Supplemental Materials for ‘Speech development between 30 and 119 months in typical children III: Interaction between speaking rate and intelligibility’

Tristan Mahr

2024-05-20

## Data

The first set of analyses considered each child’s mean speaking rate and intelligibility for each utterance length.

library(tidyverse)  
data\_model <- targets::tar\_read("data\_model\_anon")  
data\_model  
#> # A tibble: 2,196 × 14  
#> child speaking\_sps artic\_sps age\_months tocs\_level length\_longest age\_bin  
#> <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
#> 1 c464 2.71 2.71 30 3 4 36  
#> 2 c464 2.55 2.62 30 4 4 36  
#> 3 c017 2.04 2.08 30 3 3 36  
#> 4 c315 2.03 2.20 30 3 3 36  
#> 5 c468 1.87 1.90 30 3 3 36  
#> 6 c154 2.25 2.41 30 3 3 36  
#> 7 c338 2.59 2.69 31 3 3 36  
#> 8 c110 2.69 2.79 31 3 4 36  
#> 9 c110 2.66 2.74 31 4 4 36  
#> 10 c431 3.30 3.30 31 3 3 36  
#> # ℹ 2,186 more rows  
#> # ℹ 7 more variables: intelligibility <dbl>, mwi <dbl>, speaking\_sps\_3 <dbl>,  
#> # artic\_sps\_3 <dbl>, age\_48 <dbl>, tocs\_length <fct>, tocs\_level\_3 <dbl>

where

* child: child identifier
* speaking\_sps, artic\_sps, speaking\_sps\_3, artic\_sps\_3: speaking rate and articulation rate in syllables per second. The \_3 versions have 3 subtracted so that the values are centered at 3 sps.
* age\_months, age\_bin, age\_48: child age in months, rough age group of child (for plotting), age in months minus 48 (so that the values are centered at 48 months)
* tocs\_level, tocs\_length, tocs\_level\_3: utterance length as a number, factor, number centered at 3 word.
* length\_longest: the longest utterance reached by the child.
* intelligibility, mwi: observed intelligibility and intelligibility compressed into the range .001–.999 for beta regression.

## Correlations

data\_for\_cors <- data\_model |>   
 group\_by(tocs\_length) |>   
 mutate(  
 # age-residualized  
 r\_intel = residuals(lm(intelligibility ~ splines::ns(age\_months, 3))),  
 r\_rate = residuals(lm(speaking\_sps\_3 ~ splines::ns(age\_months, 3))),  
 ) |>   
 ungroup()  
  
data\_for\_cors |>   
 select(tocs\_length, age\_months, intelligibility, speaking\_sps, r\_intel, r\_rate) |>   
 group\_by(tocs\_length) |>   
 reframe(  
 pick(everything()) |> as.matrix() |> Hmisc::rcorr() |> broom::tidy()  
 ) |>   
 mutate(  
 estimate = round(estimate, 3),  
 p.value = scales::label\_pvalue()(p.value)  
 ) |>   
 knitr::kable(align = "rllrrr")

| tocs\_length | column1 | column2 | estimate | n | p.value |
| --- | --- | --- | --- | --- | --- |
| 3 | intelligibility | age\_months | 0.688 | 538 | <0.001 |
| 3 | speaking\_sps | age\_months | 0.330 | 538 | <0.001 |
| 3 | speaking\_sps | intelligibility | 0.216 | 538 | <0.001 |
| 3 | r\_intel | age\_months | 0.000 | 538 | >0.999 |
| 3 | r\_intel | intelligibility | 0.571 | 538 | <0.001 |
| 3 | r\_intel | speaking\_sps | -0.069 | 538 | 0.107 |
| 3 | r\_rate | age\_months | 0.000 | 538 | >0.999 |
| 3 | r\_rate | intelligibility | -0.042 | 538 | 0.329 |
| 3 | r\_rate | speaking\_sps | 0.942 | 538 | <0.001 |
| 3 | r\_rate | r\_intel | -0.074 | 538 | 0.087 |
| 4 | intelligibility | age\_months | 0.655 | 478 | <0.001 |
| 4 | speaking\_sps | age\_months | 0.440 | 478 | <0.001 |
| 4 | speaking\_sps | intelligibility | 0.281 | 478 | <0.001 |
| 4 | r\_intel | age\_months | 0.000 | 478 | >0.999 |
| 4 | r\_intel | intelligibility | 0.653 | 478 | <0.001 |
| 4 | r\_intel | speaking\_sps | -0.071 | 478 | 0.120 |
| 4 | r\_rate | age\_months | 0.000 | 478 | >0.999 |
| 4 | r\_rate | intelligibility | -0.052 | 478 | 0.255 |
| 4 | r\_rate | speaking\_sps | 0.891 | 478 | <0.001 |
| 4 | r\_rate | r\_intel | -0.080 | 478 | 0.081 |
| 5 | intelligibility | age\_months | 0.566 | 422 | <0.001 |
| 5 | speaking\_sps | age\_months | 0.438 | 422 | <0.001 |
| 5 | speaking\_sps | intelligibility | 0.176 | 422 | <0.001 |
| 5 | r\_intel | age\_months | 0.000 | 422 | >0.999 |
| 5 | r\_intel | intelligibility | 0.768 | 422 | <0.001 |
| 5 | r\_intel | speaking\_sps | -0.126 | 422 | 0.010 |
| 5 | r\_rate | age\_months | 0.000 | 422 | >0.999 |
| 5 | r\_rate | intelligibility | -0.108 | 422 | 0.027 |
| 5 | r\_rate | speaking\_sps | 0.894 | 422 | <0.001 |
| 5 | r\_rate | r\_intel | -0.141 | 422 | 0.004 |
| 6 | intelligibility | age\_months | 0.514 | 386 | <0.001 |
| 6 | speaking\_sps | age\_months | 0.466 | 386 | <0.001 |
| 6 | speaking\_sps | intelligibility | 0.230 | 386 | <0.001 |
| 6 | r\_intel | age\_months | 0.000 | 386 | >0.999 |
| 6 | r\_intel | intelligibility | 0.826 | 386 | <0.001 |
| 6 | r\_intel | speaking\_sps | -0.026 | 386 | 0.616 |
| 6 | r\_rate | age\_months | 0.000 | 386 | >0.999 |
| 6 | r\_rate | intelligibility | -0.024 | 386 | 0.639 |
| 6 | r\_rate | speaking\_sps | 0.883 | 386 | <0.001 |
| 6 | r\_rate | r\_intel | -0.029 | 386 | 0.570 |
| 7 | intelligibility | age\_months | 0.525 | 372 | <0.001 |
| 7 | speaking\_sps | age\_months | 0.508 | 372 | <0.001 |
| 7 | speaking\_sps | intelligibility | 0.214 | 372 | <0.001 |
| 7 | r\_intel | age\_months | 0.000 | 372 | >0.999 |
| 7 | r\_intel | intelligibility | 0.829 | 372 | <0.001 |
| 7 | r\_intel | speaking\_sps | -0.082 | 372 | 0.113 |
| 7 | r\_rate | age\_months | 0.000 | 372 | >0.999 |
| 7 | r\_rate | intelligibility | -0.080 | 372 | 0.125 |
| 7 | r\_rate | speaking\_sps | 0.857 | 372 | <0.001 |
| 7 | r\_rate | r\_intel | -0.096 | 372 | 0.064 |

## Beta regression model

Analyses were carried in the R programming language (vers. 4.4.0, R Core Team, 2024). HMC sampling was performed using Stan (vers. 2.34.1, Stan Development Team, 2024) through the brms R package (vers. 2.21.0, Bürkner, 2017) and the cmdstanr backend (vers. 0.7.1, Gabry, Češnovar, & Johnson, 2024). We compared models with approximate leave-one-out cross validation with the loo R package (vers. 2.7.0, Vehtari, Gelman, & Gabry, 2017).

I wrote a custom function for fitting beta regression models. A user passes in the data and a flavor which flavor determines which model formula to use. Models were seeded using dates (seed) for reproducibility.

library(splines)  
library(brms)  
fit\_beta\_model <- function(  
 data,  
 flavor = NULL,  
 priors = NULL,  
 formula = NULL,  
 file = NULL,  
 seed = 20221013,  
 adapt\_delta = .8,  
 iter = 2000  
) {  
 formulas <- list(  
 ri\_main\_interaction = bf(  
 mwi ~ ns(age\_48, knots = c(9, 25), Boundary.knots = c(-18, 71)) +  
 tocs\_length +  
 speaking\_sps\_3 +  
 speaking\_sps\_3:tocs\_length +  
 (1 | child),  
 phi ~ ns(age\_48, knots = 17, Boundary.knots = c(-18, 71)),  
 family = Beta()  
 ),  
 ri\_no\_main\_length = bf(  
 mwi ~ ns(age\_48, knots = c(9, 25), Boundary.knots = c(-18, 71)) +  
 speaking\_sps +  
 (1 | child),  
 phi ~ ns(age\_48, knots = 17, Boundary.knots = c(-18, 71)),  
 family = Beta()  
 ),  
 ri\_no\_main\_interaction = bf(  
 mwi ~ ns(age\_48, knots = c(9, 25), Boundary.knots = c(-18, 71)) +  
 tocs\_length +  
 speaking\_sps\_3 +  
 (1 | child),  
 phi ~ ns(age\_48, knots = 17, Boundary.knots = c(-18, 71)),  
 family = Beta()  
 ),  
 rs\_for\_rate = bf(  
 mwi ~ ns(age\_48, knots = c(9, 25), Boundary.knots = c(-18, 71)) +  
 tocs\_length +  
 speaking\_sps\_3 +  
 speaking\_sps\_3:tocs\_length +  
 (speaking\_sps\_3 | child),  
 phi ~ ns(age\_48, knots = 17, Boundary.knots = c(-18, 71)),  
 family = Beta()  
 ),  
 rs\_monotonic\_length = bf(  
 mwi ~ ns(age\_48, knots = c(9, 25), Boundary.knots = c(-18, 71)) +  
 mo(tocs\_level) \* speaking\_sps\_3 +  
 (mo(tocs\_level) | child),  
 phi ~ ns(age\_48, knots = 17, Boundary.knots = c(-18, 71)),  
 family = Beta()  
 ),  
 rs\_for\_length = bf(  
 mwi ~ ns(age\_48, knots = c(9, 25), Boundary.knots = c(-18, 71)) +  
 tocs\_length +  
 speaking\_sps\_3 +  
 speaking\_sps\_3:tocs\_length +  
 (tocs\_length | child),  
 phi ~ ns(age\_48, knots = 17, Boundary.knots = c(-18, 71)),  
 family = Beta()  
 ),  
 rs\_for\_length\_plus\_rate = bf(  
 mwi ~ ns(age\_48, knots = c(9, 25), Boundary.knots = c(-18, 71)) +  
 tocs\_length +  
 speaking\_sps\_3 +  
 speaking\_sps\_3:tocs\_length +  
 (speaking\_sps\_3 + tocs\_length | child),  
 phi ~ ns(age\_48, knots = 17, Boundary.knots = c(-18, 71)),  
 family = Beta()  
 ),  
 rs\_for\_length\_colon\_rate = bf(  
 mwi ~ ns(age\_48, knots = c(9, 25), Boundary.knots = c(-18, 71)) +  
 tocs\_length +  
 speaking\_sps\_3 +  
 speaking\_sps\_3:tocs\_length +  
 (speaking\_sps\_3:tocs\_length | child),  
 phi ~ ns(age\_48, knots = 17, Boundary.knots = c(-18, 71)),  
 family = Beta()  
 ),  
 rs\_for\_length\_colon\_rate\_no\_intercept = bf(  
 mwi ~ ns(age\_48, knots = c(9, 25), Boundary.knots = c(-18, 71)) +  
 tocs\_length +  
 speaking\_sps\_3 +  
 speaking\_sps\_3:tocs\_length +  
 (0 + speaking\_sps\_3:tocs\_length | child),  
 phi ~ ns(age\_48, knots = 17, Boundary.knots = c(-18, 71)),  
 family = Beta()  
 ),  
 rs\_for\_length\_by\_rate = bf(  
 mwi ~ ns(age\_48, knots = c(9, 25), Boundary.knots = c(-18, 71)) +  
 tocs\_length +  
 speaking\_sps\_3 +  
 speaking\_sps\_3:tocs\_length +  
 (speaking\_sps\_3 \* tocs\_length | child),  
 phi ~ ns(age\_48, knots = 17, Boundary.knots = c(-18, 71)),  
 family = Beta()  
 )  
 )  
  
 files <- c(  
 ri\_main\_interaction = "models/ri\_main\_interaction",  
 ri\_no\_main\_length = "models/ri\_no\_main\_length",  
 ri\_no\_main\_interaction = "models/ri\_no\_main\_interaction",  
 rs\_for\_rate = "models/rs\_for\_rate",  
 rs\_for\_length = "models/rs\_for\_length",  
 rs\_for\_length\_plus\_rate = "models/rs\_for\_length\_plus\_rate",  
 rs\_for\_length\_colon\_rate = "models/rs\_for\_length\_colon\_rate",  
 rs\_for\_length\_colon\_rate\_no\_intercept = "models/rs\_for\_length\_colon\_rate\_no\_intercept",  
 rs\_for\_length\_by\_rate = "models/rs\_for\_length\_by\_rate",  
 rs\_monotonic\_length = "models/rs\_monotonic\_length"  
 )  
  
 formula <- formula %||% formulas[[flavor]]  
 file <- file %||% files[[flavor]]  
  
  
 if (is.null(priors)) {  
 priors <- c(  
 set\_prior("normal(0, 2.5)", class = "b"),  
 set\_prior("normal(0, 1)", class = "sd")  
 )  
 has\_element <- function(x, y) any(x %in% y)  
 has\_correlation <- get\_prior(formula, data = data) |>  
 getElement("class") |>  
 has\_element("cor")  
 if (has\_correlation) {  
 priors <- c(  
 priors,  
 set\_prior("lkj(2)", class = "cor")  
 )  
 }  
 }  
  
  
 t <- brm(  
 formula = formula,  
 data = data,  
 prior = priors,  
 backend = "cmdstanr",  
 seed = seed,  
 file = file,  
 file\_refit = "on\_change",  
 chains = 4,  
 cores = 4,  
 iter = iter,  
 save\_pars = save\_pars(all = TRUE),  
 control = list(adapt\_delta = adapt\_delta),  
 refresh = 50  
 )  
 add\_criterion(t, c("loo", "loo\_R2"))  
}

The models compared with in the manuscript were as follows:

loo\_comparison <- targets::tar\_read("loo\_comparison")  
labels <- tibble::tribble(  
 ~model, ~random\_effects,  
 "model\_rs\_length", "1 + length | child",   
 "model\_rs\_rate", "1 + rate | child",   
 "model\_rs\_length\_plus\_rate", "1 + length + rate | child",   
 "model\_main", "1 | child",   
 "model\_rs\_length\_colon\_rate", "1 + length:rate | child",   
 "model\_rs\_monotonic\_length", "1 + monotonic-length | child",   
)  
  
loo\_table\_data <- loo\_comparison |>   
 inner\_join(labels, by = "model") |>   
 mutate(  
 across(c(2:3), \(x) round(x, 1)),  
 across(c(4:9), \(x) round(x, 0)),  
 across(c(loo\_r2\_mean), \(x) round(x, 2)),  
 across(c(loo\_r2\_sd), \(x) round(x, 3)),  
 ) |>   
 relocate(model, random\_effects)   
  
loo\_table\_data |>   
 knitr::kable()

| model | random\_effects | elpd\_diff | se\_diff | elpd\_loo | se\_elpd\_loo | p\_loo | se\_p\_loo | looic | se\_looic | loo\_r2\_mean | loo\_r2\_sd |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| model\_rs\_length | 1 + length | child | 0.0 | 0.0 | 3814 | 49 | 852 | 20 | -7628 | 98 | 0.84 | 0.009 |
| model\_rs\_length\_plus\_rate | 1 + length + rate | child | -28.8 | 5.3 | 3785 | 50 | 784 | 20 | -7571 | 100 | 0.82 | 0.010 |
| model\_rs\_monotonic\_length | 1 + monotonic-length | child | -98.9 | 9.1 | 3715 | 51 | 523 | 16 | -7431 | 102 | 0.79 | 0.011 |
| model\_rs\_rate | 1 + rate | child | -101.4 | 9.8 | 3713 | 52 | 490 | 15 | -7426 | 104 | 0.78 | 0.011 |
| model\_main | 1 | child | -107.2 | 9.7 | 3707 | 52 | 442 | 15 | -7414 | 104 | 0.78 | 0.011 |
| model\_rs\_length\_colon\_rate | 1 + length:rate | child | -107.9 | 9.8 | 3706 | 52 | 479 | 15 | -7413 | 104 | 0.78 | 0.011 |

All models (except model\_rs\_monotonic\_length) formula includes a categorical effect of utterance length tocs\_length, speaking rate centered at 3 sps tocs\_length, an adjustment to the rate effect for each length speaing\_sps\_3:tocs\_length, and the 3-df spline for age ns(...). The phi ~ regression lines allows the dispersion of the beta distribution to change with age. What differentiated these models were the variables used in the random (by-child) effects. model\_rs\_monotonic\_length model\_rs\_monotonic\_length model applies the method in Bürkner & Charpentier (2020), using mo() to mark a variable as monotonic.

The winning model on the basis of model comparison was model\_rs\_length (by-child length effects).

model\_rs\_length <- targets::tar\_read(model\_rs\_length)   
summary(model\_rs\_length)  
#> Family: beta   
#> Links: mu = logit; phi = log   
#> Formula: mwi ~ ns(age\_48, knots = c(9, 25), Boundary.knots = c(-18, 71)) + tocs\_length + speaking\_sps\_3 + speaking\_sps\_3:tocs\_length + (tocs\_length | child)   
#> phi ~ ns(age\_48, knots = 17, Boundary.knots = c(-18, 71))  
#> Data: data (Number of observations: 2196)   
#> Draws: 4 chains, each with iter = 3000; warmup = 1500; thin = 1;  
#> total post-warmup draws = 6000  
#>   
#> Multilevel Hyperparameters:  
#> ~child (Number of levels: 538)   
#> Estimate Est.Error l-95% CI u-95% CI Rhat  
#> sd(Intercept) 0.80 0.04 0.71 0.88 1.01  
#> sd(tocs\_length4) 0.50 0.09 0.28 0.65 1.02  
#> sd(tocs\_length5) 0.38 0.11 0.11 0.55 1.02  
#> sd(tocs\_length6) 0.44 0.09 0.22 0.59 1.02  
#> sd(tocs\_length7) 0.54 0.09 0.34 0.68 1.02  
#> cor(Intercept,tocs\_length4) -0.27 0.10 -0.43 -0.02 1.01  
#> cor(Intercept,tocs\_length5) -0.45 0.10 -0.62 -0.23 1.00  
#> cor(tocs\_length4,tocs\_length5) 0.41 0.18 -0.08 0.66 1.01  
#> cor(Intercept,tocs\_length6) -0.31 0.13 -0.49 -0.01 1.00  
#> cor(tocs\_length4,tocs\_length6) 0.57 0.13 0.26 0.77 1.00  
#> cor(tocs\_length5,tocs\_length6) 0.53 0.19 0.01 0.77 1.01  
#> cor(Intercept,tocs\_length7) -0.26 0.12 -0.44 0.03 1.01  
#> cor(tocs\_length4,tocs\_length7) 0.54 0.12 0.25 0.73 1.00  
#> cor(tocs\_length5,tocs\_length7) 0.45 0.18 -0.03 0.70 1.01  
#> cor(tocs\_length6,tocs\_length7) 0.86 0.10 0.67 0.96 1.01  
#> Bulk\_ESS Tail\_ESS  
#> sd(Intercept) 191 280  
#> sd(tocs\_length4) 120 163  
#> sd(tocs\_length5) 125 181  
#> sd(tocs\_length6) 136 160  
#> sd(tocs\_length7) 144 163  
#> cor(Intercept,tocs\_length4) 290 306  
#> cor(Intercept,tocs\_length5) 2818 915  
#> cor(tocs\_length4,tocs\_length5) 369 270  
#> cor(Intercept,tocs\_length6) 380 273  
#> cor(tocs\_length4,tocs\_length6) 662 480  
#> cor(tocs\_length5,tocs\_length6) 423 248  
#> cor(Intercept,tocs\_length7) 232 239  
#> cor(tocs\_length4,tocs\_length7) 718 463  
#> cor(tocs\_length5,tocs\_length7) 384 241  
#> cor(tocs\_length6,tocs\_length7) 567 280  
#>   
#> Regression Coefficients:  
#> Estimate Est.Error l-95% CI  
#> Intercept -0.42 0.11 -0.64  
#> phi\_Intercept 3.82 0.39 3.08  
#> nsage\_48knotsEQc925Boundary.knotsEQcM18711 3.25 0.15 2.95  
#> nsage\_48knotsEQc925Boundary.knotsEQcM18712 6.80 0.28 6.26  
#> nsage\_48knotsEQc925Boundary.knotsEQcM18713 3.63 0.21 3.22  
#> tocs\_length4 -0.10 0.05 -0.20  
#> tocs\_length5 -0.31 0.05 -0.41  
#> tocs\_length6 -0.17 0.06 -0.28  
#> tocs\_length7 -0.51 0.06 -0.63  
#> speaking\_sps\_3 -0.08 0.11 -0.31  
#> tocs\_length4:speaking\_sps\_3 -0.05 0.13 -0.30  
#> tocs\_length5:speaking\_sps\_3 -0.45 0.12 -0.69  
#> tocs\_length6:speaking\_sps\_3 -0.25 0.13 -0.51  
#> tocs\_length7:speaking\_sps\_3 -0.34 0.13 -0.59  
#> phi\_nsage\_48knotsEQ17Boundary.knotsEQcM18711 0.13 0.67 -1.22  
#> phi\_nsage\_48knotsEQ17Boundary.knotsEQcM18712 0.53 0.23 0.07  
#> u-95% CI Rhat Bulk\_ESS Tail\_ESS  
#> Intercept -0.19 1.00 2038 3248  
#> phi\_Intercept 4.59 1.02 118 212  
#> nsage\_48knotsEQc925Boundary.knotsEQcM18711 3.55 1.00 2596 3175  
#> nsage\_48knotsEQc925Boundary.knotsEQcM18712 7.35 1.00 1873 3560  
#> nsage\_48knotsEQc925Boundary.knotsEQcM18713 4.03 1.00 3657 3703  
#> tocs\_length4 -0.01 1.00 3297 3803  
#> tocs\_length5 -0.22 1.00 3023 4015  
#> tocs\_length6 -0.06 1.00 3206 4399  
#> tocs\_length7 -0.41 1.00 2993 3682  
#> speaking\_sps\_3 0.14 1.00 1971 2564  
#> tocs\_length4:speaking\_sps\_3 0.21 1.00 3431 4386  
#> tocs\_length5:speaking\_sps\_3 -0.21 1.00 3030 4074  
#> tocs\_length6:speaking\_sps\_3 0.01 1.00 3197 3824  
#> tocs\_length7:speaking\_sps\_3 -0.08 1.00 2527 3713  
#> phi\_nsage\_48knotsEQ17Boundary.knotsEQcM18711 1.33 1.02 140 352  
#> phi\_nsage\_48knotsEQ17Boundary.knotsEQcM18712 0.97 1.01 290 843  
#>   
#> 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).

However, it bears noting that a more appropriate model comparison would compare them using leave-one-child-out cross-validation. This approach was computational prohibitive because it would require fitting 538 versions of each model.

### Marginal means

With a model in hand, we compute marginal predictions by simulating new children from each posterior sample. The basic recipe is as follows.

Create a grid of data to make predictions for, giving child an out-of-sample value. To make things easier, I break the data into batches in a list.

newdata <- tidyr::crossing(  
 age\_48 = (3:7) \* 12 - 48,  
 tocs\_length = c("3", "4", "5", "6", "7"),  
 speaking\_sps\_3 = c(0, .5),  
 child = "fake"  
)   
newdata  
#> # A tibble: 50 × 4  
#> age\_48 tocs\_length speaking\_sps\_3 child  
#> <dbl> <chr> <dbl> <chr>  
#> 1 -12 3 0 fake   
#> 2 -12 3 0.5 fake   
#> 3 -12 4 0 fake   
#> 4 -12 4 0.5 fake   
#> 5 -12 5 0 fake   
#> 6 -12 5 0.5 fake   
#> 7 -12 6 0 fake   
#> 8 -12 6 0.5 fake   
#> 9 -12 7 0 fake   
#> 10 -12 7 0.5 fake   
#> # ℹ 40 more rows

newdata\_batches <- newdata |>  
 split(~ age\_48:speaking\_sps\_3)

Next I sample 1000 children from each draw. The 1000 childen are 1000 draws from the multivariate normal distribution used in the random effects. First I extract the variance-covariance matrix of the random effects. I use an rvar() which allows me to write code for the 6000 x 5 x 5 array of posterior draws as if it were a 5 x 5 matrix. The random effects are centered at 0 so each row of newdata is given a mean of 0.

cov <- model\_rs\_length |>  
 VarCorr(summary = FALSE) |>  
 \_$child$cov |>  
 posterior::rvar()  
cov  
#> rvar<6000>[5,5] mean ± sd:  
#> Intercept tocs\_length4 tocs\_length5 tocs\_length6   
#> Intercept 0.64 ± 0.071 -0.11 ± 0.054 -0.14 ± 0.055 -0.11 ± 0.056   
#> tocs\_length4 -0.11 ± 0.054 0.26 ± 0.084 0.09 ± 0.051 0.13 ± 0.056   
#> tocs\_length5 -0.14 ± 0.055 0.09 ± 0.051 0.16 ± 0.075 0.10 ± 0.052   
#> tocs\_length6 -0.11 ± 0.056 0.13 ± 0.056 0.10 ± 0.052 0.20 ± 0.075   
#> tocs\_length7 -0.12 ± 0.062 0.15 ± 0.060 0.10 ± 0.055 0.21 ± 0.070   
#> tocs\_length7   
#> Intercept -0.12 ± 0.062   
#> tocs\_length4 0.15 ± 0.060   
#> tocs\_length5 0.10 ± 0.055   
#> tocs\_length6 0.21 ± 0.070   
#> tocs\_length7 0.30 ± 0.087

means <- rep(0, ncol(cov))  
  
rows\_batches <- vapply(newdata\_batches, nrow, integer(1))  
stopifnot(all(rows\_batches == length(means)))  
  
num\_children <- 1000  
sim\_children <- posterior::rdo(mvtnorm::rmvnorm(num\_children, means, cov))

Given the simulated children, I walk through each batch of new data, get the linear predictions (predictions in logits using just the fixed effects), add the simulated children to each linear prediction, convert the logits into proportions and then average over the 1000 children for that draw.

l <- newdata\_batches |>  
 lapply(tidybayes::add\_linpred\_rvars, model\_rs\_length, re\_formula = NA) |>  
 lapply(  
 function(x) {  
 x$.marginal <- (sim\_children + t(x$.linpred)) |>  
 brms::inv\_logit\_scaled() |>  
 posterior::rvar\_apply(2, posterior::rvar\_mean)  
 x  
 }  
 ) |>  
 bind\_rows()

This part is time-consuming so I load in the precomputed value.

targets::tar\_read(marginal\_means\_model\_rs\_length)  
#> # A tibble: 50 × 6  
#> age\_48 tocs\_length speaking\_sps\_3 child .linpred .marginal  
#> <dbl> <chr> <dbl> <chr> <rvar[1d]> <rvar[1d]>  
#> 1 -12 3 0 fake 0.231 ± 0.080 0.55 ± 0.018  
#> 2 -12 4 0 fake 0.130 ± 0.080 0.53 ± 0.019  
#> 3 -12 5 0 fake -0.078 ± 0.077 0.48 ± 0.019  
#> 4 -12 6 0 fake 0.061 ± 0.082 0.51 ± 0.020  
#> 5 -12 7 0 fake -0.283 ± 0.084 0.43 ± 0.020  
#> 6 0 3 0 fake 1.448 ± 0.065 0.78 ± 0.011  
#> 7 0 4 0 fake 1.347 ± 0.063 0.78 ± 0.010  
#> 8 0 5 0 fake 1.138 ± 0.058 0.75 ± 0.011  
#> 9 0 6 0 fake 1.278 ± 0.064 0.77 ± 0.011  
#> 10 0 7 0 fake 0.933 ± 0.066 0.71 ± 0.013  
#> # ℹ 40 more rows

The .marginal column has 6000 marginal means for each row of the prediction grid. Subtracting between two rows would give 6000 differences. Estimates and comparisons were computed by taking the median and 95% quantile intervals of the means and differences.

## Logistic regression model

For the within-child effects of rate, we use the item-level data for the 5-, 6-, and 7-word utterances.

data\_model\_by\_item\_anon <- targets::tar\_read(data\_model\_by\_item\_anon)  
data\_model\_by\_item\_anon  
#> # A tibble: 11,282 × 18  
#> child item speaking\_sps artic\_sps age\_months tocs\_level length\_longest  
#> <chr> <chr> <dbl> <dbl> <dbl> <dbl> <dbl>  
#> 1 c130 S5T01 2.47 2.47 34 5 5  
#> 2 c130 S5T02 2.26 2.26 34 5 5  
#> 3 c130 S5T03 2.55 2.55 34 5 5  
#> 4 c130 S5T04 2.77 2.77 34 5 5  
#> 5 c130 S5T05 3.01 3.01 34 5 5  
#> 6 c130 S5T06 1.89 2.12 34 5 5  
#> 7 c130 S5T07 2.39 2.39 34 5 5  
#> 8 c130 S5T08 2.20 2.50 34 5 5  
#> 9 c130 S5T09 2.83 2.83 34 5 5  
#> 10 c227 S5T01 3 3 35 5 5  
#> # ℹ 11,272 more rows  
#> # ℹ 11 more variables: age\_bin <dbl>, intelligibility <dbl>,  
#> # n\_words\_correct <dbl>, n\_words\_total <dbl>, mwi <dbl>,  
#> # speaking\_sps\_3 <dbl>, artic\_sps\_3 <dbl>, age\_48 <dbl>, tocs\_length <fct>,  
#> # tocs\_level\_3 <dbl>, child\_mean\_rate <dbl>

New here are

* item: item identifier
* n\_words\_correct, n\_words\_total: number of words correctly transcribed by the listeners and number of words total, so the intelligibility is the proportion of words correctly transcribed
* child\_mean\_rate: a child’s average speaking rate over all items

The fitted model was the following:

model\_logistic <- targets::tar\_read(  
 "model\_rs\_rate\_ri\_item\_no\_length\_binom\_mundlak"  
)  
model\_logistic  
#> Family: binomial   
#> Links: mu = logit   
#> Formula: n\_words\_correct | trials(n\_words\_total) ~ ns(age\_48, knots = c(9, 25), Boundary.knots = c(-18, 71)) + speaking\_sps\_3 + child\_mean\_rate + (speaking\_sps\_3 | child) + (1 | item)   
#> Data: data (Number of observations: 11282)   
#> Draws: 4 chains, each with iter = 3000; warmup = 1500; thin = 1;  
#> total post-warmup draws = 6000  
#>   
#> Multilevel Hyperparameters:  
#> ~child (Number of levels: 422)   
#> Estimate Est.Error l-95% CI u-95% CI Rhat  
#> sd(Intercept) 0.85 0.03 0.78 0.92 1.00  
#> sd(speaking\_sps\_3) 0.50 0.03 0.44 0.56 1.00  
#> cor(Intercept,speaking\_sps\_3) -0.15 0.07 -0.28 -0.01 1.00  
#> Bulk\_ESS Tail\_ESS  
#> sd(Intercept) 1367 2649  
#> sd(speaking\_sps\_3) 2518 3793  
#> cor(Intercept,speaking\_sps\_3) 1917 2597  
#>   
#> ~item (Number of levels: 29)   
#> Estimate Est.Error l-95% CI u-95% CI Rhat Bulk\_ESS Tail\_ESS  
#> sd(Intercept) 0.71 0.09 0.55 0.92 1.00 1108 2461  
#>   
#> Regression Coefficients:  
#> Estimate Est.Error l-95% CI u-95% CI  
#> Intercept 0.49 0.33 -0.16 1.14  
#> nsage\_48knotsEQc925Boundary.knotsEQcM18711 2.93 0.20 2.52 3.33  
#> nsage\_48knotsEQc925Boundary.knotsEQcM18712 4.72 0.66 3.42 6.02  
#> nsage\_48knotsEQc925Boundary.knotsEQcM18713 3.30 0.27 2.74 3.83  
#> speaking\_sps\_3 -0.23 0.04 -0.30 -0.16  
#> child\_mean\_rate -0.12 0.15 -0.42 0.17  
#> Rhat Bulk\_ESS Tail\_ESS  
#> Intercept 1.00 879 1846  
#> nsage\_48knotsEQc925Boundary.knotsEQcM18711 1.00 1064 1742  
#> nsage\_48knotsEQc925Boundary.knotsEQcM18712 1.00 1098 1950  
#> nsage\_48knotsEQc925Boundary.knotsEQcM18713 1.00 1810 2818  
#> speaking\_sps\_3 1.00 4260 4387  
#> child\_mean\_rate 1.00 1031 2004  
#>   
#> 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).

The random effects include a by-child random intercept, by-child rate effects, and by-item intercepts. The same age-spline is used as before for the fixed effects. The other two predictors are the speaking rate on the given utterance and the child’s average speaking rate. The child\_mean\_rate is meant to absorb child-level reflect a child’s habitual rate so that the speaking\_sps\_3 can estimate the within child effects of rate.

We can extract the posterior means of each child’s rate coefficient like so:

# Posterior draws of the slopes  
child\_slopes <- model\_logistic |>  
 coef(summary = FALSE) |>  
 getElement("child") |>  
 \_[, , c("Intercept", "speaking\_sps\_3")] |>   
 posterior::rvar()  
  
slope\_means <- mean(child\_slopes)[, "speaking\_sps\_3"]  
  
f\_percent <- scales::label\_percent(.1)  
lslopes <- list(  
 n\_positive = sum(slope\_means > 0),  
 n\_negative = sum(slope\_means < 0),  
 pct\_positive = f\_percent(sum(slope\_means > 0) / length(slope\_means)),  
 pct\_negative = f\_percent(sum(slope\_means < 0) / length(slope\_means))  
)  
str(lslopes)  
#> List of 4  
#> $ n\_positive : int 107  
#> $ n\_negative : int 315  
#> $ pct\_positive: chr "25.4%"  
#> $ pct\_negative: chr "74.6%"

The plotted lines used tidybayes::add\_epred\_rvars() to get child-level predictions when trials is set to 1. We take a grid of 50 points along each child’s observed speaking rate, get posterior expectations for each of those point and plot the mean of those expectations for each child.

data\_binom\_rate\_grid <- data\_model\_by\_item\_anon |>  
 group\_by(child, age\_months, age\_bin, age\_48) |>  
 reframe(  
 child\_mean\_rate = mean(speaking\_sps\_3),  
 speaking\_sps\_3 = c(  
 seq(min(speaking\_sps\_3), max(speaking\_sps\_3), length.out = 50),  
 child\_mean\_rate  
 ),  
 item = "fake",  
 n\_words\_total = 1  
 )  
  
data\_binom\_rate\_grid <- tidybayes::add\_epred\_rvars(  
 data\_binom\_rate\_grid,  
 model\_logistic,  
 allow\_new\_levels = TRUE  
)

## References

Bürkner, P.-C. (2017). brms: An R package for Bayesian multilevel models using Stan. *Journal of Statistical Software*, *80*(1), 1–28. doi:[10.18637/jss.v080.i01](https://doi.org/10.18637/jss.v080.i01)

Bürkner, P.-C., & Charpentier, E. (2020). Modelling monotonic effects of ordinal predictors in Bayesian regression models. *British Journal of Mathematical and Statistical Psychology*, *73*(3), 420–451. doi:[10.1111/bmsp.12195](https://doi.org/10.1111/bmsp.12195)

Gabry, J., Češnovar, R., & Johnson, A. (2024). *cmdstanr: R interface to CmdStan*. Retrieved from <https://mc-stan.org/cmdstanr/>

R Core Team. (2024). *R: A language and environment for statistical computing*. Vienna, Austria: R Foundation for Statistical Computing. Retrieved from <https://www.R-project.org/>

Stan Development Team. (2024). *Stan modeling language users guide and reference manual*. Retrieved from <https://mc-stan.org>

Vehtari, A., Gelman, A., & Gabry, J. (2017). Practical Bayesian model evaluation using leave-one-out cross-validation and WAIC. *Statistics and Computing*, *27*, 1413–1432. doi:[10.1007/s11222-016-9696-4](https://doi.org/10.1007/s11222-016-9696-4)