Tutorials and worked examples for simulation,
curve fitting, statistical analysis, and plotting.
http://www.simfit.org.uk

Polynomial regression is used for data smoothing, detecting trends in noisy data, and for creating calibration curves for inverse prediction. It is not much used for modeling data, as polynomial curves are too flexible, they do not accommodate horizontal asymptotes, and they cannot be used for extrapolation. In many applications nowadays they have been replaced by piecewise cubic splines.

From the main SimFIT menu choose the [A/Z] option, open program **polnom**, then browse the default test file `polnom.tf1` which contains the following data set.

x	y	s
0.0	0.098421	0.0056072
0.0	0.10950	0.0056072
0.0	0.10248	0.0056072
2.0	3.8448	0.052139
2.0	3.8647	0.052139
2.0	3.9434	0.052139
4.0	6.8490	0.38867
4.0	6.1469	0.38867
4.0	6.2091	0.38867
6.0	8.5864	0.22982
6.0	9.0156	0.22982
6.0	8.6585	0.22982
8.0	9.8616	0.45524
8.0	9.8748	0.45524
8.0	9.0798	0.45524
10.0	9.5218	0.51790
10.0	9.3098	0.51790
10.0	10.294	0.51790

The columns are for data simulated by SimFIT according to $y = 0.1 + 2.0x + 0.1x^2$ and have the following meanings.

1. The first column contains the independent variable x_i in triplicate.
2. The second column contains the dependent variable y_i arising from evaluating the model equation using SimFIT program **makdat**, then adding 5% relative error using SimFIT program **adderr** to simulate experimental error.
3. The third column are the sample standard deviations s_i calculated by SimFIT program **adderr** to use for weights $w_i = 1/s_i^2$. In the absence of replicates to calculate sample standard deviations for y_i at fixed x_i, the third column could be replaced by $s_i = 1$, or simply omitted, whereupon a default value of $s_i = 1$ would be used for unweighted regression.

Program **polnom** will then proceed to fit polynomials of degree m according to

$$f(x) = \theta_0 + \theta_1 x + \theta_2 x^2 + \theta_3 x^3 + \cdots + \theta_6 x^6$$

for $m = 0, 1, 2, \ldots, k$ where $k \leq 6$ depends on the number of distinct values of x. That is, $m = 0$ for a constant term, $m = 1$ for a straight line, $m = 2$ for a quadratic, $m = 3$ for a cubic, and so on. After fitting each degree, several statistics are output to assess goodness of fit and determine the highest degree that can be justified.

The idea of this systematic procedure is to determine if there is statistical evidence to justify a trend line or progressive curvature in noisy data, or to select a model equation to use as a calibration curve for inverse prediction. To appreciate this aspect consider the following results tables when the data are analyzed.

Table 1: Degree fitted and Chebyshev coefficients

m	A_0	A_1	A_2	A_3	A_4	A_5
0	0.31113					
1	16.034	7.9080				
2	12.737	4.8194	-1.4456			
3	12.735	4.8132	-1.4591	-0.0083774		
4	12.762	4.8342	-1.4387	-0.055083	-0.059600	
5	12.654	4.6602	-1.3858	-0.087456	-0.035275	0.22979

Another table of statistics required to determine the degree of the polynomial required is also displayed as follows.

Table 2: Statistics to determine degree of the fitted polynomial

m	σ	%change	$WSSQ$	%change	$P(\chi^2 \geq WSSQ)$	5%	FV	$P(F \geq FV)$	5%
0	36.703		22901		0.0000	no			
1	8.0833	77.98	1045.4	95.44	0.0000	no	334.50	0.0000	yes
2	0.9914	87.73	14.744	98.59	0.4700	yes	1048.6	0.0000	yes
3	1.0253	3.42	14.718	0.18	0.3977	yes	0.0249	0.8769	no
4	1.0511	2.52	14.363	2.41	0.3488	yes	0.3213	0.5805	no
5	1.0000	4.87	11.999	16.46	0.4457	yes	2.3639	0.1501	no

Here m is the degree fitted, $\sigma = \sqrt{WSSQ/NDOF}$, and FV is the F value for assessing the significance of variance reduction by adding higher degree terms.

There are many results displayed in Tables 1 and 2 in order to suggest the highest degree that can be justified statistically. The qualitative conclusions do not use a Bonferroni correction, but the actual significance levels are also provided for purists. At this point SIMFIT program **polnom** outputs the next table to aid decision.

Table 3: information to help you select a best-fit polynomial

Lowest degree where < 10% change in σ	2
Lowest degree where < 10% change in $WSSQ$	2
Lowest degree by chi-sq. at 5% significance level	2
Lowest degree by chi-sq. at 1% significance level	2
Lowest degree by F test at 5% significance level	2
Lowest degree by F test at 1% significance level	2

Accepting the recommendations of Table 3 leads to Table 4 for the best-fit quadratic.

Table 4: Results for weighted fitting $(w = 1/s^2)$

Parameter	Value	Std. error	Lower95%cl	Upper95%cl	p
θ_0	0.10347	0.0032091	0.096630	0.11031	0.0000
θ_1	2.1203	0.019731	2.0783	2.1624	0.0000
θ_2	-0.11565	0.0035714	-0.12326	-0.10803	0.0000

Correlation matrix

1		
-0.0960	1	
0.0516	-0.8432	1

If you selected to predict x from y the following warning is issued.

> You must be very careful if you wish to use this best-fit curve as a calibration curve for predicting x given y since there are turning points for $X_{min} \le x \le X_{max}$ as follows:

x-value	y-value
9.1673	9.8224

This is because the quadratic has a turning point within the range of the data, and so predicting x from y could be misleading if a horizontal line for $y = y_0$ for some y_0 intersected the best fit curve twice. So you have to choose whether to search upwards or downwards along the x axis for the prediction required. If a spurious prediction results you have to change the search order. For degrees greater than two there may be multiple turning points, so using degrees greater than two is not normally recommended for inverse prediction. Table 5 results from choosing to predict x from y and evaluate y given x along with 95% confidence ranges using the data supplied in test files `polnom.tf2` and `polnom.tf3`.

Table 5: Predicting x given y and evaluating y given x

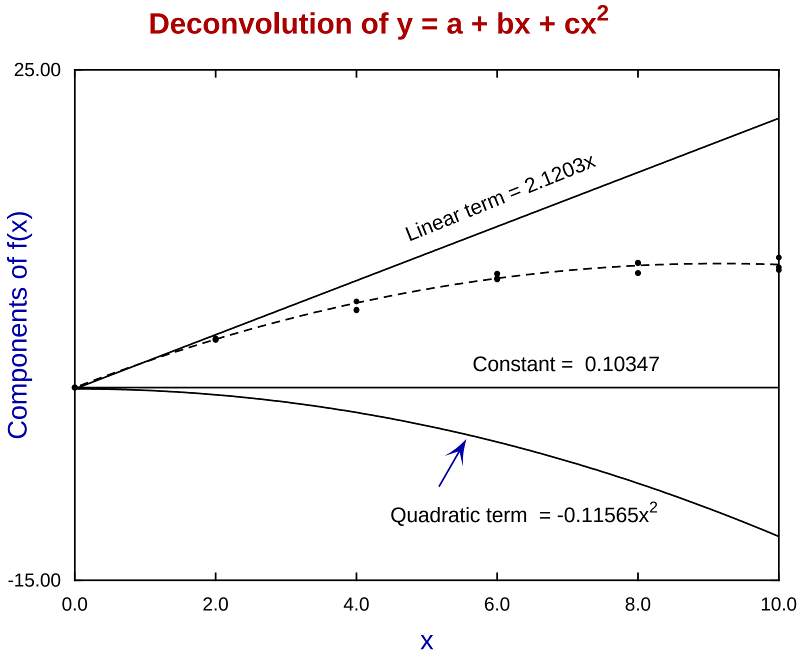
Evaluation data for program **polnom**: x = 2, 4, 6, 8

x-input	y-calculated	95% confidence limits
2.0	3.8816	3.8212, 3.9419
4.0	6.7345	6.6424, 6.8267
6.0	8.6623	8.5137, 8.8108
8.0	9.6649	9.3927, 9.9370

Inverse prediction data for program **polnom**: y = 2, 4, 6, 8

y-measured	x-predicted	95% confidence limits
2.0	0.94293	0.92529, 0.96118
4.0	2.0718	2.0347, 2.1100
6.0	3.4182	3.3566, 3.4819
8.0	5.1976	5.0739, 5.3342

This next graph, constructed using SIMFIT program **qnfit**, shows the data and best-fit quadratic along with contributions of the individual components to the fit.



Theory

It is possible to fit polynomials using multilinear regression with a constant term but with variables defined as $x_1 = x, x_2 = x^2, x_3 = x^3, \ldots, x_m = x^m$. However, this is regarded as an inefficient and numerically inaccurate technique. The best technique is to transform the original variables x into new variables $-1 \leq \tilde{x} \leq 1$ according to

$$\tilde{x} = \frac{2x - x_{max} - x_{min}}{x_{max} - x_{min}}$$

Then a polynomial of degree m is fitted using Chebyshev polynomials as follows

$$g_m(\tilde{x}) = 0.5A_{m+1,1}T_0(\tilde{x}) + A_{m+1,2}T_1(\tilde{x}) + A_{m+1,3}T_2(\tilde{x}) + \cdots + A_{m+1,m+1}T_m(\tilde{x}).$$

In this expression the $T_k(\tilde{x})$ are Chebyshev polynomials of the first kind of degree k defined as follows.

$$\begin{aligned}
T_k(x) &= \cos(k\cos^{-1}(x)), k \geq 0\\
T_{k+1}(x) &= 2xT_k(x) - T_{k-1}x, k \geq 1\\
\text{For instance, } T_0(x) &= 1\\
T_1(x) &= x\\
T_2(x) &= 2x^2 + 1\\
T_3(x) &= 4x^3 - 3x.
\end{aligned}$$

The magnitude of the coefficients $A_{m+1,j}$ indicates the contribution of the corresponding Cheyshev polynomial to the corresponding power of x. When fitting polynomials sequentially the coefficients $A_{m+1,j}$ will tend to stabilize for powers of x that are contributing to the fit, but will often tend to diminish as further irrelevant powers are added to the polynomial. So Table 1 provides a quick method for assessing the highest degree polynomial required for a satisfactory fit. Of course, the coefficients and best-fit curve are transformed back into the original space after a satisfactory degree has been decided.

The techniques used by SIMFIT for calculating confidence limits for evaluation and inverse prediction are based on extending the methods used for standard unweighted straight line fitting to the case of fitting polynomials to weighted data.