Otto-von-Guericke-University Magdeburg

Faculty of Computer Science

Institut für Simulation und Graphik

Visual Analysis of Drosophila Larval Locomotion Quantification

Bachelor Thesis

Author:

Michael Thane

Examiner and Supervisor:

Priv.-Doz. Dr.-Ing.habil. Dirk Joachim Lehmann

2nd Examiner:

M. Sc. Vishnu Unnikrishnan

Supervisor:

Dr. Michael Schleyer

Magdeburg, 28.01.2020

Contents

Abstract

The larva of *Drosophila melongaster* is often used in genetics as a model organism. Studying how learning leads to subtle changes in behavior of the insect is a challenging task which involves a lot of processing steps. This bachelor thesis tries to face these challenges using several visualization methods. The visualization is created using the R programming language and a library for interactive web-pages called "Shiny". Using this package an interactive tool was created that can help the user to explore the data sets in great detail.

As there are more than 50 derived behavioral measurements a high-dimensional data analysis was needed. Using high-dimensional data visualization and Machine Learning we found out what features are important to distinguish different experimental groups and in which features there are significant differences in the distributions. The results of the high-visualizations brought up interesting biological questions that might lead to further research in the field.

Acknowledgements

I am thanking my supervisor Dr. Michael Schleyer, Professor Gerber and the team of Genetics of Learning and Memory at the Leibniz-Institute for Neurobiology for the great support and many interesting discussions.

Also I want to acknowledge my backup from the Faculty of Computer Science Vishnu Viswanathan and PD Dirk Lehmann.

Furthermore I want to thank Emmanouil Paisios for answering questions regarding the tracking software and Panagiotis Sakagiannis for suggestions on what can be implemented in the dashboard.

Statement of Authorship / Selbstständigkeitserklärung

Hiermit erkläre ich, dass ich die vorliegende Bachelorarbeit selbstständig und außschließlich unter Verwendung der angegebenen Literatur und Hilfsmittel angefertigt habe.
Die aus fremden Quellen direkt oder indirekt übernommenen Stellen sind als solche kenntlich gemacht.

Die Arbeit wurde bisher in gleicher oder ähnlicher Form weder einer anderen Prüfungsbehörde vorgelegt noch anderweitig veröffentlicht.

Unterschrift	Datum

Index of Notation

Variables

HC	Head cast
IS	Inter step, a step is referring to each peristaltic the larvae does
TS	Time series
PREF	Preference

Quantities & Functions

1 Introduction

The learning behaviour of the larva *Drosophila melongaster* is a interesting and well known topic amongst biologists. The insect is used widely in Biology and Neuroscience and can be used as a model in genetic studies.

Using classical conditioning one can train larvae, such that they associate stimuli, in this case an odour, with a reward or a punishment. To understand the impact of different substances, mutations or optogonetic treatment is the main goal of the group Genetics of Learning and Memory at the Leibniz-Institute for Neurobiology. Especially understanding how memory is changing the micro-behavior is a difficult task.

1.1 Motivation

There are a lot of derived variables in the data sets. That means one has to find a way to present the data in a meaningful way. It would help to understand how those variables change between different treated animals. If one can automatize as much steps as possible in the data analysis pipeline one can save time and make the analysis more efficient. The current analysis is focusing on average values per Petri dish. An analysis per individual larva is potentially interesting because it increases the sample size of the data set. On top of that visualizing a single animal, its time series values and trajectory is needed to get a better insight into the detailed behavior.

Over the time there were more and more variables derived from the data set. With the growing number of derived variables it becomes more and more difficult to compare all the combinations of experimental conditions and interesting findings might be overlooked. Dealing with a high-dimensional data set needs visualization techniques that allow plotting multiple variables at a time instead of only looking at single distributions.

1.2 Goals

My bachelor thesis aims to support users of the larval tracking software by building a Graphical User Interface (GUI) to enable the analysis and interactive exploration of the data sets. I want to improve the current analysis to make it more maintainable and comprehensive. The implementation of the software will be in R using the „shiny“ package [19] which is made for interactive web dashboards.

The UI will enable selecting, filtering and visualization of the data sets. This way I will tackle the issues that I mentioned in the motivation.

On top of that I want to use methods for high-dimensional visualization to analyze the data sets. Using Visualization and Machine Learning I want to cluster various behavior attributes and look at the correlation between those attributes. On top of that I want to investigate differences in several experimental groups.

1.3 Thesis Outline

In chapter 2 I want to give an overview about the research that has made this project possible. I will explain the background in Neurobiology and the experimental setup in the lab. I will start with the theory of plotting and machine learning and will explain methods that I am using in my visualizations.

In chapter 3 I will explain briefly what other research people have done in the field of behavior quantification, larvae locomotion analysis and related machine learning work.

In the main part I want to explain how I analysed the problem and designed different visualizations. In 4 I will write about the pipeline I am using for the behavioral quantification. After I explained the data processing workflow I will introduce the dashboard in chapter 5 that will be used for starting the analysis and visualizing the data.

In 7 I will do a quantitative and qualitative evaluation of the system and the results. I will use example data to evaluate if the visualizations can help to improve the larvae locomotion research. Also I want to evaluate the discussion with the users to find out how good the visualization approach works.

2 Prerequisites

The topic of my thesis is interdisciplinary and before explaining my Implementation I want to write about the practical and theoretical concepts that this project is related to. I will explain why analyzing behaviour of *Drosophila* is an important research topic and why the fruit fly is a good animal for studying learning and behavior. Afterwards I will briefly introduce visualization and Machine Learning methods that I am using to analyze the data.

2.1 Drosophila and Neurobiology

Drosophila melongaster are a widely used animal for studying learning and behaviour. Its genes are very well researched and the larval brain has about 10,000 neurons [11, 25]. The adult fruit fly has about 100.000 neurons. At the Leibniz-Institute for Neurobiology the research focus lies on the learning and the memory of the fruit fly. Using fructose and an odour (amylacetate) one can train the larvae to associate the odour with a reward. This works for different sugars, amino acids and also for punishing stimuli like salt or a bitter taste like quinine. With our tracking system we can then evaluate the odour preference and other variables of the behaviour. It has been shown that training the animals has an impact of their behaviour. [20, 16, 23] It has been found out that for reaching the odor the animal is changing its head cast and turning behavior based on the orientation towards the odor.

Another astonishing method apart from associative learning is to do optogenetic experiments. That means certain neurons can be activated through light of a certain wavelength and intensity. This allows to activate almost every neuron in the fruit fly brain and potentially get an insight into behavioral changes due to activation of single neurons [24].

In the following I will explain a standard *Drosophila* learning experiment. The larvae are trained 3 times for 1.5 minutes in 9 cm Petri dishes. Afterwards they are recorded for 3 minutes in a test Petri dish. First, 15-20 larvae are washed before training and put on a Petri dish with a brush. In the training phase there are

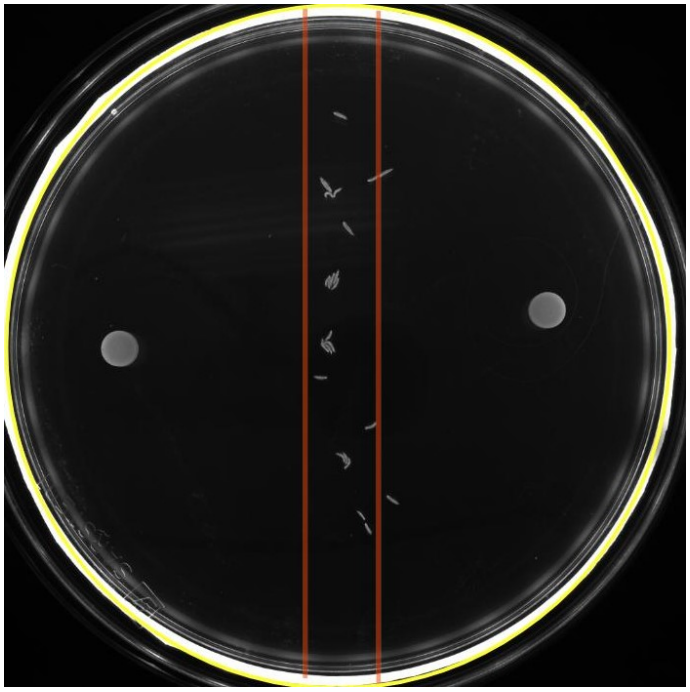


Figure 2.1: Frame of example video

An example frame of the beginning of a test video. The grey oval shaped areas are the larvae. The white dots are the odor containers. The red lines indicate the region where the animals start. When counting the animal position (left/right) this region is being excluded.

two kinds of dishes. One dish (Dish A) contains a reward stimulus e.g. fructose together with an odor container which is filled with the odor amylacetate. A different dish (Dish B) contains no odor and agarose a non-rewarding substance. The larvae are put on dish A and B for each 1.5 minutes. This procedure is repeated three times.

The first treatment is called paired treatment, because the odor is paired with a reward. In the other treatment Dish A contains agarose with an odor and Dish B contains a reward and no odor. This treatment is called unpaired. Now one can observe that the paired group is showing an appetitive behavior which means they are moving towards the odor. The unpaired group shows an aversive behavior. The larvae are moving away from the odor. In 2.1 one can see a frame of such a test video.

2.2 Statistical graphics

If we want to evaluate the behaviour which is quantified by the tracking software one needs to understand the data. For this we use descriptive statistics and hypothesis testing. Furthermore there is a number of techniques for visualizing high dimensional data.

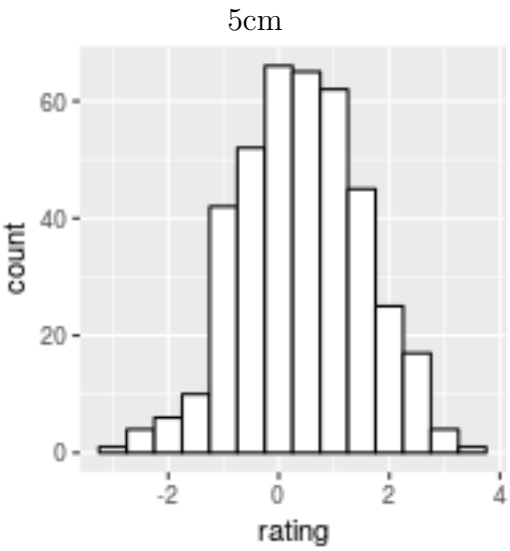


Figure 2.2: Histogram [1]

To show the distribution of a certain variable one can use **histograms** like in 2.2. A histogram is dividing the variable into a set of different intervals called bins. This way the user can see how the distribution looks like. Is it a normal distribution, an oblique distribution or does the distribution has multiple subdistributions. Analyzing the histogram can tell us what methods we could try out with the data. For example we can find out if the data is suitable for a certain statistical test or model.

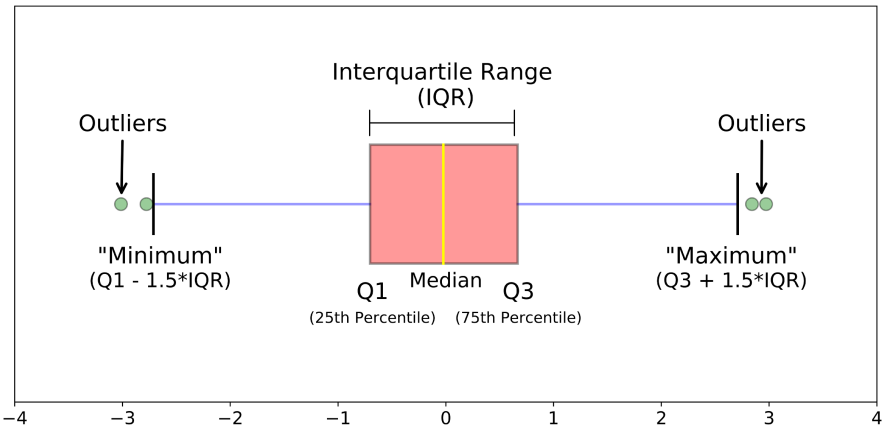


Figure 2.3: Boxplot [10]

Boxplots like in 2.3 are very often used in biology if one wants to compare distributions. Instead of dividing the distribution into a number of bins it shows the different quantiles of the distribution. The box is containing the lower quantile

(25 percentile), the median quantile (50th percentile), the upper quantile (75 percentile). The whiskers indicate the minimum and the maximum of the variable. Often also outliers are shown which can have for example have two categories (mild and extreme outliers). A different version of the boxplot is the violin plot. It is showing everything the boxplot shows, but also the width of the box is showing the distribution of the variable.

Scatterplots are projections of the data from the data domain into a spatial domain. The standard scatterplot usually projects data into two dimensions. Each point is representing one sample of the dataset. A scatterplot makes it possible to see a trajectory of an object or correlation of independent variables and is therefore often used in the trajectory analysis. Also it can make high dense areas or clusters visible. An possibility to plot more than two variables is the scatter plot matrix. Here all the possible combinations of variables are plotted in a grid way. 2.4 shows an example of such a scatter plot matrix.

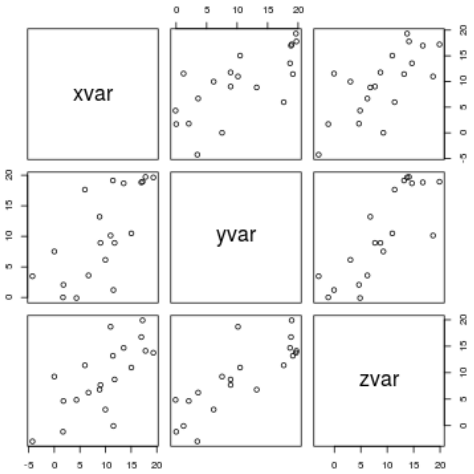


Figure 2.4: SPLOM [2]

Example of a scatter plot matrix.

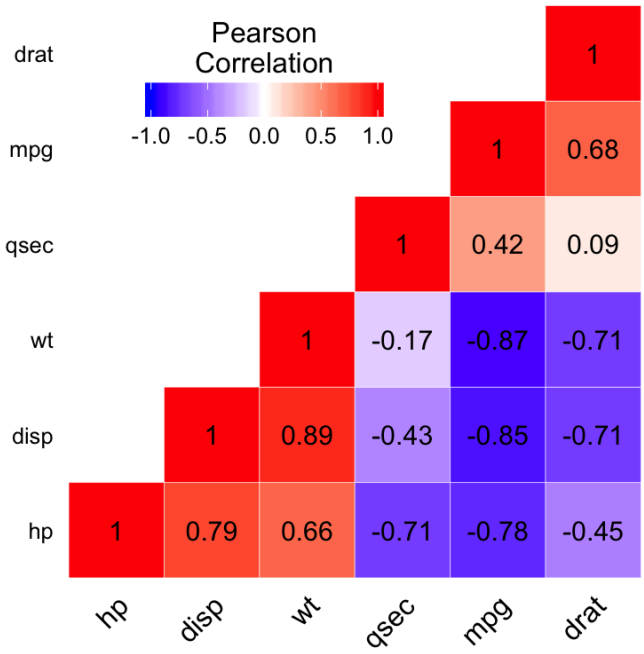


Figure 2.5: Correlation Matrix [3]

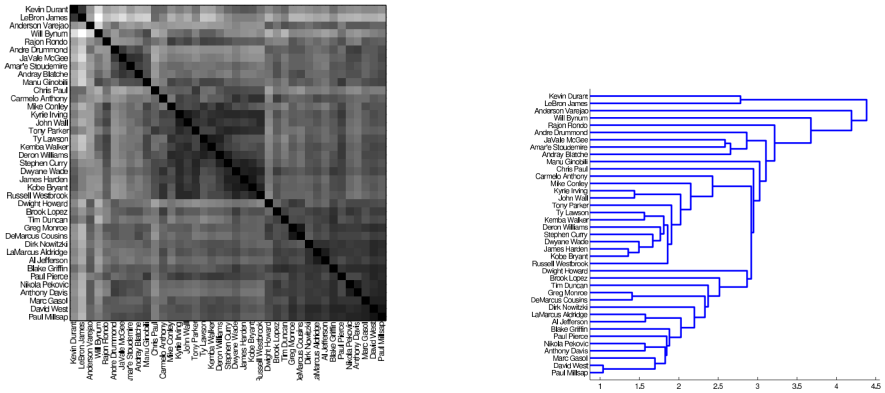
Correlation matrices can help to overcome the problems of limited space when using scatter plot matrices. The correlation matrix visualizes all combinations of correlations in a grid way using color coding. In 2.5 one can see an example of such a correlation matrix and the Pearson correlation. If one is interested merely in the correlation of all combinations and does not need to visualize all points a correlation matrix is a good choice.

2.3 Machine Learning

Machine Learning is especially useful when working with large and high dimensional data sets. The field of Machine Learning grew with the rise of computing power and is used nowadays in all science fields. One can distinguish supervised and unsupervised methods. Supervised Machine Learning requires that each data point has a certain label so that the system can "learn" a certain function based on training data. After training one can evaluate the quality of the classifier using a test set. In my thesis I will use Hierarchical Clustering and a Random Forest Classifier which I will explain in this section.

Hierarchical Clustering is an unsupervised Machine Learning method that is

often used in Life Sciences and Bioinformatics. One has to mention though that clustering algorithms always find clusters even if there are no real clusters in the data.[15] For agglomerative hierarchical clustering one assumes that in the beginning of the algorithm every data point is a cluster. Then in the next step all combination of distances are calculated to find out which clusters are most similar to each other. This is done by using an appropriate distance function e.g. the euclidean distance. The nearest clusters are then merged into one cluster. Again all distances between clusters are calculated and the nearest will be merged. This is done until there are only two clusters left.



(a) Pairwise Distances

(b) Single-Linkage Dendrogram

(a) **(b)**

Figure 2.6: Hierarchical Clustering [4]

To decide the similarity between clusters one can chose different methods. The most common methods are single linkage and complete linkage. In the single linkage method the similarity between two clusters is defined by the closest distance of all the points belonging to both clusters. In complete linkage the maximum is chosen instead. Other methods are e.g. average linkage, where the average of all distances are chosen or centroid linkage where the distance between the centroids of both clusters is calculated. In 2.6 one can see a distance matrix and its corresponding single linkage dendrogram [4].

One very common method of supervised learning is to build **decision trees**. Decision trees are a method for classification of data. The idea is to recursively partition the data into subsets maximising the information gain [14]. The information gain is based on an impurity measure. Common impurity measures are entropy or the Gini coefficient.

They have the advantage that the training data can contain errors and missing attribute values. Also decision trees can help to better understand how the

classification is working. In an Artificial Neural Network for example it is very hard to understand how the classification works as it can be seen as a black box.

Decision trees on the other side can be visualized and so one can see what attributes are important for the classification and which attributes are not so important regarding the correct classification.

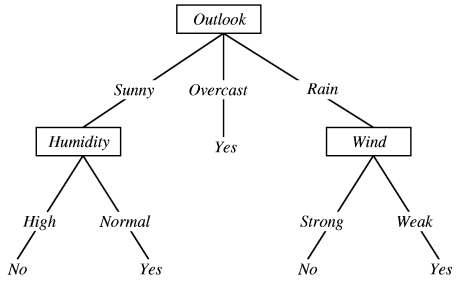


Figure 2.7: Decision tree with ID3 [14]

In this example one might say that the Outlook attribute is the most predictive attribute, because it is placed as a root node.

A popular method for training a decision tree model is the CART (Classification And Regression Trees) algorithm. The CART algorithm produces in contrast for example to the ID3 algorithm only binary trees and can also be used for regression.

Random Forests are a classification method based on decision trees. [7] Random Forests belong to the ensemble methods in Machine Learning meaning they use multiple models to improve the predictive power. They use random subspaces of the data set to build multiple uncorrelated decision trees. [7] Random Forests decide the prediction of a certain class based on a majority voting of the collective models.[7] They therefore have similar advantage as decision trees and are widely used among many research disciplines.

3 Related Work

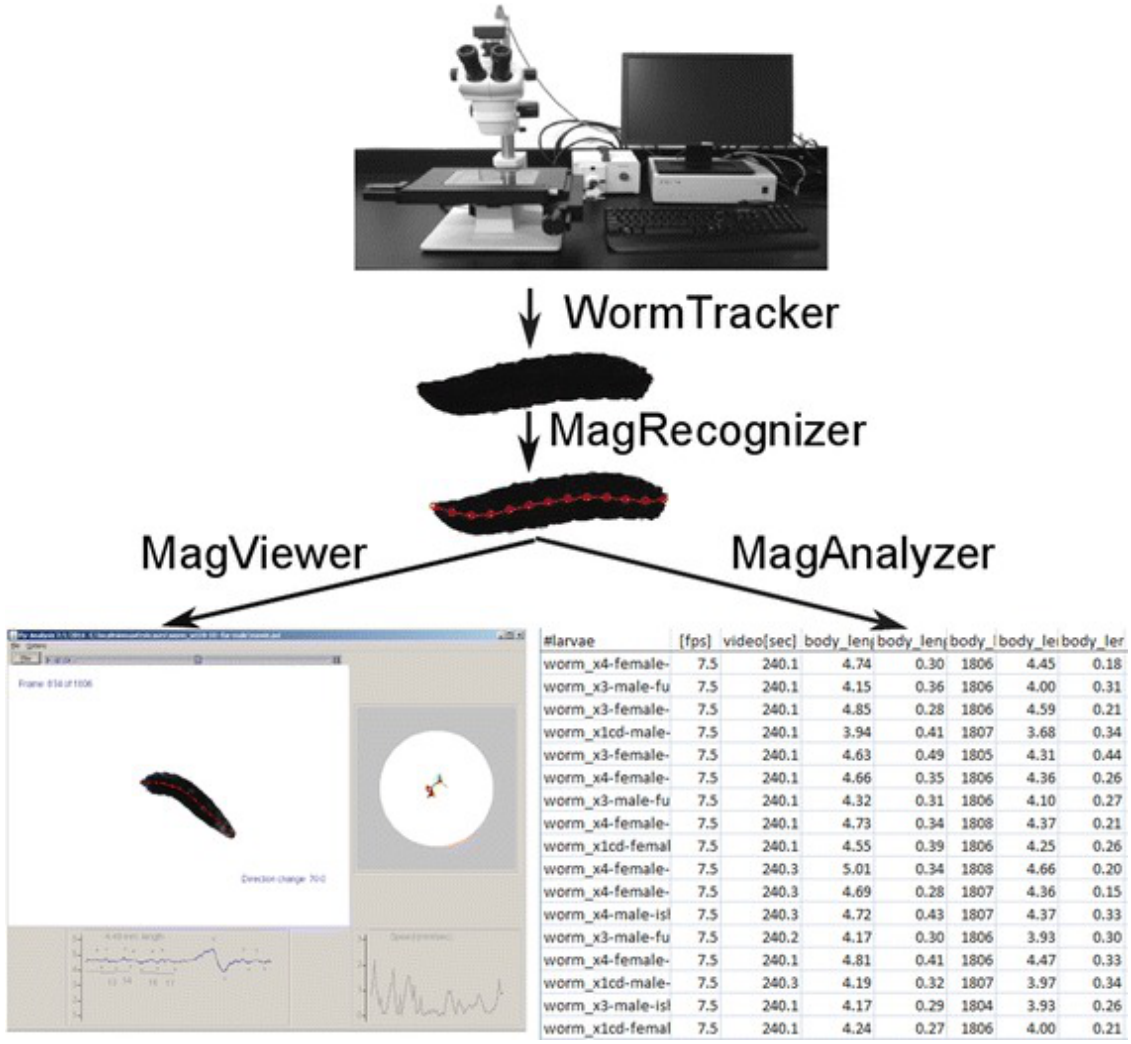
Quantifying the behaviour of the larvae is a task which involves three main components:

1. Experimental setup and video recording
2. Processing of the video
3. Analysis of the trajectory

There are several approaches how to quantify larvae locomotion behavior. In the following I want to introduce different approaches on how to quantify *Drosophila* behavior.

The paper "An automated system for quantitative analysis of *Drosophila* larval locomotion" [5] is using a single animal approach. A software called "MaggotTracker" has been implemented which can handle the image processing of the video, the analysis of the variables and the visualization of the larvae body and the time series. They record the larvae using a camera and they are using a high magnification microscope. First the larvae are being tracked and then in a second step, artificial spine points and contours are being recognized. They implemented a "MagViewer" to explore the time series of the single animal and a "MagAnalyzer" which calculated mean parameter values from the videos. They derive 22 different parameters to analyze the behavior. They use scatter plots to show correlations of the variables and histograms to show factors affecting locomotive behavior. In 3.1 one can see an overview of their setup.

The paper "Dynamic analysis of larval locomotion in *Drosophila* chordotonal organ mutants" [8] is also analyzing the trajectory of the larvae. For doing this they use the "Data Integration and Analysis (DIAS)". They calculate simple parameters like speed, but also measure the peristaltic wave that the animal is doing by comparing body length vs. the time. They are smoothening the data to then find the maximum to get a "stride period". Also they identify the "turns" and "retreats" that the animals are doing. They do this by a threshold by saying that "turns" changes in direction $<30°$ followed by a linear locomotion. "Retreats" are changes in direction $>30°$ followed by a linear locomotion.



#larvae	[fps]	video[sec]	body_len	body_len	body_l	body_le	body_ler
worm_x4-female-	7.5	240.1	4.74	0.30	1806	4.45	0.18
worm_x3-male-fu	7.5	240.1	4.15	0.36	1806	4.00	0.31
worm_x3-female-	7.5	240.1	4.85	0.28	1806	4.59	0.21
worm_x1cd-male-	7.5	240.1	3.94	0.41	1807	3.68	0.34
worm_x3-female-	7.5	240.1	4.63	0.49	1805	4.31	0.44
worm_x4-female-	7.5	240.1	4.66	0.35	1806	4.36	0.26
worm_x3-male-fu	7.5	240.1	4.32	0.31	1806	4.10	0.27
worm_x4-female-	7.5	240.1	4.73	0.34	1808	4.37	0.21
worm_x1cd-femal	7.5	240.1	4.55	0.39	1806	4.25	0.26
worm_x4-female-	7.5	240.3	5.01	0.34	1808	4.66	0.20
worm_x4-female-	7.5	240.3	4.69	0.28	1807	4.36	0.15
worm_x4-male-isl	7.5	240.3	4.72	0.43	1807	4.37	0.33
worm_x3-male-fu	7.5	240.2	4.17	0.30	1806	3.93	0.30
worm_x4-female-	7.5	240.1	4.81	0.41	1806	4.47	0.33
worm_x1cd-male-	7.5	240.3	4.19	0.32	1807	3.97	0.34
worm_x3-male-isl	7.5	240.1	4.17	0.29	1804	3.93	0.26
worm_x1cd-femal	7.5	240.1	4.24	0.27	1806	4.00	0.21

Figure 3.1: Worm Tracking Setup [5]

In Neuroethology Machine Learning can be used to automatically categorize different states of behaviour or classifying different experimental groups and thus improve the process of behavioural quantification. In the following I will explain some examples of how Machine Learning can be applied to larval trajectory data.

There a several examples of research using machine learning methods to understand trajectory data of the fruit fly or other animals like penguins, rats, bats, seabirds etc.. The STEFTR [9] method uses the trajectory data of different animals to cluster their behavioural states. Using a Gaussian Mixture model one can identify different behavioral states of the animals. The model is computed based only on the trajectory data using averages (Ave) and the variances (Var) of velocity (V), bearing (B), time-differential of V (dV) and B (dB) as the basic behavioral variables.

A supervised Learning Approach was used in [22] to classify different treated experimental groups in larval learning experiments. The data set consisted of nine different experimental groups which could not be separated in their behaviour by the human observer. Using SLT-NN (Short Long Term Neural Networks) a model was build to classify the different groups.

In the literature it is often considered that the animals have two modes of behavior, "runs" and "turns". Defining those states usually relies on thresholds that are picked through experience rather than being chosen automatically although unsupervised methods for state estimation exist [9]. Others consider that the animals foraging behavior is a result of permanent oscillation of the animals head and that it is unclear if there are distinct behavioral states of the animal [6].

4 Data Processing

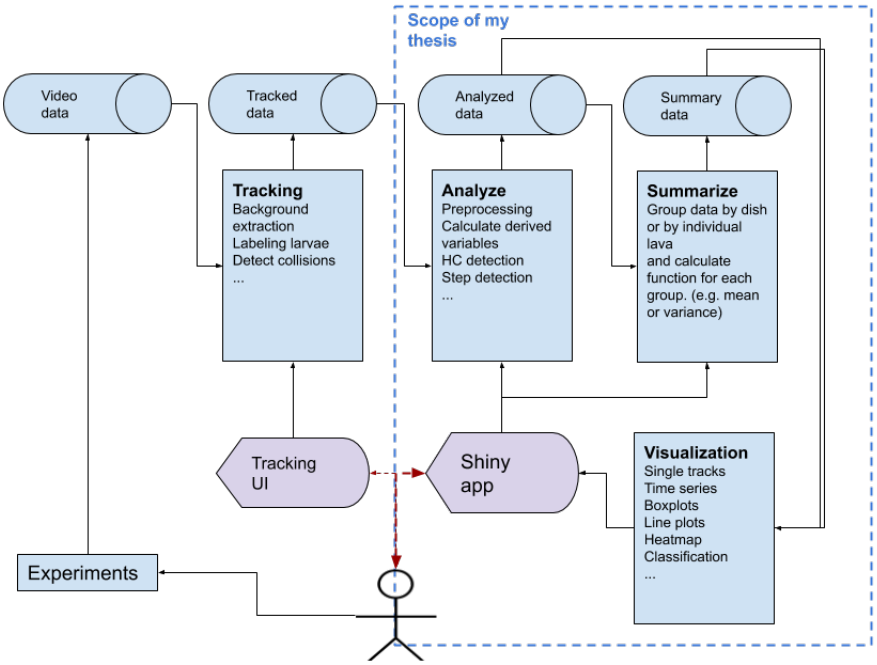


Figure 4.1: Data Processing Workflow

The data processing work flow can be seen as a "human in the loop" approach. The scientist can generate ideas based on the visualizations and improvements can be made in every step of the cycle.

In the following I want to explain how the data for the analysis is gathered and what processing steps are necessary before the visualization of the data. I want to illustrate the experimental setup and how the video data is gathered. Afterwards I want to briefly outline the tracking process and what software is used for that. In 4.1 one can see how the data is being processed: In the beginning the scientist does an experiments and collects video data. Afterwards the Tracking-UI can be

used to track the video data 4.2. When this is done, the user can analyze the data and see different visualizations.

4.1 Tracking

The videos of the experiments are recorded with a resolution of 2048px*2048px and with 16 fps. First the videos have to be processed by the tracking software. It is written in c++ and uses the open cv libraries and the cvblob library to detect the larvae using background subtraction. It tries to resolve collisions of the larvae which often occur in the experimental data. The tracking software produces CSV-files and an annotated video for each dish which contains the spines, contours, head and tail of the larvae. It can be used either via the command line or with a user interface. The user interface in 4.2 is written in python 2.7 and uses the Qt library.

Then the trials are being initialized and a progress bar shows how many trials have been processed. The data output will be used for my software project. Below one can see the folder structure of the raw data that will be used for the data analysis:

```
experiment
└── group
    └── condition
        └── trial
            ├── vidAndLogs
            │   ├── metadata.txt
            │   └── annotatedVideo.mp4
            └── track.csv
```

4.2 Data Analysis

The analysis section of the application allows the user to choose a folder with larvae trajectory data. Then the user can click on an action button to start the analysis. The current status of the analysis is shown in a console on the web page. The data which I will use for the analysis is the output of the tracking software in a table form. This raw data table contains the group, experimental condition, dish, track id, the centroid and the spine points. The analysis performs the following steps for each track in the data set:

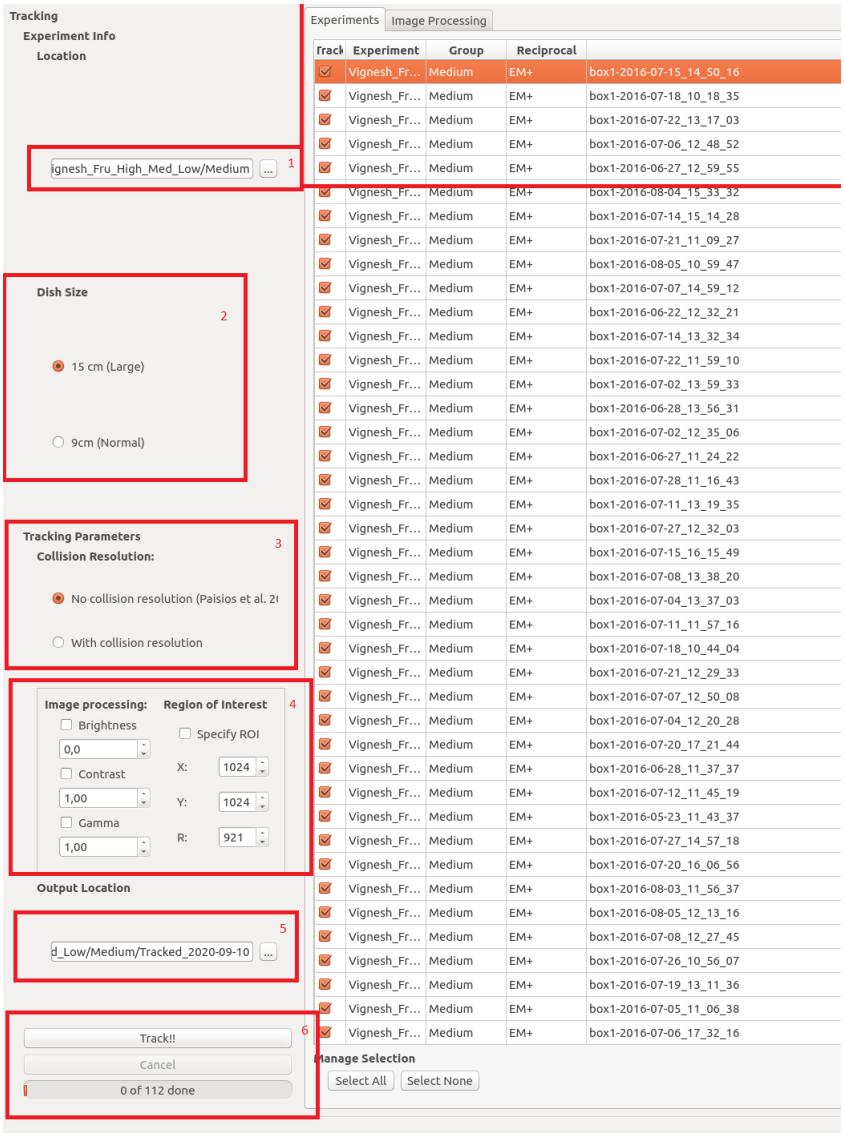


Figure 4.2: Tracking UI

First the user has to select a path where the video data is stored (1). Then the user can select the diameter of the Petri dish (2). Collisions can either be ignored or one can select tracking with collision resolution (3) The tracking software allows the user to change different parameters like a region of interest or a the contrast value for the processing of the video (4). An example frame is shown by the Tracking-UI so that the user can see how the parameters influence the quality of the video. When the user has selected a data set to analyze then he can click on a button and start the tracking. (6)

- Filter out collisions
- Rotate data points
- Convolve data points
- Calculate head and tail vectors
- Calculate midpoint speed and midpoint distance
- Calculate speed in direction to odor
- Calculate head and tail vector angular speed
- Calculate head and tail speed forward
- Filter out head and tail confusion
- Calculate bending, bearing and heading angle
- Calculate HC-variables
- Calculate Run-variables
- Calculate Preference-variables
- Calculate distance to odor

4.2.1 Preprocessing

To make the analysis easier in later stage, the data has to be preprocessed. In the following paragraphs I will describe the necessary steps for that.

First the user has to select a folder which contains the output of the tracker. The software searches then for all the CSV-files containing the tracks of each larva. It takes those CSV-files and concatenates them to a table. With this information the analysis is performed. The user can click on an action button to start the analysis.

Because the raw data is noisy, a simple moving average over all the time series is computed before further analysis.

$$P_{SM} = \frac{P_M + P_M - 1 + , ..., + P_M - (n-1)}{n} = \frac{1}{n} \sum_{i=0}^{n-1} p_{M-i}$$

In the experimental setup the odor is placed either on the right or the left side of the dish. To make the analysis of the odor preference easier the odor is rotated in

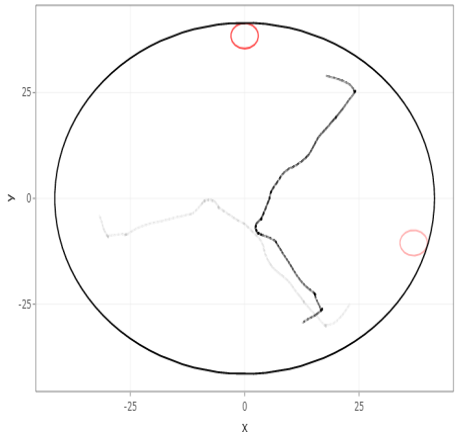


Figure 4.3: Rotated Track

that manner that the odor is always placed on top of the dish. In 4.3 one can see an example of such a rotated track.

Let v_o be the vector of the odor position. and l the length of v_{odor}.
Then $a = \begin{pmatrix} 0 \\ l \end{pmatrix}$ and $b = v_{odor}$
First the clockwise angle between two vectors a and b is calculated.

$$sign(x) = \begin{cases} -1 & x \leq 0 \\ 1 & x > 0 \end{cases}$$

$$\phi = \arctan 2(b_2, b_1) - \arctan 2(a_2, a_1)$$

$$\alpha = \begin{cases} \phi * sign(\phi 2\pi) & |\phi| \leq \pi \\ (\phi - 1) * sign(\phi 2\pi) & |\phi| > \pi \end{cases}$$

For all the columns of the x and y coordinates a rotation by angle alpha is performed. Where

$$X = x_1, ..., x_n \ , \ Y = y_1, ..., y_n$$

$$M = \begin{pmatrix} x_1 & y_1 \\ ... & ... \\ x_n & y_n \end{pmatrix}, \ M_{rot} = \begin{pmatrix} \cos\alpha & \sin\alpha \\ -\sin\alpha & \cos\alpha \end{pmatrix}$$

$$M' = (M * M_{rot}^{\mathsf{T}})^{\mathsf{T}}$$

The desired output for the rotated positions is:

$$X' = M'_{1,j} \ , \ Y' = M'_{2,j}$$

To filter out implausible behavior of the animal data is filtered based on the head and tail vector angular speed. If the absolute value of or both of them is over 180°/s the data is filtered out of the summarized data.

In the following I will list all variables that are derived from the data set. Each variable has a name, a definition, a unit and ,if necessary, a description.

4.2.2 Derived Variables

When the preprocessing of the tracking data is finished the next step is to calculate the derived measurements that are of interest for the behavior analysis. With the information various derived measurements can be calculated. The description of all variables can be found in A.

TS-variables

Like in Paisios et al. [16] [20] I defined the head and the tail vector so that the head vector starts at the 9th spine point and ends at the 11th spine point. The tail vector starts at the tail and ends at the 6th spine point.

The heading angle is defined as the angle between the head vector and the vector v_h that starts at the 9th spine point and ends at the odour source.

The bearing angle is defined as the angle between the shortened tail vector (beginning at the second spine point and ending at 6th spine point) and the vector starting from the second spine point and ending at the odor position.

The bending angle is defined as the angle between the head vector and the extended tail vector.

Also it is possible to calculate the clockwise change of the head vector and the tail vector. This is called head vector angular speed or tail vector angular speed. In 4.4 an illustration of the time series variables can be seen and in A.1 a detailed description can be found.

HC-variables

A HC is detected when the absolute head vector angular speed is over 35°. Also the tail vector angular speed has to be below 45 for the whole HC. A description of all the different HC-variables can be found in A.2.

Run-variables

A run is defined as every time point which is not a HC or 1.5 seconds before or after a HC. A step is detected when the tail speed forward is > 0.6. (local extrema) A description of all the different Run-variables can be found in A.3.

Preference-variables

The Preference-variables (see:A.4) are a measurement of how strong the animals attraction to the odor is. Some are based on the location of the animal and others on the orientation.

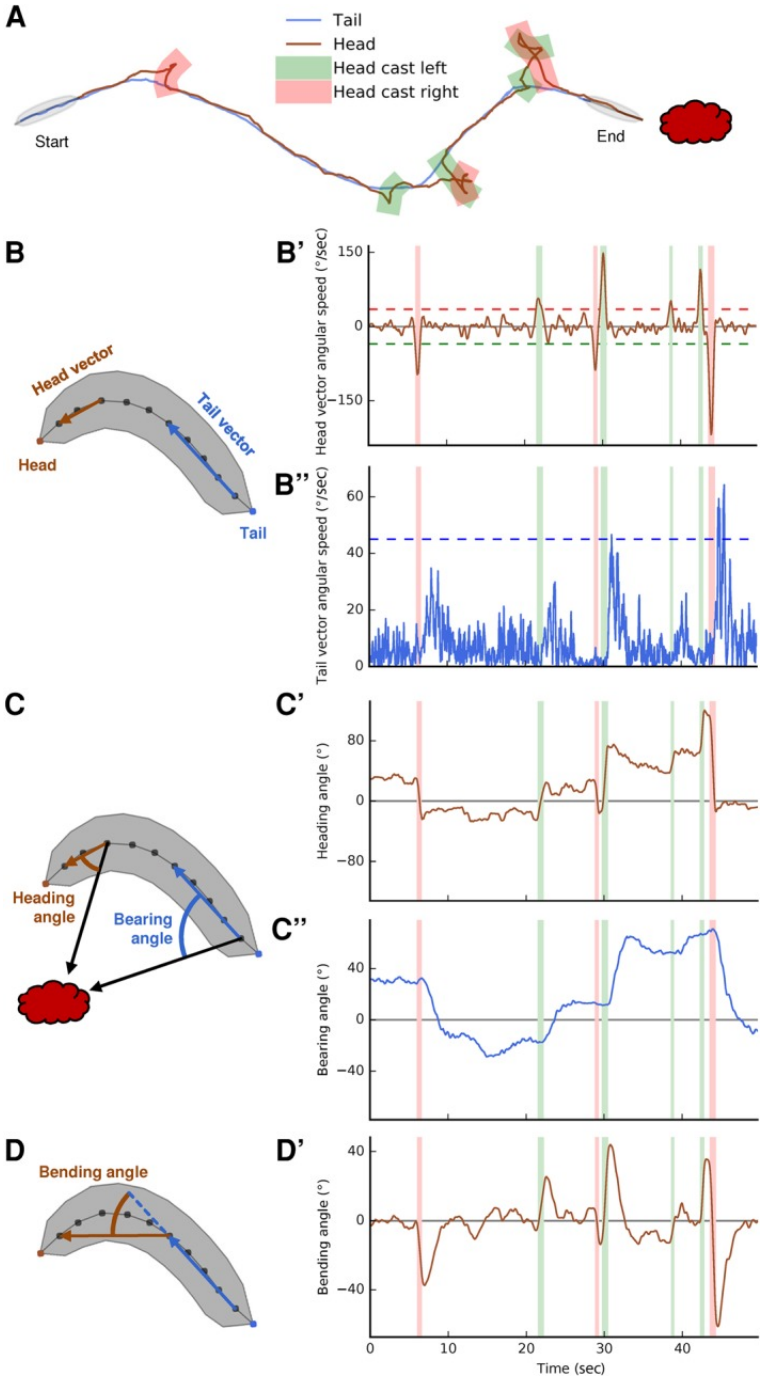


Figure 4.4: TS-variables

Figure A shows a larval track with head casts to the left are marked as green and head casts to right as red. Figure B shows the head vector and the tail vector. B' shows an example of head vector angular speed and B" the tail vector angular speed. D' shows the bending angle. [16]

5 Shiny App

The following chapter deals with the Shiny App. The Shiny App will make uploading and analysing the data possible and also will be used for the visualization of the data. The advantage of having a dashboard is that for the visualization no programmer needs to be involved and the researcher can choose a data set from the computer.

5.1 Requirements

For creating an interactive web dashboard the requirements need to be set. The analysis should include different visualization approaches. With the software one should be able to visualize single tracks and its time series data. Also it should enable to show the path of each larvae on the Petri dish. The user should be able to produce a lot of derived measurement including transformed and filtered data. A list of all requirements can be seen in B.

5.2 Implementation

For the implementation I am using the **R** programming language [17]. R is a language which supports mathematical and statistical programming. It is designed for statistically analysing and visualizing data sets. Also it supports linear and nonlinear modeling, statistical tests, time-series analysis, classification, clustering and more. R is widely used in research and teaching, but can be used for business applications and prototyping.[18] One of the reasons why R is so popular in science is because of its easy to create and high quality graphics. I use R for this project because it fit well to the focus of the project. Besides R having the ability to cope with time series data very well it is furthermore good for designing interactive data analysis.

RStudio is an R IDE which has a lot of useful functions to make the programming more easy and efficient. It supports git and **github** and therefore easy version

control. A special feature which lets RStudio stand out is that one can do so called Profiling. Profiling helps to understand which line of code uses how much memory and computing time. This can really help to improve the performance and get an insight what may make the application slow.

I am using the **Shiny** package [19] in the application, because it allows to create user web dashboards for visualization. RStudio and Shiny are open source, so anyone can use it and work with the application. Also the programming language R is made for statistical analysis. The **tidyverse** package library allows to use data selection with the dplyr package and plotting with ggplot2. To improve the plots allowing mouseover interaction I am using the plotly library. The option to upload the application to a webpage from Shinyapps allows the user to scale the performance and memory of the application. A different method is hosting the app on the Network or offline on a single system. This means it is generally very flexible in its setup and users from different labs can use the system.

The application has two different functions, the server and the user interface. The server gets input information based on the UI-elements which are defined in that function.

ggplot2 is used for the plotting. It has a wide range of different visualization techniques and is the most popular in R. I am using ggplot2 because of its easy readable grammar and support for a large variety of plots. The R package **dplyr** is a tool for processing data in a pipeline and it can help with various data operations and transformations. [12] I use the package because its grammar is easy to learn and read and also because of its good performance. On top of that I am using **data.table** because it can operate fast on large data sets.

5.3 User Interface

The user interface is programmed with the R Shiny package [19]. In 5.1 one can see the dashboard. A hideable sidebar is used to help navigate through the different tabs.

5.3.1 Choose Data and Processing

The user needs to be able to **load and analyze the data** before the visualization. This section of the application ensures that the user can automatically analyze folders of raw data. Also it is possible to upload raw data as CSV instead of a the folder.

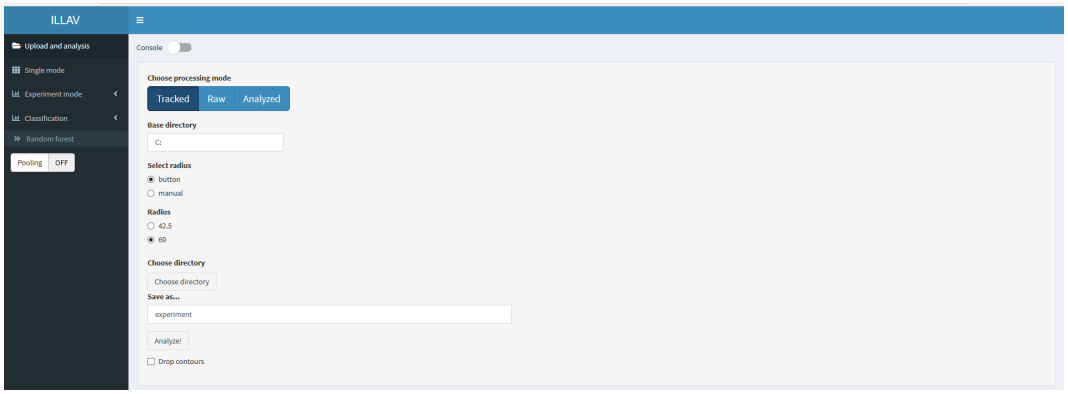


Figure 5.1: The Shiny Dashboard

When the user has uploaded the raw data he or she can choose the radius of the Petri dish and start the analysis and see the progress of the analysis process. For implementing the UI I use the "reactiveUI" function in shiny which creates the possibility to let the user interface change based on a certain condition. This way the UI adapts to the corresponding processing mode. I used the package "shinyFiles" to deal with the data selection. For choosing a folder I used the "shinyDirChoose" function and for choosing files the "shinyFileChoose" function. Furthermore there is the possibility to upload already analyzed data and then directly see the analysis plots.

5.3.2 Single Mode

The single mode facilitates the visualization of different variables for a single larva. In the previous analysis it was only possible to manually visualize single larvae behavior. It is now possible to click on a table entry and immediately see the trajectory of the animal. Also one can plot the different time series variables of the track. This allows the user to see the data in detail and see what the individual larva is doing in the test.

Using the **track selection** 5.2 one can see a table where each entry is a track with its respective id, name trial, condition and group. By clicking on an entry the user can see the trajectory of the track on the right side of the panel. I used the package "DT" for showing the data. This allows selecting rows and sorting the data. When the selected row changes one can see the trajectory on the right side is changing. In the bottom of the panel there is a time slider. Using the "sliderInput" function of shiny one can filter the data from a start point to an end point.

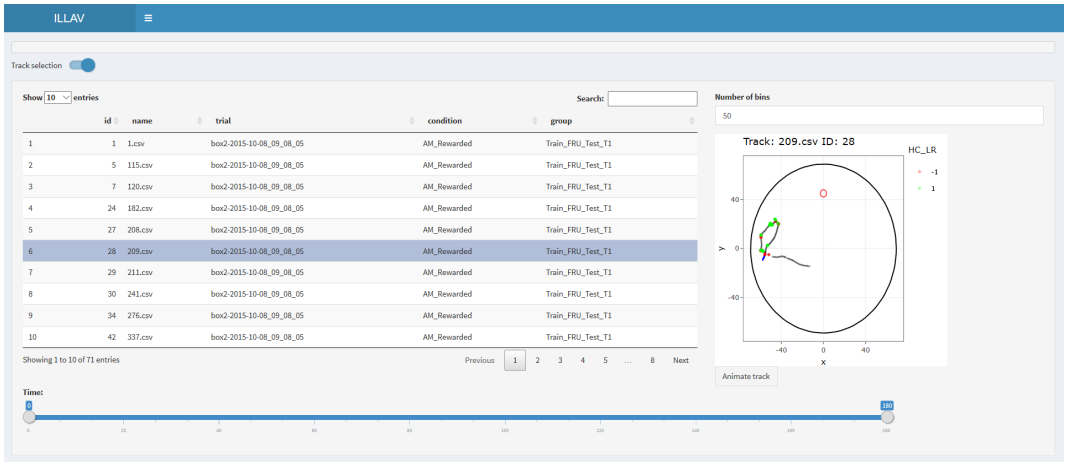


Figure 5.2: Track Selection Panel

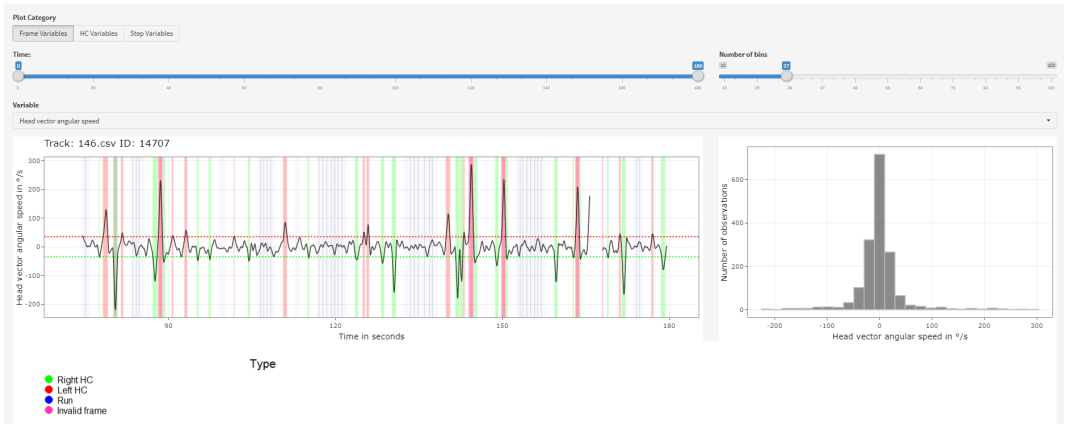


Figure 5.3: Time Series Plot

In the **time series panel** 5.3 the user can choose between TS-variables, HC-variables and Run-variables. For the TS-variables the user can see a line plot and for the HC variables and Run-variables a scatter plot. The user can select a TS-variable and plot it for the selected track. Using a slider the user can filter the time interval of the data. A left head cast is coloured in red and a right head cast is coloured in green and a run is colored in blue. On the right side of the panel a histogram of the value is shown.

5.3.3 Experiment Mode

In the experiment mode the user can see summarized values like the mean, median or variance for each individual or for each Petri dish. Boxplots visualize the distribution of the different groups and conditions. Also it is possible to calculate binned line plots of some of the variables. This is needed to see if e.g. a certain HC-variable changes with respect to the bearing angle or if the preference changes over time. Here again the different groups and conditions are shown by different lines.

Below this panel one can see the summarized data and also download it.

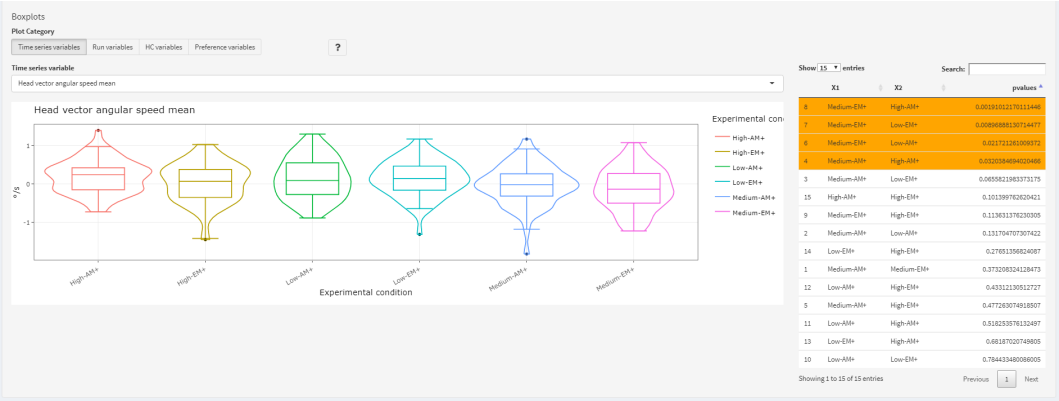


Figure 5.4: Violin plots

Violin plots of head vector angular speed using ggplot2 [26] and plotly. [21]

Boxplots are shown for each group and condition 5.4. With a drop down menu the user can select the desired variable. The user can select between the TS-variables, HC-variables, Run-variables and Preference-variables. When the user selects a category the drop down menu is being switched using the renderUI function of shiny. Also a table of the pairwise Mann-Whitney-U-Test are shown. Significant results are highlighted in the table in orange. This can help the user to derive knowledge from the data. The plotly boxplot allows the user to go with the mouse over the boxplot and see the median and the quantiles of the box.

5.3.4 Binning

Binning is used so that the data is divided into intervals of equal length using two variables of the same dimension and then a function is performed for each interval. This can help to understand the relation between those variables. In the **line plot** section the user can do binning based on three variables:

- Time
- Bearing angle
- Distance to odour

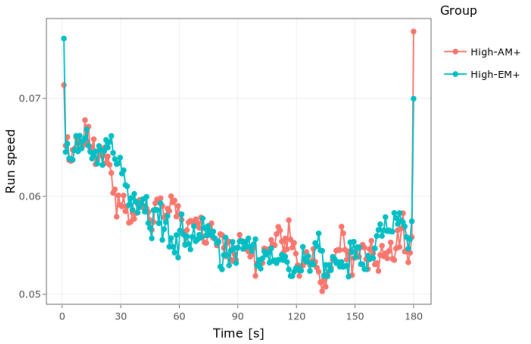


Figure 5.5: Line Plot I.

This should help the user to understand how certain variables of the larvae are changing over time, bearing and distance. As an example I show a line plot of the run speed of a sample data set.

On can see that run speed is decreasing with the time. Critical for the visualization is finding the right number of intervals. One problem is that when there are not enough points that one might overlook changes during a certain period of time. On the other hand when picking to many points the line is influenced by the variability of the distribution. One solution for this is smoothing the plot using a mean average.

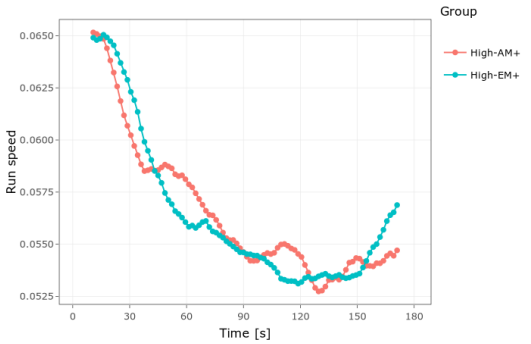


Figure 5.6: Line Plot II.

In figure 5.5 one can see a line plot figure without smoothing. One can see that in the beginning and the end there are very high values in run speed. It is difficult to interpret what that means. Another figure 5.6 shows a smoothed plot where a moving average with 11 time points was chosen. Here one can see differences between both lines, but one can not be sure if those differences are actually significant. To overcome the invisible variability one can plot the confidence interval. In 5.7 one can see such a plot.

In addition to this form of binning **2-dimensional binning** can be helpful for understanding dependencies between variables. A color coded heat map can be used to visualize this data. In 5.8 one can see the HC reorientation with respect to the distance to the odor and the bearing angle for two experimental groups. In this figure we can observe that there a a lot of little peaks in the color code and it is hard to see a pattern. Using kernel density estimation a filter is smoothing the data. One can see that in 5.9 applying a higher bandwidth can lead to visible patterns in the graph.

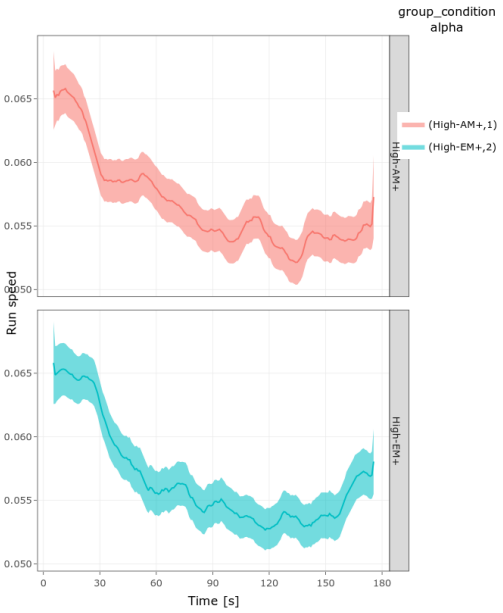


Figure 5.7: Line Plot III.

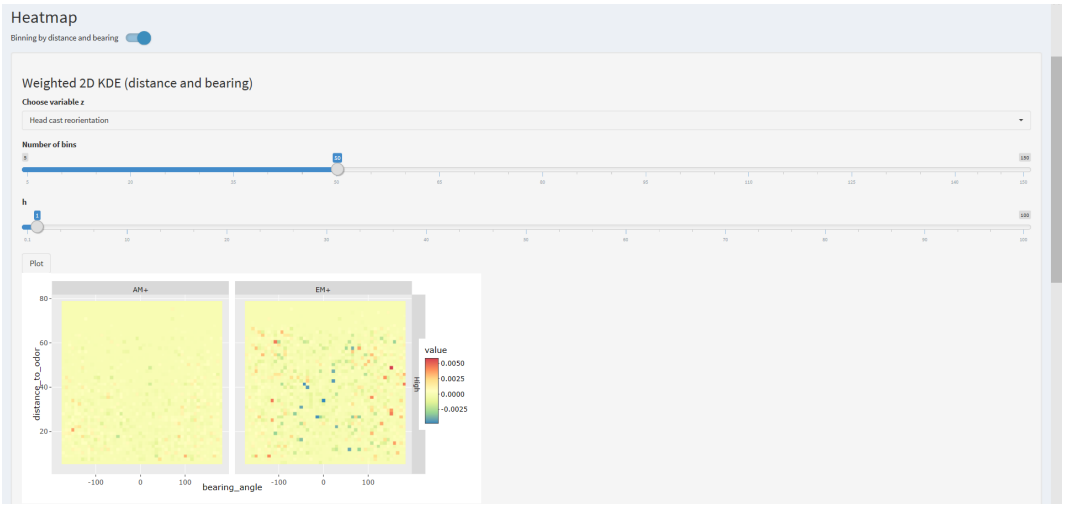


Figure 5.8: 2-d Binning Without Filtering

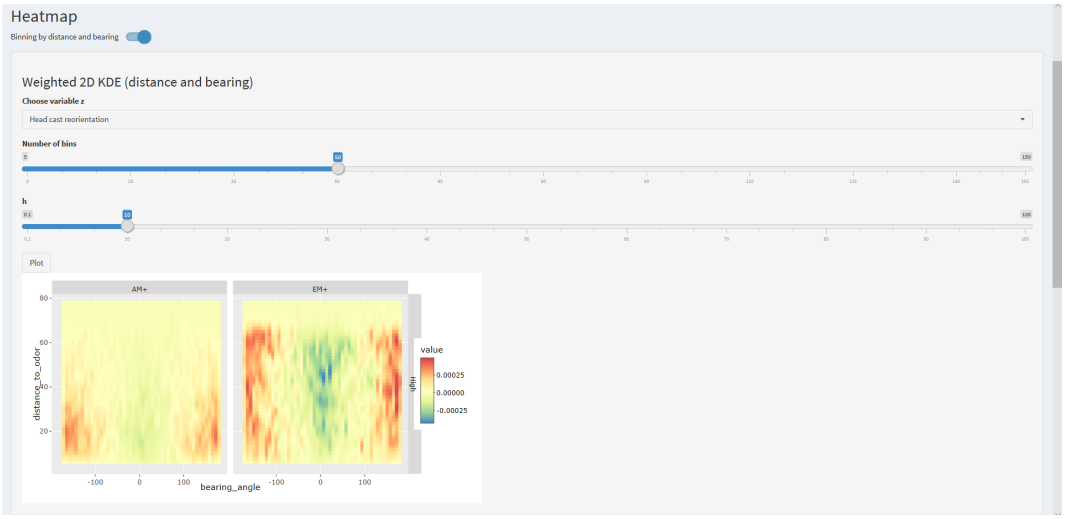
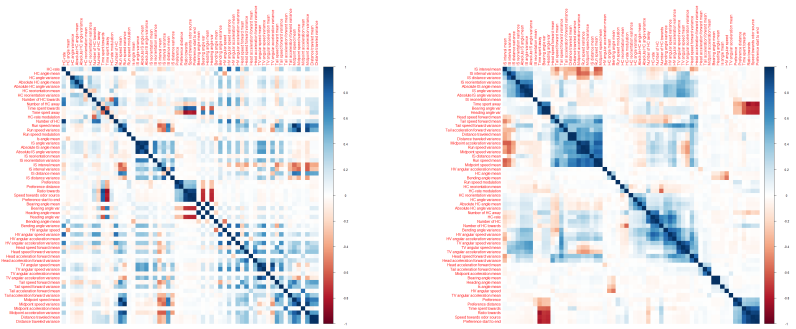


Figure 5.9: 2-d Binning With KDE

6 High-dimensional Visualizations and Models

Because the number of behavioral variables is too large to look at all combinations of variable selections high-dimensional techniques are required. Such techniques could be Parallel Coordinates, Correlation Matrix or Clustering. High-dimensional data sets also allow building predictive models e.g. a Decision Tree Classifier or Random Forest Classifier.

6.1 Correlation Matrix



(a) Not Sorted **(b)** Sorted With Hclust

Figure 6.1: Correlation Matrices

In 6.1 a correlation matrix of all the derived measurements can be seen. In this case the correlation matrix is not sorted, but sorting the table can improve the understanding of the matrix. In C.1 the correlation matrix is sorted using hierarchical clustering. Here one can observe that there are multiple clusters of correlation. For determining the optimal group of clusters plotting the silhouette coefficient for different numbers of clusters can help (see: C.2). According to the silhouette

coefficient 7 is the optimal number of clusters. In C.3 a dendrogram shows the hierarchical clustering.

6.2 Decision Trees

As an example I want to show how a decision tree looks like when visualized (see: C.4). Because a decision tree model changes with how one decides to split the data set I am using a Random Forest Classifier to evaluate the performance of the classification.

6.3 Random Forest

I want to test how good a classifier can separate the paired group from the unpaired group. For this I am training a random forest classifier model on a test data set. As a measure for the feature importance I want to look at the mean decrease of the Gini-coefficient. I am training a random forest classifier on the values per dish and per individual larvae. For evaluating the model cross validation is used 3-times repeated 10-manifold cross validation. The confusion matrices can be seen in C.5 and C.6. The variable importance can be seen in C.7 and C.6.

6.4 U-tests

To test the distributions for significant differences Wilcoxon-Mann-Whitney-Test [13] were done for all combinations of experimental groups and all variables. These plots can be found in C.9, C.9, C.10, C.11, C.12, C.13 and C.13.

7 Evaluation

7.1 Qualitative Evaluation

For the qualitative evaluation I want to first gather user feedback regarding the Shiny app and the different visualization approaches. Afterwards I want to look at some of the visualizations in more detail using a case study. With the case study I want to try to answer some of the research questions that lead to the ideas.

7.1.1 User Feedback

The feedback from the user helps to evaluate the quality of the visual analysis. First a field study was planned. We wanted to ask the users to evaluate the system using a survey. Due to time limits the user feedback is limited to the feedback that I got in the seminar and from my supervisor.

First of all in the seminar it was mentioned that choosing single larvae in table and directly see their trajectory and different time series helps to understand the data in more detail. Drop down menus and buttons would help the navigation and decrease the search time for the desired figure. Also it was remarked positively that the web-page can potentially be hosted on a server.

I also got feedback from my supervisor who is already using the app for the analysis of his data. He said that having the visualizations on a web page makes it easy to visualize his data. He thinks that the dashboard is broadly applicable and offers a wide range of different visualization approaches. The exploration and analysis has been improved with this interactive tool. Also he mentioned that it is from special interest what variables can be seen as important to distinguish certain experimental groups.

7.1.2 Case Study: Naive Larvae vs. Learned Larvae

For a case study I am choosing an experiment where there are 4 groups of animals. One group is the paired group. Here the odour has been paired with a sugar reward. In the unpaired group the sugar is not paired with a reward. Here it is important to mention that these both groups learn. The paired group learned that the odor predicts reward and the unpaired group learned that the odor predicts the absence of a reward. The third group is containing naive larvae which where not exposed to the learning treatment. The fourth group is containing naive larvae and in the test there is a bitter tasting substance present. In the following evaluation I will refer to C.5, C.6, C.7, C.8 and C. The groups have the following names:

- Naive: "Naive_Test_PUR-EM_Rewarded"
- Naive on quinine: "Naive_Test_QUI-EM_Rewarded"
- Paired: "Train_FRU_Test_T1-AM_Rewarded"
- Unpaired: "Train_FRU_Test_T1-EM_Rewarded"

Correlation Analysis

The correlation matrix in C shows that there are groups of variables which are highly correlated. There seem to be a number of dependent variables. Some correlations are trivial, but there are also some correlations that can not be derived by the calculation of the variables. Those are the variables that are especially of interest for the researcher. It is trivial to see that e.g. the variance of a certain variable is correlating with the mean or that the preference variables are correlated. But there are also examples that are not trivial e.g. that the HC-rate-modulation is correlated with the HC reorientation.

Is there significant difference in the distributions?

To test which groups differ significantly in the distributions of the derived measurements I performed U-tests for all combinations of groups and variables. This has been done for all combinations of groups of the data set of the case study. The figures can be found in C

The results show that in all combinations there are significant differences in some of the variables. When comparing both naive groups one can see that they differ mainly in speed and HC dependent values. The paired and the unpaired group are

differing mainly in the odor dependent values. Also it made clear that it is worth to look at the values of individual larvae because it increases the sample size drastically meaning increasing the statistical power can reveal more differences. One side effect of the individual measurements is that it also increases the variability of the distributions.

Classification Results

The confusion matrices for the classification results can be found in C.5 and C.6. The overall accuracy of the classification results for the data set per dish is about 0.74. That is above the chance level which would be 0.25. In the confusion matrix one can see that the different groups are all similar accurate. The accuracy of the data set per individual is 0.51. Here one can see that the model can predict the naive group on quinine the best. Overall the results are promising and should be evaluated further using different data sets. The random forest seems to be a useful method for detecting subtle changes in behavior. In the U-test plots in C one could see that the data set per individual seems to be easier to distinguish than the data set per dish. In accuracy of the random forest albeit one can see that the classification of the data set per individual has a lower accuracy than the other. Concluding from this one can say that variability and statistical power have both influence on the results.

What variables are most important?

Looking at the variable importance of the random forest in C.7 and C.8 one can see that the most important variables for the classification using the per dish data set are the preference distance and the bending angle variance. For the individual data set the bending angle variance is the most important followed by four variables that are similar important:

- Tail acceleration forward variance
- IS distance mean
- Preference distance
- Tail speed forward mean

Interesting is here that the values bending variance and tail acceleration forward variance have not been investigated before, but still seem to be important for distinguishing the animals behavior. This calls for future research into these behavioral modifications.

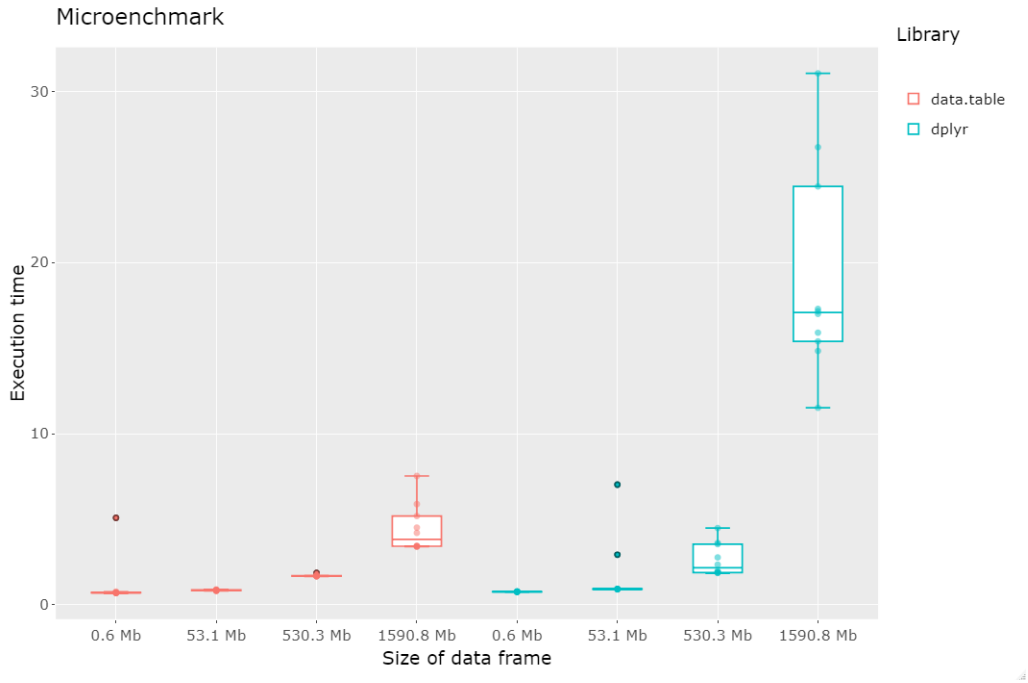


Figure 7.1: Performance Benchmark

7.2 Quantitative Evaluation

The data set which contains all the trajectories are sometimes >10GB and contain more than 10 million rows. This might be critical for the performance especially when the data set is larger then the RAM.

I want to find out what the fastest way is to calculate the derived variables using the "create_summarized_analysis" function. So I implemented a data.table and a dplyr version of the summarized analysis.

For my performance analysis I used the package "microbenchmark" to see how the size of the data set is impacting the execution time of the summary function. First I sample different sized chunks of data for 1k, 100k,1000k and 3000k rows. Then I create a microbenchmark for data.table and dplyr. Afterwards I am plotting the data frame using ggplot2 and plotly.

The boxplot in 7.1 shows that the execution time of data.table is similar for data sets with less than 500Mb. It shows that data.table is much faster in calculating the summarised analysis and therefore should be used by the system.

8 Conclusion

On the basis of the results of this research, it can be concluded that there is a large number of behavioral variables to analyze and it is promising to continue working on the quantification of the behavior.

The main goal of the thesis was to create an interactive dashboard to explore the larvae locomotion data set. First I explained the basics of *Drosophila* locomotion analysis and introduced some graphical statistics which are used to quantify the behavior. Also I set the problem into to a context by describing some other quantification methods that are used in the research. The requirements for the project were set and a dashboard was implemented that fulfills all the requirements. Furthermore high-dimensional visualizations and Machine Learning was used to analyze the data sets. In the evaluation I showed an example data set and explained the visualizations I used in the analysis of the data. In a quantitative study I evaluated the performance of the analysis and I could improve the performance in the process of the development.

The project had many different challenges in the field of computation, performance, visualization and user interface development. One issue was keeping the performance high, because sometimes the memory of the data sets are higher than the RAM of the laptop I was working with. I found a way to cope with this issue using libraries like dplyr and data.table. Nevertheless there is room for improvement. One possibility is decomposing the large table into different smaller tables. Also it would be helpful to use a database in the future.

Regarding the results of the visualizations it can be concluded that a random forest classifier can be used to distinguish certain experimental groups. This method can be used to reveal the importance of behavioral variables. It seems promising to continue the research and find appropriate methods for visualizing the data set and find important behavioral variables.

Bibliography

[1] Cookbook for r - plotting distributions. http://www.cookbook-r.com/Graphs/Plotting_distributions_(ggplot2)/. Accessed: 2021-20-01.

[2] Cookbook for r - scatterplot. http://www.cookbook-r.com/Graphs/Scatterplot/. Accessed: 2021-20-01.

[3] ggplot2 : Quick correlation matrix heatmap - r software and data visualization. http://www.sthda.com/english/wiki/ggplot2-quick-correlation-matrix-heatmap-r-software-and-data-visualization. Accessed: 2021-20-01.

[4] Ryan P. Adams. Hierarchical clustering. *Princeton University*, 2019.

[5] Boanerges Aleman-Meza, Sang-Kyu Jung, and Weiwei Zhong. An automated system for quantitative analysis of drosophila larval locomotion. *BMC Developmental Biology*, 15, 2015.

[6] Barbara Webb Antoine Wystrach, Konstantinos Lagogiannis. Continuous lateral oscillations as a core mechanism for taxis in drosophila larvae. *Elife*, 2016.

[7] Leo Breiman. Random forests. *Machine Learning*, 2001.

[8] Jason C. Caldwell, Matthew M. Miller, Susan Wing, David R. Soll, and Daniel F. Eberl. Dynamic analysis of larval locomotion in drosophila chordotonal organ mutants. *Proceedings of the National Academy of Sciences*, 100(26):16053–16058, 2003.

[9] Shuhei Yamazaki et al. Steftr: A hybrid versatile method for state estimation and feature extraction from the trajectory of animal behavior. *frontiers in Neuroscience*, 2019.

[10] Michael Galarnyk. Understanding boxplots. *towards data science*, 2018.

[11] Tanimura T Thum AS Gerber B, Stocker RF. Smelling, tasting, learning: Drosophila as a study case. *Results Probl Cell Differ*, 2009.

[12] Hadley Wickham, Romain François, Lionel Henry, Kirill Müller . *dplyr: A Grammar of Data Manipulation*, 2020.

[13] John H. McDonald. *Handbook of Biological Statistics*. 2014.

[14] Tom M. Mitchell. *Machine Learning*. McGraw-Hill Science/Engineering/-Math, 1997.

[15] Martin Krzywinski Naomi Altman. Clustering. *Nature methods*, 2017.

[16] Pamir E Paisios E, Rjosk A and Schleyer M. Common microbehavioral "footprint" of two distinct classes of conditioned aversion. *Cold Spring Harbor Laboratory Press*, 2017.

[17] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2013.

[18] David Robinson. The impressive growth of r, 2017.

[19] RStudio, Inc. *Easy web applications in R.*, 2013. URL: http://www.rstudio.com/shiny/.

[20] Michael Schleyer, Samuel Reid, Evren Pamir, Timo Saumweber, Emmanouil Paisios, Alexander Davies, Bertram Gerber, and Matthieu Louis. The impact of odor–reward memory on chemotaxis in larval drosophila. *Cold Spring Harbor Laboratory Press*, 2015.

[21] Carson Sievert. *Interactive Web-Based Data Visualization with R, plotly, and shiny*. Chapman and Hall/CRC, 2020.

[22] Jonathan Spiegel. Classification of differently trained larvae based on changes in their trajectories using artificial neural networks. Master's thesis, Otto-von-Guericke Universität Magdeburg, 2018.

[23] Michael Thane, Vignesh Viswanathan, Tessa Christin Meyer, Emmanouil Paisios, and Michael Schleyer. Modulations of microbehaviour by associative memory strength in drosophila larvae. *PLoS ONE*, 2019.

[24] André Fiala Thomas Riemensperger, Robert J Kittel. Optogenetics in drosophila neuroscience. *Methods Mol Biol.*, 2016.

[25] Gerber B Thum AS. Connectomics and function of a memory network: the mushroom body of larval drosophila. *Current Opinion in Neurobiology*, 2019.

[26] Hadley Wickham. *ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag New York, 2016.

List of Figures

A Variable Tables

Variable	Unit	Definition	Description
Spine points	mm	Position of spine points 1-12 relative to middle of the dish.	The spine points are not based on the real anatomy of the larvae.
Contour points	mm	Position of contour points 1-12 relative to middle of the dish.	
Midpoint speed	mm/s	Speed of the sixth spine point.	
Head and tail vector		The head vector is defined as the vector from spine point 8 to spine point 11. The tail vector is defined from spine point 2 to spine point 6.	The head and tail vector are needed for calculating the bearing angle, the heading angle and the bending angle.
Bending angle	°	The bending angle is defined as the angle between head vector and the extended tail vector.	The bending angle is needed for calculating the HC and IS angle.
Bearing angle	°	The bearing angle is the angle between the tail vector and the vector from the animal towards the odour.	When the absolute bearing angle is $< 90°$ the animal is moving towards the odor. Otherwise it is moving away from the odor.
Heading angle	°	The heading angle is the angle between the tail vector and the bearing towards the odour.	The heading angle is needed for calculating the HC and IS reorientation.
Head/tail vector angular speed	°/s	The head/tail vector angular speed is the clockwise (+) or counterclockwise (-) change of the head vector.	

Head and tail speed forward	mm/s	The head/tail speed forward is the speed in parallel to the head vector.	
Distance traveled	mm	Length of the trajectory path of the animal.	
Distance from starting point	mm	Euclidean distance from the starting point of the animal.	

Table A.1: Table of TS-variables.

Variable	**Unit**	**Definition**	**Description**
HCs		Number of HCs	
HC-rate	Hz	Number of HCs per second	
HC angle	°	Bending angle at the end - bending angle at start	
Absolute HC angle	°	absolute of HC angle	Size of the HC
HC reorientation	°	Heading angle at end - heading angle at start	How much does the animals head is reorienting towards the odor source.
HC rate modulation		(#Hcs towards - #HCs away)/#HCs	Positive when more HCs towards the odor and negative when more HCs away from the odor.

Table A.2: Table of HC-variables.

Variable	**Unit**	**Definition**	**Description**
Run speed	mm/s	Midpoint speed during runs	
Run speed modulation		(Run speed towards) - (Run speed away) /run speed	
IS angle	°	Bending angle difference between two steps	
Absolute IS angle	°	Absolute bending angle difference between two steps	

IS reorientation	°	Difference in heading angle between two steps	How much does the animals head is reorienting towards the odor source.
IS distance	mm	Distance between steps	
IS interval	s	Time passed between steps	

Table A.3: Table of Run-variables.

Variable	**Unit**	**Definition**	**Description**
Preference		(Time spent on odor side)/time	
Preference distance		Distance to odor (scaled from -1 to 1)	
Ratio towards		(Time spent towards odor)/time	
Preference distance start to end		(Distance to odor at start) - (distance to odor at end) (scaled from -1 to 1)	

Table A.4: Table of Preference-variables.

B Requirements

B.1 Functional Requirements

- **F1** The user should be able to visualized all TS-variables, HC-variables and Run-variables.
- **F2** The user should be able to visualized histograms of all all TS-variables, HC-variables and Run-variables.
- **F3** The user should be able to see a table of all tracks with respective name, dish, condition and group.
- **F4** The user should be able to select a certain track which is listed in the track table.
- **F5** The user should be able to visualize TS-variables, HC-variables, Run-variables and Preference-variables using a boxplot.
- **F6** The user should be able to visualize all variables as boxplot of different groups using a boxplot.
- **F7** The user should be able to visualize all variables as line plot of different groups binned per bearing angle interval using a line plot.
- **F8** The user should be able to visualize those variables as line plot of different groups binned per time interval using a line plot.
- **F9** The user should be able to summarize data per Petri dish.
- **F10** The user should be able to summarize data per track.
- **F11** The user should be able to change the statistical parameter to mean/-median/variance .
- **F12** The user should be able to filter the data by bearing angle, distance to odor and time.
- **F13** The user should be able to look at scatterplots of summarized variables.

B.2 Non-functional Requirements

- **N1** The software should be easy to use.
- **N2** The calculations should be fast.
- **N3** The calculations should involve low memory.
- **N4** The software should be maintainable.

C Figures

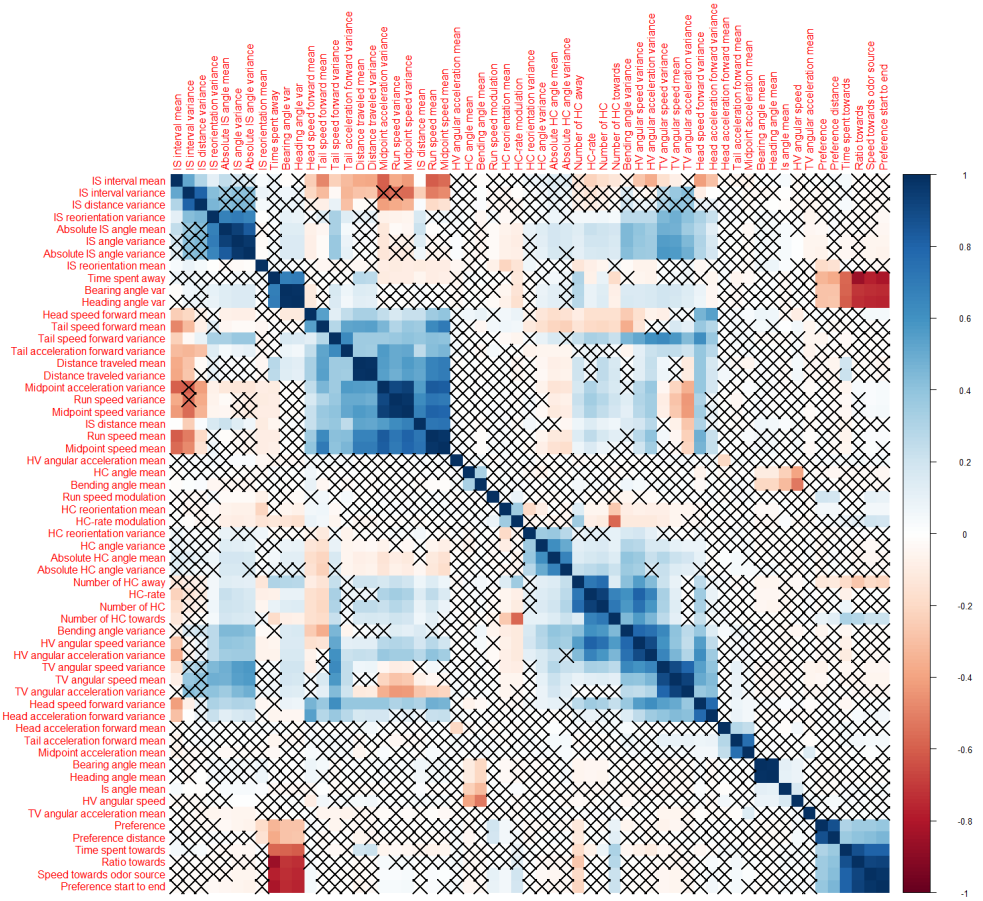


Figure C.1: Correlation Matrix

Correlation matrix of the complete data set. Blue values indicate positive correlation and red values negative correlation.

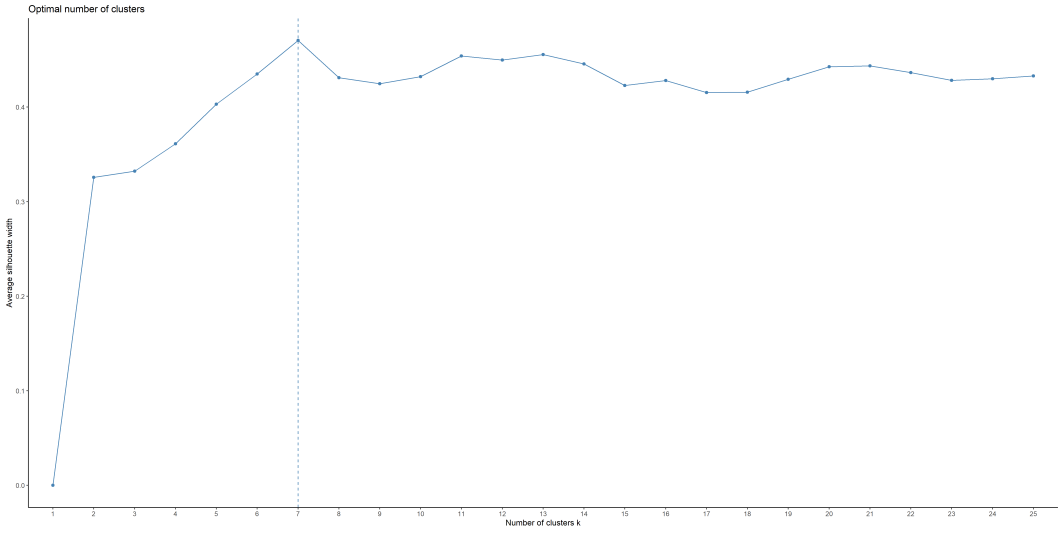


Figure C.2: Silhouette coefficient

Silhouette coefficient for different numbers of clusters.

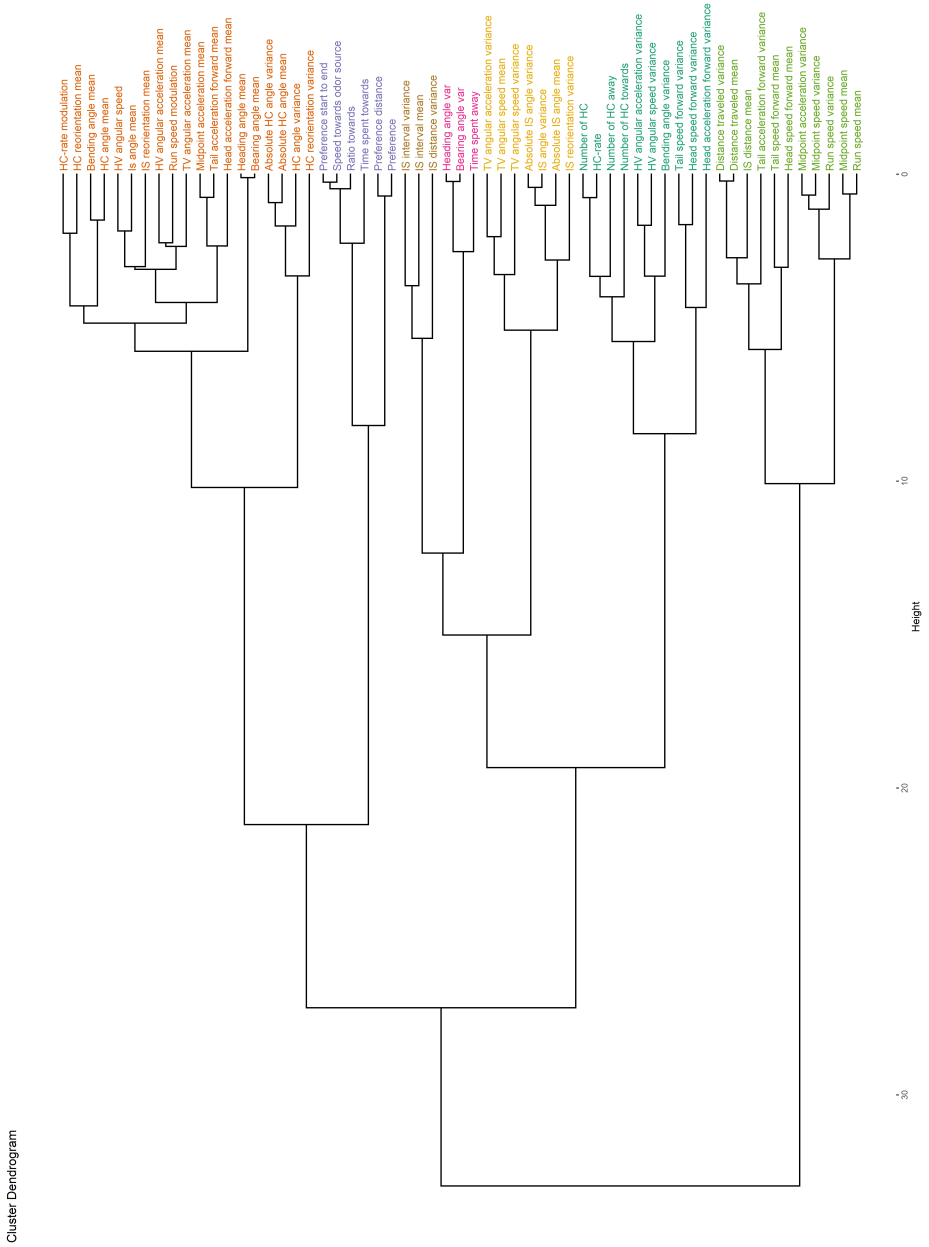


Figure C.3: Dendrogram

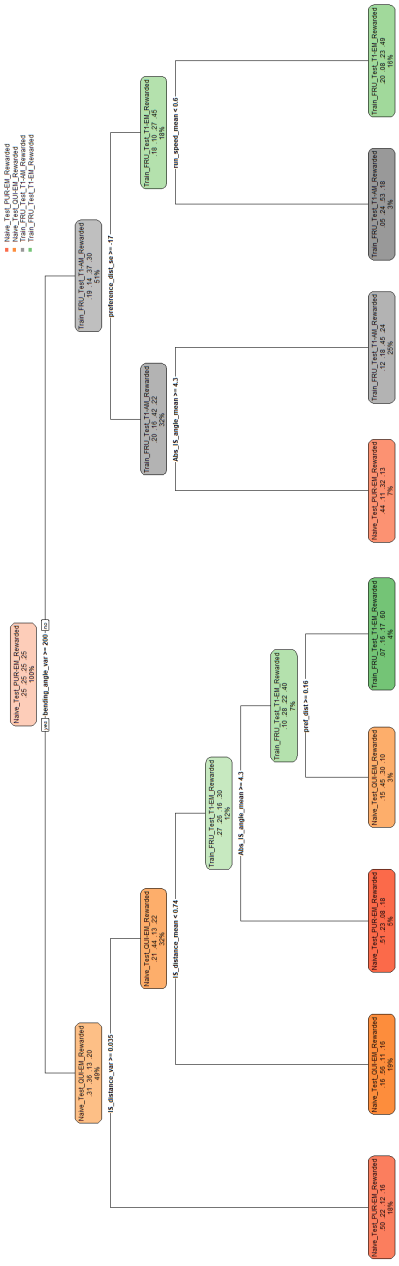


Figure C.4: Decision tree

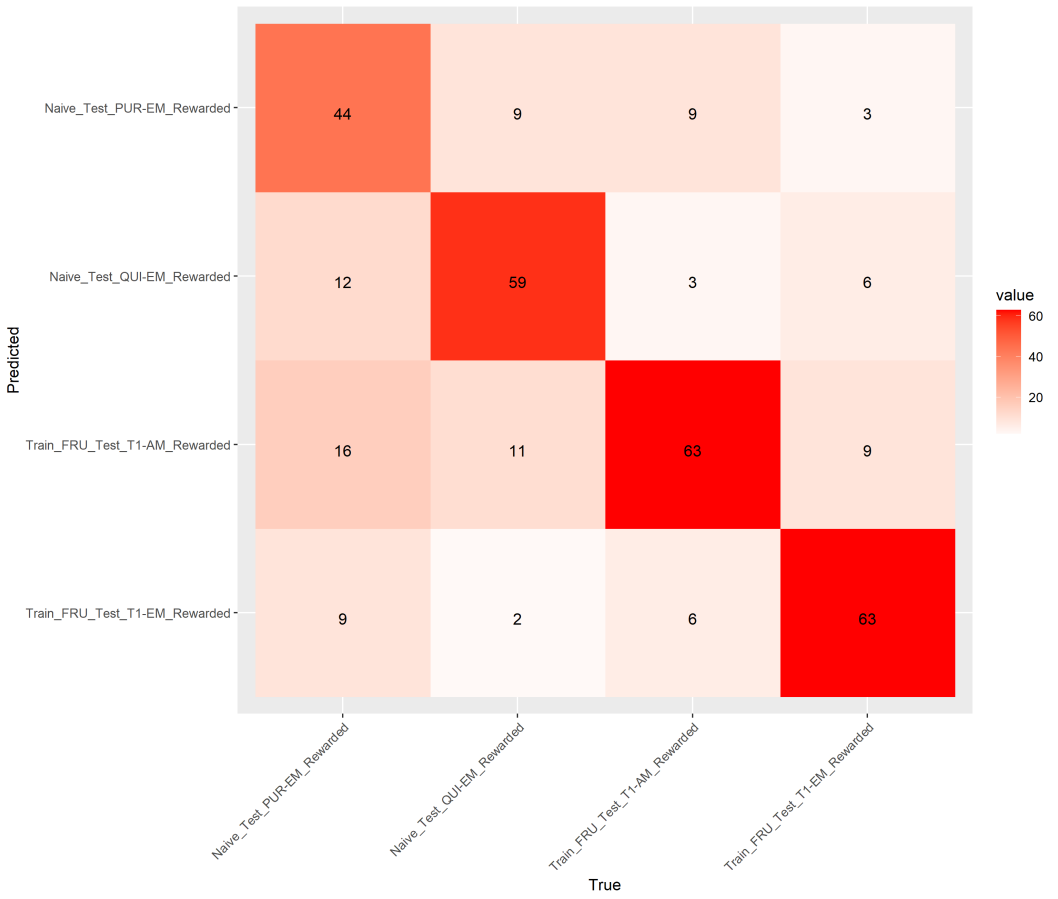


Figure C.5: Confusion Matrix - Per Dish

Figure C.6: Confusion Matrix - Per Individual

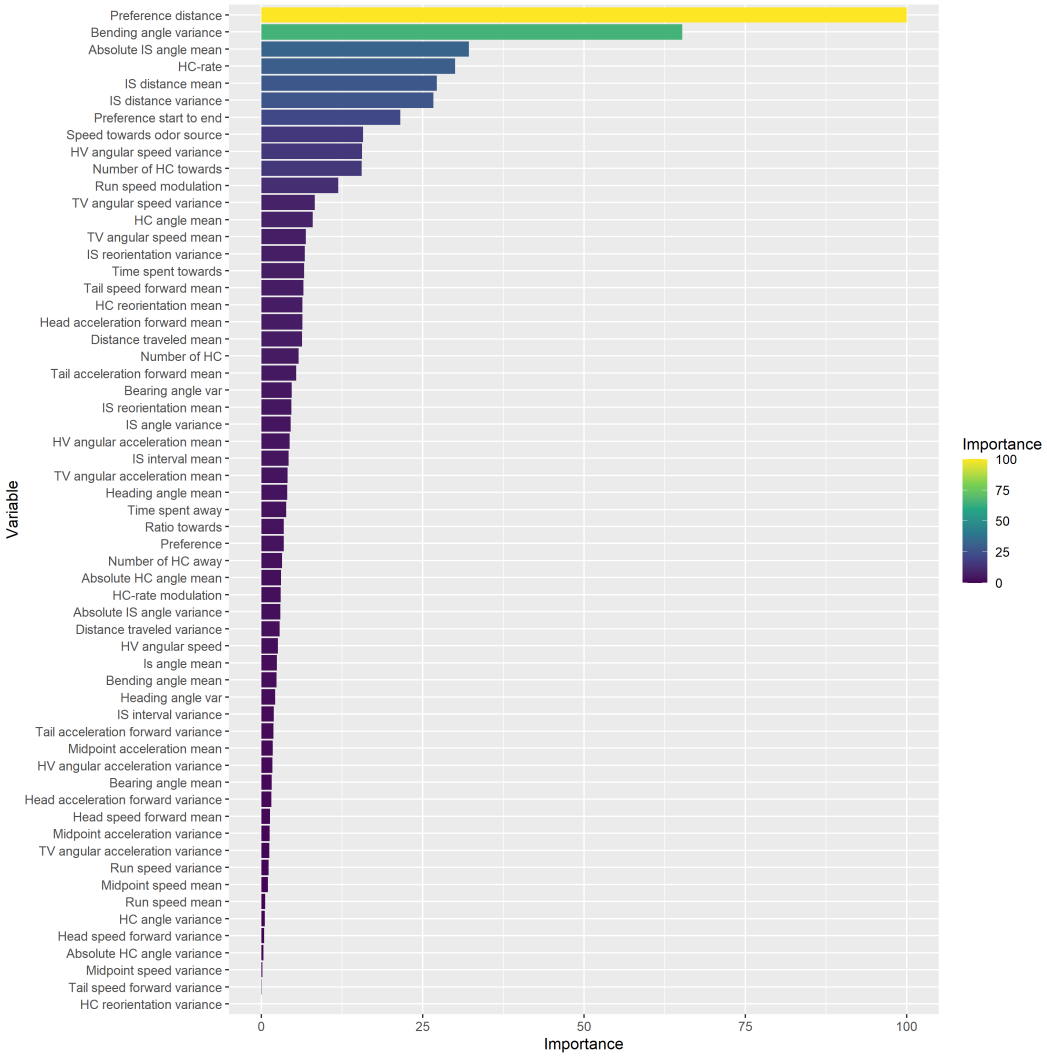


Figure C.7: Variable Importance - Per Dish

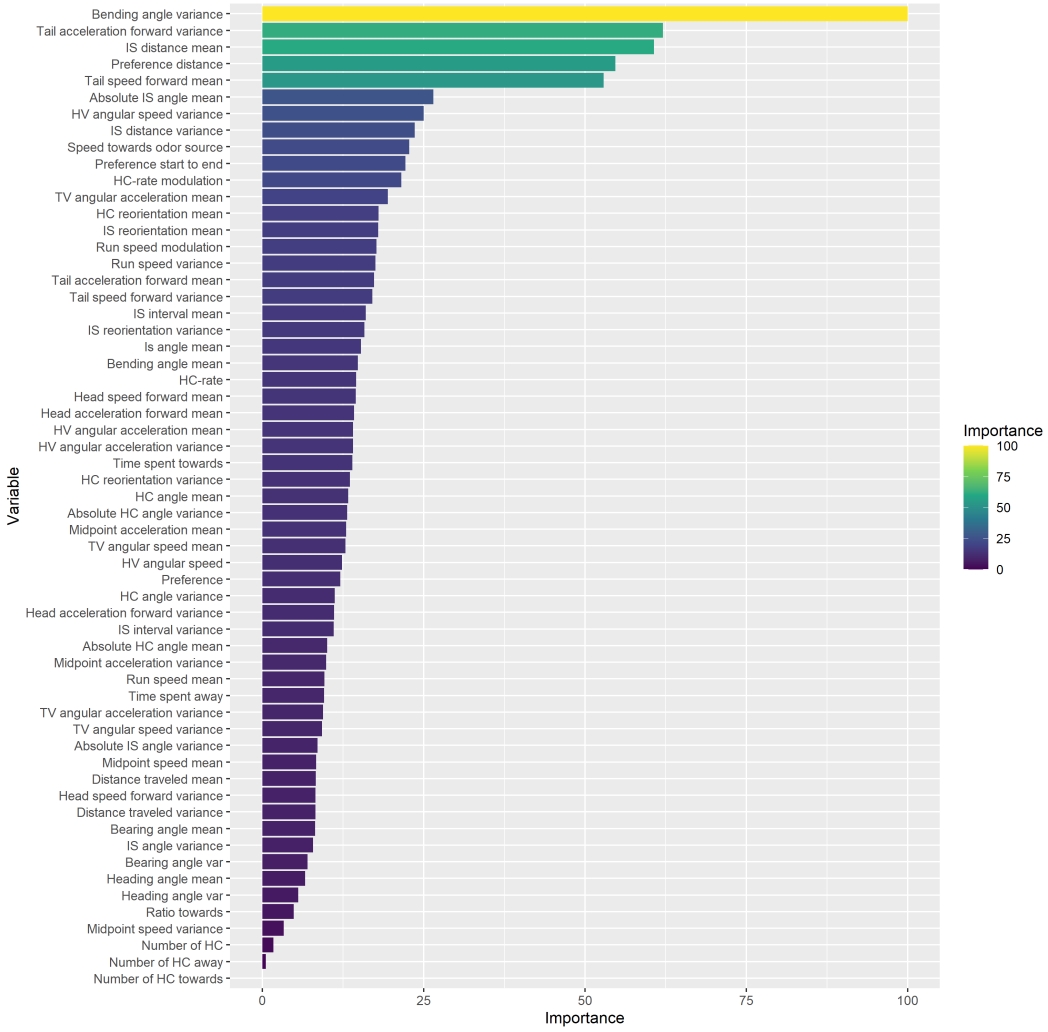


Figure C.8: Variable Importance - Per Individual

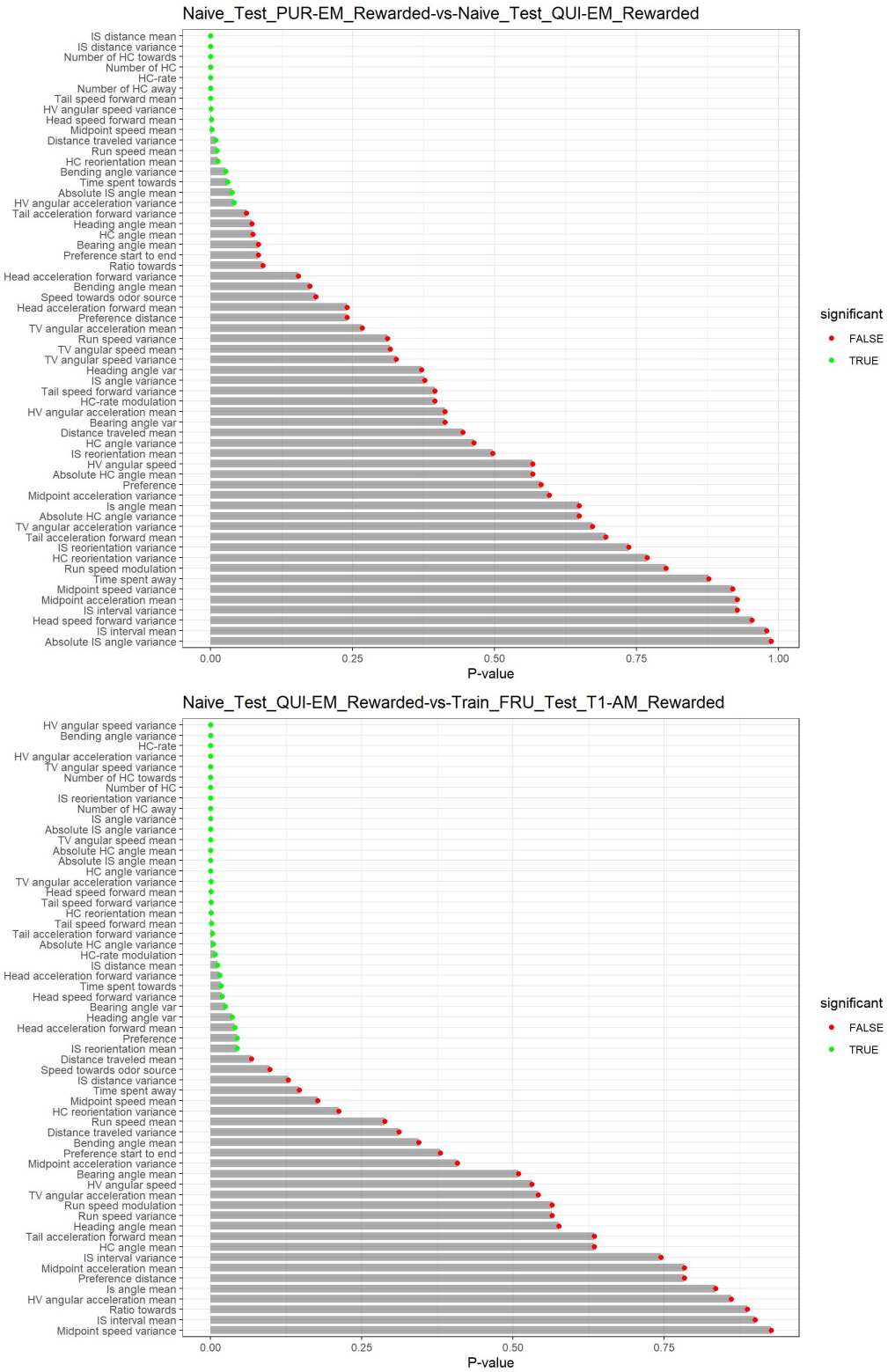


Figure C.9: U-tests per Petri dish I.

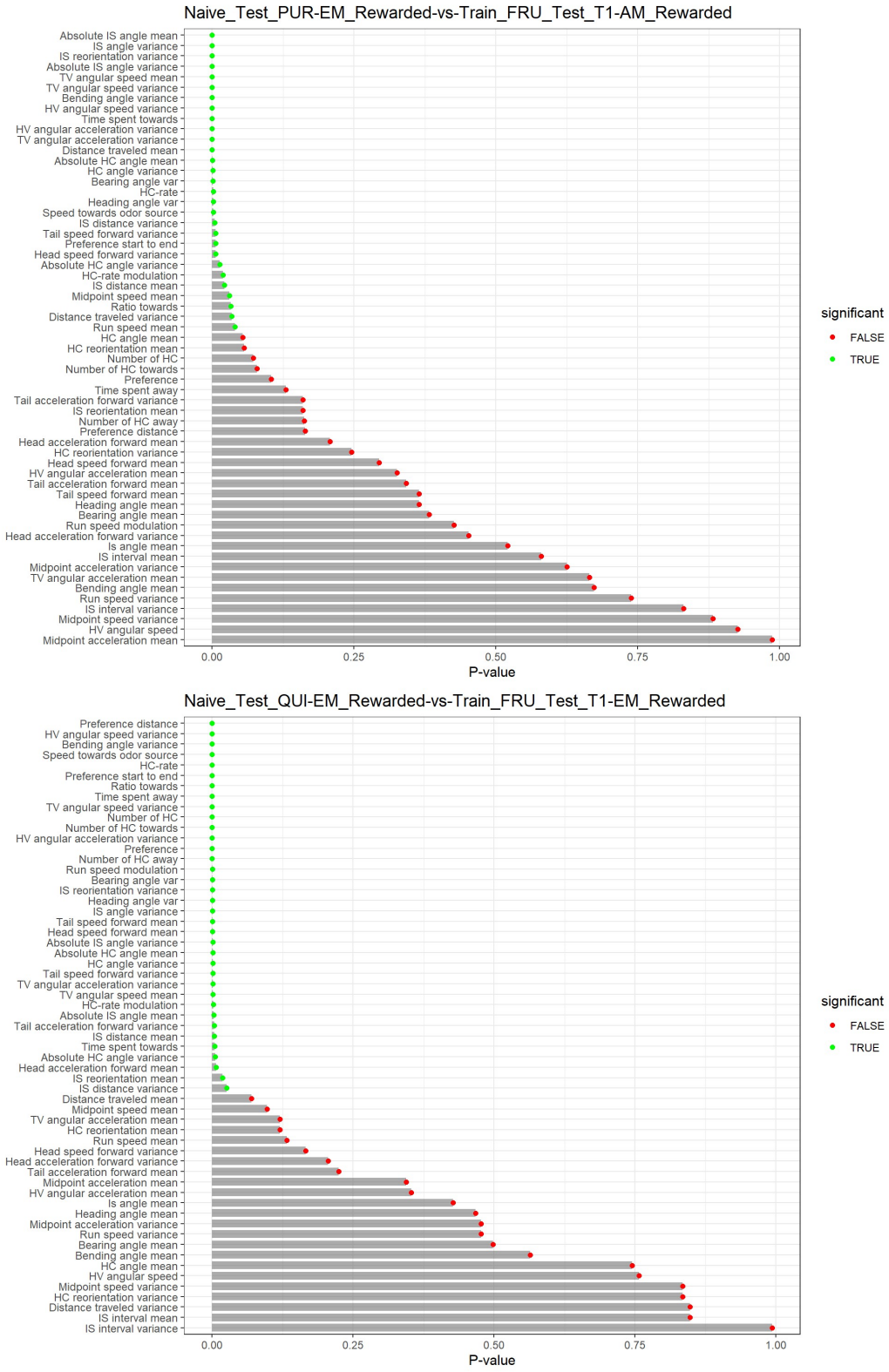


Figure C.10: U-tests per Petri dish II.

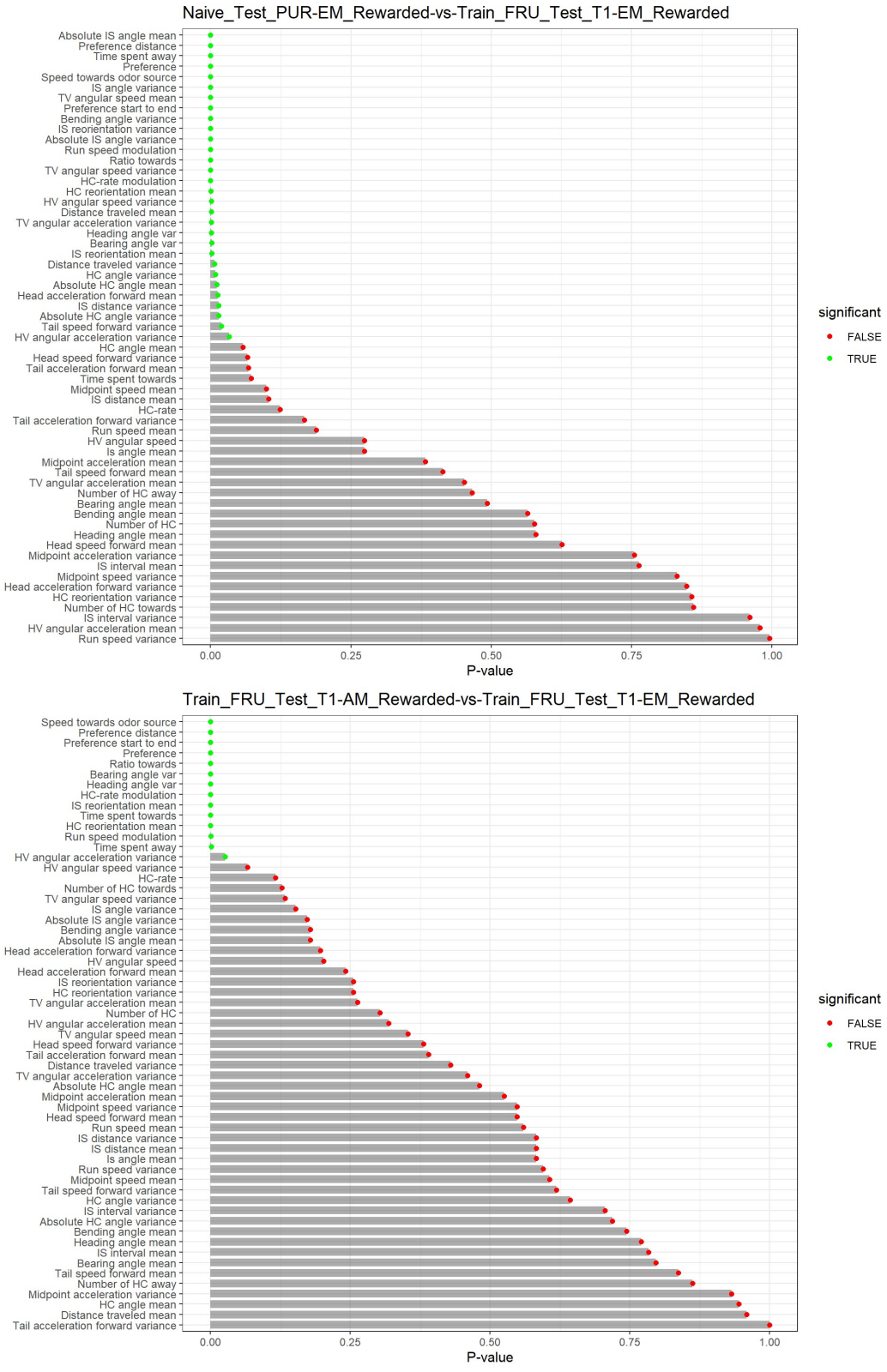


Figure C.11: U-tests per petri dish III.

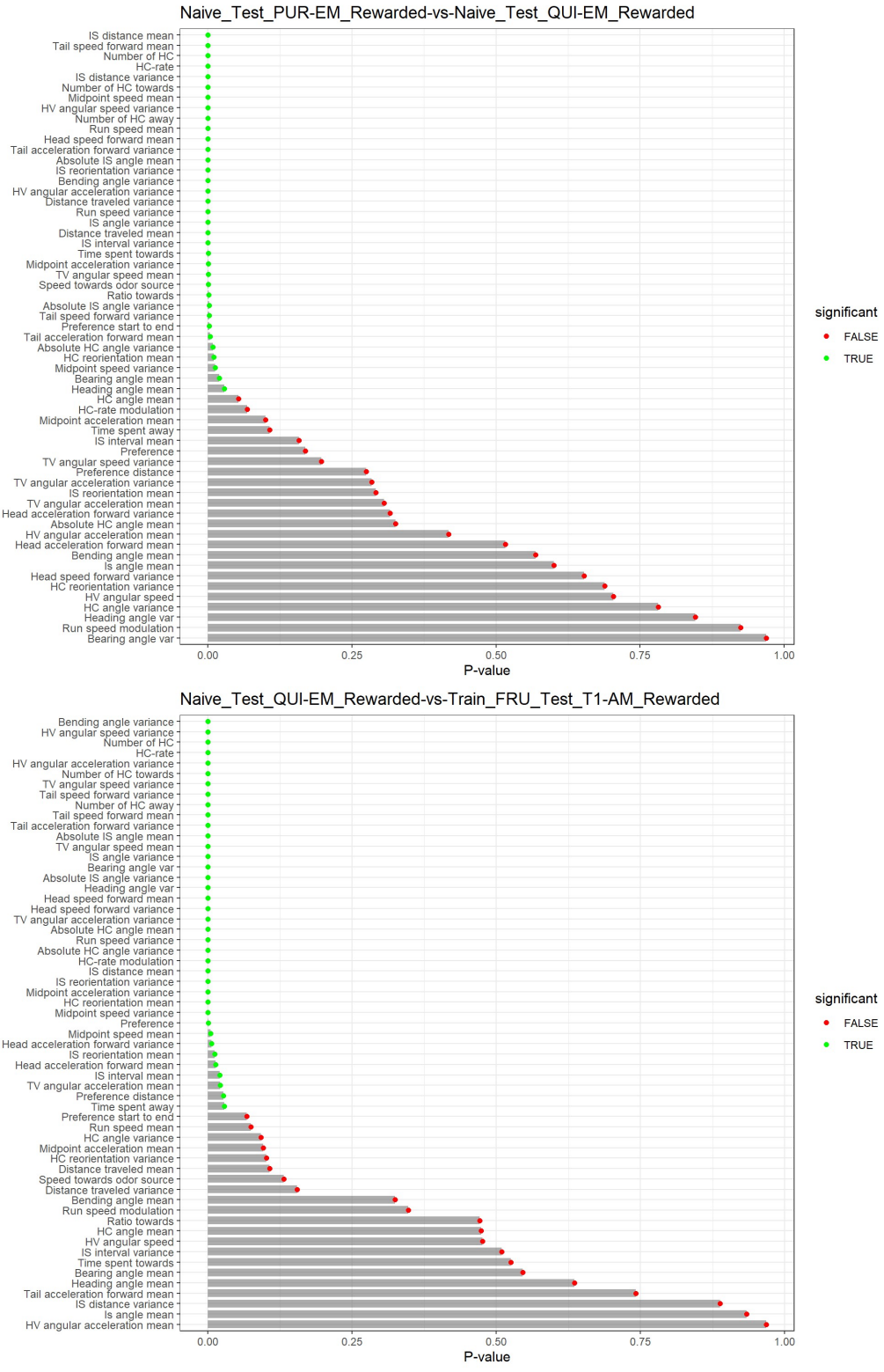


Figure C.12: U-tests per individual larvae I.

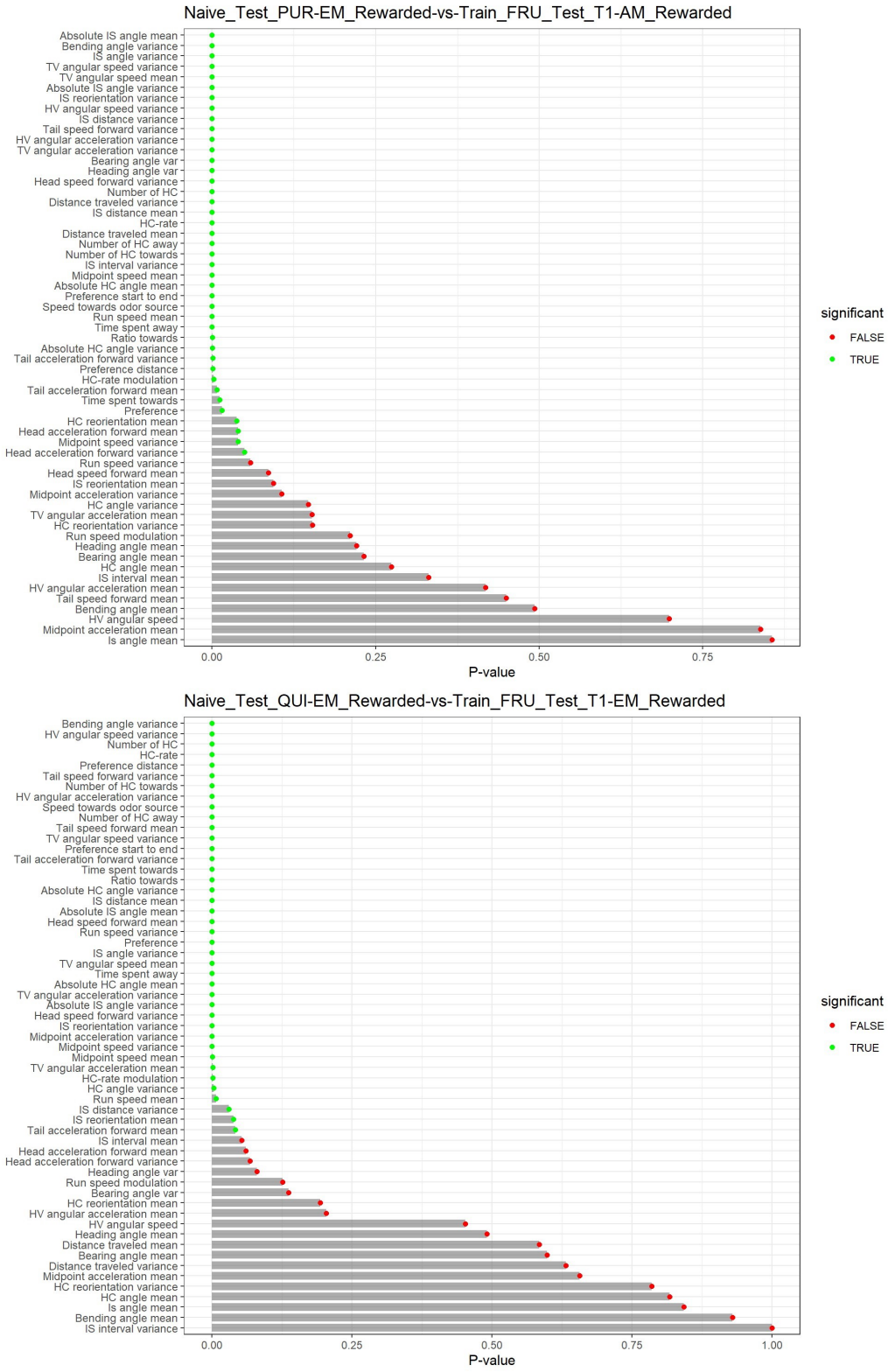


Figure C.13: U-tests per individual larvae II.

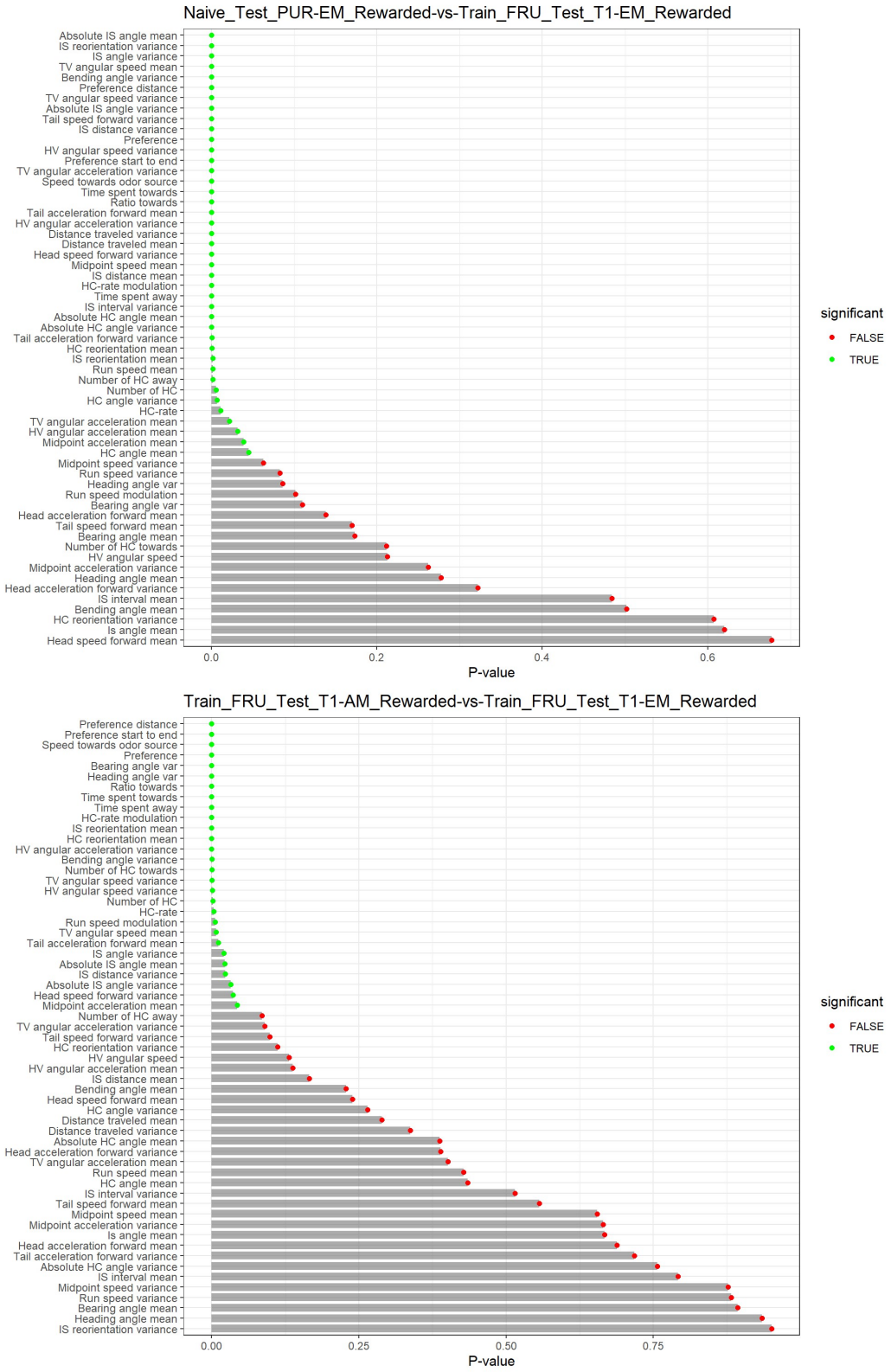


Figure C.14: U-tests per individual larvae III.