# Model Checking

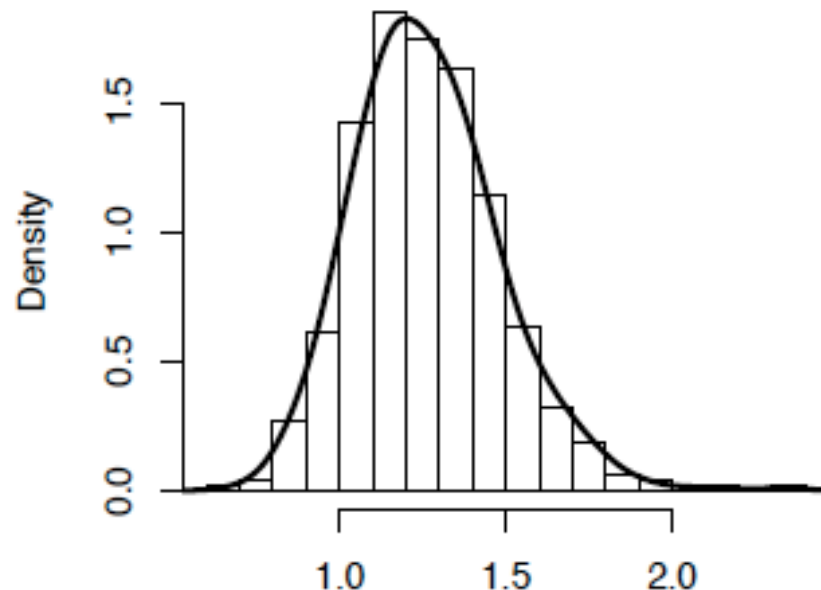Bayesian Modeling for Socio-Environmental Data

N. Thompson Hobbs

July 27, 2016

The first question we should ask after fitting a model: *Are the predictions of the model consistent with the data?*

- Is our deterministic model a reasonable representation of the process?

- Have we made the right choices of distributions to represent the uncertainties?

# Marginal distributions of random variables A an B

|        | $a_1$           | $a_2$           | $a_3$           | $a_4$           | $\Pr(B)\downarrow$ |
|--------|-----------------|-----------------|-----------------|-----------------|--------------------|
| $b_1$  | $\frac{2}{20}$  | $0$             | $\frac{1}{20}$  | $\frac{3}{20}$  | $\frac{6}{20}$     |
| $b_2$  | $0$             | $\frac{4}{20}$  | $0$             | $\frac{6}{20}$  | $\frac{10}{20}$    |
| $b_3$  | $\frac{1}{20}$  | $\frac{1}{20}$  | $\frac{1}{20}$  | $0$             | $\frac{3}{20}$     |
| $b_4$  | $0$             | $0$             | $0$             | $\frac{1}{20}$  | $\frac{1}{20}$     |
| $\Pr(A)\rightarrow$ | $\frac{3}{20}$ | $\frac{5}{20}$ | $\frac{2}{20}$ | $\frac{10}{20}$ | $\frac{20}{20}$ |

If we have a function $f(A,B)$ specifying the joint probability of the discrete random variables A and B, then

$$\sum_A f(A,B)$$ is the marginal probabilty of B

and

$$\sum_B f(A,B)$$ is the marginal probability of A.

This same idea applies to any number of jointly distributed random variables. We simply sum over all but one.

**A**

**B**

**C**

**D**
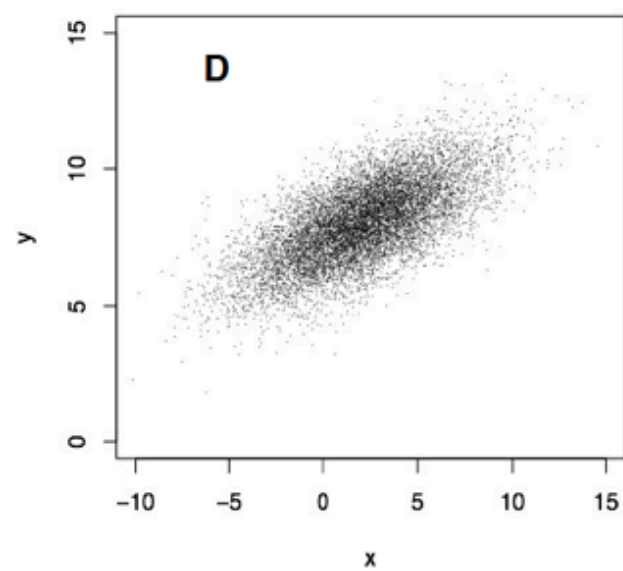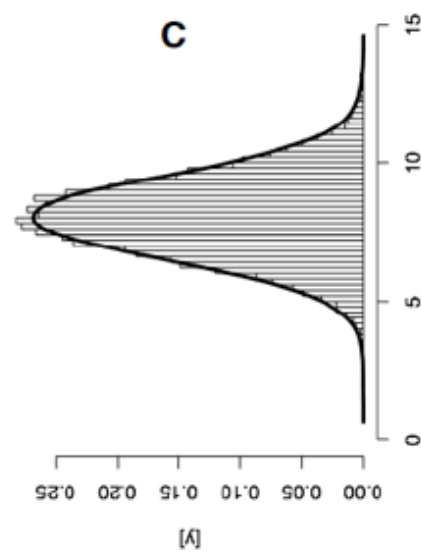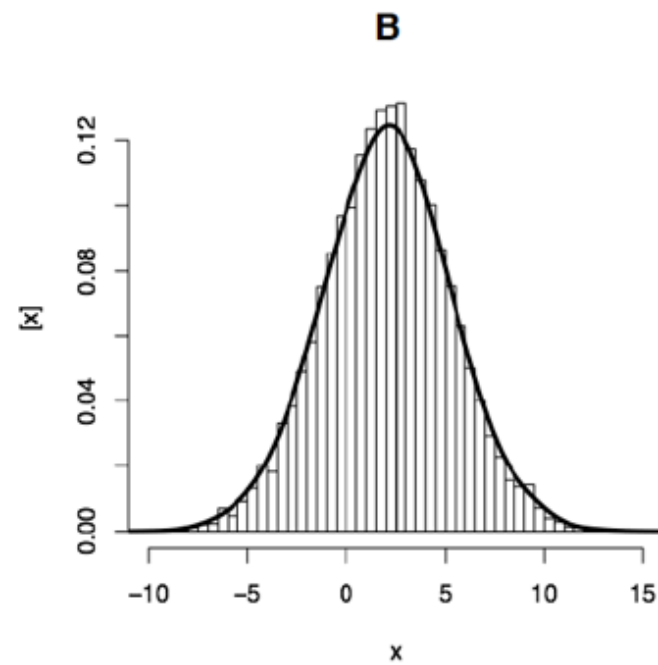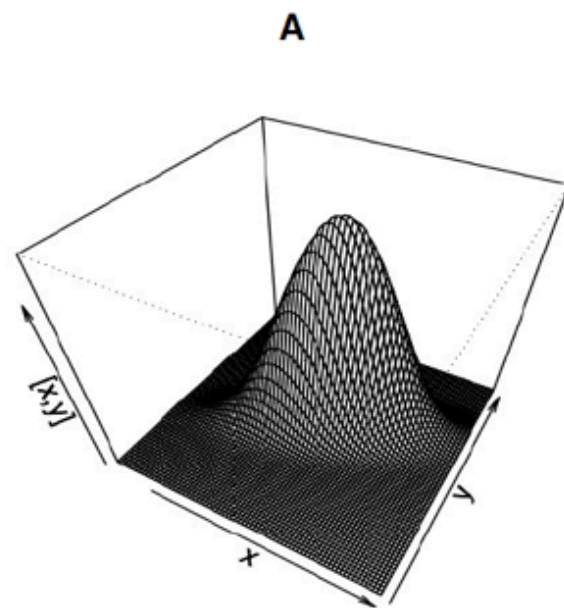
If we have a function $f(A,B)$ specifying the joint probability of the continuous random variables A and B, then

$$\int_B f(A,B)\,dB \text{ is the marginal probabilty of A}$$

and

$$\int_A f(A,B)\,dA \text{ is the marginal probability of B.}$$

This same idea applies to any number of jointly distributed random variables. We simply integrate over all but one.

# Posterior predictive checks

$$\left[ y^{new} \mid \mathbf{y} \right] = \underbrace{\int_{\theta_1} \dots \int_{\theta_n} [y^{new} \mid \theta_1 \dots \theta_n][\theta_1 \dots \theta_n \mid \mathbf{y}] d\theta_1 \dots d\theta_n}_{\text{Posterior Predictive Distribution}}$$

It is called posterior because it is conditional on the observed $\mathbf{y}$ and predictive because it is a prediction for an observable $y^{new}$. It gives the probability of a new prediction of y conditional on $\boldsymbol{\theta}$, which, in turn, is conditional on the data in hand, $\mathbf{y}$. Note that it is a marginal distribution because we are integrating over the $\boldsymbol{\theta}$.

$$\mu_i = g(\theta_1, \theta_2, \theta_3, x_i)$$

$$y_i \sim \text{normal}(\mu_i, \sigma^2)$$

Also see box 8.1 in
Hobbs and Hooten

A new data set at each iteration

| $k$ | $\theta_1$ | $\theta_1$ | $\theta_3$ | $i=1$ | $i=2$ | $i=3$ | ... | $i=Y$ |
|---|---|---|---|---|---|---|---|---|
| 1 | .42 | 3.3 | 20.3 | $y^{new}_{1,1}$ | $y^{new}_{1,2}$ | $y^{new}_{1,3}$ | $\cdots$ | $y^{new}_{1,Y}$ |
| 2 | .41 | 2.3 | 18.5 | $y^{new}_{2,1}$ | $y^{new}_{1,2}$ | $y^{new}_{1,3}$ | $\cdots$ | $y^{new}_{1,Y}$ |
| 3 | .46 | 3.1 | 16.6 | $y^{new}_{3,1}$ | $y^{new}_{1,2}$ | $y^{new}_{1,3}$ | $\cdots$ | $y^{new}_{1,Y}$ |
| $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ | | $\vdots$ |
| $K$ | .39 | 3.4 | 22.1 | $y^{new}_{n,1}$ | $y^{new}_{n,2}$ | $y^{new}_{n,3}$ | $\cdots$ | $y^{new}_{1,Y}$ |

# This is easier done than said.

We have a model $g(\theta, x)$ that predicts a response $y$.

We estimate the posterior distribtuion, $[\theta \mid y]$.

For any given value of $x_i$, we can simulate the posterior predictive distribution $y_i^{new}$ by making a draw from $[y_i^{new} \mid g(\theta, x_i), \sigma^2]$. In MCMC, this simply means making draws from the data model at each iteration because each draw is conditional on the current values of the parameters. We simulate a new dataset by repeating these draws for all values of the $x$.

Accumlating many of these draws defines the posterior predictive distribution in exactly the same way that many draws allow us to define the posterior distribution of the parameters.

$$g(b_0, b_1, x_i) = b_0 + b_1 x_i$$

$$[b_0, b_1, \tau \mid \mathbf{y}] \propto \prod_{i=1}^{n} \text{normal}\left(y_i \mid g(b_0, b_1, x_i)_i, \tau\right) \times$$

$$\text{normal}\left(b_0 \mid 0.0001\right) \text{normal}(b_1 \mid 0, .0001) \text{gamma}\left(\tau \mid .01 .01\right)$$

```
model{
b0 ~ dnorm(0,.0001)
b1 ~ dnorm(0,.0001)
tau ~ dgamma(.01,.01)
sigma<-1/sqrt(tau)
for(i in 1:length(y)){
  mu[i] <- b0 + b1*x[i]
  y[i] ~ dnorm(mu[i],tau)
  #posterior predictive distribution of y.new[i]
  y.new[i] ~ dnorm(mu[i],tau)
  }
}
```

# Posterior Predictive Checks

$T(\mathbf{y}, \theta)$ is a test statistic (e.g., mean, standard deviation, CV, quantile, or sums of squares discrepancy) calcuated from the observed data.

$T(\mathbf{y}^{new}, \theta)$ is the corresponding statistic from the new "data" from the posterior predictive distribution.

We calcuate:

$$P_B = \Pr\left(T(\mathbf{y}^{new}, \theta) \geq T(\mathbf{y}, \theta) \mid \mathbf{y}\right)$$

If $P_B$ is very large or very small, then the difference between the observed data and the simulated data cannot be attributed to chance. This indicates lack of fit.

# Candidates for test statistics

- Mean

- variance

- Coefficient of variation

- quantiles

- maximum, minimum

- discrepancy: (observation - prediction)$^2$

- chi-square: $T(y,\theta) = \sum_i = \dfrac{\left(y_i - \mathrm{E}(y_i \mid \theta)\right)}{\mathrm{var}(y_i \mid \theta)}$

- deviance: $T(y,\theta) = -2\log[y \mid \theta]$

# R. A. Fischer's Ticks

A simple example: We want to know (for some reason) the average number of ticks on sheep. We round up 60 sheep and count ticks on each one. Does a Poisson distribution fit the distribution of the data?

$$[\lambda \mid \mathbf{y}] \propto \prod_{i=1}^{60} \text{Poisson}(y_i \mid \lambda)[\lambda]$$

For each value of $\lambda$ in the MCMC chain, we generate a new data set, $\mathbf{y}^{new}$, by sampling from

$$y_i^{\,new} \sim \text{Poisson}(\lambda)$$

```
model{
lambda ~ dgamma(0.001,0.001)
for(i in 1:60){
    y[i] ~ dpois(lambda)
    y.new[i] ~ dpois(lambda)   #simulate a new data set of 60 points
}
cv.y <- sd(y[ ])/mean(y[ ])
cv.y.new <- sd(y.new[])/mean(y.new[ ])
pvalue.cv <- step(cv.y.new-cv.y)  # find Bayesian P value--the mean of
many 0's and 1's returned by the step function, one for each iteration in
the chain. The function step(z) returns a 1 if z > 0, returns 0
otherwise.
mean.y <-mean(y[])
mean.y.new <-mean(y.new[])
pvalue.mean <-step(mean.y.new - mean.y)
for(j in 1:60){
    sq[j] <- (y[j]-lambda)^2
    sq.new[j] <- (y.new[j]-lambda)^2
}
fit <- sum(sq[])
fit.new <- sum(sq.new[])
pvalue.fit <- step(fit.new-fit)
} #end of model
```

Key bit!
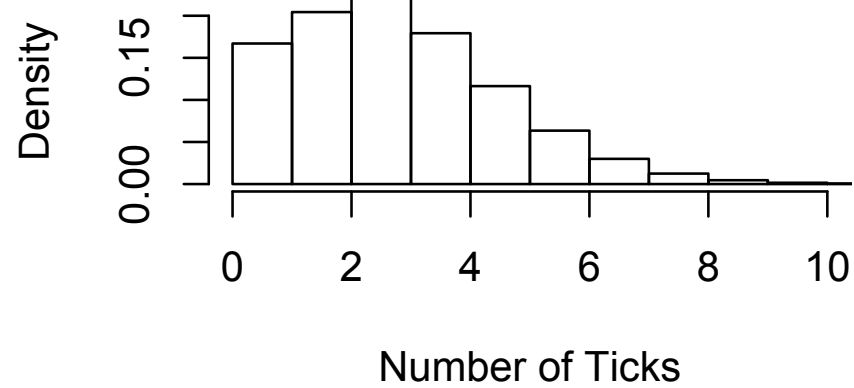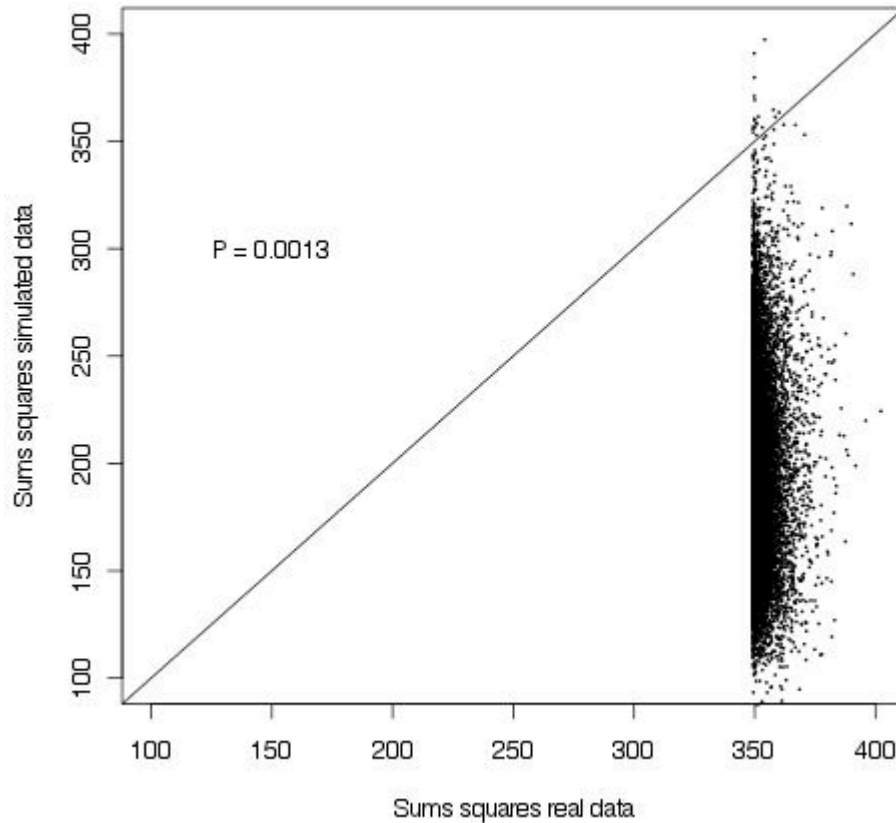
**Real Data**

Density

0.20

0.00

0   2   4   6   8   10

Number of Ticks
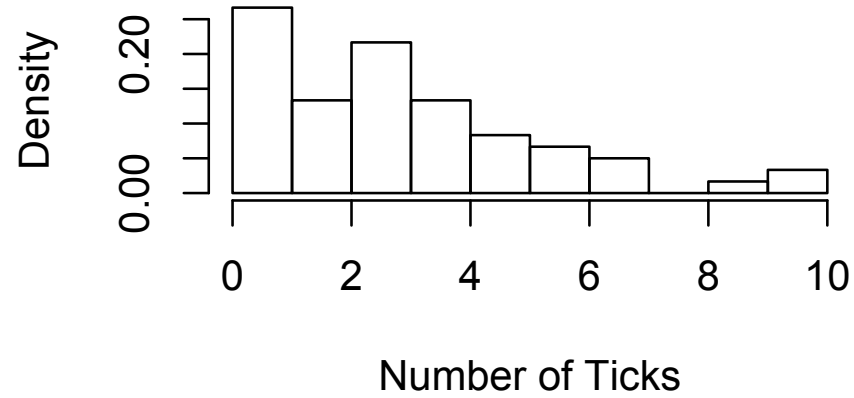
**Simple model**

**Simulated Data**

Density

0.15

0.00

0   2   4   6   8   10

Number of Ticks

# Posterior predictive check



```
        }
        fit <- sum(sq[])
        fit.new <- sum(sq.new[])
        pvalue.fit <- step(fit.new-fit)
        } #end of model
```

## Simple model

P value for CV= .0013
P value for mean = .51

Remember, this is a two-tailed probability, so values close to 0 and 1 indicate lack of fit.

How could you modify this model to allow "extra" variance?   Draw a Bayesian network and write out the posterior and joint distributions.  You may not use the negative binomial.  Hint- remember the owls fecundity model.
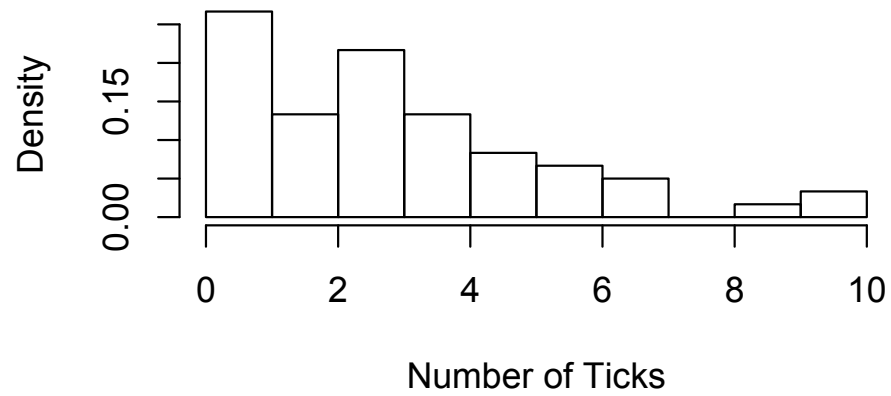
## Hierarchical model

```
model{
a~ dgamma(.001,.001)
b~ dgamma(.001,.001)
for(i in 1:60){
    lambda[i] ~ dgamma(a,b)
     y[i] ~ dpois(lambda[i])
     y.sim[i] ~ dpois(lambda[i])
}
cv.y <- sd(y[ ])/mean(y[ ])
cv.y.sim <- sd(y.sim[])/mean(y.sim[ ])
pvalue.cv <- step(cv.y.sim-cv.y)  # find Bayesian P
 value--the mean of many 0's and 1's returned by
 the step function, one for each step in the chain
mean.y <-mean(y[])
mean.y.sim <-mean(y.sim[])
pvalue.mean <-step(mean.y.sim - mean.y)
for(j in 1:60){
    sq[j] <- (y[j]-lambda[j])^2
    sq.new[j] <- (y.sim[j]-lambda[j])^2
}
fit <- sum(sq[])
fit.new <- sum(sq.new[])
pvalue.fit <- step(fit.new-fit)
} #end of model
```

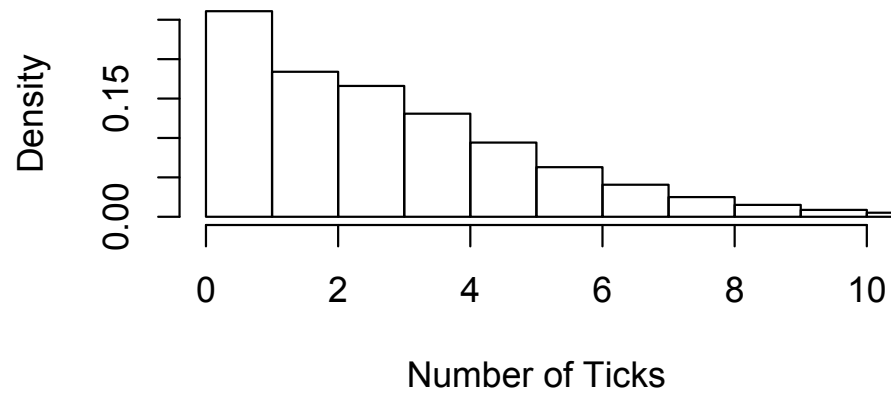$$[a, b, \boldsymbol{\lambda}|\mathbf{y}] \propto \prod_{i=1}^{60} [y_i|\lambda_i] [\lambda_i|a, b] [a] [b]$$

Include **pvalue.fit** in variable names list for `coda.samples` or `jags.samples`. Report the mean of **pvalue.fit**
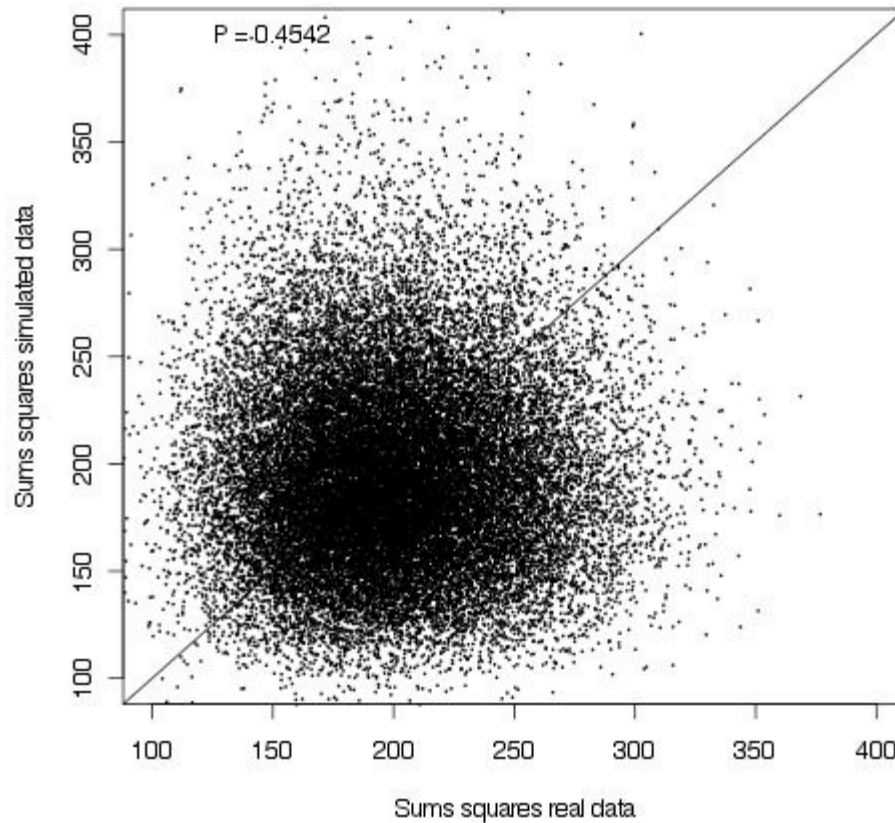
**Real Data**

Density

**Simulated Data**

Density

Number of Ticks

Hierarchical model

# Posterior predictive  check



P =-0.4542

Sums squares simulated data

Sums squares real data

## Hierarchical model

P value for CV= .45
P value for mean = .50