K-nearest neighbour for

Problem with Parzen window : How to choose V?

If V is too small → may cells are empty
leads to discontinuous
estimates.

V too Large → oversmoothing due to averaging.

Soln: fix K & make V a function of data

Called the Knn approach for density estimation

To estimate $P(x)$ at a point x,

Centre a cell around x & keep growing it
till it encounters K_n samples.

Then $$P(x) = \frac{K_n}{V_n \cdot n}.$$

V_n - Volume that encompasses K_n samples.

If $p(x)$ is high around x, V_n will be small

→ leads to a good resolution.

If $p(x)$ is low then V_n would be large without oversmoothing.

This method also leads to a simple classifier : Knn.

$$D^n = \{x_1, x_2, \dots x_n\}$$

Knn assigns a new point x' with the label that of a $x' \in D^n$ that is closest to x.

To know why this works, consider the following:

Given a test point x, suppose we want to estimate the posterior $P(y = y_i \mid x)$.

Now, Lets place ~~a~~ cell of volume V around x that captures K samples.

Let K_i be the number of samples that are of class i. $K_i \leq K$.

Now, An estimate for

$$\hat{P}(x, Y = y_i) = \frac{K_i/n}{V}$$

$$\Rightarrow P(Y = y_i \mid x) = \frac{P(x, Y = y_i)}{\sum_{j=1}^{M} P(x, Y = y_j)} = \frac{K_i}{K} \left[\frac{\frac{K_i/n}{V}}{\frac{K/n}{V}}\right]$$

Thus, $Q_i(x)$ is the number of points belonging to class i, within a volume relative to number of points in the cell.

Thus, given a test point (x) one could select K nearest point to it [growing a cell] & decide its class based on the ratio of number of points belonging to each class within those K- ~~classes~~ Points.

(91).

Scanned by CamScanner

Thus Knn is a sort of Baye's Classifier with the density estimates being Knn.

In fact, one can show that

$$\text{Prob}(\text{error})_{KNN} \text{ is atmost twice the}$$

Bayes error.

Thus, One should always try Knn on a data first.

Clustering

Given $D = \{x_1, x_2, \dots x_n\}$ without Labels,

Partition D into K disjoint subsets. (called ~~Ca~~ clusters)

such that within-cluster 'similarity' is higher

& 'inter-cluster' similarity' is lower.

One popular algorithm - K-means Clustering.

Input: D, $x_i \in R^n$ & number of clusters K.

Initialize K means (centroids) $\mu_1, \dots \mu_K \in R^d$

Iterate:

1. Assign each example to its 'closest' cluster Center.

$$C_K = \{n : K = \underset{K}{\operatorname{argmin}} \|x_n - \mu_K\|^2\}$$

set of all examples assigned to μ_K.

2. uptate μ_1, $\quad \mu_K = \text{mean}(C_K) = \frac{1}{\|C_K\|} \sum_{n \in C_K} x_n.$

. Repeate 1, & 2

Very sensitive to initialization.

Convergence criteria are many - one possible way - stop with μ_s do not change.

Formal analysis of k-means:

For every training example x_i,

Define $Z_i = \{0, 0, \ldots 1, \ldots\} \in R^k$.

i^{th} place

such that $Z_{ij} = \mathbb{I}[x_i \in \text{cluster } j]$

$$Z_i = [z_{i1}, z_{i2}, \ldots, z_{ik}].$$

Now objective of k-means: Find μ & C

such that $$L_{\theta}(z, \mu) = \sum_{i=1}^{n} \sum_{j=1}^{k} z_{ij} \| x - \mu_j \|_2^2.$$

∈ The above can be interpreted as the loss suffered on assigning points in D to $\{\mu_j\}_{j=1}^{k}$

→ ~~No~~ $\cancel{L}$ is non-convex & NP-hard.

(9a).

Solution: Iteratively minimize over μ & z.

Let us assume we know z, then

$$L = \sum_{i=1}^{n} \sum_{j=1}^{k} z_{ij} \| x - \mu_j \|_2^2$$

$$= \sum_{I[i \mid x_i \in j]} \| x_i - \mu_j \|^2$$

$$\Rightarrow \mu_j = \frac{1}{I_{i \mid x_i \in j}} \sum x_i$$

For fixed μ, minimizing over z is possible,

$$L = \sum_{i=1}^{n} \sum_{j=1}^{k} z_{ij} \| x_i - \mu_j \|^2$$

$\forall\, i$, we have one of $z_{ij} \| x_i - \mu_j \|^2$

term nonzero.

$$\therefore \cancel{\text{Let}}\ z_{ij} = I\left[j = \arg\min_c \| x_i - \mu_c \|^2 \right).$$

Exactly EM!

This suggests that one can use GMM for clustering as well.

Given data, one can fit a K-Component GMM to it with every component resulting in a cluster.

Now, recall $w_{ij} = P\left[z_i = j \mid x_i\right]$

The hidden variable z represents the cluster.

Observe that a testpoint x_i can "belong" to all clusters with some probability unlike K-means.

re

Recall that

$$w_{ij} \propto \alpha_{ij} \cdot P(x_i \mid z_i = j).$$

if ~~iff~~ $P(x_i \mid z_i = j) \sim \mathcal{N}(\mu_j, \Sigma_j)$

& $\Sigma_j = \sigma^2 I$

then $w_{ij} \propto \alpha_j \exp\left\{ \frac{-1}{2\sigma^2} \| x_i - \mu_j \|^2 \right\}.$

Now if $\sigma^2 \to 0$,

$$w_{ij} = \frac{\alpha_j \exp\left\{ \frac{-1}{2\sigma^2} \| x_i - \mu_j \|^2 \right\}}{\sum_{j=1}^{K} \alpha_j \exp\left\{ \frac{1}{2\sigma^2} \| x_i - \mu_j \|^2 \right\}}$$

The summation in the denominator is dominated by the smallest $\| x_i - \mu_j \|^2$.

$\therefore$ For that j,

$$w_{ij} \approx \frac{\alpha_j \exp\left\{ \frac{1}{2\sigma^2} \| x_i - \mu_j \|^2 \right\}}{\alpha_j \exp\left\{ \frac{1}{2\sigma^2} \| x_i - \mu_j \|^2 \right\}} = 1.$$

for all other clusters, $i \neq j$,

$$w_{ij} = 0$$

(9j).

This is a hard assignment rule. ~~wit~~

Thus, with $\Sigma_j = \sigma^2 I$ & $\sigma^2 \to 0$, GMM reduces to K-means.

Principal Component Analysis.

Often data suffers from the curse of dimensionality. [we saw one example with Parzen windows]

In high dimensions, data points are sparsely occupied.

In other words, most of the 'useful' information is contained in a ~~few~~ lower dimension sub-manifold inside the data-space.

Thus, it is desirable if one can 'transform' the data into a lower-dim space such that 'most' of the information is preserved.

PCA is one linear technique to do it.