HHC 2024

And Polynomials Begat Polyterms

Presented by

Namir *The Heretic* Shammas

Math is the chemistry of numbers

– A young chemist from the city of the Arabian Nights

HHC 2024

Dedication

To my University of Baghdad School of Engineering classmate, Saad Jasim PhD chemical engineering. Water treatment consultant and Ozone treatment expert!

A classmate who I am very proud of!

Dedication

To my chemical reactor design teacher at the University of Michigan, Professor

H. Scott Fogler (1939 – 2021).

We were made of the same metal!

HHC 2024

A Question From The Past

In previous HHC conferences we often asked, "What would be the impact on numerical analysis, ***curve fitting***, number theory, and so on, if legacy mathematicians, like Newton, Gauss, and Euler had programmable calculators?" Let me add, "How about today's laptops**?" This talk focuses on the impact on polynomial curve fitting**.

A Question From The Past (Cont.)

HHC 2024

Legacy mathematicians pushed things to the limit of *their* computing recourses—hired people who could do calculations by hand! Methods like Gaussian Quadrature and linear/multiple regression pushed the limits then! Even *non-programmable* analog slide rules did not make things a breeze!

A Question From The Past (Cont.)

HHC 2024

- And then it happened!
- The HP-35 was the first calculator that killed the slide rule! The first step forward!
- The HP-65 was the first handheld and personal programmable calculator!
- The Math and Stat pacs of the HP-65 handled many complex algorithms.

A Question From The Past (Cont.)

HHC 2024

- This was a historical breakthrough—the dawn of personal computing.
- Our communities of PPC and CHHU clubs and chapters saw the light!
- We are still here today because of programmable calculators!

HHC 2024

Classical Polynomials

- We have the general form.
- F(x) = $c_0 + \sum_{i=1}^{n} c(i) * x^i$
- Typical powers use p(i) = i
- Even powers use p(i) = 2*i
- Odd powers use p(i) = 2*i−1
- General term for the power is p(i) = a*i+b , where a is 1 or 2 and b is 0 or −1 .

Regression with Classical Polynomials

HHC 2024

- Form a Vandermonde matrix X for a polynomial of order m.
- First column is 1 for each data point.,
- Second column has the observed x values.
- Third column has the squares of the observed values of x.
- And so on until we have m+1 columns needed to calculate the intercept and m slopes.

matrix

Regression with Classical Polynomials

HHC 2024

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} 1 & x_1 & x_1^2 & \dots & x_1^m \\ 1 & x_2 & x_2^2 & \dots & x_2^m \\ 1 & x_3 & x_3^2 & \dots & x_3^m \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & x_n^2 & \dots & x_n^m \end{bmatrix} \cdot \begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \\ \vdots \\ \beta_m \end{bmatrix} + \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \\ \vdots \\ \varepsilon_n \end{bmatrix},$$

Regression with Classical Polynomials (Cont.)

HHC 2024

- Solve for the polynomial coefficients using:

$$\beta = (X^T X)^{-1} (X^T y)$$

Where X^T is the transpose of matrix X.

The matrix $X^T X$ is a square matrix that contains the summations of x data raised to various powers. The column vector $X^T y$ contains the summations of y times x raised to different powers.

HHC 2024

Meanwhile, back at the ranch!

Back to the equation for the polynomial powers p(i) = a*i + b

Variant Polynomials

- What if a and b are NOT integers? **We change the tempo of the polynomial power sequence** and create Polyterms in hope of getting better curve fitting!
- You may have seen the Shammas Polynomials I presented at HHC2008 in Corvallis.
- F(x) = $c_0 + \sum_{i=1}^{n} c(i) * x^{a*i+b}$
- Or more generally.
- F(x) = $c_0 + \sum_{i=1}^{n} c(i) * x^{a*\mathrm{p}(i)+b}$
- p(i) could be p(i) = i, p(i) = 1/i, p(i) = ln(1+i),and so on.
- My current preferred form is p(i) = i.

Variant Polynomials (cont.)

- Good to chose a < 1 and select a small value for b.
- The above choice reduces the powers of the polyterms.
- Smaller powers reduce rounding off errors when performing polynomial/polyterm regression.
- Each term in the Shammas polynomial uses the same values for a and b.
- Determining the best values for a and b requires the optimization of the best polyterm fit.

HHC 2024

Simplest Polyterms

- Simplest polyterms appear when power shift b = 0, a < 1, and p(i) = i.
- F(x) = $c_0 + \sum_{i=1}^{n} c(i) * x^{a*i}$
- And a = 0.5 (powers progress in square root multiples). The value of a can also be 1/3, or ¼.
- Since we choose a and b, the above approach does not require optimization and can use regular polynomial regression software by simply transforming x into x^0.5 or x^1/3.

Simplest Polyterms (cont.)

- Another version of the simple polyterms is to include terms with both positive and negative integer powers—The extended polyterms. Data must exclude x = 0.
- The symmetric extended polyterm has the same number of terms with positive and negative integer powers. It is defined by x and a single order.
- F(x,n) = $\sum_{i=1}^{n} d(i)/\, x^i + \; c_0 + \; \sum_{i=1}^{n} c(i) * x^i$
- The asymmetric extended polyterm has the different number of terms with positive and negative integer powers. It is defined by x and two orders.
- F(x,n,m) = $\sum_{i=1}^{m} d(i)/\, x^i + \; c_0 + \; \sum_{i=1}^{n} c(i) * x^i$

Simplest Polyterms (cont.)

- Optimizing symmetrical extended polyterms can use a single loop that examines the curve fit of polyterms with different integer orders and select the one with the highest *adjusted R-square value*.

Simplest Polyterms (cont.)

- Optimizing asymmetrical extended polyterms can use nested loops that examine the curve fit of polyterms with different positive and negative integer orders and select the one with the highest *adjusted R-square value*.

HHC 2024

Quantum Shammas Polynomials

- Not related to quantum computing!
- Inspired by the concept of probabilistic orbits of electrons in the atom.
- F(x) = $c_0 + \sum_{i=1}^{n} c(i) * x^{a(i)*i}$
- Where a(i) is a random value in the following range 0.7+(i-1) <= a(i) <= 1.3+(i-1).
- Each term has its own a(i) value.
- Requires optimization step.

HHC 2024

Double Power Polyterms

- The double power polyterms.
- Pn (x) = a0 + a1* z^q(1) + a2 *z^q(2) + a3* z^q(3) + … + an *z^ q(n)
- Where z = x^log10(p(i,x)).
- p(i,x) is a polynomial of order i and q(i) is i-1+a(i), and a(i) is a random number in the range around 1, like (0.7, 1.3).
- When p(i,x) is a linear equation, we get very good curve fits!

HHC 2024

Shifted Shammas Polyterms

- We can shift the x values.
- F(x) = c_0 + $\sum_{i=1}^{n} c(i)$ * $(\mathrm{z} - \mathrm{s(i)})^{a*i+b}$
- Where z = (x – min(x))/(max(x)-min(x))+1 to give z in [1, 2].
- Select s(i) as a random value in the range [-1, 1].
- Can also create a Shifted Quantum Shammas Polyterm.

HHC 2024

General Templates of Polyterms

- Polyterms that resemble polynomials.
- Polyterms that resemble extended polynomials. Must use x >= 0 data.
- Polyterms that resemble rational Pade Polynomials.

Optimization

- Legacy constrained optimization methods may not do well.
- Using evolutionary optimization algorithms (most use trust ranges) like Particle Swarm Optimization.. These methods use random numbers
- Using random search optimization. Simple to code and very fast. Works well because the unknow coefficients have relatively narrow ranges.
- Function to optimize (minimize) calculates 1 – adjusted R-square value. Computation effort depends on number of observations, the number of terms, and transformations.

HHC 2024

Optimization (Cont.)

- Give function F to minimize. It takes data, parameters to optimize, builds the regression matrices, and then solves for 1 – Adjusted R-square.
- Random Search Optimization: Given function F, lower and upper ranges for the variables X to optimize, and maximum number of iterations.
- Maximum number of iterations can be very high since the calculations are much simpler than those using evolutionary algorithms.

Optimization (Cont.)

- Set bestFx = a very high number, like 1E99.
- For iter = 1 to MaxIters
  - Generate random vector of variables , X, to optimize within given lower and upper ranges.
  - Calculate Fx(X)
  - If Fx(X) < bestFx then update bestFx =Fx(X) and bestX = X
- Return bestFx and bestX.

HP Prime Polyterm Programs at hpmuseum.com

HHC 2024

- Shammas Polynomial curve fitting.
- Quantum Shammas Polynomial curve fitting.
- The Symmetric Shammas Polynomial curve fitting.
- The Symmetric Quantum Shammas Polynomial curve fitting
- Above programs use random search optimization.

HP-71B Polyterm Programs at hpmuseum.com

HHC 2024

- Shammas Polynomial curve fitting.
- Quantum Shammas Polynomial curve fitting.
- Above programs use MATH ROM to perform matrix-based regression calculations.
- Above programs use random search optimization.

HHC 2024

Using ML Ensemble Approach

- Successive runs to calculate polyterms yield results with some variation.
- Can use several models to calculate sets of results.
- And then average these results.

HHC 2024

Closing Comments

- Polyterms tend to offer better curve fitting than classical polynomials when the latter does not model the fitted data/function well.
- *Better curve fitting* does not automatically mean spectacular curve fitting.

Closing Comments (Cont.)

- Polyterms for two or more independent variables also do well. The number of terms (including cross products) increases very quickly as the order of the multi-variable polyterm increases.

HHC 2024

Answering The Question!

Computing tools like programmable calculators and personal computers allow us to implement algorithms that legacy mathematicians could only dream of calculating with ease if at all! These tools enhance our math/stat skills to legacy genius levels!

HHC 2024

The complete study (with code in MATLAB) is on my website namirshammas.com. Click on the **The New World of Polyterms (beta 1)** hyperlink to download a ZIP file of about 380 Mbytes. You need to read at least the PDF file *Introducing the New World of Polyterms* in the root folder.

And that’s how they do Polyterm curve fitting in Richmond!