Estimation in the Presence of Heteroskedasticity of Unknown Form: A Lasso-based Approach

Emilio González-Coya*
Department of Economics, Boston University
and
Pierre Perron
Department of Economics, Boston University

October 24, 2022

Abstract

We study the Feasible Generalized Least-Squares (FGLS) estimation of the parameters of a linear regression model in the presence of heteroskedasticity of unknown form in the errors. We suggest a Lasso based procedure to estimate the skedastic function of the residuals. The advantage of using Lasso is that it can handle a large number of potential covariates, yet still yields a parsimonious specification. Using extensive simulation experiments, we show that our suggested procedure always provide some improvements in the precision of the parameter of interest (lower Mean-Squared Errors) when heteroskedasticity is present and is equivalent to OLS when there is none. It also performs better than previously suggested procedures. Since the fitted value of the skedastic function falls short of the true specification, we form confidence intervals using a bias-corrected version of the usual heteroskedasticity-robust covariance matrix estimator. These have the correct size and substantially shorter length than when using OLS. Our method is applicable to both cross-section (with a random sample) and time series models, though here we concentrate on the former.

Keywords: Feasible Generalized Least-Squares, Mean-Squared Error, Confidence Intervals, Non-parametric Methods, Linear Model

JEL Classification: C12, C13, C14, C21

*The authors gratefully acknowledge the computing support from the Boston University Shared Computing Cluster.

1 Introduction

If the error term in a regression exhibits heteroskedasticity, Ordinary Least Squares (OLS) is no longer efficient and the usual OLS standard errors are in general invalid. There are two proposed solutions to this problem: efficient estimation by Generalized (or Weighted) Least Squares (GLS or WLS), and OLS estimation using heteroskedasticity robust (HR) standard errors. The latter has been by far the most widely adopted route in practice following the seminal work of White (1980). This can be explained by several factors. First, the resulting inference is (asymptotically) valid in the presence of heteroskedasticity of unknown form, even if less efficient. Second, a common critique of GLS estimators is that "if the conditional variance model is a poor approximation or if the estimates of it are very noisy, [the weighted] estimators may have worse finite-sample properties than unweighted estimators"; Angrist and Pischke (2008) Section 3.4.1. Third, it can be difficult to obtain a good approximation to the skedastic function, making Feasible GLS (FGLS) a futile attempt for improvements. This consensus has not been unanimous. Leamer (2010) contends that researchers should be working to model the heteroskedasticity in order to determine whether sensible reweighting affects the estimates. In the context of a random sample for which only heteroskedasticity is a concern, Romano and Wolf (2017) shows that GLS can improve the efficiency of the estimates of the relevant parameters even when the skedastic function of the errors is misspecified, though it is assumed that the correct covariates are used. Even when the true form of the heteroskedasticity is unknown, HR standard errors can be used to base valid inference; e.g., DiCiccio et al. (2019).

Here, we focus on the linear model and propose a FGLS framework based on a non-parametric estimate of the skedastic function via Lasso. Note that we do not have in mind any oracle model, though it is possible to achieve a consistent estimate if the true variables and their transforms are in the candidate set. However, our aim is to be agnostic

about such knowledge and devise a method that is as robust as possible, yet still allow a reduction in the Mean-Squared Error (MSE) over OLS. We use the fact that Lasso is able to provide a good in-sample fit to the skedastic function. This should be useful to achieve an improvement in the MSE of the estimate of interest. Also, Lasso generally yields a parsimonious model, which can help avoid inflating the MSE when no heteroskedasticity is present. Simulations show that this is the case, contrary to other methods proposed earlier. However, since the skedastic function is not consistently estimated, there is a need to further correct the variance estimate of the FGLS estimator using a HR version.

Given our aim, we shall not consider its consistency properties or any other large sample distribution. Instead, we report extensive simulation experiments that mimics various cases of practical interest. In all cases reported (and others not included), the FGLS procedure based on Lasso never performs worst than OLS in terms of MSE, whether or not heteroskedasticity is present, and for all types of specifications for the covariates used to approximate the skedastic function. Hence, there are no costs, only benefits. All that is needed for improvements is to include at least one covariate (and some non-linear transforms) correlated with the true variable influencing the skedastic function. This is unlike most previously proposed methods and shows some strong robustness, alleviating most concerns advanced to discredit FGLS in favor of OLS plus HR.

However, since the Lasso-based procedure is not intended to estimate any oracle (or true) skedastic function, only a useful approximation, it is important to apply a further heteroskedasticity robust correction for the construction of the confidence intervals. This is not a defect of the suggested procedure. In finite samples, no procedure can achieve what infeasible GLS can do, even with the correct specification.

Our work follows and borrows from several prior studies. As in Robinson (1987), we believe that unless the form of heteroskedasticity is of interest in itself, it may be better to avoid attempting to parametrize it. He suggested estimating the skedastic function of the

residuals by a Nearest Neighbor nonparametric regression. Kuk (1999) studied an iterative FGLS estimation based on a nonparametric estimate of the skedastic function via Locally Weighted Polynomial regressions. Miller and Startz (2019) proposed a FGLS procedure based on Support Vector Regression. Romano and Wolf (2017) use FGLS combined with HR standard errors. Their procedure allows for valid inference and lead to efficiency gains over OLS, even if the proposed skedastic function is misspecified. They also proposed an Adaptive Least Squares estimator: first carry out a test of conditional heteroskedasticity based on the same covariates entering the skedastic function used for weighting the data. If the test rejects, use FGLS; otherwise, stick with OLS. Independent of the outcome of the test, always use HR standard errors. To address the unsatisfying finite sample performance of the asymptotic approximation of the FGLS estimator, DiCiccio et al. (2019) proposed to use resampling methods. They established that the wild and pairs bootstrap approximations to the sampling distribution of the FGLS estimator are consistent under unknown heteroskedasticity, and deliver better finite sample properties than standard asymptotic approximations. Another concern that has discouraged the use of FGLS is that the estimator may be less efficient than OLS when the model used to estimate the skedastic function is misspecified. To address this problem, they proposed using a linear combination of the OLS and FGLS estimators, which is shown to be asymptotically at least as efficient as both the OLS and WLS estimator. Using our approach, no such device will be needed. For improved finite-sample performance, we consider a bias-corrected version of the usual heteroskedasticity-robust covariance matrix estimator based on Rothenberg (1988) and used by Miller and Startz (2019). Our suggested procedure then delivers estimates with high precision and confidence intervals having the correct size and substantially shorter lengths than with OLS. Our method is applicable to both cross-sections (with a random sample) and time series models, though here we concentrate on the former. Perron and González-Coya (2022) discusses FGLS procedures in more depth for the time series

case in which correcting for both serial correlation and heteroskedasticity is needed.

We study and compare the finite sample performance of the FGLS estimator based on Lasso with a family of FGLS estimators based on other non-parametric methods that have been previously used in the literature to estimate the variance: Nearest Neighbor (Robinson, 1987), Support Vector Regression (Miller and Startz, 2019), Local Linear (Fan and Yao, 1998) and Random Forest (Breiman, 2001). Our simulation results suggest that Lasso and SVR outperform the other methods. Hence, in the main text we concentrate on results pertaining to Lasso, SVR and, for comparisons, two versions of FGLS proposed by Romano and Wolf (2017). Results for Nearest Neighbor, Local Linear and Random Forest were noticeably inferior and, accordingly, are relegated to an online supplement.

The paper is organized as follows. Section 2 describes the main setup. Section 3 provides a detailed exposition of the non-parametric methods used to estimate the skedastic function. Details about estimating the confidence intervals are presented in Section 4. The simulation results are discussed in Section 5. Section 6 provides brief concluding remarks.

2 General Setup

We are interested in the estimation of and inference about the parameters in linear models when the errors may be heteroskedastic, given by

$$y = X\beta + u, \tag{1}$$

where y is an $T \times 1$ vector of outcomes of interest, $X = [x_1', ..., x_T']'$ is an $T \times k$ matrix of predictors, β is a $k \times 1$ parameter vector, and $u = [u_1, ..., u_T]'$ is a $T \times 1$ vector of unobserved error terms with covariance matrix W. We assume throughout that u exhibits heteroskedasticity of unknown form given by the skedastic function $\nu(\eta) = [\nu_1(\eta), ..., \nu_T(\eta)] = diag[W]$, which is non-constant and for which we may not have any prior or theoretical guidance as to its specification. Here, $\eta = [\eta_1, ..., \eta_T]$ with η_t a $d_\eta \times 1$ vector of covariates affect-

ing the skedastic function. It is also assumed that the matrix $X'X$ is invertible and that $p\lim_{T\to\infty} T^{-1}X'X = Q_X$, a fixed positive-definite matrix. We further assume that the regressors are exogenous so that $E[u|X] = 0$.

We focus on the FGLS estimate of model (1) using the following steps: 1) Estimate (1) via OLS with $\hat{\beta}_{OLS}$ the estimate of β and $\hat{u}_t = y_t - \hat{\beta}'_{OLS}x_t$, the residuals; 2) Estimate a model $\hat{u}_t^2 = v_t(z)$, with $z = [z'_1, ..., z'_T]'$ where z_t $(t = 1, ..., T)$ is a $1 \times d$ vector consisting of a pre-specified set of candidate covariates that may include some or all elements of x_t and/or η_t, or transformations of them. The elements of z_t and η_t may overlap, but this is unknown *a priori*. Denote the fitted values by $\tilde{u}_t^2$; 3) Construct the FGLS estimator,

$$\hat{\beta}_{FGLS} = \left(\sum_{t=1}^{T} x_t x'_t \tilde{u}_t^{-2}\right)^{-1} \sum_{t=1}^{T} x_t y_t \tilde{u}_t^{-2},$$

or equivalently $\hat{\beta}_{FGLS} = (X'\tilde{W}^{-1}X)^{-1}X'\tilde{W}^{-1}y$, where $\tilde{W}$ is a diagonal matrix with entries $\tilde{w}_{tt} = \tilde{u}_t^2 = \tilde{v}_t(z)$ $(t = 1, ..., T)$. The potential benefit of this procedure is that the estimator remains consistent but with a good estimate of the skedastic function it is asymptotically, and in finite samples, more efficient than OLS. Still, it is very difficult in practice to obtain estimates that are close to what could be achieved using the true unknown skedastic function. Hence, when constructing standard errors, we further correct for possible remaining heteroskedasticity. The details of the suggested procedure are in Section 4.

3 Estimation of the Skedastic Function

Several approaches have been proposed for the estimation of the skedastic function. The simplest is to estimate a linear model. The following are standard suggestions for the estimation of $v(z)$. Wooldridge (2010) suggests using a linear regression with the same covariates as those in the model, i.e., $z_t = x_t$, while Romano and Wolf (2017) suggest using the log of the absolute value of regressors. This leads to the following two versions:

- WLS-S1: $\tilde{v}_t(z) = \exp(\tilde{c} + \tilde{\gamma}'[\log(z_{1,t}) + ... + \log(z_{d,t})])$.

- WLS-S2: $\tilde{v}_t(z) = \exp(\tilde{c} + \tilde{\gamma}'[z_{1,t} + ... + z_{d,t}])$.

They suggested using $z_t = x_t$, the same covariates included as regressors. However, we shall allow extra covariates in our simulations.

Nonparametric approaches to estimating $v(\cdot)$ include Kernel Regression (Carroll, 1982), Nearest Neighbors (Robinson, 1987), Series Estimation (Kuk, 1999) and more recently, Support Vector Regression (SVR) (Miller and Startz, 2019). We follow the non-parametric approach since it can handle various forms of nonlinearities and allows incorporating various extraneous variables to fit the skedastic function. Therefore, it is best suited to avoid misspecifications due to an incorrect choice of the covariates or the functional form.

Our results will point to the advantage of using Lasso. The main reason is fairly intuitive. Lasso allows as input variables an arbitrary number of regressors, which, in principle, can be much larger than the sample size. Hence, a good in-sample fit for the skedastic function is more likely, which is ultimately what matters. Other non-parametric approaches such as Kernel Regressions, Nearest Neighbors and Series Estimation generally requires a tight pre-specification of the nature (and number) of covariates. This makes such methods more sensitive to misspecification, even though they can capture various forms of non-linearities quite well. The SVR approach is a close contender though on balance Lasso is more robust. Also, SVR is suitable for prediction but the estimated coefficients lack economic meaning; Lasso delivers a sparse model with few non zero coefficients, which can inform us about the nature of the true skedastic function. Below, we describe the non-parametric methods entertained as candidates, Lasso and SVR. We also considered the methods of Nearest-Neighbors, Local Linear as well as a Random Forest. These delivered inferior results and, hence, the results are reported in an online supplement.

3.1 Lasso

Lasso is a non-parametric estimation method first proposed by Tibshirani (1996). It selects regressors amongst a potentially large set by imposing the ℓ_1 penalty on their size. With candidate values z_{tj} $(j = 1, ..., d)$, where d can be very large, the candidate model is:

$$\log(\hat{u}_t^2) = \gamma_0 - \sum_{j=1}^{d} z_{tj}\gamma_j + v_t.$$

The Lasso coefficients minimize a penalized residual sum of squares,

$$\hat{\gamma}^{LAS} = \arg\min_{\gamma} \sum_{t=1}^{T} \left(\log(\hat{u}_t^2) - \gamma_0 - \sum_{j=1}^{d} z_{tj}\gamma_j \right)^2,$$

subject to $\sum_{j=1}^{d} |\gamma_j| \leq \vartheta$, where ϑ is some threshold, adaptively chosen to minimize an estimate of the expected prediction error, and $\gamma = (\gamma_0, \gamma_1, ..., \gamma_d)'$. We can also write the Lasso problem in the equivalent Lagrangian form

$$\hat{\gamma}^{LAS} = \arg\min_{\gamma} \left\{ (1/2) \sum_{t=1}^{T} \left(\log(\hat{u}_t^2) - \gamma_0 - \sum_{j=1}^{d} z_{tj}\gamma_j \right)^2 + \lambda \sum_{j=1}^{d} |\gamma_j| \right\}.$$

Lasso forces the coefficients to be equally penalized. We can, however, assign different weights to different coefficients. If the weights are data-dependent and properly chosen, this can enhance the properties of Lasso, in particular when the the irrelevant covariates are highly correlated with the relevant covariates. To that effect, Zou (2006) considered a weighted or adaptive Lasso given by

$$\hat{\gamma}^{\mathrm{A}} = \arg\min_{\gamma} \left\{ (1/2) \sum_{t=1}^{T} \left(\log(\hat{u}_t^2) - \gamma_0 - \sum_{j=1}^{d} x_{tj}\gamma_j \right)^2 + \lambda \sum_{j=1}^{d} \hat{\omega}_j |\gamma_j| \right\}, \tag{2}$$

where $\hat{\omega}_j = |\hat{\gamma}_j|^{-\psi}$, $\psi > 0$ and $\hat{\gamma}$ is a root-T-consistent estimator of γ. For linear predictions with high dimensional data, where d is very large and the irrelevant covariates are potentially highly correlated with the relevant ones, other methods have been proposed for improved performance, such as two stage estimators. These include the Lasso-OLS hybrid (Efron et al., 2004), Relaxed Lasso (Meinshausen, 2007), and post-Lasso estimator.

Liu and Yu (2013) proposed a Lasso procedures in conjunction with a Ridge or a modified Least Squares estimator. The latter is close to Lasso-OLS (Belloni and Chernozhukov, 2013), which uses OLS in the second stage. We performed extensive simulation experiments about the finite sample properties of these various Lasso-based estimators with the MSE and coverage rate of the resulting FGLS estimate as the criteria of interest. We found that the Adaptive Lasso performed best. Henceforth, "Lasso" refers to this particular version. The implementation of Adaptive Lasso to obtain a fit to the skedastic function is as follows.

1) Compute the first-step estimate of γ as the solution to the Ridge regression problem:

$$\hat{\gamma}^{\text{ridge}} = \arg\min_{\gamma} \left\{ (1/2) \sum_{t=1}^{T} \left(\log(\hat{u}_t^2) - \gamma_0 - \sum_{j=1}^{d} z_{tj}\gamma_j \right)^2 + \lambda^r \sum_{j=1}^{d} \gamma_j^2 \right\}$$

where λ^r is selected via cross-validation. Note that, in principle, one can use any method that sets non-zero coefficients, such as OLS, Ridge regression or SVR. The Lasso estimator is not suitable since it generates sparse models (i.e., with only few nonzero estimates) and thus the resulting weights would be indeterminate for some j. In settings with a large set of covariates, OLS is not recommended as the variability of the estimates is too large. SVR with a linear kernel could be used to compute the first-step estimates. Simulations suggest that the results are similar to those with the Ridge regression.

2) Compute the weights as $\hat{\omega}_j = |\hat{\gamma}_j^{ridge}|^{-\psi}$. The Adaptive Lasso estimates are then

$$\hat{\gamma}^{\text{A}} = \arg\min_{\gamma} \left\{ (1/2) \sum_{t=1}^{T} \left(\log(\hat{u}_t^2) - \gamma_0 - \sum_{j=1}^{d} x_{tj}\gamma_j \right)^2 + \lambda^A \sum_{j=1}^{d} |\hat{\gamma}_j^{ridge}|^{-\psi} |\gamma_j| \right\}$$

where the two tuning parameters, λ^A and ψ are selected via the following K-cross-validation method: a) Fix a vector of L possible values for ψ; we use $L = 6$ and $\psi^c = (0, 0.25, 0.5, 0.75, 1, 2)$. b) Fix a partition for the K-fold cross-validation, i.e., split the data into K roughly equal-sized parts. We use $K = 10$. Let $\kappa : \{1, \ldots, T\} \mapsto \{1, \ldots, K\}$ be an indexing function that indicates the partition to which observation t is allocated to by the randomization. c) For every ψ_i^c, compute the optimal cross-validated λ_i^A and the mean cross-validated error.

For the kth part, we fit the model to the other $K-1$ parts of the data, and calculate the prediction error of the fitted model when predicting the kth part of the data. We do this for $k = 1, \ldots, K$ and combine the K estimates of the prediction error. Denote by $\hat{f}_i^{-k}(z)$ the fitted function, computed with the kth part of the data removed and using ψ_i^c. Then the cross-validation estimate of the prediction error is

$$\mathrm{CV}(\hat{f}_i) = T^{-1} \sum_{t=1}^{T} \mathrm{MSE}\left(\log(\hat{u}_t^2), \hat{f}_i^{-\kappa(t)}(z)\right).$$

Let λ_i^A be the value that minimizes the mean cross-validated error $\mathrm{CV}(\hat{f}_i)$. Denote the minimum mean cross-validated error for λ_i^A and ψ_i^c as $\mathrm{CV}(\lambda_i^A, \psi_i^c)$. Note that the resulting minimum cross-validated error $\mathrm{CV}(\lambda_i^A, \psi_i^c)$ is comparable among $i = 1, \ldots, L$ as the K-fold randomization of step (b) is the same for every i. d) The cross-validated pair $(\lambda^{A*}, \psi^{c*})$ used is the one that minimizes $\mathrm{CV}(\lambda_i^A, \psi_i^c)$ for $i = 1, \ldots, L$.

Note that the Lasso penalty $\sum_{j=1}^{d} |\gamma_j|$ makes the solutions nonlinear in $\log(\hat{u}_t^2)$, and there is no closed form expression. Computing the Lasso solution is a quadratic programming problem and efficient algorithms are available (e.g., the Least Angle Regression). Intuitively, Lasso and its variants are akin to doing a continuous subset selection. Note that we do not have in mind any oracle model. Our aim is to be agnostic about such knowledge and to try to devise a method as robust as possible that allows a reduction in the MSE over OLS. Since the skedastic function is, in general, not consistently estimated, there is a need to further correct the variance estimate of the FGLS estimator.

3.2 Support Vector Regression

A Support Vector Regression approach was proposed by Drucker et al. (1997) and adopted by Miller and Startz (2019) to construct a FGLS estimate. It seeks to minimize a penalized function of the residuals. In this formulation, the SVM is a regularized function estimation problem, where the coefficients γ are shrunk toward zero. The use of kernels allows the SVR

to model nonlinearities present in the data. In the specific form of the SVR we consider, the ϵ-insensitive loss function for the residuals, which imposes no penalty on residuals smaller than $|\epsilon|$ and penalizes residuals linearly as they exceed $|\epsilon|$. The penalty function used is the ℓ_2 norm, which reduces the variance of the model and, hence, overfitting. More specifically, SVR solves the following problem:

$$\min_{\gamma=[\gamma_1,\ldots,\gamma_d]} \sum_{t=1}^{T} L_\epsilon \left(\log\left(\hat{u}_t^2\right) - g\left(z_t, \gamma\right) \right) + \lambda \sum_{j=1}^{d} \gamma_j^2,$$

where $g\left(z_t, \gamma\right) = \sum_{j=1}^{d} \gamma_j h_j\left(z_t\right) = \gamma' h\left(z_t\right)$, and $L_\epsilon(r) = 0$ if $|r| < \epsilon$ and $|r| - \epsilon$, otherwise. Here, $L_\epsilon\left(\cdot\right)$ is the ϵ-insensitive loss function, λ is a tuning parameter selected by cross-validation and $h_j(\cdot)$ is a basis function which maps the input data into d dimensions. The the solution depends only on the inner product of the basis functions $h(z_i)$ and $h(z_t)$, which is equivalent to a kernel function $K(z_i, z_t)$ applied to the original data vectors z_i and z_t. The default kernel used is the Gaussian one, so that $h_j(\cdot)$ are also called radial basis. This formulation connects SVR to kernel regressions.

4 Constructing Confidence Intervals

In order to construct confidence intervals for the parameter β of interest, introducing some finite sample refinements can be beneficial. Here, we describe the particular form adopted. Out treatment follows Miller and Startz (2019) and Rothenberg (1988). Consider the following Taylor expansion to approximate the additional variance introduced by the estimation of the weights when performing FGLS. Denoting the cth element of the FGLS and GLS estimators by $\hat{\beta}_{FGLS,c}$ and $\hat{\beta}_{GLS,c}$, respectively, we have:

$$\begin{aligned} & \operatorname{Var}\left(\sqrt{T}\left(\hat{\beta}_{FGLS,c} - \hat{\beta}_{GLS,c}\right)\right) \\ \approx & \sum_{i=1}^{T}\sum_{j=1}^{T} C' A X W^{-1} \left[W^{-1}\left(I_T - H_{GLS}\right)\right]_{ij} \operatorname{Cov}\left(\tilde{u}_i^2, \tilde{u}_j^2\right) W^{-1} X A C, \end{aligned} \tag{3}$$

where $A = (X'W^{-1}X)^{-1}$, $H_{GLS} = X(X'W^{-1}X)^{-1}X'W^{-1}$ and C is a vector containing a one in position c and zeros elsewhere. The ijth element of the term in square brackets can be interpreted as the influence of the jth outcome on the weighted GLS residual for observation i. Note that when $i = j$ the method used to estimate $\tilde{u}_i^2$ contributes to the variance of $\hat{\beta}_{FGLS,c}$ in proportion to the influence of the i^{th} element on its own weighted GLS residual. Equation (3) suggests that inference using FGLS estimates should account for the difference between $\hat{\beta}_{FGLS,c}$ and $\hat{\beta}_{GLS,c}$. Romano and Wolf (2017) partly incorporate this issue, illustrating that various forms of heteroskedasticity-robust standard errors will provide asymptotically valid inference even if the estimated weights do not correspond to the inverse of the skedastic function $v_t(z)$. We focus on the estimate of the asymptotic variance of the FGLS estimator:

$$\text{Avar}\left(\hat{\beta}_{FGLS}\right) = \left(T^{-1}X'\hat{W}^{-1}X\right)^{-1}\hat{\Omega}\left(T^{-1}X'\hat{W}^{-1}X\right)^{-1}, \tag{4}$$

where $\hat{\Omega} = T^{-1}X'\hat{\Sigma}X$ and $\hat{\Sigma}$ is a diagonal matrix whose construction is discussed below, see (6). In particular, Romano and Wolf (2017) suggest using

$$\text{Avar}\left(\hat{\beta}_{FGLS}\right) = (T^{-1}X'\hat{W}^{-1}X)^{-1}\hat{\Omega}_1(T^{-1}X'\hat{W}^{-1}X)^{-1}, \tag{5}$$

where $\hat{\Omega}_1 = T^{-1}\sum_{t=1}^{T}(\tilde{\varepsilon}_t^2 x_t x_t'/\left(\tilde{u}_t^2\right)^2)$, with $\tilde{u}_t^2$ the fitted value of the skedastic function at observation t using any of the method. Here, $\tilde{\varepsilon}_t$ are the so-called HC3 OLS residuals, suggested by MacKinnon and White (1985), involving a correction shown to improve the finite sample accuracy defined by $\tilde{\varepsilon}_t = \hat{u}_t^2/(1 - h_{t,OLS})^2$, with $\hat{u}_t$ the OLS residuals from regression (1) and $h_{t,OLS} = [X(X'X)^{-1}X]_{tt}$. Miller and Startz (2019) claim that while the approach proposed by Romano and Wolf (2017) offers consistent estimation of standard errors, inference in finite samples may still suffer. They proposed an alternative standard error correction that blends the proposal from Romano and Wolf (2017) with insights from the approximation (3). Their correction requires only an estimate of the degrees of freedom used in the skedastic function estimation. Specifically, they proposed the following revised

estimator $\hat{\Sigma}_{tt}^{FGLS}$ entering in the construction of $\hat{\Omega} = T^{-1}X'\hat{\Sigma}X$ in (4):

$$\hat{\Sigma}_{tt}^{FGLS} = \frac{\hat{u}_{t,FGLS}^2}{\left(\tilde{u}_t^2\right)^2}\left(\frac{1}{\left(1-h_{t,FGLS}\right)^2} + 4\frac{h_{t,OLS}}{k}\hat{df}\right), \tag{6}$$

where $\hat{u}_{FGLS} = [\hat{u}_{1,FGLS}, ..., \hat{u}_{T,FGLS}]'$ are the estimated residuals from the FGLS regression, i.e., $\hat{u}_{FGLS} = y - \hat{\beta}_{FGLS}X$. $\hat{df}$ is an estimate of the degrees of freedom used in the estimation of the weights. For Lasso, the number of nonzero coefficients is an unbiased estimate for the degrees of freedom (Zou et al., 2007), while for SVR it is $\hat{df} = |\epsilon_+| + |\epsilon_-|$ where ϵ_+ and ϵ_- are the set of observations with residual equal to ϵ and $-\epsilon$ (Gunter and Zhu, 2007), where ϵ is the sensitivity parameter for the loss function $L_\epsilon(\cdot)$. Also, $h_{t,FGLS} = [X(X'\hat{W}^{-1}X)^{-1}X\hat{W}^{-1}]_{tt}$. The first term in parenthesis of (6) is the standard multiplier used in the so-called HC3 standard errors, MacKinnon and White (1985). The second term is a correction which addresses the fact that estimated weights are used in FGLS; see Miller and Startz (2019) for details. The confidence intervals for the kth coefficient is then obtained using $\hat{\beta}_{FGLS,k} \pm z_{1-\alpha/2}SE(\hat{\beta}_{FGLS_k})$,where $z_{1-\alpha/2}$ is the $1-\alpha/2$ quantile of the normal distribution and $SE\left(\beta_{FGLS,k}\right) := \left(Avar\left(\beta_{FGLS,k}\right)\right)^{1/2}$, with $Avar\left(\beta_{FGLS,k}\right)$ defined in (4) and $\hat{\Omega} = T^{-1}X'\hat{\Sigma}^{FGLS}X$, with $\hat{\Sigma}^{FGLS}$ a diagonal matrix with entries defined in (6). To summarize, for WLS-S1-S2, we use the covariance matrix (5), while for Lasso and SVR, we use (4) with $\hat{\Sigma}$ replaced by $\hat{\Sigma}_{tt}^{FGLS}$ as defined by (6).

5 Simulation Results

In this section, we consider the linear model (1) with independent errors. We present simulation results for the FGLS estimator

$$\hat{\beta}_{FGLS} = (X'\tilde{W}^{-1}X)^{-1}X'\tilde{W}^{-1}y,$$

where $\tilde{W}$ is a diagonal matrix with entries $\tilde{w}_{tt} = \tilde{u}_t^2 = \hat{v}_t(z)$, the predicted values obtained from the various procedures to fit the skedastic function $v_t(\eta)$. As in DiCiccio et al. (2019)

and Romano and Wolf (2017), we consider the response variable $\log(\max\{\hat{u}_t^2, \delta^2\})$, where $\delta = 0.1$ is some small positive number to avoid dealing with residuals that are nearly zero. The implementation of the various procedure are done using the follows packages: a) our Lasso implementation in `R` is based on `GLMNET` package, Friedman et al. (2010); b) for SVR, the `svm` function from the `e1071` package in `R` with a Gaussian kernel; Meyer et al. (2021). The parameters ϵ, λ and ψ are chosen via cross-validation.

5.1 Base-case Results

We shall compare our results to those of Romano and Wolf (2017). Hence, our first set of simulations follow the set-up in their Section 5. We consider a univariate regression model

$$y_t = \alpha + x_t\beta + \sqrt{\nu_t(\eta)}\varepsilon_t, \tag{7}$$

based on an *i.i.d.* sample $\{x_t\}_{t=1}^T$ where $x_t \sim U[1,4]$, $\varepsilon_t \sim N(0,1)$ and ε_t is independent of x_t. The sample size is $T \in \{100, 200, 400\}$. The parameters of interest is β and we set $(\alpha, \beta) = (1, 1)$. We consider the same four parametric specifications for the skedastic function $\nu_t(\cdot)$ as in Romano and Wolf (2017), which involve the single regressor $\eta_t = x_t$:

•Power function: $\nu_t(x)_1 = x_t^\gamma$, $\gamma \in \{0, 1, 2\}$;

•Squared log function: $\nu_t(x)_2 = [\log(x_t)]^2$;

•Exponential of a second-degree polynomial: $\nu_t(x)_3 = \exp(0.2x_t + 0.2x_t^2)$;

•The step function $\nu_t(x)_4 = 1$, for $1 \leq x_t < 2$, 4 for $2 \leq x_t < 3$ and 9 for $3 \leq x_t < 4$.

The performance measure is the empirical mean squared error (eMSE),

$$eMSE(\tilde{\beta}) := B^{-1}\sum_{b=1}^{B}(\tilde{\beta}_b - \beta)^2,$$

where $B = 10,000$ is the number of replications and $\tilde{\beta}_b$ the estimate of β in the bth one.

In Table 1, we present simulation results for model (7) which are directly comparable with those in Tables C.1-C.3 from Romano and Wolf (2017). We present the eMSE

of the estimators of β relative ot OLS. The response variable is $\log(\max\{\hat{u}_t^2, \delta^2\})$. For WLS-S1-S2, $z_t = \{x_t\}$, while for Lasso and SVR, we consider the input vector $z_t = (1, x_t, \log(x_t)^2, x_t^2, \cos(x_t), \cos(2x_t))$. The results are presented for each combination of heteroskedasticity scenario (row groups) and the estimator used for the skedastic function (columns). We compare 5 estimators in the following columns: (1) is the (unfeasible) GLS estimator based on the known skedastic function, (2) and (3) are the WLS-S1-S2 based FGLS from Romano and Wolf (2017), (4) and (5) are the FGLS estimators with the skedastic function estimated using Lasso and SVR, respectively.

The first panel of Table 1 pertains to the case of homoskedastic errors, for which OLS and infeasible GLS are best. We would expect the FGLS estimators to have increased variance and be less precise. This is the case, in general, though the results show that Lasso is able to achieve the same precision. Hence, nothing is lost in terms of precision using Lasso even when not needed. The other methods show some increases in eMSE, especially SVR. When the data generating process is characterized by heteroskedasticity, we clearly notice advantages of using a FGLS approach. The next groups of results in Table 1 correspond to moderate ($\nu_t(x)_1$ with $\gamma = 1$, and $\nu_t(x)_4$) and severe heteroskedasticity ($\nu_t(x)_1$ with $\gamma = 2$, $\nu_t(x)_2$ and $\nu_t(x)_3$). For all the skedastic functions considered, the FGLS estimator based on Lasso outperforms OLS, the WLS-S1-S2 and SVR-based FGLS estimates. The highest reductions in eMSE are observed for $v(x)_2$ and $v(x)_3$ for which the Lasso-based FGLS estimate achieves a reduction of up to 55% in eMSE relative to OLS, though all methods fall short of what could be achieved using infeasible GLS. For the other cases, the reductions in eMSE is close to what could be achieved using infeasible GLS, though the reduction relative to OLS are not as extreme, namely of the order of 8 to 30%. Overall, the highest eMSE reductions are observed for the FGLS estimate based on Lasso and, for some skedastic specifications, these are close to what could be achieved using the (unfeasible) GLS estimate.

Table 1: eMSE of estimators of β relative to OLS.

	GLS	WLS-S1	WLS-S2	Lasso	SVR
$\nu(x) = 1$					
$T = 100$	1.00	1.03	1.03	1.01	1.14
$T = 200$	1.00	1.02	1.02	1.00	1.11
$T = 400$	1.00	1.01	1.01	1.00	1.08
$\nu(x) = x$					
$T = 100$	0.88	0.91	0.91	0.92	1.00
$T = 200$	0.89	0.91	0.91	0.92	0.99
$T = 400$	0.90	0.91	0.91	0.92	0.97
$\nu(x) = x^2$					
$T = 100$	0.67	0.68	0.69	0.70	0.77
$T = 200$	0.67	0.68	0.69	0.70	0.75
$T = 400$	0.68	0.69	0.70	0.70	0.73
$\nu(x) = [\log(x)]^2$					
$T = 100$	0.37	0.54	0.56	0.54	0.56
$T = 200$	0.30	0.47	0.53	0.46	0.48
$T = 400$	0.31	0.48	0.54	0.46	0.46
$\nu(x) = \exp(0.2x + 0.2x^2)$					
$T = 100$	0.44	0.49	0.49	0.49	0.52
$T = 200$	0.43	0.48	0.48	0.45	0.49
$T = 400$	0.44	0.48	0.47	0.45	0.48
$\nu(x) = \nu(x)_4$					
$T = 100$	0.68	0.74	0.74	0.74	0.79
$T = 200$	0.70	0.74	0.74	0.74	0.79
$T = 400$	0.71	0.74	0.74	0.73	0.77

In Table 2 , we present the coverage rates and average length of the confidence intervals for β. Here, and throughout, the nominal significance level is 95%. For WLS-S1-S2, we use the covariance matrix (5), while for Lasso and SVR, we use (4) with $\hat{\Sigma}$ replaced by $\hat{\Sigma}_{tt}^{FGLS}$ as defined by (6). In general, the coverage rates are near 95% for all methods, though, overall, the Lasso-based FGLS has the smallest departures. The SVR-based FGLS can be conservative, especially when $T = 100$. Also, the lengths of the confidence intervals are smaller when using Lasso. They are noticeably larger when using SVR. Note that in the absence of homoskedasticity (the first group of results in Table 2), the standard error correction delivers confidence intervals with coverage probabilities near 95% and the average length of the confidence interval is close to that of OLS with HR standard errors.

Again, the results show that there is no noticeable cost in using FGLS based on Lasso when there is no heteroskedasticity, even for small samples with $T = 100$.

5.2 Robustness Experiments

We now provide further results to address commonly voiced concerns about using FGLS. For instance, Angrist and Pischke (2008) contend that if the skedastic function is misspecified or if its estimate is very noisy, FGLS estimators may have worse finite-sample properties than unweighted estimators. Responses to this critique have been expressed in the literature; Romano and Wolf (2017) proposed a FGLS procedure that leads to efficiency gains over OLS, even if the skedastic function is misspecified, though they used the correct covariate. A common characteristic of this literature, which includes, among others, Robinson (1987) and Miller and Startz (2019), is that they assume the variable affecting the variance function to be known. This is undesirable as most often we may not know anything about which variables drive the skedastic function. To address this important problem, and other issues, in this section we provide simulation evidence to shed light on the performance of FGLS relative to OLS for various specifications of the covariates z_t. In all cases, many covariates and some transforms are included, which are correlated or not with the true covariate. To cover cases that are as close as possible to those of practical interest, we consider the following five cases: Case 1: Baseline scenario, the covariates include the variable that drives the heteroskedasticity with the correct functional form. Case 2: The covariates include the variable that drives the heteroskedasticity but not the correct functional form. Case 3: The covariates do not include the variable that drives the heteroskedasticity but are correlated with it. Case 4: The covariates do not include the variable that drives the heteroskedasticity and only one is correlated with it. Case 5: The covariates do not include the variable that drives the heteroskedasticity and none of them is correlated with it.

For the simulations, we consider model (7) with the same specifications of the skedastic

Table 2: Coverage rate and average length of confidence intervals for β.

		OLS	WLS-S1	WLS-S2	Lasso	SVR
$T = 100$						
$\nu(x) = 1$	Coverage	0.96	0.95	0.95	0.95	0.97
	Length	0.47	0.46	0.47	0.47	0.51
$\nu(x) = x$	Coverage	0.96	0.95	0.95	0.96	0.97
	Length	0.74	0.70	0.74	0.72	0.76
$\nu(x) = x^2$	Coverage	0.96	0.95	0.95	0.95	0.97
	Length	1.28	1.07	1.17	1.10	1.14
$\nu(x) = [\log(x)]^2$	Coverage	0.96	0.95	0.95	0.96	0.98
	Length	0.44	0.31	0.33	0.32	0.32
$\nu(x) = \exp\left(0.2x + 0.2x^2\right)$	Coverage	0.96	0.95	0.95	0.95	0.97
	Length	1.90	1.35	1.53	1.34	1.52
$\nu(x) = \nu(x)_4$	Coverage	0.96	0.95	0.95	0.95	0.97
	Length	1.04	0.89	0.99	0.91	0.93
$T = 200$						
$\nu(x) = 1$	Coverage	0.95	0.95	0.95	0.95	0.98
	Length	0.32	0.32	0.32	0.33	0.41
$\nu(x) = x$	Coverage	0.95	0.94	0.94	0.94	0.98
	Length	0.51	0.49	0.49	0.50	0.61
$\nu(x) = x^2$	Coverage	0.95	0.95	0.95	0.95	0.99
	Length	0.89	0.74	0.75	0.76	0.89
$\nu(x) = [\log(x)]^2$	Coverage	0.96	0.96	0.96	0.96	0.98
	Length	0.31	0.21	0.23	0.21	0.25
$\nu(x) = \exp\left(0.2x + 0.2x^2\right)$	Coverage	0.95	0.95	0.95	0.95	0.98
	Length	1.33	0.93	0.90	0.92	1.06
$\nu(x) = \nu(x)_4$	Coverage	0.95	0.94	0.95	0.95	0.98
	Length	0.72	0.62	0.62	0.63	0.73
$T = 400$						
$\nu(x) = 1$	Coverage	0.96	0.95	0.95	0.95	0.97
	Length	0.23	0.23	0.23	0.23	0.27
$\nu(x) = x$	Coverage	0.96	0.95	0.95	0.96	0.97
	Length	0.36	0.34	0.34	0.35	0.40
$\nu(x) = x^2$	Coverage	0.96	0.95	0.95	0.95	0.97
	Length	0.63	0.52	0.52	0.53	0.60
$\nu(x) = [\log(x)]^2$	Coverage	0.96	0.95	0.95	0.96	0.98
	Length	0.21	0.15	0.16	0.15	0.16
$\nu(x) = \exp\left(0.2x + 0.2x^2\right)$	Coverage	0.96	0.95	0.95	0.95	0.97
	Length	0.93	0.65	0.63	0.63	0.71
$\nu(x) = \nu(x)_4$	Coverage	0.96	0.95	0.95	0.95	0.97
	Length	0.51	0.44	0.44	0.44	0.49

function $v(\cdot)$ as in the previous section. We also set $\alpha = 1$, $\beta = 1$, $T = 200$ and the number of replications is 10,000 (the results for $T = 100$ and $T = 400$ are similar). The candidate covariates z_t to model the skedastic function include some transforms of the regressors x_t as well as $k_w = 20$ covariates w_{tj} $(j = 1, ..., k_w)$ having a $U(1, 4)$ distribution. The results are presented Table 3, where the layout is as in Table 1 .

Since we want to be agnostic about the functional form of the possible non-linearities in the skedastic function, we adopt a flexible form that can approximate such non-linearities by using trigonometric transforms given by $\{\cos(w_t), \cos(2w_t)\}$. We tried using additional terms such as $\cos(3w)$, but the results were the same. However, adding the term $\cos(2w)$ helped increase the precision. Hence, two terms appear enough for the cases considered, though one could use additional ones if desired. The use of such orthogonal polynomials is useful to avoid multicolinearity problems. Things are quite different for the WLS-S1-S2 based FGLS estimators. Since these are based on some unconstrained regression, one is bound to use much fewer covariates. The precision decreases notably when including too many. Our specifications below are guided by this feature.

Case 1. In the first experiment, we assume that the covariates w_{tj} $(j = 1, \ldots, k_w)$ are correlated with x_t via the specification: $w_{tj} = \phi_{tj} x_t + (1 - \phi_{tj}) q_{tj}$ with the following distributions for the $T \times 1$ vectors $q_j \sim U(1, 4)$ and $\phi_j \sim Be(\rho_j)$ with $\rho_j \sim U(0.4, 0.8)$; hence w_{tj} is uniform and the correlation between w_{tj} and x_t is ρ_j. For the Lasso and SVR based estimates of the skedastic function, the included covariates are $z_t = \{x_t, x_t^2, \log(x_t)^2, w_t, \cos(w_t), \cos(2w_t)\}$, a total of 63 covariates. For the WLS-S1-S2 based FGLS estimates, $z_t = \{x_t, w_t\}$ (21 covariates). The results are provided in Table 3 . The FGLS estimator based on Lasso provides the highest reduction in eMSE relative to OLS among almost all the estimators considered for all the skedastic functions. The reduction in eMSE can be substantial (e.g., 48%), though none of the methods achieve a reduction close to that obtained using the infeasible GLS procedure, especially when the

correct functional form is not included; e.g., $\nu_t(x)_4$ with the step function. Moreover, under homoskedasticity ($\nu(x) = x^\gamma$ with $\gamma = 0$) FGLS based on Lasso has an eMSE similar to OLS, while all other methods induces a noticeable increase in eMSE. The increase is especially severe for the WLS-S1-S2 based methods, which show a 50% increase in eMSE, compared to 11% for SVR and 2% for Lasso. They also show a substantial increase when the heteroskedasticity is mild, e.g., $\upsilon(x) = x$.

Case 2. In the second experiment, the additional covariates w_t are correlated with x_t as in Case 1. For the FGLS estimators based on Lasso and SVR, the included covariates are $z_t = \{x_t, w_t, \cos(w_t), \cos(2w_t)\}$. Hence, the total number of covariates is 61. The covariates x_t^2 and $\log(x_t)^2$ are excluded so that the correct functional form is not included for any of the specifications of the skedastic function. For the WLS-S1-S2 based FGLS estimators the covariates are again $z_t = \{x_t, w_t\}$ (21 covariates). The results are provided in Table 3, panel 2. The reduction in eMSE among all the estimators and heteroskedastic cases is qualitatively the same as in the previous case, though Lasso allows an even larger reduction in eMSE compared to SVR. This suggests that including the correct functional form of the skedastic function does not necessarily improve the precision of the FGLS estimator. Other covariates that are correlated with x_t can provide a useful fit to the skedastic function. What transpires from these results is that including many covariates that are correlated with the variable influencing the skedastic function along with some transforms (e.g., cosine functions) that can fit some non-linear functions works well in providing a good fit.

Case 3. In the third experiment, the included covariates w_t are correlated with x_t as in Cases 1 and 2, but we do not include the variable x_t that influences the skedastic function. For the FGLS estimators based on Lasso and SVR, the included covariates are $z_t = \{w_t, \cos(w_t), \cos(2w_t)\}$ (the total number is 60). For the WLS-S1-S2 based FGLS estimators $z_t = \{w_t\}$. The results are provided in Table 3, panel 3. In this case, the reduction in eMSE for all estimators and heteroskedastic specification is similar to the

case in which we include the variable x_t that influences the skedastic function. The Lasso-based method is the most precise, in general, whether heteroskedasticity is present or not. This is likely the set of results of most interest in practice. It states that as long as the specified covariates to model the skedastic function are correlated with the true variables that generates it, one can do as well as when the correct covariates are included, even the correct functional form. This shows strong robustness as in practice it is difficult to have knowledge of the specific covariates generating the true skedastic function.

Case 4. In the fourth experiment, we assume that only one covariate, w_{t1}, is correlated with x_t and specified by $w_{t1} = \phi_{t1}x_t + (1 - \phi_{t1})q_{t1}$ with $q_1 \sim U(1, 4)$, $\phi_1 \sim Be(\rho_1)$ and $\rho_1 \sim U(0.4, 0.8)$. The remaining covariates w_{tj} $(j = 2, \ldots, k_w)$ are generated as independent $U(1, 4)$ variables uncorrelated with x_t. For the FGLS estimators based on Lasso and SVR, the included covariates are $z_t = \{w_t, \cos(w_t), \cos(2w_t)\}$ (the total number is 60). For the WLS-S1-S2 based FGLS estimators $z_t = \{w_t\}$. The results are provided in Table 3, panel 4. Even when not including the covariate that characterizes the skedastic function and just one of the covariates is correlated with it, we still observe eMSE reductions when using FGLS relative to OLS of the order of up to 15%. Moreover, the FGLS estimates based on Lasso and SVR clearly outperform the WLS-S1-S2 based FGLS estimators in all cases. Note that for all cases, the WLS-S1-S2 based FGLS estimators are noticeably less precise than OLS whether heteroskedasticity is present or not, by as much as 40% in the latter case and up to 36% in the former. The estimator based on SVR is more precise than the Lasso-based with $\nu(x)_2$, though at the expense of larger eMSE with the other cases.

Case 5. Here we assume that none of the variables w_t are correlated with x_t and again generated as independent $U(1, 4)$ variables. For the FGLS estimators based on Lasso and SVR, the included covariates are $z_t = \{w_t, \cos(w_t), \cos(2w_t)\}$ (the total number is 60). For the WLS-S1-S2 based FGLS estimators $z_t = \{w_t\}$. The results are provided in Table 3, panel 5. First note that the WLS-S1-S2 based FGLS estimators are noticeably less precise

than OLS for all specifications. For the Lasso-based FGLS estimator we still observe eMSE reductions relative to OLS of the order of up to 14%. Again, the estimator based on SVR is more precise than the Lasso-based with $\nu(x)_2$. Overall, we achieve very little gain in precision, as expected. However, with Lasso and SVR, there is no loss.

Remark 1 One may argue that the comparison between the WLS-S1-S2 and Lasso or SVR methods is not fair because more covariates are included for the latter. But this is a central argument in favor of adopting a Lasso-based approach. It can deal with a (virtually) arbitrary large number of covariates, since it discards most of them and only keeps what is relevant given the (finite) data. The WLS-S1-S2 methods consider unrestricted regressions so that only a relatively small subset of parameters can be included, otherwise the variability of the estimates is too large and results in unreliable inference.

5.2.1 The Confidence Intervals: Coverage Rates and Average Lengths.

Table 4 present simulation results for the coverage rates and average lengths of the confidence intervals for Cases 1-5 when $T = 200$ (the results with $T = 100$, 400 are similar). The nominal level of the coverage rate is 95%. For WLS-S1-S2, we use the covariance matrix (5), while for Lasso and SVR, we use (4) with $\hat{\Sigma}$ replaced by $\hat{\Sigma}_{tt}^{FGLS}$ (eq. 6).

For Cases 1 and 2, with the regressor x_t included in z_t, the Lasso-based method has coverage rates nearest to the nominal level. The estimates based on SVR are slightly liberal whereas the WLS-S1-S2 based FGLS estimates show the largest liberal size distortions. The length of the CI are roughly comparable. For Cases 3 to 5, with the regressor x_t not included in z_t, the Lasso-based estimator performs at least as well as OLS, namely short CI with coverage rates near 95%. Again, the estimates based on WLS-S1-S2 and SVR have short CI but with coverage rates below 95%.

Overall, the Lasso-based estimates have coverage rates nearest to the 95% nominal

Table 3: eMSE of estimators of β relative to OLS; $T = 200$.

	GLS	WLS-S1	WLS-S1	Lasso	SVR
Case 1					
$\nu(x)_1 = x^\gamma$					
$\gamma = 0$	1.00	1.50	1.48	1.02	1.11
$\gamma = 1$	0.81	1.30	1.28	0.95	0.99
$\gamma = 2$	0.57	0.99	0.95	0.76	0.81
$\nu(x)_2 = [\log(x)]^2$	0.29	0.64	0.67	0.59	0.57
$\nu(x)_3 = \exp(0.2x + 0.2x^2)$	0.35	0.72	0.64	0.52	0.63
$\nu(x)_4$	0.58	1.02	0.96	0.79	0.83
Case 2					
$\nu(x)_1 = x^\gamma$					
$\gamma = 0$	1.00	1.35	1.35	1.03	1.11
$\gamma = 1$	0.84	1.20	1.20	0.98	1.03
$\gamma = 2$	0.60	0.92	0.92	0.77	0.89
$\nu(x)_2 = [\log(x)]^2$	0.32	0.59	0.66	0.58	0.64
$\nu(x)_3 = \exp(0.2x + 0.2x^2)$	0.37	0.69	0.63	0.52	0.70
$\nu(x)_4$	0.62	0.99	0.97	0.81	0.90
Case 3					
$\nu(x)_1 = x^\gamma$					
$\gamma = 0$	1.00	1.33	1.33	1.03	1.06
$\gamma = 1$	0.79	1.19	1.19	0.96	0.98
$\gamma = 2$	0.57	0.92	0.91	0.76	0.84
$\nu(x)_2 = [\log(x)]^2$	0.30	0.61	0.67	0.56	0.61
$\nu(x)_3 = \exp(0.2x + 0.2x^2)$	0.36	0.71	0.64	0.53	0.67
$\nu(x)_4$	0.59	0.99	0.97	0.77	0.84
Case 4					
$\nu(x)_1 = x^\gamma$					
$\gamma = 0$	1.00	1.40	1.38	1.03	1.11
$\gamma = 1$	0.79	1.25	1.24	1.00	1.06
$\gamma = 2$	0.56	1.13	1.15	0.92	0.95
$\nu(x)_2 = [\log(x)]^2$	0.30	1.36	1.24	1.01	0.72
$\nu(x)_3 = \exp(0.2x + 0.2x^2)$	0.35	1.11	1.09	0.85	0.89
$\nu(x)_4$	0.58	1.26	1.22	0.95	0.97
Case 5					
$\nu(x)_1 = x^\gamma$					
$\gamma = 0$	1.00	1.39	1.38	1.04	1.06
$\gamma = 1$	0.79	1.39	1.39	1.00	1.00
$\gamma = 2$	0.57	1.35	1.35	0.94	0.92
$\nu(x)_2 = [\log(x)]^2$	0.30	1.35	1.34	1.01	0.69
$\nu(x)_3 = \exp(0.2x + 0.2x^2)$	0.36	1.29	1.32	0.86	0.83
$\nu(x)_4 = \nu(x)_4$	0.59	1.35	1.37	0.95	0.92

level, similar to OLS. However, the lengths of the Lasso-based CI are substantially shorter than those obtained with OLS. The other methods have coverage rates that are, in general, below the nominal level and in most cases with larger lengths. Hence, the Lasso-based approach yields accurate inference, similar to OLS. Given that it also provides substantial improvements in the precision of the estimates (lower eMSE) and shorter lengths of the confidence intervals, it dominates all other contenders.

5.3 General Discussion of the Simulation Results

We note the following features from the simulations reported and others that we performed but do not report given space constraints.

• There is a trade off in the performance of Lasso between the number of covariates w_t and the functional forms of the covariates included (e.g., linear, quadratic, trigonometric). Given a sample size T and a fixed number of covariates k_w, including an extra functional form of w_t (e.g., additional cosine terms) did not improve the precision of the estimate.

• There is a computational trade off between the number of included covariates w_t and the functional forms of the covariates. In general, the Lasso implementation in `R` can handle a lot of covariates, but issues of convergence arise when several functional forms are included as covariates. In particular, including polynomial terms in w_t causes problems. This is, however, not surprising as there is substantial correlation across the different polynomial terms so that a near multicollinearity problem is present.

• The best strategy that we found is to include a large number of regressors w and a small number of functional forms of w. In order to solve the near multi-collinearity issue that arises when using standard polynomials, we suggest using orthogonal polynomials such as Chebyshev polynomials of the first kind, i.e., $T_n(\cos\theta) = \cos(n\theta)$.

Table 5 present the Lasso variable selection, i.e., the number of times that a regressor has a non-zero coefficient, for 10,000 replications with $T = 200$ and $k_w = 20$, the number of

Table 4: Coverage rates and average length of confidence intervals; $T = 200$.

		OLS	WLS-S1	WLS-S1	Lasso	SVR	OLS	WLS-S1	WLS-S1	Lasso	SVR
		Case 1					Case 2				
$\nu(x)_1 = 1$	Coverage	0.95	0.88	0.88	0.95	0.95	0.95	0.89	0.89	0.95	0.95
	Length	0.32	0.31	0.31	0.33	0.36	0.32	0.31	0.31	0.32	0.36
$\nu(x)_1 = x$	Coverage	0.95	0.87	0.89	0.95	0.94	0.95	0.90	0.90	0.95	0.94
	Length	0.51	0.46	0.46	0.50	0.52	0.51	0.46	0.46	0.50	0.52
$\nu(x)_1 = x^2$	Coverage	0.95	0.87	0.88	0.94	0.92	0.94	0.89	0.90	0.95	0.92
	Length	0.89	0.71	0.72	0.78	0.77	0.89	0.70	0.71	0.78	0.77
$\nu(x)_2$	Coverage	0.95	0.89	0.91	0.94	0.94	0.95	0.92	0.92	0.95	0.94
	Length	0.31	0.21	0.22	0.23	0.23	0.30	0.21	0.22	0.23	0.23
$\nu(x)_3$	Coverage	0.94	0.85	0.87	0.95	0.89	0.95	0.87	0.89	0.95	0.88
	Length	1.33	0.89	0.87	0.96	0.93	1.32	0.89	0.87	0.96	0.92
$\nu(x)_4$	Coverage	0.95	0.88	0.90	0.95	0.92	0.95	0.91	0.91	0.95	0.92
	Length	0.72	0.59	0.60	0.64	0.63	0.72	0.59	0.60	0.64	0.63
		Case 3					Case 4				
$\nu(x)_1 = 1$	Coverage	0.95	0.89	0.90	0.94	0.95	0.94	0.89	0.89	0.94	0.92
	Length	0.32	0.31	0.31	0.33	0.36	0.32	0.32	0.32	0.33	0.32
$\nu(x)_1 = x$	Coverage	0.95	0.90	0.90	0.95	0.94	0.94	0.90	0.90	0.94	0.90
	Length	0.51	0.47	0.47	0.50	0.52	0.51	0.49	0.50	0.51	0.45
$\nu(x)_1 = x^2$	Coverage	0.94	0.89	0.90	0.95	0.92	0.94	0.90	0.91	0.95	0.86
	Length	0.89	0.72	0.72	0.79	0.77	0.89	0.83	0.84	0.87	0.69
$\nu(x)_2$	Coverage	0.94	0.90	0.90	0.94	0.94	0.94	0.91	0.91	0.95	0.88
	Length	0.31	0.21	0.22	0.23	0.23	0.31	0.31	0.30	0.31	0.21
$\nu(x)_3$	Coverage	0.94	0.88	0.89	0.94	0.88	0.94	0.89	0.90	0.95	0.81
	Length	1.33	0.91	0.89	0.97	0.92	1.33	1.19	1.19	1.25	0.89
$\nu(x)_4$	Coverage	0.94	0.89	0.90	0.95	0.92	0.94	0.90	0.90	0.95	0.85
	Length	0.72	0.61	0.61	0.65	0.63	0.72	0.71	0.71	0.71	0.55
		Case 5									
$\nu(x)_1 = 1$	Coverage	0.95	0.89	0.89	0.94	0.92					
	Length	0.32	0.32	0.32	0.33	0.32					
$\nu(x)_1 = x$	Coverage	0.95	0.88	0.89	0.94	0.91					
	Length	0.51	0.50	0.51	0.51	0.47					
$\nu(x)_1 = x^2$	Coverage	0.94	0.90	0.90	0.94	0.88					
	Length	0.89	0.88	0.88	0.87	0.73					
$\nu(x)_2$	Coverage	0.94	0.90	0.91	0.94	0.90					
	Length	0.31	0.31	0.31	0.31	0.23					
$\nu(x)_3$	Coverage	0.94	0.89	0.90	0.95	0.85					
	Length	1.33	1.28	1.28	1.25	1.00					
$\nu(x)_4$	Coverage	0.94	0.89	0.89	0.94	0.86					
	Length	0.72	0.72	0.72	0.71	0.58					

covariates in w_t. The simulations are based on Case 1. We present the average frequency with which each variable is selected among the 20 covariates. The results with 10 covariates are similar and not reported. The exact selection procedure is as follows. First, we compute a 3-dimensional array of dimensions (S, J, K) where S is the number of replications, $J = 6$ is the number of skedastic functions and K is the total number of covariates (in this case 63). For every pair (s, j), we keep the vector of size K of the variables selected by Lasso in replication s and skedastic function j. We condensate this information in a 6×6 matrix (number of skedastic functions and number of covariates in groups). We have 3 single covariates (x_t, x_t^2, $\log(x_t)^2$) and 3 families of multidimensional covariates $(w_t, \cos(w_t), \cos(2w_t))$. For the covariates $k = 1, 2, 3$, we sum (over $s = 1, ..., S$) the number of times variable k is selected for the skedastic function j. For the covariates $k = 4, \ldots, 63$, we first sum (over $s = 1, \ldots, S$) the number of times variable k is selected for the skedastic function j. Then, we take either the maximum or the mean within each of the three groups $(w, \cos(w), \cos(2w))$.

The following observations are noteworthy. a) Lasso does not necessarily select the correct functional of the skedastic function. See for example column x for $\nu(x) = x^\gamma, \gamma = 1$, column x^2 for $\nu(x) = x^\gamma, \gamma = 2$, and column $\log(x)^2$ for $\nu(x) = [\log(x)]^2$. The correct functional forms are selected only in the high heteroskedastic cases (the last two rows). b) Correlated covariates w_t are often selected for all skedastic functions. c) Chebyshev polynomials terms are relevant; the covariates $\cos(w_t)$ and $\cos(2w_t)$ are selected almost with the same frequency for all skedastic specifications. One way to interpret these results in light of the fact that the Lasso procedure delivers the most precise estimates is the following. In any given finite sample, the realized skedastic function is a noisy version of the true skedastic function. Lasso appears best suited to fit the relevant finite sample pattern of the heteroskedasticity, which is what matters for FGLS to improve upon OLS.

The commonly voiced concern that using FGLS or WLS can do more harm than good,

Table 5: Lasso variable selection, $T = 200$, w includes 20 covariates; average percentage across 10,000 replications.

	x	x^2	$\log(x)^2$	w	$\cos(w)$	$\cos(2w)$
			Average			
$\nu(x) = x^\gamma$						
$\gamma = 0$	0.0	0.1	0.1	2.0	2.0	3.0
$\gamma = 1$	3.5	1.5	1.1	5.3	5.2	8.6
$\gamma = 2$	24.0	4.9	5.3	9.1	9.0	16.4
$\nu(x) = [\log(x)]^2$						
	61.6	0.8	0.4	10.	10.9	27.6
$\nu(x) = \exp(0.2x + 0.2x^2)$						
	18.3	38.0	15.3	13.2	12.9	10.0
$\nu(x) = \nu(x)_4$						
	16.5	3.9	19.5	6.1	6.0	19.8

probably arises from a concern about Cases 3-5, especially Case 4, in which case the covariates used do not include the correct covariates nor the functional form affecting the skedastic function. This concern is well founded if one restricts attention to the standard linear regression model to estimate the skedastic function as done when using the WLS-S1-S2 based FGLS suggested by Romano and Wolf (2017); see, in particular the results for Case 4 for which this approach yields much worse results than OLS whether heteroskedasticity is present or not. However, our results show that broadening the class of estimators eliminates this concern. Note that in all cases reported (and others not included), the FGLS procedure based on Lasso does not perform worse than OLS, whether or not heteroskedasticity is present, and for all types of specifications for the covariates used to approximate the skedastic function. There are only a few minor exceptions for which using Lasso implies, at most, a small 4% increase in eMSE when no heteroskedasticity is present, while important reductions occur when it is present. As documented from Case 5, even if we completely miss any of the features of the variables influencing the skedastic function (i.e., all variables are uncorrelated), we can still do as well as OLS and sometimes better.

However, since the Lasso-based procedure is not intended to estimate any oracle (or

true) skedastic function, only a useful approximation, it is important to apply a further heteroskedasticity robust correction for the construction of the confidence intervals. This is not a defect of the suggested procedure. In finite samples, no procedures can achieve what infeasible GLS can do, even when the correct specification is adopted. Hence, a correction is always desirable. The correction suggested yields confidence intervals with the correct size and with shorter lengths than OLS when heteroskedasticity is present. When none is present the results are (virtually) the same as with OLS. Hence, there are (virtually) no costs, only benefits, in using the suggested procedure.

5.4 An Empirically-based Experiment

We now evaluate the performance of the various FGLS-based procedures using simulations calibrated to some actual empirical implementation. To this end we revisit the widely used data set of Boston's housing prices originally created and analyzed by Harrison Jr and Rubinfeld (1978). We use this dataset as it allow us to compare our results with Romano and Wolf (2017) and Miller and Startz (2019). The goal is to explain the logarithm of house prices ($\log(price)$) for communities in the Boston area in the year 1970. We first note that the dataset includes 506 observations for the following variables: *prices*: the median price of a house in US$; *nox*: the nitrogen oxide concentration (parts per million); *dist*: the weighted distance (miles) from employment centers; *rooms*: the mean rooms per house; *stratio*: the mean student-to-teacher ratio; *crime*: crimes committed per capita; *radial*: an accessibility index to radial highways; *proptax*: property tax per $1000; *lowstat*: the percentage of people "lower status"; *zn*: the proportion of residential land zoned for lots over 25,000 sq.ft; *indus*: the proportion of non-retail business acres per town; *chas*: a Charles River dummy variable (taking value 1 if the tract bounds the river; 0 otherwise); *age*: the proportion of owner-occupied units built prior to 1940; $b = 1000(B-0.63)^2$, where B is the proportion of black households within a town or community.

Following Wooldridge (2012), the model used by Romano and Wolf (2017) and Miller and Startz (2019) is the following estimated by OLS (standard errors in parenthesis):

$$\log(price) = \underset{(0.18)}{2.00} - \underset{(0.12)}{0.96}\log(nox) - \underset{(0.04)}{0.13}\log(dist) + \underset{(0.02)}{0.25}rooms - \underset{(0.01)}{0.05}stratio + u; R^2 = 0.59. \tag{8}$$

Given the dataset available and the specification used, it appears to us that the model is misspecified in the sense that relevant variables are omitted from the regression. Moreover some of these omitted variables are likely to be correlated with the included regressors. This is important in the current context. Omitted variables will a) induce heteroskedastic errors even if none is present in the true errors; b) if correlated with the included regressors the OLS estimates will be inconsistent. To that effect, we performed variable addition tests (e.g., Engle (1982)). This suggested the following specification:

$$\begin{aligned}\log(price) = \beta_0 + \beta_1\log(nox) + \beta_2\log(dist) + \beta_3 rooms + \beta_4 stratio \\ +\beta_5 chas + \beta_6\log(crime) + \beta_7\log(radial) + \beta_8\log(proptax) \\ +\beta_9\log(b) + \beta_{10}\log(lowstat) + u;\ R^2 = 0.77.\end{aligned} \tag{9}$$

The OLS estimates are presented in Table 6, all of which are significant and the R^2 increases to 0.77. The coefficients estimates are different from those obtained with the smaller specification: the coefficient on *nox* reduces by 70%, the one for *rooms* decreases by 68%, while the other two are roughly the same.

The estimates obtained using Lasso-FGLS are presented in the second column of Table 6. The estimates are significant, except for one, and the R^2 is increased to 0.82. Comparing the OLS and FGLS estimates for the full model, some important difference emerge: the estimates of the coefficients on *nox* and *rooms* are higher, that on *chas* and *lowstat* lower and *crime* is no longer significant. Hence, the fact that the parameter estimates are noticeably different, in conjunction with a higher R^2 for the Lasso-FGLS, suggest that meaningful difference can be obtained using FGLS.

Table 6: Estimation results for the extended specification; Model (9) with log(*price*) as the dependent variable.

	OLS	Lasso-FGLS
(Intercept)	4.61^{***}	3.82^{***}
	(0.33)	(0.26)
$\log(nox)$	-0.28^{***}	-0.34^{***}
	(0.10)	(0.08)
$\log(dist)$	-0.17^{**}	-0.16^{***}
	(0.03)	(0.03)
$rooms$	0.08^{***}	0.16^{***}
	(0.02)	(0.02)
$stratio$	-0.03^{***}	-0.03^{***}
	$(5E-3)$	$(3E-3)$
$chas$	0.11^{**}	0.07^{*}
	(0.04)	(0.03)
$\log(crime)$	-0.03^{*}	-0.01
	(0.01)	(0.01)
$\log(radial)$	0.09^{***}	0.07^{***}
	(0.02)	(0.02)
$\log(proptax)$	-0.21^{***}	-0.21^{***}
	(0.04)	(0.03)
$\log(b)$	0.06^{***}	0.08^{***}
	(0.01)	(0.01)
$\log(lowstat)$	-0.39^{***}	-0.28^{***}
	(0.02)	(0.02)
R^2	0.77	0.82

$^{***}p < 0.001$; $^{**}p < 0.01$; $^{*}p < 0.05$,, $p = p\text{-}value$

In the remaining of this section, we use the extended model estimated by OLS to analyze the properties of the various FGLS procedures via simulations. This follows Romano and Wolf (2017) and Miller and Startz (2019) but with the extended model (9) instead of their smaller model (8). We use the same wild bootstrap approach as in Romano and Wolf (2017), generating 10,000 replicates. In this approach, the true value of β for the artificial data is the OLS estimate. Thus, the eMSE is calculated based on deviations of the FGLS estimates from the OLS estimate in the original data. For the Lasso and SVR-based FGLS, the input vector to estimate the skedastic function is

$$z = \{1, w, w^2, \log(w), \cos(w), \cos(2w), \cos(3w)\},$$

where w is a vector that includes all variables in the data set even if not included in the regression, i.e., $w = \{x, zn, indus, age\}$, where

$$\begin{aligned} x &= \{\log(nox), \log(dist), rooms, stratio, chas, crime, \\ &\quad \log(radial), \log(proptax), \log(lowstat), \log(b)\}. \end{aligned}$$

For the FGLS estimates based on WLS-S1-S2, $z = \{w\}$. The results presented in Tables 7 show that the performance of the Lasso-based FGLS estimator is superior to all others, though the SVR-based FGLS estimator is a close contender. When constructing confidence intervals, all methods yield a coverage rate near the nominal 95% level. However, in the majority of cases, the average length is smaller with Lasso. These simulation results conform to those presented in Sections 5.1 and 5.2.

6 Conclusions

For the standard linear model with heteroskedastic errors, we proposed a Feasible Generalized Least Squares (FGLS) framework based on a non-parametric estimate of the skedastic function via Lasso. Our aim was to use the fact that Lasso is able to provide a good in-sample fit to the skedastic function. This turned out to be useful to achieve an improvement in the Mean-Squared Error of the estimate of interest. Also, Lasso generally yields a parsimonious model, which can help avoid inflating the MSE when no heteroskedasticity is present. We showed via simulations that this is the case, contrary to other methods proposed earlier. However, since the skedastic function is not consistently estimated, there is a need to further correct the variance estimate of the FGLS estimator. In all cases reported (and others not included), the FGLS procedure based on Lasso never performed worse than OLS (with very minor exceptions), whether or not heteroskedasticity is present, and for all types of specifications for the covariates used to approximate the skedastic function. It also provides confidence intervals with accurate coverage rate and shorter lengths. Hence,

Table 7: eMSE relative to OLS, coverage rate and average length of confidence intervals of estimators of β_j in Model (9)

		OLS	WLS-S1	WLS-S2	Lasso	SVR
	MSE	-	0.75	0.77	0.60	0.64
β_0	Coverage	0.95	0.96	0.93	0.94	0.94
	Length	1.46	1.45	1.20	1.09	1.33
	MSE	-	0.72	0.70	0.61	0.64
β_1	Coverage	0.95	0.95	0.95	0.96	0.96
	Length	0.47	0.41	0.34	0.36	0.41
	MSE	-	0.67	0.70	0.59	0.60
β_2	Coverage	0.95	0.96	0.93	0.95	0.95
	Length	0.15	0.15	0.12	0.11	0.13
	MSE	-	0.63	0.72	0.52	0.58
β_3	Coverage	0.94	0.95	0.92	0.94	0.93
	Length	0.10	0.10	0.08	0.08	0.09
	MSE	-	0.98	0.92	0.78	0.78
β_4	Coverage	0.96	0.95	0.94	0.95	0.96
	Length	0.02	0.02	0.02	0.01	0.02
	MSE	-	0.62	0.68	0.55	0.59
β_5	Coverage	0.95	0.97	0.93	0.95	0.95
	Length	0.15	0.15	0.11	0.11	0.13
	MSE	-	0.67	0.58	0.52	0.52
β_6	Coverage	0.95	0.95	0.94	0.95	0.96
	Length	0.05	0.07	0.06	0.03	0.04
	MSE	-	0.84	0.85	0.72	0.75
β_7	Coverage	0.95	0.96	0.95	0.95	0.96
	Length	0.07	0.14	0.13	0.06	0.07
	MSE	-	0.85	0.85	0.70	0.71
β_8	Coverage	0.95	0.95	0.94	0.94	0.95
	Length	0.15	0.07	0.07	0.13	0.15
	MSE	-	0.97	0.95	0.82	0.79
β_9	Coverage	0.96	0.95	0.94	0.94	0.93
	Length	0.07	0.14	0.11	0.06	0.07
	MSE	-	0.61	0.69	0.54	0.60
β_{10}	Coverage	0.95	0.95	0.94	0.94	0.94
	Length	0.14	0.13	0.13	0.10	0.12

the are no costs, only benefits, in using the suggested procedure. All that is needed for improvements is to include at least one covariate (and some non-linear transforms) correlated with the true variable influencing the skedastic function. This shows some strong robustness of the method suggested, alleviating most of the concerns advanced to discredit using FGLS and instead follow the OLS plus robust variance correction approach. Our results should indeed encourage researchers to use the FGLS procedure suggested. All the evidence suggest that much can be gained in precision without costs for reliable inference.

References

Angrist, J. D. and J.-S. Pischke (2008). *Mostly Harmless Econometrics: An Empiricist's Companion*. Princeton University Press, Princeton, NJ.

Belloni, A. and V. Chernozhukov (2013). Least Squares after model selection in high-dimensional sparse models. *Bernoulli 19*(2), 521–547.

Breiman, L. (2001). Random Forests. *Machine Learning 45*(1), 5–32.

Carroll, R. J. (1982). Adapting for heteroscedasticity in linear models. *The Annals of Statistics 10*(4), 1224–1233.

DiCiccio, C. J., J. P. Romano, and M. Wolf (2019). Improving Weighted Least Squares inference. *Econometrics and Statistics 10*, 96–119.

Drucker, H., C. J. Burges, L. Kaufman, A. Smola, and V. Vapnik (1997). Support Vector Regression Machines. *Advances in Neural Information Processing Systems 9*, 155–161.

Efron, B., T. Hastie, I. Johnstone, and R. Tibshirani (2004). Least Angle Regression. *The Annals of Statistics 32*(2), 407–499.

Engle, R. F. (1982). A general approach to Lagrange multiplier model diagnostics. *Journal of Econometrics 20*(1), 83–104.

Fan, J. and Q. Yao (1998). Efficient estimation of conditional variance functions in stochastic regression. *Biometrika 85*(3), 645–660.

Friedman, J., T. Hastie, and R. Tibshirani (2010). Regularization paths for generalized linear models via Coordinate Descent. *Journal of Statistical Software 33*(1), 1–22.

Gunter, L. and J. Zhu (2007). Efficient computation and model selection for the Support Vector Regression. *Neural Computation 19*(6), 1633–1655.

Harrison Jr, D. and D. L. Rubinfeld (1978). Hedonic housing prices and the demand for clean air. *Journal of Environmental Economics and Management 5*(1), 81–102.

Kuk, A. Y. (1999). Nonparametrically Weighted Least Squares estimation in heteroscedastic linear regression. *Biometrical Journal 41*(4), 401–410.

Leamer, E. E. (2010). Tantalus on the road to asymptopia. *Journal of Economic Perspectives 24*(2), 31–46.

Liu, H. and B. Yu (2013). Asymptotic properties of Lasso+ mLS and Lasso+ Ridge in sparse high-dimensional linear regression. *Electronic Journal of Statistics 7*, 3124–3169.

MacKinnon, J. G. and H. White (1985). Some heteroskedasticity-consistent covariance matrix estimators with improved finite sample properties. *Journal of Econometrics 29*(3), 305–325.

Meinshausen, N. (2007). Relaxed Lasso. *Computational Statistics & Data Analysis 52*(1), 374–393.

Meyer, D., E. Dimitriadou, K. Hornik, A. Weingessel, and F. Leisch (2021). *e1071: Misc Functions of the Department of Statistics, Probability Theory Group (Formerly: E1071), TU Wien*. R package version 1.7-9.

Miller, S. and R. Startz (2019). Feasible Generalized Least Squares using machine learning. *Economics Letters 175*(1), 28–31.

Perron, P. and González-Coya (2022). Feasible GLS for time series regression. *Working Paper, Department of Economics, Boston University*.

Robinson, P. M. (1987). Asymptotically efficient estimation in the presence of heteroskedasticity of unknown form. *Econometrica 55*(4), 875–891.

Romano, J. P. and M. Wolf (2017). Resurrecting Weighted Least Squares. *Journal of Econometrics 197*(1), 1–19.

Rothenberg, T. J. (1988). Approximate power functions for some robust tests of regression coefficients. *Econometrica 56*(5), 997–1019.

Tibshirani, R. (1996). Regression shrinkage and selection via the Lasso. *Journal of the Royal Statistical Society: Series B (Methodological) 58*(1), 267–288.

White, H. (1980). A heteroskedasticity-consistent covariance matrix estimator and a direct test for heteroskedasticity. *Econometrica 48*(4), 817–838.

Wooldridge, J. M. (2010). *Econometric Analysis of Cross Section and Panel Data*. MIT Press, Cambridge, MA.

Wooldridge, J. M. (2012). *Introductory Econometrics: A Modern Approach, 5th ed.* South-Western, Mason, Ohio.

Zou, H. (2006). The Adaptive Lasso and its oracle properties. *Journal of the American Statistical Association 101*(476), 1418–1429.

Zou, H., T. Hastie, and R. Tibshirani (2007). On the degrees of freedom of the Lasso. *The Annals of Statistics 35*(5), 2173–2192.

Estimation in the Presence of Heteroskedasticty of Unknown Form:
A Lasso-based Approach
by Emilio Gonzalez-Coya and Pierre Perron
Supplementary Material (Not for publication)

In this supplement, we present extended sets of results using alternative non-parametric estimation methods that have been proposed in the literature before to estimate the residual variance: Nearest Neighbor (Robinson, 1987), Local Linear regression (Fan and Yao, 1998) and Random Forest (Ramosaj and Pauly, 2019). In Section S1, we describe this estimation methods and in Section S2 we reproduce the full set of simulation results using these alternative methods. As stated in the main text, the FGLS estimators based on these methods were found to have poor finite sample properties relative to the FGLS estimators based on Lasso and SVR.

S1 Description of the estimation methods

S1.1 Nearest Neighbor

The k-Nearest Neighbor regression is a non parametric method which has been previously used to estimate of the skedastic functions. Robinson (1987) considers a FGLS estimator based on a k-Nearest Neighbor estimate of the variance function. Liitiäinen et al. (2008) suggests a method to estimate the residual variance based on Nearest Neighbor graphs in a general setting which covers non-additive heteroskedastic noise under non-i.i.d. sampling. The k-Nearest Neighbor regression aim to learn a function $\hat{v} : \mathbb{R}^d \to \mathbb{R}$ for every z_t. The k-NN regression computes the mean of the function values of its k-nearest neighbors

$$\hat{v}_t(z) = \frac{1}{k} \sum_{j \in N_k(z_t)} \log(\hat{u}_t^2),$$

where $N_k(z_t)$ is a set containing the indices of the k closest (in Euclidean distance) points

to z_t. In local neighborhoods of z_t, observations z_j are expected to have similar continuous associated value $\hat{v}_t(z)$, close to $\log(\hat{u}_t^2)$. Hence, even for an unknown z_t^*, the associated value must be similar to the value of one of the closest points, which is modeled by the average of the dependent variable value, $\log(\hat{u}_t^2)$, of the k-nearest points in $N_k(z_t)$. Note that the neighborhood size k of k-NN is an important parameter. For $k = 1$, the k-NN regression tends to the value of the nearest neighbor of z_t, whereas for $k = T$ it averages over all observations $z_j, j = 1, \ldots, T$.

Following Robinson (1987), we can further characterize the k-Nearest Neighbor regression as follows. Consider $D \leq d$ elements of z_t and let

$$s_m = (T-1)^{-1} \sum_{t=1}^{T} \left\{ z_{tm} - \left(T^{-1} \sum_{t=1}^{T} z_{tm} \right) \right\}^2, \quad 1 \leq m \leq D,$$

$$\rho_{ij} = \sum_{m=1}^{D} \frac{(z_{tm} - z_{jm})^2}{s_m}, \quad \text{for } (t, j = 1, \ldots, T, t \neq j).$$

Note that we first divide by the sample standard deviation prior to applying the Euclidean metric (we assume in any case that z_t has finite variance). For a given integer $k < T$, let c_t, $(1 \leq t \leq T)$ be constants satisfying

$$c_t > 0, \quad 1 \leq t \leq k,$$

$$c_t = 0, \quad t > k,$$

$$\sum_{t=1}^{k} c_t = 1.$$

Let

$$p_{ij} = 1 + \textstyle\sum_l I\left(\rho_{ij} < \rho_{lj}\right), \quad i \neq j,$$

$$q_{ij} = 1 + \textstyle\sum_l I\left(\rho_{lj} = \rho_{ij}\right), \quad i \neq j,$$

where the sums are over $1 \leq l \leq n, l \neq i, j$, and $I(\cdot)$ is the usual indicator function. Let

$$W_{ij} = q_{ij}^{-1} \sum_{l=p_{ij}}^{p_{ij}+q_{ij}-1} c_l, \quad i \neq j,$$

$$= 0, \quad i = j,$$

and define the k-NN estimator as

$$\hat{v}_t(z) = \sum_{j=1}^{T} \log(\hat{u}_j^2) W_{tj}.$$

The standard characterization of the k-NN regression method sets $c_t = k^{-1}$. We estimate the k-NN regression using the `knn.reg` function from the `FFN` package in `R` (Beygelzimer et al., 2019). We set $k = \sqrt{T}$, as usual in the literature. The estimated degree of freedom is $\hat{df} = T/(2k)$.

S1.2 Local Linear

Fan and Yao (1998) propose an efficient and adaptive method for estimating the conditional variance for (nonlinear) time series models. They apply a Local Linear regression to the squared residuals and demonstrate that, without knowing the skedastic function, the estimated conditional variance can be asymptotically as good as if it were known. This result holds as long as the correct covariates are included in the specification of the estimated skedastic function.

Local Linear regression is a one-dimensional kernel smoother. It solves a separate weighted least squares problem at each target point z_0. In view of the relation $E(\hat{u}^2 \mid z = z_0) = v(z_0)$, where $\hat{u}_t^2 = \left(y_t - \beta'_{\text{OLS}} x_t\right)^2$, the Local Linear estimator with kernel W and bandwidth h_1 is given by $\log(\tilde{v}(z_0)) = \hat{\alpha}(z_0) + \hat{\beta}(z_0) z_0$ where,

$$(\hat{\alpha}(z_0), \hat{\beta}(z_0)) = \arg \min_{\alpha(z_0), \beta(z_0)} \sum_{t=1}^{T} \left\{ \log(\hat{u}_t^2) - \alpha(z_0) - \beta(z_0)\left(z_t - z_0\right) \right\}^2 W\left(\frac{z_t - z_0}{h_1}\right)$$

Note that although we fit an entire linear model to the data in the region, we only use it to evaluate the fit at the single point z_0. A drawback of the Local Linear regression is that it can only handle a small set of covariates as there is a computational restriction on the dimension of each target point z_0.

We estimate the Local Linear regression using the `locfit` function from the `locfit` package in `R` (Loader, 2020). We use the (default) Tricube Kernel. The degree of freedom (df) is estimated by the number of covariates used in the skedastic estimation.

S1.3 Random Forest

The Random Forest procedure was suggested by (Breiman, 2001). It can be viewed as amn adaptively weighted nearest neighbors method[1] (Lin and Jeon, 2006). Random Forest has been used for consistent estimation of the residual variance. Mendez and Lohr (2011) proposes two such estimators: the first is based on the residual sum of squares and uses a bootstrap bias correction. The second is a difference-based estimator that uses proximity measures as weights. Ramosaj and Pauly (2019) prove that the first estimator is asymptotically L_1-consistent.

Random Forest is a substantial modification of bagging (bootstrap aggregation) that builds a large collection of de-correlated trees, and then averages them (Hastie et al., 2009). Bagging methods aim to reduce the variance by averaging many noisy but approximately unbiased models. Trees are ideal candidates for bagging, since they can capture complex interaction structures in the data, and if grown sufficiently deep, have relatively low bias. Since trees are notoriously noisy, they benefit greatly from the averaging. The baseline tree bagging procedure draws B different bootstrap samples of the data, fits a separate regression tree to each, then averages their forecasts. The idea in Random Forests is to improve the variance reduction of bagging by reducing the correlation between the trees. To this end, it uses a variation on bagging designed to reduce the correlation among trees in different bootstrap samples. The forest method de-correlates (i.e. reduces the correlation between) trees using a method known as "dropout," which considers only a randomly drawn

[1]Specifically, Lin and Jeon (2006) introduce the concept of potential nearest neighbors (k-PNNs) and show that Random Forests can be viewed as adaptively weighted k-PNN methods.

Algorithm 1: Random Forest for Regression (Hastie et al., 2009).

The data set is $D = \{\log(\hat{u}_t^2), z_t\}_{t=1}^{T}$. Fix B, the number of bootstrap samples and depth L.

for $b = 1, \ldots, B$ **do**

1. Draw a bootstrap sample D_b of size T from the data set.
2. Grow a random-forest tree $T_b(z)$ to the bootstrapped data, by recursively repeating the following steps for each terminal node of the tree, until the minimum node size n_{min} is reached.
   (a) Select m variables at random from the d variables z_t.
   (b) Pick the best variable/split-point among the m.
   (c) Split the node into two daughter nodes.
3. Output the ensemble of trees $\{T_b(z)\}_{b=1}^{B}$.

end

Result: The Random Forest (regression) predictor is

$$\hat{f}_{\text{rf}}^{B}(z) = \frac{1}{B}\sum_{b=1}^{B} T_b(z)$$

subset of predictors for splitting at each potential branch. Th depth L of the trees and the number of bootstrap samples B are the tuning parameters optimized via validation.

The tree-growing process using the "dropout" method is described in Algorithm 1 (Hastie et al., 2009). For a more detailed exposition, see Ramosaj and Pauly (2019). The degrees of freedom for Random Forest are estimated by the maximum number of terminal nodes (or leaves) among all the trees. We estimate the Random Forest using the `randomForest` function from the `randomForest` package in `R` (Liaw and Wiener, 2002).

S2 Simulation Results

In this section we present simulation results for the FGLS estimator based on Nearest Neighbor (k-NN), Local Linear regression and Random Forest (RF). In Subsection S2.1 we present results corresponding to Tables 1 and 2 of Section 5.1 of the main text. Subsection S2.2 presents results for the robustness experiments of Section 5.2 of the main text.

S2.1 Results Corresponding to Section 5.1

In Table S.1, we present simulation results for model (6) which are directly comparable with those in Table 1 of the main text. We present the eMSEs of the estimators of β relative to OLS with the percent reduction in eMSE relative to OLS in parentheses. The response variable is $\log(\max\{\hat{u}_t^2, \delta^2\})$ and the input matrix is

$$z_t = (1, x_t, \log(x_t), x_t^2, \log(x_t)^2, \cos(x_t), \cos(2x_t)).$$

These results are presented for each combination of heteroskedasticity scenario (row groups) and the estimator used for the skedastic function (columns). We compare 4 estimators in the following columns: (1) is the (unfeasible) GLS estimator based on the known skedastic function, (2-4) are for the FGLS estimators based on Local Linear regression, k-Nearest Neighbor and Random Forest, respectively. The first panel of Table S.1, pertains to the case of homoskedastic errors, for which OLS and infeasible GLS are best. With a small sample ($T = 100$) the FGLS estimators based on Local Linear regression, k-NN and Random Forest have increased variance and are less precise. This is not the case, however, of the Lasso-based FGLS estimator; in Table 1 we show that Lasso is able to achieve the same precision as OLS with homoskedastic errors. The next groups of results in Table S.1 correspond to moderate ($\nu_t(x)_1$ with $\gamma = 1$, and $\nu_t(x)_4$) and severe heteroskedasticity ($\nu_t(x)_1$ with $\gamma = 2$, $\nu_t(x)_2$ and $\nu_t(x)_3$). In general, the Local Linear-based FGLS estimator outperforms the other FGLS estimators considered, for all the skedastic functions. This

estimator presents a reduction in eMSE similar to the Lasso-based FGLS estimator (Table 1) when heteroskedasticity is present.

In Table S.2, we present the coverage rates and average length of the confidence intervals for β for $T = 100, 200, 400$. Here, and throughout, the nominal significance level is 95%. For the class of FGLS estimators based on a non parametric estimation of the skedastic function, we use the standard error correction of Miller and Startz (2018), given by (5). Under heteroskedasticiy, the Local Linear-based FGLS estimator has confidence intervals near the nominal level with average length considerably smaller than those of OLS. This results holds even for small samples ($T = 100$). Nearest Neighbor and Random Forest deliver large confidence intervals with coverage rates above the nominal level. The standard error correction of Miller and Startz (2018) is sensitive to the estimate of the degrees of freedom of the skedastic estimation. This results suggests that we over-estimate the degrees of freedom for Nearest Neighbor and Random Forest.

The FGLS estimator based on Local Linear regression has a performance similar to Lasso when heteroskedasticity is present. However, this result is true as long as we include the correct variable that characterizes the skedastic function. As discussed above, a drawback of the Local Linear regression is that it cannot handle a large set of covariates. Thus, in real world applications in which the variables influencing the skedastic function are unknown, Local Linear regression is no longer robust.

Table S.1: Relative eMSE of estimators of β. In parentheses are the percent eMSE reduction relative to OLS.

	OLS	GLS	Local Linear	k-NN	RF
$\nu(x) = 1$					
$T = 100$	1.00	1.00	1.09 (-0.09)	1.26 (-0.26)	1.12 (-0.12)
$T = 200$	1.00	1.00	1.03 (-0.03)	1.18 (-0.18)	1.08 (-0.08)
$T = 400$	1.00	1.00	1.02 (-0.02)	1.11 (-0.11)	1.13 (-0.13)
$\nu(x) = x$					
$T = 100$	1.00	0.89 (0.11)	0.96 (0.04)	1.17 (-0.17)	1.05 (-0.05)
$T = 200$	1.00	0.89 (0.11)	0.93 (0.07)	1.09 (-0.09)	1.03 (-0.03)
$T = 400$	1.00	0.90 (0.10)	0.95 (0.05)	1.05 (0.05)	1.05 (0.05)
$\nu(x) = x^2$					
$T = 100$	1.00	0.68 (0.32)	0.73 (0.27)	0.94 (0.06)	0.91 (0.09)
$T = 200$	1.00	0.67 (0.33)	0.72 (0.28)	0.88 (0.12)	0.93 (0.07)
$T = 400$	1.00	0.67 (0.33)	0.75 (0.25)	0.84 (0.16)	0.93 (0.07)
$\nu(x) = [\log(x)]^2$					
$T = 100$	1.00	0.37 (0.63)	0.55 (0.45)	0.71 (0.29)	0.68 (0.32)
$T = 200$	1.00	0.35 (0.65)	0.49 (0.51)	0.60 (0.40)	0.64 (0.36)
$T = 400$	1.00	0.31 (0.69)	0.50 (0.50)	0.56 (0.44)	0.64 (0.36)
$\nu(x) = \exp(0.2x + 0.2x^2)$					
$T = 100$	1.00	0.43 (0.57)	0.47 (0.53)	0.62 (0.38)	0.80 (0.20)
$T = 200$	1.00	0.43 (0.57)	0.47 (0.53)	0.58 (0.42)	0.82 (0.18)
$T = 400$	1.00	0.43 (0.57)	0.47 (0.52)	0.55 (0.45)	0.82 (0.18)
$\nu(x) = \nu(x)_4$					
$T = 100$	1.00	0.70 (0.30)	0.76 (0.24)	0.94 (0.06)	0.93 (0.07)
$T = 200$	1.00	0.70 (0.30)	0.76 (0.24)	0.90 (0.10)	0.93 (0.07)
$T = 400$	1.00	0.68 (0.32)	0.76 (0.24)	0.86 (0.14)	0.93 (0.07)

Table S.2: Coverage and average length of confidence intervals for β.

		OLS	Local Linear	k-NN	RF
$T = 100$					
$\nu(x) = 1$	Coverage	0.96	0.94	0.97	0.99
	Length	0.47	0.48	0.60	0.65
$\nu(x) = x$	Coverage	0.96	0.94	0.97	0.97
	Length	0.74	0.72	0.91	0.91
$\nu(x) = x^2$	Coverage	0.96	0.93	0.97	0.95
	Length	1.28	1.10	1.41	1.32
$\nu(x) = [\log(x)]^2$	Coverage	0.96	0.93	0.97	0.97
	Length	0.44	0.31	0.40	0.42
$\nu(x) = \exp(0.2x + 0.2x^2)$	Coverage	0.96	0.94	0.97	0.90
	Length	1.90	1.33	1.71	1.51
$\nu(x) = \nu(x)_4$	Coverage	0.96	0.94	0.96	0.95
	Length	1.04	0.91	1.15	1.09
$T = 200$					
$\nu(x) = 1$	Coverage	0.95	0.94	0.96	0.99
	Length	0.32	0.33	0.39	0.48
$\nu(x) = x$	Coverage	0.95	0.93	0.96	0.99
	Length	0.51	0.50	0.59	0.67
$\nu(x) = x^2$	Coverage	0.95	0.93	0.95	0.94
	Length	00.89	0.76	0.91	0.95
$\nu(x) = [\log(x)]^2$	Coverage	0.96	0.95	0.96	0.99
	Length	0.31	0.21	0.25	0.30
$\nu(x) = \exp(0.2x + 0.2x^2)$	Coverage	0.95	0.93	0.95	0.89
	Length	1.33	0.91	1.09	1.08
$\nu(x) = \nu(x)_4$	Coverage	0.95	0.93	0.95	0.95
	Length	0.72	0.63	0.74	0.79
$T = 400$					
$\nu(x) = 1$	Coverage	0.96	0.96	0.97	0.99
	Length	0.23	0.23	0.26	0.33
$\nu(x) = x$	Coverage	0.96	0.95	0.98	0.99
	Length	0.36	0.34	0.39	0.47
$\nu(x) = x^2$	Coverage	0.96	0.95	0.97	0.96
	Length	0.63	0.52	0.60	0.65
$\nu(x) = [\log(x)]^2$	Coverage	0.96	0.94	0.95	0.98
	Length	0.21	0.14	0.16	0.20
$\nu(x) = \exp(0.2x + 0.2x^2)$	Coverage	0.96	0.95	0.96	0.91
	Length	0.93	0.63	0.72	0.74
$\nu(x) = \nu(x)_4$	Coverage	0.96	0.95	0.96	0.96
	Length	0.51	0.44	0.49	0.55

S2.2 Results Corresponding to Section 5.2

In this section we reproduce the set of results from the robustness experiments presented in Section 5.2 of the main text (Table 3). We exclude the FGLS estimator based on Local Linear regression, as this method is not suitable to handle a large number of covariates. In Table S.3, we present relative eMSE reduction results for Cases 1 to 5. For Cases 1, 2 and 3, the k-NN-based FGLS estimator outperforms the Random Forest-based FGLS estimator under heteroskedasticity and it presents an eMSE reduction similar to the WLS-S1 and WLS-S2 estimators. However, under homoskedasticity, the k-NN-based FGLS estimator looses precision relative to OLS; this is not the case of he Random Forest-based FGLS estimator. For Cases 4 and 5, in which one or none of the covariates w_t are correlated with x_t, the k-NN-based FGLS estimator severely deteriorates whereas the Random Forest-based FGLS estimator delivers a large reduction in eMSE under heteroskedasticty. The performance of the latter estimator under this cases is inferior to the SVR-based FGLS estimator.

In Table S.4 we present the coverage rates and average length of the confidence intervals for β with $T = 200$ for Cases 1 to 5. Again, the nominal significance level is 95%. The Random Forest-based FGLS estimator exhibits confidence intervals larger than those for OLS and have a coverage probability exceeding the nominal level. Confidence intervals for the k-NN-based FGLS estimator are, in some cases, smaller than those for OLS but it has a coverage probability closer to the nominal level.

Table S.3: Relative eMSE of estimators of β. In parentheses are the percent eMSE reduction relative to OLS; $T = 200$.

	GLS	k-NN	RF
Case 1			
$\nu(x)_1 = x^\gamma$			
$\gamma = 0$	1.00	1.35 (-0.35)	1.02 (-0.02)
$\gamma = 1$	0.89 (0.11)	1.23 (-0.23)	1.03 (-0.02)
$\gamma = 2$	0.67 (0.33)	0.96 (0.04)	0.93 (0.07)
$\nu(x)_2 = [\log(x)]^2$	0.31 (0.69)	0.70 (0.30)	0.78 (0.22)
$\nu(x)_3 = \exp(0.2x + 0.2x^2)$	0.45 (0.55)	0.65 (0.35)	0.88 (0.12)
$\nu(x)_4$	0.70 (0.30)	1.00 (0.00)	0.94 (0.06)
Case 2			
$\nu(x)_1 = x^\gamma$			
$\gamma = 0$	1.00	1.47 (-0.47)	1.03 (-0.03)
$\gamma = 1$	0.87 (0.13)	1.30 (-0.30)	1.01 (-0.01)
$\gamma = 2$	0.65 (0.35)	0.99 (0.01)	0.95 (0.05)
$\nu(x)_2 = [\log(x)]^2$	0.37 (0.63)	0.68 (0.32)	0.80 (0.20)
$\nu(x)_3 = \exp(0.2x + 0.2x^2)$	0.42 (0.58)	0.68 (0.32)	0.90 (0.10)
$\nu(x)_4$	0.68 (0.32)	1.04 (-0.04)	0.95 (0.05)
Case 3			
$\nu(x)_1 = x^\gamma$			
$\gamma = 0$	1.00	1.38 (-0.38)	1.02 (-0.02)
$\gamma = 1$	0.91 (0.09)	1.26 (-0.26)	1.02 (-0.02)
$\gamma = 2$	0.71 (0.29)	1.02 (-0.02)	0.95 (0.05)
$\nu(x)_2 = [\log(x)]^2$	0.35 (0.65)	0.72 (0.28)	0.81 (0.19)
$\nu(x)_3 = \exp(0.2x + 0.2x^2)$	0.46 (0.56)	0.70 (0.30)	0.90 (0.10)
$\nu(x)_4$	0.72 (0.28)	1.02 (-0.02)	0.96 (0.04)
Case 4			
$\nu(x)_1 = x^\gamma$			
$\gamma = 0$	1.00	1.53 (-0.53)	1.04 (-0.04)
$\gamma = 1$	0.88 (0.12)	1.40 (-0.40)	1.00 (0.00)
$\gamma = 2$	0.66 (0.34)	1.36 (-0.36)	0.95 (0.05)
$\nu(x)_2 = [\log(x)]^2$	0.29 (0.71)	1.37 (-0.37)	0.84 (0.16)
$\nu(x)_3 = \exp(0.2x + 0.2x^2)$	0.43 (0.57)	1.23 (-0.23)	0.92 (0.08)
$\nu(x)_4$	0.68 (0.32)	1.37 (-0.37)	0.96 (0.04)
Case 5			
$\nu(x)_1 = x^\gamma$			
$\gamma = 0$	1.00	1.64 (-0.64)	1.02 (-0.02)
$\gamma = 1$	0.91 (0.09)	1.53 (-0.54)	1.01 (-0.01)
$\gamma = 2$	0.69 (0.31)	1.47 (-0.47)	0.97 (0.03)
$\nu(x)_2 = [\log(x)]^2$	0.36 (0.64)	1.62 (-0.62)	0.86 (0.14)
$\nu(x)_3 = \exp(0.2x + 0.2x^2)$	0.44 (0.56)	1.59 (-0.50)	0.93 (0.07)
$\nu(x)$	0.71 (0.29)	1.36 (-0.36)	0.98 (0.02)

Table S.4: Coverage rate and average length of confidence intervals; $T = 200$.

		OLS	k-NN	RF	OLS	k-NN	RF
		Case 1			Case 2		
$\nu(x) = 1$	Coverage	0.95	0.97	0.99	0.94	0.97	0.99
	Length	0.32	0.40	0.70	0.33	0.40	0.74
$\nu(x) = x$	Coverage	0.94	0.97	0.99	0.95	0.96	0.99
	Length	0.51	0.61	1.06	0.52	0.60	1.07
$\nu(x) = x^2$	Coverage	0.94	0.97	0.98	0.96	0.97	0.98
	Length	0.89	0.93	1.52	0.90	0.93	1.55
$\nu(x) = [\log(x)]^2$	Coverage	0.94	0.97	0.98	0.96	0.97	0.97
	Length	0.31	0.26	0.41	0.31	0.26	0.41
$\nu(x) = \exp(0.2x + 0.2x^2)$	Coverage	0.94	0.97	0.95	0.95	0.96	0.94
	Length	1.33	1.12	1.72	1.33	1.14	1.70
$\nu(x) = \nu(x)_4$	Coverage	0.94	0.96	0.98	0.96	0.96	0.97
	Length	0.72	0.76	1.19	0.72	0.76	1.22
		Case 3			Case 4		
$\nu(x) = 1$	Coverage	0.95	0.95	0.99	0.95	0.93	0.99
	Length	0.32	0.41	0.96	0.33	0.42	0.91
$\nu(x) = x$	Coverage	0.95	0.96	0.99	0.95	0.96	0.98
	Length	0.51	0.61	1.46	0.51	0.65	1.32
$\nu(x) = x^2$	Coverage	0.96	0.95	0.99	0.95	0.95	0.99
	Length	0.90	0.93	1.99	0.90	1.10	2.05
$\nu(x) = [\log(x)]^2$	Coverage	0.96	0.95	0.98	0.95	0.93	0.98
	Length	0.31	0.27	0.55	0.31	0.38	0.56
$\nu(x) = \exp(0.2x + 0.2x^2)$	Coverage	0.95	0.96	0.98	0.95	0.96	0.95
	Length	1.33	1.14	2.33	1.33	1.59	2.32
$\nu(x) = \nu(x)_4$	Coverage	0.96	0.97	0.99	0.96	0.94	0.98
	Length	0.72	0.76	1.63	0.72	0.89	1.63
		Case 5					
$\nu(x) = 1$	Coverage	0.94	0.94	0.98			
	Length	0.32	0.42	0.96			
$\nu(x) = x$	Coverage	0.95	0.96	0.98			
	Length	0.51	0.66	1.34			
$\nu(x) = x^2$	Coverage	0.95	0.97	0.98			
	Length	0.89	1.17	2.06			
$\nu(x) = [\log(x)]^2$	Coverage	0.96	0.97	0.98			
	Length	0.31	0.42	0.58			
$\nu(x) = \exp(0.2x + 0.2x^2)$	Coverage	0.96	0.96	0.97			
	Length	1.32	1.73	2.32			
$\nu(x) = \nu(x)_4$	Coverage	0.96	0.96	0.99			
	Length	0.72	0.94	1.60			

References

Beygelzimer, A., S. Kakadet, J. Langford, S. Arya, D. Mount, and S. Li (2019). *FNN: Fast Nearest Neighbor Search Algorithms and Applications*. R package version 1.1.3.

Breiman, L. (2001). Random Forests. *Machine Learning 45*(1), 5–32.

Fan, J. and Q. Yao (1998). Efficient estimation of conditional variance functions in stochastic regression. *Biometrika 85*(3), 645–660.

Hastie, T., R. Tibshirani, and J. Friedman (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. New York: Springer.

Liaw, A. and M. Wiener (2002). Classification and regression by randomForest. *R News 2*(3), 18–22.

Liitiäinen, E., F. Corona, and A. Lendasse (2008). On nonparametric residual variance estimation. *Neural Processing Letters 28*(3), 155–167.

Lin, Y. and Y. Jeon (2006). Random Forests and adaptive Nearest Neighbors. *Journal of the American Statistical Association 101*(474), 578–590.

Loader, C. (2020). *locfit: Local Regression, Likelihood and Density Estimation*. R package version 1.5-9.4.

Mendez, G. and S. Lohr (2011). Estimating residual variance in Random Forest regression. *Computational Statistics & Data Analysis 55*(11), 2937–2950.

Ramosaj, B. and M. Pauly (2019). Consistent estimation of residual variance with Random Forest out-of-bag errors. *Statistics & Probability Letters 151*, 49–57.

Robinson, P. M. (1987). Asymptotically efficient estimation in the presence of heteroskedasticity of unknown form. *Econometrica 55*(4), 875–891.