Preflight Summary-Report für: sample.pdf

Profil: Konformität mit PDF/A-1a prüfen (Geprüfte Seiten 1 bis 6)

Geprüft von axel_2, Datum: 06.03.2012 08:03

Ergebnisse (Zusammenfassung)

✓ **Keine Probleme gefunden**

Dokumentinformation

Dateiname: "sample.pdf"
Pfad: "Z:\pdfa\tex\latex_new"
PDF-Versionsnummer: "1.4"
Größe der Datei (in KB): 238.6
Titel: "General Least Squares Fitting"
Erstellt mit: "LaTeX with hyperref package"
Erzeugt mit: "Acrobat Distiller 10.1.2 (Windows)"
Erstellt: "06.03.2012 07:56"
Geändert: "06.03.2012 07:56"
Überfüllung: "Unknown"
Anzahl der Farbauszüge: 4
Namen der Farbauszüge: "(Cyan) (Magenta) (Yellow) (Black) "

Umgebung

Preflight, 10.1.0 (088)
Acrobat-Version: 10.12
Betriebssystem: Microsoft Windows Service Pack 1 (Build 7601)

Chapter 1

General Least Squares Fitting

1.1 Introduction

1.1.1 Non-linear

Linearizable

some equations, such as $x \ln y$

$$y = Ae^{(Bx)} \tag{1.1}$$

can be treated fairly simply. Linearize and do a linear least squares fit, as you have done in the past. (Note: "Least Squares" applies to transformed quantities, not original ones so gives a different answer than you would get from a least squares fit in the untransformed quantities; remember in general the idea of a line of "best fit" is not unique. For example, for the equation shown, $\ln y$ vs. x is linear, so you can do a least squares fit in $\ln y$, but this will not give the same result as a least squares fit in y, since the sum of squares of $\ln y$ will depend on the data in a different way than the sum of squares in y.)

1.1.2 Linear

General

Some equations, such as this

$$y = b_0 + b_1 x_1 + b_2 x_2 + \cdots b_k x_k \tag{1.2}$$

are linear, although in multiple variables. We can create a matrix of independent data

$$A = \begin{pmatrix} 1 & x_{11} & x_{12} & \dots & x_{1k} \\ 1 & x_{21} & x_{22} & \dots & x_{2k} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n1} & x_{n2} & \dots & x_{nk} \end{pmatrix} \tag{1.3}$$

from the x values, where x_{ij} means variable x_j for data point i and form a vector of dependent data

$$b = \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix} \tag{1.4}$$

where y_i is the y data for data point i.

This creates a system which can be solved using the "regression" feature of a spreadsheet. (Be sure to disable the calculation of the y–intercept, as the first coefficient calculated will be the y-intercept, and standard errors will be given for each parameter.)

Polynomial

Consider an equation such as this:

$$y = b_0 + b_1 x + b_2 x^2 + \cdots b_k x^k \tag{1.5}$$

This is just a special case of the previous situation above, eg. $x_1 = x$, $x_2 = x^2$, $x_3 = x^3$, etc. (or $x_1 = 1/x$, $x_2 = 1/x^2$, etc.)

What about fit with skipped orders? — eg. $y = a + b/x^2 + c/x^5$

In this case, $x_1 = 1/x^2$, $x_2 = 1/x^5$.

1.1.3 Goodness of fit

Often you must choose between different fits because you do not know what type of equation to use. In this case you want to be able to answer the question "Which fit is better?"

1. If both fits have the same number of parameters, then the better fit is the one with the smaller SSE in the *same* quantity. (In other words, if you're comparing a fit in y vs. x to one in $\ln y$ vs. x, you will first have to calculate the SSE of both in y vs. x. If you have linearized an equation to calculate a fit, you can still use that fit to calculate the SSE in the original quantity afterward.)

2. One or both of the fits may have some parameters which are not "statistically significant"; (i.e. lots of parameters close to 0 are probably meaningless.) How close to 0 is "close enough"?

    - RULE: Adding more parameters → smaller SSE, (however a small change in SSE may not be significant.) Whether or not the added parameters are significant can be determined statistically *if the fit is a linear one or one which can be linearized.*

The following example illustrates how to do this for a linear fit.

Example

Consider the data shown in Table 1.1 and plotted in Figure 1.1. (Error bars have been omitted for simplicity.)

It should be obvious that some possible equations for a fit to this data may be polynomials in $1/x$.

Some of these are shown in some other Figures.

1. Do fit with $g + 1$ parameters (as above); calculate the sum of squares error and call it SSE_1. If you use regression from a spreadsheet, you can determine SSE from the results. Remember $SSE_1 = {s_1}^2\nu_1$; in this case $\nu_1 = n - (g + 1)$.

    This gives us the next Table and the resulting graph is not shown.

    Notice that the curve cannot "bend enough", and so we will see what happens if we add another parameter.

x	y
100	1
85	2
70	4
50	8
36	15
20	25
10	45

Table 1.1: Sample Data

Figure 1.1: The logo of TeX

In our case, to compare the 2 parameter fit to the three parameter fit we do this by creating a matrix

$$A_1 = \begin{pmatrix} 1 & 1/x_1 \\ 1 & 1/x_2 \\ \vdots & \vdots \\ 1 & 1/x_n \end{pmatrix} \tag{1.6}$$

and we solve as described earlier.

2. Do fit with $k+1$ parameters (as above); calculate SSE_2. As above, $SSE_2 = {s_2}^2\nu_2$ and in this case $\nu_2 = n - (k+1)$.

   In our case, we do this by creating a matrix

$$A_2 = \begin{pmatrix} 1 & 1/x_1 & 1/x_1^2 \\ 1 & 1/x_2 & 1/x_2^2 \\ \vdots & \vdots & \vdots \\ 1 & 1/x_n & 1/x_n^2 \end{pmatrix} \tag{1.7}$$

   and repeat.

3. Calculate s_3 as follows:

$$s_3 = \sqrt{\frac{SSE_1 - SSE_2}{k-g}} \tag{1.8}$$

   and let $\nu_3 = k - g$.

4. Calculate F as follows:

$$F = \frac{{s_3}^2}{{s_2}^2} \tag{1.9}$$

   If F is *big*, then include the extra parameters. (In this case, it means the SSE changed a lot by adding the extra parameters, which is what would happen if they were really important.) How big is “big”?

5. Look up F_{α,ν_3,ν_2} from a table of the F distribution in a statistics text, where α determines the confidence interval; typically $\alpha = 0.05$ for a 95% confidence interval. If the F you calculated is *greater* than the table value, then keep the extra parameters. *Note: In the table, you are given quantities ν_1 and ν_2; you should use your calculated value of ν_3 in place of ν_1 in the table. Doing it this way keeps the table in the same form you will find it in a statistics text.*

(Note that in some of the figures, the fit curve is not quite smooth, due to an insufficient number of plotting points used.) It is not immediately obvious which of the above curves fits the data "best". We could even go on adding higher and higher powers of $1/x$ until we had no more *degrees of freedom*[1] left, but once we get no significant change, it's time to stop.

Usually we want to compare two fits; in this example, we will compare 3 fits to illustrate the process more clearly. We will compare 2 fits at a time, and in each case we will use $g+1$[2] to denote the number of parameters in the "smaller" fit, and $k+1$ to denote the number of parameters in the "bigger" fit, so k is always bigger than g.

Now you can go to any of the equations in this document:

- Equation 1.1
- Equation 1.2
- Equation 1.3
- Equation 1.4
- Equation 1.5
- Equation 1.6

[1] The number of degrees of freedom in a fit is the number of data points beyond the bare minimum for that fit. So, for an average it is $n-1$, since only one value is needed; for a straight line it is $n-2$, since two points are needed, etc. In general,

$$\nu = n - m \tag{1.10}$$

where m is the number of parameters in the fit to be determined. Note that when you have no degrees of freedom, you have no idea of the "goodness" of your data, and thus cannot determine the standard deviation. Once you have even one degree of freedom, you can do so.

[2] Why not just g? Because g is the *degree* of the polynomial, which has $g+1$ parameters. For example a polynomial of degree 2, such as $Ax^2 + Bx + C$ has 3 parameters, namely A, B, and C.

- Equation 1.7
- Equation 1.8
- Equation 1.9
- Equation 1.10

(Note that the link will put the referenced item at the very top of the view.)

Compositatest none-none-none-none-none-none-none-none-none-none-none-none-none-none-none-none-none-none-none-none-none-none-none-none-none-none Previously you have done curve fitting in two dimensions. Now you will learn how to extend that to multiple dimensions.

none none none none none none none none none none none none Nord-Suedpol