

Testing {asar} and {stockplotr}

Directions:

We have prepared a workflow for writing a stock assessment report using the `asar` and `stockplotr` R packages. Run code step-by-step, in the [activity](#) section. Open a new script when indicated. A stock synthesis output file is provided in the repository for use during this workflow. The example provided is for the 2024 yelloweye rockfish (*Sebastes rubberimus*) update on the U.S. West Coast. All materials to complete this activity can be found in the [Workflows Workthrough repo](#).

Please consider the following during your work through:

- How would this template work based on your current guidelines?
- How easy was the template to make?
- How similar was this to your current workflow?
- Did you encounter any issues or errors?
- Could you see the future of its use into your workflow?
- What improvements would you make to the process as a whole or individual parts of the workflow?

Activity:

Initialize your workflow

1. Create a repository from the [template repo in GitHub](#).
2. Identify where you want the folder containing your report files as well as the figures and tables files to be placed*. We recommend putting it in a folder containing the rest of your materials for a particular assessment species and year.

```
setwd("~/Documents/GitHub/")
```

**{asar} defaults to your working directory; however, you are able to tell it where to place the files through the `file_dir` argument.*

3. Open a new script that will be the basis for creating the reproducible workflow.
4. Install the most recent version of {asar} and {stockplotr}

```
install.packages(  
  "asar",  
  repos = c("https://nmfs-ost.r-universe.dev",
```

```

    "https://cloud.r-project.org"))
install.packages(
  "stockplotr",
  repos = c("https://nmfs-ost.r-universe.dev",
    "https://cloud.r-project.org"))

```

Alternative methods for installation, can be found on the package's respective GitHub Pages, [asar](#) and [stockplotr](#).

5. If you haven't already, install tinytex expanded version:

```

install.packages("tinytex")
tinytex::install.tinytex(bundle = "TinyTeX")

```

If you want to go above and beyond, download the TinyTeX bundle 2 instead of the above option which might be necessary for later process (WARNING: this will take upwards of ~30-45 minutes).

```

tinytex::install.tinytex(bundle = "TinyTeX-2")

```

Automation

1. Create a blank report template using the following arguments (or a region/species of your choice):
 - a. region = "U.S. West Coast"
 - b. species = "Yelloweye rockfish"
 - c. spp_latin = "Sebastes rubberimus"
 - d. year = 2024
 - e. Author (use your name)
 - f. Office (use your office)

Unset

```

asar::create_template(
  office = "NWFSC",
  region = "U.S. West Coast",
  species = "Yellowtail rockfish",
  spp_latin = "Sebastes ruberrimus",
  year = 2024,
  author = "Your name"
)

```

Render the skeleton file and see what it looks like blank!

2. Convert the report.sso file in the repo to a standard dataframe. Load it into your environment.

Unset

```
output <- asar::convert_output(  
  output_file = "Report.sso",  
  outdir = getwd(),  
  model = "SS3"  
)
```

Note: An example Report.sso file has been provided for you. If you work with BAM or SS3, feel free to use your own model output for this exercise.

3. Create all of the tables and figures from `stockplotr`

Unset

```
stockplotr::exp_all_figs_tables(  
  dat = output,  
  ref_line = "target",  
  ref_line_sb = "target",  
  indices_unit_label = ""  
)
```

4. Add the tables and figures into your outline.

Unset

```
asar::create_tables_doc(  
  subdir = file.path(getwd(), "report"),  
  rda_dir = getwd()  
)  
asar::create_figures_doc(  
  subdir = file.path(getwd(), "report"),  
  rda_dir = getwd()  
)
```

5. Update your preamble so it includes the converted output file.

Unset

```
# Save std. output file
```

```

write.csv(
  output,
  file = file.path(getwd(), "report", "std_output.csv"),
  row.names = FALSE
)
# Add to template
asar::create_template(
  rerender_skeleton = TRUE,
  file_dir = file.path(getwd(), "report"),
  model_results = "std_output.csv",
  resdir = file.path(getwd(), "report")
)

```

Re-render your quarto file to check on your progress! 😊

Alterations to the template

1. Woops! You forgot to add an author to the outline. Please add “Patrick Star” as an additional author into your report.

Unset

```

asar::create_template(
  rerender_skeleton = TRUE,
  file_dir = file.path(getwd(), "report"),
  author = "Patrick Star"
)

```

2. Of course...The SSC requested a new section be added to the report for this species. Please add the section “Social Indicators” to your outline in the data section then add another section, “Management”, to your outline after the assessment sections.

Unset

```

asar::create_template(
  rerender_skeleton = TRUE,
  file_dir = file.path(getwd(), "report"),
  custom = TRUE,

```

```

new_section = c("Social Indicators", "Management"),
section_location = c("in-data", "after-assessment")
)

```

3. You need to generate a plot that is not currently in stockplotr. Add the provided image (SPRratio.png) from the repo into the figures doc.

Hint: make sure to incorporate a caption and alternative text in their proper location.

Hint #2: Check out the [Custom Figures & Tables vignette](#) for more help.

```

Unset
file.copy(
  from = file.path(getwd(), "SPRratio.png"),
  to = file.path(getwd(), "report", "SPRratio.png"),
  overwrite = TRUE
)

image_line <- paste0(
  "![My caption](SPRratio.png){#fig-sprratio}"
)
# Add to figure.qmd
figs_qmd <- readLines(
  file.path(getwd(), "report", "09_figures.qmd"),
  warn = FALSE
)
new_lines <- append(
  figs_qmd,
  image_line
)
writeLines(
  new_lines,
  file.path(getwd(), "report", "09_figures.qmd")
)
# Add image bundle to captions.alt_text.csv
# This can be done manually if preferred
label <- "fig-sprratio"
alt_text <- "My alternative text."
cap <- ""
my_img <- data.frame("label" = label, "type" = "figure", "caption" = cap,
  "alt_text" = alt_text)

```

```
write.table(
  my_img,
  file = file.path(getwd(), "captions_alt_text.csv"),
  sep = ",",
  row.names = FALSE,
  col.names = FALSE,
  append = TRUE
)
```

4. Check the captions and alternative text for your figures and tables. Find the line associated with the biomass figure and remove the sentence in the caption about the limit reference point, since it doesn't appear in the figure.

Hint: Check out the [Accessibility vignette](#) for more help.

- Go to your working directory
- Open the "captions_alt_text.csv" file
- Navigate to the line with label "biomass" and type "figure"
- Delete this text: " The horizontal dashed line represents the limit reference point (B.ref.pt mt)."
- Save the file.

Final Actions

1. Remove draft watermark.

Unset

```
asar::remove_draft()
```

2. Add alternative text and PDF tagging.

Unset

```
# set wd to the "report" folder
setwd("~/folder/report")
# run function
asar::add_accessibility(
  rda_dir = "~/folder",
  alttext_csv_dir = "~/folder",
  rename = "2025_yelloweye_SAR_501compliance"
)
```