CS412: Lecture #8

Mridul Aanjaneya

February 12, 2015

Polynomial Interpolation

A commonly used approach is to use a properly crafted polynomial function

$$f(x) = \mathcal{P}_n(x) = a_0 + a_1 x + a_2 x^2 + \ldots + a_{n-1}x^{n-1} + a_n x^n$$

to interpolate the points $(x_0, y_0), \ldots, (x_k, y_k)$. Some benefits:

- Polynomials are relatively simple to evaluate. They can be evaluated very efficiently using *Horner's method*, also known as *nested evaluation* or *synthetic division*:

  $$\mathcal{P}_n(x) = a_0 + x(a_1 + x(a_2 + x(\ldots(a_{n-1} + x a_n)\ldots)))$$

  which requires only n additions and n multiplications. For example,

  $$1 - 4x + 5x^2 - 2x^3 + 3x^4 = 1 + x(-4 + x(5 + x(-2 + 3x)))$$

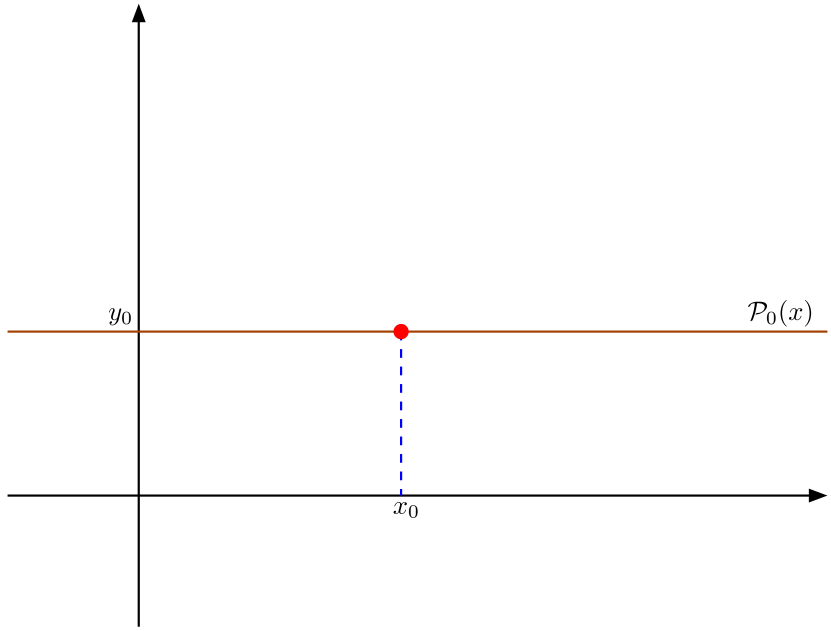
- We can easily compute derivatives $\mathcal{P}'_n, \mathcal{P}''_n$ if desired.
- Reasonably established procedure to determine the coefficients a_i.
- Polynomial approximations are *familiar* from, e.g., Taylor series.

And some disadvantages:

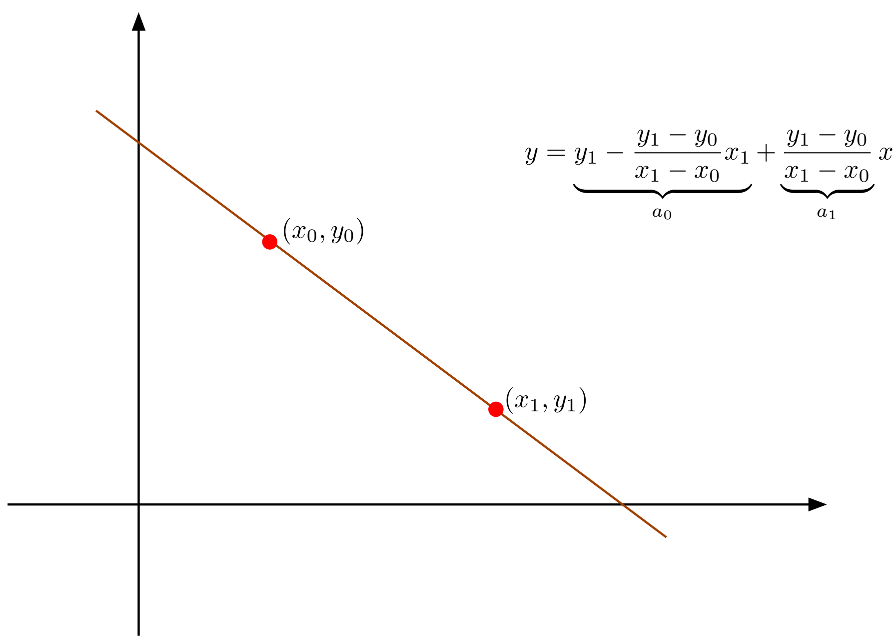
- Fitting polynomials can be problematic, when
  1. We have *many* data points (i.e., k is large), or
  2. Some of the samples are too close together (i.e., $|x_i - x_j|$ is small).

In the interest of simplicity (and for some other reasons), we try to find the most basic, yet adequate, $\mathcal{P}_n(x)$ that interpolates $(x_0, y_0), \ldots, (x_k, y_k)$. For example,

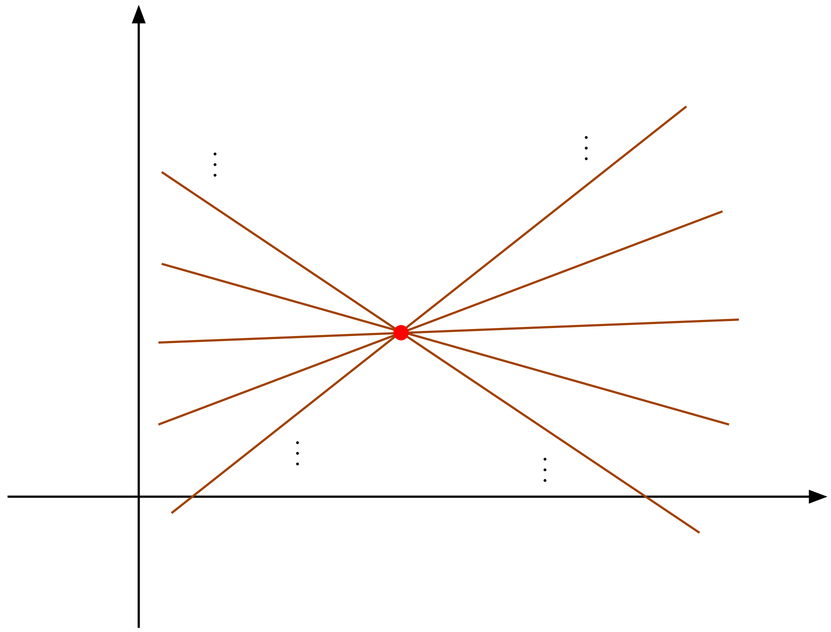
- If $k = 0$ (only one data sample), a 0-degree polynomial (i.e., a constant function) will be adequate.



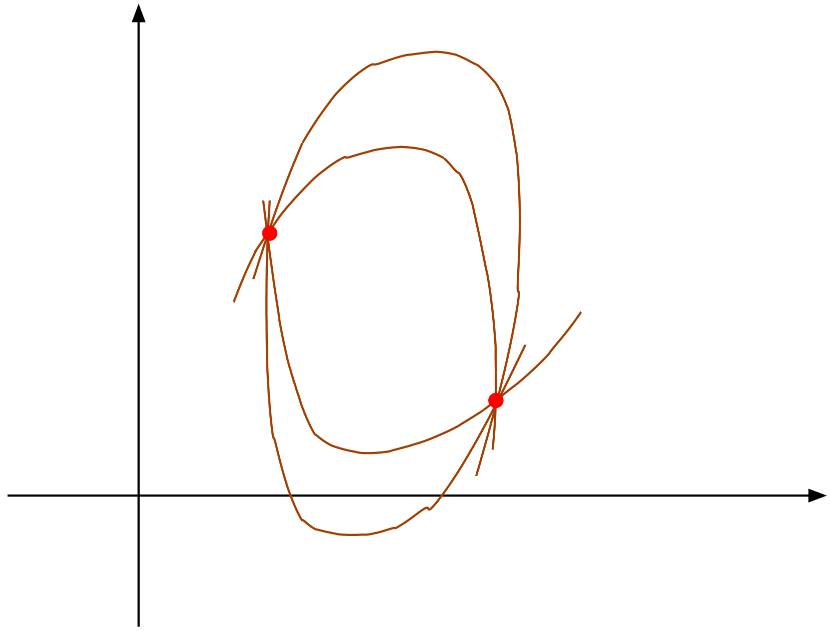
- If $k = 1$, we have two points (x_0, y_0) and (x_1, y_1). A 0-degree polynomial $\mathcal{P}_0(x) = a_0$ will not always be able to pass through both points (unless $y_0 = y_1$), but a 1-degree polynomial $\mathcal{P}_1(x) = a_0 + a_1 x$ always can.



These are not the only polynomials that accomplish the task, e.g., when $k = 0$,



or



The problem with using a degree higher than the minimum necessary is that:

- More than 1 solution becomes available, with the "right" one being unclear.
- Wildly varying curves become permissible, producing questionable approximations.

In fact, we can show that using a polynomial $\mathcal{P}_n(x)$ of degree n is the *best* choice when interpolating $n+1$ points. In this case, the following properties are assured:

- **Existence:** Such a polynomial *always* exists (assuming that all the x_i's are different! It would be impossible for a function to pass through 2 points on the same vertical line). We will show this later, by explicitly constructing such a function. For now, we can at least show that such a task would have been impossible (in general) if we were only allowed to use degree-$(n-1)$ polynomials. In fact, consider the points

  $$(x_0, y_0 = 0), (x_1, y_1 = 0), \ldots, (x_{n-1}, y_{n-1} = 0), (x_n, y_n = 1)$$

  Thus, if a degree-$(n-1)$ polynomial was able to interpolate these points, we would have:

  $$\mathcal{P}_{n-1}(x_0) = \mathcal{P}_{n-1}(x_1) = \ldots = \mathcal{P}_{n-1}(x_{n-1}) = 0$$

  $\mathcal{P}_{n-1}(x)$ can only equal zero at *exactly* $n-1$ locations *unless* it is the zero polynomial. Since $\mathcal{P}_{n-1}(x)$ is zero at n locations, we conclude that $\mathcal{P}_{n-1}(x) \equiv 0$. This is a contradiction as $\mathcal{P}_{n-1}(x_n) \neq 0$!

- **Uniqueness:** We can sketch a proof by contradiction. Assume that

  $$\begin{aligned} \mathcal{P}_n(x) &= p_0 + p_1 x + \ldots + p_n x^n \\ \mathcal{Q}_n(x) &= q_0 + q_1 x + \ldots + q_n x^n \end{aligned}$$

  both interpolate every (x_i, y_i), i.e., $\mathcal{P}_n(x_i) = \mathcal{Q}_n(x_i) = y_i$, for all $0 \leq i \leq n$. Define another n-degree polynomial

  $$\mathcal{R}_n(x) = \mathcal{P}_n(x) - \mathcal{Q}_n(x) = r_0 + r_1 x + \ldots + r_n x^n$$

  Apparently, $\mathcal{R}_n(x_i) = 0$ for all $0 \leq i \leq n$. From algebra, we know that *every* polynomial of degree n has at most n real roots, *unless* it is the zero polynomial, i.e., $r_0 = r_1 = \ldots = r_n = 0$. Since we have $\mathcal{R}_n(x) = 0$ for $n+1$ distinct values, we must have $\mathcal{R}_n(x) = 0 \Rightarrow \mathcal{P}_n(x) = \mathcal{Q}_n(x)$!

The most basic procedure to determine the coefficients $a_0, a_1, \ldots, a_n$ of the interpolating polynomial $\mathcal{P}_n(x)$ is to write a linear system of equations as follows:

$$\begin{aligned} a_0 + a_1 x_1 + a_2 x_1^2 + \ldots + a_{n-1} x_1^{n-1} + a_n x_1^n &= \mathcal{P}_n(x_1) = y_1 \\ a_0 + a_1 x_2 + a_2 x_2^2 + \ldots + a_{n-1} x_2^{n-1} + a_n x_2^n &= \mathcal{P}_n(x_2) = y_2 \\ &\vdots \\ a_0 + a_1 x_n + a_2 x_n^2 + \ldots + a_{n-1} x_n^{n-1} + a_n x_n^n &= \mathcal{P}_n(x_n) = y_n \end{aligned}$$

or, in matrix form:

$$\underbrace{\begin{bmatrix} 1 & x_1 & x_1^2 & \dots & x_1^{n-1} & x_1^n \\ 1 & x_2 & x_2^2 & \dots & x_2^{n-1} & x_2^n \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & x_n & x_n^2 & \dots & x_n^{n-1} & x_n^n \end{bmatrix}}_{V_{(n+1)\times(n+1)}} \underbrace{\begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_n \end{bmatrix}}_{a_{(n+1)}} = \underbrace{\begin{bmatrix} y_0 \\ y_1 \\ \vdots \\ y_n \end{bmatrix}}_{y_{(n+1)}}$$

The matrix V is called a *Vandermonde matrix.* The set of functions $\{1, x, x^2, \dots, x^n\}$ represent the *monomial basis.* We will see that V is non-singular, thus, we can solve the system $V\tilde{a} = \tilde{y}$ to obtain the coefficients $\tilde{a} = (a_0, a_1, \dots, a_n)$. Let's evaluate the merit and drawbacks of this approach:

- Cost to determine the polynomial $\mathcal{P}_n(x)$: *very costly.*

  Since a dense $(n+1)\times(n+1)$ linear system has to be solved. This will generally require time proportional to n^3, making large interpolation problems intractable. In addition, the Vandermonde matrix is notorious for being challenging to solve (especially with Gaussian elimination) and prone to large errors in the computed coefficients $\{a_i\}$, when n is large and/or $x_i \approx x_j$.

- Cost to evaluate $f(x)$ (x=arbitrary) if coefficients are known: *very cheap.* Using Horner's method:

  $$a_0 + a_1 x + a_2 x^2 + \dots a_n x^n = a_0 + x(a_1 + x(a_2 + x(\dots(a_{n-1} + x a_n)\dots)))$$

- Availability of derivatives: *very easy.* For example,

  $$\mathcal{P}_n'(x) = a_1 + 2a_2 x + 3a_3 x^2 + \dots + (n-1)a_{n-1}x^{n-2} + n a_n x^{n-1}$$

- Allows incremental interpolation: *no!*

  This property examines if interpolating through $(x_0, y_0), \dots, (x_n, y_n)$ is *easier* if we already know a polynomial (of degree $n-1$) that interpolates through $(x_0, y_0), \dots, (x_{n-1}, y_{n-1})$. In our case, the system $V\tilde{a} = \tilde{y}$ would have to be solved from scratch for the $n+1$ data points.

To illustrate polynomial interpolation using the monomial basis, we will determine the polynomial of degree 2 interpolating the three data points $(-2, -27)$, $(0, -1)$, $(1, 0)$. In general, there is a unique polynomial

$$\mathcal{P}_2(x) = a_0 + a_1 x + a_2 x^2$$

Writing down the Vandermonde system for this data gives

$$\begin{bmatrix} 1 & -2 & 4 \\ 1 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} -27 \\ -1 \\ 0 \end{bmatrix}$$

Solving this system by Gaussian elimination yields the solution $\tilde{a} = (-1, 5, -4)$ so that the interpolating polynomial is

$$\mathcal{P}_2(x) = -1 + 5x - 4x^2$$