Natural Language Processing (NLP)

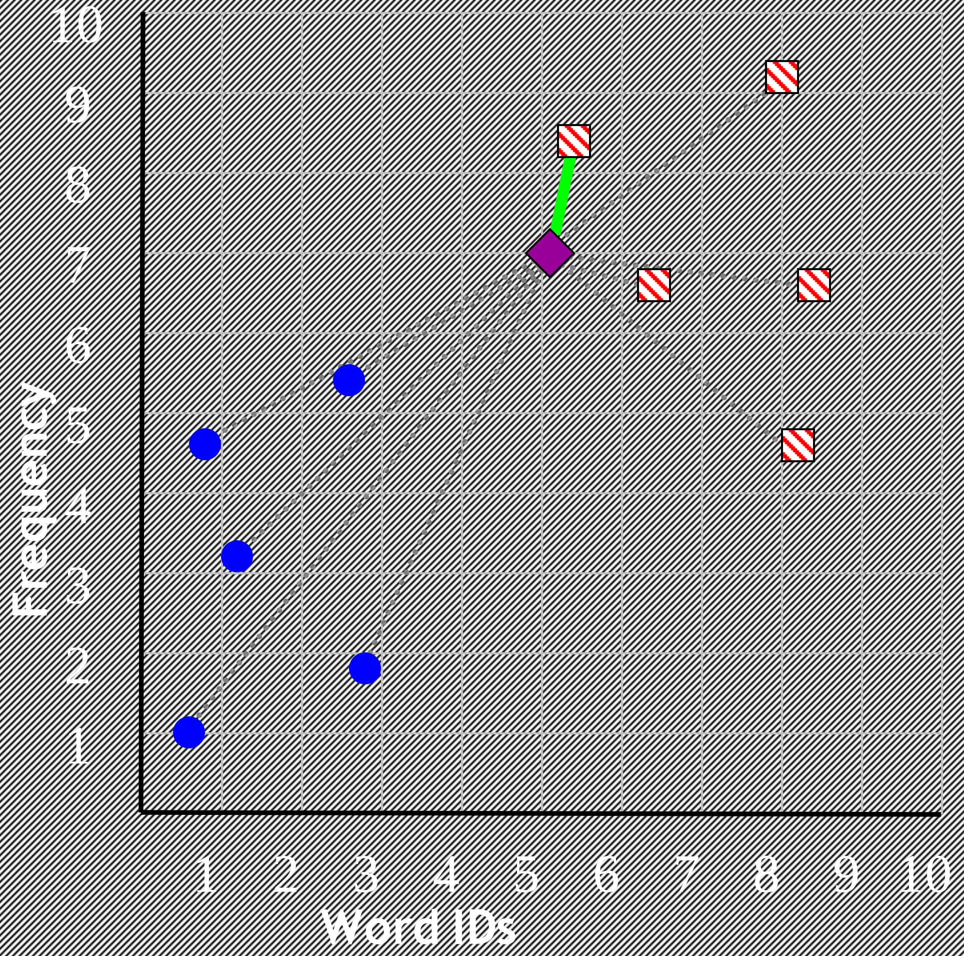
Lecture 11: Document Classification

Dr. Muhammad Taimoor Khan

Nearest Neighbor Classifier

- Calculate distance of the test document with all the training documents.
- The training document that has the shortest distance (or highest similarity) with the test document has its label assigned to the test document.

Nearest Neighbor Classifier



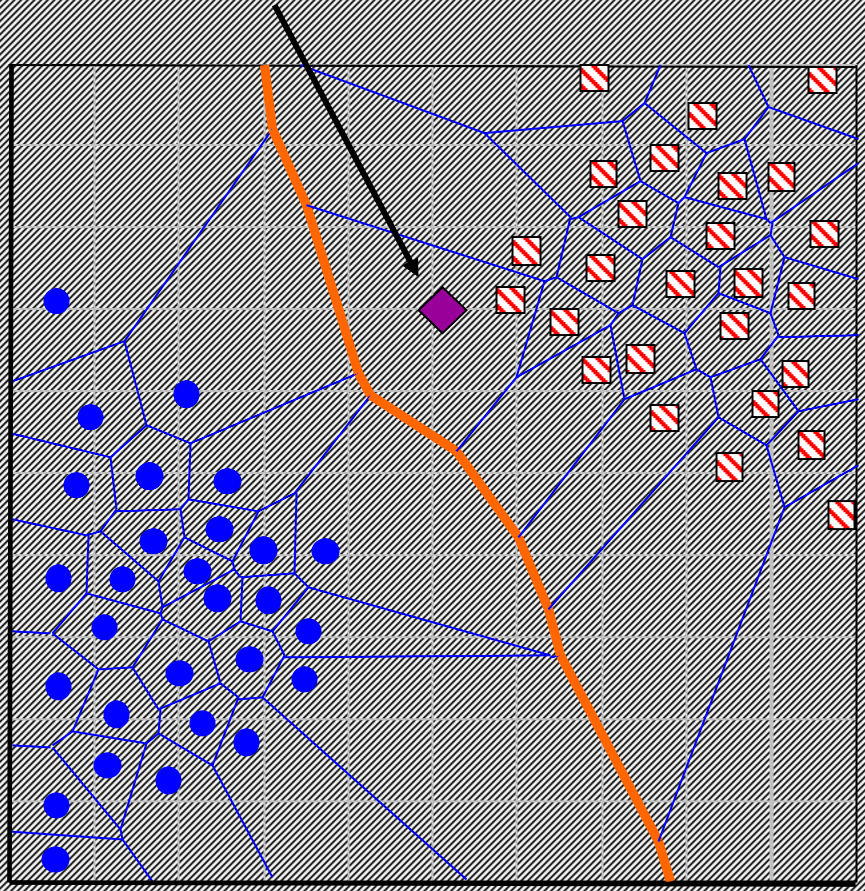
```
If the nearest document to the previously unseen document is a physics
    class is physics
else
    class is chemistry
```

▨ physics
● chemistry

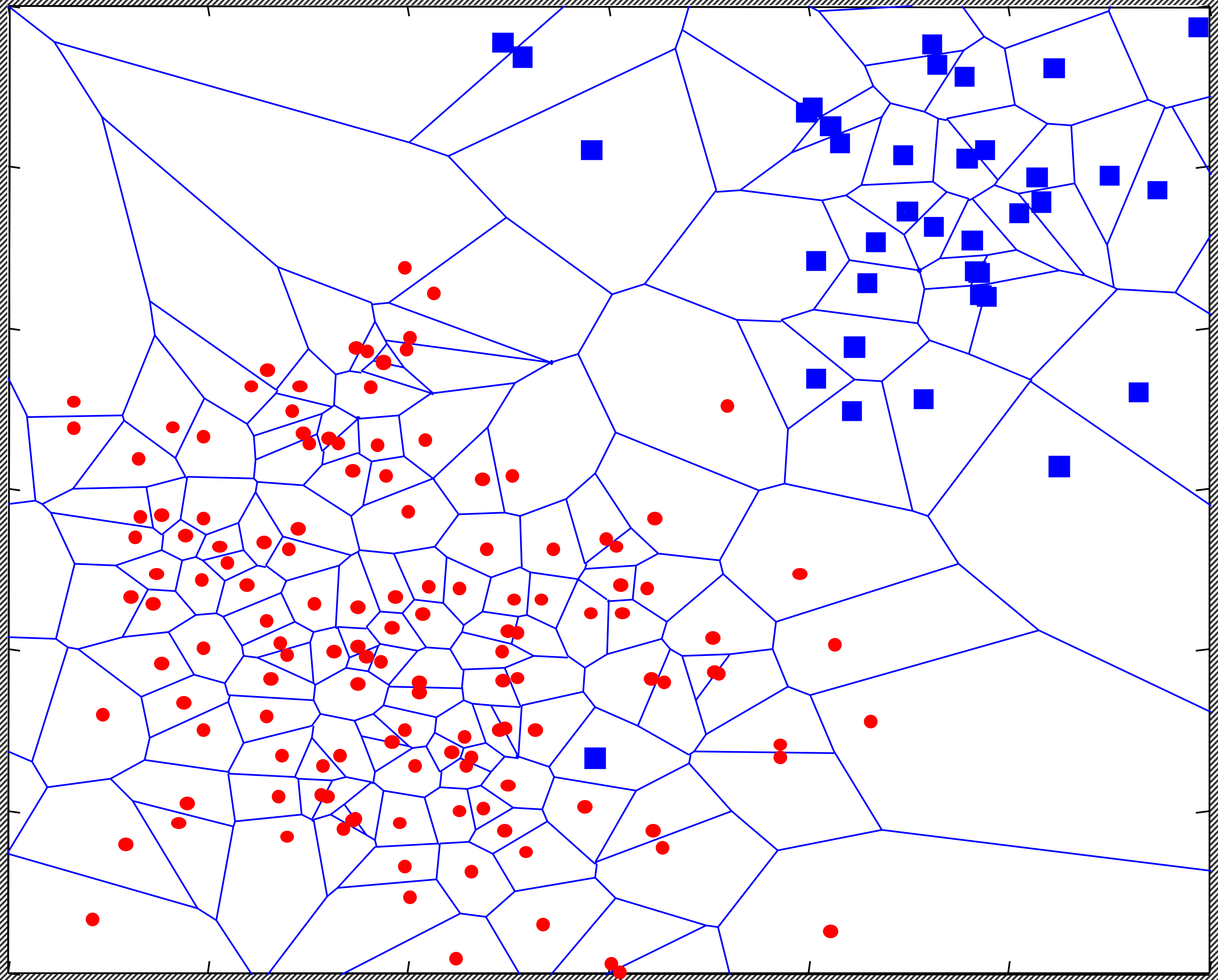
We can visualize the nearest neighbor algorithm in terms of a decision surface…

Note the we don't actually have to construct these surfaces, they are simply the implicit boundaries that divide the space into regions "belonging" to each instance.

This division of space is called Dirichlet Tesselation (or Voronoi diagram, or Theissen regions).



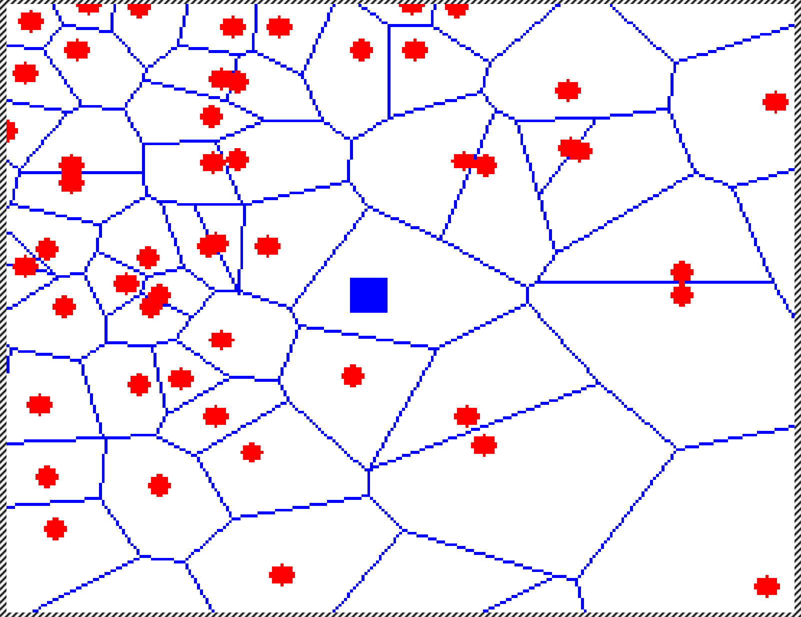
The nearest neighbor algorithm is sensitive to outliers…



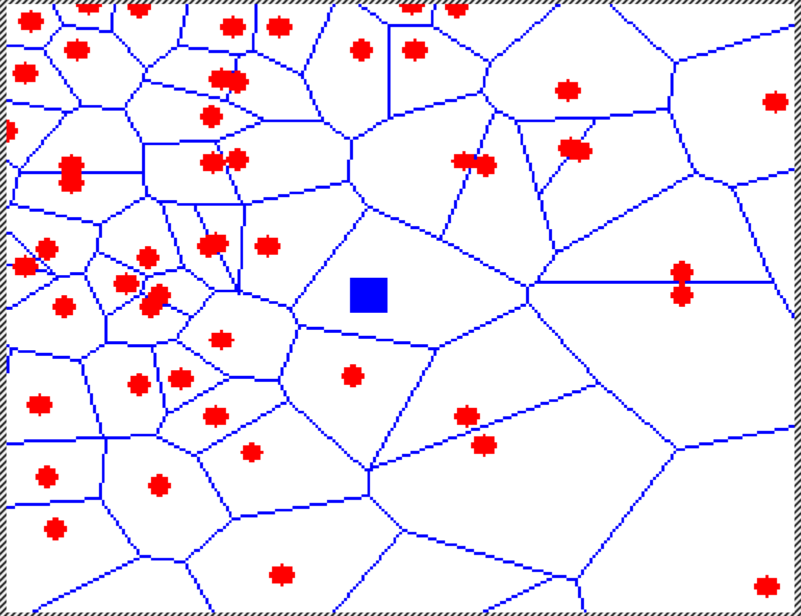
The solution is to…

We can generalize the nearest neighbor algorithm to the K- nearest neighbor (KNN) algorithm.

We measure the distance to the nearest K instances, and let them vote. K is typically chosen to be an odd number.



K = 1



K = 3

K-Nearest Neighbor

- Voting among K nearest neighbors *(One example may not be enough for decision making)*
- Makes no special attempt to learn (Model) from the data
- Provides no special value in finding generalized patterns

A type of Instance based learning

Instead of performing explicit generalization, compare new instances with instances in training data

Choosing the value of K

- Too small: sensitive to noise
- Too large: may include points from other classes
- Fine tune values by evaluating various values with cross validation

Normalization

- To bring different attribute values to the same scale [0, 1].
- Any given range can be adjusted to have the lowest value set to 0 while the highest set to 1.

- Allows to balance the impact of each attribute on the final classification

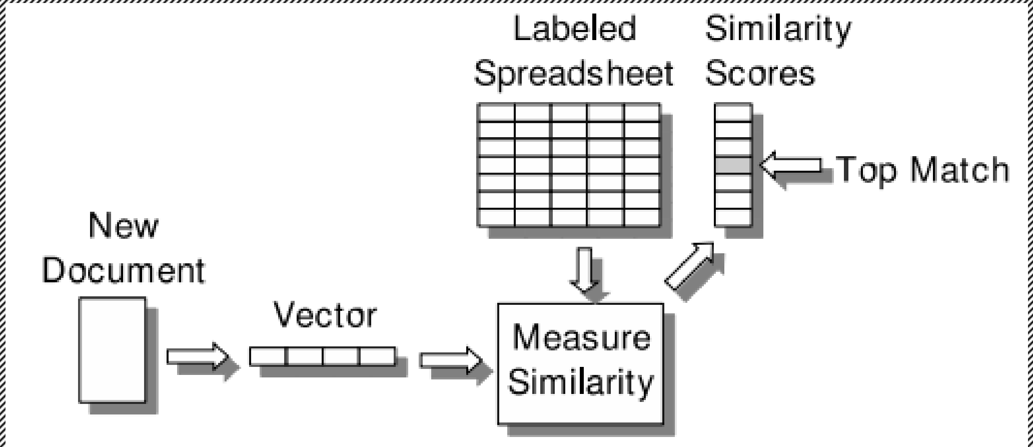
Distance Measure

- Document to document distance / similarity
- Distance measures:

$$Manhattan\ Dist(x,y) = |x_1 - y_1| + |x_2 - y_2| + \cdots + |x_n - y_n|$$

$$Euclidean\ Dist(x,y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$$

Cosine similarity

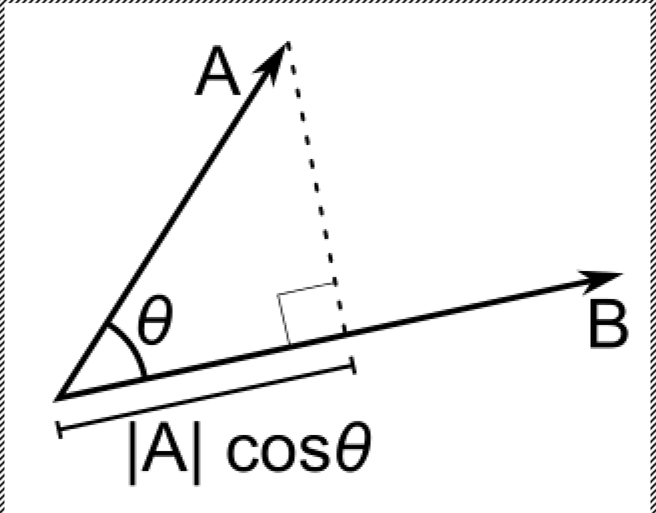


Why cosine similarity!

- Unlike other distance measures, it is least effected by the frequency change of the occurring words

Cosine similarity

$$\cos(\theta) = \frac{\sum_{i=1}^{n} x_i y_i}{\sqrt{\sum_{i=1}^{n} x_i^2}\sqrt{\sum_{i=1}^{n} y_i^2}}$$



Instance vs Model based Learning

- **Instance based Learning**
  - Also called memory based learning
  - No model to learn, the stored instances themselves represent the knowledge
  - Training instances are searched for new instance so that most closely resemble new instance *(Just a retrieval program under the best circumstances)*
  - Lazy learning
- **Model based Learning**
  - Produce a generalized rules to learn a model from the data
  - The training data is not required once the model is trained

Parametric

- **Parametric Learning (parameters to tune)**
  - Make assumptions about mapping of input variables to output variables
  - The assumptions simplify the learning process (to a known form) but limits the learning

- Summarize data into a set of parameters of fixed size
- It has two steps
  - Select a form for the function
  - Learn the coefficients (weights) for the function from the training data
- Linear classification, Neural Networks, Naïve bayes

- **Non-parametric Learning (No parameters to tune)**
  - Make fewer or no assumptions about the target function and in turn require a lot of training data
  - Are free to learn any functional form from the data
  - Are slower to train and have high model complexity

- But can result in more powerful models
- Not concerned about choosing the right features
- Examples: KNN, Decision tree, SVM

Other classifiers: Naïve Bayes

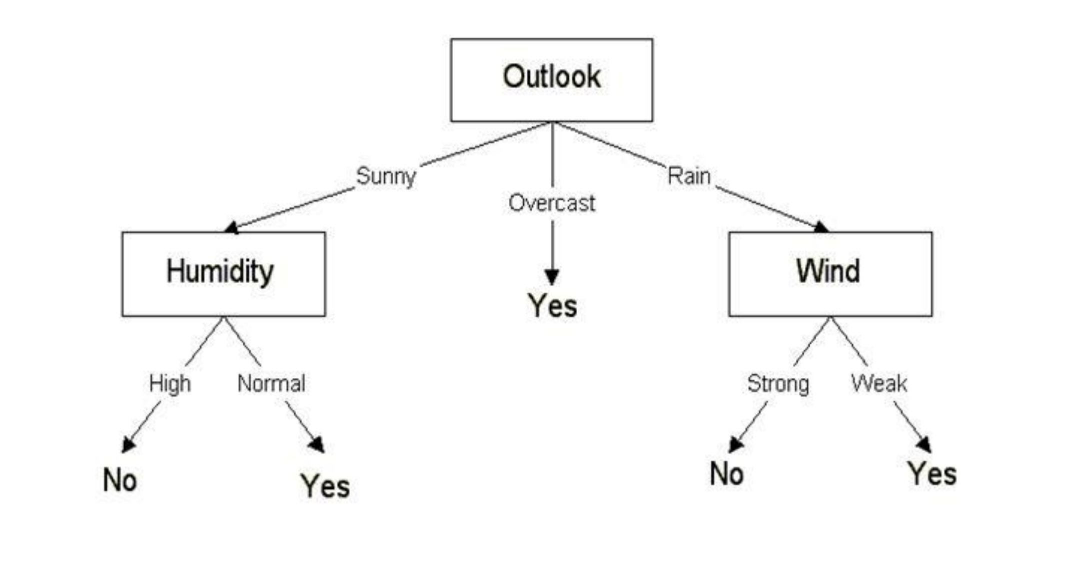
- Calculate probabilities for document labels and the words occurring in documents of those labels

- Each word indicate a document label with some probability

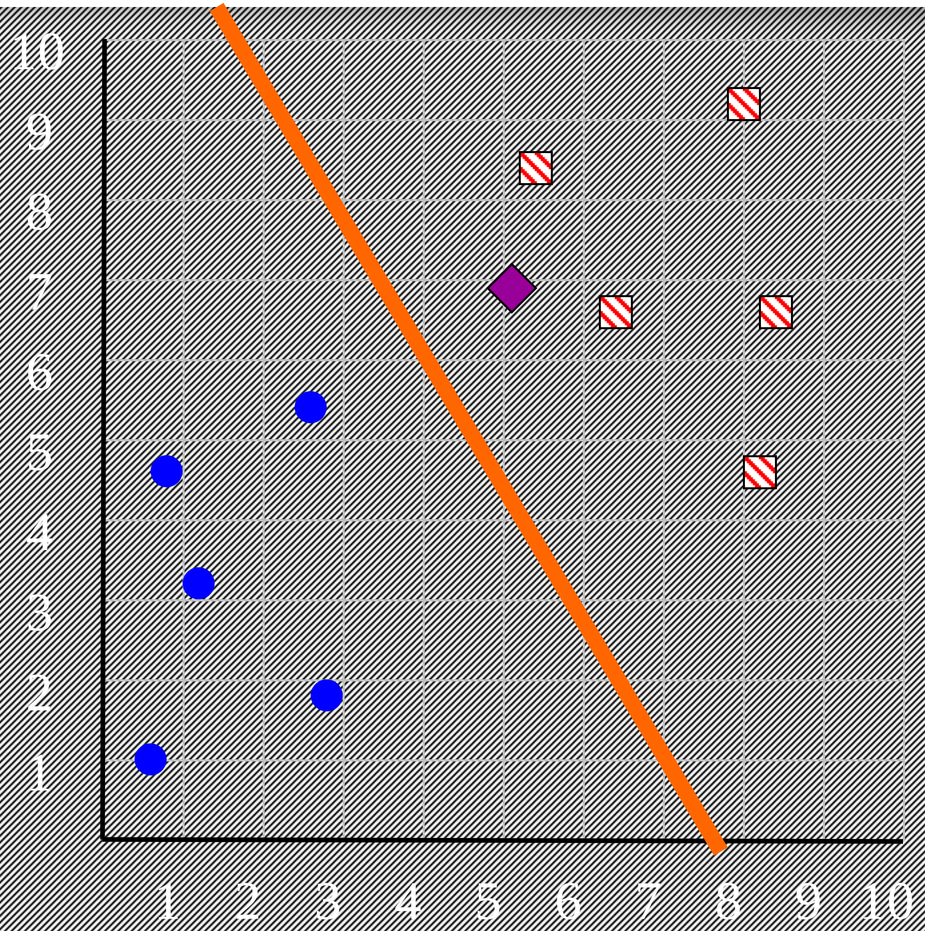
- The label probabilities of words in the test document are accumulated to predict a label for the test document

Decision Tree

- Input at root is a test document
- The intermediate nodes represent conditions on attributes
- The leaf nodes are the possible labels



Linear Classifier



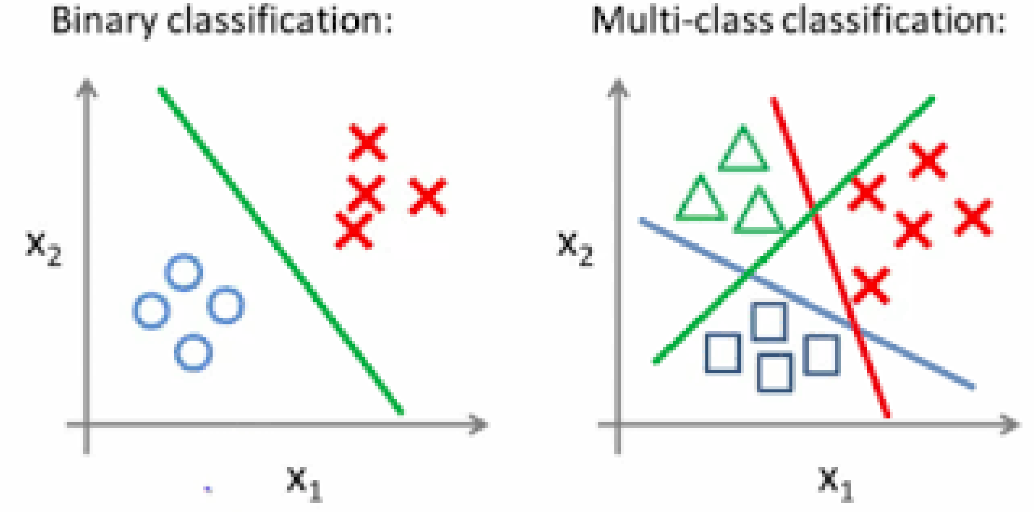
- Draw a straight line among the two types (label) of documents

- The test document is labeled based on which side of the line it lies.

- $w_0+w_1x_1 + w_2x_2 = b$

Multi-class Classification

- We may have to choose among multiple labels


Binary classification:
Multi-class classification:
x_2
x_1
x_2
x_1