Bayesian models - lab 2

< your name here>

BIOHOPK 143H - Winter 2021

Load and plot data

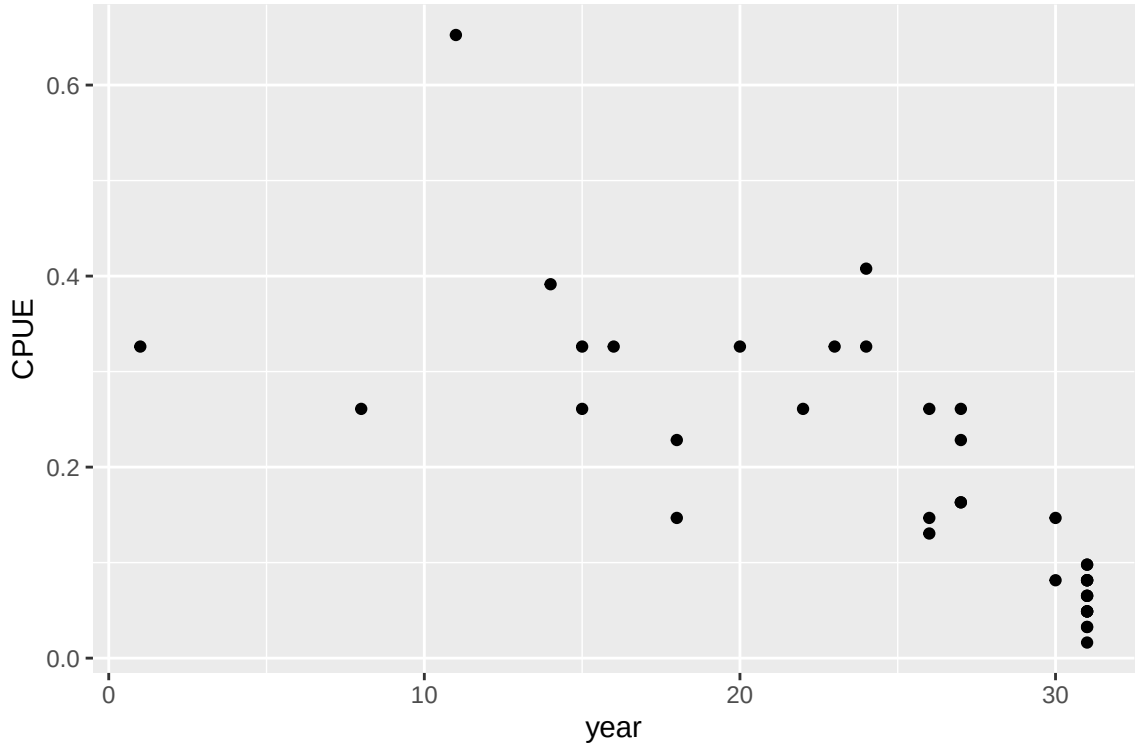
These data are from interviews conducted with fishers by Dr. Tim White on the island of Teraina (Central Pacific, Northern Line Islands). At the time the data were collected, there were 8 active fishing vessels and 17 active fishers on the island. The number of active fishing vessels is known for all years prior (stored in `boat_data`), and the catch-per-unit-effort (CPUE) from fisher interviews is stored in `CPUE_data`.

The CPUE data have been pre-adjusted for fisher bias by taking inventory of stores of dried shark fins, which are stored by fishers until larger ships visit the island to purchase them, and are thus a reliable estimate of fisher catch. The CPUE data have also been adjusted for the ratio of sharks caught which are grey reef sharks, the focal species of this case study. The assumptions implicit in these adjustments are that fisherman bias and the proportion of sharks caught which are grey reef sharks are constant over time.

Catch per unit effort

Plotting the adjusted CPUE (grey reef sharks caught per day of fishing) over time, we can make out a general decline in catch per day over time.

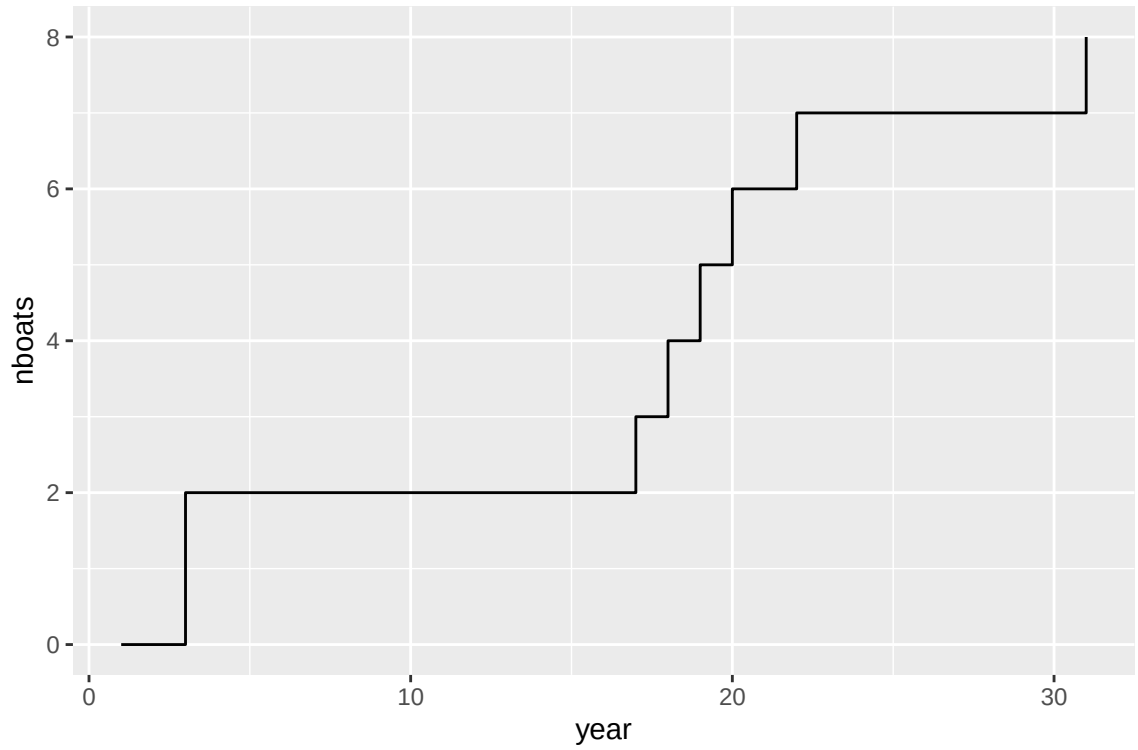
```
GRS_data <- readRDS("GRS_data.rds")
attach(GRS_data)

CPUE_data %>%
  ggplot(aes(year, CPUE)) +
  geom_point()
```



Number of fishing boats

We see that, simultaneously, the number of active fishing vessels over time has increased.

```
boat_data %>%
  ggplot(aes(year, nboats)) +
  geom_step()
```



The Beverton-Holt Model

We used a Beverton-Holt model to approximate the dynamics of the grey reef shark population on Teraina over time. We initialize the population at the carrying capacity K, which we approximate by taking previous estimates of the carrying capacity from a nearby island, and scaling it down based on the difference in the extent of reef habitat between the two islands:

$$N_0 = K$$

For each time step ($\Delta t = 1$ year) after $t = 0$, the number of grey reef sharks in the next time step is the number of individuals in the current time step, plus the change in population size due to natural processes, minus the number of reef sharks removed across the fishing season:

$$N_{t+1} = N_t^+ e^{-\bar{q} d_t B_t}$$

Where N_t^+ is given by the Beverton-Holt model:

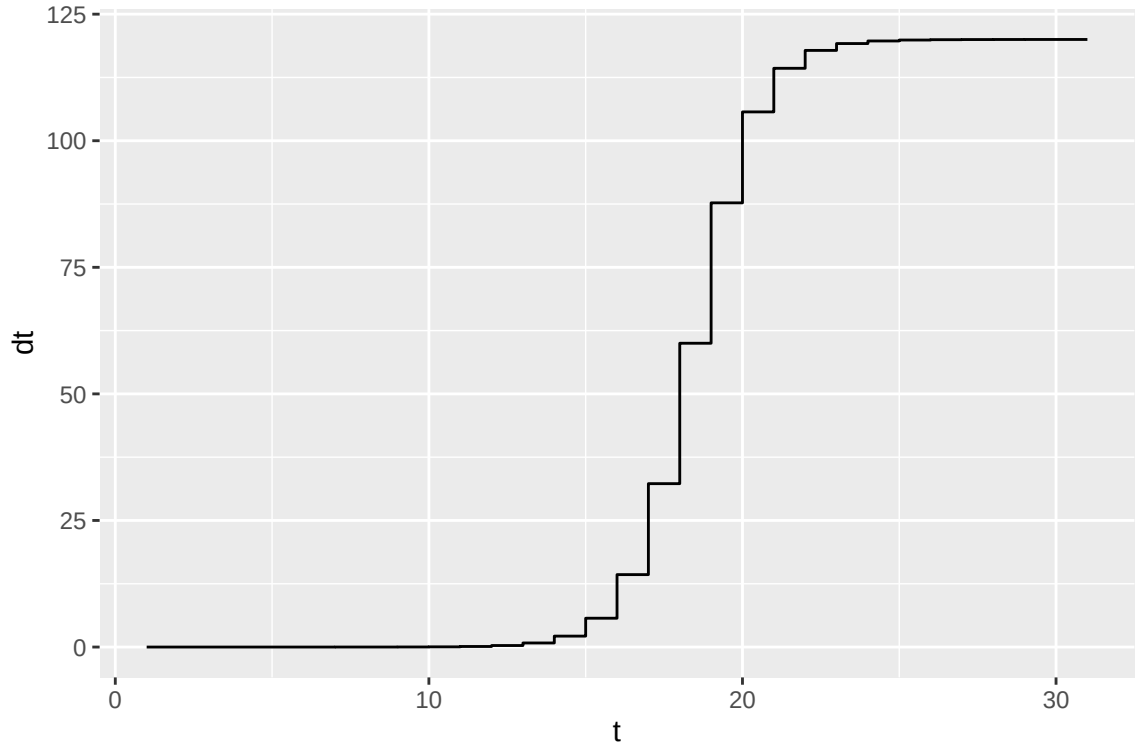
$$N_t^+ = \frac{\lambda N_{t-1}}{1 + N_t \left(\frac{\lambda - 1}{K}\right)}$$

and where $\bar{q}$ is the overall fisherman catchability, B_t is the active number of fishing boats in year t, and d_t is the number of fishing days in the fishing season for year t. It is given by a sigmoid function with a slope β_d, a midpoint t_{mid}, and a maximum number of fishing days $d_{\max}$:

$$d_t = \frac{d_{\max}}{1 + e^{(-\beta_d \times (t - t_{\text{mid}}))}}$$

You can examine the behavior of this function by changing the parameters `dmax`, `midpoint`, and `slope` in the function call below. Describe how each of these paramters affect the shape of the sigmoid function - what are the implications of these for the model?

```
### sigmoid curve for number of fishing days
nfdays = function(t, dmax, midpoint, slope){
  dmax/(1 + exp(-slope * (t - midpoint)))
}

t <- 1:nrow(boat_data)
dt <- nfdays(t, dmax = 120, midpoint = 18, slope = 1) ## change pars here

qplot(t, dt, geom = "step")
```



Finally, the catch per unit effort under this model is given by:

$$\mu_{\text{CPUE},t,i} = N_t^+ \left(1 - e^{-q_i}\right) \; ; \; i \in (1, 2, \ldots, n_{\text{fisher}})$$

Where each fisher has their own catchability q_i, which is connected to the overall catchability $\bar{q}$ as described below. The catch-per-unit effort estimates from this model are what connects the underlying population trends to the observed data.

Bayesian Model Structure

The full structure of the Bayesian model is:

$$
\begin{aligned}
\text{CPUE}_{t,i} &\sim \text{Normal}(\mu_{\text{CPUE},t,i}, \sigma) \\
N_{t+1} &= N_t^+ e^{-\bar{q} d_t B_t} \; ; \; d_t = \frac{d_{\max}}{1 + e^{(-\beta_d \times (t - t_{\text{mid}}))}} \\
\mu_{\text{CPUE},t,i} &= N_t^+ \left(1 - e^{-q_i}\right) \; ; \; i \in (1, 2, \ldots, n_{\text{fisher}}) \\
N_t^+ &= \frac{\lambda N_{t-1}}{1 + N_t \left(\frac{\lambda - 1}{K}\right)} \\
N_0 &= K \\
q_i &\sim \text{Normal}(\bar{q}, \sigma_q)[0, \infty] \\
\bar{q} &\sim \text{Normal}(0.01, 0.05)[0, \infty] \\
\sigma_q &\sim \text{Half-Normal}(0, 0.05) \\
d_{\max} &\sim \text{Normal}(115, 18) \\
\sigma &\sim \text{Half-Normal}(0, 0.5)
\end{aligned}
$$

Where the first line indicates that we are assuming a Normal likelihood, the next four lines are our deterministic model as described above, and the lines that follow are our **priors**.

Some of the parameters in this model are fixed - it turns out, for example, that the slope parameter for the sigmoid fishing days function has virtually no effect on the fit of the data to the model, so we just chose a value for it and fixed it there. And, the finite growth rate λ and the carrying capacity K have been estimated by previous studies, so we'll fix those as well (although, really, it would be better to propogate the uncertainty in those studies forward into our study by including the error around their estimates).

This leaves the catchability q (at both the overall and fisher levels, as described below) as well as the maximum number of fishing days $d_{\max}$, and a couple of error terms σ.

Besides the ways I've mentioned already, can you think of at least one way in which this model could be improved? Think, for example, about the distributions I've used here for either the likelihood or the priors - what other distributions might be more appropriate, and why?

Prior Distributions

Maximum number of fishing days

Since the maximum number of fishing days, $d_{\max}$, scales the amount of fishing effort, the CPUE data contain information about this parameter. However, we have other, more direct information about this parameter that we'd like to include - besides recording the number of fins caught over his time on the island, Tim also recorded the number of days that six of the fishers spent on the water that year. We can incorporate these data into the model using a prior - in this case, we've set the mean for the prior on $d_{\max}$ to the mean number of days from these six fishermen, and the standard deviation to the standard error of the mean:

$$
d_{\max} \sim \text{Normal}(115, 18)
$$

Catchability

In this model, the data are not independent - they're clustered by fisher. Some fishers have reported CPUEs for just one time step, while others have reported CPUEs for two or three time steps - there are 39 data points, but only 17 fishers.

One way to account for the lack of independence in this data is to use a **hierarchical model** - one in which the errors, or the parameters, are clustered. In this model, we could add a second error term so that there is observation error and additional error due to differences among fishermen:

$$\text{CPUE}_{t,i} = \mu_{\text{CPUE},t} + \epsilon + \epsilon_{\text{fisher}}$$

Or, we could choose one or more of the model parameters and allow those to vary by fisher. We have to be careful with this, however, as if we allow parameters which describe *one* underlying population to vary among fishers, we'd actually be giving each fisher their own population trajectory, which doesn't make sense.

So, what we'll do here is allow each fisher to have their own catchability, q_i, but we'll only allow there to be one catchability value, $\bar{q}$, that impacts the overall population trajectory. $\bar{q}$ represents the mean of a Normal distribution that the individual fisher catchabilities are drawn from, similar to how the observations are normally distributed around a grand mean:

$$\begin{aligned} q_i &\sim \text{Normal}(\bar{q}, \sigma_q)[0, \infty] \\ \bar{q} &\sim \text{Normal}(0.01, 0.05)[0, \infty] \\ \sigma_q &\sim \text{Half-Normal}(0, 0.05) \end{aligned}$$

We call the parameters $\bar{q}$ and σ_q **hyperparameters**, and the prior on $\bar{q}$ a **hyperprior**. Otherwise, there's nothing very special about these priors - I've constrained them to be non-negative, and I've made them *extremely* wide so that these priors are almost entirely determined by the data - when you compare the posterior for $\bar{q}$ to these priors, it should become apparent that the posterior is much, much narrower than these priors are.

Would it be possible to implement a similar hierarchical structure for the maximum number of fishing days? Why or why not?

Compile and sample from model

Now, let's compile this model and sample from the posterior distributions of our parameters. We'll take 5000 samples from the posterior distribution.

```
### Compile Stan model
grs_model <- stan_model("GRS_Multilevel_Model.stan")

### set number of samples from the posterior distribution
n_warmup <- 1000
n_samples <- 5000

### Data list to pass to Stan model
grs_stan_data <- list(
  n_obs = nrow(CPUE_data), ## number of observations
  n_fisher = length(unique(CPUE_data$id)), ## number of fishermen
  fisher_id = CPUE_data$id, ## fishermen IDs
  year = CPUE_data$year, ## years corresponding to observed CPUE
  CPUE = CPUE_data$CPUE, ## observed CPUE
  n_boats = boat_data$nboats, ## number of boats over time
  tmax = max(CPUE_data$year), ## maximum time step in data
  K = pars$K, ## carrying capacity
  lambda = pars$lambda, ## finite rate of increase
  alpha = pars$alpha, ## ordinarily, (lambda - 1)/K
  midpoint = pars$midpoint, ## midpoint of sigmoid fishing days curve
  slope = pars$slope ## slope of sigmoid fishing days curve
)
```

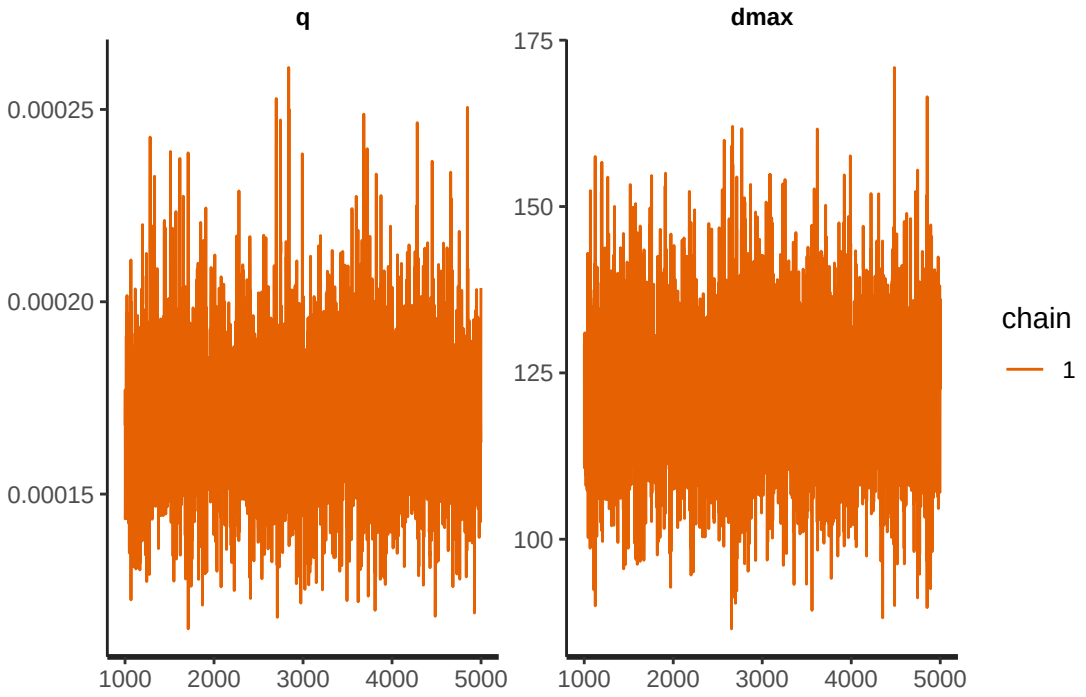
```
### sample from the posterior
grs_samples <- sampling(
  grs_model,
  data = grs_stan_data,
  iter = n_samples,
  warmup = n_warmup,
  chains = 1
)
```

```
##
## SAMPLING FOR MODEL 'GRS_Multilevel_Model' NOW (CHAIN 1).
## Chain 1:
## Chain 1: Gradient evaluation took 0 seconds
## Chain 1: 1000 transitions using 10 leapfrog steps per transition would take 0 seconds.
## Chain 1: Adjust your expectations accordingly!
## Chain 1:
## Chain 1:
## Chain 1: Iteration:    1 / 5000 [  0%]  (Warmup)
## Chain 1: Iteration:  500 / 5000 [ 10%]  (Warmup)
## Chain 1: Iteration: 1000 / 5000 [ 20%]  (Warmup)
## Chain 1: Iteration: 1001 / 5000 [ 20%]  (Sampling)
## Chain 1: Iteration: 1500 / 5000 [ 30%]  (Sampling)
## Chain 1: Iteration: 2000 / 5000 [ 40%]  (Sampling)
## Chain 1: Iteration: 2500 / 5000 [ 50%]  (Sampling)
## Chain 1: Iteration: 3000 / 5000 [ 60%]  (Sampling)
## Chain 1: Iteration: 3500 / 5000 [ 70%]  (Sampling)
## Chain 1: Iteration: 4000 / 5000 [ 80%]  (Sampling)
## Chain 1: Iteration: 4500 / 5000 [ 90%]  (Sampling)
## Chain 1: Iteration: 5000 / 5000 [100%]  (Sampling)
## Chain 1:
## Chain 1:  Elapsed Time: 5.945 seconds (Warm-up)
## Chain 1:                18.829 seconds (Sampling)
## Chain 1:                24.774 seconds (Total)
## Chain 1:
```

Traceplots

Take a look at the traceplots for two of the parameters - the overall catchability $\bar{q}$ and the maximum number of fishing days $d_{\max}$. What are these supposed to look like for a "healthy" model? How do these traceplots look?

```
traceplot(grs_samples, c("q", "dmax"))
```

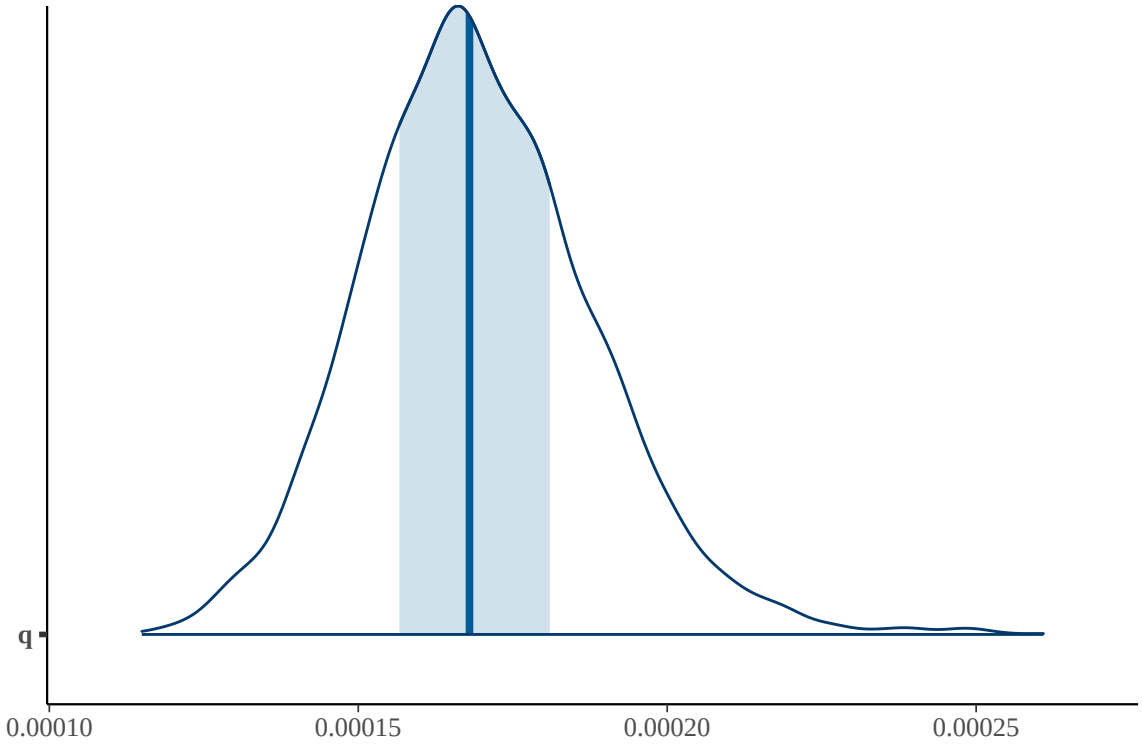


Posterior Distributions

Overall Catchability ($\bar{q}$)

The posterior mean for the overall catchability is very small, and the standard deviation is about 0.00002 - compare that to the standard deviation of the prior, which as 0.05. This suggests that the data provide *a lot* of information about the catchability, or, equivalently, that the catchability paramter effects the fit of the model to the data quite a bit.

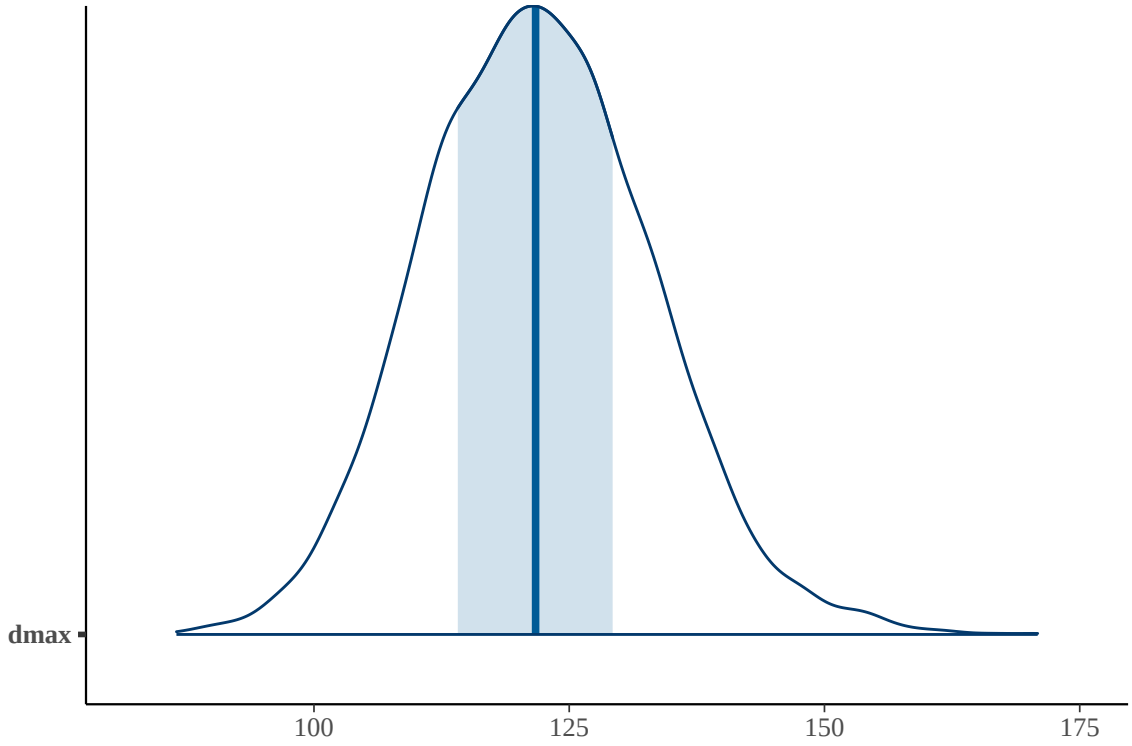
```
mcmc_areas(grs_samples, "q")
```



Maximum Number of Fishing Days

By contrast, there's quite a bit of uncertainty in the maximum number of fishing days, $d_{\max}$ - the posterior mean (≈ 120) is only a little higher than the prior mean (≈ 115), and the sd (≈ 10) is only a little narrower than the prior sd (≈ 18). This suggests that, in contrast to the catchability paramter, the CPUE data do not contain all that much information about the number of fishing days.

```
mcmc_areas(grs_samples, "dmax")
```



Posterior Predictions

Now, we'll extract the posterior samples for the overall catchability and the maximum number of fishing days, and loop over those samples, storing a simulated population trajectory for each one.

Obtaining Predictions

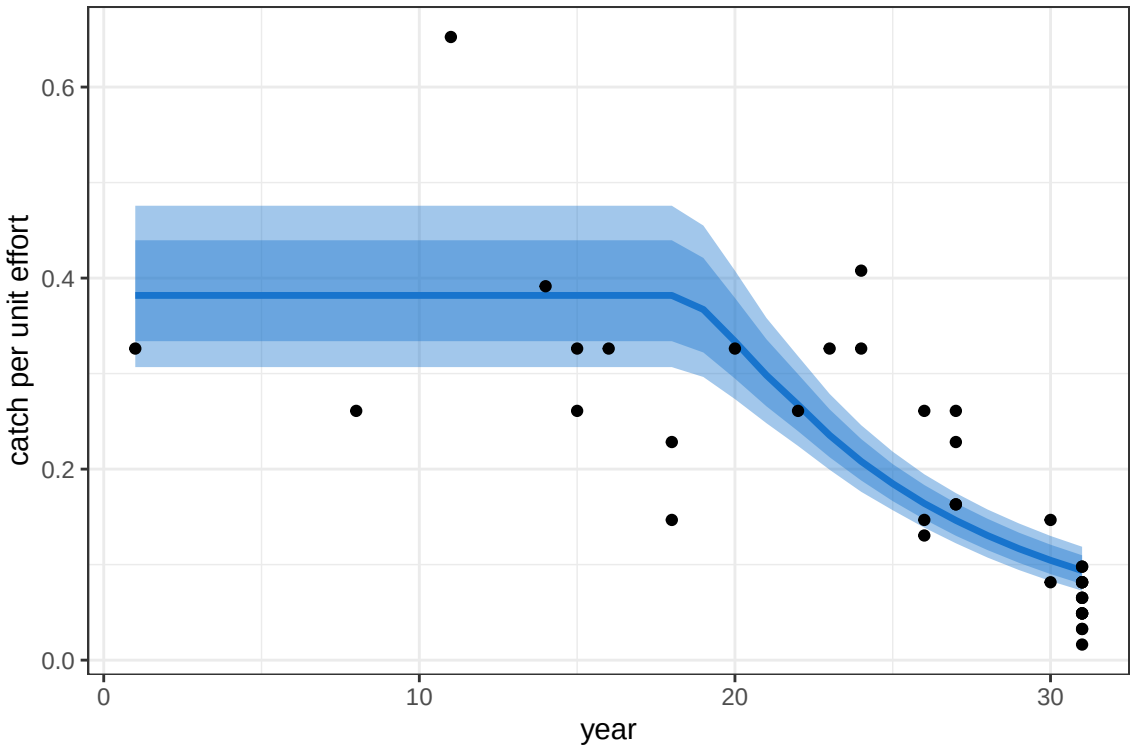
```
### extract posterior samples
q_samples <- c(rstan::extract(grs_samples, "q", inc_warmup = FALSE)$q)
dmax_samples <- c(rstan::extract(grs_samples, "dmax", inc_warmup = FALSE)$dmax)

### years to plot conditional
sim_years <- 1:max(CPUE_data$year)

### list to store fitted curves
fit_sims <- vector("list", length(q_samples))

### loop over samples, store fitted values
for (i in 1:length(q_samples)) {

  ### empty vectors to store values
  N <- rep(NA, length(sim_years))
  Np <- rep(NA, length(sim_years))
  Yield <- rep(NA, length(sim_years))
  CPUE <- rep(NA, length(sim_years))

  N[1] <- pars$K ## initial population size
```

```
  for (j in seq(length(sim_years))) {

    nfdays_t <- nfdays(j, dmax_samples[i], pars$midpoint, pars$slope)
    nboats_t <- boat_data$nboats[boat_data$year == j]

    Np[j] <- pars$lambda*N[j] / (1 + pars$alpha*N[j])
    Yield[j] <- Np[j] * (1 - exp(-q_samples[i] * nfdays_t * nboats_t))
    CPUE[j] <- Np[j] * (1 - exp(-q_samples[i]))
    if (j < length(sim_years)) {
      N[j + 1] <- Np[j] * exp(-q_samples[i] * nfdays_t * nboats_t)
    }
  }

  fit_sims[[i]] <- data.frame(
    sample = i, year = sim_years, N = N, Yield = Yield, CPUE = CPUE
  )
}

fit_sims <- do.call("rbind", fit_sims)
```

Catch Per Unit Effort

Here's our posterior distribution for the catch per unit effort over time - or, at least, here's our posterior mean with an 80% and 95% compatability band. Keeping in mind the assumptions of using a Normal likelihood specifically, how does this model look to you?

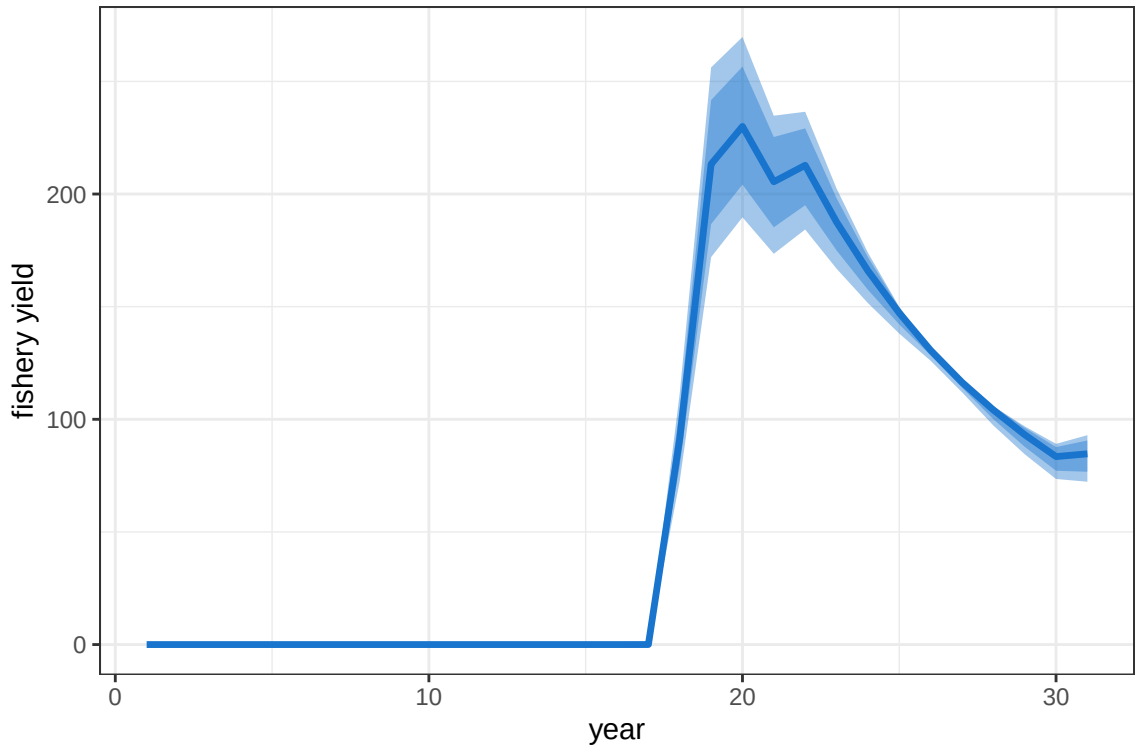
```
fit_sims %>%
  group_by(year) %>%
  summarise(
    CPUE = quantile(CPUE, c(0.025, 0.1, 0.5, 0.9, 0.975)),
    q = c(0.025, 0.1, 0.5, 0.9, 0.975)
  ) %>%
  pivot_wider(names_from = q, values_from = CPUE) %>%
  ggplot(aes(year, `0.5`)) +
  geom_ribbon(aes(ymin = `0.025`, ymax = `0.975`), alpha = 0.4, fill = "dodgerblue3") +
  geom_ribbon(aes(ymin = `0.1`, ymax = `0.9`), alpha = 0.4, fill = "dodgerblue3") +
  geom_line(color = "dodgerblue3", size = 1.2) +
  geom_point(aes(x = year, y = CPUE), data = CPUE_data) +
  ylab("catch per unit effort") +
  theme_bw()
```



Fishery Yield

Here is the trend in fishery yield implied by this model, with 80% and 95% compatability bands. How might this look if we were to allow the midpoint of the number of fishing days sigmoid function to vary in the model?

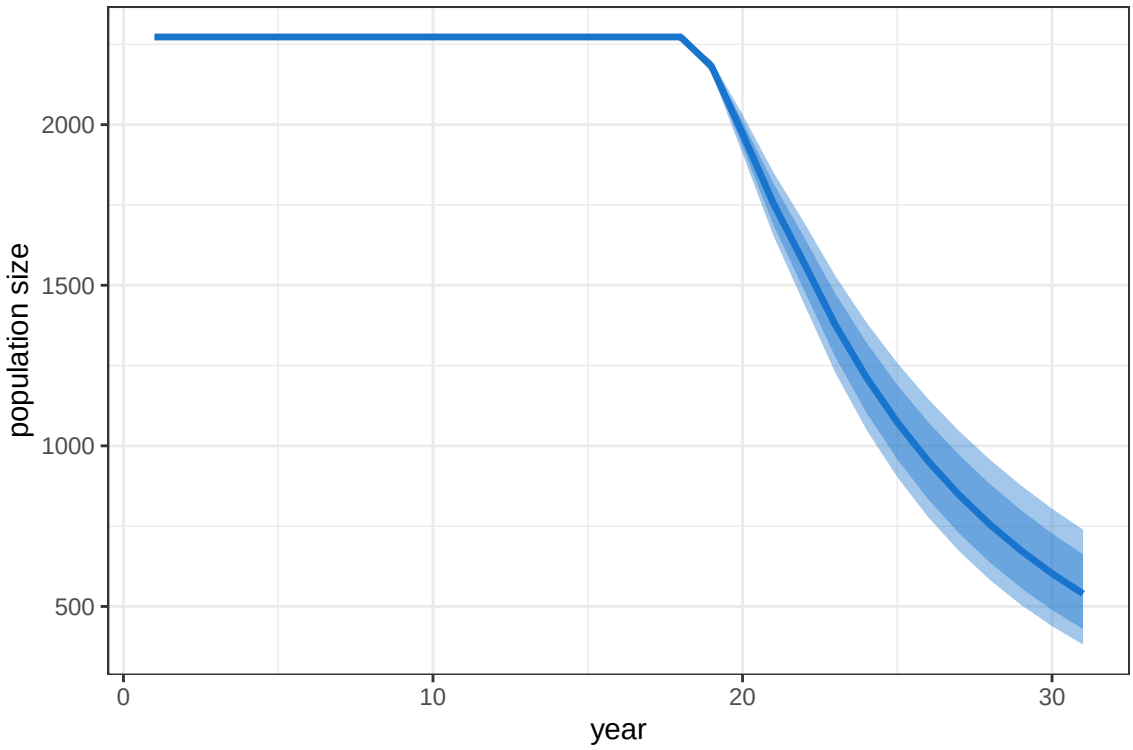
```
fit_sims %>%
  group_by(year) %>%
  summarise(
    Yield = quantile(Yield, c(0.025, 0.1, 0.5, 0.9, 0.975)),
    q = c(0.025, 0.1, 0.5, 0.9, 0.975)
  ) %>%
  pivot_wider(names_from = q, values_from = Yield) %>%
  ggplot(aes(year, `0.5`)) +
  geom_ribbon(aes(ymin = `0.025`, ymax = `0.975`), alpha = 0.4, fill = "dodgerblue3") +
  geom_ribbon(aes(ymin = `0.1`, ymax = `0.9`), alpha = 0.4, fill = "dodgerblue3") +
  geom_line(color = "dodgerblue3", size = 1.2) +
  ylab("fishery yield") +
  theme_bw()
```



Population Size

And here is the implied trend in population size over time, with 80% and 95% compatability bands. How might this look if we were to put a prior distribution on the carrying capacity / initial population size, instead of fixing it at a specific value?

```
fit_sims %>%
  group_by(year) %>%
  summarise(
    N = quantile(N, c(0.025, 0.1, 0.5, 0.9, 0.975)),
    q = c(0.025, 0.1, 0.5, 0.9, 0.975)
  ) %>%
  pivot_wider(names_from = q, values_from = N) %>%
  ggplot(aes(year, `0.5`)) +
  geom_ribbon(aes(ymin = `0.025`, ymax = `0.975`), alpha = 0.4, fill = "dodgerblue3") +
  geom_ribbon(aes(ymin = `0.1`, ymax = `0.9`), alpha = 0.4, fill = "dodgerblue3") +
  geom_line(color = "dodgerblue3", size = 1.2) +
  ylab("population size") +
  theme_bw()
```


population size
2000
1500
1000
500
0
10
20
30
year