# UCD Senior Maths Enrichment: Polynomials

Tianyiwa Xie

October 19th 2024

## 1 Basics

What does a polynomial look like?

$$P(x) = a_n x^n + a_{n-1}x^{n-1} + ... + a_0$$

We specify that $a_n \neq 0$. The **degree** of the polynomial is $n$.

It is also important to specify where does the coefficients live. If the coefficients are in $\mathbb{Z}$, we say $P(x) \in \mathbb{Z}[x]$. Similarly, if coefficients are in $\mathbb{Q}$ or $\mathbb{C}$, $P(x) \in \mathbb{Q}[x]$ or $\mathbb{C}[x]$.

### 1.1 Identities

Let's look at some basic polynomial identities.

$$(x+y)^n = \sum_{i=0}^{n} \binom{n}{i} x^n y^{n-i}$$

$$x^n - y^n = (x-y)(x^{n-1} + x^{n-2}y + x^{n-3}y^2 + ... + xy^{n-2} + y^{n-1})$$

And when $n$ is odd, we can also substitute $y$ for $-y$ [1] in the previous expression to get

$$x^n + y^n = (x+y)(x^{n-1} - x^{n-2}y + x^{n-3}y^2 - ... - xy^{n-2} + y^{n-1})$$

The second identity above immediately gives us this theorem (try proving it!):

**Theorem 1.** *Take $P(x) \in \mathbb{Z}[x]$ and $a, b \in \mathbb{Z}$. Then $(a-b) \mid P(a) - P(b)$*

**Corollary 1.** *Suppose $p$ is a prime and $a \equiv b \pmod{p}$. Take $P(x) \in \mathbb{Z}[x]$. Then $P(a) \equiv P(b) \pmod{p}$.*

## 2 Roots

What happens to the graphs of the polynomial when the degree $n$ is odd or even?

To guess the position of the root, we can make use of special values. For instance, suppose that $P(x) \in \mathbb{Z}[x]$, $P(a) < 0$ and $P(b) > 0$, then there must be a zero between $a$ and $b$. Be careful though, since this doesn't tell us anything about the number of zeros in this interval.

Remember that if $r$ is a root of $P(x)$, then $x - r$ is a factor of $p$, and we can write $p = (x-r)p'$ for some other polynomial $p'$.

**Theorem 2** (Fundamental Theorem of Algebra)**.** *Every polynomial $P(x) \in \mathbb{C}[x]$ with degree $n$ has $n$ complex roots, counting multiplicity. That is, is can be factorised into*

$$P(x) = c(x-r_1)(x-r_2)\cdots(x-r_n)$$

*for some $c \in \mathbb{C}$.*

**Corollary 2.** *Every polynomial of degree $n$ has at most $n$ distinct roots.*

But be careful! This factorization is done in $\mathbb{C}$. If we have a polynomial in $\mathbb{Z}$, say, it doesn't always have a root in $\mathbb{Z}$, or even $\mathbb{Q}$. Consider $x^2 + 1$, for instance.

Indeed, $x^2 + 1$ is irreducible in $\mathbb{Z}[x]$.

[1] Careful! This only works when $n$ is odd

# 3 Degree

Let's talk a bit more about the degree of the polynomial. In $P(x) = a_n x^n + \cdots + a_0$, the leading term $a_n x^n$ is the most important. It completely dominates the polynomial's behaviour as $|x|$ goes to infinity. So when studying asymptotic behaviour, in complexity theory, for example, we often only care about the leading term.

## 3.1 Lagrange interpolation polynomial

Another important question. We are given $n$ data points. Can we find a polynomial that passes through all of them?

**Theorem 3.** *Suppose we are given $n+1$ points $\mathbf{x_1}, \mathbf{x_2}, \ldots, \mathbf{x_{n+1}}$, such that no two lie on a vertical line. Then there is exactly one polynomial of degree $n$ that passes through them.*

There is good reason to believe this theorem is true, and that is the idea of **degrees of freedom**.

Here, "degree of freedom" means how many free variables there are that governs the behaviour of an object. For example, for an unknown polynomial $P(x) = a_n x^n + \cdots + a_0$, the degree of freedom is $n+1$, because the behaviour of $P(x)$ is completely determined by the $n+1$ unknown coefficients.

In general, If the degree of freedom is $n$, you need $n$ data points to determine the polynomial. This concept also shows up in linear equations:

If you want to solve for one unknown $x$, you need one equation (e.g. $3x = 5$).
If you want to solve for two unknowns, you need two equations to do that. Just knowing $x + y = 1$ wouldn't tell you what are $x$ and $y$. But once you have another equation, for example, $3x + 2y = 3$, you can solve for $x$ and $y$.
In general, if you want to solve for $n$ unknowns, then you need $n$ equations to do that[a].

[a] To be more precise you need at least $n$ equations, to account for "useless" equations; it is also possible to have unsolvable equations.

*Sketch proof.* Suppose that $\mathbf{x_i} = (\alpha_i, \beta_i)$. Take a polynomial $P(x) = a_n x^n + a_{n-1} x^{n-1} + \cdots + a_1 x + a_0$, we try to find the coefficients by substituting the values in:

$$\begin{aligned} a_n \alpha_1^n + a_{n-1} \alpha_1^{n-1} + \cdots + a_1 \alpha_1 + a_0 &= \beta_1 \\ a_n \alpha_2^n + a_{n-1} \alpha_2^{n-1} + \cdots + a_1 \alpha_2 + a_0 &= \beta_2 \\ &\cdots \\ a_n \alpha_{n+1}^n + a_{n-1} \alpha_{n+1}^{n-1} + \cdots + a_1 \alpha_{n+1} + a_0 &= \beta_{n+1} \end{aligned}$$

This now becomes a linear system with $a_0, a_1, ..., a_n$ as the unknown variables. We can now solve by systematically multiplying each line by certain factors and cancelling out terms.

We have $n+1$ variables and $n+1$ equations, so in general the equation can be solve uniquely[2]. □

The polynomial that fits all the points are called **Lagrange interpolation polynomial**.

### 3.1.1 Polynomial fitting

A very basic problem in Machine Learning is polynomial fitting. Given a set of data, how to predict what comes next using this data? One way is to model this set of data using polynomials.

1. There is a set of training data, we use this to train the model; Then there's the test data, we use this to test how good is the model.
2. Fix the degree of the polynomial that you want to approximate with.
3. Get the polynomial that minimalises the root mean error [3].
4. Use test set to calculate test error, to see how good is the fitting.

[2] There are some special cases, but as long as the points doesn't lie on vertical lines the system of equations can be solved.

[3] This is just a way to measure error, using the sum of the square of the error of the model at each point.

Minimizing the data set error is not always the best idea, because as we have seen, it is always possible[4] to find a polynomial that fits all the data points perfectly.

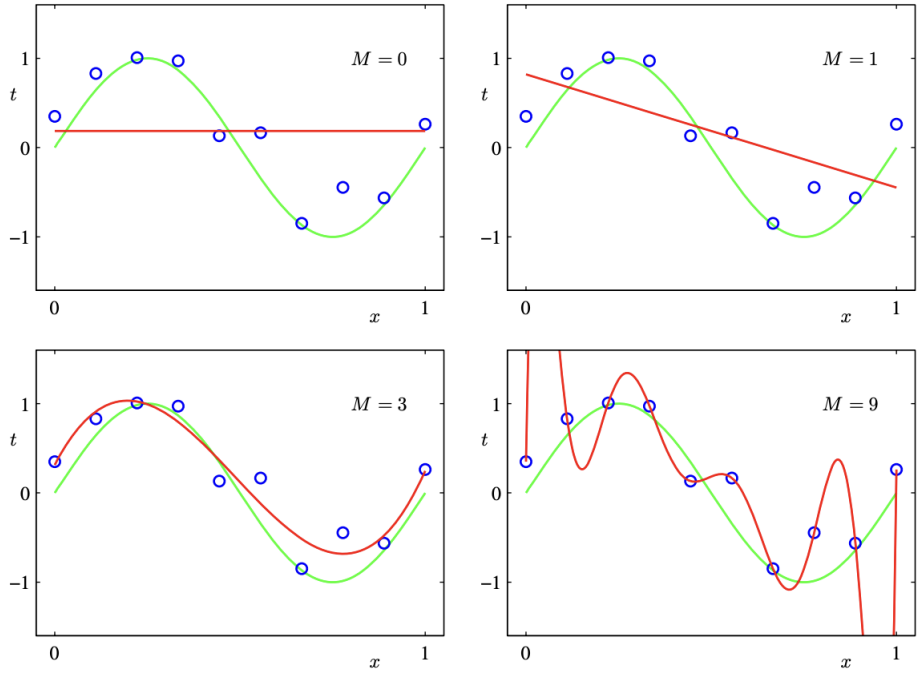
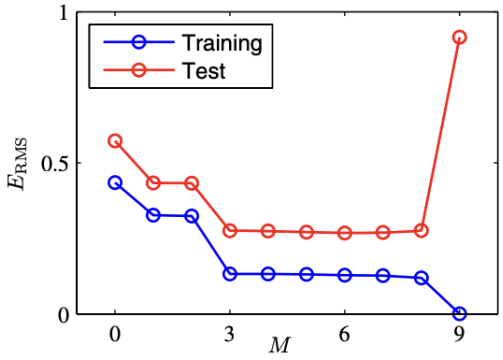



**Figure 1.4** Plots of polynomials having various orders $M$, shown as red curves, fitted to the data set shown in Figure 1.2.

To fit all the data, your polynomial has to become very squiggly, and that causes **overfitting**. It makes worse predictions, because we are learning too much from the noise of the data. As you can see in the following graph, when overfitting occurs, the training error goes to zero, but the test error becomes large[5].

**Figure 1.5** Graphs of the root-mean-square error, defined by (1.3), evaluated on the training set and on an independent test set for various values of $M$.




# 4 Irreducibility

**Definition 1.** $P(x)$ is irredicible in $\mathbb{Z}[x]$ if it cannot be written as $P(x) = P_1(x)P_2(x)$, for non-trivial polynomials $P_1$ and $P_2$. Similarly for $P(x) \in \mathbb{Q}$.

As it happens, irreducibility in $\mathbb{Z}[x]$ is already very powerful:

**Theorem 4** (Gauss)**.** *If a polynomial with integer coefficients is reducible over $\mathbb{Q}$, then it is reducible over $\mathbb{Z}$.*

**Theorem 5** (Eisenstein)**.** *Let $P(x) = a_n x^n + a_{n-1}x^{n-1} + ... + a_0$ be a polynomial with integer coefficients such that $p \mid a_i$ for $0 \leq i \leq n-1$, $p \nmid a_n$ and $p^2 \nmid a_0$. Then $P(x)$ is irreducible over $\mathbb{Z}$.*

[4] as long as there are no two points on the same vertical line

[5] Image credit: *Pattern Recognition and Machine Learning* by Christopher M. Bishop.

*Proof.* Prove this by contradiction. Suppose not. Then we can write $P(x) = g(x)h(x)$, where

$$g(x) = b_m x^m + b_{m-1}x^{m-1} + \cdots + b_1 x^1 + b_0$$
$$h(x) = c_{n-m}x^{n-m} + c_{n-m-1}x^{n-m-1} + \cdots + c_1 x^1 + c_0$$

and $m < n$. This is to ensure the factorization is not trivial, and is important later on. Also, we may assume that $m \leq (n-m)$.

Note:

$$\begin{aligned} a_0 &= b_0 c_0 \\ a_1 &= b_0 c_1 + b_1 c_0 \\ a_2 &= b_0 c_2 + b_1 c_1 + b_2 c_0 \\ &\cdots \\ a_k &= b_k c_0 + b_{k-1} c_1 + b_{k-2} c_2 + ... + b_1 c_{k-1} + b_0 c_k \end{aligned}$$

Then $P(x) = g(x)h(x)$ gives us that $p|a_0 = b_0 c_0$. Without loss of generality, suppose that $p|b_0$, then $p \nmid c_0$ because $p^2 \nmid a_0$.

Now we can do this one coefficient at a time.

$$\begin{aligned} &p|a_0 = b_0 c_0. \text{ Suppose that } p|b_0. \\ &p|a_1 = b_0 c_1 + b_1 c_0 \implies p|b_1 c_0 \implies p|b_1 \\ &p|a_2 = b_0 c_2 + b_1 c_1 + b_2 c_0 \implies p|b_2 c_0 \implies p|b_2 \\ &\cdots \\ &p|a_m = b_m c_0 + b_{m-1} c_1 + \cdots + b_0 c_m \implies p|b_m c_0 \implies p|b_m \end{aligned}$$

But then, $p|b_m c_{n-m} = a_m$, which is a contradiction, because $p \nmid a_m$. □

*Exercise* 1. Let $p$ be a prime number. Show that $P(x) = x^{p-1} + x^{p-2} + \cdots + x + 1$ is irreducible.

*Proof.* First, notice that $P(x) = \frac{x^p - 1}{x-1}$. Let's change $x$ to $y+1$. Then:

$$\begin{aligned} P(y+1) &= \frac{(y+1)^p - 1}{y} \\ &= \frac{y^p + \binom{p}{1}y^{p-1} + \cdots + \binom{p}{1}y}{y} \\ &= y^{p-1} + \binom{p}{1}y^{p-2} + \binom{p}{2}y^{p-3} + \cdots + \binom{p}{1} \end{aligned}$$

Eisenstein criterion applies, so $P(y+1)$ is irreducible, so $P(x)$ is. □

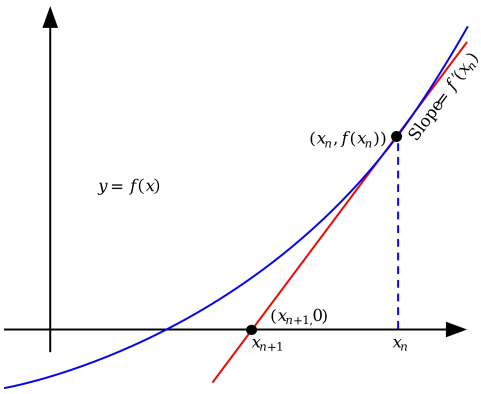
# 5 Newton-Raphson

In numerical analysis, the Newton–Raphson method, also known simply as Newton's method, is a root-finding algorithm. The most basic version starts with a real-valued function $f$, its derivative $f'$, and an initial guess $x_0$ for a root of $f$. If $f$ satisfies certain assumptions and the initial guess is close, then

$$x_1 = x_0 - \frac{f(x_0)}{f'(x_0)}$$

is a better approximation of the root than $x_0$. Geometrically, this is the intersection on the x axis by the tangent at point $x_0$, see graph [6].

[6] Image credit: wikipedia




The process is then repeated

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

until a close enough approximation is reached.

# 6 Homework

1. Prove that there is no polynomial $P(x)$ with integral coefficients with the property $P(7) = 5$ and $P(15) = 9$.
2. Prove that there is no polynomial which has the property that $P(k) = 2^k$ for all positive integers $k$.
3. $P(x) = a_n + a_{n-1}x^{n-1} + \cdots + a_0$ is a polynomial with integer coefficients and $a_n a_0 \neq 0$. Prove that if $r = \frac{p}{q}$ (in lowest terms) is a rational root of $P(x)$ then $p$ is a divisor of $a_0$ and $q$ is a divisor of $a_n$.
4. Let $P(z)$ and $Q(z)$ be polynomials with complex coefficients of degree greater than or equal to 1 with the property that $P(z) = 0$ if and only if $Q(z) = 0$ and $P(z) = 1$ if and only if $Q(z) = 1$. Prove that the polynomials are equal.
5. Find all polynomials $P$ such that $P(x)P(x+2) = P(x^2)$.
6. (IMO 1993) Let $f(x) = x^n + 5x^{n-1} + 3$, where $n > 1$ is an integer. Prove that $f(x)$ cannot be expressed as the product of two non-constant polynomials with integer coefficients.
7. Bonus: try to implement Newton-Raphson yourself using a computer program.[7]

[7] This is the basis of the Cambridge introductory computational project.