How Good is Polynomial Interpolation

- Reasons we want to use polynomials for interpolation is because
  - ◇ Polynomials are easy to generate (say, by the Newton's formula),
  - ◇ Polynomials are easy to manipulate (say, for differentiation or integration),
  - ◇ Polynomials *fill up* the function space:
    - ▷ Let $f(x)$ be a piecewise continuous function over the interval $[a, b]$. Then for any $\epsilon > 0$, there exist an integer n and numbers $a_0, \ldots, a_n$ such that $\int_a^b \{f(x) - \sum_{i=0}^{n} a_i x^i\}^2 dx < \epsilon$.
    - ▷ (Weierstrass Approximation Theorem) Let $f(x)$ be a continuous function on $[a, b]$. For any $\epsilon > 0$, there exist an integer n and a polynomial $p_n(x)$ of degree n such that $\max_{x\in[a,b]} |f(x) - p_n(x)| < \epsilon$. In fact, if $[a, b] = [0, 1]$, then the Bernstein polynomial

$$B_n(x) := \sum_{k=0}^{n} \binom{n}{k} x^k (1-x)^{n-k} f(\frac{k}{n}) \qquad (1)$$

      converges to $f(x)$ as $n \to \infty$.

- However, polynomial interpolation *cannot* do all the magic. There are severe limitations:
  - ◇ Weierstrass's theoretical result, while valid, may require very high degree polynomials.
  - ◇ Polynomials can do a better job in *interpolation* than *extrapolation*. That is, a polynomial outside the range of its interpolation may not represent the function well.
  - ◇ Even within the range of interpolation, like the Runge's example, equally spaced interpolation can diverge.