Reservoir Computing with Applications to Time Series Forecasting

Lyudmila Grigoryeva

University of Konstanz, Germany

Summer School
University of Vienna
Vienna, 2020

Outline

1. Universal reservoir system: Echo State Network (ESN)
2. Applications to stochastic processes
3. Conventional tools
4. Forecasting with ESN
5. Empirical study

Universal reservoir system: Echo State Network (ESN)

Echo State Network is given by:

$$\begin{cases} \mathbf{x}_t = \boldsymbol{\sigma}\left(A\mathbf{x}_{t-1} + \gamma C\mathbf{z}_t + s\boldsymbol{\zeta}\right) & (1) \\ \mathbf{y}_t = W^\top \mathbf{x}_t & (2) \end{cases}$$

The reservoir map $F : \mathbb{R}^N \times \mathbb{R}^d \to \mathbb{R}^N$ is prescribed by:

- the activation function $\boldsymbol{\sigma} : \mathbb{R}^N \longrightarrow \mathbb{R}^N$
- reservoir matrix $A \in \mathbb{M}_N$
- input mask $C \in \mathbb{M}_{N,d}$
- input scaling $\gamma \in \mathbb{R}^+$
- input shift $\boldsymbol{\zeta} \in \mathbb{R}^N$
- input shift scaling $s \in \mathbb{R}^+$

Architecture choice: number of neurons N, the law for the elements of A, C, $\boldsymbol{\zeta}$.
Only the reservoir readout is $W \in \mathbb{M}_{N,m}$ is subject to training.
In some cases hyperparameters $\boldsymbol{\theta} := (\rho(A), \gamma, s)$ need to be tuned.

Architecture choice for ESNs

Approaches to the choice of ESN architecture

- hyperparameters $\boldsymbol{\theta}$ given by the solution of the ERM optimization problem constructed using some loss function of interest or randomly sampled
- A is taken as a sparse matrix with the connectivity degree often set up as $c = \min\{10/N, 1\}$
- $A \in \mathbb{M}_N$ and $C \in \mathbb{M}_{N,d}$ ($\boldsymbol{\gamma} \in \mathbb{R}^N$) are randomly drawn (Gaussian, uniform).

Training of ESNs: Estimating the linear readout W

Consider

- initial reservoir state $\mathbf{x}_0 \in \mathbb{R}^N$, T length of the total training sample
- $Z \in \mathbb{M}_{d,T}$ observation matrix of the training input
- $Y \in \mathbb{M}_{m,T}$ observation matrix of the training target (for the forecasting task the target process $\mathbf{y} = T_{-1}(\mathbf{z})$)
- $X \in \mathbb{M}_{N,T}$ contains the T states of the ESN

Then the estimation of the **linear readout** (up to bias term) of the ESN is implemented via Tikhonov-regularized LS regression (ERM w.r.t. squared loss), namely:

$$\widehat{W} = \underset{W \in \mathbb{M}_{N,m}}{\arg\min} \left\{ \| Y - W^\top X \|_F^2 + \lambda \| W \|_F^2 \right\}, \quad \lambda \in \mathbb{R}^+,$$

with the closed-form solution

$$\widehat{W} = (XX^\top + \lambda \mathbb{I}_N)^{-1} XY^\top.$$

Forecasting with ESN (deterministic setup)

One step ahead:

$$\begin{cases} \mathbf{x}_t = \boldsymbol{\sigma}\left(A\mathbf{x}_{t-1} + \gamma C\mathbf{z}_t + s\boldsymbol{\zeta}\right) \\ \hat{\mathbf{z}}_{t+1} = \widehat{W}^\top \mathbf{x}_t, \end{cases}$$

which is called in-sample (training) for $t = 1, \ldots, T$ and out-of-sample (testing) for $t = T+1, \ldots, T+T^{tst}$ with T^{tst} the length of the testing sample. The forecast testing error is given by

$$\hat{R}_T(\widehat{W}) = \frac{1}{T^{tst}} \sum_{t=T+1}^{T+T^{tst}} L(\hat{\mathbf{z}}_t, \mathbf{z}_t)$$

h-steps ahead:

$$\begin{cases} \mathbf{x}_t = \boldsymbol{\sigma}\left(A\mathbf{x}_{t-1} + \gamma C\hat{\mathbf{z}}_t + s\boldsymbol{\zeta}\right) \\ \hat{\mathbf{z}}_{t+1} = \widehat{W}^\top \mathbf{x}_t \end{cases}$$

where $t = 1, \ldots, T+h$, $h \leq T^{tst}$, and $\hat{\mathbf{z}}_t = \mathbf{z}_t$ for $t = 1, \ldots, T$.

Stochastic setup

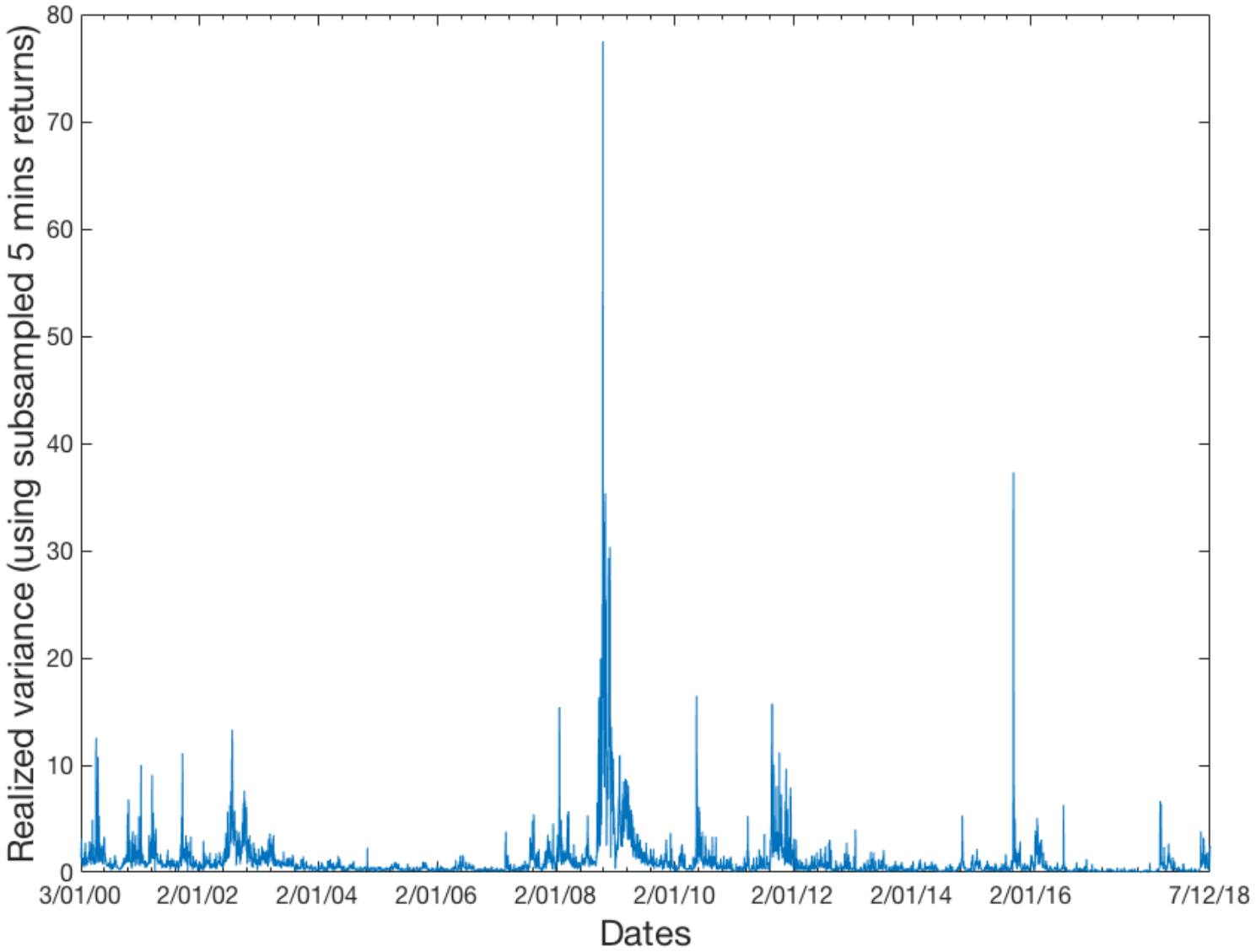
See [GO20] for the universal approximation properties of the ESNs and [GGO19] for their generalization error bounds for weakly dependent processes (non-sharp). Empirical question: how difficult is to find ESN which outperforms benchmark competitors in the forecasting exercise.

Application to stochastic processes

Task: forecasting realized (co)variances of intradaily (multiple) asset returns

- Application of interest for financial mathematicians and financial econometricians
- Realized (co)variances exhibit the stylized features of financial time series along with the features of long memory processes
- Existing parametric models suffer from the curse of dimensionality when applied to high number of financial assets
- Only short sample of historical observations is available (poor statistical inference guarantees)
- Time series show signs of extreme behaviour and regime changing (crisis events)

Realized variance of financial asset returns



Signs of long memory behavior

Realized volatility time series exhibit long memory behavior features:

(i) Slow decay in lag k in sample auto-correlations

$$\hat{\rho}(k) = \frac{1}{n}\sum_{t=1}^{n-|k|}(z_t - \overline{z}_n)(z_{t+|k|} - \overline{z}_n) \text{ with } \overline{z}_n = \sum_{t=1}^{n} z_t.$$

The decay of $\hat{\rho}(k)$ is hyperbolic with a rate $k^{-\alpha}$, $0 < \alpha < 1 \implies$ non-summability.

(ii) $\mathrm{Var}(\overline{z}_n) \to 0$ at a slower rate than n^{-1}. Define

$$S_m^2 = 1/(n_m - 1)\sum_{i=1}^{n_m}\left(\overline{z}_{(i-1)m,im} - \overline{z}_n\right)^2, \quad \overline{z}_{t,m} = \frac{1}{m}\sum_{j=1}^{m} z_{t+j}, \quad n_m = \left\lceil \frac{n}{m} \right\rceil.$$

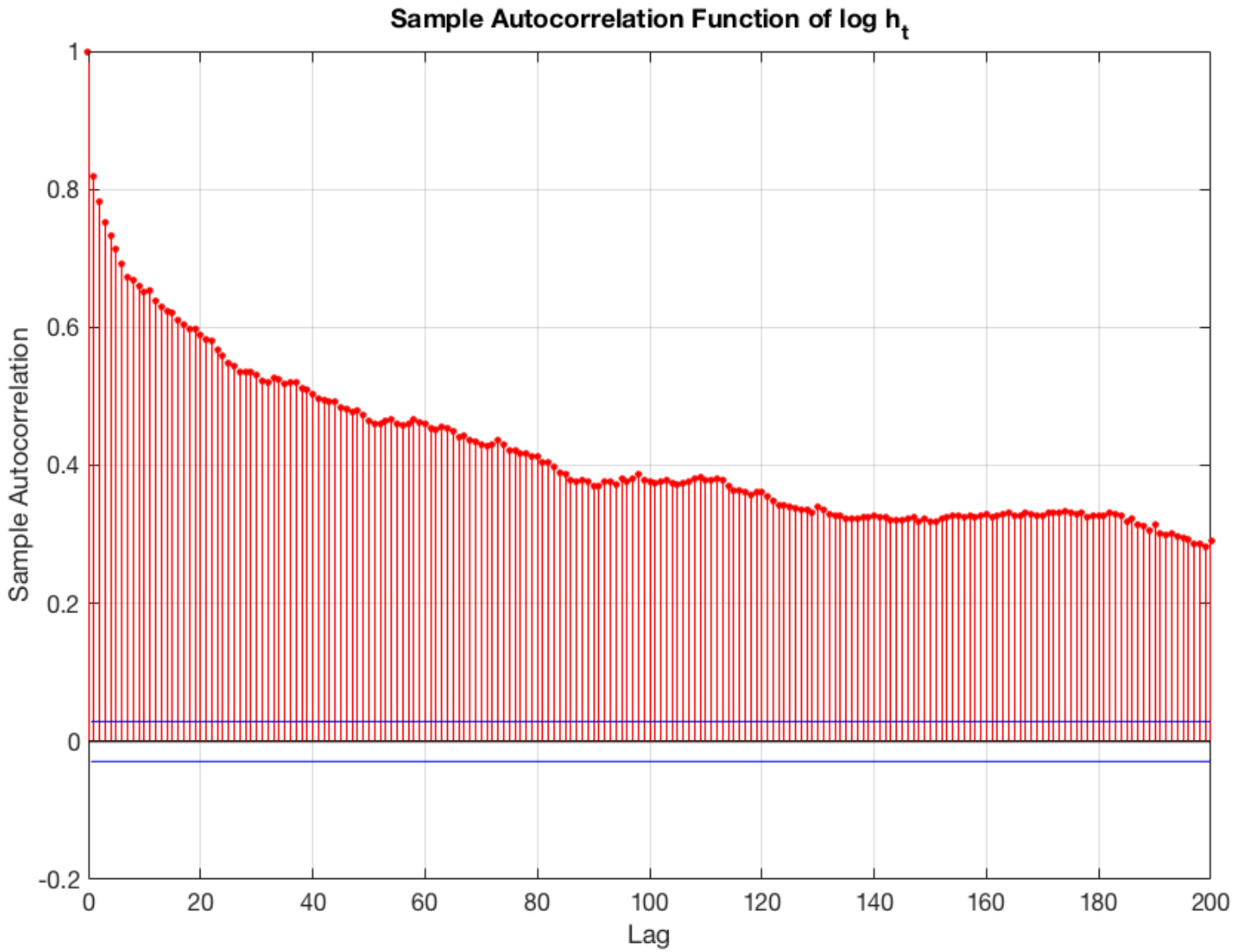
Plot S_m^2 against $\log m$ where m is the number of observations in the block.

(iii) **R/S statistic** [?]: Let $y_i = \sum_{i=1}^{j} z_i$ and define the adjusted range

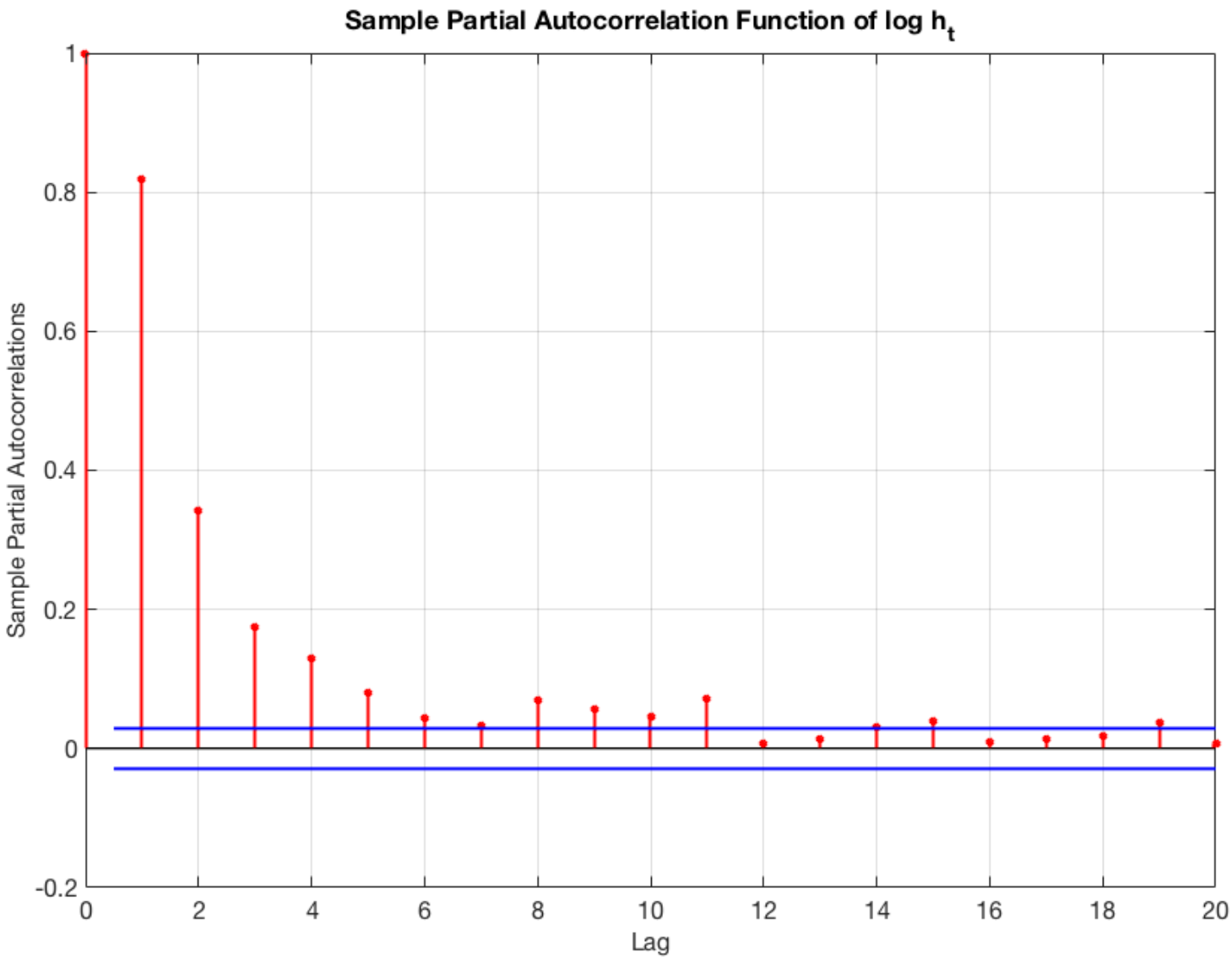
$$R(t,k) := \max_{0\le i\le k}\{y_{t+i} - y_t - \frac{i}{k}(y_{t+k} - y_t)\} - \min_{0\le i\le k}\{y_{t+i} - y_t - \frac{i}{k}(y_{t+k} - y_t)\}.$$

Let $\overline{z}_{t,k} = k^{-1}\sum_{i=t+1}^{t+k} z_i$ and define $S(t,k) := \sqrt{k^{-1}\sum_{i=t+1}^{t+k}(z_i - \overline{z}_{t,k})^2}$. The ratio $\frac{R(t,k)}{S(t,k)}$ is called the R/S statistic. As $k \to \infty$, $\log E[R/S] \approx a + H \log k$ with $H > \frac{1}{2}$ for short-range processes R/S behaves as $\sqrt{k}$, that is as $k \to \infty$, $\log R/S$ slope

Autocorrelation of log realized variance



Partial autocorrelation of log realized variance



Long memory processes

Standard characterisation of memory for time series process, which are second-order stationary, builds upon properties of autocovariance functions $\gamma_z(h), h \in \mathbb{Z}$ or spectral density functions

$$f_z(\omega) = \frac{1}{2n} \sum_{h=-\infty}^{\infty} \gamma_z(h) \exp(-ih\omega), \omega \in [-\pi, \pi]. \quad (3)$$

Definition ([Ber94])

We say that a second-order stationary process $z_t, t \in \mathbb{Z}$

- has long memory, if as $|\omega| \to 0 \quad f_z(\omega) \to \infty$ or $\sum_{h\in\mathbb{Z}} \gamma_z(h) = \infty$
- has short memory, if as $|\omega| \to 0 \quad f_z(\omega) \to c$ or $0 < \sum_{h\in\mathbb{Z}} \gamma_z(h) < \infty$
- shows antipersistence, if as $|\omega| \to 0 \quad f_z(\omega) \to 0$ or $\sum_{h\in\mathbb{Z}} \gamma_z(h) = 0$.

Using the equiv. characterization of processes [Ber94], one can define memory of a given process using the behavior of $\gamma_z(h), h \in \mathbb{Z}$ and use in practice heuristics given above.

Definition

Let x_t be a stationary process. x_t is called a process with **long memory, long range dependence or strong dependence** if there exists a constant $c_\gamma > 0$ such that

$$\lim_{|h|\to\infty} \gamma_z(h) = c_\gamma h^{-\alpha}, \text{ with } \alpha < 1. \quad (4)$$

ARFIMA process

An ARFIMA(p, d, q) (Granger and Joyeux [1980], Hosking [1981]) process is given by

$$\Phi(L)(1-L)^d(z_t-\nu)=\Theta(L)\epsilon_t, \quad \epsilon \sim IID(0,\sigma_\epsilon^2)$$

with

$$\Phi(L)=1-\varphi_1 L-\varphi_2 L-\ldots-\varphi_p L^p$$
$$\Theta(L)=1+\theta_1 L+\theta_2 L+\ldots+\theta_q L^q$$

and the fractional difference operator $(1-L)^d$ defined by

$$(1-L)^d=\sum_{k=0}^{\infty}\frac{\Gamma(k-d)L^k}{\Gamma(-d)\Gamma(k+1)} \tag{5}$$

z_t is invertible and stationary if all roots of Φ and Θ are outside the unit circle and $|d|<0.5$. A stationary ARFIMA(p, d, q) process with $d\in(0,\frac{1}{2})$ is a long memory process. It is easy to verify that it that case, the condition (4) holds with $\alpha=2d-1$.

HAR [Cor09]

The daily return process $\{r_t\}_{t\in\mathbb{Z}}$ is given by

$$r_t = \sigma_t^{(d)}\varepsilon_t, \;\; \varepsilon_t \overset{IID}{\sim} N(0,1), \;\; t \in \mathbb{Z}, \tag{6}$$

with $\sigma_t^{(d)}$ the daily integrated volatility. Idea: the hierarchical model for $\sigma_t^{(d)}$ using a cascade of models for the partial latent volatilities at lower frequencies. Introduce the daily partial realized volatilities $\widetilde{\sigma}_t^{(d)}$ such that $\widetilde{\sigma}_t^{(d)} = \sigma_t^{(d)}$, the weekly $\widetilde{\sigma}_t^{(w)}$ and monthly $\widetilde{\sigma}_t^{(m)}$ latent partial volatilities. Assume they follow for all $t \in \mathbb{Z}$ hold

$$\widetilde{\sigma}_{t+1m}^{(m)} = \alpha^{(m)} + \phi^{(m)} RV_t^{(m)} + \widetilde{\epsilon}_{t+1m}^{(m)}, \tag{7}$$

$$\widetilde{\sigma}_{t+1w}^{(w)} = \alpha^{(w)} + \phi^{(w)} RV_t^{(w)} + \gamma^{(w)}\mathbb{E}_t[\widetilde{\sigma}_{t+1m}^{(m)}] + \widetilde{\epsilon}_{t+1w}^{(w)}, \tag{8}$$

$$\widetilde{\sigma}_{t+1d}^{(d)} = \alpha^{(d)} + \phi^{(d)} RV_t^{(d)} + \gamma^{(w)}\mathbb{E}_t[\widetilde{\sigma}_{t+1w}^{(w)}] + \widetilde{\epsilon}_{t+1d}^{(d)}, \tag{9}$$

where $RV_t^{(d)}$ is the observed daily realized volatility, $RV_t^{(w)} = \frac{1}{5}\sum_{j=0}^{4} RV_{t-jd}^{(d)}$ and $RV_t^{(m)} = \frac{1}{22}\sum_{j=0}^{21} RV_{t-jd}^{(d)}$, $\widetilde{\epsilon}_{t+1m}^{(m)}$, $\widetilde{\epsilon}_{t+1w}^{(w)}$, and $\widetilde{\epsilon}_{t+1d}^{(d)}$ are contemp. and serially independent zero-mean innovations, which are left tail truncated.

HAR

The hierarchical approach yields

$$\sigma^{(d)}_{t+1d} = \alpha + \beta^{(d)} RV^{(d)}_t + \beta^{(w)} RV^{(w)}_t + \beta^{(m)} RV^{(m)}_t + \tilde{\epsilon}^{(d)}_{t+1d}, \quad t \in \mathbb{Z}, \tag{10}$$

where $\alpha = \alpha^{(d)} + \gamma^{(d)}\alpha^{(w)} + \gamma^{(d)}\gamma^{(w)}\alpha^{(m)}$, $\beta^{(d)} = \phi^{(d)}$, $\beta^{(w)} = \gamma^{(d)}\phi^{(w)}$, and $\beta^{(m)} = \gamma^{(d)}\gamma^{(w)}\phi^{(m)}$. With

$$\sigma^{(d)}_{t+1d} = RV^{(d)}_{t+1d} + \epsilon^{(d)}_{t+1d}, \tag{11}$$

results in HAR model

$$RV^{(d)}_{t+1} = \alpha + \beta^{(d)} RV^{(d)}_t + \beta^{(w)} RV^{(w)}_t + \beta^{(m)} RV^{(m)}_t + \epsilon_{t+1}, \quad t \in \mathbb{Z}, \tag{12}$$

with $\epsilon_{t+1} = \tilde{\epsilon}^{(d)}_{t+1d} - \epsilon^{(d)}_{t+1d}$ with the temporal increments taken in the daily time scale.

HAR is a AR(22) model parametrized in a parsimonious way.

Other versions of HAR model [AK16, AHO19]; produce superior quality forecasts in the calm market periods; suffer from losses of performance in periods of high volatile market behavior; remain very strong competitors in the univariate and multivariate setups [SSKM18].

Pitfalls and tools

- transforms preserving positivity (positive definiteness) of (co)volatilities - Box-Cox
- choice of a particular loss function and regularization at the time of training which leads to good performance in terms of other evaluation criteria used in the literature
- residual (block) bootstrapping
- changing the architecture as a response to regime switching
- random ESNs with the Hedge boosting algorithm

Box-Cox

Consider the original time series y_t, $t \in \{1, \ldots, T\}$ of realized variances. Since y_t is a non-negative process and its realizations often exhibit non-Gaussian behavior, one applies the Box-Cox transformation, namely for all $t \in \{1, \ldots, T\}$

$$z_t = f_{\mathrm{BC}}(y_t; \lambda) = \begin{cases} \dfrac{y_t^{\lambda} - 1}{\lambda}, & \lambda \neq 0, \\ \ln y_t, & \lambda = 0 \end{cases} \tag{13}$$

and whose inverse we denote by $g_{\mathrm{BC}}(z_t, \lambda) = f_{\mathrm{BC}}^{-1}(z_t, \lambda) = y_t$. Based on z_t, $t \in \{1, \ldots, T\}$ for each $h \in \{1, \ldots, h_{\max}\}$ construct the forecast $\hat{z}_{t+h} := \mathrm{E}[z_{t+h}|\mathcal{F}_t]$. The ultimate goal of the forecaster however is to obtain the forecast in the original representation $\hat{y}_{t+h} := \mathrm{E}[y_{t+h}|\mathcal{F}_t]$. The naïve approach of obtaining a forecast as evaluation of the inverse function $g_{\mathrm{BC}}(\hat{z}_{t+h}, \lambda)$ for convex functions by Jensen's inequality leads to under-predictions as $g_{\mathrm{BC}}(\mathrm{E}[z_{t+h}|\mathcal{F}_t], \lambda) \leq \mathrm{E}[g_{\mathrm{BC}}(z_{t+h}; \lambda)|\mathcal{F}_t]$ and may result in incorrect rankings of models.

Forecast adjustment

Adjustment in [Tay17] based on the conditional expectation of a Taylor series expansion of $g_{\mathrm{BC}}(z_{t+h};\lambda)$ at z_{t+h}. Denote by $\mu^{(k)}_{t+h|t}$, $k \in \mathbb{N}$, the k-th conditional central moments of z_{t+h} that is $\mu^{(1)}_{t+h|t} = \mathrm{E}[z_{t+h}|\mathcal{F}_t]$ and $\mu^{(k)}_{t+h|t} = \mathrm{E}[z^k_{t+h}|\mathcal{F}_t] - (\mu^{(1)}_{t+h|t})^k$ for $k > 1$ and the *full adjustment* expression is given by

$$\mathrm{E}[g_{\mathrm{BC}}(z_{t+h};\lambda)|\mathcal{F}_t] = g_{\mathrm{BC}}(\mu_{t+h|t};\lambda)\left(1 + \sum_{k=1}^{\infty} g^{(k)}(\mu_{t+h|t};\lambda)\mu^{(k)}_{t+h|t}\right) \tag{14}$$

where

$$g^{(k)}(\mu_{t+h|t};\lambda) = \frac{1-\lambda(k-1)}{k(1+\lambda\mu_{t+h|t})} g^{(k-1)}(\mu_{t+h|t};\lambda), \quad \text{with} \quad g^{(0)}(\mu_{t+h|t};\lambda) = 0.$$

In particular, in the case of the logarithmic transformation ($\lambda = 0$ in (13)) the full adjustment has the form

$$\hat{y}_{t+h} \approx e^{\mu_{t+h|t}}\left(1 + \sum_{k=1}^{\infty} \frac{1}{k!}\mu^{(k)}_{t+h|t}\right) \tag{15}$$

Doob decomposition implications for forecasting

Theorem (Doob decomposition)

Let $(\Omega, \mathcal{F}, \mathbb{P})$ be a probability space and let $(\mathcal{F}_t)_{t\in\mathbb{N}}$ be a filtration of $\mathcal{A}$. Let $\mathbf{z} = (z_t)_{t\in\mathbb{N}^+}$ be an adapted stochastic process with $E[|z_t|] < \infty$ for all $t \in \mathbb{N}^+$. Then there exists a martingale $M = (M_t)_{t\in\mathbb{N}^+}$ and an integrable and predictable process $A = (A_t)_{t\in\mathbb{N}}$ starting at $A_1 = 0$, such that $z_t = A_t + M_t$ for all $t \in \mathbb{N}^+$. This decomposition is almost surely unique.

For all $t \in \mathbb{N}^+$

$$A_t = \sum_{j=2}^{t} \left(\mathrm{E}[z_j|\mathcal{F}_{j-1}] - z_{j-1}\right), A_1 = 0 \tag{16}$$

$$M_t = z_1 + \sum_{j=2}^{t} \left(z_j - \mathrm{E}[z_j|\mathcal{F}_{j-1}]\right), M_1 = z_1 \tag{17}$$

are used to verify the decomposition. Additionally, one needs to check that $E[\underbrace{(M_t - M_{t-1})}_{\epsilon_t}|\mathcal{F}_{t-1}] = 0$ almost surely, which follows from (17). Consider now the implications of using Doob's decompsition for the forecasting exercise.

Doob decomposition for forecasting

Let $\mathbf{z} = \mathbf{z}_{\text{train}} = (z_0, z_1, \ldots, z_{\text{train}-1})^T \in \mathbb{R}^{T_{\text{train}}}$. Let $T = T_{\text{train}}$ and let $\hat{z}_t = \mathrm{E}[z_t|\mathcal{F}_{t-1}], t \in \{2, \ldots, T\}$ with $\hat{z}_1 = z_1$ be the forecast provided by ESN previously trained on the 1-step ahead forecasting task. Additionally define $\hat{\epsilon}_t := z_t - \hat{z}_t$, the in sample residuals produced by the trained ESN. In the end of the training phase, at $t = T$, one has:

$$A_T = \sum_{t=2}^{T}(\hat{z}_t - z_{t-1}) \text{ and } M_T = z_1 + \sum_{t=2}^{T}(z_t - \hat{z}_t) = z_1 + \sum_{t=2}^{T}\hat{\epsilon}_t$$

Implementation

Let N_r be the number of bootstrap replications. We construct N_r random subsamples of residuals at the time of training, each of length $h_{\max}$, namely $\hat{E}^{(j)} = \left(\hat{\epsilon}_1^{(j)}, \hat{\epsilon}_2^{(j)}, \ldots, \hat{\epsilon}_{h_{\max}}^{(j)}\right), j = \{1, \ldots, N_r\}$.

- $h = 1$:

$$\tilde{z}_{T+1}^{(j)} = \hat{A}_{T+1} + \tilde{M}_{T+1}^{(j)}, j = 1, \ldots, N_r, \text{with}$$

$$\hat{A}_{T+1} = \underbrace{\sum_{t=2}^{T} (\hat{z}_t - z_{t-1})}_{A_T} + (\hat{z}_{T+1} - z_T)$$

$$\tilde{M}_{T+1}^{(j)} = \underbrace{z_1 + \sum_{t=2}^{T} (z_t - \hat{z}_t)}_{M_T} + \underbrace{(z_{T+1} - \hat{z}_{T+1})}_{\text{not available } \hat{\epsilon}_{T+1}} =: M_T + \hat{\epsilon}_1^{(j)}$$

with $\hat{z}_{T+1}$ provided by the ESN.

- $h = 2$:

$$\tilde{z}_{T+2}^{(j)} = \tilde{A}_{T+2}^{(j)} + \tilde{M}_{T+2}^{(j)}, j = 1, \ldots, N_r, \text{with}$$
$$\tilde{A}_{T+2}^{(j)} = \hat{A}_{T+1} + (\hat{z}_{T+2} - z_{T+1})$$
$$\tilde{M}_{T+2}^{(j)} = \tilde{M}_{T+1}^{(j)} + \hat{\epsilon}_2^{(j)}$$

- until $h = h_{\max}$:

$$\tilde{z}_{T+h_{\max}}^{(j)} = \tilde{A}_{T+h_{\max}}^{(j)} + \tilde{M}_{T+h_{\max}}^{(j)}, j = 1, \ldots, N_r, \text{with}$$
$$\tilde{A}_{T+h_{\max}}^{(j)} = \hat{A}_{T+h_{\max}-1} + (\hat{z}_{T+h_{\max}} - z_{T+h_{\max}-1})$$
$$\tilde{M}_{T+h_{\max}}^{(j)} = \tilde{M}_{T+h_{\max}-1}^{(j)} + \hat{\epsilon}_{h_{\max}}^{(j)}$$

Finally construct forecasts as:

$$\hat{z}_{t+h}^{\text{esn}} = \frac{1}{N_r} \sum_{j=1}^{N_r} \tilde{z}_{t+h}^{(j)}, h = 1, \ldots, h_{\max}, j = 1, \ldots, N_r.$$

$$\tilde{z}_{T+1}^{(j)} = \hat{z}_{T+1}^{\text{esn}} + \hat{\epsilon}_1^{(j)}$$

$$\vdots$$

$$\tilde{z}_{T+h_{\max}}^{(j)} = \hat{z}_{T+h_{\max}}^{\text{esn}} + \hat{\epsilon}_{h_{\max}}^{(j)}$$

Forecasting with ESNs

h-step ahead forecast of realized (co)variances is constructed as follows:

- Fix the number N_r of replications in the bootstrapping exercise.
- Randomly draw N_r h-long subsamples $\left\{\epsilon_j^{*(1)}, \ldots, \epsilon_j^{*(h)}\right\}_{j\in\{1,\ldots,N_r\}}$ of residuals from the available ESN residuals $\{\widehat{\epsilon}_2, \ldots, \widehat{\epsilon}_{T_{est}}\}$.
- Follow according to the ESN model, for each $t \in \{T_{est}, \ldots, T-h\}$ let $\mathbf{z}_t := \mathbf{RV}_t^d$, then construct s-step forecasts, $s = 1, \ldots, h$, each N_r times

$$\begin{cases} \mathbf{x}_{t+s-1}^j = \sigma\left(A\mathbf{x}_{t+s-2}^j + \gamma C(\widehat{\mathbf{z}}_{t+s-1}^j + \epsilon_j^{*(s)}) + \zeta\right), \\ \widehat{\mathbf{RV}}_{t+s}^{d,j} := \widehat{\mathbf{z}}_{t+s}^j = \widehat{W}\mathbf{x}_{t+s-1}^j, \quad j \in \{1, \ldots, N_r\}, \ \ s \in \{1, \ldots, h\} \end{cases}$$

- Finally construct each s-step forecast as $\widehat{\mathbf{RV}}_{t+s}^d := \frac{1}{N_r}\sum_{j=1}^{N_r} \widehat{\mathbf{RV}}_{t+s}^{d,j}$

Experts Advice (the Hedge algorithm [FS97], [FS99])

Setup:

- K is the number of ESNs-experts
- T is the length of the sample for forecasting with $\{1, \ldots, T\}$
- consider h-step forecasts $h_{t+h}^{(j)}$, $t \in \{1, \ldots, T-H\}$, $h \in \{1, \ldots, H\}$, produced by each jth expert, $j \in \{1, \ldots, K\}$
- performance of each expert (accuracy of the produced forecast) is assessed with the help of the loss function $\ell : \mathbb{R} \longrightarrow \mathbb{R}^+$ of a player's choice
- fix the initial weights $\mathbf{w}_0^h := (w_0^{(1),h}, \ldots, w_0^{(K),h})^\top$ s.t. $\sum_{j=1}^K w_0^{(j),h} = 1$ and the updating rate $\eta \in \mathbb{R}^+$

The **Hedge algorithm** consists of the following steps:

1. Let $t = t_0$, the initial weights are used $\mathbf{w}_t^h = \mathbf{w}_0^h$
2. Produce h-step ahead forecasts $\mathbf{h}_{t,h} = (h_t^{(1),h}, \ldots, h_t^{(K),h})^\top$. The player's forecast is constructed as $\hat{h}_t^h = \mathbf{w}_t^h \mathbf{h}_{t,h}^\top$
3. Let $t = t + 1$
4. The true observation σ_t is revealed; one can assess performance of each 1-step forecast of the jth expert is assessed; one gets $(\ell_{t-1}^{(1),1}, \ldots, \ell_{t-1}^{(K),1})$ with $\ell_{t-1}^{(j),1} := \ell(\sigma_t, h_{t-1}^{(j),1})$. Compute the player's loss $\ell_{t-1}^1 = \ell(\sigma_t, \hat{h}_{t-1}^1)$
5. Update only the weights with $h = 1$ via the rule $w_t^{(j),1} := w_{t-1}^{(j),1} \cdot \exp(-\eta \ell_{t-1}^{(j),1})$, $j \in \{1, \cdots, K\}$. Normalize these weights and leave the other weights unchanged : $\mathbf{w}_t^2 = \mathbf{w}_{t-1}^2, \cdots, \mathbf{w}_t^H = \mathbf{w}_{t-1}^H$.
6. Continue steps (2)-(5), each time more true values get revealed and eventually whenever $t \geq t_0 + H$ all the weights get updates. Continue until $t = T - h$.

Datasets

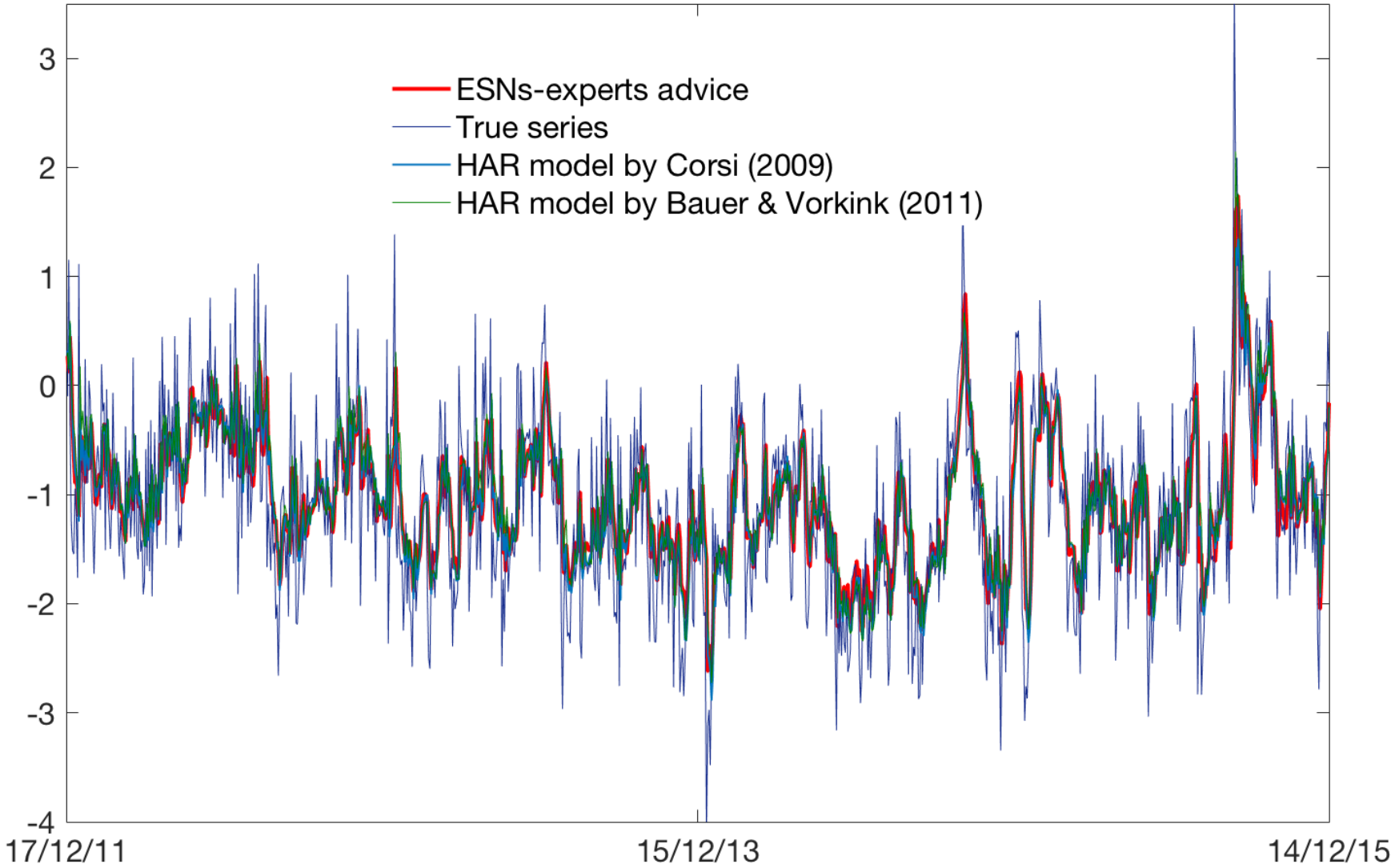
1. Univariate historical data of the 5-min subsampled realized variances of the major equity indices in the *period* 04/01/2000 - 07/12/2018:
   - S&P500
   - FTSE 100
   - NIKKEI225
2. Multivariate time series of historically observed realized covariance matrix processes:
   - 4 assets:
     CTL (CenturyLink), MKR (Merck), JPM (JPMorgan), PFE (Pfizer)
     Period: 10/09/2003–06/03/2018
   - 6 assets:
     AXP (American Express), C (Citigroup), GS (Goldman Sachs), BLK (BlackRock), AA (Alcoa), GE (General Electric)
     Period: 04/01/2001–16/04/2018
   - 29 constituents of the DJIA index
     Period: 04/01/2001–16/04/2018
   - 128 the most liquid S&P500 index assets-components
     Period: 04/01/2001–16/04/2018

Forecast evaluation

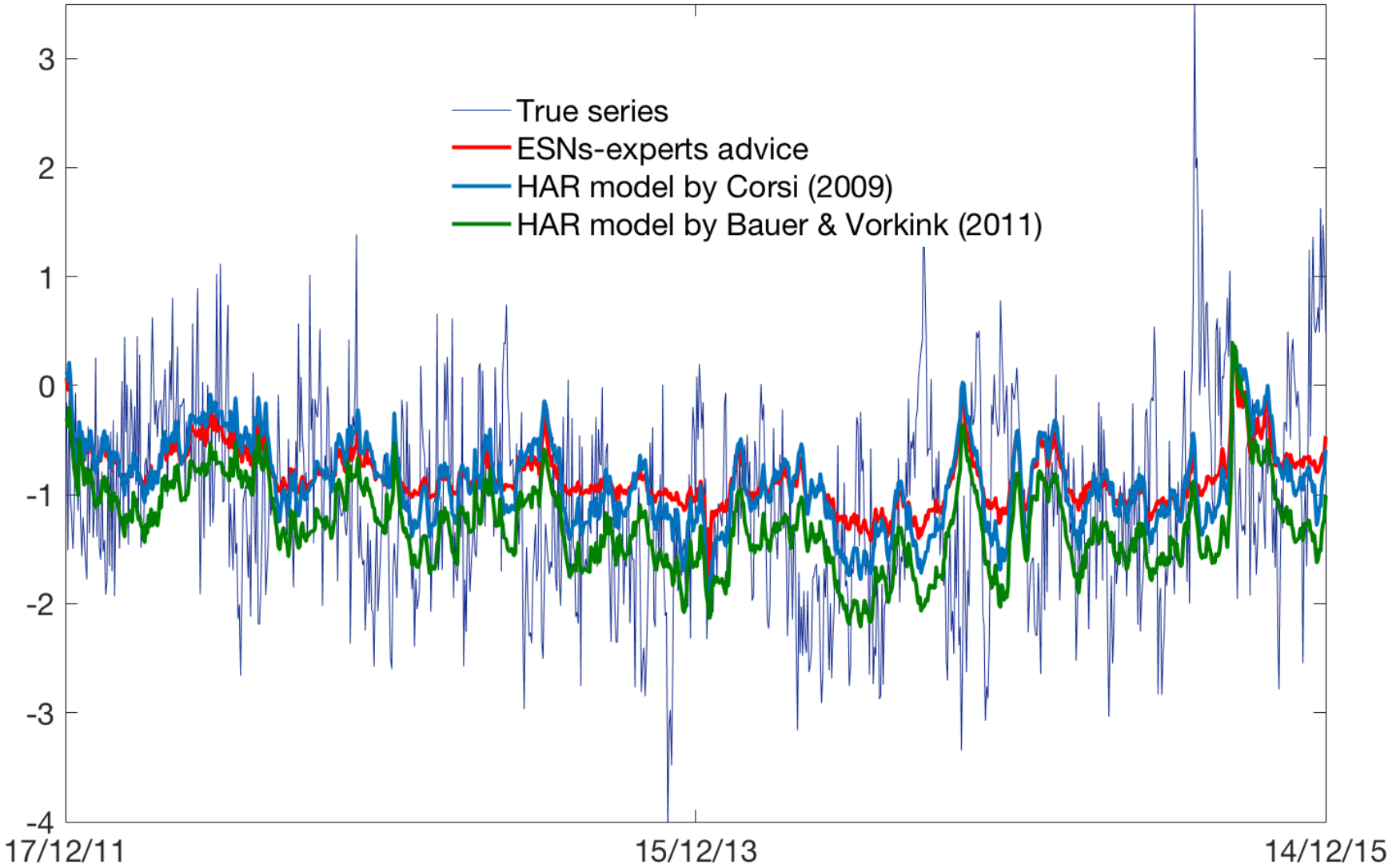
Consider (in)consistent loss functions [Pat11], [LRV13] to evaluate the in-sample fit and the out-of-sample forecasting ability of the models:

- Univariate case:
  - Mean absolute error (MAE), ℓ_1-norm based $\mathcal{L}_1(\sigma_t^2, h_t^2) := |\sigma_t^2 - h_t^2|$
  - Mean square error (MSE), Euclidean dist. based $\mathcal{L}_2(\sigma_t^2, h_t^2) := \left(\sigma_t^2 - h_t^2\right)^2$
  - QLIK error based on QLIK loss $\mathcal{L}_{QLIK}(\sigma_t^2, h_t^2) := \log \frac{\sigma_t^2}{h_t^2} + \frac{\sigma_t^2}{h_t}$
- Multivariate case. Let $\boldsymbol{\sigma}_t^v := \operatorname{vech}(\Sigma_t), \boldsymbol{h}_t^v := \operatorname{vech}(H_t) \in \mathbb{R}^{N^*}$:
  - Mean absolute error (MAE), ℓ_1-norm based $\mathcal{L}_1(\Sigma_t, H_t) := \sum_{i=1}^{N^*} |(\boldsymbol{\sigma}_t^v)_i - (\boldsymbol{h}_t^v)_i|$
  - Mean square error (MSE), Euclidean dist. based $\mathcal{L}_E(\Sigma_t, H_t) := (\boldsymbol{\sigma}_t^v - \boldsymbol{h}_t^v)^\top (\boldsymbol{\sigma}_t^v - \boldsymbol{h}_t^v)$
  - QLIK error based on QLIK loss: $\mathcal{L}_{QLIK}(\Sigma_t, H_t) := \log\det H_t + \operatorname{trace}\left\{H_t^{-1}\Sigma_t\right\}$
  - Frobenius norm based: $\mathcal{L}_F(\Sigma_t, H_t) := \|\Sigma_t - H_t\|_{\mathrm{F}} = \operatorname{trace}\left\{(\Sigma_t - H_t)^\top (\Sigma_t - H_t)\right\}$

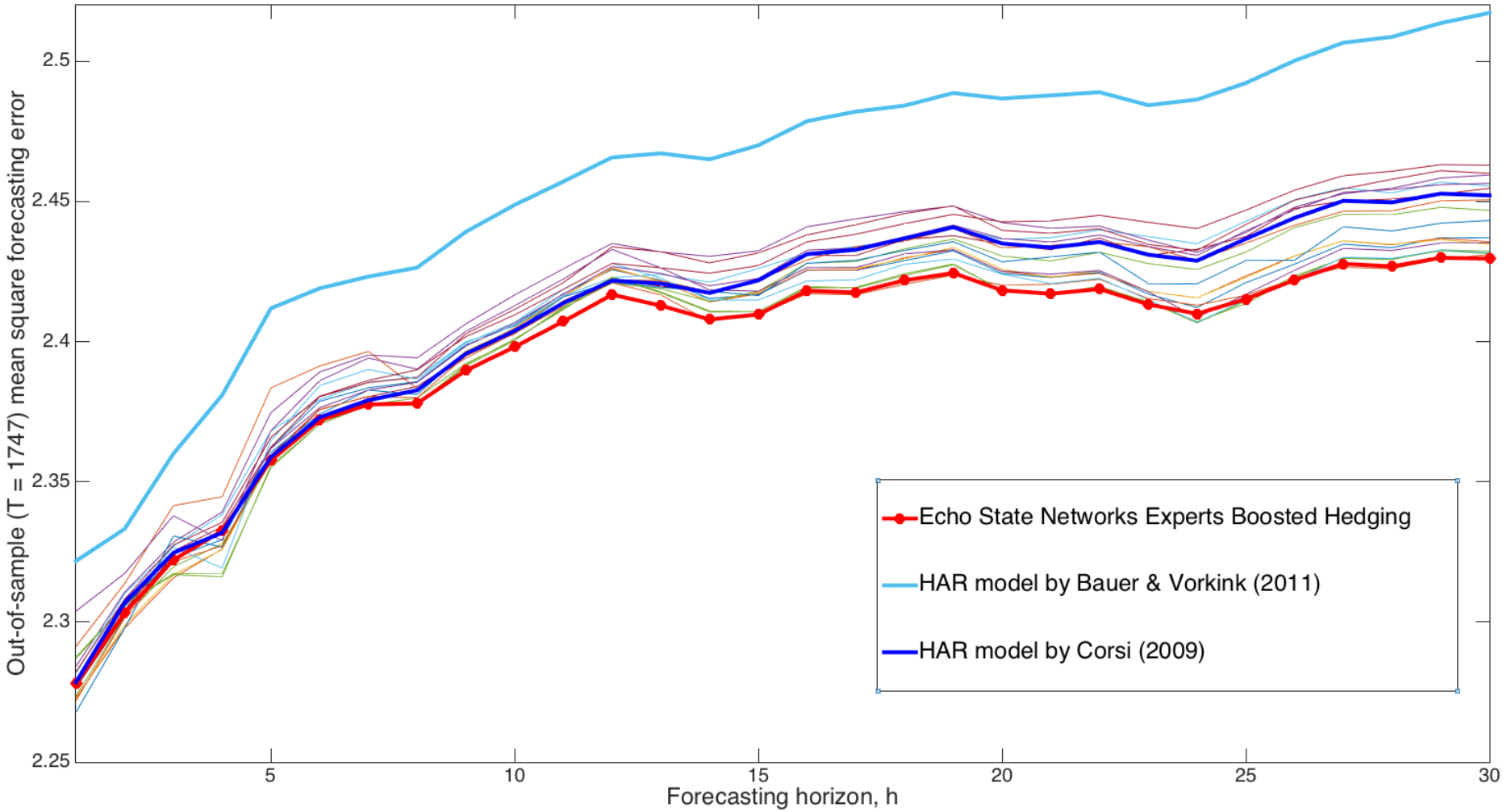
Forecasting S&P500 RV ($T_{est} = 3000$, $h = 1$)



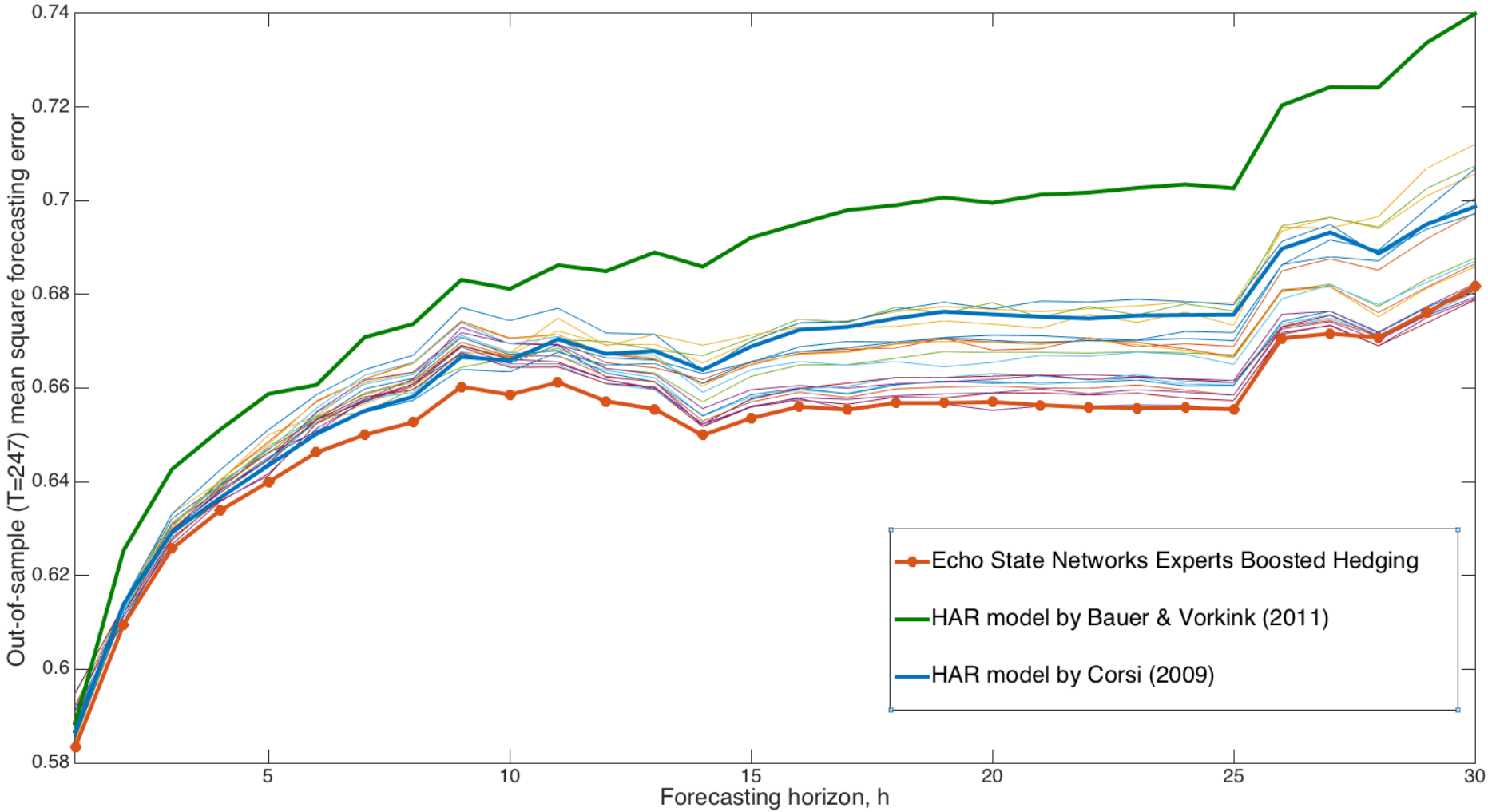
Forecasting S&P500 RV ($T_{est} = 3000$, $h = 30$)



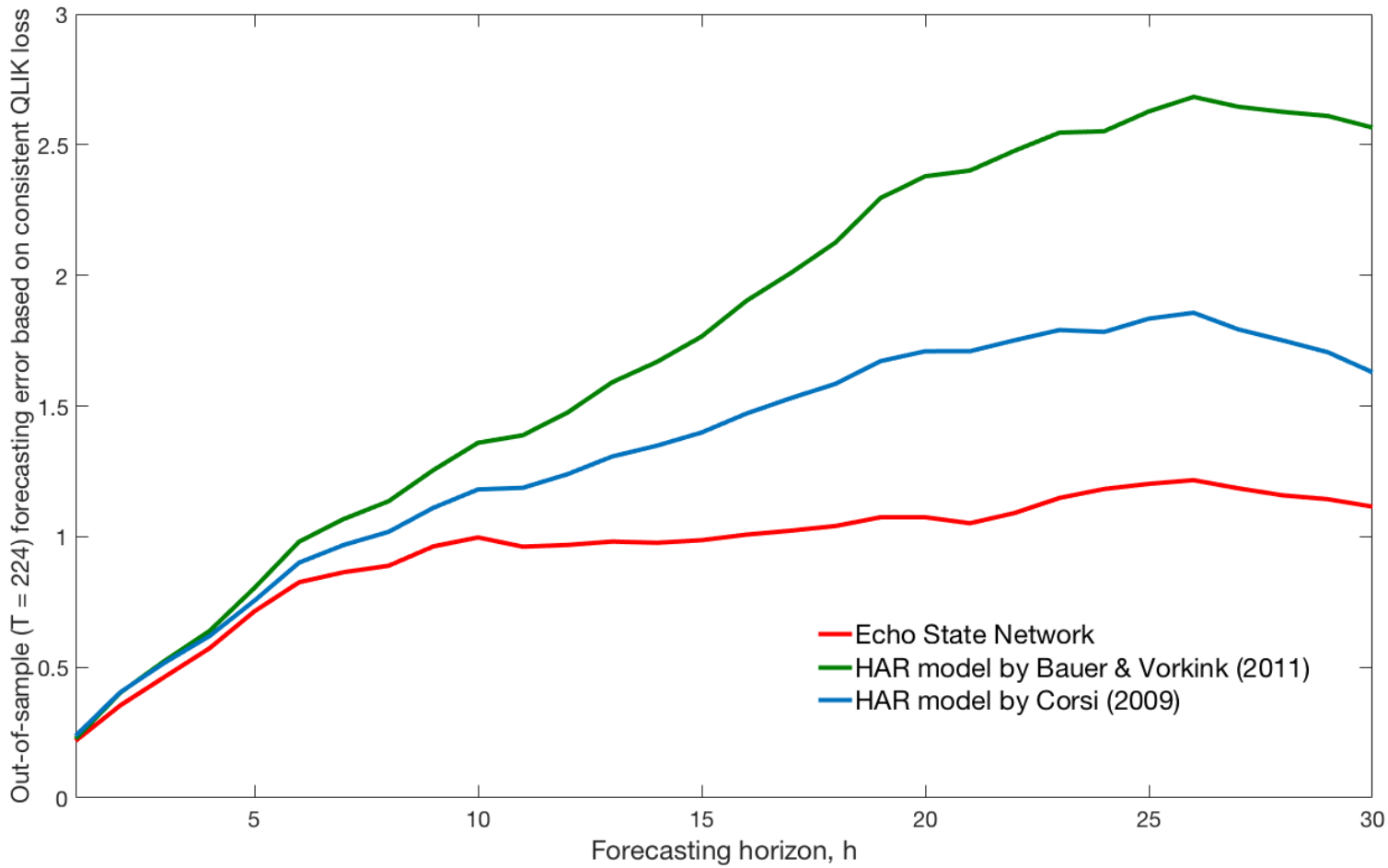
Forecasting of FTSE realized variance ($T_{est} = 3000$)



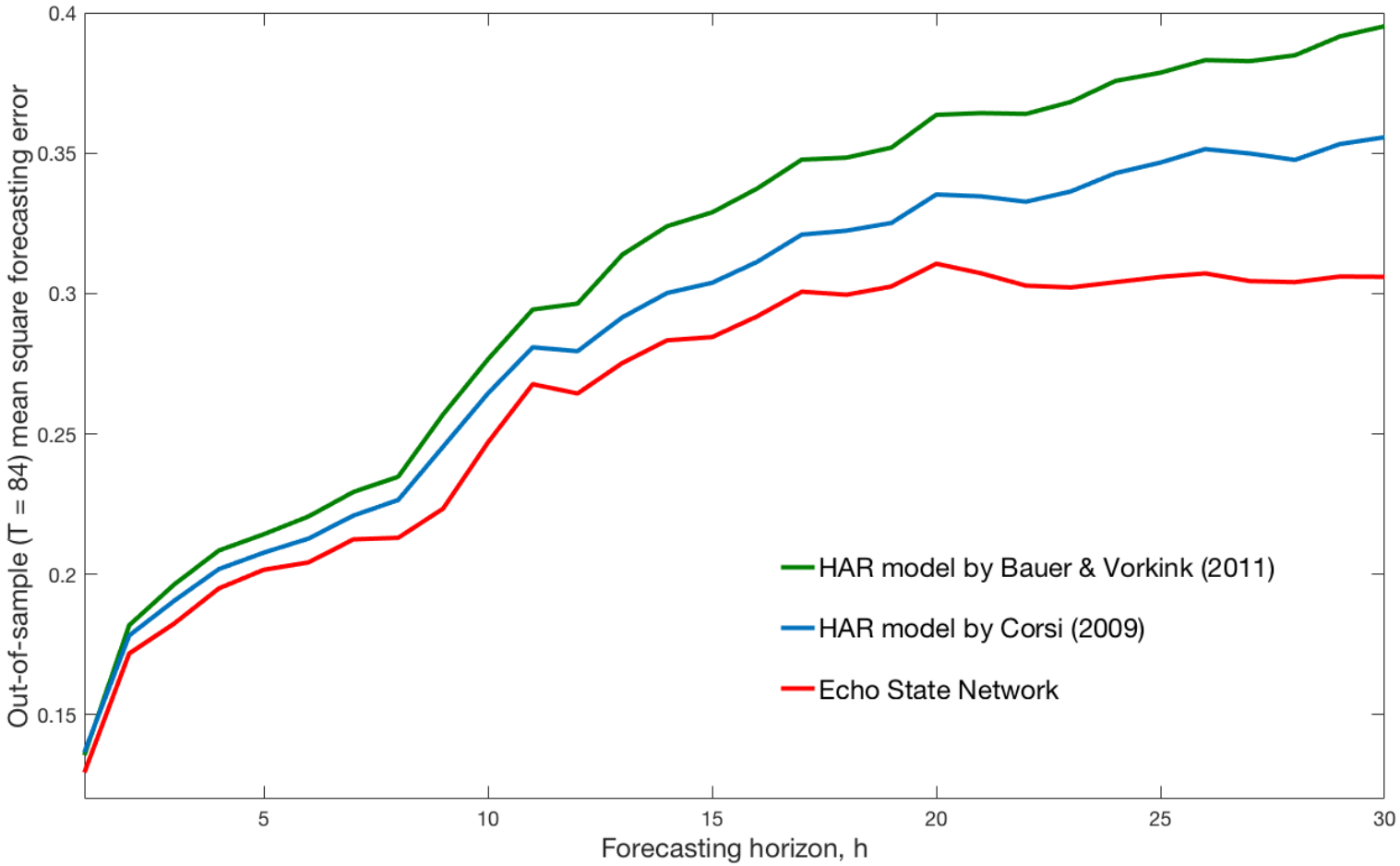
Forecasting of FTSE realized variance ($T_{est} = 4500$)



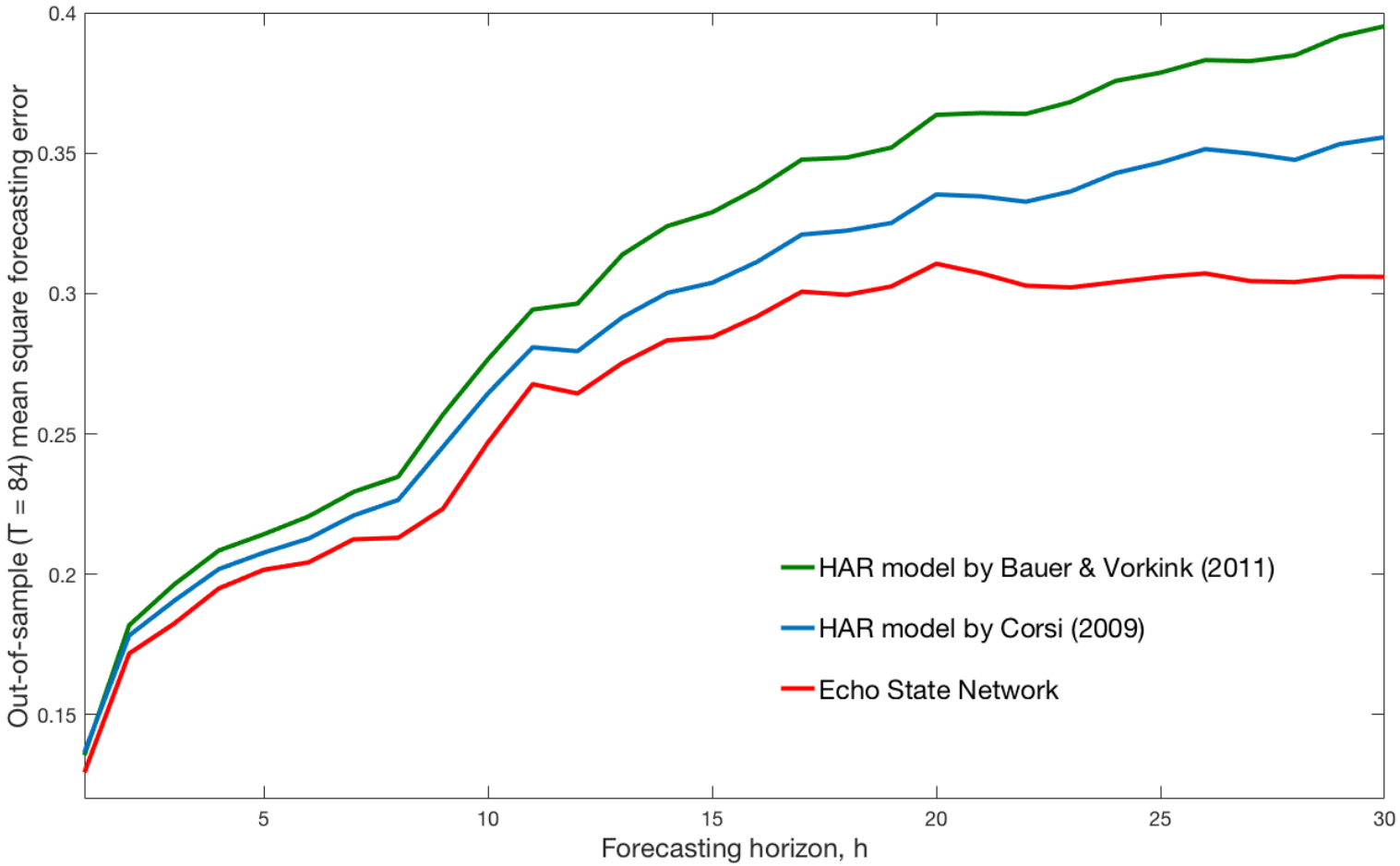
Forecasting of S&P500 realized variance ($T_{est} = 4500$)



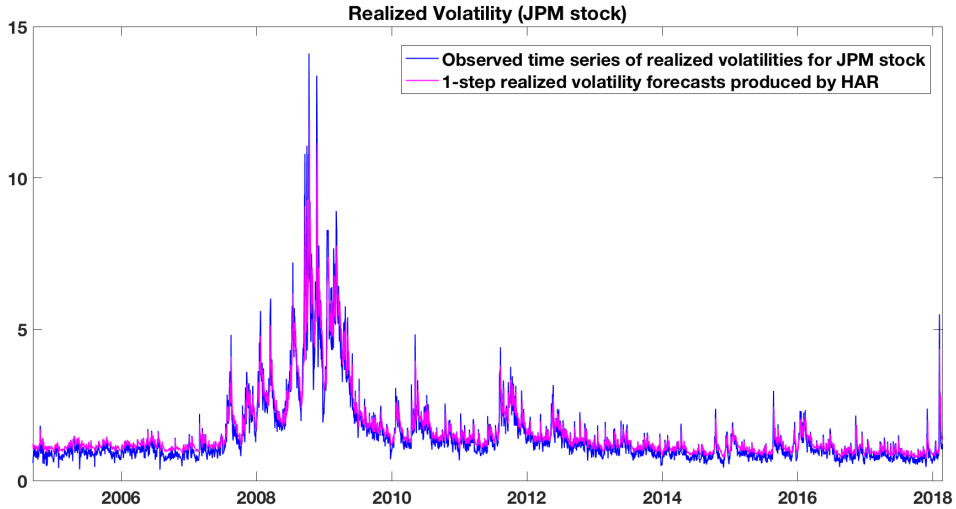
Forecasting of NIKKEI realized variance ($T_{est} = 4500$)

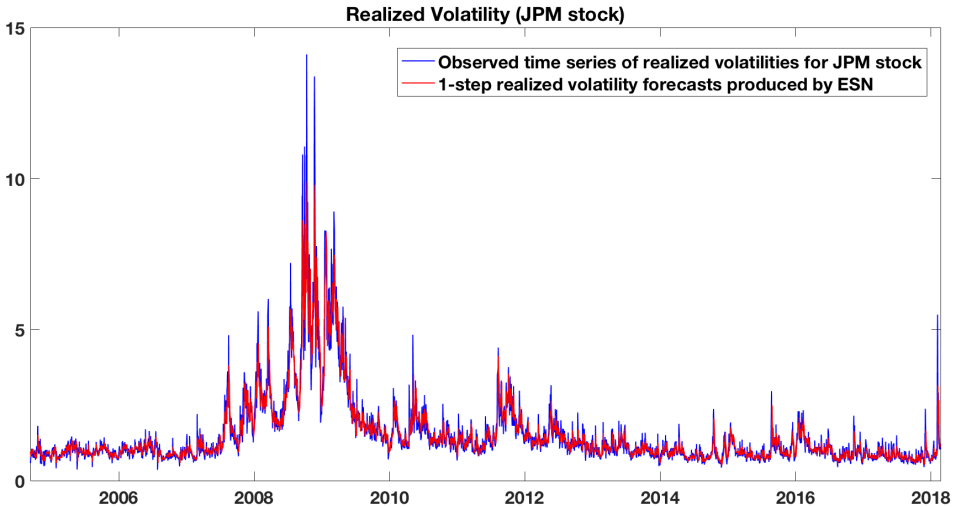


Forecasting of NIKKEI realized variance ($T_{est} = 4500$)

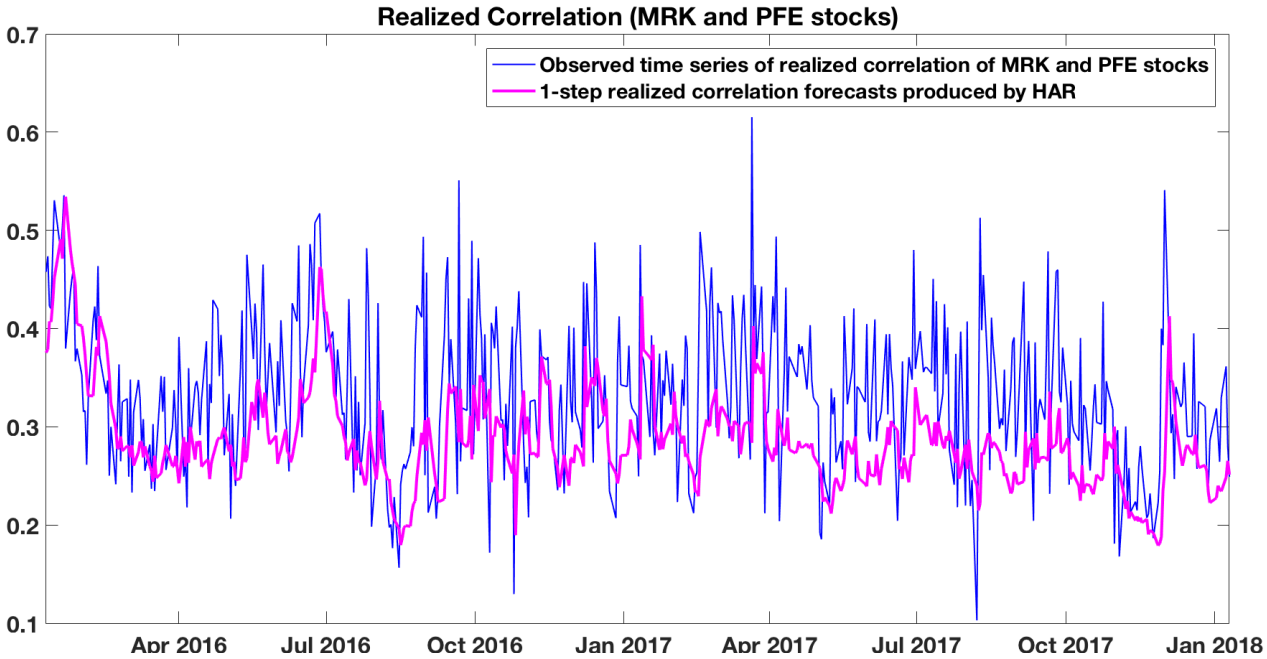


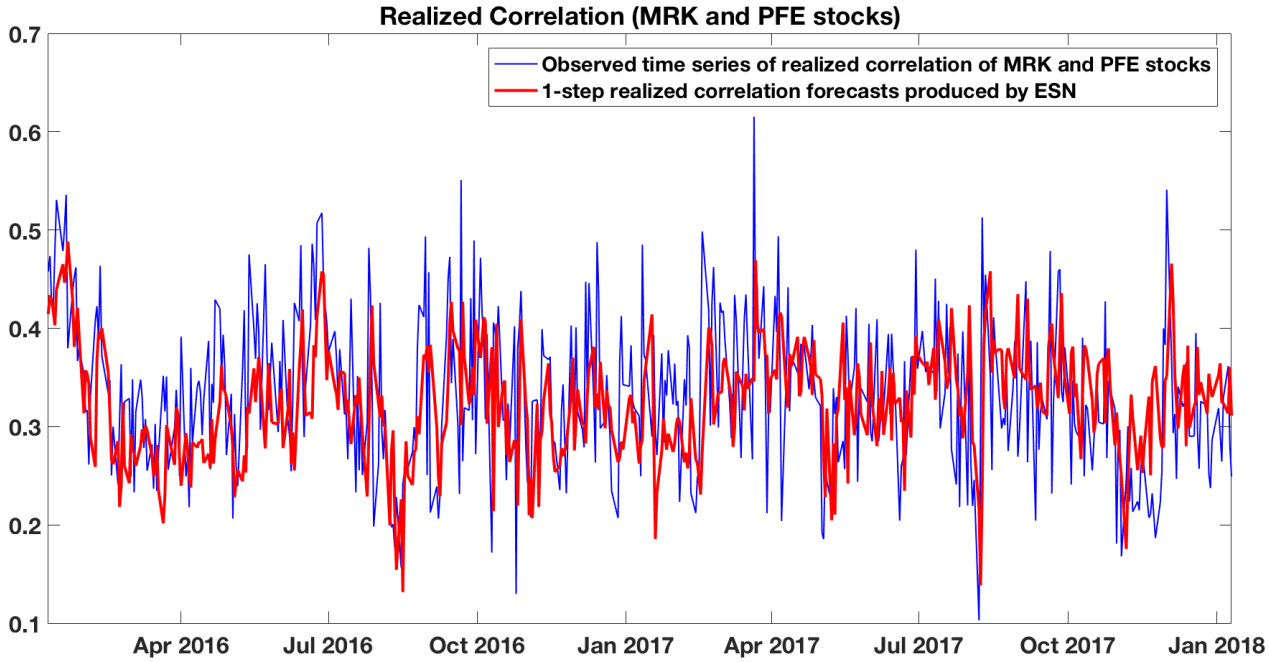
Realized correlation forecasting





Figures taken from the master thesis of Larissa Zimmermann.


Realized Correlation (MRK and PFE stocks)
Observed time series of realized correlation of MRK and PFE stocks
1-step realized correlation forecasts produced by HAR
0.7
0.6
0.5
0.4
0.3
0.2
0.1
Apr 2016
Jul 2016
Oct 2016
Jan 2017
Apr 2017
Jul 2017
Oct 2017
Jan 2018


Realized Correlation (MRK and PFE stocks)
Observed time series of realized correlation of MRK and PFE stocks
1-step realized correlation forecasts produced by ESN
0.7
0.6
0.5
0.4
0.3
0.2
0.1
Apr 2016
Jul 2016
Oct 2016
Jan 2017
Apr 2017
Jul 2017
Oct 2017
Jan 2018

Empirical results

- ESN outperforms state-of-the-art models for realized volatility in forecasting tasks
- ESN trained on a given training dataset of one given index is showed to perform well in forecasting of a number of other indices

References I

Francesco Audrino, Chen Huang, and Okhrin Ostap.

Flexible HAR model for realized volatility.
Studies in Nonlinear Dynamics and Econometrics, 23(3), 2019.

F. Audrino and S.D. Knaus.

Lassoing the HAR model: A model selection perspective on realized volatility dynamics.
Econometric Reviews, 35(8-10):1485–1521, 2016.

George E. P. Box and D. R. Cox.

An analysis of transformations (with discussion).
Journal of the Royal Statistical Society. Series B, 26:211–246, 1964.

Jan Beran.

Statistics for Long-Memory Processes.
CRC Press, 1994.

Nicolo Cesa-Bianchi and Gábor Lugosi.

Prediction, Learning, and Games.
Cambridge University Press, 2006.

F. Corsi.

A simple approximate long-memory model of realized volatility.
Journal of Financial Econometrics, 7(2):174–196, 2009.

Yoav Freund and Robert E. Schapire.

A decision-theoretic generalization of online learning and an application to boosting.
Journal of Computer and System Sciences, 55(1):119–139, 1997.

References II

Yoav Freund and Robert E. Schapire.

Adaptive game playing using multiplicative weights.
Games and Economic Behavior, 29:79–103, 1999.

Lukas Gonon, Lyudmila Grigoryeva, and Juan-Pablo Ortega.

Risk bounds for reservoir computing.
Preprint, 2019.

Sílvia Gonçalves and Nour Meddahi.

Box–Cox transforms for realized volatility.
Journal of Econometrics, 160:129–144, 2011.

Lukas Gonon and Juan-Pablo Ortega.

Reservoir computing universality with stochastic inputs.
IEEE Transactions on Neural Networks and Learning Systems, 31(1):100–112, 2020.

Sébastien Laurent, J. Rombouts, and Francesco Violante.

On loss functions and ranking forecasting performances of multivariate volatility models.
Journal of Econometrics, 173(1):1–10, 2013.

A. J. Patton.

Volatility forecast comparison using imperfect volatility proxies.
Journal of Econometrics, 160(1):246–256, 2011.

Tommaso Proietti and Helmut Lütkepohl.

Does the Box-Cox transformation help in forecasting macroeconomic time series?
2011.

References III

Efthymia Symitsi, Lazaros Symeonidis, Apostolos Kourtis, and Raphael Markellos.
Covariance forecasting in equity markets.
Journal of Banking & Finance2, 96:153–168, 2018.

Nick Taylor.
Realised variance forecasting under Box-Cox transformations.
International Journal of Forecasting, 33:770–785, 2017.