Estimating Perception Bias From 2024-2025 NARWSS 1000 ft Two-Team Surveys

Jason J. Roberts

2026-06-30

Table of contents

1 Introduction **1**

2 Strategy **2**

3 Data preparation **2**
3.1 Load the data . . . 2
3.2 Clean up some records . . . 4
3.3 Wrangle the data into what `ddf()` needs . . . 5
3.4 Add covariates . . . 10
3.4.1 Species . . . 10
3.4.2 Group size . . . 11
3.4.3 Observation conditions . . . 13

4 Analysis **19**

5 Discussion **27**
5.1 Model evaluation . . . 27
5.2 Final `p(0)` estimates . . . 30
5.3 Applicability to other survey programs . . . 32

Revision History

Date	Description
2025-08-19	Analysis finalized and first version of this document released.
2026-06-30	In response to a NOAA May 2026 peer review, I revised section 4 to include AICc and associated statistics that are appropriate for small sample sizes, and updated the discussion in section 5.1 accordingly.

1 Introduction

The North Atlantic Right Whale Sighting Survey (NARWSS), conducted by NOAA Northeast Fisheries Science Center, has long used a single observer team. To correct these data for perception bias in the NARW density model, we have traditionally relied on a perception bias estimate developed for NOAA's Atlantic Marine Assessment Program for Protected Species (AMAPPS) aerial surveys (Palka et al. 2017, 2021). However, the AMAPPS surveys were conducted at 600 ft. altitude and differed from NARWSS in other important ways. Density and abundance

scale inversely with perception bias, and a large error in perception bias estimation will yield a correspondingly large (inverse) error in density and abundance. It has therefore remained an important priority for NARWSS to develop its own perception bias estimate using data collected during its own surveys.

In winter 2024 through summer 2025, NARWSS conducted surveys with a second observer on the right side, collecting data sufficient to be analysed with mark-recapture distance sampling (MRDS). This notebook presents an analysis of the data available as of 17 July 2025.

2 Strategy

Following the flowchart from Burt et al.'s (2014) MRDS overview paper, we first ask the question: is responsive movement anticipated? I.e., do whales detect the observers and move in response, before observers detect the animals and record a distance? The answer is no: we have no evidence of responsive movement. Observations suggest the aircraft flies high enough that whales rarely exhibit responses to it, and fast enough that observers can bring animals abeam to collect perpendicular distances before they move substantially. Therefore, according to MRDS methodology, we can safely make the *point independence assumption*, which is that the probability of detection by the observer setting up an observation trial is independent of the probability of detection by the other observer *only at distance 0* (i.e. on the transect line). In other words, we can safely assume that the animal distribution is uniform with respect to the observers.

We do not need to make the *full independence assumption*, which assumes that probability of detection by the observer setting up the trial is independent of the probability of detection by the other observer at *all distances*. The advantage of a full independence model is that it is robust to responsive movement, but it can be quite sensitive to unmodeled heterogeneity in detectability. Because of that, avoiding it is recommended unless it is needed to address responsive movement.

3 Data preparation

3.1 Load the data

Load packages we need. We'll use tidyverse for most data wrangling.

```
library(furrr)
```

```
Loading required package: future
```

```
library(hms)
library(janitor)
```

```
Attaching package: 'janitor'
```

```
The following objects are masked from 'package:stats':

    chisq.test, fisher.test
```

```
library(lubridate)
```

```
Attaching package: 'lubridate'

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

    hms

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

    date, intersect, setdiff, union
```

```
library(mrds)
```

```
This is mrds 3.0.1
Built: R 4.5.1; ; 2025-08-20 02:07:30 UTC; windows
```

```
library(readxl)
library(tidyverse)
```

```
-- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
v dplyr   1.1.4     v readr   2.1.5
v forcats 1.0.1     v stringr 1.5.2
v ggplot2 4.0.0     v tibble  3.3.0
v purrr   1.1.0     v tidyr   1.3.1

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

```
library(withr)

# Monkey patch the mrds plot function to plot observations in different colors
# when exactly one covariate is used.

source("my_plot_cond.R")
assignInNamespace("plot_cond", my_plot_cond, ns = "mrds")
```

Read the first sheet of the .xlsx, called “Ind Obs & VOR R Whales”.

```
sheet1 <- read_excel("Perception Bias Sightings Data 2024-2025.xlsx", sheet = 1)
```

Reconstruct the DATETIME_ET column, which we need for joining to “VOR raw data” sheet.

```r
sheet1 <- sheet1 |>
  mutate(
    date_part = if_else(as_date(Date) == as.Date("2024-04-23"), # Fix wrong dates
                        as.Date("2025-04-23"),
                        as_date(Date)),
    secs_since_mid = as.numeric(as_hms(Time)),     # Get the secs since midnight
    datetime = as.POSIXct(date_part) + secs_since_mid,  # Add them
    DATETIME_ET = force_tz(datetime, "America/New_York")   # Force back to EST/EDT
  )
```

To analyze the data with the R `mrds` package, we need to wrangle it into this structure. Start by selecting and renaming the columns we need. Call our new data frame `detections`, like they do in the distance sampling training materials (this is not required).

```r
stopifnot(sheet1$Observer %in% c("Obs 1", "Obs 2"))

detections <- sheet1 |>
  transmute(
    object = `Trial Number`,
    observer = if_else(Observer == "Obs 1", 1, 2),  # Just get the number
    detected = 1,
    distance = 1000*0.3048 * tan(ANGLE*pi/180),     # ft to m, deg to rad
    size = GROUP_SIZE,
    SPCODE,
    DATETIME_ET
  )
```

3.2 Clean up some records

We need to clean up some records before proceeding. `Trial Number` (i.e. `object`) 31 has two records for Obs 1. We need to choose one and discard the other. Choose the one that reported RIWH as the `SPCODE` (CUE was ANIMAL) rather than UNLW (CUE of FOOTPRINT).

```r
detections <- detections |>
  filter(is.na(object) | !(object == 31 & observer == 1 & SPCODE == "UNLW"))
```

`object` 19 is a sighting made by Obs 2 after refueling but before Obs 1 had come back on effort yet. This sighting should not be counted as a trial. Drop it.

```r
detections <- detections |>
  filter(is.na(object) | !(object == 19))
```

`object` 10 has `SPCODE` of UNLW but comment says "we broke on this and it ended up being 2 humpbacks". Change `SPCODE` to HUWH and `size` to 2. We presume that because this was a large whale sighting and the plane broke and obtained a better species ID and group size, the final edit of the data would utilize that ID and group size. This sighting would therefore be included in the detection function for large whales and in the HUWH density model. So we want to include it in our perception bias estimate as well.

```
detections <- detections |>
  mutate(
    SPCODE = if_else(!is.na(object) & object == 10, "HUWH", SPCODE),
    size = if_else(!is.na(object) & object == 10, 2, size)
  )
```

3.3 Wrangle the data into what `ddf()` needs

Each `detection` must have an `object` number. In the spreadsheet, all Obs 2 records had a `Trial Number` but Obs 1 records only had one if the detection was also made by Obs 2. Although detections made by Obs 1 but not Obs 2 cannot be used in the mark-recapture part of the model (`mrmodel`), they can still be used to fit the detection function in the distance sampling part of the model (`dsmodel`). I'm not sure this will make a difference to us, since our intent is to discard the detection function produced by the `dsmodel` and only extract the `p(0)` parameter (a.k.a. `g(0)`) from the `mrmodel`. That `p(0)` represents our perception bias correction that we will apply to surveys used in the NARW density model, which are all from earlier years and have their own single-team detection functions. But because it is not hard to keep these Obs 1-only detections, we'll do it. To make them easy to distinguish from detections also made by Obs 2, we'll assign them `object` numbers of 1000 and higher.

```
# Fail there are any Obs 2 detections without an object (i.e. Trial Number)

stopifnot(!is.na(detections$object[detections$observer == 2]))

# Assign object numbers starting at 1000 to detections only made by Obs 1, i.e.
# those without an object number already.

detections <- detections |>
  mutate(object = coalesce(object, 999 + cumsum(is.na(object))))

# Now fail there are any Obs 1 or 2 detections without an object

stopifnot(!is.na(detections$object))
```

When doing a two-team analysis, the `mrds ddf()` function requires that `detections` have two rows for every object. When an observer misses a detection, their row should have `detected = 0`. Create those rows.

```
detections <- detections |>
  complete(
    object,
    observer = c(1, 2),
    fill = list(detected = 0)
  )
```

If I recall correctly, `ddf()` also requires that for a given object, both observers have the same `distance`, `size`, and any covariates such as `SPCODE`. So for each object, we need to compare what each observer reported, and if they are different, choose the best one. Doing this in R will be much easier if we pivot the current data frame so that each object just has one row with separate columns for each observer for `detected`, `distance`, `size`, `SPCODE`, and `DATETIME_ET`.

```
detections_wide <- detections |>
  pivot_wider(
    id_cols = object,
```

```
    names_from = observer,
    values_from = c(detected, distance, size, SPCODE, DATETIME_ET),
    names_sep = ""    # E.g. distance becomes distance1 and distance2
  )
```

For `distance`, `size`, and `DATETIME_ET`, use observer 1's values if observer 1 made the sighting; otherwise use observer 2's values. For `SPCODE`, use observer 1's values unless observer 1 reported an ID beginning with "UN", in which case use observer 2's, unless observer 1 reported "UNFS" and observer 2 reported "UNLW". Ultimately, for sightings that were circled, we don't care about what the original ID was, only the final ID, so we assume that whichever ID from the two observers had the highest taxonomic resolution was the final ID (with observer 1 winning if they reported different IDs and neither was "UN"-something).

```
detections_wide <- detections_wide |>
  mutate(
    distance = if_else(!is.na(distance1), distance1, distance2),
    size = if_else(!is.na(size1), size1, size2),

    SPCODE = case_when(
      !is.na(SPCODE1) & is.na(SPCODE2) ~ SPCODE1,
      is.na(SPCODE1) & !is.na(SPCODE2) ~ SPCODE2,
      SPCODE1 == "UNFS" & SPCODE2 == "UNLW" ~ SPCODE1,
      str_starts(SPCODE1, "UN") & !str_starts(SPCODE2, "UN") ~ SPCODE2,
      TRUE ~ SPCODE1
    ),

    DATETIME_ET = if_else(!is.na(DATETIME_ET1), DATETIME_ET1, DATETIME_ET2)
  )
```

Drop the per-observer columns other than `detected1` and `detected2` and then pivot back to a long-form data frame, so that each observer gets their own row. Finally, order the columns like they appear in the distance sampling training. This is not required but is nice for comparison.

```
detections2 <- detections_wide |>
  select(object, detected1, detected2, distance, size, SPCODE, DATETIME_ET) |>
  pivot_longer(
    cols = c(detected1, detected2),
    names_to = "observer",
    values_to = "detected",
    names_pattern = "detected(\\d)"
  ) |>
  select(object, observer, detected, distance, size, SPCODE, DATETIME_ET)
```

There are a few detections without group sizes. For all of these, assume group size was 1.

```
# Print the records without group sizes.

detections2 |> filter(is.na(size))
```

object	observer	detected	distance	size	SPCODE	DATETIME_ET
9	1	0	NA	NA	UNLW	2025-02-08 14:57:06

object	observer	detected	distance	size	SPCODE	DATETIME_ET
9	2	1	NA	NA	UNLW	2025-02-08 14:57:06
16	1	0	2899.9783	NA	UNLW	2025-02-15 09:50:38
16	2	1	2899.9783	NA	UNLW	2025-02-15 09:50:38
21	1	0	4358.8431	NA	UNLW	2025-02-15 16:06:23
21	2	1	4358.8431	NA	UNLW	2025-02-15 16:06:23
25	1	0	8728.3300	NA	UNLW	2025-02-28 15:50:48
25	2	1	8728.3300	NA	UNLW	2025-02-28 15:50:48
43	1	0	837.4311	NA	FIWH	2025-03-28 14:51:07
43	2	1	837.4311	NA	FIWH	2025-03-28 14:51:07

```
# Set NA to 1 for these records.

detections2 <- detections2 |> mutate(size = if_else(is.na(size), 1, size))

# Verify there are now zero rows with a size of NA.

any(is.na(detections2$size))
```

```
[1] FALSE
```

There are a few detections without distances, e.g. because the plane was on a turn or because they were "alt obs" sightings. We cannot use these; remove them from the data frame.

```
# Print the records without distances.

detections2 |> filter(is.na(distance))
```

object	observer	detected	distance	size	SPCODE	DATETIME_ET
9	1	0	NA	1	UNLW	2025-02-08 14:57:06
9	2	1	NA	1	UNLW	2025-02-08 14:57:06
98	1	1	NA	1	SEWH	2025-04-21 13:18:56
98	2	1	NA	1	SEWH	2025-04-21 13:18:56
1002	1	1	NA	1	FIWH	2024-12-14 13:21:26
1002	2	0	NA	1	FIWH	2024-12-14 13:21:26
1004	1	1	NA	2	RIWH	2024-12-15 13:36:09
1004	2	0	NA	2	RIWH	2024-12-15 13:36:09
1009	1	1	NA	1	FIWH	2025-02-08 15:09:40
1009	2	0	NA	1	FIWH	2025-02-08 15:09:40
1015	1	1	NA	1	HUWH	2025-02-28 13:36:16
1015	2	0	NA	1	HUWH	2025-02-28 13:36:16
1043	1	1	NA	1	MIWH	2025-05-08 11:38:42
1043	2	0	NA	1	MIWH	2025-05-08 11:38:42

```
# Filter out these records.

detections2 <- detections2 |> filter(!is.na(distance))

# Verify there are now zero rows with a distance of NA.
```

```r
any(is.na(detections2$distance))
```

```
[1] FALSE
```

Summarize counts of sightings for our current data frame by SPCODE and by which observers made the sighting. As noted above, a sighting is represented by two rows of the data frame, one for each observer. If a sighting is made by both observers, we have to be sure not to double count it in our summary. We will need to `pivot_wider()` again to do this.

```r
# Define a function for creating this summary. (Thanks to ChatGPT for help.)

summarize_categorical_counts <- function(detections_df, by_col) {
  detections_df |>
    # Decide detection category for each object
    group_by(object, {{ by_col }}) |>
    summarise(
      obs1 = any(observer == 1 & detected == 1),
      obs2 = any(observer == 2 & detected == 1),
      .groups = "drop"
    ) |>
    mutate(category = case_when(
      obs1 &  obs2 ~ "Both Obs",
      obs1 & !obs2 ~ "Obs 1 Only",
      !obs1 & obs2 ~ "Obs 2 Only"
    )) |>

    # Count objects per by_col per category
    count({{ by_col }}, category, name = "n") |>

    # Pivot wide so categories become columns
    pivot_wider(
      names_from  = category,
      values_from = n,
      values_fill = 0
    ) |>

    # Add the per-row total column
    mutate(Total = `Obs 1 Only` + `Obs 2 Only` + `Both Obs`) |>

    # Append the Total row
    adorn_totals(where = "row", name = "Total") |>

    # Move columns
    relocate(`Both Obs`, .after = `Obs 2 Only`) |>
    relocate(Total, .after = `Both Obs`)
}

# Print a table summarizing counts our current detections data frame

detections2 |>
  summarize_categorical_counts(SPCODE)
```

SPCODE	Obs 1 Only	Obs 2 Only	Both Obs	Total
FIWH	6	10	20	36
HUWH	11	10	39	60
MIWH	6	5	8	19
Probable tuna	0	1	0	1
RIWH	26	9	37	72
SEWH	3	1	11	15
SPWH	2	0	4	6
UNFS	1	0	1	2
UNID	0	1	0	1
UNLW	6	12	2	20
UNMW	1	0	0	1
Total	62	49	122	233

The perception bias estimate we are developing is intended to be applied in the density models to NARWSS detection functions developed for those models. The model of main interest at the time of this writing is the upcoming NARW U.S.-Canada transboundary model, but in the 2026-2028 timeframe we will be updating all of the other models for the U.S. Navy. These detection functions are built from fully-identified sightings of large whales, plus UNFS (unidentified fin or sei whale) sightings. In order for the perception bias estimate to match the detection functions, we have to exclude other taxa such as minke whales or UNLW (unidentified large whales). Print a summary of the sightings we'll exclude.

```
# Define the taxa we're going to include.

SPCODE_of_interest <- c("BLWH", "FIWH", "HUWH", "RIWH", "SEWH", "SPWH", "UNFS")

# Print a table summarizing counts for SPCODEs we're excluding.

detections2 |>
  filter(!(SPCODE %in% SPCODE_of_interest)) |>
  summarize_categorical_counts(SPCODE)
```

SPCODE	Obs 1 Only	Obs 2 Only	Both Obs	Total
MIWH	6	5	8	19
Probable tuna	0	1	0	1
UNID	0	1	0	1
UNLW	6	12	2	20
UNMW	1	0	0	1
Total	13	19	10	42

Now drop those species from our data frame and summarize what's left.

```
# Filter to species of interest.

detections3 <- detections2 |>
  filter(SPCODE %in% SPCODE_of_interest)

# Print a table summarizing counts for SPCODEs we're keeping.
```

```
detections3 |>
  summarize_categorical_counts(SPCODE)
```

SPCODE	Obs 1 Only	Obs 2 Only	Both Obs	Total
FIWH	6	10	20	36
HUWH	11	10	39	60
RIWH	26	9	37	72
SEWH	3	1	11	15
SPWH	2	0	4	6
UNFS	1	0	1	2
Total	49	30	112	191

3.4 Add covariates

Detection probability often varies by species, group size, and observation conditions such as sea state. The NARWSS program collected a number of observation conditions that have previously been shown to influence detection probability. To consider the influence of species and these covariates for the perception bias model, I will fit a set of exploratory candidate models that try each covariate and several combinations of them.

Detection models fitted with the `mrds` package can have covariates that are either categorical (e.g. glare or quality, as NARWSS currently defines them) or continuous (e.g. SST or seafloor depth, although we don't typically use those as covariates in detection models). One exception is Beaufort sea state, which is often treated as both.

An important consideration for categorical covariates is that there must be a sufficient number of sightings in each category (a.k.a. *level*). If there are too few for a category, its parameter estimate (the strength of the effect) will be smaller than its standard error estimate. In that situation, the parameter estimate is not significantly different from zero in a formal statistical sense. In that case, my practice is to exclude that model formulation from further consideration.

There is no widely recognized recommendation for the minimum number of sightings required for each categorical level. In my experience, extreme standard errors occur with only 2 or 3 sightings, but the standard error may drop below the parameter estimate (making it significant) with only 5 or 10. However, I like to have more if possible, and feel most comfortable when there are 30 or more, based on that threshold being a old classroom rule-of-thumb for minimum sample size (but do not cite me as recommending this).

3.4.1 Species

We can see in the previous table that there were only 6 total SPWH sightings, only 2 total UNFS sightings, and that there was only 1 SEWH sighting made only by Obs 2. These numbers are too low. To mitigate this, I will pool FIWH, SEWH, and UNFS, and pool HUWH and SPWH. This follows how we pooled these species for our most recent NEFSC AMAPPS detection function. I'll call this pooled covariate `SPCODE2`.

```
detections3 <- detections3 |>
  mutate(
    SPCODE3 = case_when(
      SPCODE %in% c("FIWH", "SEWH", "UNFS")  ~ "FIWH+SEWH+UNFS",
      SPCODE %in% c("HUWH", "SPWH")  ~ "HUWH+SPWH",
      TRUE ~ "RIWH"
    ))

covariates <- c("SPCODE3")
```

```
detections3 |>
  summarize_categorical_counts(SPCODE3)
```

SPCODE3	Obs 1 Only	Obs 2 Only	Both Obs	Total
FIWH+SEWH+UNFS	10	11	32	53
HUWH+SPWH	13	10	43	66
RIWH	26	9	37	72
Total	49	30	112	191

In some data I've worked with, only RIWH have been substantially different. To allow for this, create another covariate `SPCODE2` that splits out RIWH and pools everything else together.

```
detections3 <- detections3 |>
  mutate(
    SPCODE2 = case_when(
      SPCODE == "RIWH" ~ "RIWH",
      TRUE ~ "Non-RIWH"
    ))

covariates <- c(covariates, "SPCODE2")

detections3 |>
  summarize_categorical_counts(SPCODE2)
```

SPCODE2	Obs 1 Only	Obs 2 Only	Both Obs	Total
Non-RIWH	23	21	75	119
RIWH	26	9	37	72
Total	49	30	112	191

Seeing that the ratios of `Obs 2 Only` to `Both Obs` is about the same for `Non-RIWH` and `RIWH`, I'm not optimistic that this covariate will make a difference, but we'll try it anyway.

3.4.2 Group size

Larger groups are easier to see than smaller groups. Let's look at the counts of sightings by group size.

```
detections3 |>
  summarize_categorical_counts(size)
```

size	Obs 1 Only	Obs 2 Only	Both Obs	Total
1	43	25	84	152
2	4	5	25	34
3	0	0	3	3
4	1	0	0	1
9	1	0	0	1
Total	49	30	112	191

size	Obs 1 Only	Obs 2 Only	Both Obs	Total

This is similar to what occurred with the NEFSC AMAPPS large whale `p(0)` estimation work done by Palka et al. (2021): nearly all the sightings were of either 1 or 2 animals. Because of that, on our previous model, we applied their `p(0)` to sightings of 1 or 2 animals and then assumed that `p(0) = 1` for sightings of 3 or more animals. The data here would tenuously support that decision, insofar as all three sightings that were made by Obs 2 were also made by Obs 1, a 100% success rate. Of course, it is only 3 sightings.

These data also suggest there is a difference between 1 animal and 2 animals. Support for this is stronger, by virtue of having 30 trials instead of only 3.

To explore these possibilities, we'll define two categorical covariates. Because size is a numerical value, we have to explicitly use `factor()` to treat the values as categorical. First, we'll define the `size3` covariate, with three levels of 1, 2, and 3 or more.

```
detections3 <- detections3 |>
  mutate(
    size3 = factor(case_when(
      size == 1 ~ "1",
      size == 2 ~ "2",
      TRUE ~ "3 or more"
    )))

covariates <- c(covariates, "size3")

detections3 |>
  summarize_categorical_counts(size3)
```

size3	Obs 1 Only	Obs 2 Only	Both Obs	Total
1	43	25	84	152
2	4	5	25	34
3 or more	2	0	3	5
Total	49	30	112	191

Given that there are only 5 sightings of 3 or more, and only 3 of them were made by Obs 2, we'll also define a `size2` covariate for comparison, with just two levels: 1 and 2 or more.

```
detections3 <- detections3 |>
  mutate(
    size2 = factor(case_when(
      size == 1 ~ "1",
      TRUE ~ "2 or more"
    )))

covariates <- c(covariates, "size2")

detections3 |>
  summarize_categorical_counts(size2)
```

size2	Obs 1 Only	Obs 2 Only	Both Obs	Total
1	43	25	84	152
2 or more	6	5	28	39
Total	49	30	112	191

3.4.3 Observation conditions

We'd also like to consider observation conditions, specifically VISIBILITY_NM, BEAUFORT, CLOUD_CODE, GLARE_R, and QUALITY_R. To get these, we need to read the "VOR raw data" spreadsheets into a data frame and then join on DATETIME_ET. But there is a complication: not all records logged by Obs 2 have exactly matching records in the VOR raw data. I presume this was because Obs 2 logged their sightings with separate software, while the data recorder logged Obs 1's sightings with VOR. Because of this, we can't require an exact match between the two tables on DATETIME_ET. Instead, we have to find the record in the VOR raw data that is the the closest DATETIME_ET that is less than or equal to that in our detections table.

```
vor_files <- list.files(path = "P.B. Study VOR raw data - for Jason",
                        pattern = "^f_.*\\.xlsx$",
                        full.names = TRUE)

vor_raw_data <- vor_files |>
  map_dfr(read_excel,
          range = cell_cols("A:U")) |>   # Skip columns we don't need that cause problems
  mutate(
    DATETIME_ET = force_tz(DATETIME_ET, "America/New_York")
  ) |>
  filter(!(DATETIME_ET == "2025-04-18 12:03:37" & EVENT_NUMBER == 1000)) |>  # Delete duplicate recor
  arrange(DATETIME_ET)    # Sort in ascending DATETIME_ET order

detections4 <- detections3 |>
  left_join(
    vor_raw_data,
    join_by(closest(DATETIME_ET >= DATETIME_ET))
  )
```

Check that all records have a VOR raw data record that is close to them in time.

```
detections4 |>
  mutate(
    time_diff = DATETIME_ET.x - DATETIME_ET.y
  ) |>
  filter(observer == 1) |>   # time_diff is the same for both obs; just print one
  select(time_diff) |>
  as.vector()
```

```
$time_diff
Time differences in secs
  [1] 0 0 0 0 0 0 0 6 0 0 0 0 4 4 0 0 0 0 0 0 4 0 0 0 6 0 0 0 0 0 0 0 4 0 3 3 5
 [38] 7 0 0 0 2 4 6 0 0 4 0 0 0 0 0 0 0 0 0 0 0 6 2 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 [75] 0 0 0 0 0 0 1 0 7 0 0 1 0 6 0 0 0 0 0 0 5 4 0 0 0 0 0 0 0 1 0 0 0 0 1 0 3
[112] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 6 0 0 0 0 0 0 8 0 0 0 0 0 0
```

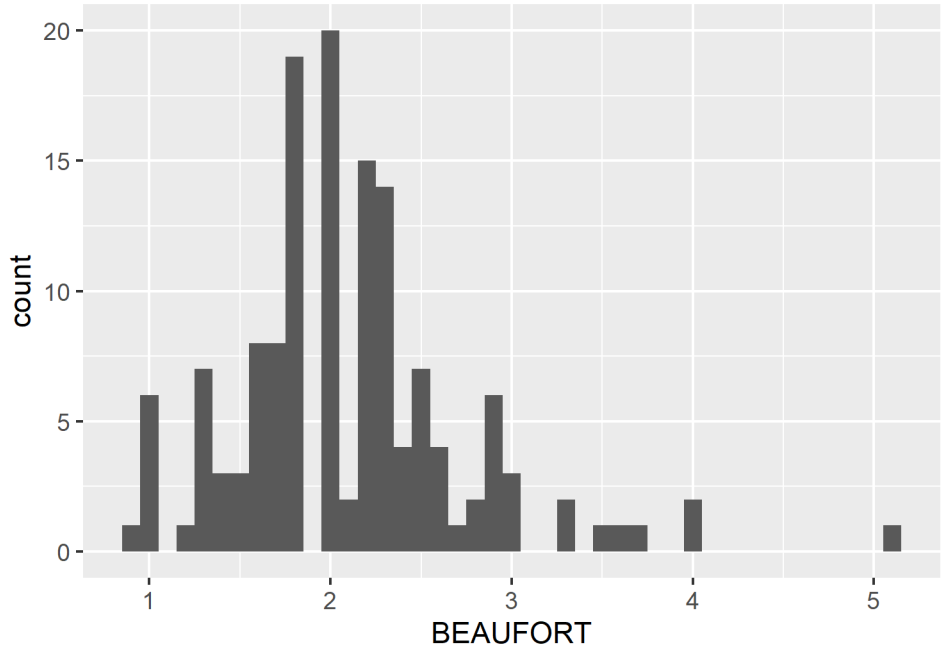
```
[149] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
[186] 0 0 0 0 0 0
```

All records have a VOR raw data record collected within a few seconds. The observation conditions that occurred then are likely to still be valid. We can now add covariates for these.

3.4.3.1 Beaufort sea state

Although Beaufort wind force is a categorical scale with integer levels of 0 to 12, NARWSS reports Beaufort in units of 0.1:

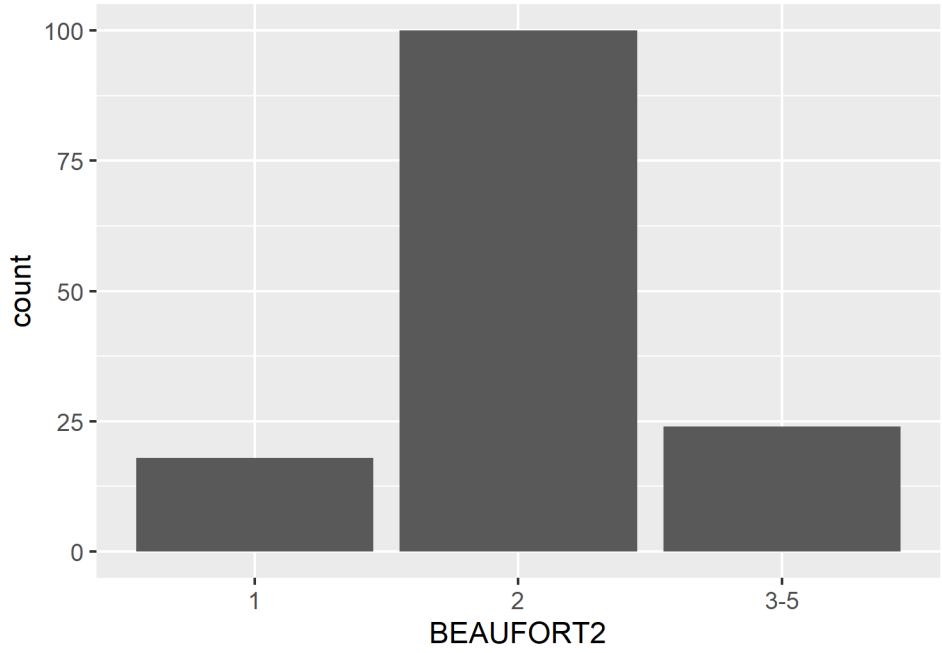
```
detections4 |>
  filter(observer == 2 & detected == 1) |>
  ggplot(aes(x = BEAUFORT)) +
    geom_histogram(binwidth = 0.1)
```



We'll try it as a continuous covariate, but also as a categorical covariate, classified to the nearest integer value. Because there were so few sightings at Beaufort 4 and 5, pool them with 3.

```
detections5 <- detections4 |>
  mutate(BEAUFORT2 = round(BEAUFORT)) |>
  mutate(BEAUFORT2 = factor(case_when(
      BEAUFORT2 == 1 ~ "1",
      BEAUFORT2 == 2 ~ "2",
      TRUE ~ "3-5"
    )))

detections5 |>
  filter(observer == 2 & detected == 1) |>
  ggplot(aes(x = BEAUFORT2)) +
    geom_bar()
```



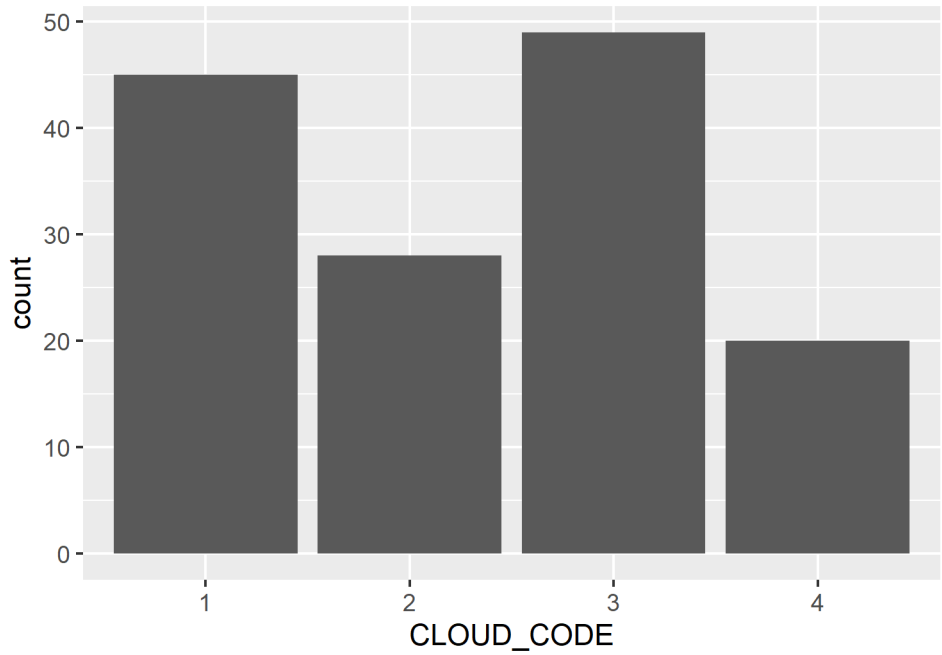

```
covariates <- c(covariates, "BEAUFORT", "BEAUFORT2")
```

3.4.3.2 Cloud code

Cloud code has a good spread of values. We'll try it as a categorical covariate.

```
detections5 <- detections5 |>
  mutate(CLOUD_CODE = factor(CLOUD_CODE))

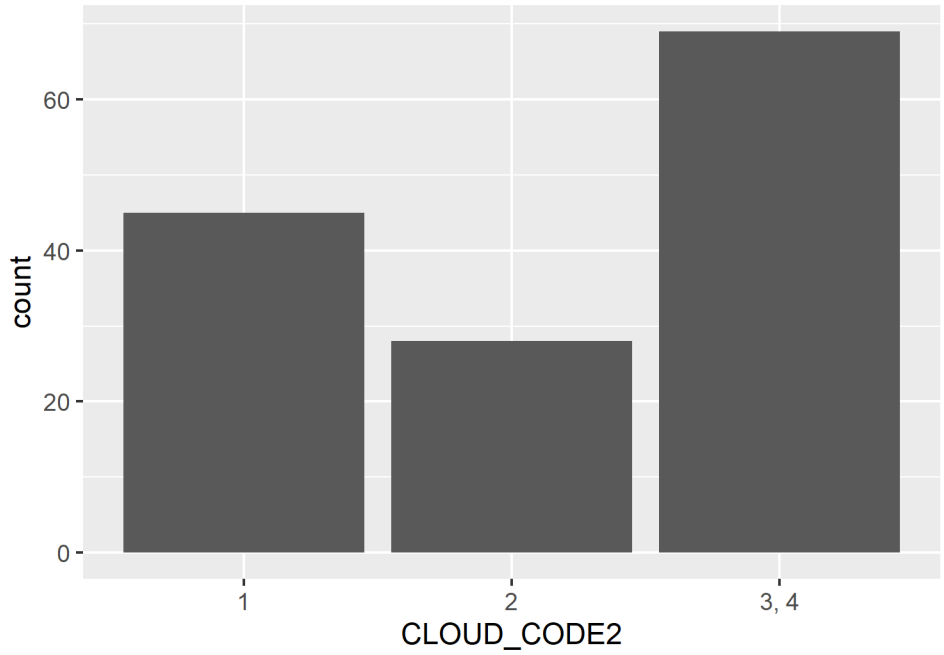
detections5 |>
  filter(observer == 2 & detected == 1) |>
  ggplot(aes(x = CLOUD_CODE)) +
    geom_bar()
```



Exploration of candidate models suggested that codes 3 (broken, 50-90% cloud cover) and 4 (overcast, >90% cloud cover) yielded results that were not significantly different. Define a `CLOUD_CODE2` covariate that pools these two into a single category.

```
detections5 <- detections5 |>
  mutate(CLOUD_CODE2 = factor(case_when(
      CLOUD_CODE == "1" ~ "1",
      CLOUD_CODE == "2" ~ "2",
      TRUE ~ "3, 4"
    )))

detections5 |>
  filter(observer == 2 & detected == 1) |>
  ggplot(aes(x = CLOUD_CODE2)) +
    geom_bar()
```

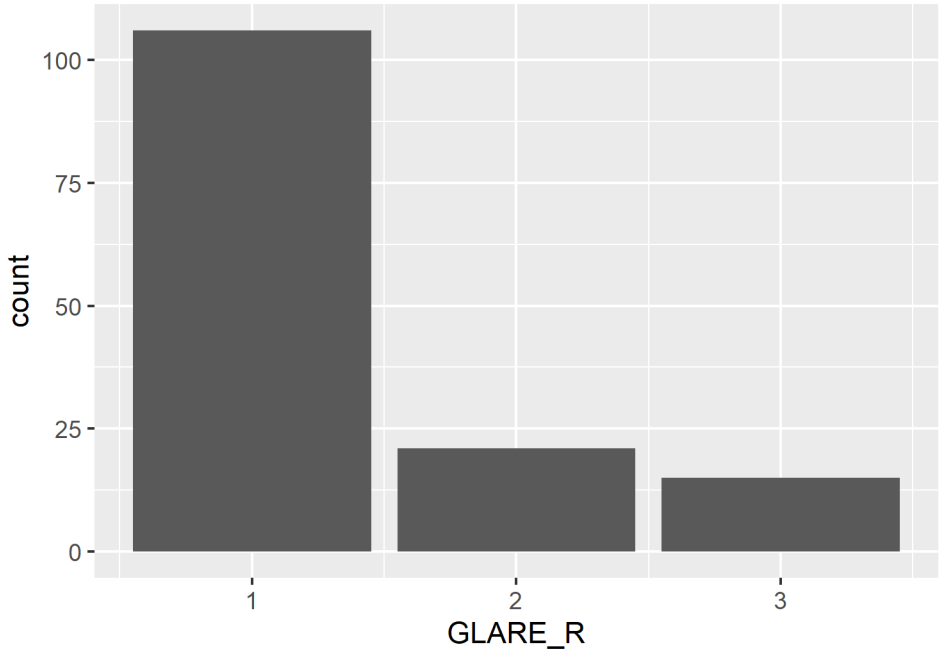



```
covariates <- c(covariates, "CLOUD_CODE", "CLOUD_CODE2")
```

3.4.3.3 Glare

Most records had a glare of `1`, but there were sufficient counts of `2` and `3` to try them without pooling:

```
detections5 |>
  filter(observer == 2 & detected == 1) |>
  ggplot(aes(x = GLARE_R)) +
    geom_bar()
```



We'll try glare as both a continuous and categorical covariate, like Beaufort.

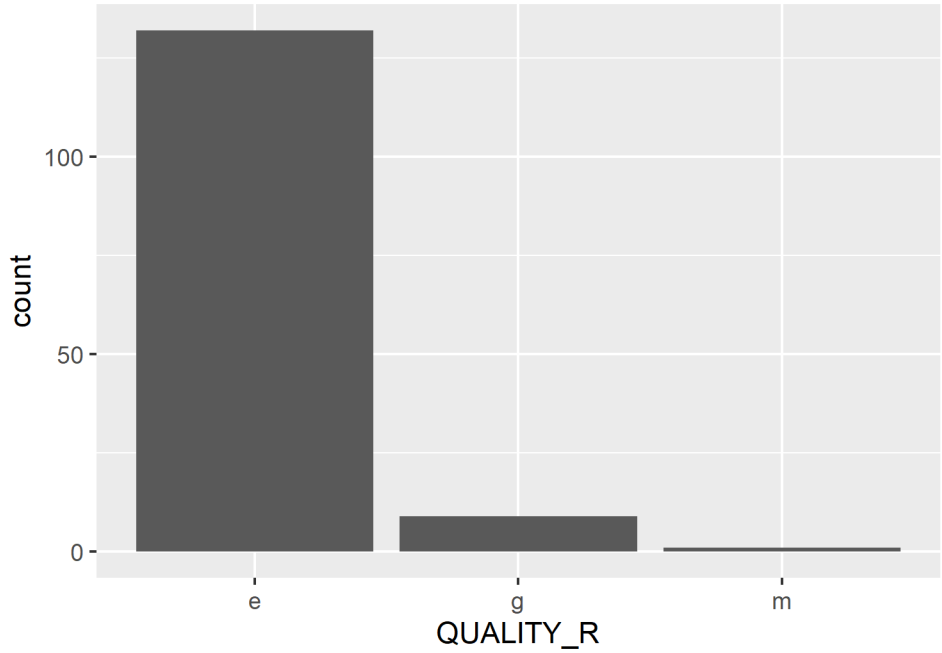
```
detections5 <- detections5 |>
  mutate(GLARE_R2 = factor(GLARE_R))

covariates <- c(covariates, "GLARE_R", "GLARE_R2")
```

3.4.3.4 Quality

Nearly all of the records had `e` (excellent) quality:

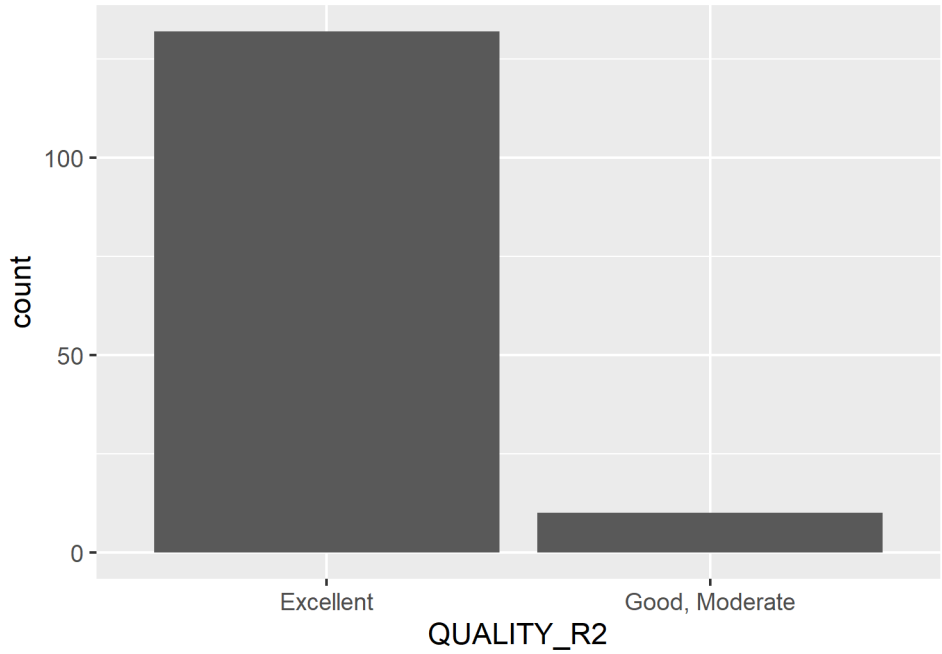
```
detections5 |>
  filter(observer == 2 & detected == 1) |>
  ggplot(aes(x = QUALITY_R)) +
    geom_bar()
```



Because there are too few records with quality `m` (moderate) for it to be used as its own level, we'll combine it with `g` (good).

```
detections5 <- detections5 |>
  mutate(QUALITY_R2 = factor(case_when(
      QUALITY_R == "e" ~ "Excellent",
      TRUE ~ "Good, Moderate"
    )))

detections5 |>
  filter(observer == 2 & detected == 1) |>
  ggplot(aes(x = QUALITY_R2)) +
    geom_bar()
```

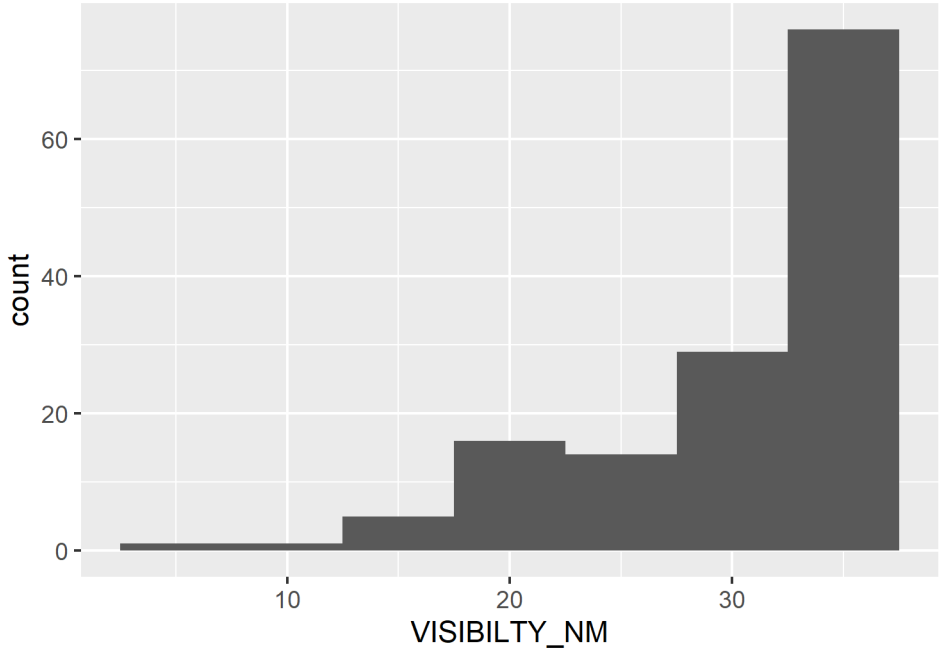


```
covariates <- c(covariates, "QUALITY_R2")
```

3.4.3.5 Visibility

Visibility has a moderate spread of values, with a bias toward the maximum value, indicating the best possible visibility. We'll try it as a continuous covariate.

```
detections5 |>
  filter(observer == 2 & detected == 1) |>
  ggplot(aes(x = VISIBILTY_NM)) +
    geom_histogram(binwidth = 5)
```




```
covariates <- c(covariates, "VISIBILTY_NM")
```

4 Analysis

For our detection functions, we'll use a right truncation distance of 6981 m. This corresponds to an inclination of 87.5 degrees at an altitude of 1000 ft, which ensures we include sightings made at 87 degrees. Define a `meta.data` structure specifying the truncation distance.

```
meta.data <- list(width = 6981)
```

Define a function we can use to construct and execute a call to `ddf()` to fit a single detection function.

```
fit_ddf <- function(key, mr_covariate, ds_covariate, detections, meta.data) {

  # Construct the inner formulas.

  mr_formula <- if (is.na(mr_covariate)) {
    ~ distance
```

```
  } else {
    as.formula(paste("~ distance +", mr_covariate))
  }

  ds_formula <- if (is.na(ds_covariate)) {
    ~ 1
  } else {
    as.formula(paste("~", ds_covariate))
  }

  # Construct expressions for the outer formulas (thanks, ChatGPT).

  mr_expr <- bquote(~glm(link = "logit", formula = .(mr_formula)))
  ds_expr <- bquote(~cds(key = .(as.character(key)), formula = .(ds_formula)))

  # Call ddf with those expressions

  ddf(
    method = "trial",
    data = as.data.frame(detections),   # Ensure data.frame; tibble is problematic
    meta.data = meta.data,
    mrmodel = mr_expr,
    dsmodel = ds_expr
  )
}
```

Construct a data frame enumerating all of the mark-recapture models we'll examine. Include a model with no covariates, a univariate model for each covariate, and models for all combinations of two variables and of three variables, in case several are influential. Omit any combinations that include the same underlying variable more than once. (For example, omit the bivariate combination "BEAUFORT + BEAUFORT2".) Trying up to three combinations yields a lot of candidate models, but most of them will not have statistically significant parameter estimates.

Because we don't care about the distance sampling model, don't generate combinations of different key functions or covariates for the distance sampling model; just use a hazard rate (**hr**) key and no covariates for the distance sampling model.

```
get_prefix <- function(x) str_extract(x, "^[A-Za-z]+")

combos_2var <- combn(covariates, 2, simplify = FALSE)
combos_2var <- Filter(function(x) length(unique(get_prefix(x))) == 2, combos_2var)
combos_2var <- vapply(combos_2var, paste, collapse = " + ", FUN.VALUE = character(1))

combos_3var <- combn(covariates, 3, simplify = FALSE)
combos_3var <- Filter(function(x) length(unique(get_prefix(x))) == 3, combos_3var)
combos_3var <- vapply(combos_3var, paste, collapse = " + ", FUN.VALUE = character(1))

models <- expand(
  tibble(),
  key = "hr",
  mr_covariate = c(NA, covariates, combos_2var, combos_3var),
  ds_covariate = NA
)
```

Fit the models in parallel.

```
models_fitted <- with(
  plan(multisession),
  {
    suppressPackageStartupMessages(library(mrds))
    models |>
      mutate(
        model = future_pmap(
          list(key, mr_covariate, ds_covariate),
          fit_ddf,
          detections = detections5,
          meta.data = meta.data,
          .options = furrr_options(seed = TRUE)
        )
      )
  }
)
```

Generate a table of statistics for each mark-recapture model. Because sample size may be low relative to the number of model parameters, report AICc instead of AIC. Following recommended information-theoretic model-selection practice, report the maximized log-likelihood, number of estimated parameters K, AICc, ΔAICc, and Akaike weight w_i for each candidate model. Rank them by AICc.

```
make_aicc_table <- function(models_fitted, n_aicc) {
  models_fitted |>
    mutate(
      logLik      = map_dbl(model, ~ as.numeric(logLik(.x$mr))),
      K           = map_int(model, ~ attr(logLik(.x$mr), "df")),
      AIC         = -2 * logLik + 2 * K,
      delta_AIC   = AIC - min(AIC, na.rm = TRUE),
      AICc        = AIC + (2 * K * (K + 1)) / (n_aicc - K - 1),
      delta_AICc  = AICc - min(AICc, na.rm = TRUE),
      weight_AICc = {
        rel_lik <- exp(-0.5 * delta_AICc)
        rel_lik / sum(rel_lik, na.rm = TRUE)
      }
    ) |>
    arrange(AICc)
}

aicc_table <- make_aicc_table(models_fitted, n_aicc = 137)

aicc_table |>
  select(mr_covariate, logLik, K, AICc, delta_AICc, weight_AICc) |>
  knitr::kable(
    digits = c(NA, 3, 0, 3, 3, 5),
    booktabs = TRUE,
    longtable = TRUE,
    escape = FALSE,
    col.names = c(
      "MR covariate",
      "logLik",
```

```
      "K",
      "AICc",
      "$\\Delta$AICc",
      "$w_i$"
    )
  )
```

MR covariate	logLik	K	AICc	ΔAICc	w_i
CLOUD_CODE2	-61.914	4	132.131	0.000	0.09357
SPCODE2 + CLOUD_CODE2	-61.555	5	133.567	1.436	0.04563
size2 + CLOUD_CODE2	-61.675	5	133.809	1.677	0.04045
CLOUD_CODE2 + VISIBILTY_NM	-61.749	5	133.956	1.825	0.03757
CLOUD_CODE2 + GLARE_R	-61.851	5	134.161	2.030	0.03392
BEAUFORT + CLOUD_CODE2	-61.900	5	134.258	2.127	0.03231
CLOUD_CODE	-61.911	5	134.280	2.149	0.03195
CLOUD_CODE2 + QUALITY_R2	-61.913	5	134.285	2.154	0.03188
SPCODE2 + size2 + CLOUD_CODE2	-61.283	6	135.213	3.082	0.02004
size3 + CLOUD_CODE2	-61.346	6	135.338	3.207	0.01883
SPCODE3 + CLOUD_CODE2	-61.404	6	135.453	3.322	0.01777
SPCODE2 + CLOUD_CODE2 + VISIBILTY_NM	-61.489	6	135.625	3.494	0.01631
SPCODE2 + BEAUFORT + CLOUD_CODE2	-61.518	6	135.681	3.550	0.01586
SPCODE2 + CLOUD_CODE	-61.526	6	135.699	3.567	0.01572
SPCODE2 + CLOUD_CODE2 + GLARE_R	-61.533	6	135.713	3.582	0.01561
size2 + CLOUD_CODE2 + VISIBILTY_NM	-61.550	6	135.746	3.615	0.01535
SPCODE2 + CLOUD_CODE2 + QUALITY_R2	-61.550	6	135.747	3.615	0.01535
size2 + CLOUD_CODE2 + GLARE_R	-61.565	6	135.775	3.644	0.01513
size2 + BEAUFORT + CLOUD_CODE2	-61.643	6	135.931	3.800	0.01399
size2 + CLOUD_CODE	-61.667	6	135.981	3.849	0.01365
size2 + CLOUD_CODE2 + QUALITY_R2	-61.669	6	135.984	3.853	0.01363
CLOUD_CODE2 + GLARE_R + VISIBILTY_NM	-61.703	6	136.051	3.920	0.01318
CLOUD_CODE2 + GLARE_R2	-61.733	6	136.113	3.982	0.01278
BEAUFORT + CLOUD_CODE2 + VISIBILTY_NM	-61.740	6	136.127	3.996	0.01269
CLOUD_CODE2 + QUALITY_R2 + VISIBILTY_NM	-61.748	6	136.141	4.010	0.01260
CLOUD_CODE + VISIBILTY_NM	-61.749	6	136.144	4.013	0.01258
BEAUFORT2 + CLOUD_CODE2	-61.769	6	136.184	4.052	0.01233
BEAUFORT + CLOUD_CODE2 + GLARE_R	-61.810	6	136.266	4.134	0.01184
CLOUD_CODE + GLARE_R	-61.841	6	136.329	4.197	0.01147
CLOUD_CODE2 + GLARE_R + QUALITY_R2	-61.845	6	136.337	4.205	0.01143
BEAUFORT + CLOUD_CODE	-61.899	6	136.444	4.312	0.01083
BEAUFORT + CLOUD_CODE2 + QUALITY_R2	-61.900	6	136.446	4.315	0.01082
CLOUD_CODE + QUALITY_R2	-61.910	6	136.467	4.336	0.01071
SPCODE2 + size3 + CLOUD_CODE2	-60.989	7	136.846	4.714	0.00886
SPCODE3 + size2 + CLOUD_CODE2	-61.122	7	137.112	4.981	0.00775
SPCODE2 + size2 + CLOUD_CODE	-61.234	7	137.336	5.205	0.00693
size3 + CLOUD_CODE2 + GLARE_R	-61.260	7	137.389	5.257	0.00675

MR covariate	logLik	K	AICc	ΔAICc	w_i
size3 + CLOUD_CODE2 + VISIBILTY_NM	-61.274	7	137.416	5.285	0.00666
SPCODE3 + CLOUD_CODE2 + VISIBILTY_NM	-61.301	7	137.471	5.340	0.00648
size3 + BEAUFORT + CLOUD_CODE2	-61.303	7	137.475	5.344	0.00647
size3 + CLOUD_CODE	-61.342	7	137.553	5.421	0.00622
size3 + CLOUD_CODE2 + QUALITY_R2	-61.345	7	137.557	5.426	0.00621
SPCODE3 + BEAUFORT + CLOUD_CODE2	-61.359	7	137.587	5.456	0.00611
SPCODE3 + CLOUD_CODE	-61.363	7	137.594	5.462	0.00609
SPCODE2 + BEAUFORT2 + CLOUD_CODE2	-61.367	7	137.603	5.471	0.00607
SPCODE3 + CLOUD_CODE2 + GLARE_R	-61.380	7	137.629	5.497	0.00599
SPCODE3 + CLOUD_CODE2 + QUALITY_R2	-61.395	7	137.658	5.527	0.00590
SPCODE2 + CLOUD_CODE2 + GLARE_R2	-61.439	7	137.746	5.615	0.00565
size2 + CLOUD_CODE2 + GLARE_R2	-61.474	7	137.817	5.686	0.00545
SPCODE2 + CLOUD_CODE + VISIBILTY_NM	-61.475	7	137.819	5.688	0.00545
SPCODE2 + CLOUD_CODE + GLARE_R	-61.496	7	137.861	5.730	0.00533
SPCODE2 + BEAUFORT + CLOUD_CODE	-61.498	7	137.865	5.733	0.00532
SPCODE2 + CLOUD_CODE + QUALITY_R2	-61.521	7	137.911	5.780	0.00520
size2 + CLOUD_CODE + GLARE_R	-61.541	7	137.949	5.818	0.00510
size2 + CLOUD_CODE + VISIBILTY_NM	-61.549	7	137.967	5.836	0.00506
size2 + BEAUFORT2 + CLOUD_CODE2	-61.567	7	138.003	5.872	0.00497
BEAUFORT2 + CLOUD_CODE2 + GLARE_R	-61.574	7	138.017	5.885	0.00493
BEAUFORT2 + CLOUD_CODE2 + VISIBILTY_NM	-61.579	7	138.026	5.895	0.00491
size2 + BEAUFORT + CLOUD_CODE	-61.639	7	138.146	6.015	0.00462
CLOUD_CODE2 + GLARE_R2 + VISIBILTY_NM	-61.654	7	138.176	6.045	0.00456
size2 + CLOUD_CODE + QUALITY_R2	-61.661	7	138.190	6.058	0.00452
BEAUFORT + CLOUD_CODE2 + GLARE_R2	-61.696	7	138.260	6.128	0.00437
CLOUD_CODE + GLARE_R + VISIBILTY_NM	-61.701	7	138.271	6.140	0.00434
CLOUD_CODE + GLARE_R2	-61.728	7	138.324	6.193	0.00423
CLOUD_CODE2 + GLARE_R2 + QUALITY_R2	-61.733	7	138.335	6.203	0.00421
BEAUFORT + CLOUD_CODE + VISIBILTY_NM	-61.740	7	138.348	6.216	0.00418
CLOUD_CODE + QUALITY_R2 + VISIBILTY_NM	-61.747	7	138.363	6.232	0.00415
BEAUFORT2 + CLOUD_CODE	-61.768	7	138.405	6.273	0.00406
BEAUFORT2 + CLOUD_CODE2 + QUALITY_R2	-61.768	7	138.405	6.274	0.00406
BEAUFORT + CLOUD_CODE + GLARE_R	-61.804	7	138.476	6.345	0.00392
CLOUD_CODE + GLARE_R + QUALITY_R2	-61.835	7	138.537	6.406	0.00380
BEAUFORT + CLOUD_CODE + QUALITY_R2	-61.899	7	138.666	6.534	0.00357
SPCODE3 + size3 + CLOUD_CODE2	-60.800	8	138.725	6.594	0.00346
SPCODE2 + size3 + CLOUD_CODE	-60.952	8	139.028	6.897	0.00297
SPCODE3 + size2 + CLOUD_CODE	-61.056	8	139.237	7.105	0.00268
NA	-67.605	2	139.300	7.168	0.00260
size3 + CLOUD_CODE2 + GLARE_R2	-61.160	8	139.446	7.315	0.00241
SPCODE3 + BEAUFORT2 + CLOUD_CODE2	-61.240	8	139.605	7.473	0.00223

MR covariate	logLik	K	AICc	ΔAICc	w_i
size3 + BEAUFORT2 + CLOUD_CODE2	-61.244	8	139.613	7.482	0.00222
size3 + CLOUD_CODE + GLARE_R	-61.246	8	139.617	7.486	0.00222
size3 + CLOUD_CODE + VISIBILTY_NM	-61.274	8	139.672	7.541	0.00216
SPCODE3 + CLOUD_CODE + VISIBILTY_NM	-61.280	8	139.685	7.554	0.00214
SPCODE3 + CLOUD_CODE2 + GLARE_R2	-61.298	8	139.720	7.589	0.00210
size3 + BEAUFORT + CLOUD_CODE	-61.303	8	139.731	7.599	0.00209
SPCODE3 + CLOUD_CODE + GLARE_R	-61.328	8	139.781	7.650	0.00204
SPCODE3 + BEAUFORT + CLOUD_CODE	-61.330	8	139.785	7.654	0.00204
size3 + CLOUD_CODE + QUALITY_R2	-61.341	8	139.807	7.676	0.00202
BEAUFORT2 + CLOUD_CODE2 + GLARE_R2	-61.350	8	139.825	7.693	0.00200
SPCODE3 + CLOUD_CODE + QUALITY_R2	-61.354	8	139.832	7.701	0.00199
SPCODE2 + BEAUFORT2 + CLOUD_CODE	-61.358	8	139.842	7.711	0.00198
SPCODE2 + CLOUD_CODE + GLARE_R2	-61.413	8	139.950	7.819	0.00188
size2 + CLOUD_CODE + GLARE_R2	-61.458	8	140.042	7.910	0.00179
size2	-66.967	3	140.115	7.983	0.00173
size2 + BEAUFORT2 + CLOUD_CODE	-61.567	8	140.259	8.127	0.00161
BEAUFORT2 + CLOUD_CODE + VISIBILTY_NM	-61.568	8	140.261	8.129	0.00161
BEAUFORT2 + CLOUD_CODE + GLARE_R	-61.574	8	140.272	8.141	0.00160
CLOUD_CODE + GLARE_R2 + VISIBILTY_NM	-61.653	8	140.431	8.300	0.00148
BEAUFORT + CLOUD_CODE + GLARE_R2	-61.693	8	140.511	8.380	0.00142
CLOUD_CODE + GLARE_R2 + QUALITY_R2	-61.728	8	140.580	8.449	0.00137
BEAUFORT2 + CLOUD_CODE + QUALITY_R2	-61.768	8	140.661	8.530	0.00132
GLARE_R	-67.252	3	140.685	8.554	0.00130
SPCODE2	-67.331	3	140.843	8.712	0.00120
SPCODE3 + size3 + CLOUD_CODE	-60.747	9	140.911	8.779	0.00116
BEAUFORT	-67.503	3	141.186	9.055	0.00101
size3	-66.470	4	141.242	9.111	0.00098
size2 + GLARE_R	-66.527	4	141.356	9.225	0.00093
QUALITY_R2	-67.592	3	141.365	9.233	0.00092
VISIBILTY_NM	-67.602	3	141.384	9.252	0.00092
SPCODE2 + size2	-66.665	4	141.634	9.502	0.00081
size3 + CLOUD_CODE + GLARE_R2	-61.153	9	141.723	9.591	0.00077
SPCODE3 + BEAUFORT2 + CLOUD_CODE	-61.223	9	141.862	9.731	0.00072
size3 + BEAUFORT2 + CLOUD_CODE	-61.244	9	141.906	9.774	0.00071
SPCODE3 + CLOUD_CODE + GLARE_R2	-61.257	9	141.932	9.801	0.00070
BEAUFORT2 + CLOUD_CODE + GLARE_R2	-61.349	9	142.115	9.984	0.00064
size2 + BEAUFORT	-66.927	4	142.157	10.026	0.00062
size2 + VISIBILTY_NM	-66.945	4	142.193	10.062	0.00061
size2 + QUALITY_R2	-66.965	4	142.233	10.102	0.00060
BEAUFORT2	-66.990	4	142.284	10.152	0.00058
SPCODE2 + GLARE_R	-67.052	4	142.407	10.276	0.00055
size3 + GLARE_R	-66.096	5	142.650	10.519	0.00049
GLARE_R + QUALITY_R2	-67.174	4	142.652	10.521	0.00049
SPCODE3	-67.185	4	142.672	10.541	0.00048
BEAUFORT + GLARE_R	-67.233	4	142.769	10.637	0.00046

MR covariate	logLik	K	AICc	ΔAICc	w_i
GLARE_R + VISIBILTY_NM	-67.241	4	142.785	10.653	0.00045
SPCODE2 + size3	-66.163	5	142.785	10.653	0.00045
GLARE_R2	-67.244	4	142.791	10.660	0.00045
SPCODE2 + BEAUFORT	-67.257	4	142.816	10.685	0.00045
SPCODE2 + VISIBILTY_NM	-67.296	4	142.896	10.765	0.00043
SPCODE2 + QUALITY_R2	-67.316	4	142.934	10.803	0.00042
SPCODE2 + size2 + GLARE_R	-66.305	5	143.069	10.937	0.00039
BEAUFORT + QUALITY_R2	-67.455	4	143.213	11.082	0.00037
size3 + VISIBILTY_NM	-66.406	5	143.269	11.138	0.00036
BEAUFORT + VISIBILTY_NM	-67.502	4	143.307	11.176	0.00035
size3 + BEAUFORT	-66.437	5	143.332	11.201	0.00035
size3 + QUALITY_R2	-66.469	5	143.397	11.265	0.00033
SPCODE3 + size2	-66.483	5	143.424	11.293	0.00033
size2 + GLARE_R + VISIBILTY_NM	-66.486	5	143.430	11.299	0.00033
size2 + BEAUFORT2	-66.503	5	143.465	11.333	0.00032
size2 + GLARE_R + QUALITY_R2	-66.508	5	143.475	11.343	0.00032
QUALITY_R2 + VISIBILTY_NM	-67.587	4	143.477	11.346	0.00032
size2 + GLARE_R2	-66.525	5	143.508	11.377	0.00032
size2 + BEAUFORT + GLARE_R	-66.526	5	143.510	11.379	0.00032
SPCODE2 + size2 + VISIBILTY_NM	-66.585	5	143.628	11.496	0.00030
SPCODE2 + size2 + BEAUFORT	-66.640	5	143.738	11.606	0.00028
SPCODE2 + size2 + QUALITY_R2	-66.664	5	143.785	11.654	0.00028
BEAUFORT2 + GLARE_R	-66.709	5	143.877	11.745	0.00026
SPCODE2 + BEAUFORT2	-66.809	5	144.076	11.945	0.00024
SPCODE3 + GLARE_R	-66.902	5	144.262	12.131	0.00022
size2 + BEAUFORT + VISIBILTY_NM	-66.912	5	144.282	12.151	0.00022
size2 + BEAUFORT + QUALITY_R2	-66.926	5	144.310	12.179	0.00021
BEAUFORT2 + QUALITY_R2	-66.941	5	144.340	12.208	0.00021
size2 + QUALITY_R2 + VISIBILTY_NM	-66.944	5	144.346	12.215	0.00021
SPCODE2 + size3 + GLARE_R	-65.867	6	144.380	12.249	0.00020
SPCODE2 + GLARE_R + QUALITY_R2	-66.980	5	144.419	12.287	0.00020
BEAUFORT2 + VISIBILTY_NM	-66.990	5	144.439	12.307	0.00020
SPCODE2 + GLARE_R + VISIBILTY_NM	-67.005	5	144.469	12.337	0.00020
SPCODE3 + size3	-65.916	6	144.478	12.347	0.00019
SPCODE2 + BEAUFORT + GLARE_R	-67.039	5	144.537	12.406	0.00019
SPCODE2 + GLARE_R2	-67.048	5	144.553	12.422	0.00019
size3 + GLARE_R + VISIBILTY_NM	-66.010	6	144.666	12.535	0.00018
SPCODE2 + size3 + VISIBILTY_NM	-66.011	6	144.669	12.538	0.00018
size3 + BEAUFORT2	-66.015	6	144.676	12.545	0.00018
SPCODE3 + BEAUFORT	-67.119	5	144.696	12.564	0.00017
BEAUFORT + GLARE_R + QUALITY_R2	-67.131	5	144.719	12.588	0.00017
GLARE_R + QUALITY_R2 + VISIBILTY_NM	-67.154	5	144.766	12.634	0.00017
size3 + GLARE_R + QUALITY_R2	-66.064	6	144.775	12.643	0.00017
SPCODE3 + QUALITY_R2	-67.164	5	144.786	12.655	0.00017
SPCODE3 + VISIBILTY_NM	-67.167	5	144.793	12.661	0.00017
GLARE_R2 + QUALITY_R2	-67.171	5	144.801	12.670	0.00017
size3 + GLARE_R2	-66.092	6	144.831	12.700	0.00016
size3 + BEAUFORT + GLARE_R	-66.095	6	144.837	12.706	0.00016
SPCODE3 + size2 + GLARE_R	-66.106	6	144.858	12.727	0.00016
SPCODE2 + BEAUFORT + QUALITY_R2	-67.210	5	144.879	12.747	0.00016

MR covariate	logLik	K	AICc	ΔAICc	w_i
size2 + BEAUFORT2 + GLARE_R	-66.124	6	144.895	12.764	0.00016
GLARE_R2 + VISIBILTY_NM	-67.223	5	144.903	12.772	0.00016
BEAUFORT + GLARE_R2	-67.224	5	144.906	12.774	0.00016
BEAUFORT + GLARE_R + VISIBILTY_NM	-67.224	5	144.907	12.776	0.00016
SPCODE2 + BEAUFORT + VISIBILTY_NM	-67.233	5	144.923	12.792	0.00016
SPCODE2 + size3 + BEAUFORT	-66.144	6	144.934	12.802	0.00016
SPCODE2 + size3 + QUALITY_R2	-66.163	6	144.971	12.840	0.00015
SPCODE2 + QUALITY_R2 + VISIBILTY_NM	-67.274	5	145.005	12.874	0.00015
SPCODE2 + size2 + BEAUFORT2	-66.266	6	145.178	13.047	0.00014
SPCODE2 + size2 + GLARE_R2	-66.305	6	145.256	13.125	0.00013
BEAUFORT + QUALITY_R2 + VISIBILTY_NM	-67.453	5	145.364	13.232	0.00013
size3 + BEAUFORT + VISIBILTY_NM	-66.386	6	145.417	13.286	0.00012
size3 + QUALITY_R2 + VISIBILTY_NM	-66.403	6	145.453	13.322	0.00012
size3 + BEAUFORT + QUALITY_R2	-66.429	6	145.504	13.373	0.00012
SPCODE3 + size2 + VISIBILTY_NM	-66.433	6	145.512	13.380	0.00012
SPCODE3 + size2 + BEAUFORT	-66.465	6	145.576	13.444	0.00011
size2 + GLARE_R2 + VISIBILTY_NM	-66.473	6	145.591	13.460	0.00011
SPCODE3 + size2 + QUALITY_R2	-66.483	6	145.612	13.480	0.00011
size2 + BEAUFORT2 + VISIBILTY_NM	-66.499	6	145.644	13.513	0.00011
size2 + BEAUFORT2 + QUALITY_R2	-66.499	6	145.645	13.514	0.00011
size2 + GLARE_R2 + QUALITY_R2	-66.508	6	145.662	13.530	0.00011
size2 + BEAUFORT + GLARE_R2	-66.525	6	145.695	13.564	0.00011
SPCODE2 + BEAUFORT2 + GLARE_R	-66.557	6	145.760	13.629	0.00010
BEAUFORT2 + GLARE_R + QUALITY_R2	-66.595	6	145.836	13.705	0.00010
BEAUFORT2 + GLARE_R2	-66.669	6	145.985	13.854	0.00009
BEAUFORT2 + GLARE_R + VISIBILTY_NM	-66.707	6	146.061	13.930	0.00009
SPCODE3 + size3 + GLARE_R	-65.612	7	146.092	13.960	0.00009
SPCODE3 + BEAUFORT2	-66.739	6	146.124	13.993	0.00009
SPCODE2 + BEAUFORT2 + QUALITY_R2	-66.763	6	146.172	14.041	0.00008
size3 + BEAUFORT2 + GLARE_R	-65.686	7	146.239	14.108	0.00008
SPCODE2 + BEAUFORT2 + VISIBILTY_NM	-66.799	6	146.244	14.113	0.00008
SPCODE3 + GLARE_R + QUALITY_R2	-66.818	6	146.282	14.151	0.00008
SPCODE2 + size3 + BEAUFORT2	-65.761	7	146.390	14.259	0.00007
SPCODE3 + GLARE_R + VISIBILTY_NM	-66.875	6	146.397	14.265	0.00007
SPCODE3 + BEAUFORT + GLARE_R	-66.892	6	146.431	14.299	0.00007
SPCODE3 + GLARE_R2	-66.900	6	146.446	14.315	0.00007
SPCODE3 + size3 + VISIBILTY_NM	-65.810	7	146.488	14.357	0.00007
BEAUFORT2 + QUALITY_R2 + VISIBILTY_NM	-66.940	6	146.527	14.396	0.00007
SPCODE2 + size3 + GLARE_R2	-65.866	7	146.599	14.468	0.00007
SPCODE2 + GLARE_R2 + QUALITY_R2	-66.979	6	146.605	14.473	0.00007
SPCODE2 + GLARE_R2 + VISIBILTY_NM	-66.984	6	146.614	14.483	0.00007
SPCODE3 + size3 + BEAUFORT	-65.904	7	146.676	14.545	0.00006
SPCODE3 + size3 + QUALITY_R2	-65.913	7	146.694	14.563	0.00006
SPCODE2 + BEAUFORT + GLARE_R2	-67.034	6	146.715	14.583	0.00006
SPCODE3 + BEAUFORT + QUALITY_R2	-67.067	6	146.780	14.649	0.00006
size3 + GLARE_R2 + VISIBILTY_NM	-65.978	7	146.824	14.693	0.00006
size3 + BEAUFORT2 + VISIBILTY_NM	-65.993	7	146.854	14.722	0.00006
SPCODE3 + BEAUFORT + VISIBILTY_NM	-67.108	6	146.862	14.731	0.00006
size3 + BEAUFORT2 + QUALITY_R2	-65.999	7	146.867	14.736	0.00006

MR covariate	logLik	K	AICc	ΔAICc	w_i
BEAUFORT + GLARE_R2 + QUALITY_R2	-67.128	6	146.902	14.770	0.00006
SPCODE3 + QUALITY_R2 + VISIBILTY_NM	-67.141	6	146.928	14.797	0.00006
GLARE_R2 + QUALITY_R2 + VISIBILTY_NM	-67.143	6	146.931	14.800	0.00006
size3 + GLARE_R2 + QUALITY_R2	-66.063	7	146.994	14.862	0.00006
size3 + BEAUFORT + GLARE_R2	-66.092	7	147.052	14.921	0.00005
BEAUFORT + GLARE_R2 + VISIBILTY_NM	-67.207	6	147.059	14.928	0.00005
size2 + BEAUFORT2 + GLARE_R2	-66.104	7	147.076	14.944	0.00005
SPCODE3 + size2 + GLARE_R2	-66.106	7	147.080	14.948	0.00005
SPCODE3 + size2 + BEAUFORT2	-66.172	7	147.212	15.081	0.00005
SPCODE3 + BEAUFORT2 + GLARE_R	-66.480	7	147.829	15.698	0.00004
SPCODE2 + BEAUFORT2 + GLARE_R2	-66.525	7	147.919	15.787	0.00003
BEAUFORT2 + GLARE_R2 + QUALITY_R2	-66.569	7	148.006	15.875	0.00003
BEAUFORT2 + GLARE_R2 + VISIBILTY_NM	-66.655	7	148.178	16.047	0.00003
SPCODE3 + BEAUFORT2 + QUALITY_R2	-66.689	7	148.245	16.114	0.00003
SPCODE3 + BEAUFORT2 + VISIBILTY_NM	-66.735	7	148.338	16.206	0.00003
SPCODE3 + size3 + GLARE_R2	-65.611	8	148.347	16.216	0.00003
SPCODE3 + size3 + BEAUFORT2	-65.614	8	148.352	16.221	0.00003
size3 + BEAUFORT2 + GLARE_R2	-65.657	8	148.439	16.308	0.00003
SPCODE3 + GLARE_R2 + QUALITY_R2	-66.818	7	148.504	16.373	0.00003
SPCODE3 + GLARE_R2 + VISIBILTY_NM	-66.863	7	148.595	16.464	0.00002
SPCODE3 + BEAUFORT + GLARE_R2	-66.890	7	148.648	16.516	0.00002
SPCODE3 + BEAUFORT2 + GLARE_R2	-66.456	8	150.036	17.905	0.00001

5 Discussion

5.1 Model evaluation

In this analysis, we explored models built from a set of covariates known *a priori* to affect aerial survey detectability. Because the mark-recapture sample size was limited (137 trials), we selected the simplest well-supported model and treated additional covariates in near-top models as potentially uninformative unless they produced meaningful likelihood improvement.

Single-covariate models that included either `CLOUD_CODE` or `CLOUD_CODE2` yielded substantially better log-likelihoods and AICc than single-covariate models for other covariates. When multi-covariate models were included in the ranking, the top 94 included either `CLOUD_CODE` or `CLOUD_CODE2`. The 95th-ranked model—the highest ranked one that did not include `CLOUD_CODE` or `CLOUD_CODE2`—did not include any covariates. This further indicates that the other covariates are relatively weak correlates with perception bias, and the improvement gained by including them in the model is not worth the added complexity, according to AICc's conception of this tradeoff.

The top model contained only `CLOUD_CODE2`. Adding other covariates produced little additional support and did not justify the extra parameters, so we selected the top model as our final choice.

```
summary(aicc_table$model[[1]]$mr)
```

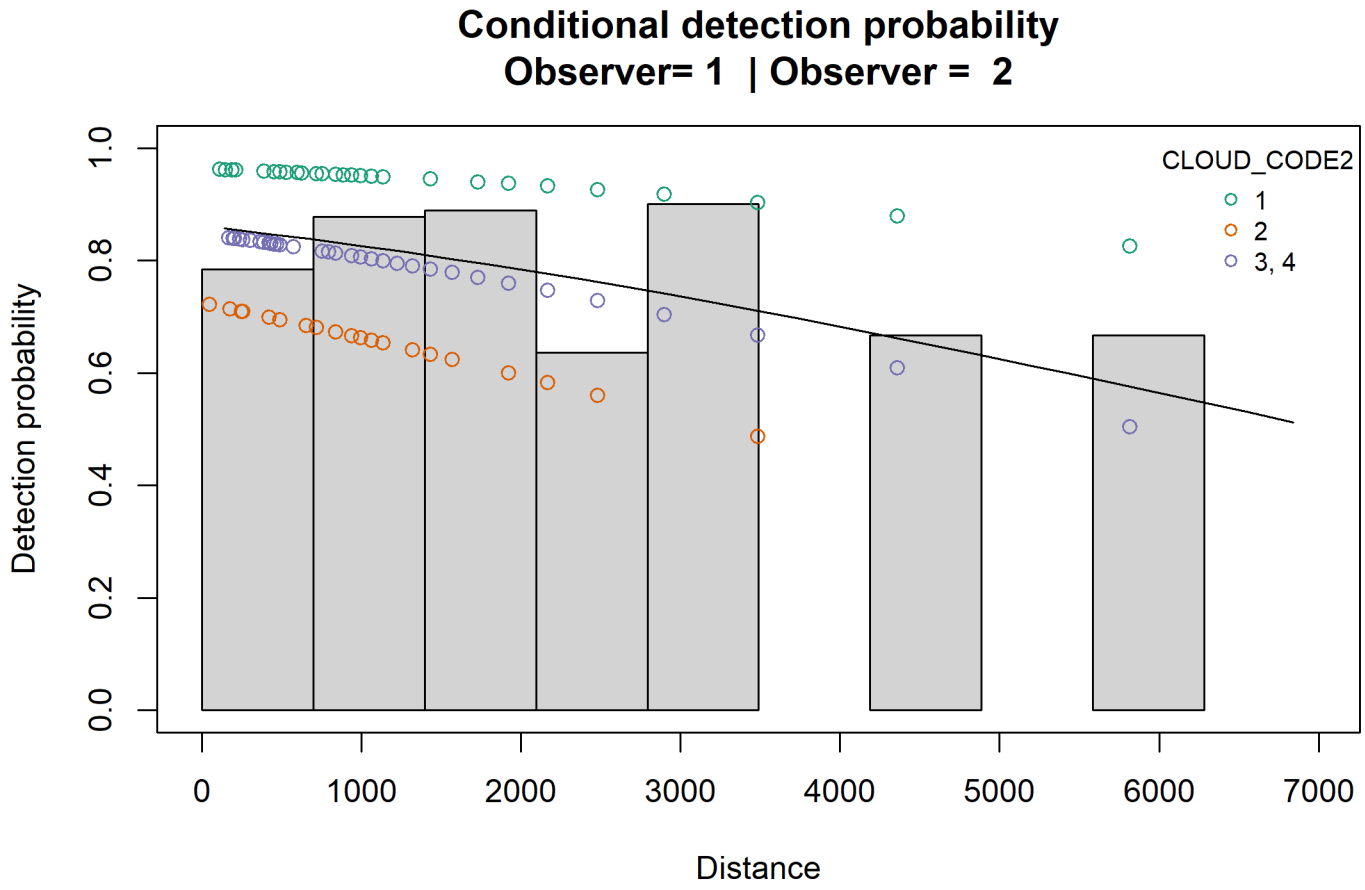
```
Summary for trial.fi object
Number of observations              :  185
Number seen by primary              :  158
```

```
Number seen by secondary (trials)    :  137
Number seen by both (detected trials):  110
AIC                                  :  131.8282


Conditional detection function parameters:
                     estimate          se
(Intercept)      3.2500834051 0.746982109
distance        -0.0002916144 0.000181621
CLOUD_CODE22    -2.2856218756 0.767919399
CLOUD_CODE23, 4 -1.5384796222 0.682851940

                         Estimate          SE        CV
Average p              0.6770680  0.09303408 0.1374073
Average primary p(0)   0.8473943  0.11661587 0.1376170
N in covered region  233.3591140 33.84149597 0.1450190
```

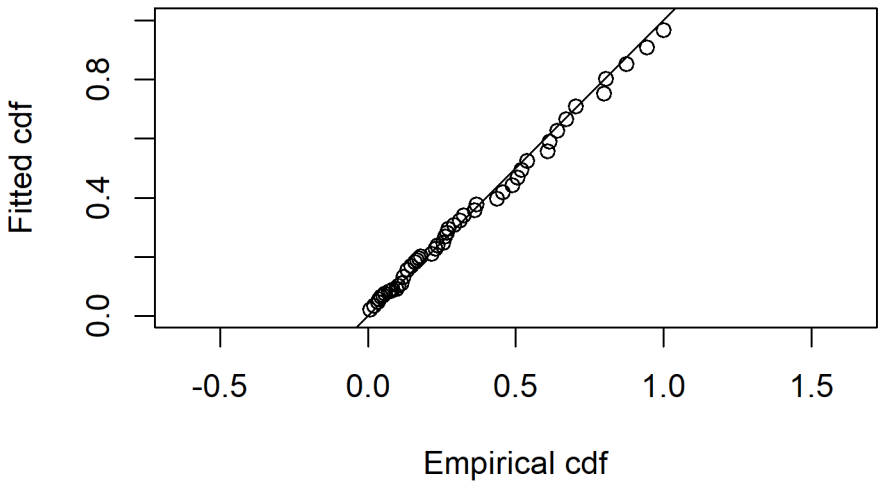
```r
plot(aicc_table$model[[1]], which = 2)
```



As expected, the `distance` covariate was statistically significant in all models and had a negative effect on detection probability. In other words, the chance of Observer 1 also making a sighting seen by Observer 2 decreased with distance from the trackline. At best, we would expect that this relationship be flat but not increasing, and the finding that it is decreasing simply reflects the expected outcome that distant animals are harder to detect than nearby animals.

We did not reserve a holdout data set to evaluate our selected model because we only had 137 trials. Instead, we evaluated the selected model using standard MRDS goodness-of-fit diagnostics implemented in `ddf.gof()`, including a Q-Q plot and chi-square and Cramér–von Mises tests. These diagnostics evaluate whether the selected model reproduces the observed detection process, while AICc provides the model-selection criterion penalizing additional parameters.

```
mrds::ddf.gof(
  aicc_table$model[[1]],
  breaks = c(0, 1000, 2000, 3000, 4000, 5000, 6000, meta.data$width),
  qq = TRUE
)
```




```
Goodness of fit results for ddf object

Chi-square tests

Distance sampling component:
          [0,1e+03] (1e+03,2e+03] (2e+03,3e+03] (3e+03,4e+03] (4e+03,5e+03]
Observed     72.000        34.000        21.000        11.000        11.000
Expected     66.357        41.132        20.767        12.112         7.922
Chisquare     0.480         1.237         0.003         0.102         1.196
          (5e+03,6e+03] (6e+03,6.98e+03]   Total
Observed          9.000            0.000 158.000
Expected          5.599            4.109 158.000
Chisquare         2.065            4.109   9.192

P = 0.056473 with 4 degrees of freedom

Mark-recapture component:
Capture History 01
          [0,1e+03] (1e+03,2e+03] (2e+03,3e+03] (3e+03,4e+03] (4e+03,5e+03]
Observed         10             5             9             0             2
```

```
Expected         10             7             5             2             2
Chisquare         0             1             3             2             0
          (5e+03,6e+03] (6e+03,6.98e+03] Total
Observed              1                0    27
Expected              1               NA    NA
Chisquare             0               NA    NA
Capture History 11
          [0,1e+03] (1e+03,2e+03] (2e+03,3e+03] (3e+03,4e+03] (4e+03,5e+03]
Observed         53            28            15             8             4
Expected         53            26            19             6             4
Chisquare         0             0             1             1             0
          (5e+03,6e+03] (6e+03,6.98e+03] Total
Observed              2                0   110
Expected              2               NA    NA
Chisquare             0               NA    NA

MR total chi-square = 7.2049  P = 0.065645 with 3 degrees of freedom


Total chi-square = 16.397  P = 0.021726 with 7 degrees of freedom

Distance sampling Cramer-von Mises test (unweighted)
Test statistic = 0.0769848 p-value = 0.709476
```

The Q-Q plot showed relatively minor offsets from the diagonal and the Cramér–von Mises test was not significant. Neither the chi-square test for the distance sampling model nor the mark-recapture model were significant at the 0.05 significance level. Together these results indicate an acceptable goodness of fit. However, we caution that the total chi-square test was significant, and that different binnings (e.g. fewer but wider bins) yielded significant results for either the distance sampling model or the mark-recapture model. This sensitivity is likely a side effect of the small sample size (137 trials), and confidence in this outcome could be improved by collecting more data and repeating the analysis.

When the NARWSS team reviewed the relationship fitted between `CLOUD_CODE2` and perception bias, they judged it appropriate, as discussed in the next section.

5.2 Final `p(0)` estimates

The final model yielded a mean perception bias estimate of `p(0) = 0.847 (CV = 0.117)`, when averaged across all of the data used to fit the model. This does not differ significantly from the mean estimate I recently obtained for the Canada DFO NARW program of `p(0) = 0.846 (CV = 0.121)`, developed for a similar number of trials (n = 117) collected by Cessna 337s and a Twin Otter at 800 ft. altitude. The DFO model used group size, Beaufort sea state, and surface-active group (SAG) behavior as covariates. (Each of those covariates was binned into just two categories, as the number of trials was not large enough to discriminate fine differences between the full ranges of covariate values.)

In the NARW v12 density model, we used perception bias corrections derived by Palka et al. (2021) from data collected by the AMAPPS program. For all aerial surveys other than the NEFSC AMAPPS and pre-AMAPPS marine mammal abundance surveys, we used the SEFSC AMAPPS estimate of `p(0) = 0.86`. As discussed in the supplementary report of Roberts et al. (2024), we preferred the SEFSC estimate over the lower NEFSC estimate of `0.67`, on the basis that a number of other large whale analyses reported or assumed values much closer to `1`. Our new estimate of `0.847` is not significantly different from the SEFSC estimate, which suggests our approach for the v12 density model was reasonable. In any case, we can now replace it with our new, NARWSS-derived estimate.

Because the final mark-recapture model selected here utilized the `CLOUD_CODE2` covariate, it would be ideal to correct the new NARW density model on a per-segment basis according to the value of that covariate:

```
# Select the best ranked model as our final choice.

final_model <- aicc_table$model[[1]]$mr

results <-
  # Create a grid of all possible values of the covariate used in the final model
  expand(
    tibble(),
    CLOUD_CODE2 = unique(detections5$CLOUD_CODE2)
  ) |>

  # Add columns needed by the mrds predict() function
  mutate(
    object = row_number(),
    observer = 1,
    distance = 0
  ) %>%               # Use %>% here, not |>, so the code below can use .

  # Predict the final model on the grid of combos
  mutate(
    p0 = predict(final_model, newdata = .)$p1
  ) |>

  # Count the number of sightings by Observer 1 for each value
  inner_join(
    detections5 |>
      filter(observer == 1 & distance <= 6981) |>
      count(CLOUD_CODE2, .drop = FALSE),
    by = join_by(CLOUD_CODE2)
  ) |>

  # Select just the columns we want to show, and give them friendly names
  transmute(
    CLOUD_CODE = CLOUD_CODE2,
    `p(0)` = p0,
    `Obs 1 Sightings` = n
  )

# Print the results.

knitr::kable(results, digits = 3, booktabs = TRUE)
```

CLOUD_CODE	p(0)	Obs 1 Sightings
1	0.963	60
2	0.724	38
3, 4	0.847	87

The model indicated that for `CLOUD_CODE` 1 (clear, <10% cloud cover), `p(0) = 0.962` was close to 1, meaning

that perception bias was almost negligible when weather was clear. This accorded well with the NARWSS team's belief that they rarely miss large whales in clear weather.

In `CLOUD_CODE` 2 (scattered, 10-50% cloud cover), perception bias was stronger, `p(0) = 0.724`, than in `CLOUD_CODE` 3 (broken, 50-90% cloud cover) or 4 (overcast, >90% cloud cover), `p(0) = 0.847`. This result indicates that having more clouds is better than having less, although not as good as having no clouds. When I reviewed this result with the NARWSS team, they agreed with the interpretation that scattered clouds result in the least uniform ocean surface, with bright sunny patches intermixed with dark shadowed patches, and whales are harder to detect with so much variability in illumination than in the more uniform illumination that occurs in overcast or near-overcast conditions.

5.3 Applicability to other survey programs

The definition of the `CLOUD_CODE` covariate is standardized by the North Atlantic Right Whale Consortium Sightings Database, and several other teams use this definition, including New England Aquarium, FWRI/FWC, and Clearwater Marine Aquarium Research Institute. If we assumed the effect of cloud cover was the same for those programs as for NARWSS, we could apply our final model to them directly.

For programs that did not collect a compatible `CLOUD_CODE` covariate, we could apply the best model that did not utilize this covariate. This was the 95th ranked model, which used no covariates and estimated a uniform `p(0) = 0.847`.