BAYESIAN MODELING WITH RJAGS

A simple Bayesian regression model

Alicia Johnson
Associate Professor, Macalester College

Chapter 3 goals

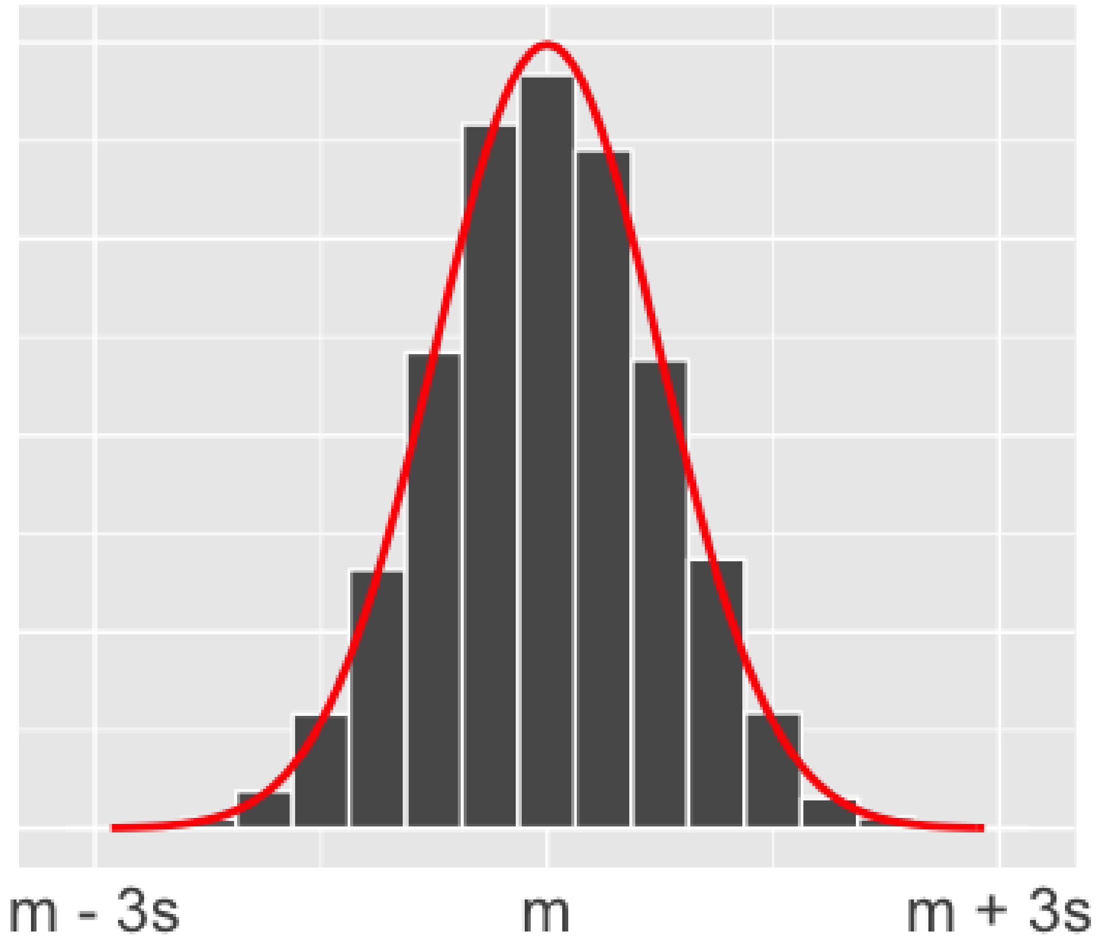
- Engineer a simple Bayesian regression model
- Define, compile, and simulate regression models in RJAGS
- Use Markov chain simulation output for posterior inference & prediction

Modeling weight

Y_i = weight of adult i (kg)

Model

$Y_i \sim N(m, s^2)$

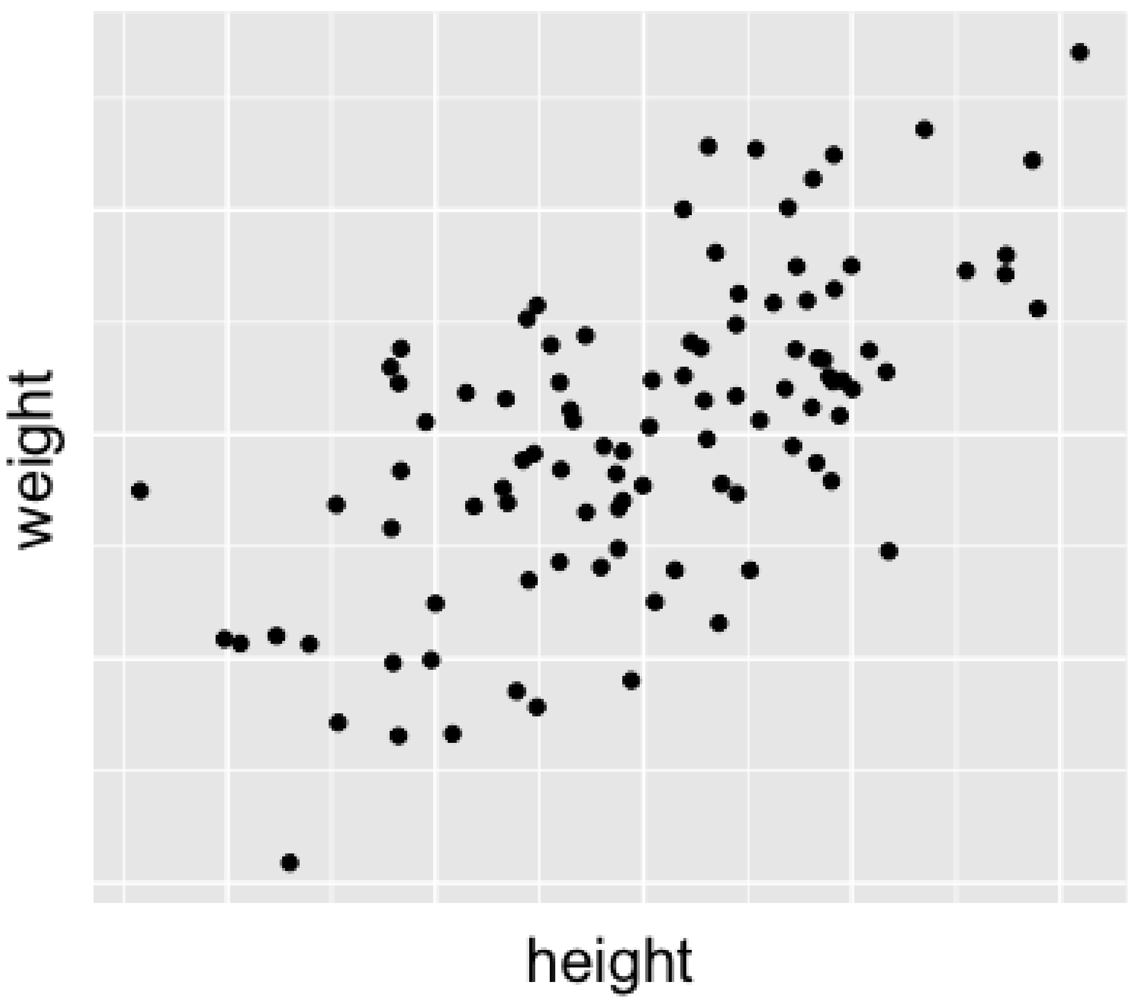


Modeling weight by height

Y_i = weight of adult i (kg)

Model

$Y_i \sim N(m, s^2)$

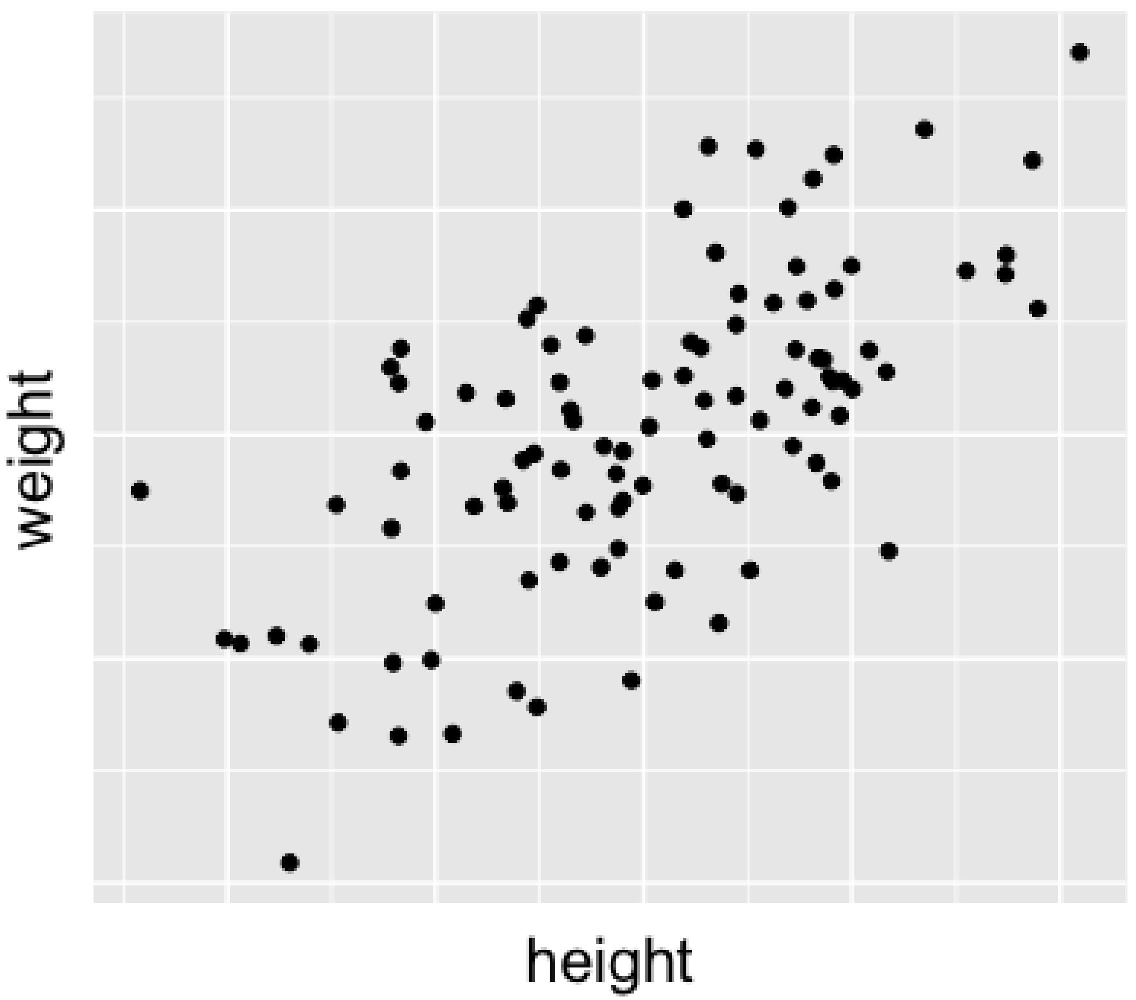


Modeling weight by height

Y_i = weight of adult i (kg)
X_i = height of adult i (cm)

Model

$Y_i \sim N(m_i, s^2)$

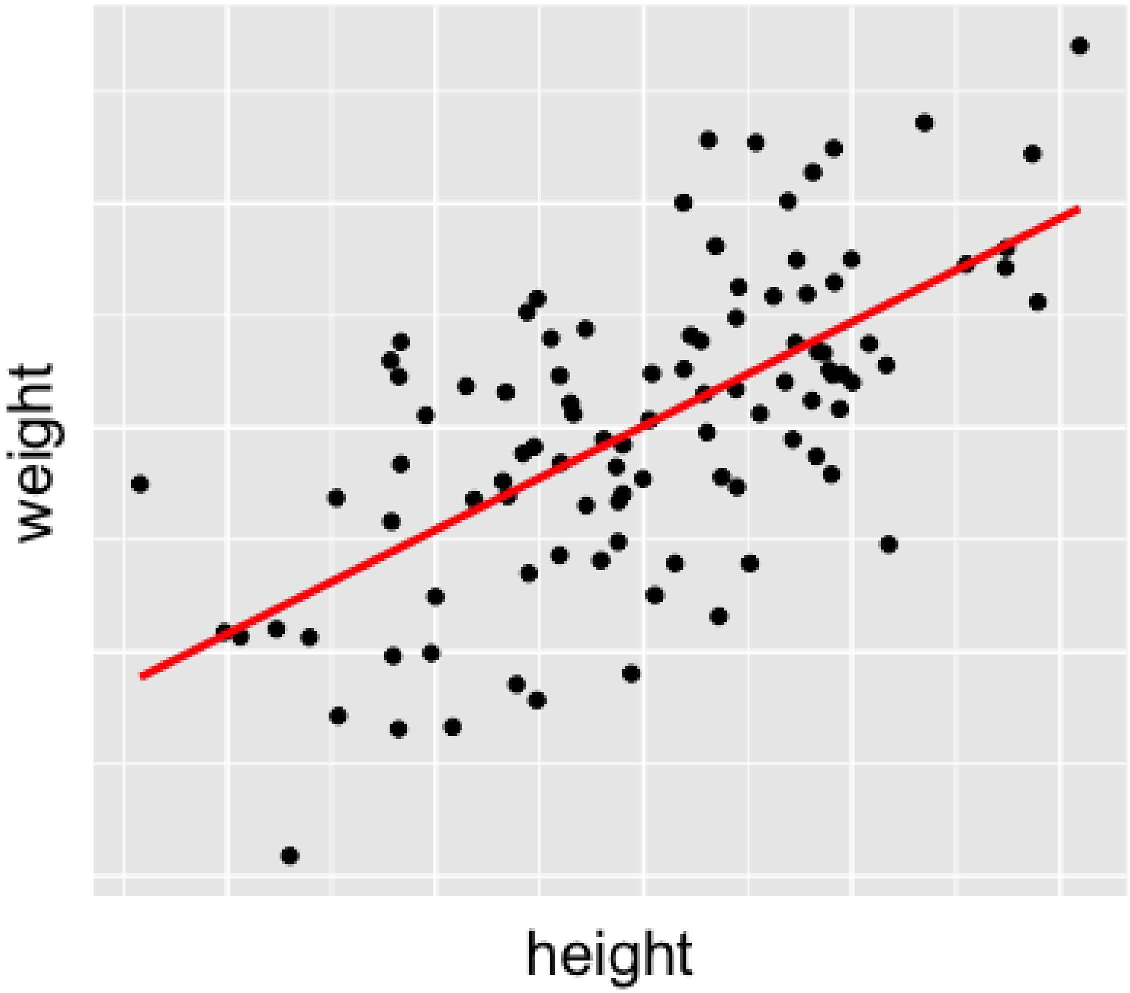


Modeling weight by height

Y_i = weight of adult i (kg)
X_i = height of adult i (cm)

Model

$Y_i \sim N(m_i, s^2)$
$m_i = a + bX_i$

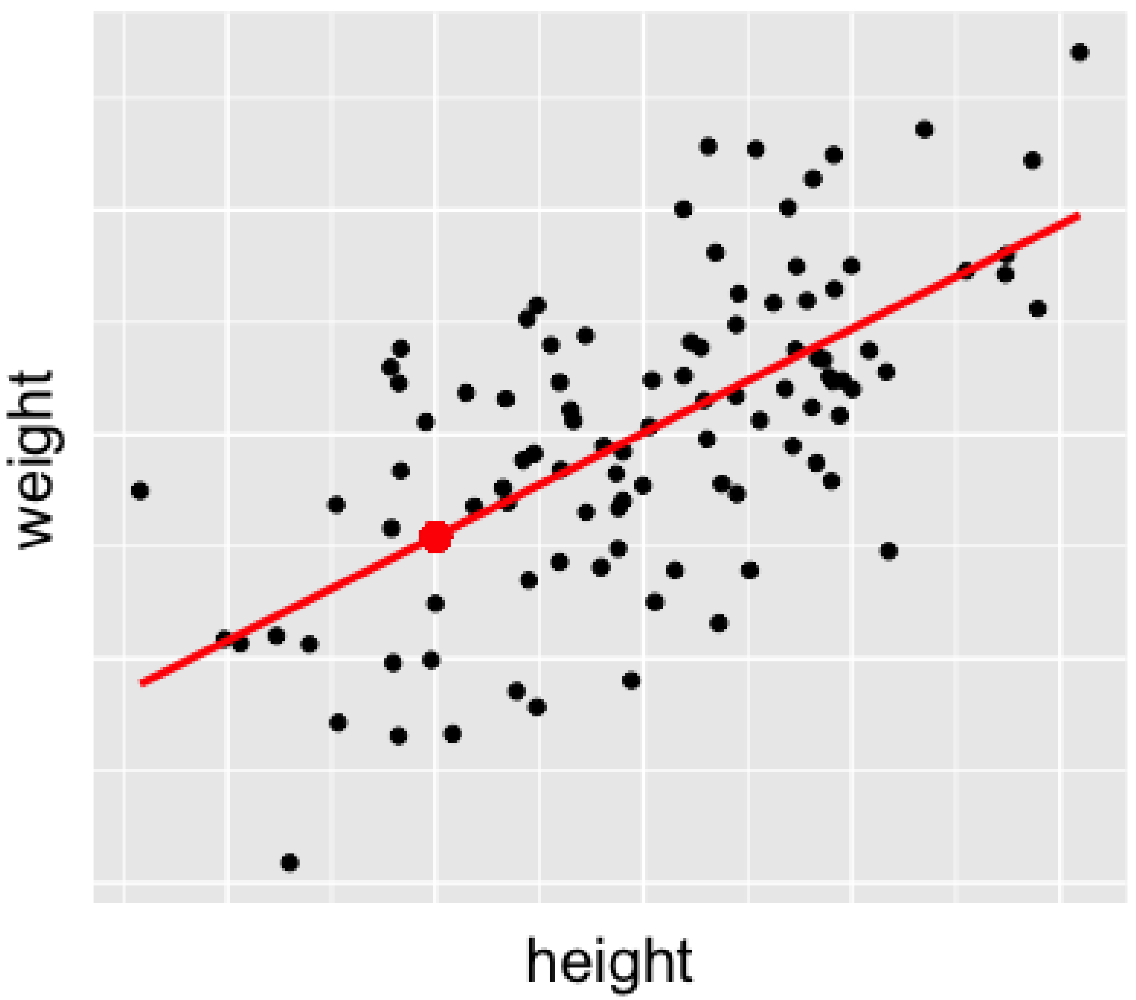


Modeling weight by height

Y_i = weight of adult i (kg)
X_i = height of adult i (cm)

Model

$Y_i \sim N(m_i, s^2)$
$m_i = a + bX_i$



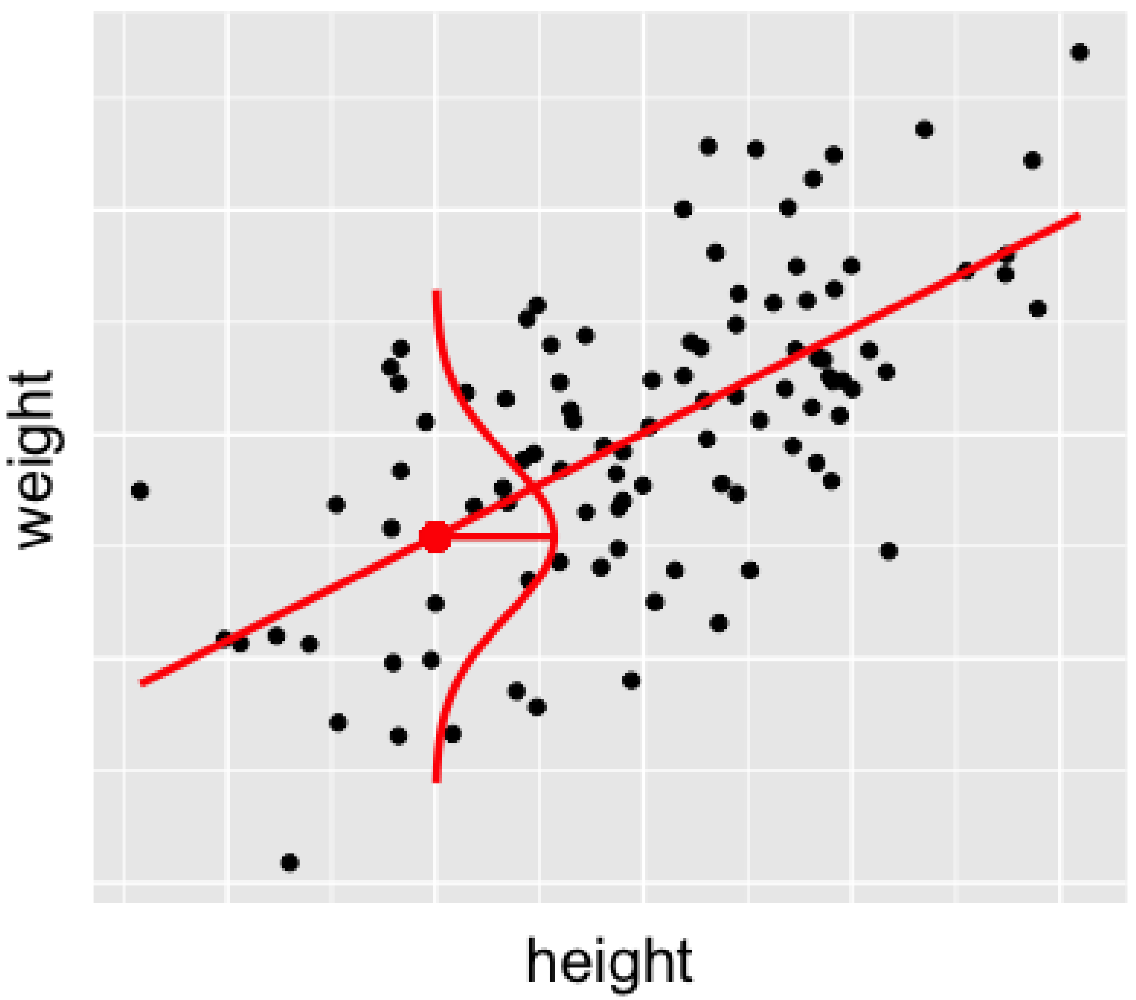
Modeling weight by height

Y_i = weight of adult i (kg)
X_i = height of adult i (cm)

Model

$Y_i \sim N(m_i, s^2)$

$m_i = a + bX_i$

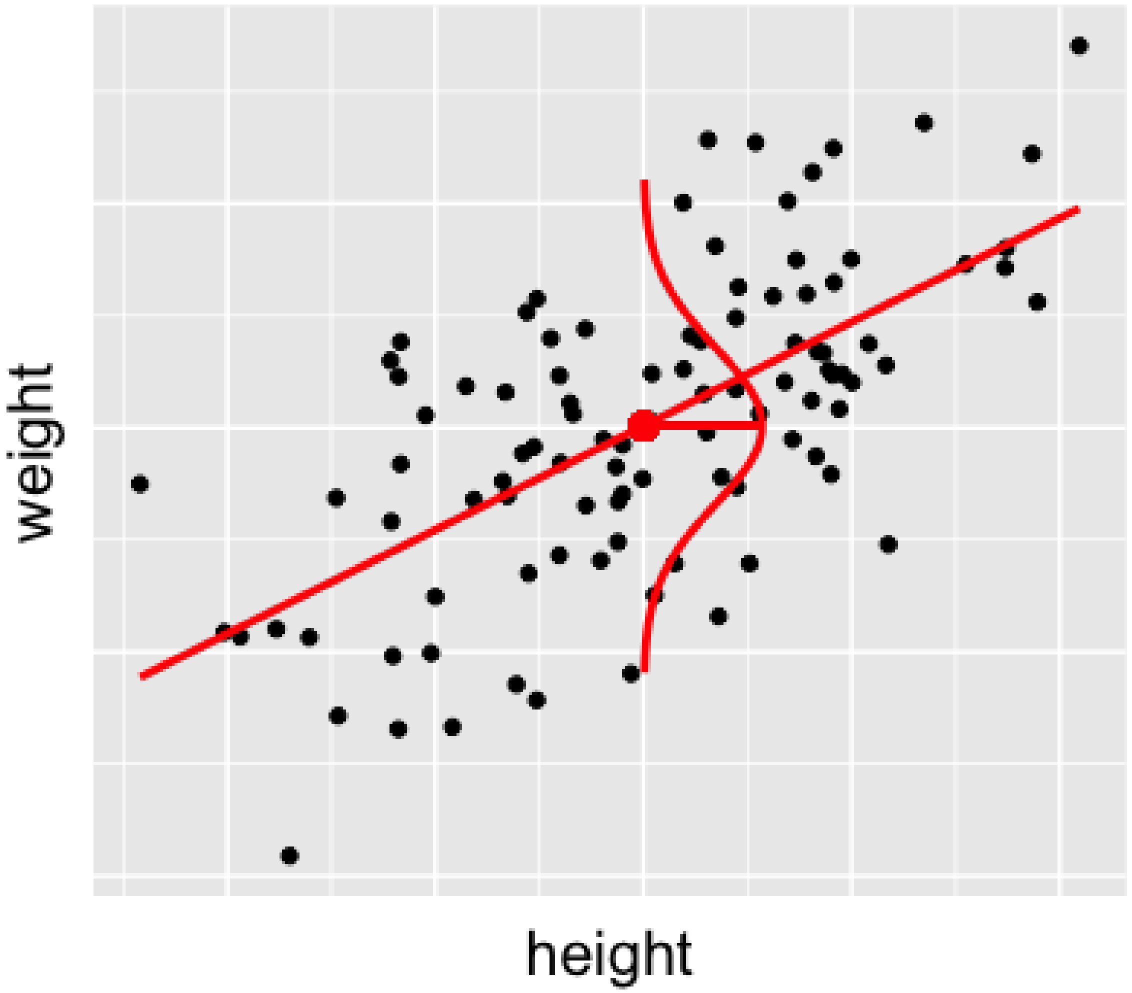


Modeling weight by height

Y_i = weight of adult i (kg)
X_i = height of adult i (cm)

Model

$Y_i \sim N(m_i, s^2)$
$m_i = a + bX_i$



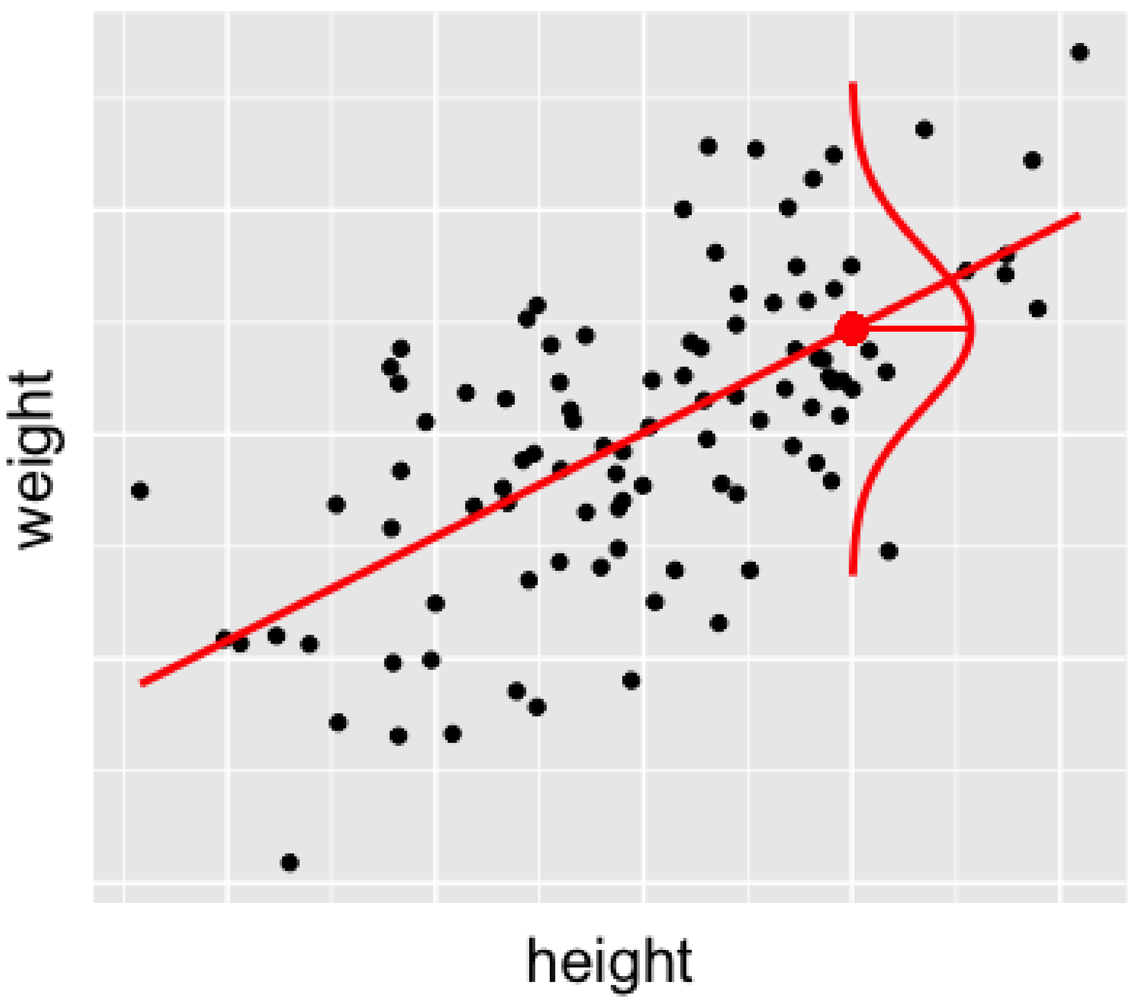
Modeling weight by height

Y_i = weight of adult i (kg)
X_i = height of adult i (cm)

Model

$Y_i \sim N(m_i, s^2)$

$m_i = a + bX_i$



Bayesian regression model

$Y_i \sim N(m_i, s^2)$
$m_i = a + bX_i$

- a = y-intercept

  value of m_i when $X_i = 0$

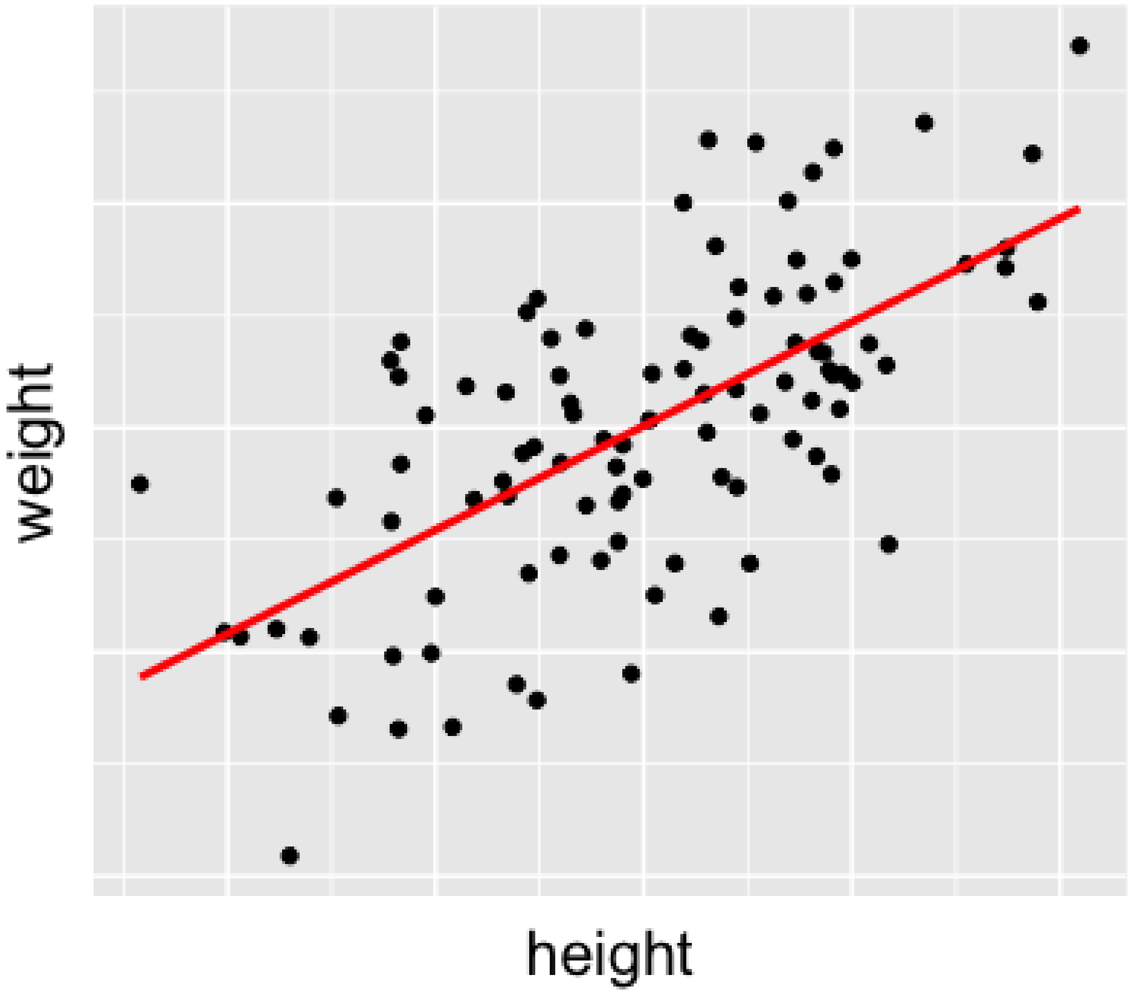
- b = slope

  rate of change in weight (kg) per 1 cm

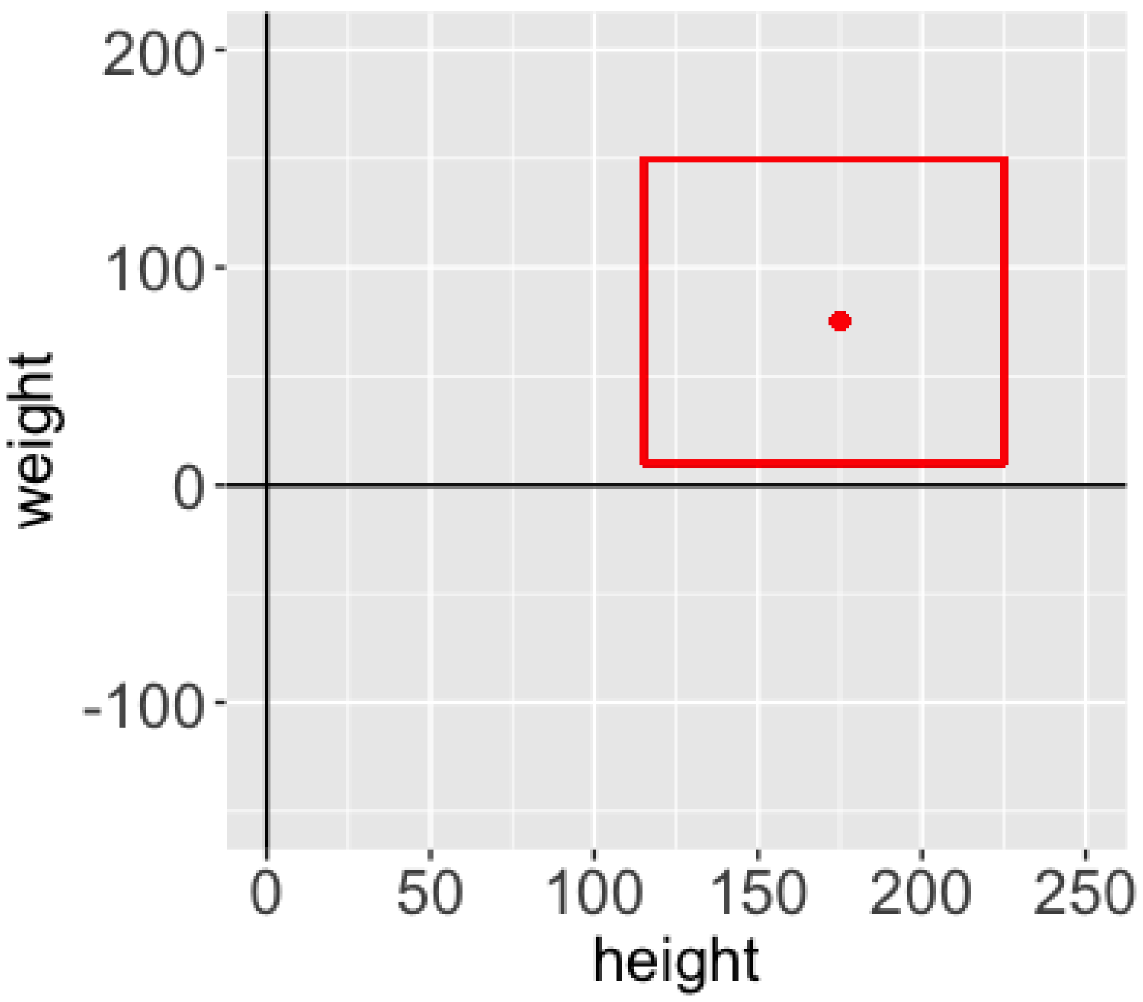
  increase in height

- s = residual standard deviation

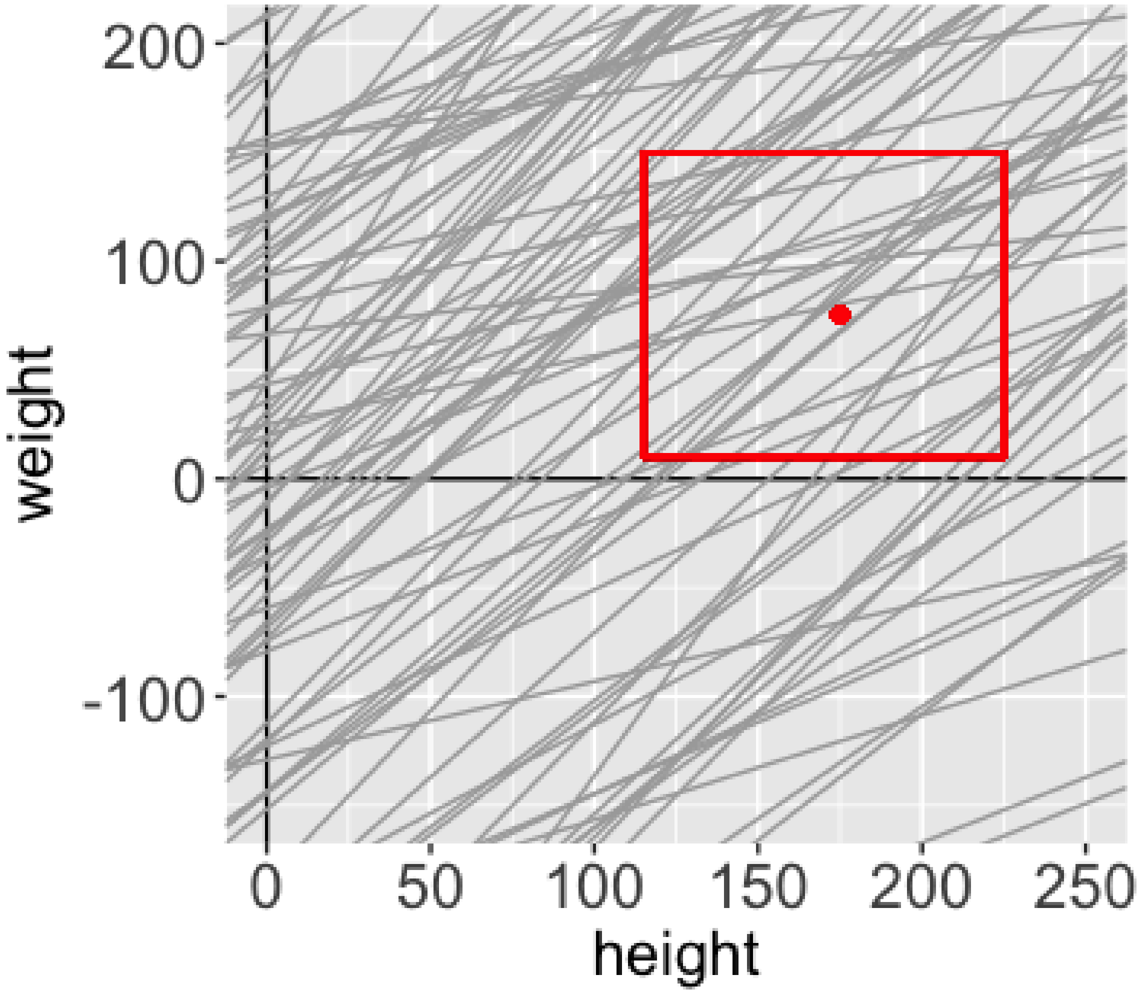
  individual deviation from trend m_i

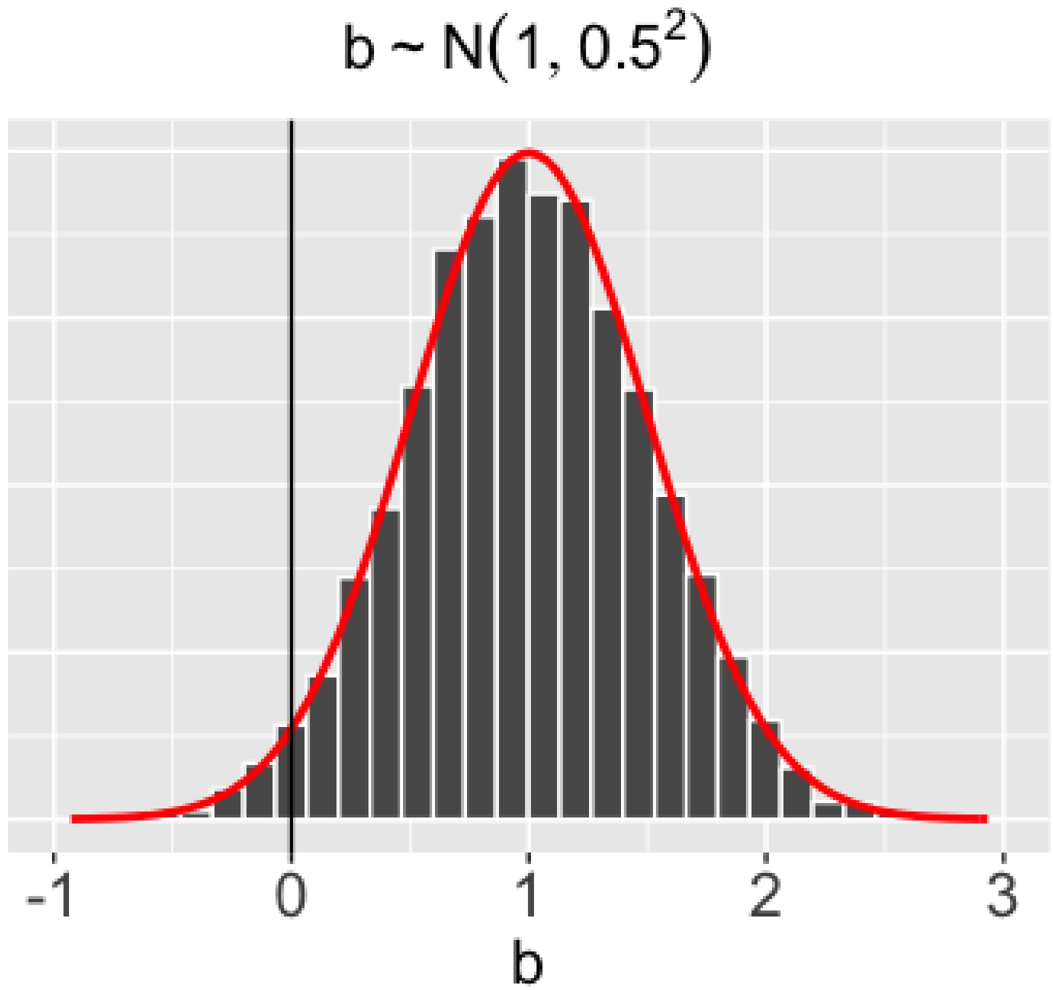


Priors for the intercept & slope

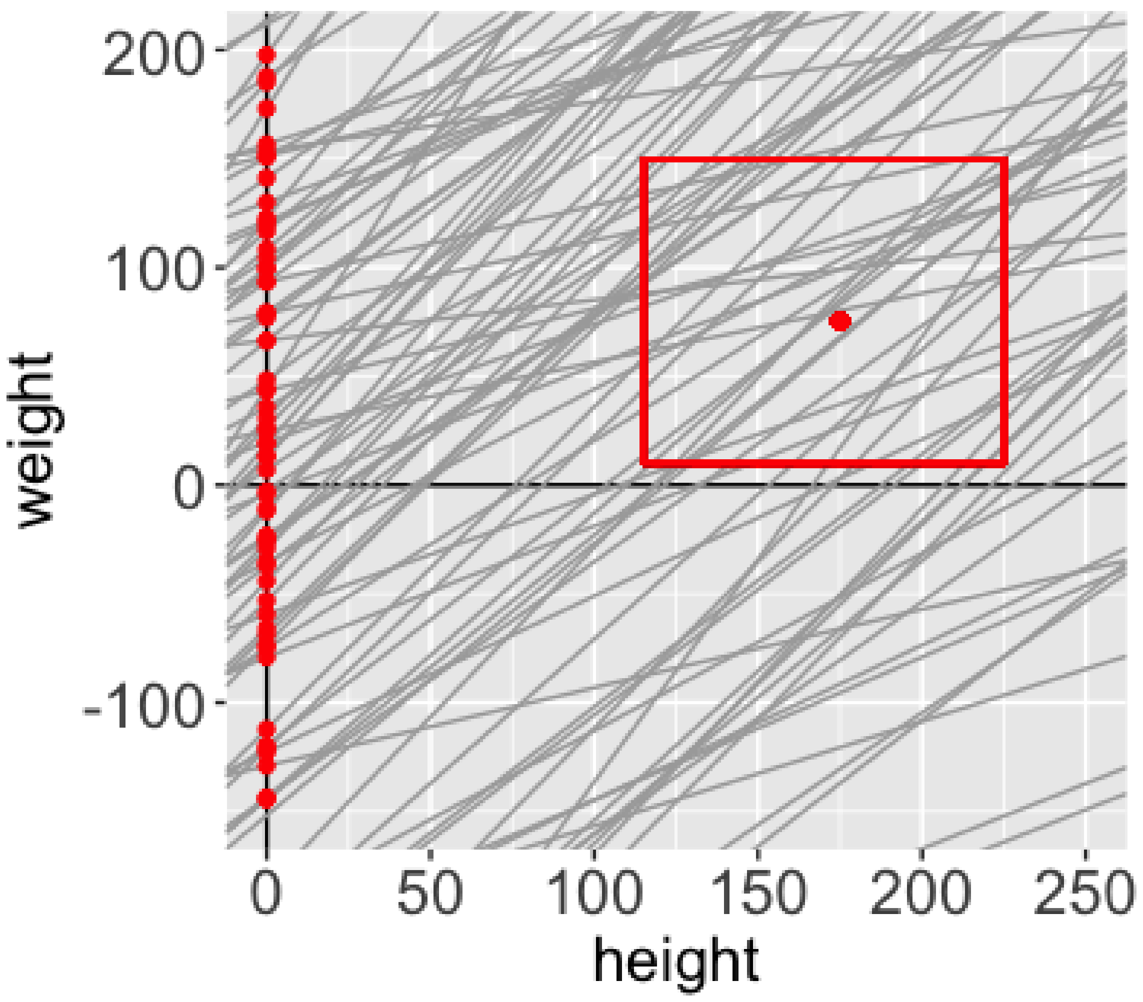


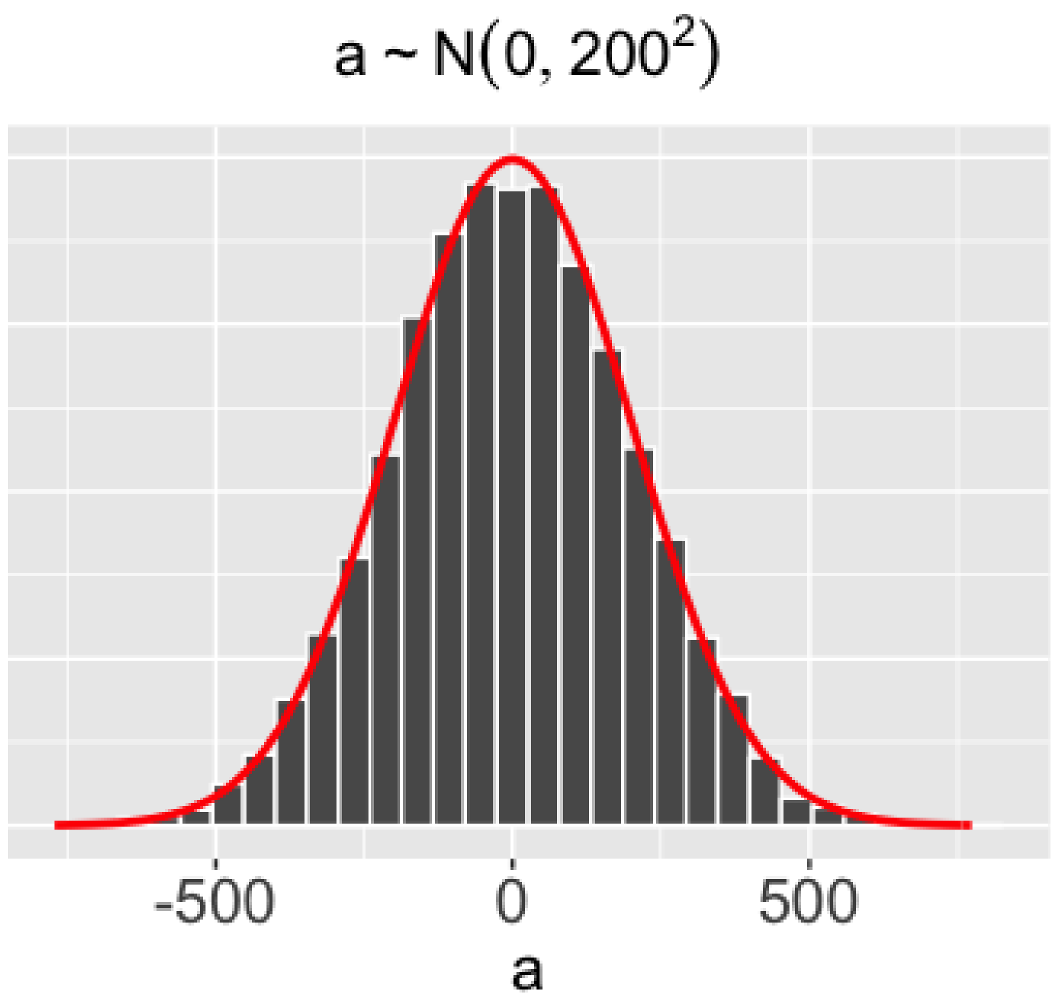
Priors for the intercept & slope



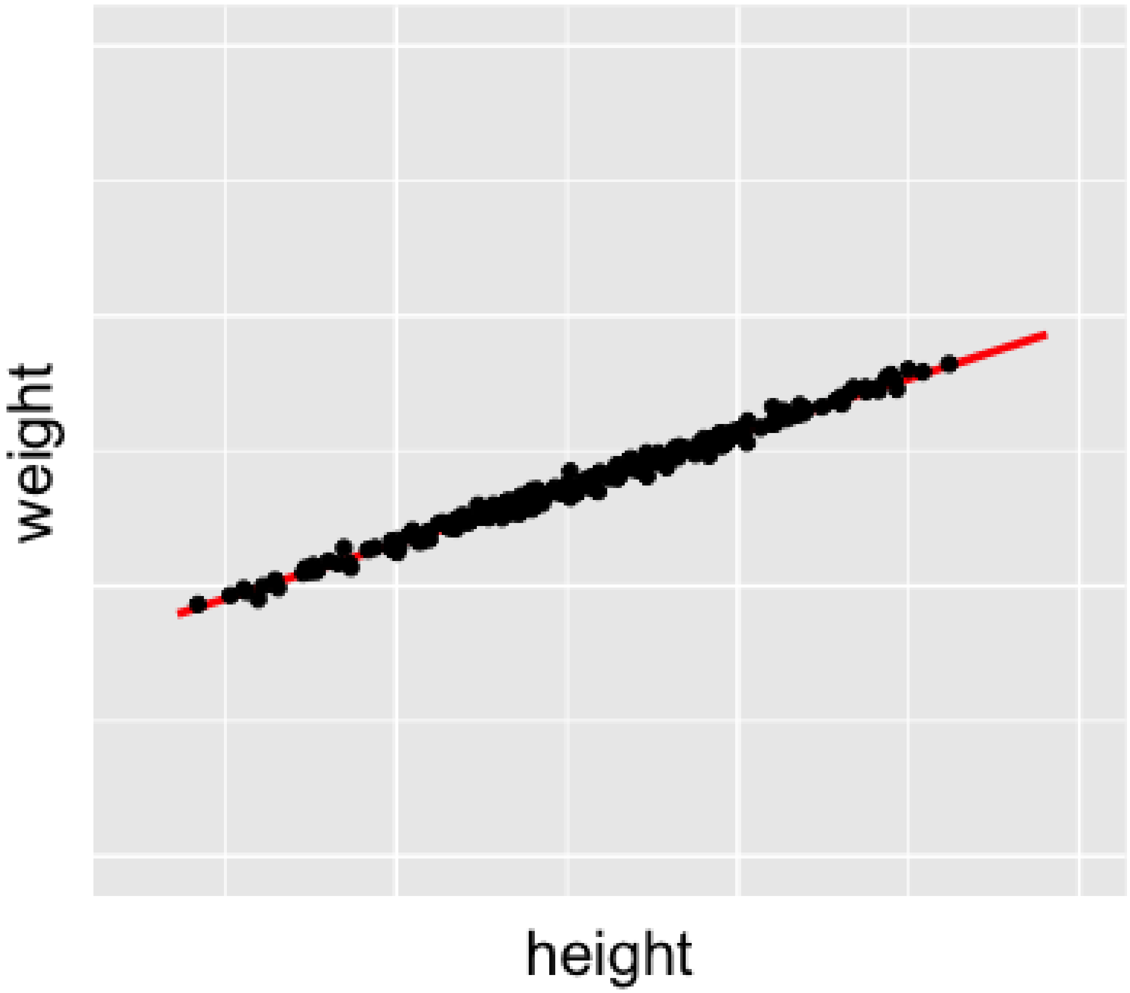


Priors for the intercept & slope

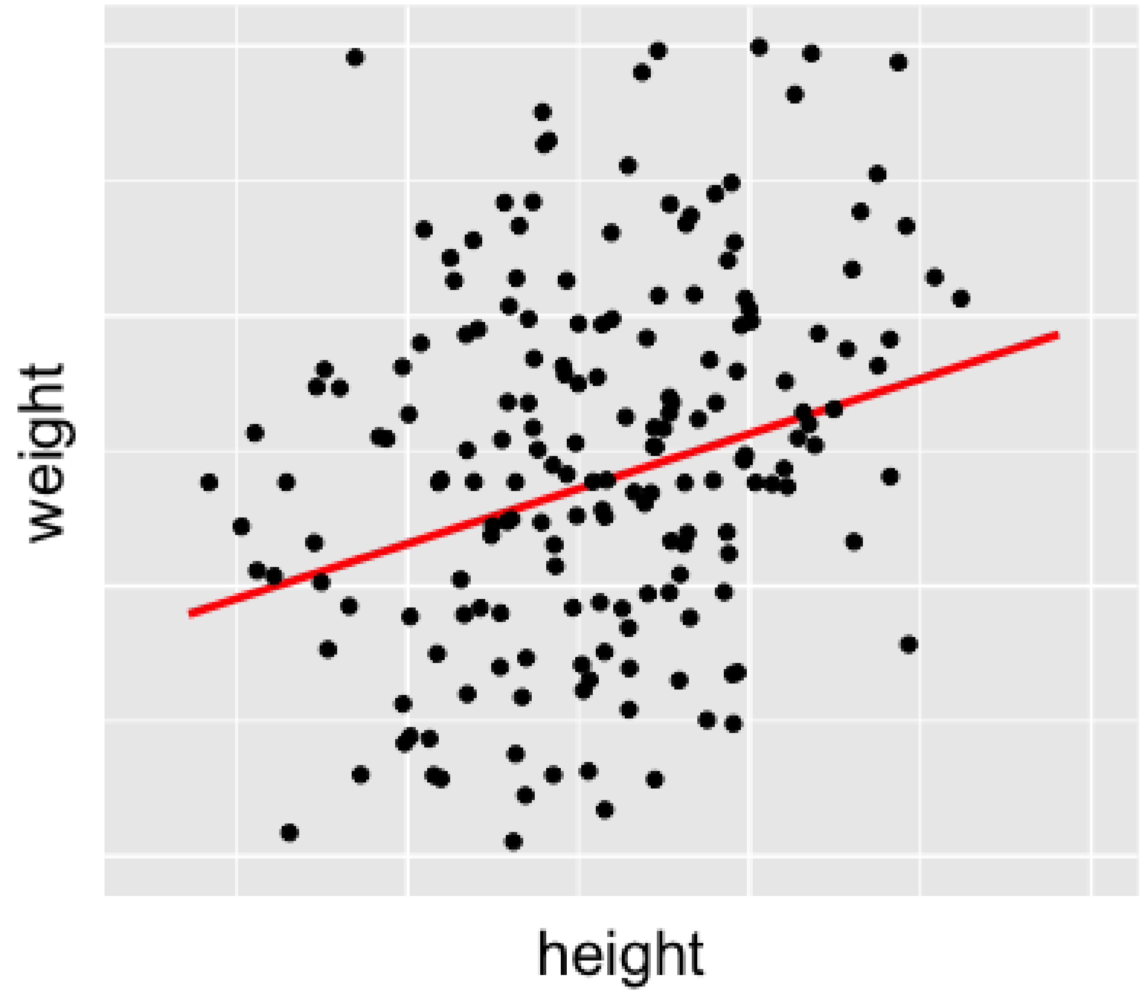


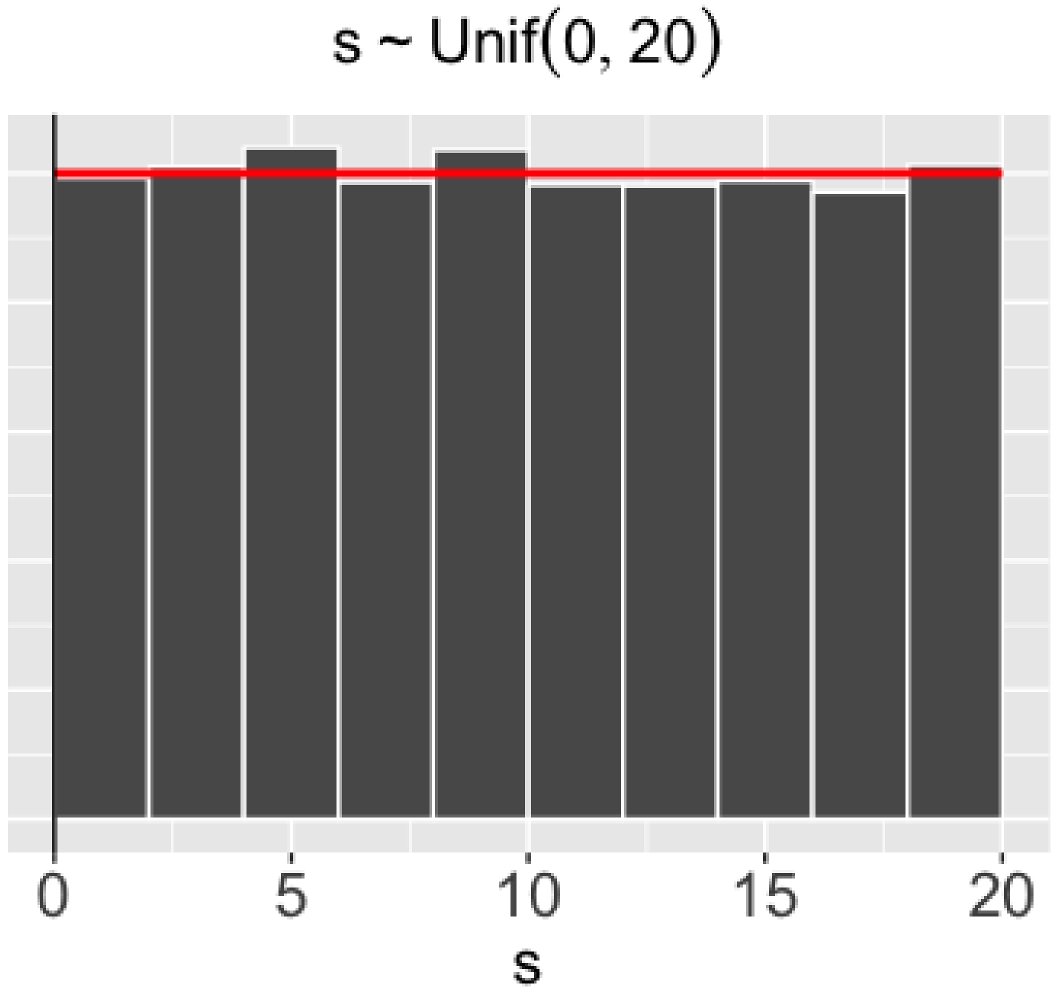


Prior for the residual standard deviation



Prior for the residual standard deviation

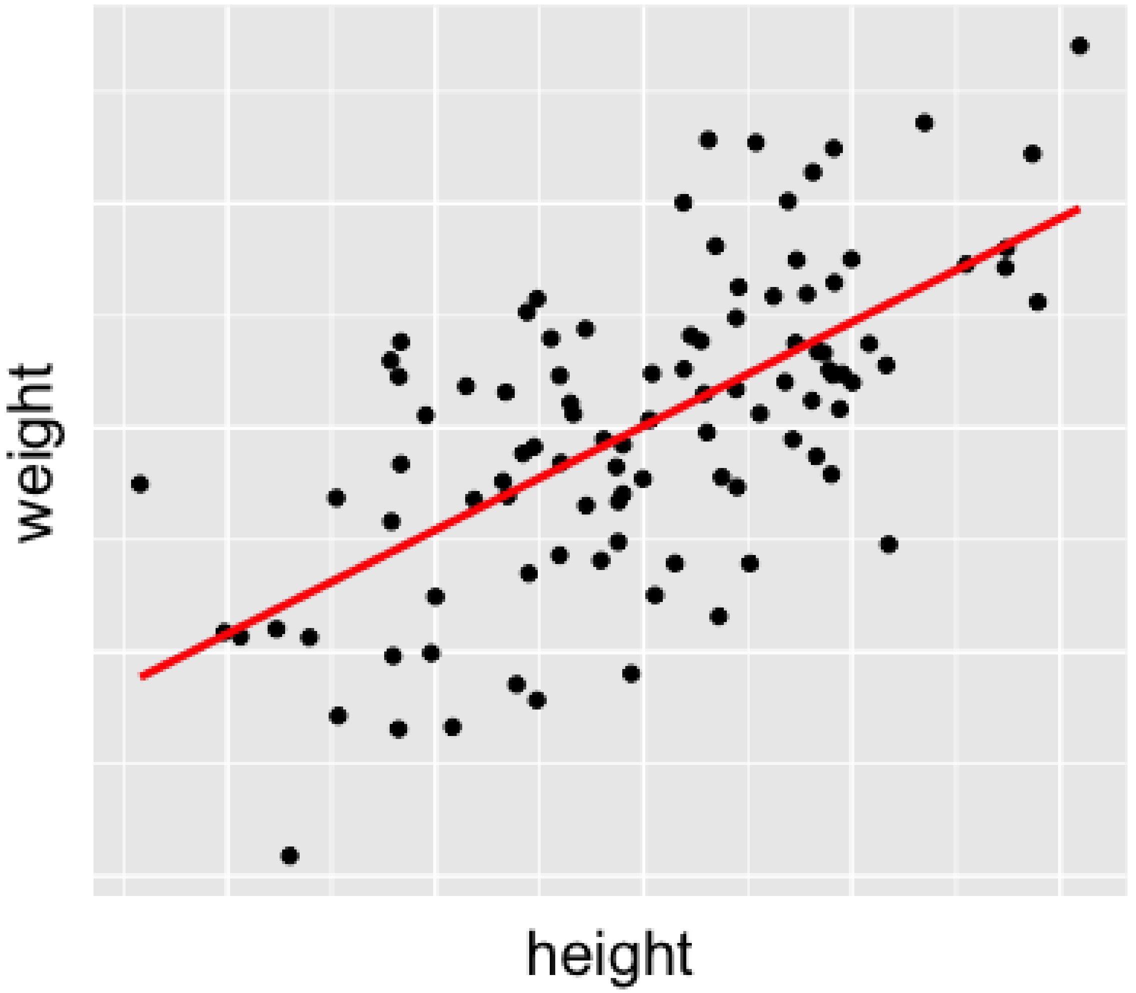




Bayesian regression model

$$Y_i \sim N(m_i, s^2)$$
$$m_i = a + bX_i$$

$$a \sim N(0, 200^2)$$
$$b \sim N(1, 0.5^2)$$
$$s \sim Unif(0, 20)$$



BAYESIAN MODELING WITH RJAGS

Let's practice!

BAYESIAN MODELING WITH RJAGS

Bayesian regression in RJAGS

Alicia Johnson
Associate Professor, Macalester College

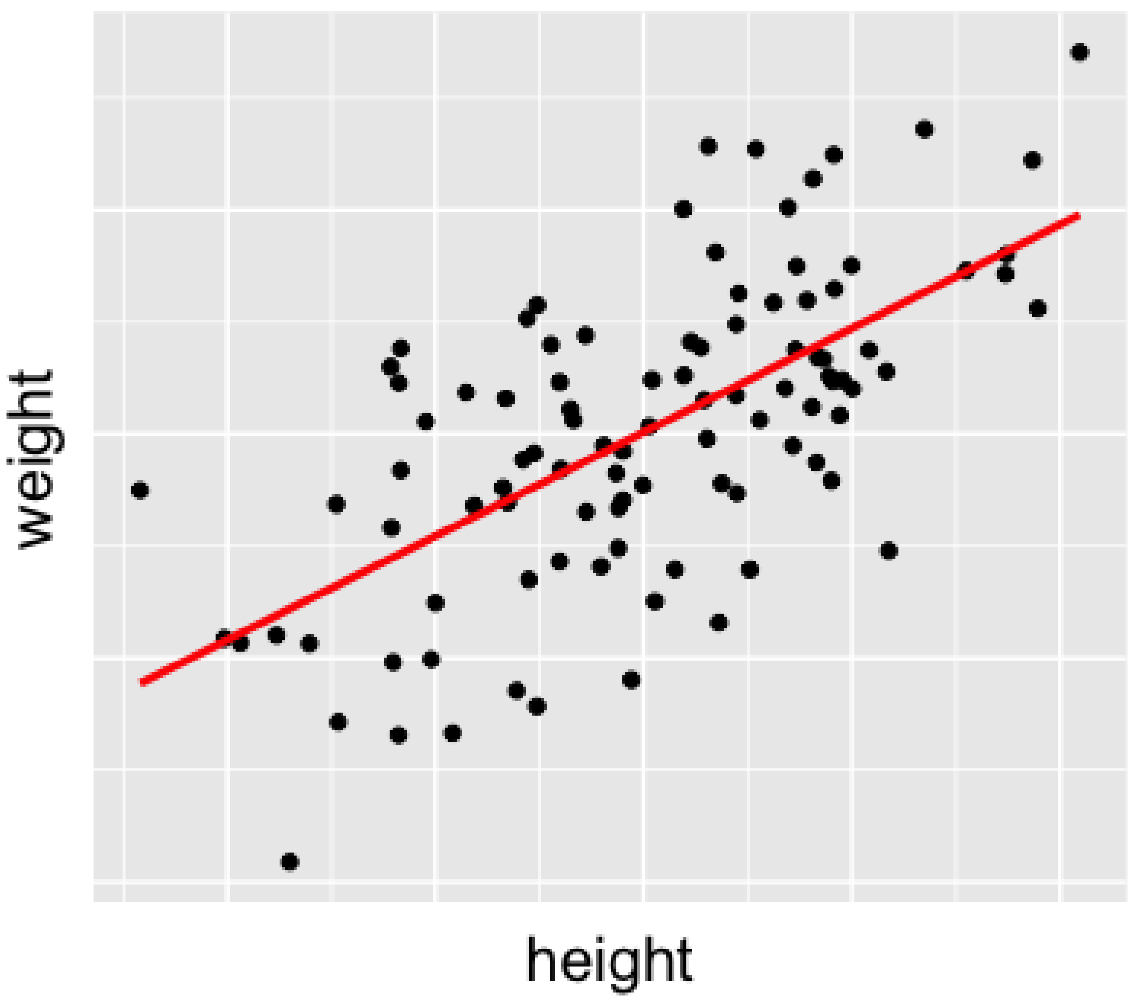
Bayesian regression model

Y_i = weight of adult i (kg)
X_i = height of adult i (cm)

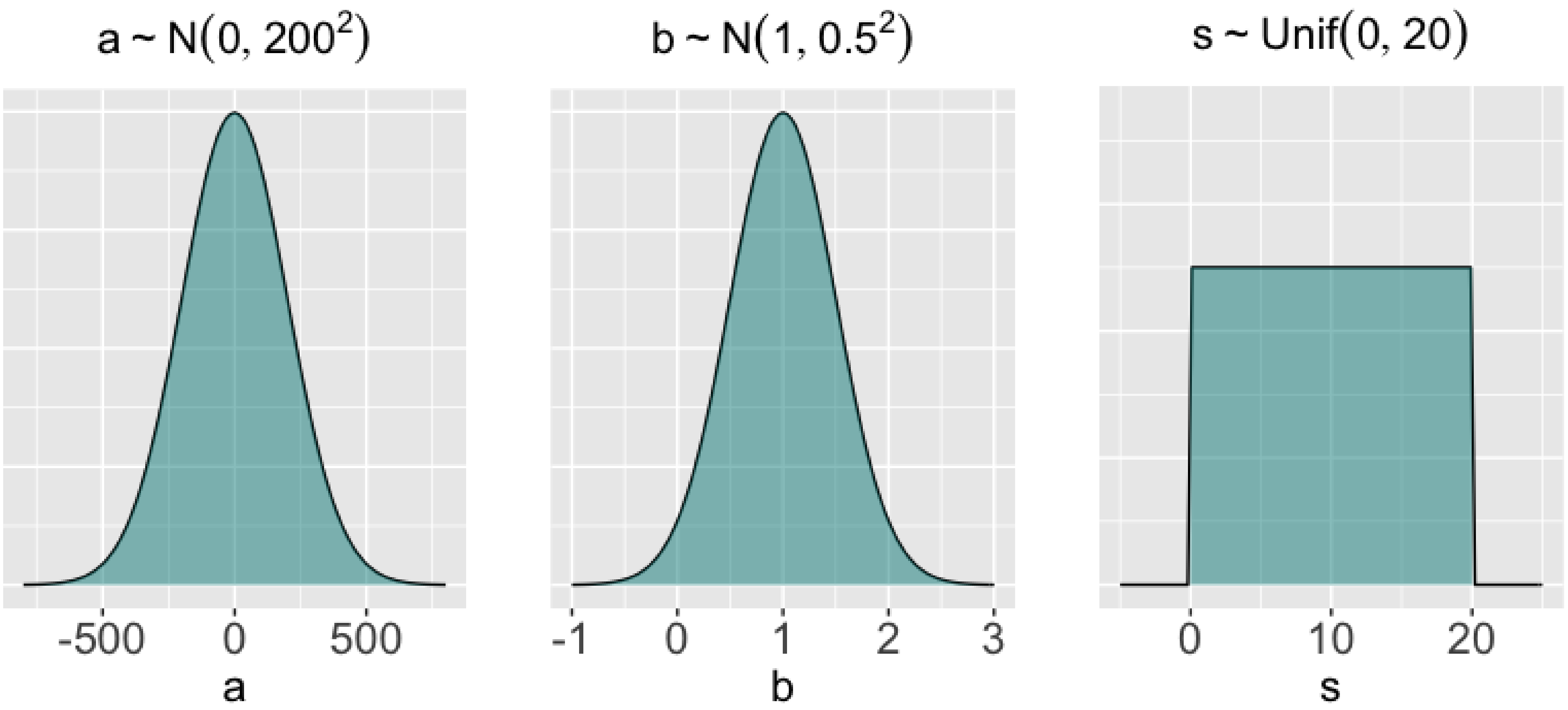
Model

$$Y_i \sim N(m_i, s^2)$$
$$m_i = a + bX_i$$

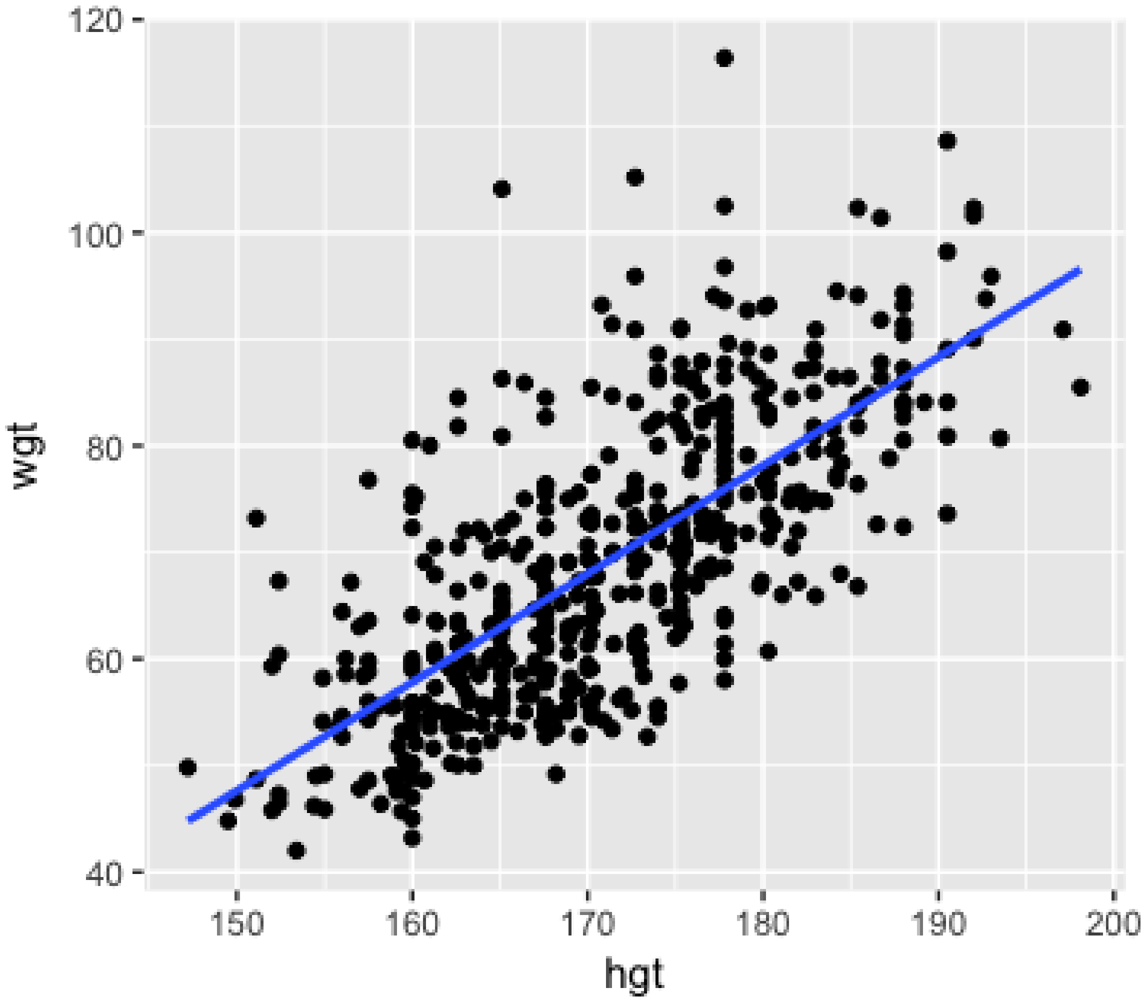
$$a \sim N(0, 200^2)$$
$$b \sim N(1, 0.5^2)$$
$$s \sim Unif(0, 20)$$



Prior insight



Insight from the observed weight & height data



$$Y_i \sim N(m_i, s^2)$$
$$m_i = a + bX_i$$

```
> wt_mod <- lm(wgt ~ hgt, bdims)

> coef(wt_mod)
(Intercept)          hgt
-105.011254    1.017617

> summary(wt_mod)$sigma
[1] 9.30804
```

DEFINE the regression model

```
weight_model <- "model{
    # Likelihood model for Y[i]





    # Prior models for a, b, s




}"
```

DEFINE the regression model

```
weight_model <- "model{
    # Likelihood model for Y[i]
    for(i in 1:length(Y)) {


    }

    # Prior models for a, b, s




}"
```

- $Y_i \sim N(m_i, s^2)$ for i from 1 to 507

DEFINE the regression model

```
weight_model <- "model{
    # Likelihood model for Y[i]
    for(i in 1:length(Y)) {
      Y[i] ~ dnorm(m[i], s^(-2))

    }

    # Prior models for a, b, s




}"
```

- $Y_i \sim N(m_i, s^2)$ for i from 1 to 507

DEFINE the regression model

```
weight_model <- "model{
    # Likelihood model for Y[i]
    for(i in 1:length(Y)) {
      Y[i] ~ dnorm(m[i], s^(-2))
      m[i] <- a + b * X[i]
    }

    # Prior models for a, b, s



}"
```

- $Y_i \sim N(m_i, s^2)$ for i from 1 to 507

  $m_i = a + bX_i$

  NOTE: use "<-" not "~"

DEFINE the regression model

```
weight_model <- "model{
    # Likelihood model for Y[i]
    for(i in 1:length(Y)) {
      Y[i] ~ dnorm(m[i], s^(-2))
      m[i] <- a + b * X[i]
    }

    # Prior models for a, b, s
    a ~ dnorm(0, 200^(-2))
    b ~ dnorm(1, 0.5^(-2))
    s ~ dunif(0, 20)

}"
```

- $Y_i \sim N(m_i, s^2)$ for i from 1 to 507

  $m_i = a + bX_i$

- $a \sim N(0, 200^2)$

  $b \sim N(1, 0.5^2)$

  $s \sim Unif(0, 20)$

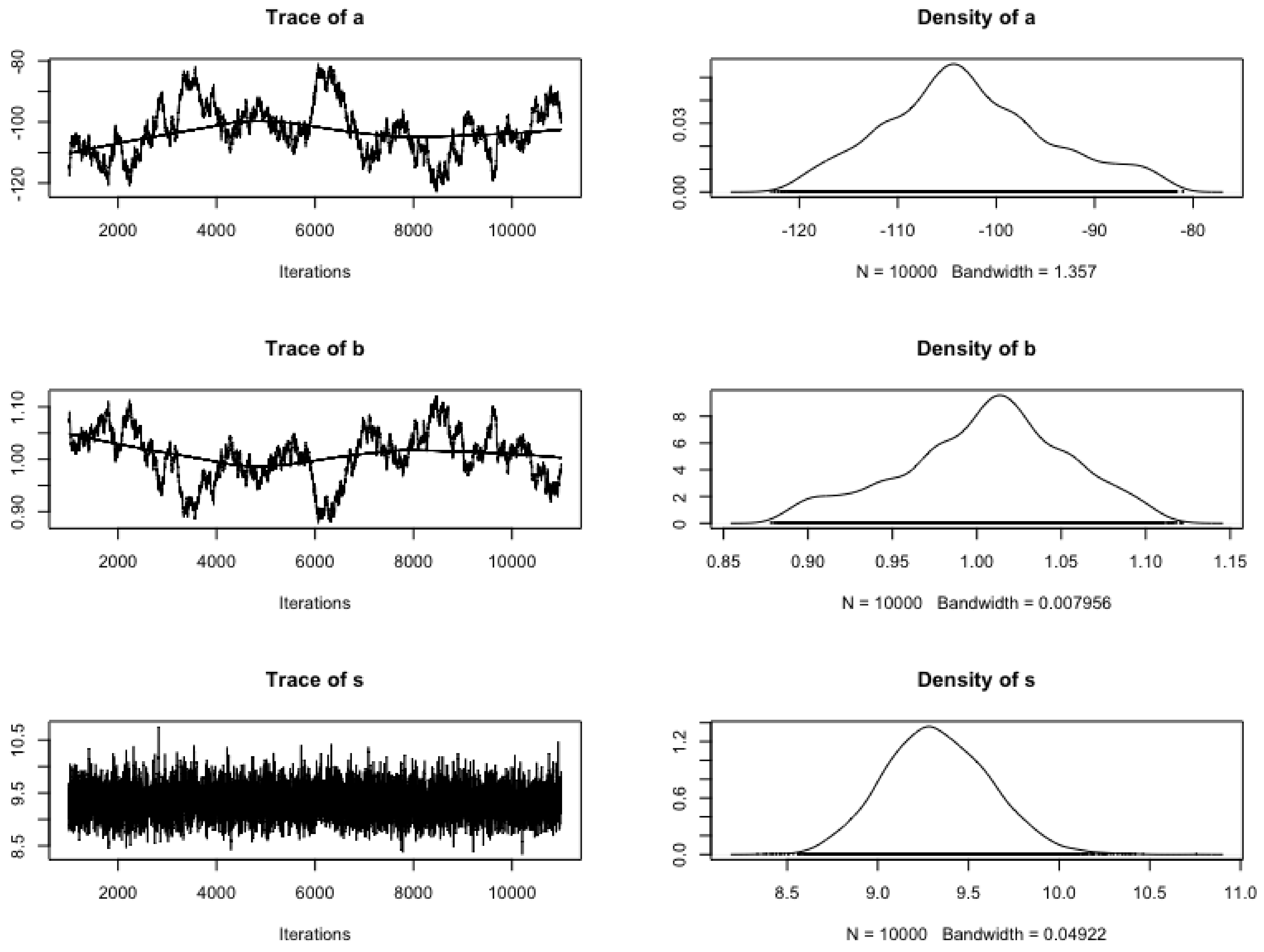
COMPILE the regression model

```
# COMPILE the model
weight_jags <- jags.model(textConnection(weight_model),
    data = list(X = bdims$hgt, Y = bdims$wgt),
    inits = list(.RNG.name = "base::Wichmann-Hill", .RNG.seed = 2018))
```

```
> dim(bdims)
[1] 507  25

> head(bdims$hgt)
[1] 174.0 175.3 193.5 186.5 187.2 181.5

> head(bdims$wgt)
[1] 65.6 71.8 80.7 72.6 78.8 74.8
```

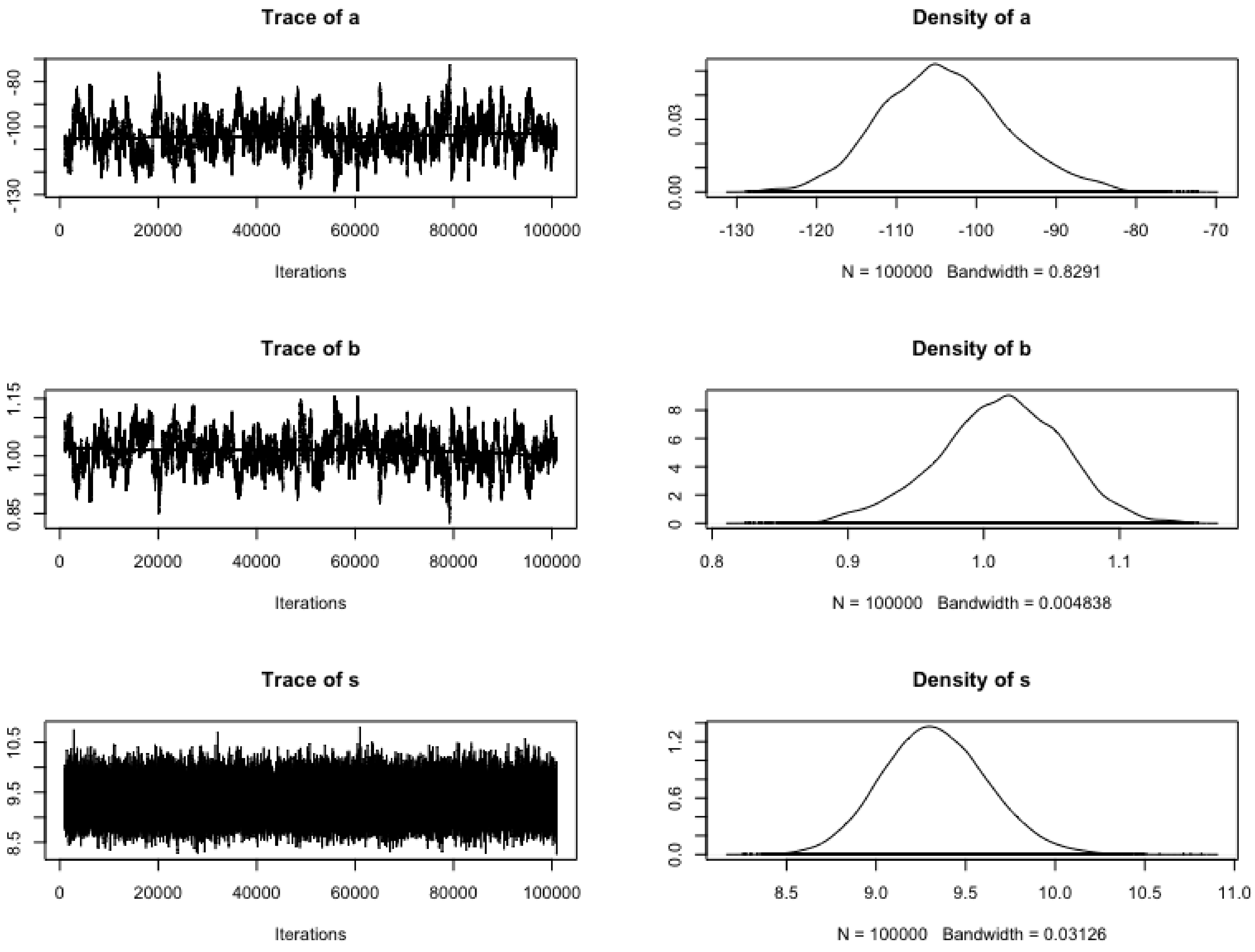
SIMULATE the regression model

```
# COMPILE the model
weight_jags <- jags.model(textConnection(weight_model),
    data = list(X = bdims$hgt, Y = bdims$wgt),
    inits = list(.RNG.name = "base::Wichmann-Hill", .RNG.seed = 2018))

# SIMULATE the posterior
weight_sim <- coda.samples(model = weight_jags,
    variable.names = c("a", "b", "s"),
    n.iter = 10000)
```

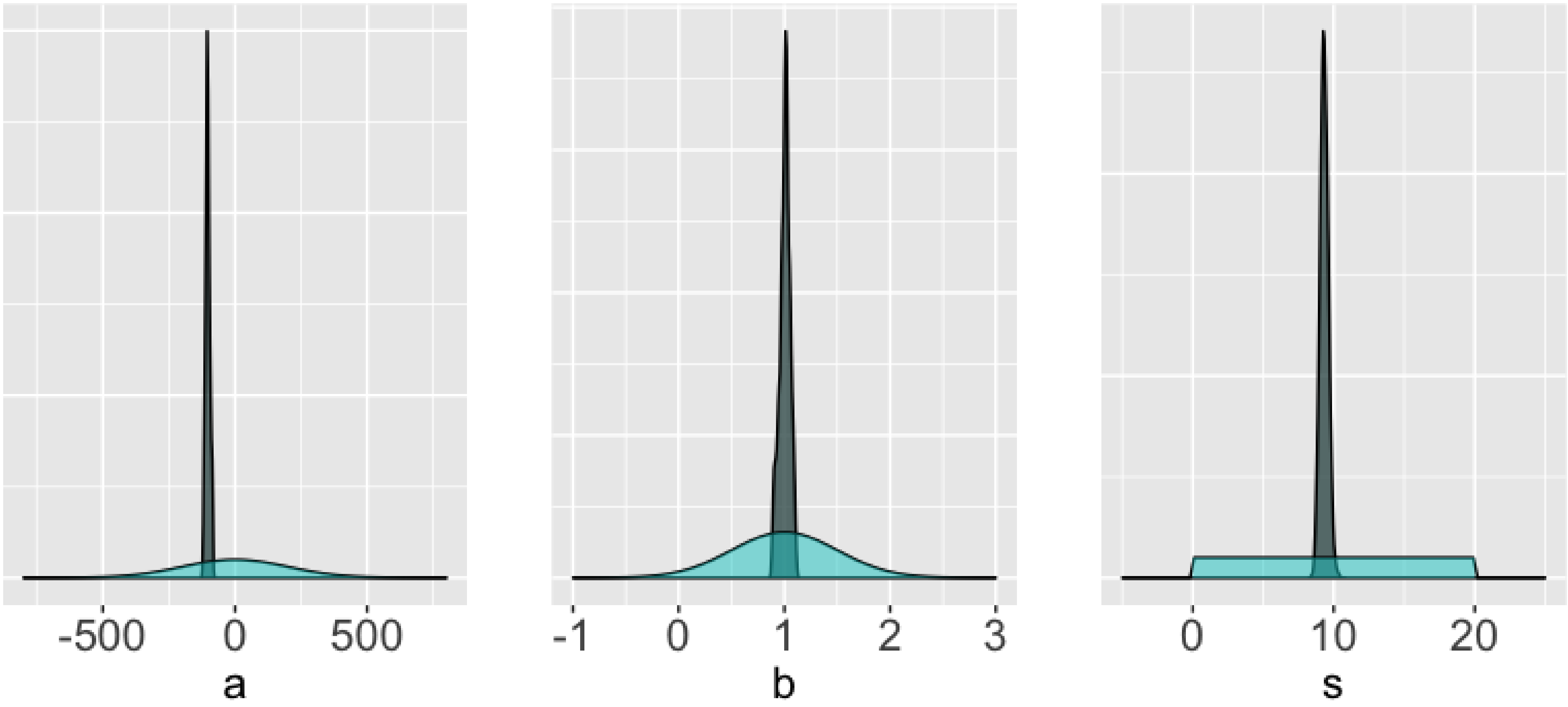

Trace of a
Density of a
Iterations
N = 10000 Bandwidth = 1.357
Trace of b
Density of b
Iterations
N = 10000 Bandwidth = 0.007956
Trace of s
Density of s
Iterations
N = 10000 Bandwidth = 0.04922

Addressing Markov chain instability

- Standardize the height predictor (subtract the mean and divide by the standard deviation).
- Increase chain length.


Trace of a
Iterations
Density of a
N = 100000 Bandwidth = 0.8291
Trace of b
Iterations
Density of b
N = 100000 Bandwidth = 0.004838
Trace of s
Iterations
Density of s
N = 100000 Bandwidth = 0.03126

Posterior insights



BAYESIAN MODELING WITH RJAGS

Let's practice!

BAYESIAN MODELING WITH RJAGS

Posterior estimation & inference

Alicia Johnson
Associate Professor, Macalester College

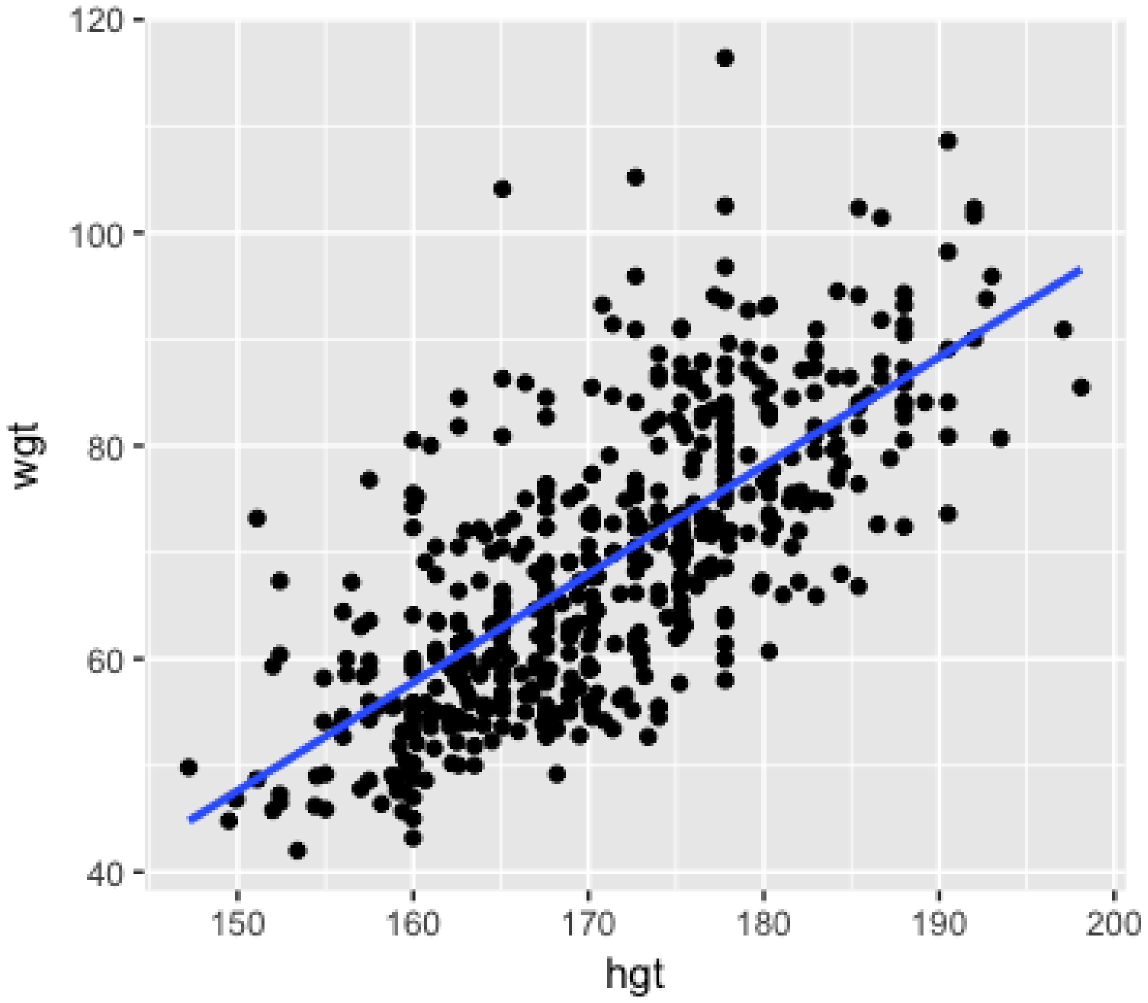
Bayesian regression model

Y_i = weight of adult i (kg)
X_i = height of adult i (cm)

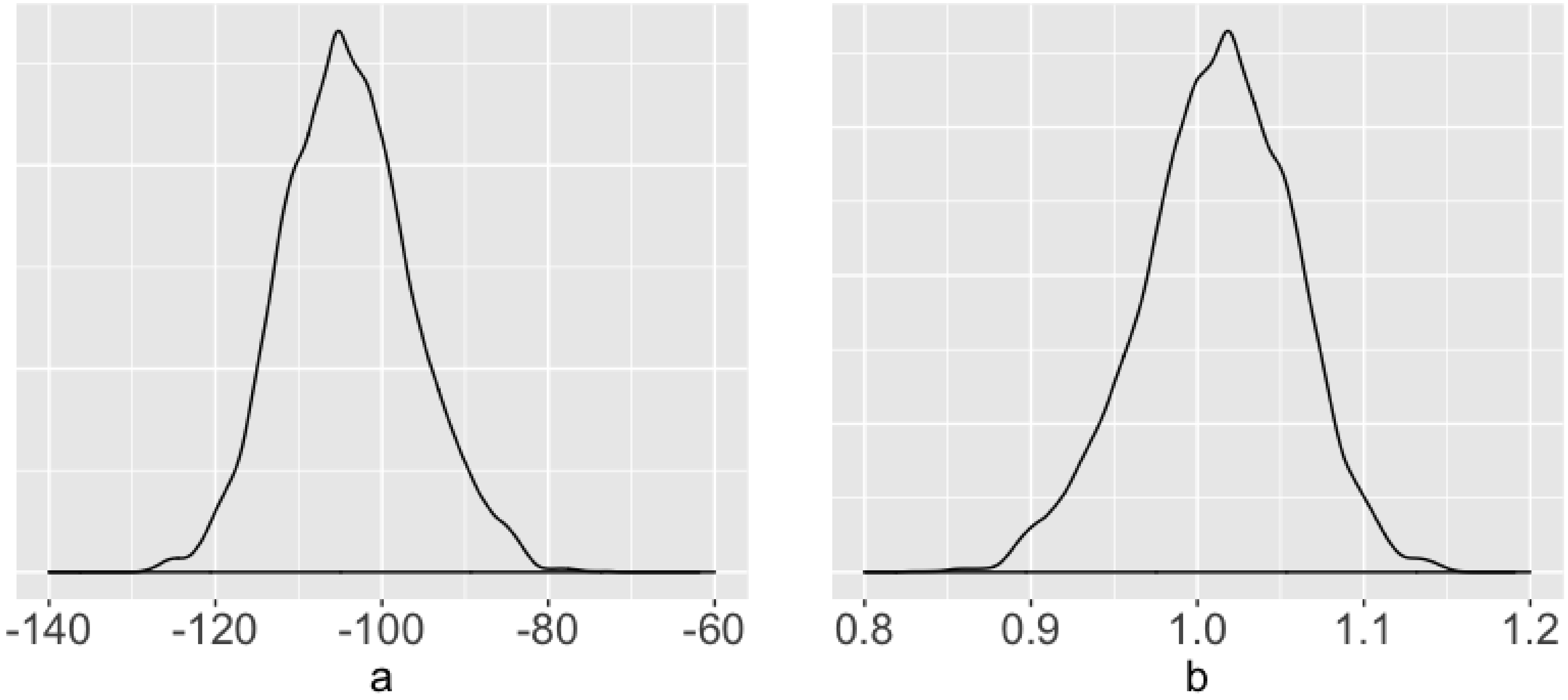
Model

$Y_i \sim N(m_i, s^2)$
$m_i = a + bX_i$

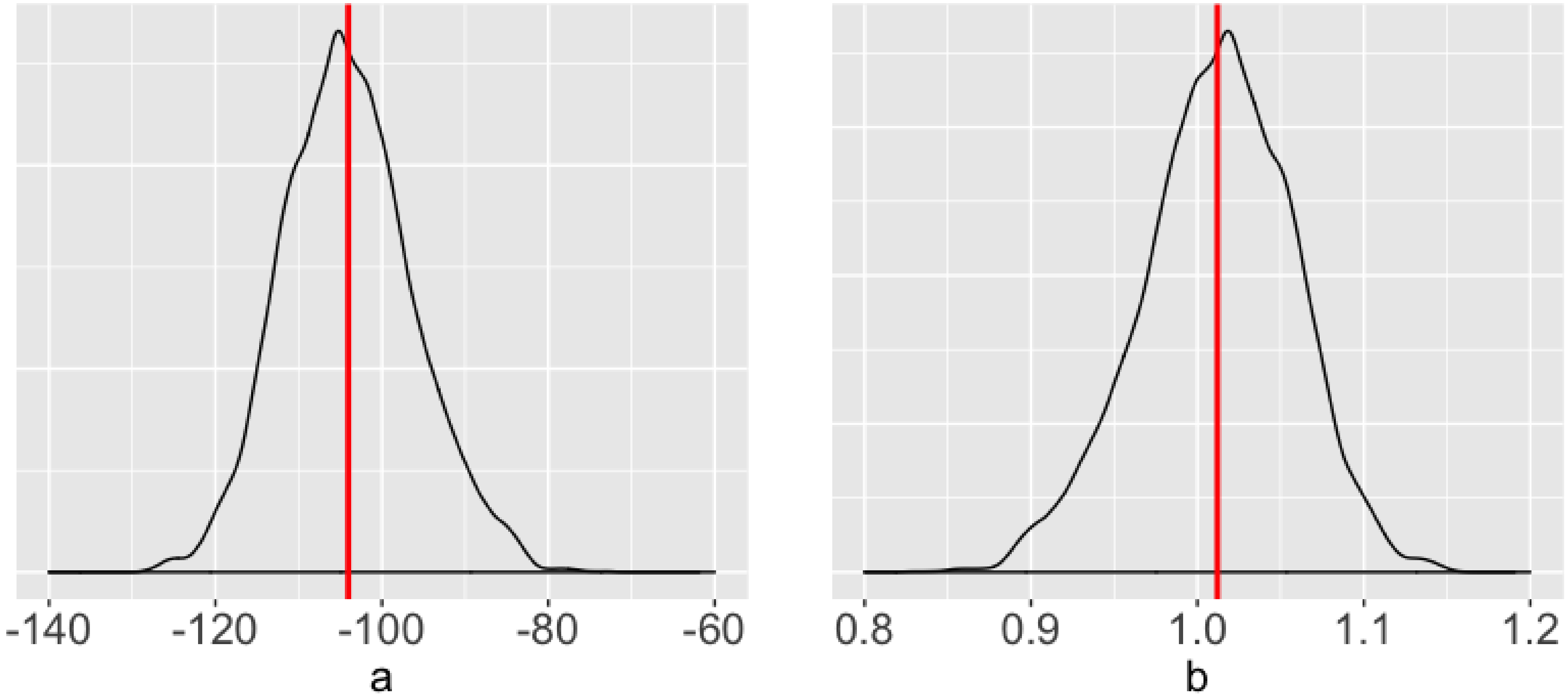
$a \sim N(0, 200^2)$
$b \sim N(1, 0.5^2)$
$s \sim Unif(0, 20)$



Posterior point estimation



Posterior point estimation



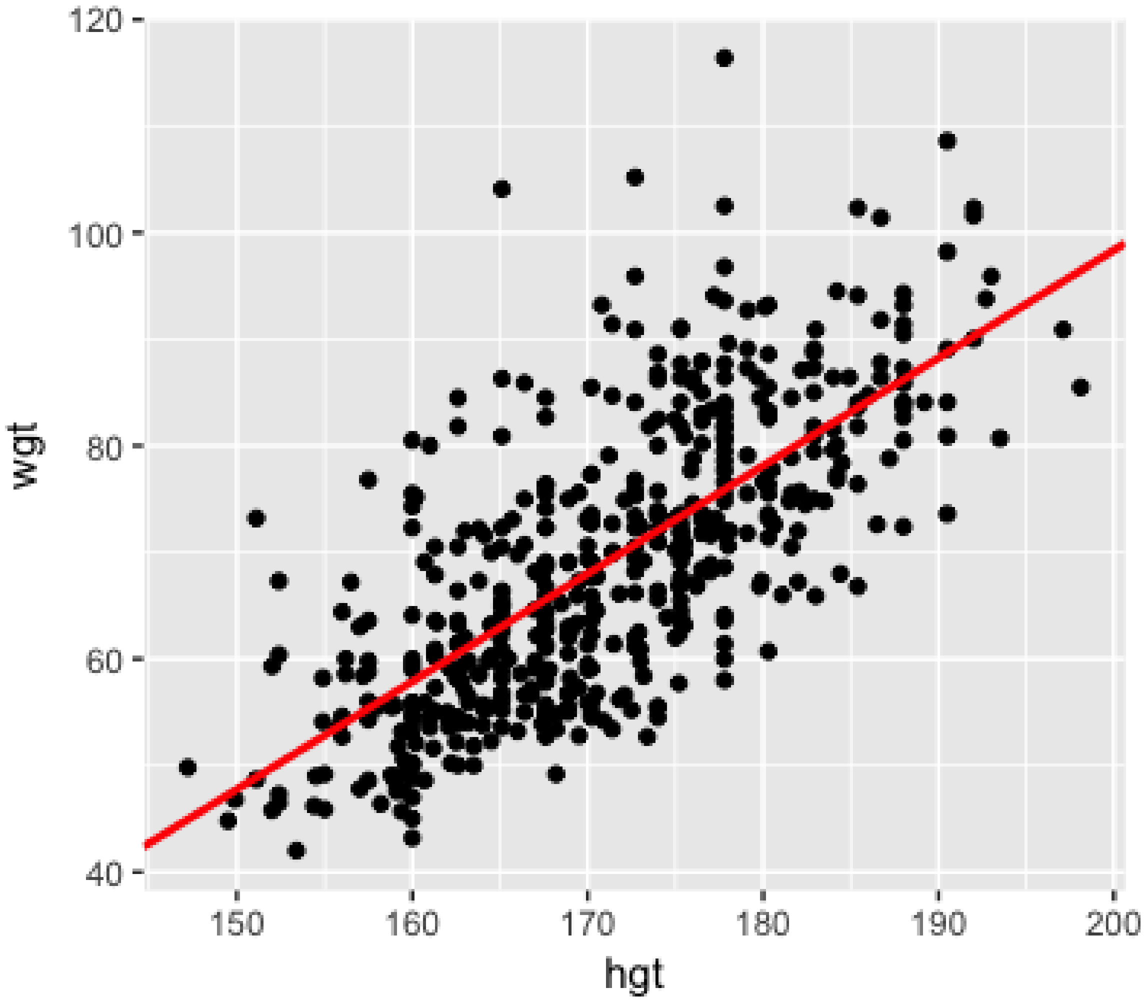
Posterior point estimation

```
> summary(weight_sim_big)

1. Empirical mean and standard deviation for each variable,
   plus standard error of the mean:

      Mean      SD  Naive SE Time-series SE
a -104.038 7.85296 0.0248332       0.661515
b    1.012 0.04581 0.0001449       0.003849
s    9.331 0.29495 0.0009327       0.001216

2. Quantiles for each variable:

       2.5%       25%      50%     75%   97.5%
a -118.6843 -109.5171 -104.365 -99.036 -87.470
b    0.9152    0.9828    1.014   1.044   1.098
s    8.7764    9.1284    9.322   9.524   9.933
```

posterior mean of $a \approx$ -104.038

posterior mean of $b \approx$ 1.012

Posterior point estimation



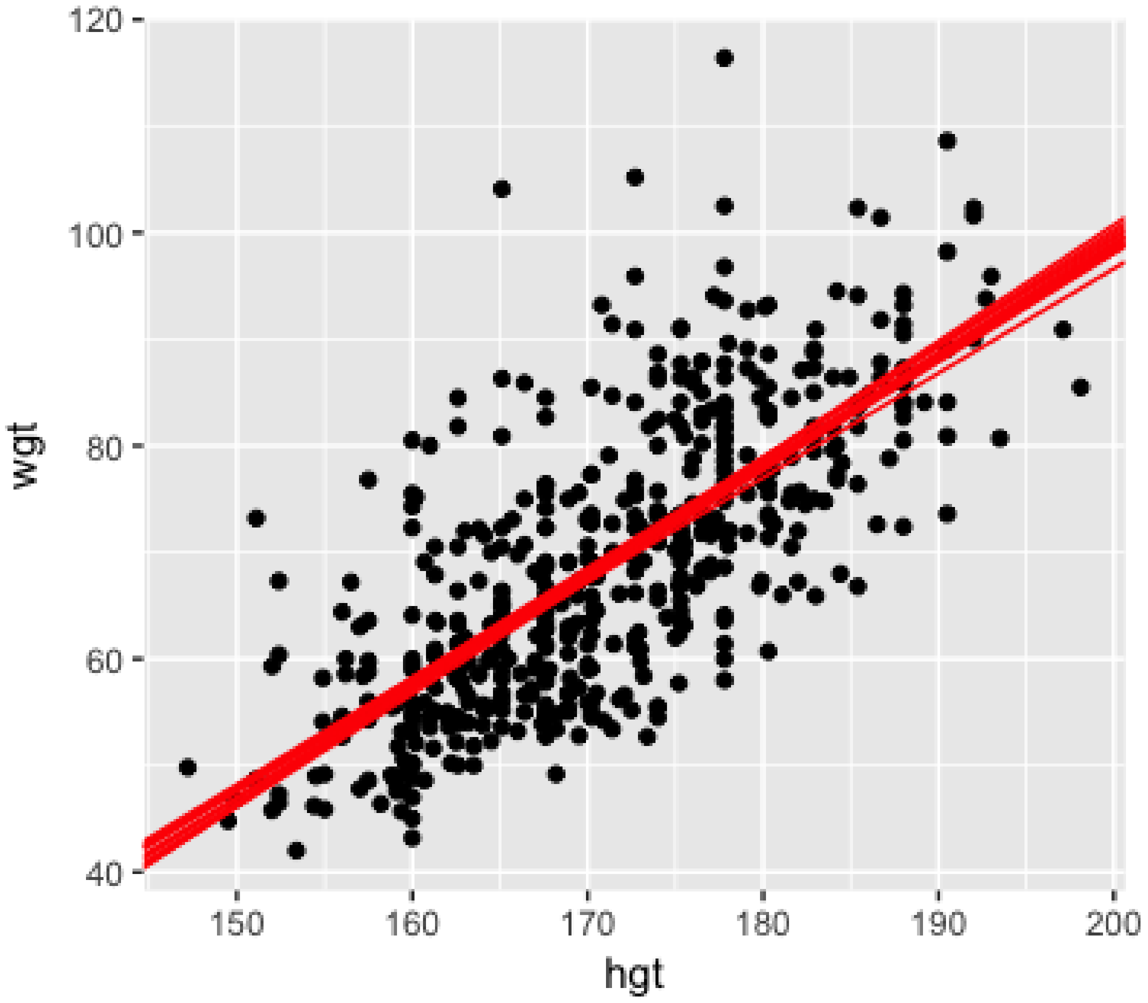
Posterior mean trend:

$$m_i = -104.038 + 1.012X_i$$

Markov chain output:

```
> head(weight_chains)
             a        b        s
[1,] -113.9029 1.072505 8.772007
[2,] -115.0644 1.077914 8.986393
[3,] -114.6958 1.077130 9.679812
[4,] -115.0568 1.072668 8.814403
[5,] -114.0782 1.071775 8.895299
[6,] -114.3271 1.069477 9.016185
```

Posterior uncertainty



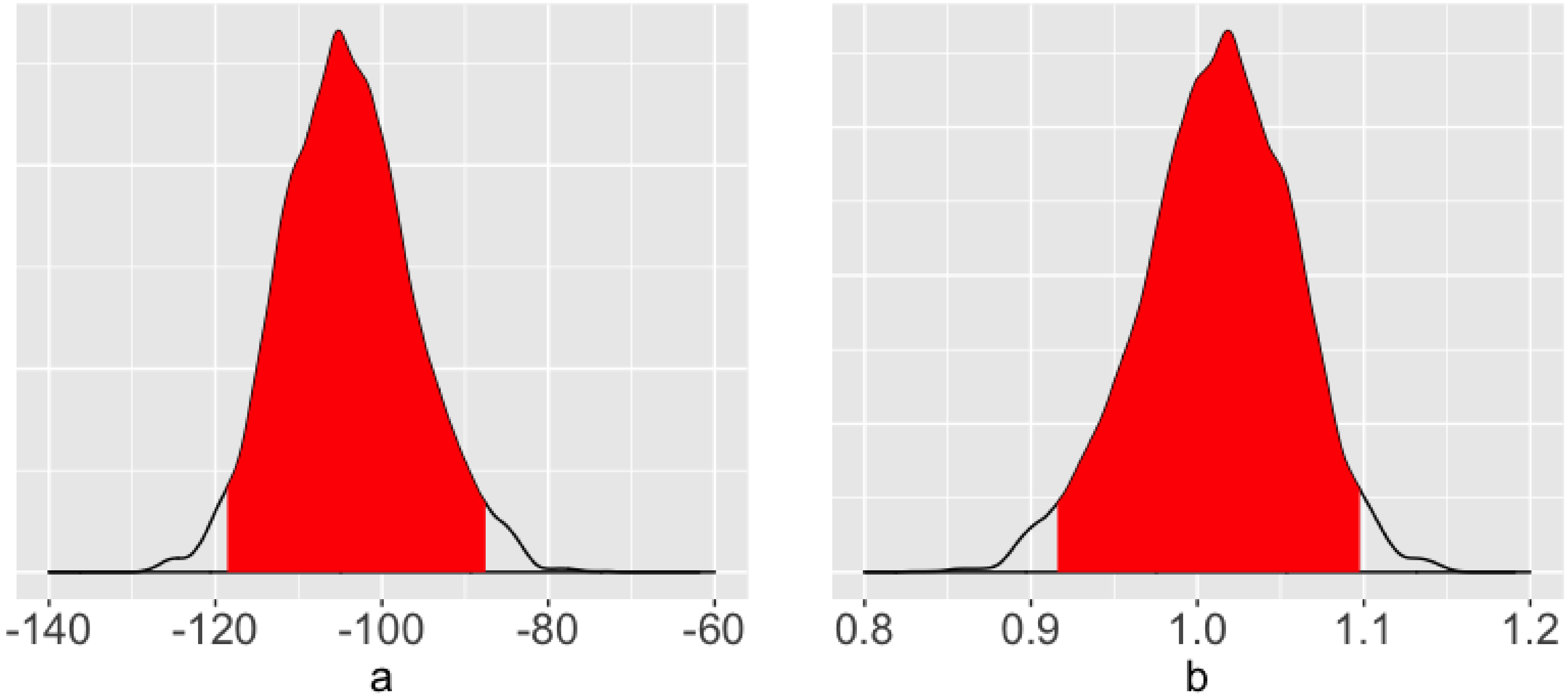
Posterior mean trend:

$$m_i = -104.038 + 1.012X_i$$

Markov chain output:

```
> head(weight_chains)
             a        b        s
[1,] -113.9029 1.072505 8.772007
[2,] -115.0644 1.077914 8.986393
[3,] -114.6958 1.077130 9.679812
[4,] -115.0568 1.072668 8.814403
[5,] -114.0782 1.071775 8.895299
[6,] -114.3271 1.069477 9.016185
```

Posterior credible intervals



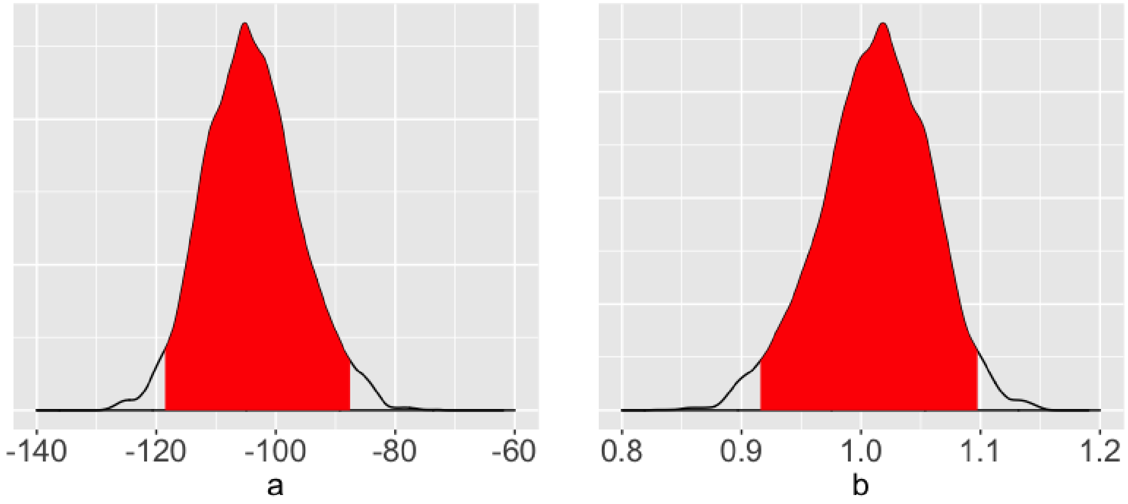
Posterior credible intervals

```
> summary(weight_sim_big)

1. Empirical mean and standard deviation for each variable,
   plus standard error of the mean:

       Mean       SD  Naive SE Time-series SE
a -104.038 7.85296 0.0248332       0.661515
b    1.012 0.04581 0.0001449       0.003849
s    9.331 0.29495 0.0009327       0.001216

2. Quantiles for each variable:

        2.5%       25%       50%      75%   97.5%
a -118.6843 -109.5171 -104.365 -99.036 -87.470
b    0.9152    0.9828    1.014   1.044   1.098
s    8.7764    9.1284    9.322   9.524   9.933
```

95% posterior credible interval for a: (-118.6843, -87.470)

95% posterior credible interval for b: (0.9152, 1.098)

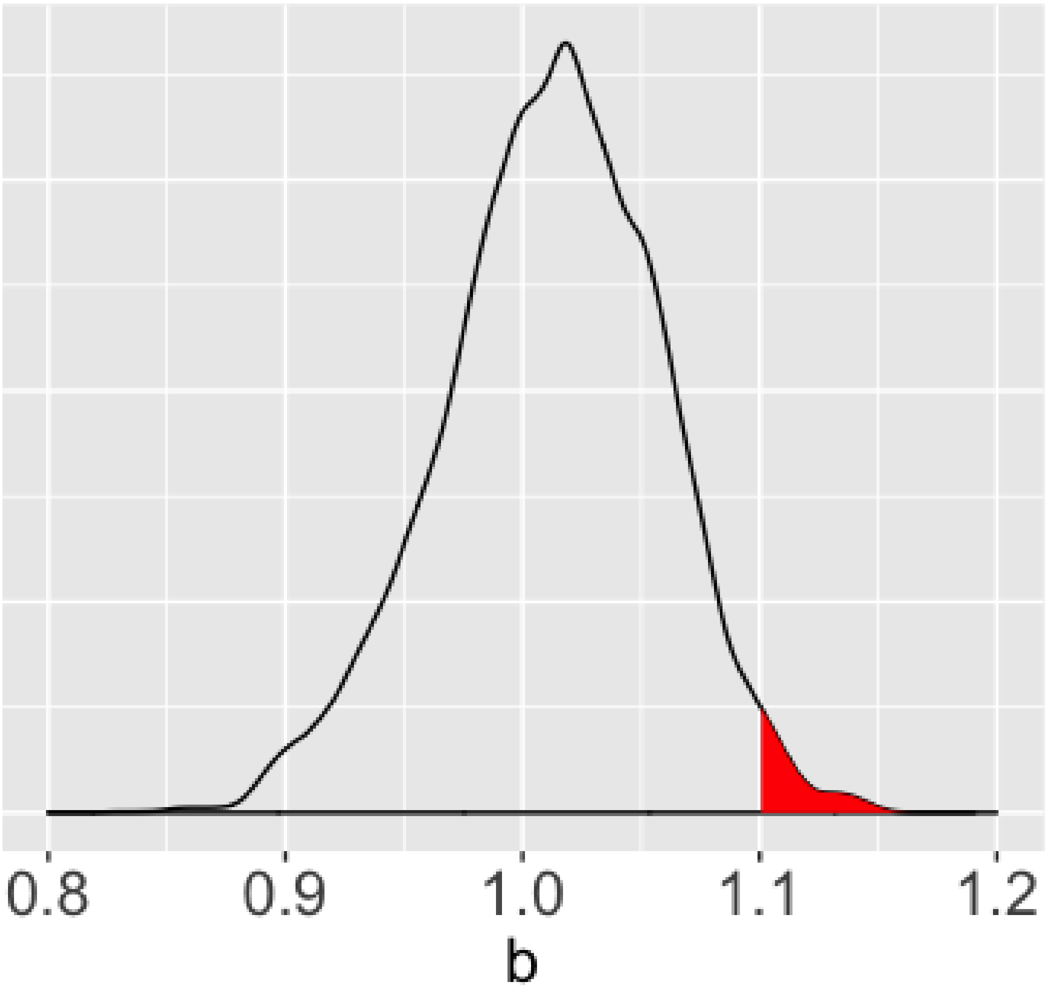
Posterior credible intervals



Interpretation

In light of our priors & observed data, there's a 95% (posterior) chance that b is between 0.9152 & 1.098 kg/cm.

Posterior probabilities




```
> table(weight_chains$b > 1.1)

FALSE  TRUE
97835  2165

> mean(weight_chains$b > 1.1)
[1] 0.02165
```

Interpretation:
There's a 2.165% posterior chance that b exceeds 1.1 kg/cm.

BAYESIAN MODELING WITH RJAGS

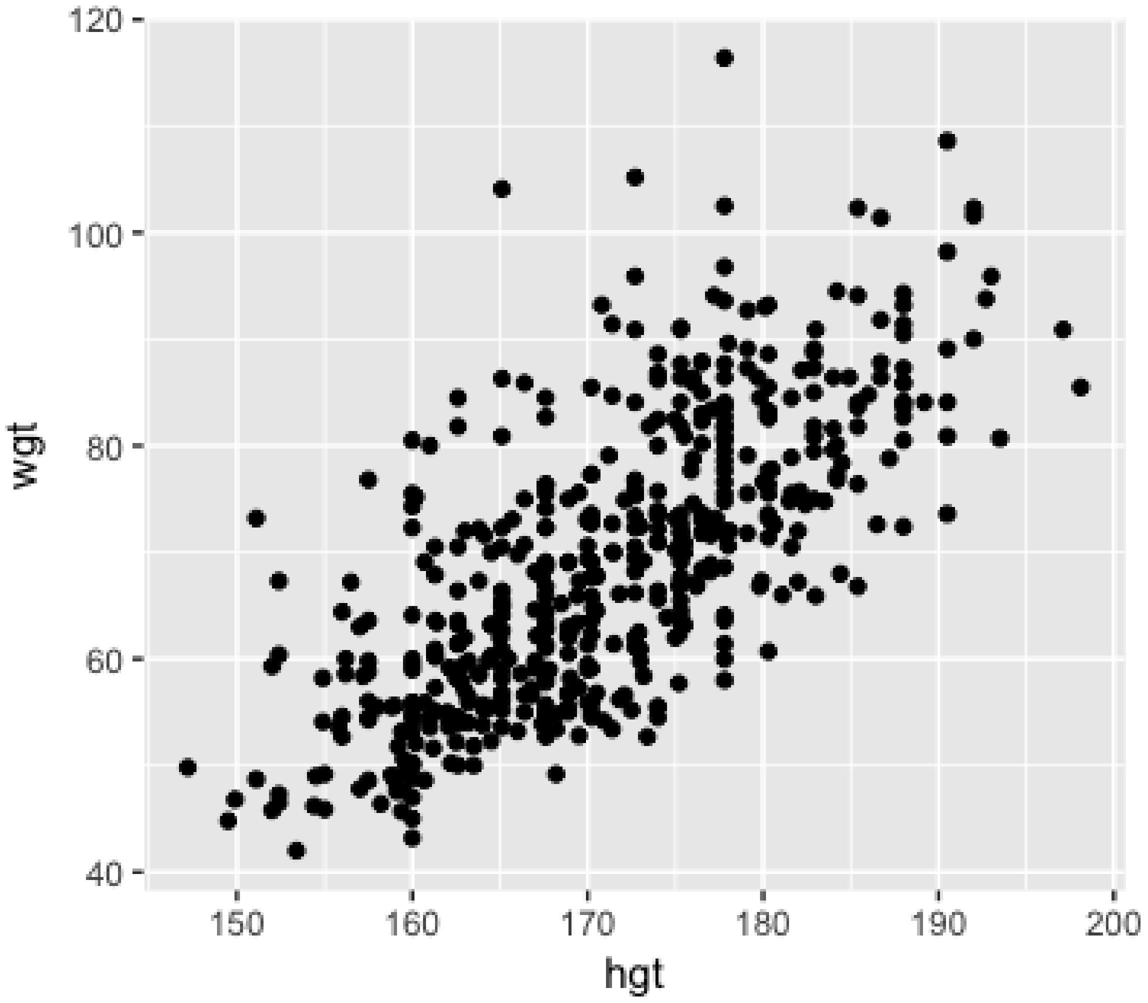
Let's practice!

BAYESIAN MODELING WITH RJAGS

Posterior prediction

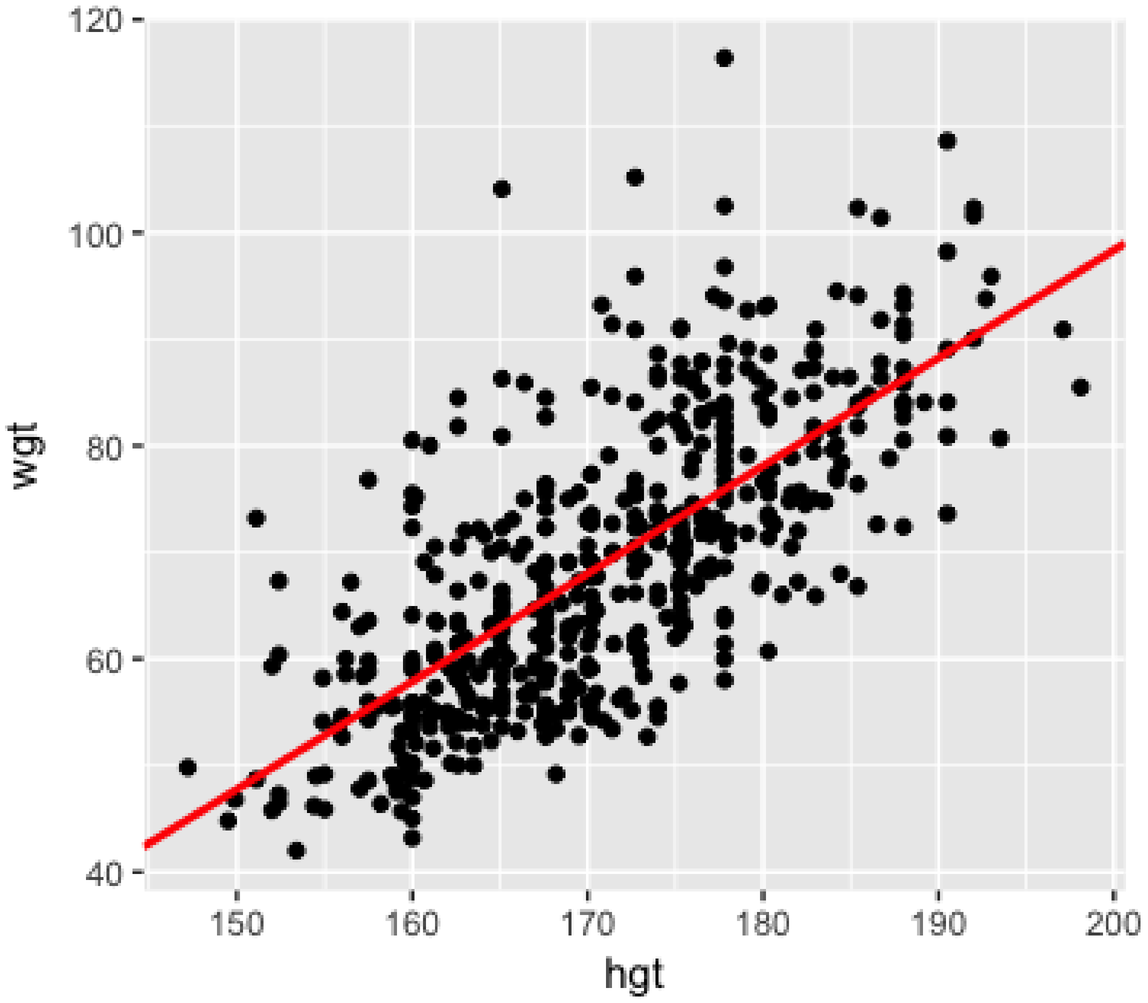
Alicia Johnson
Associate Professor, Macalester College

Posterior trend



$$Y \sim N(m, s^2)$$
$$m = a + bX$$

Posterior trend

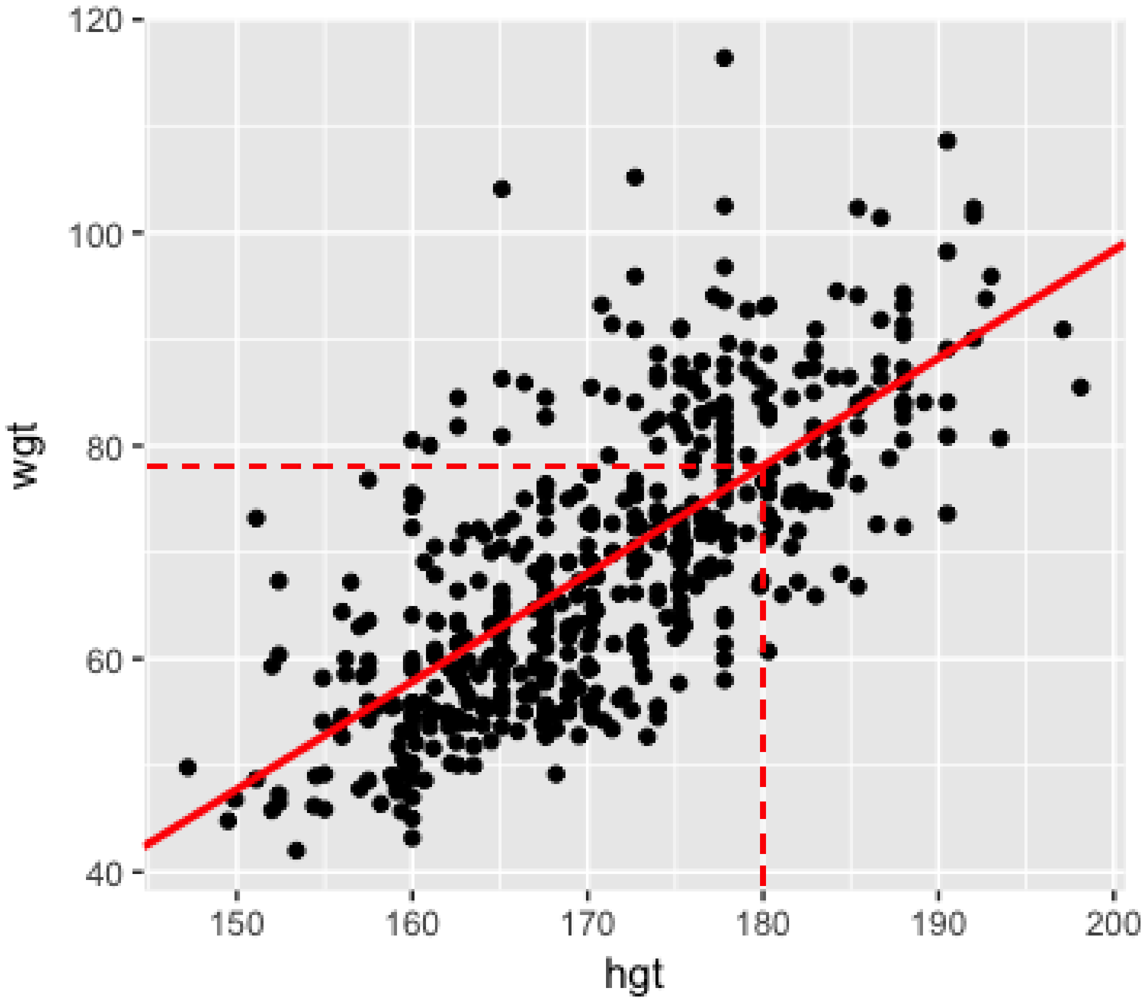


$$Y \sim N(m, s^2)$$
$$m = a + bX$$

Posterior mean trend

$$m = -104.038 + 1.012X$$

Posterior trend when height = 180 cm



$$Y \sim N(m, s^2)$$
$$m = a + bX$$

Posterior mean trend

$$m = -104.038 + 1.012X$$

```
> -104.038 + 1.012 * 180
[1] 78.122
```

Estimating posterior trend when height = 180 cm

```
> -104.038 + 1.012 * 180
[1] 78.122
```

```
> head(weight_chains)
          a        b        s
1 -113.9029 1.072505 8.772007
2 -115.0644 1.077914 8.986393
3 -114.6958 1.077130 9.679812
4 -115.0568 1.072668 8.814403
5 -114.0782 1.071775 8.895299
6 -114.3271 1.069477 9.016185
```

Estimating posterior trend when height = 180 cm

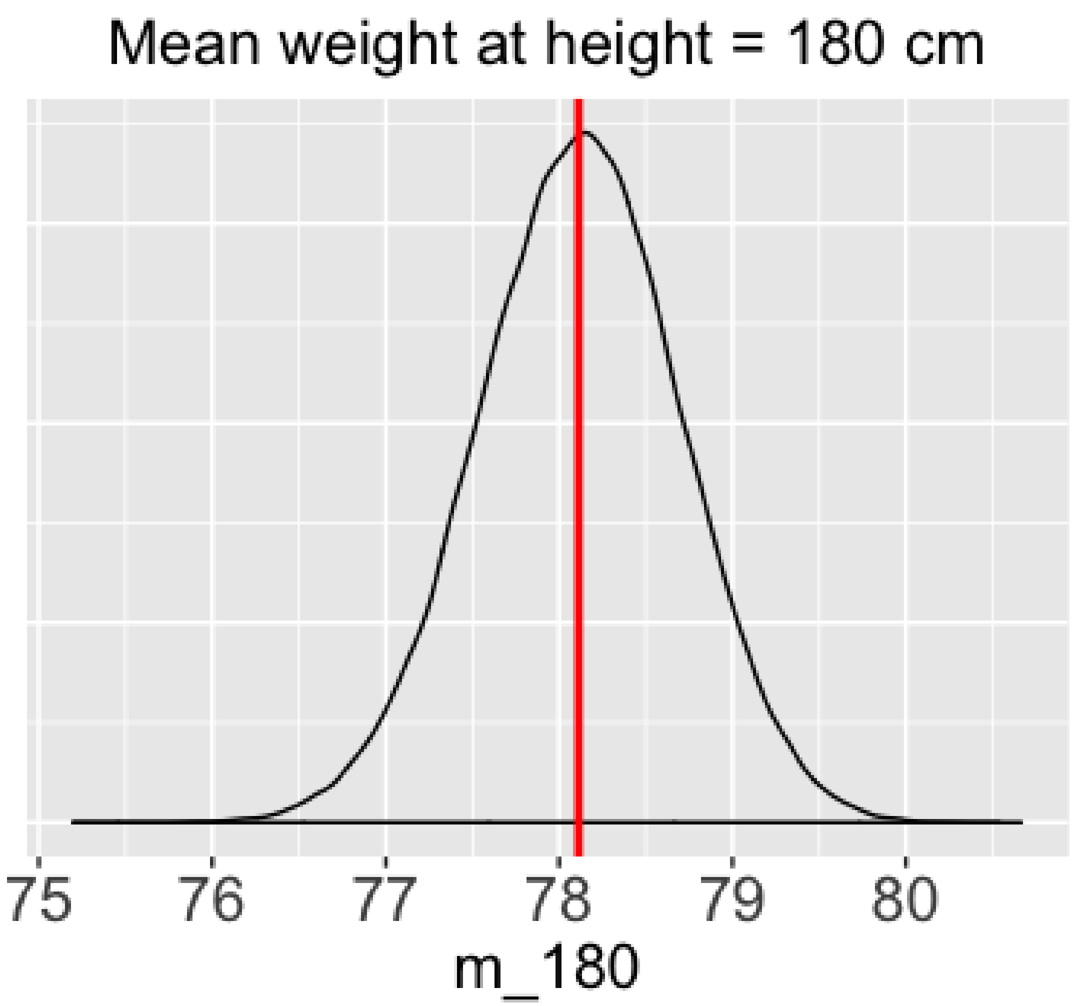
```
> -104.038 + 1.012 * 180
[1] 78.122
```

```
> weight_chains <- weight_chains  %>%
    mutate(m_180 = a + b * 180)

> head(weight_chains)
          a        b        s      m_180
1 -113.9029 1.072505 8.772007   79.14803
2 -115.0644 1.077914 8.986393   78.96014
3 -114.6958 1.077130 9.679812   79.18771
4 -115.0568 1.072668 8.814403   78.02352
5 -114.0782 1.071775 8.895299   78.84138
6 -114.3271 1.069477 9.016185   78.17877
```

```
> -113.9029 + 1.072505 * 180
[1] 79.148
```

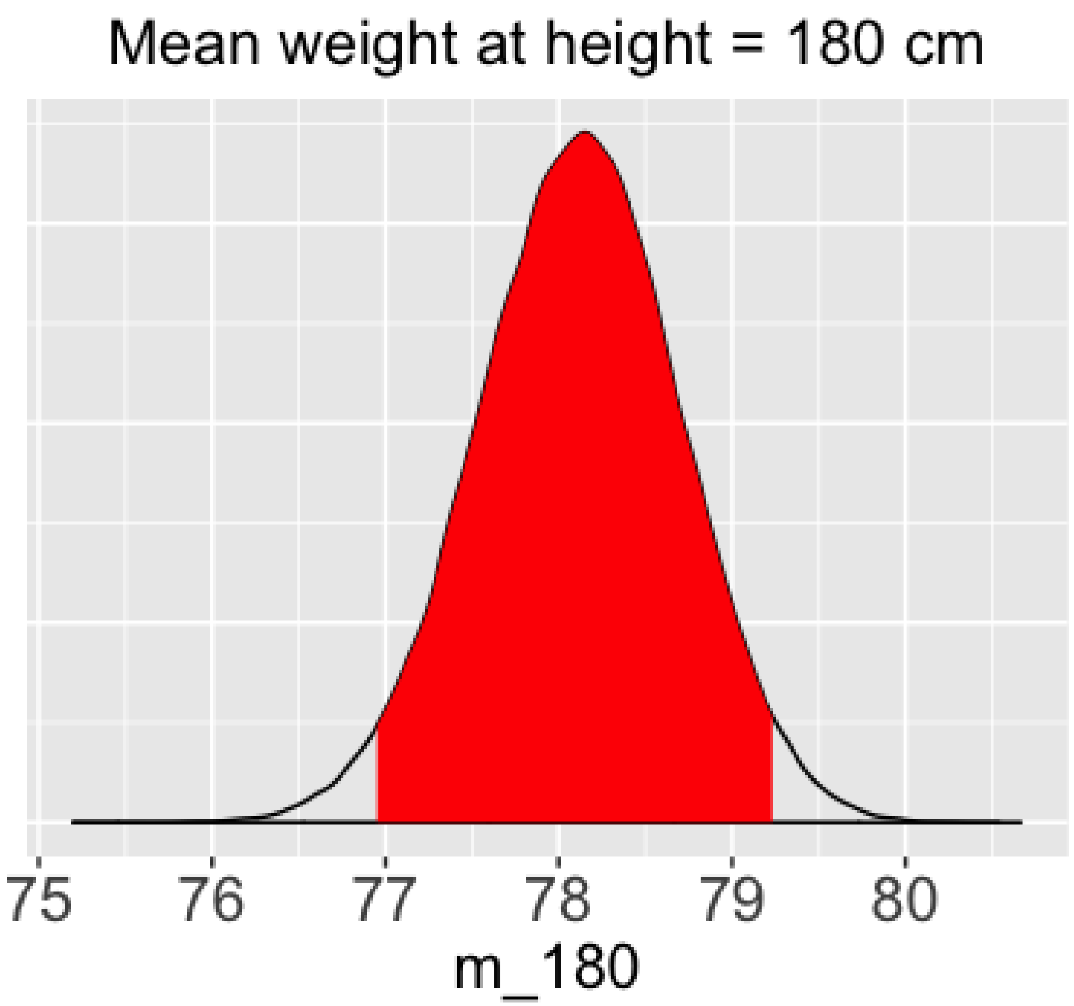
Posterior distribution of trend

```
> -104.038 + 1.012 * 180
[1] 78.122

> head(weight_chains$m_180)
[1] 79.14803
[2] 78.96014
[3] 79.18771
[4] 78.02352
[5] 78.84138
[6] 78.17877
```



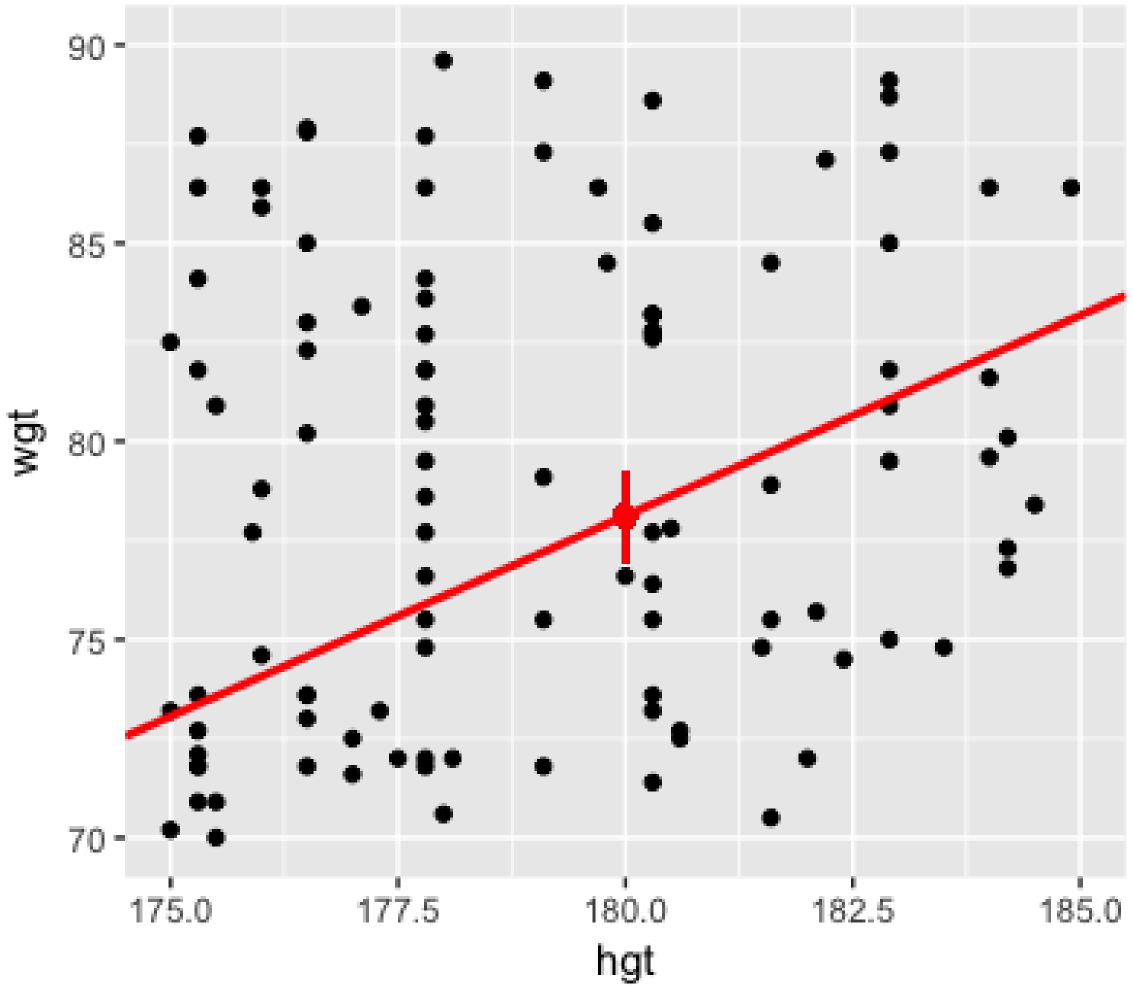
Credible interval for posterior trend

```
> -104.038 + 1.012 * 180
[1] 78.122

> head(weight_chains$m_180)
[1] 79.14803
[2] 78.96014
[3] 79.18771
[4] 78.02352
[5] 78.84138
[6] 78.17877

> quantile(weight_chains$m_180,
    c(0.025, 0.975))
    2.5%    97.5%
76.95054 79.23619
```

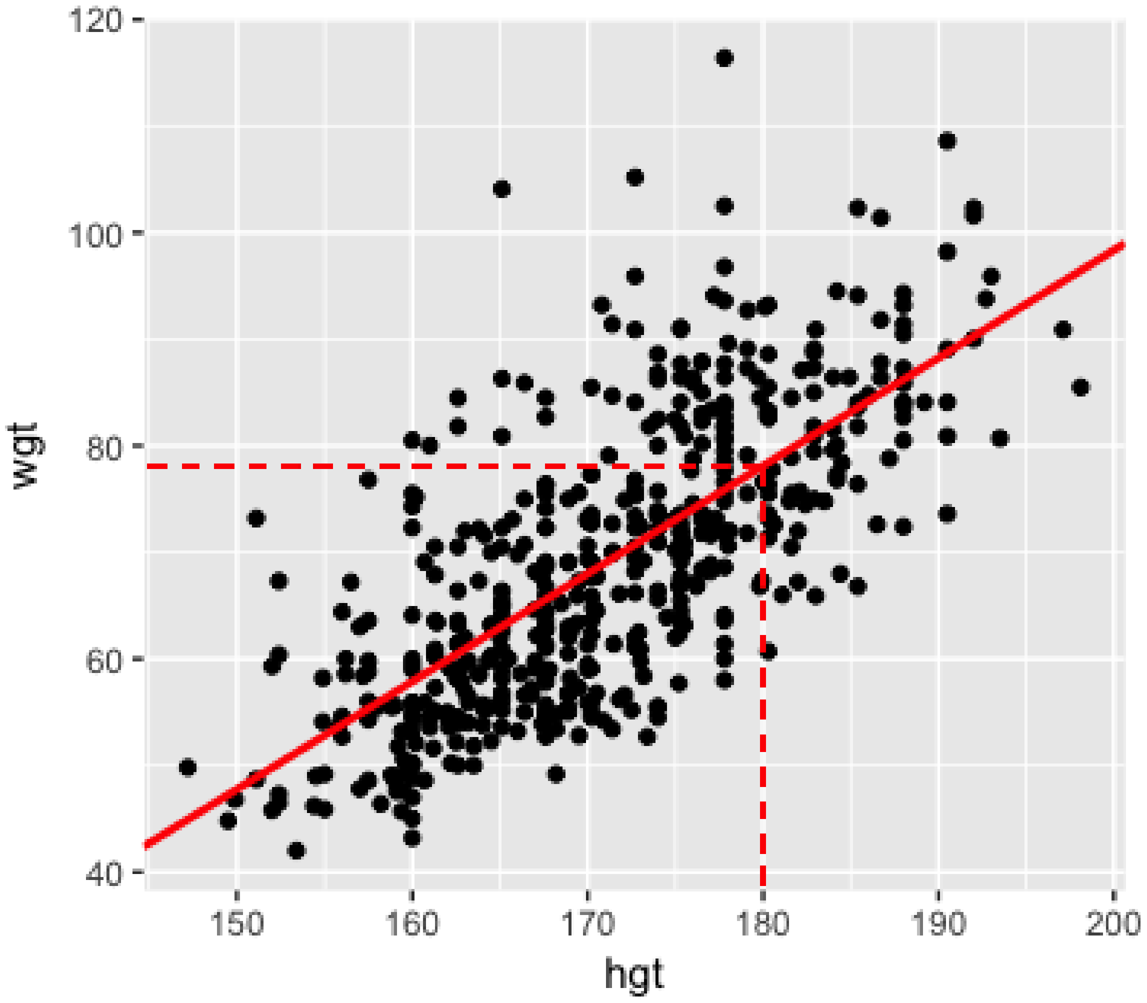


Visualizing posterior trend

```
> -104.038 + 1.012 * 180
[1] 78.122

> head(weight_chains$m_180)
[1] 79.14803
[2] 78.96014
[3] 79.18771
[4] 78.02352
[5] 78.84138
[6] 78.17877

> quantile(weight_chains$m_180,
    c(0.025, 0.975))
    2.5%    97.5%
76.95054 79.23619
```



Posterior trend vs posterior prediction



Posterior *mean* weight (or trend) among *all* 180 cm tall adults

```
> -104.038 + 1.012 * 180
[1] 78.122
```

Posterior *predicted* weight of a *specific* 180 cm tall adult

```
> -104.038 + 1.012 * 180
[1] 78.122
```

Predicting weight when height = 180 cm

$$Y \sim N(m_{180}, s^2)$$

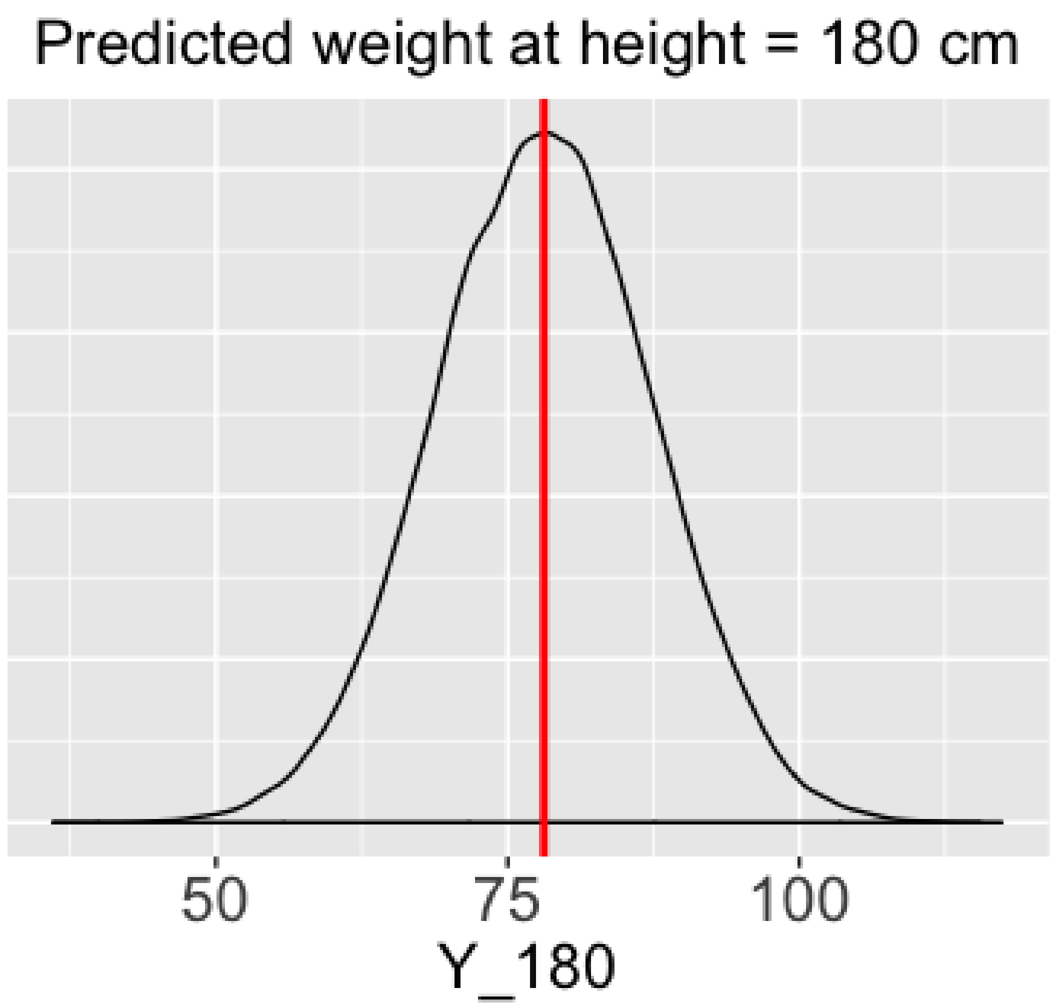
$$m_{180} = a + b * 180$$

```
> head(weight_chains, 3)
          a        b        s     m_180
1 -113.9029 1.072505 8.772007  79.14803
2 -115.0644 1.077914 8.986393  78.96014
3 -114.6958 1.077130 9.679812  79.18771
```

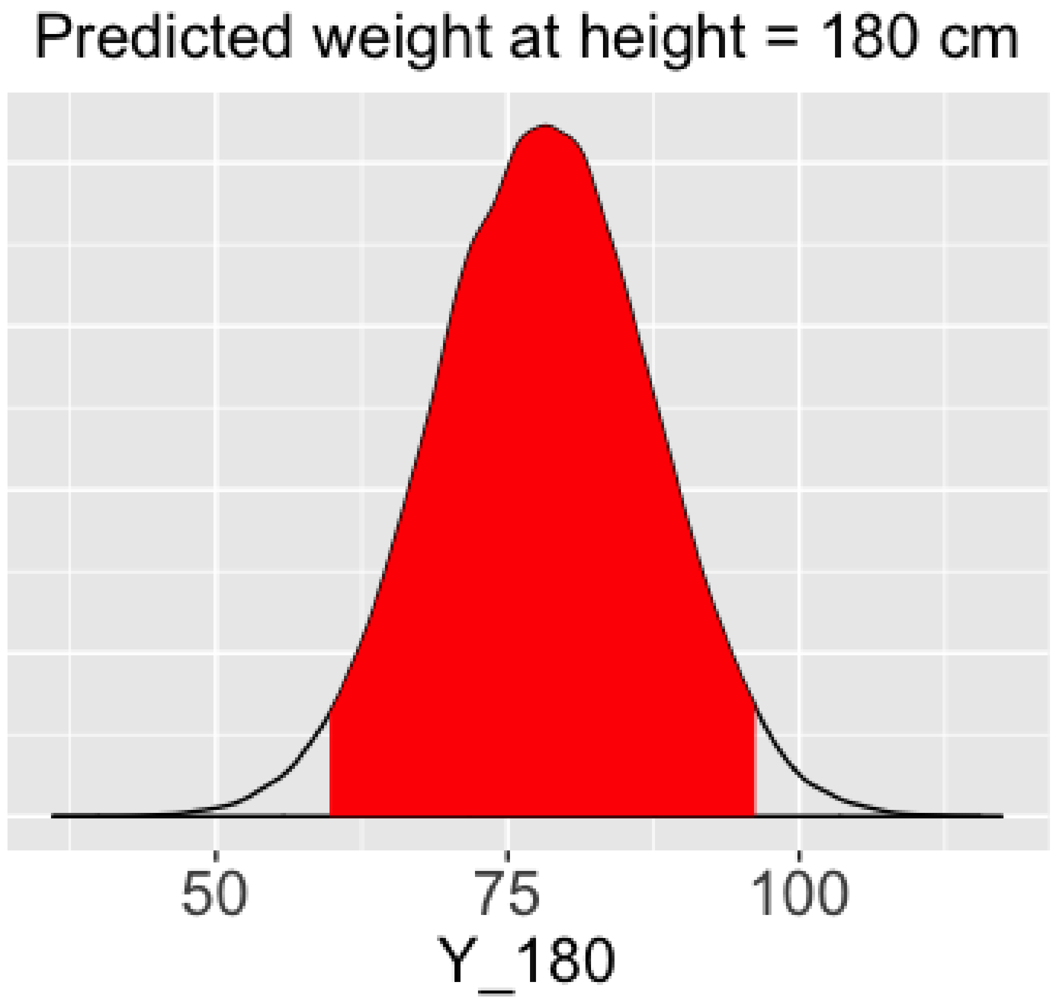
```
> set.seed(2000)

> rnorm(n = 1, mean = 79.14803, sd = 8.772007)
[1] 71.65811
```

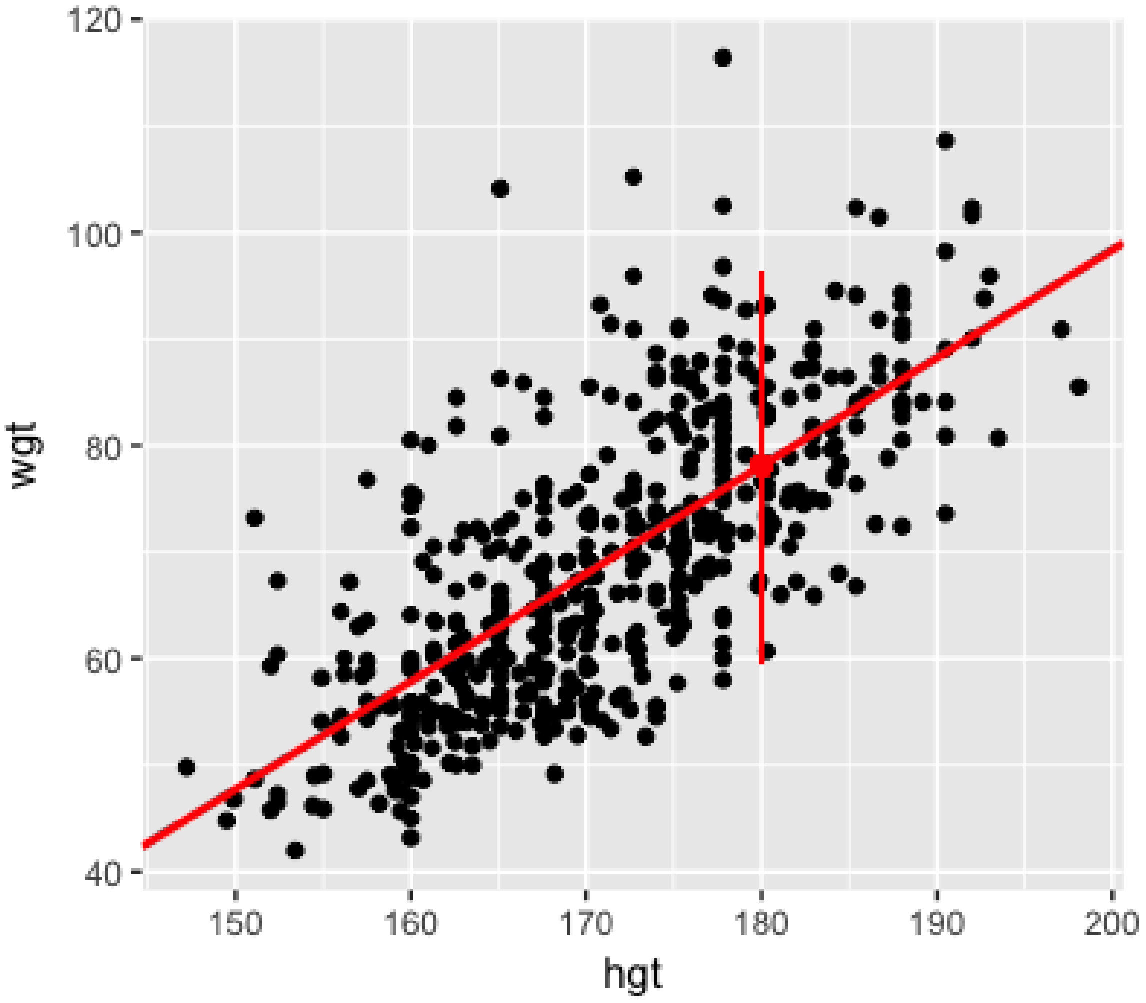
```
> rnorm(n = 1, mean = 78.96014, sd = 8.986393)
[1] 75.78894

> rnorm(n = 1, mean = 79.18771, sd = 9.679812)
[1] 87.80419
```

Posterior predictive distribution



Posterior prediction interval





BAYESIAN MODELING WITH RJAGS

Let's practice!