trees, as well as additional tree based methods, are introduced. Prediction accuracy of the different tree based methods are then compared in univariate and multivariate Monte Carlo simulations in Chapter 4.3. An empirical analysis of S&P 500 stock return predictions follows for both market-level and firm-level data in Chapter 4.4. Furthermore, we process implausible predictions due to outliers or breaks in the data. Also, we compute performance-based combinations of tree based methods with the historic mean. Finally, Chapter 4.5 concludes.

4.2 Regression Trees and Local Linear Trees

In this paper we are going to compare the prediction performance of tree based methods (TBMs). The goal of trees is to employ p regressor variables $x_1, \ldots, x_p$ to explain or predict a dependent variable y. To grow trees, the predictor space (i.e. the set of possible values for $x_1, \ldots, x_p$) is partitioned into J distinct regions $R_1, \ldots, R_J$ through multiple recursive binary splits. The splits within each predictor (sub-)space are determined by minimizing the Residual Sum of Squares

$$RSS = \sum_{j=1}^{J} \sum_{i \in R_j} (y_i - \hat{y}_i)^2. \tag{4.1}$$

The most prominent trees are regression trees (c.f. Breiman et al. (1984)). For regression trees, $\hat{y}_i$ in Equation (4.1) is replaced by $\overline{y}_{R_j}$, i.e. the mean of all y within region j.

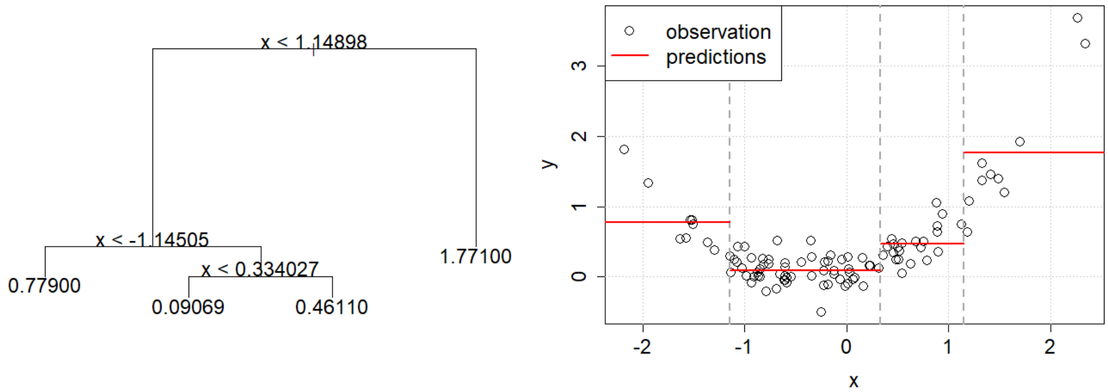
In the following, for explanatory purpose and comparison of different trees, we employ the simple univariate DGP:

$$y_i = 0.5x_i^2 + 0.3x_i + \varepsilon_i \tag{4.2}$$

with $i = 1, ..., 100$ and $\varepsilon \overset{iid}{\sim} N(0, 1)$. Applying the R package *tree* for regression trees results in the tree depicted in the left panel of Figure 4.1. The tree consists of three splits and thus four distinct regions/terminal nodes. Predictions for each terminal node are computed as the average value from all dependent variables within this terminal node.

The right panel of Figure 4.1 illustrates the splitting results in a scatter plot of the original observations. The original (simulated) data points are depicted as circles, while the dashed vertical lines indicate the three splits, dividing the surface into four distinct regions/terminal nodes. The red horizontal lines indicate the average value of the dependent variable within each terminal node, i.e. potential predictions of the regression tree.

Figure 4.1: Regression tree plus illustration of observations and corresponding predictions.



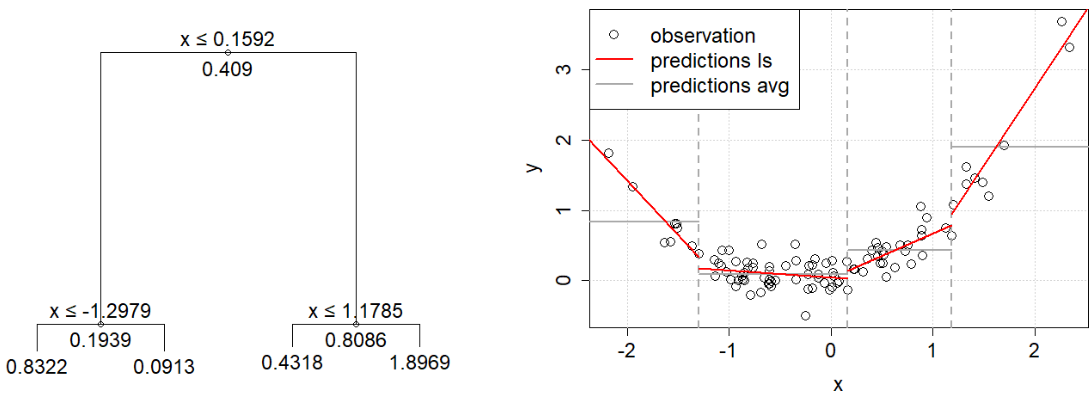
Put together, the illustration in the left panel shows the exact prediction values and the illustration in the right panel depicts the prediction accuracy. For our example, the illustration in the right panel of Figure 4.1 shows low prediction accuracy, especially in two outer regions. Thus, predictions from a least squares regression within a single node might be a more suitable choice compared to node averages. For this, we only have to change the procedure from regression trees in a minor way. We substitute the within node least squares prediction $\hat{y}_i^{LS}$ for $\hat{y}_i$ in Equation (4.1):

$$RSS = \sum_{j=1}^{J} \sum_{i \in R_j} (y_i - \hat{y}_i^{LS})^2. \tag{4.3}$$

Trees grown by this procedure are further referred to as local linear trees.

The illustrations in Figure 4.2 show the results of employing local linear trees to our data. The depicted tree now has a different shape due to new split points. Note that the values below each node are still regional means but do not function as predictions in this method. For local linear trees, predictions are derived from least squares regressions within nodes. These are represented by the red lines in the right panel of Figure 4.2. They show a better fit to the true observations than those in Figure 4.1. In fact, local linear trees should be able to outperform regression trees in most in-sample applications, since local linear trees nest regression trees.

Figure 4.2: Local linear tree plus illustration of observations and corresponding predictions.



***Notes*:** Grey lines in right panel indicate predictions from node averages are while red lines indicate predictions from linear regressions within nodes.

Trees, especially locally linear trees, have the advantage that they can flexibly approximate unknown regression functions. They are able to cope with structural (in-sample) breaks within the data by splitting the data set into different regions. Thus, trees perform a model selection and estimate a separate model for each region.

When constructing local linear trees, we can also easily obtain a hybrid version of regression trees and local linear trees. This hybrid version performs splits according to linear models (as in local linear trees), but generates predictions as the mean within each terminal node (as in regression trees). Predictions of this technique are illustrated as grey horizontal lines in the right panel of Figure 4.2 which correspond to the values below the terminal nodes in the left panel. Since this method comes with almost no (computing) cost when local linear trees are grown, we add it to our comparisons.

In our example when applying a minmal terminal node size of 10, the three methods result in in-sample MSEs of 0.137 for the regression tree (tree), 0.124 for the local linear tree with node averages as predictions (loli.tree.avg) and 0.035 for local linear trees with least squares predictions (loli.tree.ls). However, note that by construction, the order after only one single split should always be MSE(loli.ls.avg) $\geq$ MSE(tree) $\geq$ MSE(loli.tree.ls). Though, this does not hold for additional splits in general, as seen in our example. In this univariate setup loli.tree.ls yields the lowest in-sample MSE, by far. A more extensive comparison with these and other tree based methods follows in the next chapters.

4.2.1 Combining and Refining Trees

In addition to fully grown trees, we also consider refinements of single trees and combine multiple trees within tree ensembles.

Bagging is a tree ensemble method with which lower variance can be obtained compared to individual trees. In order to do this, B sample sets are bootstrapped from the original data and single trees are grown independently for each bootstrapped data set. Subsequently, predictions are determined for each tree, which are then averaged to obtain the final prediction for tree bagging.

Random Forests work similar to bagging of trees. However, instead of considering all available regressor variables when determining a split, only a random subsample of the regressor variables is considered to induce each split. Trees grown in this process are less correlated than those employed in the bagging process. This leads to an even lower variance. Therefore, we consider one (random) third of the available regressors at each split.

(Gradient) boosting is a third tree ensemble method. In contrast to bagging and random forests, trees in this ensemble depend on previously grown trees. In a first step, a single tree is grown and the residuals for the training data are computed. In the following step, a new tree is grown. However, modified residuals from the previously grown tree are employed as dependent variable in this process. Residuals are modified by regularization by shrinkage (c.f. Friedman (2001)). This step is repeated for a previously chosen number of times.

Pruning of trees reduces fully grown trees to single trees with less branches and splits. Fully grown trees might suffer from overfitting, while this is unlikely for pruned trees. Thus, fully grown trees lead to smaller IS errors than pruned trees. However, pruned trees might have considerably lower OOS errors than fully grown trees. We perform cost complexity pruning with cross validation (c.f. Friedman et al. (2001)). Unfortunately, cross validation leads to a multiplication in computing time, which is why we rarely apply this method.

4.2.2 Tree Based Methods

We compare the performance of local linear trees with those of different tree based forecasting methods in the remainder of this paper. We provide a brief overview of the competing procedures in this section. The principles of these methods are beyond the scope of this article.

- **tree** from the R-package *tree* grows a single regression tree (c.f Breiman et al. (1984)).
- **tree.bag** and **tree.rf** are random forests and bagging of regression trees. Bagging and random forests are identical when only one predictor variable is considered. We employ the R-package *randomForest*.
  In addition, we perform bagging and random forests for tree stumps, i.e. trees with only one split. These methods are indicated by names with suffix **.1**.
- **tree.boost** performs gradient boosting, which is an additional ensemble method for regression trees. We apply the package *gbm* with shrinkage parameter 0.1. Further details to gradient boosting can be found in Friedman (2001).
- **loli.tree.ls** indicates a local linear tree as described in Section 4.2. Similar to regression trees, we add bagging, random forests and boosting for the local linear trees, denoted by **loli.bag.ls**, **loli.rf.ls** and **loli.boost.ls**, respectively. The shrinkage parameter for boosting is set to 0.1 as in tree.boost.
  Note that in order to have sufficient degrees of freedom when performing linear regressions, the minimal node size of local linear trees has to increase with additional predictor variables.
- **loli.tree.avg** grows identical linear trees as loli.tree.ls, but its predictions are based on node averages instead of linear regressions within these nodes.
- **loli.tree.ls*** works similar to loli.tree.ls, but predictions are solely based on linear regressions of split relevant variables. Both methods are identical when only one regressor is employed.
- **loli.tree.rid** is a modification of loli.tree.ls which applies ridge regressions instead of OLS. Furthermore, in **loli.tree.rid.p** we consider pruning of local linear ridge trees using five-fold-cross-validation to find the optimal tree shape/size.
- **ctree** from the R-package *partykit* grows a single conditional regression tree and it works similar to regression trees. Ctree differs by testing for independence between regressor variables and response variable prior to performing splits. The algorithm does not perform further splits when independence cannot be rejected. Hence, it tends to split less often than regression trees. Ctree has no option to control for the minimal node size, instead it controls for the depth of a tree, i.e. how many splits and therefore terminal nodes are allowed for. For details regarding its implementation we refer to Hothorn et al. (2006).

- **bart** is an ensemble technique for bayesian additive regression trees. We employ the $BayesTree$ package which does not control for minimal node size. For references please check Chipman et al. (2010).

- **grf** from the package grf grows generalized random forests (c.f. Athey et al. (2019)). We only display results for honest forests in this paper and drop non-honest forests, since the honest ones perform better throughout our computations. Honest forests use half the training set to determine the splits/shape of a tree and the other half to compute the labels/values for these splits and nodes. Non-honest forests, on the other hand, use the full training set for both tasks.

4.3 Monte Carlo Simulations

The goal of this paper is to explore the potential of tree based methods (TBMs) in predicting stock returns. However, it is uncertain how these different TBMs perform when dealing with persistent regressors as is common for stock return predictions (c.f. Campbell and Yogo (2006), Stambaugh (1999), Demetrescu and Hillmann (2020)). Therefore, we first investigate the behavior of TBMs in Monte Carlo Simulations with persistent regressors before applying them to real data in Chapter 4.4.

4.3.1 Simulations: Univariate DGP

We begin our Monte Carlo simulations by analysing the behavior of TBMs when facing persistent regressors in a univariate setup. We employ a DGP with an autoregressive predictor variable which is allowed to be endogenous, i.e. the contemporaneous correlation of the innovations u_t and v_t may be nonzero (c.f. Stambaugh (1999), Campbell and Yogo (2006)):

$$y_t = \beta x_{t-1} + u_t \tag{4.4}$$

$$x_t = \rho x_{t-1} + v_t \tag{4.5}$$

where

$$\begin{pmatrix} u_t \\ v_t \end{pmatrix} \overset{iid}{\sim} N\left(0, \begin{pmatrix} 1 & \delta \\ \delta & 1 \end{pmatrix}\right). \tag{4.6}$$

We consider different degrees of persistence in the autoregressive process (4.5) with $\rho = 1 - c/T$ and $c \in \{0, 5, 15\}$ for samples of size $T \in \{100, 500\}$. The slope parameter β is constructed as $\beta = b/T$ with $b \in \{5, 20\}$ and the correlation δ is set to -0.95. We employ the first $T - 1$ observations as training set and only the last observation as test set to mimic the prediction of a time series. Finally, we compute 2000 Monte Carlo replications and grow 100 trees for each tree ensemble method.

Although the default minimal node sizes in the applied R-packages are equal to five or ten, we choose to apply a minimal node size of 25. We make an exception for ctree and bart since they do not control for minimal node size. We do so for two reasons. First, we need sufficient observations within each terminal node to perform predictions for local linear trees in the multivariate setup with up to eleven regressor variables in later chapters. Second, it showed that for our data, the applied R-packages yield better results with a minimum node size of 25 than with their default minimal node size.

Out-of-sample (OOS) MSEs for the TBMs are summarized in Table 4.1 while in-sample (IS) results can be found in Table 4.13 in the Appendix. We do not display MSEs in absolute terms but relative to the MSE of the historic mean, i.e. a value below one indicates a better performance than the historic mean. The three best performing methods (with respect to OOS MSEs) for each Monte Carlo setup are highlighted in each column of Table 4.1.

Note that we excluded random forests from this comparison since they are identical to bagging when only one predictor variable is at hand. We also excluded local linear trees relying their prediction on split relevant variables only (loli.ls*), since they are identical to local linear trees (loli.ls) in a univariate setup. Finally, we also exclude local linear ridge trees because their computation requires at least two predictor variables.

Overall, bagging of local linear tree stumps (loli.bag.ls.1) seems to be the best performing procedure throughout the different parameter values and sample sizes. It yields the lowest average relative MSE of 0.899, followed by bagging of regression tree stumps (tree.bag.1, 0.911) and boosting of local linear trees with predictions from node averages (loli.boost.avg, 0.911). Bagging of fully grown regression trees, on the other hand, is the least performing method in this comparison (1.156), followed by local linear trees (1.034) and bagging of local linear trees (1.009).

All in all, bagging of fully grown trees performs considerably worse than bagging of tree stumps with respect to OOS MSEs. It does, however, perform much better than tree stumps according to IS MSEs. Both aspects lead to the conclusion that ensembles from fully grown trees suffer from overfitting. We also tested bagging of trees with two and three splits, but they did not perform as well as with tree stumps. We did not perform cross validation for tree size due to restraints in computing time. More specifically, a single MC iteration combined for loli.bag.ls and loli.bag.avg takes about 270 seconds for each setup with sample size $n = 500$, i.e. roughly 150 hours for 2000 MC replications. Pruning applying k-fold cross validation would roughly take k times as long, i.e. 1 or 2 months for $k = 5$ or $k = 10$, respectively. Computation times dramatically increase when moving from univariate to multivariate computations in the following chapters, which is

Table 4.1: Relative univariate OOS forecasting performance.

c	0	0	5	5	15	15
b	5	20	5	20	5	20
			$T = 100$			
tree	1.035	0.490	1.198	0.909	1.227	1.080
tree.bag	1.125	0.520	1.296	0.986	1.341	1.164
tree.bag.1	0.938	0.535	**1.001**	0.786	**1.026**	**0.890**
tree.boost	0.940	0.448	1.028	0.810	1.056	0.937
loli.tree.ls	1.132	0.517	1.320	0.949	1.251	1.106
loli.bag.ls	1.033	0.463	1.174	0.856	1.145	0.993
loli.bag.ls.1	0.948	**0.427**	1.026	**0.746**	1.031	**0.888**
loli.boost.ls	1.129	0.493	1.164	0.833	1.223	1.076
loli.tree.avg	0.974	0.462	1.082	0.843	1.091	0.961
loli.bag.avg	0.938	0.441	1.047	0.801	1.059	0.935
loli.bag.avg.1	0.942	0.579	**0.989**	0.802	**1.003**	**0.911**
loli.boost.avg	**0.926**	0.523	**0.987**	**0.773**	1.027	0.922
ctree	0.973	0.492	1.036	0.832	**1.012**	0.961
bart	**0.923**	**0.429**	1.047	0.791	1.045	0.915
grf.honest	**0.918**	**0.435**	1.022	**0.783**	1.039	0.917
			$T = 500$			
tree	1.005	0.842	1.021	0.987	1.011	1.018
tree.bag	1.246	1.036	1.302	1.231	1.330	1.294
tree.bag.1	0.990	0.812	1.013	**0.943**	1.005	0.992
tree.boost	**0.984**	**0.811**	1.018	0.969	1.025	0.999
loli.tree.ls	1.065	0.866	1.086	1.007	1.068	1.039
loli.bag.ls	1.108	0.910	1.133	1.061	1.122	1.105
loli.bag.ls.1	**0.988**	**0.789**	1.014	**0.937**	1.005	**0.988**
loli.boost.ls	1.031	**0.811**	1.062	0.980	1.042	1.013
loli.tree.avg	0.994	0.858	1.026	0.971	1.015	1.006
loli.bag.avg	1.018	0.842	1.065	1.017	1.067	1.042
loli.bag.avg.1	0.990	0.855	**1.004**	0.957	**1.001**	**0.991**
loli.boost.avg	0.990	0.846	**1.003**	**0.954**	**1.001**	**0.984**
ctree	**0.984**	0.830	**1.001**	0.977	**1.004**	0.998
bart	0.995	0.819	1.022	0.973	1.031	1.001
grf.honest	0.999	0.825	1.037	0.990	1.039	1.020

Notes: Average OOS MSEs relative to average OOS MSEs of historic mean for DGPs with different degrees of persistence in Monte Carlo Simulation.

why we drop pruning of tree ensembles. In future research one could address this problem by distributing these computations to multiple CPU cores. However, we conjecture that in most cases cross validation would result in stumps anyway.

Table 4.1 furthermore shows an increase of relative OOS MSEs for most TBMs when c grows, indicating a better performance of TBMs relative to the historic mean with increasing degree of persistence (decrease of c).

Summing up, several tree based methods are able to outperform the historic mean in terms of OOS MSEs in our univariate setup. In the case of weak signals (small value of b), this is especially true for persistent regressor variables.

4.3.2 Simulations: Multivariate DGP

While we only considered univariate data sets in the previous chapter, we now turn our attention to multivariate predictions. We do this, hoping to increase the prediction performance of TBMs by preventing an omitted variable bias.

We choose to apply the realistic design DGP from Xu and Guo (2019) to test the behavior of TBMs in the presence of multiple persistent regressors. Therefore, we consider a model with m demeaned predictor variables x_{t-1}:

$$y_t = \mu_y + \boldsymbol{x'_{t-1}\beta} + u_{0t} \tag{4.7}$$

$$\boldsymbol{x_t} = \boldsymbol{\rho x_{t-1}} + \boldsymbol{u_{xt}} \tag{4.8}$$

where $\boldsymbol{x_t} = (x_{1t}, \ldots, x_{mt})'$, $\boldsymbol{u_t} = (u_{0t}, \boldsymbol{u'_{xt}})'$ and $\boldsymbol{\Sigma} = Var(\boldsymbol{u_t})$.

First, we substitute real data from the empirical application in Chapter 4.4 for y_t and $\boldsymbol{x_t}$ and fit (4.7) via OLS and (4.8) with a VAR(m) to obtain estimates $\hat{\mu}_y$, $\hat{\boldsymbol{\beta}}$, $\hat{\boldsymbol{\rho}}$ and $\hat{\boldsymbol{\Sigma}}$. Afterwards, we use these estimates as true parameters in the model above to generate data sets for the Monte Carlo replications, assuming $u_t = (u_{0t}, u_{xt})' \overset{iid}{\sim} N(0, \hat{\boldsymbol{\Sigma}})$. We generate Monte Carlo samples of size $T \in \{60, 120, 240, 480\}$.

This procedure is performed for the first $m \in \{2, 4, 6, 8\}$ regressor variables of our empirical data set. Since the ordering of these variables effects the results (when only the first m variables are used), we perform this procedure for 4 different arrangements of these regressors; named model set 0 to model set 3.

For model set 0 we use the default/random order in which we loaded the data. For model set 1 we rearrange the regressors according to the p-values of their slope parameter in univariate OLS regressions. Finally, the regressors are sorted according to their degree of persistence in ascending and descending order for model set 2 and model set 3, respectively. Note that the four model sets lead to identical results when all eight available predictor variables are employed, independent of their ordering.

Details on the applied data set can be found in Chapter 4.4. Some preliminary data analysis regarding this data and the regressor ordering for the different model sets can be found in Table 4.2. The estimated β parameters for all model sets and number of regressor variables is displayed in Table 4.3.

Table 4.2: Preliminary data analysis 1990M01 to 2019M08

Predictor	mean	sd	ρ_i	β_i	p_i	#M0	#M1	#M2	#M3
logDP	-3.9146	0.2703	0.983	0.016	0.050	1	3	7	4
logEP	-3.1105	0.3483	0.975	0.008	0.178	2	4	6	2
DY	-3.9087	0.2706	0.984	0.016	0.044	3	2	8	5
LTR	0.0068	0.0290	0.027	0.098	0.192	4	5	1	3
SVAR	0.0026	0.0045	0.705	-1.182	0.014	5	1	3	1
BM	0.2873	0.0827	0.971	0.030	0.247	6	7	5	8
INFL	0.0020	0.0033	0.468	0.328	0.621	7	8	2	6
DFY	0.0095	0.0039	0.964	-0.702	0.210	8	6	4	7

Notes: The column ρ_i contain the estimated AR(1) parameters for each regressor variable. The column β_i displays the estimated OLS slope coefficient while p_i contains the corresponding p-value. #M0 to #M3 show the variable ordering in model sets 0 to 3.

Table 4.3: β coefficients for DGP

Model Set	m	β_1	β_2	β_3	β_4	β_5	β_6	β_7	β_8
0	2	0.014	0.007						
	4	-0.020	0.006	0.034	0.094				
	6	0.052	0.004	-0.023	0.124	-1.429	-0.054		
	8	0.054	0.006	-0.025	0.125	-1.563	-0.062	-0.023	0.313
1	2	-1.198	0.016						
	4	-1.311	-0.025	0.042	0.001				
	6	-1.379	-0.020	0.038	0.000	0.122	-0.150		
	8	-1.563	-0.025	0.054	0.006	0.125	0.313	-0.062	-0.023
2	2	0.112	0.557						
	4	0.134	0.045	-1.453	0.291				
	6	0.126	0.001	-1.359	0.177	0.023	0.002		
	8	0.125	-0.023	-1.563	0.313	-0.062	0.006	0.054	-0.025
3	2	0.027	-0.011						
	4	0.025	0.003	0.010	-0.056				
	6	-0.028	0.057	0.006	-0.058	0.200	-1.379		
	8	-0.025	0.054	0.006	-0.062	0.313	-1.563	-0.023	0.125

Notes: Values of β coefficients for Monte Carlo DGP of models 0 to 3 with $m \in \{2, 4, 6, 8\}$ regressor variables.

Similar to the previous univariate Monte Carlo simulations, we again use $T-1$ observations for the training set and only the last observation as test set. Relative OOS MSEs for sample size $T = 240$ can be found in Table 4.4, while results for more sample sizes, as well as IS results, can be found in Tables 4.14 to 4.16 in the Appendix.

Results for pruning of local linear ridge trees with sample size $T = 60$ are missing since this number of observations is too small for a reasonable cross validation in the pruning process. Furthermore, there are no results for random forests of ridge trees with

$m = 2$ or $m = 4$ regressor variables available, random ridge forests require at least $m = 6$ regressor variables.

Prediction performances superior (relative MSE < 1.000) or slightly inferior (relative MSE < 1.010) to the historic mean are highlighted in Table 4.4 by blue and light blue tiles, respectively. Light red and red tiles indicate relative MSEs above 1.500 and 2.000.

Overall, bagging and random forests perform better when applied to stumps instead of fully grown trees, confirming the results of our previous univariate Monte Carlo Simulations. Local linear stumps are quite flexible, they can produce monotonic convex and concave as well as V-shaped functions. The best performing procedure over all sample sizes, model sets and number of regressors are random forests of local linear stumps with predictions from node averages (loli.rf.avg.1), followed by generalized random forests and random forests of regression tree stumps (tree.rf.1). However, loli.rf.avg.1 is the only procedure with average relative MSE below 1. Local linear trees and boosting thereof are the least performing TBMs over all setups. This ranking confirms our results from previous univariate simulations.

Loli.rf.avg.1 performs best for model set 0, followed by model sets 3, 1 and 2. However, this ranking does not hold for all tree based methods, none of the regressor arrangements is superior to the others in general.

No uniformly best number of regressor variables m emerges over all TBMs, model sets and sample sizes. However, local linear trees with predictions from OLS or ridge regressions tend to perform better with a smaller amount of regressor variables.

Relative MSEs for different sample sizes are not (directly) comparable, though absolute MSEs indicate a considerable improvement in the performance of all TBMs with increasing sample size. However, the same holds for historic mean prediction which is why relative MSEs do not change much over different sample sizes. Still, the relative performance of most TBMs increases with an increasing sample size.

In summary, many tree based methods are able to outscore the historic mean for certain regressor variables, but no method is able to outperform the historic mean for all combinations of predictor variables and sample sizes. Random forests of local linear stumps with predictions based on node averages and generalized random forests are the most promising methods in the comparison made.

Table 4.4: Relative multivariate OOS forecasting performance for $T = 240$.

m	2				4				6				8
model	0	1	2	3	0	1	2	3	0	1	2	3	all
tree	1.046	1.085	1.050	1.027	1.100	1.091	1.103	1.147	1.079	1.142	1.050	1.098	1.104
tree.bag	1.071	1.086	1.094	1.065	1.068	1.069	1.055	1.141	1.095	1.103	1.021	1.085	1.111
tree.bag.1	0.995	1.012	1.002	1.000	1.017	0.995	0.991	1.007	1.002	1.009	0.984	0.994	1.033
tree.rf	1.049	1.066	1.068	1.043	1.029	1.032	1.024	1.084	1.072	1.071	1.014	1.053	1.057
tree.rf.1	0.993	1.001	1.000	0.997	0.999	0.987	0.989	0.992	0.995	0.996	0.986	0.993	1.008
tree.boost	1.037	1.061	1.077	1.029	1.040	1.107	1.068	1.099	1.063	1.065	1.035	1.054	1.103
loli.tree.ls	1.222	1.261	1.139	1.159	1.308	1.431	1.436	1.522	1.720	1.582	1.603	1.699	1.910
loli.bag.ls	1.056	1.087	1.045	1.065	1.084	1.091	1.040	1.159	1.161	1.135	1.111	1.218	1.167
loli.bag.ls.1	1.008	1.036	1.001	1.010	1.030	1.016	0.992	1.058	1.038	1.027	1.019	1.072	1.065
loli.rf.ls	1.042	1.065	1.040	1.048	1.067	1.062	1.019	1.114	1.125	1.106	1.078	1.144	1.129
loli.rf.ls.1	1.000	1.027	1.003	1.006	1.018	1.007	0.982	1.041	1.027	1.027	1.011	1.051	1.044
loli.boost.ls	1.130	1.186	1.136	1.134	1.267	1.294	1.238	1.410	1.482	1.403	1.444	1.495	1.497
loli.tree.avg	1.035	1.053	1.040	1.024	0.993	1.068	1.060	1.061	1.041	1.036	1.013	1.017	1.080
loli.bag.avg	1.000	1.018	1.020	1.008	0.976	1.010	1.006	1.009	0.998	1.001	0.988	0.994	1.003
loli.bag.avg.1	1.000	1.018	1.020	1.008	0.976	1.009	1.006	1.009	0.998	1.000	0.987	0.994	1.003
loli.rf.avg	0.992	1.017	1.017	1.003	0.983	1.002	0.997	1.002	0.996	1.001	0.994	0.987	0.997
loli.rf.avg.1	0.987	1.003	1.002	0.996	0.989	0.996	0.995	0.991	0.993	0.999	0.993	0.994	1.001
loli.boost.avg	0.980	1.022	1.008	1.001	0.985	1.007	1.013	1.009	1.014	1.004	0.989	0.995	1.003
loli.tree.ls*	1.214	1.256	1.139	1.158	1.272	1.364	1.329	1.438	1.497	1.348	1.423	1.434	1.425
loli.bag.ls*	1.054	1.083	1.044	1.062	1.055	1.060	1.030	1.126	1.091	1.073	1.036	1.116	1.064
loli.bag.ls*.1	1.002	1.010	1.000	1.001	0.993	0.999	0.996	1.004	0.995	0.997	0.990	0.999	1.002
loli.rf.ls*	1.040	1.063	1.039	1.046	1.045	1.047	1.009	1.096	1.071	1.059	1.023	1.078	1.061
loli.rf.ls*.1	0.995	1.007	1.001	1.001	0.997	0.994	0.990	0.997	0.993	0.999	0.990	0.998	1.003
loli.boost.ls*	1.069	1.102	1.067	1.061	1.085	1.063	1.063	1.161	1.071	1.054	1.029	1.060	1.084
loli.tree.rid	1.137	1.167	1.130	1.109	1.255	1.147	1.218	1.249	1.409	1.310	1.316	1.573	1.410
loli.tree.rid.p	1.086	1.103	1.048	1.023	1.099	1.039	1.082	1.150	1.169	1.074	1.156	1.165	1.193
loli.bag.rid	1.035	1.067	1.035	1.043	1.060	1.057	1.025	1.107	1.122	1.082	1.071	1.129	1.115
loli.bag.rid.1	0.999	1.028	0.999	1.005	1.012	1.004	0.989	1.031	1.026	1.004	1.006	1.039	1.042
loli.rf.rid	-	-	-	-	-	-	-	-	1.049	1.050	1.026	1.074	1.077
loli.rf.rid.1	-	-	-	-	-	-	-	-	1.005	1.005	0.993	1.021	1.026
loli.boost.rid	1.102	1.147	1.093	1.097	1.156	1.205	1.116	1.250	1.307	1.220	1.232	1.281	1.336
ctree	1.013	1.019	0.998	0.996	1.015	1.012	0.999	1.006	0.998	1.021	0.992	1.003	1.009
bart	1.040	1.029	1.038	1.012	1.032	1.035	1.023	1.075	1.051	1.034	1.008	1.057	1.053
grf	0.988	1.010	1.004	0.997	0.996	0.993	0.990	1.000	0.995	1.002	0.992	0.990	1.001

Notes: Average IS MSEs relative to average IS MSEs of historic mean for multivariate predictions of S&P 500 stock returns. Variable m indicates the number of regressor variables while *model* indicates the model set from which regressor variables are drawn. The best predictor per procedure are highlighted by bold relative MSEs. See text for details to the Monte Carlo DGP.

4.4 Stock Return Predictability

We assessed multiple TBMs on predicting simulated data in the previous chapter, while we now apply all TBMs to real data. In the following applications, out-of-sample rolling window predictions for monthly stock returns are performed. This is done for two different types of data. In Chapter 4.4.1 we resort to aggregated market data only, while we add firm-specific data in Chapter 4.4.3.

4.4.1 Univariate S&P 500 Predictions

In this section, we perform univariate stock return predictions with S&P 500 data from the webpage of Amit Goyal[1]. We employ stock returns as dependent variable and log dividend-price ratio, log earnings-price ratio, dividend yield, long term rate of returns, stock variance, book-to-market ratio, inflation and default yield spread as possible lagged predictor variables. We use monthly data from the recent period 1990M01 to 2019M08 which matches the time span for the available data in later computations in Chapter 4.4.3. Further details regarding these variables can be found in Welch and Goyal (2008).

In this univariate empirical application, we apply the same TBMs as in our previous univariate Monte Carlo simulations in Chapter 4.3. We perform moving window predictions with windows of $n \in \{60, 120, 240\}$ months for each predictor variable individually. Relative OOS MSEs of these predictions can be found in Table 4.5. We apply the same color scheme as in previous predictions to highlight good/poor performances. Additionally, the best performance in each row of Table 4.5 is highlighted to indicate which predictor variable performs best for each TBM.

Most TBMs perform on a similar level as the historical mean but no method is able to outperform the historic mean for all predictor variables and window lengths. The lowest average relative OOS MSEs over all sample sizes and regressor variables are achieved by generalized forests (1.008), conditional trees (1.014) and boosting of local linear trees with predictions from node averages (1.022). Bagging of fully grown regression trees, on the other hand, is the least performing procedure (1.171).

There exists no single best predictor variable for all TBMs in our comparison. Long term rate of returns yields the lowest relative MSE on average, however, the overall best performing combination is bagging of local linear trees with predictions from node averages in small samples ($n = 60$) with default yield spread as predictor (0.931).

Univariate TBM tend to perform better in our comparison in small and large samples ($n \in \{60, 240\}$) compared to medium-sized ones ($n = 120$). Though, we need to keep the different prediction periods for different sample sizes in mind when comparing these re-

[1] We would like to thank Amit Goyal for making the data available on his webpage `//www.hec.unil.ch/agoyal/`

sults. With data available since 1990, first predictions for windows of size $n = 60, 120, 240$ months are performed in 1995, 2000 and 2010, respectively. Thus, short and medium sized windows perform predictions during the stock market downturn of 2002 and the Great Recession (2007 - 2009) while predictions with large windows only start after the Great Recession with less volatile stock returns.

To illustrate the behavior of TBMs over time, we exemplarily plot the differences in cumulated squared forecast errors (DCSFEs) of historic mean and TBM predictions for logDP and logEP in Figure 4.3. A DCSFE value above zero indicates a lower cumulated squared error of a TBM than for the historic mean, i.e. a relative better performance of the TBM in recent periods. Most of the time, some procedures perform quite well indicated by a horizontal or slightly increasing DCSFE line. However, at certain moments they perform very poorly. For panels with $n \in \{60, 120\}$, one can clearly identify the time of the Great Recession where various DCSFEs experience a sharp decline. The least and the two best performing TBMs are highlighted in red, black and blue. Corresponding OOS MSEs are listed in the legend.

Summing up, univariate TBMs do not perform as well in our empirical application as in our previous Monte Carlo simulations. Especially local linear TBMs do not perform well in volatile business cycle periods. Generalized random forests leave the best impression, in particular for logEP as predictor variable with relative OOS MSE below 1 for all window lengths in our setup.

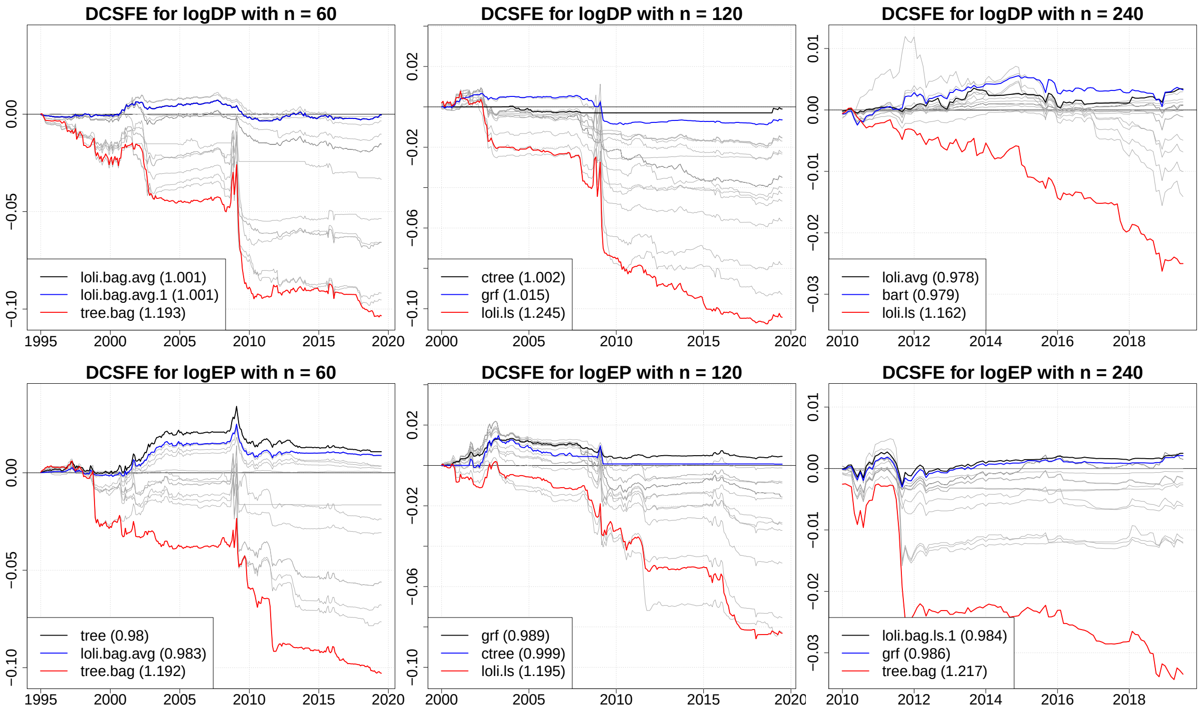
4.4.2 Multivariate S&P 500 Predictions

Univariate TBMs did not perform as well as expected compared to the historic mean in predicting stock returns in the previous section. Thus, we hope for more precise predictions for multi-regressor predictions which utilize more information than univariate ones.

We employ the same data as for the univariate predictions but apply more than one regressor per prediction. However, we do not utilize the full set of regressor variables in every computation. We do this because it is uncertain whether TBMs perform better the more regressors are employed or whether it performs best for a certain number of regressors.

Thus, we apply a similar procedure as in the multivariate Monte Carlo simulations in Chapter 4.3.2 to choose different combinations of regressor variables. The regressors are arranged in four model sets in different orders. For model set 0, we us the predictor variables in the order presented in Chapter 4.4.1. Since we did not put much thought into the order, it is considered random. For model set 1, we perform univariate OLS regressions of stock returns on each predictor variable separately. Subsequently, we rank the predictors according to the p-value of their estimated slope parameters. Model sets

Figure 4.3: Difference in cumulated squared forecast errors.



***Notes*:** Difference in cumulated squared forecast errors of historic mean and TBM (DCSFE) for different TBMs over time; left to right: window length = 60, 120 and 240 months; logDP and logEP as regressor variables on top and bottom panels, respectively. Two best and least performing procedures (w.r.t. OOS MSEs) are highlighted in black, blue and red, respectively, corresponding relative MSE values are depicted in legend.

2 and 3 rank the predictor variables according to their degree of persistence in ascending and descending order, respectively. We estimate univariate AR(1) coefficients for each predictor variable as indicator for their degree of persistence. Further details regarding these different orderings can be found in Table 4.2.

For each of these arrangements, we perform rolling window estimations for the first $m \in \{2, 4, 6, 8\}$ variables. All model sets lead to identical predictions when the full set of $m = 8$ regressor variables is employed. Hence, we are left with 13 different combinations of regressor variables.

We employ the same TBMs as in Chapter 4.3.2 and perform rolling window predictions with windows of size 5, 10 or 20 years as in Chapter 4.4.1. Relative average OOS MSEs of these rolling window predictions are displayed in Table 4.6, 4.7 and 4.8. The same color pattern as in previous tables is used to indicate how well methods perform; blue tiles indicate a performance superior to the historic mean while light blue tiles represent relative MSEs below 1.010. Red tiles indicate performances which are at least 50 percent worse than the historic mean in terms of OOS MSEs.

Table 4.5: Relative univariate OOS forecasting performance for S&P 500 stock returns.

Predictor	logDP	logEP	DY	LTR	SVAR	BM	INFL	DFY	∅
				$n = 60$					
tree	1.029	**0.980**	1.019	1.022	1.037	1.029	1.041	1.042	1.025
tree.bag	1.193	1.192	1.267	**1.038**	1.140	1.215	1.136	1.127	1.163
tree.bag.1	1.100	1.057	1.147	**1.025**	1.088	1.104	1.058	1.060	1.080
tree.boost	1.019	**0.994**	1.015	1.019	1.037	1.025	1.030	1.040	1.022
loli.tree.ls	1.177	1.126	1.181	1.130	1.283	1.242	**1.097**	1.176	1.177
loli.bag.ls	1.122	1.104	1.127	1.074	1.247	1.202	**1.060**	1.135	1.134
loli.bag.ls.1	1.122	1.104	1.127	1.074	1.247	1.202	**1.060**	1.135	1.134
loli.boost.ls	1.171	1.142	1.168	**1.072**	1.287	1.228	1.091	1.155	1.164
loli.tree.avg	1.006	**0.994**	1.012	1.013	1.027	1.027	1.040	1.049	1.021
loli.bag.avg	1.001	0.983	1.009	1.010	1.026	1.018	1.025	**0.931**	1.000
loli.bag.avg.1	1.001	0.984	1.009	1.010	1.026	1.018	1.025	**0.931**	1.000
loli.boost.avg	1.001	**0.996**	1.012	1.013	1.027	1.024	1.030	1.048	1.019
ctree	1.062	1.031	1.064	1.003	1.023	1.034	**1.000**	1.039	1.032
bart	1.123	1.044	1.094	**1.041**	1.111	1.116	1.076	1.081	1.086
grf	1.003	**0.999**	1.002	1.002	1.000	1.002	1.001	1.002	1.001
				$n = 120$					
tree	1.082	**1.037**	1.039	1.040	1.068	1.040	1.044	1.045	1.049
tree.bag	1.218	1.194	1.236	**1.104**	1.173	1.200	1.145	1.121	1.174
tree.bag.1	1.055	1.019	1.062	**1.007**	1.055	1.058	1.030	1.045	1.041
tree.boost	1.101	1.068	1.080	1.060	1.054	**1.052**	1.054	1.057	1.066
loli.tree.ls	1.245	1.195	1.254	1.091	1.258	1.225	**1.083**	1.181	1.192
loli.bag.ls	1.133	1.114	1.150	**1.050**	1.168	1.159	1.066	1.155	1.124
loli.bag.ls.1	1.093	1.076	1.099	**1.030**	1.133	1.100	1.039	1.104	1.084
loli.boost.ls	1.184	1.177	1.214	1.093	1.218	1.201	**1.085**	1.155	1.166
loli.tree.avg	1.053	1.039	1.024	1.035	1.034	1.053	**1.009**	1.057	1.038
loli.bag.avg	1.036	1.020	1.042	1.023	1.030	1.025	**1.010**	1.042	1.029
loli.bag.avg.1	1.036	1.020	1.041	1.023	1.030	1.025	**1.010**	1.041	1.028
loli.boost.avg	1.040	1.003	1.039	1.028	1.016	1.032	**0.995**	1.036	1.024
ctree	1.002	**0.999**	1.014	1.000	1.029	1.000	1.000	1.077	1.015
bart	1.110	1.067	1.086	**1.021**	1.063	1.081	1.049	1.050	1.066
grf	1.015	**0.989**	1.020	1.001	1.007	1.011	1.003	1.012	1.007
				$n = 240$					
tree	0.995	**0.990**	0.992	1.000	1.029	1.059	1.090	1.147	1.038
tree.bag	1.092	1.217	1.205	**1.030**	1.210	1.186	1.205	1.254	1.175
tree.bag.1	0.992	1.003	0.997	0.984	**0.971**	1.005	1.018	1.029	1.000
tree.boost	1.044	1.039	1.051	**1.004**	1.164	1.073	1.104	1.152	1.079
loli.tree.ls	1.162	1.074	**1.064**	1.091	1.232	1.158	1.113	1.192	1.136
loli.bag.ls	1.065	1.079	1.043	**1.022**	1.132	1.088	1.107	1.152	1.086
loli.bag.ls.1	**0.981**	0.984	0.992	0.997	1.001	1.016	1.026	1.053	1.006
loli.boost.ls	1.032	1.078	1.038	**1.017**	1.166	1.107	1.110	1.154	1.088
loli.tree.avg	**0.978**	1.015	1.028	1.020	1.052	1.056	1.040	1.121	1.039
loli.bag.avg	**1.007**	1.011	1.018	1.023	1.086	1.038	1.086	1.135	1.051
loli.bag.avg.1	**1.007**	1.010	1.018	1.024	1.085	1.037	1.085	1.134	1.050
loli.boost.avg	**0.989**	1.017	1.006	1.008	1.024	1.012	1.053	1.087	1.024
ctree	1.001	1.000	1.002	1.000	**0.956**	1.000	1.000	1.000	0.995
bart	**0.979**	1.040	1.041	0.984	0.988	1.074	1.045	1.101	1.031
grf	0.994	**0.986**	1.004	1.009	1.026	1.018	1.035	1.044	1.015

Notes: Average OOS MSEs relative to average OOS MSEs of historic mean for univariate predictions of S&P 500 stock returns. Last column contains row averages, best predictor per procedure are highlighted by bold relative MSEs.

Table 4.6: Relative multivariate OOS forecasting performance for S&P 500 stock returns with $n = 60$.

m	2				4				6				8
model	0	1	2	3	0	1	2	3	0	1	2	3	all
tree	**1.019**	1.036	1.044	1.037	1.042	1.041	1.068	1.037	1.051	1.061	1.062	1.046	1.070
tree.bag	1.172	1.182	**1.104**	1.244	1.189	1.146	1.130	1.211	1.178	1.155	1.164	1.177	1.168
tree.bag.1	1.092	1.121	**1.043**	1.131	1.095	1.084	1.061	1.127	1.100	1.079	1.087	1.096	1.096
tree.rf	1.116	1.130	1.079	1.188	1.089	1.085	**1.077**	1.113	1.110	1.094	1.088	1.101	1.090
tree.rf.1	1.036	1.080	1.033	1.107	**1.023**	1.042	1.024	1.044	1.050	1.051	1.029	1.055	1.047
tree.boost	**1.033**	1.037	1.053	1.042	1.060	1.053	1.112	1.067	1.082	1.097	1.143	1.099	1.144
loli.tree.ls	1.295	1.427	**1.164**	1.259	1.644	1.745	1.919	2.062	2.595	5.285	4.196	3.081	3.681
loli.bag.ls	1.158	1.333	**1.102**	1.143	1.366	1.504	1.393	1.589	1.843	1.853	1.823	2.107	1.973
loli.bag.ls.1	1.158	1.333	**1.102**	1.143	1.366	1.504	1.393	1.589	1.843	1.853	1.823	2.107	1.973
loli.rf.ls	1.113	1.303	**1.088**	1.124	1.238	1.447	1.275	1.422	1.726	1.534	1.584	1.879	1.741
loli.rf.ls.1	1.113	1.303	**1.088**	1.124	1.238	1.447	1.275	1.422	1.726	1.534	1.584	1.879	1.741
loli.boost.ls	1.340	1.551	1.257	**1.207**	1.837	2.135	5.114	2.417	5.509	6.539	4.414	3.080	6.486
loli.tree.avg	1.014	1.039	1.027	1.026	1.022	1.021	1.031	1.018	**0.993**	1.008	1.017	0.997	1.008
loli.bag.avg	0.988	1.023	1.018	1.011	**0.983**	0.997	1.011	0.984	0.995	1.001	1.000	0.994	0.994
loli.bag.avg.1	0.988	1.023	1.018	1.011	**0.983**	0.996	1.011	0.985	0.995	1.001	1.000	0.994	0.994
loli.rf.avg	0.985	1.009	1.014	1.004	**0.980**	0.994	1.019	0.987	1.001	0.999	1.005	0.999	1.001
loli.rf.avg.1	0.985	1.009	1.014	1.004	**0.980**	0.994	1.019	0.987	1.001	0.999	1.005	0.999	1.001
loli.boost.avg	1.013	1.045	1.027	1.022	1.000	1.007	1.026	1.008	**0.976**	1.015	1.018	0.988	1.007
loli.tree.ls*	1.179	1.319	1.099	1.239	1.167	1.136	1.132	1.139	**1.020**	1.080	1.192	1.124	1.247
loli.bag.ls*	1.075	1.154	1.048	1.117	**0.987**	1.043	1.037	1.063	1.023	0.995	1.061	1.055	1.020
loli.bag.ls*.1	1.075	1.154	1.048	1.117	**0.987**	1.043	1.037	1.063	1.023	0.995	1.061	1.055	1.020
loli.rf.ls*	1.059	1.091	1.042	1.104	1.026	1.056	1.029	1.058	1.031	**1.014**	1.024	1.049	1.020
loli.rf.ls*.1	1.059	1.091	1.042	1.104	1.026	1.056	1.029	1.058	1.031	**1.014**	1.024	1.049	1.020
loli.boost.ls*	1.189	1.290	1.127	1.173	1.116	1.105	1.129	1.095	**1.026**	1.086	1.173	1.084	1.131
loli.tree.rid	1.193	1.207	**1.082**	1.200	1.328	1.303	1.552	1.480	3.820	1.575	1.993	1.668	1.936
loli.tree.rid.p	-	-	-	-	-	-	-	-	-	-	-	-	-
loli.bag.rid	1.067	1.156	**1.061**	1.095	1.214	1.236	1.256	1.333	1.467	1.404	1.485	1.497	1.478
loli.bag.rid.1	1.067	1.156	**1.061**	1.095	1.214	1.236	1.256	1.333	1.467	1.404	1.485	1.497	1.478
loli.rf.rid	-	-	-	-	-	-	-	-	1.238	**1.185**	1.208	1.265	1.302
loli.rf.rid.1	-	-	-	-	-	-	-	-	1.238	**1.185**	1.208	1.265	1.302
loli.boost.rid	1.178	1.226	**1.129**	1.163	1.343	1.324	1.470	1.483	3.022	2.992	1.814	1.693	1.854
ctree	1.047	1.042	**1.003**	1.048	1.026	1.045	1.029	1.012	1.025	1.031	1.035	1.025	1.019
bart	**1.045**	1.097	1.075	1.100	1.055	1.051	1.106	1.057	1.068	1.052	1.106	1.061	1.081
grf	1.001	1.003	1.000	1.006	1.000	1.002	1.002	0.998	1.001	**0.998**	1.001	1.005	0.998

Notes: Average OOS MSEs relative to average OOS MSEs of historic mean for multivariate predictions of S&P 500 stock returns. Variable m indicates the number of regressor variables while *model* indicates the model set from which regressor variables are drawn. The best predictor per procedure are highlighted by bold relative MSEs.

Table 4.7: Relative multivariate OOS forecasting performance for S&P 500 stock returns with $n = 120$.

m	2				4				6				8
model	0	1	2	3	0	1	2	3	0	1	2	3	all
tree	1.093	1.083	**1.068**	1.105	1.132	1.121	1.080	1.108	1.104	1.115	1.107	1.109	1.102
tree.bag	1.145	1.144	1.111	1.224	1.159	1.120	1.111	1.175	1.124	**1.080**	1.092	1.085	1.092
tree.bag.1	1.041	1.052	**1.013**	1.060	1.045	1.038	1.044	1.062	1.047	1.028	1.030	1.036	1.041
tree.rf	1.091	1.115	1.076	1.192	1.077	1.077	1.066	1.098	1.084	1.059	**1.051**	1.051	1.055
tree.rf.1	0.997	1.040	1.014	1.051	0.999	1.016	1.000	1.011	1.018	1.017	**0.996**	1.025	1.014
tree.boost	1.097	1.133	**1.087**	1.132	1.147	1.136	1.161	1.171	1.181	1.157	1.216	1.157	1.241
loli.tree.ls	1.391	1.682	**1.217**	1.297	1.741	1.606	1.679	1.841	2.834	2.377	3.758	2.282	3.136
loli.bag.ls	1.142	1.283	**1.086**	1.154	1.216	1.299	1.261	1.351	1.512	1.338	1.443	1.489	1.498
loli.bag.ls.1	1.109	1.232	**1.054**	1.110	1.120	1.199	1.212	1.259	1.410	1.244	1.316	1.357	1.409
loli.rf.ls	1.099	1.218	**1.066**	1.148	1.164	1.293	1.228	1.256	1.410	1.328	1.295	1.415	1.391
loli.rf.ls.1	1.067	1.157	**1.041**	1.104	1.104	1.197	1.183	1.180	1.287	1.220	1.209	1.280	1.268
loli.boost.ls	1.403	1.788	**1.271**	1.329	1.787	1.956	1.952	2.078	3.227	2.609	2.541	2.265	2.401
loli.tree.avg	1.045	1.030	1.032	**1.029**	1.086	1.051	1.085	1.048	1.056	1.054	1.119	1.101	1.046
loli.bag.avg	1.020	1.021	1.014	1.037	1.018	1.018	1.016	1.020	1.023	**1.013**	1.032	1.026	1.021
loli.bag.avg.1	1.020	1.022	1.014	1.037	1.018	1.018	1.016	1.021	1.023	**1.013**	1.032	1.026	1.020
loli.rf.avg	**1.004**	1.031	1.015	1.032	1.007	1.006	1.019	1.011	1.017	1.007	1.022	1.021	1.013
loli.rf.avg.1	**0.991**	1.015	1.006	1.000	0.997	0.996	1.013	1.000	1.005	0.999	1.010	1.009	1.003
loli.boost.avg	1.016	1.046	1.026	1.038	1.062	1.010	1.047	1.016	**1.009**	1.039	1.059	1.049	1.039
loli.tree.ls*	1.342	1.622	**1.179**	1.286	1.450	1.398	1.248	1.676	1.763	1.291	1.594	1.355	1.315
loli.bag.ls*	1.117	1.220	**1.065**	1.126	1.096	1.098	1.117	1.162	1.133	1.084	1.149	1.115	1.109
loli.bag.ls*.1	1.072	1.103	1.022	1.083	1.018	1.047	**1.013**	1.042	1.037	1.023	1.026	1.039	1.033
loli.rf.ls*	1.070	1.155	**1.047**	1.130	1.086	1.120	1.093	1.125	1.130	1.094	1.082	1.121	1.099
loli.rf.ls*.1	1.030	1.067	1.019	1.082	1.022	1.039	1.012	1.044	1.034	1.025	1.007	1.029	**1.007**
loli.boost.ls*	1.256	1.399	**1.152**	1.222	1.435	1.354	1.165	1.506	1.379	1.206	1.375	1.377	1.363
loli.tree.rid	1.231	1.256	**1.170**	1.188	1.385	1.332	1.370	1.468	1.709	1.553	1.963	1.701	3.055
loli.tree.rid.p	1.178	1.231	**1.148**	1.174	1.297	1.266	1.285	1.350	1.546	1.442	1.801	1.354	2.502
loli.bag.rid	1.102	1.162	**1.060**	1.124	1.145	1.195	1.170	1.218	1.324	1.216	1.284	1.291	1.342
loli.bag.rid.1	1.070	1.123	**1.038**	1.075	1.078	1.122	1.137	1.169	1.241	1.143	1.204	1.198	1.244
loli.rf.rid	-	-	-	-	-	-	-	-	1.197	1.155	**1.137**	1.187	1.240
loli.rf.rid.1	-	-	-	-	-	-	-	-	1.128	1.094	**1.092**	1.132	1.160
loli.boost.rid	1.300	1.451	**1.198**	1.229	1.500	1.561	1.548	1.681	2.571	1.790	2.050	1.936	1.903
ctree	1.003	1.053	**1.000**	1.004	1.006	1.042	1.074	1.011	1.015	1.029	1.020	1.034	1.027
bart	1.092	1.065	**1.038**	1.092	1.086	1.073	1.058	1.109	1.078	1.057	1.081	1.071	1.075
grf	1.002	1.007	1.011	1.006	1.001	1.007	1.003	1.001	1.002	1.006	1.005	**0.996**	1.001

Notes: See Table 4.6.

Table 4.8: Relative multivariate OOS forecasting performance for S&P 500 stock returns with $n = 240$.

m	2				4				6				8
model	0	1	2	3	0	1	2	3	0	1	2	3	all
tree	1.007	0.995	1.100	**0.990**	1.064	1.031	1.232	1.084	1.170	1.284	1.238	1.201	1.199
tree.bag	1.150	1.046	1.092	1.128	1.089	**1.043**	1.098	1.168	1.065	1.065	1.138	1.112	1.089
tree.bag.1	0.997	**0.969**	1.000	0.994	0.991	0.971	0.975	1.000	0.973	0.974	0.984	0.981	0.981
tree.rf	1.120	1.036	1.083	1.097	1.062	**1.036**	1.050	1.116	1.039	1.040	1.085	1.071	1.050
tree.rf.1	0.996	**0.981**	0.999	0.994	0.993	0.988	0.995	0.997	0.984	0.990	0.993	0.992	0.994
tree.boost	**1.026**	1.083	1.113	1.050	1.046	1.116	1.314	1.100	1.166	1.184	1.278	1.194	1.281
loli.tree.ls	1.311	1.513	1.398	1.296	1.482	1.936	**1.291**	1.808	1.759	1.771	1.484	2.250	2.047
loli.bag.ls	1.062	1.065	1.092	1.099	1.074	1.026	1.073	1.129	1.035	**0.992**	1.096	1.025	1.061
loli.bag.ls.1	0.988	0.993	1.010	1.016	1.004	0.980	1.066	1.027	**0.973**	1.018	1.060	1.023	1.033
loli.rf.ls	1.055	1.027	1.080	1.094	1.067	1.033	1.078	1.115	**1.005**	1.028	1.090	1.020	1.066
loli.rf.ls.1	0.987	0.982	1.014	1.029	1.000	0.980	1.017	1.030	**0.975**	1.011	1.015	1.005	1.017
loli.boost.ls	1.203	1.180	1.145	1.159	1.092	1.121	1.383	1.249	1.208	**1.082**	1.242	1.280	1.164
loli.tree.avg	1.050	1.067	1.150	1.046	1.085	1.038	1.103	1.024	1.021	**0.994**	1.036	1.019	1.018
loli.bag.avg	0.999	1.030	1.061	1.020	0.998	**0.998**	1.061	0.999	1.001	1.009	1.039	1.008	1.019
loli.bag.avg.1	0.999	1.029	1.061	1.019	0.998	**0.997**	1.061	0.999	1.001	1.009	1.039	1.008	1.019
loli.rf.avg	**0.987**	1.018	1.067	1.000	0.990	0.999	1.060	0.998	1.007	1.022	1.037	1.020	1.037
loli.rf.avg.1	0.991	1.000	1.018	**0.989**	0.993	0.994	1.020	0.991	1.001	1.003	1.013	1.007	1.010
loli.boost.avg	0.998	1.013	1.057	0.996	**0.987**	1.021	1.099	1.001	1.011	0.989	1.023	1.007	0.999
loli.tree.ls*	1.311	1.503	1.398	1.296	1.442	**1.250**	1.257	1.522	1.586	1.439	1.262	1.489	1.642
loli.bag.ls*	1.061	1.058	1.090	1.096	1.063	1.017	1.078	1.095	1.050	**1.007**	1.064	1.054	1.047
loli.bag.ls*.1	0.983	0.999	1.003	0.987	0.983	0.983	1.025	**0.978**	0.983	0.984	1.006	0.984	0.994
loli.rf.ls*	1.054	1.024	1.079	1.090	1.055	**1.018**	1.072	1.096	1.046	1.037	1.060	1.046	1.083
loli.rf.ls*.1	**0.979**	0.993	1.008	0.989	0.980	0.982	1.011	0.987	0.988	0.993	1.001	0.996	0.999
loli.boost.ls*	1.161	**1.005**	1.075	1.120	1.064	1.132	1.105	1.063	1.080	1.101	1.084	1.128	1.094
loli.tree.rid	1.083	**1.051**	1.222	1.207	1.301	1.407	1.178	1.384	1.453	1.310	1.475	1.422	1.488
loli.tree.rid.p	1.045	1.079	1.028	1.152	**0.980**	1.219	1.107	1.214	1.145	1.107	1.319	1.140	1.289
loli.bag.rid	1.035	1.048	1.081	1.062	1.037	1.027	1.072	1.074	1.037	**0.996**	1.046	1.041	1.072
loli.bag.rid.1	0.990	0.993	1.013	1.002	0.988	0.987	1.036	1.013	**0.985**	1.006	1.035	1.026	1.021
loli.rf.rid	-	-	-	-	-	-	-	-	1.034	**1.004**	1.035	1.028	1.073
loli.rf.rid.1	-	-	-	-	-	-	-	-	**0.983**	0.995	1.000	0.998	1.008
loli.boost.rid	1.184	**1.083**	1.109	1.097	1.113	1.091	1.247	1.196	1.123	1.085	1.186	1.114	1.200
ctree	1.000	0.991	1.000	0.991	1.000	0.981	1.001	1.000	**0.978**	0.978	1.007	0.978	0.996
bart	1.000	0.983	1.029	1.005	1.015	**0.970**	1.042	1.035	0.987	1.013	1.033	1.030	1.018
grf	0.984	0.998	1.021	1.000	0.994	0.984	1.006	**0.980**	0.990	1.018	1.006	0.988	1.010

Notes: See Table 4.6.

Similar to previous multivariate Monte Carlo simulations, tree ensembles from fully grown trees suffer from overfitting and therefore perform inferior to ensembles from tree stumps. Model set 2 with regressors ranked from least to most persistent does not perform as well as the other model sets.

Many local linear TBMs with predictions from node averages perform at least as well as historic mean predictions for $n = 60$, while most of the remaining TBMs are not able to compete with the historic mean. Methods based on local linear trees with predictions from regressions within nodes yield quite large relative MSEs. This is especially true when a larger amount of regressor variables is considered. These large MSEs are driven by only few large outliers; we address this issue later in Chapter 4.4.4.

Similar to the findings from univariate predictions in Chapter 4.4.1, most TBMs perform worst compared to historic mean predictions for a medium sample size of $n = 120$. Few TBMs are able to compete with the historic mean at this sample size for $m \in \{2, 4, 6\}$ regressor variables. However, no TBM is able to outperform the historic mean when the full set of regressor variables ($m = 8$) is employed.

Almost all TBMs achieve their best relative performance for rolling windows of size $n = 240$. Most TBMs are able to compete with the historic mean in terms of OOS MSEs. This does not hold for TBMs with model set 2, ensembles of fully grown trees and computations employing the full regressor set ($m = 8$). This does not necessarily mean that the TBMs perform well, the historic mean could rather perform poorly for larger sample sizes. We also need to keep in mind that predictions with $n = 240$ start in 2010 after the Great Recession, while predictions for shorter windows begin prior to this crisis.

In conclusion, predictions including all $m = 8$ regressor variables are not performing best in our comparison since they are outscored by several predictions with less than 8 regressor variables. Many TBMs even yield their best results when only one regressor is included, especially for small and medium sized windows.

4.4.3 Multivariate Predictions on Firmlevel

In this chapter, we employ firm-specific data from 212 companies which are listed in the S&P 500 stock market index between 1990M01 and 2019M08[2]. We perform two types of computations for each company. In the first set, we predict each company's stock returns depending on its firm-specific data, i.e. its dividend yield, price to book ratio and price to cash flow ratio. In a second set of predictions, we additionally consider the market-level regressor variables from Chapter 4.4.1. Unlike in Chapter 4.4.1, we do not perform additional predictions for any subsets of regressor variables. We skip this task due to computing constraints and leave it for future research.

[2]The data is available on Datastream; assembling and computation was kindly performed by Danvee Floro and Jasper Gross.

As in previous chapters, we perform OOS rolling window predictions for samples of 5, 10 and 20 years. Resulting OOS MSEs averaged over all 212 companies are displayed in Table 4.9 with previous color-coding scheme.

Table 4.9: Relative average OOS stock return forecast performance for 212 firms

Window Length	5	5	10	10	20	20
Market		✓		✓		✓
tree	1.046	1.088	1.073	1.138	1.110	1.199
tree.bag	1.237	1.189	1.238	1.168	1.221	1.114
tree.bag.1	1.080	1.078	1.052	1.052	1.023	1.023
tree.rf	1.157	1.132	1.153	1.118	1.146	1.081
tree.rf.1	1.041	1.040	1.022	1.022	1.004	1.004
tree.boost	1.113	1.301	1.108	1.253	1.140	1.328
loli.tree.ls	65.263	313.116	22.684	182.265	66.843	585.889
loli.bag.ls	36.821	24.353	8.667	27.742	37.462	139.634
loli.bag.ls.1	33.764	33.700	5.759	5.759	8.935	8.935
loli.rf.ls	25.902	23.498	9.597	8.320	27.758	18.660
loli.rf.ls.1	18.422	18.387	3.440	3.440	5.716	5.716
loli.boost.ls	248.915	119.349	65.525	68.758	233.892	182.421
loli.tree.avg	1.018	1.019	1.033	1.033	1.064	1.052
loli.bag.avg	1.001	1.001	1.004	1.005	1.012	1.007
loli.bag.avg.1	1.004	1.002	0.999	0.999	1.001	1.001
loli.rf.avg	1.000	1.001	1.003	1.003	1.010	1.006
loli.rf.avg.1	1.001	0.999	0.998	0.998	1.000	1.000
loli.boost.avg	1.013	1.016	1.019	1.026	1.017	1.014
loli.tree.ls*	7.110	1.141	10.560	4.417	48.952	104.091
loli.bag.ls*	2.202	1.086	3.123	1.245	26.786	32.635
loli.bag.ls*.1	2.118	2.114	1.257	1.257	1.335	1.335
loli.rf.ls*	2.350	1.070	3.139	1.200	19.787	2.828
loli.rf.ls*.1	1.706	1.703	1.200	1.200	1.205	1.205
loli.boost.ls*	4.941	1.706	4.976	3.039	6.665	1.950
loli.tree.rid	45.967	19.914	44.362	11.517	112.041	59.751
ctree	1.042	1.039	1.030	1.042	1.035	1.034
bart	1.084	1.093	1.094	1.102	1.092	1.101
grf	1.000	1.000	0.997	1.001	1.005	1.003

Notes: Average OOS MSE relative to average OOS MSE firm level data. Predictions applying market data employ 11 regressor variables, predictions without market data employ 3 firm specific regressors.

The best performing procedures over all considered setups are random forests and bagging of local linear stumps with predictions from node averages (loli.rf.avg.1 and loli.bag.avg.1), followed by generalized random forests. Note that only loli.rf.avg.1 yields an average relative MSE (slightly) below 1.

Over all combinations of TBMs and window lengths, the inclusion of market level data improves (decreases) the OOS predictions in 50% (31%) of the considered cases (42% (40%) when neglecting local linear trees). These improvements occur in small as well as in large samples. However, the effect size tends to be quite small (apart from local linear TBMs).

Local linear trees basing their predictions on least squares estimates within a terminal node fail completely in this setup. The performance slightly improves when predictions are based on split-relevant variable only (loli.ls*) but still remains rather poor. These large relative MSEs for local linear TBMs are driven by few extremely large outliers or structural breaks within the data of the corresponding predictions. We already observed this issue in previous chapters and address it in Chapter 4.4.4.

Similar to the univariate and multivariate predictions for S&P 500 data in previous sections, we find some TBMs performing on the same level as historic mean predictions. Nevertheless, no TBM is able to clearly outscore the historic mean in terms of OOS predictions in our comparison.

4.4.4 Prediction Thresholds

Comparisons in previous chapters have shown poor performances of TBMs with predictions from linear regression within terminal nodes. Large OOS MSEs of these methods are driven by only few outliers or structural breaks in regressor variables. This is exemplarily shown for stock return predictions of a single company from the data of Section 4.4.3 in Figure 4.4. One of the regressor variables jumps to a new level, immediately leading to a (false) jump of the predictions from local linear TBMs. However, TBMs with predictions from node averages are robust to these kind of structural breaks or outliers.

To address this issue, we decide to check the predictions for implausible values. We then replace local linear tree predictions $\hat{y}_t^{loli}$ by historic mean predictions $\hat{y}_t^{HM}$ if implausible threshold values θ_t are exceeded:

$$\hat{y}_t^{loli.t} = \begin{cases} \hat{y}_t^{loli} & \text{if } \theta_{1;t} < \hat{y}_t^{loli} < \theta_{2;t} \\ \hat{y}_t^{HM} & \text{else.} \end{cases} \tag{4.9}$$

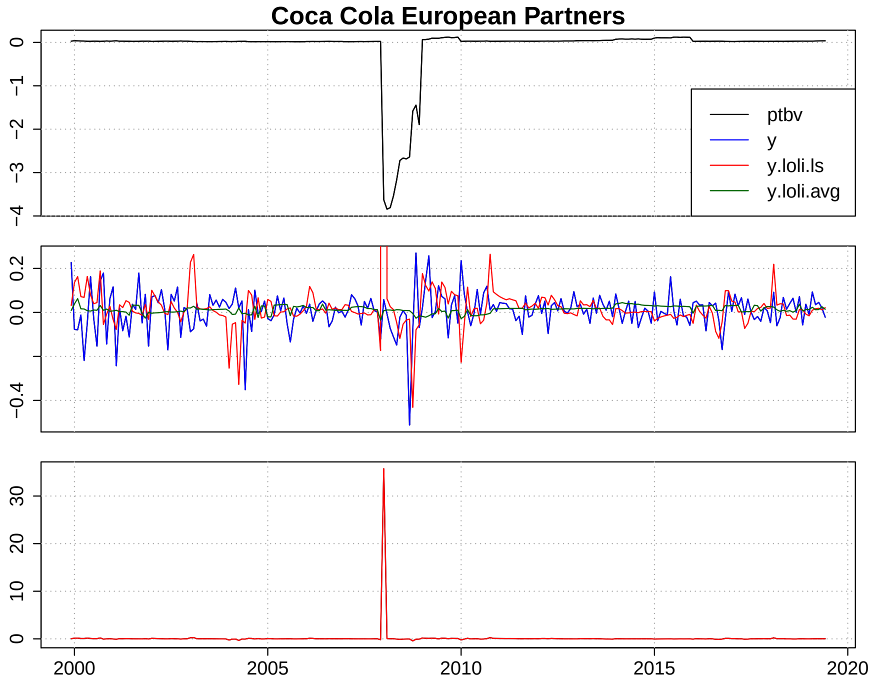
We allow the threshold values θ_t to evolve over time and choose to compute them depending on observed data from the recent 5 years. For the lower threshold $\theta_{1;t}$, we add 5 times the difference of mean and minimum of this data to its mean value. The upper threshold is computed using the maximum instead of the minimum:

$$\theta_{1;t+1} = \frac{1}{60}\sum_{i=t-59}^{t} y_i - 5\left(\frac{1}{60}\sum_{i=t-59}^{t} y_i - min\{y_{t-59}, \ldots, y_t\}\right) \tag{4.10}$$

$$\theta_{2;t+1} = \frac{1}{60}\sum_{i=t-59}^{t} y_i + 5\left(max\{y_{t-59}, \ldots, y_t\} - \frac{1}{60}\sum_{i=t-59}^{t} y_i\right) \tag{4.11}$$

The thresholds from (4.10) and (4.11) are violated by local linear TBMs with predictions from linear regressions within terminal nodes only, predictions from different TBMs do not

Figure 4.4: Outliers in data.



***Notes*:** Data from Coca Cola European Partners, listed in S&P 500. Upper panel shows regressor variable price to book ratio with a different level in 2007. Middle panel contains true stock returns (blue), predictions by loli.tree.ls (red) and loli.avg (green). Lower panel shows predictions of loli.tree.ls in different scaling to fully include outlier.

violate these bounds. Relative OOS MSEs for effected TBMs after adjusting the outliers are summarized in Table 4.10, the percentage of effected predictions for each TBM is displayed in Table 4.11.

Table 4.10: Stock return forecast performance for 212 firms after adjusting for outliers.

Window Length Market	5	5 ✓	10	10 ✓	20	20 ✓
loli.tree.ls	1.716	4.749	1.825	4.365	1.857	4.419
loli.bag.ls	1.571	2.566	1.504	2.290	1.451	2.495
loli.bag.ls.1	1.689	1.689	1.394	1.394	1.272	1.272
loli.rf.ls	1.502	2.280	1.454	1.989	1.465	2.159
loli.rf.ls.1	1.452	1.452	1.373	1.373	1.177	1.177
loli.boost.ls	2.526	5.118	2.675	3.596	2.478	3.720
loli.tree.ls*	1.193	1.138	1.513	1.459	1.730	1.884
loli.bag.ls*	1.119	1.052	1.245	1.118	1.401	1.657
loli.bag.ls*.1	1.109	1.109	1.078	1.078	1.083	1.083
loli.rf.ls*	1.120	1.029	1.258	1.088	1.467	1.465
loli.rf.ls*.1	1.105	1.105	1.066	1.066	1.039	1.039
loli.boost.ls*	1.170	1.132	1.386	1.313	1.300	1.211
loli.tree.rid	1.424	1.980	1.445	2.164	1.469	2.103

Notes: Average OOS MSE relative to average OOS MSE firm level data, thresholds are computed as 5-year-average plus (minus) 5 times the difference of 5-year-average and 5-year-maximum (5-year-minimum).

Table 4.11: Percentage of predictions effected by threshold procedure.

Window Length Market	5	5 ✓	10	10 ✓	20	20 ✓
loli.tree.ls	0.222	0.604	0.223	0.684	0.250	0.906
loli.bag.ls	0.147	0.185	0.141	0.173	0.230	0.283
loli.bag.ls.1	0.141	0.141	0.080	0.080	0.123	0.123
loli.rf.ls	0.125	0.150	0.104	0.161	0.193	0.209
loli.rf.ls.1	0.120	0.120	0.060	0.060	0.066	0.066
loli.boost.ls	0.416	0.886	0.468	0.508	0.591	0.853
loli.tree.ls*	0.032	0.002	0.100	0.044	0.205	0.164
loli.bag.ls*	0.027	0.005	0.074	0.020	0.180	0.086
loli.bag.ls*.1	0.029	0.029	0.020	0.020	0.041	0.041
loli.rf.ls*	0.026	0.006	0.068	0.024	0.156	0.086
loli.rf.ls*.1	0.021	0.021	0.030	0.030	0.041	0.041
loli.boost.ls*	0.035	0.003	0.108	0.032	0.139	0.021
loli.tree.rid	0.067	0.096	0.116	0.155	0.164	0.304

Notes: Percentage of predictions effected by threshold procedure for 212 firms with 295, 235 and 115 predictions each, depending on window length; e.g. 0.222% of 212x295 loli.tree.ls predictions w/o market and window length of 5 years are altered by threshold procedure.

After adjusting for outliers, OOS MSEs of local linear TBMs in Table 4.10 are considerably lower than corresponding MSEs in Table 4.9. However, they are still not able to compete with historic mean predictions. One might employ tighter threshold values to also correct small outliers, risking to falsely declare predictions as outliers in volatile business periods (e.g. Great Recession).

Additionally, we also adjusted the data from Chapter 4.4.2 for outliers. Though, in this case we apply tighter threshold values due to less volatility in the data. Thresholds are generated by applying a multiplication factor of 2 instead of 5 in Equations (4.10) and (4.11). Again, only methods from local linear trees are effected by this procedure, despite the use of tighter thresholds. The performance of the TBMs increases considerably after adjusting for outliers. However, they are not able to outperform the historic mean predictions. Relative OOS MSEs after adjusting for outliers are available upon request.

4.4.5 Combinations of Historic Mean and Tree Based Methods

In this section we aim to further improve the prediction accuracy of TBMs from previous chapters. We apply a diversification strategy by combining methods of different characteristics. This combination approach is supposed to reduce the risk of overfitting and therefore result in more accurate OOS predictions. We apply a weighted combination of TBM predictions with predictions from the historic mean. This could also be interpreted as shrinkage of the TBMs. In order to do so, we follow Zhang et al. (2020) with

$$\hat{y}_{comb,t+1} = \delta_{HM,t}\hat{y}_{HM,t+1} + \delta_{TBM,t}\hat{y}_{TBM,t+1} \tag{4.12}$$

where $\hat{y}_{HM}$ is the historic mean prediction as shrinkage target and $\hat{y}_{TBM}$ a tree based forecast. The performance based weight δ is the shrinkage factor which depends inversely on the recent performances of the combined methods (c.f. Stock and Watson (2004)) as

$$\delta_{i,t} = \phi_{i,t}^{-1} / \sum_{j=1}^{N} \phi_{j,t}^{-1} \tag{4.13}$$

with

$$\phi_{i,t} = \sum_{s=m}^{t-1} \theta^{t-1-s} (y_{s+1} - \hat{y}_{i,s+1})^2, \tag{4.14}$$

and i denoting either the historic mean or a tree based method. Following Rapach et al. (2010), we choose $\theta = 0.9$ for the discount parameter. Since the weights δ depend on past values, a holdout period of length $t - m$ is required. Rapach et al. (2010) suggest to employ a holdout period of 10 years for their quarterly data, i.e. 40 quarters. We choose to use a different length for the holdout period for two reasons: first, we apply monthly data and second, we only perform predictions for 295, 235 or 115 months, depending on the employed rolling window size. The length of the holdout period is a tradeoff between including more information in the weights δ and not losing too many observations for the predictions. We choose to settle with a holdout period of two years (24 months).

We apply the combination method for firm-specific, as well as for aggregated S&P 500 data, both after adjusting for outliers as described in the previous chapter. The resulting relative OOS MSEs for firm-level data can be found in Table 4.12, while results for S&P 500 data are shifted to the Appendix in Tables 4.26 to 4.28.

All relative OOS MSEs for the market as well as for the firm data reach values close to 1, indicating that no combination method performs overly better or worse than the historic mean in predicting stock returns. For the firm level data, combinations including local linear trees are still the least performing procedures in our comparison. Boosting of local linear trees yields the largest average relative OOS MSE (1.189). On the other hand, combinations with local linear TBMs with predictions from node averages perform best, e.g. loli.bag.avg.1 scores on average the lowest relative OOS MSE (0.998).

For market level data (Tables 4.26 - 4.28), the results differ for different sample lengths. Various combinations do a little better than historic mean predictions for short sample sizes of 60 months, while only few are able to do so for medium sized samples of a 120 month length. For large samples of 240 months, most combinations (except for model set 2) yield a relative OOS MSE below 1. We already observed similar results for these three different sample sizes in Chapter 4.4.2. Though, in this section it is even more pronounced. On average, combinations of historic mean and generalized random forest

Table 4.12: Average relative OOS stock return forecast performance for 212 firms

Window Length Market	5	5 ✓	10	10 ✓	20	20 ✓	∅
tree	1.004	1.017	1.014	1.029	1.020	1.038	1.020
tree.bag	1.047	1.040	1.053	1.033	1.038	1.024	1.039
tree.bag.1	1.013	1.013	1.011	1.011	1.004	1.004	1.009
tree.rf	1.033	1.028	1.038	1.025	1.027	1.018	1.028
tree.rf.1	1.006	1.006	1.006	1.006	1.000	1.000	1.004
tree.boost	1.018	1.057	1.021	1.050	1.026	1.070	1.040
loli.tree.ls	1.116	1.260	1.144	1.206	1.101	1.175	1.167
loli.bag.ls	1.103	1.171	1.101	1.140	1.052	1.076	1.107
loli.bag.ls.1	1.119	1.119	1.093	1.093	1.029	1.029	1.080
loli.rf.ls	1.096	1.166	1.096	1.155	1.046	1.057	1.103
loli.rf.ls.1	1.090	1.090	1.099	1.099	1.030	1.030	1.073
loli.boost.ls	1.185	1.226	1.202	1.197	1.151	1.172	1.189
loli.tree.avg	1.000	1.003	1.007	1.007	1.015	1.013	1.008
loli.bag.avg	0.997	1.000	1.000	1.001	1.001	1.001	1.000
loli.bag.avg.1	0.997	0.997	0.998	0.998	0.999	0.999	0.998
loli.rf.avg	0.996	1.000	0.999	1.000	1.000	1.000	0.999
loli.rf.avg.1	0.997	0.997	0.999	0.999	0.999	0.999	0.998
loli.boost.avg	0.999	1.002	1.004	1.006	1.003	1.002	1.003
loli.tree.ls*	1.039	1.032	1.108	1.102	1.094	1.106	1.080
loli.bag.ls*	1.024	1.012	1.056	1.028	1.062	1.015	1.033
loli.bag.ls*.1	1.021	1.021	1.020	1.020	1.010	1.010	1.017
loli.rf.ls*	1.023	1.004	1.062	1.019	1.057	1.015	1.030
loli.rf.ls*.1	1.021	1.021	1.017	1.017	1.009	1.009	1.016
loli.boost.ls*	1.035	1.031	1.079	1.072	1.045	1.021	1.047
loli.tree.ridge	1.088	1.146	1.089	1.156	1.069	1.123	1.112
ctree	1.008	1.009	1.006	1.011	1.007	1.009	1.008
bart	1.015	1.017	1.023	1.019	1.018	1.020	1.019
grf	1.000	1.000	0.998	1.000	0.999	1.000	1.000

Notes: Average OOS MSE of combination relative to average OOS MSE firm level data after adjusting for outliers.

perform best over all possible TBMs, sample sizes and regressor variables with an average relative OOS MSE of 0.998. The overall best performance is achieved by historic mean combined with boosting of local linear trees (loli.boost.ls) at window length of $n = 240$ months with $m = 6$ regressor variables from model set 1 and a relative MSE of 0.934.

We exemplarily show the DCSFEs of historic mean and TBM predictions for model set 1 in Figure 4.5, applying the same color scheme as in Figure 4.3. These plots illustrate the performance of predictions from combinations of the historic mean predictions with different TBMs. The DCSFEs of some combinations experience considerable drops at particular points in time, despite adjusting for outliers in advance. The timing of these drops mainly coincides with financial crises. Finding the cause for drops at other times is left for future research.

Furthermore, we exemplarily display the values of the weights δ computed for model set 1 in Figure 4.6 in the Appendix. We interpret these weights as a measure indicating how well a tree based method performed compared to the historic mean in predicting stock returns in the past two years.

4.5 Concluding remarks

In this essay, we discussed different tree based methods and their accuracy in univariate as well as multivariate stock return predictions. We employed two different types of data in this process, market-level and firm-level data and performed rolling window predictions for different sample sizes.

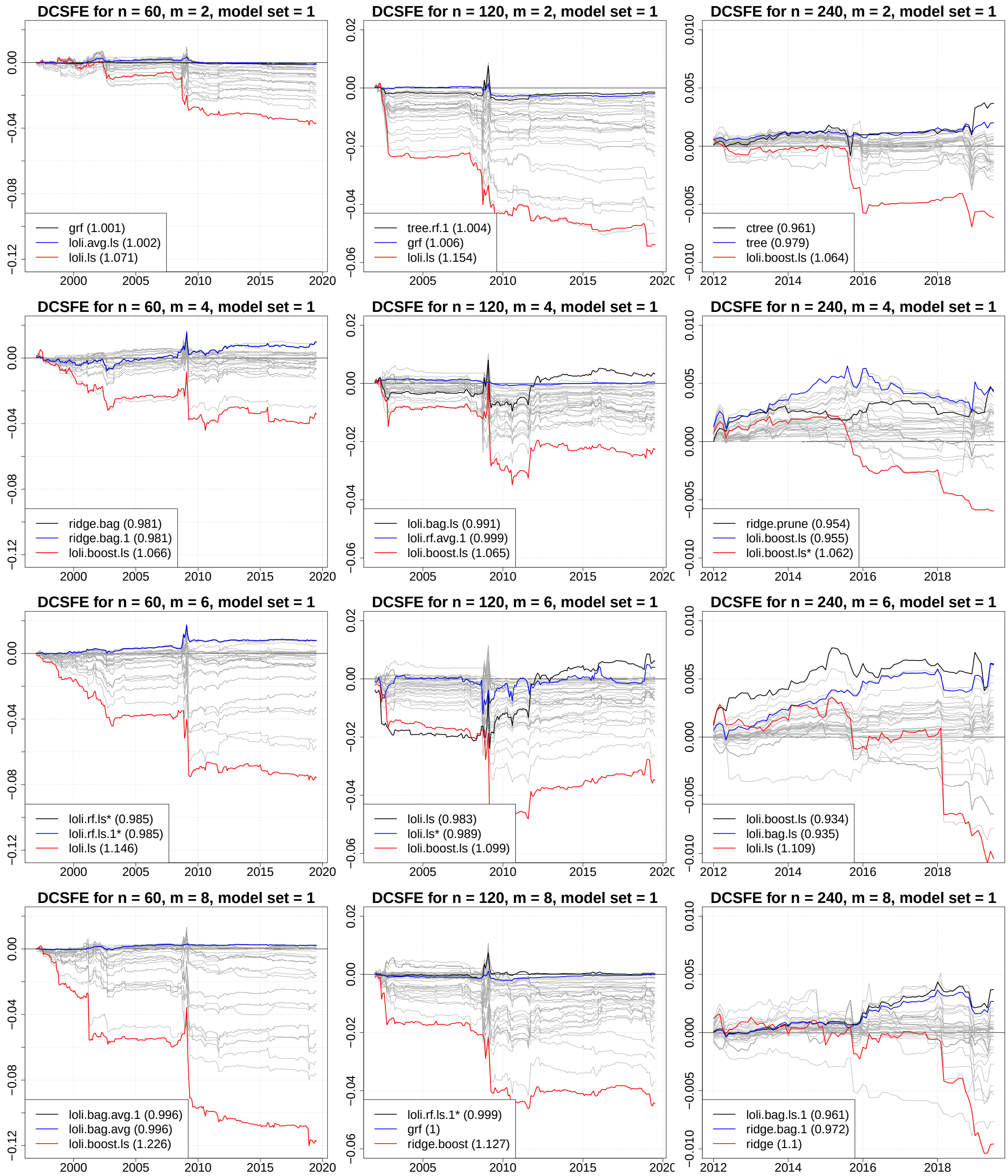
We find that some tree based methods are able to outperform the historic mean in terms of prediction accuracy. However, we find no single best method or set of best predictor variables.

Local linear trees can handle in-sample outliers reasonably well. However, they severely suffer from out-of-sample outliers. Methods with predictions from node averages, on the other hand, are robust to outliers.

Despite having a built-in predictor selector, employing all possible regressor variables does not improve OOS MSE in our comparison. TBMs oftentimes perform better in univariate predictions than in any multivariate prediction. Investigating whether this is caused by multiple regressors leading to overfitting and trying to preventing this issue by cross validation is left for future research.

To conclude, we find potential in TBMs for stock market behavior. Though, we advise apply to cross validations for more suitable parameter selection. Also, the model selection and estimation process uses up a lot of information. Thus, additional shrinkage of trees might improve predictions in the presence weak signals.

Figure 4.5: Difference in cumulated squared forecast errors for model set = 1.



Notes: Difference in cumulated squared forecast errors of historic mean and TBM (DCSFE) for different TBMs over time for market level data and model set = 1; top to bottom: number of regressor variables = 2,4,6 and 8; left to right: window length = 60, 120 and 240 months. Two best and least performing procedures (w.r.t. OOS MSE) are highlighted in black, blue and red, respectively; corresponding relative MSE values are depicted in legend.

Appendix

Additional figures

Table 4.13: Relative univariate IS forecasting performance.

c	0	0	5	5	15	15
b	5	20	5	20	5	20
$T = 100$						
tree	0.688	0.417	0.729	0.576	0.742	0.662
tree.bag	0.295	0.172	0.315	0.243	0.323	0.286
tree.boost	0.775	0.457	0.829	0.645	0.845	0.750
loli.tree.ls	0.687	0.405	0.738	0.570	0.757	0.666
loli.bag.ls	0.790	0.470	0.845	0.660	0.858	0.763
loli.boost.ls	0.592	0.350	0.639	0.491	0.656	0.577
loli.tree.avg	0.832	0.509	0.886	0.703	0.897	0.803
loli.bag.avg	0.789	0.469	0.844	0.659	0.858	0.762
loli.boost.avg	0.917	0.663	0.952	0.823	0.947	0.896
ctree	0.866	0.482	0.947	0.706	0.979	0.837
bart	1.164	1.774	1.082	1.306	1.056	1.161
grf.honest	0.825	0.480	0.885	0.683	0.904	0.798
$T = 500$						
tree	0.951	0.818	0.971	0.905	0.980	0.944
tree.bag	0.313	0.271	0.317	0.299	0.319	0.311
tree.boost	0.931	0.803	0.946	0.890	0.950	0.926
loli.tree.ls	0.922	0.803	0.935	0.882	0.941	0.918
loli.bag.ls	0.834	0.723	0.845	0.798	0.848	0.828
loli.boost.ls	0.881	0.768	0.894	0.844	0.900	0.878
loli.tree.avg	0.969	0.883	0.979	0.946	0.981	0.968
loli.bag.avg	0.833	0.723	0.845	0.798	0.848	0.827
loli.boost.avg	0.986	0.923	0.992	0.969	0.991	0.986
ctree	0.969	0.830	0.988	0.924	0.995	0.963
bart	1.063	1.284	1.036	1.096	1.030	1.051
grf.honest	0.887	0.770	0.900	0.850	0.904	0.881

Notes: Average IS MSEs relative to average IS MSEs of historic mean for DGPs with different degrees of persistence.

Table 4.14: Relative multivariate OOS forecasting performance for $T = 60$.

m	2				4				6				8
model	0	1	2	3	0	1	2	3	0	1	2	3	all
tree	1.041	1.029	1.051	1.021	1.041	1.045	1.071	1.049	1.064	1.084	1.109	1.085	1.070
tree.bag	1.177	1.146	1.126	1.151	1.102	1.071	1.181	1.087	1.100	1.152	1.152	1.100	1.085
tree.bag.1	1.084	1.052	1.049	1.063	1.041	1.025	1.093	1.009	1.030	1.069	1.063	1.043	1.021
tree.rf	1.133	1.102	1.100	1.116	1.039	1.033	1.093	1.041	1.049	1.090	1.096	1.065	1.034
tree.rf.1	1.041	1.031	1.025	1.043	0.998	1.002	1.030	0.995	1.007	1.038	1.029	1.020	1.004
tree.boost	1.030	1.042	1.051	1.023	1.079	1.020	1.075	1.049	1.086	1.086	1.094	1.095	1.118
loli.tree.ls	1.257	1.213	1.132	1.209	1.298	1.298	1.406	1.414	1.543	1.512	1.713	1.678	2.005
loli.bag.ls	1.172	1.119	1.082	1.150	1.176	1.140	1.215	1.215	1.240	1.211	1.335	1.312	1.398
loli.bag.ls.1	1.172	1.119	1.082	1.150	1.176	1.140	1.215	1.215	1.240	1.211	1.335	1.312	1.398
loli.rf.ls	1.142	1.092	1.067	1.135	1.141	1.117	1.169	1.166	1.207	1.187	1.266	1.251	1.332
loli.rf.ls.1	1.142	1.092	1.067	1.135	1.141	1.117	1.169	1.166	1.207	1.187	1.266	1.251	1.332
loli.boost.ls	1.304	1.225	1.210	1.225	1.455	1.366	1.528	1.544	1.864	1.721	2.118	1.849	2.329
loli.tree.avg	1.039	1.043	1.008	1.003	1.037	1.021	1.009	1.016	1.028	1.026	1.038	1.014	1.001
loli.bag.avg	1.016	1.018	1.004	1.004	1.007	0.997	1.008	1.003	1.006	1.004	1.002	1.006	0.996
loli.bag.avg.1	1.016	1.018	1.004	1.004	1.008	0.997	1.007	1.003	1.006	1.004	1.002	1.006	0.995
loli.rf.avg	1.006	1.012	1.006	1.002	0.993	0.987	1.003	0.996	1.001	1.005	1.004	1.006	0.994
loli.rf.avg.1	1.006	1.012	1.006	1.002	0.993	0.987	1.003	0.996	1.001	1.005	1.004	1.006	0.994
loli.boost.avg	1.024	1.036	1.007	0.992	1.021	1.007	1.013	1.015	1.025	1.014	1.018	1.012	1.006
loli.tree.ls*	1.167	1.133	1.083	1.124	1.063	1.124	1.141	1.089	1.126	1.087	1.181	1.085	1.041
loli.bag.ls*	1.098	1.060	1.042	1.078	1.008	1.022	1.053	1.018	1.029	1.030	1.054	1.011	1.005
loli.bag.ls*.1	1.098	1.060	1.042	1.078	1.008	1.022	1.053	1.018	1.029	1.030	1.054	1.011	1.005
loli.rf.ls*	1.066	1.038	1.035	1.071	0.998	1.009	1.032	1.009	1.017	1.032	1.037	1.011	1.005
loli.rf.ls*.1	1.066	1.038	1.035	1.071	0.998	1.009	1.032	1.009	1.017	1.032	1.037	1.011	1.005
loli.boost.ls*	1.126	1.120	1.088	1.122	1.050	1.106	1.106	1.072	1.105	1.078	1.155	1.028	1.050
loli.tree.rid	1.156	1.108	1.064	1.125	1.199	1.140	1.215	1.201	1.284	1.190	1.360	1.363	1.413
loli.tree.rid.p	-	-	-	-	-	-	-	-	-	-	-	-	-
loli.bag.rid	1.110	1.070	1.042	1.087	1.102	1.080	1.127	1.114	1.143	1.124	1.218	1.180	1.212
loli.bag.rid.1	1.110	1.070	1.042	1.087	1.102	1.080	1.127	1.114	1.143	1.124	1.218	1.180	1.212
loli.rf.rid	-	-	-	-	-	-	-	-	1.076	1.072	1.118	1.097	1.116
loli.rf.rid.1	-	-	-	-	-	-	-	-	1.076	1.072	1.118	1.097	1.116
loli.boost.rid	1.183	1.131	1.104	1.140	1.191	1.114	1.218	1.202	1.223	1.221	1.348	1.319	1.305
ctree	1.027	1.032	1.002	1.028	1.020	1.016	1.010	1.007	1.017	1.049	1.030	1.018	1.010
bart	1.130	1.084	1.073	1.073	1.062	1.045	1.108	1.045	1.050	1.065	1.088	1.060	1.075
grf	0.999	1.002	1.004	0.999	1.003	1.004	1.007	1.002	1.001	1.000	1.000	0.997	1.001

Notes: See Table 4.4

Table 4.15: Relative multivariate OOS forecasting performance for $T = 120$.

m	2				4				6				8
model	0	1	2	3	0	1	2	3	0	1	2	3	all
tree	1.026	1.026	1.093	1.031	1.081	1.034	1.108	1.098	1.094	1.063	1.112	1.083	1.114
tree.bag	1.127	1.072	1.089	1.103	1.158	1.050	1.085	1.122	1.124	1.090	1.118	1.085	1.077
tree.bag.1	1.010	0.998	1.006	0.999	1.037	1.007	1.005	1.018	1.021	1.002	1.034	1.000	1.001
tree.rf	1.097	1.052	1.070	1.075	1.092	1.030	1.047	1.069	1.082	1.055	1.081	1.051	1.048
tree.rf.1	0.998	0.992	1.002	0.998	1.015	1.000	0.997	1.000	1.011	0.997	1.014	0.994	0.995
tree.boost	1.023	1.042	1.042	1.006	1.063	1.056	1.058	1.073	1.077	1.062	1.037	1.065	1.081
loli.tree.ls	1.232	1.156	1.173	1.136	1.578	1.300	1.300	1.478	1.588	1.808	1.591	1.563	1.678
loli.bag.ls	1.079	1.036	1.067	1.066	1.209	1.038	1.117	1.155	1.198	1.160	1.141	1.199	1.182
loli.bag.ls.1	1.027	1.024	1.037	1.017	1.116	1.014	1.064	1.101	1.121	1.100	1.083	1.135	1.144
loli.rf.ls	1.060	1.030	1.047	1.055	1.140	1.017	1.070	1.112	1.146	1.113	1.108	1.153	1.174
loli.rf.ls.1	1.022	1.010	1.026	1.015	1.081	1.010	1.040	1.067	1.097	1.064	1.063	1.101	1.125
loli.boost.ls	1.305	1.141	1.233	1.203	1.689	1.318	1.462	1.554	1.715	1.673	1.591	1.808	1.744
loli.tree.avg	1.015	1.033	1.039	1.012	1.054	1.033	1.036	1.060	1.024	1.033	1.042	1.022	1.062
loli.bag.avg	1.006	1.015	1.033	0.997	1.016	1.009	1.012	1.022	1.005	1.005	1.005	1.000	1.010
loli.bag.avg.1	1.006	1.015	1.034	0.997	1.016	1.009	1.012	1.022	1.005	1.005	1.005	1.000	1.010
loli.rf.avg	1.008	1.009	1.017	1.001	1.012	1.008	1.009	1.017	1.012	1.009	1.014	1.003	1.014
loli.rf.avg.1	1.003	1.001	1.006	1.001	1.003	1.004	1.003	1.009	1.006	1.002	1.008	1.000	1.006
loli.boost.avg	1.024	1.031	1.023	1.005	1.032	1.020	1.036	1.042	1.004	1.022	1.030	1.002	1.040
loli.tree.ls*	1.189	1.131	1.156	1.109	1.371	1.149	1.183	1.313	1.231	1.261	1.212	1.204	1.246
loli.bag.ls*	1.053	1.017	1.050	1.047	1.091	1.001	1.038	1.067	1.046	1.018	1.031	1.037	1.022
loli.bag.ls*.1	1.010	0.996	1.015	0.998	1.019	1.004	0.999	1.015	1.010	1.000	1.007	1.007	1.004
loli.rf.ls*	1.043	1.009	1.035	1.039	1.054	1.004	1.019	1.043	1.034	1.018	1.033	1.024	1.020
loli.rf.ls*.1	1.007	0.985	1.008	0.999	1.014	1.001	0.999	1.002	1.005	1.004	1.010	0.997	0.998
loli.boost.ls*	1.192	1.074	1.136	1.156	1.246	1.075	1.125	1.205	1.166	1.158	1.142	1.111	1.153
loli.tree.rid	1.119	1.068	1.123	1.090	1.330	1.153	1.251	1.254	1.285	1.426	1.359	1.254	1.322
loli.tree.rid.p	1.116	1.042	1.097	1.086	1.255	1.101	1.169	1.180	1.215	1.330	1.254	1.264	1.274
loli.bag.rid	1.049	1.018	1.052	1.037	1.143	1.012	1.078	1.093	1.128	1.104	1.087	1.122	1.123
loli.bag.rid.1	1.015	1.007	1.025	1.004	1.077	1.007	1.037	1.060	1.082	1.064	1.046	1.082	1.087
loli.rf.rid	-	-	-	-	-	-	-	-	1.070	1.044	1.042	1.061	1.080
loli.rf.rid.1	-	-	-	-	-	-	-	-	1.044	1.025	1.021	1.037	1.050
loli.boost.rid	1.198	1.100	1.149	1.144	1.414	1.122	1.272	1.342	1.401	1.370	1.321	1.400	1.372
ctree	1.000	1.002	0.999	1.000	1.000	1.009	1.003	1.002	1.006	1.016	1.017	1.008	1.000
bart	1.038	1.022	1.035	1.023	1.076	1.025	1.037	1.062	1.078	1.069	1.075	1.037	1.068
grf	0.991	0.989	1.001	0.985	1.002	0.999	0.999	0.995	1.002	0.991	1.004	0.991	0.996

Notes: See Table 4.4

Table 4.16: Relative multivariate OOS forecasting performance for $T = 480$.

m	2				4				6				8
model	0	1	2	3	0	1	2	3	0	1	2	3	all
tree	1.017	1.017	1.002	1.029	1.068	1.084	0.994	1.089	1.051	1.136	1.088	1.047	1.207
tree.bag	1.094	1.102	1.067	1.074	1.029	1.098	1.042	1.084	1.077	1.115	1.130	1.065	1.148
tree.bag.1	1.001	1.000	0.997	0.998	1.007	1.000	1.000	1.005	0.979	1.022	0.998	0.997	1.007
tree.rf	1.074	1.085	1.052	1.049	1.023	1.071	1.023	1.052	1.063	1.081	1.074	1.053	1.106
tree.rf.1	0.992	0.998	0.998	0.998	1.006	0.997	1.004	1.005	0.992	1.018	0.992	0.993	1.006
tree.boost	1.042	1.057	1.024	1.042	1.139	1.076	1.018	1.051	1.154	1.142	1.115	1.084	1.117
loli.tree.ls	1.186	1.189	1.130	1.124	1.325	1.431	1.114	1.453	1.497	1.596	1.838	2.057	1.846
loli.bag.ls	1.077	1.076	1.061	1.065	1.069	1.125	0.993	1.094	1.100	1.066	1.150	1.155	1.169
loli.bag.ls.1	1.003	1.009	1.013	1.011	1.013	1.029	0.975	1.046	1.017	1.044	1.014	1.053	1.074
loli.rf.ls	1.057	1.064	1.053	1.053	1.037	1.105	1.005	1.071	1.069	1.077	1.112	1.128	1.150
loli.rf.ls.1	0.999	1.005	1.008	1.006	1.025	1.021	0.984	1.039	1.018	1.050	1.021	1.069	1.050
loli.boost.ls	1.091	1.085	1.049	1.063	1.117	1.172	1.023	1.144	1.268	1.229	1.395	1.380	1.320
loli.tree.avg	1.028	1.011	1.043	1.038	1.092	1.054	0.996	1.120	1.053	1.021	1.099	1.050	1.053
loli.bag.avg	1.020	1.014	1.025	1.027	1.045	1.029	0.979	1.035	1.005	1.012	1.018	0.999	1.016
loli.bag.avg.1	1.020	1.014	1.025	1.026	1.044	1.028	0.979	1.035	1.005	1.011	1.018	0.999	1.016
loli.rf.avg	1.017	1.008	1.014	1.022	1.038	1.019	0.997	1.025	0.994	1.015	1.009	0.987	1.022
loli.rf.avg.1	0.995	0.993	0.999	1.003	1.001	1.000	0.999	1.004	0.998	1.000	1.000	0.990	0.998
loli.boost.avg	0.994	0.993	1.001	1.004	1.023	1.007	0.991	1.009	1.004	1.029	0.977	0.999	0.988
loli.tree.ls*	1.186	1.191	1.135	1.122	1.313	1.429	1.117	1.462	1.337	1.492	1.794	2.001	1.881
loli.bag.ls*	1.077	1.076	1.061	1.065	1.071	1.121	0.994	1.092	1.091	1.051	1.132	1.129	1.102
loli.bag.ls*.1	0.997	1.000	1.002	1.002	1.010	0.993	1.005	1.012	1.002	1.009	0.997	1.006	0.996
loli.rf.ls*	1.057	1.064	1.053	1.053	1.038	1.101	1.005	1.073	1.060	1.067	1.093	1.109	1.111
loli.rf.ls*.1	0.993	0.997	1.000	1.000	1.012	0.997	1.004	1.014	0.998	1.007	1.005	0.993	0.996
loli.boost.ls*	1.029	1.043	1.011	1.028	0.979	1.058	0.953	1.094	1.022	1.022	1.001	1.115	1.000
loli.tree.rid	1.108	1.113	1.062	1.091	1.262	1.320	0.931	1.226	1.531	1.256	1.630	1.548	1.312
loli.tree.rid.p	1.033	1.055	1.012	1.034	1.035	1.132	0.955	1.133	1.025	1.147	1.225	1.097	1.114
loli.bag.rid	1.058	1.054	1.044	1.043	1.039	1.088	0.984	1.039	1.101	1.055	1.110	1.076	1.121
loli.bag.rid.1	0.999	1.006	1.010	1.008	1.013	1.020	0.977	1.030	1.016	1.040	0.995	1.025	1.041
loli.rf.rid	-	-	-	-	-	-	-	-	1.063	1.030	1.057	1.059	1.075
loli.rf.rid.1	-	-	-	-	-	-	-	-	1.012	1.033	1.003	1.017	1.025
loli.boost.rid	1.067	1.075	1.039	1.049	1.062	1.122	0.992	1.073	1.211	1.147	1.189	1.274	1.148
ctree	1.008	1.025	0.998	1.001	1.019	1.020	0.985	1.011	0.954	1.035	1.042	0.990	1.025
bart	1.031	1.054	1.013	1.016	1.063	1.043	1.031	1.039	1.088	1.069	1.087	1.054	1.081
grf	1.002	1.005	1.008	1.008	1.020	1.007	0.989	1.013	0.993	1.018	1.010	0.997	1.006

Notes: See Table 4.4

Table 4.17: Relative multivariate IS forecasting performance for $T = 60$.

m	2				4				6				8
model	0	1	2	3	0	1	2	3	0	1	2	3	all
tree	0.951	0.947	0.953	0.955	0.935	0.935	0.934	0.935	0.924	0.927	0.922	0.926	0.915
tree.bag	0.645	0.639	0.646	0.654	0.584	0.584	0.576	0.587	0.546	0.547	0.537	0.548	0.519
tree.bag.1	0.868	0.861	0.871	0.880	0.839	0.840	0.834	0.843	0.819	0.822	0.813	0.821	0.803
tree.rf	0.675	0.668	0.675	0.684	0.654	0.655	0.648	0.657	0.597	0.599	0.588	0.599	0.587
tree.rf.1	0.885	0.879	0.887	0.894	0.879	0.880	0.877	0.880	0.850	0.853	0.846	0.851	0.845
tree.boost	0.917	0.916	0.921	0.919	0.866	0.864	0.863	0.866	0.822	0.826	0.820	0.824	0.783
loli.tree.ls	0.860	0.853	0.868	0.864	0.756	0.751	0.755	0.756	0.656	0.659	0.656	0.657	0.573
loli.bag.ls	0.850	0.844	0.859	0.858	0.732	0.728	0.728	0.734	0.619	0.623	0.617	0.624	0.528
loli.bag.ls.1	0.850	0.844	0.859	0.858	0.732	0.728	0.728	0.734	0.619	0.623	0.617	0.624	0.528
loli.rf.ls	0.866	0.859	0.872	0.870	0.771	0.767	0.769	0.772	0.660	0.661	0.657	0.663	0.565
loli.rf.ls.1	0.866	0.859	0.872	0.870	0.771	0.767	0.769	0.772	0.660	0.661	0.657	0.663	0.565
loli.boost.ls	0.735	0.728	0.744	0.760	0.477	0.477	0.457	0.493	0.255	0.263	0.239	0.275	0.124
loli.tree.avg	0.969	0.970	0.972	0.971	0.970	0.971	0.970	0.970	0.974	0.975	0.975	0.975	0.974
loli.bag.avg	0.960	0.957	0.961	0.962	0.957	0.956	0.957	0.957	0.957	0.959	0.957	0.958	0.959
loli.bag.avg.1	0.960	0.957	0.961	0.962	0.957	0.956	0.957	0.957	0.957	0.959	0.957	0.958	0.959
loli.rf.avg	0.962	0.959	0.963	0.965	0.959	0.958	0.959	0.958	0.953	0.955	0.952	0.954	0.952
loli.rf.avg.1	0.962	0.959	0.963	0.965	0.959	0.958	0.959	0.958	0.953	0.955	0.952	0.954	0.952
loli.boost.avg	0.969	0.968	0.969	0.967	0.971	0.971	0.971	0.969	0.973	0.975	0.974	0.974	0.973
loli.tree.ls*	0.906	0.903	0.909	0.913	0.909	0.911	0.907	0.910	0.913	0.917	0.911	0.914	0.917
loli.bag.ls*	0.894	0.888	0.894	0.903	0.885	0.886	0.882	0.888	0.883	0.885	0.882	0.886	0.885
loli.bag.ls*.1	0.894	0.888	0.894	0.903	0.885	0.886	0.882	0.888	0.883	0.885	0.882	0.886	0.885
loli.rf.ls*	0.900	0.894	0.901	0.908	0.893	0.893	0.892	0.894	0.877	0.880	0.875	0.879	0.871
loli.rf.ls*.1	0.900	0.894	0.901	0.908	0.893	0.893	0.892	0.894	0.877	0.880	0.875	0.879	0.871
loli.boost.ls*	0.869	0.864	0.864	0.877	0.878	0.887	0.877	0.886	0.891	0.896	0.887	0.898	0.904
loli.tree.rid	0.883	0.877	0.893	0.886	0.810	0.802	0.809	0.808	0.741	0.745	0.738	0.742	0.693
loli.tree.rid.p	-	-	-	-	-	-	-	-	-	-	-	-	-
loli.bag.rid	0.960	0.958	0.962	0.964	0.956	0.956	0.956	0.956	0.949	0.951	0.949	0.950	0.947
loli.bag.rid.1	0.960	0.958	0.962	0.964	0.956	0.956	0.956	0.956	0.949	0.951	0.949	0.950	0.947
loli.rf.rid	-	-	-	-	-	-	-	-	0.760	0.763	0.752	0.768	0.683
loli.rf.rid.1	-	-	-	-	-	-	-	-	0.760	0.763	0.752	0.768	0.683
loli.boost.rid	0.807	0.800	0.814	0.819	0.665	0.662	0.656	0.672	0.544	0.554	0.536	0.561	0.468
ctree	0.989	0.984	0.993	0.992	0.989	0.992	0.994	0.991	0.991	0.989	0.989	0.989	0.991
bart	1.084	1.083	1.058	1.089	1.090	1.096	1.076	1.090	1.090	1.090	1.085	1.101	1.096
grf	1.001	1.001	1.001	1.001	1.001	1.001	1.001	1.001	1.001	1.001	1.001	1.001	1.001

Notes: See Table 4.4

Table 4.18: Relative multivariate IS forecasting performance for $T = 120$.

m	2				4				6				8
model	0	1	2	3	0	1	2	3	0	1	2	3	all
tree	0.928	0.919	0.929	0.933	0.906	0.904	0.900	0.904	0.888	0.889	0.886	0.889	0.877
tree.bag	0.607	0.598	0.605	0.613	0.543	0.541	0.536	0.544	0.504	0.504	0.499	0.506	0.479
tree.bag.1	0.928	0.918	0.928	0.933	0.910	0.908	0.906	0.911	0.896	0.897	0.894	0.897	0.888
tree.rf	0.637	0.626	0.634	0.644	0.613	0.610	0.607	0.615	0.555	0.555	0.551	0.557	0.545
tree.rf.1	0.937	0.928	0.937	0.940	0.932	0.930	0.930	0.932	0.915	0.915	0.913	0.914	0.913
tree.boost	0.901	0.894	0.902	0.909	0.860	0.859	0.854	0.861	0.824	0.826	0.822	0.827	0.797
loli.tree.ls	0.837	0.828	0.841	0.842	0.733	0.728	0.729	0.732	0.635	0.638	0.630	0.636	0.548
loli.bag.ls	0.826	0.815	0.829	0.836	0.702	0.699	0.696	0.703	0.589	0.595	0.586	0.595	0.501
loli.bag.ls.1	0.907	0.895	0.909	0.911	0.839	0.836	0.835	0.838	0.773	0.777	0.771	0.773	0.717
loli.rf.ls	0.846	0.836	0.847	0.854	0.752	0.748	0.746	0.754	0.637	0.641	0.634	0.642	0.542
loli.rf.ls.1	0.917	0.906	0.919	0.920	0.866	0.862	0.863	0.865	0.801	0.804	0.800	0.801	0.746
loli.boost.ls	0.581	0.564	0.578	0.652	0.284	0.285	0.259	0.306	0.121	0.125	0.111	0.138	0.050
loli.tree.avg	0.959	0.953	0.957	0.958	0.956	0.956	0.954	0.955	0.960	0.957	0.955	0.958	0.963
loli.bag.avg	0.939	0.929	0.936	0.943	0.932	0.931	0.927	0.932	0.933	0.934	0.931	0.934	0.937
loli.bag.avg.1	0.939	0.929	0.936	0.943	0.932	0.931	0.928	0.932	0.933	0.934	0.931	0.934	0.937
loli.rf.avg	0.943	0.935	0.941	0.945	0.936	0.935	0.933	0.935	0.927	0.927	0.925	0.927	0.925
loli.rf.avg.1	0.972	0.967	0.971	0.974	0.969	0.969	0.967	0.969	0.966	0.967	0.965	0.966	0.967
loli.boost.avg	0.955	0.952	0.953	0.961	0.959	0.957	0.957	0.957	0.963	0.960	0.959	0.963	0.966
loli.tree.ls*	0.855	0.847	0.859	0.857	0.827	0.820	0.827	0.823	0.825	0.826	0.824	0.827	0.834
loli.bag.ls*	0.839	0.827	0.841	0.847	0.793	0.790	0.787	0.793	0.779	0.782	0.777	0.782	0.780
loli.bag.ls*.1	0.935	0.925	0.933	0.937	0.929	0.926	0.927	0.928	0.929	0.930	0.928	0.930	0.933
loli.rf.ls*	0.855	0.845	0.856	0.862	0.818	0.814	0.814	0.818	0.782	0.783	0.781	0.782	0.768
loli.rf.ls*.1	0.935	0.926	0.935	0.939	0.929	0.925	0.927	0.927	0.921	0.921	0.920	0.920	0.921
loli.boost.ls*	0.711	0.692	0.717	0.733	0.721	0.714	0.703	0.715	0.737	0.746	0.733	0.750	0.765
loli.tree.rid	0.860	0.850	0.863	0.865	0.777	0.775	0.773	0.778	0.707	0.710	0.699	0.707	0.645
loli.tree.rid.p	0.882	0.873	0.886	0.887	0.820	0.822	0.818	0.818	0.762	0.771	0.763	0.766	0.718
loli.bag.rid	0.941	0.934	0.940	0.944	0.933	0.932	0.930	0.933	0.923	0.923	0.922	0.924	0.921
loli.bag.rid.1	0.971	0.966	0.970	0.973	0.968	0.967	0.965	0.968	0.964	0.965	0.963	0.964	0.964
loli.rf.rid	-	-	-	-	-	-	-	-	0.732	0.736	0.724	0.737	0.655
loli.rf.rid.1	-	-	-	-	-	-	-	-	0.848	0.851	0.843	0.849	0.803
loli.boost.rid	0.688	0.674	0.686	0.725	0.472	0.470	0.450	0.486	0.321	0.326	0.305	0.343	0.229
ctree	0.994	0.988	0.995	0.995	0.994	0.993	0.990	0.993	0.991	0.992	0.990	0.990	0.993
bart	1.071	1.081	1.056	1.057	1.078	1.066	1.078	1.081	1.082	1.078	1.077	1.091	1.085
grf	0.980	0.973	0.981	0.980	0.978	0.977	0.977	0.976	0.977	0.977	0.977	0.976	0.977

Notes: See Table 4.4

Table 4.19: Relative multivariate IS forecasting performance for $T = 240$.

m	2				4				6				8
model	0	1	2	3	0	1	2	3	0	1	2	3	all
tree	0.928	0.920	0.937	0.936	0.898	0.895	0.889	0.895	0.872	0.871	0.872	0.873	0.857
tree.bag	0.582	0.576	0.584	0.589	0.519	0.518	0.511	0.521	0.480	0.480	0.475	0.484	0.455
tree.bag.1	0.955	0.949	0.960	0.961	0.948	0.946	0.945	0.949	0.939	0.941	0.939	0.941	0.935
tree.rf	0.612	0.605	0.613	0.621	0.589	0.588	0.583	0.592	0.530	0.531	0.527	0.535	0.521
tree.rf.1	0.960	0.955	0.965	0.965	0.960	0.959	0.960	0.961	0.950	0.951	0.950	0.951	0.950
tree.boost	0.835	0.827	0.838	0.842	0.772	0.771	0.763	0.775	0.722	0.723	0.715	0.729	0.684
loli.tree.ls	0.817	0.811	0.824	0.826	0.708	0.707	0.705	0.710	0.611	0.610	0.608	0.611	0.525
loli.bag.ls	0.798	0.792	0.803	0.811	0.670	0.669	0.663	0.675	0.557	0.558	0.554	0.562	0.464
loli.bag.ls.1	0.941	0.936	0.949	0.948	0.910	0.905	0.907	0.908	0.871	0.873	0.873	0.871	0.839
loli.rf.ls	0.821	0.814	0.826	0.830	0.725	0.722	0.719	0.728	0.606	0.608	0.603	0.612	0.507
loli.rf.ls.1	0.948	0.941	0.954	0.953	0.924	0.920	0.923	0.923	0.887	0.890	0.890	0.888	0.856
loli.boost.ls	0.681	0.674	0.686	0.717	0.438	0.439	0.423	0.448	0.285	0.285	0.274	0.293	0.182
loli.tree.avg	0.945	0.942	0.949	0.947	0.944	0.940	0.943	0.943	0.944	0.943	0.942	0.945	0.949
loli.bag.avg	0.916	0.910	0.919	0.923	0.911	0.908	0.906	0.912	0.911	0.910	0.910	0.912	0.915
loli.bag.avg.1	0.917	0.910	0.920	0.923	0.911	0.908	0.907	0.913	0.911	0.911	0.911	0.912	0.915
loli.rf.avg	0.923	0.918	0.927	0.925	0.918	0.914	0.915	0.916	0.905	0.904	0.904	0.904	0.902
loli.rf.avg.1	0.979	0.977	0.982	0.981	0.980	0.979	0.979	0.980	0.979	0.979	0.978	0.979	0.979
loli.boost.avg	0.972	0.969	0.973	0.973	0.975	0.973	0.973	0.976	0.977	0.977	0.978	0.977	0.979
loli.tree.ls*	0.818	0.814	0.826	0.828	0.751	0.745	0.745	0.746	0.716	0.711	0.706	0.718	0.709
loli.bag.ls*	0.800	0.793	0.805	0.812	0.704	0.702	0.696	0.707	0.654	0.656	0.651	0.660	0.635
loli.bag.ls*.1	0.958	0.953	0.962	0.963	0.959	0.958	0.957	0.960	0.960	0.960	0.960	0.959	0.961
loli.rf.ls*	0.822	0.815	0.827	0.831	0.748	0.744	0.742	0.749	0.676	0.679	0.676	0.681	0.639
loli.rf.ls*.1	0.957	0.953	0.963	0.963	0.958	0.956	0.957	0.957	0.954	0.955	0.954	0.954	0.954
loli.boost.ls*	0.815	0.808	0.823	0.812	0.824	0.824	0.809	0.829	0.839	0.835	0.830	0.848	0.852
loli.tree.rid	0.839	0.834	0.848	0.848	0.753	0.754	0.748	0.754	0.679	0.679	0.667	0.680	0.611
loli.tree.rid.p	0.918	0.914	0.925	0.921	0.886	0.880	0.883	0.884	0.854	0.857	0.855	0.855	0.835
loli.bag.rid	0.922	0.916	0.926	0.923	0.915	0.912	0.913	0.913	0.901	0.901	0.901	0.900	0.897
loli.bag.rid.1	0.979	0.977	0.982	0.981	0.980	0.979	0.979	0.979	0.978	0.978	0.977	0.978	0.978
loli.rf.rid	-	-	-	-	-	-	-	-	0.704	0.706	0.695	0.711	0.623
loli.rf.rid.1	-	-	-	-	-	-	-	-	0.911	0.913	0.910	0.912	0.886
loli.boost.rid	0.745	0.738	0.749	0.764	0.577	0.576	0.569	0.582	0.461	0.460	0.451	0.466	0.373
ctree	0.990	0.982	0.997	0.994	0.994	0.991	0.991	0.993	0.991	0.992	0.992	0.994	0.993
bart	1.058	1.053	1.036	1.045	1.062	1.061	1.060	1.061	1.067	1.061	1.062	1.070	1.068
grf	0.966	0.960	0.972	0.970	0.967	0.963	0.966	0.965	0.963	0.964	0.965	0.963	0.964

Notes: See Table 4.4

Table 4.20: Relative multivariate IS forecasting performance for $T = 480$.

m	2				4				6				8
model	0	1	2	3	0	1	2	3	0	1	2	3	all
tree	0.967	0.956	0.980	0.973	0.951	0.941	0.948	0.948	0.923	0.922	0.919	0.924	0.907
tree.bag	0.566	0.562	0.570	0.575	0.504	0.502	0.498	0.506	0.464	0.463	0.461	0.468	0.440
tree.bag.1	0.971	0.967	0.979	0.975	0.968	0.965	0.969	0.968	0.960	0.963	0.964	0.962	0.963
tree.rf	0.595	0.591	0.599	0.606	0.575	0.572	0.569	0.576	0.514	0.514	0.512	0.518	0.506
tree.rf.1	0.974	0.970	0.981	0.977	0.975	0.973	0.977	0.974	0.967	0.969	0.970	0.968	0.971
tree.boost	0.877	0.871	0.884	0.882	0.836	0.831	0.834	0.838	0.795	0.798	0.799	0.801	0.776
loli.tree.ls	0.845	0.845	0.855	0.859	0.704	0.695	0.699	0.698	0.601	0.595	0.596	0.597	0.513
loli.bag.ls	0.782	0.777	0.789	0.796	0.653	0.646	0.645	0.653	0.536	0.534	0.533	0.538	0.444
loli.bag.ls.1	0.964	0.958	0.972	0.968	0.948	0.941	0.949	0.945	0.923	0.924	0.928	0.922	0.910
loli.rf.ls	0.806	0.800	0.812	0.817	0.710	0.703	0.704	0.709	0.585	0.584	0.585	0.588	0.486
loli.rf.ls.1	0.967	0.961	0.975	0.970	0.956	0.950	0.957	0.953	0.931	0.933	0.937	0.931	0.919
loli.boost.ls	0.808	0.805	0.820	0.827	0.628	0.622	0.616	0.632	0.490	0.487	0.484	0.494	0.393
loli.tree.avg	0.954	0.952	0.960	0.958	0.939	0.933	0.939	0.937	0.935	0.933	0.937	0.934	0.938
loli.bag.avg	0.905	0.898	0.910	0.913	0.896	0.892	0.895	0.898	0.896	0.894	0.896	0.898	0.902
loli.bag.avg.1	0.905	0.899	0.911	0.913	0.896	0.893	0.896	0.899	0.896	0.895	0.897	0.899	0.903
loli.rf.avg	0.913	0.908	0.920	0.915	0.905	0.900	0.907	0.904	0.889	0.889	0.893	0.890	0.888
loli.rf.avg.1	0.986	0.985	0.990	0.987	0.987	0.986	0.988	0.987	0.986	0.987	0.987	0.986	0.988
loli.boost.avg	0.983	0.982	0.984	0.983	0.983	0.983	0.985	0.985	0.986	0.986	0.986	0.987	0.987
loli.tree.ls*	0.846	0.847	0.856	0.860	0.715	0.703	0.703	0.705	0.636	0.625	0.622	0.635	0.590
loli.bag.ls*	0.782	0.777	0.789	0.796	0.658	0.651	0.650	0.658	0.564	0.562	0.559	0.566	0.509
loli.bag.ls*.1	0.973	0.970	0.980	0.976	0.976	0.973	0.978	0.975	0.975	0.975	0.977	0.976	0.978
loli.rf.ls*	0.806	0.800	0.812	0.817	0.713	0.706	0.707	0.712	0.603	0.602	0.603	0.605	0.533
loli.rf.ls*.1	0.973	0.969	0.980	0.976	0.974	0.972	0.976	0.973	0.971	0.972	0.974	0.972	0.974
loli.boost.ls*	0.910	0.903	0.916	0.907	0.888	0.898	0.897	0.896	0.890	0.901	0.896	0.905	0.916
loli.tree.rid	0.870	0.869	0.881	0.882	0.752	0.745	0.745	0.745	0.659	0.660	0.654	0.662	0.596
loli.tree.rid.p	0.941	0.935	0.951	0.948	0.917	0.912	0.914	0.913	0.896	0.898	0.896	0.885	0.873
loli.bag.rid	0.912	0.906	0.919	0.914	0.903	0.898	0.905	0.901	0.886	0.885	0.889	0.887	0.884
loli.bag.rid.1	0.986	0.985	0.990	0.987	0.986	0.986	0.988	0.986	0.985	0.986	0.986	0.986	0.987
loli.rf.rid	-	-	-	-	-	-	-	-	0.684	0.685	0.678	0.689	0.603
loli.rf.rid.1	-	-	-	-	-	-	-	-	0.943	0.945	0.947	0.943	0.934
loli.boost.rid	0.841	0.837	0.851	0.852	0.721	0.713	0.716	0.719	0.625	0.624	0.622	0.625	0.557
ctree	0.988	0.981	0.999	0.991	0.990	0.987	0.993	0.989	0.985	0.988	0.991	0.989	0.992
bart	1.048	1.056	1.025	1.044	1.040	1.044	1.037	1.038	1.055	1.041	1.068	1.055	1.051
grf	0.960	0.954	0.968	0.964	0.960	0.956	0.963	0.959	0.954	0.955	0.961	0.955	0.958

Notes: See Table 4.4

Table 4.21: Relative univariate IS forecasting performance for S&P 500 stock returns.

Predictor	logDP	logEP	DY	LTR	SVAR	BM	INFL	DFY
				$n = 60$				
tree	0.947	0.938	0.961	0.980	0.984	0.963	0.984	0.981
tree.bag	0.654	0.657	0.658	0.681	0.667	0.661	0.738	0.704
tree.bag.1	0.857	0.853	0.863	0.908	0.884	0.871	0.929	0.903
tree.boost	0.938	0.932	0.952	0.963	0.966	0.950	0.962	0.971
loli.tree.ls	0.895	0.873	0.912	0.924	0.918	0.916	0.954	0.933
loli.bag.ls	0.895	0.871	0.908	0.928	0.916	0.913	0.950	0.931
loli.bag.ls.1	0.895	0.871	0.908	0.928	0.916	0.913	0.950	0.931
loli.boost.ls	0.854	0.831	0.877	0.875	0.870	0.852	0.888	0.914
loli.tree.avg	0.968	0.950	0.974	0.980	0.985	0.970	0.988	0.982
loli.bag.avg	0.957	0.946	0.965	0.979	0.980	0.965	0.979	1.034
loli.bag.avg.1	0.957	0.946	0.965	0.979	0.980	0.965	0.979	1.034
loli.boost.avg	0.967	0.952	0.976	0.979	0.982	0.967	0.980	0.981
ctree	0.950	0.967	0.963	1.000	0.966	0.986	1.000	0.995
bart	1.171	1.167	1.128	1.045	1.129	1.128	1.038	1.131
grf	1.001	1.001	1.001	1.001	1.001	1.001	1.001	1.001
				$n = 120$				
tree	0.942	0.928	0.950	0.969	0.972	0.949	0.955	0.948
tree.bag	0.641	0.632	0.639	0.636	0.645	0.629	0.668	0.678
tree.bag.1	0.919	0.904	0.916	0.943	0.916	0.923	0.952	0.932
tree.boost	0.853	0.849	0.863	0.862	0.865	0.854	0.852	0.884
loli.tree.ls	0.904	0.862	0.908	0.911	0.889	0.905	0.933	0.922
loli.bag.ls	0.897	0.860	0.902	0.916	0.885	0.898	0.926	0.912
loli.bag.ls.1	0.937	0.895	0.936	0.957	0.931	0.936	0.959	0.939
loli.boost.ls	0.841	0.818	0.852	0.845	0.802	0.825	0.829	0.881
loli.tree.avg	0.965	0.938	0.965	0.970	0.989	0.953	0.977	0.958
loli.bag.avg	0.948	0.930	0.949	0.964	0.970	0.947	0.957	0.948
loli.bag.avg.1	0.948	0.930	0.949	0.964	0.970	0.948	0.957	0.949
loli.boost.avg	0.964	0.943	0.967	0.975	0.986	0.963	0.974	0.957
ctree	0.982	0.963	0.982	1.000	0.943	1.000	1.000	0.994
bart	1.120	1.132	1.094	1.030	1.092	1.075	1.037	1.136
grf	0.964	0.939	0.968	0.991	0.969	0.977	0.990	0.978
				$n = 240$				
tree	0.975	0.976	0.972	0.999	0.999	0.943	0.973	0.944
tree.bag	0.635	0.620	0.641	0.628	0.655	0.630	0.651	0.680
tree.bag.1	0.956	0.952	0.955	0.963	0.940	0.951	0.971	0.961
tree.boost	0.888	0.900	0.895	0.889	0.901	0.880	0.884	0.890
loli.tree.ls	0.908	0.925	0.918	0.914	0.923	0.903	0.927	0.932
loli.bag.ls	0.888	0.899	0.904	0.887	0.886	0.886	0.902	0.912
loli.bag.ls.1	0.970	0.943	0.965	0.978	0.954	0.950	0.975	0.969
loli.boost.ls	0.880	0.897	0.900	0.877	0.867	0.878	0.896	0.906
loli.tree.avg	0.973	0.986	0.972	0.991	0.996	0.949	0.979	0.943
loli.bag.avg	0.954	0.961	0.948	0.957	0.972	0.932	0.951	0.936
loli.bag.avg.1	0.954	0.961	0.949	0.958	0.972	0.932	0.951	0.936
loli.boost.avg	0.988	0.988	0.991	0.996	0.998	0.998	0.974	0.952
ctree	0.996	1.000	0.992	1.000	0.945	1.000	1.000	1.000
bart	1.027	1.029	1.031	1.018	1.010	1.032	1.017	1.032
grf	0.970	0.964	0.973	0.986	0.973	0.964	0.976	0.968

Notes: Average IS MSEs relative to average IS MSEs of historic mean for multivariate predictions of S&P 500 stock returns. Variable m indicates the number of regressor variables while *model* indicates the model set from which regressor variables are drawn. The best predictor per procedure are highlighted by bold relative MSEs.

Table 4.22: Relative multivariate IS forecasting performance for S&P 500 stock returns with $n = 60$.

m	2				4				6				8
model	0	1	2	3	0	1	2	3	0	1	2	3	all
tree	0.920	0.956	0.973	0.943	0.916	0.919	0.960	0.914	0.912	0.916	0.923	0.913	0.911
tree.bag	0.569	0.593	0.638	0.595	0.522	0.529	0.565	0.519	0.497	0.495	0.507	0.498	0.486
tree.bag.1	0.812	0.826	0.890	0.835	0.801	0.781	0.848	0.794	0.773	0.780	0.791	0.773	0.773
tree.rf	0.596	0.619	0.675	0.630	0.592	0.585	0.642	0.578	0.535	0.539	0.554	0.533	0.541
tree.rf.1	0.831	0.846	0.907	0.850	0.843	0.830	0.887	0.833	0.808	0.815	0.829	0.807	0.819
tree.boost	0.881	0.926	0.936	0.915	0.838	0.845	0.902	0.837	0.800	0.805	0.816	0.806	0.769
loli.tree.ls	0.777	0.832	0.884	0.829	0.705	0.683	0.754	0.662	0.567	0.571	0.611	0.535	0.481
loli.bag.ls	0.765	0.825	0.883	0.825	0.676	0.661	0.732	0.631	0.526	0.535	0.572	0.503	0.439
loli.bag.ls.1	0.765	0.825	0.883	0.825	0.676	0.661	0.732	0.631	0.526	0.535	0.572	0.503	0.439
loli.rf.ls	0.785	0.836	0.892	0.835	0.714	0.699	0.769	0.674	0.567	0.570	0.608	0.545	0.473
loli.rf.ls.1	0.785	0.836	0.892	0.835	0.714	0.699	0.769	0.674	0.567	0.570	0.608	0.545	0.473
loli.boost.ls	0.653	0.725	0.743	0.733	0.416	0.457	0.416	0.420	0.213	0.216	0.192	0.224	0.091
loli.tree.avg	0.946	0.974	0.981	0.967	0.954	0.955	0.977	0.958	0.965	0.966	0.970	0.969	0.971
loli.bag.avg	0.935	0.962	0.975	0.956	0.940	0.943	0.969	0.943	0.953	0.951	0.957	0.955	0.958
loli.bag.avg.1	0.935	0.962	0.975	0.956	0.940	0.943	0.969	0.943	0.953	0.951	0.957	0.955	0.958
loli.rf.avg	0.939	0.966	0.974	0.957	0.947	0.950	0.972	0.946	0.946	0.948	0.953	0.948	0.949
loli.rf.avg.1	0.939	0.966	0.974	0.957	0.947	0.950	0.972	0.946	0.946	0.948	0.953	0.948	0.949
loli.boost.avg	0.952	0.973	0.979	0.970	0.949	0.958	0.976	0.961	0.968	0.963	0.966	0.969	0.971
loli.tree.ls*	0.863	0.904	0.924	0.894	0.886	0.878	0.914	0.898	0.897	0.902	0.910	0.911	0.919
loli.bag.ls*	0.845	0.880	0.918	0.884	0.852	0.851	0.898	0.865	0.868	0.865	0.880	0.876	0.884
loli.bag.ls*.1	0.845	0.880	0.918	0.884	0.852	0.851	0.898	0.865	0.868	0.865	0.880	0.876	0.884
loli.rf.ls*	0.852	0.883	0.920	0.889	0.860	0.858	0.903	0.862	0.854	0.852	0.869	0.857	0.863
loli.rf.ls*.1	0.852	0.883	0.920	0.889	0.860	0.858	0.903	0.862	0.854	0.852	0.869	0.857	0.863
loli.boost.ls*	0.812	0.844	0.871	0.847	0.815	0.845	0.908	0.859	0.862	0.871	0.886	0.895	0.904
loli.tree.rid	0.798	0.856	0.915	0.853	0.764	0.724	0.839	0.707	0.644	0.657	0.705	0.611	0.593
loli.tree.rid.p	-	-	-	-	-	-	-	-	-	-	-	-	-
loli.bag.rid	0.789	0.849	0.907	0.848	0.728	0.701	0.788	0.680	0.596	0.610	0.638	0.583	0.530
loli.bag.rid.1	0.789	0.849	0.907	0.848	0.728	0.701	0.788	0.680	0.596	0.610	0.638	0.583	0.530
loli.rf.rid	-	-	-	-	-	-	-	-	0.680	0.685	0.719	0.668	0.606
loli.rf.rid.1	-	-	-	-	-	-	-	-	0.680	0.685	0.719	0.668	0.606
loli.boost.rid	0.713	0.786	0.826	0.785	0.594	0.589	0.661	0.571	0.460	0.476	0.488	0.456	0.382
ctree	0.956	0.966	1.000	0.960	0.963	0.960	0.996	0.960	0.963	0.970	0.985	0.963	0.971
bart	1.180	1.151	1.043	1.145	1.148	1.171	1.109	1.147	1.146	1.147	1.142	1.147	1.134
grf	1.001	1.001	1.001	1.001	1.001	1.001	1.001	1.001	1.001	1.001	1.001	1.001	1.001

Notes: Average IS MSEs relative to average IS MSEs of historic mean for multivariate predictions of S&P 500 stock returns. Variable m indicates the number of regressor variables while *model* indicates the model set from which regressor variables are drawn. The best predictor per procedure are highlighted by bold relative MSEs.

Table 4.23: Relative multivariate IS forecasting performance for S&P 500 stock returns with $n = 120$.

m	2				4				6				8
model	0	1	2	3	0	1	2	3	0	1	2	3	all
tree	0.907	0.939	0.945	0.935	0.896	0.892	0.918	0.899	0.889	0.883	0.889	0.882	0.877
tree.bag	0.541	0.560	0.584	0.589	0.497	0.496	0.522	0.489	0.462	0.461	0.462	0.461	0.446
tree.bag.1	0.889	0.880	0.932	0.906	0.883	0.855	0.890	0.880	0.851	0.854	0.857	0.851	0.851
tree.rf	0.566	0.584	0.614	0.618	0.561	0.551	0.580	0.544	0.498	0.501	0.503	0.495	0.496
tree.rf.1	0.898	0.894	0.941	0.914	0.905	0.891	0.921	0.900	0.877	0.879	0.884	0.875	0.883
tree.boost	0.739	0.792	0.791	0.810	0.676	0.676	0.673	0.668	0.610	0.612	0.583	0.613	0.538
loli.tree.ls	0.769	0.818	0.865	0.824	0.699	0.671	0.681	0.669	0.557	0.563	0.601	0.544	0.482
loli.bag.ls	0.744	0.798	0.848	0.813	0.661	0.641	0.663	0.612	0.513	0.519	0.539	0.490	0.425
loli.bag.ls.1	0.831	0.871	0.930	0.885	0.792	0.764	0.809	0.751	0.681	0.676	0.714	0.658	0.618
loli.rf.ls	0.774	0.818	0.864	0.831	0.713	0.699	0.722	0.671	0.568	0.562	0.584	0.542	0.464
loli.rf.ls.1	0.856	0.877	0.933	0.896	0.824	0.798	0.838	0.791	0.716	0.705	0.743	0.689	0.648
loli.boost.ls	0.529	0.556	0.577	0.662	0.264	0.311	0.214	0.260	0.106	0.099	0.084	0.112	0.038
loli.tree.avg	0.939	0.961	0.978	0.961	0.932	0.946	0.973	0.939	0.972	0.965	0.968	0.953	0.973
loli.bag.avg	0.919	0.945	0.957	0.943	0.921	0.925	0.945	0.923	0.937	0.936	0.933	0.933	0.938
loli.bag.avg.1	0.919	0.945	0.958	0.943	0.921	0.926	0.945	0.923	0.937	0.936	0.934	0.934	0.938
loli.rf.avg	0.920	0.950	0.957	0.944	0.929	0.929	0.951	0.925	0.929	0.925	0.928	0.924	0.924
loli.rf.avg.1	0.947	0.973	0.981	0.966	0.958	0.958	0.978	0.957	0.962	0.959	0.963	0.960	0.961
loli.boost.avg	0.941	0.968	0.966	0.968	0.938	0.964	0.973	0.946	0.980	0.973	0.965	0.964	0.976
loli.tree.ls*	0.788	0.860	0.872	0.840	0.787	0.783	0.826	0.786	0.806	0.797	0.833	0.772	0.827
loli.bag.ls*	0.761	0.820	0.856	0.828	0.758	0.737	0.789	0.728	0.741	0.743	0.754	0.729	0.751
loli.bag.ls*.1	0.893	0.916	0.947	0.931	0.901	0.900	0.934	0.905	0.914	0.915	0.914	0.915	0.924
loli.rf.ls*	0.789	0.829	0.871	0.842	0.786	0.770	0.811	0.762	0.742	0.738	0.757	0.727	0.735
loli.rf.ls*.1	0.894	0.913	0.947	0.933	0.907	0.899	0.928	0.905	0.904	0.901	0.906	0.901	0.909
loli.boost.ls*	0.683	0.797	0.683	0.761	0.686	0.670	0.742	0.684	0.722	0.771	0.642	0.761	0.807
loli.tree.rid	0.790	0.841	0.893	0.844	0.751	0.712	0.748	0.701	0.637	0.641	0.676	0.617	0.583
loli.tree.rid.p	0.813	0.857	0.914	0.868	0.791	0.759	0.808	0.750	0.701	0.698	0.728	0.682	0.662
loli.bag.rid	0.768	0.824	0.876	0.840	0.715	0.685	0.717	0.664	0.586	0.597	0.606	0.570	0.516
loli.bag.rid.1	0.844	0.883	0.942	0.902	0.823	0.789	0.839	0.784	0.723	0.723	0.752	0.707	0.673
loli.rf.rid	-	-	-	-	-	-	-	-	0.671	0.673	0.689	0.656	0.592
loli.rf.rid.1	-	-	-	-	-	-	-	-	0.778	0.775	0.801	0.763	0.725
loli.boost.rid	0.601	0.656	0.712	0.722	0.448	0.440	0.398	0.428	0.293	0.285	0.254	0.295	0.184
ctree	0.966	0.906	1.000	0.985	0.970	0.899	0.956	0.970	0.911	0.910	0.935	0.911	0.936
bart	1.150	1.110	1.034	1.099	1.124	1.137	1.096	1.125	1.109	1.118	1.114	1.117	1.103
grf	0.946	0.961	0.990	0.965	0.957	0.947	0.979	0.955	0.956	0.955	0.962	0.951	0.960

Notes: See Table 4.22

Table 4.24: Relative multivariate IS forecasting performance for S&P 500 stock returns with $n = 240$.

m	2				4				6				8
model	0	1	2	3	0	1	2	3	0	1	2	3	all
tree	0.958	0.970	0.963	0.970	0.939	0.940	0.931	0.897	0.891	0.901	0.893	0.878	0.873
tree.bag	0.544	0.564	0.565	0.576	0.505	0.506	0.521	0.487	0.472	0.468	0.474	0.462	0.452
tree.bag.1	0.948	0.922	0.959	0.951	0.945	0.919	0.932	0.942	0.917	0.919	0.921	0.917	0.917
tree.rf	0.565	0.579	0.594	0.605	0.561	0.548	0.566	0.537	0.499	0.500	0.504	0.492	0.492
tree.rf.1	0.951	0.932	0.963	0.954	0.952	0.938	0.950	0.949	0.931	0.933	0.935	0.930	0.935
tree.boost	0.823	0.835	0.840	0.851	0.780	0.775	0.750	0.751	0.725	0.720	0.708	0.718	0.675
loli.tree.ls	0.741	0.783	0.818	0.801	0.695	0.669	0.629	0.646	0.544	0.540	0.574	0.488	0.475
loli.bag.ls	0.735	0.772	0.802	0.789	0.642	0.630	0.619	0.581	0.501	0.508	0.532	0.482	0.423
loli.bag.ls.1	0.931	0.914	0.962	0.932	0.901	0.880	0.898	0.869	0.828	0.820	0.847	0.812	0.783
loli.rf.ls	0.771	0.808	0.827	0.809	0.698	0.695	0.691	0.650	0.550	0.547	0.571	0.531	0.454
loli.rf.ls.1	0.941	0.920	0.964	0.942	0.918	0.897	0.916	0.888	0.844	0.845	0.866	0.830	0.806
loli.boost.ls	0.640	0.618	0.675	0.709	0.427	0.426	0.356	0.391	0.250	0.246	0.242	0.244	0.159
loli.tree.avg	0.978	0.940	0.968	0.962	0.957	0.960	0.917	0.966	0.974	0.972	0.966	0.955	0.972
loli.bag.avg	0.934	0.935	0.943	0.946	0.932	0.931	0.918	0.930	0.940	0.931	0.924	0.929	0.931
loli.bag.avg.1	0.934	0.935	0.943	0.946	0.932	0.931	0.919	0.931	0.941	0.932	0.925	0.929	0.931
loli.rf.avg	0.936	0.944	0.945	0.944	0.936	0.934	0.935	0.923	0.928	0.927	0.923	0.920	0.920
loli.rf.avg.1	0.981	0.990	0.989	0.985	0.984	0.984	0.988	0.983	0.987	0.985	0.984	0.985	0.985
loli.boost.avg	0.983	0.991	0.977	0.991	0.981	0.985	0.961	0.986	0.993	0.995	0.977	0.991	0.995
loli.tree.ls*	0.741	0.788	0.818	0.801	0.710	0.726	0.702	0.689	0.665	0.656	0.696	0.655	0.674
loli.bag.ls*	0.735	0.777	0.802	0.791	0.676	0.667	0.664	0.620	0.611	0.621	0.633	0.593	0.607
loli.bag.ls*.1	0.950	0.951	0.971	0.966	0.958	0.951	0.967	0.956	0.958	0.959	0.949	0.956	0.960
loli.rf.ls*	0.772	0.809	0.828	0.810	0.724	0.720	0.722	0.679	0.634	0.634	0.653	0.616	0.601
loli.rf.ls*.1	0.950	0.946	0.970	0.967	0.955	0.945	0.960	0.949	0.948	0.950	0.947	0.948	0.952
loli.boost.ls*	0.718	0.949	0.800	0.829	0.824	0.842	0.886	0.786	0.824	0.822	0.743	0.822	0.897
loli.tree.rid	0.773	0.809	0.855	0.821	0.736	0.716	0.729	0.676	0.621	0.637	0.648	0.583	0.569
loli.tree.rid.p	0.867	0.885	0.948	0.892	0.863	0.850	0.875	0.837	0.804	0.826	0.835	0.787	0.804
loli.bag.rid	0.762	0.802	0.832	0.819	0.696	0.677	0.675	0.634	0.566	0.584	0.594	0.557	0.508
loli.bag.rid.1	0.939	0.921	0.968	0.942	0.920	0.895	0.914	0.890	0.849	0.848	0.874	0.842	0.820
loli.rf.rid	-	-	-	-	-	-	-	-	0.656	0.662	0.674	0.640	0.578
loli.rf.rid.1	-	-	-	-	-	-	-	-	0.877	0.880	0.895	0.868	0.847
loli.boost.rid	0.695	0.695	0.747	0.752	0.570	0.552	0.507	0.519	0.432	0.427	0.424	0.403	0.351
ctree	1.000	0.913	1.000	0.998	1.000	0.985	0.992	1.000	0.995	0.995	0.997	0.995	0.998
bart	1.035	1.029	1.017	1.027	1.035	1.032	1.020	1.023	1.021	1.030	1.020	1.020	1.018
grf	0.957	0.960	0.983	0.971	0.966	0.957	0.979	0.957	0.960	0.958	0.965	0.952	0.961

Notes: See Table 4.22

Table 4.25: Relative average IS stock return forecast performance for 212 firms

Window Length	5	5	10	10	20	20
Market		✓		✓		✓
tree	0.922	0.898	0.905	0.861	0.911	0.851
tree.bag	1.245	1.133	1.227	1.109	1.207	1.096
tree.bag.1	0.805	0.803	0.867	0.867	0.912	0.912
tree.rf	1.176	1.092	1.162	1.074	1.148	1.067
tree.rf.1	0.842	0.840	0.894	0.894	0.931	0.931
tree.boost	0.775	0.468	0.761	0.504	0.754	0.539
loli.tree.ls	0.737	0.299	0.713	0.303	0.700	0.307
loli.bag.ls	0.723	0.270	0.695	0.270	0.674	0.264
loli.bag.ls.1	0.725	0.723	0.798	0.798	0.866	0.866
loli.rf.ls	0.738	0.303	0.715	0.294	0.697	0.282
loli.rf.ls.1	0.763	0.761	0.834	0.834	0.893	0.893
loli.boost.ls	0.494	0.000	0.293	0.000	0.318	0.000
loli.tree.avg	0.964	0.975	0.957	0.965	0.951	0.962
loli.bag.avg	0.946	0.958	0.932	0.944	0.921	0.935
loli.bag.avg.1	0.949	0.947	0.965	0.965	0.978	0.978
loli.rf.avg	0.945	0.950	0.931	0.928	0.920	0.916
loli.rf.avg.1	0.951	0.948	0.966	0.966	0.978	0.978
loli.boost.avg	0.969	0.975	0.962	0.969	0.978	0.982
loli.tree.ls*	0.897	0.925	0.797	0.820	0.727	0.662
loli.bag.ls*	0.871	0.879	0.761	0.746	0.689	0.568
loli.bag.ls*.1	0.871	0.869	0.913	0.913	0.946	0.946
loli.rf.ls*	0.867	0.857	0.767	0.711	0.708	0.540
loli.rf.ls*.1	0.874	0.872	0.907	0.907	0.941	0.941
loli.boost.ls*	0.873	0.912	0.691	0.804	0.805	0.869
loli.tree.rid	0.783	0.474	0.756	0.437	0.742	0.422
ctree	0.953	0.961	0.957	0.954	0.967	0.958
bart	0.675	0.490	0.692	0.493	0.729	0.547
grf	1.000	1.000	0.948	0.956	0.943	0.944

Notes: Average IS MSE relative to average IS MSE firm level data. Predictions applying market data employ 11 regressor variables, predictions without market data employ 3 firm specific regressors.

Table 4.26: Relative multivariate OOS forecasting performance for S&P 500 stock returns after adjusting for outliers with $n = 60$.

m	2				4				6				8
model	0	1	2	3	0	1	2	3	0	1	2	3	all
tree	**0.994**	1.007	1.017	1.002	1.003	1.004	1.026	1.000	1.006	1.012	1.014	1.005	1.015
tree.bag	1.020	1.037	1.030	1.042	1.032	**1.014**	1.031	1.034	1.028	1.023	1.027	1.027	1.028
tree.bag.1	1.006	1.024	1.017	1.017	1.010	**1.004**	1.010	1.016	1.010	1.005	1.007	1.008	1.010
tree.rf	1.009	1.026	1.027	1.030	1.010	1.007	1.023	1.015	1.013	1.010	1.014	**1.006**	1.012
tree.rf.1	**0.990**	1.016	1.015	1.014	0.996	1.001	1.006	0.998	1.000	1.003	1.000	0.999	1.002
tree.boost	**0.989**	1.007	1.023	1.007	0.994	0.993	1.042	1.001	1.004	1.010	1.040	1.013	1.031
loli.tree.ls	**1.032**	1.071	1.050	1.047	1.117	1.057	1.119	1.143	1.153	1.146	1.192	1.143	1.133
loli.bag.ls	**0.995**	1.045	1.038	1.012	1.055	1.013	1.090	1.108	1.109	1.064	1.129	1.117	1.112
loli.bag.ls.1	**0.995**	1.045	1.038	1.012	1.055	1.013	1.090	1.108	1.109	1.064	1.129	1.117	1.112
loli.rf.ls	**0.983**	1.035	1.034	1.008	1.020	1.005	1.053	1.090	1.080	1.046	1.093	1.091	1.096
loli.rf.ls.1	**0.983**	1.035	1.034	1.008	1.020	1.005	1.053	1.090	1.080	1.046	1.093	1.091	1.096
loli.boost.ls	1.033	1.053	1.081	**1.027**	1.108	1.066	1.121	1.109	1.205	1.110	1.221	1.101	1.226
loli.tree.avg	0.995	1.014	1.011	1.005	1.001	1.002	1.012	0.999	**0.990**	0.997	1.003	0.991	0.996
loli.bag.avg	0.988	1.009	1.010	1.000	0.988	0.995	1.006	**0.987**	0.995	0.999	0.999	0.994	0.996
loli.bag.avg.1	0.988	1.009	1.010	0.999	0.988	0.994	1.006	**0.988**	0.995	0.999	0.999	0.994	0.996
loli.rf.avg	**0.987**	1.002	1.008	0.996	0.987	0.994	1.010	0.989	0.999	0.998	1.002	0.997	1.000
loli.rf.avg.1	**0.987**	1.002	1.008	0.996	0.987	0.994	1.010	0.989	0.999	0.998	1.002	0.997	1.000
loli.boost.avg	0.998	1.018	1.011	1.005	0.990	0.997	1.011	0.995	**0.983**	1.001	1.003	0.986	0.997
loli.tree.ls*	1.026	1.044	1.030	1.052	1.027	1.024	1.021	1.021	**0.983**	0.997	1.053	1.022	1.073
loli.bag.ls*	0.998	1.015	1.019	1.015	**0.981**	0.996	1.002	1.013	1.000	0.986	1.010	1.008	0.998
loli.bag.ls.1*	0.998	1.015	1.019	1.015	**0.981**	0.996	1.002	1.013	1.000	0.986	1.010	1.008	0.998
loli.rf.ls*	0.995	1.005	1.017	1.012	0.996	0.996	0.998	1.005	0.999	**0.985**	1.001	0.998	0.999
loli.rf.ls.1*	0.995	1.005	1.017	1.012	0.996	0.996	0.998	1.005	0.999	**0.985**	1.001	0.998	0.999
loli.boost.ls*	1.023	1.028	1.043	1.032	1.018	1.012	1.017	1.006	**0.987**	1.001	1.049	1.005	1.031
loli.tree.rid	1.012	1.025	1.029	1.045	1.049	**1.007**	1.071	1.075	1.135	1.101	1.145	1.086	1.120
loli.tree.rid.p	-	-	-	-	-	-	-	-	-	-	-	-	-
loli.bag.rid	0.982	1.015	1.026	1.010	1.029	**0.981**	1.058	1.060	1.062	1.027	1.094	1.044	1.065
loli.bag.rid.1	0.982	1.015	1.026	1.010	1.029	**0.981**	1.058	1.060	1.062	1.027	1.094	1.044	1.065
loli.rf.rid	-	-	-	-	-	-	-	-	1.035	**1.013**	1.032	1.028	1.049
loli.rf.rid.1	-	-	-	-	-	-	-	-	1.035	**1.013**	1.032	1.028	1.049
loli.boost.rid	1.004	1.033	1.046	1.028	1.049	**0.999**	1.117	1.072	1.099	1.070	1.134	1.089	1.147
ctree	1.010	1.013	1.001	1.013	1.003	1.012	1.011	**0.994**	1.001	1.007	1.007	1.001	1.000
bart	**0.981**	1.010	1.029	1.013	0.994	0.983	1.027	0.994	0.998	0.991	1.021	0.994	1.009
grf	1.000	1.001	1.000	1.003	1.000	1.001	1.001	0.999	1.001	**0.999**	1.000	1.002	0.999

Notes: Average OOS MSE of combination relative to average OOS MSE firm level data after adjusting for outliers. Thresholds to identify outliers are computed as 5-year-average plus (minus) 2 times the difference of 5-year-average and 5-year-maximum (5-year-minimum).

Table 4.27: Relative multivariate OOS forecasting performance for S&P 500 stock returns after adjusting for outliers with $n = 120$.

m	2				4				6				8
model	0	1	2	3	0	1	2	3	0	1	2	3	all
tree	**1.015**	1.039	1.018	1.047	1.032	1.024	1.018	1.025	1.025	1.020	1.032	1.022	1.021
tree.bag	1.029	1.040	1.018	1.058	1.032	1.019	1.012	1.043	1.027	**1.010**	1.019	1.015	1.019
tree.bag.1	1.007	1.008	0.999	1.018	1.008	1.001	0.998	1.015	1.004	0.999	**0.997**	1.000	1.003
tree.rf	1.020	1.032	1.013	1.052	1.023	1.019	1.010	1.037	1.028	1.010	**1.009**	1.013	1.016
tree.rf.1	0.996	1.004	1.002	1.015	0.998	1.001	**0.991**	1.008	1.004	0.998	0.994	1.006	1.003
tree.boost	**1.015**	1.067	1.026	1.052	1.041	1.043	1.052	1.037	1.060	1.056	1.080	1.050	1.083
loli.tree.ls	1.042	1.154	1.055	1.085	1.131	1.039	1.043	1.101	1.078	**0.983**	1.132	1.012	1.062
loli.bag.ls	1.016	1.062	1.020	1.043	1.061	**0.991**	1.037	1.073	1.031	0.998	1.107	1.000	1.023
loli.bag.ls.1	1.024	1.046	**1.012**	1.035	1.035	1.034	1.033	1.050	1.017	1.012	1.069	1.035	1.038
loli.rf.ls	1.011	1.044	1.013	1.042	1.044	1.011	1.037	1.064	1.036	1.042	1.085	**1.005**	1.026
loli.rf.ls.1	1.013	1.025	**1.009**	1.034	1.029	1.031	1.022	1.047	1.052	1.014	1.049	1.023	1.029
loli.boost.ls	1.041	1.136	1.045	1.090	1.143	1.065	1.145	1.114	1.135	1.099	1.186	**1.028**	1.106
loli.tree.avg	1.013	1.024	**1.011**	1.018	1.037	1.025	1.031	1.012	1.028	1.029	1.048	1.033	1.014
loli.bag.avg	1.008	1.015	**1.004**	1.023	1.010	1.012	1.008	1.013	1.014	1.011	1.017	1.019	1.014
loli.bag.avg.1	1.007	1.015	**1.004**	1.023	1.009	1.011	1.007	1.013	1.014	1.011	1.017	1.019	1.013
loli.rf.avg	**1.000**	1.018	1.005	1.019	1.004	1.003	1.010	1.009	1.012	1.005	1.012	1.015	1.009
loli.rf.avg.1	**0.995**	1.008	1.002	1.001	1.000	0.999	1.006	1.002	1.005	1.001	1.006	1.009	1.003
loli.boost.avg	1.003	1.027	1.008	1.023	1.023	1.010	1.023	**0.999**	1.011	1.016	1.030	1.016	1.021
loli.tree.ls*	1.032	1.143	1.034	1.081	1.099	1.044	1.008	1.113	1.146	**0.989**	1.133	1.012	1.037
loli.bag.ls*	1.013	1.056	1.013	1.038	1.033	1.013	**1.012**	1.050	1.037	1.019	1.043	1.025	1.027
loli.bag.ls.1*	1.018	1.029	1.002	1.034	1.008	1.016	**0.988**	1.018	1.008	1.004	1.000	1.011	1.008
loli.rf.ls*	**1.006**	1.036	1.006	1.041	1.030	1.027	1.012	1.045	1.045	1.020	1.021	1.025	1.026
loli.rf.ls.1*	1.007	1.018	1.003	1.033	1.010	1.010	**0.994**	1.023	1.015	1.002	0.996	1.008	0.999
loli.boost.ls*	1.033	1.117	**1.022**	1.065	1.097	1.034	1.045	1.107	1.100	1.041	1.074	1.064	1.073
loli.tree.rid	1.037	1.098	1.056	1.051	1.076	**1.035**	1.043	1.085	1.123	1.037	1.182	1.089	1.052
loli.tree.rid.p	1.021	1.088	1.048	1.051	1.064	1.029	1.034	1.068	1.118	**1.016**	1.157	1.021	1.037
loli.bag.rid	1.013	1.044	1.017	1.041	1.045	1.030	1.030	1.050	1.031	1.034	1.078	**0.998**	1.028
loli.bag.rid.1	1.014	1.028	**1.010**	1.028	1.026	1.018	1.026	1.041	1.015	1.013	1.058	1.026	1.042
loli.rf.rid	-	-	-	-	-	-	-	-	1.058	**1.027**	1.044	1.039	1.045
loli.rf.rid.1	-	-	-	-	-	-	-	-	1.033	**1.008**	1.025	1.020	1.020
loli.boost.rid	**1.022**	1.120	1.044	1.063	1.103	1.031	1.130	1.117	1.105	1.075	1.186	1.068	1.127
ctree	0.991	1.009	1.000	1.001	**0.989**	1.001	1.036	0.993	0.992	0.999	1.002	1.004	1.004
bart	1.014	1.016	1.003	1.033	1.017	1.009	**0.997**	1.034	1.019	1.002	1.013	1.018	1.017
grf	1.003	1.006	1.006	1.006	1.001	1.005	0.999	1.001	1.000	1.003	1.001	**0.996**	1.000

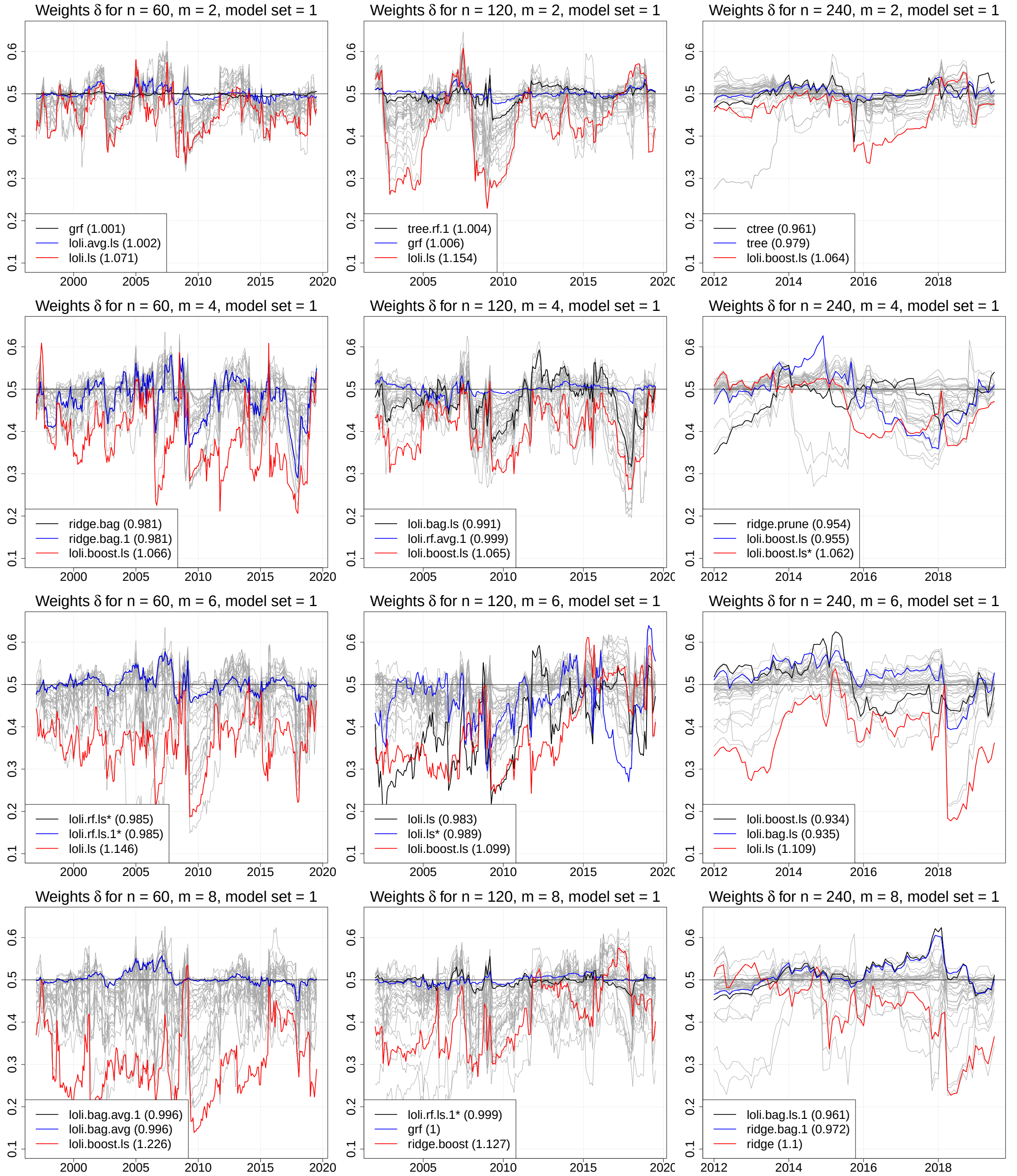
Notes: See Table 4.26

Table 4.28: Relative multivariate OOS forecasting performance for S&P 500 stock returns after adjusting for outliers with $n = 240$.

m	2				4				6				8
model	0	1	2	3	0	1	2	3	0	1	2	3	all
tree	**0.952**	0.979	1.044	0.986	1.012	1.003	1.082	1.007	1.026	1.069	1.027	1.057	1.003
tree.bag	0.989	1.014	1.006	1.041	0.985	**0.975**	1.025	1.019	1.000	0.989	1.037	1.021	1.013
tree.bag.1	0.988	0.991	0.994	0.992	**0.985**	0.990	0.992	0.994	0.990	0.989	0.994	0.993	0.992
tree.rf	0.987	1.015	1.017	1.036	**0.983**	0.989	1.019	1.018	0.991	0.985	1.018	1.005	0.995
tree.rf.1	**0.988**	0.995	0.997	0.992	0.990	0.991	0.999	0.993	0.990	0.991	0.996	0.995	0.995
tree.boost	0.980	1.027	1.039	1.024	**0.979**	1.025	1.118	1.027	1.051	1.030	1.095	1.052	1.080
loli.tree.ls	**0.987**	1.023	1.056	1.073	1.038	0.995	1.031	1.121	1.085	1.109	1.048	1.069	1.056
loli.bag.ls	0.975	1.015	1.020	1.038	0.989	0.960	1.018	1.038	0.972	**0.935**	1.034	0.986	0.981
loli.bag.ls.1	0.979	0.991	0.989	0.996	0.981	0.974	1.006	1.010	0.972	**0.958**	0.998	0.982	0.961
loli.rf.ls	0.973	1.008	1.014	1.033	0.981	0.971	1.021	1.018	0.982	**0.946**	1.027	0.980	1.007
loli.rf.ls.1	0.979	0.988	0.993	1.003	0.983	0.976	0.994	1.008	0.973	**0.967**	0.993	0.980	0.979
loli.boost.ls	1.000	1.064	1.013	1.078	0.956	0.955	1.030	1.052	1.009	**0.934**	1.118	1.004	0.985
loli.tree.avg	1.011	1.015	1.051	1.013	1.028	1.012	1.008	0.983	0.986	**0.982**	0.993	1.006	0.990
loli.bag.avg	**0.986**	1.014	1.028	0.997	0.987	0.987	1.034	0.992	0.988	0.993	1.011	0.994	0.997
loli.bag.avg.1	**0.986**	1.013	1.028	0.997	0.987	0.987	1.033	0.992	0.988	0.993	1.011	0.994	0.997
loli.rf.avg	**0.975**	1.005	1.027	0.985	0.978	0.983	1.024	0.988	0.997	1.003	1.013	0.998	1.003
loli.rf.avg.1	**0.990**	0.999	1.009	0.993	0.994	0.993	1.006	0.992	0.998	0.997	1.005	0.997	1.001
loli.boost.avg	0.994	1.002	1.020	0.992	**0.981**	1.011	1.041	0.992	0.993	0.988	0.997	0.994	0.997
loli.tree.ls*	**0.987**	1.018	1.056	1.073	1.045	0.999	1.035	1.083	1.063	1.038	1.008	1.045	1.028
loli.bag.ls*	0.974	1.017	1.020	1.037	0.994	**0.967**	1.026	1.028	0.994	0.972	1.024	1.003	0.999
loli.bag.ls.1*	**0.979**	0.997	0.998	0.984	0.982	0.984	1.006	0.985	0.986	0.984	0.999	0.989	0.990
loli.rf.ls*	**0.972**	1.007	1.014	1.032	0.984	0.973	1.025	1.017	0.998	0.979	1.011	0.997	1.015
loli.rf.ls.1*	**0.978**	0.998	1.000	0.986	0.983	0.985	1.002	0.988	0.990	0.991	0.999	0.995	0.996
loli.boost.ls*	1.004	0.994	1.004	1.050	0.993	1.062	1.028	1.012	1.015	**0.972**	1.013	1.032	1.018
loli.tree.rid	0.989	1.021	1.092	1.072	1.036	**0.975**	1.045	1.150	1.087	1.082	1.110	1.036	1.100
loli.tree.rid.p	0.989	1.019	0.997	1.060	0.970	**0.954**	1.037	1.077	1.002	1.037	0.987	1.006	1.053
loli.bag.rid	0.968	1.013	1.023	1.023	0.984	0.973	1.023	1.032	0.992	**0.951**	1.023	0.986	0.985
loli.bag.rid.1	0.986	0.993	0.998	0.993	0.980	0.982	1.003	1.007	0.975	**0.965**	1.001	0.985	0.972
loli.rf.rid	-	-	-	-	-	-	-	-	0.992	**0.969**	1.006	0.994	0.996
loli.rf.rid.1	-	-	-	-	-	-	-	-	0.986	**0.977**	0.991	0.986	0.978
loli.boost.rid	0.991	1.031	0.998	1.043	0.984	0.975	1.048	1.033	0.989	**0.972**	1.078	1.014	1.048
ctree	1.000	**0.961**	1.000	0.993	1.000	0.984	1.000	1.000	0.981	0.981	1.005	0.981	0.997
bart	**0.968**	0.995	0.996	0.991	0.976	0.974	1.016	1.007	0.988	0.983	1.008	0.999	0.993
grf	**0.978**	0.998	1.006	0.989	0.983	0.987	1.002	0.982	0.992	1.001	0.998	0.993	0.996

Notes: See Table 4.26

Figure 4.6: Weights δ for the combination method over time.



Notes: Weights δ for the combination method over time for market level data and model set $= 1$; top to bottom: number of regressor variables $=$ 2,4,6 and 8; left to right: window length $=$ 60, 120 and 240 months. Two best and least performing procedures (w.r.t. OOS MSE) are highlighted in black, blue and red, respectively; corresponding relative MSE values are depicted in legend.

Bibliography

Amemyia, T. (1985). *Advanced Econometrics.* Harvard University Press.

Amihud, Y. and C. M. Hurvich (2004). Predictive regressions: A reduced-bias estimation method. *Journal of Financial and Quantitative Analysis 39*(4), 813–841.

Andersen, P. K. and R. D. Gill (1982). Cox's regression model for counting processes: A large sample study. *The Annals of Statistics 10*(4), 1100–1120.

Athey, S., J. Tibshirani, S. Wager, et al. (2019). Generalized random forests. *The Annals of Statistics 47*(2), 1148–1178.

Basak, S., S. Kar, S. Saha, L. Khaidem, and S. R. Dey (2019). Predicting the direction of stock market prices using tree-based classifiers. *The North American Journal of Economics and Finance 47*, 552–567.

Boswijk, H. P., G. Cavaliere, I. Georgiev, and A. Rahbek (2019). Bootstrapping non-stationary stochastic volatility. Tinbergen Institute Discussion Paper 2019-083/III.

Breiman, L., J. Friedman, C. J. Stone, and R. A. Olshen (1984). *Classification and regression trees.* CRC press.

Breitung, J. and M. Demetrescu (2015). Instrumental variable and variable addition based inference in predictive regressions. *Journal of Econometrics 187*(1), 358–375.

Campbell, J. Y. (2008). Viewpoint: Estimating the equity premium. *Canadian Journal of Economics/Revue canadienne d'économique 41*(1), 1–21.

Campbell, J. Y. and M. Yogo (2006). Efficient tests of stock return predictability. *Journal of Financial Economics 81*(1), 27–60.

Cavaliere, G. and I. Georgiev (2020). Inference under random limit bootstrap measures. *Econometrica 88*(6), 2547–2574.

Cenesizoglu, T. and A. Timmermann (2008). Is the distribution of stock returns predictable? SSRN working paper 1107185.

Chipman, H. A., E. I. George, R. E. McCulloch, et al. (2010). Bart: Bayesian additive regression trees. *The Annals of Applied Statistics 4*(1), 266–298.

Davidson, J. (1994). *Stochastic Limit Theory.* Oxford University Press.

Demetrescu, M., I. Georgiev, P. Rodrigues, and A. M. R. Taylor (2021). Extensions to IVX methods of inference for return predictability. *Working paper*.

Demetrescu, M., I. Georgiev, P. Rodrigues, and R. Taylor (2020). Testing for episodic predictability in stock returns. *Journal of Econometrics forthcoming.*

Demetrescu, M. and B. Hillmann (2020). Nonlinear predictability of stock returns? parametric versus nonparametric inference in predictive regressions. *Journal of Business & Economic Statistics*, 1–16.

Demetrescu, M. and C. Roling (2021). Testing the predictive ability of possibly persistent variables under asymmetric loss. Working paper, University of Kiel.

Elliott, G., I. Komunjer, and A. Timmermann (2005). Estimation and testing of forecast rationality under flexible loss. *Review of Economic Studies 72*(4), 1107–1125.

Elliott, G., U. K. Müller, and M. W. Watson (2015). Nearly optimal tests when a nuisance parameter is present under the null hypothesis. *Econometrica 83*(2), 771–811.

Elliott, G. and J. H. Stock (1994). Inference in Time Series Regression when the Order of Integration of a Regressor Is Unknown. *Econometric Theory 10*(3-4), 672–700.

Fama, E. F. (1995). Random walks in stock market prices. *Financial analysts journal 51*(1), 75–80.

Friedberg, R., J. Tibshirani, S. Athey, and S. Wager (2020). Local linear forests. *Journal of Computational and Graphical Statistics* (just-accepted), 1–40.

Friedman, J., T. Hastie, R. Tibshirani, et al. (2001). *The elements of statistical learning*, Volume 1. Springer series in statistics New York.

Friedman, J. H. (2001). Greedy function approximation: a gradient boosting machine. *Annals of statistics*, 1189–1232.

Georgiev, I., D. I. Harvey, S. J. Leybourne, and A. M. R. Taylor (2018). Testing for parameter instability in predictive regression models. *Journal of Econometrics 204*(1), 101–118.

Georgiev, I., D. I. Harvey, S. J. Leybourne, and A. M. R. Taylor (2019). A bootstrap stationarity test for predictive regression invalidity. *Journal of Business & Economic Statistics 37*(3), 528–541.

Gonzalo, J. and J.-Y. Pitarakis (2012). Regime-specific predictability in predictive regressions. *Journal of Business & Economic Statistics 30*(2), 229–241.

Granger, C. W. J. (1999). Outline of forecast theory using generalized cost functions. *Spanish Economic Review 1*(2), 161–173.

Greene, W. H. (2003). *Econometric analysis.* Pearson Education India.

Hansen, B. E. (1992). Convergence to stochastic integrals for dependent heterogeneous processes. *Econometric Theory 8*(4), 489–500.

Herce, M. A. (1996). Asymptotic theory of LAD estimation in a unit root process with finite variance errors. *Econometric Theory 12*(1), 129–153.

Hothorn, T., K. Hornik, and A. Zeileis (2006). Unbiased recursive partitioning: A conditional inference framework. *Journal of Computational and Graphical statistics 15*(3), 651–674.

Huber, P. J. (1981). *Robust statistics.* Wiley.

Juhl, T. (2014). A nonparametric test of the predictive regression model. *Journal of Business & Economic Statistics 32*(3), 387–394.

Karalic, A. (1992). Employing linear regression in regression tree leaves. In *Proceedings of ECAI'92.* Wiley & Sons.

Kasparis, I., E. Andreou, and P. C. Phillips (2015). Nonparametric predictive regression. *Journal of Econometrics 185*(2), 468–494.

Kostakis, A., T. Magdalinos, and M. P. Stamatogiannis (2015). Robust econometric inference for stock return predictability. *Review of Financial Studies 28*(5), 1506–1553.

Kurtz, T. G. and P. Protter (1991). Weak limit theorems for stochastic integrals and stochastic differential equations. *Annals of Probability 19*(3), 1035–1070.

Lee, J. H. (2016). Predictive quantile regression with persistent covariates: IVX-QR approach. *Journal of Econometrics 192*(1), 105–118.

Lucas, A. (1995). Unit root tests based on m estimators. *Econometric Theory 11*(2), 331–346.

Meligkotsidou, L., E. Panopoulou, I. D. Vrontos, and S. D. Vrontos (2014). A quantile regression approach to equity premium prediction. *Journal of Forecasting 33*(7), 558–576.

Meligkotsidou, L., E. Panopoulou, I. D. Vrontos, and S. D. Vrontos (2019). Out-of-sample equity premium prediction: A complete subset quantile regression approach. *The European Journal of Finance*, forthcoming.

Menze, B. H., B. M. Kelm, D. N. Splitthoff, U. Koethe, and F. A. Hamprecht (2011). On oblique random forests. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 453–469. Springer.

Newey, W. K. and J. L. Powell (1987). Asymmetric least squares estimation and testing. *Econometrica 55*(4), 819–847.

Paye, B. S. and A. Timmermann (2006). Instability of return prediction models. *Journal of Empirical Finance 13*(3), 274–315.

Phillips, P. C. B. (1987). Towards a Unified Asymptotic Theory for Autoregression. *Biometrika 74*(3), 535–547.

Phillips, P. C. B. (2015). Pitfalls and possibilities in predictive regression. *Journal of Financial Econometrics 13*(3), 521–555.

Phillips, P. C. B. and J. H. Lee (2013). Predictive regression under various degrees of persistence and robust long-horizon regression. *Journal of Econometrics 177*(2), 250–264.

Rapach, D. E., J. K. Strauss, and G. Zhou (2010). Out-of-sample equity premium prediction: Combination forecasts and links to the real economy. *The Review of Financial Studies 23*(2), 821–862.

Stambaugh, R. F. (1999). Predictive regressions. *Journal of Financial Economics 54*(3), 375–421.

Stock, J. H. and M. W. Watson (2004). Combination forecasts of output growth in a seven-country data set. *Journal of forecasting 23*(6), 405–430.

Welch, I. and A. Goyal (2008). A comprehensive look at the empirical performance of equity premium prediction. *The Review of Financial Studies 21*(4), 1455–1508.

Xu, K.-L. and J. Guo (2019). A dimensionality-robust test in multiple predictive regression. *Available at SSRN 3458074*.

Zhang, H., Q. He, B. Jacobsen, and F. Jiang (2020). Forecasting stock returns with model uncertainty and parameter instability. *Journal of Applied Econometrics*.