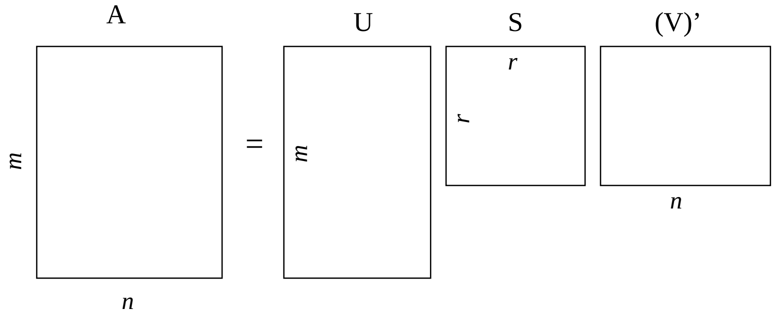
Latent Ambiguity in Latent Semantic Analysis?

Martin Emms and Alfredo Maldonado-Guerra

February 16 2013



Theorem (SVD)

*a $m \times n$ matrix **A** can be factorised as $\boldsymbol{A} = \boldsymbol{USV}'$ where:*

1. ***U** has the eigen-vectors of $\boldsymbol{A} \times \boldsymbol{A}'$ for its first r columns*
2. ***S**'s diagonal = square roots the eigen-values of **U***
3. ***V** has the eigen-vectors of $\boldsymbol{A}' \times \boldsymbol{A}$ for its first r columns*

Theorem (Low rank approximation)

If $\mathbf{U} \times \mathbf{S} \times \mathbf{V}'$ is the SVD of A, then $\hat{\mathbf{A}} = \mathbf{U}_k \times \mathbf{S}_k \times \mathbf{V}'_k$ is a optimum rank-k approx of $\mathbf{A}$ where

1. *$\mathbf{S}_k$ is diagonal with top-most k values from $\mathbf{S}$*
2. *$\mathbf{U}_k$ is just first k columns of $\mathbf{U}$*
3. *$\mathbf{V}_k$ is just first k columns of $\mathbf{V}$*

$\mathbf{U}_k \times \mathbf{S}_k \times \mathbf{V}'_k$ can be termed the 'rank k reduced SVD of $\mathbf{A}$'.

HCI/Graph example (from Deerwester et al. (1990))

two sets of article titles, one about HCI (titles c1–c5), the other about graph theory (titles m1–m4.

c1	*Human machine interface for ABC computer applications*
c2	*A survey of user opinion of computer system response time*
c3	*The EPS user interface management system*
c4	*System and human system engineering testing of EPS*
c5	*Relation of user perceived response time to error measurement*
m1	*The generation of random, binary, ordered trees*
m2	*The intersection graph of paths in trees*
m3	*Graph minors IV: Widths of trees and well-quasi-ordering*
m4	*graph minors:a survey*

HCI/Graph example (from Deerwester et al. (1990))

two sets of article titles, one about HCI (titles c1–c5), the other about graph theory (titles m1–m4.

gives $\boldsymbol{A}$ a 12×9 term-by-document matrix

	c1	c2	c3	c4	c5	m1	m2	m3	m4
human	1	0	0	1	0	0	0	0	0
interface	1	0	1	0	0	0	0	0	0
computer	1	1	0	0	0	0	0	0	0
user	0	1	1	0	1	0	0	0	0
system	0	1	1	2	0	0	0	0	0
respones	0	1	0	0	1	0	0	0	0
time	0	1	0	0	1	0	0	0	0
EPS	0	0	1	1	0	0	0	0	0
survey	0	1	0	0	0	0	0	0	1
trees	0	0	0	0	0	1	1	1	0
graph	0	0	0	0	0	0	1	1	1
minor	0	0	0	0	0	0	0	1	1

$A =$

	c1	c2	c3	c4	c5	m1	m2	m3	m4
human	1	0	0	1	0	0	0	0	0
interface	1	0	1	0	0	0	0	0	0
computer	1	1	0	0	0	0	0	0	0
user	0	1	1	0	1	0	0	0	0
system	0	1	1	2	0	0	0	0	0
respones	0	1	0	0	1	0	0	0	0
time	0	1	0	0	1	0	0	0	0
EPS	0	0	1	1	0	0	0	0	0
survey	0	1	0	0	0	0	0	0	1
trees	0	0	0	0	0	1	1	1	0
graph	0	0	0	0	0	0	1	1	1
minor	0	0	0	0	0	0	0	1	1

The rank-2 reduced SVD of $\boldsymbol{A}$ is $\boldsymbol{U}_2 \times \boldsymbol{S}_2 \times \boldsymbol{V}_2'$, where

$$\boldsymbol{U}_2 = \begin{bmatrix} 0.22 & -0.11 \\ 0.20 & -0.07 \\ 0.24 & 0.04 \\ 0.40 & 0.06 \\ 0.64 & -0.17 \\ 0.27 & 0.11 \\ 0.27 & 0.11 \\ 0.30 & -0.14 \\ 0.21 & 0.27 \\ 0.01 & 0.49 \\ 0.04 & 0.62 \\ 0.03 & 0.45 \end{bmatrix} \quad \boldsymbol{S}_2 = \begin{bmatrix} 3.34 & 0 \\ 0 & 2.54 \end{bmatrix} \quad \boldsymbol{V}_2 = \begin{bmatrix} 0.20 & -0.06 \\ 0.61 & 0.17 \\ 0.46 & -0.13 \\ 0.54 & -0.23 \\ 0.28 & 0.11 \\ 0.00 & 0.19 \\ 0.01 & 0.44 \\ 0.02 & 0.62 \\ 0.08 & 0.53 \end{bmatrix}$$

The rank-2 reduced SVD of $\boldsymbol{A}$ is $\boldsymbol{U}_2 \times \boldsymbol{S}_2 \times \boldsymbol{V}_2'$, where

$$\boldsymbol{U}_2 = \begin{bmatrix} 0.22 & -0.11 \\ 0.20 & -0.07 \\ 0.24 & 0.04 \\ 0.40 & 0.06 \\ 0.64 & -0.17 \\ 0.27 & 0.11 \\ 0.27 & 0.11 \\ 0.30 & -0.14 \\ 0.21 & 0.27 \\ 0.01 & 0.49 \\ 0.04 & 0.62 \\ 0.03 & 0.45 \end{bmatrix} \quad \boldsymbol{S}_2 = \begin{bmatrix} 3.34 & 0 \\ 0 & 2.54 \end{bmatrix} \quad \boldsymbol{V}_2 = \begin{bmatrix} 0.20 & -0.06 \\ 0.61 & 0.17 \\ 0.46 & -0.13 \\ 0.54 & -0.23 \\ 0.28 & 0.11 \\ 0.00 & 0.19 \\ 0.01 & 0.44 \\ 0.02 & 0.62 \\ 0.08 & 0.53 \end{bmatrix}$$

Note:

$\boldsymbol{U}_2$ is $|terms| \times 2$ ie. 12×2 $\quad$ $\boldsymbol{V}_2$ is $|docs| \times 2$ ie. 9×2

Latent Semantic Analysis (LSA) = using SVD to make lower dimension versions of document vectors

We claim literature has two *contendors for this SVD-based dimensionality reduction:*

R1

R2

Latent Semantic Analysis (LSA) = using SVD to make lower dimension versions of document vectors

We claim literature has two *contendors for this SVD-based dimensionality reduction:*

R1

R2

- 'native' i^{th} col of $\boldsymbol{A} \Rightarrow i^{th}$-th *row* of $\boldsymbol{V}_k$
  $\boldsymbol{V}_k^i$ is i^{th} row of $\boldsymbol{V}_k$ (ie. $[\boldsymbol{V}(i,1) \ldots \boldsymbol{V}(i,k)]$) = $\boldsymbol{V}_k^i$)

Latent Semantic Analysis (LSA) = using SVD to make lower dimension versions of document vectors

We claim literature has two *contendors for this SVD-based dimensionality reduction:*

R1

- 'native' i^{th} col of $\boldsymbol{A} \Rightarrow \boldsymbol{V}^i_k \times \boldsymbol{S}_k$

R2

- 'native' i^{th} col of $\boldsymbol{A} \Rightarrow i^{th}$-th *row* of $\boldsymbol{V}_k$
  $\boldsymbol{V}^i_k$ is i^{th} row of $\boldsymbol{V}_k$ (ie. $[\boldsymbol{V}(i,1)\ldots\boldsymbol{V}(i,k)]) = \boldsymbol{V}^i_k$)

Latent Semantic Analysis (LSA) = using SVD to make lower dimension versions of document vectors

We claim literature has two *contendors for this SVD-based dimensionality reduction:*

R1

- 'native' i^{th} col of $\boldsymbol{A} \Rightarrow \boldsymbol{V}^i_k \times \boldsymbol{S}_k$

R2

- arbitray m-dim doc vector $\boldsymbol{d} \Rightarrow \boldsymbol{d} \times \boldsymbol{U}_k \times \boldsymbol{S}^{-1}$
- 'native' i^{th} col of $\boldsymbol{A} \Rightarrow i^{th}$-th *row* of $\boldsymbol{V}_k$
  $\boldsymbol{V}^i_k$ is i^{th} row of $\boldsymbol{V}_k$ (ie. $[\boldsymbol{V}(i,1)\ldots\boldsymbol{V}(i,k)]$) = $\boldsymbol{V}^i_k$)

Latent Semantic Analysis (LSA) = using SVD to make lower dimension versions of document vectors

We claim literature has two *contendors for this SVD-based dimensionality reduction:*

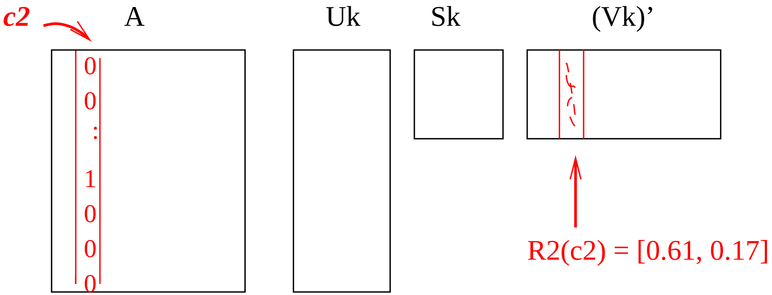
R1

- arbitray *m*-dim doc vector $\boldsymbol{d} \Rightarrow \boldsymbol{d} \times \boldsymbol{U}_k$
- 'native' i^{th} col of $\boldsymbol{A} \Rightarrow \boldsymbol{V}^i_k \times \boldsymbol{S}_k$

R2

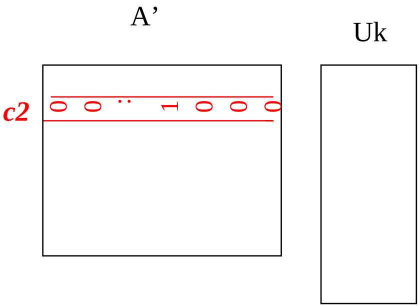
- arbitray *m*-dim doc vector $\boldsymbol{d} \Rightarrow \boldsymbol{d} \times \boldsymbol{U}_k \times \boldsymbol{S}^{-1}$
- 'native' i^{th} col of $\boldsymbol{A} \Rightarrow i^{th}$-th *row* of $\boldsymbol{V}_k$
  $\boldsymbol{V}^i_k$ is i^{th} row of $\boldsymbol{V}_k$ (ie. $[\boldsymbol{V}(i,1)\ldots\boldsymbol{V}(i,k)]) = \boldsymbol{V}^i_k$)

R_2 in pictures



according to R_2 image of c2 is corresponding col of $\boldsymbol{V}_k'$

R_1 in pictures



= [2.02,0.42] = R1(c2)

according to R_1 image of c2 via products with cols of $\boldsymbol{U}_k$

R_2 in Literature

R1

- arbitray: $R_1(\boldsymbol{d}) = \boldsymbol{d} \times \boldsymbol{U}_k$
- 'native': $R_1(\boldsymbol{d}) = \boldsymbol{V}^i_k \times \boldsymbol{S}_k$

R2

- arbitray: $R_2(\boldsymbol{d}) = \boldsymbol{d} \times \boldsymbol{U}_k \times \boldsymbol{S}^{-1}$
- 'native': $R_2(\boldsymbol{d}) = \boldsymbol{V}^i_k$

Gong and Liu (2001) have

> . . . projects each column vector i in matrix $\boldsymbol{A}$ . . . to column vector $[\boldsymbol{V}(i,1)\ldots\boldsymbol{V}(i,k)]'$ of matrix $\boldsymbol{V}'$

Zelikovitz and Hirsh (2001) have:

> . . . a query is represented ... by multiplying the transpose of the term vector of the query with matrices $\boldsymbol{U}$ and S^{-1}

. . . lots of others

R_1 in Literature

R1

- arbitray: $R_1(\mathbf{d}) = \mathbf{d} \times \mathbf{U}_k$
- 'native': $R_1(\mathbf{d}) = \mathbf{V}^i_k \times \mathbf{S}_k$

R2

- arbitray: $R_2(\mathbf{d}) = \mathbf{d} \times \mathbf{U}_k \times \mathbf{S}^{-1}$
- 'native': $R_2(\mathbf{d}) = \mathbf{V}^i_k$

Papadimitriou et al. (2000) have

> The rows of $\mathbf{V}_k\mathbf{S}_k$ above are then used to represent the documents

Kontostathis and Pottenger (2006) have

> Queries are represented in the reduced space by $\mathbf{q} \times \mathbf{U}_k$. . . . Queries are compared to the reduced document vectors . . . $\mathbf{V}_k \times \mathbf{S}_k$

. . . lots of others

R_1/R_2 relationship

R1

- arbitray: $R_1(\boldsymbol{d}) = \boldsymbol{d} \times \boldsymbol{U}_k$
- 'native': $R_1(\boldsymbol{d}) = \boldsymbol{V}_k^i \times \boldsymbol{S}_k$

R2

- arbitray: $R_2(\boldsymbol{d}) = \boldsymbol{d} \times \boldsymbol{U}_k \times \boldsymbol{S}^{-1}$
- 'native': $R_2(\boldsymbol{d}) = \boldsymbol{V}_k^i$

R_1/R_2 relationship

R1

- arbitray: $R_1(\boldsymbol{d}) = \boldsymbol{d} \times \boldsymbol{U}_k$
- 'native': $R_1(\boldsymbol{d}) = \boldsymbol{V}^i_k \times \boldsymbol{S}_k$

R2

- arbitray: $R_2(\boldsymbol{d}) = \boldsymbol{d} \times \boldsymbol{U}_k \times \boldsymbol{S}^{-1} = R_1(\boldsymbol{d}) \times \boldsymbol{S}^{-1}$
- 'native': $R_2(\boldsymbol{d}) = \boldsymbol{V}^i_k$

R_1/R_2 relationship

R1

- arbitray: $R_1(\boldsymbol{d}) = \boldsymbol{d} \times \boldsymbol{U}_k$
- 'native': $R_1(\boldsymbol{d}) = \boldsymbol{V}_k^i \times \boldsymbol{S}_k$

R2

- arbitray: $R_2(\boldsymbol{d}) = \boldsymbol{d} \times \boldsymbol{U}_k \times \boldsymbol{S}^{-1} = R_1(\boldsymbol{d}) \times \boldsymbol{S}^{-1}$
- 'native': $R_2(\boldsymbol{d}) = \boldsymbol{V}_k^i = R_1(\boldsymbol{d}) \times \boldsymbol{S}^{-1}$

R_1/R_2 relationship

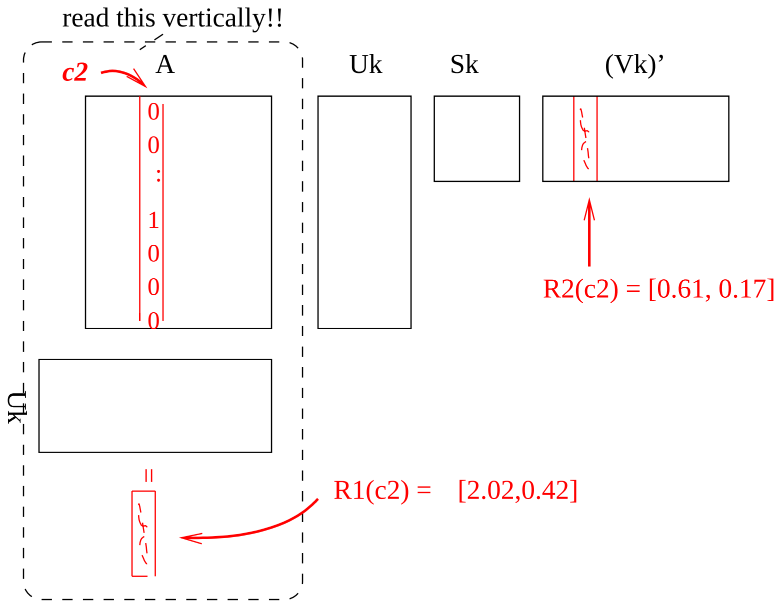
R1

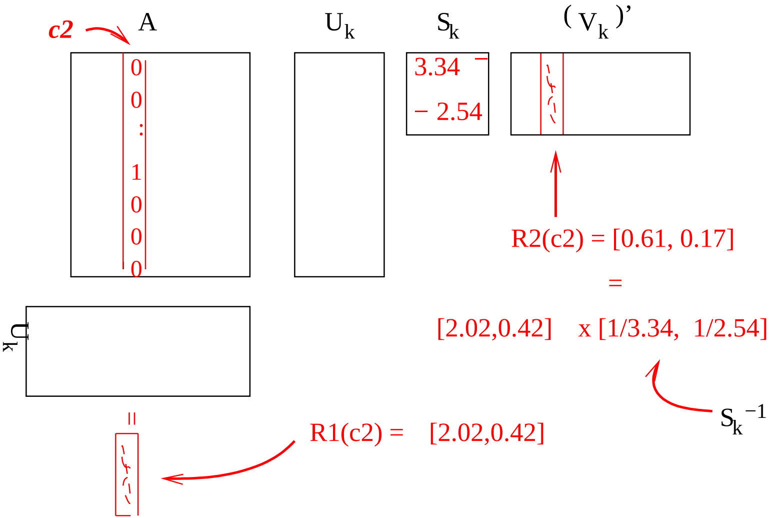
- arbitray: $R_1(\boldsymbol{d}) = \boldsymbol{d} \times \boldsymbol{U}_k$
- 'native': $R_1(\boldsymbol{d}) = \boldsymbol{V}_k^i \times \boldsymbol{S}_k$

R2

- arbitray: $R_2(\boldsymbol{d}) = \boldsymbol{d} \times \boldsymbol{U}_k \times \boldsymbol{S}^{-1} = R_1(\boldsymbol{d}) \times \boldsymbol{S}^{-1}$
- 'native': $R_2(\boldsymbol{d}) = \boldsymbol{V}_k^i = R_1(\boldsymbol{d}) \times \boldsymbol{S}^{-1}$

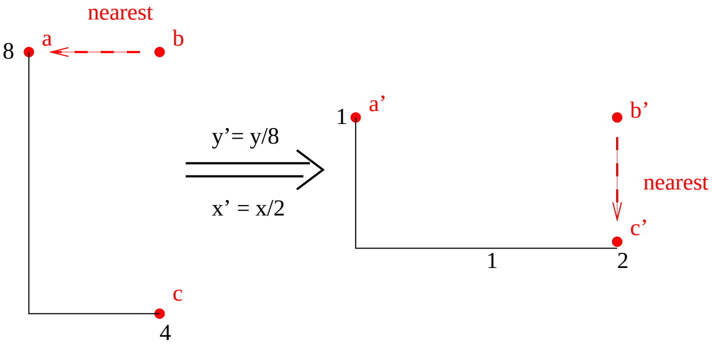
R_2 is a scaling of R_1


read this vertically!!
c2
A
Uk
Sk
(Vk)'
0
0
:
1
0
0
0
R2(c2) = [0.61, 0.17]
Uk
R1(c2) = [2.02,0.42]


c2
A
0
0
:
1
0
0
0
U k
S k
3.34
2.54
(V k)'
R2(c2) = [0.61, 0.17]
=
[2.02,0.42] x [1/3.34, 1/2.54]
S k −1
U k
=
R1(c2) = [2.02,0.42]

Scaling

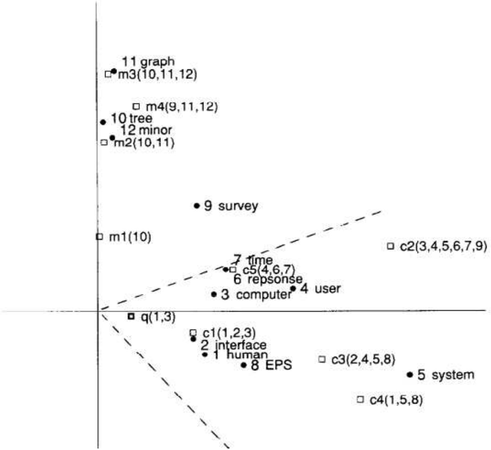
the relationship between the R_1 and R_2 is: $R_2(\boldsymbol{d}) = R_1(\boldsymbol{d}) \times \boldsymbol{S}^{-1}$. But as entries on diagonal are unequal this scaling changes the essential geometry, in particular the nearest neighbours



So should really expect R_1 and R_2 to give diverging outcomes in a system

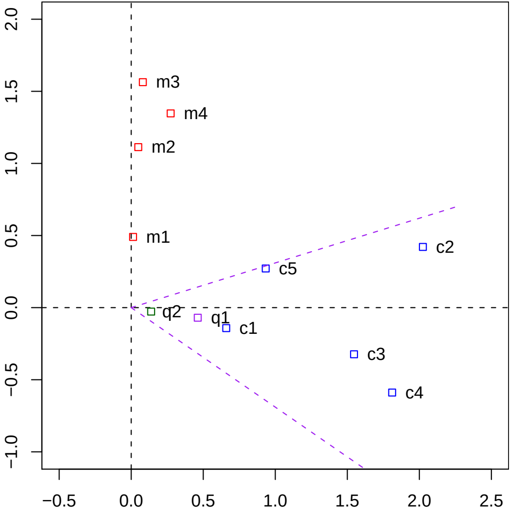
- on the basis of these works (and many others like them), there seems to be a R_1-vs-R_2 ambiguity in the formulation of LSA.
- what about in the earliest works on LSA ?

HCI/Graph docs in *R*? from Deerwester et al. (1990)



Deerwester et al. (1990) has plot of HCI/Graph docs in *R*? projection
also for $\boldsymbol{q} = [1,0,1,0,0,0,0,0,0,0,0,0]$ its plot in *R*?
but which ?

HCI/Graph docs in R_1



plot of docs in R_1

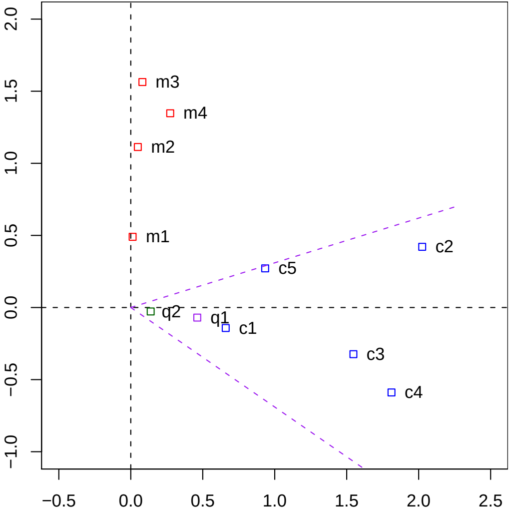
$\boldsymbol{q} = [1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0]$

$R_1(\boldsymbol{q}) = [0.46, -0.07] = q1$

$R_2(\boldsymbol{q}) = [0.14, -0.03] = q2$

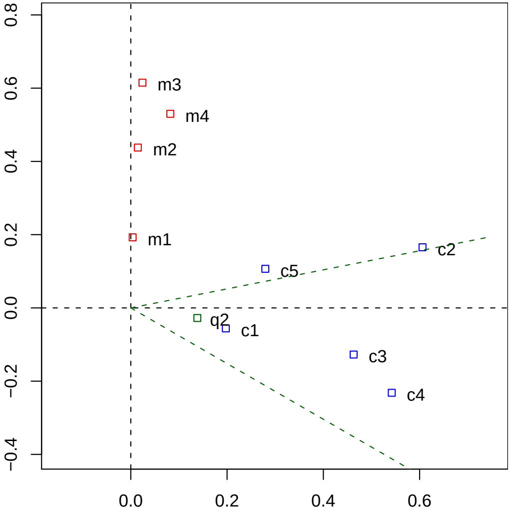
comparing to previous plot have to conclude that they have *documents* in R_1 projection *query* in the *the* R_2 *projection*

query cone in R_1



On the R_1 projection, the representations of c1–c5 are all included in the cone around the query.

query cone in R_1



on the R_2 projection the representations of c5 and c2 are *not* included.
note non-uniform shrinkage relative to R_1
first dimensinos shrinks by 0.29
second dimension shrinks by 0.39

Clustering expts

Consider *occurrences* of an ambiguous word, and the words in a *context* window of (+/- 10 words to left and right:

[... interest ...]

[*mortage* .. interest *..rise*]

[... *bank* interest .. *rate* ...]

[... interest ...]

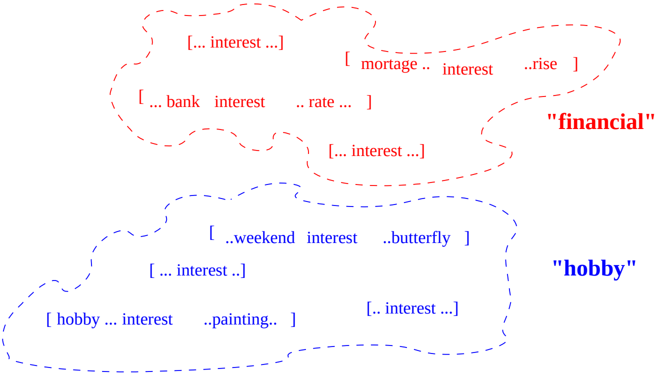
[*..weekend* interest *..butterfly*]

[... interest ..]

[*hobby* ... interest *..painting..*]

[.. interest ...]

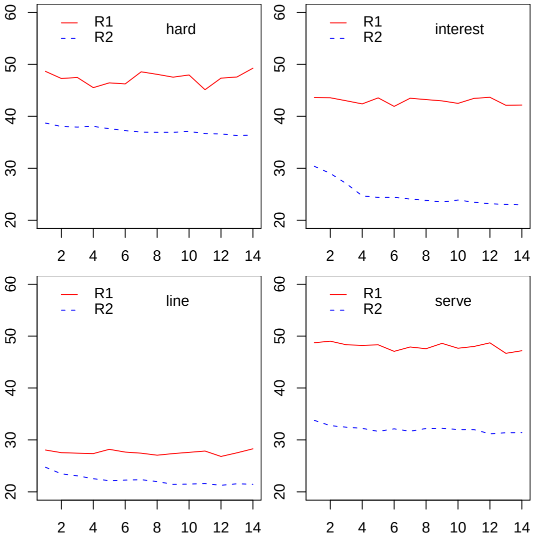
hunch: that if cluster these context windows as vectors the clusters will reflect different *senses* of the word:



these context vectors are **high** dimensionality: $\approx 10^4$
so apply SVD-based dimensionality reduction

- Do R_1 and R_2 work differently ?
- Is one consistently better ?

Unsupervised clustering results using R_1 and R_2



- vertical axis is accuracy
- horizontal axis is % reduction of dimensions
- R_1 and R_2 outcomes consistently different

Conclusions

- R_1 and R_2 give different geometries to the space of reduced representations, ie. different nearest-neighbour sets implying should expect different system outcomes
- However some researchers give the name 'LSA' to R_1 and some give the same 'LSA' to R_2
- One a couple of expts we found R_1 better, but arguably people should test both R_1 and R_2

S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman. Indexing by latent semantic analysis. *JOURNAL OF THE AMERICAN SOCIETY FOR INFORMATION SCIENCE*, 41(6):391–407, 1990.

Y. Gong and X. Liu. Generic text summarization using relevance measure and latent semantic analysis. In *SIGIR*, pages 19–25, 2001.

A. Kontostathis and W. M. Pottenger. A framework for understanding latent semantic indexing (lsi) performance. *INFORMATION PROCESSING AND MANAGEMENT*, 42(1):56–73, 2006.

C. H. Papadimitriou, P. Raghavan, H. Tamaki, and S. Vempala. Latent semantic indexing: A probabilistic analysis. *J. Comput. Syst. Sci.*, 61(2): 217–235, 2000.

S. Zelikovitz and H. Hirsh. Using lsi for text classification in the presence of background text. In *Proceedings of CIKM-01, 10TH ACM International Conference on information and knowledge management*, pages 113–118. ACM Press, 2001.