2/2/15

① Weighted / Kernel - NN Regression & Classification

→ Motivation: Not all neighbours are the same.
We trust closer neighbours more.
Can we give them weights?
¡Si, se puede!

→ Use $k=n$ (ie. all training pts as neighbours)
→ Given a test point $\vec{x}_{test}$ or $\vec{x}$
assign a weight
$w_i(\vec{x}_{test}) \in \mathbb{R} \quad \forall\, i = 1, \ldots, n$

or w_i for simplicity.
[But remember that w_i changes for different $\vec{x}_{test}$]

→ Prediction Rule
→ Regression $g(\vec{x}) = \dfrac{\sum_{i=1}^{n} w_i y_i}{\sum_{i=1}^{n} w_i}$ } weighted avg.

→ Classification
$g(\vec{x})$ = weighted majority vote

$$= \underset{c}{\operatorname{argmax}} \sum_{i=1}^{n} w_i \cdot \underbrace{I(y_i = c)}_{\text{Indicator function}}$$

What's a good weight?

→ Something that decays with $d(\vec{x}, \vec{x}_i)$

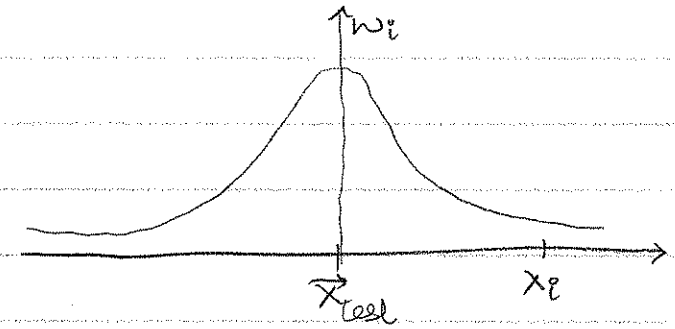
→ Many possibilities $1/d$, $1/d^2$, $I[d \leq a]$, etc

→ One mathematically convenient choice

$$w_i = e^{\frac{-d^2(\vec{x}, \vec{x}_i)}{\sigma^2}}$$

σ^2 ← Bandwidth or Kernel width

Gaussian Kernel



② Effect of Bandwidth

(2.1) $\sigma \to \infty \Rightarrow w_i \to 1$

or more specifically, $w_i(\vec{x}) \to 1 \quad \forall i \quad \forall \vec{x}$

$\Rightarrow$ Equal wt at all training pts $\forall$ input $\vec{x}_{test}$

$\Rightarrow$ All predictions the same

$\hat{y} = \frac{1}{n} \Sigma y_i$ } dataset average.

(2.2) $\sigma \to 0$?

$$w_i = e^{\frac{-d^2(\vec{x}, \vec{x}_i)}{\sigma^2}} \to 0$$

So $\hat{y} = 0/0$?

A little more careful analysis. Say Regression

$$\hat{y} = \frac{\sum_{i=1}^{n} e^{\frac{-d^2(\vec{x}, \vec{x}_i)}{\sigma^2}} \cdot y_i}{\sum_{i=1}^{n} e^{\frac{-d^2(\vec{x}, \vec{x}_i)}{\sigma^2}} \cdot 1} \cdot \frac{e^{\frac{+d(\vec{x}, \vec{x}_j)}{\sigma^2}}}{e^{\frac{+d(\vec{x}, \vec{x}_j)}{\sigma^2}}}$$

where j = index of 1-NN of $\vec{x}$

$$= \underset{i=1,\ldots,n}{\operatorname{argmin}} \; d(\vec{x}, \vec{x}_i)$$

[Assume unique argmin for simplicity]

$$\Rightarrow \hat{y} = \frac{\sum_{i \neq j} e^{\frac{-[d^2(\vec{x}, \vec{x}_i) - d^2(\vec{x}, \vec{x}_j)]}{\sigma^2}} \cdot y_i + e^0 \cdot y_j}{\sum_{i \neq j} e^{\frac{-[d^2(\vec{x}, \vec{x}_i) - d^2(\vec{x}, \vec{x}_j)]}{\sigma^2}} \cdot 1 + e^0 \cdot 1}$$

$$e^{\frac{-(+ve)}{\sigma^2}} \to 0$$

as $\sigma \to 0$

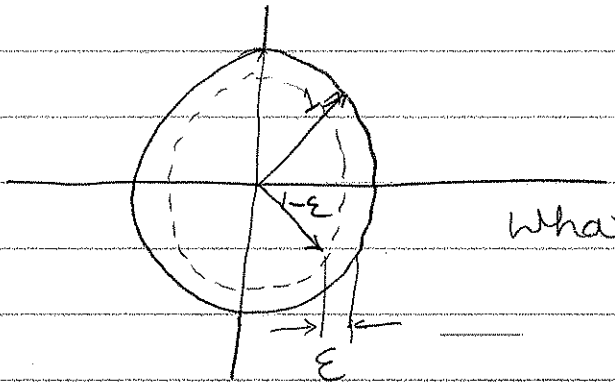
$\Rightarrow \hat{y} = y_j \equiv$ 1-NN prediction

(3) Curse of Dimensionality

Learning in high-dimensional space ≡ d-large is difficult.

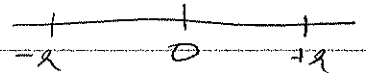
In particular, NN "shouldn't work". Why?
∵ distances/neighbours become meaningless.

→ Example #1: Consider sphere in $\mathbb{R}^d$ centred at $\vec{0}$
radius $r = 1$

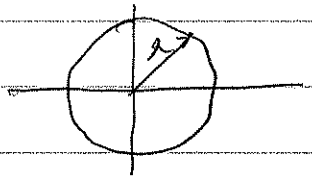


What is volume of outer ε-shell?

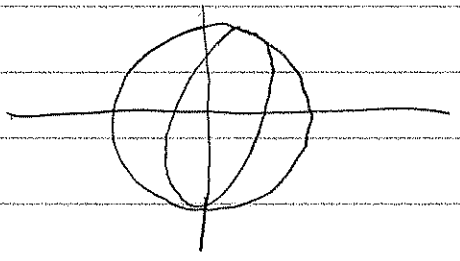
Well, what is volume of sphere?

1D 

$2r$

2D 

πr^2

3D 
$\frac{4}{3}\pi r^3$

$\equiv k_d r^d$

Now, $\frac{\text{Volume (Shell)}}{\text{Volume (Sphere)}} = \frac{k_d (1)^d - k_d (1-\varepsilon)^d}{k_d \, 1^d}$

$= 1 - (1-\varepsilon)^d$

$\to 1 \quad \text{as} \quad d \to \infty$

$\Rightarrow$ Nearly all volume lies in outer ε-shell
Assume uniform density of data [Hint: Problem!]

$\Rightarrow$ Nearly all mass lies in shell

$\Rightarrow$ Nearly all data-points lie in shell

$\Rightarrow$ All neighbours are equally apart!

Example 2: $\vec{x} = (x_1, \ldots, x_d)$

Assume $x_1, \ldots x_d$ are I.I.D random vars
[Hint: Problem]

Consider Normalized distance^2 to origin:

$$D = \frac{1}{d} \|\vec{x} - \vec{0}\|_2^2 = \frac{1}{d} \sum_{i=1}^{d} x_i^2$$

Recall, Central Limit Theorem

If $z_1, \ldots, z_n$ are I.I.D RVs with

$$E[z_i] = \mu \quad \forall i$$
$$Var(z_i) = \sigma^2 \quad \forall i$$

then $\frac{1}{n}\sum z_i \rightarrow N\left(\mu, \frac{\sigma^2}{n}\right)$

as $n \rightarrow \infty$

So in our case $z_i = x_i^2$

$$D = \frac{1}{d}\sum x_i^2 \rightarrow N\left(E[x_i^2], \frac{Var(x_i^2)}{d}\right)$$

$Var(x_i^2) \equiv$ constant

$$\Rightarrow \frac{Var(x_i^2)}{d} \rightarrow 0 \text{ as } d \rightarrow \infty$$

$\Rightarrow$ D is nearly a constant.

Problem!