Making Sense of Distributional Semantic Models

Stefan Evert[1]

based on joint work with Marco Baroni[2] and Alessandro Lenci[3]

[1]University of Osnabrück, Germany
[2]University of Trento, Italy
[3]University of Pisa, Italy

Amsterdam, 22 Sep 2010

Outline

Introduction

The distributional hypothesis
Three famous DSM examples

Taxonomy of DSM parameters

Definition of DSM & parameter overview
Examples

Usage and evaluation of DSM

Using & interpreting DSM distances
Evaluation: attributional similarity

Singular Value Decomposition

Which distance measure?
Dimensionality reduction and SVD

Discussion

Outline

Introduction

- The distributional hypothesis
- Three famous DSM examples

Taxonomy of DSM parameters

- Definition of DSM & parameter overview
- Examples

Usage and evaluation of DSM

- Using & interpreting DSM distances
- Evaluation: attributional similarity

Singular Value Decomposition

- Which distance measure?
- Dimensionality reduction and SVD

Discussion

Meaning & distribution

- “Die Bedeutung eines Wortes ist sein Gebrauch in der Sprache.”

  — Ludwig Wittgenstein

- “You shall know a word by the company it keeps!”

  — J. R. Firth (1957)

- Distributional hypothesis (Zellig Harris 1954)

What is the meaning of “**bardiwac**”?

What is the meaning of "**bardiwac**"?

- He handed her her glass of bardiwac.

What is the meaning of "**bardiwac**"?

- He handed her her glass of bardiwac.
- Beef dishes are made to complement the bardiwacs.

What is the meaning of "**bardiwac**"?

- He handed her her glass of bardiwac.
- Beef dishes are made to complement the bardiwacs.
- Nigel staggered to his feet, face flushed from too much bardiwac.

What is the meaning of “**bardiwac**”?

- He handed her her glass of bardiwac.
- Beef dishes are made to complement the bardiwacs.
- Nigel staggered to his feet, face flushed from too much bardiwac.
- Malbec, one of the lesser-known bardiwac grapes, responds well to Australia’s sunshine.

What is the meaning of "**bardiwac**"?

- He handed her her glass of bardiwac.
- Beef dishes are made to complement the bardiwacs.
- Nigel staggered to his feet, face flushed from too much bardiwac.
- Malbec, one of the lesser-known bardiwac grapes, responds well to Australia's sunshine.
- I dined off bread and cheese and this excellent bardiwac.

What is the meaning of “**bardiwac**”?

- He handed her her glass of bardiwac.
- Beef dishes are made to complement the bardiwacs.
- Nigel staggered to his feet, face flushed from too much bardiwac.
- Malbec, one of the lesser-known bardiwac grapes, responds well to Australia’s sunshine.
- I dined off bread and cheese and this excellent bardiwac.
- The drinks were delicious: blood-red bardiwac as well as light, sweet Rhenish.

What is the meaning of "**bardiwac**"?

- He handed her her glass of bardiwac.
- Beef dishes are made to complement the bardiwacs.
- Nigel staggered to his feet, face flushed from too much bardiwac.
- Malbec, one of the lesser-known bardiwac grapes, responds well to Australia's sunshine.
- I dined off bread and cheese and this excellent bardiwac.
- The drinks were delicious: blood-red bardiwac as well as light, sweet Rhenish.

☞ bardiwac is a heavy red alcoholic beverage made from grapes

The examples above are handpicked, of course. But in a corpus like the BNC, you will find at least as many informative sentences.

A thought experiment: deciphering hieroglyphs

(knife)	51	20	84	0	3	0
(cat)	52	58	4	4	6	26
???	115	83	10	42	33	17
(boat)	59	39	23	4	0	0
(cup)	98	14	6	2	1	0
(pig)	12	17	3	2	9	27
(banana)	11	2	2	0	18	0

A thought experiment: deciphering hieroglyphs

(knife)		51	20	84	0	3	0
(cat)		52	58	4	4	6	26
???		115	83	10	42	33	17
(boat)		59	39	23	4	0	0
(cup)		98	14	6	2	1	0
(pig)		12	17	3	2	9	27
(banana)		11	2	2	0	18	0

sim(,) = 0.770

A thought experiment: deciphering hieroglyphs

(knife)		51	20	84	0	3	0
(cat)		52	58	4	4	6	26
???		115	83	10	42	33	17
(boat)		59	39	23	4	0	0
(cup)		98	14	6	2	1	0
(pig)		12	17	3	2	9	27
(banana)		11	2	2	0	18	0

sim(,) = 0.939

A thought experiment: deciphering hieroglyphs

(knife)	51	20	84	0	3	0
(cat)	52	58	4	4	6	26
???	115	83	10	42	33	17
(boat)	59	39	23	4	0	0
(cup)	98	14	6	2	1	0
(pig)	12	17	3	2	9	27
(banana)	11	2	2	0	18	0

sim(,) = 0.961

English as seen by the computer . . .

	get 𓏌𓁹𓏏	see 𓋴𓇋𓉐	use 𓇋𓋹𓋴	hear 𓉔𓇋𓂋	eat 𓇋𓇋𓏏	kill 𓎡𓇋𓃭
knife 𓈖𓄿𓇋𓆑	51	20	84	0	3	0
cat 𓎡𓁹𓏏	52	58	4	4	6	26
dog 𓂧𓍯𓏌	115	83	10	42	33	17
boat 𓃀𓁹𓋹𓏏	59	39	23	4	0	0
cup 𓎡𓄿𓊪	98	14	6	2	1	0
pig 𓊪𓇋𓏌𓇋𓉐	12	17	3	2	9	27
banana 𓈖𓄿𓈖𓄿	11	2	2	0	18	0

verb-object counts from British National Corpus

Geometric interpretation

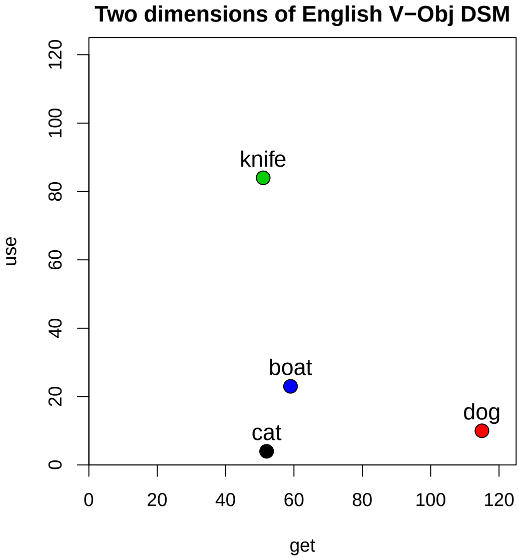
- row vector $\mathbf{x}_{\text{dog}}$ describes usage of word *dog* in the corpus
- can be seen as coordinates of point in n-dimensional Euclidean space $\mathbb{R}^n$

	get	see	use	hear	eat	kill
knife	51	20	84	0	3	0
cat	52	58	4	4	6	26
dog	115	83	10	42	33	17
boat	59	39	23	4	0	0
cup	98	14	6	2	1	0
pig	12	17	3	2	9	27
banana	11	2	2	0	18	0

co-occurrence matrix $\mathbf{M}$

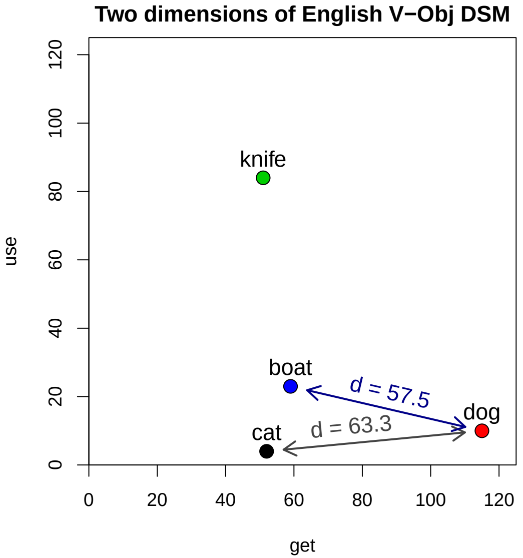
Geometric interpretation

- row vector $\mathbf{x}_{\text{dog}}$ describes usage of word *dog* in the corpus
- can be seen as coordinates of point in *n*-dimensional Euclidean space $\mathbb{R}^n$
- illustrated for two dimensions: *get* and *use*
- $\mathbf{x}_{\text{dog}} = (115, 10)$



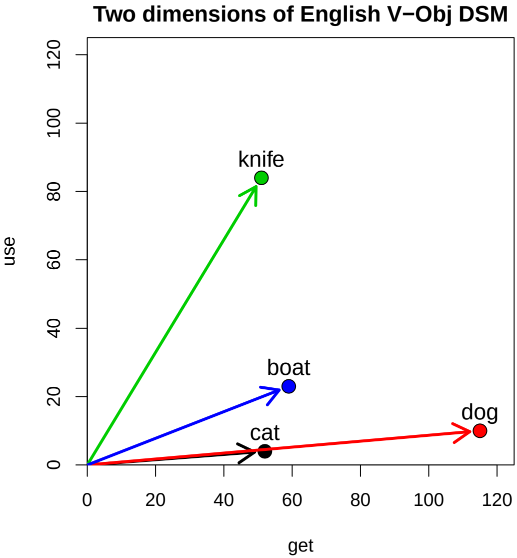
Geometric interpretation

- similarity = spatial proximity (Euclidean dist.)
- location depends on frequency of noun ($f_{\text{dog}} \approx 2.7 \cdot f_{\text{cat}}$)



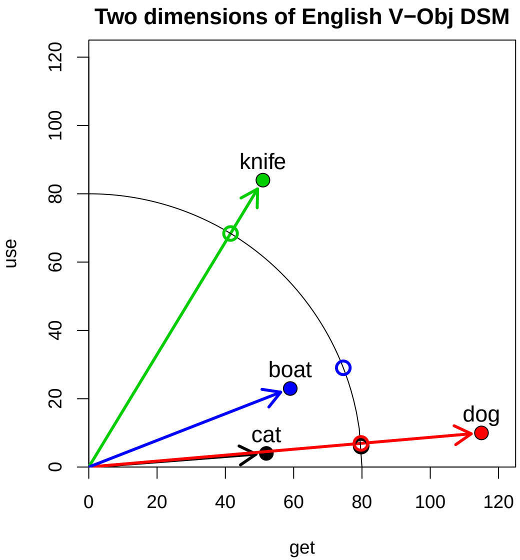
Geometric interpretation

- similarity = spatial proximity (Euclidean dist.)
- location depends on frequency of noun ($f_{\text{dog}} \approx 2.7 \cdot f_{\text{cat}}$)
- direction more important than location



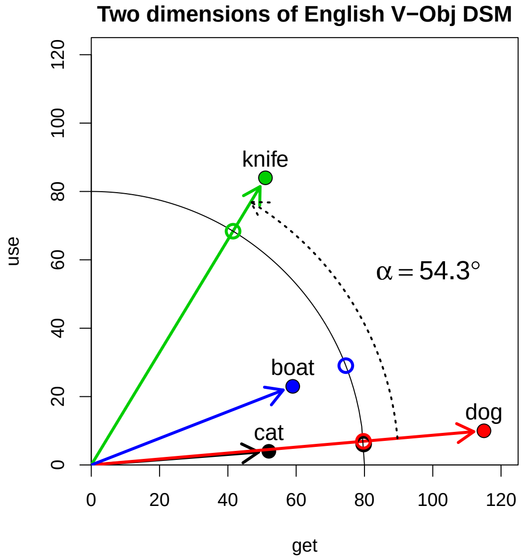
Geometric interpretation

- similarity = spatial proximity (Euclidean dist.)
- location depends on frequency of noun ($f_{\text{dog}} \approx 2.7 \cdot f_{\text{cat}}$)
- direction more important than location
- normalise "length" $\|\mathbf{x}_{\text{dog}}\|$ of vector



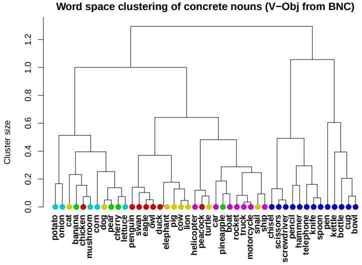
Geometric interpretation

- similarity = spatial proximity (Euclidean dist.)
- location depends on frequency of noun ($f_{\text{dog}} \approx 2.7 \cdot f_{\text{cat}}$)
- direction more important than location
- normalise "length" $\|\mathbf{x}_{\text{dog}}\|$ of vector
- or use angle α as distance measure

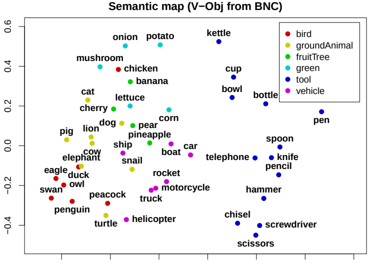


Semantic distances

- main result of distributional analysis are “semantic” distances between words
- typical applications
  - nearest neighbours
  - clustering of related words
  - construct semantic map



Word space clustering of concrete nouns (V−Obj from BNC)



Semantic map (V−Obj from BNC)

A very brief history of DSM

- Introduced to computational linguistics in early 1990s following the probabilistic revolution (Schütze 1992, 1998)
- Other early work in psychology (Landauer and Dumais 1997; Lund and Burgess 1996)
  - ☞ influenced by Latent Semantic Indexing (Dumais *et al.* 1988) and efficient software implementations (Berry 1992)

A very brief history of DSM

- Introduced to computational linguistics in early 1990s following the probabilistic revolution (Schütze 1992, 1998)
- Other early work in psychology (Landauer and Dumais 1997; Lund and Burgess 1996)
  - ☞ influenced by Latent Semantic Indexing (Dumais *et al.* 1988) and efficient software implementations (Berry 1992)
- Renewed interest in recent years
  - 2007: CoSMo Workshop (at Context '07)
  - 2008: ESSLLI Lexical Semantics Workshop & Shared Task, Special Issue of the Italian Journal of Linguistics
  - 2009: GeMS Workshop (EACL 2009), DiSCo Workshop (CogSci 2009), ESSLLI Advanced Course on DSM
  - 2010: 2nd GeMS Workshop (ACL 2010), ESSLLI Workhsop on Compositionality & DSM, Special Issue of JNLE (in prep.), Computational Neurolinguistics Workshop and DSM tutorial (NAACL-HLT 2010)

Some applications in computational linguistics

- Unsupervised part-of-speech induction (Schütze 1995)
- Word sense disambiguation (Schütze 1998)
- Query expansion in information retrieval (Grefenstette 1994)
- Synonym tasks & other language tests (Landauer and Dumais 1997; Turney *et al.* 2003)
- Thesaurus compilation (Lin 1998a; Rapp 2004)
- Ontology & wordnet expansion (Pantel *et al.* 2009)
- Attachment disambiguation (Pantel 2000)
- Probabilistic language models (Bengio *et al.* 2003)
- Subsymbolic input representation for neural networks
- Many other tasks in computational semantics: entailment detection, noun compound interpretation, identification of noncompositional expressions, . . .

Outline

Introduction

- The distributional hypothesis
- Three famous DSM examples

Taxonomy of DSM parameters

- Definition of DSM & parameter overview
- Examples

Usage and evaluation of DSM

- Using & interpreting DSM distances
- Evaluation: attributional similarity

Singular Value Decomposition

- Which distance measure?
- Dimensionality reduction and SVD

Discussion

Latent Semantic Analysis (Landauer and Dumais 1997)

- Corpus: 30,473 articles from Grolier's *Academic American Encyclopedia* (4.6 million words in total)
  - ☞ articles were limited to first 2,000 characters
- Word-article frequency matrix for 60,768 words
  - row vector shows frequency of word in each article
- Logarithmic frequencies scaled by word entropy
- Reduced to 300 dim. by singular value decomposition (SVD)
  - borrowed from LSI (Dumais *et al.* 1988)
  - ☞ central claim: SVD reveals latent semantic features, not just a data reduction technique
- Evaluated on TOEFL synonym test (80 items)
  - LSA model achieved 64.4% correct answers
  - also simulation of learning rate based on TOEFL results

Word Space (Schütze 1992, 1993, 1998)

- Corpus: ≈ 60 million words of news messages (*New York Times* News Service)
- Word-word co-occurrence matrix
  - 20,000 target words & 2,000 context words as features
  - row vector records how often each context word occurs close to the target word (co-occurrence)
  - co-occurrence window: left/right 50 words (Schütze 1998) or ≈ 1000 characters (Schütze 1992)
- Rows weighted by inverse document frequency (tf.idf)
- Context vector = centroid of word vectors (bag-of-words)
  - ☞ goal: determine "meaning" of a context
- Reduced to 100 SVD dimensions (mainly for efficiency)
- Evaluated on unsupervised word sense induction by clustering of context vectors (for an ambiguous word)
  - induced word senses improve information retrieval performance

HAL (Lund and Burgess 1996)

- HAL = Hyperspace Analogue to Language
- Corpus: 160 million words from newsgroup postings
- Word-word co-occurrence matrix
  - same 70,000 words used as targets and features
  - co-occurrence window of 1 – 10 words
- Separate counts for left and right co-occurrence
  - i.e. the context is *structured*
- In later work, co-occurrences are weighted by (inverse) distance (Li *et al.* 2000)
- Applications include construction of semantic vocabulary maps by multidimensional scaling to 2 dimensions

Many parameters . . .

- Enormous range of DSM parameters and applications
- Examples showed three entirely different models, each tuned to its particular application

➥ We need to . . .

. . . get an overview of available DSM parameters
. . . learn about the effects of parameter settings
. . . understand what aspects of meaning are encoded in DSM

Outline

Introduction
- The distributional hypothesis
- Three famous DSM examples

Taxonomy of DSM parameters
- Definition of DSM & parameter overview
- Examples

Usage and evaluation of DSM
- Using & interpreting DSM distances
- Evaluation: attributional similarity

Singular Value Decomposition
- Which distance measure?
- Dimensionality reduction and SVD

Discussion

General definition of DSMs

A **distributional semantic model** (DSM) is a scaled and/or transformed co-occurrence matrix **M**, such that each row **x** represents the distribution of a target term across contexts.

	get	see	use	hear	eat	kill
knife	0.027	-0.024	0.206	-0.022	-0.044	-0.042
cat	0.031	0.143	-0.243	-0.015	-0.009	0.131
dog	-0.026	0.021	-0.212	0.064	0.013	0.014
boat	-0.022	0.009	-0.044	-0.040	-0.074	-0.042
cup	-0.014	-0.173	-0.249	-0.099	-0.119	-0.042
pig	-0.069	0.094	-0.158	0.000	0.094	0.265
banana	0.047	-0.139	-0.104	-0.022	0.267	-0.042

Term = word form, lemma, phrase, morpheme, word pair, . . .

General definition of DSMs

Mathematical notation:

- $m \times n$ co-occurrence matrix $\mathbf{M}$ (example: 7×6 matrix)
  - m rows = target terms
  - n columns = features or **dimensions**

$$\mathbf{M} = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1n} \\ x_{21} & x_{22} & \cdots & x_{2n} \\ \vdots & \vdots & & \vdots \\ x_{m1} & x_{m2} & \cdots & x_{mn} \end{bmatrix}$$

- distribution vector $\mathbf{x}_i$ = i-th row of $\mathbf{M}$, e.g. $\mathbf{x}_3 = \mathbf{x}_{\text{dog}}$
- components $\mathbf{x}_i = (x_{i1}, x_{i2}, \ldots, x_{in})$ = features of i-th term:

$$\begin{aligned} \mathbf{x}_3 &= (-0.026, 0.021, -0.212, 0.064, 0.013, 0.014) \\ &= (x_{31}, x_{32}, x_{33}, x_{34}, x_{35}, x_{36}) \end{aligned}$$

Overview of DSM parameters

Linguistic pre-processing (annotation, definition of terms)

Overview of DSM parameters

Linguistic pre-processing (annotation, definition of terms)

⇓

Term-context vs. term-term matrix

Overview of DSM parameters

Linguistic pre-processing (annotation, definition of terms)

⇓

Term-context vs. term-term matrix

⇓

Size & type of context / structured vs. unstructered

Overview of DSM parameters

Linguistic pre-processing (annotation, definition of terms)

⇓

Term-context vs. term-term matrix

⇓

Size & type of context / structured vs. unstructered

⇓

Geometric vs. probabilistic interpretation

Overview of DSM parameters

Linguistic pre-processing (annotation, definition of terms)

⇓

Term-context vs. term-term matrix

⇓

Size & type of context / structured vs. unstructered

⇓

Geometric vs. probabilistic interpretation

⇓

Feature scaling

Overview of DSM parameters

Linguistic pre-processing (annotation, definition of terms)

⇓

Term-context vs. term-term matrix

⇓

Size & type of context / structured vs. unstructered

⇓

Geometric vs. probabilistic interpretation

⇓

Feature scaling

⇓

Similarity / distance measure & normalisation

Overview of DSM parameters

Linguistic pre-processing (annotation, definition of terms)

⇓

Term-context vs. term-term matrix

⇓

Size & type of context / structured vs. unstructered

⇓

Geometric vs. probabilistic interpretation

⇓

Feature scaling

⇓

Similarity / distance measure & normalisation

⇓

Dimensionality reduction

Overview of DSM parameters

Linguistic pre-processing (annotation, definition of terms)

⇓

Term-context vs. term-term matrix

⇓

Size & type of context / structured vs. unstructered

⇓

Geometric vs. probabilistic interpretation

⇓

Feature scaling

⇓

Similarity / distance measure & normalisation

⇓

Dimensionality reduction

Corpus pre-processing

- Linguistic analysis & annotation
  - minimally, corpus must be tokenised (➜ identify terms)
  - part-of-speech tagging
  - lemmatisation / stemming
  - word sense disambiguation (rare)
  - shallow syntactic patterns
  - dependency parsing

Corpus pre-processing

- Linguistic analysis & annotation
  - minimally, corpus must be tokenised (➜ identify terms)
  - part-of-speech tagging
  - lemmatisation / stemming
  - word sense disambiguation (rare)
  - shallow syntactic patterns
  - dependency parsing
- Generalisation of terms
  - often lemmatised to reduce data sparseness:
    go, goes, went, gone, going ➜ *go*
  - POS disambiguation (*light*/N vs. *light*/A vs. *light*/V)
  - word sense disambiguation (*bank*$_{\text{river}}$ vs. *bank*$_{\text{finance}}$)

Corpus pre-processing

- Linguistic analysis & annotation
  - minimally, corpus must be tokenised (➜ identify terms)
  - part-of-speech tagging
  - lemmatisation / stemming
  - word sense disambiguation (rare)
  - shallow syntactic patterns
  - dependency parsing
- Generalisation of terms
  - often lemmatised to reduce data sparseness: *go, goes, went, gone, going* ➜ *go*
  - POS disambiguation (*light*/N vs. *light*/A vs. *light*/V)
  - word sense disambiguation (*bank*$_{\text{river}}$ vs. *bank*$_{\text{finance}}$)
- Trade-off between deeper linguistic analysis and
  - need for language-specific resources
  - possible errors introduced at each stage of the analysis
  - even more parameters to optimise / cognitive plausibility

Effects of pre-processing

Nearest neighbours of *walk* (BNC)

word forms

- stroll
- walking
- walked
- go
- path
- drive
- ride
- wander
- sprinted
- sauntered

lemmatised corpus

- hurry
- stroll
- stride
- trudge
- amble
- wander
- walk-nn
- walking
- retrace
- scuttle

Effects of pre-processing

Nearest neighbours of *arrivare* (Repubblica)

word forms

- giungere
- raggiungere
- arrivi
- raggiungimento
- raggiunto
- trovare
- raggiunge
- arrivasse
- arriverà
- concludere

lemmatised corpus

- giungere
- aspettare
- attendere
- arrivo-nn
- ricevere
- accontentare
- approdare
- pervenire
- venire
- piombare

Overview of DSM parameters

Linguistic pre-processing (annotation, definition of terms)

⇓

Term-context vs. term-term matrix

⇓

Size & type of context / structured vs. unstructered

⇓

Geometric vs. probabilistic interpretation

⇓

Feature scaling

⇓

Similarity / distance measure & normalisation

⇓

Dimensionality reduction

Term-context vs. term-term matrix

Term-context matrix records frequency of term in each individual context (typically a sentence or document)

	doc_1	doc_2	doc_3	$\cdots$
boat	1	3	0	$\cdots$
cat	0	0	2	$\cdots$
dog	1	0	1	$\cdots$

- Appropriate contexts are non-overlapping textual units (Web page, encyclopaedia article, paragraph, sentence, . . .)

Term-context vs. term-term matrix

Term-context matrix records frequency of term in each individual context (typically a sentence or document)

	doc_1	doc_2	doc_3	$\cdots$
boat	1	3	0	$\cdots$
cat	0	0	2	$\cdots$
dog	1	0	1	$\cdots$

- Appropriate contexts are non-overlapping textual units (Web page, encyclopaedia article, paragraph, sentence, . . .)
- Can also be generalised to **context types**, e.g.
  - bag of content words
  - specific pattern of POS tags
  - subcategorisation pattern of target verb
- Term-context matrix is usually very **sparse**

Term-context vs. term-term matrix

Term-term matrix records co-occurrence frequencies of context terms for each target term (often target terms ≠ context terms)

	see	use	hear	⋯
boat	39	23	4	⋯
cat	58	4	4	⋯
dog	83	10	42	⋯

Term-context vs. term-term matrix

Term-term matrix records co-occurrence frequencies of context terms for each target term (often target terms $\neq$ context terms)

	see	use	hear	···
boat	39	23	4	···
cat	58	4	4	···
dog	83	10	42	···

- Different types of contexts (Evert 2008)
  - **surface context** (word or character window)
  - **textual context** (non-overlapping segments)
  - **syntactic contxt** (specific syntagmatic relation)
- Can be seen as smoothing of term-context matrix
  - average over similar contexts (with same context terms)
  - data sparseness reduced, except for small windows

Overview of DSM parameters

Linguistic pre-processing (annotation, definition of terms)

⇓

Term-context vs. term-term matrix

⇓

Size & type of context / structured vs. unstructered

⇓

Geometric vs. probabilistic interpretation

⇓

Feature scaling

⇓

Similarity / distance measure & normalisation

⇓

Dimensionality reduction

Surface context

Context term occurs within a window of k words around target.

The silhouette of the sun beyond a wide-open bay on the lake; the sun still glitters although evening has arrived in Kuhmo. It's midsummer; the living room has its instruments and other objects in each of its corners.

Parameters:

- window size (in words or characters)
- symmetric vs. one-sided window
- uniform or "triangular" (distance-based) weighting
- window clamped to sentences or other textual units?

Effect of different window sizes

Nearest neighbours of *dog* (BNC)

2-word window

- cat
- horse
- fox
- pet
- rabbit
- pig
- animal
- mongrel
- sheep
- pigeon

30-word window

- kennel
- puppy
- pet
- bitch
- terrier
- rottweiler
- canine
- cat
- to bark
- Alsatian

Textual context

Context term is in the same linguistic unit as target.

The silhouette of the sun beyond a wide-open bay on the lake; the sun still glitters although evening has arrived in Kuhmo. It's midsummer; the living room has its instruments and other objects in each of its corners.

Parameters:

- type of linguistic unit
  - sentence
  - paragraph
  - turn in a conversation
  - Web page

Syntactic context

Context term is linked to target by a syntactic dependency (e.g. subject, modifier, . . .).

The silhouette of the sun beyond a wide-open bay on the lake; the sun still glitters although evening has arrived in Kuhmo. It's midsummer; the living room has its instruments and other objects in each of its corners.

Parameters:

- types of syntactic dependency (Padó and Lapata 2007)
- direct vs. indirect dependency paths
- homogeneous data (e.g. only verb-object) vs. heterogeneous data (e.g. all children and parents of the verb)
- maximal length of dependency path

“Knowledge pattern” context

Context term is linked to target by a lexico-syntactic pattern (text mining, cf. Hearst 1992, Pantel & Pennacchiotti 2008, etc.).

In Provence, Van Gogh painted with bright colors such as red and yellow. These colors produce incredible effects on anybody looking at his paintings.

Parameters:

- inventory of lexical patterns
  - lots of research to identify semantically interesting patterns (cf. Almuhareb & Poesio 2004, Veale & Hao 2008, etc.)
- fixed vs. flexible patterns
  - patterns are mined from large corpora and automatically generalised (optional elements, POS tags or semantic classes)

Structured vs. unstructured context

- In **unstructered** models, context specification acts as a **filter**
  - determines whether context tokens counts as co-occurrence
  - e.g. linked by specific syntactic relation such as verb-object
- In **structured** models, context words are **subtyped**
  - depending on their position in the context
  - e.g. left vs. right context, type of syntactic relation, etc.

Structured vs. unstructured surface context

A dog bites a man. The man's dog bites a dog. A dog bites a man.

unstructured	bite
dog	4
man	3

A dog bites a man. The man's dog bites a dog. A dog bites a man.

structured	bite-l	bite-r
dog	3	1
man	1	2

Structured vs. unstructured dependency context

A dog bites a man. The man's dog bites a dog. A dog bites a man.

unstructured	bite
dog	4
man	2

A dog bites a man. The man's dog bites a dog. A dog bites a man.

structured	bite-subj	bite-obj
dog	3	1
man	0	2

Comparison

- Unstructured context
  - data less sparse (e.g. *man kills* and *kills man* both map to the *kill* dimension of the vector $\mathbf{x}_{\text{man}}$)
- Structured context
  - more sensitive to semantic distinctions (*kill-subj* and *kill-obj* are rather different things!)
  - dependency relations provide a form of syntactic "typing" of the DSM dimensions (the "subject" dimensions, the "recipient" dimensions, etc.)
  - important to account for word-order and compositionality

Overview of DSM parameters

Linguistic pre-processing (annotation, definition of terms)

$\Downarrow$

Term-context vs. term-term matrix

$\Downarrow$

Size & type of context / structured vs. unstructured

$\Downarrow$

Geometric vs. probabilistic interpretation

$\Downarrow$

Feature scaling

$\Downarrow$

Similarity / distance measure & normalisation

$\Downarrow$

Dimensionality reduction

Geometric vs. probabilistic interpretation

- Geometric interpretation
  - row vectors as points or arrows in n-dim. space
  - very intuitive, good for visualisation
  - use techniques from geometry and linear algebra
- Probabilistic interpretation
  - co-occurrence matrix as observed sample statistic
  - “explained” by generative probabilistic model
  - recent work focuses on hierarchical Bayesian models
  - probabilistic LSA (Hoffmann 1999), Latent Semantic Clustering (Rooth *et al.* 1999), Latent Dirichlet Allocation (Blei *et al.* 2003), etc.
  - explicitly accounts for random variation of frequency counts
  - intuitive and plausible as topic model

☞ focus exclusively on geometric interpretation in this talk

Overview of DSM parameters

Linguistic pre-processing (annotation, definition of terms)

⇓

Term-context vs. term-term matrix

⇓

Size & type of context / structured vs. unstructered

⇓

Geometric vs. probabilistic interpretation

⇓

Feature scaling

⇓

Similarity / distance measure & normalisation

⇓

Dimensionality reduction

Feature scaling

Feature scaling is used to compress wide magnitude range of frequency counts and to "discount" less informative features

- Logarithmic scaling: $x' = \log(x + 1)$
  (cf. Weber-Fechner law for human perception)
- Relevance weighting, e.g. tf.idf (information retrieval)

Feature scaling

Feature scaling is used to compress wide magnitude range of frequency counts and to "discount" less informative features

- Logarithmic scaling: $x' = \log(x + 1)$
  (cf. Weber-Fechner law for human perception)
- Relevance weighting, e.g. tf.idf (information retrieval)
- Statistical **association measures** (Evert 2004, 2008) take frequency of target word and context feature into account
  - the less frequent the target word and (more importantly) the context feature are, the higher the weight given to their observed co-occurrence count should be (because their expected chance co-occurrence frequency is low)
  - different measures – e.g., mutual information, log-likelihood ratio – differ in how they balance observed and expected co-occurrence frequencies

Association measures: Mutual Information (MI)

word$_1$	word$_2$	f_{obs}	f_1	f_2
dog	small	855	33,338	490,580
dog	domesticated	29	33,338	918

Association measures: Mutual Information (MI)

word$_1$	word$_2$	f_{obs}	f_1	f_2
dog	small	855	33,338	490,580
dog	domesticated	29	33,338	918

Expected co-occurrence frequency:

$$f_{\text{exp}} = \frac{f_1 \cdot f_2}{N}$$

Association measures: Mutual Information (MI)

word$_1$	word$_2$	f_{obs}	f_1	f_2
dog	small	855	33,338	490,580
dog	domesticated	29	33,338	918

Expected co-occurrence frequency:

$$f_{\text{exp}} = \frac{f_1 \cdot f_2}{N}$$

Mutual Information compares observed vs. expected frequency:

$$\text{MI}(w_1, w_2) = \log_2 \frac{f_{\text{obs}}}{f_{\text{exp}}} = \log_2 \frac{N \cdot f_{\text{obs}}}{f_1 \cdot f_2}$$

Association measures: Mutual Information (MI)

word$_1$	word$_2$	f_{obs}	f_1	f_2
dog	small	855	33,338	490,580
dog	domesticated	29	33,338	918

Expected co-occurrence frequency:

$$f_{\text{exp}} = \frac{f_1 \cdot f_2}{N}$$

Mutual Information compares observed vs. expected frequency:

$$\text{MI}(w_1, w_2) = \log_2 \frac{f_{\text{obs}}}{f_{\text{exp}}} = \log_2 \frac{N \cdot f_{\text{obs}}}{f_1 \cdot f_2}$$

Disadvantage: MI overrates combinations of rare terms.

Other association measures

Log-likelihood ratio (Dunning 1993) has more complex form, but its “core” is known as local MI (Evert 2004).

$$\text{local-MI}(w_1, w_2) = f_{\text{obs}} \cdot \text{MI}(w_1, w_2)$$

Other association measures

Log-likelihood ratio (Dunning 1993) has more complex form, but its “core” is known as local MI (Evert 2004).

$$\text{local-MI}(w_1, w_2) = f_{\text{obs}} \cdot \text{MI}(w_1, w_2)$$

word$_1$	word$_2$	f_{obs}	MI	local-MI
dog	small	855	3.96	3382.87
dog	domesticated	29	6.85	198.76
dog	sgjkj	1	10.31	10.31

Other association measures

Log-likelihood ratio (Dunning 1993) has more complex form, but its "core" is known as local MI (Evert 2004).

$$\text{local-MI}(w_1, w_2) = f_{\text{obs}} \cdot \text{MI}(w_1, w_2)$$

word$_1$	word$_2$	f_{obs}	MI	local-MI
dog	small	855	3.96	3382.87
dog	domesticated	29	6.85	198.76
dog	sgjkj	1	10.31	10.31

The t-score measure (Church and Hanks 1990) is popular in lexicography:

$$\text{t-score}(w_1, w_2) = \frac{f_{\text{obs}} - f_{\text{exp}}}{\sqrt{f_{\text{obs}}}}$$

Details & many more measures: http://www.collocations.de/

Overview of DSM parameters

Linguistic pre-processing (annotation, definition of terms)

⇓

Term-context vs. term-term matrix

⇓

Size & type of context / structured vs. unstructered

⇓

Geometric vs. probabilistic interpretation

⇓

Feature scaling

⇓

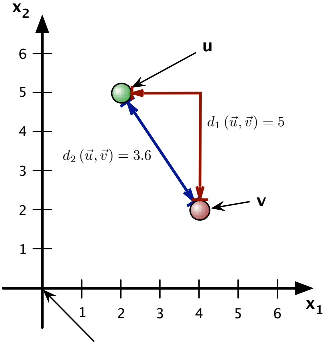
Similarity / distance measure & normalisation

⇓

Dimensionality reduction

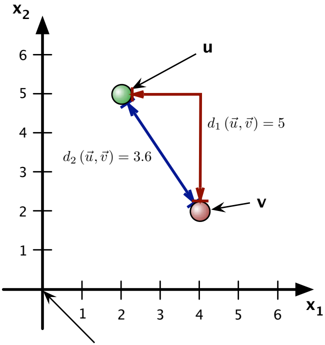
Geometric distance

- **Distance** between vectors $\mathbf{u}, \mathbf{v} \in \mathbb{R}^n$ ➜ (dis)**similarity**
    - $\mathbf{u} = (u_1, \ldots, u_n)$
    - $\mathbf{v} = (v_1, \ldots, v_n)$



Geometric distance

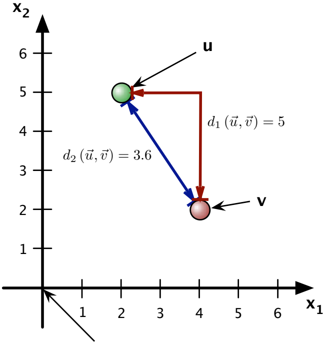
- **Distance** between vectors $\mathbf{u}, \mathbf{v} \in \mathbb{R}^n$ ➜ (dis)**similarity**
    - $\mathbf{u} = (u_1, \ldots, u_n)$
    - $\mathbf{v} = (v_1, \ldots, v_n)$
- **Euclidean** distance $d_2(\mathbf{u}, \mathbf{v})$



$$d_2(\mathbf{u}, \mathbf{v}) := \sqrt{(u_1 - v_1)^2 + \cdots + (u_n - v_n)^2}$$

Geometric distance

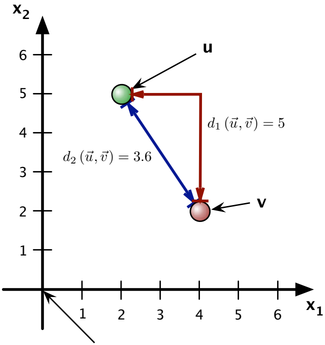
- **Distance** between vectors $\mathbf{u}, \mathbf{v} \in \mathbb{R}^n$ ➜ (dis)**similarity**
  - $\mathbf{u} = (u_1, \ldots, u_n)$
  - $\mathbf{v} = (v_1, \ldots, v_n)$
- **Euclidean** distance $d_2(\mathbf{u}, \mathbf{v})$
- "City block" **Manhattan** distance $d_1(\mathbf{u}, \mathbf{v})$



$$d_1(\mathbf{u}, \mathbf{v}) := |u_1 - v_1| + \cdots + |u_n - v_n|$$

Geometric distance

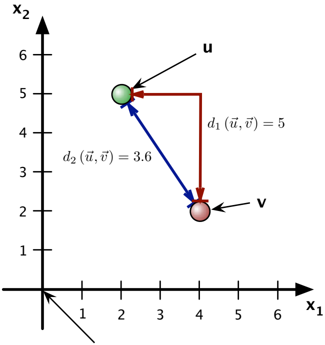
- **Distance** between vectors $\mathbf{u}, \mathbf{v} \in \mathbb{R}^n$ ➜ (dis)**similarity**
  - $\mathbf{u} = (u_1, \ldots, u_n)$
  - $\mathbf{v} = (v_1, \ldots, v_n)$
- **Euclidean** distance $d_2(\mathbf{u}, \mathbf{v})$
- "City block" **Manhattan** distance $d_1(\mathbf{u}, \mathbf{v})$
- Both are special cases of the **Minkowski** p-distance $d_p(\mathbf{u}, \mathbf{v})$ (for $p \in [1, \infty]$)



$$d_p(\mathbf{u}, \mathbf{v}) := \left(|u_1 - v_1|^p + \cdots + |u_n - v_n|^p\right)^{1/p}$$

Geometric distance

- **Distance** between vectors $\mathbf{u}, \mathbf{v} \in \mathbb{R}^n$ ➜ (dis)**similarity**
  - $\mathbf{u} = (u_1, \ldots, u_n)$
  - $\mathbf{v} = (v_1, \ldots, v_n)$
- **Euclidean** distance $d_2(\mathbf{u}, \mathbf{v})$
- "City block" **Manhattan** distance $d_1(\mathbf{u}, \mathbf{v})$
- Both are special cases of the **Minkowski** p-distance $d_p(\mathbf{u}, \mathbf{v})$ (for $p \in [1, \infty]$)



$$d_p(\mathbf{u}, \mathbf{v}) := \left(|u_1 - v_1|^p + \cdots + |u_n - v_n|^p\right)^{1/p}$$

$$d_\infty(\mathbf{u}, \mathbf{v}) = \max\left\{|u_1 - v_1|, \ldots, |u_n - v_n|\right\}$$

Other distance measures

- Information theory: **Kullback-Leibler** (KL) **divergence** for probability vectors (non-negative, $\|\mathbf{x}\|_1 = 1$)

$$D(\mathbf{u}\|\mathbf{v}) = \sum_{i=1}^{n} u_i \cdot \log_2 \frac{u_i}{v_i}$$

Other distance measures

- Information theory: **Kullback-Leibler** (KL) **divergence** for probability vectors (non-negative, $\|\mathbf{x}\|_1 = 1$)

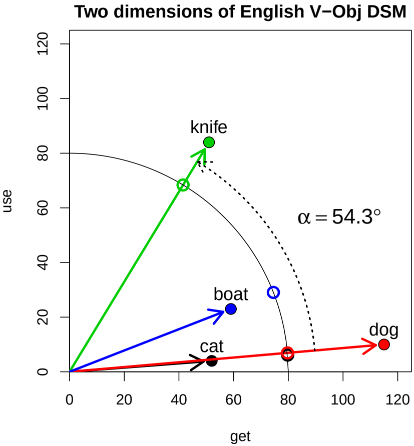
$$D(\mathbf{u}\|\mathbf{v}) = \sum_{i=1}^{n} u_i \cdot \log_2 \frac{u_i}{v_i}$$

- Properties of KL divergence
  - most appropriate in a probabilistic interpretation of **M**
  - not symmetric, unlike all other measures
  - alternatives: skew divergence, Jensen-Shannon divergence

Similarity measures

- angle α between two vectors $\mathbf{u}, \mathbf{v}$ is given by

$$\cos\alpha = \frac{\sum_{i=1}^{n} u_i \cdot v_i}{\sqrt{\sum_i u_i^2} \cdot \sqrt{\sum_i v_i^2}} = \frac{\langle \mathbf{u}, \mathbf{v} \rangle}{\|\mathbf{u}\|_2 \cdot \|\mathbf{v}\|_2}$$



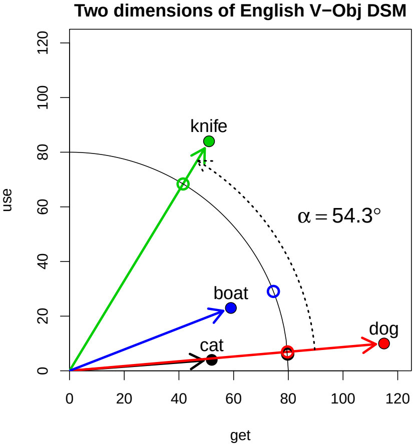
Similarity measures

- angle α between two vectors $\mathbf{u}, \mathbf{v}$ is given by

$$\cos\alpha = \frac{\sum_{i=1}^{n} u_i \cdot v_i}{\sqrt{\sum_i u_i^2} \cdot \sqrt{\sum_i v_i^2}}$$

$$= \frac{\langle \mathbf{u}, \mathbf{v} \rangle}{\|\mathbf{u}\|_2 \cdot \|\mathbf{v}\|_2}$$

- **cosine** measure of similarity: $\cos\alpha$
  - $\cos\alpha = 1$ ➜ collinear
  - $\cos\alpha = 0$ ➜ orthogonal

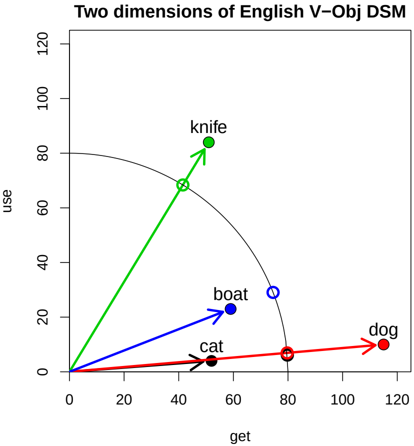


Normalisation of row vectors

- geometric distances only make sense if vectors are normalised to unit length
- divide vector by its length:

$$\mathbf{x}/\|\mathbf{x}\|$$

- normalisation depends on distance measure!
- special case: scale to relative frequencies with $\|\mathbf{x}\|_1 = |x_1| + \cdots + |x_n|$



Scaling of column vectors (standardisation)

- In statistical analysis and machine learning, features are usually centred and scaled so that

$$\begin{aligned} \text{mean} \quad & \mu = 0 \\ \text{variance} \quad & \sigma^2 = 1 \end{aligned}$$

- In DSM research, this step is less common for columns of **M**
  - centring is a prerequisite for certain dimensionality reduction and data analysis techniques (esp. PCA)
  - scaling may give too much weight to rare features

Scaling of column vectors (standardisation)

- In statistical analysis and machine learning, features are usually centred and scaled so that

$$\begin{aligned} \text{mean} \quad & \mu = 0 \\ \text{variance} \quad & \sigma^2 = 1 \end{aligned}$$

- In DSM research, this step is less common for columns of **M**
  - centring is a prerequisite for certain dimensionality reduction and data analysis techniques (esp. PCA)
  - scaling may give too much weight to rare features
- It does not make sense to combine column-standardisation with row-normalisation! (Do you see why?)
  - but variance scaling without centring may be applied

Overview of DSM parameters

Linguistic pre-processing (annotation, definition of terms)

⇓

Term-context vs. term-term matrix

⇓

Size & type of context / structured vs. unstructered

⇓

Geometric vs. probabilistic interpretation

⇓

Feature scaling

⇓

Similarity / distance measure & normalisation

⇓

Dimensionality reduction

Dimensionality reduction = data compression

- Co-occurrence matrix **M** is often unmanageably large and can be extremely sparse
  - Google Web1T5: 1M × 1M matrix with one trillion cells, of which less than 0.05% contain nonzero counts (Evert 2010)

➥ Compress matrix by reducing dimensionality (= columns)

Dimensionality reduction = data compression

- Co-occurrence matrix **M** is often unmanageably large and can be extremely sparse
  - Google Web1T5: 1M × 1M matrix with one trillion cells, of which less than 0.05% contain nonzero counts (Evert 2010)

➥ Compress matrix by reducing dimensionality (= columns)

- **Feature selection**: columns with high frequency & variance
  - measured by entropy, chi-squared test, . . .
  - may select correlated (➜ uninformative) dimensions
  - joint selection of multiple features is expensive

Dimensionality reduction = data compression

- Co-occurrence matrix **M** is often unmanageably large and can be extremely sparse
  - Google Web1T5: 1M × 1M matrix with one trillion cells, of which less than 0.05% contain nonzero counts (Evert 2010)

➥ Compress matrix by reducing dimensionality (= columns)

- **Feature selection**: columns with high frequency & variance
  - measured by entropy, chi-squared test, . . .
  - may select correlated (➜ uninformative) dimensions
  - joint selection of multiple features is expensive
- **Projection** into (linear) subspace
  - principal component analysis (PCA)
  - independent component analysis (ICA)
  - random indexing (RI)

  ☞ intuition: preserve distances between data points

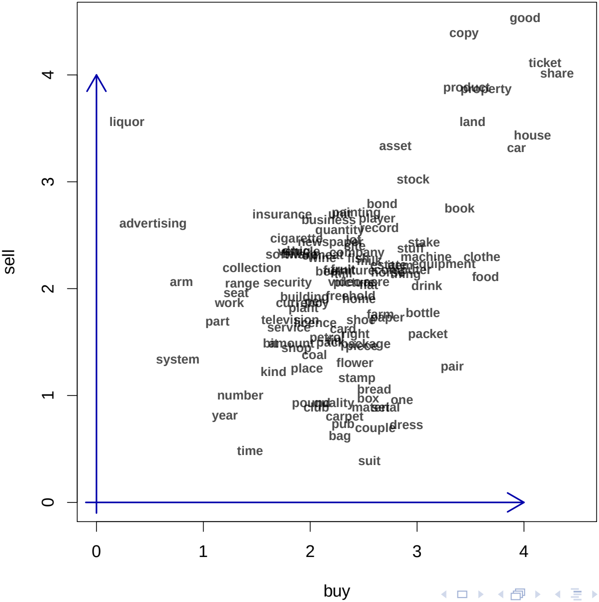
Dimensionality reduction & latent dimensions

Landauer and Dumais (1997) claim that LSA dimensionality reduction (and related PCA technique) uncovers **latent dimensions** by exploiting correlations between features.

- Example: term-term matrix
- V-Obj cooc's extracted from BNC
  - targets = noun lemmas
  - features = verb lemmas
- feature scaling: association scores (modified log Dice coefficient)
- $k = 111$ nouns with $f \geq 20$ (must have non-zero row vectors)
- $n = 2$ dimensions: *buy* and *sell*

noun	*buy*	*sell*
bond	0.28	0.77
cigarette	-0.52	0.44
dress	0.51	-1.30
freehold	-0.01	-0.08
land	1.13	1.54
number	-1.05	-1.02
per	-0.35	-0.16
pub	-0.08	-1.30
share	1.92	1.99
system	-1.63	-0.70

Dimensionality reduction & latent dimensions



Motivating latent dimensions & subspace projection

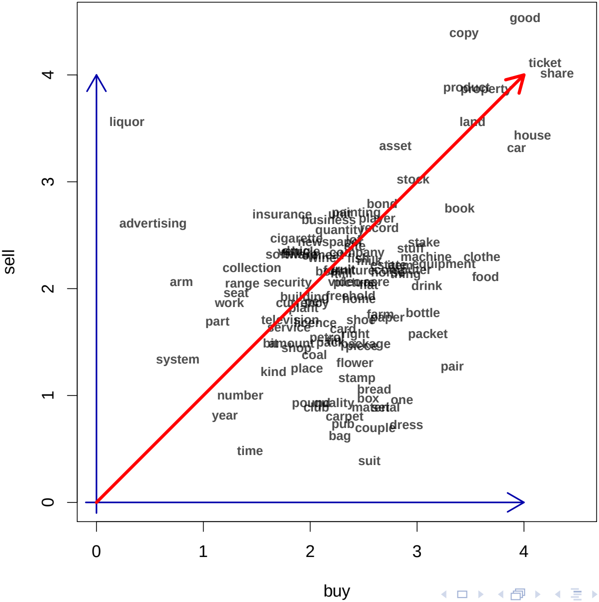
- The **latent property** of being a commodity is "expressed" through associations with several verbs: *sell*, *buy*, *acquire*, ...
- Consequence: these DSM dimensions will be **correlated**

Motivating latent dimensions & subspace projection

- The **latent property** of being a commodity is "expressed" through associations with several verbs: *sell*, *buy*, *acquire*, . . .
- Consequence: these DSM dimensions will be **correlated**

- Identify **latent dimension** by looking for strong correlations (or weaker correlations between large sets of features)
- Projection into subspace V of $k < n$ latent dimensions as a "**noise reduction**" technique ➜ **LSA**
- Assumptions of this approach:
  - "latent" distances in V are semantically meaningful
  - other "residual" dimensions represent chance co-occurrence patterns, often particular to the corpus underlying the DSM

The latent “commodity” dimension



Outline

Introduction

- The distributional hypothesis
- Three famous DSM examples

Taxonomy of DSM parameters

- Definition of DSM & parameter overview
- Examples

Usage and evaluation of DSM

- Using & interpreting DSM distances
- Evaluation: attributional similarity

Singular Value Decomposition

- Which distance measure?
- Dimensionality reduction and SVD

Discussion

Some well-known DSM examples

Latent Semantic Analysis (Landauer and Dumais 1997)

- term-context matrix with document context
- weighting: log term frequency and term entropy
- distance measure: cosine
- dimensionality reduction: SVD

Hyperspace Analogue to Language (Lund and Burgess 1996)

- term-term matrix with surface context
- structured (left/right) and distance-weighted frequency counts
- distance measure: Minkowski metric ($1 \leq p \leq 2$)
- dimensionality reduction: feature selection (high variance)

Some well-known DSM examples

Infomap NLP (Widdows 2004)

- term-term matrix with unstructured surface context
- weighting: none
- distance measure: cosine
- dimensionality reduction: SVD

Random Indexing (Karlgren & Sahlgren 2001)

- term-term matrix with unstructured surface context
- weighting: various methods
- distance measure: various methods
- dimensonality reduction: random indexing (RI)

Some well-known DSM examples

Dependency Vectors (Padó and Lapata 2007)

- term-term matrix with unstructured dependency context
- weighting: log-likelihood ratio
- distance measure: information-theoretic (Lin 1998b)
- dimensionality reduction: none

Distributional Memory (Baroni & Lenci 2009)

- both term-context and term-term matrices
- context: structured dependency context
- weighting: local-MI association measure
- distance measure: cosine
- dimensionality reduction: none

Outline

Introduction

The distributional hypothesis

Three famous DSM examples

Taxonomy of DSM parameters

Definition of DSM & parameter overview

Examples

Usage and evaluation of DSM

Using & interpreting DSM distances

Evaluation: attributional similarity

Singular Value Decomposition

Which distance measure?

Dimensionality reduction and SVD

Discussion

Nearest neighbours

DSM based on verb-object relations from BNC, reduced to 100 dim. with SVD

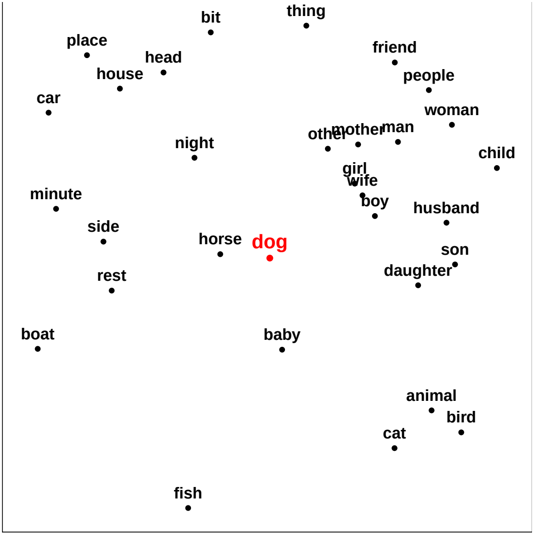
Neighbours of **dog** (cosine angle):

☞ girl (45.5), boy (46.7), horse(47.0), wife (48.8), baby (51.9), daughter (53.1), side (54.9), mother (55.6), boat (55.7), rest (56.3), night (56.7), cat (56.8), son (57.0), man (58.2), place (58.4), husband (58.5), thing (58.8), friend (59.6), . . .

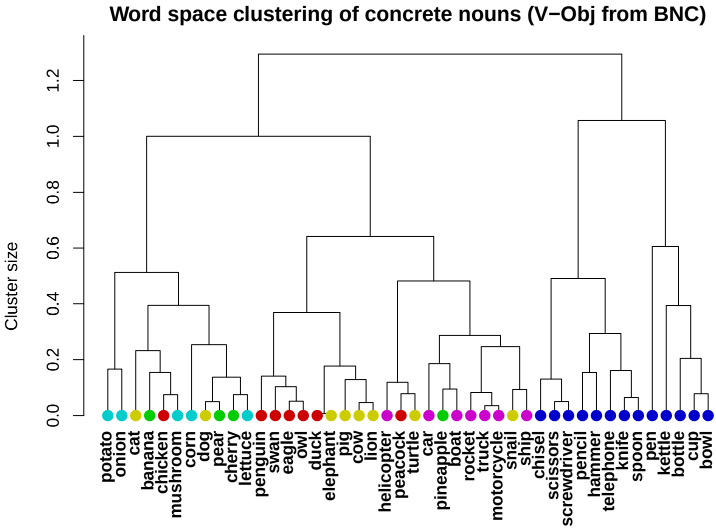
Neighbours of **school**:

☞ country (49.3), church (52.1), hospital (53.1), house (54.4), hotel (55.1), industry (57.0), company (57.0), home (57.7), family (58.4), university (59.0), party (59.4), group (59.5), building (59.8), market (60.3), bank (60.4), business (60.9), area (61.4), department (61.6), club (62.7), town (63.3), library (63.3), room (63.6), service (64.4), police (64.7), . . .

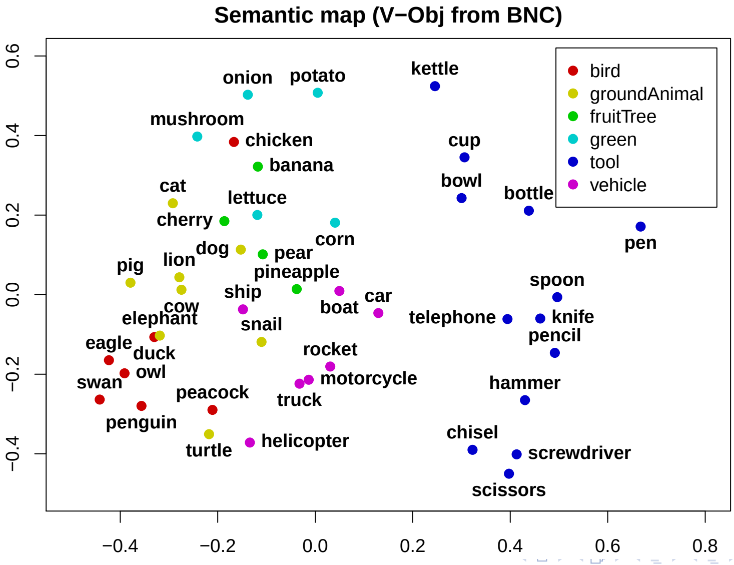
Nearest neighbours



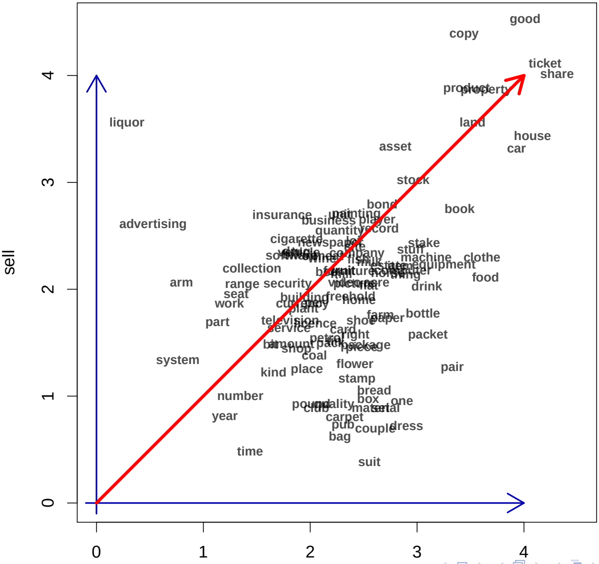
Clustering


Word space clustering of concrete nouns (V−Obj from BNC)
Cluster size
0.0
0.2
0.4
0.6
0.8
1.0
1.2
potato
onion
cat
banana
chicken
mushroom
corn
dog
pear
cherry
lettuce
penguin
swan
eagle
owl
duck
elephant
pig
cow
lion
helicopter
peacock
turtle
car
pineapple
boat
rocket
truck
motorcycle
snail
ship
chisel
scissors
screwdriver
pencil
hammer
telephone
knife
spoon
pen
kettle
bottle
cup
bowl

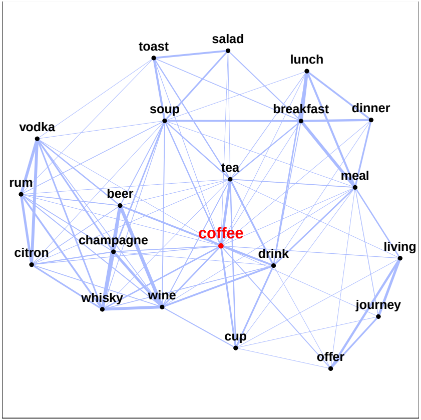
Semantic maps



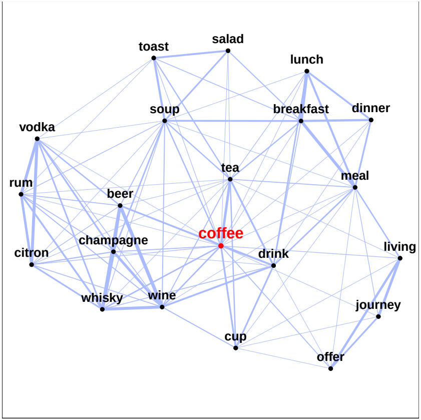
Latent dimensions

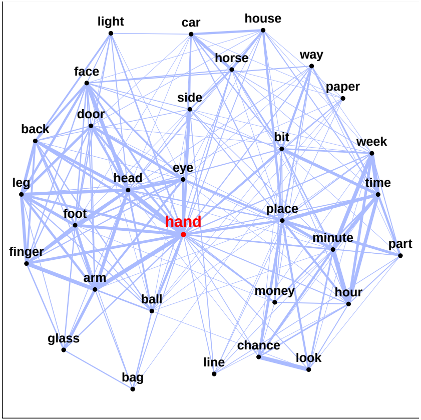


Semantic similarity graph (topological structure)



Semantic similarity graph (topological structure)





Distributional similarity as semantic similarity

- DSMs interpret semantic similarity as a quantitative notion
  - if $\mathbf{x}_A$ is closer to $\mathbf{x}_B$ than to $\mathbf{x}_C$ in the distributional vector space, then A is more semantically similar to B than to C

rhino	fall	rock
woodpecker	rise	lava
rhinoceros	increase	sand
swan	fluctuation	boulder
whale	drop	ice
ivory	decrease	jazz
plover	reduction	slab
elephant	logarithm	cliff
bear	decline	pop
satin	cut	basalt
sweatshirt	hike	crevice

Types of semantic relations in DSMs

- Neighbors in DSMs have different types of semantic relations

car (InfomapNLP on BNC; n = 2)

- van co-hyponym
- vehicle hyperonym
- truck co-hyponym
- motorcycle co-hyponym
- driver related entity
- motor part
- lorry co-hyponym
- motorist related entity
- cavalier hyponym
- bike co-hyponym

car (InfomapNLP on BNC; n = 30)

- drive function
- park typical action
- bonnet part
- windscreen part
- hatchback part
- headlight part
- jaguar hyponym
- garage location
- cavalier hyponym
- tyre part

Semantic similarity and relatedness

- Semantic similarity - two words sharing a high number of salient features (attributes)
  - synonymy (*car/automobile*)
  - hyperonymy (*car/vehicle)*
  - co-hyponymy (*car/van/truck*)

Semantic similarity and relatedness

- Semantic similarity - two words sharing a high number of salient features (attributes)
  - synonymy (*car/automobile*)
  - hyperonymy (*car/vehicle)*
  - co-hyponymy (*car/van/truck*)

- Semantic relatedness (Budanitsky & Hirst 2006) - two words semantically associated without being necessarily similar
  - meronymy (*car/tyre*)
  - function (*car/drive*)
  - attribute (*car/fast*)
  - location (*car/road*)

Outline

Introduction

- The distributional hypothesis
- Three famous DSM examples

Taxonomy of DSM parameters

- Definition of DSM & parameter overview
- Examples

Usage and evaluation of DSM

- Using & interpreting DSM distances
- Evaluation: attributional similarity

Singular Value Decomposition

- Which distance measure?
- Dimensionality reduction and SVD

Discussion

DSMs and semantic similarity

- Most DSM models emphasize paradigmatic similarity
  - words that tend to occur in the same contexts
- Words that share many contexts will correspond to concepts that share many attributes (attributional similarity), i.e. concepts that are taxonomically/ontologically similar
  - synonyms (*rhino*/*rhinoceros*)
  - antonyms and values on a scale (*good*/*bad*)
  - co-hyponyms (*rock*/*jazz*)
  - hyper- and hyponyms (*rock*/*basalt*)
- Taxonomic similarity is seen as the fundamental semantic relation, allowing categorization, generalization, inheritance

Evaluation of attributional similarity

- Synonym identification
  - TOEFL test
- Modeling semantic similarity judgments
  - the Rubenstein/Goodenough norms
- Noun categorization
  - the ESSLLI 2008 dataset
- Semantic priming
  - the Hodgson dataset

The TOEFL synonym task

- The TOEFL dataset
  - 80 items
  - Target: *levied*
    Candidates: *imposed, believed, requested, correlated*

The TOEFL synonym task

- The TOEFL dataset
  - 80 items
  - Target: *levied*
    Candidates: *imposed*, *believed*, *requested*, *correlated*

The TOEFL synonym task

- The TOEFL dataset
  - 80 items
  - Target: *levied*
    Candidates: *imposed*, *believed*, *requested*, *correlated*

- DSMs and TOEFL
  1. take vectors of the target ($\mathbf{t}$) and of the candidates ($\mathbf{c}_1 \dots \mathbf{c}_n$)
  2. measure the distance between $\mathbf{t}$ and $\mathbf{c}_i$, with $1 \leq i \leq n$
  3. select $\mathbf{c}_i$ with the shortest distance in space from $\mathbf{t}$

Humans vs. DSMs on the synonym task

- Humans (Landauer and Dumais 1997; Rapp 2004)
  - Foreign test takers: 64.5%
  - Macquarie non-natives: 86.75%
  - Macquarie natives: 97.75%

- Machines
  - Classic LSA (Landauer and Dumais 1997): 64.4%
  - Padó and Lapata's (2007) dependency-based model: 73%
  - Rapp's (2003) SVD model on lemmatized BNC: 92.5%

Semantic similarity judgments

Dataset Rubenstein and Goodenough (1965) (R&G) of 65 noun pairs rated by 51 subjects on a 0-4 scale

car	*automobile*	3.9
food	*fruit*	2.7
cord	*smile*	0.0

Semantic similarity judgments

Dataset Rubenstein and Goodenough (1965) (R&G) of 65 noun pairs rated by 51 subjects on a 0-4 scale

car	*automobile*	3.9
food	*fruit*	2.7
cord	*smile*	0.0

- DSMs vs. Rubenstein & Goodenough
  1. for each test pair (w_1, w_2), take vectors $\mathbf{w}_1$ and $\mathbf{w}_2$
  2. measure the distance (e.g. cosine) between $\mathbf{w}_1$ and $\mathbf{w}_2$
  3. measure (Pearson) correlation between vector distances and R&G average judgments (Padó and Lapata 2007)

Semantic similarity judgments

Dataset Rubenstein and Goodenough (1965) (R&G) of 65 noun pairs rated by 51 subjects on a 0-4 scale

car	*automobile*	3.9
food	*fruit*	2.7
cord	*smile*	0.0

- DSMs vs. Rubenstein & Goodenough
  1. for each test pair (w_1, w_2), take vectors $\mathbf{w}_1$ and $\mathbf{w}_2$
  2. measure the distance (e.g. cosine) between $\mathbf{w}_1$ and $\mathbf{w}_2$
  3. measure (Pearson) correlation between vector distances and R&G average judgments (Padó and Lapata 2007)

model	*r*
dep-filtered+SVD	0.8
dep-filtered	0.7
dep-linked (DM)	0.64
window	0.63

Categorization

- In categorization tasks, subjects are typically asked to assign experimental items – objects, images, words – to a given category or group items belonging to the same category
  - categorization requires an understanding of the relationship between the items in a category
- Categorization is a basic cognitive operation presupposed by further semantic tasks
  - inference
    - ⋆ if X is a CAR then X is a VEHICLE
  - compositionality
    - ⋆ λy : FOOD λx : ANIMATE; $eat(x, y)$
- "Chicken-and-egg" problem for relationship of categorization and similarity (cf. Goodman 1972, Medin et al. 1993)

Noun categorization

Dataset 44 concrete nouns (ESSLLI 2008 Shared Task)

- 24 natural entities
  - 15 animals:
    7 birds (*eagle*), 8 ground animals (*lion*)
  - 9 plants: 4 fruits (*banana*), 5 greens (*onion*)
- 20 artifacts
  - 13 tools (*hammer*), 7 vehicles (*car*)

Noun categorization

Dataset 44 concrete nouns (ESSLLI 2008 Shared Task)

- 24 natural entities
  - 15 animals:
    7 birds (*eagle*), 8 ground animals (*lion*)
  - 9 plants: 4 fruits (*banana*), 5 greens (*onion*)
- 20 artifacts
  - 13 tools (*hammer*), 7 vehicles (*car*)

- DSMs and noun categorization
  - categorization can be operationalized as a clustering task
    1. for each noun w_i in the dataset, take its vector $\mathbf{w}_i$
    2. apply a clustering method to the set of vectors $\mathbf{w}_i$
    3. evaluate whether clusters correspond to gold-standard semantic classes (purity, entropy, . . .)

Noun categorization

- Clustering experiments with CLUTO (Karypis 2003)
  - repeated bisection algorithm
  - 6-way (birds, ground animals, fruits, greens, tools and vehicles), 3-way (animals, plants and artifacts) and 2-way (natural and artificial entities) clusterings
- Clusters evaluation
  - entropy – whether words from different classes are represented in the same cluster (best $= 0$)
  - purity – degree to which a cluster contains words from one class only (best $= 1$)
  - global score across the three clustering experiments

$$\sum_{i=1}^{3} \text{Purity}_i - \sum_{i=1}^{3} \text{Entropy}_i$$

Noun categorization: results

model	*6-way*		*3-way*		*2-way*		*global*
	P	*E*	*P*	*E*	*P*	*E*	
Katrenko	89	13	100	0	80	59	197
Peirsman+	82	23	84	34	86	55	140
dep-typed (DM)	77	24	79	38	59	97	56
dep-filtered	80	28	75	51	61	95	42
window	75	27	68	51	68	89	44
Peirsman−	73	28	71	54	61	96	27
Shaoul	41	77	52	84	55	93	-106

`Katrenko`, `Peirsman+/-`, `Shaoul`: ESSLLI 2008 Shared Task
`DM`: Baroni & Lenci (2009)

Semantic priming

- Hearing/reading a "related" prime facilitates access to a target in various lexical tasks (naming, lexical decision, reading)
  - the word *pear* is recognized/accessed faster if it is heard/read after *apple*
- Hodgson (1991) single word lexical decision task, 136 prime-target pairs (cf. Padó and Lapata 2007)
  - similar amounts of priming for different semantic relations between primes and targets (approx. 23 pairs per relation):
    - ★ synonyms (synonym): *to dread/to fear*
    - ★ antonyms (antonym): *short/tall*
    - ★ coordinates (coord): *train/truck*
    - ★ super- and subordinate pairs (supersub): *container/bottle*
    - ★ free association pairs (freeass): *dove/peace*
    - ★ phrasal associates (phrasacc): *vacant/building*

Simulating semantic priming

McDonald & Brew (2004), Padó & Lapata (2007)

- DSMs and semantic priming
  1. for each related prime-target pair, measure cosine-based similarity between pair items (e.g., *to dread/to fear*)
  2. to estimate unrelated primes, take average of cosine-based similarity of target with other primes from same relation data-set (e.g., *value/to fear*)
  3. similarity between related items should be significantly higher than average similarity between unrelated items

- Significant effects ($p < .01$) for all semantic relations
  - strongest effects for synonyms, antonyms & coordinates

Outline

Introduction

The distributional hypothesis

Three famous DSM examples

Taxonomy of DSM parameters

Definition of DSM & parameter overview

Examples

Usage and evaluation of DSM

Using & interpreting DSM distances

Evaluation: attributional similarity

Singular Value Decomposition

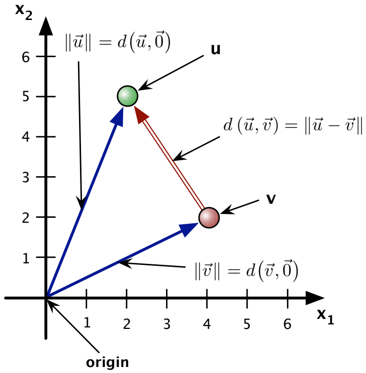
Which distance measure?

Dimensionality reduction and SVD

Discussion

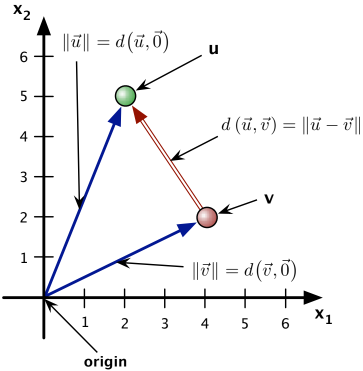
Distance vs. norm

- Intuitively, geometric **distance** $d(\mathbf{u}, \mathbf{v})$ corresponds to **length** $\|\mathbf{u} - \mathbf{v}\|$ of displacement vector $\mathbf{u} - \mathbf{v}$
  - $d(\mathbf{u}, \mathbf{v})$ is a **metric**
  - $\|\mathbf{u} - \mathbf{v}\|$ is a **norm**
  - $\|\mathbf{u}\| = d(\mathbf{u}, \mathbf{0})$
- Such a metric is always translation-invariant


$\mathbf{x_2}$
6
5
4
3
2
1
$\|\vec{u}\| = d(\vec{u}, \vec{0})$
u
$d(\vec{u}, \vec{v}) = \|\vec{u} - \vec{v}\|$
v
$\|\vec{v}\| = d(\vec{v}, \vec{0})$
1
2
3
4
5
6
$\mathbf{x_1}$
origin

Distance vs. norm

- Intuitively, geometric **distance** $d(\mathbf{u}, \mathbf{v})$ corresponds to **length** $\|\mathbf{u} - \mathbf{v}\|$ of displacement vector $\mathbf{u} - \mathbf{v}$
  - $d(\mathbf{u}, \mathbf{v})$ is a **metric**
  - $\|\mathbf{u} - \mathbf{v}\|$ is a **norm**
  - $\|\mathbf{u}\| = d(\mathbf{u}, \mathbf{0})$
- Such a metric is always translation-invariant



$\mathbf{x_2}$
$\|\vec{u}\| = d(\vec{u}, \vec{0})$
u
6
5
4
3
2
1
$d(\vec{u}, \vec{v}) = \|\vec{u} - \vec{v}\|$
v
$\|\vec{v}\| = d(\vec{v}, \vec{0})$
1
2
3
4
5
6
$\mathbf{x_1}$
origin

- $d_p(\mathbf{u}, \mathbf{v}) = \|\mathbf{v} - \mathbf{u}\|_p$
- **Minkowski *p*-norm** for $p \in [1, \infty]$:

$$\|\mathbf{u}\|_p := \left(|u_1|^p + \cdots + |u_n|^p\right)^{1/p}$$

Which distance measure should I use?

- Choice of metric or norm is one of the parameters of a DSM

Which distance measure should I use?

- Choice of metric or norm is one of the parameters of a DSM
- Measures of **distance** between points:
  - intuitive Euclidean norm $\|\cdot\|_2$
  - "city-block" Manhattan distance $\|\cdot\|_1$
  - maximum distance $\|\cdot\|_\infty$
  - general Minkowski *p*-norm $\|\cdot\|_p$
  - and many other formulae …

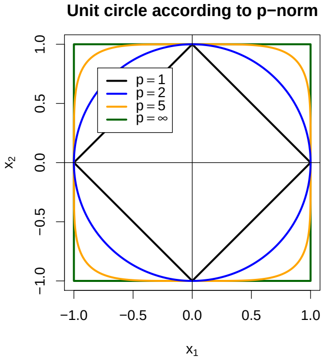
Which distance measure should I use?

- Choice of metric or norm is one of the parameters of a DSM
- Measures of **distance** between points:
  - intuitive Euclidean norm $\|\cdot\|_2$
  - "city-block" Manhattan distance $\|\cdot\|_1$
  - maximum distance $\|\cdot\|_\infty$
  - general Minkowski p-norm $\|\cdot\|_p$
  - and many other formulae . . .
- Measures of the **similarity** of arrows:
  - "cosine distance" $\sim u_1 v_1 + \cdots + u_n v_n$
  - Dice coefficient (matching non-zero coordinates)
  - and, of course, many other formulae . . .
  - ☞ these measures determine **angles** between arrows

Which distance measure should I use?

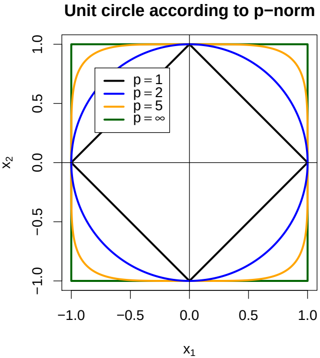
- Choice of metric or norm is one of the parameters of a DSM
- Measures of **distance** between points:
  - intuitive Euclidean norm $\|\cdot\|_2$
  - "city-block" Manhattan distance $\|\cdot\|_1$
  - maximum distance $\|\cdot\|_\infty$
  - general Minkowski *p*-norm $\|\cdot\|_p$
  - and many other formulae …
- Measures of the **similarity** of arrows:
  - "cosine distance" $\sim \; u_1 v_1 + \cdots + u_n v_n$
  - Dice coefficient (matching non-zero coordinates)
  - and, of course, many other formulae …
  - ☞ these measures determine **angles** between arrows
- **Information-theoretic** measures
  - KL-divergence, skew divergence, …
  - most sensible in a probabilistic analysis of the DSM matrix

The family of Minkowski p-norms



- visualisation of norms in $\mathbb{R}^2$ by plotting **unit circle** for each norm, i.e. points $\mathbf{u}$ with $\|\mathbf{u}\| = 1$
- here: p-norms $\|\cdot\|_p$ for different values of p
- triangle inequality $\iff$ unit circle is **convex** $\iff$ holds for $p \geq 1$

The family of Minkowski *p*-norms



- visualisation of norms in $\mathbb{R}^2$ by plotting **unit circle** for each norm, i.e. points $\mathbf{u}$ with $\|\mathbf{u}\| = 1$
- here: *p*-norms $\|\cdot\|_p$ for different values of *p*
- triangle inequality $\Longleftrightarrow$ unit circle is **convex** $\Longleftrightarrow$ holds for $p \geq 1$

- Consequence for DSM: $p \gg 2$ "favours" small differences in many coordinates, $p \ll 2$ differences in few coordinates
- Rotation-invariance of Euclidean norm ➜ many intuitive and convenient geometric properties (orthogonality, angles, ...)

Euclidean norm & inner product

- The Euclidean norm $\|\mathbf{u}\|_2 = \sqrt{\langle \mathbf{u}, \mathbf{u} \rangle}$ is special because it can be derived from the **inner product**:

$$\langle \mathbf{u}, \mathbf{v} \rangle := \mathbf{x}^T \mathbf{y} = x_1 y_1 + \cdots + x_n y_n$$

Euclidean norm & inner product

- The Euclidean norm $\|\mathbf{u}\|_2 = \sqrt{\langle \mathbf{u}, \mathbf{u} \rangle}$ is special because it can be derived from the **inner product**:

$$\langle \mathbf{u}, \mathbf{v} \rangle := \mathbf{x}^T \mathbf{y} = x_1 y_1 + \cdots + x_n y_n$$

- **Angle** ϕ between vectors $\mathbf{u}, \mathbf{v} \in \mathbb{R}^n$:

$$\cos \phi := \frac{\langle \mathbf{u}, \mathbf{v} \rangle}{\|\mathbf{u}\| \cdot \|\mathbf{v}\|}$$

☞ Euclidean norm closely related to cosine similarity $\cos \phi$

Euclidean norm & inner product

- The Euclidean norm $\|\mathbf{u}\|_2 = \sqrt{\langle \mathbf{u}, \mathbf{u} \rangle}$ is special because it can be derived from the **inner product**:

$$\langle \mathbf{u}, \mathbf{v} \rangle := \mathbf{x}^T \mathbf{y} = x_1 y_1 + \cdots + x_n y_n$$

- **Angle** ϕ between vectors $\mathbf{u}, \mathbf{v} \in \mathbb{R}^n$:

$$\cos \phi := \frac{\langle \mathbf{u}, \mathbf{v} \rangle}{\|\mathbf{u}\| \cdot \|\mathbf{v}\|}$$

☞ Euclidean norm closely related to cosine similarity $\cos \phi$

- $\mathbf{u}$ and $\mathbf{v}$ are **orthogonal** iff $\langle \mathbf{u}, \mathbf{v} \rangle = 0$
  - the **shortest connection** between a point $\mathbf{u}$ and a subspace U is orthogonal to all vectors $\mathbf{v} \in U$

Euclidean distance or cosine similarity?

- Which is better, Euclidean distance or cosine similarity?

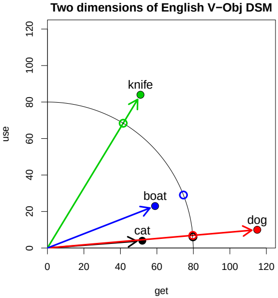
Euclidean distance or cosine similarity?

- Which is better, Euclidean distance or cosine similarity?
- They are equivalent: if vectors are normalised ($\|\mathbf{u}\|_2 = 1$), both lead to the same neighbour ranking

Euclidean distance or cosine similarity?

- Which is better, Euclidean distance or cosine similarity?
- They are equivalent: if vectors are normalised ($\|\mathbf{u}\|_2 = 1$), both lead to the same neighbour ranking

$$\begin{aligned} d_2(\mathbf{u},\mathbf{v}) &= \sqrt{\|\mathbf{u}-\mathbf{v}\|_2} \\ &= \sqrt{\langle \mathbf{u}-\mathbf{v}, \mathbf{u}-\mathbf{v}\rangle} \\ &= \sqrt{\langle \mathbf{u},\mathbf{u}\rangle + \langle \mathbf{v},\mathbf{v}\rangle - 2\,\langle \mathbf{u},\mathbf{v}\rangle} \\ &= \sqrt{\|\mathbf{u}\|_2 + \|\mathbf{v}\|_2 - 2\,\langle \mathbf{u},\mathbf{v}\rangle} \\ &= \sqrt{2 - 2\cos\phi} \end{aligned}$$



Outline

Introduction

- The distributional hypothesis
- Three famous DSM examples

Taxonomy of DSM parameters

- Definition of DSM & parameter overview
- Examples

Usage and evaluation of DSM

- Using & interpreting DSM distances
- Evaluation: attributional similarity

Singular Value Decomposition

- Which distance measure?
- Dimensionality reduction and SVD

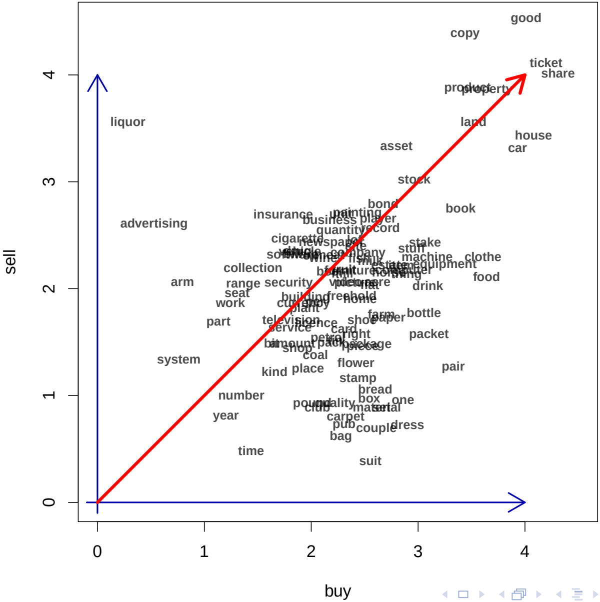
Discussion

Motivating latent dimensions & subspace projection

- The **latent property** of being a commodity is “expressed” through associations with several verbs: *sell*, *buy*, *acquire*, . . .
- Consequence: these DSM dimensions will be **correlated**

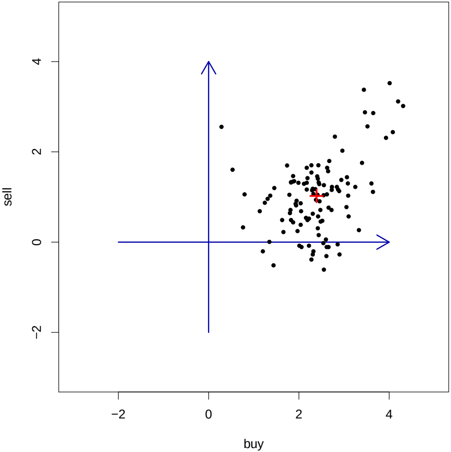
- Identify **latent dimension** by looking for strong correlations (or weaker correlations between large sets of features)
- Projection into subspace V of $k < n$ latent dimensions as a “**noise reduction**” technique ➜ **LSA**
- Assumptions of this approach:
  - “latent” distances in V are semantically meaningful
  - other “residual” dimensions represent chance co-occurrence patterns, often particular to the corpus underlying the DSM

The latent “commodity” dimension



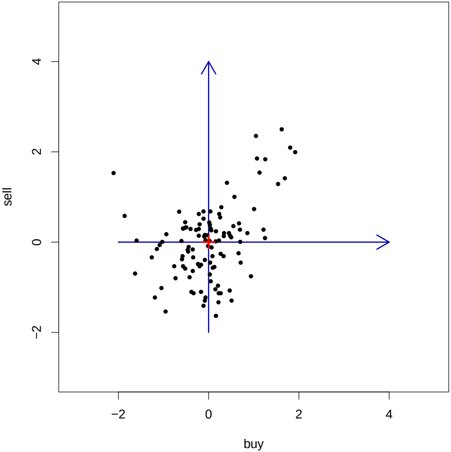
Centering and variance

- **Uncentered data set**
- Centered data set
- Variance of centered data



Centering and variance

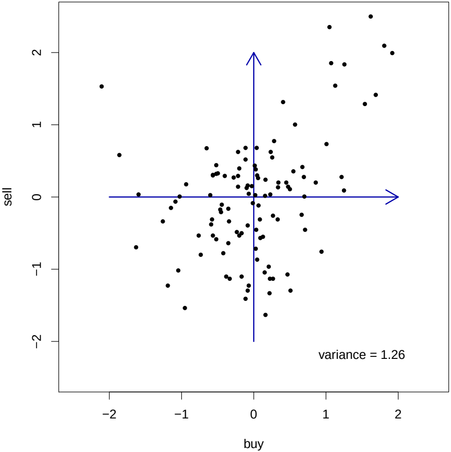
- Uncentered data set
- **Centered data set**
- Variance of centered data



Centering and variance

- Uncentered data set
- Centered data set
- **Variance of centered data**

$$\sigma^2 = \frac{1}{m-1} \sum_{i=1}^{m} \|\mathbf{x}_i\|^2$$



Principal components analysis (PCA)

- We want to project the data points to a lower-dimensional subspace, but preserve their mutual distances as well as possible
- Insight 1: variance = average squared distance

$$\frac{1}{m(m-1)}\sum_{i=1}^{m}\sum_{j=1}^{m}\|\mathbf{x}_i-\mathbf{x}_j\|^2=\frac{2}{m-1}\sum_{i=1}^{m}\|\mathbf{x}_i\|^2=2\sigma^2$$

- Insight 2: for an orthogonal projection, loss of variance corresponds to average change in distances between points

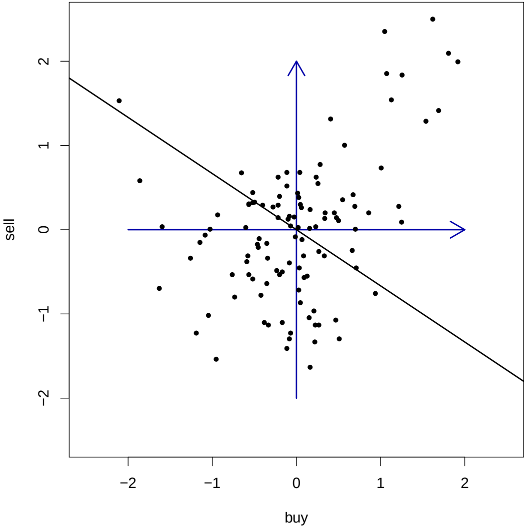
Principal components analysis (PCA)

- We want to project the data points to a lower-dimensional subspace, but preserve their mutual distances as well as possible
- Insight 1: variance = average squared distance

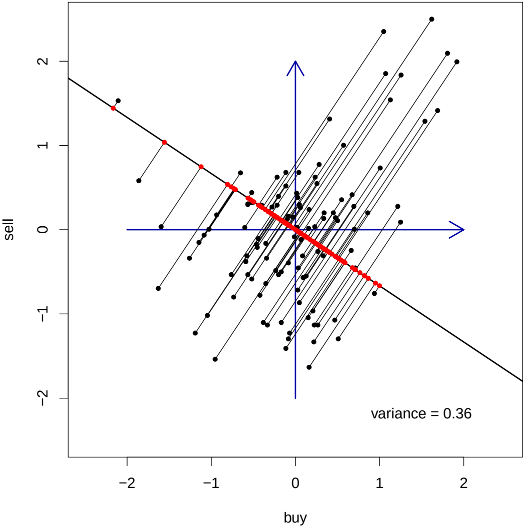
$$\frac{1}{m(m-1)} \sum_{i=1}^{m} \sum_{j=1}^{m} \|\mathbf{x}_i - \mathbf{x}_j\|^2 = \frac{2}{m-1} \sum_{i=1}^{m} \|\mathbf{x}_i\|^2 = 2\sigma^2$$

- Insight 2: for an orthogonal projection, loss of variance corresponds to average change in distances between points
- If we reduced the data set to just a single dimension, which dimension would preserve the most variance?
- Mathematically, we project the points onto a line through the origin and calculate one-dimensional variance on this line
  - we'll see in a moment how to compute such projections
  - but first, let us look at a few examples

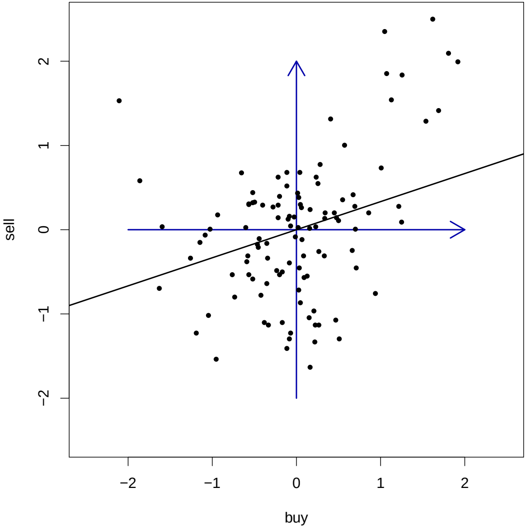
Projection and preserved variance: examples



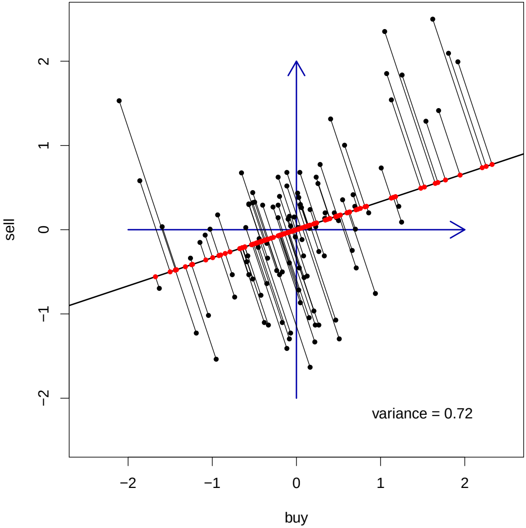
Projection and preserved variance: examples



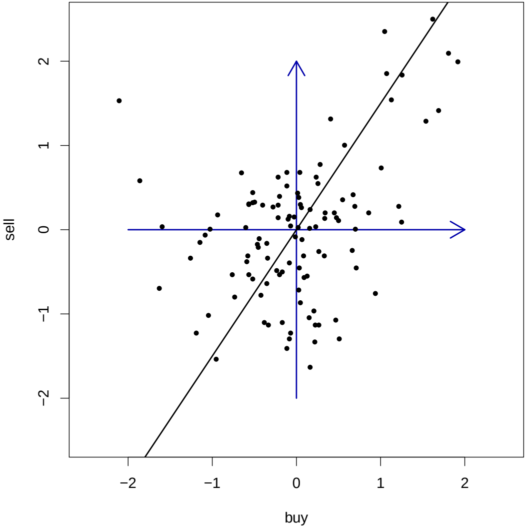
Projection and preserved variance: examples



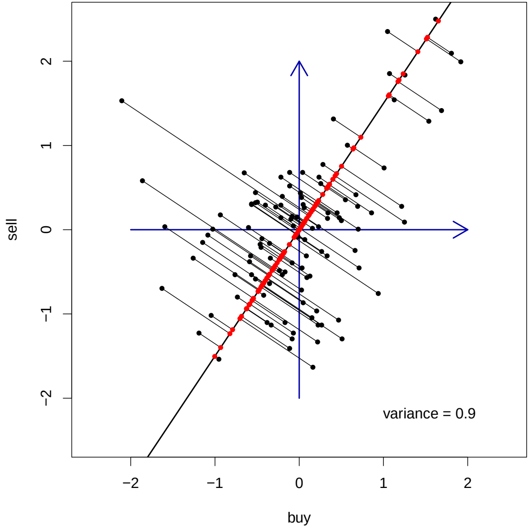
Projection and preserved variance: examples



Projection and preserved variance: examples



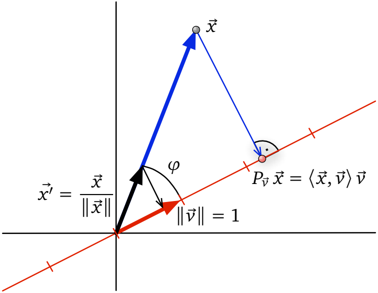
Projection and preserved variance: examples



The covariance matrix

- 1-D subspace described by unit vector $\|\mathbf{v}\| = 1$
- Orthogonal projection $P_{\mathbf{v}}$ onto this line

$$P_{\mathbf{v}}\mathbf{x} = \langle \mathbf{x}, \mathbf{v} \rangle\, \mathbf{v}$$


$\vec{x}$
φ
$\vec{x}' = \frac{\vec{x}}{\|\vec{x}\|}$
$P_{\vec{v}}\,\vec{x} = \langle \vec{x}, \vec{v} \rangle\, \vec{v}$
$\|\vec{v}\| = 1$

- Residual variance given by

$$\sigma_{\mathbf{v}}^2 = \frac{1}{m-1} \sum_{i=1}^{m} \langle \mathbf{x}_i, \mathbf{v} \rangle^2 = \mathbf{v}^T \mathbf{C} \mathbf{v}$$

where $\mathbf{C} = \frac{1}{m-1} \mathbf{M}^T \mathbf{M}$ is the covariance matrix of the DSM $\mathbf{M}$

Maximizing preserved variance

- In our example, we want to find the axis $\mathbf{v}_1$ that preserves the largest amount of variance by maximizing $\mathbf{v}_1^T \mathbf{C} \mathbf{v}_1$

Maximizing preserved variance

- In our example, we want to find the axis $\mathbf{v}_1$ that preserves the largest amount of variance by maximizing $\mathbf{v}_1^T \mathbf{C} \mathbf{v}_1$
- For higher-dimensional data set, we also want to find the axis $\mathbf{v}_2$ with the second largest amount of variance, etc.
  - ☞ Should not include variance that has already been accounted for: $\mathbf{v}_2$ must be orthogonal to $\mathbf{v}_1$, i.e. $\langle \mathbf{v}_1, \mathbf{v}_2 \rangle = 0$

Maximizing preserved variance

- In our example, we want to find the axis $\mathbf{v}_1$ that preserves the largest amount of variance by maximizing $\mathbf{v}_1^T \mathbf{C} \mathbf{v}_1$
- For higher-dimensional data set, we also want to find the axis $\mathbf{v}_2$ with the second largest amount of variance, etc.
  - ☞ Should not include variance that has already been accounted for: $\mathbf{v}_2$ must be orthogonal to $\mathbf{v}_1$, i.e. $\langle \mathbf{v}_1, \mathbf{v}_2 \rangle = 0$
- Orthogonal dimensions $\mathbf{v}_1, \mathbf{v}_2, \ldots$ **partition** variance:

$$\sigma^2 = \sigma^2_{\mathbf{v}_1} + \sigma^2_{\mathbf{v}_2} + \ldots$$

Maximizing preserved variance

- In our example, we want to find the axis $\mathbf{v}_1$ that preserves the largest amount of variance by maximizing $\mathbf{v}_1^T \mathbf{C} \mathbf{v}_1$
- For higher-dimensional data set, we also want to find the axis $\mathbf{v}_2$ with the second largest amount of variance, etc.
  - ☞ Should not include variance that has already been accounted for: $\mathbf{v}_2$ must be orthogonal to $\mathbf{v}_1$, i.e. $\langle \mathbf{v}_1, \mathbf{v}_2 \rangle = 0$
- Orthogonal dimensions $\mathbf{v}_1, \mathbf{v}_2, \dots$ **partition** variance:

$$\sigma^2 = \sigma^2_{\mathbf{v}_1} + \sigma^2_{\mathbf{v}_2} + \dots$$

- Useful result from linear algebra: every symmetric matrix $\mathbf{C} = \mathbf{C}^T$ has an **eigenvalue decomposition** with orthogonal **eigenvectors** $\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n$ and corresponding **eigenvalues** $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$

Eigenvalue decomposition

- The eigenvalue decomposition of $\mathbf{C}$ can be written in the form

$$\mathbf{C} = \mathbf{U} \cdot \mathbf{D} \cdot \mathbf{U}^T$$

where $\mathbf{U}$ is an orthogonal matrix of eigenvectors (columns) and $\mathbf{D} = \mathrm{Diag}(\lambda_1, \ldots, \lambda_n)$ a diagonal matrix of eigenvalues

$$\mathbf{U} = \begin{bmatrix} \vdots & \vdots & & \vdots \\ \vdots & \vdots & & \vdots \\ \mathbf{a}_1 & \mathbf{a}_2 & \cdots & \mathbf{a}_n \\ \vdots & \vdots & & \vdots \\ \vdots & \vdots & & \vdots \end{bmatrix} \qquad \mathbf{D} = \begin{bmatrix} \lambda_1 & & & & \\ & \lambda_2 & & & \\ & & \ddots & & \\ & & & \ddots & \\ & & & & \lambda_n \end{bmatrix}$$

  - note that both $\mathbf{U}$ and $\mathbf{D}$ are $n \times n$ square matrices

An aside: orthogonal matrices

- A $n \times n$ matrix $\mathbf{U}$ with orthonormal columns $\mathbf{a}_i$, i.e.

$$\langle \mathbf{a}_i, \mathbf{a}_j \rangle = \delta_{ij} = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases}$$

  is called an **orthogonal** matrix

An aside: orthogonal matrices

- A $n \times n$ matrix $\mathbf{U}$ with orthonormal columns $\mathbf{a}_i$, i.e.

  $$\langle \mathbf{a}_i, \mathbf{a}_j \rangle = \delta_{ij} = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases}$$

  is called an **orthogonal** matrix

- The **inverse** of an orthogonal matrix is simply its transpose:

  $$\mathbf{U}^{-1} = \mathbf{U}^T \quad \text{if } \mathbf{U} \text{ is orthogonal}$$

  i.e. we have $\mathbf{U}^T\mathbf{U} = \mathbf{U}\mathbf{U}^T = \mathbf{I}$ (the identity matrix)

An aside: orthogonal matrices

- A $n \times n$ matrix $\mathbf{U}$ with orthonormal columns $\mathbf{a}_i$, i.e.

$$\langle \mathbf{a}_i, \mathbf{a}_j \rangle = \delta_{ij} = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases}$$

  is called an **orthogonal** matrix

- The **inverse** of an orthogonal matrix is simply its transpose:

$$\mathbf{U}^{-1} = \mathbf{U}^T \quad \text{if } \mathbf{U} \text{ is orthogonal}$$

  i.e. we have $\mathbf{U}^T\mathbf{U} = \mathbf{U}\mathbf{U}^T = \mathbf{I}$ (the identity matrix)

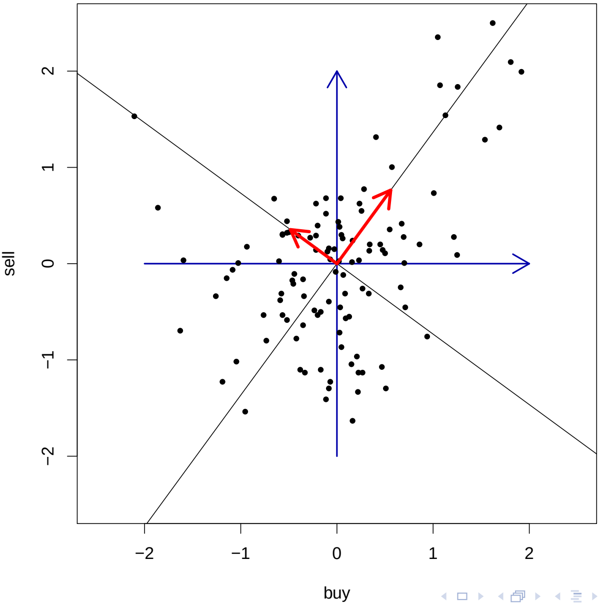
- Multiplication with an orthogonal matrix preserves Euclidean norm and inner product (i.e. angle):

$$\|\mathbf{U}\mathbf{x}\|_2 = \|\mathbf{x}\|_2 \quad \text{and} \quad \langle \mathbf{U}\mathbf{x}, \mathbf{U}\mathbf{y} \rangle = \langle \mathbf{x}, \mathbf{y} \rangle$$

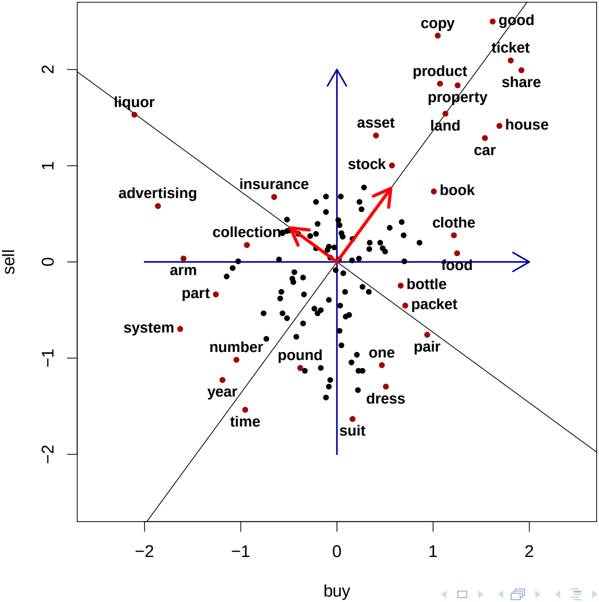
The PCA algorithm

- The eigenvectors $\mathbf{a}_i$ of the covariance matrix $\mathbf{C}$ are called the **principal components** of the data set
- The amount of variance preserved (or "explained") by the i-th principal component is given by the eigenvalue λ_i
- Since $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$, the first principal component accounts for the largest amount of variance etc.
- Coordinates of a point $\mathbf{x}$ in PCA space are given by $\mathbf{U}^T\mathbf{x}$ (note: these are the projections on the principal components)
- For the purpose of "noise reduction", only the first $k \ll n$ principal components (with highest variance) are retained, and the other dimensions in PCA space are dropped
  - ☞ i.e. data points are projected into the subspace V spanned by the first k column vectors of $\mathbf{U}$

PCA example



PCA example


copy
good
ticket
product
share
property
liquor
asset
land
house
car
stock
insurance
book
advertising
collection
clothe
arm
food
bottle
part
packet
system
pair
number
pound
one
year
dress
time
suit
sell
buy
−2
−1
0
1
2
−2
−1
0
1
2

Singular value decomposition (SVD)

- The idea of eigenvalue decomposition can be generalised to an arbitrary (non-symmetric, non-square) matrix **A**
  - ☞ such a matrix need not have any eigenvalues

Singular value decomposition (SVD)

- The idea of eigenvalue decomposition can be generalised to an arbitrary (non-symmetric, non-square) matrix **A**
  - ☞ such a matrix need not have any eigenvalues
- **Singular value decomposition** (**SVD**) factorises **A** into

$$\mathbf{A} = \mathbf{U} \cdot \mathbf{\Sigma} \cdot \mathbf{V}^T$$

  where **U** and **V** are orthogonal coordinate transformations and **Σ** is a rectangular-diagonal matrix of **singular values** (with customary ordering $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n \geq 0$)
- SVD is an important tool in linear algebra and statistics
  - ☞ in particular, PCA can be computed from SVD decomposition

SVD illustration

$$\begin{array}{cc} & n \\ m & \begin{bmatrix} \\ \\ \mathbf{A} \\ \\ \\ \end{bmatrix} \end{array} = \begin{array}{cc} & m \\ m & \begin{bmatrix} \\ \\ \mathbf{U} \\ \\ \\ \end{bmatrix} \end{array} \cdot \begin{array}{cc} & n \\ m & \begin{bmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_n \\ & \mathbf{\Sigma} & \\ & & \end{bmatrix} \end{array} \cdot \begin{array}{cc} & n \\ n & \begin{bmatrix} \\ \mathbf{V}^T \\ \\ \end{bmatrix} \end{array}$$

(This illustration assumes $m > n$, i.e. **A** has more rows than columns. For $m < n$, $\mathbf{\Sigma}$ is a horizontal rectangle with diagonal elements $\sigma_1, \ldots, \sigma_m$.)

PCA by singular value decomposition

- PCA needs to find an eigenvalue decomposition of the covariance matrix $\mathbf{C} = \frac{1}{m-1}\mathbf{M}^T\mathbf{M}$, or equivalently of $\mathbf{M}^T\mathbf{M}$
- Like every matrix, $\mathbf{M}$ has a singular value decomposition

$$\mathbf{M} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^T$$

PCA by singular value decomposition

- PCA needs to find an eigenvalue decomposition of the covariance matrix $\mathbf{C} = \frac{1}{m-1}\mathbf{M}^T\mathbf{M}$, or equivalently of $\mathbf{M}^T\mathbf{M}$
- Like every matrix, $\mathbf{M}$ has a singular value decomposition

$$\mathbf{M} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^T$$

- By inserting the SVD, we obtain

$$\begin{aligned}
\mathbf{M}^T\mathbf{M} &= \big(\mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^T\big)^T\mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^T \\
&= (\mathbf{V}^T)^T\boldsymbol{\Sigma}^T\underbrace{\mathbf{U}^T\mathbf{U}}_{\mathbf{I}}\boldsymbol{\Sigma}\mathbf{V}^T \\
&= \mathbf{V}(\underbrace{\boldsymbol{\Sigma}^T\boldsymbol{\Sigma}}_{\boldsymbol{\Sigma}^2})\mathbf{V}^T
\end{aligned}$$

PCA by singular value decomposition

- We have found the eigenvalue decomposition

$$\mathbf{M}^T\mathbf{M} = \mathbf{V}\mathbf{\Sigma}^2\mathbf{V}^T$$

with

$$\mathbf{\Sigma}^2 = \mathbf{\Sigma}^T\mathbf{\Sigma} = \begin{bmatrix} (\sigma_1)^2 & n & \\ n & \ddots & \\ & & (\sigma_n)^2 \end{bmatrix}$$

PCA by singular value decomposition

- We have found the eigenvalue decomposition

$$\mathbf{M}^T\mathbf{M} = \mathbf{V}\mathbf{\Sigma}^2\mathbf{V}^T$$

with

$$\mathbf{\Sigma}^2 = \mathbf{\Sigma}^T\mathbf{\Sigma} = \begin{bmatrix} (\sigma_1)^2 & n & \\ n & \ddots & \\ & & (\sigma_n)^2 \end{bmatrix}$$

- The column vectors of **V** are **latent dimensions**

PCA by singular value decomposition

- We have found the eigenvalue decomposition

$$\mathbf{M}^T\mathbf{M} = \mathbf{V}\mathbf{\Sigma}^2\mathbf{V}^T$$

with

$$\mathbf{\Sigma}^2 = \mathbf{\Sigma}^T\mathbf{\Sigma} = \begin{bmatrix} (\sigma_1)^2 & n & \\ n & \ddots & \\ & & (\sigma_n)^2 \end{bmatrix}$$

- The column vectors of **V** are **latent dimensions**
- The corresponding squared **singular values** partition variance: $(\sigma_1)^2 / \sum_i (\sigma_i)^2$ = proportion along first latent dimension
  - ☞ intuitively, singular value shows importance of latent dimension

PCA by singular value decomposition

- We have found the eigenvalue decomposition

$$\mathbf{M}^T\mathbf{M} = \mathbf{V}\mathbf{\Sigma}^2\mathbf{V}^T$$

with

$$\mathbf{\Sigma}^2 = \mathbf{\Sigma}^T\mathbf{\Sigma} = \begin{bmatrix} (\sigma_1)^2 & n & \\ n & \ddots & \\ & & (\sigma_n)^2 \end{bmatrix}$$

- The column vectors of $\mathbf{V}$ are **latent dimensions**
- The corresponding squared **singular values** partition variance: $(\sigma_1)^2 / \sum_i (\sigma_i)^2$ = proportion along first latent dimension
  - ☞ intuitively, singular value shows importance of latent dimension
- Interpretation of $\mathbf{U}$ is less intuitive (**latent families** of words?)

Transforming the DSM matrix

- We can directly transform the columns of **M** into PCA space:

$$\mathbf{M}\mathbf{V} = \mathbf{U}\boldsymbol{\Sigma}(\mathbf{V}^T\mathbf{V}) = \mathbf{U}\boldsymbol{\Sigma}$$

Transforming the DSM matrix

- We can directly transform the columns of **M** into PCA space:

$$\mathbf{MV} = \mathbf{U\Sigma}(\mathbf{V}^T\mathbf{V}) = \mathbf{U\Sigma}$$

- For "noise reduction", project into m-dimensional subspace by dropping all but the first $k \ll n$ columns of $\mathbf{U\Sigma}$

➥ Sufficient to calculate the first m **singular values** $\sigma_1, \dots, \sigma_m$ and **left singular vectors** $\mathbf{a}_1, \dots, \mathbf{a}_m$ (columns of **U**)

Transforming the DSM matrix

- We can directly transform the columns of **M** into PCA space:

$$\mathbf{MV} = \mathbf{U\Sigma}(\mathbf{V}^T\mathbf{V}) = \mathbf{U\Sigma}$$

- For "noise reduction", project into *m*-dimensional subspace by dropping all but the first $k \ll n$ columns of $\mathbf{U\Sigma}$

➥ Sufficient to calculate the first *m* **singular values** $\sigma_1, \ldots, \sigma_m$ and **left singular vectors** $\mathbf{a}_1, \ldots, \mathbf{a}_m$ (columns of **U**)

- What is the difference between SVD and PCA?

Transforming the DSM matrix

- We can directly transform the columns of **M** into PCA space:

$$\mathbf{MV} = \mathbf{U\Sigma}(\mathbf{V}^T\mathbf{V}) = \mathbf{U\Sigma}$$

- For "noise reduction", project into *m*-dimensional subspace by dropping all but the first $k \ll n$ columns of $\mathbf{U\Sigma}$

➥ Sufficient to calculate the first *m* **singular values** $\sigma_1, \ldots, \sigma_m$ and **left singular vectors** $\mathbf{a}_1, \ldots, \mathbf{a}_m$ (columns of **U**)

- What is the difference between SVD and PCA?
  - we forgot to center and rescale the data!
  - if **M** contains only non-negative values, first latent dimension points from origin towards positive sector ➜ "uninteresting"
  - for a sparse cooccurrence matrix **M**, direct SVD application (as used in LSA) may be more sensible than standard PCA

Time for discussion

- Mathematical insights (based on SVD and other arguments)
  - LSA is a topic model ➜ probabilistic topic models
  - term-document DSM = first-order association,
    term-term DSM = second-order association
  - term-document + SVD vs. term-term vs. higher-order models
  - context types: between term-term and term-context models
- Visualisation of high-dimensional spaces
- How to explore DSM parameters
- Kernel PCA, Isomap, and other nonlinear methods
- Compositionality & holographic memory
- Word senses, polysemy and context-dependence
- Beyond matrices: multi-way relations

Further information

- DSM tutorial & other materials available from

  `http://wordspace.collocations.de/`

  ☞ will be extended during the next few months

- Ongoing work on R package for a DSM toy laboratory:

  `http://r-forge.r-project.org/projects/wordspace/`

- Compact DSM textbook in preparation for *Synthesis Lectures on Human Language Technologies* (Morgan & Claypool)

References I

Bengio, Yoshua; Ducharme, Réjean; Vincent, Pascal; Jauvin, Christian (2003). A neural probabilistic language model. *Journal of Machine Learning Research*, **3**, 1137–1155.

Berry, Michael W. (1992). Large scale singular value computation. *International Journal of Supercomputer Applications*, **6**(1), 13–49.

Blei, David M.; Ng, Andrew Y.; Jordan, Michael, I. (2003). Latent dirichlet allocation. *Journal of Machine Learning Research*, **3**, 993–1022.

Church, Kenneth W. and Hanks, Patrick (1990). Word association norms, mutual information, and lexicography. *Computational Linguistics*, **16**(1), 22–29.

Dumais, S. T.; Furnas, G. W.; Landauer, T. K.; Deerwester, S.; Harshman, R. (1988). Using latent semantic analysis to improve access to textual information. In *CHI '88: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 281–285.

Dunning, Ted E. (1993). Accurate methods for the statistics of surprise and coincidence. *Computational Linguistics*, **19**(1), 61–74.

Evert, Stefan (2004). *The Statistics of Word Cooccurrences: Word Pairs and Collocations*. Dissertation, Institut für maschinelle Sprachverarbeitung, University of Stuttgart. Published in 2005, URN urn:nbn:de:bsz:93-opus-23714. Available from `http://www.collocations.de/phd.html`.

References II

Evert, Stefan (2008). Corpora and collocations. In A. Lüdeling and M. Kytö (eds.), *Corpus Linguistics. An International Handbook*, chapter 58. Mouton de Gruyter, Berlin.

Evert, Stefan (2010). Google Web 1T5 n-grams made easy (but not for the computer). In *Proceedings of the 6th Web as Corpus Workshop (WAC-6)*, Los Angeles, CA.

Firth, J. R. (1957). A synopsis of linguistic theory 1930–55. In *Studies in linguistic analysis*, pages 1–32. The Philological Society, Oxford. Reprinted in Palmer (1968), pages 168–205.

Grefenstette, Gregory (1994). *Explorations in Automatic Thesaurus Discovery*, volume 278 of *Kluwer International Series in Engineering and Computer Science*. Springer, Berlin, New York.

Harris, Zellig (1954). Distributional structure. *Word*, **10**(23), 146–162.

Hoffmann, Thomas (1999). Probabilistic latent semantic analysis. In *Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence (UAI'99)*.

Landauer, Thomas K. and Dumais, Susan T. (1997). A solution to Plato's problem: The latent semantic analysis theory of acquisition, induction and representation of knowledge. *Psychological Review*, **104**(2), 211–240.

References III

Li, Ping; Burgess, Curt; Lund, Kevin (2000). The acquisition of word meaning through global lexical co-occurences. In E. V. Clark (ed.), *The Proceedings of the Thirtieth Annual Child Language Research Forum*, pages 167–178. Stanford Linguistics Association.

Lin, Dekang (1998a). Automatic retrieval and clustering of similar words. In *Proceedings of the 17th International Conference on Computational Linguistics (COLING-ACL 1998)*, pages 768–774, Montreal, Canada.

Lin, Dekang (1998b). An information-theoretic definition of similarity. In *Proceedings of the 15th International Conference on Machine Learning (ICML-98)*, pages 296–304, Madison, WI.

Lund, Kevin and Burgess, Curt (1996). Producing high-dimensional semantic spaces from lexical co-occurrence. *Behavior Research Methods, Instruments, & Computers*, **28**(2), 203–208.

Padó, Sebastian and Lapata, Mirella (2007). Dependency-based construction of semantic space models. *Computational Linguistics*, pages 161–199.

Pantel, Patrick; Lin, Dekang (2000). An unsupervised approach to prepositional phrase attachment using contextually similar words. In *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics*, Hongkong, China.

References IV

Pantel, Patrick; Crestan, Eric; Borkovsky, Arkady; Popescu, Ana-Maria; Vyas, Vishnu (2009). Web-scale distributional similarity and entity set expansion. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 938–947, Singapore.

Rapp, Reinhard (2003). Discovering the meanings of an ambiguous word by searching for sense descriptors with complementary context patterns. In *Proceedings of the 5èmes Rencontres Terminologie et Intelligence Artificielle (TIA-2003)*, Strasbourg, France.

Rapp, Reinhard (2004). A freely available automatically generated thesaurus of related words. In *Proceedings of the 4th International Conference on Language Resources and Evaluation (LREC 2004)*, pages 395–398.

Rooth, Mats; Riezler, Stefan; Prescher, Detlef; Carroll, Glenn; Beil, Franz (1999). Inducing a semantically annotated lexicon via EM-based clustering. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics*, pages 104–111.

Schütze, Hinrich (1992). Dimensions of meaning. In *Proceedings of Supercomputing '92*, pages 787–796, Minneapolis, MN.

Schütze, Hinrich (1993). Word space. In *Proceedings of Advances in Neural Information Processing Systems 5*, pages 895–902, San Mateo, CA.

References V

Schütze, Hinrich (1995). Distributional part-of-speech tagging. In *Proceedings of the 7th Conference of the European Chapter of the Association for Computational Linguistics (EACL 1995)*, pages 141–148.

Schütze, Hinrich (1998). Automatic word sense discrimination. *Computational Linguistics*, **24**(1), 97–123.

Turney, Peter D.; Littman, Michael L.; Bigham, Jeffrey; Shnayder, Victor (2003). Combining independent modules to solve multiple-choice synonym and analogy problems. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP-03)*, pages 482–489, Borovets, Bulgaria.

Widdows, Dominic (2004). *Geometry and Meaning*. Number 172 in CSLI Lecture Notes. CSLI Publications, Stanford.