Small-area estimation using GaussianProcess grouped IRT regression andpost-stratification

Noah Dasanaike, Jacob Montgomery, Bryant Moy & Santiago Olivella

Harvard, Washington University in St. Louis, UNC-CH

May 7, 2021

Small area estimation

- Data are often collected considering cost-representativeness trade-offs

Small area estimation

- Data are often collected considering cost-representativeness trade-offs
  - How do we overcome inferential issues induced by small sample sizes at low-levels of aggregation (e.g. socio-demographic groups in census blocks)?

Small area estimation

- Data are often collected considering cost-representativeness trade-offs
  - How do we overcome inferential issues induced by small sample sizes at low-levels of aggregation (e.g. socio-demographic groups in census blocks)?
- Multilevel-regression + post-stratification (MrP) is a popular solution

Small area estimation

- Data are often collected considering cost-representativeness trade-offs
  - How do we overcome inferential issues induced by small sample sizes at low-levels of aggregation (e.g. socio-demographic groups in census blocks)?
- Multilevel-regression + post-stratification (MrP) is a popular solution
  - Often not flexible enough

Small area estimation

- Data are often collected considering cost-representativeness trade-offs
  - How do we overcome inferential issues induced by small sample sizes at low-levels of aggregation (e.g. socio-demographic groups in census blocks)?
- Multilevel-regression + post-stratification (MrP) is a popular solution
  - Often not flexible enough
  - [Insert preferred predictive model] + Post-Stratification

Small area estimation

- Data are often collected considering cost-representativeness trade-offs
  - How do we overcome inferential issues induced by small sample sizes at low-levels of aggregation (e.g. socio-demographic groups in census blocks)?
- Multilevel-regression + post-stratification (MrP) is a popular solution
  - Often not flexible enough
  - [Insert preferred predictive model] + Post-Stratification
- MrP idea can be extended to define a latent-variable, measurement model

Small area estimation

- Data are often collected considering cost-representativeness trade-offs
    - How do we overcome inferential issues induced by small sample sizes at low-levels of aggregation (e.g. socio-demographic groups in census blocks)?
- Multilevel-regression + post-stratification (MrP) is a popular solution
    - Often not flexible enough
    - [Insert preferred predictive model] + Post-Stratification
- MrP idea can be extended to define a latent-variable, measurement model
    - E.g. Caughey & Warshaw 2017…

Small area estimation

- Data are often collected considering cost-representativeness trade-offs
  - How do we overcome inferential issues induced by small sample sizes at low-levels of aggregation (e.g. socio-demographic groups in census blocks)?
- Multilevel-regression + post-stratification (MrP) is a popular solution
  - Often not flexible enough
  - [Insert preferred predictive model] + Post-Stratification
- MrP idea can be extended to define a latent-variable, measurement model
  - E.g. Caughey & Warshaw 2017…
  - … but it can take months to get estimates back!

Our proposed approach

- We propose to define a couple of simple models based on the expressive Gaussian Process regression

Our proposed approach

- We propose to define a couple of simple models based on the expressive Gaussian Process regression
  - A grouped hierarchical IRT model

Our proposed approach

- We propose to define a couple of simple models based on the expressive Gaussian Process regression
  - A grouped hierarchical IRT model
  - A simple GP binomial regression model

Our proposed approach

- We propose to define a couple of simple models based on the expressive Gaussian Process regression
  - A grouped hierarchical IRT model
  - A simple GP binomial regression model
- We derive a fast expectation-maximization algorithm based on the a Pólya-Gamma data augmentation strategy

Our proposed approach

- We propose to define a couple of simple models based on the expressive Gaussian Process regression
  - A grouped hierarchical IRT model
  - A simple GP binomial regression model
- We derive a fast expectation-maximization algorithm based on the a Pólya-Gamma data augmentation strategy
- We implement the proposed approach in an open-source package based on R and C++.

Section 1

A Gaussian Process Primer

A distribution over functions

- Consider a Normal regression model, with $\epsilon \sim N(\mathbf{0}, \Sigma_y)$

$$\mathbf{y}_i = f(\mathbf{x}_i) + \epsilon_i$$

A distribution over functions

- Consider a Normal regression model, with $\epsilon \sim N(\mathbf{0}, \Sigma_y)$

$$\mathbf{y}_i = f(\mathbf{x}_i) + \epsilon_i$$

- Now imagine the value of each $f(\mathbf{x}_i)$ is a random variable

A distribution over functions

- Consider a Normal regression model, with $\epsilon \sim N(\mathbf{0}, \Sigma_y)$

$$\mathbf{y}_i = f(\mathbf{x}_i) + \epsilon_i$$

- Now imagine the value of each $f(\mathbf{x}_i)$ is a random variable
  - If $\{f(\mathbf{x}_i)\}$ is a (potentially infinite) collection of random variables s.t. any finite subset has a joint Gaussian distribution, then it will form a **Gaussian Process**

$$f(\mathbf{x}) \sim \mathrm{GP}(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}'))$$

A distribution over functions

- Consider a Normal regression model, with $\epsilon \sim N(\mathbf{0}, \Sigma_y)$

$$\mathbf{y}_i = f(\mathbf{x}_i) + \epsilon_i$$

- Now imagine the value of each $f(\mathbf{x}_i)$ is a random variable
    - If $\{f(\mathbf{x}_i)\}$ is a (potentially infinite) collection of random variables s.t. any finite subset has a joint Gaussian distribution, then it will form a **Gaussian Process**

$$f(\mathbf{x}) \sim \mathrm{GP}(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}'))$$

    - A prior distribution over functions!

A distribution over functions

- Consider a Normal regression model, with $\epsilon \sim N(\mathbf{0}, \Sigma_y)$

$$\mathbf{y}_i = f(\mathbf{x}_i) + \epsilon_i$$

- Now imagine the value of each $f(\mathbf{x}_i)$ is a random variable
  - If $\{f(\mathbf{x}_i)\}$ is a (potentially infinite) collection of random variables s.t. any finite subset has a joint Gaussian distribution, then it will form a **Gaussian Process**

$$f(\mathbf{x}) \sim \mathrm{GP}(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}'))$$

  - A prior distribution over functions!
- The choice of covariance (or **kernel**) determines the types of functions that this prior can generate

A distribution over functions

- Consider a Normal regression model, with $\epsilon \sim N(\mathbf{0}, \Sigma_y)$

$$\mathbf{y}_i = f(\mathbf{x}_i) + \epsilon_i$$

- Now imagine the value of each $f(\mathbf{x}_i)$ is a random variable
  - If $\{f(\mathbf{x}_i)\}$ is a (potentially infinite) collection of random variables s.t. any finite subset has a joint Gaussian distribution, then it will form a **Gaussian Process**

$$f(\mathbf{x}) \sim \mathrm{GP}(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}'))$$

  - A prior distribution over functions!
- The choice of covariance (or **kernel**) determines the types of functions that this prior can generate
  - Must be continuous, symmetric, and (preferably in optimization problems) positive definite.

Kernels and the functions they support

- Consider, for instance,

$$k(\mathbf{x}, \mathbf{x'}) = \mathbf{x}^\top \mathbf{x}$$

which induces sampled (hyper-)planes over input space.

Kernels and the functions they support

- Consider, for instance,

$$k(\mathbf{x}, \mathbf{x'}) = \mathbf{x}^\top \mathbf{x}$$

  which induces sampled (hyper-)planes over input space.
- We will be working with a version of a simple squared-exponential kernel, given by

$$k(\mathbf{x}, \mathbf{x'}) = \exp\left[-\frac{1}{2}\frac{\sum_d (x_d - x'_d)^2}{\rho}\right]$$

  which induces sampled surfaces that are more or less rugged depending on the value of the *length-scale* $\rho > 0$

Kernels and the functions they support

- Consider, for instance,

$$k(\mathbf{x}, \mathbf{x}') = \mathbf{x}^\top \mathbf{x}$$

  which induces sampled (hyper-)planes over input space.
- We will be working with a version of a simple squared-exponential kernel, given by

$$k(\mathbf{x}, \mathbf{x}') = \exp\left[-\frac{1}{2}\frac{\sum_d (x_d - x'_d)^2}{\rho}\right]$$

  which induces sampled surfaces that are more or less rugged depending on the value of the *length-scale* $\rho > 0$
- For positive-definite kernels there is and equivalent representation using (an inner product of) basis functions that map inputs onto higher dimensional spaces.

The Kernel Trick

The Kernel Trick

- Mapping inputs onto higher dimensional spaces dramatically increases the expressiveness of simple models

The Kernel Trick

- Mapping inputs onto higher dimensional spaces dramatically increases the expressiveness of simple models
  - Think of polynomial curve fitting

The Kernel Trick

- Mapping inputs onto higher dimensional spaces dramatically increases the expressiveness of simple models
  - Think of polynomial curve fitting
- The secret sauce behind GPs is that, instead of relying on explicit definitions of these (unknown) basis functions, we focus defining their corresponding kernels.

The Kernel Trick

- Mapping inputs onto higher dimensional spaces dramatically increases the expressiveness of simple models
  - Think of polynomial curve fitting
- The secret sauce behind GPs is that, instead of relying on explicit definitions of these (unknown) basis functions, we focus defining their corresponding kernels.
- Used to great success in many disciplines!

The Kernel Trick

- Mapping inputs onto higher dimensional spaces dramatically increases the expressiveness of simple models
  - Think of polynomial curve fitting
- The secret sauce behind GPs is that, instead of relying on explicit definitions of these (unknown) basis functions, we focus defining their corresponding kernels.
- Used to great success in many disciplines!
  - In political science, see e.g. Hainmueller and Hazlett (2014) or Hartman et al.'s KPop

Section 2

GrP: GP (IRT) regression and Post-stratification

The 2-PL (grouped) IRT

- **Group** of n_{ij} respondents who share the same demographic profile $i \in 1, \ldots, N$ answer survey item $j \in 1, \ldots, J$

The 2-PL (grouped) IRT

- **Group** of n_{ij} respondents who share the same demographic profile $i \in 1, \ldots, N$ answer survey item $j \in 1, \ldots, J$
    - Different numbers of respondents for any given item

The 2-PL (grouped) IRT

- **Group** of n_{ij} respondents who share the same demographic profile $i \in 1, \ldots, N$ answer survey item $j \in 1, \ldots, J$
  - Different numbers of respondents for any given item
- $0 \leq y_{ij} \leq n_{ij}$: number of respondents who answer an item in the affirmative

The 2-PL (grouped) IRT

- **Group** of n_{ij} respondents who share the same demographic profile $i \in 1, \dots, N$ answer survey item $j \in 1, \dots, J$
  - Different numbers of respondents for any given item
- $0 \leq y_{ij} \leq n_{ij}$: number of respondents who answer an item in the affirmative
- Standard 2-PL IRT:

$$
\begin{aligned}
y_{ij} &\overset{\text{indep.}}{\sim} \text{Binomial}(n_{ij}, \pi_{ij}) \\
\pi_{ij} &= \text{logit}^{-1}(\mu_{ij}) \\
\mu_{ij} &= \theta_i \beta_{j1} - \beta_{j2}
\end{aligned}
$$

The 2-PL (grouped) IRT

- **Group** of n_{ij} respondents who share the same demographic profile $i \in 1, \ldots, N$ answer survey item $j \in 1, \ldots, J$
  - Different numbers of respondents for any given item
- $0 \leq y_{ij} \leq n_{ij}$: number of respondents who answer an item in the affirmative
- Standard 2-PL IRT:

$$
\begin{aligned}
y_{ij} &\overset{\text{indep.}}{\sim} \text{Binomial}(n_{ij}, \pi_{ij}) \\
\pi_{ij} &= \text{logit}^{-1}(\mu_{ij}) \\
\mu_{ij} &= \theta_i \beta_{j1} - \beta_{j2}
\end{aligned}
$$

  - One dimensional latent trait θ_i

The 2-PL (grouped) IRT

- **Group** of n_{ij} respondents who share the same demographic profile $i \in 1, \ldots, N$ answer survey item $j \in 1, \ldots, J$
  - Different numbers of respondents for any given item
- $0 \leq y_{ij} \leq n_{ij}$: number of respondents who answer an item in the affirmative
- Standard 2-PL IRT:

$$
\begin{aligned}
y_{ij} &\overset{\text{indep.}}{\sim} \text{Binomial}(n_{ij}, \pi_{ij}) \\
\pi_{ij} &= \text{logit}^{-1}(\mu_{ij}) \\
\mu_{ij} &= \theta_i \beta_{j1} - \beta_{j2}
\end{aligned}
$$

  - One dimensional latent trait θ_i
- Gaussian prior over the vector of item parameters

$$
\boldsymbol{\beta}_j \overset{\text{iid}}{\sim} N_2(\mathbf{0}, \boldsymbol{\Lambda}_{\boldsymbol{\beta}}^{-1})
$$

GP Regression of Ideal Points

- Our main innovation: use a flexible GP regression to model ideal points as function of $\mathbf{Z}_{N \times D}$ matrix of demographic predictive features

$$\theta_i \overset{\text{indep.}}{\sim} N(f_i, 1.0)$$
$$\mathbf{f} \sim \text{GP}(\mathbf{0}, \mathbf{K}_\rho)$$

where

$$\mathbf{K}_{ij} = k(\mathbf{z}_i, \mathbf{z}_j) = \exp\left[-\frac{1}{2}\sum_{d=1}^{D}\frac{(z_{id} - z_{jd})^2}{\rho_d}\right]$$

GP Regression of Ideal Points

- Our main innovation: use a flexible GP regression to model ideal points as function of $\mathbf{Z}_{N \times D}$ matrix of demographic predictive features

$$\theta_i \overset{\text{indep.}}{\sim} N(f_i, 1.0)$$
$$\mathbf{f} \sim \text{GP}(\mathbf{0}, \mathbf{K}_\rho)$$

where

$$\mathbf{K}_{ij} = k(\mathbf{z}_i, \mathbf{z}_j) = \exp\left[-\frac{1}{2}\sum_{d=1}^{D}\frac{(z_{id} - z_{jd})^2}{\rho_d}\right]$$

- $\mathbf{Z}$ can include standard demographics...

GP Regression of Ideal Points

- Our main innovation: use a flexible GP regression to model ideal points as function of $\mathbf{Z}_{N \times D}$ matrix of demographic predictive features

$$\theta_i \overset{\text{indep.}}{\sim} N(f_i, 1.0)$$
$$\mathbf{f} \sim \text{GP}(\mathbf{0}, \mathbf{K}_\rho)$$

where

$$\mathbf{K}_{ij} = k(\mathbf{z}_i, \mathbf{z}_j) = \exp\left[-\frac{1}{2}\sum_{d=1}^{D}\frac{(z_{id} - z_{jd})^2}{\rho_d}\right]$$

- $\mathbf{Z}$ can include standard demographics...
  - but also time $\rightsquigarrow$ **non-linear time trends**!

GP Regression of Ideal Points

- Our main innovation: use a flexible GP regression to model ideal points as function of $\mathbf{Z}_{N\times D}$ matrix of demographic predictive features

$$\theta_i \overset{\text{indep.}}{\sim} N(f_i, 1.0)$$
$$\mathbf{f} \sim \mathrm{GP}(\mathbf{0}, \mathbf{K}_\rho)$$

where

$$\mathbf{K}_{ij} = k(\mathbf{z}_i, \mathbf{z}_j) = \exp\left[-\frac{1}{2}\sum_{d=1}^{D} \frac{(z_{id} - z_{jd})^2}{\rho_d}\right]$$

- $\mathbf{Z}$ can include standard demographics...
  - but also time $\rightsquigarrow$ **non-linear time trends**!
  - ...and also geo-coordinates $\rightsquigarrow$ **Kriging**!

GP Regression of Ideal Points

- Our main innovation: use a flexible GP regression to model ideal points as function of $\mathbf{Z}_{N \times D}$ matrix of demographic predictive features

$$\theta_i \overset{\text{indep.}}{\sim} N(f_i, 1.0)$$
$$\mathbf{f} \sim \text{GP}(\mathbf{0}, \mathbf{K}_\rho)$$

where

$$\mathbf{K}_{ij} = k(\mathbf{z}_i, \mathbf{z}_j) = \exp\left[-\frac{1}{2}\sum_{d=1}^{D}\frac{(z_{id} - z_{jd})^2}{\rho_d}\right]$$

- $\mathbf{Z}$ can include standard demographics...
  - but also time $\rightsquigarrow$ **non-linear time trends**!
  - ...and also geo-coordinates $\rightsquigarrow$ **Kriging**!
- Allows prediction

GP Regression of Ideal Points

- Our main innovation: use a flexible GP regression to model ideal points as function of $\mathbf{Z}_{N\times D}$ matrix of demographic predictive features

$$\theta_i \overset{\text{indep.}}{\sim} N(f_i, 1.0)$$
$$\mathbf{f} \sim \mathrm{GP}(\mathbf{0}, \mathbf{K}_\rho)$$

where

$$\mathbf{K}_{ij} = k(\mathbf{z}_i, \mathbf{z}_j) = \exp\left[-\frac{1}{2}\sum_{d=1}^{D}\frac{(z_{id} - z_{jd})^2}{\rho_d}\right]$$

- **Z** can include standard demographics...
  - but also time $\rightsquigarrow$ **non-linear time trends**!
  - ...and also geo-coordinates $\rightsquigarrow$ **Kriging**!
- Allows prediction
  - **Post-stratification**!

Pólya-Gamma Augmentation for Logit IRT

- The joint likelihood of aggregated responses $\mathbf{Y}$ is given by

$$p(\mathbf{Y} \mid \boldsymbol{\theta}, \boldsymbol{\beta}) \propto \prod_{ij} \frac{\exp(\mu_{ij})^{y_{ij}}}{[1 + \exp(\mu_{ij})]^{n_{ij}}}$$
$$= \prod_{ij} \exp \kappa_{ij}\mu_{ij} \; \mathbb{E}_{\omega_{ij}}[\exp(-\omega_{ij}\mu_{ij}^2/2)]$$

where $\kappa_{ij} = y_{ij} - n_{ij}/2$, ω_{ij} is a Pólya-Gamma (PG) random variable distributed $\mathrm{PG}(n_{ij}, 0)$, and the equality obtains from the integral identity derived by Polson & Scott (2013).

Pólya-Gamma Augmentation for Logit IRT

- The joint likelihood of aggregated responses $\mathbf{Y}$ is given by

$$p(\mathbf{Y} \mid \boldsymbol{\theta}, \boldsymbol{\beta}) \propto \prod_{ij} \frac{\exp(\mu_{ij})^{y_{ij}}}{[1 + \exp(\mu_{ij})]^{n_{ij}}}$$
$$= \prod_{ij} \exp \kappa_{ij} \mu_{ij} \; \mathbb{E}_{\omega_{ij}}[\exp(-\omega_{ij} \mu_{ij}^2 / 2)]$$

  where $\kappa_{ij} = y_{ij} - n_{ij}/2$, ω_{ij} is a Pólya-Gamma (PG) random variable distributed $\mathrm{PG}(n_{ij}, 0)$, and the equality obtains from the integral identity derived by Polson & Scott (2013).
    - Weird marginal likelihood…

Pólya-Gamma Augmentation for Logit IRT

- The joint likelihood of aggregated responses $\mathbf{Y}$ is given by

$$
\begin{aligned}
p(\mathbf{Y} \mid \boldsymbol{\theta}, \boldsymbol{\beta}) &\propto \prod_{ij} \frac{\exp(\mu_{ij})^{y_{ij}}}{[1+\exp(\mu_{ij})]^{n_{ij}}} \\
&= \prod_{ij} \exp \kappa_{ij}\mu_{ij} \; \mathbb{E}_{\omega_{ij}}[\exp(-\omega_{ij}\mu_{ij}^2/2)]
\end{aligned}
$$

  where $\kappa_{ij} = y_{ij} - n_{ij}/2$, ω_{ij} is a Pólya-Gamma (PG) random variable distributed $\mathrm{PG}(n_{ij}, 0)$, and the equality obtains from the integral identity derived by Polson & Scott (2013).
  - Weird marginal likelihood…
  - … incredibly simple conditionals! $\rightsquigarrow$ Great for EM! (See Goplerud 2019)

A fast ECM algorithm for estimation I

- We begin by deriving the conditional expectation of the log joint posterior under the posterior of ω_{ij}:

$$
\begin{aligned}
Q(\mathbf{f}, \boldsymbol{\theta}, \beta, \boldsymbol{\rho}) = \sum_{ij} \kappa_{ij}\mu_{ij} &- \mathbb{E}_{\boldsymbol{\omega}}[\omega_{ij} \mid \boldsymbol{\theta}^{(t-1)}, \boldsymbol{\beta}^{(t-1)}]\mu_{ij}^2/2 \\
&- \frac{1}{2}\Big(\log\left[\det(\mathbf{K}_\rho)\right] + \mathbf{f}^\top \mathbf{K}_\rho^{-1}\mathbf{f})\Big) \\
&- \frac{1}{2}\sum_i(\theta_i^2 - 2\theta_i f_i) \\
&- \frac{1}{2}\sum_j \boldsymbol{\beta}_j^\top \boldsymbol{\Lambda}_\beta \boldsymbol{\beta}_j + \text{const.}
\end{aligned}
$$

A fast ECM algorithm for estimation II

- E-step: Evaluate Q-function, which requires

$$\mathbb{E}_{\boldsymbol{\omega}}[\omega_{ij} \mid \boldsymbol{\theta}^{(t-1)}, \boldsymbol{\beta}^{(t-1)}] = \frac{n_{ij}}{2\mu_{ij}^{(t-1)}} \tanh\left(\mu_{ij}^{(t-1)}/2\right)$$

A fast ECM algorithm for estimation II

- E-step: Evaluate Q-function, which requires

$$\mathbb{E}_{\boldsymbol{\omega}}[\omega_{ij} \mid \boldsymbol{\theta}^{(t-1)}, \boldsymbol{\beta}^{(t-1)}] = \frac{n_{ij}}{2\mu_{ij}^{(t-1)}} \tanh\left(\mu_{ij}^{(t-1)}/2\right)$$

  - Extremely easy to compute!

A fast ECM algorithm for estimation II

- E-step: Evaluate Q-function, which requires

$$\mathbb{E}_{\boldsymbol{\omega}}[\omega_{ij} \mid \boldsymbol{\theta}^{(t-1)}, \boldsymbol{\beta}^{(t-1)}] = \frac{n_{ij}}{2\mu_{ij}^{(t-1)}} \tanh\left(\mu_{ij}^{(t-1)}/2\right)$$

  - Extremely easy to compute!
  - Hard to sample from corresponding $\mathrm{PG}(n_{ij}, \mu_{ij})$ (though check out *Ultimate* PGS by Frühwirth-Schnatter et al.!)

A fast ECM algorithm for estimation II

- Conditional M steps optimize Q function w.r.t parameters (and hyper-parameters in Kernel).

where $\boldsymbol{\Omega}_j = \text{diag}\left(\{\omega_{ij}^{(t)}\}_{i=1}^{N}\right)$, matrix $\mathbf{X}$ has rows $\mathbf{x}_i = [\theta_i^{(t-1)}, -1]$, and $\boldsymbol{\kappa}_j = [\kappa_{1j}, \ldots, \kappa_{Nj}]^\top$

A fast ECM algorithm for estimation II

- Conditional M steps optimize Q function w.r.t parameters (and hyper-parameters in Kernel).
- The update for the item parameters $\boldsymbol{\beta}_j$ is given by its conditional posterior mean,

$$\boldsymbol{\beta}_j^{(t)} = \left(\boldsymbol{\Lambda}_\beta + \mathbf{X}^\top \boldsymbol{\Omega}_j \mathbf{X}\right)^{-1} \mathbf{X}^\top \boldsymbol{\kappa}_j \tag{1}$$

where $\boldsymbol{\Omega}_j = \mathrm{diag}\left(\{\omega_{ij}^{(t)}\}_{i=1}^N\right)$, matrix $\mathbf{X}$ has rows $\mathbf{x}_i = [\theta_i^{(t-1)}, -1]$, and $\boldsymbol{\kappa}_j = [\kappa_{1j}, \ldots, \kappa_{Nj}]^\top$

A fast ECM algorithm for estimation III

- The update for ideal point θ_i is again given by a conditional posterior mean...

$$\theta_i^{(t)} = \left(1.0 + \sum_j \left(\beta_{j1}^{(t)}\right)^2\right)^{-1} \left(f_i + \sum_j \beta_{j1}^{(t)}(\kappa_{ij} + \beta_{j2}^{(t)})\right)$$

A fast ECM algorithm for estimation III

- The update for ideal point θ_i is again given by a conditional posterior mean...

$$\theta_i^{(t)} = \left(1.0 + \sum_j \left(\beta_{j1}^{(t)}\right)^2\right)^{-1} \left(f_i + \sum_j \beta_{j1}^{(t)}(\kappa_{ij} + \beta_{j2}^{(t)})\right)$$

- ... and the update for $\mathbf{f}$ is given by

$$\mathbf{f}^{(t)} = \mathbf{K}_\rho^{(t-1)} \left(\mathbf{K}_\rho^{(t-1)} + \mathbf{I}_N\right)^{-1} \boldsymbol{\theta}$$

A fast ECM algorithm for estimation IV

- Finally, we take empirical Bayes route, and optimize (marginal) Q over hyper-parameters

$$\boldsymbol{\rho}^{(t)} = \arg\max_{\rho} \left[-\frac{1}{2} (\boldsymbol{\theta}^{(t)})^\top \left(\mathbf{K}_\rho + \mathbf{I}_N \right)^{-1} \boldsymbol{\theta}^{(t)} - \frac{1}{2} \log \left| (\mathbf{K}_\rho + \mathbf{I}_N) \right| \right]$$

A fast ECM algorithm for estimation IV

- Finally, we take empirical Bayes route, and optimize (marginal) Q over hyper-parameters

$$\boldsymbol{\rho}^{(t)} = \arg\max_{\rho} \left[-\frac{1}{2} (\boldsymbol{\theta}^{(t)})^\top (\mathbf{K}_\rho + \mathbf{I}_N)^{-1} \boldsymbol{\theta}^{(t)} - \frac{1}{2} \log |(\mathbf{K}_\rho + \mathbf{I}_N)| \right]$$

  - Most computationally expensive step (requires inversion of dense $\mathbf{K}_\rho + \mathbf{I}_N$)

A fast ECM algorithm for estimation IV

- Finally, we take empirical Bayes route, and optimize (marginal) Q over hyper-parameters

$$\boldsymbol{\rho}^{(t)} = \arg\max_{\rho} \left[-\frac{1}{2} (\boldsymbol{\theta}^{(t)})^{\top} \left(\mathbf{K}_{\rho} + \mathbf{I}_N\right)^{-1} \boldsymbol{\theta}^{(t)} - \frac{1}{2} \log \left| \left(\mathbf{K}_{\rho} + \mathbf{I}_N\right) \right| \right]$$

  - Most computationally expensive step (requires inversion of dense $\mathbf{K}_{\rho} + \mathbf{I}_N$)
  - Use one pass of conjugate gradients to take single climbing step

A fast ECM algorithm for estimation IV

- Finally, we take empirical Bayes route, and optimize (marginal) Q over hyper-parameters

$$\boldsymbol{\rho}^{(t)} = \arg\max_{\rho} \left[-\frac{1}{2} (\boldsymbol{\theta}^{(t)})^{\top} \left(\mathbf{K}_{\rho} + \mathbf{I}_N \right)^{-1} \boldsymbol{\theta}^{(t)} - \frac{1}{2} \log \left| (\mathbf{K}_{\rho} + \mathbf{I}_N) \right| \right]$$

  - Most computationally expensive step (requires inversion of dense $\mathbf{K}_{\rho} + \mathbf{I}_N$)
  - Use one pass of conjugate gradients to take single climbing step
  - Define inverse Gamma prior to bound away from zero and away from large values.

Simulation results

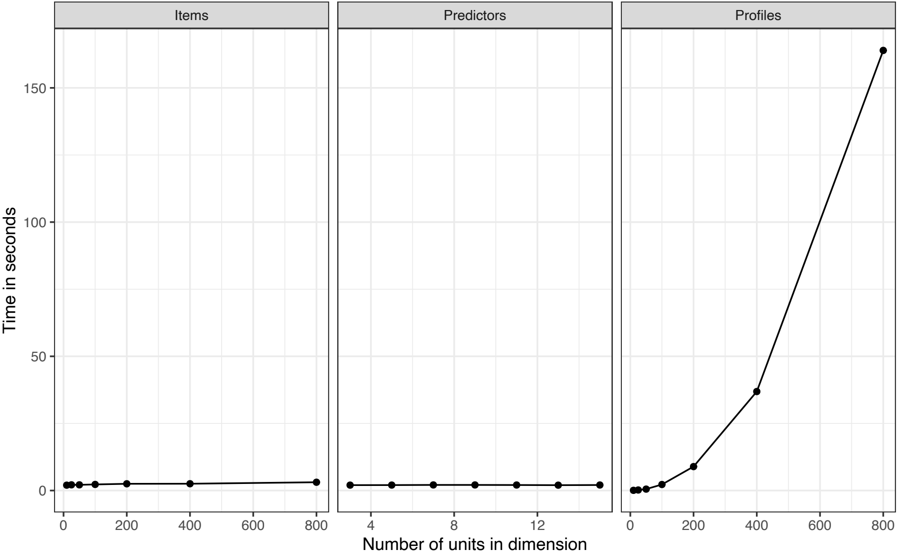


Figure 2: SimResTimes

Binomial GP Regression + Post-stratification

- For single items, we can sidestep measurement model

$$
\begin{aligned}
y_i &\overset{\text{indep.}}{\sim} \text{Binomial}(n_i, \pi_i) \\
\pi_i &= \text{logit}^{-1}(f_i) \\
\mathbf{f} &\sim N(\mathbf{0}, \mathbf{K}_\rho)
\end{aligned}
$$

Binomial GP Regression: posterior predictive

- First, we compute the posterior predictive distribution of the latent function $f_\star$ for a hypothetical demographic profile, given by

$$p(f_\star \mid \mathbf{y}, \mathbf{X}, \mathbf{x}_\star) = \int p(f_\star \mid \mathbf{X}, \mathbf{x}_\star, \mathbf{f}) p(\mathbf{f} \mid \mathbf{y}, \mathbf{X}) \, d\mathbf{f}$$

Binomial GP Regression: posterior predictive

- First, we compute the posterior predictive distribution of the latent function $f_\star$ for a hypothetical demographic profile, given by

$$p(f_\star \mid \mathbf{y}, \mathbf{X}, \mathbf{x}_\star) = \int p(f_\star \mid \mathbf{X}, \mathbf{x}_\star, \mathbf{f}) p(\mathbf{f} \mid \mathbf{y}, \mathbf{X}) \, d\mathbf{f}$$

- Then, conditional on $f_\star$, we obtain a posterior predictive expectation of the probability someone with that profile answers the item in the affirmative, $\hat{\pi}_\star$,

$$\hat{\pi}_\star = \int \text{logit}^{-1}(f_\star) p(f_\star \mid \mathbf{y}, \mathbf{X}) \, df_\star$$

Binomial GP Regression: posterior predictive

- First, we compute the posterior predictive distribution of the latent function $f_\star$ for a hypothetical demographic profile, given by

$$p(f_\star \mid \mathbf{y}, \mathbf{X}, \mathbf{x}_\star) = \int p(f_\star \mid \mathbf{X}, \mathbf{x}_\star, \mathbf{f}) p(\mathbf{f} \mid \mathbf{y}, \mathbf{X}) \, d\mathbf{f}$$

- Then, conditional on $f_\star$, we obtain a posterior predictive expectation of the probability someone with that profile answers the item in the affirmative, $\hat{\pi}_\star$,

$$\hat{\pi}_\star = \int \mathrm{logit}^{-1}(f_\star) p(f_\star \mid \mathbf{y}, \mathbf{X}) \, df_\star$$

- With a logistic inverse-link, neither integral has a simple closed-form solution.

Binomial GP Regression: posterior predictive

- First, we compute the posterior predictive distribution of the latent function $f_\star$ for a hypothetical demographic profile, given by

$$p(f_\star \mid \mathbf{y}, \mathbf{X}, \mathbf{x}_\star) = \int p(f_\star \mid \mathbf{X}, \mathbf{x}_\star, \mathbf{f}) p(\mathbf{f} \mid \mathbf{y}, \mathbf{X}) \, d\mathbf{f}$$

- Then, conditional on $f_\star$, we obtain a posterior predictive expectation of the probability someone with that profile answers the item in the affirmative, $\hat{\pi}_\star$,

$$\hat{\pi}_\star = \int \text{logit}^{-1}(f_\star) p(f_\star \mid \mathbf{y}, \mathbf{X}) \, df_\star$$

- With a logistic inverse-link, neither integral has a simple closed-form solution.
  - Use a Laplace approximation to $p(f_\star \mid \mathbf{y}, \mathbf{X}, \mathbf{x}_\star)$

Binomial GP Regression: posterior predictive

- First, we compute the posterior predictive distribution of the latent function $f_\star$ for a hypothetical demographic profile, given by

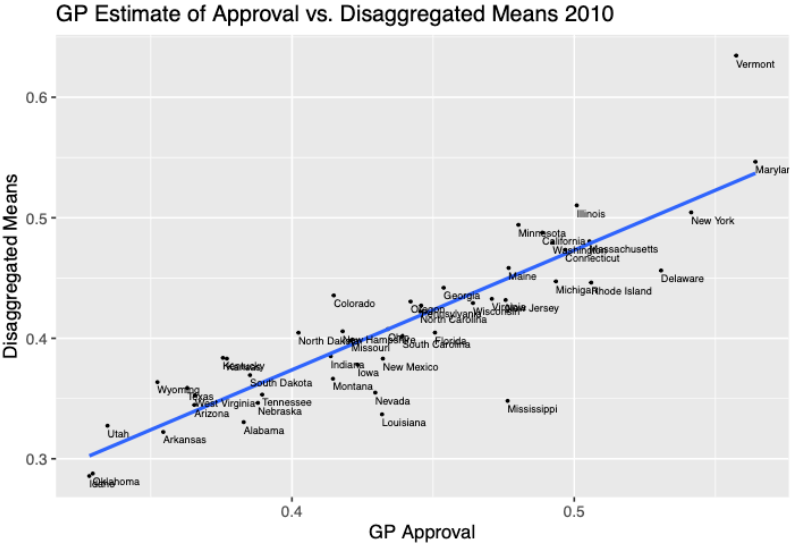
$$p(f_\star \mid \mathbf{y}, \mathbf{X}, \mathbf{x}_\star) = \int p(f_\star \mid \mathbf{X}, \mathbf{x}_\star, \mathbf{f}) p(\mathbf{f} \mid \mathbf{y}, \mathbf{X}) \, d\mathbf{f}$$

- Then, conditional on $f_\star$, we obtain a posterior predictive expectation of the probability someone with that profile answers the item in the affirmative, $\hat{\pi}_\star$,

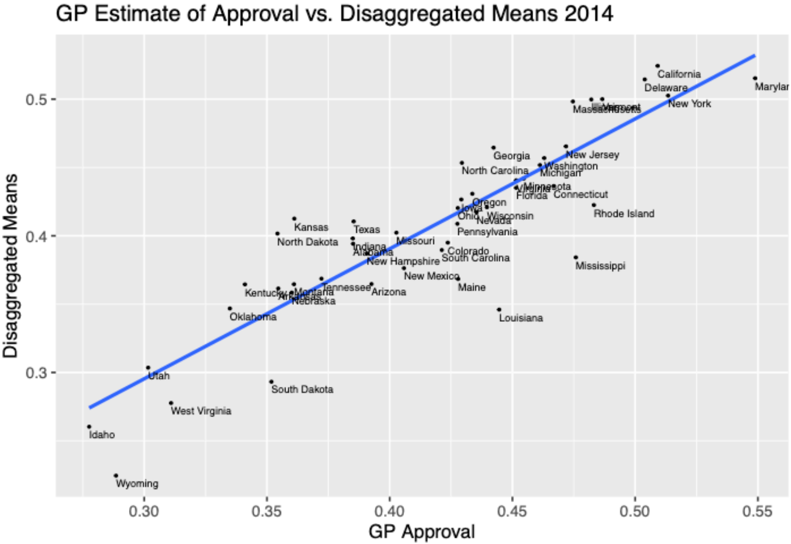
$$\hat{\pi}_\star = \int \text{logit}^{-1}(f_\star) p(f_\star \mid \mathbf{y}, \mathbf{X}) \, df_\star$$

- With a logistic inverse-link, neither integral has a simple closed-form solution.
  - Use a Laplace approximation to $p(f_\star \mid \mathbf{y}, \mathbf{X}, \mathbf{x}_\star)$
  - Empirical Bayes for hyper-parameters

Example: Obama approval 2010



Example: Obama approval 2014



Section 3

Conclusion & in-progress

A fast, flexible alternative to IRT + MrP

- Our proposed set of models offer fast, flexible alternatives to models that can take weeks to train

A fast, flexible alternative to IRT + MrP

- Our proposed set of models offer fast, flexible alternatives to models that can take weeks to train
  - Although fairly fast already, still not taking advantage of further developments to reduce $\mathcal{O}(N^3)$ complexity of inverting kernel

A fast, flexible alternative to IRT + MrP

- Our proposed set of models offer fast, flexible alternatives to models that can take weeks to train
  - Although fairly fast already, still not taking advantage of further developments to reduce $\mathcal{O}(N^3)$ complexity of inverting kernel
- User-friendly software implementation

A fast, flexible alternative to IRT + MrP

- Our proposed set of models offer fast, flexible alternatives to models that can take weeks to train
  - Although fairly fast already, still not taking advantage of further developments to reduce $\mathcal{O}(N^3)$ complexity of inverting kernel
- User-friendly software implementation
  - `GrP` package in R (in development, available at https://github.com/solivella/GPRP; send me an email and I'll add you to the private repo!)

A fast, flexible alternative to IRT + MrP

- Our proposed set of models offer fast, flexible alternatives to models that can take weeks to train
  - Although fairly fast already, still not taking advantage of further developments to reduce $\mathcal{O}(N^3)$ complexity of inverting kernel
- User-friendly software implementation
  - `GrP` package in R (in development, available at https://github.com/solivella/GPRP; send me an email and I'll add you to the private repo!)
- We are still ironing out the details of optimization of length-scale parameters