STATISTICAL RETHINKING 2022
WEEK 2 SOLUTIONS

1. The heights that interest us are all adult heights, so we can analyze only the adults and make an okay linear approximation. If you did something else, that's okay. What matters is being able to use the model, whatever model you used, to make new predictions for new individuals. Loading the data, selecting out adults, and doing the regression from the lecture:

```
library(rethinking)
data(Howell1)
d <- Howell1
d2 <- d[ d$age >= 18 , ]
Hbar <- mean(d2$height)
dat <- list(W=d2$weight,H=d2$height,Hbar=Hbar)

m1 <- quap(
    alist(
        W ~ dnorm( mu , sigma ) ,
        mu <- a + b*( H - Hbar ) ,
        a ~ dnorm( 60 , 10 ) ,
        b ~ dlnorm( 0 , 1 ) ,
        sigma ~ dunif( 0 , 10 )
    ) , data=dat )
```

Now we need posterior predictions for each case in the table. Easiest way to do this is to use `sim`. We need `sim`, not just `link`, because we are trying to predict an individual's height. So the relevant compatibility interval includes the Gaussian variance from `sigma`. If you provided only the compatibility interval for μ, that's okay. But be sure you understand the difference.

```
dat2 <- list( H=c(140,160,175) , Hbar=Hbar )
h_sim <- sim( m1 , data=dat2 )
Ew <- apply(h_sim,2,mean)
h_ci <- apply(h_sim,2,PI,prob=0.89)
```

Now all in table form:

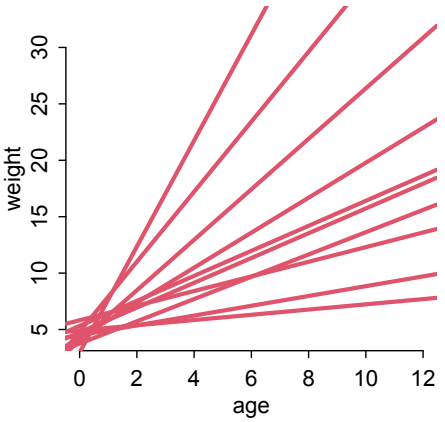
```
datr <- cbind( H=c(140,160,175) , EW , L89=h_ci[1,] , U89=h_ci[2,] )
round(datr,1)
```

```
       H   EW  L89  U89
[1,] 140 35.7 29.1 42.8
[2,] 160 48.4 42.0 55.1
[3,] 175 57.9 51.1 64.7
```

The columns are in order: Height, expected weight, lower bound of 89% interval, upper bound of 89% interval.

2. Since we want the total effect of age, we just need a linear regression of weight on age. Let's set up the data and then simulate some priors.

```
library(rethinking)
data(Howell1)
d <- Howell1
d <- d[ d$age < 13 , ]

# sim from priors
n <- 10
a <- rnorm(n,5,1)
b <- rlnorm(n,0,1)
plot( NULL , xlim=range(d$age) , ylim=range(d$weight) ,
    xlab="age" , ylab="weight" )
for ( i in 1:n ) abline( a[i] , b[i] , lwd=3 , col=2 )
```



These were my first guess, given that the relationship must be positive and that weight at age zero is birth weight, an average birth weight is around 5 kilograms (but varies a lot).

Here's the model.

```
m2 <- quap(
    alist(
        W ~ dnorm( mu , sigma ),
        mu <- a + b*A,
        a ~ dnorm(5,1),
        b ~ dlnorm(0,1),
        sigma ~ dexp(1)
    ), data=list(W=d$weight,A=d$age) )
precis(m2)
```

```
      mean   sd 5.5% 94.5%
a     7.18 0.34 6.64  7.73
b     1.37 0.05 1.29  1.46
sigma 2.51 0.15 2.27  2.74
```

The causal effect of each year of growth is given (in this case) by the parameter `b`. So its 89% interval is 1.29 to 1.46 kilograms per year.

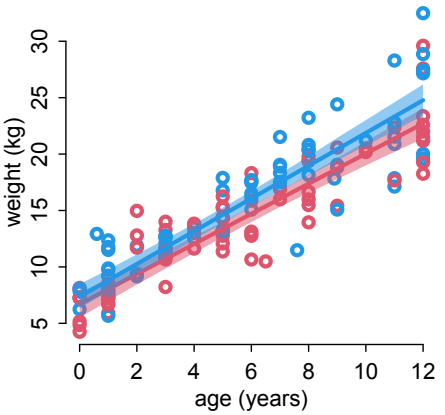
3. We can modify the model above to stratify by sex. We just need to make an index variable (S), just like in the example from the lecture.

```
library(rethinking)
data(Howell1)
d <- Howell1
d <- d[ d$age < 13 , ]

dat <- list(W=d$weight,A=d$age,S=d$male+1)

m3 <- quap(
    alist(
        W ~ dnorm( mu , sigma ),
        mu <- a[S] + b[S]*A,
        a[S] ~ dnorm(5,1),
        b[S] ~ dlnorm(0,1),
        sigma ~ dexp(1)
    ), data=dat )
```

Let's plot the data and overlay the regression lines:

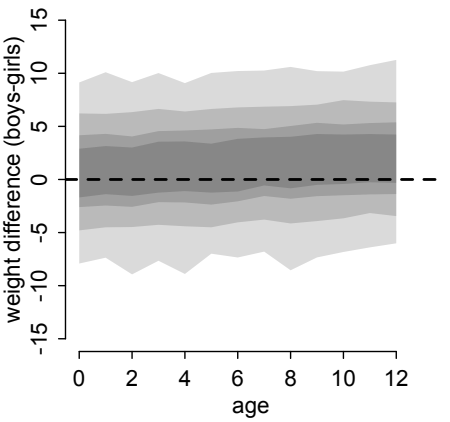
```
plot( d$age , d$weight , lwd=3, col=ifelse(d$male==1,4,2) ,
    xlab="age (years)" , ylab="weight (kg)" )
Aseq <- 0:12
```

```
# girls
muF <- link(m3,data=list(A=Aseq,S=rep(1,13)))
shade( apply(muF,2,PI,0.99) , Aseq , col=col.alpha(2,0.5) )
lines( Aseq , apply(muF,2,mean) , lwd=3 , col=2 )

# boys
muM <- link(m3,data=list(A=Aseq,S=rep(2,13)))
shade( apply(muM,2,PI,0.99) , Aseq , col=col.alpha(4,0.5) )
lines( Aseq , apply(muM,2,mean) , lwd=3 , col=4 )
```



So boys look a little heavier than girls at all ages and seem to increase slightly faster as well. Let's do a posterior contrast across ages though, so we can get make sure.

```
# contrast at each age
Aseq <- 0:12
mu1 <- sim(m3,data=list(A=Aseq,S=rep(1,13)))
mu2 <- sim(m3,data=list(A=Aseq,S=rep(2,13)))
mu_contrast <- mu1
for ( i in 1:13 ) mu_contrast[,i] <- mu2[,i] - mu1[,i]
plot( NULL , xlim=c(0,13) , ylim=c(-15,15) , xlab="age" ,
    ylab="weight difference (boys-girls)" )

for ( p in c(0.5,0.67,0.89,0.99) )
shade( apply(mu_contrast,2,PI,prob=p) , Aseq )

abline(h=0,lty=2,lwd=2)
```



These contrasts use the entire weight distribution, not just the expectations. Boys do tend to be heavier than girls at all ages, but the distributions overlap a lot. The difference increases with age.

This is good moment to repeat my sermon on zero. The fact that these contrasts all overlap zero is no reason to assert that there is no difference in weight between boys and girls. That would be silly. But that is exactly what researchers do every time they look if an interval overlaps zero and then act as if the estimate was exactly zero.

4 - OPTIONAL CHALLENGE. There are two tasks here. The first is to convert the data from height measurement to increments in height. The second is to model the increments so they are always positive.

To convert the data to increments, we just subtract each height from the previous height for the same child. This means that the first occasion of measurement has no increment. So we start with the second occasion. There are lots of ways to do this in code. Here is how I did it.

```
data(Oxboys)
d <- Oxboys

d$delta <- NA
for ( i in 1:nrow(d) ) {
    if ( d$Occasion[i] > 1 )
        d$delta[i] <- d$height[i] - d$height[i-1]
}
d <- d[ !is.na(d$delta) , ]
```

The data frame `d` now has a new column `delta` with the increments. And I deleted the first occasion for each boy, since we won't model it.

Now we need a statistical model. There are a few ways to constrain the distribution of the increments to be positive. The easy way is to think of them as log-normal measurements. So you could log them first and them do an ordinary linear regression with them. You just need to exponentiate them later to get them back on the right scale. Or you can use a log-normal regression. I'll do that. Here's the model code.

```
m4 <- quap(
    alist(
        delta ~ dlnorm( alpha , sigma ),
        alpha ~ dnorm( 0 , 0.1 ),
        sigma ~ dexp( 3 )
    ), data=list(delta=d$delta) )
```

I use a log-normal distribution for the delta values. The trick with log-normal distributions is that the parameters refer to the log distribution. So alpha above is the mean of a normal distribution, not a log-normal distribution. Confusing, I know. The mean of the log-normal we are estimating is $\exp(\alpha + \sigma^2/2)$.

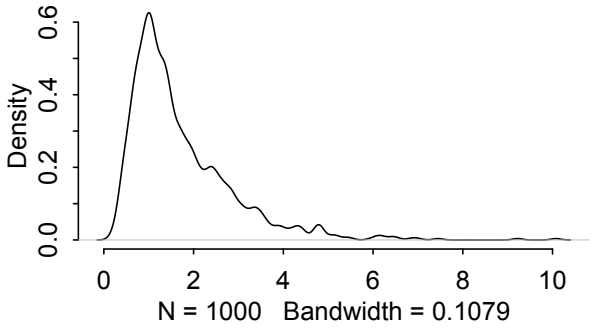
If you play around with prior predictive simulation, you'll see that variance in the priors leads to really explosive means. Here is the code I used:

```
# simulation from priors
n <- 1e3
alpha <- rnorm(n,0,0.1)
sigma <- rexp(n,3)
delta_sim <- rlnorm(n,alpha,sigma)
dens(delta_sim)
```

If you let σ be wider, it will make the prior mean way too high. This is typical of the log-normal. Normal distributions are nice and well-behaved. Log-normal distributions are not. They are tinder boxes.

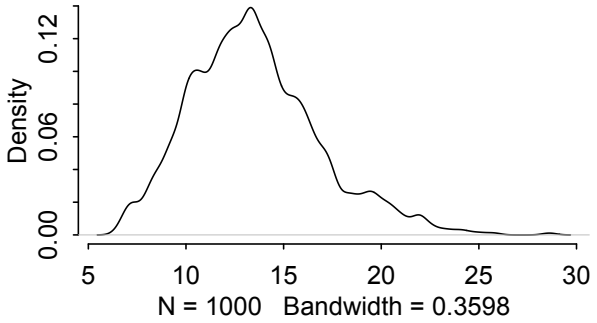
Okay, we have the posterior from `m4`. Now we can plot the posterior distribution of increments and their sum over 8 occasions. First the increment distribution:

```
post <- extract.samples(m4)
dsim <- rlnorm(1e3,post$alpha,post$sigma)
dens(dsim)
```



And the sum over 8 occasions of growth is:

```
inc_sum <- sapply( 1:1000 ,
    function(s) sum(rlnorm(8,post$alpha[s],post$sigma[s])) )
dens(inc_sum)
```



A source of variation that we have ignored is variation among boys in their growth rates. Later in the course, you'll see how to deal with this.