

Distinguishable and Exchangeable Dyads: Bayesian Multilevel Modelling

Cross-Sectional and Intensive Longitudinal APIM and DIM

Pascal M. Küng

2026-07-12

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 and show how to extract dyad-level variance–covariance components.

Interactive HTML version (recommended): The tutorial is best viewed as [interactive slides](#). This PDF is a static archival version for indexing, citation, and long-term reference.

DOI: [10.5281/zenodo.17400655](https://doi.org/10.5281/zenodo.17400655)

Suggested citation: Küng, P. M. (2026). *Distinguishable and Exchangeable Dyads: Bayesian Multilevel Modelling*. University of Zurich.

The [GitHub repository](#) contains the source, helpers, model code, and quick guides.

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 Quick Access

For readers who want the model code and reporting tables without the full tutorial:

- [Distinguishable dyads quick guide](#)
- [Exchangeable dyads quick guide](#)

0.3 Preparing Dyadic Data with `interdep`

`interdep` prepares dyadic data for multilevel models (Küng, 2026). It:

- validates long-format dyadic data
- supports cross-sectional and intensive longitudinal designs
- creates model-ready variables for APIM, DIM, and undirected DSM specifications
- handles distinguishable, exchangeable, and mixed dyad compositions

Install the development version from GitHub with [pak](#):

```
# install.packages("pak")
pak::pak("Pascal-Kueng/interdep")
```

[Getting started](#) · [Function reference](#) · [Source code](#)

0.4 Readable Names in This Tutorial

`interdep` deliberately returns explicit, self-documenting names. After data preparation, this tutorial drops the generated `.i_` prefix. For longitudinal predictors it otherwise retains the package’s component order (for example, `cwp_actor` and `cbp_partner`) and shortens `within_dyad_deviation` to `dev`. The remaining aliases are:

General and cross-sectional aliases

Canonical <code>interdep</code> name	Tutorial alias	Interpretation
<code>.i_is_female_x_male_female</code>	<code>is_female</code>	female member of a female-male dyad
<code>.i_is_female_x_male_male</code>	<code>is_male</code>	male member of a female-male dyad
<code>.i_diff_female_x_male_arbitrary</code>	<code>diff</code>	female-male exchangeability contrast
<code>.i_diff_assumed_exchangeable_arbitrary</code>	<code>diff</code>	role-free exchangeability contrast
<code>.i_communication_actor</code>	<code>communication_actor</code>	actor’s communication score
<code>.i_communication_partner</code>	<code>communication_partner</code>	partner’s communication score
<code>.i_communication_dyad_mean_gm</code>	<code>communication_dyad_mean_gmc</code>	grand-mean-centered dyad mean
<code>.i_communication_within_dyad_deviation</code>	<code>communication_dyad_dev</code>	member’s within-dyad deviation

0.5 Applying the Tutorial Aliases

To keep these aliases consistent, this tutorial uses `rename_interdep_for_tutorial()`, defined in [00_R_Functions/PrepareData.R](#). The helper renames only generated columns that are present:

```
model_data <- prepared_data |>
  rename_interdep_for_tutorial(
    predictor = "provided_support"
  )
```

The mapping is cosmetic: values and statistical meanings are unchanged. Apply it only to the final model-ready output; continue using canonical names while calling `interdep` preparation functions.

0.6 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(interdep)
library(brms)
library(glmmTMB)
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'))

print_apim_preview <- function(df) {
  if (knitr::is_latex_output()) {
    preview_cols <- c(
      "userID", "coupleID", "gender", "is_male", "is_female",
      "satisfaction", "communication_actor", "communication_partner"
    )
    df <- df |>
      dplyr::select(dplyr::any_of(preview_cols))
  }
  print_df(head(df))
}

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

1 Distinguishable Dyads

2 Distinguishable Dyads - Cross-Sectional APIM

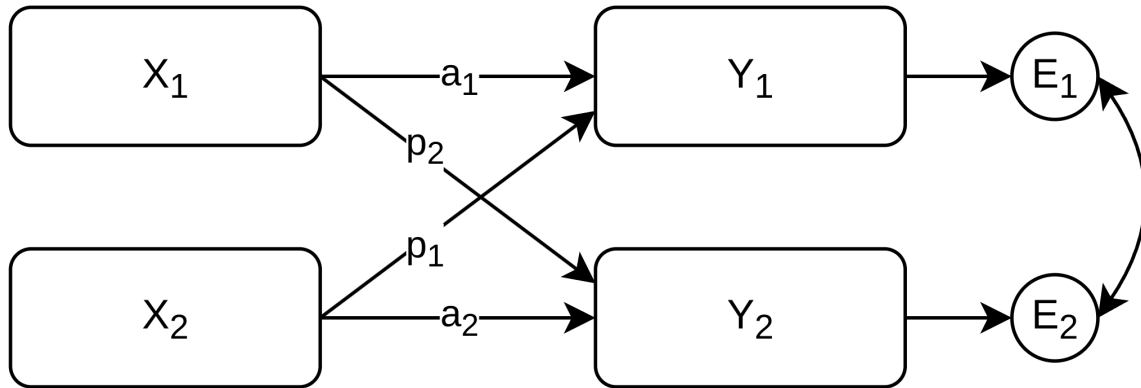
2.1 Distinguishable Dyads - Data: Simulated Dyads

```
print_df(head(df))
```

userID	coupleID	gender	comm	satisfaction
1 1	1	female	4.851107	4.841332
1 2	1	male	6.004533	5.577165
2 1	2	female	5.881321	7.248741
2 2	2	male	6.433723	6.960293

3 1	3	female	4.283971	6.928699
3 2	3	male	5.516060	6.077688

2.2 Distinguishable Dyads - Cross-Sectional APIM



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

2.3 Distinguishable Dyads - Preparing Data

```

df_apim <- prepare_interdep_data(
  data = df,
  group = coupleID,
  member = userID,
  role = gender,
  predictors = communication,
  model_type = "apim",
  include_compositions = "female-male"
) %>%
  rename_interdep_for_tutorial(
    predictor = "communication"
  )

print_apim_preview(df_apim)

```

2.4 Distinguishable Dyads - Preparing Data

userID	coupleID	gender	is male	is female	satisfaction	comm actor	comm partner
1 1	1	female	0	1	4.841332	4.851107	6.004533
1 2	1	male	1	0	5.577165	6.004533	4.851107
2 1	2	female	0	1	7.248741	5.881321	6.433723
2 2	2	male	1	0	6.960293	6.433723	5.881321
3 1	3	female	0	1	6.928699	4.283971	5.516060
3 2	3	male	1	0	6.077688	5.516060	4.283971

For cross-sectional APIMs, `communication_actor` and `communication_partner` contain the original communication scores. The earlier version of this tutorial grand-mean centered both columns manually. Removing that centering does not change either slope or fitted value, but the role-specific intercepts now refer to `communication` = 0 rather than average communication.

2.5 Distinguishable Dyads - Fitting the Model in glmmTMB

- glmmTMB models residual variance/dispersion via `dispformula`, but residual covariance is handled through random-effect covariance structures.
- For Gaussian mixed models, `dispformula = ~ 0` fixes the residual variance to a tiny value. This pushes the Gaussian residual variance into the random-effect block.
- In this Gaussian dyadic model, the `us(0 + is_male + is_female | coupleID)` block therefore represents the 2x2 residual variance-covariance structure between partners.
- For non-Gaussian models, this is not an ordinary residual covariance on the response scale. The same random-effect trick adds latent Gaussian heterogeneity/correlation on the linear predictor scale.
- Do **not** use `dispformula = ~ 0` as a general non-Gaussian residual-covariance trick. For families with an estimable dispersion parameter, `dispformula` models that family-specific parameter via a log link; for families without one, such as Poisson or binomial, it is ignored. Partner-specific dispersion, e.g. `dispformula = ~ 0 + is_male + is_female`, is only appropriate when the family has a dispersion parameter and this is substantively intended.

2.6 Distinguishable Dyads - Fitting the Model in glmmTMB

```
1 model_dist_apim_glmmTMB <- glmmTMB(  
2   satisfaction ~  
3  
4   # Remove global intercept, introduce male and female intercepts.  
5   0 + is_male + is_female +  
6  
7   # Actor effect for male and female  
8   communication_actor:is_male +  
9   communication_actor:is_female +  
10  
11  # Partner effect for male and female  
12  communication_partner:is_male +  
13  communication_partner:is_female +  
14  
15  # Modelling residual non-independence with an unstructured 2x2 covariance.  
16  us(0 + is_male + is_female | coupleID)  
17  
18  # Gaussian response:  
19  # dispformula = ~ 0 fixes residual variance to a tiny value,  
20  # pushing the Gaussian residual variance into this random-effect block.  
21  , dispformula = ~ 0  
22  
23  # Non-Gaussian responses:  
24  # dispformula is family-specific dispersion, or ignored for families  
25  # without a dispersion parameter. Do not use ~0 as a residual trick.  
26  # If dispersion exists and should differ by partner, use e.g.:  
27  # dispformula = ~ 0 + is_male + is_female  
28  
29  , data = df_apim  
30  , family = gaussian  
31  )
```

2.7 Distinguishable Dyads - Fitting the Model in glmmTMB

Note that ‘unstructured’ is the default in glmmTMB, so instead of

```
us(0 + is_male + is_female | coupleID)
```

we could also remove `us` and simply write

```
(0 + is_male + is_female | coupleID).
```

An unstructured covariance matrix gives `is_male` and `is_female` their own variance and freely estimates their correlation. With more than two members (e.g., triads), it would estimate a separate correlation for every pair.

For two partners, the covariance matrix is:

$$\Sigma = \begin{pmatrix} \sigma_{female}^2 & \rho \sigma_{female} \sigma_{male} \\ \rho \sigma_{female} \sigma_{male} & \sigma_{male}^2 \end{pmatrix}.$$

2.8 Distinguishable Dyads - Fitting the Model in glmmTMB

```
Family: gaussian ( identity )
Formula:      satisfaction ~ 0 + is_male + is_female + communication_actor:is_male +
      communication_actor:is_female + communication_partner:is_male +
      communication_partner:is_female + us(0 + is_male + is_female |      coupleID)
Dispersion:      ~0
Data: df_apim

      AIC      BIC    logLik -2*log(L)  df.resid
3776.7    3821.7   -1879.3    3758.7     1091

Random effects:

Conditional model:
Groups   Name      Variance Std.Dev. Corr
coupleID is_male    2.559    1.600
          is_female 1.682    1.297    0.51
Number of obs: 1100, groups:  coupleID, 550

Conditional model:
              Estimate Std. Error z value Pr(>|z|)
is_male      -5.46305    0.28124  -19.42  < 2e-16 ***
is_female    -3.91806    0.22802  -17.18  < 2e-16 ***
is_male:communication_actor    1.84363    0.04931   37.39  < 2e-16 ***
is_female:communication_actor    1.59248    0.04112   38.72  < 2e-16 ***
is_male:communication_partner    0.16765    0.05072    3.31 0.000949 ***
is_female:communication_partner    0.31084    0.03997    7.78 7.48e-15 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

2.9 glmmTMB: Current limitation

Usually, simulation-based tools like DHARMa (Hartig, 2026) are great for residual diagnostics in complex (generalized) MLMs.

However, when we remove the native residuals `dispformula = ~ 0` the residual is, in practice, set to a tiny constant. This confuses model diagnostic packages!

Therefore, residuals need to be manually simulated and compared.

2.10 Distinguishable Dyads - Fitting the Model in brms

In brms, we can use brms-provided residual covariance structures directly. We do not have to set residuals internally to zero.

```

1 formula_dist_apim_b <- bf(
2   satisfaction ~
3
4   # Remove global intercept, introduce male and female intercepts.
5   0 + is_male + is_female +
6
7   # Actor effect for male and female
8   communication_actor:is_male +
9   communication_actor:is_female +
10
11  # Partner effect for male and female
12  communication_partner:is_male +
13  communication_partner:is_female +
14
15  # Gender indexes the two within-couple residual positions.
16  unstr(time = gender, gr = coupleID),
17
18  # Allowing two distinct residual variances for males vs females:
19  sigma = ~ 0 + is_male + is_female
20
21 )
22
23 priors_dist_apim_b <- c(
24   # prior(normal(2, 3), class = "Intercept"),
25   prior(normal(0, 5), class = "b"),
26   prior(student_t(3, 0, 1.5), class = "b", dpar = "sigma"),
27   prior(lkj(2), class = "cortime")
28 )
29
30 model_dist_apim_b <- brm(
31   formula = formula_dist_apim_b,
32   data = df_apim,
33   family = gaussian(link = identity),
34   prior = priors_dist_apim_b,
35   chains = 4,
36   cores = 4,
37   iter = 2000,
38   warmup = 1000,
39   seed = 123,
40   file = file.path('brms_cache', 'example1_dist_apim_b') # Cache the model
41 )

```

2.11 Distinguishable Dyads - Fitting the Model in brms

```

Family: gaussian
Links: mu = identity; sigma = log
Formula: satisfaction ~ 0 + is_male + is_female + communication_actor:is_male +
↪ communication_actor:is_female + communication_partner:is_male + communication_partner:is_female +
↪ unstr(time = gender, gr = coupleID)
sigma ~ 0 + is_male + is_female
Data: df_apim (Number of observations: 1100)
Draws: 4 chains, each with iter = 2000; warmup = 1000; thin = 1;
total post-warmup draws = 4000

```

Correlation Structures:

	Estimate	Est.Error	l-95% CI	u-95% CI	Rhat	Bulk_ESS	Tail_ESS
cortime(female,male)	0.51	0.03	0.44	0.57	1.00	2895	2470

Regression Coefficients:

	Estimate	Est.Error	l-95% CI	u-95% CI	Rhat	Bulk_ESS	Tail_ESS
is_male	-5.44	0.28	-5.99	-4.88	1.00	2458	2617
is_female	-3.90	0.23	-4.37	-3.47	1.00	2401	2641
is_male:communication_actor	1.84	0.05	1.75	1.94	1.00	2678	2997

is_female:communication_actor	1.59	0.04	1.51	1.67	1.00	2371	2549
is_male:communication_partner	0.17	0.05	0.07	0.26	1.00	2555	2726
is_female:communication_partner	0.31	0.04	0.23	0.39	1.00	2398	2724
sigma_is_male	0.47	0.03	0.42	0.53	1.00	3555	2827
sigma_is_female	0.26	0.03	0.21	0.33	1.00	3317	2788

Draws were sampled using `sample(hmc)`. For each parameter, Bulk_ESS and Tail_ESS are effective sample size measures, and Rhat is the potential scale reduction factor on split chains (at convergence, Rhat = 1).

2.12 Distinguishable Dyads - Fitting the Model in brms

For families that have no residuals, this does not work and we need to model the non-independence in the random effects:

```

1 formula_dist_apim_c <- bf(
2   as.integer(satisfaction + 6) ~
3
4   # Remove global intercept, introduce male and female intercepts.
5   0 + is_male + is_female +
6
7   # Actor effect for male and female
8   communication_actor:is_male +
9   communication_actor:is_female +
10
11  # Partner effect for male and female
12  communication_partner:is_male +
13  communication_partner:is_female +
14
15  # Modelling residual correlation as random effects, because we cannot specify
16  # a sigma parameter, as this distribution has no residuals.
17
18  (0 + is_male + is_female | coupleID)
19
20  # sigma ~ 0 + is_male + is_female
21 )
22
23 priors_dist_apim_c <- c(
24   prior(normal(0, 5), class = "b"),
25   prior(student_t(3, 0, 1.5), class = "sd"),
26   prior(lkj(2), class = "cor")
27 )
28
29 model_dist_apim_c <- brm(
30   formula = formula_dist_apim_c,
31   data = df_apim,
32   family = poisson(),
33   prior = priors_dist_apim_c,
34   chains = 4,
35   cores = 4,
36   iter = 2000,
37   warmup = 1000,
38   seed = 123,
39   file = file.path('brms_cache', 'example1_dist_apim_c') # Cache the model
40 )

```

2.13 Distinguishable Dyads - Fitting the Model in brms

```

Family: poisson
Links: mu = log
Formula: as.integer(satisfaction + 6) ~ 0 + is_male + is_female + communication_actor:is_male +
↪ communication_actor:is_female + communication_partner:is_male + communication_partner:is_female + (0 +
↪ is_male + is_female | coupleID)
Data: df_apim (Number of observations: 1100)
Draws: 4 chains, each with iter = 2000; warmup = 1000; thin = 1;

```



```

total post-warmup draws = 4000

Multilevel Hyperparameters:
~coupleID (Number of levels: 550)

```

	Estimate	Est.Error	1-95% CI	u-95% CI	Rhat	Bulk_ESS	Tail_ESS
sd(is_male)	0.01	0.01	0.00	0.04	1.00	2950	1558
sd(is_female)	0.01	0.01	0.00	0.03	1.00	3336	1803
cor(is_male,is_female)	0.02	0.44	-0.80	0.81	1.00	6459	2872

```

Regression Coefficients:

```

	Estimate	Est.Error	1-95% CI	u-95% CI	Rhat	Bulk_ESS	Tail_ESS
is_male	1.29	0.06	1.18	1.41	1.00	4332	2701
is_female	1.52	0.06	1.40	1.63	1.00	4999	2882
is_male:communication_actor	0.18	0.01	0.16	0.20	1.00	5510	3239
is_female:communication_actor	0.14	0.01	0.12	0.16	1.00	5619	2960
is_male:communication_partner	0.02	0.01	-0.00	0.03	1.00	5119	3116
is_female:communication_partner	0.03	0.01	0.01	0.05	1.00	6257	2679

Draws were sampled using sample(hmc). For each parameter, Bulk_ESS and Tail_ESS are effective sample size measures, and Rhat is the potential scale reduction factor on split chains (at convergence, Rhat = 1).

2.14 Distinguishable Dyads - Fitting the Model in glmmTMB

```

model_glmmTMB_dist_apim_c <- glmmTMB(
  as.integer(satisfaction + 6) ~

  # Remove global intercept, introduce male and female intercepts.
  0 + is_male + is_female +

  # Actor effect for male and female
  communication_actor:is_male +
  communication_actor:is_female +

  # Partner effect for male and female
  communication_partner:is_male +
  communication_partner:is_female +

  # Modelling residual correlation as random effects, because we cannot specify
  # a sigma parameter, as this distribution has no residuals.

  (0 + is_male + is_female | coupleID)
, data = df_apim
, family = poisson()
)

```

2.15 Distinguishable Dyads - Fitting the Model in glmmTMB

```

Family: poisson ( log )
Formula:      as.integer(satisfaction + 6) ~ 0 + is_male + is_female + communication_actor:is_male +
      communication_actor:is_female + communication_partner:is_male +
      communication_partner:is_female + (0 + is_male + is_female |      coupleID)
Data: df_apim

```

	AIC	BIC	logLik	-2*log(L)	df.resid
	NA	NA	NA	NA	1091

Random effects:

Conditional model:

Groups	Name	Variance	Std.Dev.	Corr
coupleID	is_male	1.331e-10	1.154e-05	
	is_female	2.502e-12	1.582e-06	-1.00

Number of obs: 1100, groups: coupleID, 550

```
Conditional model:
              Estimate Std. Error z value Pr(>|z|)
is_male      1.294677   0.057667  22.451 < 2e-16 ***
is_female    1.516475   0.055264  27.441 < 2e-16 ***
is_male:communication_actor 0.180139 0.009598  18.768 < 2e-16 ***
is_female:communication_actor 0.143973 0.009580  15.028 < 2e-16 ***
is_male:communication_partner 0.015476 0.010003   1.547 0.12183
is_female:communication_partner 0.028344 0.009180   3.088 0.00202 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

2.16 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:

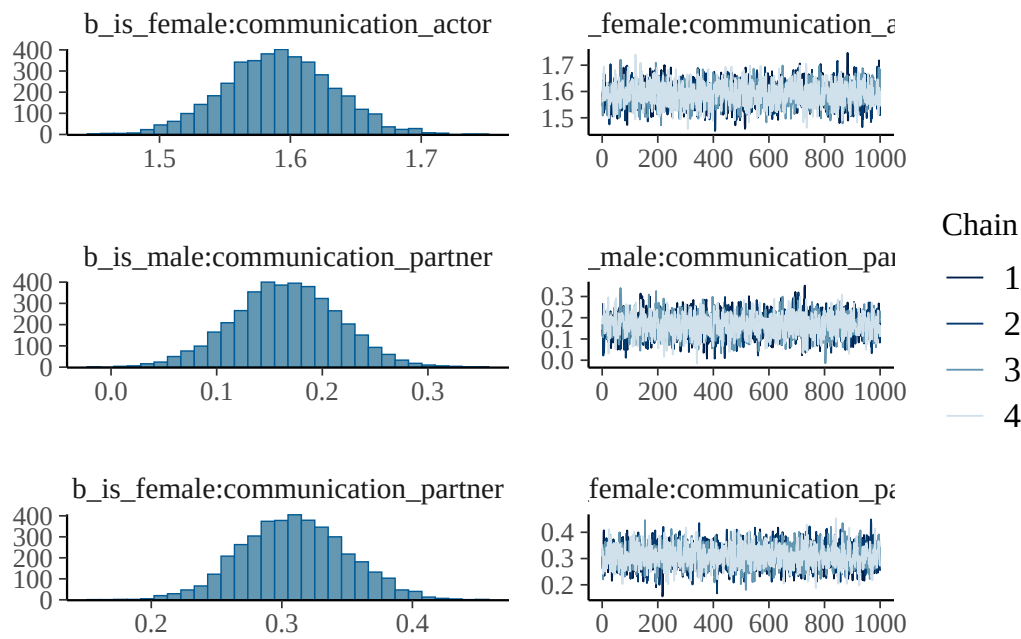
0 of 4000 iterations saturated the maximum tree depth of 10.

Energy:

E-BFMI indicated no pathological behavior.

2.17 Distinguishable Dyads - Check Model Convergence and Fit

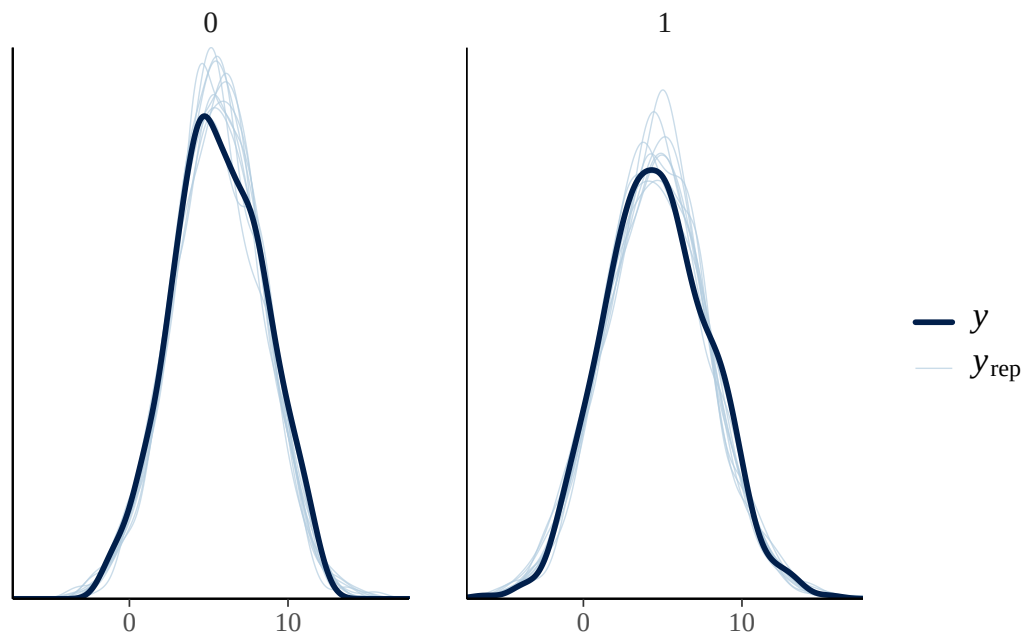
```
plot(model_dist_apim_b, ask = FALSE)
```



2.18 Distinguishable Dyads - Check Model Convergence and Fit

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

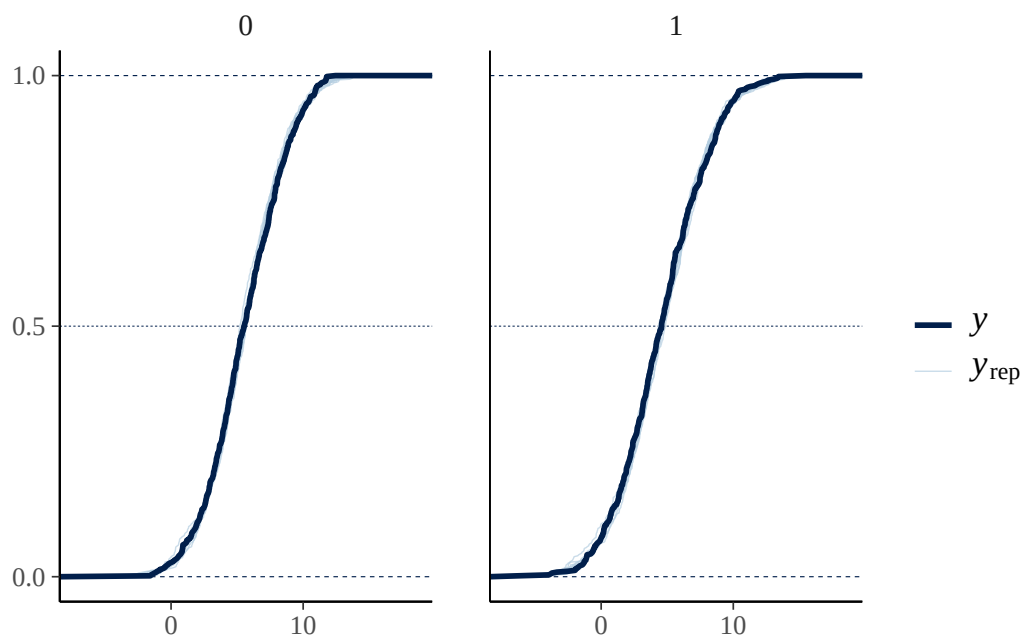
Using 10 posterior draws for ppc type 'dens_overlay_grouped' by default.



2.19 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.



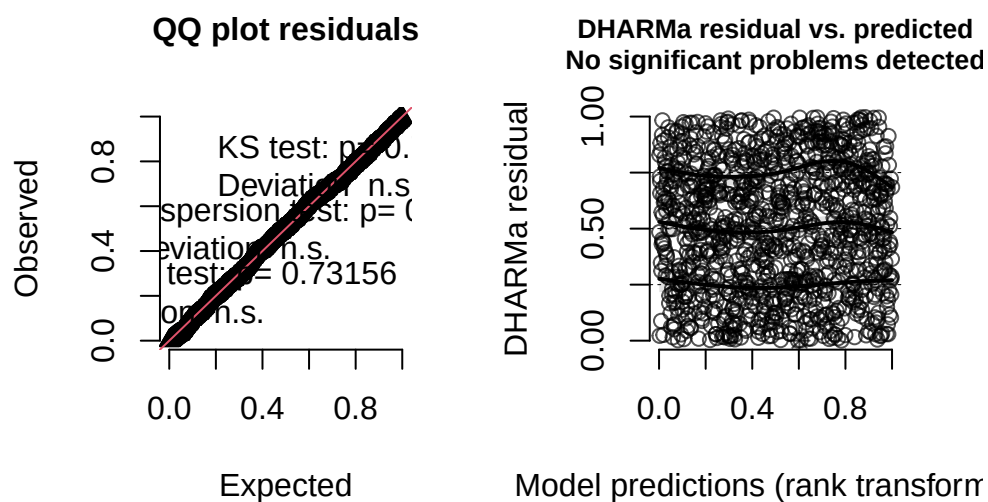
2.20 Distinguishable Dyads - Check Model Convergence and Fit

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

2.21 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)
```

DHARMA residual



2.22 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).

2.23 Excursion: Generalized Dyadic Models

For Gaussian outcomes, residual covariance has a direct response-scale interpretation:

- partner-specific residual variances
- partner residual correlations
- same-day residual covariance in ILD data

For non-Gaussian outcomes, the same syntax usually does **not** estimate an ordinary residual covariance.

Instead, random-effect covariance terms represent latent Gaussian heterogeneity on the link scale.

2.24 Excursion: Count vs Binary Outcomes

For count outcomes, latent partner covariance is often useful and can be estimable:

- Poisson or negative-binomial models can include dyad-level latent covariance terms
- estimates live on the log-link scale
- covariance parameters need diagnostics and sensitivity checks

For dichotomous outcomes, the same idea is much harder:

- each partner contributes only 0 or 1 per occasion
- many latent variance-correlation combinations can imply similar pair probabilities
- same-day binary covariance is especially weakly identified

2.25 Excursion: Practical Warning

Binary dyadic covariance models can converge mechanically while still being unreliable.

Watch for:

- variance estimates near zero or extremely large
- correlations near -1 or 1
- low ESS, divergent transitions, or strong prior sensitivity
- different software giving very different covariance estimates

2.26 Excursion: Practical Strategy

Practical default:

- estimate simpler structures first
- use informative priors if fitting latent binary covariance in brms
- treat same-day binary partner covariance as a specialized model, not a routine extension
- if the main goal is fixed effects, consider simpler random effects or robust/clustered inference

2.27 Distinguishable Dyads - Results

	Est.	95% CI	Rhat	Bulk ESS	Tail ESS
Fixed Effects					
is male	-5.44	[-5.99, -4.88]	1.003	2458	2617
is female	-3.90	[-4.37, -3.47]	1.001	2401	2641
is male: comm actor	1.84	[1.75, 1.94]	1.000	2678	2997
is female: comm actor	1.59	[1.51, 1.67]	1.001	2371	2549
is male: comm partner	0.17	[0.07, 0.26]	1.001	2555	2726
is female: comm partner	0.31	[0.23, 0.39]	1.000	2398	2724
Residual Structure					
cortime(female, male)	0.51	[0.44, 0.57]	1.000	2895	2470
sigma is male	1.60	[1.51, 1.71]	1.000	3555	2827
sigma is female	1.30	[1.23, 1.39]	1.001	3317	2788

NOTE: brms estimates sigma on the log scale. In the tables, sigmas are exponentiated to show residual SDs on the response scale, like glmmTMB output.

3 Distinguishable Dyads

4 Distinguishable Dyads - Intensive Longitudinal APIM (L-APIM)

4.1 Distinguishable Dyads - L-APIM - Simulated Dataset

```
print_df(head(df_long))
```

userID	coupleID	day	gender	closeness	support
1 1	1	0	female	5.659395	4.733482
1 1	1	1	female	7.249842	5.019950
1 1	1	2	female	6.040587	5.365179
1 1	1	3	female	6.898253	5.148704
1 1	1	4	female	5.350049	3.702039
1 1	1	5	female	6.191282	4.844602

```
df_long_apim <- prepare_interdep_data(  
  data = df_long,  
  group = coupleID,  
  member = userID,  
  role = gender,  
  time = diaryday,  
  predictors = provided_support,  
  model_type = "apim"  
) %>%
```

```

rename_interdep_for_tutorial(
  # Keep interdep's cwp_actor/cbp_partner component ordering.
  predictor = "provided_support"
) %>%
mutate(
  diaryday_c = diaryday - mean(diaryday)
)

```

userID	coupleID	day	day c	gender	is male	is female	closeness	support WP actor	support WP partner	support BP actor	support BP partner
31 1	31	0	-27	female	0	1	5.47	0.41	1.54	0.65	0.58
31 1	31	1	-26	female	0	1	7.69	0.51	1.4	0.65	0.58
31 1	31	2	-25	female	0	1	5.83	0.49	0.79	0.65	0.58
31 1	31	3	-24	female	0	1	6.55	1.38	-0.15	0.65	0.58
...	...	...	...	...	...	...	...	...	...	...	...
31 2	31	0	-27	male	1	0	5.19	1.54	0.41	0.58	0.65
31 2	31	1	-26	male	1	0	6.96	1.4	0.51	0.58	0.65
31 2	31	2	-25	male	1	0	6.9	0.79	0.49	0.58	0.65
31 2	31	3	-24	male	1	0	7.59	-0.15	1.38	0.58	0.65

4.2 Distinguishable Dyads - L-APIM - Centering

With repeated measures we need to center variables in order to not confound 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

Although the data contain couple, person, and day structure, the APIM is written at the person-day level after decomposing predictors into between-person and within-person components. Dyad-level dependence is still modelled explicitly through partner-specific random effects and residual covariance structures.

4.3 Distinguishable Dyads - L-APIM - Centering

`prepare_interdep_data()` performs this predictor decomposition before creating the actor and partner columns. The centered `diaryday_c` variable above is a separate time transformation for the models, not part of the dyadic predictor construction.

4.4 Distinguishable Dyads - L-APIM - Model nlme

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

This package is old, but still often used and cited. So here, we briefly show it to demonstrate that we can achieve the same model with much more modern and powerful methods.

```

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_cbp_actor +

```

```

is_male:provided_support_cbp_partner +

is_female:provided_support_cbp_actor +
is_female:provided_support_cbp_partner +

# Within-Person APIM
is_male:provided_support_cwp_actor +
is_male:provided_support_cwp_partner +

is_female:provided_support_cwp_actor +
is_female:provided_support_cwp_partner,

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_cwp_actor
#+ is_male:provided_support_cwp_partner
#+ is_female:provided_support_cwp_actor
#+ is_female:provided_support_cwp_partner
| 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)

```

4.5 Distinguishable Dyads - L-APIM - Model glmmTMB

```

1 model_dist_apim_long_glmmTMB <- glmmTMB(
2   closeness ~ 0 + is_male + is_female +
3
4   # Time Slopes
5   is_male:diaryday_c +
6   is_female:diaryday_c +
7
8   # Between-Person APIM
9   is_male:provided_support_cbp_actor +
10  is_male:provided_support_cbp_partner +
11
12  is_female:provided_support_cbp_actor +
13  is_female:provided_support_cbp_partner +
14
15  # Within-Person APIM
16  is_male:provided_support_cwp_actor +
17  is_male:provided_support_cwp_partner +
18
19  is_female:provided_support_cwp_actor +
20  is_female:provided_support_cwp_partner +
21
22  # Accounting for non-independence between partners' means and trajectories
23  # and effect sensitivities via random effects:
24  (0 + is_male + is_female
25   + is_male:diaryday_c + is_female:diaryday_c
26   | coupleID) +
27
28  # Accounting for daily non-independence
29  us(0 + is_male + is_female | coupleID:diaryday)
30
31  # heteroscedastic residuals
32  , dispformula = ~ 0
33  , data = df_long_apim
34 )

```


4.6 Distinguishable Dyads - L-APIM - Model glmmTMB

```

Family: gaussian ( identity )
Formula:      closeness ~ 0 + is_male + is_female + is_male:diaryday_c + is_female:diaryday_c +
  is_male:provided_support_cbp_actor + is_male:provided_support_cbp_partner +
  is_female:provided_support_cbp_actor + is_female:provided_support_cbp_partner +
  is_male:provided_support_cwp_actor + is_male:provided_support_cwp_partner +
  is_female:provided_support_cwp_actor + is_female:provided_support_cwp_partner +
  (0 + is_male + is_female + is_male:diaryday_c + is_female:diaryday_c |
    coupleID) + us(0 + is_male + is_female | coupleID:diaryday)
Dispersion:      ~0
Data: df_long_apim

      AIC      BIC    logLik -2*log(L)  df.resid
30001.8  30184.4 -14975.9  29951.8    10975

Random effects:

Conditional model:
Groups          Name                Variance Std.Dev. Corr
coupleID        is_male              0.6813649 0.82545
                is_female            0.7929829 0.89050  0.19
                is_male:diaryday_c    0.0002637 0.01624  0.55  0.03
                is_female:diaryday_c  0.0002571 0.01603  0.21  0.54 -0.01
coupleID:diaryday is_male            1.2583395 1.12176
                is_female            0.5206603 0.72157  0.04
Number of obs: 11000, groups:  coupleID, 100; coupleID:diaryday, 5500

Conditional model:
                Estimate Std. Error z value Pr(>|z|)
is_male        4.486375   0.085960  52.19 < 2e-16 ***
is_female      5.736720   0.091763  62.52 < 2e-16 ***
is_male:diaryday_c -0.003258  0.001883  -1.73  0.08355 .
is_female:diaryday_c  0.010606  0.001717   6.18 6.46e-10 ***
is_male:provided_support_cbp_actor  1.128018  0.101922  11.07 < 2e-16 ***
is_male:provided_support_cbp_partner 0.298902  0.098817   3.02  0.00249 **
is_female:provided_support_cbp_actor 1.380753  0.105607  13.07 < 2e-16 ***
is_female:provided_support_cbp_partner 0.587254  0.108927   5.39 7.00e-08 ***
is_male:provided_support_cwp_actor  0.184479  0.020564   8.97 < 2e-16 ***
is_male:provided_support_cwp_partner 0.124548  0.020571   6.05 1.41e-09 ***
is_female:provided_support_cwp_actor 0.419162  0.013277  31.57 < 2e-16 ***
is_female:provided_support_cwp_partner 0.170127  0.013270  12.82 < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

4.7 Comparing outputs between nlme and glmmTMB

Let's compare the random effects outputs to see if the models are truly equivalent.

Residual variance in `nlme` must first be constructed from the output by taking the residual SD and multiplying it by the parameter estimate under the variance function for each partner.

`nlme`:

`glmmTMB`:

4.8 Distinguishable Dyads - L-APIM - Model brms

Equivalent `brms` model.

In contrast to `glmmTMB`, `brms` provides native variance-covariance structures to model residuals directly. Therefore, we do NOT set the residuals to zero!

```

1 formula_dist_apim_long_simple <- bf(
2   closeness ~ 0 + is_male + is_female +
3
4   # Time Slopes
5   is_male:diaryday_c +
6   is_female:diaryday_c +
7
8   # Between-Person APIM
9   is_male:provided_support_cbp_actor +
10  is_male:provided_support_cbp_partner +
11
12  is_female:provided_support_cbp_actor +
13  is_female:provided_support_cbp_partner +
14
15  # Within-Person APIM
16  is_male:provided_support_cwp_actor +
17  is_male:provided_support_cwp_partner +
18
19  is_female:provided_support_cwp_actor +
20  is_female:provided_support_cwp_partner +
21
22  # Accounting for non-independence between partners' means and trajectories
23  # and effect sensitivities via random effects:
24  (0 + is_male + is_female
25   + is_male:diaryday_c + is_female:diaryday_c
26   | coupleID) +
27
28  # Gender indexes the two partner residual positions within each couple-day.
29  unstr(time = gender, gr = coupleID:diaryday)
30
31  # heteroscedastic residuals
32  , sigma ~ 0 + is_male + is_female
33 )
34
35 # brms' defaults for group-level SDs and correlations are sensible weak defaults
36 # (half Student-t priors for SDs and LKJ priors for correlation matrices). Here we
37 # set weakly regularizing priors explicitly so the modelling assumptions are visible.
38
39 priors_dist_apim_long_simple <- c(
40   prior(normal(4, 2), class = "b", coef = "is_male"), # male intercept
41   prior(normal(4, 2), class = "b", coef = "is_female"), # female intercept
42   prior(normal(0, 2), class = "b"), # other beta coefficients
43   prior(exponential(1), class = "sd"),
44   prior(lkj(2), class = "cor"),
45   prior(normal(0, 0.7), class = "b", dpar = "sigma"),
46   prior(lkj(2), class = "cortime")
47 )
48
49 model_dist_apim_long_simple <- brm(
50   formula = formula_dist_apim_long_simple,
51   data = df_long_apim,
52   family = gaussian(link = identity), # student() is also often a nice robust option.
53   prior = priors_dist_apim_long_simple,
54   chains = 4,
55   cores = 4,
56   iter = 2000,
57   warmup = 1000,
58   seed = 123,
59   file = file.path('brms_cache', 'example1_dist_apim_long_simpel') # Cache the model
60 )

```

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

	Est.	95% CI	Rhat	Bulk ESS	Tail ESS
Fixed Effects					
is male	4.49	[4.32, 4.65]	1.007	324	794
is female	5.74	[5.56, 5.91]	1.009	355	656
is male: day c	0.00	[-0.01, 0.00]	1.007	1072	2113
is female: day c	0.01	[0.01, 0.01]	1.002	880	1778
is male: support BP actor	1.14	[0.91, 1.34]	1.006	489	791

is male: support BP partner	0.30	[0.10, 0.51]	1.004	553	716
is female: support BP actor	1.38	[1.18, 1.58]	1.005	507	1077
is female: support BP partner	0.60	[0.36, 0.81]	1.005	440	851
is male: support WP actor	0.18	[0.14, 0.23]	1.000	5328	3057
is male: support WP partner	0.12	[0.08, 0.16]	1.003	7228	3389
is female: support WP actor	0.42	[0.39, 0.45]	1.002	5990	3068
is female: support WP partner	0.17	[0.14, 0.20]	1.001	5829	3105
Random Effects					
sd(is male)	0.84	[0.73, 0.97]	1.010	520	1077
sd(is female)	0.92	[0.80, 1.06]	1.006	556	998
sd(is male: day c)	0.02	[0.01, 0.02]	1.001	1744	2637
sd(is female: day c)	0.02	[0.01, 0.02]	1.001	1358	2043
cor(is male, is female)	0.18	[-0.05, 0.35]	1.010	321	688
cor(is male, is male: day c)	0.51	[0.32, 0.67]	1.004	1528	1804
cor(is female, is male: day c)	0.03	[-0.21, 0.24]	1.002	1315	2204
cor(is male, is female: day c)	0.19	[-0.01, 0.38]	1.003	753	1691
cor(is female, is female: day c)	0.51	[0.33, 0.65]	1.002	1073	1832
cor(is male: day c, is female: day c)	-0.02	[-0.25, 0.22]	1.008	867	1748
cortime(female, male)	0.04	[0.01, 0.07]	1.000	6904	3067
Residual Structure					
sigma is male	1.12	[1.10, 1.14]	1.001	6996	2514
sigma is female	0.72	[0.71, 0.74]	0.999	6880	2868

Note: Sigma rows are shown on the response scale; they were exponentiated from brms' log-sigma parameterization.

4.10 Distinguishable Dyads - L-APIM - AR(1)?

In this model, same-day partner interdependence is handled by `unstr(time = gender, gr = coupleID:diaryday)`. In current univariate brms models, we cannot combine this residual correlation structure with `ar()` in the same formula.

glmmTMB currently also does not allow partner interdependence across time, but I have a fork of glmmTMB with added support and am currently working on integrating this with upstream glmmTMB.

4.11 Excursion: Dynamic Workaround

A dynamic MLM can account for temporal dependence by including lagged within-person centered outcomes as predictors (include both actor inertia and partner spillover).

But:

- The APIM coefficients become effects conditional on prior closeness, not total within-person associations.
- Lagged-outcome models can also suffer from dynamic-panel bias, often called Nickell bias (Nickell, 1981), because the lagged outcome is related to person/couple effects and prior residuals. This is especially problematic with short time series.
 - One solution for the Nickell bias is to use multilevel DSEM through MPLUS.

4.12 Maximal Model Specification (non-dynamic/no AR(1))

We add all random slopes and their correlations. This is neither converging in `nlme` nor `glmmTMB`.

```

1 formula_dist_apim_long_complex <- bf(
2   closeness ~
3
4   # Intercepts
5   0 + is_male + is_female +
6
7   # Time Slopes
8   is_male:diaryday_c +
9   is_female:diaryday_c +
10
11  # Between-Person APIM
12  is_male:provided_support_cbp_actor +
13  is_male:provided_support_cbp_partner +
14
15  is_female:provided_support_cbp_actor +
16  is_female:provided_support_cbp_partner +
17
18  # Within-Person APIM
19  is_male:provided_support_cwp_actor +
20  is_male:provided_support_cwp_partner +
21
22  is_female:provided_support_cwp_actor +
23  is_female:provided_support_cwp_partner +
24
25  # Accounting for non-independence between partners' means and trajectories
26  # and effect sensitivities via random effects:
27  (0 + is_male + is_female +
28
29   is_male:diaryday_c + is_female:diaryday_c +
30
31   is_male:provided_support_cwp_actor +
32   is_male:provided_support_cwp_partner +
33   is_female:provided_support_cwp_actor +
34   is_female:provided_support_cwp_partner
35   | coupleID ) +
36
37  # Accounting for daily non-independence
38  unstr(time = gender, gr = coupleID:diaryday),
39
40  # No more AR1. You could also test MA or ARMA.
41
42  sigma ~ 0 + is_male + is_female # heteroscedastic residuals
43 )
44
45 priors_dist_apim_long_complex <- c(
46   prior(normal(4, 2), class = "b", coef = "is_male"),
47   prior(normal(4, 2), class = "b", coef = "is_female"),
48   prior(normal(0, 2), class = "b"),
49   prior(exponential(1), class = "sd"),
50   prior(lkj(2), class = "cor"),
51   prior(normal(0, 0.7), class = "b", dpar = "sigma"),
52   prior(lkj(2), class = "cortime")
53 )
54
55
56 model_dist_apim_long_complex <- brm(
57   formula = formula_dist_apim_long_complex,
58   data = df_long_apim,
59   family = gaussian(link = identity),
60   prior = priors_dist_apim_long_complex,
61   chains = 4,
62   cores = 4,
63   iter = 2000,
64   warmup = 1000,
65   seed = 123,
66   file = file.path('brms_cache', 'example1_dist_apim_long_complex') # Cache the model
67 )

```

4.13 Maximal Model Specification: Results

	Est.	95% CI	Rhat	Bulk ESS	Tail ESS
Fixed Effects					
is male	4.49	[4.31, 4.67]	1.006	334	722
is female	5.73	[5.54, 5.92]	1.006	271	770
is male: day c	0.00	[-0.01, 0.00]	1.002	1249	1965
is female: day c	0.01	[0.01, 0.01]	1.005	840	2033
is male: support BP	1.12	[0.92, 1.32]	1.004	688	1529
actor					
is male: support BP	0.31	[0.10, 0.51]	1.002	656	1267
partner					
is female: support BP	1.37	[1.16, 1.59]	1.002	491	987
actor					
is female: support BP	0.60	[0.35, 0.82]	1.003	432	740
partner					
is male: support WP	0.18	[0.14, 0.23]	1.000	5755	3046
actor					
is male: support WP	0.13	[0.08, 0.18]	1.000	6028	3158
partner					
is female: support WP	0.42	[0.39, 0.45]	1.000	5549	3243
actor					
is female: support WP	0.17	[0.14, 0.20]	1.000	5548	3014
partner					
Random Effects					
sd(is male)	0.85	[0.74, 0.98]	1.002	857	1449
sd(is female)	0.91	[0.79, 1.05]	1.006	515	1265
sd(is male: day c)	0.02	[0.01, 0.02]	1.000	1973	2634
sd(is female: day c)	0.02	[0.01, 0.02]	1.001	1361	2175
sd(is male: support WP	0.09	[0.02, 0.15]	1.002	1162	1085
actor)					
sd(is male: support WP	0.14	[0.06, 0.20]	1.000	1012	947
partner)					
sd(is female: support	0.08	[0.02, 0.12]	1.004	923	905
WP actor)					
sd(is female: support	0.06	[0.01, 0.10]	1.001	1002	1720
WP partner)					
cor(is male, is female)	0.15	[-0.04, 0.34]	1.022	350	819
cor(is male, is male:	0.48	[0.28, 0.63]	1.001	2211	2867
day c)					
cor(is female, is male:	0.00	[-0.20, 0.22]	1.003	1801	3156
day c)					
cor(is male, is female:	0.17	[-0.03, 0.36]	1.005	940	2321
day c)					
cor(is female, is female:	0.49	[0.30, 0.64]	1.008	1172	2078
day c)					
cor(is male: day c, is	-0.03	[-0.26, 0.19]	1.003	1108	1730
female: day c)					
cor(is male, is male:	0.37	[-0.06, 0.70]	1.001	4857	2732
support WP actor)					
cor(is female, is male:	0.19	[-0.22, 0.57]	1.003	5018	2783
support WP actor)					
cor(is male: day c, is	0.33	[-0.13, 0.68]	1.001	5222	3270
male: support WP actor)					
cor(is female: day c, is	0.27	[-0.17, 0.64]	1.000	4899	2758
male: support WP actor)					
cor(is male, is male:	0.06	[-0.26, 0.38]	1.000	5278	3217
support WP partner)					
cor(is female, is male:	0.10	[-0.22, 0.43]	1.001	4991	2571
support WP partner)					
cor(is male: day c, is	0.04	[-0.31, 0.40]	1.000	4298	2455
male: support WP					
partner)					
cor(is female: day c, is	0.11	[-0.22, 0.45]	1.000	4766	2466
male: support WP					
partner)					
cor(is male: support	0.24	[-0.27, 0.68]	1.005	1124	2137
WP actor, is male:					
support WP partner)					
cor(is male, is female:	-0.06	[-0.44, 0.29]	1.001	4620	2478
support WP actor)					
cor(is female, is female:	0.21	[-0.17, 0.55]	1.000	4730	2276
support WP actor)					
cor(is male: day c, is	-0.14	[-0.51, 0.25]	1.001	4056	3026
female: support WP					
actor)					

(continued)

	Est.	95% CI	Rhat	Bulk ESS	Tail ESS
cor(is female: day c, is female: support WP actor)	-0.07	[-0.43, 0.30]	1.000	5099	3132
cor(is male: support WP actor, is female: support WP actor)	-0.17	[-0.61, 0.37]	1.002	1540	2245
cor(is male: support WP partner, is female: support WP actor)	-0.10	[-0.56, 0.40]	1.001	2566	2969
cor(is male, is female: support WP partner)	0.27	[-0.21, 0.63]	1.000	5732	2872
cor(is female, is female: support WP partner)	0.07	[-0.37, 0.48]	1.000	5615	2735
cor(is male: day c, is female: support WP partner)	0.12	[-0.34, 0.53]	1.000	5813	3459
cor(is female: day c, is female: support WP partner)	0.15	[-0.31, 0.56]	1.002	5869	2966
cor(is male: support WP actor, is female: support WP partner)	0.16	[-0.39, 0.64]	1.001	2540	3043
cor(is male: support WP partner, is female: support WP partner)	0.06	[-0.48, 0.57]	1.002	3413	3154
cor(is female: support WP actor, is female: support WP partner)	0.11	[-0.44, 0.60]	1.001	3345	3240
cortime(female, male)	0.04	[0.01, 0.07]	1.002	8477	2860
Residual Structure					
sigma is male	1.11	[1.09, 1.14]	1.002	6681	2536
sigma is female	0.72	[0.70, 0.73]	1.001	6554	2959

4.14 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.

4.15 Maximal Model Specification: Diagnostics

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

loo::pareto_k_table(loo1)
loo::pareto_k_table(loo2)
```

High Pareto-k values indicate influential observations and/or unreliable importance-sampling approximations. They are not automatically evidence of overfitting, but they should be checked; `loo(model, moment_match = TRUE)` is often recommended when `loo` flags problematic values.

The absolute ELPD values are not interpreted in isolation, but differences in ELPD can be used to compare models.

4.16 Maximal Model Specification: Diagnostics

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

No obvious overfitting is detected here. The more complex model appears more predictive, but ELPD differences should be interpreted relative to their standard errors; rules such as 2x SE are rough heuristics, not hard decision thresholds.

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)

4.17 Maximal Model Specification: Diagnostics

One optional summary is the `report` package’s comparison output, including its approximate p-value; this should be treated as an aid to interpretation rather than a definitive model-selection test.

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

The difference in predictive accuracy, as indexed by Expected Log Predictive Density (ELPD-LOO), suggests that ‘model_dist_apim_long_complex’ is the best model (ELPD = -14608.23), followed by ‘model_dist_apim_long_simple’ (diff-ELPD = -15.44 +- 6.68, p = 0.021).

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).

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

Group	Parameter	True (Simulated)	Estimate	95% CI	Coverage	Abs. Error	Recovery
Fixed	Intercept (male)	4.5659893	4.4890395	[4.305, 4.668]	Yes	0.077	Covered
Fixed	Intercept (female)	5.7823087	5.7316403	[5.537, 5.919]	Yes	0.051	Covered
Fixed	Time slope (male)	-0.0050000	-0.0029689	[-0.007, 0.001]	Yes	0.002	Covered
Fixed	Time slope (female)	0.0100000	0.0104637	[0.007, 0.014]	Yes	0.000	Covered
Between-person	Actor BP (male)	1.1855503	1.1204791	[0.917, 1.322]	Yes	0.065	Covered
Between-person	Partner BP (male)	0.2707719	0.3078625	[0.104, 0.511]	Yes	0.037	Covered
Between-person	Actor BP (female)	1.4298780	1.3697538	[1.156, 1.594]	Yes	0.060	Covered
Between-person	Partner BP (female)	0.5291829	0.5924361	[0.351, 0.817]	Yes	0.063	Covered
Within-person	Actor WP (male)	0.2000000	0.1812234	[0.135, 0.227]	Yes	0.019	Covered
Within-person	Partner WP (male)	0.1000000	0.1270781	[0.079, 0.176]	Yes	0.027	Covered
Within-person	Actor WP (female)	0.4000000	0.4205947	[0.390, 0.451]	Yes	0.021	Covered
Within-person	Partner WP (female)	0.2000000	0.1713545	[0.143, 0.200]	Yes	0.029	Covered
Random effects (SD)	Couple intercept SD (male)	0.7919045	0.8495693	[0.740, 0.984]	Yes	0.058	Covered
Random effects (SD)	Couple intercept SD (female)	0.8979616	0.9154796	[0.793, 1.055]	Yes	0.018	Covered
Random effects (SD)	Time slope SD (male)	0.0149848	0.0167471	[0.014, 0.020]	Yes	0.002	Covered
Random effects (SD)	Time slope SD (female)	0.0150338	0.0164231	[0.014, 0.019]	Yes	0.001	Covered
Random effects (SD)	Actor WP slope SD (male)	0.1006336	0.0899573	[0.021, 0.153]	Yes	0.011	Covered
Random effects (SD)	Partner WP slope SD (male)	0.0806085	0.1344609	[0.055, 0.201]	Yes	0.054	Covered
Random effects (SD)	Actor WP slope SD (female)	0.0955935	0.0759606	[0.019, 0.120]	Yes	0.020	Covered
Random effects (SD)	Partner WP slope SD (female)	0.0678586	0.0542037	[0.005, 0.102]	Yes	0.014	Covered
Random effects (cor)	cor(Intercept (male), Intercept (female))	0.1803231	0.1521969	[-0.042, 0.335]	Yes	0.028	Covered
Random effects (cor)	cor(Intercept (male), Time slope (male))	0.5083773	0.4708544	[0.283, 0.632]	Yes	0.038	Covered
Random effects (cor)	cor(Intercept (male), Time slope (female))	0.1921701	0.1708417	[-0.033, 0.365]	Yes	0.021	Covered
Random effects (cor)	cor(Intercept (male), Actor WP slope (male))	0.2262539	0.3543439	[-0.059, 0.696]	Yes	0.128	Covered
Random effects (cor)	cor(Intercept (male), Partner WP slope (male))	0.0953425	0.0605446	[-0.260, 0.378]	Yes	0.035	Covered
Random effects (cor)	cor(Intercept (male), Actor WP slope (female))	0.0398182	-0.0614619	[-0.436, 0.294]	Yes	0.101	Covered
Random effects (cor)	cor(Intercept (male), Partner WP slope (female))	0.3479821	0.2520320	[-0.206, 0.629]	Yes	0.096	Covered
Random effects (cor)	cor(Intercept (female), Time slope (male))	0.0526005	0.0061449	[-0.201, 0.221]	Yes	0.046	Covered
Random effects (cor)	cor(Intercept (female), Time slope (female))	0.6138595	0.4831640	[0.303, 0.636]	Yes	0.131	Covered
Random effects (cor)	cor(Intercept (female), Actor WP slope (male))	0.1319990	0.1866849	[-0.225, 0.569]	Yes	0.055	Covered
Random effects (cor)	cor(Intercept (female), Partner WP slope (male))	-0.0241164	0.1010175	[-0.221, 0.429]	Yes	0.125	Covered
Random effects (cor)	cor(Intercept (female), Actor WP slope (female))	0.1385319	0.2046311	[-0.172, 0.549]	Yes	0.066	Covered
Random effects (cor)	cor(Intercept (female), Partner WP slope (female))	-0.0263927	0.0667585	[-0.371, 0.485]	Yes	0.093	Covered
Random effects (cor)	cor(Time slope (male), Time slope (female))	0.1339757	-0.0304431	[-0.257, 0.190]	Yes	0.164	Covered
Random effects (cor)	cor(Time slope (male), Actor WP slope (male))	0.2400640	0.3130248	[-0.133, 0.682]	Yes	0.073	Covered
Random effects (cor)	cor(Time slope (male), Partner WP slope (male))	0.0918211	0.0427317	[-0.313, 0.397]	Yes	0.049	Covered
Random effects (cor)	cor(Time slope (male), Actor WP slope (female))	0.0024579	-0.1337158	[-0.512, 0.253]	Yes	0.136	Covered
Random effects (cor)	cor(Time slope (male), Partner WP slope (female))	0.2635152	0.1105994	[-0.343, 0.527]	Yes	0.153	Covered
Random effects (cor)	cor(Time slope (female), Actor WP slope (male))	0.0711586	0.2634154	[-0.166, 0.641]	Yes	0.192	Covered
Random effects (cor)	cor(Time slope (female), Partner WP slope (male))	0.0441926	0.1093859	[-0.225, 0.448]	Yes	0.065	Covered

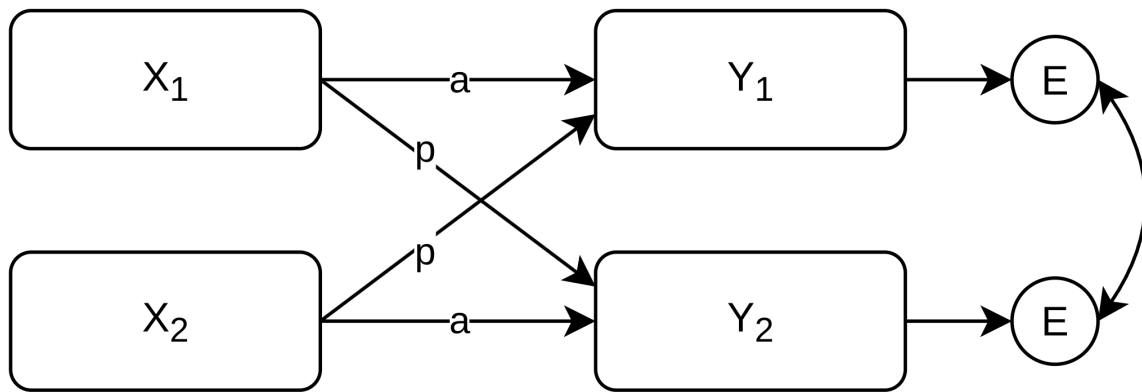
(continued)

Group	Parameter	True (Simulated)	Estimate	95% CI	Coverage	Abs. Error	Recovery
Random effects (cor)	cor(Time slope (female), Actor WP slope (female))	0.1380171	-0.0687735	[-0.428, 0.300]	Yes	0.207	Covered
Random effects (cor)	cor(Time slope (female), Partner WP slope (female))	0.0279326	0.1470529	[-0.313, 0.558]	Yes	0.119	Covered
Random effects (cor)	cor(Actor WP slope (male), Partner WP slope (male))	0.1128802	0.2345585	[-0.266, 0.682]	Yes	0.122	Covered
Random effects (cor)	cor(Actor WP slope (male), Actor WP slope (female))	0.0478755	-0.1584750	[-0.610, 0.370]	Yes	0.206	Covered
Random effects (cor)	cor(Actor WP slope (male), Partner WP slope (female))	0.1135981	0.1524683	[-0.386, 0.639]	Yes	0.039	Covered
Random effects (cor)	cor(Partner WP slope (male), Actor WP slope (female))	-0.0107839	-0.0987572	[-0.562, 0.404]	Yes	0.088	Covered
Random effects (cor)	cor(Partner WP slope (male), Partner WP slope (female))	0.1384787	0.0602770	[-0.484, 0.573]	Yes	0.078	Covered
Random effects (cor)	cor(Actor WP slope (female), Partner WP slope (female))	0.1494376	0.1005532	[-0.442, 0.599]	Yes	0.049	Covered
Residuals	Same-day residual correlation	0.0366176	0.0413748	[0.015, 0.068]	Yes	0.005	Covered
Residuals	Residual SD (male) (exp)	1.1210517	1.1147509	[1.093, 1.136]	Yes	0.006	Covered
Residuals	Residual SD (female) (exp)	0.7173353	0.7181891	[0.705, 0.732]	Yes	0.001	Covered

5 Exchangeable Dyads

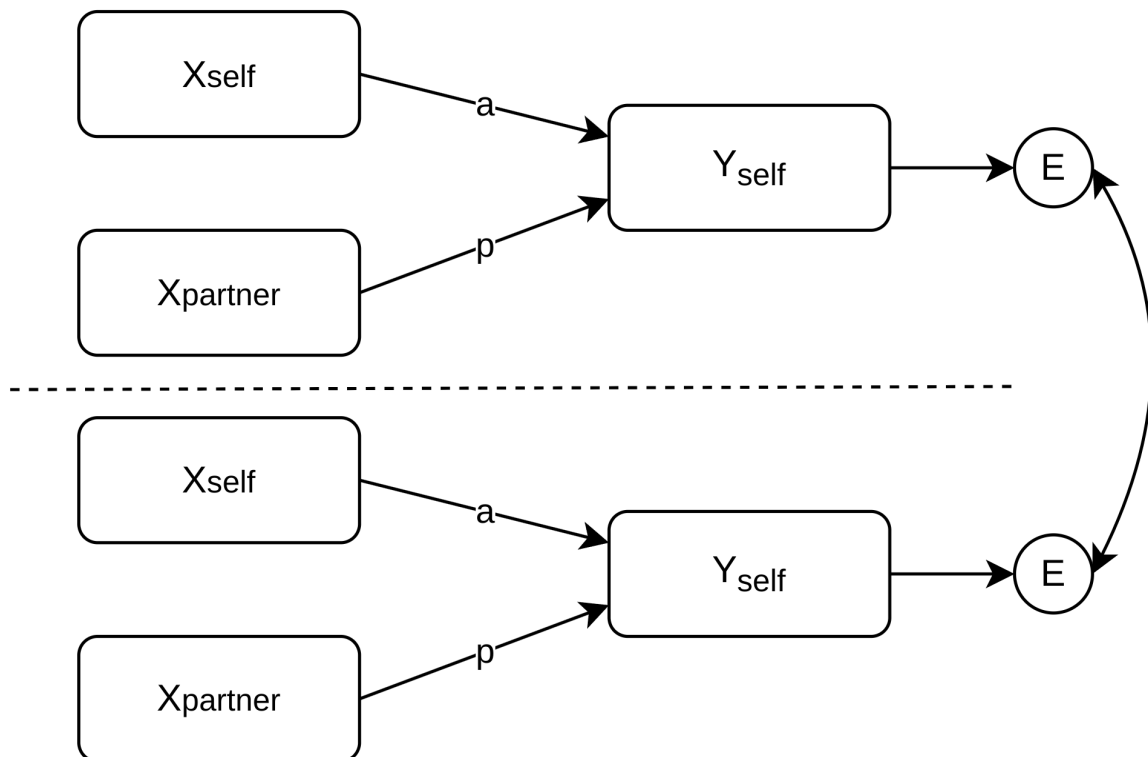
6 Exchangeable Dyads - Cross-Sectional

6.1 Exchangeable Dyads - Cross-Sectional APIM



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

6.2 Exchangeable Dyads - Cross-Sectional APIM



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

6.3 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)

6.4 Exchangeable Dyads - Cross-Sectional APIM: Data

For illustration, we now prepare the same female-male dyads under an exchangeability constraint. This is a modelling constraint/test, not a claim that the data-generating dyads are truly exchangeable. Starting again from the raw data makes that changed assumption explicit.

```
df_exchangeable <- prepare_interdep_data(  
  data = df,  
  group = coupleID,  
  member = userID,  
  role = gender,  
  predictors = communication,  
  model_type = c("apim", "dim"),  
  include_compositions = "female-male",  
  set_exchangeable_compositions = "female-male",  
  seed = 123  
) %>%  
  rename_interdep_for_tutorial(  
    predictor = "communication"  
  )
```

```
print_apim_preview(df_exchangeable)
```

userID	coupleID	gender	satisfaction	comm actor	comm partner
1 1	1	female	4.841332	4.851107	6.004533
1 2	1	male	5.577165	6.004533	4.851107
2 1	2	female	7.248741	5.881321	6.433723
2 2	2	male	6.960293	6.433723	5.881321
3 1	3	female	6.928699	4.283971	5.516060
3 2	3	male	6.077688	5.516060	4.283971

6.5 Exchangeable Dyads - Cross-Sectional APIM: glmmTMB

- For the fixed effects, we can pool the intercepts `is_male` and `is_female` by simply reintroducing a single shared intercept 1.
- For the actor and partner effects, we can remove the interactions to get a pooled estimate.
- For the residual, we want to constrain both partners' variances to be equal. Often, this is attempted via: `(1 | coupleID)`.
 - This gives both partners the same dyad-level random intercept variance.
 - With the usual Gaussian residual variance still present, the implied partner correlation is positive but not freely estimated: $\frac{\tau^2}{\tau^2 + \sigma^2}$.

- If residual variance is removed with `dispformula = ~ 0`, the shared random intercept would imply a correlation near 1, which is too restrictive.

6.6 Exchangeable Dyads - Cross-Sectional APIM: glmmTMB

One solution in glmmTMB is the following:

```

1 model_exch_apim_glmmTMB <- glmmTMB(
2   satisfaction ~
3
4   # Pooled single intercept
5   1 +
6
7   # Pooled actor and partner effects
8   communication_actor + communication_partner +
9
10  # Independent common and member-deviation terms parameterize a homogeneous
11  # two-member covariance using interdep's arbitrary contrast.
12  (1 | coupleID) + (0 + idiff | coupleID)
13
14  # Since we model residuals via a cs random effects structure, glmmTMB requires us to remove an
15  # additional independent residual
16  , dispformula = ~ 0
17
18  , data = df_exchangeable
19 )

```

6.7 Exchangeable Dyads - Cross-Sectional APIM: glmmTMB

This successfully returns a single pooled residual variance, and a correlation between partners:

```

Family: gaussian ( identity )
Formula:      satisfaction ~ 1 + communication_actor + communication_partner +
          (1 | coupleID) + (0 + idiff | coupleID)
Dispersion:      ~0
Data: df_exchangeable

      AIC      BIC    logLik -2*log(L)  df.resid
4026.6   4051.7   -2008.3   4016.6     1095

Random effects:

Conditional model:
Groups      Name      Variance Std.Dev.
coupleID    (Intercept) 1.6146   1.2707
coupleID.1  idiff       0.7881   0.8877
Number of obs: 1100, groups:  coupleID, 550

Conditional model:
              Estimate Std. Error z value Pr(>|z|)
(Intercept)   -4.70971    0.22332  -21.09 < 2e-16 ***
communication_actor  1.71990    0.03177   54.13 < 2e-16 ***
communication_partner 0.24137    0.03177    7.60 3.04e-14 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

6.8 Exchangeable Dyads - Cross-Sectional APIM: glmmTMB

- The gender-coded cross-sectional version does not transfer directly to ILD.
- For repeated measures, we instead index the homogeneous two-partner residual structure at the couple-day level, as shown later.

6.9 Exchangeable Dyads - Cross-Sectional APIM: brms

In brms, this issue can be solved more easily with a native residual correlation structure and one pooled residual variance:

```
1 formula_exch_apim <- bf(
2   satisfaction ~ 1 +
3     communication_actor + communication_partner +
4
5   # To achieve a homogeneous two-partner covariance structure we use an
6   # unstructured residual correlation over the two member positions.
7   unstr(time = idiff, gr = coupleID)
8
9   # ... but now, we do not allow individual variances anymore.
10  # We want one pooled residual variance. So we do not specify a sigma formula.
11  # This implies:
12  # , sigma = ~ 1
13 )
14
15 priors_exch_apim <- c(
16   prior(normal(2, 10), class = "Intercept"),
17   prior(normal(0, 5), class = "b"),
18   prior(student_t(3, 0, 1.5), class = "sigma"),
19   prior(lkj(2), class = "cortime")
20 )
21
22 model_exch_apim <- brm(
23   formula = formula_exch_apim,
24   data = df_exchangeable,
25   family = gaussian(link = identity),
26   prior = priors_exch_apim,
27   chains = 4,
28   cores = 4,
29   iter = 2000,
30   warmup = 1000,
31   seed = 123,
32   file = file.path('brms_cache', 'example1_ind_apim') # Cache the model
33 )
```

6.10 Exchangeable Dyads - Cross-Sectional APIM: Results

	Est.	95% CI	Rhat	Bulk ESS	Tail ESS
Fixed Effects					
Intercept	-4.71	[-5.13, -4.26]	1.004	4681	2769
comm actor	1.72	[1.66, 1.78]	1.002	4425	2895
comm partner	0.24	[0.18, 0.30]	1.000	4971	3242
Residual Structure					
cortime(-1, 1)	0.34	[0.26, 0.42]	1.000	4665	3140
sigma	1.55	[1.49, 1.63]	1.001	4114	3191

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

Leave-one-out (loo) cross-validation compares out-of-sample predictive accuracy. It can be meaningful for non-nested models when they are fit to the same outcome and observations, and when the pointwise log-likelihoods are comparable.

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

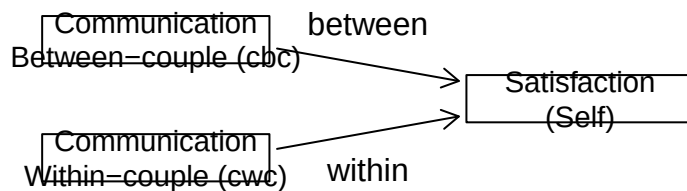
Model	elpd_diff	se_diff
model_dist_apim	0.0	0.0
model_exch_apim	-172.9	16.5

```
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_exch_apim’ (diff-ELPD = -172.85 +- 16.53, $p < .001$). See: documentation of “report”

6.12 Exchangeable Dyads - Cross-Sectional DIM

Dyad-Individual Model (DIM)



6.13 Exchangeable Dyads - Cross-Sectional DIM: Data

The same `prepare_interdep_data()` call generated the DIM columns directly from the raw long-format observations.

userID	coupleID	satisfaction	comm	comm dyad mean	comm dyad dev	idiff
1 1	1	4.841332	4.851107	0.4213527	-0.5767130	-1
1 2	1	5.577165	6.004533	0.4213527	0.5767130	1
2 1	2	7.248741	5.881321	1.1510550	-0.2762013	-1
2 2	2	6.960293	6.433723	1.1510550	0.2762013	1
3 1	3	6.928699	4.283971	-0.1064514	-0.6160447	-1
3 2	3	6.077688	5.516060	-0.1064514	0.6160447	1

6.14 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

6.15 Exchangeable Dyads - Cross-Sectional DIM: Centering

communication_dyad_mean_gmc is the grand-mean-centered dyad mean, while communication_dyad_dev is each member's deviation from that dyad mean.

6.16 Exchangeable Dyads - Cross-Sectional DIM: Centering

userID	coupleID	satisfaction	comm dyad mean	comm dyad dev	idiff
1 1	1	4.841332	0.4213527	-0.5767130	-1
1 2	1	5.577165	0.4213527	0.5767130	1
2 1	2	7.248741	1.1510550	-0.2762013	-1
2 2	2	6.960293	1.1510550	0.2762013	1
3 1	3	6.928699	-0.1064514	-0.6160447	-1
3 2	3	6.077688	-0.1064514	0.6160447	1

6.17 Exchangeable Dyads - Cross-Sectional DIM: Model

```

1 formula_exch_dim <- bf(
2   satisfaction ~ 1 +
3     communication_dyad_mean_gmc + communication_dyad_dev +
4     unstr(time = idiff, gr = coupleID)
5
6   # sigma ~ 1 is implied.
7 )
8
9 priors_exch_dim <- c(
10  prior(normal(2, 10), class = "Intercept"),
11  prior(normal(0, 5), class = "b"),
12  prior(student_t(3, 0, 1.5), class = "sigma"),
13  prior(lkj(2), class = "cortime")
14 )
15
16 model_exch_dim <- brm(
17   formula = formula_exch_dim,
18   data = df_exchangeable,
19   family = gaussian(link = identity),
20   prior = priors_exch_dim,
21   chains = 4,
22   cores = 4,
23   iter = 2000,
24   warmup = 1000,
25   seed = 123,
26   file = file.path('brms_cache', 'example1_ind_dim') # Cache the model
27 )

```

6.18 Exchangeable Dyads - Cross-Sectional DIM: Results

	Est.	95% CI	Rhat	Bulk ESS	Tail ESS
Fixed Effects					
Intercept	5.11	[5.00, 5.22]	1.001	4390	2769
comm dyad mean	1.96	[1.87, 2.05]	1.003	4658	3120
comm dyad dev	1.48	[1.39, 1.57]	1.002	4285	2998
Residual Structure					
cortime(-1, 1)	0.34	[0.27, 0.42]	1.001	4164	2873
sigma	1.55	[1.49, 1.62]	1.000	4560	3396

6.19 Equivalence APIM and DIM

The APIM uses raw actor and partner predictors, whereas the generated DIM dyad mean is grand-mean centered. The two fitted models therefore use different origins: their slope and covariance parameters are directly comparable, but their printed intercepts are not. For the equations below, we shift the APIM intercept to average communication:

$$b_{0, APIM(GMC)} = b_{0, APIM(raw)} + \bar{x}(b_{actor} + b_{partner}).$$

	Est.	95% CI
Fixed Effects		
Intercept	-4.71	[-5.13, -4.26]
comm actor	1.72	[1.66, 1.78]
comm partner	0.24	[0.18, 0.30]
Residual Structure		
cortime(-1, 1)	0.34	[0.26, 0.42]
sigma	1.55	[1.49, 1.63]

	Est.	95% CI
Fixed Effects		
Intercept	5.11	[5.00, 5.22]
comm dyad mean	1.96	[1.87, 2.05]
comm dyad dev	1.48	[1.39, 1.57]
Residual Structure		
cortime(-1, 1)	0.34	[0.27, 0.42]
sigma	1.55	[1.49, 1.62]

$$b_{actor} + b_{partner} = b_{cbc}$$

$$b_{actor} - b_{partner} = b_{cwc}$$

(Bolger et al., 2025)

6.20 Equivalence APIM and DIM

	Est.	95% CI
Fixed Effects		
Intercept	-4.71	[-5.13, -4.26]
comm actor	1.72	[1.66, 1.78]
comm partner	0.24	[0.18, 0.30]

	Est.	95% CI
Fixed Effects		
Intercept	5.11	[5.00, 5.22]
comm dyad mean	1.96	[1.87, 2.05]
comm dyad dev	1.48	[1.39, 1.57]

6.21 Equivalence APIM and DIM

In the interactive HTML slides, this section contains sliders that demonstrate the APIM–DIM reparameterization. In this static version:

- actor and partner predictors are rewritten as a dyad-level mean and a within-dyad half-difference; and
- the DIM coefficients are a linear reparameterization of the APIM coefficients, so fitted values are unchanged.

```
x_cbc = (x_actor_gmc + x_partner_gmc) / 2  
x_cwc = (x_actor_gmc - x_partner_gmc) / 2
```

6.22 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:

$$b_{cbc} = b_{actor} + b_{partner}$$

- 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:

$$b_{cwc} = b_{actor} - b_{partner}$$

(Bolger et al., 2025)

6.23 Equivalence APIM and DIM

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

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

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

6.24 Equivalence APIM and DIM

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

```
a <- hypothesis(
  model_exch_apim,
  "communication_actor + communication_partner = 0"
)

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

	Estimate	Est.Error	CI.Lower	CI.Upper
1	1.96	0.04	1.87	2.04

	Est.	95% CI
Fixed Effects		
Intercept	-4.71	[-5.13, -4.26]
comm actor	1.72	[1.66, 1.78]
comm partner	0.24	[0.18, 0.30]

	Est.	95% CI
Fixed Effects		
Intercept	5.11	[5.00, 5.22]
comm dyad mean	1.96	[1.87, 2.05]
comm dyad dev	1.48	[1.39, 1.57]

6.25 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

6.25.1 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)

6.26 Exchangeable Dyads - L-DIM/L-APIM: Simulated Data

We start again from the raw longitudinal data and omit `role`, which tells `interdep` to treat the two members as exchangeable. Requesting both model types creates the APIM and DIM predictors in the same model-ready object.

```
df_long_dim <- prepare_interdep_data(
  data = df_long,
  group = coupleID,
  member = userID,
  time = diaryday,
  predictors = provided_support,
  model_type = c("apim", "dim"),
  seed = 123
) %>%
  rename_interdep_for_tutorial(
    # Keep interdep's cwp_actor/cbp_partner component ordering.
    predictor = "provided_support"
  ) %>%
  mutate(
    diaryday_c = diaryday - mean(diaryday)
  )
```

userID	coupleID	idiff	day	day c	closeness	support WP actor	support WP partner	support BP actor	support BP partner
31 1	31	-1	0	-27	5.47	0.41	1.54	0.65	0.58
31 1	31	-1	1	-26	7.69	0.51	1.4	0.65	0.58
31 1	31	-1	2	-25	5.83	0.49	0.79	0.65	0.58
31 1	31	-1	3	-24	6.55	1.38	-0.15	0.65	0.58
...	...	...	...	...	...	...	...	...	...
31 2	31	1	0	-27	5.19	1.54	0.41	0.58	0.65
31 2	31	1	1	-26	6.96	1.4	0.51	0.58	0.65
31 2	31	1	2	-25	6.9	0.79	0.49	0.58	0.65
31 2	31	1	3	-24	7.59	-0.15	1.38	0.58	0.65

7 Exchangeable Dyads - Intensive Longitudinal

7.1 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.

The generated `provided_support_cwp_dyad_mean` and `provided_support_cwp_dev` columns form the within-person DIM. The corresponding `provided_support_cbp_dyad_mean` and `provided_support_cbp_dev` columns form the between-person DIM.

7.2 Exchangeable Dyads - L-DIM/L-APIM: Preparing the Mean-Deviation idiff Approach

The `idiff` alias refers to `interdep`'s generated arbitrary $-1/+1$ member contrast. The fixed seed makes that arbitrary assignment reproducible.

userID	coupleID	idiff	day	day c	support BP dyad mean	support BP dev	support WP dyad mean	support WP dev
31 1	31	-1	0	-27	0.61	0.04	0.97	-0.57
31 1	31	-1	1	-26	0.61	0.04	0.96	-0.44
31 1	31	-1	2	-25	0.61	0.04	0.64	-0.15
31 1	31	-1	3	-24	0.61	0.04	0.61	0.77
...	...	...	...	...	...	...	...	...
31 2	31	1	0	-27	0.61	-0.04	0.97	0.57
31 2	31	1	1	-26	0.61	-0.04	0.96	0.44
31 2	31	1	2	-25	0.61	-0.04	0.64	0.15
31 2	31	1	3	-24	0.61	-0.04	0.61	-0.77

7.3 Exchangeable Dyads - L-DIM/L-APIM: Preparing the Mean-Deviation idiff Approach

userID	coupleID	idiff	day	day c	support BP dyad mean	support BP dev	support WP dyad mean	support WP dev
31 1	31	-1	0	-27	0.61	0.04	0.97	-0.57
31 1	31	-1	1	-26	0.61	0.04	0.96	-0.44
31 1	31	-1	2	-25	0.61	0.04	0.64	-0.15
31 1	31	-1	3	-24	0.61	0.04	0.61	0.77
31 1	31	-1	4	-23	0.61	0.04	0.36	0.72
...	...	...	...	...	...	...	...	...
31 2	31	1	0	-27	0.61	-0.04	0.97	0.57
31 2	31	1	1	-26	0.61	-0.04	0.96	0.44
31 2	31	1	2	-25	0.61	-0.04	0.64	0.15
31 2	31	1	3	-24	0.61	-0.04	0.61	-0.77
31 2	31	1	4	-23	0.61	-0.04	0.36	-0.72

7.4 Exchangeable Dyads - L-DIM/L-APIM: glmmTMB

```

1 model_exch_long_apim_glmmTMB <- glmmTMB(
2   closeness ~ 1 +
3
4   diaryday_c +
5
6   # Within-person APIM
7   provided_support_cwp_actor +
8   provided_support_cwp_partner +
9   # Equivalent to within-person DIM:
10  # provided_support_cwp_dyad_mean +
11  # provided_support_cwp_dev +
12
13  # Between-person APIM
14  provided_support_cbp_actor +
15  provided_support_cbp_partner +
16  # Equivalent to between-person DIM
17  # provided_support_cbp_dev +
18  # provided_support_cbp_dyad_mean +
19
20  # Dyad-Level intercept and slopes for time-varying predictors
21  (1 + diaryday_c + provided_support_cwp_actor + provided_support_cwp_partner | coupleID) +
22
23  # Both partners' deviations from these dyad-level means and slopes.
24  # Note: separate random-effects block to make them uncorrelated with dyad-level REs.
25  (0 + idiff +
26    I(idiff * diaryday_c) +
27    I(idiff * provided_support_cwp_actor) +
28    I(idiff * provided_support_cwp_partner) | coupleID) +
29
30  # After accounting for stable differences between partners, they may still be
31  # non-independent each day due to external occurrences and daily exchange processes
32  # not captured in the fixed effects. Thus, residuals are not yet independent.
33  # Homogeneous same-day residual covariance between partners:
34  (1 | coupleID:diaryday) + (0 + idiff | coupleID:diaryday)
35
36  # We remove the additional independent residual variance
37  # (only if the family has a native residual!)
38  , dispformula = ~ 0
39  , data = df_long_dim
40 )

```

7.5 Exchangeable Dyads - L-DIM/L-APIM: glmmTMB

In this example, the glmmTMB model does not converge properly. The random-effects structure is too complex for the information available in these data, so we would have to simplify it. Instead, we use brms below.

7.6 Exchangeable Dyads - L-DIM/L-APIM: brms

In brms, the same model can be estimated as:

Unlike the distinguishable longitudinal model, this exchangeable specification keeps one pooled residual `sigma` (`sigma ~ 1` is implied). Its residual SD prior therefore targets `class = "sigma"` directly.

```
1 formula_exch_long_apim <- bf(
2   closeness ~ 1 +
3
4   diaryday_c +
5
6   # Within-person APIM
7   provided_support_cwp_actor +
8   provided_support_cwp_partner +
9
10  # Between-person APIM
11  provided_support_cbp_actor +
12  provided_support_cbp_partner +
13
14  # Dyad-Level intercept and slopes for time-varying predictors
15  (1 + diaryday_c + provided_support_cwp_actor + provided_support_cwp_partner | coupleID) +
16
17  # Both partners' deviations from these dyad-level means and slopes.
18  # Note: separate random-effects block to make them uncorrelated with dyad-level REs.
19  (0 + idiff +
20    I(idiff * diaryday_c) +
21    I(idiff * provided_support_cwp_actor) +
22    I(idiff * provided_support_cwp_partner) | coupleID) +
23
24  # Modelling Level 1 residuals in brms:
25  unstr(time = idiff, gr = coupleID:diaryday)
26
27  # Again, no need to model heterogeneous residual variances (sigma)
28  # Implied: sigma = ~ 1
29 )
30
31 priors_exch_long_apim <- c(
32   prior(normal(4, 2), class = "Intercept"),
33   prior(normal(0, 2), class = "b"),
34   prior(exponential(1), class = "sd"),
35   prior(lkj(2), class = "cor"),
36   prior(student_t(3, 0, 1.5), class = "sigma"),
37   prior(lkj(2), class = "cortime")
38 )
39
40 model_exch_long_apim <- brm(
41   formula = formula_exch_long_apim,
42   data = df_long_dim,
43   family = gaussian(link = identity),
44   prior = priors_exch_long_apim,
45   chains = 4,
46   cores = 4,
47   iter = 2000,
48   warmup = 1000,
49   seed = 123,
50   file = file.path('brms_cache', 'model_apim_ind_long_apim')
51 )
```

7.7 Exchangeable Dyads - L-DIM/L-APIM: brms

The equivalent DIM specification:

```
1 # DIM Version
2 formula_exch_long_dim <- bf(
3   closeness ~ 1 +
4
```

```

5     diaryday_c +
6
7     provided_support_cwp_dyad_mean +
8     provided_support_cwp_dev +
9
10    provided_support_cbp_dyad_mean +
11    provided_support_cbp_dev +
12
13    (1 + diaryday_c + provided_support_cwp_actor + provided_support_cwp_partner | coupleID) +
14    (0 + idiff +
15      I(idiff * diaryday_c) +
16      I(idiff * provided_support_cwp_actor) +
17      I(idiff * provided_support_cwp_partner) | coupleID) +
18
19    unstr(time = idiff, gr = coupleID:diaryday)
20  )
21
22  priors_exch_long_dim <- c(
23    prior(normal(4, 2), class = "Intercept"),
24    prior(normal(0, 2), class = "b"),
25    prior(exponential(1), class = "sd"),
26    prior(lkj(2), class = "cor"),
27    prior(student_t(3, 0, 1.5), class = "sigma"),
28    prior(lkj(2), class = "cortime")
29  )
30
31  model_exch_long_dim <- brm(
32    formula = formula_exch_long_dim,
33    data = df_long_dim,
34    family = gaussian(link = identity),
35    prior = priors_exch_long_dim,
36    chains = 4,
37    cores = 4,
38    iter = 2000,
39    warmup = 1000,
40    seed = 123,
41    file = file.path('brms_cache', 'model_apim_ind_long_dim')
42  )

```

7.8 Exchangeable Dyads - L-DIM/L-APIM: Explanation idiff

For appropriate exchangeable random effects, we can use mean-deviation `idiff` coding (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 average/common intercept of the two partners. (1 | coupleID)
- A column of 1s and -1s represents each partner's deviation from the couple-level effect, with partners contributing equally in opposite directions (+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.)

7.9 Exchangeable Dyads - L-DIM/L-APIM: Common and idiff Components

For two partners coded with `idiff = +1` and `idiff = -1`, the `glmmTMB` structure (1 | coupleID:diaryday) + (0 + idiff | coupleID:diaryday) uses independent common and member-deviation components to imply a homogeneous two-partner residual covariance.

This parameterization assumes exactly two rows per couple-day, independent common/couple-mean and `idiff` deviation random-effect blocks, and `dispformula = ~ 0`.

The model code uses these common and deviation components directly, so no manually constructed partner-position indicators are required.

It also does not need back-transformation.

7.10 Exchangeable Dyads - L-DIM/L-APIM: Results

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

	Est.	95% CI
Fixed Effects		
Intercept	5.10	[4.97, 5.24]
day c	0.00	[0.00, 0.01]
support WP actor	0.30	[0.27, 0.33]
support WP partner	0.15	[0.12, 0.18]
support BP actor	1.32	[1.14, 1.48]
support BP partner	0.38	[0.21, 0.56]

	Est.	95% CI
Fixed Effects		
Intercept	5.11	[4.98, 5.23]
day c	0.00	[0.00, 0.01]
support WP dyad mean	0.45	[0.41, 0.49]
support WP dev	0.15	[0.11, 0.19]
support BP dyad mean	1.70	[1.55, 1.86]
support BP dev	0.93	[0.61, 1.21]

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

7.11 Exchangeable Dyads - L-DIM/L-APIM: Results

APIM Results in Detail:

	Est.	95% CI	Rhat	Bulk ESS	Tail ESS
Fixed Effects					
Intercept	5.10	[4.97, 5.24]	1.017	175	225
day c	0.00	[0.00, 0.01]	1.005	463	1553
support WP actor	0.30	[0.27, 0.33]	1.001	2845	2572
support WP partner	0.15	[0.12, 0.18]	1.000	3249	2819
support BP actor	1.32	[1.14, 1.48]	1.017	171	286
support BP partner	0.38	[0.21, 0.56]	1.005	212	492
Random Effects					
sd(Intercept)	0.67	[0.58, 0.77]	1.022	389	702
sd(day c)	0.01	[0.01, 0.01]	1.001	1581	2780
sd(support WP actor)	0.05	[0.01, 0.09]	1.004	990	953
sd(support WP partner)	0.08	[0.04, 0.12]	1.002	1158	1197
sd(idiff)	0.84	[0.74, 0.97]	1.049	218	539
sd(lidiffMUday c)	0.01	[0.01, 0.02]	1.010	1152	1947
sd(lidiffMUsupport WP actor)	0.14	[0.11, 0.18]	1.004	1154	2284
sd(lidiffMUsupport WP partner)	0.07	[0.01, 0.11]	1.004	754	627
cor(Intercept, day c)	0.56	[0.38, 0.70]	1.004	1129	1909
cor(Intercept, support WP actor)	0.49	[0.02, 0.82]	1.002	2958	2405
cor(day c, support WP actor)	0.37	[-0.10, 0.76]	1.002	2789	2668
cor(Intercept, support WP partner)	0.21	[-0.12, 0.53]	1.001	2870	2182
cor(day c, support WP partner)	0.19	[-0.19, 0.54]	1.003	2961	2724
cor(support WP actor, support WP partner)	0.35	[-0.27, 0.80]	1.002	687	1343
cor(idiff, lidiffMUday c)	0.60	[0.44, 0.73]	1.014	788	1455
cor(idiff, lidiffMUsupport WP actor)	0.66	[0.46, 0.81]	1.004	1299	1965
cor(lidiffMUday c, lidiffMUsupport WP actor)	0.42	[0.15, 0.63]	1.008	1619	2359
cor(idiff, lidiffMUsupport WP partner)	0.12	[-0.29, 0.53]	1.000	2816	1776
cor(lidiffMUday c, lidiffMUsupport WP partner)	0.11	[-0.32, 0.55]	1.003	3148	1966
cor(lidiffMUsupport WP actor, lidiffMUsupport WP partner)	0.34	[-0.17, 0.76]	1.000	1647	2123

cortime(-1, 1)	0.04	[0.01, 0.07]	1.001	4908	2263
Residual Structure					
sigma	0.94	[0.92, 0.95]	1.000	6231	2668

7.12 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, based on (Kenny & Ackerman, 2023). Just as we would in a SEM model with equality constraints!

For example, the within-partner variance of the random intercept is:

$$\text{Var}(u_{\text{common intercept}}) + \text{Var}(u_{\text{idiff intercept}})$$

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

$$\text{Var}(u_{\text{common intercept}}) - \text{Var}(u_{\text{idiff intercept}})$$

In the model output, the common-intercept variance is labelled **Intercept**, and the deviation-intercept variance is labelled **idiff**. This rotation only holds when the common/couple-mean and **idiff** deviation random effects are uncorrelated. This is why we put them in separate random-effects blocks.

7.13 Extract Full APIM Random Effects Variance-Covariance Matrix

```
a <- summarize_exchangeable_apim_brms(
  model_exch_long_apim,
  gr = "coupleID",
  idiff = "idiff",
  partner_labels = c("PartnerA", "PartnerB")
)
```

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

```
print_df(a$within_person_covariance_matrix)
```

	Intercept	day c	support WP actor	support WP partner
Intercept	1.1802004	0.0112332	0.0955767	0.0179927
day c	0.0112332	0.0003159	0.0010075	0.0002727
support WP actor	0.0955767	0.0010075	0.0238464	0.0042531
support WP partner	0.0179927	0.0002727	0.0042531	0.0118398

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


```
print_df(a$cross_person_covariance_matrix)
```

	Intercept	day c	support WP actor	support WP partner
Intercept	-0.2635988	-0.0024112	-0.0630951	0.0048753
day c	-0.0024112	-0.0000434	-0.0005769	0.0000857
support WP actor	-0.0630951	-0.0005769	-0.0179709	-0.0015373
support WP partner	0.0048753	0.0000857	-0.0015373	0.0024947

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

```
print_wide_matrix(a$full_covariance_matrix, font_size = 6)
```

	PA Int	PA day c	PA sup WP act	PA sup WP prt	PB Int	PB day c	PB sup WP act	PB sup WP prt
PA Int	1.180	0.011	0.096	0.018	-0.264	-0.002	-0.063	0.005
PA day c	0.011	0.000	0.001	0.000	-0.002	0.000	-0.001	0.000
PA sup WP act	0.096	0.001	0.024	0.004	-0.063	-0.001	-0.018	-0.002
PA sup WP prt	0.018	0.000	0.004	0.012	0.005	0.000	-0.002	0.002
PB Int	-0.264	-0.002	-0.063	0.005	1.180	0.011	0.096	0.018
PB day c	-0.002	0.000	-0.001	0.000	0.011	0.000	0.001	0.000
PB sup WP act	-0.063	-0.001	-0.018	-0.002	0.096	0.001	0.024	0.004
PB sup WP prt	0.005	0.000	-0.002	0.002	0.018	0.000	0.004	0.012

The helper function `summarize_exchangeable_apim_brms()` is defined in `00_R_Functions/ReportModels.R`. It rotates every posterior draw, so the interval estimates for the transformed SDs, correlations, covariances, and residual quantities are computed on the transformed scale.

For `glmmTMB`, the same back-transformation is algebraically possible, but interval estimation is less direct because there are no posterior draws. One would usually transform parametric bootstrap draws or asymptotic draws from the fitted parameter covariance matrix.

7.17 Random effects SDs

```
print_df(
  a$sd_summary %>%
    mutate(across(where(is.numeric), ~ round(.x, 3)))
)
```

parameter	estimate	est error	q low	q high
PartnerA Intercept	1.085	0.056	0.987	1.206
PartnerA day c	0.018	0.001	0.016	0.020
PartnerA support WP actor	0.153	0.018	0.119	0.191
PartnerA support WP partner	0.107	0.020	0.066	0.144
PartnerB Intercept	1.085	0.056	0.987	1.206
PartnerB day c	0.018	0.001	0.016	0.020
PartnerB support WP actor	0.153	0.018	0.119	0.191
PartnerB support WP partner	0.107	0.020	0.066	0.144

7.18 Random effect correlation matrix

```
print_wide_matrix(a$full_correlation_matrix, font_size = 6)
```

	PA Int	PA day c	PA sup WP act	PA sup WP prt	PB Int	PB day c	PB sup WP act	PB sup WP prt
PA Int	1.000	0.582	0.570	0.152	-0.223	-0.125	-0.376	0.041
PA day c	0.582	1.000	0.367	0.141	-0.125	-0.137	-0.210	0.044
PA sup WP act	0.570	0.367	1.000	0.253	-0.376	-0.210	-0.754	-0.091
PA sup WP prt	0.152	0.141	0.253	1.000	0.041	0.044	-0.091	0.211
PB Int	-0.223	-0.125	-0.376	0.041	1.000	0.582	0.570	0.152
PB day c	-0.125	-0.137	-0.210	0.044	0.582	1.000	0.367	0.141
PB sup WP act	-0.376	-0.210	-0.754	-0.091	0.570	0.367	1.000	0.253
PB sup WP prt	0.041	0.044	-0.091	0.211	0.152	0.141	0.253	1.000

7.19 Full reporting summary

```
section_runs <- rle(a$reporting_summary$section)
section_ends <- cumsum(section_runs$lengths)
section_starts <- section_ends - section_runs$lengths + 1
section_labels <- c(
  "Fixed effects" = "Fixed Effects",
  "Back-transformed random-effect SDs" = "Back-Transformed SDs",
  "Back-transformed random-effect correlations" = "Back-Transformed Correlations",
  "Residual structure" = "Residual Structure"
)
reporting_rows_to_pack <- Map(c, section_starts, section_ends)
names(reporting_rows_to_pack) <- unname(section_labels[section_runs$values])

print_df(
  a$reporting_summary %>%
    mutate(across(where(is.numeric), ~ round(.x, 3))),
  rows_to_pack = reporting_rows_to_pack,
  scroll_height = '50vh'
)
```

section	parameter	estimate	est error	q low	q high
Fixed Effects					
Fixed effects	Intercept	5.102	0.070	4.966	5.240
Fixed effects	day c	0.004	0.001	0.001	0.006
Fixed effects	support WP actor	0.300	0.014	0.273	0.328
Fixed effects	support WP partner	0.149	0.015	0.120	0.177
Fixed effects	support BP actor	1.316	0.088	1.140	1.483
Fixed effects	support BP partner	0.382	0.086	0.212	0.557
Back-Transformed SDs					
Back-transformed random-effect SDs	PartnerA Intercept	1.085	0.056	0.987	1.206
Back-transformed random-effect SDs	PartnerA day c	0.018	0.001	0.016	0.020
Back-transformed random-effect SDs	PartnerA support WP actor	0.153	0.018	0.119	0.191
Back-transformed random-effect SDs	PartnerA support WP partner	0.107	0.020	0.066	0.144
Back-transformed random-effect SDs	PartnerB Intercept	1.085	0.056	0.987	1.206
Back-transformed random-effect SDs	PartnerB day c	0.018	0.001	0.016	0.020
Back-transformed random-effect SDs	PartnerB support WP actor	0.153	0.018	0.119	0.191
Back-transformed random-effect SDs	PartnerB support WP partner	0.107	0.020	0.066	0.144
Back-Transformed Correlations					
Back-transformed random-effect correlations	PartnerA day c with PartnerA Intercept	0.581	0.056	0.470	0.681
Back-transformed random-effect correlations	PartnerA support WP actor with PartnerA Intercept	0.572	0.083	0.403	0.723

(continued)

section	parameter	estimate	est error	q low	q high
Back-transformed random-effect correlations	PartnerA support WP actor with PartnerA day c	0.369	0.101	0.169	0.558
Back-transformed random-effect correlations	PartnerA support WP partner with PartnerA Intercept	0.157	0.117	-0.069	0.382
Back-transformed random-effect correlations	PartnerA support WP partner with PartnerA day c	0.146	0.129	-0.114	0.400
Back-transformed random-effect correlations	PartnerA support WP partner with PartnerA support WP actor	0.266	0.152	-0.028	0.558
Back-transformed random-effect correlations	PartnerB Intercept with PartnerA Intercept	-0.221	0.095	-0.402	-0.039
Back-transformed random-effect correlations	PartnerB Intercept with PartnerA day c	-0.124	0.084	-0.288	0.040
Back-transformed random-effect correlations	PartnerB Intercept with PartnerA support WP actor	-0.377	0.103	-0.572	-0.163
Back-transformed random-effect correlations	PartnerB Intercept with PartnerA support WP partner	0.043	0.125	-0.199	0.281
Back-transformed random-effect correlations	PartnerB day c with PartnerA Intercept	-0.124	0.084	-0.288	0.040
Back-transformed random-effect correlations	PartnerB day c with PartnerA day c	-0.136	0.118	-0.362	0.094
Back-transformed random-effect correlations	PartnerB day c with PartnerA support WP actor	-0.211	0.111	-0.415	0.019
Back-transformed random-effect correlations	PartnerB day c with PartnerA support WP partner	0.046	0.138	-0.231	0.310
Back-transformed random-effect correlations	PartnerB day c with PartnerB Intercept	0.581	0.056	0.470	0.681
Back-transformed random-effect correlations	PartnerB support WP actor with PartnerA Intercept	-0.377	0.103	-0.572	-0.163
Back-transformed random-effect correlations	PartnerB support WP actor with PartnerA day c	-0.211	0.111	-0.415	0.019
Back-transformed random-effect correlations	PartnerB support WP actor with PartnerA support WP actor	-0.755	0.151	-0.983	-0.412
Back-transformed random-effect correlations	PartnerB support WP actor with PartnerA support WP partner	-0.096	0.160	-0.412	0.215
Back-transformed random-effect correlations	PartnerB support WP actor with PartnerB Intercept	0.572	0.083	0.403	0.723
Back-transformed random-effect correlations	PartnerB support WP actor with PartnerB day c	0.369	0.101	0.169	0.558
Back-transformed random-effect correlations	PartnerB support WP partner with PartnerA Intercept	0.043	0.125	-0.199	0.281
Back-transformed random-effect correlations	PartnerB support WP partner with PartnerA day c	0.046	0.138	-0.231	0.310
Back-transformed random-effect correlations	PartnerB support WP partner with PartnerA support WP actor	-0.096	0.160	-0.412	0.215
Back-transformed random-effect correlations	PartnerB support WP partner with PartnerA support WP partner	0.230	0.394	-0.570	0.960
Back-transformed random-effect correlations	PartnerB support WP partner with PartnerB Intercept	0.157	0.117	-0.069	0.382
Back-transformed random-effect correlations	PartnerB support WP partner with PartnerB day c	0.146	0.129	-0.114	0.400
Back-transformed random-effect correlations	PartnerB support WP partner with PartnerB support WP actor	0.266	0.152	-0.028	0.558
Residual Structure					
Residual structure	residual variance	0.878	0.012	0.854	0.903
Residual structure	sigma	0.937	0.007	0.924	0.950

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. (2025). *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. (2025). *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. (2026). *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.
- Küng, P. (2026). *Interdep: Prepare dyadic data for multilevel models*. <https://github.com/Pascal-Kueng/interdep>
- Lüdtke, D., Makowski, D., Ben-Shachar, M. S., Patil, I., Wiernik, B. M., Bacher, E., & Thériault, R. (2026). *Easystats: Framework for easy statistical modeling, visualization, and reporting*. <https://easystats.github.io/easystats/>
- Nickell, S. (1981). Biases in dynamic models with fixed effects. *Econometrica*, 49(6), 1417–1426. <https://doi.org/10.2307/1911408>
- Pinheiro, J., Bates, D., & R Core Team. (2026). *Nlme: Linear and nonlinear mixed effects models*. <https://svn.r-project.org/R-packages/trunk/nlme/>
- Ripley, B., & Venables, B. (2025). *MASS: Support functions and datasets for venables and ripley's MASS*. <http://www.stats.ox.ac.uk/pub/MASS4/>
- 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. (2025). *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>