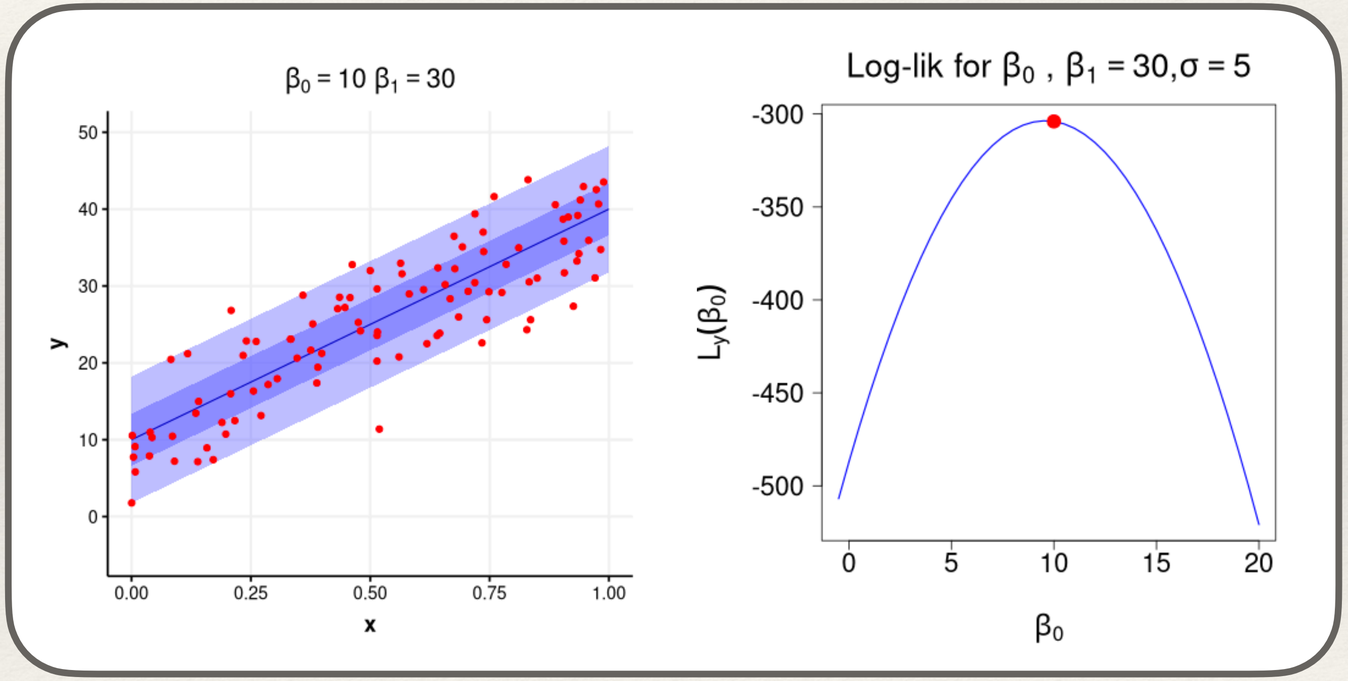
PROBABILISTIC MODELING

Introduction to Bayesian Statistics

Diogo Melo
Lewis-Sigler Institute of Integrative Genomics
damelo@princeton.edu

MAXIMUM LIKELIHOOD ESTIMATION

- Up to this point we have been using the **Maximum Likelihood principle**
- ML has many advantages:
  - General
  - (mostly) Automatic
  - Good theoretical justification
  - Good performance



THE MAXIMUM LIKELIHOOD WAY

Given some data...

$$y = \{y_1, \cdots, y_n\}$$

Define an observational model:

$$P(y \mid \theta) = L_y(\theta)$$

Maximize the likelihood over the parameter space:

$$\hat{\theta} = \underset{\theta \in \Omega}{\operatorname{argmax}} \left[L_y(\theta) \right]$$

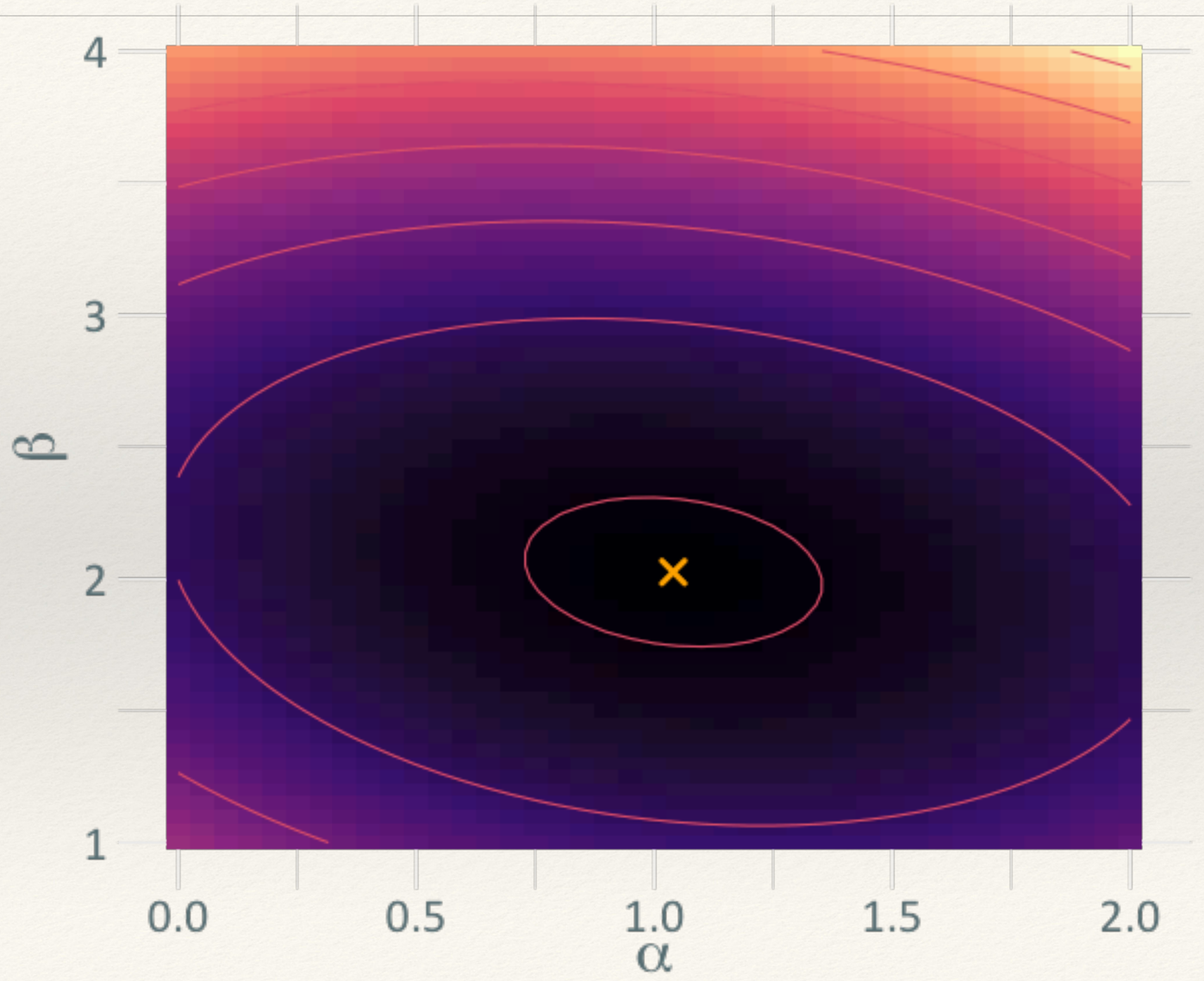
Any further inference uses this ML estimator:

$$\rho = f(\hat{\theta})$$

LOG LIKELIHOOD SURFACE

Likelihood $P(y \mid \theta)$

$$y \sim Normal(\alpha + \beta x, \sigma)$$

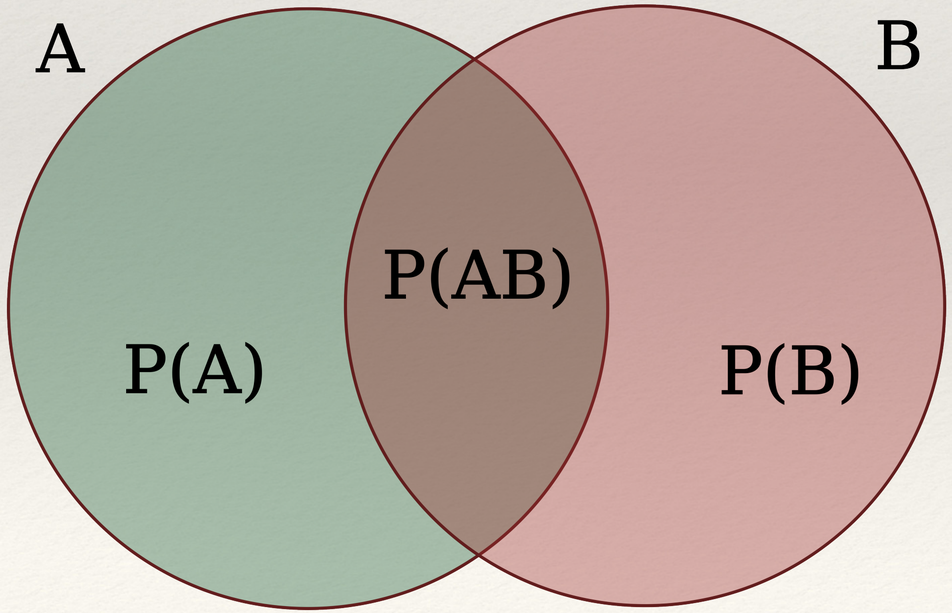


INTO THE BAYES WORLD

REV THOMAS BAYES (1701 – 1761)

Product rule (or Bayes Theorem)

$$P(AB) = P(A \mid B)P(B) = P(B \mid A)P(A)$$



WHAT IS BAYESIAN STATISTICS?

We can think of Bayesian Statistics as an extension of ML

ML: What is the parameter value that maximizes the probability of having generated the data:

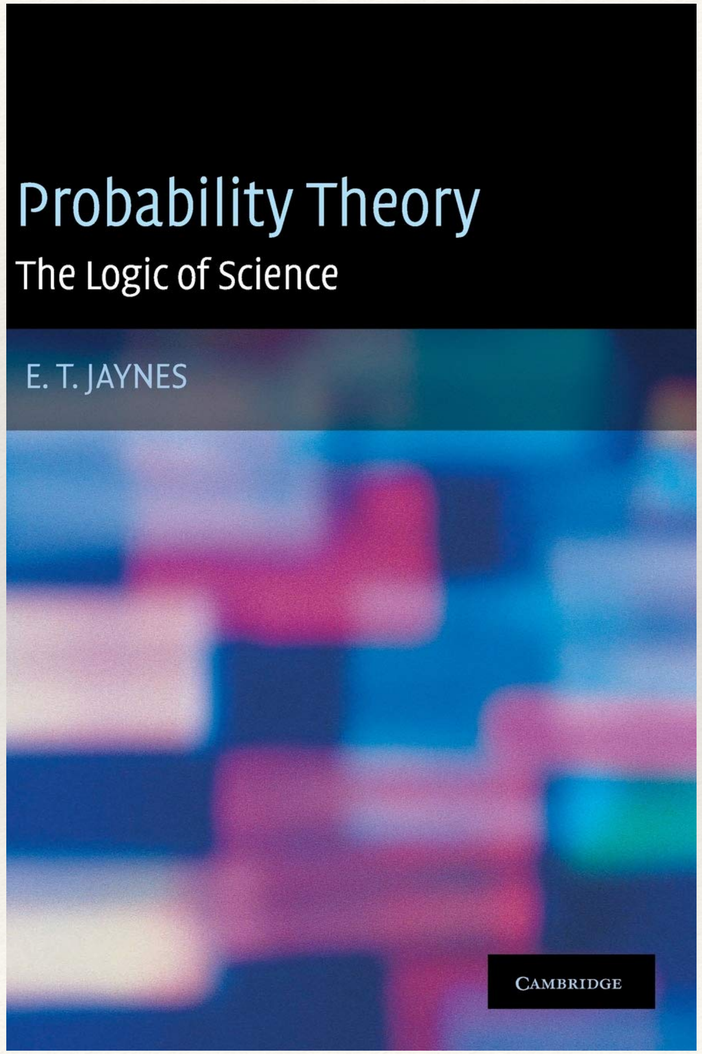
$$\underset{\theta}{\operatorname{argmax}} \left[P(y \mid \theta) \right]$$

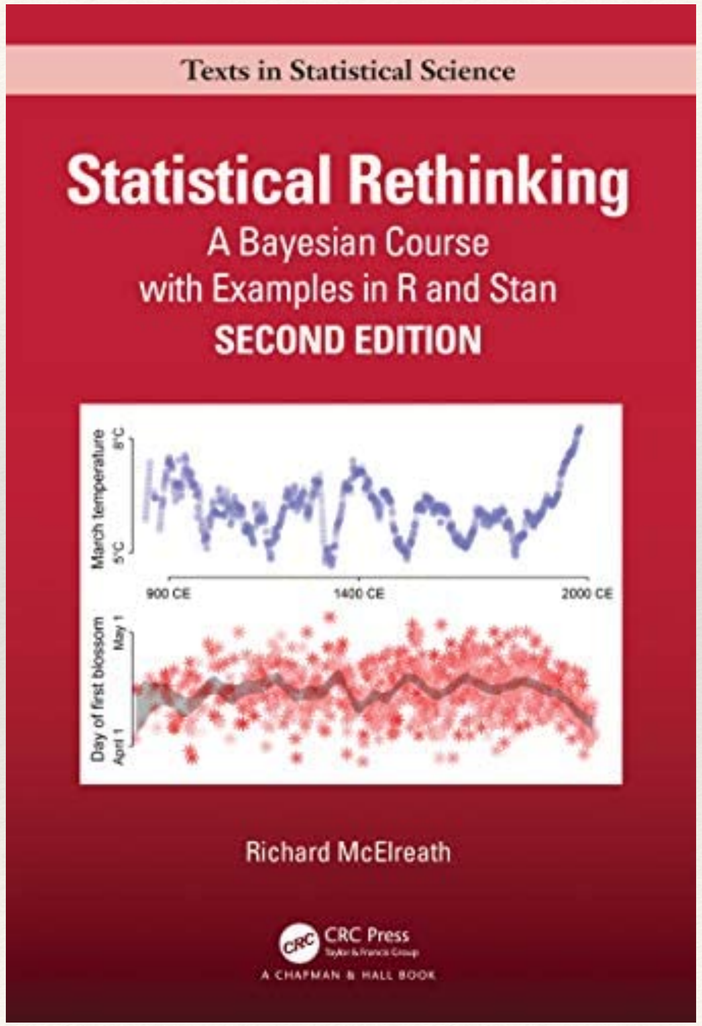
Bayesian: What is the probability distribution of parameter values given the data:

$$P(\theta \mid y) \propto P(\theta) P(y \mid \theta)$$

Attention!
No maximization!

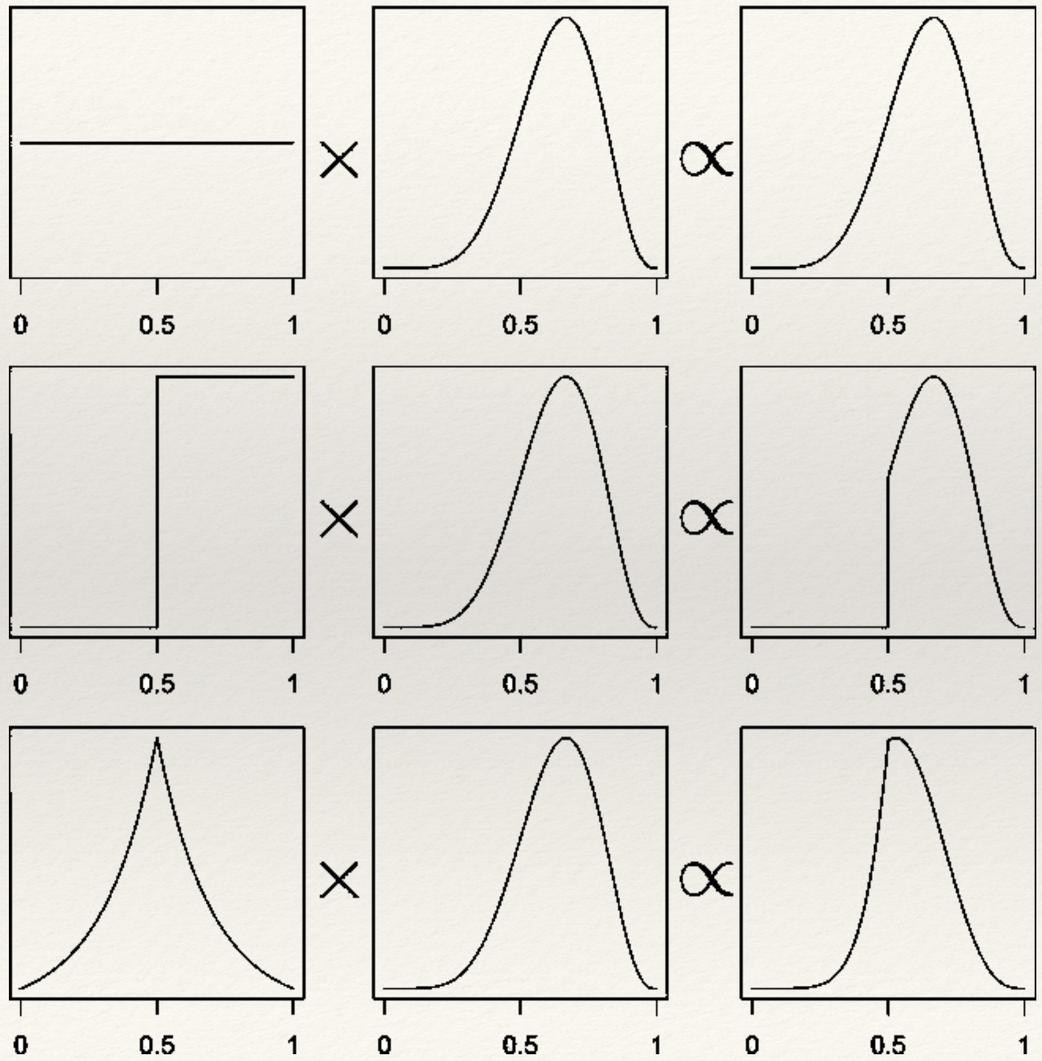
SOME GOOD REFERENCES FOR THE CONCEPTUAL PART





PRIOR × LIKELIHOOD ∝ POSTERIOR

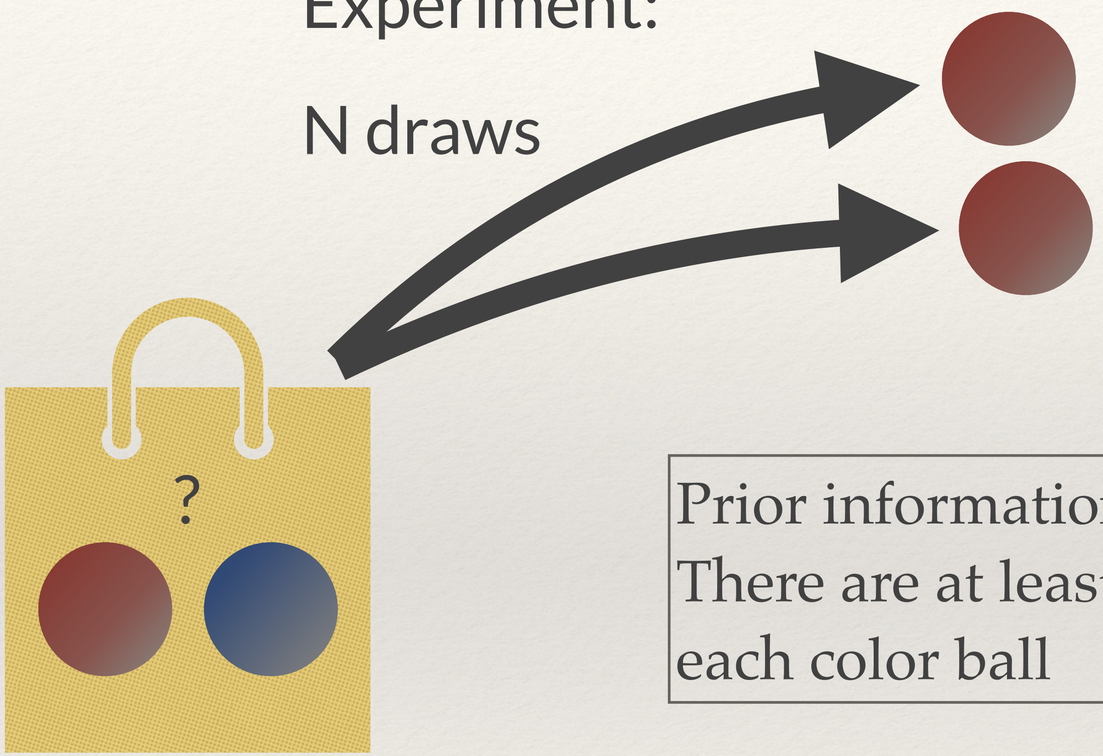
$$P(A \mid B) = \frac{P(B \mid A)P(A)}{P(B)}$$



ESTIMATING THE PROPORTION OF RED BALLS

Experiment:

N draws



Data:

y = # red balls

y = 2

ML estimator:

$$p = \frac{y}{N} = \frac{2}{2} = 1$$

Prior information:
There are at least one of each color ball

Laplace estimator:

$$p = \frac{y+1}{N+2} = \frac{2+1}{2+2} = \frac{3}{4}$$

$p = P(\bullet) =$ proportion of red balls

"Before the experiment, I observed one of each color ball"

WHY USE THE POSTERIOR?

- Allows us to use probability in more contexts.
- $P(\theta \mid y)$ represents our knowledge of parameters using probability.
    - this representation fully encapsulates our beliefs.
- $P(\theta)$, the prior, can encode useful information.
    - parameter scale, shared structure, permitted values…
- Isn't the MLE the best estimator? (depends on the criteria...)
    - Sometimes... but not $\rho = f(\hat{\theta})$
- Expands the range of models we can fit

USING THE POSTERIOR

POSTERIOR ESTIMATORS

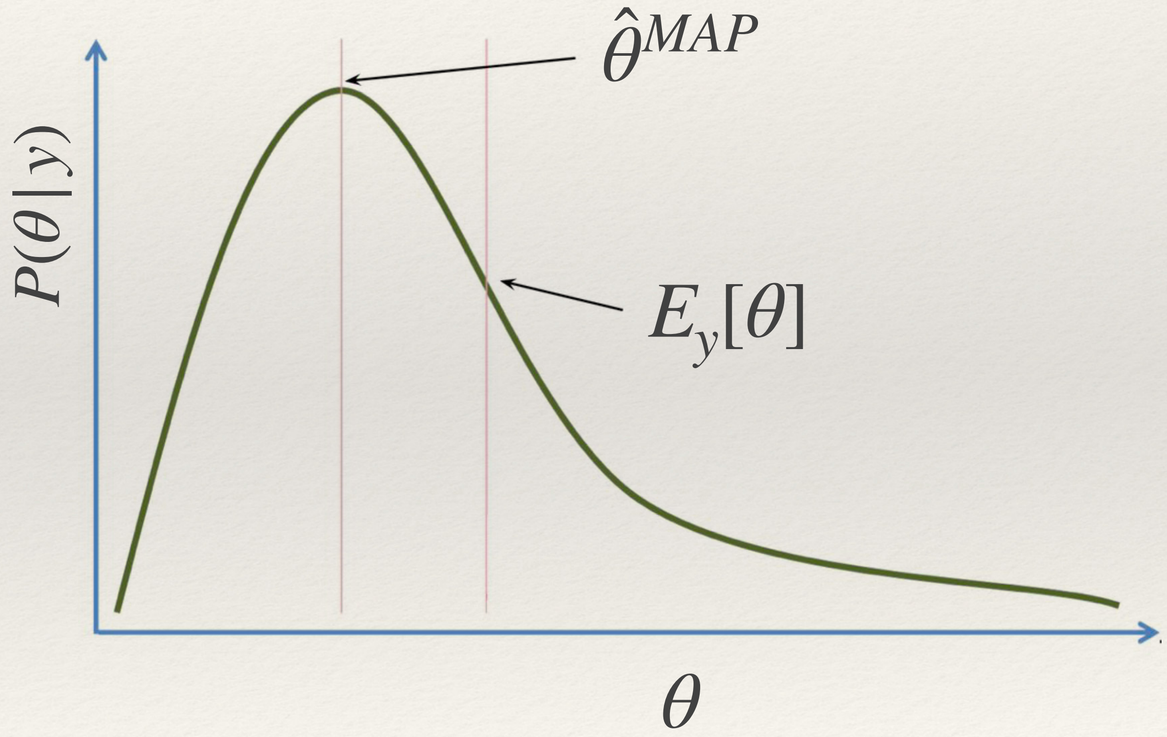
- Bayesian equivalent to MLE is the **M**aximum **A** **P**osteriori (MAP):

$$\hat{\theta}^{MAP} = \underset{\theta \in \Omega}{\operatorname{argmax}} \left[P(\theta \mid y) \right]$$

- The posterior mean is more common:

$$E_y[\theta] = \sum_{\theta \in \Omega} \theta P(\theta \mid y)$$

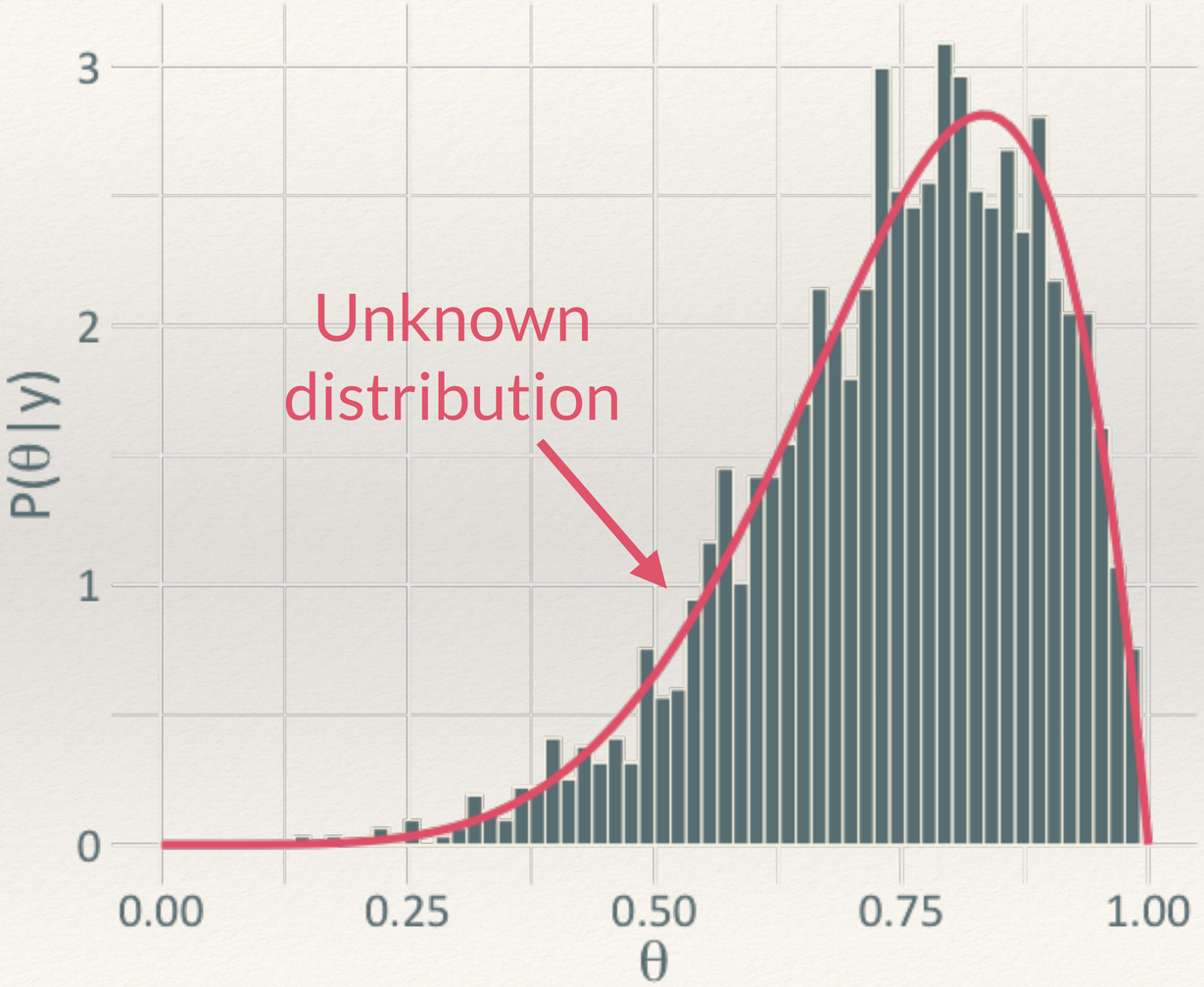
(Posterior median is also used occasionally.)



POSTERIOR APPROXIMATIONS

- For a small number of models we can write the posterior distribution directly (really small, don't bother).
- For most models, we use posterior samples to approximate the posterior.

$$\{\theta_1, \cdots, \theta_N\} \sim P(\theta \,|\, y)$$



POSTERIOR DERIVED QUANTITIES

- This sample can be used to calculate any quantity of interest:

$$\{\theta_1, \cdots, \theta_N\} \sim P(\theta \mid y)$$

For example, the posterior mean is just:

$$\frac{\theta_1 + \theta_2 + \cdots + \theta_N}{N} \approx \sum_{\theta \in \Omega} \theta P(\theta \mid y)$$

Other quantities

- Any other functions of the parameters can be estimated from the samples.
- A common use is to calculate contrast between categorical levels, estimating the difference between groups.
- Quantiles, values above a value, confidence intervals…

BUILDING A MODEL

My first bayesian regression model

- Given the matched pairs:

$$(x, y) = \{(x_1, y_1), \cdots, (x_n, y_n)\}$$

- Define a likelihood:

$$\left.\begin{aligned} y_i &\sim Normal(\mu_i, \sigma) \\ \mu_i &= \alpha + \beta x_i \end{aligned}\right\} P(y \mid \theta)$$

- And a set of priors on the parameters:

$$\left.\begin{aligned} \alpha &\sim P(\alpha) \\ \beta &\sim P(\beta) \\ \sigma &\sim P(\sigma) \end{aligned}\right\} P(\theta)$$

HOW DO WE CHOOSE THE PRIORS?!

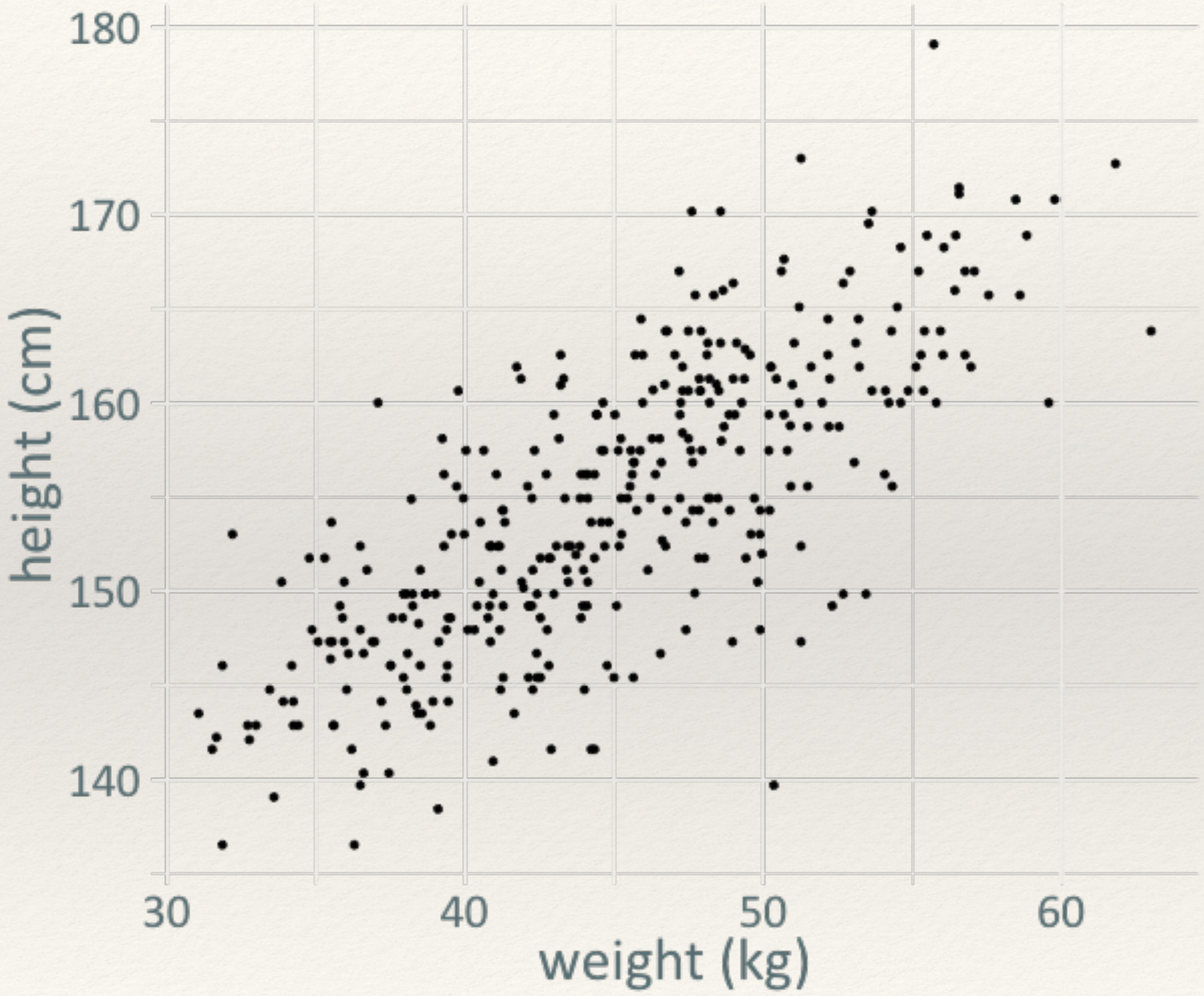
$$\alpha \sim P(\alpha)\,?$$
$$\beta \sim P(\beta)\,?$$
$$\sigma \sim P(\sigma)\,?$$

- Agnostic choices
  - Laplace and the Principle of indifference
  - "Uninformative" priors
- Maximum entropy priors
  - priors that encode the least amount of information given constraints
- Jeffreys priors
  - invariant under a change of coordinates
- Hard constraints
  - restricted domains (e.g. variance must be positive)

Good prior choices

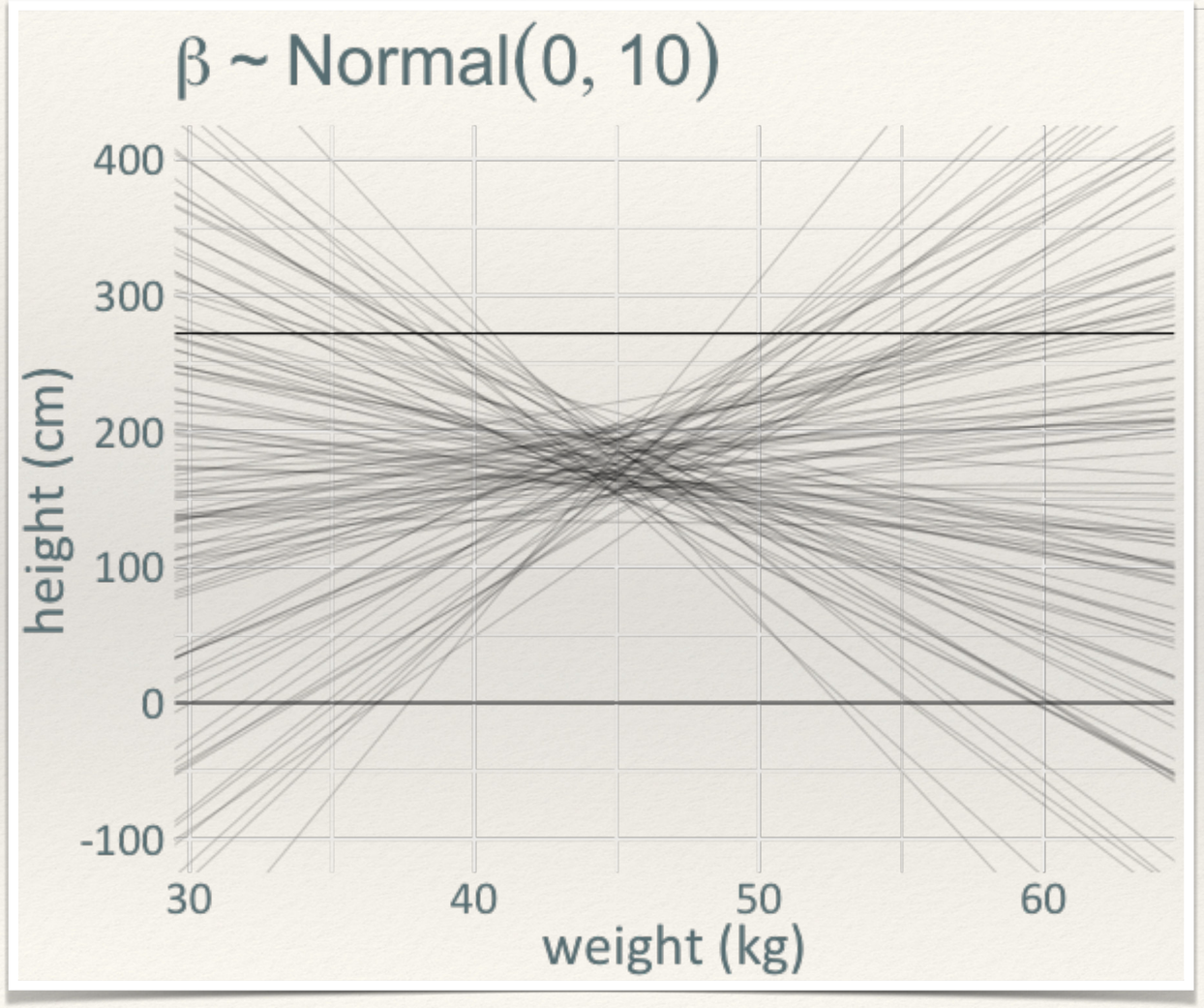
- Use domain expertise!
- Knowledge of scale (height by weight example)
- Experimental design (more in the hierarchical models class)
- Using simulations to understand the implications of priors

PRIORS CAN BE USED TO ENCODE SCALE INFORMATION

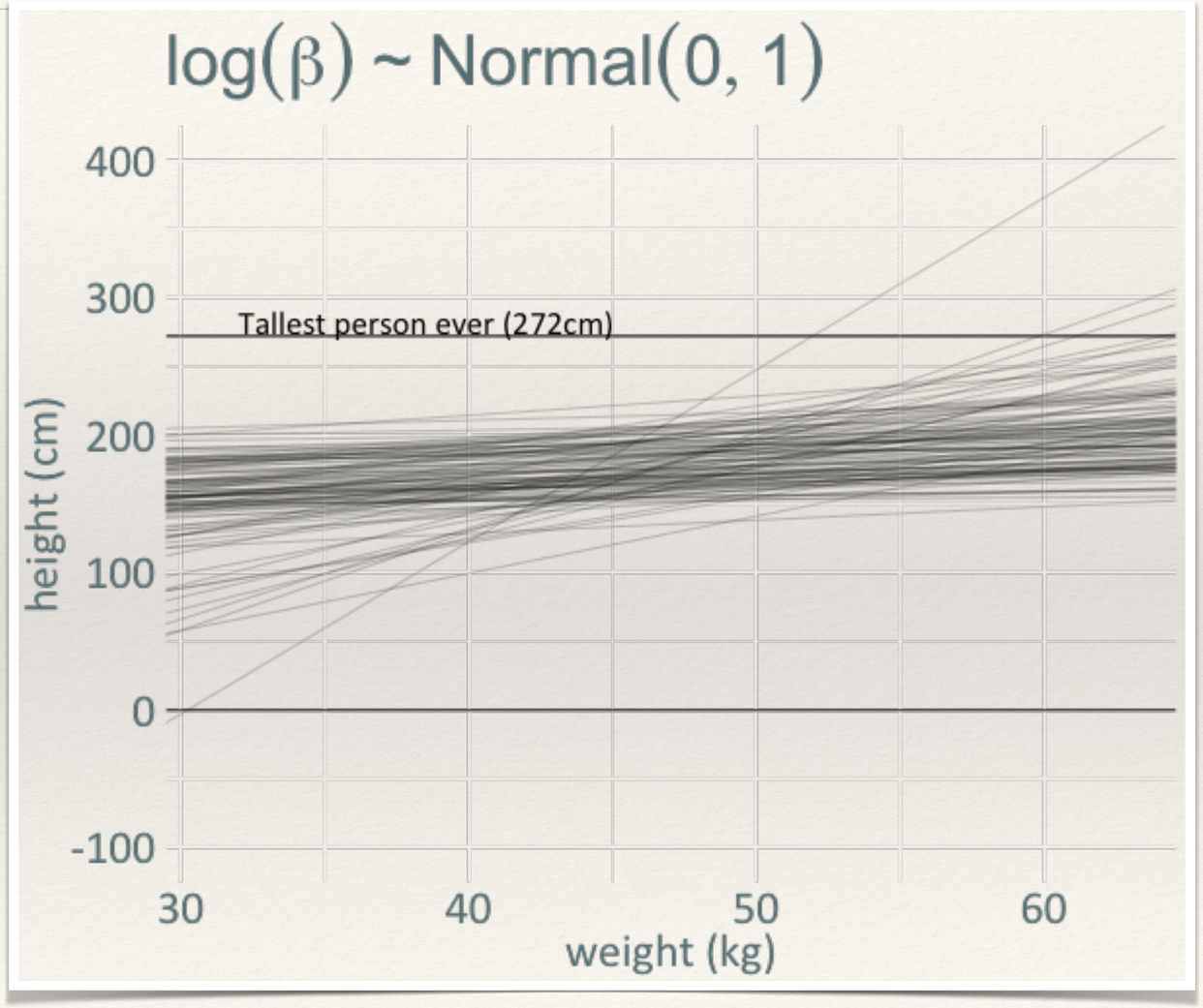


Adapted from Statistical Rethinking

WIDE VS NARROW PRIOR



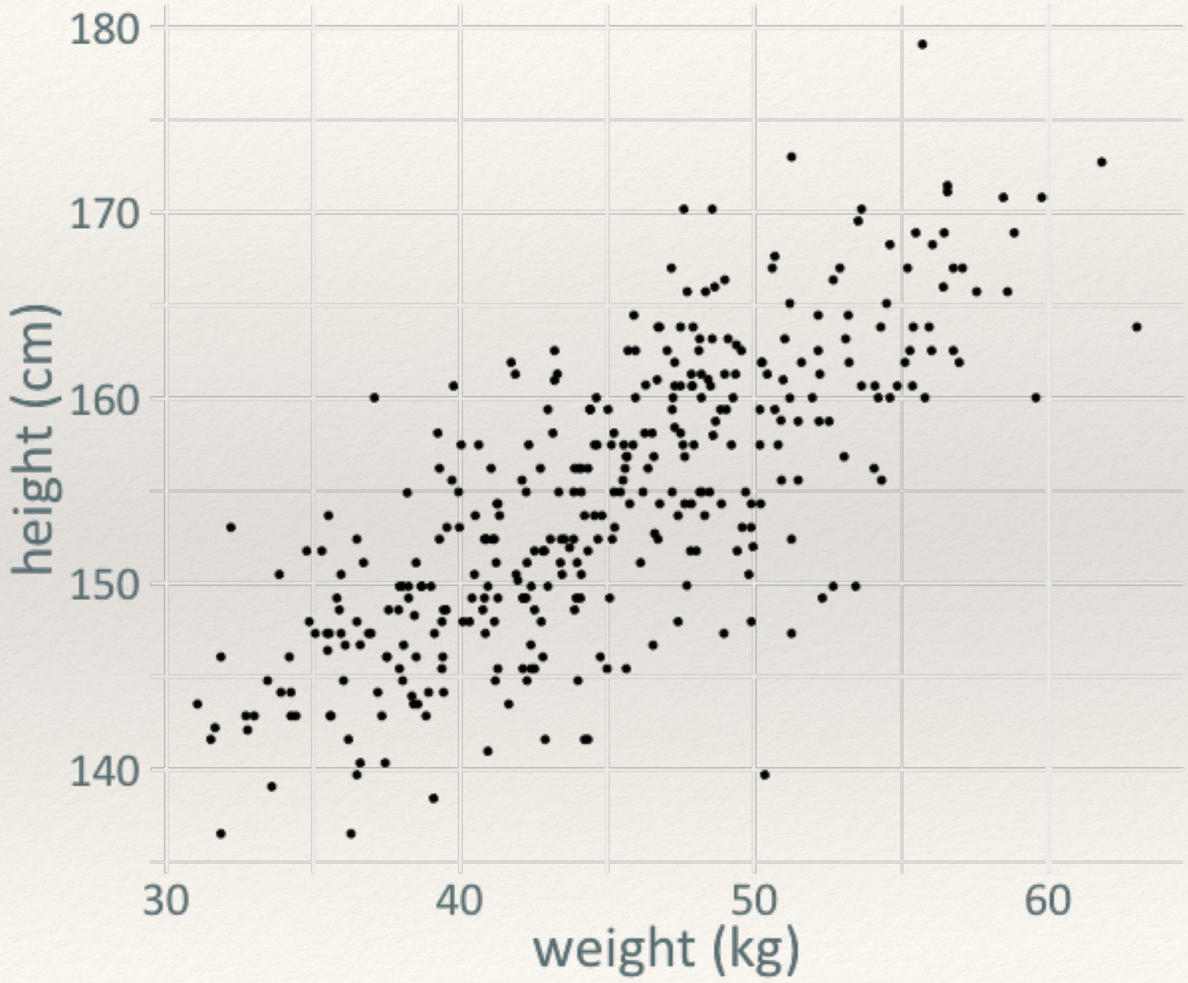
This is sometimes called a non-informative prior



This prior is informative, but in a good way!

OUR MODEL FOR THE HEIGHT DATA

$$y_i \sim Normal(\mu_i, \sigma)$$
$$\mu_i = \alpha + \beta x_i$$
$$\alpha \sim Normal(0, 20)$$
$$\beta \sim lognormal(0, 1)$$
$$\sigma \sim Exponential(1)$$



POSTERIOR SAMPLES

$$y_i \sim Normal(\mu_i, \sigma)$$
$$\mu_i = \alpha + \beta x_i$$
$$\alpha \sim Normal(0, 20)$$
$$\beta \sim lognormal(0, 1)$$
$$\sigma \sim Exponential(1)$$

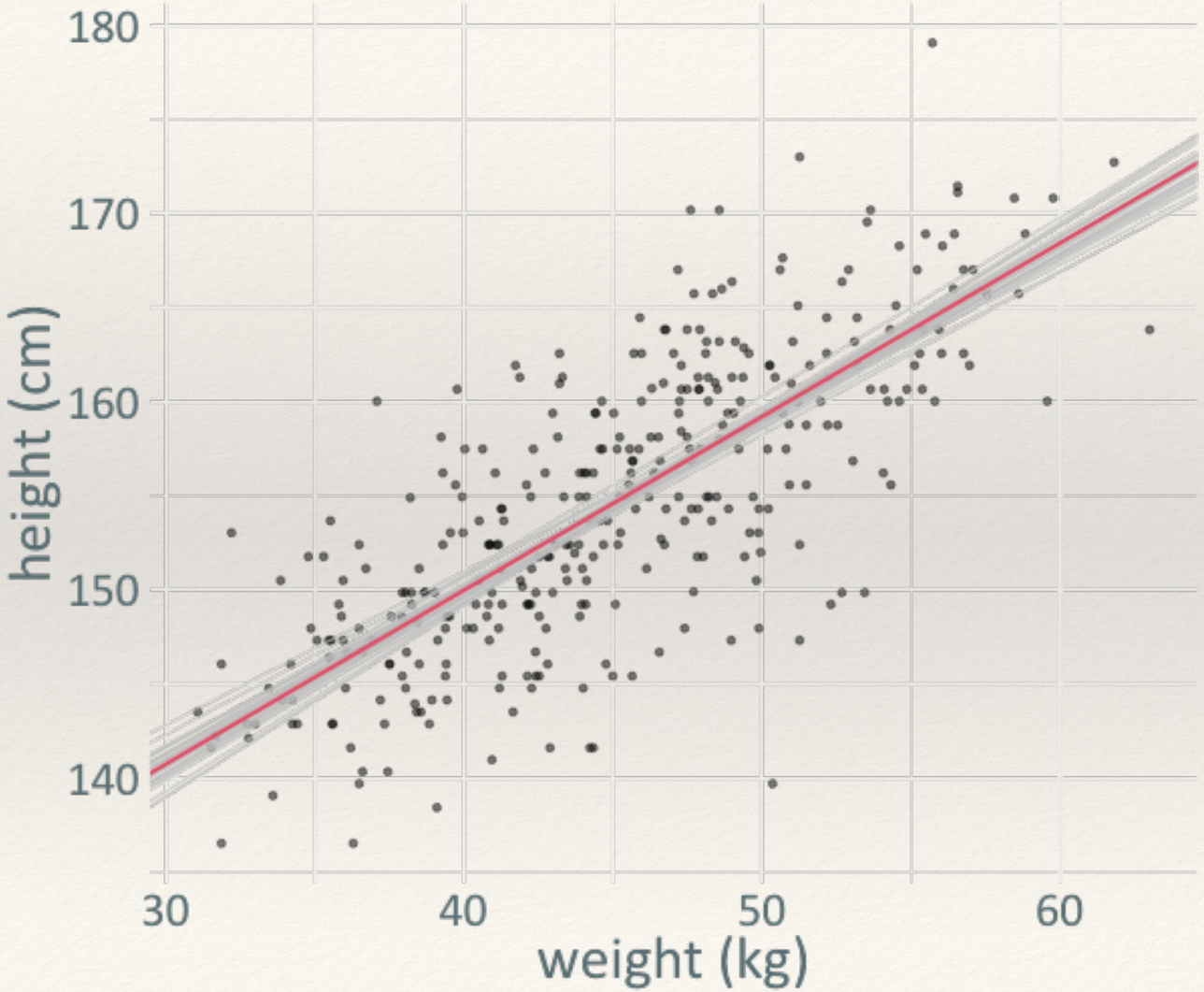

```
> samples
# A tibble: 2,000 × 3
          a         b     sigma
  <dbl[1d]> <dbl[1d]> <dbl[1d]>
1      115.     0.889      4.78
2      109.     1.02       5.30
3      112.     0.928      5.07
4      111.     0.949      5.30
5      111.     0.955      5.04
6      115.     0.872      5.19
7      109.     1.01       5.13
8      117.     0.844      5.00
9      115.     0.882      4.94
10     112.     0.939      4.95
# … with 1,990 more rows
```

POSTERIOR MEAN ESTIMATES

$$E_y[\theta] =$$

```
> colMeans(samples)
         a          b      sigma
112.9296580  0.9253803  5.0453651
```

MODEL FIT

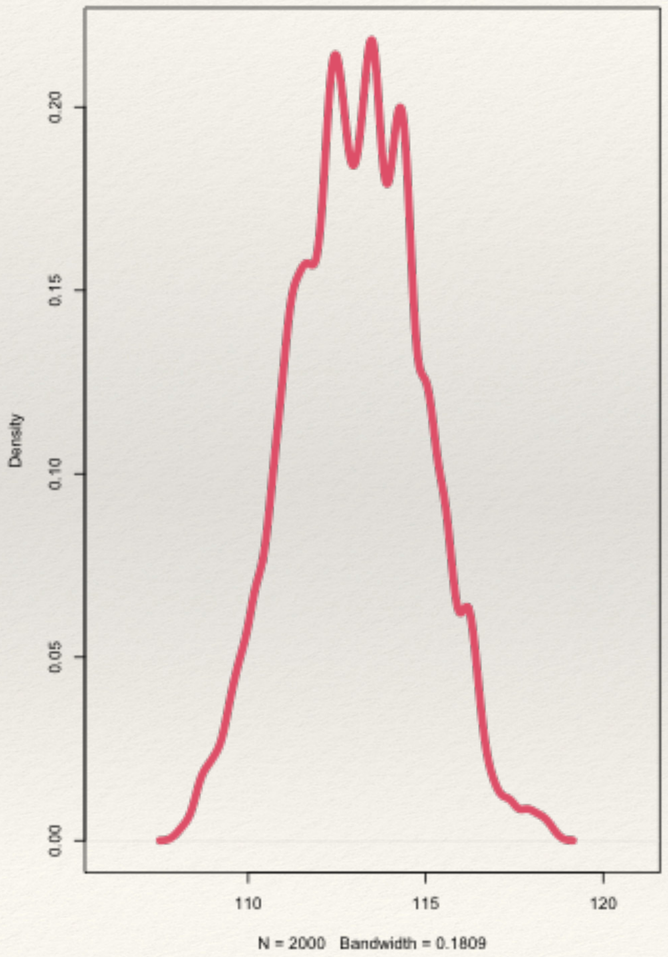



```
> colMeans(samples)
         a          b      sigma
112.9296580  0.9253803  5.0453651
```

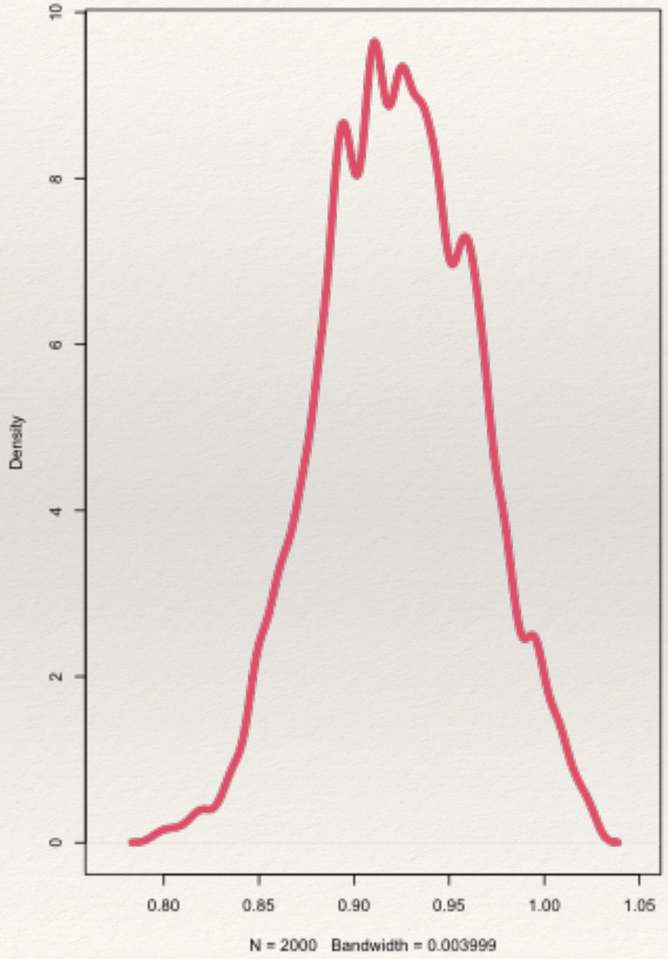
```
> samples
# A tibble: 2,000 × 3
      a     b sigma
  <dbl> <dbl> <dbl>
1  115.  0.889  4.78
2  109.  1.02   5.30
3  112.  0.928  5.07
4  111.  0.949  5.30
5  111.  0.955  5.04
6  115.  0.872  5.19
7  109.  1.01   5.13
8  117.  0.844  5.00
9  115.  0.882  4.94
10 112.  0.939  4.95
# … with 1,990 more rows
```

THE POSTERIOR PARAMETER DISTRIBUTION

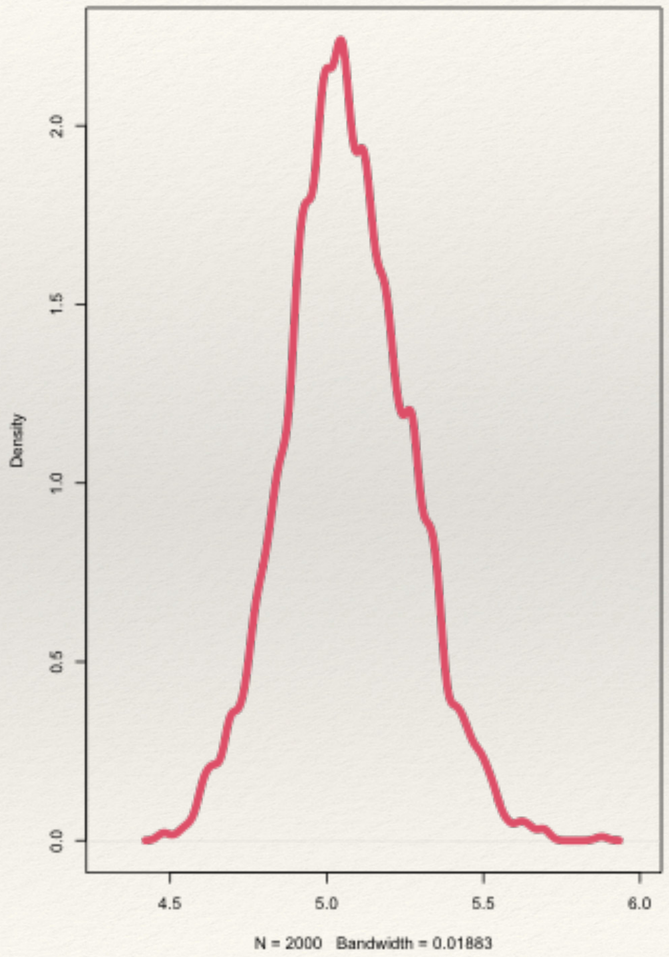
a



b



sigma




```
> samples
# A tibble: 2,000 × 3
       a     b sigma
   <dbl> <dbl> <dbl>
 1  115. 0.889  4.78
 2  109. 1.02   5.30
 3  112. 0.928  5.07
 4  111. 0.949  5.30
 5  111. 0.955  5.04
 6  115. 0.872  5.19
 7  109. 1.01   5.13
 8  117. 0.844  5.00
 9  115. 0.882  4.94
10  112. 0.939  4.95
# … with 1,990 more rows
```

CATEGORIAL PREDICTORS AND CONTRASTS

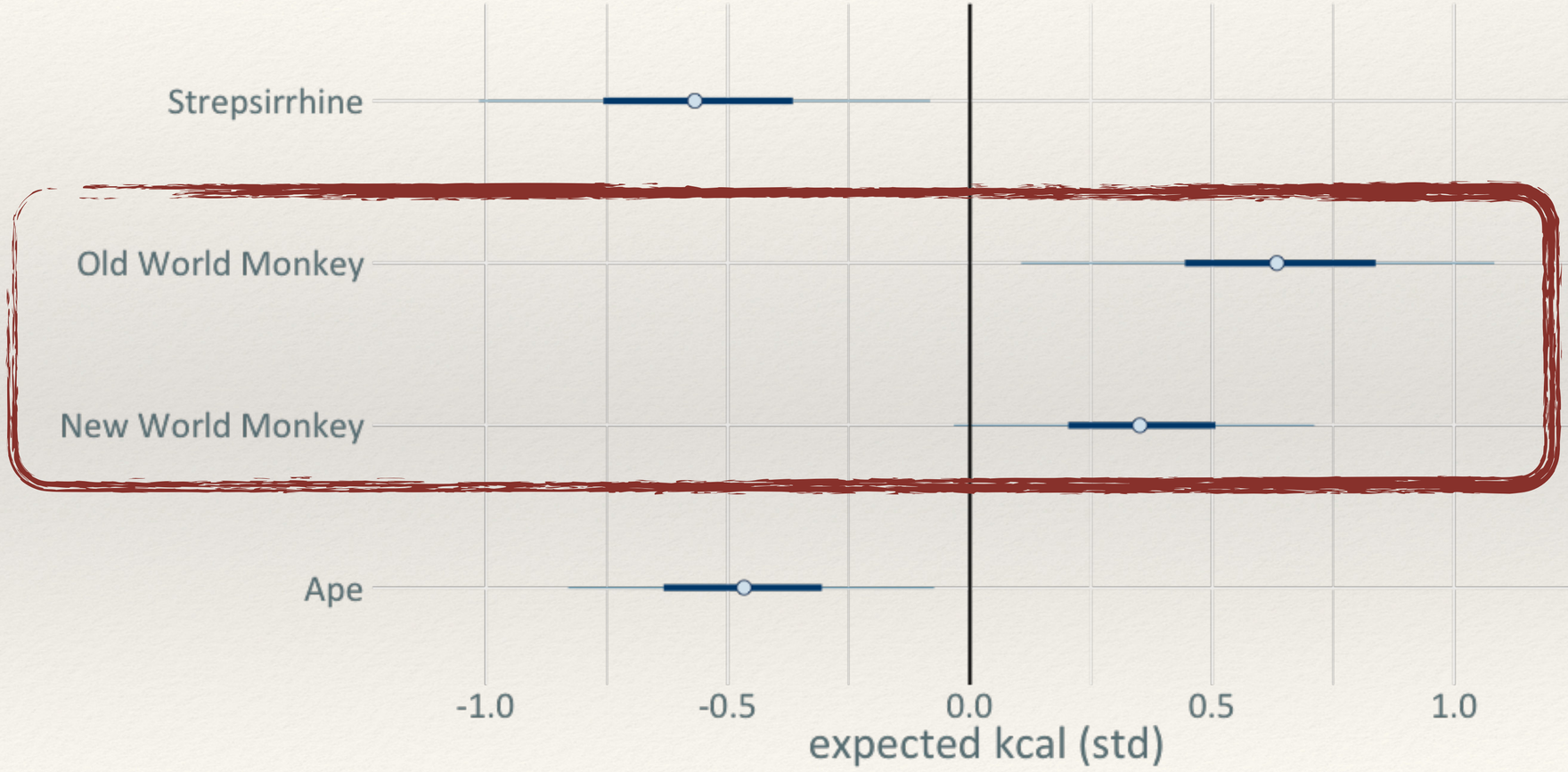
We can also use categorical predictors to estimate per-group averages.

- K_i: caloric content of milk in several monkey groups
- $CLADE$: categorical variable for the monkey groups

$$
\begin{aligned}
K_i &\sim Normal(\mu_i, \sigma) \\
\mu_i &= \alpha_{CLADE[i]} \\
\alpha_i &\sim Normal(0, 0.5) \\
\sigma &\sim Exponential(1)
\end{aligned}
$$

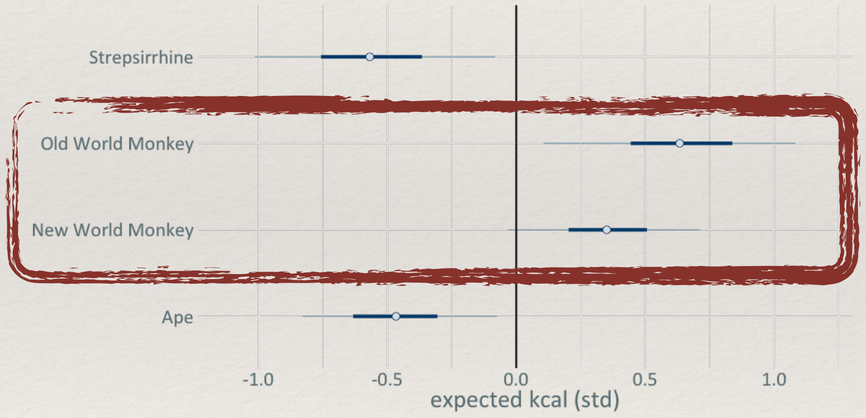
PER CLADE MILK CONTENT

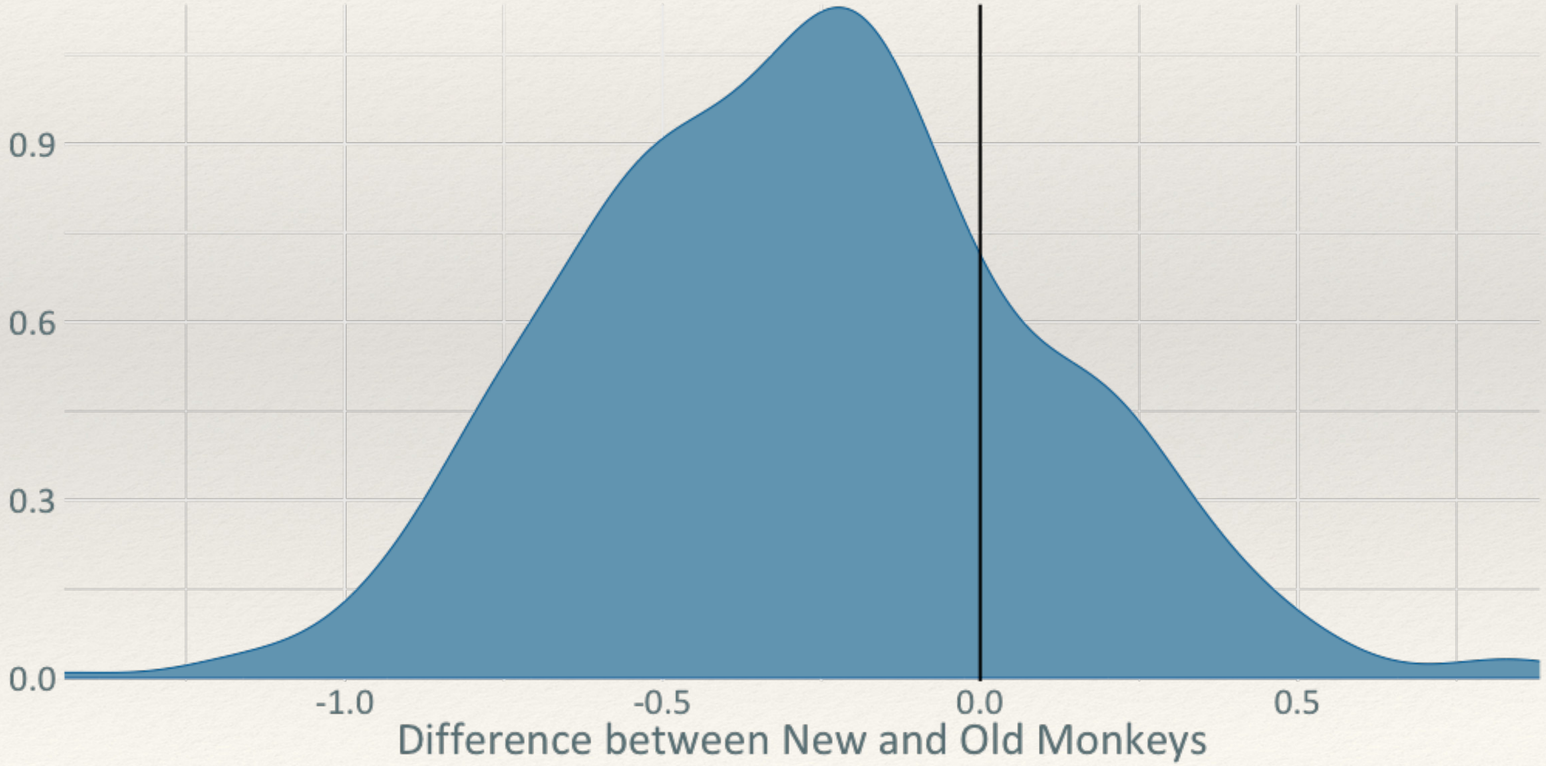
$$K_i \sim Normal(\mu_i, \sigma)$$
$$\mu_i = \alpha_{CLADE[i]}$$
$$\alpha_i \sim Normal(0, 0.5)$$
$$\sigma \sim Exponential(1)$$



CONTRASTS

To compare coefficient estimates we must look at the distribution of differences.





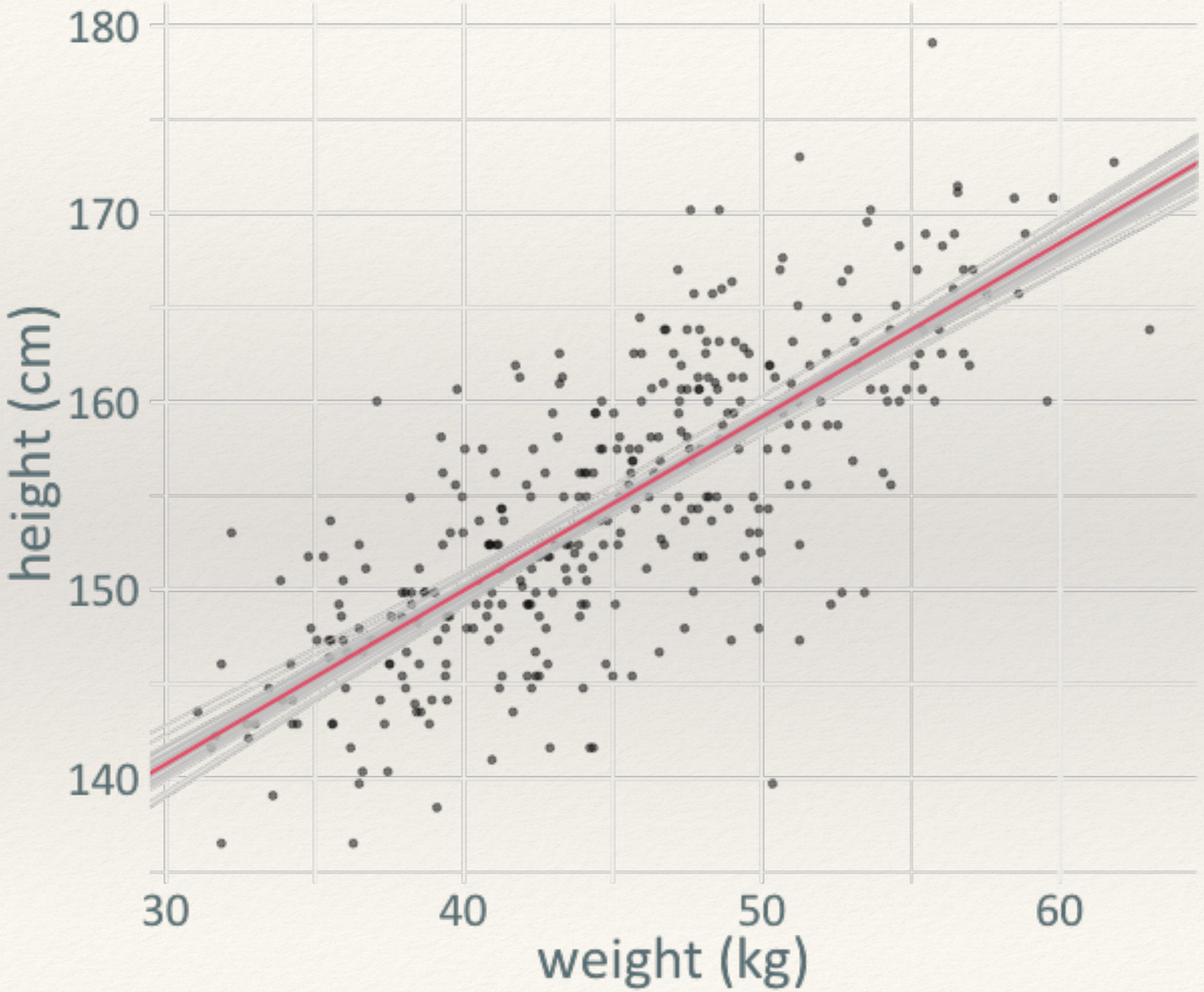
INTERMISSION

PROBABILISTIC MODELING

Fitting Bayesian models and working with the Posterior

Diogo Melo
Lewis-Sigler Institute of Integrative Genomics
damelo@princeton.edu

MODEL FIT




```
> colMeans(samples)
         a          b      sigma
112.9296580  0.9253803  5.0453651
```

```
> samples
# A tibble: 2,000 × 3
      a     b sigma
  <dbl> <dbl> <dbl>
1  115.  0.889  4.78
2  109.  1.02   5.30
3  112.  0.928  5.07
4  111.  0.949  5.30
5  111.  0.955  5.04
6  115.  0.872  5.19
7  109.  1.01   5.13
8  117.  0.844  5.00
9  115.  0.882  4.94
10 112.  0.939  4.95
# … with 1,990 more rows
```

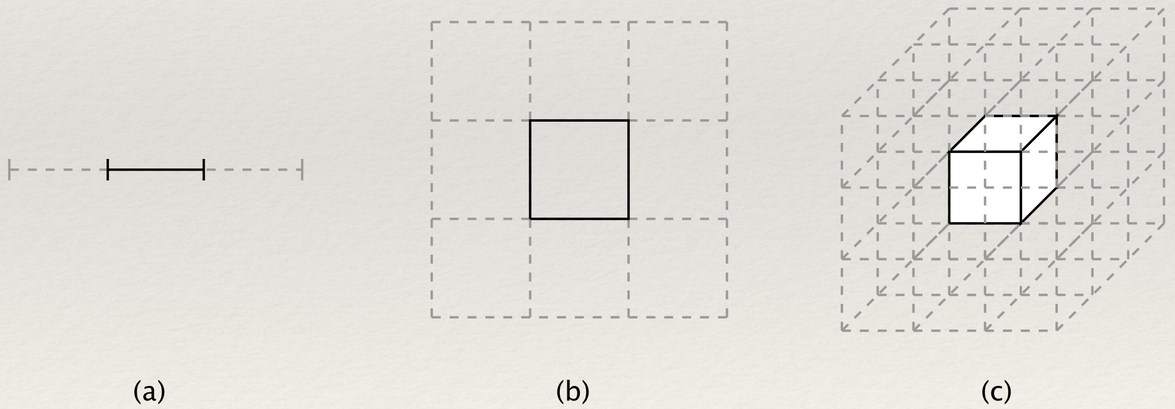
POSTERIOR SAMPLES ARE EVERYTHING!

- ML methods use estimated values for parameters for everything
  - $\rho = f(\hat{\theta})$
- Bayesian methods use the posterior distribution of the parameters for everything

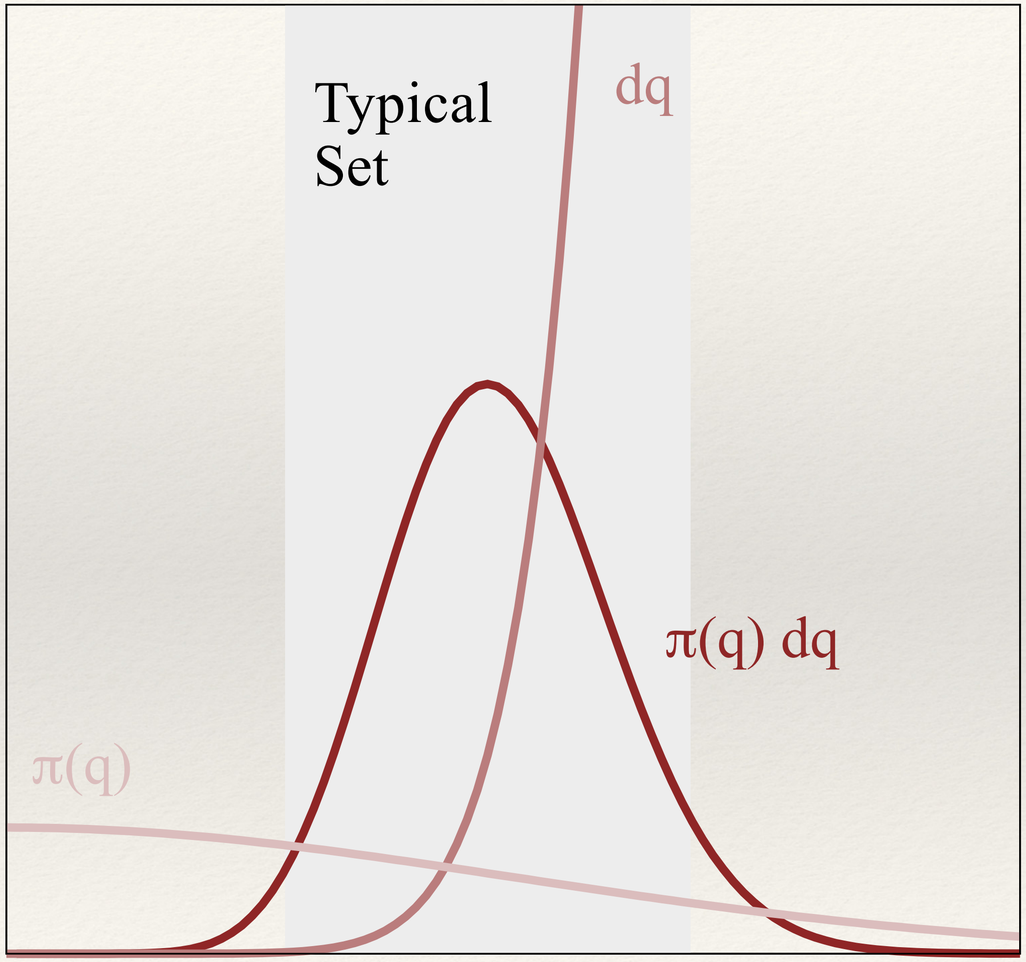
```
> samples
# A tibble: 2,000 × 3
          a         b     sigma
  <dbl[1d]> <dbl[1d]> <dbl[1d]>
1      115.     0.889      4.78
2      109.     1.02       5.30
3      112.     0.928      5.07
4      111.     0.949      5.30
5      111.     0.955      5.04
6      115.     0.872      5.19
7      109.     1.01       5.13
8      117.     0.844      5.00
9      115.     0.882      4.94
10     112.     0.939      4.95
# … with 1,990 more rows
```

HOW TO GET POSTERIOR SAMPLES?

- There is no general method to find high probability regions in arbitrary probability distributions.
- This mean most models are fit using purely computational methods.
- For simple parameters spaces, we can do grid search or some brute force method to find high probability regions
- This breaks down quickly as the number of parameters increases



TYPICAL SET



Betancourt, M. A Conceptual Introduction to Hamiltonian Monte Carlo. arXiv [stat.ME] (2017)

FINDING THE TYPICAL SET

Find a sequence of points in the parameter space that converge to the typical set:

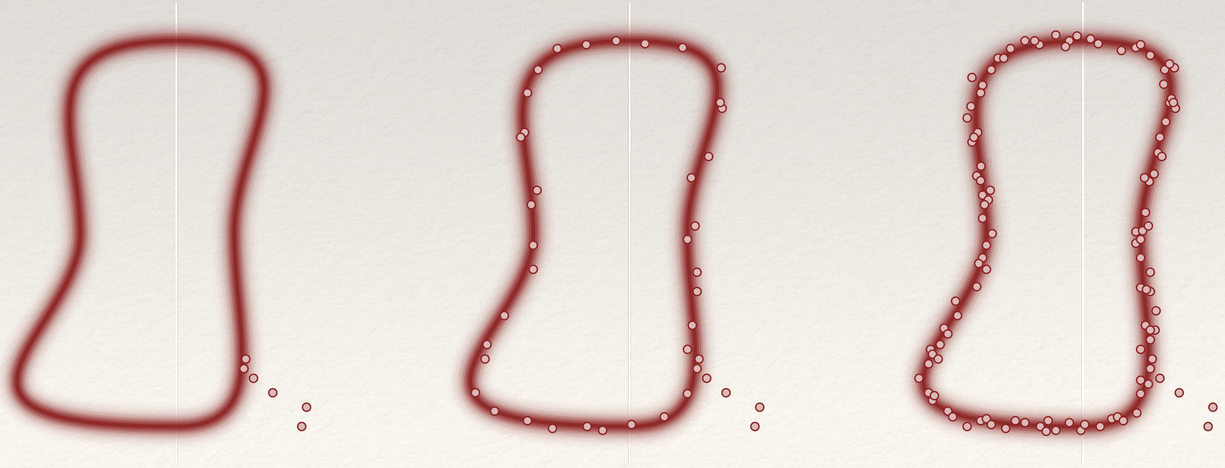
$$\theta_1 \rightarrow \theta_2 \rightarrow \theta_3 \rightarrow \theta_4 \rightarrow \cdots$$

Such that:

$$\{\theta_1, \cdots, \theta_n\} \sim P(\theta \mid y)$$

Typical set

MCMC sampling of the typical set



MCMC SAMPLERS

- Metropolis–Hastings algorithms (broad class of samplers, very general).
  - Most methods in the wild are some flavor of this.
- Reversible Jump MCMC (used in many phylogenetic packages).
  - Allows for posterior distributions with variable dimensionality.
- Usable non-mcmc methods: R-INLA - integrated nested Laplace approximation.
  - Great for structural equation modeling, much faster for some classes of models.
- Gibbs samplers.
  - Mostly surpassed, but still in wide use.
  - Can sample discrete parameters.
  - Requires particular types of priors.
  - Software: WinBugs, Bugs, Jags…
- Hamiltonian Monte Carlo samplers
  - Discrete parameters must be integrated.
  - Can fit dynamic models using differential equations.
  - Software: PyMC3, Edward, Stan (rethinking engine)...

WHAT MAKES THESE SAMPLERS DIFFERENT?

Basically the transition proposal distribution

$$P(\theta_{i+1} \mid \theta_i)$$

$$\theta_1 \rightarrow \theta_2 \rightarrow \theta_3 \rightarrow \theta_4 \rightarrow \cdots$$

We can visualize what is going on with different samplers:

https://chi-feng.github.io/mcmc-demo/app.html

OUR MODEL FROM LAST CLASS

$$y_i \sim Normal(\mu_i, \sigma)$$
$$\mu_i = \alpha + \beta x_i$$
$$\alpha \sim Normal(0, 20)$$
$$\beta \sim lognormal(0, 1)$$
$$\sigma \sim Exponential(1)$$

```
# Data
library(rethinking)
d2 <- Howell1[ Howell1$age >= 18 , ]

# Model
ulam(alist(
  y ~ normal(mu, sigma),
  mu <- a + b * x,
  a ~ normal(0, 20),
  b ~ lognormal(0, 1),
  sigma ~ exponential(1)),
  data = list(y = d2$height,
              x = d2$weight),
  iter = 1000, chains = 4, cores = 4)
```

RETHINKING GENERATES STAN CODE

```
data{
    vector[352] y;
    vector[352] x;
}
parameters{
    real a;
    real<lower=0> b;
    real<lower=0> sigma;
}
model{
    vector[352] mu;
    sigma ~ exponential( 1 );
    b ~ lognormal( 0 , 1 );
    a ~ normal( 0 , 20 );
    for ( i in 1:352 ) {
        mu[i] = a + b * x[i];
    }
    y ~ normal( mu , sigma );
}
```

Stan

https://mc-stan.org/

Stan Dev

https://github.com/stan-dev/stan

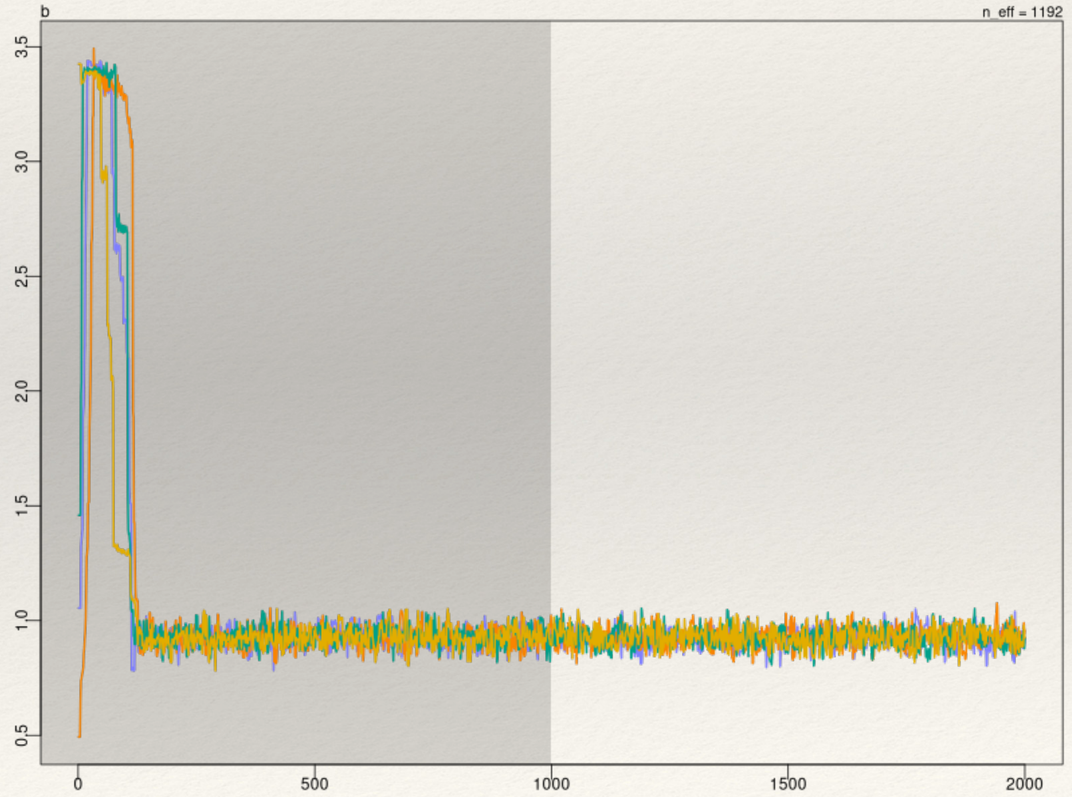
SAMPLER ARGUMENTS

- Chains: fit the model several times
- Cores: do the fits in parallel
- Iterations: how many total samples
- Warm-up (or burn-in): starting samples that get discarded

Model summary

```
> precis(fit)
          mean   sd   5.5%  94.5% n_eff Rhat4
a       112.97 2.00 109.83 116.19  1013  1.01
b         0.92 0.04   0.86   0.99  1004  1.01
sigma     5.08 0.20   4.78   5.40  1543  1.00
```

Chains and convergence



MODEL CHECKING

After fitting the model, we can use the posterior to simulate synthetic data and compare to the data used to fit the model. Discrepancies can suggest paths to improve the model.

$$y_{sim} \sim P(y_{sim}|y) = P(y_{sim}|\theta)P(\theta|y)$$

For each value of the parameters ($\theta_i = \{a_i, b_i, \sigma_i\}$) we can simulate a synthetic dataset y_{sim} and compare to the observed data y.

$$y_{sim} = Normal(a_i + b_i x, \sigma_i)$$

STEP BY STEP FOR POSTERIOR SIMULATIONS

1. Extract the **posterior samples** for the parameters a, b, σ from the fitted model.

2. For each set of parameter values (a_i, b_i, σ_i):

 - Compute the predicted outcome: $y_{pred} = a + bx$.

 - Add random noise to y_{pred}, where the noise is drawn from a normal distribution with mean 0 and standard deviation σ_i. This gives the synthetic data y_{sim}.

3. Compare the synthetic data y_{sim} to the observed data y.

 - Compute summary statistics (e.g., mean, variance, **quantiles**) for both y_{sim} and y.

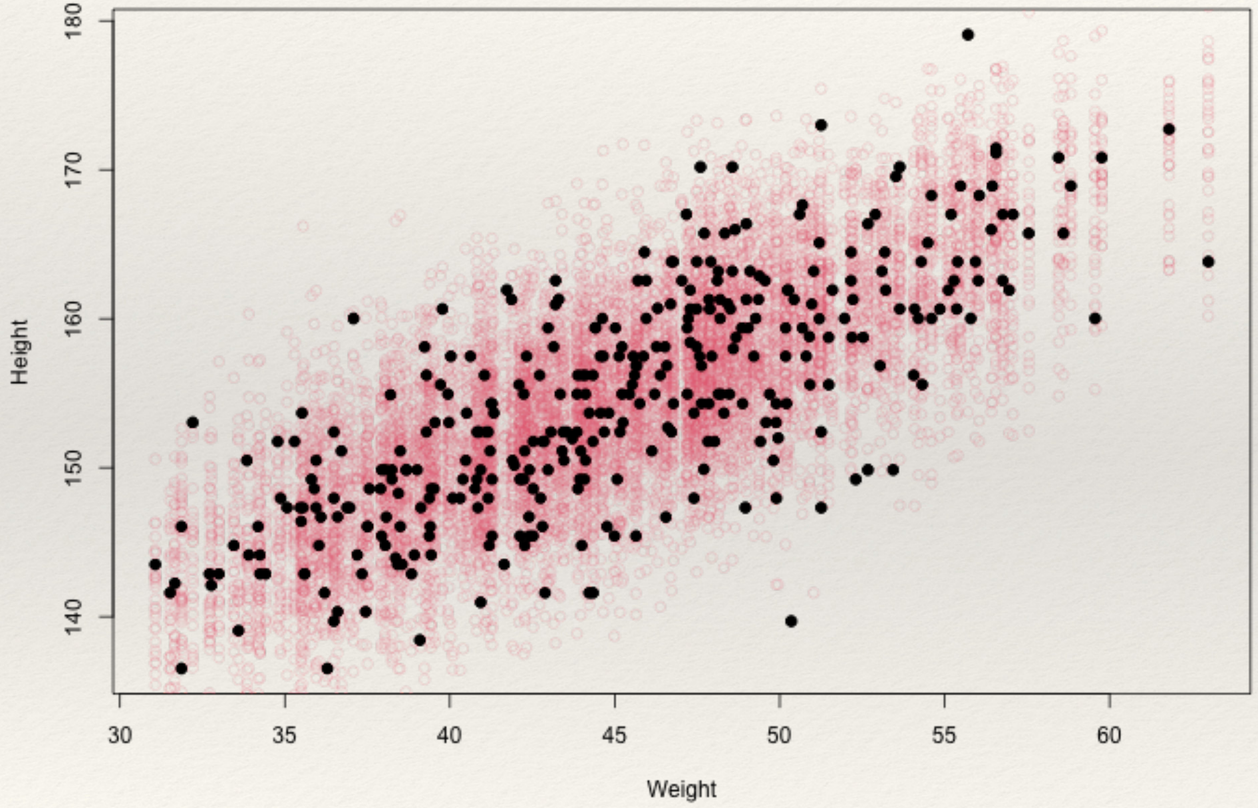
 - If the summary statistics are similar for y_{sim} and y, this suggests that the model is a good fit to the data.

4. Repeat steps 2-3 for all sets of parameter values to get a distribution of summary statistics for the synthetic data.

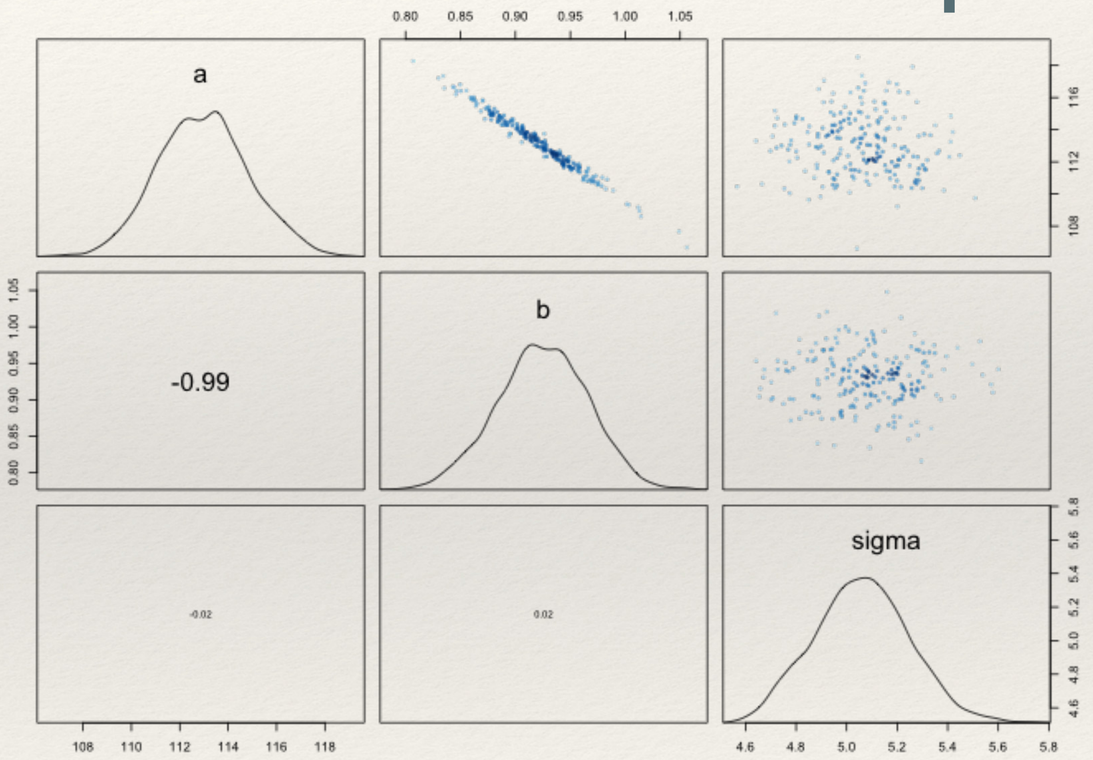
5. Compare the distribution of summary statistics for the synthetic data to the corresponding summary statistics for the observed data. If they are similar, this suggests that the model is a good fit to the data. If they are not similar, this suggests that the model may need to be improved.

MODEL CHECK

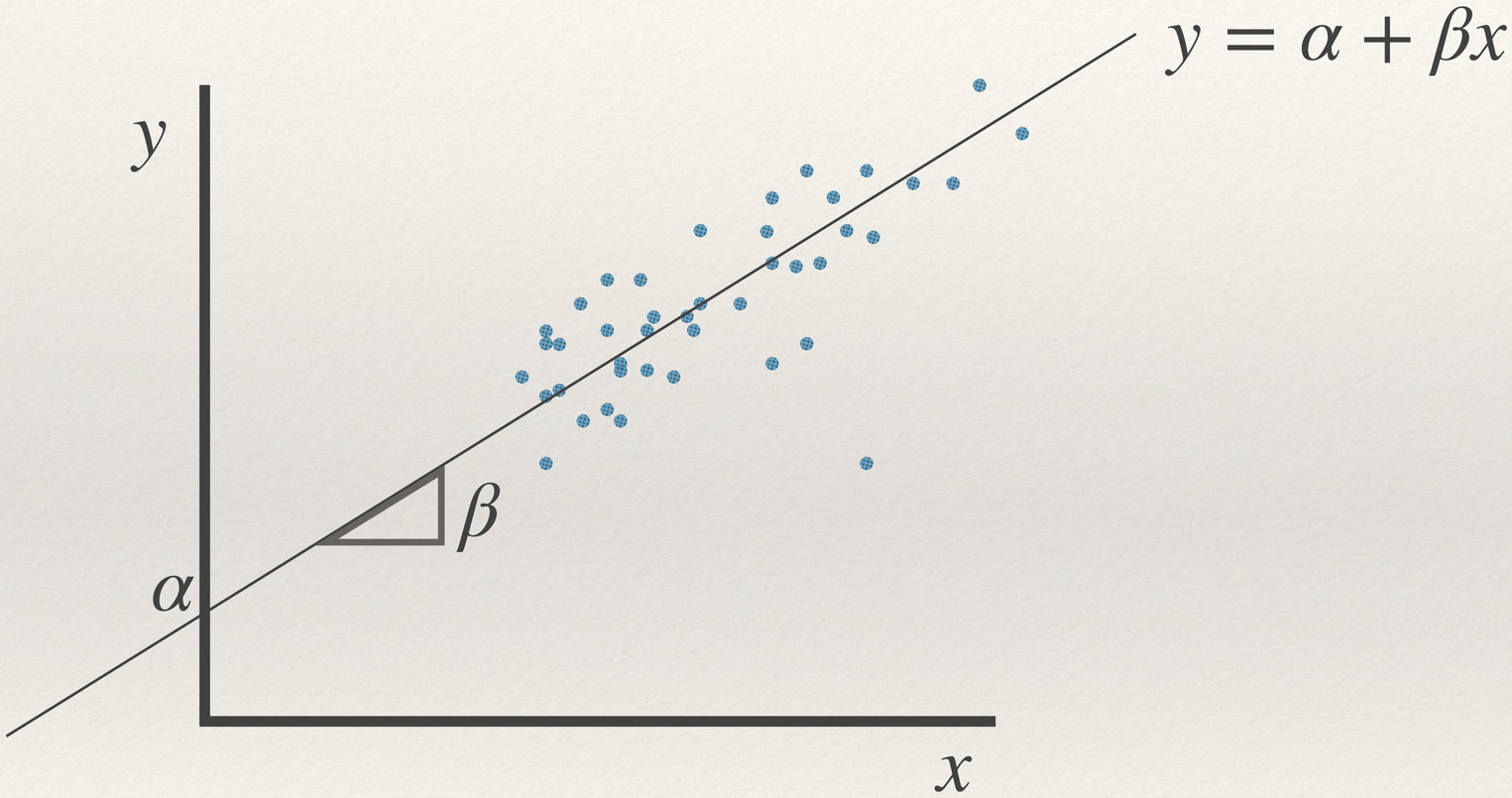
Posterior simualtions



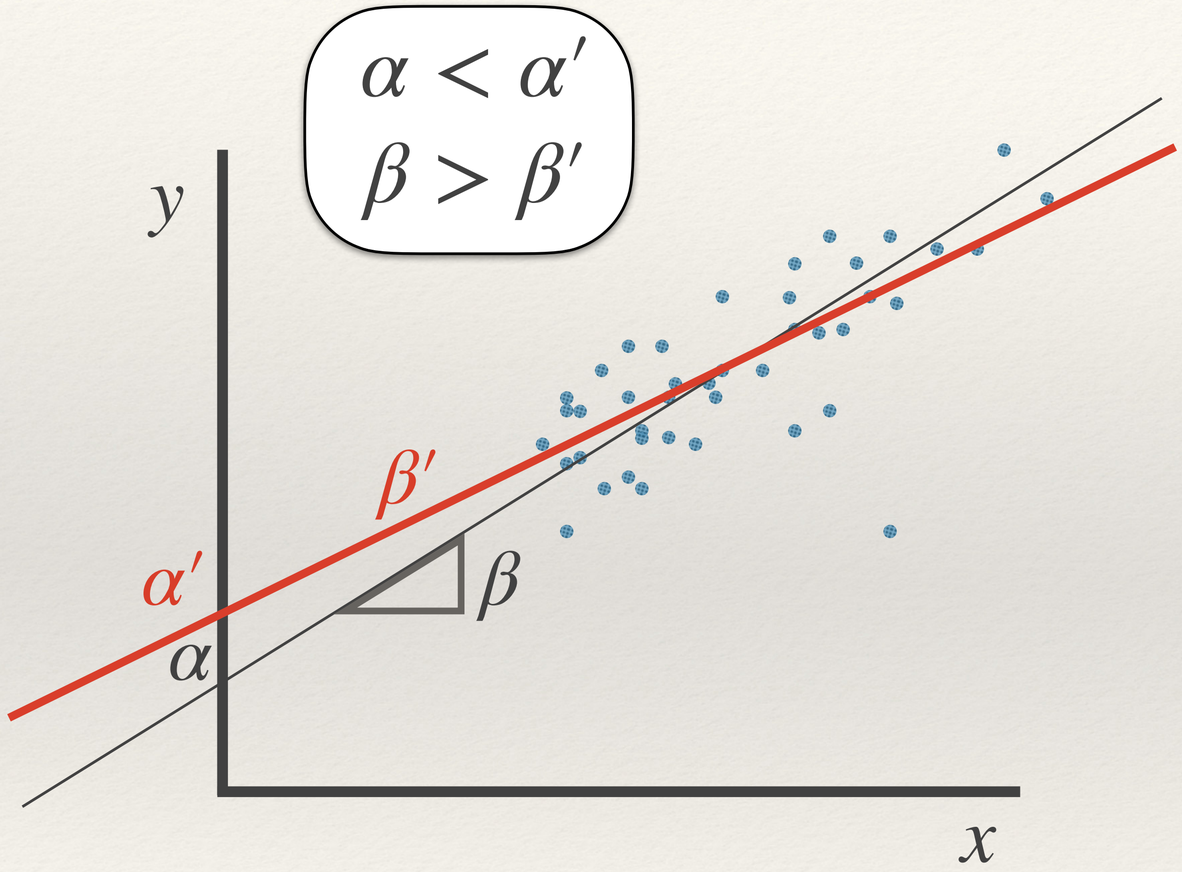
Pairs plot



SLOPE-INTERCEPT CORRELATIONS

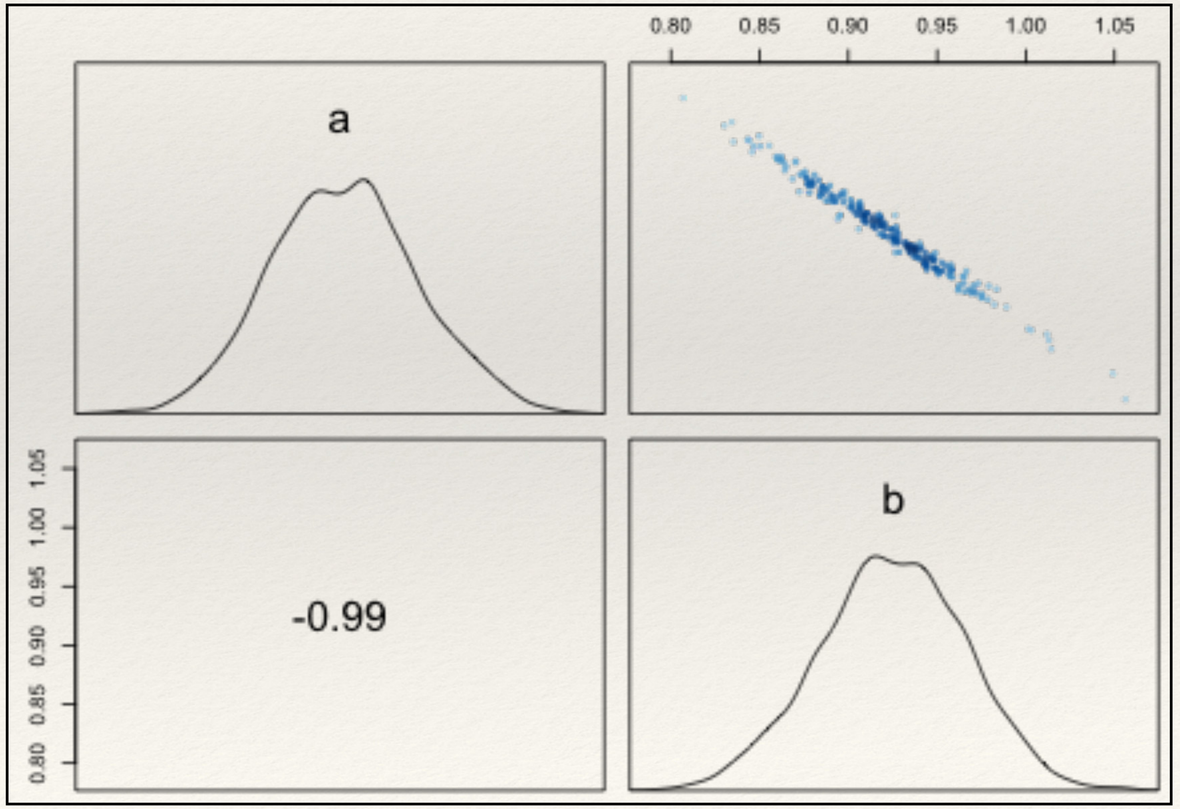


SLOPE-INTERCEPT CORRELATIONS

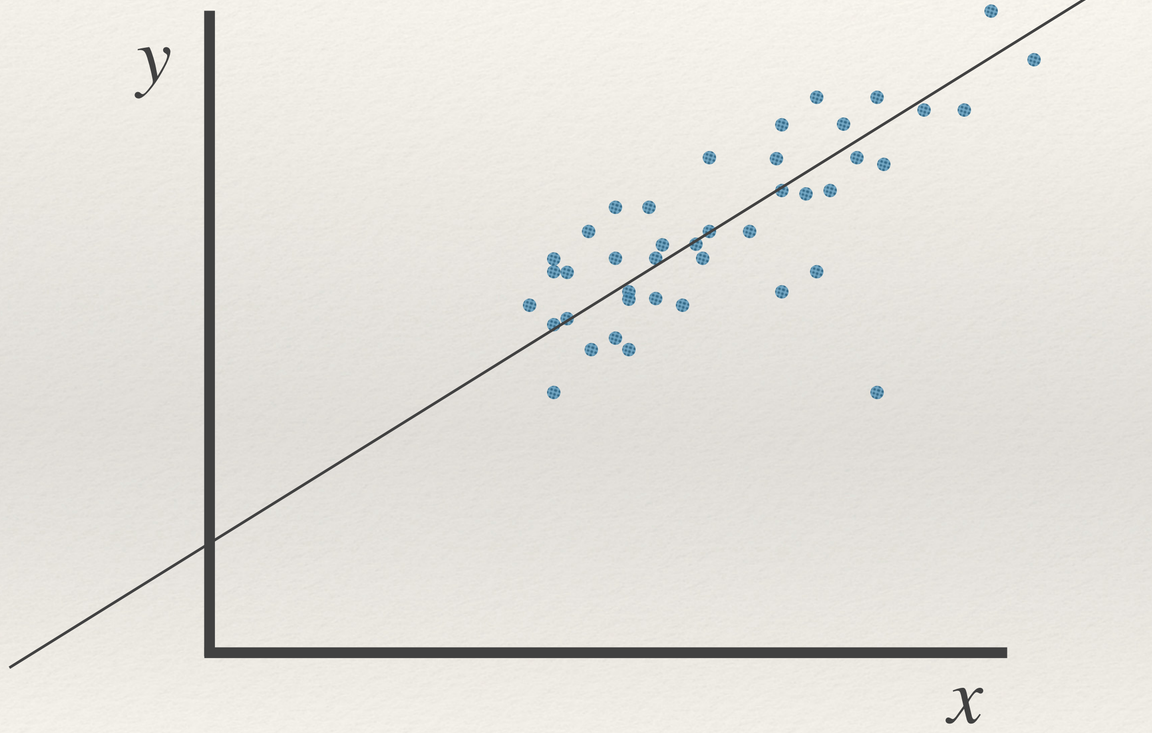


$$y = \alpha + \beta x$$
$$y = \alpha' + \beta' x$$

Pairs plot



LET'S GET RID OF THESE POSTERIOR CORRELATIONS



$$y = \alpha + \beta x$$

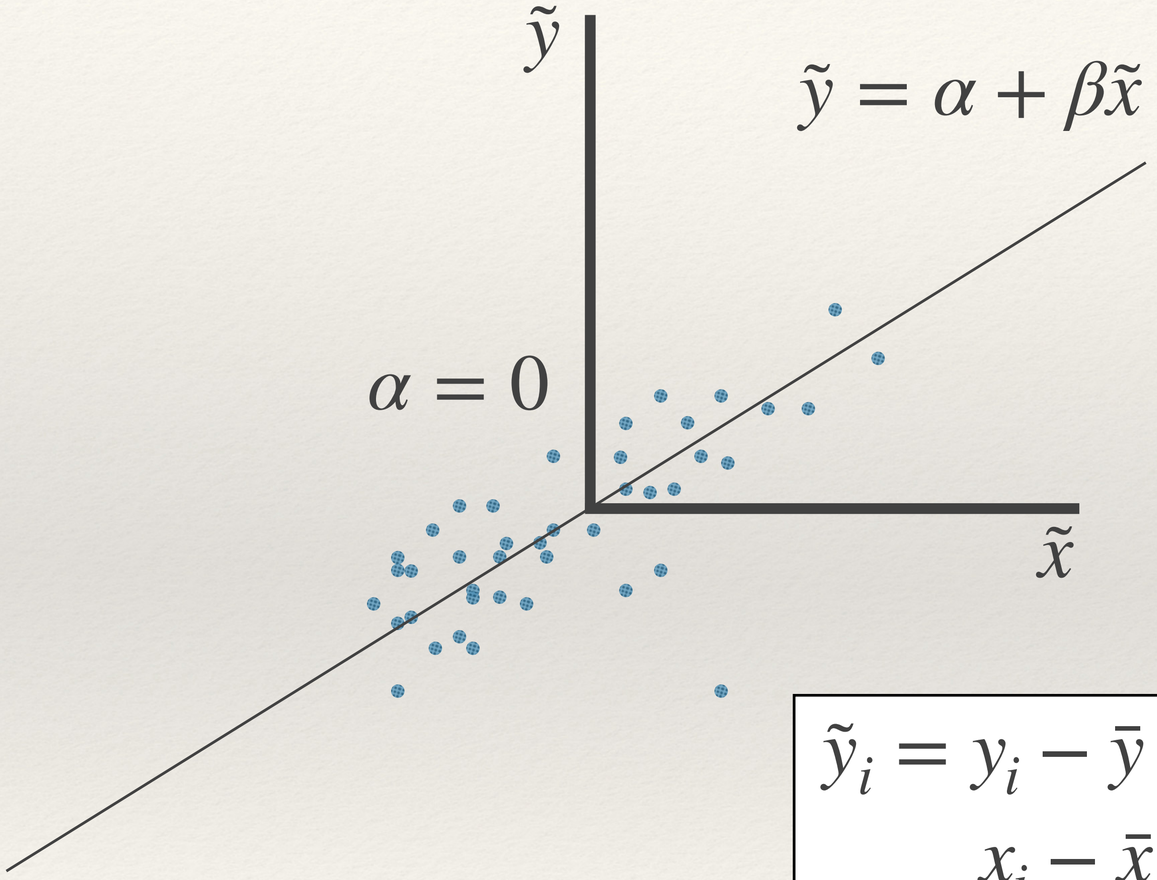
Scaling and shifting parameters can help with convergence

- Centering both x and y values
- Scaling x to unit variance

$$\tilde{y}_i = y_i - \bar{y}$$

$$\tilde{x}_i = \frac{x_i - \bar{x}}{sd(x)}$$

LET'S GET RID OF THESE POSTERIOR CORRELATIONS



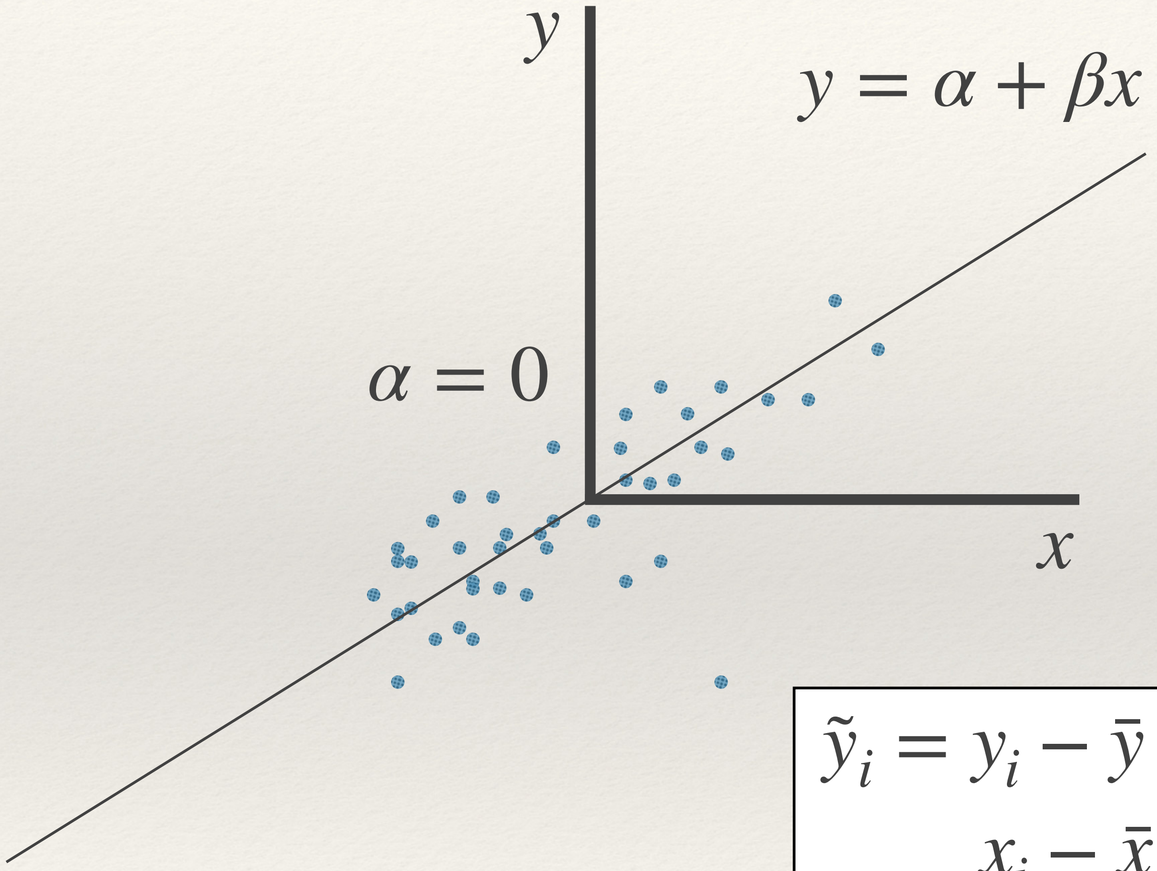
$$\tilde{y}_i = y_i - \bar{y}$$

$$\tilde{x}_i = \frac{x_i - \bar{x}}{sd(x)}$$

```
# Pre-calculate means and sds
mean_x = mean(d2$weight)
sd_x = sd(d2$weight)
mean_y = mean(d2$height)

# Model
ulam(alist(
  y ~ normal(mu, sigma),
  mu <- a + b * x,
  a ~ normal(0, 1),
  b ~ lognormal(0, 1),
  sigma ~ exponential(1)),
  data = list(
    y =  d2$height - mean_y,
    x = (d2$weight - mean_x)/sd_x),
  iter = 1000, chains = 4, cores = 4)
```

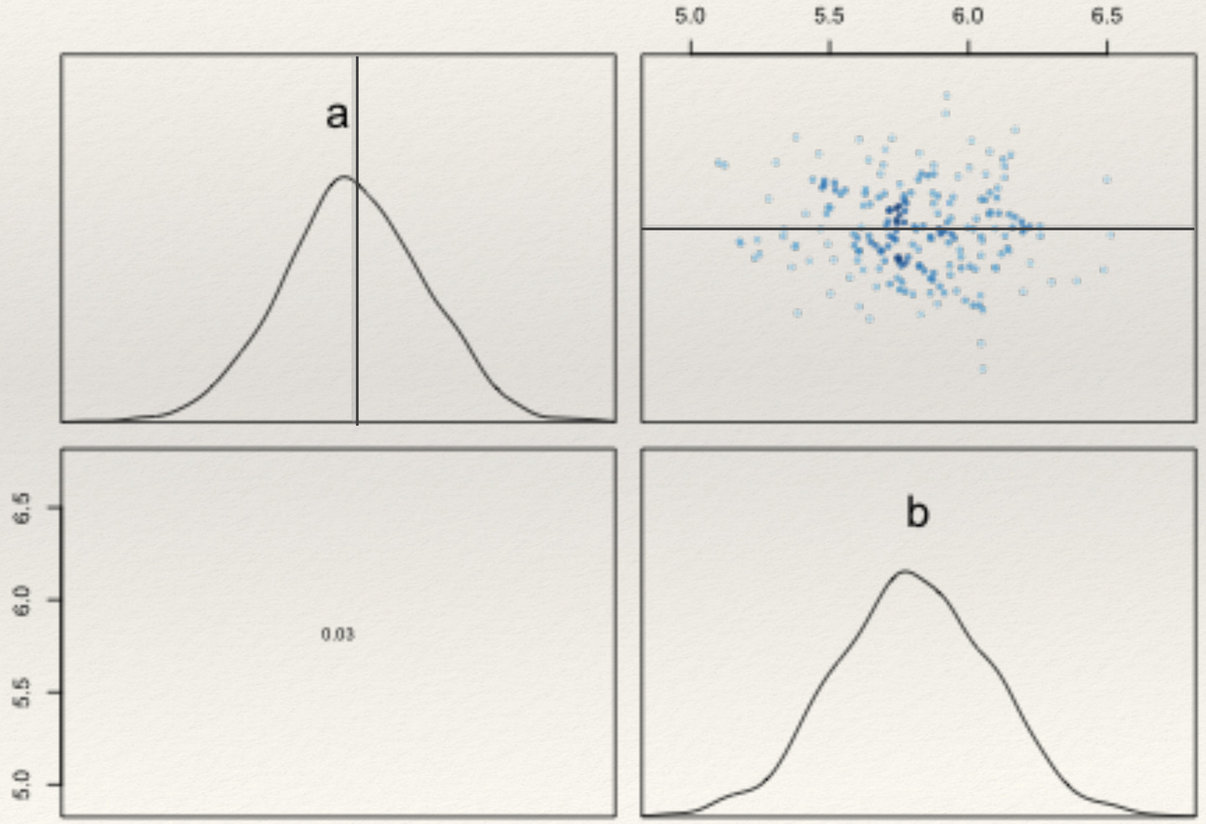
LET'S GET RID OF THESE POSTERIOR CORRELATIONS



$$\tilde{y}_i = y_i - \bar{y}$$

$$\tilde{x}_i = \frac{x_i - \bar{x}}{sd(x)}$$

Pairs plot of the centered model



TAKE HOME MESSAGES

- The posterior distribution contains a lot of useful information not accessible by other methods, all including uncertainty.
- We need to understand the computation methods we use to probe the posterior.
  - Convergence checks are fundamental and can help us diagnose bad models.
  - Stan provides many (many!) tools for model checking.
- There are no true residuals in Bayesian models, but can use posterior simulations to make and understand our predictions.