Cost-effectiveness of stunning intervention

Sagar Shah, Rethink Priorities

2024-02-06

Generic prep

Open libraries

```
rm(list=ls())
library(tidyverse)
options(dplyr.summarise.inform = FALSE)
library(scales)
library(rlang)
library(gt)
library(RColorBrewer)
library(forcats)
library(magrittr)
library(webshot2)
```

Run code with assumptions

```
#Useful monte carlo functions
source("0a Useful functions.R")
#Take on board assupmtions for seabream, seabass and trout
source("0b Assumptions for seabream,seabass and trout.R")
#Modelling assumptions
source("0c  Modelling assumptions.R")
```

Helper functions

Write helper functions to call object/vectors

```
consumption <- function(country,species)  obtain(species,country,"tons")
mshare <- function(scenario,country) obtain("mshare",scenario,country)
psuccess <- function(scenario) obtain("psuccess",scenario)
weight <- function(species) obtain("weight",species)
slaughter_minutes <- function(species) obtain("slaughter_minutes",species)
stun_share <- function(species) obtain(species,"stunned")
cost <- function(scenario) obtain("cost",scenario)
```

Calculations

Impact per dollar

Core calculation function

```
# Number of fish affected per year
no_fish_affected_per_year <- function(country,species,scenario) {
consumption(country,species)*1E3/
  weight(species)*
  (1-stun_share(species))*
  mshare(scenario,country)*
  psuccess(scenario)*
  implementation_discount*
  fish_grocery
}

# Fish affected across all years of credit
no_fish_affected <- function(country,species,scenario) {
no_fish_affected_per_year(country,species,scenario)*years_credit
}

# Fish minutes affected
fish_hours_affected <- function(country,species,scenario) {
no_fish_affected(country,species,scenario)*slaughter_minutes(species)/60
}
```

Functions to apply calculations across countries, species and scenarios

I now produce a table that produces a possible combination of country and species

```
no_of_countries <- length(country_list)
combo <- expand.grid(country=country_list,species=species_list) %>%
  mutate(country_species=str_c(country,"_",species))
```

I now write a function that:

- Works out the total number of fish (or fish minutes) affected in a given scenario, country and species
- Sums the result for all species and countries
- Divides by the total program costs for a specific scenario

```
output_per_dollar <- function(fish_function,program) {

temp_output<- map2(                 # Allows repeated calcs across different input vectors
                combo$country,      # Input vector 1: countries to run function over
                combo$species,      # Input vector 2: species to run function over
                get(fish_function), # Calculation function (no of fish or minutes)
                scenario=program    # Scenario (pilot or scale)
                  ) %>%
         as.data.frame() %>%       # Convert to data.frame
         mutate(
           total=rowSums(.))

return(
  temp_output$total/(no_of_countries*cost(program))
)
}
```

Generate table for combinations of output and scenario

```
results_combination <- expand_grid(
  measure=c("fish_hours_affected","no_fish_affected","no_fish_affected_per_year"),
  program=c("pilot","scale")) %>%
  mutate(description=str_c(program,"___",measure))
```

Run calculations

I now write a function that works out the number of fish minutes affected and total number of fish affected per dollar across both the pilot and scale scenarios. The results of the montecarlo simulation are in a single table called "impact_per_dollar".

```
impact_per_dollar <- map2(                          # Function to iterate over vectors
                      results_combination$measure,  # Vector of different measures
                      results_combination$program,  # Vector of different scenarios
                      output_per_dollar) %>%        # Function to repeat
                      as.data.frame()

# Assign names to data.frame
names(impact_per_dollar) <- results_combination$description

# Convert to long format
impact_per_dollar %<>%
  pivot_longer(
    cols=1:nrow(results_combination),
    names_to="description",
    values_to="number"
) %>%
  separate_wider_delim(description,delim="___",names=c("scenario","measure"))
```

$/DALY "welfare range" calculations

Prep

I first produce estimates of fish years affected per dollar in both scenarios.

```
fypd_pilot <- output_per_dollar("fish_hours_affected","pilot")*hours_to_years
fypd_scale <- output_per_dollar("fish_hours_affected","scale")*hours_to_years
fypd <- function(scenario) obtain("fypd",scenario)
```

Core calculation functions

```
wr_dollar_per_daly <- function(scenario) {
   1/(
    fypd(scenario)*
    salmon_wr*
    duration_share*
    fish_welfarerange_impact_stun/
    DALY_share_of_welfare_range)
}
```

Apply calculations

Calculate results and place into data frame

```
results_dollar_per_daly <-
tibble(
          # core calculations
  pilot=wr_dollar_per_daly("pilot"),
  scale=wr_dollar_per_daly("scale")) %>%
          # pivot to long format
  pivot_longer(
    cols=1:2,
    values_to="dollars_per_daly",
    names_to="scenario"
  ) %>%
          # reformat for table production
  mutate(scenario=str_to_sentence(scenario))
```

Share of simulations beating $/DALY benchmark

Core calculation functions

```
#Moral value approach
share_mv <- function(xvar,scenario,bar) {
  mean(bar>(1/(fypd(scenario)*xvar)))
}

# Welfare range approach
share_wr <- function(xvar,scenario,bar) {
  mean(bar>
      (1/
        (xvar*
        fypd(scenario)*
        salmon_wr/
        DALY_share_of_welfare_range
        )))
}
```

Define parameters over which to calculate results. Definition of xvar provided in the table below.

Approach	Meaning of ‘xvar’ variable means in context of calculations
Moral Value	Moral value of improving fish welfare for 1 year through intervention relative to gaining 1 DALY
Welfare Range	Welfare gain from intervention - expressed as % of entire fish welfare range (negative to positive)

Run calculations

```
mv_table_values <- c(0.01,0.1,0.25,0.5,0.75,1,5,10,25,50,75,100,500,1000,5000)
wr_table_values <- c(0.01,0.05,0.1,0.25,0.5,0.75,0.9,0.95,0.99,1)
chart_values<- 10^seq(-4,4,0.01)
xvar_values <- c(chart_values,mv_table_values,wr_table_values)
scenario_list <- c("pilot","scale")
```

Calculate results

```
share_sims_results_table <-
# Generate one row for unique combination of xvar_values and scenario lists
  expand_grid(
    xvar=xvar_values,
    scenario=scenario_list,
    bar=bar_values) %>%
# Core calcualtions using share_mv and share_wr calculation functions
    rowwise() %>%
    mutate(
      moral_value=share_mv(xvar,scenario,bar),
      welfare_range=share_wr(xvar,scenario,bar)
    ) %>%
# Convert pivot into longer format to make results easier to plot
  pivot_longer(
    cols=c(moral_value,welfare_range),
    names_to="approach",
    values_to="share_of_simulations") %>%
# Change descriptions of variables to make results easier to interpert
  mutate(
    scenario=str_to_sentence(scenario),
    approach=str_to_sentence(str_replace_all(approach,"_"," ")),
    bar_factor=
      factor(bar,levels=bar_values,labels=c("50","1K","70K")),
    bar_description=case_when(
```

```
    bar==50 ~ "50 - Open Philanthropy GHW bar",
    bar==1000 ~ "1K - Low HDI country HE av.",
    bar==70000 ~ "70K - Very high HDI country HE av.",
    TRUE ~ "ERROR")
  )
```

Results tables/charts

Impact per dollar

```
# Code to create impact per dollar density plot

impact_density_plot <- function(desired_filter,description) {
  impact_per_dollar %>%
  filter(measure==desired_filter) %>%
  ggplot(
    aes(
      x=number,
      color=str_to_sentence(scenario)
    )
  ) +
  geom_density() +
  theme_light() +
  scale_x_log10(labels = scales::comma_format(drop0trailing = TRUE),n.breaks=8) +
    labs(
    title=description,
    y="Probability density",
    x=paste0(description," (log scale)"),
    color="Scenario")
}

# Code to create impact per dollar summary table

impact_table <- function(desired_filter,description) {
impact_per_dollar %>%
  filter(measure==desired_filter) %>%
  mutate(scenario=str_to_sentence(scenario)) %>%
  group_by(scenario) %>%
  summarise(
```

```
    Mean=mean(number),
    p5=quantile(number,0.05),
    Median=median(number),
    p95=quantile(number,0.95)
  ) %>%
  pivot_longer(cols=2:5,names_to="Statistic",values_to="Value") %>%
  pivot_wider(names_from = "scenario",values_from = "Value") %>%
  gt() %>%
  fmt_number(decimals = 1) %>%
  tab_header(
  title = description) %>%
  tab_spanner(
    label = "Scenario",
    columns = c(Pilot,Scale))
}

# Generate results

fig9_fish_pd <- impact_density_plot("no_fish_affected","Number of fish affected per dollar
tab8_fish_pd<- impact_table("no_fish_affected","Number of fish affected per dollar")
tab9_fish_pdpy <- impact_table("no_fish_affected_per_year","Number of fish affected per do
fig10_fish_hours_pd <- impact_density_plot("fish_hours_affected","Fish hours affected per
tab10_fish_hours_pd <- impact_table("fish_hours_affected","Fish hours affected per dollar"
```

$/DALY Range

Density plot

```
fig11_dolperdaly_density <-
  results_dollar_per_daly %>%
  ggplot(
    aes(
      x=dollars_per_daly,
      color=scenario
    )
  ) +
  geom_density() +
  theme_light() +
  scale_x_log10(
    breaks=10^seq(0,15,1),
```

```
    labels = scales::label_number(scale_cut = scales::cut_short_scale())) +
    labs(
    title = "$/DALY Density Plot",
    y="Probability density",
    x="$/DALY (log-scale)",
    color="Scenario")
```

Summary stats table

```
tab11_dolperdaly_density <-
results_dollar_per_daly %>%
  group_by(scenario) %>%
  summarise(
    Mean=mean(dollars_per_daly),
    p1=quantile(dollars_per_daly,0.01),
    p5=quantile(dollars_per_daly,0.05),
    p10=quantile(dollars_per_daly,0.10),
    p25=quantile(dollars_per_daly,0.25),
    Median=median(dollars_per_daly),
    p75=quantile(dollars_per_daly,0.75),
    p90=quantile(dollars_per_daly,0.90),
    p95=quantile(dollars_per_daly,0.95),
    p99=quantile(dollars_per_daly,0.99)
  ) %>%
  pivot_longer(cols=2:11,names_to="Statistic",values_to="$/DALY") %>%
  pivot_wider(names_from = "scenario",values_from = "$/DALY") %>%
  gt() %>%
  tab_spanner(
    label = "Scenario",
    columns = c(Pilot,Scale)) %>%
  fmt_number(columns=2:3,suffixing=TRUE,n_sigfig =3)
```

Share of simulations beating benchmark

Chart

```
share_sims_chart <- function (
                        chart_approach,
                        xvar_lower_lim=0,
```

```
                          xvar_upper_lim=max(xvar_values),
                          xlab_des
                          ) {

share_sims_results_table %>%
    filter(
    approach==chart_approach,
    xvar<=xvar_upper_lim,
    xvar>=xvar_lower_lim
    ) %>%
  ggplot(
    aes(
      x=xvar,
      y=share_of_simulations,
      color=as.factor(scenario),
      linetype=bar_factor))+
  geom_line(lwd=0.8)+
  scale_linetype_manual(values=c("solid","dotted","twodash"))+
  scale_color_brewer(palette = "Dark2")+
  theme_light()+
  scale_y_continuous(limits=c(0,1),n.breaks=10,labels = scales::percent_format(accuracy =
  labs(
    subtitle = str_to_title(paste0(chart_approach," approach")),
    title="Share of simulations where intervention beats $/DALY benchmark",
    y="Share of simulations",
    x=xlab_des,
    linetype="$/DALY Benchmark",
    color="Intervention",
    caption= "Note log scale on x-axis")

}


xlab_wr="
    Welfare gain from intervention as a % of total fish welfare range"

xlab_mv="Moral value of improving a fish-life year via intervention
relative to gaining a human DALY"
```

Produce share of simulations charts

```r
fig12_wr_benchmark <-share_sims_chart("Welfare range",xvar_upper_lim=1,xlab_des=xlab_wr)+
  scale_x_log10(limits=c(0.001,1),labels = scales::percent_format(drop0trailing = TRUE),n.
  scale_y_continuous(limits=c(0,0.6),n.breaks=10,labels = scales::percent_format(accuracy
```

```
Scale for y is already present.
Adding another scale for y, which will replace the existing scale.
```

```r
fig13_mv_benchmark <- share_sims_chart("Moral value",xvar_lower_lim=0.003,xlab_des=xlab_mv
  scale_x_log10(labels = scales::label_number_si(drop0trailing = TRUE),n.breaks=11)
```

```
Warning: `label_number_si()` was deprecated in scales 1.2.0.
i Please use the `scale_cut` argument of `label_number()` instead.
```

Table

Share of simulations at various welfare range / moral value assumptions.

```r
share_beating_bar_table <- function(values,chart_approach) {
share_sims_results_table %>%
  filter(
      xvar %in% values,
      approach==chart_approach) %>%
  select(-approach) %>%
  distinct() %>%
  mutate(
    scenario_bar=paste(scenario,bar,sep="_")) %>%
  pivot_wider(id_cols=xvar,names_from=scenario_bar,values_from=share_of_simulations) %>%
  arrange(xvar) %>%
  gt() %>%
  tab_header(title = "Share of simulations beating $DALY bar") %>%
  cols_label(xvar=chart_approach)
}

tab12_wr_benchmark <- share_beating_bar_table(wr_table_values,"Welfare range") %>%
    fmt_percent(columns=1,decimals=0) %>%
    fmt_percent(columns=2:7,decimals=1)


tab13_mv_benchmark <- share_beating_bar_table(mv_table_values,"Moral value") %>%
```

```
  fmt_percent(columns=2:7,decimals=1)
```

Minimum moral value needed to achieve given share of simulations to beat $/DALY bar.

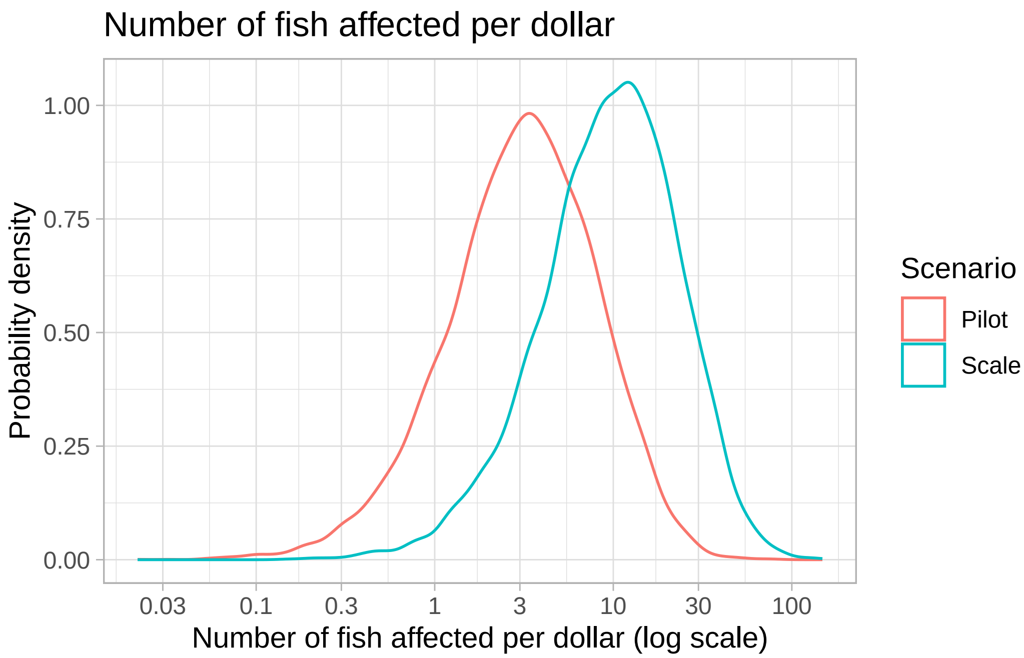
```
sim_cuts<- c(0,0.01,0.1,0.25,0.5,0.75,0.90,0.95,0.99,1,1.01)

tab14_min_mv <- share_sims_results_table %>%
filter(approach=="Moral value") %>%
  mutate(
    sims_bin=
      cut(share_of_simulations,breaks=sim_cuts,right=FALSE,labels=head(sim_cuts,-1))) %>%
  group_by(scenario,bar,sims_bin) %>%
  summarise(min_moral_value=min(xvar)) %>%
  mutate(scenario_bar=str_c(scenario,bar,sep="_")) %>%
  pivot_wider(id_cols=sims_bin,names_from=scenario_bar,values_from=min_moral_value) %>%
  mutate(sims_bin=head(sim_cuts,-1)) %>%
  gt() %>%
  fmt_percent(columns=1,decimals=0) %>%
  fmt_number(columns=2:7,decimals=1) %>%
  tab_header(title = "Minimum moral value assumption needed for to achieve share of simula
```

Print charts and tables

Impact per dollar charts and tables

```
fig9_fish_pd
```




```
tab8_fish_pd
```

Number of fish affected per dollar

	Scenario	
Statistic	Pilot	Scale
Mean	4.6	13.3
p5	0.6	1.9
Median	3.2	9.9
p95	13.5	36.0

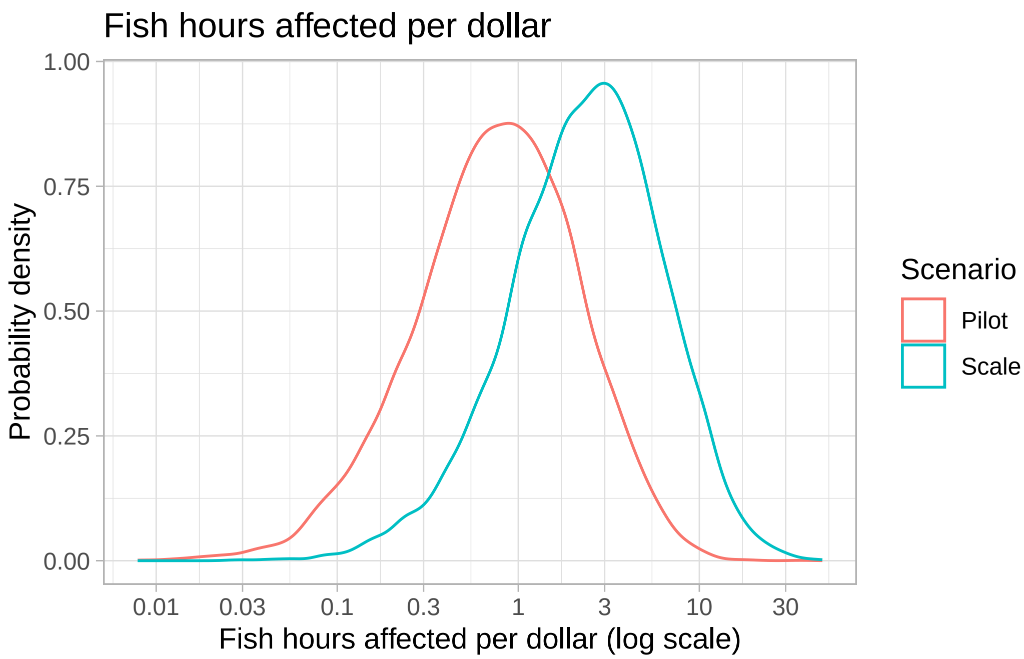
```
tab9_fish_pdpy
```

Number of fish affected per dollar per year

	Scenario	
Statistic	Pilot	Scale
Mean	0.4	1.2
p5	0.1	0.2
Median	0.3	1.0

p95	1.1	2.8

```
fig10_fish_hours_pd
```



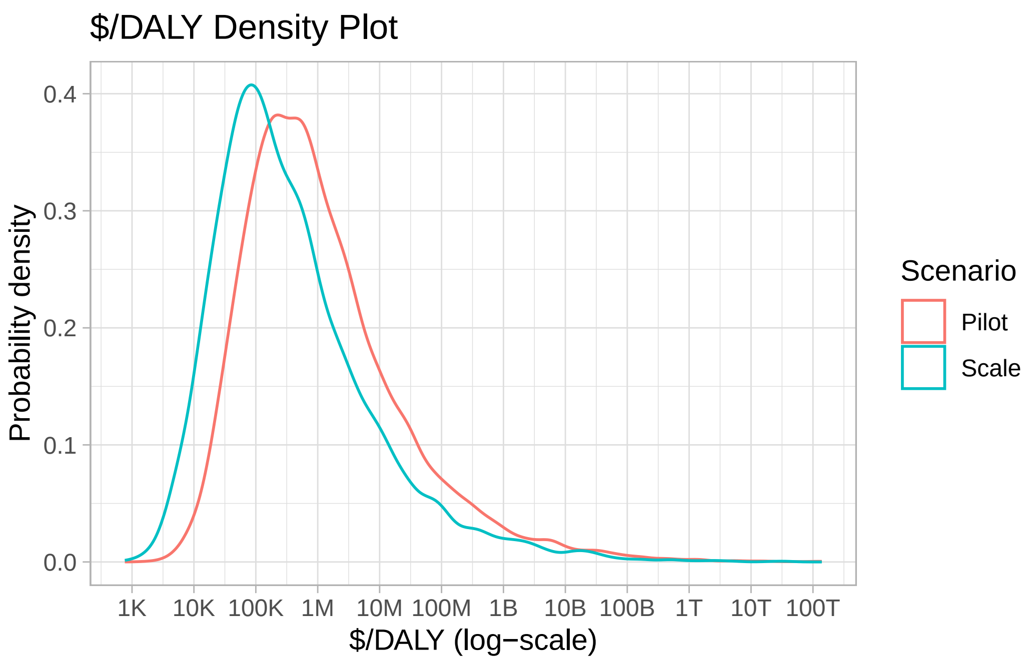

```
tab10_fish_hours_pd
```

Fish hours affected per dollar

	Scenario	
Statistic	Pilot	Scale
Mean	1.2	3.6
p5	0.1	0.4
Median	0.8	2.5
p95	3.8	10.4

Dollar per daly distribution

```
fig11_dolperdaly_density
```



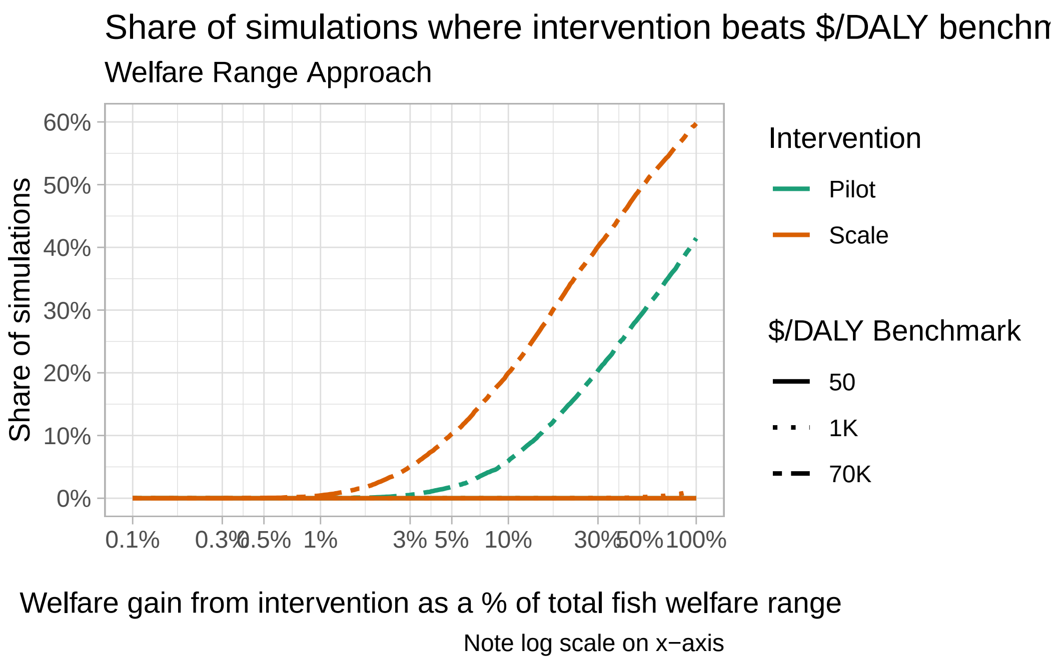

```
tab11_dolperdaly_density
```

	Scenario	
Statistic	Pilot	Scale
Mean	$41.1B$	$12.3B$
p1	$11.8K$	$4.11K$
p5	$29.9K$	$10.4K$
p10	$52.3K$	$17.8K$
p25	$151K$	$51.0K$
Median	$680K$	$216K$
p75	$4.92M$	$1.61M$
p90	$61.0M$	$18.1M$
p95	$366M$	$114M$
p99	$26.9B$	$9.98B$

Welfare range benchmarks

```
fig12_wr_benchmark
```

```
Warning: Removed 600 rows containing missing values (`geom_line()`).
```



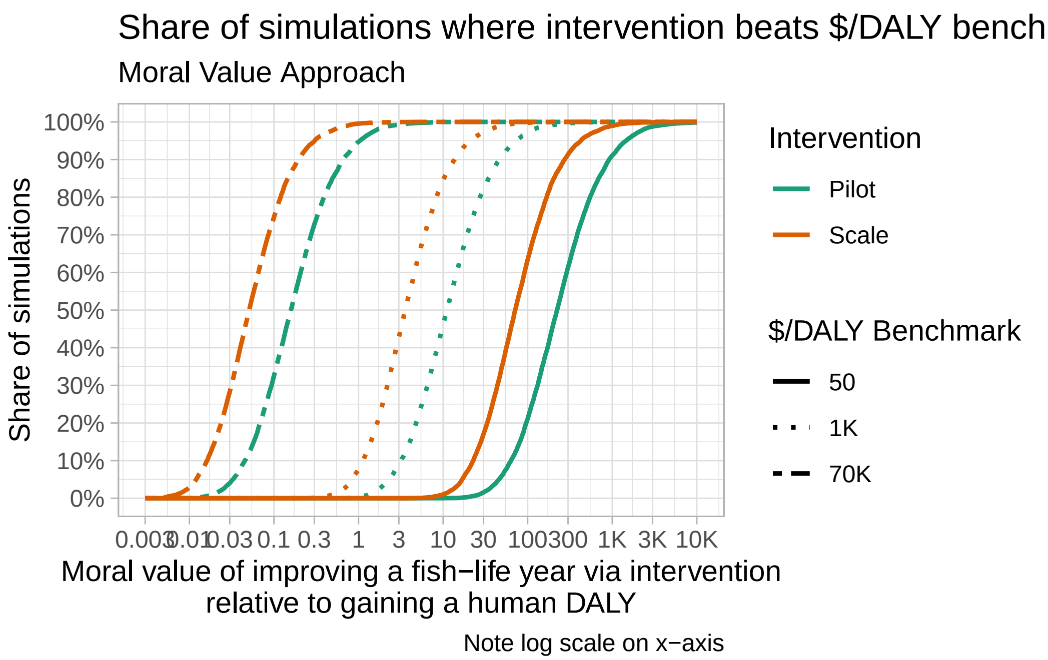

```
tab12_wr_benchmark
```

Share of simulations beating $DALY bar

Welfare range	Pilot_50	Pilot_1000	Pilot_70000	Scale_50	Scale_1000	Scale_70000
1%	0.0%	0.0%	0.0%	0.0%	0.0%	0.4%
5%	0.0%	0.0%	1.8%	0.0%	0.0%	10.3%
10%	0.0%	0.0%	5.9%	0.0%	0.0%	20.0%
25%	0.0%	0.0%	17.5%	0.0%	0.0%	37.0%
50%	0.0%	0.0%	29.0%	0.0%	0.2%	49.2%
75%	0.0%	0.0%	36.1%	0.0%	0.5%	55.5%
90%	0.0%	0.0%	39.4%	0.0%	0.9%	58.2%
95%	0.0%	0.0%	40.4%	0.0%	1.0%	59.1%
99%	0.0%	0.0%	41.2%	0.0%	1.1%	59.6%
100%	0.0%	0.0%	41.4%	0.0%	1.2%	59.8%

Moral value benchmarks

```
fig13_mv_benchmark
```




```
tab13_mv_benchmark
```

Share of simulations beating $DALY bar

Moral value	Pilot_50	Pilot_1000	Pilot_70000	Scale_50	Scale_1000	Scale_70000
0.01	0.0%	0.0%	0.1%	0.0%	0.0%	2.9%
0.10	0.0%	0.0%	32.6%	0.0%	0.0%	74.7%
0.25	0.0%	0.0%	66.8%	0.0%	0.1%	93.3%
0.50	0.0%	0.0%	85.3%	0.0%	1.0%	97.8%
0.75	0.0%	0.1%	91.5%	0.0%	3.5%	99.0%
1.00	0.0%	0.4%	94.7%	0.0%	7.6%	99.6%
5.00	0.0%	21.1%	99.7%	0.1%	63.4%	100.0%
10.00	0.0%	46.1%	100.0%	1.0%	84.6%	100.0%
25.00	0.8%	77.7%	100.0%	12.2%	96.3%	100.0%
50.00	6.1%	91.1%	100.0%	35.3%	98.9%	100.0%
75.00	13.2%	95.3%	100.0%	52.1%	99.6%	100.0%
100.00	21.1%	97.2%	100.0%	63.4%	99.8%	100.0%
500.00	77.7%	99.9%	100.0%	96.3%	100.0%	100.0%
1000.00	91.1%	100.0%	100.0%	98.9%	100.0%	100.0%
5000.00	99.5%	100.0%	100.0%	100.0%	100.0%	100.0%

```
tab14_min_mv
```

Minimum moral value assumption needed for to achieve share of simulations beating bar

sims_bin	Pilot_50	Pilot_1000	Pilot_70000	Scale_50	Scale_1000	Scale_70000
0%	0.0	0.0	0.0	0.0	0.0	0.0
1%	26.9	1.3	0.0	10.0	0.5	0.0
10%	64.6	3.2	0.0	22.9	1.1	0.0
25%	114.8	5.6	0.1	38.9	1.9	0.0
50%	223.9	11.2	0.2	72.4	3.6	0.1
75%	457.1	22.9	0.3	144.5	7.1	0.1
90%	933.3	46.8	0.7	275.4	13.8	0.2
95%	1,479.1	74.1	1.0	426.6	21.4	0.3
99%	3,467.4	173.8	2.5	1,023.3	51.3	0.7
100%	NA	1,122.0	16.2	6,456.5	323.6	4.7

Export charts and tables

```
save_chart("fig9_fish_pd.png",fig9_fish_pd)
save_chart("fig10_fish_hours_pd.png",fig10_fish_hours_pd)
save_chart("fig11_dolperdaly_density.png",fig11_dolperdaly_density)
save_chart("fig12_wr_benchmark.png",fig12_wr_benchmark)
```

```
Warning: Removed 600 rows containing missing values (`geom_line()`).
```

```
save_chart("fig13_mv_benchmark.png",fig13_mv_benchmark)
save_chart("fig13_mv_benchmark.png",fig13_mv_benchmark)



save_table(tab8_fish_pd,"tab8_fish_pd.html")
save_table(tab9_fish_pdpy,"tab9_fish_pdpy.html")
save_table(tab10_fish_hours_pd,"tab10_fish_hours_pd.html")
save_table(tab11_dolperdaly_density,"tab11_dolperdaly_density.html")
save_table(tab12_wr_benchmark,"tab12_wr_benchmark.html")
save_table(tab13_mv_benchmark ,"tab13_mv_benchmark.html")
save_table(tab14_min_mv,"tab14_min_mv.html")
```

```
saveRDS(fypd_scale,file= "../3_intermediate_data/fypd_scale.rds")
```

Additional analyses

Sensitivity to DALY as a share of human welfare range assumption

This section examines how the share of simulations beating various benchmarks under an assumption that an intervention that improves welfare by 10% of the human welfare range for a year is equivalent to averting a DALY, rather than 50%.

Calculation function

```
share_wr_alt <- function(xvar,scenario,bar) {
  mean(bar>
      (1/
        (xvar*
        fypd(scenario)*
        salmon_wr_alt/
        DALY_share_of_welfare_range_alt
        )))
}
```

Generate output table

```
# Generate one row for unique combination of xvar_values and scenario lists

alt_output_table <- expand_grid(
    xvar=wr_table_values,
    scenario="scale",
    bar=bar_values) %>%
# Core calcualtions using share_mv and share_wr calculation functions
    rowwise() %>%
    mutate(
      base=share_wr(xvar,scenario,bar),
      alt=share_wr_alt(xvar,scenario,bar)
    ) %>%
   select(-scenario) %>%
   pivot_longer(
     cols=c(base,alt),
     names_to="wr_assumptions",
```

```
      values_to="share_of_sims") %>%
  mutate(
      bar_factor=
      factor(bar,levels=bar_values,labels=c("50","1K","70K"))) %>%
  pivot_wider(
    id_cols=xvar,
    names_from=c(wr_assumptions,bar_factor),
    values_from=share_of_sims) %>%
  gt() %>%
  fmt_percent(decimals =1) %>%
  fmt_percent(columns=xvar,decimals =0) %>%
    cols_label(
    xvar = "Welfare range impact")
```

Output table is printed below. Overall conclusions do not appear overly sensitive to this assumption. If the stunning intervention improves fish welfare by an average 50% of the fish welfare range for the entire (non-stunning) slaughter duration, it will beat a benchmark of $1K/DALY in only 6.6% of simulations.

```
alt_output_table
```

Welfare range impact	base_50	alt_50	base_1K	alt_1K	base_70K	alt_70K
1%	0.0%	0.0%	0.0%	0.0%	0.4%	10.3%
5%	0.0%	0.0%	0.0%	0.0%	10.3%	37.0%
10%	0.0%	0.0%	0.0%	0.2%	20.0%	49.2%
25%	0.0%	0.0%	0.0%	1.9%	37.0%	62.6%
50%	0.0%	0.0%	0.2%	6.6%	49.2%	71.1%
75%	0.0%	0.0%	0.5%	10.9%	55.5%	75.0%
90%	0.0%	0.0%	0.9%	13.3%	58.2%	76.6%
95%	0.0%	0.0%	1.0%	14.1%	59.1%	77.0%
99%	0.0%	0.0%	1.1%	14.8%	59.6%	77.2%
100%	0.0%	0.0%	1.2%	14.9%	59.8%	77.3%

How might animals affected per dollar from the fish stunning intervention compare to marginal chicken grants

I this section I perform a quick BOTEC to see how the animals affected per dollar from the fish stunning intervention in the scale scenario might compare to marginal chicken grants.

I first import raw sample data (5000 simulations) for chickens affected per dollar from the Guesstimate model from Šimčikas (2019).

```
# Import data on chickens affected per dollar from Guesstimate samples
chickens_per_dollar_historic <-
  read_csv("../1_input_data/chickens_per_dollar.csv",show_col_types=FALSE) %>%
  pull(chickens_per_dollar) %>%
  sample(sims,replace=TRUE)
```

I then work estimate marginal chickens affected per dollar, conditional on an assumption that marginal grants may impact 3.5 to 5 times fewer animals per dollar, then historic ones.

```
# Assumption about guesstimate samples
decrease_in_chicken_effectiveness <- runi(3.5,5)
chickens_per_dollar_marginal <- chickens_per_dollar_historic/decrease_in_chicken_effective
```

I then work out the share of simulations where marginal chicken grants affect more animals per dollar then fish stunning grants in the scale scenario. About half the time.

```
# Assumption about guesstimate samples
mean(chickens_per_dollar_marginal>output_per_dollar("no_fish_affected","scale"))
```

```
[1] 0.4926
```

I then look at how the overall distributions compare.

```
summarystats(chickens_per_dollar_marginal,output_per_dollar("no_fish_affected","scale"))
```

statistic	chickens_per_dollar_marginal	output_per_dollar(”no_fish_affected”, ”scale”)
mean	15.0	13.3
sd	17.7	11.8
median	9.13	9.90
p5	2.14	1.89
p10	2.92	2.89
p25	4.96	5.37
p75	18.1	17.4
p90	33.6	27.9
p95	47.3	36.0