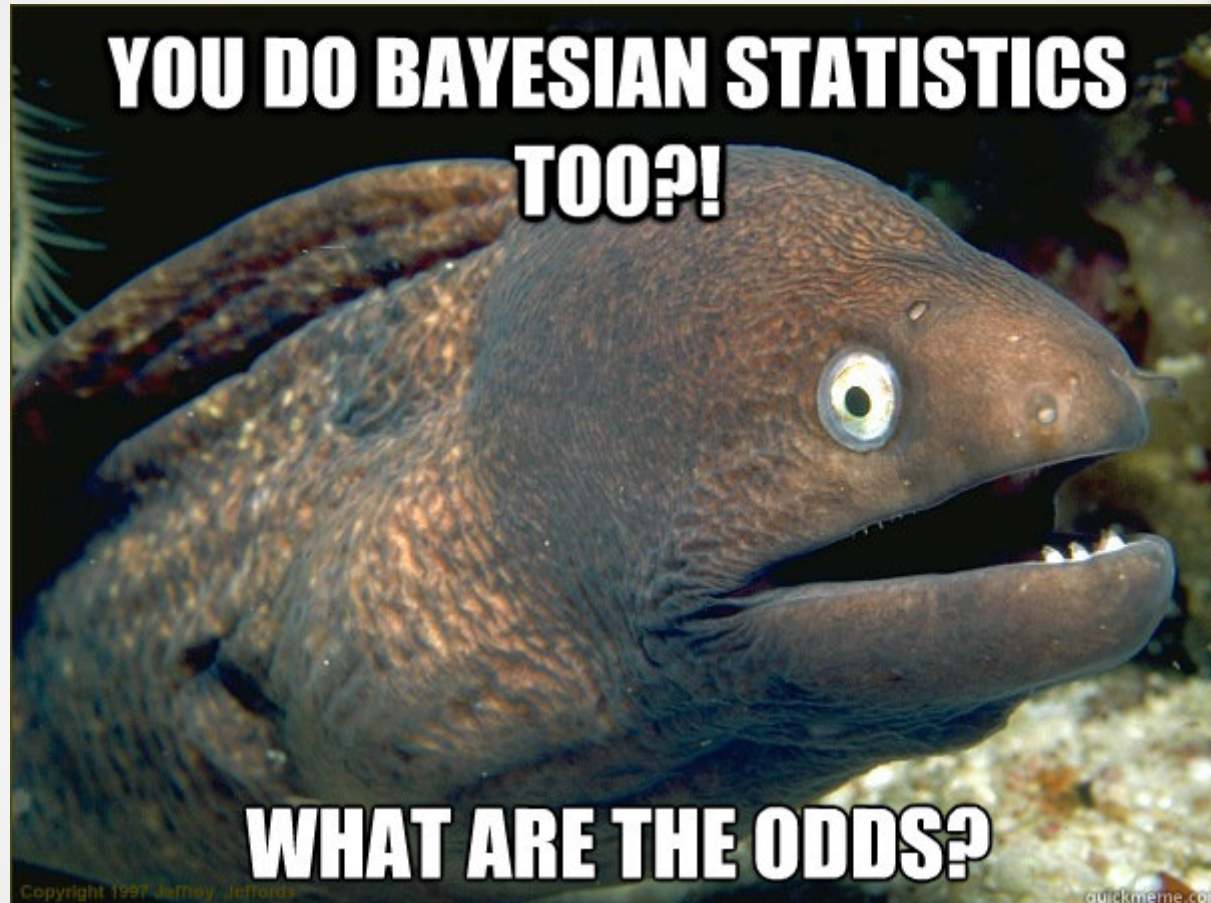# STATS 331

Introduction to Bayesian Statistics
Semester 2, 2016

Today

Today we will look at a Bayesian equivalent of the classical t-test.

We'll look at **3** different models. The models will have the same likelihood, but different priors.

A t-test situation
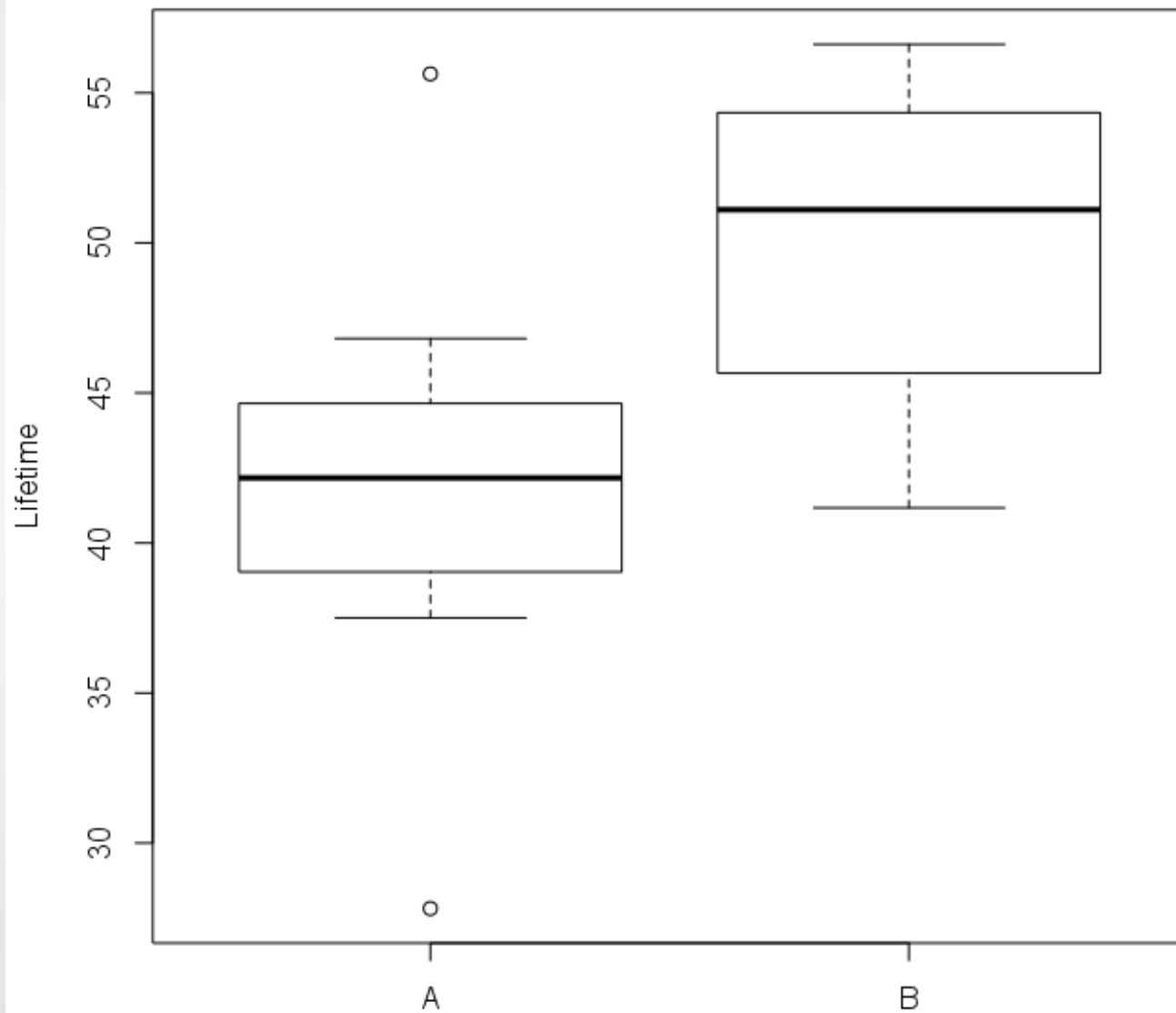
* Manufacturer A's widgets:

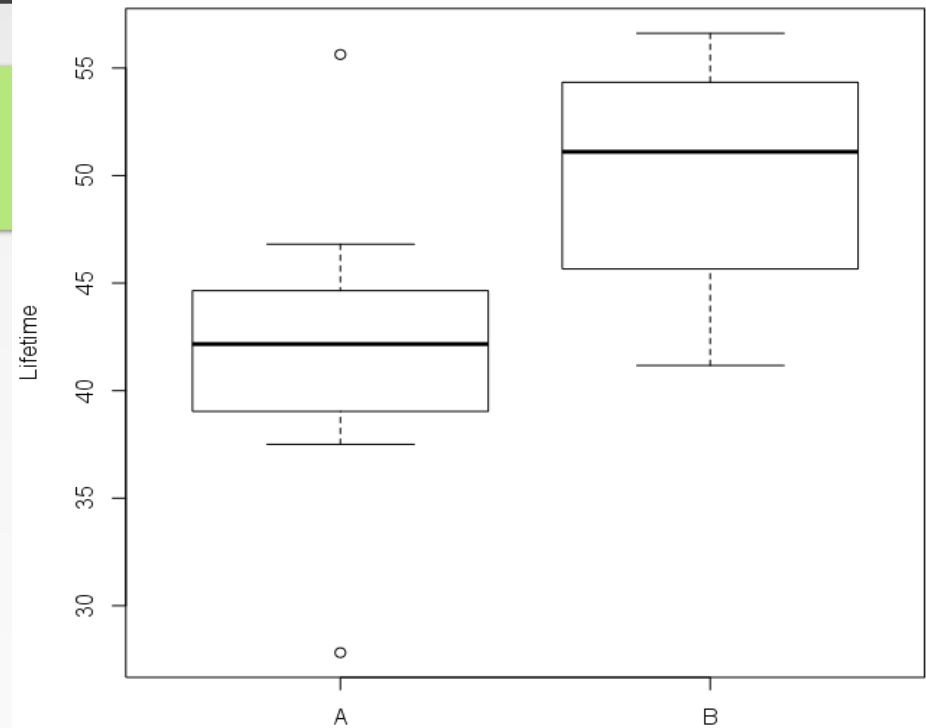Mean lifetime = 42 hours, SD = 7.48 hours based on N=9

* Manufacturer B's widgets:

Mean lifetime = 50 hours, SD = 6.48 hours based on N=4.

Is one manufacturer better than the other, on average?

Inspect Data

What do you think?

a) Manufacturer A is better, on average

b) Manufacturer B is better, on average

c) Some evidence that B is better, but not overwhelming

c) They are probably the same

Classical T-Test

$$H_0: \quad \mu_1 = \mu_2$$
$$H_1: \quad \mu_1 \neq \mu_2$$

Classical T-Test

```
> t.test(data$x1, data$x2)


Welch Two Sample t-test


data:   data$x1 and data$x2

t = -1.9568, df = 6.721, p-value = 0.09296

alternative hypothesis: true difference in means is not equal to 0

95 percent confidence interval:

 -17.749161    1.749161

sample estimates:

mean of x mean of y

      42           50
```

Meaning of the p-value?

The p-value is the probability, assuming H_0 is true, that the data would show this amount of difference, or more.

The posterior probability is the probability that H_0 is true, given the data. That's what we want!

Assumptions of the classical t-test

* Equal variance for the two samples

* Normally distributed samples

* Hints for how to construct likelihood!

Formulating a Bayesian Version

Try to think of it as **parameter estimation**

Estimate μ_1 and μ_2 from the data

(i.e. get the posterior distribution)

Then, calculate the posterior probability they are equal (if that's what you're interested in)

The Sampling Distribution / Likelihood

$$x_{1,i} \sim \mathcal{N}(\mu_1, \sigma^2)$$ Data values in group 1

$$x_{2,i} \sim \mathcal{N}(\mu_2, \sigma^2)$$ Data values in group 2

Likelihood in JAGS

```
for(i in 1:N1)
{
  x1[i] ~ dnorm(mu1, 1/sigma^2)
}
for(i in 1:N2)
{
  x2[i] ~ dnorm(mu2, 1/sigma^2)
}
```
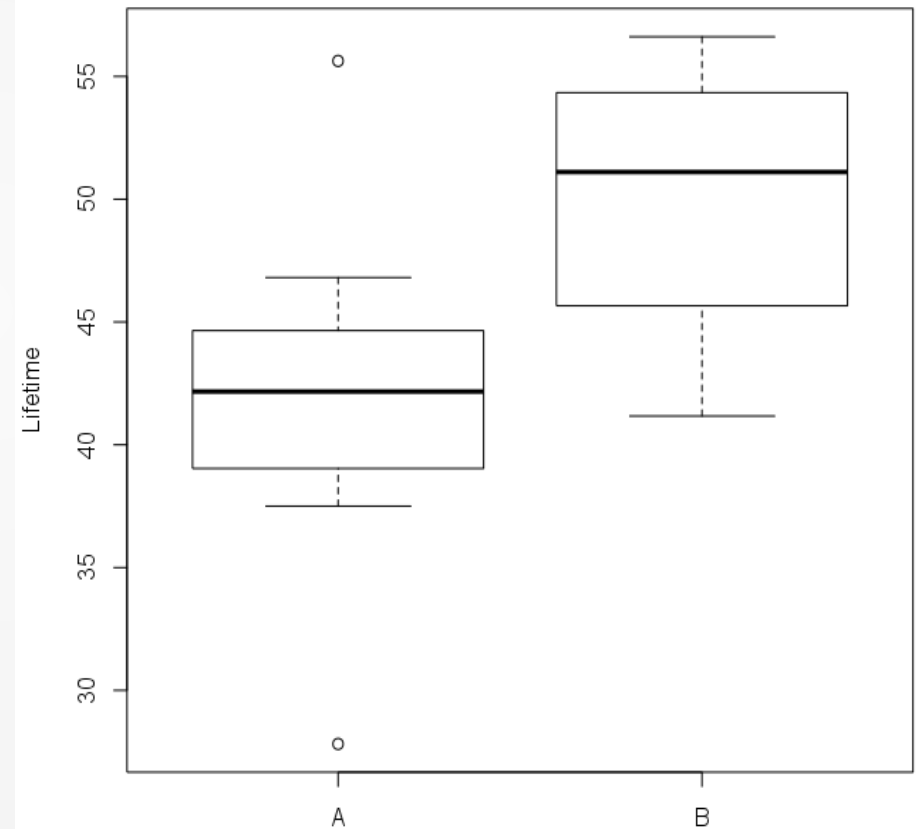
Model 1 – Priors

```
# Prior for the two means
mu1 ~ dnorm(0, 1/1000^2)
mu2 ~ dnorm(0, 1/1000^2)

# Prior for the standard deviation
log_sigma ~ dunif(-10, 10)
sigma <-  exp(log_sigma)
```

Results from JAGS

```
> mean(results$mu2 > results$mu1)

[1] 0.9421

> mean(results$mu2 == results$mu1)

[1] 0
```
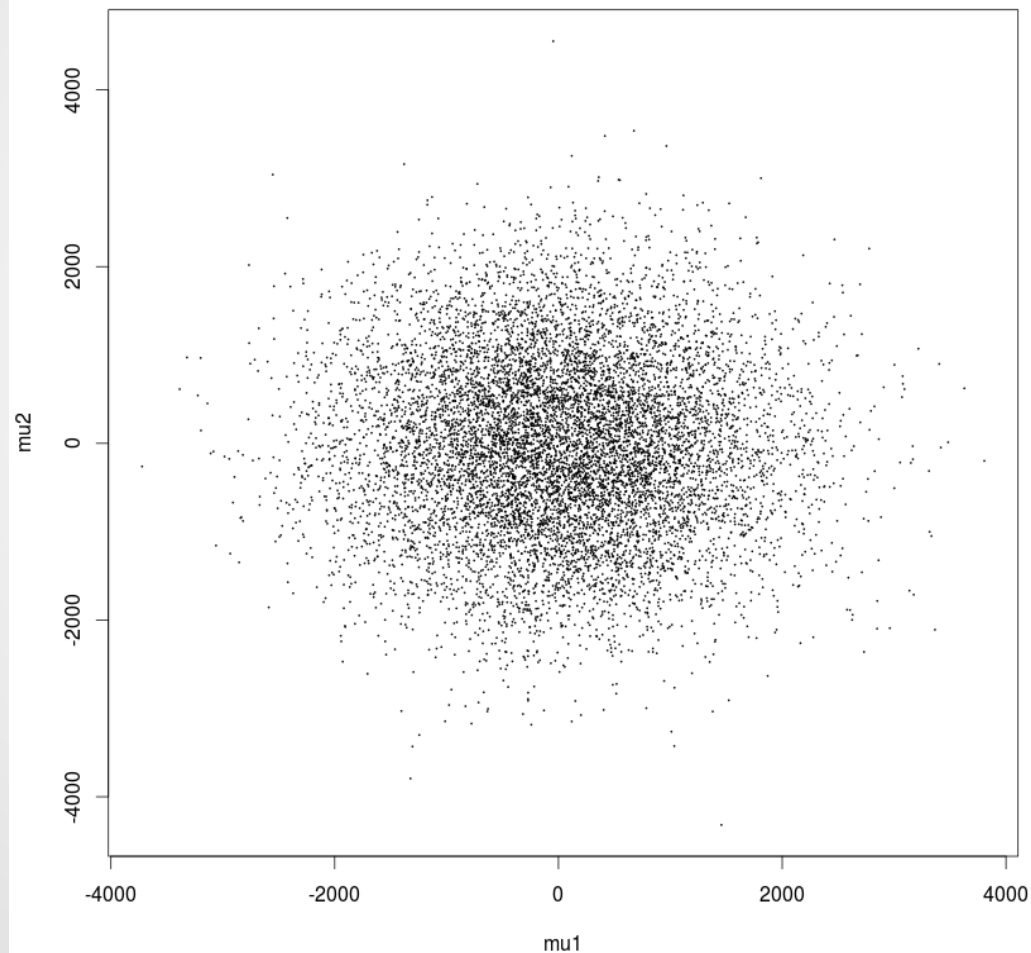
Are the parameters equal or not?

Zero posterior probability that $\boldsymbol{\mu_1} = \boldsymbol{\mu_2}$

That's because it had zero prior probability!

Model 1 may be useful, but not for answering the question "are the means equal". It assumes they aren't.

Model 1's Prior

Hardly any prior probability for the hypothesis that the two parameters might have similar values!

Model 2: A "Testing Prior" in JAGS

- Let's construct a prior that assigns 50% prior probability to the hypothesis $\mu_1 = \mu_2$. How?

  - Leave the μ_1 prior the same

  - Make a prior for **difference** which has a 50% probability of being zero

  - Define μ_2 as μ_1 + **difference**

Prior for Difference

Away from zero, the shape is exponential.

"If the difference is not precisely zero, it's probably small."
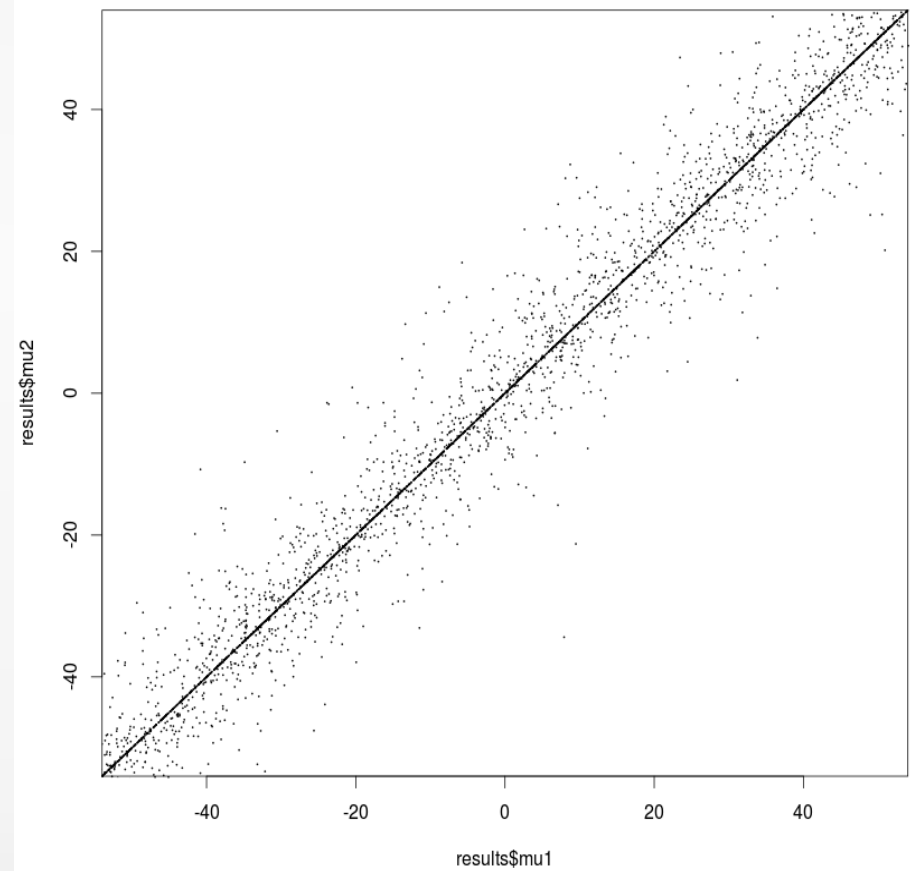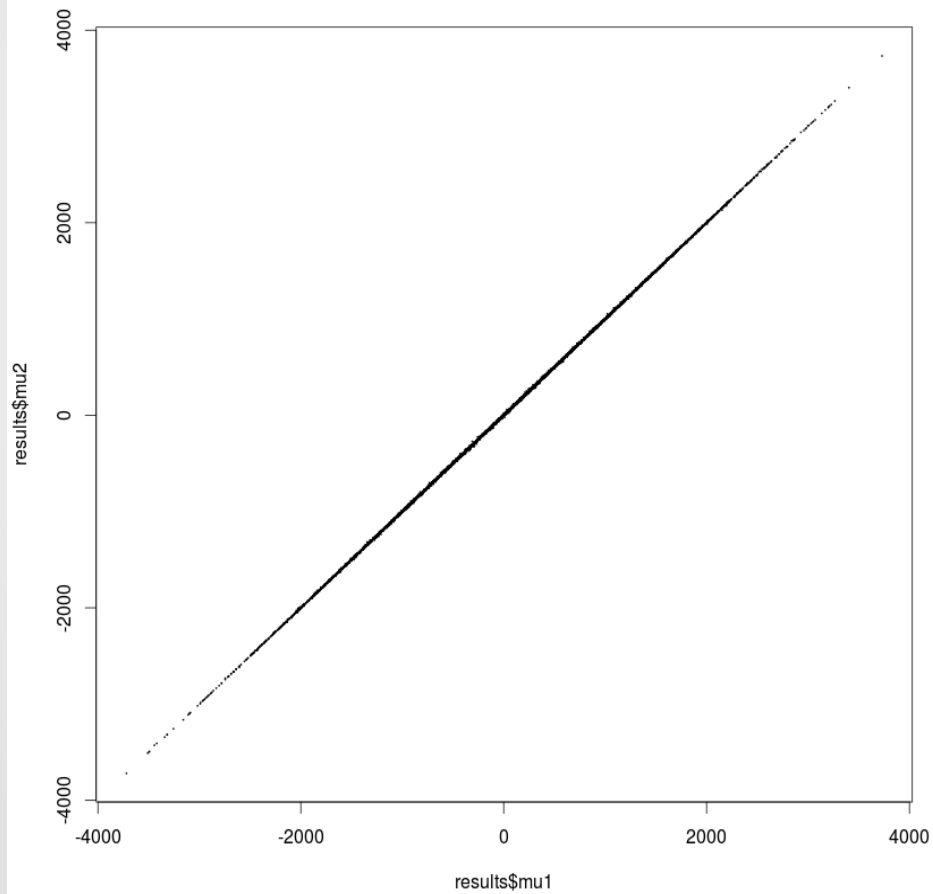

Histogram of results$difference
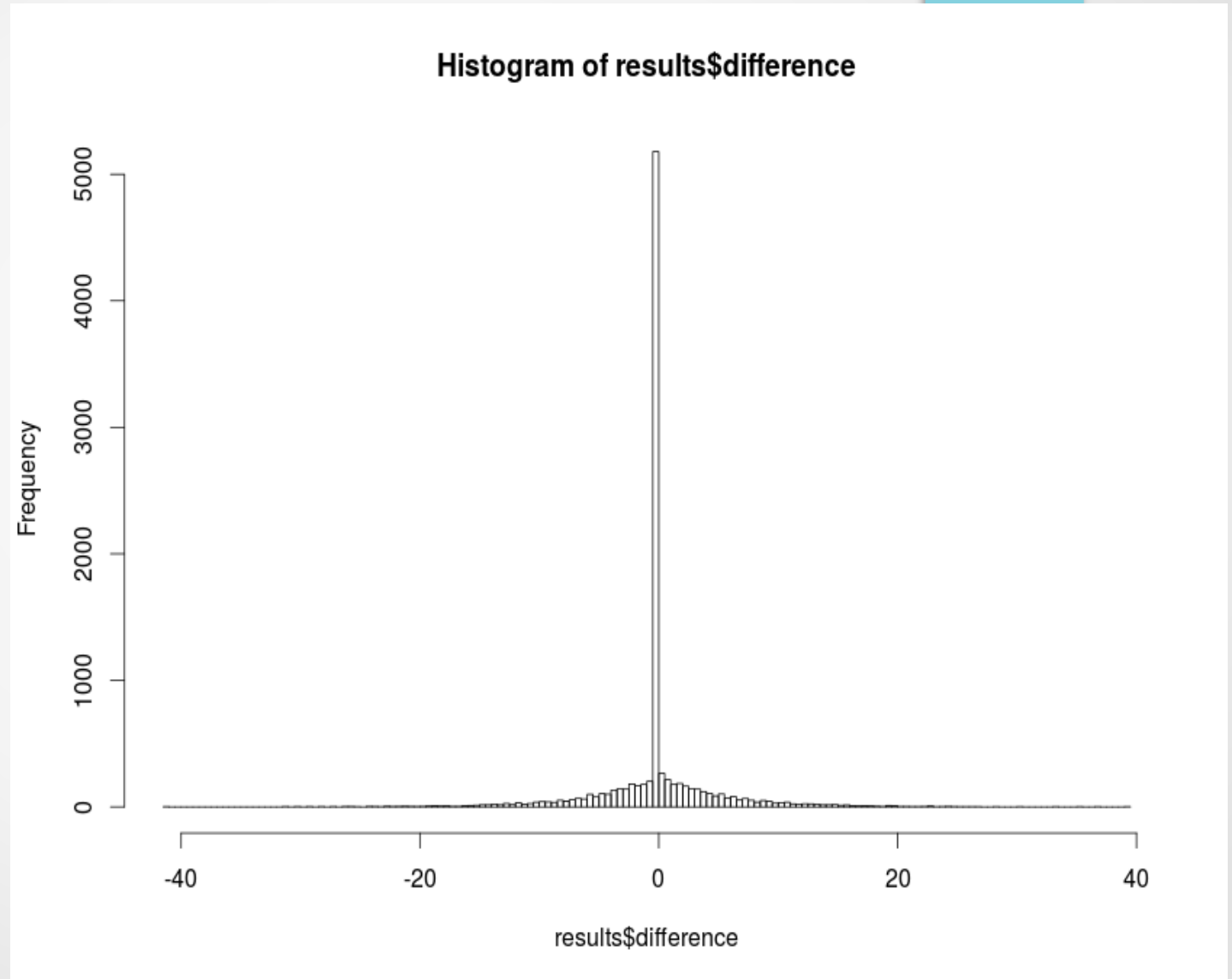
Implementation

```
u ~ dunif(-1, 1)
# Length of exponential prior if difference
# is not zero
L <- 5
size_of_difference <- step(u)*(-L*log(1 - u))


# To make the difference positive or negative
C ~ dbin(0.5, 1)
difference <- (2*C - 1)*size_of_difference
mu2 <- mu1 + difference
```

Model 2's Prior

Zoomed in

We can now run the model

- And we can actually get the posterior probability of the null hypothesis!

Summary of Models 1 and 2

- Model 1: "μ_1 and μ_2 could be anything"

- Model 2: "μ_1 could be anything. μ_2 might be exactly equal to μ_1. If not, it's at least close". L describes how close it might be.

Model 3

- Model 3 will express the idea "$\boldsymbol{\mu_1}$ and $\boldsymbol{\mu_2}$ might be close together (not exactly equal), or they might be far apart".

- A hierarchical model

Hierarchical Prior

Model 1 had:

```
mu1 ~ dnorm(0, 1/1000^2)

mu2 ~ dnorm(0, 1/1000^2)
```

- Let's change it to

```
mu1 ~ dnorm(grand_mean, 1/diversity^2)

mu2 ~ dnorm(grand_mean, 1/diversity^2)
```

Hyperparameters need priors too!

```
# Priors for the parameters
mu1 ~ dnorm(grand_mean, 1/diversity^2)
mu2 ~ dnorm(grand_mean, 1/diversity^2)

# Priors for the hyperparameters
grand_mean ~ dnorm(0, 1/1000^2)
log_diversity ~ dunif(-10, 10)
diversity <- exp(log_diversity)
```
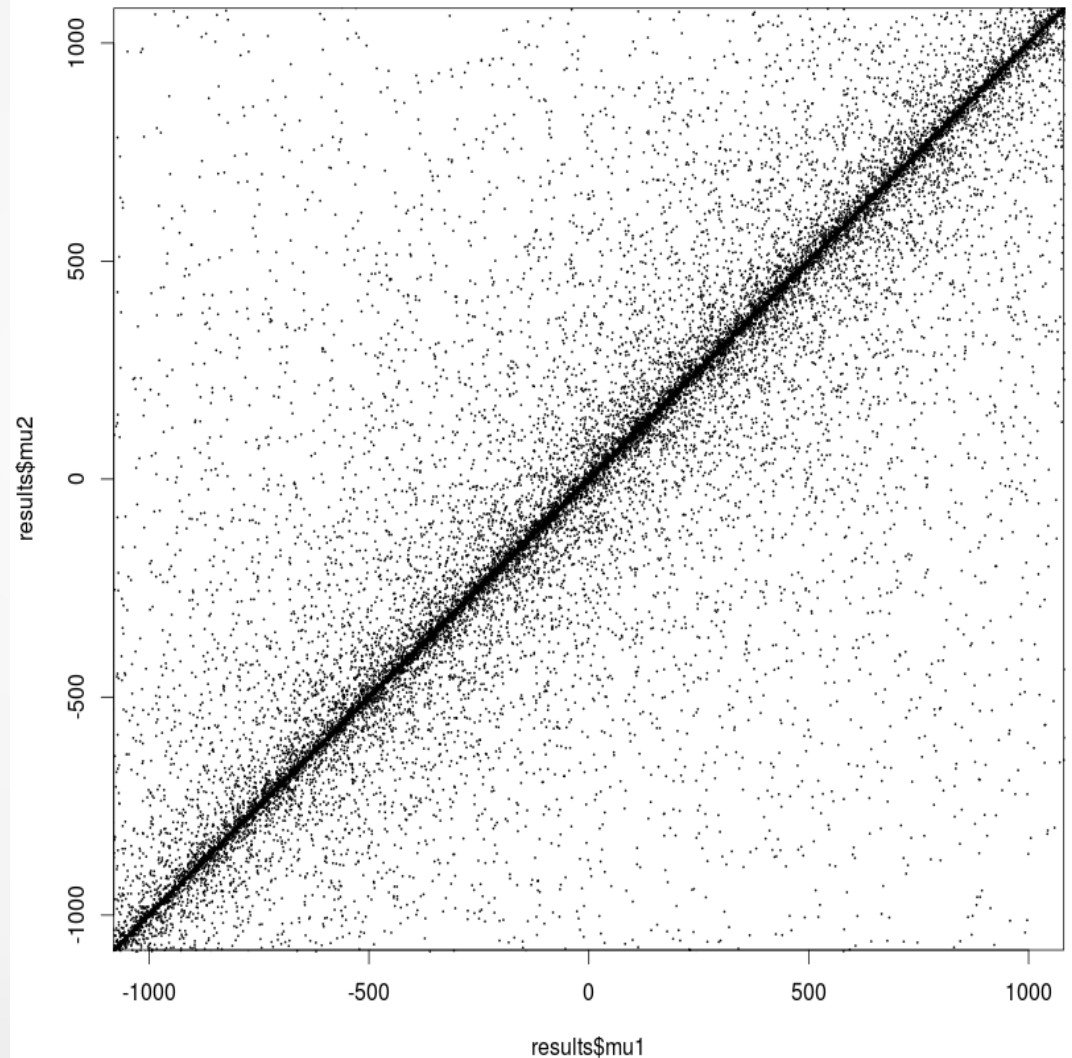
Model 3's Prior

- Zoomed in

- Allows μs to be similar or different (not exactly the same)

Model 3: Results on Widget Data

```
> mean(results$mu2 > results$mu1)
[1] 0.6302


> mean(results$mu2 == results$mu1)
[1] 0
```

Summary of three "t-test" models

- Model 1: "μ_1 and μ_2 could be anything"

- Model 2: "μ_1 could be anything. μ_2 might be exactly equal to μ_1. If not, it's at least close".

- Model 3: "μ_1 and μ_2 aren't precisely equal, that's silly. But they could be very similar. Or not."