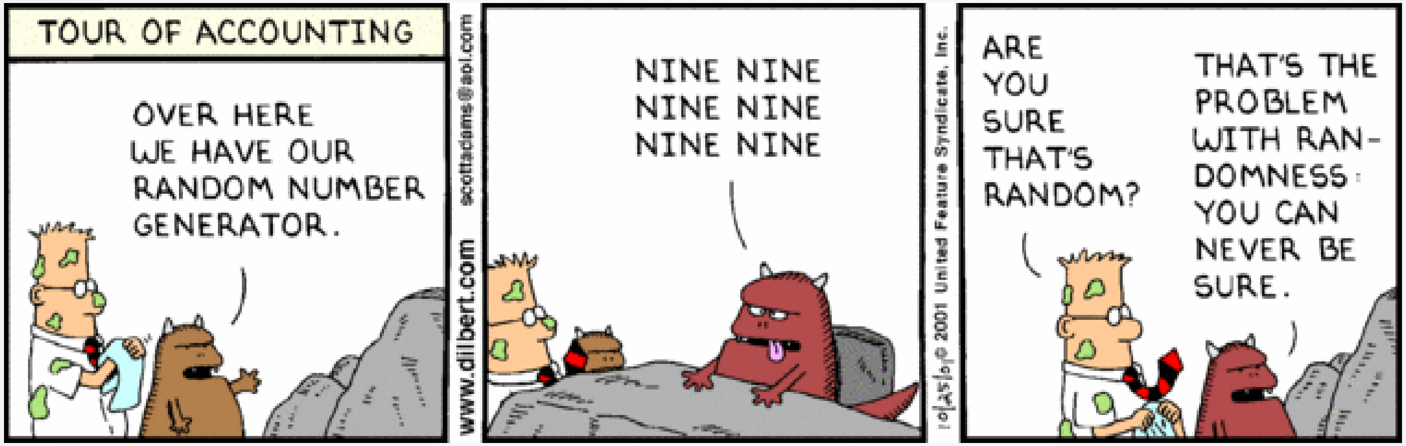
STATS331



Introduction to Bayesian Statistics
Semester 2, 2016

Today's Lecture

Getting to know JAGS

and

Midterm Test

Midterm Test

- Will be held in class first ***Wednesday*** back. ***14th September***

  Surname A-L: this room

  Surname M-Z: MLT3/303-101

- Worth 20%

Please remember

- Please arrive **on time** as the test will start at exactly 5 minutes past the hour

- Bring **pens** and a **calculator**

Mid Semester Test

- What to expect:
  - One or more Bayes' Boxes for you to read/interpret/complete
  - Simple Bayes' rule
  - Parameter estimation, hypothesis testing
  - Some Metropolis
  - NO JAGS

Practicing for the test

- I have put the 2012--2015 tests and solutions on Canvas.

- You can expect something roughly similar this year but with a few parts **asking for R code**

- You can also check out past exams, ignoring JAGS parts.

R code in the test

- I might put a snippet of R code in a question to show how a calculation was done

- I might ask questions along these lines:

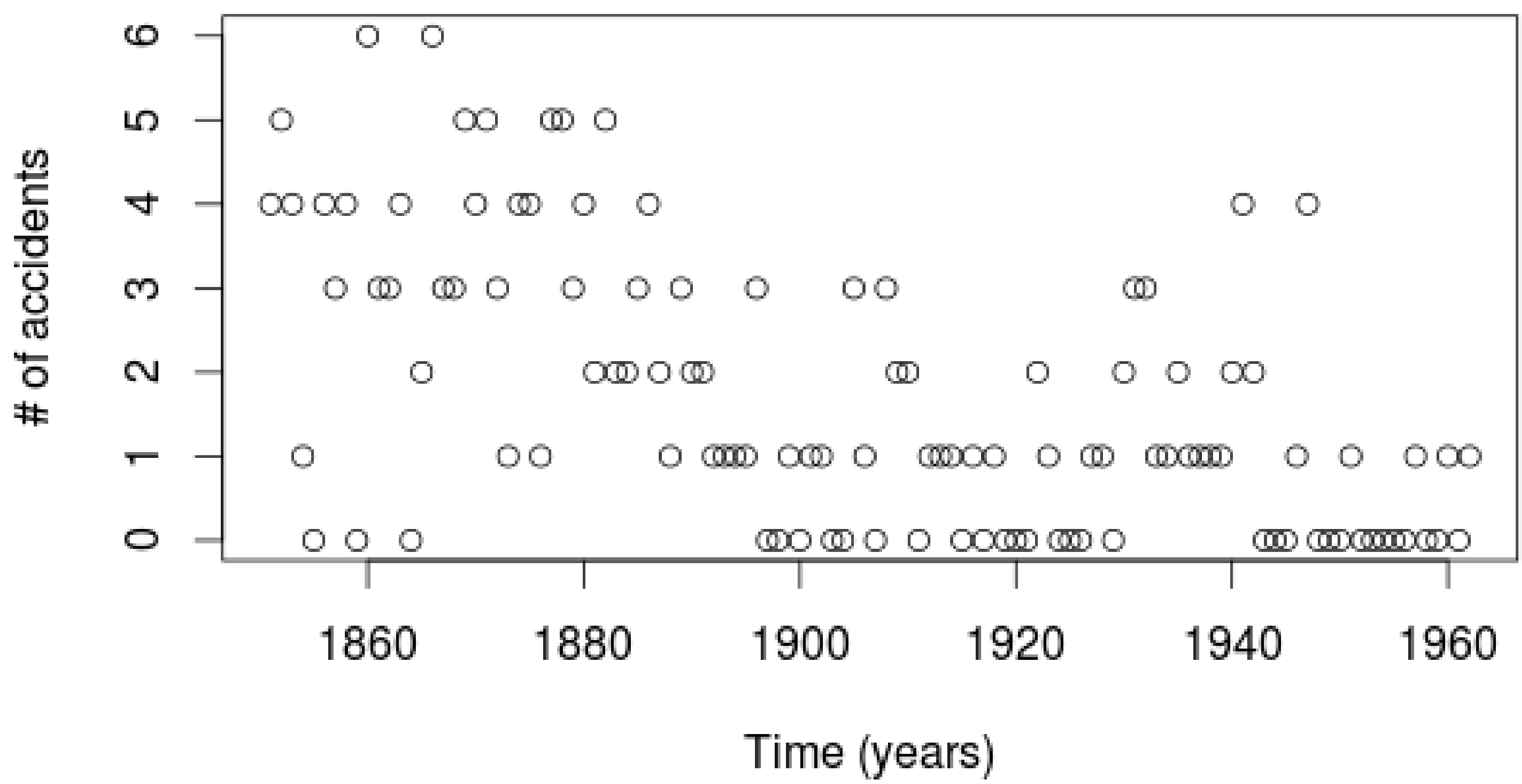
  Write down the R code you would use to calculate the posterior distribution shown in the Bayes' Box

  Write down the R code you would use to calculate ***some summary or prediction***

R code in the test

- While R will appear in the test, it will be in a couple of parts, not everywhere
- I will be fairly forgiving of minor syntax errors when I mark these questions

British Coal Mining Accidents

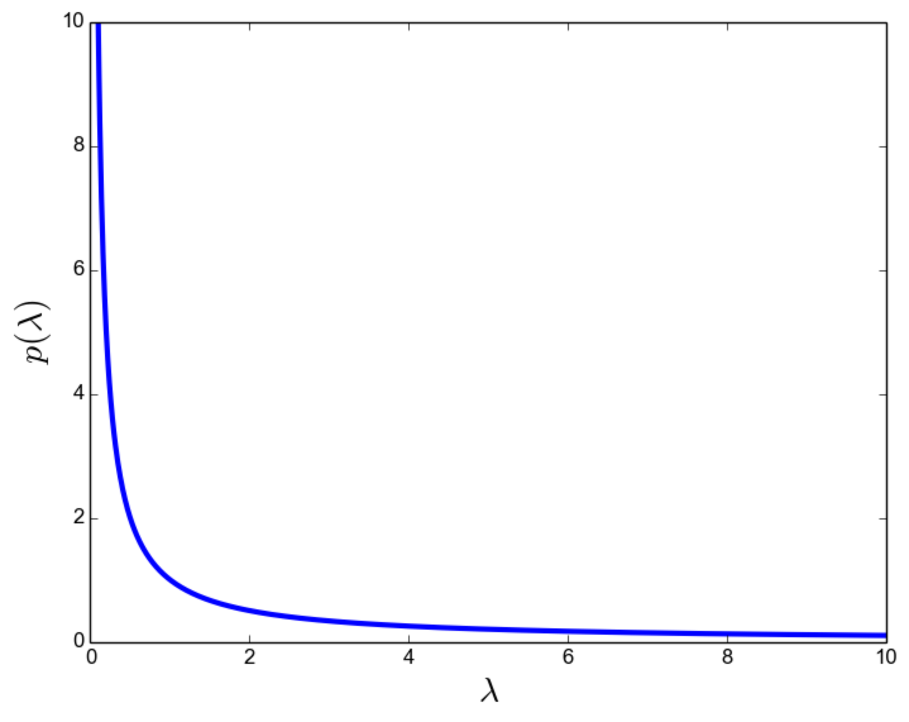


Coal Mining Accidents – Simple Model

```
model
{
  lambda ~ dunif(0, 10)

  for(i in 1:N)
  {
    y[i] ~ dpois(lambda)
  }
}
```

Log-Uniform Prior

- First view of log-uniform prior: density is p(λ) a 1/λ



- Second view: prior for log(λ) is uniform

Log-Uniform Prior in JAGS

```
log_lambda ~ dunif(-10, 10)
lambda <- exp(log_lambda)
```

Note use of “`<-`” to define one variable in terms of another

Terminology: `log_lambda` is a “stochastic node”, `lambda` a “deterministic node”

Log-Uniform Prior in JAGS

```
log_lambda ~ dunif(-10, 10)
lambda <- exp(log_lambda)
```

Lower limit for lambda is $e^{-10} = 0.000045$

Upper limit for lambda is $e^{10} = 22000$

A More Complex Model

- Let's revisit the “change-point” model, but change the priors for `lambda1` and `lambda2` to be log-uniform

The idea behind the change-point model is:

One lambda value applies at the beginning of the dataset

At some point, it jumps to another value

Change-Point Model

```
model
{
  log_lambda ~ dunif(-10, 10)
  lambda <- exp(log_lambda)
  log_lambda2 ~ dunif(-10, 10)
  lambda2 <- exp(log_lambda2)
  change_year ~ dunif(1851, 1962)

  for(i in 1:N)
  {
    mu[i] <- lambda + step(t[i] - change_year)*(lambda2 - lambda)
    y[i] ~ dpois(mu[i])
  }
}
```

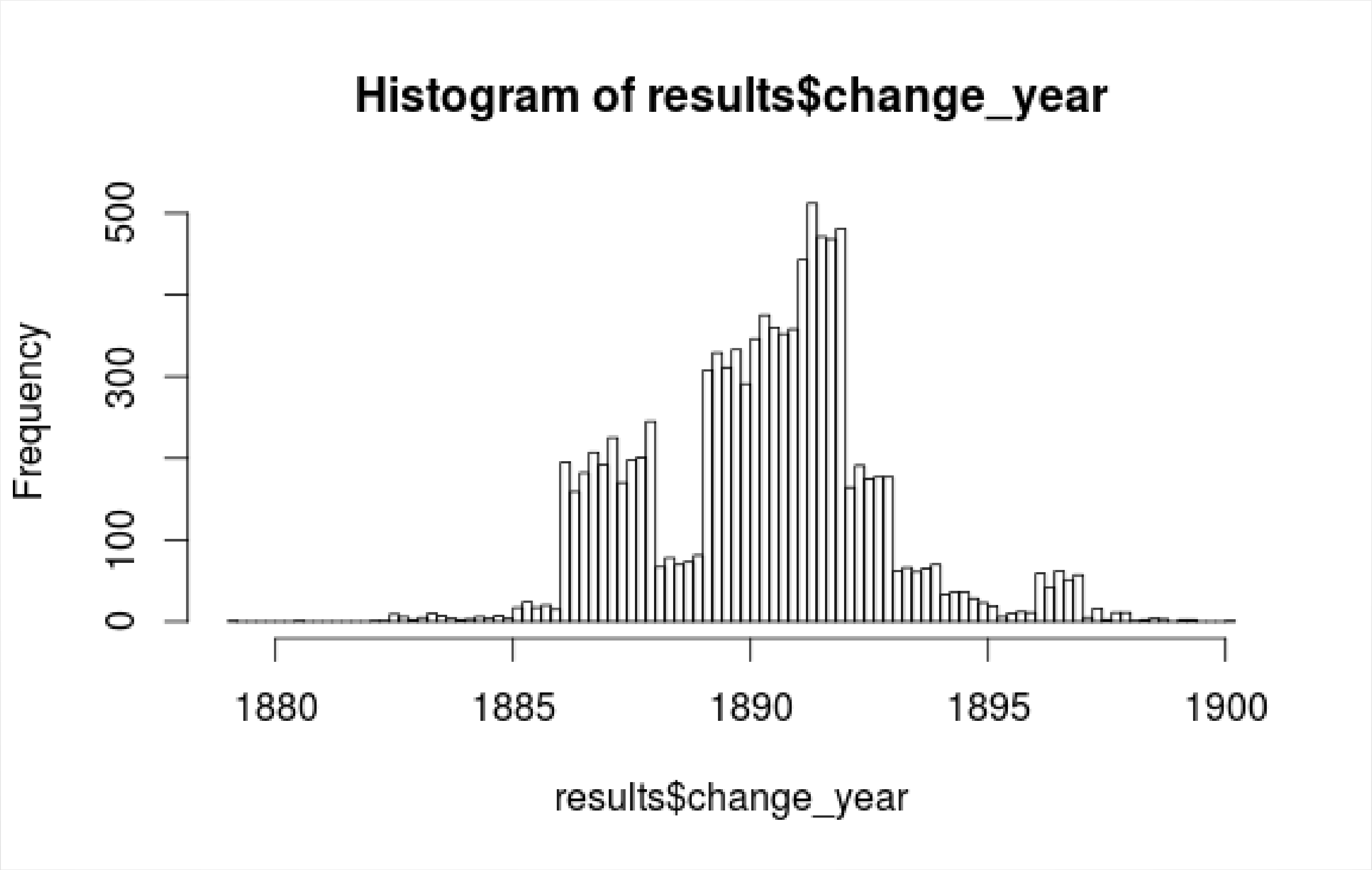
'step' function in JAGS

- Step returns 0 if the argument is negative and 1 otherwise.
- It was used to make our change-point model:

```
mu[i] <- lambda + step(t[i] - change_year)*(lambda2 - lambda)
```

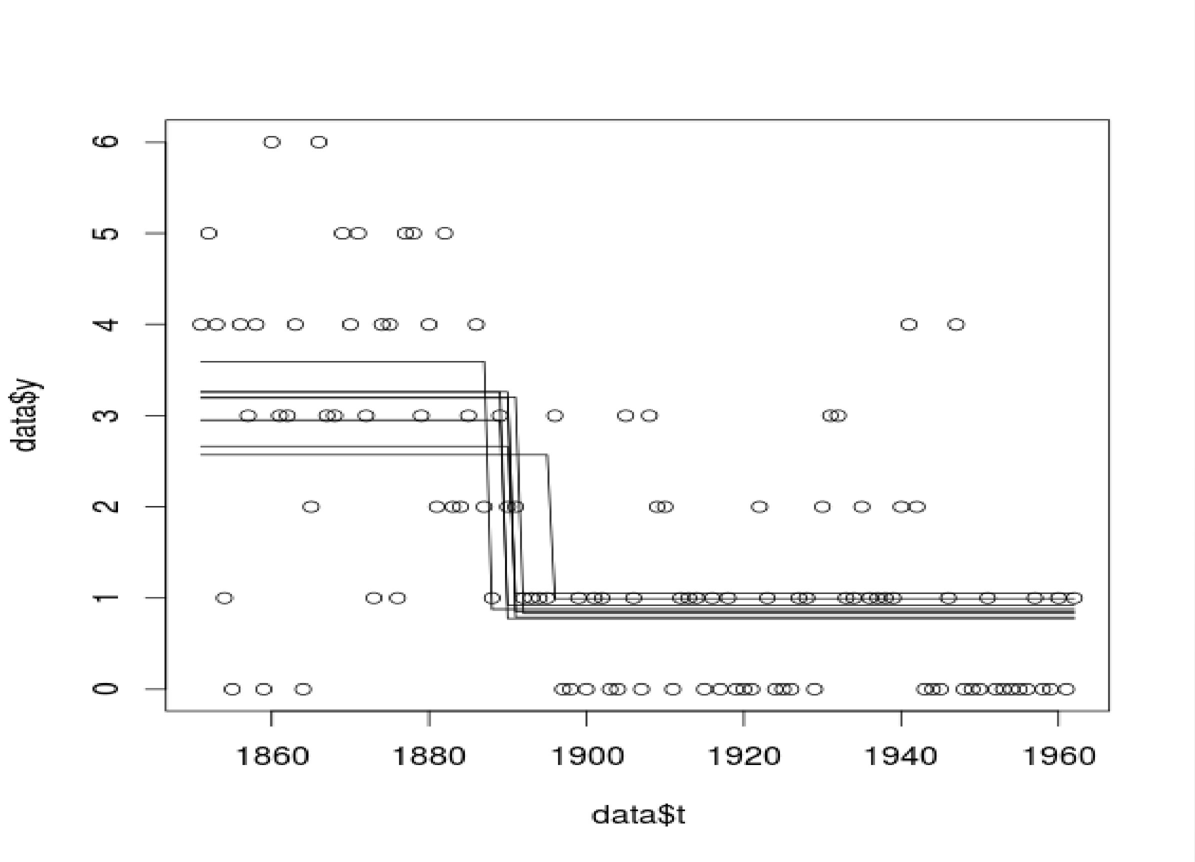
Results

- Main result: marginal posterior for change year



Plotting the model(s) through the data

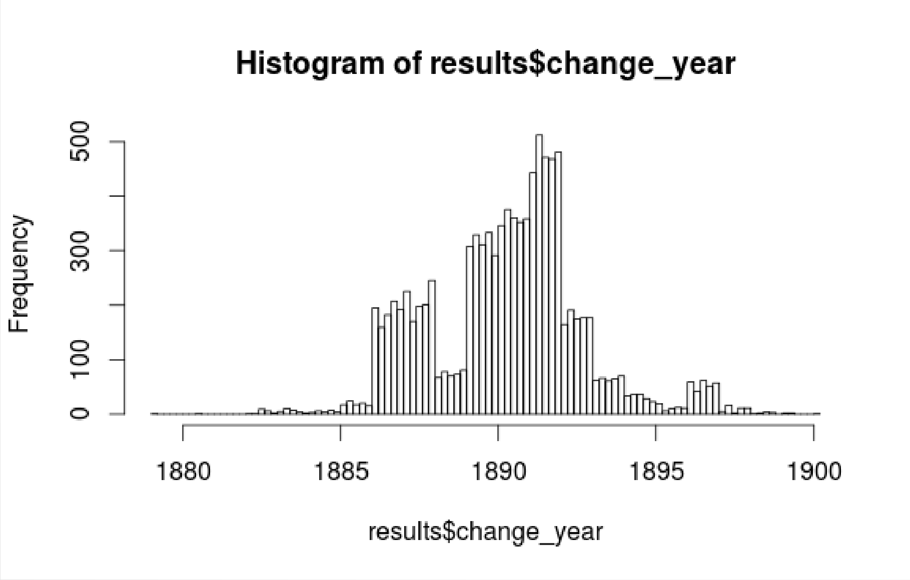
```
plot(data$t, data$y)

lines(data$t, results$mu[1, ])

lines(data$t, results$mu[50, ])
```



Summaries

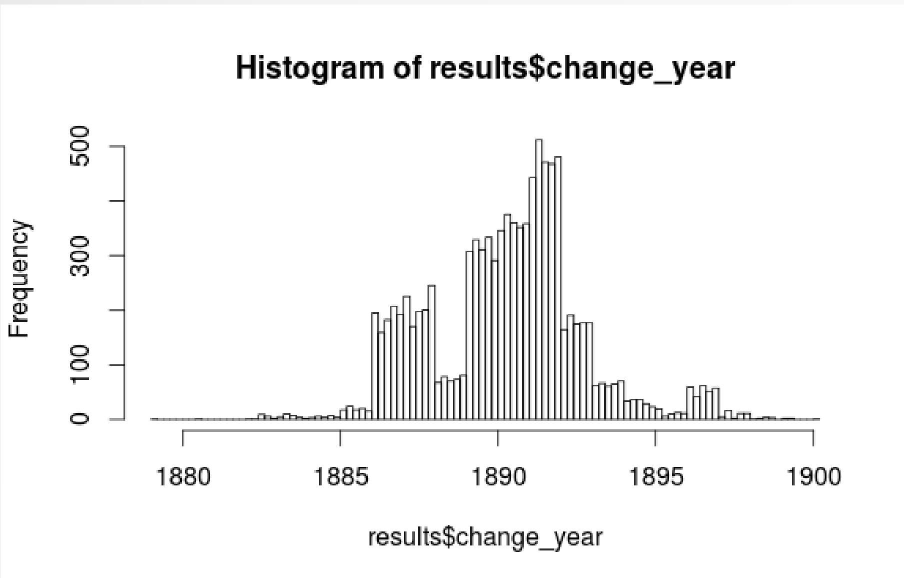
- Let's summarise the posterior for `change_year`
- The methods based on samples (MCMC output) are different (easier!) than the previous methods

Posterior Mean & Standard Deviation




```
> mean(results$change_year)
[1] 1890.358
> sd(results$change_year)
[1] 2.333427
```

95% Credible Interval



Find 2.5% and 97.5% quantiles of the posterior samples

```
> temp = sort(results$change_year)
> temp[0.025*length(temp)]
[1] 1886.142
> temp[0.975*length(temp)]
[1] 1896.03
```

Enjoy your break!

- But come to lab! :-)