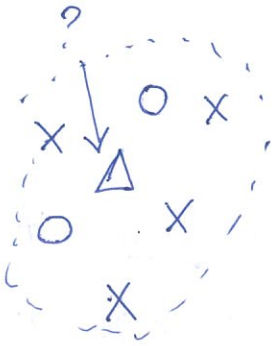
Lecture 5

Classifier: a learned program to make predictions.

Cover & Hart (1967)

NN classifier's assumption: Close (similar) points should have similar labels.

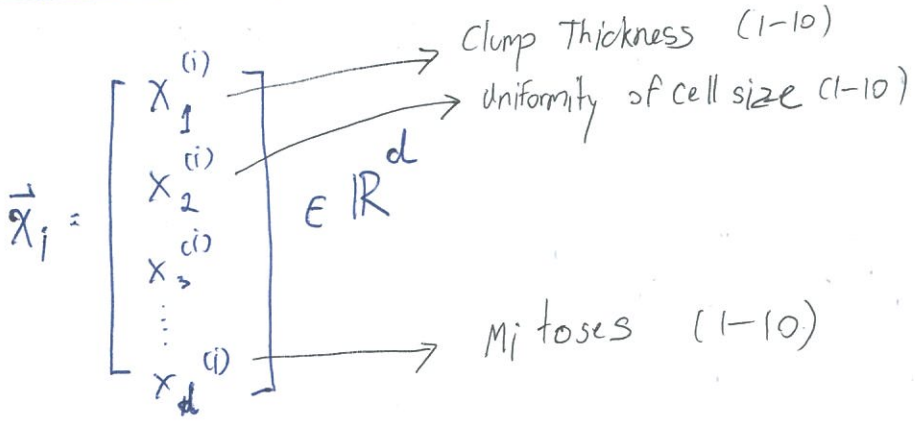


Rule: The label of the test point is determined by the majority of k-NN.

Implication of the assumption: NN classifier makes good prediction if the distance function reflects the similarity among points.

Downside: What if it doesn't reflect at all?

Dense / sparse representation of feature vectors:



- Patient data: d is relatively small.
- Text document (B.o.W.): d is quite large (in English $d \approx 170k$).
- Image data: d is very large (7 MP $\Rightarrow d = 21$ M)

• Dense/sparse Feature vectors: A feature vector $\vec{x}_i$ is dense if the number of nonzero coordinates in $\vec{x}_i$ is large relative to d. Otherwise, it is sparse.

Quiz II: (a.) Is the feature vector representing patient data is dense or sparse? Why?

(b.) Is the feature vector representing text document is dense or sparse? Why?

(c.) Is the feature vector representing image is dense or sparse? Why?

(d.) Why would there be a problem if we have sparse representation of feature vectors? (bad complexity)

Curse of Dimensionality:

- In high dimensional space, points drawn from a probability distribution tend to never be close together.

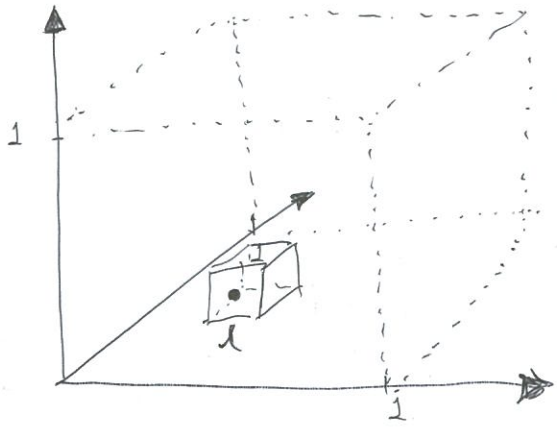
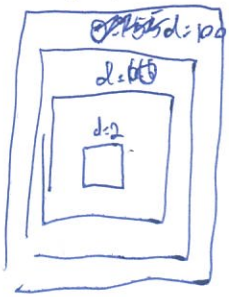


Illustration: imagine the unit (hyper) cube $[0,1]^d$. All training data is sampled uniformly within the cube.

Let's consider $k = 10$ NN of a test point.

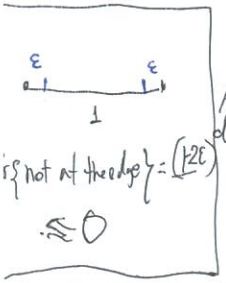
- Let l be the edge length of the smallest hyper-cube that contains all k-NN. Then, $l^d \approx \frac{k}{n} \Rightarrow l \approx \left(\frac{k}{n}\right)^{1/d}$

- If $n = 1000$, how big is l?



d	l
2	0.1
10	0.63
100	0.955
1000	0.9954

Data with high dimensional structure will be cursed with k-NN



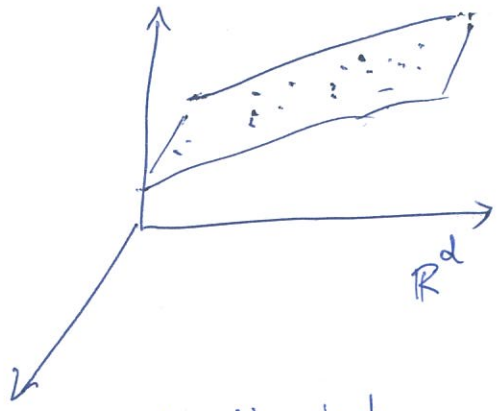
As $d \gg 0$, almost the entire space is needed to find 10-NN. ($\Rightarrow$ 10-NN are not closer) located on the edges of the cube.

Rescue: Increase the number of training samples, n, until the nearest neighbors are truely close to the test point

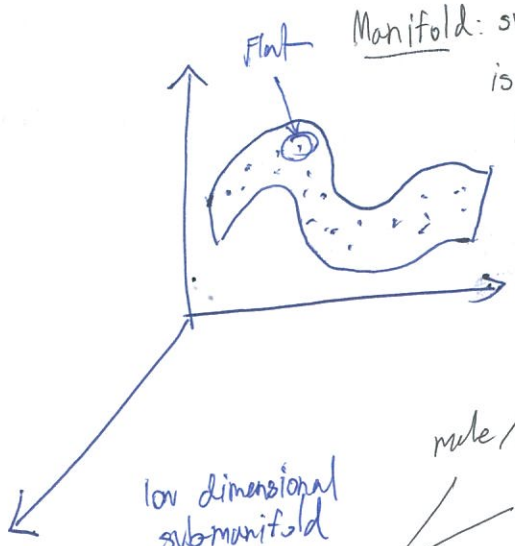
- Problem: $l = \frac{1}{10} = 0.1 \Rightarrow n = \frac{k}{l^d} = k \cdot 10^d$!!

Data with low dimensional structure (special case)

- Data may lie in low-dimensional subspace.

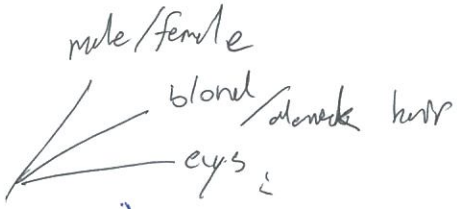


low dimensional subspace (plane)



Manifold: subspace that is locally Euclidean but globally not Euclidean

low dimensional submanifold



- Ex: Human face image (18M → 50 attributes)
  *** Pictures are not uniformly distributed (PCA, SVD finds new system coordinates to reduce to low dimension)

K-NN summary:

- K-NN is a good classifier if distances reflect a notion of similarity.
- As $d \gg 0$, the k-assumption breaks down
- Remark: As $n \to +\infty$, k-NN becomes provably accurate, but also slow.

Demo:

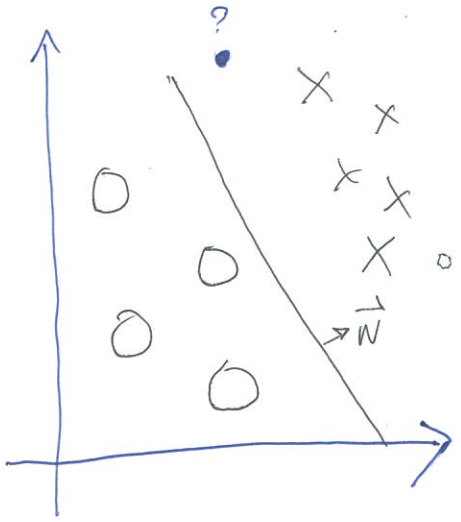
- NN w/ different value of k
- Cross validation
- Demo For curse of dimensionality

Tips: - Apply pre-processing to reduce dimension of your data (time complexity $O(nd)$)

Downside

go to every single coordinate and treating data

The Perceptron: (Frank Rosenblatt 1957)



Assumption: There is a hyperplane that separates one class from another

- For high dimensional space, it almost always holds.
- Training Goal: Find such hyperplane

$$\mathcal{H} = \{ x \mid \vec{w}^T \vec{x} + b = 0 \}$$

In two dimensional space a hyperplane is just a line.

Testing $\text{sign}(\vec{w}^T \vec{x} + b)$

$y = \{-1, +1\}$; Goal: learn $\vec{w}$ and b

$$\vec{x}_i \rightarrow \begin{pmatrix} \vec{x}_i \\ 1 \end{pmatrix} \qquad \vec{w} \rightarrow \begin{pmatrix} \vec{w} \\ b \end{pmatrix}$$

$$w^T x + b$$