Master's thesis
Master's Programme in Data Science

Differentially Private Markov Chain Monte Carlo

Ossi Räisä

October 6, 2020

Supervisor(s): Professor Antti Honkela

Examiner(s): Professor Antti Honkela
Dr. Antti Koskela

University of Helsinki
Faculty of Science

P. O. Box 68 (Pietari Kalmin katu 5)
00014 University of Helsinki

HELSINGIN YLIOPISTO — HELSINGFORS UNIVERSITET — UNIVERSITY OF HELSINKI

Tiedekunta — Fakultet — Faculty	Koulutusohjelma — Utbildningsprogram — Degree programme
Faculty of Science	Master's Programme in Data Science

Tekijä — Författare — Author

Ossi Räisä

Työn nimi — Arbetets titel — Title

Differentially Private Markov Chain Monte Carlo

Työn laji — Arbetets art — Level	Aika — Datum — Month and year	Sivumäärä — Sidantal — Number of pages
Master's thesis	October 6, 2020	22

Tiivistelmä — Referat — Abstract

ACM Computing Classification System (CCS):

Avainsanat — Nyckelord — Keywords

Differential Privacy, Markov Chain Monte Carlo

Säilytyspaikka — Förvaringsställe — Where deposited

Muita tietoja — Övriga uppgifter — Additional information

Contents

1. Introduction

2. Background

2.1 Differential Privacy

Differential privacy [4] is a property of an algorithm that quantifies the amount of information about private data an adversary can gain from the publication of the algorithm's output. The most commonly used definition uses two real numbers, ϵ and δ, to quantify the information gain, or, from the perspective of a data subject, the privacy loss of the algorithm.

The most common definition is called (ϵ, δ)-DP, approximate DP or ADP [4]. The case where $\delta = 0$ is called ϵ-DP or pure DP.

Definition 1. *An algorithm $\mathcal{M}\colon \mathcal{X} \to \mathcal{U}$ is (ϵ, δ)-ADP if for all neighbouring inputs $x \in \mathcal{X}$ and $x' \in \mathcal{X}$ and all measurable sets $S \subset \mathcal{U}$*

$$P(\mathcal{M}(x) \in S) \leq e^{\epsilon} P(\mathcal{M}(x') \in S) + \delta$$

The neighbourhood relation in the definition is domain specific. With tabular data the most common definitions are the add/remove neighbourhood and substitute neighbourhood.

Definition 2. *Two tabular datasets are said to be add/remove neighbours if they are equal after adding or removing at most one row to or from one of them. The datasets are said to be in substitute neighbours if they are equal after changing at most one row in one of them.*

The neighbourhood relation is denoted by $\sim$. The definitions and theorems of this section are valid for all neighbourhood relations.

There many other definitions of differential privacy that are mostly used to compute (ϵ, δ)-bounds for ADP. This thesis uses two of them: Rényi-DP (RDP) [7] and zero-concentrated differential privacy (zCDP) [2]. Both are based on Rényi divergence [7], which is a particular way of measuring the difference between random variables.

Definition 3. *For random variables with density or probability mass functions P and*

Q the Rényi divergence of order $1 < \alpha < \infty$ is

$$D_\alpha(P \mid\mid Q) = \frac{1}{\alpha - 1} \ln E_{x \sim Q} \left(\frac{P(x)^\alpha}{Q(y)^\alpha} \right)$$

Orders $\alpha = 1$ and $\alpha = \infty$ are defined by continuity:

$$D_1(P \mid\mid Q) = \lim_{\alpha \to 1-} D_\alpha(P \mid\mid Q)$$

$$D_\infty(P \mid\mid Q) = \lim_{\alpha \to \infty} D_\alpha(P \mid\mid Q)$$

Both Rényi-DP and zCDP can be expressed as bounds on the Rényi divergence between the outputs of an algorithm with neighbouring inputs:

Definition 4. *An algorithm $\mathcal{M}$ is (α, ϵ)-Rényi DP if for all $x \sim x'$*

$$D_\alpha(\mathcal{M}(x) \mid\mid \mathcal{M}(x')) \leq \epsilon$$

$\mathcal{M}$ is ρ-zCDP if for all $\alpha > 1$ and all $x \sim x'$

$$D_\alpha(\mathcal{M}(x) \mid\mid \mathcal{M}(x')) \leq \rho\alpha$$

A very useful property of all of these definitions is composition [4]: if algorithms $\mathcal{M}$ and $\mathcal{M}'$ are DP, the algorithm first computing $\mathcal{M}$ and then $\mathcal{M}'$, outputting both results, is also DP, although with worse bounds. More precisely

Definition 5. *Let $\mathcal{M} \colon \mathcal{X} \to \mathcal{U}$ and $\mathcal{M}' \colon \mathcal{X} \times \mathcal{U} \to \mathcal{U}'$ be algorithms. Their composition is the algorithm outputting $(\mathcal{M}(x), \mathcal{M}'(x, \mathcal{M}(x)))$ for input x.*

Theorem 1. *Let $\mathcal{M} \colon \mathcal{X} \to \mathcal{U}$ and $\mathcal{M} \colon \mathcal{X} \times \mathcal{U} \to \mathcal{U}'$ be algorithms. Then*

1. *If $\mathcal{M}$ is (ϵ, δ)-ADP and $\mathcal{M}'$ is (ϵ', δ')-ADP, then their composition is $(\epsilon+\epsilon', \delta+\delta')$-ADP [4]*
2. *If $\mathcal{M}$ is (α, ϵ)-RDP and $\mathcal{M}'$ is (α, ϵ')-RDP, then their composition is $(\alpha, \epsilon + \epsilon')$-RDP [7]*
3. *If $\mathcal{M}$ is ρ-zCDP and $\mathcal{M}'$ is ρ'-zCDP, then their composition is $(\rho + \rho')$-zCDP [2]*

All of the composition results can be extended to any number of compositions by induction. Note that any step of the composition can depend on the results of the previous steps, not only on the private data.

As any algorithm that does not use private data in any way is $(0, 0)$-ADP, 0-zCDP and $(\alpha, 0)$-RDP with all α, theorem 1 has the following corollary, called post-processing immunity:

Theorem 2. *Let $\mathcal{M}\colon \mathcal{X} \to \mathcal{U}$ be an ADP, RDP or zCDP algorithm with some privacy parameters. Let $f\colon \mathcal{U} \to \mathcal{U}'$ be any algorithm not using the private data. Then the composition of $\mathcal{M}$ and f is ADP, RDP or zCDP with the same privacy parameters.*

There are many different DP algorithms that are commonly used, which are also called mechanisms [4]. This thesis only requires one of the most commonly used ones: the Gaussian mechanism [4].

Definition 6. *The Gaussian mechanism with parameter σ^2 is an algorithm that, with input x, outputs a sample from $\mathcal{N}(x, \sigma^2)$, where $\mathcal{N}$ denotes the normal distribution.*

The RDP and zCDP bounds for the Gaussian mechanism are quite simple. The ADP bound is more complicated:

Theorem 3. *If for all inputs x and x', $||x - x'||_2 \leq \Delta$, the Gaussian mechanism is*

1. *$(\alpha, \frac{\alpha\Delta^2}{2\sigma^2})$-RDP [7]*
2. *$\frac{\Delta^2}{2\sigma^2}$-zCDP [2]*
3. *n compositions of the Gaussian mechanism are $(\epsilon, \delta(\epsilon))$-ADP [8] with*

$$\delta(\epsilon) = \frac{1}{2}\left(\operatorname{erfc}\left(\frac{\sigma(\epsilon - n\mu)}{\sqrt{2n}\Delta}\right) - e^{\epsilon}\operatorname{erfc}\left(\frac{\sigma(\epsilon + n\mu)}{\sqrt{2n}\Delta}\right)\right)$$

where $\mu = \frac{\Delta^2}{2\sigma^2}$ and erfc *is the complementary error function.*

The most common use case for the Gaussian mechanism is computing a function $f\colon \mathcal{X} \to \mathbb{R}$ of private data and feeding the result into the Gaussian mechanism to privately release the function value. The condition that the inputs of the Gaussian mechanism cannot vary too much leads into the concept of sensitivity of a function

Definition 7. *The l_p-sensitivity Δ_p, with neighbourhood relation $\sim$, of a function $f\colon \mathcal{X} \to \mathbb{R}^n$ is*

$$\Delta_p f = \sup_{x \sim x'} ||f(x) - f(x')||_p$$

Theorem 3 implies that the value of any function with finite l_2-sensitivity can be privately released using the Gaussian mechanism with appropriate noise variance σ^2. Of course, the usefulness of the released value depends on the magnitude of σ^2 compared to the actual value.

2.2 Bayesian Inference and Markov Chain Monte Carlo

In Bayesian inference, the parameters of a statistical model are inferred from observed data using Bayes' theorem [5]. The result is not just a point estimate of the parameters, but a probability distribution describing the likelihood of different values of the parameters.

Bayes' theorem relates the *posterior* belief of the parameters $p(\theta \mid D)$ to the *prior* belief $p(\theta)$ through the observed data D and the likelihood of the data $p(D \mid \theta)$ as follows:

$$p(\theta \mid D) = \frac{p(D \mid \theta)p(\theta)}{\int p(D \mid \theta)p(\theta)d\theta}$$

It is theoretically possible to compute $p(\theta \mid D)$ given any likelihood, prior and data, but the integral in the denominator is in many cases difficult to compute [5]. In such cases the posterior cannot be feasibly computed. However, many of the commonly used summary statistics of the posterior, such as the mean, variance and credible intervals, can be approximated from a sample of the posterior. *Markov chain Monte Carlo* (MCMC) is a widely used algorithm to obtain such samples [5].

Markov chain Monte Carlo algorithms sequentially sample values of θ with the goal of eventually having the chain of sampled values converge to a given distribution [5]. While this can be done in many ways, this thesis focuses on a particular MCMC algorithm: *Metropolis-Hastings* (MH).

The Metropolis-Hastings algorithm samples from a distribution π of θ_i by first picking a proposal θ^* from a proposal distribution $q(\theta_{i-1})$ at iteration i [5]. A density ratio is calculated

$$r = \frac{\pi(\theta^*)}{\pi(\theta_{i-1})} \frac{q(\theta_{i-1} \mid \theta^*)}{q(\theta^* \mid \theta_{i-1})}$$

and the proposal is accepted with probability $\min\{1, r\}$. If the proposal is accepted, $\theta_i = \theta^*$, otherwise $\theta_i = \theta_{i-1}$.

It can be shown that, with a suitable proposal distribution, the chain of θ_i values converges to π [5]. The Gaussian distribution centered at the current value is a commonly used proposal.

When MCMC is used in Bayesian inference, the distribution to approximate is

$$\pi(\theta) = p(\theta \mid D) = \frac{p(D \mid \theta)p(\theta)}{\int p(D \mid \theta)p(\theta)d\theta}$$

The difficult integral $\int p(D \mid \theta)p(\theta)d\theta$ in the denominator cancels out when computing r, so only the likelihood and prior are needed. For numerical stability, r is usually

computed in log space, which makes the acceptance probability $\min\{1, e^{\lambda}\}$ where

$$\lambda = \ln \frac{p(\theta^* \mid D)}{p(\theta_{i-1} \mid D)} + \ln \frac{p(\theta^*)}{p(\theta_{i-1})} + \ln \frac{q(\theta_{i-1} \mid \theta^*)}{q(\theta^* \mid \theta_{i-1})}$$

The dataset D is typically a table with n independent rows. The likelihood is given as

$$p(\theta \mid D_j)$$

for row D_j. The independence means that

$$p(\theta \mid D) = \prod_{j=1}^{k} p(\theta \mid D_j)$$

which means that the log likelihood ratio term of λ is

$$\ln \frac{p(\theta^* \mid D)}{p(\theta_{i-1} \mid D)} = \sum_{j=1}^{n} \ln \frac{p(\theta^* \mid D_j)}{p(\theta_{i-1} \mid D_j)}$$

Algorithm 1 puts all of this together to summarise the MH algorithm used for Bayesian inference.

Algorithm 1: Metropolis-Hastings: number of iterations k, proposal distribution q and initial value θ_0 and dataset D as input

for $1 \leq i \leq k$ **do**

sample $\theta^* \sim q(\theta_{i-1})$

$\lambda = \sum_{j=1}^{n} \ln \frac{p(\theta^*|D_j)}{p(\theta_{i-1}|D_j)} + \ln \frac{p(\theta^*)}{p(\theta_{i-1})} + \ln \frac{q(\theta_{i-1}|\theta^*)}{q(\theta^*|\theta_{i-1})}$

$$\theta_i = \begin{cases} \theta^* & \text{with probability } \min\{1, e^{\lambda}\} \\ \theta_{i-1} & \text{otherwise} \end{cases}$$

end

return $(\theta_1, \ldots, \theta_k)$

3. Differentially Private MCMC

As seen in Section 2.1, an algorithm can be made differentially private by adding Gaussian noise the its output. The noise could also be added to any intermediate value calculated by the algorithm, and post processing immunity will guarantee that the same DP bounds that hold for releasing the intermediate value also hold for releasing the final result of the algorithm.

In 2019, Yildirim and Ermis [10] realised that if Gaussian noise is added to the exact value of λ, the noise can be corrected for yielding a differentially private MCMC algorithm which converges to the correct distribution. In the same year, Heikkilä et. al. [6] developed another DP MCMC algorithm, called DP Barker, which uses subsampling to amplify privacy.

3.1 DP Penalty

In 1999, Ceperley and Dewing [3] developed a variant of Metropolis-Hastings called the penalty algorithm, where only a noisy approximation of λ is known. They developed the algorithm for simulations in physics where computing λ requires computing energies of complex systems, which can only be approximated. The penalty algorithm modifies the acceptance probability to account for the noise added to λ and still converges to the correct distribution if the noise is Gaussian with known variance.

3.2 DP Barker

The DP Barker algorithm of Heikkilä et. al. [6] is based on the Barker acceptance test [1] instead of the Metropolis-Hastings test. Instead of using the MH acceptance probability, the Barker acceptance test samples $V_{log} \sim \text{Logistic}(0, 1)$ and accepts if

$$\lambda + V_{log} > 0$$

If Gaussian noise with variance σ^2 is added to λ, there exists a correction distribution V_{corr} such that $\mathcal{N}(0, \sigma^2) + V_{corr}$ has the same distribution as V_{log}. Because the

variance of V_{log} is $\frac{\pi^2}{3}$, the variance of V_{corr} must be $\frac{\pi^2}{3} - \sigma^2$ which means that there is an upper bound to the noise variance: $\sigma^2 < \frac{\pi^2}{3}$. Testing whether $\lambda + \mathcal{N}(0, \sigma^2) + V_{corr} > 0$ is equivalent to testing whether $\lambda + V_{log} > 0$, which means that it is possible to derive a DP MCMC algorithm based on the Barker acceptance test if the correction distribution can be sampled from.

However, the analytical form of V_{corr} is not known [6]. Heikkilä et. al. [6] derive a method to accurately approximate the distribution and draw samples from the approximation. This means that their algorithm does only approximately converges to the correct distribution, but the error in approximating V_{corr} can be made very small.

If the sum in λ was only computed over a subset of the data, the algorithm would take less computation to run, and would be less sensitive to changes in the data. The latter property is called *subsampling amplification* of differential privacy [9]. The using the λ computed with subsampling instead of the full data λ introduces an additional error that must be correct for to have the algorithm converge to the correct distribution.

The *central limit theorem* (CLT) states that the distribution of a sum of random variables approaches a Gaussian distribution as more random variables are summed, if some conditions on the independence and variance of the random variables are met [6]. With the CLT, it can be argued that the error from using the subsampled λ instead of the full data λ has an approximately Gaussian distribution, if the subsample is large enough [6].

The variance of the error from subsampling can be estimated by the sample variance of the individual terms in the sum in λ [6]. This allows combining the errors from subsampling and the Gaussian noise from the Gaussian mechanism to a single Gaussian noise value. The V_{corr} distribution can then be used to approximate the

Barker acceptance test as above. See algorithm 2 for the DP Barker algorithm *.

Algorithm 2: DP Barker

sample $\theta^* \sim q(\theta_{i-1})$
sample $B \subset \{1, \ldots, n\}$
for $1 \leq i \leq k$ **do**
  for $j \in B$ **do**
    $l_j = \ln \frac{p(\theta^*|D_j)}{p(\theta_{i-1}|D_j)}$
  end
  $\sigma_b^2 = \mathrm{Var}\{l_j \mid j \in B\}$
  $\lambda = \frac{nT}{|B|} \sum_{j \in B} l_j + \ln \frac{p(\theta^*)}{p(\theta_{i-1})} + \ln \frac{q(\theta_{i-1}|\theta^*)}{q(\theta^*|\theta_{i-1})}$
  sample $s \sim \mathcal{N}(0, \sigma^2 - \sigma_b^2)$
  sample $c \sim V_{corr}^{\sigma^2}$
  $\theta_i = \begin{cases} \theta^* & \text{if } \lambda + s + c > 0 \\ \theta_{i-1} & \text{otherwise} \end{cases}$
end
return $(\theta_1, \ldots, \theta_k)$

3.3 Comparing DP Penalty and DP Barker

*See [6] for the sampling procedure of V_{corr}.

4. Variations of the Penalty Algorithm

4.1 The Penalty Algorithm with Subsampling

4.2 DP Metropolis-Adjusted Langevin Algorithm

5. The Gauss-Bernoulli Algorithm

6. Experiments

7. Conclusions

Bibliography

[1] A. A. Barker. Monte carlo calculations of the radial distribution functions for a proton-electron plasma. *Australian Journal of Physics*, 18(2):119–134, 1965.

[2] M. Bun and T. Steinke. Concentrated differential privacy: Simplifications, extensions, and lower bounds. In *Theory of Cryptography - 14th International Conference, TCC 2016-B, Beijing, China, October 31 - November 3, 2016, Proceedings, Part I*, pages 635–658, 2016.

[3] D. Ceperley and M. Dewing. The penalty method for random walks with uncertain energies. *The Journal of chemical physics*, 110(20):9812–9820, 1999.

[4] C. Dwork and A. Roth. The algorithmic foundations of differential privacy. *Foundations and Trends in Theoretical Computer Science*, 9(3-4):211–407, 2014.

[5] A. Gelman, J. B. Carlin, H. S. Stern, D. B. Dunson, A. Vehtari, and D. B. Rubin. *Bayesian data analysis*. Chapman & Hall/CRC texts in statistical science series. CRC Press, Boca Raton, third edition, 2014.

[6] M. A. Heikkilä, J. Jälkö, O. Dikmen, and A. Honkela. Differentially private markov chain monte carlo. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, 8-14 December 2019, Vancouver, BC, Canada*, pages 4115–4125, 2019.

[7] I. Mironov. Rényi differential privacy. In *30th IEEE Computer Security Foundations Symposium, CSF 2017, Santa Barbara, CA, USA, August 21-25, 2017*, pages 263–275, 2017.

[8] D. M. Sommer, S. Meiser, and E. Mohammadi. Privacy loss classes: The central limit theorem in differential privacy. *PoPETs*, 2019(2):245–269, 2019.

[9] Y. Wang, B. Balle, and S. P. Kasiviswanathan. Subsampled renyi differential privacy and analytical moments accountant. In *The 22nd International Conference on Artificial Intelligence and Statistics, AISTATS 2019, 16-18 April 2019, Naha, Okinawa, Japan*, pages 1226–1235, 2019.

[10] S. Yildirim and B. Ermis. Exact MCMC with differentially private moves - revisiting the penalty algorithm in a data privacy framework. *Statistics and Computing*, 29(5):947–963, 2019.