Least Squares Approximation

So far, we have been trying to fit data \((x_i, y_i)\) exactly with functions

\(e.g.\) polynomials, splines

But often data is "noisy" and while it may originate from a simple function \(e.g. ax + b\), due to noise/error, no, \(e.g.\) straight line may fit the data

\[(x_i, y_i)\]
\(i = 1, \ldots, m\)

Can we do better, by trying, \(e.g.\) to find the best straight line fit?
Many possible notions of "better" examples want to pick $a_0, a_1$ to minimize

$$E_{\infty}(a_0, a_1) = \max_{1 \leq i \leq n} |y_i - (a_1 x_i + a_0)|$$

"minimax error"

$$E_1(a_0, a_1) = \sum_{i=1}^{n} |y_i - (a_1 x_i + a_0)|$$

"absolute deviation"

$$E_2(a_0, a_1) = \sum_{i=1}^{n} (y_i - (a_1 x_i + a_0))^2$$

"least squares"

Each has pros & cons

(computational complexity, robustness to outliers, etc...)

(see book)
We will focus on Least Squares.

Warmup: linear least squares (regression)

To minimize

\[ E(a_0, a_1) = \sum_{i=1}^{m} (y_i - (a_1 x_i + a_0))^2 \]

we solve

\[ \frac{\partial E}{\partial a_0} = 0, \quad \frac{\partial E}{\partial a_1} = 0 \]

\[ \Rightarrow \begin{cases} 0 = \frac{\partial}{\partial a_0} \sum_{i=1}^{m} (y_i - (a_1 x_i + a_0))^2 \\ 0 = \frac{\partial}{\partial a_1} \sum_{i=1}^{m} (y_i - (a_1 x_i + a_0))^2 \end{cases} \]

\[ \Rightarrow \begin{cases} 0 = \sum_{i=1}^{m} (y_i - (a_1 x_i + a_0)) (-1) \\ 0 = \sum_{i=1}^{m} (y_i - (a_1 x_i + a_0)) (x_i) \end{cases} \]
The normal equations for a linear regression model are given by:

\[
\begin{align*}
sa_0 + a_1 \sum_{i=1}^{m} x_i &= \sum_{i=1}^{m} y_i \\
\sum_{i=1}^{m} a_0 x_i + a_1 \sum_{i=1}^{m} x_i^2 &= \sum_{i=1}^{m} x_i y_i
\end{align*}
\]

To solve for the unknowns \(a_0\) and \(a_1\), we can express them as:

\[
a_0 = \frac{\sum_{i=1}^{m} x_i^2 \sum_{i=1}^{m} y_i - \sum_{i=1}^{m} x_i y_i \sum_{i=1}^{m} x_i}{m \left( \sum_{i=1}^{m} x_i^2 \right) - \left( \sum_{i=1}^{m} x_i \right)^2}
\]

\[
a_1 = \frac{m \sum_{i=1}^{m} x_i y_i - \sum_{i=1}^{m} x_i \sum_{i=1}^{m} y_i}{m \left( \sum_{i=1}^{m} x_i^2 \right) - \left( \sum_{i=1}^{m} x_i \right)^2}
\]
(we have all the $x_i$'s & $y_i$'s so we can find $a_0$ & $a_1$)

**Polynomial least squares**

Same idea: if $P_n(x)$ is a polynomial of degree $n$, $P_n(x) = \sum_{i=0}^{n} a_i x^i$

and we have $m$ data points $(x_i, y_i), i = 1, ..., m$ with $n < m-1$, we can minimize

$$E = \sum_{i=1}^{m} (y_i - P_n(x_i))^2$$

$$= \sum_{i=1}^{m} y_i^2 - 2 \sum_{i=1}^{m} P_n(x_i) y_i + \sum_{i=1}^{m} (P_n(x_i))^2$$

$$a_0 + a_1 x + a_2 x^2 + ... + a_n x^n$$
Taking derivatives

\[ \frac{\partial E}{\partial a_j} = 0, \quad j = 0, \ldots, n \]

gives us \( n+1 \) normal equations
to solve for \( a_0, \ldots, a_n \)
\((n+1 \text{ unknowns})\)

Normal Equations have a unique solution when the \( x_i \) are distinct.

Can use the same idea to fit different functions.

e.g. \( y = be^{ax} \) or \( y = bx^a \)

Procedure: (1) Write \( E(a, b) \)

(2) \( \frac{\partial E}{\partial a} = 0, \quad \frac{\partial E}{\partial b} = 0 \)

(3) Solve to find \( a, b \)