1 Description

The aim of this software is to find the levels (open, closed, ...) in a signal originating from a single-channel patch clamp experiment. For this purpose, the histogram of the signal is analyzed. Since a histogram is defined by a discrete number of $M \in \mathbb{N}$ bins or intervals, it can be viewed as a piecewise linear function $h_M : I \to \mathbb{R}$ with nodes defined by the ascending interval centers x_j, $j = 1, \ldots, M$ and function values given by the number of cases in the corresponding bin. The domain of the function h_M is the interval $I = [x_1, x_M] \subset \mathbb{R}$. For the analysis, the histogram function $h_M : I \to \mathbb{R}$ is approximated by a function $f_N : I \to \mathbb{R}$ that can be represented as the sum of an arbitrary but fixed number $N \in \mathbb{N}$ of scaled Gaussian distributions, i.e.

$$f_N(x) := \sum_{i=1}^{N} \frac{\rho_i}{\sqrt{2\,\pi\,\sigma_i^2}} \exp\left[-\frac{(x-\mu_i)^2}{2\,\sigma_i^2}\right] \tag{1}$$

with scaling factors ρ_i, mean values μ_i and standard deviations σ_i, $i = 1, \ldots, N$.

The corresponding optimization problem can then be formalized as follows.

Problem 1 *Let $h_M : I \to \mathbb{R}$ be a given piecewise linear histogram function and for $N \in \mathbb{N}$ let $f_N : I \to \mathbb{R}$ be defined as in Eq. 1. Find parameters ρ_i, μ_i and σ_i, $i = 1, \ldots, N$, such that the norm $\|h_M - f_N\|$ is minimized.*

To simplify the optimization, parameters r_i, m_i, and s_i are introduced as

$$r_i := \frac{\rho_i}{\sqrt{2\,\pi}\,\sigma_i}, \quad m_i := \mu_i, \quad \text{and} \quad s_i := \frac{1}{\sqrt{2}\,\sigma_i} \tag{2}$$

and the function $f_N : I \to \mathbb{R}$ as given in Eq. 1 can be written equivalently as

$$f_N(x) = \sum_{i=1}^{N} \frac{\rho_i}{\sqrt{2\,\pi\,\sigma_i^2}} \exp\left[-\frac{(x-\mu_i)^2}{2\,\sigma_i^2}\right] = \sum_{i=1}^{N} r_i \exp\left[-(x-m_i)^2\, s_i^2\right].$$

Motivated by the above simplification, we define the function $g_N : I \to \mathbb{R}$ as

$$g_N(x) := \sum_{i=1}^{N} r_i \exp\left[-(x-m_i)^2\, s_i^2\right] \tag{3}$$

and Problem 1 can be reformulated as follows.

Problem 2 *Let $h_M : I \to \mathbb{R}$ be a given piecewise linear histogram function and for $N \in \mathbb{N}$ let $g_N : I \to \mathbb{R}$ be defined as in Eq. 3. Find parameters r_i, m_i and s_i, $i = 1, \ldots, N$, such that the norm $\|h_M - g_N\|$ is minimized.*

A Python script to interactively create, plot, fit, and inspect the histogram and it's approximation was implemented. To solve the optimization Problem 2, a curve-fitting approach available in the scientific computing package *scipy* was applied. From (2), the optimal scaling factors ρ_i, mean values μ_i and standard deviations σ_i solving Problem 1 are derived.

2 Pseudocode

Algorithm 1 An algorithm with caption

Input:

$\boldsymbol{trace}$: 2D array holding time and value data.
$\boldsymbol{numlvls}$: number of Gaussian distributions to use.
$\boldsymbol{numbins}$: number of bins of the histogram.

Output:

$\boldsymbol{\varrho}, \boldsymbol{\mu}, \boldsymbol{\sigma}$: parameters of the Gaussian distributions.

\\\\create histogram from input trace
$\boldsymbol{hist} \leftarrow$ histogram($\boldsymbol{trace}, \boldsymbol{numbins}$)
\\\\create histogram function h_M
$h_M \leftarrow$ histogram_function($\boldsymbol{hist}$)
\\\\get initial values for the parameters
\\\\$r_i$, m_i, and s_i as defined in Eq. (2) for $i = 1, \ldots, \boldsymbol{numlvls}$
$\boldsymbol{r}, \boldsymbol{m}, \boldsymbol{s} \leftarrow$ get_initial_parameters($\boldsymbol{hist}$)
\\\\optimization loop
repeat
  \\\\define the function g_N as in Eq. (3)
  $g_N \leftarrow \sum_{i=1}^{\boldsymbol{numlvls}} r_i \exp\left[-(x - m_i)^2\, s_i^2\right]$
  \\\\compute the error
  $\boldsymbol{err} \leftarrow \|h_M - g_N\|$
  \\\\check if the error is greater than the tolerance
  if $\boldsymbol{err} > \delta$ **then**
    \\\\update parameters
    $\boldsymbol{r}, \boldsymbol{m}, \boldsymbol{s} \leftarrow$ update_parameters($\boldsymbol{r}, \boldsymbol{m}, \boldsymbol{s}, \boldsymbol{hist}$)
  end if
until $\boldsymbol{err} < \delta$
\\\\convert according to Eq. (2)
$\boldsymbol{\varrho}, \boldsymbol{\mu}, \boldsymbol{\sigma} \leftarrow$ convert_parameters($\boldsymbol{r}, \boldsymbol{m}, \boldsymbol{s}$)
return $\boldsymbol{\varrho}, \boldsymbol{\mu}, \boldsymbol{\sigma}$