Machine Learning Based Semiparametric Time Series Conditional Variance: Estimation and Forecasting

Justin Dang* and Aman Ullah†

November, 2021

Abstract

This paper proposes a new combined semiparametric estimator of the conditional variance that takes the product of a parametric estimator and a nonparametric estimator based on machine learning. A popular kernel based machine learning algorithm, known as kernel regularized least squares estimator, is used to estimate the nonparametric component. We discuss how to estimate the semiparametric estimator using real data and how to use this estimator to make forecasts for the conditional variance. Simulations are conducted to show the dominance of the proposed estimator in terms of mean squared error. An empirical application using S&P 500 daily returns is analyzed, and the semiparametric estimator effectively forecasts future volatility.

Keywords: Conditional variance; Nonparametric estimator; Semiparametric models; Forecasting; Machine Learning; Kernel Regularized Least Squares.

JEL Classification: C01, C14, C51, C53, C58

---

*Department of Economics, University of California, Riverside. Email: jdang015@ucr.edu
†Department of Economics, University of California, Riverside. Email: aman.ullah@ucr.edu

1   Introduction

It is well known that financial time series data, especially stock returns, are not predictable. On the other hand, volatility (risk) or the conditional variance of the returns are easier to predict since autocorrelations are shown to be significant in various time series data. There are a plethora of models that try and capture the dynamics of the volatility of a series, including both parametric and nonparametric models. In this paper, both a parametric and nonparametric model of the conditional variance is combined in a multiplicative way, resulting in a semiparametric estimator of the conditional variance. The parametric component of the conditional variance is used mainly to capture the overall dynamics whereas the nonparametric part analyzes the residual variance and catches the nonlinearities in the data that the parametric component fails to capture. The generalized autoregressive conditional heteroskedasticity (GARCH) family of models are considered for the parametric estimation. For the nonparametric estimation, a popular machine learning (ML) algorithm is used, known as Kernel Regularized Least Squares (KRLS), which does not rely on linearity or additivity assumptions Hainmueller and Hazlett (2014).

The GARCH family of models is a common method to modeling volatility as the conditional variance, σ_t^2, for a stocastic process y_t. The first among these started with Engle (1982) for the ARCH model and Bollerslev (1986) for the GARCH model of the conditional variance. Later, many papers following their work paved the way for other GARCH-type models that capture many stylized facts including being able to capture volatility dynamics and incorporating leverage effects (Glosten et al. (1993), Nelson (1991), and Ding et al. (1993)). All of these models are considered to be parametric and are subject to misspecification. If a GARCH model, or any other parametric model, is correctly specified, then it can be shown that the model can consistently estimate the conditional variance. However, when a parametric model is misspecified, then the volatility estimator is often inconsistent for σ_t^2. In this paper, we will use ARCH(1) and GARCH(1, 1) models to estimate the parametric conditional variance and use a ML based nonparametric method to enhance the conditional variance estimates.

Nonparametric and semiparametric estimation of the conditional variance have also proven to capture dynamics in the volatility well. To estimate the conditional variance, Ziegelmann (2002) models the squared residuals, from the conditional mean equation, nonparametrically via the local exponential estimator. However, this procedure may not be applicable for the GARCH-type conditional variation. In this paper, however, the conditional variance function is considered to be semiparametric as the product of a parametric part and nonparametric part similar to Mishra et al. (2010). This framework stems from the seminal work by Glad (1998) where a multiplicative semiparametric model of the conditional mean function is formulated. In this paper, the framework is based on Glad (1998) and Mishra et al. (2010) for the conditional variance by multiplying the parametric component by the nonparametric component. However, instead of using local exponential (or polynomial) smoothing for estimating the nonparametric component as they do, we will use another nonparametric technique known as KRLS based ML.

KRLS is a common ML technique in the nonparametric ML literature. The method uses kernels to find the best fit surface in a flexible hypothesis space by minimizing a penalized least squares problem, see Hainmueller and Hazlett (2014). De Brebanter et al. (2011) refers

to the same method as Least Squares Support Vector Machines (LSSVM) and proposes bias corrected confidence intervals for the estimator. However, these papers study the conditional mean function, and not the conditonal variance function. This paper differs by estimating the conditional variance function with the help of ML. That is, standardized squared residuals, obtained from the conditional mean equation, are used in a ML regression to estimate the nonparametric component of the conditional variance. With the help of ML, the nonparametric component can hopefully capture the nonlinearites in the true volatility function, where the parametric component failed to do so.

Overall, this paper proposes a semiparametric estimator based on the ML technique KRLS. The proposed estimator is capable of handling in and out of sample data, where forecasting the future conditional variance based on a conditioning set is achieved. We show in simulation that the semiparametric estimator beats the parametric counterpart in terms of mean squared error. An empirical example with real data from S&P 500 daily returs is also given to illustrate the advantage of using the semiparametric ML estimator of the conditional variance, where news impact curves and out of sample forecasts are evaluated. In the empirical example, the semiparametric estimator is shown to be superior to the parametric estimators when evaluating out of sample forecasts.

The structure of this paper is as follows: section 2 discusses the model framework of the conditional variance, section 3 proposes the semiparametric estimator using ML to estimate the nonparametric component, section 4 shows how forecasts are made via the semiparametric estimator, section 5 runs through a simulation example, section 6 illustrates an empirical example using S&P 500 daily returns, and section 7 concludes the paper.

2 The Model

Consider the following model:

$$y_t = \alpha_t + \varepsilon_t, \tag{1}$$
$$\alpha_t = g(U_t, \boldsymbol{\beta}) \tag{2}$$

where ε_t satisfies $\mathbb{E}[\varepsilon_t|\mathcal{F}_{t-1}] = 0$ and $\mathbb{E}[\varepsilon_t^2|\mathcal{F}_{t-1}] = \sigma_t^2$ and $U_t \in \mathcal{F}_{t-1}$ represent a vector of regressors, and $\boldsymbol{\beta}$ represent the vector of parametric parameters for the mean regression. The function $g(\cdot)$ can be any parametric function of the data U_t and parameter vector $\boldsymbol{\beta}$, such as an ARMA function or a constant function for the conditional mean function. Note that since $U_t \in \mathcal{F}_{t-1}$, $\text{Var}[y_t|\mathcal{F}_{t-1}] = \text{Var}[y_t - \alpha_t|\mathcal{F}_{t-1}] = \text{Var}[\varepsilon_t|\mathcal{F}_{t-1}] = \sigma_t^2$.

Similar to Glad (1998) and Mishra et al. (2010), we can redefine the conditional variance, $\sigma_t^2 = \mathbb{E}[\varepsilon_t^2|\mathcal{F}_{t-1}]$ as

$$\mathbb{E}[\varepsilon_t^2|\mathcal{F}_{t-1}] = \sigma_{p,t}^2\mathbb{E}\left[\left(\frac{\varepsilon_t}{\sigma_{p,t}}\right)^2\middle|\mathcal{F}_{t-1}\right], \tag{3}$$

where $\sigma_{p,t}^2 \in \mathcal{F}_{t-1}$ is found by a parametric specification of the conditional variance. Let $\sigma_{np,t}^2 = \mathbb{E}[(\varepsilon_t/\sigma_{p,t})^2|\mathcal{F}_{t-1}] \in \mathcal{F}_{t-1}$ denote the nonparametric specification of the conditional variance. We can think of $(\varepsilon_t/\sigma_{p,t})^2$ as a new dependent variable in the nonparametric

estimation stage. Then, the final semiparametric estimator of the conditional variance is

$$\sigma_t^2 = \sigma_{p,t}^2 \sigma_{np,t}^2. \tag{4}$$

The idea is that if the parametric form of the conditional variance fails to capture some feature of the true conditional variance, the nonparametric form of the conditional variance will have a multiplicative correction factor that will hopefully be easier to estimate. If the parametric conditional variance underestimates the true conditional variance, in which the parametric model will be subject to misspecification, the nonparametric component, $\sigma_{np,t}^2$, will be greater than one and vice versa. In the extreme case, where the parametric form of the conditional variance is correctly specified, the nonparametric component, $\sigma_{np,t}^2$, will hopefully be constant over time.

3   The Semiparametric ML Estimator

Since, ε_t is unobservable, the first step is to obtain the residuals of the estimated regression equation.

$$\hat{\varepsilon}_t = y_t - \widehat{\alpha}_t, \quad t = 1, \dots, T \tag{5}$$

where $\widehat{\alpha}_t = \widehat{g}(U_t, \widehat{\boldsymbol{\beta}})$ is the estimated regression function, e.g. ARMA process and T is the total number of observations in the sample. We will then use the residuals in Eq. (5) to estimate both the parametric and nonparametric components of the conditional variance.

3.1   Estimating the Parametric Component

Let $\sigma_{p,t}^2 = \sigma_p^2(X_{1,t})$ denote the paramatric conditional variance, which depends on regressors $X_{1,t} \in \mathbb{R}^{d_1}$. We can estimate the parametric conditional variance, $\sigma_p^2(X_{1,t})$, by using the squared residuals obtained from Eq. (5), with any parametric form, such as an ARCH(1) model, where $\sigma_p^2(X_{1,t}) = \omega + \alpha_1 \varepsilon_{t-1}^2$ and $X_{1,t} = \varepsilon_{t-1}$. Any GARCH type model may also be used, which is often estimated by the quasi maximum likelihood approach. Note that the parametric form may be misspecified. The idea is that even if the parametric form is misspecified, the nonparametric component will hopefully pick up the parts of the conditional variance that the parametric part failed to estimate correctly. Once the parametric conditional variance is estimated, predictions can be made by $\widehat{\sigma}_p^2(x_1)$ for some test observation x_1.

3.2   Estimating the Nonparametric Component via ML

To get the new dependent variable to be used in the nonparametric estimation of the conditional variance, we can estimate the standardized residuals as $z_t = \widehat{\varepsilon}_t / \widehat{\sigma}_p(X_{1,t})$, $t = 1, \dots, T$, using the estimated residuals and the estimated parametric conditional variance evaluated at each data point in the sample. Let $z_t^2 = z^2(X_{2,t})$ depend on regressors $X_{2,t} \in \mathbb{R}^{d_2}$. Note that the regressors, $X_{j,t} \in \mathcal{F}_{t-1}$ for $j = 1, 2$, are considered past information. Then we can regress z_t^2 on $X_{2,t}$ to get the nonparametric component of the conditional variance.

For KRLS based ML, the function $\sigma_{np}^2(\cdot)$ can be approximated by some function in the space of functions constituted by

$$\sigma_{np}^2(x_2) = \sum_{i=1}^{T} c_t K_\gamma(X_{2,i}, x_2), \tag{6}$$

where $X_{2,i}$, $i = 1, \ldots, T$ is the vector of regressors considered for the nonparametric estimation for some test observation x_2 and where c_i, $i = 1, \ldots, T$ are the parameters of interest, which can be thought of as the weights of the kernel functions $K_\gamma(\cdot)$. The subscript of the kernel function, $K_\gamma(\cdot)$, indicates that the kernel depends on the bandwidth parameter, γ. We will use the Radial Basis Function (RBF) kernel,

$$K_\gamma(X_i, x_2) = \mathrm{e}^{-\frac{1}{\gamma^2}||X_i - x_2||^2}. \tag{7}$$

Notice that the RBF kernel is very similar to the Gaussian kernel, in that it does not have the normalizing term out in front and that γ is proportional to the bandwidth h in the Gaussian kernel often used in nonparametric local polynomial regression. This functional form is justified by a regularized least squares problem with a feature mapping function that maps x into a higher dimension, see Hainmueller and Hazlett (2014), where this derivation of KRLS is also known as Kernel Ridge Regression (KRR). Overall, KRLS uses a quadratic loss with a weighted L_2-regularization. Then, in matrix notation, the minimization problem is

$$\arg\min_{\mathbf{c}} \ (\mathbf{z} - \mathbf{K}_\gamma \mathbf{c})^\top (\mathbf{z} - \mathbf{K}_\gamma \mathbf{c}) + \lambda \mathbf{c}^\top \mathbf{K}_\gamma \mathbf{c}, \tag{8}$$

where $\mathbf{z} = \left(\widehat{\varepsilon}_1/\widehat{\sigma}_p(X_{1,1}), \ \ldots, \ \widehat{\varepsilon}_T/\widehat{\sigma}_p(X_{1,T})\right)^\top$ is the vector of the standardized residuals to be used as the responses in this regression and $\mathbf{K}_\gamma$ is the kernel matrix, with $K_{\gamma,i,j} = K_\gamma(X_i, X_j)$ for $i, j = 1, \ldots, T$ and $\mathbf{c}$ is the vector of coefficients that is optimized over. The solution to this minimization problem is

$$\widehat{\mathbf{c}} = (\mathbf{K}_\gamma + \lambda \mathbf{I})^{-1} \mathbf{z}. \tag{9}$$

The kernel function and its parameters are user specified but can be found via cross validation along with the regularization parameter λ. In simulation and in the empirical application, leave one out cross validation is used to find the hyperparameters, γ and λ. Finally, predictions for the nonparametric ML conditional variance estimator can be made by

$$\widehat{\sigma}_{np}^2(x_2) = \sum_{i=1}^{T} \widehat{c}_i K_\gamma(X_i, x_2), \tag{10}$$

for some test observation x_2.

3.3 The Semiparametric ML Estimator

Now that we have both the estimated parametric, $\widehat{\sigma}_p^2(x_1)$, and nonparametric, $\widehat{\sigma}_{np}^2(x_2)$, components of the conditional variance, we can estimate the final semiparametric conditional

variance in a multiplicative way and predictions can be made by

$$\widehat{\sigma}_{sp}^2(x) = \widehat{\sigma}_p^2(x_1)\widehat{\sigma}_{np}^2(x_2), \tag{11}$$

for some test observation point $x = \left(x_1^\top, \ x_2^\top\right)^\top$.

Notice that both X_1 and X_2 can be unobservable. To illustrate, we will discuss the GARCH$(1,1)$ model of the conditional variance, $\sigma_t^2(X_{1,t}) = \omega + \alpha\varepsilon_{t-1} + \beta_1\sigma_{t-1}^2$. Here, $X_{1,t} = \left(\varepsilon_{t-1}, \ \sigma_{t-1}^2\right)^\top$. The estimate for $\sigma_{p,t}^2$ can be obtained by any parametric model based on quasi maximum likelihood estimation (QMLE), which is standard for ARCH and GARCH models. Now, suppose that we estimate the nonparametric component in a similar fashion but instead of imposing a parametric form of the conditional variance, we will use ML. That is, we estimate $\sigma^2(X_{2,t})$ nonparametrically via ML, where $X_{2,t} = \left(\varepsilon_{t-1}, \ \sigma_{t-1}^2\right)^\top$. In this case, we estimate the lagged residuals by Eq. (5) and we will use the parametric conditional variance estimates for σ_{t-1}^2. Therefore, we use $\widehat{X}_{2,t} = \left(\widehat{\varepsilon}_{t-1}, \ \widehat{\sigma}_{p,t-1}^2\right)^\top$ as regressors for the nonparametric estimation. Then, Eq. (11) can be used to obtain the final conditional variance estimate. This estimator will be denoted as SPMLGARCH estimator for using a GARCH model for the parametric component and ML for the nonparametric component. Without loss of generality, the procedure can be extended to higher order lags.

4 Forecasting

To forecast the conditional variance, we can use Eq. (10) and Eq. (11). Let $h \in \mathbb{Z}^+$ denote the number of periods ahead we wish to forecast, e.g. $h = 1$ for a one step ahead forecast. First, we use the training sample to fit a parametric model of the conditional variance, $\widehat{\sigma}_p^2$, and forecast h periods ahead using standard techniques, $\widehat{\sigma}_p^2(x_{1,T+h})$. Then, we use the standardized residuals to fit a nonparametric model of the conditional variance via ML, $\widehat{\sigma}_{np}^2$, and forecast h periods ahead using Eq. (10), $\widehat{\sigma}_{np}^2(x_{2,T+h})$. Finally combining the two components gives the h period ahead forecast for the semiparametric conditional variance,

$$\widehat{\sigma}_{sp}^2(x_{T+h}) = \widehat{\sigma}_p^2(x_{1,T+h})\widehat{\sigma}_{np}^2(x_{2,T+h}), \tag{12}$$

where $x_{T+h} = \left(x_{1,T+h}^\top, \ x_{2,T+h}^\top\right)^\top$.

In the case of the GARCH$(1,1)$ model, parametric forecasts can be made in the standard way. However, for the nonparametric case, the forecasts may depend on regressors that include future residuals and conditional variances that are unobservable. Recall that a possile vector of regressors for the nonparametric regression of $\sigma^2(X_{2,t})$ can be $X_{2,t} = \left(\varepsilon_{t-1}, \ \sigma_{t-1}^2\right)^\top$ as in the SPMLGARCH example discussed previously. To deal with this, we estimate future residuals, ε_{T+h-1}, to be 0 for $h > 1$. When $h = 1$, i.e. one step ahead forecast, ε_T can be estimated by the T-th residual from the estimated regression function. To be consistent with the estimation procedure, the parametric conditional variance forecasts will be used for future predictions of σ_{t-1}^2 in $X_{2,t}$ when forecasting the nonparametric component of the conditional variance. That is, let $x_{2,T+h} = \left(\widehat{\varepsilon}_{T+h-1}, \ \widehat{\sigma}_{p,T+h-1}^2\right)^\top$ denote the test observation point, or the conditioning set, to be used in forecasting the nonparametric component of the

conditional variance in Eq. (12).

5   Simulation

This section evaluates the semiparametric ML estimator in a simulation setting. The data generating process will come from GJR's model of conditional variance, which takes the leverage effect into consideration, see Glosten et al. (1993). Consider the following data generating process (DGP):

$$
\begin{aligned}
y_t &= \varepsilon_t \\
\sigma_t^2 &= \omega + \alpha_1 \varepsilon_{t-1}^2 + \beta_1 \sigma_{t-1}^2 + \gamma_1 \varepsilon_{t-1}^2 \mathbf{1}(\varepsilon_{t-1} \leq 0)
\end{aligned}
\tag{13}
$$

where we assume that the mean function $\alpha_t = 0$, $\omega = 0.03$, $\alpha_1 = 0.5$, $\beta_1 = 0.4$, and $\gamma_1 = 0.03$. We also assume that $\varepsilon_t = \sigma_t \nu_t$, where $\nu_t \overset{i.i.d.}{\sim} N(0,1)$, $\mathbb{E}[\varepsilon_t | \mathcal{F}_{t-1}] = 0$, and $\mathbb{E}[\varepsilon_t^2 | \mathcal{F}_{t-1}] = \sigma_t^2$. The leverage effect comes from the indicator function, $\mathbf{1}(\cdot)$, which makes the effect of negative shocks different; a positive shock is captured by α_1 and a negative shock is captured by $\alpha_1 + \gamma_1$. The simulated data and GARCH models are estimated by the R package `rugarch` from Ghalanos (2020)[1].

We evaluate four models: ARCH(1), see Engle (1982), denoted by ARCH, GARCH(1, 1), see Bollerslev (1986), denoted by GARCH, and their semiparametric ML counterparts, denoted by SPMLARCH and SPMLGARCH, where ARCH and GARCH models are estimated in the first stage. For the second stage in SPMLARCH, we let $X_{2,t} = \varepsilon_{t-1}$ and in SPML-GARCH, we let $X_{2,t} = \left(\varepsilon_{t-1}, \ \widehat{\sigma}_{p,t-1}^2 \right)^\top$, where $\widehat{\sigma}_{p,t-1}^2$ is the fitted parametric conditional variance estimates in the first stage. To avoid the starting-out effect, we discard the first 500 observations and use a sample size of $n = 200$. The number of replications is $M = 200$. To evalaute these four models, we will use the mean squared error (MSE) criteria.

$$
MSE = \frac{1}{M} \sum_{m=1}^{M} \frac{1}{n-1} \sum_{t=2}^{n} \left(\widehat{\sigma}_{t,m}^2 - \sigma_t^2 \right)^2,
\tag{14}
$$

where $\widehat{\sigma}_{t,m}^2$ is the estimated conditional variance at time t for replication m and σ_t^2 is the true condtional variance at time t. The lowest MSE suggests the best model.

Table 1 displays the MSE's for each of the four estimators considered. For this DGP, both the parametric and the semiparametric ARCH models perform the worst. Although the parametric GARCH model performs adequately well, SPMLGARCH improves upon the parametric GARCH and outperforms all other models in terms of MSE. Since the ARCH models only considers lagged values of the residuals, these models fail to pick up the importance of the GARCH effects in the true conditional variance. The SPMLARCH model actually is the least preferred and shows that the nonparametric estimation stage may not help if GARCH effects are not considered when these effects are actually present. Regardless, the SPMLGARCH model, where GARCH effects are considered in both the parametric and

---

[1]All other packages used include Wickham et al. (2021), Hyndman et al. (2020), Warnes et al. (2017), Borchers (2021), and Zeileis and Grothendieck (2005).

nonparametric estimation stages, is deemed the most appropriate model in this scenario, which has the lowest MSE compared to the parametric GARCH model as well as compared to the others.

<div align="center">

MSE's of Estimators for Conditional Variance

ARCH	GARCH	SPMLARCH	SPMLGARCH
0.2294	0.0317	0.3768	0.0209

</div>

Table 1: *The table shows MSE's calculated by Eq. (14) for the ARCH, GARCH, SPM-LARCH, and SPMLGARCH models under the DGP specified in Eq. (13). The sample size for each estimation is $n = 200$ and the number of replications is $M = 200$.*

6 Application

In this section, we fit the semiparametric ML model to real data, obtained from Federal Reserve Economic Data (FRED).[2] The data contain S&P 500 daily returns from August 5, 2016 to August 5, 2021 (1305 observations). Missing oversations due to non-trading days are imputed as means of the previous and following daily returns. We allow for the mean component α_t to be an ARMA process. For the conditional variance, we evaluate four models ARCH, GARCH, SPMLARCH, and SPMLGARCH. That is, in the first-stage parametric estimation, we estimate α_t by an ARMA model, and estimate the parametric conditional variance models ARCH(1) and GARCH(1, 1) by using the Gaussian QMLE method. In the second-stage nonparametric modeling, we estimate $\sigma_{np,t}^2$ by using the conditioning set ε_{t-1} for SPMLARCH and $\left(\varepsilon_{t-1}, \quad \sigma_{t-1}^2 \right)^\top$ for SPMLGARCH. To evaluate out of sample predictions, we split the full sample into a training data set where $t = 1, \ldots, 925$ and the testing data set where $t = 926, \ldots, 1305$. The data are split in this way so that the volatility can be forecasted for a time period in which there is high volatility. The date $t = 925$ corresponds to Febuary 20, 2020, which is right when COVID-19 started to have a huge impact on the U.S. as well as its financial markets. A rolling window scheme with the window size of $T = 925$ is used for one and five steps ahead forecasts.

It is common practice to study the impact of the innovation, ε_t on the current conditional variance, holding all other factors fixed, which is known as the news impact curve, see Engle and Ng (1993). Due to the nature of ARCH and GARCH models, these parametric models often have symmetric news impact curves where both negative and positive shocks in absolute value have the same effect on the conditional variance, which is uncommon in financial data. The goal of estimating the nonparametric component is to pick up the asymmetry, and other nonlinearities, in the data. To see this, we can plot the nonparametric component of the conditional variance evalauted across different values of ε_t. In the extreme case where the parametric model is correctly specified, the nonparametric component will be a constant and the curve will be a horizontal line at one.

---

[2]The data can be obtained from https://fred.stlouisfed.org/series/SP500#.
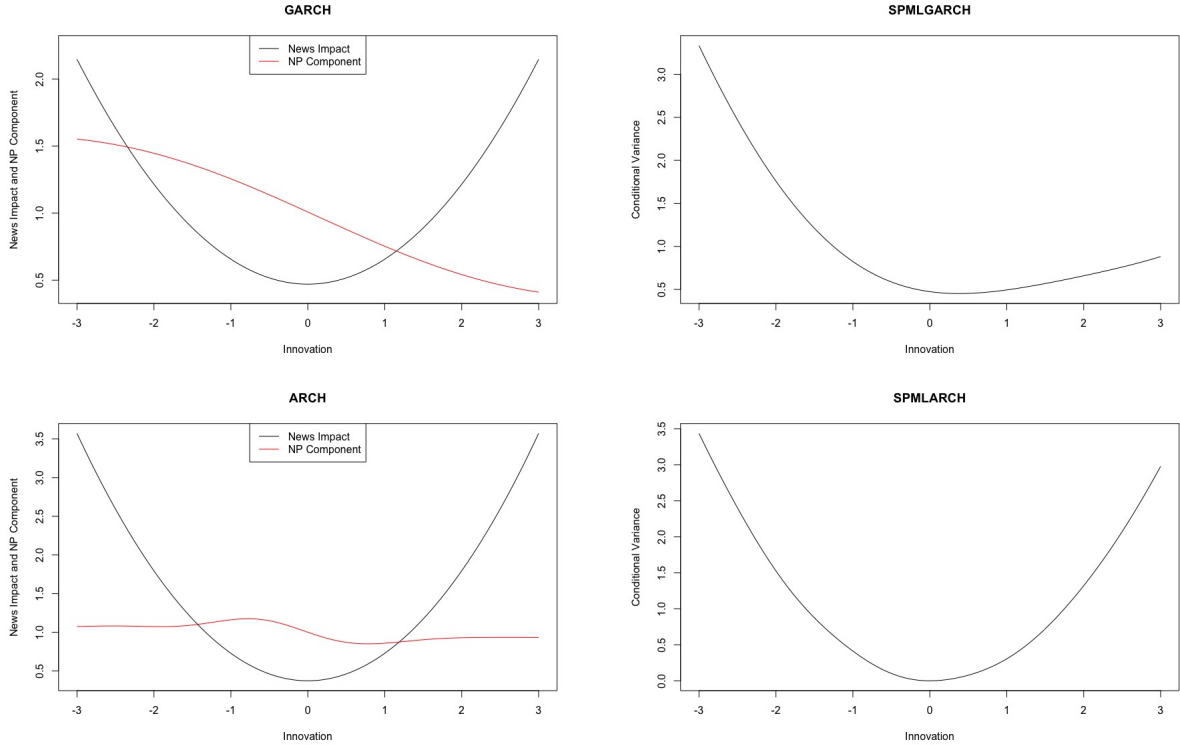
Figure 1: *The left two plots show the news impact curves in black and second-stage nonlinearities in red for S&P 500 daily returns. The right two plots show the conditional variance estimates for SPMLGARCH and SPMLGARCH.*

Using the training data set from $t = 1, \ldots, 925$, the estimated news impact curve and second-stage nonlinearities are shown in the left two plots of Figure 1. Since the GARCH model has two variables in the conditioning set $\left(\varepsilon_{t-1}, \quad \sigma_{t-1}^2 \right)^\top$, we evaluate the curves at 100 evenly spaced points from -3 to 3 for ε_{t-1} while holding $\sigma_{t-1}^2 = \bar{\sigma}_p^2$ fixed at the mean of the estimated lagged parametric conditional variances. For the ARCH model, the conditioning set only contains one variable, ε_{t-1}, where we evaluate the ARCH model at 100 evenly spaced points from -3 to 3. As expected, the news impact curve for both the ARCH and GARCH models seem to be symmetric. However, the nonparametric curve is not constant at one and in fact is somewhat downwards sloping. This implies that the nonparamtric component is able to detect the nonlinearity in the data and shows the leverage effect is present in this data, which is common in various financial data.

The right two plots of Figure 1 show the conditional variance estimates for innovations spanning -3 to 3, holding other factors fixed. Visually, these plots are found by multiplying the nonparametric component and the news impact curve in the left two plots. Using Eq. (11), the final conditional variance estimates do in fact show an asymmetry effect where negative shocks appear to correspond with higher conditional variances and positive shocks appear to correspond with lower conditional variances. In addition, we also estimate the semiparametric conditional variance on the entire sample uing all 1305 observations, where the news impact curves are also symmetric, the nonparametric components are all downwards

Estimated Parameters in the First Stage Estimation
and Absolute Percentage Change in the Variance

	ω_1	α_1	β_1	$|\%\Delta|$ due to NP
GARCH(1,1)	0.0382	0.1864	0.7481	12.06
	(0.0076)	(0.0328)	(0.0352)	
ARCH(1)	0.3725	0.3554		9.02
	(0.0229)	(0.0611)		
GARCH(1,1)	0.0407	0.2779	0.7032	15.8
(FULL)	(0.0069)	(0.0354)	(0.0284)	
ARCH(1)	0.4986	0.7443		27.53
(FULL)	(0.0296)	(0.0822)		

Table 2: *The table reports coefficients of the* ARCH(1) *and* GARCH(1, 1) *models with standard errors in parantheses. The results from the top panel are based on estimates obtained from the training data set,* $t = 1, \ldots, 925$, *whereas the results from the bottom panel are based on estimates obtained from the entire data set* $t = 1, \ldots, 1305$. ω_1 *is the intercept of the conditional variance,* α_1 *is the coefficient of the ARCH term, and* β_1 *is the coefficient of the GARCH term. The last column contains the estimates of the mean absolute percentage change of the conditional variance due to the nonparametric component, calculated by Eq. (15).*

sloping, and the final semiparametric conditional variance estimates all exhibit the existence of the leverage effect. On the other hand, if a parametric model is estimated, the model would fail to pick up the inherent leverage effect present in the data.

To investigate the first stage estimation step, Table 2 reports the coefficients and standard errors of the ARCH(1) and GARCH(1, 1) models. We have estimated these models based on the training data set as well as the full data set and the results are similar. A key takeaway from this table is that the coefficient the GARCH term is around 0.7 and is statistically significant at the 5% level in both sample sizes, indicating that there is a GARCH effect present in the data. In addition, all ARCH effects are siginicant at the 5% level.

Now, to have a better understanding of how estimating the nonparametric component can help estimate the conditional variance, we evaluate the mean absolute percentage change in the semiparametric conditional variance due to the nonparametric component by the following

$$|\%\Delta_{np}| = \frac{1}{T-1} \sum_{t=2}^{T} \left| \frac{\widehat{\sigma}^2_{sp,t} - \widehat{\sigma}^2_{p,t}}{\widehat{\sigma}^2_{p,t}} \right| \times 100 = \frac{1}{T-1} \sum_{t=2}^{T} \left| \widehat{\sigma}^2_{np,t} - 1 \right| \times 100. \tag{15}$$

The idea here is to see what percentage change is present of the semiparamtric estimate relative to the parametric estimate, and using the identity in Eq. (4), this percentage change can be thought of as the amount that we need to adjust the parametric component by. The absolute percentage change is used to take into consideration that the nonparametric component
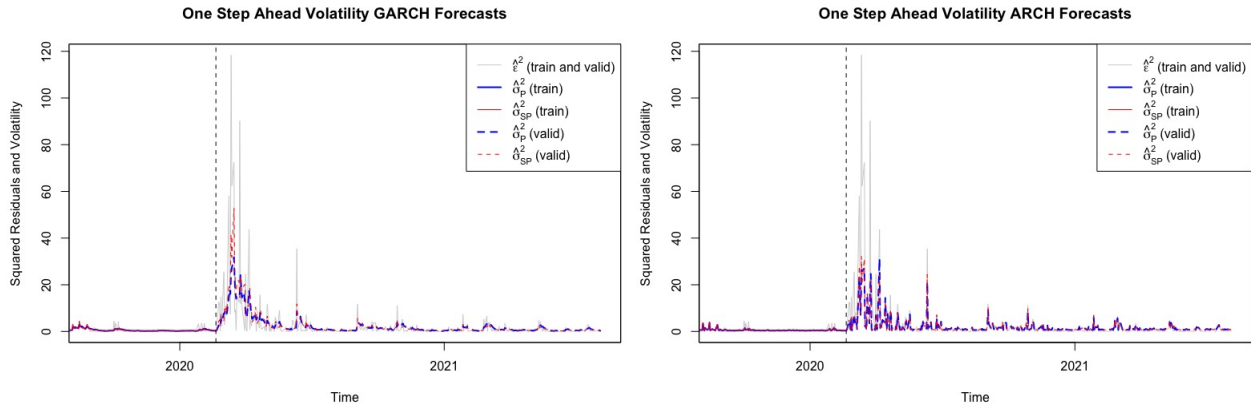
Figure 2: *The left plot shows the one step ahead GARCH and SPMLGARCH forecasts and the right plot shows the one step ahead ARCH and SPMLARCH forecasts. The forecast period is from $t = 926, \ldots, 1305$, or Febuary 21, 2021 to August 5, 2021. The vertical black dashed line indicates the start of the forecasting period. The red and blue curves indicate the semiparametric and parametric models respectively. The solid curves represent fitted values of the conditional variance based on the training data set and the dashed curves denote the forecasts based on the validation data set. The forecasts were calcuated using a rolling window scheme of window size 925.*

may in fact increase or decrease the parametric conditional variance. These mean percentage changes are reported in the last column of Table 2. For example, the GARCH$(1,1)$ model of the conditional variance, under the sample using $t = 1, \ldots, 925$ observations, is on average changed by 12% via the multiplicative adjustment from the nonparametric component. The ARCH(1) model is on average affected by a 9% change. And in the full sample these estimates are even larger, indicating that the estimation of the nonparametric component is vital in picking up nonlinearities or other signals from the conditional variance that the parametric models failed to pick up. Lastly, the estimate for the ARCH(1) model is the largest at around 28%. A possible explanation is that, under the full sample, which includes the highly volatile time period due to COVID-19, the parametric ARCH model may fail to pick up the these highly volatile peaks in the data, so the nonparametric part may have to uncover more of the signal from the data, and as a result the parametric component may change drastically.

Now, to evaluate forecasts, the validation set is used with a rolling window scheme. A window width of 925 observations is used to re-estimate a model and to forecast $h = 1$ and $h = 5$ periods ahead. First, we train a model, both parametric and nonparametric components, from the first 925 observations and forecast h periods ahead. For the nonparametric component, the hyperparameters, γ and λ, are found via leave one out cross validation (LOOCV). Then, we roll the window h periods forward and re-estimate the parametric component using the new data set. For the nonparametric component, $\widehat{\mathbf{c}}$ is re-estimated using the new data, however, instead of performing cross validation repeatedly to estimate the hyperparameters, γ and λ, these hyperparameters will be the ones found in the original sample using $t = 1, \ldots, 925$ observations, i.e. observations in the first window.

The one step ahead, $h = 1$, forecasts are plotted in Figure 2. The left plot contains the GARCH and SPMLGARCH forecasts and the right plot contains the ARCH and SPM-LARCH forecasts. Note that the last 506 observations are plotted to save space in the figures but the rolling window procedure uses the entire sample of 1305 observations. Since the true conditional variance is unknown we will use the squared residuals as a proxy. In order to do so, we estimate a parametric model, e.g. ARMA process, of the mean and obtain the residuals for the entire sample using Eq. (5) and square them as an estimate of the conditional variance. This will be used for reference when comparing the conditional variance estimates obtained from the four models.

As we can see from the data, the conditional variance is very high during March of 2020, depicted by the grey curves indicating the squared residuals. The red and blue curves indicate the semiparametric and parametric estimates of the conditional variance. All four models appear to pick up the peaks of the conditional variances especially around March of 2020, the most volatile period of the series. However, looking closely at the figures, both SPMLGARCH and SPMLARCH pick up more of the volatility peaks around this time, indicating that the semiparametric ML models may be better suited for forecasting S&P 500 returns.

As mentioned previously, the true conditional variance is unknown, and as a proxy we will use the squared residuals estimated from an ARMA process of the entire sample. For this data set, we found α_t to follow an MA(2), under the full sample. To assess forecasting ability, we will use the root mean squared forecast error (RMSFE) as

$$\text{RMSFE} = \sqrt{\frac{1}{T'} \sum_{t=1}^{T'} (\widehat{\sigma}_t^2 - \widehat{\varepsilon}_t^2)^2} \tag{16}$$

where $\widehat{\varepsilon}_t$ are the estimated residuals under the full sample. Table 3 reports the RMSFE's of the one and five steps ahead forecasts. For both $h = 1$ and $h = 5$, SPMLGARCH outperforms all other models in terms of having the lowest MSFE. In addition both the parametric and semiparametric GARCH models outperform the ARCH models, implying that GARCH effects may be present in the data, as we have seen from Table 2. In both ARCH and GARCH cases, the semiparametric counterparts perform better than the parametric ones. Overall, SPMLGARCH appears to be the best model for forecasting S&P 500 daily returns.

RMSFE's of Estimators for Forecasting Conditional Variance

	SPMLGARCH	GARCH	SPMLARCH	ARCH
One Step Ahead	8.5628	8.7587	9.6357	9.6594
Five Steps Ahead	8.9648	9.0916	10.0129	10.0471

Table 3: *The table reports RMSFE's for $h = 1$ and $h = 5$ steps ahead forecasts and for the following estimators: SPMLGARCH, SPMLARCH, GARCH, and ARCH.*

7   Conclusion

This paper proposes a semiparametric estimator for the conditional variance of a stochastic series. The estimator combines a parametric component and a nonparaemtric component via ML in a multiplicative way. The estimator is able to pick up asymmetric effects even when the parametric component can not. The ability to find asymmetric effects is directly due to the nonparametric ML estimation stage, where nonlinearities can be estimated via a kernel based machine learning algorithm. A metric is also created to show the importance of the nonparametric component by providing on average how much the parametric component is changed by. We show that the semiparametric estimator performs better in simulations, and in the empirical application, the proposed estimator is superior to the other considered models and is able to forecast future volatility given a conditioning set.

References

Borchers, H. W., pracma: Practical Numerical Math Functions. **2021**, R package version 2.3.3.

Bollerslev, T. Generalized Autoregressive Conditional Heteroskedasticity. *J. Econom.* **1986**, 307–327.

De Brebanter, K., De Brebanter, J., Suykens, J. A. K., De Moor, B. Approximate Confidence and Prediction Intervals for Least Squares Support Vector Regression. *IEEE Trans. Neural Netw. Learn. Syst.* **2011** *22(1)*, 110–120.

Ding, Z., Granger, C. W. J., Engle, R. F. A Long Memory Property of Stock Market Returns and a New Model. *J. Empir. Finance.* **1993**, *1(1), 83–106.*

Engle, R. F. Autoregressive Conditional Heteroskedasticity With Estimates of the Variance of UK Inflation. *Econometrica.* **1982**, 987–1008.

Engle, R. F., and Ng, V. K. Measuring and Testing the Impact of News on Volatility. *J. Finance.* **1993**, 1749–1778.

Ghalanos, A. rugarch: Univariate GARCH models. **2020**, R package version 1.4-4.

Glad, I. Parametrically Guided Non-parametric Regression. *Scand. J. Stat.* **1998**, *25*, 649–668.

Glosten, L. R., Jagannathan, R., and Runkle, D. On the Relationship Between the Expected Value and the Volatility of the Nominal Excess Return on Stocks. *J. Finance.* **1993**, *48*, 1779–1801.

Hainmueller, J. and Hazlett, C. Kernel Regularized Least Squares: Reducing Misspecification Bias with a Flexible and Interpretable Machine Learning Approach. *Polit. Anal.* **2014**, *22*, 143–168.

Hyndman, R., Athanasopoulos, G., Bergmeir, C., Caceres, G., Chhay, L., O'Hara-Wild, M., Petropoulos, F., Razbash, S., Wang, E., and Yasmeen, F. forecast: Forecasting functions for time series and linear models. **2020**, R package version 8.13.

Mishra, S., Su, L., Ullah, A. Semiparametric Estimator of Time Series Conditional Variance. *J. Bus. Econ. Stat.* **2010**, *28* , 256–274. Research Collection School Of Economics.

Nelson, D. B. Conditional Heteroskedasticity in Asset Returns: A New Approach. *Econometrica.* **1991**, *59(2)*, 347–370.

Warnes, G., Bolker, B., Gorjanc, G., Grothendieck, G., Korosec, A., Lumley, T., MacQueen, D., Magnusson, A., Rogers, J., and others. gdata: Various R Programming Tools for Data Manipulation. **2017**, R package version 2.18.0.

Wickham, H., François, R., Henry, L., and Müller, K. dplyr: A Grammar of Data Manipulation. **2021**, R package version 1.0.7.

Zeileis, A. and Grothendieck, G. zoo: S3 Infrastructure for Regular and Irregular Time Series. *J. Stat. Softw.* **2005**, *14(6)*, 1–27.

Ziegelmann, F., Nonparametric Estimation of Volatility Functions: The Local Exponential Estimator. *Econom. Theory.* **2002**, *18*, 985–991.