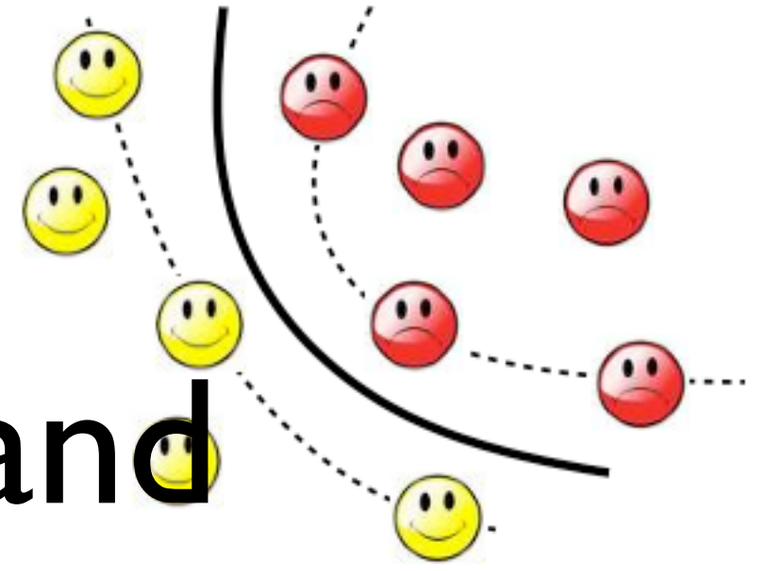
THE UNIVERSITY OF SYDNEY



Machine Learning and Data Mining

Basic Matrix Analysis and Singular Value Decomposition

Dr Tongliang Liu

What is Machine Learning?

THE UNIVERSITY OF SYDNEY

Informally: Making predictions from data

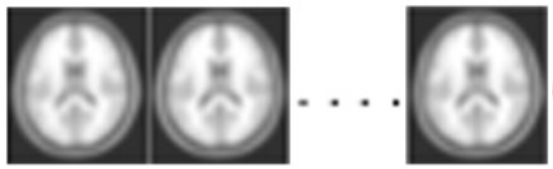
Formally: The construction of a statistical model that is an underlying distribution from which the data is drawn from.

Elements of Machine Learning

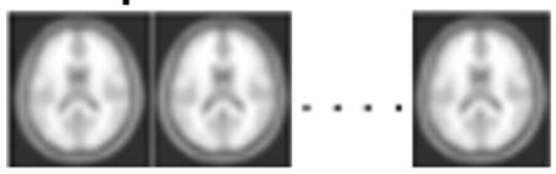
THE UNIVERSITY OF SYDNEY

Input training data

Group 1



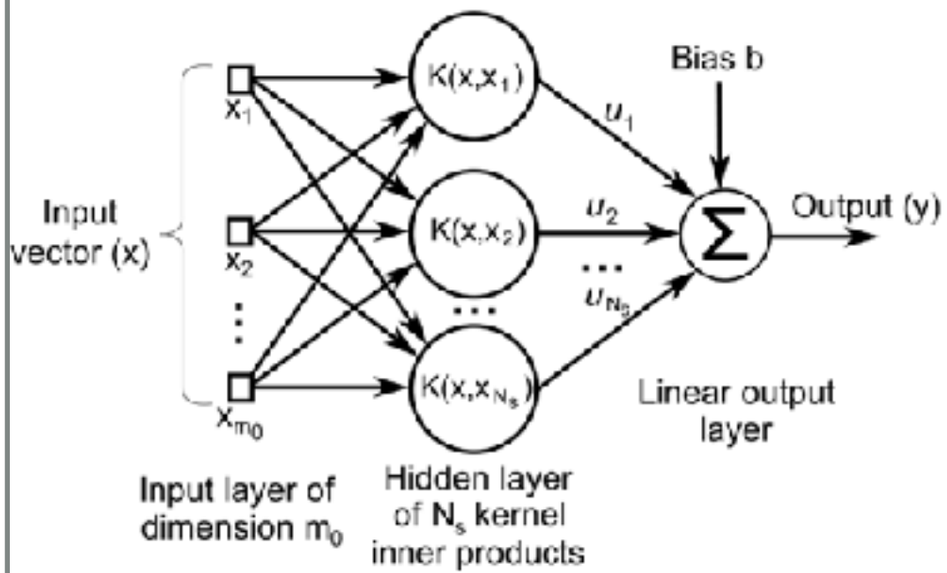
Group 2

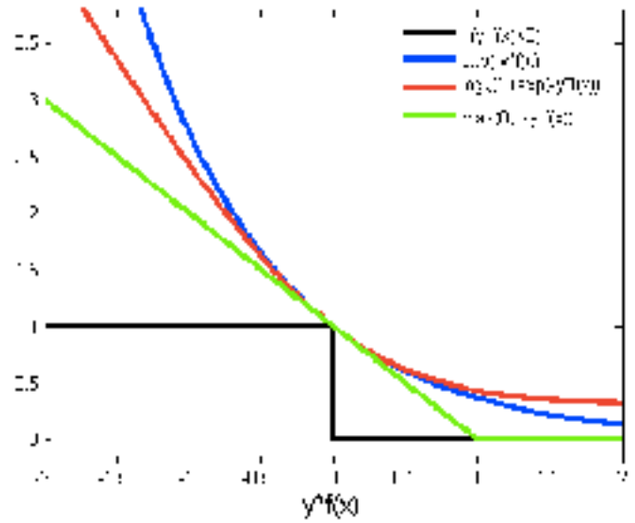


Data

Input predefined function class

Mathematical Model

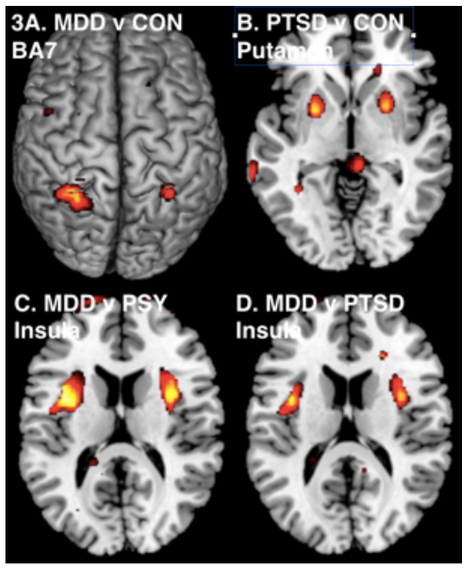




Objective function

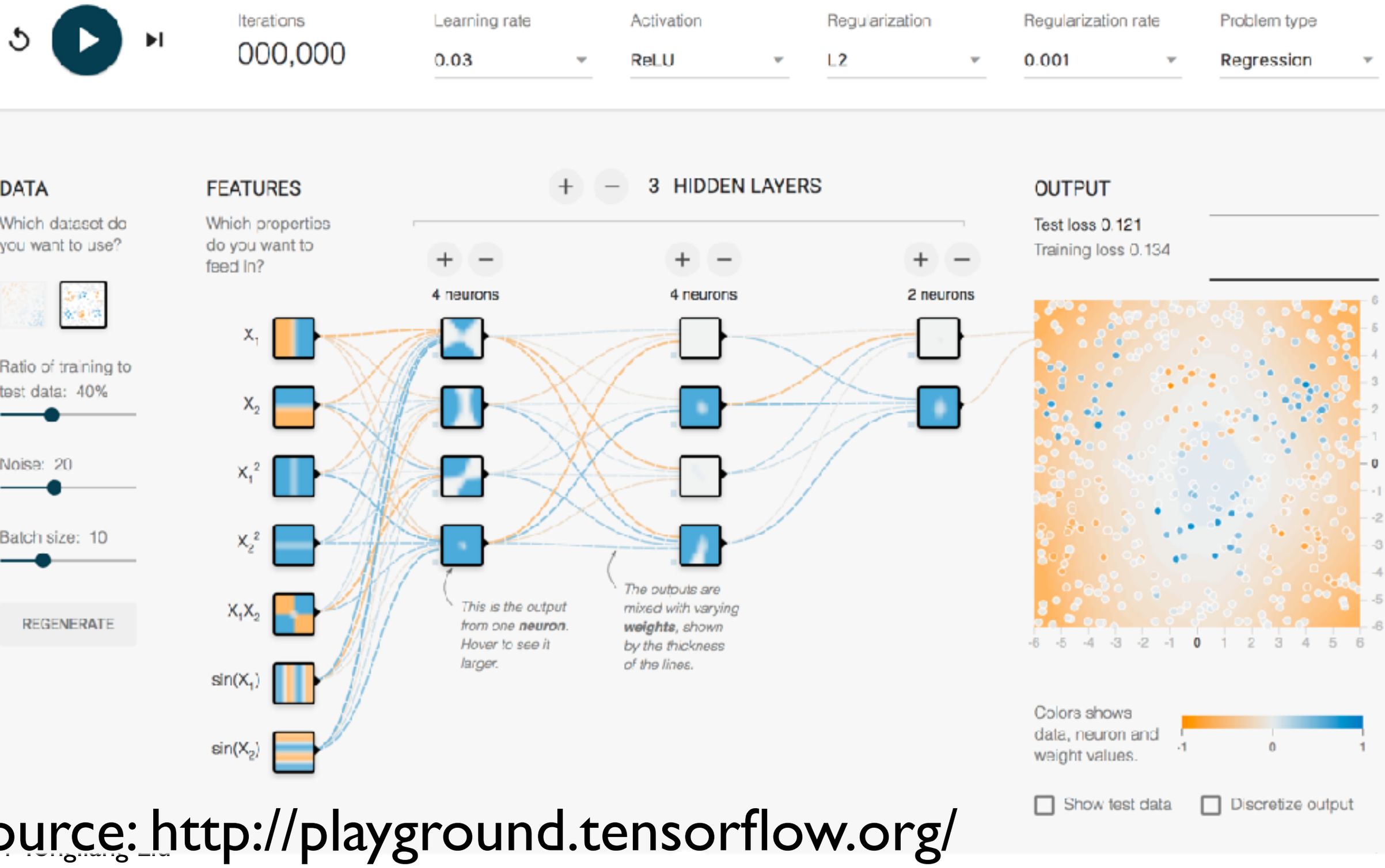
Optimisation method

Output hypothesis
Predictions/Patterns



Neural Networks

THE UNIVERSITY OF SYDNEY



Source: http://playground.tensorflow.org/

Neural Networks

Iterations
000,000

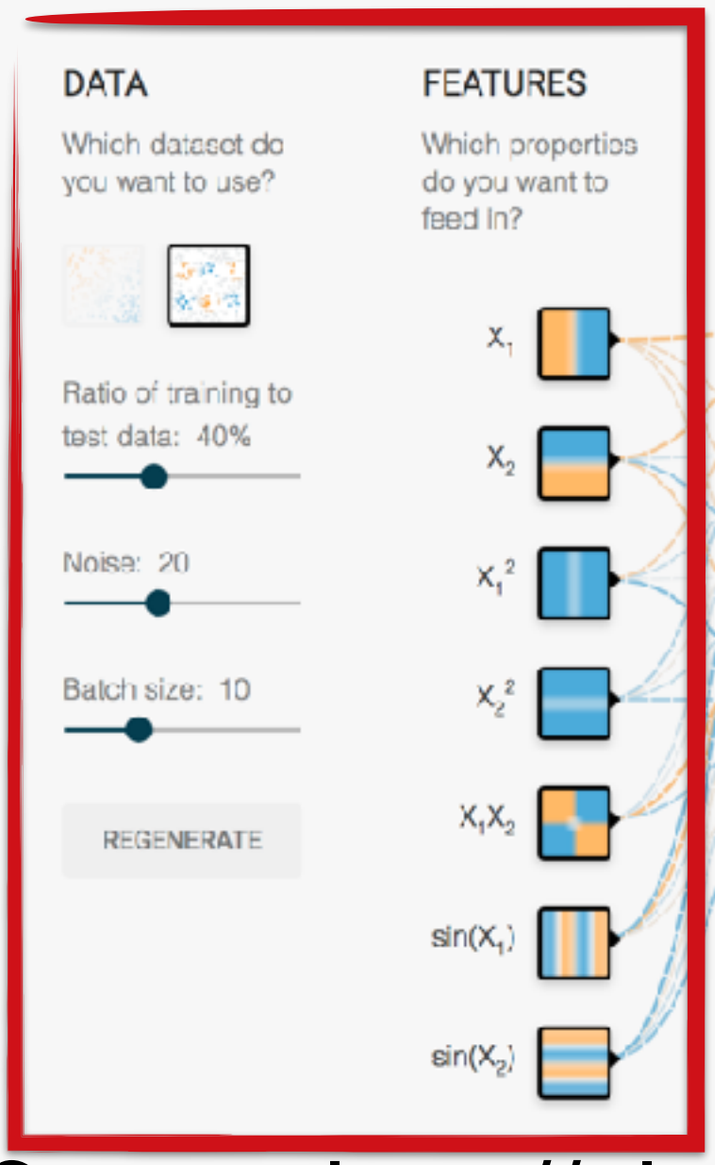
Learning rate
0.03

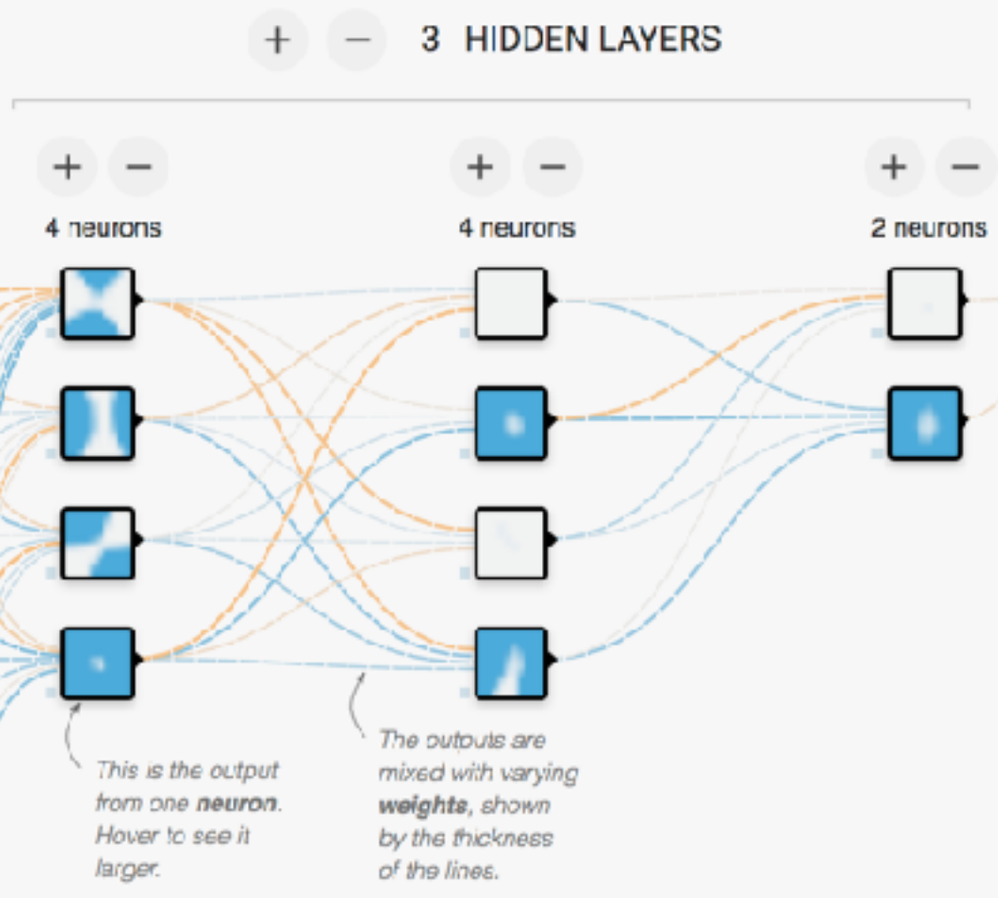
Activation
ReLU

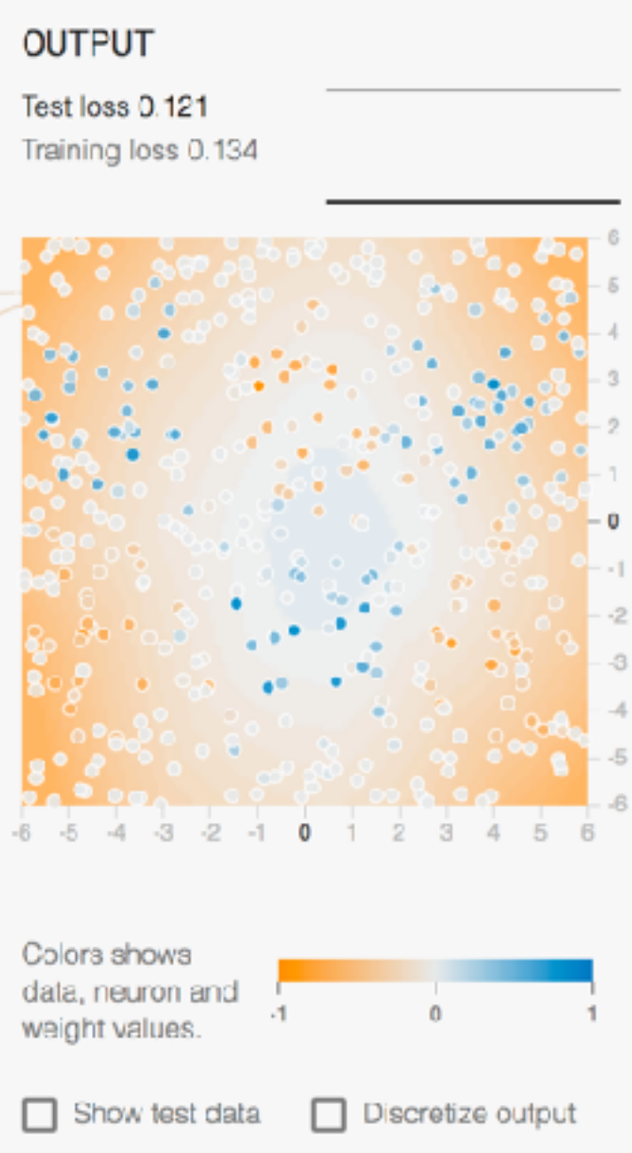
Regularization
L2

Regularization rate
0.001

Problem type
Regression







Source: http://playground.tensorflow.org/

Common representation

THE UNIVERSITY OF SYDNEY

IMAGE/ VIDEO

TEXT/ COMMENT

TIME SERIES

SYSTEM LOGS

NETWORK

TABULAR/ RATING

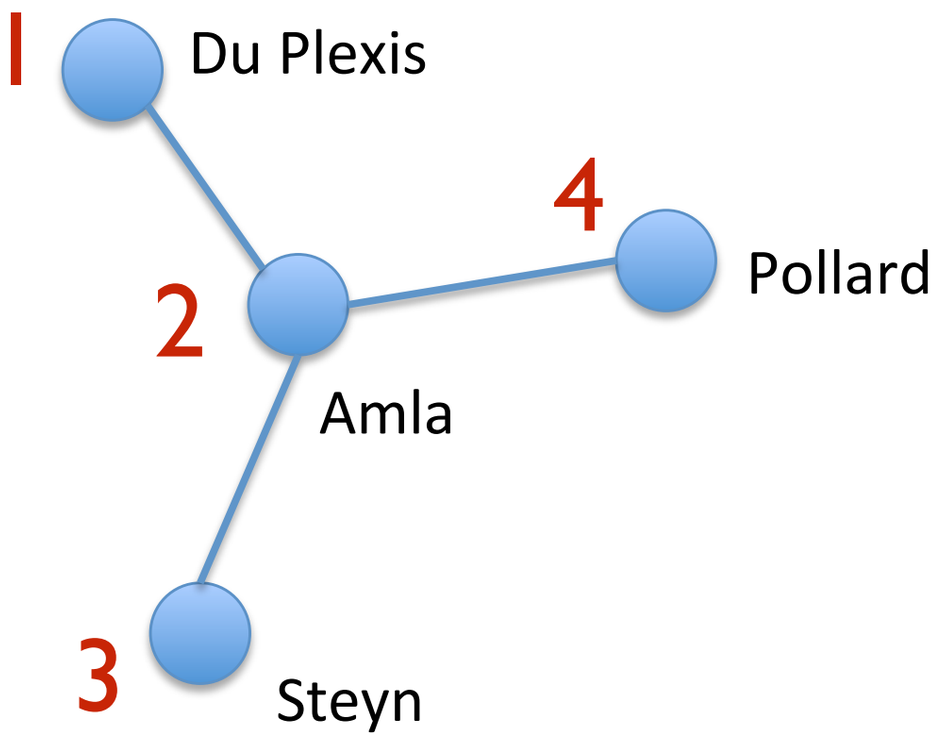
Is there a common way to represent data of different modalities ?

Text to matrix

- Document- Word Matrix
- Document 1:“AACCBBAAA”
- Document 2:“CCAABBDD”

$$\begin{bmatrix} A & B & C & D \\ 5 & 2 & 2 & 0 \\ 2 & 2 & 2 & 2 \end{bmatrix}$$

Network data



Nodes

Nodes

$$\begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

Image data

www.sydney.visitorsbureau.com.au

700 x 500

4	45	6
6	12	33
22	17	44

4	45	6	6	12	33	22	17	44

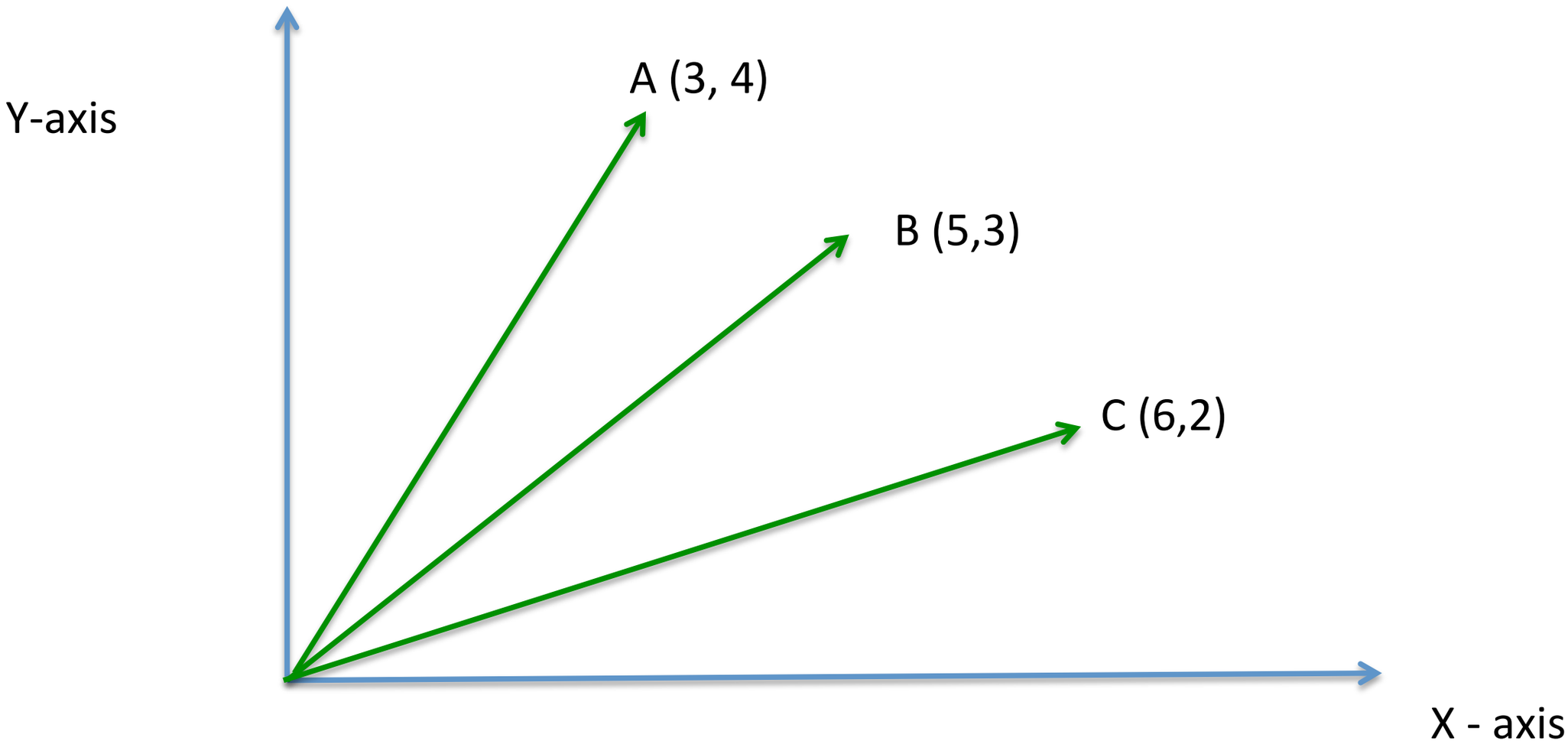
Similarity Computation

THE UNIVERSITY OF SYDNEY

- We can now represent most data types as a matrix.
- A special case of a matrix is a vector.
- Now lets compute similarities with these objects.

Similarity Computation

How can we quantify
similarity between A, B and C ?



Similarity Computation

THE UNIVERSITY OF SYDNEY

- Dot product

$$x = (x_1, x_2, ..., x_n); \quad y = (y_1, y_2, ..., y_n);$$

$$x.y = (x_1 y_1 + x_2 y_2 + ... + x_n y_n);$$

- Norm (length) of a vector

$$\|x\| = (x.x)^{1/2} = (x_1.x_1 + x_2.x_2 + x_n.x_n)^{1/2}$$

Similarity Computation

THE UNIVERSITY OF SYDNEY

- The similarity between two vectors x and y is given by

$$sim(x, y) = x.y/(\|x\| \|y\|)$$

Image search engine

Query Vector (Q)

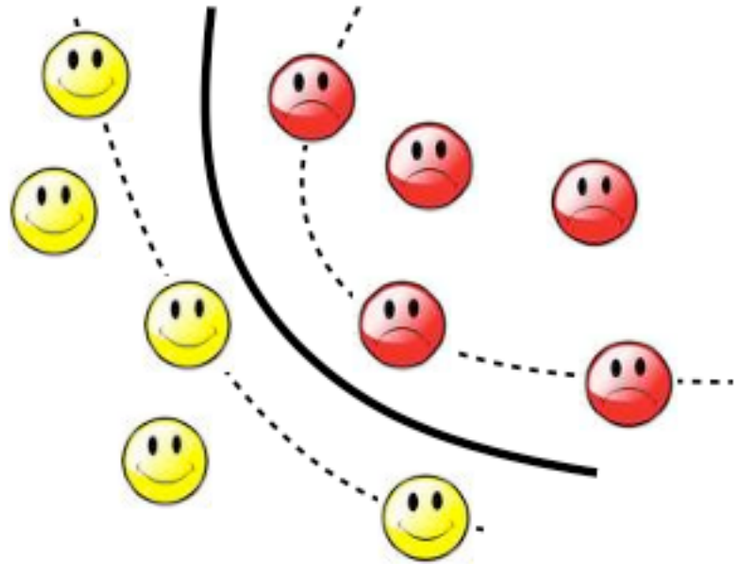
Sim(.,Q)

Most similar image to Q

Image Database

Images As vectors

THE UNIVERSITY OF SYDNEY



Matrix Algebra

Why Matrix Algebra?

THE UNIVERSITY OF SYDNEY

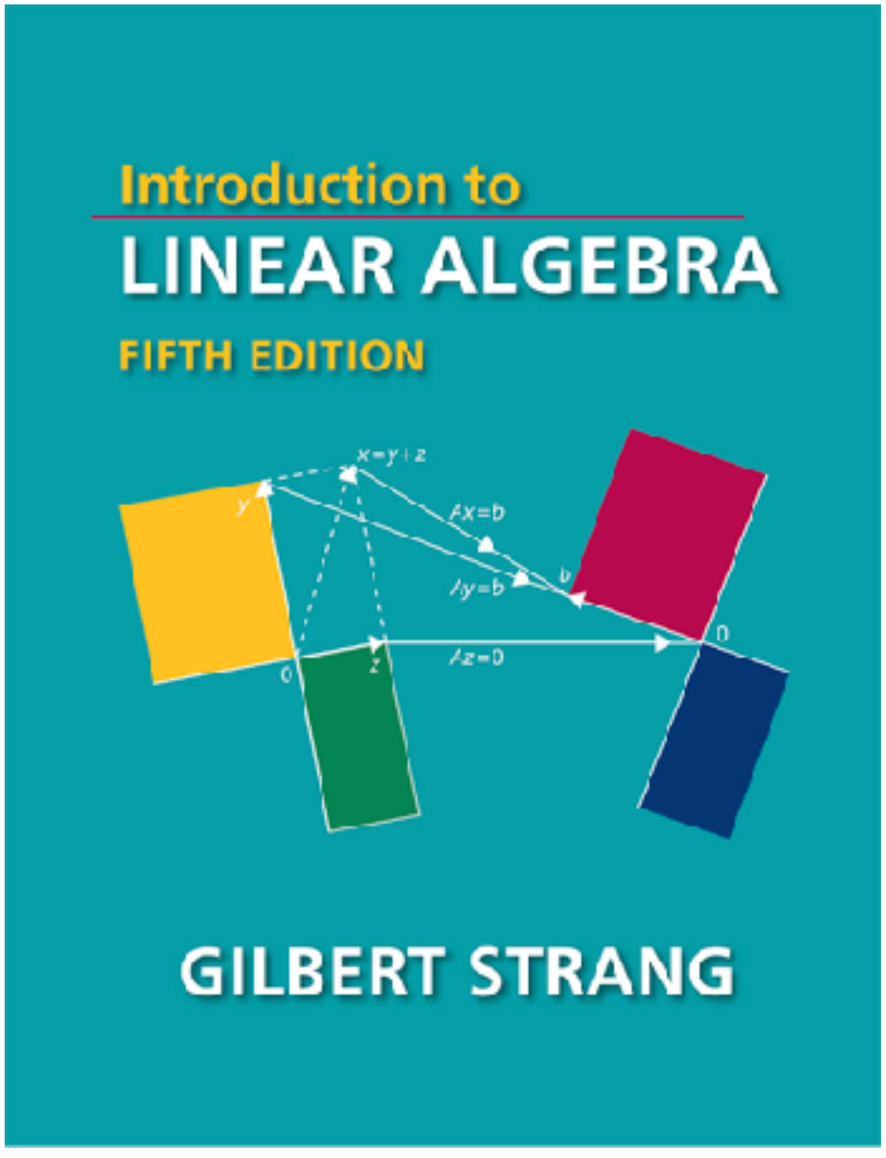
- Data Mining: Computation for large data sets
- Key idea: Data Reduction leads to "Knowledge Discovery"
- Matrix algebra provides simple algorithms with great power for data management, e.g., data reduction or data summarisation.

Why Matrix Algebra?

THE UNIVERSITY OF SYDNEY

- In database management, the fundamental entity is a *relation*.
  - SQL (relational algebra) is a collection of operations on relations – Select, project, join, group-by
- In machine learning and data mining, the fundamental entity is a *matrix*
  - We therefore need to know the basic operations on matrices
  - One important application is to summarise data or extract patterns from data or compress data
  - In SIT several people apply data mining for different domains: Chinese medicine, bioinformatics, student learning, multimedia, image processing, quality of cloud computing service, text analysis, medical imaging, robotics

Suggested book



http://math.mit.edu/~gs/linearalgebra/

Numbers vs Matrices

THE UNIVERSITY OF SYDNEY

Numbers

- Can add and subtract numbers
- Multiply numbers
- Divide two numbers a/b as long as b is not equal to 0.
- Can factorise positive numbers into product of primes

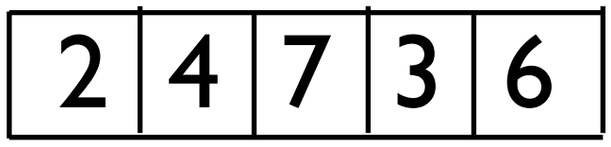
Matrices

- Can add and subtract compatible metrics
- Multiply and divide matrices
- Division of matrices is complicated
- Can factorise any matrix to get data patterns (using a technique called Singular Value Decomposition)

Linear Algebra

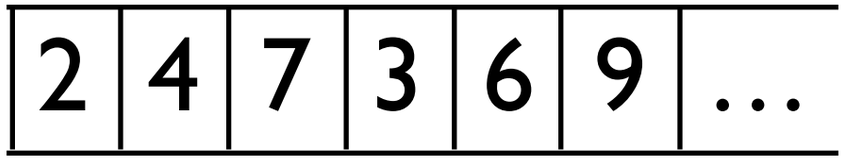
- Area in maths that deals with *vector spaces* and *linear mappings* between these spaces

- The generalisation of LA to infinite dimensions is known as *functional analysis*



D = 5

Linear algebra



D = ∞

Functional analysis

Definition for vector space: 8 Axioms

- Associativity of addition: $(u+v)+w=u+(v+w)$
- Commutativity of addition: $u+v=v+u$
- Identity element of addition: $\exists 0, v+0=v$
- Inverse elements of addition: $\forall v, \exists -v, v+(-v)=0$
- Compatibility of scalar multiplication with field multiplication: (a and b are scalars) $a(bv)=(ab)v$
- Identity element of scalar multiplication: $\exists 1, 1v=v$
- Distributivity of scalar multiplication with respect to vector addition: (a is a scalar)
  $a(u+v)=au+av$
- Distributivity of scalar multiplication with respect to field addition: (a and b are scalers)
  $(a+b)v=av+bv$

Basics I

$$S = \begin{bmatrix} 3 & 0 & 1 \\ 0 & 2 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

- This is a *3 x 3* matrix.

- In general $m \times n$.
  - m rows and n columns
  - Square matrix when $m = n$

- Each row or column could represent one object. If rows are objects then columns are features/attributes/components

Basics II

- Identity matrix I

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

- If A is a square matrix, $AI = IA = A$
- I is an example of a **diagonal** matrix.
- If $A = [a_1, ...a_m]$ is matrix where a_i are the columns, then
  - A is <u>orthogonal</u> if $a_i.a_j = 0$ for $i \neq j$
  - A is <u>orthonormal</u> if above and $a_i.a_i = 1$

Basics III

- Every vector can be written as a linear combination of some finitely many "special" vectors.
- These are called basis-vectors.

$$S = \begin{bmatrix} 3 \\ 2 \\ 2 \end{bmatrix} = 3 \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} + 2 \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} + 2 \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

Linear independence

- Intuitively, a set of vectors is linearly independent if any element of the set cannot be expressed as a linear combination of the others.
- The columns are not linearly independent:

$$S = \begin{bmatrix} 3 & 0 & 1 \\ 0 & 2 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

Exercise

THE UNIVERSITY OF SYDNEY

Given $v = (1, 2, 3)$, calculate:

1. The length of v
2. A unit vector in the same direction as v
3. A set orthogonal bases for v
4. A vector perpendicular to v

Solution

THE UNIVERSITY OF SYDNEY

Given $v = (1, 2, 3)$, calculate:

1. The length of v is 3
2. A unit vector in the same direction $(1/\sqrt{14}, 2/\sqrt{14}, 3/\sqrt{14})$
3. A set orthogonal bases $(0, 0, 1); (0, 1, 0); (0, 0, 1)$
4. A vector perpendicular to $v = (-1, -1, 1)$

Determinant

- The determinant of a square matrix is a single number telling us if the matrix is invertible, $detA^{-1} = 1/(detA)$
- For 2 by 2 matrix

$$|A| = \begin{vmatrix} a & b \\ c & d \end{vmatrix} = ad - bc$$

- For 3 by 3 matrix

$$|A| = \begin{vmatrix} a & b & c \\ d & e & f \\ g & h & i \end{vmatrix} = a\begin{vmatrix} e & f \\ h & i \end{vmatrix} - b\begin{vmatrix} d & f \\ g & i \end{vmatrix} + c\begin{vmatrix} d & e \\ g & h \end{vmatrix}$$

$$= aei + bfg + cdh - ceg - bdi - afh.$$

Rank of a matrix I

- Given a matrix M, the **rank** of a matrix is the maximum number of linearly independent columns.

- A rank 2 matrix:

$$S = \begin{bmatrix} 3 & 0 & 1 \\ 0 & 2 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

Rank of a matrix II

- Can we automatically determine the rank of a matrix

$$S = \begin{bmatrix} 3 & 1 & 3.5 \\ 2 & 2 & 3 \\ 4 & 2 & 5 \end{bmatrix}$$

- Why is rank important anyways…?

Transpose of a matrix

- The transpose of a matrix A , denoted A^T is another matrix B such that $B(i,j) = A(j,i)$
  (just flip the matrix)
- The transpose of S is S^T

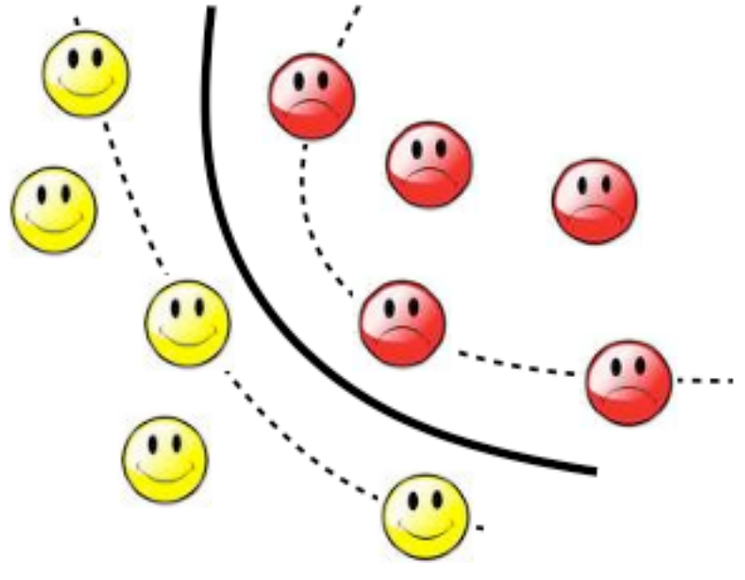
$$S = \begin{bmatrix} 3 & 3 & 1 \\ 0 & 2 & 4 \\ 0 & 0 & 0 \end{bmatrix} \qquad S^T = \begin{bmatrix} 3 & 0 & 0 \\ 3 & 2 & 0 \\ 1 & 4 & 0 \end{bmatrix}$$

Exercise

- What is the rank of the following matrix:

$$\begin{bmatrix} 1 & 2 & 3 \\ 3 & 4 & 5 \\ 5 & 4 & 3 \\ 0 & 2 & 4 \\ 1 & 3 & 5 \end{bmatrix}$$

THE UNIVERSITY OF SYDNEY



Matrix Decompositions

Eigen Decomposition

- For any square matrix A we say that λ is an eigenvalue and $\mathbf{u}$ is its eigenvector if

$$A\mathbf{u} = \lambda\mathbf{u}, \quad \mathbf{u} \neq 0.$$

- Stacking up all eigenvectors/values gives

$$AU = U\Lambda = \begin{bmatrix} | & | & & | \\ \mathbf{u}_1 & \mathbf{u}_2 & \dots & \mathbf{u}_n \\ | & | & & | \end{bmatrix} \begin{bmatrix} \lambda_1 & & & \\ & \lambda_2 & & \\ & & \ddots & \\ & & & \lambda_n \end{bmatrix}$$

Eigen Decomposition

- If A is symmetric, all its e-vals are real, and all its e-vecs are orthonormal, $\mathbf{u}_i^T \mathbf{u}_j = \delta_{ij}$
- Hence $U^T U = U U^T = I, \quad |U| = 1.$
- and $A = U \Lambda U^T = \sum_{i=1}^{n} \lambda_i \mathbf{u}_i \mathbf{u}_i^T$

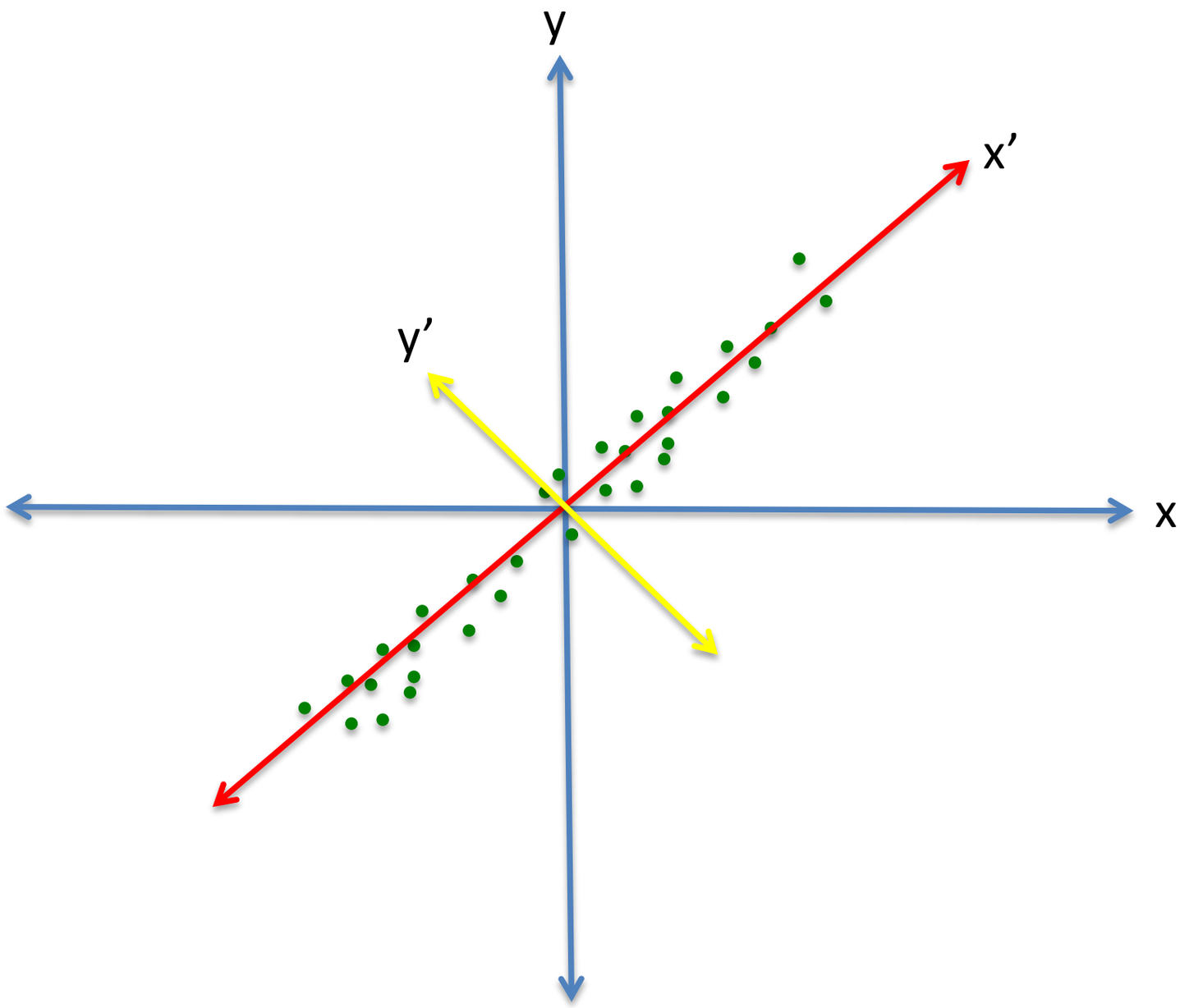
$$A = \begin{bmatrix} | & | & & | \\ \mathbf{u}_1 & \mathbf{u}_2 & \dots & \mathbf{u}_n \\ | & | & & | \end{bmatrix} \begin{bmatrix} \lambda_1 & & & \\ & \lambda_2 & & \\ & & \ddots & \\ & & & \lambda_n \end{bmatrix} \begin{bmatrix} — & \mathbf{u}_1^T & — \\ — & \mathbf{u}_2^T & — \\ & \vdots & \\ — & \mathbf{u}_n^T & — \end{bmatrix}$$

$$= \lambda_1 \begin{bmatrix} | \\ \mathbf{u}_1 \\ | \end{bmatrix} \begin{bmatrix} — & \mathbf{u}_1^T & — \end{bmatrix} + \dots + \lambda_n \begin{bmatrix} | \\ \mathbf{u}_n \\ | \end{bmatrix} \begin{bmatrix} — & \mathbf{u}_n^T & — \end{bmatrix}$$

Principal component

- Given a data matrix $X \in \mathbb{R}^{n \times d}$, the principle components of X are the eigenvectors of $X^\top X$
- Principal component analysis (PCA) for X is to find the eigenvectors and eigenvalues of the matrix $X^\top X$.

Geometric intuition



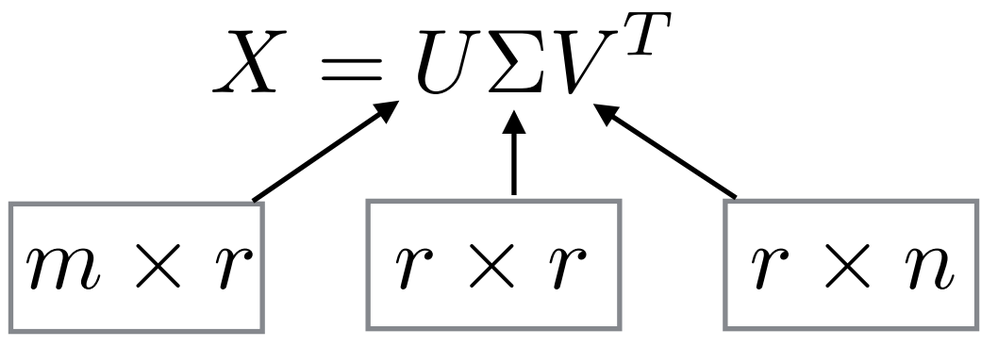
Exercise

- Find the eigenvectors and eigenvalues for the following matrix

$$\begin{bmatrix} 3 & 2 \\ 2 & 6 \end{bmatrix}$$

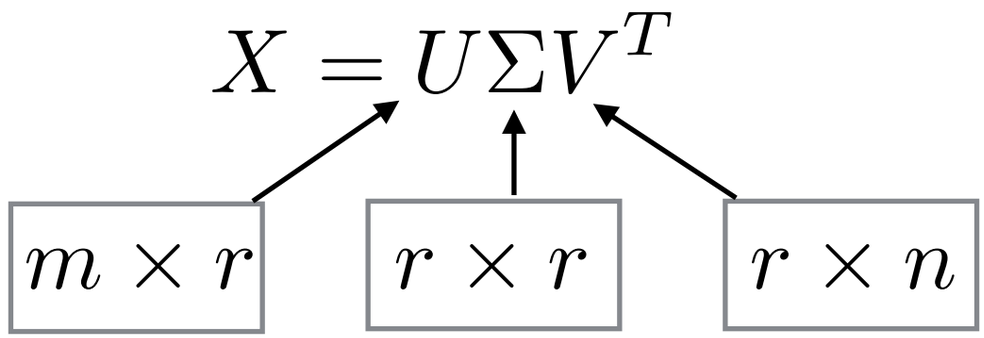
Singular Value Decomposition

- Given **any** real matrix X of size (m, n) it can be expressed as:



- r is the rank of matrix X
- U is a (m, r) column-orthonormal matrix
- V is a (n, r) column-orthonormal matrix
- Σ is diagonal r x r matrix

Singular Value Decomposition



$$X = \lambda_1 \mathbf{u}_1 \times \mathbf{v}_1^t + \lambda_2 \mathbf{u}_2 \times \mathbf{v}_2^t + \cdots + \lambda_r \mathbf{u}_r \times \mathbf{v}_r^t$$

SVD in Python

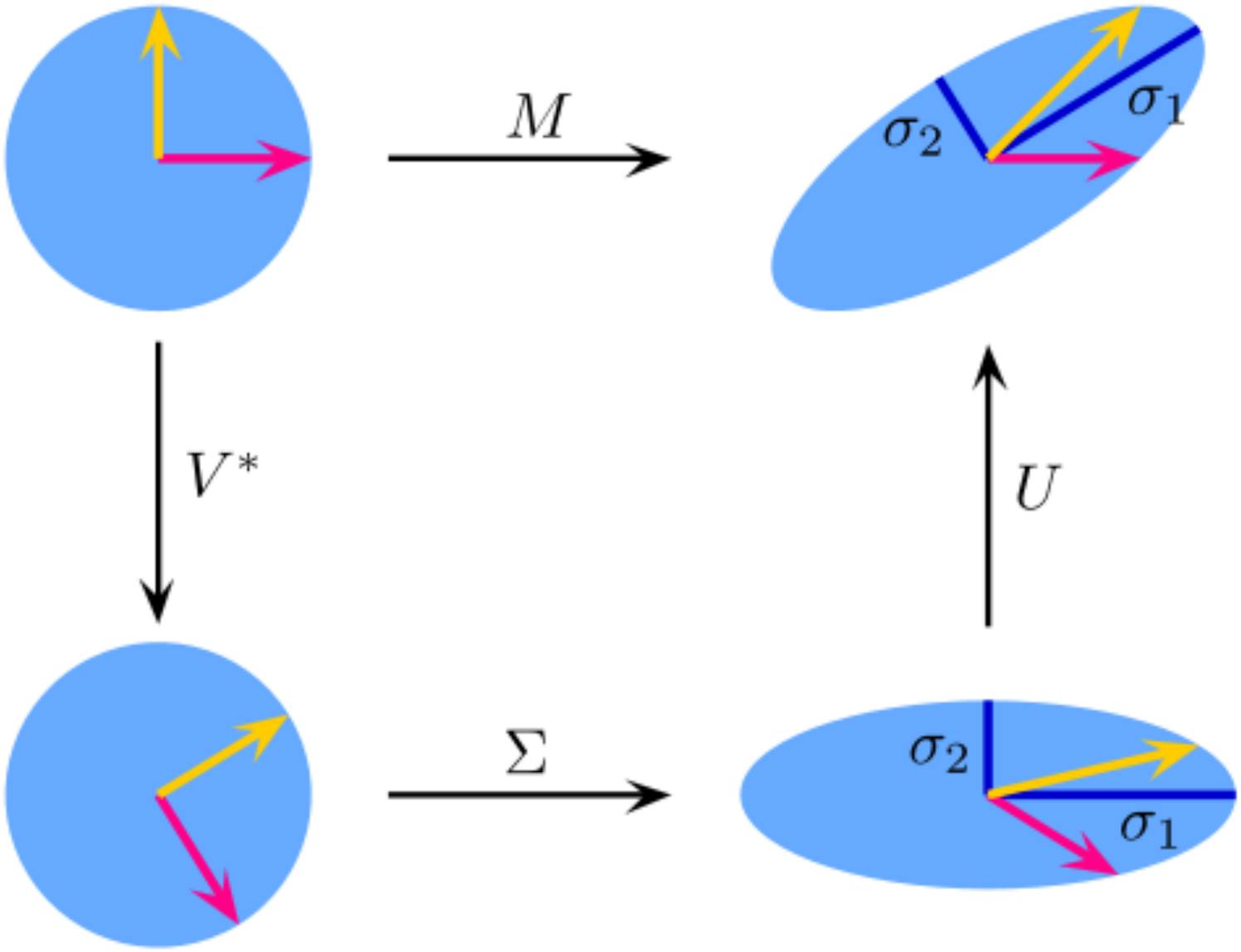
- One line command:

```
>>> U, s,V = np.linalg.svd(a, full_matrices = True)
```

	Wed	Thur	Friday	Sat	Sun
ABC Ltd	1	1	1	0	0
DEF Ltd	2	2	2	0	0
GHI Inc	1	1	1	0	0
KLM Co	5	5	5	0	0
Smith	0	0	0	2	2
Johnson	0	0	0	3	3
Thompson	0	0	0	1	1

$$X = \begin{bmatrix} 0.18 & 0 \\ 0.36 & 0 \\ 0.18 & 0 \\ 0.90 & 0 \\ 0 & 0.53 \\ 0 & 0.80 \\ 0 & 0.27 \end{bmatrix} \times \begin{bmatrix} 9.64 & 0 \\ 0 & 5.29 \end{bmatrix} \times \begin{bmatrix} 0.58 & 0.58 & 0.58 & 0 & 0 \\ 0 & 0 & 0 & 0.71 & 0.71 \end{bmatrix}$$

SVD as a sequence of operations



$$M = U \cdot \Sigma \cdot V^*$$

Qualitative use of SVD

$$X = \begin{bmatrix} 0.18 & 0 \\ 0.36 & 0 \\ 0.18 & 0 \\ 0.90 & 0 \\ 0 & 0.53 \\ 0 & 0.80 \\ 0 & 0.27 \end{bmatrix} \times \begin{bmatrix} 9.64 & 0 \\ 0 & 5.29 \end{bmatrix} \times \begin{bmatrix} 0.58 & 0.58 & 0.58 & 0 & 0 \\ 0 & 0 & 0 & 0.71 & 0.71 \end{bmatrix}$$

- Two kinds of customers (businesses and individuals)
- Two kinds of days (weekday and weekends)
- U is a customer-pattern matrix
- V is a day-pattern matrix
- $V(1,2) = 0$ means Wednesday has zero similarity with the "weekend pattern."

Data reduction with SVD

THE UNIVERSITY OF SYDNEY

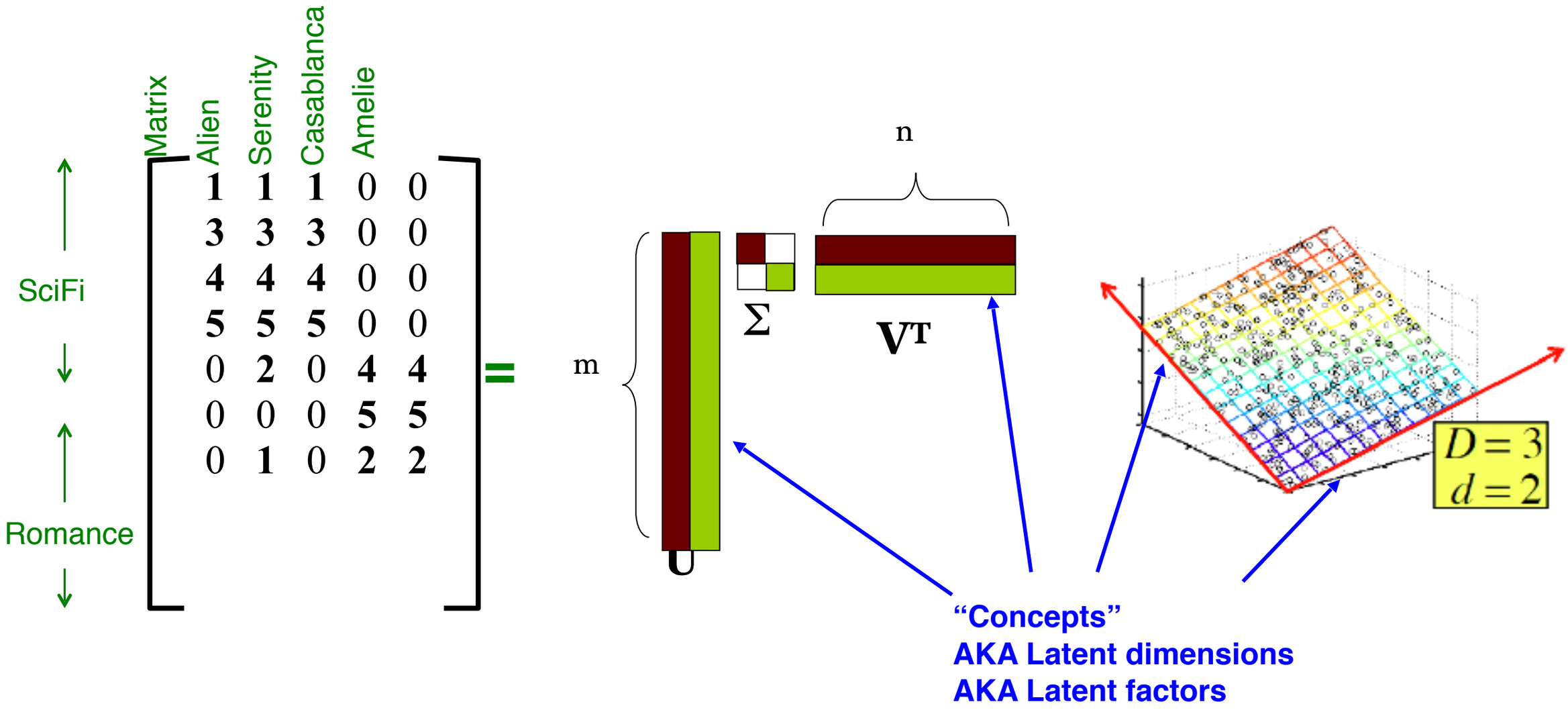
- Now how can we use an SVD decomposition…

- The first way is **qualitative**…
  - Customer-day pattern; or Customer-Pattern-day matrix…

- The second way is **quantitative**…
  - When X is very large, we can compress X into a smaller matrix but still retain important information…

Example: Users to Movies

- $A = U\Sigma V^T$- example: Users to Movies



Example: Users to Movies

- $A = U\Sigma V^T$ - example: Users to Movies

***U* is "user-to-concept" similarity matrix**

SciFi-concept

Romance-concept

"strength" of the SciFi-concept

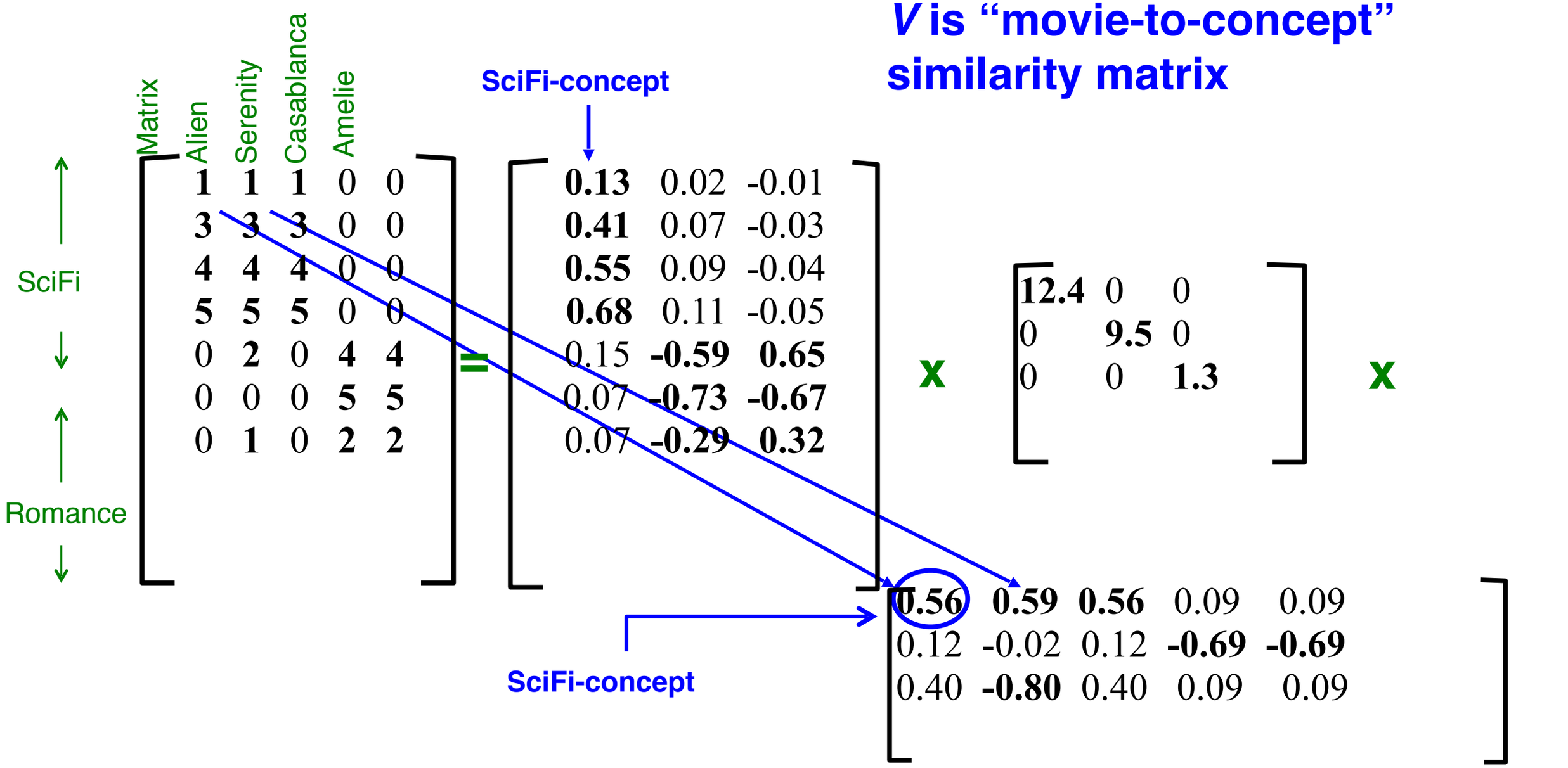
	Matrix	Alien	Dr Strange	American	Capt	Amelie
SciFi		1	1	1	0	0
		3	3	3	0	0
		4	4	4	0	0
		5	5	5	0	0
		0	2	0	4	4
Romance		0	0	0	5	5
		0	1	0	2	2

$$=$$

$$\begin{bmatrix} 0.13 & 0.02 & -0.01 \\ 0.41 & 0.07 & -0.03 \\ 0.55 & 0.09 & -0.04 \\ 0.68 & 0.11 & -0.05 \\ 0.15 & -0.59 & 0.65 \\ 0.07 & -0.73 & -0.67 \\ 0.07 & -0.29 & 0.32 \end{bmatrix} \times \begin{bmatrix} 12.4 & 0 & 0 \\ 0 & 9.5 & 0 \\ 0 & 0 & 1.3 \end{bmatrix} \times \begin{bmatrix} 0.56 & 0.59 & 0.56 & 0.09 & 0.09 \\ 0.12 & -0.02 & 0.12 & -0.69 & -0.69 \\ 0.40 & -0.80 & 0.40 & 0.09 & 0.09 \end{bmatrix}$$

Example: Users to Movies

- $A = U\Sigma V^T$ - example:



Interpretation

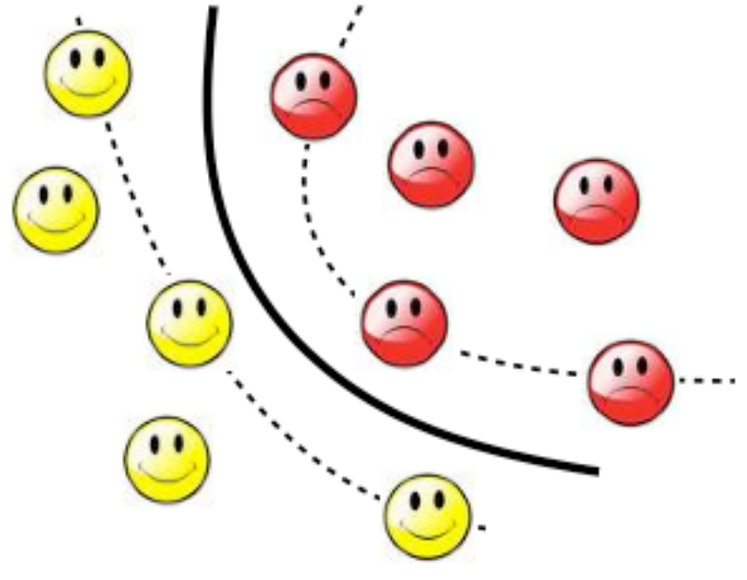
‘**movies**’, ‘**users**’ and ‘**concepts**’:

- ***U***: user-to-concept similarity matrix

- ***V***: movie-to-concept similarity matrix

- Σ: its diagonal elements:
  ‘strength’ of each concept

THE UNIVERSITY OF SYDNEY



Data Reduction and Matrix compression

Advice Querying on X

- Suppose X was very very large… (all mobile users in China)
- Now the query is "Find the amount spent by Jack Ma on Friday"
  - The original answer $X(JackMa, Friday)$
  - The approximate answer $\hat{X}(JackMa, Friday)$
  - Is it a good approximation?

Spectral representation

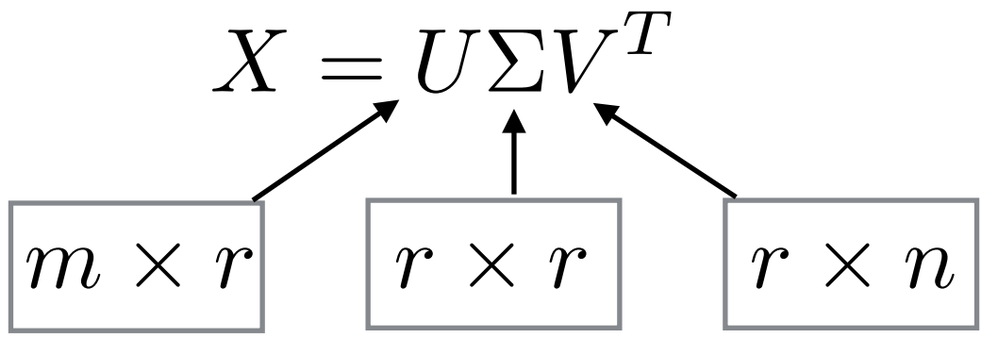
- Matrix X can also be written as:

$$X = \lambda_1 \mathbf{u}_1 \times \mathbf{v}_1^t + \lambda_2 \mathbf{u}_2 \times \mathbf{v}_2^t + \cdots + \lambda_r \mathbf{u}_r \times \mathbf{v}_r^t$$

- The above is called **spectral** representation.

$$X = 9.64 \times \begin{bmatrix} 0.18 \\ 0.36 \\ 0.18 \\ 0.90 \\ 0 \\ 0 \\ 0 \end{bmatrix} \times \begin{bmatrix} 0.58 & 0.58 & 0.58 & 0 & 0 \end{bmatrix} + 5.29 \times \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0.53 \\ 0.80 \\ 0.27 \end{bmatrix} \times \begin{bmatrix} 0 & 0 & 0.71 & 0.71 \end{bmatrix}$$

Example: compression of X



- Size of $X = mn$
- Size of $U + \Sigma + V$ is $mr + r + nr$
- Thus compression ratio is

$$\frac{mr + r + nr}{mn} = \frac{r(m + 1 + n)}{mn} \approx \frac{rm}{mn} = \frac{r}{n}$$

- Can we do better?

Example: compression of X

- To get better compression we should look at the λ values.
  - These are called **singular values**.
- We can arrange them in descending order:

$$\lambda_1 \geqslant \lambda_2 \geqslant \cdots \geqslant \lambda_r > 0$$

- Now recall….

$$X = \lambda_1 \mathbf{u}_1 \times \mathbf{v}_1^t + \lambda_2 \mathbf{u}_2 \times \mathbf{v}_2^t + \cdots + \lambda_r \mathbf{u}_r \times \mathbf{v}_r^t$$

Example: compression of X

- Now a compact way of writing the spectral representation is:

$$X = \sum_{i=1}^{r} \lambda_i \mathbf{u}_i \times \mathbf{v}_i^t$$

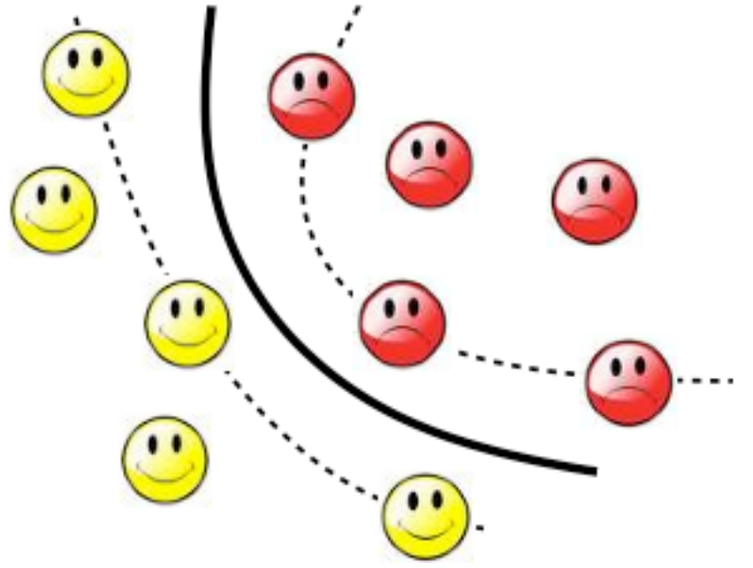
- However, can approximate it as:

$$\hat{X} = \sum_{i=1}^{k} \lambda_i \mathbf{u}_i \times \mathbf{v}_i^t$$

- This new compression ratio is:

$$\frac{mk + k + nk}{mn} = \frac{k(m + 1 + n)}{mn} \approx \frac{km}{mn} = \frac{k}{n} \leqslant \frac{r}{n}$$

THE UNIVERSITY OF SYDNEY



Thanks!