Political Analysis (2024), **32**, 157–171
doi:10.1017/pan.2023.27

PA

ARTICLE

Generalized Kernel Regularized Least Squares

Qing Chang[1] and Max Goplerud[2]

[1]Ph.D. Candidate, Department of Political Science, University of Pittsburgh, Pittsburgh, PA, USA; [2]Assistant Professor, Department of Political Science, University of Pittsburgh, Pittsburgh, PA, USA.

Corresponding author: Max Goplerud; Email: mgoplerud@pitt.edu

(Received 27 September 2022; revised 14 March 2023; accepted 31 March 2023; published online 1 September 2023)

Abstract

Kernel regularized least squares (KRLS) is a popular method for flexibly estimating models that may have complex relationships between variables. However, its usefulness to many researchers is limited for two reasons. First, existing approaches are inflexible and do not allow KRLS to be combined with theoretically motivated extensions such as random effects, unregularized fixed effects, or non-Gaussian outcomes. Second, estimation is extremely computationally intensive for even modestly sized datasets. Our paper addresses both concerns by introducing generalized KRLS (`gKRLS`). We note that KRLS can be re-formulated as a hierarchical model thereby allowing easy inference and modular model construction where KRLS can be used alongside random effects, splines, and unregularized fixed effects. Computationally, we also implement random sketching to dramatically accelerate estimation while incurring a limited penalty in estimation quality. We demonstrate that `gKRLS` can be fit on datasets with tens of thousands of observations in under 1 min. Further, state-of-the-art techniques that require fitting the model over a dozen times (e.g., meta-learners) can be estimated quickly.

Keywords: kernel ridge regression; hierarchical modeling; machine learning; heterogeneous effects

Edited by: Jeff Gill

1. Introduction

Designing models that can correctly estimate complex interactions between covariates or nonlinear effects of continuous predictors is an important but challenging problem. These models are increasingly popular not only as a robustness test to check the impact of functional form assumptions, but also as key constituent components to a variety of increasingly popular machine learning algorithms designed to estimate causal effects.

One popular method in political science to estimate a highly flexible model while maintaining good out-of-sample predictive performance is "kernel regularized least squares" (KRLS; Hainmueller and Hazlett 2014), also known as "kernel ridge regression" (e.g., Yang, Pilanci, and Wainwright 2017). This method provides a flexible approach to estimate a possibly complex underlying function and can easily capture interactions between covariates or nonlinear effects of certain predictors. It is simple to use as it only requires the researcher to provide a matrix of relevant predictors. Hainmueller and Hazlett (2014) describe other attractive features. However, it has two noticeable drawbacks that have likely limited its more widespread adoption. First, traditional approaches to estimating KRLS are rather inflexible as they require that all variables are included in a single kernel and regularized.[1] This prevents common extensions such as (unregularized) fixed effects, random effects, or multiple kernels for different sets of predictors from being included; further, it is challenging to estimate models with non-Gaussian

[1] `KSPM` (Schramm *et al.* 2020) is an exception, although it has some limitations discussed in Section E of the Supplementary Material.

outcomes (e.g., binary, ordered, or categorical outcomes) and difficult to implement alternative standard errors (e.g., cluster-robust standard errors). In many applied settings, researchers desire a "modular" approach like that found when using hierarchical models where different variables can be included in the model in different ways based on the researcher's theoretical beliefs.

Second, and equally importantly, traditional versions of KRLS are highly computationally expensive as the cost of estimation is dominated by the cube of the number of observations (Mohanty and Shaffer 2019; Yang *et al.* 2017). Without additional modification, it is difficult to fit these models with more than 10,000 observations—and even this may take many hours.

We introduce "generalized KRLS" (`gKRLS`) to tackle these issues. Our solution has two parts; first, some existing literature shows that (regular) KRLS can be re-formulated as a carefully chosen hierarchical model (e.g., Liu, Lin, and Ghosh 2007; Zhang, Dai, and Jordan 2011). Theoretically, this reformulation facilitates a modular model building strategy that can contain multiple kernels in addition to random effects, other smooth terms, and unpenalized fixed effects. However, using rich modular models can considerably complicate estimation using existing approaches given the need to tune multiple different regularization parameters. Fortunately, this hierarchical perspective also facilitates estimation techniques for fast tuning of the regularization parameters without expensive grid searches or cross-validation. These techniques also immediately extend to non-Gaussian outcomes and provide well-calibrated standard errors on key quantities of interest (Wood 2017). This reformulation alone, however, is insufficient to make `gKRLS` practical on large datasets given the cubic cost noted previously. We address this by using the popular "sub-sampling sketching" to reduce the cost of estimation by building the kernel based on a random sample of the original dataset (Drineas and Mahoney 2005; Yang *et al.* 2017).

Our paper proceeds as follows: Sections 2 and 3 describe `gKRLS`. Section 4 provides two simulations to illustrate its advantages; first, we examine the scalability of `gKRLS`.[2] While maintaining accurate estimates, `gKRLS` takes around 6 s for a dataset with 10,000 observations and two covariates and around 2 min with 100,000 observations without any parallelization and only 8GB of RAM. This compares with hours needed for existing approaches. Our second simulation shows the importance of having a flexible modular approach. We consider a data generating process that includes fixed effects for a group membership *outside* of the kernel. Traditional KRLS includes the fixed effects in the kernel which assumes the effect of all covariates can vary by group. We find this model is too flexible for modestly sized datasets and performance can be improved by including the fixed effects as unregularized terms "outside" the kernel.

Finally, we conduct two empirical analyses. Section 5 reanalyzes Newman's (2016) study of gender and beliefs in meritocracy. Building on theory from the original paper, we use the modular nature of `gKRLS` to estimate a logistic regression includes three hierarchical terms (random effects, splines, and KRLS) as well as unpenalized covariates (fixed effects). Estimation takes around 10 min with 8GB of RAM. Section 6 explores Gulzar, Haas, and Pasquale's (2020) study of the implications of political affirmative action for development in India. This is a larger dataset (around 30,000 observations), and our preferred model includes many unpenalized covariates and a single kernel. To address regularization bias, we also use `gKRLS` in algorithms that require fitting `gKRLS` between 10 and 15 times (e.g., double/debiased machine learning (DML); Chernozhukov *et al.* 2018). Estimation takes a few minutes.

2. Generalizing KRLS

There are many different approaches to presenting KRLS (Hainmueller and Hazlett 2014). We focus on the penalized regression presentation to build connections with hierarchical models. In this view, KRLS creates covariates that measure the similarity of observations (e.g., the transformed distance between covariate vectors) while penalizing the estimated coefficients to encourage estimation of conditional expectation functions that are relatively *smooth* and penalize excessively "wiggly" functions where the

[2]Chang and Goplerud (2023) contains the code to replicate these analyses.

outcome would vary dramatically given small changes in the predictors (Hainmueller and Hazlett 2014). This is a common goal for smoothing methods, and different underlying models lead to different design matrices and penalty terms (Wood 2017, Chapter 5). KRLS is especially useful when there are multiple variables that could interact in complex and possibly nonlinear ways as it does not require the explicit formulation of which interactions or non-linearities may be relevant. This differs from sparsity-based frameworks such as the LASSO that require creating a set of possibly relevant interactions and bases *before* deciding which ones are relevant.

Formally, assume the dataset has N observations with covariate vectors $\boldsymbol{w}_i$. We assume that $\{\boldsymbol{w}_i\}_{i=1}^N$ has been standardized—as our software does automatically—to ensure different covariates are comparable in scale. This prevents arbitrary changes (e.g., changing units from meters to feet) from affecting the analysis. Hainmueller and Hazlett (2014) center each covariate to have mean zero and variance one. We use Mahalanobis distance to also address potentially correlated input covariates; we thus assume that a mean-centering and whitening transformation has been applied to $\{\boldsymbol{w}_i\}_{i=1}^N$ such that the covariance of the stacked $\boldsymbol{w}_i$ equals the identity matrix.

Given this standardized data, we create an $N \times N$ kernel matrix $\boldsymbol{K}$ that contains the similarity between two observations. We use the popular Gaussian kernel, but our method can be used with other kernels. Equation (1) defines $\boldsymbol{K}$ that depends on a transformation of the squared Euclidean distance between the observations scaled by the kernel bandwidth which we fix to P—the number of covariates in $\boldsymbol{w}_i$—following Hainmueller and Hazlett (2014).[3]

$$\boldsymbol{K}_{ij} = \exp\left(-\frac{||\boldsymbol{w}_i - \boldsymbol{w}_j||^2}{P}\right). \tag{1}$$

In traditional KRLS, $\boldsymbol{K}$ becomes the design matrix in a least-squares problem with parameters $\boldsymbol{\alpha}$ to predict the outcome y_i with error variance σ^2. To prevent overfitting, KRLS includes a term that penalizes the wiggliness of the estimated function where a parameter λ determines the strength of the penalty. As λ grows very large, all observations are predicted the same value (i.e., there is no effect of any covariate on the outcome). As λ approaches zero, the function becomes increasingly wiggly, and predicted values might change dramatically for small changes in the covariates.

Equation (2) presents the KRLS objective where $\boldsymbol{k}_i$ denotes row i of kernel $\boldsymbol{K}$. It is equivalent to traditional KRLS as maximizing Equation (2), for a fixed λ, gives coefficient estimates $\hat{\boldsymbol{\alpha}}_\lambda$ (denoting the dependence on λ) that are identical to Hainmueller and Hazlett (2014).

$$\hat{\boldsymbol{\alpha}}_\lambda = \underset{\boldsymbol{\alpha}}{\operatorname{argmax}} \left\{ -\frac{1}{2\sigma^2}\left[\sum_{i=1}^{N}(y_i - \boldsymbol{k}_i^T\boldsymbol{\alpha})^2 + \lambda\boldsymbol{\alpha}^T\boldsymbol{K}\boldsymbol{\alpha}\right]\right\}; \quad \hat{\boldsymbol{\alpha}}_\lambda = (\boldsymbol{K} + \lambda\boldsymbol{I})^{-1}\boldsymbol{y}. \tag{2}$$

We start by viewing the problem from a more Bayesian perspective and choose a Gaussian prior for $\boldsymbol{\alpha}$ that implies a posterior mode on $\boldsymbol{\alpha}$, conditional on σ^2 and λ, that is identical to the penalized objective (see also Appendix 2 of Hainmueller and Hazlett 2014). This prior, sometimes known as the "Silverman g-prior," can also be derived from an independent and identically distributed Gaussian prior on each of the coefficients from the underlying feature space associated with the kernel $\boldsymbol{K}$ (Zhang *et al.* 2011). Thus, KRLS can be viewed as a traditional random effects model (or ridge regression) on the feature space associated with $\boldsymbol{K}$. Equation (3) displays this generative view of KRLS where $\boldsymbol{K}^-$ denotes the pseudo-inverse of $\boldsymbol{K}$ in the case of a non-invertible kernel.

$$y_i \sim N(\boldsymbol{k}_i^T\boldsymbol{\alpha}, \sigma^2); \quad \boldsymbol{\alpha} \sim N\left(\boldsymbol{0}, \frac{\sigma^2}{\lambda}\boldsymbol{K}^-\right). \tag{3}$$

A key advantage of this Bayesian view is that KRLS becomes simply a hierarchical model with particular choice of design and prior. This leads to the idea of "modular" model construction where

[3] If the design matrix of stacked $\boldsymbol{w}_i$ is not full rank, we use its rank instead of P and use a generalized inverse in the whitening transformation.

different priors are used for different components of the model. For example, it is common to have unpenalized terms (e.g., "fixed effects") alongside the regularized terms. Alternatively, theory may call for the inclusion of more traditional random effects for a geographic unit such as county. We define `gKRLS`, therefore, as a hierarchical model with at one least KRLS term on some covariates. Equation (4) presents the general model. Fixed effects ($\boldsymbol{\beta}$) have design ($\boldsymbol{x}_i$) for each observation $\boldsymbol{x}_i$. There are J penalized terms, indexed by $j \in \{1,\ldots,J\}$, with parameters $\boldsymbol{\alpha}_j$ and designs $\boldsymbol{z}_{ij}$. As is standard for hierarchical models, each $\boldsymbol{\alpha}_j$ has a multivariate normal prior with *precision* $\boldsymbol{S}_j$. Each hierarchical term j has its own parameter λ_j that governs the amount of regularization.

$$y_i \sim N\left(\boldsymbol{x}_i^T\boldsymbol{\beta} + \sum_{j=1}^{J}\boldsymbol{z}_{ij}^T\boldsymbol{\alpha}_j,\ \sigma^2\right); \quad \boldsymbol{\alpha}_j \sim N\left(\boldsymbol{0}, \frac{\sigma^2}{\lambda_j}\boldsymbol{S}_j^-\right) \quad \text{for} \quad j \in \{1,\cdots,J\}, \tag{4a}$$

$$\ln p\left(\boldsymbol{\beta}, \{\boldsymbol{\alpha}_j\}_{j=1}^J \middle| \{y_i\}_{i=1}^N, \sigma^2, \{\lambda_j\}_{j=1}^J\right) \propto -\frac{1}{2\sigma^2}\left[\begin{array}{l} \sum_{i=1}^N\left(y_i - \boldsymbol{x}_i^T\boldsymbol{\beta} - \sum_{j=1}^J \boldsymbol{z}_{ij}^T\boldsymbol{\alpha}_j\right)^2 + \\ \sum_{j=1}^J \lambda_j \boldsymbol{\alpha}_j^T \boldsymbol{S}_j \boldsymbol{\alpha}_j \end{array}\right]. \tag{4b}$$

Specific choices of design and prior give well-known models. If $\boldsymbol{z}_{ij}$ is a vector of group membership indicators and $\boldsymbol{S}_j$ is an identity matrix, this is a traditional random intercept. If $\boldsymbol{z}_{ij} = \boldsymbol{k}_i$ and $\boldsymbol{S}_j = \boldsymbol{K}$, we recover KRLS from Equation (3).

If one fixes σ^2 and $\{\lambda_j\}_{j=1}^J$, point estimates can be obtained by maximizing the log-posterior (Equation (4b)). Equation (5) shows the estimates, noting their dependence on the vector of smoothing parameters denoted as $\boldsymbol{\lambda} = \{\lambda_j\}_{j=1}^J$. Despite our different presentation, this gives identical point estimates to classical presentations of multilevel models (e.g., Hazlett and Wainstein 2022; see our Section A.1 of the Supplementary Material). We use $\boldsymbol{X}$ for the design of the fixed effects; $\boldsymbol{Z}$ denotes the matrix corresponding to all of the design matrices for the hierarchical effects stacked together and $\boldsymbol{\alpha}$ denotes the concatenated parameters $\{\boldsymbol{\alpha}_j\}_{j=1}^J$. $\boldsymbol{S}_{\boldsymbol{\lambda}}$ represents the block-diagonal concatenation of each penalty term $\lambda_j\boldsymbol{S}_j$.

$$\left[\begin{array}{c} \hat{\boldsymbol{\beta}}_{\boldsymbol{\lambda}} \\ \hat{\boldsymbol{\alpha}}_{\boldsymbol{\lambda}} \end{array}\right] = \left[\begin{array}{cc} \boldsymbol{X}^T\boldsymbol{X} & \boldsymbol{X}^T\boldsymbol{Z} \\ \boldsymbol{Z}^T\boldsymbol{X} & \boldsymbol{Z}^T\boldsymbol{Z} + \boldsymbol{S}_{\boldsymbol{\lambda}} \end{array}\right]^{-1}\left[\begin{array}{c} \boldsymbol{X}^T \\ \boldsymbol{Z}^T \end{array}\right]\boldsymbol{y}. \tag{5}$$

A key difficulty in using (generalized) KRLS is choosing the appropriate amount of regularization, that is, calibrating $\{\lambda_1, \lambda_2, \ldots, \lambda_J\}$. In the case of a single KRLS term (e.g., $J = 1$) and a Gaussian likelihood, Hainmueller and Hazlett (2014) use an efficient method where the leave-one-out cross-validated error can be computed as a function of λ and requires only a single decomposition of the kernel $\boldsymbol{K}$. One could employ K-fold cross-validation to tune λ if a non-Gaussian likelihood were used (Sonnet and Hazlett 2018). However, existing strategies encounter considerable challenges when there are multiple hierarchical terms ($J > 1$). Since Hainmueller and Hazlett's (2014) method may not be available, a popular alternative—grid searches across different possible values for each λ_j to minimize some criterion (e.g., cross-validated error)—is very costly even for modest J.

Our hierarchical and Bayesian perspective provides a different strategy for tuning $\boldsymbol{\lambda}$ for any choice of J: restricted maximum likelihood (REML).[4] This approach observes that $\boldsymbol{\beta}$ has a flat (improper) prior and considers the marginal likelihood after integrating out $\boldsymbol{\beta}$ and all $\boldsymbol{\alpha}_j$. A REML strategy estimates $\boldsymbol{\lambda}$ and σ^2 by maximizing the log of this marginal likelihood; this is also referred to as an empirical Bayes approach (Wood 2017, 263). Equation (6) shows this objective, noting that it is a function of $\boldsymbol{\lambda}$ and σ^2. $\ell(\hat{\boldsymbol{\beta}}_{\boldsymbol{\lambda}}, \hat{\boldsymbol{\alpha}}_{\boldsymbol{\lambda}})$ denotes the log-likelihood (Equation 4b) evaluated at the penalized estimates given $\boldsymbol{\lambda}$ (Equation 5). $|\boldsymbol{S}|_+$ denotes the product of the nonzero eigenvalues of $\boldsymbol{S}$; M_p is the dimension of the null

[4] Wood (2017) discusses other criterion, for example, generalized cross-validation, that could be employed.

space of $\boldsymbol{S}_{\boldsymbol{\lambda}}$.

$$\hat{\boldsymbol{\lambda}}, \widehat{\sigma^2} = \underset{\boldsymbol{\lambda}, \sigma^2}{\operatorname{argmax}} \left\{ \begin{array}{l} \ell(\hat{\boldsymbol{\beta}}_{\boldsymbol{\lambda}}, \hat{\boldsymbol{\alpha}}_{\boldsymbol{\lambda}}) - \dfrac{\hat{\boldsymbol{\alpha}}_{\boldsymbol{\lambda}}^T \boldsymbol{S}_{\boldsymbol{\lambda}} \hat{\boldsymbol{\alpha}}_{\boldsymbol{\lambda}}}{2\sigma^2} + \dfrac{\ln |\boldsymbol{S}_{\boldsymbol{\lambda}}/\sigma^2|_+}{2} + \\ \quad - \dfrac{1}{2} \ln \left| \dfrac{1}{\sigma^2} \begin{bmatrix} \boldsymbol{X}^T\boldsymbol{X} & \boldsymbol{X}^T\boldsymbol{Z} \\ \boldsymbol{Z}^T\boldsymbol{X} & \boldsymbol{Z}^T\boldsymbol{Z} + \boldsymbol{S}_{\boldsymbol{\lambda}} \end{bmatrix} \right| + \dfrac{M_p}{2} \ln(2\pi) \end{array} \right\}. \tag{6}$$

After finding $\hat{\boldsymbol{\lambda}}$ and $\widehat{\sigma^2}$, point estimates for $\boldsymbol{\beta}$ and $\boldsymbol{\alpha}$ are obtained by plugging the estimated $\hat{\boldsymbol{\lambda}}$ into Equation (5). Liu *et al.* (2007) use the REML approach for a single KRLS hierarchical term (e.g., $J = 1$), and we push that intuition further by noting that that KRLS can be part of a general J approach to hierarchical and generalized additive models.

In practical terms, Wood (2017) summarizes the extensive research into numerically stable and efficient approaches to optimizing Equation (6) and describes well-established and high-quality software (`mgcv` in `R`). For very large problems (in terms of the number of observations or parameters), further acceleration may be needed. Section A.2 of the Supplementary Material discusses a set of less stable but faster estimation techniques implemented in the same software.

The final piece of inference is quantifying uncertainty. The Bayesian perspective on hierarchical models suggests using the inverse of the Hessian of the log-posterior on $\{\boldsymbol{\beta}, \boldsymbol{\alpha}\}$ for the estimated variance matrix (Wood 2017).[5] In the linear case, this is the first term in Equation (5), scaled by $\widehat{\sigma^2}$. Section A.1 of the Supplementary Material summarizes existing literature that suggests this should have good frequentist coverage.

2.1. Extensions to Generalized KRLS

The above presentation focused on a Gaussian outcome with arbitrary J and homoskedastic errors. We discuss four important extensions that our hierarchical perspective facilitates. First, the preceding exposition is easily generalized to non-Gaussian likelihoods: One changes the likelihood in Equation (4), for example, $y_i \sim \text{Poisson}(\exp(\psi_i))$, where $\psi_i = \boldsymbol{x}_i^T \boldsymbol{\beta} + \sum_{j=1}^{J} \boldsymbol{z}_{ij}^T \boldsymbol{\alpha}_j$, and adjusts the objective in Equation (6). This is justified using a Laplace approximation for evaluating the integral of the log-posterior; $\hat{\boldsymbol{\beta}}_{\boldsymbol{\lambda}}$ and $\hat{\boldsymbol{\alpha}}_{\boldsymbol{\lambda}}$ are obtained using penalized iteratively re-weighted least squares (Wood 2017, Chapter 3).

Second, the hierarchical perspective also justifies robust and/or clustered standard errors. Section A.1 of the Supplementary Material provides a detailed justification of the typical "sandwich" formula with slight modifications. We also show existing standard errors for KRLS (Hainmueller and Hazlett 2014) differ from those derived using the Bayesian perspective discussed above. A simple example suggests that using the Bayesian perspective results in considerably better coverage.

Third, a key use for `gKRLS` is in machine learning techniques such as stacking or double/debiased machine learning. We provide a software integration of `gKRLS` (and `mgcv`) into popular packages for both methods. Section F of the Supplementary Material provides details.

Finally, we provide new software for easily calculating marginal effects and predicted outcomes for a variety of likelihoods (e.g., Gaussian, binomial, multinomial, etc.). Among other quantities, this allows users to calculate the "average marginal effect" (i.e., the partial derivative of the prediction with respect to a specific covariate averaged across all observations in the data; Hainmueller and Hazlett 2014). Section A.3 of the Supplementary Material provides details. We are able to properly incorporate uncertainty for *both* fixed and random effects for these quantities.

3. Improving Scalability of Generalized KRLS

The optimism of the above discussion, however, elides a critical limitation of `gKRLS` as currently proposed. We focus on the traditional KRLS case ($J = 1$, no fixed effects) to illustrate the problem.

[5] Wood, Pya, and Säfken (2016) discuss how to incorporate uncertainty from estimating $\hat{\boldsymbol{\lambda}}$.

Recall that the model has N observations but requires the estimation of N coefficients. Estimation is extremely time- and memory-intensive as the computational cost is roughly cubic in the number of observations and requires storing a possibly huge $N \times N$ matrix (Hainmueller and Hazlett 2014; Yang *et al.* 2017). While some work in political science has focused on reducing this cost, the fundamental problem remains and, in practice, limits its applicability to around 10,000 observations with 8GB of memory (Mohanty and Shaffer 2019) and possibly taking hours to estimate—as Section 4 shows. Thus, using `gKRLS` without modifications is simply impractical for most applied settings. Further, if one needs to fit the model repeatedly (e.g., for cross-validation), it is prohibitively expensive.

Fortunately, there is a large literature on how to approximately estimate kernel methods on large datasets. We employ "random sketching," focusing on "sub-sampling sketching" or "uniform sampling" (e.g., Drineas and Mahoney 2005; Lee and Ng 2020; Yang *et al.* 2017) to dramatically accelerate the estimation; other methods could be explored in future research (e.g., random features; Rahimi and Recht 2007).[6] The sub-sampling sketching method takes a random sample of M data points and uses them to build the kernel, reducing the size of the design to $N \times M$. If M is much smaller than N, this can reduce the cost of estimation considerably. Formally, define the M sampled observations as $\boldsymbol{w}_m^*$, $m \in \{1, \ldots, M\}$. If $k(\boldsymbol{w}_i, \boldsymbol{w}_m)$ is the function to evaluate the kernel (e.g., Equation 1), define the sketched kernel $\boldsymbol{K}^*$ as an $N \times M$ matrix with the (i,m)th element as follows:

$$\boldsymbol{K}_{im}^* = k(\boldsymbol{w}_i, \boldsymbol{w}_m^*). \tag{7}$$

Equivalently, one can define $\boldsymbol{K}^*$ by multiplying $\boldsymbol{K}$ by a sketching matrix $\boldsymbol{S}$ with dimensionality $M \times N$, that is, $\boldsymbol{K}^* = \boldsymbol{K}\boldsymbol{S}^T$. For sub-sampling sketching, $\boldsymbol{S}$ is proportional to a sparse matrix of zeros where each row m contains a "1" for the column index corresponding to the sampled observation m. Returning to simplest version of KRLS (Equation 2), Equation (8) shows the sketched version. $\boldsymbol{\alpha}_S$ denotes a $M \times 1$ vector of coefficients for the *sketched* kernel, where $\boldsymbol{k}_i^*$ is the i-th row of $\boldsymbol{K}^*$. The analog for more complex models is straightforward.

$$\hat{\boldsymbol{\alpha}}_S = \underset{\boldsymbol{\alpha}_S}{\operatorname{argmax}} \left\{ -\frac{1}{2\sigma^2} \left[\sum_{i=1}^{N} (y_i - \left[\boldsymbol{k}_i^*\right]^T \boldsymbol{\alpha}_S)^2 + \lambda \boldsymbol{\alpha}_S^T \boldsymbol{P} \boldsymbol{\alpha}_S \right] \right\}; \quad \boldsymbol{P} = \boldsymbol{S}\boldsymbol{K}\boldsymbol{S}^T. \tag{8}$$

3.1. Calibrating the Sketched Kernel

We note two key points to consider when using sub-sampling sketching. First, the sketching dimension M clearly affects performance. As M increases, the model will likely perform better (see Section C.5 of the Supplementary Material). Inspired by some literature on the Laplace approximation for standard hierarchical models (e.g., Shun and McCullagh 1995), the default setting in our software sets $M = \delta N^{1/3}$, for example, growing at a rate of $N^{1/3}$ times a (constant) sketching multiplier δ; this can be manually increased by the researcher as appropriate.

We show that $\delta = 5$ often provides good performance, but one could use a larger multiplier such as $\delta = 15$ if feasible. The sub-sampling sketching method can be used on very large datasets with this slowly growing M; for example, if $N = 100{,}000$, then $M = 232$ with a multiplier of five and $M = 696$ with a multiplier of fifteen. Section 4 shows both can be fit quite rapidly.

Even if M is relatively large, the sub-sampling sketching method may sometimes fail to provide a good representation of the original data (Yang *et al.* 2017). We also find some evidence of this when the kernel is complex (see Section C.5 of the Supplementary Material). Lee and Ng (2020) review the literature on how to improve these methods; future research could explore these techniques.

Second, sub-sampling sketching will not generate identical estimates if the model is re-estimated due to different sketching matrices. While this randomness is common to some statistical methods (e.g., random forests), researchers should carefully examine the sensitivity of their results to the specific

[6] Section B.2 of the Supplementary Material discusses an alternative form of sketching ("Gaussian sketching") and shows it incurs a significantly higher computational cost at little systematic improvement in performance.

sketching matrix chosen. Exactly characterizing the impact of this variability is outside of the scope of this paper, although it may often be relatively small especially when $\delta = 15$. Sections C.5 and E of the Supplementary Material examine this for our simulations and applied examples. Corroborating the above discussion about potential limitations of the sub-sampling sketching method, we find that when the kernel is relatively simple, there is a high degree of stability. When the kernel is complex, a larger multiplier may be needed to ensure stable estimates. Assuming it is computationally feasible, a researcher might fit the model multiple times with different sketching matrices to show robustness. If the quantity of interest seems to vary considerably, we suggest increasing the size of the sketching dimension.

4. Evaluating the Performance of Generalized KRLS

We evaluate the scalability of `gKRLS` when performing the tasks used in standard applications: estimating the model, calculating average marginal effects, and generating predictions on a new dataset of the same size as the training data. We compare `gKRLS` against popular existing implementations: KRLS (Hainmueller and Hazlett 2014) and bigKRLS (Mohanty and Shaffer 2019)—where we examine truncating the eigenvalues to speed estimation ("bigKRLS (T)" using a truncation threshold of 0.001) and not doing so ("bigKRLS (NT)").[7] Finally, to examine the role of the sketching multiplier, we fit `gKRLS` with $\delta \in \{5, 15\}$ ["gKRLS (5)" and "gKRLS (15)," respectively]. All numerical results in this paper are run on a single core with 8GB of RAM. We explore a range of sample sizes spaced from 100 to 1,000,000—spaced evenly on the log-10 scale. For this initial examination, we rely on a generative model from Hainmueller and Hazlett (2014) ("Three Hills, Three Valleys") shown below:

$$y_i \sim N(\mu_i, 0.25); \quad \mu_i = \sin(x_{i,1}) \cdot \cos(x_{i,2}). \tag{9}$$

We generate 50 datasets and calculate the average estimation time and accuracy across the simulations. We stop estimating methods once costs increase dramatically to limit computational burden. Figure 1 reports the estimation time: KRLS and bigKRLS (with truncation) can be estimated quickly when the number of observations is relatively small, but this increases rapidly as the sample size grows (around the rate of N^3). When there are more than 10,000 observations, even bigKRLS would take hours to estimate. By contrast, `gKRLS` is at least an order of magnitude faster.

Figure 1b illustrates this more starkly by reporting the logarithm of time on the vertical axis. Even with the large multiplier ("gKRLS (15)"), `gKRLS` takes a few minutes for 100,000 observations. Section B.2 of the Supplementary Material calculates an empirical estimate of the computational complexity of `gKRLS` and shows it is substantially lower than traditional methods. Even with 1 million observations, `gKRLS` ($\delta = 5$) takes under 1 h. Section A.2 of the Supplementary Material discusses an alternative estimation technique (`bam`) that decreases this time to around 3 min with no decline in performance.

Figure 2 demonstrates that sketching does not come at a material expense of performance in this simple case. We assess the out-of-sample predictive accuracy by generating a test dataset of equivalent size to the training data and report the root mean squared error (RMSE) of the predicted values. With the exception of bigKRLS with truncation ("bigKRLS(T)") that performs considerably worse, Figure 2 shows all that methods have similar performance. Section B.1 of the Supplementary Material examines the error on estimating the average marginal effect; it shows similarly equivalent performance.

4.1. Kernels and Fixed Effects

Traditional KRLS usually requires that one include all covariates in a single kernel. This has the benefit of allowing the marginal effect of each variable to depend on all others. However, this could be *too* flexible and require enormous amounts of data to reliably learn the underlying relationship. This problem is likely especially severe when considering fixed effects for group membership. Allowing all marginal

[7] KRLS also can truncate eigenvalues and returns nearly identical results to bigKRLS.

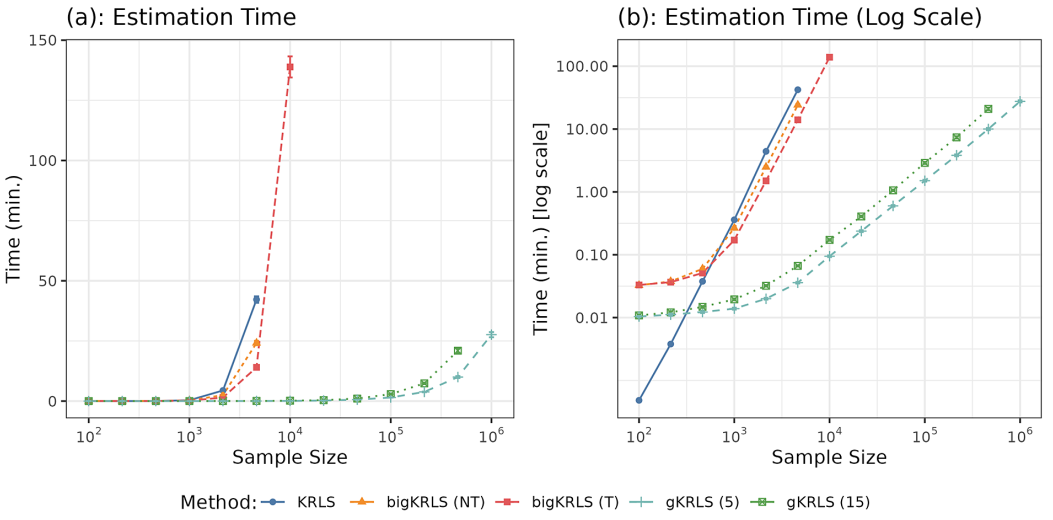

Figure 1. Comparison of running time for different models. This figure shows the average computational time in minutes averaged across simulations with 95% confidence intervals. Panel (a) presents average time in minutes. Panel (b) uses a logarithmic scale.

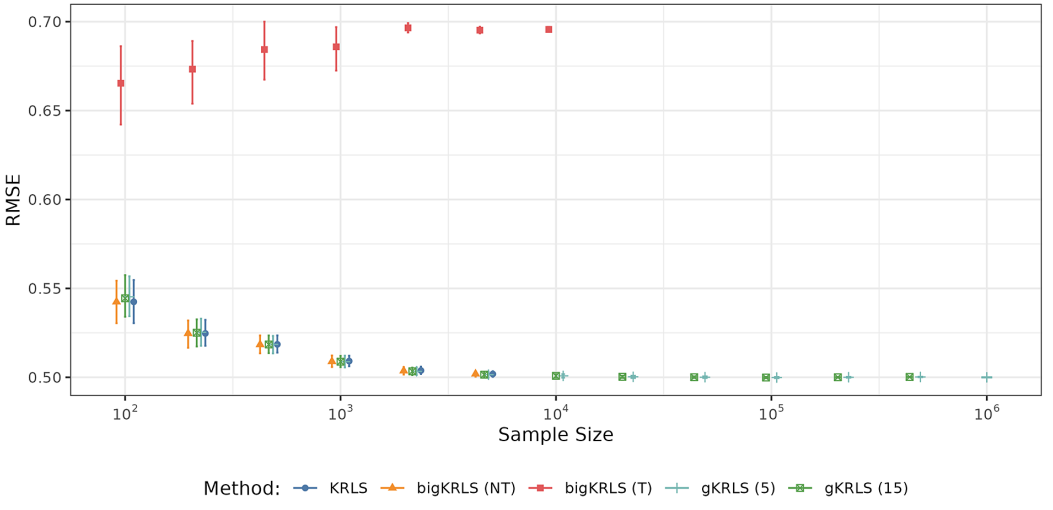

Figure 2. Performance on out of sample predictions. This figure shows the RMSE of predicting the outcome, averaged across 50 simulations. 95% confidence intervals using a percentile bootstrap (1,000 bootstrap samples) are shown.

effects to vary by group (e.g., a nonlinear analog to interacting group indicators with all covariates) is often too flexible given the potentially limited data in each group.

However, if one has theoretical reason to believe parts of the underlying model are additive (e.g., including fixed effects to address [additive] unobserved confounding), then including indicators for group *outside* the kernel (i.e., in β) will likely improve performance for modestly sized datasets. Since the group indicators are unregularized, this ensures that the usual "within-group" and "de-meaning" interpretation associated with fixed effects holds; this would not occur if they were included in the kernel.

We use a simulation environment that mimics traditional explorations of fixed effects (e.g., Bell and Jones 2015) but where the functional form of two continuous covariates is possibly nonlinear. One

of these covariates ($x_{i,1}$) is correlated with the fixed effects; and thus, its estimation should be more challenging as the correlation increases. The data generating process is shown below:

- Assume there are J groups with some number of observations T.
- Define $f_{\text{linear}}(x_1,x_2) = 0.5x_1 + 0.2x_2$. Define $f_{\text{nonlinear}}(x_1,x_2)$ as follows, following Table 3 in Hainmueller and Hazlett (2014):

$$f_{\text{nonlinear}}(x_1,x_2) = \exp\left(\frac{-(x_1-0.15)^2-(x_2-0.15)^2}{4}\right) + 2.5\cdot\exp\left(\frac{-(x_1-0.5)^2-(x_2-0.5)^2}{2.5}\right).$$

- Assign each observation i to some group j at random.
- Generate the covariates for each observation as follows: First, draw a fixed effect μ_j and a group level mean $\bar{x}_j$ for each group. ρ controls the amount of correlation. Larger ρ implies "random effects" should perform less well.

$$\begin{bmatrix} \mu_j \\ \bar{x}_j \end{bmatrix} \sim N\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 3 & \rho \\ \rho & 0.3 \end{bmatrix}\right); \quad x_{i,1} \sim N(\bar{x}_{j[i]},1); \quad x_{i,2} \sim N(0,1)$$

- Generate the outcome as follows for each $m \in \{\text{linear},\text{nonlinear}\}$:

$$y_i = f_m(x_{i,1},x_{i,2}) + \mu_{j[i]} + \epsilon_i; \quad \epsilon_i \sim N(0,1.25).$$

In our analysis, we set $\rho \in \{0,0.3,0.6,0.9\}$ to vary the degree of correlation between $x_{i,1}$ and μ_j.[8] We assume a reasonable number of groups ($J = 50$) and 10 observations per group ($T = 10$). We compare the following models: (linear) OLS, fixed, and random effect models. We also examine two kernel methods: bigKRLS (without truncation; with all variables in the kernel) and `gKRLS` (with a multiplier of five). For `gKRLS`, we use a kernel on $x_{i,1}$ and $x_{i,2}$ and include indicators for group membership outside the kernel as unregularized fixed effects (β). We run each simulation 1,000 times. We expect that all kernel methods should incur some penalty versus linear fixed effects when the true data generating process is linear. Figure 3 reports the RMSE of estimating the average marginal effect (following Hainmueller and Hazlett 2014) on the correlated covariate $x_{i,1}$.

First considering the linear data generating process (left panel), the traditional estimators (OLS, random effects, and fixed effects) behave as expected: OLS and random effects perform increasingly poorly as ρ increases. In the nonlinear data generating process, the same pattern holds although all three linear models perform less well as they are not able to capture the true underlying non-linearity.

When we compare the kernel methods used in the linear data generating process, both perform worse than fixed effects—that is, a correctly specified model—and neither method is affected much by ρ. However, `gKRLS` consistently outperforms bigKRLS by a considerable margin. In the nonlinear case, we see that both kernel methods perform well versus the linear alternatives, although `gKRLS` still has a considerable and constant advantage over bigKRLS. Section C.4 of the Supplementary Material shows that including the two covariates as fixed effects (β) in addition to their inclusion in the kernel improves performance considerably on the linear data generating process but incurs some penalty for the nonlinear case.

Section C of the Supplementary Material provides additional simulations. Section C.1 of the Supplementary Material considers alternative metrics for assessing the performance of the methods, for example, out of sample predictive accuracy. The results show a similar story: `gKRLS` is either

[8] The "true R^2" (i.e., the R^2 of a model that knew the true function) are similar to those in Hainmueller and Hazlett (2014), on average falling between 0.35 and 0.50.

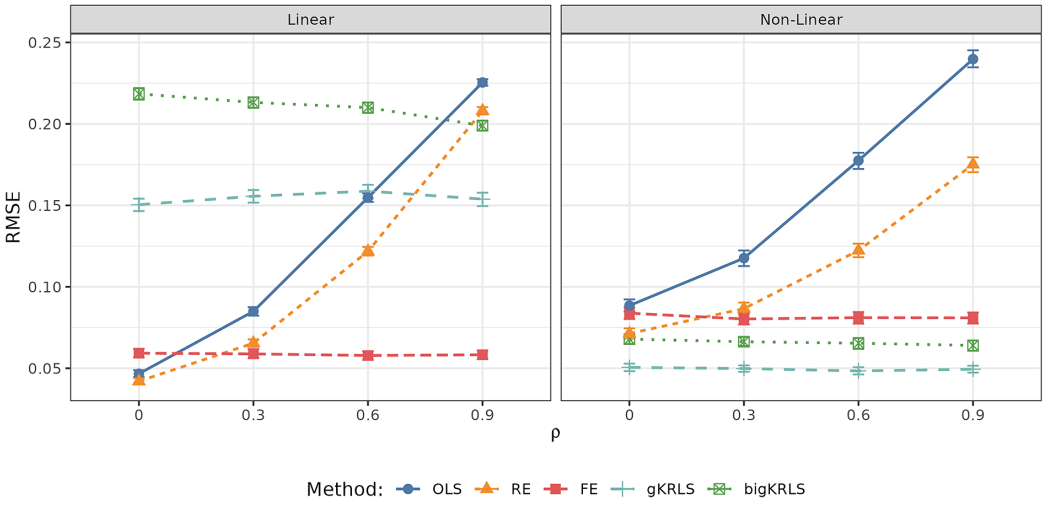

Figure 3. Performance for average marginal effect. The figure reports the RMSE of the estimated average marginal effect on $x_{i,1}$ as ρ varies. Each panel shows a different data generating process (linear or nonlinear). 95% confidence intervals using a percentile bootstrap (1,000 bootstrap samples) are shown.

close to bigKRLS or beats it by a considerable margin. Section C.2 of the Supplementary Material also explores the performance on estimating the effect of the second covariate ($x_{i,2}$): `gKRLS` outperforms bigKRLS. Section C.3 of the Supplementary Material considers an increasing number of observations per group (T). As T grows, both kernel methods improve—although `gKRLS` continues to perform better even when $T = 50$. To better understand why `gKRLS` improves upon bigKRLS, Section C.4 of the Supplementary Material shows that the improvement can be attributed solely to including the fixed effects *outside* the kernel—not additional changes such as how the smoothing parameter is selected, using Mahalanobis distance for creating the kernel, or sub-sampling sketching.

Finally, Section C.5 of the Supplementary Material explores the impact of sketching in this more complex case. It estimates models with different sketching matrices for fixed multiplier δ to understand the impact on the RMSE versus the unsketched estimates. It finds that sketching incurs some penalty on the accuracy of the estimated average marginal effect, although this declines as the sketching multiplier increases. When fixed effects are included in the kernel, this decline is considerably slower. When fixed effects are not included in the kernel, virtually any sketching multiplier can recover nearly identically accurate estimates to the corresponding unsketched procedure.

5. Generalized KRLS for Observational Data

Our first empirical application examines an observational study by Newman (2016). The paper focuses on the contextual effects of gender-based earnings inequality for women's belief in meritocracy. The key theoretical discussion concerns how gender inequality in earnings in the local area where a woman lives affects their rejection of a belief in meritocracy (e.g., "hard work and determination are no guarantee of success for most people"). Newman (2016, 1009–1111) compares a number of theoretical perspectives: Some (e.g., relative deprivation theory) suggest that women in areas with more economic inequality between men and women should show more rejection of meritocracy. However, Newman's (2016) preferred theoretical expectation, drawing on literature on "glass ceilings" and rising expectations theory, suggests a nonlinear effect: Rejection of meritocracy should be highest when women have come close to—but not quite achieved—economic parity as they have experienced large gains but still have failed to achieve equality. Once parity is achieved, the rejection of meritocracy should fall. Specifically, Newman (2016, 1011) expects a "nonlinear, concave quadratic effect of local gender-based earnings

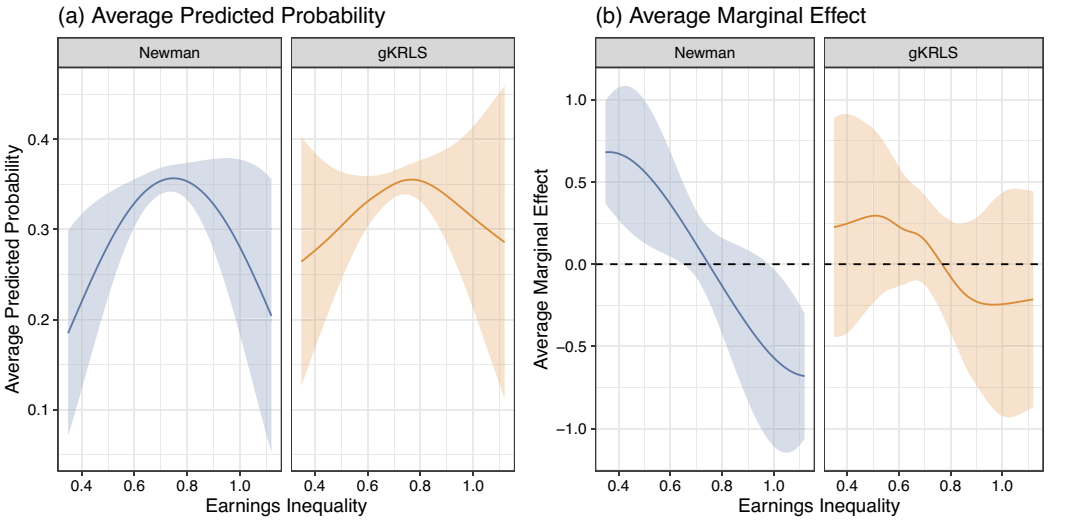

Figure 4. Re-analysis of Newman (2016). The average predicted probability (a) and average marginal effect (b) with 95% confidence intervals are shown.

inequality on women's likelihood of rejecting meritocracy." Newman (2016) tests this using hierarchical logistic regressions where the key variable (earnings inequality, operationalized as the ratio of female median income to male median income at the county of residence) is included quadratically.

gKRLS's modularity allows us to more robustly test Newman's (2016) argument. Our first hierarchical term is a kernel including all covariates to capture possible interactions or non-linearities omitted by the original (additive) model and thereby improve the robustness of the reported results. We also include a random intercept for county, following Newman (2016), to address the nested nature of the data.

However, Section 4 illustrated that relying exclusively on gKRLS given limited data may be undesirable as it could be too flexible. An additional risk of relying exclusively on KRLS is that if the estimated λ were very large, that would effectively exclude all covariates and mimic an intercept-only model. A more modular approach uses a KRLS term to flexibly estimate interactions or nonlinear effects while additionally including "primary" covariates of interest.

We include additional terms following Newman (2016). First, we include all controls in the fixed effects ($\boldsymbol{\beta}$). Second, we perform a more robust examination of the effect of earnings inequality. Rather than assuming the relationship is quadratic, we additively include a thin plate regression spline (Wood 2017, 216) on earnings inequality. This does not impose a specific functional form and allows the data to *reveal* whether the relationship is quadratic or has some other shape. This expanded model ensures that we include a specification that is comparable to Newman (2016) while also allowing for extra interactions using KRLS.

Overall, we estimate a logistic regression with four parts ($J = 3$): (i) a KRLS term including all controls and earnings inequality, (ii) a random effect for county; (iii) a spline on earnings inequality; and (iv) 24 controls entered in linearly and unpenalized (in $\boldsymbol{\beta}$). The three tuning parameters (separate λ_j for [i], [ii], and [iii]) are estimated using REML.

Figure 4 shows (a) the average predicted probability of rejecting meritocracy and (b) the average marginal effect across a grid of earning inequality values from the lowest to the highest value in the data—following Newman (2016). Section D of the Supplementary Material provides the question wording and definition of these quantities. Figure 4 reports the original specification in Newman (2016) as well as gKRLS.

The results partially support Newman (2016). The point estimates from gKRLS show a nonlinear inverted "u-shaped" relationship that is similar to the original results ("Newman"), although the curve is noticeably flatter for extreme values of earnings inequality. This occurs because gKRLS estimates relatively constant average marginal effects at extreme values of earnings inequality versus the mechanically increasing or decreasing values assumed by a quadratic specification.

When considering estimated uncertainty, however, we note that the 95% confidence intervals for the marginal effect from gKRLS cross zero at all points—unlike the original model. Section D of the Supplementary Material provides additional tests (e.g., average second derivative, difference in the average marginal effects at the extreme values) that show the same result (confidence intervals that contain zero for gKRLS). Thus, despite similar point estimates, relaxing the strong functional form assumptions in Newman (2016) returns limited evidence for a statistically detectable nonlinear relationship. Section D of the Supplementary Material corroborates this with other examples from the original paper: Using five other questions (binary and ordered logistic regressions), gKRLS generally finds an inverted "u-shaped" in the point estimates but little evidence of a statistically detectable nonlinear relationship.

6. Generalized KRLS with Machine Learning

Our second empirical replication considers a geographic regression discontinuity analysis in Gulzar *et al.* (2020). They focus on the effects of improving political representation using quotas on the economic welfare of various groups in society. They examine how electoral quotas for members of Scheduled Tribes affect the economic welfare of members of that group, members of a different historically disadvantaged group *not* affected by the quota (members of Scheduled Castes), members in neither group ("Non-Minorities"), as well as the total population.

We focus on their analysis of three economic outcome variables from the National Rural Employment Guarantee Scheme that offers 100 days of employment for rural households (Gulzar *et al.* 2020, 1231). The outcomes we consider are "(log) jobcards" (the total number of documents issued to prospective workers under the program), "(log) households" (the number of households who participated in the program), and "(log) workdays" (the total number of days worked by individuals in the program). The treatment is whether a village is part of a scheduled area that imposes an electoral quota. Across the three outcomes, the key findings from Gulzar *et al.* (2020) are that (i) there is no effect on the total economic welfare, (ii) the targeted minorities (Scheduled Tribes) see increases in economic welfare; (iii) the non-targeted minority groups (Scheduled Castes) do not see any significant changes; and (iv) non-minority groups see decreases in economic outcomes.

gKRLS can improve the original analysis in two ways. First, Gulzar *et al.* (2020) include the interaction of fourth-order polynomials on latitude and longitude following previous work on geographic regression discontinuity designs (replicated as "GHP" in Figure 5). gKRLS enables a more flexible solution, even on this larger dataset (32,461 observations), by using a kernel on the geographic coordinates[9] while including the treatment and other covariates linearly as unpenalized terms ($\boldsymbol{\beta}$). We denote this model as "gKRLS (Geog.)." Second, Gulzar et al. (2020, 1238) report some imbalance on certain pre-treatment covariates; they include controls additively and linearly to improve the robustness of their results. Including these variables (and treatment) in a KRLS term provides additional robustness. We use "gKRLS (All)" for this model that includes the KRLS term ($J = 1$) as well as all variables linearly in the fixed effects ($\boldsymbol{\beta}$) to ensure their inclusion. In both models, we use cluster-robust standard errors following the original specification.

The use of penalized terms, however, raises a concern about regularization bias in the estimated treatment effect; we address this using double/debiased machine learning (DML) that removes such bias (Chernozhukov *et al.* 2018). We use the specification from "gKRLS (All)" (after removing the treatment

[9]In this specification only, we rely on raw Euclidean distance, without standardization, due to the direct meaning of geographic distance.

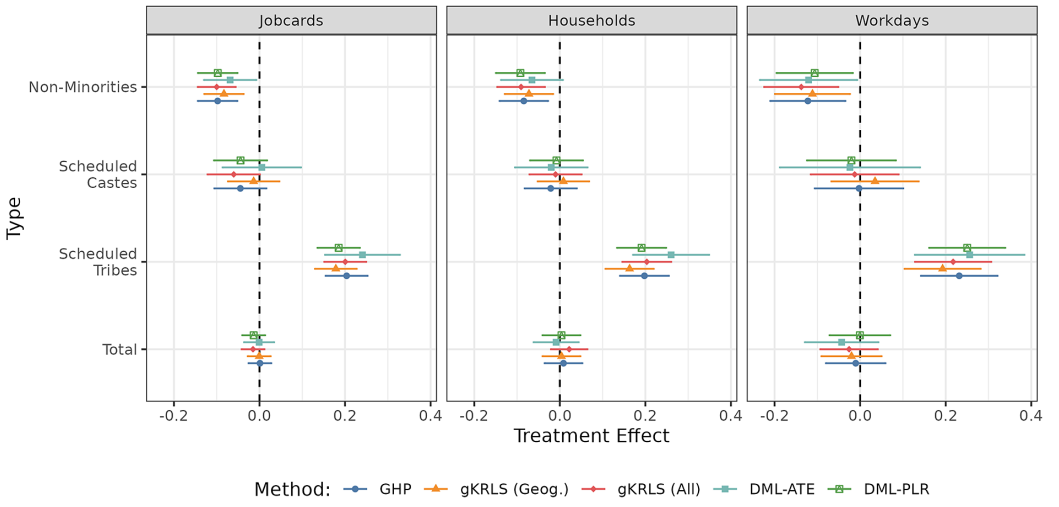

Figure 5. Effects of electoral quotas. This figure reports estimated treatment effects for all groups and outcomes. 95% confidence intervals are shown.

indicator) for our machine learning model. Estimation with five folds requires fitting `gKRLS` 10 or 15 times depending on whether one uses the partially linear model ("DML-PLR") or the dedicated algorithm for estimating the ATE ("DML-ATE"), respectively.[10] Both procedures estimate conditional expectation functions with Gaussian outcomes, while the latter (DML-ATE) also estimates a propensity score for being treated using a binomial outcome with a logistic link. Either procedure takes only a few minutes to estimate. To address clustering within the data, we use stratified sampling to create the folds for DML and produce the standard error on the treatment effect using an analog to the usual cluster-robust estimator (Chiang *et al.* 2022).

Figure 5 presents the results. The results are generally robust regardless of the specification chosen. The one exception is DML-ATE that has consistently larger standard errors (around 40% greater than other specifications) and somewhat larger point estimates for effects on Scheduled Tribes across two outcome variables.

Section E of the Supplementary Material provides additional analyses. Section E.1 of the Supplementary Material repeats the analysis 50 times to examine variability across different sketching matrices. It finds relatively low variability of the point estimates relative to the magnitude of the estimated standard errors. Section E.2 of the Supplementary Material uses `gKRLS` with a machine learning algorithm to estimate heterogeneous treatment effects ("R-learner"; Nie and Wager 2021). Even though this method requires fitting `gKRLS` over a dozen times (with both Gaussian and binomial outcomes), estimation takes only a few minutes. We find that one state (Himachal Pradesh) has noticeably larger treatment effects than other states.

7. Conclusion

Our paper generalized KRLS in two meaningful directions by drawing together different existing literatures. First, we recast the original model into the modular framework of hierarchical and generalized additive models where adding a kernel on some variables can be thought of as simply adding one additional hierarchical term (i.e., increasing J by one). This allows researchers using `gKRLS` to modularly build their model by including variables in different ways based on their substantive knowledge. For models with multiple hierarchical terms and/or non-Gaussian outcomes, a hierarchical

[10] Following Chernozhukov *et al.* (2018), we trim the estimated propensity scores at 0.01 and 0.99.

perspective on KRLS allows for easy tuning of the regularization parameters, efficient estimation, and well-calibrated standard errors. Empirically, we show that in a stylized example with additive fixed effects, thinking carefully about how to include different terms in the model (e.g., unregularized fixed effects versus including them in the kernel) can be critically important to performance. The second generalization employed sub-sampling sketching to allow `gKRLS` to be easily scalable to most datasets encountered in social science. By breaking the requirement that the cost of the model depends on the cube of the number of observations, sub-sampling sketching allows the model to be estimated very quickly on tens or hundreds of thousands of observations. Even for methods that require repeated estimation of `gKRLS` (e.g., DML), models can be estimated with limited computational cost. Our paper and accompanying software, therefore, allows KRLS to become a more widely used part of the applied researcher's toolkit.

Acknowledgments. We thank Michael Auslen, Saad Gulzar, Chad Hazlett, Adeline Lo, Marc Ratkovic, Brandon Stewart, and participants at MPSA 2022 and APSA 2022 for helpful feedback on this project.

Supplementary Material. For supplementary material accompanying this paper, please visit https://doi.org/10.1017/pan.2023.27.

Data Availability Statement. Replication data and code are available at https://doi.org/10.7910/DVN/WNW0AD. An `R` package to implement the methods in this paper is available at https://CRAN.R-project.org/package=gKRLS or https://github.com/mgoplerud/gKRLS. Section E of the Supplementary Material provides a demonstration.

Funding. This research was supported in part by the University of Pittsburgh Center for Research Computing, RRID:SCR_022735, through the resources provided. Specifically, this work used the H2P cluster, which is supported by the NSF award number OAC-2117681.

References

Bell, A., and K. Jones. 2015. "Explaining Fixed Effects: Random Effects Modeling of Time-Series Cross-Sectional and Panel Data." *Political Science Research and Methods* 3 (1): 133–153.

Chang, Q., and M. Goplerud. 2023. "Replication Data for: Generalized Kernel Regularized Least Squares." Harvard Dataverse. https://doi.org/10.7910/DVN/WNW0AD.

Chernozhukov, V., D. Chetverikov, M. Demirer, E. Duflo, C. Hansen, W. Newey, and J. Robins. 2018. "Double/Debiased Machine Learning for Treatment and Structural Parameters." *Econometrics Journal* 21 (1): 1–68.

Chiang, H. D., K. Kato, Y. Ma, and Y. Sasaki. 2022. "Multiway Cluster Robust Double/Debiased Machine Learning." *Journal of Business & Economic Statistics* 40 (3): 1046–1056.

Drineas, P., and M. W. Mahoney. 2005. "On the Nyström Method for Approximating a Gram Matrix for Improved Kernel-Based Learning." *Journal of Machine Learning Research* 6 (12): 2153–2175.

Gulzar, S., N. Haas, and B. Pasquale. 2020. "Does Political Affirmative Action Work, and for Whom? Theory and Evidence on India's Scheduled Areas." *American Political Science Review* 114 (4): 1230–1246.

Hainmueller, J., and C. Hazlett. 2014. "Kernel Regularized Least Squares: Reducing Misspecification Bias with a Flexible and Interpretable Machine Learning Approach." *Political Analysis* 22 (2): 143–168.

Hazlett, C., and L. Wainstein. 2022. "Understanding, Choosing, and Unifying Multilevel and Fixed Effect Approaches." *Political Analysis* 30 (1): 46–65.

Lee, S., and S. Ng. 2020. "An Econometric Perspective on Algorithmic Subsampling." *Annual Review of Economics* 12: 45–80.

Liu, D., X. Lin, and D. Ghosh. 2007. "Semiparametric Regression of Multidimensional Genetic Pathway Data: Least-Squares Kernel Machines and Linear Mixed Models." *Biometrics* 63 (4): 1079–1088.

Mohanty, P., and R. Shaffer. 2019. "Messy Data, Robust Inference? Navigating Obstacles to Inference with bigKRLS." *Political Analysis* 27 (2): 127–144.

Newman, B. J. 2016. "Breaking the Glass Ceiling: Local Gender-Based Earnings Inequality and Women's Belief in the American Dream." *American Journal of Political Science* 60 (4): 1006–1025.

Nie, X., and S. Wager. 2021. "Quasi-Oracle Estimation of Heterogeneous Treatment Effects." *Biometrika* 108 (2): 299–319.

Rahimi, A., and B. Recht. 2007. Random Features for Large-Scale Kernel Machines. In *Advances in Neural Information Processing Systems*, Vol. 20, pp. 1177–1184, https://proceedings.neurips.cc/paper/2007/hash/013a006f03dbc5392effeb8f18fda755-Abstract.html.

Schramm, C., S. Jacquemont, K. Oualkacha, A. Labbe, and C. M. T. Greenwood. 2020. "KSPM: A Package for Kernel Semi-Parametric Models." *The R Journal* 12 (2): 82–106.

Shun, Z., and P. McCullagh. 1995. "Laplace Approximation of High Dimensional Integrals." *Journal of the Royal Statistical Society: Series B (Methodological)* 57 (4): 749–760.
Sonnet, L., and C. Hazlett. 2018. "Kernel Regularized Logistic Regression: Avoiding Misspecification Bias while Maintaining Interpretability for Binary Outcome Regressions." Working Paper, 1–23.
Wood, S. N. 2017. *Generalized Additive Models*. Boca Raton, FL: Chapman and Hall/CRC.
Wood, S. N., N. Pya, and B. Säfken. 2016. "Smoothing Parameter and Model Selection for General Smooth Models." *Journal of the American Statistical Association* 111 (516): 1548–1563.
Yang, Y., M. Pilanci, and M. J. Wainwright. 2017. "Randomized Sketches for Kernels: Fast and Optimal Nonparametric Regression." *Annals of Statistics* 45 (3): 991–1023.
Zhang, Z., G. Dai, and M. I. Jordan. 2011. "Bayesian Generalized Kernel Mixed Models." *Journal of Machine Learning Research* 12: 111–139.

Cite this article: Chang, Q. and Goplerud, M. (2024). Generalized Kernel Regularized Least Squares. *Political Analysis*, 157–171. https://doi.org/10.1017/pan.2023.27