

Distinguishable and Exchangeable Dyads: Bayesian Multilevel Modelling

Cross-Sectional and Intensive Longitudinal APIM and DIM

Pascal Küng

2025-11-02

This tutorial presents applied Bayesian multilevel modelling for dyadic data, covering both distinguishable and exchangeable dyads. We develop cross-sectional and intensive longitudinal variants of the Actor–Partner Interdependence Model (APIM) and the Dyad–Individual Model (DIM), demonstrate their algebraic equivalence, and discuss implications for interpretation. Using simulated data, we provide fully reproducible code to rebuild the models and assess parameter recovery. The tutorial offers comprehensive `{brms}` implementations, including random-effects structures, heterogeneous residual variances, compound-symmetry residuals, and optional AR(1) processes. We illustrate diagnostics and model comparison (e.g., posterior predictive checks and LOO), and show how to extract dyad-level variance–covariance components for exchangeable-dyad models. All code is open and includes data simulation routines that can be run to test parameter retrieval. The material serves as a hands-on guide for analyzing dyadic processes in both cross-sectional and longitudinal settings.

Interactive HTML Version (recommended):

This tutorial is best viewed as interactive slides here:

<https://pascal-kueng.github.io/05DyadicDataAnalysis/DyadicDataAnalysis.html>

The PDF version below is a static archival version for indexing, citation, and long-term reference.

0.1 Overview

- Distinguishable Dyads
 - Cross-Sectional APIM
 - Longitudinal APIM
- Exchangeable Dyads
 - Cross-Sectional APIM
 - Cross-Sectional DIM
 - Equivalence between APIM and DIM
 - Longitudinal DIM
 - Longitudinal APIM

0.2 Loading Libraries and setting CmdStan Backend

Some functions used in this presentation are sourced from the files available in the folder 00_R_Functions.

```
library(tidyverse)
library(brms)
library(bmlm)
library(easystats)
library(DiagrammeR)
library(DHARMA)
library(MASS)
library(purrr)
source(file.path('00_R_Functions', 'ReportModels.R'))
source(file.path('00_R_Functions', 'PrettyTables.R'))
source(file.path('00_R_Functions', 'PrepareData.R'))

# table font size
font_size <- 9

options(
  brms.backend = 'cmdstan',
  brms.file_refit = 'on_change'
)
```

1 Distinguishable Dyads

1.1 Distinguishable Dyads - Cross-Sectional APIM

1.2 Distinguishable Dyads - Data: Simulated Dyads

```
print_df(head(df),
  font_size = font_size)
```

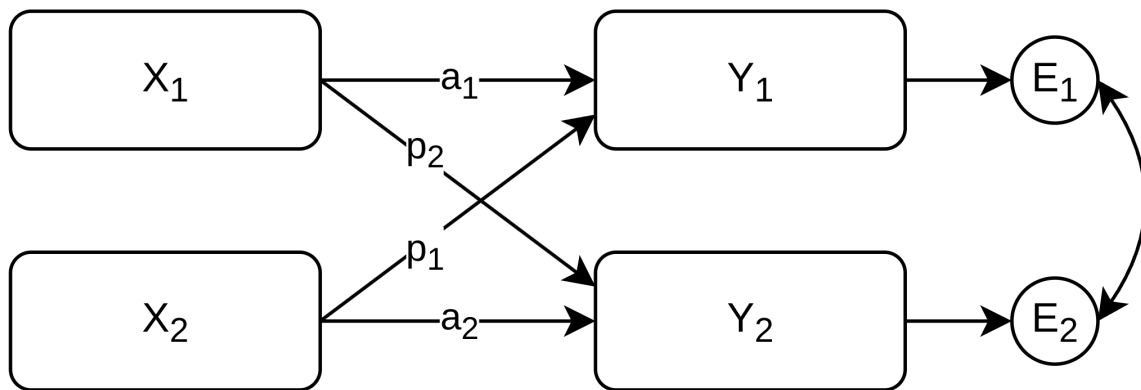
Attaching package: 'kableExtra'

The following object is masked from 'package:dplyr':

group_rows

userID	coupleID	gender	communication	satisfaction
1_1	1	1	4.851107	4.841332
1_2	1	2	6.004533	5.577165
2_1	2	1	5.881321	7.248741
2_2	2	2	6.433723	6.960293
3_1	3	1	4.283971	6.928699
3_2	3	2	5.516060	6.077688

1.3 Distinguishable Dyads - Cross-Sectional APIM



(e.g., Kenny et al., 2006; Kenny & Cook, 1999; Kenny & Kashy, 2011)

1.4 Distinguishable Dyads - Preparing Data

```
df_apim <- df %>%
  # Reshaping to Actor-Partner format (4-field)
  reshape_dyadic_data(
    person_id = 'userID',
    dyad_id = 'coupleID',
    vars_to_reshape = 'communication'
  ) %>%
  mutate(
    # Optional: grand-mean centering
    communication_actor_gmc = communication_actor - mean(communication_actor),
    communication_partner_gmc = communication_partner -
      ↪ mean(communication_partner),

    # Create Dummy-Variables for male and female
    is_female = ifelse(gender == 1, 1, 0),
    is_male = ifelse(gender == 2, 1, 0),
  )

print_df(head(df_apim),
  font_size = font_size)
```

1.5 Distinguishable Dyads - Preparing Data

userID	coupleID	gender	is_male	is_female	communication	satisfaction	communication_actor	communication_partner	communication_actor_gmc	communication_partner_gmc
1_1	1	1	0	1	4.851107	4.841332	4.851107	6.004533	-0.1553604	0.0000000
1_2	1	2	1	0	6.004533	5.577165	6.004533	4.851107	0.9980657	0.0000000
2_1	2	1	0	1	5.881321	7.248741	5.881321	6.433723	0.8748537	0.0000000
2_2	2	2	1	0	6.433723	6.960293	6.433723	5.881321	1.4272563	0.0000000
3_1	3	1	0	1	4.283971	6.928699	4.283971	5.516060	-0.7224960	0.0000000
3_2	3	2	1	0	5.516060	6.077688	5.516060	4.283971	0.5095933	0.0000000

1.6 Distinguishable Dyads - Fitting the Model in BRMS

Model non-independence either via a shared random intercept or correlated residuals (not both). Posterior predictive checks, residual diagnostics and leave one out cross validation (e.g., via `loo_compare()`) may help inform which one to use.

For other families like ordinal regression with brms, one might work better than the other.

1.7 Distinguishable Dyads - Fitting the Model in BRMS

```
1 # Option A: Random-intercept model for non-independence
2
3 formula <- bf(
4   satisfaction ~
5
6   # Remove global intercept, introduce male and female intercepts.
7   0 + is_male + is_female +
8
9   # Actor effect for male and female
10  communication_actor_gmc:is_male +
11  communication_actor_gmc:is_female +
12
13  # Partner effect for male and female
14  communication_partner_gmc:is_male +
15  communication_partner_gmc:is_female +
16
17  # Option 0: Random intercept for male and female (correlated)
18  # NOT identifiable in the cross-sectional case:
19  # (0 + is_male + is_female | coupleID)
20
21  # Option A: Shared dyad effect - interpretable dyad-level random
22  ↪ intercept.
23  (1 | coupleID),
24
25  # Two distinct residual variances for males vs females:
26  sigma = ~ 0 + is_male + is_female
27 )
28
29 priors <- c(
30   # prior(normal(2, 3), class = "Intercept"),
31   prior(normal(0, 5), class = "b"),
32   prior(student_t(3, 0, 1.5), class = "b", dpar = "sigma"),
33   prior(student_t(3, 0, 2.5), class = "sd")
34 )
35
36 model_dist_apim_a <- brm(
37   formula = formula,
38   data = df_apim,
39   family = gaussian(link = identity),
40   prior = priors,
```

```

40 chains = 4,
41 cores = 4,
42 iter = 2000,
43 warmup = 1000,
44 seed = 123,
45 file = file.path('brms_cache', 'example1_dist_apim_a') # Cache the model
46 )

```

1.8 Distinguishable Dyads - Fitting the Model in BRMS

```

1 # Option B: Residual CS for non-independence (no random intercept)
2
3 formula <- bf(
4   satisfaction ~
5
6   # Remove global intercept, introduce male and female intercepts.
7   0 + is_male + is_female +
8
9   # Actor effect for male and female
10  communication_actor_gmc:is_male +
11  communication_actor_gmc:is_female +
12
13  # Partner effect for male and female
14  communication_partner_gmc:is_male +
15  communication_partner_gmc:is_female +
16
17  # Option B: Model non-independence by using compound symmetry of the
18  ↪ residuals.
19  # Aligns with the SEM conceptualization / literature
20  cosy(gr = coupleID),
21
22  # Two distinct residual variances for males vs females:
23  sigma = ~ 0 + is_male + is_female
24 )
25
26 priors <- c(
27   # prior(normal(2, 3), class = "Intercept"),
28   prior(normal(0, 5), class = "b"),
29   prior(student_t(3, 0, 1.5), class = "b", dpar = "sigma"),
30   prior(beta(1, 3), class = "cosy")
31 )
32
33 model_dist_apim_b <- brm(
34   formula = formula,
35   data = df_apim,
36   family = gaussian(link = identity),
37   prior = priors,
38   chains = 4,

```

```

38   cores = 4,
39   iter = 2000,
40   warmup = 1000,
41   seed = 123,
42   file = file.path('brms_cache', 'example1_dist_apim_b') # Cache the model
43 )

```

1.9 Distinguishable Dyads - Check Model Convergence and Fit

Check Rhats and Effective Sample Sizes (ESS_tail and ESS_bulk) directly from the brms summary. Additionally you can (using Model B as an example):

```

rstan::check_hmc_diagnostics(model_dist_apim_b$fit)

```

Divergences:

0 of 4000 iterations ended with a divergence.

Tree depth:

0 of 4000 iterations saturated the maximum tree depth of 10.

Energy:

E-BFMI indicated no pathological behavior.

```

loo::pareto_k_table(loo(model_dist_apim_b))

```

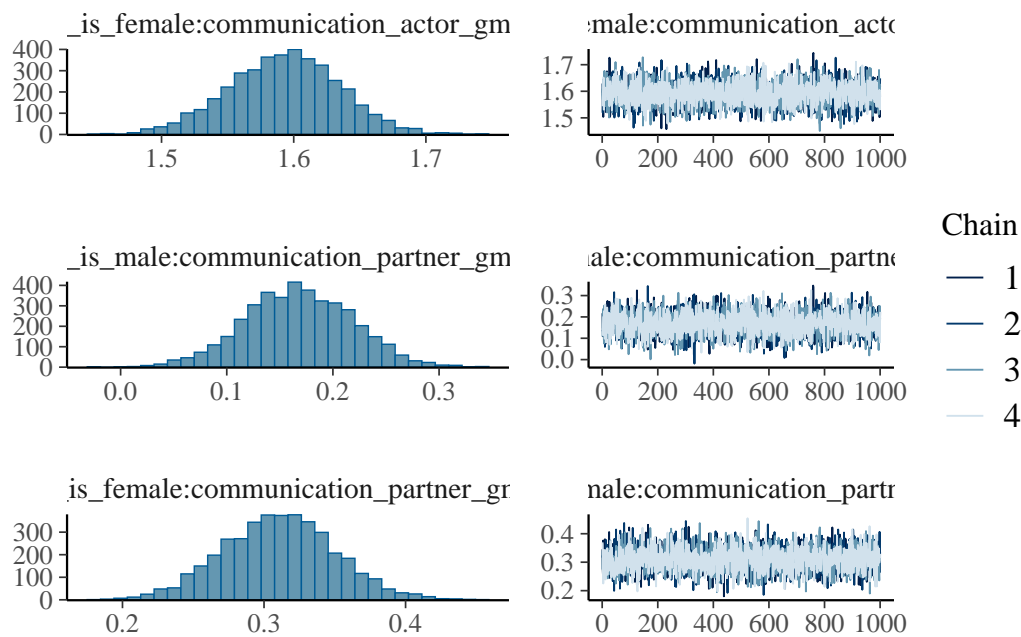
All Pareto k estimates are good ($k < 0.7$).

1.10 Distinguishable Dyads - Check Model Convergence and Fit

```

plot(model_dist_apim_b, ask = FALSE)

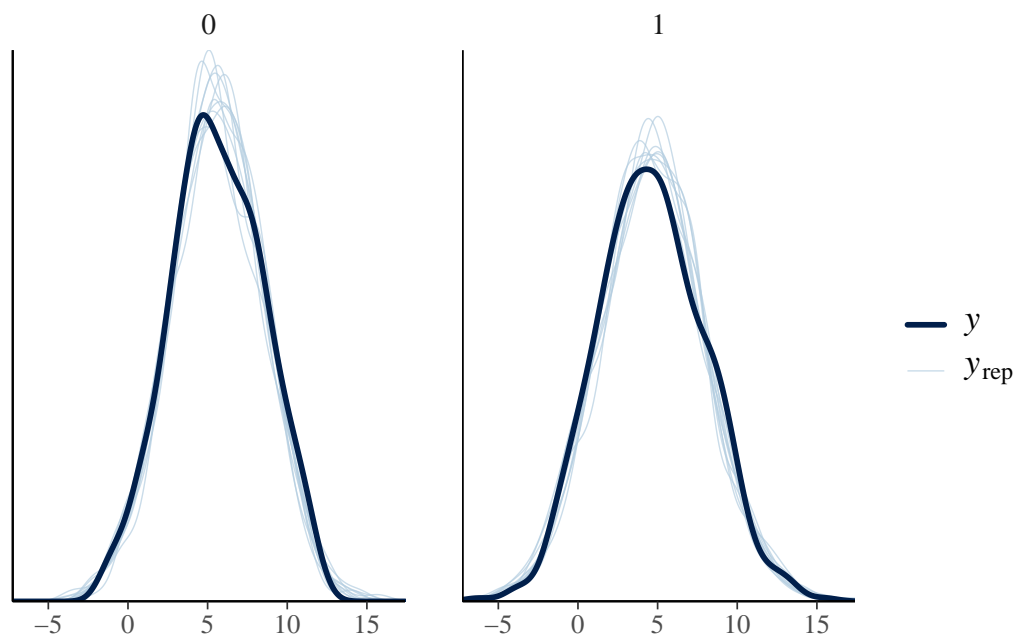
```



1.11 Distinguishable Dyads - Check Model Convergence and Fit

```
pp_check(model_dist_apim_b, 'dens_overlay_grouped', group = 'is_male')
```

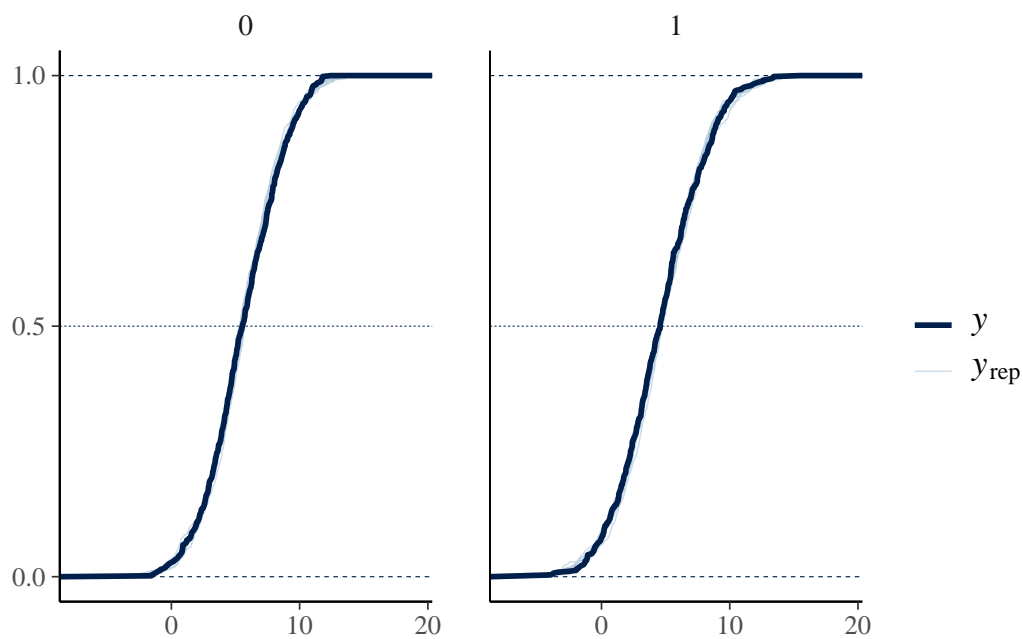
Using 10 posterior draws for ppc type 'dens_overlay_grouped' by default.



1.12 Distinguishable Dyads - Check Model Convergence and Fit

```
pp_check(model_dist_apim_b, 'ecdf_overlay_grouped', group = 'is_male')
```


Using 10 posterior draws for ppc type 'ecdf_overlay_grouped' by default.

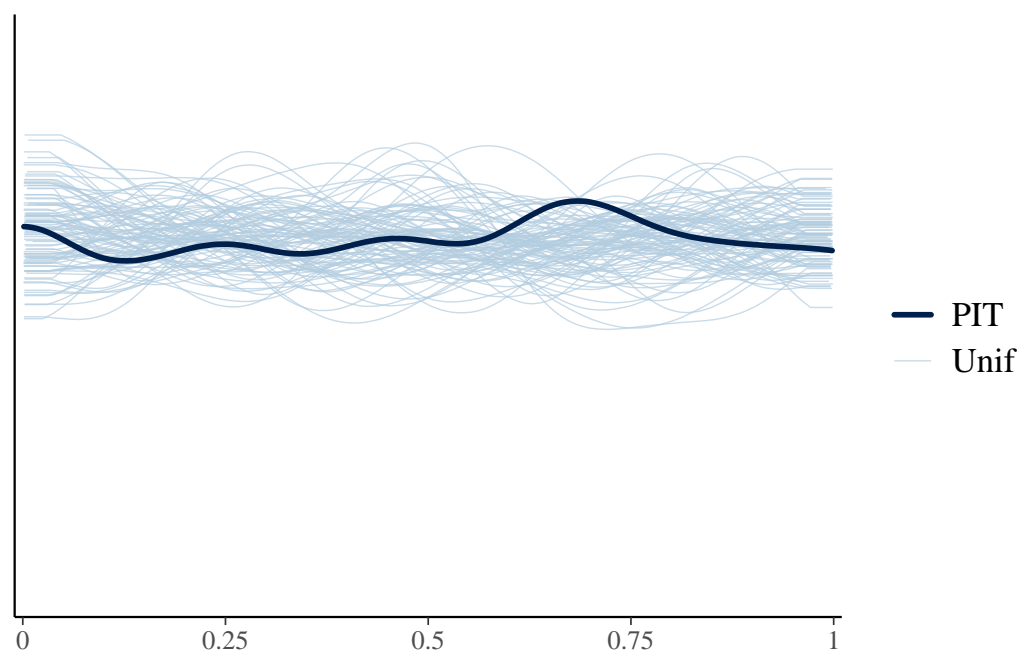


1.13 Distinguishable Dyads - Check Model Convergence and Fit

```
pp_check(model_dist_apim_b, type = "loo_pit_overlay")
```

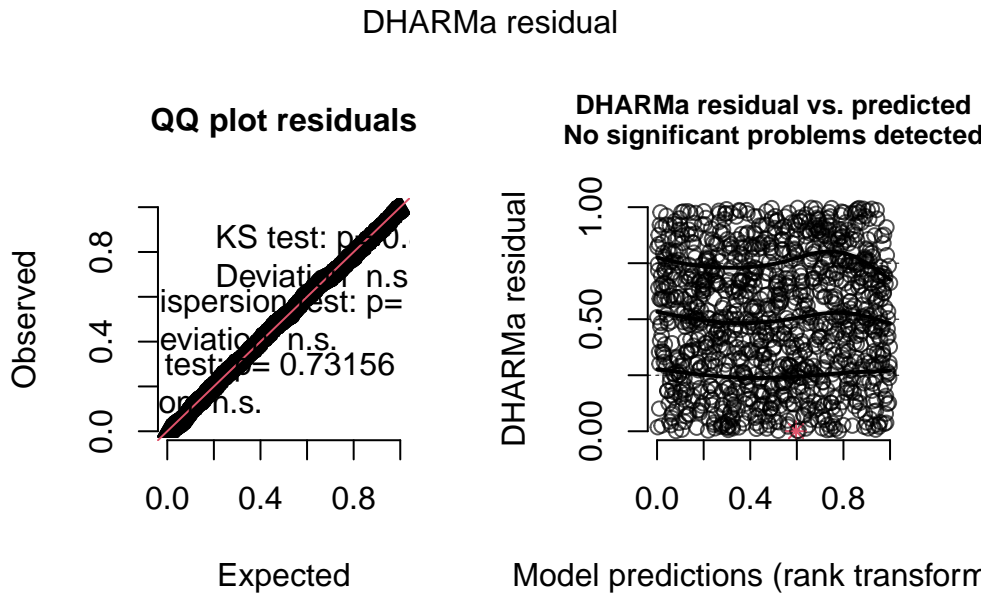
Using all posterior draws for ppc type 'loo_pit_overlay' by default.

NOTE: The kernel density estimate assumes continuous observations and is not optimal for discrete data.



1.14 Distinguishable Dyads - Check Model Convergence and Fit

```
# Custom function to make DHARMA work with brms (see file 'Functions')
DHARMA.check_brms(model_dist_apim_b)
```



1.15 Distinguishable Dyads - Check Model Convergence and Fit

In case of non-convergence, bad residuals or low predictive accuracy, the model is likely misspecified.

You can try different families that match your data by using the `family()` argument. For instance, you can easily conduct an ordinal regression by simply setting `family = cumulative()` and adjusting your priors. For dichotomous outcomes you can use `family = bernoulli()`. Brms supports a variety of families and link functions for these families. In general, you should use the default link function (check `?brms::family.brmsfit()`).

Note that for some of the later models in this tutorial, some ESS values are too low. In this case, usually we would increase iterations (and warmup), and/or `adapt_delta` (see brms documentation).

1.16 Distinguishable Dyads - Results

	Est.	95% CI	Rhat	Bulk_ESS	Tail_ESS
Fixed Effects					
is_male	4.60	[4.47, 4.74]	1.001	5163	3375
is_female	5.61	[5.50, 5.72]	1.000	4706	3152
is_male:communication_actor_gmc	1.84	[1.75, 1.94]	1.002	3380	3262
is_female:communication_actor_gmc	1.59	[1.51, 1.67]	1.002	3812	2935
is_male:communication_partner_gmc	0.17	[0.07, 0.27]	1.000	4128	3151
is_female:communication_partner_gmc	0.31	[0.23, 0.39]	1.000	3663	3475
Residual Structure					
cosy	0.50	[0.44, 0.56]	1.001	4076	3169
sigma_is_male	0.47	[0.41, 0.53]	1.000	4362	2672

sigma__is_female	0.26	[0.20, 0.32]	1.001	4595	3241
------------------	------	--------------	-------	------	------

Note that sigma is reported on the log scale (see `?brms::family.brmsfit()`). Using `exp(value)` retrieves the simulated SD values well.

1.17 Distinguishable Dyads - (Intensive) Longitudinal APIM

1.18 Distinguishable Dyads - L-APIM - Simulated Dataset

userID	coupleID	diaryday	diaryday_c	gender	is_male	is_female	closeness	provided_support_actor	provided_support_partner
31_1	31	0	-27	1	0	1	5.47	6.02	7.09
31_1	31	1	-26	1	0	1	7.69	6.13	6.95
31_1	31	2	-25	1	0	1	5.83	6.1	6.34
31_1	31	3	-24	1	0	1	6.55	6.99	5.39
...
31_2	31	0	-27	2	1	0	5.19	7.09	6.02
31_2	31	1	-26	2	1	0	6.96	6.95	6.13
31_2	31	2	-25	2	1	0	6.9	6.34	6.1
31_2	31	3	-24	2	1	0	7.59	5.39	6.99

1.19 Distinguishable Dyads - L-APIM - Centering

With repeated measures we need to center variables in order to not conflate levels of analysis:

- Between-Person: The person-mean in relation to the grand mean
- Within-Person: Daily fluctuations of an individual from their person-mean

While it may seem like we have 3-levels (couple/person/day), by including the means of both partners and their correlations in the model, all information about the couple-level (level 3) is included. This will become clear when comparing the exchangeable APIM to the DIM. We thus use a 2-level model with the dyad as the level of analysis.

1.20 Distinguishable Dyads - L-APIM - Centering

```
df_long_apim <- bmlm::isolate(  
  df_long_apim,  
  by = 'userID', # NOT coupleID  
  value = c('provided_support_actor', 'provided_support_partner'),  
  which = 'both'  
) %>%  
# renaming to avoid confusion with between- and within COUPLE centred DIM  
  variables  
  rename(  
    provided_support_actor_cwp = provided_support_actor_cw,  
    provided_support_partner_cwp = provided_support_partner_cw,  
    provided_support_actor_cbp = provided_support_actor_cb,  
    provided_support_partner_cbp = provided_support_partner_cb,  
  ) %>%  
# Centering time  
  mutate(  
    diaryday_c = scale(diaryday, center = TRUE, scale = FALSE)  
  )
```

1.21 Distinguishable Dyads - L-APIM - Model nlme

nlme model of this data for reference (simplified random effects structure to achieve convergence).

```
library(nlme)  
  
model1 <- lme(  
  fixed =  
    # Intercepts  
    closeness ~ 0 + is_male + is_female +  
  
    # Time Slopes  
    is_male:diaryday_c +  
    is_female:diaryday_c +
```

```

# Between-Person APIM
is_male:provided_support_actor_cbp +
is_male:provided_support_partner_cbp +

is_female:provided_support_actor_cbp +
is_female:provided_support_partner_cbp +

# Within-Person APIM
is_male:provided_support_actor_cwp +
is_male:provided_support_partner_cwp +

is_female:provided_support_actor_cwp +
is_female:provided_support_partner_cwp,

random =
  ~ 0 + is_male + is_female

  + is_male:diaryday_c + is_female:diaryday_c

# in nlme we need to remove many random slopes to achieve convergence
# alternatively, we could try to remove correlations between the slopes.
#+ is_male:provided_support_actor_cwp
#+ is_male:provided_support_partner_cwp
#+ is_female:provided_support_actor_cwp
#+ is_female:provided_support_partner_cwp
| coupleID,

weights = varIdent(form = ~ 1 | gender), # heterogeneous residual variances
corr = corCompSymm(form = ~ 1 | coupleID/diaryday), # compound symmetry
  ↪ (partner's values on each day)

data = df_long_apim,
control = list(maxIter=1000)
)

summary(model1)

```

1.22 Distinguishable Dyads - L-APIM - Model brms

Equivalent brms model with simplified random effects structure.

```

formula <- bf(
  closeness ~ 0 + is_male + is_female +

  # Time Slopes
  is_male:diaryday_c +
  is_female:diaryday_c +

  # Between-Person APIM

```

```

is_male:provided_support_actor_cbp +
is_male:provided_support_partner_cbp +

is_female:provided_support_actor_cbp +
is_female:provided_support_partner_cbp +

# Within-Person APIM
is_male:provided_support_actor_cwp +
is_male:provided_support_partner_cwp +

is_female:provided_support_actor_cwp +
is_female:provided_support_partner_cwp +

# Accounting for non-independence between partners' means and trajectories
# and effect sensitivities via random effects:
(0 + is_male + is_female
 + is_male:diaryday_c + is_female:diaryday_c
 | coupleID) +

# Accounting for daily non-independence (mimicking nlme's corCompSym)
# modelled via common 'shocks' with a random intercept:
(1 | coupleID:diaryday)

# heteroscedastic residuals
, sigma ~ 0 + is_male + is_female
)

priors <- c(
  prior(normal(4, 2), class = "b", coef = "is_male"), # male intercept
  prior(normal(4, 2), class = "b", coef = "is_female"), # female intercept
  prior(normal(0, 2), class = "b") # other beta coefficients
  # Other priors can be specified, but brms choosed defaults well.
  # Defaults can be seen via default_prior(formula, data)
)

model_dist_apim_long_simple <- brm(
  formula = formula,
  data = df_long_apim,
  family = gaussian(link = identity), # student() is also often a nice robust
  ↪ option.
  prior = priors,
  chains = 4,
  cores = 4,
  iter = 2000,
  warmup = 1000,
  seed = 123,
  file = file.path('brms_cache', 'example1_dist_apim_long_simpel') # Cache the
  ↪ model
)

```

```
summary(model_dist_apim_long_simple)
```

1.23 Distinguishable Dyads - L-APIM - Results (scrollable)

	Est.	95% CI	Rhat	Bulk_ESS	Tail_ESS
Fixed Effects					
is_male	4.47	[4.29, 4.65]	1.014	324	594
is_female	5.71	[5.52, 5.90]	1.042	160	327
is_male:diaryday_c	0.00	[-0.01, 0.00]	1.002	1141	1991
is_female:diaryday_c	0.01	[0.01, 0.01]	1.007	668	2282
is_male:provided_support_actor_cbp	1.13	[0.92, 1.34]	1.004	458	1051
is_male:provided_support_partner_cbp	0.29	[0.09, 0.50]	1.006	451	800
is_female:provided_support_actor_cbp	1.38	[1.18, 1.61]	1.006	395	861
is_female:provided_support_partner_cbp	0.58	[0.37, 0.80]	1.010	364	757
is_male:provided_support_actor_cwp	0.18	[0.14, 0.23]	1.001	7460	3197
is_male:provided_support_partner_cwp	0.12	[0.08, 0.17]	1.001	7634	2827
is_female:provided_support_actor_cwp	0.42	[0.39, 0.44]	1.000	6781	2891
is_female:provided_support_partner_cwp	0.17	[0.14, 0.20]	1.002	6277	3108
Random Effects					
coupleID: sd(is_male)	0.85	[0.74, 0.99]	1.003	593	1179
coupleID: sd(is_female)	0.91	[0.79, 1.06]	1.007	539	1022
coupleID: sd(is_male:diaryday_c)	0.02	[0.01, 0.02]	1.003	1725	2529
coupleID: sd(is_female:diaryday_c)	0.02	[0.01, 0.02]	1.001	1326	2239
coupleID: cor(is_male,is_female)	0.18	[-0.02, 0.36]	1.023	265	510
coupleID: cor(is_male,is_male:diaryday_c)	0.53	[0.33, 0.68]	1.000	1554	2377
coupleID: cor(is_female,is_male:diaryday_c)	0.03	[-0.19, 0.24]	1.005	1184	2163
coupleID: cor(is_male,is_female:diaryday_c)	0.19	[-0.03, 0.39]	1.017	573	1399
coupleID: cor(is_female,is_female:diaryday_c)	0.51	[0.33, 0.65]	1.003	1105	1959
coupleID: cor(is_male:diaryday_c,is_female:diaryday_c)	-0.01	[-0.24, 0.22]	1.003	861	1792
coupleID:diaryday: sd(Intercept)	0.18	[0.10, 0.23]	1.010	202	418
Residual Structure					
sigma_is_male	0.10	[0.08, 0.12]	1.002	1514	2410
sigma_is_female	-0.36	[-0.39, -0.33]	1.005	339	1096

1.24 Distinguishable Dyads - L-APIM - AR1 Model

We can try to add an AR1 residual structure.

```
... +
  ar(time = diaryday, gr = coupleID:userID, p = 1),

sigma ~ 0 + is_male + is_female
)
```

1.25 Distinguishable Dyads - L-APIM - AR1 Model

Warning: Often, the daily shocks and the time slopes are enough and adding AR1 additionally may “soak up” too much variance. this can bias fixed effects estimates towards 0.

You may want to check which model performs better!


```
loo1 <- loo(model_dist_apim_long_simple)
loo2 <- loo(model_dist_apim_long_ar)
```

Warning: Found 2 observations with a `pareto_k > 0.7` in model 'model_dist_apim_long_ar'. We recommend to set 'moment_match = TRUE' in order to perform moment matching for problematic observations.

```
loo::pareto_k_table(loo1)
```

All Pareto k estimates are good ($k < 0.7$).

```
loo::pareto_k_table(loo2)
```

Pareto k diagnostic values:

		Count	Pct.	Min.	ESS
(-Inf, 0.7]	(good)	10998	100.0%	166	
(0.7, 1]	(bad)	2	0.0%	<NA>	
(1, Inf)	(very bad)	0	0.0%	<NA>	

1.26 Distinguishable Dyads - L-APIM - AR1 Model

```
a <- loo_compare(loo1, loo2)
print(a)
```

	elpd_diff	se_diff
model_dist_apim_long_simple	0.0	0.0
model_dist_apim_long_ar	-689.5	37.8

```
report::report(a)
```

The difference in predictive accuracy, as indexed by Expected Log Predictive Density (ELPD), suggests that 'model_dist_apim_long_simple' is the best model (ELPD = -14624.87), followed by 'model_dist_apim_long_ar' (diff-ELPD = -689.47 \pm 37.77, $p < .001$)

In this instance, the AR1 model performs much worse, indicating potential overfitting or other issues. Even in the case of no clear difference, it may be advisable to choose the more parsimonious model, which is the one without AR1.

1.27 Maximal Model Specification (no AR(1))

We add all random slopes and their correlations. This is not converging in `nlme`.

```

formula <- bf(
  closeness ~

  # Intercepts
  0 + is_male + is_female +

  # Time Slopes
  is_male:diaryday_c +
  is_female:diaryday_c +

  # Between-Person APIM
  is_male:provided_support_actor_cbp +
  is_male:provided_support_partner_cbp +

  is_female:provided_support_actor_cbp +
  is_female:provided_support_partner_cbp +

  # Within-Person APIM
  is_male:provided_support_actor_cwp +
  is_male:provided_support_partner_cwp +

  is_female:provided_support_actor_cwp +
  is_female:provided_support_partner_cwp +

  # Accounting for non-independence between partners' means and trajectories
  ↪
  # and effect sensitivities via random effects:
  (0 + is_male + is_female +

    is_male:diaryday_c + is_female:diaryday_c +

    is_male:provided_support_actor_cwp +
    is_male:provided_support_partner_cwp +
    is_female:provided_support_actor_cwp +
    is_female:provided_support_partner_cwp
  | coupleID ) +

  # Accounting for daily non-independence
  (1 | coupleID:diaryday),

  # No more AR1. You could also test MA or ARMA.

  sigma ~ 0 + is_male + is_female # heteroscedastic residuals
)

priors <- c(
  prior(normal(4, 2), class = "b", coef = "is_male"),
  prior(normal(4, 2), class = "b", coef = "is_female"),
  prior(normal(0, 2), class = "b")
)

```

```
)

model_dist_apim_long_complex <- brm(
  formula = formula,
  data = df_long_apim,
  family = gaussian(link = identity),
  prior = priors,
  chains = 4,
  cores = 4,
  iter = 2000,
  warmup = 1000,
  seed = 123,
  file = file.path('brms_cache', 'example1_dist_apim_long_complex') # Cache
  ↪ the model
)
```

1.28 Maximal Model Specification: Results

	Est.	95% CI	Rhat	Bulk_ESS	Tail_ESS
Fixed Effects					
is_male	4.48	[4.30, 4.66]	1.006	401	718
is_female	5.73	[5.55, 5.92]	1.009	349	673
is_male:diaryday_c	0.00	[-0.01, 0.00]	1.001	1488	2443
is_female:diaryday_c	0.01	[0.01, 0.01]	1.001	978	2082
is_male:provided_support_actor_cbp	1.12	[0.91, 1.33]	1.018	478	954
is_male:provided_support_partner_cbp	0.31	[0.10, 0.52]	1.008	608	1175
is_female:provided_support_actor_cbp	1.37	[1.13, 1.59]	1.008	405	795
is_female:provided_support_partner_cbp	0.59	[0.36, 0.81]	1.008	314	759
is_male:provided_support_actor_cwp	0.18	[0.14, 0.22]	1.001	5103	3318
is_male:provided_support_partner_cwp	0.13	[0.08, 0.18]	1.000	5244	3128
is_female:provided_support_actor_cwp	0.42	[0.39, 0.45]	1.000	5005	3222
is_female:provided_support_partner_cwp	0.17	[0.14, 0.20]	1.001	5938	3510
Random Effects					
coupleID: sd(is_male)	0.85	[0.73, 0.98]	1.003	610	1176
coupleID: sd(is_female)	0.92	[0.81, 1.07]	1.004	668	1097
coupleID: sd(is_male:diaryday_c)	0.02	[0.01, 0.02]	1.003	1819	2959
coupleID: sd(is_female:diaryday_c)	0.02	[0.01, 0.02]	1.001	1783	2683
coupleID: sd(is_male:provided_support_actor_cwp)	0.09	[0.02, 0.15]	1.002	1120	1248
coupleID: sd(is_male:provided_support_partner_cwp)	0.13	[0.03, 0.20]	1.003	557	461
coupleID: sd(is_female:provided_support_actor_cwp)	0.08	[0.02, 0.12]	1.005	858	675
coupleID: sd(is_female:provided_support_partner_cwp)	0.05	[0.00, 0.10]	1.006	756	1118
coupleID: cor(is_male,is_female)	0.16	[-0.04, 0.34]	1.015	314	781
coupleID: cor(is_male,is_male:diaryday_c)	0.49	[0.29, 0.65]	1.002	2259	2781
coupleID: cor(is_female,is_male:diaryday_c)	0.01	[-0.22, 0.22]	1.001	1274	2169
coupleID: cor(is_male,is_female:diaryday_c)	0.18	[-0.03, 0.37]	1.001	836	1230
coupleID: cor(is_female,is_female:diaryday_c)	0.50	[0.33, 0.65]	1.000	1275	2344
coupleID: cor(is_male:diaryday_c,is_female:diaryday_c)	-0.03	[-0.25, 0.20]	1.002	1014	1533
coupleID: cor(is_male,is_male:provided_support_actor_cwp)	0.41	[-0.07, 0.75]	1.000	4911	3042
coupleID: cor(is_female,is_male:provided_support_actor_cwp)	0.21	[-0.21, 0.61]	1.001	5471	3222
coupleID: cor(is_male:diaryday_c,is_male:provided_support_actor_cwp)	0.36	[-0.13, 0.73]	1.000	4157	2633
coupleID: cor(is_female:diaryday_c,is_male:provided_support_actor_cwp)	0.31	[-0.15, 0.69]	1.000	4881	3127
coupleID: cor(is_male,is_male:provided_support_partner_cwp)	0.06	[-0.28, 0.43]	1.001	4181	2214
coupleID: cor(is_female,is_male:provided_support_partner_cwp)	0.12	[-0.24, 0.45]	1.001	5603	2758
coupleID: cor(is_male:diaryday_c,is_male:provided_support_partner_cwp)	0.05	[-0.34, 0.43]	1.001	3330	2187
coupleID: cor(is_female:diaryday_c,is_male:provided_support_partner_cwp)	0.12	[-0.25, 0.48]	1.000	3809	2656

coupleID: cor(is_male:provided_support_actor_cwp,is_male:provided_support_partner_cwp)	0.29	[-0.31, 0.75]	1.005	884	1162
coupleID: cor(is_male,is_female:provided_support_actor_cwp)	-0.07	[-0.44, 0.29]	1.000	3826	2826
coupleID: cor(is_female,is_female:provided_support_actor_cwp)	0.22	[-0.16, 0.58]	1.001	4594	2147
coupleID: cor(is_male:diaryday_c,is_female:provided_support_actor_cwp)	-0.15	[-0.55, 0.24]	1.001	3647	2949
coupleID: cor(is_female:diaryday_c,is_female:provided_support_actor_cwp)	-0.08	[-0.46, 0.32]	1.001	4413	1987
coupleID: cor(is_male:provided_support_actor_cwp,is_female:provided_support_actor_cwp)	-0.20	[-0.68, 0.37]	1.004	1130	2087
coupleID: cor(is_male:provided_support_partner_cwp,is_female:provided_support_actor_cwp)	-0.13	[-0.63, 0.40]	1.001	1733	2599
coupleID: cor(is_male,is_female:provided_support_partner_cwp)	0.29	[-0.27, 0.72]	1.006	4349	2048
coupleID: cor(is_female,is_female:provided_support_partner_cwp)	0.07	[-0.41, 0.53]	1.002	5425	2887
coupleID: cor(is_male:diaryday_c,is_female:provided_support_partner_cwp)	0.13	[-0.39, 0.60]	1.002	5132	2484
coupleID: cor(is_female:diaryday_c,is_female:provided_support_partner_cwp)	0.18	[-0.35, 0.62]	1.002	4223	2456
coupleID: cor(is_male:provided_support_actor_cwp,is_female:provided_support_partner_cwp)	0.20	[-0.44, 0.71]	1.001	2034	3036
coupleID: cor(is_male:provided_support_partner_cwp,is_female:provided_support_partner_cwp)	0.09	[-0.52, 0.63]	1.002	2829	3099
coupleID: cor(is_female:provided_support_actor_cwp,is_female:provided_support_partner_cwp)	0.12	[-0.50, 0.64]	1.002	2555	2614
coupleID:diaryday: sd(Intercept)	0.18	[0.07, 0.23]	1.017	190	147
Residual Structure					
sigma_is_male	0.10	[0.07, 0.12]	1.004	925	1505
sigma_is_female	-0.36	[-0.39, -0.33]	1.011	314	396

1.29 Maximal Model Specification: Diagnostics

In complex models like this, we want to make sure we are not **overfitting**.

- Conduct regular model convergence checks (inspecting chains, Rhats, ESS, Multicollinearity, Residual diagnostics etc.). If these are bad (but good for a simpler model), it could be a sign of overfitting.
- Leave-one-out cross-validation is a powerful tool for investigating overfitting. It tests out of sample prediction and thus punishes overfitting.

1.30 Maximal Model Specification: Diagnostics

```
loo1 <- loo(model_dist_apim_long_simple)
loo2 <- loo(model_dist_apim_long_complex)

loo::pareto_k_table(loo1)
```

All Pareto k estimates are good ($k < 0.7$).

```
loo::pareto_k_table(loo2)
```

All Pareto k estimates are good ($k < 0.7$).

In case of High Pareto-Ks there are influential datapoints present. This *can* be a sign of overfitting but it is not necessarily a problem. But it is recommended to use `loo(model, moment_match = TRUE)` in these cases (the function will tell you).

The loo values (elpd) themselves cannot be interpreted individually, but we can compare models again (next page)

1.31 Maximal Model Specification: Diagnostics

```
a <- loo_compare(loo1, loo2)
print(a)
```

	elpd_diff	se_diff
model_dist_apim_long_complex	0.0	0.0
model_dist_apim_long_simple	-15.3	6.7

No obvious overfitting detected. In fact, the more complex model seems to be more predictive. Some rule of thumbs say that if the difference is higher than 2x the SE or 4x the SE we can say there is a difference.

If there was no obvious difference, we could follow our theory and the philosophy of the maximal random effects structure (Barr et al., 2013). Alternatively, we could follow the philosophy of parsimony, especially in the territory of such complex models.

(continued on next page)

1.32 Maximal Model Specification: Diagnostics

One approach is to use the `report` package's approach to obtain a p-value for the comparison (see their documentation for which assumptions they make):

```
report::report(a)
```

The difference in predictive accuracy, as indexed by Expected Log Predictive Density (ELPD), suggests that 'model_dist_apim_long_complex' is the best model (ELPD = -14609.55), followed 'model_dist_apim_long_simple' (diff-ELPD = -15.32 +- 6.67, p = 0.022)

In this instance, the maximal model (no AR) seems to be the preferred one.

In the case of overfitting, we could try to improve the model by subsequently excluding the smallest random effects (or correlations between them) and compare various models. For more guidance on random effects check out del Rosario & West (2025).

1.33 Maximal Model Specification: Comparing Estimates to Simulated Values (Ground Truth)

This is posterior version 1.6.1

Attaching package: 'posterior'

The following objects are masked from 'package:stats':

mad, sd, var

The following objects are masked from 'package:base':

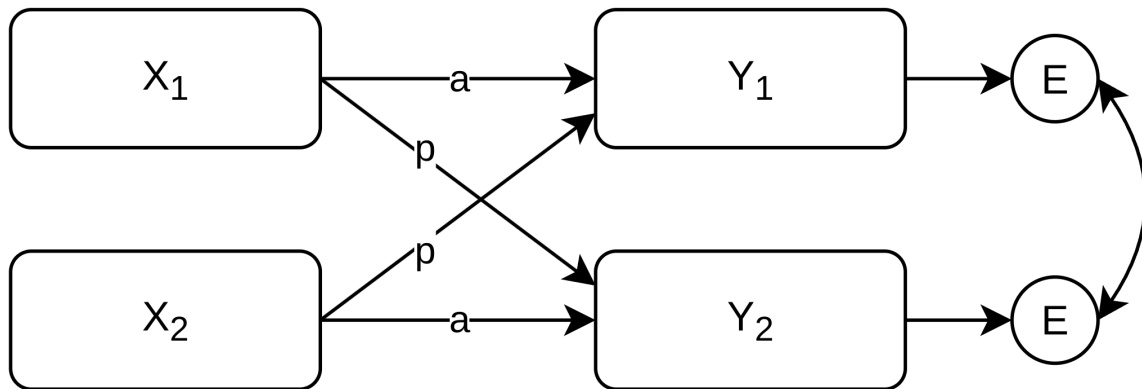
%in%, match

Warning: Argument 'pars' is deprecated. Please use 'variable' instead.
Warning: Argument 'pars' is deprecated. Please use 'variable' instead.

Group	Parameter	True (Simulated)	Estimate	95% CI	Coverage	% Error	Recovery
Fixed	Intercept (male)	4.5650000	4.4832559	[4.304, 4.656]	Yes	-1.79%	Excellent
Fixed	Intercept (female)	5.7700000	5.7329883	[5.547, 5.916]	Yes	-0.64%	Excellent
Fixed	Time slope (male)	-0.0050000	-0.0030886	[-0.007, 0.001]	Yes	-38.23%	OK
Fixed	Time slope (female)	0.0100000	0.0105112	[0.007, 0.014]	Yes	5.11%	Good
Between-person	Actor BP (male)	1.2000000	1.1194834	[0.912, 1.326]	Yes	-6.71%	Good
Between-person	Partner BP (male)	0.3000000	0.3077934	[0.102, 0.522]	Yes	2.60%	Excellent
Between-person	Actor BP (female)	1.5000000	1.3691118	[1.132, 1.589]	Yes	-8.73%	Good
Between-person	Partner BP (female)	0.5000000	0.5882102	[0.364, 0.812]	Yes	17.64%	OK
Within-person	Actor WP (male)	0.2000000	0.1817432	[0.138, 0.224]	Yes	-9.13%	Good
Within-person	Partner WP (male)	0.1000000	0.1268500	[0.079, 0.177]	Yes	26.85%	OK
Within-person	Actor WP (female)	0.4000000	0.4204154	[0.389, 0.450]	Yes	5.10%	Good
Within-person	Partner WP (female)	0.2000000	0.1713362	[0.143, 0.200]	Yes	-14.33%	Good
Random effects (SD)	Couple intercept SD (male)	0.6819392	0.8500226	[0.731, 0.982]	No	24.65%	Miss

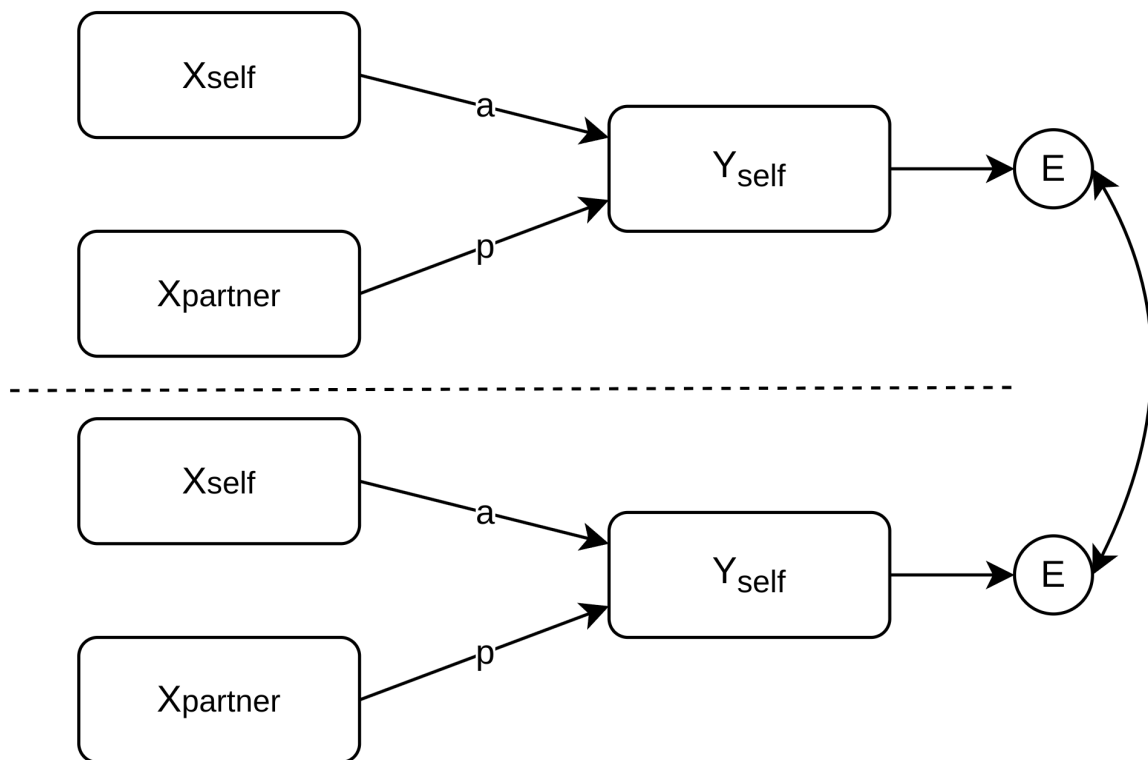
Random effects (SD)	Couple intercept SD (female)	0.7236049	0.9282122	[0.808, 1.073]	No	28.28%	Miss
Random effects (SD)	Time slope SD (male)	0.0149848	0.0168064	[0.014, 0.020]	Yes	12.16%	Good
Random effects (SD)	Time slope SD (female)	0.0150338	0.0165778	[0.014, 0.020]	Yes	10.27%	Good
Random effects (SD)	Same-day couple×day SD	0.1709812	0.1716311	[0.068, 0.232]	Yes	0.38%	Excellent
Residuals	Residual SD (male) (exp)	1.1063074					—
Residuals	Residual SD (female) (exp)	0.6969923					—

1.34 Exchangeable Dyads - Cross-Sectional APIM



(e.g., del Rosario & West, 2025; Kenny & Ackerman, 2023)

1.35 Exchangeable Dyads - Cross-Sectional APIM



(e.g., del Rosario & West, 2025; Kenny & Ackerman, 2023)

1.36 Exchangeable Dyads - Main Assumptions

Partners are exchangeable, i.e., not *systematically* different.

- Equal actor effects
- Equal partner effects

- Equal means
- Equal residual variances

But they should still be allowed to vary within each couple, while being correlated.

(e.g., del Rosario & West, 2025; Kenny & Ackerman, 2023)

1.37 Exchangeable Dyads - Cross-Sectional APIM: Data

Same cross-sectional data in the same 4-field actor-partner format as before.

```
print_df(head(df_apim),  
         font_size = font_size)
```

userID	coupleID	gender	is_male	is_female	communication	satisfaction	communication_actor	communication_partner	communication_actor_gmc	communication_partner_gmc
1_1	1	1	0	1	4.851107	4.841332	4.851107	6.004533	-0.1553604	0.9980657
1_2	1	2	1	0	6.004533	5.577165	6.004533	4.851107	0.9980657	0.9980657
2_1	2	1	0	1	5.881321	7.248741	5.881321	6.433723	0.8748537	0.8748537
2_2	2	2	1	0	6.433723	6.960293	6.433723	5.881321	1.4272563	1.4272563
3_1	3	1	0	1	4.283971	6.928699	4.283971	5.516060	-0.7224960	-0.7224960
3_2	3	2	1	0	5.516060	6.077688	5.516060	4.283971	0.5095933	0.5095933

1.38 Exchangeable Dyads - Cross-Sectional APIM: Model

```
1 formula <- bf(
2   satisfaction ~ 1 +
3     communication_actor_gmc + communication_partner_gmc +
4
5     # Option 1: A single couple level random intercept.
6     # In the cross sectional case this is the maximal random effects structure
7     # (1 | coupleID)
8
9     # Option 2: using a compound symmetry residual structure.
10    cosy(gr = coupleID)
11
12    # Note: no need to model separate sigmas for each partner.
13    # Homogeneous residual variance is estimated.
14    # Implied: sigma = ~ 1
15  )
16
17 priors <- c(
18   prior(normal(2, 10), class = "Intercept"),
19   prior(normal(0, 5), class = "b"),
20   prior(student_t(3, 0, 1.5), class = "sigma"),
21   prior(beta(1, 3), class = "cosy")
22 )
23
24 model_ind_apim <- brm(
25   formula = formula,
26   data = df_apim,
27   family = gaussian(link = identity),
28   prior = priors,
29   chains = 4,
30   cores = 4,
31   iter = 2000,
32   warmup = 1000,
33   seed = 123,
34   file = file.path('brms_cache', 'example1_ind_apim') # Cache the model
35 )
```

1.39 Exchangeable Dyads - Cross-Sectional APIM: Results

	Est.	95% CI	Rhat	Bulk_ESS	Tail_ESS
Fixed Effects					
Intercept	5.11	[5.00, 5.22]	1.000	4801	3093
communication_actor_gmc	1.72	[1.66, 1.78]	1.000	5048	3183
communication_partner_gmc	0.24	[0.18, 0.30]	1.001	5120	2749
Residual Structure					
cosy	0.34	[0.27, 0.41]	1.002	4332	2705
sigma	1.55	[1.49, 1.62]	1.002	4215	2828

1.40 Exchangeable Dyads - Cross-Sectional APIM: Test for Distinguishability

Leave-one-out (loo) cross-validation for model comparison (even if not nested).

```
a <- loo_compare(
  loo(model_ind_apim),
  loo(model_dist_apim_b)
)
print(a)
```

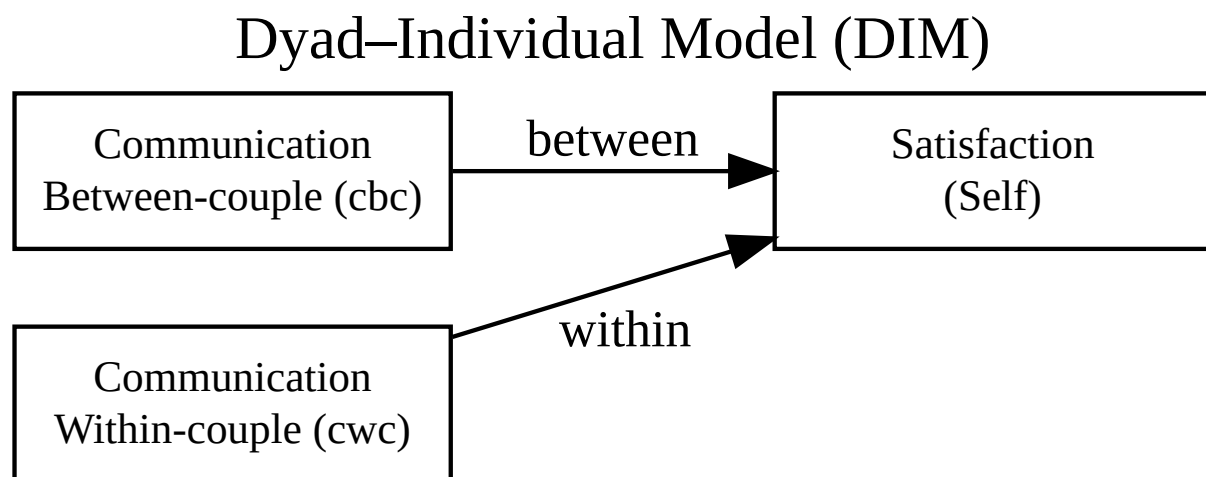
Model	elpd_diff	se_diff
model_dist_apim	0.0	0.0
model_ind_apim	-172.9	16.5

```
report::report(a)
```

The difference in predictive accuracy, as indexed by Expected Log Predictive Density (ELPD-LOO), suggests that 'model_dist_apim_b' is the best model (ELPD = -1806.42), followed by 'model_ind_apim' (diff-ELPD = -172.85 +- 16.53, $p < .001$). See: documentation of "report"

1.41 Exchangeable Dyads - Cross-Sectional DIM

file:///tmp/Rtmptah2kd/file73865be372e4/widget738643d57a3e.html screenshot completed



1.42 Exchangeable Dyads - Cross-Sectional DIM: Data

Starting from scratch (no 4-field data needed)

userID	coupleID	communication	satisfaction
1_1	1	4.851107	4.841332

1_2	1	6.004533	5.577165
2_1	2	5.881321	7.248741
2_2	2	6.433723	6.960293
3_1	3	4.283971	6.928699
3_2	3	5.516060	6.077688

1.43 Exchangeable Dyads - Cross-Sectional DIM: Centering

Decompose communication variance into:

- **Between-couple** (cbc): Couple-mean communication skills in relation to other couples
- **Within-couple** (cwc): Individuals' communication skills in relation to their couple-mean

Same assumption about exchangeability as in the APIM

1.44 Exchangeable Dyads - Cross-Sectional DIM: Centering

```
df_dim <- df_apim2 %>%
  group_by(coupleID) %>%
  mutate(
    communication_cm = mean(communication, na.rm = TRUE),
    communication_cwc = communication - communication_cm
  ) %>%
  ungroup() %>%
  mutate(
    communication_cbc = communication_cm - mean(communication_cm, na.rm = TRUE)
  ) %>%
  dplyr::select(c('userID', 'coupleID', 'satisfaction', 'communication_cwc',
    ↪ 'communication_cbc'))

print_df(head(df_dim),
  font_size = font_size)
```

userID	coupleID	satisfaction	communication_cwc	communication_cbc
1_1	1	4.841332	-0.5767130	0.4213527
1_2	1	5.577165	0.5767130	0.4213527
2_1	2	7.248741	-0.2762013	1.1510550
2_2	2	6.960293	0.2762013	1.1510550
3_1	3	6.928699	-0.6160447	-0.1064514
3_2	3	6.077688	0.6160447	-0.1064514

1.45 Exchangeable Dyads - Cross-Sectional DIM: Model

```
1 formula <- bf(
2   satisfaction ~ 1 +
3     communication_cbc + communication_cwc +
4     cosy(gr = coupleID))
```

```

5 )
6
7 priors <- c(
8   prior(normal(2, 10), class = "Intercept"),
9   prior(normal(0, 5), class = "b"),
10  prior(student_t(3, 0, 1.5), class = "sigma"),
11  prior(beta(1, 3), class = "cosy")
12 )
13
14 model_ind_dim <- brm(
15   formula = formula,
16   data = df_dim,
17   family = gaussian(link = identity),
18   prior = priors,
19   chains = 4,
20   cores = 4,
21   iter = 2000,
22   warmup = 1000,
23   seed = 123,
24   file = file.path('brms_cache', 'example1_ind_dim') # Cache the model
25 )

```

1.46 Exchangeable Dyads - Cross-Sectional DIM: Results

	Est.	95% CI	Rhat	Bulk_ESS	Tail_ESS
Fixed Effects					
Intercept	5.11	[5.01, 5.21]	1.003	5216	2981
communication_cbc	1.96	[1.88, 2.04]	1.000	4664	3077
communication_cwc	1.48	[1.39, 1.57]	1.003	4715	3276
Residual Structure					
cosy	0.34	[0.26, 0.41]	1.000	4312	3208
sigma	1.55	[1.48, 1.62]	1.000	4479	3342

1.47 Equivalence APIM and DIM

	Est.	95% CI
Fixed Effects		
Intercept	5.11	[5.00, 5.22]
communication_actor_gmc	1.72	[1.66, 1.78]
communication_partner_gmc	0.24	[0.18, 0.30]
Residual Structure		
cosy	0.34	[0.27, 0.41]
sigma	1.55	[1.49, 1.62]

	Est.	95% CI
Fixed Effects		
Intercept	5.11	[5.01, 5.21]
communication_cbc	1.96	[1.88, 2.04]
communication_cwc	1.48	[1.39, 1.57]

Residual Structure

cosy	0.34	[0.26, 0.41]
sigma	1.55	[1.48, 1.62]

$$b_{actor_gmc} + b_{partner_gmc} = b_{cbc}$$

$$b_{actor_gmc} - b_{partner_gmc} = b_{cwc}$$

(Bolger et al., 2025)

1.48 Equivalence APIM and DIM

	Est.	95% CI
Fixed Effects		
Intercept	5.11	[5.00, 5.22]
communication_actor_gmc	1.72	[1.66, 1.78]
communication_partner_gmc	0.24	[0.18, 0.30]

	Est.	95% CI
Fixed Effects		
Intercept	5.11	[5.01, 5.21]
communication_cbc	1.96	[1.88, 2.04]
communication_cwc	1.48	[1.39, 1.57]

1.49 Equivalence APIM and DIM

- If the couple mean goes up by 1 and the within-couple deviation from the couple mean stays fixed, this means that both partners' predictors must go up by 1 unit. Thus, the effects are linear combinations:

$$b_{cbc} = b_{actor_gmc} + b_{partner_gmc}$$

- Similarly, if a person's deviation from their couple mean is one unit higher and the couple mean stays fixed, this means their partners' deviation must be one unit lower. Thus, the effects are a linear combination:

$$b_{cwc} = b_{actor_gmc} - b_{partner_gmc}$$

(Bolger et al., 2025)

1.50 Equivalence APIM and DIM

- Conversely, to obtain the actor and partner effects given the DIM estimates:

$$b_{actor_gmc} = \frac{b_{cbc} + b_{cwc}}{2}$$

$$b_{partner_gmc} = \frac{b_{cbc} - b_{cwc}}{2}$$

1.51 Equivalence APIM and DIM

Example: using APIM coefficients to obtain DIM coefficients and computing *Credible Intervals* of DIM coefficients.

```
a <- hypothesis(
  model_ind_apim,
  "communication_actor_gmc + communication_partner_gmc = 0"
)

round(a$hypothesis[,c(2,3,4,5)], 2)
```

	Estimate	Est.Error	CI.Lower	CI.Upper
1	1.96	0.04	1.88	2.04

	Est.	95% CI
Fixed Effects		
Intercept	5.11	[5.00, 5.22]
communication_actor_gmc	1.72	[1.66, 1.78]
communication_partner_gmc	0.24	[0.18, 0.30]

	Est.	95% CI
Fixed Effects		
Intercept	5.11	[5.01, 5.21]
communication_cbc	1.96	[1.88, 2.04]
communication_cwc	1.48	[1.39, 1.57]

1.52 Equivalence APIM and DIM: Takeaway

- **APIM** and **DIM** are reparametrizations of the same model
- **APIM**: intuitive actor/partner framing
- **DIM**: clean separation of between vs within components
- Estimating one model allows for directly obtaining estimates of the other
- Same random-effects structure at dyad level and same assumptions

1.52.1 Side Note:

In the **distinguishable** case, the **Dyadic Score Model (DSM)** (e.g., Stadler et al., 2023) is equivalent to the APIM! (Bolger et al., 2025; Iida et al., 2018)

1.53 Exchangeable Dyads - Longitudinal APIM/DIM: Simulated Data

We use the same data as in the exchangeable case. But we don't need the gender-related variables anymore, as we assume that partners are indistinguishable.

userID	coupleID	diaryday	diaryday_c	closeness	provided_support_actor	provided_support_partner	provided_support_actor_cbp	provided_support_actor_cwp	provided_support_partner_cbp	provided_support_partner_cwp
31_1	31	0	-27	5.47	6.02	7.09	0.65	0.41	0.65	0.58
31_1	31	1	-26	7.69	6.13	6.95	0.65	0.51	0.65	0.58
31_1	31	2	-25	5.83	6.1	6.34	0.65	0.49	0.65	0.58
31_1	31	3	-24	6.55	6.99	5.39	0.65	1.38	0.65	0.58
...
31_2	31	0	-27	5.19	7.09	6.02	0.58	1.54	0.58	0.65
31_2	31	1	-26	6.96	6.95	6.13	0.58	1.4	0.58	0.65
31_2	31	2	-25	6.9	6.34	6.1	0.58	0.79	0.58	0.65
31_2	31	3	-24	7.59	5.39	6.99	0.58	-0.15	0.58	0.65

1.54 Exchangeable Dyads - Longitudinal DIM/APIM

- We need an APIM or DIM on both the between person level and the within-person level.
- Due to equivalence, we could have a DIM on one level and an APIM on the other.

If we want a within-person DIM:

```
df_long_dim <- df_long_apim2 %>%
  mutate(

    # Between Person Level DIM (centering between couples and within couple
    ↪ between person)
    provided_support_cbc = (provided_support_actor_cbp +
    ↪ provided_support_partner_cbp) / 2,
    provided_support_cwcbp = (provided_support_actor_cbp -
    ↪ provided_support_partner_cbp) / 2,

    # Within Person Level DIM (couple mean and deviation on each day)
    provided_support_cwp_mean = (provided_support_actor_cwp +
    ↪ provided_support_partner_cwp) / 2,
    provided_support_cwp_halfdiff = (provided_support_actor_cwp -
    ↪ provided_support_partner_cwp) / 2
  )
```

1.55 Exchangeable Dyads - Longitudinal DIM/APIM: Preparing the sum and difference approach

We need to randomly assign -1 and +1 for each Partner within each couple. This will be needed for contrast coding the intercept.

For example:

```
df_long_dim <- df_long_dim %>%
  group_by(coupleID) %>%
  mutate(
    base = ifelse(userID == min(userID), 1, -1),
    flip = 1 - 2 * rbinom(1, 1, 0.5), # yields +1 or -1 once per couple
    Idiff = base * flip
  ) %>%
  ungroup() %>%
  relocate(Idiff, .after = coupleID) %>%
  dplyr::select(-base, -flip)

print_df(print_couple_preview(df_long_dim, couple_id = "31"),
  font_size = font_size)
```

1.56 Exchangeable Dyads - Longitudinal DIM/APIM: Preparing the Sum and Difference Approach

userID	coupleID	Idiff	diaryday	diaryday_c	closeness	provided_support_actor	provided_support_partner	prov
31_1	31	-1	0	-27	5.47	6.02	7.09	0.65
31_1	31	-1	1	-26	7.69	6.13	6.95	0.65
31_1	31	-1	2	-25	5.83	6.1	6.34	0.65
31_1	31	-1	3	-24	6.55	6.99	5.39	0.65
31_1	31	-1	4	-23	7.03	6.7	5.19	0.65
...
31_2	31	1	0	-27	5.19	7.09	6.02	0.58
31_2	31	1	1	-26	6.96	6.95	6.13	0.58
31_2	31	1	2	-25	6.9	6.34	6.1	0.58
31_2	31	1	3	-24	7.59	5.39	6.99	0.58
31_2	31	1	4	-23	7.55	5.19	6.7	0.58

1.57 Exchangeable Dyads - Longitudinal APIM/DIM: Model

```

formula <- bf(
  closeness ~ 1 +

    diaryday_c +

    # Within-person APIM
    provided_support_actor_cwp +
    provided_support_partner_cwp +
    # Equivalent to within-person DIM:
    # provided_support_cwp_mean +
    # provided_support_cwp_halfdiff +

    # Between-person APIM
    provided_support_actor_cbp +
    provided_support_partner_cbp +
    # Equivalent to between-person DIM
    # provided_support_cwcbp+
    # provided_support_cbc +

    # Dyad-Level intercept and slopes for time-varying predictors
    (1 + diaryday_c + provided_support_actor_cwp +
    ↪ provided_support_partner_cwp | coupleID) +
    # Both partners' deviations from these dyad-level means and slopes.
    # Note: separate random effects block to make them uncorrelated form
    ↪ dyad-level REs.
    (0 + Idiff +
      I(Idiff * diaryday_c) +
      I(Idiff * provided_support_actor_cwp) +
      I(Idiff * provided_support_partner_cwp) | coupleID) +

    # Accounting for same-day shocks/coupling
    (1 | coupleID:diaryday)

```

```

# Autocorrelated residuals
# Due to bad fit in the distinguishable case, we omit AR1 from the start.
# ar(time = diaryday, gr = coupleID:userID, p = 1)

# Again, no need to model heterogeneous residual variances (sigma)
# Implied: sigma = ~ 1
)

priors <- c(
  prior(normal(4, 2), class = "Intercept"),
  prior(normal(0, 2), class = "b")

  # We could set priors for other things, but brms sets good default priors
  #prior(student_t(3, 0, 1.5), class = "sd"),
  #prior(lkj(2), class = "cor"), # correlation prior for random effect matrix
  #prior(student_t(3, 0, 1.5), class = "sigma")
)

model_apim_ind_long <- brm(
  formula = formula,
  data = df_long_dim,
  family = gaussian(link = identity),
  prior = priors,
  chains = 4,
  cores = 4,
  iter = 2000,
  warmup = 1000,
  seed = 123,
  file = file.path('brms_cache', 'model_apim_ind_long_apim')
)

```

1.58 Exchangeable Dyads - Longitudinal APIM/DIM: Explanation Idiff

For appropriate random effects, we can use the Sum and Difference approach (del Rosario & West, 2025; Kenny & Ackerman, 2023):

- Randomly give one partner a constant of -1 and the other a constant of 1
- The couple level intercept represents the mean (or sum) of both partners' intercepts. (1 | coupleID)
- A column of 1s and -1s represent deviations (difference) of each partner from the couple intercept with each partner contributing equally in one direction (+1 and -1)

Exchangeability is retained: if partners are flipped, the result is the same.

(del Rosario & West (2025) provide practical guidance on how to reduce the random effects structure in case of non-convergence.)

1.59 Exchangeable Dyads - Longitudinal APIM/DIM: Results

Comparing APIM (left) and DIM (right) output of fixed effects.

	Est.	95% CI
Fixed Effects		
Intercept	5.12	[4.99, 5.24]
diaryday_c	0.00	[0.00, 0.01]
provided_support_actor_cwp	0.30	[0.27, 0.33]
provided_support_partner_cwp	0.15	[0.12, 0.18]
provided_support_actor_cbp	1.31	[1.14, 1.47]
provided_support_partner_cbp	0.40	[0.22, 0.56]

	Est.	95% CI
Fixed Effects		
Intercept	5.11	[4.98, 5.25]
diaryday_c	0.00	[0.00, 0.01]
provided_support_cwp_mean	0.45	[0.41, 0.49]
provided_support_cwp_halfdiff	0.15	[0.11, 0.19]
provided_support_cbc	1.70	[1.53, 1.85]
provided_support_cwcbp	0.90	[0.62, 1.18]

Equivalence holds on both levels. Numeric deviations arise due to priors, sampling noise, and rounding.

1.60 Exchangeable Dyads - Longitudinal APIM/DIM: Results

APIM Results in Detail:

	Est.	95% CI	Rhat	Bulk_ESS	Tail_ESS
Fixed Effects					
Intercept	5.12	[4.99, 5.24]	1.005	325	765
diaryday_c	0.00	[0.00, 0.01]	1.001	1050	1887
provided_support_actor_cwp	0.30	[0.27, 0.33]	1.000	4964	2708
provided_support_partner_cwp	0.15	[0.12, 0.18]	1.000	5544	3401
provided_support_actor_cbp	1.31	[1.14, 1.47]	1.006	357	803
provided_support_partner_cbp	0.40	[0.22, 0.56]	1.003	389	725
Random Effects					
coupleID: sd(Intercept)	0.68	[0.59, 0.79]	1.007	608	1085
coupleID: sd(diaryday_c)	0.01	[0.01, 0.01]	1.003	1705	2481
coupleID: sd(provided_support_actor_cwp)	0.05	[0.01, 0.09]	1.002	1321	1129
coupleID: sd(provided_support_partner_cwp)	0.08	[0.04, 0.12]	1.001	1218	1284
coupleID: sd(Idiff)	0.85	[0.75, 0.99]	1.017	370	744
coupleID: sd(IIdiffMUdiaryday_c)	0.01	[0.01, 0.02]	1.005	1276	2323
coupleID: sd(IIdiffMUprovided_support_actor_cwp)	0.14	[0.11, 0.18]	1.003	1906	2697
coupleID: sd(IIdiffMUprovided_support_partner_cwp)	0.06	[0.01, 0.10]	1.003	787	1219
coupleID: cor(Intercept,diaryday_c)	0.58	[0.40, 0.72]	1.000	1811	2711
coupleID: cor(Intercept,provided_support_actor_cwp)	0.56	[0.05, 0.90]	1.001	3440	2281
coupleID: cor(diaryday_c,provided_support_actor_cwp)	0.45	[-0.12, 0.84]	1.000	3891	2960
coupleID: cor(Intercept,provided_support_partner_cwp)	0.22	[-0.12, 0.55]	1.000	4211	2831
coupleID: cor(diaryday_c,provided_support_partner_cwp)	0.21	[-0.18, 0.57]	1.001	3273	2922
coupleID: cor(provided_support_actor_cwp,provided_support_partner_cwp)	0.44	[-0.32, 0.89]	1.006	506	816
coupleID: cor(Idiff,IIdiffMUdiaryday_c)	0.61	[0.46, 0.74]	1.002	1266	1837
coupleID: cor(Idiff,IIdiffMUprovided_support_actor_cwp)	0.68	[0.47, 0.84]	1.002	1624	2429
coupleID: cor(IIdiffMUdiaryday_c,IIdiffMUprovided_support_actor_cwp)	0.44	[0.17, 0.66]	1.000	2260	3094
coupleID: cor(Idiff,IIdiffMUprovided_support_partner_cwp)	0.15	[-0.29, 0.60]	1.001	4898	2801
coupleID: cor(IIdiffMUdiaryday_c,IIdiffMUprovided_support_partner_cwp)	0.13	[-0.34, 0.61]	1.001	3660	2662
coupleID: cor(IIdiffMUprovided_support_actor_cwp,IIdiffMUprovided_support_partner_cwp)	0.43	[-0.11, 0.86]	1.002	1956	2850
coupleID:diaryday: sd(Intercept)	0.17	[0.04, 0.23]	1.018	220	129
Residual Structure					
sigma	0.92	[0.90, 0.94]	1.012	376	640

1.61 Extract Full APIM Random Effects Variance-Covariance Matrix

We can rotate the random effects structure back to obtain the full APIM Random effects variance-covariance matrix (Kenny & Ackerman, 2023). Just as we would in a SEM model with equality constraints!

For example, the within-person variance of the intercept is:

$$var(Intercept) + var(Idiff)$$

and the cross-person covariance of both partners' intercepts is:

$$var(Intercept) - var(Idiff)$$

Note: This only holds true when the means and Idiffs are uncorrelated. This is why we put them in separate random effects blocks.

1.62 Extract Full APIM Random Effects Variance-Covariance Matrix

```
# Custom function
rotate_apim_covariance <- function(
  fit,
  gr = "coupleID",
  Idiff = "Idiff"
) {
  random_effects <- fit$ranef
  random_effects <- random_effects[random_effects$group == gr,]

  SUM <- random_effects$coef[!grepl(Idiff, random_effects$coef)]
  DIFF <- random_effects$coef[grepl(Idiff, random_effects$coef)]

  varcor <- VarCorr(fit)[[gr]]

  within_person_covariance_matrix <- varcor$cov[SUM,'Estimate',SUM] + varcor$cov[DIFF,'Estimate',DIFF]
  cross_person_covariance_matrix <- varcor$cov[SUM,'Estimate',SUM] - varcor$cov[DIFF,'Estimate',DIFF]

  p <- nrow(within_person_covariance_matrix)

  Full <- rbind(
    cbind(within_person_covariance_matrix, cross_person_covariance_matrix),
    cbind(cross_person_covariance_matrix, within_person_covariance_matrix)
  )

  # nice labels
  base <- SUM
  rn <- c(paste0("PartnerA_", base), paste0("PartnerB_", base))
  colnames(Full) <- rn
  rownames(Full) <- rn
}
```

```

return(list(within_person_covariance_matrix=within_person_covariance_matrix,
  ↪ cross_person_covariance_matrix=cross_person_covariance_matrix, full_covariance_matrix=Full))
})

a <- rotate_apim_covariance(model_apim_ind_long)

```

1.63 Extract Full APIM Random Effects Variance-Covariance Matrix: Within-Person Matrix

```

print_df(a$within_person_covariance_matrix,
  font_size = font_size)

```

	Intercept	diaryday_c	provided_support_actor_cwp	provided_support_partner_cwp
Intercept	1.2048891	0.0116371	0.1017152	0.0193292
diaryday_c	0.0116371	0.0003188	0.0010799	0.0002998
provided_support_actor_cwp	0.1017152	0.0010799	0.0236386	0.0050707
provided_support_partner_cwp	0.0193292	0.0002998	0.0050707	0.0112751

1.64 Extract Full APIM Random Effects Variance-Covariance Matrix: Cross-Person Matrix

```

print_df(a$cross_person_covariance_matrix,
  font_size = font_size)

```

	Intercept	diaryday_c	provided_support_actor_cwp	provided_support_partner_cwp
Intercept	-0.2722670	-0.0024934	-0.0653774	0.0044172
diaryday_c	-0.0024934	-0.0000441	-0.0005912	0.0000889
provided_support_actor_cwp	-0.0653774	-0.0005912	-0.0181144	-0.0019421

provided_support_partner_cwp	0.0044172	0.0000889	-0.0019421	0.0027529
------------------------------	-----------	-----------	------------	-----------

1.65 Extract Full APIM Random Effects Variance-Covariance Matrix: Full Matrix

```
print_df(a$full_covariance_matrix,
  font_size = font_size)
```

	PartnerA_Intercept	PartnerA_diaryday_c	PartnerA_provided_support_actor_cwp	PartnerA_provided_support_partner_cwp	PartnerB_Intercept
PartnerA_Intercept	1.2048891	0.0116371	0.1017152	0.0193292	
PartnerA_diaryday_c	0.0116371	0.0003188	0.0010799	0.0002998	
PartnerA_provided_support_actor_cwp	0.1017152	0.0010799	0.0236386	0.0050707	
PartnerA_provided_support_partner_cwp	0.0193292	0.0002998	0.0050707	0.0112751	
PartnerB_Intercept	-0.2722670	-0.0024934	-0.0653774	0.0044172	
PartnerB_diaryday_c	-0.0024934	-0.0000441	-0.0005912	0.0000889	
PartnerB_provided_support_actor_cwp	-0.0653774	-0.0005912	-0.0181144	-0.0019421	
PartnerB_provided_support_partner_cwp	0.0044172	0.0000889	-0.0019421	0.0027529	

1.66 Convert to correlation matrix of random effects

```
# Custom function
sd_cor <- function(Sigma) {
  sds <- sqrt(diag(Sigma))
  cors <- cov2cor(Sigma)
  list(sd = sds, cor = cors)
}

full <- sd_cor(a$full_covariance_matrix)
```

1.67 Random effects SDs

```
print_df(as.data.frame(round(full$sd, 3)),
         font_size = font_size)
```

	round(full\$sd, 3)
PartnerA_Intercept	1.098
PartnerA_diaryday_c	0.018
PartnerA_provided_support_actor_cwp	0.154
PartnerA_provided_support_partner_cwp	0.106
PartnerB_Intercept	1.098
PartnerB_diaryday_c	0.018
PartnerB_provided_support_actor_cwp	0.154
PartnerB_provided_support_partner_cwp	0.106

1.68 Random effect correlation matrix

```
print_df(as.data.frame(round(full$cor, 3)),
         font_size = font_size)
```

	PartnerA_Intercept	PartnerA_diaryday_c	PartnerA_provided_support_actor_cwp	PartnerA_provided_support_partner_cwp	PartnerB_Intercept
PartnerA_Intercept	1.000	0.594	0.603	0.166	
PartnerA_diaryday_c	0.594	1.000	0.393	0.158	
PartnerA_provided_support_actor_cwp	0.603	0.393	1.000	0.311	
PartnerA_provided_support_partner_cwp	0.166	0.158	0.311	1.000	
PartnerB_Intercept	-0.226	-0.127	-0.387	0.038	1.000
PartnerB_diaryday_c	-0.127	-0.138	-0.215	0.047	
PartnerB_provided_support_actor_cwp	-0.387	-0.215	-0.766	-0.119	
PartnerB_provided_support_partner_cwp	0.038	0.047	-0.119	0.244	

References

- Barr, D. J., Levy, R., Scheepers, C., & Tily, H. J. (2013). Random effects structure for confirmatory hypothesis testing: Keep it maximal. *Journal of Memory and Language*, 68(3), 255–278. <https://doi.org/10.1016/j.jml.2012.11.001>
- Bolger, N., Laurenceau, J.-P., & DiGiovanni, A. (2025). Unified analysis model for indistinguishable and distinguishable dyads. *Innovations in Interpersonal Relationships and Health Research: Advancing the Integration of Interdisciplinary Approaches to Dyadic Behavior Change*. <https://doi.org/10.17605/OSF.IO/WYDCJ>
- Bürkner, P.-C. (2024). *Brms: Bayesian regression models using stan*. <https://github.com/paul-buerkner/brms>
- del Rosario, K. S., & West, T. V. (2025). A Practical Guide to Specifying Random Effects in Longitudinal Dyadic Multilevel Modeling. *Advances in Methods and Practices in Psychological Science*, 8(3), 25152459251351286. <https://doi.org/10.1177/25152459251351286>
- Gabry, J., Češnovar, R., Johnson, A., & Bröder, S. (2024). *Cmdstanr: R interface to CmdStan*. <https://mc-stan.org/cmdstanr/>
- Guo, J., Gabry, J., Goodrich, B., Johnson, A., Weber, S., & Badr, H. S. (2025). *Rstan: R interface to stan*. <https://mc-stan.org/rstan/>
- Hartig, F. (2024). *DHARMA: Residual diagnostics for hierarchical (multi-level / mixed) regression models*. <http://florianhartig.github.io/DHARMA/>
- Iida, M., Seidman, G., & Shrout, P. E. (2018). Models of interdependent individuals versus dyadic processes in relationship research. *Journal of Social and Personal Relationships*, 35(1), 59–88. <https://doi.org/10.1177/0265407517725407>
- Kenny, D. A., & Ackerman, R. A. (2023). *Estimation of random effects with over-time dyadic data using multilevel modeling: The sum and difference method*. OSF Preprints. <https://doi.org/10.31219/osf.io/fju72>
- Kenny, D. A., & Cook, W. (1999). Partner effects in relationship research: Conceptual issues, analytic difficulties, and illustrations. *Personal Relationships*, 6(4), 433–448. <https://doi.org/10.1111/j.1475-6811.1999.tb00202.x>
- Kenny, D. A., & Kashy, D. A. (2011). Dyadic Data Analysis Using Multilevel Modeling. In *Handbook of Advanced Multilevel Analysis*. Routledge.
- Kenny, D. A., Kashy, D. A., & Cook, W. L. (2006). *Dyadic data analysis*. Guilford Press.
- Lüdtke, D., Makowski, D., Ben-Shachar, M. S., Patil, I., Wiernik, B. M., Bacher, E., & Thériault, R. (2025). *Easystats: Framework for easy statistical modeling, visualization, and reporting*. <https://easystats.github.io/easystats/>
- Stadler, G., Scholz, U., Bolger, N., Shrout, P. E., Knoll, N., & Lüscher, J. (2023). How is companionship related to romantic partners' affect, relationship satisfaction, and health behavior? Using a longitudinal dyadic score model to understand daily and couple-level effects of a dyadic predictor. *Applied Psychology: Health and Well-Being*, 15(4), 1530–1554. <https://doi.org/10.1111/aphw.12450>
- Vehtari, A., Gabry, J., Magnusson, M., Yao, Y., Bürkner, P.-C., Paananen, T., & Gelman, A. (2024). *Loo: Efficient leave-one-out cross-validation and WAIC for bayesian models*. <https://mc-stan.org/loo/>
- Vuorre, M. (2023). *Bmlm: Bayesian multilevel mediation*. <https://github.com/mvuorre/bmlm/>
- Wickham, H. (2023). *Tidyverse: Easily install and load the tidyverse*. <https://tidyverse.tidyverse.org>