Corpus Structure Discovery

Unstructured Data

CS 4417B/9117/9647

The University of Western Ontario

Latent Semantic Analysis

- SVD
  - Interpretation as compression
- Decomposition
  - Documents as sums of term-collections
  - Terms as sums of document-collections
- Representations
  - Term-term similarity
  - Document-document similarity
  - Term-document similarity
- Demo

Term-Document Matrix

```
        Docs
Terms     11224 15871 15875 17396 17953 2521 5214 5985 6657 7589
  bank        2     0     0    10     3    1    0    2    0   10
  billion     0     0     0     0     2    2    0    4    1    6
  compani     0     0     0     0     0    0    1    8    0    0
  dlrs        1     0    10     0     0    2    0    1    1    3
  mln         1     0    29     0     0    4   12    1    8    0
  pct         3     0     0     8     0    1    2   13    0   10
  reuter      6     1     1     1     1    2    2    0    2    1
  said       13     0     0    28    18   22   20   24   13    5
  will        4     1     0     2     2    8    1   14    2    3
  year        1     0     3     1     1   10   19    5   12    5
```

Each column vector represents a document, bag-of-words.

Term-Document Matrix

Docs Terms	11224	12701	14635	15871	15875	17474	17497	17953	5214	6657
bank	5.41	0.00	0.00	0.00	0.00	8.11	0.00	8.11	0.00	0.00
billion	0.00	0.00	2.73	0.00	0.00	2.73	10.92	5.46	0.00	2.73
cts	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
dlrs	1.50	5.98	1.50	0.00	14.95	0.00	5.98	0.00	0.00	1.50
mln	1.38	1.38	0.00	0.00	39.96	4.13	11.02	0.00	16.53	11.02
pct	5.56	3.71	1.85	0.00	0.00	11.12	1.85	0.00	3.71	0.00
said	6.39	4.43	1.97	0.00	0.00	5.41	3.44	8.85	9.84	6.39
share	7.27	4.85	0.00	0.00	0.00	16.97	0.00	2.42	0.00	0.00
will	6.37	6.37	4.78	1.59	0.00	6.37	4.78	3.19	1.59	3.19
year	1.67	3.34	8.35	0.00	5.01	20.04	8.35	1.67	31.73	20.04

Each column vector represents a document, TF-IDF.

Will call this matrix M throughout these slides.

Vector representations

- Previously
  - Documents represented as vectors
  - Can represent similarity with dot prod or cosine
- Idea in This lesson
  - Compressing the TDM
    - Saves space
    - Reveals patterns
    - Improves retrieval
  - Bonus idea:
    - Each row represents a term! 🙄

Factoring the TDM

	D1	D2	D3	D4	D5	D6
cat	1	0	1	1	1	0
dog	1	0	1	1	1	0
horse	1	0	1	1	1	0
apple	0	1	0	0	1	1
orange	0	1	0	0	1	1

V^T	D1	D2	D3	D4	D5	D6
W1	1	0	1	1	1	0
W2	0	1	0	0	1	1

U	T1	T2
cat	1	0
dog	1	0
horse	1	0
apple	0	1
orange	0	1

1	0	1	1	1	0
1	0	1	1	1	0
1	0	1	1	1	0
0	1	0	0	1	1
0	1	0	0	1	1

V^T	D1	D2	D3	D4	D5	D6
W1	1	0	1	1	1	0
W2	0	1	0	0	1	1

U	T1	T2
cat	1	0
dog	1	0
horse	1	0
apple	0	1
orange	0	1

1	0	1	1	1	0
1	0	1	1	1	0
1	0	1	1	1	0
0	1	0	0	1	1
0	1	0	0	1	1

V^T	D1	D2	D3	D4	D5	D6
W1	1	0	1	1	1	0
W2	0	1	0	0	1	1

U	T1	T2
cat	1	0
dog	1	0
horse	1	0
apple	0	1
orange	0	1

1	0	1	1	1	0
1	0	1	1	1	0
1	0	1	1	1	0
0	1	0	0	1	1
0	1	0	0	1	1

V^T	D1	D2	D3	D4	D5	D6
W1	1	0	1	1	1	0
W2	0	1	0	0	1	1

U	T1	T2
cat	1	0
dog	1	0
horse	1	0
apple	0	1
orange	0	1

1	0	1	1	1	0
1	0	1	1	1	0
1	0	1	1	1	0
0	1	0	0	1	1
0	1	0	0	1	1

V^T	D1	D2	D3	D4	D5	D6
W1	1	0	1	1	1	0
W2	0	1	0	0	1	1

U	T1	T2
cat	1	0
dog	1	0
horse	1	0
apple	0	1
orange	0	1

1	0	1	1	1	0
1	0	1	1	1	0
1	0	1	1	1	0
0	1	0	0	1	1
0	1	0	0	1	1

V^T	D1	D2	D3	D4	D5	D6
W1	1	0	1	1	1	0
W2	0	1	0	0	1	1

U	T1	T2
cat	1	0
dog	1	0
horse	1	0
apple	0	1
orange	0	1

1	0	1	1	1	0
1	0	1	1	1	0
1	0	1	1	1	0
0	1	0	0	1	1
0	1	0	0	1	1

V^T	D1	D2	D3	D4	D5	D6
W1	1	0	1	1	1	0
W2	0	1	0	0	1	1

U	T1	T2
cat	1	0
dog	1	0
horse	1	0
apple	0	1
orange	0	1

1	0	1	1	1	0
1	0	1	1	1	0
1	0	1	1	1	0
0	1	0	0	1	1
0	1	0	0	1	1

V^T	D1	D2	D3	D4	D5	D6
W1	1	0	1	1	1	0
W2	0	1	0	0	1	1

U	T1	T2
cat	1	0
dog	1	0
horse	1	0
apple	0	1
orange	0	1

1	0	1	1	1	0
1	0	1	1	1	0
1	0	1	1	1	0
0	1	0	0	1	1
0	1	0	0	1	1

Singular Value Decomposition

Singular Value Decomposition

$M_{m \times n} \approx \dot{M}_{m \times n} = U_{m \times p} \, \Sigma_{p \times p} \, V^\mathsf{T}_{p \times n}$

Such that squared reconstruction error

$(\|M_{m \times n} - \dot{M}_{m \times n}\|_\mathsf{F})^2$

is as small as possible.

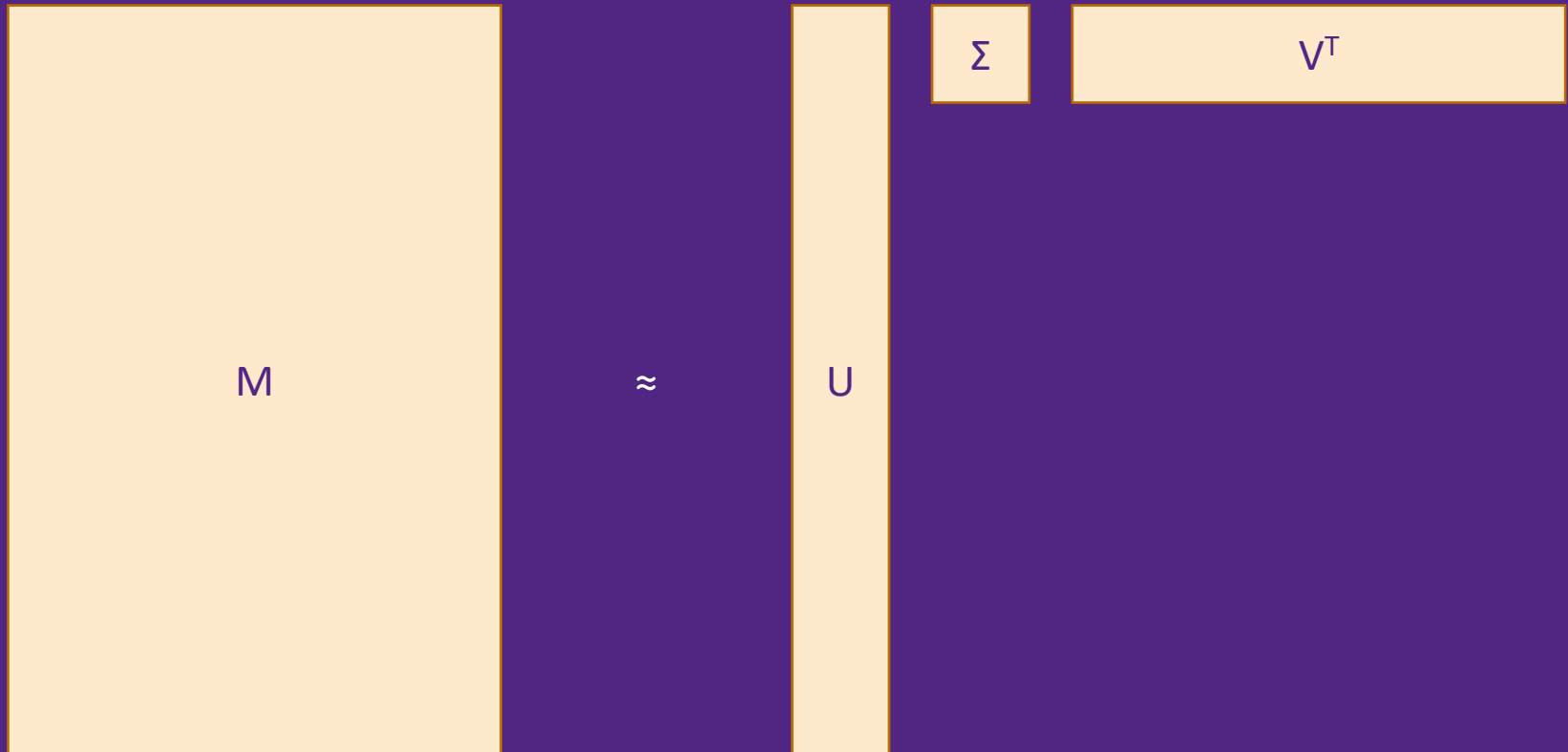Elements of Σ are called *singular values*

$V^\mathsf{T}_{p \times n} \, V_{n \times p} = I_{p \times p}$

$U^\mathsf{T}_{p \times m} \, U_{m \times p} = I_{p \times p}$

Singular Value Decomposition

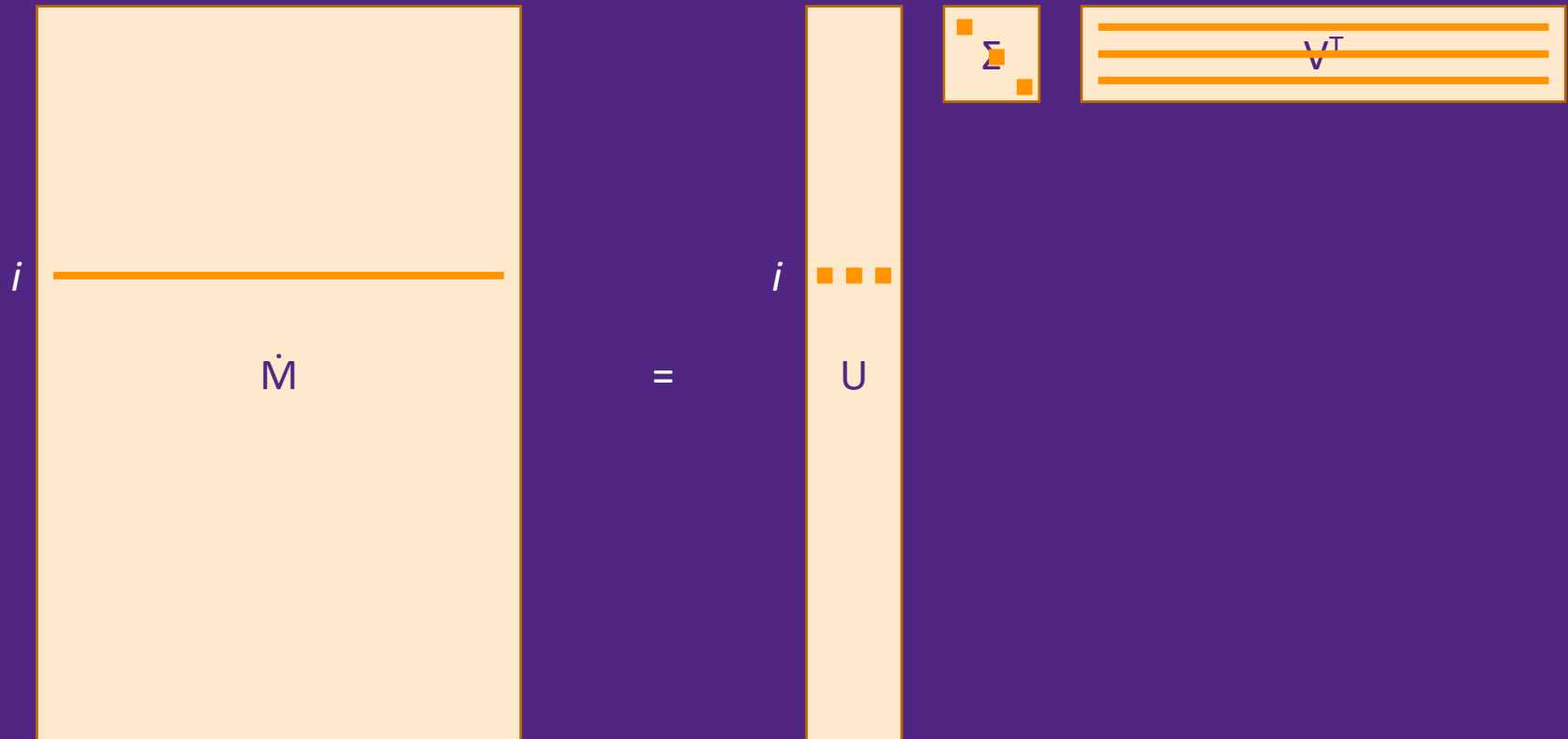$M_{m \times n} \approx \dot{M}_{m \times n} = U_{m \times p} \; \Sigma_{p \times p} \; V^T_{\;p \times n}$

Such that squared reconstruction error

$(\|M_{m \times n} - \dot{M}_{m \times n}\|_F)^2$

is as small as possible.

Elements of Σ are called *singular values*
$V^T_{\;p \times n} \; V_{n \times p} = I_{p \times p}$
$U^T_{\;p \times m} \; U_{m \times p} = I_{p \times p}$

Singular Value Decomposition

$$M_{m \times n} \approx \dot{M}_{m \times n} = U_{m \times p} \, \Sigma_{p \times p} \, V^T_{p \times n}$$

Such that squared reconstruction error

$$(\, \|M_{m \times n} - \dot{M}_{m \times n}\|_F \,)^2$$

is as small as possible.

Elements of Σ are called *singular values*

$$V^T_{p \times n} \, V_{n \times p} = I_{p \times p}$$
$$U^T_{p \times m} \, U_{m \times p} = I_{p \times p}$$

Singular Value Decomposition

$$M_{m \times n} \approx \dot{M}_{m \times n} = U_{m \times p} \, \Sigma_{p \times p} \, V^{T}_{p \times n}$$

Approximating a column of M

- $\dot{M}_{\cdot,j} = U \; \Sigma \; V^T_{j,\cdot}$
- *A column of Ṁ is a weighted sum of the columns of* U

$\dot{M}$

$=$

U

Σ

V^T

j

j
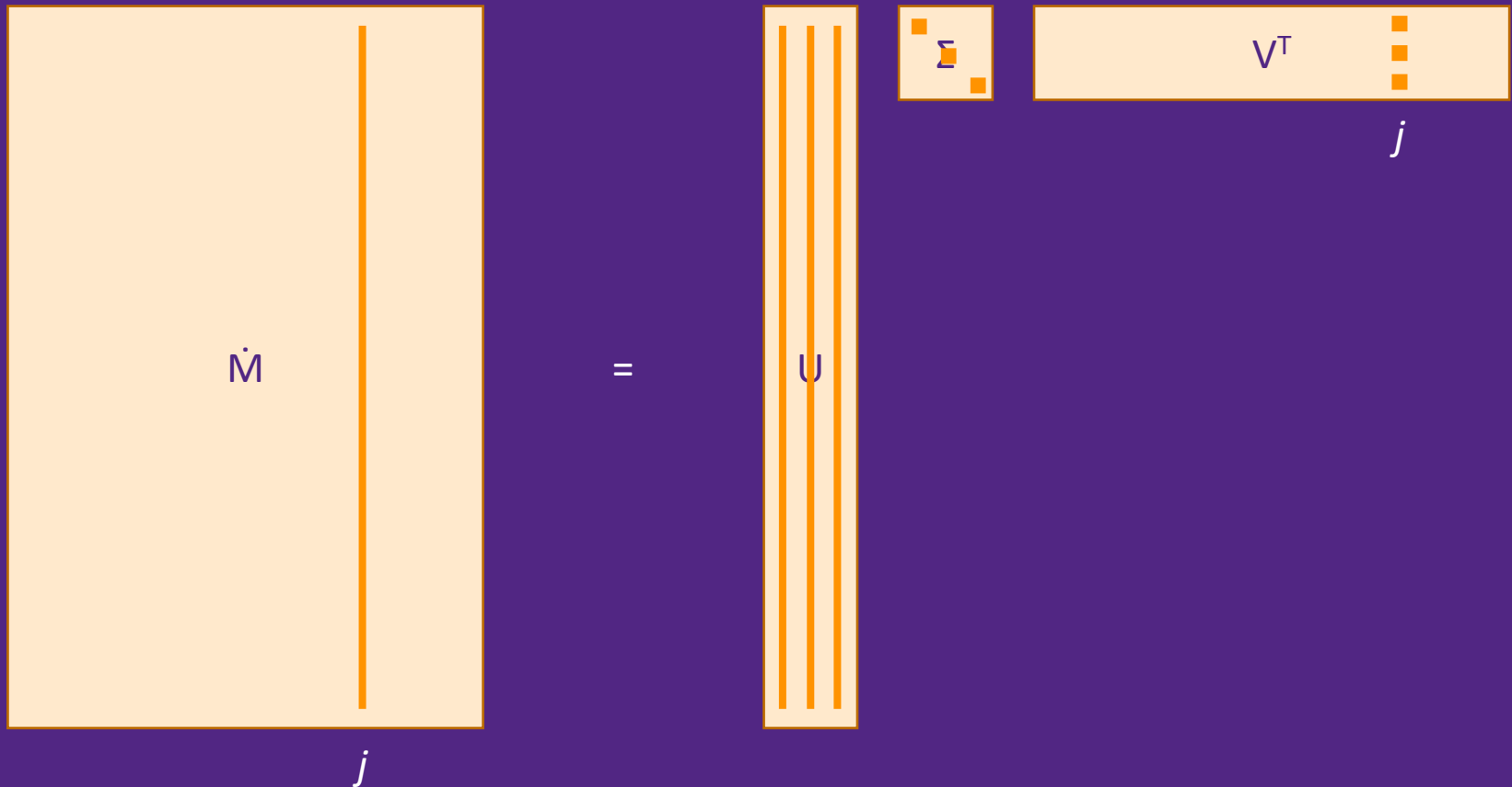
Approximating a row of M

- $\dot{M}_{i,\cdot} = U_{i,\cdot} \Sigma V^T$
- *A row of $\dot{M}$ is a weighted sum of the rows of V^T*

Latent Semantic Analysis

Latent Semantic Analysis

- Take the term-document matrix M
- Approximate it using SVD, giving

$$M_{m \times n} \approx \dot{M}_{m \times n} = U_{m \times p} \ \Sigma_{p \times p} \ V^T_{p \times n}$$

- New representations:
    - Columns of V^T represent documents using p dimensions
    - Rows of U represent terms using p dimensions
- New structure
    - Columns of U: terms that co-occur in docs (dimension m)
    - Rows of V^T: docs with similar words (dimension n)

V^T	D1	D2	D3	D4	D5	D6
W1	1	0	1	1	1	0
W2	0	1	0	0	1	1

Columns of V^T are new vector representations for each document

U	T1	T2
cat	1	0
dog	1	0
horse	1	0
apple	0	1
orange	0	1

1	0	1	1	1	0
1	0	1	1	1	0
1	0	1	1	1	0
0	1	0	0	1	1
0	1	0	0	1	1

Rows of U are new vector representations for each term

V^T	D1	D2	D3	D4	D5	D6
W1	1	0	1	1	1	0
W2	0	1	0	0	1	1

Rows of V^T identify documents that contain the same words

U	T1	T2
cat	1	0
dog	1	0
horse	1	0
apple	0	1
orange	0	1

1	0	1	1	1	0
1	0	1	1	1	0
1	0	1	1	1	0
0	1	0	0	1	1
0	1	0	0	1	1

Columns of U identify terms that co-occur in the same document

Compression

- $\dot{M}_{m \times n} = U_{m \times p} \, \Sigma_{p \times p} \, V^T_{p \times n}$

- Consider 1M documents, 1M terms. Original matrix M is a terabyte if 1 byte/entry if dense. (It isn't dense, but consider.)

- If p = 5, then we have 5M + 5 + 5M ≈ 10M

- Smaller by a factor of 100,000

A new basis for documents

- $\dot{M}_{m \times n} = U_{m \times p} \, \Sigma_{p \times p} \, V^{T}_{p \times n}$

- All approximate document vectors must be constructed from same p columns of U.

- Terms that have large weights, same sign, in the same column of U, must enter the document together.

Approximating a column of M

- $\dot{M}_{\cdot,j} = U \Sigma V^T_{\cdot,j}$
- *A column of $\dot{M}$ is a weighted sum of the columns of U*
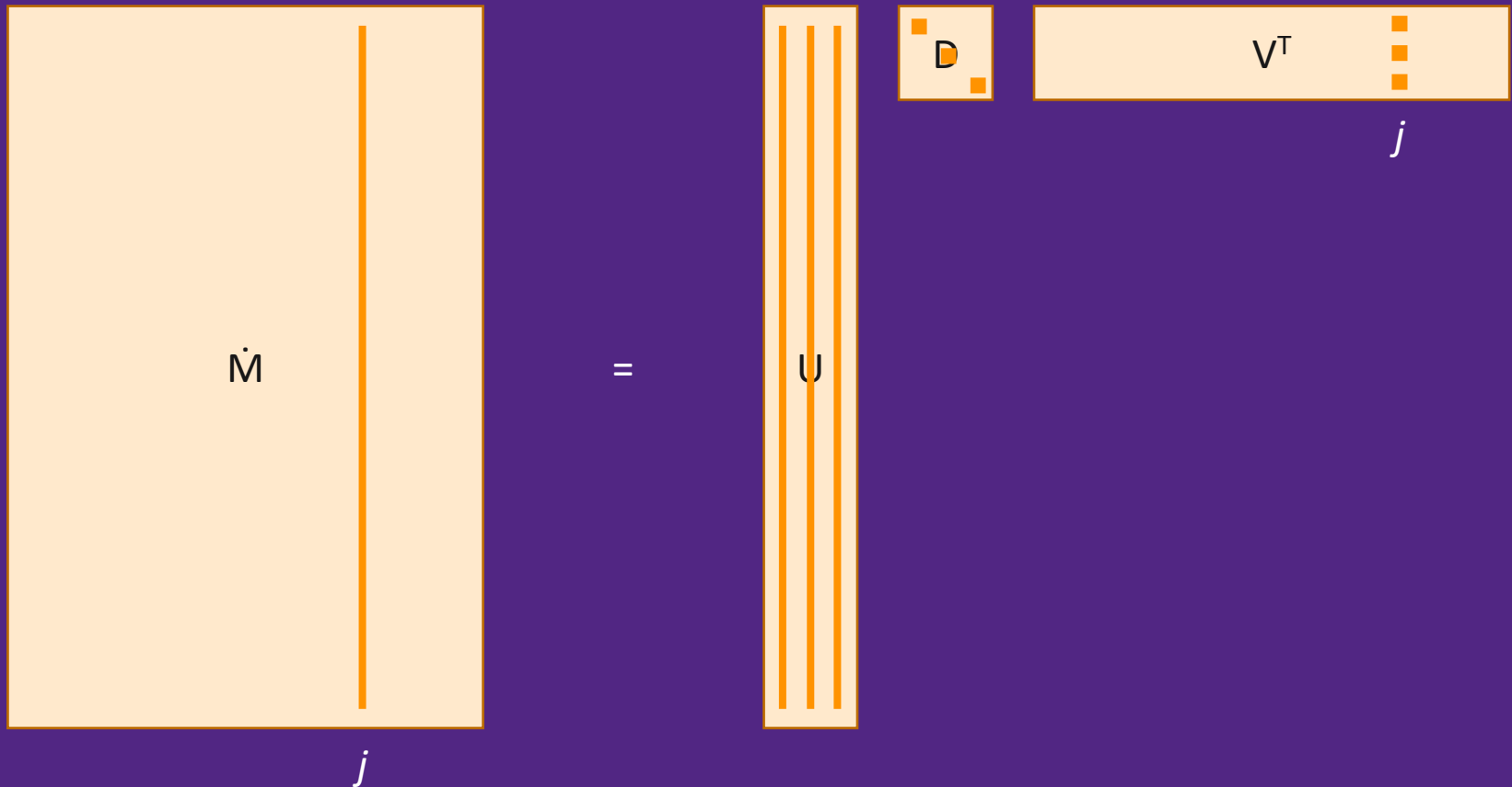
Topics from columns of U

```
share    common    stock    inc      offer    compani corp     cts
0.294    0.181     0.181    0.171    0.129    0.112   0.108    0.108
```

(This is a row of U^T; the software outputs in this format.)

Topics from U (sort of)

share	common	stock	inc	offer	compani	corp	cts
0.294	0.181	0.181	0.171	0.129	0.112	0.108	0.108

pct	bank	tonn	stg	januari	rose	februari	billion
-0.0962	-0.0971	-0.0993	-0.1021	-0.1022	-0.1060	-0.1114	-0.5970

(This is a row of U^T; the software outputs in this format.)

We sort of get 2 topics per vector; the + elements and the – elements.
Words at either end of a topic vector are "anticorrelated" in documents.

A new basis for term-distributions

- $\dot{M}_{m \times n} = U_{m \times p} \; \Sigma_{p \times p} \; V^T{}_{p \times n}$

- If $p < \min(m,n)$, then we can't adjust the elements of a row of a M independently – we must take the "document amounts" from an entire row of V at once.

- Documents that have large weights, same sign, in the same row of V^T, must contribute together to a term's row.

Approximating a row of M

- $\dot{M}_{i,\cdot} = U_{i,\cdot} \Sigma V^T$

- *A row of $\dot{M}$ is a weighted sum of the rows of V^T*
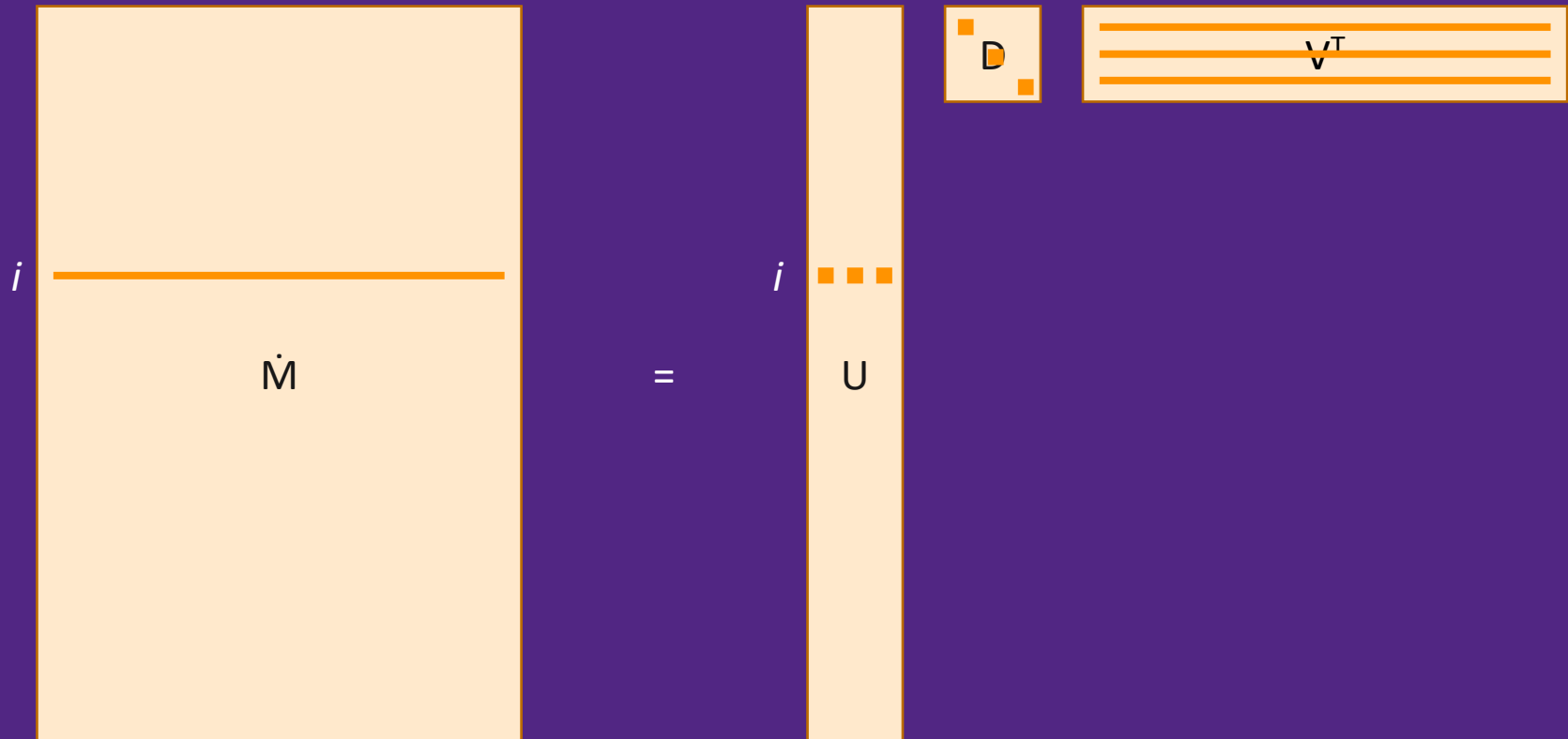
Document clusters from Vᵀ

```
  7200     1836   11504     2375   19897 ...
0.0762 0.0732 0.0702 0.0650 0.0605 ...
```

Structure from the corpus

...squared reconstruction error

$$\|M_{m \times n} - U_{m \times p}\ \Sigma_{p \times p}\ V^T_{p \times n}\|_F^2$$

is as small as possible.

- The SVD algorithm *has* to find structure in the corpus to represent it using a small number of columns of U and rows of V^T.
  - Has to find words that tend to co-occur
  - Has to find documents that tend to have similar words

- This procedure is called **Latent Semantic Analysis**
  - "Latent" because structure is hidden but discovered through the analysis. "Semantic" because the structure that is found often corresponds to groups of terms that describe similar concepts.