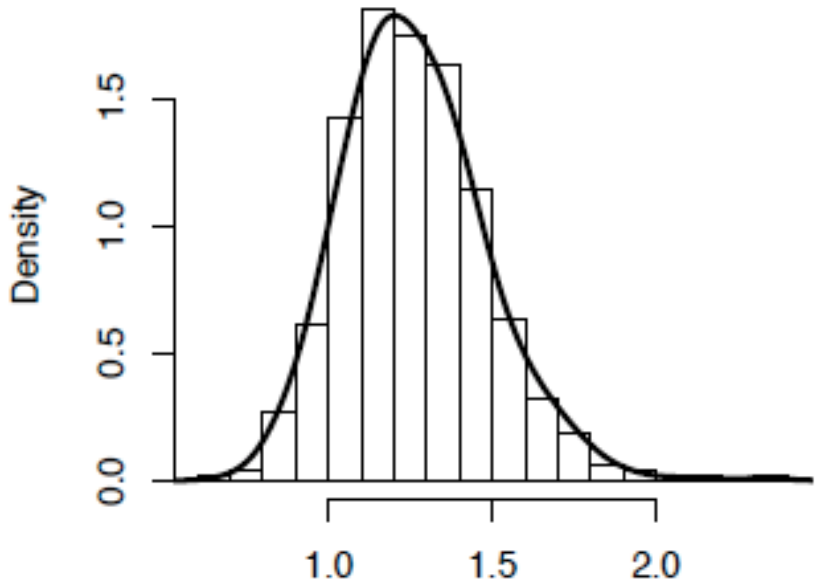
Can my model give rise to the data?

Posterior Predictive Checks

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

- Is our process model a reasonable representation?
- Have we made the right choices of distributions to represent the uncertainties?



Posterior predictive checks

$$P\left(\mathbf{y}^{new} \mid \mathbf{y}\right) = \underbrace{\int_{\theta} P\left(\mathbf{y}^{new} \mid \theta\right) P\left(\theta \mid \mathbf{y}\right) \mathrm{d}\theta}_{\text{Posterior Predictive Distribution}}$$

It is called posterior because it is conditional on the observed **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 **θ**, which, in turn, is conditional on the data in hand, **y**. Note that it is a marginal distribution because we are integrating over the **θ.**

A new data set at each iteration

i	θ_1	θ_1	θ_3					
1	.42	3.3	20.3	$y_{1,1}^{new}$	$y_{1,2}^{new}$	$y_{1,3}^{new}$	$\cdots$	$y_{1,Y}^{new}$
2	.41	2.3	18.5	$y_{2,1}^{new}$	$y_{1,2}^{new}$	$y_{1,3}^{new}$	$\cdots$	$y_{1,Y}^{new}$
3	.46	3.1	16.6	$y_{3,1}^{new}$	$y_{1,2}^{new}$	$y_{1,3}^{new}$	$\cdots$	$y_{1,Y}^{new}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$		$\vdots$
n	.39	3.4	22.1	$y_{n,1}^{new}$	$y_{n,2}^{new}$	$y_{n,3}^{new}$	$\cdots$	$y_{1,Y}^{new}$

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 $[y_i^{new} \mid g(\theta, x_i), \sigma]$. 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

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

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

We calcuate:

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

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 $T(y)$, $T(y^{new})$

- Mean
- variance
- Coefficient of variation
- quantiles
- maximum, minimum
- discrepancy: (observation - prediction)^2
- chi-square: $T(y,\theta) = \sum_i = \frac{\left(y_i - E(y_i \mid \theta)\right)}{\mathrm{var}(y_i \mid \theta)}$
- deviance: $T(y,\theta) = -2\log(p(y \mid \theta))$

R. A. Fischer's Ticks

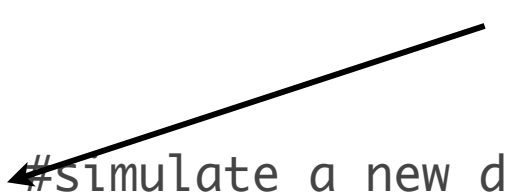
A simple example: We want to know (for some reason) the average number of ticks on sheep. We round up 60 sheep and count ticks on each one. Does a Poison distribution fit the distribution of the data?

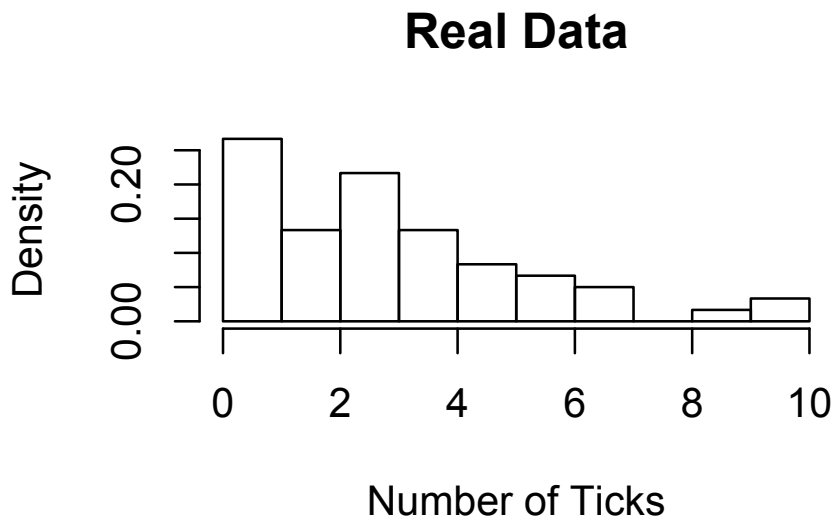
$$[\lambda \mid \mathbf{y}] \propto \prod_{i=1}^{60} \text{Poisson}\left(y_i \mid \lambda\right)[\lambda]$$

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

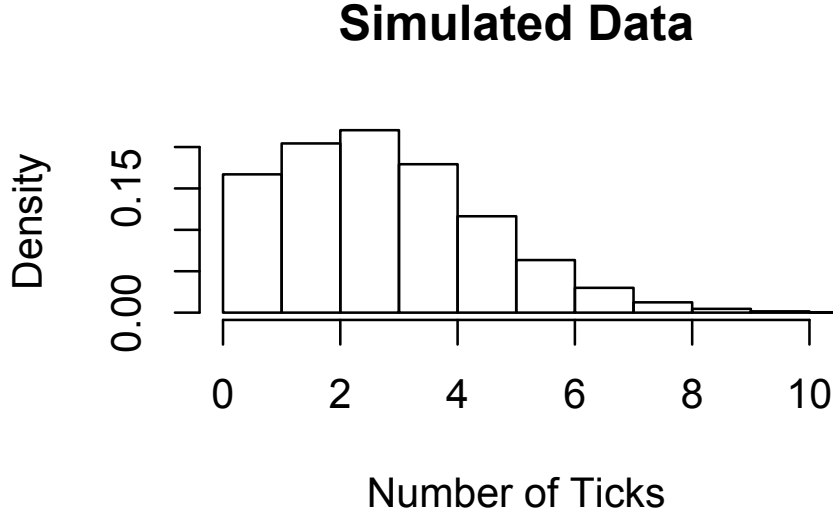
$$[\mathbf{y}^{new} \mid \lambda][\lambda \mid \mathbf{y}]$$

```
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
```

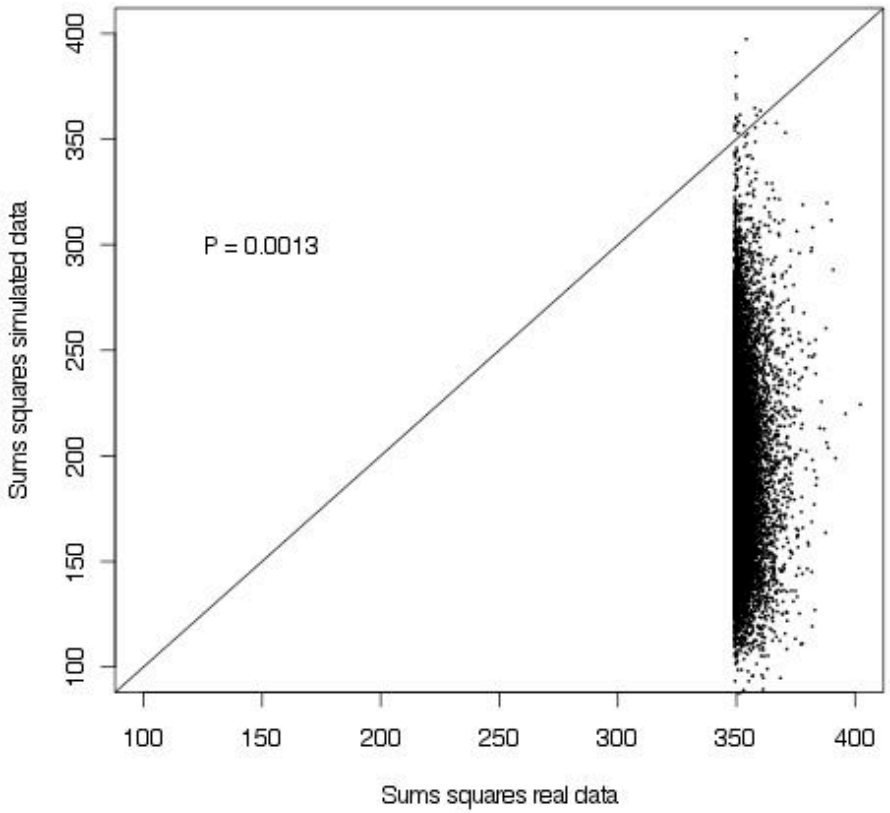




Simple model



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

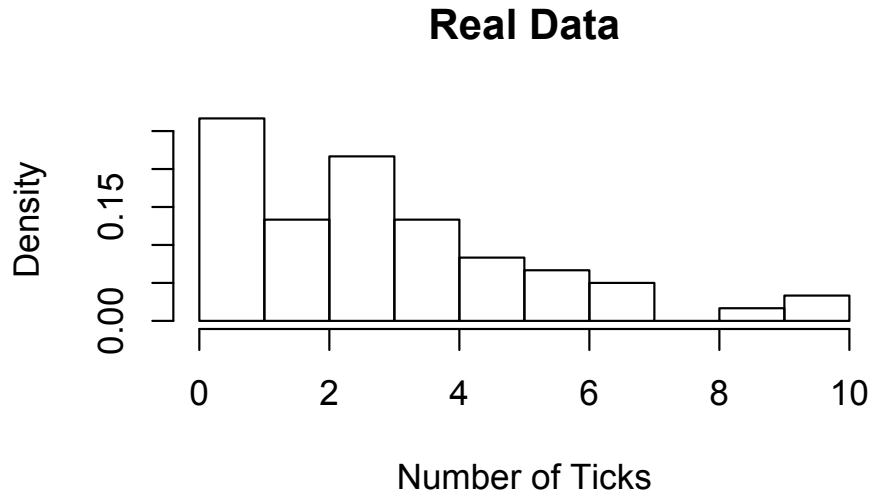
Remember, this is a two-tailed probability, so values close to 0 and 1 indicate lack of fit.

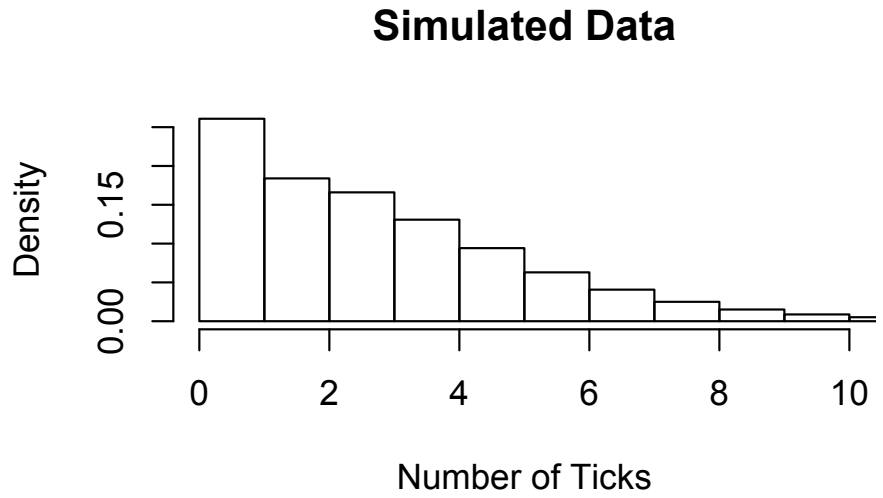
How could you modify this model to allow “extra” variance? Draw a Bayesian network and write out the posterior and joint distributions. Hint- remember Poisson plants.

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 chaing
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
```

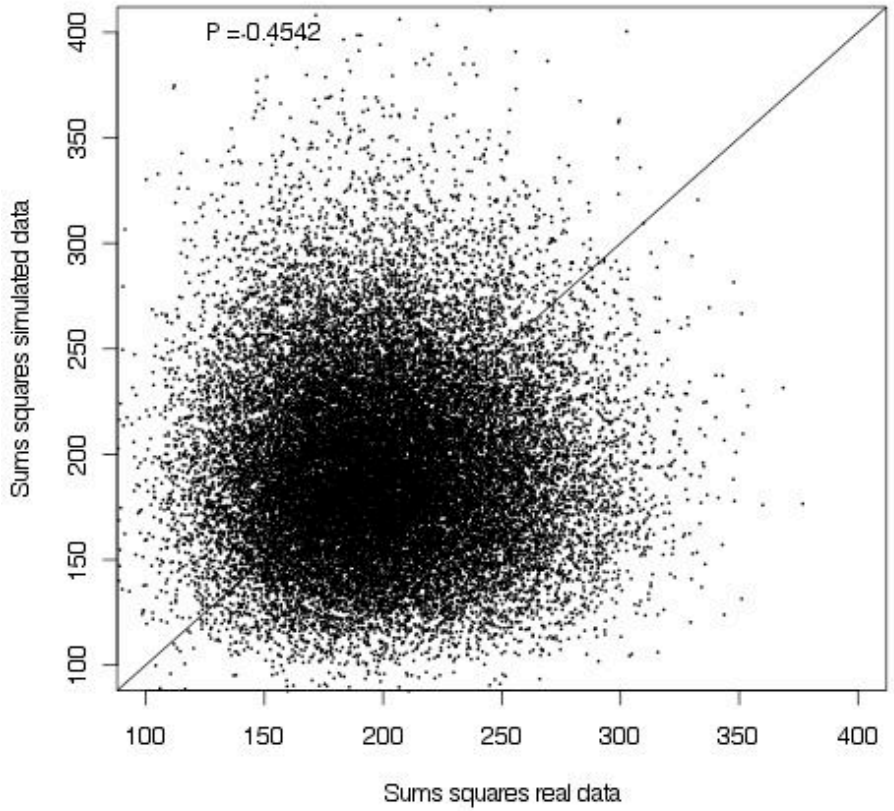
$$[a, b, \boldsymbol{\lambda}|\mathbf{y}] \propto \prod_{i=1}^{60} [y_i|\lambda_i]\,[\lambda_i|a, b]\,[a]\,[b]$$





Hierarchical model

Posterior predictive check



Hierarchical model

P value for CV= .45

P value for mean = .50