**STAT 447C: Bayesian Statistics** **Fall 2024**

# Exercise 4

*Junsong Tang* junsong.tang@stat.ubc.ca

## 1 logistic rocket improvement

1.

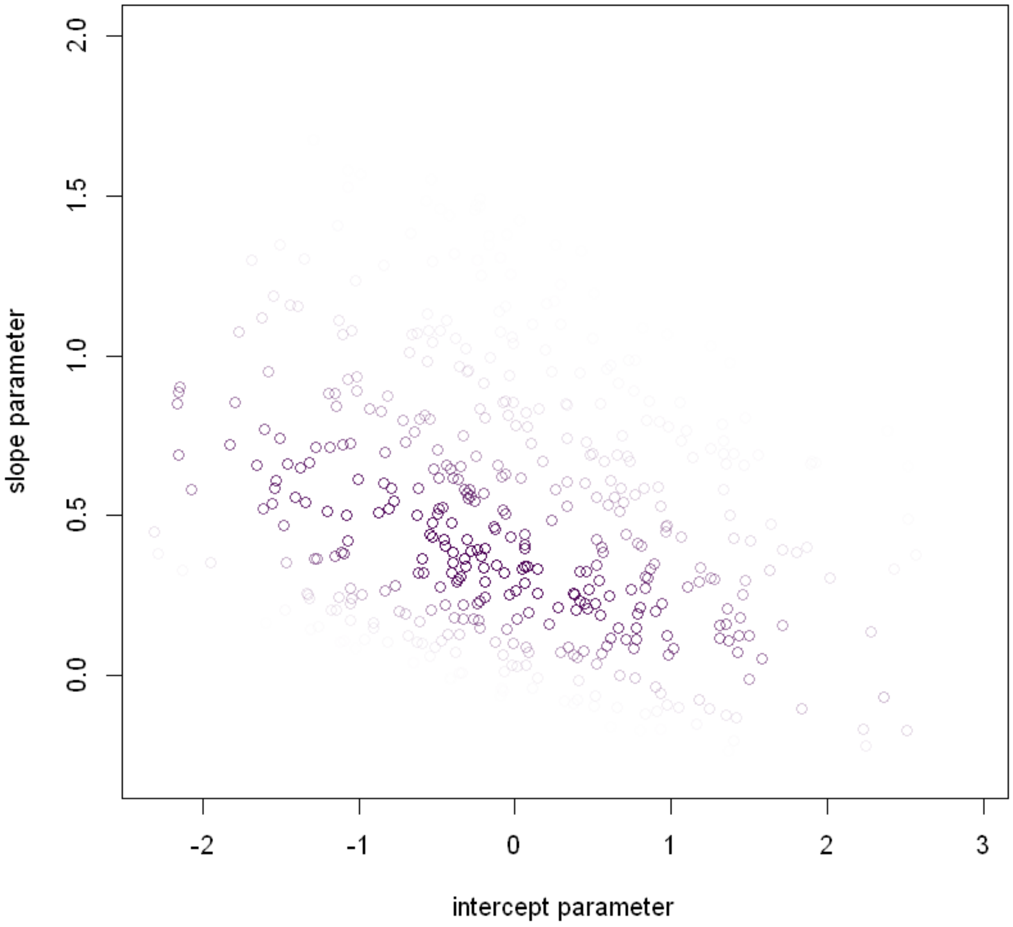
```r
suppressPackageStartupMessages(library(extraDistr))
suppressPackageStartupMessages(library(distr))
source("./simple.R")
source("./simple_utils.R")
set.seed(2024)

success_indicators = c(1, 0, 1, 1, 0, 1, 1, 1, 1, 1, 1)

# (1)
logistic_regression = function() {
  n = length(success_indicators)
  intercept = simulate(Norm(0, 1))
  slope = simulate(Norm(0, 1))
  thetas = plogis(intercept + slope * (1:n))
  for (i in (1:n)) {
    observe(success_indicators[i], Bern(thetas[i]))
  }
  next_theta = intercept + slope * (n + 1)
  pred = ifelse(next_theta >= 1 / 2, 1, 0)
  return (c(intercept, slope, pred))
}
```

2.

```r
# (2)
posterior = posterior_particles(logistic_regression, 1000)
weighted_scatter_plot(posterior, plot_options = list(xlab="intercept parameter", ylab="slope parameter"))
```



3.
```
# (3)
post_obj = posterior(logistic_regression, 1000)
intercept = post_obj[1]
slope = post_obj[2]
plogis(slope * (length(success_indicators) + 1) + intercept)
# 0.990931632727317
```

4.
```
# (4)
logistic_regression_2 = function() {
  n = length(success_indicators)
  intercept = simulate(Norm(0, 1))
  theta = plogis(intercept)
  for (i in (1:n)) {
    observe(success_indicators[i], Bern(theta))
  }
  pred = ifelse(theta >= 1 / 2, 1, 0)
  return (c(intercept, pred))
}
plogis((posterior(logistic_regression_2, 1000))[1])
# 0.733947433558989
```

## 2 choosing a model

```r
unified_model = function() {
  post_logis = logistic_regression()
  post_simple = logistic_regression_2()
  p_logis = plogis(post_logis[1] + post_logis[2] * length(success_indicators))
  p_simple = plogis(post_simple[1])
  return (ifelse(p_logis > p_simple, 1, 0))
}
posterior(unified_model, 1000)
# 0.955874166557784
```