

KNN & visualize Algo performance

→ we have a dataset i.e., labelled into two classes A & B

→ we have random points in the space and we need to classify.

→ Take distance from each of the labelled points and choosing K-nearest neighbours, if majority are from one class it inherits that class.

$K=1$ → nearest / adj. neighbour.

Euclidean distance.

We trained data and a query for which query it belongs to.

→ If out of our $K=7$ points 5 belong to A, 2 belong to B
⇒ class A

Iris Dataset

weighted KNN → the nearest K must have the highest say

Dist ↓

Visualizing Data

Voronoi diagrams help you to visualize the dataset by partitioning the plane into regions that are close to a given set of points.

→ Each partition has its own dominance.

ex: miss-classified; in green region some are

decision boundaries → classification tasks.

weighted KNN

* Inductive bias of K-NN

- ↳ Any model, assume regressⁿ/classⁿ. if n feature space close \Rightarrow same class (NBD)
- Nearby instances should have the same label [that most of the cases in a small NBD in feature space belong to the same class].
- All features are equally imp (standardized)
- Complexity is tuned by the K parameter.

Ex: mm (x-axis) stretched } \Rightarrow Imp of standardization.
cm (y-axis) short

* weighted KNN

- If there is a tie in majority labels, one can do weighted voting.

Q. how is wt determined.

- Samples are weighted by inverse of distance to point P .

$w = 1/D$ weighing the samples.

$$w_i = \frac{1}{1 + d(P_i, x_i)}$$

larger wt

its own neighbour

distance : 0 \Rightarrow add 1.

wt may blow up

range must be b/w 0 & 1

Ex: Blue:

dist: 1

wt: