PDManager USER MANUAL

Last updated: 2015-02-26

If this manual is outdated an updated version of the manual can be found in the git repository.

Setup

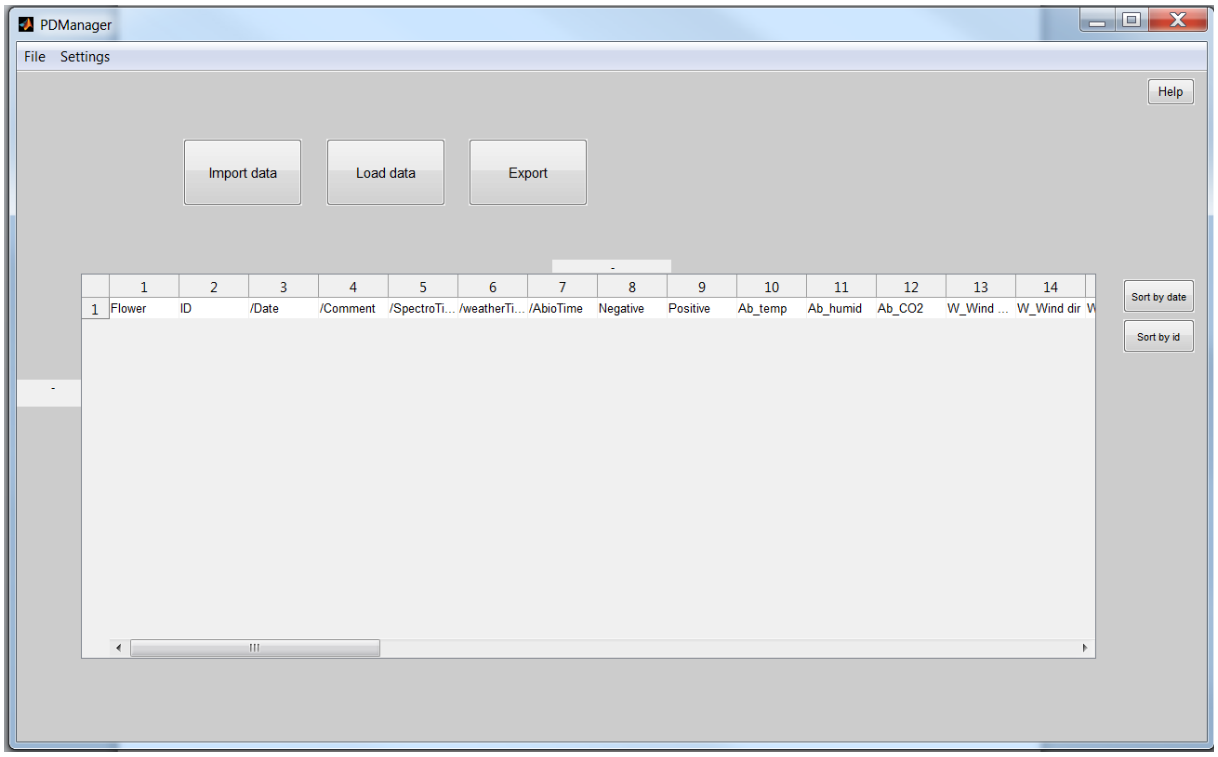
Prerequisites:

- Matlab 12 or later (might work with 2010-11 but tested on 2009 and it does not work)
- Windows
- Microsoft word (only required for the function ”Export metadata”)

1. Download (or clone) the program at https://github.com/jakelamotta/PollinationDataManager
2. Extract the zip file anywhere on the computer.
3. Open matlab
4. Set a path to the location of the program folder
5. Done! Launch the program by typing “main” in the Matlab command window

Usage

In the main window of PDManager (Fig. 1) there are three main choices; *Import data*, which imports and organizes your collected data into a hierarchical folder structure, *Load data,* which loads your data into the main window and creates a dataset with one row for each observation, *Export,* which exports your data set to an Excel file. An observation is defined as a unique flower where studies have been made at a unique time. If there are several flowers studied at the same time, they are all different observations. If one flower is studied on different occasions, every occasion is one observation.

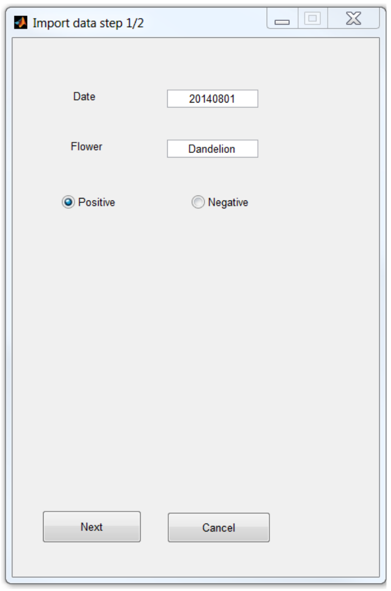


	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1	Flower	ID	/Date	/Comment	/SpectroTi...	/weatherTi...	/AbioTime	Negative	Positive	Ab_temp	Ab_humid	Ab_CO2	W_Wind ...	W_Wind dir

1 **PDManager main window before any data is loaded.**

Importing data

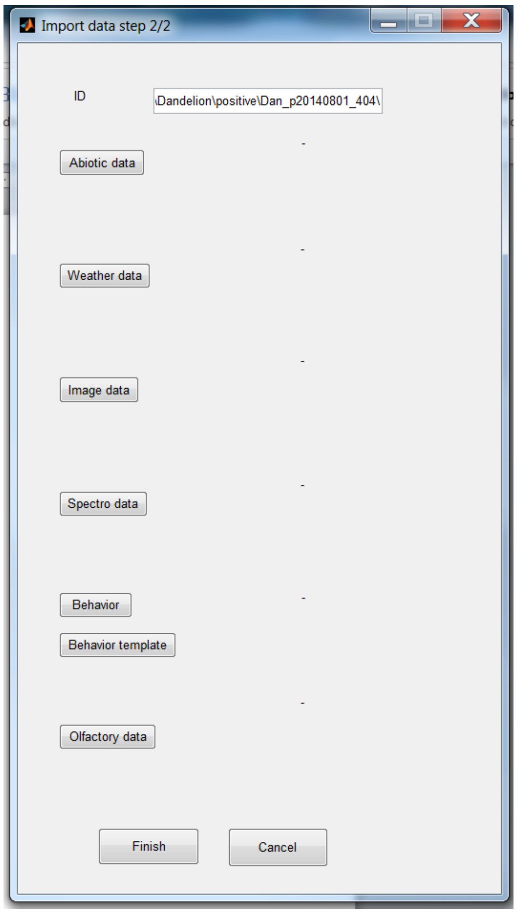
As a new user of ”PDManager” the first thing you need to do is import some data, otherwise there is not going to be anything to export. Importing is done by pressing the “Import Data” button. The following window (Fig. 2) will ask you to enter a date (the date you collected your data), a flower name and choose positive (visited by insect of interest) or negative. Make sure this information is correct as it will be used to identify the observation. An observation is named accordingly: The first three letters of the flower name, followed by p for positive or n for negative, followed by the observation date and a three digit number to make sure each observation gets a unique ID. Example: Dan_p20140801_403 for a Dandelion visited by an insect of interest (positive) on August 1st 2014 with the identifying number 403. The last three digits are generated automatically from the interval 001-999, after which it will continue on 001 again.



Date 20140801
Flower Dandelion
[✓ Positive] Negative

2 PDManager. Import data window.

The next step in the import data process is to import the actual data (Fig. 3). You can e.g. choose abiotic data (.txt files), weather (.dat files), image files (.jpg files), spectral data (.txt files), behavior (only relevant for positive observations) or olfactory data (.csv files). When importing behavior, you can click “Behavior” to import a video file and automatically an excel behavior file (named “template”) is created in the same folder, or you can just click “Behavior template” to import a template excel file without any video.

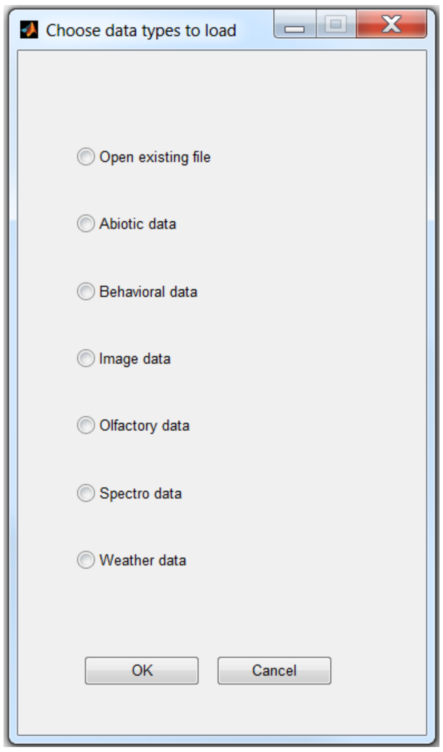


3 Importing data, showing the types of data you can import.

In the "Import data step 2/2" window it is still possible to change the suggested ID by re-naming it at the top of the window. If you want to add data to an existing observation, you fill in the first window (Fig. 2) with the correct information and in the second window you change the ID number to the existing observation and import the desired files as usual. This will load the data into folders structured in a specific hierarchy (abiotic, behavior, image, olfactory, spectra and weather).

Loading data

When data is imported onto the computer the user can now start to load it into "PDManager". The user needs to select what type of data to load and then which folder to search in. Here it is important to note that the user does not need to select exactly the folder where the data is stored as long as it is a parent folder to the data folder. Then "PDManager" will automatically load all data from all subfolders matching the selected type into the program.



4 Loading data, showing the data types that can be loaded.

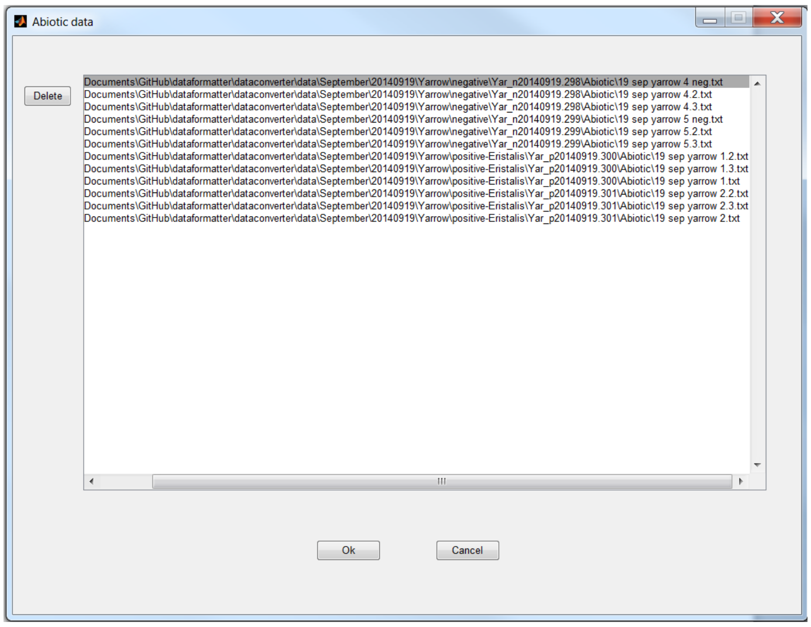
For example, if the user wants all observations from one specific date she can select the folder corresponding to that date and all data from only that date will be loaded.

If you have previously exported part of a data set into excel, this can be loaded back into PDManager by clicking “open existing file”.

You can choose as many data types as you wish.

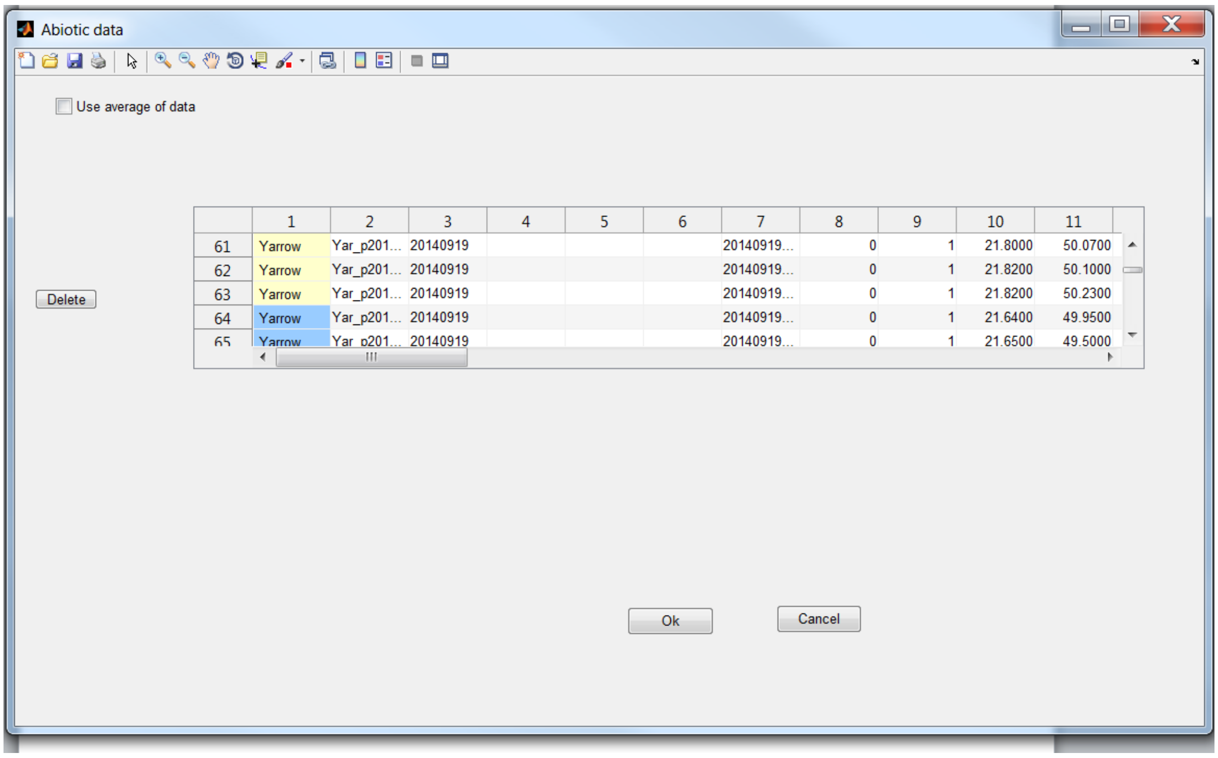
Abiotic data

First you chose the parent folder for the abiotic data you wish to load. If an observation contains multiple files of abiotic data you get a window (Fig. 5) where you can chose to delete certain files. If you do not wish to delete any files, just click ok. A new window appears (Fig. 6) where every line is a measurement and lines from the same observation have the same color on the flower name. Every second observation is yellow and every second is blue. You can use one file by deleting the unwanted ones, or the average of multiple files or measurements for each observation. Check the “Use average of data” to get the average of the rows shown for each observation.



```
Documents\GitHub\dataformatter\dataconverter\data\September\20140919\Yarrow\negative\Yar_n20140919.298\Abiotic\19 sep yarrow 4 neg.txt
Documents\GitHub\dataformatter\dataconverter\data\September\20140919\Yarrow\negative\Yar_n20140919.298\Abiotic\19 sep yarrow 4.2.txt
Documents\GitHub\dataformatter\dataconverter\data\September\20140919\Yarrow\negative\Yar_n20140919.298\Abiotic\19 sep yarrow 4.3.txt
Documents\GitHub\dataformatter\dataconverter\data\September\20140919\Yarrow\negative\Yar_n20140919.299\Abiotic\19 sep yarrow 5 neg.txt
Documents\GitHub\dataformatter\dataconverter\data\September\20140919\Yarrow\negative\Yar_n20140919.299\Abiotic\19 sep yarrow 5.2.txt
Documents\GitHub\dataformatter\dataconverter\data\September\20140919\Yarrow\negative\Yar_n20140919.299\Abiotic\19 sep yarrow 5.3.txt
Documents\GitHub\dataformatter\dataconverter\data\September\20140919\Yarrow\positive-Eristalis\Yar_p20140919.300\Abiotic\19 sep yarrow 1.2.txt
Documents\GitHub\dataformatter\dataconverter\data\September\20140919\Yarrow\positive-Eristalis\Yar_p20140919.300\Abiotic\19 sep yarrow 1.3.txt
Documents\GitHub\dataformatter\dataconverter\data\September\20140919\Yarrow\positive-Eristalis\Yar_p20140919.300\Abiotic\19 sep yarrow 1.txt
Documents\GitHub\dataformatter\dataconverter\data\September\20140919\Yarrow\positive-Eristalis\Yar_p20140919.301\Abiotic\19 sep yarrow 2.2.txt
Documents\GitHub\dataformatter\dataconverter\data\September\20140919\Yarrow\positive-Eristalis\Yar_p20140919.301\Abiotic\19 sep yarrow 2.3.txt
Documents\GitHub\dataformatter\dataconverter\data\September\20140919\Yarrow\positive-Eristalis\Yar_p20140919.301\Abiotic\19 sep yarrow 2.txt
```

5 Loading Abiotic data with multiple files for an observation.



☐ Use average of data

	1	2	3	4	5	6	7	8	9	10	11
61	Yarrow	Yar_p201...	20140919				20140919...	0	1	21.8000	50.0700
62	Yarrow	Yar_p201...	20140919				20140919...	0	1	21.8200	50.1000
63	Yarrow	Yar_p201...	20140919				20140919...	0	1	21.8200	50.2300
64	Yarrow	Yar_p201...	20140919				20140919...	0	1	21.6400	49.9500
65	Yarrow	Yar_p201...	20140919				20140919...	0	1	21.6500	49.5000

6 Loading Abiotic data, showing different colors on different observations.

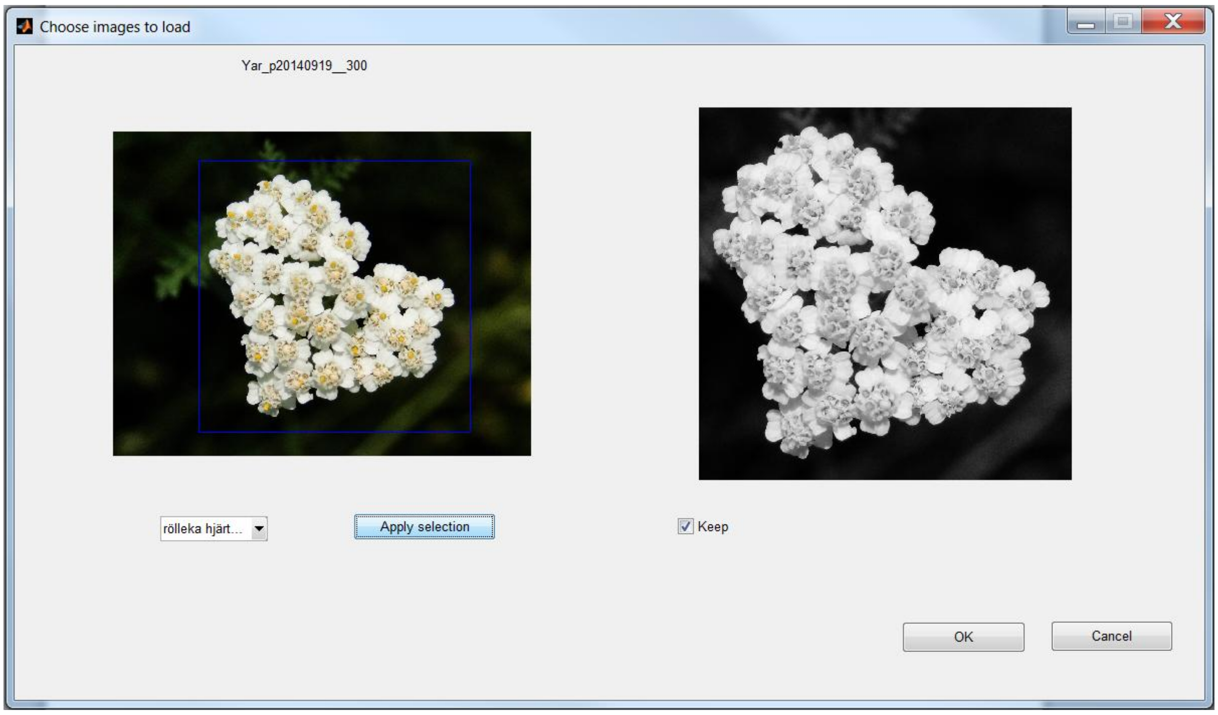
Behavior data

The results of your video file or observation in the field should be put into the file called template.xls. Under the column “Behavior” you fill in the name of the insect that has been pollinating. It is very important that you spell correctly as the name used in the “Behaviour” column must match a name in the text file called “insects” located in the data folder. Under the column “Freq” you fill in the number of visits and under “Dur” you fill in the duration (in seconds) of the visits for the specified insect in “Behavior”. If there are several species visiting a flower you just fill in the information in a second row. Under the column "Time" you fill in the observation time in the format min,sec. For example if you watch for 2 min and 25 seconds you fill in 2,25. PDManager will calculate your observations on a per minute basis after loading the file.

Image data

Once you have chosen the parent folder that includes your image files you will get a window (Fig. 7) with the name of the observation at the top. To the left is oneimage from that observation shown. Below this you have a dropdown menu where you can choose one of your other images for this observation. To crop the image, click and drag on the image. A blue line shows the approximate cropped area. However, because of the upcoming analysis, PDManager recalculates this frame into a square. Click "Apply selection" to get a preview to the right. If you are not satisfied with the cropped image, uncheck "keep" and click and drag for a new selection on the image to the left. If you do not want to crop the image, just click "Apply selection". Always make sure that the "Keep" box is checked for the image you want to use for the analysis. When you are satisfied, click OK and the next observation's images will show up.

If you chose to keep several pictures for an observation, a new window appears (similar to Fig 6). You can use one file by deleting the unwanted ones, or the average of multiple files for each observation. Check the "Use average of data" to get the average of the lines shown for each observation. If you press "cancel" all images you have analyzed so far will be included in PDManager main window.

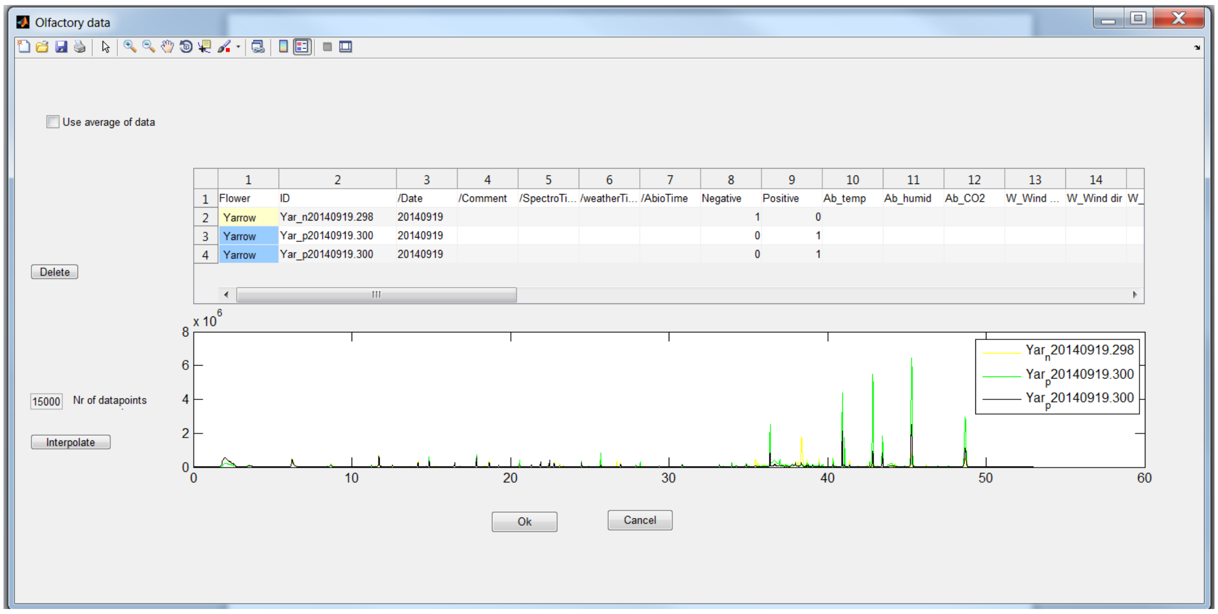


7 Loading image data.

Olfactory data

When you have chosen the parent folder containing the olfactory data you get a window (Fig. 8) where you at the top see your observations. The flower name is colored in alternating yellow or blue. If an observation has multiple files they are shown on top of each other with the same color. You can use one file by deleting the unwanted ones, or the average of multiple files for each observation. Check the “Use average of data” to get the average of the lines shown for each observation.

At the bottom you see a plot of the olfactory data. Use the magnifying glass to zoom in at certain parts to inspect that everything looks all right before deciding to keep or delete data. If you want to use a smaller number of data points, enter the desired number to the left of the diagrams and click “interpolate”. The interpolated lines will be shown in red and you can use the magnifying glass to compare it to the original data. Click OK to add the information to the PDManager main window.



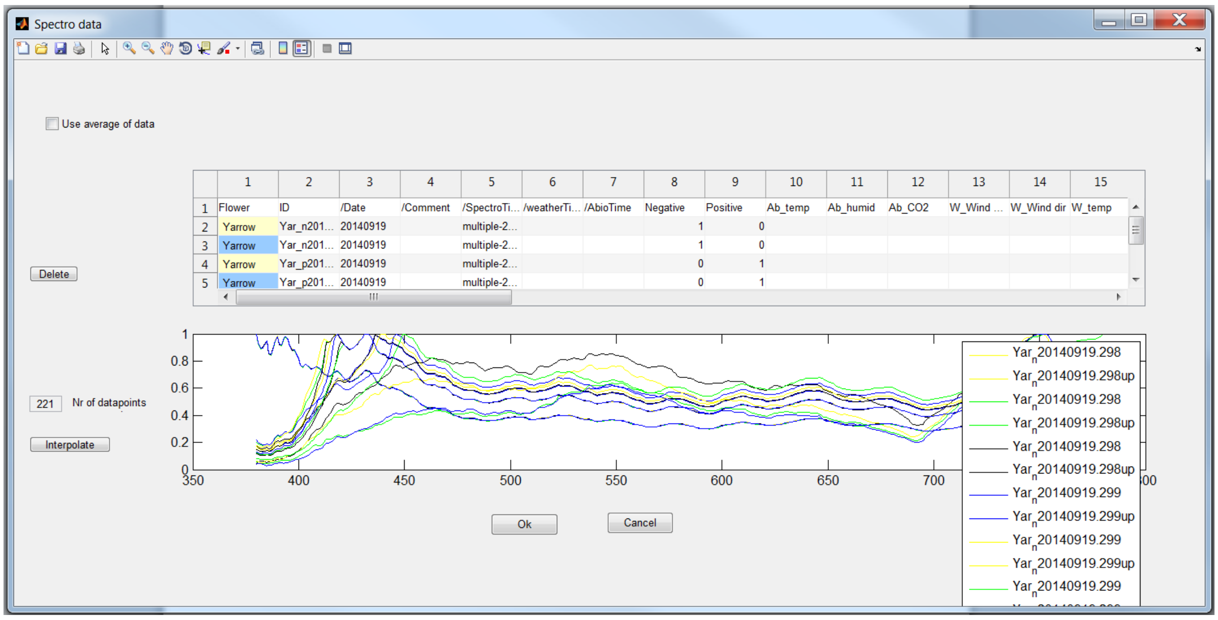
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	
1	Flower	ID	/Date	/Comment	/SpectroTi...	/weatherTi...	/AbioTime	Negative	Positive	Ab_temp	Ab_humid	Ab_CO2	W_Wind ...	W_Wind dir	W_
2	Yarrow	Yar_n20140919.298	20140919					1	0						
3	Yarrow	Yar_p20140919.300	20140919					0	1						
4	Yarrow	Yar_p20140919.300	20140919					0	1						

8 Loading Olfactory data.

Spectral data

Chose the parent folder for the observations you want to analyze. A window (Fig. 9) appears, showing your observations at the top in alternating yellow and blue. If an observation has multiple files they are shown on top of each other with the same color. Below you have a diagram showing a plot of your spectral data. Use the magnifying glass to study it in more detail. You can use one file by deleting the unwanted ones, or the average of multiple files for each observation. Check the “Use average of data” to get the average of the lines shown for each observation.

If you want to use an interpolation of the data you can change the numbers to the left to the desired number of data points for each observation, then click “Interpolate”. Click OK to add it to PDManager main window.



	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	Flower	ID	/Date	/Comment	/SpectroTi...	/weatherTi...	/AbioTime	Negative	Positive	Ab_temp	Ab_humid	Ab_CO2	W_Wind ...	W_Wind dir	W_temp
2	Yarrow	Yar_n201...	20140919		multiple-2...			1	0						
3	Yarrow	Yar_n201...	20140919		multiple-2...			1	0						
4	Yarrow	Yar_p201...	20140919		multiple-2...			0	1						
5	Yarrow	Yar_p201...	20140919		multiple-2...			0	1						

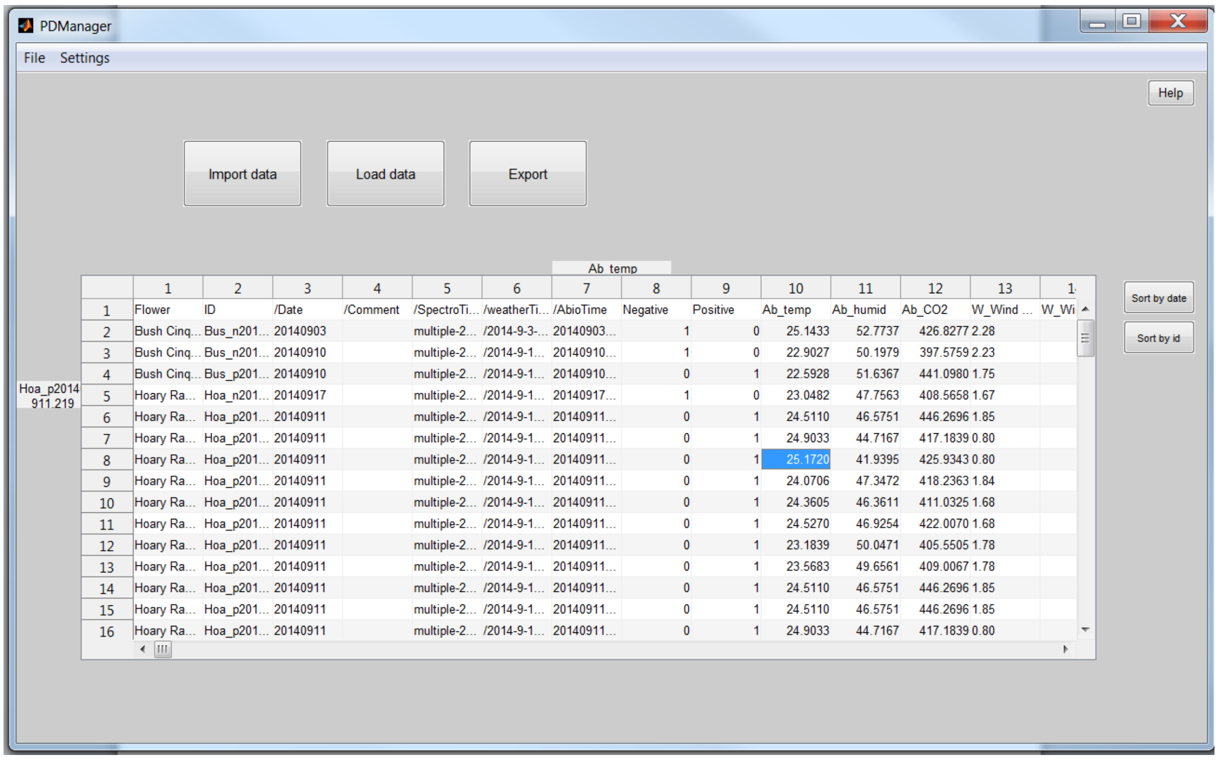
9 Loading Spectral data.

There are two versions of the spectrophotometer data, depending on the type of source file. As two different devices are used to collect data the output is slightly different. One is relative and one is absolute, the former is what in PDManager is known as “Spectro Data” and the latter is called “Spectro Jaz”. To be able to use “Spectro Jaz” the source file needs to be saved as an excel-file (xslx). For example by opening in Excel and then saving.

The PDManager main window

When you have loaded your data you can see the variables and values in the PDManager main window (Fig 10). As the olfactory data is so large, this is excluded from this window by default but the user can choose to include it via the Settings menu. When you scroll down or to the right the observation ID and variable names will be out of sight. However, if you click on a cell, the observation ID will be shown to the left of the loaded data, and the variable name is shown above the loaded data.

You can change the width of a column by clicking on the line at the right hand side of the column number and dragging it to the left or right. You can sort your observations by observation date or ID name by clicking the buttons to the right.



	1	2	3	4	5	6	7	8	9	10	11	12	13	1
1	Flower	ID	/Date	/Comment	/SpectroTi...	/weatherTi...	/AbioTime	Negative	Positive	Ab_temp	Ab_humid	Ab_CO2	W_Wind ...	W_Wi
2	Bush Cinq...	Bus_n201...	20140903		multiple-2...	/2014-9-3-...	20140903...	1	0	25.1433	52.7737	426.8277	2.28	
3	Bush Cinq...	Bus_n201...	20140910		multiple-2...	/2014-9-1...	20140910...	1	0	22.9027	50.1979	397.5759	2.23	
4	Bush Cinq...	Bus_p201...	20140910		multiple-2...	/2014-9-1...	20140910...	0	1	22.5928	51.6367	441.0980	1.75	
5	Hoary Ra...	Hoa_n201...	20140917		multiple-2...	/2014-9-1...	20140917...	1	0	23.0482	47.7563	408.5658	1.67	
6	Hoary Ra...	Hoa_p201...	20140911		multiple-2...	/2014-9-1...	20140911...	0	1	24.5110	46.5751	446.2696	1.85	
7	Hoary Ra...	Hoa_p201...	20140911		multiple-2...	/2014-9-1...	20140911...	0	1	24.9033	44.7167	417.1839	0.80	
8	Hoary Ra...	Hoa_p201...	20140911		multiple-2...	/2014-9-1...	20140911...	0	1	25.1720	41.9395	425.9343	0.80	
9	Hoary Ra...	Hoa_p201...	20140911		multiple-2...	/2014-9-1...	20140911...	0	1	24.0706	47.3472	418.2363	1.84	
10	Hoary Ra...	Hoa_p201...	20140911		multiple-2...	/2014-9-1...	20140911...	0	1	24.3605	46.3611	411.0325	1.68	
11	Hoary Ra...	Hoa_p201...	20140911		multiple-2...	/2014-9-1...	20140911...	0	1	24.5270	46.9254	422.0070	1.68	
12	Hoary Ra...	Hoa_p201...	20140911		multiple-2...	/2014-9-1...	20140911...	0	1	23.1839	50.0471	405.5505	1.78	
13	Hoary Ra...	Hoa_p201...	20140911		multiple-2...	/2014-9-1...	20140911...	0	1	23.5683	49.6561	409.0067	1.78	
14	Hoary Ra...	Hoa_p201...	20140911		multiple-2...	/2014-9-1...	20140911...	0	1	24.5110	46.5751	446.2696	1.85	
15	Hoary Ra...	Hoa_p201...	20140911		multiple-2...	/2014-9-1...	20140911...	0	1	24.5110	46.5751	446.2696	1.85	
16	Hoary Ra...	Hoa_p201...	20140911		multiple-2...	/2014-9-1...	20140911...	0	1	24.9033	44.7167	417.1839	0.80	

10 PDManager main window with loaded data.

Exporting

Exporting the data to an excel file is a trivial step, as the data is loaded into the system. When the data is shown in the main window of PDManager the user can simply press “Export” and then select an xls-file to export to. This file can either be an existing one or a new file. If the file already exists the old data will not be overwritten but rather the new data will be added to the file at the bottom.

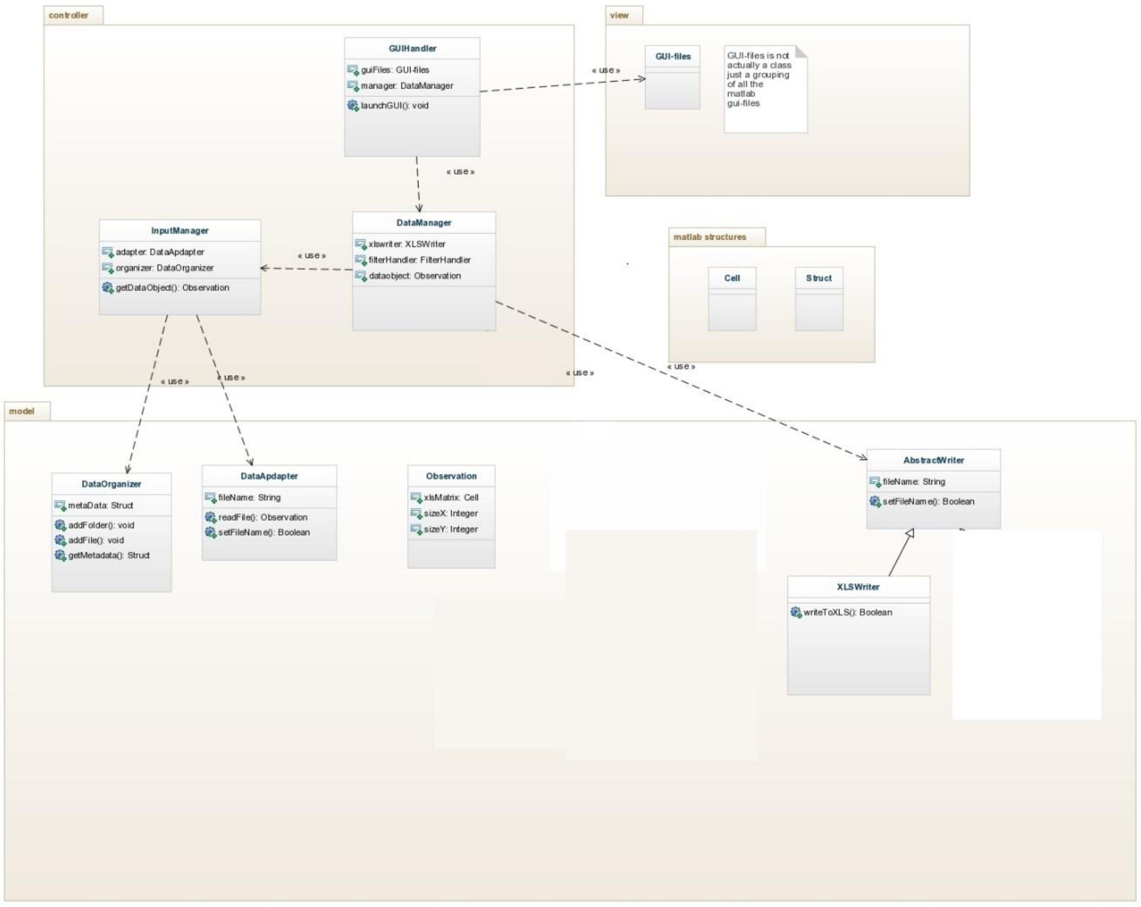
As well as exporting to excel a mat-file containing the Observation object will be saved every time. The file is named with the date and time when the file is saved.

Export metadata

There is a function for exporting a list of the contents of the data folder to a word document, effectively creating a more transparent view over the data. To do this, select “File”→”Export metadata” and wait for the word document to be written and closed. This metadata-file will include all files and folders that have been imported to the “data” folder.

For advanced users

Class tree



The class diagram is from the early design of the system so it is not completely accurate regarding methods and naming of fields and methods, but the classes and interaction between classes are correct. This should provide the advanced user with a clearer picture of the structure of the program. The idea of the design is to decrease coupling between modules and classes. No data processing or similar should be done in the GUI files but rather they should just get data from the underlying data layer and present it to the user at the same time as it passes user input to the data layer.

Observation

The Observation object is probably the most important class of the program as it is the internal representation of what an observation is. It uses a cell array to store all data. Using a matrix was considered as it would have made operations on the data simpler. But since matrices only stores integers using a cell was the only way to keep all the information in one structure.

DataAdapters

The different data adapters contain the code for reading the raw data from source-files and parsing them accordingly. The code for the more complex ones can be quite messy but this is where to look if the format of the input changes.

It is important that the output of the getDataObject that is a function in all data adapters is a

Observation object. This way the code can be changed within the adapter as long as it follows that simple requirement.

DataManager

Important class that act as a mediator between most of the other classes and functions in the project. It also holds the Observation object that is displayed in the main window.

InputManager

Also an important class but not as central. Takes care of anything that relates to the raw data and the input. All the backend of "Import data" is located here, as well as exporting of metadata.

GUIHandler

Basically defines the main GUI and its callbacks. All calls to figure files are made from here. Passes data between the DataManager and the figure files.

This is also the starting point of the application.

Figures

There are many different figures, sometimes not perfectly named, but apart from the main window, if you want to do changes to the interface among these is the place to look.

Adding more weather data variables

Find the WeatherDataAdapter. This is the only file changes needs to be made in.

First find the constructor (function called WeatherDataAdapter) at line ~15. First uncomment line 1 and then comment or remove line 2:

1. `%this.cell_ = {'/weatherTime','W_temp','W_humid','W_Wind speed','W_Wind dir','W_Pressure','W_Radiation'};`
2. `this.cell_ = {'/weatherTime','W_temp','W_humid','W_Wind speed'};`

Then change *`this.nrOfNewVariables`* on row 21 so that it equals 3. It should now work. If not it is in this file you should look to solve potential errors.

Changing the excel template in general

There exists an empty template that variable names (column names) are retrieved from.

There are no problems introducing new variables or removing existing ones but it is very important that Constants file is update accordingly. In Constants.m the position of the Olfactory and Spectrophotometer data arrays are defined and these numbers need to be set to where they actually are located in the Observation cell. The program will not work at all if these do not match exactly. To check where they are located, open PDManager and do not load any data, they will appear in the last columns.

Adding new variables

1. Add the new variable in variables.xls. It can be placed in anywhere after the 9 first columns.
2. Find the DataAdapter corresponding to the data type where the variable belongs. Update the constructor so that the new variable is used (the constructor is the function with the same name as the file, normally located in the top). Note that the variables don't need to be in a specific order in the adapter file, only have exactly the same name as they will be mapped to the correct place automatically.
3. Update the getObservation function in the dataadapter file so that the new variable gets a correct value.
4. Change Constants so that all constants point to the correct column.

Adding behavior variables is a special case as you only need to add the new insect type to the insects.txt document (you still need to update Constants though). If completely new variables are added that are not set in the same way, this is done according to the steps above.

Adding new data types

1. Create a new DataAdapter for the given data.
   - It does not need to have a specific name but following the convention will probably make it easier for future changes.
   - It does need to have certain code parts
     - Extend DataAdapter class (See row 1 in any of the other adapters).
     - Have a method called getObservation(this,paths) that returns a Observation object (again see how the other classes does it)

2. Update the GUI to provide user input to use the new data type.
   - Add case statement in loaddatastep2-->updateSource function.
   - Add new button and text in loaddatastep2 for new data, also add a callback for the button.
   - Add a new radiobutton and corresponding callback function in ImportWindow.m.
   - Add an if-statement for the new type in ímportWindow --> okBtn_callback and update length of types in the same function.

3. Register the new DataAdapter in AdapterFactory by following the pattern for the other data types.
4. Add all the new variables to variables.xls.
5. Change Constants accordingly.