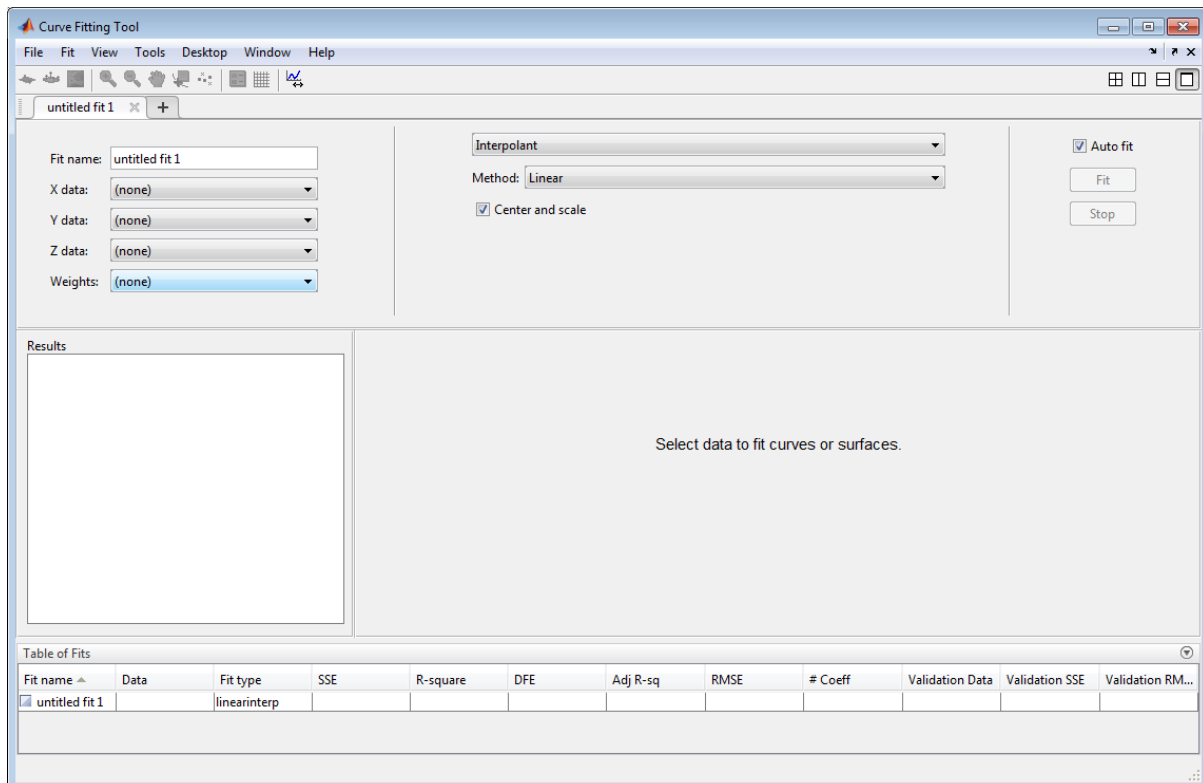
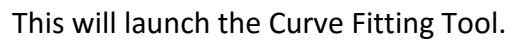


MATLAB has a curve fitting tool which, as well as fitting to standard models such as linear, polynomial, exponential, etc., can fit to custom equations. To launch the curve fitting tool, click “Curve fitting” under the “Apps” menu.



Curve Fitting Tool

File Fit View Tools Desktop Window Help

untitled fit 1

Fit name: **untitled fit 1**

X data: x

Y data: y

Z data: (none)

Weights: (none)

Polynomial

Degree: 1

Robust: Off

☐ Center and scale

☒ Auto fit

Fit

Stop

Fit Options...

Results

Linear model Poly1:
 $f(x) = p_1x + p_2$
 Coefficients (with 95% confidence bounds):
 $p_1 = 1.556e-18$ (-0.0002142, 0.0002142)
 $p_2 = 1.617$ (1.493, 1.74)

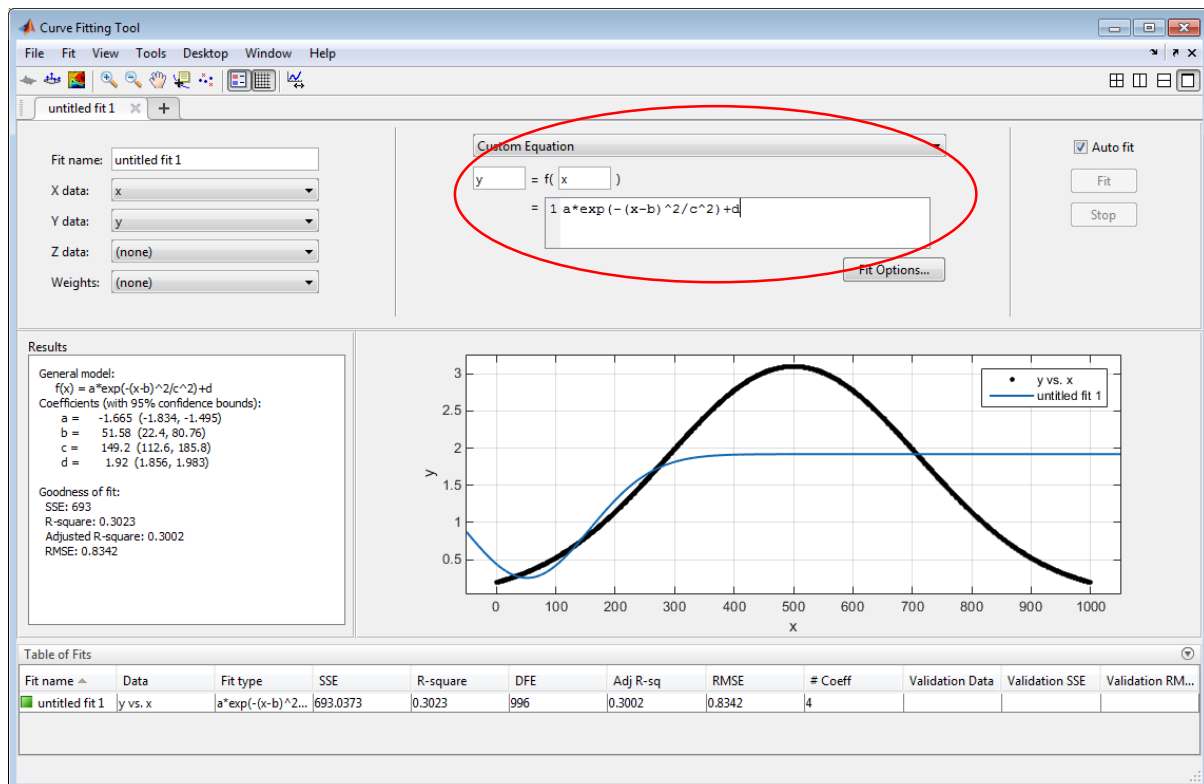
Goodness of fit:
 SSE: 993.3
 R-square: 2.22e-16
 Adjusted R-square: -0.001002
 RMSE: 0.9976

Table of Fits

Fit name	Data	Fit type	SSE	R-square	DFE	Adj R-sq	RMSE	# Coeff	Validation Data	Validation SSE	Validation RM...
untitled fit 1	y vs. x	poly1	993.2684	2.2204e-16	998	-0.0010	0.9976	2			

The screenshot shows the Curve Fitting Tool (CFT) interface. The 'Custom Equation' dropdown is highlighted with a red circle. The equation editor shows the equation $y = -(x)$ and $= 1$. The 'Results' panel shows an error message: 'Cannot create an equation from an empty expression.' The plot shows a bell-shaped curve labeled 'y vs. x'. The 'Table of Fits' at the bottom shows one fit named 'untitled fit 1' with 'y vs. x' data.

We are then presented with a box in which we can type a custom equation (Note: you don't use the MATLAB variable names you selected for the x/y data here. Instead use the names specified in the text boxes just above the "custom equation" box).



The fit may not immediately work. In this case, you may need to provide better initial guesses for the *free parameters*. You can do this via the "Fit Options" button, which opens a dialog allowing you to specify the initial guess and the boundaries of these parameters. Here we provided a better initial guess for "b", the x offset in our equation.

Fit Options

Method: NonlinearLeastSquares

Robust: Off

Algorithm: Trust-Region

DiffMinChange: 1.0e-8

DiffMaxChange: 0.1

MaxFunEvals: 600

MaxIter: 400

TolFun: 1.0e-6

TolX: 1.0e-6

Unknowns	StartPoint	Lower	Upper
a	0.3554	-Inf	Inf
b	500	-Inf	Inf
c	0.9248	-Inf	Inf
d	0.9134	-Inf	Inf

Close

This then results in an excellent fit, and you can read off the values (and the 95% confidence bounds) from the results text box on the left hand side of the curve fitting tool window.

