August model results

Justin Pomeranz

2025-05-28

Overview

This document summarizes the model fitting and posterior for mean stream temperatures in August for the Teton Alpine Stream Research (TASR) project.

Setup

- Load libraries

```
library(tidyverse)
library(brms)
library(tidybayes)
```

- Load data
- Data has already been modified in the following ways:
  - Only data for the month of August is included
  - `year_s` has been centered at 0 and scaled by the `sd(year)`
  - `temp_1` has been scaled to be from 0-1 as: $ temp_1 = temp_i / max(temp)$

```
fit_data_orig <- readRDS(
  here::here("Temperature/brms_models/all_sites/august/fit_data.rds"))
fit_data_year_summary <- readRDS(
  here::here("Temperature/brms_models/all_sites/August/fit_data_year_summary.rds"))
```

- Load model

```
brm1 <- readRDS(here::here("Temperature/brms_models/all_sites/august/fit_rand_slopes_hurdle_aug_rand_si
```

Model details

- 4 chains with 2000 iterations
  - half discarded as warmup
  - 4000 posterior iterations

- The model has the following formula

```
brm1$formula
```

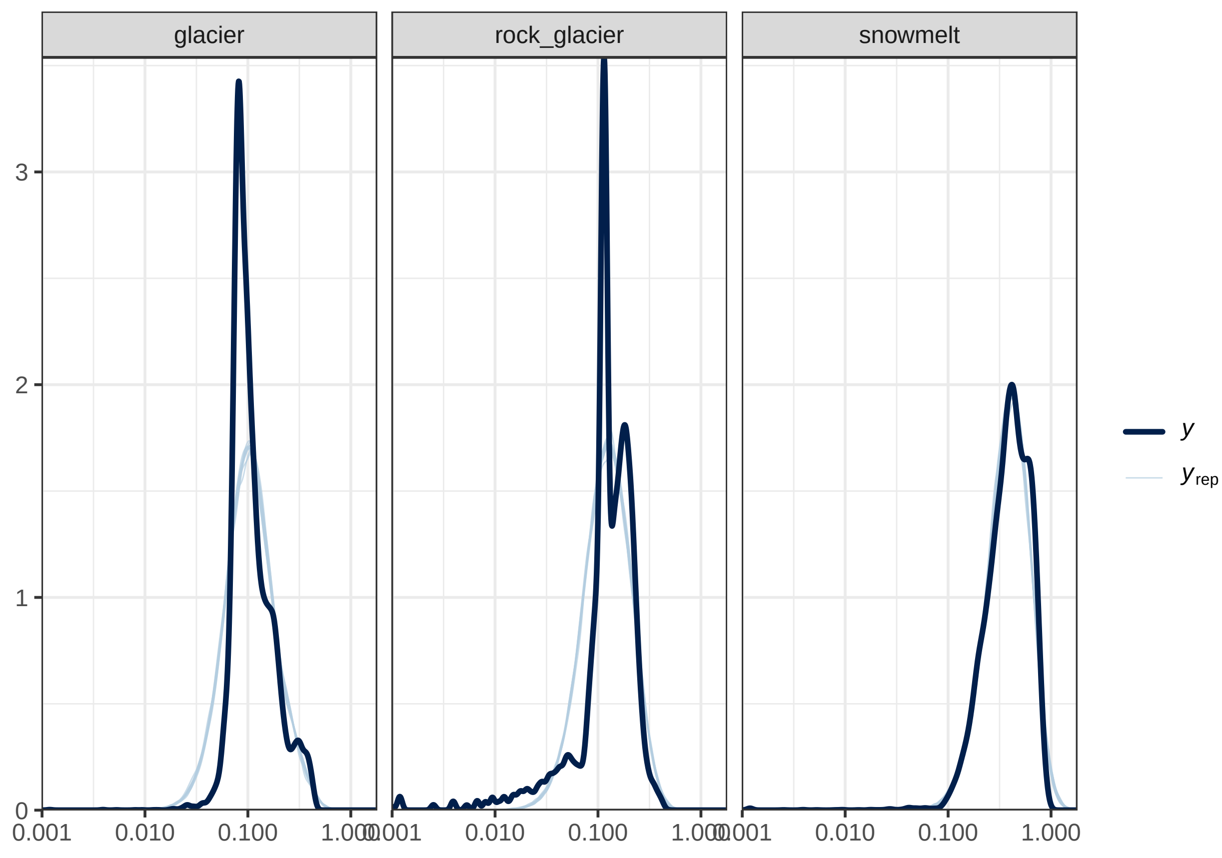
```
## temp_1 ~ year_s + source + year_s:source + (1 + year_s | site)
```

Where `source` is a categorical factor with the following levels: `rock_glacier`, `glacier`, and `snowfield`.

Posterior Prediction

- Density overlay for sources

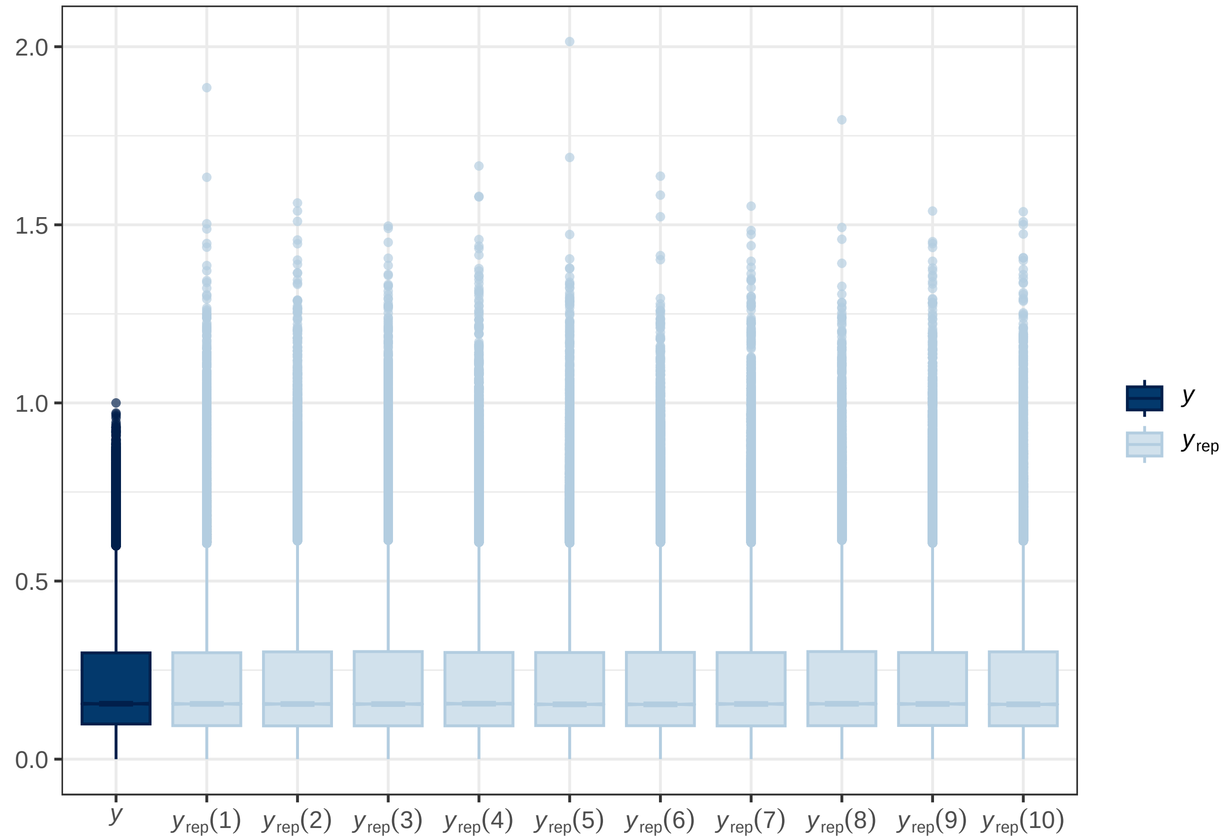
```
pp_check(brm1,
         type = "dens_overlay_grouped",
         group = "source") +
  scale_x_log10() +
  theme_bw()
```



- boxplot of posterior

- Note that this model correctly captures the lower bound of 0, and does not predict negative numbers.

```r
pp_check(brm1,
         type = "boxplot") +
  theme_bw()
```



Bayesian R^2

```r
bayes_R2(brm1)
```

```
##     Estimate  Est.Error      Q2.5     Q97.5
## R2 0.6364416 0.00258655 0.6313652 0.6415157
```

- This model explains 63.6% (95% credible interval 63.1 to 64.2) of the variation in August temperatures.

Model Posteriors

- Now we will explore the model posteriors to understand the effects of stream source on August temperatures

Source effect

Data for prediction

- First, calculate the average and SD of year from the original data, and the maximum temperature
- These are used to back calculate model posteriors to original scales as appropriate.

```
mean_year <- fit_data_year_summary$mean_year
sd_year <- fit_data_year_summary$sd_year
max_temp <-  max(fit_data_orig$temp_c)
```

- Make a data frame with the original:
    - source
    - year_s (Z_score of year)
- We will then make a date of August 15th for each year for plotting purposes
- This data will be used to add posterior draws based on the model in the next step

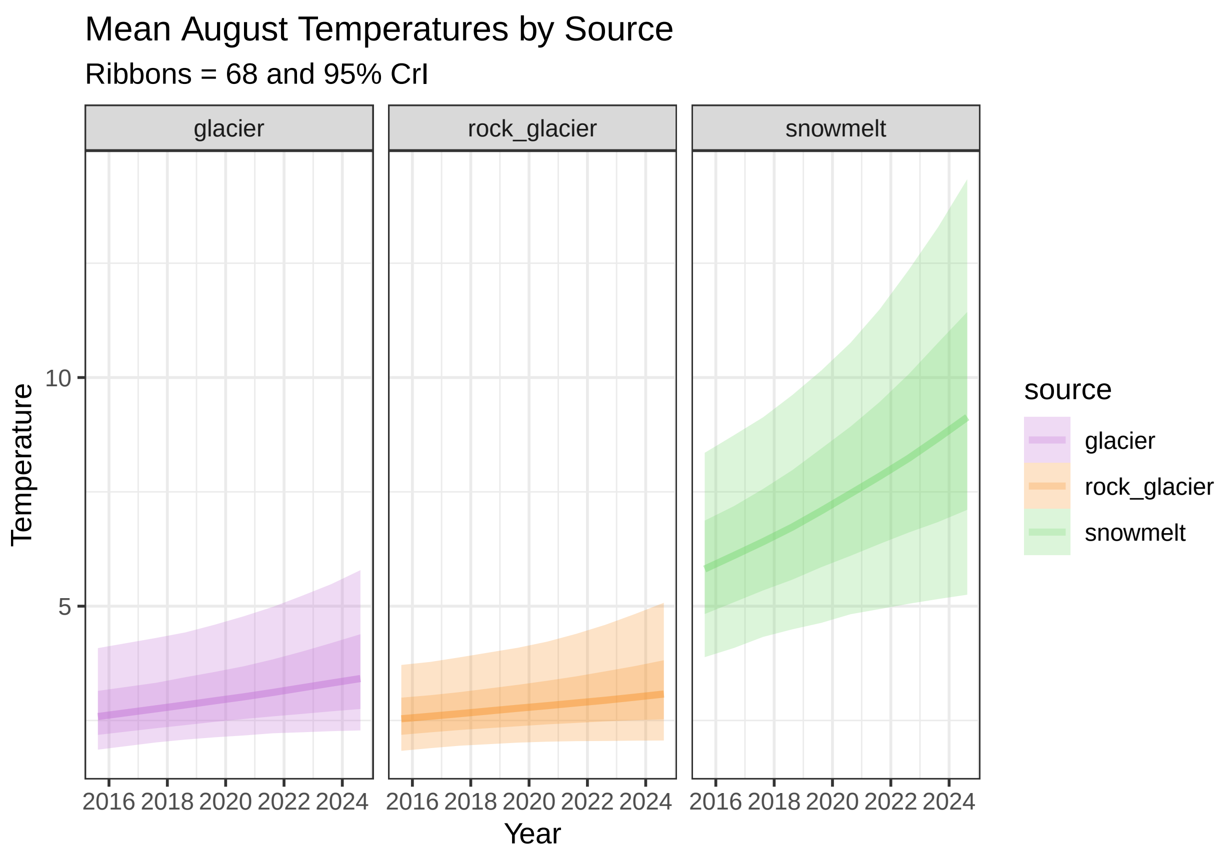
```
source_data <- brm1$data |>
  select(source, year_s) |>
  distinct() |> # all combinations of source and year_s from empirical data
  mutate(year = (year_s*sd_year)+mean_year, # back calculate year Z-score to original year
         month = 8, # manually set month number
         day = 15, # and day number
         date = make_date(year, month, day)) |> # set this as a `date` variable type for plotting
  select(source, year_s, date) # only keep variables we need later

head(source_data) # see first few rows as an example
```

```
##         source     year_s       date
## 1 rock_glacier -0.7290059 2019-08-15
## 2 rock_glacier -0.2928012 2020-08-15
## 3 rock_glacier  0.1434035 2021-08-15
## 4 rock_glacier  0.5796081 2022-08-15
## 5 rock_glacier  1.0158128 2023-08-15
## 6 rock_glacier  1.4520175 2024-08-15
```

- Visualize the effect of year and source on August stream temperatures

```
# add posterior draws
source_time_plot <- add_epred_draws(
  newdata = source_data, # new data made above to add expected (mean) temperature values based on the p
  brm1, # the model fit
  re_formula = NA) |> # ignores random effects of site and just shows the overall fixed main and intera
  mutate(.epred = (.epred*max_temp)) |> # converts back to original temperature scale
  ggplot(aes(x = date,
             y = .epred,
             fill = source,
             color = source)) +
  stat_lineribbon(aes(y = .epred),
                  .width = c(0.95, 0.68),
                  alpha = 1/4) +
  theme_bw() +
```

```r
  labs(title = "Mean August Temperatures by Source",
       subtitle = "Ribbons = 68 and 95% CrI",
       x = "Year",
       y = "Temperature") +
  facet_wrap(~source) +
  scale_fill_manual(values = c("#C26ED6",
                               "#F89225",
                               "#76D96F")) +
  scale_color_manual(values = c("#C26ED6",
                                "#F89225",
                                "#76D96F"))
source_time_plot
```



Magnitude of change per year

- Calculate the change from year to year
- This is a Gamma model, so the interpretation of the "slope" (β_{year}) coefficient is different from a standard generalized linear model
  - Instead of saying "Temperatures increase by"x" each year", we say "Temperatures chnage each year by"x percentage".
  - This is why there is a "curve" in the plots above.

- In order to estimate the percent change, I will create a dataframe with only 2 years from the middle of the data.
  - The expected values will be added with `add_epred_draws()`
  - The absolute difference will be calculated by subtracting the value in the last year from the first year
  - The percent change will then be calculated by dividing the difference by the value in the first year times 100
  - The distribution of the percent changes will be reported.
- Using other year-pairs from the data does not change the results (SI).

```
# 2019 and 2020
y_19_20 <- data.frame(
  year_real = c(2019, 2020)) |>
  mutate(year_s = (year_real - mean_year)/sd_year) |>
  expand_grid(source = unique(brm1$data$source)) |>
  add_epred_draws(brm1,
                  re_formula = NA,
                  allow_new_levels = TRUE) |>
  ungroup()|>
  mutate(.epred = .epred * max_temp) |>
  select(-.row, -.chain, -.iteration, -year_s) |>
  pivot_wider(names_from = "year_real",
              values_from = ".epred") |>
  mutate(slope_new = `2020` - `2019`) |>
  # slope_new = change in actual temperature right scale
  mutate(prop = slope_new / `2019` * 100)

y_19_20 |>
  group_by(source) |>
  mean_qi(prop)
```

```
## # A tibble: 3 x 7
##   source        prop .lower .upper .width .point .interval
##   <chr>        <dbl>  <dbl>  <dbl>  <dbl> <chr>  <chr>
## 1 glacier       3.15 -1.18    7.96   0.95 mean   qi
## 2 rock_glacier  2.21 -1.67    6.31   0.95 mean   qi
## 3 snowmelt      5.09  0.185   9.68   0.95 mean   qi
```

- `snowfield` streams increase by an average of 5.09% (CrI 0.18, 9.68) per year.
  - 95% CrI is entirely positive (see half eye plots below)
- `glacier` streams increase by an average of 3.15% (CrI -1.18, 8) per year.
  - 95% CrI is mostly positive, but there are also negative estimates
- `rock_glacier` streams increase by an average of 2.21% (CrI -1.67, 6.31) per year.
  - 95% CrI is entirely positive (see halfeye plots below)

Probability of positive slopes?

- The mean magnitude of the percent change form year to year is reported above

- But what is the probability that these slopes are positive?
- Calculate the proportion of slopes which are positive

- make a halfeye plot with the positive iterations highlighted

Pr(positive)

```
pr_pos <- y_19_20 |>
  group_by(source) |>
  mutate(pos = slope_new >0) |>
  summarize(pr_pos = mean(pos))

pr_pos
```

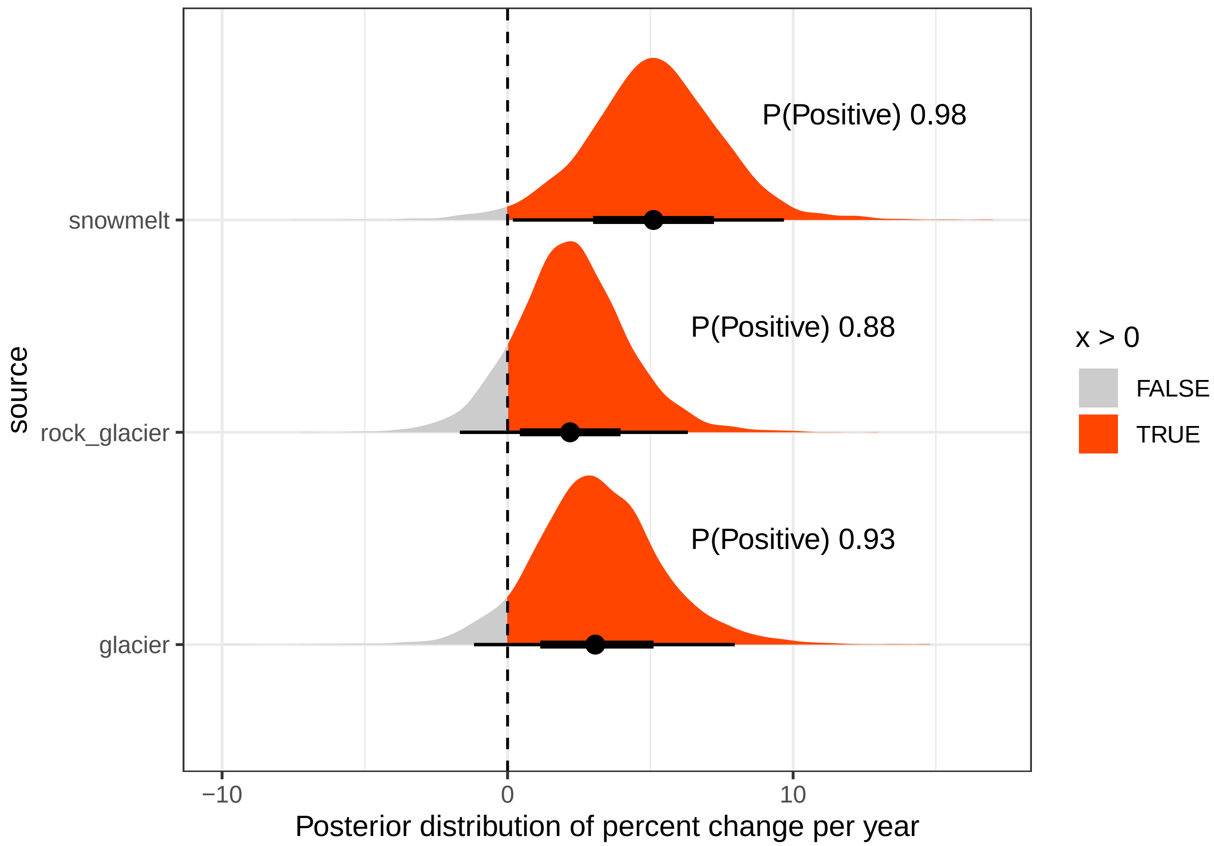
```
## # A tibble: 3 x 2
##   source       pr_pos
##   <chr>         <dbl>
## 1 glacier       0.930
## 2 rock_glacier  0.879
## 3 snowmelt      0.978
```

- `snowfield` streams have a 97.8% chance of having increasing mean August temperatures
- `glacier` streams have a 93% chance of having increasing mean August temperatures
- `rock_glacier` streams have an 88% chance of having increasing mean August temperatures

Halfeye plot

```
y_19_20 |>
  ggplot(
    aes(y = source,
        x = prop,
        fill = after_stat(x >0))) +
  stat_halfeye() +
  geom_vline(xintercept = 0,
             linetype = "dashed") +
  scale_fill_manual(values = c("gray80", "orangered1")) +
  theme_bw() +
  labs(x = "Posterior distribution of percent change per year") +
  annotate("text", # rock_glacier
           x = 10,
           y = 2.5,
```

```
           label = paste("P(Positive)",
                         round(pr_pos[2,2],2),
                         sep = " ")) +
  annotate("text", # Glacier
           x = 10,
           y = 1.5,
           label = paste("P(Positive)",
                         round(pr_pos[1,2],2),
                         sep = " ")) +
  annotate("text", # snow
           x = 12.5,
           y = 3.5,
           label = paste("P(Positive)",
                         round(pr_pos[3,2],2),
                         sep = " ")) +
  NULL
```



Source-specific changes across period of record

Here, we use the model posteriors to calculate the the average change in degrees C from 2015 to 2024 for each source category.

- First, make source-level data

- add expected posterior draws
- summarize and plot results

```
source_temps_data <- brm1$data |>
  select(source) |>
  distinct() |>
  expand_grid(year_real = c(2015, 2024)) |>
  mutate(year_s = (year_real - mean_year)/sd_year)

source_temps <- source_temps_data |>
  add_epred_draws(
  brm1,
  re_formula = NA) |> # does not include random effects
  mutate(.temp = (.epred*max_temp)) |>
  ungroup()

head(source_temps)
```

```
## # A tibble: 6 x 9
##   source      year_real year_s  .row .chain .iteration .draw
##   <chr>           <dbl>  <dbl> <int>  <int>      <int> <int>
## 1 rock_glaci~      2015  -2.47     1     NA         NA     1
## 2 rock_glaci~      2015  -2.47     1     NA         NA     2
## 3 rock_glaci~      2015  -2.47     1     NA         NA     3
## 4 rock_glaci~      2015  -2.47     1     NA         NA     4
## 5 rock_glaci~      2015  -2.47     1     NA         NA     5
## 6 rock_glaci~      2015  -2.47     1     NA         NA     6
## # i 2 more variables: .epred <dbl>, .temp <dbl>
```

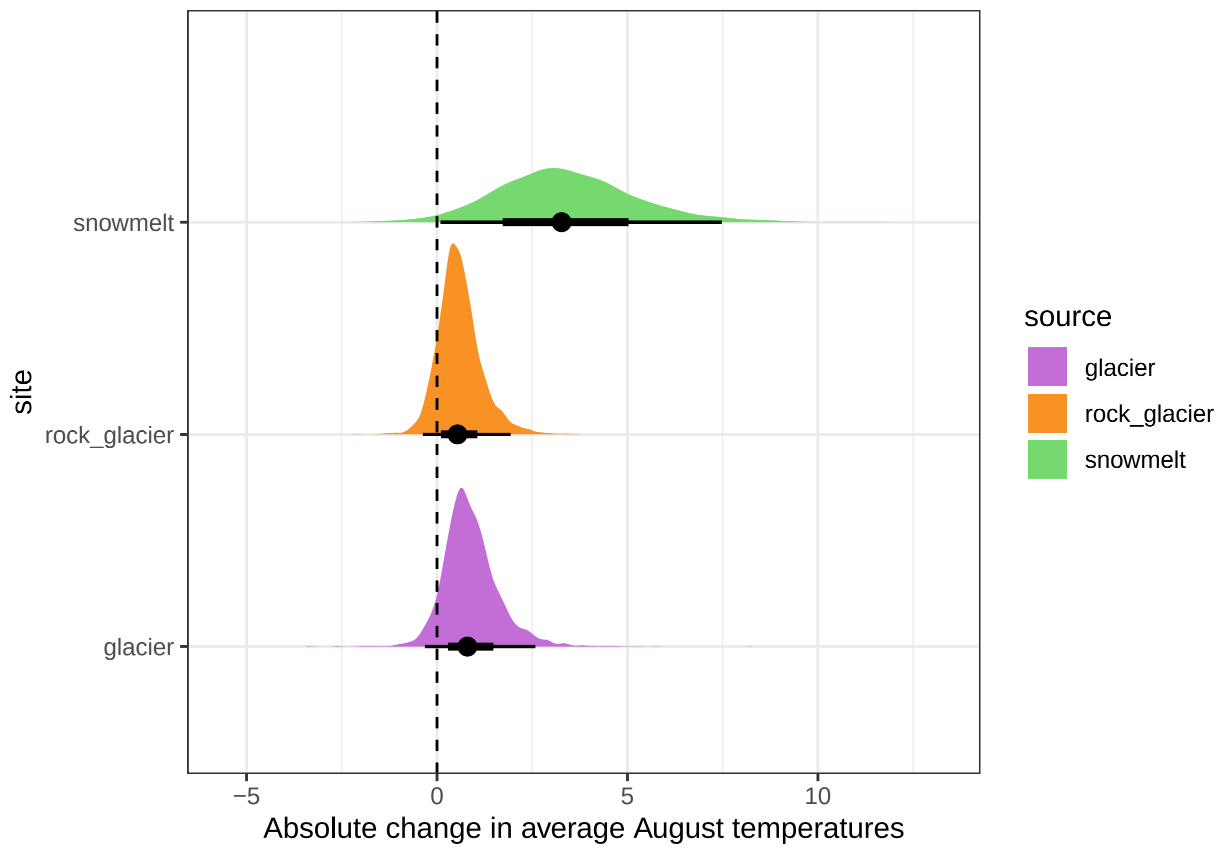
- for each draw:
- Subtract the beginning temperature from the ending temperature
- Plot distributions of absolute change in temperatures per source

```
source_temp_change <- source_temps |>
  select(year_s, .temp, .draw, source) |>
  group_by(source, .draw) |>
  mutate(year_cat = case_when(
    year_s == min(year_s) ~ "start",
         .default = "end")) |>
  select(-year_s) |>
  pivot_wider(id_cols = c(.draw, source),
              names_from = year_cat,
              values_from = .temp) |>
  mutate(delta = end - start)

head(source_temp_change)
```

```
## # A tibble: 6 x 5
## # Groups:   source, .draw [6]
```

```
##   .draw source       start   end  delta
##   <int> <chr>        <dbl> <dbl>  <dbl>
## 1     1 rock_glacier  2.63  2.46 -0.161
## 2     2 rock_glacier  2.90  2.63 -0.268
## 3     3 rock_glacier  3.05  4.98  1.93
## 4     4 rock_glacier  2.91  4.00  1.09
## 5     5 rock_glacier  2.11  2.37  0.259
## 6     6 rock_glacier  2.97  3.57  0.600
```

```r
source_temp_change_plot <- source_temp_change |>
  ggplot(aes(y = source,
             x = delta,
             fill = source)) +
  stat_halfeye()+
  scale_fill_manual(values = c("#C26ED6",
                               "#F89225",
                               "#76D96F")) +
  theme_bw() +
  geom_vline(xintercept = 0,
             linetype = "dashed") +
  labs(y = "site",
       x = "Absolute change in average August temperatures")

source_temp_change_plot
```



- Summary of temp change by source

```r
source_temp_change |>
  group_by(source) |>
  mean_qi(delta)
```

```
## # A tibble: 3 x 7
##   source       delta  .lower .upper .width .point .interval
##   <chr>        <dbl>   <dbl>  <dbl>  <dbl> <chr>  <chr>
## 1 glacier      0.889 -0.318    2.58   0.95 mean   qi
## 2 rock_glacier 0.592 -0.372    1.93   0.95 mean   qi
## 3 snowmelt     3.41   0.0905   7.48   0.95 mean   qi
```

- here `delta` is the absolute change in degrees C from 2015 to 2024
- calculated as temp in 2024 - temp in 2015
  - so positive values are *increases* in temperature
  - Negative values are decreases in temperature

Site-specific predictions

- Here, we have the results for each site
- First, make site-level data
- add expected posterior draws
- summarize and plot results

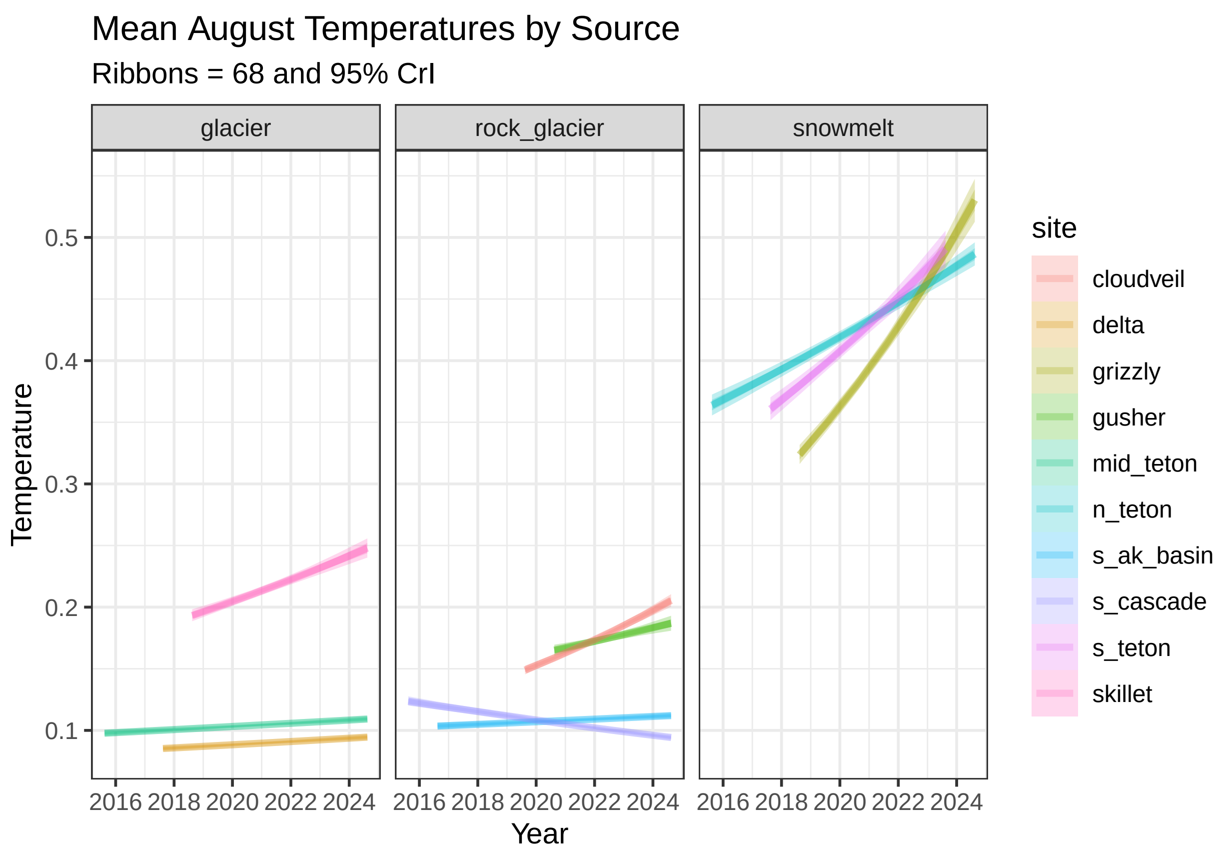
```r
site_data <- brm1$data |>
  select(source, year_s, site) |>
  distinct() |>
  group_by(site) |>
  mutate(year = (year_s*sd_year)+mean_year,
         month = 8,
         day = 15,
         date = make_date(year, month, day))

site_temps <- add_epred_draws(
  newdata = site_data,
  brm1,
  re_formula = NULL) |> # includes random effects
  mutate(.temp = (.epred*max_temp))

head(site_temps)
```

```
## # A tibble: 6 x 13
## # Groups:   source, year_s, site, year, month, day, date,
## #   .row [1]
##   source     year_s site   year month   day date        .row
##   <chr>       <dbl> <chr> <dbl> <dbl> <dbl> <date>     <int>
```

```
## 1 rock_glac~ -0.729 clou~  2019     8    15 2019-08-15     1
## 2 rock_glac~ -0.729 clou~  2019     8    15 2019-08-15     1
## 3 rock_glac~ -0.729 clou~  2019     8    15 2019-08-15     1
## 4 rock_glac~ -0.729 clou~  2019     8    15 2019-08-15     1
## 5 rock_glac~ -0.729 clou~  2019     8    15 2019-08-15     1
## 6 rock_glac~ -0.729 clou~  2019     8    15 2019-08-15     1
## # i 5 more variables: .chain <int>, .iteration <int>,
## #   .draw <int>, .epred <dbl>, .temp <dbl>
```

Plot through time

```
ggplot(site_temps,
       aes(x = date,
           y = .epred,
           fill = site,
           color = site)) +
  stat_lineribbon(aes(y = .epred),
                  .width = c(0.95, 0.68),
                  alpha = 1/4) +
  theme_bw() +
  labs(title = "Mean August Temperatures by Source",
       subtitle = "Ribbons = 68 and 95% CrI",
       x = "Year",
       y = "Temperature") +
  facet_wrap(~source)
```



- Notice that nearly all streams increase in temperature through time
- However, `s_cascade` actually *declines* in temperature through time
  - This may be of interest to discuss in the paper?
  - perhaps this rock glacier is melting faster, hence putting more cold water into the stream?
  - Or maybe something else?

Site-specific percent change

```
s_19_20 <- brm1$data |>
  select(source, site) |>
  distinct() |>
  expand_grid(year_real = c(2019, 2020)) |>
  mutate(year_s = (year_real - mean_year)/sd_year) |>
  add_epred_draws(brm1,
                  re_formula = NULL) |>
  ungroup()|>
  mutate(.epred = .epred * max_temp) |>
  select(-.row, -.chain, -.iteration, -year_s) |>
  pivot_wider(names_from = "year_real",
              values_from = ".epred") |>
```

```
  mutate(slope_new = `2020` - `2019`) |>
  # slope_new = change in actual temperature right scale
  mutate(prop = slope_new / `2019` * 100)

s_19_20 |>
  group_by(site, source) |>
  mean_qi(prop) |>
  arrange(source, site, prop)
```

```
## # A tibble: 10 x 8
##    site  source   prop .lower .upper .width .point .interval
##    <chr> <chr>   <dbl>  <dbl>  <dbl>  <dbl> <chr>  <chr>
##  1 delta glaci~  1.48   0.955   1.99   0.95 mean   qi
##  2 mid_~ glaci~  1.24   0.833   1.65   0.95 mean   qi
##  3 skil~ glaci~  4.24   3.42    5.05   0.95 mean   qi
##  4 clou~ rock_~  6.63   5.78    7.49   0.95 mean   qi
##  5 gush~ rock_~  3.16   1.85    4.48   0.95 mean   qi
##  6 s_ak~ rock_~  0.991  0.529   1.46   0.95 mean   qi
##  7 s_ca~ rock_~ -2.97  -3.43   -2.54   0.95 mean   qi
##  8 griz~ snowm~  8.58   7.70    9.45   0.95 mean   qi
##  9 n_te~ snowm~  3.28   2.87    3.71   0.95 mean   qi
## 10 s_te~ snowm~  5.25   4.40    6.09   0.95 mean   qi
```

Site-specific Pr(+)

```
s_19_20 |>
  group_by(site) |>
  mutate(pos = slope_new >0) |>
  summarize(pr_pos = mean(pos))
```

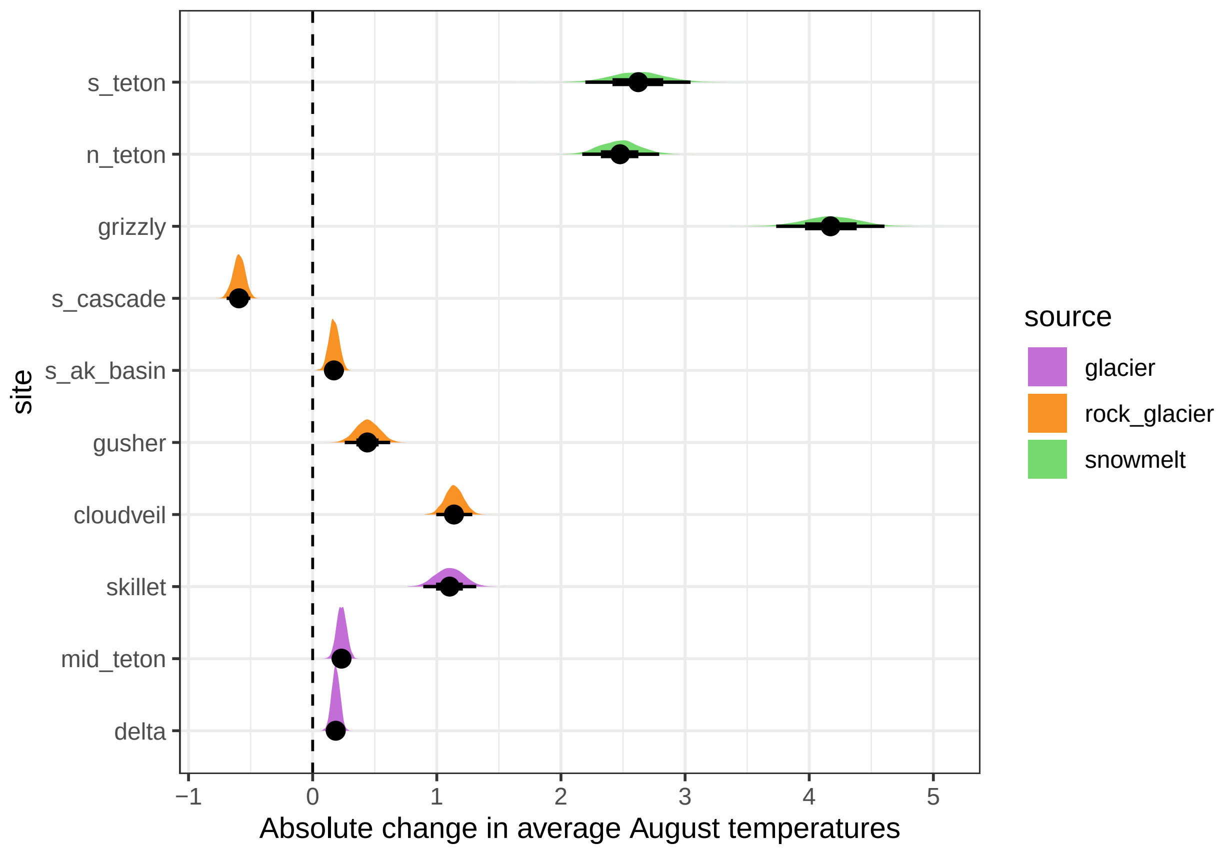
```
## # A tibble: 10 x 2
##    site       pr_pos
##    <chr>       <dbl>
##  1 cloudveil       1
##  2 delta           1
##  3 grizzly         1
##  4 gusher          1
##  5 mid_teton       1
##  6 n_teton         1
##  7 s_ak_basin      1
##  8 s_cascade       0
##  9 s_teton         1
## 10 skillet         1
```

site-specific change in temperatures

- mean temperature change
- first year with empirical data

- last year with empirical data (mostly 2024)

```
site_temp_change_plot <- brm1$data |>
  select(source, year_s, site) |>
  distinct() |>
  group_by(site) |>
  filter(year_s == min(year_s) |
           year_s == max(year_s)) |>
  mutate(year = (year_s*sd_year)+mean_year) |>
  add_epred_draws(
  brm1,
  re_formula = NULL) |> # includes random effects
  mutate(.temp = (.epred*max_temp)) |>
  ungroup() |>
  select(site, year, .temp, .draw, source) |>
  group_by(source, site, .draw) |>
  mutate(year_cat = case_when(
    year == min(year) ~ "start",
       .default = "end")) |>
  select(-year) |>
  pivot_wider(id_cols = c(site, .draw, source),
              names_from = year_cat,
              values_from = .temp) |>
  mutate(delta = end - start,
         group = cur_group_id()) |>
  ggplot(aes(y = fct_reorder(site, group),
             x = delta,
             fill = source)) +
  stat_halfeye()+
  scale_fill_manual(values = c("#C26ED6",
                               "#F89225",
                               "#76D96F")) +
  theme_bw() +
  geom_vline(xintercept = 0,
             linetype = "dashed") +
  labs(y = "site",
       x = "Absolute change in average August temperatures")
site_temp_change_plot
```



- range of years for each site

```
site_temps |>
  ungroup() |>
  select(site, year) |>
  group_by(site) |>
  filter(year == min(year) |
           year == max(year)) |>
  distinct(site, year) |>
  mutate(year_cat = case_when(
    year == min(year) ~ "start",
         .default = "end")) |>
  pivot_wider(id_cols = c(site),
              names_from = year_cat,
              values_from = year) |>
  mutate(year_range = end - start) |>
  arrange(start, end)
```

```
## # A tibble: 10 x 4
## # Groups:   site [10]
##    site        start   end year_range
##    <chr>       <dbl> <dbl>      <dbl>
##  1 mid_teton    2015  2024          9
##  2 n_teton      2015  2024          9
##  3 s_cascade    2015  2024          9
```

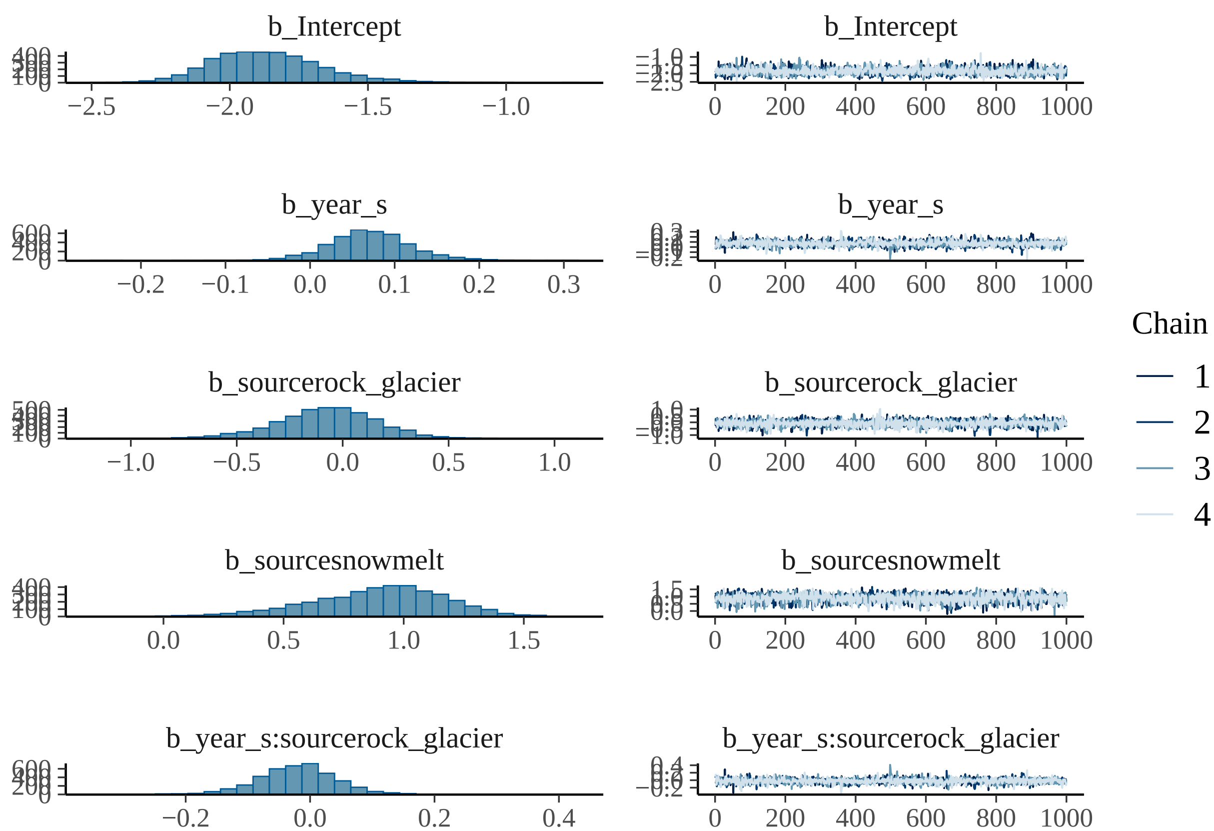
```
## 4 s_ak_basin 2016 2024 8
## 5 s_teton 2017 2023 6
## 6 delta 2017 2024 7
## 7 grizzly 2018 2024 6
## 8 skillet 2018 2024 6
## 9 cloudveil 2019 2024 5
## 10 gusher 2020 2024 4
```

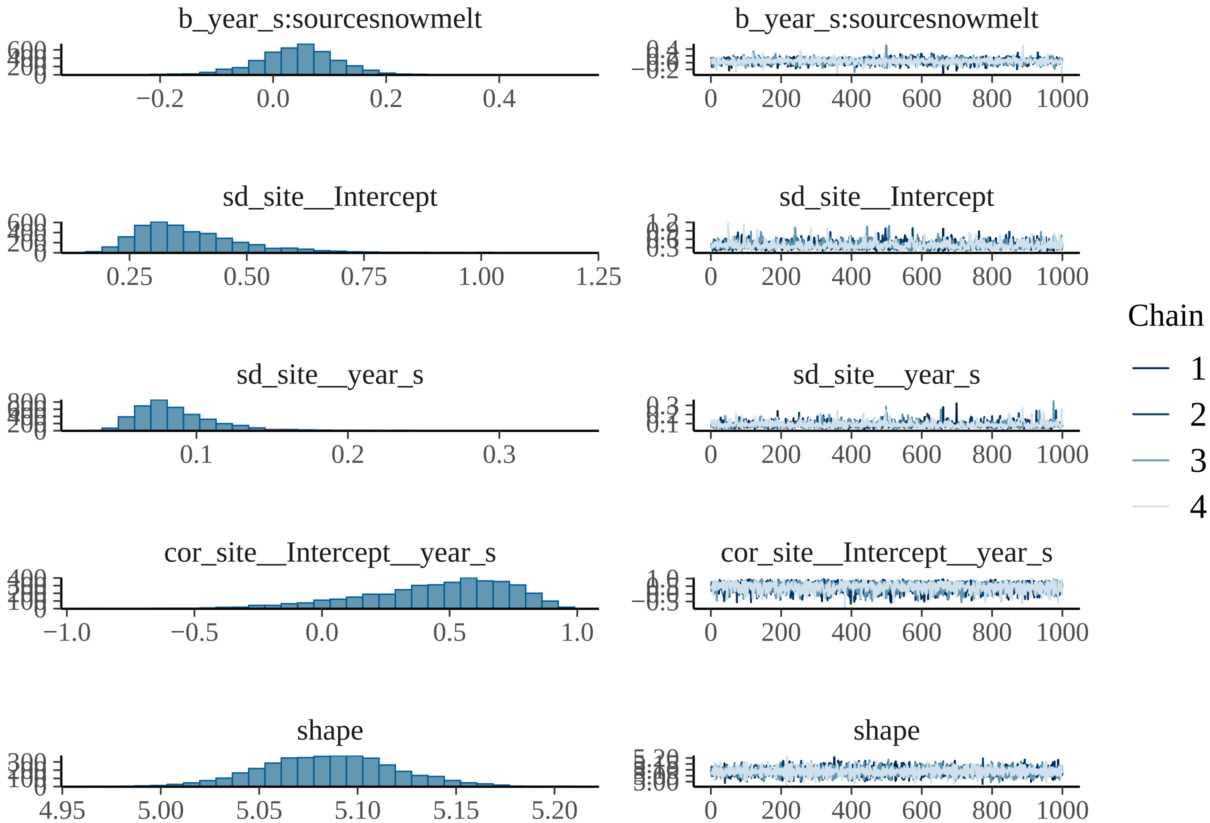
SI

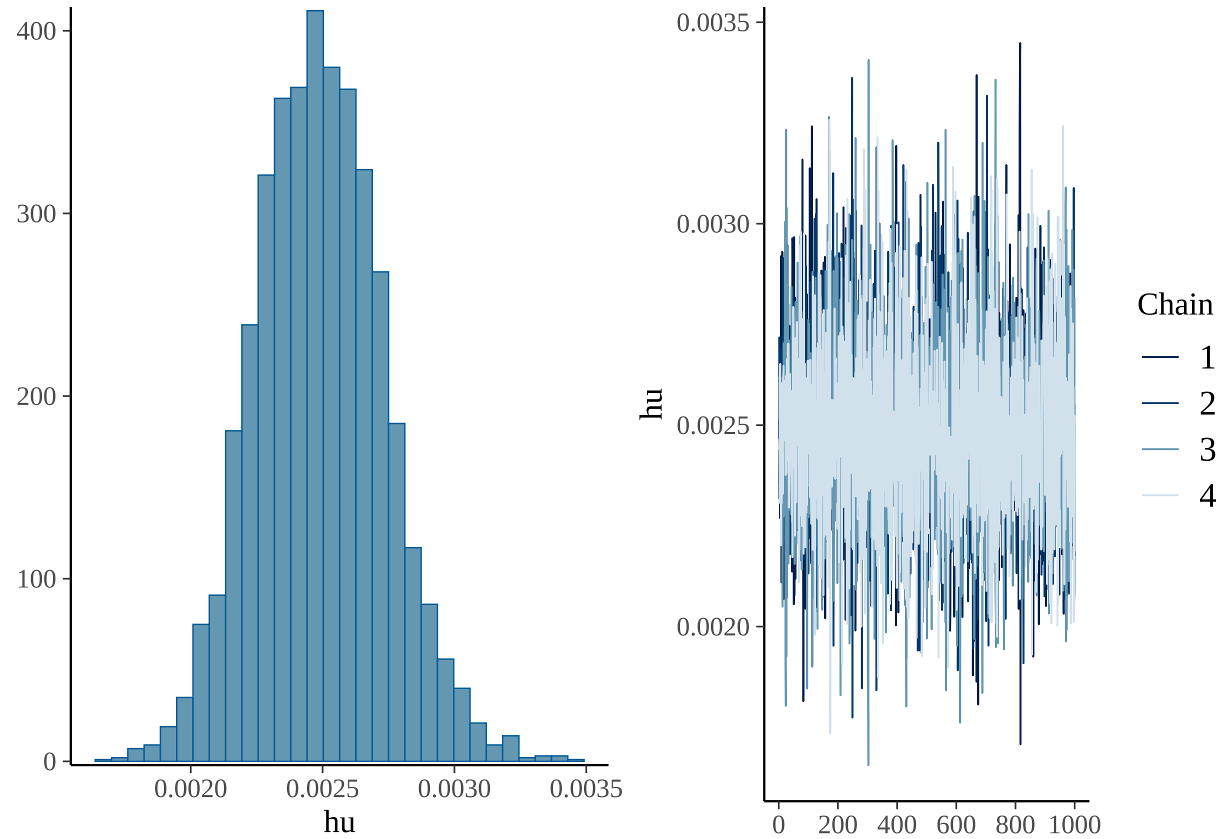
Model Fit

Model MCMC Chains and posterior of parameters

```r
plot(brm1, ask = FALSE)
```




b_year_s:sourcesnowmelt
sd_site__Intercept
sd_site__year_s
cor_site__Intercept__year_s
shape
Chain
1
2
3
4



All Rhats < 1.1

```
summary(brm1)
```

```
##  Family: hurdle_gamma
##   Links: mu = log; shape = identity; hu = identity
## Formula: temp_1 ~ year_s + source + year_s:source + (1 + year_s | site)
##    Data: mod_dat (Number of observations: 43371)
##   Draws: 4 chains, each with iter = 2000; warmup = 1000; thin = 1;
##          total post-warmup draws = 4000
##
## Multilevel Hyperparameters:
## ~site (Number of levels: 10)
##                       Estimate Est.Error l-95% CI u-95% CI
## sd(Intercept)             0.38      0.12     0.22     0.68
## sd(year_s)                0.09      0.03     0.05     0.16
## cor(Intercept,year_s)     0.45      0.29    -0.24     0.87
##                       Rhat Bulk_ESS Tail_ESS
## sd(Intercept)         1.00     1690     1989
## sd(year_s)            1.00     1654     2503
## cor(Intercept,year_s) 1.00     2697     2455
##
## Regression Coefficients:
##                           Estimate Est.Error l-95% CI
```

```
## Intercept                      -1.86      0.20    -2.21
## year_s                          0.07      0.05    -0.03
## sourcerock_glacier             -0.08      0.24    -0.57
## sourcesnowmelt                  0.89      0.28     0.28
## year_s:sourcerock_glacier      -0.02      0.06    -0.15
## year_s:sourcesnowmelt           0.04      0.07    -0.11
##                             u-95% CI Rhat Bulk_ESS Tail_ESS
## Intercept                      -1.41 1.00     2255     2217
## year_s                          0.18 1.00     2617     2375
## sourcerock_glacier              0.36 1.00     2650     2154
## sourcesnowmelt                  1.37 1.00     2201     2653
## year_s:sourcerock_glacier       0.10 1.00     2698     2465
## year_s:sourcesnowmelt           0.18 1.00     2464     2098
##
## Further Distributional Parameters:
##       Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS
## shape     5.09      0.03     5.02     5.15 1.00     6936
## hu        0.00      0.00     0.00     0.00 1.00     7675
##       Tail_ESS
## shape     2669
## hu        2703
##
## 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).
```

Percent change

Comparison between other year pairs

```r
data.frame(
  year_real = c(2015, 2016)) |>
  mutate(year_s = (year_real - mean_year)/sd_year) |>
  expand_grid(source = unique(brm1$data$source)) |>
  add_epred_draws(brm1,
                  re_formula = NA,
                  allow_new_levels = TRUE) |>
  ungroup()|>
  mutate(.epred = .epred * max_temp) |>
  select(-.row, -.chain, -.iteration, -year_s) |>
  pivot_wider(names_from = "year_real",
            values_from = ".epred") |>
  mutate(slope_new = `2016` - `2015`) |>
  # slope_new = chnage in actual temperature right scale
  mutate(prop = slope_new / `2015` * 100) |>
  group_by(source) |>
  mean_qi(prop)
```

2015 to 2016

```
## # A tibble: 3 x 7
```

```
##   source         prop .lower .upper .width .point .interval
##   <chr>         <dbl>  <dbl>  <dbl>  <dbl> <chr>  <chr>
## 1 glacier        3.15 -1.18    7.96   0.95 mean   qi
## 2 rock_glacier   2.21 -1.67    6.31   0.95 mean   qi
## 3 snowmelt       5.09  0.185   9.68   0.95 mean   qi
```

```r
data.frame(
  year_real = c(2023, 2024)) |>
  mutate(year_s = (year_real - mean_year)/sd_year) |>
  expand_grid(source = unique(brm1$data$source)) |>
  add_epred_draws(brm1,
                  re_formula = NA,
                  allow_new_levels = TRUE) |>
  ungroup()|>
  mutate(.epred = .epred * max_temp) |>
  select(-.row, -.chain, -.iteration, -year_s) |>
  pivot_wider(names_from = "year_real",
              values_from = ".epred") |>
  mutate(slope_new = `2024` - `2023`) |>
  # slope_new = chnage in actual temperature right scale
  mutate(prop = slope_new / `2023` * 100) |>
  group_by(source) |>
  mean_qi(prop)
```

2023 to 2024

```
## # A tibble: 3 x 7
##   source         prop .lower .upper .width .point .interval
##   <chr>         <dbl>  <dbl>  <dbl>  <dbl> <chr>  <chr>
## 1 glacier        3.15 -1.18    7.96   0.95 mean   qi
## 2 rock_glacier   2.21 -1.67    6.31   0.95 mean   qi
## 3 snowmelt       5.09  0.185   9.68   0.95 mean   qi
```

- Exact same results as 2019-2020 reported above