Bayesian linear (additive) mixed model with spline-based smooths

Exploration for the `{brms.mmrm}` package - DRAFT

PUBLISHED
February 25, 2024

About

A Bayesian linear (additive) mixed model fitted to an example data set using the `{brms}` and the `{mgcv}` package. The models contain spline-based smooths, stratified by treatment arm. Contrasts at custom time points are calculated (using the `{emmeans}` package) and compared across the two estimations methods/ packages.

Results from this type of model may be useful to construct priors for MMRMs when the historical data and the new data are misaligned w.r.t. the visit schedule.

The computation of prediction intervals with and without `{emmeans}` (among other things) warrants a closer look.

Prerequisites

Load general packages and modeling packages:

```
> packages <- c("dplyr", "tibble", "tidyr", "ggplot2", "gt",
+               "brms", "mgcv", "emmeans")
> invisible(lapply(packages, library, character.only = TRUE))
```

Set seed:

```
> set.seed(123)
```

Data

```
> data("fev_data", package = "mmrm")
```

Preprocessing

Adding a fifth visit:

```
> fev_data2 <- bind_rows(
+   fev_data,
+   filter(fev_data, AVISIT == "VIS4") |> mutate(VISITN = 5)
+   ) |>
```

```
+   mutate(AVISIT = paste0("VIS", VISITN) |> factor()) |>
+   arrange(USUBJID, AVISIT)
```

Making trajectories (clearly) non-linear:

```
> fev_data2 <- mutate(
+   fev_data2,
+   FEV1 = ifelse(AVISIT == "VIS1", FEV1 -  5, FEV1),
+   FEV1 = ifelse(AVISIT == "VIS2", FEV1 + 10, FEV1),
+   FEV1 = ifelse(AVISIT == "VIS3", FEV1 + 11, FEV1),
+   FEV1 = ifelse(AVISIT == "VIS4", FEV1 + 18, FEV1),
+   FEV1 = ifelse(AVISIT == "VIS5", FEV1 + 20, FEV1)
+   )
```

Increasing differences between treatment groups:

```
> fev_data2 <- mutate(
+   fev_data2,
+   FEV1 = ifelse(AVISIT == "VIS1" & ARMCD == "TRT", FEV1 +  2, FEV1),
+   FEV1 = ifelse(AVISIT == "VIS2" & ARMCD == "TRT", FEV1 +  6, FEV1),
+   FEV1 = ifelse(AVISIT == "VIS3" & ARMCD == "TRT", FEV1 +  8, FEV1),
+   FEV1 = ifelse(AVISIT == "VIS4" & ARMCD == "TRT", FEV1 + 10, FEV1),
+   FEV1 = ifelse(AVISIT == "VIS5" & ARMCD == "TRT", FEV1 + 12, FEV1)
+   )
```

Adding a continuous time variable:

```
> fev_data2 <- mutate(fev_data2,
+   WEEK = case_when(
+     AVISIT == "VIS1" ~ 2,
+     AVISIT == "VIS2" ~ 4,
+     AVISIT == "VIS3" ~ 8,
+     AVISIT == "VIS4" ~ 12,
+     AVISIT == "VIS5" ~ 16,
+     .default = NA
+     )
+   )
```

Adding a ‘change from baseline’ variable:

```
> fev_data2 <- mutate(fev_data2, FEV1_CHG = FEV1 - FEV1_BL)
```

Removing unnecessary variables:

```
> fev_data2 <- select(fev_data2, -c(RACE, SEX, WEIGHT, AVISIT, VISITN, VISITN2))
```

Descriptive statistics

```
> summary(fev_data2)
    USUBJID     ARMCD        FEV1_BL           FEV1             WEEK
```

```
 PT1     :  5   PBO:525   Min.   :14.34   Min.   : 14.28   Min.   : 2.0
 PT2     :  5   TRT:475   1st Qu.:34.26   1st Qu.: 46.83   1st Qu.: 4.0
 PT3     :  5             Median :40.30   Median : 59.24   Median : 8.0
 PT4     :  5             Mean   :40.19   Mean   : 58.38   Mean   : 8.4
 PT5     :  5             3rd Qu.:46.61   3rd Qu.: 71.78   3rd Qu.:12.0
 PT6     :  5             Max.   :60.24   Max.   :116.08   Max.   :16.0
 (Other):970                              NA's   :329
    FEV1_CHG
 Min.   :-31.438
 1st Qu.:  4.107
 Median : 18.793
 Mean   : 18.270
 3rd Qu.: 31.701
 Max.   : 78.651
 NA's   :329
```

```
> dim(fev_data2)
[1] 1000    6
```

Plot

```
> fev_data2 |>
+   ggplot(aes(
+     x = WEEK,
+     y = FEV1_CHG,
+     group = interaction(ARMCD, WEEK),
+     fill = factor(ARMCD)
+   )) +
+   geom_hline(yintercept = 0,
+              col = "grey",
+              linewidth = 1.2) +
+   geom_boxplot(na.rm = TRUE, position = position_dodge(width = 1.7)) +
+   labs(x = "Week",
+        y = "Change from baseline",
+        fill = "Treatment") +
+   scale_fill_manual(values = c("darkgoldenrod2", "coral2")) +
+   scale_x_continuous(breaks = unique(fev_data2$WEEK)) +
+   theme_bw()
```

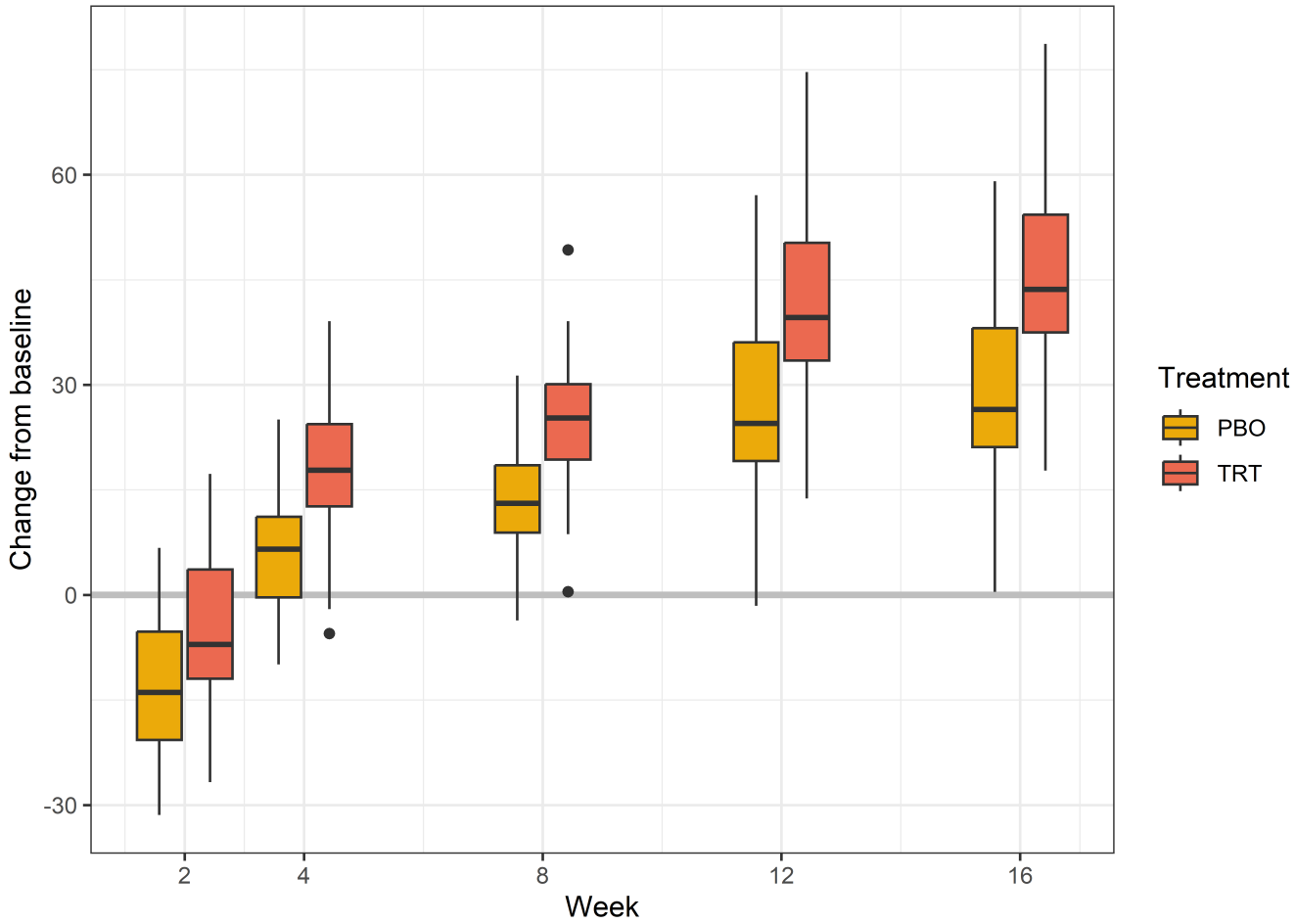

Figure: Boxplots by treatment arm and visit.

Analysis

Using `brms`

Fit model

```
> m_form_brms <- brms::bf(
+   FEV1_CHG ~ ARMCD + s(WEEK, by = ARMCD, k = 3) + (1 | USUBJID),
+   family = gaussian()
+   )
> m_fit_brms <- brms::brm(
+   formula = m_form_brms,
+   data = filter(fev_data2, !is.na(FEV1_CHG)),
+   iter = 2000,
+   warmup = 1000,
+   chains = 4,
+   cores = 4,
+   silent = 2,
+   refresh = 0,
+   seed = 123,
+   control = list(adapt_delta = 0.99)
+ )
```

Model summary

```
> summary(m_fit_brms)
 Family: gaussian
  Links: mu = identity; sigma = identity
Formula: FEV1_CHG ~ ARMCD + s(WEEK, by = ARMCD, k = 3) + (1 | USUBJID)
   Data: filter(fev_data2, !is.na(FEV1_CHG)) (Number of observations: 671)
  Draws: 4 chains, each with iter = 2000; warmup = 1000; thin = 1;
         total post-warmup draws = 4000

Smooth Terms:
                     Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
sds(sWEEKARMCDPBO_1)    40.31     24.01    16.59    96.80 1.00     2371     2231
sds(sWEEKARMCDTRT_1)    43.60     23.71    18.40   104.39 1.00     2125     2518

Group-Level Effects:
~USUBJID (Number of levels: 197)
              Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
sd(Intercept)     8.46      0.56     7.43     9.63 1.01     1143     2089

Population-Level Effects:
                 Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
Intercept           12.04      0.95    10.16    13.91 1.00      836     1340
ARMCDTRT            12.74      1.36    10.08    15.45 1.00      825     1210
sWEEK:ARMCDPBO_1   -13.24      0.46   -14.15   -12.34 1.00     4452     3245
sWEEK:ARMCDTRT_1   -16.84      0.43   -17.68   -16.01 1.00     4109     2609

Family Specific Parameters:
      Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
sigma     7.62      0.25     7.15     8.13 1.00     2752     3098

Draws were sampled using sampling(NUTS). 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).
```

Plot predictions

Samples from the posterior predictive distribution are used to compute marginal means along with uncertainty intervals (alternatively the `{emmeans}` package could be used):

```
> # Set up grid for predictions
> dat_pred_brms <- expand_grid(
+   WEEK = seq(min(fev_data2$WEEK), max(fev_data2$WEEK), by = 0.5),
+   ARMCD = unique(fev_data2$ARMCD)
+   )
> # Add mean estimate and uncertainty intervals from
> # posterior predictive distribution
> dat_pred_brms <- bind_cols(
+   dat_pred_brms,
+   predict(
+     object = m_fit_brms,
+     newdata = dat_pred_brms,
+     re_formula = NA,
```

```
+     summary = TRUE,
+     probs = c(0.025, 0.975)
+   ) |>
+     as_tibble()
+ )
```

Of note, the residual error is considered in the predictions from `predict()` / `posterior_predict()`. The posterior predictive distribution without consideration of the residual error is obtained by `posterior_epred.brmsfit()`. The uncertainty intervals are thus wider with the former functions.

```
> # Create plot
> ggplot(dat_pred_brms, aes(x = WEEK, y = Estimate, color = ARMCD)) +
+   geom_ribbon(aes(
+     ymin = `Q2.5`,
+     ymax = `Q97.5`,
+     fill = ARMCD),
+     linetype = 0,
+     alpha = 0.15
+   ) +
+   geom_line(linewidth = 1.2) +
+   scale_x_continuous(
+     breaks = seq(min(dat_pred_brms$WEEK), max(dat_pred_brms$WEEK), by = 2)
+   ) +
+   scale_color_manual(values = c("darkgoldenrod", "coral2")) +
+   scale_fill_manual(values = c("darkgoldenrod", "coral2"), guide = "none") +
+   labs(x = "Time (weeks)",
+        y = "Predicted change from baseline and 95% CrI",
+       color = "Treatment",
+        title = "Predictions from brms::brm() fit") +
+   theme_bw()
```

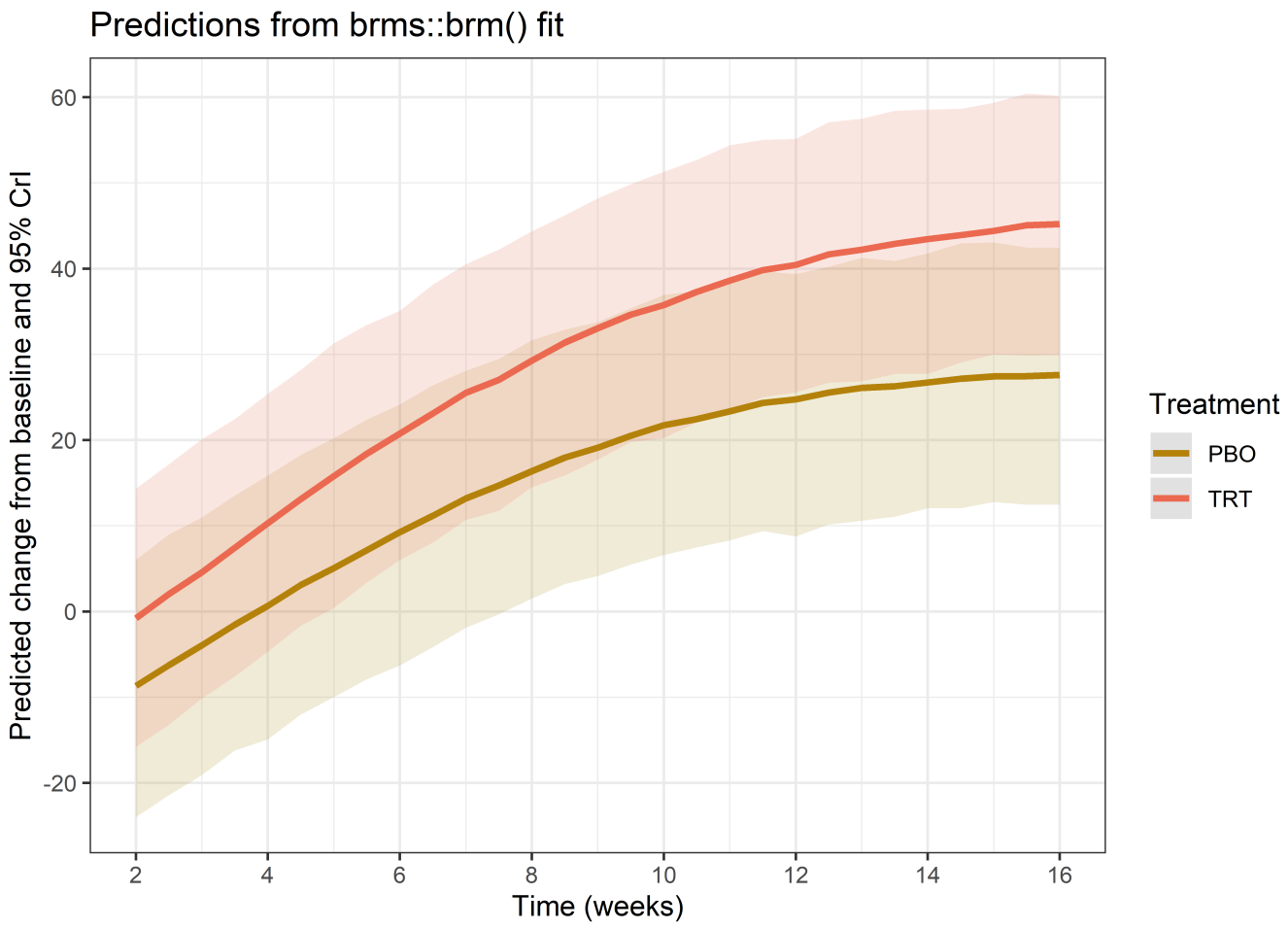

Figure: Marginal means from {brms} fit.

Contrasts

Contrasts at custom time points are computed (for convenience here the {emmeans} package is used):

```
> custom_time_points <- c(2, 4, 8, 12, 16)
> contrasts_brms <- emmeans(
+   object = m_fit_brms,
+   specs = ~ ARMCD | WEEK,
+   at = list(WEEK = custom_time_points)
+   ) |>
+   contrast(method = "pairwise") |>
+   as.data.frame()
> contrasts_brms |> gt()
```

contrast	WEEK	estimate	lower.HPD	upper.HPD
PBO - TRT	2	-7.770979	-10.93395	-4.355566
PBO - TRT	4	-9.682616	-12.45645	-6.795808
PBO - TRT	8	-13.085125	-16.13190	-9.932615
PBO - TRT	12	-15.690240	-18.82150	-12.799607

contrast	WEEK	estimate	lower.HPD	upper.HPD
PBO - TRT	16	-17.622324	-21.04430	-14.058599

Note that the credible interval here is a *highest posterior density interval*.

Using `mgcv`

Fit model

```
> m_form_mgcv <- formula(
+   FEV1_CHG ~ ARMCD + s(WEEK, by = ARMCD, k = 3) +
+     s(USUBJID, bs = "re", by = dum)
+   )
> m_fit_mgcv <- mgcv::gam(
+   formula = m_form_mgcv,
+   data = filter(fev_data2, !is.na(FEV1_CHG)) |> mutate(dum = 1),
+   method = "REML"
+ )
```

See [this StackExchange site] (https://stats.stackexchange.com/questions/131106/predicting-with-random-effects-in-mgcv-gam) on prediction with random effects in `mgcv::gam()`.

Model summary

```
> summary(m_fit_mgcv)

Family: gaussian
Link function: identity

Formula:
FEV1_CHG ~ ARMCD + s(WEEK, by = ARMCD, k = 3) + s(USUBJID, bs = "re",
    by = dum)

Parametric coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  11.9614     0.9336  12.812   <2e-16 ***
ARMCDTRT     12.7834     1.3555   9.431   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Approximate significance of smooth terms:
                     edf Ref.df       F p-value
s(WEEK):ARMCDPBO   1.988      2 495.768  <2e-16 ***
s(WEEK):ARMCDTRT   1.990      2 804.121  <2e-16 ***
s(USUBJID):dum   152.984    195   4.166  <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
R-sq.(adj) =  0.864   Deviance explained = 89.6%
-REML = 2470.7  Scale est. = 57.737    n = 671
```

Plot predictions

Marginal means and 95% confidence intervals are computed (alternatively the `{emmeans}` package could be used):

```
> # Set up grid for predictions
> dat_pred_mgcv <- expand_grid(
+   WEEK = seq(min(fev_data2$WEEK), max(fev_data2$WEEK), by = 0.5),
+   ARMCD = unique(fev_data2$ARMCD)
+   ) |>
+   mutate(
+     `(Intercept)` = 1,
+     USUBJID = "PT1",
+     dum = 0
+   )
> # Add predicted mean and standard error
> dat_pred_mgcv <- bind_cols(
+   dat_pred_mgcv,
+   predict(
+     object = m_fit_mgcv,
+     newdata = dat_pred_mgcv,
+     se.fit = TRUE,
+     type = "response",
+     unconditional = TRUE
+     ) |>
+     as_tibble()
+   ) |>
+   # Add 95% confidence limits
+   mutate(
+     lcl = fit - qnorm(0.975) * se.fit,
+     ucl = fit + qnorm(0.975) * se.fit
+   )
```

```
> # Create plot
> ggplot(dat_pred_mgcv, aes(x = WEEK, y = fit, color = ARMCD)) +
+   geom_ribbon(aes(
+     ymin = lcl,
+     ymax = ucl,
+     fill = ARMCD),
+     linetype = 0,
+     alpha = 0.15
+   ) +
+   geom_line(linewidth = 1.2) +
+   scale_x_continuous(
+     breaks = seq(min(dat_pred_mgcv$WEEK), max(dat_pred_mgcv$WEEK), by = 2)
+   ) +
+   scale_color_manual(values = c("darkgoldenrod", "coral2")) +
+   scale_fill_manual(values = c("darkgoldenrod", "coral2"), guide = "none") +
+   labs(x = "Time (weeks)",
+        y = "Predicted change from baseline and 95% CI",
```

```
+        color = "Treatment",
+        title = "Predictions from mgcv::gam() fit") +
+   theme_bw()
```

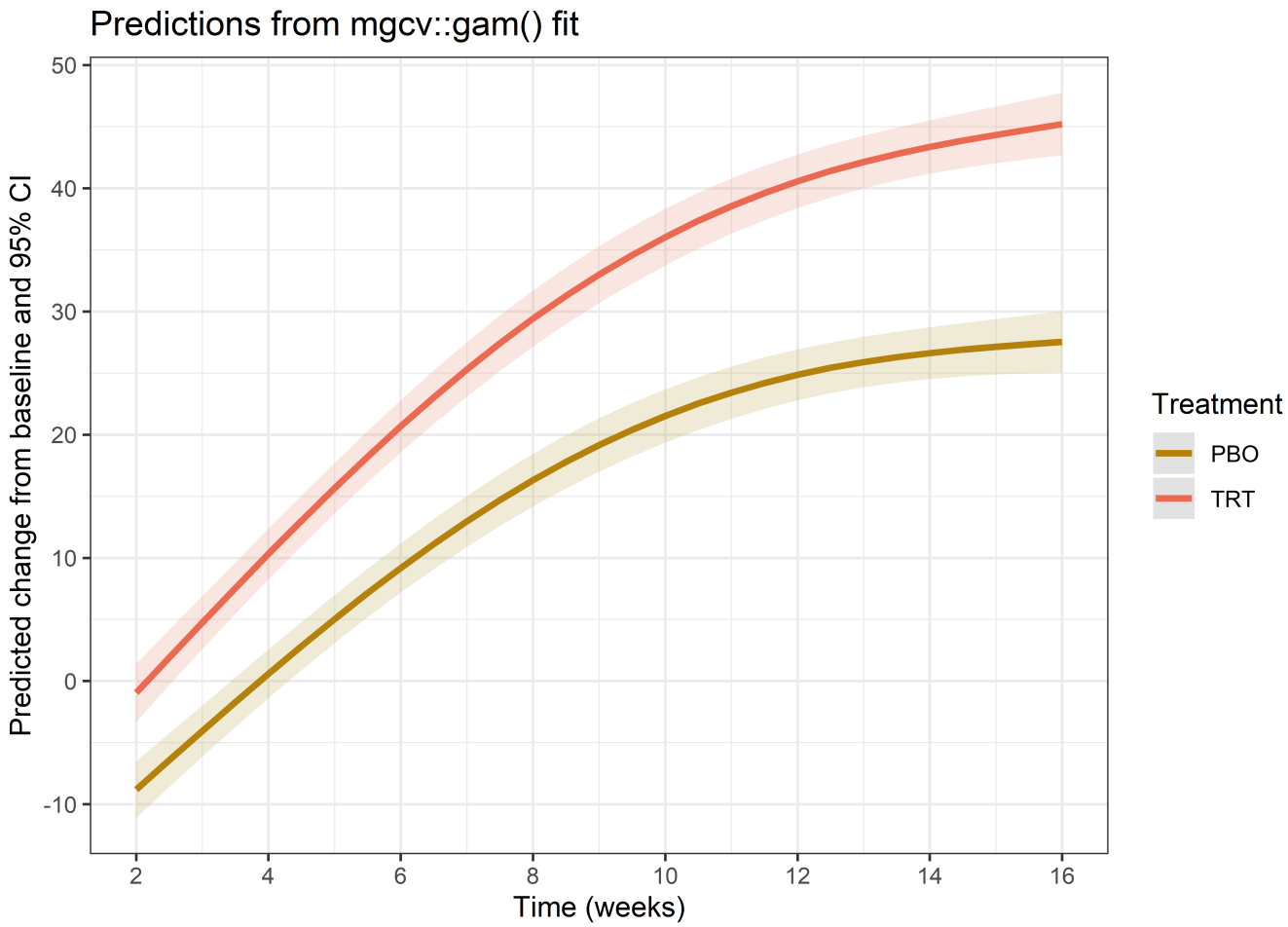

Figure: Marginal means from `{mgcv}` fit - with markedly narrower intervals.

Contrasts

Contrasts at custom time points are computed (for convenience here the `{emmeans}` package is used):

```
> contrasts_mgcv <- emmeans(
+   object = m_fit_mgcv,
+   specs = ~ ARMCD | WEEK,
+   at = list(WEEK = custom_time_points)
+   ) |>
+   contrast(method = "pairwise") |>
+   as.data.frame() |>
+   select(-c("df", "t.ratio", "p.value")) |>
+   mutate(
+     lcl = estimate - qnorm(0.975) * SE,
+     ucl = estimate + qnorm(0.975) * SE
+   )
> contrasts_mgcv |> gt()
```

contrast	WEEK	estimate	SE	lcl	ucl
PBO - TRT	2	-7.861396	1.671540	-11.13755	-4.585238
PBO - TRT	4	-9.723898	1.446282	-12.55856	-6.889237
PBO - TRT	8	-13.110691	1.587791	-16.22270	-9.998678
PBO - TRT	12	-15.708818	1.527442	-18.70255	-12.715087
PBO - TRT	16	-17.661315	1.811411	-21.21162	-14.111015

Comparison

Collect data:

```
> comparison <- contrasts_brms |>
+   as_tibble() |>
+   select(
+     week = WEEK,
+     brms_est = estimate,
+     brms_lcl = lower.HPD,
+     brms_ucl = upper.HPD
+   ) |>
+   bind_cols(
+     select(contrasts_mgcv |> as_tibble(),
+            mgcv_est = estimate,
+            mgcv_lcl = lcl,
+            mgcv_ucl = ucl)
+   ) |>
+   mutate(
+     diff_est = brms_est - mgcv_est,
+     diff_lcl = brms_lcl - mgcv_lcl,
+     diff_ucl = brms_ucl - mgcv_ucl
+   )
```

Contrasts by week and estimation method:

```
> comparison |>
+   gt() |>
+   fmt_number(
+     columns = contains(c("_est", "_lcl", "_ucl")),
+     decimals = 2
+   ) |>
+   tab_spanner(
+     label = "brms",
+     columns = c(brms_est, brms_lcl, brms_ucl)
+   ) |>
+   tab_spanner(
```

```
+     label = "mgcv",
+     columns = c(mgcv_est, mgcv_lcl, mgcv_ucl)
+   ) |>
+   tab_spanner(
+     label = "Difference",
+     columns = c(diff_est, diff_lcl, diff_ucl)
+   ) |>
+   cols_label(
+     brms_est = "Est.",
+     brms_lcl = "LCL",
+     brms_ucl = "UCL",
+     mgcv_est = "Est.",
+     mgcv_lcl = "LCL",
+     mgcv_ucl = "UCL",
+     diff_est = "Est.",
+     diff_lcl = "LCL",
+     diff_ucl = "UCL"
+   )
```

	brms			mgcv			Difference		
week	Est.	LCL	UCL	Est.	LCL	UCL	Est.	LCL	UCL
2	−7.77	−10.93	−4.36	−7.86	−11.14	−4.59	0.09	0.20	0.23
4	−9.68	−12.46	−6.80	−9.72	−12.56	−6.89	0.04	0.10	0.09
8	−13.09	−16.13	−9.93	−13.11	−16.22	−10.00	0.03	0.09	0.07
12	−15.69	−18.82	−12.80	−15.71	−18.70	−12.72	0.02	−0.12	−0.08
16	−17.62	−21.04	−14.06	−17.66	−21.21	−14.11	0.04	0.17	0.05

Discussion points

- Use samples from posterior predictive distribution directly from `{brms}` fit or use `{emmeans}` package
- Residual error in predictions from `brms` model
- Synthesis for *multiple* historical studies
  - May be treated as mixture of multivariate normals
  - For Bayesian multivariate meta-analysis: NICE DSU TSD 20