Gauss fitter program and how it works

Tymoteusz Basak

December 15, 2018

Contents

1 General purpose

The goal of the program is to enable automatic fitting of data to a Gauss-like curve, namely a Gauss curve multiplied by a constant. The curve can be described with a formula

$$\frac{k}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

where μ is a median, σ^2 — a variance and k — a coefficient. The script finds optimal values of μ, σ^2 and k, which yield a minimum error, measured according to the least squares method.

2 Input

As an input, the script takes a file with argument-value pairs. Each pair should be separated by the line feed character, and between argument and value there should be a space or a horizontal tab.[1] As for now, the input pairs should be sorted from the lowest argument to the highest one, this issue should be removed in following versions.

Also, all the values should be positive (arguments, of course, can be any real number).

3 Initial guesses

Firstly, program roughly estimates the optimal values of μ, σ^2 and k and ranges, in which the real values should be included.

3.1 Median

Median μ is estimated simply as an argument with the highest value.

3.2 Variance

Variance σ^2 is estimated, using the neighboring argument to the median. If the median is the highest of the lowest argument in the input, it has only one neighbour. If it's in the middle, out of two neighbors this one is used, which is placed further away from the median.

The neighbour's value should be non-zero. If it is not, it is set to be $^1/_{1000}$ of a median's value.

Variance is then computed, based on the function's value for median and neighour:

$$f(\mu) = \frac{k}{\sqrt{2\pi\sigma^2}} e^{-\frac{(\mu-\mu)^2}{2\sigma^2}} = \frac{k}{\sqrt{2\pi\sigma^2}} e^{-\frac{0}{2\sigma^2}} = \frac{k}{\sqrt{2\pi\sigma^2}} e^0 = \frac{k}{\sqrt{2\pi\sigma^2}}$$

[1] Probably some other whitespaces should work too, but usage of these two most common ones is recommended as they were checked thoroughly.

$$f\left(\mu \pm \Delta\mu\right) = \frac{k}{\sqrt{2\pi\sigma^2}}\, e^{-\frac{(\mu-\mu\mp\Delta\mu)^2}{2\sigma^2}} = \frac{k}{\sqrt{2\pi\sigma^2}}\, e^{-\frac{(\Delta\mu)^2}{2\sigma^2}} = f(\mu)\, e^{-\frac{(\Delta\mu)^2}{2\sigma^2}}$$

$$\frac{f\left(\mu \pm \Delta\mu\right)}{f\left(\mu\right)} = e^{-\frac{(\Delta\mu)^2}{2\sigma^2}}$$

$$\ln\left(\frac{f\left(\mu \pm \Delta\mu\right)}{f\left(\mu\right)}\right) = -\frac{(\Delta\mu)^2}{2\sigma^2}$$

$$\ln\left(\frac{f\left(\mu\right)}{f\left(\mu \pm \Delta\mu\right)}\right) = \frac{(\Delta\mu)^2}{2\sigma^2}$$

$$\sigma^2 = \frac{(\Delta\mu)^2}{2\ln\left(\frac{f(\mu\pm\Delta\mu)}{f(\mu)}\right)}$$

3.3 Coefficient

Coefficient k is estimated based on the median value:

$$f\left(\mu\right) = \frac{k}{\sqrt{2\pi\sigma^2}} \Rightarrow k = f\left(\mu\right)\, \sqrt{2\pi\sigma^2}$$

3.4 Ranges

Median range is estimated with a guessed variance: $\Delta\mu = \sigma^2$. Well, variance is all about this. The ranges of variance and coefficient are simply set to $\Delta\sigma^2 = \Delta k = 10$. As they cannot be negative, the range values are expressed as a factor ($\sigma^2_{real} \underset{\sim}{\in} \left\{\frac{\sigma^2}{1+\Delta\sigma^2};\ \sigma^2(1+\Delta\sigma^2)\right\}$, and analogously for k) and they are fitted very easily, so using such a wide range lengthens the algorithm only slightly.

Median range is expressed in units ($\mu_{real} \underset{\sim}{\in} \{\mu - \Delta\mu;\ \mu + \Delta\mu\}$) and converges less easily, therefore it is much more important to guess close enough to the optimal value.

4 Fitting values

At the beginning of each optimisation step, the program computes the error value of function for estimated values of μ, σ^2 and k, and:

- $\mu \pm \Delta\mu$, σ^2 and k,
- μ, $\sigma^2 \times (1+\Delta\sigma^2)^{\pm 1}$ and k,
- μ, σ^2 and $k \times (1+\Delta k)^{\pm 1}$.

Based on these results, two different optimization steps can be performed:

Value change

If any of computed error values for one of neighboring argument proves to be smaller than the value for estimated ones, the estimated value changes, according to minimisation of an error.

Uncertainty tightening

If the estimated values prove to be the best of all examined, the uncertainty region is tightened, namely the values of $\Delta\mu$, $\Delta\sigma^2$ and Δk are reduced by the factor of 1.2 or 1.5 — the one with the biggest error difference is reduced by 1.5, and the others by 1.2. This lets uncertainties to be reduced more uniformly (according to errors, they yield).

5 Terminating the algorithm

The algorithm terminates if any of the following conditions occurs:

- the sum of all error values for neighboring arguments is less than 10^{-30} units bigger than the error value for estimated ones,
- the uncertainty values for all three arguments are lower than 10^{-30},
- the algorithm performed 1000 optimisation steps.

In all these cases, continuation of the optimisation is considered pointless.

6 Potential improvements and development

One of potential improvements was already mentioned earlier — automatic sorting of argument-value pairs in the input. This issue should be possible to be corrected very easily. There should be also possibility to compute the error values using different methods. Testing of the algorithm proved, that the values should be sometimes fitted with more respect to small (especially zero) values from input. Gauss-like curves' values approach to zero quite quickly when increasing the distance from the median, and therefore low input values contribute much less to the total error value, than the high ones. This issue could also be improved by adjusting the function's formula, which leads to the next paragraph.

For the development — it would be great to enable fitting into other functions. Their formulae should be also provided in an input, together with set of values to be fitted, their possible ranges — for $\mathbb{R}$, the ranges should be expressed in units, and for $\mathbb{R}_+$ — as a factor, and formulae for their guessed ranges — analogously to the original `gauss-fitter`.