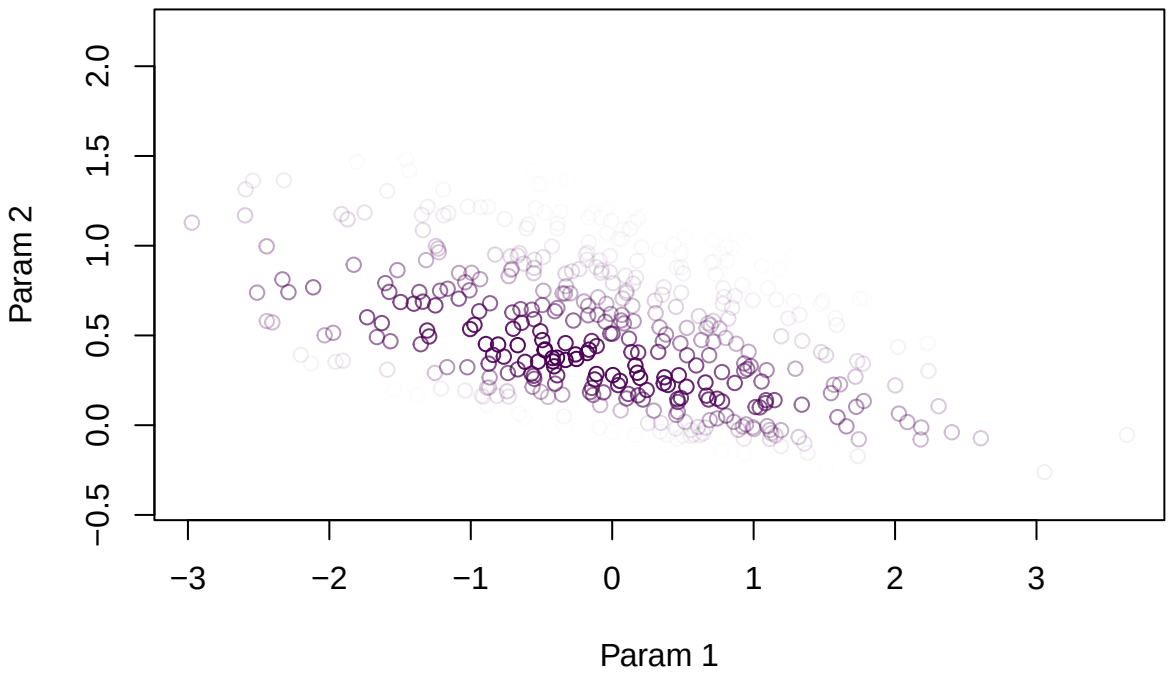
```r
source("../../simple.R")
source("../../simple_utils.R")
set.seed(1)

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

logistic_regression = function() {
  n = length(success_indicators)
  xs = 1:n

  intercept = simulate(Norm(0, 1))
  slope = simulate(Norm(0, 1))

  for (i in 1:n) {
    prob = plogis(intercept + slope * xs[i])
    observe(success_indicators[i], Bern(prob))
  }

  success = plogis(intercept + slope * (n + 1))

  return(c(intercept, slope, success))
}

logistic_regression()
```

```
## [1] -0.6264538  0.1836433  0.8288170
```

```r
set.seed(1)

posterior_samples = posterior_particles(logistic_regression, 1000)

weighted_scatter_plot(posterior_samples)
```




```
set.seed(1)

nexts = length(success_indicators) + 1

probs = plogis(posterior_samples$samples[,1] + posterior_samples$samples[,2] * nexts)

sum(probs * posterior_samples$weights) / sum(posterior_samples$weights)
```

```
## [1] 0.9517229
```

```
set.seed(1)
logistic_regression1 = function() {
  n = length(success_indicators)
  xs = 1:n

  intercept = simulate(Norm(0, 1))

  for (i in 1:n) {
    prob = plogis(intercept)
    observe(success_indicators[i], Bern(prob))
  }

  success = plogis(intercept)

  return(c(intercept, success))
```

```
}

posterior_samples = posterior_particles(logistic_regression1, 1000)

nexts = length(success_indicators) + 1

probs = plogis(posterior_samples$samples[,1])

sum(probs * posterior_samples$weights) / sum(posterior_samples$weights)
```

```
## [1] 0.7277376
```

```
set.seed(1)
logistic_regression2 = function() {
  n = length(success_indicators)
  xs = 1:n

  a = simulate(Bern(0.5))

  intercept = simulate(Norm(0, 1))
  if (a == 0) {
    slope = 0
  } else {
    slope = simulate(Norm(0, 1))
  }

  for (i in 1:n) {
    prob = plogis(intercept + slope * xs[i])
    observe(success_indicators[i], Bern(prob))
  }

  success = plogis(intercept + slope * (n + 1))

  return(c(intercept, slope, success))
}

posterior_samples = posterior_particles(logistic_regression2, 1000)

nexts = length(success_indicators) + 1

probs = plogis(posterior_samples$samples[,1] + posterior_samples$samples[,2] * nexts)

sum(probs * posterior_samples$weights) / sum(posterior_samples$weights)
```

```
## [1] 0.8568196
```