**Assignment 4**

**Info**

- Indicate which task the code belongs to by adding a line `#tasknumber` over the code for the respective task.
- `# comment` your code by describe what the code is supposed to do.
- At the beginning of the script, specify which packages you used (`library(package)`)

## Ecological Fallacy

1. Write a generative simulation involving a predictor variable $X$, an outcome variable $Y$ and a grouping variable $G \in \{1, 2\}$. The simulation should *randomly* generate a total number of $N = 500$ observations and match the following constraints:

- Approximately half of the observations are clustered in each group, $N_1, N_2 \approx 250$.
- $X_G$ is normally distributed with group means $\overline{X_1} \approx 100$ and $\overline{X_2} \approx 80$ and standard deviations of 15.
- $Y_{G,i}$ is normally distributed with a standard deviation of 10 and group means $\overline{Y}_{G,i} = 50 + b_G X_i$.
- The effects of $X$ on $Y$ are $b_1 = 0$ and $b_2 = -.5$.

2. Create one or more visualizations of the relation between $X$ and $Y$ illustrating the group differences in $X$, $Y$, and $b$ and the ecological fallacy that would result from ignoring the group differences.

3. Analyze the simulated data using a Bayesian linear model predicting $Y$ from $X$. Use quadratic approximation or MCMC. Use informative priors that don't match the parameters from the generative simulation, e.g., place a larger proportion of prior mass on parameter values you know to be false. (Intentionally setting "bad" priors here illustrates how the models learn from the data.)

- *Model 1*: Run a model that doesn't account for group differences to statistically corroborate the ecological fallacy shown in the previous visualizations.
- *Model 2*: Run a model that does account for group differences and captures the parameter settings from the generative simulation. How many priors are required to estimate the model?

For Tasks 4 and 5, continue with *Model 2*.

4. Compute and plot the posterior distribution of differences for the group-specific slope parameters using 1,000 samples from the joint posterior distribution.

5. Conduct a posterior predictive check.

- For each $X$ value simulated in Task 1, predict a $Y$ value using the posterior distributions from *Model 2*.
- Create one or more visualizations of the posterior predictions illustrating the ability of the model to capture and predict the association of $X$ and $Y$.
  Tipp: To predict values for N people, you should use/draw N samples from the posterior distribution. Also consider possible differences in group size.

## More Groups in Data

Read in the data set `WorldData.csv`. For the remaining tasks, you will focus on the relations between *life expectancy*, *freedom of choice*, and the regional data (*region*/*region_index*/*country*). Briefly familiarize yourself with the variables using visualizations and descriptive statistics.

*Tipp*: All numerical variables but `log_gdp` are standardized.

The goal is to study the association between *freedom of choice* and *life expectancy*. There will be many possible and rather complex causal routes linking the two measures, involving the influence of large set of confounds. A proper statistical model would require a lot more

thought than can be asked for in this assignment. In the following, we only stratify by region, assuming that some of the more impactful confounds are embedded therein, and then we hope for the best.

6. Compute the frequency distribution for the variable *region*. Briefly discuss potential problems we could run in when stratifying for region—that is, when computing a separate regression model for each region.

7. Analyze the data while stratifying for region using quadratic approximation or MCMC.

- *Model 3*: Estimate a Bayesian Gaussian model for the variable *life expectancy*.
- *Model 4*: Estimate a Bayesian linear model predicting *life expectancy* from *freedom of choice*.

8. Below, you find the code and output of a multilevel model for the association between of *freedom of choice*and *life expectancy*. How and why do the estimates differ compared to the previous *Model 4*?

```
m5 <- ulam(
  alist(
    life_expectancy ~ dnorm(mu, sigma) ,
    mu <- a[region_index] + b[region_index]*freedom_of_choice ,
    a[region_index] ~ dnorm(a_bar, a_tau) ,
    b[region_index] ~ dnorm(b_bar, b_tau) ,
    a_bar ~ dnorm(0,2) ,
    a_tau ~ dexp(1) ,
    b_bar ~ dnorm(0,2) ,
    b_tau ~ dexp(1) ,
    sigma ~ dexp(1)
    ), data = subset, chains = 8, cores = 8, iter = 4000)
```

```
precis(m5, depth = 2)
```

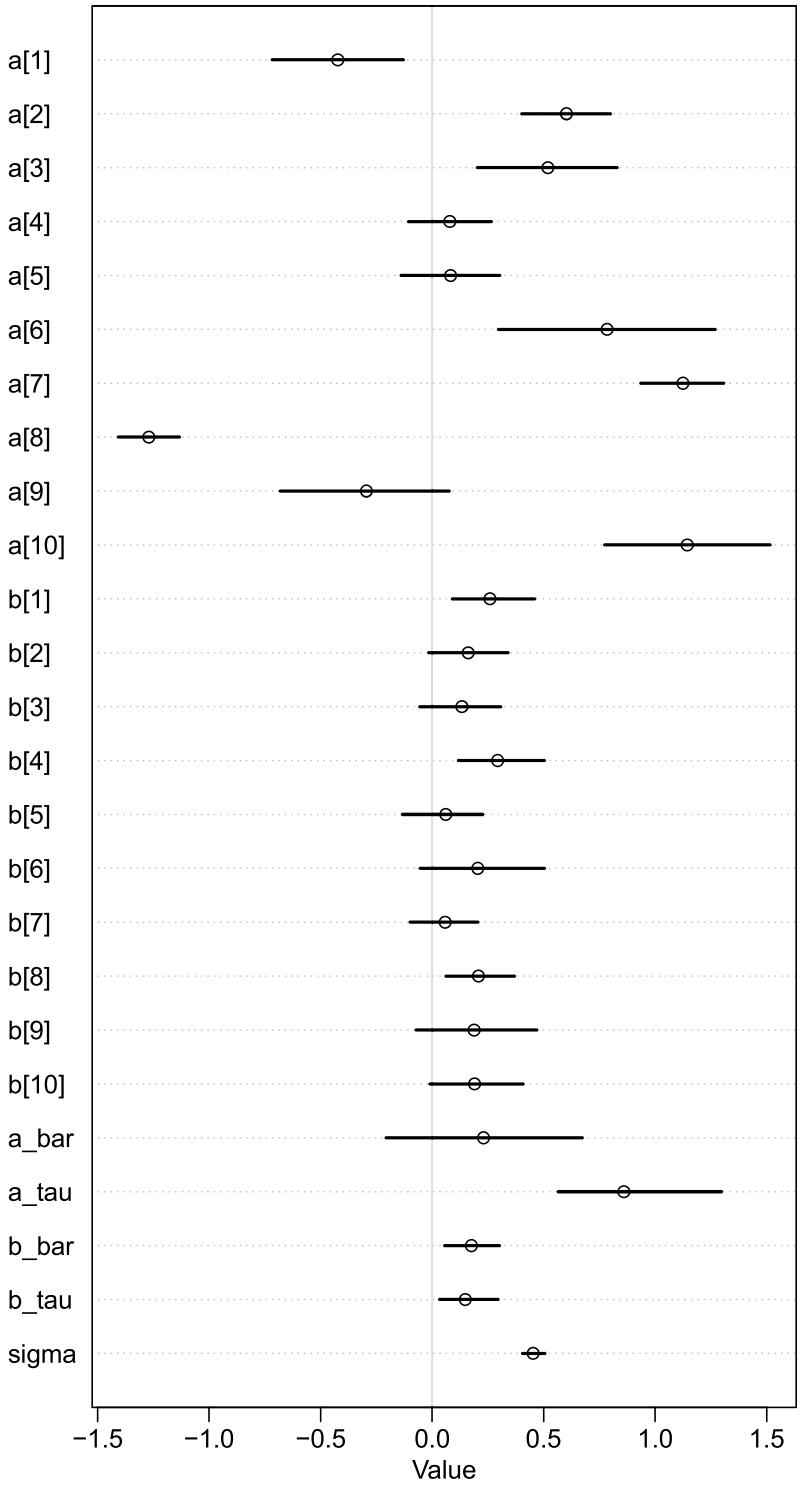
```
             mean         sd         5.5%       94.5%     n_eff     Rhat4
a[1]  -0.42271128 0.18514882 -0.717001610 -0.12916557 11540.646 1.0007896
a[2]   0.60255019 0.12481627  0.402043105  0.79974828 16697.476 0.9998638
a[3]   0.51951862 0.19625399  0.203082915  0.82908843 11612.011 1.0002615
```

```
a[4]   0.07922651 0.11626363 -0.105526390  0.26533700  9319.283 1.0005897
a[5]   0.08301678 0.13766385 -0.139250495  0.30375438 15689.531 1.0001304
a[6]   0.78494412 0.30232107  0.297588000  1.26900550 10726.762 1.0002412
a[7]   1.12442517 0.11612825  0.936129305  1.30744210  2350.246 1.0026805
a[8]  -1.26950620 0.08552206 -1.407420000 -1.13312780 17151.565 1.0002948
a[9]  -0.29467879 0.23949301 -0.680453920  0.07632125 10175.867 1.0001582
a[10]  1.14383817 0.23150345  0.773119000  1.51595605  5361.909 1.0022752
b[1]   0.25966331 0.11577386  0.090956716  0.45938838  5987.312 1.0008757
b[2]   0.16208709 0.11113995 -0.015446859  0.34031415 13409.270 1.0000330
b[3]   0.13367955 0.11214443 -0.054984530  0.30716686  9862.227 1.0006713
b[4]   0.29419460 0.12097040  0.118037250  0.50328242  3845.571 1.0014651
b[5]   0.06133624 0.11435377 -0.133647330  0.22727237  3751.034 1.0012596
b[6]   0.20448960 0.18141178 -0.053347687  0.50346782  8252.991 1.0002278
b[7]   0.05835526 0.09556652 -0.098893821  0.20577855  2185.732 1.0021278
b[8]   0.20808099 0.09625125  0.062614991  0.36888992 10885.327 1.0005495
b[9]   0.18845623 0.17226148 -0.071231129  0.46941874  8244.555 1.0007063
b[10]  0.19047474 0.13043986 -0.009212721  0.40795770 12000.799 1.0007516
a_bar  0.23034056 0.27989371 -0.206787245  0.67347886 12941.707 0.9998289
a_tau  0.85969889 0.23505070  0.565240710  1.29692055  2879.880 1.0035596
b_bar  0.17648592 0.07914785  0.055746823  0.30182655  7334.869 1.0014379
b_tau  0.14852799 0.08568764  0.033404189  0.29654882  1550.834 1.0026400
sigma  0.45298680 0.03161672  0.405544780  0.50520294  6392.021 1.0013227
```

```
coef(m5)[1:20]
```

```
      a[1]       a[2]       a[3]       a[4]       a[5]       a[6]
-0.42271128  0.60255019  0.51951862  0.07922651  0.08301678  0.78494412
      a[7]       a[8]       a[9]      a[10]       b[1]       b[2]
 1.12442517 -1.26950620 -0.29467879  1.14383817  0.25966331  0.16208709
      b[3]       b[4]       b[5]       b[6]       b[7]       b[8]
 0.13367955  0.29419460  0.06133624  0.20448960  0.05835526  0.20808099
      b[9]      b[10]
 0.18845623  0.19047474
```

```
plot(precis(m5, depth = 2))
```


a[1]
a[2]
a[3]
a[4]
a[5]
a[6]
a[7]
a[8]
a[9]
a[10]
b[1]
b[2]
b[3]
b[4]
b[5]
b[6]
b[7]
b[8]
b[9]
b[10]
a_bar
a_tau
b_bar
b_tau
sigma
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
Value