



sgoldenlab /
simba



Code

Issues 61

Pull requests 1

Actions

Projects

Security



simba / docs / tutorial.md



sronilsson 6 months ago



547 lines (313 loc) · 44.1 KB

Preview

Code

Blame

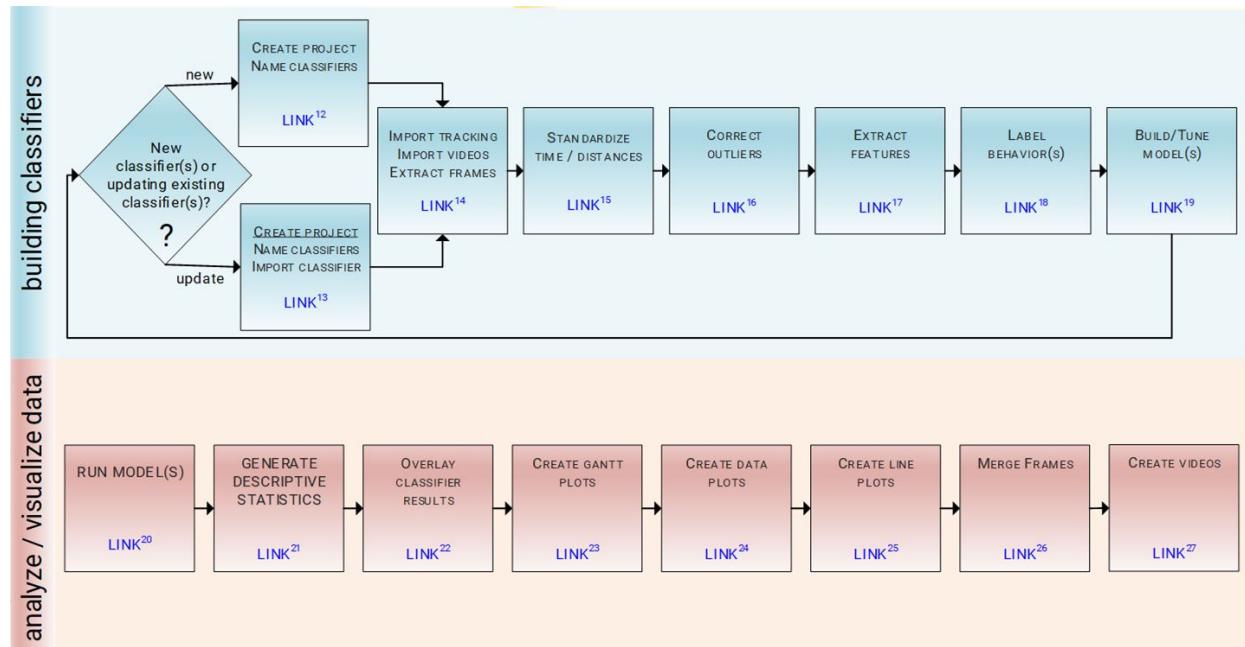


🔗 Generic SimBA Tutorial

This is a generic step by step tutorial to start using SimBA to create behavioral classifiers. For more detailed, up-to-date, information about the menus and options and their use cases, see the [SimBA Scenario tutorials](#)

Pipeline breakdown

The analysis pipeline is split into sections. These sections are indexed below along with their corresponding functions:



Part 1: Create a new project

- Generate project config (create new classifiers or import existing classifiers)

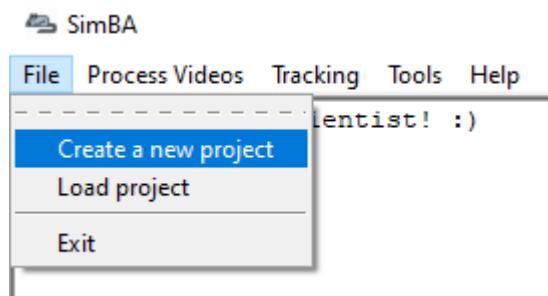
- Import videos into project folder
- Import DLC Tracking Data (if have any)
- Extract Frames into project folder

Part 2: Load project

- Load the project.ini
- Set video parameters
- Outlier correction
- Extract Features
- Label Behavior
- Train Machine Model
- Run Machine Model
- Analyze Machine Results
- Visualization
- Plot Graphs
- Merge Frames
- Create Video

Part 1: Create a new project

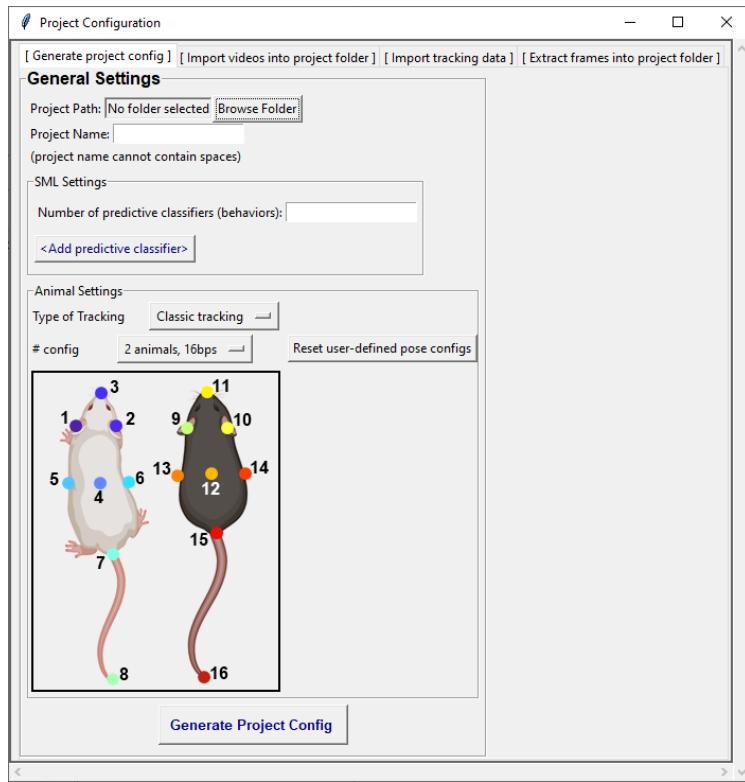
This section describes how to create a new project in SimBA.



Step 1: Generate Project Config

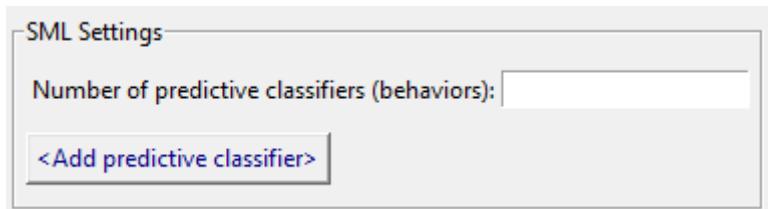
In this step we will create a main project folder with all the required sub-directories.

1. In the main SimBA window, click on `File` and then `Create a new project`. The following windows will pop up.



2. Navigate to the [Generate project config] tab. Under **General Settings**, specify a **Project Path** which is the directory which will contain your main project folder.
3. Specify a **Project Name**. This is the name of your project. *Keep recommend not to use spaces in your project name. Instead use underscore "_"*
4. Under **SML Settings**, specify the number of classifiers that you wish to create. For an example, if you have three behaviors that you are interested in, put **3** in the entry box. *Note: If you are using SimBA only for region of interest (ROI) analysis, and do not wish to create any classifiers, enter **1** in this entry box*

5. Click **<Add predictive classifier>** and it creates a row as shown in the below image. In each entry box, fill in the name of the behavior that you want to classify. *Note: If you are using SimBA only for region of interest (ROI) analysis, and do not wish to create any classifiers, enter any name in the the single entry box.*

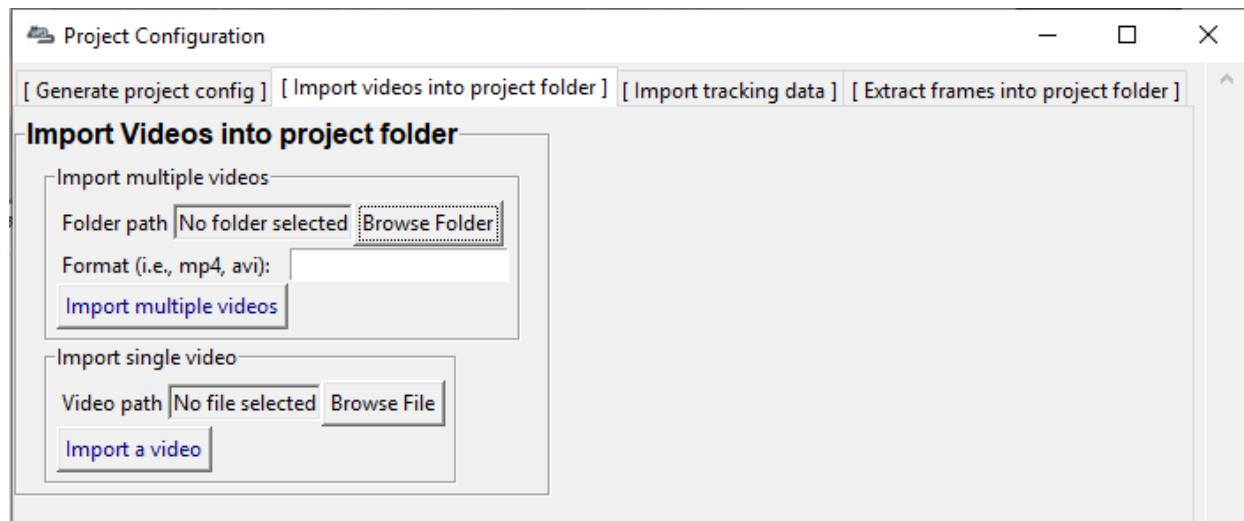


6. Type of Tracking allows the user to choose multi-animal tracking or the classic tracking.

7. Use the **Animal Settings** dropdown to specify the number of animals and body parts that that the pose estimation tracking data contains. Select your pose-estimation body-part configuration from the appropriate drop-down menu. If you can't see your species and/or body-part configuration, please create your own *User defined body-part* configuration*. Click [HERE](#) for documentation on how to create user-defined body-part configurations.
8. Click on **Generate Project Config** to create your project. The project folder will be located in the specified **Project Path**.

Step 2: Import videos into project folder

Next, you can choose to import either one or multiple videos. The imported videos are used for visualizing predictions and standardizing distances across videos by calculating metric distances from pixel distances.



To import multiple videos

1. Navigate to the **[Import videos into project folder]** tab.
2. Under the **Import multiple videos** heading, click on **Browse Folder** to select a folder that contains all the videos that you wish to import into your project.
3. Enter the file type of your videos. (e.g., *mp4*, *avi*, *mov*, etc) in the **Video type** entry box.
4. Click on **Import multiple videos**.

Note: If you have a lot of videos, it might take a few minutes before all the videos are imported.

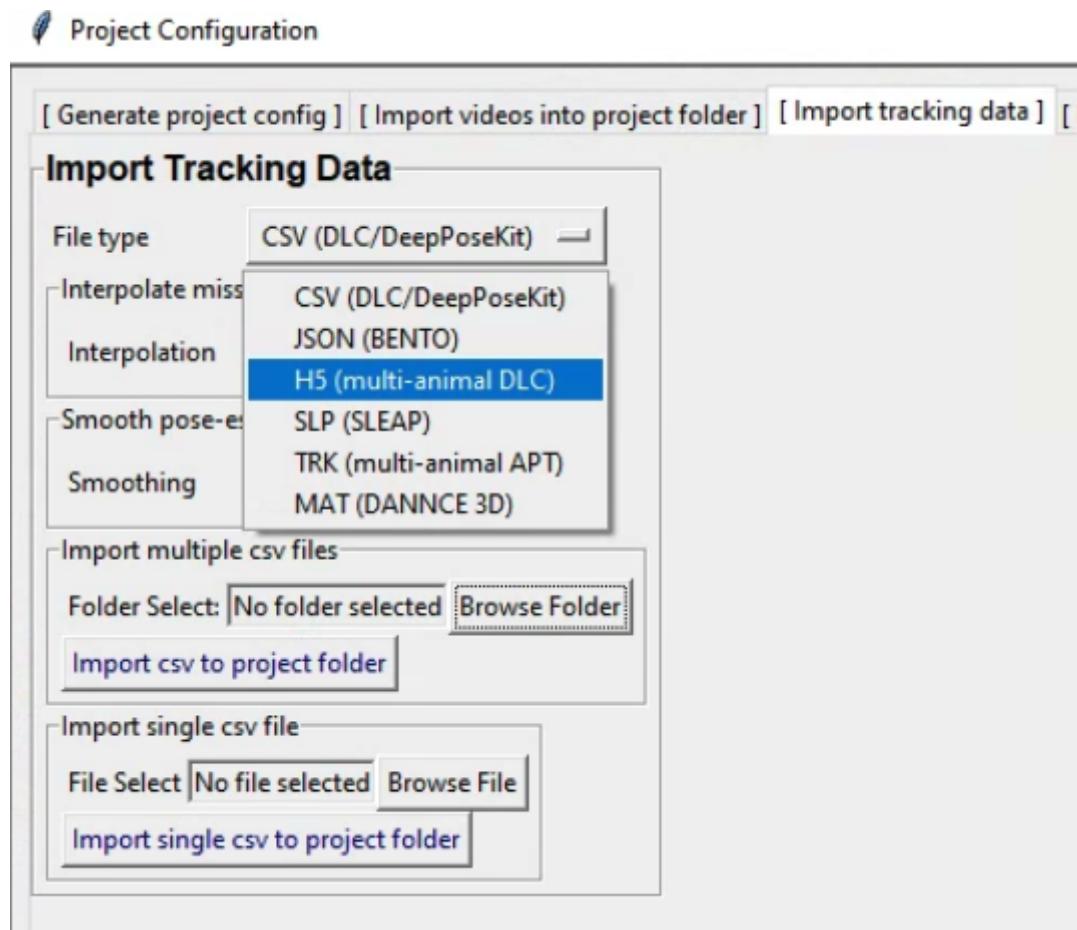
To import a single video

1. Under the **Import single video** heading, click on **Browse File** to select your video.

2. Click on Import a video to select the path of your video.

Step 3: Import Tracking Data

Next, we will import your pose-estimation tracking data.



To import csv/json/h5/trk/deepposekit pose-estimation tracking files

1. Navigate to the [Import tracking data] tab. Under the Import tracking data click on the File type drop down menu.
2. From the drop down menu, select the type of pose-estimation data you are importing into SimBA.
3. To import multiple files, choose the folder that contains the files by clicking Browse Folder , then click Import csv to project folder .
4. To import a single file, choose the file by clicking Browse File , then click Import single csv to project folder .

Note: Below the File type drop-down menu, there are two option menus that can help correct and improve the incoming pose-estimation data (*interpolation*, and *smoothing*). Both of these menus default to *None* , meaning no corrections will be performed on the incoming data. If you are interested in removing missing values (interpolation), and/or correcting "jitter" (smoothing), please see sections 2 and 3 in [TUTORIAL SCENARIO 1: IMPORT TRACKING DATA](#) which details these functions, and including video examples of expected results.

Below follows short examples for how to import pose-estimation from two of the more popular poset-estimation packages (multi-animal DLC and SLEAP).

To import h5 files (multi-animal DLC)

Please note that you can only import the h5 tracking data after you have imported the videos into the project folder.

1. From the `File type` drop down menu, select `H5 (multi-animal DLC)`.
2. Under `Animal settings`, enter the number of animals in the videos in the `No of animals` entry box, and click `Confirm`.
3. Enter the names for each of the animal in the video.
4. `Tracking type` is the type of tracking from DeepLabCut multi animal tracking.
5. Select the folder that contains all the h5 files by clicking `Browse Folder`.
6. Click `Import h5` to start importing your pose-estimation data.

To import SLP files (SLEAP)

1. From the `File type` drop down menu, select `SLP (SLEAP)`.
2. Under `Animal settings`, enter the number of animals in the videos in the `No of animals` entry box, and click `Confirm`.
3. Enter the names for each of the animal in the video.
4. Select the folder that contains all the slp files by clicking `Browse Folder`.
5. Click on `Import .slp` to start importing your pose-estimation data.

You have now successfully created your SimBA project and imported your data. You should see your videos inside the `project_folder/videos` directory, and your pose estimation data (one file for each of the videos in the project) inside the `project_folder/csv/input_csv` directory.

Note: An earlier version of this documentation suggested that users had to extract frames from each of the imported videos. **Extracting frames is not necessary.**

Now it's time to load the project, create and/or use some classifiers, and analyze some data.

Part 2: Load project

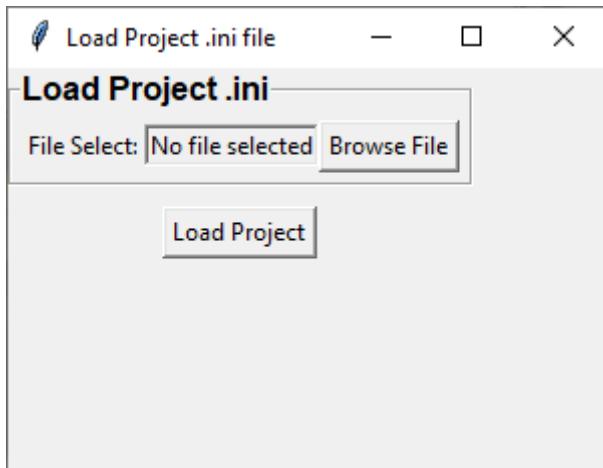
This section describes how to load and work with created projects.

Step 1: Load Project Config

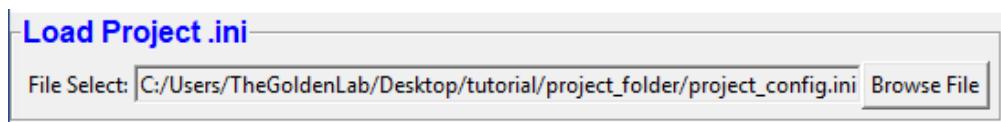
In this step you will load the `project_config.ini` file that was created.

Note: A `project_config.ini` should always be loaded before any other process.

1. In the main SimBA window, click on File and Load project . The following windows will pop up.



2. Click on the Browse File button. Then navigate to the directory that you created your project in and click on your *project folder*. Locate the *project_config.ini* file and select it. Once this step is completed, it should look like the following, and you should no longer see the text *No file selected*.



In this image, you can see the Desktop is my selected working directory, tutorial is my project name, and the last two sections of the folder path is always going to be project_folder/project_config.ini .

3. Click on the Load Project button.

Step 2 (Optional) : Import further pose-estimation tracking data and videos

Once we have loaded our project, we can choose to import more pose estimation data in and/or more videos. If this isn't relevant then you can skip this Step 2.

<p>Import further DLC tracking data</p> <p>Import multiple csv files <input type="button" value="Folder selected: No folder selected"/> <input type="button" value="Browse Folder"/> <input type="button" value="Import csv to project folder"/></p> <p>Import single csv files <input type="button" value="File selected"/> <input type="button" value="No file selected"/> <input type="button" value="Browse File"/> <input type="button" value="Import single csv to project folder"/></p>	<p>Extract further frames into project folder</p> <p><input type="button" value="Extract frames"/></p>
<p>Import further videos into project folder</p> <p>Import multiple videos <input type="button" value="Folder path"/> <input type="button" value="No folder selected"/> <input type="button" value="Browse Folder"/> <input type="button" value="File format (i.e., mp4/avi):"/> <input type="button" value="Import multiple videos"/></p> <p>Import single video <input type="button" value="Video Path"/> <input type="button" value="No file selected"/> <input type="button" value="Browse File"/> <input type="button" value="Import a video"/></p>	<p>Import frame folders</p> <p>Main frame directory <input type="button" value="No folder selected"/> <input type="button" value="Browse Folder"/> <input type="button" value="Import frames"/></p>

1. Click on the [Further imports (data/video/frames)] tab. From here (at the bottom left of the window) you can import more data or videos into the project folder. The imported files will be placed in the `project_folder/csv/input_csv` directory, and the imported videos will be placed in the `project_folder/videos` directory.

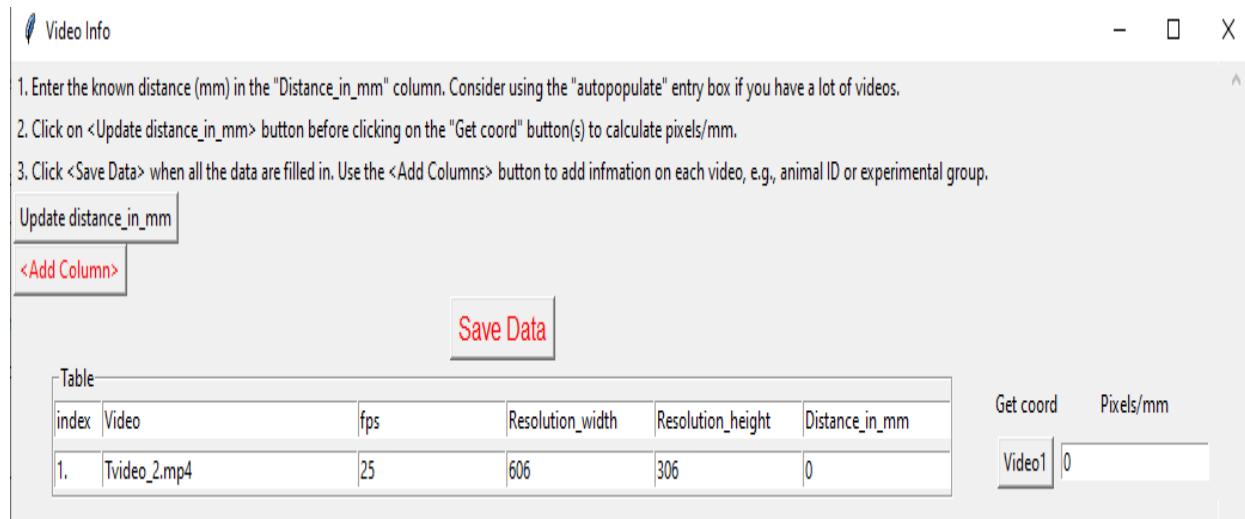
Step 3: Set video parameters

In this step, we will specify the meta parameters for each of your videos (fps, resolution, metric distances). We will also set the **pixels per millimeter** for your videos. You will be using a tool that requires the known distance between two points (e.g., the cage width or the size of an object in the image) in order to calculate **pixels per millimeter**. The real life distance between the two points is called **Distance in mm**. This is important for standardizing measurements across different videos where the camera might have moved in-between recordings.

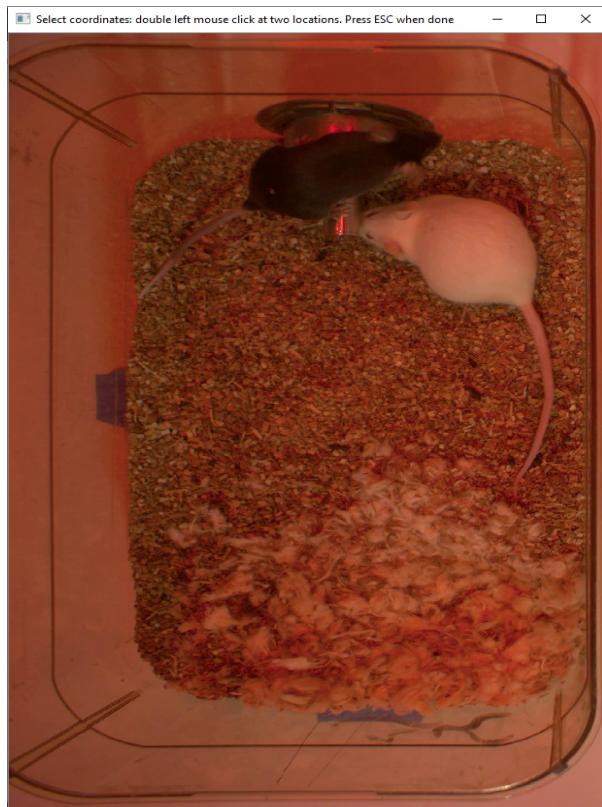
Video parameters (fps, resolution, ppx/mm, etc.)	
Known distance (mm)	<input type="text"/> <input type="button" value="Autopopulate table"/>
<input type="button" value="Set video parameters"/>	

1. Under Set video parameters(distances,resolution,etc.), the entry box named Distance in mm is the known distance between two points in the videos in millimeter. If the known distance is the same in all the videos in the project, then enter the value (e.g.: 245) and click on Auto populate Distance in mm in tables . and it will auto-populate the table in the next step (see below). If you leave the Distance in mm entry box empty, the known distance will default to zero and we will be forced to specify the the value for each video, individually.

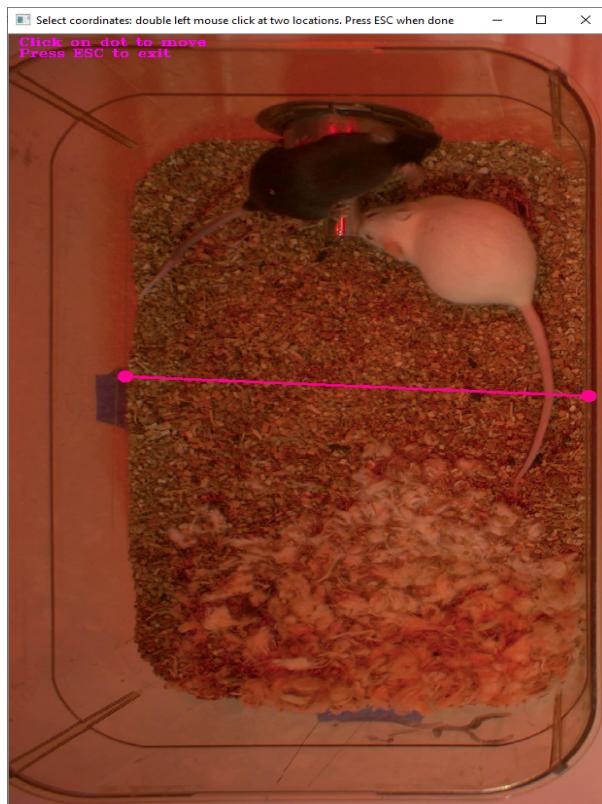
2. Click on Set Video Parameters and the following windows will pop up.



3. In the above example we imported four videos, and their names are listed the left-most Video column. We auto-populated the known distance to 10 millimeter in the previous step, and this value is now displayed in the Distance in mm column.
4. We can click on the values in the entry boxes and change them manually until satisfied. Then, we click on Update distance_in_mm and this will update the whole table.
5. Next, to get the Pixels/mm for the first video, click on Video1 , and the following window will pop up. The window that pops up displays the first frame of the Video1 video.



6. Now, double left mouse click to select two points in the image that defines the known distance in real life. In this case, I know that the two **pink connected dots** represent a distance of 10 millimeter in real life.



7. If you misplaced one, or both of the dots, you can double click on either of the dots to place them somewhere else in the image. Once you are done, hit `Esc` key on your keyboard.



8. If every step is done correctly, the `Pixels/mm` column in the table should populate with the number of pixels that represent one millimeter in your video.

Video Info

1.Enter the known distance (mm)
2.Click <Update distance_in_mm> button before clicking on Video1,Video2,Video3... to get the pixels/mm
3.Click <Save Data> once all the data are filled.

-Table

index	Video	fps	Resolution_width	Resolution_height	Distance_in_mm	Get_coord	Pixels/mm
1.	Tvideo_76.mp4	25.0	624.0	360.0	100	Video1	9.14289341510662
2.	Video5.mp4	70.0	1200.0	2056.0	100	Video2	1

9. Repeat the steps for every video in the table, and once it is done, click the `Save Data` button. This will generate a csv file named `video_info.csv` in `project_folder/log` directory that contains a table with your video meta data.

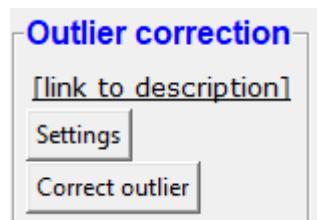
Note: If you know the ...

10. You can also chose to add further columns to the meta data file (e.g., AnimalID or experimental group) by clicking on the Add Column button. This information will be saved in additional columns to your `video_info.csv` file.

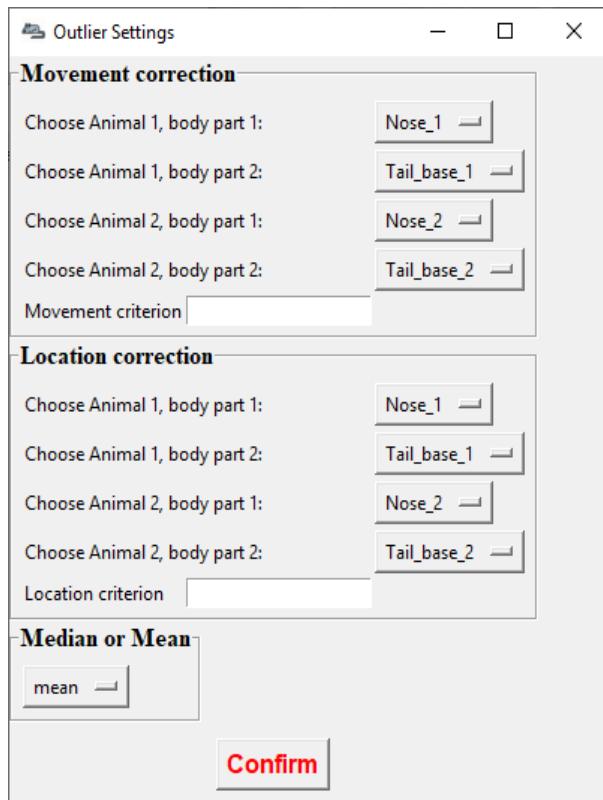
Step 4: Outlier Correction

Outlier correction is used to correct gross tracking inaccuracies by detecting outliers based on movements and locations of body parts in relation to the animal body length. For more details, please click [here](#).

To skip Outlier Correction, click the Skip outlier correction button in red font in the [Outlier correction] tab.



1. Click on `Settings` and the following window will pop up. The Outlier Settings window varies with the number of animals in the project. The images below shows settings for a project containing two animals.



2. Select the body parts for Animal 1 and Animal 2 that you want to use to calculate a reference value. The reference value will be the mean or median Euclidian distance in millimeters between the two body parts of the two animals in all frames.

3. Enter values for the Movement criterion and the Location criterion .

- Movement criterion . A body part coordinate will be flagged and corrected as a "movement outlier" if the body part moves the *reference value \times the criterion value* across two sequential frames.
- Location criterion . A body part coordinate will be flagged and correct as a "location outlier" if the distance between the body part and at least two other body parts belonging to the same animal are longer than the *reference value \times the criterion value* within a frame.

Body parts flagged as movement or location outliers will be re-placed in their last reliable coordinate.

4. Choose to calculate the median or mean Euclidian distance in millimeters between the two body parts and click on Confirm Config .

5. Click to run the outlier correction. You can follow the progress in the main SimBA window. Once complete, two new csv log files will appear in the /project_folder/log folder. These two files contain the number of body parts corrected following the two outlier correction methods for each video in the project.

Step 5: Extract Features

Based on the coordinates of body parts in each frame (as well as the frame rate and the pixels per millimeter values) the feature extraction step calculates a larger set of features used for behavioral classification. Features are values such as metric distances between body parts, angles, areas, movement, paths, and their deviations and rank in individual frames and across rolling windows. This set of features will depend on the body-parts tracked during pose-estimation (which is defined when creating the project). Click [here](#) for an example list of features when tracking 2 mice and 16 body parts. Click [here]

(https://github.com/sgoldenlab/simba/blob/master/misc/Feature_description.csv for an example list of feature categories when tracking user-defined body-parts in SimBA.

1. Click on Extract Features under the [Extract features] tab.

Step 6: Label Behavior

This step is used to label (a.k.a annotate) behaviors in each frames of a video. This data will be concatenated with the features and used for creating behavioral classifiers. If you have annotations from third-party tools you want to append, and skip labelling behavior in SimBA, [check out the third-party annotation page](#).

There are two options: (i) start anew video annotation (ii) continue on where you last left off. Both are essentially the same, except the latter will start with the frame where you last saved. For example, one day, you started a new video by clicking Select video (create new video annotation) and you feel tired of annotating the videos. You can now click Generate/Save button to save your work, for your co-worker to continue. Your co-worker can continue by clicking Select video (continue existing video annotation) and select the the video that you have annotated half-way and take it from there!

The screenshot shows a software interface for managing video annotations. At the top, there are several tabs: [Further imports (data/video/frames)], [Video parameters], [Outlier correction], [ROI], [Extract features], and [Label behavior].

Label Behavior

- [Select video (create new video annotation)]
- [Select video (continue existing video annotation)]

Pseudo Labelling

- Video Path: [No file selected] [Browse File]
- Threshold: nesting [input field]
- Correct label [button]

Import Third-Party behavior labels

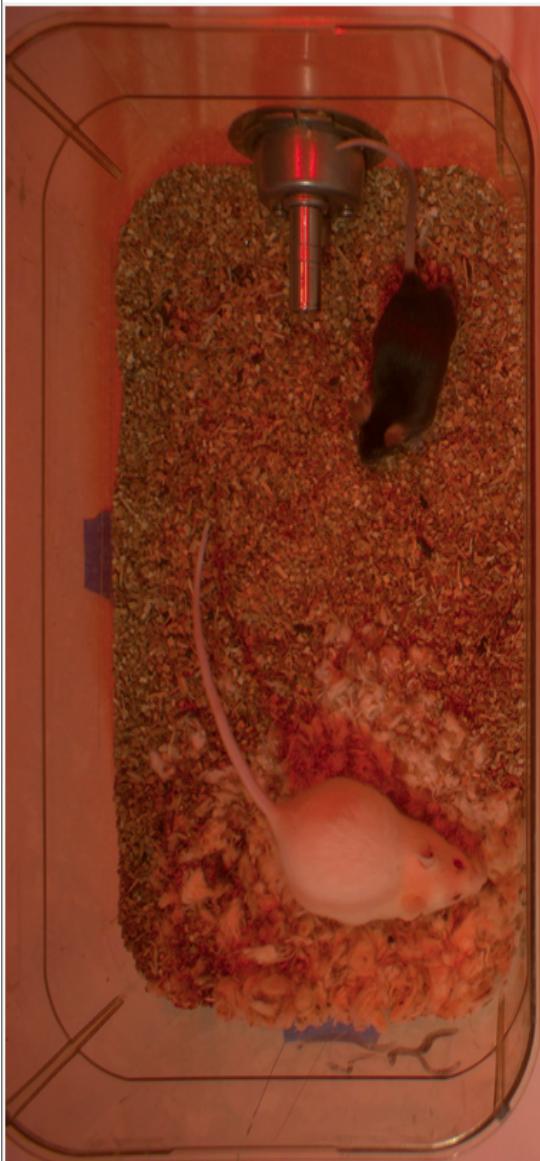
- [Import MARS Annotation (select folder with .annot files)]
- [Import Boris Annotation (select folder with .csv files)]
- [Import Solomon Annotation (select folder with .csv files)]
- [Import Ethovision Annotation (select folder with .xls/xlsx files)]

Advance Label Behavior

Note that you will have to specify behavior and non-behavior on your own.

- [Select video (create new video annotation)]
- [Select video (continue existing video annotation)]

1. Click on Select video (create new video annotation). In your project folder, navigate to the /project_folder/videos/ directory and select the video file you wish to annotate. The following window should pop up.



Open Current Video

For Video:
p = Pause/Play

After pressing pause:
o = Forward 2 Frames
e = Forward 10 Frames
w = Forward 1 Second

t = Back up 2 Frames
s = Back up 10 Frames
x = Back up 1 Second

q = Quit

Check Behaviors:

- tail_rattle_prediction
- pursuit_prediction
- anogenital_prediction
- lateral_threat_prediction
- attack_prediction

Show current video frame

Key Presses:
Right Arrow = +1 Frame
Left Arrow = -1 Frame
Ctrl + s = Save and +1 Frame
Ctrl + l = Last Frame
Ctrl + o = First Frame

Frame Number

<<	<	244	>	>>
----	---	-----	---	----

Jump to selected frame

Jump Size: 0 << >>

Save and Advance to the next frame

Frame Range to

Generate/Save csv

Please click [here](#) for more detailed information on how to use the behavior annotation interface in SimBA.

3. Once finished, click the **Generate/Save** button and a new file will be created in the *project_folder/csv/targets_inserted* directory representing your video.

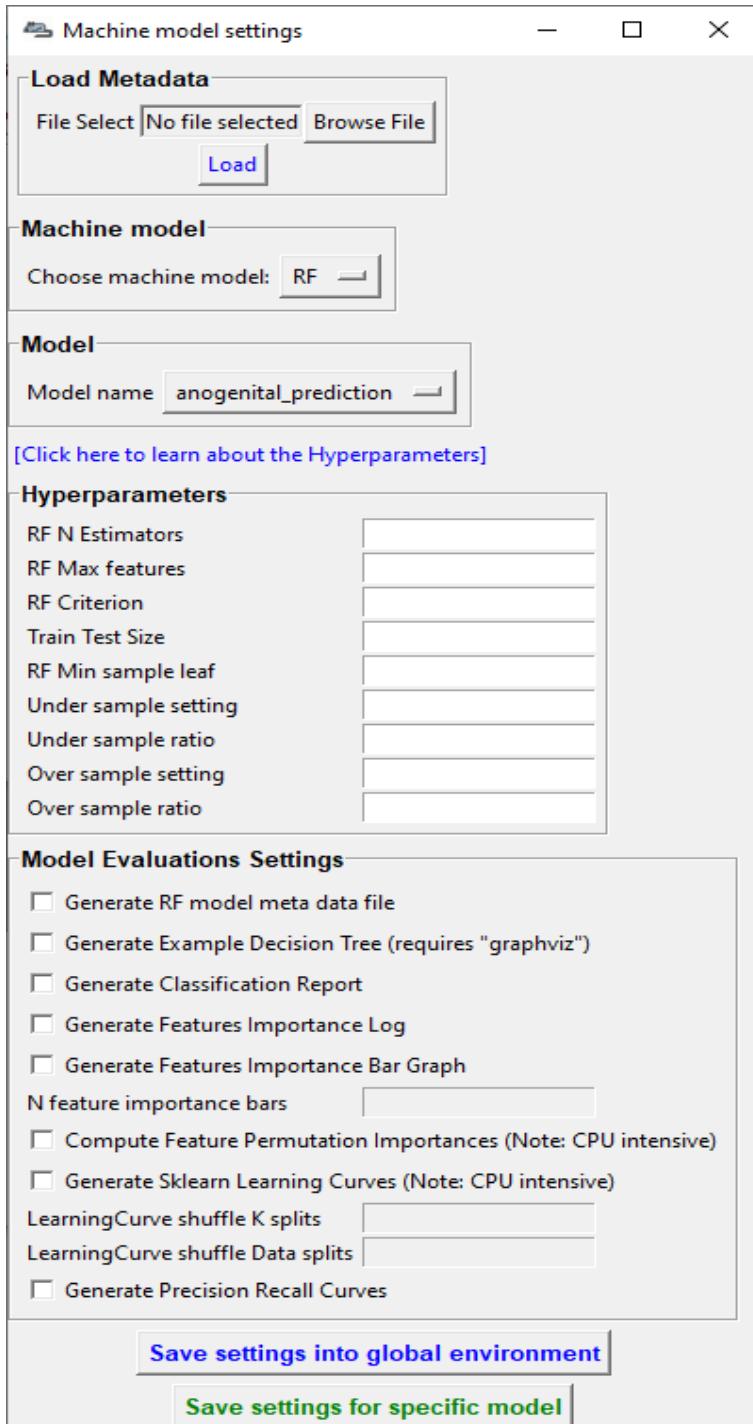
Step 7: Train Machine Model

This step is used for training new machine models for behavioral classifications.

Note: If you import existing models, you can skip this step and go straight to **Step 8** to run machine models on new video data.

Train single model

1. Click on `Settings` and the following window will pop up.



Note: If you have a .csv file containing hyper-parameter meta data, you can import this file by clicking on `Browse File` and then click on `Load`. This will autofill all the hyper-parameter entry boxes and model evaluation settings. You can find an example hyper-parameter meta data file [HERE](#).

2. Under Machine Model, choose a machine model from the drop down menu: `RF`, `GBC`, `Xboost`.

- `RF` : Random forest

- GBC : Gradient boost classifier
 - Xgboost : eXtreme Gradient boost
3. Under the **Model** heading, use the dropdown menu to select the behavioral classifier you wish to define the hyper-parameters for.
4. Under **Hyperparameters**, select the hyper-parameter settings for your model. For more details, please click [here](#). Alternatively, import the recommended settings from a meta data file (see above, **Step 1**).
- RF N estimators : Number of decision trees in the decision ensemble.
 - RF Max features : Number of features to consider when looking for the best split.
 - RF Criterion : The metric used to measure the quality of each split, i.e "gini" or "entropy".
 - Train Test Size : The ratio of the dataset withheld for testing the model (e.g., 0.20).
 - RF Min sample leaf : The minimum number of samples required to be at a leaf node.
 - Under sample setting : "Random undersample" or "None". If "Random undersample", a random sample of the majority class will be used in the train set. The size of this sample will be taken as a ratio of the minority class and should be specified in the "under sample ratio" box below. For more information, click [here](#).
 - Under sample ratio : The ratio of samples of the majority class to the minority class in the training data set. Applied only if "Under sample setting" is set to "Random undersample". Ignored if "Under sample setting" is set to "None" or NaN.
 - Over sample setting : "SMOTE", "SMOTEN" or "None". If "SMOTE" or "SMOTEN", synthetic data will be generated in the minority class based on k-means to balance the two classes. For more details, click [here](#). Alternatively, import recommended settings from a meta data file (see **Step 1**).
 - Over sample ratio : The desired ratio of the number of samples in the minority class over the number of samples in the majority class after over sampling.

To learn more about machine learning hyperparameters, click [HERE] to go to the Scenario 1 tutorial (<https://github.com/sgoldenlab/simba/blob/master/docs/tutorial.md#step-7-train-machine-model>) or see the [SimBA pre-print](#)

5. Under Model Evaluation Settings.

- Generate RF model meta data file : Generates a .csv file listing the hyper-parameter settings used when creating the model. The generated meta file can be used to create further models by importing it in the **Load Settings** menu (see above, **Step 1**).
 - Generate Example Decision Tree : Saves a visualization of a random decision tree in .pdf and .dot formats. Requires [graphviz](#). For more information, click [here](#).
 - Generate Classification Report : Saves a classification report truth table in .png format. Depends on [yellowbrick](#). For more information, click [here](#).
 - Generate Features Importance Log : Creates a .csv file that lists the importance's ([gini importances](#)) of all features for the classifier.
 - Generate Features Importance Bar Graph : Creates a bar chart of the top N features based on gini importances. Specify N in the `N feature importance bars` entry box below.
 - `N feature importance bars` : Integer defining the number of top features to be included in the bar graph (e.g., 15).
 - Compute Feature Permutation Importance's : Creates a .csv file listing the importance's (permutation importance's) of all features for the classifier. For more details, please click [here](#)). **Note:** Calculating permutation importance's is computationally expensive and takes a long time.
 - Generate Sklearn Learning Curve : Creates a .csv file listing the f1 score at different test data sizes. For more details, please click [here](#)). This is useful for estimating the benefit of annotating further data.
 - `LearningCurve shuffle K splits` : Number of cross validations applied at each test data size in the learning curve.
 - `LearningCurve shuffle Data splits` : Number of test data sizes in the learning curve.
 - Generate Precision Recall Curves : Creates a .csv file listing precision at different recall values. This is useful for titration of the false positive vs. false negative classifications of the models.
6. Click on the `Save settings into global environment` button to save your settings into the `project_config.ini` file and use the settings to train a single model.

7. Alternatively, click on the `Save settings for specific model` button to save the settings for one model. To generate multiple models - for either multiple different behaviors and/or using multiple different hyper-parameters - re-define the Machine model settings and click on `Save settings for specific model` again. Each time the `Save settings for specific model` is clicked, a new config file is generated in the `/project_folder/configs` folder. In the next step (see below), a model for each config file will be created if pressing the **Train multiple models, one for each saved settings** button.
8. If training a single model, click on `Train Model`.

To train multiple models

It is often necessary to train multiple models, in order to explore several different hyperparameters, or get classifiers for multiple different behaviors. To train multiple models at once,

1. Click on `Settings`.
2. Under **Machine Model**, choose the machine model from the drop down menu, RF , GBC , Xboost .
3. Under **Model**, select the model you wish to train from the drop down menu.
4. Then, set the **Hyperparameters**.
5. Click the `Save settings for specific model` button. This generates a `meta.csv` file, located in your `project_folder/configs` directory, which contains your selected hyperparameters. Repeat the steps to generate multiple models. One model will be generated for each of the `meta.csv` files in the `project_folder/configs` directory.
6. Close the `Machine models settings` window.
7. Click on the green `Train Multiple Models`, one for each saved settings button.

Optional step before running machine model on new data

The user can validate each model (*sav format*). In this validation step the user specifies the path to a previously created model in `.sav` file format, and a `.csv` file containing the features extracted from a video. This process will (i) run the classifications on the video, and (ii) create a video with the predictions overlaid together with a gantt plot showing predicted behavioral bouts. [Click here](#) for an example validation video.

1. Click `Browse File` and select the `project_config.ini` file and click `Load Project`.

2. Under [Run machine model] tab --> validate Model on Single Video, select your features file (.csv). It should be located in `project_folder/csv/features_extracted`.

3. Under Select model file , click on Browse File to select a model (.sav file).

4. Click on Run Model .

5. Once, it is completed, it should print "*Predictions generated.*", now you can click on Generate plot . A graph window and a frame window will pop up.

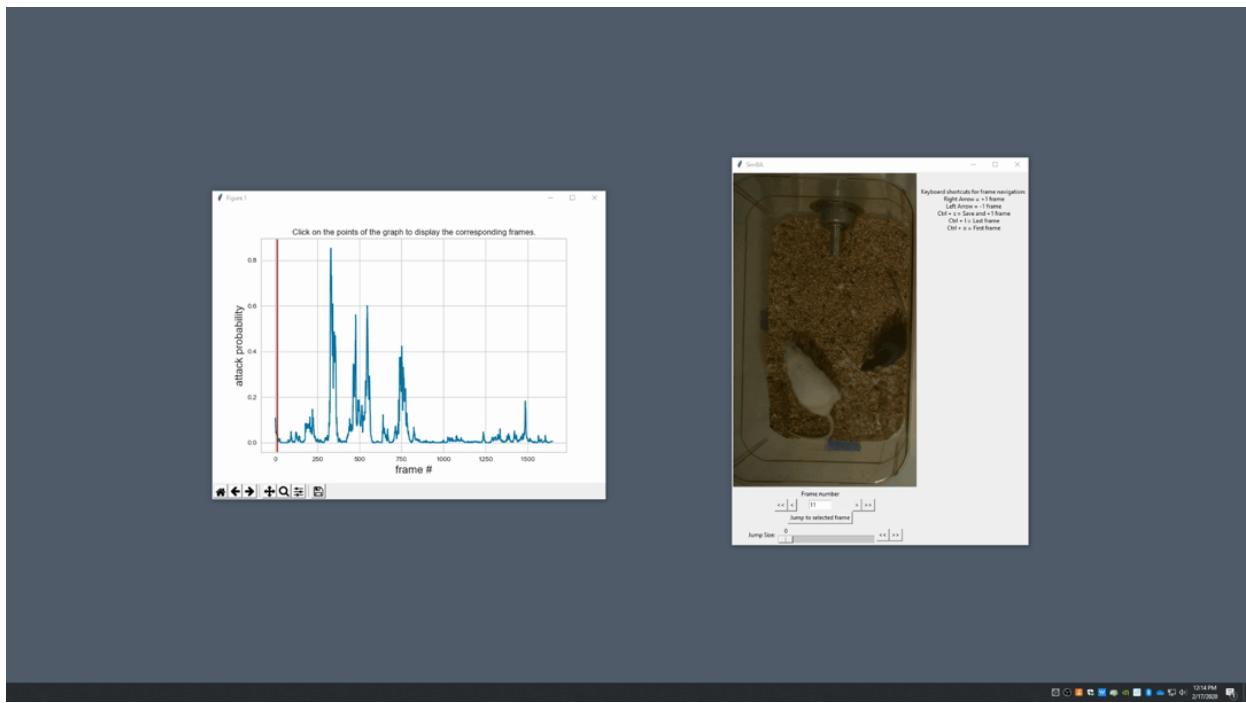
- Graph window : model prediction probability versus frame numbers will be plot. The graph is interactive, click on the graph and the frame window will display the selected frames.
- Frame window : Frames of the chosen video with controls.



7. Click on the points on the graph and picture displayed on the other window will jump to the corresponding frame. There will be a red line to show the points that you have clicked.



- Once it jumps to the desired frame, you can navigate through the frames to determine if the behavior is present. This step is to find the optimal threshold to validate your model.



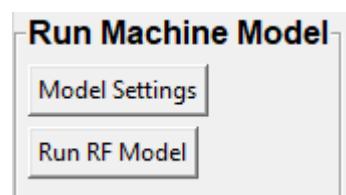
- Once the threshold is determined, enter the threshold into the `Discrimination threshold` entry box and the desire minimum behavior bout length into the `Minimum behavior bout lenght(ms)` entrybox.

- **Discrimination threshold** : The level of probability required to define that the frame belongs to the target class. Accepts a float value between 0.0-1.0. For example, if set to 0.50, then all frames with a probability of containing the behavior of 0.5 or above will be classified as containing the behavior. For more information on classification threshold, click [here](#)
- **Minimum behavior bout length (ms)** : The minimum length of a classified behavioral bout. **Example:** The random forest makes the following attack predictions for 9 consecutive frames in a 50 fps video: 1,1,1,1,0,1,1,1,1. This would mean, if we don't have a minimum bout length, that the animals fought for 80ms (4 frames), took a brake for 20ms (1 frame), then fought again for another 80ms (4 frames). You may want to classify this as a single 180ms attack bout rather than two separate 80ms attack bouts. With this setting you can do this. If the minimum behavior bout length is set to 20, any interruption in the behavior that is 20ms or shorter will be removed and the behavioral sequence above will be re-classified as: 1,1,1,1,1,1,1,1 - and instead classified as a single 180ms attack bout.

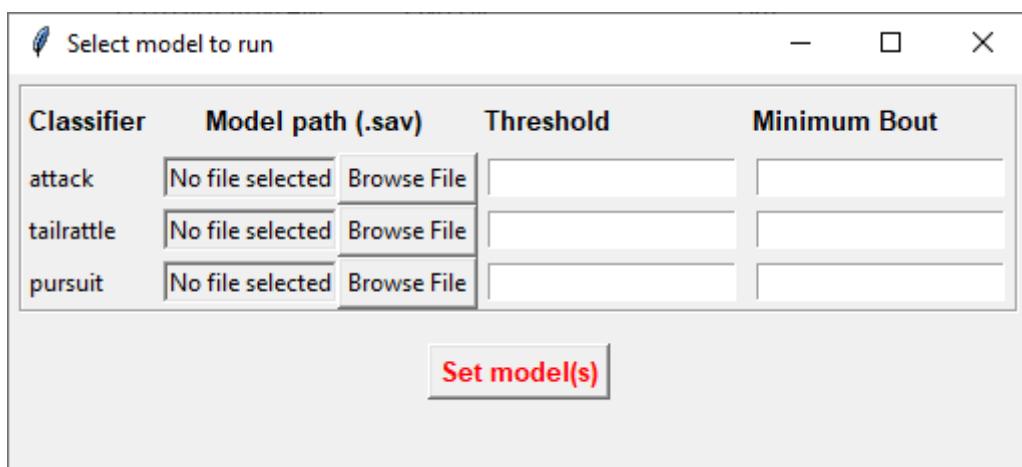
10. Click **Validate** to validate your model. Note that this step will take a long time as it will generate a lot of frames.

Step 8: Run Machine Model

This step runs behavioral classifiers on new data.



1. Under the **Run Machine Model** heading, click on **Model Selection**. The following window with the classifier names defined in the *project_config.ini* file will pop up.



2. Click on **Browse File** and select the model (.sav) file associated with each of the classifier names.

3. Once all the models have been chosen, click on `Set Model` to save the paths.

4. Fill in the `Discrimination threshold`.

- `Discrimination threshold` : The level of probability required to define that the frame belongs to the target class (see above).

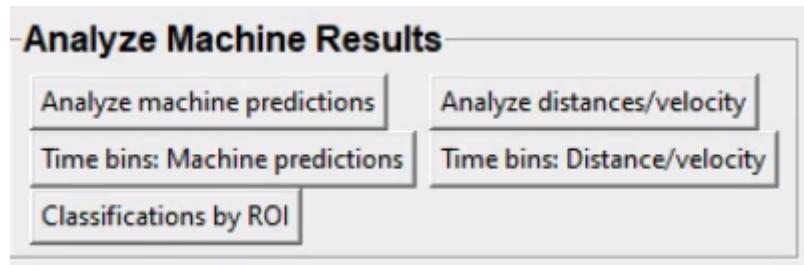
5. Fill in the `Minimum behavior bout length`.

- `Minimum behavior bout length (ms)` : The minimum length of a classified behavioral bout(see above).

6. Click on `Set model(s)` and then click on `Run RF Model` to run the machine model on the new data.

Step 9: Analyze Machine Results

Access this menu through the `Load project` menu and the `Run machine model` tab. This step performs summary analyses and presents descriptive statistics in .csv file format. There are currently (08/02/21) five forms of summary analyses in this menu: `Analyze machine predictions` , `Analyze distance/velocity` , `Time bins machine predictions` , `Time bins: Distance velocity` , and `Classifications by ROI` .



- `Analyze machine predictions` : This button generates descriptive statistics for each predictive classifier in the project, including the total time, the number of frames, total number of 'bouts', mean and median bout interval, time to first occurrence, and mean and median interval between each bout. A date-time stamped output csv file with the data is saved in the `/project_folder/log` folder. Clicking this button brings up a menu with tick boxes for the different output metrics (see image below). Select the metrics you would like to calculate and click `Analyze` .

<input type="checkbox"/> # bout events
<input type="checkbox"/> total events duration (s)
<input type="checkbox"/> mean bout duration (s)
<input type="checkbox"/> median bout duration (s)
<input type="checkbox"/> first occurrence (s)
<input type="checkbox"/> mean interval (s)
<input type="checkbox"/> median interval (s)
<input type="button" value="Analyze"/>

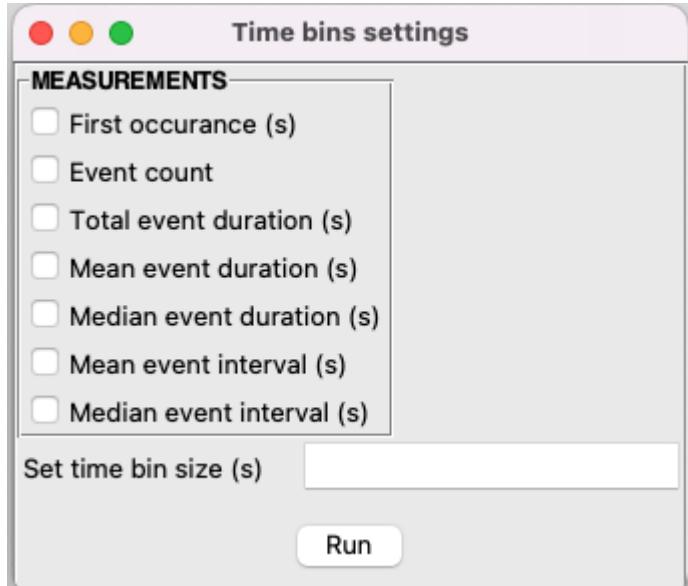
- **Analyze distance/velocity :** This button generates descriptive statistics for mean and median movements of the animals, and distances between animals (if you have more than 1 animal). The date-time stamped output csv file with the data is saved in the `/project_folder/log` folder. Clicking this button brings up a menu (see image below) where you select the number of animals you want distance/velocity data for, and the body-parts you would like to use when calculating the data. You can also omit calculations for body-parts below a user-defined probability threshold. Define your settings and click `Run`.

Analyze distances/velocity

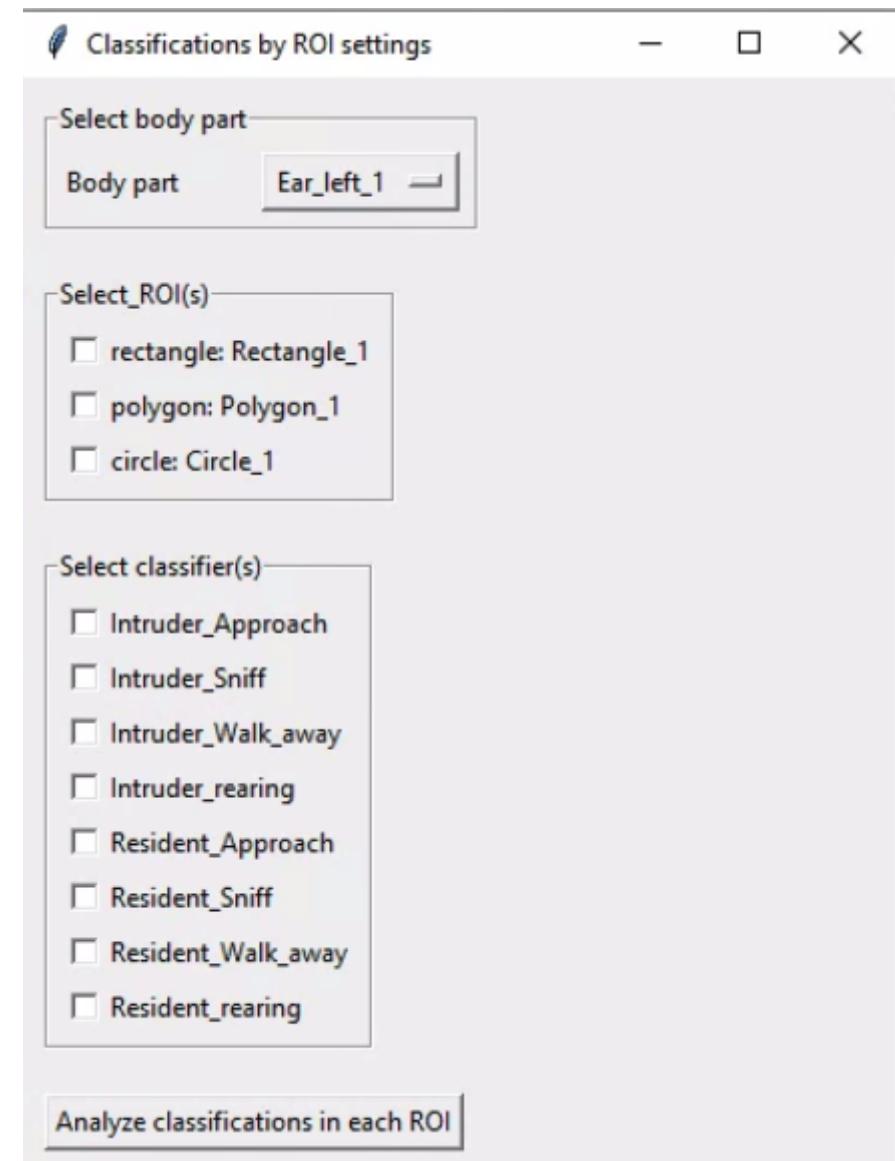
Select number of animals
of animals <input type="text" value="2"/> <input type="button" value="Confirm"/>
Choose bodyparts
Animal 1
<input type="text" value="Ear_left_1"/>
Animal 2
Bodypart
<input type="text" value="Ear_left_2"/>
Bp probability threshold <input type="text" value="0.0"/>
<input type="checkbox"/> Calculate distance moved within ROI
<input type="button" value="Run"/>

- **Time bins: Machine predictions :** This button generates descriptive statistics for selected predictive classifier in the project within user-defined time-bins. The metrics available for every time bin are Number of events , Total event duration , Mean event duration , Median event duration , Time of first occurrence , Mean interval duration (time between event in each time-bin), and Median interval duration . A date-time stamped output CSV file with the data is saved in the `/project_folder/log` folder. Clicking this button bring up a menu (see below) where the user inserts the length of each time-bin and the metrics required. For example, if your videos are 10 minutes long, and you specify `60` in the Time bin

(s) entry box, then SimBA will divide each video into 10 sequential bins and provide the chosen data for each of the sequential time bins.



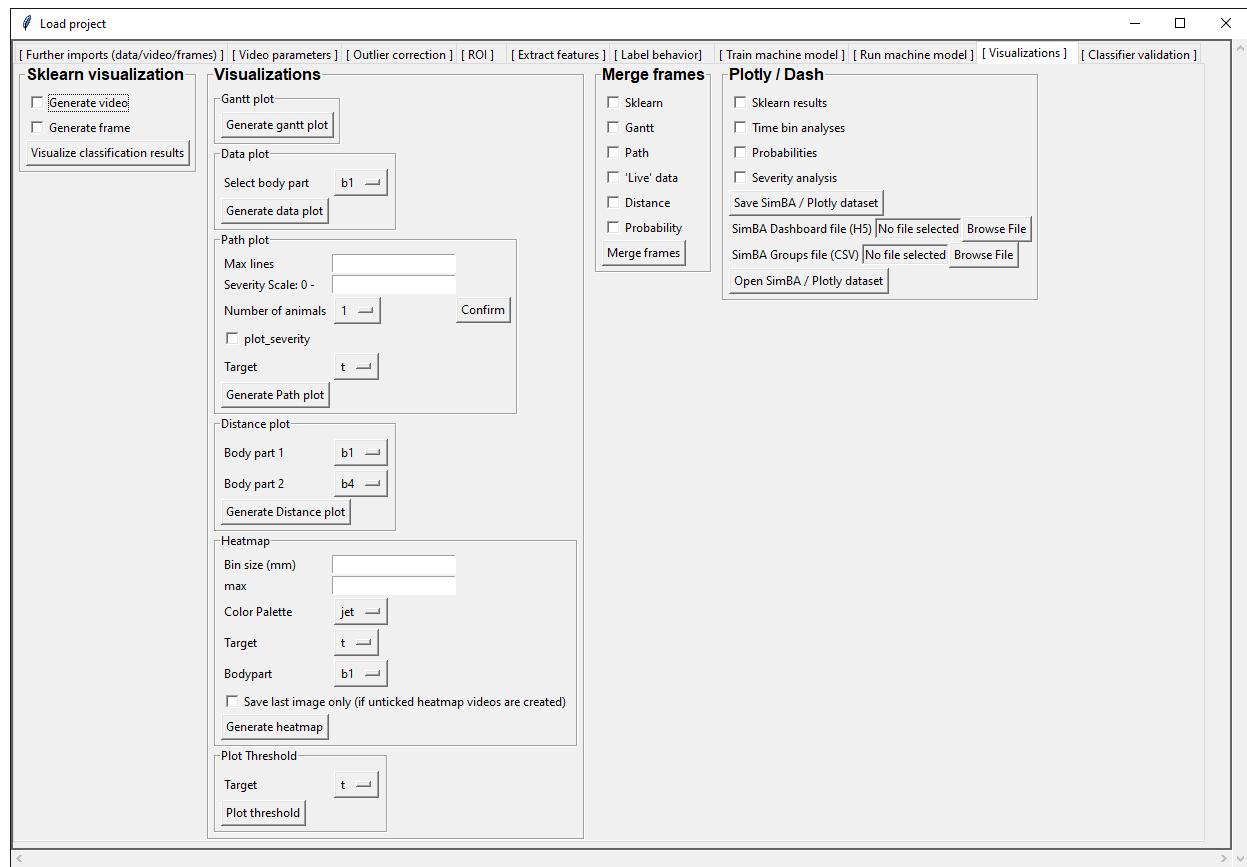
- **Time bins: Distance / Velocity** : This button generates descriptive statistics for mean and median movements of the animals, and distances between animals (if you have more than 1 animal) in user defined time-bins. Clicking this button brings up a menu where the user defines the size of the time-bins.
- **Classifications by ROI** : This button generates summary statistics for each classifier within each user-defined region-of-interest (ROI). If you have not defined ROIs, then first head to the [ROI tab](#) before proceeding with this analysis. You can read more about defining ROIs on SimBA [HERE](#). Clicking this button brings up a menu where the user defines the ROIs and classifiers they would like to calculate summary statistics for (see below). The user is also asked to define the body-part determining the current location of the animal(s). Once complete, click on `Run classifications in each ROI` . A date-time stamped output CSV file with the data is saved in the `/project_folder/log` folder named something like `Classification_time_by_ROI_20210801173201.csv` . For an example output data file (with 5 videos, 3 ROIs, and 3 behaviors analyzed) click [HERE](#).



- **Analyze severity** : Calculates the 'severity' of each frame classified as containing attack behavior based on a user-defined scale. **Example:** the user sets a 10-point scale. One frame is predicted to contain an attack, and the total body-part movements of both animals in that frame is in the top 10% percentile of movements in the entire video. In this frame, the attack will be scored as a 10 on the 10-point scale. A date-time stamped output .csv file containing the 'severity' data is saved in the `/project_folder/log` folder.

Step 10: Sklearn Visualization

These steps generate visualizations of features and machine learning classification results. This includes images and videos of the animals with prediction overlays, gantt plots, line plots, paths plots and data plots etc. In this step the different videos can also be merged into video mp4 format.



- Under the **Sklearn visualization** heading, check on the box and click on `Visualize classification results`.

`Generate video` : This generates a video of the classification result

`Generate frame` : This generates frames(images) of the classification result

Note: Generate frames are required if you want to merge frames into videos in the future.

This step grabs the frames of the videos in the project, and draws circles at the location of the tracked body parts, the convex hull of the animal, and prints the behavioral predictions on top of the frame. For an example, click [here](#).

Step 11: Visualizations

The user can also create a range of plots: **gantt plot**, **Data plot**, **Path plot**, **Distance plot**, and **Heatmap**.

Visualizations

Gantt plot

Data plot

Path plot

Max lines

Severity Scale: 0 -

Bodyparts

plot_severity

Distance plot

Body part 1

Body part 2

Heatmap

Bin size (px)

Scale increments

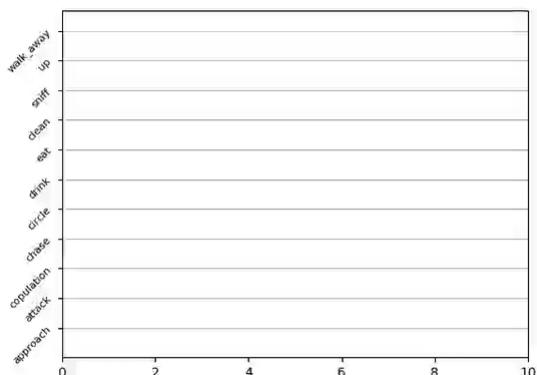
Scale increment (s)

Color Palette

Target

Gantt plot

Gantt plot generates gantt plots that display the length and frequencies of behavioral bouts for all the videos in the project.



- Under the **Gantt plot** heading, click on **Generate Gantt plot** and gantt plot frames will be generated in the `project_folder/frames/output/gantt_plots` folder.

Data plot

Generates 'live' data plot frames for all of the videos in the project that display current distances and velocities.

```
Mean velocity black mouse 22.43 cm/s
Mean velocity white mouse: 16.4 cm/s
Total movement black mouse 22.43 cm
Total movement white mouse: 16.4 cm
Current velocity black mouse 22.43 cm/s
Current velocity white mouse: 16.4 cm/s
Centroid distance: 4.2 cm
Nose to nose distance: 1.5 cm
```

1. Under the **Data plot** heading, click on `Generate Data plot` and data plot frames will be generated in the `project_folder/frames/output/live_data_table` folder.

Path plot

Generates path plots displaying the current location of the animal trajectories, and location and severity of attack behavior, for all of the videos in the project.



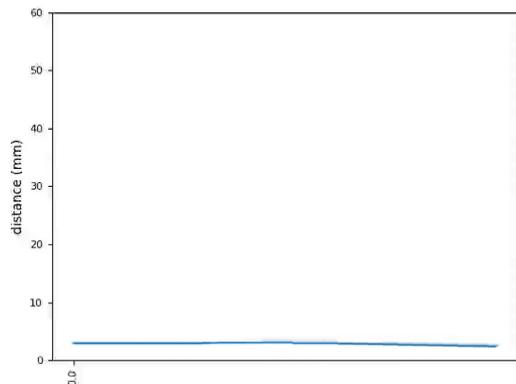
1. Under the **Path plot** heading, fill in the following user defined values.
 - **Max Lines** : Integer specifying the max number of lines depicting the path of the animals. For example, if 100, the most recent 100 movements of animal 1 and animal 2 will be plotted as lines.
 - **Severity Scale** : Integer specifying the scale on which to classify 'severity'. For example, if set to 10, all frames containing attack behavior will be classified from 1 to 10 (see above).

- **Bodyparts** : String to specify the bodyparts tracked in the path plot. For example, if Nose_1 and Centroid_2, the nose of animal 1 and the centroid of animal 2 will be represented in the path plot.
- **plot_severity** : Tick this box to include color-coded circles on the path plot that signify the location and severity of attack interactions.

2. Click on `Generate Path plot`, and path plot frames will be generated in the `project_folder/frames/output/path_plots` folder.

Distance plot

Generates distance line plots between two body parts for all of the videos in the project.



1. Fill in the `Body part 1` and `Body part 2`

- `Body part 1` : String that specifies the the bodypart of animal 1. Eg., Nose_1
- `Body part 2` : String that specifies the the bodypart of animal 1. Eg., Nose_2

2. Click on `Generate Distance plot`, and the distance plot frames will be generated in the `project_folder/frames/output/line_plot` folder.

Heatmap

Generates heatmap of behavior that happened in the video.

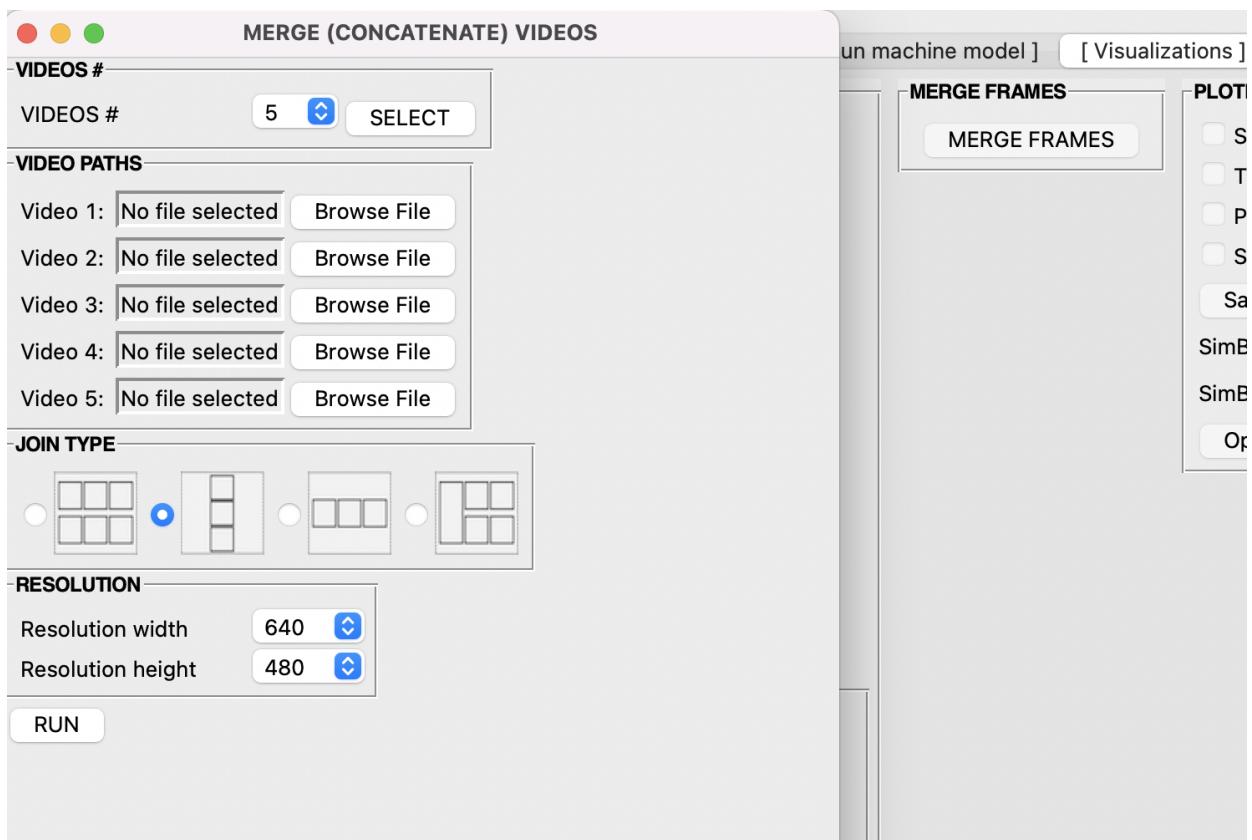
To generate heatmaps, SimBA needs several user-defined variables:

- `Bin size(mm)` : Pose-estimation coupled with supervised machine learning in SimBA gives information on the location of an event at the single pixel resolution, which is too-high of a resolution to be useful in heatmap generation. In this entry box, insert an integer value (e.g., 100) that dictates, in pixels, how big a location is. For example, if the user inserts 100, and the video is filmed using 1000x1000 pixels, then SimBA will generate a heatmap based on 10x10 locations (each being 100x100 pixels large).
- `max` (integer, or auto): How many color increments on the heatmap that should be generated. For example, if the user inputs 11, then a 11-point scale will be created (as in the gifs above). If the user inserts auto in this entry box, then SimBA will calculate the ideal number of increments automatically for each video.
- `Color Palette` : Which color palette to use to plot the heatmap. See the gifs above for different output examples.
- `Target` : Which target behavior to plot in the heatmap. As the number of behavioral target events increment in a specific location, the color representing that region changes.
- `Bodypart` : To determine the location of the event in the video, SimBA uses a single body-part coordinate. Specify which body-part to use here.
- `Save last image only` : Users can either choose to generate a "heatmap video" for every video in your project. These videos contain one frame for every frame in your video. Alternative, users may want to generate a **single image** representing the final heatmap and all of the events in each video - with one png for every video in your project. If you'd like to generate single images, tick this box. If you do not tick this box, then videos will be generated (which is significantly more time-consuming).

2. Click `Generate heatmap` to generate heatmap of the target behavior. For more information on heatmaps based on behavioral events in SimBA - check the [tutorial for scenario 2 - visualizing machine predictions](#)

Step 12: Merge Frames

Next, we may want to merge (concatenate) several of the videos we have created in the prior steps into a single video file. To do this, click the `MERGE FRAMES` button in the [VISUALIZATIONS] tab, and you should see this pop up to the left:



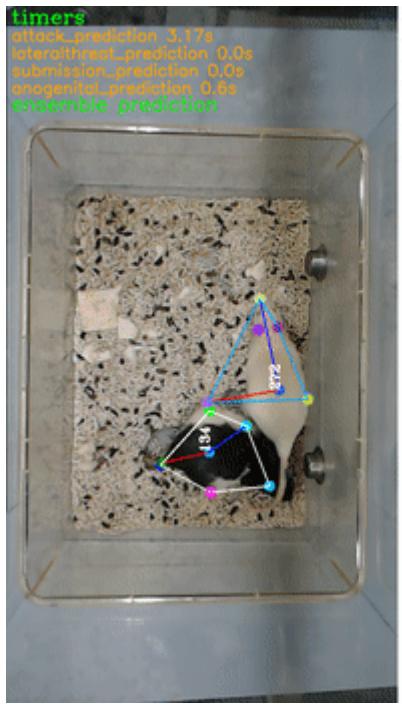
Begin by selecting how many videos you want to concatenate together in the `VIDEOS #` drop-down menu and click `SELECT`. A table, with one row representing each of the videos, will show up titled `VIDEO PATHS`. Here, click the `BROWSE FILE` button and select the videos that you want to merge into a single video.

Next, in the `JOIN TYPE` sub-menu, we need to select how to join the videos together, and we have 4 options:

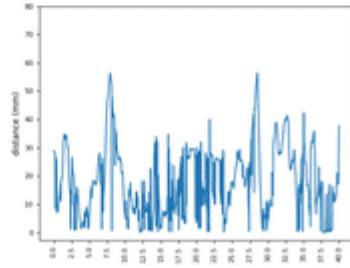
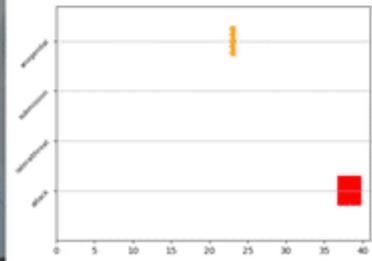
- **MOSAIC:** Creates two rows with half of your chosen videos in each row. If you have an unequal number of videos you want to concatenate, then the bottom row will get an additional blank space.
- **VERTICAL:** Creates a single column concatenation with the selected videos.
- **HORIZONTAL:** Creates a single row concatenation with the selected videos.
- **MIXED MOSAIC:** First creates two rows with half of your chosen videos in each row. The video selected in the `Video 1` path is concatenated to the left of the two rows.

Finally, we need to choose the resolution of the videos in the `Resolution width` and the `Resolution height` drop-down menus. If choosing the **MOSAIC**, **VERTICAL**, or **HORIZONTAL** join type, this is the resolution of each panel video in the output video. If choosing **MIXED MOSAIC**, then this is the resolution of the smaller videos in the panel.

After clicking `RUN`, you can follow the progress in the main SimBA terminal and the OS terminal. Once complete, a new output video with a date-time stamp in the filename is saved in the `project_folder/frames/output/merged` directory of your SimBA project.



Mean velocity Lister:
11.14 cm/s
Mean velocity Sprague:
17.1 cm/s
Total movement Lister:
456.59 cm
Total movement Sprague:
701.21 cm
Current velocity Lister:
15.85 cm/s
Current velocity Sprague:
11.8 cm/s
Centroid distance:
9.1 cm
Nose to nose distance:
19.9 cm



Plotly

Please click [here](#) to learn how to use plotly.