Kaeding, Matthias

Working Paper

Fast, approximate MCMC for Bayesian analysis of large data sets: A design based approach

Ruhr Economic Papers, No. 660

Provided in Cooperation with:
RWI – Leibniz-Institut für Wirtschaftsforschung, Essen

Suggested Citation: Kaeding, Matthias (2016) : Fast, approximate MCMC for Bayesian analysis of large data sets: A design based approach, Ruhr Economic Papers, No. 660, ISBN 978-3-86788-766-3, RWI - Leibniz-Institut für Wirtschaftsforschung, Essen, https://doi.org/10.4419/86788766

This Version is available at:
https://hdl.handle.net/10419/148310

RUHR
ECONOMIC PAPERS

Matthias Kaeding

Fast, Approximate MCMC for Bayesian Analysis of Large Data Sets: A Design Based Approach

#660

Imprint

Ruhr Economic Papers

Published by

Ruhr-Universität Bochum (RUB), Department of Economics
Universitätsstr. 150, 44801 Bochum, Germany

Technische Universität Dortmund, Department of Economic and Social Sciences
Vogelpothsweg 87, 44227 Dortmund, Germany

Universität Duisburg-Essen, Department of Economics
Universitätsstr. 12, 45117 Essen, Germany

RWI Leibniz-Institut für Wirtschaftsforschung
Hohenzollernstr. 1-3, 45128 Essen, Germany

Editors

Prof. Dr. Thomas K. Bauer
RUB, Department of Economics, Empirical Economics
Phone: +49 (0) 234/3 22 83 41, e-mail: thomas.bauer@rub.de

Prof. Dr. Wolfgang Leininger
Technische Universität Dortmund, Department of Economic and Social Sciences
Economics – Microeconomics
Phone: +49 (0) 231/7 55-3297, e-mail: W.Leininger@tu-dortmund.de

Prof. Dr. Volker Clausen
University of Duisburg-Essen, Department of Economics
International Economics
Phone: +49 (0) 201/1 83-3655, e-mail: vclausen@vwl.uni-due.de

Prof. Dr. Roland Döhrn, Prof. Dr. Manuel Frondel, Prof. Dr. Jochen Kluve
RWI, Phone: +49 (0) 201/81 49-213, e-mail: presse@rwi-essen.de

Editorial Office

Sabine Weiler
RWI, Phone: +49 (0) 201/81 49-213, e-mail: sabine.weiler@rwi-essen.de

Ruhr Economic Papers #660

Responsible Editor: Volker Clausen

ISSN 1864-4872 (online) – ISBN 978-3-86788-766-3

Ruhr Economic Papers #660

Matthias Kaeding

Fast, Approximate MCMC for Bayesian Analysis of Large Data Sets: A Design Based Approach

UNIVERSITÄT DUISBURG ESSEN

RWI

Bibliografische Informationen der Deutschen Nationalbibliothek

Die Deutsche Bibliothek verzeichnet diese Publikation in der deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über: *http://dnb.d-nb.de* abrufbar.

Das RWI wird vom Bund und vom Land Nordrhein-Westfalen gefördert.

http://dx.doi.org/10.4419/86788766
ISSN 1864-4872 (online)
ISBN 978-3-86788-766-3

Matthias Kaeding[1]

Fast, Approximate MCMC for Bayesian Analysis of Large Data Sets: A Design Based Approach

Abstract

We propose a fast approximate Metropolis-Hastings algorithm for large data sets embedded in a design based approach. Here, the loglikelihood ratios involved in the Metropolis-Hastings acceptance step are considered as data. The building block is one single subsample from the complete data set, so that the necessity to store the complete data set is bypassed. The subsample is taken via the cube method, a balanced sampling design, which is defined by the property that the sample mean of some auxiliary variables is close to the sample mean of the complete data set. We develop several computationally and statistically efficient estimators for the Metropolis-Hastings acceptance probability. Our simulation studies show that the approach works well and can lead to results which are close to the use of the complete data set, while being much faster. The methods are applied on a large data set consisting of all German diesel prices for the first quarter of 2015.

JEL Classification: C11, C55, C83

Keywords: Bayesian inference; big data; approximate MCMC; survey sampling

October 2016

1 *Matthias Kaeding, University of Duisburg- Essen and RWI.* – *We thank Rilana Decker and Christopher Hanck for their helpful comments and suggestions.* – *All correspondence to: Matthias Kaeding, RWI, Hohenzollernstr. 1-3, 45128 Essen, Germany, e-mail: matthias.kaeding@rwi-essen.de*

1 Introduction

Consider the update step in the Metropolis-Hastings algorithm for the simulation of the posterior distribution

$$\pi(\boldsymbol{\theta}|\mathcal{D}) \propto \mathcal{L}(\boldsymbol{\theta}|\mathcal{D})\pi(\boldsymbol{\theta}),$$

where $\boldsymbol{\theta}$ is the parameter vector, $\pi(\boldsymbol{\theta})$ is the prior, $\mathcal{D}$ is the available data and $\mathcal{L}$ is the likelihood

$$\mathcal{L}(\boldsymbol{\theta}|\mathcal{D}) = \prod_{k=1}^{N} \mathcal{L}(\boldsymbol{\theta}|\mathcal{D}_k).$$

We restrict attention to the case where the observations are independent given $\boldsymbol{\theta}$, most common under a regression framework. The update step consists of the following steps: Given the current value $\boldsymbol{\theta}^c$, a proposal $\boldsymbol{\theta}^\star$ is drawn from the proposal distribution $G(\cdot|\boldsymbol{\theta}^c)$. The current value is set to the proposal with probability

$$\alpha(\boldsymbol{\theta}^c, \boldsymbol{\theta}^\star) = 1 \wedge \frac{\mathcal{L}(\boldsymbol{\theta}^\star|\mathcal{D})\pi(\boldsymbol{\theta}^\star)G(\boldsymbol{\theta}^c|\boldsymbol{\theta}^\star)}{\mathcal{L}(\boldsymbol{\theta}^c|\mathcal{D})\pi(\boldsymbol{\theta}^c)G(\boldsymbol{\theta}^\star|\boldsymbol{\theta}^c)}. \tag{1}$$

The computational cost for the computation of $\alpha(\boldsymbol{\theta}^c, \boldsymbol{\theta}^\star)$ is linear in N. As a result, large data sets pose a problem for the algorithm. Due to the onset of large data sets, accelerating the algorithm is a highly relevant issue Green et al. (2015). The aim of this paper is to accelerate the update step by using a computationally efficient and precise design based estimator for $\alpha(\cdot, \cdot)$ using a single *fixed* subsample of the data. Compared to repeatedly subsampling the data, this approach avoids two types of overhead: (1) subsampling the data, via a possibly computationally expensive sampling algorithm (see, e.g, Quiroz et al. 2014). (2) storing and accessing the data. This is especially relevant for data sets which are too large to be read in memory.

Several proposals have been made to accelerate the algorithm by drawing a subsample for every update step: Korattikara et al. (2014) and Bardenet et al. (2014) represent the update step as a hypothesis test. Maclaurin and Adams (2014) use a completion of the posterior distribution via auxiliary variables. For their approach, a computationally cheap lower bound must be available for all likelihood contributions. Quiroz et al. (2014) use a pseudo-marginal argument where an unbiased estimator for the likelihood is needed: They construct this estimator by debiasing an exponentiated estimator for the loglikelihood, relying on a normality assumption. To the best of our knowledge, Maire et al. (2015) is the only article considering the use of subsamples which stay fixed during several iterations. We take the same perspective as these authors: Our algorithm belongs to the class of *noisy* Monte

Carlo algorithms: Here, we obtain deviates from a (close) approximation to the posterior distribution using the complete data set.

The paper is structured as follows: Section 2 gives background on noisy MCMC, Section 3 on sampling theory. Section 4 describes the proposed algorithm, Section 5 reports a simulation study. Section 6 applies the methods to a large data set on gas prices. Section 7 concludes with a discussion.

2 Noisy Metropolis-Hastings

It is convenient to use the following representation of the update step of the Metropolis-Hastings algorithm, used by Korattikara et al. (2014) and Bardenet et al. (2014): Draw $\boldsymbol{\theta}^\star \sim G(\cdot|\boldsymbol{\theta}^c)$, $u \sim Unif(0,1)$, and accept the proposal if

$$\log\left[u\frac{\pi(\boldsymbol{\theta}^c)G(\boldsymbol{\theta}^\star|\boldsymbol{\theta}^c)}{\pi(\boldsymbol{\theta}^\star)G(\boldsymbol{\theta}^c|\boldsymbol{\theta}^\star)}\right] > \dot{\phi}, \tag{2}$$

where $\dot{\phi} = \sum_{k=1}^{N}\phi_k$, is the sum of loglikelihood ratios:

$$\phi_k = \phi(\boldsymbol{\theta}^\star, \boldsymbol{\theta}^c|\mathcal{D}_k) = \log L(\boldsymbol{\theta}^\star|\mathcal{D}_k) - \log L(\boldsymbol{\theta}^c|\mathcal{D}_k).$$

The MH algorithm simulates a Markov chain with transition kernel K which is invariant under $\pi(\boldsymbol{\theta}|\mathcal{D})$. Replacing the left hand side or the right hand side in (2) by an estimator implies the use of an approximate transition kernel $\hat{K}$ which in general is not invariant under $\pi(\boldsymbol{\theta}|\mathcal{D})$. One exception is given by the pseudo-marginal approach by Andrieu and Roberts (2009) , where an unbiased estimator of the likelihood is available, used by Quiroz et al. (2014). However, getting an unbiased estimator of the likelihood in a fixed subsample context is not feasible without questionable assumptions and might lead to trading small bias with high variance.

Although the theoretical properties of noisy MCMC are not completely understood, there are some encouraging results: Alquier et al. (2014) have shown that the stationary distribution of the Markov chained obtained using $\widehat{\alpha(\cdot,\cdot)}$ is an useful approximation to $\pi(\boldsymbol{\theta}|\mathcal{D})$, provided that $|\alpha(\cdot,\cdot) - \widehat{\alpha(\cdot,\cdot)}|$ is bounded. Nicholls et al. (2012) show that, if a Gaussian unbiased estimator for $\log\pi(\boldsymbol{\theta}^\star|\mathcal{D}) - \log\pi(\boldsymbol{\theta}^c|\mathcal{D})$ with known variance is available, the update step of the Metropolis-Hastings algorithm can be adjusted via a method called *the penalty method*, so that the chain targets $\pi(\boldsymbol{\theta}|\mathcal{D})$ as desired. Usually, such an estimator is not available. Nicholls et al. (2012) show that plugging an estimate of

Algorithm 1 Metropolis Hastings update step for posterior simulation

Draw u from $U(0,1)$
Set ρ to $\log[u\frac{\pi(\boldsymbol{\theta}^c)G(\boldsymbol{\theta}^\star|\boldsymbol{\theta}^c)}{\pi(\boldsymbol{\theta}^\star)G(\boldsymbol{\theta}^c|\boldsymbol{\theta}^\star)}]$
if $\rho > \dot{\phi}$ **then**
  Set $\boldsymbol{\theta}^c$ to $\boldsymbol{\theta}^\star$
end if
return $\boldsymbol{\theta}^c$

Algorithm 2 Metropolis Hastings algorithm for posterior simulation, generic version

for $r = 1$ **to** $R-1$ **do**
  Draw $\boldsymbol{\theta}^\star$ from $G(\cdot|\boldsymbol{\theta}^c)$
  Set $\dot{\phi}$ to $\sum_{i=1}^N L_i(\boldsymbol{\theta}^\star|\mathcal{D}_i) - L_i(\boldsymbol{\theta}^c|\mathcal{D}_i)$
  Do Metropolis-Hastings update step for posterior simulation, using $\boldsymbol{\theta}^\star, \dot{\phi}$
  Set $\boldsymbol{\theta}^{(r)}$ to $\boldsymbol{\theta}^c$
end for
return $\boldsymbol{\theta}^{(0)}, \ldots, \boldsymbol{\theta}^{(R-1)}$

$\log \pi(\boldsymbol{\theta}^\star|\mathcal{D}) - \log \pi(\boldsymbol{\theta}^c|\mathcal{D})$ into α leads to a chain which is very close to the penalty method, provided that the expected absolute error $E|\alpha(\cdot,\cdot) - \widehat{\alpha(\cdot,\cdot)}|$ is small. As such, the main objective here is to find a computationally cheap estimator for $\dot{\phi}$, so that $|\alpha(\cdot,\cdot) - \widehat{\alpha(\cdot,\cdot)}|$ is small.

3 Some sampling theory

Let $\mathcal{U} = \{1, \ldots, N\}$ be a set of labels associated with a finite population of N units, here given by the complete data set. The aim of survey-sampling is to estimate a finite population total, $\dot{y} = \sum_{k\in\mathcal{U}} y_k$ of a variable y, using a sample $\mathcal{S} \subseteq \mathcal{U}$. Totals are denoted by dots. The set $\mathcal{S}$ is selected via a stochastic selection scheme, called the *sampling design*. Design based sampling is used when the cost of obtaining the value y_k for all units $k \in \mathcal{U}$ is too expensive, so the sample size $n \coloneqq |\mathcal{S}|$ is usually much smaller than N. For this article, cost is given by computation time. In a design based approach, all randomization is due to the sampling design, while the values $y_1, \ldots, y_N$ of the study variable are unknown constants.

A sample can be represented by a random vector

$$\boldsymbol{i} = (i_1, \ldots, i_k, \ldots, i_N)^\top,$$

where i_k takes the value 1 if unit k is in the sample $\mathcal{S}$ and 0 otherwise. The sampling design $f(\cdot)$ is a probability distribution on a support $\mathcal{Q}$, so that

$$E_f[\boldsymbol{i}] = \sum_{\boldsymbol{i} \in \mathcal{Q}} \boldsymbol{i} f(\boldsymbol{i}) =: \boldsymbol{\eta},$$

where the *inclusion probability* $P(k \in \mathcal{S}) = P(i_k = 1)$ of an element k is given by η_k. Here, attention is restricted to without-replacement sampling designs with fixed sample size, so that

$$\mathcal{Q} = \Big\{ \boldsymbol{i} \in \{0,1\}^N | \sum_{k \in \mathcal{U}} i_k = n \Big\}.$$

Note that the empty sample $\boldsymbol{i} = (0, \ldots, 0)^\top$ and the census $\boldsymbol{i} = (1, \ldots, 1)^\top$ are in $\mathcal{Q}$. For more on sampling theory see Tillê (2006) or Särndal et al. (2003).

3.1 Sampling design: Cube sampling

Denoting sums of the form $\sum_{k \in \mathcal{S}} k$ as $\sum_{\mathcal{S}} k$, the benchmark estimator for a total $\dot{y}$ is the Horvitz-Thompson estimator:

$$\hat{\dot{y}}^{HT} = \sum_{\mathcal{U}} \frac{i_k y_k}{\eta_k} = \sum_{\mathcal{S}} y_k d_k,$$

where $d_k = 1/\eta_k$ is called the design weight. Here, only sampling designs are used where all inclusion probabilities are equal, so that without loss of generality $\eta_k = n/N$ for $k \in \mathcal{U}$ and

$$\hat{\dot{y}}^{HT} = \frac{N}{n} \sum_{\mathcal{S}} y_k = N \bar{y},$$

where $\bar{y}$ is the sample mean of y.

The variance of the Horwitz-Thompson estimator can be lowered by exploiting additional information: Assume $\boldsymbol{z}_{i,k}$, $i \in \{cube, greg\}$ is known for all elements in the population, where $\boldsymbol{z}_{i,k} = (z_{i,k,1}, \ldots, z_{i,k,p_i})^\top$ is a vector of values of p_i auxiliary variable for unit k. The auxiliary variables $z_{cube,1}, \ldots, z_{cube,p_{cube}}$ and $z_{greg,1}, \ldots, z_{greg,p_{greg}}$ may intersect and may be identical or distinct. Let $\dot{\boldsymbol{z}}_{cube} = \sum_{\mathcal{U}} \boldsymbol{z}_{cube,k}$ denote the known total of the auxiliary variables used for cube sampling. The objective of balanced sampling is to obtain a sample, respecting the vector of inclusion probabilities $\boldsymbol{\eta}$, so that the constraint

$$\hat{\dot{\boldsymbol{z}}}^{HT}_{cube} = \dot{\boldsymbol{z}}_{cube} \tag{3}$$

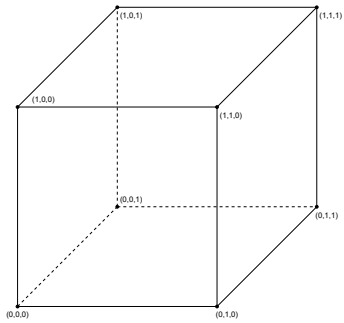

Figure 1: Set of all samples as represented as cube, for the case $N = 3$, giving $2^3 = 8$ possible samples.

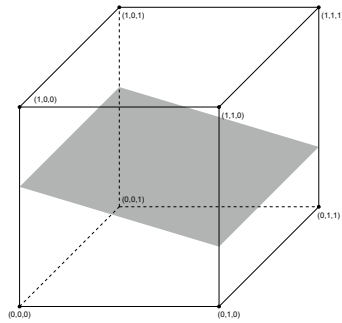

Figure 2: Set of all samples with constraint space. In this case, there are no samples in the constraint space.

holds. For equal inclusion probability designs as used here, this is equivalent to the constraint

$$n^{-1} \sum_{\mathcal{S}} \boldsymbol{z}_{cube,k} = N^{-1} \sum_{\mathcal{U}} \boldsymbol{z}_{cube,k},$$

so that the sample mean of the auxiliary variables used for cube sampling is equal to their population mean. Usually, an exact solution is not possible for all auxiliary variables, so that samples are found which satisfy (3) approximately. Here, the role of the balanced sampling algorithm is to find a sample which is "close" to the complete data set. The choice of auxiliary variables is discussed in section 4. We will use the cube method Deville and Till (2004), which is based on a geometrical representation of a sampling design: Let $\mathcal{C} = [0, 1]^N$ denote a cube equipped with 2^N vertices. $\mathcal{C}$ represents the set of all 2^N possible samples from a population of size N, where every vertex of $\mathcal{C}$ is associated with a sample, see figure 1.

Define the vector $\boldsymbol{a}_k \coloneqq \boldsymbol{z}_{cube,k}/\eta_k$ and the $p_{cube} \times N$ matrix $\boldsymbol{A} \coloneqq (\boldsymbol{a}_1, .., \boldsymbol{a}_k, \ldots, \boldsymbol{a}_N)$, Then the balancing equations (3) can be written as

$$\boldsymbol{A}\boldsymbol{i} = \boldsymbol{A}\boldsymbol{\eta}. \tag{4}$$

The system of equations (4) defines the hyperplane

$$\mathcal{P} = \boldsymbol{\eta} + K(\boldsymbol{A})$$

in $\mathcal{R}^N$, where $K(\boldsymbol{A})$ is the kernel of A. The basic idea of cube sampling is to choose a sample as a vertex of $\mathcal{C}$ in $\mathcal{P}$ or, if that is not possible, near $\mathcal{P}$. Cube sampling consists of two phases: The *flight phase* is a martingale with initial value $\boldsymbol{\eta}$ in the constraint space $\mathcal{K} = \mathcal{C} \cap \mathcal{P}$, where the constraint (3) is satisfied. At the end of the flight phase a vector $\boldsymbol{i}^\star$ is obtained. If all elements of $\boldsymbol{i}^\star$ are either 1 or 0, the *landing phase* is not necessary as a vertex of $\mathcal{C}$ is reached. If this is not the case, the balancing equations are relaxed and the elements of $\boldsymbol{i}^\star$ are randomly rounded so that

$$E[\boldsymbol{i}] = \boldsymbol{\eta},$$

hence, given inclusion probabilities are respected. Deville and Till (2004) show that it is always possible to satisfy one balancing equation. It holds that $\sum_{\mathcal{U}} \eta_k = n$. As such, setting $\boldsymbol{z}_{cube,1} = \boldsymbol{\eta}$ guarantees a fixed sample size, as the balancing equations are relaxed starting with the last auxiliary variable $z_{cube,p_{cube}}$. The authors show that the cube method achieves the bound

$$\frac{|\dot{z}_{cube,j} - \hat{\dot{z}}^{HT}_{cube,j}|}{N} = O(p_{cube}/n), \text{ for } j = 1, \ldots, p_{cube}, \tag{5}$$

where $f = O(g)$ if there is an upper bound of $|f|$ which is proportional to g. Due to the bound (5), the error becomes small if the sample size is large, compared to the number of auxiliary variables. We will use the implementation by Chauvet and Tillé (2006) with computational cost linear in N.

3.2 Regression estimator

The Horvitz-Thompson estimator is the only estimator which is unbiased for all sampling designs. However, there are better estimators in an MSE-sense if this condition is relaxed. A large classe is given by the generalized regression estimator (greg). Model the study variable y via

$$y_k = \boldsymbol{\beta}^\top \boldsymbol{z}_{greg,k} + \epsilon_k, \ \epsilon_k \sim N(0, \omega^2).$$

The generalized regression estimator $\hat{y}^{greg}$ for a total $\dot{y}$ is given by the sum of fitted values plus the Horvitz-Thompson estimator for the prediction error:

$$\hat{y}^{greg} = \sum_{\mathcal{U}} \hat{y}_k + \hat{e}^{HT} \tag{6}$$

$$= \hat{\boldsymbol{\beta}}^\top \dot{\boldsymbol{z}}_{greg} + \frac{N}{n} \sum_{\mathcal{S}} (y_k - \hat{y}_k), \tag{7}$$

where $\hat{y}_k = \boldsymbol{z}_k^\top \hat{\boldsymbol{\beta}}$ is the fitted value of element k, $\hat{\boldsymbol{\beta}}$ is

$$\hat{\boldsymbol{\beta}} = \left(\sum_{\mathcal{S}} \boldsymbol{z}_{greg,k} \boldsymbol{z}_{greg,k}^\top \right)^{-1} \sum_{\mathcal{S}} y_k \boldsymbol{z}_{greg,k},$$

and e_k is the residual $e_k = y_k - \hat{y}_k$. The generalized regression estimator can also be written as weighted sum $\sum_{\mathcal{S}} w_k y_k$, where the weights

$$w_k = N/n \Big[1 + (N/n) (\dot{\boldsymbol{z}}_{greg} - \hat{\dot{\boldsymbol{z}}}_{greg}^{HT})^\top \times \left(\sum_{\mathcal{S}} \boldsymbol{z}_{greg,k} \boldsymbol{z}_{greg,k}^\top \right) \boldsymbol{z}_{greg,k} \Big], \text{ for } k \in \mathcal{S}, \tag{8}$$

depend on the sample but not on the study variable. Hence, the same weight vector can be used for several variables, so that the estimator is cheap to compute. The greg weights (8) satisfy the calibration equation

$$\sum_{\mathcal{S}} w_k \boldsymbol{z}_{greg,k} = \dot{\boldsymbol{z}}_{greg}.$$

For equal inclusion probabilities, the approximate variance of the generalized regression estimator is, up to a constant, given by the sum $\sum_{\mathcal{U}} (y_i - \boldsymbol{z}_{greg,k}^\top \boldsymbol{b})^2$, where

$$\boldsymbol{b} = \left(\sum_{\mathcal{U}} \boldsymbol{z}_{greg,k} \boldsymbol{z}_{greg,k}^\top \right)^{-1} \sum_{\mathcal{U}} y_k \boldsymbol{z}_{greg,k}.$$

As such, predictive power for y is the main requirement for the auxiliary variables.

4 Description of algorithm

Algorithm 3 Approximate Metropolis-Hastings - phase 1

Select subsect $\mathcal{S} \subset \mathcal{U}$ via cube method, using auxiliary variables $z_{cube,1}, ..., z_{cube,p_{cube}}$
Set i to 1
for $q = 1$ to $p_{\text{greg}} \times 100$ **do**
  Draw $\boldsymbol{\theta}^\star$ from $G(\cdot|\boldsymbol{\theta}^c)$
  Set $\dot{\phi}$ to $\sum_{\mathcal{S}} \phi_k$
  Do Metropolis-Hastings update step for posterior simulation, using $\boldsymbol{\theta}^\star, \dot{\phi}$
  Set $\boldsymbol{\theta}^{(q)}$ to $\boldsymbol{\theta}^c$
  if q modulo $100 = 0$ **then**
    for $k \in \mathcal{U}$ **do**
      Set $z_{greg,k,i}$ to ϕ_k
    end for
    Set i to $i + 1$
  end if
end for

Algorithm 4 Approximate Metropolis-Hastings - greg, basic version

Do approximate Metropolis-Hastings - phase 1
for $k \in \mathcal{S}$ **do**
  Compute w_k
end for
for $r = 1$ to R **do**
  Draw $\boldsymbol{\theta}^\star$ from $G(\cdot|\boldsymbol{\theta}^c)$
  for $k \in \mathcal{S}$ **do**
    Compute ϕ_k
  end for
  Set $\hat{\dot{\phi}}$ to $\sum_{\mathcal{S}} w_k \phi_k$
  Do Metropolis-Hastings update step for posterior simulation, using $\boldsymbol{\theta}^\star, \hat{\dot{\phi}}$
  Set $\boldsymbol{\theta}^r$ to $\boldsymbol{\theta}^c$
end for
return $\boldsymbol{\theta}^{(0)}, \ldots, \boldsymbol{\theta}^{(R-1)}$

The objective is to estimate the total $\dot{\phi} = \sum_{\mathcal{U}} \phi_k$. The basic idea is to combine a sample obtained by cube-sampling with a regression estimator.

Denote the posterior distribution associated with a subsample $\mathcal{S}$ by $\pi_{\mathcal{S}}$. The algorithm consists of two phases. In the first phase, a sample is chosen via cube sampling, so that the posterior distribution from this subsample is close to π. To find a measure for closeness, the result of Maire et al. (2015) is used. These authors have shown that the minimal Kullback-Leibler distance from $\pi_{\mathcal{S}}$ to $\pi_{\mathcal{U}}$ is minimized if the sufficient statistics for $\boldsymbol{\theta}$ in $\mathcal{S}$ are equal to the sufficient statistics in the population. For many nontrivial cases, there exists no such sufficient statistic which can also be written as sum (as would be necessary for the use of cube sampling). However, the result serves as a general guide, in that statistics which summarize the complete data $\mathcal{D}$ set are used. Such statistics can be derived from

Algorithm 5 Approximate Metropolis-Hastings - greg, ridge variant

Do approximate Metropolis-Hastings - phase 1
for $r = 1$ to R **do**
  Draw $\boldsymbol{\theta}^\star$ from $G(\cdot|\boldsymbol{\theta}^c)$
  for $k \in \mathcal{S}$ **do**
    Compute ϕ_k
  end for
  Set $\boldsymbol{c}$ to $\boldsymbol{M}^\top \boldsymbol{\phi}$
  for $i = 1$ **to** p_{greg} **do**
    $\hat{\alpha}_i \leftarrow c_i/\lambda_i$
  end for
  Set $\hat{\sigma}^2$ to $(n - p_{greg})^{-1}(\boldsymbol{\phi} - \boldsymbol{M}\hat{\boldsymbol{\alpha}})^\top(\boldsymbol{\phi} - \boldsymbol{M}\hat{\boldsymbol{\alpha}})$
  Set $\hat{\kappa}$ to $\frac{p_{greg}\hat{\sigma}^2}{\hat{\boldsymbol{\xi}}^\top\hat{\boldsymbol{\xi}}}$
  for $i = 1$ **to** p_{greg} **do**
    Set $\hat{\xi}_i(\hat{\kappa})$ to $c_i/(\lambda_i + \hat{\kappa})$
  end for
  Set $\hat{\dot{\phi}}$ to $\boldsymbol{\xi}(\hat{\kappa})^\top \tilde{\dot{\boldsymbol{z}}}_{greg} + (N/n)\left((\sum_{\mathcal{S}} \phi_k) - \boldsymbol{\xi}(\hat{\kappa})^\top \tilde{\dot{\boldsymbol{z}}}_{greg,\mathcal{S}}\right)$
  Do Metropolis-Hastings update step for posterior simulation, using $\boldsymbol{\theta}^\star$, $\hat{\dot{\phi}}$
end for
return $\boldsymbol{\theta}^{(0)}, \ldots, \boldsymbol{\theta}^{(R-1)}$

inspection of the posterior distribution. Note that the computation time of cube sampling algorithm scales badly with the number of auxiliary variables, so that a low number of auxiliary variables for the cube sampling algorithm is preferable. Also during the first phase, auxiliary variables for the regression estimator are computed. A good predictor for the value of $\phi(\boldsymbol{\theta}^\star, \boldsymbol{\theta}^c|\mathcal{D}_k)$ is given by $\phi(\cdot, \cdot|\mathcal{D}_k)$, for arguments near $\boldsymbol{\theta}^\star$ and $\boldsymbol{\theta}^c$. This is used to obtain the auxiliary variables for the regression estimator: If the subsample is obtained, the posterior distribution $\pi_{\mathcal{S}}$ is an overdispersed approximation to π so that the support of $\boldsymbol{\theta}$ is covered by $\pi_{\mathcal{S}}$. The auxiliary variables for the regression estimator are subsequently computed as followed: A regular Metropolis-Hastings algorithm is run using the subsample for $100p_{cube}$ iterations. Every 100th iteration, the loglikelihood ratio obtained from the current and proposed value of $\boldsymbol{\theta}$ is used as auxiliary variables, so that the auxiliary variables are computed as

$$z_{greg,k,j} = \phi(\boldsymbol{\theta}^\star, \boldsymbol{\theta}^{(r)}|\mathcal{D}_k),$$

for $k \in \mathcal{S}, r = 100, 200, \ldots, 100p_{greg}$, where $j = r/100$ varies with r.

The complete data set only has to be available for the first phase. For the second phase, the Metropolis-Hastings algorithm proceeds using the subsample with $\dot{\phi}$ replaced by an estimator using

the derived auxiliary variables $z_{greg,1}, ..., z_{greg,p_{greg}}$. We use the posterior mode from the trial run of the first phase as starting values.

4.1 Ridge variant

The GREG estimator can be improved by improving the predictions for the values of ϕ_k. We use the ridge estimator to achieve better prediction performance. This estimator has the following advantages: (1) It is available in closed form[1], as

$$\hat{\boldsymbol{\beta}}(\kappa) = \left(\sum_{\mathcal{S}} \boldsymbol{z}_{greg,k} \boldsymbol{z}_{greg,k}^\top + \boldsymbol{I}\kappa \right)^{-1} \sum_{\mathcal{S}} y_k \boldsymbol{z}_{greg,k}.$$

(2) Shrinkage is controlled by a single parameter κ. (3) The complete data posterior distribution is better covered with a higher number of auxiliary variable. This results in multicollinearity, as the auxiliary variables $z_{greg,1}, ..., z_{greg,p_{greg}}$ are strongly correlated, so that the matrix $\sum_{\mathcal{S}} \boldsymbol{z}_{greg,k} \boldsymbol{z}_{greg,k}^\top$ is ill conditioned for large p_{greg}. However, this is a desired property, as it indicates strong correlation with future values of $\phi(\cdot, \cdot|\mathcal{D}_k), k \in \mathcal{U}$. Hence, multicollinearity is a consequence of a set of useful auxiliary variables, which in turn is solved by the ridge estimator.

The third point suggests a simple heuristic to set p_{greg}: It is set at least so large that the matrix $\sum_{\mathcal{S}} \boldsymbol{z}_{greg,k} \boldsymbol{z}_{greg,k}^\top$ is ill conditioned. Following Hoerl et al. (1975), the shrinkage parameter κ is set as

$$\kappa \coloneqq \frac{k\hat{\sigma}^2}{\hat{\boldsymbol{\beta}}^\top \hat{\boldsymbol{\beta}}},$$

where $\hat{\sigma}^2$ is an estimate for $var(\phi|z_{greg,1}, ..., z_{greg,p_{greg}})$. While determining κ via cross validation might be preferable, using a closed form expression is much faster. In addition, cross-validation is used to estimate the out-of sample prediction error. However, the procedure used here is based on the assumption that the subsample is similar to the full data set. Given κ, the ridge variant of the greg estimator can be quickly computed using precomputed entities. For details see appendix A.1.

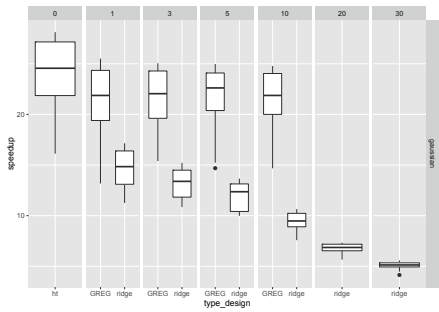

Figure 3: Speed up relative to the use of the complete data set for the gaussian likelihood. Columns: Varying values of of p_{greg}.

5 Simulation study

5.1 Setup

Here, results of a simulation study will be reported. We generated $M = 100$ data sets from the model

$$y_k = \boldsymbol{x}_k^\top \boldsymbol{\beta} + \epsilon_k, \epsilon_k \sim \mathcal{N}(0, \sigma^2), k = 1, \ldots, N = 100000,$$

where $\boldsymbol{x}_1 = (1, \ldots, 1)^\top$ and the elements of $\boldsymbol{x}_2, ..., \boldsymbol{x}_5$ are draws from an uniform distribution with minimum and maximum given by -2 and 2. The elements of the parameter vector $\boldsymbol{\theta} = (\boldsymbol{\beta}^\top, \log \sigma)^\top$ are deviates from a standard normal distribution. The posterior distribution is simulated using the algorithm described in section 4. We use a random walk Metropolis-Hastings sampler, where the variance of the proposal distribution is scaled during burnin so that the acceptance probability is around 0.234. We use $r = 100000$ iterations, taking the first half of the chain as burnin. For reference, we also simulate the posterior distribution using the complete data set and use a variant where no auxiliary variables are used, so that $\dot{\phi}$ is estimated via the sample mean. To study the gain of the cube sampling procedure, the sample was additionally chosen using simple random sampling (si); which is equivalent to cube sampling with a single auxiliary variable given by the inclusion probabilities. We use $p_{greg} = 1, 3, 5, 10, 20, 30$ auxiliary variables for the regression estimator. For the basic greg estimator, it was not possible to set $p_{greg} \geq 20$ due to multicollinearity. In addition, we generated data sets from the model $y_k \sim Poisson(\exp(\boldsymbol{x}_k^\top \boldsymbol{\beta}))$, with the same configurations as above otherwise. For both models, the auxiliary variables used for cube sampling are derived from

[1]Unlike e.g. the LASSO estimator, which might be preferable from a prediction perspective.

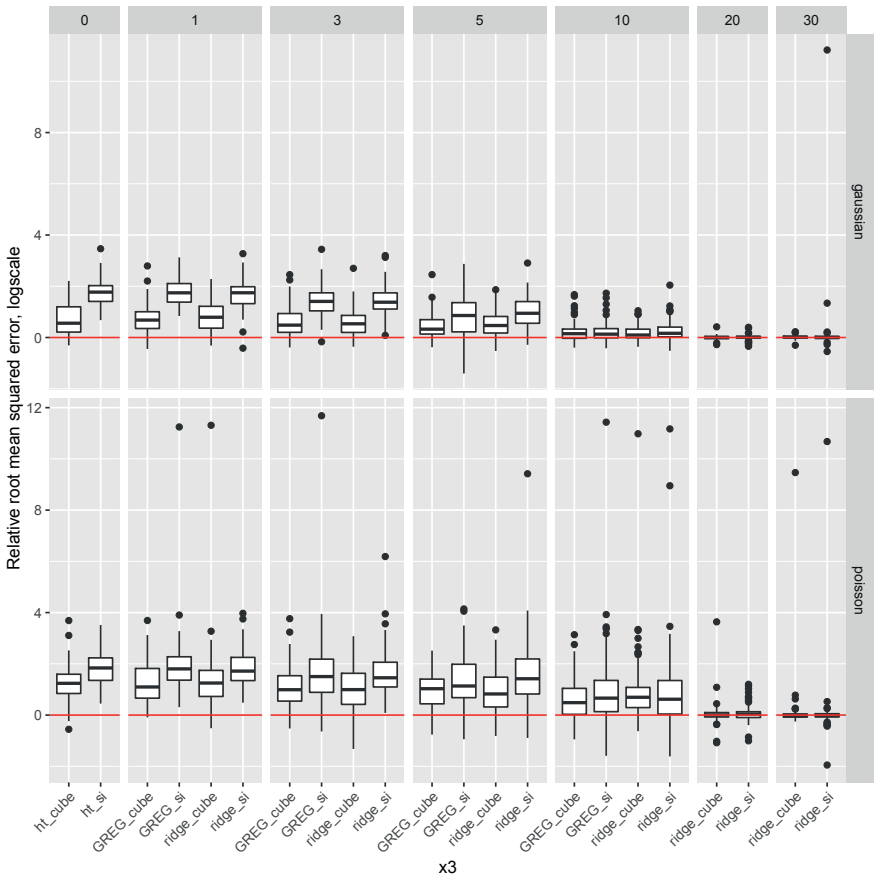

Figure 4: Root mean squared error for $\boldsymbol{\theta}$, relative to root mean squared error obtained with complete data set, plotted on log scale for visualization. Rows: Gaussian and poisson likelihood. Columns: Varying values of of p_{greg}.

the likelihood: For the gaussian likelihood, we take the residuals and the squared residuals obtained from the maximum likelihood estimator $\boldsymbol{\beta}_{ml} = (\boldsymbol{X}^\top \boldsymbol{X})^{-1}\boldsymbol{X}^\top \boldsymbol{y}$ as $z_{cube,1}$ and $z_{cube,2}$. In addition, we generate three draws from $N(\beta_{ls}, 3\hat{\sigma}^2 \boldsymbol{X}^\top \boldsymbol{X})^{-1})$, where $\hat{\sigma}^2 = N^{-1}\sum_{\mathcal{U}}(y_k - \boldsymbol{x}_k^\top \boldsymbol{\beta})^2$ and use the associated residuals (and the squared residuals) as auxiliary variables. For the poisson likelihood, we use the loglikelihood kernel $y_k \boldsymbol{x}_k^\top \boldsymbol{\beta} - \exp(\boldsymbol{x}_k^\top \boldsymbol{\beta})$ as auxiliary variables. The maximum likelihood estimator is obtained via numerical methods, otherwise values of $\boldsymbol{\beta}$ were obtained as above.

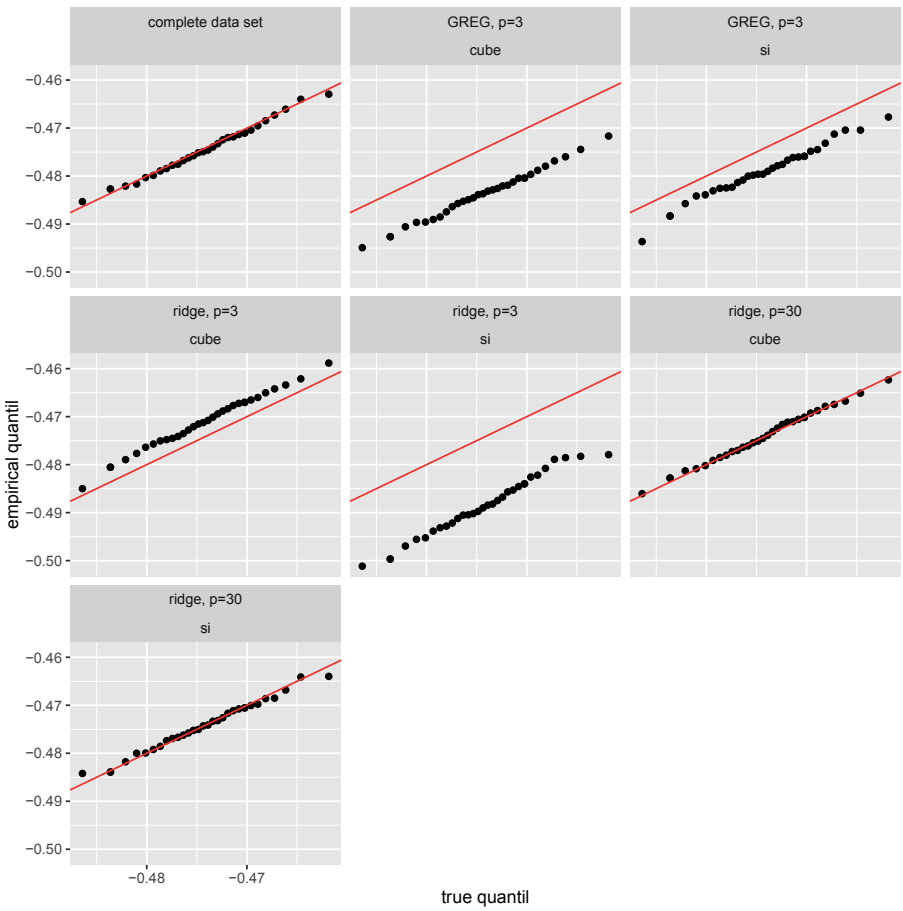

Figure 5: Exemplary quantile-quantile plots for a single parameter.

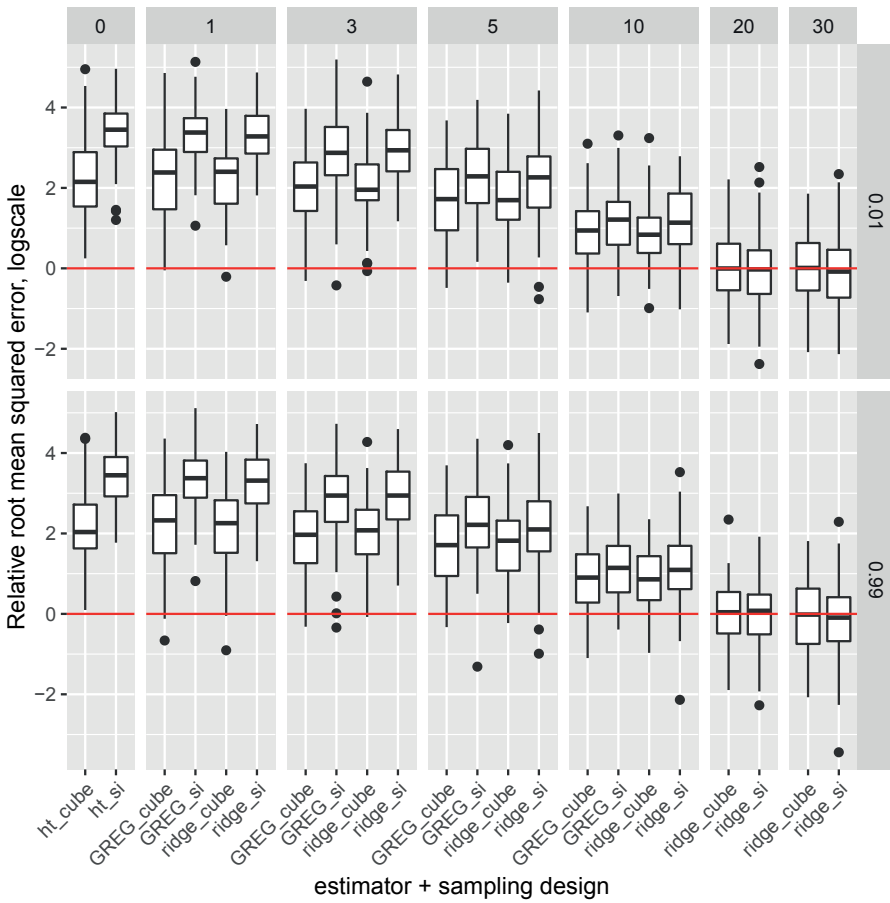

Figure 6: Gaussian likelihood with known variance: Root mean squared error for quantiles of $\boldsymbol{\theta}$, relative to root mean squared error obtained with complete data set, for quantile $q = 0.01, 0.99$.

5.2 Results

The main criterion is the root mean squared error (rmse) for the true parameter vector $\boldsymbol{\theta}$:

$$rmse(\hat{\boldsymbol{\theta}}) = \sqrt{\frac{1}{6}\sum_{i=1}^{6}(\hat{\theta}_i - \theta_i)^2},$$

where $\hat{\theta}_i$ is estimated using the posterior mean. We report the relative root mean squared error, in reference to the use of the complete data set: $rmse(\hat{\boldsymbol{\theta}})/rmse(\hat{\boldsymbol{\theta}}^{\mathcal{U}})$, where $rmse(\hat{\boldsymbol{\theta}}^{\mathcal{U}})$ is the rmse obtained by the use of the complete data set.

Speedup: The speedup, defined as computation time using the given method, divided by the computation time using the complete data set, is shown in figure 3. The basic greg variant is by far fastest, while the speedup for the ridge variant drops for larger number of auxiliary variables.

Parameter vector: Figure 4 shows the relative root mean squared error for the true parameter vector $\boldsymbol{\theta}$. It can be seen that the error mainly depends on p_{greg}: If there are enough auxiliary variables for the estimation (in this case, around 20), the relative rmse can approach 1. This holds for both likelihoods. Cube sampling reduces the error, although the effect is larger for the gaussian likelihood. We interpret this as a sign that the auxiliary variables are better chosen for the gaussian likelihood. For a large number of auxiliary variables used for estimation, this difference becomes small. Given a fixed value of p_{greg}, the difference in error between the basic greg estimator and the ridge estimator are small. However, the ridge variant allows the use of a larger number of auxiliary variables and is therefore to be preferred.

Quantiles: To study the accuracy of the simulated posterior distribution, we set σ equal to one and assume it to be known, so that the posterior distribution under a non-informative prior $\beta \propto constant$ is given by a multivariate normal distribution with variance $\boldsymbol{\Sigma} = (\boldsymbol{X}^\top \boldsymbol{X})^{-1}$ and mean $\boldsymbol{\Sigma}\boldsymbol{X}^\top \boldsymbol{y}$. As such, we can evaluate how well the true posterior distribution is simulated as whole, beyond point estimation. Exemplary quantile-quantile plots for a single parameter are shown in figure 5. The results are similar to the ones regarding $\boldsymbol{\theta}$: More auxiliary variables lead to a better approximation of the results of the complete data set, cube sampling improves over simple random sampling. Figure

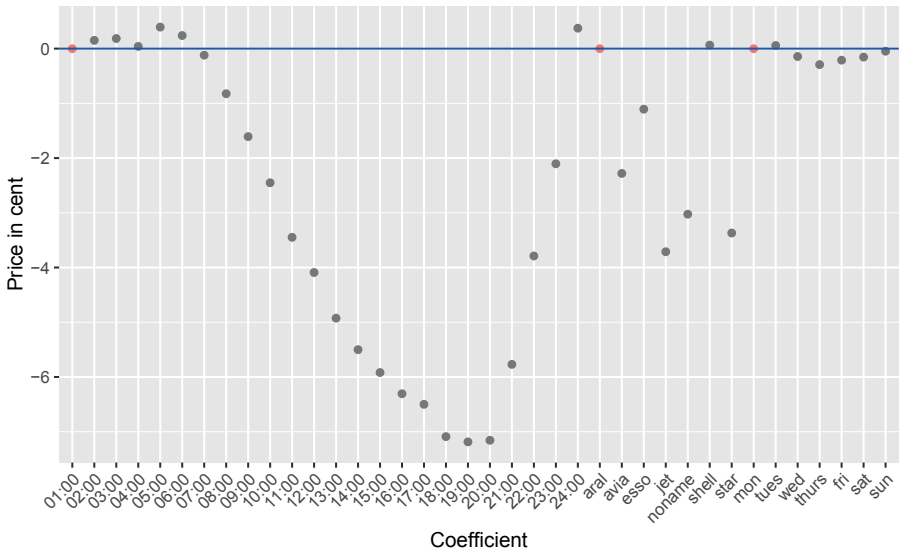

Figure 7: Coefficient plot (posterior means). Reference catagories are added as red points. Posterior intervals for 0.01-quantile and 0.99-quantile are barely visible and henceforth omitted.

6 shows the relative root mean squared error for the marginal quantiles of $\theta_1, \ldots, \theta_5$. The root mean squared error of the quantile vector is computed as:

$$rmse(Q_q(\boldsymbol{\theta})) = \sqrt{\frac{1}{5}\sum_{i=1}^{5}(Q_q(\theta_i) - \hat{Q}_q(\theta_i))^2},$$

for $q = 0.01, 0.99$, where $Q_q(x)$ is the q-Quantile of x. Here, the results are similar to the ones regarding $\boldsymbol{\theta}$ as well, however the relative root moon squared error does not approach 1 as fast as for point estimation.

6 Application

The methods presented here will be applied on a large data set consisting of the most recent gas price for every hour for every gas station in Germany for the first quarter of 2015. The data set consists of 31 million rows. We estimate the regression model:

$$\begin{aligned} dieselprice_{i,t} =& \beta_1 + f_{periodic}(weekday) + f_{periodic}(hour) + \\ & \sum_{j=1}^{5} \beta_j I[brand_i = j] + \epsilon_{i,t}, \epsilon_{i,t} \sim N(0, \sigma^2), \end{aligned}$$

where the periodic effects are estimated using dummies, setting one category equal to zero, so that $\boldsymbol{\theta} = (\beta_1, \boldsymbol{\beta}_{day}^\top, \boldsymbol{\beta}_{hour}^\top, \boldsymbol{\beta}_{brand}^\top, \sigma)^\top$. We use the ridge variant with a sample size of $n = 10000$ and $p_{greg} = 50$ auxiliary variables. For the cube sampling, 3 partial least squares latent variables are used. Partial least-squares linear combinations are fast to compute, and summarize the design matrix while preserving the correlation with y. The algorithm takes about 11 minutes on a Intel xeon cpu e5 with 2.4GHZ. Figure 7 reports the estimated posterior means of the regression coefficents. Most variation is explained by the hourly effects, diesel price is most expensive in the morning, least expensive in the evening around 19:00 with a difference of around 7 cents. The day of the week effect is very small, compared to the hour effect: Gas prices are highest on mondays, with an estimated price difference of less than 0.5 cent. The most expensive brand is Shell, followed by Aral. The smaller brands (Star and Jet) are more than 3 cents cheaper than Aral. These results are country-wide and do not take local effects into account, mainly the effect of the location of the gas station. Furthermore, there might be an interaction between the day of the week and the hour of the day. These topics should be the subject of a separate study.

7 Discussion

We developed an approximate version of the Metropolis-Hastings using a single subsample. Our simulation study shows that the algorithm can produce posterior simulations which are almost identical to the use of the full data while being several orders of magnitudes faster. The methods were validated using a simulation comparing the inferences obtained from the approximate posterior to the complete data posterior. The results suggest that the error obtained from using the algorithm is small enough

for practical purposes for standard models. However, further work regarding more complex models is necessary.

We propose to use this approximate version for the simulation of a posterior distribution which is used directly for inference. However, there are further possible applications for the algorithm; the algorithm might be used for the case when a fast approximation to the posterior distribution is of use, e.g., during the burnin of an adaptive Metropolis-Hastings algorithm.

References

Alquier, P., Friel, N., Everitt, R. and Boland, A. (2014). Noisy Monte Carlo: Convergence of Markov chains with approximate transition kernels, *Statistics and Computing* pp. 1–19.

Andrieu, C. and Roberts, G. O. (2009). The pseudo-marginal approach for efficient Monte Carlo computations, *The Annals of Statistics* **37**(2): 697–725.
URL: *http://projecteuclid.org/euclid.aos/1236693147*

Bardenet, R., Doucet, A. and Holmes, C. (2014). Towards scaling up Markov chain Monte Carlo: an adaptive subsampling approach, *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, pp. 405–413.

Chauvet, G. and Tillê, Y. (2006). A fast algorithm for balanced sampling, *Computational Statistics* **21**(1): 53–62.
URL: *http://link.springer.com/10.1007/s00180-006-0250-2*

Deville, J.-C. and Till, Y. (2004). Efficient balanced sampling: the cube method, *Biometrika* **91**(4): 893–912.

Green, P. J., atuszyski, K., Pereyra, M. and Robert, C. P. (2015). Bayesian computation: a summary of the current state, and samples backwards and forwards, *Statistics and Computing* **25**(4): 835–862.
URL: *http://link.springer.com/10.1007/s11222-015-9574-5*

Hoerl, A. E., Kannard, R. W. and Baldwin, K. F. (1975). Ridge regression:some simulations, *Communications in Statistics* **4**(2): 105–123.
URL: *http://www.tandfonline.com/doi/abs/10.1080/03610927508827232*

Korattikara, A., Chen, Y. and Welling, M. (2014). Austerity in mcmc land: Cutting the metropolis-hastings budget, *Proceedings of The 31st International Conference on Machine Learning*, pp. 181–189.

Maclaurin, D. and Adams, R. (2014). Firefly monte carlo: Exact mcmc with subsets of data, *Proceedings of the Thirtieth Conference Annual Conference on Uncertainty in Artificial Intelligence (UAI-14)*, AUAI Press, Corvallis, Oregon, pp. 543–552.

Maire, F., Friel, N. and Alquier, P. (2015). Light and Widely Applicable MCMC: Approximate Bayesian Inference for Large Datasets, *arXiv preprint arXiv:1503.04178* .
URL: *http://arxiv.org/abs/1503.04178*

Nicholls, G. K., Fox, C. and Watt, A. M. (2012). Coupled MCMC with a randomized acceptance probability, *arXiv preprint arXiv:1205.6857* .
URL: *http://arxiv.org/abs/1205.6857*

Quiroz, M., Villani, M. and Kohn, R. (2014). Speeding up MCMC by efficient data subsampling, *arXiv preprint arXiv:1404.4178* .
URL: *http://arxiv.org/abs/1404.4178*

Särndal, C.-E., Swensson, B., Wretman, J. H. and Srndal-Swensson-Wretman (2003). *Model assisted survey sampling*, Springer series in statistics, 1. softcover print edn, Springer, New York, NY.

Tillê, Y. (2006). *Sampling algorithms*, Springer series in statistics, Springer, New York.

A Appendix

A.1 Computation of Ridge estimator

For the model

$$\boldsymbol{\phi} = \boldsymbol{Z}_{greg}\boldsymbol{\beta} + \boldsymbol{e},$$

the ridge estimator for a given smoothing parameter κ is given by $\boldsymbol{\beta} = (\boldsymbol{Z}_{greg}^\top \boldsymbol{Z}_{greg} + \kappa)^{-1}\boldsymbol{Z}_{greg}\boldsymbol{\phi}$, where the rows of $\boldsymbol{Z}_{greg}$ are given by $\boldsymbol{z}_{greg,k}$, for $k \in \mathcal{S}$. Define $\boldsymbol{M} \coloneqq \boldsymbol{Z}_{greg}\boldsymbol{G}$, $\boldsymbol{\xi} \coloneqq \boldsymbol{G}^\top\boldsymbol{\beta}$, then

$$\boldsymbol{\phi} = \boldsymbol{Z}_{greg}\boldsymbol{G}\boldsymbol{G}^\top\boldsymbol{\beta} + \boldsymbol{e} = \boldsymbol{M}\boldsymbol{\xi} + \boldsymbol{e};$$

where $\boldsymbol{Z}_{greg}^\top \boldsymbol{Z}_{greg} = \boldsymbol{G}\boldsymbol{\Lambda}\boldsymbol{G}^\top$ is the eigen-decomposition of $\boldsymbol{Z}_{greg}^\top \boldsymbol{Z}_{greg}$ with orthonormal $\boldsymbol{G}$, so that $\Lambda = diag(\lambda_1, \ldots, \lambda_k)$,

$$\lambda_1 \geq \lambda_2, \geq \cdots \geq \lambda_p$$

where $\lambda_1, \ldots, \lambda_p$ are the eigenvalues of $\boldsymbol{Z}_{greg}^\top \boldsymbol{Z}_{greg}$. It holds that $\boldsymbol{Z}_{greg}^\top \boldsymbol{Z}_{greg} + c\boldsymbol{I} = \boldsymbol{G}(\boldsymbol{\Lambda} + c\boldsymbol{I})\boldsymbol{G}^\top$, so that

$$(\boldsymbol{Z}_{greg}^\top \boldsymbol{Z}_{greg} + \kappa \boldsymbol{I})^{-1} = \boldsymbol{G}(\Lambda + \kappa \boldsymbol{I})^{-1}\boldsymbol{G}^\top,$$

and

$$\boldsymbol{\beta}(\kappa) = \boldsymbol{G}(\Lambda + \kappa \boldsymbol{I})^{-1}\boldsymbol{M}^\top \boldsymbol{\phi} \rightarrow$$

$$\boldsymbol{\xi}(\kappa) = (\Lambda + \kappa \boldsymbol{I})^{-1}\boldsymbol{M}^\top \boldsymbol{\phi}.$$

Define $\boldsymbol{c} \coloneqq \boldsymbol{M}^\top \boldsymbol{\phi}$. Than, the elements of $\hat{\boldsymbol{\xi}}(\kappa)$ are

$$\hat{\xi}_i = \frac{c_i}{\lambda_i + \kappa}, i = 1, \ldots, p_{greg}.$$

Following Hoerl et al. (1975), the ridge parameter is set as:

$$\hat{\kappa} = \frac{p_{greg}\hat{\sigma}^2}{\hat{\boldsymbol{\xi}}(0)^\top \hat{\boldsymbol{\xi}}(0)}. \tag{9}$$

The residual variance is estimated using the usual unbiased estimator

$$\begin{aligned}\hat{\sigma}^2 &= (n - p_{greg})^{-1} \sum_{i \in \mathcal{S}} (\phi_i - \boldsymbol{z}_i^\top \hat{\boldsymbol{\beta}}(0))^2 \\ &= (n - p_{greg})^{-1} (\boldsymbol{\phi} - \boldsymbol{M}\hat{\boldsymbol{\xi}}(0))^\top (\boldsymbol{\phi} - \boldsymbol{M}\hat{\boldsymbol{\xi}}(0)).\end{aligned}$$

The vector $\hat{\boldsymbol{\beta}}(\hat{\kappa})$ does not have be computed, using representation (7) of the greg estimator: Define $\tilde{\boldsymbol{z}}_{greg} \coloneqq \boldsymbol{z}_{greg}^\top \boldsymbol{G}$ and $\tilde{\boldsymbol{z}}_{greg,\mathcal{S}} \coloneqq (\sum_{\mathcal{S}} \boldsymbol{z}_i)^\top G$, then $\dot{\phi}$ is estimated via:

$$\boldsymbol{\xi}(\hat{\kappa})^\top \tilde{\boldsymbol{z}}_{greg} + (N/n) \left((\sum_{\mathcal{S}} \phi_k) - \boldsymbol{\xi}(\hat{\kappa})^\top \tilde{\boldsymbol{z}}_{greg,\mathcal{S}} \right).$$

The vectors $\tilde{\boldsymbol{z}}_{greg}$ and $\tilde{\boldsymbol{z}}_{greg,\mathcal{S}}$ only have to be computed once.