



NOAA Technical Memorandum NMFS-XXX-##

CPS Acoustic Classification

Alice Beittel

January 2023

U.S. DEPARTMENT OF COMMERCE

National Oceanic and Atmospheric
Administration

National Marine Fisheries Service
Northwest Fisheries Science Center



**NOAA
FISHERIES**

CPS Acoustic Classification

Alice Beittel¹

1. NOAA Southwest Fisheries Science Center, Southwest Fisheries Science Center

Table of contents

How do we use sound to count fish?	1
Short and sweet overview of survey	1
Cite This Data	1
NOAA README	2
NOAA License	2
1 Survey Background	3
1.1 GitHub Set-up	3
2 Data Acquisition	4
2.1 Survey Design	4
2.2 Survey Equipment	4
2.3 Software	5
2.4 Raw Acoustic Data Format	6
3 Data Workflow	7
4 Customize	8
4.1 Edit and add your pages	8
4.2 Add your pages the project	8
5 Customization	9
5.1 Quarto documentation	9
5.2 Examples	9
6 Rendering	10
6.1 Step 1. Make sure you have a recent RStudio	10
6.2 Step 2. Clone and create RStudio project	10
6.3 Step 3. Render within RStudio	10

Table of contents

7	Figures and Tables	11
7.1	Code	11
7.2	Including Plots	11
7.3	Including Tables	12
8	Rendering with Code	13
8.1	Modify the GitHub Action	13
8.2	Render locally and publish to gh-pages branch	14
9	References	15
	References	16

List of Figures

2.1	Transducer locations on the bottom of the centerboard aboard Lasker.	4
7.1	Plot of pressure	12

List of Tables

7.1 Iris Data	12
-------------------------	----

How do we use sound to count fish?

Last updated: 2024-11-14 20:28:51 UTC

Document Objective: This resource will serve as a tutorial to demonstrate how the SWFSC uses acoustic data generate biomass estimates of Coastal Pelagic Species from Baja, Mexico to Vancouver, Canada.

As part of our commitment to open science, reproducibility, and transparency, we provide this metadata guide to compliment our public-domain data.

Please consider this resource to be a **Living Document**. The code in this repository is regularly being updated and improved.

Do not hesitate to reach out (to us at either alice.beittel@noaa.gov or GitHub issues, especially if you find discrepancies in the data or want to suggest improvements to infrastructure. Thank you in advance for your collaboration and partnership with us as we develop our future data universe.

Short and sweet overview of survey

The SWFSC's ATM surveys of CPS in the CCE began in 2006 with a focus on the northern stock of Pacific Sardine. Since then, we have expanded in scope and objectives to include the larger forage-fish assemblage and krill. This evolution, and the migratory behavior of Pacific Sardine, serve to explain the present survey region and design.

Cite This Data

[enter text on how to do this]

NOAA README

NOAA README

This repository is a scientific product and is not official communication of the National Oceanic and Atmospheric Administration, or the United States Department of Commerce. All NOAA GitHub project code is provided on an 'as is' basis and the user assumes responsibility for its use. Any claims against the Department of Commerce or Department of Commerce bureaus stemming from the use of this GitHub project will be governed by all applicable Federal law. Any reference to specific commercial products, processes, or services by service mark, trademark, manufacturer, or otherwise, does not constitute or imply their endorsement, recommendation or favoring by the Department of Commerce. The Department of Commerce seal and logo, or the seal and logo of a DOC bureau, shall not be used in any manner to imply endorsement of any commercial product or activity by DOC or the United States Government.

NOAA License

Software code created by U.S. Government employees is not subject to copyright in the United States (17 U.S.C. §105). The United States/Department of Commerce reserve all rights to seek and obtain copyright protection in countries other than the United States for Software authored in its entirety by the Department of Commerce. To this end, the Department of Commerce hereby grants to Recipient a royalty-free, nonexclusive license to use, copy, and create derivative works of the Software outside of the United States.

1 Survey Background

This is a template for a simple Quarto book output to html, PDF or docx format. It includes a GitHub Action that will build the website automatically when you make changes to the files. The NOAA palette and fonts has been added to `theme.scss`. The webpage will be on the `gh-pages` branch. Serving the website files from this branch is a common way to keep all the website files from cluttering your main branch.

The GitHub Action installs R so you can have R code in your `qmd` or `Rmd` files. Note, you do not need to make changes to your `Rmd` files unless you need Quarto features like cross-references.

1.1 GitHub Set-up

- Click the green “use template” button to make a repository with this content. Make sure to make your repo public (since GitHub Pages doesn’t work on private repos unless you have a paid account) and check box to include all the branches (so that you get the `gh-pages` branch).
- Turn on GitHub Pages under Settings > Pages . You will set pages to be made from the `gh-pages` branch and root directory.
- Turn on GitHub Actions under Settings > Actions > General
- Edit the repo description and Readme to add a link to the webpage. When you edit the description, you will see the link url in the url box or you can click on the Actions tab or the Settings > Pages page to find the url.

2 Data Acquisition

2.1 Survey Design

2.2 Survey Equipment

2.2.1 Acoustic Instruments

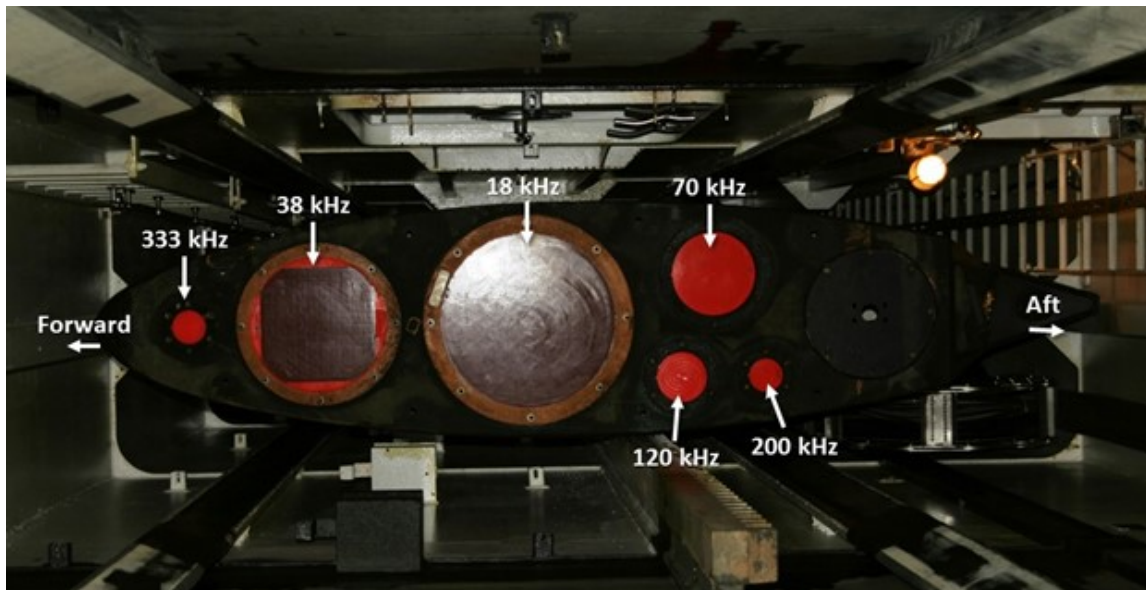


Figure 2.1: Transducer locations on the bottom of the centerboard aboard Lasker.

On *Lasker* and *Shimada*, multi-frequency Wideband Transceivers (Simrad EK80 WBTs; Kongsberg) were configured with split-beam transducers (Simrad ES18, ES38-7, ES70-7C, ES120-7C, ES200-7C, and ES333-7C on *Lasker* and ES18, ES38B, ES70-7C, ES120-7C, and ES200-7C on *Shimada*; Kongsberg). The transducers were mounted on the bottom of a retractable keel or “centerboard”. The keel was retracted (transducers ~5-m

2 Data Acquisition

depth) during calibration, and extended to the intermediate position (transducers ~7-m depth) during the survey. Exceptions were made during shallow water operations, when the keel was retracted; or during times of heavy weather, when the keel was extended (transducers ~9-m depth) to provide extra stability and reduce the effect of weather-generated noise. Transducer position and motion were measured at 5 Hz using an inertial motion unit (Applanix POS-MV; Trimble).

2.2.2 Underway CTD

On *Lasker* and *Shimada*, conductivity and temperature profiles were measured down to 300 m using calibrated sensors on a probe cast from the vessel while underway (UnderwayCTD, or UCTD; Teledyne Ocean- science). Casts were typically conducted between two to four times along each transect. These data indicate the depth of the surface mixed layer, above which most epipelagic CPS reside during the day. These data were also used to estimate the time-averaged sound speed (Demer, 2004), for estimating ranges to the sound scatterers, and frequency-specific sound absorption coefficients, for compensating signal attenuation of the sound pulse between the transducer and scatterers (Simmonds and MacLennan, 2005).

2.3 Software

2.3.1 Echosounder Software

EK80

2.3.2 NetTime

On *Lasker* and *Shimada*, the computer clocks were synchronized with the GPS clock (UTC) using a synchronization software called NetTime.

2.3.3 EAL

The 38-, 70-, 120-, 200-, and 333-kHz echosounders were controlled by the EK80 Adaptive Logger (EAL2, Renfree and Demer, 2016). The EAL optimizes the pulse interval based on the seabed depth, while avoiding aliased seabed echoes, and was programmed such that once an hour the echosounders would record three pings in passive mode, for obtaining estimates of the background noise level.

2.3.4 K Sync

To minimize acoustic interference on *Lasker* and *Shimada*, transmit pulses from the EK80s, acoustic Doppler current profiler and echosounder (Simrad-Kongsberg EC150-3C), multibeam echosounder (Simrad-Kongsberg ME70), imaging sonar (Simrad-Kongsberg MS70), scanning sonar (Simrad-Kongsberg SX90), and a separate acoustic Doppler current profiler (Teledyne RD Instruments OS75 ADCP) were triggered using a synchronization system (Simrad K-Sync; Kongsberg). The K-Sync trigger rate, and thus the echosounder ping interval, was modulated by the EAL using the 18-kHz seabed depth provided by the Scientific Computing System (SCS).

2.4 Raw Acoustic Data Format

Measurements of volume backscattering strength (S_v ; dB re 1 m² m⁻³) and target strength (TS; dB re 1 m²), indexed by time and geographic positions provided by GPS receivers, were stored in Simrad-Kongsberg .raw format with a 1-GB maximum file size. During daytime, the echosounders operated in CW mode and logged to 60 m beyond the detected seabed range or to a maximum range of 500, 500, 500, 300, and 150 m for 38, 70, 120, 200, and 333 kHz, respectively. During nighttime, the echosounders operated in FM mode and logged to 100 m. For each acoustic instrument, the prefix for each file name is a concatenation of the survey name (e.g., 2307RL), the operational mode (CW or FM), and the logging commencement date and time from the EK80 software (v21.15.1). For example, a file generated by the Simrad-Kongsberg EK80 software for a WBT operated in CW mode is named 2307RL_CW-D20220826-T155651.raw.

3 Data Workflow

4 Customize

4.1 Edit and add your pages

Edit the qmd or md files in the `content` folder. qmd files can include code (R, Python, Julia) and lots of Quarto markdown bells and whistles (like call-outs, cross-references, auto-citations and much more).

Each page should start with

```
---  
title: your title  
---
```

and the first header will be the 2nd level, so `##`. Note, there are situations where you leave off

```
---  
title: your title  
---
```

and start the qmd file with a level header `#`, but if using the default title yaml (in the `---` fence) is a good habit since it makes it easy for Quarto convert your qmd file to other formats (like into a presentation).

4.2 Add your pages the project

- Add the files to `_quarto.yml`

5 Customization

5.1 Quarto documentation

Quarto allow many bells and whistles to make nice output. Read the documentation [here](#) Quarto documentation.

5.2 Examples

Looking at other people's Quarto code is a great way to figure out how to do stuff. Most will have a link to a GitHub repo where you can see the raw code. Look for a link to edit page or see source code. This will usually be on the right. Or look for the GitHub icon somewhere.

- [Quarto gallery](#)
- [nmfs-openscapes](#)
- [Faye lab manual](#)
- [quarto-titlepages](#) Note the link to edit is broken. Go to repo and look in `documentation` directory.

6 Rendering

The repo includes a GitHub Action that will render (build) the website automatically when you make changes to the files. It will be pushed to the `gh-pages` branch.

But when you are developing your content, you will want to render it locally.

6.1 Step 1. Make sure you have a recent RStudio

Have you updated RStudio since about August 2022? No? Then update to a newer version of RStudio. In general, you want to keep RStudio updated and it is required to have a recent version to use Quarto.

6.2 Step 2. Clone and create RStudio project

First, clone the repo onto your local computer. How? You can click `File > New Project` and then select “Version Control”. Paste in the url of the repository. That will clone the repo on to your local computer. When you make changes, you will need to push those up.

6.3 Step 3. Render within RStudio

RStudio will recognize that this is a Quarto project by the presence of the `_quarto.yml` file and will see the “Build” tab. Click the “Render website” button to render to the `_site` folder.

Previewing: You can either click `index.html` in the `_site` folder and specify “preview in browser” or set up RStudio to preview to the viewer panel. To do the latter, go to `Tools > Global Options > R Markdown`. Then select “Show output preview in: Viewer panel”.

7 Figures and Tables

Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see <http://rmarkdown.rstudio.com>.

7.1 Code

You can embed an R code chunk like this:

```
summary(cars)
```

speed		dist	
Min.	: 4.0	Min.	: 2.00
1st Qu.:	12.0	1st Qu.:	26.00
Median	:15.0	Median	: 36.00
Mean	:15.4	Mean	: 42.98
3rd Qu.:	19.0	3rd Qu.:	56.00
Max.	:25.0	Max.	:120.00

7.2 Including Plots

You can also embed plots and reference them, like so Figure 7.1.

Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.

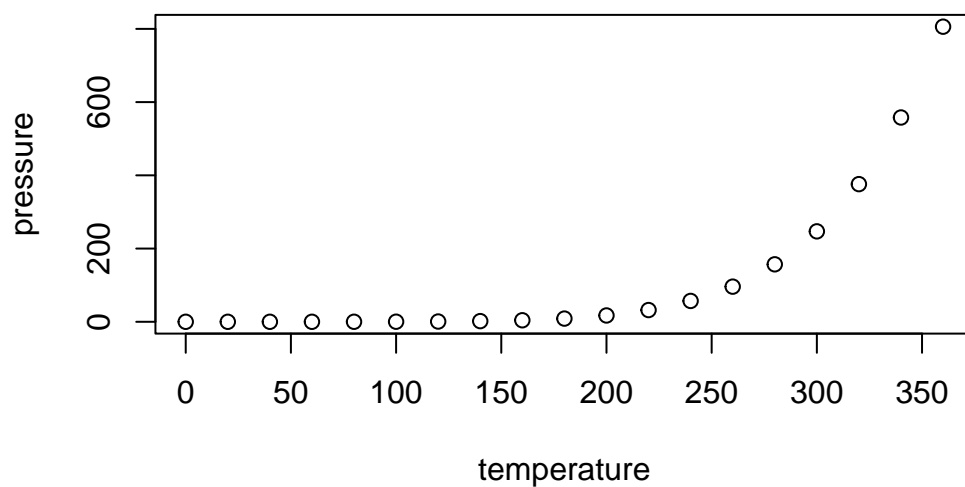


Figure 7.1: Plot of pressure

7.3 Including Tables

You can also embed tables and reference them with Table 7.1.

```
library(knitr)
kable(head(iris))
```

Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
5.1	3.5	1.4	0.2	setosa
4.9	3.0	1.4	0.2	setosa
4.7	3.2	1.3	0.2	setosa
4.6	3.1	1.5	0.2	setosa
5.0	3.6	1.4	0.2	setosa
5.4	3.9	1.7	0.4	setosa

8 Rendering with Code

You can have code (R, Python or Julia) in your qmd file. You will need to have these installed on your local computer, but presumably you do already if you are adding code to your qmd files.

```
x <- c(5, 15, 25, 35, 45, 55)
y <- c(5, 20, 14, 32, 22, 38)
lm(x ~ y)
```

Call:

```
lm(formula = x ~ y)
```

Coefficients:

(Intercept)	y
1.056	1.326

8.1 Modify the GitHub Action

You will need to change the GitHub Action in `.github/workflows` to install these and any needed packages in order for GitHub to be able to render your webpage. The GitHub Action install R since I used that in `code.qmd`. If you use Python or Julia instead, then you will need to update the GitHub Action to install those.

If getting the GitHub Action to work is too much hassle (and that definitely happens), you can always render locally and publish to the `gh-pages` branch. If you do this, make sure to delete or rename the GitHub Action to something like

```
render-and-publish.old.yml
```

so GitHub does not keep trying to run it. Nothing bad will happen if you don't do this, but if you are not using the action (because it keeps failing), then you don't need GitHub to run it.

8.2 Render locally and publish to gh-pages branch

To render locally and push up to the `gh-pages` branch, open a terminal window and then `cd` to the directory with the Quarto project. Type this in the terminal:

```
quarto render gh-pages
```

9 References

Quarto has powerful references functionality. You can easily insert citations from Zotero libraries that you maintain in the cloud (on Zotero). This allows the whole team to update the library and you can sync up to that library. Read about this on the Quarto documentation on citations. Google youtube videos on this also to see it in action.

Add a `.bib` file in to your project or add a linked Zotero library via RStudio in Visual mode with Tools > Project Options... > R Markdown > select custom libraries from the Zotero dropdown.

Then you can type @ and you will see a dropdown of the references in your libraries. You can then select the ones to add. If you don't see the one you need, you can paste in the DOI and it will be added to your references file (with all the info). The references will be added to your references section of your book automatically.

See the `references.qmd` file for how to include the references.

- @ansley1981 will produce Ansley and Davis (1981)
- [@ansley1981] will produce (Ansley and Davis 1981).

References

Ansley, H. L. H., and C. D. Davis. 1981. "Migration and Standing Stock of Fishes Associated with Artificial and Natural Reefs on Georgia's Outer Continental Shelf." Brunswick, Georgia, USA.