Time in pain - fish stunning vs historic chicken interventions

Sagar Shah, Rethink Priorities

2024-02-06

This file explores the extent to which the fish stunning intervention in the scale scenario might avert more time in various pain intensities than historic estimates for chicken campaigns.

Conclusion is that it is unlikely that the fish stunning intervention can have an impact that comes close to the impact of chicken campaigns for annoying, hurtful or disabling pain. The direction of the impact here is entirely driven by uncertainty of whether the chicken intervention increases or decreases time in pain.

But for excruciating pain, its plausible that the fish stunning intervention could reduce more time in pain if at least 10-15% of the duration of conventional fish slaughter for affected species is excruciating pain, and most of this is mitigated through stunning. Further research may provide evidence on the extent to which this might be the case.

Nevertheless, it seems that fish stunning has the potential to reduce more time in excruciating pain than historic chicken campaigns, and this may make the intervention appealing for those concerned about reducing time spent in the most intense types of pain.

General prep

Open packages

```
rm(list=ls())
library(tidyverse)
```

```
-- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
v dplyr     1.1.2     v readr     2.1.4
v forcats   1.0.0     v stringr   1.5.0
v ggplot2   3.4.2     v tibble    3.2.1
```

```
v lubridate 1.9.2     v tidyr     1.3.0
v purrr     1.0.1
-- 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 becor
```

```
library(scales)
```

```
Attaching package: 'scales'

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

    discard

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

    col_factor
```

```
library(rlang)
```

```
Attaching package: 'rlang'

The following objects are masked from 'package:purrr':

    %@%, flatten, flatten_chr, flatten_dbl, flatten_int, flatten_lgl,
    flatten_raw, invoke, splice
```

```
library(gt)
library(RColorBrewer)
library(forcats)
library(readxl)
library(ggridges)
library(forcats)
```

Import data from welfare footprint and estimates of fish impact per dollar

```r
# Import data
wfpt <- read_excel("../1_input_data/welfarefootprintdatachickens.xlsx")
fypd_scale <- readRDS("../3_intermediate_data/fypd_scale.rds")
source("0a Useful functions.R")
```

Clean and process welfare footprint data

I now clean and process welfare footprint data into a format that enables comparisons with the fish cost-effectiveness estimates. I generate two main data frames.

The first contains a montecarlo simulation of the change in duration of time in each pain intensity as a result of transitioning from conventional broilers and conventional/furnished layers to reformed status, assuming the durations provided by the welfare footprint project are normally distrbuted. Given overlapping confidence intervals, many of the transitions result in increased duration of pain in a given intensity for a reasonably large share of simulations.

I then produce an alternative table where I assume the change in duration for each welfare transition is equal to the *difference in means*.

Clean welfare footprint data

```r
# Change collumn titles to lower case and remove spaces
names(wfpt)[[6]] <- "mean"
names(wfpt) <-   str_to_lower(names(wfpt)) %>%
  str_replace_all(" ","_")

# Change data into easier to process names
wfpt %<>%
  mutate(
    animal=case_when (
      str_detect(animal,"Broiler") ~ "broiler",
      str_detect(animal,"Laying") ~ "layer",
      TRUE ~ "ERROR") ,
    scenario=case_when(
      str_detect(scenario,"Conventional") ~ "conventional",
      str_detect(scenario,"Furnished") ~ "furnished",
      TRUE ~ "reformed"),
    intensity=str_to_lower(intensity)
  ) %>%
  select(-challenge)
```

Define time_weights to convert seconds into hours

```
time_weights <- data.frame(
  `time_unit` = c("seconds","hour"),
  duration = c(1/(60*60),1)
)
```

Work out standard deviation of welfare footprint duration estimates, assuming normally distributed data and 90% CIs and convert all duration estimates into hours.

```
wfpt<- wfpt %>%
#Estimate standard deviation
  mutate(
    sd=(upper_bound-lower_bound)/(2*1.645)) %>%
# Add duration conversion data
 left_join(time_weights,by="time_unit") %>%
# Create new columns for duration in hours
   mutate(
   mean_hours=mean*duration,
   sd_hours=mean*duration)
```

Create montecarlo simulation of time in pain for chickens given mean, standard deviation and assumption of normally distribtued data

```
# Produce list of vectors
wfpt_list <- map2(wfpt$mean_hours,wfpt$sd_hours,rnorm,n=sims)
# Assign names
names(wfpt_list)<- str_c(wfpt$animal,wfpt$scenario,wfpt$intensity,sep="_")
# Convert to data frame
wfpt_list <- as.data.frame(wfpt_list)
```

Work out change in time in each pain intensity from transitioning from conventional/furnished to reformed

```
# Define names of transitions we want to keep
chicken_transitions <- expand_grid(
      start=c("broiler_","layer_conv_","layer_furn_"),
      end=c("excruciating","disabling","hurtful","annoying")) %>%
    mutate(name=str_c(start,end)) %>%
    select(name)

# Calculate time change
chicken_benefits_full <- wfpt_list %>%
  mutate(
```

```
    broiler_excruciating=broiler_conventional_excruciating-broiler_reformed_excruciating,
    broiler_disabling=broiler_conventional_disabling-broiler_reformed_disabling,
    broiler_hurtful=broiler_conventional_hurtful-broiler_reformed_hurtful,
    broiler_annoying=broiler_conventional_annoying-broiler_reformed_annoying,
    layer_conv_excruciating=layer_conventional_excruciating-layer_reformed_excruciating,
    layer_conv_disabling=layer_conventional_disabling-layer_reformed_disabling,
    layer_conv_hurtful=layer_conventional_hurtful-layer_reformed_hurtful,
    layer_conv_annoying=layer_conventional_annoying-layer_reformed_annoying,
    layer_furn_excruciating=layer_furnished_excruciating-layer_reformed_excruciating,
    layer_furn_disabling=layer_furnished_disabling-layer_reformed_disabling,
    layer_furn_hurtful=layer_furnished_hurtful-layer_reformed_hurtful,
    layer_furn_annoying=layer_furnished_annoying-layer_reformed_annoying) %>%
  # Keep only the transitions we need
    select(chicken_transitions$name)
```

Work out change in time in each pain intensity, assuming that each chicken experiences the *mean* estimated time in each pain intensity.

```
chicken_benefits_mean <- wfpt %>%
# Select only columns needed
  select(animal,scenario,intensity,mean_hours) %>%
# Convert into wider format to facilitate calculations
    pivot_wider(id_cols=c(animal,intensity),names_from=scenario,values_from=mean_hours) %>
# Run calculations
  mutate(
    conv=conventional-reformed,
    furn=furnished-reformed) %>%
# Remove columns no longer needed
  select(-c(reformed,conventional,furnished)) %>%
# Convert back into long format
    pivot_longer(cols=c(conv,furn),names_to="reform",values_to="mean_hours") %>%
# Remove transitions where calculations generated an error (e.g. broiler furnished)
  filter(!is.na(mean_hours)) %>%
# Generate names that will align with the full montecarlo estimates
    mutate(
    intervention=str_c(animal,reform,intensity,sep="_"),
    intervention=str_replace(intervention,"broiler_conv_","broiler_")
  ) %>%
# Select only columns needed
  select(-c(animal,intensity,reform)) %>%
# Convert into wider format (to match full montecarlo data frame)
```

```
  pivot_wider(names_from=intervention,values_from=mean_hours) %>%
# Add 'sims' rows of data (again to match full montecarlo data frame)
  slice(rep(1,each=sims))
```

Estimating chicken duration in pain averted per dollar

I now use Šimčikas (2019)'s estimates of chickens affected per dollar to work out the duration of pain in each intensity averted per dollar spent on historical chicken corporate commitment campaigns.

I first define functions to describe the number of layers and broilers affected per dollar. And then make an assumption about the share of layer hens affected by campaigns that would transition from furnished cages. This is based on Šimčikas (2019)'s guesstimate values for the number of chickens affected in non-US countries, an assumption that most of the chickens affected by non-US commitments are EU countries where cages are typically "enriched".

```
historic_chicken_results <- read_csv("../1_input_data/chickens_per_dollar.csv",show_col_ty

extr <- function(x) (historic_chicken_results[[x]] %>% sample(sims,replace=TRUE))

broiler_pd<- extr("broilers_pd")
layer_pd<- extr("layer_hens_pd")
us_hens <- extr("us_cage_free_commitments")
other_hens <- extr("other_cage_free_commitments")
furnished_layer_share_non_us<-rbeta_ci(0.5,0.9)
furnished_layer_share<- furnished_layer_share_non_us*other_hens/(us_hens+other_hens)
```

I then write some helper functions to convert the duration in pain estimates per chicken, to duration in pain per dollar.

```
# Define adjustment functions for each type of change
fbroiler <- function(x) (x*broiler_pd)
flayer <- function(x) (x*layer_pd)
fconv <- function(x) (x*furnished_layer_share)
ffurn <- function(x) (x*(1-furnished_layer_share))


per_dollar <- function(x) {
  x %>%
# Apply adjustment functions
```

```
    mutate_at(vars(contains("broiler")),fbroiler) %>%
    mutate_at(vars(contains("layer")),flayer) %>%
    mutate_at(vars(contains("_conv_")),fconv) %>%
    mutate_at(vars(contains("_furn_")),ffurn) %>%
# Combine layer transitions
    mutate(
      layer_excruciating=layer_conv_excruciating+layer_furn_excruciating,
      layer_disabling=layer_conv_disabling+layer_furn_disabling,
      layer_hurtful=layer_conv_hurtful+layer_furn_hurtful,
      layer_annoying=layer_conv_annoying+layer_furn_annoying
          ) %>%
    select(-contains("_conv_")) %>%
    select(-contains("_furn_"))
}
```

I then apply the functions to both the "full" and "mean" duration estimates.

```
chicken_benefits_full_pd <- per_dollar(chicken_benefits_full)
chicken_benefits_mean_pd <- per_dollar(chicken_benefits_mean)
```

Convert

```
fish_hours_per_dollar <- fypd_scale*(24*365.25)
```

Work out proportion of simulations where future fish slaughter campaigns might beat historical chicken campaigns

I now work out the share of simulations where future fish slaughter corporate commitment work might result in a greater reduction in time in pain per dollar spent than historical chicken campaigns.

Given an absence of data/evidence of the time in different pain states for fish under conventional slaughter methods and under electrical stunning, the results are presented conditional on the amount of time in each pain intensity is reduced as a result of a stunning commitment, expressed as a percentage of the slaughter duration prior to a stunning commitment.

```
#Define time share assumption range
timeshare_assumptions <-
  tibble(
    fish_slaughter_timeshare=seq(-0.2,1,0.001))
```

```
#Define function to work out proportion of simulations where fish slaughter looks like it
fish_slaughter_v_chicken_fun <- function(x,timeshare) {
  mean(fish_hours_per_dollar*timeshare>x)
}

#Define function to produce formmatted table
fish_v_chicken <-  function(chicken_benefits_table) {
    timeshare_assumptions %>%
    rowwise() %>%
    mutate(
  #perform calculation for each combo
    results=
      list(
        map(
          chicken_benefits_table,
          fish_slaughter_v_chicken_fun,
          timeshare=fish_slaughter_timeshare
          ))) %>%
  # Format table
    unnest_wider(col=results) %>%
    pivot_longer(cols=2:9,names_to="comparison",values_to="share_of_simulations") %>%
    separate_wider_delim(comparison,"_",names=c("chicken_intervention","pain_type")) %>%
    mutate(
      across(where(is.character), str_to_sentence),
      pain_type= factor(
                str_c(pain_type," Pain"),
                levels= c("Excruciating Pain","Disabling Pain","Hurtful Pain","Annoying Pa
}

fish_v_chicken_full <- fish_v_chicken(chicken_benefits_full_pd)
fish_v_chicken_mean <- fish_v_chicken(chicken_benefits_mean_pd)
```

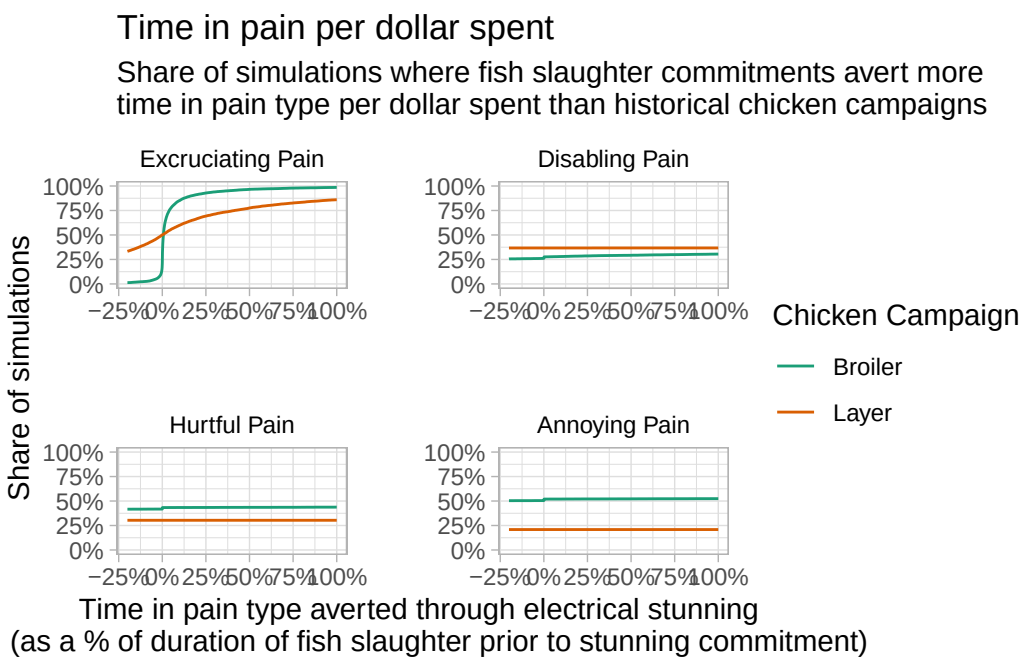
Create output charts

Write function to generate output charts

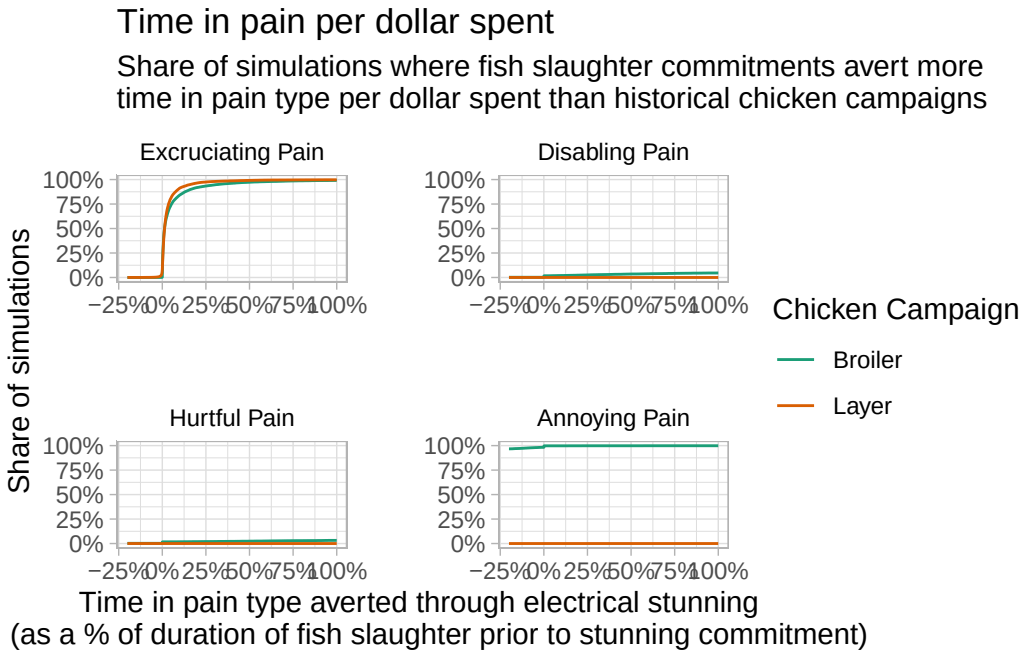
```
output_chart <- function(data) {
  data %>%
  ggplot(aes(
```

```
    x=fish_slaughter_timeshare,
    y=share_of_simulations,
    color=chicken_intervention))+
  geom_line() +
  facet_wrap(vars(pain_type),scales='free') +
  theme_light() +
  scale_color_brewer(palette="Dark2")+
    labs(
    title ="Time in pain per dollar spent ",
    subtitle="Share of simulations where fish slaughter commitments avert more
time in pain type per dollar spent than historical chicken campaigns",
    y="Share of simulations" ,
    x="Time in pain type averted through electrical stunning
    (as a % of duration of fish slaughter prior to stunning commitment)",
    color="Chicken Campaign")+
   theme(strip.background = element_rect(fill="white"),
         strip.text = element_text(colour = 'black'))+
    theme(panel.spacing = unit(2, "lines"))+
  scale_y_continuous(limits=c(0,1),labels = scales::percent) +
  scale_x_continuous(labels = scales::percent)
}
```

```
output_chart(fish_v_chicken_full)
```




```
output_chart(fish_v_chicken_mean)
```



Summary table for excruciating pain panel for full method.

```
duration_share <- c((-1:5)/100,0.10,0.15,0.2,0.25,0.50,0.75,1)
fish_v_chicken_full %>%
  filter(
    pain_type=="Excruciating Pain",
    round(fish_slaughter_timeshare,3) %in% duration_share) %>%
  select(-pain_type) %>%
  pivot_wider(names_from=chicken_intervention,values_from=share_of_simulations) %>%
  gt() %>%
  fmt_percent(decimals=0) %>%
   tab_header(
    title = "Share of simulations where fish stunning intervention averts more excruciatin
  tab_spanner(
    label = "Chicken intervention",
    columns = 2:3) %>%
  cols_label(
    fish_slaughter_timeshare = "Excruciating pain averted through
    fish stunning intervention
    (expressed as a % of conventional
    slaughter duration)")
```

Share of simulations where fish stunning intervention averts more excruciating than historical chicken campaigns

Excruciating pain averted through fish stunning intervention (expressed as a % of conventional slaughter dura

Chart description

The charts seek to explore the plausibility of a fish stunning intervention reducing more time in specific pain categories compared to historic chicken campaigns.

Pain categories are from the Welfare Footprint Project (https://welfarefootprint.org/) and historic chicken estimates are from Šimčikas (2019)'s Guesstimate model.

The x-axis expresses the time in pain potentially averted through stunning of fish, expressed as a % of the duration of fish slaughter prior to the stunning commitment.

The y-axis expresses the share of simulations where the fish stunning intervention appears to reduce more time in pain than historic chicken campaigns, conditional on this assumption.

Each panel represents a pain intensity from the Welfare Footprint project paintrack framework.

The "full" chart takes into account uncertainty in WFP's estimates of time in pain averted, and includes the possiblity that historic chicken campaigns could increase time in pain.

The "mean" chart assumes that the time in pain reduction for each category is based on the difference in means between the average time in pain in the reformed and unreformed categories.

Chart interpretation and conclusions

- For annoying, disabling and hurtful pain, there is not much variation between the duration of time in pain averted from the fish stunning intervention and whether the fish intervention beats the chicken intervention.
- For these pain categories, the extent to which the fish intervention beats the chicken intervention is almost entirely driven by the share of simulations where the chicken intervention *increases* time in pain. The time pain averted/gained through the chicken intervention is much larger than anything that might be possible through the fish stunning intervention - mainly because chicken interventions affect a much larger amount of time.
- For excruciating pain however, the extent to which the fish stunning intervention beats the chicken intervention is highly sensitive to relatively small changes. If a fish experiences excruciating pain for at least 10-15% of the duration of conventional fish slaughter, and most of this is averted as a result of stunning, it seems plausible that the fish stunning intervention could avert more time in excruciating pain than historic chicken campaigns.
- For readers who primarily care about averting high intensity pain, fish stunning interventions may be more appealing than chicken campaigns.

- Further research on the intensity of pain experienced during conventional slaughter and the extent to which this is mitigated by stunning may be helpful.