# SOLUTION OF REGRESSION MODELS

## Polynomial Regression Model

**2007 *Fabian Farelo , Autar Kaw, Jamie Trahan***
**University of South Florida**
**United States of America**
**kaw@eng.usf.edu**
**http://numericalmethods.eng.usf.edu**

Note: This Worksheet simulates the polynomial regression model. It also helps you tc find the optimum order of polynomial to use.

### Introduction

Given $n$ data points $(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$ use least squares method to regress the data to a $m^{th}$ order polynomial.

$$y = a_0 + a_1x + a_2x^2 + \ldots + a_mx^m, \;\; m<n \qquad (1)$$

The residual at each data point is given by

$$E_i = y_i - a_0 - a_ix^i \ldots.-a_mx_i^m \qquad (2)$$

The sum of the square of the residuals is given by

$$S_r = \sum_{i=1}^{n} (E_i)^2 \qquad (3)$$

To find the constants of the polynomial regression model, we set the derivatives with respect to $a_i$ equal to zero, that is,

$$\frac{\partial}{\partial a_0}S_r = \sum_{i=1}^{n} \left[2\cdot\left[y_i - a_0 - a_1x_i - .. - a_m(x_i)^m\right](-1)\right] = 0 \qquad (4.a)$$

$$\frac{\partial}{\partial a_1}S_r = \sum_{i=1}^{n} \left[2\cdot\left[y_i - a_0 - a_1x_i - .. - a_m(x_i)^m\right](-x_i)\right] = 0 \qquad (4.b)$$

. . . .
. . . .

$$\frac{\partial}{\partial a_m} S_r = \sum_{i=1}^{n} \left[ 2 \cdot \left[ y_i - a_0 - a_1 x_i - .. - a_m \left(x_i\right)^m \right] \left[ -\left(x_i\right)^m \right] \right] = 0 \qquad (4.m)$$

Setting those equations in matrix form gives

$$\begin{bmatrix} n & \sum_{i=1}^{n} x_i & .. & \sum_{i=1}^{n} \left(x_i\right)^m \\ \sum_{i=1}^{n} x_i & \sum_{i=1}^{n} \left(x_i\right)^2 & .. & \sum_{i=1}^{n} \left(x_i\right)^{m+1} \\ .. & .. & .. & .. \\ \sum_{i=1}^{n} \left(x_i\right)^m & \sum_{i=1}^{n} \left(x_i\right)^{m+1} & .. & \sum_{i=1}^{n} \left(x_i\right)^{2m} \end{bmatrix} \begin{pmatrix} a_0 \\ a_1 \\ .. \\ a_m \end{pmatrix} = \begin{bmatrix} \sum_{i=1}^{n} y_i \\ \sum_{i=1}^{n} \left(x_i y_i\right) \\ .. \\ \sum_{i=1}^{n} \left[\left(x_i\right)^m y_i\right] \end{bmatrix}$$

The above simultaneous linear equations are solved for the ($m$+1) constants $a_0$, $a_1$, ..., $a_m$. To learn more about polynomial regression see the **Nonlinear Regression** worksheet.

## Section 1: Input data

This is the input data that can be modified by the user. The user can change the values clicking on the table and entering the *x*-coordinate in the first column and the *y*-coordinate in the second column.The order of the polynomial can also be selected b the user.

***Note*****:** the origin has been set to 1 to redefine the starting index of all arrays. The user SHOULD NOT change this value.

ORIGIN := 1

data :=

| | 1 | 2 |
|---|---|---|
| 1 | 80 | 6.47 |
| 2 | 40 | 6.24 |
| 3 | 0 | 6 |
| 4 | -40 | 5.72 |
| 5 | -80 | 5.43 |
| 6 | -120 | 5.09 |
| 7 | -160 | 4.72 |

- Enter the order of the polynomial regression model:

order_poly := 1

- Enter the lowest order of polynomial to check for optimun order:

low_order := 1

- Enter the highest order of polynomial to check for optimun order. **Note:** ***high_order*** **must be ≤ n-1.**

high_order := 6

## Section 2: Extracting the data

The following steps extract data from the input data table, allowing us to use of the values in the procedures below.

$x := data^{\langle 1 \rangle}$

$y := data^{\langle 2 \rangle}$

$n := rows(x)$

$n = 1.1 \times 10^{1}$

## Section 3: Defining the system of simultaneous linear equations in matrix form

In this section, the coefficient matrix "M" and the right hand side vector "B" are calcu and subsequently used to determine the solution vector "a" that contains the coefficie the polynomial model $a_0, a_1, ..., a_m$

- **Calculating the coefficient matrix, "M"**

Creating the matrix size according to the order of the polynomial model:

$M_{order\_poly+1,\, order\_poly+1} := 0$

Finding each value of the coefficient matrix:

$$
M := \left|
\begin{array}{l}
M \leftarrow M \\
M_{1,1} \leftarrow n \\
\text{for } i \in 2 .. \text{order\_poly} + 1 \\
\quad \text{for } j \in 1 .. n \\
\quad \quad M_{1,i} \leftarrow M_{1,i} + \left(x_j\right)^{i-1} \\
\text{for } i \in 1 .. \text{order\_poly} + 1 \\
\quad \text{for } k \in 2 .. \text{order\_poly} + 1 \\
\quad \quad \text{for } j \in 1 .. n \\
\quad \quad \quad M_{k,i} \leftarrow M_{k,i} + \left(x_j\right)^{i+k-2} \\
M
\end{array}
\right.
$$

The values of the coefficient matrix "M" are:

$$M = \begin{pmatrix} 1.1 \times 10^{1} & -1.32 \times 10^{3} \\ -1.32 \times 10^{3} & 3.344 \times 10^{5} \end{pmatrix}$$

- **Calculating the right hand side vector, "B"**

Creating the vector size according to the polynomial model order:

$$B_{order\_poly+1,1} := 0$$

Finding each value of the array:

$$B := \left| \begin{array}{l} B \leftarrow B \\ \text{for } \ i \in 1 .. n \\ \quad B_{1,1} \leftarrow B_{1,1} + y_i \\ \text{for } \ i \in 2 .. order\_poly + 1 \\ \quad \text{for } \ j \in 1 .. n \\ \qquad B_{i,1} \leftarrow B_{i,1} + \left[ \left( x_j \right)^{i-1} \right] \cdot y_j \\ B \end{array} \right.$$

The values for the RHS vector, "B" are

$$B = \begin{pmatrix} 5.389 \times 10^{1} \\ -4.8568 \times 10^{3} \end{pmatrix}$$

**Section 4: Solving the system of simultaneous linear equations M*a = B**

Now that the right hand side vector "B" and coefficient matrix "M" have been calcul they can be used to solve for the solution vector "a" which contains the coefficients polynomial model.

Merging the matrices, "M" and "B" to use Mathcad to reduce it.

$$E := \text{augment}(M, B)$$

Returning the row-reduced echelon form of the augmented matrix, "E".

$$a := \text{rref}(E)$$

Selecting the last column of "E" that contains the solution to the coefficient values.

$$a := a^{\langle order\_poly+2 \rangle}$$

$$a = \begin{pmatrix} 5.9968182 \times 10^{0} \\ 9.1477273 \times 10^{-3} \end{pmatrix}$$

This array contains the coefficients of the model as a = [$a_0$, $a_1$, ...., $a_m$]. The plot show data points as well as the regression model.

Generating an array of *x* and *y* values used to plot the model.

$$N := \frac{\max(x) - \min(x)}{0.001}$$

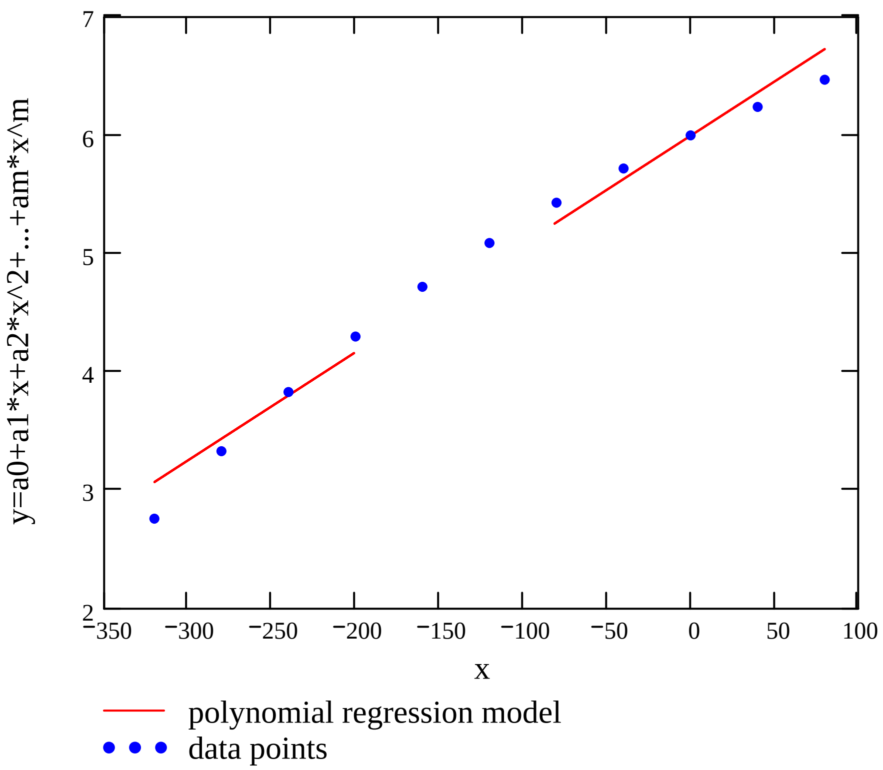
$$X := \left| \begin{array}{l} X_1 \leftarrow \min(x) \\ \text{for } i \in 2..N \\ \quad X_i \leftarrow X_{i-1} + 0.001 \\ X \end{array} \right.$$

$$
Y := \left|\begin{array}{l} Y_{N,1} \leftarrow 0 \\ \text{for } i \in 1..N \\ \quad \left|\begin{array}{l} Y_i \leftarrow a_1 \\ \text{for } j \in 1..\text{order\_poly} \\ \quad Y_i \leftarrow Y_i + a_{j+1} \cdot (X_i)^j \end{array}\right. \\ Y \end{array}\right.
$$

Recall that the desired order of the polynomial regression model is:

order_poly = 1

**Figure 1: Polynomial Regression of order *m***



## Section 5: Optimum Order

In this section we will determine the optimum order of the polynomial model by plotting the variance defined as

$$\frac{S_r}{n-(m+1)}$$

where *n* is the number of data points, $S_r$ is the sum of the square of residuals and *m* is the order of the polynomial. The optimum order is considered as to be the one where the value of the variance $S_r/[n-(m+1)]$ is minimum or where its value is significantly decreasing.

In the following procedure, a polynomial regression model is calculated for each order specified in the *low_order* to *high_order* range. Mathcad then computes the variance of each model.The worksheet does not choose the order of the optimum polynomial for regression for you.  Look at the plot of the variance as a function of the order of the polynomial.  The optimum polynomial is one after which there is no statistical significant decrease in the variance.

Many a times, the variance may show signs of decreasing and then increasing as a function of the order of the polynomial regression model.  Such increases in the variance are normal as the variance is calculated as the ratio between the sum of the squares of the residuals and the difference between the number of data points and number of constants of the polynomial model.  Both the numerator and denominator decrease as the order of the polynomial is increased.  However, as the order of the polynomial increases, the coefficient matrix in the calculation of the constants of the model becomes more ill-conditioned. This ill-conditioning of the coefficient matrix results in fewer significant digits that can be trusted to be correct in the coefficients of the polynomial model, and hence artificially amplify the value of the variance.

**Procedure for calculating the optimum order of polynomial to use:**

```
variance := | Sr[high_order] ← 0
            | for  m ∈ low_order .. high_order
            |   | M[m+1, m+1] ← 0
            |   | M[1, 1] ← n
            |   | for  i ∈ 2 .. m + 1
            |   |   | M[1, i] ← 0
            |   |   | for  j ∈ 1 .. n
            |   |   |   M[1, i] ← M[1, i] + (x[j])^(i−1)
            |   | for  i ∈ 1 .. m + 1
            |   |   for  k ∈ 2 .. m + 1
            |   |     | M[k, i] ← 0
            |   |     | for  j ∈ 1 .. n
            |   |     |   M[k, i] ← M[k, i] + (x[j])^(i+k−2)
            |   | b[m+1, 1] ← 0
            |   | b[1, 1] ← 0
            |   | for  i ∈ 1 .. n
            |   |   b[1, 1] ← b[1, 1] + y[i]
            |   | for  i ∈ 2 .. m + 1
            |   |   | b[i, 1] ← 0
            |   |   | for  j ∈ 1 .. n
            |   |   |   b[i, 1] ← b[i, 1] + (x[j])^(i−1)·y[j]
            |   | F ← augment(M, b)
            |   | A ← rref (F)
            |   | A ← A^⟨m+2⟩
            |   | for  i ∈ 1 .. n
            |   |   | summ ← 0
            |   |   | for  j ∈ 1 .. m
            |   |   |   summ ← summ + A[j+1]·(x[i])^j
            |   |   | Sr[m] ← Sr[m] + [y[i] − (A[1] + summ)]^2
            | var[high_order, 1] ← 0
            | for  i ∈ low_order .. high_order
            |   var[i, 1] ← Sr[i] / (n − (i + 1))
            | var
```

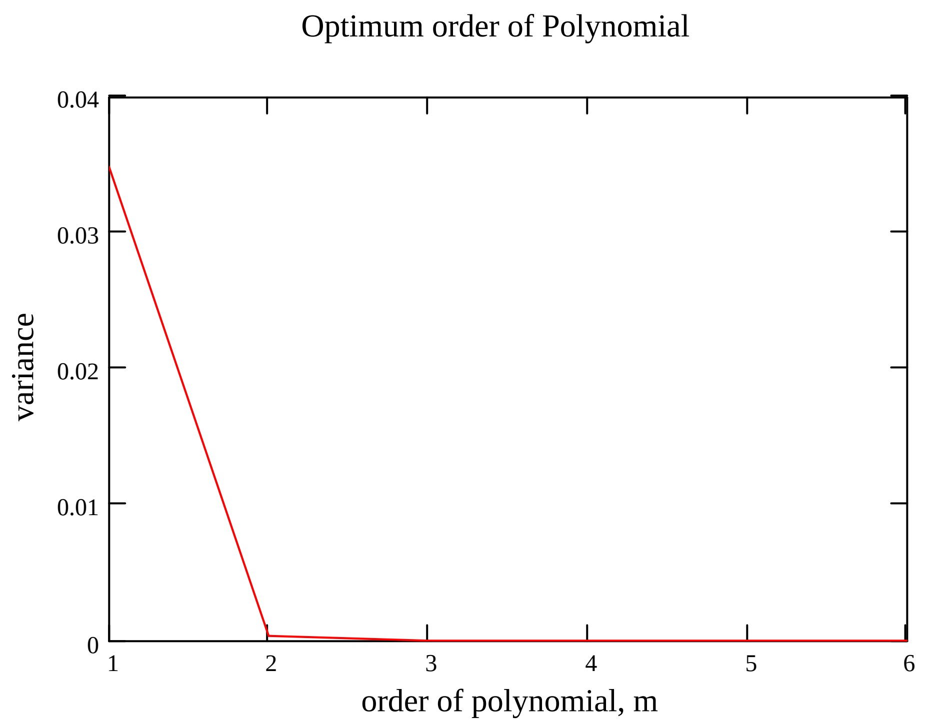
val

The calculated variance is:

$$variance = \begin{pmatrix} 3.4872222 \times 10^{-2} \\ 3.8088578 \times 10^{-4} \\ 2.7372627 \times 10^{-5} \\ 2.6107226 \times 10^{-5} \\ 3.0815851 \times 10^{-5} \\ 3.2494858 \times 10^{-5} \end{pmatrix}$$

Below, these values are illustrated:

$m := low\_order, low\_order + 1 .. high\_order$




## References


[1] Autar Kaw, *Holistic Numerical Methods Institute, http://numericalmethods.eng.usf.edu/mcd*, See **How does Nonlinear Regression work?**

### Conclusion

Mathcad helped us apply our knowledge of least squares regression method to regres
given data set to a polynomial model of order *m*.

Question 1: Water is flowing through a pipe of radius 0.5 feet and flow velocity measurements are made from the center of the wall of the pipe as follows:

$i := 1 .. 7$

| Radial location, (ft) | Velocity (ft/s) |
|---|---|
| $r_i :=$ | $v_i :=$ |
| 0 | 10 |
| 0.08 | 9.7 |
| 0.17 | 8.9 |
| 0.25 | 7.5 |
| 0.32 | 5.6 |
| 0.42 | 3.1 |
| 0.50 | 0 |

a) Regress the data to $v = a_0(1-(r^2/a^2))$, where *a* is the radius of the pipe.

b) Find the flow rate, *Q* through the pipe. (Hint: $Q = \int_0^a 2\pi r v\, dr$)

Question 2: Thermal expansion coefficient α, of steel varies with temperature as gi the table below.

$i := 1 .. 22$

| Temperature, *T* (*degrees C*) | Thermal expansion coefficient, α (E-06 *in/in F*) |
|---|---|
| $T_i :=$ | $\alpha_i :=$ |
| 80 | 6.47 |
| 60 | 6.36 |
| 40 | 6.24 |
| 20 | 6.12 |
| 0 | 6.00 |
| −20 | 5.86 |
| −40 | 5.72 |
| −60 | 5.58 |
| −80 | 5.43 |
| −100 | 5.28 |
| −120 | 5.09 |
| −140 | 4.91 |
| −160 | 4.72 |
| −180 | 4.52 |
| −200 | 4.30 |
| −220 | 4.08 |
| −240 | 3.83 |
| −260 | 3.58 |
| −280 | 3.33 |
| −300 | 3.07 |
| −320 | 2.76 |
| −340 | 2.45 |

a) Regress the data to a second order polynomial

$$\alpha = a_0 + a_1T + a_2T^2$$

b) Find the optimum order of polynomial for the regression model

c) Find the reduction in the diameter of a steel cylinder of diameter 12.5" if it is cool from room temperature of 80 degrees F to -180 degrees F.