

732A73 Bayesian Learning

Computer lab 3

Johannes Hedström & Mikael Montén

STIMA
Department of Computer and Information Science
Linköpings universitet

2024-05-02

Contents

1	Gibbs sampling for the logistic regression	1
1.1	a	1
1.2	b	1
2	Metropolis Random Walk for Poisson regression	1
2.1	a)	2
2.2	b)	2
2.3	c)	2
2.4	d)	3
3	Time series models in Stan	3
3.1	a)	3
3.2	b)	3

```
library(coda)
library(readxl)
library(mvtnorm)
library(tidyr)
library(tinytex)
```

1 Gibbs sampling for the logistic regression

Consider again the logistic regression model in problem 2 from the previous computer lab 2. Use the prior $\beta \sim N(0, \tau^2)$ where $\tau = 3$

1.1 a

Implement (code!) a Gibbs sampler that simulates from the joint posterior $p(w, \beta | x)$ by augmenting the data with Polya-gamma latent variables $w_i, i = 1 \dots n$. The full conditional posteriors are given on the slides from Lecture 7. Evaluate the convergence of the gibbs sampler by calculating the Inefficiency Factors (IFs) and by plotting the trajectories of the samples Markov chains.

1.2 b

Use the posterior draws from a) to compute 90% equal tail credible interval for $Pr(y = 1 | x)$ where x corresponds to a 38-year-old woman with 1 child (3 years old), 12 years of education and 7 years of experience and a husband with an income of 22. A 90% equal tail credible interval (a,b) cuts the off 5% percent of the posterior probability mass to the left of a, and 5% to the right of b.

2 Metropolis Random Walk for Poisson regression

Consider the following Poisson regression model

$$y_i | \beta \stackrel{iid}{\sim} \text{Poisson}[\exp(x_i^T \beta)]$$

where y_i is the count for the i th observation in the sample and x_i is the p -dimensional vector with covariate observations for the i th observation. Use the data set `eBayNumberOfBidderData_2024.dat`. This dataset contains observations from 800 eBay auctions of coins. The response variable is `nBids` and records the number of bids in each auction. The remaining variables are features/covariates (x):

- `Const` (for the intercept)
- `PowerSeller` (equal to 1 if the seller is selling large volumes on eBay)
- `VerifyID` (equal to 1 if the seller is a verified seller by eBay)
- `Sealed` (equal to 1 if the coin was sold in an unopened envelope)
- `MinBlem` (equal to 1 if the coin has a minor defect)

- MajBlem (equal to 1 if the coin has a major defect)
- LargNeg (equal to 1 if the seller received a lot of negative feedback from customers)
- LogBook (logarithm of the book value of the auctioned coin according to expert sellers. Standardized)
- MinBidShare (ratio of the minimum selling price (starting price) to the book value. Standardized).

2.1 a)

Obtain the maximum likelihood estimator of β in the Poisson regression model for the eBay data [Hint: `glm.R`, don't forget that `glm()` adds its own intercept so don't input the covariate `Const`]. Which covariates are significant?

2.2 b)

Let's do a Bayesian analysis of the Poisson regression. Let the prior be $\beta \sim N[0, 100 \cdot (X^T X)^{-1}]$, where X is the $n \times p$ covariate matrix. This is a commonly used prior, which is called Zellner's g-prior. Assume first that the posterior density is approximately multivariate normal:

$$\beta \sim y \sim N(\hat{\beta}, J_y^{-1}(\hat{\beta}))$$

where $\hat{\beta}$ is the posterior mode and $J_y(\hat{\beta})$ is the negative hessian at the posterior mode. $\hat{\beta}$ and $J_y(\hat{\beta})$ can be obtained by numerical optimization(`optim.R`) exactly like you've already did for the first logistic regression in Lab 2 (but with the log posterior function replaced by the corresponding one for the Poisson model, which you have code up.).

2.3 c)

Let's simulate from the actual posterior of β using the Metropolis algorithm and compare the results with the approximate results in b). Program a general function that uses the Metropolis algorithm to generate random draws from an arbitrary posterior density. In order to show that it is a general function for any model, we denote the vector of model parameters by θ . Let the proposal density be the multivariate normal density mentioned in Lecture 8 (random walk Metropolis):

$$\theta_p | \theta^{(i-1)} \sim N(\theta^{(i-1)}, c \cdot \Sigma)$$

$\Sigma = J_y^{-1}(\hat{\beta})$ was obtained in b). The value c is a tuning parameter and should be an input to your Metropolis function. The user of your Metropolis function should be able to supply her own posterior density function, not necessarily for the Poisson regression, and still be able to use your Metropolis function. This is not so straightforward, unless you have come across function objects in R. The note `HowToCodeRWM.pdf` in `Lisam` describes how you can do this in R.

Now, use your new Metropolis function to sample from the posterior of β in the Poisson regression for the eBay dataset. Assess MCMC convergence by graphical methods

2.4 d)

Use the MCMC draws from c) to simulate from the predictive distribution of the number of bidders in a new auction with the characteristics below. Plot the predictive distribution. What is the probability of no bidders in this new auction?

- PowerSeller = 1
- VerifyID = 0
- Sealed = 1
- MinBlem = 0
- MajBlem = 1
- LargNeg = 0
- LogBook = 1.2
- MinBidShare = 0.8

3 Time series models in Stan

3.1 a)

Write a function in R that simulates data from the AR(1)-process

$$x_t = \mu + \phi(x_{t-1} - \mu) + \epsilon_t, \epsilon_t \stackrel{iid}{\sim} N(0, \sigma^2)$$

for given values of μ, ϕ and σ^2 . Start the process at $x_1 = \mu$ and then simulate values for x_t for $t = 2, 3, \dots, T$ and return the vector $x_{1:T}$ containing all time points. Use $\mu = 9$, $\sigma = 4$ and $T = 250$ and look at some different realizations (simulations) of $x_{1:T}$ for values of ϕ between -1 and 1 (this is the interval of ϕ where the AR(1)-process is stationary). Include a plot of at least one realization in the report. What effect does the value of ϕ have on $x_{1:T}$?

3.2 b)

Use your function from a) to simulate two AR(1)-processes, $x_{1:T}$ with $\phi = 0.3$ and $y_{1:T}$ with $\phi = 0.97$. Now, treat your simulated vectors as synthetic data, and treat the values of μ , ϕ and σ^2 as unknown parameters. Implement Stancode that samples from the posterior of the three parameters, using suitable non-informative priors of your choice. [Hint: Look at the time-series models examples in the Stan user's guide/reference manual, and note the different parameterization used here.]

- Report the posterior mean, 95% credible intervals and the number of effective posterior samples for the three inferred parameters for each of the simulated AR(1)-process. Are you able to estimate the true values?
- For each of the two data sets, evaluate the convergence of the samplers and plot the joint posterior of μ and ϕ . Comments?