RESEARCH ARTICLE

# Turning Simulation into Estimation: Generalized Exchange Algorithms for Exponential Family Models

**Maarten Marsman[1]*, Gunter Maris[1,2], Timo Bechger[2], Cees Glas[3]**

**1** Department of Psychology, University of Amsterdam, Amsterdam, the Netherlands, **2** Psychometric Research Center, Cito, Arnhem, the Netherlands, **3** Department of Research Methodology, Measurement and Data Analysis, University of Twente, Enschede, the Netherlands

* m.marsman@uva.nl

OPEN ACCESS

**Citation:** Marsman M, Maris G, Bechger T, Glas C (2017) Turning Simulation into Estimation: Generalized Exchange Algorithms for Exponential Family Models. PLoS ONE 12(1): e0169787. doi:10.1371/journal.pone.0169787

**Editor:** Kewei Chen, Banner Alzheimer's Institute, UNITED STATES

**Received:** August 18, 2016

**Accepted:** December 21, 2016

**Published:** January 11, 2017

**Data Availability Statement:** No "real" data were used in this manuscript. We now include R-files as supporting information.

**Funding:** The authors GM and TB are paid a salary by CITO, a nonprofit commercial educational testing company, and they worked on this paper during their salaried time. CITO provided support in the form of salaries for authors GM and TB, but did not have any additional role in the study design, data collection and analysis, decision to publish, or preparation of the manuscript. The specific roles of

## Abstract

The Single Variable Exchange algorithm is based on a simple idea; any model that can be simulated can be estimated by producing draws from the posterior distribution. We build on this simple idea by framing the Exchange algorithm as a mixture of Metropolis transition kernels and propose strategies that automatically select the more efficient transition kernels. In this manner we achieve significant improvements in convergence rate and autocorrelation of the Markov chain without relying on more than being able to simulate from the model. Our focus will be on statistical models in the Exponential Family and use two simple models from educational measurement to illustrate the contribution.

## 1 Introduction

In Bayesian statistical modeling, researchers formalize their substantive theories in a statistical model $\pi(x \mid \theta)$ for the distribution of the data $x$ and a prior distribution $\pi(\theta)$ for the parameter $\theta$. Together, the statistical model and the prior distribution lead to the posterior distribution $\pi(\theta \mid x)$. The desired posterior distribution is often intractable, and simulating draws from such posterior distributions can be a difficult task. However, simulating data from the model is often simple: first generate a parameter value $\theta^*$ from the prior distribution $\pi(\theta)$ and then generate data $x^*$ from the statistical model $\pi(x \mid \theta^*)$. From this composite sampling scheme we obtain draws from the joint distribution [1]:

$$\pi(x, \theta) = \pi(\theta)\pi(x \mid \theta) = \pi(x)\pi(\theta \mid x), \tag{1}$$

showing that the generated value $\theta^*$ is also a draw from the posterior distribution $\pi(\theta^* \mid x^*) \propto \pi(x^* \mid \theta^*)\pi(\theta^*)$ [2]. The posterior distribution $\pi(\theta^* \mid x^*)$ is equal to the desired posterior distribution $\pi(\theta \mid x)$ only if the generated data $x^*$ are equal to the observed data $x$. Here we improve on an algorithm that uses the composite data sampling scheme to obtain draws from the posterior distribution $\pi(\theta \mid x)$ to be used in a wide range of settings.

To simulate draws from the desired posterior $\pi(\theta \mid x)$, Murray, Ghahramani and MacKay [3] cleverly utilized the posteriors $\pi(\theta \mid x^*)$ as proposal distributions in an independence chain

these authors are articulated in the 'author contributions' section.

**Competing Interests:** The authors MM and CG have declared that no competing interests exist. The employment for authors GM and TB by CITO does not alter our adherence to PLOS ONE policies on sharing data and materials.

Metropolis algorithm [4]. This Metropolis algorithm operates by constructing a discrete-time Markov chain with state space $\Omega_\theta$ that has the desired posterior $\pi(\theta \mid x)$ as invariant distribution [5], and can be characterized as follows:

$$\Theta_{t+1} = \begin{cases} \Theta^* = \theta^* & \text{with probability } \phi(\theta',\, \theta^* \mid x,\, x^*) \\ \Theta_t = \theta' & \text{with probability } 1 - \phi(\theta',\, \theta^* \mid x,\, x^*) \end{cases} \sim \Theta_t \sim \pi(\cdot \mid x),$$

where the state $\theta'$ of the chain at a time $t$ is a draw from the invariant distribution $\pi(\theta \mid x)$, the proposed value $\theta^*$ is a draw from the proposal distribution $\pi(\theta \mid x^*)$, with $\theta'$ and $\theta^* \in \Omega_\theta$. The proposed point $\theta^*$ is accepted if $U < \phi$, with $U \sim \text{Uniform}(0, 1)$ and:

$$\phi(\theta',\, \theta^* \mid x,\, x^*) = \min\left(1, \frac{\pi(\theta^* \mid x)\pi(\theta' \mid x^*)}{\pi(\theta' \mid x)\pi(\theta^* \mid x^*)}\right) = \min\left(1, \frac{\pi(x \mid \theta^*)\pi(x^* \mid \theta')}{\pi(x \mid \theta')\pi(x^* \mid \theta^*)}\right). \qquad (2)$$

Note that the proposed value $\theta^*$ is always accepted, i.e., $\phi(\theta', \theta^* \mid x, x^*) = 1$, if the proposed setting $\pi(x \mid \theta^*)\pi(x^* \mid \theta')$ is more likely than the current setting $\pi(x \mid \theta')\pi(x^* \mid \theta^*)$, otherwise it is accepted with a probability proportional to the ratio of likelihoods in the proposed and current setting. This is known as the Single-Variable-Exchange (SVE) algorithm [3] and makes simulating draws from a posterior distribution as simple as simulating data.

Although the SVE algorithm presents a practical and simple solution to sampling from intractable posterior distributions, its application and development has focused exclusively on statistical models $\pi(x \mid \theta)$ with computationally intractable normalizing constants [6]. In fact, the SVE algorithm was originally developed for statistical models in the Exponential Family [7–9]; i.e., models that can be written as:

$$\pi(x \mid \theta) = \frac{1}{Z_\theta} h(x) \exp\{\theta \cdot t(x)\},$$

where $t(x)$ is a (vector of) statistic(s) sufficient for $\theta$ and $Z_\theta$ a normalizing constant. Observe that for models in the Exponential Family, the acceptance probability is of a particular simple form:

$$\phi(\theta',\, \theta^* \mid t(x),\, t(x^*)) = \min(1,\; \exp\{\, (\theta^* - \theta') \cdot (t(x) - t(x^*))\,\}), \qquad (3)$$

and does not involve the normalizing constant $Z_\theta$; making the SVE algorithm a practical tool for Bayesian inference of models where $Z_\theta$ is intractable, such as Exponential Random Graphs [10, 11] and Markov Random Fields [12, 13]. Despite the simplicity with which the SVE algorithm operates, especially for models in the Exponential Family (e.g., generalized linear models), its application to tractable statistical models $\pi(x \mid \theta)$ has been completely unexplored.

Simple implementations do not guarantee efficient Markov chains, and in practice we often see that the SVE algorithm operates with low efficiency; requiring many thousands of iterations to obtain accurate estimates and wasting expensive computations on rejected proposals. This inefficiency results from generating data sets $x^*$ that are unlikely to occur under the current setting $\theta'$ (or $x$ under $\theta^*$); i.e. statistics $t(x^*)$ that are far from $t(x)$ in Eq (3). To this aim, several approaches have been proposed that replace the simple proposal generating mechanism with more elaborate schemes, using, for instance, random walks [3], parallel tempering [3], population Markov chain Monte Carlo methods [10, 14], Langevine diffusions [15], or delayed rejection [16]. Although these approaches improve the statistical efficiency, they often fail to generalize the simple implementation of the original SVE algorithm.

Consequently, our goals are two-fold. Our primary goal is to show several developments that improve the statistical efficiency of the original SVE algorithm and result from reformulating it as an instance of what Tierney refers to as a mixture of Metropolis kernels [4, 17] in Section 2.1. Our efforts focus on simultaneously sampling from the posterior distribution of a large number of latent variables (e.g., random effects in generalized linear mixed models or Bayesian hierarchical models) in Sections 3.2 and 3.3, and on sampling from highly concentrated posterior distributions in Sections 3.1, 3.4 and 3.5. The strategies that we present improve their efficiency as the sample size grows (driving the autocorrelation to zero), and allow the utilization of the cheap parallelism that is available in modern day computing [18]. A simple Exponential Family model serves to illustrate the development.

Our secondary goal is to introduce the SVE algorithm as a general purpose method that makes Bayesian inference simple, even for relatively complicated models. As the SVE algorithm does not require much more than the ability to generate data from the statistical model, we believe that it is a practical tool for applied researchers and also serves as a simple introduction to Markov chain Monte Carlo methods for students novel to the field. The extensions that we propose in this paper also fit these two purposes in that they are simple and intuitive extensions of the original SVE approach. To illustrate the original SVE approach and our proposed extensions, we have included annotated R [19] code as supporting information. Specifically, S1–S6 Scripts can be used to reproduce our results (i.e., Figs 1–6), and S7 Script contains the original SVE algorithm and our proposed algorithms in isolation.

Even though we will specifically focus on models in the Exponential Family, we note that our approach also applies to other models by replacing the sufficient statistic with an auxiliary statistic to relate generated data to a parameter. In general one often has a good idea how data and parameters are related, such that it is simple to find efficient auxiliary statistics, an idea that is regularly exploited in Approximate Bayesian Computation [20–22].

Clearly, the main drawback of our approach is the assumption that one is capable of simulating data from the model. That is, we assume that routines to sample (directly) from $\pi(x \mid \theta)$ and $\pi(\theta)$ are available. For most models efficient sampling routines are readily available in standard statistical software such as R [19], or can be constructed using general procedures [23, 24]. Extensions of the SVE algorithm where data are sampled using a Markov chain have also been considered [25, 26], and although not investigated here, we anticipate that our approach also extends in this direction. The more general notion is this: if the problem of simulating data is solved, the SVE algorithm turns data simulation into parameter estimation by producing draws from the posterior distribution.

# 2 Methods

## 2.1 A Mixture of Transition Kernels

The factorization in Eq (1) reveals that by first sampling a parameter value $\theta^* \sim \pi(\theta)$ and then data $x^* \sim \pi(x \mid \theta^*)$, with probability (density) $\pi(x^*)$ we have generated $\theta^*$ from the proposal $\pi(\theta \mid x^*)$. With the acceptance probabilities $\phi$ defined as in Eq (2), we have that each generated proposal $\pi(\theta \mid x^*)$ corresponds to a unique Metropolis transition kernel, i.e., a transition probability distribution with density:

$$\pi(\theta_{t+1} \mid \theta_t,\ x,\ x^*) = \begin{cases} \pi(\theta^* \mid x^*)\,\phi(\theta_t,\ \theta^* \mid x,\ x^*) & \text{if } \theta_t \neq \theta_{t+1} \\ 1 - \int_{\Omega_\theta} \pi(\theta^* \mid x^*)\,\phi(\theta_t,\ \theta^* \mid x,\ x^*)\,\mathrm{d}\theta^* & \text{if } \theta_t = \theta_{t+1} \end{cases}$$

for which the posterior $\pi(\theta \mid x)$ is the invariant distribution:

$$\pi(\theta \mid x) = \int_{\Omega_\theta} \pi(\theta \mid \theta',\ x^*,\ x)\,\pi(\theta' \mid x)\,\mathrm{d}\theta'.$$

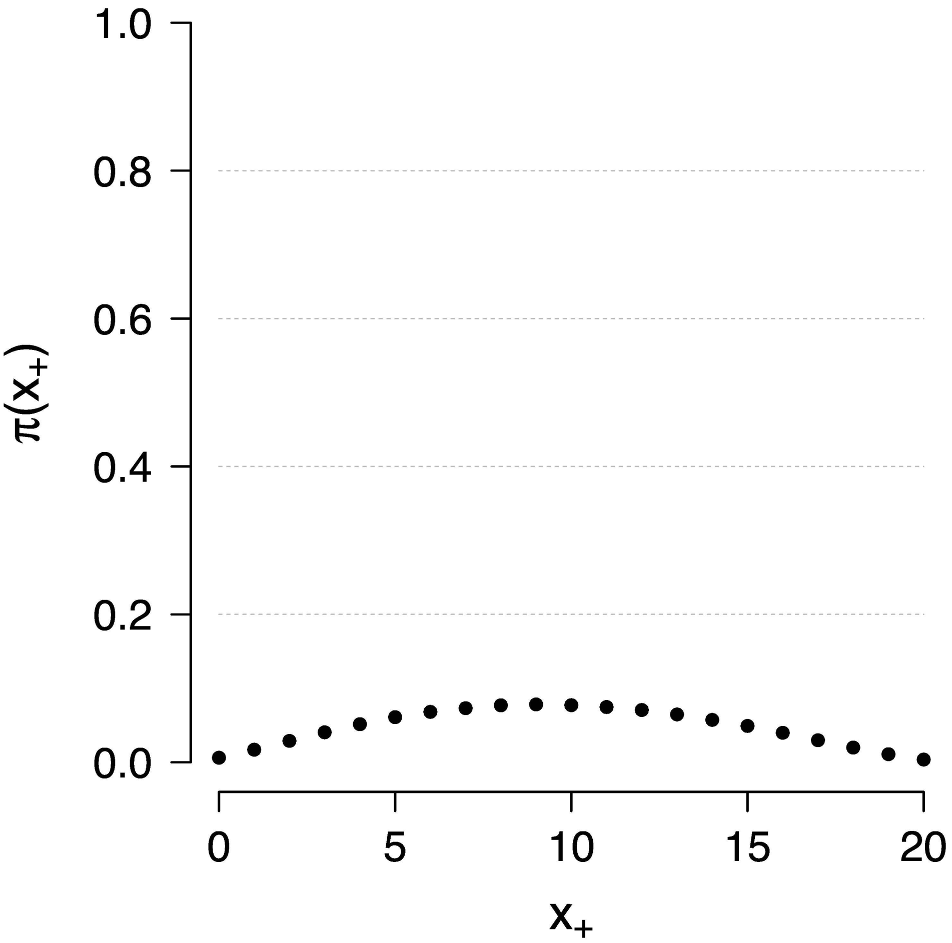

**Fig 1. A mixing distribution $\pi(x^*_+)$ for the original SVE.** The distribution of transition kernels, i.e., $\pi(x^*_+)$, for the original SVE algorithm in the Rasch model example with $k = 20$ items. In this example the average acceptance rate for sampling from the posterior $\pi(\theta \mid x_+ = 9)$ was approximately 37% (see S1 Script).

doi:10.1371/journal.pone.0169787.g001

Since each Metropolis kernel in the SVE algorithm has the same invariant distribution, so does their mixture [4, 17]:

$$\pi(\theta^* \mid x) = \int_{\Omega_{x^*}} \int_{\Omega_\theta} \pi(\theta^* \mid \theta,\ x^*,\ x)\, \pi(\theta \mid x)\, \pi(x^*)\, \mathrm{d}\theta\, \mathrm{d}x^*,$$

where the integration is over all possible generated datasets $\Omega_{x^*}$. This formulation shows that the inefficiency of the SVE algorithm is caused by the generation of kernels that have a low probability of making a move, i.e., kernels corresponding to statistics $t(x^*)$ that are far from the observed statistic $t(x)$ (c.f. Eq (3)).

To illustrate, consider a simple example from educational measurement; the Rasch model [27]. The Rasch model describes the distribution of a pupil's item responses as a function of an ability parameter $\theta$, and a difficulty parameter $\delta$ for each of $k$ items:

$$\pi(x \mid \theta) = \frac{\exp\left\{\sum_i x_i \theta - \sum_i x_i \delta_i\right\}}{\prod_i (1 + \exp\{\theta - \delta_i\})},$$

where $X = 1$ denotes a correct response and $X = 0$ an incorrect response. The test score $x_+ =$

$\sum_i x_i$ is sufficient for $\theta$ such that its posterior depends on the data only through the test score [28], and the mixture ranges over the $k + 1$ possible test scores $\Omega_{x_+^*} = \{0, 1, \ldots, k\}$:

$$\pi(\theta^* \mid x_+) = \sum_{\Omega_{x_+^*}} \int_{\Omega_\theta} \pi(\theta^* \mid \theta,\ x_+^*,\ x_+)\, \pi(\theta \mid x_+)\, \mathrm{d}\theta\, \pi(x_+^*).$$

The mixing distribution (i.e., test score distribution) $\pi(x_+)$ is shown in Fig 1 for a test consisting of $k = 20$ items, and confirms that it gives much weight to kernels corresponding to values of $t(x^*) = x_+^*$ that are far from the observed value $t(x) = x_+$. For instance, for a pupil with test score $t(x) = x_+ = 9$, say, the probability to generate a kernel corresponding to values $t(x^*) = x_+^*$, with $|x_+^* - x_+| > 4$, is approximately 35% in this example (see S1 Script).

## 3 Results

### 3.1 Oversampling for Single Parameter Updates

We wish to assign more weight to kernels with a high probability of accepting a move, while preserving the correct invariant distribution $\pi(\theta \mid x)$. A simple way to achieve this is to generate $m \geq 1$ proposed points and then select the one that yielded a sufficient statistic $t(x^*)$ most similar to $t(x)$ (c.f. Eq (3)), where $m = 1$ results in the original SVE algorithm. Just as in the original SVE algorithm we have that the posterior distribution $\pi(\theta \mid x^*)$ is the invariant distribution. Fig 2 illustrates the improvement of this procedure in our Rasch model example for a test score $x_+ = 9$; improving the probability of directly generating from the target (where $x_+^* = x_+$) from 0.1 to about 0.4 with $m = 5$ samples and about 0.8 with $m = 20$ samples. Even when no

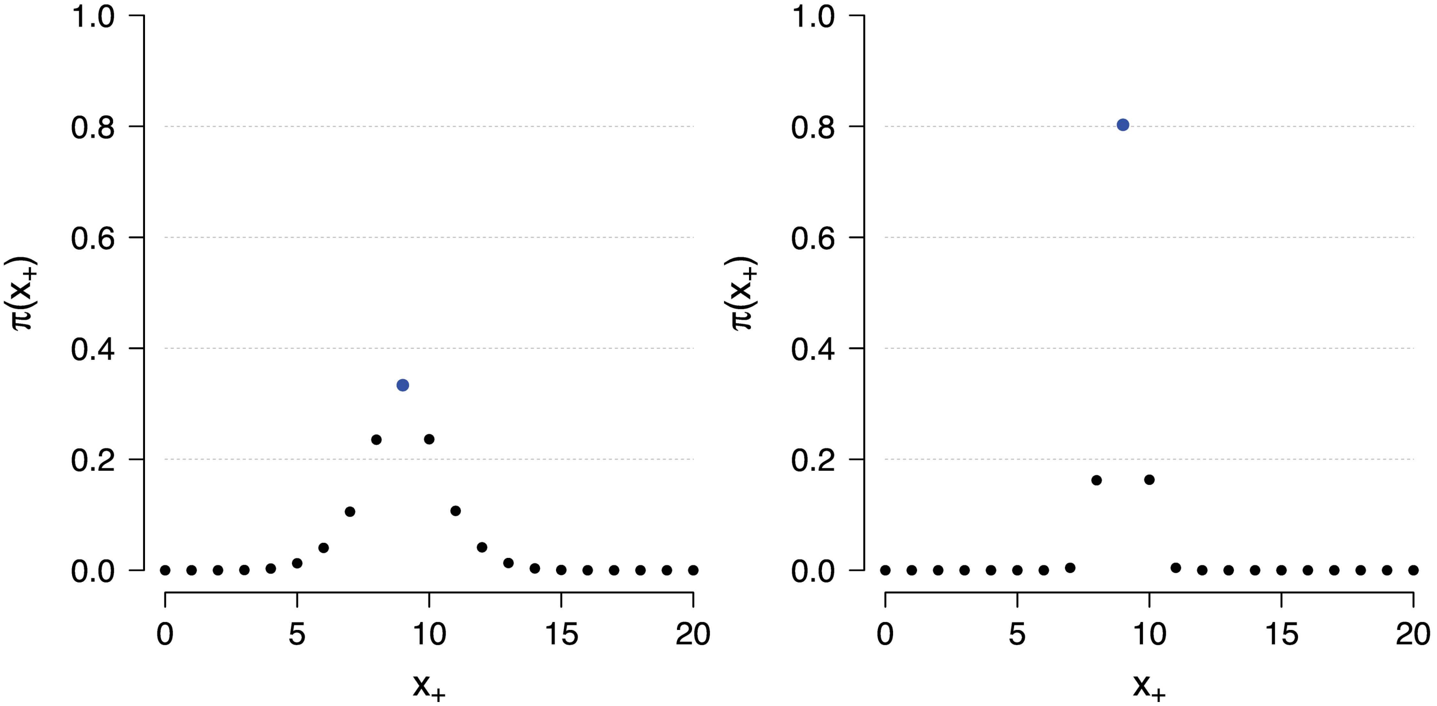

**Fig 2. A mixing distribution $\pi(x_+^*)$ for SVE with oversampling.** The distribution of transition kernels, i.e., $\pi(x_+^*)$, for sampling from the posterior $\pi(\theta \mid x_+ = 9)$ when choosing the best one out of $m = 5$ generated proposals (left panel) and $m = 20$ generated proposals (right panel). In this example the acceptance rate was equal to 75% when generating $m = 5$ proposals and equal to 95% when generating $m = 20$ proposals.

doi:10.1371/journal.pone.0169787.g002

direct sample was produced, the proposal distributions became increasingly more similar to the target distribution, thus increasing the overall probability of making a move.

In the application above, we have used functions $t()$ of the observed $x$ and simulated data $x^*$ to select a proposal distribution (i.e., a transition kernel). In practice, however, we might have more information available that informs the selection of good proposal distributions, and one can use functions $f()$ that incorporate this information. In the next section we illustrate such a function that incorporates, for instance, covariates that are used in the statistical model. Observe that we do not use the current state of the parameter, $\theta'$, or the proposed point, $\theta^*$, to select a proposal distribution. This ensures that the posterior distribution $\pi(\theta \mid x)$ remains the correct invariant distribution of the Markov chain.

Fig 2 reveals that using the sufficient statistic we are able to select statistically more efficient proposals as $m$ increases. This follows from inspecting the acceptance probability in Eq (3), and observing that the statistically more efficient proposals are those for which $|t(x^*) - t(x)|$ is at a minimum, and furthermore that the minimum $\min_m \{|t(x^*) - t(x)|\}$ over $m$ proposals is non-increasing with $m$, i.e.,

$$\min_{m} \{|t(x^*) - t(x)|\} \geq \min_{m+1} \{|t(x^*) - t(x)|\}.$$

It is important to note that the $m$ proposals can be generated in parallel so that the oversampling of proposals need not increase the computational burden. However, only one of the $m$ proposals is subsequently accepted by the Markov chain. As we will see next, all generated proposals can be put to good use when simultaneously sampling from more than one target distribution.

## 3.2 Matching for Multiple Parameter Updates

With the commonly assumed conditional independence of observations in hierarchical models, we have independent posterior distributions for each of $n$ random effects (or latent variables) [28]:

$$\pi(\boldsymbol{\theta}^* \mid \mathbf{x}) = \prod_{p=1}^{n} \pi(\theta_p \mid x_p).$$

Since proposals are also independently generated, it is convenient to consider the joint application of $n$ independent SVE kernels:

$$\pi(\boldsymbol{\theta}^* \mid \mathbf{x}) = \int_{\Omega_{\mathbf{x}}} \prod_{p=1}^{n} \int_{\Omega_\theta} \pi(\theta_p^* \mid \theta,\ x_p^*,\ x_p)\, \pi(\theta \mid x_p)\, \mathrm{d}\theta\, \pi(\mathbf{x}^*)\, \mathrm{d}\mathbf{x}^*,$$

where the original SVE algorithm has

$$\pi(x_p^* \mid \mathbf{x}_{\backslash p}^*) = \pi(x_p^* \mid x_1^*, \cdots, x_{p-1}^*, x_{p+1}^*, \cdots, x_n^*) = \pi(x_p^*),$$

due to independence. We wish to modify $\pi(\mathbf{x}^*)$ such that each component $\pi(x_p^*) = \int \pi(\mathbf{x}^*)\ \mathrm{d}\mathbf{x}_{\backslash p}^*$ assigns more weight to kernels with a high probability of accepting a move.

Similar to our oversampling procedure we can generate $m \geq 1$ proposals and assign each of the generated proposals to a target distribution. Here, we choose the number of generated proposals to be equal to the number of target distributions, which implies that we simply rearrange the $m = n$ generated proposals. We wish to rearrange the proposals such that each of the $n$ kernels has a high probability of accepting the proposed point; i.e., match proposals to targets such that for each target distribution the proposal statistic $t(x^*)$ is close to the observed statistic.

However, even for relatively small values of $n$ it becomes expensive to search the $n!$ possible arrangements for the statistically most efficient one, which suggests to use a simple rule to automatically choose an arrangement given a generated dataset $\mathbf{x}^*$.

We illustrate such a simple rule with sampling from the posterior distribution of $n$ ability parameters in the Rasch model;

$$\pi(\boldsymbol{\theta} \mid \mathbf{x}_+) = \prod_{p=1}^{n} \pi(\theta_p \mid x_{p+}).$$

We order the posterior distributions based on the corresponding test scores; those corresponding to a low test score are placed first whereas those corresponding to a high test score are placed last. Next, we generate $m = n$ proposal distributions which are ordered in the same way as the target distributions; those corresponding to a low generated test score are placed first and those corresponding to a high generated test score are placed last. It is clear that the first proposal is likely to be a good proposal for the first target distribution, the second proposal for the second target distribution, and so on. That this procedure improves the statistical efficiency is shown in Fig 3. The solid horizontal line in Fig 3 shows the average acceptance rate of the original SVE algorithm applied separately to each of the $n$ latent variables, the efficiency of

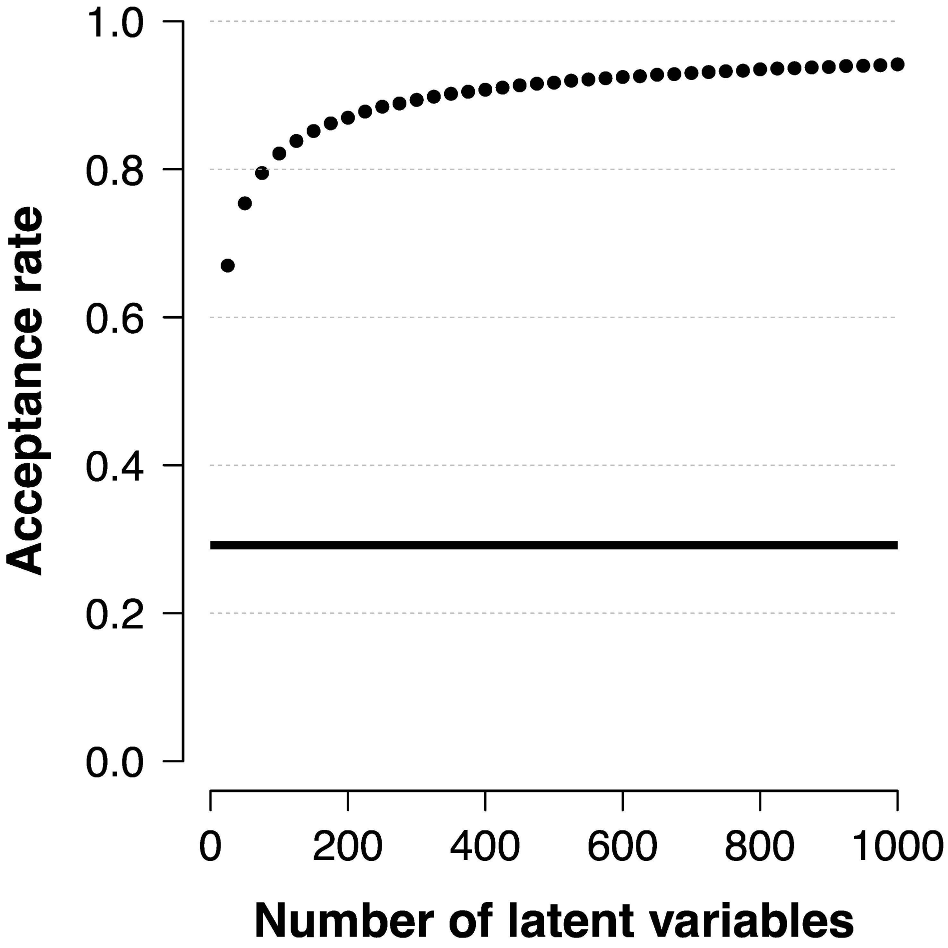

**Fig 3. Acceptance rates of the original SVE and SVE with matching.** The average proportion of accepted points when simultaneously sampling from $n$ target distributions $\pi(\theta \mid x_+)$ in the original SVE algorithm (solid line) and the proposal matching procedure (points).

doi:10.1371/journal.pone.0169787.g003

which is independent of $n$, and the points refer to the acceptance rate using our kernel matching procedure. Even with as little as $n = 25$ latent variables there is a substantial improvement to the statistical efficiency when the proposals are matched, with an average acceptance rate of 29% in the original SVE algorithm and 67% when the proposals are matched. Moreover, Fig 3 reveals that the statistical efficiency continues to improve as $n$ increases.

Fig 3 reveals that if $t(x)$ is sufficient for $\theta$, we have a good way to match proposals to targets and as $n$ becomes sufficiently large, each kernel tends to make a move such that we sample approximately i.i.d. from each of the $n$ posteriors. We note that the proposals can be generated and subsequently accepted in parallel. The only non-parallizable part of the procedure is in matching the proposals, although one can find clever ways to do this. Sorting the statistics $t(x^*)$ (posterior distributions) is of an order of complexity that is often usually $n \log n$ but at most $n^2$, which compares favorably to the order of complexity $n!$ that is needed to find the statistically most efficient rearrangement.

Sampling-unit specific prior distributions $\pi_p(\theta)$ are easily handled by incorporating the information encoded in the prior distributions, such as covariates, in matching the proposals. Since this information is also encoded in the mixing probabilities, it is available for matching proposals to target distributions. The only difference is that when a point drawn from $\pi_q(\theta)$ is proposed to a posterior with prior density $\pi_p(\theta)$, $p \neq q$, the prior distributions do not cancel from the expression in Eq (3), and we accept $\theta^*$ with probability:

$$\min\left(1,\ \exp\{(\theta^* - \theta')\cdot(t(x) - t(x^*))\} \times \frac{\pi_p(\theta^*)\,\pi_q(\theta')}{\pi_p(\theta')\,\pi_q(\theta^*)}\right),$$

ensuring that $\pi(\theta \mid x) \propto \pi(x \mid \theta)\pi_p(\theta)$ remains the invariant. For most prior distributions, the ratio of prior distributions is easily computed as many parts cancel in the expression.

To illustrate, consider a latent regression model in which each of $n$ abilities $\theta$ is assigned a unique prior distribution:

$$\theta_p \mid y_p \sim \pi_p(\theta) = \mathcal{N}\left(y_p^{\mathsf{T}}\beta,\ \sigma^2\right),$$

where $y_p$ constitutes a covariate vector for person $p$ and $\beta$ a vector of regression weights. Assuming that a point drawn from $\pi_q(\theta)$ is proposed to a posterior with prior density $\pi_p(\theta)$, $p \neq q$, we consequently accept $\theta^*$ with probability:

$$\min\left(1,\ \exp\{(\theta^* - \theta')\cdot([x_{p+} + y_p^{\mathsf{T}}\beta/\sigma^2] - [x^*_{q+} + y_q^{\mathsf{T}}\beta/\sigma^2])\}\right),$$

which also reveals that it is opportune to use the statistic $x_{p+} + y_p^{\mathsf{T}}\beta/\sigma^2$ to match proposals to targets.

## 3.3 A Rejection Algorithm

When $t(X^*)$ is a discrete random variable, a simple Rejection procedure is to generate proposals until an exact match $t(x^*) = t(x)$ is produced [2]. The matching of kernels points to an extension of the Rejection algorithm for sampling from $n \geq 1$ posteriors. Consider sampling from the posterior distribution of $n$ ability parameters in the Rasch model using a common prior. There are at most $k + 1$ different posterior distributions to sample from; one for every possible test score. Let $n_{x_+}$ denote the number of observations for a test score $x_+$. We generate a proposal corresponding to a test score $x^*_+$; if $n_{x^*_+} > 0$ we retain the proposed point and decrease $n_{x^*_+}$ by one, otherwise we reject the proposed point. This procedure is repeated until

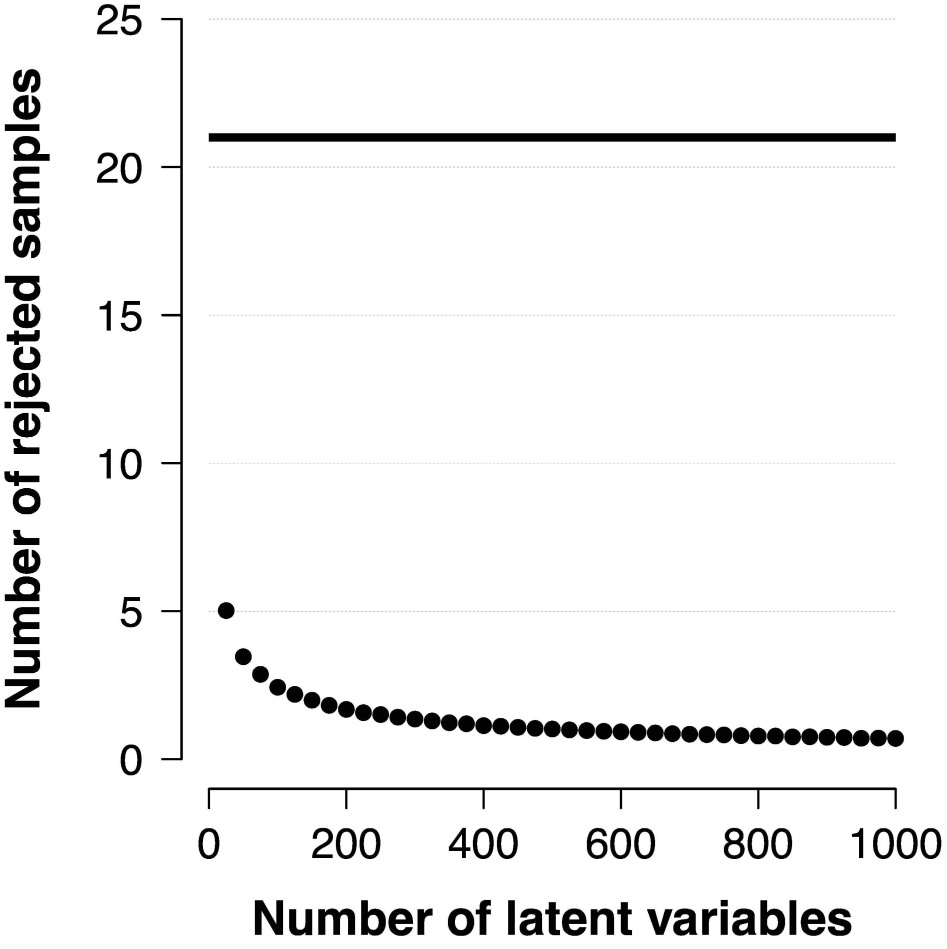

**Fig 4. Number of rejected samples for the rejection algorithms.** The average number of rejected samples per posterior $\pi(\theta \mid x_+)$ out of $n$ target distributions in the original Rejection algorithm (solid line) and when rejected values are recycled among the target distributions (points).

doi:10.1371/journal.pone.0169787.g004

$n_{x_+} = 0$ for each possible test score, after which the generated values can be assigned to the target distributions. Note that this simplifies to the original rejection algorithm when $n = \sum_{x_+} n_{x_+} = 1$.

Fig 4 shows that recycling the otherwise wasted proposals can significantly improve the computational efficiency (reduce the order of complexity). The solid horizontal line shows the average number of proposals that need to be generated when the original rejection algorithm is applied separately to each of $n$ latent variables, the efficiency of which is independent of $n$, and the points show the average number of proposals that need to be generated when the proposals are recycled. Most significant is the reduction of the computational expense when samples are required from increasingly larger numbers of target distributions. When $n$ becomes sufficiently large, only $n$ proposals need to be generated to sample once from each of the $n$ target distributions.

## 3.4 Binning the Statistics

The rejection algorithm is unsuited for applications in which $t(X)$ is a continuous random variable or a discrete random variable with many possible realizations. Even though repeated sampling does not guarantee an exact match, oversampling revealed that we do continue to produce better proposals. In general, a good proposal is one for which the statistic $t(x^*)$ is "sufficiently close" to $t(x)$; i.e., $t(x^*)$ is in some range $\mathcal{T}_a = (t(x) - a,\ t(x) + a)$, with $a > 0$. This suggests that we generate proposals until $t(x^*) \in \mathcal{T}_a$, with the value of $a$ controlling the quality of our proposals.

To illustrate, consider a simple extension of our Rasch model known as the two-parameter logistic model. The two-parameter logistic model describes the distribution of a pupil's item responses as a function of the ability parameter $\theta$ and an item discrimination $\alpha_i$ and difficulty $\delta_i$ for each of $k$ items:

$$\pi(x \mid \theta) = \frac{\exp\{\sum_i \alpha_i x_i \theta - \sum_i x_i \delta_i\}}{\prod_i (1 + \exp\{\alpha_i \theta - \delta_i\})},$$

where the weighted test score $t(x) = \sum_i \alpha_i\, x_i$ is sufficient for $\theta$. Since the discrimination parameters $\alpha_i$ are real-valued (typically positive) we have that $t(X)$ is a discrete random variable with $2^k$ possible realizations, one for every possible vector of item scores.

We consider sampling from a posterior $\pi(\theta \mid t(x))$ for a weighted test score $t(x) \approx 19$ ($x_+ = 9$) based on a $k = 20$ item test with discriminations $\alpha_i$ that are sampled uniformly between 0 and 4. Fig 5 reveals that generating proposals until $t(x^*)$ falls in a bin $\mathcal{T}_a$ increases the quality of proposals as a function of $a$ (using $a \in \{\infty, 5, 3, 2\}$); improving the overall acceptance rate from approximately 17% for $a = \infty$ (the original SVE algorithm) to approximately 74% for $a = 5$.

The idea of generating proposals until the statistic $t(x^*)$ falls within a certain range $\mathcal{T}_a$ is closely related to the idea behind the ABC-rejection algorithm, where one simply accepts a proposed point when $t(x^*) \in \mathcal{T}_a$ [21, 22]. When $a$ is "sufficiently small" the proposed point $\theta^*$ will be drawn from a posterior distribution $\pi(\theta \mid t(x^*))$ that is "close" to the target distribution $\pi(\theta \mid t(x))$. For the SVE approach $a$ need not be "sufficiently small" as the Metropolis kernel ensures that the correct posterior distribution $\pi(\theta \mid x)$ is the invariant distribution. It is clear that decreasing the value of $a$ implies higher acceptance rates, but also that more samples are required to produce a value $t(x^*)$ in $\mathcal{T}_a$, on average. However, when there are multiple target posterior distributions one could bin the observed statistics into $m$ non-overlapping ranges, and apply recycling to the $m$ bins to reduce the number of samples that need to be produced.

## 3.5 A Data Augmentation Procedure

Matching and oversampling use auxiliary- or sufficient statistics $t(x^*)$ to choose more efficient proposals. Marsman, Maris, Bechger and Glas [29] generalized this approach by making clever use of the augmented variables that are often used to sample from $\pi(x \mid \theta)$. For example, logistic random variables are commonly used to sample item scores in the Rasch model:

$$\pi(X = 1 \mid \theta,\ \delta) = \int_{-\infty}^{\theta-\delta} \frac{\exp(-z)}{(1 + \exp(-z))^2}\, dz. \tag{4}$$

Although Gibbs samplers are frequently used to sample from the augmented posteriors $\pi(z, \theta \mid x)$, such procedures tend to converge slowly. The solution to this problem is to only use the augmented variables to generate proposals from a slightly different model that, when used in an independence chain (i.e., the SVE algorithm), does not suffer from this slow convergence.

Consider the distribution of $\mathbf{w} = \{z_1, \ldots, z_k, \theta\}$; the joint distribution of the augmented variables and the parameter. We have that the conditional distribution $\pi(w^*_{k+1} \mid \mathbf{w}^*_{\setminus k+1})$ corresponds to a unique posterior distribution $\pi(w^*_{k+1} = \theta^* \mid x^*)$, where $x^*$ is a function of $\mathbf{w}^*$ defined through relations as Eq (4) (i.e., $x^*_i = 1$ if $w^*_i < w^*_{k+1}$, and 0 otherwise). Note that this

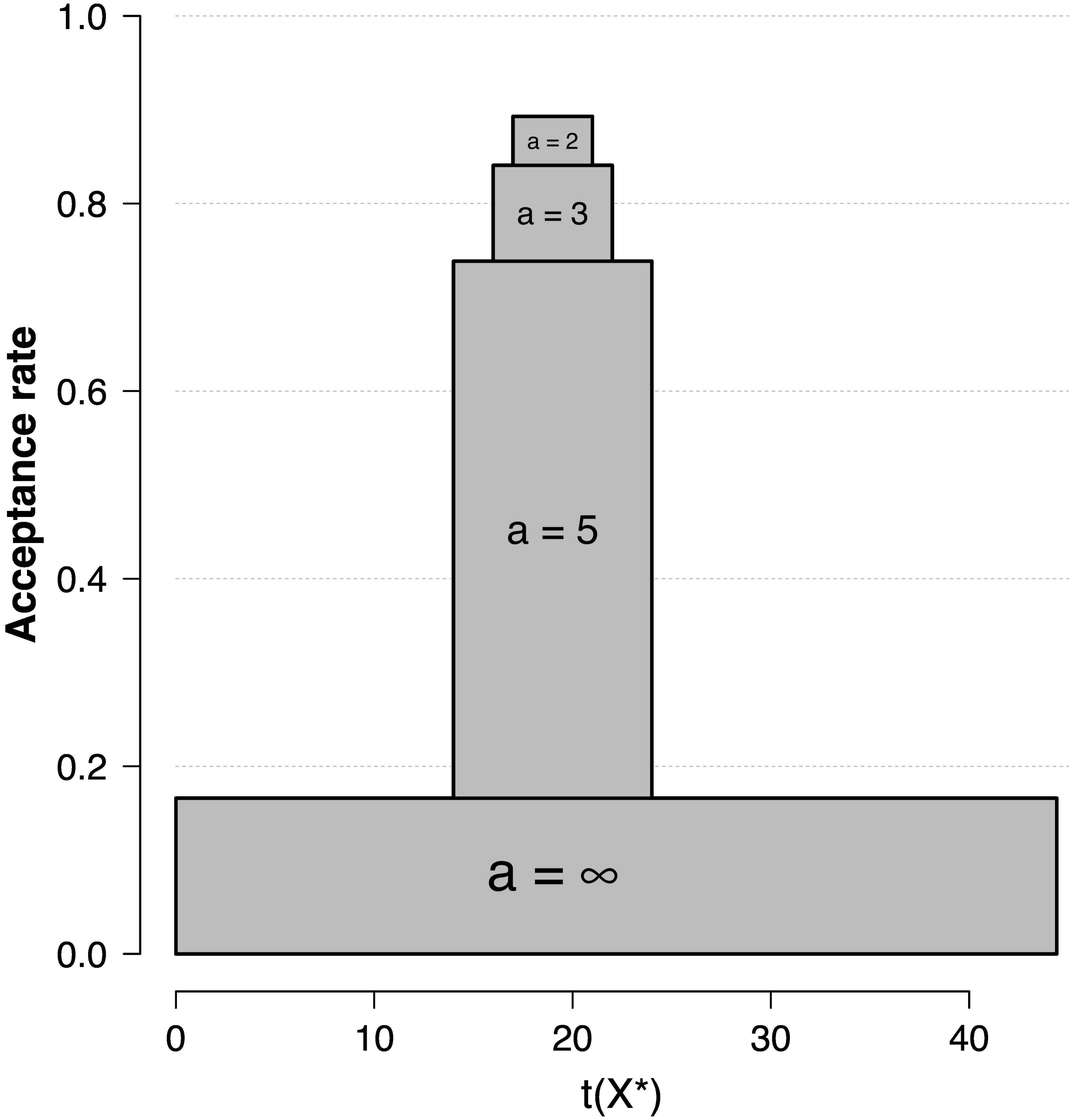

**Fig 5. Acceptance rates for the SVE algorithm with different bin sizes *a*.** The average proportion of accepted points when proposals are generated until $t(x^*)$ falls in the range $(t(x) - a, t(x) + a)$ using $a \in \{\infty, 5, 3, 2\}$. The gray bars reflect both the range (left and right endpoints) and the proportion of accepted points (top).

doi:10.1371/journal.pone.0169787.g005

relation can also be used to define posteriors $\pi(w_i^* \mid y^*)$ for each of the $k$ remaining conditional distributions $\pi(w_i^* \mid \mathbf{w}_{\setminus i}^*)$. The difference between the posterior $\pi(w_{k+1}^* \mid x^*)$ and $\pi(w_i^* \mid y^*)$ is that $w_{k+1} = \theta$ is used as the augmented variable in $\pi(w_i^* \mid y^*)$ and $w_i = z_i$ is the proposed point, whereas $w_{k+1} = \theta$ is the proposed point in $\pi(w_{k+1}^* \mid x^*)$ and $w_i = z_i$ is used as the augmented variable to generate $x_i^*$. This means that $\pi(w_i^* \mid y^*)$ is a posterior distribution that corresponds to a slightly different model that uses the same model components, but where one of the likelihood

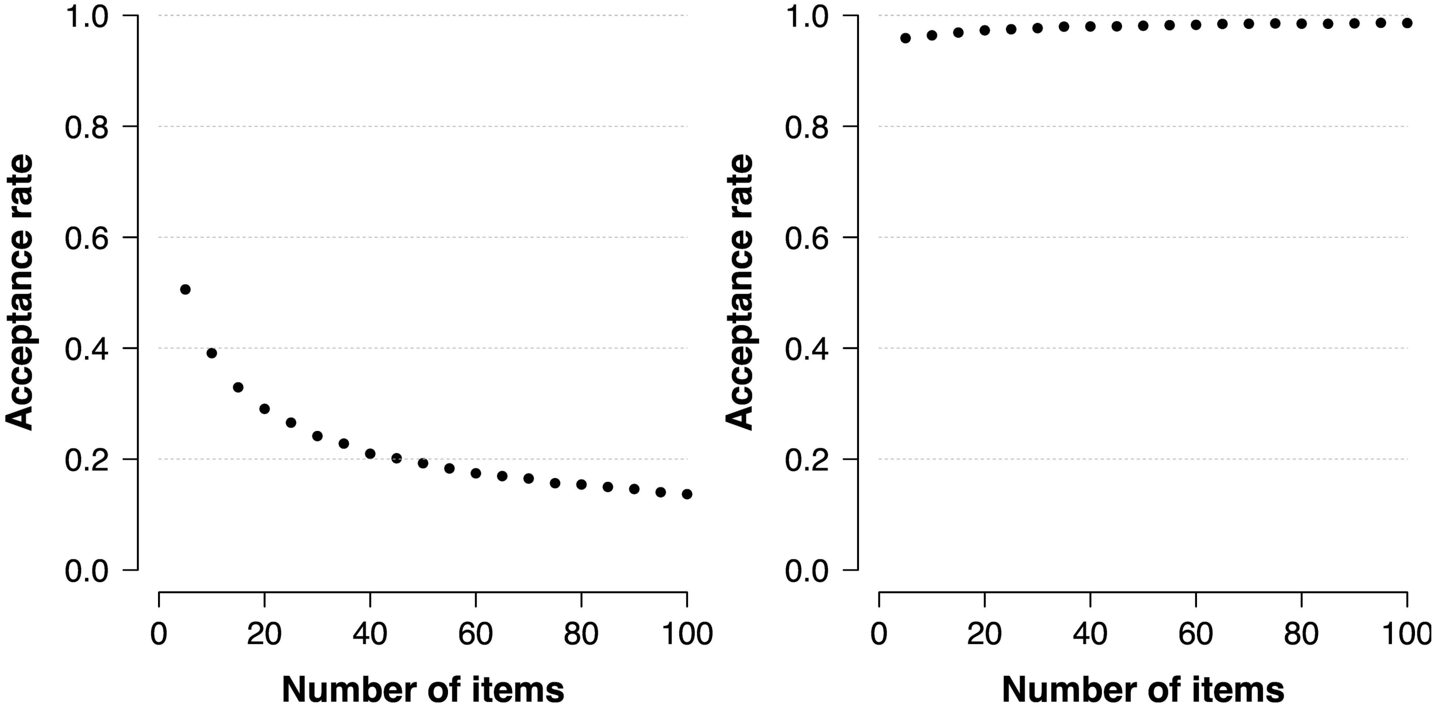

**Fig 6. Acceptance rates for the original SVE algorithm and SVE using $\pi(w^* \mid y_+^*)$ as proposal.** The average acceptance rate for sampling from posterior distributions $\pi(\theta \mid x_+)$ when using the original SVE algorithm (left panel) and when using the proposal distribution $\pi(w^* \mid y_+^* = x_+)$ (right panel).

doi:10.1371/journal.pone.0169787.g006

functions switched places with the prior distribution (see Ref. [29] for more details). What makes this approach interesting is that from generating a single data vector we obtain $k + 1$ proposal distributions and we can choose the statistically most efficient one.

Fig 6 illustrates the approach with sampling from posterior distributions $\pi(\theta \mid x_+)$ in the Rasch model using $\pi(w^* \mid y_+^* = x_+)$ as the proposal distribution (right panel). Note that the procedure is statistically efficient and further improves when more observations become available; even though the posteriors become more concentrated. Also shown in Fig 6 is the original SVE approach (left panel), which becomes less efficient as the number of observations increases.

The procedure applies also to Logistic regression models, and, when the augmented variables have a non-logistic distribution, for instance a normal distribution, we obtain other Bernoulli regression models, such as probit regression [30]. Marsman et al. [29] used the procedure to estimate Ising network models using a full-data-information procedure, utilizing a latent variable expression of the Ising model, where the conditional distribution $\pi(x \mid \theta)$ was found to be a multidimensional two-parameter logistic model [31].

Exponential Family models are closed under conditioning, that is, $\pi(x \mid \theta, x \in \omega \subset \Omega_x)$ is also in the Exponential Family. In this manner, we find that generating responses to two Rasch items corresponds to a three category Partial Credit Model [32] whenever

$$(x_1, x_2) \in \{(0, 0), (1, 0), (1, 1)\} \subset \Omega_x,$$

and the procedure similarly extends to such situations. In principle, this procedure can be used to generate other models, such as multinomial logit models [33], and extends the framework of Marsman et al. [29] to Potts network models.

## 4 Discussion

With the SVE algorithm a powerful yet simple idea was introduced; any model that can be simulated can be estimated. Based on a mixture of Metropolis kernels representation we have built upon the idea introduced with the original SVE algorithm and suggested several approaches that produce significant improvements to the convergence and autocorrelation of the Markov chain. To keep things simple, we have focused explicitly on statistical models $\pi(x \mid \theta)$ that are in the Exponential Family. However, the approaches that we have proposed in this paper are more generally applicable and simple to implement.

## Supporting Information

**S1 Script. Annotated R-Code for Fig 1.**
(TXT)

**S2 Script. Annotated R-Code for Fig 2.**
(TXT)

**S3 Script. Annotated R-Code for Fig 3.**
(TXT)

**S4 Script. Annotated R-Code for Fig 4.**
(TXT)

**S5 Script. Annotated R-Code for Fig 5.**
(TXT)

**S6 Script. Annotated R-Code for Fig 6.**
(TXT)

**S7 Script. Annotated R-Code with Proposed Procedures.**
(TXT)

## Acknowledgments

We thank Lourens Waldorp for his valuable comments on an earlier version of the manuscript.

## Author Contributions

**Conceptualization:** MM GM.

**Formal analysis:** MM GM TB CG.

**Methodology:** MM GM.

**Software:** MM.

**Supervision:** GM TB CG.

**Visualization:** MM.

**Writing – original draft:** MM.

**Writing – review & editing:** MM GM TB CG.

## References

1. Tanner M. Tools for Statistical Inference: Methods for the Exploration of Posterior Distributions and Likelihood Functions. 2nd ed. New York, NY: Springer-Verlag; 1993. doi: 10.1007/978-1-4684-0192-9
2. Rubin D. Bayesianly justifiable and relevant frequency calculations for the applied statistician. The Annals of Statistics. 1984; 12(4):1151–1172. doi: 10.1214/aos/1176346785
3. Murray I, Ghahramani Z, MacKay DJC. MCMC for doubly-intractable distributions. In: Proceedings of the 22nd Annual Conference in Artificial Intelligence. UAI; 2006.
4. Tierney L. Markov chains for exploring posterior distributions. The Annals of Statistics. 1994; 22 (4):1701–1762. doi: 10.1214/aos/1176325755
5. Metropolis N, Rosenbluth A, Rosenbluth M, Teller A. Equation of State Calculations by Fast Computing Machines. The Journal of Chemical Physics. 1953; 21(6):1087–1092. doi: 10.1063/1.1699114
6. Möller J, Pettitt A, Reeves R, Berthelsen K. An efficient Markov chain Monte Carlo method for distributions with intractable normalising constants. Biometrika. 2006; 93(2):451–485. doi: 10.1093/biomet/93.2.451
7. Fisher RA. Two New Properties of Mathematical Likelihood. Proceedings of the Royal Society of London Series A, Containing Papers of a Mathematical and Physical Character. 1934; 144(852):285–307. doi: 10.1098/rspa.1934.0050
8. Pitman E. Sufficient Statistics and Intrinsic Accuracy. Mathematical Proceedings of the Cambridge Philosophical Society. 1936; 32(4):567–579. doi: 10.1017/S0305004100019307
9. Koopman B. On Distributions Admitting a Sufficient Statistic. Transactions of the American Mathematical Society. 1936; 39(3):399–409. doi: 10.1090/S0002-9947-1936-1501854-3
10. Caimo A, Friel N. Bayesian Inference for Exponential Random Graph Models. Social Networks. 2011; 33(1):41–55. doi: 10.1016/j.socnet.2010.09.004
11. Hunter D, Krivitsky P, Schweinberger M. Computational Statistical Methods for Social Network Models. Journal of Computational and Graphical Statistics. 2012; 21(4):856–882. doi: 10.1080/10618600.2012.732921 PMID: 23828720
12. Friel N, Pettitt A. Classification Using Distance Nearest Neighbours. Statistics and Computing. 2011; 21 (3):431–437. doi: 10.1007/s11222-010-9179-y
13. Cucala L, Marin J. Bayesian Inference on a Mixture Model With Spatial Dependence. Journal of Computational and Graphical Statistics. 2013; 22(3):584–597. doi: 10.1080/10618600.2013.805652
14. Friel N. Evidence and Bayes Factor Estimation for Gibbs Random Fields. Journal of Computational and Graphical Statistics. 2013; 22(3):518–532. doi: 10.1080/10618600.2013.778780
15. Alquier P, Friel N, Everitt R, Boland A. Noisy Monte Carlo: Convergence of Markov Chains With Approximate Transition Kernels. Statistics and Computing. 2016; 26(1):29–47. doi: 10.1007/s11222-014-9521-x
16. Caimo A, Mira A. Efficient Computational Strategies for Doubly Intractable Problems With Applications to Bayesian Social Networks. Statistics and Computing. 2015; 25(1):113–125. doi: 10.1007/s11222-014-9516-7
17. Tierney L. A Note on Metropolis-Hastings Kernels for general state spaces. Annals of applied probability. 1998; 8(1):1–9. doi: 10.1214/aoap/1027961031
18. Jacob P, Robert C, Smith M. Using Parallel Computation to Improve Independent Metropolis-Hastings Based Estimation. Journal of Computational and Graphical Statistics. 2011; 20(3):616–635. doi: 10.1198/jcgs.2011.10167
19. R Core Team. R: A Language and Environment for Statistical Computing. Vienna, Austria; 2016. Available from: https://www.R-project.org/.
20. Pritchard J, Seielstad M, Perez-Lezaun A, Feldman M. Population Growth of Human Y Chromosomes: A Study of Y Chromosome Microsatellites. Molecular Biology and Evolution. 1999; 16(12):1791–1798. doi: 10.1093/oxfordjournals.molbev.a026091 PMID: 10605120
21. Marjoram P, Molitor J, Plagnol V, Tavare S. Markov chain Monte Carlo Without Likelihoods. Proceedings of the National Academy of Sciences of the United States of America. 2003; 100(26):15324–15328. doi: 10.1073/pnas.0306899100 PMID: 14663152
22. Sisson S, Fan Y, Tanaka M. Sequential Monte Carlo without Likelihoods. Proceedings of the National Academy of Sciences of the United States of America. 2007; 104(6):1760—1765. doi: 10.1073/pnas.0607208104 PMID: 17264216
23. Ripley BD. Stochastic Simulation. Wiley; 1987. doi: 10.1002/9780470316726
24. Ross S. Simulation. 5th ed. Academic Press; 2013.

25. Liang F. A Double Metropolis-Hastings Sampler for Spatial Models with Intractable Normalizing Constants. Journal of Statistical Computation and Simulation. 2010; 80(9):1007–1022. doi: 10.1080/00949650902882162

26. Liang F, Jin I, Sing Q, Liu J. An Adaptive Exchange Algorithm for Sampling From Distributions With Intractable Normalizing Constants. Journal of the American Statistical Association. 2015; 111 (513):377–393. doi: 10.1080/01621459.2015.1009072

27. Rasch G. Probabilistic Models for some Intelligence and Attainment Tests. Copenhagen: The Danish Institute of Educational Research; 1960.

28. Dawid A. Conditional Independence in Statistical Theory. Journal of the Royal Statistical Society, Series B (Methodological). 1979; 41(1):1–31.

29. Marsman M, Maris G, Bechger T, Glas C. Bayesian Inference for Low-Rank Ising Networks. Scientific Reports. 2015; 5(9050):1—7. doi: 10.1038/srep09050 PMID: 25761415

30. Albert J, Chib S. Bayesian Analysis of Binary and Polytomous Response Data. Journal of the American Statistical Association. 1993; 88(422):669–679. doi: 10.1080/01621459.1993.10476321

31. Reckase MD. Multidimensional item response theory. Springer; 2009. doi: 10.1007/978-0-387-89976-3

32. Masters G. A Rasch Model for Partial Credit Scoring. Psychometrika. 1982; 47(2):149–174. doi: 10.1007/BF02296272

33. Scott S. Data Augmentation, Frequentist Estimation, and the Bayesian analysis of multinomial logit models. Statistical Papers. 2011; 52(1):87–109. doi: 10.1007/s00362-009-0205-0