Data Augmentation for Privacy Aware Analysis

DP Group

September 2023

Abstract

This paper serves as a reference and introduction on using the **dadp** R package. The goal of this package is to provide some tools for exploring the impact of different privacy regimes on a Bayesian analysis. A strength of this framework is the ability to target the exact posterior in settings where the likelihood is too complex to analytically express.

Methodology

(Insert from DA paper?)

Using dadp

Introduce some basic notation (here or in intro) Consider combining this section with methodology.

Using the dadp package consist of specifying four components

1. $\pi(\theta \mid x)$.
2. $f(x \mid \theta)$.
3. $\eta(s_{dp} \mid x)$.
4. $T(x)$.

Differentially Private Simple Linear Regression

(Alabi et al. 2020)[1] considers adding noise to a sufficient static to create a differentially private algorithm for simple linear regression called **NoisyStats**

Suppose we would like to explore the potential impact of the **NoisyStats** mechanism on analysis. Assume the true data generating process is

$$x_i \sim Unif(0, 1)$$
$$y_i \sim N(-2 + 3x_i, 3^2)$$

We would like to perform inference on (α, β) given privatized statistic $(\tilde{s}_1, \tilde{s}_2)$.

Algorithm 1 NoisyStats: $(\epsilon, 0)$-DP Algorithm (closer to original paper)

Data: $\{(x_i, y_i)\}_{i=1}^n$
Privacy Parameter: ϵ
$\Delta_1 = \Delta_2 = (1 - 1/n)$ ▷ *Set global sensitivity*
Sample $L_1 \sim Lap(0, 3\Delta_1/\epsilon)$
Sample $L_2 \sim Lap(0, 3\Delta_2/\epsilon)$
if $nvar(x) + L_2 > 0$ **then**
    $\tilde{\beta} = \frac{ncov(x,y) + L_1}{nvar(x) + L_2}$
    $\Delta_3 = (1/n)(1 + |\tilde{\alpha}|)$
    Sample $L_3 \sim Lap(0, 3\Delta_3/\epsilon)$
    $\tilde{\alpha} = (\bar{y} - \tilde{\beta}\bar{x}) + L_3$
    return $(ncov(x,y) + L_1, nvar(x) + L_2, \tilde{\alpha})$
return NA

Algorithm 2 NoisyStats: $(\epsilon, 0)$-DP Algorithm (easy for me!)

Data: $\{(x_i, y_i)\}_{i=1}^n$
Privacy Parameter: ϵ
$\Delta_1 = \Delta_2 = (1 - 1/n)$ ▷ *Set global sensitivity*
$\Delta_3 = \Delta_4 = 1/n$
Sample $L_1 \sim Lap(0, 3\Delta_1/\epsilon)$, $L_2 \sim Lap(0, 3\Delta_2/\epsilon)$
Sample $L_3 \sim Lap(0, 3\Delta_3/\epsilon)$, $L_4 \sim Lap(0, 3\Delta_4/\epsilon)$
$\tilde{s}_1 = ncov(x,y) + L_1$
$\tilde{s}_2 = nvar(x) + L_2$
$\tilde{s}_3 = \bar{y} + L_3$
$\tilde{s}_4 = \bar{x} + L_4$
return $(\tilde{s}_1, \tilde{s}_2, \tilde{s}_3, \tilde{s}_4)$

Sampling from likelihood under complete data

likelihood function $f(x) \sim Unif(0, 1)$.

$$\begin{aligned} f(y \mid x, \alpha, \beta) &= f(x \mid \mu_x, \sigma_x) f(y \mid x, \alpha, \beta) \\ &= \phi(x; \mu_x, \sigma_x)\phi(y; \alpha + \beta x, \sigma) \end{aligned}$$

```
lik_smpl <- function(theta) {
  alpha <- theta[1]
  beta <- theta[2]
  x <- runif(1)
  y <- rnorm(1, mean = alpha + beta * x, sd = 3)
  c(x,y)
}
```

Posterior given complete data

Assume $f(\alpha, \beta) \sim N(0, 10^{-2} I_{2\times 2})$

$$\begin{aligned} \mu_p &= (1/9)\Sigma_p^{-1} X^T y \\ \Sigma_p^{-1} &= (1/9)X^T X + (1/100)I^{-1} \end{aligned}$$

```
post_smpl <- function(dmat, theta) {
  x <- dmat[,1]
  y <- dmat[,2]
  xm <- cbind(1, x)
  Si <- (1/9) * t(xm) %*% xm + (1/100) * diag(2)
  mu <- (1/9) * solve(Si) %*% t(xm) %*% y
  MASS::mvrnorm(1, mu = mu, Sigma = solve(Si))
}
```

Statistic

NoisyStat computes four summary statistics

$$\begin{aligned} nvar(x) &= \sum_{i=1}^{n} (x_i - \bar{x}) \\ ncov(x, y) &= \sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y}) \end{aligned}$$

```
st_calc <- function(dmat) {
  x <- dmat[,1]
  y <- dmat[,2]
  n <- length(y) - cov(x,y)/var(x)
  s1 <- (n-1) * cov(x,y)
  s2 <- (n-1) * var(x)
  s3 <- mean(y)
  s4 <- mean(x)
  c(s1, s2, s3, s4)
}
```

Privacy Mechanism

NoisyStat consist of adding independent Laplace errors to each of the three summary statistics

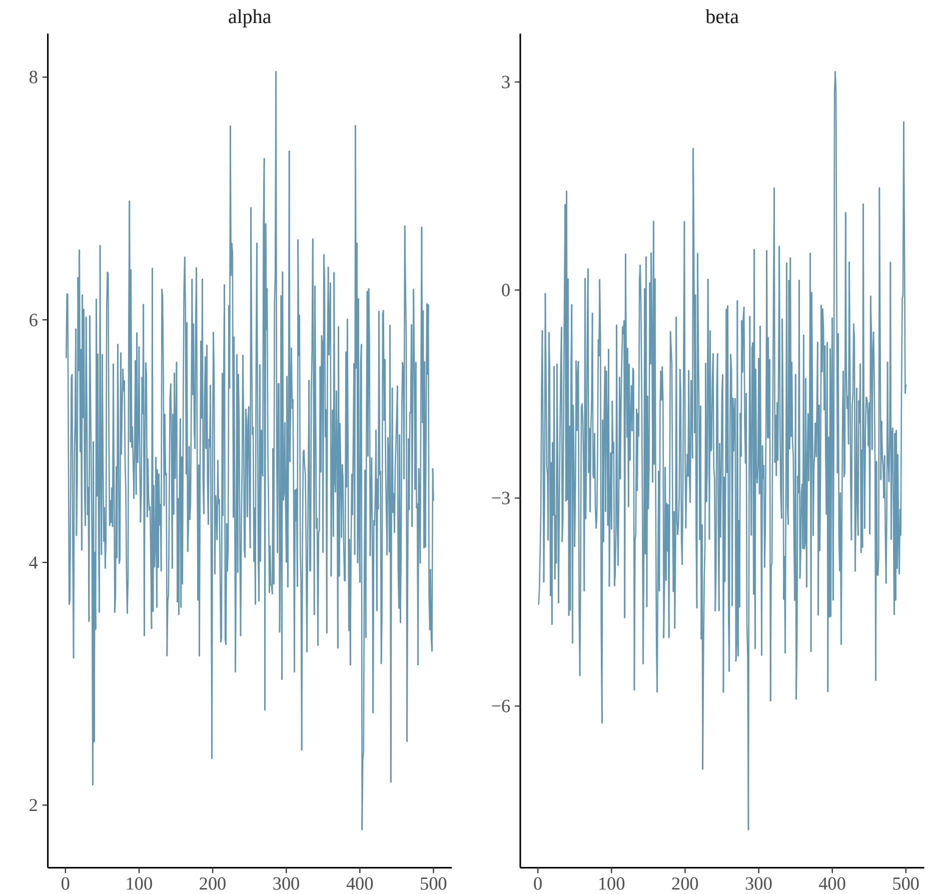
```
#check vectorization?
priv_mech_factory <- function(n, epsilon) {
  function(sdp, xt) {
    delta1 <- (1- 1/n)
    delta3 <- 1/n
    t1 <- VGAM::dlaplace(sdp[1] - xt[1], 0, 3 * delta1/epsilon, TRUE)
    t2 <- VGAM::dlaplace(sdp[2] - xt[2], 0, 3 * delta1/epsilon, TRUE)
    t3 <- VGAM::dlaplace(sdp[3] - xt[3], 0, 3 * delta3/epsilon, TRUE)
    t4 <- VGAM::dlaplace(sdp[4] - xt[4], 0, 3 * delta3/epsilon, TRUE)
    sum(c(t1,t2,t3,t4))
  }
}
```

Chain diagnostics?

```
summary(tmp)

## [1] "Average Acceptance Probability: 0.59056"
## # A tibble: 2 x 10
##   variable  mean median    sd   mad    q5   q95  rhat ess_bulk ess_tail
##   <chr>    <num>  <num> <num> <num> <num> <num> <num>    <num>    <num>
## 1 alpha     4.84   4.78 0.987  1.05  3.34 6.40  1.00      329.     369.
## 2 beta     -2.31  -2.35 1.67   1.76 -5.03 0.400 0.999     336.     396.

bayesplot::mcmc_trace(tmp$chain)
```



alpha
beta
8
6
4
2
3
0
−3
−6
0
100
200
300
400
500

A maybe...

As an example, use data from (Gelman). Problem consist of estimating the proportion of boys and girls. Data: 251,527 boys and 241,945 girls born in Paris from 1745 to 1770. Describe set up below

Privatize by adding noise, $\eta \sim N(0, 4000)$: [Use DP framework?]

```
n_g <- 241945
n_b <- 251527

eta <- rnorm(2,0,4000)

n_g + eta[1]

## [1] 249361.9

n_b + eta[2]

## [1] 247182.7
```

Sampling from likelihood under complete data

binomial distribution

$$f(x \mid \theta) = \binom{n}{n_g} \theta^{n_g} (1-\theta)^{n-n_g}$$

```
lik_smpl <- function(theta) {
  t1 <- rbinom(1, 493472, theta)
  t2 <- 493472 - t1
  c(t1,t2)
}
```

Posterior given complete data

Using Jeffrey's prior $Beta(1/2, 1/2)$.

$$\frac{\Gamma(\alpha)\Gamma(\beta)}{\Gamma(\alpha+\beta)} x^{\alpha-1}(1-x)^{\beta-1}$$

conjugate model:

References

[1] Daniel Alabi, Audra McMillan, Jayshree Sarathy, Adam Smith, and Salil Vadhan, *Differentially private simple linear regression*, (2020).