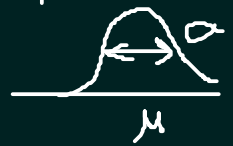
Parametric Learning : MAP → MLE, Prior → parameters

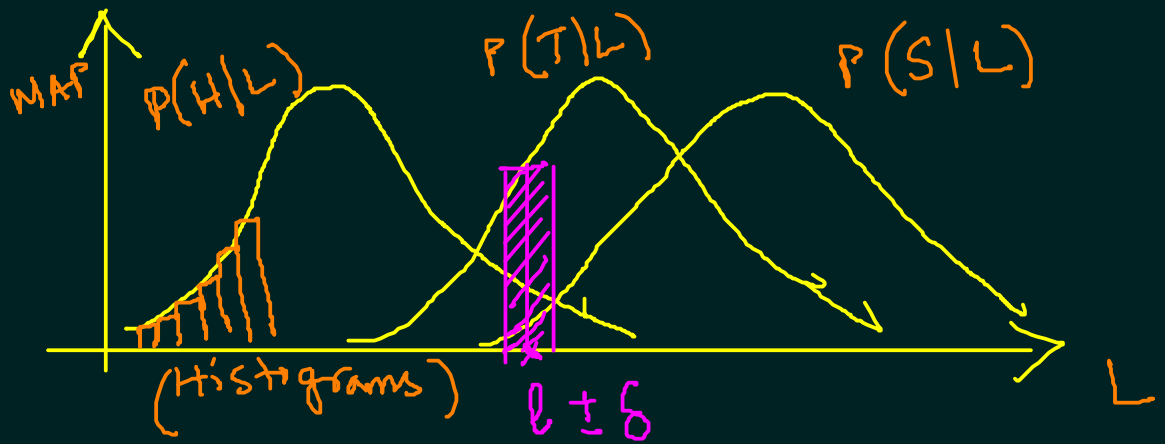
θ

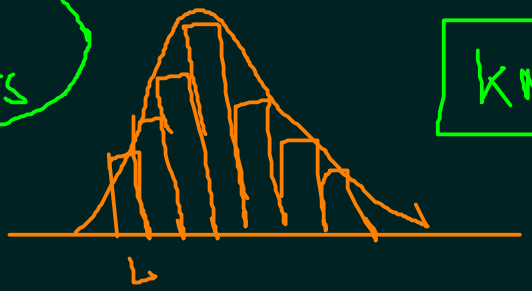


Non-parametric Learning :

k-Nearest Neighbours

KNN





$l = 2\,ft$

$\delta = 0.1$

$1.9\,ft \longleftrightarrow 2.1\,ft$

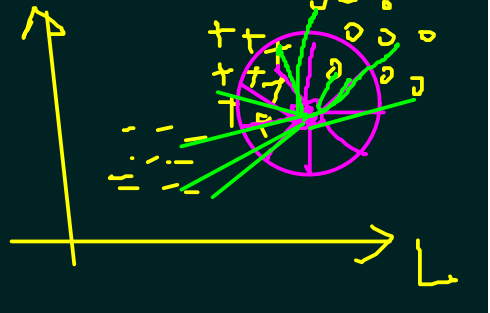
$N = N_H + N_T + N_S$

Choose class for l { vote and wins } → $f_H = \frac{N_H}{N}$, $f_T = \frac{N_T}{N}$, $f_S = \frac{N_S}{N}$

majority (N_H, N_T, N_S)

Top k

[Approximate MAP → (relative count of elements)]

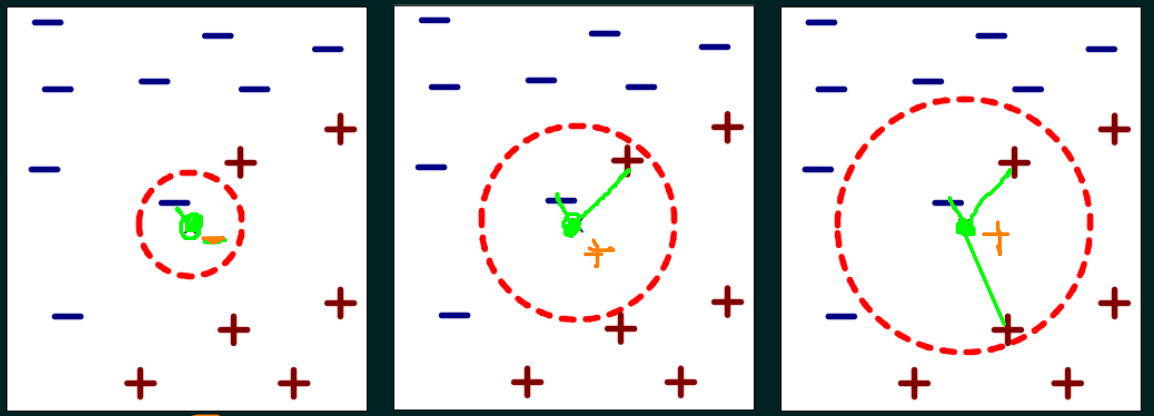


class ← [majority (N_H, N_T, N_S) ← rad = δ

$\checkmark$ $K_H + K_T + K_S = k$

distance between points

Nearest Neighbour Problem



k = 1 k = 2 k = 3

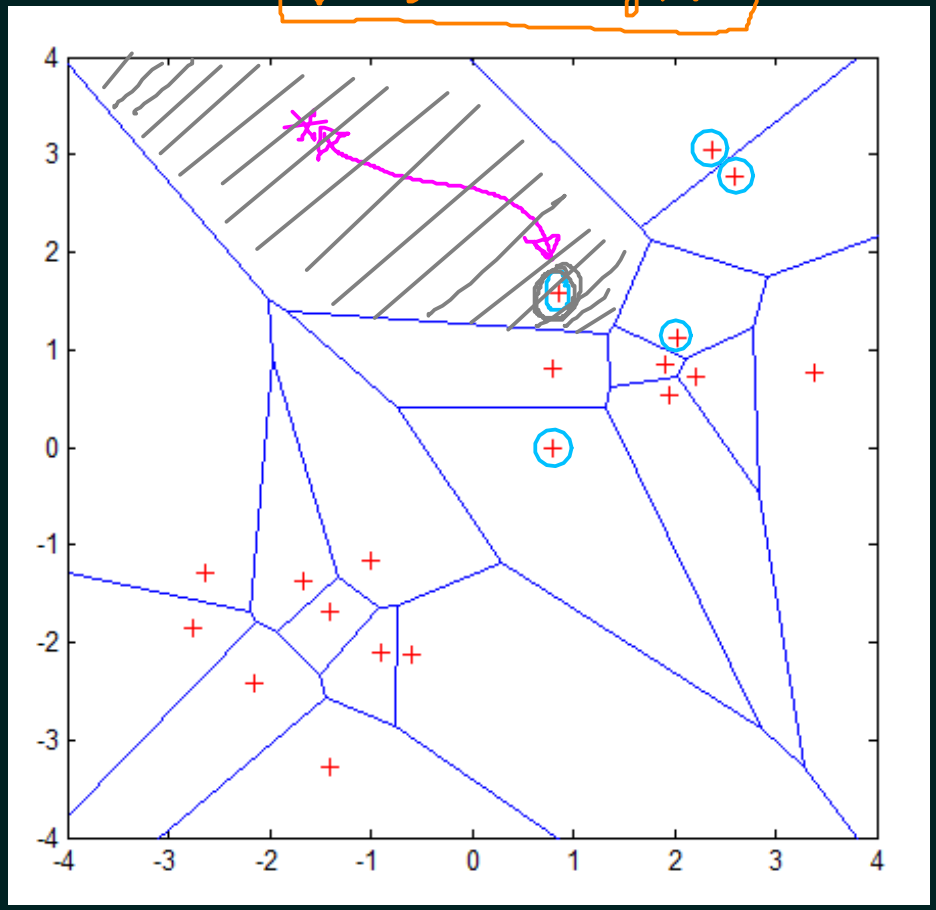
▶ Voronoi Diagram :

- Tessellation Cell
- Any unknown point
  ↳ Which cell it belongs

- Algorithm wise Straightfwd.
  ↳ Train → ✓✓
  ↳ Test → Simple

⊕ to your final

Voronoi Diagram



Distance weighted K-NN :

$$\hat{f}(q) = \underset{v \in V}{\operatorname{argmax}} \sum_{j=1}^{k} \delta(v, f(x_{ij}))$$

KNN

$x_1, x_2, \ldots, x_i, \ldots, x_n$ (+, +, −, *) — V: (+, −, o, *)

$[x_{i_1} \ldots x_{i_k}]$ — k top

q: x_1, x_2, x_3

$f(x_{i_1}) = +$

$f(x_{i_2}) = -$

$f(x_{i_3}) = o$

$f(x_{i_4}) = o$

$\delta(a,b) = 1, a=b$

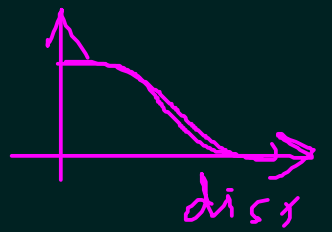
$= 0, a \neq b$

$v = o \rightarrow$ count =

$v = * \rightarrow$ count =

$v = + \rightarrow$ count =

$$\hat{f}(q) = \underset{v \in V}{\operatorname{argmax}} \sum_{j=1}^{k} \frac{1}{d(q, x_{ij})^2} \delta(v, f(x_{ij}))$$

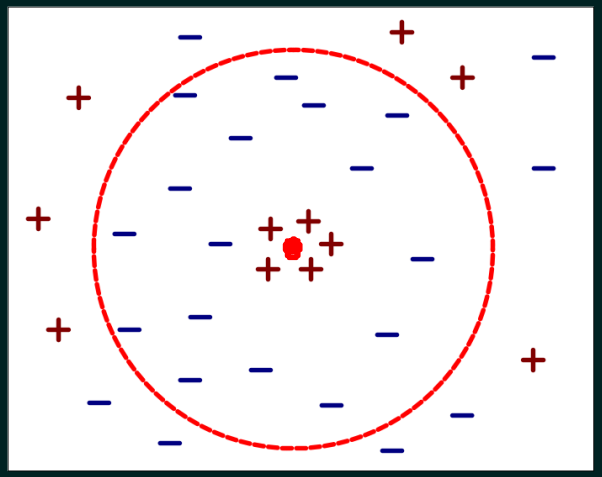


$\hat{f}(q) = \underset{v \in V}{\operatorname{argmax}} \sum_{j=1}^{k} w_j \, \delta(v, f(x_{ij}))$

— Continuous values

$$\hat{f}(q) = \frac{\sum_{j=1}^{k} w_j f(x_{ij})}{\sum_{j=1}^{k} w_j}$$

weighted Avg



- What is a good 'K'=?
  - Small K → too local
    - → noise ✗
  - K = N (all TE)
    - too global ←
    - → Prior (MAP), MLE (data likelihood)
- Typical : $K = \sqrt{N}$, $N = \#TE$; $K = \frac{N}{10}$ } practical est.
  - → (OPEN PROBLEM) ☹

Minkowsky:

$$D(\boldsymbol{x},\boldsymbol{y}) = \left(\sum_{i=1}^{m} |x_i - y_i|^r\right)^{1/r}$$

Euclidean:

$$D(\boldsymbol{x},\boldsymbol{y}) = \sqrt{\sum_{i=1}^{m} (x_i - y_i)^2}$$

Manhattan / city-block:

$$D(\boldsymbol{x},\boldsymbol{y}) = \sum_{i=1}^{m} |x_i - y_i|$$

Camberra:

$$D(\boldsymbol{x},\boldsymbol{y}) = \sum_{i=1}^{m} \frac{|x_i - y_i|}{|x_i + y_i|}$$

Chebychev:

$$D(\boldsymbol{x},\boldsymbol{y}) = \max_{i=1}^{m} |x_i - y_i|$$

Quadratic:

$$D(\boldsymbol{x},\boldsymbol{y}) = (\boldsymbol{x} - \boldsymbol{y})^T Q(\boldsymbol{x} - \boldsymbol{y}) = \sum_{j=1}^{m}\left(\sum_{i=1}^{m}(x_i - y_i)q_{ji}\right)(x_j - y_j)$$

Q is a problem-specific positive definite $m \times m$ weight matrix

Mahalanobis:

$$D(\boldsymbol{x},\boldsymbol{y}) = [\det V]^{1/m}(\boldsymbol{x} - \boldsymbol{y})^{\mathrm{T}} V^{-1}(\boldsymbol{x} - \boldsymbol{y})$$

V is the covariance matrix of $A_1..A_m$, and A_j is the vector of values for attribute j occuring in the training set instances $1..n$.

Correlation:

$$D(\boldsymbol{x},\boldsymbol{y}) = \frac{\sum_{i=1}^{m}(x_i - \overline{x_i})(y_i - \overline{y_i})}{\sqrt{\sum_{i=1}^{m}(x_i - \overline{x_i})^2 \sum_{i=1}^{m}(y_i - \overline{y_i})^2}}$$

$\overline{x_i} = \overline{y_i}$ and is the average value for attribute i occuring in the training set.

Chi-square:

$$D(\boldsymbol{x},\boldsymbol{y}) = \sum_{i=1}^{m} \frac{1}{sum_i}\left(\frac{x_i}{size_x} - \frac{y_i}{size_y}\right)^2$$

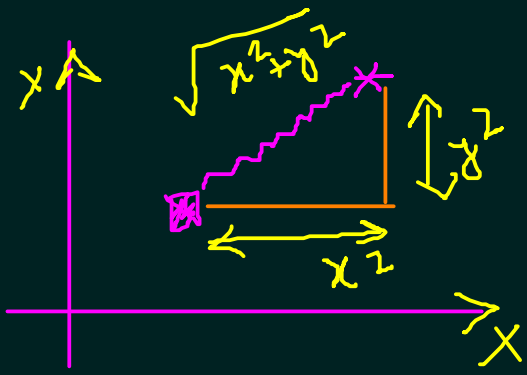
sum_i is the sum of all values for attribute i occuring in the training set, and $size_x$ is the sum of all values in the vector $\boldsymbol{x}$.

Kendall's Rank Correlation:

$$D(\boldsymbol{x},\boldsymbol{y}) = 1 - \frac{2}{n(n-1)}\sum_{i=1}^{m}\sum_{j=1}^{i-1}\mathrm{sign}(x_i - x_j)\mathrm{sign}(y_i - y_j)$$

sign(x)=-1, 0 or 1 if $x < 0$, $x = 0$, or $x > 0$, respectively.

Figure 1. Equations of selected distance functions. ($\boldsymbol{x}$ and $\boldsymbol{y}$ are vectors of m attribute values).



- Euclidean
- Manhattan
- Minkowsky

↳ Correlation

Distance Metrics

Issue 1

	a_1	a_2	a_3	a_4	a_5
q =	0	1	1	1	1
x =	1	1	1	1	0

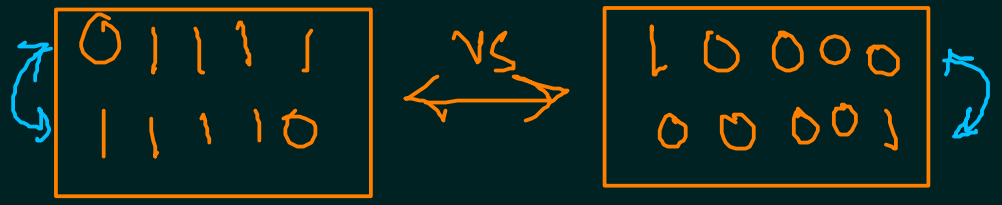
$\Rightarrow d = \sqrt{2}$

$q = 1\ 0\ 0\ 0\ 0$

$y = 0\ 0\ 0\ 0\ 1$

$d = \sqrt{2}$

Euclidean



→ Data org.

→ Curse of dimensionality

Issue 2

a_1 = age [0 – 100]

a_2 = blood sugar [ ]

a_3 = bp

a_4 = [0.01 — 0.04]

↳ Normalize

$X_i = (x_{ij})$

μ_i = Mean (X_i)

σ_i = SD (X_i)

$$Z_{ij} = \left(\frac{x_{ij} - \mu_i}{\sigma_i}\right)$$

N = 20

$K = \lfloor \sqrt{N} \rfloor = 4$ → 5 (odd)

GB	CF	LW	HL
x_1 = Y	N	Y	Y
x_2 = Y	Y	N	N

$d(x_1, x_2)$ → 1

carefully $= \frac{3}{4}$

Name	Give Birth	Can Fly	Live in Water	Have Legs	Class
human	yes	no	no	yes	mammals
python	no	no	no	no	non-mammals
salmon	no	no	yes	no	non-mammals
whale	yes	no	yes	no	mammals
frog	no	no	sometimes	yes	non-mammals
komodo	no	no	no	yes	non-mammals
bat	yes	yes	no	yes	mammals
pigeon	no	yes	no	yes	non-mammals
cat	yes	no	no	yes	mammals
leopard shark	yes	no	yes	no	non-mammals
turtle	no	no	sometimes	yes	non-mammals
penguin	no	no	sometimes	yes	non-mammals
porcupine	yes	no	no	yes	mammals
eel	no	no	yes	no	non-mammals
salamander	no	no	sometimes	yes	non-mammals
gila monster	no	no	no	yes	non-mammals
platypus	no	no	no	yes	mammals
owl	no	yes	no	yes	non-mammals
dolphin	yes	no	yes	no	mammals
eagle	no	yes	no	yes	non-mammals

d = 1/4 (salmon), d = 0 (whale), d = 0 (leopard shark), d = 1/4 (eel), d = 0 (dolphin)

q:

Give Birth	Can Fly	Live in Water	Have Legs	Class
yes	no	yes	no	?

$d(x_1, q) = 1/2$

$d(x_2, q) = 1/2$

$d(x_3, q) = 1/4$

$d(x_4, q) = 0$

Top k

2-class problem

→ K = odd

— Algo. Intricacies

NON-MAMMEL