Research Log

Tuesday January 9th 2018, 2pm

Learn Shiny

Use Shinyapps.io to host your shiny apps

If you eventually need to host it on AWS or something else check out Shiny Serve, some open source software for setting up a shiny app server

Stopped watching video at 1:02 and started learning how to import data in R, still cannot figure that out

--------End Jan 9 2018---------------------------------------------------------------------------------------------------------------

Jan 11

Learned how to read in data in R from csv

Meet with Dr Gill and Piccolo.

Gill want to be able to upload multiple files representing various loggers and compare values from.

Piccolo recommends starting with just two files and then scale up from there

Be able to compare the difference between the data gathered from each of the loggers by their averages, max and min.

Be able to compare two columns from the same logger

Gill said he wants to be able to aggregate the data. As in get all the reads from a day, a week or a month and compare those too data from other loggers in the same way

I set up my Weather directory as a git repo connected to my Bitbucket repo called Weather Dashboard.

I continued watching the shiny tutorials. Stopped watching at 1:33.

Figured out how to upload files with Shiny. But how to I load them just once? – Use the reactive() function

Plotted the difference between two different sets of data

Trying to make the displayed column selectable by using a drop down, but the function listening for it’s response does not seem to work. Maybe it is because I have started nesting a tone of functions?

--------------------END JAN 11-------------------------------------------------------------------------------------------------------

Jan 12

Looked in to solving the selectInput not responding issue: It turns out that in order to effect any change the code must be in a render function, mine was not before it was in a reactive() function

Set up the graphs to graph whatever is selected in the drop down menu

Ensured that Ylim is being calculated prior to plotting both sets of points in the first graph so that one set of data is not off the screen.

----------------END JAN 12----------------------------------------------------------------------------------------------------------

Jan 16th

Created radio buttons for selecting the time frame you wish to compare by as well as how you want to compare them, e.i. max, min or mean.

Created a function that merges the two input files on the timestamp column. This function then “compresses:” the data down. By this I mean that it takes the 96 rows constituting one day and takes the mean of each column. This function will eventually be abstracted out to integrate with the radio button options.

Continued watching the tutorial starting at 1:33 ending at 1:50.

---------------END JAN 16------------------------------------------------------------------------------------------------------------

Jan 18th

Fixed the diff graph so that it’s x axis displays in days, instead of being stuck in 15 minute increments

Found out that when you want to add a number to another number (or variable) after the “:” when defining a range it must be in “()” otherwise it does not work.

For example

What I was doing

Browse[2]> combind[10:10+5,2]

[1] 451

What I wanted

Browse[2]> combind[10:(10+5),2]

[1] 38 114 356 407 431 451

Hooked up the Average, Max and Min functions to their buttons

I need to find away to cache the merged file, have to remerge it and reload all the files everytime a setting is changed is too expensive for run time

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_END JAN 18\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Jan 19

Looked briefly for a answers to the caching idea, I could find nothing that seemed to be an answer. Leaving this problem to be solved at a future date when it is more important. Maybe ask Dr Piccolo about it.

Got distracted by Rex with BRG stuff.

Separated the function for merging files out into its own script to improve readability.

Trying to make the combined dataset downloadable, currently not working.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_END JAN 19\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Jan 23

Fixed the download button

Added the ability to compare a single logger by which ever two columns you want but it is not working 100% as expected. When you choose anything except time as the X axis it plots it as a 3rd axis and the does not make visual sense at all.

-------------------------------------------END JAN 23--------------------------------------------------------------------------------

JAN 25

Fixed the plotting issue. I am not sure if the data is being sorted by date properly. I am having difficulties getting the the data to sort by date, just download a library: lubridate that may help.

Went to Meeting with Gill & Pic: Gill want to be able to see error bars on the raw data graph (the non-diff graph) He also want the raw graph to have the day month and year compressing function applied to it.

Got the sorting by date working but it does not work when the incremental period is 15 minutes because I still cannot figure out how to sort by time…

-----------------------------------------END JAN 25----------------------------------------------------------------------------------

Jan 26th

Tried to convert time to something usefull for sorting but I cannot isolate the time from the date or put it all into a useable format

things <- t(as.data.frame(strsplit(dates,' ')))

names2 <- c(names[1],names[ncol(data),names[2:(ncol(data)-1)]]

insert a blank column into the dataframe at position two and shift everything else to the left

the following command can split the column

data[,1] <- t(as.data.frame(strsplit(as.character(data[,1]),' ')))

the following command can convert the the date to a date object

as.Date(datafile()[,1],'%d/%b/%Y')

Feb 1

Added time and compare method functionality to the single logger comparer and the raw data graph.

Dr Piccolo told me to cast factors as characters and then as numerics to convert them to numbers. This solves the problem of temperature being in the hundereds.

----------------End Feb 1-------------------------------------------------------------------------------------------------------------

Feb 2

Fixed the error of factors being cast as numerics that are not what they represent. I had to first cast them as characters and then as numeric.

diff <- as.numeric(as.character(col1)) - as.numeric(as.character(col2))

Started ggplot2 tutorials.

Feb 6

Added a ggplot graph to the original app

Started a new app with the intent of making it cleaner an using tibble instead of dataframe

Currently I am trying to figure out how to combined two single column tibbles into one new tibble

END Feb 6-----------------------------------------------------------------------------------------------------------------------------

Feb 8th

Got an example of ggplot up and running, outside of the Shiny app

-------------------------------END Feb 9th--------------------------------------------------------------------------------------------

Feb 13th

Got a basic ggplot example running on the new app, hard coded to average based on day

Work on selecting day, week, month of 15 min time increments now . - Done

Select Amount of Time Displayed. – Done

End Feb 13th--------------------------------------------------------------------------------------------------------------------------

Feb 15th

Add functionality to the function selector drop down: Done, but when you change the selected option it resets the Amount of time drop down. Unsure why.

Column Selection: - the whole script freezes when trying to generate the list from the colnames of the input data. Hard coding it as a vector works fine though.

Select Type of Graph: Done, but you cannot compare a single logger by any two columns, because I cannot select columns….

Error bars: added the option to include error bars on the raw and single graphs. They represent the standard error.

----------------------------------END Feb 15th---------------------------------------------------------------------------------------

February 27th

Make the graph downloadable: made it downloadable, add dimension options

------------------End February 27th-------------------------------------------------------------------------------------------------

March 1, 2018

Made the dimensions of the graph change able

Make Label’s changeable: done

Select a single column: colnames still breaks everything

Try adding a t test to and stars to the graph

-----------------------END March 1-----------------------------------------------------------------------------------------

March 6, 2018

Debugged by the debugger was not working: had to change R versions

Changing R version caused some bugs, spent time fixing those: Things that were previously ints by default were being returned as strings which broke a lot of functions

# Learning how to use: stat\_compare\_means with ggplot to plot stars for significance – gave up

# Started created my own function for seeing if error bars overlap, see line 164, it is partially done and completely un tested. Finish writing to the vector and make sure that the logic works (I am unsure if using vectors in condition checks like that works the way I want)

# When it comes time to add stars, add them as text

# -------------------------------END March 6th 2018---------------------------------------------------------------------------------

March 8, 2018

Mostly added functionality to add stars for significance , it adds stars but there are always stars showing, some are just more opaque than the others

Met with Piccolo and Gill

Gill want there to be logger column selection again and multiple loggers

Remove the significance stars, add statistics later

--------------------------------END March 8, 2018---------------------------------------------------------------------------------

March 15, 2018

Remove the stars:Done

Select Columns: Created the drop down

Using read\_csv instead of read.csv breaks the ability to use the colnames in the drop down menu, I have no idea why but thus it is

Select Single Column: Done

Select Two Columns for a single logger: Done!

--------------------END March 15---------------------------------------------------------------------------------------------------

March 20, 2018

Added the ability to upload multiple files, currently only working in the first file selector

Working on combining each group of files into one master data frame

I got up to the part where I have each all of the column names that are in common among the loggers

Next I need to extract those columns

----------------END March 20-------------------------------------------------------------------------------------------------------

March 22, 2018

Created a function for extracting the columns that in common within every logger.

Added code for averageing all of the common columns together: it is waaaaaaaay to slow, 15 minutes later was still not halfway through the loop.

TODO: use apply() to accomplish the needed looping but quicker

------------------------END March 22-----------------------------------------------------------------------------------------------

March 23, 2018

I implemented the averaging function using apply(), but it does not save to the new\_data object like it does out side of the shiny app,

I need to find away to pass by reference

---------------------------------END March 23 --------------------------------------------------------------------------------------

March 29, 2018

Abstracted out the cobind group function:

* double check that it still works, appears to have not been or be going way slower: work just make sure the first group is uploaded before trying to upload the second group

After averaging the groups, we want to only keep columns that are in common between the two groups

* debug what happed to give me all NA values near line 282: it is being converted to a factor some where…

Visually there is nothing new on the app. But I have added the functionality to upload groups of loggers for comparison. This in includes a function for averaging all the columns that are held in common across loggers. The first version of this function took over 30 minutes to average 3 loggers of ~30,000 rows. I refactored the averaging functions and got the run time down to ~2 minutes. The averaging function also keeps track of the number of loggers that contributed to each averaged value. This way if one or more loggers have missing values at a point we will know. I remember Dr Gill mentioned this would be useful so we can color portions of the graph based on logger coverage. Right now I am extending the app so it can have up to 3 groups of loggers uploaded for comparison.

April 5 2018

Sped things up a little, loading 3 loggers now takes a little less than a minute.

Fixed bug causing the columns to be converted to factors instead of numerics

Make interface for selecting columns:

Found 2 packages for displaying tables

One is interactive: rhandsontable

One is not: DT

DT can render large files

Rhandsontable breaks down the entire app if you try to load a huge file: use this just to make the column names changeable

ToDo: learn to dynamically create a data.frame that will contain only the column names, use HOT to display col names and DT for the actual data

HOT cannot add or remove rows if you they are using column types to disable col types use the following parameter: useTypes = F for example:

rhandsontable(DF, useTypes = F, stretchH = "all")

April 12, 2018

Create a matrix of 1 row and N cols, fill with colnames

End of Semester

Start of time in Germany

May 11, 2018

Work on colnames being responsive to changes made on the front end

But first: figure out where and how the break points are caught for updating column names

May 14, 2018

Figures out how the hots respond and isolated the column name that got deleted

Next: delete the data column that is associated with the removed column name

* removed it from the values object, now that needs to be reflected in the front end HOT table: they are removed with no additional code

May 16, 2018

If someone removed a row from the main data section, remove that column name as well: This is not working. I am unable to figure out why. I will leave this undone for now, desired functionality still can be achieved but required an additional step, deleting the data column and then the name column

TODO: makes sure that this part of the app still works when it is given a larger and a real logger

May 29th 2018

Trying to get columns to line up between the colnames and the data and respond to scolling on each other. Currently trying to force cells to be 50 X 10 but it will not be 10 high for some reason

June 29th 2018

Disable scrolling .ht\_master .wtHolder add element style overflow: none to disable

^^ apply

That crap up there doesn’t seem to work

New idea, screw HOT and use a generic table rendered with normal shiny, the use shinyjs to regester onclick events for each column, when you click it will highlight the column and give you the option to rename or remove it.

To do this first

Control highlighting from the JS on user end

When you click on something fill in a field with that column number, make that field something generated from Shiny, this way you can access that info on the backend to update the dataframe

July 6th 2018

The new tabletest is working pretty well

I can select table columns and it will highlight them. Working on the back end now

TODO: debug why the renaming functionality is not working. The selected col number always comes out as NA

July 9th 2018

Renaming is now good to go using Shiny JS interface

I used this tutorial as reference: https://shiny.rstudio.com/articles/js-send-message.html

TODO: delete a column of data : DONE

Prevent from deleting all the columns: there must be at least 2 : Done

TODO: Learn to make a multi page application. Current idea:

Use tabs, have a load files tab, preview data tab and analysis tab

Create file display widget

* Click event for cabinet files is not working\*\*

July 10th 2018

Resolved Clicking event issue

* Js from within the www/ directory cannot be found for a still unknown problem: move the code into scrolling-overload.js
* Rather than using the normal click() function you must use on() referenceing the ID of a parent
  + Example: $(‘#parent’).on(‘click’, ‘.child’ function(){ /\*Code\*/ })

Make the widget take an array of filenames as input and load the cabinet with all of those: Done

Add the buttons for Add to Group, Group dropdown, Name, Create Group : Done

Add the file loading section: Done

Save files in a master data structure that can be shared and appended to: Done

Populate cabinet with file names: Done

Add group button: Done

Create filing cabinet for the currently selected group:

* When new group is created create a new list for files :done

Take the selected groups and add them to the selected group:

Every time a file is clicked on in the waiting list, send a list of all selected file indexes back to the server: This is in progress: finished the empty function in JS

July 11th 2018

JS now send an array of index back, they are zero based, R in 1 based. Clicking too quickly will confuse the JS’s ability to track what is and is not selected.

Move the selected files into the chosen group: when you click add to group they are then added to the groups’s data structure it the desired location.

They are NOT currently removed from the waiting list

TODO tomorrow:

* Create the group preview area
* Remove groups from the waiting list after joining a group

July 12th 2018

Created the group preview area

* Remove groups from the waiting list after they are added to a group:
* When you change selected group, change what is in the group display

TOFIX tomorrow

Adding to a second group does not work

* There are still files in the selected from waiting list when it gets to that portion of the code
* It adds the index relative to what is currently in the waiting list, once things are removed from the waiting list this is no longer consistent with what is in the files list

July 13th 2018

I decided to ditch the current file loading page and make a new one that just has an upload file selector and a DT datatable for displaying the file names and the group they belong too, much easier.

July 14th 2018

Combined file loader, data previewer together into navApp.R, they work together well.

Currently rebuilding the plotting functionality

BUG \*\*\* : I am giving up on for now: when plotting 3 sets of data on one plot the 3rd color does not get it’s own color, it is give that of the first one and it’s group name is not included in the legend

Select Columns: Done and it feeds to the main plot

Add graphing options:

* Line: done
* Axis labels: done
* Plot title: done
* Download plot : done, but it doesn’t auto name…

Next big thing: add options to generate averaged files to compare with

* Created function average\_time\_frames that takes a tibble and summarized the data into hour, day and month increments of the mean max and min
* Next create function that combines multiple files together:

BUG TO FIX TOMORROW

My combining files function currently runs out of stack memory while trying to make the final data structure convertion

Ideas to fix:

* Break the function up into many functions
* Return a matrix and 2 vectors, the future row and col names and covert to tibble out of the function

July 16th 2018

Fixed the above listed bug by returning a matrix and 2 vectors and combining them outside of that function

Added functionality to create the group files and the monthly, daily and hourly datasets

BUG: if an hourly, daily or monthly data set already exists we need to overright the old one, not create a duplicate. DO LATER

TODO: move all needed code into a clean separate repo