Discrete Least Squares Approximation

WHEATON
COLLEGE
For Christ & His Kingdom

Carson Slater

December 9, 2022

Overview

In this review, we will cover

- Lagrange Interpolation
- Linear Least Squares Approximation
- Polynomial Least Squares Approximation

The Problem

- Consider n data points, $\{(x_0, y_0), (x_1, y_1), ..., (x_n, y_n)\}$.
- What methods exists to approximate a line to a set of numerical data points?

The Problem

Theorem (Weierstrass Approximation Theorem)

Suppose f is defined and continuous on $[a, b]$. For each $\epsilon > 0$, $\exists$ a polynomial $P(x)$ such that

$$|f(x) - P(x)| < \epsilon, \forall\, x \in [a, b].$$

The Problem

- Assuming $f(x)$ exists, we want to minimize $|f(x) - P(x)|$.
- We can use any n^{th} degree polynomial to approximate $f(x)$.
- We are only covering non-piecewise methods, so we will consider Lagrange interpolation as one method.

Interpolation (Review)

- To start, we define a **Lagrange Polynomial** as:

$$L_0(x) = \frac{x - x_1}{x_0 - x_1} \quad \text{and} \quad L_1(x) = \frac{x - x_0}{x_1 - x_0}.$$

- Note that these two polynomials would be constructed in the case of two data points, or *nodes*, $\{(x_0, y_0), (x_1, y_1)\}$.

Interpolation (Review)

- We then define a degree $n = 1$ **Lagrange Interpolation Polynomial** as

$$P_1(x) = L_0(x)f(x_0) + L_1(x)f(x_1)$$

$$= \frac{x - x_1}{x_0 - x_1} f(x_0) + \frac{x - x_0}{x_1 - x_0} f(x_1).$$

Interpolation (Review)

- More generally, we have that the **Lagrange Interpolation Basis** for $n+1$ nodes is

$$L_{n,k}(x) = \frac{(x-x_0)(x-x_1)...(x-x_{k-1})(x-x_{k+1})...(x-x_n)}{(x_k-x_0)(x_k-x_1)...(x_k-x_{k-1})(x_k-x_{k+1})...(x_k-x_n)}.$$

$$= \prod_{\substack{i=0 \\ i \neq k}}^{n} \frac{x-x_i}{x_k-x_i}.$$

Interpolation (Review)

- So the function we are looking for, $f(x)$, can be approximated with

$$P_n(x) = L_{n,0}(x)f(x_0) + L_{n,1}f(x_1) + ... + L_{n,n}f(x_n)$$

$$= \sum_{i=0}^{n} f(x_i)L_{n,i}.$$

Interpolation (Review)

- We even can see Interpolation in action. Consider Runge's function:

$$f(x) = \frac{1}{1 + 25x^2}.$$

Interpolation (Review)

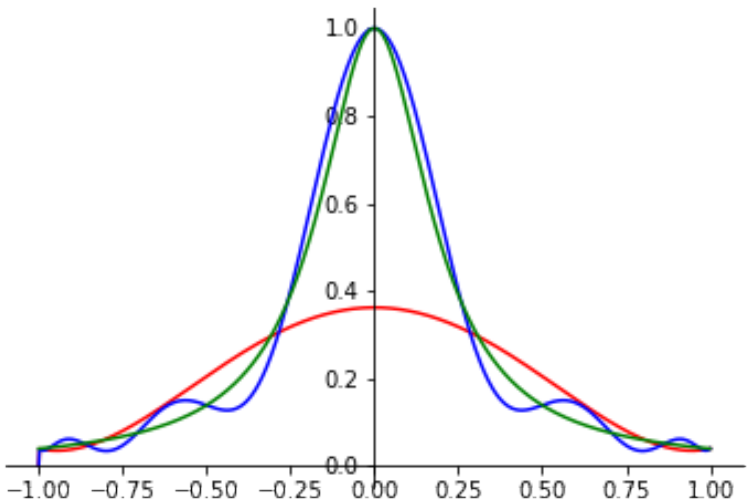


Figure: We have Runge's function on $[-1, 1]$ plotted in green, $P_5(x)$ in red and $P_{12}(x)$ in blue. Interpolation using *Clenshaw-Curtis* points, which are beyond the scope of this review.

A New Problem

- In the last example, we knew the function beforehand and were able to choose points (*nodes*).

A New Problem

- In the last example, we knew the function beforehand and were able to choose points.
- **What if we were unable to choose the points?**

A New Problem

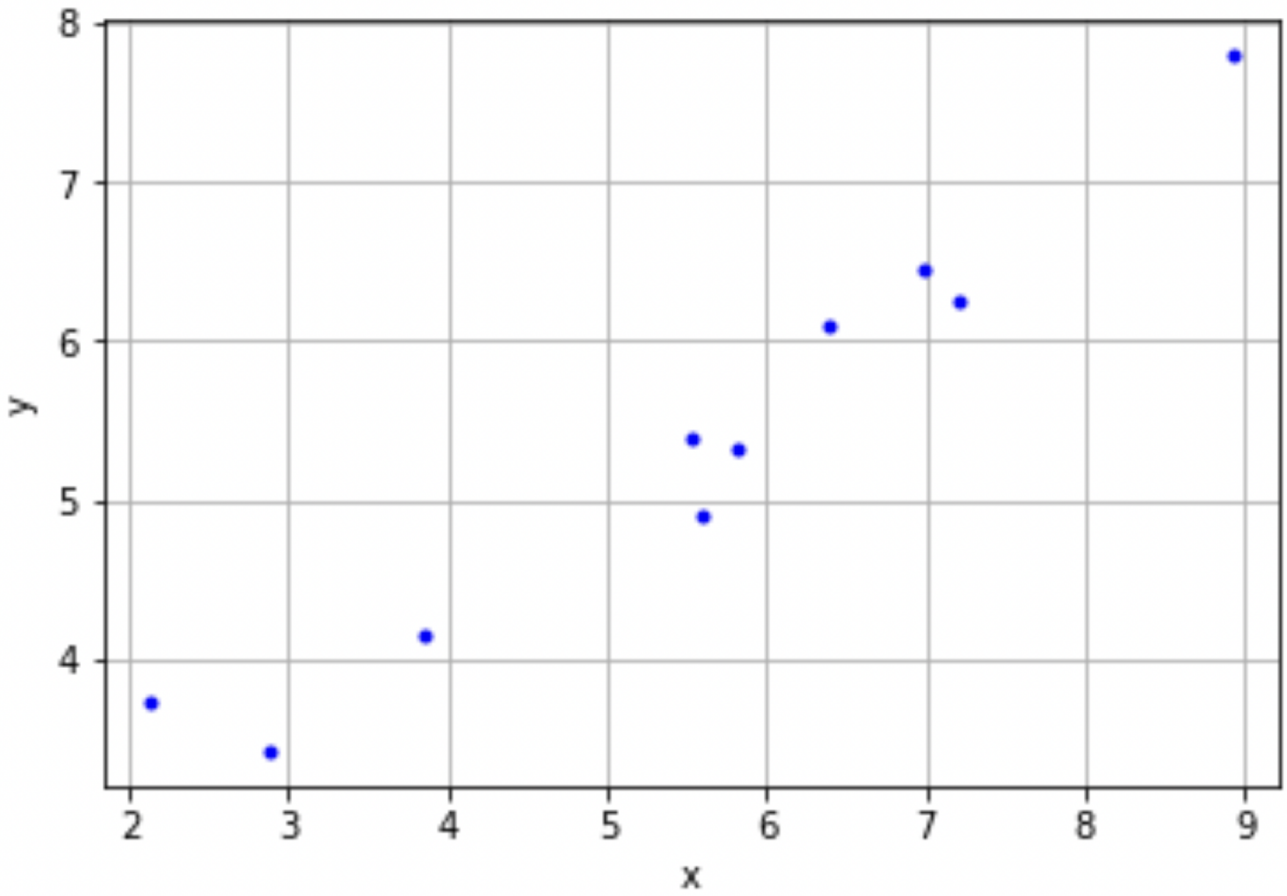


Figure: Plot of simulated data with positive correlation.

A New Problem

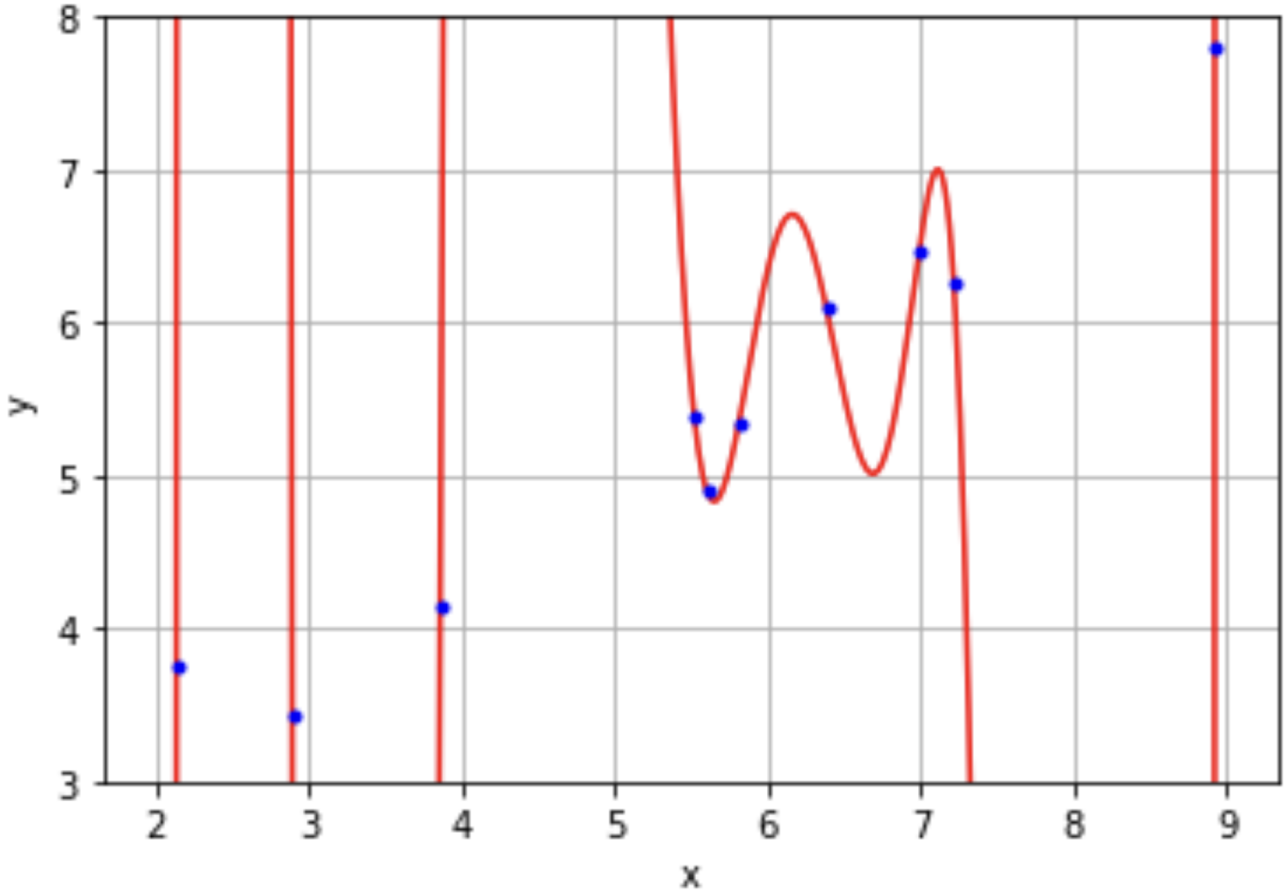


Figure: Plot of $P_9(x)$, the Lagrange interpolating polynomial for the simulated data.

A New Problem

- Lagrange Interpolation agrees with each point $\{(x_i, y_i)\}$.
- Given a high enough degree n, $P_n(x)$ will exhibit oscillatory, and hence unsatisfactory behavior.

A New Problem

- What if we could create a lower degree polynomial that approximates $f(x)$, that does not necessarily have to agree with $\{(x_i, y_i)\}$?

Discrete Least Squares Approximation

- What if we described y as:

$$y = \beta_0 + \beta_1 x?$$

Discrete Least Squares Approximation

- If we have a lower degree polynomial, we must find the (β_0, β_1) such that the error is minimized.

Discrete Least Squares Approximation

- If we have a lower degree polynomial, we must find the (β_0, β_1) such that the error is minimized.
- Error $E(\beta_0, \beta_1)$ is the distance between the observation (x_i, y_i) and $(P(x_i), y_i)$.
- Note that now $P(x)$ is the Discrete Least Squares approximation for $f(x)$ now.

Discrete Least Squares Approximation

- We can measure distance using the ℓ_1, ℓ_2, and ℓ_∞ norms.
- Which norm do we use?

Error Minimization: ℓ_∞

- Consider the **minimax** function:

$$E_\infty(\beta_0, \beta_1) = \max_{1 \leq i \leq n} \{|y_i - (\beta_1 x_i + \beta_0)|\}.$$

- This is not a good distance to minimize because it too heavily favors the maximum deviation.

Error Minimization: ℓ_1

- Consider the **absolute deviation**:

$$E_1(\beta_0, \beta_1) = \sum_{i=1}^{n} |y_i - (\beta_1 x_i + \beta_0)|.$$

- This is better, because it considers all of the deviations.

Error Minimization: ℓ_1

- We minimize the **absolute deviation** by taking the partials with respect to β_0 and β_1):

$$0 = \frac{\partial}{\partial \beta_0} \sum_{i=1}^{n} |y_i - (\beta_1 x_i + \beta_0)| \quad \text{and} \quad 0 = \frac{\partial}{\partial \beta_1} \sum_{i=1}^{n} |y_i - (\beta_1 x_i + \beta_0)|.$$

- The problem is that the **absoute deviation** is not differentiable at 0 with respect to β_1.

Error Minimization: ℓ_2

- We lastly consider the ℓ_2 norm, or **Euclidean distance**:

$$E_2(\beta_0, \beta_1) = \sum_{i=1}^{n} [y_i - (\beta_1 x_i + \beta_0)]^2.$$

- This is optimal, because it considers all of the deviations, and we can differentiate this function at 0 with respect to β_1.

Least Squares Minimization (ℓ_2)

We then minimize with differentiation:

$$0 = \frac{\partial}{\partial \beta_0} \sum_{i=1}^{n} [y_i - (\beta_1 x_i + \beta_0)]^2 \quad \text{and} \quad 0 = \frac{\partial}{\partial \beta_1} \sum_{i=1}^{n} [y_i - (\beta_1 x_i + \beta_0)]^2,$$

yielding:

$$0 = 2 \sum_{i=1}^{n} [y_i - (\beta_1 x_i + \beta_0)](-1) \quad \text{and} \quad 0 = 2 \sum_{i=1}^{n} [y_i - (\beta_1 x_i + \beta_0)(x_i)].$$

The Normal Equations

The prior result gives us the **normal equations**, written as:

$$\beta_0 \cdot n + \beta_1 \sum_{i=1}^{n} x_i = \sum_{i=1}^{n} y_i \quad \text{and} \quad \beta_0 \sum_{i=1}^{n} x_i + \beta_1 \sum_{i=1}^{n} x_i^2 = \sum_{i=1}^{n} x_i y_i.$$

The solution to these equations for β_0 and β_1 minimize the error for $y = \beta_0 + \beta_1 x$.

Reframing with Linear Algebra

We can write this result in terms of matrices, comparable to the elementary linear algebraic equation, $\mathbf{A}\vec{x} = \vec{b}$.

The result is as follows:

$$\begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ 1 & x_3 \\ \vdots & \vdots \\ 1 & x_n \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_n \end{bmatrix}.$$

Reframing with Linear Algebra

- $\mathbf{A}$ is a $n \times 2$ matrix.
- This system is not solvable when multiplying by $\mathbf{A}^{-1}$ on both sides, because $\mathbf{A}$ is not invertible.
- We can fix this problem by multiplying by $\mathbf{A}^T$ on both sides, giving us $\mathbf{A}^T\mathbf{A}\vec{x} = \mathbf{A}^T\vec{b}$.
- We will use notation $\mathbf{X}^T\mathbf{X}\vec{\beta} = \mathbf{X}^T\vec{y}$ to be consistent with the normal equations notation.

Reframing with Linear Algebra

So we go from $\mathbf{X}\vec{\beta} = \vec{y}$ to $\mathbf{X}^T\mathbf{X}\vec{\beta} = \mathbf{X}^T\vec{y}$, giving us

$$\begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ x_1 & x_2 & x_3 & \cdots & x_n \end{bmatrix} \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ 1 & x_3 \\ \vdots & \vdots \\ 1 & x_n \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ x_1 & x_2 & x_3 & \cdots & x_n \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_n \end{bmatrix}.$$

Reframing with Linear Algebra

This simplifies to

$$\begin{bmatrix} n & \sum_{i=1}^{n} x_i \\ \sum_{i=1}^{n} x_i & \sum_{i=1}^{n} x_i^2 \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^{n} y_i \\ \sum_{i=1}^{n} y_i x_i \end{bmatrix}$$

and gives us the **normal equations:**

$$\beta_0 \cdot n + \beta_1 \sum_{i=1}^{n} x_i = \sum_{i=1}^{n} y_i$$

$$\beta_0 \sum_{i=1}^{n} x_i + \beta_1 \sum_{i=1}^{n} x_i^2 = \sum_{i=1}^{n} y_i x_i.$$

A Claim Regarding the **Normal Equations**

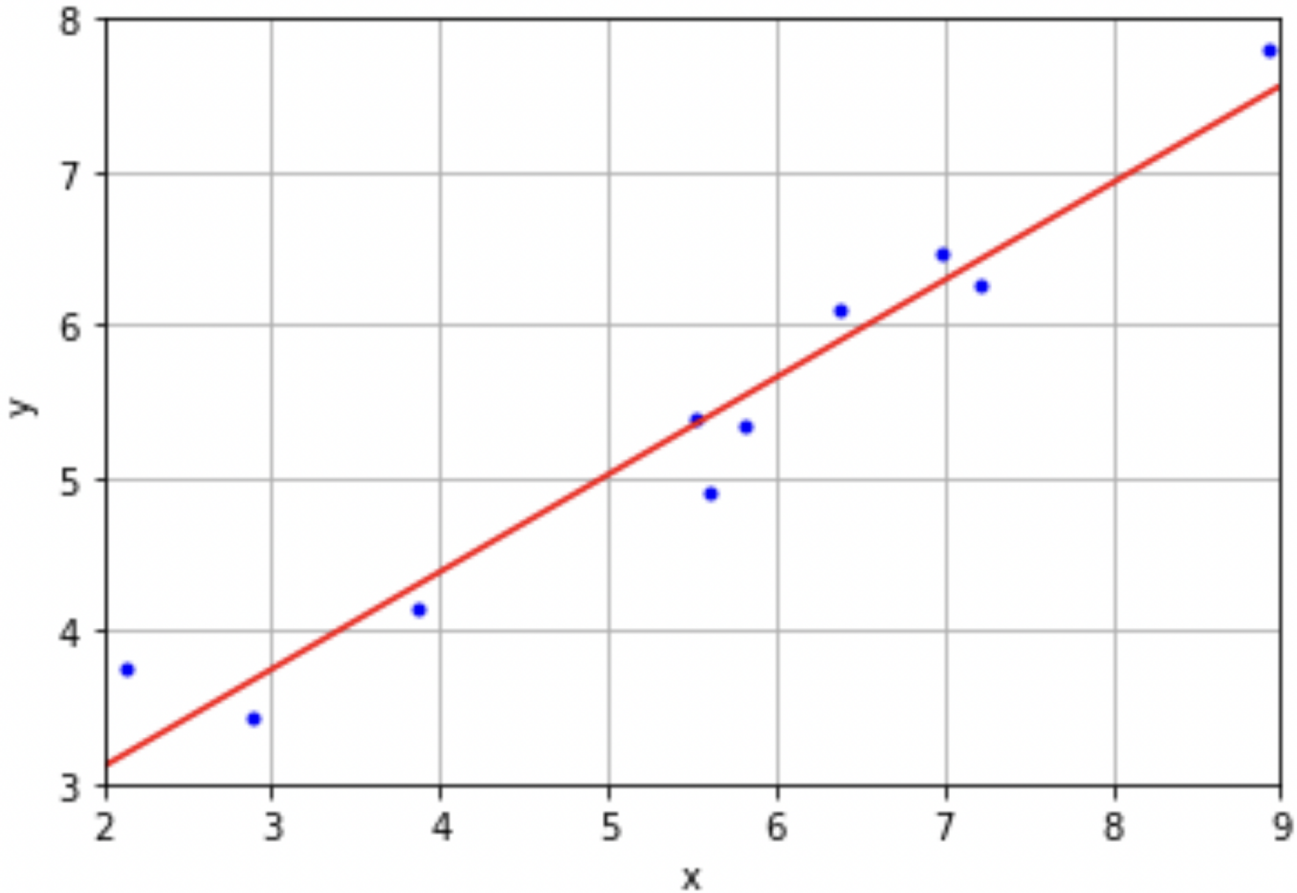


Figure: Solving the **normal equations** yields an optimal set of weights (β_0, β_1) that minimizes the squared errors, generating a polynomial as such.

Another Problem

- This is great, but what if $\{(x_i, y_i) : 1 \leq i \leq n\}$ appear to have a quadratic, cubic, or even quartic relationship?
- A 1^{st} degree polynomial will not be the best approximation for nonlinear cases.
- With Lagrange Interpolation, a quartic polynomial will only result from $n = 5$ data points.

Polynomial Least Squares Approximation

We can represent quadratic, cubic, etc., relationships using **Polynomial Least Squares** approximation, described as

$$y = \beta_0 + \beta_1 x + \beta_2 x^2 + ... + \beta_k x^k.$$

We describe the error for Polynomial Least Squares approximation as

$$E_2(\beta_0, \beta_1, \beta_2, ..., \beta_k) = \sum_{i=1}^{n} [y_i - (\beta_0 + \beta_1 x_i + \beta_2 x_i^2 + ... + \beta_k x_i^k)]^2.$$

Least Squares Minimization (ℓ_2)

We can expand E_2,

$$E_2 = \sum_{i=1}^{n} [y_i - P_n(x_i)]^2 = \sum_{i=1}^{n} y_i^2 - 2\sum_{i=1}^{n} P_n(x_i)y_i + \sum_{i=1}^{n} (P_n(x_i))^2,$$

yielding

$$E_2 = \sum_{i=1}^{n} y_i^2 - 2\sum_{j=0}^{k} \beta_j \left(\sum_{i=1}^{n} y_i x_i^k \right) + \sum_{j=0}^{k} \sum_{\ell=0}^{k} \beta_j \beta_\ell \left(\sum_{i=1}^{n} x_i^{j+\ell} \right).$$

Least Squares Minimization (ℓ_2)

We now minimize the error by taking partial derivatives with respect to β_j, giving us

$$0 = \frac{\partial E_2}{\partial \beta_j} = -2\sum_{i=1}^{n} y_i x_i^k + 2\sum_{\ell=0}^{k} \beta_\ell \sum_{i=1}^{n} x_i^{j+\ell},$$

which implies

$$\sum_{\ell=0}^{k} \beta_\ell \sum_{i=1}^{n} x_i^{j+\ell} = \sum_{i=1}^{n} y_i x_i^j.$$

Now we have $k+1$ **normal equations** for $k+1$ unknowns, β_ℓ.

Reframing with Linear Algebra

Like the degree $k = 1$ case, we can solve this with linear algebra, using $\mathbf{X}^T\mathbf{X}\vec{\beta} = \mathbf{X}^T\vec{y}$.

The result is as follows:

$$\begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ x_1 & x_2 & x_3 & \cdots & x_n \\ x_1^2 & x_2^2 & x_3^2 & \cdots & x_n^2 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_1^k & x_2^k & x_3^k & \cdots & x_n^k \end{bmatrix} \begin{bmatrix} 1 & x_1 & x_1^2 & \cdots & x_1^k \\ 1 & x_2 & x_2^2 & \cdots & x_2^k \\ 1 & x_3 & x_3^2 & \cdots & x_3^k \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & x_n^2 & \cdots & x_n^k \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \\ \vdots \\ \beta_k \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ x_1 & x_2 & x_3 & \cdots & x_n \\ x_1^2 & x_2^2 & x_3^2 & \cdots & x_n^2 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_1^k & x_2^k & x_3^k & \cdots & x_n^k \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_n \end{bmatrix}$$

Reframing with Linear Algebra

This result simplifies to

$$\begin{bmatrix} n & \sum_{i=1}^{n} x_i & \sum_{i=1}^{n} x_i^2 & \cdots & \sum_{i=1}^{n} x_i^k \\ \sum_{i=1}^{n} x_i & \sum_{i=1}^{n} x_i^2 & \sum_{i=1}^{n} x_i^3 & \cdots & \sum_{i=1}^{n} x_i^{k+1} \\ \sum_{i=1}^{n} x_i^2 & \sum_{i=1}^{n} x_i^3 & \sum_{i=1}^{n} x_i^4 & \cdots & \sum_{i=1}^{n} x_i^{k+2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \sum_{i=1}^{n} x_i^k & \sum_{i=1}^{n} x_i^{k+1} & \sum_{i=1}^{n} x_i^{k+2} & \cdots & \sum_{i=1}^{n} x_i^{2k} \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \\ \vdots \\ \beta_k \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^{n} y_i \\ \sum_{i=1}^{n} y_i x_i \\ \sum_{i=1}^{n} y_i x_i^2 \\ \vdots \\ \sum_{i=1}^{n} y_i x_i^k \end{bmatrix}.$$

Note that $\mathbf{X}^T\mathbf{X}$ is clearly symmetric.

Yet Another Problem

- How can one know that the solution is to the normal equations is unique?
- Does there exist multiple, or even infinitely many solutions?

Yet Another Problem

Theorem (Invertibility of $\mathbf{X}^T\mathbf{X}$)

Let $\mathbf{X} \in \mathbb{R}^{n \times m}$ be an $n \times m$ data matrix with columns containing $\{1, x_i, x_i^2, ..., x_i^k\}$ for all $i = 1, 2, ..., n$, with the columns being linearly independent of each other and $n > m$. Then $\mathbf{X}^T\mathbf{X}$ is symmetric positive definite, and hence is invertible.

Useful Information

Theorem (Symmetric Eigenvalue Decomposition)

We can decompose any symmetric matrix $\mathbf{X}$ *in* $\mathbf{R}^n$ *with the symmetric eigenvalue decomposition (SED):*

$$\mathbf{X} = \sum_{i=1}^{n} \lambda_i u_i u_j = \mathbf{U}^T \Lambda \mathbf{U}, \ \Lambda = \mathbf{diag}(\lambda_1, ..., \lambda_n)$$

where $\mathbf{U} := [\vec{u_1}, ..., \vec{u_n}]$ *is orthogonal (i.e.* $\mathbf{U}^T\mathbf{U} = \mathbf{I}_n$*) and contains the eigenvectors of* $\mathbf{X}$*, while the diagonal matrix* Λ *contains the eigenvalues of* $\mathbf{X}$*.*

Definition (Positive Definite)

A matrix is positive definite if is symmetric, and all its pivots are positive.

Proof that $(\mathbf{X}^T\mathbf{X})^{-1}$ Exists

Proof:

Consider $\mathbf{S} = \mathbf{X}^T\mathbf{X}$; all x_i's are unique. We know that matrix $\mathbf{S}$ matrix $\mathbf{S}$ is symmetric because

$$[s_{j\ell}] = \sum_{i=1}^{n} x_i^{j+\ell}.$$

We also know that all of the diagonal entries are strictly positive, as

$$[s_{jj}] = \sum_{i=1}^{n} x_i^{2j}.$$

So $\mathbf{S}$ is symmetric and positive definite.

Proof that $(\mathbf{X}^T\mathbf{X})^{-1}$ Exists

Also consider that there exists an eigenpair $(\lambda, \vec{x})$ such that $\mathbf{S}\vec{x} = \lambda\vec{x}$, where $\vec{x} \neq \vec{0}$.

Without loss of generality, we assume $\vec{x}^T\vec{x} = 1$. Then

$$0 < \vec{x}^T\mathbf{S}\vec{x} = \vec{x}^T(\lambda\vec{x}) = \lambda\vec{x}^T\vec{x} = \lambda,$$

implying all eigenvalues are positive.

So $\mathbf{S}$ is symmetric positive definite if and only if all its eigenvalues are positive.

Proof that $(\mathbf{X}^T\mathbf{X})^{-1}$ Exists

Now, suppose for contradiction that $\mathbf{S}$, the symmetric positive definite matrix were to be able uphold in the equality:

$$\mathbf{S}\vec{x} = 0 = 0 \cdot \vec{x}.$$

$\vec{x}$ is an eigenvector of $\mathbf{S}$ with $\lambda = 0$, as $\mathbf{S}\vec{x} = \lambda\vec{x}$.

Because $\det(\mathbf{S}) = \prod_{i=1}^{n} \lambda_i$, then this would imply that $\det(\mathbf{S}) = 0$, which is a contradiction.

$$\therefore \det(A) \neq 0 \iff \mathbf{S}^{-1} \text{ exists.}$$

The Solution to the Normal Equations is Unique

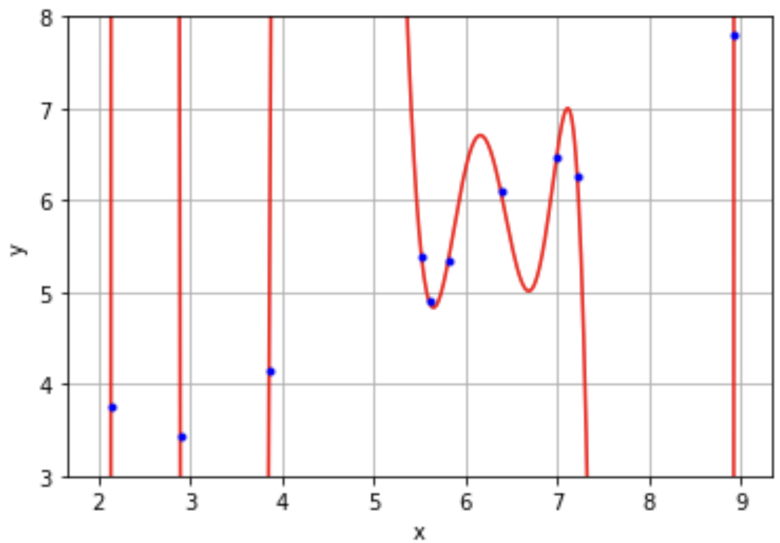
We see that if $\vec{c} = \mathbf{X}^T y$, then the unique solution to the system $\mathbf{S}\vec{\beta} := \vec{c}$ exists, and is

$$\vec{\beta} = \mathbf{S}^{-1}\vec{c} \iff \vec{\beta} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\vec{y}.$$

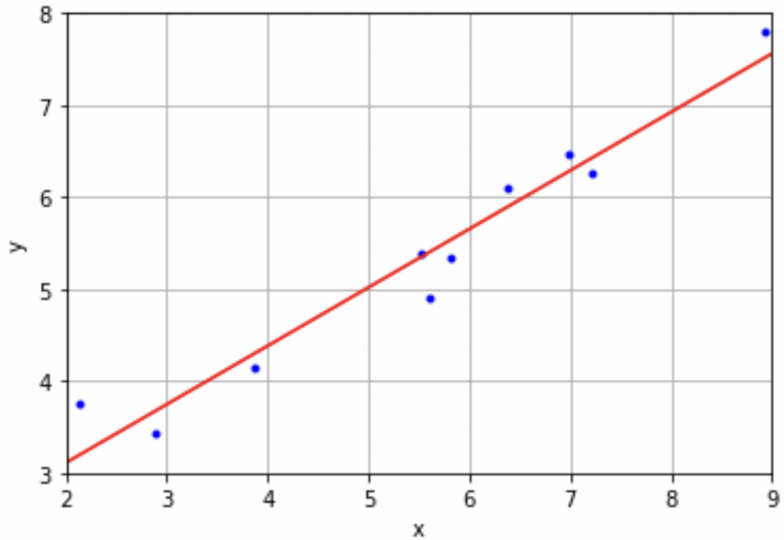
We also already know that it is the minimizer of

$$\sum_{i=1}^{n} [y_i - P_n(x_i)]^2.$$

Comparing Use Cases: Random Data



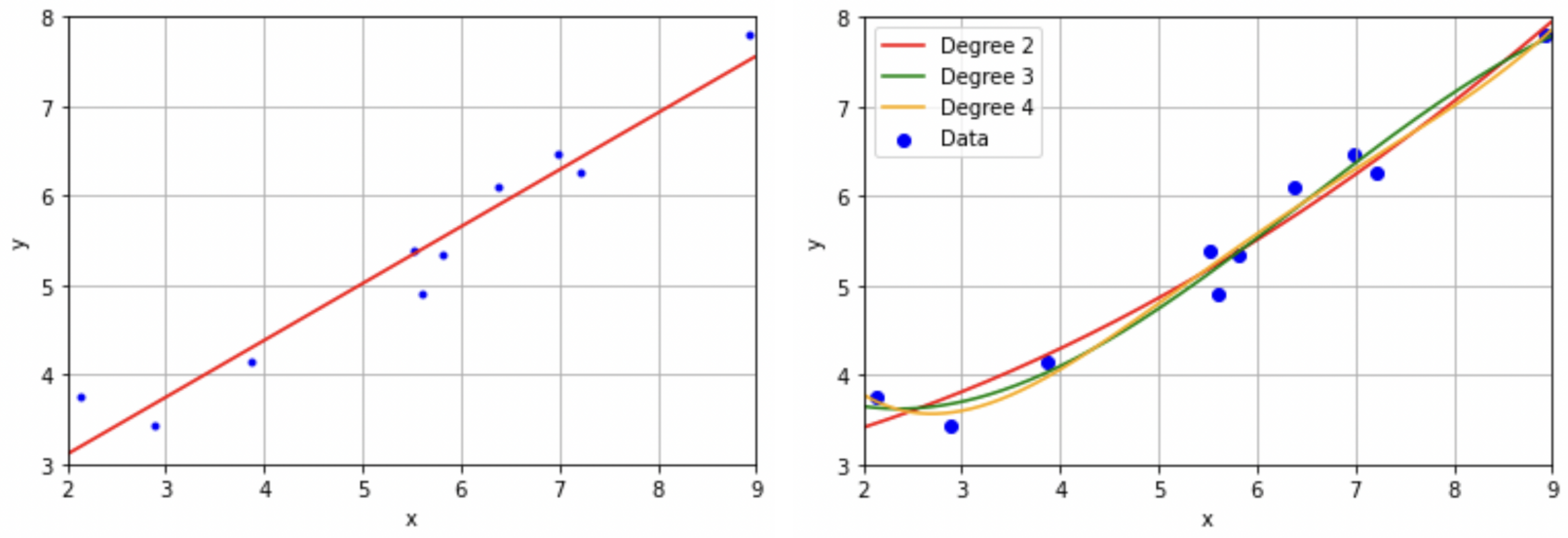
(a)



(b)

Figure: (a) Interpolation for randomly simulated data. (b) Linear Least Squares approximation for the same data.

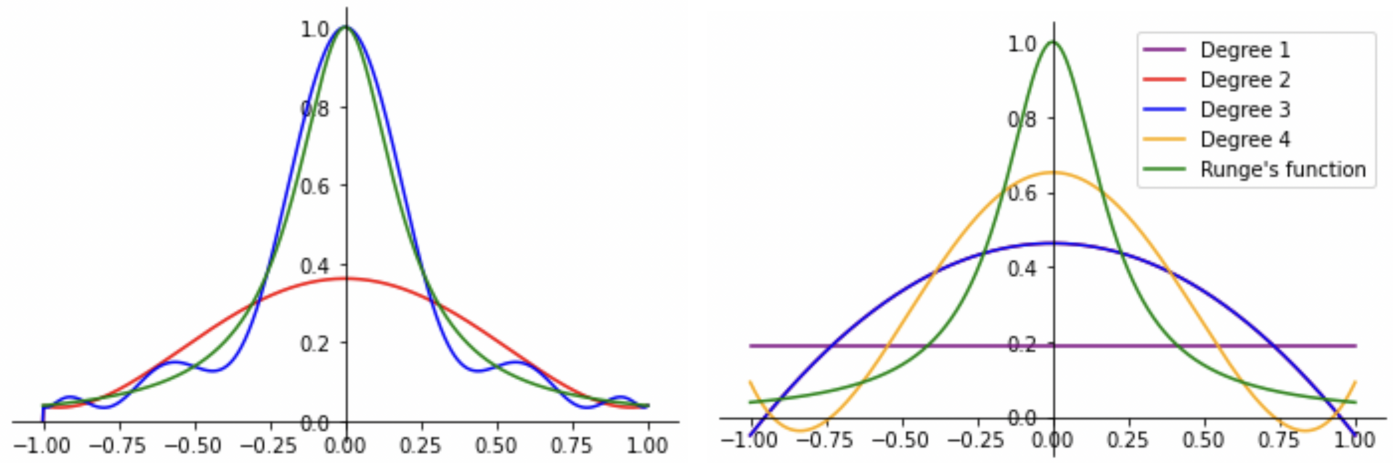
Comparing Use Cases: Random Data



(a) (b)

Figure: (a) The Linear Least Squares approximation for random data, the $n = 1$ degree case of Discrete Least Squares approximation. (b) Select Polynomial Least Squares approximations for the same random data.

Comparing Use Cases: Selected Data



(a) (b)

Figure: (a) Runge's function, as well as $P_5(x)$ and $P_{12}(x)$ found with *Clenshaw-Curtis* points. (b) Discrete Least Squares approximations using the same points. *Note that the degree* 2 *polynomial is covered by the degree* 3 *one.*

Comparing Use Cases

Interpolation might shine brightest when

- Time series observations when observations can be measured at chosen times.
- Experiments where sensors can be placed a strategic locations.
- Approximating a complicated, known function, with a simpler one on a given domain.

Comparing Use Cases

Linear Least Squares would be very optimal when:

- There exists a strong correlation between the dependent and independent variable,
- it would be useful to know how to describe the relationship between the two with a simple weight, β_1, or
- there are sufficiently high data points, such that Lagrange interpolation would produce unsatisfactory results.

Comparing Use Cases

Lastly, Polynomial Least Squares would be very useful when:

- data visualization reveals the relationship between the dependent and independent variables are quadratic, cubic, or quartic, etc.,
- data analysis reveals that there are no outliers within the data, as this has a greater effect on model fitting than Linear Least Squares, or
- in real world practices such as studying isotopes of sediments, and the rise of a disease in a population.

Questions?