Metropolis Algorithm

```
# load libraries
library(tidyverse)
```

```
-- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
v dplyr     1.1.3     v readr     2.1.4
v forcats   1.0.0     v stringr   1.5.0
v ggplot2   3.4.4     v tibble    3.2.1
v lubridate 1.9.3     v tidyr     1.3.0
v purrr     1.0.2
-- Conflicts ------------------------------------------ tidyverse_conflicts() --
x dplyr::filter() masks stats::filter()
x dplyr::lag()    masks stats::lag()
i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to beco
```

```
library(mvtnorm)
library(coda)
```

```
yX = structure(c(3, 1, 1, 2, 0, 0, 6, 3, 4, 2, 1, 6, 2, 3, 3, 4, 7,
2, 2, 1, 1, 3, 5, 5, 0, 2, 1, 2, 6, 6, 2, 2, 0, 2, 4, 1, 2, 5,
1, 2, 1, 0, 0, 2, 4, 2, 2, 2, 2, 0, 3, 2, 1, 1, 1, 1, 1, 1, 1,
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, 1, 1, 3, 3, 1, 1, 1, 1, 1, 1, 1, 1, 2, 2, 2, 2, 2, 2, 2, 2,
2, 5, 5, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 5, 4, 4, 4, 4, 5,
5, 5, 5, 3, 3, 3, 3, 3, 3, 3, 6, 1, 1, 9, 9, 1, 1, 1, 1, 1, 1,
1, 1, 4, 4, 4, 4, 4, 4, 4, 4, 4, 25, 25, 16, 16, 16, 16, 16,
16, 16, 16, 16, 16, 16, 16, 25, 16, 16, 16, 16, 25, 25, 25, 25,
9, 9, 9, 9, 9, 9, 9, 36, 1, 1), .Dim = c(52L, 4L), .Dimnames = list(
    NULL, c("fledged", "intercept", "age", "age2")))
```

```r
yX %>%
  head(n = 5)
```

```
     fledged intercept age age2
[1,]       3         1   3    9
[2,]       1         1   3    9
[3,]       1         1   1    1
[4,]       2         1   1    1
[5,]       0         1   1    1
```

```r
y = yX[,1]
X = yX[,-1]
```

```r
set.seed(360)
n = length(y)
p = ncol(X)

pmn.beta = rep(0, p) # prior mean beta
psd.beta = rep(10, p) # prior sd beta

var.prop = var(log(y + 1/2)) * solve(t(X) %*% X) # proposal variance
S = 10000
beta = rep(0, p); acs = 0
BETA = matrix(0, nrow = S, ncol = p)
set.seed(1)

for (s in 1:S) {
  beta.p = t(rmvnorm(1, beta, var.prop))

  lhr = sum(dpois(y, exp(X %*%beta.p), log = TRUE)) -
    sum(dpois(y, exp(X %*% beta), log = TRUE)) +
    sum(dnorm(beta.p, pmn.beta, psd.beta, log = TRUE)) -
    sum(dnorm(beta, pmn.beta, psd.beta, log = TRUE))

  if (log(runif(1)) < lhr) {
    beta = beta.p ; acs = acs + 1
  }

  BETA[s,] = beta
}
```

```r
BETA %>%
  apply(2, effectiveSize)
```

```
[1] 867.4750 825.6214 692.0495
```