TPM Data Analysis

Samuel Schwab

July 10, 2021

Description

The TPM Analysis tool was made for the Macromolecular Biochemistry research group at Leiden University to analyse tethered particle motion data. The programme loads pre-processed TPM data from data_good.txt files. These files contain the root mean square values of beads recorded during the TPM experiment. Each data_good.txt represents a single TPM experiment done at a certain protein concentration. The data from one data_good.txt file is referred to as a dataset. From these datasets, the programme produces five types of figures: histograms, empirical distribution function plots, 2-dimensional histograms, violin plots, and simple scatter plots. The histograms are used to fit a gaussian distribution. The mean parameter(s) from the fit, also known as the max RMS value, is important for us as it estimates the RMS value with the highest likelihood. Population distribution of all datasets are visualised in a 2-dimensional histogram and a violin plot. The simple scatter plot shows the averaged max RMS values. The empirical distribution function plots visualise the single datapoints of each dataset together with the fit function.

Installation

Build

The latest build files are available at: `https://github.com/SamuelSchwab`. Once downloaded, unpack the zip file and run the TPM.exe.

Source

The latest source files are available at: `https://github.com/SamuelSchwab`. The programme is written in Python 3.9.5, although newer also work. Create a virtual environment using the command prompt. I use virtualenv: `https://pypi.org/`

`project/virtualenv/`. An example to create and run a virtual environment in C:/Python/TPM Data Analysis:

```
cd /
cd Python
cd "TPM Data Analysis"
virtualenv venv
call venv/scripts/activate.bat
```

Next place the source files in this folder and install all required modules:

```
pip install -r requirements.txt
```

Finally, to run the program:

```
python TPM.py
```

How to use

Some example data, called Example Data, can be found in the data folder. To run analysis on this, start the program and in the Data Analysis tab select the data folder in the "Data Folder" option. Enter a minimum RMS value (50 should be good) and a maximum RMS value (170 should be good). Click the Start button in the bottom right. The program now analyses the data. During analyses, it takes the data_good.txt files, creates Kernel Density Estimators, bins the data, and fits the binned data. When it is done it will save a file containing the analysed data in the folder output/Example Data. This file is called configExport.yaml. To generate plots from this analysed data, go to the Plot tab in the programme and select the configExport.yaml file we just generated in the "Load export config" option. Click on one of the buttons at the top to get a preview of a plot. For the histogram previews the programme will show a histogram from a random dataset. The options for the plots can be edited on the right hand side. Plots can be saved by checking the Saved option for the plots that you want to save. Then, click the bottom Save right button. Plots are saved at the location of the the configExport.yaml. Note: the checkboxes next to Histogram, 2D Histogram, Violinplot, Simple plot, and the ECDF only collapse or expand the options for these plots. These checkboxes serve no other purpose.

Options

Data Analysis

Data Folder: Select a folder. The programme will search through the directory tree to find all data_good.txt files. Then, it will parse with regular expression (regex)

the name of each folder containing a data_good.txt file to assign a concentration.

Blacklist: Select a folder to add it to the list. Multiple folders can be added to the list. This folder and all other folders within it will be blacklisted from the data_good.txt file search.

Blacklist Concentrations: Input one numer to add it to the list. Multiple numbers can be added to the list. Data_good.txt files with these concentration values will be blacklisted from the data_good.txt file search.

Minimum RMS: Input a number. All RMS values below this lower threshold value will be filtered out.

Maximum RMS: Input a number. All RMS values above this upper threshold value will be filtered out.

Bin Method: Select one of the options. Bin edges are automatically determined by the selected method. See the abbreviations section.

KDE Method: Select one of the options. The bandwidth of the kernels are automatically determined by the selected method. See the abbreviations section.

p0 a: Input a number. The initial guess of the a parameter(s) of equations 1 and 2.

p0 b

p0 c: Input a number. The initial guess of the c parameter(s) of equations 1 and 2.

Mode Detection: Enable or disable. If enabled, the KDE of a dataset is used to find peaks. If 1 peak is found, the dataset is flagged as unimodal. If 2 peaks are found, the dataset is flagged as bimodal. Higher modes are generally not observed in TPM data. When 3 or more peaks are found, the dataset is flagged as bimodal and the two highest peaks are fitted. If disabled, all datasets are flagged as unimodal.

Minimal Height: Input a number. The required height of a peak to be detected.

Minimal Prominence: Input a number. The required prominence of a peak to be detected.

Minimal Distance: Input a number. The required horizontal distance between neighbouring peaks.

Plot

Load export config: Select a folder. The data in the export config will be used to generate the plots.

Use LaTeXto generate text: Enable or disable. If enabled, all text in the plots are generated with LaTeXand with the Computer Modern font. Requires a working

LaTeXinstallation. If disabled, all text in the plots are generated with matplotlib's mathtext and with the DejaVu Sans font. Note: rendering text with LaTeXcan be very slow.

Show error bars: Enable or disable. Enables or disables the error bars in the plot.

Save: Enable or disable. If enabled, this plot will be saved along with other plot types that have the save option enabled. To save plots press the Save button at the bottom right.

Bin Method: Select one of the options. Bin edges in the y dimension of the 2-dimensional histogram are automatically determined by the selected method. See the abbreviations section.

Normalisation Method: Select one of the options. Each column in the 2-dimensional histogram is normalised by the selected method. By area: the sum of each column is normalised to 1. By amplitude: the values in each column are scaled so that the highest value equals 1.

Normalise by area: Enable or disable. If enabled, the violins are normalised by area.

Width: Input a number. Width of the plot in inches.

Height: Input a number. Height of the plot in inches.

Color Histogram: Input a hex code (with the #) or a color name. Color of the histogram.

Color Fit: Input a hex code (with the #) or a color name. Color of the fit line in the histogram.

Color KDE: Input a hex code (with the #) or a color name. Color of the KDE line in the histogram.

Color Unimodal Points: Input a hex code (with the #) or a color name. Color of the max RMS points from unimodal datasets in the 2-dimensional histogram.

Color Bimodal Points: Input a hex code (with the #) or a color name. Color of the max RMS points from bimodal datasets in the 2-dimensional histogram.

Color Points: Input a hex code (with the #) or a color name. Color of the max RMS points in the simple plot.

Color Error: Input a hex code (with the #) or a color name. Color of the error bars.

Color Violin: Input a hex code (with the #) or a color name. Color of the violins.

Point Size: Input a hex code (with the #) or a color name. Size of the max RMS points. In the 2-dimensional histogram, these points are max RMS values of the datasets. In the simple plot, these points are the averaged max RMS values.

Alpha Histogram: Input a number between 0 and 1. Alpha value (=transparancy) of the histogram.

Alpha Fit: Input a number between 0 and 1. Alpha value (=transparancy) of the fit line in the ECDF.

Line Size: Input a number. Linewidth of the KDE and fit lines.

Error Size: Input a number. Linewidth of the error bar lines.

Label Font Size: Input a number. Font size of the axis labels.

Tick Font Size: Input a number. Font size of the axis ticks.

File Type: Select on of the options. Plot will we saved as the selected file type.

DPI: Input a number. Only relevant if the File Type is png or jpeg. The dots per inch (DPI) value determines the amount of pixels of the saved plot.

For all possible color names, see `https://matplotlib.org/stable/gallery/color/named_colors.html`.

$$y = a * \exp(-0.5 * \frac{(x-b)^2}{c^2}) \tag{1}$$

$$y = a_1 * \exp(-0.5 * \frac{(x-b_1)^2}{c_1^2}) + a_2 * \exp(-0.5 * \frac{(x-b_2)^2}{c_2^2}) \tag{2}$$

Abbreviations

ss: Shimazaki and Shinomoto global method.

ssv: Shimazaki and Shinomoto local method.

auto: Maximum of Sturges and Freedman Diaconis Estimators.

fd: Freedman Diaconis Estimator.

sqrt: Square root (of data size) estimator.