BTO Visualisation Tool Documentation

Summary

The BTO Visualisation tool (btovis.github.io) is a front-end web application that processes CSV files exported from the BTO pipeline to display useful plots and graphs.

Usage Instructions

To begin, the user can drag and drop a CSV from the BTO pipeline into the web page, or click “Browse Files” on the left. As a prior warning, extremely large CSVs (with millions of rows) may overwhelm the browser.

This will generate a default panel with **preset filters** and default widget.

Behavioural Quirks

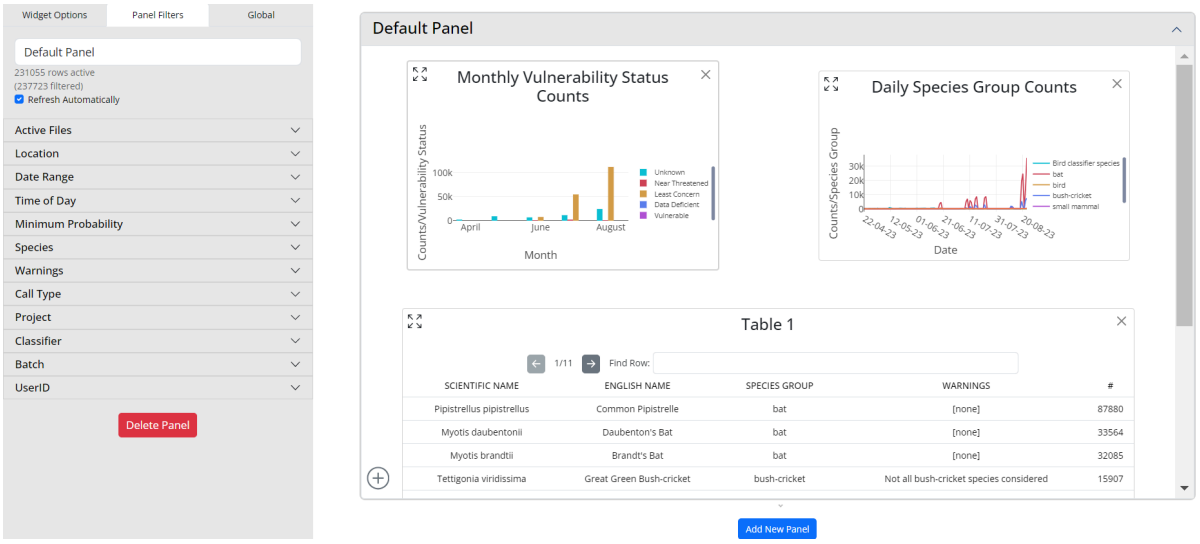
This is a list of behavioural quirks that may result in odd data views. The user should be aware of these.

Vulnerability status was embedded in the application on Feb 20, 2024 and does not reflect new changes.

Additionally, bird CSVs (e.g. NJ3353_Cairnty.csv) have missing columns (i.e. no location or scientific name), so they have reduced functionality, like no vulnerability status, and no location sorting. **Additionally, bird CSVs will default to their english name for their scientific name** to preserve functionality.

By default, the entire application will display and filter by the **Survey Date** column and not the Actual Date column. You can display by Survey Date for each individual widget, and choose to filter by Actual Date with the selector in the Date Range filter.

Default Panel



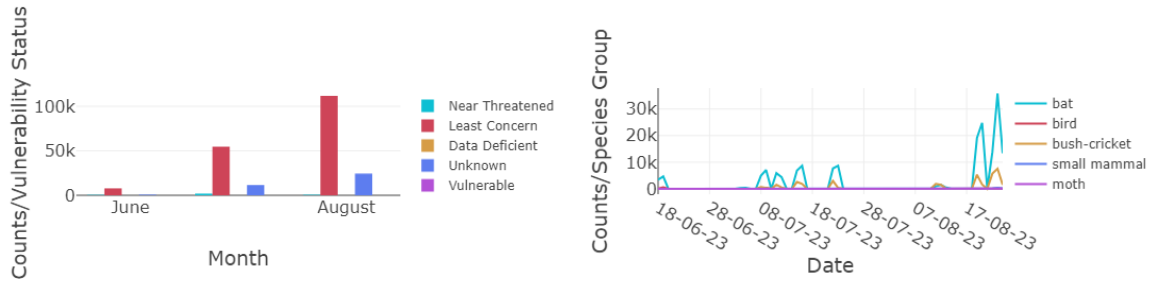
An example of a default panel

The preset filters applied will:

- Floor Minimum Probability at 50%
- Ignore rows where the scientific name is empty

The filters can be edited by going to the sidebar and adjusting them as needed.

The default panel contains the following widgets:



2 charts, vulnerability status against month, and species group against day

Table 1

1/7 Find Row:

SCIENTIFIC NAME	ENGLISH NAME	SPECIES GROUP	WARNINGS	#
Pipistrellus pipistrellus	Common Pipistrelle	bat	[none]	87880
Myotis daubentonii	Daubenton's Bat	bat	[none]	33564
Myotis brandtii	Brandt's Bat	bat	[none]	32085
Tettigonia viridissima	Great Green Bush-cricket	bush-cricket	Not all bush-cricket species considered	15907
Pholidoptera griseoaptera	Dark Bush-cricket	bush-cricket	Not all bush-cricket species considered	9126
Nyctalus leisleri	Leisler's Bat	bat	[none]	5755
Roeseliana roeselii	Roesel's Bush-cricket	bush-cricket	Not all bush-cricket species considered	4528
Nyctalus noctula	Noctule	bat	[none]	3588

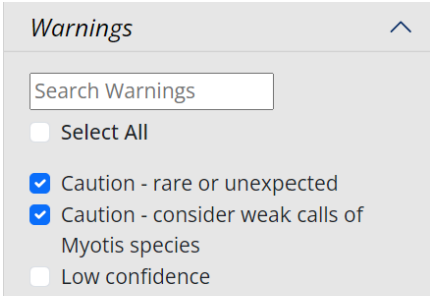
A table showing the most common warnings (including rows with no warnings)

A map showing where the data was recorded

Panel Filters

Filters can be adjusted on the left for each panel. Widgets inside a panel will apply its filters. For example, to ignore rows with “Low confidence” in warnings,

1. Click on the panel you want to apply the filter on
2. Open the “Warnings” tab on the sidebar
3. Uncheck “Low confidence”



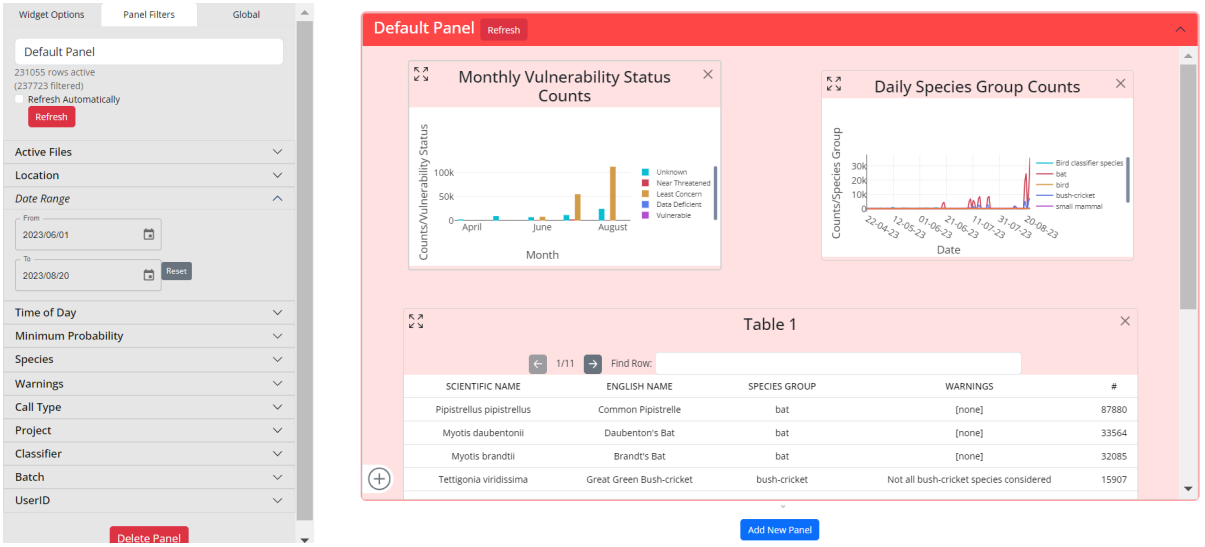
An example of the Warnings Filter with “low confidence” ignored

Most filters will look like this. The rest of the section describes the other filters.

Automatic Refresh

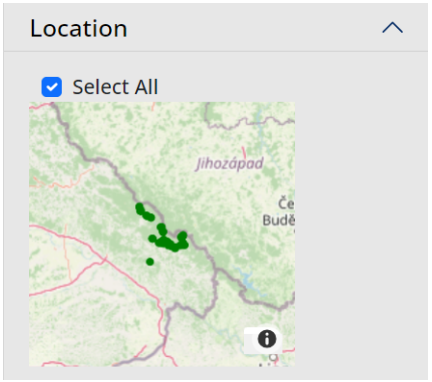
By default, the application will apply filters immediately, and render the updated graphs. For very large CSVs, this may cause frequent lag spikes. In that case, you may want to uncheck Automatic Refresh.

This provides access to a manual refresh button, allowing you to select all the relevant filters before triggering a refresh. Widget options do not apply to this feature.



Example of a panel without applying the current filters. Press “Refresh” on the sidebar or on the panel header to begin the graph rendering

Location

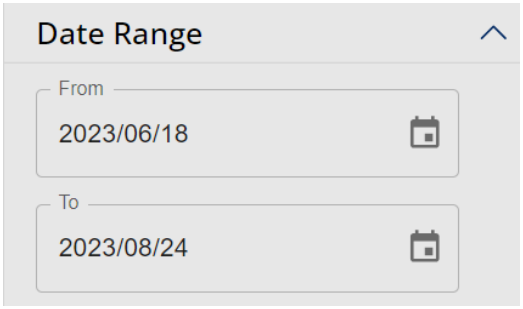


An image of the Location filter

You can click and drag on the location filter to choose the box of locations to include. When hovering over the map, zoom, pan and box select options will appear. Use these options as needed to move around the map.

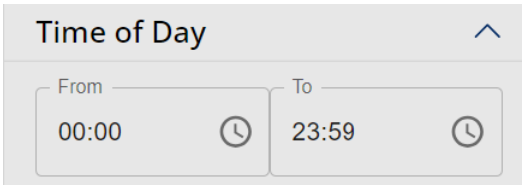
Green points are currently included in the panel’s data, while red points are filtered away.

Date Range



This refers to the inclusive range of dates. Click on the calendar, or type a date to use it

Time of Day

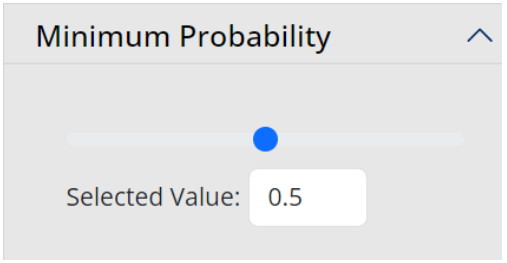


This filter is meant to select the time range for each day.

For example, a time range of 05:00-23:59 **excludes** any data that occurs from 00:00-04:59.

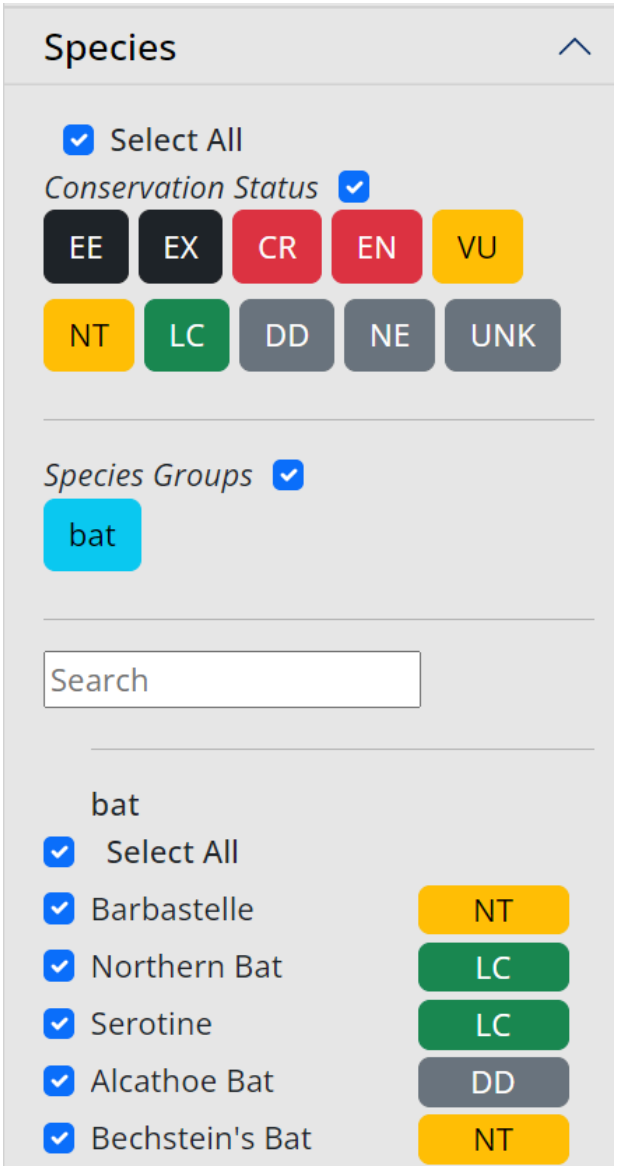
On the other hand, an inverted range like 07:00-00:00 will **exclude** data from 00:00-07:00, while including data from every other time.

Minimum Probability



Excludes any row with a probability below the set value.

Species



This selector will operate on empty species rows (i.e. scientific name is empty)

You can ignore specific conservation statuses or species groups by clicking on them.

Additionally, specific species can be selected in the list of checkboxes below.

The “Invert” button next to each species group will flip the checkboxes under the group.

The species filter can be quite heavy, so it will apply itself after about half a second of no activity to facilitate rapid consecutive clicks. Expect a very brief latency.

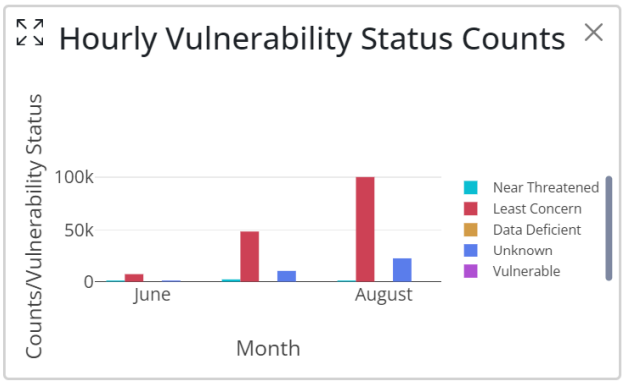
Widgets

At the bottom of each panel, there is a “+” icon that lets you add a new widget. You can delete widgets by pressing the “X” icon on the top right of each widget.

Each widget has its own specific settings that you can access by clicking on a created widget, and then looking at the “Widget Options” tab on the left.

You can fullscreen a widget by clicking the fullscreen icon on the top left of each widget. Charts can be exported as images when you hover over them and press the camera icon.

Bar Chart



Bar charts will count the number of groups per unit. The graph above counts the vulnerability status occurrences per month.

Both charts will use Survey Date by default. This can be changed in the charts’ individual widget options.

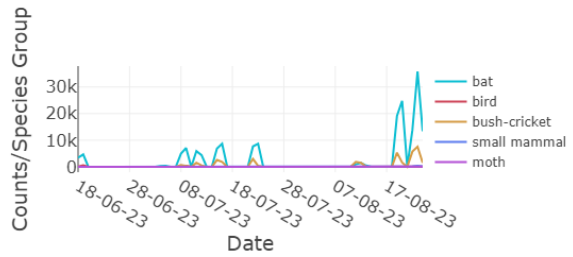
Bar Chart X-Axis Units

- Hour - Time of Day to group by (Not linear time)
- Date - Groups by date (24 hour time slices)
- Month - Groups by month (month time slices)
- Year - Groups by year (year time slices)
- Project Name - Groups by the name of the project
- Batch Name - Groups by the Batch Name column
- Filename - Groups according to the uploaded file name

Bar Chart Y-Axis Groups

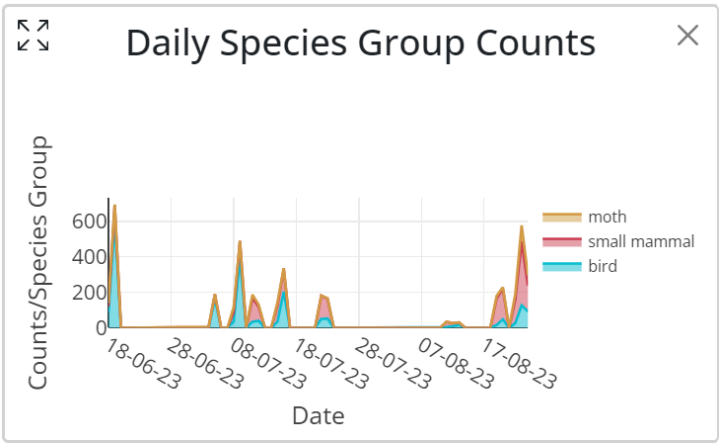
- Animal - English name
- Species - Scientific name
- Species Group - Species group
- Vulnerability Status

Line Chart



Line charts are very similar to bar charts, except that the X-Axis can only be time groups.

Additionally, you can turn the line chart into a stacked line chart if needed:



Both charts will use Survey Date by default. This can be changed in the charts' individual widget options.

Line Chart X-Axis Units

- Date - Groups by date (24 hour time slices)
- Month - Groups by month (month time slices)
- Year - Groups by year (year time slices)

Line Chart Y-Axis Groups

- Animal - English name
- Species - Scientific name
- Species Group - Species group
- Vulnerability Status

Table

Table 1

1/7 Find Row:

SCIENTIFIC NAME	ENGLISH NAME	SPECIES GROUP	WARNINGS	#
Pipistrellus pipistrellus	Common Pipistrelle	bat	[none]	87880
Myotis daubentonii	Daubenton's Bat	bat	[none]	33564
Myotis brandtii	Brandt's Bat	bat	[none]	32085
Tettigonia viridissima	Great Green Bush-cricket	bush-cricket	Not all bush-cricket species considered	15907
Pholidoptera griseoaptera	Dark Bush-cricket	bush-cricket	Not all bush-cricket species considered	9126
Nyctalus leisleri	Leisler's Bat	bat	[none]	5755
Roeseliana roeselii	Roesel's Bush-cricket	bush-cricket	Not all bush-cricket species considered	4528
Nyctalus noctula	Noctule	bat	[none]	3588

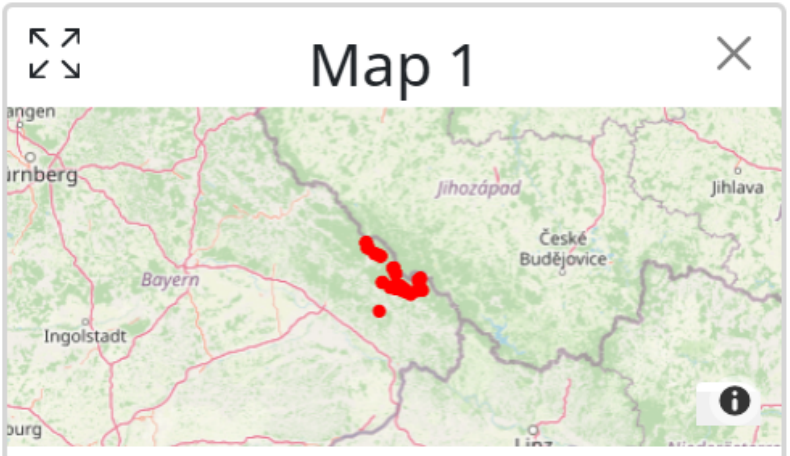
The table will account for the columns selected in its widget options, then sort the rows by how frequent they are.

In the above example, the Common Pipistrelle row **with no warnings** is the most common row, so it appears at the top.

Rows with “[none]” in them can be ignored with the “Cull Empty Cells” option.

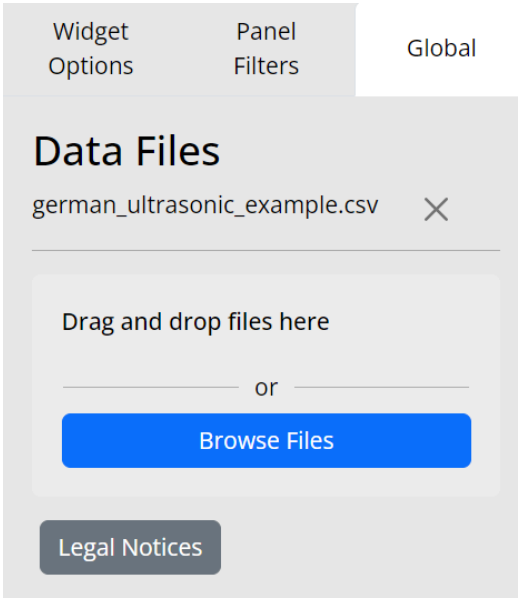
Tables cannot be exported as images.

Map



Shows points on a map where species groups appear. Each point will show which species groups appear, though in testing, this doesn’t tend to be different for each point.

Global Sidebar



More files can be added by dragging and dropping them as per normal, or via the browse files button.

Files can be removed by pressing the X.

“Legal Notices” contains the licences for all the code that was used.

Technical Documentation

The app is built using React and Vite as the build tool. Btovis.github.io is currently hosted on Github pages.

When the user uploads the CSV, the data object processes it using addCSV. A default panel will be created if the visit is new, if not users can add a new panel. Each panel can control its own filters, and widgets inside each panel will display filtered data. Each panel accesses data through DataFilterer. There is a single Data class created once per application run. It is updated through addCSV and removeCSV.

The panel filters on the sidebar are called input options. They are unique to each panel, so changing these for one panel doesn’t impact other panels. They have a getQuery method that describes how data should be filtered for that panel. Queries from an option overwrite the previous from the same option. Data is recalculated every time one of the filters changes. There’s no caching between filters, so DataFilterer filters the original data with every filter every time when any filter changes.

Adding a CSV will involve reordering of the columns of the new CSV to match the previous table. New columns will be added as the last columns.

Scalability

Javascript arrays are limited to at most about 4.2 billion entries. This means that the database is hard limited at this number of rows. There is another problem making it impossible to reach this limit. To allow faster filtering and lower memory usage, considering that Javascript strings are primitive values copied whenever assigned, a class, SetElement, is used for wrapping strings in objects, so that strings aren't repeated. These are contained in a set and in a map, one of each for each column that uses strings. Recording file name doesn't use this because these are unique for each row. User ID and species do use this because these have lots of repetitions. Since all, for example latin species names, are stored in a set and a map, the map max size limit applies. On Safari and Firefox, these structures are limited to 67.1 million entries and on Chrome, 16.7 million entries. Therefore it isn't possible to have more than this number of scientific species names, or more than this number of user IDs, or more than this number of files etc. loaded in the database at any time, but deleting the previous files before uploading them would work. In real world terms, if a CSV file has one unique species for each group of 4 rows (a ratio), one can use a CSV with up to 4*67.1=~268.4 million rows.

If one CSV is larger than 4 gigabytes, it'll likely be necessary to partition it to smaller CSVs and only then upload them. Dragging and dropping all of them at once or choosing multiple CSVs from the file picker at once will likely make the process slightly faster.