

Problem: "Curse-of-dimensionality"

Geometric formulation:

$$\underline{x} = \{x_1 \cdots x_m\} \in \mathbb{R}^m$$

training sample $\{\underline{x}_i\}_1^N \in \mathbb{R}^m$

$$\{(y_i, \underline{x}_i)\}_1^N \in \mathbb{R}^{m+1}$$

$f(\underline{x})$ is a surface "above" \mathbb{R}^m
that passes near $\{y_i, \underline{x}_i\}_1^N$

If $f(\underline{x})$ is arbitrarily complex
and (more or less) completely unknown

\Rightarrow need dense sample to learn
it well

but, dense samples are hard to
get in high dimensions.

- "curse" even for the very largest
data sets

(a) sampling density $\sim N^{\frac{1}{m}}$

let $N_1 =$ dense sample in \mathbb{R}^1

then $N_m = N_1^m =$ sample size
for same density in \mathbb{R}^m

(b) interpoint distances are all
large, and about equal in \mathbb{R}^m

dist $\sim (\text{vol})^{\frac{1}{m}}$

neighborhoods that contain even
a few points have large radii.

consider $x \sim U^m(0, 1)$

neighborhood (hypercube)
containing fraction p points,
has edge length

$$e_m(p) = p^{\frac{1}{m}}$$

$$e_{10}(.01) = .63, \quad e_{10}(.1) = .80$$

$$e_{100}(.01) = .95$$

$$d(\text{closest point}) / d(1000 \text{th}) = \frac{1}{2} \quad m=10$$

(c) (nearly) all points close to edge
(boundary)

$$\underline{x} \sim U^m(0, 1)$$

expected L_∞ dist. to closest point

$$r(m, N) = \frac{1}{2} \left(\frac{1}{N}\right)^{\frac{1}{m}}$$

$$r(10, 1000) \approx 0.5$$

$$r(100, 10^6) \approx 0.45$$

= max. dist. to ~~all~~ ^{closest} edges

◦ nearly all points are closer to an edge than to another training point. (L_∞ dist. most favorable)

Prediction much harder near edges
(extrap. vs. interp.)

Most points on or near convex hull of training sample.

◦◦ every point sees itself as an outlier with (nearly) all other points to one side clumped near the origin.

Note: no problem for theory

imagine $N \rightarrow \infty \Rightarrow$ arbitrarily dense samples (asymptopia)

Only a practical problem

(a) how to get and use $N \rightarrow \infty$, or

* (b) how to deal with (relatively) small (sparse) samples

"Curse-of-dimensionality" is a diversion - does not exist.

Problem is complexity rather than dimensionality

Illustrations:

Kolmogorov's Th^m (Hilbert's 13th prob.)

any continuous

$$f(x_1 \cdots x_m) = \sum_{j=1}^{2^{m+1}} g_f \left[\sum_{i=1}^m \lambda_i \varphi_j(x_i) \right]$$

$\{\lambda_i\}_1^N$ = "universal constants"

$\{\varphi_j(z)\}_1^{2^{m+1}}$ = universal transformations
(don't depend on f)
(don't depend on f)

$g_f(u)$ = continuous, totally
characterizes $f(x_1 \cdots x_m)$

∴ ∃ a 1-dim. continuous function $g_f(u)$ that characterizes any continuous function of n -arguments

⇒ ∄ functions | dim > 1

\Rightarrow no curse-of-dimensionality

However, complexity of $f(\underline{x})$ has not been reduced.

for $f(x_1 \cdots x_n)$ fairly mild, $\underline{x} \in \mathbb{R}^n$

$\Rightarrow g_f(u)$ very wild for $u \in \mathbb{R}^1$

Hilbert: "th^m not true."

intuition was correct: "bad" functions cannot be represented in a simple way by "good" functions.

G.G. Lorentz (1986): "Kolmogorov's th^m shows only that the number of variables n is not a satisfactory characteristic of badness"

G.G. Lorentz (1986). Approximation of Functions. Chelsea, N.Y.
chapt. 11.

"Beat curse-of-dim." \Rightarrow read fine print

target function $f(\underline{x})$
 $y = E[y|\underline{x}] + (y - E[y|\underline{x}]) \leftarrow$ "noise" ε

"Curse" can be overcome:

Example: $y = f(\underline{x}) + \varepsilon$, $\text{var}(\varepsilon) = \sigma^2$

suppose $f(\underline{x}) = \sum_{j=1}^m \alpha_j x_j$ (linear fun.)

$\hat{f}(\underline{x}) = \sum_{j=1}^m \hat{\alpha}_j x_j$ (linear approx.)

$$\{\hat{\alpha}_j\}_1^m = \underset{\{\alpha_j\}_1^m}{\text{argmin}} \sum_{i=1}^N [y_i - \sum_{j=1}^m \alpha_j x_{ij}]^2$$

(linear least squares fit)

$$E[f(\underline{x}) - \hat{f}(\underline{x})]^2 = \frac{m\sigma^2}{N}$$

increases linearly with m ,
(for fixed N)

not exponentially as "curse"

would suggest.

More generally, suppose

$$f(\underline{x}) = \sum_{m=1}^M \alpha_m B_m(\underline{x})$$

\updownarrow m -dim funcs (fixed)

$$\hat{f}(\underline{x}) = \sum_{m=1}^M \hat{\alpha}_m B_m(\underline{x})$$

$$\{\hat{\alpha}_m\}_1^M = \operatorname{argmin}_{\{\alpha_m\}_1^M} \sum_{i=1}^N [y_i - \sum_{m=1}^M \alpha_m B_m(x_i)]^2$$

then

$$E[f(\underline{x}) - \hat{f}(\underline{x})]^2 = \frac{M\sigma^2}{N} \quad \begin{array}{l} \text{indep.} \\ \text{of} \\ M \end{array}$$

What happened?

$f(\underline{x})$ happens to be in a very restricted class of functions, and we chose an estimator (method) especially appropriate for that class.

either:

- (1) knew class in advance.
- (2) got lucky.

Another (final) illustration:

suppose

$$f_n(x_1 \cdots x_n) = \sum_{j=1}^n \alpha_j x_j \quad (n\text{-dim.})$$

$$f_1(x_1) = \sum_{j=1}^n \alpha_j g_j(x_1) \quad (1\text{-dim.})$$

e.g. $g_j(x_1) = x_1^{j-1}$ $n-1$ degree polynomial

we take

$$\hat{f}_n(x) = \sum_{j=1}^n \hat{\alpha}_j x_j$$

$$\hat{f}_1(x_1) = \sum_{j=1}^n \hat{\alpha}_j g_j(x_1)$$

then complexity of \hat{f}_n and \hat{f}_1 is the same

\hat{f}_n = simple function of n -vars.

\hat{f}_1 = complicated " " 1-var.

The basic reason for the "curse" is that high dimensional functions have the potential to be much more complicated than low dimensional ones, and those complications are harder to discern.

The only way to beat the "curse" (i.e. be successful) is to incorporate knowledge outside the data (assumptions) about $f(x)$, that is correct.

Choosing a method (implicitly or explicitly) does this.

Methods differ on:

- (1) the particular nature of the knowledge they assume (impose)
- (2) the strength of that imposition
- (3) their robustness to violations of these assumptions.

Need a toolkit of methods

Not just a hammer