hw12 for stat341

Zhihong Zhang

April 5th,2017

Q: 7H3 8H3- 8H4

7H3. Consider again the data(rugged) data on economic development and terrain ruggedness, examined in this chapter. One of the African countries in that example, Seychelles, is far outside the cloud of other nations, being a rare country with both relatively high GDP and high ruggedness.Seychelles is also unusual, in that it is a group of islands far from the coast of mainland Africa, and its main economic activity is tourism. One might suspect that this one nation is exerting a strong influence on the conclusions. In this problem, I want you to drop Seychelles from the data and re-evaluate the hypothesis that the relationship of African economies with ruggedness is different from that on other continents. (a) Begin by using map to fit just the interaction model: $y_i $Normal$(_i,)$ $\mu_i = \alpha + \beta_A A_i + \beta_R R_i + \beta_{AR} A_i R_i$ where y is log GDP per capita in the year 2000 (log of rgdppc_2000); A is cont_africa, the dummy variable for being an African nation; and R is the variable rugged. Choose your own priors. Compare the inference from this model fit to the data without Seychelles to the same model fit to the full data. Does it still seem like the effect of ruggedness depends upon continent? How much has the expected relationship changed?

b. Now plot the predictions of the interaction model, with and without Seychelles. Does it still seem like the effect of ruggedness depends upon continent? How much has the expected relationship changed?

c. Finally, conduct a model comparison analysis, using WAIC. Fit three models to the data without Seychelles: Model 1 : $yi $ Normal(μ_i, σ) $i = + _RR_i + $ Model 2 : y$yi $ Normal(μ_i, σ) $i = + _AA_i + _RR_i $ Model 3 : $yi $ Normal(μ_i, σ) $\mu i = \alpha + \beta_A A_i + \beta_R R_i + \beta_{AR} A_i R_i$ Use whatever priors you think are sensible. Plot the model-averaged predictions of this model set. Do your inferences differ from those in (b)? Why or why not?

Solutions:

a.

```
data(rugged)
Nations <- rugged %>%
  mutate(log_gdp = log(rgdppc_2000)) %>%
  filter(!is.na(rgdppc_2000))

Nationswosyc<- Nations[c(1:144,146:170),]  #select the countries without Seychelles whi
ch is in line 145

model <-
  map(
    alist(
      log_gdp ~ dnorm(mu, sigma),
      mu <- a + bA * cont_africa + bR * rugged + bAR * rugged * cont_africa ,
      a ~ dnorm(8, 100),
      bA ~ dnorm(0, 1),
      bR ~ dnorm(0, 1),
      bAR ~ dnorm(0, 1),
      sigma ~ dunif(0, 10)
    ),
    data = Nations
  )


modelwosyc <-
  map(
    alist(
      log_gdp ~ dnorm(mu, sigma),
      mu <- a + bA * cont_africa + bR * rugged + bAR * rugged * cont_africa ,
      a ~ dnorm(8, 100),
      bA ~ dnorm(0, 1),
      bR ~ dnorm(0, 1),
      bAR ~ dnorm(0, 1),
      sigma ~ dunif(0, 10)
    ),
    data = Nationswosyc
  )
#WAIC for model with all nations
WAIC(model)
```

```
## Constructing posterior predictions
```

```
## [ 100 / 1000 ]
[ 200 / 1000 ]
[ 300 / 1000 ]
[ 400 / 1000 ]
[ 500 / 1000 ]
[ 600 / 1000 ]
[ 700 / 1000 ]
[ 800 / 1000 ]
[ 900 / 1000 ]
[ 1000 / 1000 ]
```

```
## [1] 469.809
## attr(,"lppd")
## [1] -229.5121
## attr(,"pWAIC")
## [1] 5.392372
## attr(,"se")
## [1] 15.11217
```

```
#WAIC for model without Seychelles
WAIC(modelwosyc)
```

```
## Constructing posterior predictions
```

```
## [ 100 / 1000 ]
[ 200 / 1000 ]
[ 300 / 1000 ]
[ 400 / 1000 ]
[ 500 / 1000 ]
[ 600 / 1000 ]
[ 700 / 1000 ]
[ 800 / 1000 ]
[ 900 / 1000 ]
[ 1000 / 1000 ]
```

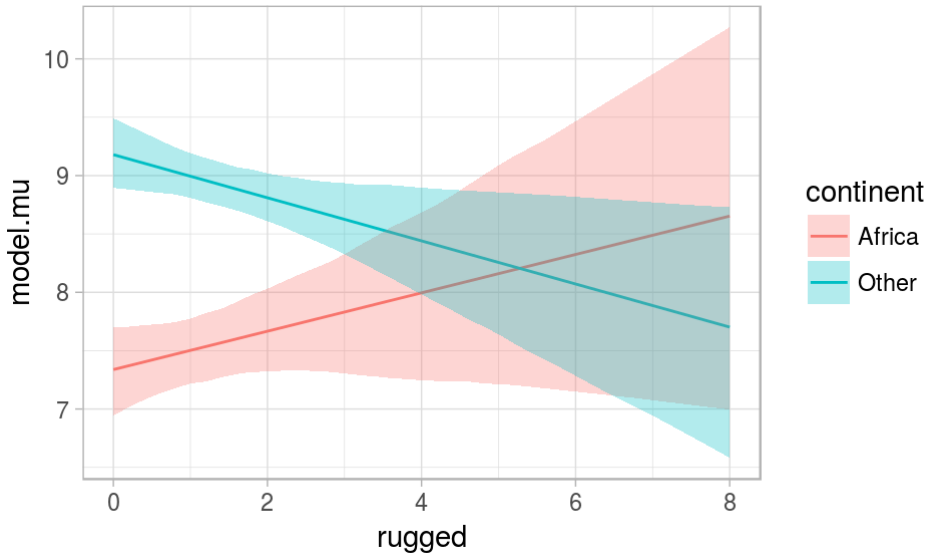
```
## [1] 463.4362
## attr(,"lppd")
## [1] -227.0951
## attr(,"pWAIC")
## [1] 4.623012
## attr(,"se")
## [1] 15.00738
```

b. For the models with Seychelles

```
Loggdp_predict <-
  expand.grid(
    cont_africa = 0:1,
    rugged = seq(from = 0, to = 8, by = 0.2)
  ) %>%
  mutate(
    continent = ifelse(cont_africa, "Africa", "Other")
  )


model_link <- link(model, data = Loggdp_predict, refresh = 0)

# add in means and intervals
Loggdp_predict <-
  Loggdp_predict %>%
  mutate(
   model.mu = apply(model_link, 2, mean),
    model.lo = apply(model_link, 2, PI, prob = 0.97)[1,],
    model.hi = apply(model_link, 2, PI, prob = 0.97)[2,]
  )
gf_line( model.mu ~ rugged + color:continent,
        data = Loggdp_predict) %>%
gf_ribbon( model.lo +  model.hi ~ rugged + fill:continent,
        data = Loggdp_predict)
```

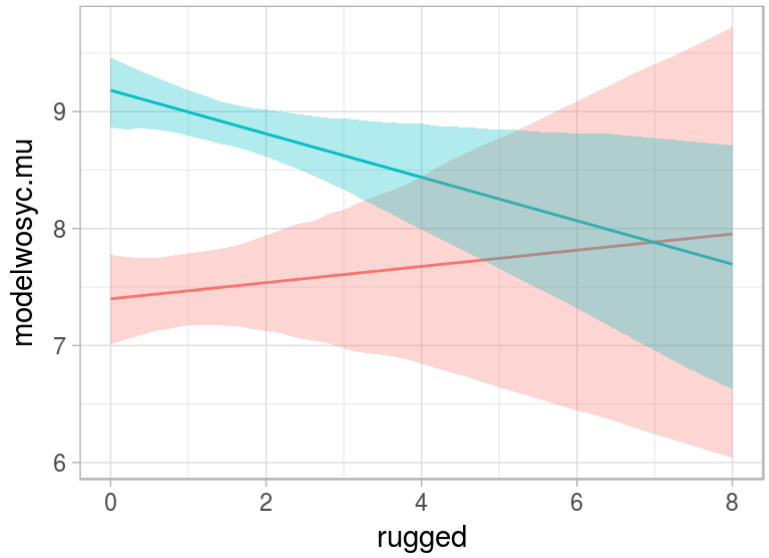


For the models without Seychelles

```
Loggdp_predict <-
  expand.grid(
    cont_africa = 0:1,
    rugged = seq(from = 0, to = 8, by = 0.2)
  ) %>%
  mutate(
    continent = ifelse(cont_africa, "Africa", "Other")
  )


modelwosyc_link <- link(modelwosyc, data = Loggdp_predict, refresh = 0)

# add in means and intervals
Loggdp_predict <-
  Loggdp_predict %>%
  mutate(
  modelwosyc.mu = apply(modelwosyc_link, 2, mean),
  modelwosyc.lo = apply(modelwosyc_link, 2, PI, prob = 0.97)[1,],
  modelwosyc.hi = apply(modelwosyc_link, 2, PI, prob = 0.97)[2,]
  )
gf_line( modelwosyc.mu ~ rugged + color:continent,
        data = Loggdp_predict) %>%
gf_ribbon( modelwosyc.lo +  modelwosyc.hi ~ rugged + fill:continent,
        data = Loggdp_predict)
```



Comparing two plot, the effect of ruggedness does depend on the continent. the slope of the Africa contry decrease a bit after dropping Seychelles

c.

```
model1 <-
  map(
    alist(
      log_gdp ~ dnorm(mu, sigma),
      mu <- a + bR * rugged  ,
      a ~ dnorm(8, 100),

      bR ~ dnorm(0, 1),
      sigma ~ dunif(0, 10)
    ),
    data =Nationswosyc
  )


model2 <-
  map(
    alist(
      log_gdp ~ dnorm(mu, sigma),
      mu <- a + bA * cont_africa + bR * rugged,
      a ~ dnorm(8, 100),
      bA ~ dnorm(0, 1),
      bR ~ dnorm(0, 1),
      sigma ~ dunif(0, 10)
    ),
    data = Nationswosyc
  )







model3 <-
  map(
    alist(
      log_gdp ~ dnorm(mu, sigma),
      mu <- a + bA * cont_africa + bR * rugged + bAR * rugged * cont_africa ,
      a ~ dnorm(8, 100),
      bA ~ dnorm(0, 1),
      bR ~ dnorm(0, 1),
      bAR ~ dnorm(0, 1),
      sigma ~ dunif(0, 10)
    ),
    data =Nationswosyc
  )


compare(model1,model2,model3)
```

```
##         WAIC pWAIC dWAIC weight    SE   dSE
## model3 463.7   4.7   0.0   0.74 15.08    NA
## model2 465.8   3.8   2.1   0.26 14.23  3.26
## model1 536.0   2.6  72.4   0.00 13.39 15.21
```

```
model.ensemble <- ensemble(model1,model2,model3,data = Loggdp_predict)
```

```
## Constructing posterior predictions
```

```
## [ 100 / 1000 ]
[ 200 / 1000 ]
[ 300 / 1000 ]
[ 400 / 1000 ]
[ 500 / 1000 ]
[ 600 / 1000 ]
[ 700 / 1000 ]
[ 800 / 1000 ]
[ 900 / 1000 ]
[ 1000 / 1000 ]
```

```
## Constructing posterior predictions
```

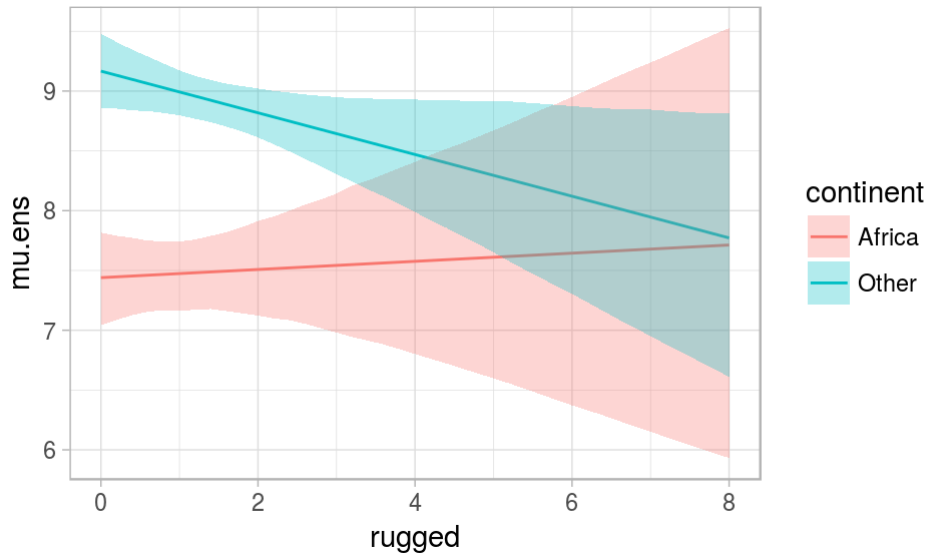
```
## [ 100 / 1000 ]
[ 200 / 1000 ]
[ 300 / 1000 ]
[ 400 / 1000 ]
[ 500 / 1000 ]
[ 600 / 1000 ]
[ 700 / 1000 ]
[ 800 / 1000 ]
[ 900 / 1000 ]
[ 1000 / 1000 ]
```

```
## Constructing posterior predictions
```

```
## [ 100 / 1000 ]
[ 200 / 1000 ]
[ 300 / 1000 ]
[ 400 / 1000 ]
[ 500 / 1000 ]
[ 600 / 1000 ]
[ 700 / 1000 ]
[ 800 / 1000 ]
[ 900 / 1000 ]
[ 1000 / 1000 ]
```

```
Loggdp_predict <-
 Loggdp_predict %>%
  mutate(
    mu.ens =  apply(model.ensemble$link, 2, mean),
    mu.ens.lo =  apply(model.ensemble$link, 2, PI,prob=0.97)[1,],
    mu.ens.hi =  apply(model.ensemble$link, 2, PI,prob=0.97)[2,]
  )
#combined model for all 3 models

gf_line( mu.ens ~ rugged + color:continent,
        data = Loggdp_predict) %>%
gf_ribbon( mu.ens.lo +  mu.ens.hi ~ rugged + fill:continent,
        data = Loggdp_predict)
```



The model-average production is more closed to model 3 and it is really similar to part(b)

8H3. Sometimes changing a prior for one parameter has unanticipated effects on other parameters. This is because when a parameter is highly correlated with another parameter in the posterior, the prior influences both parameters. Here is an example to work and think through. Go back to the leg length example in Chapter 5. Here is the code again, which simulates height and leg lengths for 100 imagined individuals:

```
 #R code8.21
set.seed(100)
N <- 100 # number of individuals
height <- rnorm(N,10,2) # sim total height of each
leg_prop <- runif(N,0.4,0.5) # leg as proportion of height
leg_left <- leg_prop*height + # sim left leg as proportion + error
rnorm( N , 0 , 0.02 )
leg_right <- leg_prop*height + # sim right leg as proportion + error
rnorm( N , 0 , 0.02 )
d <- data.frame(height,leg_left,leg_right)
```

And below is the model you fit before, resulting in a highly correlated posterior for the two beta parameters. This time, fit the model using map2stan: R code

```
show(m5.8s)
```

```
## map2stan model fit
## 4000 samples from 4 chains
##
## Formula:
## height ~ dnorm(mu, sigma)
## mu <- a + bl * leg_left + br * leg_right
## a ~ dnorm(10, 100)
## bl ~ dnorm(2, 10)
## br ~ dnorm(2, 10)
## sigma ~ dcauchy(0, 1)
##
## Log-likelihood at expected values: -93.56
## Deviance: 187.12
## DIC: 194.66
## Effective number of parameters (pD): 3.77
##
## WAIC (SE): 194.87 (14)
## pWAIC: 3.78
```

Compare the posterior distribution produced by the code above to the posterior distribution produced when you change the prior for br so that it is strictly positive:

```
show(m5.8s2)
```

```
## map2stan model fit
## 4000 samples from 4 chains
##
## Formula:
## height ~ dnorm(mu, sigma)
## mu <- a + bl * leg_left + br * leg_right
## a ~ dnorm(10, 100)
## bl ~ dnorm(2, 10)
## br ~ dnorm(2, 10) & T[0, ]
## sigma ~ dcauchy(0, 1)
##
## Log-likelihood at expected values: -93.93
## Deviance: 187.86
## DIC: 194.51
## Effective number of parameters (pD): 3.32
##
## WAIC (SE): 194.71 (13.9)
## pWAIC: 3.36
```

Note that T[0,] on the right-hand side of the prior for br. What the T[0,] does is truncate the normal distribution so that it has positive probability only above zero. In other words, that prior ensures that the posterior distribution for br will have no probability mass below zero. Compare the two posterior distributions for m5.8s and m5.8s2. What has changed in the posterior distribution of both beta parameters? Can you explain the change induced by the change in prior?

Solutions:

Compare these two posterior distributions for m5.8s and m5.8s2, both model has very closed WAIC values.

8H4. For the two models fit in the previous problem, use DIC or WAIC to compare the effective numbers of parameters for each model. Which model has more effective parameters? Why?

Solutions:

For using DIC and WAIC, model m5.8s has more effective number.

```
compare(m5.8s,m5.8s2)
```

```
##         WAIC pWAIC dWAIC weight    SE  dSE
## m5.8s2 194.7   3.4   0.0   0.52 13.88   NA
## m5.8s  194.9   3.8   0.2   0.48 14.05 1.52
```

```
DIC(m5.8s,m5.8s2)
```

```
## [1] 194.6589
## attr(,"pD")
## [1] 3.767392
```