Distributional Semantic Models
Part 4: Elements of matrix algebra

Stefan Evert
with Alessandro Lenci, Marco Baroni and Gabriella Lapesa

1Friedrich-Alexander-Universität Erlangen-Nürnberg, Germany
2University of Pisa, Italy
3University of Trento, Italy
4University of Stuttgart, Germany

http://wordspace.collocations.de/doku.php/course:start

Outline
Matrix algebra
Roll your own DSM
Matrix multiplication
Association scores & normalization

Geometry
Metrics and norms
Angles and orthogonality

Dimensionality reduction
Orthogonal projection
PCA & SVD

Matrices and vectors

- $k \times n$ matrix $M \in \mathbb{R}^{k \times n}$ is a rectangular array of real numbers

$$M = \begin{bmatrix}
m_{11} & \cdots & m_{1n} \\
\vdots & \ddots & \vdots \\
m_{k1} & \cdots & m_{kn}
\end{bmatrix}$$

- Each row $m_i \in \mathbb{R}^n$ is an $n$-dimensional vector

$$m_i = (m_{i1}, m_{i2}, \ldots, m_{in})$$

- Similarly, each column is a $k$-dimensional vector $\in \mathbb{R}^k$

```r
> options(digits=3)
> M <- DSM_TermTerm$M
> M[2, ] # row vector $m_2$ for "dog"
> M[, 5] # column vector for "important"
```
Matrices and vectors

- Vector \( x \in \mathbb{R}^n \) as single-row or single-column matrix
  - \( x = x^T = n \times 1 \) matrix ("vertical")
  - \( x^T = 1 \times n \) matrix ("horizontal")
- Transposition operator, \( T \), swaps rows & columns of matrix
- We need vectors \( r \in \mathbb{R}^k \) and \( c \in \mathbb{R}^n \) of marginal frequencies
- Notation for cell \( ij \) of co-occurrence matrix:
  - \( m_{ij} = O \) ... observed co-occurrence frequency
  - \( r_i = R \) ... row marginal (target)
  - \( c_j = C \) ... column marginal (feature)
  - \( N \) ... sample size

```r
> r <- DSM_TermTerm$rows$f
> c <- DSM_TermTerm$cols$f
> N <- DSM_TermTerm$globals$N
> t(r) # "horizontal" vector
> t(t(r)) # "vertical" vector
```

The outer product

- Compute matrix \( E \in \mathbb{R}^{k \times n} \) of expected frequencies
  \[
e_{ij} = \frac{r_i c_j}{N}
\]
  i.e. \( r[i] \times c[j] \) for each cell \( ij \)
- This is the outer product of \( r \) and \( c \)
  \[
  \begin{bmatrix}
  r_1 \\
  \vdots \\
  r_k
  \end{bmatrix}
  \begin{bmatrix}
  c_1 & c_2 & \cdots & c_n
  \end{bmatrix}
  =
  \begin{bmatrix}
  r_1 c_1 & r_1 c_2 & \cdots & r_1 c_n \\
  \vdots & \vdots & \cdots & \vdots \\
  r_k c_1 & r_k c_2 & \cdots & r_k c_n
  \end{bmatrix}
  \]
- The inner product of \( x, y \in \mathbb{R}^n \) is the sum \( x_1 y_1 + \ldots + x_n y_n \)

```r
> outer(r, c) / N
```

Scalar operations

- Scalar operations perform the same transformation on each element of a vector or matrix, e.g.
  - add / subtract fixed shift \( \mu \in \mathbb{R} \)
  - multiply / divide by fixed factor \( \sigma \in \mathbb{R} \)
  - apply function (\( \log, \sqrt{\cdot}, \ldots \)) to each element
- Allows efficient processing of large sets of values
- Element-wise binary operators on matching vectors / matrices
  - \( x + y \) = vector addition
  - \( x \odot y \) = element-wise multiplication (Hadamard product)

```r
> log(M + 1) # discounted log frequency weighting
> (M["cause", ] + M["effect", ]) / 2 # centroid vector
```
Matrix multiplication

\[
\begin{bmatrix}
  a_{ij} \\
  \vdots \\
  a_{in}
\end{bmatrix}
= 
\begin{bmatrix}
  b_{i1} & \cdots & b_{in}
\end{bmatrix}
\cdot 
\begin{bmatrix}
  c_{1j} \\
  \vdots \\
  c_{nj}
\end{bmatrix}
\]

\[A = B \cdot C\]

\((k \times m) (k \times n) (n \times m)\)

- \(B\) and \(C\) must be conformable (in dimension \(n\))
- Element \(a_{ij}\) is the inner product of the \(i\)-th row of \(B\) and the \(j\)-th column of \(C\)

\[a_{ij} = b_{i1}c_{1j} + \cdots + b_{in}c_{nj} = \sum_{r=1}^{n} b_{ir}c_{rj}\]

Transposition and multiplication

- The transpose \(A^T\) of a matrix \(A\) swaps rows and columns:

\[A^T = \begin{bmatrix}
  a_1 & a_2 & a_3 \\
  b_1 & b_2 & b_3
\end{bmatrix}
= 
\begin{bmatrix}
  a_1 & b_1 \\
  a_2 & b_2 \\
  a_3 & b_3
\end{bmatrix}\]

- Properties of the transpose:
  - \((A + B)^T = A^T + B^T\)
  - \((\lambda A)^T = \lambda (A^T) =: \lambda A^T\)
  - \((A \cdot B)^T = B^T \cdot A^T\) [note the different order of \(A\) and \(B\)]
  - \(I^T = I\)
- \(A\) is called symmetric iff \(A^T = A\)
- Symmetric matrices have many special properties that will become important later (e.g. eigenvalues)

Some properties of matrix multiplication

- Associativity: \(A(BC) = (AB)C =: ABC\)
- Distributivity: \(A(B + B') = AB + AB'\)
- \((A + A')B = AB + A'B\)
- Scalar multiplication: \((\lambda A)B = A(\lambda B) = \lambda(AB) =: \lambda AB\)
- Not commutative in general: \(AB \neq BA\)
- The \(k\)-dimensional square-diagonal identity matrix

\[I_k := \begin{bmatrix}
  1 & \cdots & 0 \\
  \vdots & \ddots & \vdots \\
  0 & \cdots & 1
\end{bmatrix}\]

with \(I_k \cdot A = A \cdot I_n = A\)

The outer product as matrix multiplication

- The outer product is a special case of matrix multiplication

\[E = \frac{1}{n} (r \cdot c^T)\]

- The other special case is the inner product

\[x^T y = \sum_{i=1}^{n} x_i y_i\]

- NB: \(x \cdot x\) and \(x^T \cdot x^T\) are not conformable

### three ways to compute the matrix of expected frequencies

- \(E \leftarrow \text{outer}(r, c) / N\)
- \(E \leftarrow (r \%*\% t(c)) / N\)
- \(E \leftarrow \text{tcrossprod}(r, c) / N\)
- \(E\)
Computing association scores

- Association scores = element-wise combination of $M$ and $E$, e.g. for (pointwise) Mutual Information
  
  $$S = \log_2(M \odot E)$$

- $\odot$ = element-wise division similar to Hadamard product $\odot$

- For sparse AMs such as PPMI, we need to compute $\max \{s_{ij}, 0\}$ for each element of the scored matrix $S$

```r
> log2(M / E)
> S <- pmax(log2(M / E), 0) # not max()!
> S
```

Normalizing vectors

- Compute Euclidean norm of vector $x \in \mathbb{R}^n$:
  
  $$\|x\|_2 = \sqrt{x_1^2 + \ldots + x_n^2}$$

- Normalized vector $\|x_0\|_2 = 1$ by scalar multiplication:
  
  $$x_0 = \frac{1}{\|x\|_2} x$$

```r
> x <- S[2,]
> b <- sqrt(sum(x ^ 2)) # Euclidean norm of x
> x0 <- x / b           # normalized vector
> sqrt(sum(x0 ^ 2))
```

Normalizing matrix rows

- Compute vector $b \in \mathbb{R}^k$ of norms of row vectors of $S$

- Can you find an elegant way to multiply each row of $S$ with the corresponding normalization factor $b_i^{-1}$?

- Multiplication with diagonal matrix $D_b^{-1}$

  $$S_0 = D_b^{-1} \cdot S$$

  $$S_0 = \begin{bmatrix}
  b_1^{-1} & \cdots & s_{1n} \\
  \vdots & \ddots & \vdots \\
  b_k^{-1} & \cdots & s_{kn}
  \end{bmatrix} \cdot \begin{bmatrix}
  s_{11} & \cdots & s_{1n} \\
  \vdots & \ddots & \vdots \\
  s_{k1} & \cdots & s_{kn}
  \end{bmatrix}$$
Normalizing matrix rows

- Compute vector \( b \in \mathbb{R}^k \) of norms of row vectors of \( S \)
- Can you find an elegant way to multiply each row of \( S \) with the corresponding normalization factor \( b_i^{-1} \)?
- Multiplication with diagonal matrix \( D_b^{-1} \)

\[ S_0 = D_b^{-1} \cdot S \]

```r
> b <- sqrt(rowSums(S^2)) # more efficient
> S0 <- normalize.rows(S, method="euclidean") # the easy way
```

Outline

- Matrix algebra
- Roll your own DSM
- Matrix multiplication
- Association scores & normalization

Geometry

- Metrics and norms
- Angles and orthogonality
- Dimensionality reduction
- Orthogonal projection
- PCA & SVD

Norm: a measure of length

- A general norm \( \|u\| \) for the length of a vector \( u \) must satisfy the following axioms:
  - \( \|u\| > 0 \) for \( u \neq 0 \)
  - \( \|\lambda u\| = |\lambda| \cdot \|u\| \) (homogeneity)
  - \( \|u + v\| \leq \|u\| + \|v\| \) (triangle inequality)
- Every norm induces a metric

\[ d(u, v) := \|u - v\| \]

with two desirable properties
- translation-invariant: \( d(u + x, v + x) = d(u, v) \)
- scale-invariant: \( d(\lambda u, \lambda v) = |\lambda| \cdot d(u, v) \)
- \( d_p(u, v) \) is induced by the Minkowski norm for \( p \geq 1 \):

\[ \|u\|_p := (|u_1|^p + \cdots + |u_n|^p)^{1/p} \]
Norm: a measure of length

- Visualisation of norms in $\mathbb{R}^2$ by plotting unit circle, i.e. points $u$ with $\|u\| = 1$
- Here: $p$-norms $\|\cdot\|_p$ for different values of $p$
- Triangle inequality $\iff$ unit circle is convex
- This shows that $p$-norms with $p < 1$ would violate the triangle inequality

Consequence for DSM: $p \ll 2$ sensitive to small differences in many coordinates, $p \gg 2$ to larger differences in few coord.

Euclidean norm & inner product

- The Euclidean norm $\|u\|_2 = \sqrt{u^T u}$ is special because it can be derived from the inner product:
  \[ x^T y = x_1 y_1 + \cdots + x_n y_n \]
- The inner product is a positive definite and symmetric bilinear form with an important geometric interpretation:
  \[ \cos \phi = \frac{u^T v}{\|u\|_2 \cdot \|v\|_2} \]
  for the angle $\phi$ between vectors $u, v \in \mathbb{R}^n$
  - the value $\cos \phi$ is known as the cosine similarity measure
  - In particular, $u$ and $v$ are orthogonal iff $u^T v = 0$

Cosine similarity in R

- Cosine similarities can be computed very efficiently if vectors are pre-normalized, so that $\|u\|_2 = \|v\|_2 = 1$
- just need all inner products $m_i^T m_j$ between row vectors of $M$

\[
(M \cdot M^T)_{ij} = m_i^T m_j
\]

# cosine similarities for row-normalized matrix:
> sim <- tcrossprod(S0)
> angles <- acos(pmin(sim, 1)) * (180 / pi)
We can now prove that Euclidean distance and cosine similarity are equivalent: if vectors are normalised ($\|u\|_2 = \|v\|_2 = 1$), both lead to the same neighbour ranking

\[
d_2(u, v) = \sqrt{\|u - v\|_2} = \sqrt{(u - v)^T(u - v)}
\]

\[
= \sqrt{u^Tu + v^Tv - 2u^Tv}
\]

\[
= \sqrt{\|u\|_2 + \|v\|_2 - 2u^Tv}
\]

\[
= \sqrt{2 - 2\cos \phi}
\]

$\Rightarrow d_2(u, v)$ is a monotonically increasing function of $\phi$

- A linear **subspace** $B \subseteq \mathbb{R}^n$ of rank $r \leq n$ is spanned by a set of $r$ linearly independent basis vectors

  \[
  B = \{b_1, \ldots, b_r\}
  \]

- Every point $u$ in the subspace is a unique linear combination of the basis vectors

  \[
  u = x_1b_1 + \ldots + x_rb_r
  \]

  with coordinate vector $x \in \mathbb{R}^r$

- Basis matrix $V$ with column vectors $b_i$:

  \[
  u = Vx
  \]

- Particularly convenient with orthonormal basis:

  \[
  \|b_i\|_2 = 1
  \]

  \[
  b_i^Tb_j = 0 \quad \text{for } i \neq j
  \]

- Corresponding basis matrix $V$ is (column)-**orthogonal**

  \[
  V^TV = I_r
  \]
The mathematics of projections

- 1-d subspace spanned by basis vector $\|b\|_2 = 1$
- For any point $u$, we have
  $$\cos \varphi = \frac{b^T u}{\|b\|_2 \cdot \|u\|_2}$$
- Trigonometry: coordinate of point on the line is $x = \|u\|_2 \cdot \cos \varphi = b^T u$
- The projected point in original space is then given by
  $$b \cdot x = b(b^T u) = (bb^T)u = Pu$$

where $P$ is a projection matrix of rank 1

The mathematics of projections

- Decomposition also applies to squared Euclidean distances:
  $$\|u - v\|^2 = \|Pu - Pv\|^2 + \|Qu - Qv\|^2$$
- $\|Qu\|^2$ as measure for “loss” resulting from projection:
  $$\frac{\|Qu\|^2}{\|u\|^2} = 1 - \frac{\|Pu\|^2}{\|u\|^2} = 1 - R^2$$

where $R^2$ is the proportion of vector length “preserved” by $P$, similar to the explained variance $R^2$ in linear regression

Optimal projections and subspaces

- Optimal subspace maximises $R^2$ across a data set $M$, which is now specified in terms of row vectors $m_i^T$:
  $$x_i^T = m_i^T V$$
  $$m_i^T P = m_i^T VV^T$$
  $$X = MV$$
  $$MP = MVV^T$$

- Our “faithfulness” measure is thus given by
  $$R^2 = \frac{\sum_{i=1}^k |m_i^T P|^2}{\sum_{i=1}^k |m_i|^2} = \frac{|MP|^2}{|M|^2}$$

with the (squared) Frobenius norm
  $$|M|^2 = \sum_{i,j} (m_{ij})^2 = \sum_{i=1}^k |m_i|^2$$

For an orthogonal basis matrix $V$ with columns $b_1, \ldots, b_r$, the projection into the rank-$r$ subspace $B$ is given by

$$Pu = \left( \sum_{i=1}^r b_i b_i^T \right) u = VV^T u$$

and its subspace coordinates are $x = Vu$
Optimal projections and subspaces

- For a centered data set with \( \sum_i m_i = 0 \), the Frobenius norm corresponds to the average (squared) distance between points

\[
\sum_{i,j=1}^{k} ||m_i - m_j||^2 = \sum_{i,j=1}^{k} (||m_i||^2 + ||m_j||^2 - 2m_i^T m_j) = \sum_{i=1}^{k} ||M||^2_F + \sum_{j=1}^{k} ||M||^2_F - 2 \sum_{i=1}^{k} m_i^T (\sum_{j=1}^{k} m_j) = 2k \cdot ||M||^2_F.
\]

- “loss” of distances: \( \sum_{i,j=1}^{k} ||(m_i - m_j)Q||^2 = 2k \cdot ||MQ||^2_F \)

\( R^2 \) is a measure of how well distances are preserved.

Singular value decomposition

- Fundamental result of matrix algebra: \textbf{singular value decomposition} (SVD) factorises any matrix \( M \) into

\[
M = U \Sigma V^T
\]

where \( U \) and \( V \) are orthogonal and \( \Sigma \) is a diagonal matrix of \textbf{singular values} \( \sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_m > 0 \)

\[
\begin{bmatrix} n \\ k \\ m \\ \end{bmatrix} = \begin{bmatrix} n \\ k \\ m \\ \end{bmatrix} \begin{bmatrix} \sigma_1 & m \\ m & \ddots & \vdots \\ m & \ddots & \Sigma \\ \sigma_1 & \cdots & \sigma_m \\ \end{bmatrix} \begin{bmatrix} n \\ m \\ \end{bmatrix}
\]

- \( m \leq \min\{k,n\} \) is the inherent dimensionality (rank) of \( M \)

- Columns \( a_i \) of \( U \) are called left singular vectors, columns \( b_i \) of \( V \) (= rows of \( V^T \)) are right singular vectors

- Recall the “outer product” view of matrix multiplication:

\[
UV^T = \sum_{i=1}^{m} a_i b_i^T
\]

- Hence the SVD corresponds to a sum of rank-1 components

\[
M = U \Sigma V^T = \sum_{i=1}^{m} \sigma_i a_i b_i^T
\]
Singular value decomposition

- Key property of SVD: the first $r$ components give the best rank-$r$ approximation to $M$ with respect to the Frobenius norm, i.e. they minimize the loss

$$\|U_r, \Sigma_r, V_r^T - M\|_F^2 = \|M_r - M\|_F^2$$

- **Truncated SVD**
  - $U_r, V_r$ = first $r$ columns of $U, V$
  - $\Sigma_r$ = diagonal matrix of first $r$ singular values
  - It can be shown that

$$\|M\|_F^2 = \sum_{i=1}^m \sigma_i^2 \quad \text{and} \quad \|M_r\|_F^2 = \sum_{i=1}^r \sigma_i^2$$

SVD dimensionality reduction

- Columns of $V_r$ form an orthonormal basis of the optimal rank-$r$ subspace because

$$MP = MV_r^T = U \Sigma V_r^T V_r^T = U_r \Sigma_r V_r^T = M_r$$

- **Dimensionality reduction** uses the subspace coordinates

$$MV_r = U_r \Sigma_r$$

- If $M$ is centered, this also gives the best possible preservation of pairwise distances $\Rightarrow$ **principal component analysis (PCA)**

  - but centering is usually omitted in order to preserve sparseness, so SVD captures vector lengths rather than distances

Scaling SVD dimensions

- Singular values $\sigma_i$ can be seen as weighting of the latent dimensions, which determines their contribution to

$$\|MV_r\|_F = \sigma_1^2 + \ldots + \sigma_r^2$$

- Weighting can be adjusted by **power scaling** of the singular values:

$$U_r \Sigma_p = \begin{bmatrix} \vdots & \vdots & \vdots \\ \sigma_p a_1 & \cdots & \sigma_p a_r \\ \vdots & \vdots & \vdots \end{bmatrix}$$

  - $p = 1$: normal SVD projection
  - $p = 0$: dimension weights equalized
  - $p = 2$: more weight given to first latent dimensions

  - Other weighting schemes possible (e.g. skip first dimensions)