Linear regression

Frank Edwards

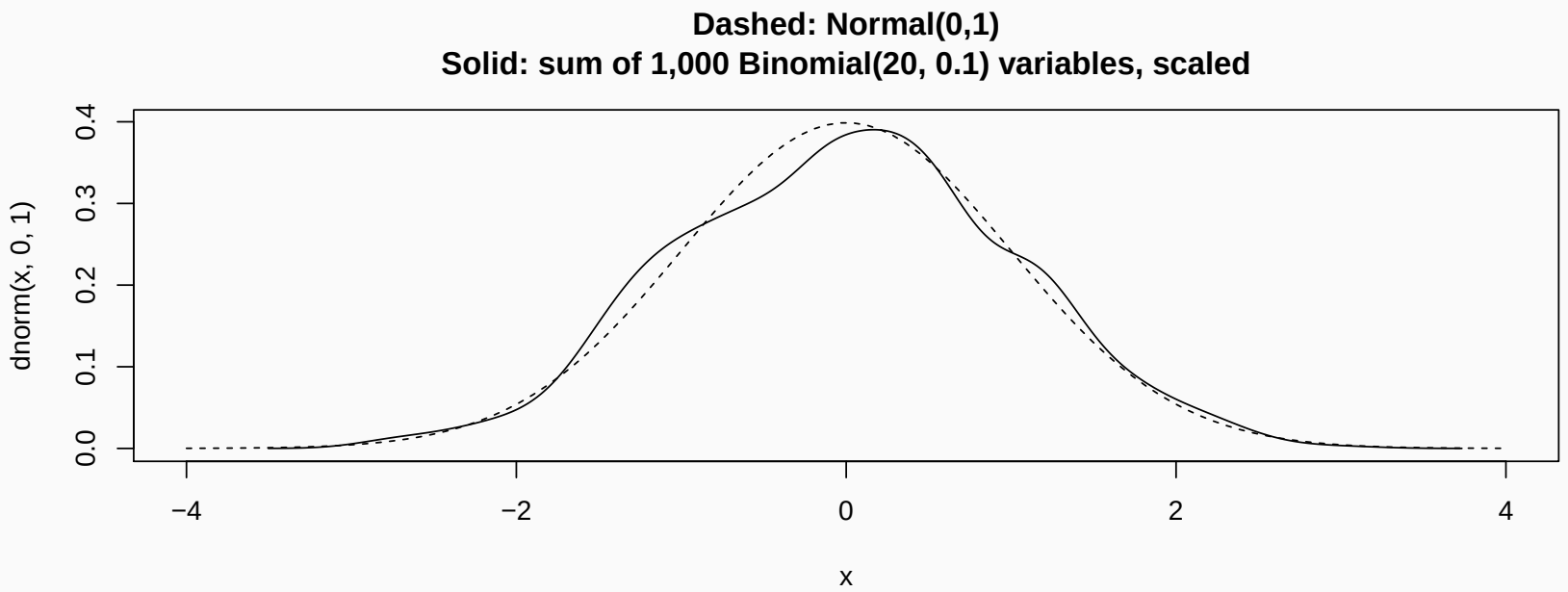
2/7/2020

Is reality linear?

- Linear regression is the dominant model in social science
- It is (obviously) an inadequate scientific model
- But still useful

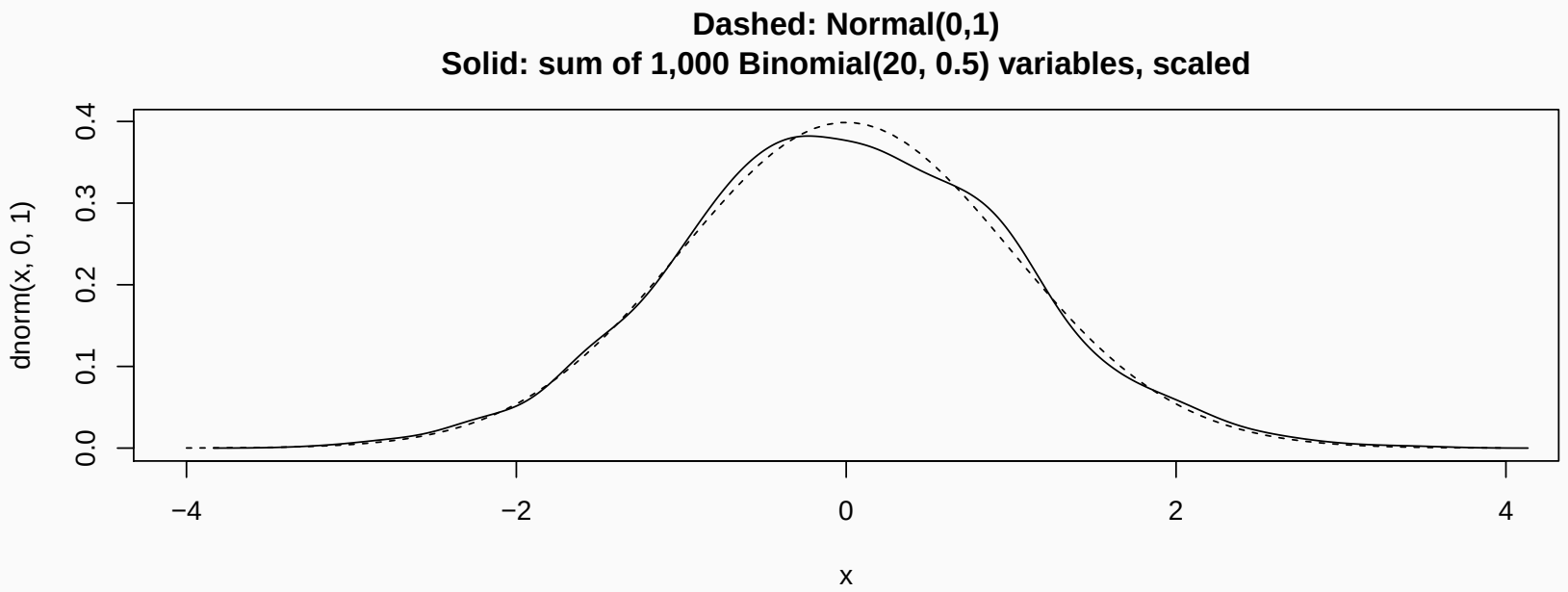
Why Normal distributions are so common

```
output<-list()
for(i in 1:1000){output[[i]]<-rbinom(1000, 20, 0.1)}
combined<-bind_cols(output)
combined_sum<-apply(combined, 1, sum)
x<-seq(-4, 4, length.out=100)
plot(x, dnorm(x, 0, 1),
     lty = 2, type = "l",
     main = "Dashed: Normal(0,1)\nSolid: sum of 1,000 Binomial(20, 0.1) variables, scaled")
lines(density(scale(combined_sum)))
```



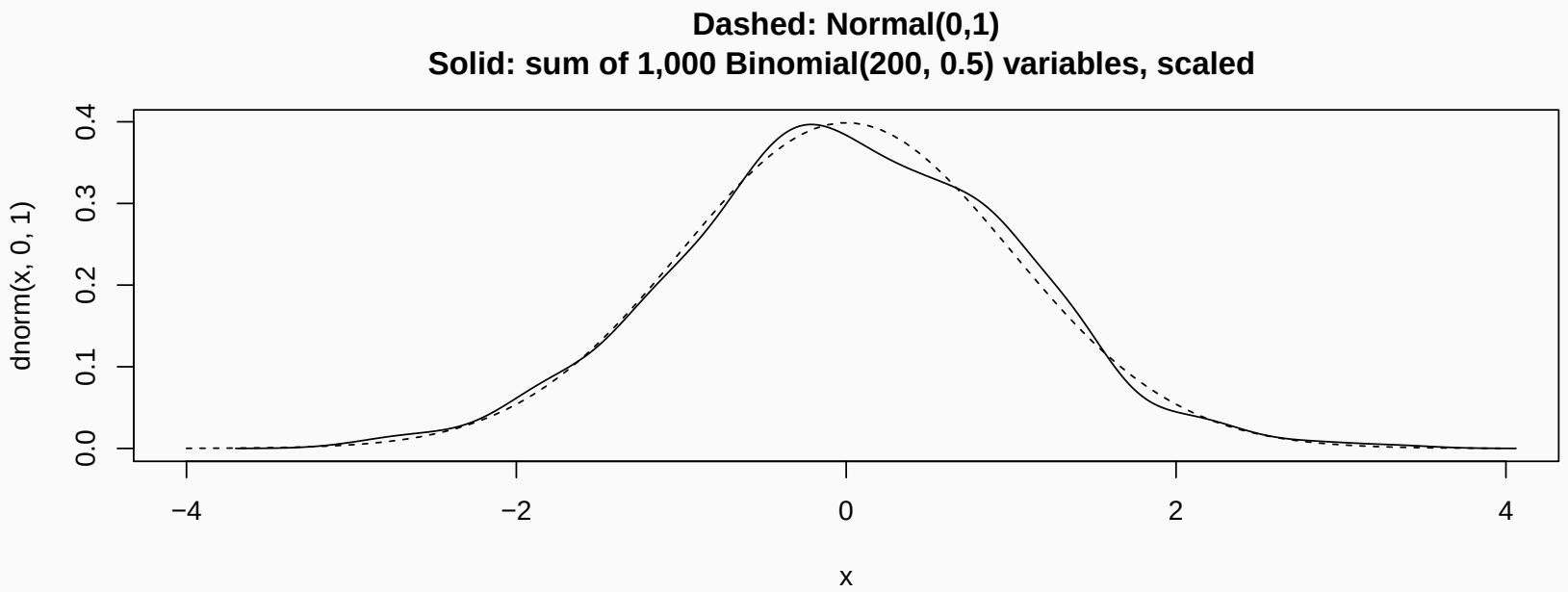
Why Normal distributions are so common

```
output<-list()
for(i in 1:1000){output[[i]]<-rbinom(1000, 20, 0.5)}
combined<-bind_cols(output)
combined_sum<-apply(combined, 1, sum)
x<-seq(-4, 4, length.out=100)
plot(x, dnorm(x, 0, 1),
     lty = 2,
     type = "l",
     main = "Dashed: Normal(0,1)\nSolid: sum of 1,000 Binomial(20, 0.5) variables, scaled")
lines(density(scale(combined_sum)))
```



Why Normal distributions are so common

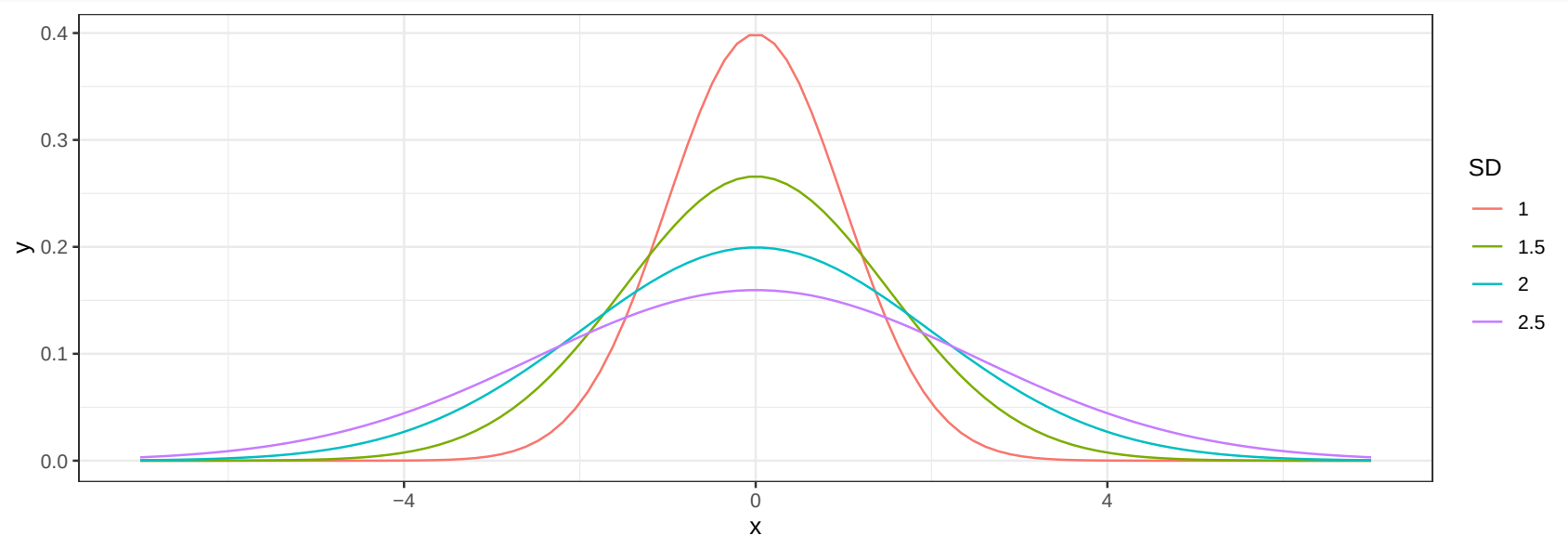
```
output<-list()
for(i in 1:1000){output[[i]]<-rbinom(1000, 200, 0.5)}
combined<-bind_cols(output)
combined_sum<-apply(combined, 1, sum)
x<-seq(-4, 4, length.out=100)
plot(x, dnorm(x, 0, 1),
     lty = 2, type = "l",
     main = "Dashed: Normal(0,1)\nSolid: sum of 1,000 Binomial(200, 0.5) variables, scaled ")
lines(density(scale(combined_sum)))
```



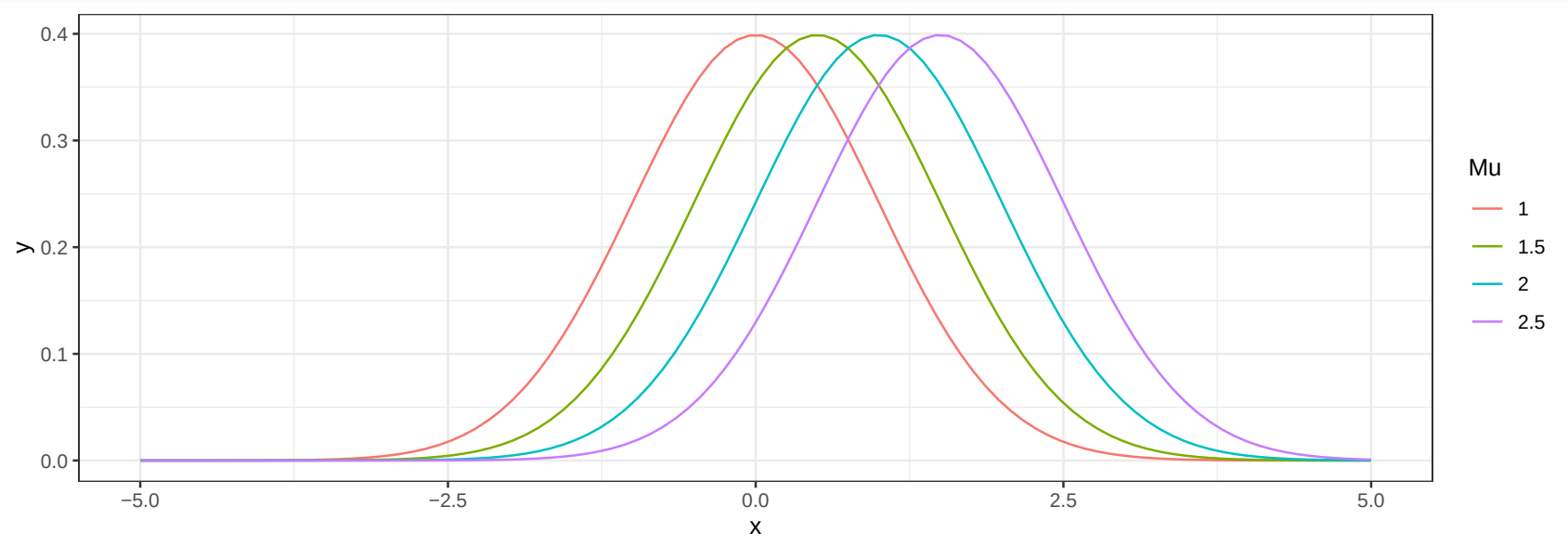
Getting to know the Normal

- Normal (Gaussian) have two parameters: mean (μ) and standard deviation (σ)
- These two parameters describe the location and spread of the distribution

```r
xrange<-seq(-7,7,length.out=100)

normals<-data.frame(x = rep(xrange, 4),
                    y = c(dnorm(xrange, 0, 1),
                          dnorm(xrange, 0, 1.5),
                          dnorm(xrange, 0, 2),
                          dnorm(xrange, 0, 2.5)),
                    SD = rep(c("1", "1.5", "2", "2.5"), each = 100))

ggplot(normals, aes(x = x,y = y, color = SD)) +
  geom_line()
```

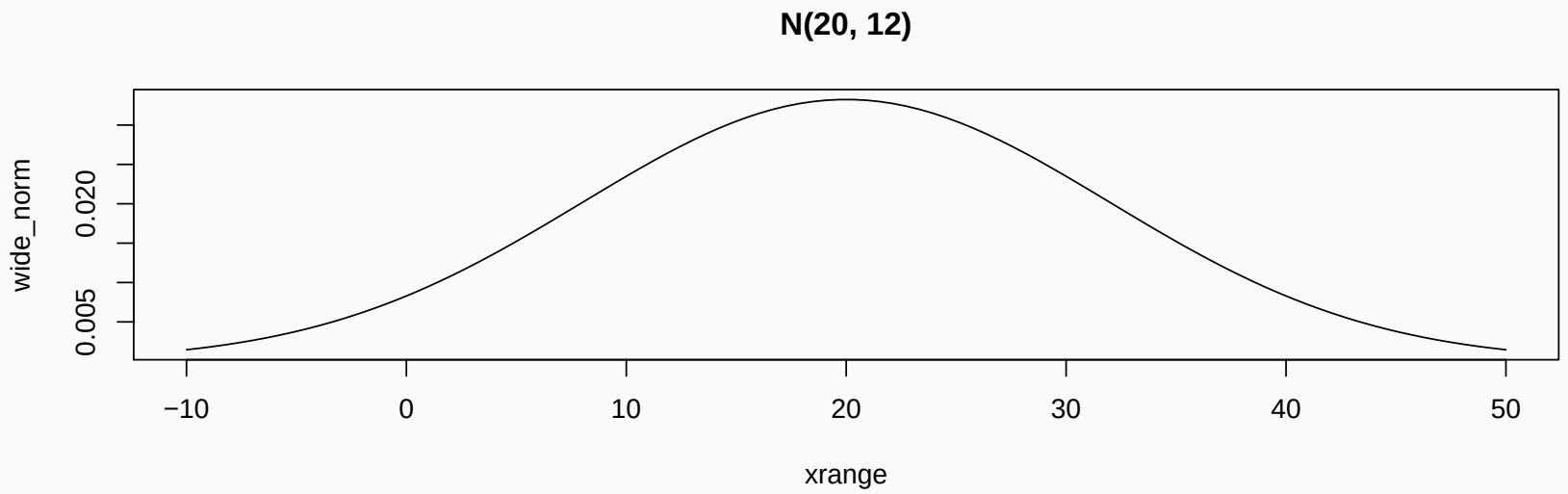


```r
xrange<-seq(-5,5,length.out=100)

normals<-data.frame(x = rep(xrange, 4),
                    y = c(dnorm(xrange, 0, 1),
                          dnorm(xrange, 0.5, 1),
                          dnorm(xrange, 1, 1),
                          dnorm(xrange, 1.5, 1)),
                    Mu = rep(c("1", "1.5", "2", "2.5"), each = 100))

ggplot(normals, aes(x = x,y = y, color = Mu)) +
  geom_line()
```



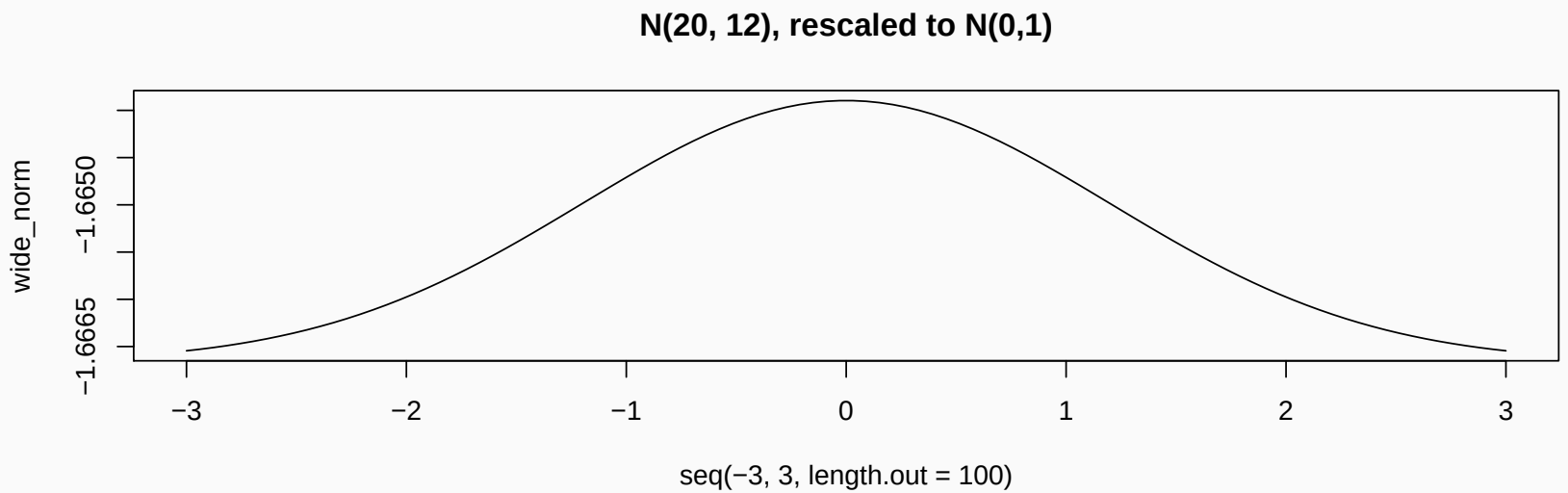
Fun fact: every normal can be a N(0,1)

```r
xrange<-seq(-10,50,length.out=100)
wide_norm<-dnorm(xrange, mean = 20, sd = 12)
plot(xrange, wide_norm, type ="l",
     main = "N(20, 12)")
```



Rescaling N(20,12)

```
wide_norm<-(wide_norm-20)/12
## this is the same result as scale(wide_norm)
## also called z-scores
plot(seq(-3,3,length.out=100), wide_norm, type ="l",
     main = "N(20, 12), rescaled to N(0,1)")
```



Model notation: a primer

Likelihood: $W \sim Binomial(N, p)$

Prior: $p \sim Uniform(0, 1)$

Model notation: a primer

Likelihood: $W \sim Binomial(N, p)$

Prior: $p \sim Uniform(0, 1)$

Note the relationship between W and p.

Building a model of human height

```
library(rethinking)
data("Howell1")
d<-Howell1
head(d)
```

```
##    height   weight age male
## 1 151.765 47.82561  63    1
## 2 139.700 36.48581  63    0
## 3 136.525 31.86484  65    0
## 4 156.845 53.04191  41    1
## 5 145.415 41.27687  51    0
## 6 163.830 62.99259  35    1
```

Building a model of human height

$$h_i \sim \text{Normal}(\mu, \sigma)$$

Building a model of human height

$$h_i \sim \text{Normal}(\mu, \sigma)$$

What are the parameters in this model?

Building a model of human height

$$h_i \sim \text{Normal}(\mu, \sigma)$$

What are the parameters in this model?

What is our next step?

Set priors!

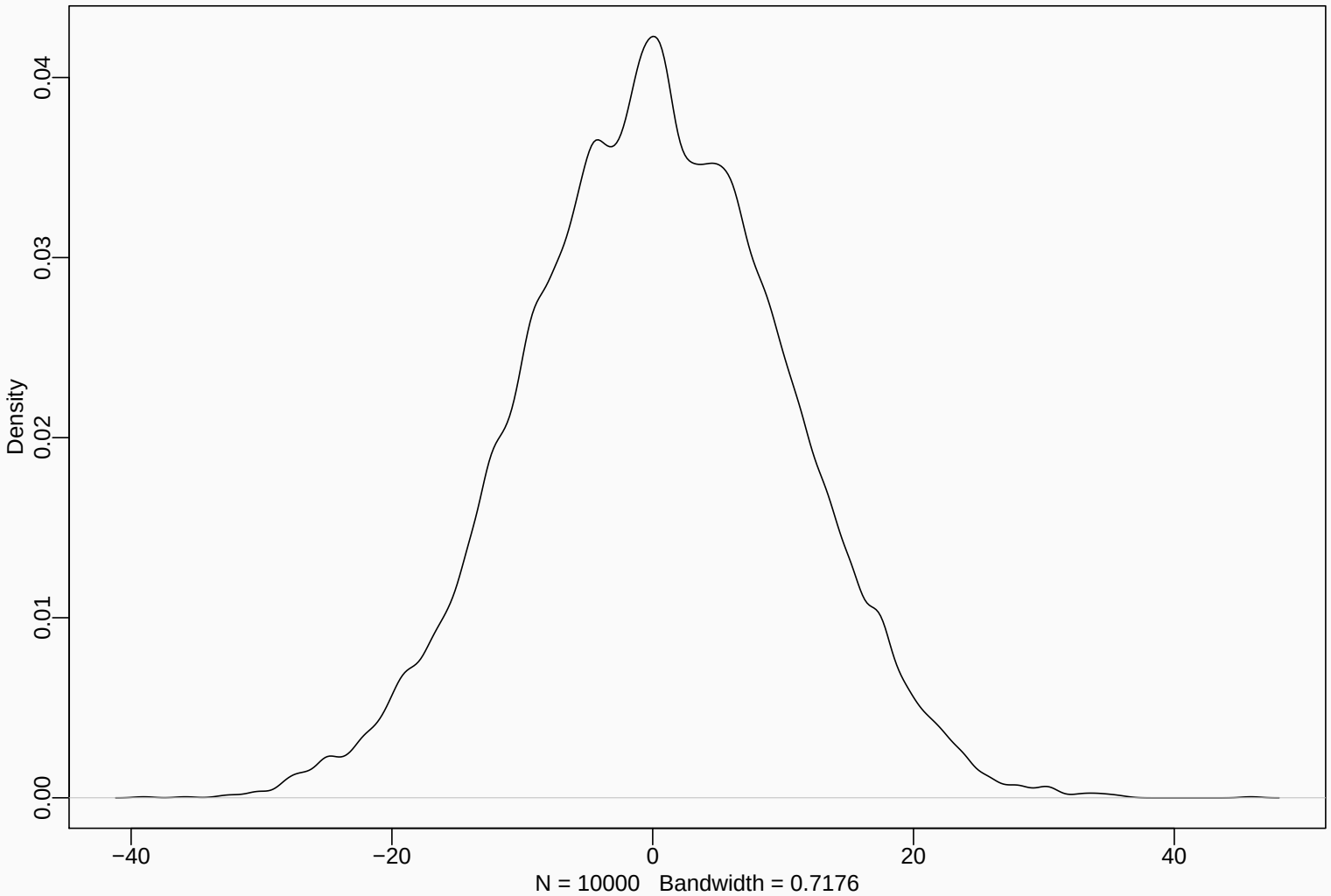
What should our prior for μ be?

How about $Normal(0, 10)$?

$$\mu \sim N(0, 10)$$

What does our prior imply?

```
prior_sims<-rnorm(1e4, 0, 10)
dens(prior_sims)
```



Rethink the prior for μ

The average adult male is 176.5 cm, the average adult female is 163 cm.

Maybe we should set the mean as $\frac{176.5+163}{2} = 169.75$

Rethink the prior for μ

The average adult male is 176.5 cm, the average adult female is 163 cm. Maybe we should set the mean as $\frac{176.5+163}{2} = 169.75$

How certain are we that the true mean for this sample will be equal to 169.75?

Rethink the prior for μ

The average adult male is 176.5 cm, the average adult female is 163 cm. Maybe we should set the mean as $\frac{176.5+163}{2} = 169.75$

How certain are we that the true mean for this sample will be equal to 169.75?

- Completely certain? $\mu \sim N(169.75, 0.01)$

Rethink the prior for μ

The average adult male is 176.5 cm, the average adult female is 163 cm. Maybe we should set the mean as $\frac{176.5+163}{2} = 169.75$

How certain are we that the true mean for this sample will be equal to 169.75?

- Completely certain? $\mu \sim N(169.75, 0.01)$
- Pretty darn sure? $\mu \sim N(169.75, 2)$

Rethink the prior for μ

The average adult male is 176.5 cm, the average adult female is 163 cm. Maybe we should set the mean as $\frac{176.5+163}{2} = 169.75$

How certain are we that the true mean for this sample will be equal to 169.75?

- Completely certain? $\mu \sim N(169.75, 0.01)$
- Pretty darn sure? $\mu \sim N(169.75, 2)$
- Probably in the ballpark $\mu \sim N(169.75, 10)$
- OK, not too sure, but not a bad guess $\mu \sim N(169.75, 25)$

Simulate the priors for μ

```
prior_1<-rnorm(1e4, 169.75, 1)
prior_2<-rnorm(1e4, 169.75, 2)
prior_3<-rnorm(1e4, 169.75, 10)
prior_4<-rnorm(1e4, 168.75, 25)
```

Check out the simulations for μ

```
quantile(prior_1, c(0.05, 0.5, 0.9))
```

```
##       5%      50%      90%
## 168.1063 169.7659 171.0110
```

```
quantile(prior_2, c(0.05, 0.5, 0.9))
```

```
##       5%      50%      90%
## 166.5096 169.7547 172.3133
```

```
quantile(prior_3, c(0.05, 0.5, 0.9))
```

```
##       5%      50%      90%
## 153.5469 169.7145 182.6973
```

```
quantile(prior_4, c(0.05, 0.5, 0.9))
```

```
##       5%      50%      90%
## 128.0699 168.8352 201.1157
```

Now for σ

How much do individuals vary on average from the population average? How variable is height?

Now for σ

How much do individuals vary on average from the population average? How variable is height?

Recall what a standard deviation is:

$$sd = \sqrt{\frac{\sum (x_i - \mu)^2}{n}}$$

The average deviation of an observation from the mean

Now for σ

How much do individuals vary on average from the population average? How variable is height?

Recall what a standard deviation is:

$$sd = \sqrt{\frac{\sum (x_i - \mu)^2}{n}}$$

The average deviation of an observation from the mean

Note: σ is constrained to be positive by definition

The impact of σ on our prior

Our prior for σ captures our beliefs about how much variance there is within the population.

The impact of σ on our prior

Our prior for σ captures our beliefs about how much variance there is within the population.

Always be sure to think through the scale of the outcome we are modeling (e.g. height)

- No idea how variable the population is: $\sigma \sim \text{Uniform}(0, 50)$
- We know there's not much variation: $\sigma \sim \text{Uniform}(0, 10)$

Understanding the prior: prior predictive simulation

1. Simulate μ
2. Simulate σ
3. Draw predictions from $N(\mu, \sigma)$

Understanding the prior: prior predictive simulation

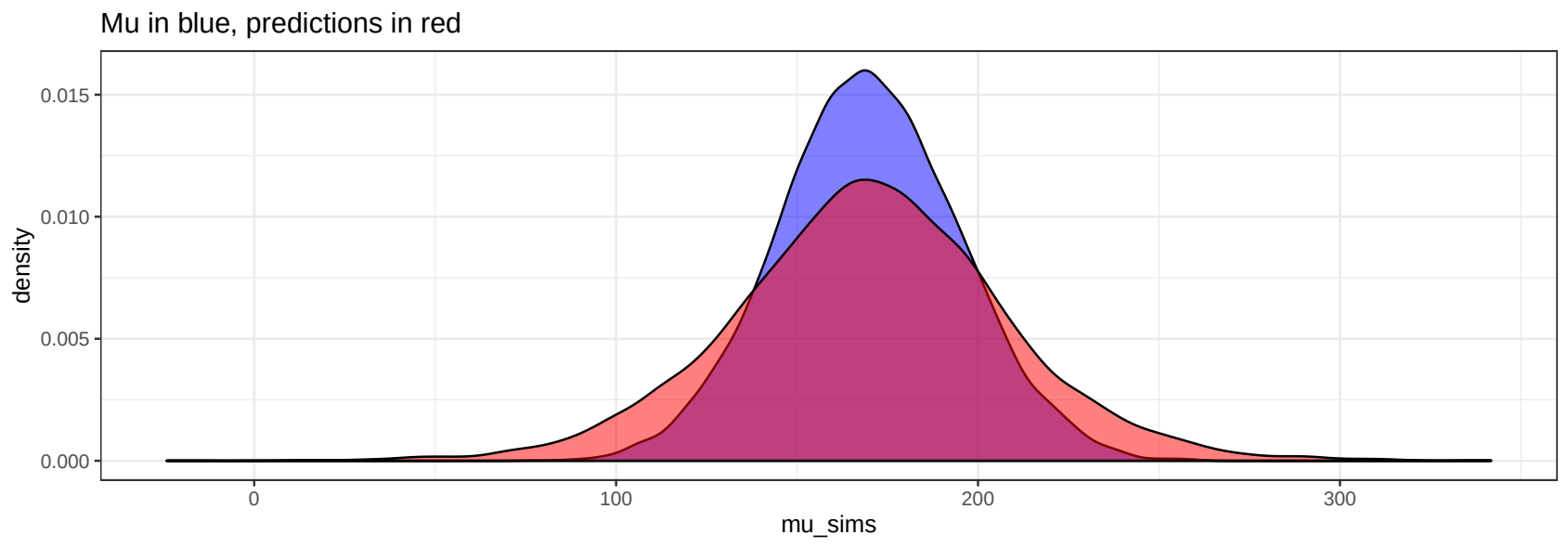
1. Simulate μ
2. Simulate σ
3. Draw predictions from $N(\mu, \sigma)$

```
### simulate mu prior: mu ~ N(169.75, 25) prior
mu_sims<-rnorm(1e4, 169.75, 25)
### simulate sigma prior: sigma ~ U(0,50)
sigma_sims<-runif(1e4, 0, 50)
### simulate heights: N(mu, sigma)
height_preds<-rnorm(1e4, mu_sims, sigma_sims)

summary(height_preds)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  -24.17  146.21  170.00  169.91  193.95  341.76
```

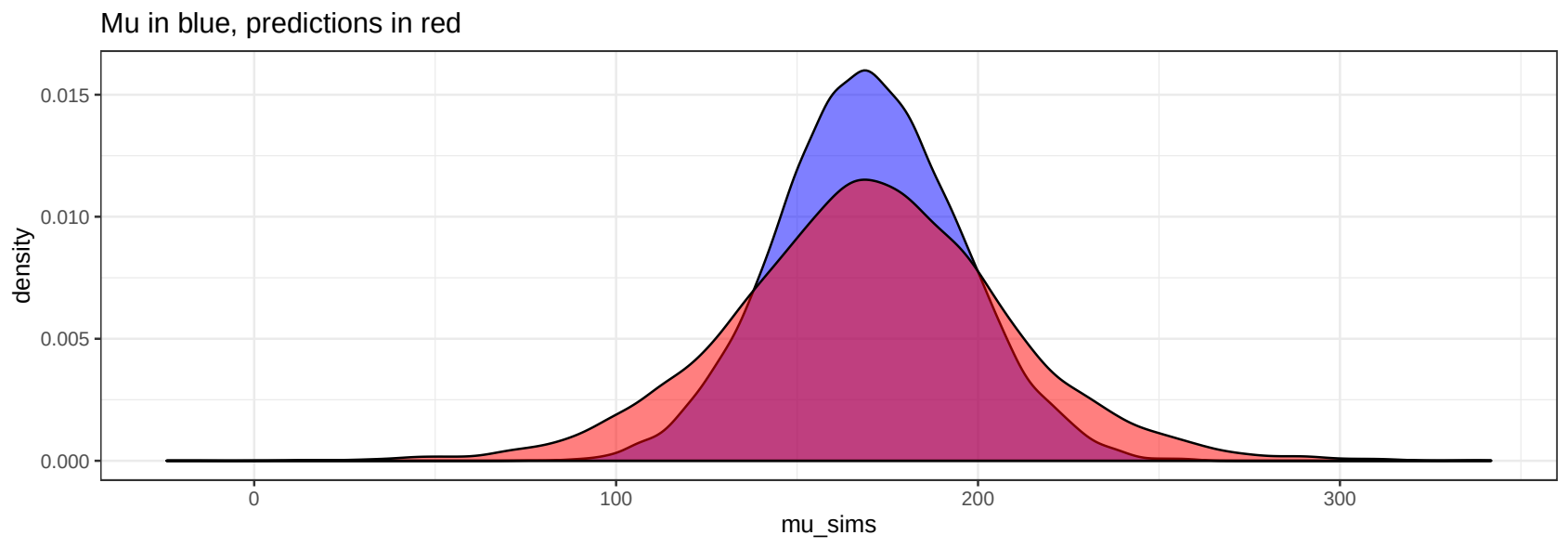
Visualize these predictions

```
plot_dat<-tibble(height_preds, mu_sims)
ggplot(plot_dat, aes(x = mu_sims)) +
  geom_density(fill = "blue", alpha = 0.5) +
  geom_density(aes(x = height_preds), fill = "red", alpha = 0.5) +
  labs(title = "Mu in blue, predictions in red")
```



Visualize these predictions

```
plot_dat<-tibble(height_preds, mu_sims)
ggplot(plot_dat, aes(x = mu_sims)) +
  geom_density(fill = "blue", alpha = 0.5) +
  geom_density(aes(x = height_preds), fill = "red", alpha = 0.5) +
  labs(title = "Mu in blue, predictions in red")
```



Predictions include both parameter uncertainty **and** sampling uncertainty

Model notation

Likelihood: $h_i \sim Normal(\mu, \sigma)$

μ Prior: $\mu \sim Normal(169.75, 25)$

σ Prior: $\sigma \sim Uniform(0, 50)$

Now translate into an R formula

Likelihood: $h_i \sim Normal(\mu, \sigma)$

μ Prior: $\mu \sim Normal(169.75, 25)$

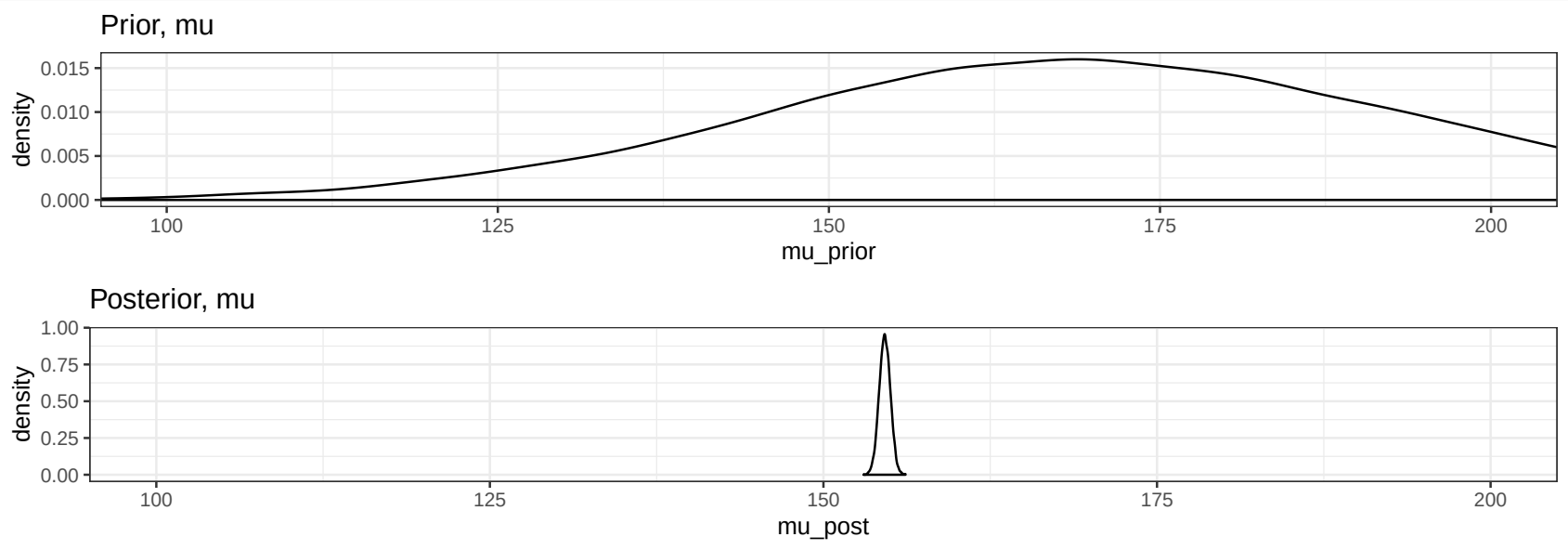
σ Prior: $\sigma \sim Uniform(0, 50)$

```
## remove children with filter()
d2<- d %>%
  filter(age>=18)
## define model
model_formula<-alist(
  height ~ dnorm(mu, sigma),
  mu ~ dnorm(169.75, 25),
  sigma ~ dunif(0,50)
)
## estimate with quap
m0<-quap(model_formula, data = d2)
summary(m0)
```

```
##             mean        sd       5.5%      94.5%
## mu    154.601182 0.4120254 153.942686 155.259678
## sigma   7.731329 0.2913855   7.265638   8.197019
```

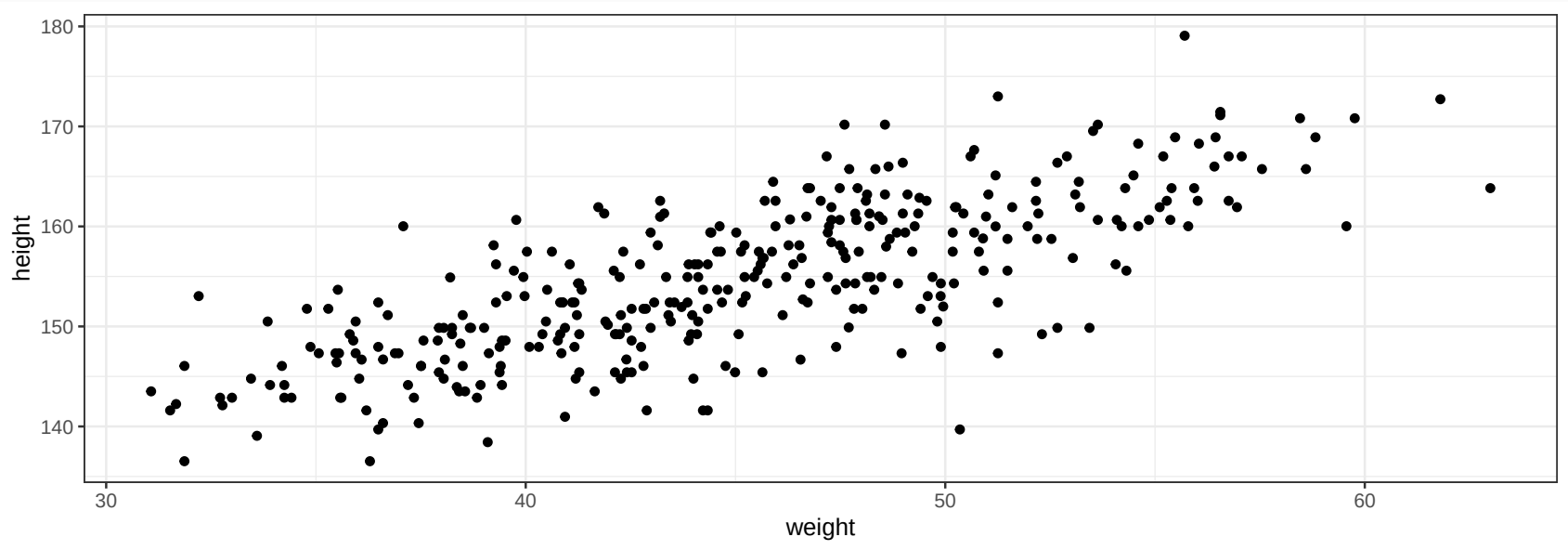
Compare results to the prior

```
library(gridExtra)
post<-extract.samples(m0)
plot_dat<-data.frame(mu_post = post$mu, mu_prior = mu_sims)
p1<-ggplot(plot_dat, aes(x = mu_prior)) +
  geom_density() +
  ggtitle("Prior, mu")+
  coord_cartesian(xlim=c(100, 200))
p2<-ggplot(plot_dat,
           aes(x = mu_post)) +
  geom_density() +
  ggtitle("Posterior, mu") +
  coord_cartesian(xlim=c(100, 200))
grid.arrange(p1, p2)
```



Predicting height with other variables

```
ggplot(d2,
       aes(x = weight, y = height)) +
  geom_point()
```



What does it mean to assume a linear relationship between height and weight?

The anatomy of a linear model

Begin with the Normal model of height

$$h_i \sim Normal(\mu, \sigma)$$
$$\mu \sim Normal(169.75, 25)$$
$$\sigma \sim Uniform(0, 50)$$

Let x_i be the weight of the person in row i

$$h_i \sim Normal(\mu_i, \sigma)$$
$$\mu_i = \alpha + \beta(x_i - \bar{x})$$
$$\alpha \sim Normal(169.75, 25)$$
$$\beta \sim Uniform(0, 10)$$

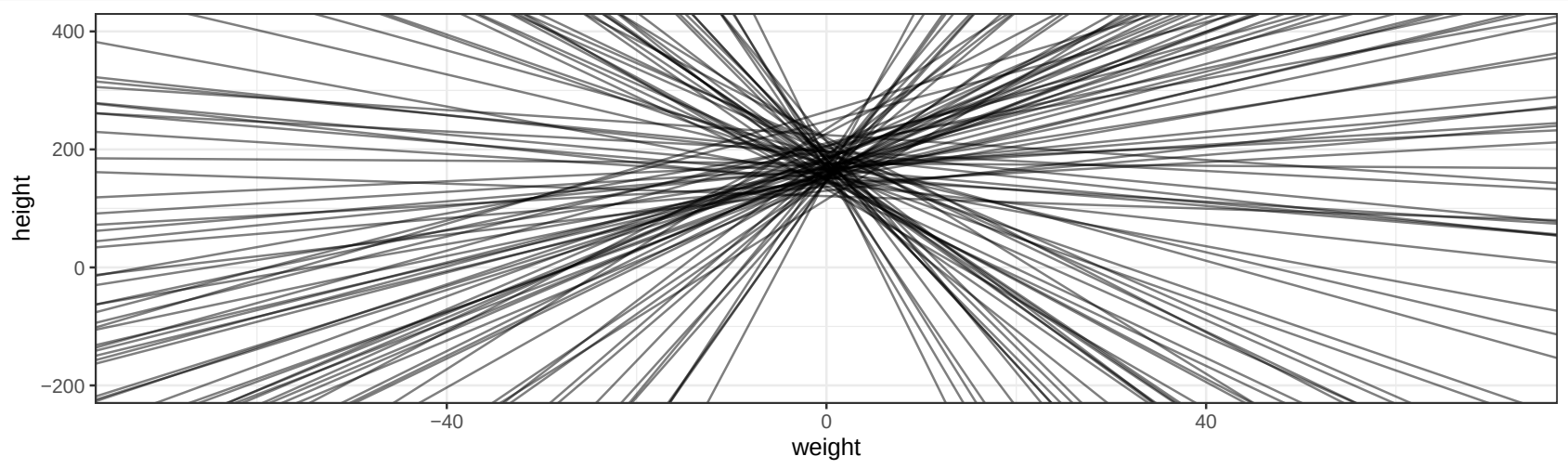
What does our prior imply?

Our prior now describes a line: $mu = \alpha + \beta x$

What does our prior imply?

Our prior now describes a line: $mu = \alpha + \beta x$

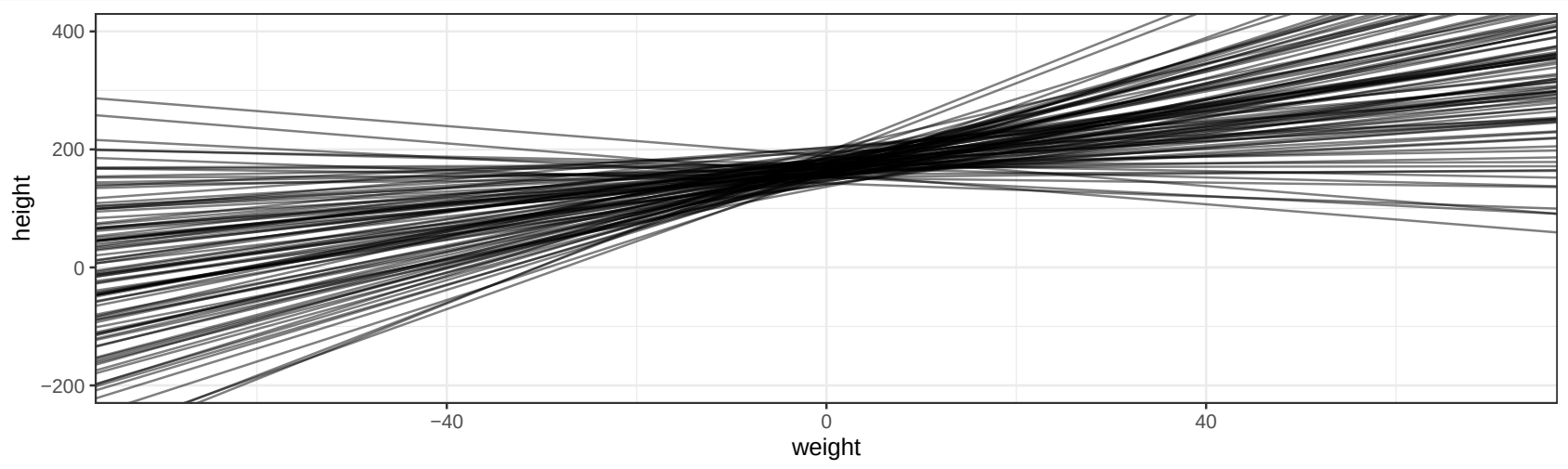
```
## Simulate regression lines from prior and visualize
n<-100
alpha<-rnorm(n, 169.75, 25)
beta<-rnorm(n, 0, 10)
plot_dat<-data.frame(alpha = alpha, beta=beta, sim_n = factor(1:100),
                     weight = seq(-70, 70, length.out=100),
                     height = seq(-200,400, length.out=100))
ggplot(plot_dat,
       aes(x = weight, y = height)) +
  geom_blank() +
  geom_abline(aes(intercept = alpha,
                  slope = beta,
                  group= sim_n),
              alpha = 0.5)
```



Let's try again

Let's shrink the variance on α, and set β to be more likely positive

```
## Simulate regression lines from prior and visualize
n<-100
alpha<-rnorm(n, 169.75, 15)
beta<-rnorm(n, 2, 2)
plot_dat<-data.frame(alpha = alpha, beta=beta, sim_n = factor(1:100),
                     weight = seq(-70, 70, length.out=100),
                     height = seq(-200,400, length.out=100))
ggplot(plot_dat,
       aes(x = weight, y = height)) +
  geom_blank() +
  geom_abline(aes(intercept = alpha,
                  slope = beta,
                  group= sim_n),
              alpha = 0.5)
```



Estimating the model

Likelihood for height: $h_i \sim Normal(\mu_i, \sigma)$

Linear model for mean: $\mu_i = \alpha + \beta x_i$

Prior for intercept: $\alpha \sim Normal(169.75, 15)$

Prior for slope: $\beta \sim Normal(2, 2)$

Prior for standard deviation: $\sigma \sim Uniform(0, 50)$

```
data(Howell1)
d<-Howell1 %>%
  filter(age>=18)
xbar<-mean(d$weight)

m1<-quap(
  alist(
    height ~ dnorm( mu , sigma) ,
    mu <- a + b* weight,
    a ~ dnorm(178 , 20),
    b ~ dlnorm(0 , 1),
    sigma ~ dunif(0 , 50)
  ),
  data = d
)
```

Interpreting the posterior: table

```
summary(m1)
```

```
##              mean         sd        5.5%       94.5%
## a     114.534318 1.89774726 111.5013512 117.5672846
## b       0.890730 0.04175799   0.8239927   0.9574674
## sigma   5.072719 0.19124893   4.7670660   5.3783715
```

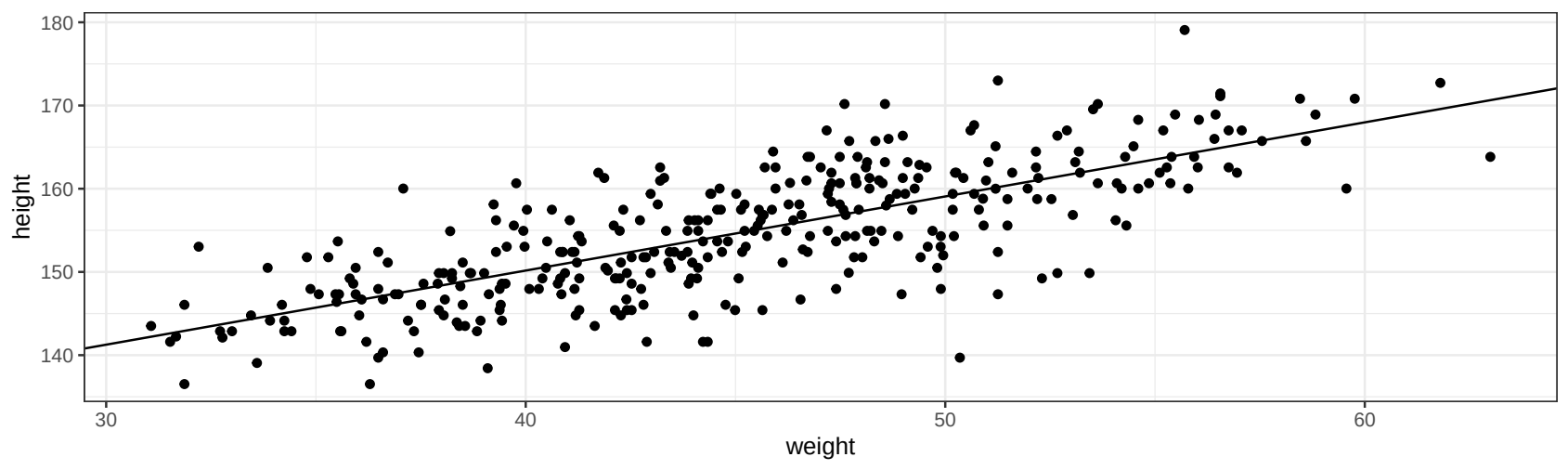
Examining posterior samples

```
post_m1<-extract.samples(m1)
head(post_m1)
```

```
##          a         b    sigma
## 1 114.0870 0.8958745 4.786056
## 2 117.2360 0.8368378 4.925024
## 3 117.3856 0.8397231 5.240202
## 4 119.2280 0.7849737 5.154901
## 5 115.1533 0.8658036 4.928546
## 6 116.7038 0.8448853 4.835857
```

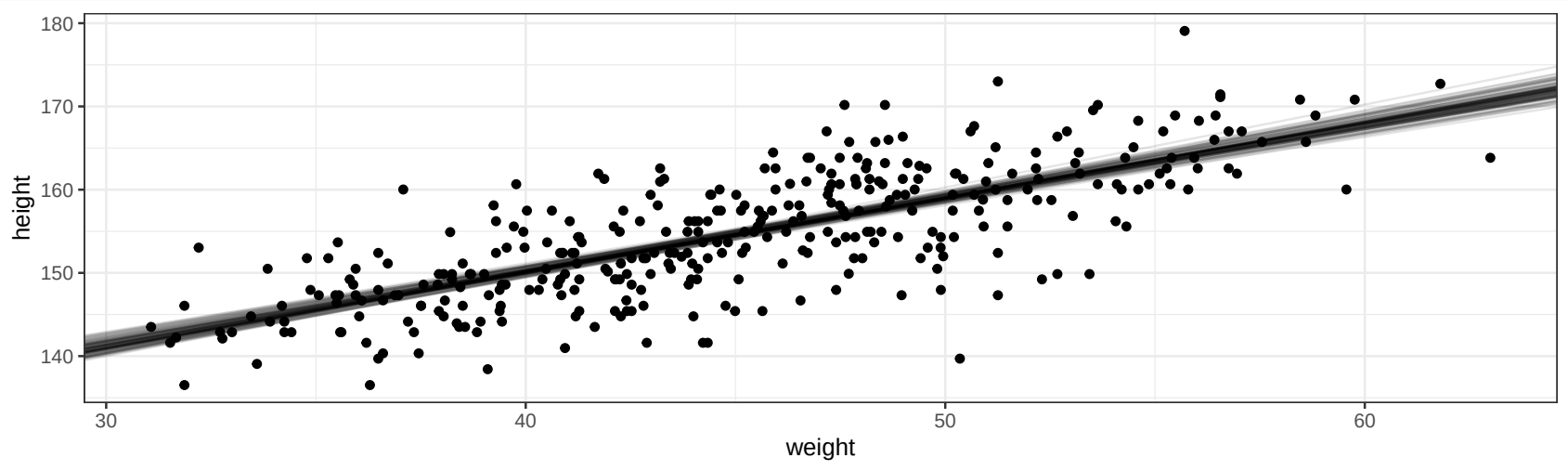
Visualizing the estimated posterior: posterior mean

```
plot_dat<-post_m1 %>%
  summarise(a = mean(a), b = mean(b))
ggplot(d,
       aes(x = weight, y = height)) +
  geom_point() +
  geom_abline(data = plot_dat,
              aes(intercept = a, slope = b))
```



But the posterior is more than one line

```
ggplot(d,
       aes(x = weight, y = height)) +
  geom_point() +
  geom_abline(data = post_m1[1:100,],
              aes(intercept = a, slope = b), alpha = 0.1)
```

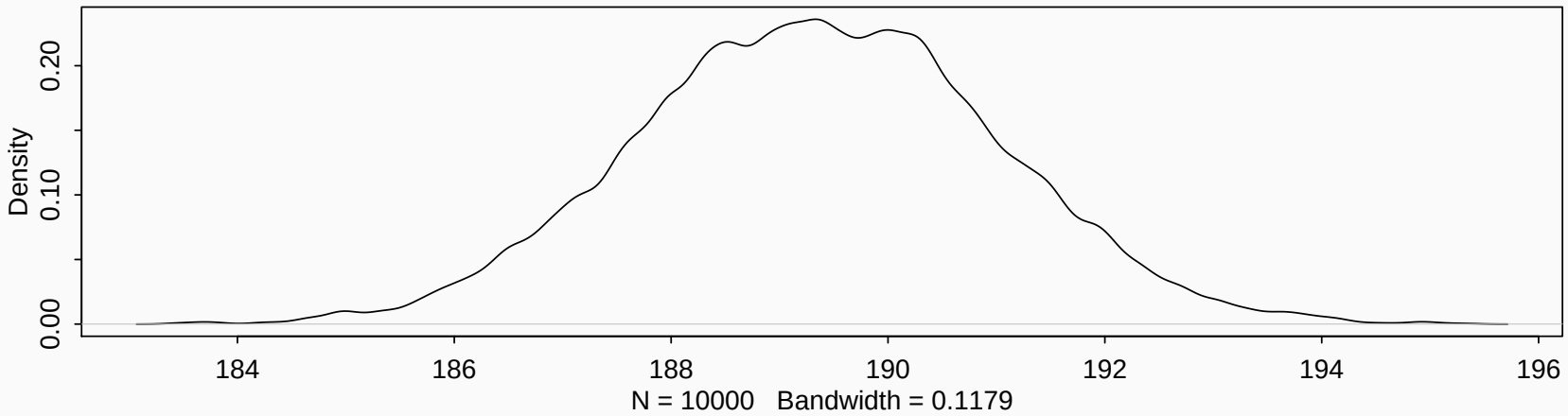


At a single value of x

I weight about 84 kg. Let's solve for h

$$E(h_i) = \mu_i = \alpha + \beta x_i$$

```
mu_FE<-post_m1$a + post_m1$b * 84
dens(mu_FE)
```



These are the values of μ for individuals of weight 84kg that are compatible with the data

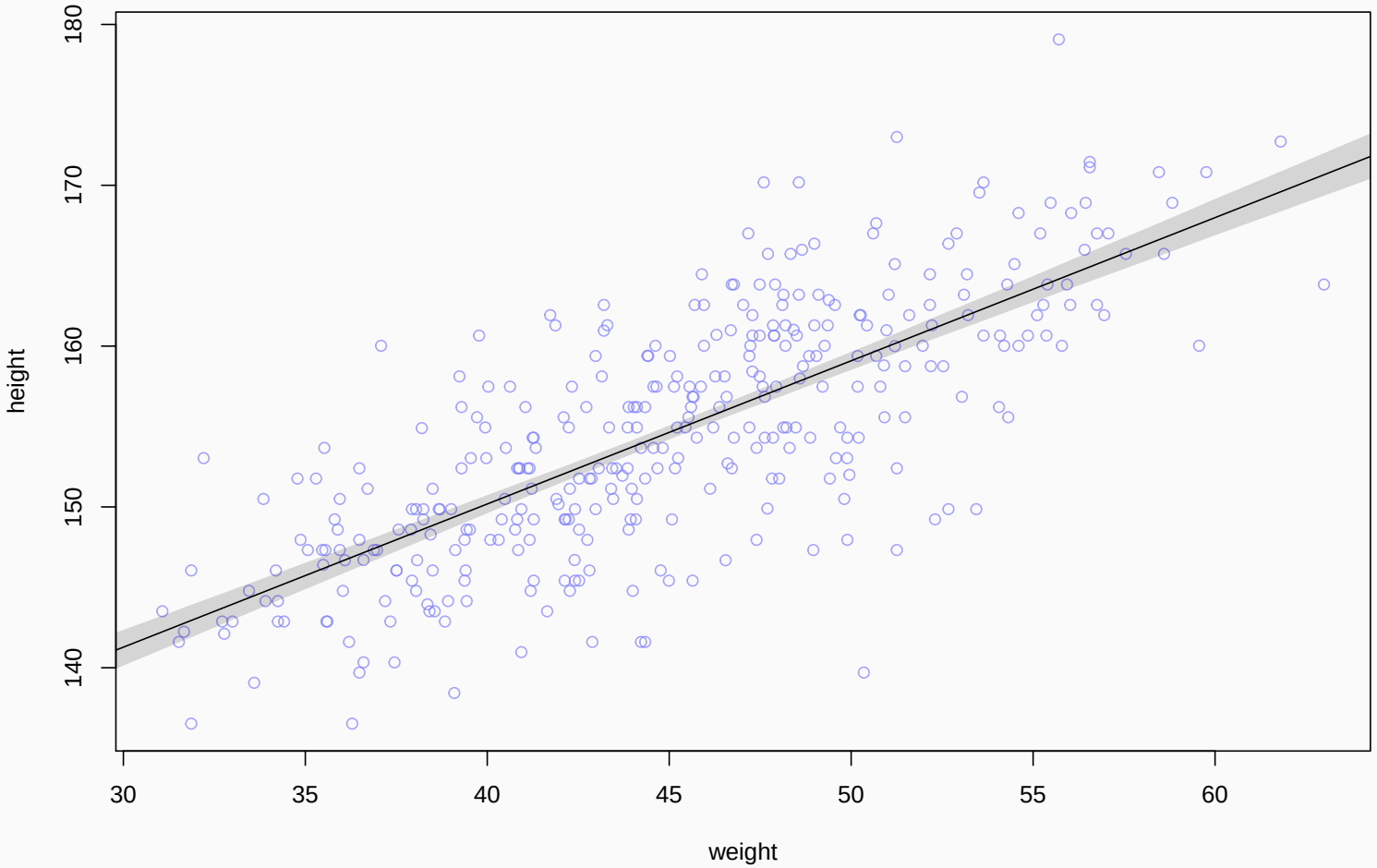
Summarizing the posterior over the full range of the data

```
## create a vector for the sequence of values to predict
weights<-seq(25, 70, by=1)
sim_dat<-data.frame(weight = weights)
## link calculates mu for each weight and each posterior sample
mu<-link(m1, data = sim_dat)
dim(mu)
```

```
## [1] 1000   46
```

```
## compute the mean, and 89% PI
mu.mean<-apply(mu, 2, mean)
mu.PI<-apply(mu, 2, PI, prob = 0.89)
```

Visualizing the posterior for mu: 89 percent PI



Predicting new data; validating the model

Recall the method for posterior prediction:

1. Draw samples of the parameters from the posterior
2. Use these samples to draw new predictions using the likelihood

Predicting new data; validating the model

Recall the method for posterior prediction:

1. Draw samples of the parameters from the posterior
2. Use these samples to draw new predictions using the likelihood

```
## already got the samples in post_m1
head(post_m1)
```

```
##          a         b    sigma
## 1 114.0870 0.8958745 4.786056
## 2 117.2360 0.8368378 4.925024
## 3 117.3856 0.8397231 5.240202
## 4 119.2280 0.7849737 5.154901
## 5 115.1533 0.8658036 4.928546
## 6 116.7038 0.8448853 4.835857
```

We haven't used σ yet. It will tell our model how much sampling uncertainty to produce

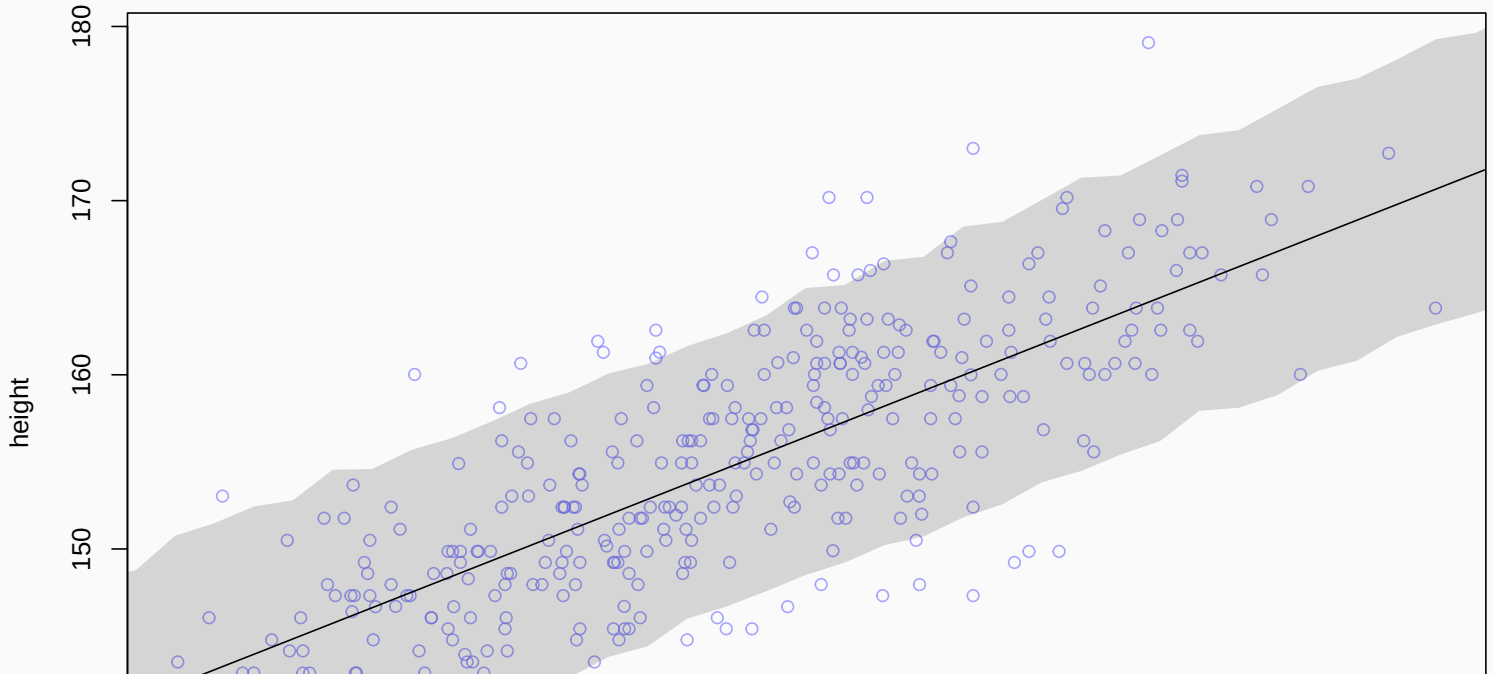
Method for predictions

We could do it by hand, or we could just use sim(). It will give us 1,000 predictions for each of 43 unique height values we provided

```
height_preds<-sim(m1, data = sim_dat)
height.PI<-apply(height_preds, 2, PI, prob = 0.89)
```

Prediction uncertainty

```
# plot raw data
plot( height ~ weight , d2 , col=col.alpha(rangi2,0.7) )
# draw MAP line
lines( weights , mu.mean )
# draw PI region for simulated heights
shade( height.PI , weights )
```



Next time

- Polynomials and splines
- Regression with multiple predictors

Next time

- Polynomials and splines
- Regression with multiple predictors
- I'll post the new homework on Friday

Next time

- Polynomials and splines
- Regression with multiple predictors
- I'll post the new homework on Friday
- A note on Q5:

```
boys_after_girls<-birth2[birth1==0]
```