# EBC4257 Machine Learning - RNN-HAR

Enrico Wegner

June 11, 2022

**Abstract**

We derive and introduce the RNN-HAR model: a semi-parametric time-varying coefficient HAR model. Using a simulation study we show that the model can outperform the standard HAR model and can recover non-linear time variation. We apply the knowledge obtained in the simulation study to a real-data case of predicting the Realised Volatilty of the S&P500. We find good performance for the MAP estimate but difficulties with the model when estimating it using Bayes-By-Backprop (BBB) or Stochastic Gradient Nose-Hoover Thermostat (SGNHT-S). We then discuss from a retrospective perspective what to learn from the failure and discuss whether a paradigm shift might be needed.

## 1 Introduction

Realised Volatility plays an important role in fincance. This is due to its importance in portfolio allocation exercises and its influence in option pricing. As such, the ability to accurately model and predict realised volatility would greatly improve many financial practices. Reseach into how to model and predict RV has become an integral part of the financial academic literature. However, as Corsi et al., 2012 argue, RV is characterised by long-range dependence (long-memory), levarage effects, and jumps. All of these characteristics make modelling and predicting RV inherently difficult. First approaches involved ARFIMA models, which are long-memory processes but also difficult to estimate. As an alternative to the exact long-memory processes, Corsi, 2009 introduced the Heterogenous-Autoregressive (HAR) model of RV. Motivated by the heterogenous market hypothesis, stating that the market is driven by investors of different investing horizons, the HAR model can be interpreted as a cascading model in which lower frequency traders have influence on higher frequency traders and thus also on the volatility at higher frequencies. Corsi, 2009 showed that although the model is only approximately long-memory, it can model the long-range dependence of RV and accurately predict one period ahead.

The simplicity of the HAR model and its intuitive behavioural interpretation likely contributed to its success. The original HAR model was only able to model the long-range dependence of RV though, leaving the jumps and levarage-effects, as well as the changing variance of realised volatility unexplained. Thus, much of recent research has tried to extend the HAR model in ways that allow it to also model parts of the other characteristics. Corsi and Reno, 2009 extend the HAR model to both levarge and jump effects, and Fernandes et al., 2014 use the HAR model with exogenous variables (HARX) and add assymetry effects, yielding the AHARX model. Huang et al., 2016 introduce the HAR-GARCH model and thus model the conditional variance of realised volatility.

Although all of the above extensions to the HAR model usually came with improvement in preditive performance and left most of the intuitive interpretation of the original HAR model unchanged, they all rely on fixed coefficient estimates. Thus, all the models above assume that the dynamics of realised volatility do either not change over time, or are well approximated by a constant model. Liu and Maheu, 2008 showed that structural changes in realised volatility exist, and thus, the assumption of constant coeffcients might be violated.

Motivated by the evidence of structural breaks and the thus possible inefficiency of the HAR model has motivated some researchers to investigate whether allowing for time varying coefficients (TVC) and/or allowing for other non-linearities would improve the standard HAR model's preditive performance. Wang et al., 2016 study realised volatility forecasting of the S&P500 and allow for various forms of time variation, including a contiously changing time variation in coefficients, as well as a discrete markov switching model. They find that allowing for time variation results in economically meaningful results in the sense that it results in better portfolio allocations. Similar results are obtained by Chen et al., 2018. Fernandes et al., 2014 take another approach and instead of modelling time variation in coefficinets, model non-linearities by adding a neural network to the HAR model. Hamid and Iqbal, 2004 completely drop the HAR specification and instead use the NN and compare forecasts obtained using the NN model to implied volatility forecasts. They find the NN model outperforms. Similarly, Bucci, 2020 completely drops the HAR specification and instead goes for LSTM and NARX models. They also depart from most of the other research by focussing on monthly realised volatility and by including macro-economic variables. Lastly, Basturk et al., 2021 recently have explored another avenue by exploiting the realtionship between realised volatilities and values at risk. They use a joint LSTM network with outputs for both RV and VaR predictions and observe good performance.

Besides the efforts by Wang et al., 2016 and Chen et al., 2018, all the other works mentioned above suffer from a loss of interpretability. Bucci, 2020 even argues that their work focussed more on superior preditive performance than on interpretability and that a general problem with using NN models is the loss of interpretation. Indeed, as argued above, likely one of the contributing factors to the popularity of the HAR model is its intuitive interpretation. Although not a causal model, the simple structure makes it easy to communicate results and understand predictions and thus makes it easier for a financial analyst to judge whether the forecast can be trusted in the current state of the world. This latter point should not be forgotten: No model is perfect and it remains an integral part of a financial analyst's job to judge whether a model is appropriate. If a model is very complex and difficult to understand, a financial analyst can no-longer follow up on that responsibility and thus might be less prone to using the more complex model, dispite its likely superior preditive performance.

This paper tries to fill this gap by introducing a recurrent neural network (RNN) into the HAR model in a way that leaves large parts of the interpretation untouched. This will be done by using the RNN output as the coefficients of the HAR model. Thus, this paper could be interpreted in the HAR paradigm as modeling the changing behaviour of daily, weekly, and monthly horizon investors through a RNN model. Additionally, we will estimate the model using Bayesian methods. The most similar works to this is by Donfack and Dufays, 2021 who also use a neural network to model the time varying parameters. They focus on GARCH models though and use single- and multi-factor neural networks, which were not specifically designed for sequences. Using single- and multi-factor NN models, they are able to obtain global identification guarantees though, which will be impossible in the current work. We would argue though, that for the purposes of this work, global identification would make it easier to sample from the posterior, but are not needed for the actual analysis.

# 2 TVC-HAR Model

We can follow Corsi, 2009 by using the standard continous time process

$$dp(t) = \mu(t)dt + \sigma(t)dW(t) \tag{1}$$

where $p(t)$ is the instantaneous price, $\mu(t)$ is a Cadlag finite variation process, $W(t)$ is standard Brownian motion, and $\sigma(t)$ is a process independent of $W(t)$.

The integrated variance over day $t$ is then defined as

$$IV_t^{(d)} = \int_{t-1d}^{t} \sigma^2(\omega)d\omega$$

and we define the integrated volatility just like Corsi as the square roote of the integrated variance: $\sigma_t^{(d)} = \sqrt{(IV_t^{(d)})}$.

Since Andersen et al., 2001; Barndorff-Nielsen and Shephard, 2002a, 2002b it is known that the integrated volatility can be approximated by the sum of interday squared returns up to an arbitrary precision. This approximation is known as the Realised Volatility, and we will follow the standard and denote it by $RV_t^{(d)}$ for the daily realised volatility. Further following Corsi, 2009 we define the realised volatility over lower frequencies, weekly and monthly, as

$$RV_t^{(w)} = \frac{1}{5}(RV_t^{(d)} + RV_{t-1}^{(d)} + ... + RV_{t-4}^{(d)})$$

$$RV_t^{(m)} = \frac{1}{22}(RV_t^{(d)} + ... + RV_{t-21}^{(d)})$$

The standard HAR model (Corsi, 2009) is then motivated by defining the latent partial volatility $\tilde{\sigma}_t^{(.)}$ as the volatility generated by a certain market component, noting that $\sigma_t^{(d)} = \tilde{\sigma}_t^{(d)}$, and defining the the following cascading structure

$$\tilde{\sigma}_{t+1m}^{(m)} = c^{(m)} + \gamma^{(m)} RV_t^{(m)} + \tilde{\omega}_{t+1m}^{(m)}$$

$$\tilde{\sigma}_{t+1w}^{(w)} = c^{(w)} + \gamma^{(w)} RV_t^{(w)} + \mathbb{E}_t[\tilde{\sigma}_{t+1m}^{(m)}] + \tilde{\omega}_{t+1w}^{(w)}$$

$$\tilde{\sigma}_{t+1d}^{(d)} = c^{(d)} + \gamma^{(d)} RV_t^{(d)} + \mathbb{E}_t[\tilde{\sigma}_{t+1w}^{(w)}] + \tilde{\omega}_{t+1d}^{(d)}$$

Instead of using the above cascading structure, we will instead use a non-linear structure. For this , we will denote $v_{t-h:t} = [v_{t-h}, ..., v_t]'$. Then note that for any function $f(v_{t-h:t})$ we may use a first order Taylor approximation to obtain

$$f(v_{t-h:t}) \approx f([0, v_{t-h:t-1}]) + g([0, v_{t-h:t-1}])v_t = c_t + \gamma_t v_t \tag{2}$$

where $[0, v_{t-h:t-1}] = [0, v_{t-h}, ..., v_{t-1}]$, $c_t = f([0, v_{t-h:t-1}])$ and $\gamma_t = g([0, v_{t-h:t-1}]) = \nabla f_1|_{v=[0,v_{t-h:t-1}]}$, i.e. the first element of the gradient of $f$ evaluated at $[0, v_{t-h:t-1}]$. Using this, we can then define the cascading structure as

$$\tilde{\sigma}_{t+1m}^{(m)} = f^{(m)}(RV_{t-hm:t}^{(m)}) + \tilde{\omega}_{t+1m}^{(m)} \approx c_t^{(m)} + \gamma_t^{(m)} RV_t^{(m)} + \tilde{\omega}_{t+1m}^{(m)}$$

$$\begin{aligned}\tilde{\sigma}_{t+1w}^{(w)} &= f^{(w)}(RV_{t-hw:t}^{(w)}) + \mathbb{E}_t[\tilde{\sigma}_{t+1m}^{(m)}] + \tilde{\omega}_{t+1w}^{(w)} \\ &\approx c_t^{(w)} + \gamma_t^{(w)} RV_t^{(w)} + \mathbb{E}_t[\tilde{\sigma}_{t+1m}^{(m)}] + \tilde{\omega}_{t+1w}^{(w)}\end{aligned}$$

$$\begin{aligned}\tilde{\sigma}_{t+1d}^{(d)} &= f^{(d)}(RV_{t-hd:t}^{(d)}) + \mathbb{E}_t[\tilde{\sigma}_{t+1w}^{(w)}] + \tilde{\omega}_{t+1d}^{(d)} \\ &\approx c_t^{(d)} + \gamma_t^{(d)} RV_t^{(d)} + \mathbb{E}_t[\tilde{\sigma}_{t+1w}^{(w)}] + \tilde{\omega}_{t+1d}^{(d)}\end{aligned}$$

**Remark 1:** Note that we are still modeling the expectations about lower frequency volatilities additively. This could be generalised by putting it into the non-linear function and generalising the first order Taylor

approximation, which would then result in a time varying coefficient in front of the expectation terms. Since this coefficient would be mixed with other time varying coefficients later on, we decided to stick to the additive formulation.

**Remark 2:** $c_t^{(m)}$ and $\gamma_t^{(m)}$ are functions of $RV_{t-hm:t-1}^{(m)}$ and thus are functions of $RV_{t-hm-21d:t-1}^{(d)}$. Similarly, for the time varying coefficients of the weekly components. As such, all time varying coefficients are functions of $RV_{t-hm-21d:t-1}^{(d)}$.

Using this modified cascading structure and following Corsi, 2009 for the remainder of the HAR derivation results in the following time varying coefficient (TVC) HAR model:

$$RV_{t+1d}^{(d)} = c_t + \gamma_t^{(d)} RV_t^{(d)} + \gamma_t^{(w)} RV_t^{(w)} + \gamma_t^{(m)} RV_t^{(m)} + \omega_{t+1d} \tag{3}$$

As remarked above, $c_t$, $\gamma_t^{(.)}$ are all functions of $RV_{t-hm-21d:t-1}$ for some $h \in \mathbb{N}$. In this work, we will be modeling the time varying coefficients by a recurrent neural network with input sequences $RV_{t-hm-21d:t-1}$ as detailed in the next section.

**A Note on Interpretation:** We argued above that the current ways that Neural Networks have been used in HAR models for RV forecasting in general is unsatisfactory since the model loses its behavioural interpretation. Why then would this not be the case here? We argue that this is the case because we are leaving the basic HAR structure unchanged. We are essentially still deriving our dynamics from a cascade of investors' behaviours. The only difference that we are making is that these behaviours can change over time. So while in the standard HAR model $\gamma_t^{(m)} = \gamma^{(m)}$ can be interpreted as the sensitivity of daily Realised Volatility to monthly frequency traders, in the TVC-HAR model, these sensitivities are allowed to change such that we might have periods in which monthly frequency traders do not matter for daily Realised Volatilities but rather that in these periods RV is mostly driven by weekly and daily frequency trader. Think for example of GameStop. Our use of Neural Networks is thus only to model this change in importance. So while also our model will not easily be interpretable when it comes to what changes these sensitivities, the final prediction, given the output of the Neural Network is interpretable in the same sense as the standard HAR model.

# 3 RNN-HAR

Define $\eta_t = (c_t, \gamma_t^{(d)}, \gamma_t^{(w)}, \gamma_t^{(m)})'$ and $x_t = (1, RV_t^{(d)}, RV_t^{(w)}, RV_t^{(m)})'$. Then write (3) as

$$RV_{t+1d}^{(d)} = \eta_t' x_t + \omega_{t+1d} \tag{4}$$

We will use the following functional form for $\eta_t$

$$\begin{aligned} \eta_t &= W_\eta h_t + b_\eta \\ h_t &= \tanh(W_{\tilde{x}} \tilde{x}_{t-1} + W_h h_{t-1} + b_h) \end{aligned} \tag{5}$$

This represents a single layer Recurrent Unit with $\tilde{x}$ as input (discussed below), a tanh activation function and a linear output layer. The dimension of $h$ can be chosen freely. Higher dimensional $h$ will allow to estimate more complex functions but also increases the risk of overfitting and increases the computational cost. This can be generalised in various ways, including $M$ Dense layers with non-linear activation functions after the Recurrent layer, or even multiple Recurrent layers. (6) represents the former.

$$
\begin{aligned}
\eta_t &= W_\eta d_t^{(M)} + b_\eta \\
d_t^{(M)} &= \sigma(W_d^{(M)} d_t^{(M-1)} + b_d^{(M)}) \\
d_t^{(M-1)} &= \sigma(W_d^{(M-1)} d_t^{(M-2)} + b_d^{(M-1)}) \\
&\vdots \\
d_t^{(2)} &= \sigma(W_d^{(2)} d_t^{(1)} + b_d^{(2)}) \\
d_t^{(1)} &= \sigma(W_d^{(1)} h_t + b_d^{(1)}) \\
h_t &= \tanh(W_{\tilde{x}} \tilde{x}_{t-1} + W_h h_{t-1} + b_h)
\end{aligned} \tag{6}
$$

Although multi-layer neural networks do not seem to be used frequently in the finance literature, the machine learning literature has found large success using deeper networks, and thus it might be interesting to compare these two architectures. The success of deeper networks is usually attributed to the earlier layers learning very fine structures while the later layers learn larger structures. As such, one could imagine the earlier layers learning the high frequency changes in investors behaviour, while the later layers learn the low frequency changes. Whether deeper networks perform better is an empirical question though, and thus we will compare single layer structures as in (5) with multi-layer structures as in (6).

(4) together with (5) or (6) thus describes the RNN-HAR model used in the present research. What remains to be discussed is the implementation of this model. The next section will discuss Bayesian estimation of the RNN-HAR model, but before going there, we will still discuss the general estimation procedure.

There are two ways in which the RNN-HAR model could be implemented given that we are dealing with a single time series. The first is a so called sequence-to-sequence structure in which the whole time series is fed through the network and each time a data point is fed through a prediction is made. Although this seems natural at first, it comes with the disadvantage of essentially representing only a single training example (a single sequence) and thus standard batching techniques would not be possible. This would drastically increase the computational burden.

If we can make the assumption that observations $K$ time periods ago do not play a role for the prediction today, then the above can be improved by splitting the long time series in overlapping subsequences of length $K$. This assumption was already made, although not explicitly stated, when deriving the TVC-HAR. There we concluded that the time varying coefficients are only functions of $RV^{(d)}_{t-hm-21d:t-1d}$. Thus, the present research splits the original long realised volatility series into overlapping subsequences and thus follows the so called sequence-to-one methodology in which a (sub)sequence is fed through the network and an output is only obtained after the last element of the (sub)sequence has been fed through. This last output will then be used as the coefficients of the TVC-HAR model.

To be consistent with the derivations of the TVC-HAR model, if we use (4) to make a prediction of RV at $t + 1d$, with either of (5) or (6) as the networks providing the time varying coefficients, then the input (sub)sequences must be of the form

$$
\tilde{x}_t = (x'_{t-K} \cdots, x'_{t-1})' \tag{7}
$$

which directly follows from the first-order Taylor approximation around $x_t$.

# 4 Bayesian Estimation

This paper aims to estimate the model in a Bayesian manner for the following related reasons:

1. While point predictions are often easy to obtain and often show great performance, we would argue that they are insufficient for any financial and truly any other application. It is not enough to know that on average, given today's information, the volatility will be some value. The probability that this event will actually happen is zero, and thus a credible interval, say an 80% interval, is much more useful, since, if correctly specified, we at least know that 80% of the time, tomorrow's value will fall inside that interval. This is an event that actually can happen!
2. Additionally, the task of a financial analyst does not usually end when she obtained a prediction for tomorrow's value. Instead, given this prediction she must likely make a series of decisions, which eventually will lead to some feedback, i.e. the profit on a trade. The Bayesian paradigm, and especially the availability of a distribution over tomorrow's outcomes, is integral part of rigorous decision-making.

To be able to estimate the model in a Bayesian manner, we must complete it with a likelihood and priors on all parameters, including those of the RNN. The likelihood is naturally given by the HAR model, although two choices can be made. First, one could model the data using Gaussian disturbances, which is the choice made here. Second, one could use Student-t distributed disturbances. We will leave the latter for future research/thesis. The likelihood is thus of the form:

$$RV_{t+1d}^{(d)}|\eta_t, x_t \sim N(\eta_t' x_t, \sigma^2) \tag{8}$$

For the disturbance standard deviation, $\sigma$, we chose a standard $Gamma(2.0, 0.5)$ prior, which is depicted in figure 1 and was chosen because 95% of its density lies between 0.12 and 2.78 with a mean of 1.0 and a mode 0f 0.5. We believe that this represents roughly the range in which we would expect the variance to fall.

To finalise the model, a prior for the network parameters, $\theta_i$, is needed. For this work, we chose to put prior independent Gaussian priors on all network parameters, which corresponds to the standard weight decay penalisation.

$$\sigma \sim Gamma(2.0, 0.5) \tag{9}$$
$$\theta_i \sim N(0, 0.5) \tag{10}$$

While in more traditional settings the goal is usually to obtain good estimates of the parameters of a model, in the context here, the parameters, $\theta_i$, are not interpretable and not even identified since multiple parameterisations of the RNN lead to the same output. The variables of interest in this paper, are instead, next day's Realised Volatility, as well as the coefficients of the HAR part. Thus, we are interested in $p(RV_{t+1}^{(d)}, \eta_t|D)$ where $D$ denotes our observed data. Noting that the above is equivalent to

$$p(RV_{t+1}^{(d)}, \eta_t|D) = \int_\Theta p(RV_{t+1}^{(d)}, \eta_t|\theta, D)p(\theta|D)d\theta$$

we can use standard Monte Carlo techniques and first obtain draws of the network parameters, $\theta$, and use those to obtain $RV_{t+1}^{(d)}$ and $\eta_t$ and average the latter two over all draws of $\theta$. This essentially corresponds to having a large ensemble of models.

Two obtain draws of $\theta$, we will be using three common techniques:

1. **Adaptive Metropolis Hastings (AMH)**: To have a baseline to which to compare our results, we will be estimating a standard HAR(1, 5, 22) model. To sample from this model we will be using Adaptive Metropolis Hastings (AMH) (Haario et al., 2001). Although the use of Hamiltonian Monte Carlo (HMC) techniques is more common, we found that our baseline model is well estimated using AMH. AMH is implemented in *BFlux* and thus we can use the same library we are using to estimate the RNN-HAR model.

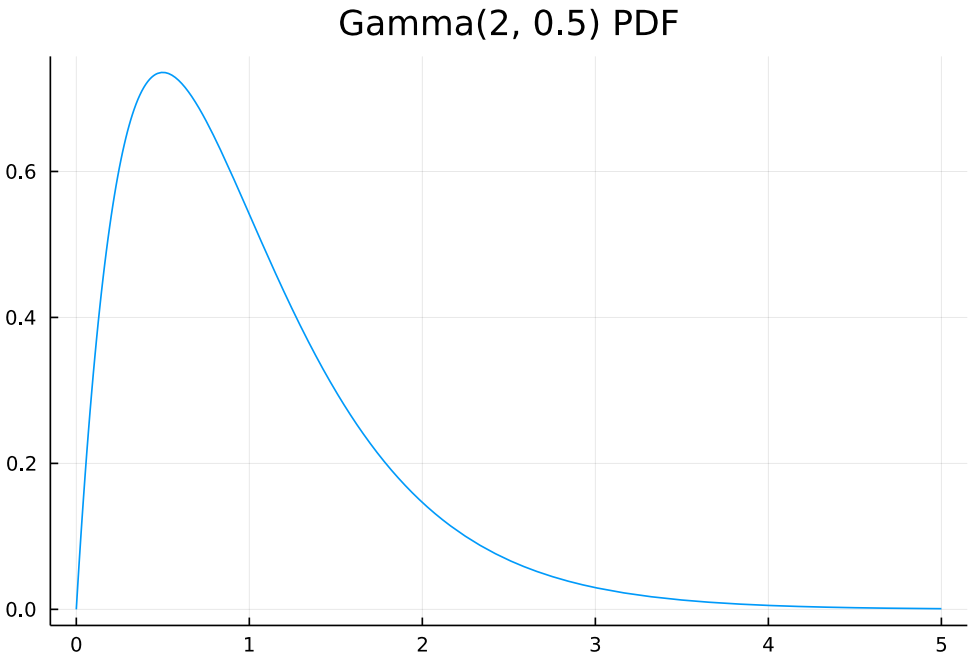



Figure 1

2. **Variational Inference via Bayes By Backprop (BBB)** (Blundell et al., 2015): We will be using the now common Bayes By Backprop, which is a variational technique and thus minimises the divergence between the posterior of $\theta$ and a variational family, which is chosen to be a multivariate Gaussian with off-diagonal elements restricted to be zero. This latter choice was made so as to keep the number of parameters low. BBB is implemented in *BFlux* and allows for using stochastic gradients, which will be made use off in this paper.

3. **Stochastic Gradient Nose-Hoover Thermostat (SGNT-S)** (Leimkuhler & Shang, 2016): SGNT-S was proposed as an alternative to the more common Stochastic Gradient Langevin Dynamics (SGLD) and Stochastic Gradient Markov Chain Monte Carlo (SGMCMC) (Nemeth & Fearnhead, 2019). Leimkuhler and Shang, 2016 argue that SGNHT-S performs better than the other methods by keeping the temperature and mean kinetic energy in the dynamic system stable (thermostat part). SGNHT-S represents the MCMC estimation family but does not implement a Metropolis-Hastings correction step. As such SGNHT-S, just like many of the other methods introduced in recent years (Nemeth & Fearnhead, 2019) is difficult to monitor, the exception being Garriga-Alonso and Fortuin, 2021. We will later on argue that this might not be as big of a problem as it may first seem.

# 5 Simulated Data

To confirm that RNN-HAR would be able to discover non-linear time-variation in parameters we simulated data with time varying parameters. The simulated model is closely related to a model with three distinct states, however, instead of jumping discretely between states, we use smooth transitions. This was achieved by defining two transition functions

$$t_1(x) = \text{sigmoid}(19x + 9) = \frac{1}{1 + e^{-19x-9}} \tag{11}$$

$$t_2(x) = \text{sigmoid}(9x - 3) = \frac{1}{1 + e^{-9x+3}} \tag{12}$$

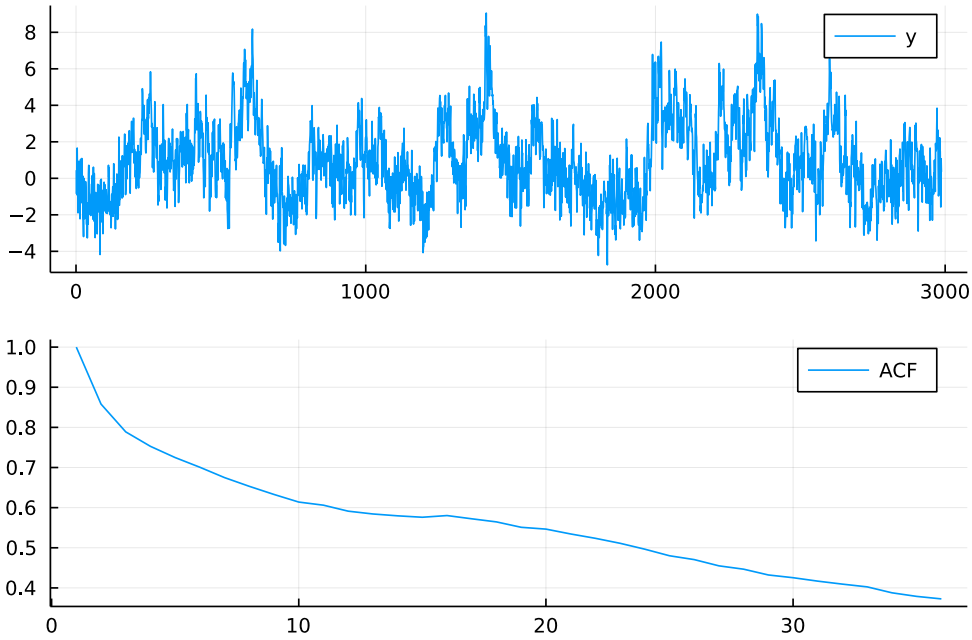



Figure 2: Simulated Data with smooth transitions between three states.

and then defining the coefficients in each period as

$$s_1 = (0.3, 0.2, 0.4)' \tag{13}$$
$$s_2 = (0.8, 0.1, 0.05)' \tag{14}$$
$$s_3 = (0.05, 0.6, 0.2)' \tag{15}$$
$$z_1 = t_1(RV_{t-2}^{(m)}) \tag{16}$$
$$z_2 = t_2(RV_{t-2}^{(m)}) \tag{17}$$
$$\beta_t = (1 - z_1)s_1 + z_1(z_2 s_2 + (1 - z_2)s_3) \tag{18}$$

Next periods simulated outcome was then defined as

$$RV_t^{(d)} = (RV_{t-1}^{(d)}, RV_{t-1}^{(w)}, RV_{t-1}^{(m)})\beta_t + \epsilon \tag{19}$$
$$\epsilon \sim Normal(0, 1) \tag{20}$$

Simulating data in this way resulted in non-linear time variation in the parameters that depends on past observables. As figure 2 shows it also resulted in data looking similar to the real data with a slowly decaying dependence.

**OLS Estimation**

Since we are able to obtain good predictions using the standard HAR model for real world problems, we checked whether the same is the case for our simulated data which clearly has time varying coefficients. We indeed found that this is the case by estimating the standard HAR model using OLS. Figure 3 shows the good visual match between the OLS predictions and the actual data. The Root Mean Squared Error (RMSE) of one day ahead predictions was 1.0470. It must be emphasised though that this is in-sample data. No test data was simulated as the purpose of this exercise was not to judge the performance but rather to see

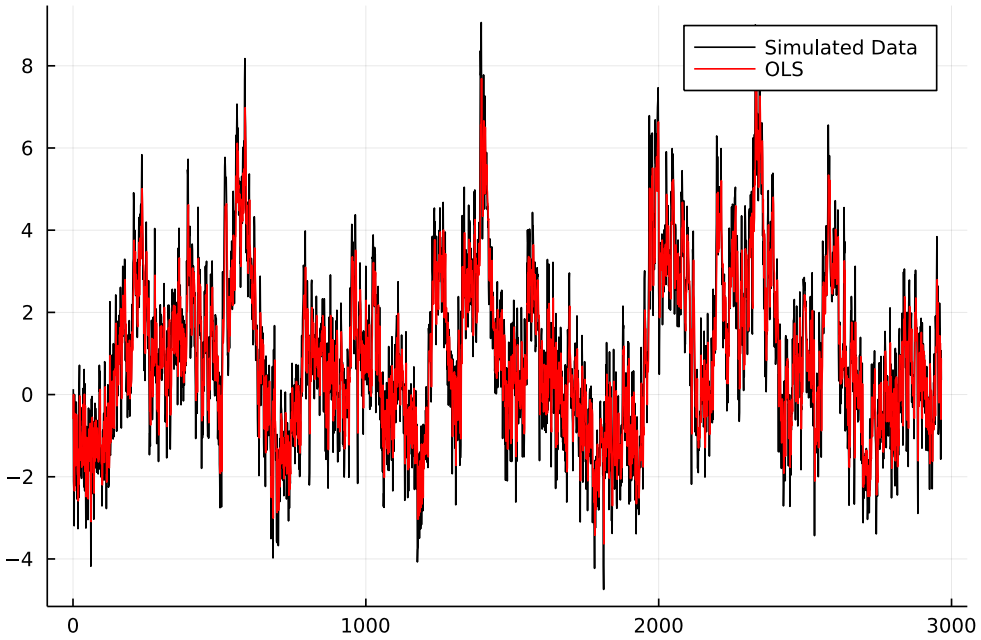



Figure 3: Simulated data and one-day ahead predictions using standard OLS estimates of the HAR(1, 5, 22) model

whether we would be able to uncover the non-linear time-variation and to gain some insights into the model which will hopefully inform some of our design choices in the experiments with real data.

**MAP Estimation**

The RNN part of the RNN-HAR model was chosen as a first RNN layer with three inputs and a hidden state of size three and a linear output layer which takes as input the RNN output and has four outputs corresponding to the intercept, daily coefficient, weekly coefficient, and monthly coefficient. In *BFlux* syntax this would be *Chain(RNN(3, 3), Dense(3, 4))*. To check whether this model is at all able to uncover the time variation, we first used the MAP estimate of the model. The MAP is similar to Maximum Likelihood in the sense that it is the Maximum A-Posteriori estimate: The maximum of the posterior distribution.

Predictions showed overall a good match between the simulated data and the one period ahead forecasts of the model using the MAP estimate. The in-sample RMSE is 1.0145 which is better than OLS, although it might not be economically meaningful in real-world applications.

Figure 4 shows the coefficient estimates as outputted by the RNN part of the RNN-HAR model. Clearly the model is able to capture the time-variation, although this is far from perfect. We think this could be due to

1. The daily, weekly, and monthly Realised Volatility variables as simulated here are highly correlated since the weekly and monthly RV are nothing else than running averages of the daily RV. This might lead to high collinearity problems, and thus we might be able to explain the data just as well by loading up on, i.e. daily and monthly and decreasing the weekly coefficient. Thus, the high collinearity might make it too difficult to uncover the true time variation in a finite dataset.

2. When looking at the time variation in the coefficient, we retrospectively must say that to the human eye there seem to be mostly two states with the middle one being hardly recognisable. If a state is not regularly visited or not clearly differentiable from other states, then in any finite dataset it might not be feasible to identify all states. A similar argument would apply in our smooth transition model. Time variation might not be fully uncoverable if there are periods in which the time variation is small.

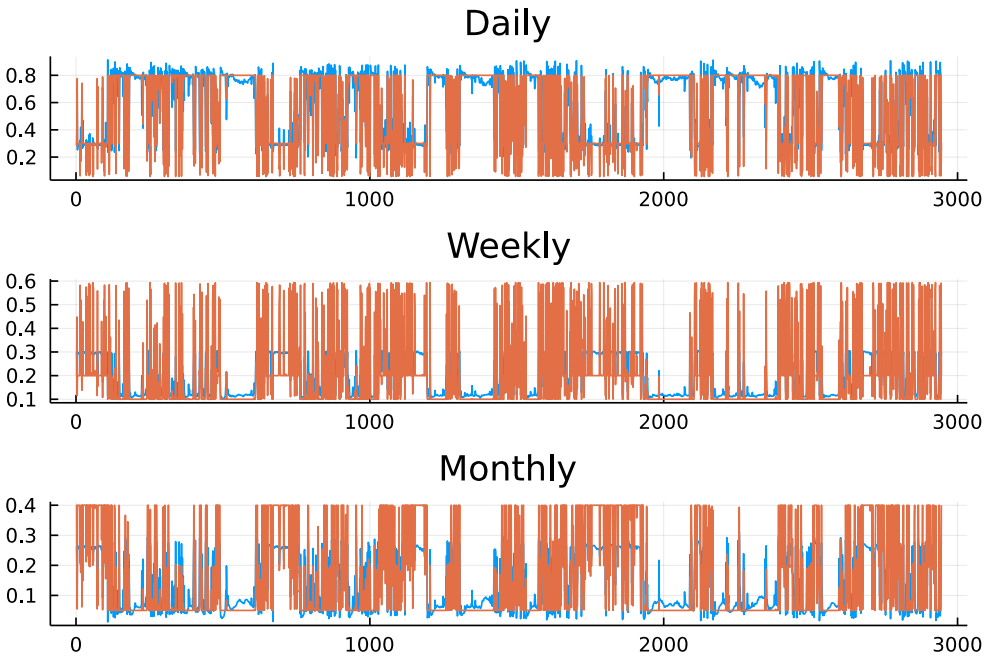



Figure 4: Coefficients of HAR model over time. Only the MAP estimate was used for the RNN model producing the coefficients. **Orange** shows the true coefficient at time $t$ while **Blue** shows the recovered coefficient.

The above results are a warning for real-data applications. If the time-variation we are trying to uncover is small or the variables are highly collinear, we might not be able to uncover the time-variation or might not correctly uncover it. While this would be a downfall for any situation in which the time variation itself is the element of interest, we believe this is less of a problem in the current research, since although we are allowing for time variation and hypothesise it is present, we are inherently only interested in good interval predictions; The time variation is only of second order interest.

**Bayes By Backprop (BBB)**

We next checked the model performance using BBB. We again observed a good visual match between the simulated data and the one period ahead forecasts. These forecasts were made by first obtaining draws of the network coefficients from the variational posterior, then using these to obtain posterior predictive draws for each period and averaging over these ten-thousand posterior predictive draws. As such, this would be comparable to having an ensemble of ten-thousand networks.

The RMSE of the in-sample predictions is 1.0153 and thus worse than the MAP estimates. Although not shown here, BBB had the same problems as MAP for the time varying coefficients and no clear difference between the recovered coefficients of BBB and MAP could be found.

Contrary to the MAP estimate, we can also obtain credible intervals and can thus make predictions that have a non-zero probability of occurring and that can be used in more extensive decision-making algorithms. Figure 5 shows a quantile-quantile plot. These plots were created by first simulating posterior predictive draws, then taking for each time period the qth quantile and checking how many percent of the actual data fall below this quantile. If the model was well calibrated/would explain the data well, then we would expect the line marked as theoretical - q% fall below the qth quantile. Although only done on in-sample data for the simulations, it shows a very good match between posterior predictive draws and the actual data. Thus, the model could be used to make accurate interval forecasts despite not being able to correctly uncover all the time variation.

**SGNHT-S**

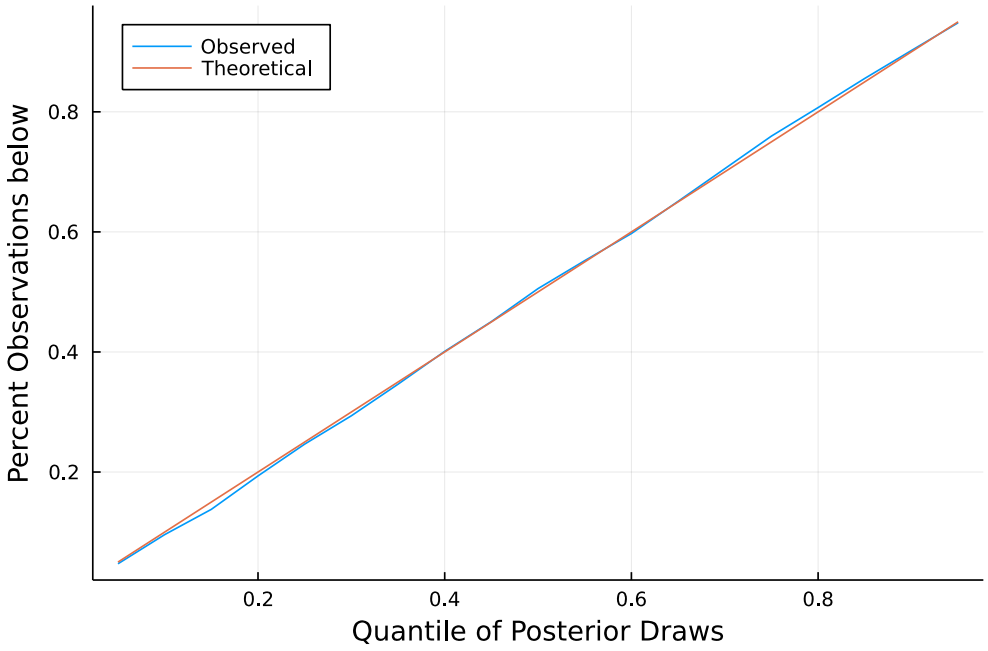



Figure 5: Quantile-Quantile plot for posterior predictive values obtained using BBB on the simulated data. The plot was obtained by first drawing 10K posterior predictive draws and then obtain the qth quantile of these draws. We then measured how many percent of the actual data fall below this quantile and plotted the target rate on the x-axis and the actual percent below the quantile on the y-axis. The theoretical line shows what we would like to see.

During our experiments with SGNHTS and the simulated data we noticed that performance is very sensitive to the hyperparameter settings. While it is relatively simple to find parameter settings for which $\sigma$ mixes well, finding parameter settings for which the output of the RNN also mixes well was a lot more difficult and was not completely achieved. Indeed, the maximum rhat of all coefficients across all time periods was 1.21 after sampling one-hundred-thousand draws and keeping the last fifty-thousand. This is far from perfect (a value of under 1.01 is often aimed at) and thus the chains should be considered carefully.

Additionally, SGNHTS was introduced in order to regulate the temperature in the system, which is closely linked to the average kinetic energy in the system (Leimkuhler & Shang, 2016). In the ideal case, this would equal one or be close to one. In all the chains that we found mixed well, the average kinetic energy was rather of the order of 1e-2 and thus far from the ideal case. As such, it is unclear how well the samples obtained from SGNHTS truly reflect posterior samples.

Note that the network parameters and how well they mixed was not mentioned thus far. That is because mixing in network parameters is pretty much a hopeless goal. Not just is the posterior surface much to complex to sample from - just as the loss surface of common neural network estimation comes with many topological problems (Garipov et al., 2018) - but the network parameters are also not uniquely identified and thus many equivalent parameterisations exist, leading to many modes.

We often found chains that looked like there were slowly walking between modes. Garipov et al., 2018 argue that in the loss surfaces of regular NN, channels between local optima exist along which the loss does hardly change. If these also exist in our case, then the chain could possibly just walk along these channels making it very difficult to obtain good mixing within and across chains. These problems are likely to be even worse in RNN model than in standard Feedforward Neural Networks, due to the non-linear recurrence and thus the highly non-linear relationship between today's value and the value 22 days ahead.

Since we are not interested in the network parameters itself, we focus on the coefficient output and the predictive values and the mixing therein. This will also be done in the real world application

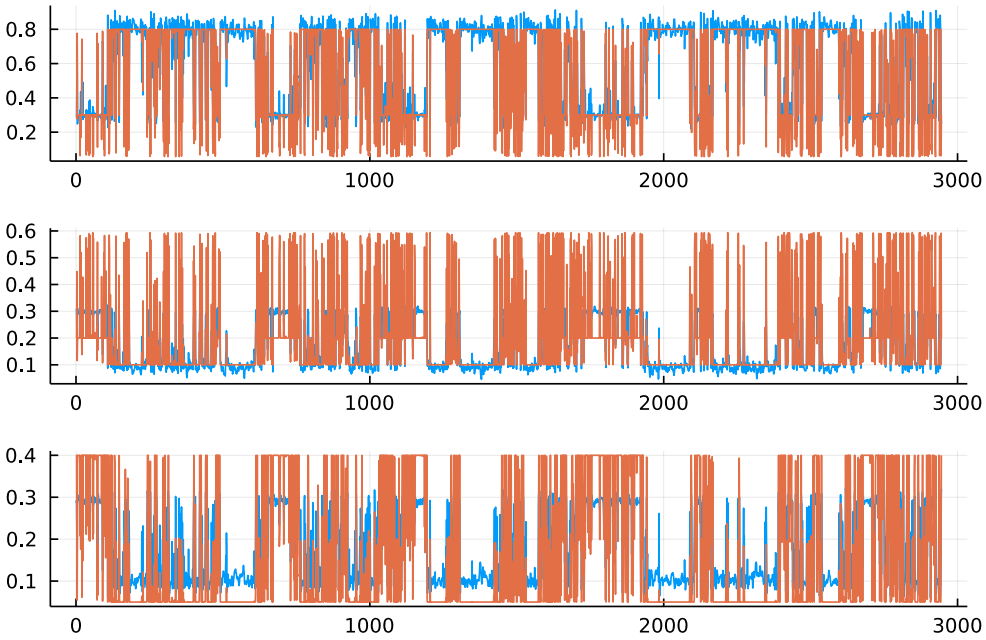



Figure 6: Time variation of coefficients for the simulated data obtained using SGNHT-S

Figure 6 shows the uncovered time variation in the parameters and figure 7 shows the quantile-quantile plots. Both look very similar to the previous plots for MAP and BBB. Despite the bad mixing in the coefficient output and the far from optimal mean kinetic energy, the sampler still seems to have resulted in what we would rather term useful samples than calling them accurate samples, which we cannot truly evaluate.

The RMSE of in-sample predictions is 1.0133 and thus better than all other estimation methods despite the problems mentioned above. The quantile-quantile plot, figure 7 also shows a close match and thus the model estimated using SGNHT-S would provide good interval estimates. Thus, despite the bad mixing and low Effective Sample Size of the chain obtained using SGNHT-S, predictive performance is good in both a point and interval sense.

**What to take away from the simulations?**

The simulations were meant to inform us about choices we will need to make in the real data applications. As such, we think the following points can be taken away from the simulations

1. Even when clear time variation exist the method does not completely manage to uncover this. As such, if the underlying time variation in the real data is minimal, we might not be able to uncover it, even if it was truly present. But if the time variation is only minimal, then not being able to uncover it might not even be a problem.

2. Obtaining MAP and BBB estimates is significantly easier than obtaining good chains using any of the MCMC samplers introduced in recent years and implemented in *BFlux*.

3. Even if the samplers result in chains that in the strict Bayesian sense are having too bad diagnostic statistics to be accurate samples from the posterior, they might still be useful and result in good predictions in both point and interval sense. Care should be taken though to not misinterpret these chains though. Instead of interpreting them as samples from a posterior, which they are very unlikely to be, they should rather be interpreted as an ensemble model, similar to Random Forest Regressions, in which the underlying probabilistic model gives structure to the kind of ensemble members we are after. Using this ensemble view, we can still make point and interval predictions. In the simulated case here, the ensemble obtained using SGNHTS resulted in accurate point and interval predictions.

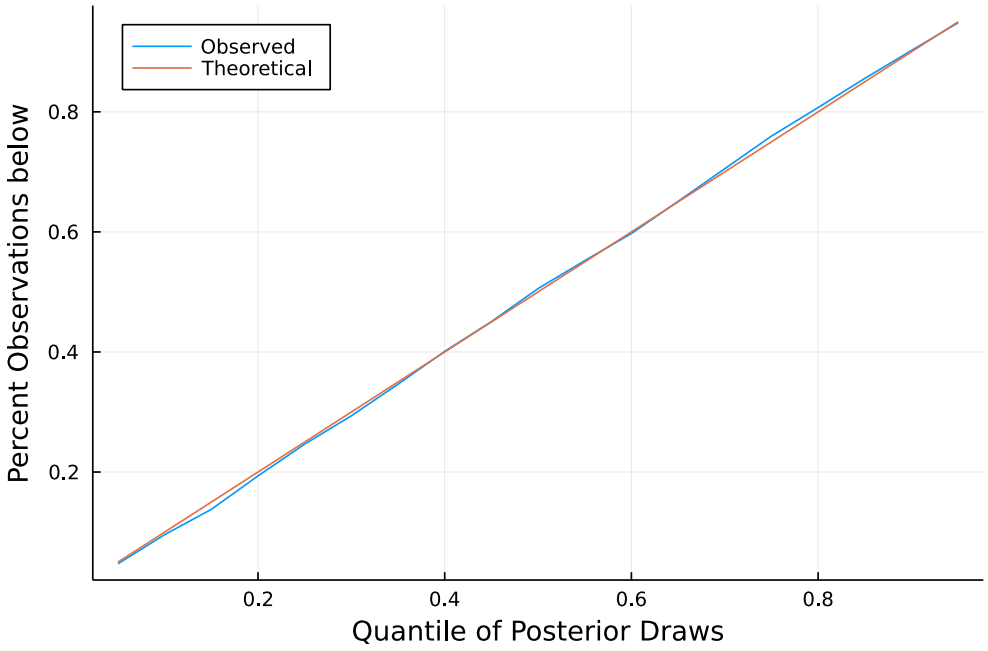



Figure 7: Quantile-Quantile plots as explainded above for the simulated data. Obtained using SGNHT-S

# 6 Real Data Application

## 6.1 Data

The data used is taken from Oxford-man Institute's Realized Library (Heber et al., 2009), which provides various realised measures for 31 assets. For this research we will only focus on the log of Realised Volatility which can be obtained as the log of the square root of the Realised Variance provided by Heber et al., 2009. Additionally, we will only focus on RV obtained with 5min returns, while the dataset also provides RV obtained using 10min returns.

The standard measure of RV used is sensitive to market microstructure noise as pointed out in Hansen and Lunde, 2006 and various more robust alternatives have been proposed, among other the Realised Kernel Variance (Barndorff-Nielsen, Hansen, et al., 2008). Additionally, downside risk might be of interest to investors and researchers. This is captured by the Realised Semi-variance proposed in Barndorff-Nielsen, Kinnebrock, et al., 2008. Both, Realised Semi-variance and Realised Kernel Variance are provided in the dataset but will be left out of this research due to the desire of keeping the exposition simple.

We will further narrow down the data considered to only SPX which track the S&P500. Figures 8 and 9 show some of the characteristics of the data. Figure 8 shows the time series of the log Realised Volatility for SPX. As expected, the weekly and monthly frequencies are just smoothed (via running means) versions of the daily series. Figure 9 shows the typical persistency for SPX's Realised Volatility. Noteworthy is, that the PACF quickly declines. Although not explicitly displayed here, we estimated a HAR model based on the SPX data and simulated HAR models using those estimated coefficients. These simulations were in line with the slowly decreasing ACF but sudden drop in PACF and thus the ACF and PACFs of the SPX series seem to be visually consistent with a HAR model.

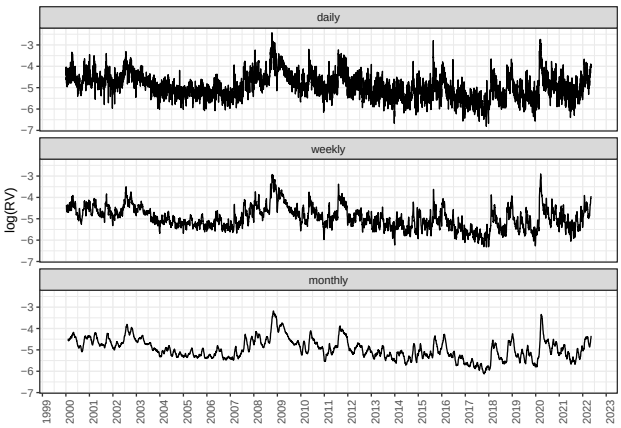



Figure 8: Log of RV at Different Frequencies

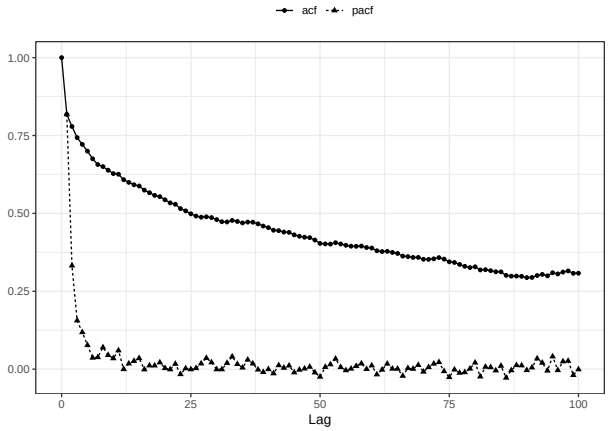



Figure 9: ACF and PACF of SPX RV series

| Method | Train | Validation | Test |
|---|---|---|---|
| HAR [AMH] | 0.5579 | 0.6740 | 0.6608 |
| RNN-HAR [MAP] | 0.5550 | 0.6695 | 0.6538 |
| RNN-HAR [BBB] | 0.5576 | 0.6742 | 0.6610 |
| RNN-HAR [SGNHTS] | 0.5573 | 0.6768 | 0.6638 |

Table 1: Point prediction performance measured using RMSE. First term denotes the model while the term in square brackets denoted the estimation method.

## 6.2 Baseline HAR(1, 5, 22)

To have a baseline to which to compare the RNN-HAR model when estimated using different methodologies, we first estimated a standard HAR(1, 5, 22) model in a Bayesian way. The likelihood is the same as in 8 except that $\eta$ is now time invariant. Thus, the Bayesian HAR model is just a standard Bayesian linear regression. The prior for $\sigma$ was still chosen to be a $Gamma(2, 0.5)$ for the same reasons mentioned above. As a prior for the time-invariant coefficients we chose $Normal(0, 0.5)$ because past estimations of HAR models all put the coefficients somewhere between -1 and 1.

Although the HAR model specified in the way above is conjugate and thus closed form solutions for the posterior exist, we chose to obtain samples using MCMC methods to be consistent with the other models. We chose to first find the MAP of the posterior via Stochastic Gradient Descent (SGD) and then start sampling from there using Adaptive Metropolis-Hastings (AMH) as introduced in an earlier section.

Four chains were run in parallel. Since the model is identified, we would expect all chains to look similar if they have converged. Figure 10 shows that this is the case for both the trace and the density plots. Additionally, no problems were observed in the chain diagnostic statistics and the mean acceptance rate of all chains was around 29%, a value common of Metropolis-Hastings chains.

The method of running multiple chains in parallel and then comparing them to judge the convergence to the posterior distribution would not make much sense in the RNN-HAR case since the model is not identified and multimodel and comes with many other problems as previously discussed. As such, it is highly unlikely that we would ever obtain four chains that perfectly mix across chains - it is usually already problematic to obtain chains that mix within.

Table 1 reports the train, validation, and test RMSE values for one-day-ahead predictions using the standard HAR model. Since nothing needed to be tuned in this baseline setting, the validation data was never touched and could thus also be seen as part of the test data.

Figure 11 shows the quantile-quantile plots for the training, validation and test data. In all cases interval predictions perform well, although both the validation and test data show too light left tails. More actual data falls below quantiles than should be the case if the model was accurate.

Overall, just like was found many times in the literature, we also find that the standard HAR model performs well and additionally to good point estimates also produces good interval predictions.

## 6.3 Model Specification and MAP Estimates

To not have to estimate multiple models using costly MCMC or Variational Inference methods, we first estimated a set of proposed network structures using MAP. The proposed structures were chosen rather arbitrarily since not much connection could be made between the network structure and the final output. All proposed structures are listed in table 2

We found no big differences between the validation and training performance of of these networks and thus chose to go with the small model (bold in the table) which in *BFlux* syntax is *Chain(RNN(3, 3), Dense(3, 4))*.

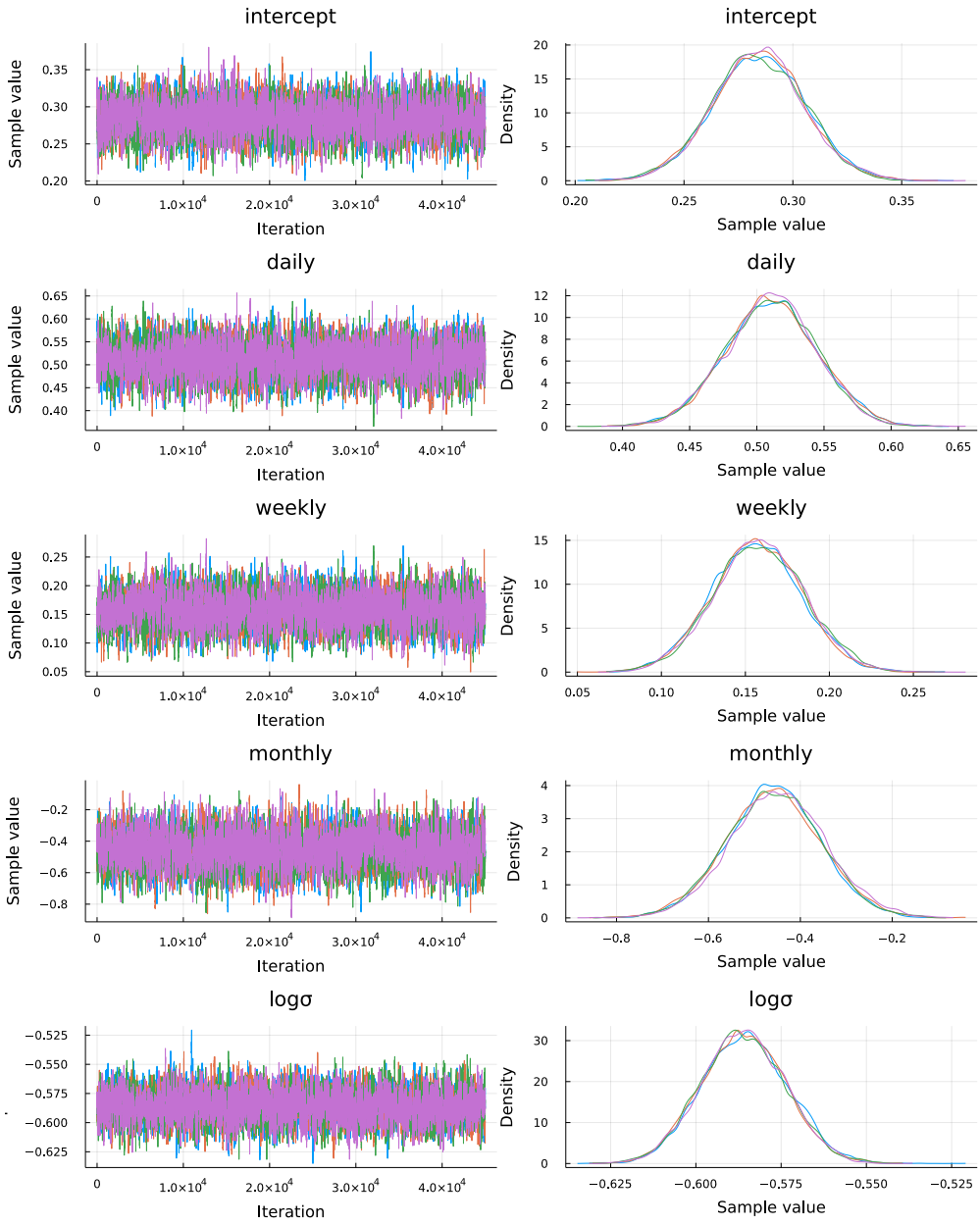
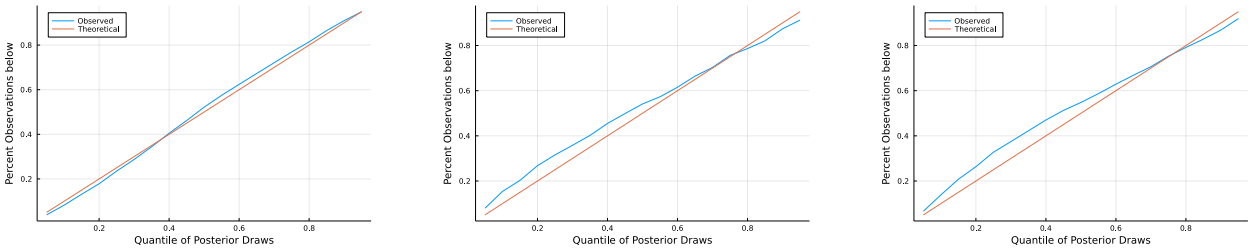



Figure 10: Diagnostic Plots for HAR(1, 5, 22) estimated by first finding the mode, then sampling using AMH Haario et al., 2001




(a) Training (b) Validation (c) Test

Figure 11: Quantile-Quantile plots for the baseline HAR model.

| Layer 1 | Layer 2 | Layer 3 | RMSE [MAP] Validation Data |
|---|---|---|---|
| RNN(3, 10) | Linear(10, 4) | | 0.683789 |
| **RNN(3, 3)** | **Linear(3, 4)** | | **0.671309** |
| RNN(3, 20) | Linear(20, 4) | | 0.666374 |
| RNN(3, 10) | RNN(10, 4) | Linear(4, 4) | 0.671759 |
| RNN(3, 10) | RNN(10, 10) | Linear(10, 4) | 0.671799 |
| Dense(3, 4, sigmoid) | RNN(4, 4) | Linear(4, 4) | 0.676713 |

Table 2: Table contains the network structures that were considerd. First number in paranthesis always denotes input size, second number denotes output size. Linear donotes a Dense layer with identity activation function. The bold-faced structure is the final one.

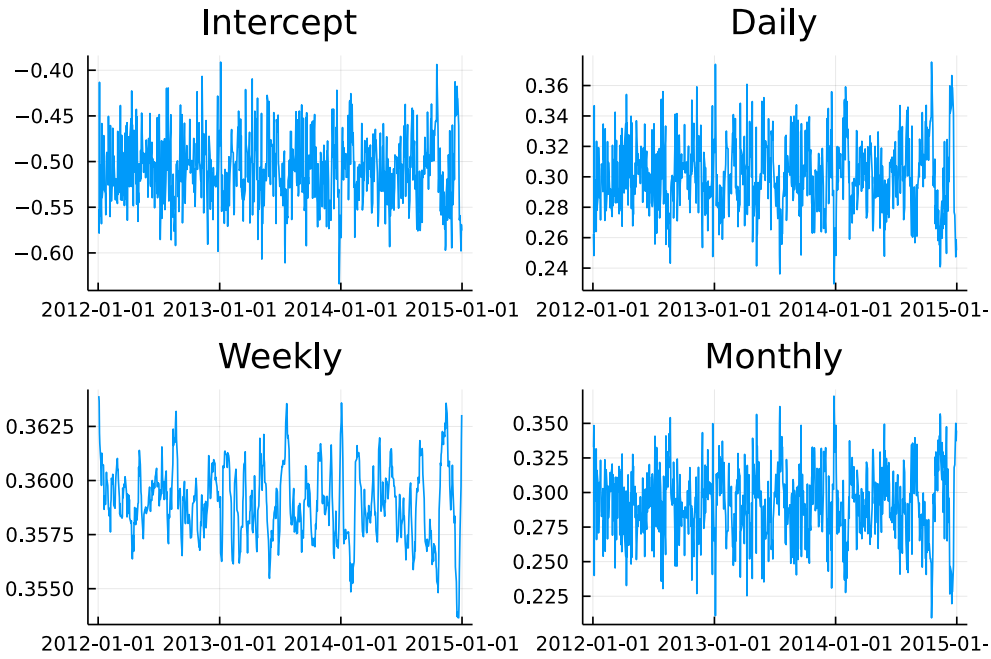



Figure 12: Time variation of coefficients in validation set using the MAP estimate.

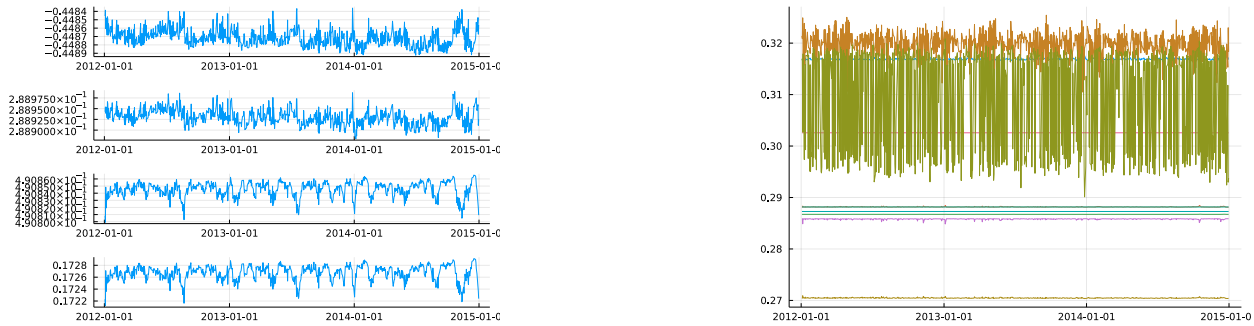
Since larger networks are generally better able to capture highly non-linear relationships, the fact that large and small networks perform similarly could be an indicator for that the relationships in the data are not as non-linear as first hypothesised.

Table 1 reports the training, validation, and test RMSE. This time the validation data was used to select the network structure and can thus no-longer be seen as part of the test data. The MAP estimate of the RNN-HAR model performs better than the standard HAR model in all subsets, although this is again only a small difference and thus whether the difference would be economically meaningful is unclear.

Contrary to the baseline HAR model, the RNN-HAR model allows for time variation in coefficients. Figure 12 shows the uncovered time variation using the MAP estimate for the validation data. Figure 12 looks rather noisy, although some sustained periods of higher or lower coefficient estimates exist, e.g. at the beginning of 2015. Thus, the ability to account for time variation seems to have helped in obtaining better predictions.

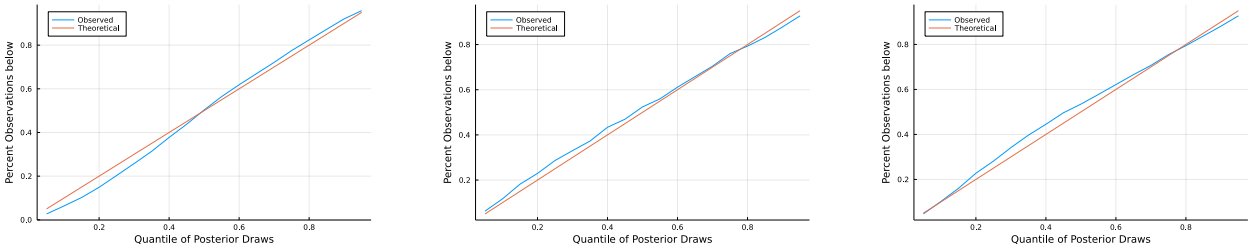
### 6.4 Bayes By Backprop

Although the MAP estimates of the RNN-HAR model showed good performance and time variation, they are not able to provide intervals forecasts and thus still suffer from the same critique expressed earlier. As




(a) (b)

Figure 13: (a) Time variation of coefficients in validation set using BBB. (b) Ten randomly drawn daily coefficient sequences for the validation dataset obtain using BBB.




(a) Training (b) Validation (c) Test

Figure 14: Quantile-Quantile plots for RNN-HAR using BBB

such, the same model was also estimated using Bayes By Backprop (BBB). The training, validation, and test set RMSE are reported in table 1 All RMSE are worse than in the MAP estimate.

Since BBB predictions were made using the average of the posterior predictive values, they effectively represent large ensemble predictions. Large ensemble predictions often perform better than single point predictions, and thus it is rather surprising that the BBB estimated model performed worse than the MAP estimate. Figure 13a shows a potential reason for the underperformance. Shown is the coefficient time variation plot for the validation set. Although the variation is much more visible, the magnitude of the variation is much smaller than in the MAP estimate. If time variation is truly what helped the MAP estimate outperform the baseline, then the very small magnitude changes here would explain the deterioration in performance.

The above still leaves open why the time variation is so much smaller in magnitude here than in the single MAP estimate. Figure 13b shows ten randomly selected coefficient series from the posterior. Clearly, a huge variety exists, with some looking like no-variation really exists, while others look like white noise. In general, the draws obtained from the variational posterior do not seem to produce one consistent set of time varying coefficients. This might explain why the mean coefficient is hardly time varying.

This raises another question: Why do we observe such large variation? At this point we are unable to provide a clear answer to this question, but we hypothesise that it is due to the high variance of the variational posterior. Additionally, it could be due to not allowing for any correlations in the variational posterior. This latter fact allows network parameters to freely move and thus draws from the variational posterior might regularly end up in regions that we would have not ended up in if we had allowed for correlations. One way of overcoming this in the future would be by allowing for correlation structures, but this would let the parameters to be estimated explode and thus might not be feasible for larger models.

Figure 14 shows the quantile-quantile plots on the train, validation, and test set for the BBB estimated RNN-HAR model. Despite the problems discussed above, posterior predictive values obtained accurately reflect the actual data and hence interval forecasts would be good performing.

### 6.5 SGNHT-S

We also estimated the proposed model and network structure using SGNHT-S, just like we did in the simulation study. This time we found parameter settings leading to better mixing within the chain. The maximum rhat for all coefficients across all time periods was 1.07 and thus although not perfect, much closer to what we would like from a traditional Bayesian perspective. The mean kinetic energy was also much closer to one than it was in the simulation study (around 0.5).

Table 1 shows the RMSE for the train, validation, and test set. For all subsets, the mean of the posterior predictive values obtained using SGNHT-S are worse than the baseline and all other results obtained here. This is again rather surprising since we again are having an ensemble which usually performs better than single point estimation (MAP).

Figure 15a shows the time variation within the validation data. We again observe only minimal time variation with magnitudes being much smaller than for the MAP estimate. This might explain the worse performance.

Additionally, it seems like this time the bad performance could also come from that the chain is just not matching the data distribution well. Figure 15b shows quantile-quantile plot for the validation data (they looked similar for train and test). In all cases the tails of the posterior predictive distribution are much thicker than what we observe in the actual data. This is so extreme that the actual percentage of validation data lying below the 20% quantile of the posterior predictive distribution for the validation set is practically zero.

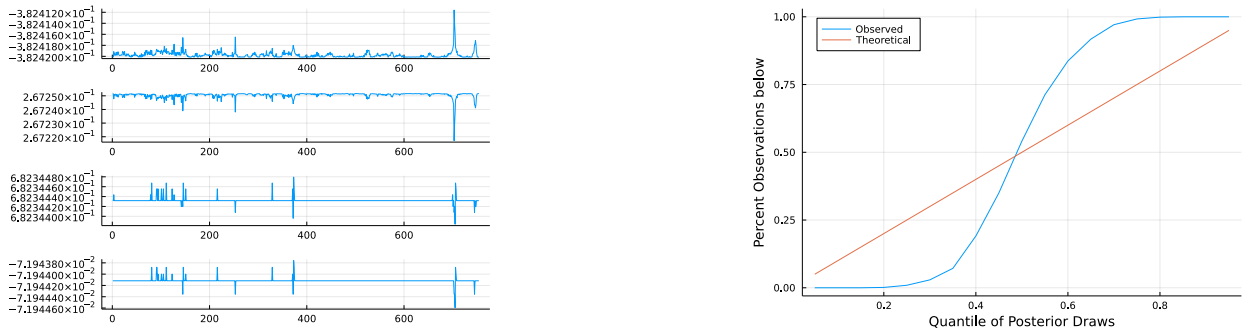
This mismatch between the posterior predictive and observed data could be due to various reasons:

1. Our chain might inaccurately sample from the posterior distribution: This is very likely to be the case no matter the diagnostic statistics that we obtain. The simulations have shown though that this does not need to imply that the performance will be bad. So although a possible reason, there is no way to say for certain.
2. We did not run the sampling procedure for long enough: We admit that this might be a possibility. Although diagnostic statistics and trace plots looked like the chain had approached some stationary distribution, this might have not been the case. However, we would also argue that this is just the common gold-digger problem. We will never know what would have happened if we had dug for longer. We might have found gold and hence better performance, but we might have just as well wasted valuable time that could have been spent on other activities like evaluating alternative models.
3. The probabilistic model proposed is bad. Like Box said: all models are wrong, but some are useful. The proposed model might not be useful in the sense that even when accurately sampled from it might not be accurately capturing the true data distribution. Since there is no way to separate the model from the chain there is no way to tell whether the chain is the culprit or the model.

The only conclusion that can be drawn about the SGNHT-S chain and the model is that the ensemble prediction obtained using those two together are rather poorly performing and of no practical importance due to the huge mismatch in quantile-quantile plots.

## 7 Discussion

In previous sections we derived the rationale for using RNN-HAR models, showed that they were able to uncover some of the non-linear time variation in a small simulation study, and compared the RNN-HAR model to a standard HAR model. In this latter real-data scenario, results were rather mixed and while interval predictions were good for the BBB estimated version, interval predictions obtained from the SGNHT-S estimated versions were practically useless. The MAP estimated version has shown though that the RNN-HAR model does have the capability to outperform the baseline, although only minimally.




(a) (b)

Figure 15: (a) Time variation of coefficients in validation set using SGNHT-S. (b) Quantile-Quantile plot for validation set using SGNHT-S

In this section we will, in a retrospective view, analyse the results of this paper, will discuss how future research could improve on our results and what other valuable lessons and possible paradigm shifts can be taken away.

**Methodological Improvements**

The methodology applied in this research could be improved and criticized in various ways. Former and foremost, the question could be raised why we used an RNN to model the time variation instead of some other model. We admit, many ways of modelling the time variation exist, and some of which would have made specifying appropriate priors easier, such as Gaussian Processes, but we would also argue that given the ability of RNNs to learn any dynamic well-behaved function, makes them a-priori the ideal way of modelling unknown possibly non-linear time-variation. Retrospectively, we would argue though that the input to the RNNs should have been more than just the lagged Realised Volatilities. The non-linear feedback mechanisms from RV to RV might be minimal when compared to the non-linear feedback mechanisms between returns and RV and even returns and RV from other assets to the asset under consideration. In the former case, an abundance of evidence of so-called leverage effects already exists. Thus, future research should investigate whether better results can be obtained when the input to the RNN is extended to also include lagged returns and potentially even information about other assets/asset indices.

We also observed the disconnect between the time variation shown by the MAP estimate in the real-data application and that of Bayes-By-Backprop and argued that this could be due to the high variance in the variational posterior and the restrictions to no-correlations. We argued that it might be difficult to extend BBB to full covariance matrices. At this point we would like to complement this previous argument by pointing out that parameters could be limited to a certain extent by only allowing correlations between neighbouring nodes or layers. Although this would still increase the number of parameters to be estimated, it would do so less than using a full covariance matrix. We are currently thinking about implementing this in *BFlux*.

**Do we need a paradigm shift?**

For both the simulation study and the real-data application we have argued that making any of the commonly used MCMC methods, and here especially SGNHT-S work well in the traditional Bayesian sense (good diagnostic statistics) is nearly impossible. This raises the question whether we should even aim for it? Is there any good reason to believe that any MCMC method will ever be able to accurately sample from Bayesian Neural Network posterior. We start to believe that this is not the case.

Evidence seems to pile up that the traditional loss surfaces of NNs are highly complex (see for example Garipov et al., 2018 and references therein) even when penalisations are used. Since a penalised loss is nothing else than the negative log-posterior, all this evidence essentially implies that Bayesian Neural Network posteriors are highly complex. If we cannot expect traditional optimisation to work perfectly, how can we expect to properly map out the entire loss surface, corresponding to sampling from the entire posterior distribution.

We would simply state that we cannot! And would even go as far as saying that we cannot even in the output space of the network. That is because, as Garipov et al., 2018 argue the mode-connecting path is not a path necessarily a path of equivalent parameterisations, but often contains parameterisations that result in different predictions on validation and test sets.

The complexities of Bayesian Neural Network posteriors and the good performance of bad chains in the traditional sense make us believe that a paradigm shift might be needed in order to make BNNs work. Instead of aiming to obtain good samples from the posterior, we should rather consider the sampling algorithms as ensemble algorithms. The underlying probabilistic model gives structure to the kind of ensembles we would like to have, similar to the structure of a neural network itself. Combining the samples and the probabilistic model and forming approximate posterior predictive values then provides us with an ensemble forecast which to any practical purposes can be used to obtain point forecasts or interval forecasts. The simulation study has shown that these traditionally bad chains can indeed provide accurate point and interval forecasts and in any practical application these ensemble forecasts can always be avaluated and diagnosed on a validation set.

Going the ensemble view might then call for other diagnostic statistics then are currently used. For example, instead of aiming to obtain good mixing chains in this research we could have aimed at obtaining chains that result in a close quantile-quantile match on the training set, which could have then be validated on the validation set. This could have changed the results in this paper drastically, since the aim for a good mixing chain lead to a very bad quantile-quantile match in the real-data application. Future research will investigate how different the performance could have been and whether a paradigm shift is indeed worth it.

Taking the ensemble view does not mean though that one should completely forget about traditional diagnostic statistics. Mixing statistics and effective sample size are likely to still provide valuable information. Chains with very bad mixing are still unlikely to be good ensemble models, but we hypothesise that there is a mixing rate much higher than is traditionally aimed at, at which the ensemble already performs good despite being a bad chain from a traditional viewpoint. And this mixing rate might be quite high in some circumstances. Imagine, for example, a chain that walks along the mode-connecting path (Garipov et al., 2018). At the beginning of the chain we are in one mode, while at the end we are at another. If the chain does not significantly backtrack in-between, then rhat will be very high and the chain would traditionally be seen as badly mixing and essentially useless. However, in the ensemble view, the chain still covered a large set of very likely explanations of the data and can thus still provide good point and accurate interval predictions.

Although the paradigm shift could have been helpful in this research and will be investigated in the future, it should not be taken to the extreme. The only reason why the paradigm works here, is that we do not wish to make any causal statements and at most are interested in approximate explanations of predictions. Since any causal analysis is inherently tied to the model used to estimate the coefficients and accurate estimates are needed, the paradigm would not apply to causal analysis.

# 8 Conclusion

We derived and proposed the RNN-HAR model which allows for flexible non-linear time variation in the coefficients of the traditional HAR(1, 5, 22) model. Simulation experiments have shown that if non-linear time variation is present, we are able to uncover it, although not perfectly. The same simulation study has also shown that despite not having perfect MCMC chains or being able to perfectly uncover the time variation, good point and accurate interval predictions can still be made. We then applied the model to Realised Volatility data of SPX. The results here were rather poor. While the MAP estimate slightly outperformed the baseline HAR model, the economic significance is unknown and likely minimal.

Estimations obtained using Bayes-By-Backprop (BBB) resulted in worse point predictive performance than the baseline, but nonetheless delivered accurate out-of-sample interval forecasts. We argued that the underperformance is likely due to the large variational posterior variance and not allowing for correlations. In the discussion we pointed out how this could be improved in the future by either allowing for a full covariance

matrix or restricting correlations to only neighbouring nodes, and thus still somewhat keeping the number of parameters small.

MCMC chains obtained using SGNHTS mixed badly in a traditional sense in both the simulation and real-data application, although better in the latter. While in the simulation study, the samples obtained from the chains were still useful and provided accurate point and interval forecasts, they were, for all practical purposes, useless in the real-data application.

The difficulty in obtaining good mixing chains even in the output space lead us to reconsider the paradigm we are working under. We argued that trying to obtain accurate posterior draws is a hopeless goal in Bayesian Neural Networks, and argued that samples obtained using MCMC algorithms should rather be seen as ensemble forecasts structured by the underlying probabilistic model. The simulation study has shown that these bad mixing chains can provide accurate descriptions of the data distribution.

# References


Andersen, T. G., Bollerslev, T., Diebold, F. X., & Labys, P. (2001). The Distribution of Realized Exchange Rate Volatility. *Journal of the American Statistical Association*, *96*(453), 42–55. https://doi.org/10.1198/016214501750332965
_eprint: https://doi.org/10.1198/016214501750332965

Barndorff-Nielsen, O. E., Hansen, P. R., Lunde, A., & Shephard, N. (2008). Designing Realized Kernels to Measure the ex post Variation of Equity Prices in the Presence of Noise. *Econometrica*, *76*(6), 1481–1536. https://doi.org/10.3982/ECTA6495
_eprint: https://onlinelibrary.wiley.com/doi/pdf/10.3982/ECTA6495

Barndorff-Nielsen, O. E., Kinnebrock, S., & Shephard, N. (2008). *Measuring Downside Risk - Realised Semivariance* (SSRN Scholarly Paper No. 1262194). Social Science Research Network. Rochester, NY. https://doi.org/10.2139/ssrn.1262194

Barndorff-Nielsen, O. E., & Shephard, N. (2002a). Econometric analysis of realized volatility and its use in estimating stochastic volatility models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, *64*(2), 253–280. https://doi.org/10.1111/1467-9868.00336
_eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/1467-9868.00336

Barndorff-Nielsen, O. E., & Shephard, N. (2002b). Estimating quadratic variation using realized variance. *Journal of Applied Econometrics*, *17*(5), 457–477. https://doi.org/10.1002/jae.691
_eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/jae.691

Basturk, N., Schotman, P. C., & Schyns, H. (2021). A neural network with shared dynamics for multi-step prediction of value-at-risk and volatility. *Available at SSRN 3871096*.

Blundell, C., Cornebise, J., Kavukcuoglu, K., & Wierstra, D. (2015). Weight Uncertainty in Neural Networks. *International conference on machine learning*, 1613–1622.

Bucci, A. (2020). Realized volatility forecasting with neural networks. *Journal of Financial Econometrics*, *18*(3), 502–531.

Chen, X. B., Gao, J., Li, D., & Silvapulle, P. (2018). Nonparametric estimation and forecasting for time-varying coefficient realized volatility models. *Journal of Business & Economic Statistics*, *36*(1), 88–100.

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

Corsi, F., Audrino, F., & Renó, R. (2012). Har modeling for realized volatility forecasting.

Corsi, F., & Reno, R. (2009). Har volatility modelling with heterogeneous leverage and jumps. *Available at SSRN*, *1316953*.

Donfack, M. N., & Dufays, A. (2021). Modeling time-varying parameters using artificial neural networks: A garch illustration. *Studies in Nonlinear Dynamics & Econometrics*, *25*(5), 311–343.

Fernandes, M., Medeiros, M. C., & Scharth, M. (2014). Modeling and predicting the cboe market volatility index. *Journal of Banking & Finance*, *40*, 1–10.

Garipov, T., Izmailov, P., Podoprikhin, D., Vetrov, D. P., & Wilson, A. G. (2018). Loss Surfaces, Mode Connectivity, and Fast Ensembling of DNNs. *Advances in Neural Information Processing Systems*, *31*.

Garriga-Alonso, A., & Fortuin, V. (2021). Exact Langevin Dynamics with Stochastic Gradients.

Haario, H., Saksman, E., & Tamminen, J. (2001). An Adaptive Metropolis Algorithm. *Bernoulli*, *7*(2), 223. https://doi.org/10.2307/3318737

Hamid, S. A., & Iqbal, Z. (2004). Using neural networks for forecasting volatility of s&p 500 index futures prices. *Journal of Business Research*, *57*(10), 1116–1125.

Hansen, P. R., & Lunde, A. (2006). Realized Variance and Market Microstructure Noise. *Journal of Business & Economic Statistics*, *24*(2), 127–161. https://doi.org/10.1198/073500106000000071
_eprint: https://doi.org/10.1198/073500106000000071

Heber, G., Lunde, A., Shephard, N., & Shephard, K. (2009). Oxford-man institute's realized library (version 0.3). https://realized.oxford-man.ox.ac.uk

Huang, Z., Liu, H., & Wang, T. (2016). Modeling long memory volatility using realized measures of volatility: A realized har garch model. *Economic Modelling*, *52*, 812–821.

Leimkuhler, B., & Shang, X. (2016). Adaptive Thermostats for Noisy Gradient Systems. *SIAM Journal on Scientific Computing*, *38*(2), A712–A736. https://doi.org/10.1137/15M102318X

Liu, C., & Maheu, J. M. (2008). Are there structural breaks in realized volatility? *Journal of Financial Econometrics*, *6*(3), 326–360.

Nemeth, C., & Fearnhead, P. (2019). Stochastic gradient Markov chain Monte Carlo.

Wang, Y., Ma, F., Wei, Y., & Wu, C. (2016). Forecasting realized volatility in a changing world: A dynamic model averaging approach. *Journal of Banking & Finance*, *64*, 136–149.