7750: Mathematical Foundations of Machine Learning

Instructor: Ashwin Pananjady (ashwinpm@gatech.edu)

Core problem in supervised learning

$$Y = f_0(X) + \varepsilon$$

"noise"

response $\in \mathbb{R}$

covariates / predictors / features $\in \mathbb{R}^d$.

E.g.
X : pixels of image
Y : label.

Take samples / data points $\{x_i, y_i\}_{i=1}^n$.

$y_i = f_0(x_i) + \varepsilon_i$, modeled as i.i.d draws of training data.

Goal: ① Fit a function f to training data

② Predict on new sample $\tilde{x}$ with $f(\tilde{x})$.

Modeling / representation

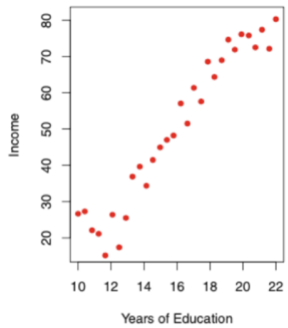
What are reasonable assumptions on f_0? Posit model class $\mathcal{F}$.

↳ Physical constraints

↳ Domain expertise

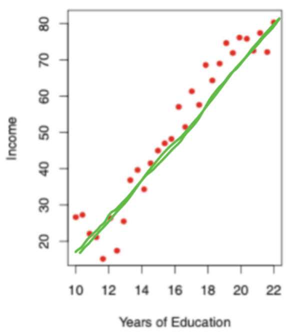
ISL.

response.

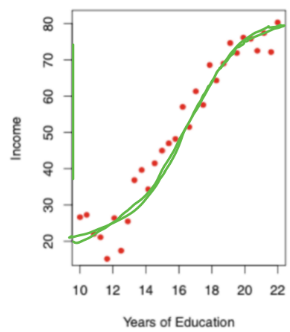


predictor

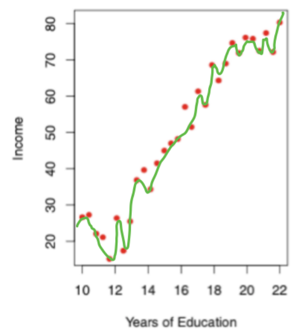
$\mathcal{F}$: linear



$\mathcal{F}$: cubic



$\mathcal{F}$: polynomial high degree.



$\mathcal{F}$.

simple → complex.

Today's focus: Properties of linear representations

$$f(x) = \sum_k \alpha_k h_k(x).$$

Example: Linear regression

$$\mathcal{F} = \left\{ f : \mathbb{R}^d \to \mathbb{R} \;\middle|\; f(x) = w^T x + b, \; w \in \mathbb{R}^d, \; b \in \mathbb{R} \right\}.$$

w_j measures "rate of change" in response when feature j is varied.

Example: Polynomial regression

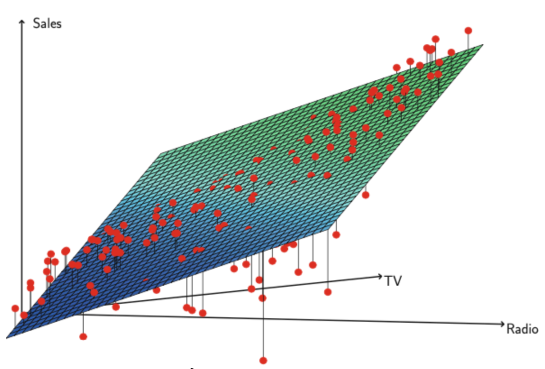
Let $P_1(x), \ldots, P_M(x)$ denote all monomials of degree $\leq \ell$, e.g., $1, x_1, \ldots, x_d, x_1^2, x_2^2, \ldots, x_1 x_2, \ldots,$

$$\mathcal{F} = \left\{ f : \mathbb{R}^d \to \mathbb{R} \;\middle|\; f(x) = \sum_{j=1}^{M} \alpha_j \cdot P_j(x). \quad \alpha_j \in \mathbb{R} \;\; \forall j = 1, \ldots, M \right\}.$$

Ex! How big is M as a function of (d, ℓ)? $\left[\binom{d+\ell-1}{\ell}, \text{ will prove in HW1} \right]$

Note: These are both linear representations, in terms of basis functions.

Useful to think geometrically whenever these are introduced. (More on this in HW).



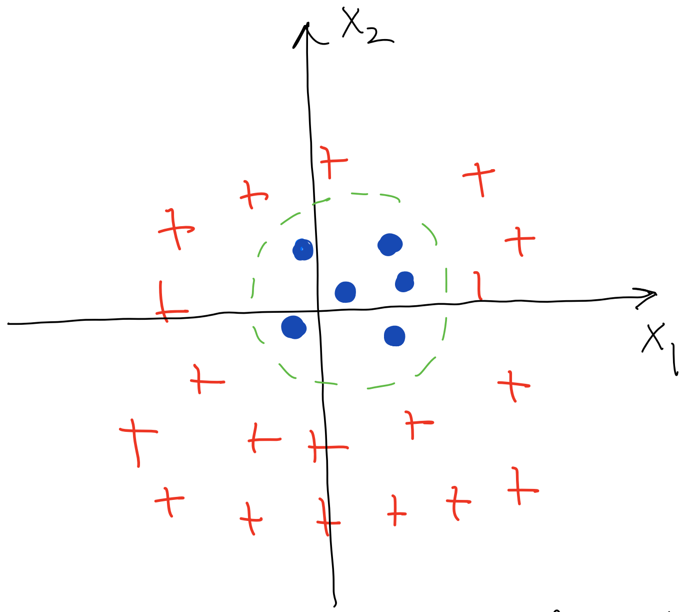
Advertising dataset, ISL.

Linear regression fits a hyperplane to data.

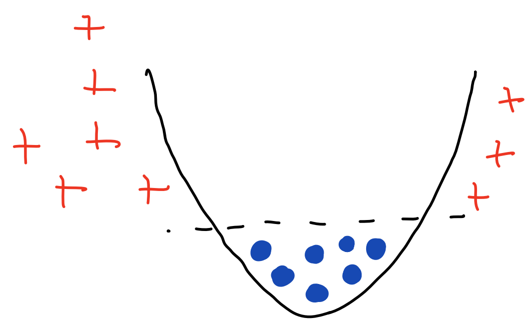
E.g. $y =$ "Sales"

$x_1 =$ TV

$x_2 =$ Radio.



$$f(x_1, x_2) = x_1^2 + x_2^2 - 5$$



More examples of polynomial approximation in 1D

Example 1: Taylor series with monomial basis fns.

Analytic functions on an open set $\mathcal{D}$ can be described by Taylor series $f(x) = \sum_{j=0}^{\infty} \alpha_j (x - x_0)^j$ for all $x \in \mathcal{D}$.

$$e^x = \sum_{j=0}^{\infty} \frac{1}{j!} x^j. \qquad \log(1+x) = \sum_{j=0}^{\infty} \frac{(-1)^{j+1}}{j} \cdot x^j.$$

$$\sin x = \sum_{j=0}^{\infty} \frac{(-1)^j}{(2j+1)!} x^{2j+1}$$

Taylor's theorem :

Let $f : \mathbb{R} \to \mathbb{R}$ be k-times differentiable at $a \in \mathbb{R}$

Then there exists $h_k : \mathbb{R} \to \mathbb{R}$ s.t.

$$\lim_{x \to a} h_k(x) = 0 \quad \text{and.}$$

$$f(x) = \sum_{j=0}^{k} \frac{f^{(j)}(a)}{j!} (x-a)^j + h_k(x)(x-a)^k.$$

Example 2 (Aside): Interpolating polynomials

Given $\{x_i, y_i\}_{i=1}^{n}$, is there a unique $(n-1)$-degree polynomial that interpolates ?

$$f(x) = \sum_{k=1}^{n} y_k \cdot \prod_{\substack{j \neq k \\ 1 \leq j \leq n}} \frac{(x - x_j)}{(x_k - x_j)}$$

Lagrange's theorem : f is unique interpolation by degree. $(n-1)$ polynomial if $\{x_i\}_{i=1}^{n}$ distinct

HW1 will walk you through proof.

Example 3: Polynomial Splines

Key idea: Interpolate with different polynomials between data points, maintaining smoothness / stability

$f(x_k) = y_k \quad , \quad k = 1, \ldots, n.$

$f(x)$ is piecewise l-th order polynomial with "kinks" at $x_1, \ldots, x_n$.

$f(x)$ has $l-1$ continuous derivatives at $\{x_i\}_{i=1}^{n}$.

See figures in Prof. Romberg's notes

Ex: We have n distinct points $(x_i, y_i)_{i=1}^{n}$. and want to interpolate with cubic spline. How many unknowns and how many constraints?

Can we think of splines through basis fn. expansions? Yes!

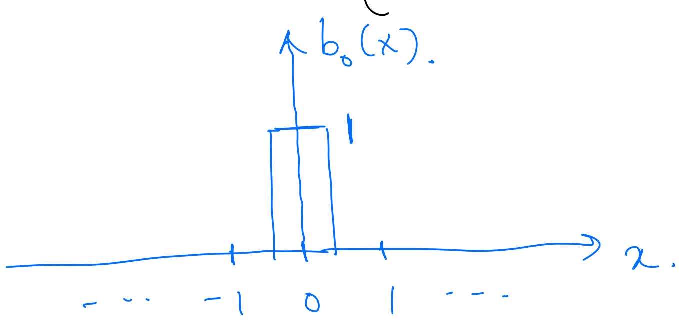
Basis functions are so-called "B-splines".

Q: Suppose we want to find basis functions for polynomial splines with $l = 0$, where x_i are integers.

A: Use the basis functions $\{b_0(x-k)\}_{k \in \mathbb{Z}}$,

where

$$b_0(x) = \begin{cases} 1, & -1/2 \le x < 1/2. \\ 0 & \text{otherwise} \end{cases}$$



Then such a spline interpolation is given by

$$f(x) = \sum_{k \in \mathbb{Z}} y_k \cdot b_0(x - x_k).$$

Two key observations:

→ The basis function is centered at your training points, this extends to different sampling patterns.

→ The training data does still influence the basis functions you choose, even if $\ell=0$.

Q: How to extend to ℓ-th order spline?

For $\ell = 1$, $b_1(x) = \begin{cases} x+1, & -1 \leq t \leq 0 \\ 1-x, & 0 \leq t \leq 1 \\ 0 & \text{otherwise} \end{cases}$

Working with this basis will allow piecewise linear interpolation between equi-space data on integers.

In general, $b_\ell(x) = \overbrace{b_0 * \cdots * b_0}^{\ell \text{ times}}(x)$

is basis for ℓ-th order splines.

<u>Example 4:</u> <u>Fourier series</u>: We will not cover this in detail, but it is foundational material to know if you haven't been exposed to it. See Prof. Romberg's notes.

Key takeaways

- Once we pick a basis (i.e. collection of basis functions), continuous-valued functions can be represented as a sequence of real values. This unlocks the linear algebra toolbox.

$$\begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} \approx \begin{bmatrix} h_1(x_1) & \cdots & h_M(x_1) \\ h_1(x_2) & \cdots & h_M(x_2) \\ & \vdots & \\ h_1(x_n) & \cdots & h_M(x_n) \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_M \end{bmatrix}$$

- Picking a basis is nontrivial, depends on application
- Assessing in what fashion we want to approximate f is also important.
- Always a tradeoff: Large $M \Rightarrow$ more complex