Snapshots of modern mathematics
from Oberwolfach

№ 7/2017

A few shades of interpolation

Justyna Szpond

The topic of this snapshot is *interpolation*. In the ordinary sense, interpolation means to insert something of a different nature into something else. In mathematics, interpolation means constructing new data points from given data points. The new points usually lie in between the already-known points. The purpose of this snapshot is to introduce a particular type of interpolation, namely, polynomial interpolation. This will be explained starting from basic ideas that go back to the ancient Babylonians and Greeks, and will arrive at subjects of current research activity.

1 Introduction

In both experimental science and practical applications, one is often faced with a situation in which a finite collection of data depending on a finite number of parameters is given and it is desirable to *interpolate* this set, that is, to insert more data points without having to do more experiments or more measurements. The aim is to find a suitably simple mathematical formula to allow us to do the interpolation on the given data set. This means that we try to find a function which for a provided list of parameter values attains the measured given values and (hopefully) predicts in a meaningful way data which would have been obtained if the given initial parameters had been perturbed a little bit.

One common example of such a situation is found in map making. If we want to create a terrain map with contour lines, as shown in Figure 1, we must measure the altitude of the terrain over sea level. This can only be done at a finite number of discrete points, usually arranged in a grid formation. In other words, when location data is collected it does not make up a continuous coverage of information, there are gaps between the known values. To create a continuous data coverage by filling in the contour lines of the map, an interpolation function is used. Figure 1 illustrates the outcome of applying a *linear* interpolation, which means that the given data is completed using a linear function. The measured points are the points at the intersection of the grid lines.

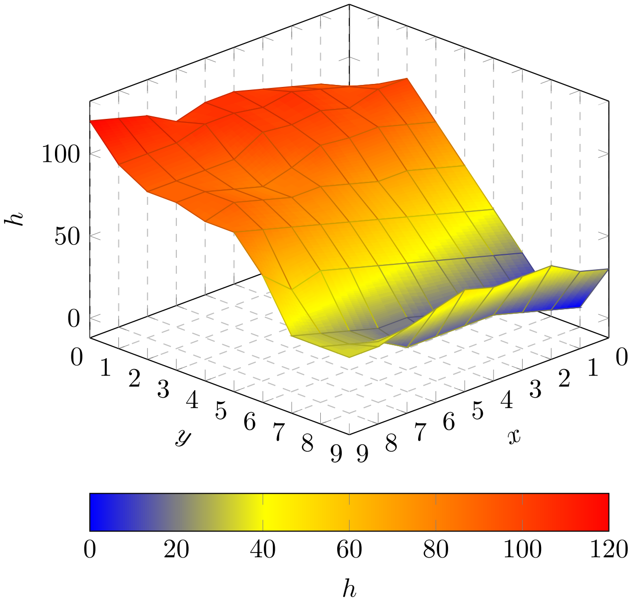

Figure 1: Terrain map

How is the interpolation function obtained? In order to explain the ideas behind this, we will first illustrate how to interpolate a data set using a function depending on one parameter only.

As an example, let us consider the average exchange rate of USD versus EUR in five consecutive years. The data is given in Table 1. If we simply plot the five

year	2011	2012	2013	2014	2015
USD/EUR	1.39243	1.28577	1.31797	1.32898	1.11040

Table 1: Exchange rates

values in a graph of rate vs. year, we obtain the five discrete points as indicated in Figure 2. If we then join the points in Figure 2 using straight-line segments,

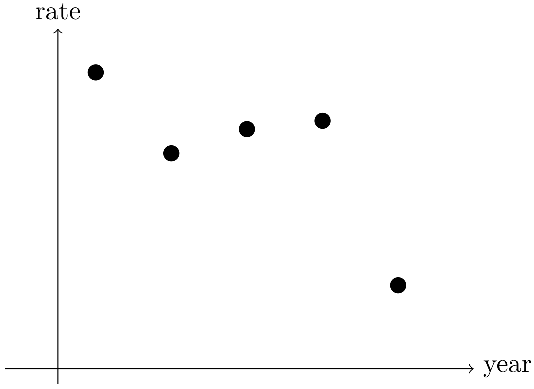

Figure 2: Data in Table 1 as points in the coordinate system

we obtain the zig-zag pictured in Figure 3. This is called *linear interpolation* and is perhaps the easiest, although somewhat rough, way to interpolate the given data.

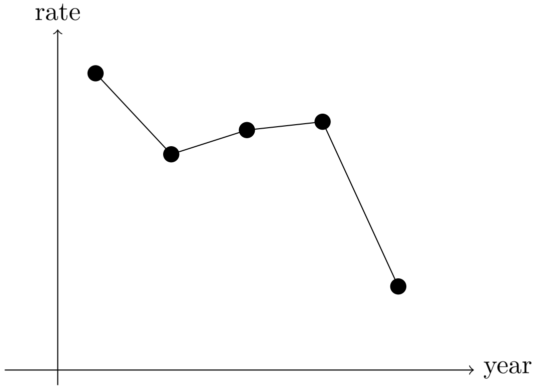

Figure 3: Linear interpolation of points in Figure 2

There is plenty of evidence for the use of linear interpolation in Mesopotamia c. 300 BC and also in ancient Greece, where around the year 150 BC Hipparchus of Rhodes used linear interpolation in order to predict positions of the moon. However, traces of the use of linear interpolation can be found as far back as

2000–1700 BC, in the Old Babylonian period, as described in a survey article for the general reader by Higgins [**?**]. Linear interpolation is very simple to use, but the problem is that for most applications it is too rough a technique. Most physical processes depend in a smooth way on changing parameters, so the actual functions describing them do not make any sudden turns as the zig-zag curve does. Moreover, linear interpolation is usually not meaningful for future predictions. For example, extending the function in Figure 3 in a natural way would lead us to make conclusions about the exchange rate in 2016 based only on the last segment, that is, only on the data from 2014 and 2015, not taking into account all the previous data.

The idea is to find a "reasonable" function $f(x)$, where by reasonable we mean that the function can be considered in some sense mathematically simple, with the property that

$$\begin{aligned} f(2011) &= 1.39243 \\ f(2012) &= 1.28577 \\ f(2013) &= 1.31797 \\ f(2014) &= 1.32898 \\ f(2015) &= 1.11040 \end{aligned} \tag{1}$$

There are infinitely many functions $f(x)$ satisfying the conditions in (1), since, in principle, we can "join the dots" in Figure 2 with any curve we like. Therefore, we impose further conditions. For instance, we might ask that the function should be continuous (as the zig-zag curve is) or perhaps that it be "differentiable" everywhere (which the zig-zag curve is not). "Differentiable" roughly means that the function is smooth, so the graph does not have any sharp corners. Polynomials are a family of functions which satisfy both of these (and more) conditions and can also be considered simple, or at least familiar. Thus the attention of mathematicians has naturally turned to the the family of polynomials.

2 The interpolation polynomial

Let us recall first of all that a polynomial $f(x)$ is defined by the sum formula:

$$f(x) = a_d x^d + a_{d-1} x^{d-1} + \ldots + a_1 x + a_0, \tag{2}$$

where each summand $a_k x^k$ is a term x^k multiplied by a real number a_k. The numbers a_k are referred to as the *coefficients* of $f(x)$. The number d appearing in the formula (2) is said to be the *degree* of $f(x)$, provided that $a_d \neq 0$.

Thus each polynomial $f(x)$ is determined by a finite set of data (namely, finitely many terms x^k and their coefficients) and its value $f(x_0)$ for any given number x_0 can be computed by finitely many arithmetic operations, that is, additions and multiplications. This is in contrast, for example, with trigonometric functions like $\sin(x)$, which require an infinite number of arithmetic operations to determine their values at given points[1]. This computational property of polynomials has been a great advantage over the ages and remains so today in the computer era, because our computers are also only capable of performing a finite number of operations on finite sets of data. We refer also to [**?**] for an even more explicit introduction to polynomials.

As noted above, polynomials are not only continuous but also differentiable functions. One can introduce their derivatives in a formal way, avoiding the fairly involved analytic machinery, as follows. The *first order derivative* of the polynomial in (2) is denoted by $f'(x)$ and is given by the formula

$$f'(x) = d \cdot a_d x^{d-1} + (d-1) \cdot a_{d-1} x^{d-2} + \ldots + 2 \cdot a_2 x + a_1. \quad (3)$$

The derivatives of higher order are defined recursively, that is, the second order derivative of $f(x)$, denoted by $f''(x)$, is the first order derivative of $f'(x)$ and so on. For example, for the cubic polynomial $f(x) := x^3 + 2x^2 + 3x + 4$ we find that

$$f'(x) = 3x^2 + 4x + 3 \quad \text{and} \quad f''(x) = 6x + 4.$$

Looking at formula (2), it is clear that a polynomial of small degree is simpler than a polynomial of large degree because it involves fewer coefficients. Thus we define the *interpolation polynomial* of a finite set of data

$$\{(x_1, y_1), (x_2, y_2), (x_3, y_3), \ldots, (x_s, y_s)\}$$

to be the polynomial $f(x)$ of smallest degree such that

$$f(x_i) = y_i \quad \text{for all} \quad i = 1, \ldots, s.$$

For example the interpolation polynomial $f(x)$ of the data in Table 1 is given by the formula

$$-0.201458 \cdot x^4 + 16.190720 \cdot x^3 - 48795.17890 \cdot x^2 + 65358734 \cdot x - 32829187100 \quad (4)$$

and depicted in Figure 4.

Now it is natural to ask the following question: How was this polynomial computed? We will give the answer in the next section.

[1] This is the Taylor series representation. For the definition of the Taylor series, see https://en.wikipedia.org/wiki/Taylor_series.

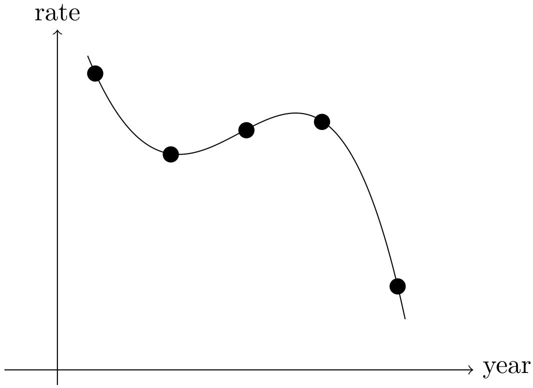

Figure 4: The interpolation polynomial of the data in Table 1

3 Interpolation polynomial formulas

As can be seen in formula (4), working with explicit data quickly leads to coefficients with many digits, which obscure the ideas leading to their computation. For this reason, it is more convenient to work with abstract data encoded in mathematical symbols.

Interpolation Problem. Let $\{(x_1, y_1), (x_2, y_2), \dots, (x_s, y_s)\}$ be a given finite set of data points. We now want to find the interpolation polynomial $f(x)$ for this data set, that is, the polynomial of least degree that satisfies the conditions

$$f(x_i) = y_i \quad \text{for} \quad i = 1, \dots, s. \tag{5}$$

If there is just one point (x_1, y_1) given, then the simplest polynomial interpolating this point is the constant polynomial $f(x) = y_1$. It is a polynomial of degree 0. If there are two points (x_1, y_1) and (x_2, y_2) given, then they determine exactly one line and this line is given by the linear polynomial

$$f(x) = a_1 x + a_0$$

and the problem is to determine the coefficients a_1 and a_0. One way is to write down the conditions in (5):

$$a_1 x_1 + a_0 = y_1 \quad \text{and} \quad a_1 x_2 + a_0 = y_2.$$

The above pair of equations determines a_0 and a_1. The solutions are given by the formulas

$$a_0 = y_1 - \frac{y_2 - y_1}{x_2 - x_1} \cdot x_1 \quad \text{and} \quad a_1 = \frac{y_2 - y_1}{x_2 - x_1}.$$

Therefore we have that $f(x)$ is given by the equation

$$f(x) = \frac{y_2 - y_1}{x_2 - x_1}(x - x_1) + y_1. \tag{6}$$

If $y_1 = y_2$, then $a_1 = 0$ and $a_0 = y_1$, so that we obtain again the constant interpolation polynomial $f(x) = y_1$, otherwise the interpolation polynomial $f(x)$ is of degree 1. Typically we would expect the values of y_1 and y_2 to differ, so that in general we expect for two given data points an interpolation polynomial of degree 1. Given s data points (or, to use technical language, *interpolation nodes*), we expect the interpolation polynomial to be of degree $s-1$. Motivated by formula (6) and following Sir Isaac Newton (1642–1727) we introduce the following notation:

$$\begin{aligned}
[y_1] &= y_1 \\
[y_1, y_2] &= \tfrac{y_2 - y_1}{x_2 - x_1} \\
[y_1, y_2, \ldots, y_{k-1}, y_k] &= \tfrac{[y_2, \ldots, y_{k-1}, y_k] - [y_1, y_2, \ldots, y_{k-1}]}{x_k - x_1}
\end{aligned} \tag{7}$$

The formulas in (7) define in a recursive way what are now known as *divided differences.* For example

$$[y_1, y_2, y_3] = \frac{[y_2, y_3] - [y_1, y_2]}{x_3 - x_1} = \frac{1}{x_3 - x_1}\left(\frac{y_3 - y_2}{x_3 - x_2} - \frac{y_2 - y_1}{x_2 - x_1}\right).$$

With this notation the *Newton interpolation polynomial* of the data in (5) is given by

$$f(x) = [y_1, y_2, \ldots, y_s](x - x_1)(x - x_2)\cdot\ldots\cdot(x - x_{s-1}) + \ldots + [y_1, y_2](x - x_1) + [y_1].$$

It is worth mentioning here that this formula did not occur to Newton out of the blue. His work was preceded by that of Sir Thomas Harriot (1560–1621) who expanded ancient interpolation formulas to polynomials of degree three and four and by Henry Briggs (1561–1630). Newton introduced divided differences in his book *Regula Differentiarum* in 1676, see [**?**]. The idea of using divided differences, though brilliant and, in principle, allowing interpolation on any finite set of data, has the hidden drawback of the complication of actually computing the divided differences. About one hundred years after Newton, another interpolation formula was proposed by Edward Waring (1736–1798) and Leonard Euler (1707–1783), and published in 1795 by Joseph-Louis Lagrange (1736–1813). The formula, which today is called the *Lagrange polynomial*, is given by

$$L(x) = y_1\ell_1(x) + \ldots + y_s\ell_s(x), \tag{8}$$

where the polynomials $\ell_k(x)$ are defined in the following way:

$$\ell_k(x) = \frac{(x - x_1) \cdot \ldots \cdot (x - x_{k-1})(x - x_{k+1}) \cdot \ldots \cdot (x - x_s)}{(x_k - x_1) \cdot \ldots \cdot (x_k - x_{k-1})(x_k - x_{k+1}) \cdot \ldots \cdot (x_k - x_s)}.$$

Note that we have

$$l_k(x_i) = \begin{cases} 0 & \text{for } i \neq k \\ 1 & \text{for } i = k \end{cases}$$

so that $L(x)$ does indeed solve the interpolation problem.

The main advantage of Lagrange's approach lies in the flexibility of the polynomial when new nodes are added. That is, we can continue to use the previously computed terms and then simply compute some further terms, instead of having to re-compute everything from the beginning. The addition of new data points is a common occurrence in experimental science, so the Lagrange polynomial can be more useful than the Newton one in this setting.

4 Hermitian interpolation

In this section we take a brief look at how the Interpolation Problem as defined in Section 3 can be generalized by imposing conditions on the derivatives of $f(x)$. In its simplest form, this kind of data can be introduced if we know additionally that the function attains a local maximum or minimum at a given node. In our initial example provided by Table 1, let us say that in the year 2015 the average exchange rate was minimal, at least in the couple of years before and after 2015. This assumption translates into the additional condition

$$f'(2015) = 0,$$

where $f'(x)$ is the first derivative of $f(x)$ as explained in (3). We have the value zero because it is a known fact that the first derivative of a differentiable function f is zero at all minima and maxima of f. This condition is not satisfied by the interpolation polynomial stated in (4), indeed the graph of the polynomial in Figure 4 goes through the node $(2015, 1.11040)$ and drops further down, whereas the local minimum condition requires the graph to go down to the node and then go back up. The correct formula in this case is given by a polynomial of degree one more than that in (4). The explicit formula is

$$0.17288x^5 - 2.61345x^4 + 14.6301x^3 - 37.3099x^2 + 42.2946x - 3.24995$$

and its graph is shown in Figure 5. In general we have the following problem to solve.

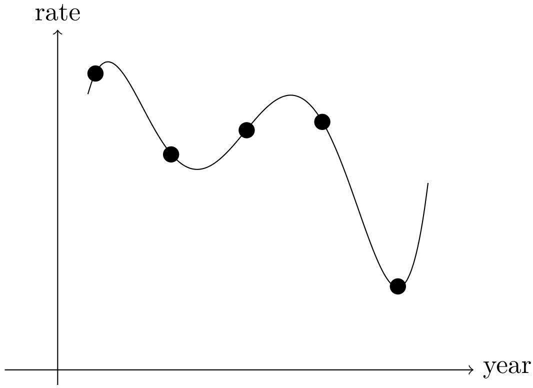

Figure 5: The Hermite interpolation polynomial

Hermitian Interpolation Problem. Given a finite set of data

$$\{(x_1; y_{1,0}, y_{1,1}, \ldots, y_{1,k_1}), (x_2; y_{2,0}, y_{2,1}, \ldots, y_{2,k_2}), \ldots, (x_s; y_{s,0}, y_{s,1}, \ldots, y_{s,k_s})\}$$

we seek the interpolation polynomial $H(x)$, that is, the polynomial of the smallest degree which satisfies that

$$H(x_i) = y_{i,0} \quad \text{for} \quad i = 1, \ldots, s, \tag{9}$$

and, for each $i = 1, \ldots, s$, the j-th derivative of $H(x)$ evaluated at the point x_i is equal to $y_{i,j}$ for each $j = 1, \ldots, k_i$.

For our example of exchange rates, with the added assumption that 2015 was a local minimum, the data (to 2 decimal places) written in the form above looks like:

$$\{(2011; 1.39), (2012; 1.20), (2013; 1.32), (2014; 1.33), (2015; 1.11, 0)\}$$

This kind of interpolation was explored by Charles Hermite (1822–1901). There is a formula, based on properly interpreted divided differences which allows one to compute the Hermite interpolation polynomial $H(x)$. Its degree is at most $k_1 + k_2 + \ldots + k_s - 1$ and if the initial data is sufficiently general, then this sum gives the actual degree.

5 Multivariate Hermitian interpolation

So far we have considered the Interpolation Problem for data depending on only one parameter (or variable). We have seen already in Figure 1 that it is useful in applications to interpolate data depending on two or more parameters. In fact most physical or industrial processes depend on multiple parameters. So it is necessary to develop what is called a *multivariate*, that is, depending on multiple variables, version of the theory. We have already seen that *univariate* interpolation goes back to ancient Babylon and Greece, but the multivariate version of the problem is much more recent. According to [**?**], it originates in the works of Carl Wilhelm Borchardt (1817–1880) and Leopold Kronecker (1823–1891), with their first publications devoted to rather special cases around 1860. Compared to the univariate interpolation, the multivariate version is still in its beginning stages. In fact it is a subject of intensive ongoing research in many branches of mathematics. Our motivation is more theoretical, and the particular problem we want to mention here belongs to algebraic geometry and commutative algebra. The Multivariate Interpolation Problem itself is a straightforward generalization of the one-dimensional version of the problem stated in the previous section, the only difference being that we seek now a polynomial $f(y_1, \dots, y_n)$ depending on n variables and we may impose conditions on the values taken by the polynomial as well as on the values of its derivatives taken with respect to various variables. For a particular special case (where all values and derivatives are equal to zero), the problem for just two variables can be stated as follows. To avoid any possible confusion, note here that (x_i, y_i) denotes a point in the plane.

Nagata Problem. Given a set of points $\{(x_1, y_1), \dots, (x_s, y_s)\} \in \mathbb{R}^2$, find the smallest degree d of a non-zero polynomial $f(x, y)$ such that the polynomial and all its derivatives up to order $m \geq 0$ are equal to zero at the given set of points.

Interestingly, Masayoshi Nagata (1927–2008) came to this problem in 1959 after he solved another problem which was stated by David Hilbert (1862–1943) at the International Congress of Mathematicians in Paris in 1900. Important problems in mathematics may take years, sometimes even centuries, to settle. Nagata made the following conjecture:

Nagata Conjecture. If $s \geq 10$ and the points in the Nagata Problem are sufficiently general, then

$$d > m\sqrt{s}.$$

It is known that the conjecture fails for $s \leq 9$ points. This conjecture can be put in a larger framework. Let Z be a finite set of sufficiently general points in the plane (here the number of points is irrelevant). For a positive integer m, we denote by $I(mZ)$ the set of all polynomials $f(x, y)$ such that f "vanishes to order at least m" at all points in Z, which means that all the derivatives of f up to order $m - 1$ are equal to zero at all points in Z. These sets are non-empty as among polynomials of very large degree, there will be certainly some satisfying this requirement. There are obvious containment relations

$$I(Z) \supset I(2Z) \supset \ldots \supset I((m-1)Z) \supset I(mZ) \supset I((m+1)Z) \supset \ldots$$

Let $\alpha_m(Z)$ denote the minimal degree of all polynomials in $I(mZ)$. The sequence of integers $\{\alpha_m(Z)\}$ is called the *initial sequence* of Z. The determination of this sequence would solve the Nagata problem and an expected answer is provided by the Nagata Conjecture. Regardless of the fact that the exact values of $\alpha_m(Z)$ are not known and hard to compute, some information on the structure of the initial sequence is available. For example the sequence is monotonically increasing, so

$$\alpha_1(Z) < \alpha_2(Z) < \ldots < \alpha_{m-1}(Z) < \alpha_m(Z) < \alpha_{m+1}(Z) < \ldots$$

The rate of growth of the initial sequence is measured by the Waldschmidt constant $\hat{\alpha}(Z)$, which is defined to be

$$\hat{\alpha}(Z) = \inf_{m \geq 1} \frac{\alpha_m(Z)}{m}.$$

Here, the symbol "inf" stands for the *infimum* or smallest value of $\alpha_m(Z)/m$ obtained as we let m range over the natural numbers. The Waldschmidt constant has been studied in the past in the area of complex analysis and has gained a lot of interest recently in algebraic geometry and commutative algebra, in particular because of the following problem, which we state generally, in an n-dimensional space (for intuition, $n = 2$ is the plane), see [**?**] for recent account of this problem.

Chudnovsky-Demailly Conjecture. For all $m \geq 1$

$$\hat{\alpha}(Z) \geq \frac{\alpha_m(Z) + n - 1}{m + n - 1}.$$

It is striking that the Nagata problem, even though its formulation is very simple, has been waiting already for more than half a century for a solution. On the other hand, problems of this kind propel a lot of research in mathematics, sometimes in unexpected areas. We recommend the very nice survey [**?**] for further reading. Recent progress on problems revolving around the Nagata Conjecture is reported in [**?**] and [**?**].

Acknowledgements

I would like to thank Thomas Bauer, Brian Harbourne and Tomasz Szemberg for valuable conversations.

Image credits

All images were created by the author.

Justyna Szpond *is an associate professor of pure mathematics at the Department of Mathematics of the Pedagogical University in Cracow.*

Mathematical subjects
Algebra and Number Theory, Geometry and Topology

Connections to other fields
Chemistry and Earth Science, Engineering and Technology

DOI
10.14760/SNAP-2017-007-EN

Snapshots of modern mathematics from Oberwolfach provide exciting insights into current mathematical research. They are written by participants in the scientific program of the Mathematisches Forschungsinstitut Oberwolfach (MFO). The snapshot project is designed to promote the understanding and appreciation of modern mathematics and mathematical research in the interested public worldwide. All snapshots are published in cooperation with the IMAGINARY platform and can be found on www.imaginary.org/snapshots and on www.mfo.de/snapshots.

Junior Editors
Sara Munday, Johannes Niediek
junior-editors@mfo.de

Senior Editor
Carla Cederbaum
senior-editor@mfo.de

Mathematisches Forschungsinstitut
Oberwolfach gGmbH
Schwarzwaldstr. 9 – 11
77709 Oberwolfach
Germany

Director
Gerhard Huisken

Mathematisches
Forschungsinstitut
Oberwolfach

Member of Leibniz
Leibniz Association

IMAGINARY
open mathematics