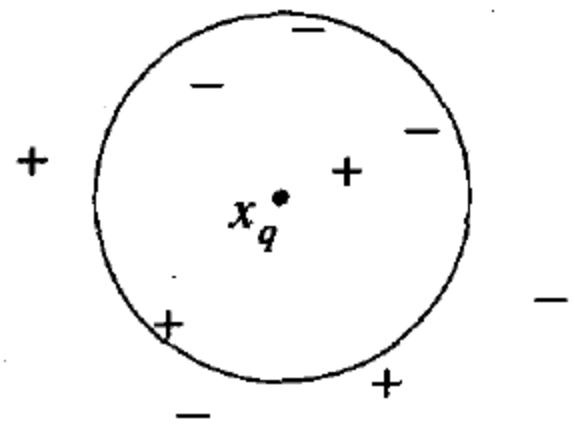
K-Nearest Neighbor Learning

K-Nearest Neighbor

- Here we assumes all instances correspond to points in the n-dimensional space
- let an arbitrary instance $\boldsymbol{x}$ be described by the feature vector $\begin{bmatrix} a_1(x) \\ a_2(x) \\ \vdots \\ a_n(x) \end{bmatrix}$
- where $a_r(x)$ denotes the value of the r^{th} attribute of instance $\boldsymbol{x}$.

Classification

- The class of an unknown instance (test instance) is predicted using the classes of its nearest training instances in the space



Distance Measure

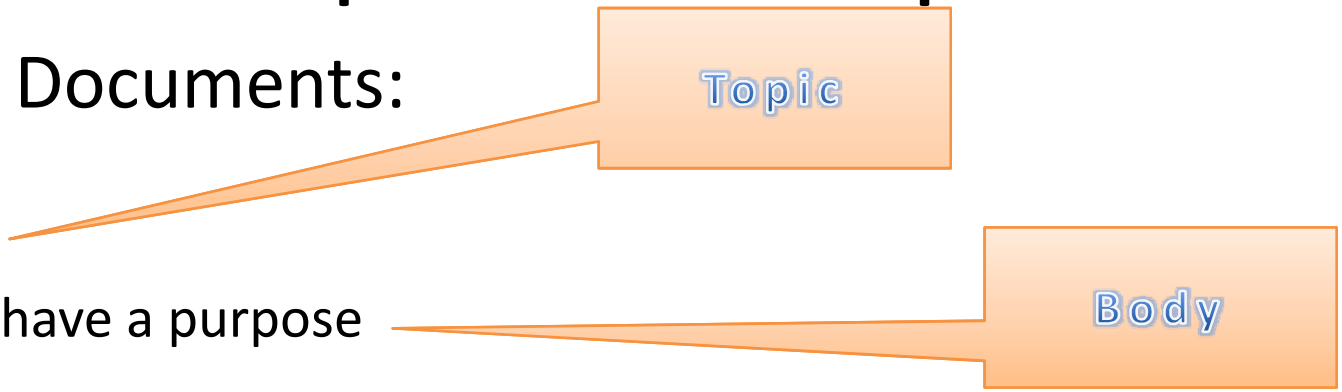
- We need to measure distance between 2 instances
  - Hamming Distance
  - Euclidean Distance
  - Cosine Similarity

Text Classification

- The goal is to identify the topic (the class) for a piece of text (document)
- Now lets use KNN for Text Classification
- Here each document is an instance

A Simplified Example

- Training Documents:
  - D1:
    - **Life**
    - We have a purpose
  - D2:
    - **Death**
    - We will die
- Test Documents
  - D3:
    - **???**
    - A real purpose
- Here each **word** is a feature
- We represent each **document** as a vector



Hamming Distance

- $\begin{bmatrix} we \\ have \\ purpose \\ will \\ die \\ real \end{bmatrix}$ D1: $\begin{bmatrix} 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$ D2: $\begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \end{bmatrix}$ D3: $\begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 1 \end{bmatrix}$
- hd(D1,D3)=3 , hd(D2,D3)=5
- So D1 is the nearest neighbor to D3

Euclidean Distance

- Put word frequencies of the document in the corresponding cell
- Then the Euclidean distance between two instances $\boldsymbol{x_i}$ and $\boldsymbol{x_j}$

$$d(x_i, x_j) \equiv \sqrt{\sum_{r=1}^{n} (a_r(x_i) - a_r(x_j))^2}$$

Cosine similarity with TF-IDF weights

- now each number is the TF-IDF weight for the corresponding word
- Let **w** be a word, **d** be a document, **N(d,w)** be the number of occurrences of **w** in **d**
- **TF(d,w)** = **N(d,w)** / **W(d),** where **W(d)** is the total number of words in **d**
- **IDF(d,w)** = log(**D** / **C(w)**), where **D** is the total number of documents, and **C(w)** is the total number of documents that contains the word **w**
- The TF-IDF weight for **w** in **d** is **TF(d,w)*IDF(d,w)**

Cosine similarity

- Cosine similarity between documents D_1 & D_2

  $$\cos\theta = \frac{D_1 . D_2}{|D1||D2|}$$

- **Notice**
  - with a distance measure, the k-nearest neighbors are the ones with the smallest distance from the test point
  - whereas with a similarity measure, they are the ones with the highest similarity scores.

- Try k = 1, k = 3 and k = 5 with each of the Distance measure