Polynomial Interpolation

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

\[ f(x) = \mathcal{P}_n(x) = a_0 + a_1x + a_2x^2 + \ldots + a_{n-1}x^{n-1} + a_nx^n \]

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:

  \[ \mathcal{P}_n(x) = a_0 + x(a_1 + x(a_2 + x(\ldots(a_{n-1} + xa_n)\ldots))) \]

  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 $P_0(x) = a_0$ will not always be able to pass through both points (unless $y_0 = y_1$), but a 1-degree polynomial $P_1(x) = a_0 + a_1 x$ always can.
These are not the only polynomials that accomplish the task, e.g., when $k = 0,$
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 \( P_n(x) \) 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:
  \[
  P_{n-1}(x_0) = P_{n-1}(x_1) = \ldots = P_{n-1}(x_{n-1}) = 0
  \]
  \( P_{n-1}(x) \) can only equal zero at exactly \( n-1 \) locations unless it is the zero polynomial. Since \( P_{n-1}(x) \) is zero at \( n \) locations, we conclude that \( P_{n-1}(x) \equiv 0 \). This is a contradiction as \( P_{n-1}(x_n) \neq 0! \)

- **Uniqueness**: We can sketch a proof by contradiction. Assume that
  \[
  P_n(x) = p_0 + p_1 x + \ldots + p_n x^n
  \]
  \[
  Q_n(x) = q_0 + q_1 x + \ldots + q_n x^n
  \]
  both interpolate every \((x_i, y_i)\), i.e., \( P_n(x_i) = Q_n(x_i) = y_i \), for all \( 0 \leq i \leq n \). Define another \( n \)-degree polynomial
  \[
  R_n(x) = P_n(x) - Q_n(x) = r_0 + r_1 x + \ldots + r_n x^n
  \]
  Apparently, \( R_n(x_i) = 0 \) 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 \( R_n(x) = 0 \) for \( n + 1 \) distinct values, we must have \( R_n(x) = 0 \Rightarrow P_n(x) = Q_n(x)! \)

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

\[
\begin{align*}
  a_0 + a_1 x_1 + a_2 x_1^2 + \ldots + a_{n-1} x_1^{n-1} + a_n x_1^n &= P_n(x_1) = y_1 \\
  a_0 + a_1 x_2 + a_2 x_2^2 + \ldots + a_{n-1} x_2^{n-1} + a_n x_2^n &= P_n(x_2) = y_2 \\
  &\vdots \\
  a_0 + a_1 x_n + a_2 x_n^2 + \ldots + a_{n-1} x_n^{n-1} + a_n x_n^n &= P_n(x_n) = y_n
\end{align*}
\]
or, in matrix form:

$$
\begin{bmatrix}
1 & x_1 & x_1^2 & \cdots & x_1^{n-1} & x_1^n \\
1 & x_2 & x_2^2 & \cdots & x_2^{n-1} & x_2^n \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
1 & x_n & x_n^2 & \cdots & x_n^{n-1} & x_n^n
\end{bmatrix}
\begin{bmatrix}
a_0 \\
a_1 \\
\vdots \\
a_n
\end{bmatrix}
= 
\begin{bmatrix}
y_0 \\
y_1 \\
\vdots \\
y_n
\end{bmatrix}
$$

The matrix $V$ is called a Vandermonde matrix. The set of functions $\{1, x, x^2, \ldots, x^n\}$ 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 $\tilde{a} = (a_0, a_1, \ldots, a_n)$. Let’s evaluate the merit and drawbacks of this approach:

- Cost to determine the polynomial $P_n(x)$: 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:
  
  $$a_0 + a_1 x + a_2 x^2 + \cdots + a_n x^n = a_0 + x(a_1 + x(a_2 + x(\ldots (a_{n-1} + xa_n)\ldots)))$$

- Availability of derivatives: very easy. For example,
  
  $$P_n'(x) = a_1 + 2a_2 x + 3a_3 x^2 + \ldots + (n - 1)a_{n-1} x^{n-2} + na_n x^{n-1}$$

- Allows incremental interpolation: no!

  This property examines if interpolating through $(x_0, y_0), \ldots, (x_n, y_n)$ is easier if we already know a polynomial (of degree $n-1$) that interpolates through $(x_0, y_0), \ldots, (x_{n-1}, y_{n-1})$. 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

$$P_2(x) = a_0 + a_1 x + a_2 x^2$$

Writing down the Vandermonde system for this data gives

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

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

$$P_2(x) = -1 + 5x - 4x^2$$