



NOAA Technical Memorandum NMFS-XXX-##

GAP Production Data Documentation

Bering Sea Survey Team, Gulf of Alaska and Aleutian Island Survey Team

U.S. DEPARTMENT OF COMMERCE

National Oceanic and Atmospheric Administration
National Marine Fisheries Service
Northwest Fisheries Science Center



**NOAA
FISHERIES**

GAP Production Data Documentation

Bering Sea Survey Team^{1,*} and Gulf of Alaska and Aleutian Island Survey Team^{1,*}

1. NOAA Fisheries Alaska Fisheries Science Center, Groundfish Assessment Program

* Correspondence: Bering Sea Survey Team afsc.gap.metadata@noaa.gov

* Correspondence: Gulf of Alaska and Aleutian Island Survey Team afsc.gap.metadata@noaa.gov

Table of contents

I. Welcome	1
Our Objective	2
User Resources	2
Background of the gap_products repo	2
Major Advantages	4
1. Survey background	6
1.1. What is the research objective?	6
1.2. Who is conducting the research?	6
1.3. Bottom trawl surveys and regions	7
2. Workflow	12
2.1. Data workflow from boat to production	12
2.2. Organization	12
2.3. Data levels	13
3. News	15
3.1. Future plans	15
3.2. Previous updates	15
4. Code of Conduct	17
4.1. What are Codes of Conduct?	17
II. GAP Production Data	19
Data Description	20
Cite this data	20
5. Data description	21
5.1. Data tables	21

Table of contents

III. AKFIN	37
The Alaska Fisheries Information Network	38
Cite this data	38
6. Data description	39
6.1. Other resources	39
6.2. Data description	39
6.3. Data tables	39
7. Access data	76
Access data via Oracle (AFSC only)	76
Data SQL Query Examples:	76
8. Access API data using R	98
8.1. Ex. 1: Load lingcod data	98
IV. Public Data (FOSS)	100
V. Collaborators and data users	102
Access Constraints	103
Cite this data	103
9. Data description	105
9.1. Data tables	106
10. Using the FOSS platform	128
10.1. Select and filter	128
10.2. Select data format	130
10.3. Run report	132
11. Access via API and R	133
11.1. Ex. 1: Load the first 25 rows (default) of data	133
11.2. Ex. 2: Load the first 10000 rows of data	134
11.3. Ex. 3: Filter by Year	134
11.4. Ex. 4: Filter by species name	135
11.5. Ex. 5: Combination of year and name filters	136
11.6. Ex. 6: Combination of year, srvy, stratum	137
11.7. Ex. 7: Visualize CPUE data in distribution map	138
12. Access via API and Python	141

Table of contents

13. Access via Oracle and R (AFSC only)	147
VI. Product Outputs	152
14. Open source code	156
14.1.R Packages	156
VII. Contact us	157
This code is primarily maintained by:	158
15. Production run notes	159
16. R Version Metadata	160
17. Acknowledgments	162
18. Community Acknowledgments	163
19. Land Acknowledgements	164
20. Technical Acknowledgments	165
20.1.Partners	165
20.2.Collaborators	165
21. References	166

List of Figures

1. Sorting and weighing fish on deck on the 2022 Bering Sea groundfish survey aboard the F/V Alaska Knight. Credit: Emily Markowitz/NOAA Fisheries.	3
1.1. Strata used in the Eastern Bering Sea Survey.	10
1.2. Strata used in the Bering Sea Slope Survey.	11
2.1. Simplified data workflow from boat to production.	12
7.1. Ex. 6: EBS Pacific Ocean perch CPUE and akgfmaps map.	83
7.2. Ex. 1: GOA Pacific Ocean perch biomass and abundance.	86
7.3. Ex. 2: AI Rock sole size compositions and ridge plot.	88
7.4. Ex. 3: EBS Walleye Pollock Age Compositions and Age Pyramid.	91
7.5. Ex. 4: NBS Pacific cod biomass and abundance.	94
7.6. Ex. 5: GOA Pacific Ocean perch biomass and line plot.	97
10.1. AFSC Groundfish and Crab Assessment Program Bottom Trawl Survey data interface on the Fisheries One Stop Shop platform.	129
10.2. Diagram of selection and search tools available on the FOSS platofrom.	130
10.3. Diagram of selection and search tools available on the FOSS platofrom.	131
10.4. Diagram of the pre-set data format options.	131
10.5. Example data returned from running the report.	132
11.1. Ex. 7: Visualize CPUE data in distribution map.	140

List of Tables

1.1. Survey summary stats	7
7.1. Ex. 7: Summary table of Number hauls where species were observed (N_HAULS) and number of length measurements that were collected for this species (N_LENGTHS) for EBS and NBS, for arrowtooth flounder (10110), Bering flounder (10140), and Alaska skate (435).	79
7.2. Ex. 8: CPUE for all stations contained in the Shumagin region (AREA_ID = 919).	80
7.3. Ex. 6: EBS Pacific Ocean perch CPUE and akgfmaps map.	82
7.4. Ex. 1: GOA Pacific Ocean perch biomass and abundance.	84
7.5. Ex. 2: AI Rock sole size compositions and ridge plot.	87
7.6. Ex. 3: EBS Walleye Pollock Age Compositions and Age Pyramid.	90
7.7. Ex. 4: NBS Pacific cod biomass and abundance.	92
7.8. Ex. 5: GOA Pacific Ocean perch biomass and line plot.	95
11.1.Ex. 1: Load the first 25 rows (default) of data.	134
11.2.Ex. 3: Filter by Year.	135
11.3.Ex. 4: Filter by species name.	136
11.4.Ex. 5: Combination of year and name filters.	137
11.5.Ex. 6: Combination of year, srvy, stratum.	137
11.6.Ex. 7: Visualize CPUE data in distribution map.	138
13.1.Survey of products developed by GAP	153

Part I.

Welcome

Our Objective

Please consider this resource to be a **Living Document**. The code in this repository is regularly being updated and improved. Please refer to releases for finalized products and project milestones.

At this time, these master production and AKFIN tables are **provisional and we are welcoming feedback before the 2024 survey season**. We look forward to hearing from you. Do not hesitate to reach out (to us at either afsc.gap.metadata@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.

Our Objective

As part of our commitment to open science and transparency, we provide this interactive metadata guide to compliment our public-domain data. Please refer to our Draft Data Changes Brief. Once finalized, this language will be included here.

User Resources

- Groundfish Assessment Program Bottom Trawl Surveys
- AFSC's Resource Assessment and Conservation Engineering Division
- Survey code books
- Publications and Data Reports
- Research Surveys conducted at AFSC

Background of the gap_products repo

This work is the result of the massive efforts of three concurrent GAP working groups:

- (1) **Index Computation Working Group:** consolidation of the methods used to produced design-based estimates of abundance and size/age composition between the Bering Sea and AI/GOA survey regions.

Background of the gap_products repo



Figure 1.: Sorting and weighing fish on deck on the 2022 Bering Sea groundfish survey aboard the F/V Alaska Knight. Credit: Emily Markowitz/NOAA Fisheries.

Major Advantages

- (2) **Data Processes Working Group:** reorganization of the Oracle data infrastructure that houses the standard data products produced by GAP.
- (3) **Gulf of Alaska Survey Restratiification Working Group:** implementation of a new stratified random survey design in the Gulf of Alaska bottom trawl survey.

We began this effort in collaboration with the Status of Stocks team (SSMA) to present both the orientation and opportunity to interact with Gulf of Alaska data from the restratiified survey design that we will be implementing in the 2025 field season. As that part of the project evolved, the Data Processes Working Group identified the opportunity and need for gaining efficiencies by redesigning and consolidating the Oracle objects (tables and materialized views) that have historically served these data. The Index Computation Working Group also identified an opportunity to gain efficiencies by consolidating the various scripts that were developed independently by both survey region groups into a workflow that was more accessible and documented.

The Index Computation Working Group developed the gapindex R package, a code repository that consolidates the code that calculates the various standard GAP products (e.g., CPUE, total biomass, size/age composition) for both the Bering Sea and AIGOA survey regions. The Data Processes Working Group was responsible for compiling the data structures needed to support data product tables that were consistent across all of the AFSC GAP survey regions as well as the creation of the GAP_PRODUCTS oracle schema that will house these consolidated products in the future.

This gap_products GitHub repository houses the code that will conduct the “standard production run” that produces the new data tables via the gapindex R package and upload those tables to the GAP_PRODUCTS Oracle schema.

Major Advantages

- Consolidated production tables include all standard data products for all surveys. Data will be provided in the same format, with the same units, and created using the same mathematical methodology. This should limit data pulls, reduce complexity for data access, and reduce complicated secondary data wrangling.
- Consistent naming conventions for schemata, tables, and column metadata. Columns across all tables will use the same naming conventions, units, and data types. Restricting standard data product table content to absolutely necessary columns.

Major Advantages

- Removal of redundant data columns that can be acquired by joining to reference tables is key for providing consistent and up-to-date data while limiting data table sizes.
- Consolidation and repurposing of Oracle schemata. This will help the GAP team limit unnecessary access to unprocessed or problematic data by outside users.
- Vetted data methods. All code and data inclusion decisions and wrangling are documented in the {gapindex} R package. Streamlined and rapid data production. Improved and consolidated data creation and documentation provide data creators and users with greater confidence in the data products and enhanced ability to share the data.

1. Survey background

1.1. What is the research objective?

The objectives of these surveys are to:

- monitor trends in the marine ecosystem of the Bering Sea, Aleutian Islands, and Gulf of Alaska
- produce fishery-independent biomass and abundance estimates for commercially important fish and crab species
- collect biological and environmental data for use in ecosystem-based fishery management.

Learn more about the program

1.2. Who is conducting the research?

Scientists from the Alaska Fisheries Science Center conduct these bottom trawl surveys with participation from the Alaska Department of Fish & Game (ADF&G), the International Pacific Halibut Commission (IPHC), and universities. This research is conducted on chartered fishing vessels.

1. Survey background

1.3. Bottom trawl surveys and regions

Bottom Trawl Survey Regions

AFSC RACE Groundfish and Shellfish Public Data Coverage



Each survey conducted by the Groundfish Assessment Program are multispecies bottom trawl surveys. We collect environmental and biological data to assess how climate variability and loss of sea ice are affecting bottom-dwelling marine life on the Bering Sea shelf. We monitor trends in the distribution (location and movement patterns) and abundance of groundfish and crab species as well as oceanographic data (e.g., water temperature, depth). We collect biological information such as organism weight, length, stomachs to learn about diets, and otoliths to determine fish ages. We use this information in annual stock assessments and to assess the state of the ecosystem. This research is conducted on fishing industry contract vessels.

Table 1.1.: Survey summary stats

Survey	Survey Definition ID	Years	Depth (m)	Area (km ²)	# Statistical Areas	# Possible Stations
Aleutian Islands Bottom Trawl Survey	52	2022 - 1980 (16)	1 - 500	64,415.0	80	1,312
Eastern Bering Sea Slope Bottom Trawl Survey	78	2016 - 2002 (6)	201 - 800	21,134.2	4	

1. Survey background

Survey	Survey Definition ID	Years	Depth (m)	Area (km2)	# Statistical Areas	# Possible Stations
Eastern Bering Sea Crab/Groundfish Bottom Trawl Survey	98	2023 - 1982 (41)	1 - 200	492,989	29	515
Gulf of Alaska Bottom Trawl Survey	47	2023 - 1984 (18)	1 - 1,000	314,087.4	39	6,939
Northern Bering Sea Crab/Groundfish Survey - Eastern Bering Sea Shelf Survey Extension	143	2023 - 2010 (6)	1 - 100	198,866	4	144

1.3.1. Aleutian Islands

(Von Szalay and Raring, 2020)

- Upper Continental Slope of the Aleutian Islands from Unimak Pass to Stalemate Bank
- Triennial (1990s)/Biennial since 2000 in even years, since 1992
- Modified Index-Stratified Random of Successful Stations Survey Design
- Important commercial fish species include Atka mackerel, Pacific ocean perch, walleye pollock, Pacific cod, sablefish, and other rockfish species.

1.3.2. Gulf of Alaska

(Von Szalay and Raring, 2018)

- Continental Shelf and Upper Slope of the Gulf of Alaska extending from the Islands of Four Mountains 2,300 km east to Dixon Entrance
- Triennial (1990s)/Biennial since 2001 in odd years, since 1991
- Stratified Random Survey Design
- Important commercial species in the Gulf of Alaska include Pacific ocean perch, walleye pollock, Pacific cod, flatfish, and other rockfish species.

1. Survey background

1.3.3. Eastern Bering Sea Shelf

(Markowitz et al., 2023)

- The continental shelf of the eastern Bering Sea from the Aleutian Islands to the Bering Strait
- Conducted annually since 1982.
- Uses a stratified systematic sampling survey design with fixed stations at center of 20 x 20 nm grid.
- Similar in design to the northern Bering Sea shelf bottom trawl survey.
- Focus species for the Bering Sea include walleye pollock, Pacific cod, Greenland turbot, yellowfin sole, northern rock sole, red king crab, and snow and Tanner crabs.

1.3.4. Northern Bering Sea

(Markowitz et al., 2023)

- The continental shelf of the northern Bering Sea, including the area north of St. Lawrence Island and Norton Sound
- Biennial/Annual; conducted intermittently since 2010
- Uses a stratified systematic sampling survey design with fixed stations at center of 20 x 20 nm grid.
- Similar in design to the eastern Bering Sea shelf bottom trawl survey.

1.3.5. Eastern Bering Sea Upper Continental Slope

(Hoff, 2016)

- The eastern Bering Sea upper continental slope survey area extends from Unalaska and Akutan Islands to the U.S.-Russian Maritime Boundary at 61° N near the International Date Line (166° E to 180° W) at depths from 200 to 1,200 m
- Conducted intermittently since 2002 (funding dependent)
- Modified Index-Stratified Random of Successful Stations Survey Design
- Focus species for the Bering Sea slope include giant grenadier, Pacific ocean perch, popeye grenadier, walleye pollock, and arrowtooth flounder.

1. Survey background



Figure 1.1.: Strata used in the Eastern Bering Sea Survey.

1. Survey background



Figure 1.2.: Strata used in the Bering Sea Slope Survey.

2. Workflow

2.1. Data workflow from boat to production

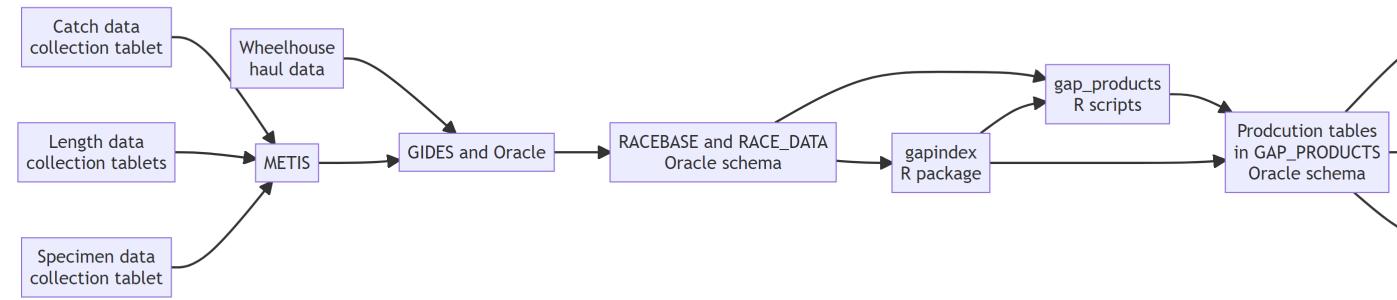


Figure 2.1.: Simplified data workflow from boat to production.

2.2. Organization

The code/run.R script houses the sequence of programs that calculate the standard data products resulting from the NOAA AFSC GAP bottom trawl surveys. Standard data products are the CPUE, BIOMASS, SIZECOMP, and AGECOMP tables in the GAP_PRODUCTS Oracle schema. The tables are slated to be updated twice a year, once after the survey season following finalization of that summer's bottom trawl survey

2. Workflow

data to incorporate the new catch, size, and effort data and once prior to an upcoming survey to incorporate new age data that were processed after the prior summer's survey season ended. This second pre-survey production run will also incorporate changes in the data due to the specimen voucher process as well as other post-hoc changes in the survey data.

Below is a summary of the workflow:

1. Import versions of the tables in GAP_PRODUCTS locally within the gap_products repository to compare with the updated production tables. Any changes to a production table will be compared and checked to make sure those changes are intentional and documented.
2. Use the gapindex R package to calculate the four major standard data products: CPUE, BIOMASS, SIZECOMP, AGECOMP. These tables are compared and checked to their respective locally saved copies and any changes to the tables are vetted and documented. These tables are then uploaded to the GAP_PRODUCTS Oracle schema.
3. Calculate the various materialized views for AKFIN and FOSS purposes. Since these are derivative of the tables in GAP_PRODUCTS as well as other base tables in RACEBASE and RACE_DATA, it is not necessary to check these views in addition to the data checks done in the previous steps.

2.3. Data levels

GAP produces numerous data products that are subjected to different levels of processing, ranging from raw to highly-derived. The suitability of these data products for analysis varies and there is ambiguity about which data products can be used for which purpose. This ambiguity can create challenges in communicating about data products and potentially lead to misunderstanding and misuse of data. One approach to communicating about the level of processing applied to data products and their suitability for analysis is to describe data products using a Data Processing Level system. Data Processing Level systems are widely used in earth system sciences to characterize the extent of processing that has been applied to data products. For example, the NOAA National Centers for Environmental Information (NCEI) Satellite Program uses a Data Processing Level system to describe data on a scale of 0-4, where Level 0 is raw data and Level 4 is model output or results from analysis. Example of how NASA remote sensing data products are shared through a public data portal with levels of data processing and documentation.

2. Workflow

For more information, see Sean Rohan's October 2022 SCRUGS presentation on the topic.

- **Level 0:** Raw and unprocessed data. Ex: Data on the G drive, some tables in RACE_DATA
- **Level 1A:** Data products with QA/QC applied that may or may not be expanded to analysis units, but either not georeferenced or does not include full metadata. Ex: Some tables in RACE_DATA and RACEBASE
- **Level 2:** Analysis-ready data products that are derived for a standardized extent and account for zeros and missing/bad data. Ex: CPUE tables, some data products in public-facing archives and repositories
- **Level 3:** Data products that are synthesized across a standardized extent, often inputs in a higher-level analytical product. Ex: Abundance indices, some data products in public-facing archives and repositories
- **Level 4:** Analytically generated data products that are derived from lower-level data, often to inform management. Ex: Biological reference points from stock assessments, Essential Fish Habitat layers, indicators in Ecosystem Status Reports and Ecosystem and Socioeconomic Profiles

3. News

3.1. Future plans

3.1.1. GOA 2025 Restratiification – Mock Data for Testing

The plan will be, once all are satisfied with the new GAP_PRODUCTS schema and tables, to sunset the historic product tables in 2024 and proceed with only GAP_PRODUCTS for the 2024 post-survey stock assessment season.

- December 2023 - March 2024: Meeting between GAP and stock assessment groups in early December 2023 to update progress on the GAP_PRODUCTS testing phase. **Deadline for Comments and Feedback on GAP_PRODUCTS data structures is March 8, 2024.**
- September 2024: GAP will only release data products according to the new standard. Current, historical data product tables will be archived in a new schema called "**GAP_ARCHIVE**".

3.2. Previous updates

- September 2023: Provisional data product tables – CPUE, BIOMASS, SIZECOMP, and AGECOMP – as well as provisional support tables – AREA, STRATUM_GROUPS, METADATA_COLUMN, SPECIES_YEAR, SURVEY_DESIGN – are available in the GAP_PRODUCTS Oracle schema with updated 2023 GOA and EBS survey data.
 - Additionally, the inclusion of mock data for the under the new 2025 GOA stratified random survey (labeled in the GAP_PRODUCTS tables as YEAR 2025) will provide stock authors with the opportunity to interact with data from the new survey design to be implemented in 2025.

3. News

- Provisional AKFIN and FOSS tables are also available in the GAP_PROD-UCTS Oracle schema. These include: AKFIN_AGECOMP, AKFIN_AREA, AKFIN BIOMASS, AKFIN_CATCH, AKFIN_CPUE, AKFIN_CRUISE, AKFIN_HAUL, AKFIN_LENGTH, AKFIN_METADATA_COLUMN, AKFIN_SIZECOMP, AKFIN_SPECIMEN, AKFIN_SURVEY DESIGN, AKFIN_STRATUM_GROUPS, FOSS_CATCH, FOSS_CPUE_PRESONLY, FOSS_HAUL, and FOSS_TAXON_GROUP.
- May 2023: Release of new, draft, standard data product tables, including restratified GOA data. Stock assessment authors will have the opportunity to explore differences between datasets, test workflows, and provide comments and issues during summer 2023.
- February 2023: Decision was made to include the mock restratified GOA data with the development of the new consolidated standard data products.
- December 2022: GAP and SSMA discuss integration of the restratification of the GOA survey design into standard data products.
 - Stock assessors requested a "dry run" test to work with new mock restratified GOA survey data before implementation of the new survey design.
 - This prompted the postponement of the restratified GOA design to 2025.
- October 2022: The data processes and index computation working group convened to address the development of standard survey data products (e.g., biomass/abundance, size composition, age composition, CPUE).
 - Index Computation Working Group: consolidation of index computation methods between the Bering Sea and AI-GOA regions.
 - Data Processes Working Group: consolidation, clean up, and reorganization of survey oracle schemata, tables, and other data for all surveys.

4. Code of Conduct

4.1. What are Codes of Conduct?

Codes of Conduct are voluntary sets of rules that assist creators, developers, and users of code and data with data protection compliance and accountability in specific sectors or relating to particular processing operations.

Codes can help organizations to ensure all participants follow best practices and rules designed specifically for their sector or processing operations, thus enhancing compliance and collaboration. They are developed and managed by an association or other body (the 'Code Owner') which is representative of a sector (or category of data controllers or processors), with the expert and sectoral knowledge of how to enhance data protection in their area.

4.1.1. Code of Conduct from the nmfs-opensci GitHub.

```
# NOAA Fisheries Open Science Code of Conduct
```

This code of conduct was developed and adapted from the Atom code of conduct in October 2022

```
## Our Pledge
```

In the interest of fostering an open and welcoming environment, we as contributors and main

```
## Our Standards
```

Examples of behavior that contributes to creating a positive environment include:

- * Using welcoming and inclusive language
- * Being respectful of differing viewpoints and experiences
- * Gracefully accepting constructive criticism
- * Focusing on what is best for the community

4. Code of Conduct

- * Showing empathy towards other community members

Examples of unacceptable behavior by participants include:

- * The use of sexualized language or imagery and unwelcome sexual attention or advances
- * Trolling, insulting/derogatory comments, and personal or political attacks
- * Public or private harassment
- * Publishing others' private information, such as a physical or electronic address, without consent
- * Other conduct which could reasonably be considered inappropriate in a professional setting

Our Responsibilities

Project maintainers are responsible for clarifying the standards of acceptable behavior and expectations for participation.

Project maintainers have the right and responsibility to remove, edit, or reject comments, in accordance with their terms of service.

Scope

This Code of Conduct applies both within project spaces and in public spaces when an individual is representing the project in a public capacity.

Enforcement

Instances of abusive, harassing, or otherwise unacceptable behavior may be reported by anyone using the project's designated reporting system.

Attribution

This Code of Conduct is adapted from the [Contributor Covenant] [homepage], version 1.4, available at

[homepage]: <https://contributor-covenant.org>

[version]: <https://contributor-covenant.org/version/1/4/>

Part II.

GAP Production Data

Data Description

The Resource Assessment and Conservation Engineering Division (RACE) Groundfish Assessment Program (GAP) of the Alaska Fisheries Science Center (AFSC) conducts fisheries-independent bottom trawl surveys to monitor the condition of the demersal fish and crab stocks of Alaska. These data are developed to describe the temporal distribution and abundance of commercially and ecologically important groundfish species, examine the changes in the species composition of the fauna over time and space, and describe the physical environment of the groundfish habitat. These data are created using the gapindex R package v2.1.0.

Users must read and fully comprehend the metadata prior to use. Data should not be used beyond the limits of the source scale. Acknowledgement of NOAA, as the source from which these data were obtained, in any publications and/or other representations of these data, is suggested. These data are compiled and approved annually after each summer survey season. The data from previous years are unlikely to change substantially once published. Some survey data are excluded, such as non-standard stations, surveys completed in earlier years using different/non-standard gear, and special tows and non-standard data collections.

Cite this data

Use the below bibtext citations, as cited in our group's citation repository for citing the data created and maintained in this repo (NOAA Fisheries Alaska Fisheries Science Center, Goundfish Assessment Program, 2023). Add "note = {Accessed: mm/dd/yyyy}" to append the day this data was accessed.

```
@misc{GAPPproducts,
  author = {{NOAA Fisheries Alaska Fisheries Science Center, Goundfish Assessment Program}},
  year = {2023},
  title = {AFSC Goundfish Assessment Program Design-Based Production Data},
  howpublished = {https://www.fisheries.noaa.gov/alaska/science-data/groundfish-assessment},
  publisher = {{U.S. Dep. Commer.}},
  copyright = {Public Domain}
}
```

5. Data description

5.1. Data tables

5.1.1. AGECOMP

Region-level age compositions by sex/length bin.

Number of rows: 545,107

Number of columns: 9

Column name from data

Descriptive column Name

Units

Oracle data type

Column description

AGE

NA

NA

NA

NA

AREA_ID

Area ID code

ID code

NUMBER(38,0)

5. Data description

Area ID code for each statistical area used to produce production estimates (e.g., biomass, population, age comps, length comps). Each area ID is unique within each survey.

LENGTH_MM_MEAN

Mean length at age weighted by numbers at length

numeric

NUMBER(38,3)

Mean length (millimeters)

LENGTH_MM_SD

Standard deviation of length at age weighted by numbers at length

numeric

NUMBER(38,3)

Variance of mean length.

POPULATION_COUNT

Estimated population

numeric

NUMBER(38,6)

The estimated population caught in the survey for a species, group, or total for a given survey.

SEX

Sex of a specimen

ID code

NUMBER(38,0)

Sex of a specimen where "1" = "Male", "2" = "Female", "3" = Unsexed.

SPECIES_CODE

Taxon code

ID code

NUMBER(38,0)

5. Data description

The species code of the organism associated with the 'common_name' and 'scientific_name' columns. For a complete species list, review the code books.

SURVEY_DEFINITION_ID

Survey ID

ID code

NUMBER(38,0)

This number uniquely identifies a survey. Name and description of survey. The column 'survey_id' is associated with the 'srvy' and 'survey' columns. For a complete list of surveys, review the code books.

YEAR

Survey year

year

NUMBER(10,0)

Year the observation (survey) was collected.

5.1.2. AREA

This is a table

Number of rows: 473

Number of columns: 10

Column name from data

Descriptive column Name

Units

Oracle data type

Column description

AREA_ID

Area ID code

ID code

NUMBER(38,0)

5. Data description

Area ID code for each statistical area used to produce production estimates (e.g., biomass, population, age comps, length comps). Each area ID is unique within each survey.

AREA_KM2

Area (km2)

kilometers squared

NUMBER(38,3)

Area in square kilometers.

AREA_NAME

Area ID name

text

VARCHAR2(4000 BYTE)

Descriptive name of each AREA_ID. These names often identify the region, depth ranges, or other regional information for the area ID.

CRS

Coordinate reference system

ID code

VARCHAR2(255 BYTE)

The coordinate reference system (CRS) that shapefiles were created in or areas (like AREA_KM2) are calculated in, as defined by <https://spatialreference.org/> (e.g., "+proj=longlat", "EPSG:3338").

DEPTH_MAX_M

Area ID maximum depth (m)

meters

NUMBER(38,3)

Maximum depth (meters).

DEPTH_MIN_M

Area ID minimum depth (m)

meters

5. Data description

NUMBER(38,3)

Minimum depth (meters).

DESCRIPTION

Description

text

VARCHAR2(4000 BYTE)

Description of row observation.

DESIGN_YEAR

Design year

year

NUMBER(10,0)

Year ID associated with a given value AREA_ID. This field describes the changes in the survey design over time.

SURVEY_DEFINITION_ID

Survey ID

ID code

NUMBER(38,0)

This number uniquely identifies a survey. Name and description of survey. The column 'survey_id' is associated with the 'srvy' and 'survey' columns. For a complete list of surveys, review the code books.

TYPE

NA

NA

NA

NA

5. Data description

5.1.3. BIOMASS

Stratum/subarea/region-level mean CPUE (weight and numbers), total biomass, and total abundance with associated variances.

Number of rows: 4,689,557

Number of columns: 16

Column name from data

Descriptive column Name

Units

Oracle data type

Column description

AREA_ID

Area ID code

ID code

NUMBER(38,0)

Area ID code for each statistical area used to produce production estimates (e.g., biomass, population, age comps, length comps). Each area ID is unique within each survey.

BIOMASS_MT

Estimated biomass

numeric

NUMBER(38,6)

The estimated total biomass.

BIOMASS_VAR

Estimated biomass variance

numeric

NUMBER(38,6)

The estimated variance associated with the total biomass.

CPUE_KGKM2_MEAN

5. Data description

Mean weight CPUE

kilograms per kilometers squared

NUMBER(38,6)

The mean catch weight (kilograms) per unit effort (area swept by the net, units squared kilometers).

CPUE_KGKM2_VAR

Variance of the mean weight CPUE

kilograms per kilometers squared

NUMBER(38,6)

The variance of mean catch weight (kilograms) per unit effort (area swept by the net, units squared kilometers).

CPUE_NOKM2_MEAN

Mean numeric CPUE

count per kilometers squared

NUMBER(38,6)

The mean of numerical catch per unit effort (area swept by the net, units square kilometers).

CPUE_NOKM2_VAR

Variance of the mean numeric CPUE

count per kilometers squared

NUMBER(38,6)

The variance of mean numerical catch per unit effort (area swept by the net, units square kilometers).

N_COUNT

Hauls with taxon counts

numeric

NUMBER(38,0)

Total number of hauls with positive count data.

N_HAUL

5. Data description

Valid hauls

numeric

NUMBER(38,0)

Total number of hauls.

N_LENGTH

Hauls with taxon lengths

numeric

NUMBER(38,0)

Total number of hauls with length data.

N_WEIGHT

Hauls with catch

numeric

NUMBER(38,0)

Total number of hauls with positive catch biomass.

POPULATION_COUNT

Estimated population

numeric

NUMBER(38,6)

The estimated population caught in the survey for a species, group, or total for a given survey.

POPULATION_VAR

Estimated population variance

numeric

NUMBER(38,6)

The estimated population variance caught in the survey for a species, group, or total for a given survey.

SPECIES_CODE

Taxon code

5. Data description

ID code

NUMBER(38,0)

The species code of the organism associated with the 'common_name' and 'scientific_name' columns. For a complete species list, review the code books.

SURVEY_DEFINITION_ID

Survey ID

ID code

NUMBER(38,0)

This number uniquely identifies a survey. Name and description of survey. The column 'survey_id' is associated with the 'srvy' and 'survey' columns. For a complete list of surveys, review the code books.

YEAR

Survey year

year

NUMBER(10,0)

Year the observation (survey) was collected.

5.1.4. CPUE

Haul-level zero-filled weight and numerical catch-per-unit-effort.

Number of rows: 38,965,699

Number of columns: 7

Column name from data

Descriptive column Name

Units

Oracle data type

Column description

AREA_SWEPT_KM2

Area swept (km)

5. Data description

kilometers

NUMBER(38,6)

The area the net covered while the net was fishing (kilometers squared), defined as the distance fished times the net width.

COUNT

Taxon count

count, whole number resolution

NUMBER(38,0)

Total whole number of individuals caught in haul.

CPUE_KGKM2

Weight CPUE (kg/km²)

kilograms per kilometers squared

NUMBER(38,6)

Catch weight (kilograms) per unit effort (area swept by the net, units square kilometers).

CPUE_NOKM2

Number CPUE (no/km²)

count per kilometers squared

NUMBER(38,6)

Numerical catch per unit effort (area swept by the net, units square kilometers).

HAULJOIN

Haul ID

ID code

NUMBER(38,0)

This is a unique numeric identifier assigned to each (vessel, cruise, and haul) combination.

SPECIES_CODE

Taxon code

5. Data description

ID code

NUMBER(38,0)

The species code of the organism associated with the 'common_name' and 'scientific_name' columns. For a complete species list, review the code books.

WEIGHT_KG

Sample or taxon weight (kg)

kilograms

NUMBER(38,3)

Weight (thousandths of a kilogram) of individuals in a haul by taxon.

5.1.5. SURVEY DESIGN

This is a table

Number of rows: 126

Number of columns: 4

Column name from data

Descriptive column Name

Units

Oracle data type

Column description

DESIGN_YEAR

Design year

year

NUMBER(10,0)

Year ID associated with a given value AREA_ID. This field describes the changes in the survey design over time.

SURVEY

Survey Name

text

5. Data description

VARCHAR2(255 BYTE)

Name and description of survey. The column 'survey' is associated with the 'srvy' and 'survey_id' columns.

SURVEY_DEFINITION_ID

Survey ID

ID code

NUMBER(38,0)

This number uniquely identifies a survey. Name and description of survey. The column 'survey_id' is associated with the 'srvy' and 'survey' columns. For a complete list of surveys, review the code books.

YEAR

Survey year

year

NUMBER(10,0)

Year the observation (survey) was collected.

5.1.6. METADATA_TABLE

These columns provide the table metadata for all of the tables and views in GAP_PRODUCTS. These tables are created by the Resource Assessment and Conservation Engineering Division (RACE) Groundfish Assessment Program (GAP) of the Alaska Fisheries Science Center (AFSC). The GitHub repository for the scripts that created this code can be found at https://github.com/afsc-gap-products/gap_products. These data were last updated October 12, 2023. There are no legal restrictions on access to the data. For more information about codes used in the tables, please refer to the survey code books (<https://www.fisheries.noaa.gov/resource/document/groundfish-survey-species-code-manual-and-data-codes-manual>).

Number of rows: 8

Number of columns: 3

Column name from data

Descriptive column Name

Units

5. Data description

Oracle data type
Column description
METADATA_SENTENCE
Sentence
text
VARCHAR2(255 BYTE)
Table metadata sentence.
METADATA_SENTENCE_NAME
Metadata sentence name
text
VARCHAR2(255 BYTE)
Name of table metadata sentence.
METADATA_SENTENCE_TYPE
Sentence type
text
VARCHAR2(255 BYTE)
Type of sentence to have in table metadata.

5.1.7. STRATUM_GROUPS

This is a table
Number of rows: 774
Number of columns: 4
Column name from data
Descriptive column Name
Units
Oracle data type
Column description

5. Data description

AREA_ID

Area ID code

ID code

NUMBER(38,0)

Area ID code for each statistical area used to produce production estimates (e.g., biomass, population, age comps, length comps). Each area ID is unique within each survey.

DESIGN_YEAR

Design year

year

NUMBER(10,0)

Year ID associated with a given value AREA_ID. This field describes the changes in the survey design over time.

STRATUM

Stratum ID

ID code

NUMBER(10,0)

RACE database statistical area for analyzing data. Strata were designed using bathymetry and other geographic and habitat-related elements. The strata are unique to each survey region. Stratum of value 0 indicates experimental tows.

SURVEY_DEFINITION_ID

Survey ID

ID code

NUMBER(38,0)

This number uniquely identifies a survey. Name and description of survey. The column 'survey_id' is associated with the 'srvy' and 'survey' columns. For a complete list of surveys, review the code books.

5. Data description

5.1.8. SIZECOMP

Stratum/subarea/region-level size compositions by sex. This table was created by the Resource Assessment and Conservation Engineering Division (RACE) Groundfish Assessment Program (GAP) of the Alaska Fisheries Science Center (AFSC). There are legal restrictions on access to the data. These data are not intended for public dissemination and should not be shared without the explicit written consent of the data managers and owners (NOAA Fisheries). The GitHub repository for the scripts that created this code can be found at https://github.com/afsc-gap-products/gap_products. For more information about codes used in the tables, please refer to the survey code books (<https://www.fisheries.noaa.gov/resource/document/groundfish-survey-species-code-manual-and-data-codes-manual>). These data were last updated September 25, 2023.

Number of rows: 3,186,101

Number of columns: 7

Column name from data

Descriptive column Name

Units

Oracle data type

Column description

AREA_ID

Area ID code

ID code

NUMBER(38,0)

Area ID code for each statistical area used to produce production estimates (e.g., biomass, population, age comps, length comps). Each area ID is unique within each survey.

LENGTH_MM

Length of a specimen

millimeters

NUMBER(10,0)

Length bin in millimeters.

5. Data description

POPULATION_COUNT

Estimated population

numeric

NUMBER(38,6)

The estimated population caught in the survey for a species, group, or total for a given survey.

SEX

Sex of a specimen

ID code

NUMBER(38,0)

Sex of a specimen where "1" = "Male", "2" = "Female", "3" = Unsexed.

SPECIES_CODE

Taxon code

ID code

NUMBER(38,0)

The species code of the organism associated with the 'common_name' and 'scientific_name' columns. For a complete species list, review the code books.

SURVEY_DEFINITION_ID

Survey ID

ID code

NUMBER(38,0)

This number uniquely identifies a survey. Name and description of survey. The column 'survey_id' is associated with the 'srvy' and 'survey' columns. For a complete list of surveys, review the code books.

YEAR

Survey year

year

NUMBER(10,0)

Year the observation (survey) was collected.

Part III.

AKFIN

The Alaska Fisheries Information Network

These data are used directly by stock assessors and are provided to The [Alaska Fisheries Information Network (AKFIN)].

The Alaska Fisheries Information Network

The Alaska Fisheries Information Network (AKFIN) is a regional program that consolidates and supports the collection, processing, analysis, and reporting of fisheries statistics for North Pacific and Alaskan fisheries. AKFIN integrates this information into a single data management system using consistent methods and standardized formats. The Network then reports this information on its website, in various publications, and to researchers. The resulting data enables fishery managers, scientists, and associated agencies to supervise fisheries resources more effectively and efficiently.

If you are an AFSC employee with access to data through our internal database Oracle server, use this guide to access our data. If not, reach out to AKFIN for a user account.

Cite this data

Use the below bibtext citations, as cited in our group's citation repository for citing the data created and maintained in this repo (Alaska Fisheries Information Network (AKFIN), 2023). Add "note = {Accessed: mm/dd/yyyy}" to append the day this data was accessed.

```
@misc{GAPakfin,
  author = {{Alaska Fisheries Information Network (AKFIN)}},
  institution = {{NOAA Fisheries Alaska Fisheries Science Center, Goundfish Assessment Program}},
  year = {2023},
  title = {AFSC Goundfish Assessment Program Design-Based Production Data},
  howpublished = {\url{https://www.psmfc.org/program/alaska-fisheries-information-network-akfin}},
  publisher = {{U.S. Dep. Commer.}},
  copyright = {Public Domain}
}
```

6. Data description

6.1. Other resources

(AKFIN Answers)[<https://akfin.psmfc.org/akfin-answers/>] is an Oracle BI tool used for distributing data to stock assessors and other users. Usernames and passwords are distinct from direct AKFIN database credentials.

6.2. Data description

6.3. Data tables

6.3.1. AKFIN_AGECOMP

This table is a copy of GAP_PRODUCTS.AGECOMP and does not have any other object dependencies. These data are produced by the Resource Assessment and Conservation Engineering Division (RACE) Groundfish Assessment Program (GAP) of the Alaska Fisheries Science Center (AFSC). There are legal restrictions on access to the data. These data are not intended for public dissemination and should not be shared without the explicit written consent of the data managers and owners (NOAA Fisheries). The GitHub repository for the scripts that created this code can be found at https://github.com/afsc-gap-products/gap_products. These data were last updated September 26, 2023.

Number of rows: 533,785

Number of columns: 9

Column name from data

Descriptive column Name

Units

6. Data description

Oracle data type

Column description

AGE

NA

NA

NA

NA

AREA_ID

Area ID code

ID code

NUMBER(38,0)

Area ID code for each statistical area used to produce production estimates (e.g., biomass, population, age comps, length comps). Each area ID is unique within each survey.

LENGTH_MM_MEAN

Mean length at age weighted by numbers at length

numeric

NUMBER(38,3)

Mean length (millimeters)

LENGTH_MM_SD

Standard deviation of length at age weighted by numbers at length

numeric

NUMBER(38,3)

Variance of mean length.

POPULATION_COUNT

Estimated population

numeric

NUMBER(38,6)

6. Data description

The estimated population caught in the survey for a species, group, or total for a given survey.

SEX

Sex of a specimen

ID code

NUMBER(38,0)

Sex of a specimen where "1" = "Male", "2" = "Female", "3" = Unsexed.

SPECIES_CODE

Taxon code

ID code

NUMBER(38,0)

The species code of the organism associated with the 'common_name' and 'scientific_name' columns. For a complete species list, review the code books.

SURVEY_DEFINITION_ID

Survey ID

ID code

NUMBER(38,0)

This number uniquely identifies a survey. Name and description of survey. The column 'survey_id' is associated with the 'srvy' and 'survey' columns. For a complete list of surveys, review the code books.

YEAR

Survey year

year

NUMBER(10,0)

Year the observation (survey) was collected.

6. Data description

6.3.2. AKFIN_AREA

This table is a copy of GAP_PRODUCTS.AREA and does not have any other object dependencies. These data are produced by the Resource Assessment and Conservation Engineering Division (RACE) Groundfish Assessment Program (GAP) of the Alaska Fisheries Science Center (AFSC). There are legal restrictions on access to the data. These data are not intended for public dissemination and should not be shared without the explicit written consent of the data managers and owners (NOAA Fisheries). The GitHub repository for the scripts that created this code can be found at https://github.com/afsc-gap-products/gap_products. These data were last updated September 26, 2023.

Number of rows: 473

Number of columns: 10

Column name from data

Descriptive column Name

Units

Oracle data type

Column description

AREA_ID

Area ID code

ID code

NUMBER(38,0)

Area ID code for each statistical area used to produce production estimates (e.g., biomass, population, age comps, length comps). Each area ID is unique within each survey.

AREA_KM2

Area (km2)

kilometers squared

NUMBER(38,3)

Area in square kilometers.

AREA_NAME

6. Data description

Area ID name

text

VARCHAR2(4000 BYTE)

Descriptive name of each AREA_ID. These names often identify the region, depth ranges, or other regional information for the area ID.

CRS

Coordinate reference system

ID code

VARCHAR2(255 BYTE)

The coordinate reference system (CRS) that shapefiles were created in or areas (like AREA_KM2) are calculated in, as defined by <https://spatialreference.org/> (e.g., "+proj=longlat", "EPSG:3338").

DEPTH_MAX_M

Area ID maximum depth (m)

meters

NUMBER(38,3)

Maximum depth (meters).

DEPTH_MIN_M

Area ID minimum depth (m)

meters

NUMBER(38,3)

Minimum depth (meters).

DESCRIPTION

Description

text

VARCHAR2(4000 BYTE)

Description of row observation.

DESIGN_YEAR

6. Data description

Design year

year

NUMBER(10,0)

Year ID associated with a given value AREA_ID. This field describes the changes in the survey design over time.

SURVEY_DEFINITION_ID

Survey ID

ID code

NUMBER(38,0)

This number uniquely identifies a survey. Name and description of survey. The column 'survey_id' is associated with the 'srvy' and 'survey' columns. For a complete list of surveys, review the code books.

TYPE

NA

NA

NA

NA

6.3.3. AKFIN_BIOMASS

This table is a copy of GAP_PRODUCTS.BIOMASS and does not have any other object dependencies. These data are produced by the Resource Assessment and Conservation Engineering Division (RACE) Groundfish Assessment Program (GAP) of the Alaska Fisheries Science Center (AFSC). There are legal restrictions on access to the data. These data are not intended for public dissemination and should not be shared without the explicit written consent of the data managers and owners (NOAA Fisheries). The GitHub repository for the scripts that created this code can be found at https://github.com/afsc-gap-products/gap_products. These data were last updated October 04, 2023.

Number of rows: 4,569,717

Number of columns: 16

Column name from data

6. Data description

Descriptive column Name

Units

Oracle data type

Column description

AREA_ID

Area ID code

ID code

NUMBER(38,0)

Area ID code for each statistical area used to produce production estimates (e.g., biomass, population, age comps, length comps). Each area ID is unique within each survey.

BIOMASS_MT

Estimated biomass

numeric

NUMBER(38,6)

The estimated total biomass.

BIOMASS_VAR

Estimated biomass variance

numeric

NUMBER(38,6)

The estimated variance associated with the total biomass.

CPUE_KGKM2_MEAN

Mean weight CPUE

kilograms per kilometers squared

NUMBER(38,6)

The mean catch weight (kilograms) per unit effort (area swept by the net, units squared kilometers).

CPUE_KGKM2_VAR

6. Data description

Variance of the mean weight CPUE

kilograms per kilometers squared

NUMBER(38,6)

The variance of mean catch weight (kilograms) per unit effort (area swept by the net, units squared kilometers).

CPUE_NOKM2_MEAN

Mean numeric CPUE

count per kilometers squared

NUMBER(38,6)

The mean of numerical catch per unit effort (area swept by the net, units square kilometers).

CPUE_NOKM2_VAR

Variance of the mean numeric CPUE

count per kilometers squared

NUMBER(38,6)

The variance of mean numerical catch per unit effort (area swept by the net, units square kilometers).

N_COUNT

Hauls with taxon counts

numeric

NUMBER(38,0)

Total number of hauls with positive count data.

N_HAUL

Valid hauls

numeric

NUMBER(38,0)

Total number of hauls.

N_LENGTH

6. Data description

Hauls with taxon lengths

numeric

NUMBER(38,0)

Total number of hauls with length data.

N_WEIGHT

Hauls with catch

numeric

NUMBER(38,0)

Total number of hauls with positive catch biomass.

POPULATION_COUNT

Estimated population

numeric

NUMBER(38,6)

The estimated population caught in the survey for a species, group, or total for a given survey.

POPULATION_VAR

Estimated population variance

numeric

NUMBER(38,6)

The estimated population variance caught in the survey for a species, group, or total for a given survey.

SPECIES_CODE

Taxon code

ID code

NUMBER(38,0)

The species code of the organism associated with the 'common_name' and 'scientific_name' columns. For a complete species list, review the code books.

SURVEY_DEFINITION_ID

6. Data description

Survey ID

ID code

NUMBER(38,0)

This number uniquely identifies a survey. Name and description of survey. The column 'survey_id' is associated with the 'srvy' and 'survey' columns. For a complete list of surveys, review the code books.

YEAR

Survey year

year

NUMBER(10,0)

Year the observation (survey) was collected.

6.3.4. AKFIN_CATCH

snapshot table for snapshot GAP_PRODUCTS.AKFIN_CATCH

Number of rows: 989,351

Number of columns: 6

Column name from data

Descriptive column Name

Units

Oracle data type

Column description

CATCHJOIN

Catch observation ID

ID code

NUMBER(38,0)

Unique interger ID assigned to each survey, vessel, year, and catch observation combination.

COUNT

6. Data description

Taxon count

count, whole number resolution

NUMBER(38,0)

Total whole number of individuals caught in haul.

CRUISEJOIN

Cruise ID

ID code

NUMBER(38,0)

Unique integer ID assigned to each survey, vessel, and year combination.

HAULJOIN

Haul ID

ID code

NUMBER(38,0)

This is a unique numeric identifier assigned to each (vessel, cruise, and haul) combination.

SPECIES_CODE

Taxon code

ID code

NUMBER(38,0)

The species code of the organism associated with the 'common_name' and 'scientific_name' columns. For a complete species list, review the code books.

WEIGHT_KG

Sample or taxon weight (kg)

kilograms

NUMBER(38,3)

Weight (thousandths of a kilogram) of individuals in a haul by taxon.

6. Data description

6.3.5. AKFIN_CPUE

This table is a copy of GAP_PRODUCTS.CPUE and does not have any other object dependencies. These data are produced by the Resource Assessment and Conservation Engineering Division (RACE) Groundfish Assessment Program (GAP) of the Alaska Fisheries Science Center (AFSC). There are legal restrictions on access to the data. These data are not intended for public dissemination and should not be shared without the explicit written consent of the data managers and owners (NOAA Fisheries). The GitHub repository for the scripts that created this code can be found at https://github.com/afsc-gap-products/gap_products. These data were last updated September 26, 2023.

Number of rows: 37,905,115

Number of columns: 7

Column name from data

Descriptive column Name

Units

Oracle data type

Column description

AREA_SWEPT_KM2

Area swept (km)

kilometers

NUMBER(38,6)

The area the net covered while the net was fishing (kilometers squared), defined as the distance fished times the net width.

COUNT

Taxon count

count, whole number resolution

NUMBER(38,0)

Total whole number of individuals caught in haul.

CPUE_KGKM2

Weight CPUE (kg/km2)

6. Data description

kilograms per kilometers squared

NUMBER(38,6)

Catch weight (kilograms) per unit effort (area swept by the net, units square kilometers).

CPUE_NOKM2

Number CPUE (no/km2)

count per kilometers squared

NUMBER(38,6)

Numerical catch per unit effort (area swept by the net, units square kilometers).

HAULJOIN

Haul ID

ID code

NUMBER(38,0)

This is a unique numeric identifier assigned to each (vessel, cruise, and haul) combination.

SPECIES_CODE

Taxon code

ID code

NUMBER(38,0)

The species code of the organism associated with the 'common_name' and 'scientific_name' columns. For a complete species list, review the code books.

WEIGHT_KG

Sample or taxon weight (kg)

kilograms

NUMBER(38,3)

Weight (thousandths of a kilogram) of individuals in a haul by taxon.

6. Data description

6.3.6. AKFIN_CRUISE

This is the cruise data table. These data are produced by the Resource Assessment and Conservation Engineering Division (RACE) Groundfish Assessment Program (GAP) of the Alaska Fisheries Science Center (AFSC). There are legal restrictions on access to the data. These data are not intended for public dissemination and should not be shared without the explicit written consent of the data managers and owners (NOAA Fisheries). The GitHub repository for the scripts that created this code can be found at https://github.com/afsc-gap-products/gap_products. These data were last updated September 26, 2023.

Number of rows: 187

Number of columns: 10

Column name from data

Descriptive column Name

Units

Oracle data type

Column description

CRUISE

Cruise ID

ID code

NUMBER(38,0)

This is a six-digit integer identifying the cruise number of the form: YYYY99 (where YYYY = year of the cruise; 99 = 2-digit number and is sequential; 01 denotes the first cruise that vessel made in this year, 02 is the second, etc.).

CRUISEJOIN

Cruise ID

ID code

NUMBER(38,0)

Unique interger ID assigned to each survey, vessel, and year combination.

DATE_END

End date

6. Data description

YYYY-MM-DD

DATE

The date (YYYY-MM-DD) of the end of the event (e.g., cruise).

DATE_START

Start date

YYYY-MM-DD

DATE

The date (YYYY-MM-DD) of the beginning of the event (e.g., cruise).

SPONSOR_ACRONYM

NA

NA

NA

NA

SURVEY_DEFINITION_ID

Survey ID

ID code

NUMBER(38,0)

This number uniquely identifies a survey. Name and description of survey. The column 'survey_id' is associated with the 'srvy' and 'survey' columns. For a complete list of surveys, review the code books.

SURVEY_NAME

NA

NA

NA

NA

VESSEL_ID

Vessel ID

ID code

6. Data description

NUMBER(38,0)

ID number of the vessel used to collect data for that haul. The column ‘vessel_id’ is associated with the ‘vessel_name’ column. Note that it is possible for a vessel to have a new name but the same vessel id number. For a complete list of vessel ID codes, review the code books.

VESSEL_NAME

Vessel name

text

VARCHAR2(255 BYTE)

Name of the vessel used to collect data for that haul. The column ‘vessel_name’ is associated with the ‘vessel_id’ column. Note that it is possible for a vessel to have a new name but the same vessel id number. For a complete list of vessel ID codes, review the code books.

YEAR

Survey year

year

NUMBER(10,0)

Year the observation (survey) was collected.

6.3.7. AKFIN_HAUL

This table is created by subsetting the RACEBASE.HAUL table to only hauls with ABUNDANCE_HAUL = ‘Y’. These are the hauls that are used for the standard production tables in GAP_PRODUCTS. These data are produced by the Resource Assessment and Conservation Engineering Division (RACE) Groundfish Assessment Program (GAP) of the Alaska Fisheries Science Center (AFSC). There are legal restrictions on access to the data. These data are not intended for public dissemination and should not be shared without the explicit written consent of the data managers and owners (NOAA Fisheries). The GitHub repository for the scripts that created this code can be found at https://github.com/afsc-gap-products/gap_products. These data were last updated September 26, 2023

Number of rows: 36,114

Number of columns: 25

6. Data description

Column name from data

Descriptive column Name

Units

Oracle data type

Column description

ACCESSORIES

Type of gear accessories used on the net

ID code

NUMBER(38,0)

Type of accessories used on net. For a complete list of accessories ID codes, review the code books.

BOTTOM_TYPE

Seafloor bottom type code

ID code

NUMBER(38,0)

Bottom type on sea floor at haul location. For a complete list of bottom type ID codes, review the code books.

CRUISEJOIN

Cruise ID

ID code

NUMBER(38,0)

Unique interger ID assigned to each survey, vessel, and year combination.

DATE_TIME_START

Start date and time

MM/DD/YYYY HH::MM

DATE

The date (MM/DD/YYYY) and time (HH:MM) of the beginning of the haul.

DEPTH_GEAR_M

6. Data description

Depth of gear (m)

degrees Celsius

NUMBER(38,1)

Depth of gear (meters).

DEPTH_M

Depth (m)

degrees Celsius

NUMBER(38,1)

Bottom depth (meters).

DISTANCE_FISHED_KM

Distance fished (km)

degrees Celsius

NUMBER(38,3)

Distance the net fished (thousandths of kilometers).

DURATION_HR

Tow duration (decimal hr)

hours

NUMBER(38,1)

This is the elapsed time between start and end of a haul (decimal hours).

GEAR

Type of gear used on the net

ID code

NUMBER(38,0)

Type of gear used on net. For a complete list of gear ID codes, review the code books.

GEAR_TEMPERATURE_C

Gear temperature (Degrees Celsius)

degrees Celsius

6. Data description

NUMBER(38,1)

Temperature recorded by net gear (tenths of a degree Celsius); NA indicates removed or missing values.

HAUL

Haul number

ID code

NUMBER(38,0)

This number uniquely identifies a sampling event (haul) within a cruise. It is a sequential number, in chronological order of occurrence.

HAULJOIN

Haul ID

ID code

NUMBER(38,0)

This is a unique numeric identifier assigned to each (vessel, cruise, and haul) combination.

HAUL_TYPE

Haul sampling type

ID code

NUMBER(38,0)

Type of haul sampling method. For a complete list of haul type ID codes, review the code books.

LATITUDE_DD_END

End latitude (decimal degrees)

decimal degrees

NUMBER(38,6)

Latitude (one hundred thousandth of a decimal degree) of the end of the haul.

LATITUDE_DD_START

Start latitude (decimal degrees)

decimal degrees

6. Data description

NUMBER(38,6)

Latitude (one hundred thousandth of a decimal degree) of the start of the haul.

LONGITUDE_DD_END

End longitude (decimal degrees)

decimal degrees

NUMBER(38,6)

Longitude (one hundred thousandth of a decimal degree) of the end of the haul.

LONGITUDE_DD_START

Start longitude (decimal degrees)

decimal degrees

NUMBER(38,6)

Longitude (one hundred thousandth of a decimal degree) of the start of the haul.

NET_HEIGHT_M

Net height (m)

meters

NUMBER(38,1)

Measured or estimated distance (meters) between footrope and headrope of the trawl.

NET_MEASURED

Net measured during haul

logical

BINARY_DOUBLE

Logical, describing if the net was measured (TRUE) or not (FALSE) by wheelhouse and marport programs during the haul.

NET_WIDTH_M

Net width (m)

meters

NUMBER(38,1)

6. Data description

Measured or estimated distance (meters) between wingtips of the trawl.

PERFORMANCE

Haul performance code

category

NUMBER(38,0)

This denotes what, if any, issues arose during the haul. For more information, review the code books.

STATION

Station ID

ID code

VARCHAR2(255 BYTE)

Alpha-numeric designation for the station established in the design of a survey.

STRATUM

Stratum ID

ID code

NUMBER(10,0)

RACE database statistical area for analyzing data. Strata were designed using bathymetry and other geographic and habitat-related elements. The strata are unique to each survey region. Stratum of value 0 indicates experimental tows.

SURFACE_TEMPERATURE_C

Surface temperature (Degrees Celsius)

degrees Celsius

NUMBER(38,1)

Surface temperature (tenths of a degree Celsius); NA indicates removed or missing values.

WIRE_LENGTH_M

Trawl wire length

meters

NUMBER(38,0)

6. Data description

Length of wire deployed during a given haul in meters.

6.3.8. AKFIN_LENGTH

snapshot table for snapshot GAP_PRODUCTS.AKFIN_LENGTH

Number of rows: 2,587,694

Number of columns: 7

Column name from data

Descriptive column Name

Units

Oracle data type

Column description

FREQUENCY

Count of observation

count

NUMBER(38,0)

Frequency, or count, of an observation.

HAULJOIN

Haul ID

ID code

NUMBER(38,0)

This is a unique numeric identifier assigned to each (vessel, cruise, and haul) combination.

LENGTH_MM

Length of a specimen

millimeters

NUMBER(10,0)

Length bin in millimeters.

6. Data description

LENGTH_TYPE

Length type

ID code

NUMBER(38,0)

How the taxon was measured (e.g., fork length, carapace width). For a complete list of length_type ID codes, review the code books.

SAMPLE_TYPE

Sample type

ID code

NUMBER(38,0)

Sampling information on how the taxon was sampled. For a complete list of length_type ID codes, review the code books.

SEX

Sex of a specimen

ID code

NUMBER(38,0)

Sex of a specimen where "1" = "Male", "2" = "Female", "3" = Unsexed.

SPECIES_CODE

Taxon code

ID code

NUMBER(38,0)

The species code of the organism associated with the 'common_name' and 'scientific_name' columns. For a complete species list, review the code books.

6. Data description

6.3.9. AKFIN_METADATA_COLUMN

This table is a copy of GAP_PRODUCTS.METADATA_COLUMN and does not have any other object dependencies. These data are produced by the Resource Assessment and Conservation Engineering Division (RACE) Groundfish Assessment Program (GAP) of the Alaska Fisheries Science Center (AFSC). There are legal restrictions on access to the data. These data are not intended for public dissemination and should not be shared without the explicit written consent of the data managers and owners (NOAA Fisheries). The GitHub repository for the scripts that created this code can be found at https://github.com/afsc-gap-products/gap_products. These data were last updated September 26, 2023.

Number of rows: 128

Number of columns: 5

Column name from data

Descriptive column Name

Units

Oracle data type

Column description

METADATA_COLNAME

Column name

text

VARCHAR2(255 BYTE)

Name of the column in a table.

METADATA_COLNAME_DESC

column description

text

VARCHAR2(4000 BYTE)

Descriptpion of the column.

METADATA_COLNAME_LONG

Column name spelled out

6. Data description

text
VARCHAR2(255 BYTE)
Long name for the column.
METADATA_DATATYPE
Oracle datatype code
text
VARCHAR2(255 BYTE)
Oracle data type of data column.
METADATA_UNITS
Units
category
VARCHAR2(255 BYTE)
Units of the column.

6.3.10. AKFIN_SIZECOMP

This table is a copy of GAP_PRODUCTS.SIZECOMP and does not have any other object dependencies. These data are produced by the Resource Assessment and Conservation Engineering Division (RACE) Groundfish Assessment Program (GAP) of the Alaska Fisheries Science Center (AFSC). There are legal restrictions on access to the data. These data are not intended for public dissemination and should not be shared without the explicit written consent of the data managers and owners (NOAA Fisheries). The GitHub repository for the scripts that created this code can be found at https://github.com/afsc-gap-products/gap_products. These data were last updated September 26, 2023.

Number of rows: 3,134,121
Number of columns: 7
Column name from data
Descriptive column Name
Units
Oracle data type

6. Data description

Column description

AREA_ID

Area ID code

ID code

NUMBER(38,0)

Area ID code for each statistical area used to produce production estimates (e.g., biomass, population, age comps, length comps). Each area ID is unique within each survey.

LENGTH_MM

Length of a specimen

millimeters

NUMBER(10,0)

Length bin in millimeters.

POPULATION_COUNT

Estimated population

numeric

NUMBER(38,6)

The estimated population caught in the survey for a species, group, or total for a given survey.

SEX

Sex of a specimen

ID code

NUMBER(38,0)

Sex of a specimen where "1" = "Male", "2" = "Female", "3" = Unsexed.

SPECIES_CODE

Taxon code

ID code

NUMBER(38,0)

6. Data description

The species code of the organism associated with the 'common_name' and 'scientific_name' columns. For a complete species list, review the code books.

SURVEY_DEFINITION_ID

Survey ID

ID code

NUMBER(38,0)

This number uniquely identifies a survey. Name and description of survey. The column 'survey_id' is associated with the 'srvy' and 'survey' columns. For a complete list of surveys, review the code books.

YEAR

Survey year

year

NUMBER(10,0)

Year the observation (survey) was collected.

6.3.11. AKFIN_SPECIMEN

snapshot table for snapshot GAP_PRODUCTS.AKFIN_SPECIMEN

Number of rows: 360,602

Number of columns: 13

Column name from data

Descriptive column Name

Units

Oracle data type

Column description

AGE_DETERMINATION_METHOD

NA

NA

NA

6. Data description

NA

AGE_YEARS

Age bin of taxon

year

NUMBER(38,0)

Age bin of a taxon in years estimated by the age comp estimate.

CRUISEJOIN

Cruise ID

ID code

NUMBER(38,0)

Unique interger ID assigned to each survey, vessel, and year combination.

GONAD_G

Weight of gonads (g)

grams

NUMBER(38,1)

Weight of specimen gonads (grams).

HAULJOIN

Haul ID

ID code

NUMBER(38,0)

This is a unique numeric identifier assigned to each (vessel, cruise, and haul) combination.

LENGTH_MM

Length of a specimen

millimeters

NUMBER(10,0)

Length bin in millimeters.

MATURITY

6. Data description

Specimen maturity code

ID code

NUMBER(38,0)

The maturity code or the condition identified by the maturity code.

SEX

Sex of a specimen

ID code

NUMBER(38,0)

Sex of a specimen where "1" = "Male", "2" = "Female", "3" = Unsexed.

SPECIES_CODE

Taxon code

ID code

NUMBER(38,0)

The species code of the organism associated with the 'common_name' and 'scientific_name' columns. For a complete species list, review the code books.

SPECIMEN_ID

Specimen unique ID

ID code

NUMBER(38,0)

Each individual examined must have a number assigned to it that is unique within each haul (0001 to 9999), though specimen numbers may be repeated between hauls

SPECIMEN_SAMPLE_TYPE

Specimen sample type

ID code

NUMBER(38,0)

For a complete list of Specimen sample type ID codes, review the code books.

SPECIMEN_SUBSAMPLE_METHOD

Specimen subsample method

6. Data description

ID code

NUMBER(38,0)

For a complete list of specimen subsample method ID codes, review the code books.

WEIGHT_G

Specimen weight (g)

grams

NUMBER(38,1)

Weight of specimen (grams).

6.3.12. AKFIN_STRATUM_GROUPS

This table is a copy of GAP_PRODUCTS.STRATUM_GROUPS and does not have any other object dependencies. These data are produced by the Resource Assessment and Conservation Engineering Division (RACE) Groundfish Assessment Program (GAP) of the Alaska Fisheries Science Center (AFSC). There are legal restrictions on access to the data. These data are not intended for public dissemination and should not be shared without the explicit written consent of the data managers and owners (NOAA Fisheries). The GitHub repository for the scripts that created this code can be found at https://github.com/afsc-gap-products/gap_products. These data were last updated September 26, 2023.

Number of rows: 774

Number of columns: 4

Column name from data

Descriptive column Name

Units

Oracle data type

Column description

AREA_ID

Area ID code

ID code

NUMBER(38,0)

6. Data description

Area ID code for each statistical area used to produce production estimates (e.g., biomass, population, age comps, length comps). Each area ID is unique within each survey.

DESIGN_YEAR

Design year

year

NUMBER(10,0)

Year ID associated with a given value AREA_ID. This field describes the changes in the survey design over time.

STRATUM

Stratum ID

ID code

NUMBER(10,0)

RACE database statistical area for analyzing data. Strata were designed using bathymetry and other geographic and habitat-related elements. The strata are unique to each survey region. Stratum of value 0 indicates experimental tows.

SURVEY_DEFINITION_ID

Survey ID

ID code

NUMBER(38,0)

This number uniquely identifies a survey. Name and description of survey. The column 'survey_id' is associated with the 'srvy' and 'survey' columns. For a complete list of surveys, review the code books.

6. Data description

6.3.13. AKFIN_SURVEY DESIGN

This table is a copy of GAP_PRODUCTS.SURVEY DESIGN and does not have any other object dependencies. These data are produced by the Resource Assessment and Conservation Engineering Division (RACE) Groundfish Assessment Program (GAP) of the Alaska Fisheries Science Center (AFSC). There are legal restrictions on access to the data. These data are not intended for public dissemination and should not be shared without the explicit written consent of the data managers and owners (NOAA Fisheries). The GitHub repository for the scripts that created this code can be found at https://github.com/afsc-gap-products/gap_products. These data were last updated September 26, 2023.

Number of rows: 126

Number of columns: 4

Column name from data

Descriptive column Name

Units

Oracle data type

Column description

DESIGN_YEAR

Design year

year

NUMBER(10,0)

Year ID associated with a given value AREA_ID. This field describes the changes in the survey design over time.

SURVEY

Survey Name

text

VARCHAR2(255 BYTE)

Name and description of survey. The column 'survey' is associated with the 'srvy' and 'survey_id' columns.

SURVEY_DEFINITION_ID

6. Data description

Survey ID

ID code

NUMBER(38,0)

This number uniquely identifies a survey. Name and description of survey. The column 'survey_id' is associated with the 'srvy' and 'survey' columns. For a complete list of surveys, review the code books.

YEAR

Survey year

year

NUMBER(10,0)

Year the observation (survey) was collected.

6.3.14. AKFIN_TAXONOMIC_CLASSIFICATION

NAThese data are produced by the Resource Assessment and Conservation Engineering Division (RACE) Groundfish Assessment Program (GAP) of the Alaska Fisheries Science Center (AFSC). There are legal restrictions on access to the data. These data are not intended for public dissemination and should not be shared without the explicit written consent of the data managers and owners (NOAA Fisheries). The GitHub repository for the scripts that created this code can be found at https://github.com/afsc-gap-products/gap_products. These data were last updated September 26, 2023.

Number of rows: 2,757

Number of columns: 19

Column name from data

Descriptive column Name

Units

Oracle data type

Column description

CLASS_TAXON

Class phylogenetic rank

6. Data description

category

VARCHAR2(255 BYTE)

Phylogenetic latin rank of class of a given species.

COMMON_NAME

Taxon common name

text

VARCHAR2(255 BYTE)

The common name of the marine organism associated with the 'scientific_name' and 'species_code' columns. For a complete species list, review the code books.

DATABASE

Database source

category

VARCHAR2(255 BYTE)

Taxonomic database source, either ITIS or WoRMS.

DATABASE_ID

Species ID in database

ID code

VARCHAR2(255 BYTE)

Species ID code of a species in the taxonomic "DATABASE" source.

FAMILY_TAXON

Family phylogenetic rank

category

VARCHAR2(255 BYTE)

Phylogenetic latin rank of family of a given species.

GENUS_TAXON

Genus phylogenetic rank

category

VARCHAR2(255 BYTE)

6. Data description

Phylogenetic latin rank of genus of a given species.

ID_RANK

Lowest taxonomic rank

text

VARCHAR2(255 BYTE)

Lowest taxonomic rank of a given species entry.

KINGDOM_TAXON

Kingdom phylogenetic rank

category

VARCHAR2(255 BYTE)

Phylogenetic latin rank of kingdom of a given species.

ORDER_TAXON

Order phylogenetic rank

category

VARCHAR2(255 BYTE)

Phylogenetic latin rank of order of a given species.

PHYLUM_TAXON

Phylum phylogenetic rank

category

VARCHAR2(255 BYTE)

Phylogenetic latin rank of phylum of a given species.

SPECIES_CODE

Taxon code

ID code

NUMBER(38,0)

The species code of the organism associated with the 'common_name' and 'scientific_name' columns. For a complete species list, review the code books.

SPECIES_NAME

6. Data description

Scientific name of species

text

VARCHAR2(255 BYTE)

Scientific name of species.

SUBCLASS_TAXON

Subclass phylogenetic rank

category

VARCHAR2(255 BYTE)

Phylogenetic latin rank of subclass of a given species.

SUBFAMILY_TAXON

Subfamily phylogenetic rank

category

VARCHAR2(255 BYTE)

Phylogenetic latin rank of subfamily of a given species.

SUBORDER_TAXON

Suborder phylogenetic rank

category

VARCHAR2(255 BYTE)

Phylogenetic latin rank of suborder of a given species.

SUBPHYLUM_TAXON

Subphylum phylogenetic rank

category

VARCHAR2(255 BYTE)

Phylogenetic latin rank of subphylum of a given species.

SUPERCLASS_TAXON

Superclass phylogenetic rank

category

6. Data description

VARCHAR2(255 BYTE)

Phylogenetic latin rank of superclass of a given species.

SUPERFAMILY_TAXON

Superfamily phylogenetic rank

category

VARCHAR2(255 BYTE)

Phylogenetic latin rank of superfamily of a given species.

SUPERORDER_TAXON

Superorder phylogenetic rank

category

VARCHAR2(255 BYTE)

Phylogenetic latin rank of superorder of a given species.

SUPERORDER_TAXON

Superorder phylogenetic rank

category

VARCHAR2(255 BYTE)

Phylogenetic latin rank of superorder of a given species.

7. Access data

Access data via Oracle (AFSC only)

AFSC Oracle users can access the database via SQL developer to view and pull the production data directly from the GAP_PRODUCTS Oracle schema. The user can also use SQL developer to view and pull the GAP Products data directly from the GAP_PRODUCTS Oracle schema.

7.0.1. Connect to Oracle from R

Many users will want to access the data from Oracle using R. The user will need to install the RODBC R package and ask OFIS (IT) connect R to Oracle. Then, use the following code in R to establish a connection from R to Oracle:

Here, the user can establish the oracle connection by entering their username and password in the channel `<- gapindex::oracle_connect()` function. Never save usernames or passwords in scripts that may be intentionally or unintentionally shared with others. If no username and password is entered in the function, pop-ups will appear on the screen asking for the username and password.

After you connect to VPN, you'll be able to log into Oracle.

```
library(RODBC)
channel <- gapindex::get_connected()
```

Data SQL Query Examples:

Data SQL Query Examples:

```
library(gapindex)
library(RODBC)
library(flextable)
library(ggplot2)
library(magrittr)
```

7.0.1. Ex. Select all data from a table

You can download all of the tables locally using a variation of the code below. Once connected, pull and save the tables of interest into the R environment.

```
locations <- c(
  "GAP_PRODUCTS.AKFIN_AGECOMP",
  "GAP_PRODUCTS.AKFIN_AREA",
  "GAP_PRODUCTS.AKFIN BIOMASS",
  "GAP_PRODUCTS.AKFIN_CATCH",
  "GAP_PRODUCTS.AKFIN_CPUE",
  "GAP_PRODUCTS.AKFIN_CRUISE",
  "GAP_PRODUCTS.AKFIN_HAUL",
  "GAP_PRODUCTS.AKFIN_LENGTH",
  "GAP_PRODUCTS.AKFIN_METADATA_COLUMN",
  "GAP_PRODUCTS.AKFIN_SIZECOMP",
  "GAP_PRODUCTS.AKFIN_SPECIMEN",
  "GAP_PRODUCTS.AKFIN_STRATUM_GROUPS",
  "GAP_PRODUCTS.AKFIN_SURVEY_DESIGN",
  "GAP_PRODUCTS.AKFIN_TAXONOMIC_CLASSIFICATION"
)

for (i in 1:length(locations)) {
  print(locations[i])
  a <- RODBC::sqlQuery(channel, paste0("SELECT * FROM ", locations[i]))
  write.csv(x = a, file = here::here("data", paste0(locations[i], ".csv")))
}
```

7.0.2. Ex. Summary table for EBS and NBS arrowtooth flounder (10110), Bering flounder (10140), and Bering skate (435)

- Number hauls where species were observed (N_HAULS)

Data SQL Query Examples:

- Number of length measurements that were collected for this species (N_LENGTHS)

```
dat <- RODBC::sqlQuery(channel = channel,
                        query =
""
-- Select columns for output data
SELECT
bb.SPECIES_CODE,
bb.YEAR,
bb.SURVEY_DEFINITION_ID,
bb.AREA_ID,
SUM(bb.N_HAUL) AS N_HAULS, -- Number hauls where species were observed
SUM(bb.N_LENGTH) AS N_LENGTHS -- Number of length measurements that were collected for this

-- Identify what tables to pull data from
FROM GAP_PRODUCTS.AKFIN BIOMASS bb

-- Filter data results
WHERE bb.SPECIES_CODE IN (10110, 435, 10140)
AND bb.YEAR IN (2022, 2023)
AND bb.AREA_ID IN (99900, 99902)
GROUP BY bb.SPECIES_CODE, bb.YEAR, bb.SURVEY_DEFINITION_ID, bb.AREA_ID
ORDER BY bb.SURVEY_DEFINITION_ID, bb.YEAR, bb.SPECIES_CODE;")

dat <- dat %>%
  dplyr::mutate(SPECIES_CODE = as.character(SPECIES_CODE),
                YEAR = as.character(YEAR),
                AREA_ID = as.character(AREA_ID)) %>%
  dplyr::relocate(SURVEY_DEFINITION_ID, YEAR, SPECIES_CODE) %>%
  dplyr::arrange(SPECIES_CODE)

flextable::flextable(dat) %>% theme_zebra()
```

Data SQL Query Examples:

Table 7.1.: Ex. 7: Summary table of Number hauls where species were observed (N_HAULS) and number of length measurements that were collected for this species (N_LENGTHHS) for EBS and NBS, for arrowtooth flounder (10110), Bering flounder (10140), and Alaska skate (435).

SURVEY_DEFINI-YEAR TION_ID	SPECIES_CODE	AREA_ID	N_HAULS	N_LENGTHHS
982022	10110	99900	376	228
982023	10110	99900	376	226
1432022	10110	99902	144	1
1432023	10110	99902	116	0
982022	10140	99900	376	58
982023	10140	99900	376	60
1432022	10140	99902	144	90
1432023	10140	99902	116	77
982022	435	99900	376	78
982023	435	99900	376	102

7.0.3. Ex. CPUE for all stations contained in the INPFC Shumagin region (AREA_ID = 919) for Pacific cod.

```
dat <- RODBC::sqlQuery(channel = channel,
                        query =
"
-- Select columns for output data
SELECT
HAULJOIN,
SPECIES_CODE,
STRATUM,
LATITUDE_DD_START,
LONGITUDE_DD_START,
CPUE_KGKM2,
```

Data SQL Query Examples:

```
GEAR_TEMPERATURE_C
```

```
-- Identify what tables to pull data from
FROM GAP_PRODUCTS.AKFIN_CPUE cpue
LEFT JOIN GAP_PRODUCTS.AKFIN_HAUL haul
USING (HAULJOIN)

-- Filter for P. Cod observations
WHERE SPECIES_CODE IN (21720)

-- Select all stratum within the area_id 919 (INPFC Shumagin region)
AND haul.STRATUM IN
(
SELECT
STRATUM
FROM GAP_PRODUCTS.AKFIN_STRATUM_GROUPS
WHERE AREA_ID = 919
);")
```

```
dat <- dat %>%
  dplyr::select(HAULJOIN, STRATUM, SPECIES_CODE, LATITUDE_DD_START, LONGITUDE_DD_START, CPUE_KGKM2, GEAR_TEMPERATURE_C)
  dplyr::mutate(SPECIES_CODE = as.character(SPECIES_CODE),
                STRATUM = as.character(STRATUM)) %>%
  dplyr::arrange(SPECIES_CODE)

flextable::flextable(head(dat)) %>% theme_zebra()
```

Table 7.2.: Ex. 8: CPUE for all stations contained in the Shumagin region (AREA_ID = 919).

HAULJOIN	STRATUM	SPECIES_CODE	LATITUDE_DD_START	LONGITUDE_DD_START	CPUE_KGKM2	GEAR_TEMPERATURE_C
-22,23912		21720	55.43748	-160.12697	17.63085	4.5
-22,21513		21720	54.58732	-161.1945	76.90763	4.3
-22,16811		21720	54.45823	-163.08722	004.4962	4.9
-22,204110		21720	54.51804	-160.7493	0.00000	4.7

Data SQL Query Examples:

HAULJOINSTRATUM	SPECIES_CODE	LATITUDE_DD_START	LONGITUDE_DD_START	CPUE_KGKM2	GEAR_TEMPERATURE_C
-22,188210	21720	54.38733	-159.6986	0.00000	5.5
-22,183210	21720	54.09314	-161.6996	0.00000	5.3

7.0.4. Ex. EBS Pacific Ocean perch CPUE and akgfmaps map

Pacific Ocean perch catch-per-unit-effort estimates for EBS in 2021 from GAP_PRODUCTS.AKFIN_CPUE and map constructed using akgfmaps. Here, we'll use AKFIN HAUL and CRUISES data also included in this repo, for convenience, though they are very similar to their RACEBASE analogs.

```
dat <- RODBC::sqlQuery(channel = channel,
                        query =
"
-- Select columns for output data
SELECT
(cp.CPUE_KGKM2/100) CPUE_KGHA, -- akgfmaps is expecting hectares
hh.LATITUDE_DD_START LATITUDE,
hh.LONGITUDE_DD_START LONGITUDE

-- Use HAUL data to obtain LATITUDE & LONGITUDE and connect to cruisejoin
FROM GAP_PRODUCTS.AKFIN_CPUE cp
LEFT JOIN GAP_PRODUCTS.AKFIN_HAUL hh
ON cp.HAULJOIN = hh.HAULJOIN

-- Use CRUISES data to obtain YEAR and SURVEY_DEFINITION_ID
LEFT JOIN GAP_PRODUCTS.AKFIN_CRUISE cc
ON hh.CRUISEJOIN = cc.CRUISEJOIN

-- Filter data
WHERE cp.SPECIES_CODE = 30060
AND cc.SURVEY_DEFINITION_ID = 98
AND cc.YEAR = 2021;")

flextable::flextable(head(dat)) %>% theme_zebra()
```

Data SQL Query Examples:

Table 7.3.: Ex. 6: EBS Pacific Ocean perch CPUE and akgfmaps map.

CPUE_- KGHA	LATITUDE	LONGITUDE
0.0000000	58.75863	-174.9285
0.2813533	57.32545	-173.3217
0.0000000	57.64161	-172.7963
0.0000000	59.67831	-172.5754
0.0000000	60.96936	-174.8760
0.0000000	58.64012	-173.5922

```
# devtools::install_github("afsc-gap-products/akgfmaps", build_vignettes = TRUE)
library(akgfmaps)

figure <- akgfmaps::make_idw_map(
  x = dat, # Pass data as a data frame
  region = "bs.south", # Predefined EBS area
  set.breaks = "jenks", # Gets Jenks breaks from classint::classIntervals()
  in.crs = "+proj=longlat", # Set input coordinate reference system
  out.crs = "EPSG:3338", # Set output coordinate reference system
  grid.cell = c(20000, 20000), # 20x20km grid
  key.title = "Pacific Ocean perch") # Include in the legend title

[inverse distance weighted interpolation]
[inverse distance weighted interpolation]

figure$plot +
  ggplot2::guides(fill=guide_legend(title = "Pacific Ocean perch\nCPUE (kg/km2)")) |>
  change_fill_color(new.scheme = "grey", show.plot = FALSE)
```

Data SQL Query Examples:

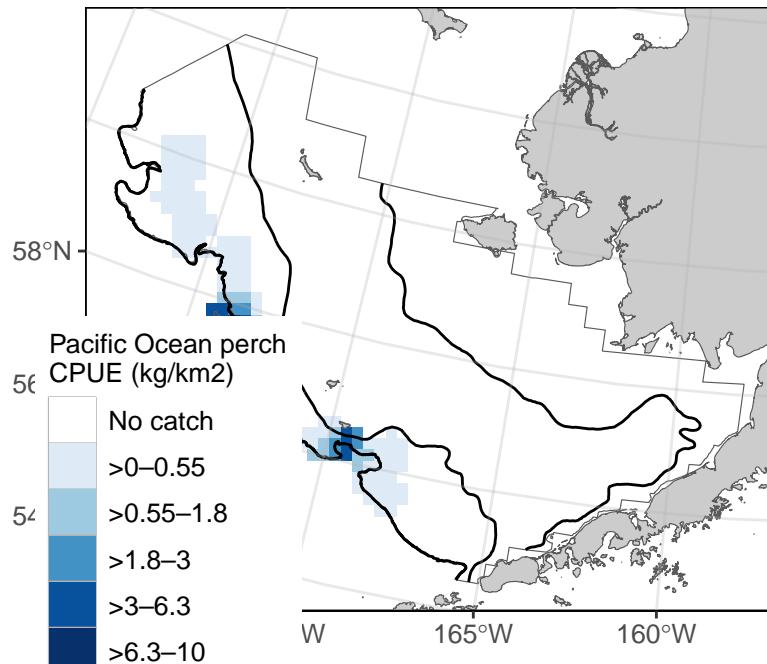


Figure 7.1.: Ex. 6: EBS Pacific Ocean perch CPUE and akfmaps map.

7.0.5. Ex. GOA Pacific Ocean perch biomass and abundance

Biomass and abundance for Pacific Ocean perch from 1990 – 2023 for the western/central/eastern GOA management areas as well as for the entire region.

```
dat <- RODBC::sqlQuery(channel = channel,
                        query =
"
-- Manipulate data to join to
WITH FILTERED_STRATA AS (
SELECT AREA_ID, DESCRIPTION FROM GAP_PRODUCTS.AKFIN_AREA
WHERE TYPE in ('REGULATORY_AREA', 'REGION')
AND SURVEY_DEFINITION_ID = 47)

-- Select columns for output data
SELECT
BIOMASS_MT,
POPULATION_COUNT,
```

Data SQL Query Examples:

```

YEAR,
DESCRIPTION

-- Identify what tables to pull data from
FROM GAP_PRODUCTS.AKFIN_BIOMASS BIOMASS
JOIN FILTERED_STRATA STRATA
ON STRATA.AREA_ID = BIOMASS.AREA_ID

-- Filter data results
WHERE BIOMASS.SPECIES_CODE = 30060")

dat0 <- dat %>%
  janitor::clean_names() %>%
  dplyr::select(biomass_mt, population_count, year, area = description) %>%
  pivot_longer(cols = c("biomass_mt", "population_count"),
               names_to = "var",
               values_to = "val") %>%
  dplyr::mutate(
    val = ifelse(var == "biomass_mt", val/1e6, val/1e9),
    var = ifelse(var == "biomass_mt", "Biomass (Mmt)", "Population (B)"),
    area = gsub(x = area, pattern = " - ", replacement = "\n"),
    area = gsub(x = area, pattern = ": ", replacement = "\n"),
    type = sapply(X = strsplit(x = area, split = "\n", fixed = TRUE), `[[`, 2)) %>%
  dplyr::arrange(type) %>%
  dplyr::mutate(
    area = factor(area, levels = unique(area), labels = unique(area), ordered = TRUE))

flextable::flextable(head(dat)) %>%
  theme_zebra() %>%
  flextable::colformat_num(x = ., j = "YEAR", big.mark = "")

```

Table 7.4.: Ex. 1: GOA Pacific Ocean perch biomass and abundance.

BIOMASS_POPULATI MT	COUNT	YEAR	DESCRIPTION
483,622.683	3,902,16	1993	GOA Region: All Strata

Data SQL Query Examples:

BIOMASS_POPULATI MT COUNT	YEARDESCRIPTION
483,622.6833,902,161	GOA 1993Region: All Strata
771,412.81,252,616,0	GOA 1996Region: All Strata
771,412.81,252,616,603	GOA 1996Region: All Strata
727,063.51,212,034,9	GOA 1999Region: All Strata
727,063.51,212,034,913	GOA 1999Region: All Strata

```
# install.packages("scales")
library(scales)
figure <- ggplot2::ggplot(
  dat = dat0,
  mapping = aes(x = year, y = val, color = type)) +
  ggplot2::geom_point(size = 3) +
  ggplot2::facet_grid(cols = vars(area), rows = vars(var), scales = "free_y") +
  ggplot2::scale_x_continuous(name = "Year", n.breaks = 3) +
  ggplot2::scale_y_continuous(name = "Estimate", labels = comma) +
  ggplot2::labs(title = 'GOA Pacific Ocean perch biomass and abundance 1990 - 2023') +
  ggplot2::guides(color=guide_legend(title = "Region Type"))+
  ggplot2::scale_color_grey() +
  ggplot2::theme_bw() +
  ggplot2::theme(legend.direction = "horizontal",
                legend.position = "bottom")

figure
```

Data SQL Query Examples:

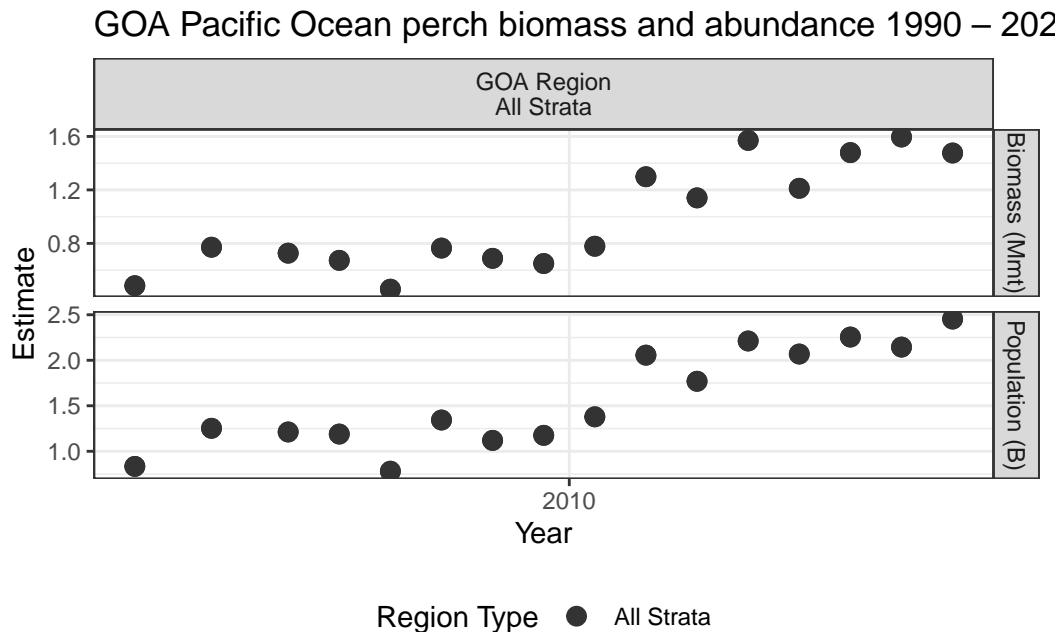


Figure 7.2.: Ex. 1: GOA Pacific Ocean perch biomass and abundance.

7.0.6. Ex. AI rock sole size compositions and ridge plot

Northern and Southern rock sole size composition data from 1991 – 2022 for the Aleutian Islands, with Ridge plot from `ggridges`.

```
dat <- RODBC::sqlQuery(channel = channel,
                        query =
"
-- Manipulate data to join to
WITH FILTERED_STRATA AS (
SELECT
AREA_ID,
DESCRIPTION

-- Identify what tables to pull data from
FROM GAP_PRODUCTS.AKFIN_AREA
WHERE TYPE = 'REGION'
AND SURVEY_DEFINITION_ID = 52)
```

Data SQL Query Examples:

```
-- Select columns for output data
SELECT
LENGTH_MM,
YEAR
FROM GAP_PRODUCTS.AKFIN_SIZECOMP SIZECOMP
JOIN FILTERED_STRATA STRATA
ON STRATA.AREA_ID = SIZECOMP.AREA_ID

-- Filter data results
WHERE SIZECOMP.SURVEY_DEFINITION_ID IN 52
AND SIZECOMP.SPECIES_CODE IN (10261, 10262)")
```

```
dat0 <- dat %>%
  janitor::clean_names() %>%
  dplyr::mutate(length_cm = length_mm/10)
flextable::flextable(head(dat)) %>%
  theme_zebra() %>%
  flextable::colformat_num(x = ., j = "YEAR", big.mark = "")
```

Table 7.5.: Ex. 2: Al Rock sole size compositions and ridge plot.

LENGTH_MM	YEAR
180	2014
190	2014
200	2014
210	2014
220	2014
230	2014

```
# install.packages("ggridges")
library(ggridges)
figure <-
  ggplot2::ggplot(
    data = dat0,
```

Data SQL Query Examples:

```
mapping = aes(x = length_cm, y = as.factor(year), fill = stat(x))) +  
ggridges::theme_ridges(center_axis_labels = TRUE) +  
ggridges::geom_density_ridges_gradient(scale = 4, show.legend = FALSE) +  
ggplot2::scale_y_discrete(name = "Year", expand = c(0.01, 0)) +  
ggplot2::scale_x_continuous(name = "Length (cm)", expand = c(0.01, 0)) +  
# ggplot2::scale_fill_grey() +  
ggplot2::labs(title = 'AI Rock sole Size Compositions 1991 - 2022')
```

```
figure
```

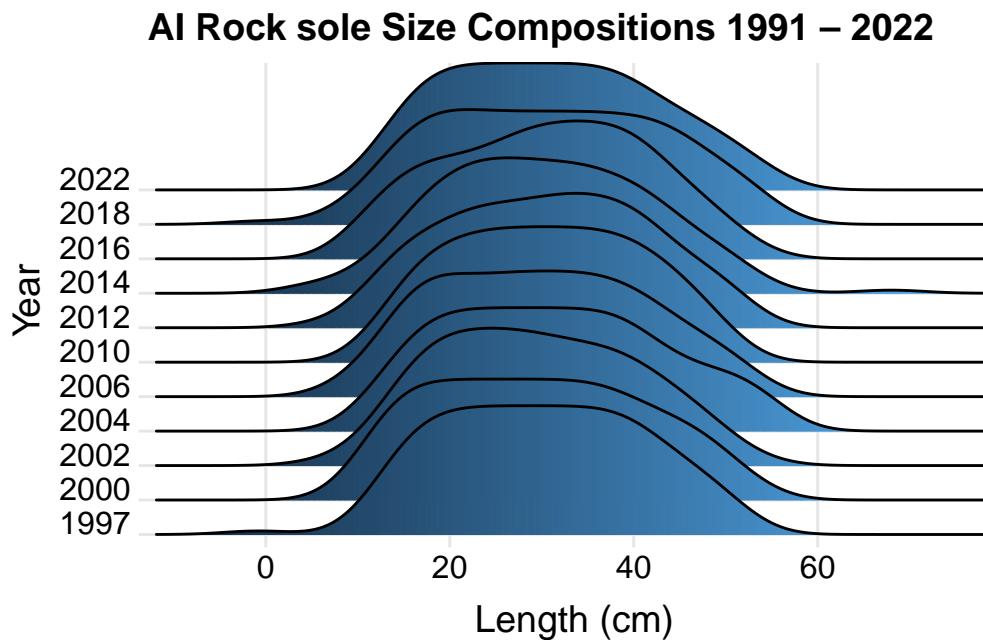


Figure 7.3.: Ex. 2: AI Rock sole size compositions and ridge plot.

7.0.7. Ex. EBS Walleye Pollock Age Compositions and Age Pyramid

Walleye pollock age composition for the EBS Standard Area from 1982 – 2022 and the EBS + NW Area from 1987 – 2022, with age pyramid plot.

```
dat <- RODBC::sqlQuery(channel = channel,  
query =
```

Data SQL Query Examples:

```
""
-- Manipulate data to join to
WITH FILTERED_STRATA AS (
SELECT
AREA_ID,
DESCRIPTION
FROM GAP_PRODUCTS.AKFIN_AREA
WHERE TYPE = 'REGION' AND
SURVEY_DEFINITION_ID = 98)

-- Select columns for output data
SELECT
AGECOMP.AGE,
AGECOMP.POPULATION_COUNT,
AGECOMP.SEX

-- Identify what tables to pull data from
FROM GAP_PRODUCTS.AKFIN_AGECOMP AGECOMP
JOIN FILTERED_STRATA STRATA
ON STRATA.AREA_ID = AGECOMP.AREA_ID

-- Filter data results
WHERE SPECIES_CODE = 21740
AND AGE >= 0")
```

```
dat0 <- dat %>%
  janitor::clean_names() %>%
  dplyr::filter(sex %in% c(1,2)) %>%
  dplyr::mutate(
    sex = ifelse(sex == 1, "M", "F"),
    population_count = # change male population to negative
      ifelse(sex=="M", population_count*(-1), population_count*1)/1e9)

flextable::flextable(head(dat)) %>% theme_zebra()
```

Data SQL Query Examples:

Table 7.6.: Ex. 3: EBS Walleye Pollock Age Compositions and Age Pyramid.

AGE	POPULATI COUNT	SEX
9	39,371	3
10	32,156	3
11	15,200	3
12	9,976	3
13	1,957	3
	1131,950,343	1

```
figure <- ggplot2::ggplot(  
  data = dat0,  
  mapping =  
    aes(x = age,  
        y = population_count,  
        fill = sex)) +  
  ggplot2::scale_fill_grey() +  
  ggplot2::geom_bar(stat = "identity") +  
  ggplot2::coord_flip() +  
  ggplot2::scale_x_continuous(name = "Age") +  
  ggplot2::scale_y_continuous(name = "Population (billions)", labels = abs) +  
  ggplot2::ggtitle(label = "EBS Walleye Pollock Age Compositions 1982 - 2022") +  
  ggplot2::guides(fill = guide_legend(title = "Sex"))+  
  ggplot2::theme_bw()  
  
figure
```

Data SQL Query Examples:

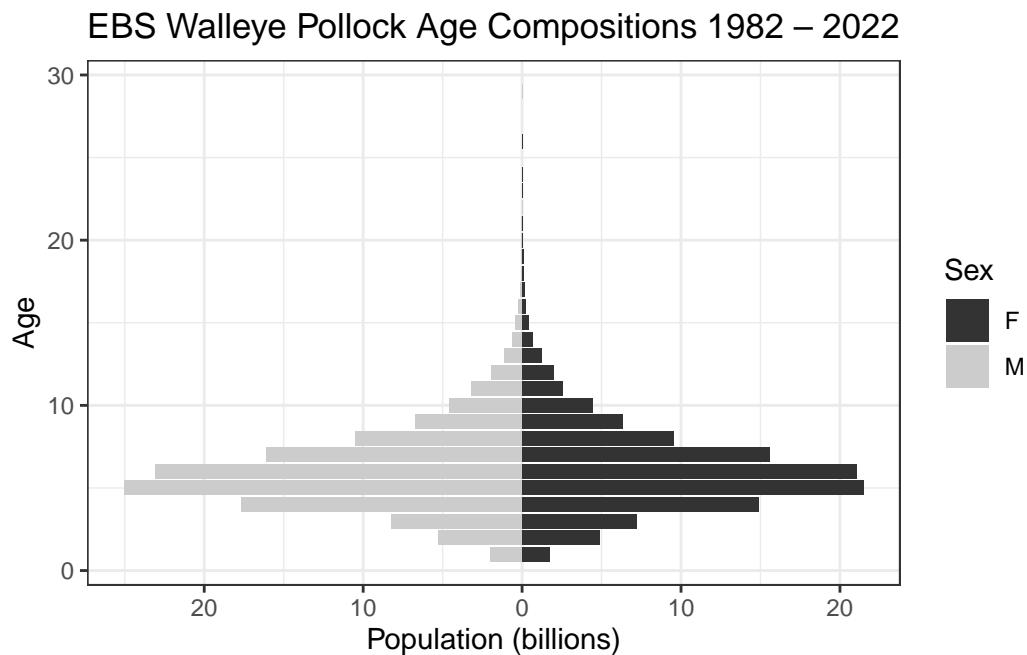


Figure 7.4.: Ex. 3: EBS Walleye Pollock Age Compositions and Age Pyramid.

7.0.8. Ex. NBS Pacific cod biomass and abundance

Pacific cod biomass and abundance data for the NBS by stratum.

```
dat <- RODBC::sqlQuery(channel = channel,
                        query =
"
-- Manipulate data to join to
WITH FILTERED_STRATA AS (
SELECT
AREA_ID,
AREA_NAME,
DESCRIPTION
FROM GAP_PRODUCTS.AKFIN_AREA
WHERE TYPE in ('STRATUM') AND
SURVEY_DEFINITION_ID = 143)

-- Select columns for output data
```

Data SQL Query Examples:

```

SELECT
BIOMASS.BIOMASS_MT,
BIOMASS.POPULATION_COUNT,
BIOMASS.YEAR,
STRATA.AREA_NAME

-- Identify what tables to pull data from
FROM GAP_PRODUCTS.AKFIN_BIOMASS BIOMASS
JOIN FILTERED_STRATA STRATA
ON STRATA.AREA_ID = BIOMASS.AREA_ID

-- Filter data results
WHERE BIOMASS.SURVEY_DEFINITION_ID IN 143
AND BIOMASS.SPECIES_CODE = 21720")

```

```

dat0 <- dat %>%
  janitor::clean_names() %>%
  dplyr::select(biomass_mt, population_count, year, area = area_name) %>%
  pivot_longer(cols = c("biomass_mt", "population_count"),
               names_to = "var",
               values_to = "val") %>%
  dplyr::mutate(
    val = ifelse(var == "biomass_mt", val/1e6, val/1e9),
    var = ifelse(var == "biomass_mt", "Biomass (Mmt)", "Population (B)"),
    area = factor(area, levels = unique(area), labels = unique(area), ordered = TRUE))
flextable::flextable(head(dat)) %>%
  theme_zebra() %>%
  flextable::colformat_num(x = ., j = "YEAR", big.mark = "")

```

Table 7.7.: Ex. 4: NBS Pacific cod biomass and abundance.

BIOMASS_POPULATI MT	YEAR	AREA_- NAME
COUNT		
95,849.98	2021	Inner Domain
368,767,498		
107,096.73	2019	Inner Domain
1002,734,142		

Data SQL Query Examples:

BIOMASS_POPULATI MT COUNT	YEAR	AREA_- NAME
76,708.43339,605,860	2023	Inner Domain
132,490.1506,187,245	2017	Inner Domain
96,500.69760,433,135	2022	Inner Domain
7,462.559 4,724,153	2010	Inner Domain

```
figure <- ggplot2::ggplot(  
  dat = dat0,  
  mapping = aes(y = val, x = year, fill = area)) +  
  ggplot2::geom_bar(position="stack", stat="identity") +  
  ggplot2::facet_grid(rows = vars(var), scales = "free_y") +  
  ggplot2::scale_y_continuous(name = "Estimate", labels = comma) +  
  ggplot2::scale_x_continuous(name = "Year", breaks = unique(dat0$year)) +  
  ggplot2::labs(title = 'NBS Pacific cod biomass and abundance by stratum') +  
  ggplot2::guides(fill=guide_legend(title = "Region Type"))+  
  ggplot2::scale_fill_grey() +  
  ggplot2::theme_bw() +  
  ggplot2::theme(legend.direction = "horizontal",  
                legend.position = "bottom")  
  
figure
```

Data SQL Query Examples:

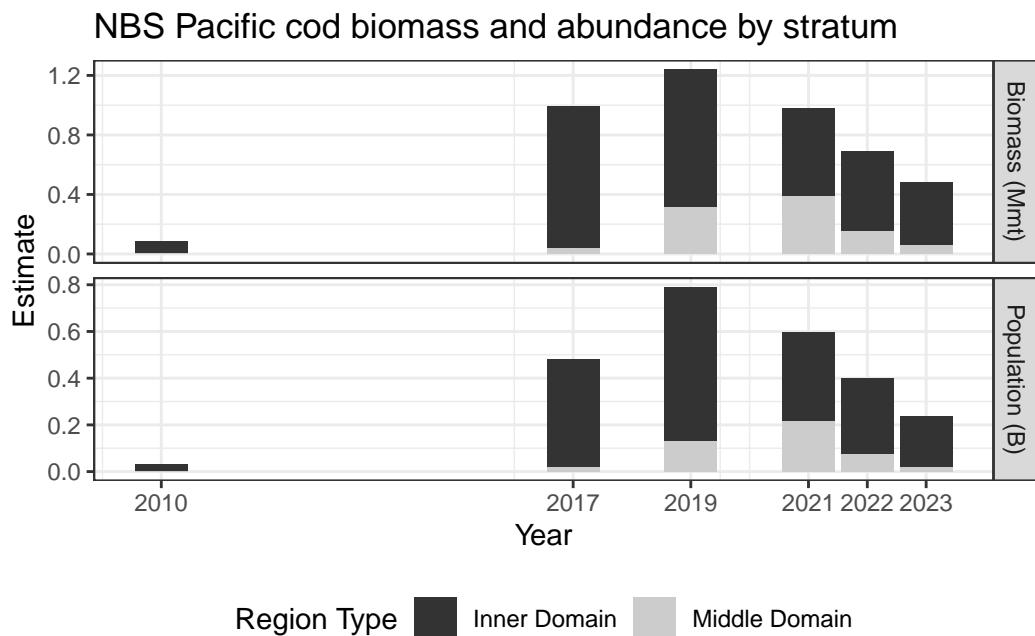


Figure 7.5.: Ex. 4: NBS Pacific cod biomass and abundance.

7.0.9. Ex. GOA Pacific Ocean perch biomass and line plot

Pacific Ocean perch biomass totals for GOA between 1984-2021 from GAP_PRODUCTS.AKFIN_BIOMASS

```
dat <- RODBC::sqlQuery(channel = channel,
                        query =
"
-- Select columns for output data
SELECT
SURVEY_DEFINITION_ID,
BIOMASS_MT,
BIOMASS_VAR,
YEAR

-- Identify what tables to pull data from
FROM GAP_PRODUCTS.AKFIN_BIOMASS
```

Data SQL Query Examples:

```
-- Filter data results
WHERE SPECIES_CODE = 30060
AND SURVEY_DEFINITION_ID = 47
AND AREA_ID = 99903
AND YEAR BETWEEN 1984 AND 2023;") %>%
  janitor::clean_names() %>%
  dplyr::mutate(biomass_kmt = biomass_mt/1000,
                # **approximate** 95% confidence interval
                biomass_kci_up = (biomass_mt + (2*sqrt(biomass_var)))/1000,
                biomass_kci_dw = (biomass_mt - (2*sqrt(biomass_var)))/1000)

flextable::flextable(head(dat)) %>%
  theme_zebra() %>%
  flextable::colformat_num(x = ., j = "year", big.mark = "")
```

Table 7.8.: Ex. 5: GOA Pacific Ocean perch biomass and line plot.

survey_definition_id	biomass_mt	biomass_var	year	biomass_kmt	biomass_kci_up	biomass_kci_dw
47	483,622.611,803,384		1993	483.6226	700.9093	266.33581
47	771,412.841,434,152,202		1996	771.41281,178.5204	364.30515	
47	727,063.5150,983,54		1999	727.06351,504.1955	-50.06854	
47	673,155.149,285,342,922		2001	673.15511,117.1611	229.14901	
47	457,421.65,186,126,!		2003	457.4216	601.4511	313.39204
47	764,901.421,499,807,010		2005	764.90141,058.1577	471.64517	

```
a_mean <- dat %>%
  dplyr::group_by(survey_definition_id) %>%
  dplyr::summarise(biomass_kmt = mean(biomass_kmt, na.rm = TRUE),
                    minyr = min(year, na.rm = TRUE),
                    maxyr = max(year, na.rm = TRUE))

figure <-
  ggplot(data = dat,
         mapping = aes(x = year,
```

Data SQL Query Examples:

```
        y = biomass_kmt)) +
ggplot2::geom_point(size = 2.5, color = "grey40") +
ggplot2::scale_x_continuous(
  name = "Year",
  labels = scales::label_number(
    accuracy = 1,
    big.mark = ""))
ggplot2::scale_y_continuous(
  name = "Biomass (Kmt)",
  labels = comma) +
ggplot2::geom_segment(
  data = a_mean,
  mapping = aes(x = minyr,
                 xend = maxyr,
                 y = biomass_kmt,
                 yend = biomass_kmt),
  linetype = "dashed",
  linewidth = 2) +
ggplot2::geom_errorbar(
  mapping = aes(ymin = biomass_kci_dw, ymax = biomass_kci_up),
  position = position_dodge(.9),
  alpha = 0.5, width=.2) +
ggplot2::ggttitle(
  label = "GOA Pacific Ocean Perch Biomass 1984-2021",
  subtitle = paste0("Mean = ",
                   formatC(x = a_mean$biomass_kmt,
                           digits = 2,
                           big.mark = ",",
                           format = "f"),
                   " Kmt")) +
ggplot2::theme_bw()

figure
```

Data SQL Query Examples:

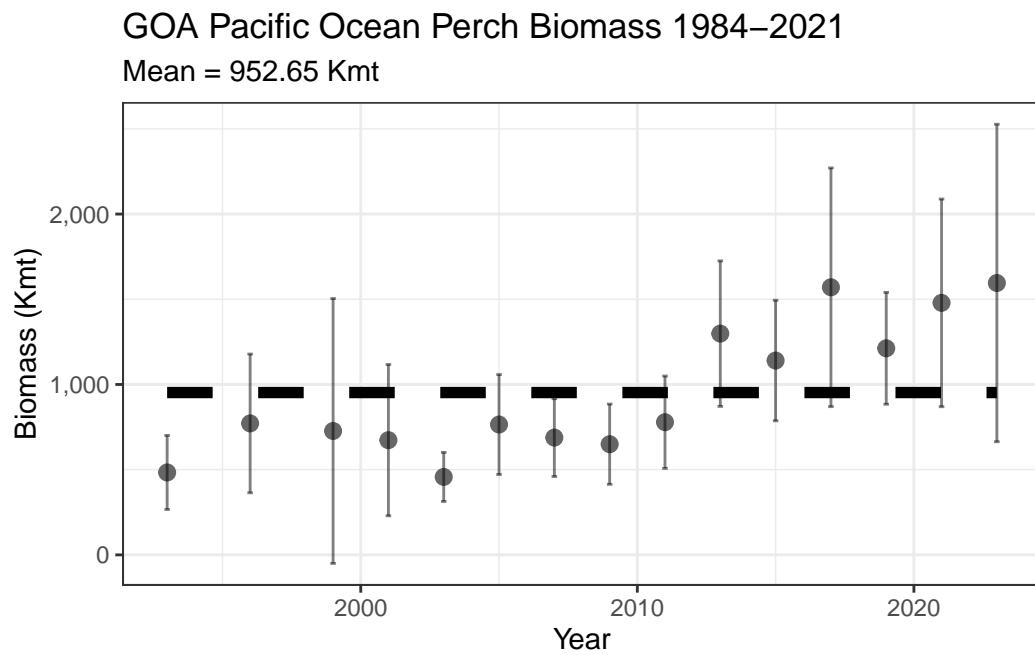


Figure 7.6.: Ex. 5: GOA Pacific Ocean perch biomass and line plot.

8. Access API data using R

Use the below function to pull AKFIN data through AKFIN's API.

```
# load libraries
library(dplyr)
library(magrittr)
library(httr)
library(flextable)

# tell R to not use scientific notation
options(scipen=999)

# function for pulling data from the api using the httr package
get_gap_biomass<-function(area_id, species_code) {
  # paste(... collapse=",") puts commas between vector elements
  area_id <- paste(area_id, collapse = ",")
  species_code <- paste(species_code, collapse = ",")
  # httr code, parameters are after the '?'
  httr::content(
    httr::GET(paste0("https://apex.psmfc.org/akfin/data_marts/akmp/gap_biomass?area_id=",
                     area_id,
                     "&species_code=",
                     species_code)),
    type = "application/json") %>%
    # convert to data frame
    bind_rows()
}
```

8.1. Ex. 1: Load lingcod data

8. Access API data using R

```
lingcod_biomass <- get_gap_biomass(area_id=c(40, 41), species_code=21910)
flextable::flextable(head(lingcod_biomass)) %>%
  flextable::theme_zebra()
```

Part IV.

Public Data (FOSS)

The final, validated survey data are publicly accessible soon after surveys are completed on the Fisheries One Stop Shop (FOSS) platform. This data includes catch, haul, and environmental data collected at each station. On the FOSS data platform, users can interactively select, view, and download data. Descriptive documentation and user-examples are available on the metadata page.

This data contains all of the catch, environmental, and haul data from the fisheries-independent Groundfish and Shellfish Assessment Program surveys in the Bering Sea, Aleutian Islands, and Gulf of Alaska. This data is sought after by the general public, private entities, and NOAA partners alike, including tribal organizations, K-12 classrooms, academic institutions, for-profit groups, and non-profit groups. This data is compiled and approved once a year after each summer survey season and is available for open access.

Part V.

Collaborators and data users

Access Constraints

Below are a few packages and products currently using this data. If you have developed a product, performed an analysis, or exhibited this data in any way, reach out so we can showcase your hard work.

- **NOAA Fisheries Distribution Mapping and Analysis Portal;** NOAA Fisheries Office of Science and Technology
- **Pull data with python and explore the in-browser visualization tool. Reference their example Python notebook;** The Eric and Wendy Schmidt Center for Data Science and the Environment at UC Berkeley, including sam.pottinger@berkeley.edu, ccmartinez@berkeley.edu, gzarpellon@berkeley.edu, and kkoy@berkeley.edu.

Access Constraints

User Constraints: Users must read and fully comprehend the metadata prior to use. Data should not be used beyond the limits of the source scale. Acknowledgment of AFSC Groundfish Assessment Program, as the source from which these data were obtained, in any publications and/or other representations of these data, is suggested.

General questions and more specific data requests can be sent to afsc.gap.metadata@noaa.gov or submitted as an issue on our GitHub Organization. The version of this data used for stock assessments can be found through the Alaska Fisheries Information Network (AKFIN). For questions about the eastern Bering Sea surveys, contact Duane Stevenson (Duane.Stevenson@noaa.gov). For questions about the Gulf of Alaska or Aleutian Islands surveys, contact Ned Laman (Ned.Laman@noaa.gov). For questions specifically about crab data in any region, contact Mike Litzow (Mike.Litzow@noaa.gov), the Shellfish Assessment Program lead.

For questions, comments, and concerns specifically about the Fisheries One Stop Shop (FOSS) platform, please contact us using the Comments page on the FOSS webpage.

Cite this data

Use the below bibtext citations, as cited in our group's citation repository for citing the data created and maintained in this repo (NOAA Fisheries Alaska Fisheries Science Center, 2023). Add "note = {Accessed: mm/dd/yyyy}" to append the day this data was accessed.

Cite this data

```
@misc{FOSSAFSCData,
  author = {{NOAA Fisheries Alaska Fisheries Science Center}},
  year = {2023},
  title = {Fisheries One Stop Shop Public Data: RACE Division Bottom Trawl Survey Data Query},
  howpublished = {\url{https://www.fisheries.noaa.gov/foss}},
  publisher = {{U.S. Dep. Commer.}},
  copyright = {Public Domain}
}
```

9. Data description

The Resource Assessment and Conservation Engineering Division (RACE) Groundfish Assessment Program (GAP) of the Alaska Fisheries Science Center (AFSC) conducts fisheries-independent bottom trawl surveys to monitor the condition of the demersal fish and crab stocks of Alaska. These data are developed to describe the temporal distribution and abundance of commercially and ecologically important groundfish species, examine the changes in the species composition of the fauna over time and space, and describe the physical environment of the groundfish habitat.

There are no legal restrictions on access to the data. They reside in the public domain and can be freely distributed. Users must read and fully comprehend the metadata prior to use. Data should not be used beyond the limits of the source scale. Acknowledgement of NOAA, as the source from which these data were obtained, in any publications and/or other representations of these data, is suggested. These data are compiled and approved annually after each summer survey season. The data from previous years are unlikely to change substantially once published.

These data are zero-filled (presence and absence) observations from surveys conducted on fishing vessels. These surveys monitor trends in distribution and abundance of groundfish, crab, and bottom-dwelling species in Alaska's marine ecosystems. These data include estimates of catch-per-unit-effort (CPUE) for all identified species for index stations. Some survey data are excluded, such as non-standard stations, surveys completed in earlier years using different/non-standard gear, and special tows and non-standard data collections.

Though not included in the public data, these surveys also collect oceanographic and environmental data, and biological data such as length, weight, stomach contents (to learn more about diet), otoliths (fish ear bones to learn about age), and tissue samples for genetic analysis, all of which can be shared upon special request. Also not included in the public data are estimated biomass (average total weight of all fish and crabs sampled) of crabs and groundfish that support the creation of annual stock assessments.

9. Data description

9.1. Data tables

9.1.1. FOSS_CATCH

These datasets, FOSS_CATCH, FOSS_CPUE_PRESONLY, FOSS_HAUL, and FOSS_SPECIES, when full joined by the HAULJOIN variable, includes zero-filled (presence and absence) observations and catch-per-unit-effort (CPUE) estimates for all identified species at for index stations. These tables were created by the Resource Assessment and Conservation Engineering Division (RACE) Groundfish Assessment Program (GAP) of the Alaska Fisheries Science Center (AFSC). There are legal restrictions on access to the data. These data are not intended for public dissemination and should not be shared without the explicit written consent of the data managers and owners (NOAA Fisheries). The GitHub repository for the scripts that created this code can be found at https://github.com/afsc-gap-products/gap_products. For more information about codes used in the tables, please refer to the survey code books (<https://www.fisheries.noaa.gov/resource/document/groundfish-survey-species-code-manual-and-data-codes-manual>). These data were last updated October 17, 2023.

Number of rows: 928,931

Number of columns: 7

Column name from data

Descriptive column Name

Units

Oracle data type

Column description

COUNT

Taxon count

count, whole number resolution

NUMBER(38,0)

Total whole number of individuals caught in haul.

CPUE_KGKM2

Weight CPUE (kg/km²)

9. Data description

kilograms per kilometers squared

NUMBER(38,6)

Catch weight (kilograms) per unit effort (area swept by the net, units square kilometers).

CPUE_NOKM2

Number CPUE (no/km2)

count per kilometers squared

NUMBER(38,6)

Numerical catch per unit effort (area swept by the net, units square kilometers).

HAULJOIN

Haul ID

ID code

NUMBER(38,0)

This is a unique numeric identifier assigned to each (vessel, cruise, and haul) combination.

SPECIES_CODE

Taxon code

ID code

NUMBER(38,0)

The species code of the organism associated with the 'common_name' and 'scientific_name' columns. For a complete species list, review the code books.

TAXON_CONFIDENCE

Taxon confidence rating

category

VARCHAR2(255 BYTE)

Confidence in the ability of the survey team to correctly identify the taxon to the specified level, based solely on identification skill (e.g., not likelihood of a taxon being caught at that station on a location-by-location basis). Quality codes follow: '**High**': High confidence and consistency. Taxonomy is stable and reliable at this level, and field identification characteristics are well known and reliable. '**Moderate**': Moderate

9. Data description

confidence. Taxonomy may be questionable at this level, or field identification characteristics may be variable and difficult to assess consistently. **'Low'**: Low confidence. Taxonomy is incompletely known, or reliable field identification characteristics are unknown. Documentation: Species identification confidence in the eastern Bering Sea shelf survey (1982-2008), Species identification confidence in the eastern Bering Sea slope survey (1976-2010), and Species identification confidence in the Gulf of Alaska and Aleutian Islands surveys (1980-2011).

WEIGHT_KG

Sample or taxon weight (kg)

kilograms

NUMBER(38,3)

Weight (thousandths of a kilogram) of individuals in a haul by taxon.

9.1.2. FOSS_CPUE_PRESONLY

These datasets, FOSS_CATCH, FOSS_CPUE_PRESONLY, FOSS_HAUL, and FOSS_SPECIES, when full joined by the HAULJOIN variable, includes zero-filled (presence and absence) observations and catch-per-unit-effort (CPUE) estimates for all identified species at for index stations. These tables were created by the Resource Assessment and Conservation Engineering Division (RACE) Groundfish Assessment Program (GAP) of the Alaska Fisheries Science Center (AFSC). There are legal restrictions on access to the data. These data are not intended for public dissemination and should not be shared without the explicit written consent of the data managers and owners (NOAA Fisheries). The GitHub repository for the scripts that created this code can be found at https://github.com/afsc-gap-products/gap_products. For more information about codes used in the tables, please refer to the survey code books (<https://www.fisheries.noaa.gov/resource/document/groundfish-survey-species-code-manual-and-data-codes-manual>). These data were last updated October 17, 2023.

Number of rows: 928,931

Number of columns: 38

Column name from data

Descriptive column Name

Units

9. Data description

Oracle data type

Column description

AREA_SWEPT_KM2

Area swept (km)

kilometers

NUMBER(38,6)

The area the net covered while the net was fishing (kilometers squared), defined as the distance fished times the net width.

BOTTOM_TEMPERATURE_C

Bottom temperature (degrees celsius)

degrees Celsius

NUMBER(38,1)

Bottom temperature (tenths of a degree Celsius); NA indicates removed or missing values.

COMMON_NAME

Taxon common name

text

VARCHAR2(255 BYTE)

The common name of the marine organism associated with the 'scientific_name' and 'species_code' columns. For a complete species list, review the code books.

COUNT

Taxon count

count, whole number resolution

NUMBER(38,0)

Total whole number of individuals caught in haul.

CPUE_KGKM2

Weight CPUE (kg/km2)

kilograms per kilometers squared

9. Data description

NUMBER(38,6)

Catch weight (kilograms) per unit effort (area swept by the net, units square kilometers).

CPUE_NOKM2

Number CPUE (no/km2)

count per kilometers squared

NUMBER(38,6)

Numerical catch per unit effort (area swept by the net, units square kilometers).

CRUISE

Cruise ID

ID code

NUMBER(38,0)

This is a six-digit integer identifying the cruise number of the form: YYYY99 (where YYYY = year of the cruise; 99 = 2-digit number and is sequential; 01 denotes the first cruise that vessel made in this year, 02 is the second, etc.).

CRUISEJOIN

Cruise ID

ID code

NUMBER(38,0)

Unique interger ID assigned to each survey, vessel, and year combination.

DATE_TIME

Date and time

MM/DD/YYYY HH::MM

DATE

The date (MM/DD/YYYY) and time (HH:MM) of the haul.

DEPTH_M

Depth (m)

degrees Celsius

9. Data description

NUMBER(38,1)

Bottom depth (meters).

DISTANCE_FISHED_KM

Distance fished (km)

degrees Celsius

NUMBER(38,3)

Distance the net fished (thousandths of kilometers).

DURATION_HR

Tow duration (decimal hr)

hours

NUMBER(38,1)

This is the elapsed time between start and end of a haul (decimal hours).

HAUL

Haul number

ID code

NUMBER(38,0)

This number uniquely identifies a sampling event (haul) within a cruise. It is a sequential number, in chronological order of occurrence.

HAULJOIN

Haul ID

ID code

NUMBER(38,0)

This is a unique numeric identifier assigned to each (vessel, cruise, and haul) combination.

ID_RANK

Lowest taxonomic rank

text

VARCHAR2(255 BYTE)

9. Data description

Lowest taxonomic rank of a given species entry.

ITIS

ITIS taxonomic serial number

ID code

NUMBER(38,0)

Species code as identified in the Integrated Taxonomic Information System (<https://itis.gov/>).

LATITUDE_DD_END

End latitude (decimal degrees)

decimal degrees

NUMBER(38,6)

Latitude (one hundred thousandth of a decimal degree) of the end of the haul.

LATITUDE_DD_START

Start latitude (decimal degrees)

decimal degrees

NUMBER(38,6)

Latitude (one hundred thousandth of a decimal degree) of the start of the haul.

LONGITUDE_DD_END

End longitude (decimal degrees)

decimal degrees

NUMBER(38,6)

Longitude (one hundred thousandth of a decimal degree) of the end of the haul.

LONGITUDE_DD_START

Start longitude (decimal degrees)

decimal degrees

NUMBER(38,6)

Longitude (one hundred thousandth of a decimal degree) of the start of the haul.

NET_HEIGHT_M

9. Data description

Net height (m)

meters

NUMBER(38,1)

Measured or estimated distance (meters) between footrope and headrope of the trawl.

NET_WIDTH_M

Net width (m)

meters

NUMBER(38,1)

Measured or estimated distance (meters) between wingtips of the trawl.

PERFORMANCE

Haul performance code

category

NUMBER(38,0)

This denotes what, if any, issues arose during the haul. For more information, review the code books.

SCIENTIFIC_NAME

Taxon scientific name

text

VARCHAR2(255 BYTE)

The scientific name of the organism associated with the 'common_name' and 'species_code' columns. For a complete taxon list, review the code books.

SPECIES_CODE

Taxon code

ID code

NUMBER(38,0)

The species code of the organism associated with the 'common_name' and 'scientific_name' columns. For a complete species list, review the code books.

SRVY

9. Data description

Survey

text abbreviated

VARCHAR2(255 BYTE)

Abbreviated survey names. The column 'srvy' is associated with the 'survey' and 'survey_id' columns. Northern Bering Sea (NBS), Southeastern Bering Sea (EBS), Bering Sea Slope (BSS), Gulf of Alaska (GOA), Aleutian Islands (AI).

STATION

Station ID

ID code

VARCHAR2(255 BYTE)

Alpha-numeric designation for the station established in the design of a survey.

STRATUM

Stratum ID

ID code

NUMBER(10,0)

RACE database statistical area for analyzing data. Strata were designed using bathymetry and other geographic and habitat-related elements. The strata are unique to each survey region. Stratum of value 0 indicates experimental tows.

SURFACE_TEMPERATURE_C

Surface temperature (Degrees Celsius)

degrees Celsius

NUMBER(38,1)

Surface temperature (tenths of a degree Celsius); NA indicates removed or missing values.

SURVEY

Survey Name

text

VARCHAR2(255 BYTE)

9. Data description

Name and description of survey. The column 'survey' is associated with the 'srvy' and 'survey_id' columns.

SURVEY_DEFINITION_ID

Survey ID

ID code

NUMBER(38,0)

This number uniquely identifies a survey. Name and description of survey. The column 'survey_id' is associated with the 'srvy' and 'survey' columns. For a complete list of surveys, review the code books.

SURVEY_NAME

NA

NA

NA

NA

TAXON_CONFIDENCE

Taxon confidence rating

category

VARCHAR2(255 BYTE)

Confidence in the ability of the survey team to correctly identify the taxon to the specified level, based solely on identification skill (e.g., not likelihood of a taxon being caught at that station on a location-by-location basis). Quality codes follow: '**HighModerateLow**

VESSEL_ID

Vessel ID

9. Data description

ID code

NUMBER(38,0)

ID number of the vessel used to collect data for that haul. The column 'vessel_id' is associated with the 'vessel_name' column. Note that it is possible for a vessel to have a new name but the same vessel id number. For a complete list of vessel ID codes, review the code books.

VESSEL_NAME

Vessel name

text

VARCHAR2(255 BYTE)

Name of the vessel used to collect data for that haul. The column 'vessel_name' is associated with the 'vessel_id' column. Note that it is possible for a vessel to have a new name but the same vessel id number. For a complete list of vessel ID codes, review the code books.

WEIGHT_KG

Sample or taxon weight (kg)

kilograms

NUMBER(38,3)

Weight (thousandths of a kilogram) of individuals in a haul by taxon.

WORMS

World Register of Marine Species Taxonomic Serial Number

ID code

NUMBER(38,0)

Species code as identified in the World Register of Marine Species (WoRMS) (<https://www.marinespecies.org/>).

YEAR

Survey year

year

NUMBER(10,0)

Year the observation (survey) was collected.

9. Data description

9.1.3. FOSS_HAUL

These datasets, FOSS_CATCH, FOSS_CPUE_PRESONLY, FOSS_HAUL, and FOSS_SPECIES, when full joined by the HAULJOIN variable, includes zero-filled (presence and absence) observations and catch-per-unit-effort (CPUE) estimates for all identified species at for index stations. These tables were created by the Resource Assessment and Conservation Engineering Division (RACE) Groundfish Assessment Program (GAP) of the Alaska Fisheries Science Center (AFSC). There are legal restrictions on access to the data. These data are not intended for public dissemination and should not be shared without the explicit written consent of the data managers and owners (NOAA Fisheries). The GitHub repository for the scripts that created this code can be found at https://github.com/afsc-gap-products/gap_products. For more information about codes used in the tables, please refer to the survey code books (<https://www.fisheries.noaa.gov/resource/document/groundfish-survey-species-code-manual-and-data-codes-manual>). These data were last updated October 17, 2023.

Number of rows: 32,626

Number of columns: 27

Column name from data

Descriptive column Name

Units

Oracle data type

Column description

AREA_SWEPT_KM2

Area swept (km)

kilometers

NUMBER(38,6)

The area the net covered while the net was fishing (kilometers squared), defined as the distance fished times the net width.

BOTTOM_TEMPERATURE_C

Bottom temperature (degrees celsius)

degrees Celsius

9. Data description

NUMBER(38,1)

Bottom temperature (tenths of a degree Celsius); NA indicates removed or missing values.

CRUISE

Cruise ID

ID code

NUMBER(38,0)

This is a six-digit integer identifying the cruise number of the form: YYYY99 (where YYYY = year of the cruise; 99 = 2-digit number and is sequential; 01 denotes the first cruise that vessel made in this year, 02 is the second, etc.).

CRUISEJOIN

Cruise ID

ID code

NUMBER(38,0)

Unique integer ID assigned to each survey, vessel, and year combination.

DATE_TIME

Date and time

MM/DD/YYYY HH::MM

DATE

The date (MM/DD/YYYY) and time (HH:MM) of the haul.

DEPTH_M

Depth (m)

degrees Celsius

NUMBER(38,1)

Bottom depth (meters).

DISTANCE_FISHED_KM

Distance fished (km)

degrees Celsius

9. Data description

NUMBER(38,3)

Distance the net fished (thousandths of kilometers).

DURATION_HR

Tow duration (decimal hr)

hours

NUMBER(38,1)

This is the elapsed time between start and end of a haul (decimal hours).

HAUL

Haul number

ID code

NUMBER(38,0)

This number uniquely identifies a sampling event (haul) within a cruise. It is a sequential number, in chronological order of occurrence.

HAULJOIN

Haul ID

ID code

NUMBER(38,0)

This is a unique numeric identifier assigned to each (vessel, cruise, and haul) combination.

LATITUDE_DD_END

End latitude (decimal degrees)

decimal degrees

NUMBER(38,6)

Latitude (one hundred thousandth of a decimal degree) of the end of the haul.

LATITUDE_DD_START

Start latitude (decimal degrees)

decimal degrees

NUMBER(38,6)

9. Data description

Latitude (one hundred thousandth of a decimal degree) of the start of the haul.

LONGITUDE_DD_END

End longitude (decimal degrees)

decimal degrees

NUMBER(38,6)

Longitude (one hundred thousandth of a decimal degree) of the end of the haul.

LONGITUDE_DD_START

Start longitude (decimal degrees)

decimal degrees

NUMBER(38,6)

Longitude (one hundred thousandth of a decimal degree) of the start of the haul.

NET_HEIGHT_M

Net height (m)

meters

NUMBER(38,1)

Measured or estimated distance (meters) between footrope and headrope of the trawl.

NET_WIDTH_M

Net width (m)

meters

NUMBER(38,1)

Measured or estimated distance (meters) between wingtips of the trawl.

PERFORMANCE

Haul performance code

category

NUMBER(38,0)

This denotes what, if any, issues arose during the haul. For more information, review the code books.

9. Data description

SRVY

Survey

text abbreviated

VARCHAR2(255 BYTE)

Abbreviated survey names. The column 'srvy' is associated with the 'survey' and 'survey_id' columns. Northern Bering Sea (NBS), Southeastern Bering Sea (EBS), Bering Sea Slope (BSS), Gulf of Alaska (GOA), Aleutian Islands (AI).

STATION

Station ID

ID code

VARCHAR2(255 BYTE)

Alpha-numeric designation for the station established in the design of a survey.

STRATUM

Stratum ID

ID code

NUMBER(10,0)

RACE database statistical area for analyzing data. Strata were designed using bathymetry and other geographic and habitat-related elements. The strata are unique to each survey region. Stratum of value 0 indicates experimental tows.

SURFACE_TEMPERATURE_C

Surface temperature (Degrees Celsius)

degrees Celsius

NUMBER(38,1)

Surface temperature (tenths of a degree Celsius); NA indicates removed or missing values.

SURVEY

Survey Name

text

VARCHAR2(255 BYTE)

9. Data description

Name and description of survey. The column 'survey' is associated with the 'srvy' and 'survey_id' columns.

SURVEY_DEFINITION_ID

Survey ID

ID code

NUMBER(38,0)

This number uniquely identifies a survey. Name and description of survey. The column 'survey_id' is associated with the 'srvy' and 'survey' columns. For a complete list of surveys, review the code books.

SURVEY_NAME

NA

NA

NA

NA

VESSEL_ID

Vessel ID

ID code

NUMBER(38,0)

ID number of the vessel used to collect data for that haul. The column 'vessel_id' is associated with the 'vessel_name' column. Note that it is possible for a vessel to have a new name but the same vessel id number. For a complete list of vessel ID codes, review the code books.

VESSEL_NAME

Vessel name

text

VARCHAR2(255 BYTE)

Name of the vessel used to collect data for that haul. The column 'vessel_name' is associated with the 'vessel_id' column. Note that it is possible for a vessel to have a new name but the same vessel id number. For a complete list of vessel ID codes, review the code books.

9. Data description

YEAR

Survey year

year

NUMBER(10,0)

Year the observation (survey) was collected.

9.1.4. FOSS_SPECIES

These datasets, FOSS_CATCH, FOSS_CPUE_PRESONLY, FOSS_HAUL, and FOSS_SPECIES, when full joined by the HAULJOIN variable, includes zero-filled (presence and absence) observations and catch-per-unit-effort (CPUE) estimates for all identified species at for index stations. These tables were created by the Resource Assessment and Conservation Engineering Division (RACE) Groundfish Assessment Program (GAP) of the Alaska Fisheries Science Center (AFSC). There are legal restrictions on access to the data. These data are not intended for public dissemination and should not be shared without the explicit written consent of the data managers and owners (NOAA Fisheries). The GitHub repository for the scripts that created this code can be found at https://github.com/afsc-gap-products/gap_products. For more information about codes used in the tables, please refer to the survey code books (<https://www.fisheries.noaa.gov/resource/document/groundfish-survey-species-code-manual-and-data-codes-manual>). These data were last updated October 17, 2023.

Number of rows: 1,936

Number of columns: 6

Column name from data

Descriptive column Name

Units

Oracle data type

Column description

COMMON_NAME

Taxon common name

text

9. Data description

VARCHAR2(255 BYTE)

The common name of the marine organism associated with the 'scientific_name' and 'species_code' columns. For a complete species list, review the code books.

ID_RANK

Lowest taxonomic rank

text

VARCHAR2(255 BYTE)

Lowest taxonomic rank of a given species entry.

ITIS

ITIS taxonomic serial number

ID code

NUMBER(38,0)

Species code as identified in the Integrated Taxonomic Information System (<https://itis.gov/>).

SCIENTIFIC_NAME

Taxon scientific name

text

VARCHAR2(255 BYTE)

The scientific name of the organism associated with the 'common_name' and 'species_code' columns. For a complete taxon list, review the code books.

SPECIES_CODE

Taxon code

ID code

NUMBER(38,0)

The species code of the organism associated with the 'common_name' and 'scientific_name' columns. For a complete species list, review the code books.

WORMS

World Register of Marine Species Taxonomic Serial Number

ID code

9. Data description

NUMBER(38,0)

Species code as identified in the World Register of Marine Species (WoRMS) (<https://www.marinespecies.org/>).

9.1.5. FOSS_SURVEY_SPECIES

This reference dataset contains the full list of species by survey to be used to zero-fill FOSS_CATCH and FOSS_HAUL for each survey. These tables were created by the Resource Assessment and Conservation Engineering Division (RACE) Groundfish Assessment Program (GAP) of the Alaska Fisheries Science Center (AFSC). There are legal restrictions on access to the data. These data are not intended for public dissemination and should not be shared without the explicit written consent of the data managers and owners (NOAA Fisheries). The GitHub repository for the scripts that created this code can be found at https://github.com/afsc-gap-products/gap_products. For more information about codes used in the tables, please refer to the survey code books (<https://www.fisheries.noaa.gov/resource/document/groundfish-survey-species-code-manual-and-data-codes-manual>). These data were last updated October 17, 2023.

Number of rows: 5,025

Number of columns: 2

Column name from data

Descriptive column Name

Units

Oracle data type

Column description

SPECIES_CODE

Taxon code

ID code

NUMBER(38,0)

The species code of the organism associated with the 'common_name' and 'scientific_name' columns. For a complete species list, review the code books.

SURVEY_DEFINITION_ID

9. Data description

Survey ID

ID code

NUMBER(38,0)

This number uniquely identifies a survey. Name and description of survey. The column 'survey_id' is associated with the 'srvy' and 'survey' columns. For a complete list of surveys, review the code books.

9.1.6. FOSS_TAXON_GROUP

This reference dataset contains suggested search groups for simplifying species selection in the FOSS data platform so users can better search through FOSS-CATCH. These tables were created by the Resource Assessment and Conservation Engineering Division (RACE) Groundfish Assessment Program (GAP) of the Alaska Fisheries Science Center (AFSC). There are legal restrictions on access to the data. These data are not intended for public dissemination and should not be shared without the explicit written consent of the data managers and owners (NOAA Fisheries). The GitHub repository for the scripts that created this code can be found at https://github.com/afsc-gap-products/gap_products. For more information about codes used in the tables, please refer to the survey code books (<https://www.fisheries.noaa.gov/resource/document/groundfish-survey-species-code-manual-and-data-codes-manual>). These data were last updated October 17, 2023.

Number of rows: 37,606

Number of columns: 3

Column name from data

Descriptive column Name

Units

Oracle data type

Column description

CLASSIFICATION

Taxonomic classification rank group

category

VARCHAR2(255 BYTE)

9. Data description

Phylogenetic classification group rank for a given species.

RANK_ID

Taxonomic rank

category

VARCHAR2(255 BYTE)

The taxonomic rank of a taxon identification.

SPECIES_CODE

Taxon code

ID code

NUMBER(38,0)

The species code of the organism associated with the 'common_name' and 'scientific_name' columns. For a complete species list, review the code books.

10. Using the FOSS platform

10.1. Select and filter

Select, filter, and package this and other NOAA Fisheries data from the Fisheries One Stop Shop (FOSS) platform. A user guide for the FOSS platform can be found here. To begin a report, select options from the boxes what you need data for.

For a given box, select one or a few options from the “options box” (list on the left) to query by highlighting them. To select multiple options, hold down the CTRL key while clicking on the options of interest, or click and drag down the list. Once the options you wish to be included in your query are highlighted, click the right-pointing arrow (>) to move them into the “selection box” (list on the right). If you accidentally select an option that you do not want to query, simply select the unwanted option from the selection box and click the left-pointing arrow (<).

If you wish to select all options from the options box and send them to the selection box, simply click the double right-pointing arrow (>>). If you want to unselect all options from the selection box, use the double left-pointing arrow (<<) or the reset icon.

To find a specific species or group more quickly you can use the Search Species option to quickly narrow the options. Search for parts of species common names in the Search Species box by entering a term and clicking the search button. The platform will return a shorter list in the Species options box of only species that contain a match to that search term.

Use the Reset All Parameters button to reset all parameters for entire form.

Filter options:

- Survey: Each survey has different in design, time series, and history. More information on each survey and their designs can be found in our annual data reports.
- Year: Surveys are not conducted in all years, so only data from the years for which the survey was conducted will be returned.

10. Using the FOSS platform

The screenshot shows the AFSC Groundfish and Crab Assessment Program Bottom Trawl Surveys data interface. At the top, the NOAA Fisheries logo is visible. The main title "AFSC Groundfish and Crab Assessment Program" and "Bottom Trawl Surveys" is displayed above a banner image of a fish swimming in water. Below the title, a message states "Survey data also available through API". On the left, a sidebar menu includes links to FUS Report, Landings, Foreign Trade, Top US Ports, Processed Products, Per Capita Consumption, Supply, USCG Vessels, AFSC GAP Survey (which is selected), AFSC GAP Metadata, Partners, Metadata and Caveats, Frequently Asked Questions, Quick Start Guide, and Comments. The main content area has a blue header "Data Caveats" which contains a detailed description of the survey's purpose and limitations. Below this is a "Parameters" section with three dropdown menus: "Survey" (listing Aleutian Islands Bottom Trawl Survey, Eastern Bering Sea Slope Bottom Trawl Survey, Eastern Bering Sea Crab/Groundfish Bottom Trawl Survey, Gulf of Alaska Bottom Trawl Survey, Northern Bering Sea Crab/Groundfish Survey - Eastern Bering Sea Shelf S), "Year" (listing years from 2014 to 2022), and "Species" (listing Abalone Jingle, Abietinaria, Abyssal Crangon, Acanthascus, Acantholithodes, Achanax johnsoni, Acoel Turbellarian, Acteocina). There is also a "Search Species" input field and "Search" and "Reset All Parameters" buttons. At the bottom, there is a "Data Format" section with radio buttons for "All Data Fields", "Catch Data: All Units", "Catch Data: Hectares", "Catch Data: Square Thousand km", "Catch Data: Square km", and "Haul Data". A large "RUN REPORT" button is at the very bottom.

Figure 10.1.: AFSC Groundfish and Crab Assessment Program Bottom Trawl Survey data interface on the Fisheries One Stop Shop platform.

10. Using the FOSS platform

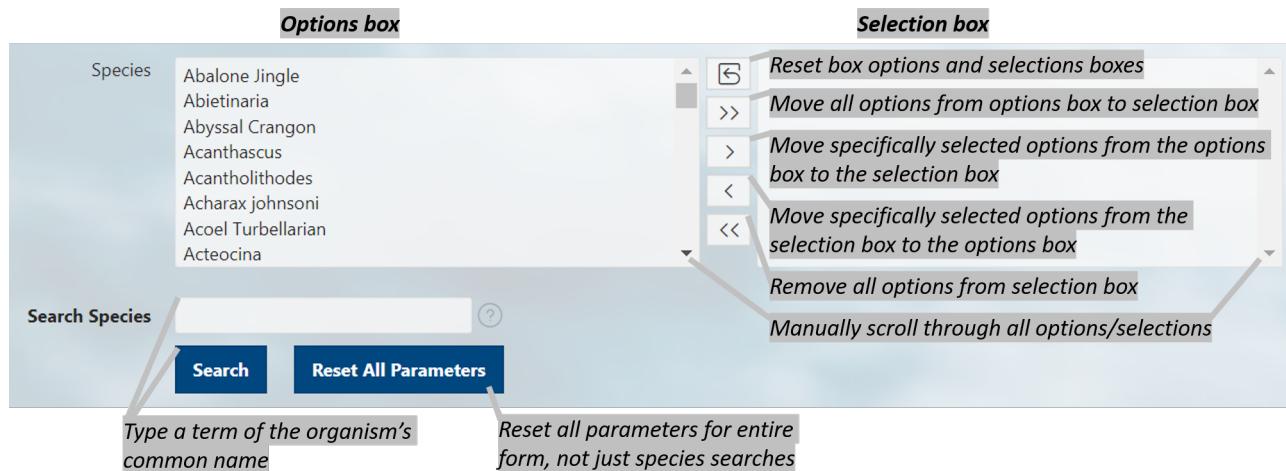


Figure 10.2.: Diagram of selection and search tools available on the FOSS platofrom.

- Species: Common name of all species ever encountered in the survey. Find more information about these species in our survey code books.

In this example, we'll select for 2022 eastern Bering Sea Pacific cod data. Here, we used the Search Species box to search for species with the term "cod" in their common names and selected "Pacific cod" from that shortened list.

10.2. Select data format

Select from the below radio list of pre-designed output tables. Once you run the report, the user can further specify filter data and select columns of interest. The tables below will only include data from the selections made in the previous step.

- All Data Fields: Presence and Absence (zero-filled): The most complete version of the data, including species, catch, haul, and environmental data. This data will include catch data for where species were caught and zeros for where the species were not caught. This is important for calculating catch-per-unit-effort data, preparing distribution plots (e.g., using the akgfmaps R package), and many statistical analyses.
- All Data Fields: Presence-only (non-zero): The second most complete version of the data, including species, catch, haul, and environmental data. However, this data only includes catch data for where species were caught and does

10. Using the FOSS platform

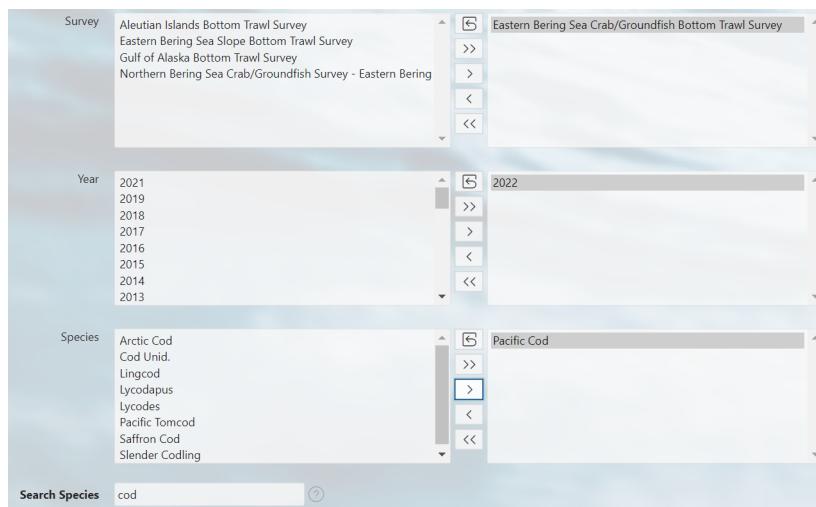


Figure 10.3.: Diagram of selection and search tools available on the FOSS platoform.

not include zeros for where the species were not caught. This will return smaller, more focused data and can be useful for quickly assessing how many species were caught or how many stations species were caught at.

- Catch data: Presence and Absence (zero-filled): This data set is similar to All Data Fields: Presence and Absence (zero-filled), but only includes catch and species data columns.
- Catch data: Presence-only (non-zero): This data set is similar to All Data Fields: Presence-only (non-zero), but only includes catch and species data columns.
- Haul Data: This data set only includes haul and environmental data collected from the survey. This data will only include one observation per haul event/station.

In this example, we'll select All Data Fields: Presence and Absence (zero-filled).

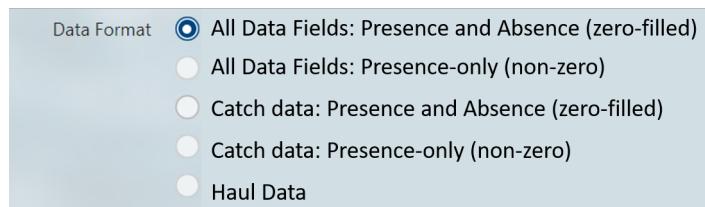
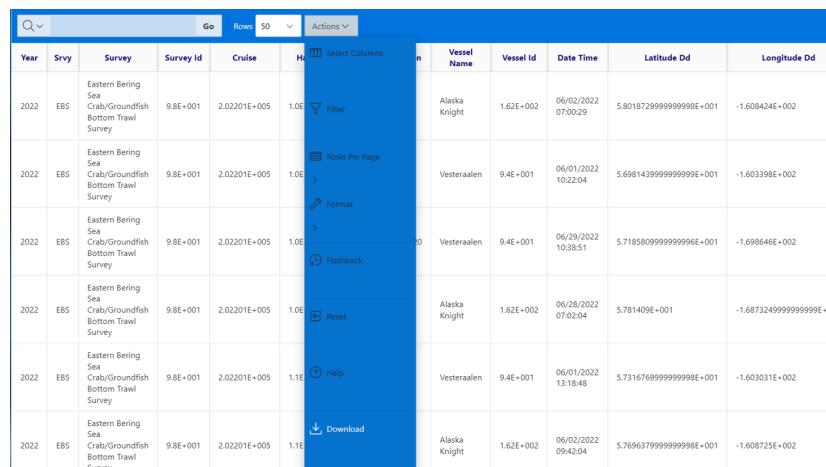


Figure 10.4.: Diagram of the pre-set data format options.

10. Using the FOSS platform

10.3. Run report

Click the RUN REPORT button. Below the select and filter area, the results of your query will appear below the page in the format you selected. To change the format, make a different selection and run the report again. Further modifications to your results can be made by clicking on the Actions button above your data. Here you can download your data, select columns included in your results, and apply a variety of filters and mathematical tools.



The screenshot shows a report interface with a table of survey data. A context menu is open over the last row of the table, listing options: Select Columns, Filter, Rows Per Page, Format, Flashback, Reset, Help, and Download. The table has columns for Year, Srvy, Survey, Survey Id, Cruise, and H. The data rows represent surveys from 2022, specifically Eastern Bering Sea Crab/Groundfish Bottom Trawl Surveys, with various vessel names and coordinates listed.

Year	Srvy	Survey	Survey Id	Cruise	H	
2022	EBS	Eastern Bering Sea Crab/Groundfish Bottom Trawl Survey	9.8E+001	2.02201E+005	1.0E	Filter
2022	EBS	Eastern Bering Sea Crab/Groundfish Bottom Trawl Survey	9.8E+001	2.02201E+005	1.0E	Rows Per Page
2022	EBS	Eastern Bering Sea Crab/Groundfish Bottom Trawl Survey	9.8E+001	2.02201E+005	1.0E	Format
2022	EBS	Eastern Bering Sea Crab/Groundfish Bottom Trawl Survey	9.8E+001	2.02201E+005	1.0E	Flashback
2022	EBS	Eastern Bering Sea Crab/Groundfish Bottom Trawl Survey	9.8E+001	2.02201E+005	1.1E	Reset
2022	EBS	Eastern Bering Sea Crab/Groundfish Bottom Trawl Survey	9.8E+001	2.02201E+005	1.1E	Help
2022	EBS	Eastern Bering Sea Crab/Groundfish Bottom Trawl Survey	9.8E+001	2.02201E+005	1.1E	Download

Figure 10.5.: Example data returned from running the report.

11. Access via API and R

An application programming interface (API) is a way for two or more computer programs to communicate with each other.

More information about how to amend API links can be found [here](#). Useful introductions to using APIs in R can be found [here](#).

11.1. Ex. 1: Load the first 25 rows (default) of data

```
# install.packages(c("httr", "jsonlite"))
library(httr)
library(jsonlite)
library(dplyr)

# link to the API
api_link <- "https://apps-st.fisheries.noaa.gov/ods/foss/afsc_groundfish_survey/"

res <- httr::GET(url = api_link)
# res # Test connection
data <- jsonlite::fromJSON(base::rawToChar(res$content))
# names(data)
tibble::as_tibble(data$items) %>%
  dplyr::mutate_if(is.character, type.convert, as.is = TRUE) %>%
  dplyr::mutate(across(where(is.numeric), round, 3)) %>%
  head(3) %>%
flextable::flextable() %>%
  flextable::theme_zebra() %>%
  flextable::colformat_num(x = ., j = c("year", "cruise", "species_code", "tsn", "ak_surv
```

11. Access via API and R

yearsrvy	survey	survey_id	cruise	haul	stratumstation	vessel_name	vessel
2002AI	Aleutian Islands Bottom Trawl Survey	52	200201	6	722307-63	Vesteraale	
2002AI	Aleutian Islands Bottom Trawl Survey	52	200201	6	722307-63	Vesteraalen	
2002AI	Aleutian Islands Bottom Trawl Survey	52	200201	6	722307-63	Vesteraale	

11.2. Ex. 2: Load the first 10000 rows of data

```
# Not run because too big:  
res <- httr::GET(url = paste0(api_link, "?offset=0&limit=10000"))  
data <- jsonlite::fromJSON(base::rawToChar(res$content))  
print(paste0("rows: ", dim(data$items)[1], "; cols: ", dim(data$items)[2]))
```

```
[1] "rows: 10000; cols: 36"
```

11.3. Ex. 3: Filter by Year

Show all the data greater than the year 2020.

11. Access via API and R

```

res <- httr::GET(url = paste0(api_link, '?q={"year":{$gt":2020}}'))
data <- jsonlite::fromJSON(base::rawToChar(res$content))

as_tibble(data$items) %>%
  mutate_if(is.character, type.convert, as.is = TRUE) %>%
  head(3) %>%
  dplyr::mutate(across(where(is.numeric), round, 3)) %>%
  dplyr::select(year, srvy, stratum, species_code, cpue_kgkm2) %>%
  flextable::flextable() %>%
  flextable::theme_zebra() %>%
  flextable::colformat_num(x = ., j = c("year", "species_code"), big.mark = "")

```

Table 11.2.: Ex. 3: Filter by Year.

year	srvy	stratum	species_code	cpue-kgkm2
2022AI		793	80540	0.361
2022AI		793	401	0.903
2022AI		793	20006	1.661

11.4. Ex. 4: Filter by species name

Show all the data where the product name contains pollock Please note that here the word pollock is case sensitive.

The notation for finding a string is to use % around it. Since % is a reserved character in a URL, you have to replace % with %25.

```

res <- httr::GET(
  url = paste0(api_link, '?q={"common_name":{"$like":"%25pollock%25"}}')
data <- jsonlite::fromJSON(base::rawToChar(res$content))

as_tibble(data$items) %>%
  mutate_if(is.character, type.convert, as.is = TRUE) %>%
  head(3) %>%
  dplyr::mutate(across(where(is.numeric), round, 3)) %>%

```

11. Access via API and R

```
dplyr::select(year, srvy, stratum, species_code, cpue_kgkm2) %>%
flextable::flextable() %>%
  flextable::theme_zebra() %>%
  flextable::colformat_num(x = ., j = c("year", "species_code"), big.mark = "")
```

Table 11.3.: Ex. 4: Filter by species name.

yearsrvy	stratum	species_code	cpue_kgkm2
2002AI	722	21740	775.322
2002AI	722	21740	10,685.806
2002AI	721	21740	0.640

11.5. Ex. 5: Combination of year and name filters

Show all the data where years > 2020 and the product name contains pollock

```
res <- httr::GET(
  url = paste0(api_link,
               '?q={"year":{"$gt":2020}, "common_name":{"$like": "%pollock%"} }'))
data <- jsonlite::fromJSON(base::rawToChar(res$content))

as_tibble(data$items) %>%
  mutate_if(is.character, type.convert, as.is = TRUE) %>%
  head(3) %>%
  dplyr::mutate(across(where(is.numeric), round, 3)) %>%
  dplyr::select(year, srvy, stratum, species_code, cpue_kgkm2) %>%
  flextable::flextable() %>%
    flextable::theme_zebra() %>%
    flextable::colformat_num(x = ., j = c("year", "species_code"), big.mark = "")
```

Table 11.4.: Ex. 5: Combination of year and name filters.

yearsrvy	stratum	species_- code	cpue_- kgkm2
2022AI	793	21740	7,853.632
2022AI	721	21740	7,235.010
2022AI	722	21740	22,754.334

11.6. Ex. 6: Combination of year, srvy, stratum

Show all the data where year = 1989, srvy = "EBS", and stratum is not equal to 81

```
res <- httr::GET(
  url = paste0(api_link, '?q={"year":1989,"srvy":"EBS","stratum":{"$ne":"81"}})')
data <- jsonlite::fromJSON(base::rawToChar(res$content))

as_tibble(data$items) %>%
  mutate_if(is.character, type.convert, as.is = TRUE) %>%
  head(3) %>%
  dplyr::mutate(across(where(is.numeric), round, 3)) %>%
  dplyr::select(year, srvy, stratum, species_code, cpue_kgkm2) %>%
  flextable::flextable() %>%
  flextable::theme_zebra() %>%
  flextable::colformat_num(x = ., j = c("year", "species_code"), big.mark = "")
```

Table 11.5.: Ex. 6: Combination of year, srvy, stratum.

yearsrvy	stratum	species_- code	cpue_- kgkm2
1989EBS	10	66548	1.164
1989EBS	10	69322	1.164
1989EBS	10	43000	2.353

11.7. Ex. 7: Visualize CPUE data in distribution map

Pacific cod catch-per-unit-effort estimates for NBS in 2021 and map constructed using akfmaps.

```
# res <- httr::GET(
#   url = paste0(api_link, "?offset=0&limit=10000"),
#   query = list(year = 2021, srvy = "EBS", species_code = 30060))
res <- httr::GET(
  url = paste0(api_link, '?q={"year":2021,"srvy":"NBS","species_code":21720}'))
data_catch <- jsonlite::fromJSON(base::rawToChar(res$content))$items %>%
  dplyr::select(stratum, station, cpue_kgkm2)

# zero-fill data (imperfectly, but effective for this example)
res <- httr::GET(
  url = paste0(api_link, '?q={"year":2021,"srvy":"NBS"}offset=0&limit=10000'))
data_haul <- jsonlite::fromJSON(base::rawToChar(res$content))$items %>%
  dplyr::select(stratum, station, latitude_dd, longitude_dd) %>%
  dplyr::mutate(across(where(is.numeric), round, 3)) %>%
  dplyr::distinct()

data <- dplyr::left_join(data_haul, data_catch) %>%
  dplyr::mutate(cpue_kgkm2 = ifelse(is.na(cpue_kgkm2), 0, cpue_kgkm2),
    dplyr::across(dplyr::everything(), as.numeric))

flextable::flextable(data[1:3,]) %>%
  flextable::theme_zebra()
```

Table 11.6.: Ex. 7: Visualize CPUE data in distribution map.

stratum	station	latitude_dd	longitude_dd	cpue_kgkm2
81		61.66434	-172.2655	2,895.258
81		62.33740	-173.1702	1,235.545
70		62.03713	-171.6528	0.000

11. Access via API and R

```
# devtools::install_github("afsc-gap-products/akgfmmaps", build_vignettes = TRUE)
library(akgfmmaps)

figure <- akgfmmaps::make_idw_map(
  CPUE_KGHA = data$cpue_kgkm2, # calculates the same, regardless of units.
  LATITUDE = data$latitude_dd,
  LONGITUDE = data$longitude_dd,
  region = "bs.north", # Predefined EBS area
  set.breaks = "jenks", # Gets Jenks breaks from classint::classIntervals()
  in.crs = "+proj=longlat", # Set input coordinate reference system
  out.crs = "EPSG:3338", # Set output coordinate reference system
  grid.cell = c(20000, 20000), # 20x20km grid
  key.title = "Pacific Ocean perch") # Include in the legend title

[inverse distance weighted interpolation]
[inverse distance weighted interpolation]

figure$plot +
  ggplot2::guides(fill=guide_legend(title = "Pacific cod\nCPUE (kg/km2)"))
```

11. Access via API and R

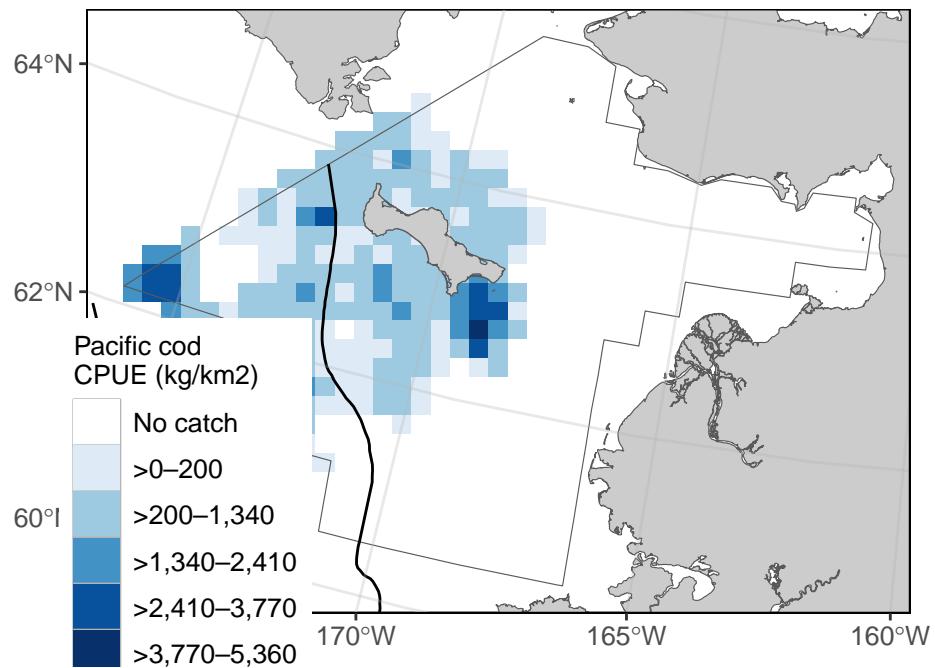


Figure 11.1.: Ex. 7: Visualize CPUE data in distribution map.

12. Access via API and Python

12.0.1. {afscgap} Library Installation

author: Sam Pottinger (sam.pottinger@berkeley.edu; GitHub::sampottinger)
date: May 13, 2023

The third-party afscgap Python package interfaces with FOSS to access AFSC GAP data. It can be installed via pip:

```
#The reticulate package provides a comprehensive set of tools for interoperability between
library(reticulate)
```

```
pip install afscgap
pip install git+https://github.com/SchmidtDSE/afscgap.git@main
```

For more information on installation and deployment, see the library documentation.

12.0.2. Basic query

This first example queries for Pacific glass shrimp (*Pasiphaea pacifica*) in the Gulf of Alaska in 2021. The library will automatically generate HTTP queries, converting from Python types to ORDS query syntax.

```
import afscgap

query = afscgap.Query()
query.filter_year(eq=2021)
query.filter_srvy(eq='GOA')
query.filter_scientific_name(eq='Pasiphaea pacifica')

results = query.execute()
```

12. Access via API and Python

The `results` variable in this example is an iterator that will automatically perform pagination behind the scenes.

12.0.3. Iterating with a for loop

The easiest way to interact with results is a simple for loop. This next example determines the frequency of different catch per unit effort where Pacific glass shrimp were reported:

```
import afscgap

# Mapping from CPUE to count
count_by_cpue = {}

# Build query
query = afscgap.Query()
query.filter_year(eq=2021)
query.filter_srvy(eq='GOA')
query.filter_scientific_name(eq='Pasiphaea pacifica')
results = query.execute()

# Iterate through results and count
for record in results:
    cpue = record.get_cpue_weight(units='kg/ha')
    cpue_rounded = round(cpue)
    count = count_by_cpue.get(cpue_rounded, 0) + 1
    count_by_cpue[cpue_rounded] = count

# Print the result
print(count_by_cpue)
```

Note that, in this example, only records with Pacific glass shrimp are included (“presence-only” data). See zero catch inference below. In other words, it reports on CPUE only for hauls in which Pacific glass shrimp were recorded, excluding some hauls like those in which Pacific glass shrimp were not found at all.

12. Access via API and Python

12.0.4. Iterating with functional programming

A for loop is not the only option for iterating through results. List comprehensions and other functional programming methods can be used as well.

```
import statistics

import afscgap

# Build query
query = afscgap.Query()
query.filter_year(eq=2021)
query.filter_srvy(eq='GOA')
query.filter_scientific_name(eq='Pasiphaea pacifica')
results = query.execute()

# Get temperatures in Celsius
temperatures = [record.get_bottom_temperature(units='c') for record in results]

# Take the median
print(statistics.median(temperatures))
```

This example reports the median temperature in Celcius for when Pacific glass shrimp was reported.

12.0.5. Load into Pandas

The results from the afscgap package are serializable and can be loaded into other tools like Pandas. This example loads Pacific glass shrimp from 2021 Gulf of Alaska into a data frame.

```
import pandas

import afscgap

query = afscgap.Query()
query.filter_year(eq=2021)
query.filter_srvy(eq='GOA')
query.filter_scientific_name(eq='Pasiphaea pacifica')
```

12. Access via API and Python

```
results = query.execute()  
  
pandas.DataFrame(results.to_dicts())
```

Specifically, `to_dicts` provides an iterator over a dictionary form of the data that can be read into tools like Pandas.

12.0.6. Advanced filtering

Queries so far have focused on filters requiring equality but range queries can be built as well.

```
import afscgap  
  
# Build query  
query = afscgap.Query()  
query.filter_year(min_val=2015, max_val=2019)    # Note min/max_val  
query.filter_srvy(eq='GOA')  
query.filter_scientific_name(eq='Pasiphaea pacifica')  
results = query.execute()  
  
# Sum weight  
weights = map(lambda x: x.get_weight(units='kg'), results)  
total_weight = sum(weights)  
print(total_weight)
```

This example queries for Pacific glass shrimp data between 2015 and 2019, summing the total weight caught. Note that most users will likely take advantage of built-in Python to ORDS query generation which dictates how the library communicates with the API service. However, users can provide raw ORDS queries as well using manual filtering.

12.0.7. Zero-catch inference

Until this point, these examples use presence-only data. However, the `afscgap` package can infer negative or “zero catch” records as well.

12. Access via API and Python

```
import afscgap

# Mapping from CPUE to count
count_by_cpue = {}

# Build query
query = afscgap.Query()
query.filter_year(eq=2021)
query.filter_srvy(eq='GOA')
query.filter_scientific_name(eq='Pasiphaea pacifica')
query.set_presence_only(False) # Added to earlier example
results = query.execute()

# Iterate through results and count
for record in results:
    cpue = record.get_cpue_weight(units='kg/ha')
    cpue_rounded = round(cpue)
    count = count_by_cpue.get(cpue_rounded, 0) + 1
    count_by_cpue[cpue_rounded] = count

# Print the result
print(count_by_cpue)
```

This example revisits the earlier snippet for CPUE counts but `set_presence_only(False)` directs the library to look at additional data on hauls, determining which hauls did not have Pacific glass shrimp. This lets the library return records for hauls in which Pacific glass shrimp were not found. This can be seen in differences in counts reported:

Rounded CPUE	Count with <code>set_presence_only(True)</code>	Count with <code>set_presence_only(False)</code>
0 kg/ha	44	521
1 kg/ha	7	7
2 kg/ha	1	1

Put simply, while the earlier example showed CPUE counts for hauls in which Pacific glass shrimp were seen, this revised example reports for all hauls in the Gulf of Alaska in 2021.

12. Access via API and Python

12.0.8. More information

Please see the API documentation for the Python library for additional details.

13. Access via Oracle and R (AFSC only)

If the user has access to the AFSC Oracle database, the user can use SQL developer to view and pull the FOSS public data directly from the GAP_PRODUCTS Oracle schema.

13.0.1. Connect to Oracle from R

Many users will want to access the data from Oracle using R. The user will need to install the RODBC R package and ask OFIS (IT) connect R to Oracle. Then, use the following code in R to establish a connection from R to Oracle:

Here, the user can write in their username and password directly into the RODBC connect function. Never save usernames or passwords in scripts that may be intentionally or unintentionally shared with others. If no username and password is entered in the function, pop-ups will appear on the screen asking for the username and password.

```
library(gapindex)
channel <- gapindex::get_connected()
```

13.0.2. Ex. 1: Join data

To join these tables in Oracle, you may use a variant of the following code:

```
SELECT
hh.YEAR,
hh.SRVY,
hh.SURVEY,
hh.SURVEY_DEFINITION_ID,
hh.SURVEY_NAME,
hh.CRUISE,
hh.CRUISEJOIN,
hh.HAUL,
hh.HAULJOIN,
```

13. Access via Oracle and R (AFSC only)

```
hh.STRATUM,
hh.STATION,
hh.VESSEL_ID,
hh.VESSEL_NAME,
hh.DATE_TIME,
hh.LATITUDE_DD_START,
hh.LONGITUDE_DD_START,
hh.LATITUDE_DD_END,
hh.LONGITUDE_DD_END,
hh.BOTTOM_TEMPERATURE_C,
hh.SURFACE_TEMPERATURE_C,
hh.DEPTH_M,
cc.SPECIES_CODE,
ss.ITIS,
ss.WORMS,
ss.COMMON_NAME,
ss.SCIENTIFIC_NAME,
ss.ID_RANK,
CASE WHEN cc.CPUE_KGKM2 IS NULL THEN 0 ELSE cc.CPUE_KGKM2 END AS CPUE_KGKM2,
CASE WHEN cc.CPUE_NOKM2 IS NULL THEN 0 ELSE cc.CPUE_NOKM2 END AS CPUE_NOKM2,
CASE WHEN cc.COUNT IS NULL THEN 0 ELSE cc.COUNT END AS COUNT,
CASE WHEN cc.WEIGHT_KG IS NULL THEN 0 ELSE cc.WEIGHT_KG END AS WEIGHT_KG,
CASE WHEN cc.TAXON_CONFIDENCE IS NULL THEN NULL ELSE cc.TAXON_CONFIDENCE END AS TAXON_CONFIDENCE,
hh.AREA_SWEEPED_KM2,
hh.DISTANCE_FISHED_KM,
hh.DURATION_HR,
hh.NET_WIDTH_M,
hh.NET_HEIGHT_M,
hh.PERFORMANCE
FROM GAP_PRODUCTS.FOSS_SURVEY_SPECIES sv
FULL OUTER JOIN GAP_PRODUCTS.FOSS_SPECIES ss
ON sv.SPECIES_CODE = ss.SPECIES_CODE
FULL OUTER JOIN GAP_PRODUCTS.FOSS_HAUL hh
ON sv.SURVEY_DEFINITION_ID = hh.SURVEY_DEFINITION_ID
FULL OUTER JOIN GAP_PRODUCTS.FOSS_CATCH cc
ON sv.SPECIES_CODE = cc.SPECIES_CODE
AND hh.HAULJOIN = cc.HAULJOIN
```

13. Access via Oracle and R (AFSC only)

13.0.3. Ex. 2: Subset data

Here, we are pulling EBS Pacific cod from 2010 - 2021:

```
# Pull data
a <- RODBC::sqlQuery(
  channel = channel,
  query =
  "SELECT * FROM GAP_PRODUCTS.FOSS_CATCH cc
  JOIN GAP_PRODUCTS.FOSS_HAUL hh
  ON cc.HAULJOIN = hh.HAULJOIN
  WHERE SRVY = 'EBS'
  AND SPECIES_CODE = 21720 -- 'Pacific cod'
  AND YEAR >= 2010
  AND YEAR < 2021")

head(a)
```

	HAULJOIN	SPECIES_CODE	CPUE_KGKM2	CPUE_NOKM2	COUNT	WEIGHT_KG	TAXON_CONFIDENCE
1	-19252	21720	413.4828	248.0897	12	20.00	1
2	-19288	21720	449.8301	1876.1759	83	19.90	1
3	-17715	21720	491.3252	218.1905	11	24.77	1
4	-17850	21720	990.1357	152.2616	7	45.52	1
5	-18731	21720	946.3481	2592.1327	118	43.08	1
6	-18165	21720	1053.2723	241.8536	12	52.26	1
	YEAR	SRVY	SURVEY	SURVEY_DEFINITION_ID			
1	2019	EBS	eastern Bering Sea		98		
2	2019	EBS	eastern Bering Sea		98		
3	2018	EBS	eastern Bering Sea		98		
4	2018	EBS	eastern Bering Sea		98		
5	2019	EBS	eastern Bering Sea		98		
6	2018	EBS	eastern Bering Sea		98		
					SURVEY_NAME	CRUISE	CRUISEJOIN
1	Eastern Bering Sea Crab/Groundfish	Bottom Trawl Survey	201901			-727	
2	Eastern Bering Sea Crab/Groundfish	Bottom Trawl Survey	201901			-727	
3	Eastern Bering Sea Crab/Groundfish	Bottom Trawl Survey	201801			-723	
4	Eastern Bering Sea Crab/Groundfish	Bottom Trawl Survey	201801			-723	
5	Eastern Bering Sea Crab/Groundfish	Bottom Trawl Survey	201901			-726	
6	Eastern Bering Sea Crab/Groundfish	Bottom Trawl Survey	201801			-723	
	HAULJOIN.1	HAUL	STRATUM	STATION	VESSEL_ID	VESSEL_NAME	DATE_TIME

13. Access via Oracle and R (AFSC only)

1	-19252	76	31	G-03	162	ALASKA	KNIGHT	2019-06-24	15:52:02
2	-19288	96	20	O-18	162	ALASKA	KNIGHT	2019-06-29	06:54:00
3	-17715	92	20	O-18	162	ALASKA	KNIGHT	2018-06-26	17:42:00
4	-17850	117	32	F-19	162	ALASKA	KNIGHT	2018-07-02	09:49:43
5	-18731	11	31	I-13	94	VESTERAALEN		2019-06-04	13:15:57
6	-18165	172	62	Q-27	162	ALASKA	KNIGHT	2018-07-26	07:49:26
	LATITUDE_DD_START	LONGITUDE_DD_START	LATITUDE_DD_END	LONGITUDE_DD_END					
1	57.01591		56.99137						
2	59.68079		59.65546						
3	59.66752		59.67396						
4	56.67170		56.67359						
5	57.69052		57.66518						
6	60.31173		60.33716						
	BOTTOM_TEMPERATURE_C	SURFACE_TEMPERATURE_C	DEPTH_M	DISTANCE_FISHED_KM					
1	4.1		9.7	74					
2	5.1		7.8	39					
3	5.9		5.9	40					
4	4.5		8.8	99					
5	5.5		7.5	54					
6	3.2		10.1	103					
	DURATION_HR	NET_WIDTH_M	NET_HEIGHT_M	AREA_SWEPT_KM2	PERFORMANCE				
1	0.528	16.795	2.126	0.048370	0				
2	0.505	15.682	2.227	0.044239	0				
3	0.517	16.912	1.736	0.050415	0				
4	0.525	15.299	2.152	0.045973	0				
5	0.520	16.029	2.200	0.045522	0				
6	0.511	17.440	2.200	0.049617	0				

13.0.4. Ex. 3: Find all species found in the eastern Bering Sea (EBS) survey in 2023

```
# Pull data
a <- RODBC::sqlQuery(
  channel = channel,
  query =
  "SELECT DISTINCT
    ss.COMMON_NAME,
    ss.SCIENTIFIC_NAME,
    ss.ID_RANK,
```

13. Access via Oracle and R (AFSC only)

```
ss.WORMS
FROM GAP_PRODUCTS.FOSS_CATCH cc -- get species codes
LEFT JOIN GAP_PRODUCTS.FOSS_SPECIES ss -- get species info
ON cc.SPECIES_CODE = ss.SPECIES_CODE
LEFT JOIN GAP_PRODUCTS.FOSS_HAUL hh -- filter by year and survey
ON cc.HAULJOIN = hh.HAULJOIN
WHERE hh.YEAR = 2023
AND hh.SURVEY_DEFINITION_ID = 98 -- EBS survey
ORDER BY COMMON_NAME")
```

```
head(a)
```

	COMMON_NAME	SCIENTIFIC_NAME	ID_RANK	WORMS
1	Alaska falsejingle	Pododesmus macrochisma	species	758426
2	Alaska great-tellin	Megangulus luteus	species	423511
3	Alaska plaice	Pleuronectes quadrituberculatus	species	254564
4	Alaska razor	Siliqua alta	species	413689
5	Alaska skate	Arctoraja parmifera	species	1577324
6	Alaska skate egg case	Arctoraja parmifera	egg case	NA

Part VI.

Product Outputs

To accompany these data, we also produce data products to make using our data more accessible and straightforward.

Table 13.1.: Survey of products developed by GAP

Product	Point of Contact GOA/AI	Point of Contact BS	Description
Data			
Finalized bottom trawl data	Ned Laman	Duane Stevenson	NOAA-NMFS-AFSC-RACE-GAP bottom trawl survey data that has completed the post-survey internal QAQC process.
Data requests	Nancy Roberson	Nancy Roberson	To request a subset of the NOAA-NMFS-AFSC-RACE-GAP bottom trawl survey raw data or a data product.
Species codebook	Nancy Roberson	Chris Anderson	List of codes used for fish and invertebrates identified in NOAA-NMFS-AFSC-RACE-GAP Division surveys.
Survey protocols	Nancy Roberson	Nancy Roberson	Documentation of NOAA-NMFS-AFSC-RACE-GAP group bottom trawl survey protocols.
Analysis			
Design-based indices for target species	Ned Laman	Rebecca Haehn	Standard design-based indices of benthic abundance from NOAA-NMFS-AFSC-RACE-GAP bottom trawl survey data.
Design-based age or length composition	Ned Laman	Rebecca Haehn	Standard design-based indices of species composition from NOAA-NMFS-AFSC-RACE-GAP bottom trawl survey data.
Model-based indices, age comps (stock assessment), area occupied, and COG (ESP)	Cecilia O'Leary	Lewis Barnett	Spatiotemporal model-based biomass abundance indices, and age composition from NOAA-NMFS-AFSC-RACE-GAP bottom trawl survey data.
Annual bottom and surface temperature summary (ESR, stock assessment)	Cecilia O'Leary	Sean Rohan & Lewis Barnett	Summary metrics for bottom trawl and surface temperatures relative to baseline.

Product	Point of Contact GOA/AI	Point of Contact BS	Description
Salinity indicator (ESR in 2024)	-	Sean Rohan & Ned Cokelet	Developing salinity indicators (available 2024).
Bering Sea cold pool index and temperature data products (ESR, ESP, stock assessment)	-	Sean Rohan & Lewis Barnett	Create annual temperature rasters for EBS, calculate the EBS cold pool index, temperature data products, and provide visualizations.
Annual fish condition (ESR)	Cecilia O'Leary	Bianca Prohaska & Sean Rohan	Groundfish morphometric condition in the Bering Sea, Aleutian Islands, and Alaska.
Rockfish indices vs environmental gradients (ESR)	Alexandra Dowlin	-	GOA/AI survey trends in distribution and abundance of 6 rockfishes across environmental gradients in the North Pacific.
Structure-Forming Invertebrates-Habitat Areas of Particular Concern (SFI-HAPC) (ESR)	Ned Laman	-	Relative abundance of sponges, hydroids, soft corals, Gorgonians, anemones, and Pennatulaceans in GOA and AI surveys.
Forage fishes (ESR)	Ned Laman	-	Relative abundance of capelin, eulachnus, sandfish, sand lance, and prickelback in GOA and AI surveys.
Miscellaneous species (ESR)	Ned Laman	Thaddeus Buser	Relative abundance of echinoderms, shrimp and eelpouts in GOA and AI surveys.
Jellies (ESR)	Ned Laman	Thaddeus Buser	Relative abundance of sea jellies in GOA surveys.
Essential fish habitat	Megsie Siple	Sean Rohan	Habitat maps for groundfish and commercial species based on species distribution models. Updated every five years.
<i>Visualization Tools</i>			
Alaska groundfish maps (CPUE, etc.)	-	Sean Rohan	Visualization tool for the Alaska survey regions.
<i>Communication</i>			

Product	Point of Contact GOA/AI	Point of Contact BS	Description
Annual survey data report	Megsie Siple & Bethany Riggle	Emily Markowitz, Liz Dawson & Chris Anderson	Alaska Fisheries Science Center NCE Technical Memorandum summary survey progress and findings. These available online and the latest published each survey are listed below (https://repository.library.noaa.gov)
ADF&G report of research activities	Alexandra Dowlin	Nicole Charriere & Rebecca Haehn	Report on AI and GOA trawl survey activity inside and outside of Alaska waters.
IPHC Report	-	Rebecca Haehn	
Plan team survey results presentation	Ned Laman	Duane Stevenson	NOAA-NMFS-AFSC-RACE-GAP presentations findings to the North Pacific Groundfish Team; presentations, recordings, and attachments located here: https://www.npfmc.org/about-the-council/plan-teams/bsai-and-goa-group
Community highlights report	TBD	Emily Markowitz	Compilation of NOAA-NMFS-AFSC-BSAI survey findings for communities around Alaska.
Bottom Trawl Survey Temperature and Progress Maps	Ned Laman	Emily Markowitz, Liz Dawson & Chris Anderson	Near real-time survey progress and temperatures recorded during the Aleutian Islands, Gulf of Alaska, and Bering Sea Trawl Surveys.

14. Open source code

14.1. R Packages

14.1.1. akgfmaps R package

Bottom trawl survey maps layers and plotting examples. **POC:** Sean Rohan

14.1.2. coldpool R package

Cold pool area and temperature data products for the Bering Sea. **POC:** Sean Rohan

14.1.3. akfishcondition R package

Groundfish morphometric condition indicators for fish in the Bering Sea, Aleutian Islands, and Gulf of Alaska. **POC:** Sean Rohan

14.1.4. gapindex R package

Calculation of Design-Based Indices of Abundance and Composition for AFSC GAP Bottom Trawl Surveys. **POC:** Zack Oyafuso and Margaret Siple

Part VII.

Contact us

This code is primarally maintained by:

Thank you for using our data guide!

This code is always in development. Find code used for various reports in the code releases.

This code is primarally maintained by:

Emily Markowitz (Emily.Markowitz AT noaa.gov; @EmilyMarkowitz-NOAA)

Zack Oyafuso (Zack.Oyafuso AT noaa.gov; @zoyafuso-NOAA)

Sarah Friedman (Sarah.Friedman AT noaa.gov; @SarahFriedman-NOAA)

Alaska Fisheries Science Center,

National Marine Fisheries Service,

National Oceanic and Atmospheric Administration,

Seattle, WA 98195

General questions and more specific data requests can be sent to afsc.gap. metadata@noaa.gov or submitted as an issue on our GitHub Organization. The version of this data used for stock assessments can be found through the Alaska Fisheries Information Network (AKFIN). For questions about the eastern Bering Sea surveys, contact Duane Stevenson (Duane.Stevenson@noaa.gov). For questions about the Gulf of Alaska or Aleutian Islands surveys, contact Ned Laman (Ned.Laman@noaa.gov). For questions specifically about crab data in any region, contact Mike Litzow (Mike.Litzow@noaa.gov), the Shellfish Assessment Program lead.

For questions, comments, and concerns specifically about the Fisheries One Stop Shop (FOSS) platform, please contact us using the Comments page on the FOSS web-page.

15. Production run notes

16. R Version Metadata

```
R version 4.3.1 (2023-06-16 ucrt)
Platform: x86_64-w64-mingw32/x64 (64-bit)
Running under: Windows 10 x64 (build 19045)
```

```
Matrix products: default
```

```
locale:
[1] LC_COLLATE=English_United States.utf8
[2] LC_CTYPE=English_United States.utf8
[3] LC_MONETARY=English_United States.utf8
[4] LC_NUMERIC=C
[5] LC_TIME=English_United States.utf8

time zone: America/Los_Angeles
tzcode source: internal

attached base packages:
[1] stats      graphics   grDevices utils      datasets  methods    base

loaded via a namespace (and not attached):
[1] compiler_4.3.1    fastmap_1.1.1    cli_3.6.1        tools_4.3.1
[5] htmltools_0.5.6   rstudioapi_0.15.0  yaml_2.3.7      rmarkdown_2.25
[9] knitr_1.44       jsonlite_1.8.7    xfun_0.40      digest_0.6.33
[13] rlang_1.1.1      evaluate_0.22
```

16.0.1. 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

16. R Version Metadata

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.

16.0.2. 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.

17. Acknowledgments

18. Community Acknowledgments

We would like to thank the many communities of Alaska and their members who have helped contribute to this body of work. The knowledge, experiences, and insights have been instrumental in expanding the scope of our science and knowledge to encompass the many issues that face this important ecosystem. We appreciate feedback from those residing in the region that are willing to share their insights and participation in an open dialog about how we can improve our collective knowledge of the ecosystem and the region.

19. Land Acknowledgements

We would like to thank the many communities of the Bering Strait region and their members who have helped contribute to this document. The knowledge, experiences, and insights of the people of the Bering Strait region have been instrumental in expanding the scope of our science and knowledge to encompass the many issues that face this important ecosystem. We appreciate feedback from those residing in the region that are willing to share their insights, including the local names used for the species covered by this document, identifying species of interest or concern that should be included in this document, and participation in an open dialog about how we can improve our collective knowledge of the ecosystem and the region.

NOAA Fisheries Alaska Fisheries Science Center's work is conducted in the waters and along the coastlines of Alaska, which include the traditional home lands and waters of the Inupiat, Yupiit, Siberian Yupiit, Unangax, Alutiiq/Sugpiaq, Eyak, Dena'ina Athabascan, Tlingit, Haida, and Tsimshian who have stewarded their lands and waters since time immemorial. We are indebted to these peoples for their wisdom and knowledge of their lands and waters.

This document was prepared in the greater Seattle area, which are the traditional lands of the Coast Salish people, including the Duwamish people, past and present. We are grateful for their continued sharing of vision, wisdom, values, and leadership.

20. Technical Acknowledgments

This quarto book is based off the NOAA-quarto-book GitHub repo designed by Eli Holmes.

This repo and GitHub Action was based on the tutorial by Openscapes quarto-website-tutorial by Julia Lowndes and Stefanie Butland.

20.1. Partners

Scientists from the Alaska Fisheries Science Center conduct these bottom trawl surveys with participation from the Alaska Department of Fish & Game (ADF&G), the International Pacific Halibut Commission (IPHC), and universities. This research is conducted on chartered fishing vessels.

20.2. Collaborators

Our data are used in many annual publications, including but not limited to the list below:

- Alaska Stock Assessments
- North Pacific Groundfish Stock Assessment and Fishery Evaluation Reports
- Groundfish Economic Status Reports for the Gulf of Alaska and Bering Sea and Aleutian Islands
- Alaska Marine Ecosystem Status Report Database
- Southeast Alaska Coastal Monitoring Survey Reports
- Alaska Fisheries Life History Database
- Essential Fish Habitat Research Plan in Alaska

21. References

- Alaska Fisheries Information Network (AKFIN). (2023). *AFSC groundfish assessment program design-based production data*. NOAA Fisheries Alaska Fisheries Science Center, Goundfish Assessment Program; <https://www.psmfc.org/program/alaska-fisheries-information-network-akfin>; U.S. Dep. Commer.
- Hoff, G. R. (2016). *Results of the 2016 eastern Bering Sea upper continental slope survey of groundfishes and invertebrate resources* (NOAA Tech. Memo. NOAA-AFSC-339). U.S. Dep. Commer. <https://doi.org/10.7289/V5/TM-AFSC-339>
- Markowitz, E. H., Dawson, E. J., Anderson, A. B., Rohan, S. K., Charriere, N. E., Prohaska, B. K., and Stevenson, D. E. (2023). *Results of the 2022 eastern and northern Bering Sea continental shelf bottom trawl survey of groundfish and invertebrate fauna* (NOAA Tech. Memo. NMFS-AFSC-469; p. 213). U.S. Dep. Commer.
- NOAA Fisheries Alaska Fisheries Science Center. (2023). *Fisheries one stop shop public data: RACE division bottom trawl survey data query*. <https://www.fisheries.noaa.gov/foss>; U.S. Dep. Commer.
- NOAA Fisheries Alaska Fisheries Science Center, Goundfish Assessment Program. (2023). *AFSC groundfish assessment program design-based production data*. <https://www.fisheries.noaa.gov/alaska/science-data/groundfish-assessment-program-bottom-trawl-surveys>; U.S. Dep. Commer.
- Von Szalay, P. G., and Raring, N. W. (2018). *Data report: 2017 Gulf of Alaska bottom trawl survey* (NOAA Tech. Memo. NMFS-AFSC-374). U.S. Dep. Commer. <https://doi.org/10.7289/V5/TM-AFSC-374>
- Von Szalay, P. G., and Raring, N. W. (2020). *Data report: 2018 Aleutian Islands bottom trawl survey* (NOAA Tech. Memo. NMFS-AFSC-409). U.S. Dep. Commer. <https://doi.org/10.25923/qe5v-fz70>