# Information Retrieval

## Vector space classification

Hamid Beigy

Sharif university of technology

November 27, 2018

# Table of contents

# Vector space representation

1. Each document is a vector, one component for each term.
2. Terms are axes.
3. High dimensionality: 100,000s of dimensions
4. Normalize vectors (documents) to unit length
5. How can we do classification in this space?

# kNN classification

1. kNN $= k$ nearest neighbors
2. kNN classification rule for $k = 1$ (1NN): Assign each test document to the class of its nearest neighbor in the training set.
3. 1NN is not very robust – one document can be mislabeled or atypical.
4. kNN classification rule for $k > 1$ (kNN): Assign each test document to the majority class of its $k$ nearest neighbors in the training set.
5. Rationale of kNN: contiguity hypothesis
6. We expect a test document $d$ to have the same label as the training documents located in the local region surrounding $d$.

# Table of contents

# Linear classifiers

1. A linear classifier classifies documents as

**Definition (Linear classifier)**

A linear classifier computes a linear combination or weighted sum $\sum_i w_i x_i$ of the feature values. Classification decision: $\sum_i w_i x_i > \theta$? where $\theta$ (the threshold) is a parameter.

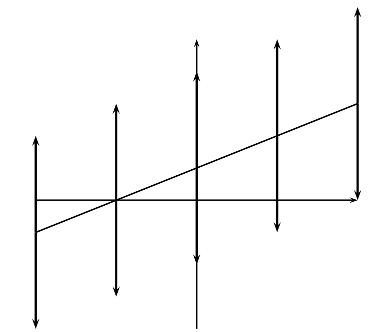
2. First, we only consider binary classifiers.
3. Geometrically, this corresponds to a line (2D), a plane (3D) or a hyperplane (higher dimensionality), the separator.
4. We find this separator based on training set.
5. Methods for finding separator: Perceptron, Rocchio, Naive Bayes – as we will explain on the next slides
6. Assumption: The classes are linearly separable.

# A linear classifier in 1D

1. A linear classifier in 1D is a point described by the equation $w_1 d_1 = \theta$
2. The point at $\theta / w_1$
3. Points $(d_1)$ with $w_1 d_1 \geq \theta$ are in the class $c$.
4. Points $(d_1)$ with $w_1 d_1 < \theta$ are in the complement class $\overline{c}$.

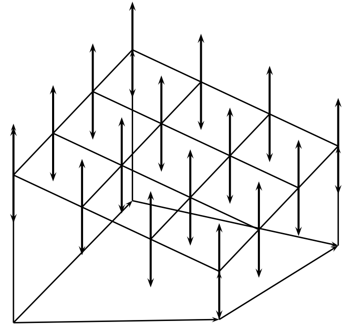
# A linear classifier in 2D

1. A linear classifier in 2D is a line described by the equation $w_1 d_1 + w_2 d_2 = \theta$
2. Example for a 2D linear classifier
3. Points $(d_1 \ d_2)$ with $w_1 d_1 + w_2 d_2 \geq \theta$ are in the class $c$.
4. Points $(d_1 \ d_2)$ with $w_1 d_1 + w_2 d_2 < \theta$ are in the complement class $\overline{c}$.



# A linear classifier in 3D

1. A linear classifier in 3D is a plane described by the equation $w_1 d_1 + w_2 d_2 + w_3 d_3 = \theta$
2. Example for a 3D linear classifier
3. Points $(d_1\ d_2\ d_3)$ with $w_1 d_1 + w_2 d_2 + w_3 d_3 \geq \theta$ are in the class $c$.
4. Points $(d_1\ d_2\ d_3)$ with $w_1 d_1 + w_2 d_2 + w_3 d_3 < \theta$ are in the complement class $\overline{c}$.



# Which classifier do I use for a given TC problem?

1. Is there a learning method that is optimal for all text classification problems?
2. No, because there is a tradeoff between bias and variance.
3. Factors to take into account:
   - How much training data is available?
   - How simple/complex is the problem? (linear vs. nonlinear decision boundary)
   - How noisy is the problem?
   - How stable is the problem over time?
   - For an unstable problem, it's better to use a simple and robust classifier.

# Table of contents

# Reading

Please read chapter 15 of Information Retrieval Book.