

*Response to review from Dr. Jeff Hollister, “SWMPPr: An R Package for Retrieving, Organizing, and Analyzing Environmental Data for Estuaries”, Beck, MW.*

Response to reviewer comments are provided in italics. The original review is included as a pdf with markup comment. The original pages and line numbers are noted below.

Page 1: I describe SWMPPr, an open-source R package, for use ...

*Changed.*

Page 2: growth

*Changed.*

Page 2: is limited

*Changed.*

Page 2: identifying the signal from the

*Changed.*

Page 2: This feels like you are questioning the use of this type of data? Also, try to transition to the next paragraph. Feels a bit abrupt.

*Modified as follows: ‘A greater focus on synthesis, exploratory-based analytical techniques, and interpretation have characterized the use of data from automated monitoring programs [6, 7]. An important monitoring network for coastal environments in the United States is the National Estuarine Research Reserve System (NERRS). This network represents 28 estuarine reserves from different biogeographic regions that were chosen to address multiple goals for long-term research, monitoring, education, and stewardship in support of coastal management.’*

Page 2: both for

*Changed.*

Page 2: Why just abiotic? You should add a bit about this. Are the other data not available, just haven’t done it yet? If the later, let me know, that’d be fun to work on!

*The description in the text is vague in this regard and I have modified for clarity: ‘Only the abiotic data are monitored continuously with automated sensor networks, whereas the remaining elements involve field surveys or mapping products that differ between reserves given site-specific requirements. As such, the SWMPPr package is developed for the continuous abiotic monitoring network that represents a majority of the SWMP data and,*

*consequently, the most challenging to evaluate. '*

Page 3: space

*Added.*

Page 3: On CRAN now, correct? IF so update installation directions.

*Installation instructions were updated*

Page 3: Would bet a lot of readers won't have any idea what this means. Since the `swmpr` object is central to the package, maybe a section describing its structure and a line or two on S3 might be a good idea.

*Agreed. This material was already in the text but not as a separate section. I added a section describing the object structure, including the following text for clarity: 'The `swmpr` object class was created for use with the organizing and analyzing functions. This object-oriented approach is standard for R (i.e., the S3 object system, [16]), such that specific methods for a generic function are developed for the object class.'*

Page 4: Why only 100? web service limit?

*I've modified all functions that have limits on the number of records that can be downloaded, so this is no longer relevant.*

Page 4: Same as above. If there are service limitations should outline those. Just saw this explained below!

*Same as above.*

Page 4: This is going to be a tough one to keep relevant. What's the reason for including it?

*Agreed, the description of this function was removed from the manuscript but it is retained in the package. I think it's useful to include but shouldn't be advertised as a primary means of importing data.*

Page 4: Couldn't you build this functionality into the function itself? Perhaps next version... No need to include prior to publishing this.

*The most recent version now includes this functionality.*

Page 5: Might be more informative to show the results for each request... dplyr style?

*I think this might create unnecessary clutter if all the examples are shown. I've added an*

*example in the next section for the swmpr object class that shows the raw data.*

Page 5: If you built in the loop to get around the limit would this not be necessary? This seems like a bandaid that will be less and less relevant over time.

*Yes, I've removed this function from the manuscript as noted above.*

Page 5: Unless it is necessary, I wouldn't even mention this.

*Omitted.*

Page 5: Would be nice to have the decompression as part of the function.

*The function was modified to include this feature. The text was updated accordingly 'The `import_local` function can be used to import files directly from the zipped folder or after the folder is decompressed'*

Page 5: What is the source of the duplicates? Does it make sense to just remove them? Are the data duplicated? Could you just flag these in the output as opposed to removing? If you did this, you could then let the user decide when to remove or not remove with `rem_reps()`.

*This section was revised for clarity, 'Occasionally, non-unique observations are present in the raw data. These duplicates may be actual replicates with unique time stamps, such as replicate samples for monthly nutrient data. Erroneous duplicates with non-unique time stamps may also be present.' I will consider adding functionality to handle duplicates but I don't think the issue of actual duplicates with non-unique time stamps is a huge issue.*

Page 6: Re: my comment above about the object... Might make this a separate section. Include a table instead of prose to describe the object. Could closely follow the output of `str(my_swmpr_obj)`.

*A table was added to replace the paragraph description of the object attributes. A separate section was also added in response to a previous comment.*

Page 7: `setstep.swmpr??`

*See response to the next comment.*

Page 7: Not sure how best to refer to these S3 methods. You do it both ways (with `.smpr` in the table) and without in the text. Could keep it that way if you add a line to table caption. I bring it up, because I think new R users would be confused by this (I was for awhile...)

*Agreed, I've removed the method identifier from each function in the tables to avoid*

*confusion.*

Page 9: Does PLOS allow Boxes? Like Box 3. Example of `qaqc()` function usage... That would be a good way to show these.

*I've checked a few papers from PLOS One that describe software packages and most follow the format I've used, i.e., presentation of code preceded by the command prompt.*

Page 9: quote

*Added.*

Page 9: quote

*Added.*

Page 10: A section on package dependencies might be a good addition.

*I've added a footnote that directs readers to the package index page on CRAN where the dependencies and package imports can be viewed. I'm not sure it's worth describing each in detail since this is readily available in the source code.*

Page 10: Include an example of this in the code examples below.

Page 10: rewrite. "a signal from noisy data" ??

Page 13: or

Page 14: May not be obvious that this is 60 min / 15 min... And character string options are a great idea!

Page 16: I'm just being picky, but the name on this one is tripping me up. I think I'd prefer to see `eco_metab`. But, I realize that may not be an easy change, so up to you!

Page 17: Do all of these functions need to be exported? Could they simply be internal functions that the users don't see? If so, you could shorten or eliminate this section. Also you could create a Viz category of functions instead of this one. In that you could talk about the plot capabilities separate from the analysis section, and then include the `reserve_map` function there.

Page 17: Should this be `map_reserve`???

Page 19: Yank this. Goes without saying.

Page 20: Some of the figures have code that directly precedes it. Other figures (like Fig 8)

it is a bit more difficult to link code to figure. Would be nice to provide a link to code for all the figures. This way a reader could download a single script to reproduce the figures.

Page 20: Are these examples necessary? Feel kinda tacked on and why have links as supplementary info and not part of the text? If you are going to keep these, I would flesh this out a bit more, add in some screen captures as figures and links directly in the text.

Page 21: Same comment as above. Be nice to have ready access to code that reproduces this table.