Messy Data, Robust Inference?
Navigating Obstacles to Inference with `bigKRLS`

Pete Mohanty, Stanford University
Robert B. Shaffer, University of Texas, Austin

International Methods Colloquium
Friday November 11th, 2016
Watch Live @ Noon Eastern

Overview

- **Part I**: Kernel Regularized Least Squares (KRLS)
  - What is it? Why use it?
  - Why optimize it?

Overview

- **Part I**: Kernel Regularized Least Squares (KRLS)
  - What is it? Why use it?
  - Why optimize it?
- **Part II**: Examples:
  - Treatment Effect Heterogeneity in a GOTV Field Experiment Gerber, Green & Larimer '08
  - Exploring Complex Interactions with Shiny (Trump vs. Hillary)

Overview

- **Part I**: Kernel Regularized Least Squares (KRLS)
  - What is it? Why use it?
  - Why optimize it?
- **Part II**: Examples:
  - Treatment Effect Heterogeneity in a GOTV Field Experiment Gerber, Green & Larimer '08
  - Exploring Complex Interactions with Shiny (Trump vs. Hillary)
- **Part III**: Introducing `bigKRLS` for `R`
  - New Software Architecture
    - `bigmemory` for Size
    - `Rcpp` for Speed

Overview

- **Part I**: Kernel Regularized Least Squares (KRLS)
  - What is it? Why use it?
  - Why optimize it?
- **Part II**: Examples:
  - Treatment Effect Heterogeneity in a GOTV Field Experiment Gerber, Green & Larimer '08
  - Exploring Complex Interactions with Shiny (Trump vs. Hillary)
- **Part III**: Introducing `bigKRLS` for `R`
  - New Software Architecture
    - `bigmemory` for Size
    - `Rcpp` for Speed
    - `Shiny` for Interactive Data Visualization

Overview

- **Part I**: Kernel Regularized Least Squares (KRLS)
  - What is it? Why use it?
  - Why optimize it?
- **Part II**: Examples:
  - Treatment Effect Heterogeneity in a GOTV Field Experiment Gerber, Green & Larimer '08
  - Exploring Complex Interactions with Shiny (Trump vs. Hillary)
- **Part III**: Introducing `bigKRLS` for `R`
  - New Software Architecture
    - `bigmemory` for Size
    - `Rcpp` for Speed
    - `Shiny` for Interactive Data Visualization
  - Leaner Algorithm
    - Better Performance with Many x Variables
    - New Estimator for Binary Variables
    - Performance Tests

Overview

- **Part I**: Kernel Regularized Least Squares (KRLS)
  - What is it? Why use it?
  - Why optimize it?
- **Part II**: Examples:
  - Treatment Effect Heterogeneity in a GOTV Field Experiment Gerber, Green & Larimer '08
  - Exploring Complex Interactions with Shiny (Trump vs. Hillary)
- **Part III**: Introducing `bigKRLS` for `R`
  - New Software Architecture
    - `bigmemory` for Size
    - `Rcpp` for Speed
    - `Shiny` for Interactive Data Visualization
  - Leaner Algorithm
    - Better Performance with Many x Variables
    - New Estimator for Binary Variables
    - Performance Tests
- **Discussion**: Data Science as Complexity vs. Interpretability

Kernel Regularized Least Squares (KRLS)

Motivations

- Maximize inference, minimize assumptions

Kernel Regularized Least Squares (KRLS)

Motivations

- Maximize inference, minimize assumptions
- Accurate out-of-sample prediction

Kernel Regularized Least Squares (KRLS)

Motivations

- Maximize inference, minimize assumptions
- Accurate out-of-sample prediction
- Estimate not just “average” but “actual” marginal effects

Kernel Regularized Least Squares (KRLS)

Motivations

- Maximize inference, minimize assumptions
- Accurate out-of-sample prediction
- Estimate not just "average" but "actual" marginal effects
- Interpretability

Kernel Regularized Least Squares (KRLS)

Motivations

- Maximize inference, minimize assumptions
- Accurate out-of-sample prediction
- Estimate not just “average” but “actual” marginal effects
- Interpretability

Related Work

- For details on statistical properties, see Hainmueller & Hazlett 2013, cf. Hastie et al 2008

Kernel Regularized Least Squares (KRLS)

Motivations

- Maximize inference, minimize assumptions
- Accurate out-of-sample prediction
- Estimate not just "average" but "actual" marginal effects
- Interpretability

Related Work

- For details on statistical properties, see Hainmueller & Hazlett 2013, cf. Hastie et al 2008
- "Kernel balancing" may help observational studies approximate experiments, see, e.g., Hazlett

Kernel Regularized Least Squares (KRLS)

Motivations

- Maximize inference, minimize assumptions
- Accurate out-of-sample prediction
- Estimate not just “average” but “actual” marginal effects
- Interpretability

Related Work

- For details on statistical properties, see Hainmueller & Hazlett 2013, cf. Hastie et al 2008
- “Kernel balancing” may help observational studies approximate experiments, see, e.g., Hazlett
- For Bayesian generalized version, Zhang, Dai & Jordan 2011

Some Details

We assume that the objective function $\mathbf{y} = \mathbf{f}(\mathbf{x})$ can be approximated by

$$\mathbf{y} = \mathbf{K}\mathbf{c}$$

Some Details

We assume that the objective function $\mathbf{y} = \mathbf{f}(\mathbf{x})$ can be approximated by

$$\mathbf{y} = \mathbf{K}\mathbf{c}$$

With $\mathbf{K}$ the Gaussian kernel,

$$\mathbf{K} = e^{-||x_i - x_j||^2/\sigma^2}.$$

Some Details

We assume that the objective function $\mathbf{y} = \mathbf{f}(\mathbf{x})$ can be approximated by

$$\mathbf{y} = \mathbf{Kc}$$

With $\mathbf{K}$ the Gaussian kernel,

$$\mathbf{K} = e^{-||x_i - x_j||^2/\sigma^2}.$$

Let $\mathbf{c}^*$ be a vector of weights which reflects a squared L_2 penalty:

$$\mathbf{c}^* = \underset{c \in \mathbb{R}^P}{\operatorname{argmin}} (\mathbf{y} - \mathbf{Kc})'(\mathbf{y} - \mathbf{Kc}) + \lambda \mathbf{c}'\mathbf{Kc}$$

Some Details

We assume that the objective function $\mathbf{y} = \mathbf{f}(\mathbf{x})$ can be approximated by

$$\mathbf{y} = \mathbf{K}\mathbf{c}$$

With $\mathbf{K}$ the Gaussian kernel,

$$\mathbf{K} = e^{-||x_i - x_j||^2/\sigma^2}.$$

Let $\mathbf{c}^*$ be a vector of weights which reflects a squared L_2 penalty:

$$\mathbf{c}^* = \underset{c \in \mathbb{R}^P}{\operatorname{argmin}} (\mathbf{y} - \mathbf{K}\mathbf{c})'(\mathbf{y} - \mathbf{K}\mathbf{c}) + \lambda \mathbf{c}'\mathbf{K}\mathbf{c}$$

Under assumptions (largely) analogous to classical regression, KRLS is unbiased and consistent (Hainmueller & Hazlett 2013).

"Actually" Marginal Effects

The marginal effects are:

$$\hat{\mathbf{\Delta}}_{\mathbf{N*P}} = [\hat{\delta}_{\mathbf{1}} \quad \hat{\delta}_{\mathbf{2}} \ldots \hat{\delta}_{\mathbf{P}}]$$

where, without loss of generality,

$$\hat{\delta}_{\mathbf{2}} = -\tfrac{2}{\sigma^2}(\mathbf{D_{(2)}} \cdot \mathbf{K}) * \hat{\mathbf{c}}^*$$

where $\mathbf{D}_{(2)}$ is an N by N matrix of simple distances on $\mathbf{X}_2$
and σ^2 is set to P.

"Actually" Marginal Effects

The marginal effects are:

$$\hat{\mathbf{\Delta}}_{\mathbf{N*P}} = [\hat{\delta}_{\mathbf{1}} \quad \hat{\delta}_{\mathbf{2}} \ldots \hat{\delta}_{\mathbf{P}}]$$

where, without loss of generality,

$$\hat{\delta}_{\mathbf{2}} = -\tfrac{2}{\sigma^2}(\mathbf{D_{(2)}} \cdot \mathbf{K}) * \hat{\mathbf{c}}^*$$

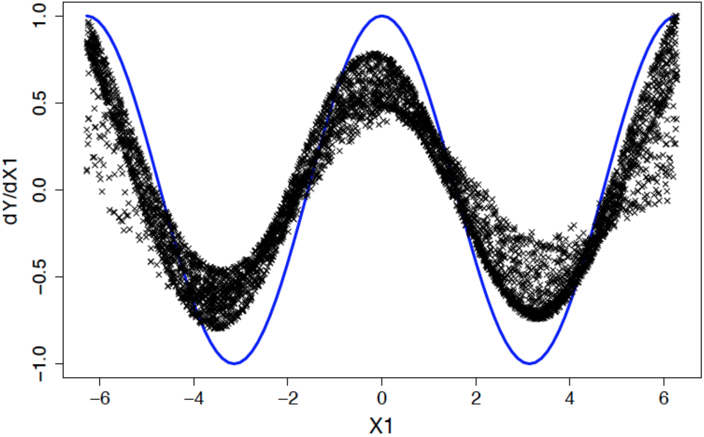
where $\mathbf{D}_{(2)}$ is an N by N matrix of simple distances on $\mathbf{X}_2$
and σ^2 is set to P.

And the average marginal effects are the column means of $\hat{\mathbf{\Delta}}_{\mathbf{N*P}}$.

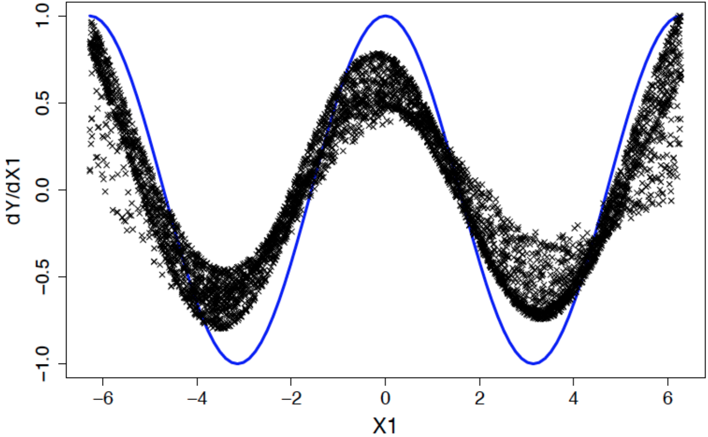
Quick Illustration...

```
library(bigKRLS)
N <- 5000
P <- 2
set.seed(11112016)
X <- matrix(runif(N * P, min = -2 * pi, max = 2 * pi),
    ncol = P)
y <- sin(X[, 1]) + X[, 2] + rnorm(N)
fit <- bigKRLS(y, X)
```

Marginal Effects of X1



Marginal Effects of X1




```r
mean((abs(fit$derivatives[,1]) < abs(cos(X[,1]))))
```

```
## [1] 0.8624
```

Why Optimize?

- Speed

Why Optimize?

- Speed
  - Luke Sonnet (UCLA) notes speed problems with `KRLS` starting at N $\approx$ 1,000 but has made substantial speed gains in Julia.

Why Optimize?

- Speed
  - Luke Sonnet (UCLA) notes speed problems with `KRLS` starting at N $\approx$ 1,000 but has made substantial speed gains in Julia.
- Size

Why Optimize?

- Speed
  - Luke Sonnet (UCLA) notes speed problems with `KRLS` starting at N $\approx$ 1,000 but has made substantial speed gains in Julia.
- Size
  - "Tikhonov regularization requires computation of weight matrices of dimension $N \times N$ which [...] may be unsuitable for large datasets" (Racine & Hayfield, authors of `np`).

Why Optimize?

- Speed
  - Luke Sonnet (UCLA) notes speed problems with `KRLS` starting at N ≈ 1,000 but has made substantial speed gains in Julia.
- Size
  - "Tikhonov regularization requires computation of weight matrices of dimension $N \times N$ which [...] may be unsuitable for large datasets" (Racine & Hayfield, authors of `np`).
  - Typical machines hit "cannot allocate vector" in base `R` limits at N ≈ 3,000 with `KRLS`.

Why Optimize?

- Speed
  - Luke Sonnet (UCLA) notes speed problems with `KRLS` starting at N $\approx$ 1,000 but has made substantial speed gains in Julia.
- Size
  - "Tikhonov regularization requires computation of weight matrices of dimension N x N which [...] may be unsuitable for large datasets" (Racine & Hayfield, authors of `np`).
  - Typical machines hit "cannot allocate vector" in base `R` limits at N $\approx$ 3,000 with `KRLS`.

Major Updates of bigKRLS

1. C++ integration. We re-implement most major computations in the model in C++ via Rcpp and RcppArmadillo so `bigKRLS` offers substantial speed gains.

Major Updates of bigKRLS

1. C++ integration. We re-implement most major computations in the model in C++ via Rcpp and RcppArmadillo so `bigKRLS` offers substantial speed gains.
2. Leaner algorithm. KRLS is inherently memory-heavy; `bigKRLS`'s new algorithm reduces peak memory usage by an order of magnitude.

Major Updates of bigKRLS

1. C++ integration. We re-implement most major computations in the model in C++ via Rcpp and RcppArmadillo so `bigKRLS` offers substantial speed gains.
2. Leaner algorithm. KRLS is inherently memory-heavy; `bigKRLS`'s new algorithm reduces peak memory usage by an order of magnitude.
3. Improved memory management. We use the bigmemory environment to handle large matrices better than base `R`.

Major Updates of bigKRLS

1. C++ integration. We re-implement most major computations in the model in C++ via Rcpp and RcppArmadillo so `bigKRLS` offers substantial speed gains.
2. Leaner algorithm. KRLS is inherently memory-heavy; `bigKRLS`'s new algorithm reduces peak memory usage by an order of magnitude.
3. Improved memory management. We use the bigmemory environment to handle large matrices better than base `R`.
4. Interactive data visualization. We've designed an `R` Shiny app that allows users to share results with collaborators or more general audiences.

Major Updates of bigKRLS

1. C++ integration. We re-implement most major computations in the model in C++ via Rcpp and RcppArmadillo so `bigKRLS` offers substantial speed gains.
2. Leaner algorithm. KRLS is inherently memory-heavy; `bigKRLS`'s new algorithm reduces peak memory usage by an order of magnitude.
3. Improved memory management. We use the bigmemory environment to handle large matrices better than base `R`.
4. Interactive data visualization. We've designed an `R` Shiny app that allows users to share results with collaborators or more general audiences.
5. Parallel Processing (*coming soon*). Most bigmemory calculations are best done on a single core but snow offers substantial speed gains for the derivatives.

Example: “Get Out the Vote” Experiment

To assess whether social pressure increases voter turnout, Gerber, Green, and Larimer ’08 conducted a field experiment by mail.

Example: “Get Out the Vote” Experiment

To assess whether social pressure increases voter turnout, Gerber, Green, and Larimer ’08 conducted a field experiment by mail.

- 180K registered Michigian voters in 2006
- Randomized what type of social pressure they received (if any)
- 5,074 voters received the “neighbors” treament (next slide)
- Control group of 24,964
- Targeted voters whose turnout was variable

Example: “Get Out the Vote” Experiment

To assess whether social pressure increases voter turnout, Gerber, Green, and Larimer ’08 conducted a field experiment by mail.

- 180K registered Michigian voters in 2006
- Randomized what type of social pressure they received (if any)
- 5,074 voters received the “neighbors” treament (next slide)
- Control group of 24,964
- Targeted voters whose turnout was variable

Green et al ’09 also used this as an experimental benchmark for polynomial regression.

The Treatment

Dear Registered Voter:

WHAT IF YOUR NEIGHBORS KNEW WHETHER YOU VOTED?

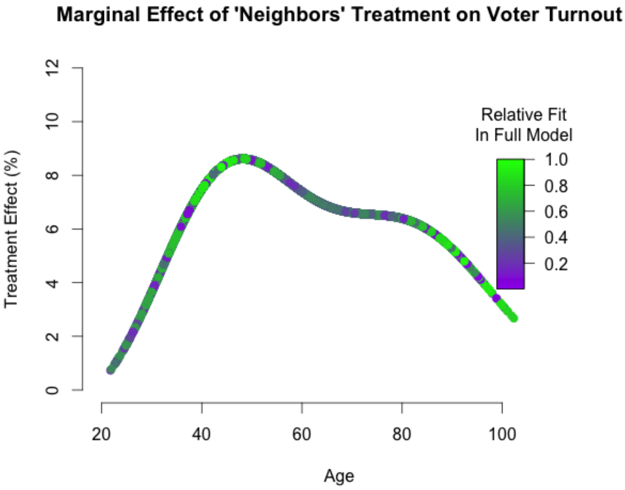
Why do so many people fail to vote? We've been talking about the problem for years, but it only seems to get worse. This year, we're taking a new approach. We're sending this mailing to you and your neighbors to publicize who does and does not vote.

The chart shows the names of some of your neighbors, showing which have voted in the past. After the August 8 election, we intend to mail an updated chart. You and your neighbors will all know who voted and who did not.

DO YOUR CIVIC DUTY — VOTE!

MAPLE DR	Aug 04	Nov 04	Aug 06
9995 JOSEPH JAMES SMITH	Voted	Voted	______
9995 JENNIFER KAY SMITH		Voted	______
9997 RICHARD B JACKSON		Voted	______
9999 KATHY MARIE JACKSON		Voted	______
9999 BRIAN JOSEPH JACKSON		Voted	______
9991 JENNIFER KAY THOMPSON		Voted	______
9991 BOB R THOMPSON		Voted	______
9993 BILL S SMITH			______
9989 WILLIAM LUKE CASPER		Voted	

Heterogeneous Treatment Effect



Analyzing Interactions

Do Americans prefer Donald Trump over Hillary Clinton?

- Data: Pew Research Center's January 2016 survey.
- DV: relative preference for Trump expressed as difference of two Likert responses:

$$\mathbf{y}_i \equiv \mathbf{Q22E}_i - \mathbf{Q22D}_i$$

- IVs: standard sociodemographics, liberalism, Obama approval, state (incl. DC), region.

Analyzing Interactions

Do Americans prefer Donald Trump over Hillary Clinton?

- Data: Pew Research Center's January 2016 survey.
- DV: relative preference for Trump expressed as difference of two Likert responses:

$$\mathbf{y}_i \equiv \mathbf{Q22E}_i - \mathbf{Q22D}_i$$

- IVs: standard sociodemographics, liberalism, Obama approval, state (incl. DC), region.

Note: costly under previous setup!

- Prohibitive given the number of categorical variables (state, race) and sample size

Trump vs. Hillary (Average Marginal Effects)

	Est	SE	t value	p
Female*	-0.289	0.079	-3.665	< 0.001
Spanish Language Interview*	-0.745	0.173	-4.317	< 0.001
Liberalism	-0.477	0.031	-15.537	< 0.001
Approve Obama*	-2.045	0.090	-22.667	< 0.001
Follows Election	0.126	0.032	3.961	< 0.001
Age	-0.001	0.002	-0.821	0.412
Bachelors*	0.010	0.064	0.151	0.880
Associates*	0.014	0.102	0.141	0.888
Some Postgrad*	-0.288	0.200	-1.441	0.150
High School*	0.064	0.072	0.877	0.380
Postgrad*	-0.197	0.095	-2.061	0.039
Some College*	0.145	0.081	1.783	0.075
Refused*	0.813	0.369	2.201	0.028
Some High School*	-0.305	0.191	-1.594	0.111
No High School*	0.112	0.249	0.450	0.652
Population Density	-0.022	0.021	-1.039	0.299
Hispanic*	-0.207	0.142	-1.461	0.144
White*	0.164	0.120	1.368	0.171
Refused*	0.219	0.219	0.998	0.318
Hispanic Latino*	-0.452	0.169	-2.669	0.008
African American*	-0.425	0.144	-2.957	0.003
Native American*	0.294	0.232	1.270	0.204
Other*	-0.766	0.333	-2.298	0.022
Asian Or Asian American*	-0.398	0.190	-2.095	0.036
Pacific Islander Or Hawaiian*	0.342	0.363	0.942	0.346
Midwest*	-0.008	0.033	-0.233	0.816
South*	0.083	0.026	3.172	0.002
Northeast*	-0.078	0.035	-2.206	0.027
West*	-0.039	0.031	-1.253	0.210

$N = 2009$. * indicates binary variable for which first differences are computed (estimates for state and DC not shown); $R^2 = 0.676$.

Quick Overview of Trump vs. Hillary Results

The model fits the data quite well...

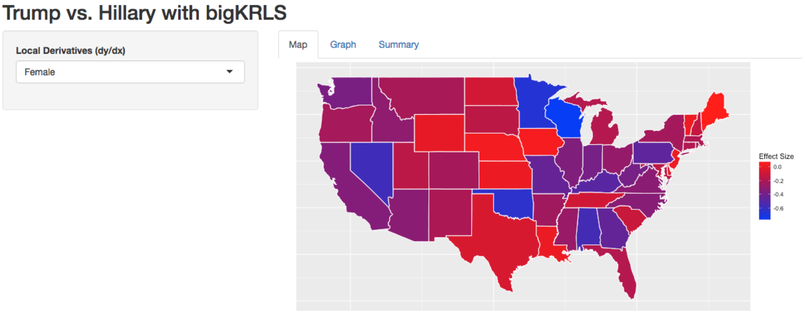
1. Even in January, long-term patterns are largely evident. Self-identified liberals, African Americans, Latinos, and women all favor Clinton, while whites (particularly in the South) tilt slightly towards Trump.

Quick Overview of Trump vs. Hillary Results

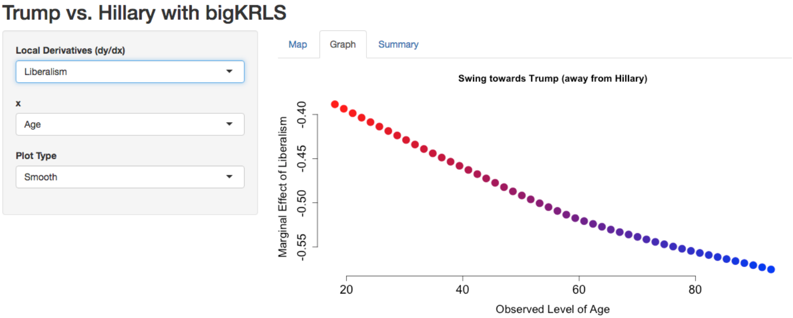
The model fits the data quite well. . .

1. Even in January, long-term patterns are largely evident. Self-identified liberals, African Americans, Latinos, and women all favor Clinton, while whites (particularly in the South) tilt slightly towards Trump.
2. $R^2 \approx 0.68$.
3. Restricting the R^2 calculation to average marginal effects suggests that the model is (primarily) linear and additive, with $R^2_{AME} \approx 0.55$.
   - Or: average marginal effects provide approximately 80% of the explanatory power in the model.
   - R2AME now reported with summary(bigKRLS).

How Does the Effect of Gender Vary by State?



The Effect of Liberalism by Age

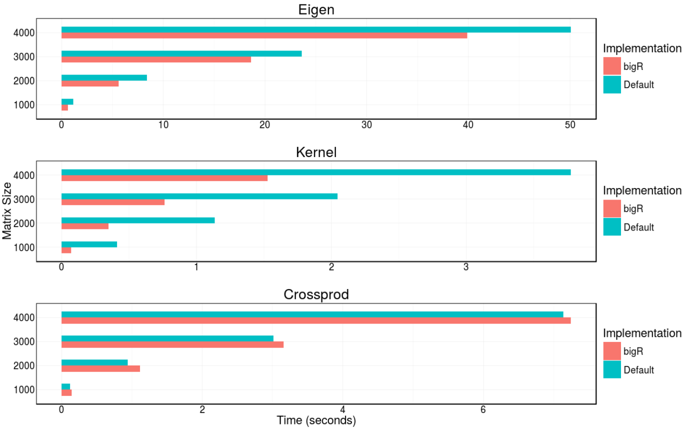


Complexity

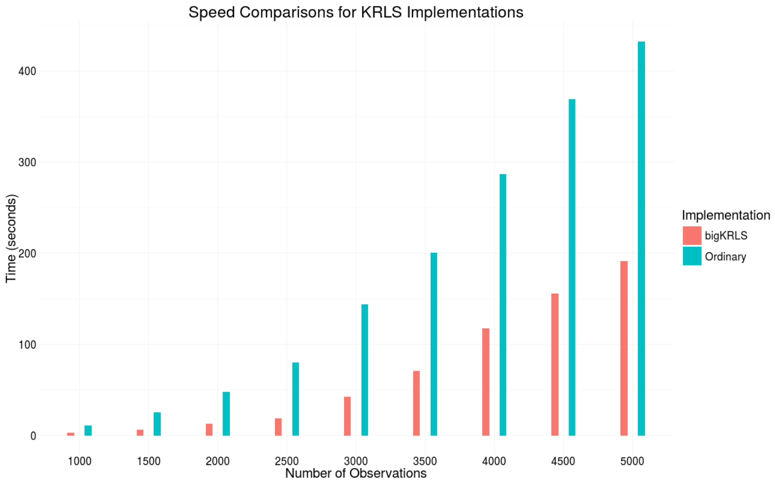
	Major Steps	Runtime	Memory
(1)	Standardize $\mathbf{X}_{\mathrm{N*P}}$, $\mathbf{y}$	—	—
(2)	Calculate kernel $\mathbf{K}_{N\times N}$	$O(N^2)$	$O(N^2)$
(3)	Eigendecompose $\mathbf{KE} = \mathbf{Ev}$	$O(N^3)$	$O(N^2)$
(4)	Regularization parameter λ	$O(N^3)$	—
(5)	Estimate weights $\hat{\mathbf{c}} = \mathbf{f}(\lambda, \mathbf{y}, \mathbf{E}, \mathbf{v})$	$O(N^3)$	—
(6)	Fit values $\hat{\mathbf{y}} = \mathbf{K}\hat{\mathbf{c}}$	—	—
(7)	Estimate local derivatives, $\hat{\mathbf{\Delta}}_{\mathbf{N*P}} = [\hat{\delta}_1 \quad \hat{\delta}_2 \ldots \hat{\delta}_{\mathbf{P}}]$	$O(PN^3)$	$O(N^2)$

Letting i, j index observations and $p = 1, 2, \ldots P$ index x variables. Steps 4-6 are followed by uncertainty estimates.

Speed Tests of Key Functions: HH's KRLS vs. bigKRLS



bigKRLS, with & without “big R” + Rcpp



First Differences

For each binary variable b, we calculate first differences $\hat{\delta}_{\mathbf{b}}$:

$$\begin{aligned}\hat{\delta}_{\mathbf{b}} &= \hat{\mathbf{y}}_{\{1\}} - \hat{\mathbf{y}}_{\{0\}} \\ &= \left(\mathbf{K}_{\{1\}} - \mathbf{K}_{\{0\}}\right) * \hat{\mathbf{c}} *\end{aligned}$$

First Differences

For each binary variable b, we calculate first differences $\hat{\delta}_{\mathbf{b}}$:

$$
\begin{aligned}
\hat{\delta}_{\mathbf{b}} &= \hat{\mathbf{y}}_{\{1\}} - \hat{\mathbf{y}}_{\{0\}} \\
&= \left(\mathbf{K}_{\{1\}} - \mathbf{K}_{\{0\}}\right) * \hat{\mathbf{c}}*
\end{aligned}
$$

Existing Algorithm

1. Construct two copies of $\mathbf{X}$ as $\mathbf{X}^{(0)}$ and $\mathbf{X}^{(1)}$; assign $\mathbf{X}_{\mathbf{b}}^{(0)} = \mathbf{0}$ and $\mathbf{X}_{\mathbf{b}}^{(1)} = \mathbf{1}$.
2. Compute $\mathbf{X}_{new} = [\mathbf{X}_{obs} \,|\, \mathbf{X}_{\mathbf{b}}^{(0)} \,|\, \mathbf{X}_{\mathbf{b}}^{(1)}]$.
   - Note $9N^2$ memory footprint!
3. Let $\mathbf{K}_{new} = [\mathbf{K}_{\{1\}} \,|\, \mathbf{K}_{\{0\}}]'$

Boolean Counterfactual Similarity

$$
\begin{aligned}
\mathbf{K_{i,j}} &= e^{-||\mathbf{x_i}-\mathbf{x_j}||^2/\sigma^2} \\
&= e^{[(\mathbf{x_{i,1}}-\mathbf{x_{j,1}})^2+(\mathbf{x_{i,2}}-\mathbf{x_{j,2}})^2+\ldots+(\mathbf{x_{i,b}}-\mathbf{x_{j,b}})^2+\ldots]} \\
&= e^{(\mathbf{x_{i,b}}-\mathbf{x_{j,b}})^2/\sigma^2} e^{[(\mathbf{x_{i,1}}-\mathbf{x_{j,1}})^2+(\mathbf{x_{i,2}}-\mathbf{x_{j,2}})^2+\ldots]} \\
&= e^{(\mathbf{x_{i,b}}-\mathbf{x_{j,b}})^2/\sigma^2} \mathbf{K^*_{i,j}} \\
&= \phi \mathbf{K^*_{i,j}}
\end{aligned}
$$

Boolean Counterfactual Similarity

$\mathbf{X}_{i,b}$	$\mathbf{X}_{j,b}$	$\mathbf{K}_{i,j}$	$\mathbf{K}_{\{1\},j}$	$\mathbf{K}_{\{0\},j}$	$\mathbf{K}_{\{1\},j} - \mathbf{K}_{\{0\},j}$
1	1	$\mathbf{K}^*_{i,j}$	$\mathbf{K}^*_{i,j}$	$\phi\mathbf{K}^*_{i,j}$	$(1-\phi) * \mathbf{K}_{i,j}$
1	0	$\phi\mathbf{K}^*_{i,j}$	$\phi\mathbf{K}^*_{i,j}$	$\mathbf{K}^*_{i,j}$	$\frac{(\phi-1)}{\phi} * \mathbf{K}_{i,j}$
0	1	$\phi\mathbf{K}^*_{i,j}$	$\mathbf{K}^*_{i,j}$	$\phi\mathbf{K}^*_{i,j}$	$\frac{(1-\phi)}{\phi} * \mathbf{K}_{i,j}$
0	0	$\mathbf{K}^*_{i,j}$	$\phi\mathbf{K}^*_{i,j}$	$\mathbf{K}^*_{i,j}$	$(\phi-1) * \mathbf{K}_{i,j}$

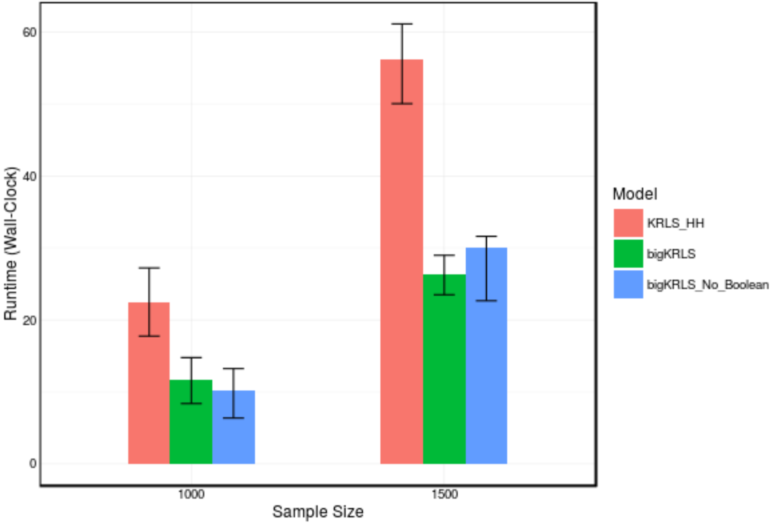
Tackling the Variance Estimator

$$(\mathbf{K}_{new}\hat{\mathbf{V}}_{\mathbf{c}^*})\mathbf{K}'_{\mathbf{new}} = \left[\mathbf{K}_{\{1\}}\mathbf{K}_{\{0\}}\right]\hat{\mathbf{V}}_{\mathbf{c}^*}\left[\begin{array}{c}\mathbf{K}'_{\{1\}}\\ \mathbf{K}'_{\{0\}}\end{array}\right]$$

$$= \begin{bmatrix}\mathbf{K}_{\{1\}}\hat{\mathbf{V}}_{\mathbf{c}^*}\mathbf{K}'_{\{\mathbf{1}\}} & \mathbf{K}_{\{1\}}\hat{\mathbf{V}}_{\mathbf{c}^*}\mathbf{K}'_{\{\mathbf{1}\}}\\ \mathbf{K}_{\{1\}}\hat{\mathbf{V}}_{\mathbf{c}^*}\mathbf{K}'_{\{\mathbf{0}\}} & \mathbf{K}_{\{0\}}\hat{\mathbf{V}}_{\mathbf{c}^*}\mathbf{K}'_{\{\mathbf{0}\}}\end{bmatrix}$$

...which allows us to factor out each individual submatrix.

Preliminary Tests of the “Boolean” Estimator



Discussion

Complexity, Interpretability, Scalability – Impossible Trilemma?

Computational Complexity Frontier

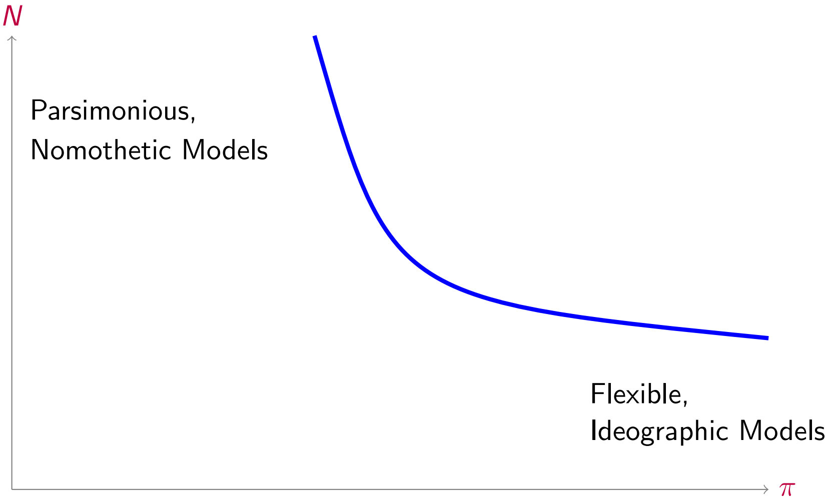


Figure: Scalability as Constraint

Conclusions

- Researchers & practioners are increasingly interested in approaches that combine desirable mathematical properties with:

Conclusions

- Researchers & practioners are increasingly interested in approaches that combine desirable mathematical properties with:
  - *flexibility*
  - *robustness*
  - *predictive accuracy*

Conclusions

- Researchers & practioners are increasingly interested in approaches that combine desirable mathematical properties with:
    - *flexibility*
    - *robustness*
    - *predictive accuracy*
- We favor approaches that prioritize *interpretability*, but models with all of these features encounter issues with *scalability*

Conclusions

- Researchers & practioners are increasingly interested in approaches that combine desirable mathematical properties with:
  - *flexibility*
  - *robustness*
  - *predictive accuracy*
- We favor approaches that prioritize *interpretability*, but models with all of these features encounter issues with *scalability*
- `bigKRLS` attempts to ease some of these constraints

Conclusions

- Researchers & practioners are increasingly interested in approaches that combine desirable mathematical properties with:
  - *flexibility*
  - *robustness*
  - *predictive accuracy*
- We favor approaches that prioritize *interpretability*, but models with all of these features encounter issues with *scalability*
- `bigKRLS` attempts to ease some of these constraints
- Directions for Future Work:
  - Integration of Data Reduction Options
  - Selective Inference

Conclusions

- Researchers & practioners are increasingly interested in approaches that combine desirable mathematical properties with:
  - *flexibility*
  - *robustness*
  - *predictive accuracy*
- We favor approaches that prioritize *interpretability*, but models with all of these features encounter issues with *scalability*
- `bigKRLS` attempts to ease some of these constraints
- Directions for Future Work:
  - Integration of Data Reduction Options
  - Selective Inference
  - Comparison with MCMC

Conclusions

- Researchers & practioners are increasingly interested in approaches that combine desirable mathematical properties with:
  - *flexibility*
  - *robustness*
  - *predictive accuracy*
- We favor approaches that prioritize *interpretability*, but models with all of these features encounter issues with *scalability*
- `bigKRLS` attempts to ease some of these constraints
- Directions for Future Work:
  - Integration of Data Reduction Options
  - Selective Inference
  - Comparison with MCMC
  - Research Questions where Geography is Particularly Important (e.g., migration, police shootings, outbreak of war)

Conclusions

- Researchers & practioners are increasingly interested in approaches that combine desirable mathematical properties with:
  - *flexibility*
  - *robustness*
  - *predictive accuracy*
- We favor approaches that prioritize *interpretability*, but models with all of these features encounter issues with *scalability*
- `bigKRLS` attempts to ease some of these constraints
- Directions for Future Work:
  - Integration of Data Reduction Options
  - Selective Inference
  - Comparison with MCMC
  - Research Questions where Geography is Particularly Important (e.g., migration, police shootings, outbreak of war)

Thanks!!

`bigKRLS` available at https://github.com/rdrr1990/bigKRLS

Pete Mohanty
Stanford University
pmohanty [at] stanford.edu

Robert B. Shaffer
University of Texas, Austin
rbshaffer [at] utexas.edu