Lecture 8.1: Multiple regression part I

** This lecture is based on chapter 4 of Statistical Rethinking by Richard McElreath.*

As before, we need to load some packages and set some options prior to running any models:

```
library(rstan)
library(shinystan)
library(car)
rstan_options(auto_write = TRUE)
options(mc.cores = parallel::detectCores())
source("../utilityFunctions.R")
```

Now that we have simple linear regression down and we understand what's going on, let's expand our modeling to encompass multiple predictor variables to model an outcome.

There are several good reasons to do this:

1. *Statistical control of confounding variables:* Confounding variables are variables that may be correlated with another variable of interest, sometimes leading to spurious associations of effect.
    - example: Waffle House density is strongly associated with divorce rates, not because frequenting Waffle Houses leads to divorce (well maybe) but because the highest densities of Waffle house are in the South, where the highest divorce rates occur.
    - Confounding variables can also mask effects (e.g., Simpson's paradox)
2. *Multiple causation:* Phenomena or biological processes may be truely driven by multiple causes.
3. *Interactions:* Even when variables are uncorrelated, the importance of each may depend on the other.
    - e.g., Plants need both light and water. In the absence of one, the other has no benefit at all.

BUT, multiple predictors can hurt too, and we will talk about things like *multicollinearity* and demonstrate why the variance inflation factor (back in Biometry) gets that name.

Spurious correlations

For our first example of multiple regression, we will use simulated data of apparent competition between sea urchins and chitons in the presence of sea stars. Chitons are the preferred prey of sea stars (urchins are pokey), but sea stars will eat urchins opportunistically.

- Say you go out to rocky reefs outcrops ($N = 53$) and measure the densities of all three marine invertebrates in multiple quadrats, then average the densities to get a mean invertebrate density at each outcrop.
  - The code to simulate this is below if interested

```
N <- 53
set.seed(1)
seaStars <- rnorm(N) + 10 # Real X variable
chitons <- rnorm(N, seaStars) # Spurious X variable
urchins <- rnorm(N,-seaStars, 1.5)
urchins <- urchins - min(urchins)

reefs <- data.frame(urchins=urchins, seaStars=seaStars, chitons=chitons)
# write.csv(reefs, file="urchinDat.csv", row.names=FALSE)
```

```
reefs <- read.csv("urchinDat.csv")
head(reefs)
```

```
   urchins  seaStars   chitons
1 4.719087  9.373546  8.244183
2 4.199190 10.183643 11.616667
3 4.429478  9.164371 11.144771
4 3.945555 11.595281 11.228059
5 1.734459 10.329508  9.285373
6 3.145573  9.179532  9.749251
```

To begin, let's analyze the predictor variables separately with a univariate regression and see what's going on.

Our model will look very similar to last week, but we are going to standardize the predictor. This is useful for a few reasons:

1. *Interpretation:* A change of one unit is equivalent to a change of one SD. This may be more interesting and revealing than the natural scale.
   - Makes comparison of multiple predictors easier
2. *Computation:* when predictors (or responses) have large values in them or a wide range of values, defining appropriate priors can be challenging.

$$
\begin{aligned}
obs_i &\sim \text{Normal}(\mu_i, \sigma) \\
\mu_i &= \alpha + \beta x_i \\
\alpha &\sim \text{Normal}(0, 10) \\
\beta &\sim \text{Normal}(0, 1) \\
\sigma &\sim \text{Cauchy}^{+}(0, 10)
\end{aligned} \tag{1}
$$

Because we z-transformed, we can set much narrower priors.

Our model, `uniMod.stan`, is formulated as follows:

```
data {
  int<lower=0> nObs;
  vector[nObs] obs;
  vector[nObs] xvar;      // x variable
  real<lower=0> aSD;      // SD of prior alpha
  real<lower=0> bSD;      // SD of prior beta
  real<lower=0> sigmaSD;  // scale for sigma
}

parameters {
  real alpha;
  real beta;
  real<lower=0> sigma;
}

transformed parameters {
  // can be useful for plotting purposes
  vector[nObs] mu;
  mu = alpha + beta*xvar;
}

model {
  alpha ~ normal(0, aSD);
  beta ~  normal(0, bSD);
  sigma ~ cauchy(0, sigmaSD);

  obs ~ normal(mu, sigma);
}
```

First we will run the chiton model,

```
nObs <- nrow(reefs)
urchins <- reefs$urchins
chitons <- reefs$chitons
seaStars <- reefs$seaStars

### Chiton model
o <- order(chitons)
chitonDat <- list(nObs=nObs, obs=urchins[o], xvar=as.vector(scale(chitons[o])),
  aSD=10, bSD=1, sigmaSD=10)

chitMod <- stan(file="uniMod.stan", data=chitonDat, iter=2000,
 chains=4, seed=3)
```

```
# extract posterior estimates of alpha, beta, and mu
chitPar <- as.matrix(chitMod, pars=c("alpha", "beta", "mu"))
```

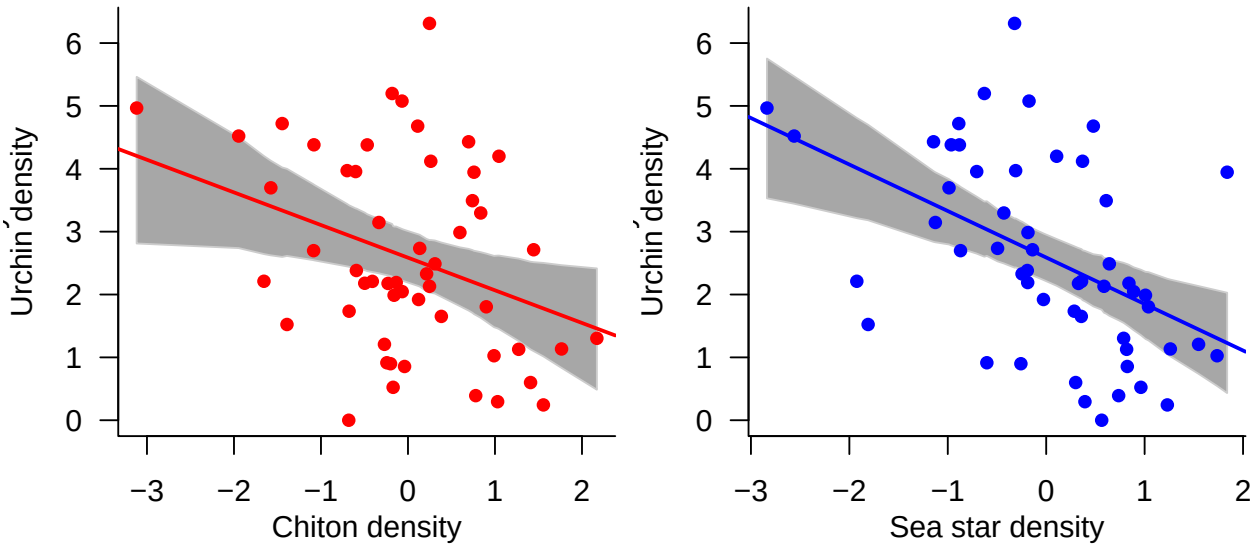
Then the sea star model,

```
# Sea star model
o <- order(seaStars)
starDat <- list(nObs=nObs, obs=urchins[o], xvar=as.vector(scale(seaStars[o])),
  aSD=10, bSD=1, sigmaSD=10)

starMod <- stan(file="uniMod.stan", data=starDat, iter=2000,
 chains=4, seed=3)

# extract posterior estimates of alpha, beta, and mu
starPar <- as.matrix(starMod, pars=c("alpha", "beta", "mu"))
```

The first thing we might want to do is print a summary of our results and look graphically.

```
par(mar=c(3,3.2,0.1,0.5))
par(mfrow=c(1,2))
# Mean & HDI for chitons
chitHDI <- apply(chitPar,2, HDI, credMass=0.95)
chitMean <- colMeans(chitPar)

# Make an empty plot
x <- chitonDat$xvar
y <- chitonDat$obs
plot(x, y, type="n", las=1, bty="l")

mtext(text = "Urchin density", side=2, line = 2.2, cex=1)
mtext(text = "Chiton density", side=1, line = 2, cex=1)

# plot uncertainty interval in mu as a polygon
polygon(x=c(x, rev(x)), y=c(chitHDI[1, -c(1:2)],
  rev(chitHDI[2, -c(1:2)])), col="#50505080", border="grey80")

# plot the data points and mean regression line
points(x, y, pch=16, col="red")
abline(a=chitMean[1], b=chitMean[2], col="red", lwd=2)

### Plot seastar resutls
starHDI <- apply(starPar,2, HDI, credMass=0.95)
starMean <- colMeans(starPar)
# Make an empty plot
x <- starDat$xvar
y <- starDat$obs
```

```
plot(x, y, type="n", las=1, bty="l")

mtext(text = "Urchin density", side=2, line = 2.2, cex=1)
mtext(text = "Sea star density", side=1, line = 2, cex=1)

# plot uncertainty interval in mu as a polygon
polygon(x=c(x, rev(x)), y=c(starHDI[1, -c(1:2)],
  rev(starHDI[2, -c(1:2)])), col="#50505080", border="grey80")

# plot the data points and mean regression line
points(x, y, pch=16, col="blue")
abline(a=starMean[1], b=starMean[2], col="blue", lwd=2)
```




```
round(summary(chitMod, pars=c("alpha", "beta"))$summary,2)
```

```
       mean se_mean   sd  2.5%   25%   50%   75% 97.5% n_eff Rhat
alpha  2.59       0 0.21  2.18  2.45  2.58  2.72  2.99  4000    1
beta  -0.52       0 0.21 -0.93 -0.66 -0.52 -0.38 -0.10  4000    1
```

```
round(summary(starMod, pars=c("alpha", "beta"))$summary,2)
```

```
       mean se_mean   sd  2.5%   25%   50%   75% 97.5% n_eff Rhat
alpha  2.59       0 0.19  2.22  2.46  2.59  2.72  2.96  4000    1
beta  -0.74       0 0.19 -1.11 -0.87 -0.74 -0.61 -0.36  4000    1
```

From the univariate model results, it looks like both chitons and sea stars have similar negative effects on urchin densities.

- A 1 SD change in chiton or sea star densites decreases urchin densities by ≈ 0.5 and ≈ 0.75 urchins respectively.

Multiple regression

Now though we will include both predictors in one model by adding more parameters and predictors to the definition of μ_i.

$$
\begin{aligned}
obs_i &\sim \text{Normal}(\mu_i, \sigma) \\
\mu_i &= \alpha + \sum_{j=1}^{n} \beta_j x_{ji} \\
&= \alpha + \beta_1 x_{1i} + \beta_2 x_{2i} + \ldots + \beta_n x_{ni} \\
\alpha &\sim \text{Normal}(0, 10) \\
\beta_j &\sim \text{Normal}(0, 1) \\
\sigma &\sim \text{Cauchy}^{+}(0, 10)
\end{aligned} \tag{2}
$$

We can set up the model (`multiMod.stan`) as follows. I am going to make use of the `transformed data` block to put our x variables into a matrix.

I am also going to simulate new data in the `generated quantities` block that I will describe in more detail below.

```
data {
  int<lower=0> nObs;
  int<lower=0> nVar;      // no. vars
  vector[nObs] obs;
  vector[nObs] x1;
  vector[nObs] x2;
  real<lower=0> aSD;      // SD of prior alpha
  real<lower=0> bSD;      // SD of prior beta
  real<lower=0> sigmaSD;  // scale for sigma
}

transformed data {
  matrix[nObs, nVar] X;

  X = append_col(x1, x2);
}

parameters {
  real alpha;
  vector[nVar] beta;
  real<lower=0> sigma;
}

transformed parameters {
```

```
  // can be useful for plotting purposes
  vector[nObs] mu;
  mu = alpha + X*beta;
}

model {
  alpha ~ normal(0, aSD);
  beta ~  normal(0, bSD);
  sigma ~ cauchy(0, sigmaSD);

  obs ~ normal(mu, sigma);
}

generated quantities {
  // Generate new counterfactual data by holding other
  // variable at mean value
  vector[nObs] muCH;
  vector[nObs] muSS;

    muCH = alpha + beta[1]*X[,1];
    muSS = alpha + beta[2]*X[,2];
}
```

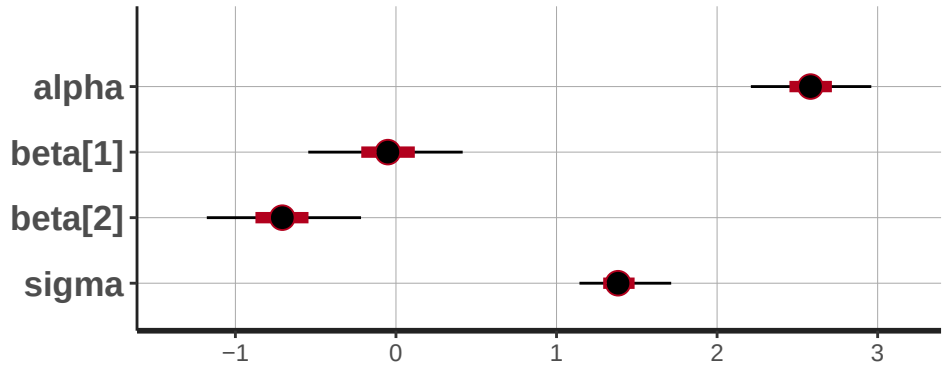
```
dat <- list(nObs=nObs, nVar=2, obs=urchins, x1=as.vector(scale(chitons)),
  x2 = as.vector(scale(seaStars)), aSD=10, bSD=1, sigmaSD=10)

multMod <- stan(file="multiMod.stan", data=dat, iter=2000,
 chains=4, seed=867.5309, pars="mu", include=FALSE)
```

```
round(summary(multMod, pars=c("alpha", "beta"))$summary,2)
```

```
         mean se_mean   sd  2.5%   25%   50%   75% 97.5%   n_eff Rhat
alpha    2.58       0 0.19  2.21  2.45  2.58  2.72  2.96 3654.10    1
beta[1] -0.05       0 0.25 -0.55 -0.22 -0.05  0.12  0.42 2910.36    1
beta[2] -0.71       0 0.25 -1.18 -0.88 -0.71 -0.54 -0.22 2984.89    1
```

The posterior mean for chitons (`beta[1]`) is now quite close to zero, with lots of probability on both sides.

The posterior mean for sea stars (`beta[2]`) is essentially unchanged.



These results can be interpreted as *Once we know the density of sea stars, there is little to no additional predictive power for urchins in also knowing chiton densities.*

Counterfactual plots

Visualizing the results of multiple regressions can be tricky. One thing that we can do to help understand the implications of the model is to simulate *counterfactual* data and plot those results.

- Show implied predictions for imaginary experiments in which different predictor values can be changed independently of each other.

In the `multMod` stan model, I added a generated quantities section:

```
generated quantities {
  // Generate new counterfactual data by holding other
  // variable at mean value
  vector[nObs] muCH;
  vector[nObs] muSS;

    muCH = alpha + beta[1]*X[,1];
    muSS = alpha + beta[2]*X[,2];
}
```

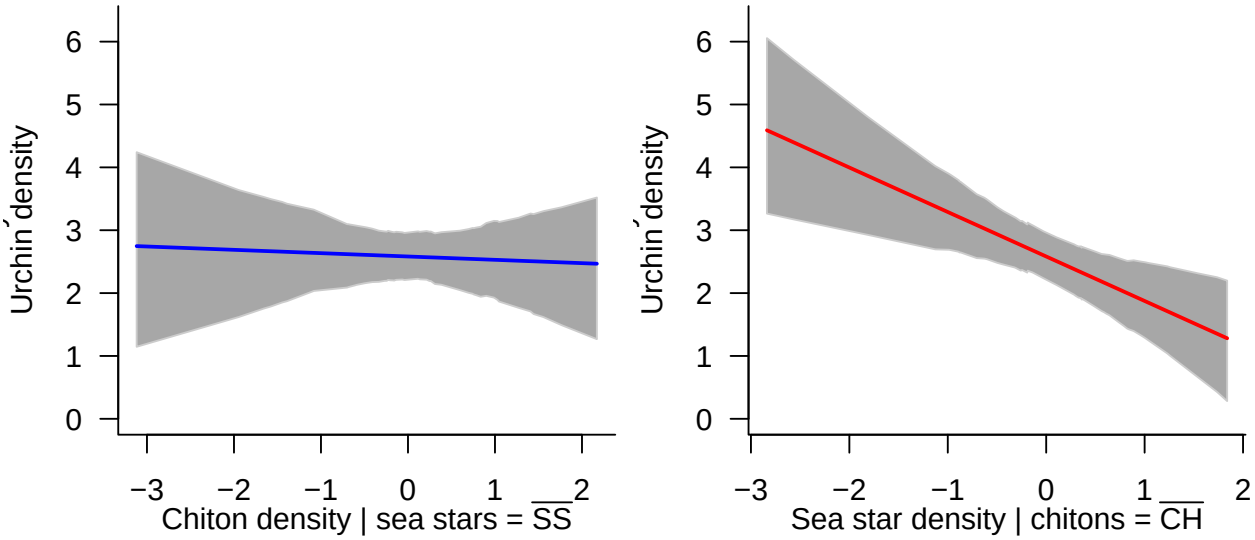
These new vectors simulate new data with one variable (e.g., chitons) while holding the other variable (sea stars) constant—at it's mean, for example.

- Because both predictors are centered and scaled, the intercept is the mean urchin density when both chitons and sea stars are at their mean densities.
  - Not including one of the predictors and coefficients thus accomplishes this.

```
# Extract results for both chitons and sea stars
oCH <- order(chitons)
muCH <- as.matrix(multMod, pars="muCH")
chitHDI <- apply(muCH,2, HDI, credMass=0.95)
chitMean <- colMeans(muCH)[oCH]
```

```r
oSS <- order(seaStars)
muSS <- as.matrix(multMod, pars="muSS")
starHDI <- apply(muSS,2, HDI, credMass=0.95)
starMean <- colMeans(muSS)[oSS]
```

```r
par(mar=c(3,3.2,0.1,0.5))
par(mfrow=c(1,2))

# Make an empty plot
x <- chitonDat$xvar
y <- chitonDat$obs
plot(x, y, type="n", las=1, bty="l")

mtext(text = "Urchin density", side=2, line = 2.2, cex=1)
mtext(text = expression(paste("Chiton density | sea stars = ", bar(SS))),
  side=1, line = 2, cex=1)

# plot uncertainty interval in mu as a polygon
polygon(x=c(x, rev(x)), y=c(chitHDI[1,oCH],
  rev(chitHDI[2,oCH])), col="#50505080", border="grey80")

# plot the data points and mean regression line
lines(x, chitMean, col="blue", lwd=2)


### Plot seastar resutls
x <- starDat$xvar
y <- starDat$obs
plot(x, y, type="n", las=1, bty="l")

mtext(text = "Urchin density", side=2, line = 2.2, cex=1)
mtext(text = expression(paste("Sea star density | chitons = ", bar(CH))),
  side=1, line = 2, cex=1)

# plot uncertainty interval in mu as a polygon
polygon(x=c(x, rev(x)), y=c(starHDI[1,oSS],
  rev(starHDI[2,oSS])), col="#50505080", border="grey80")

# plot the data points and mean regression line
lines(x, starMean, col="red", lwd=2)
```



These plots don't show any data points because the data is imaginary, but they do visually show that after taking both sea star and chiton densities into account, chiton densities don't have any direct impact on urchin densities.

Biologically, this is because urchins and chitons are not competing for food. Instead, as chiton densities increase, more sea stars congregate to eat their preferred prey.

- Because sea stars are stupid and opportunistic, urchins are collateral damage. Thus what looks like an important effect goes away when more variables are added.

Masked relationships

Here is an actual dataset with information about the composition of milk across primate species, as well as some covariates such as brain size and body mass.

- Milk is a big investment and is often more metabolically expensive than gestation.
- One hypothesis is that primates with larger brains produce more energetic milk so the brain can grow quickly. For now we will ignore phylogenetic non-independence among species

```
milk <- read.csv("milk.csv")
head(milk)
```

```
              clade            species kcal.per.g perc.fat perc.protein
1    Strepsirrhine     Eulemur fulvus       0.49    16.60        15.42
2 New World Monkey Alouatta seniculus       0.47    21.22        23.58
3 New World Monkey          A palliata       0.56    29.66        23.46
4 New World Monkey       Cebus apella       0.89    53.41        15.80
5 New World Monkey          S sciureus       0.92    50.58        22.33
6 New World Monkey   Cebuella pygmaea       0.80    41.35        20.85
```

```
  perc.lactose mass neocortex.perc
1        67.98 1.95          55.16
2        55.20 5.25          64.54
3        46.88 5.37          64.54
4        30.79 2.51          67.64
5        27.09 0.68          68.85
6        37.80 0.12          58.85
```

```
mass <- log(milk$mass)
ncp <- milk$neocortex.perc
kcal <- milk$kcal.per.g
nObs <- nrow(milk)
```

The variables we will work with are:

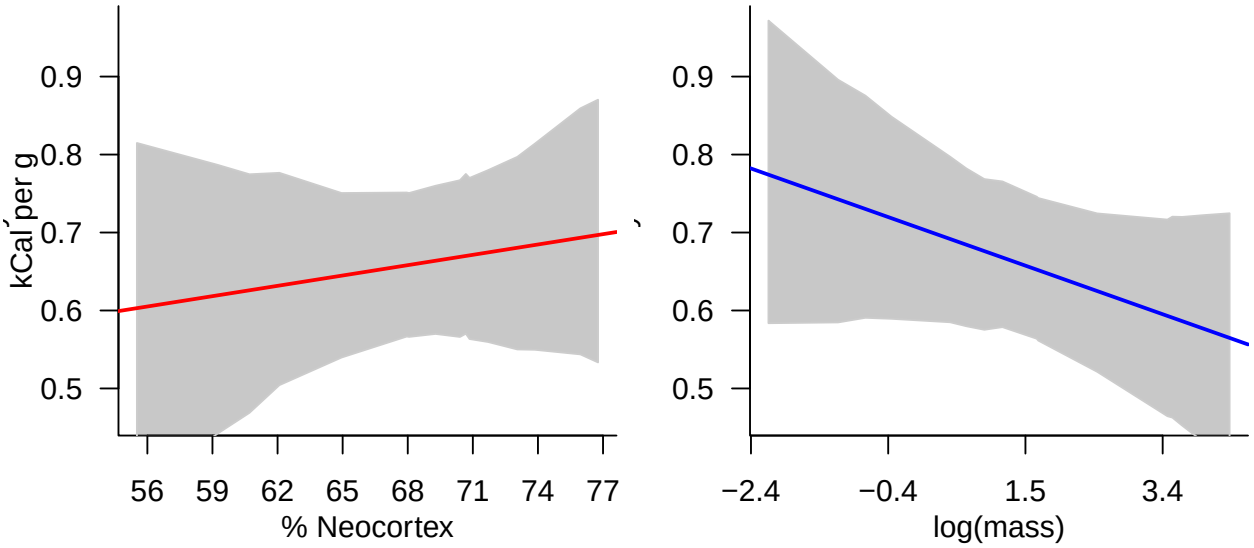
- kcal per gram: kilocalories of energy per gram milk
- mass: as the average female body mass in kg
- neocortex percent: perecnt of total brain mass that is the neocortex. This is well elaborated in primates

```
# neoCortex
o <- order(ncp)
ncpDat <- list(nObs=nObs, obs=kcal[o], xvar=as.vector(scale(ncp[o])),
  aSD=10, bSD=1, sigmaSD=10)

ncpMod <- stan(file="uniMod.stan", data=ncpDat, iter=2000,
 chains=4, seed=867.5309)

# extract posterior estimates of alpha, beta, and mu
ncpPar <- as.matrix(ncpMod, pars=c("alpha", "beta", "mu"))
```

```
# mass
o <- order(mass)
massDat <- list(nObs=nObs, obs=kcal[o], xvar=as.vector(scale(mass[o])),
  aSD=10, bSD=1, sigmaSD=10)

massMod <- stan(file="uniMod.stan", data=massDat, iter=2000,
 chains=4, seed=867.5309)

# extract posterior estimates of alpha, beta, and mu
massPar <- as.matrix(massMod, pars=c("alpha", "beta", "mu"))
```

```
par(mar=c(3,3.2,0.1,0.5))
par(mfrow=c(1,2))
# Mean & HDI for NCP
ncpHDI <- apply(ncpPar,2, HDI, credMass=0.95)
ncpMean <- colMeans(ncpPar)
```

```r
# Make an empty plot
x <- ncpDat$xvar
y <- ncpDat$obs
plot(x, y, type="n", las=1, bty="l", xaxt="n")
at <- seq(-2, 1.5, by=0.5)
axis(1, at=at, labels=round(at*sd(ncp) + mean(ncp)))
mtext(text = "kCal per g", side=2, line = 2.2, cex=1)
mtext(text = "% Neocortex", side=1, line = 2, cex=1)

# plot uncertainty interval in mu as a polygon

polygon(x=c(x, rev(x)), y=c(ncpHDI[1, -c(1:2)],
  rev(ncpHDI[2, -c(1:2)])), col="#50505050", border="grey80")

# plot the data points and mean regression line
abline(a=ncpMean[1], b=ncpMean[2], col="red", lwd=2)


### Plot mass resutls
massHDI <- apply(massPar,2, HDI, credMass=0.95)
massMean <- colMeans(massPar)
# Make an empty plot
x <- massDat$xvar
y <- massDat$obs
plot(x, y, type="n", las=1, bty="l", xaxt="n")
at <- seq(-2, 1.5, by=1)
axis(1, at=at, labels=round(at*sd(mass) + mean(mass),1))
#mtext(text = "kCal per g", side=2, line = 2.2, cex=1)
mtext(text = "log(mass)", side=1, line = 2, cex=1)

# plot uncertainty interval in mu as a polygon
polygon(x=c(x, rev(x)), y=c(massHDI[1, -c(1:2)],
  rev(massHDI[2, -c(1:2)])), col="#50505050", border="grey80")

# plot the data points and mean regression line
abline(a=massMean[1], b=massMean[2], col="blue", lwd=2)
```



If we look at the results separately, there doesn't seem to be much of an effect of neocortex, although body mass does have some impact (it is negatively correlated).

```
round(summary(ncpMod, pars=c("alpha", "beta"))$summary,2)
```

```
      mean se_mean   sd  2.5%  25%  50%  75% 97.5% n_eff Rhat
alpha 0.66       0 0.05  0.57 0.63 0.66 0.69  0.75  4000    1
beta  0.03       0 0.05 -0.07 0.00 0.03 0.06  0.12  4000    1
```

```
round(summary(massMod, pars=c("alpha", "beta"))$summary,2)
```

```
       mean se_mean   sd  2.5%   25%   50%   75% 97.5%   n_eff Rhat
alpha  0.66       0 0.05  0.56  0.63  0.66  0.69  0.75 3661.91    1
beta  -0.06       0 0.05 -0.15 -0.09 -0.06 -0.03  0.03 3305.13    1
```

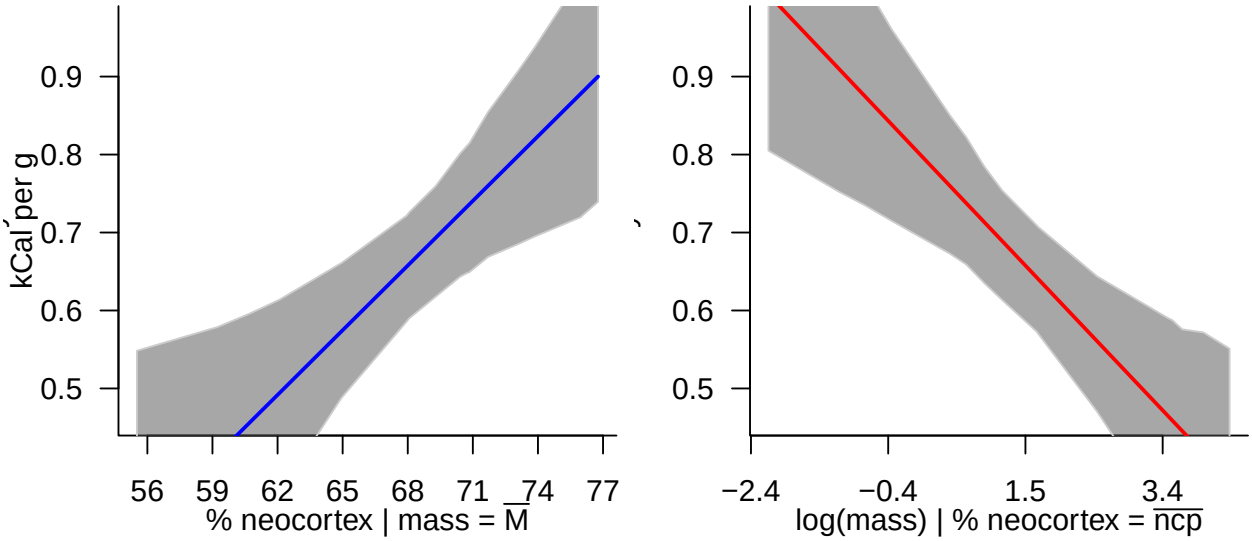
However, we get a much different result if we include both variables together:

```
dat <- list(nObs=nObs, nVar=2, obs=kcal, x1=as.vector(scale(ncp)),
  x2 = as.vector(scale(mass)), aSD=10, bSD=1, sigmaSD=10)

milkMod <- stan(file="multiMod.stan", data=dat, iter=2000,
 chains=4, seed=867.5309, pars="mu", include=FALSE)
```

```
round(summary(milkMod, pars=c("alpha", "beta"))$summary,2)
```

```
         mean se_mean   sd  2.5%   25%   50%   75% 97.5%   n_eff Rhat
alpha    0.66       0 0.03  0.59  0.64  0.66  0.68  0.72 2854.51    1
beta[1]  0.17       0 0.05  0.06  0.13  0.17  0.20  0.27 2226.40    1
beta[2] -0.19       0 0.05 -0.29 -0.22 -0.19 -0.15 -0.08 2378.59    1
```

If we visualize these with counterfactual plots, we see that the estimated association of both variables with `kcal` has increased.



- This occurs because both variables are correlated with `kcal`, but one has a postive effect and the other a negative effect.
  - Additionally, both variables are positively correlated with each other.
- Therefore the two variables tend to cancel each other out.

What the multivariate model does is ask if species that have a high % neocortex *for their body mass* have higher milk energy. We only see the effect if we control for both.