Bernoulli regression model

Case study: Space shuttle Challenger disaster

Georgios P. Karagiannis @ MATH3341/4031 Bayesian statistics III/IV (practical implementation)

Back to the main document

```
rm(list=ls())
```

Aim

Students will become able to:

- produce Monte Carlo approximations of posterior quantities required for Bayesian analysis with the RJAGS R package
- implement Bayesian posterior analysis in R with RJAGS package

Students are not required to learn by heart any of the concepts discussed

---

Reading material

The material about RJAGS package is not examinable material, but it is provided for the interested student. It contains references that students can follow if they want to further explore the concepts introdced.

- Lecture notes:
  - the examples and exercises related to the Bernoulli model with conjugate prior
- Application (optional):
  - Dalal, S. R., Fowlkes, E. B., & Hoadley, B. (1989). Risk analysis of the space shuttle: Pre-Challenger prediction of failure. Journal of the American Statistical Association, 84(408), 945-957.
- References for *rjags*:
  - JAGS homepage
  - JAGS R CRAN Repository
  - JAGS Reference Manual
  - JAGS user manual
- Reference for *R*:
  - Cheat sheet with basic commands
- Reference of *rmarkdown* (optional):
  - R Markdown cheatsheet
  - R Markdown Reference Guide
  - knitr options
- Reference for *Latex* (optional):
  - Latex Cheat Sheet

New software

- R package `rjags` functions:

  - jags.model{rjags}

  - jags.samples{rjags}
  - update{rjags}

Application: Challenger O-ring

On January 28, 1986, a routine launch was anticipated for the Challenger space shuttle. Seventy-three seconds into the flight, disaster happened: the shuttle broke apart, killing all seven crew members on board. Here is the video.

The Rogers Commission report on the space shuttle Challenger accident concluded that the accident was caused by a combustion gas leak through a joint in one of the booster rockets, which was sealed by a device called an O-ring. The Challenger accident was caused by gas leak through the 6 O-ring joints of the shuttle.

The commission further concluded that 0-rings do not seal properly at low temperatures.

Dalal, Fowlkes and Hoadley (1989) looked at the number of distressed O-rings (among the 6) for 23 previous shuttle flights, and the data-set is provided below. In the table below presents data from the 23 preaccident launches of the space shuttle is used to predict 0-ring performance under the Challenger launch conditions and relate it to the catastrophic failure of the shuttle. The the data-set is provided below, where in column *Defective.O.rings*, (1) stands for presence of at least one distressed O-ring, and (0) stands for absence of any distressed O-ring; while the rest columns are self explained.

```r
# Load R package for printing
library(knitr)
library(kableExtra)
```

```r
# load the data
mydata <- read.csv("./challenger_data.csv")
# print data
## (that's a sophisticated command with fancy output, feel free to ignore it)
kable(mydata)%>%
  kable_styling(bootstrap_options = c("striped", "hover"))
```

On the night of January 27, 1986, the night before the space shuttle Challenger accident, there was a three-hour teleconference among people at Morton Thiokol, Marshall Space Flight Center, and Kennedy Space Center. The discussion focused on the forecast of a 31F temperature for launch time the next morning, and the effect of low temperature on 0-ring performance.

We are interested in finding:

- what is the limiting frequency of occuring a defective O-ring when the temperature at the platform is 31F (namely the one on the day of the accident 1/28/86 of flight 61-I)
- what are the odds of at least one defective O-rings under the conditions above?
- what is the probability that a damaged O-ring would occur in the 24th flight (date *1/28/86*)?

To answer the above we perform Bayesian analysis based on the observed data-set on the dates from *04/12/1981* to *01/12/1986*, and the variables *Damage.Incident* and *Temperature*. So ignore the variable *Leak.check.pressure*.

Model specification & posterior sampling

Let y_i denote the presence of a defective O-ring in the ith flight (0 for absence, and 1 for presence).

Let t_i denote the temperature (in F) in the platform before the ith flight.

Regarding the statistical model, we assume that y_i can be modeled as observations generated independently from a Bernoulli distribution with with parameter p_i. Here, p_i denotes the relative frequency of defective O-rings at flight i. We drop the assumption of homogeneity in the parameters!!!

As we are interesting in studing the relation of the outcome variable y: 'presence of a defective O-ring' with the input variable t: 'temperature'. Then, given the statistical model specified, it is reasonable to link the two variables through the only parameter p of the sampling distribution.

A reasonable link would be

$$p(t;\beta) = \frac{\exp(\beta_0+\beta_1 t)}{1+\exp(\beta_0+\beta_1 t)} \iff \log(\frac{p(t;\beta)}{1-p(t;\beta)}) = \beta_0+\beta_1 t$$

where $\beta=(\beta_0,\beta_1)$ because:

- it provides a bijective transformation between spaces of $p := p(t;\beta)\in[0,1]$ and $\beta_0+\beta_1 t\in\mathbb{R}$.
- it also gives a simple link between the odds $\frac{p(t;\beta)}{1-p(t;\beta)}$ and the input variable t.

Hmm...., you could use any other function; like the CDF of the Normal, Student's T, Laptace distr., etc...

Regargind the prior model, we assign a Normal prior distribution, with mean hyper-parameter b_0 and variance hyper-parameter B_0, on the unknown parameter β to account for the uncertainty about it.

Hmmmm... we could use other priors too ... I just picked one ...

The Bayesian hierarchical model under consideration is:

$$\begin{cases} y_i|\beta \sim & \text{Bernoulli}(p(t_i;\beta)), \quad \text{for, } i=1,...,n \\ p(t;\beta) = & \frac{\exp(\beta_0+\beta_1 t)}{1+\exp(\beta_0+\beta_1 t)} \\ \beta \sim & \text{N}(b_0, B_0), \end{cases}$$

with hyper-parameter values

$$b_0 = \begin{pmatrix} 0.0 \\ 0.0 \end{pmatrix}, \text{ and } B_0 = \begin{pmatrix} 100.0 & 0.0 \\ 0.0 & 100.0 \end{pmatrix}$$

Task

Write the JAGS program implementing the hierarchical model above, in order to generate a sample of size $N = 100000$ from the posterior distribution

$$\beta^{(j)} \sim \pi(\beta|y_{1:n}) \ , \quad j=1,...,N.$$

.

Analysis using the rjags package proceeds in steps:

1. Load the library `rjags`
2. Create an input script, for rjags, containing the Bayesian hierarchical model
3. Create an input list, for jags, containing the data and fixed parameters of the model

- use `list {base}`

4. Creates an object of class “jags”. To do this you need to read the model file by using the `jags.model{rjags}` function.

   - use `jags.model{rjags}`

5. Generate a posterior sample. To do this you need to extract samples from the model object using the coda.samples function.

   - use `update{rjags}` ; `coda.samples{rjags}`

... your answer

This task has already been addressed for you

step 1

Load the library

```
# Load rjags
library("rjags")
```

step 2

Create an input script, for rjags, containing the Bayesian hierarchical model

```
# Input parameters  : n, y, b_0, invB_0
# output parameters : beta
hierarhicalmodel <- "

  model {

    for ( i in 1 : n ) {

      p[ i ] <- exp(inprod(X[i,],beta)) / (1+exp(inprod(X[i,],beta)))

      y[ i ] ~ dbern( p[ i ] )
    }

    beta ~ dmnorm( b_0 , invB_0 )

  }

"
```

step 3

Create an input list, for jags, containing the data and fixed parameters of the model

```
y_obs <- mydata[ -nrow(mydata) , 4 ] # exclude the last row, and use only the 4th column
y_obs <- as.numeric(y_obs==1)        # make it numeric

n_obs <- length( y_obs )

X_obs <- cbind( rep(1,n_obs),
                as.numeric(mydata[ -nrow(mydata),3]) )

d = dim(X_obs)[2]  #
```

```
b_0 <- rep( 0.0 , d )  # prior hyper-parameter

B_0 <- diag( rep( 100 , d ) )

invB_0 <- solve( B_0 )


data.bayes <- list(y = y_obs,
                    n = n_obs,
                   X = X_obs,
                    b_0 = b_0,
                   invB_0 = invB_0)
```

step 4

Create an input list, for jags, containing the data and fixed parameters of the model

```
model.smpl <- jags.model( file = textConnection(hierarhicalmodel),
                          data = data.bayes)
```

step 5

Initialize the sampler with $N_{\text{adapt}} = 1000$ iterations.

```
adapt(object = model.smpl,
      n.iter = 10^5)
```

step 6

Generate a posterior sample of size $N = 10000$.

Use

- `jags.samples{rjags}`
    - `=jags.samples(model=, variable.names=, n.iter=, thin=)`

We can ask function `jags.samples` to generate a sample sequence of N_{total} values (in total), but return, as a sample, only every N_{thin}-th sample value from that sequence. This can be performed. This means that the function returns a sample of size $N = N_{total}/N_{thin}$. The reason will become clear soon.

We need to pay attention on two flaqs:

- the `variable.names`: it specifies the names of the random variables correspondign to the posterior samples I am interested in generating
- the `n.iter`: the size of the total sample sequence generated.
- the `thin`: the thining interval (use `thin=1` for now). It makes the algorithm to return only every `thin`-th sample value from the total sample sequence generated. **Ask your instructor about it**.

```
N = 10^5      # the size of the sample we ll gona get
n.thin = 10^3     # the thining (improving) the sample quality
n.iter = N * n.thin # the number of the total iterations performed
output = jags.samples( model = model.smpl,          # the model
                       variable.names = c("beta"),    # names of variables to be sampled
                       n.iter = n.iter,               # size of sample
                       thin = n.thin,
                       )
```

Check the names of the variables sampled

- use `names {base}`

```
names(output)
```

```
## [1] "beta"
```

Check the dimensions of each of the variables sampled

- use `dim {base}`

```
dim( output$beta )
```

```
##          iteration     chain
##        2    100000         1
```

```
# the first dimension is the numbers of columns of the variable
# the second dimention is the size of the sample drawn
# the thirs dimention is the number of the sub-samples drawn (in our case it is just 1)
```

Copy the sample of each variable in a vector with a more friendly name…

```
beta.smpl <-output$beta
```

Task

Extract the sample drawn from the posterior distribution

$$\beta^{(j)} \sim \pi(\beta|y_{1:n}) \quad j = 1, ..., N$$

and print the trace plot of the sample.

Use functions:

- `plot {graphics}`.

Snapshot from Tem 2:

- A good quality sample for the purposes of Monte Carlo integration is the one whose trace plot looks completely uncorelated.
  - e.g. like the independence assumption in the siple linear regression in SC2.
- To improve the quality of the sample, you can go back to the sampling stem and play with the flaq values of `n.iter` and `thin` in `jags.samples{rjags}`.

… your answer

Below try to address the task by creating R chunks, and running them, and adding comments to the document.

```
# write your own R code
```

Parameteric posterior analysis of β_1

Regarding the parameter β_1, we can calculate that

$$\beta_1 = \frac{\log\left(\frac{p(t')}{1-p(t')}\right) - \log\left(\frac{p(t;\beta)}{1-p(t;\beta)}\right)}{t'-t}$$

by properly rearanging the the link function $\log(\frac{p}{1-p}) = \beta_0 + \beta_1 t$ considered.

Hence the parameter β_1 can be interpreted as:

- the rate of change of the odds of a defective O-ring (in log scale) with respect to the temperature
  - this is can be seen as $t' - t \to 0$
- the change of the odds of a defective O-ring (in log scale) if the temperature increase for 1 unit
  - this can be seen as $t' - t = 1$

So:

- $\beta_1 > 0$ means that $t \uparrow \implies p \uparrow$ and $t \downarrow \implies p \downarrow$
- $\beta_1 < 0$ means that $t \downarrow \implies p \uparrow$ and $t \uparrow \implies p \downarrow$

Task

Compute the MC approximation of the posterior probability density function, and the cumulative distribution function of β_1 as

... your answer

Below try to address the task by creating R chunks, and running them, and adding comments to the document.

```
# write your own R code
```

Task

Compute the MC approximate of the posterior expected value of β_1 as

$$\mathrm{E}_{\pi}(\beta_1|y_{1:n}) \approx \frac{1}{N}\sum_{j=1}^{N}\beta_1^{(j)}$$

... your answer

Below try to address the task by creating R chunks, and running them, and adding comments to the document.

```
# write your own R code
```

Task

Compute the MC approximate of the posterior standard deviation of β_1 as

$$\begin{aligned}\mathrm{SD}_{\pi}(\beta_1|y_{1:n}) =&\sqrt{\mathrm{E}_{\pi}(\beta_1^2|y_{1:n}) - (\mathrm{E}_{\pi}(\beta_1|y_{1:n}))^2}\\ \approx&\sqrt{\frac{1}{N}\sum_{j=1}^{N}\left(\beta_1^{(j)}\right)^2 - \left(\frac{1}{N}\sum_{j=1}^{N}\beta_1^{(j)}\right)^2}\\ =&s_{\beta_1}^2\end{aligned}$$

. . . your answer

Below try to address the task by creating R chunks, and running them, and adding comments to the document.

```
# write your own R code
```

Task

Compute the MC approximate of the posterior probability that the rate β_1 under consideration is negative;

$$\begin{aligned}\Pr_{\pi}(\beta_1 < 0|y_{1:n}) =&\mathrm{E}_{\pi}(1(\beta_1 < 0)|y_{1:n})\\ \approx&\frac{1}{N}\sum_{j=1}^{N}1(\beta_1^{(j)} < 0)\end{aligned}$$

. . . your answer

Below try to address the task by creating R chunks, and running them, and adding comments to the document.

```
# write your own R code
```

Task

Compute the MC approximate posterior 95% credible interval (equal tail) of β_1,

$$[Q_{0.025}(\beta_1|y_{1:n})\ ,\ Q_{0.975}(\beta_1|y_{1:n})]$$

where

$$Q_{\alpha}(\beta_1|y_{1:n}) = F_{\beta_1}^{-1}(\alpha|y_{1:n})$$

is the α-th quantile of the posterior distribution of β_1

. . . your answer

Below try to address the task by creating R chunks, and running them, and adding comments to the document.

```
# write your own R code
```

Posterior analysis of frequency parameter $p(t;\beta)$

Regarding the parameter $p(t;\beta)$ which is a function of the temperature t, we can calculate that

$$p(t;\beta) = \frac{\exp(\beta_0 + \beta_1 t)}{1 + \exp(\beta_0 + \beta_1 t)}$$

by properly rearanging the the link function $\log(\frac{p}{1-p}) = \beta_0 + \beta_1 t$.

Task

Compute and ploit MC approximate of the 95% equal length credible interval of the parameter $p(t;\beta)$ for themperature values $t \in (30, 82)$.

Namely, compute:

- the posterior expected value of $p(t;\beta)$ w.r.t t as

  $$\mathrm{E}_{\pi}(p(t;\beta)|y_{1:n}) \approx \frac{1}{N}\sum_{j=1}^{N} p(t;\beta^{(j)})$$

  at t.

- the 95% credible interval (equal tail) of $p(t;\beta)$ w.r.t t as

  $$[Q_{0.025}(p(t;\beta)|y_{1:n};t)\ ,\ Q_{0.925}(p(t;\beta)|y_{1:n};t)]$$

  where

  $$Q_{\alpha}(p(t;\beta)|y_{1:n};t) = F^{-1}_{p(t;\beta)}(\alpha|y_{1:n};t)$$

  is the α-th quantile of the posterior distribution of $p(t;\beta)$ at t

and plot them against the temperature $t \in (30, 82)$ in the same plot.

Can we infer anything about flight 61-I on 1/28/86.

... your answer

Below try to address the task by creating R chunks, and running them, and adding comments to the document.

```
# write your own R code
```

Posterior predictive analysis of the outcome of flight 61-I on 1/28/86.

The predictive distribution mass function of $y_{n+1}|y_{1:n}$, is

$$f_{\pi}(y_{n+1}=c|y_{1:n};t)=\int f(y_{n+1}=c|\beta;t)\pi(\beta|y_{1:n})\mathrm{d}\beta,\quad c\in\{0,1\}$$

$$=\mathrm{E}_{\pi}(f(y_{n+1}=c|\beta;t)|y_{1:n}),\quad c\in\{0,1\}$$

$$=\mathrm{E}_{\pi}(p(t;\beta)^{c}(1-p(t;\beta))^{1-c}|y_{1:n}),\quad c\in\{0,1\}$$

$$=\begin{cases}1-\mathrm{E}_{\pi}(p(t;\beta)|y_{1:n}) & \text{, c}=0\\ \mathrm{E}_{\pi}(p(t;\beta)|y_{1:n}) & \text{, c}=1\end{cases}$$

at temperature t.

Task

Well, this means that we have already computed and ploted the MC approximation for the predictive distribution mass function of $y_{n+1}|y_{1:n}$ for at $t\in(30,82)$.

So now just ...

Compute and plot the MC approximate of the predictive probability that a failed O-ring will occure at temperature $t=31$F, as

$$f(y_{n+1}=c|y_{1:n};t=31)\approx\begin{cases}1-\frac{1}{N}\sum_{j=1}^{N}p(t=31;\beta^{(j)}) & \text{, c}=0\\ \frac{1}{N}\sum_{j=1}^{N}p(t=31;\beta^{(j)}) & \text{, c}=1\end{cases}$$

... your answer

Below try to address the task by creating R chunks, and running them, and adding comments to the document.

```
# write your own R code
```

Conclusions and discussion.

task

Now, it is January 27, 1986, and you take part in the 3-hour teleconference with people from Morton Thiokol, Marshall space flight center, and Kennedy space center.

The forcast says that tommorrow the temperature will be 31F at lunch time. This is too frosty... and rare temperature for the area.

Would you cancel the lunch of the Space Shuttle 61-I on 1/28/86?

. . . your answer

Below try to address the task by creating R chunks, and running them, and adding comments to the document.