

Activity 9: Statistical reasoning 1: intro to models

Feb. 4th, 2026, Calvin Munson

Piper Miller & Tavi Maes

Let's start by reading in the relevant packages

```
library(brms) # for statistics
```

Loading required package: Rcpp

Loading 'brms' package (version 2.23.0). Useful instructions can be found by typing `help('brms')`. A more detailed introduction to the package is available through `vignette('brms_overview')`.

Attaching package: 'brms'

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

ar

```
library(tidyverse)
```

```
-- Attaching core tidyverse packages ----- tidyverse 2.0.0 --
v dplyr     1.1.4     v readr     2.1.5
v forcats   1.0.0     v stringr   1.5.1
v ggplot2   3.5.1     v tibble    3.2.1
v lubridate 1.9.4     v tidyr    1.3.1
v purrr    1.0.2

-- Conflicts ----- tidyverse_conflicts() --
x dplyr::filter() masks stats::filter()
x dplyr::lag()    masks stats::lag()
i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become non-conflicting

library(ggeffects) # for the prediction plot
library(lterdatasampler) # for built-in datasets
```

1. Fiddler crabs

```
head(pie_crab)

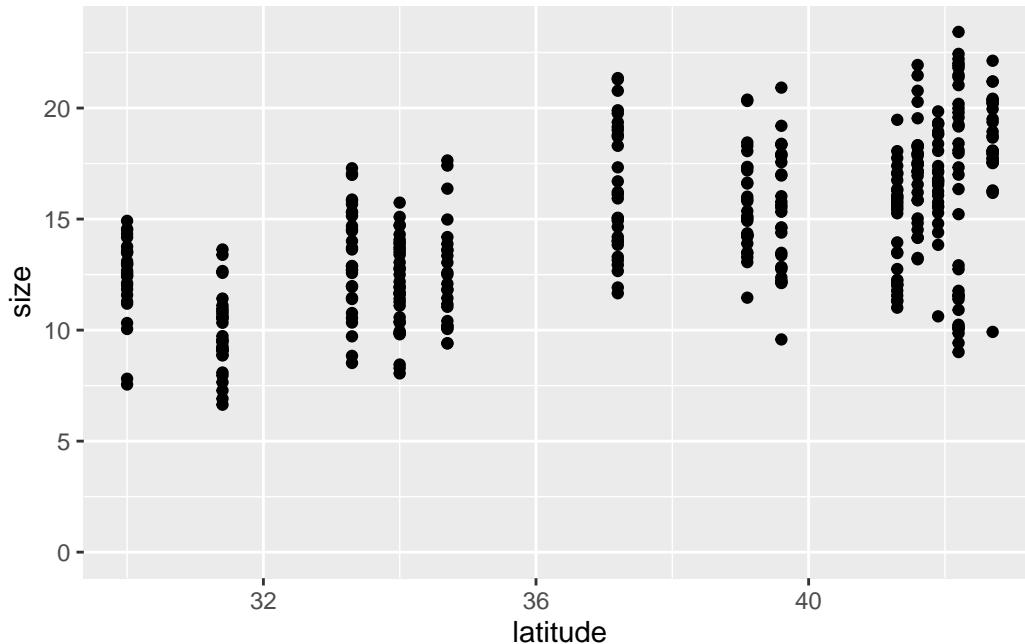
# A tibble: 6 x 9
  date      latitude site    size air_temp air_temp_sd water_temp water_temp_sd
  <date>     <dbl> <chr> <dbl>    <dbl>       <dbl>       <dbl>       <dbl>
1 2016-07-24     30 GTM    12.4     21.8      6.39      24.5      6.12
2 2016-07-24     30 GTM    14.2     21.8      6.39      24.5      6.12
3 2016-07-24     30 GTM    14.5     21.8      6.39      24.5      6.12
4 2016-07-24     30 GTM    12.9     21.8      6.39      24.5      6.12
5 2016-07-24     30 GTM    12.4     21.8      6.39      24.5      6.12
6 2016-07-24     30 GTM    13.0     21.8      6.39      24.5      6.12
# i 1 more variable: name <chr>
```

1.1 Plot data, pick the model

```

pie_crab %>%
  ggplot(aes(x = latitude, y = size)) +
  geom_point() +
  # Make the y-axis include 0
  ylim(0, NA)

```



Q1.1 Interpret the graph

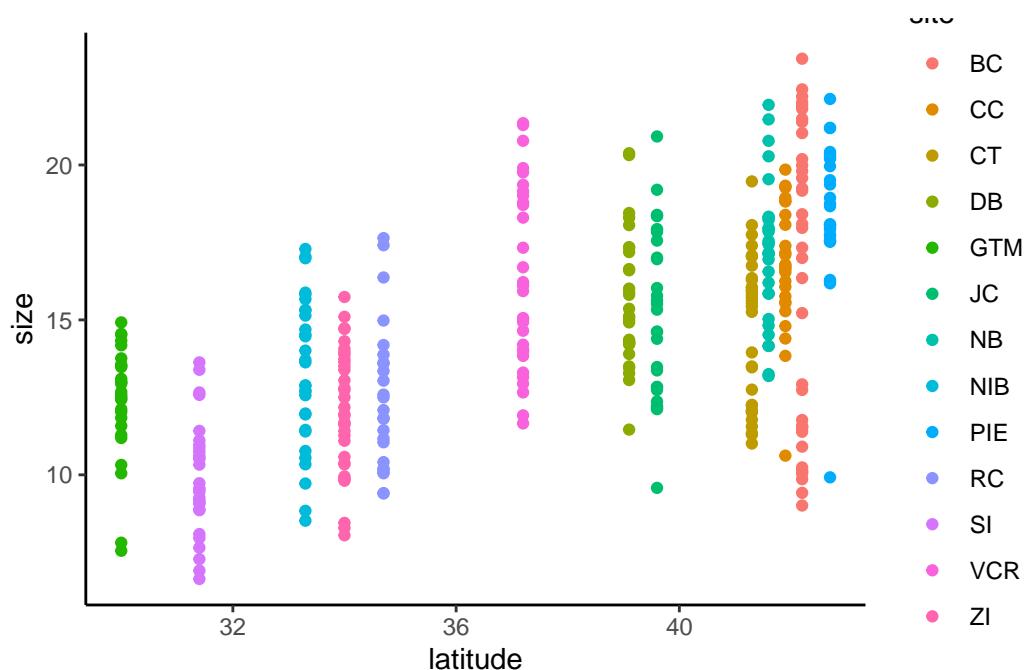
Interpret this graph in 1-2 sentences: Does it look like size increases with latitude? Describe how confident you are in this interpretation.

#A1.1: Yes it does appear that size increases with latitude. We are not that confident though because the data is very spread out.

Q1.2 Beautify this graph

Make this graph look a bit nicer! Use the skills you learned earlier in the quarter. #A1.2

```
pie_crab %>%
  ggplot(aes(x = latitude, y = size,
             color= site )) +
  geom_point() +
  theme_classic()
```



```
# Make the y-axis include 0
ylim(0, NA)
```

<ScaleContinuousPosition>
 Range:
 Limits: 0 -- 1

1.2 Fit linear regression with `brms`

Time to run the model! We will be using the `brm()` function. There's a lot here, so let's dig in line-by-line:

```
# latitude model
m.crab.lat <-
  brm(data = pie_crab, # Give the model the pie_crab data
       # Choose a gaussian (normal) distribution
       family = gaussian,
       # Specify the model here.
       size ~ latitude,
       # Here's where you specify parameters for executing the Markov chains
       # We're using similar to the defaults, except we set cores to 4 so the analysis runs fa
       iter = 2000, warmup = 1000, chains = 4, cores = 4,
       # Setting the "seed" determines which random numbers will get sampled.
       # In this case, it makes the randomness of the Markov chain runs reproducible
       # (so that both of us get the exact same results when running the model)
       seed = 4,
       # Save the fitted model object as output - helpful for reloading in the output later
       file = "output/m.crab.lat")
```

Q1.3 What does the “iter” argument do?

#A1.3: Iter is the number of total iterations per chain including warmup.

```
?brm
```

1.3 Assess model

First, we need to assess whether or not our model actually ran correctly. Let’s print a summary of the model output:

```
summary(m.crab.lat)
```

```
Family: gaussian
Links: mu = identity
Formula: size ~ latitude
Data: pie_crab (Number of observations: 392)
Draws: 4 chains, each with iter = 2000; warmup = 1000; thin = 1;
      total post-warmup draws = 4000
```

Regression Coefficients:

	Estimate	Est.Error	l-95% CI	u-95% CI	Rhat	Bulk_ESS	Tail_ESS
Intercept	-3.60	1.27	-6.05	-1.06	1.00	4312	3144
latitude	0.48	0.03	0.42	0.55	1.00	4314	3400

Further Distributional Parameters:

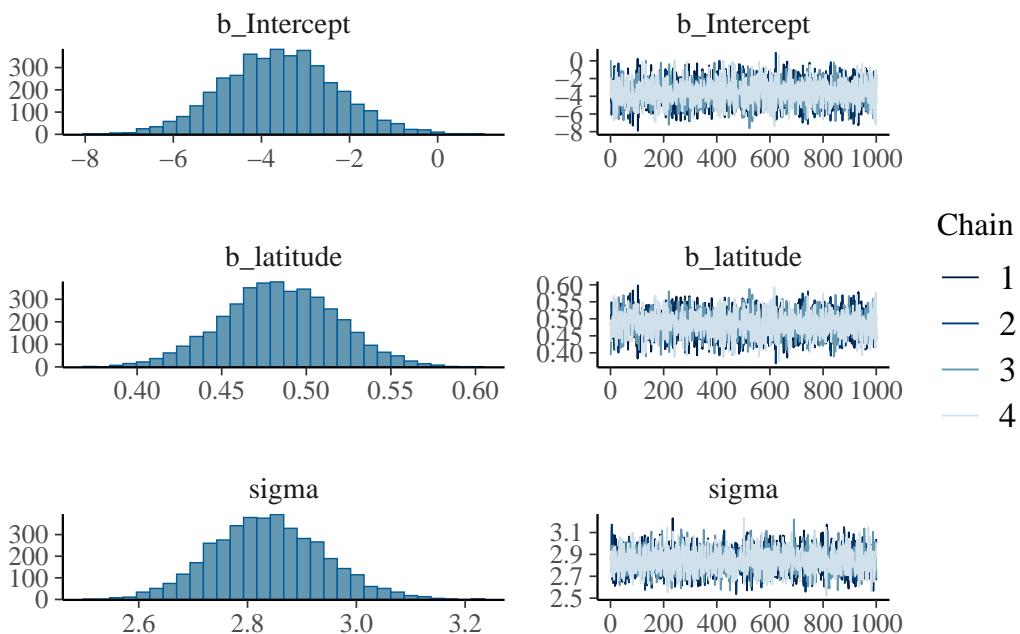
	Estimate	Est.Error	l-95% CI	u-95% CI	Rhat	Bulk_ESS	Tail_ESS
sigma	2.84	0.10	2.64	3.05	1.00	4037	2944

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).

We do this in part by looking at the Rhat (R hat) column, which should be very close to 1. If you remember from lecture, this model runs multiple chains that converge on estimates for the slope and the intercept. An R hat of 1 tells us that those four chains converged on the same estimate. This looks fine for us!

Let's also look at the chains and the “posterior distributions” in a graph.

```
plot(m.crab.lat) # show posteriors and chains
```



We're looking for three things:

1. Are the posterior samples on the left each a smooth distribution, with one clean peak, or do they have multiple clear peaks? The latter is a bad sign. They look good in this case. #yes they are all with one clean peak
2. Are the four chains on the overlapping each other, or are they clearly separate? The latter is a bad sign. We again look good in this case. #all overlap!
3. Are the four chains flat, or is there a clear trend up or down? The latter is a bad sign. We again look good in this case. #seems flat!

If we fail any of these tests, we would want to try running the MCMC chain with more iterations. We may also need to think hard about whether our model is correctly designed.

1.4 Interpret model

Now that we feel good that the model fit correctly, let's look at a summary table of our model's output.

```
summary(m.crab.lat)
```

```
Family: gaussian
Links: mu = identity
Formula: size ~ latitude
Data: pie_crab (Number of observations: 392)
Draws: 4 chains, each with iter = 2000; warmup = 1000; thin = 1;
      total post-warmup draws = 4000

Regression Coefficients:
Estimate Est.Error 1-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
Intercept -3.60     1.27    -6.05   -1.06 1.00     4312     3144
latitude   0.48     0.03     0.42     0.55 1.00     4314     3400
```

Further Distributional Parameters:

	Estimate	Est.Error	1-95% CI	u-95% CI	Rhat	Bulk_ESS	Tail_ESS
sigma	2.84	0.10	2.64	3.05	1.00	4037	2944

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).

The output reminds us of our model formula that we chose (`size ~ latitude`). Most important to answering our question are the parameter estimates in the `estimate` column. Remember, our model was $size = intercept + slope * latitude$. The `slope` parameter is going to tell us what the effect of latitude is. For every one unit change of latitude, what is the effect on size? We need to translate that interpretation to the units that those variables represent: the slope value is thus *for every one degree of latitude, carapace size changes this much in millimeters*.

Looking at our table, we see the estimate for `latitude` is 0.48: This indicates that the model estimated that for every 1 degree latitude, carapace width increases by 0.48 mm. #makes sense

Earlier you were asked to “*describe how confident you are in this interpretation*” when qualitatively interpreting the graph of `size` vs `latitude`. Now, let’s answer this quantitatively by examining how much confidence the model has in the size-latitude association. Are only positive (non-zero) slopes compatible with the data? Or would a flat (slope of zero) association also be compatible? If a slope of zero is compatible with the data, then we can’t really say that our predictor (`latitude`) has any effect on our response (`size`).

To assess this, we can look at the lower and upper 95% Credible Interval columns (`l-95% CI` and `u-95% CI`, respectively) and see if that interval range intersects with zero. We can see that our slope estimate (the estimate of `latitude`) ranges from 0.42 to 0.55 - zero is not included in this range. Therefore, we can reasonably conclude here that, given our model, the effect of latitude on body size has a 95% chance of being between those values, and importantly, NOT zero! #size must increase according to data ——

In the results section of a paper, we would write something along these lines:

We found that crab size increased with latitude, with an increase of 0.48mm of carapace width per 1 degree of latitude. Our 95% credible intervals were between 0.42 and 0.55 mm/degree, suggesting that given our model, the effect of latitude on carapace width is different from zero.

Bonus

Let’s calculate the probability of a zero slope! The MCMC chains are big columns of samples from the posterior distribution, so we can add up the proportion of slope estimates that are zero or lower.

```
as_draws_df(m.crab.lat) %>% # extract the posterior samples from the model estimate
  select(b_latitude) %>% # pull out the latitude samples from all 4 chains. we'll get a wa
  summarize(p_slope_lessthanorequalto_zero = sum(b_latitude <= .48)/length(b_latitude))
```

Warning: Dropping 'draws_df' class as required metadata was removed.

```
# A tibble: 1 x 1
  p_slope_lessthanorequalto_zero
  <dbl>
1 0.452
```

Ok! The model reports a zero chance of a slope less than or equal to zero. Feel free to try other thresholds to explore (e.g., what's the probability the slope is greater than 0.5 mm per degree latitude?).

1.5 Plot model on the data

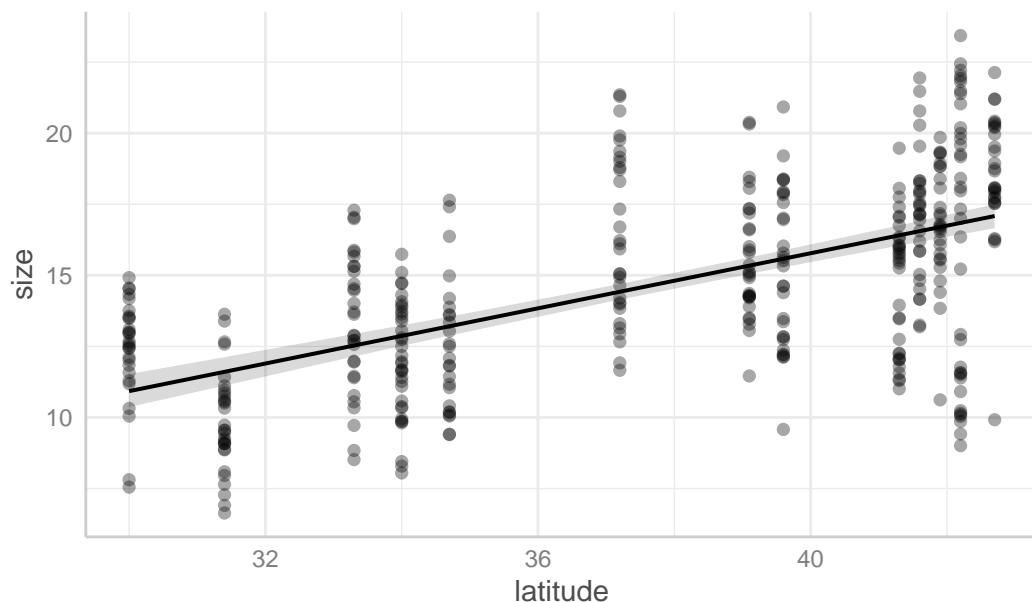
Here we are going to plot the data and the model output in two different ways:

- **Compatibility interval** shows uncertainty in the average response (the estimate for the mean)
- **Prediction interval** shows uncertainty in the data around the average response (the error)

```
# compatibility interval. the shows uncertainty in the average response.
confm.crab.lat <- predict_response(m.crab.lat)
plot(confm.crab.lat, show_data = TRUE)
```

Data points may overlap. Use the `jitter` argument to add some amount of random variation to the location of data points and avoid overplotting.

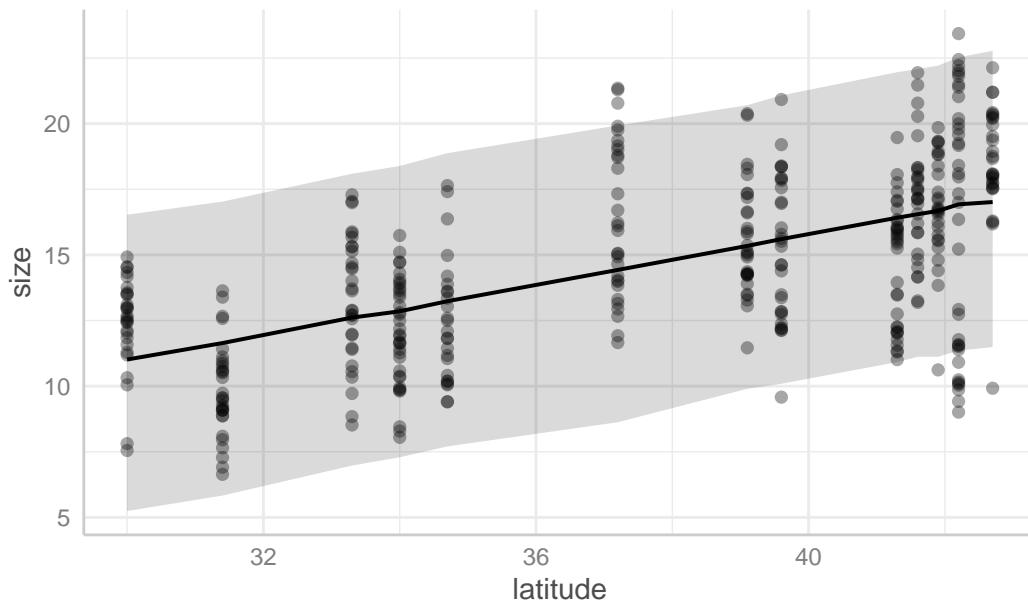
Predicted values of size



```
# prediction interval. this shows uncertainty in the data around the average response.  
confm.crab.lat <- predict_response(m.crab.lat, interval = 'prediction')  
plot(confm.crab.lat, show_data = TRUE)
```

Data points may overlap. Use the `jitter` argument to add some amount of random variation to the location of data points and avoid overplotting.

Predicted values of size



1.6 Repeat with a new variable: water temp sd

Let's repeat this example with a new variable: the water temperature standard deviation, `water_temp_sd`. The standard deviation (sd) can be used as a metric of variability: higher sd means higher variability. We can ask: *is higher variability in water temperature associated with fiddler crab body size?*

Q1.4 Make a hypothesis

Before you look at the data, what direction of an effect do you expect? Do you think higher variability would be associated with larger or smaller crabs? Why? Please write 1-2 sentences.

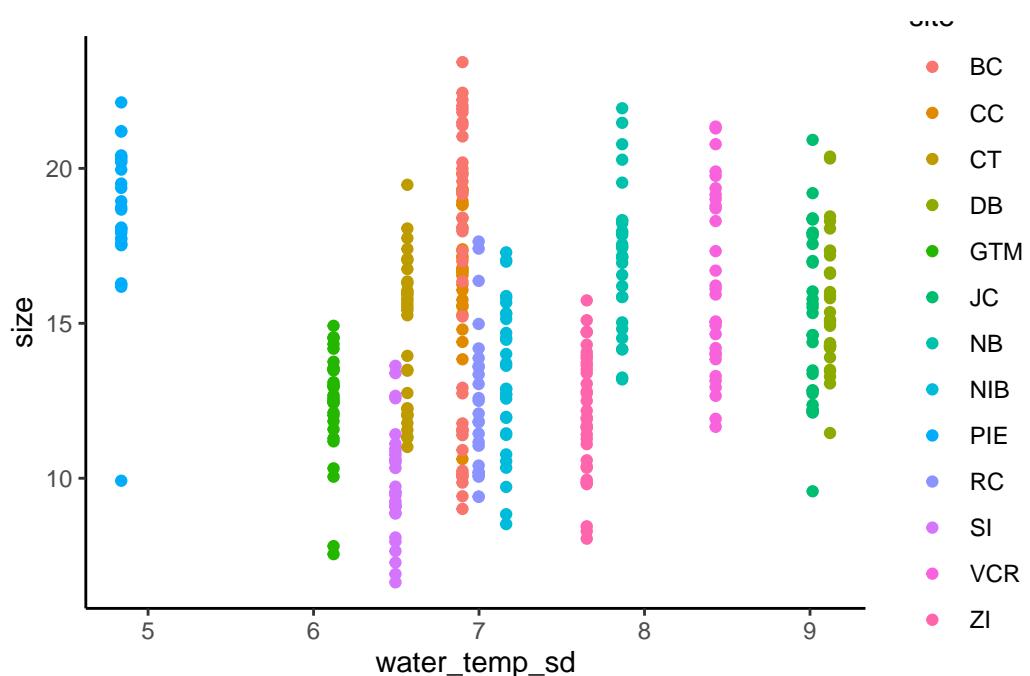
#A1.4:

Yes, there would be larger body sizes with higher variablity in water temp.

Q1.5 Graph the data

#A1.5

```
pie_crab %>%
  ggplot(aes(x = water_temp_sd, y = size,
             color= site )) +
  geom_point() +
  theme_classic()
```



```
# Make the y-axis include 0
ylim(0, NA)
```

```
<ScaleContinuousPosition>
Range:
Limits:    0 -- 1
```

Q1.6 Interpret the graph

Does it look like size changes with the sd of water temperature? Describe how confident you are in this interpretation.

#A1.6:

**It looks like there size doesn't change significantly with the sd of water temp.
We are pretty confident there is not a relationship.**

Q1.7 Set up and run this new model

#A1.7

```
# water_temp_sd model
m.crab.watersd <-
  brm(data = pie_crab, # Give the model the pie_crab data
       # Choose a gaussian (normal) distribution
       family = gaussian,
       # Specify the model here.
       size ~ water_temp_sd,
       # Here's where you specify parameters for executing the Markov chains
       # We're using similar to the defaults, except we set cores to 4 so the analysis runs faster
       iter = 2000, warmup = 1000, chains = 4, cores = 4,
       # Setting the "seed" determines which random numbers will get sampled.
       # In this case, it makes the randomness of the Markov chain runs reproducible
       # (so that both of us get the exact same results when running the model)
       seed = 4,
       # Save the fitted model object as output - helpful for reloading in the output later
       file = "output/m.crab.watersd")
```

Q1.8 Assess the model

Assess whether the model ran correctly by looking at R hat, the chains, and the posterior distributions using the plot() and summary() functions as below. Describe your thought process about whether the model ran correctly in 1-2 sentences. #A1.8

```
# show posteriors and chains  
plot(m.crab.watersd)  
  
# show summary, including rhat  
summary(m.crab.watersd)
```

Q1.9 Interpret the model

Based on the summary output, interpret your model by answering: #A1.9

1. It is 0.1. This means that for every 1 degree C, more variable the water temperature becomes. The crab carapace size increases by 0.1mm.
2. No it is not. The confidence interval does intercept with zero. The lower CI is negative.

```
# Show model output  
summary(m.crab.watersd)
```

This is a situation where our predictor variable, `water_temp_sd`, does not seem to have an effect on the body size of crabs. We would say something along the lines of: *We found an increase of 0.10mm of carapace width per 1 unit of the standard deviation of water temperature, but our 95% credible intervals included zero (-0.20 to 0.40), suggesting that given our model, the effect of water temperature standard deviation on carapace width is not different from zero.*

2. Back to Pikas!



Figure 1: adultPikaNiwotRidge_SaraMcLaughlin

We can't stay away from the cute pikas for too long! In this section you will apply the statistical thinking you've learned to the pika dataset in one of two ways: you will try and see whether or not the stress of pikas is explained by either 1) elevation or 2) day of year.

Let's look at the data again

```
head(nwt_pikas)
```

```
# A tibble: 6 x 8
  date      site    station utm_easting utm_northing sex concentration_pg_g
  <date>    <fct>   <fct>        <dbl>         <dbl> <fct>           <dbl>
1 2018-06-08 Cable Ga~ Cable ~       451373     4432963 male            11563.
```

```

2 2018-06-08 Cable Ga~ Cable ~      451411    4432985 male      10629.
3 2018-06-08 Cable Ga~ Cable ~      451462    4432991 male      10924.
4 2018-06-13 West Kno~ West K~     449317    4434093 male      10414.
5 2018-06-13 West Kno~ West K~     449342    4434141 male      13531.
6 2018-06-13 West Kno~ West K~     449323    4434273 <NA>      7799.
# i 1 more variable: elev_m <dbl>

```

Date is one of the columns, but we specifically want “day of year” as a metric to quantify how late in the season it is. This also allows us to interpret our model’s output a little more informatively.

We can extract day of year using the `lubridate` package (within `tidyverse`), which is all about working with dates and times:

```

nwt_pikas_doy <- nwt_pikas %>%
  # Add a new column called day_of_year
  # yday extracts the day of year from the date column
  mutate(day_of_year = yday(date)) %>%
  # relocate the day_of_year column after the date column
  relocate(day_of_year, .after = date)

head(nwt_pikas_doy)

# A tibble: 6 x 9
  date      day_of_year site       station      utm_easting utm_northing sex
  <date>        <dbl> <fct>       <fct>           <dbl>          <dbl> <fct>
1 2018-06-08      159 Cable Gate Cable Gate 1      451373        4432963 male
2 2018-06-08      159 Cable Gate Cable Gate 2      451411        4432985 male
3 2018-06-08      159 Cable Gate Cable Gate 3      451462        4432991 male
4 2018-06-13      164 West Knoll West Knoll 3     449317        4434093 male
5 2018-06-13      164 West Knoll West Knoll 4     449342        4434141 male
6 2018-06-13      164 West Knoll West Knoll 5     449323        4434273 <NA>
# i 2 more variables: concentration_pg_g <dbl>, elev_m <dbl>

```

Q2.1 Make a question

Clearly articulate the question that you want to ask in one sentence.

#A2.1: Is there is clear relationship between stress and elevation in Pikas?

Q2.2 Make a hypothesis

Before you look at the data, what direction of an effect do you expect? Do you think a larger value of the predictor you chose would be associated with more or less stressed pikas? Why? Please write 1-2 sentences.

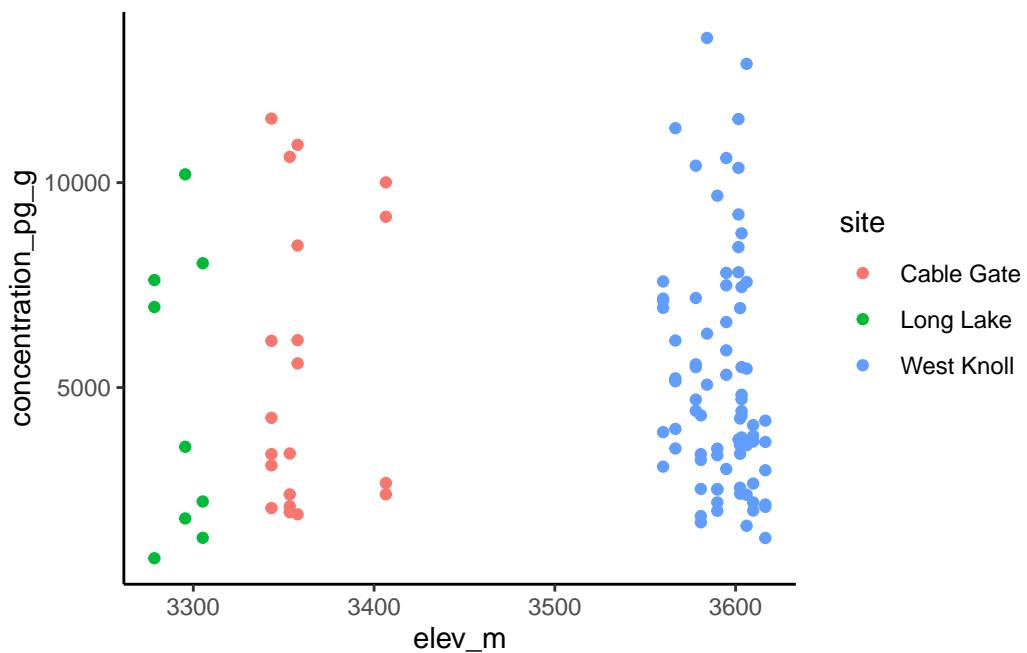
#A2.2: They will be more stressed in high elevations due to shorter growing seasons and lower temps. —

Q2.3 Graph the data

As before, your response variable (stress, which is measured as `concentration_pg_g`) should be on the y-axis, with your predictor variable on the x.

#A2.3

```
nwt_pikas_doy %>%
  ggplot(aes(x = elev_m, y = concentration_pg_g,
             color= site )) +
  geom_point() +
  theme_classic()
```



```
# Make the y-axis include 0  
ylim(0, NA)
```

```
<ScaleContinuousPosition>  
Range:  
Limits: 0 -- 1
```

Q2.4 Set up and run a model

Make sure you store your model output as something informative to you.

```
#A2.4
```

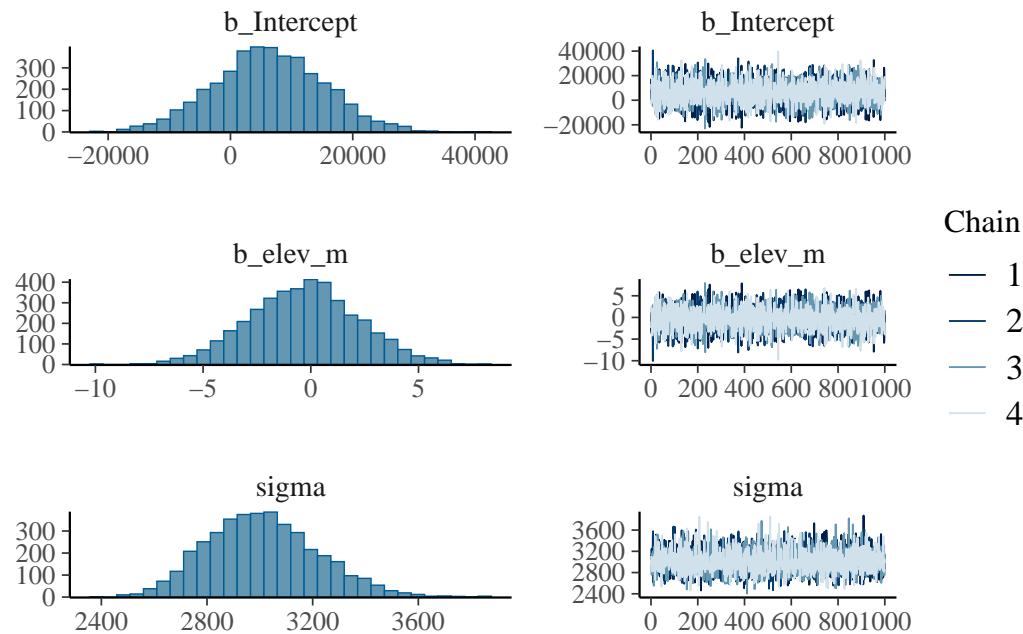
```
# pika elevation model  
m.pika.elev <-  
  brm(data = nwt_pikas_doy, # Give the model the pika data  
        # Choose a gaussian (normal) distribution  
        family = gaussian,  
        # Specify the model here.  
        concentration_pg_g ~ elev_m,  
        # Here's where you specify parameters for executing the Markov chains  
        # We're using similar to the defaults, except we set cores to 4 so the analysis runs faster  
        iter = 2000, warmup = 1000, chains = 4, cores = 4,  
        # Setting the "seed" determines which random numbers will get sampled.  
        # In this case, it makes the randomness of the Markov chain runs reproducible  
        # (so that both of us get the exact same results when running the model)  
        seed = 4,  
        # Save the fitted model object as output - helpful for reloading in the output later  
        file = "output/m.pika.elev")  
#sucessing running model
```

Q2.5 Assess the model

Assess whether the model ran correctly by looking at R hat, the chains, and the posterior distributions. Describe your thought process about whether the model ran correctly in 1-2 sentences.

#A2.5

```
# show posteriors and chains
plot(m.pika.elev)
```



```
# show summary, including rhat
summary(m.pika.elev)
```

Family: gaussian
Links: mu = identity
Formula: concentration_pg_g ~ elev_m
Data: nwt_pikas_doy (Number of observations: 109)
Draws: 4 chains, each with iter = 2000; warmup = 1000; thin = 1;
total post-warmup draws = 4000

Regression Coefficients:

	Estimate	Est.Error	1-95% CI	u-95% CI	Rhat	Bulk_ESS	Tail_ESS
Intercept	6274.92	8927.98	-11361.19	23720.60	1.00	3832	2607
elev_m	-0.31	2.53	-5.25	4.69	1.00	3830	2635

Further Distributional Parameters:

```
Estimate Est.Error 1-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
```

```
sigma 3011.79    209.67  2641.09  3445.75 1.00      4000      2896
```

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).

The model did run correctly, rhat=1. The distributions appear single peaked and the chains are overlapping without trending up or down.

Q2.6 Interpret the model

#A2.6: 1. With every increase in one meter of elevation, there is decrease of 0.31 grams of stress chemical.

2. No, the effect is not reasonably different from zero. The credible intervals straddle zero.

Q2.7 Plot the model on the data

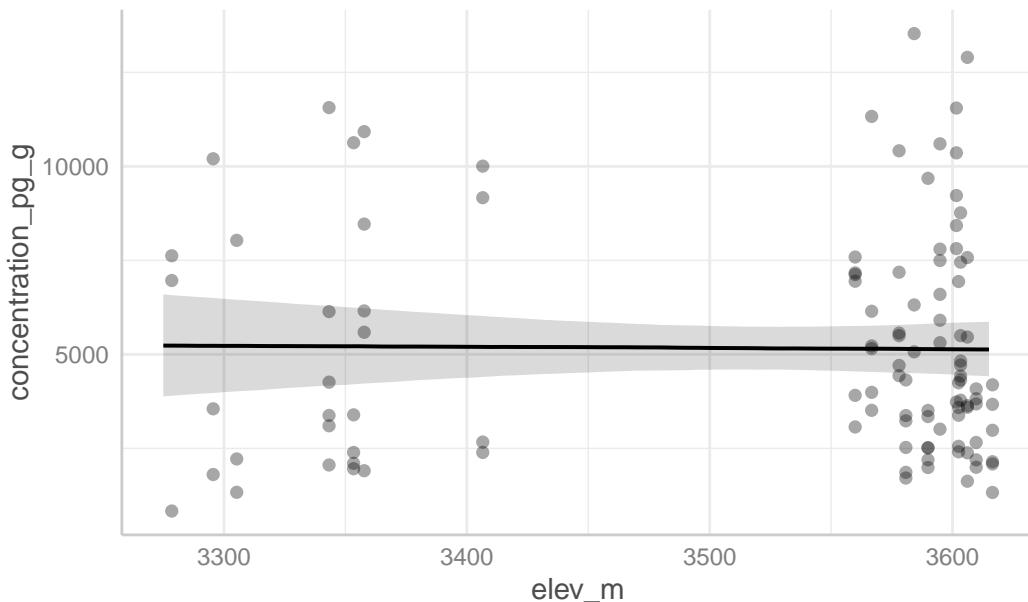
Plot either a compatibility interval or prediction interval on the data; specify which you are using. #A2.7:

we are using a compatibility interval

```
# compatibility interval. This shows uncertainty in the average response.
confm.pika.elev <- predict_response(m.pika.elev)
plot(confm.pika.elev, show_data = TRUE)
```

Data points may overlap. Use the `jitter` argument to add some amount of random variation to the location of data points and avoid overplotting.

Predicted values of concentration_pg_g



Q2.8 Write a small results paragraph

Including the information from Q2.6, write 2-3 sentences as if you were writing the results section of a scientific paper. Include a conclusion sentence that summarizes your finding.

#A2.8 We found that with every 1 meter increase in elevation, there is a 0.31 decrease in grams of stress indicator compounds in pika feces. However, the lower CI of 95% was -5.25 and the upper CI was 4.69, meaning we cannot determine the direct effect of elevation on pika stress. In summary, we could not demonstrate that elevation effected pika stress in our study. —————