STAT 656 HW 2

Satoshi Ido (ID: 34788706)

1 October 2023

```
library("ggplot2")
library("dplyr")
library("tidyr")
library("MASS")
library("fs")
library("moments")
library("rstan")
library("bayesplot")
library("StanHeaders")
```

set options to speed up the calculations

```
# to avoid recompilation of unchanged Stan programs, we recommend calling
rstan_options(auto_write = TRUE)
# for execution on a local, multicore CPU with excess RAM we recommend calling
options(mc.cores = parallel::detectCores())
# set the size of the plots
options(repr.plot.width = 12, repr.plot.height = 6)

# options(bayesplot::theme_default())
bayesplot_theme_set(theme_default(base_size = 24, base_family = "sans"))
```

```
# Create the input_dir (input directory)
current_note_path <- getwd()
INPUT_DIR <- file.path(current_note_path, "656/hw/hw2/data")

# If INPUT_DIR has not been created yet, create it
if (!dir.exists(INPUT_DIR)) {
  dir.create(INPUT_DIR)
}

# Create the output_dir (output directory)
OUTPUT_DIR <- file.path(current_note_path, "656/hw/hw2/outputs")

# If OUTPUT_DIR has not been created yet, create it
if (!dir.exists(OUTPUT_DIR)) {
  dir.create(OUTPUT_DIR)
}

# Read CSV files using a function to specify the directory automatically
read_csv <- function(name, ...) {
```

```
  path <- file.path(INPUT_DIR, paste0(name, ".csv"))
  print(paste("Load:", path))
  return(read.csv(path, ...))
}
```

Synthetic data

The file 'hw2 synthetic.csv' is a dataset of count-valued measurements $y = \{y_1, \ldots, y_n\}$, with $y_i \in \{0, 1, \ldots\}$. Each output y_i has an associated $x_i = (x_{i,1}, x_{i,2}) \in \mathbb{R}^2$, and write $x = \{x_1, \ldots, x_n\}'$ as x. We model y_i as

$$y_i|\beta \sim \textbf{Poisson}(e^{f(x_i,\beta)})$$

Here, the exponential is to ensure the Poisson rate is always positive, and the function $f(x_i, \beta) = \beta_0 + \beta_1 x_{i,1} + \beta_2 x_{i,2} + \beta_3 x_{i,1}^2 + \beta_4 x_{i,2}^2 + \beta_5 x_{i,1} x_{i,2}$.

1. **With the provided data, perform a Bayesian analysis on the parameters of the model above to decide which terms in the expression for f(x) you think are important. State clearly what your prior over beta is, and how you arrived at your conclusion, including any useful figures (especially of the posterior distribution). You can use Stan.**State clearly what your prior over beta is, and how you arrived at your conclusion, including any useful figures (especially of the posterior distribution). You can use Stan.}

```
df <- read.csv("/Users/satoshiido/Documents/statistical-analysis/656/hw/hw2/data/hw2_synthetic.csv")
head(df)
```

```
##           x1          x2 y
## 1 -0.6264538 -0.62036668 0
## 2  0.1836433  0.04211587 0
## 3 -0.8356286 -0.91092165 2
## 4  1.5952808  0.15802877 4
## 5  0.3295078 -0.65458464 3
## 6 -0.8204684  1.76728727 0
```

Summary of data

```
summary(df)
```

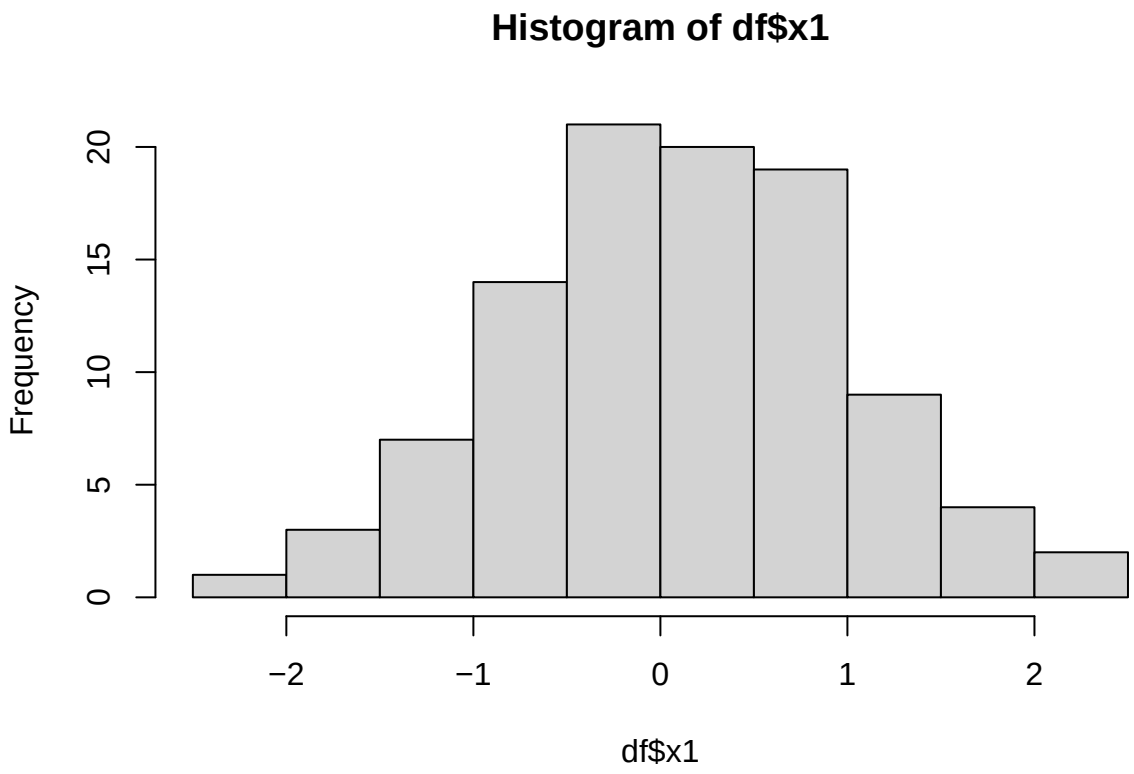
```
##        x1                 x2                  y
##  Min.   :-2.2147   Min.   :-1.91436   Min.   : 0.00
##  1st Qu.:-0.4942   1st Qu.:-0.65105   1st Qu.: 0.00
##  Median : 0.1139   Median :-0.17722   Median : 1.00
##  Mean   : 0.1089   Mean   :-0.03781   Mean   : 1.29
##  3rd Qu.: 0.6915   3rd Qu.: 0.50090   3rd Qu.: 2.00
##  Max.   : 2.4016   Max.   : 2.30798   Max.   :10.00
```

```
paste0("sd of x1:", sd(df$x1), " sd of x2: ", sd(df$x2))
```

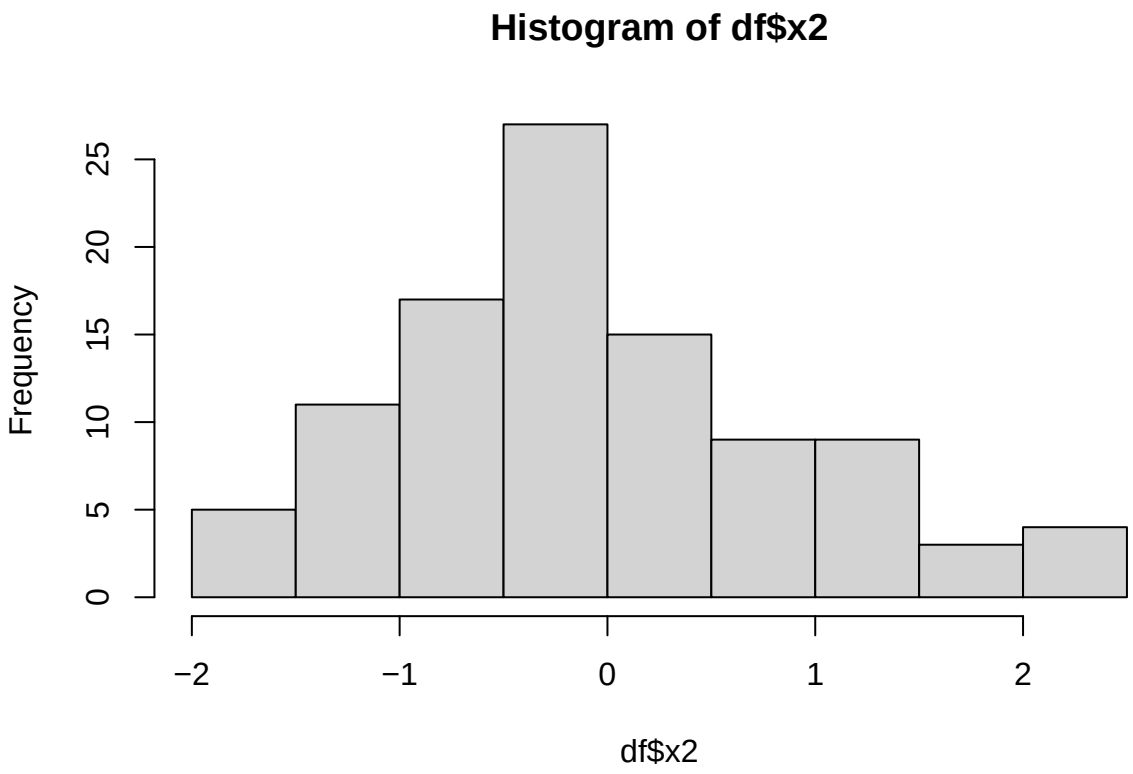
```
## [1] "sd of x1:0.898199359660041 sd of x2: 0.957879069588428"
```

Histogram of data

```
hist(df$x1)
```




```
hist(df$x2)
```



Data Creation for Stan coding

```
# create the data for poisson regression model
df$x1_sq <- df$x1^2
df$x2_sq <- df$x2^2
df$x1_x2 <- df$x1 * df$x2
df$offset <- 1
```

Stan coding

Model Specification with Stan

```
poissonreg_normal_code = "
// Poisson model with normal prior for beta

// Data are things you observe/condition on
data {
  // number of data items
  int<lower=1> N;
  // Number of beta parameters (predictors)
  int<lower=1> p;
  // std dev of the prior
  real<lower=0> pr_sd;
```

```
  // matrix of predictors
  matrix[N, p] x;

  // offset
  // real offset[N];

  // count outcome (output vector)
  int<lower=0> y[N];
}

parameters {
  // Parameters to estimate
  vector[p] beta;
}

// useful to avoid repeating calculations
// note that stan will return values of these variables for each MCMC sample
transformed parameters {
  // Linear predictor
  vector[N] mu = exp(x * beta);
}

// The actual Bayesian model goes here
// I set normal dist as a prior for beta
model {
  // priors
  // Note: beta is p-dim
  beta ~ normal(0, pr_sd);

  // likelihood
  y ~ poisson(mu);
}

// Generate quantities of interest (e.g. posterior predictions)
generated quantities {
  int<lower=0> y_rep[N];

  for (i in 1:N) {
    y_rep[i] = poisson_rng(mu[i]);
  }
}
"

# build the model before sampling
poissonreg_normal <- stan_model(model_code = poissonreg_normal_code)
```

data preparation for stan simulation

```
# create the data for stan simulation
X <- df[, c("x1", "x2", "x1_sq", "x2_sq", "x1_x2", "offset")]
y <- df$y
```

compile the model

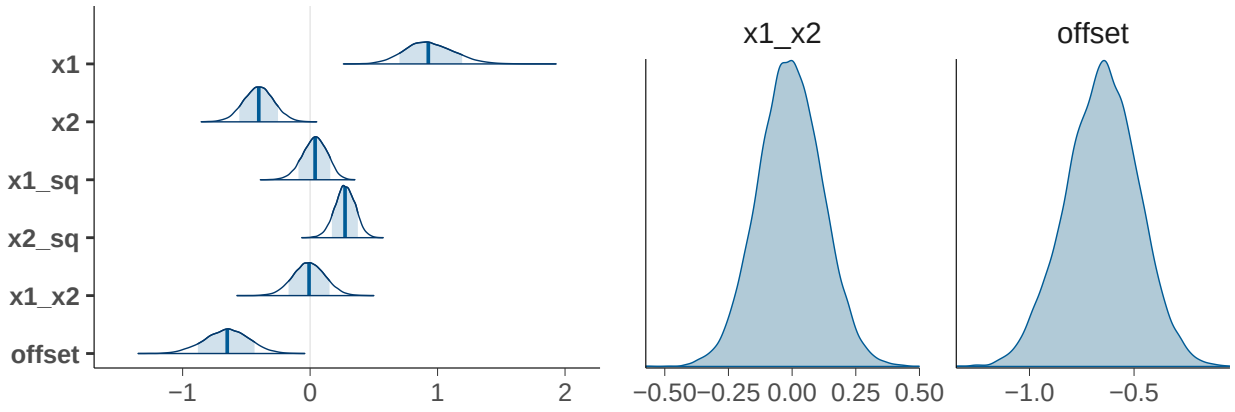
Regression model: $y = \{y_1, \ldots, y_n\}$, with $y_i \in \{0, 1, \ldots\}$. Each output y_i has an associated $x_i = (x_{i,1}, x_{i,2}) \in \mathbb{R}^2$, and write $x = \{x_1, \ldots, x_n\}'$ as x. We model y_i as

$$y_i|\beta \sim \text{Poisson}(e^{f(x_i,\beta)})$$

```
poissonreg_data <- list(N = nrow(X), p = ncol(X), pr_sd = 100, x = X, y = y)
nfit <- rstan::sampling(
  object = poissonreg_normal,
  data = poissonreg_data,
  iter = 10000,
  warmup = 2000,
  chains = 2
)
```

```
# save the model
saveRDS(nfit, file = "/Users/satoshiido/Documents/statistical-analysis/656/hw/hw2/outputs/nfit.rds")
# read the model (-> In this way, no need to run the model again)
# nfit <- readRDS(file = file.path(OUTPUT_DIR, "nfit.rds"))
```

```
post_smp <- as.data.frame(nfit)[, c(1, 2, 3, 4, 5, 6)]
colnames(post_smp) <- colnames(X)
mcmc_areas(post_smp, pars = colnames(X), prob = 0.8)
mcmc_dens(post_smp[, 5:6], alpha = 0.5)
```



Applied problem

first design selection

My first approach is to run the eperiments with 24 wells initially, followed by 48 wells in the second experiments. The reason for this is that I want to see if the 24 wells are enough to get the information about the mean and variance of the population so that I can set a reasonable prior for the second experiment. I will assign pairs of concentrations for the two chemical modulators to each well in the manner as below. The main focus is to see how the effect of the one modulator changes depending on the concentration of the other modulator. Since I have little idea of the effect of the modulators, I will assign the concentrations of the modulators somewhat randomly.

```r
design1 <- matrix(nrow = 24, ncol = 2)
# assign the concentration of the modulators randomly
## set a seed to reproduce the same result
set.seed(49)
modA <- rep(seq(0, 75, by = 15), each = 4)
modB <- rep(seq(0, 30, by = 10), times = 6)
design1 <- cbind(modA, modB)
output_path <- file.path(OUTPUT_DIR, "design1.csv")
# write a table and save it to csv
# write.table(design1, file = output_path, sep = ",", col.names = F, row.names = F)
```