

Using Alternative Max Sizes To Create a Mizer Model Using the Methodology Outlined in the Mizer Course

Load in Libraries

```
library(mizer)
library(mizerExperimental)
library(tidyverse)
library(rfishbase)
```

Model 1

Species Parameters for Model 1

Take all parameters from fish base which are possible for those not possible take them from the previous species parameters

```
cel_model_course <- readParams("cel_model_landings.rds")
params<-species_params(cel_model_course)
saveRDS(params, file = "species_params.rds")
params <- readRDS("species_params.rds")

download.file("https://github.com/gustavdelius/mizerCourse/raw/master/build/species_params_spence_et_al.rds",
              destfile = "species_params_spence_et_al.rds")
sp_spence <- readRDS("species_params_spence_et_al.rds")
sp <- select(sp_spence, species)
sp$latin_name <- c("Clupea harengus", # Herring
                  "Sprattus sprattus", # Sprat
```

```

"Gadus morhua", # Cod
"Melanogrammus aeglefinus", # Haddock
"Merlangius merlangus", # Whiting
"Micromesistius poutassou", # Blue whiting
"Trisopterus esmarkii", # Norway Pout
"Trisopterus minutus", # Poor Cod
"Merluccius merluccius", # European Hake
"Lophius piscatorius", # Monkfish
"Trachurus trachurus", # Horse Mackerel
"Scomber scombrus", # Mackerel
"Limanda limanda", # Common Dab
"Pleuronectes platessa", # Plaice
"Lepidorhombus whiffiagonis", # Megrin
"Solea solea", # Sole
"Capros aper") # Boarfish

```

Input size parameters

```

max_size_fishbase <- rfishbase::species(sp$latin_name) |>
  select(latin_name = Species, w_max = Weight, l_max = Length)

```

Joining with `by = join_by(SpecCode)`

```

max_size_fishbase <- left_join(sp, max_size_fishbase, by = "latin_name")
length_weight <- estimate(max_size_fishbase$latin_name, fields = c("Species", "a", "b"))

```

Joining with `by = join_by(SpecCode)`

```

max_size <- max_size_fishbase |>
  left_join(length_weight, by = c("latin_name" = "Species"))
max_size <- max_size |>
  mutate(w_max_calc = a * l_max ^ b)
max_size$w_max[is.na(max_size$w_max)] <- max_size$w_max_calc[is.na(max_size$w_max)]
species_params <- max_size[, -7]

```

Input maturity params including age_mat, l_mat and w_mat (following course)

```

#get estimates on maturity size and maturity age from the maturity table on fish base
maturity_tbl <- rfishbase::maturity(sp$latin_name)

```

Joining with `by = join_by(SpecCode)`

```
#get the median values over all observations where both length at maturity and age at maturity are not NA
median_maturity <- maturity_tbl |>
  group_by(Species) |>
  filter(!is.na(tm), !is.na(Lm)) |>
  summarise(age_mat = median(tm),
            l_mat = median(Lm))
median_maturity
```

```
# A tibble: 17 x 3
  Species          age_mat l_mat
  <chr>          <dbl> <dbl>
1 Capros aper          3.40  8.05
2 Clupea harengus       3    25.3
3 Gadus morhua         6.18 66.5
4 Lepidorhombus whiffiagonis 2.75 24.5
5 Limanda limanda       2.75 24.8
6 Lophius piscatorius    4.5  61
7 Melanogrammus aeglefinus 2.80 35
8 Merlangius merlangus   1.5  24.3
9 Merluccius merluccius  3.25 33.2
10 Micromesistius poutassou 2.15 22.1
11 Pleuronectes platessa  3    27.8
12 Scomber scombrus      1.80 26.2
13 Solea solea           3    26.5
14 Sprattus sprattus      2    9.95
15 Trachurus trachurus    2.5  23.6
16 Trisopterus esmarkii   2.15 16.6
17 Trisopterus minutus    1    13.3
```

```
species_params <- species_params |>
  left_join(median_maturity, by = c("latin_name" = "Species")) |>
  mutate(w_mat = a * l_mat ^ b)
```

get predation kernels

```
species_params <- left_join(species_params, select(sp_spence, species, beta, sigma))
```

Joining with `by = join_by(species)`

get observed biomass and cut off biomass

```
download.file("https://github.com/gustavdelius/mizerCourse/raw/master/build/celtic_sea_ssb.rds",
              destfile = "celtic_sea_ssb.rds")
species_params$biomass_observed <- readRDS("celtic_sea_ssb.rds")

species_params$biomass_cutoff <- species_params$w_mat
species_params >| select(species, biomass_observed, biomass_cutoff)
```

	species	biomass_observed	biomass_cutoff
1	Herring	0.300000000	121.798516
2	Sprat	0.295749801	6.970650
3	Cod	0.008179382	2780.285169
4	Haddock	0.067381049	376.468948
5	Whiting	0.070079361	107.140861
6	Blue whiting	1.188248745	64.779008
7	Norway Pout	0.172520253	32.323777
8	Poor Cod	NA	23.075213
9	European Hake	0.164362236	266.655482
10	Monkfish	0.048720611	2662.619199
11	Horse Mackerel	NA	91.220035
12	Mackerel	NA	145.943498
13	Common Dab	NA	157.089509
14	Plaice	0.022404698	211.647510
15	Megrim	0.074079322	96.920277
16	Sole	0.063519261	169.576150
17	Boarfish	NA	6.687845

```
species_to_update <- c("Poor Cod", "Horse Mackerel", "Mackerel", "Common Dab", "Boarfish")
biomass_course_model <- c(0.007968153, 0.532999576, 0.552924607, 0.021766977, 0.151676286)
rows_to_update <- species_params$species %in% species_to_update
species_params$biomass_observed[rows_to_update] <- biomass_course_model

saveRDS(species_params, "species_params.rds")
```

Load in the gear and species matrix params from spence et al.

```
download.file("https://raw.githubusercontent.com/gustavdelius/mizerCourse/master/build/celtic_gear_params.csv",
              destfile = "celtic_gear_params.csv")
celtic_gear_params <- read.csv("celtic_gear_params.csv")
```

```
download.file("https://raw.githubusercontent.com/gustavdelius/mizerCourse/master/build/celtic_interaction.csv", destfile = "celtic_interaction.csv")
celtic_interaction <- read.csv("celtic_interaction.csv", row.names = 1)
```

Now make the params object into a model

[illegible]

For the following species I will ignore your value for `l_max` because it is not consistent with the data. No `h` provided for some species, so using age at maturity to calculate it.
No `ks` column so calculating from critical feeding level.
Using `z0 = z0pre * w_max ^ z0exp` for missing `z0` values.
Using `f0`, `h`, `lambda`, `kappa` and the predation kernel to calculate `gamma`.

Project to steady state (calibrate match repeat)

```
trial_model2 <- steady(trial_model)
```

Convergence was achieved in 70.5 years.

Warning in `setBevertonHolt(params, reproduction_level = old_reproduction_level)`: The following

```
trial_model3 <- calibrateBiomass(trial_model2)
trial_model4 <- matchBiomasses(trial_model3)
```

Warning in setBevertonHolt(params): For the following species `erepro` has been increased to

```
trial_model5 <- steady(trial_model4)
```

Simulation run did not converge after 99 years. Value returned by the distance function was:

Warning in setBevertonHolt(params, reproduction_level = old_reproduction_level): The following

```
age_mat_model = age_mat(trial_model5)
age_mat_observed = params_trial$age_mat
data.frame(age_mat_model, age_mat_observed)
```

	age_mat_model	age_mat_observed
Herring	3.3489808	3.00
Sprat	2.2080445	2.00
Cod	5.8764531	6.18
Haddock	2.3616745	2.80
Whiting	1.2947979	1.50
Blue whiting	1.7444477	2.15
Norway Pout	1.7573984	2.15
Poor Cod	0.8708668	1.00
European Hake	3.4645994	3.25
Monkfish	4.1678907	4.50
Horse Mackerel	2.0793683	2.50
Mackerel	1.5143913	1.80
Common Dab	2.4002052	2.75
Plaice	2.7415105	3.00
Megrim	2.1778976	2.75
Sole	2.7982122	3.00
Boarfish	3.7804638	3.40

```
trial_model6 <- matchGrowth(trial_model5)
```

Warning in setBevertonHolt(params): For the following species `erepro` has been increased to

```
age_mat_model = age_mat(trial_model6)
data.frame(age_mat_model, age_mat_observed)
```

	age_mat_model	age_mat_observed
Herring	2.9483023	3.00
Sprat	2.0000000	2.00

Cod	4.8459894	6.18
Haddock	2.4939300	2.80
Whiting	1.2251637	1.50
Blue whiting	1.9678138	2.15
Norway Pout	1.8974272	2.15
Poor Cod	0.9441446	1.00
European Hake	2.2200156	3.25
Monkfish	19.5042557	4.50
Horse Mackerel	2.3192021	2.50
Mackerel	1.5854052	1.80
Common Dab	2.4298709	2.75
Plaice	2.5570924	3.00
Megrim	2.4796421	2.75
Sole	2.6977624	3.00
Boarfish	3.3998333	3.40

```
trial_model7 <- steady(trial_model6)
```

Simulation run did not converge after 99 years. Value returned by the distance function was:

Warning in setBevertonHolt(params, reproduction_level = old_reproduction_level): The following

```
age_mat_model = age_mat(trial_model7)
data.frame(age_mat_model, age_mat_observed)
```

	age_mat_model	age_mat_observed
Herring	2.9679955	3.00
Sprat	2.0000006	2.00
Cod	4.3084851	6.18
Haddock	2.5228278	2.80
Whiting	1.2244420	1.50
Blue whiting	2.0094086	2.15
Norway Pout	1.9261565	2.15
Poor Cod	0.9708935	1.00
European Hake	2.0514196	3.25
Monkfish	3.4320961	4.50
Horse Mackerel	2.3732592	2.50
Mackerel	1.6059573	1.80
Common Dab	2.4509999	2.75
Plaice	2.5628707	3.00
Megrim	2.5258769	2.75

Sole	2.7226944	3.00
Boarfish	3.4016111	3.40

```
trial_model8 <- trial_model7 |>
  calibrateBiomass() |> matchBiomasses() |> matchGrowth() |> steady() |>
  calibrateBiomass() |> matchBiomasses() |> matchGrowth() |> steady()
```

Warning in setBevertonHolt(params): For the following species `erepro` has been increased to

Warning in setBevertonHolt(params): For the following species `erepro` has been increased to

Convergence was achieved in 31.5 years.

Warning in setBevertonHolt(params, reproduction_level = old_reproduction_level): The followin

Warning in setBevertonHolt(params): For the following species `erepro` has been increased to

Simulation run did not converge after 99 years. Value returned by the distance function was:

Warning in setBevertonHolt(params, reproduction_level = old_reproduction_level): The followin

```
age_mat_model = age_mat(trial_model8)
data.frame(age_mat_model, age_mat_observed)
```

	age_mat_model	age_mat_observed
Herring	3.217262	3.00
Sprat	2.000094	2.00
Cod	6.375627	6.18
Haddock	3.237184	2.80
Whiting	1.803814	1.50
Blue whiting	2.798079	2.15
Norway Pout	2.798886	2.15
Poor Cod	1.386222	1.00
European Hake	3.636917	3.25
Monkfish	4.423086	4.50
Horse Mackerel	3.277534	2.50
Mackerel	2.213629	1.80
Common Dab	3.231504	2.75
Plaice	3.471072	3.00
Megrim	3.537596	2.75
Sole	3.455920	3.00
Boarfish	3.447863	3.40


```
saveParams(trial_model8, "trial_model.rds")
```

Model is unable to reach a steady state using the same methods outlined in the course, erepro values for species are required to be set at a value greater than 1, and simulation does not converge after 99 years.