WGFP_Enc_Summaries_RShinyApp How-To

Location:
U:\Projects\Colorado_River\Windy_Gap_FishMovementStudy\Data\RFID\Detections\CodingDetections\WGFP_Enc_Summaries_RShinyApp

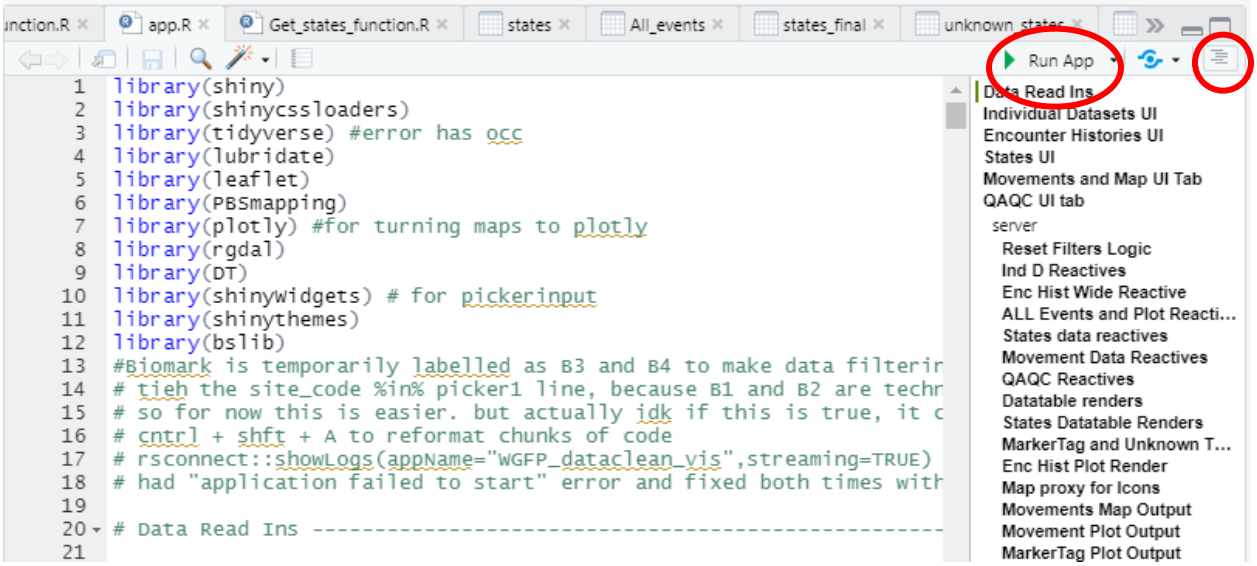
Uses

- Cleans, combines, and displays data from OregonRFID readers, Biomark antennas, mobile runs, Recaptures, and release data into one file without marker tags or the one weird ghost tag from winter 2021.
- Shows fish movement by day in map, table and plot
- Shows dataset of transitions and physical “states”
- QAQC of marker tags, lengths and weights, unknown tags

How to Open

OPENING FROM U: DRIVE

1. Navigate to App location and open the WGFP_dataclean_vis “R Project” file. Rstudio will open
2. If the “app.R” file isn’t already open in the top left panel: in the bottom right panel, navigate to the “files” tab and open the app.R file.
3. Click “Run App” in the top right of the screen. You may be required to download some packages if this is the first time running the app. To display the outline of different sections, click the panel on the right.



CSV ReadIns

The app takes 6 .csv files to run. If the column names of any important columns change in the csv's, it will require a bit of easy tweaking to the code.

When a new version of these files comes out, simply change the first argument of the read_ins to the new file name. For example, WGFP_Raw_20220203.csv would become WGFP_Raw_20220303.csv.Put the old file in an archive folder.

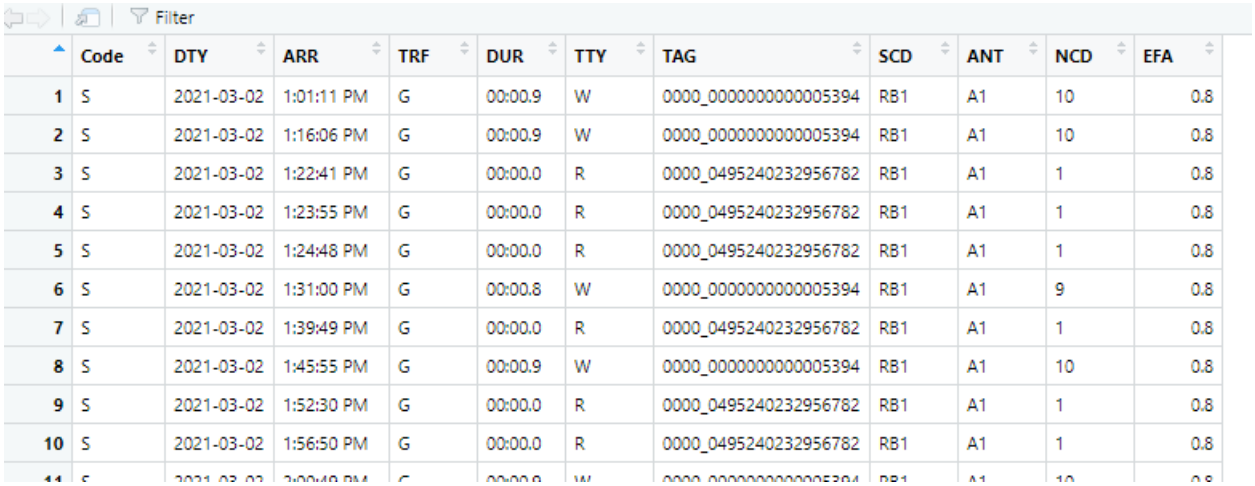
```
# if column names change in any of these read-ins, might require some modification to cc
Stationary <- read.csv(paste0("WGFP_Raw_20220203.csv")) #WGFP_Raw_20211130.csv WGFP_Raw_
Mobile <- read.csv("WGFP_Mobile_Detect_AllData.csv" , colClasses= c(rep("character",14),
Biomark <- read.csv("Biomark_Raw_20211109_1.csv", dec = ",")
Release <- read.csv("WGFP_ReleaseData_Master.csv", na.strings = c("","."),"NA"), colClass
Recaptures <- read.csv("WGFP_RecaptureData_Master.csv", na.strings = c("","."),"NA"), col
Stationdata1 <- read_csv("EncounterHistory_AllData_wStations_20220114.csv",
                         col_types = cols(
                           #OBJECTID = col_skip(), Join_Count = col_skip(), TARGET_FID =
                           TAG = col_character(), Release_Length = col_number(),
                           UTM_X = col_character(), UTM_Y = col_character(),
                           #Date_ = col_date(format = "%m/%d/%Y"),
                           Release_Weight = col_number()))
```

Column names and order matter. The read-ins and code could be tweaked so that at least column order won't matter (use read_csv and specify column types instead of read.csv), but names will still matter.

Column names outlined below are how they appear when brought into R at the beginning of the app using the code above. In Excel, names will appear slightly differently.

- Stationary
    - Combined file of all stationary detections. Obtained from the "Combine data RShiny app" found at U:\Projects\Colorado_River\Windy_Gap_FishMovementStudy\Data\RFID\Detections\CodingDetections\WGFP_CombiningData_RShinyApp.

- Most recent file is found at U:\Projects\Colorado_River\Windy_Gap_FishMovementStudy\Data\RFID\Detections\All_Stationary



Filter

	Code	DTY	ARR	TRF	DUR	TTY	TAG	SCD	ANT	NCD	EFA
1	S	2021-03-02	1:01:11 PM	G	00:00.9	W	0000_0000000000005394	RB1	A1	10	0.8
2	S	2021-03-02	1:16:06 PM	G	00:00.9	W	0000_0000000000005394	RB1	A1	10	0.8
3	S	2021-03-02	1:22:41 PM	G	00:00.0	R	0000_0495240232956782	RB1	A1	1	0.8
4	S	2021-03-02	1:23:55 PM	G	00:00.0	R	0000_0495240232956782	RB1	A1	1	0.8
5	S	2021-03-02	1:24:48 PM	G	00:00.0	R	0000_0495240232956782	RB1	A1	1	0.8
6	S	2021-03-02	1:31:00 PM	G	00:00.8	W	0000_0000000000005394	RB1	A1	9	0.8
7	S	2021-03-02	1:39:49 PM	G	00:00.0	R	0000_0495240232956782	RB1	A1	1	0.8
8	S	2021-03-02	1:45:55 PM	G	00:00.9	W	0000_0000000000005394	RB1	A1	10	0.8
9	S	2021-03-02	1:52:30 PM	G	00:00.0	R	0000_0495240232956782	RB1	A1	1	0.8
10	S	2021-03-02	1:56:50 PM	G	00:00.0	R	0000_0495240232956782	RB1	A1	1	0.8

- This is a sample of what the data should look like. The most important columns are DTY, ARR, TAG, and SCD. The ARR column can sometimes get corrupted and brought in incorrectly for whatever reason. This is easiest seen when using the CombiningData_RShinyApp and outlined in that How-To

- Mobile
  - A combined file of all mobile detections found at U:\Projects\Colorado_River\Windy_Gap_FishMovementStudy\Data\RFID\Detections\MobileRaftAntenna\Mobile_Detections
  - Usually Eric R prepares this
  - Column names and order: "Num", "River" , "MobileSite" "Date", "Time" , "T109_C" "UTM_X", "UTM_Y" ,"TagType" , "TagID", "Event", "Ant", "Pass", "Species" , "Length" , "Weight" , "TagSize" , "RS_Num", "ReleaseSite" ,"Survey", "Notes"
  - Of these, TagID, Date, Time, UTM_X, UTM_Y, and Ant are the most important columns
- Biomark
  - Combined file of all Biomark detections, found at U:\Projects\Colorado_River\Windy_Gap_FishMovementStudy\Data\RFID\Detections\Biomark
  - Made from the "Combine data RShiny app"
  - Column names and order: "Scan.Date" , "Scan.Time" , "Download.Date", "Download.Time", "Reader.ID" , "Antenna.ID", "HEX.Tag.ID" , "DEC.Tag.ID", "Temperature.C", "Signal.mV" , "Is.Duplicate", "Latitude", "Longitude", "File.Name"
  - Most important columns are Reader.ID, Scan.Date, Scan.Time, DEC.Tag.ID.
- Release

    - Master Release file of all tagged fish. Found at U:\Projects\Colorado_River\Windy_Gap_FishMovementStudy\Data\RFID\Tagging
    - Column names and order: "RS_Num", "River", "ReleaseSite", "Date" , "Time" , "UTM_X", "UTM_Y", "Species", "Length" , "Weight", "TagType", "TagID", “QAQC", "TagSize", "Ant" , “Event", "FinClip", "Mortality", "Comments"
- Recaptures
    - File of all fish that were recaptured found at U:\Projects\Colorado_River\Windy_Gap_FishMovementStudy\Data\RFID\Recaptures
    - Column names and order: "Num" , "RS_Num", "River" , "RecaptureSite" ,"Date", "Time", "UTM_X" , "UTM_Y", "Species" , "Length", "Weight", "TagType", "TagID", "QAQC" , "TagSize" , "Ant", "Event", "FinClip", "Mortality" , "Comments"
- StationData1
    - File of condensed, pertinent detections with info of “stations”, how far up or down the river you are
    - Made by downloading a condensed encounters csv from the app, then Eric R brings it into GIS and adds Stations to it
    - Colnames and order are "Date_" , "Time_", "Datetime_" , "TAG", "Event" , "Species", "Release_Length", "Release_Weight", "ReleaseSite", "Release_Date", "RecaptureSite" , "Recap_Length" , "Recap_Weight" , "UTM_X", "UTM_Y", "River", "ET_STATION"

Functions

There are currently 5 functions used in the app.

```
#functions
source("All_Combined_events_function.R")
source("Combine_events_stations_function.R")
source("enc_hist_wide_summary_function.R")
source("get_movements_function.R")
source("Get_states_function.R")
```

- All_Combined_events_function
    - Main function that combines and cleans detections and release files
    - Takes Stationary, Mobile, Biomark, Release, and Recapture files as arguments
    - Returns:
        - WGFP Clean: a clean dataset of all stationary detections, no weird tags or marker tags, or duplicate rows. All timestamps and dates are in the same format. 900_ is taken off of the tags

        - All_Events: cleaned dataset of all 5 input datasets, containing every detection/event from each tag that hit an antenna, complete with release info.
        - Marker_Tag_data: file of just marker tags, to be used in QAQC
        - Recaps_detections: a file of all antenna detections and recaptures, but no release data. This is just to be used in enc_hist_wide_summary_function
- Combine_events_stations_function.R
    - Puts stations onto a condensed All_events dataframe
    - Takes All_events dataframe from ALL_Combined_events_fuinction and StationData1
    - Returns:
        - All_events_days1: a condensed dataframe of all pertinent daily detection info for each tag with stations attached.
            - Row entries for tags with multiple detections on the same UTM_X, UTM_Y, at the same antenna, on the same day are filtered out. Leaving the first and last detections of the day, and all detections on unique antennas and UTM's in between.
- Enc_hist_wide_summary_function
    - Makes a summary dataframe of all released/tagged fish with over 60 columns of summary info, like "total number of antenna encounters", "total distance travelled", "moved through dam or not"
    - Takes recaps_detections from All_Combined_events_function, Release, and all_events_days1 from Combine_events_stations_function
    - Returns:
        - ENC_Release_wide_summary : Summary "wide" dataframe of each tagged fish and summary info
        - Unknown_Tags: dataframe of tags that have encounters, but have no release info
- Get_movements_function
    - Makes a condensed dataframe of fish "movements"
    - Takes all_events_days1 from Combine_events_stations_function
    - Returns:
        - Movement_table_no_trans: dataframe where only daily movements are included per tagged fish on unique antennas and UTM's. A movement is defined as when a fish is detected on a antenna and it has changed its station
- Get_States_function
    - Makes "states" dataframe of fish that are physically recaptured or released (Biomark and Mobile Detections, Release, Recaptures) and transitions (fish that pass over Stationary antennas so movement is inferred)
    - Takes all_events_days1 from Combine_events_stations_function
    - Returns:

- All_States: Dataframe of daily States of fish. Certain assumptions are made to infer transitions, such as if a fish hits 1/2 antennas at a site then hits an upstream antenna, it is said that an upstream transition originally occurred and it just missed 1 antenna. For more assumptions see thee Daily States Tab section
- Days_and_states_wide: same data as all_states but in wide format where days are the columns, TAG is the first column, and values are states
- Unaccounted Movements: States that couldn't be accounted for in the code

Shapefile/Polygon Readins

The file "map_polygon_readins.R" reads in all shapefiles for the movements map. Also contains some graphics options for release sites, stationary antenna sites, and labels for stations.

```
#mapping
source("map_polygon_readins.R")
```

- Updating/adding
  - Head to the GIS database and export desired layer as shapefile, saving it in "gis" folder within the app directory
    - Right click on layer in left column, scroll down to data -> export
  - Using the rgdal package, read in the .shp file

```
layer_location <- file.path("./gis/")


stationary_antennas <- readOGR(dsn = layer_location, layer = "stationary_points")
stationary_antennas <- sp::spTransform(stationary_antennas, CRS("+init=epsg:4326"))
```

  - The espg:4326 converts the projection to something usable
  - If the layer is very large, it might be good to convert the .shp file to a .rds file, which decreases resolution. For example, the stations file is a .rds file and was converted below. It keeps 10% of the original resolution

```
stations_10m <- readOGR(dsn = layer_location, layer = "stations_10m")
stations_10m <- sp::spTransform(stations_10m, CRS("+init=epsg:4326"))
simple_stations1 <- ms_simplify(stations_10m, keep = .1)
write_rds(simple_stations1, file = file.path(paste0(layer_location,"/"), paste0("simple_stations.rds")))
```

  - It is read in like so and is much faster than when read in as a shapefile

```
simple_stations2 <- read_rds(file.path("./gis/simple_stations.rds"))
```

  - If you make a new layer that you want to put on the map, you'll have to add it to the map

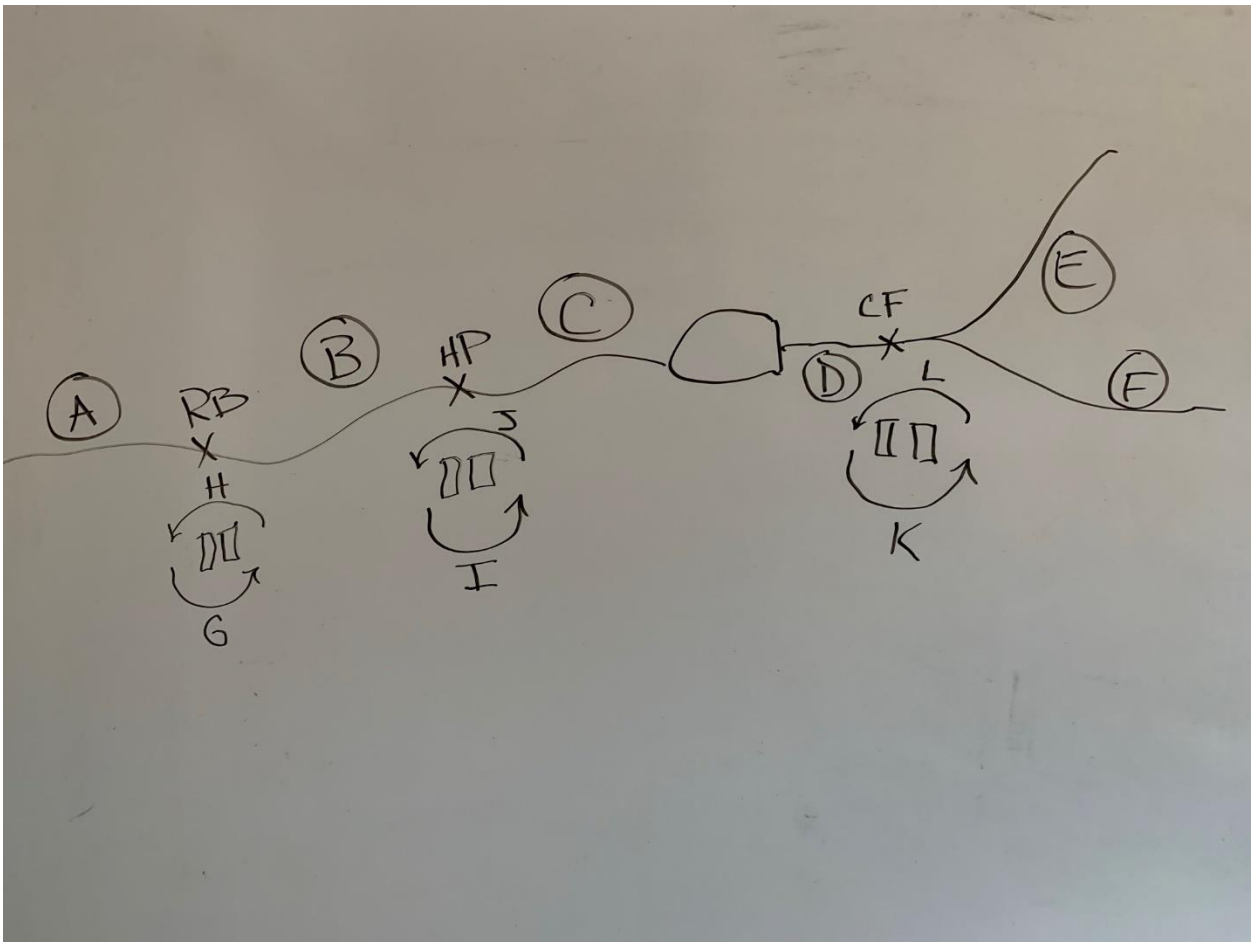
```
addAwesomeMarkers(data = releasesites@coords,
                  icon = release_icons,
                  clusterOptions = markerClusterOptions(),
                  label = releasesites@data$ReleaseSit,
                  popup = paste("Release Date1:", releasesites@data$ReleaseDat, "<br>","Release Da
                  group = "Release Sites") %>%
addPolylines(data = simple_stations2,
             label = simple_stations2@data$ET_STATION,
             labelOptions = labelOptions(noHide = T, textOnly = TRUE, style = label_style),
             group = "Stations (m)") %>%
addLayersControl(overlayGroups = c("Antennas", "Detections", "Release Sites", "Stream Centerlines"
hideGroup(c("Stream Centerlines", "Stations (m)", "Antennas", "Release Sites", "Mobile Reaches"))
```

- Add_polylines is used to bring in lines like stream_centerline, addMarkers is used for SpatialPoints, addPolygons is used for polygons.
- Add a Group argument to the data and add it in the addLayersControl function to make sure it shows up

Navigating the App

About/How to Use Tab

This tab is a concise version of this document. It includes this states diagram on how “States” are defined.



Individual Datasets Tab

This Tab shows the following tables and are all controlled by the same date filter in the sidebar. Click Render table to display all tables.

- Stationary Clean: A clean dataset of all Stationary Data (WGFP_Clean) with no weird tags or marker tags, or duplicate rows. All timestamps and dates are in the same format. 900_ is taken off of the tags
- Biomark: All Biomark combined detections from Kaibab Park and Windy Gap including Marker Tags. In the future, it might be a good idea to separate out Marker Tags from this file as it grows.
- Mobile: All Detections from all Mobile Runs
- Recaptures: All Recaptured fish
- Release: All release data

Encounter Histories Tab

- Encounter Release History Summary Wide: shows ENC_Release_wide_summary Dataframe from enc_hist_summary_wide_function.
    - Filters are pretty self-explanatory. Click Render Table/Data to display new tables with filters.
    - Columns and filters to note:
        - SiteCode_n is the raw number of detections at one antenna, or recaptured, etc.
        - SiteCode/Recapture binary columns are just weather or not a fish was detected or captured by that method
        - TotalEncounters adds the number of unique Events (a recapture, stationary detection, mobile detection, etc). There is a filter for this column in the sidebar and the highest number it could be right now is 11.
        - Through_dam tells weather a fish has gone through the dam in its history, based on release info, recaps, and detections. It does NOT tell if a fish went upstream or downstream through the dam. There is also a filter for this in the sidebar
        - Sum_dist is the total number of meters travelled by a fish in its history. There's a filter for this in the sidebar.
- All Events and Plot: Shows and plots All_Events Dataframe from All_combined_events_function. These are raw detections. 230000142723 has over 120k detections on its own at Hitching Post.
    - Most filters are self-explanatory.
    - Remove Duplicate Days, TAGs, Events and UTMs condenses detections into the first and last detections of the day as well as any other antenna detections with unique UTMs. In a way, it filters out most of the redundant detections. For example, using this filter condenses TAG 230000142723 down to 996 entries. This equates to about 259 days of detections from May 6, 2021 to Feb 3, 2022. All at Hitching Post.
        - The result of this filter is downloaded, brought into GIS, and used to make stations with.
        - This is one of the most "useful" files you can get from this app because of how it shaves off the dataset to the most relevant detections

- Remove Duplicate Tags Filter is helpful in answering questions involving “How many unique fish?”. Example: on this tab you could answer the question “How many unique rainbow trout over 250 mm hit Windy Gap Biomark and Hitching Post antennas during the day in spring 2021?”
- The plot will plot anything made with the filtered data

Daily States Tab

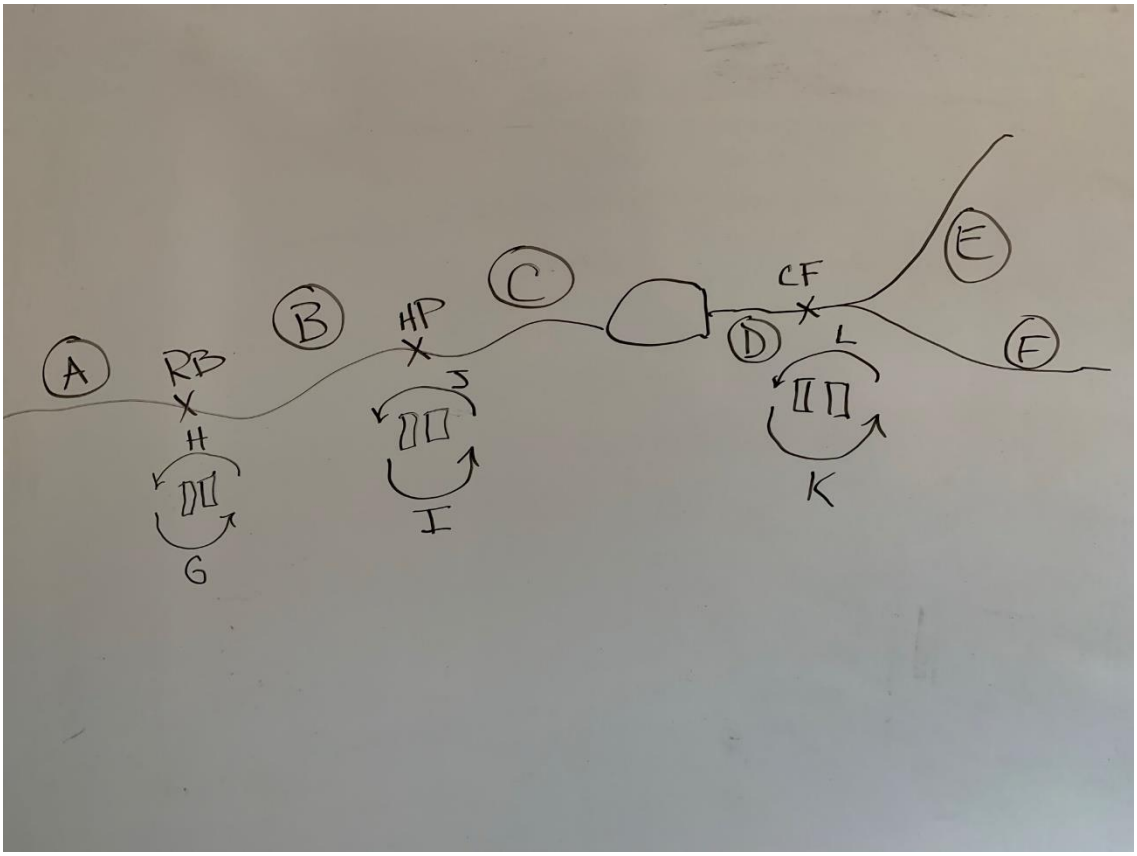
This is where “States” are displayed. A State of a fish is defined as either physically existing in a section of river (States A-F from downstream to upstream) or transitioning between physical states.

A upstream transition between states is defined as states G, I, and K at Red Barn, Hitching Post, and Confluence respectively. Downstream transition states are H, J, and L at Red Barn, Hitching Post, and Confluence respectively.

Detections across 2 stationary antennas of the same site is used to infer directionality of movement. Detections on B3, B4, Mobile Runs, and Recaptures are coded as physical states.

Transitions occur when a fish crosses 2 stationary antennas and now is assumed to exist in a different physical state. These are different from “Movements”, which occur when a fish is detected at a different station on the river than the previous event.

See the photo below for a diagram.



Since detection, release, and recapture data are rarely straightforward, there are certain assumptions that are made in order to make these states. The code tries to capture every event as a state, and the events that it is unable to do so for are found in the “unknown states” tab.

As of 02/11/2022, some brief overview of assumptions are:

- A fish resides in its release state before it is detected or captured again.
- If a fish resides in a state downstream of a stationary antenna site (RB, HP, CF) and ends the day by hitting the upstream-most antenna of that stationary site, it is said to have made an upstream transition for that day, regardless of how many raw detections occurred at that antenna site that day. Vice versa for downstream transition.
- If a fish resides in a state downstream of a stationary antenna site and is detected on both the upstream and downstream antenna of that site in a day but ends the day on the downstream-most antenna of that site, the fish is said to have made a upstream then downstream transition that day (ie GH, IJ, KL), regardless of the total number of raw detections at that stationary site that day. Vice Versa for if a fish starts upstream of the site.
- If a fish resides in a state downstream of a stationary antenna site and is detected on the downstream-most stationary antenna at that site multiple times that day without ever getting detected on the upstream antenna, it is assumed to have made no transition that day. Vice versa for if the fish starts upstream of a stationary antenna site.
- If a fish resides in a state downstream of a stationary antenna site and is detected on the downstream-most stationary antenna at that site only once that day without ever getting detected on the upstream antenna and is later detected upstream of the stationary site, it is assumed to have continued upstream and missed detection on the upstream-most antenna at that stationary site, and is coded as a upstream transition (G, I, K) for that day. Vice versa for if the fish started upstream.
- A fish can have multiple states in a day if it is detected at multiple antenna sites in a day. This also reveals "problem" fish because some fish are have said to make trips up through the dam multiple times, sometimes in a day.

States Dataframe

Here is part of the history of fish 230000228709, a sizable brown trout.

Show 10 entries Search:

Date	Datetime	TAG	State	det_type	ReleaseSite	Species	Release_Length	Release_Weight	c
							All	All	
2020-10-06	2020-10-06T13:58:00Z	230000228709	C	Release	Hitching Post	LOC	360	505	
2020-10-07	2020-10-07T18:00:28Z	230000228709	J	Hitching Post Stationary Antenna	Hitching Post	LOC	360	505	
2020-10-08	2020-10-08T06:51:01Z	230000228709	I	Hitching Post Stationary Antenna	Hitching Post	LOC	360	505	
2020-10-10	2020-10-10T23:02:31Z	230000228709	J	Hitching Post Stationary Antenna	Hitching Post	LOC	360	505	
2020-10-11	2020-10-11T02:47:46Z	230000228709	I	Hitching Post Stationary Antenna	Hitching Post	LOC	360	505	
2020-10-13	2020-10-13T18:37:49Z	230000228709	J	Hitching Post Stationary Antenna	Hitching Post	LOC	360	505	
2020-10-14	2020-10-14T02:23:45Z	230000228709	I	Hitching Post Stationary Antenna	Hitching Post	LOC	360	505	
2020-10-15	2020-10-15T00:58:31Z	230000228709	JI	Hitching Post Stationary Antenna	Hitching Post	LOC	360	505	

- The States filter in the sidebar can help determine “magic” fish that may have succumbed to avian predation, since they hit multiple weird states in a day. For example, 230000228444 has one day where it transitions at confluence, then red barn, then confluence again. All in 1 day.
- Det_type column is useful when there is only 1 state of the day, but doesn’t accurately capture everything if there are multiple states on a day
- Datetime column doesn’t really mean anything in this Dataframe either
- C_number_of_detections is the total number of raw detections for the day. Might be 22k, might be 1.
- Daily_unique_events is the total number of events that happened in a day. This can also be helpful in finding “magic” fish.

States and Days Wide

This is the same data displayed in States_dataframe, but in wide format where each unique tag gets a row and each column is a day since the beginning of the study. This may have a bit more rows than the release file because there are some fish in there without release info.

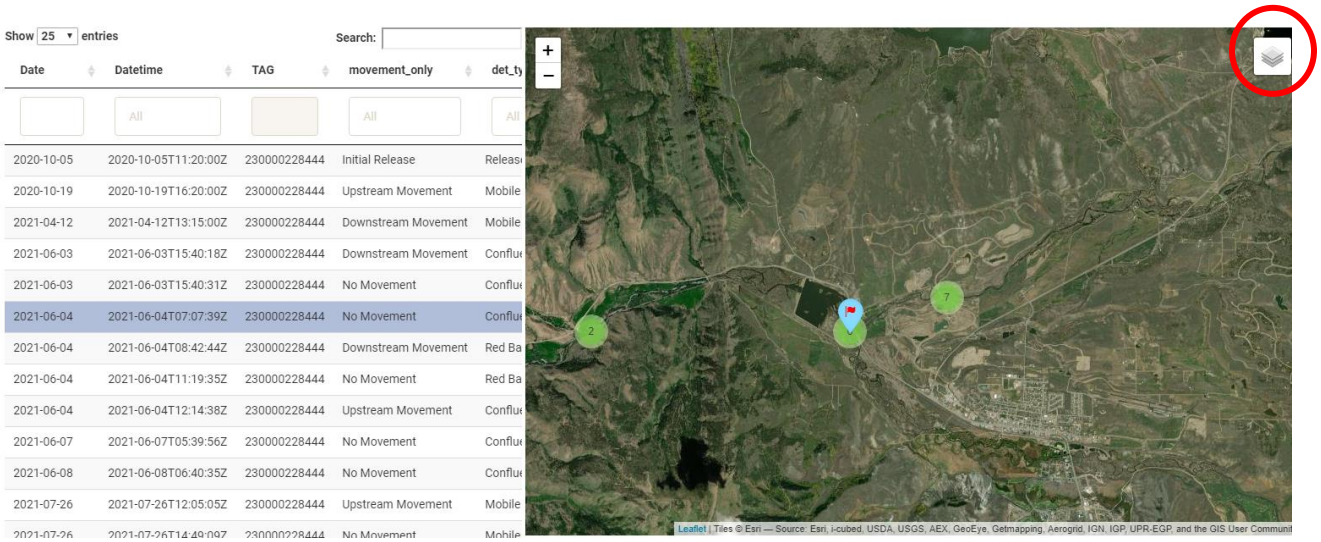
Unknown States

This is where events that were unable to be captured by states are displayed. This hopefully should be pretty small...it’s filled with tags with detections before official 'Release' such as in in May 2021 and tags without release info. Mainly tags without an idea where they came from. But if a Tag shows up where release info is known, might have to go into the get_states_function.R code to make another case_when entry to account for the new state. All in all, this is a check to see how well the get_states_function is working, and also to see if fish without release info have gotten by the other filters.

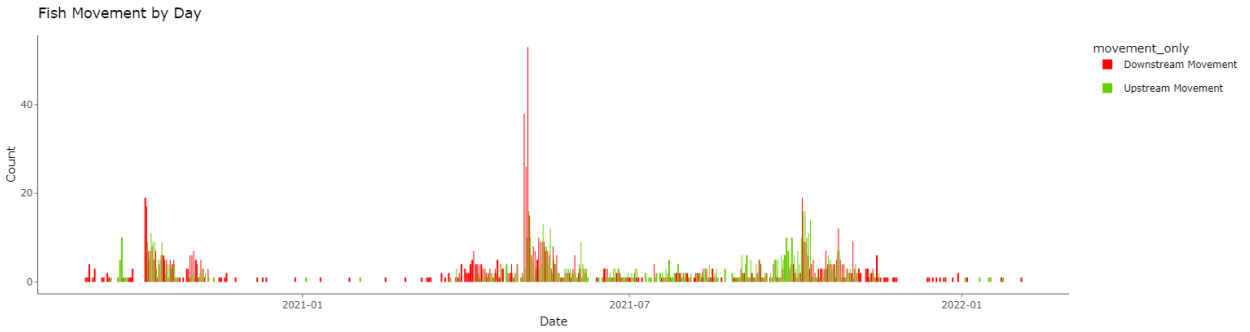
Daily Movements Map, Plot, and Data

Displays Dataframe of “movements” (Movement_table_no_trans from Get_movements_function), where a fish has changed position along the river.

- Map is interactive, where you can click and hover to get more info about a movement. Clicking on a event in the map will navigate to the event column in the table, and vice versa.
- Filters in the sidebar control what is displayed on the map
- Layer control is available in top right



- The plot is also controlled by the sidebar filters.



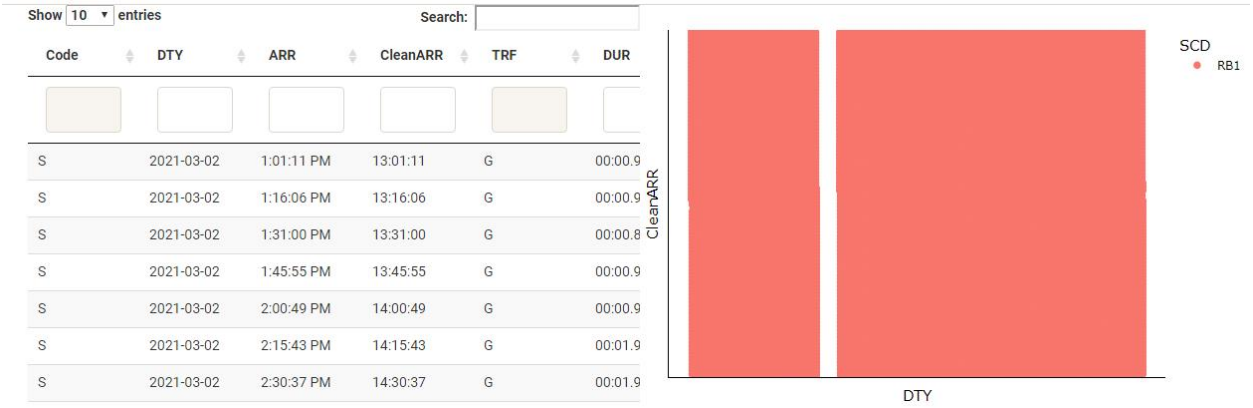
This tab can be used to answer questions like “what time of year are fish moving?” and is the easiest/funnest way to examine individual tag histories. A method I frequently use is to find fish that have moved a long distance or have weird daily states recorded, copy and paste the tag, and plug it into this tab. Or find a event on the map, click on it, copy the tag number, and plug that in.

QAQC Tab

Used to help verify data.

- Marker Tags:
  - Marker Tag Data set from All_Combined_events_function
  - Shows if there was a gap in the marker tags

- In this case, there was a gap: closer examination shows there were not detections for the month of January 2021



  - This is a large dataset to plot with a scatterplot, so it will run quickest when plotting a smaller date range with only one marker tag or site selected.
  - There is currently no functionality for Biomark marker tags but wouldn't be too hard to get it in the future
  - If the ARR column is brought in incorrectly (which can sometimes happen if the stationary ARR column gets corrupted) then there will be an abundance of detections in the 12:00-13:00 range.
- Release and Recap Lengths/Weights
  - Plot to help ensure that there were no length/weight typos in the release and recapture files.
- Unknown Tags
  - List of Tags without release info but started with 900_ initially and have detections on some sort of antenna
  - Display of Enc_hist_wide_summary_function output "Unknown Tags"

General Notes

- All plots are interactive. Hover over plot for more info and toggle what is displayed by clicking and double clicking items in the legend. Zoom in by clicking and dragging within the plot.
- Datatables are also interactive. Click the column titles of any column to sort in ascending/descending order. You can use the Search bar on the right to search/filter for any specific values. Some of the individual filters work, some don't, depending on column type I believe.
- The Tag filter has to use the exact tag number with no spaces at the end.
- It's important when adding a new updated file to the app that it had the exact same column names and order as the older
- UTM's for stationary and biomark stations are assigned within the app
- Keep a folder of CSV backups in case the time column goes wonky
- Currently there is a Release site called "Shefiff Ranch Middle Field"; if this typo is corrected in Release File, it will have to be changed in the get_states_function code as well. Otherwise all fish will show up in the "unknown states" tab.

Let me know if there are other questions and concerns, I’m happy to add to this if needed. Let me know
sfigraf@gmail.com