International Journal of Physics and Mathematics
www.physicsjournal.net
Online ISSN: 2664-8644, Print ISSN: 2664-8636
Received: 27-05-2021; Accepted: 13-06-2021; Published: 28-06-2021
Volume 3, Issue 1, 2021, Page No. 35-43

Multidirectional regression analysis

Koen Van de Moortel
Master's degree in Experimental Physics at the University of Ghent, Belgium

Abstract
The so-called ‘least squares regression’ for mathematical modeling is a widely used technique. It’s so common that one might think nothing could be improved to the algorithm anymore. But it can. By minimizing the squares of the differences between measured and predicted values not just in the vertical but also in the horizontal direction. We can call it ‘multidirectional regression analyses’. This improvement to the ‘least squares regression technique’ is usable for all kinds of invertible model functions: linear, exponential, power, logistic, and many other functions. Especially for power functions, often used in biomedical sciences, the conclusions you make from your data can change dramatically. An important example shown here is the Body Mass Index. We can now explain why scholars used to find a quadratic relationship between the mass of people and their height, against the scaling laws, and we see that the scaling laws are indeed respected if we use the improved fitting method.

Probably the most important advantage of multidirectional regression is that the fitted model is invariable if the dependant and independent variables are switched. This was a serious and neglected problem with the ordinary least squares method.

The examples were calculated with a specially developed software program, called ‘Fitting KV dm, version 1.0’.

Keywords: regression, mathematical modeling, multidirectional regression, algorithm, body mass index

Introduction

In the process of writing a book about measuring methodology and regression analysis, I thought the so called “Body Mass Index” (BMI) might be a good example of quantization, how to put a number on ‘overweight’. As you probably know, it is calculated by taking a person’s mass m (in kg) and divide it by the square of the height h (in meters). Now, this is quite awkward, since the masses of objects with the same shape and similar density distributions are proportional with the *third* power of the height (or any longitudinal dimension).

So I started digging... Why did the inventor, Adolphe Quêtelet, who happens to have lived in the same town as me (Ghent, Belgium), define this index with h^2 in 1832? I wanted to find the original data that he analyzed, the ‘reference people’ to calibrate it. Strangely, there seems to be no trace of them on the internet, and also no other dataset could be found! Thousands of sites offer ‘ideal mass’ tables or calculators, and many use obscure disclosed formulas, clearly not using h^2, some even using a linear relationship!

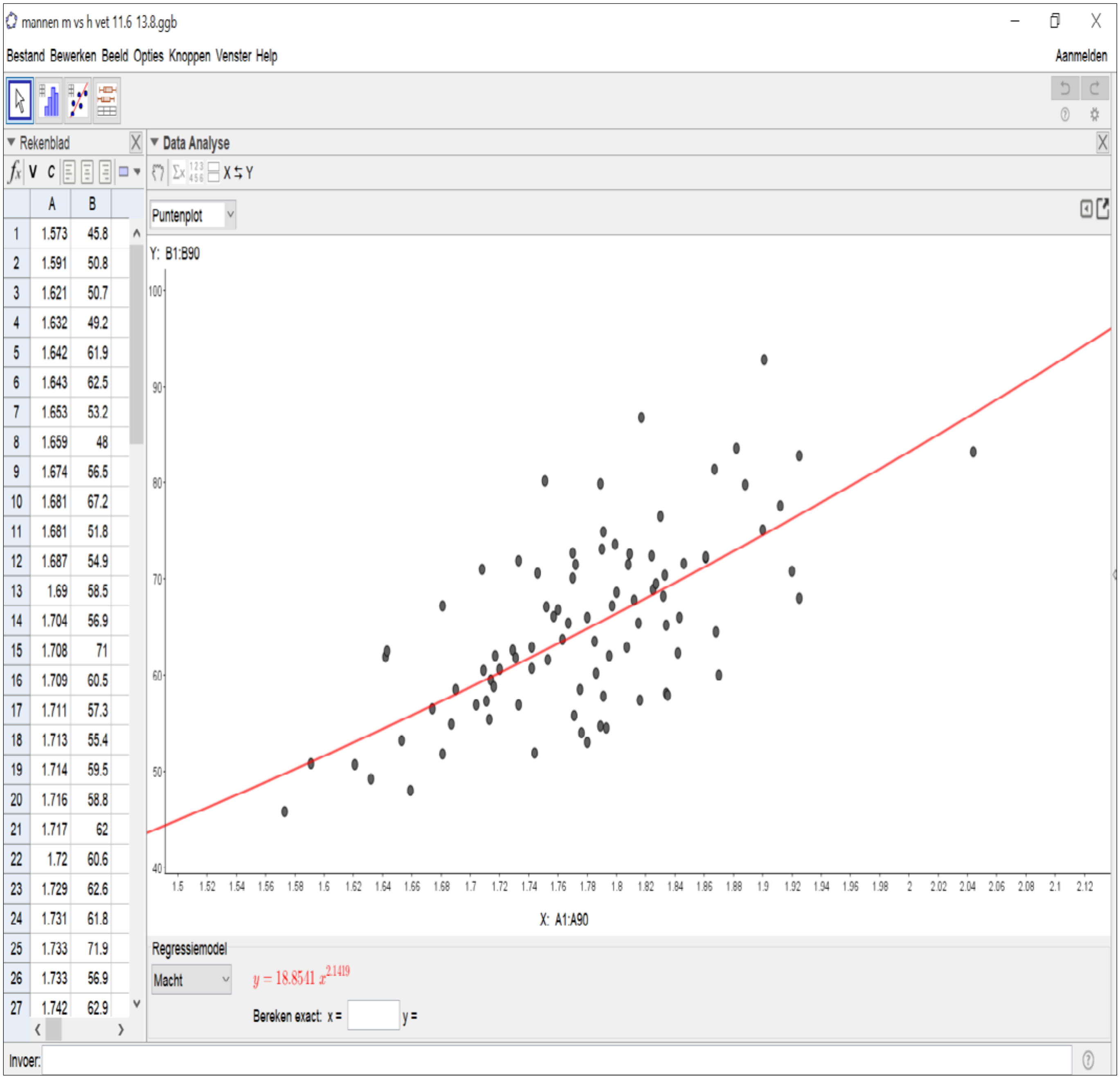
For me as a physicist, it’s hard to believe, but apparently it took almost a century before someone (Fritz Rohrer, CH) came up with the idea to calculate the index with h^3 anyway. This number: m/h^3 is then called ‘Corpulence Index’ (CI) or ‘Ponderal Index’ (PI) [Rohrer 1921]. It took another century until someone (Sultan Babar, SA) found what was to be expected: “It has the advantage that it doesn’t need to be adjusted for age after adolescence.” [Babar 2015] In spite of that, the general public still only knows the BMI.

It took until 2013 before someone like Nick Trefethen (numerical analyst at the University of Oxford, GB) raised his eyebrows and dared to make this remark in The Economist: “The body-mass index that you (and the National Health Service) count on to assess obesity is a bizarre measure. We live in a three-dimensional world, yet the BMI is defined as weight divided by height squared. It was invented in the 1840s, before calculators, when a formula had to be very simple to be usable. As a consequence of this ill-founded definition, millions of short people think they are thinner than they are, and millions of tall people think they are fatter.” And then he said: “You might think that the exponent should simply be 3, but *that doesn’t match the data at all*. It has been known for a long time that people don’t scale in a perfectly linear fashion as they grow. I propose that a better approximation to the actual sizes and shapes of healthy bodies might be given by an exponent of 2.5. So here is the formula I think is worth considering as an alternative to the standard BMI: ‘new BMI’=$1.3m/h^{2.5}$. [Trefethen, 2013 [8], my emphasis]

Now, how could it “not match the data”? I was curious now to inspect some data myself. After a long search, I came in touch with Nir Krakauer (The City College of New York), who was also doing BMI-related modeling, and he was so kind to refer me to his data:rdrr.io/github/dtkaplan/NIMBIOS/man/nhanesOriginal.html

From this large collection, I extracted the masses and heights of 90 adult men who had a more or less ‘ideal’ body fat percentage: between 11.6 and 13.8%. I’m not a medical doctor, but according to different sources these percentages seem to be considered good for young adults. The most important point for this selection was to have a more or less homogeneous group with a range of sizes, but with similar densities. Of course I know other factors like bone density and body type play a role as well, but this is the best I could do.

First, I put the data in the popular math program ‘GeoGebra’ (version 5 Classic):



mannen m vs h vet 11.6 13.8.ggb

Rekenblad

	A	B
1	1.573	45.8
2	1.591	50.8
3	1.621	50.7
4	1.632	49.2
5	1.642	61.9
6	1.643	62.5
7	1.653	53.2
8	1.659	48
9	1.674	56.5
10	1.681	67.2
11	1.681	51.8
12	1.687	54.9
13	1.69	58.5
14	1.704	56.9
15	1.708	71
16	1.709	60.5
17	1.711	57.3
18	1.713	55.4
19	1.714	59.5
20	1.716	58.8
21	1.717	62
22	1.72	60.6
23	1.729	62.6
24	1.731	61.8
25	1.733	71.9
26	1.733	56.9
27	1.742	62.9

Data Analyse

Puntenplot

Y: B1:B90

X: A1:A90

Regressiemodel: Macht

$y = 18.8541\, x^{2.1419}$

Bereken exact: x = y =

Invoer:

Fig 1: Mass vs height of ‘ideal' men, power function model in GeoGebra

This calculated the following relationship as ‘best fitting’: m = $18.8541Ah^{2.1419}$.

Aha, that must have been the reason why Quêtelet decided to round the exponent of h to 2, because the empirical value seems to be 2.1419!

Then I realized that this program takes the logarithms of the variables, in order to reduce the regression problem to a linear one. This causes errors, as I illustrated elsewhere [Van de moortel 2021 [9]].

So I decided to put the data in my own software program, called ‘FittingKVdm’, which uses an iterative algorithm to estimate the parameters. This produced: m = $19.331Ah^{2.1084}$. Now the exponent was even closer to 2! Strange! GraphPad Prism 9.0.2, a program that seems well designed to me, and also uses iteration, produced an identical result. Their writers also condemn the logarithm habit, by the way. Still not being happy, I wanted to see the difference between a fit with a fixed exponent of 2 and one with exponent 3.

The results: m = $20.5918Ah^2$ and m = $11.4640Ah^3$.

The value of 20.5918 is indeed a good BMI, and 11.4640 is close to the ‘good’ value of 12 for the CI according to Sultan Babar.

Now, a picture is worth a thousand words, so I would like you to take a look on the mass versus height graphs of both fitted curves (don’t mind if you can’t read the small letters):

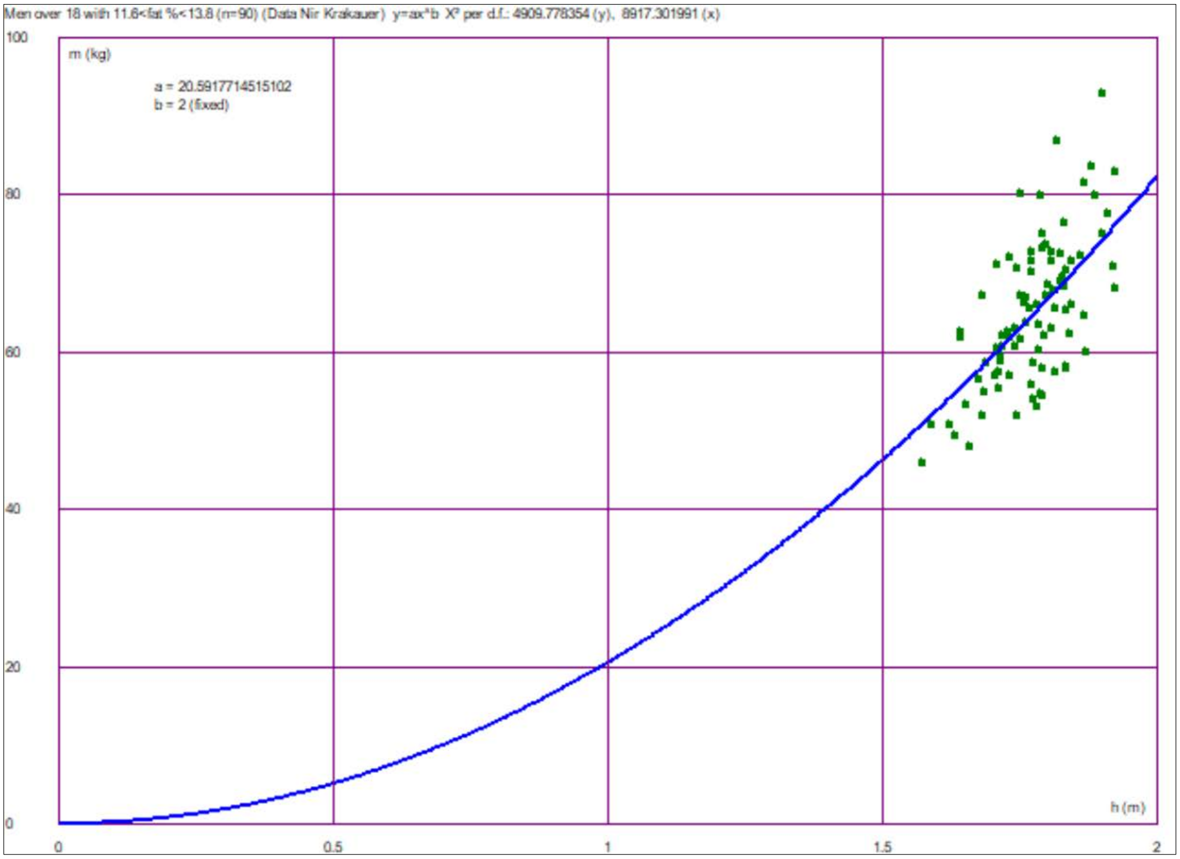


Fig 2: mass vs height, ‘ideal' men, quadratic model

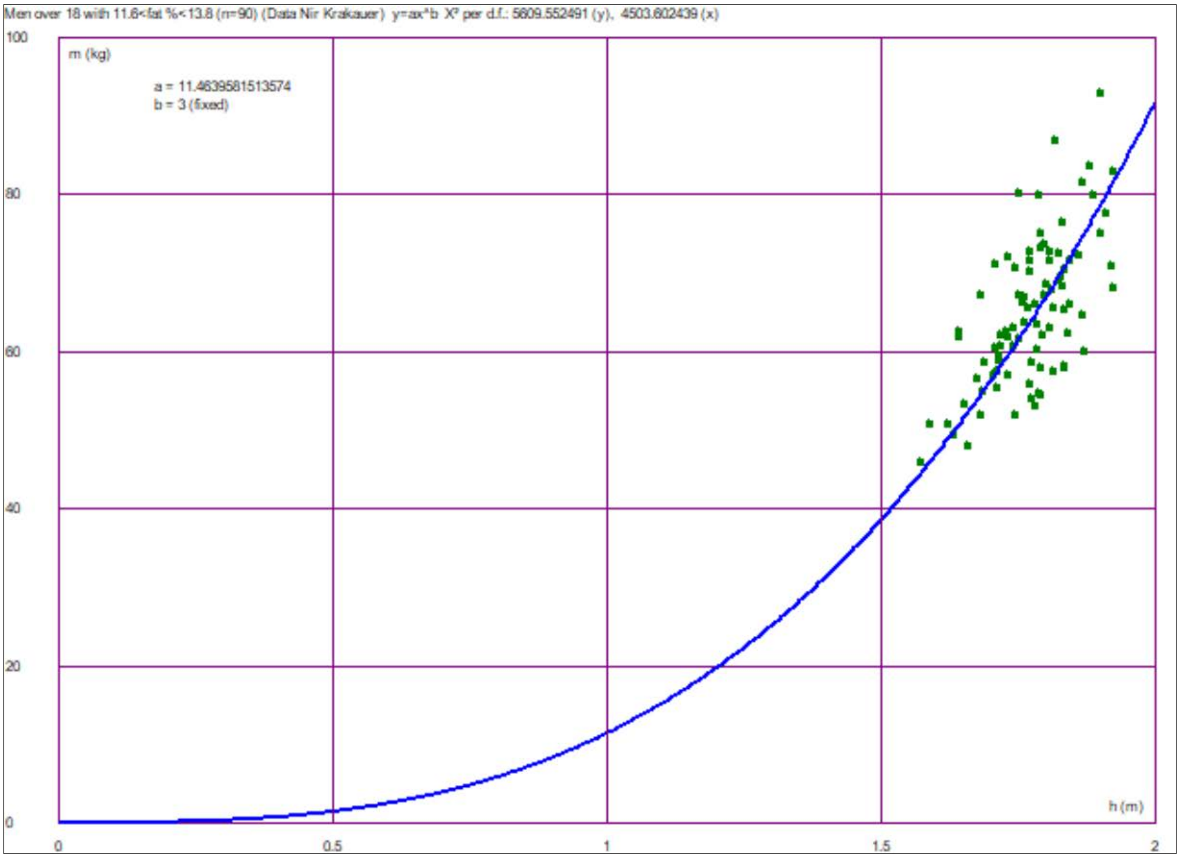


Fig 3: mass vs height, ‘ideal' men, cubic model

Which line visually fits the best through the cloud of points? Everyone I asked this question, answered: the one on the right, obviously!

So now the question came up: is there something wrong with the regression method itself? Well, there is definitely an asymmetry: the classical algorithm that everybody uses, minimizes the sum of the (weighted) *vertical* distances between the measured (y_i) and the predicted y values f (x_i). The weights are inversely proportional to the square of the measuring errors $\sigma_{y,i}$. The parameters in the model function are adjusted in order to minimize this sum:

$$S=\sum_{i=1}^{n}\frac{(y_i-f(x_i))^2}{\sigma_{yi}^2}$$

Would it make any difference if we would use the sum of the *horizontal* distances? Why is it not done? Well, in the case of non-invertible functions, especially periodical functions, there are many such distances for every y value, but for a bijective function like the one above, it's perfectly possible. The simplest way to try it, is by switching the so called 'independent' and the 'dependent' variable.

If the 'best fit' for our data, with free moving exponent, m = 19.331Ah$^{2.1084}$, was indeed the best fit, it shouldn't make any difference if we switched the h and m columns and fitted again, should it? The expected outcome of this procedure would be:

$$\Rightarrow h=\left(\frac{m}{19.331}\right)^{\frac{1}{2.1084}}=0.24534\cdot m^{0.47429}$$

Now, what was the actual outcome? h = 0.72877Am$^{0.21394}$

Or, inverted: m = 4.38814Ah$^{4.67421}$

I double-checked it using GraphPad... same result.

GeoGebra gave almost the same: h = 0.7225Am$^{0.2159}$

This is not just a small difference, like a 'rounding error' or so. This is obviously shocking and dramatic!

Is it possible that nobody ever noticed this? Or that nobody cared? Well, after a long search, I found some people who made the same observation, like Sebastian Kranz, economist at the University of Ulm (D), but he concludes his text with this strange little 'poem': "Don't make you course a mess, but just be sly, and never simultaneously regress, y on x and x on y." [Kranz 2018] [5].

Anyway, not many people seem to care, since in all the textbooks I read, the numerous Youtube lessons about regression and all the software programs I ever tried, I never saw any remark about this phenomenon. Most student textbooks they don't even mention the weights that should be used in the S sum [e.g. Cohen 2011, p. 18, Bijma e.a. 2016] [3, 2] or just briefly [e.g. Dukkipati 2010 chapter 6] [4].

I experimented with other data and other invertible functions. The same happens every time.

Solution: Multidirectional Regression

The classical regression, minimizing the squares of the vertical distances (or residues $r_{y,i}$, see the graph below), seems to pull the line through a cloud of points too much horizontally. Minimizing the horizontal distances $r_{x,i}$ (i.e. by switching x and y) pulls the line too much vertically.

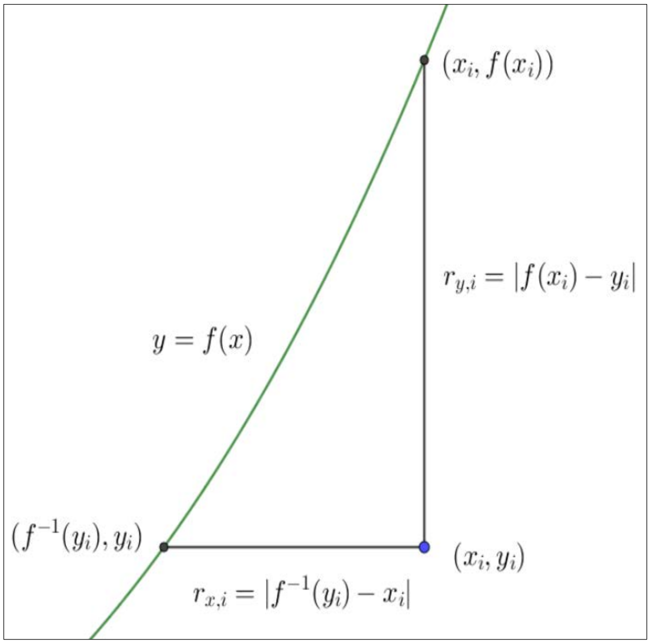


Fig 4: A measured point, the model function, and the horizontal and vertical distances, of which the (weighted) product is to be minimized.

Because of symmetry reasons, there is no reason to favor one of both if f is a bijection. Therefore it seems only logical to give the two 'pulling forces' equal rights, and minimize this sum:

$$S=\sum_{i=1}^{n}\frac{\left(y_i-f(x_i)\right)\cdot\left(x_i-f^{-1}(y_i)\right)^2}{\sigma_{yi}^2\cdot\sigma_{xi}^2}$$

I implemented this in 'FittingKVdm, version 1.0', and I would call it 'multidirectional regression', or shorter: 'xy-fitting'. It can be expanded in multiple directions of course, if there are more variables.

I will not keep you in suspense any longer: fitting the same data now, gave:

m=ah^{b}

a=10.482±0.058

b=3.1581±0.0092

That exponent is a lot closer to 3, as we physicists always expected! And 3.1581 is approximately equal to the geometric mean of 2.1084 and 4.67421, which makes sense.

Now again switching the variables, we would predict:

$$h=\left(\frac{m}{10.482}\right)^{\frac{1}{3.1581}}=0.47520\cdot m^{0.31665}$$

The actual xy-fitting produced, as expected, this result, the same if we neglect the rounding errors:

h=am^{b}

a=0.4752±0.0014*

b=0.31664±0.00069

(*The confidence intervals are estimated by doing 100 fittings with the data+random noise with the same magnitude as the probable error on the measurements, i.e. the x_i values are replaced by $x_i+g(\sigma_{x,i})$ and $y_i+g(\sigma_{y,i})$, g being a Gauss distributed random number function.)

More Examples

Of course, I tried this new method with some other data. I will show you a few examples. On the left graph you see the result of traditional regression, and on the right you see the new method applied. The dotted lines are ‘worst case scenarios’, with parameters at the limits of their confidence intervals. Again, don’t mind the small letters, just look how the line (doesn’t) go through the points, and how the results improve!

Example 1: Average heart beats (B, in pulses/s) of some mammals versus their mass (m, in g). Model function: B = aAm^b.

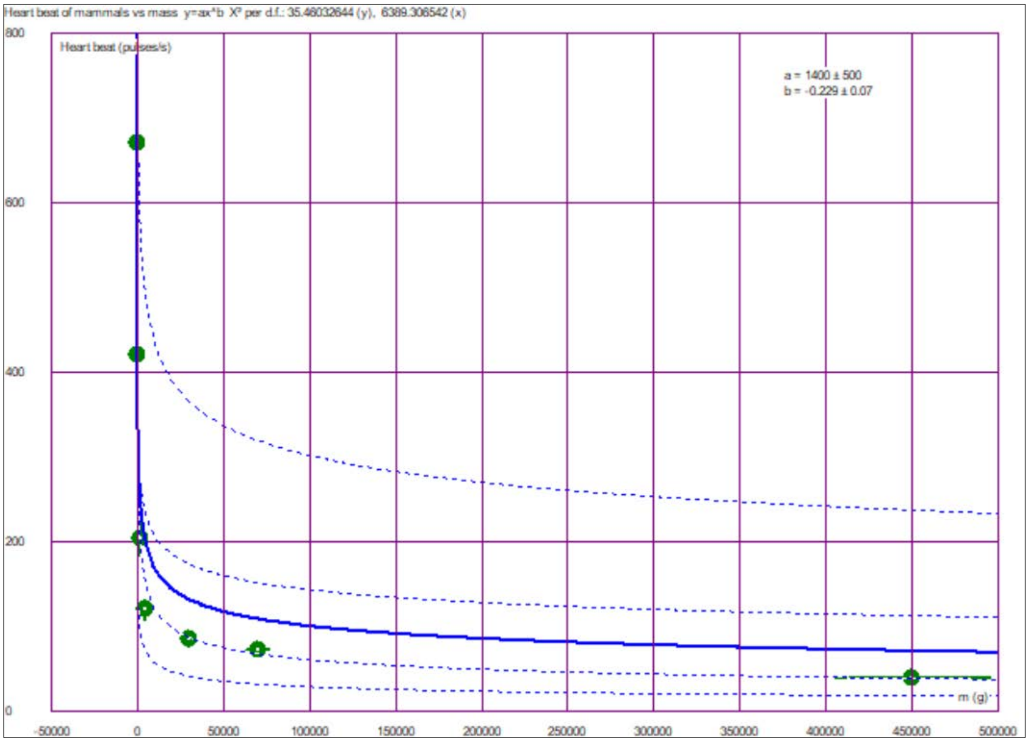


Fig 5: Classic fitting yields: a = 1400 ± 430, b = -0.229 ± 0.063

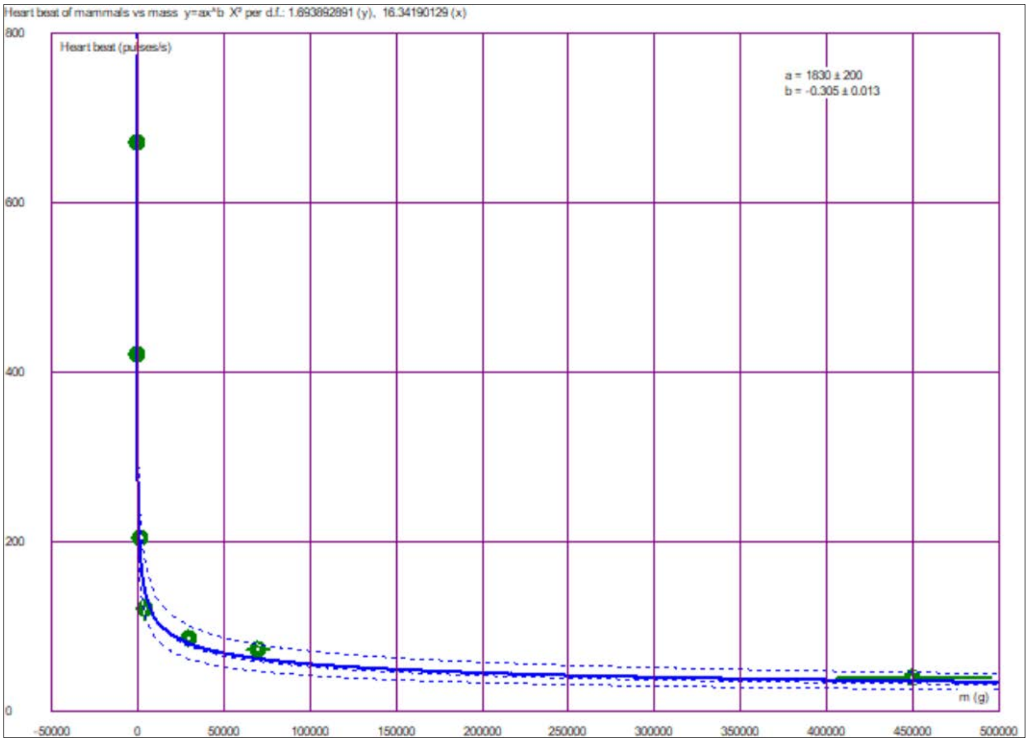


Fig 6: Multidirectional fitting yields: a = 1830 ± 230, b = -0.305 ± 0.015

Example 2: Air pressure (p, in hPa) versus temperature (θ, in °C), in a closed jar; the classical experiment to find the absolute zero temperature 0 Kelvin (-273.15°C). Model function: p = aAθ + b.

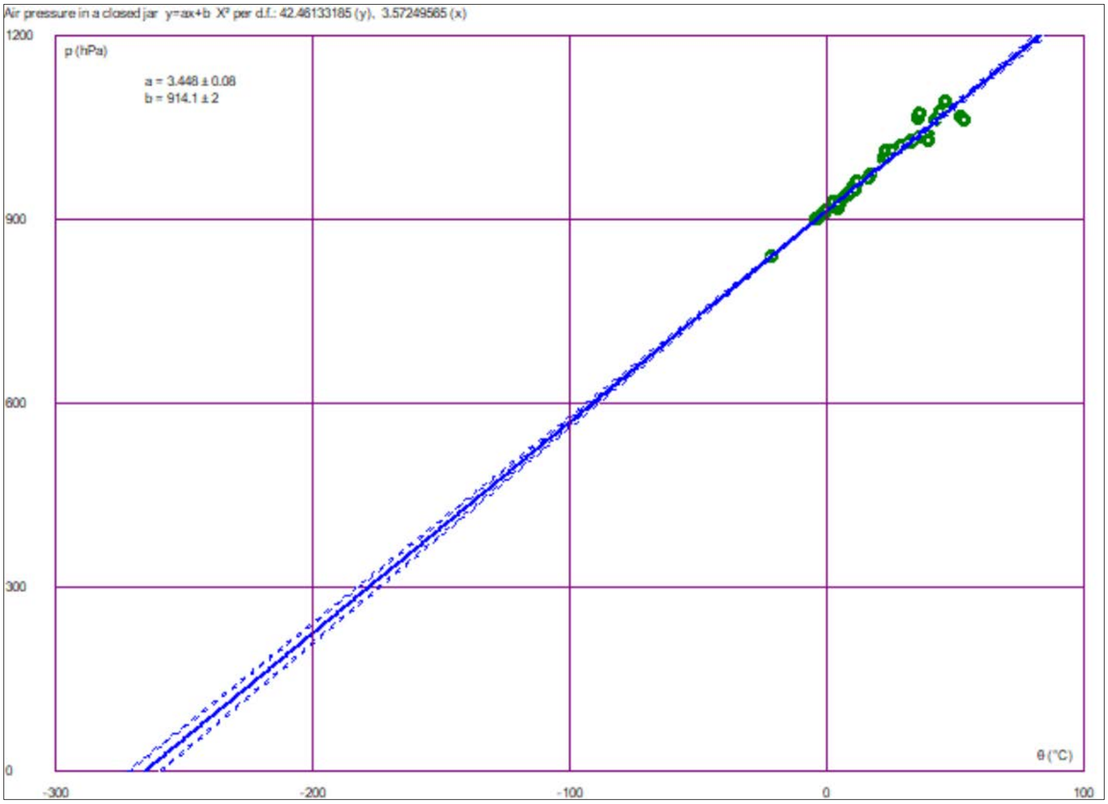


Fig 7: Classic fitting: a = 3.448 ± 0.076, b = 914.1 ± 1.8, Estimate of 0K = -b/a = -265.2 ± 5.9

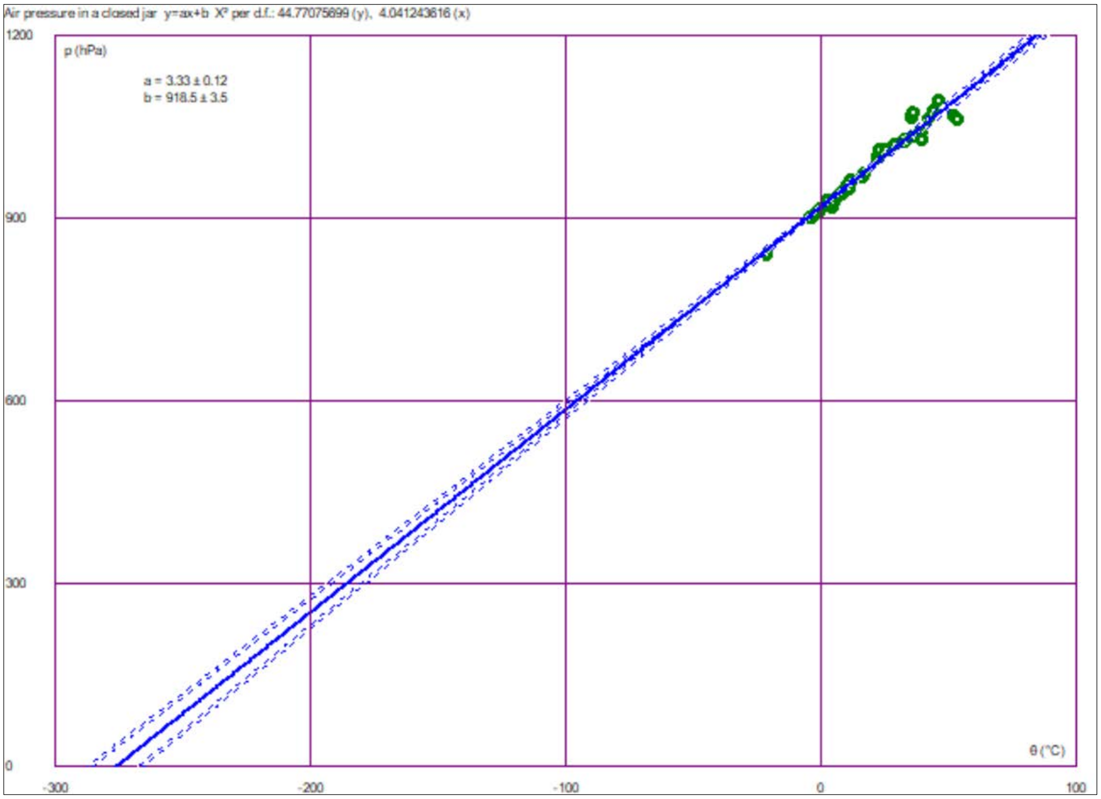


Fig 8: Multidirectional fitting: a = 3.33 ± 0.11, b = 918.5 ± 3.1, Estimate of 0K = -b/a = -275.9 ± 9.5

Example 3: Angle of refracted laser beam (α_1, in °) versus the incoming angle (α_1, in°). The object that was hit by the (532nm wavelength) beam was a cd box, which is made of polystyrene.

Model function: α_2 =Arcsin (sin (α_1)/n, n = refractive index, expected value: 1.5983 according to www.refractiveindex.info.

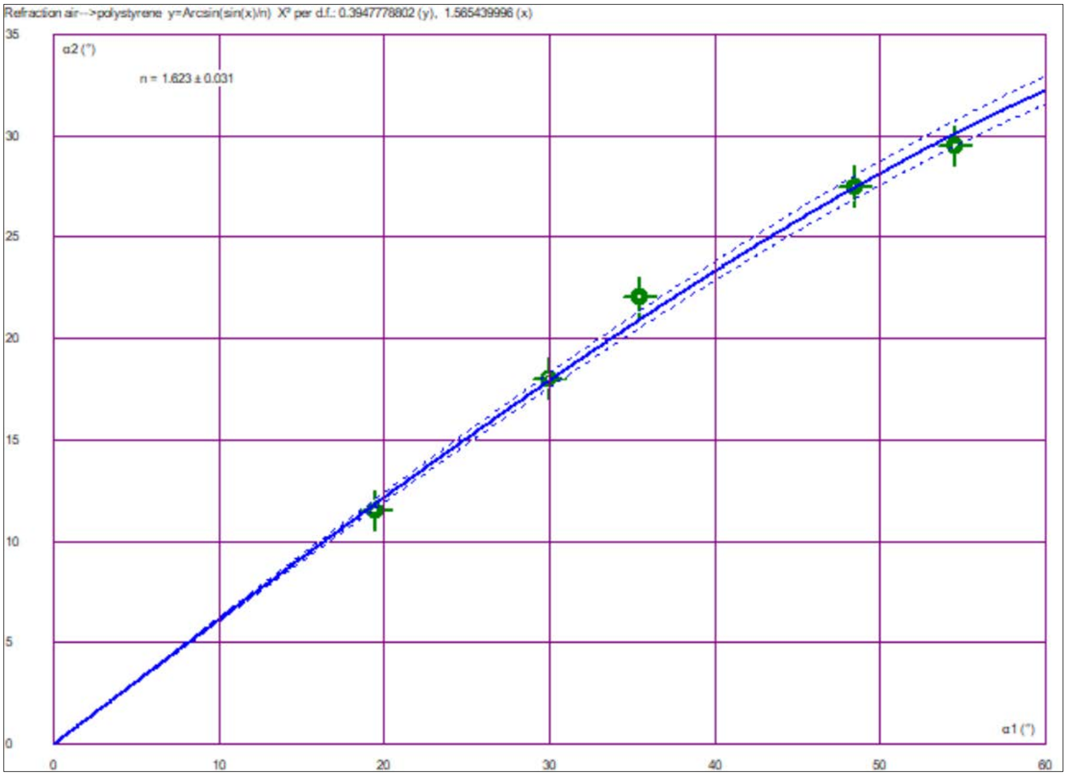


Fig 9: Classic fitting: n = 1.623 ± 0.034

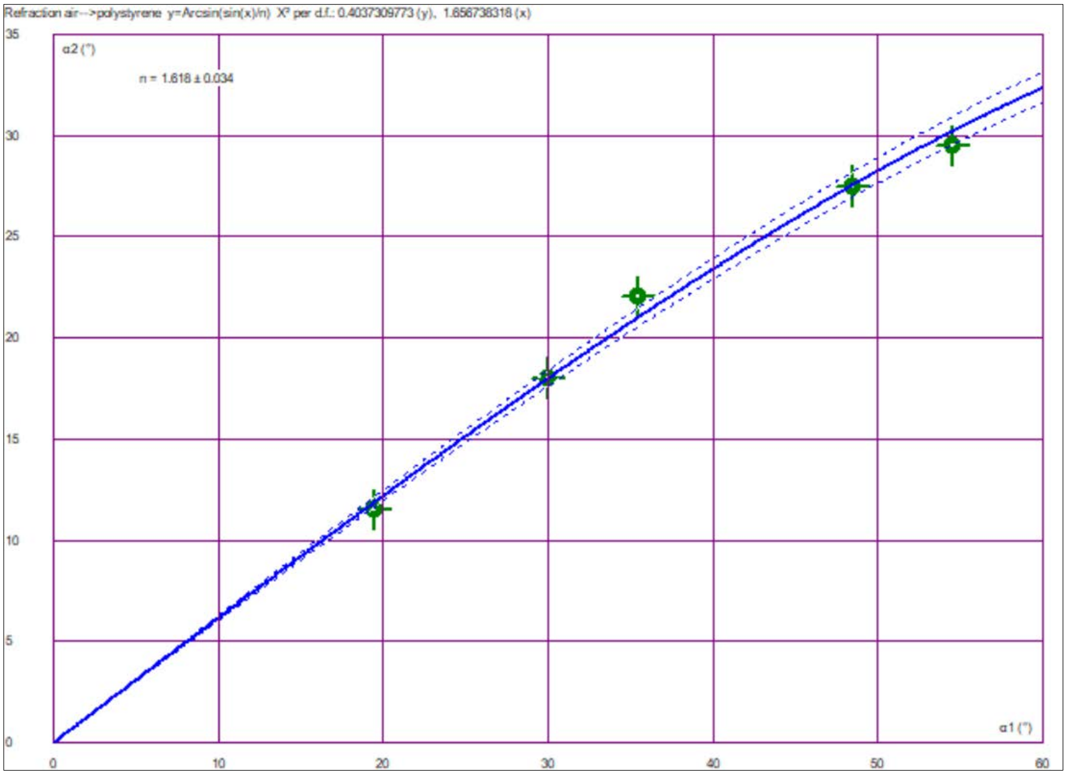


Fig 10: Multidirectional fitting: n = 1.618 ± 0.038

Example 4: Basal metabolic rate (M, in kcal/day) of some Mammal's vs mass (m, in kg) (Data: Max Kleiber 1947). Model function: M = aAm^b.

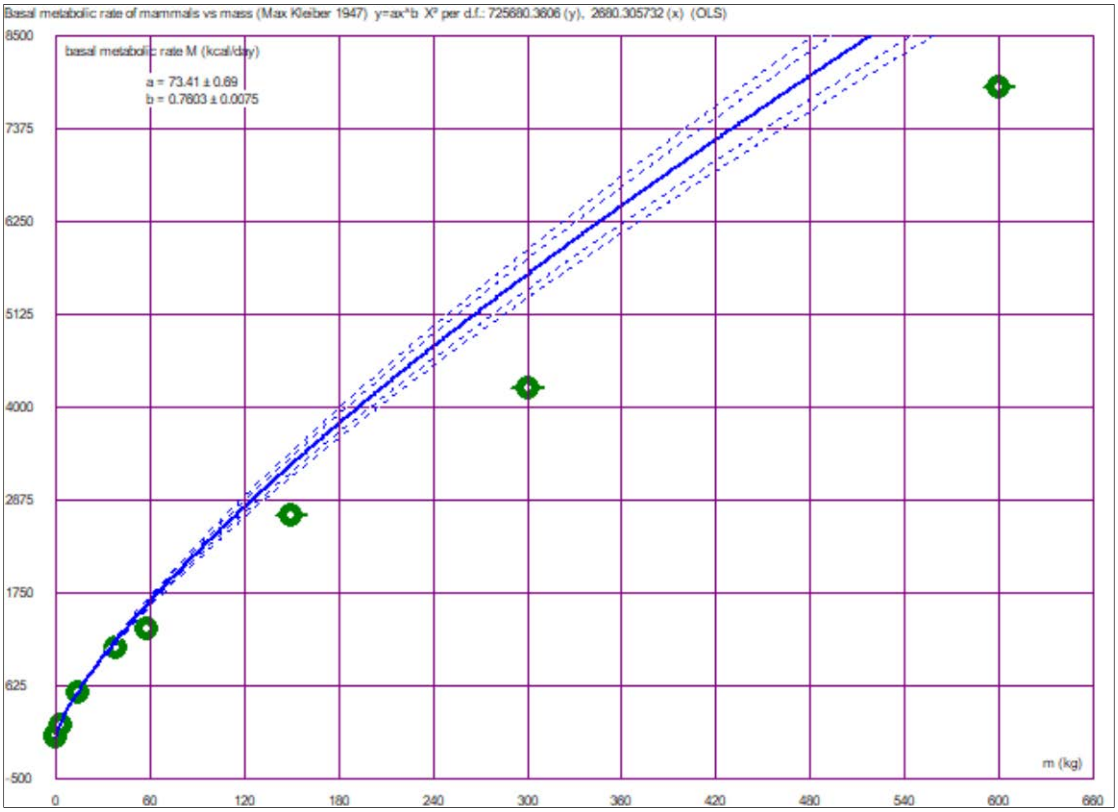


Fig 11: Classic (weighted!) fitting: a = 73.41 ± 0.69, b = 0.7603 ± 0.0075

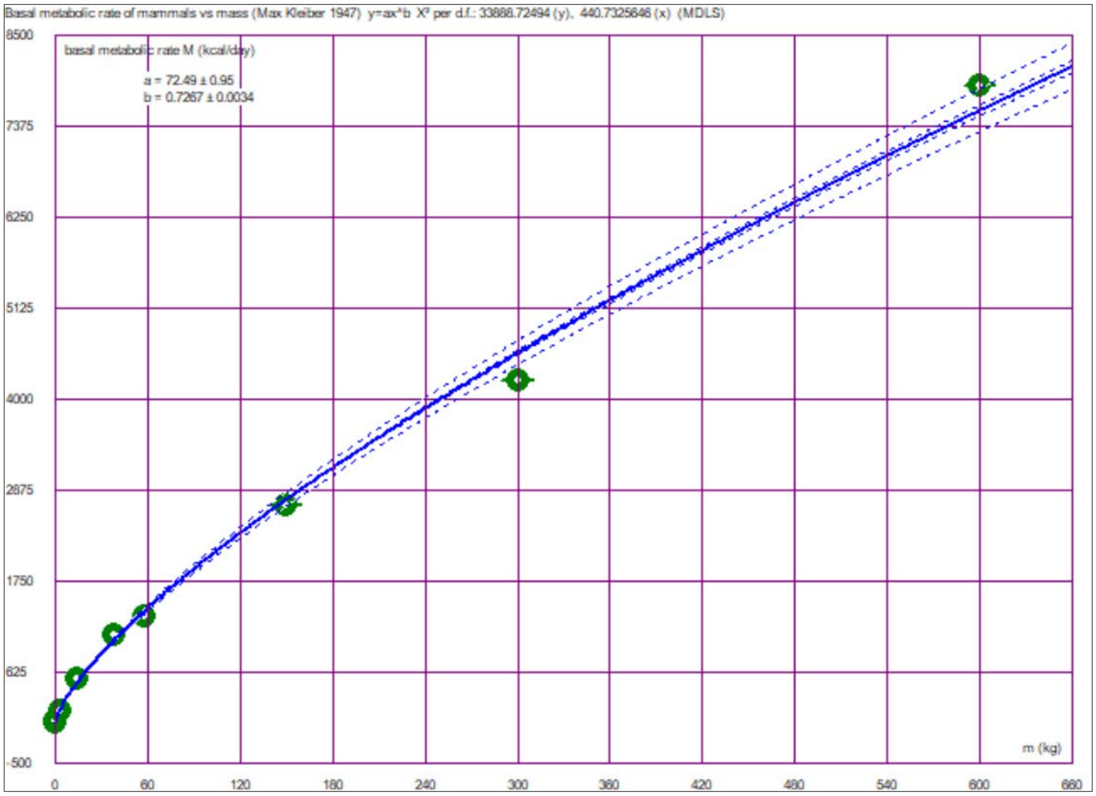


Fig 12: Multidirectional fitting (same weights): a = 72.49 ± 0.95, b = 0.7267 ± 0.0034

Discussion

I hope I have awakened your interest and you will be curious to test this method with your own data. Upon your request by e-mail, I will be happy to send you an evaluation copy of my software, including some example data files. It can also be downloaded from: www.lerenisplezant.be/fitting.htm.

The big question is, of course: in which cases it would be recommendable to use it?

In order to calculate the necessary f-1(yi) values in a unique way, the model function needs to be invertible, or the domain has to be limited to obtain this.

Typical commonly used examples: the linear function (if the slope cannot be zero), power, exponential and logistic function. Some rational functions are okay too, if the domain is limited to avoid vertical asymptotes, but that is no new problem. Y values near horizontal asymptotes might cause some difficulties, but if the 'dangerous' parameters are restricted within safe boundaries, it works.

Non-invertible functions with extrema, especially periodic functions, are not to be used if data from enough periods are available; only if a lot of data points from one phase is available.

So, practically, in all cases where you could just as well a variable y in function of x as x in function of y, use it! Now you may wonder what to do in case x is time? Time is usually seen as 'the' independent variable. And that is a good idea if you have no clue about how your y variable will change with time. For example: suppose y = the percentage of vegetarians in your country. You may fit a straight line through your data to find out if there is some trend going on. But that line doesn't have much reliable predictive value, since there is no direct causal relationship between the time and that percentage. On the other hand, if y is the voltage over a charging capacitor, there *is* such a relationship; you might actually use your voltmeter as a clock to cook your eggs if you calibrate it correctly! In that case, you can definitely use multidirectional fitting, I tried it.

Sebastian Kranz [Kranz 2018] [5], as well as Nick Trefethen and Nir Krakauer [personal communications], suggest to use the so called 'orthogonal or Deming regression' as a solution. Instead of using the vertical distances between the data points and the curve, the 'perpendicular' distances are used in that algorithm. At first sight, that seems to make sense, but first of all, that distance is much more time consuming to calculate (except for some simple functions, iterative approximation is needed), and there is a more fundamental issue with this: how do you define orthogonality if x is height and y is mass, or x is time and y is voltage? Which angle is 90° must not depend on the chosen units! My method avoids this problem.

All your remarks and suggestions are most welcome, of course.

Conflict of Interest

The author declares no conflict of interest.

References

1. Babar Sultan. "Evaluating the Performance of 4 Indices in Determining Adiposity". Clinical Journal of Sport Medicine, Lippincott Williams & Wilkins,2015:25(2):183.
2. Bijma Fetsje, Jonker Marianne, van der Vaart Aad. "An Introduction to Mathematical Statistics", Epsilon, 2016. ISBN: 978-90-5041-135-6
3. Cohen Harold. "Numerical Approximation Methods", Springer, 2011. ISBN: 978-1-4419-9836-1
4. Dukkipati Rao V. "Numerical Methods", New Age International Publishers, 2010. ISBN: 978-81-224-2978-7
5. Kranz Sebastian. "About a curious feature and interpretation of linear regressions" on his Economics and R Blog, 2018.
6. skranz.github.io//r/2018/10/29/Curious_Regression.html
7. Rohrer Fritz. "Der Index der Körperfülle als Maß des Ernährungszustandes". Münchner Med. WSCHR, 1921:68:580–582.
8. Trefethen Nick. On his own website, 2013. people.maths.ox.ac.uk/trefethen/bmi.html
9. Van de moortel, Koen. "Non-linear regression - Why you shouldn't take the logarithms of your variables", 2021. DOI: 10.13140/RG.2.2.18442.80324
10. www.researchgate.net/publication/349324179_Non-linear_regression_-Why_you_shouldn't_take_the_logarithms_of_your_variables.