

## STATISTICAL RETHINKING 2025

### WEEK 8 SOLUTIONS

1. The dyad model from the book and lecture can be used almost unmodified, except the probability of the like nominations should be binomial, because it has a known maximum of 3. Here is the modified model:

```
library(rethinking)
# if you have the file:
# dat <- read.csv("week08_Monks.csv")
# otherwise
data(Monks)
dat <- Monks

# dyad model
f_dyad <- alist(
  likeAB ~ binomial( 3 , pAB ),
  likeBA ~ binomial( 3 , pBA ),
  logit(pAB) <- a + T[D,1] ,
  logit(pBA) <- a + T[D,2] ,
  a ~ normal(0,1),

  ## dyad effects - non-centered
  transpars> matrix[N_dyads,2]:T <-
    compose_noncentered( rep_vector(sigma_T,2) , L_Rho_T , Z ),
  matrix[2,N_dyads]:Z ~ normal( 0 , 1 ),
  cholesky_factor_corr[2]:L_Rho_T ~ lkj_corr_cholesky( 2 ),
  sigma_T ~ exponential(1),

  ## compute correlation matrix for dyads
  gq> matrix[2,2]:Rho_T <-<- Chol_to_Corr( L_Rho_T )
)

n_dyads <- nrow(dat)
monk_dat <- list(
  N_dyads = nrow(dat),
  D = dat$dyad_id,
  likeAB = dat$like_AB,
  likeBA = dat$like_BA )

m1 <- ulam( f_dyad , data=monk_dat , chains=4 , cores=4 , iter=2000 )
precis(m1,depth=3,pars=c("Rho_T","sigma_T"))
```

	mean	sd	5.5%	94.5%	rhat	ess_bulk
Rho_T[1,1]	1.00	0.00	1.00	1.00	NA	NA
Rho_T[2,1]	0.63	0.11	0.45	0.79	1	1251.78
Rho_T[1,2]	0.63	0.11	0.45	0.79	1	1251.78
Rho_T[2,2]	1.00	0.00	1.00	1.00	NA	NA
sigma_T	3.39	0.42	2.75	4.10	1	1560.73

So the posterior correlation within dyads is 89% between 0.45 – 0.79. Is that a lot of reciprocity? Hard to say. We haven't done anything to deal with properties of the monks themselves and generalized liking. Also these data were collected by asking for top 3 likes. So it could be that there is more reciprocity in the total network, outside top 3 likes.

2. This model just needs dislikes sections that mimic the code for the like sections.

```
# dyad model with dislike
f_dyad2 <- alist(
  likeAB ~ binomial( 3 , pAB ),
  likeBA ~ binomial( 3 , pBA ),
  logit(pAB) <- a[1] + T[D,1] ,
  logit(pBA) <- a[1] + T[D,2] ,

  dislikeAB ~ binomial( 3 , qAB ),
  dislikeBA ~ binomial( 3 , qBA ),
  logit(qAB) <- a[2] + T2[D,1] ,
  logit(qBA) <- a[2] + T2[D,2] ,
  vector[2]:a ~ normal(0,1),

  ## like dyad effects - non-centered
  transpars> matrix[N_dyads,2]:T <-
    compose_noncentered( rep_vector(sigma_T,2) , L_Rho_T , Z ),
  matrix[2,N_dyads]:Z ~ normal( 0 , 1 ),
  cholesky_factor_corr[2]:L_Rho_T ~ lkj_corr_cholesky( 2 ),
  sigma_T ~ exponential(1),

  ## dislike dyad effects - non-centered
  transpars> matrix[N_dyads,2]:T2 <-
    compose_noncentered( rep_vector(sigma_T2,2) , L_Rho_T2 , Z2 ),
  matrix[2,N_dyads]:Z2 ~ normal( 0 , 1 ),
  cholesky_factor_corr[2]:L_Rho_T2 ~ lkj_corr_cholesky( 2 ),
  sigma_T2 ~ exponential(1),

  ## compute correlation matrix for dyads
  gq> matrix[2,2]:Rho_T <- Chol_to_Corr( L_Rho_T ),
  gq> matrix[2,2]:Rho_T2 <- Chol_to_Corr( L_Rho_T2 )
)

n_dyads <- nrow(dat)
```

```
monk_dat <- list(
  N_dyads = nrow(dat),
  D = dat$dyad_id,
  likeAB = dat$like_AB,
  likeBA = dat$like_BA,
  dislikeAB = dat$dislike_AB,
  dislikeBA = dat$dislike_BA )

m2 <- ulam( f_dyad2 , data=monk_dat , chains=4 , cores=4 , iter=2000 )
precis( m2 , depth=3 , pars=c("Rho_T","Rho_T2") )
```

	mean	sd	5.5%	94.5%	rhat	ess_bulk
Rho_T[1,1]	1.00	0.00	1.00	1.00	NA	NA
Rho_T[2,1]	0.63	0.10	0.45	0.79	1.00	1153.66
Rho_T[1,2]	0.63	0.10	0.45	0.79	1.00	1153.66
Rho_T[2,2]	1.00	0.00	1.00	1.00	NA	NA
Rho_T2[1,1]	1.00	0.00	1.00	1.00	NA	NA
Rho_T2[2,1]	0.45	0.14	0.21	0.66	1.01	1033.96
Rho_T2[1,2]	0.45	0.14	0.21	0.66	1.01	1033.96
Rho_T2[2,2]	1.00	0.00	1.00	1.00	NA	NA

The dislike correlation is a bit smaller. Let's compute the contrast however:

```
post <- extract.samples(m2)
round(quantile( post$Rho_T[,1,2] - post$Rho_T2[,1,2] ),2)
```

0%	25%	50%	75%	100%
-0.42	0.06	0.17	0.29	0.76

So more than 75% of the posterior mass is positive, indicating that reciprocity for liking is higher than for disliking. But there's a lot of uncertainty in both estimates, so still lots of uncertainty in this contrast.

For fun, let's visualize these networks.

```
T_est <- apply(post$T,2:3,mean)
T2_est <- apply(post$T2,2:3,mean)

N <- 18
n_dyads <- choose(18,2)
dyads <- t(combn(18,2))
# convert to adjacency matrix
y_est <- matrix(NA,18,18)
y2_est <- matrix(NA,18,18)
for ( i in 1:n_dyads ) {
  y_est[ dyads[i,1] , dyads[i,2] ] <- T_est[i,1]
  y_est[ dyads[i,2] , dyads[i,1] ] <- T_est[i,2]
  y2_est[ dyads[i,1] , dyads[i,2] ] <- T2_est[i,1]
```

```

      y2_est[ dyads[i,2] , dyads[i,1] ] <- T2_est[i,2]
    }#i

library(igraph)
# like
sng <- graph_from_adjacency_matrix(y_est)
lx <- layout_nicely(sng)
plot(sng , layout=lx , vertex.size=8 , edge.arrow.size=1 ,
     edge.width=4 , edge.curved=0.35 , vertex.color=2 , edge.color=grau(0.2) ,
     asp=0.9 , margin = -0.05 , vertex.label=NA )

# dislike
sng2 <- graph_from_adjacency_matrix(y2_est)
plot(sng2 , layout=lx , vertex.size=8 , edge.arrow.size=1 ,
     edge.width=4 , edge.curved=0.5 , vertex.color=6 , edge.color=grau(0.2) ,
     asp=0.9 , margin = -0.05 , vertex.label=NA )

```

The networks are on the next page. The blue one is the liking network, arranged so it is clustered by reciprocal liking. The orange one is the disliking network, using the same layout as the liking network, so you can more easily compare. It's clear that the disliking tends to point at other clusters in the network. And a few individuals tend to be nominated a lot.

3. To include generalized receiving of likes and dislikes, we can use the same structure as in the book example, but now both dimensions of the matrix are receiving: the first is likes and the second is dislikes. Having both general receiving effects in the same distribution allows us to estimate the correlation across networks.

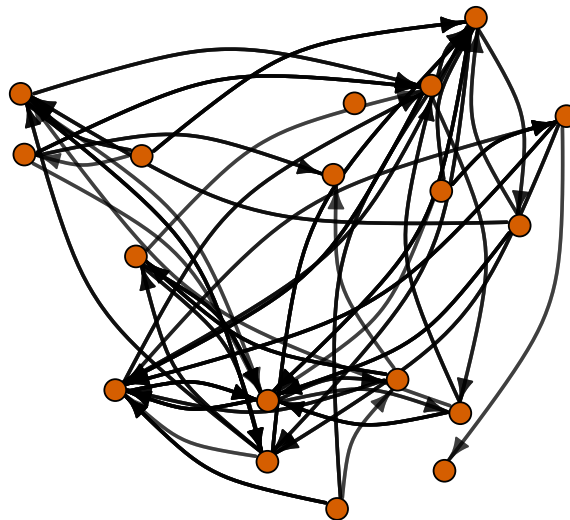
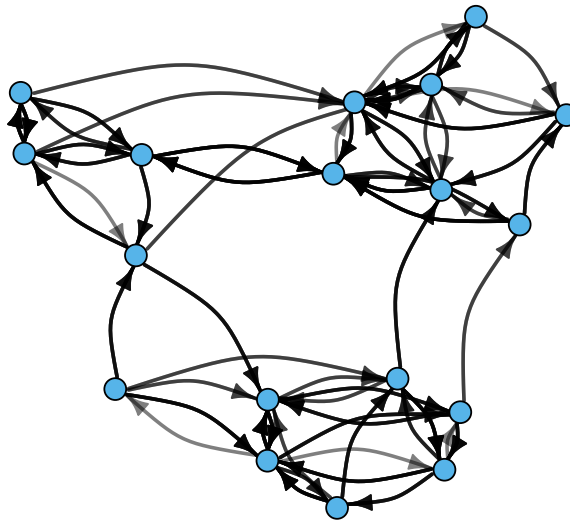
```

# general receiving model
f_dyad3 <- alist(
  likeAB ~ binomial( 3 , pAB ),
  likeBA ~ binomial( 3 , pBA ),
  logit(pAB) <- a[1] + T[D,1] + R[B,1],
  logit(pBA) <- a[1] + T[D,2] + R[A,1],

  dislikeAB ~ binomial( 3 , qAB ),
  dislikeBA ~ binomial( 3 , qBA ),
  logit(qAB) <- a[2] + T2[D,1] + R[B,2],
  logit(qBA) <- a[2] + T2[D,2] + R[A,2],
  vector[2]:a ~ normal(0,1),

  ## like dyad effects - non-centered
  transpars> matrix[N_dyads,2]:T <-
    compose_noncentered( rep_vector(sigma_T,2) , L_Rho_T , Z ),
  matrix[2,N_dyads]:Z ~ normal( 0 , 1 ),
  cholesky_factor_corr[2]:L_Rho_T ~ lkj_corr_cholesky( 2 ),

```



```

sigma_T ~ exponential(1),

## dislike dyad effects - non-centered
transpars> matrix[N_dyads,2]:T2 <-
  compose_noncentered( rep_vector(sigma_T2,2) , L_Rho_T2 , Z2 ),
matrix[2,N_dyads]:Z2 ~ normal( 0 , 1 ),
cholesky_factor_corr[2]:L_Rho_T2 ~ lkj_corr_cholesky( 2 ),
sigma_T2 ~ exponential(1),

## R matrix of receiving effects
transpars> matrix[18,2]:R <-
  compose_noncentered( sigma_R , L_Rho_R , ZR ),
matrix[2,18]:ZR ~ normal( 0 , 1 ),
cholesky_factor_corr[2]:L_Rho_R ~ lkj_corr_cholesky( 2 ),
vector[2]:sigma_R ~ exponential(1),

## compute correlation matrix for dyads
gq> matrix[2,2]:Rho_T <- Chol_to_Corr( L_Rho_T ),
gq> matrix[2,2]:Rho_T2 <- Chol_to_Corr( L_Rho_T2 ),
gq> matrix[2,2]:Rho_R <- Chol_to_Corr( L_Rho_R )
)

n_dyads <- nrow(dat)
monk_dat <- list(
  N_dyads = nrow(dat),
  D = dat$dyad_id,
  likeAB = dat$like_AB,
  likeBA = dat$like_BA,
  dislikeAB = dat$dislike_AB,
  dislikeBA = dat$dislike_BA,
  A = dat$A,
  B = dat$B )
m3 <- ulam( f_dyad3 , data=monk_dat , chains=4 , cores=4 , iter=2000 )
precis( m3 , depth=3 , pars=c("Rho_T","Rho_T2","sigma_R","Rho_R") )

```

	mean	sd	5.5%	94.5%	rhat	ess_bulk
Rho_T[1,1]	1.00	0.00	1.00	1.00	NA	NA
Rho_T[2,1]	0.65	0.11	0.47	0.81	1.00	1286.53
Rho_T[1,2]	0.65	0.11	0.47	0.81	1.00	1286.53
Rho_T[2,2]	1.00	0.00	1.00	1.00	NA	NA
Rho_T2[1,1]	1.00	0.00	1.00	1.00	NA	NA
Rho_T2[2,1]	0.52	0.16	0.25	0.76	1.00	1351.83
Rho_T2[1,2]	0.52	0.16	0.25	0.76	1.00	1351.83
Rho_T2[2,2]	1.00	0.00	1.00	1.00	NA	NA
sigma_R[1]	0.71	0.52	0.07	1.63	1.00	659.29
sigma_R[2]	2.47	0.86	1.40	4.01	1.00	1468.35

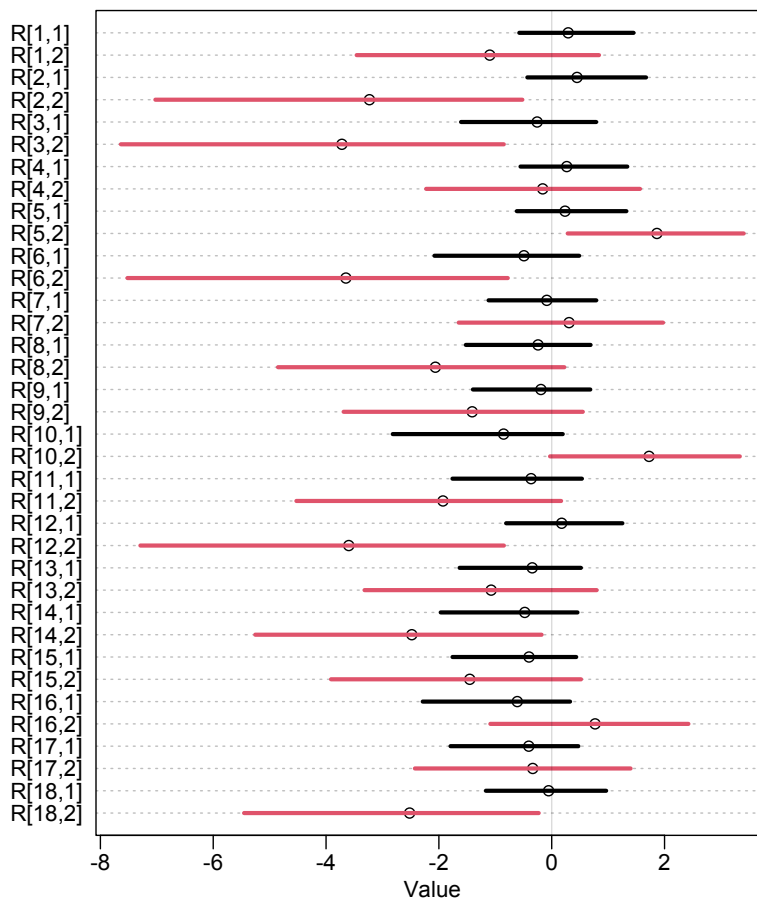
```

Rho_R[1,1]  1.00 0.00  1.00  1.00  NA      NA
Rho_R[2,1]  0.02 0.39 -0.64  0.64 1.01  627.18
Rho_R[1,2]  0.02 0.39 -0.64  0.64 1.01  627.18
Rho_R[2,2]  1.00 0.00  1.00  1.00  NA      NA

```

First note that the dyad reciprocity for disliking has increased a little, as a result of modeling the generalized disliking. And the standard deviation that corresponds to general disliking,  $\sigma_R[2]$ , is pretty large. This is consistent with some monks getting a lot dislike nominations than others. The likes network doesn't seem to show the same phenomenon. Things are more balanced there. Finally, there doesn't seem to be any evidence of a correlation in these general effects across the networks:  $Rho_R[1,2]$  is nothing much.

We can see the different patterns in general likes/dislikes by plotting the posterior distributions for the R matrix of effects. It looks like this:



Monks are listed from top to bottom (the first index), and the red intervals are the dislike elements. A few monks, 5 and 10 and 16, have large values. The rest are pretty much ignored. The black intervals, the liking estimates, are all pretty close to zero.