GSS_workflow

Rachel Kaufman

2022-11-26

My fake Exam!

Anything in italics represents questions I have about the material. I am writing them down to ask one of you in person (likely Steve because Nico is leaving), so no need to type out any answers when you look over this! My goal: I wanted to be able to think through this, but really wanted to question every detail to see where any of my confusion lies. In order to do this, I asked questions about each chunk answered them and then kept the remaining ones I was not positive of.

Hello Nico and Steve!!

My Estimand: How might income directly effect the happiness of ones marriage? Because of my interest with direct effects, I am going to introduce two additional variables to hold constant: sex and hours worked. I chose to include these because I considered them to be primary predictors of income.

Loading in the 2018 specific data into a lil object

```
gss_2018 <- gss_get_yr(2018)
```

```
## Fetching: https://gss.norc.org/documents/stata/2018_stata.zip
```

Now, I just want the variables I WANT to work on. #### **Data cleanup**

```
d <- gss_2018 %>%
  select(sex, rincome, satfin, hrs1, hapmar) %>% #hapcohab removed
  mutate(
    female = if_else(sex == 2, 1L, 0L),
    income = rincome,
    hrs_worked = hrs1) %>%
  select(female, income, hapmar, hrs_worked) %>%
  haven::zap_labels() %>%
    drop_na()   # this is saying if sex = 2 then it is assigned as 1 for the respondent being female, a
```

I also want to standardize these bad boys and compare the results between them, so I am doing that here.

```
d_standardized <- d %>%
   mutate(across(everything(),
                 ~ (.x - mean(.x)) / sd(.x)))

d_standardized2 <- d %>%
   mutate(hrs_worked = standardize(hrs_worked),
          income =  standardize(hrs_worked),
          hapmar = standardize(hrs_worked))
```

Then, here, I want to propose a dag. I need to load in the DAG software. #### **Creating a DAG**

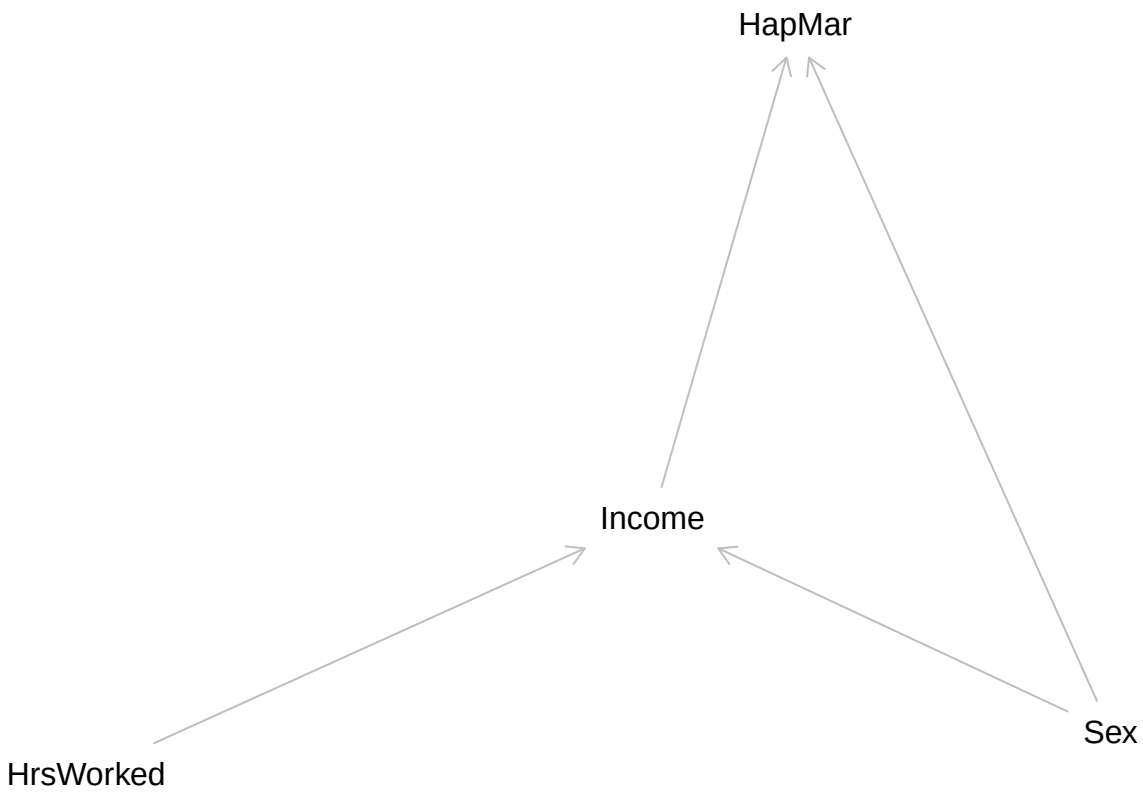
```
library(dagitty)
library(ggdag)
```

```
##
## Attaching package: 'ggdag'
```

```
## The following object is masked from 'package:stats':
##
##     filter
```

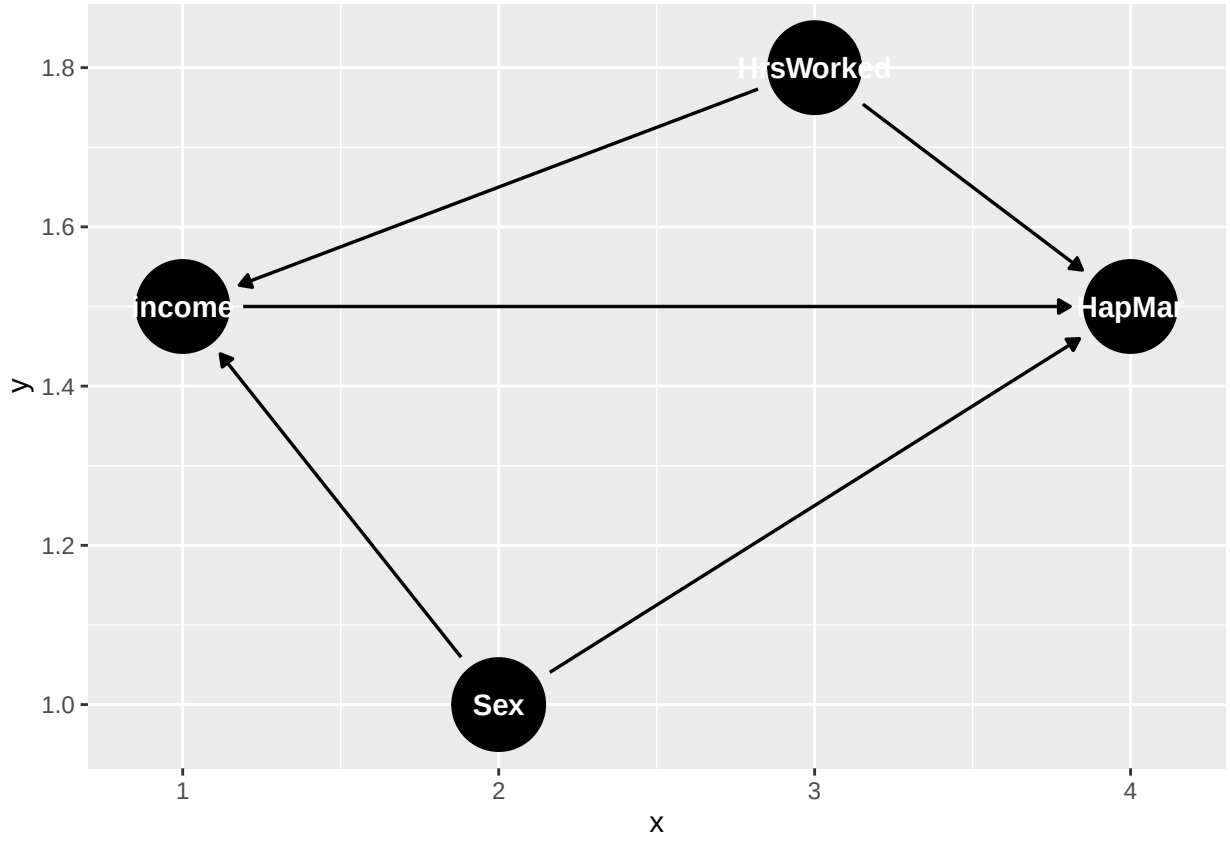
```
myDAG <- dagitty( "dag {Sex -> HapMar
                  HrsWorked -> Income Sex -> Income -> HapMar}" )
plot(myDAG)
```

```
## Plot coordinates for graph not supplied! Generating coordinates, see ?coordinates for how to set your
```




```
dag_coords <-
  tibble(
    name = c("income", "HrsWorked", "Sex", "HapMar"),
    x = c(1, 3, 2, 4),
    y = c(1.5, 1.8, 1, 1.5)
)
```

```
dag <- dagify(HapMar ~ income,
              HapMar ~ HrsWorked,
              income ~ Sex,
              HapMar ~ Sex,
              income ~ HrsWorked,
              coords = dag_coords)
ggdag(dag)
```




```
d <- tibble(d)
```

BOOM! Dagged. That was exciting. I fiddled with the coordinates for a good 20 mins. I put the diagram into dagiffy and it said to control for sex and Hrs Worked.

Time to set my priors... :) This first model I am doing is just treating each independent variable as, well, independent from each other.

```
set.seed(1201)
flist <- alist(
  hapmar ~ dnorm(mu, sigma),
  mu <- a + BI*income + BF*female + BH*hrs_worked,
  a ~ dnorm(0,1),
  BI ~ dnorm(0,2),
```

```
  BF ~ dnorm(0, .1),
  BH ~ dnorm(0, 5),
  sigma ~ dexp(1)
)
```

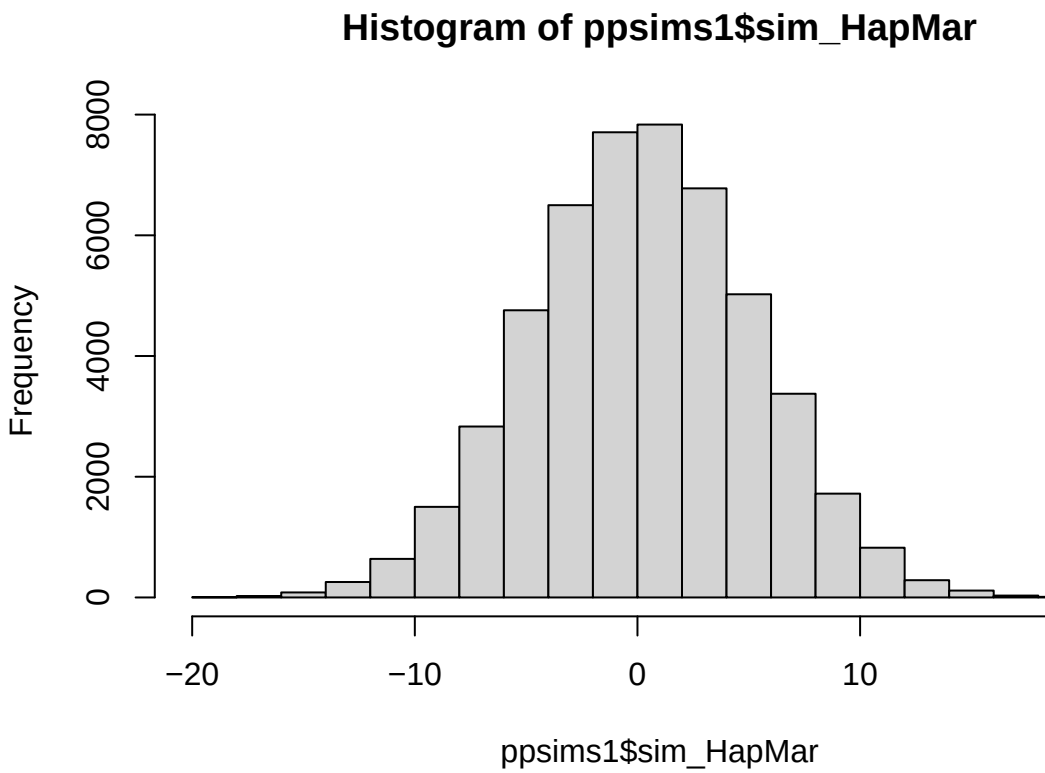
Constructing my model and setting new priors *If I were to make Female an index variable, how would I set my priors? My outcome variable is positively bound from 0-3. How might I set up my priors to reflect this?*

Now, because I am not super sure about these priors lets to a prior predictive simulation. This means that I am going to be using my independent variables and their newly assigned/chosen prior variables to predict what the happy marriage rates would be given this information. This does not use my actual outcome measurement, I will create a fake one (hence it being a simulation).

There are a lot of ways to do prior predictive simulations in my notes, but I am going to go with this way that Steve showed us on 11/17. Unsure if this is entirely correct but here we are. If it is not correct, yikes.

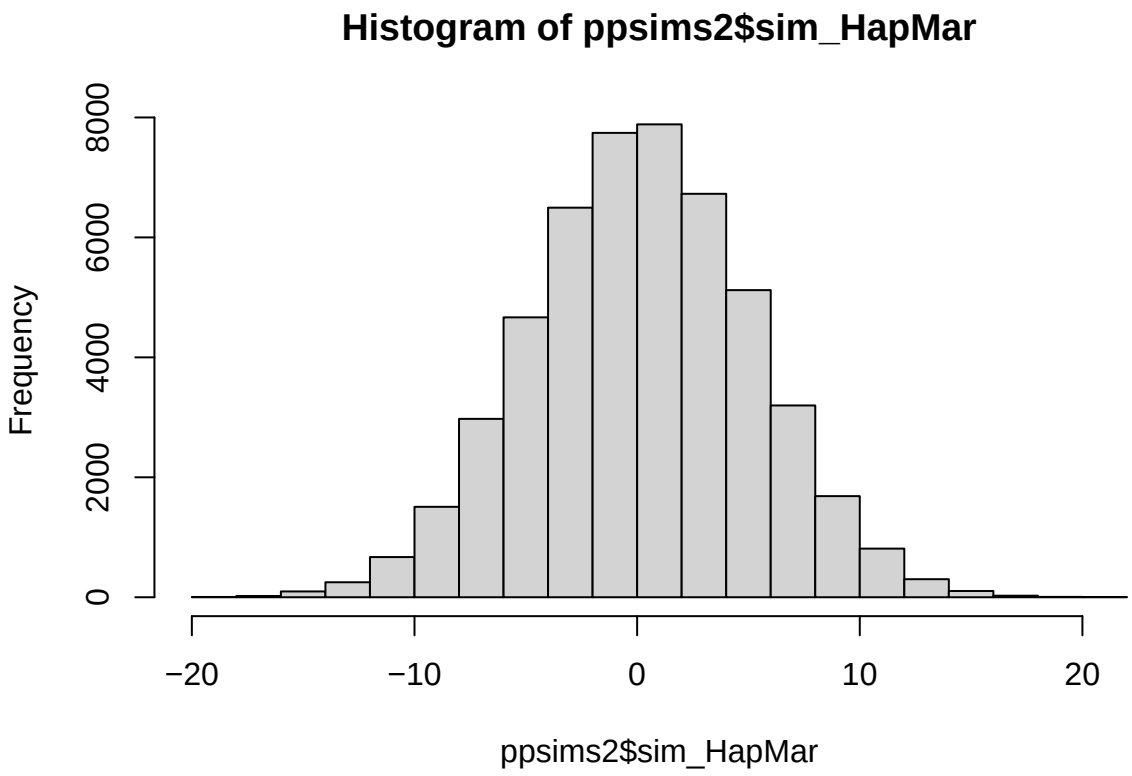
```
##for females
ppsims1 <- d %>%
  select(female) %>%
  uncount(100) %>%
  rowwise() %>%
   mutate(sim_HapMar = rnorm(1, 0.20, 0.05) + #alpha
            rnorm(1, 0, .1) + #Bi
            rnorm(1,0, .2) + #Bf
            rnorm(1,0, 5) #Bh
            )

##for Hrs Worked
ppsims2 <- d %>%
  select(hrs_worked) %>%
  uncount(100) %>%
  rowwise() %>%
   mutate(sim_HapMar = rnorm(1, 0.20, 0.05) + #alpha
            rnorm(1, 0, .1) + #Bi
            rnorm(1,0, .2) + #Bf
            rnorm(1,0, 5) #Bh
            )


##for Income
ppsims3 <- d %>%
  select(income) %>%
  uncount(100) %>%
  rowwise() %>%
   mutate(sim_HapMar = rnorm(1, 0.20, 0.05) + #alpha
            rnorm(1, 0, 1) + #Bi
            rnorm(1,0, .2) + #Bf
            rnorm(1,0, 5) #Bh
            )
```

```
hist(ppsims1$sim_HapMar)
```

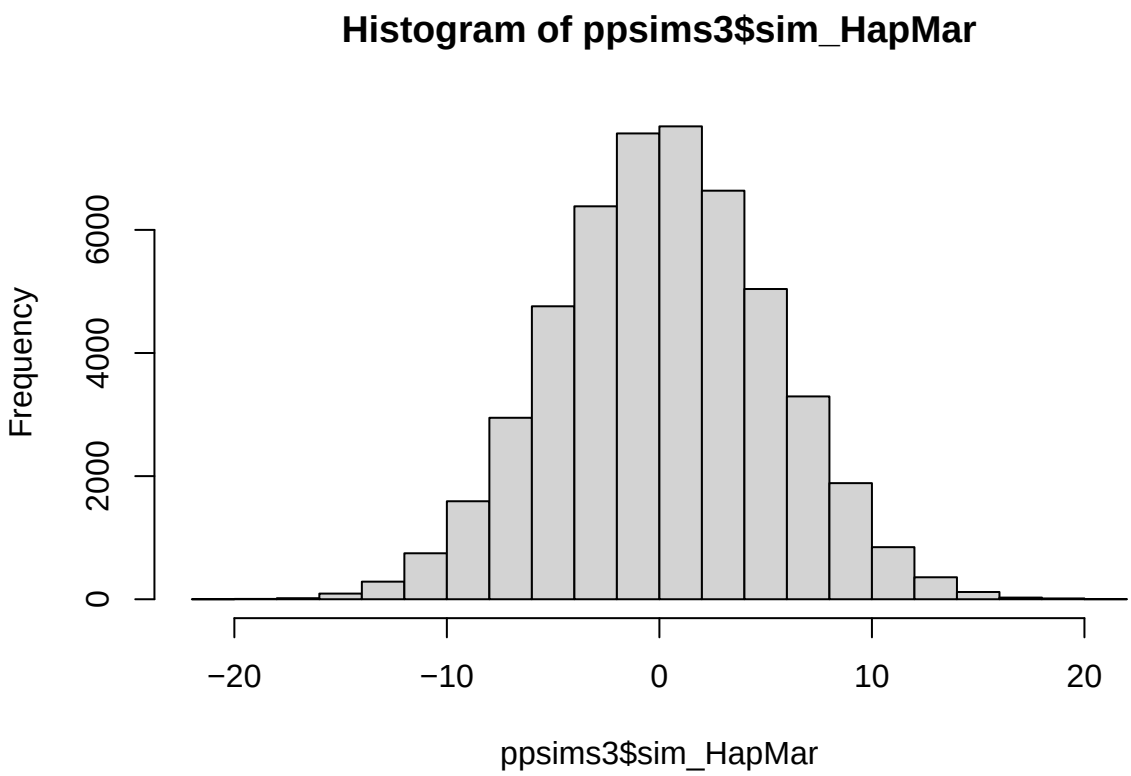


Prior predictive simulations

```
hist(ppsims2$sim_HapMar)
```




```
hist(ppsims3$sim_HapMar)
```



what is the first 1 for? Why do we use 1? Why do we keep mean values at 0 for betas? In our class example bc the outcome variable was binary, we set priors according to percentage points. When do we consider what priors are measured as percentage points and others are not?

Estimating the Model and Output Using `quap()`

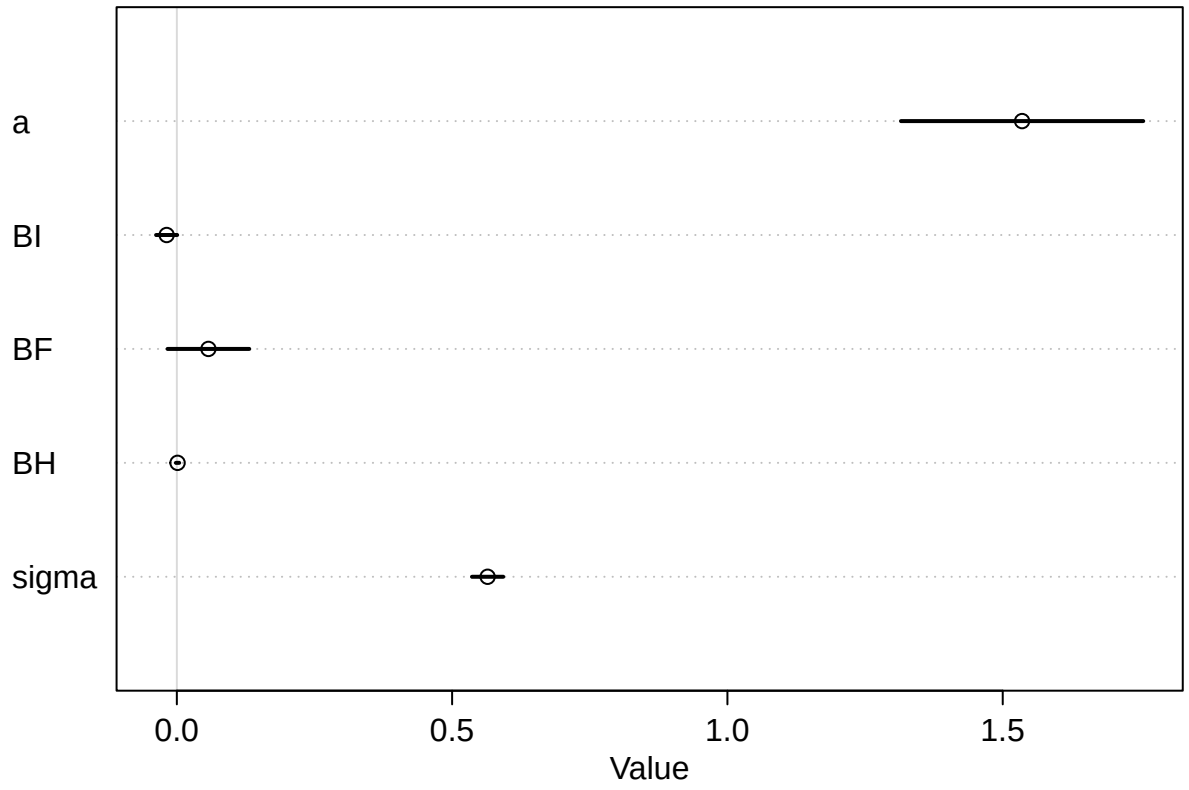
```
quap_m <- quap(flist, data = d)
quap_mstand <- quap(flist, data = d_standardized)
#quap_mstand2 <- quap(flist, data = d_standardized2)

precis_m <- precis(quap_m)
precis_mstand <- precis(quap_mstand)
#precis_mstand2 <- precis(quap_mstand2)

coef(quap_mstand)
```

```
##             a            BI            BF            BH         sigma
##  1.043667e-05 -8.039739e-02  5.015403e-02  3.123461e-02  9.933712e-01
```

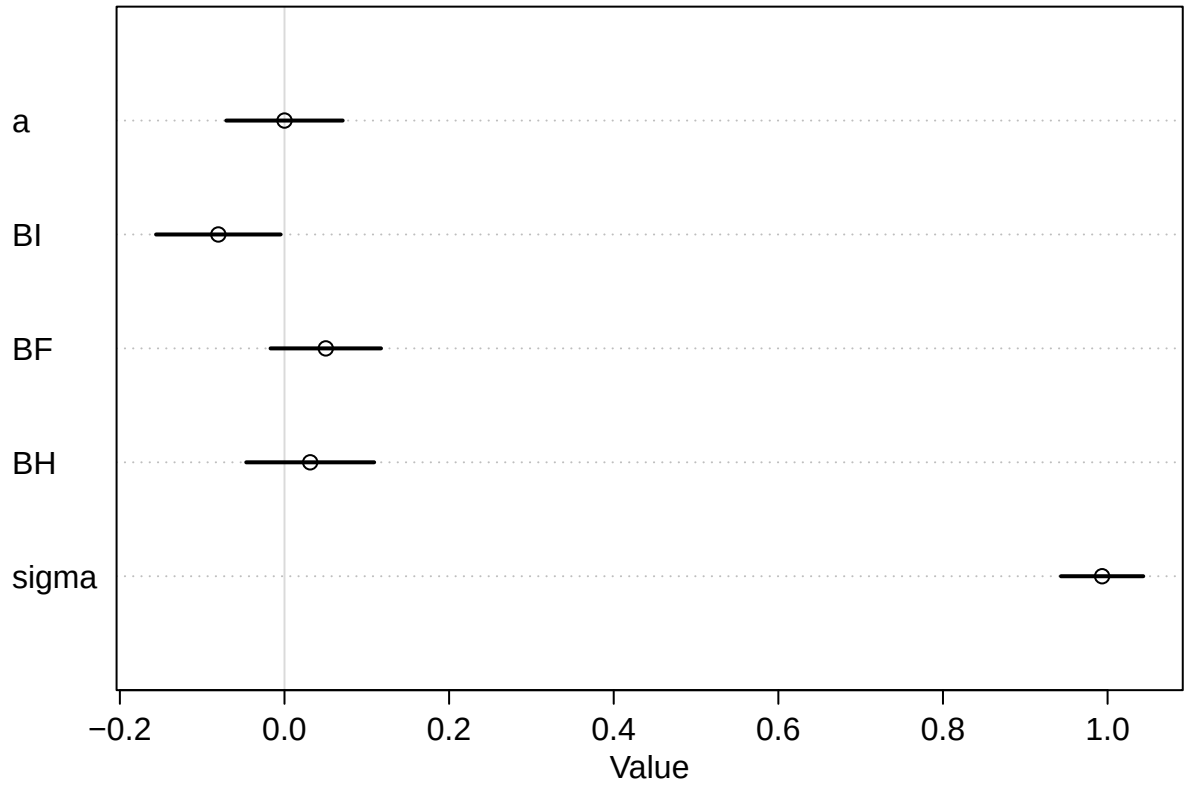
```
coef(quap_m)
```

```
##            a           BI           BF           BH        sigma
##  1.535205389 -0.018515367  0.057362609  0.001297076  0.564449990
```

```
#coef(quap_mstand2)

plot(precis_m) #unstandardized
```




```
plot(precis_mstand) #standardized
```




```
#plot(precis_mstand2)
```

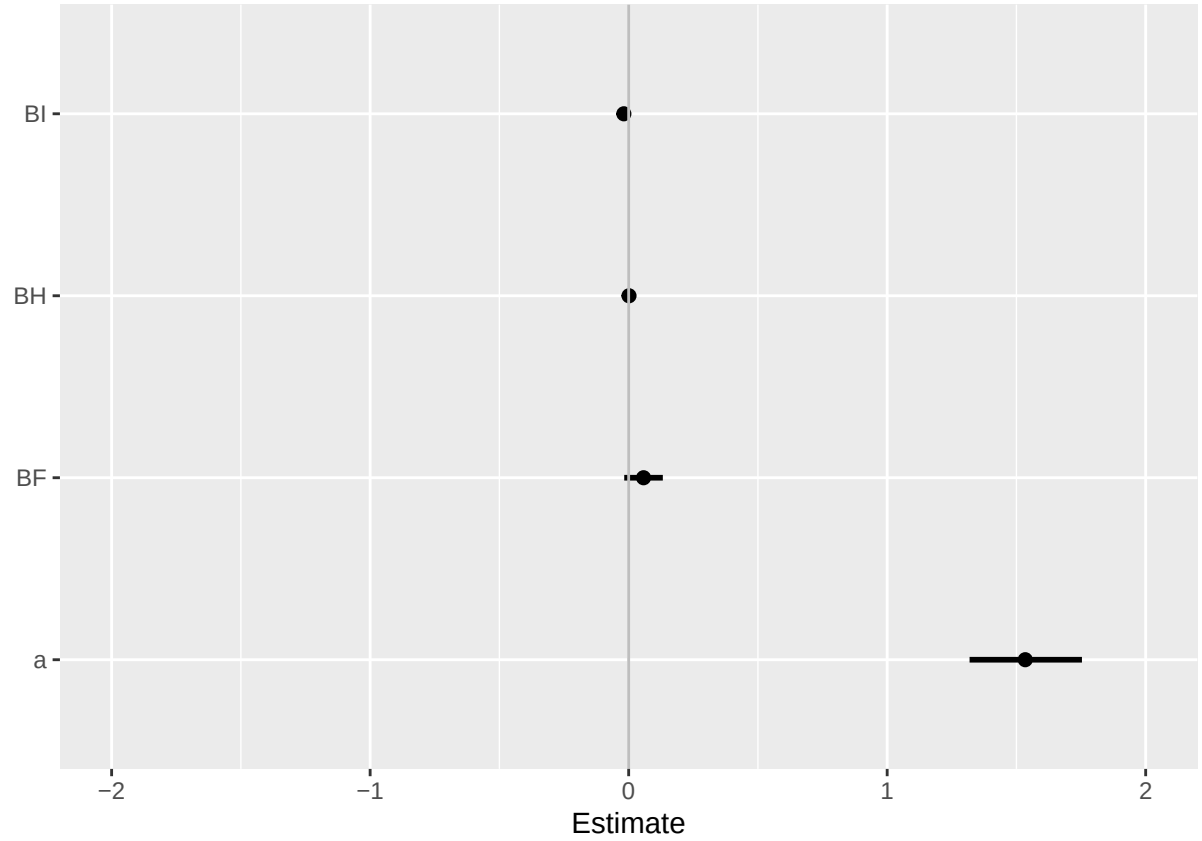
I remember that we had to add a 2 to the argument `precis_m <- precis(quap_m, 2)` *at some point in class. Why was that the case?* Well, this is kinda lame. Why is my intercept?????? So far away????? Well, I can answer this question. The mean value hovers around 1.5 for my intercept, which means when my other parameters are 0, happiness is at 1.5 which is literally the avg between 1,2 and 3... Also! This would be for men's happiness, because men are the 0 value for Bf. All my beta's seem to mean very little if not, nothing! Ha! That's okay. I think there are a few reasons this is the case in spite of it my gut saying these are important factors. This could be because I did not standardize. So, now I am going to go do that. Okay, I am back and this changes my intercept as well as my slope for income. These results are not intuitive, and the coeff plot shows a negative association for income. This means the higher the income, the more unhappy in ones marriage. So, saying at like that makees some sense, but it is hard to buy poor people being happy in their marriage. Considering its only slightly to the left (note that the sd intervals do not cross the 0), these preliminary results for a smaller magnitude in this neegative association sounds alright. I mean, that is what the data says and that can make logical sense. BUT I didn't check model fit so my rationale may just be hindsight bias. but there is an issue me thinks because my zero category was meaningful and I accidentally standardized it. Sheesh. Brb. I'm back! Okay, I had to fix the fact that I standardized my female variable on accident. AND THEN in doing that I realized, hey, maybe I should have considered adjusting my priors to match the fact that the numbers are now standardized. Ooops, im moving own with coding in tidy. I also turned the part 3 into comments because my priors were too crappy to continue. Honestly, I am probably going to keep working on this for fun tomrrow *How would indexing my variable change these results?*

Tidy style.

```
post <- tidy_draws(quap_m, n = 1e4) %>%
  select(a, BI, BH, BF)

long_post <- post %>%
  pivot_longer(everything(),
    names_to = "term",
    values_to = "values")

postsum <- long_post %>%
  group_by(term) %>%
  summarize(
    mean = mean(values),
    lb = quantile(values, .055),
    ub = quantile(values, .945))

plot
```

```
## standardGeneric for "plot" defined from package "base"
##
## function (x, y, ...)
## standardGeneric("plot")
## <environment: 0x7fdd6c6fdce0>
## Methods may be defined for arguments: x, y
## Use  showMethods(plot)  for currently available ones.
```

```
ggplot(
  postsum,
  aes(mean, term,
    xmin = lb,
    xmax = ub
  )
) +
  geom_point(size = 2) +
  geom_linerange(size = 1) +
  geom_vline(
    xintercept = 0,
    color = "gray"
  ) +
  theme(legend.position = "none") +
  xlim(-2, 2) +
  labs(
    x = "Estimate",
    y = ""
  )
```



Wooohoo! I wanted to make my coef in tidy style. I am proud for this posterior distribution check. *I know it is largely circumstantial but how do we know when it is a good idea to use a interacting or index variable*

In the future, I would have made a few more models and done comparisons of model fit... But, I timed myself to today so I could do as much as I can that I actually knew. TahDahhh

P.S.

Dr. Nico, you have been and are fantastic! No wonder why UC Davis scooped you. Thank you for all that you do! I felt confident of my choice in Duke because of your positive energy on my visit. UC Davis bettere be HONORED to have you.

Promise that is a causal claim! See you soon and safe travels.