Review Day 1 — The curse of dimensionality.

Given data $\{x_1, \dots, x_n\} \subseteq \mathbb{R}^D$

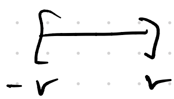
- Geometric information: Shape information for fixed distances
- Topological information: Shape information up to continuous deformations

TDA falls in this category

Geometric information is where the curse of dimensionality comes into play:
Network Analysis and Machine Learning.

1 Geometry of shapes is weird in high dimensions

Volumes of balls in $\mathbb{R}^d$

$n=1$: 

, volume $= 2r$

$n=2$: 

, volume $= \pi r^2$

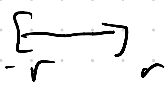
$n=3$: , Volume $= \frac{4}{3}\pi r^3$

General d: $\{x \in \mathbb{R}^d \mid \|x\| \le r\}$, $V_d = \dfrac{\pi^{\frac{d}{2}} r^d}{\left(\frac{d}{2}\right)!}$

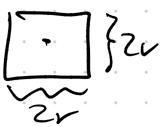
$\boxed{\text{Problem}}$ $\lim_{n\to\infty} V_n = \lim_{n\to\infty} \dfrac{\pi^{\frac{d}{2}} r^d}{\left(\frac{d}{2}\right)!} = 0.$

~ Since exponentials always grow slower than factorials.

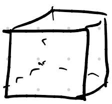
<u>Another weird issue</u> Cubes also consider distances

$n=1$ 

volume $= 2r$

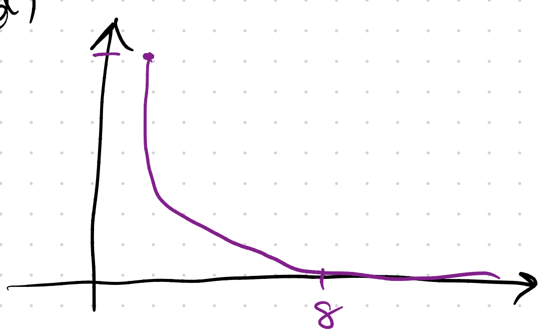
$n=2$ 

Volume $= 4r^2$

$n=3$  Volume $= 2^d r^d$

$f(d) = \dfrac{V_d\ (\text{ball})}{V_d\ (\text{cube})}$

$f(d)$



d

as d grows, a uniform sampling in cube has most points outside unit ball.

Overview of Machine learning through framework of CoD

②

Setting $\{x_1, \dots, x_n\} \subseteq \mathbb{R}^D$

In machine learning, feature matrix

$$\Omega = \begin{bmatrix} 1 & \cdots & 1 \\ \vdots & & \vdots \\ x_1 & \cdots & x_n \end{bmatrix}^T \in \mathbb{R}^{n \times (D+1)}$$

determined unique solution:

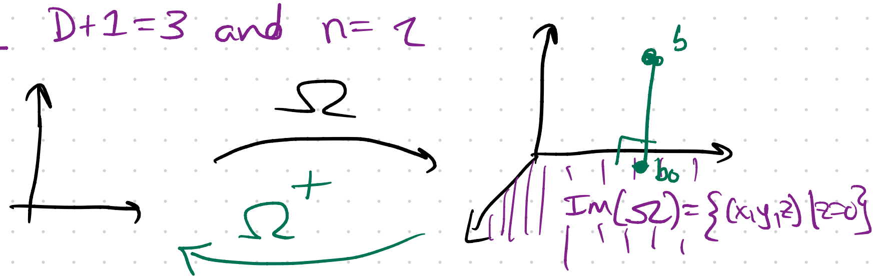
Theorems {ERM 3.8 + MLE 3.12 + Nonlinear regression 3.13}

when $\operatorname{rank}(\Omega) = D+1$, optimal θ is $\Omega^+ y$.

Recall $\operatorname{rank}(\Omega) = D+1 \Rightarrow n \leq D+1$ and full rank.

$\Rightarrow$ Prop 1.8 $\Omega^+ = (\Omega^T \Omega)^{-1} \Omega^T$ = orthogonal projection onto image with inverse on image.

if $D+1 = 3$ and $n = 2$



$r(\Omega) = n$

$\Omega^+ = \Omega^T (\Omega \Omega^T)^{-1}$

The CoD: $n << D$ reflects most situations in data science where there are many more features (D) than sample points (n).

When $n << D$, the theorems above have no unique solutions!

Add regularization parameter λ

$\boxed{\text{Theorem 3.11}}$ Unique solution is $\underbrace{(\Omega^T\Omega + n\lambda A^T A)}_{\text{Still inverting a } D+1 \times D+1 \text{ matrix, not very efficient.}}{}^{-1}\Omega^T Y$.

Still inverting a $D+1 \times D+1$ matrix, not very efficient.

Efficient computations through SVD

$\Omega \in \mathbb{R}^{n \times (D+1)}$, $\underset{\overset{\shortparallel}{r}}{\text{rank}}(\Omega) \leq n$, $n << D+1$.

$$[\ \Omega\ ] = \underset{(D+1)\times r}{[\ U\ ]}\ \underset{\boxed{r \times r}}{[\ \Sigma\ ]}\ \underset{r \times n}{[\ V^T\ ]}$$

$\sigma_i = \sqrt{\text{eigenvalues } \underbrace{\Omega\Omega^T}_{n \times n}}$

$n \times n$
small!

Efficient computations through PCA

Principal Component Analysis (PCA)

↳ Reducing dimension D+I.