RSV of the Vector Space Model

- The matching function *RSV* is the cosine of the angle between the two vectors

$$RSV(d_i, q_i) = cos(\alpha) = \frac{\sum_{k=1}^{n} w_{ki} w_{kj}}{||d_i||_2 ||q_j||_2}$$

$$= \sum_{k=1}^{n} \frac{w_{ki} w_{kj}}{\sqrt{\sum_{k=1}^{n} w_{ki}^2} \sqrt{\sum_{k=1}^{n} w_{kj}^2}}$$

Note that

- Given two vectors (either docs or queries), their similarity is determined by their *direction* and *verse* but not their *distance*! You can think of them as lying on the sphere of unit radius
- The following three option returns the same value
  - Cosine of two generic vectors $\mathbf{x}_1$ and $\mathbf{x}_2$
  - Cosine of two vectors `v(`$\mathbf{x}_1$`)` and `v(`$\mathbf{x}_2$`)`, where `v(x)` is a length-normalized vector of `x`
  - The inner product of vectors `v(`$\mathbf{x}_1$`)` and `v(`$\mathbf{x}_2$`)`

Normalized vectors

- For normalized vectors, the cosine is simply the dot product:

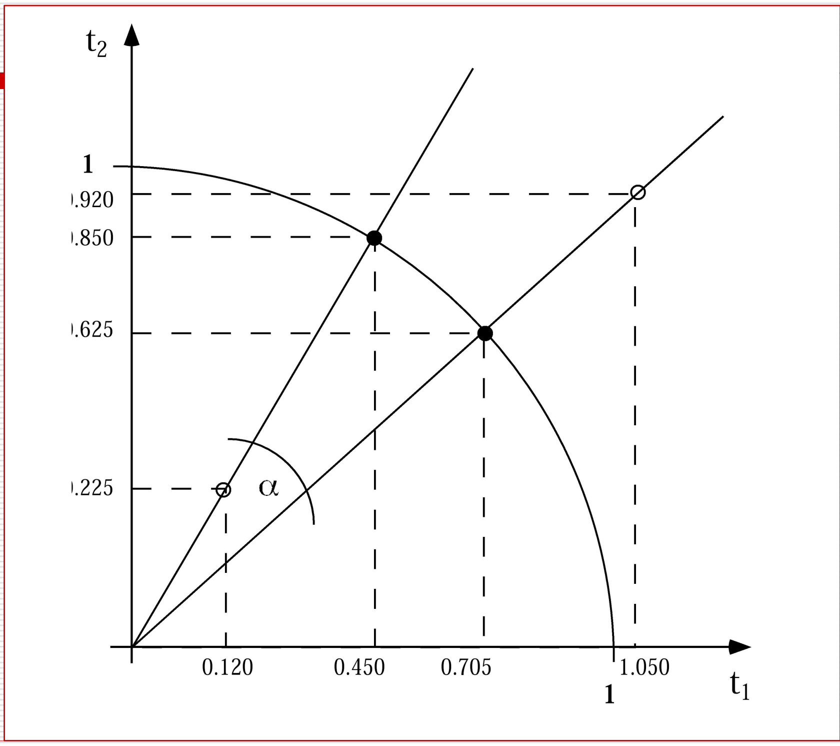
$$\cos(\vec{d}_j, \vec{d}_k) = \vec{d}_j \cdot \vec{d}_k$$

Exercise

- Euclidean distance between vectors:

$$\left| d_j - d_k \right| = \sqrt{\sum_{i=1}^{n} \left(d_{i,j} - d_{i,k} \right)^2}$$

- Show that, for normalized vectors, Euclidean distance gives the same proximity ordering as the cosine measure



The effect of normalization

- From the p.o.v. of ranking, the three previous options are also equivalent to computing the (inverse of) distance between $v(\mathbf{x}_1)$ and $v(\mathbf{x}_2)$
- Normalization implies that the RSV computes a *qualitative* similarity (what kind of information they contain) instead of *quantitative* (how much information they contain). It has experimentally be proven to yield improved results
- We can verify that in the VSM, ***RSV(d,d)=1*** (ideal document property). Boolean, Fuzzy Logic, and other models also have this property

Example

- Docs: Austen's *Sense and Sensibility*, *Pride and Prejudice*; Bronte's *Wuthering Heights*

	SaS	PaP	WH
affection	115	58	20
jealous	10	7	11
gossip	2	0	6

	SaS	PaP	WH
affection	0,996	0,993	0,847
jealous	0,087	0,120	0,466
gossip	0,017	0,000	0,254

cos(SAS, PAP) = .996 x .993 + .087 x .120 + .017 x 0.0 = 0.999
cos(SAS, WH) = .996 x .847 + .087 x .466 + .017 x .254 = 0.889

Advantages of the VSM

1. *Flexibility*. The *most decisive* factor in imposing VSM. The same intuitive geometric interpretation has been re-applied, apart from relevance feedback, in different contexts

   - Automatic document categorization
   - Automatic document filtering
   - Document clustering
   - Term-term similarity computation (terms are indexed by documents, dual)

2. Possibility to attribute weights to the IREPs of both documents and queries thus allowing *automatically indexed document bases*
3. Ranked output and output magnitude control
4. *Automatic query acquisition*
5. No output "flattening"; each query term contributes to the ranking in an equal way, depending on its weights
6. Much better effectiveness than the BM and the FLM

Disadvantages of the VSM

- Impossibility of formulating "structured" queries: there are no operators
  - Or (synonyms or quasi-synonyms)
  - And (compulsory occurrence of index terms)
  - Not (compulsory absence of index terms)
  - Prox (specification of noun phrases)
- The VSM is based on the hypothesis that the terms are pairwise stochastically independent (binary independence hypothesis). A more recent extension of the VSM relaxes this hypothesis, allowing the Cartesian axes to be non-orthogonal

On Similarity Measures

Any two arbitrary objects are equally similar unless we use domain knowledge!

- Inner Product
- Jaccard and Dice Similarity
- Overlap Coefficient
- Conversion from a distance
- Kernel Functions (beyond vectors)

	Binary case	Non-binary case
Inner Product	$\|d_i \cap q_j\|$	$d_i \cdot q_j$
Dice	$\frac{2\|d_i \cap q_j\|}{\|d_i\|+\|q_j\|}$	$\frac{2 d_i q_j}{\|\|d_i\|\|^2+\|\|q_j\|\|^2}$
Jaccard	$\frac{\|d_i \cap q_j\|}{\|d_i \cup q_j\|}$	$\frac{d_i q_j}{\|\|d_i\|\|^2+\|\|q_j\|\|^2 - d_i \cdot q_j}$
Overlap Coef.	$\frac{\|d_i \cap q_j\|}{min(\|d_i\|,\|q_j\|)}$	$\frac{d_i q_j}{min(\|\|d_i\|\|^2,\|\|q_j\|\|^2)}$

Conversion from a distance

Minkowsky Distances

$$L_p(x,z) = \left(\sum_i^n |x_i - z_i|^p\right)^{\frac{1}{p}}$$

When p = ∞, L_∞ = $\max_i$(|x_i-z_i|)

A similarity measure taking values in [0,1] can always be defined as

$$s_{p,\lambda}(x,z) = e^{-\lambda L_p(x,z)}$$

Where $\lambda \in (0,+\infty)$ is a constant parameter

Kernel functions

A kernel function K(x,z) is a (generally non-linear) function which corresponds to an inner product in some expanded feature space,

i.e. $K(x,z) = \phi(x) \cdot \phi(z)$

Example: For 2-dimensional spaces $x=(x_1,x_2)$

$$K(x,z) = (1 + x \cdot y)^2$$

is a kernel where

$$\phi(x) = (1, x_1^2, \sqrt{2}x_1 x_2, x_2^2, \sqrt{2}x_1, \sqrt{2}x_2)$$

- Polynomial Kernels:

$$K(x,z) = (x \cdot z + u)^p$$

- RBF Kernels

$$K(x,z) = e^{-\lambda ||x-z||^2}$$

- Other Kernels

  Also, there are kernels for structured data: strings, sequences, trees, graphs, an so on

- Polynomial kernels allows to model features which are conjunctions (up to the order of the polynomial)
- Radial Basis Function kernels is equivalent to map into an infinite dimensional Hilbert space
- A string kernel allows to have features that are subsequences of words