

Homework - Week 8

Caio Geraldles

The data in `week08_Monks.csv` (found on the course website) are “like” and “dislike” nominations by 18 monks living in the same monastery over three time periods. Therefore the observed variables are counts from 0 to 3 of times monk A nominated monk B as liked or disliked.¹ Each row in the data is a pair of monks (a dyad). The variables are:

- A: Index number for first monk in dyad
- B: Index number for second monk in dyad
- like_AB: Number of times A nominated B as liked
- like_BA: Number of times B nominated A as liked
- dislike_AB: Number of times A nominated B as disliked
- dislike_BA: Number of times B nominated A as disliked

```
monks <- read_csv("./week08_Monks.csv", show_col_types = FALSE)

dat <- list(
  N = nrow(monks),
  D = monks$dyad_id,
  mA = monks$A,
  mB = monks$B,
  LAB = monks$like_AB,
  LBA = monks$like_BA,
  DAB = monks$dislike_AB,
  DBA = monks$dislike_BA
)
```

Question 1

Use these data to estimate the amount of reciprocity in “like” nominations within dyads. You can ignore the “dislike” data for now. Use the social network example from the book to help, but you should modify it appropriately.

Answer

We define a model where the number of like votes from A to B, L_{AB} is a binomial distribution of size 3 and probability $p_{L_{AB}}$, with symmetric values indexed for the direction $B \rightarrow A$. The logit of a p_{L_j} is the sum of an intercept α with an effect of the positive ties T^+ for $A \rightarrow B$, correlated to its counterpart for $B \rightarrow A$. The priors are fairly flat.

$$\begin{aligned}
L_{AB} &\sim \text{Binomial}(3, p_{L_{AB}}) \\
L_{BA} &\sim \text{Binomial}(3, p_{L_{BA}}) \\
\text{logit}(p_{L_{AB}}) &= \alpha + T_{AB_i}^+ \\
\text{logit}(p_{L_{BA}}) &= \alpha + T_{BA_i}^+ \\
\alpha &\sim \text{Normal}(0, 1) \\
\begin{pmatrix} T_{AB}^+ \\ T_{BA}^+ \end{pmatrix} &\sim \text{MVNormal} \left(\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} \sigma_{T^+}^2 & \rho_{T^+} \sigma_{T^+}^2 \\ \rho_{T^+} \sigma_{T^+}^2 & \sigma_{T^+}^2 \end{pmatrix} \right) \\
\sigma_{T^+} &\sim \text{Exponential}(1) \\
\rho_{T^+} &\sim \text{LKJCorr}(2)
\end{aligned}$$

```

f.m1 <- alist(
  LAB ~ binomial(3, pLAB),
  LBA ~ binomial(3, pLBA),

  logit(pLAB) <- a + pT[D,1],
  logit(pLBA) <- a + pT[D,2],

  a ~ normal(0,1),

  transpars> matrix[N,2]:pT <-
    compose_noncentered(rep_vector(sigma_pT, 2), L_Rho_pT, z),
  matrix[2,N]:z ~ normal(0,1),
  cholesky_factor_corr[2]:L_Rho_pT ~ lkj_corr_cholesky(2),
  sigma_pT ~ exponential(1),

  gq> matrix[2,2]:Rho_pT <- Chol_to_Corr(L_Rho_pT)
)

m1 <- ulam(
  flist = f.m1,
  data = dat,
  chains=4,
  cores=4,
  iter=2000,
  log_lik=TRUE,
  file="./models/m1")

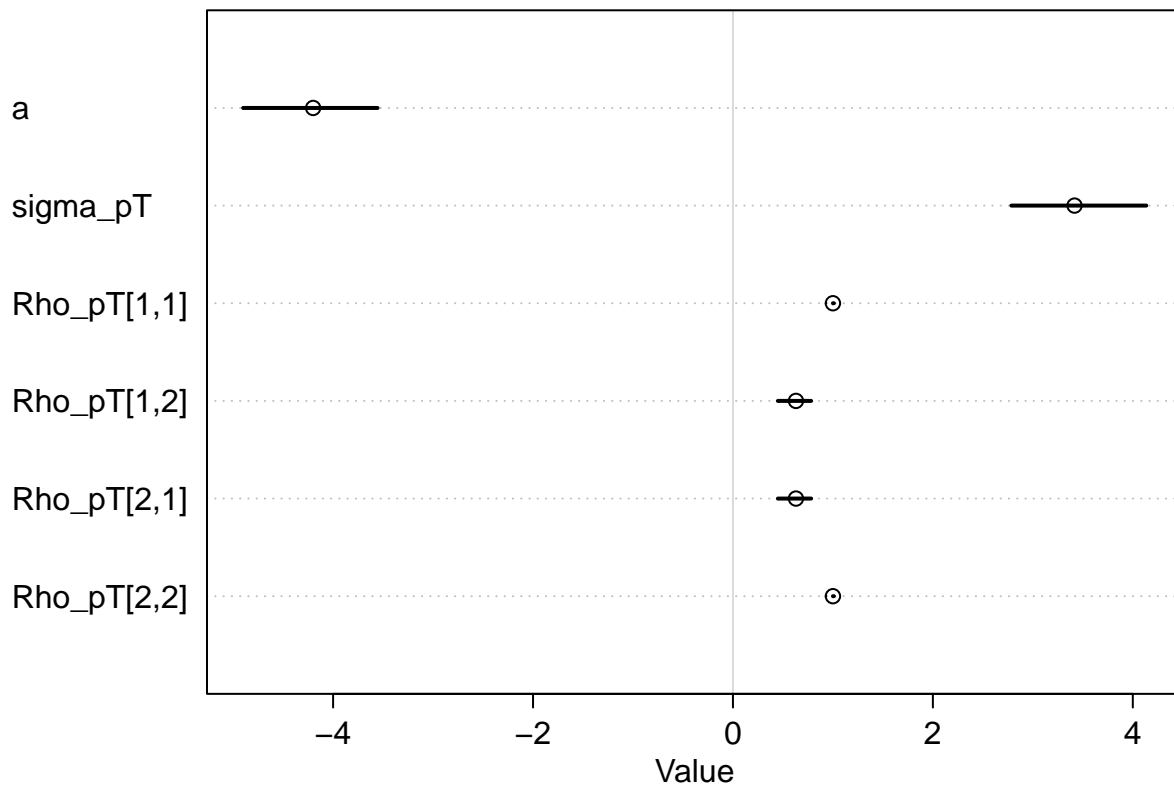
```

```

precis(m1, pars=c("a", "sigma_pT", "Rho_pT"), depth=3)

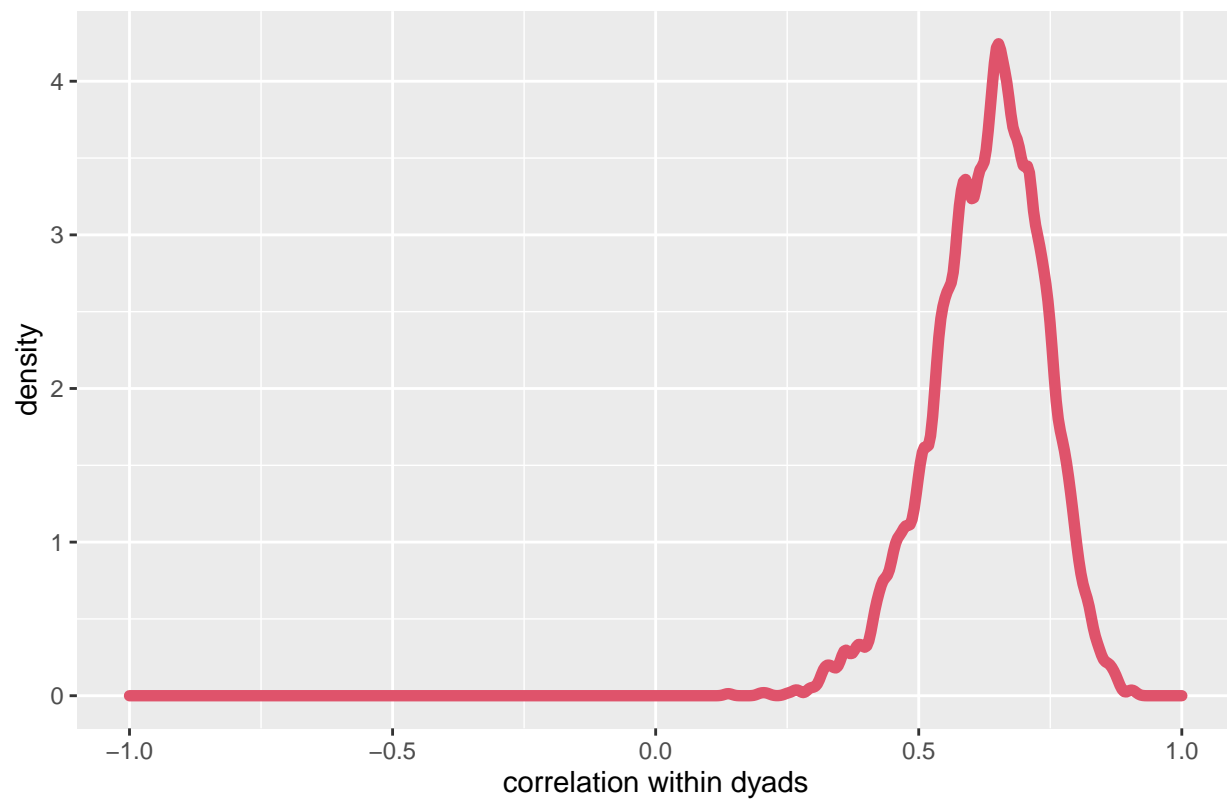
```

##		mean	sd	5.5%	94.5%	n_eff	Rhat4
##	a	-4.1998182	0.4222707	-4.9002953	-3.5552142	2724.439	1.000317
##	sigma_pT	3.4172822	0.4293001	2.7851249	4.1354549	1748.562	1.001322
##	Rho_pT[1,1]	1.0000000	0.0000000	1.0000000	1.0000000	NaN	NaN
##	Rho_pT[1,2]	0.6304678	0.1047167	0.4491242	0.7826848	1206.704	1.001078
##	Rho_pT[2,1]	0.6304678	0.1047167	0.4491242	0.7826848	1206.704	1.001078
##	Rho_pT[2,2]	1.0000000	0.0000000	1.0000000	1.0000000	NaN	NaN



```
post.m1 <- extract.samples(m1)
rho_pt.df <- tibble(x=post.m1$Rho_pT[,1,2])
```

Reciprocity between dyads of monks (likes)



Question 2

Now also analyse the “dislike” nominations. Estimate the amount of reciprocity in the “dislike” nominations and compare to the estimate from the “like” nominations. Does “like” or “dislike” have higher reciprocity? Be sure to compute the posterior contrast between the two estimates.

Answer

We define a model where the number of dislike votes from A to B, D_{AB} is a binomial distribution of size 3 and probability $p_{D_{AB}}$, with symmetric values indexed for the direction $B \rightarrow A$. The logit of a p_{D_j} is the sum of an intercept α with an effect of the negative ties T^- for $A \rightarrow B$, correlated to its counterpart for $B \rightarrow A$. The priors are fairly flat.

$$\begin{aligned}
 D_{AB} &\sim \text{Binomial}(3, p_{D_{AB}}) \\
 D_{BA} &\sim \text{Binomial}(3, p_{D_{BA}}) \\
 \text{logit}(p_{D_{AB}}) &= \alpha + T_{AB_i}^- \\
 \text{logit}(p_{D_{BA}}) &= \alpha + T_{BA_i}^- \\
 \alpha &\sim \text{Normal}(0, 1) \\
 \begin{pmatrix} T_{AB}^- \\ T_{BA}^- \end{pmatrix} &\sim \text{MVNormal} \left(\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} \sigma_{T^-}^2 & \rho_{T^-} \sigma_{T^-}^2 \\ \rho_{T^-} \sigma_{T^-}^2 & \sigma_{T^-}^2 \end{pmatrix} \right) \\
 \sigma_{T^-} &\sim \text{Exponential}(1) \\
 \rho_{T^-} &\sim \text{LKJCorr}(2)
 \end{aligned}$$

```
f.m2 <- alist(
  DAB ~ binomial(3, pDAB),
  DBA ~ binomial(3, pDBA),

  logit(pDAB) <- a + nT[D,1],
  logit(pDBA) <- a + nT[D,2],

  a ~ normal(0,1),

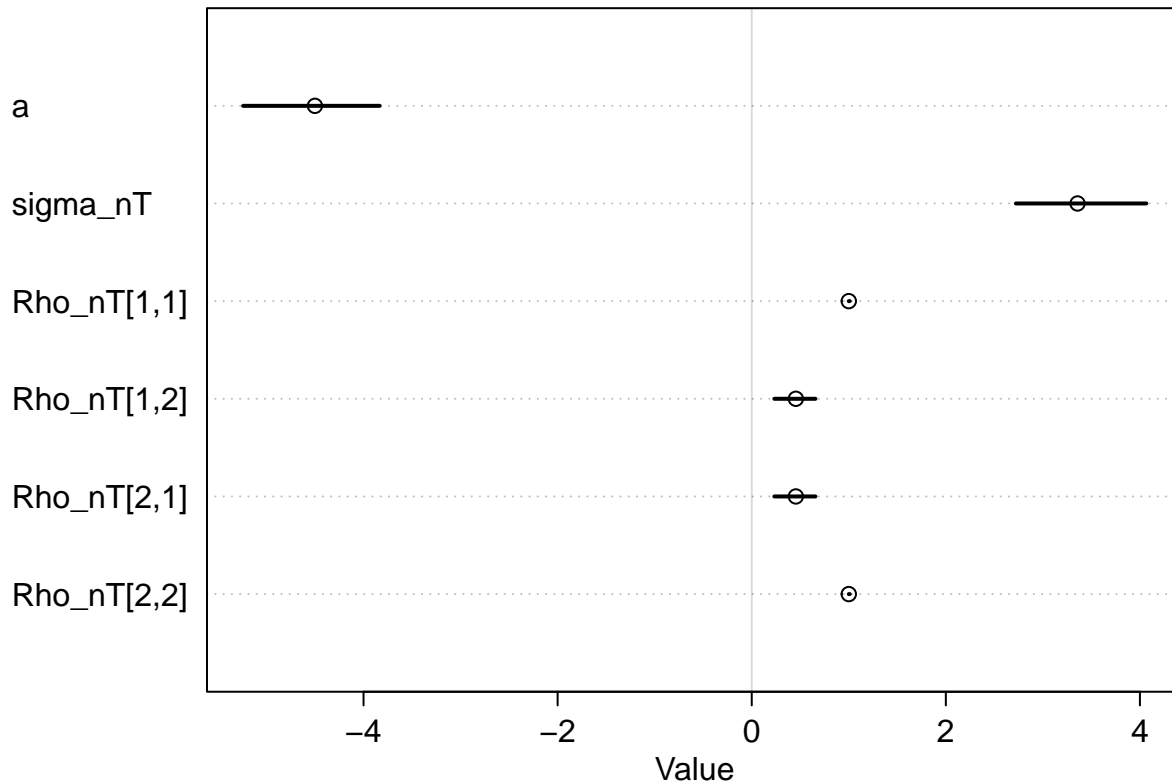
  transpars> matrix[N,2]:nT <-
    compose_noncentered(rep_vector(sigma_nT, 2), L_Rho_nT, z),
  matrix[2,N]:z ~ normal(0,1),
  cholesky_factor_corr[2]:L_Rho_nT ~ lkj_corr_cholesky(2),
  sigma_nT ~ exponential(1),

  gq> matrix[2,2]:Rho_nT <-<- Chol_to_Corr(L_Rho_nT)
)

m2 <- ulam(
  flist = f.m2,
  data = dat,
  chains=4,
  cores=4,
  iter=2000,
  log_lik=TRUE,
  file="./models/m2")

precis(m2, pars=c("a", "sigma_nT", "Rho_nT"), depth=3)
```

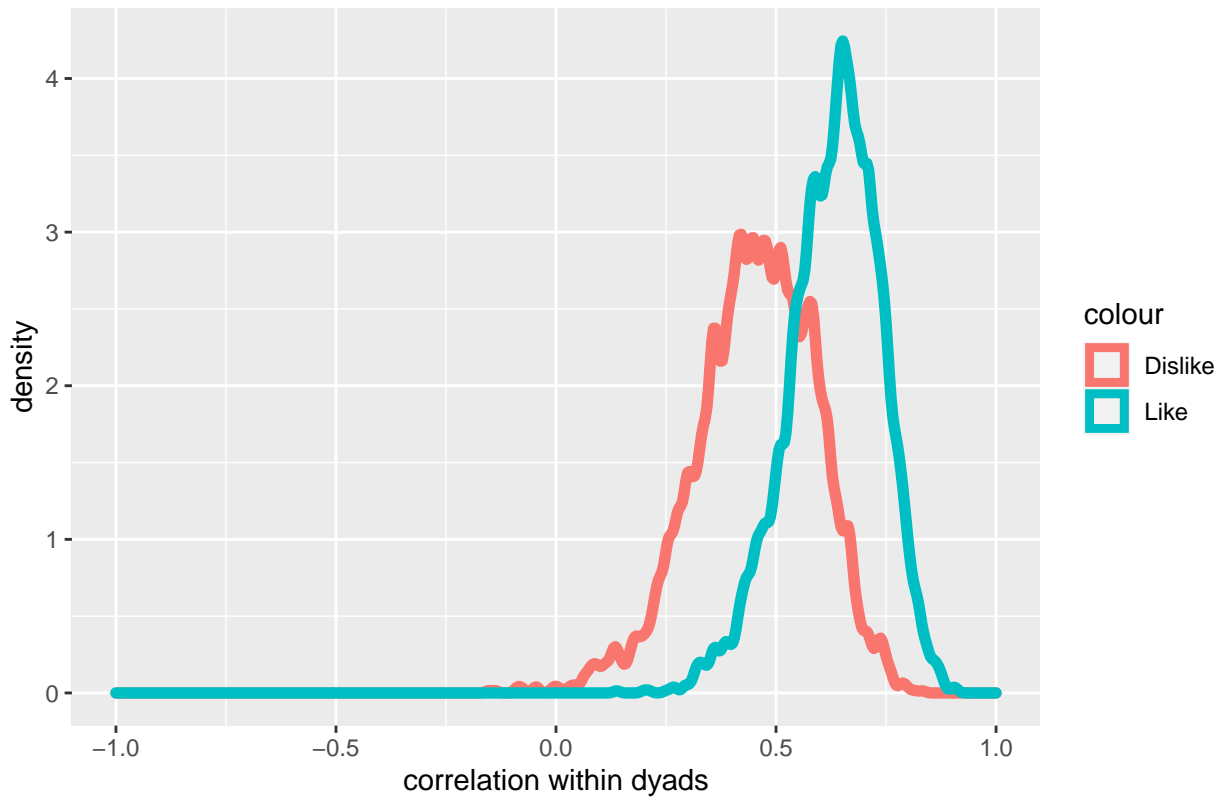
##	mean	sd	5.5%	94.5%	n_eff	Rhat4
## a	-4.5010766	0.4445605	-5.2409298	-3.8334784	2211.830	0.9997561
## sigma_nT	3.3571202	0.4281518	2.7214834	4.0659510	1526.274	1.0022049
## Rho_nT[1,1]	1.0000000	0.0000000	1.0000000	1.0000000	NaN	NaN
## Rho_nT[1,2]	0.4544655	0.1356955	0.2328227	0.6566736	1072.802	1.0011560
## Rho_nT[2,1]	0.4544655	0.1356955	0.2328227	0.6566736	1072.802	1.0011560
## Rho_nT[2,2]	1.0000000	0.0000000	1.0000000	1.0000000	NaN	NaN



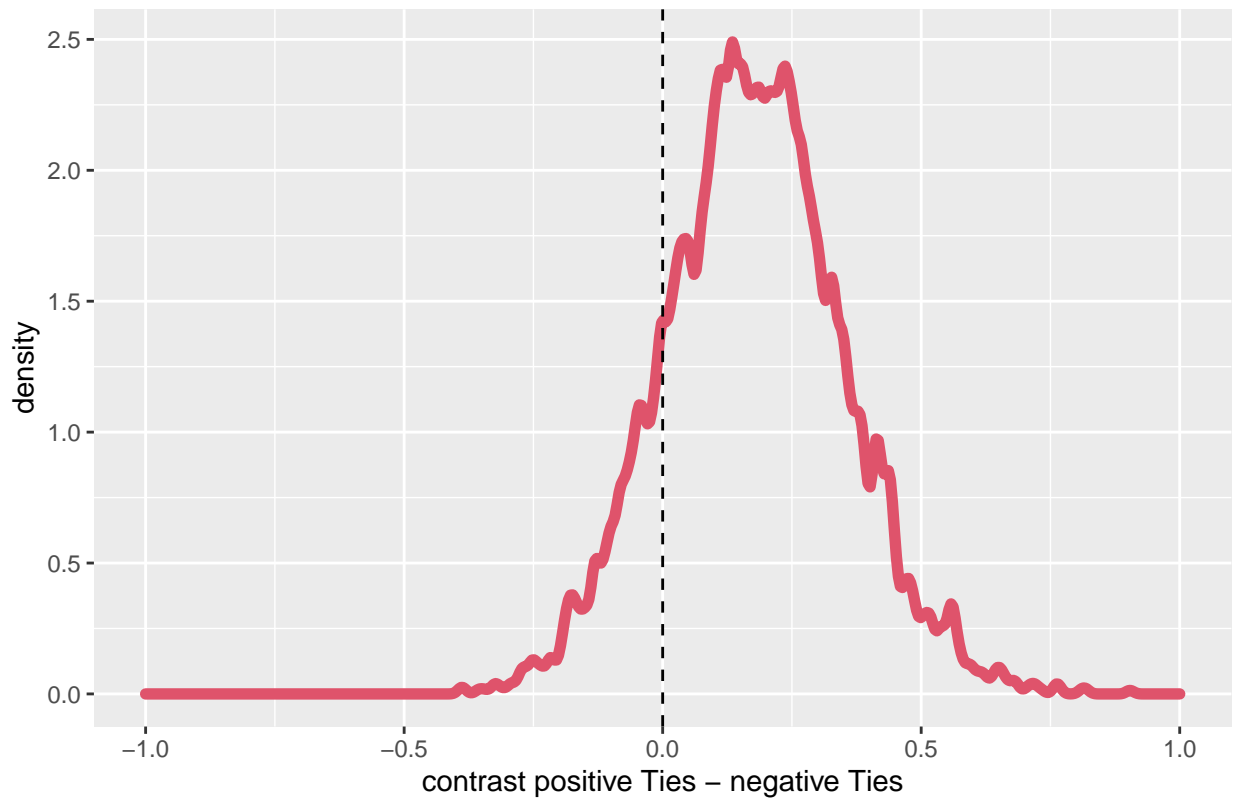
In this model, the negative ties between any dyad of monks has, on average, lower reciprocity than the positive ties in the same dyad. But notice that the overlap is not negligible.

```
post.m2 <- extract.samples(m2)
rho_nt.df <- tibble(x=post.m2$Rho_nT[,1,2])
diff.df <- tibble(x=rho_pt.df$x - rho_nt.df$x)
```

Reciprocity between dyads of monks (like, dislike)



Contrast of reciprocity



Question 3

Now consider generalized liking and disliking. Add generalized “receiving” parameters to the model, analogous to receiving varying effects from the sharing example in the book/lecture. (Each monk only named 3 likes and 3 dislikes each time, so the amount of “giving” is fixed by design in these data.) Can you identify any individuals who are particularly liked/disliked, independent of the dyadic relations?

Answer

$$\begin{aligned}
 L_{AB} &\sim \text{Binomial}(3, p_{L_{AB}}) \\
 L_{BA} &\sim \text{Binomial}(3, p_{L_{BA}}) \\
 D_{AB} &\sim \text{Binomial}(3, p_{D_{AB}}) \\
 D_{BA} &\sim \text{Binomial}(3, p_{D_{BA}}) \\
 \text{logit}(p_{L_{AB}}) &= \alpha_L + T_{AB_i}^+ + R_{LB_i} \\
 \text{logit}(p_{L_{BA}}) &= \alpha_L + T_{BA_i}^+ + R_{LA_i} \\
 \text{logit}(p_{D_{AB}}) &= \alpha + T_{AB_i}^- + R_{DB_i} \\
 \text{logit}(p_{D_{BA}}) &= \alpha + T_{BA_i}^- + R_{DA_i} \\
 \alpha_j &\sim \text{Normal}(0, 1) \\
 \begin{pmatrix} T_{AB}^+ \\ T_{BA}^+ \end{pmatrix} &\sim \text{MVNormal} \left(\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \mathbf{S}_{T^+}, \mathbf{R}_{T^+} \right) \\
 \begin{pmatrix} T_{AB}^- \\ T_{BA}^- \end{pmatrix} &\sim \text{MVNormal} \left(\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \mathbf{S}_{T^-}, \mathbf{R}_{T^-} \right) \\
 \begin{pmatrix} R_{Lj} \\ R_{Dj} \end{pmatrix} &\sim \text{MVNormal} \left(\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \mathbf{S}_{LD}, \mathbf{R}_{LD} \right) \\
 \mathbf{S}_{T^+}, \mathbf{S}_{T^-}, \mathbf{S}_{LD} &\sim \text{Exponential}(1) \\
 \mathbf{R}_{T^+}, \mathbf{R}_{T^-}, \mathbf{R}_{LD} &\sim \text{LKJCorr}(2)
 \end{aligned}$$

```
f.m6 <- alist(
  LAB ~ binomial(3, pLAB),
  LBA ~ binomial(3, pLBA),
  DAB ~ binomial(3, pDAB),
  DBA ~ binomial(3, pDBA),

  logit(pLAB) <- a[1] + pT[D,1] + rlrld[mB,1],
  logit(pLBA) <- a[1] + pT[D,2] + rlrld[mA,1],
  logit(pDAB) <- a[2] + nT[D,1] + rlrld[mB,2],
  logit(pDBA) <- a[2] + nT[D,2] + rlrld[mA,2],

  vector[2]:a ~ normal(0, 1),

  transpars>matrix[18,2]:rlrld <-
    compose_noncentered(rep_vector(sigma_rlrld, 2), L_Rho_rlrld, zR),
  matrix[2,18]:zR ~ normal(0,1),
  cholesky_factor_corr[2]:L_Rho_rlrld ~ lkj_corr_cholesky(2),
  sigma_rlrld ~ exponential(1),

  transpars> matrix[N,2]:pT <-
    compose_noncentered(rep_vector(sigma_pT, 2), L_Rho_pT, zP),
  matrix[2,N]:zP ~ normal(0,1),
  cholesky_factor_corr[2]:L_Rho_pT ~ lkj_corr_cholesky(2),
```

```

sigma_pT ~ exponential(1),

transpars> matrix[N,2]:nT <-
  compose_noncentered(rep_vector(sigma_nT, 2), L_Rho_nT, zN),
matrix[2,N]:zN ~ normal(0,1),
cholesky_factor_corr[2]:L_Rho_nT ~ lkj_corr_cholesky(2),
sigma_nT ~ exponential(1),

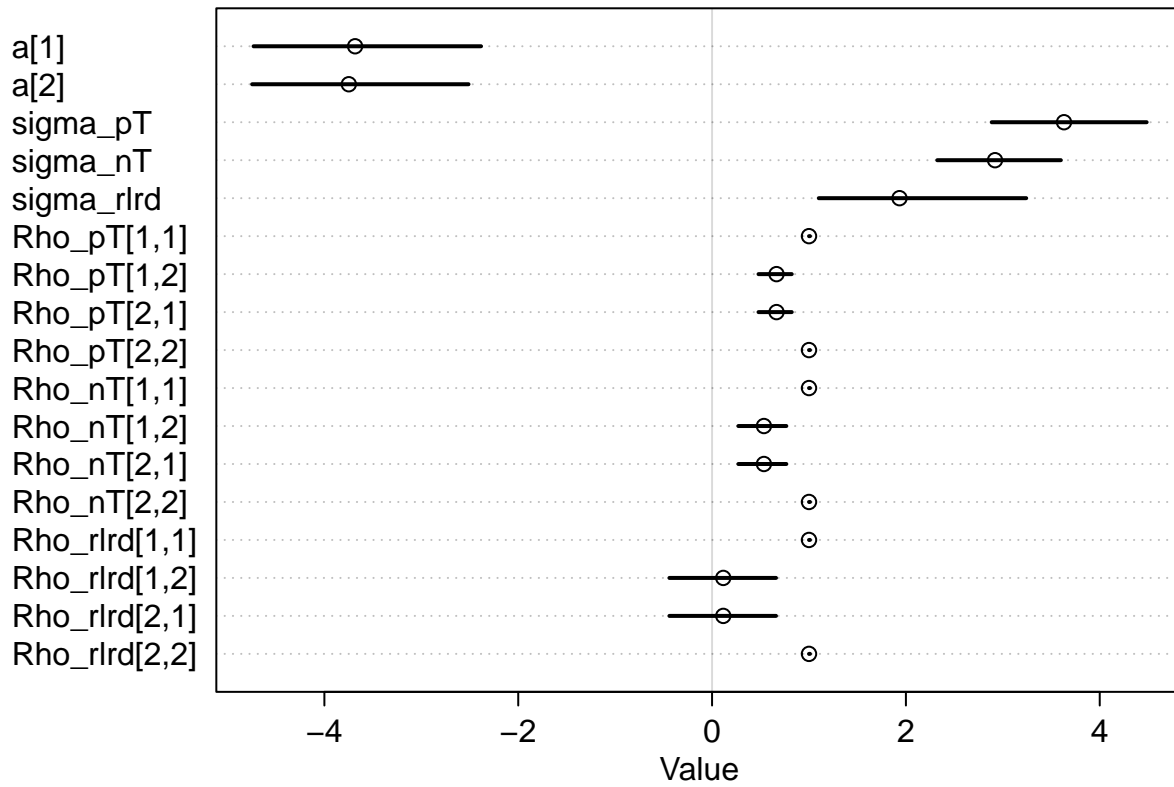
gq> matrix[2,2]:Rho_pT <- Chol_to_Corr(L_Rho_pT),
gq> matrix[2,2]:Rho_nT <- Chol_to_Corr(L_Rho_nT),
gq> matrix[2,2]:Rho_rld <- Chol_to_Corr(L_Rho_rld)
)

m6 <- ulam(flist = f.m6,
  data=dat,
  cores=4,
  chains=4,
  iter=2000,
  log_lik=TRUE,
  file="./models/m6")

precis(m6, pars=c("a", "sigma_pT", "sigma_nT", "sigma_rld",
  "Rho_pT", "Rho_nT", "Rho_rld"), depth=3)

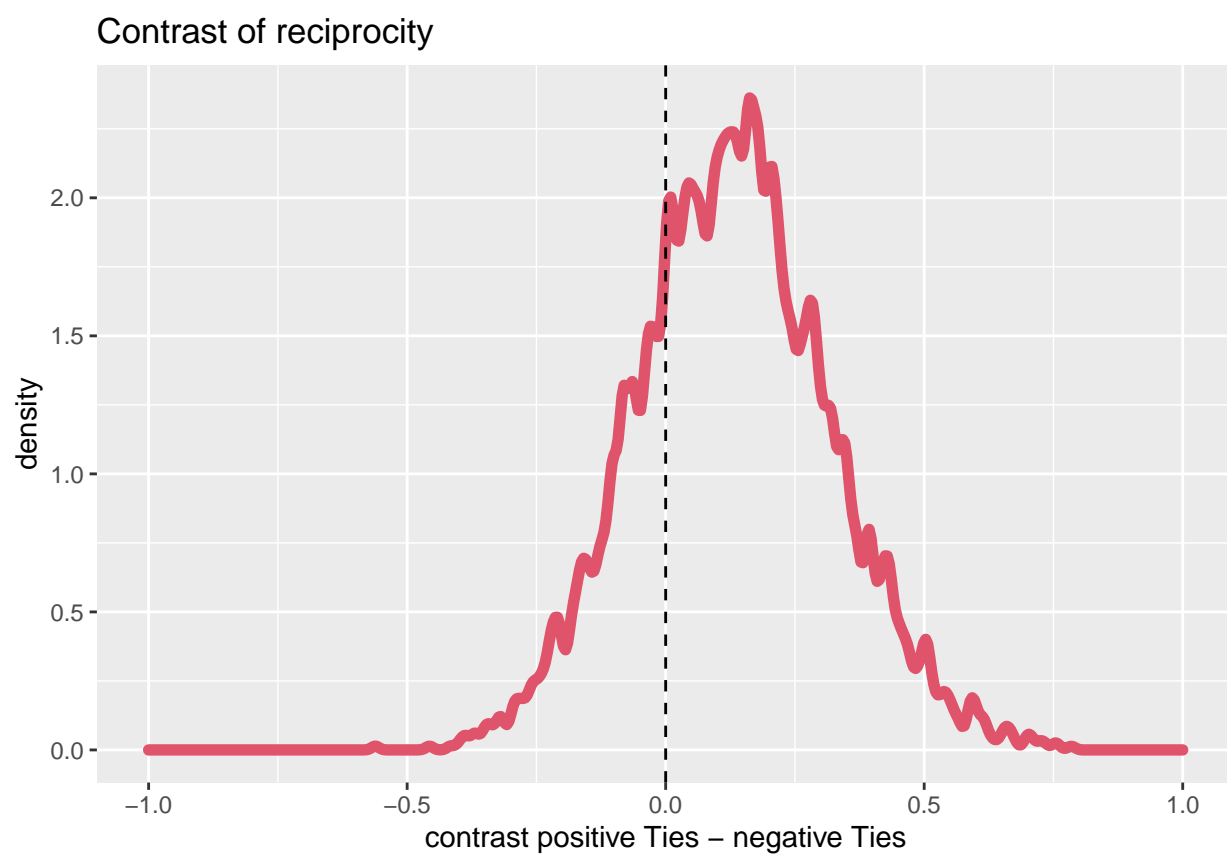
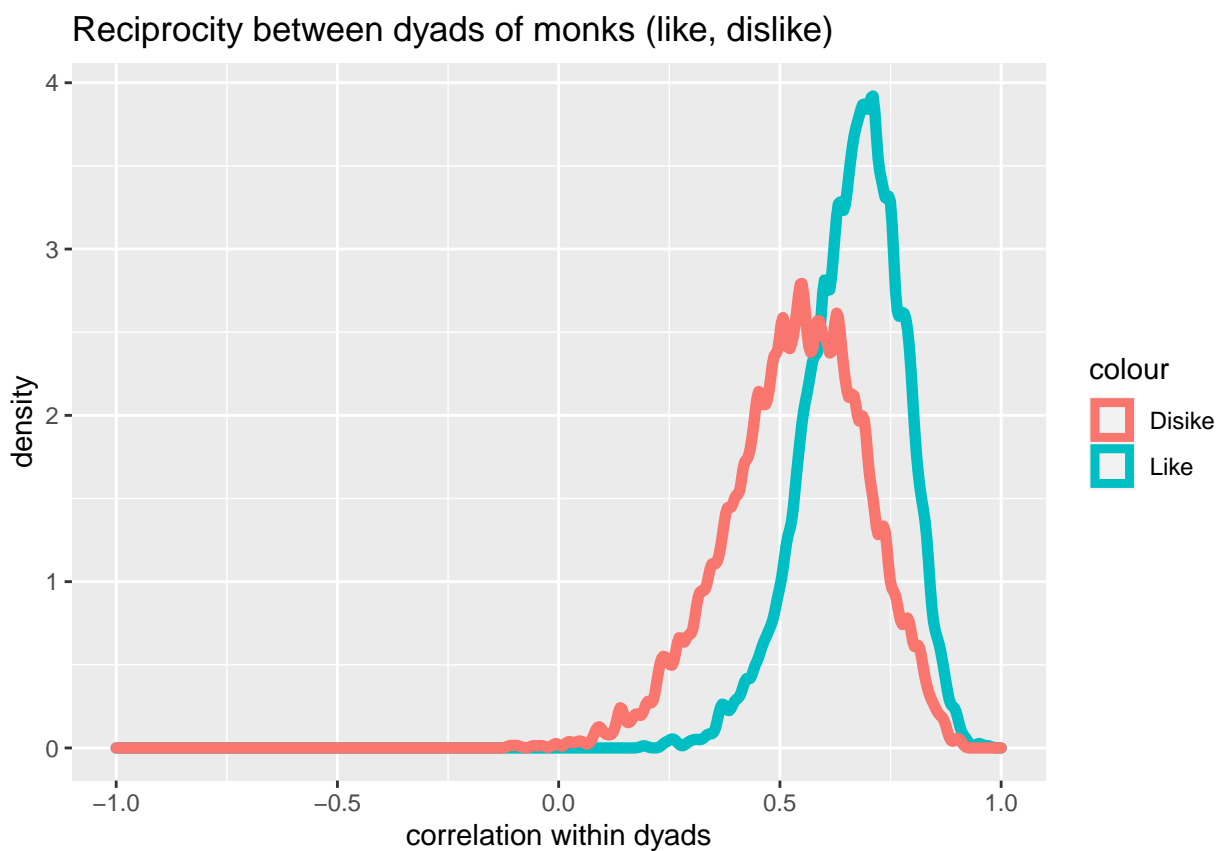
```

##		mean	sd	5.5%	94.5%	n_eff	Rhat4
##	a[1]	-3.6831612	0.7319731	-4.7309827	-2.3836368	1043.3204	0.9996800
##	a[2]	-3.7481130	0.7097196	-4.7458697	-2.5128470	1333.0989	0.9997903
##	sigma_pT	3.6314505	0.5081121	2.8849677	4.4842843	1623.1561	1.0013524
##	sigma_nT	2.9196465	0.4117690	2.3236229	3.5991569	1485.5946	1.0016727
##	sigma_rld	1.9339893	0.7000080	1.1007613	3.2416830	689.8158	1.0011356
##	Rho_pT[1,1]	1.0000000	0.0000000	1.0000000	1.0000000	NaN	NaN
##	Rho_pT[1,2]	0.6650927	0.1083009	0.4788825	0.8223094	1613.8831	0.9994749
##	Rho_pT[2,1]	0.6650927	0.1083009	0.4788825	0.8223094	1613.8831	0.9994749
##	Rho_pT[2,2]	1.0000000	0.0000000	1.0000000	1.0000000	NaN	NaN
##	Rho_nT[1,1]	1.0000000	0.0000000	1.0000000	1.0000000	NaN	NaN
##	Rho_nT[1,2]	0.5350608	0.1542301	0.2710096	0.7665078	1342.0602	1.0010733
##	Rho_nT[2,1]	0.5350608	0.1542301	0.2710096	0.7665078	1342.0602	1.0010733
##	Rho_nT[2,2]	1.0000000	0.0000000	1.0000000	1.0000000	NaN	NaN
##	Rho_rld[1,1]	1.0000000	0.0000000	1.0000000	1.0000000	NaN	NaN
##	Rho_rld[1,2]	0.1153858	0.3440382	-0.4406062	0.6601409	791.3867	1.0005227
##	Rho_rld[2,1]	0.1153858	0.3440382	-0.4406062	0.6601409	791.3867	1.0005227
##	Rho_rld[2,2]	1.0000000	0.0000000	1.0000000	1.0000000	NaN	NaN

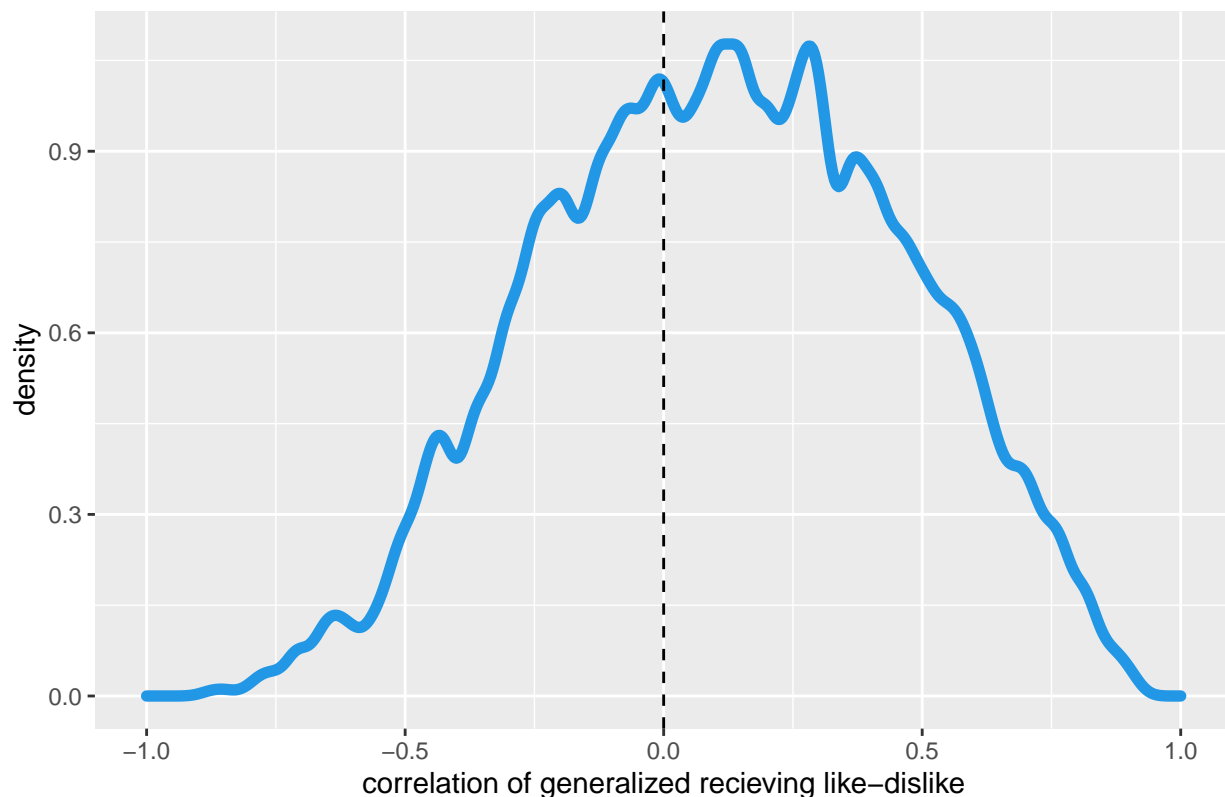


The average reciprocity of negative and positive ties has remained fairly similar to the previous model.

```
post.m6 <- extract.samples(m6)
m6.df <- tibble(pt=post.m6$Rho_pT[,1,2],
                 nt=post.m6$Rho_nT[,1,2],
                 rld=post.m6$Rho_rld[,1,2]) %>%
  mutate(diff = pt-nt)
```



Generalize receiving like–dislike



There are some monks whose dislike-like ratio seems to be very far from average, and the model is very uncertain about how far they actually are. Peter, Greg, Victor and Basil have highly above average dislike odds, and although the model is uncertain of how much disliked they actually are, they certainly fall above the like=dislike line. Boni and Bonaven are quite liked according to the model, but the model is much less certain if they are actually unevenly liked or disliked: the model is confident that they are not hated at all costs, but is not so sure about their charisma.

```
monknames <- (monks %>%
  select(B, B_name) %>%
  distinct() %>%
  add_row(B=1, B_name="ROMUL") %>%
  arrange(B))$B_name

l <- sapply(1:18, function(i) post.m6$a[,1] + post.m6$r1rd[,i,1])
d <- sapply(1:18, function(i) post.m6$a[,2] + post.m6$r1rd[,i,2])
l_mu <- apply(inv_logit(l), 2, mean)
d_mu <- apply(inv_logit(d), 2, mean)

ld.df <- tibble(monk=monknames, like=l_mu, dislike=d_mu)
```

