

Tutorials for the Ocean Health Index

Contents

1	Software for OHI	2
2	Required Software	2
2.0.1	R	2
3	Optional Software	2
3.0.2	RStudio	2
3.1	Rmarkdown	3
4	Accessing GitHub repositories	3
4.1	OHI and GitHub	3
4.1.1	OHI regional assessments and GitHub	3
4.2	Getting started	3
4.2.1	What is needed?	3
4.2.2	Install R and RStudio	3
4.2.3	Create a GitHub account	3
4.2.4	Install <i>git</i> Application	5
4.2.5	Create your ohi-[assessment] repo	5
4.2.6	Clone your repo to your computer	5
4.3	Working locally	6
4.4	Committing and pushing changes locally	6
4.4.1	Using RStudio to commit and push changes to GitHub ohi-[assessment] repository . .	6
4.4.2	Using Github App to push commits to GitHub ohi-[assessment] repository	8
4.5	Keeping synchronized	9
4.5.1	Pull commits to get changes from anyone else.	9
5	Accessing repositories without GitHub	9
5.0.2	OHI regional assessments without GitHub	9
5.1	Getting started	9
5.1.1	Install the latest version of R	9
5.1.2	Install the latest version of RStudio	9
5.1.3	Create your ohi-[assessment] repo	9
5.1.4	Download the repo to your computer	10
5.1.5	Working locally	10
5.2	Updating your fork	14

6 Rmarkdown Tips	18
7 testing	18
7.1 Rmarkdown	18
7.2 Getting started	18
7.2.1 RMarkdown with GitHub	18
7.2.2 RMarkdown with RStudio	18
7.2.3 Asking for help	21
8 Toolbox Troubleshooting	21
8.1 Loading RWorkspace on Restart	21
8.2 Calculating Pressures...	23
8.2.1 ‘The following components for [goal] are not in the aggregation layer [layer]...’	23
8.2.2 ‘Error in matrix...’	25
8.3 Calculating Resilience	25
8.3.1 ‘Error in match(x, table, nomatch = OL) : object id_num not found’	25

1 Software for OHI

The Ocean Health Index works primarily with freely available, open-source software. To work the the Toolbox, there are several software programs you will need to download.

2 Required Software

2.0.1 R

Download [R from cran.r-project.org](https://cran.r-project.org) and follow the install instructions for your computer.

If you already have R installed, check cran.r-project.org for updates. There are frequent updates available, and the current version is identified on the website. Compare what is available with what you already have on your computer by typing `sessionInfo()` into your R console (see also: [learning R](#)).

3 Optional Software

3.0.2 RStudio

We highly recommend working with [RStudio](#), which is an interface that makes working with R much easier. RStudio also has great compatibility with [GitHub](#), which we recommend for conducting regional assessments so you can collaborate with your team easily. RStudio does not get updated as often as R does, but check for updates regularly.

3.1 Rmarkdown

[Rmarkdown](#) is a powerful way to track progress as you use the Toolbox to reflect the characteristics and priorities in your study region. Here are some [Rmarkdown tips](#).

4 Accessing GitHub repositories

4.1 OHI and GitHub

The Ocean Health Index (OHI) Toolbox is written in [R](#) and uses [GitHub](#), an [open-source development platform](#) to develop and share software and data. GitHub has powerful versioning capabilities, which allow changes to be archived and tracked by each user. This is incredibly useful to not only document what work has been done, but how it differs from work done in the past, and who is responsible for the changes. GitHub enables easy collaboration with others, including the OHI team.

OHI has several repositories ('repos') where data and code are stored. From the [GitHub glossary](#):

A repository is the most basic element of GitHub. They're easiest to imagine as a project's folder. A repository contains all of the project files (including documentation), and stores each file's revision history. Repositories can have multiple collaborators and can be either public or private.

[OHI-Science](#) is the 'owner' of OHI repositories, and individual users contribute to these repositories when they have permission. This simple example illustrates GitHub's collaborative workflow with the `ohi-israel` repo owned by OHI-Science:

4.1.1 OHI regional assessments and GitHub

We recommend that groups interested in conducting OHI assessments do so through GitHub. This will enable collaboration and transparency, and will provide access to the latest developments in the Toolbox software. To get started, follow the steps below. The OHI team will create a repository for your regional assessment.

It is also possible to conduct a regional assessment without GitHub. We do not recommend this because it is more difficult for you to track progress and decisions, and for us to help or advise you. However, if this is preferable, see the instructions [here](#). We can always get your filesystem onto GitHub later.

4.2 Getting started

4.2.1 What is needed?

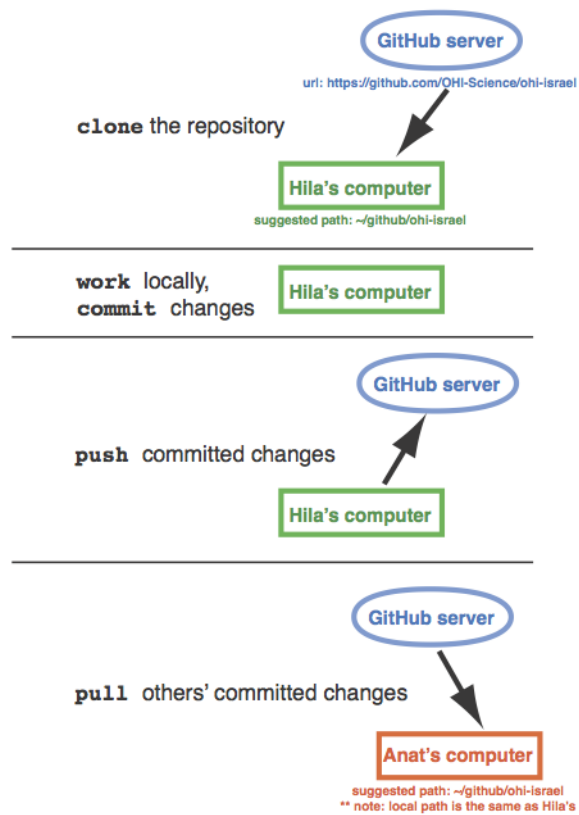
![alt text(./fig/overview_requirements.png)]

4.2.2 Install R and RStudio

Make sure you have the most current version of [R](#) and [RStudio](#).

4.2.3 Create a GitHub account

To get started, [signup](#) for a GitHub account, and provide your username to bbest@nceas.ucsb.edu or lowndes@nceas.ucsb.edu so you can access your `ohi-[assessment]` repository.



4.2.4 Install *git* Application

git is required to work behind the scenes on your computer. [Download](#) and install *git*. (Here are a few [tips](#)).

You will then need to set up your Git Identity, which identifies you with any changes made. You will use the command line:

- **Mac:** launch the Terminal application (Applications > Utilities > Terminal)
- **Windows:** go to command line in Windows (Start > Run > cmd)

Substitute your GitHub user information with the user John Doe:

```
git config --global user.name jdoe
```

and then:

```
git config --global user.email johndoe@example.com
```

You can check settings with the following:

```
git config --list
```

Quit the Terminal after typing:

```
exit
```

4.2.5 Create your ohi-[assessment] repo

Contact the OHI team (bbest@nceas.ucsb.edu or lowndes@nceas.ucsb.edu) to create a repository for your group. The repository will be stored on github.com and called `OHI-Science/ohi-[assessment]`: for example, `github.com/OHI-Science/ohi-israel`.

4.2.6 Clone your repo to your computer

Once there is a repository for your OHI regional assessment called `OHI-Science/ohi-[assessment]`, you can decide whether you will clone and work directly from that repository or from a forked repository. There are benefits to both approaches:

- Working directly from the repository is simplest: you can make changes on your local computer and push them directly to the online repository, as in the figure above. You do not need to send pull requests; simply clone from `OHI-Science/ohi-[assessment]` (see [Cloning options](#)) and push/pull from there.
- [Forking](#) a repository is best when you have multiple collaborators working on the same repository. This would allow you to commit changes to your local version, push commits up to github for offsite archiving, and eventually make a pull request to have those changes merged back to `ohi-science/ohi-[assessment]` while your collaborators do the same. This is a good way for someone who is not in the core team of the assessment to contribute comments.

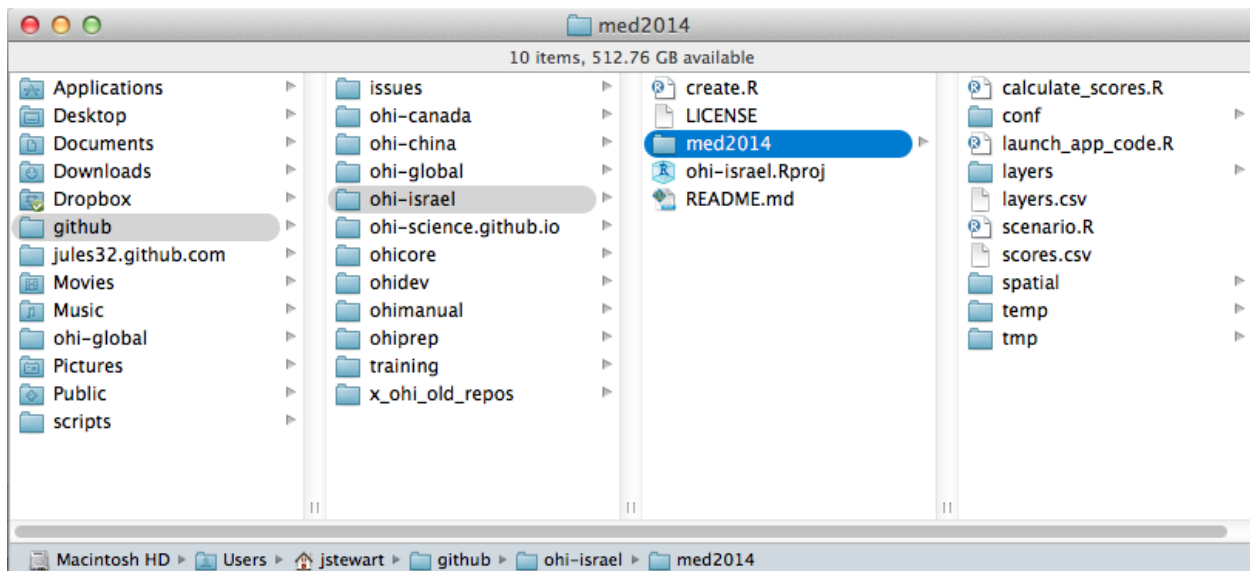
****** With either approach, we recommend creating a folder called `github` in your root directory so that you can clone the repository to this file path: `~/github/ohi-[assessment]`. This will make collaborating much simpler since everyone will work with the same file path.

4.2.6.1 Cloning options There are several options to clone the `ohi-[assessment]` repository to your local machine:

1. The GitHub App for [Mac](#) or [Windows](#). After installing the GitHub App, click **Clone in Desktop** button from the online repository (see the *Getting started guide* on a [Mac](#) and [Windows](#))
2. [The command line](#).
3. [RStudio](#). This is best after the initial clone, since RStudio occasionally has trouble with setting the username / password.

4.3 Working locally

All of the files available from the online server repository [https://github.com/OHI-Science/ohi-\[assessment\]](https://github.com/OHI-Science/ohi-[assessment]) are now cloned on your local computer, in the location you specified (we recommend cloning the repository to this file path: `~/github/ohi-[assessment]`).



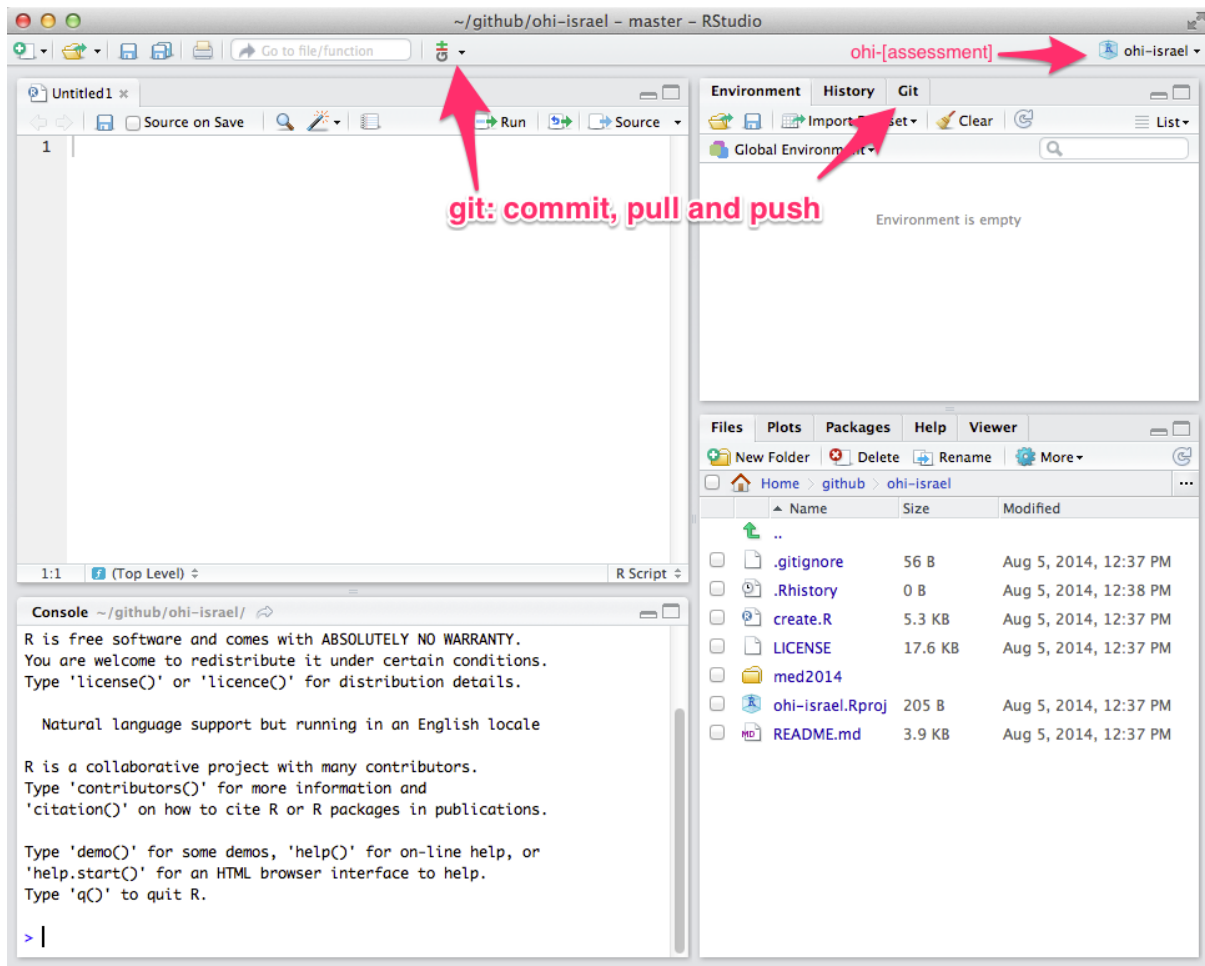
You will work on your computer to change the files in `~/github/ohi-[assessment]` to reflect the desired modifications your team has identified for your regional assessment. All changes within this folder will be tracked by GitHub when you [commit](#) these changes, *even if you make these changes outside of R or RStudio*. This means that you can delete or paste files in the Mac Finder or Windows Explorer, edit `layers.csv` and `goals.csv` in Excel or a text editor, and modify `functions.r` in RStudio, and commit all these changes to GitHub to track the changes you make.

4.4 Committing and pushing changes locally

Committing and pushing changes can be done using [RStudio](#) or the [Github App](#)

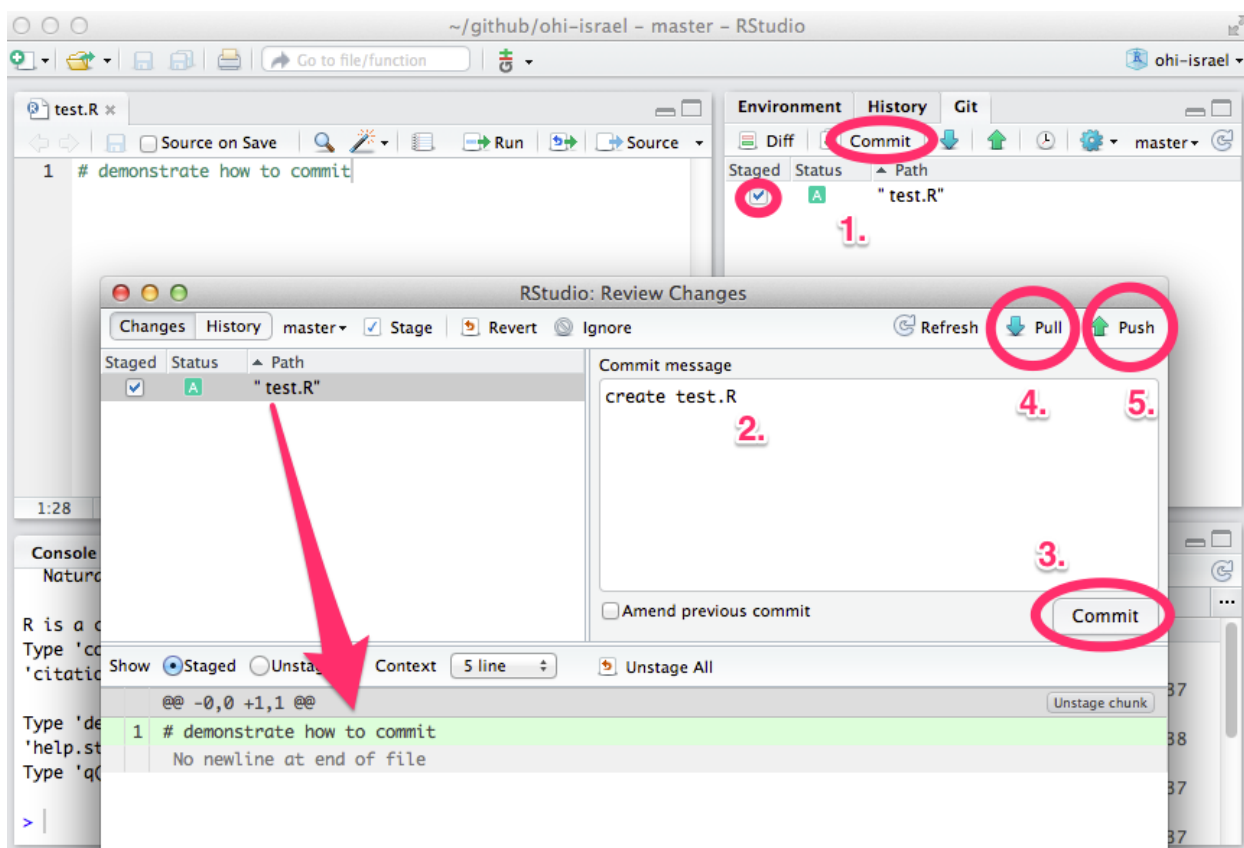
4.4.1 Using RStudio to commit and push changes to GitHub `ohi-[assessment]` repository

Launch your `ohi-[assessment]` project in RStudio by double-clicking the `ohi-[assessment].Rproj` file. From RStudio, Commit locally, associating a message with each set of changes.



When you modify or add a file, the file will appear in the 'Git' window. In the example below, the file `test.R` was created.

1. Clicking the 'Staged' box and the 'Commit' button opens a new window where you can review changes.
 2. Type a commit message that is informative to the changes you've made.
- Note 1: there will often be multiple files 'staged' at the same time, and so the same commit message will be associated with all of the updated files. It is best to commit changes often with informative commit messages.
 - Note 2: clicking on a staged file will identify additions and deletions within that file for your review
3. Click 'Commit' to commit the changes and the commit message
 4. Pull any changes that have been made to the online repo. This is important to ensure there are no conflicts with updating the online repo. See [Keeping synchronized](#).
 5. Push your committed changes to the online server at [github.com/OHI-Science/ohi-\[assessment\]](https://github.com/OHI-Science/ohi-[assessment]). Your changes are now visible online.



4.4.2 Using Github App to push commits to GitHub ohi-[assessment] repository

The Github App can also be used to commit and sync any changes made locally in the github folder. To do so, write a summary and description in the respective message bars in the Github App window, then click on commit and sync.

!alt text(/fig/github_app.png)

4.5 Keeping synchronized

Always pull before pushing commits.

4.5.1 Pull commits to get changes from anyone else.

4.5.1.1 More Information

- [presentation: Reproducible science with the Ocean Health Index](#)
- [wiki: Using GitHub](#)
- see www.oceanhealthindex.org and ohi-science.org for more information and resources.

5 Accessing repositories without GitHub

5.0.2 OHI regional assessments without GitHub

It is possible to conduct a regional assessment without having a GitHub account. You can do a one-time download from the GitHub repository and then work locally on your computer without syncing to the GitHub server. We do not recommend this because it is more difficult for you to track progress and decisions, and for us to help or advise you. We recommend [accessing a repository with GitHub](#) but provide instructions here of how to access a repository without Github. We can always get your filesystem onto GitHub later.

5.1 Getting started

5.1.1 Install the latest version of R

Download [R](#) and install this on your computer. If you already have R installed, check the website for updates. There are frequent updates to the R software, and the current version is identified on the website. Compare what is available from their website with what you already have on your computer by typing `sessionInfo()` into your R console. (This will also identify packages you have installed).

5.1.2 Install the latest version of RStudio

While not required, we highly recommend working with [RStudio](#), which is an interface that makes working with R much easier. RStudio also has great compatibility with GitHub so you can collaborate with your team directly from your R code. RStudio does not get updated as often as R does, but it is good to check for updates regularly.

5.1.3 Create your ohi-[assessment] repo

Contact the OHI team (bbest@nceas.ucsb.edu or lowndes@nceas.ucsb.edu) to create a repository for your group. The repository will be stored on github.com and called OHI-Science/ohi-[assessment]: for example, github.com/OHI-Science/ohi-israel.

5.1.4 Download the repo to your computer

When your repository has been created, enter the url in to an internet browser (for example, github.com/OHI-Science/ohi-israel). Click Download ZIP.

OHI-Science / ohi-israel

Unwatch 2 Star 1 Fork 0

Israel's Ocean Health Index — Edit

24 commits 1 branch 0 releases 1 contributor

branch: master ohi-israel / +

Update README.md

bbest authored 8 days ago latest commit 9c8c0d9f2d

med2014	updated tbx screen with scenario ohi-israel/med2014	13 days ago
.gitignore	ignoring OS-specific launch_app.*	13 days ago
LICENSE	Initial commit	2 months ago
README.md	Update README.md	8 days ago
create.R	moved scenario to med2014 subfolder	13 days ago
ohi-israel.Rproj	initialized Israel scenario files. resolves OHI-Science/ohicore#143	2 months ago

README.md

ohi-israel: Israel's Ocean Health Index

OHI Toolbox App with Israel Scenario

HTTPS clone URL
<https://github.com/OHI-Science/ohi-israel>

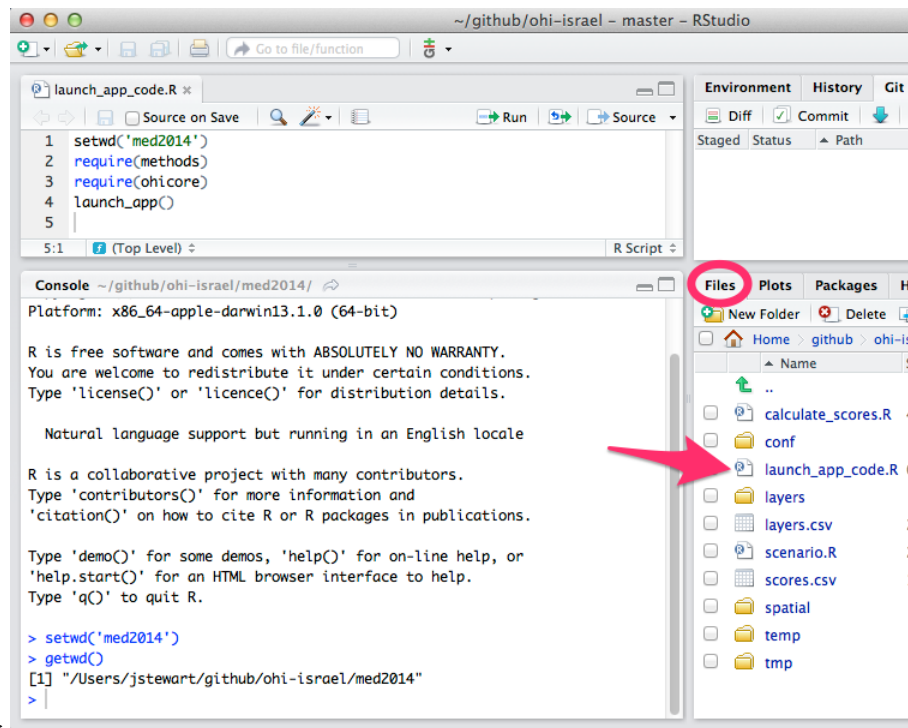
You can clone with HTTPS, SSH, or Subversion.

Clone in Desktop

Download ZIP

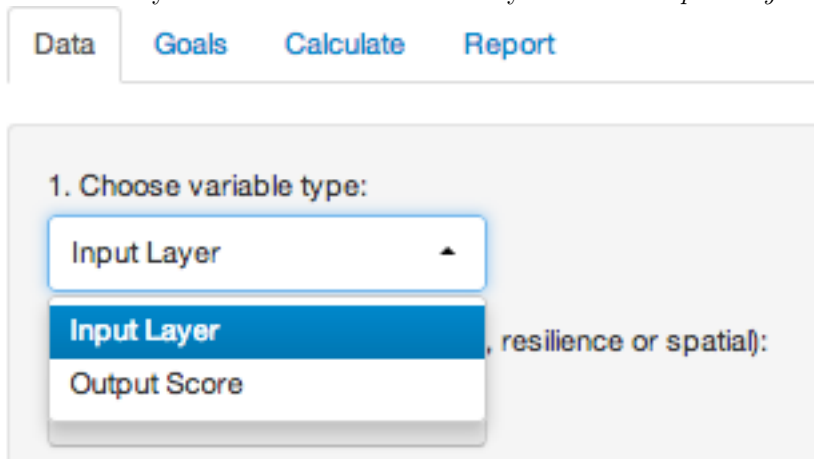
5.1.5 Working locally

1. Unzip the downloaded folder `ohi-[assessment]-master` and save the folder on your computer. We recommend creating a folder called `github` in your root directory and saving your repository to this file path: `~/github/ohi-[assessment]`. This will make collaborating much simpler since everyone will work with the same file path.
2. Double-click `ohi-[assessment]/ohi-[assessment].Rproj`: this will launch RStudio. Be sure to have the [latest version of R and RStudio](#).
3. Open `ohi-[assessment]/[scenario]/launch_app_code.R` (for example, `ohi-israel/med2014/launch_app_code.R`).



You can open this from within RStudio: >

- Launch the Toolbox user interface Run `launch_app_code.R` to launch the Toolbox user interface. Notice that when the user interface opens, it may not be able to display *Output Scores*. This is because they have not been calculated yet: select *Input Layer* instead to see the map: >



- Navigate through the Toolbox file system Become familiar with the Toolbox [file system](#). **Note: this page is under active development**
- To change data presented: Within the folder `ohi-[assessment]/[scenario]layers`, open a `.csv` file and see that there are fewer regions represented than in the [global layers folder](#).

This is because we have modified all the files to coincide with the six regions in the `ohi-israel` assessment. All these files have the suffix `_global2013.csv` because they are just copies of the global assessment, with the same value copied into each regions. For example: `alien_species_global2013.csv` has six region identifiers (`rgn_ids`, consistent with `ohi-israel`'s six regions).

alien_species_global2013.csv

Search in Sheet

Home Layout Tables Charts

Edit Font Alignment Number

Paste Arial 10 General

B I U

Align

%

A1 rgn_id

	A	B	C	D
1	rgn_id	resilience.score		
2	1	0.875		
3	4	0.875		
4	5	0.875		
5	2	0.875		
6	3	0.875		
7	6	0.875		
8				
9				
10				
11				
12				
13				

alien_species_g

Normal View Ready

At this point, only one file is specific to the assessment: `**ohi-[assessment]/[scenario]/layers/rgn_labels_[scenario].` which is the file that describes all the regions (in this example, in Hebrew):

```

"rgn_id","type","label"
1,"eez","הדרום"
2,"eez","חיפה"
3,"eez","מפרץ חיפה"
4,"eez","המרכז"
5,"eez","הצפון"
6,"eez","תל אביב"

```

For the data layers that you will update, we suggest you replace the `__global2013.csv` file, with a file that has a suffix identifying your scenario. So `alien_species_global2013.csv` would become `alien_species_med2014.csv` or `alien_species_israel2014.csv`. You can keep a copy of the old `*_global2013.csv` file in a separate folder or delete it (there is a record of it on the GitHub website ([example](#))).

7. Register the updated layers `ohi-[assessment]/[scenario]/layers.csv` is a registry (directory) of where the Toolbox looks for the data it uses. When you change a filename to `*_china2014.csv`, you will need to update the registry so the 'filename' column so the Toolbox know where to look.

	A	B	C	D	E	F	G
1	targets	layer	name	description	fid_value	units	filename
101	spatial	rgn_area_in	Region area 1 km inland	inland area 1 km from shore	area_km2	km^2	rgn_area_inland1km_global2013.csv
102	spatial	rgn_area_of	Region area offshore 3nm	offshore area 3 nautical miles from shore	area_km2	km^2	rgn_area_offshore3nm_global2013.csv
103	spatial	rgn_georegi	Georegion labels per region, at 3 georegion levels	georegions per region, at 3 georegion levels	label	label	rgn_georegion_labels_global2013.csv
104	spatial	rgn_georegi	Georegion ids per region, at 3 georegion levels	georegions per region, at 3 georegion levels	georgn_id	georegi	rgn_georegions_global2013.csv
105	spatial	rgn_global	regions used in global analysis for Nature 2012, subset	regions used in global analysis for Nature 2012, subset	label	label	rgn_global_global2013.csv
106	spatial	rgn_labels	regions by type (eez, subocean, unclaimed) and label	regions by type (eez, subocean, unclaimed) and label	label	label	rgn_labels_israel2014.csv
107	SPP	spp_status	Species lists and IUCN threat categories as a proxy for	International Union for Conservation of Nature (IUCN) R score	status_s	spp_status_global2013.csv	
108	SPP	spp_trend	Species lists and IUCN threat categories as a proxy for	International Union for Conservation of Nature (IUCN) R score	trend_sc	spp_trend_global2013.csv	
109	TR	tr_jobs_total	Total labor force	count	jobs	tr_jobs_total_global2013.csv	

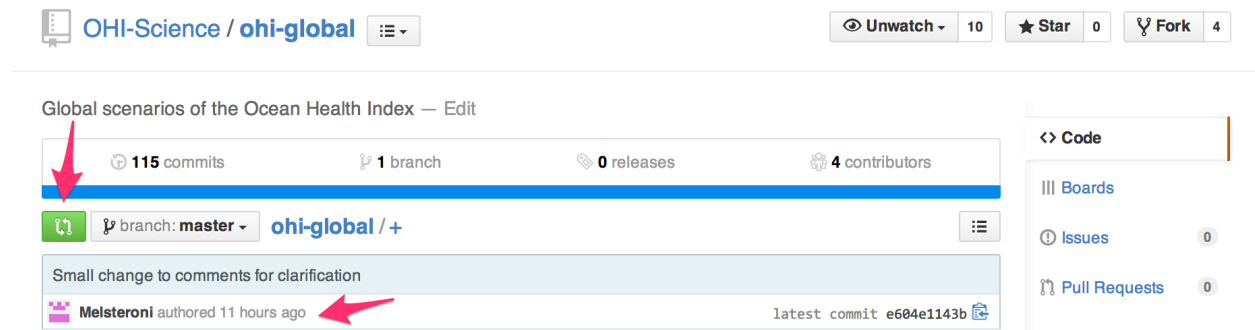
While you've been working on your forked repository locally on your computer (ex: `remi-daigle/ohi-global`), there may be committed updates in the owner's repository that is 'upstream' (ex: `OHI-Science/ohi-global`). To ensure that you are working with the most updated repository, you need to update your fork by creating a **pull request**. See some tips on process [here](#).

5.2 Updating your fork

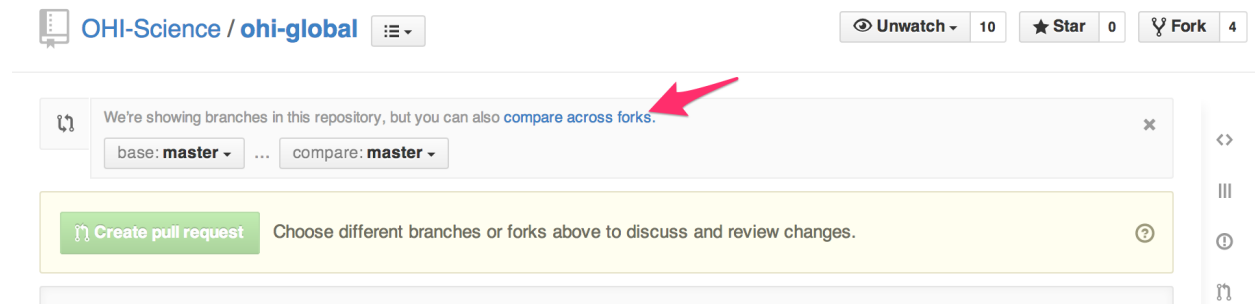
To do this on the command line, see help.github.com.

To do this through Github:

1. Go to the repository ('repo') belonging to the project owner 'upstream'. In this example, the owner is called **OHI-Science** and the repo is **ohi-global**. Note that there are recent updates by user Melsteroni. To pull the most recent version of **ohi-global** from **OHI-Science**, click on the green button.



1. You will need to sync forks (the default is to sync branches).



1. Change the base fork to your repo: in this example it is **remi-daigle\ohi-global**. The head fork is the owner's repo: **OHI-Science\ohi-global**. You can see those most recent commits by Melsteroni. Now you can click 'Create pull request'.

Note:

- when the base fork and head fork are identified, we are in the forked repo 'remi-daigle/ohi-global'. It is also possible to create a pull request the the head fork from this location.
- if that if you get an error *There isn't anything to compare*, make sure you are still comparing forks, not branches.

base fork: remi-daigle/ohi-global base: master ... head fork: OHI-Science/ohi-global compare: master

Create pull request

Discuss and review the changes in this comparison with others.

3 commits

188 files changed

0 commit comments

1 contributor

Jul 30, 2014

Meisteroni	Changes due to scenario 2012 run (july 30 2014)	610564d
Meisteroni	Changes due to scenario 2013 run on July 30 2014.	615223a
Meisteroni	Small change to comments for clarification	e604e11

Showing 188 changed files with 510,793 additions and 485,617 deletions.

Show diff stats



We're showing branches in this repository, but you can also [compare across forks](#).

base: master

...

compare: master

There isn't anything to compare.

You'll need to use two different branch names to get a valid comparison.
Check out some of these sample comparisons.

EXAMPLE COMPARISONS

master@{1day}...master 24 hours ago

1. Enter a Title and a comment, when then allows you to click *Create pull request*

The screenshot shows the GitHub interface for creating a pull request. At the top, the repository is identified as 'remi-daigle / ohi-global', forked from 'OHI-Science/ohi-global'. It shows 1 watch, 0 stars, and 4 forks. Below this, the pull request is titled 'syncing remi-daigle's repo' and compares 'remi-daigle:master' to 'OHI-Science:master'. The 'Write' tab is active, showing the commit message 'getting the latest'. A message on the right states: 'We can't automatically merge these branches. Don't worry, you can still create the pull request.' A green 'Create pull request' button is at the bottom right. The bottom of the form has a note: 'Attach images by dragging & dropping, selecting them, or pasting from the clipboard.'

1. Scrolling down on the next screen, you should either see:

The screenshot shows the review screen for the pull request. At the top, it says: 'Add more commits by pushing to the master branch on OHI-Science/ohi-global.' Below this is a green box with a GitHub logo and the text: 'This pull request can be automatically merged by project collaborators. Only those with write access to this repository can merge pull requests.' Below that is a purple box with a GitHub logo and a 'Write' tab. The 'Write' tab is active, showing a 'Leave a comment' text area. A note at the bottom says: 'Attach images by dragging & dropping, selecting them, or pasting from the clipboard.' A 'ProTip' says: 'Add comments to specific lines under Files changed.' A 'Close pull request' button is at the bottom right. The bottom of the form has a note: 'Attach images by dragging & dropping, selecting them, or pasting from the clipboard.'

- that you have been granted access
- that you must request permission before merging
- you will be notified once you have been granted permission; then continue to step 6 below

Add more commits by pushing to the **master** branch on **OHI-Science/ohi-global**.



This pull request contains merge conflicts. Please resolve them.
[Use the command line](#) to resolve conflicts before continuing.



Write Preview

Parse

Leave a comment

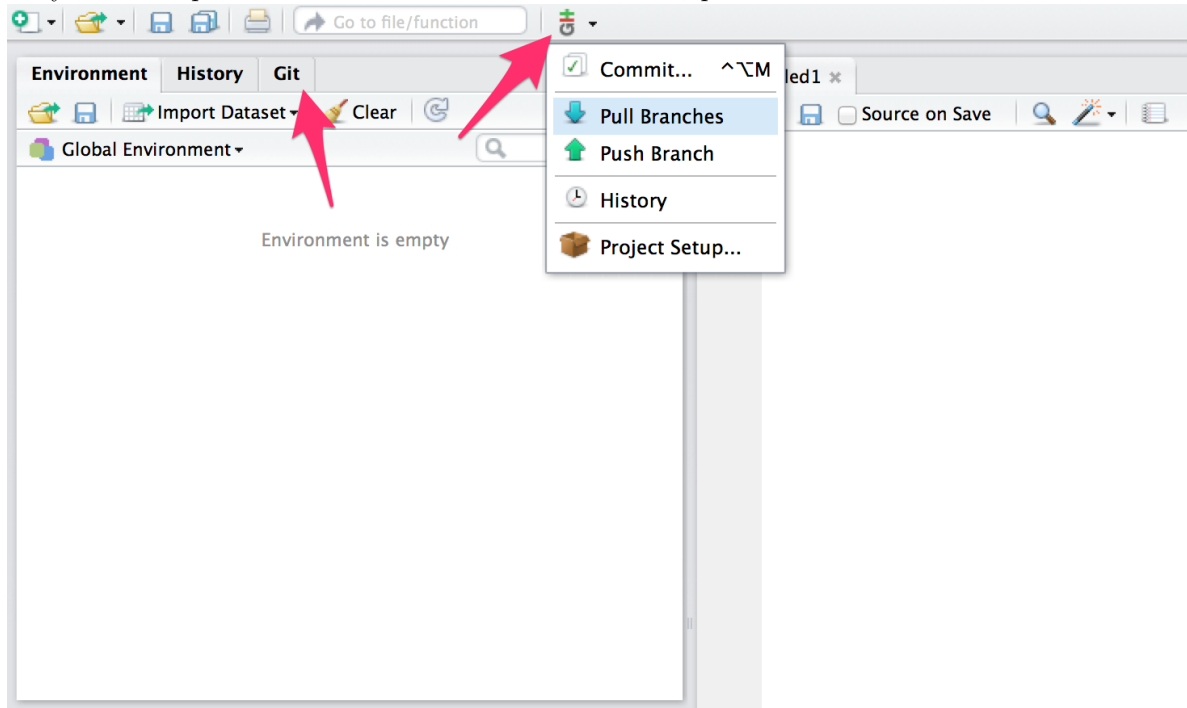
Attach images by dragging & dropping, [selecting them](#), or pasting from the clipboard.

ProTip Add comments to specific lines under Files changed.

Clos

- that you must resolve conflicts before merging
- to resolve these conflicts, follow instructions from [OHI-Science](#).

1. Once your repo on github.com is has been updated, you can pull your updated fork to your workspace in RStudio either from the Git pulldown menu or from the Git tab.



6 Rmarkdown Tips

7 testing

7.1 Rmarkdown

[Rmarkdown](#) is a simple formatting tool for creating HTML, PDF, presentations, and Microsoft Word documents, and it can render R code within those formats. See this [great post](#) as an introduction to its capabilities.

Here are a few examples of documents written in Rmarkdown, rendered in [.html](#) and [.pdf](#) formats. Also, to see how Rmarkdown files can render code, look at the comparison of three OHI scenario goals.csv files in its [raw .rmd](#) format and [rendered .md](#) format.

7.2 Getting started

With the Ocean Health Index, we use Rmarkdown documents together with GitHub. To create or edit Rmarkdown content, you can do so with R and RStudio, or directly online on the GitHub interface.

7.2.1 RMarkdown with GitHub

With an GitHub account and the correct permissions, you can [provide edits](#) to files within github.com. This will actually create a copy of the file (that GitHub calls a ‘fork’) that you can edit and then suggest changes to the owner of the file.

Because GitHub cannot render `.Rmd` files, editing those files directly is difficult because you cannot immediately see your suggested edits. You are able to see rendered edits you make to `.md` files, but this will cause a problem because the next time the `.Rmd` file is rendered, it will overwrite the edits to `.md`. The temporary solution is to make edits to the `.md` file, so you can see the rendered changes, propose the file change, and then look at the changes side-by-side while you make those same changes to the `.Rmd` file.

Select the most recent change:

Make edits side-by-side:

Your edits will be reviewed and merged into the documents by the file owner.

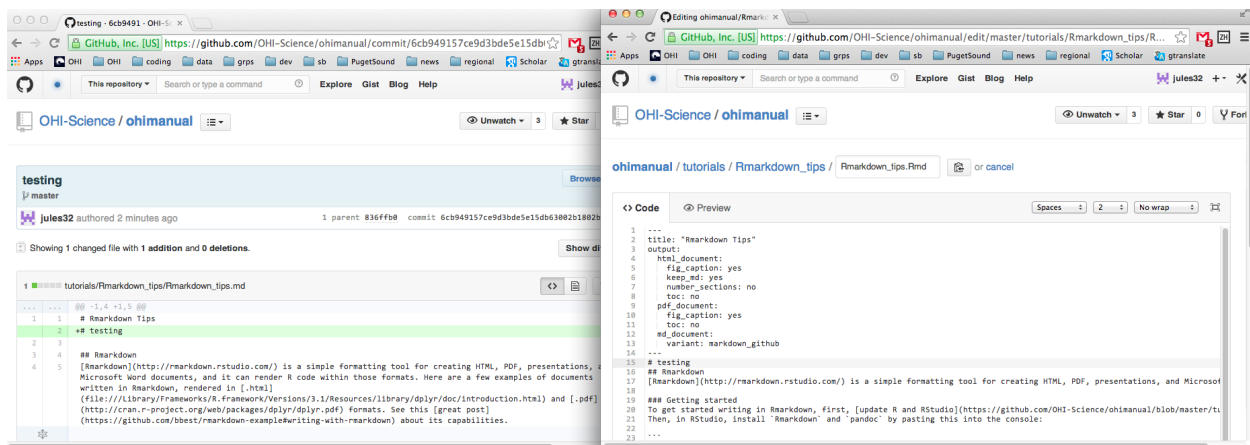
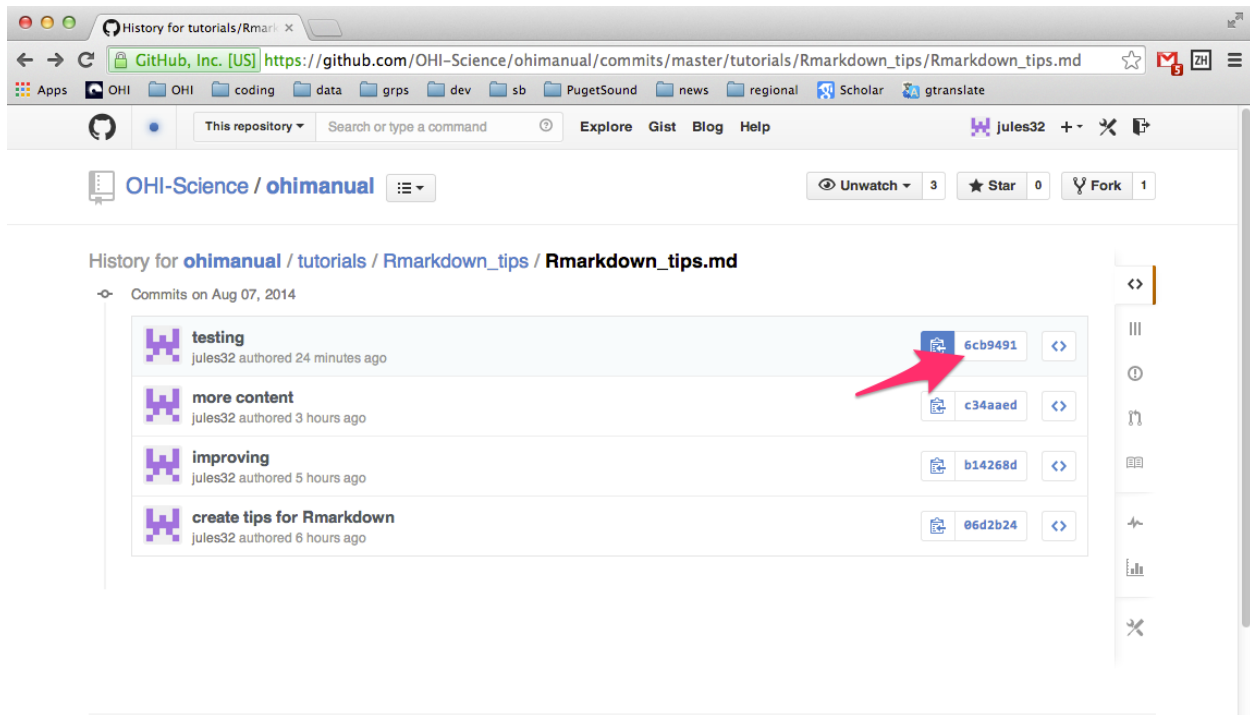
7.2.2 RMarkdown with RStudio

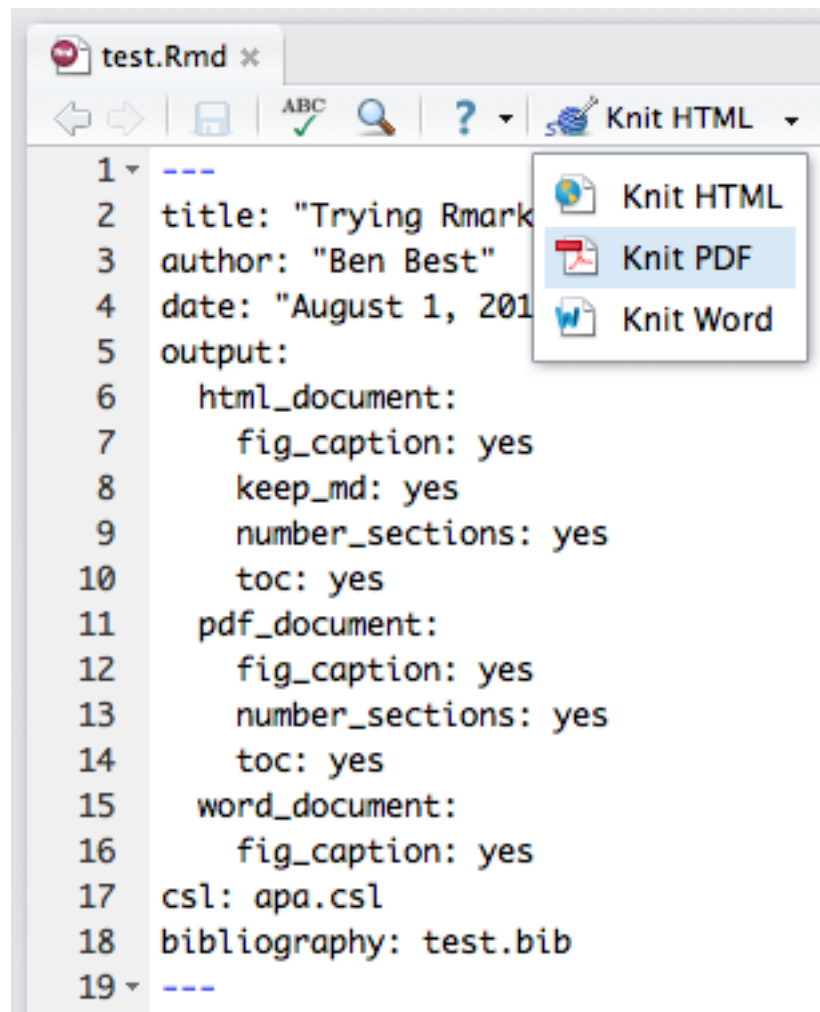
To get started writing in Rmarkdown, first, [update R and RStudio](#).

Then, in RStudio, install `Rmarkdown` (which also installs the required `pandoc`) with the pull-down menu or by pasting this into the console:

```
install.packages('rmarkdown')
```

Then, open a new [.Rmd document](#). Instead of a blank document, it will give a template that you can **Knit** to see how it renders. Try it.





7.2.2.1 Editing To add your own content, follow instructions about [syntax](#). When you click the **Knit** button, a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. Rendering the `.Rmd` as both `.html` and `.md` documents is a good way to go. The `.html` file will show the rendered content, and is a good way to test that your syntax is correct. The `.md` file needs something something more to render its format—something that GitHub can provide. So when the `md` file is on GitHub, it will be very easy to read. If you’re reading this tutorial on GitHub, you’re reading the `.md` file.

Assign the output in the following way (spacing matters). Note that there are additional options you can specify (see the *Formats* tab of the [Rmarkdown page](#) for more details).

```
---
title: "Rmarkdown output formats"
output:
  html_document:
    fig_caption: yes
    keep_md: yes
    number_sections: no
    toc: yes
  md_document:
    variant: markdown_github
---
```

7.2.3 Asking for help

Rmarkdown is a powerful tool that can create publication-worthy documents, and has a lot of built-in options. However, like any new language or software, it takes a bit of time to learn how to use it. If there is something you are trying to do, Google your problem because it is likely that someone else has had the same question, and that someone else has had the answer. For example, here are a few solutions to problems you may have:

1. [continuing numbered lists](#)
2. [saving a .md file with a different name and location than the .Rmd file](#)

8 Toolbox Troubleshooting

The Toolbox prints messages during its processing to help guide error checking and debugging. Here are a few troubleshooting tips.

(** note: this page is under development).

8.1 Loading RWorkspace on Restart

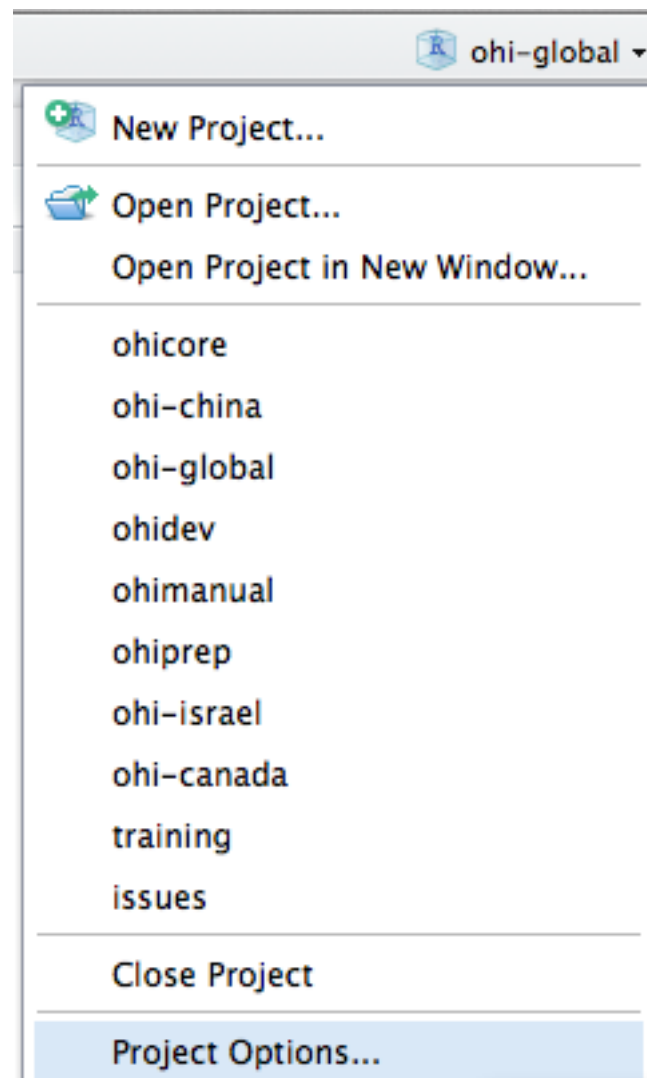
When you restart your R Session (Session > Restart R on a Mac), if you see that it is trying to load `ohicore`, it may give you an error:

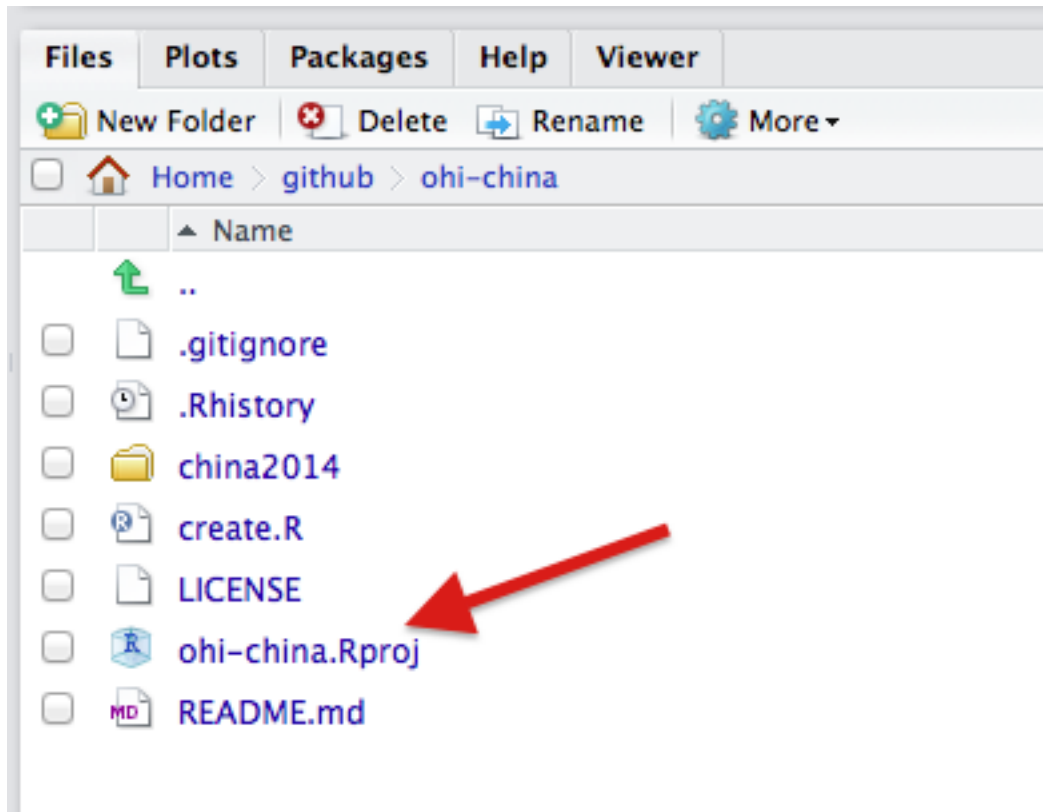
You do not want it to load `ohicore` or to save anything in your workspace. You will need to change the default setting from you `.Rproj` file. Steps to do this:

1. Go to Project Options, either in the pull-down menu or by double-clicking the `.Rproj` file:

```
Console ~/github/ohi-china/china2014/ ↗
Restarting R session...

Loading required package: ohicore
Error in .requirePackage(package) :
  unable to find required package 'ohicore'
In addition: Warning message:
In library(package, lib.loc = lib.loc, character.only = TRUE, logical.return = TRUE, :
  there is no package called 'ohicore'
Loading required package: ohicore
Error in .requirePackage(package) :
  unable to find required package 'ohicore'
In addition: Warning message:
In library(package, lib.loc = lib.loc, character.only = TRUE, logical.return = TRUE, :
  there is no package called 'ohicore'
>
```





2. Change all options to **No**:

8.2 Calculating Pressures...

8.2.1 'The following components for [goal] are not in the aggregation layer [layer]...'

Example:

```
Running Setup()...
```

```
Calculating Pressures...
```


```
The following components for NP are not in the aggregation layer np_harvest_product_weight categories (
```


```
Error in data.frame(names(P), P) :
```


```
arguments imply differing number of rows: 0, 1
```


This error means you should update your pressures matrix because it expects there to be components that your region does not have.


Project Options


General


Code Editing


Sweave


Build Tools


Git/SVN

Use (Default) to inherit the global default setting

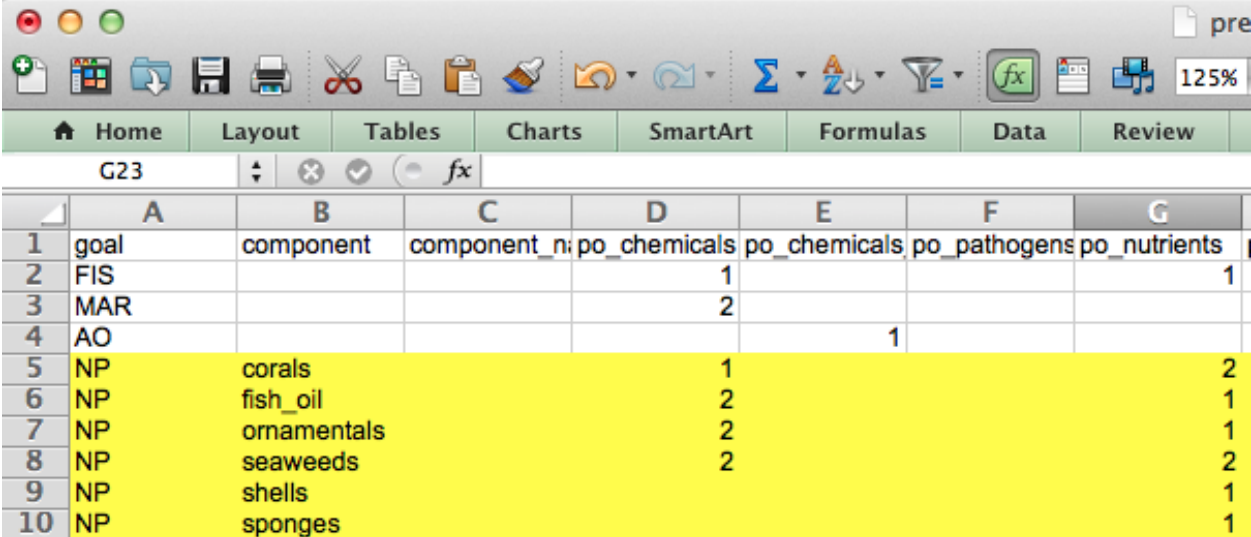
Restore .RData into workspace at startup

Save workspace to .RData on exit

Always save history (even if not saving .RData)

G23							
	A	B	C	D	E	F	G
1	goal	component	component_n	po_chemicals	po_chemicals	po_pathogens	po_nutrients
2	FIS			1			1
3	MAR			2			
4	AO				1		
5	NP	corals		1			2
6	NP	fish_oil		2			1
7	NP	ornamentals		2			1
8	NP	seaweeds		2			2
9	NP	shells					1
10	NP	sponges					1

8.2.2 'Error in matrix...'



Example: >

This error means there is an empty column in `pressures_matrix.csv`, and the Toolbox cannot handle empty columns.

8.3 Calculating Resilience ...

8.3.1 'Error in match(x, table, nomatch = OL) : object id_num not found'

```
tr_sustainability
tr_unemployment
Running Setup()...
Calculating Pressures...
Calculating Resilience...
Error in match(x, table, nomatch = OL) : object 'id_num' not found
In addition: There were 18 warnings (use warnings() to see them)
> |
```

This error means you should check that there is at least one entry for each goal (for each row) in `resilience_matrix.csv`.