18.3 knn

Zaki - Data Mining & ML, 2020

knn is a non-parametric approach

This means it does not make any assums about the underlying distribution of the data

A parametric method assumes that the data follows a specific prob dist, i.e Normal

These methods try to estimate the parameters of that distribution to make pred
i.e. mean & Standard Deviation

Non-parametric does not do this

Uses the data itself to make predictions
Without trying to fit a predefined model

The knn works by locating the near-est neighbours of known/training data compared to an unseen data point

Then uses those labels/data to create pred

There is ~~no~~ no equation or function that describes new data

If you ~~loose~~ lose the data, you lose the ability to pred

Where as lin reg can be performed w/ just the equation & new data

This is also why knn is heavy on the cost to compute

The lack of assumption allows fitting w/ complex data but exposes risk to overfitting

D = train set

n = points , $x_i \in \mathbb{R}^p$

D_i = Subset of D w/ class c_i w/ $n_i = |D_i|$

test point = $x \in \mathbb{R}^d$

K = number of neighbours

r = Dist of x test point to k^{th} neighbour in D

$$B_d(x, r) = \{ x_i \in D \mid \|x - x_i\| \leq r \}$$

↓ mean Ball

↳ eq for a ball radius around x.

$$|B_d(x, r)| = K$$

k_i = num of neighbours/Points among the k nearest neighbours of x (test point) that are labelled w/ class c_i

$$k_i = \{ x_j \in B_d(x, r) \mid y_i = c_i \}$$

$|$ = such that or where

The classes conditional probability & density @ point x can be estimated as the fraction of points from class C_i that lie within the ball/hyperball Divided by its volume

$$\hat{f}(x|C_i) = \frac{K_i/n_i}{V} = \frac{K_i}{n_i * V}$$

K_i = Points of class in ball, n_i = num of points class C_i is training, V = Volume of Ball

using postier Probability $P(C_i|x)$ eq:

$$P(C_i|x) = \frac{\hat{f}(x|C_i)\hat{P}(C_i)}{\sum \hat{f}(x|C_j)\hat{P}(C_j)}$$

This an application of Bayes theorem in the content of classification

Postier Prob = Prob that point x belongs to class C_i given observed features of x

General Bayes = $P(A|B) = P(B|A) * P(A) / P(B)$

Here, A = event that X belongs to C_i

$P(A|B) = P(C_i|x)$ = Postier Prob

Prob that x belongs to C_i given feature of x

$\hat{f}(x|C_i)$ = class conditional prob Dens Func

Prob density of observing x feature given we know it belongs to class X,

i.e. how likely are these features for class C_i

$P(C_i)$ = Prior prob of just being class C_i in the training data

$\sum \hat{f}(X_i|C_j)\hat{P}(C_j)$ = This normalization or evidence term. Sums to the product of the likelihood for all classes. ensures the posteir probs sum to 1 to form a vaalid prob distribution

In english, the equation calculates the posterior probability for class C_i by:

(1) Multiply the prob of observing x given class C_i by prior (overall) prob of class C_i in training data

(2) Normalize by sum of similar Products for all classes

However $\hat{P}(C_i) = \frac{n_i}{n}$, So:

$$\hat{f}(x \mid C_i)\,\hat{P}(C_i) \overset{(TOP)}{=} \frac{k_i}{n_i V} \cdot \frac{n_i}{n} = \frac{k_i}{nV}$$

thus posterior probability:

$$P(C_i \mid x) = \frac{k_i / nV}{\sum_{j=1}^{k} k_j / nV} = \frac{k_i}{k}$$

k_i = neighbours of Class C_i, k = Total neighbors

finally, the predicted class for x:

$$\hat{y} = \underset{c_i}{\operatorname{argmax}} \{P(c_i|x)\} = \underset{c_i}{\operatorname{argmax}} \left\{\frac{k_i}{k}\right\} = \underset{c_i}{\operatorname{argmax}} \{k_i\}$$

$\{k_i\}/k$ represents the prop of k nearest neighbours belong to c_i

$\operatorname{argmax}\{k_i\}$ means the pred $\hat{y}$ is whatever class c_i has the highest count