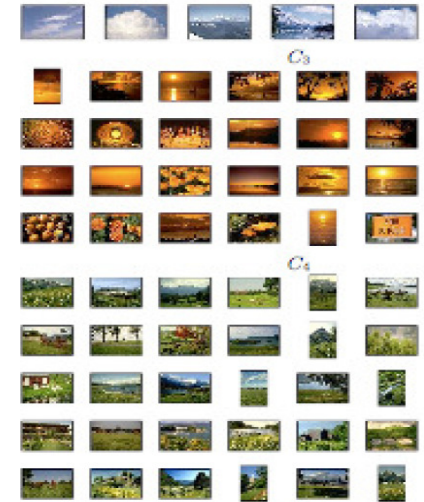
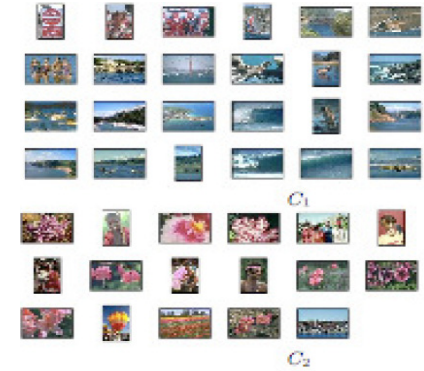
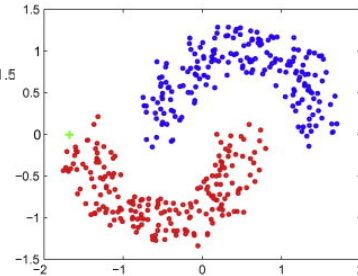
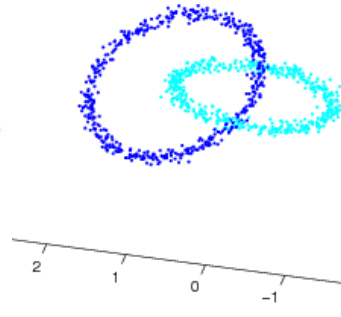
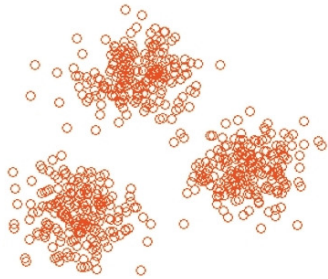


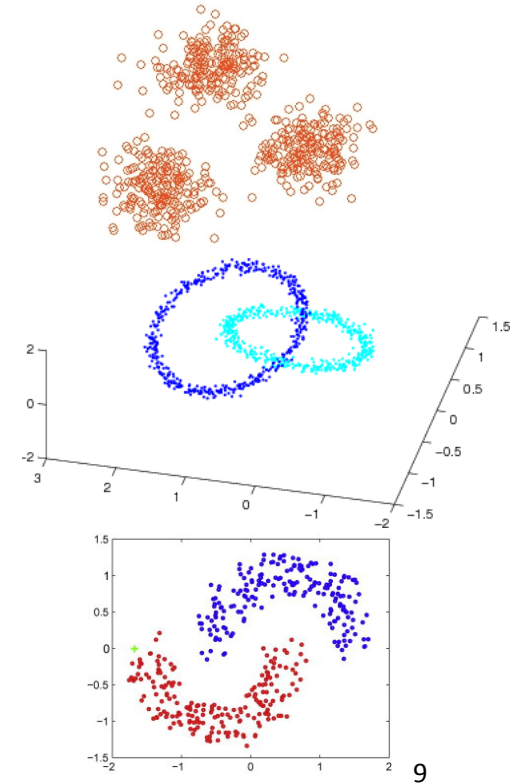
What are the relations  
between data points?



# Handwritten digits

```
72104149590690159784766540740131
347271211724235124463556041957893
74643070291732977627847361368314
17696054992194873974442567905
85665781016467317182029955156034
4654654514472327181818503925011
09031642361113952945939036557227
128417338879224159872304424195771
28268577918180301994182129759264
15429204002847124027433003196525
17930420711215339786361381051315
561851794622506563720885411403737
61621928617525440838246031773797
19217292049148184599837600302664
953323912680566637882758961841259
19754089910523789406395213136577
22632654997130383193446421825488
40023277087447969098046063548339
33778087170654380963809968685786
02402231975108462479309822927359
18020311376712580371409186774349
19317397691328336129585114431077
0794485540827084504061332626931
46251206217341054311749940402451
16471942415538314568941538032512
8344088331735963261360721714221
79611248177480231310770355216692
83522560827288887493066321322930
05781446029147473988471212232303
91740355863267663279117564951334
78911691445406223151203812671673
90122089
```

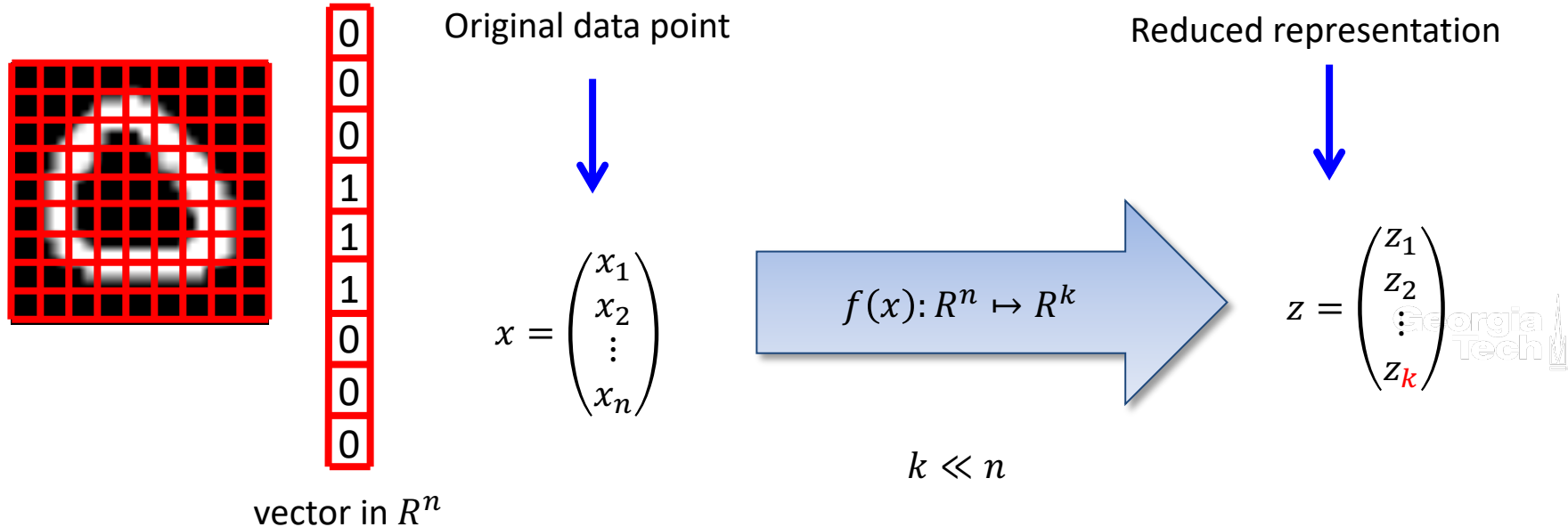
What are the relations  
between data points?



# What is dimensionality reduction?

The process of reducing the number of random variables under consideration

- One can combine, transform or select variables
- One can use linear or nonlinear operations



# Why dimensionality reduction and how to think

- The dimension-reduced data can be used for
  - Visualizing, exploring and understanding the data
  - Extracting "features" and dominant modes
  - Cleaning data
  - Speeding up subsequent learning task
  - Building simpler model later
- Applications
  - Image compression
  - Face recognition (eigenface)
  - Natural language processing (latent semantic analysis)



# Principal component analysis

Given  $m$  data points,  $\{x^1, x^2, \dots, x^m\} \in R^d$

Step 1: Estimate the mean and covariance matrix from data

$$\mu = \frac{1}{m} \sum_{i=1}^m x^i \quad \text{and} \quad C = \frac{1}{m} \sum_{i=1}^m (x^i - \mu)(x^i - \mu)^\top$$

Weight vectors



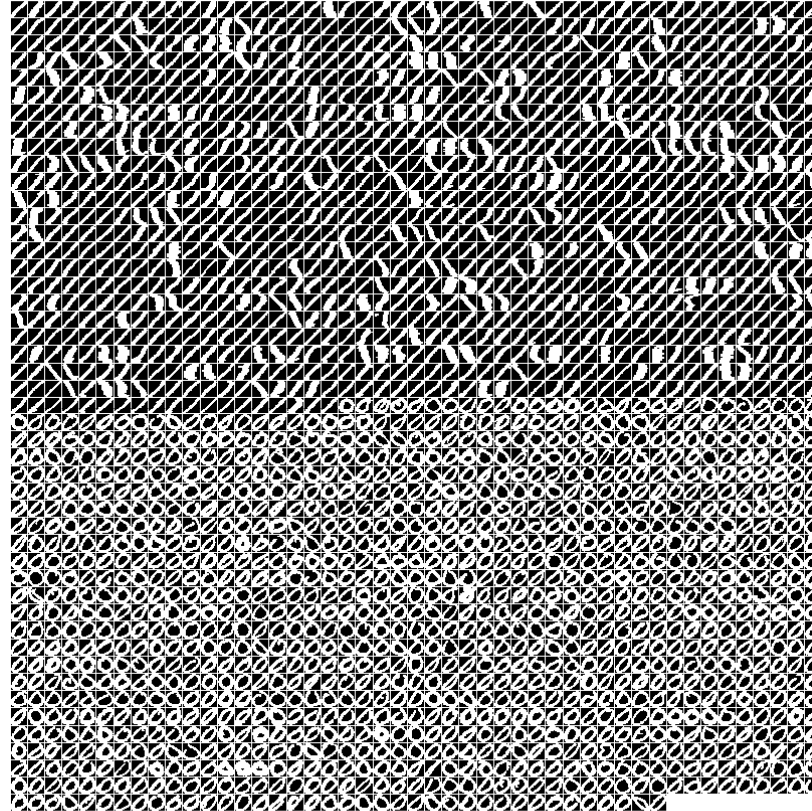
Step 2: Take the eigenvectors  $w^1, w^2, \dots$  of  $C$  corresponding to the largest eigenvalue  $\lambda_1$ , the second largest eigenvalue  $\lambda_2$  ...

Step 3: Compute reduced representation (**principle components** of a data point)

$$z^i = \begin{pmatrix} w^{1\top} (x^i - \mu) / \sqrt{\lambda_1} \\ w^{2\top} (x^i - \mu) / \sqrt{\lambda_2} \\ \vdots \end{pmatrix}$$

# Run demo PCA\_digits.m

digit 1 and 0



# Run demo PCA\_leaf.m



Shape feature	Description
<i>Eccentricity</i>	Eccentricity of the ellipse with identical second moments to $I$ . This value ranges from 0 to 1.
<i>Aspect Ratio</i>	Consider any $X, Y \in \partial I$ . Choose $X$ and $Y$ such that $d(X, Y) = D(I)$ . Find $Z, W \in \partial I$ maximizing $D^\perp = d(Z, W)$ on the set of all pairs of $\partial I$ that define a segment orthogonal to $[XY]$ . The aspect ratio is defined as the quotient $D(I)/D^\perp$ . Values close to 0 indicate an elongated shape.
<i>Elongation</i>	Compute the maximum escape distance $d_{\max} = \max_{X \in I} d(X, \partial I)$ . Elongation is obtained as $1 - 2d_{\max}/D(I)$ and ranges from 0 to 1. The minimum is achieved for a circular region. Note that the ratio $2d_{\max}/D(I)$ is the quotient between the diameter of the largest inscribed circle and the diameter of the smallest circumscribed circle.
<i>Solidity</i>	The ratio $A(I)/A(H(I))$ is computed, which can be understood as a certain measure of convexity. It measures how well $I$ fits a convex shape.
<i>Stochastic Convexity</i>	This variable extends the usual notion of convexity in topological sense, using sampling to perform the calculation. The aim is to estimate the probability of a random segment $[XY]$ , $X, Y \in I$ , to be fully contained in $I$ .
<i>Isoperimetric Factor</i>	The ratio $4\pi A(I)/L(\partial I)^2$ is calculated. The maximum value of 1 is reached for a circular region. Curvy intertwined contours yield low values.
<i>Maximal Indentation Depth</i>	Let $C_{H(I)}$ and $L(H(I))$ denote the centroid and arclength of $H(I)$ . The distances $d(X, C_{H(I)})$ and $d(Y, C_{H(I)})$ are computed $\forall X \in H(I)$ and $\forall Y \in \partial I$ . The indentation function can then be defined as $[d(X, C_{H(I)}) - d(Y, C_{H(I)})]/L(H(I))$ , which is sampled at one degree intervals. The maximal indentation depth $\mathfrak{D}$ is the maximum of this function.
<i>Lobedness</i>	The Fourier Transform of the indentation function above is computed after mean removal. The resulting spectrum is normalized by the total energy. Calculate lobedness as $F \times \mathfrak{D}^2$ , where $F$ stands for the smallest frequency at which the cumulated energy exceeds 80%. This feature characterizes how lobed a leaf is.

Texture feature	Description
<i>Average Intensity</i>	Average intensity is defined as the mean of the intensity image, $m$ .
<i>Average Contrast</i>	Average contrast is the the standard deviation of the intensity image, $\sigma = \sqrt{\mu_2(z)}$ .
<i>Smoothness</i>	Smoothness is defined as $R = 1 - 1/(1 + \sigma^2)$ and measures the relative smoothness of the intensities in a given region. For a region of constant intensity, $R$ takes the value 0 and $R$ approaches 1 as regions exhibit larger disparities in intensity values. $\sigma^2$ is generally normalized by $(L - 1)^2$ to ensure that $R \in [0, 1]$ .
<i>Third moment</i>	$\mu_3$ is a measure of the intensity histogram's skewness. This measure is generally normalized by $(L - 1)^2$ like smoothness.
<i>Uniformity</i>	Defined as $U = \sum_{i=0}^{L-1} p^2(z_i)$ , uniformity's maximum value is reached when all intensity levels are equal.
<i>Entropy</i>	A measure of intensity randomness.

## 8 shape features and 6 texture features

# Use what criterion for reduction?

There are many criteria (geometric based, information theory based, etc.)

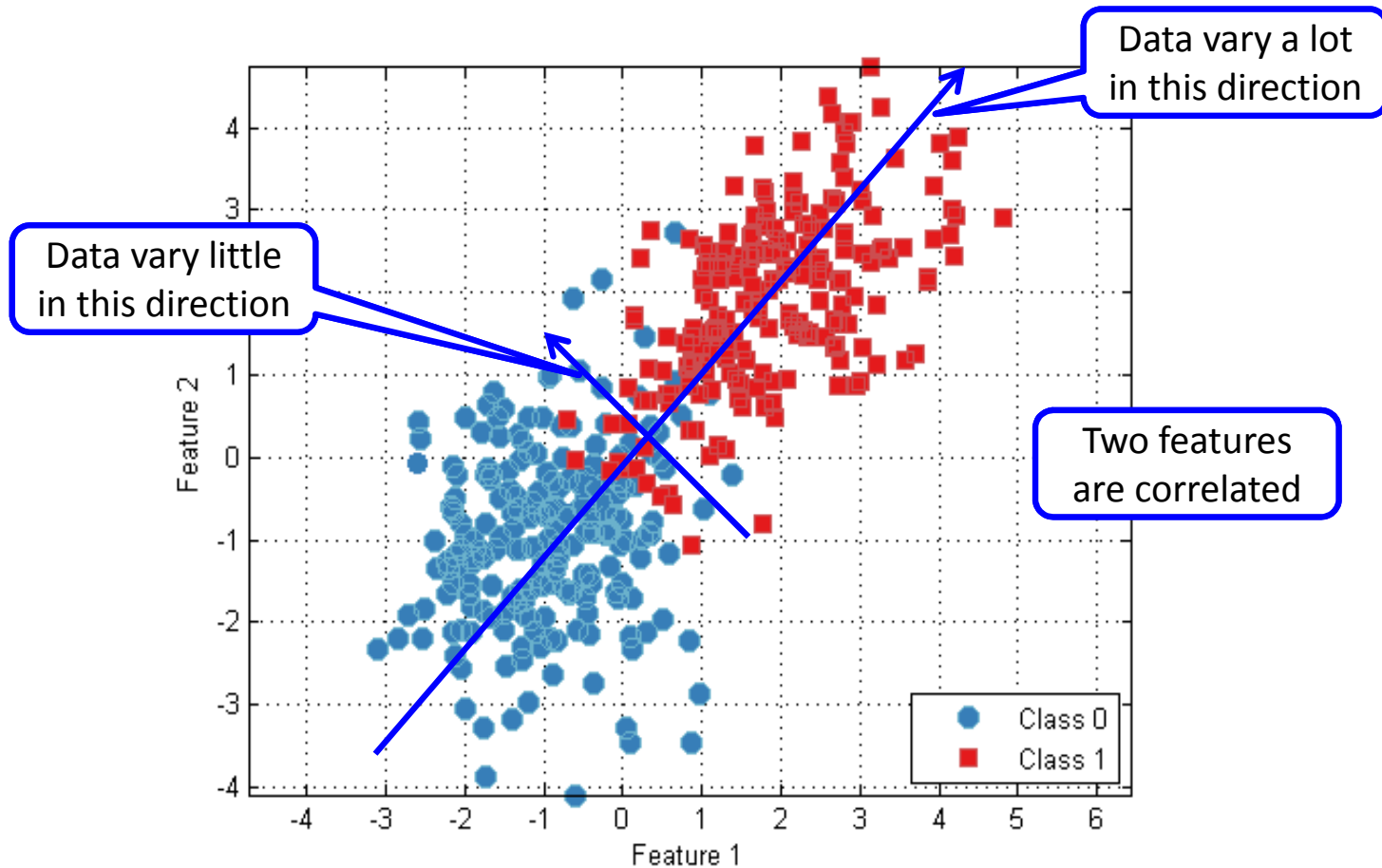
One criterion: want to capture **variation** in data

- variations are “signals” or information in the data
- need to normalize each variables first

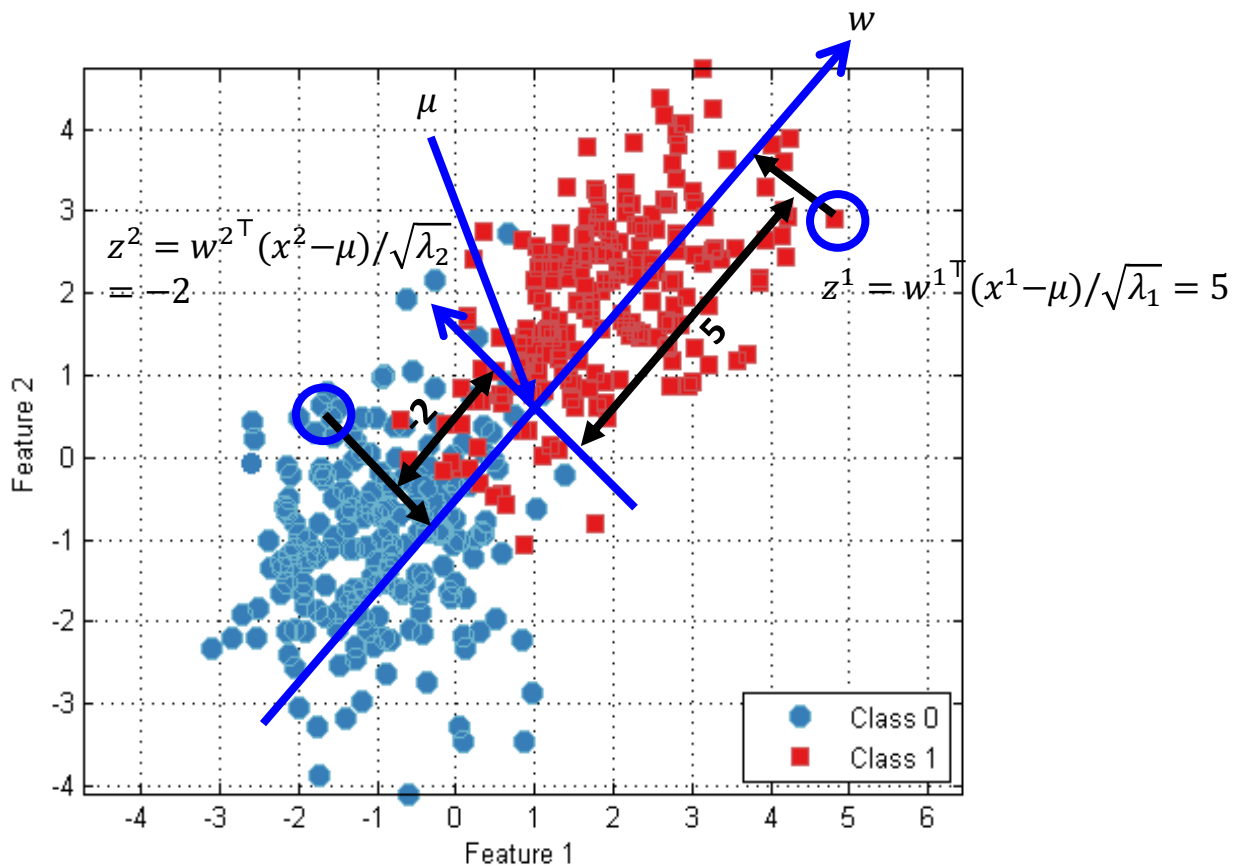
In the process, also discover variables or dimensions highly **correlated**

- represent highly related phenomena
- combine them to form a stronger signal
- lead to simpler presentation

# An example



# An example (cont.)



# How to formulate the problem

Given  $m$  data points,  $\{x^1, x^2, \dots, x^m\} \in R^n$ , with their mean  $\mu = \frac{1}{m} \sum_{i=1}^m x^i$

Find a direction  $w \in R^n$  where  $\|w\| \leq 1$

Such that the variance (or variation) of the data along direction  $w$  is maximized

$$\max_{w: \|w\| \leq 1} \underbrace{\frac{1}{m} \sum_{i=1}^m (w^\top x^i - w^\top \mu)^2}_{\text{variance}}$$



# Is it an easy optimization problem?

Manipulate the objective with linear algebra

$$\begin{aligned} & \frac{1}{m} \sum_{i=1}^m (w^\top x^i - w^\top \mu)^2 \\ &= \frac{1}{m} \sum_{i=1}^m (w^\top (x^i - \mu))^2 \\ &= \frac{1}{m} \sum_{i=1}^m w^\top (x^i - \mu) (x^i - \mu)^\top w \\ &= w^\top \underbrace{\left( \frac{1}{m} \sum_{i=1}^m (x^i - \mu) (x^i - \mu)^\top \right)}_{\text{covariance matrix } \mathcal{C}} w \end{aligned}$$

# Landscape of the optimization problem

Suppose the data has two dimension ( $n = 2$ )

$C$  is a diagonal matrix

$$C = \begin{pmatrix} 1 & 0 \\ 0 & 2 \end{pmatrix}$$

The optimization problem becomes

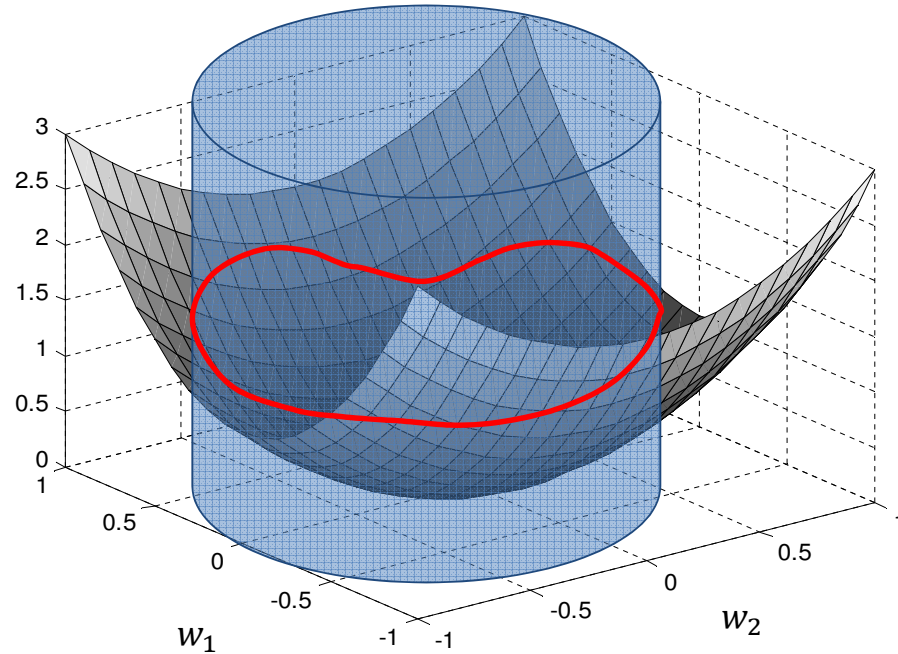
$$\max_{w: \|w\| \leq 1} w^T C w$$

$$= \max_{w: \|w\| \leq 1} (w_1, w_2) \begin{pmatrix} 1 & 0 \\ 0 & 2 \end{pmatrix} \begin{pmatrix} w_1 \\ w_2 \end{pmatrix}$$

$$= \max_{w: \|w\| \leq 1} w_1^2 + 2w_2^2$$

# Landscape of the optimization problem

- $f(w_1, w_2) = w_1^2 + 2w_2^2$



# Solving the PCA problem

$$\max_{w: \|w\| \leq 1} w^\top C w, \quad C = \frac{1}{m} \sum_{i=1}^m (x^i - \mu)(x^i - \mu)^\top$$

- Form Lagrangian function of the optimization problem

$$L(w, \lambda) = w^\top C w + \lambda(1 - \|w\|^2)$$

- If  $w$  is a maximum of the original optimization problem, then there exists a  $\lambda$ , where  $(w, \lambda)$  is a **stationary point** of  $L(w, \lambda)$
- This implies that

$$\frac{\partial L}{\partial w} = 0 = 2Cw - 2\lambda w \Leftrightarrow Cw = \lambda w$$

- The optimal solution  **$w$**  should be an eigen-vector of  $C$
- Objective function becomes  **$\lambda$**  (associated with  **$w$** )

# Variance of in the principal direction

- The optimal solution  $w$  should be an eigen-vector of  $C$

$$Cw = \lambda w$$

- Objective function becomes  $\lambda$  (associated with  $w$ )

$$w^T C w = \lambda w^T w = \lambda ||w||^2$$



eigen-value

- The problem becomes finding the largest eigenvalue of  $C$

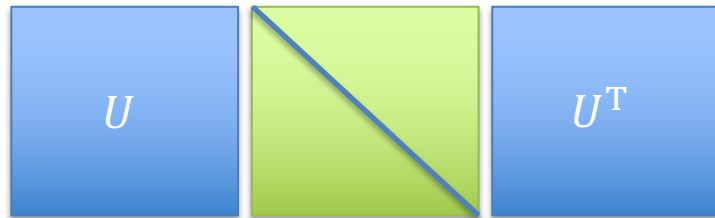
# Eigenvalue problem

- Given a symmetric matrix  $C \in R^{n \times n}$ 
  - Find a vector  $u \in R^n$  and  $\|u\| = 1$
  - Such that

$$Cu = \lambda u$$

- There will be multiple solution:  $u^1, u^2, \dots u^n$  (called the **eigenvectors**) with different  $\lambda_1, \lambda_2, \dots \lambda_n$  (called the **eigenvalues**.)
  - Eigenvectors are ortho-normal:  $u^{i\top} u^i = 1, u^{i\top} u^j = 0$
  - Eigenvalues are called spectrum
- Eigendecomposition

$$C = U \Lambda U^T$$



# Find multiple principal directions

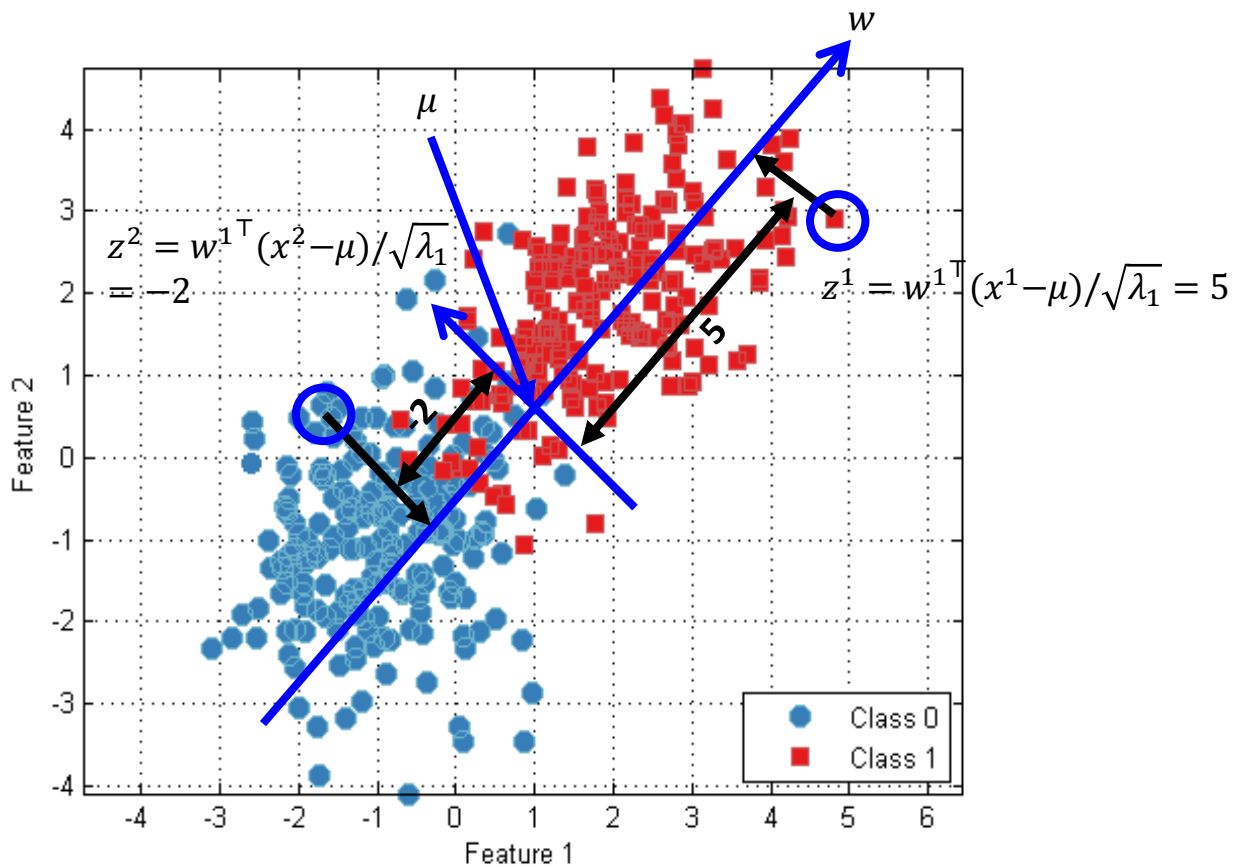
Directions  $w^1, w^2, \dots$  which has

- the largest variances
- **orthogonal** to each other

Take the eigenvectors  $w^1, w^2, \dots$  of  $C$  corresponding to

- the largest eigenvalue  $\lambda_1$ ,
  - the second largest eigenvalue  $\lambda_2$
- and so on.

# An example (cont.)





# Principal component analysis

Given  $m$  data points,  $\{x^1, x^2, \dots, x^m\} \in R^d$

Step 1: Estimate the mean and covariance matrix from data

$$\mu = \frac{1}{m} \sum_{i=1}^m x^i \quad \text{and} \quad C = \frac{1}{m} \sum_{i=1}^m (x^i - \mu)(x^i - \mu)^\top$$

Weight vectors



Step 2: Take the eigenvectors  $w^1, w^2, \dots$  of  $C$  corresponding to the largest eigenvalue  $\lambda_1$ , the second largest eigenvalue  $\lambda_2 \dots$

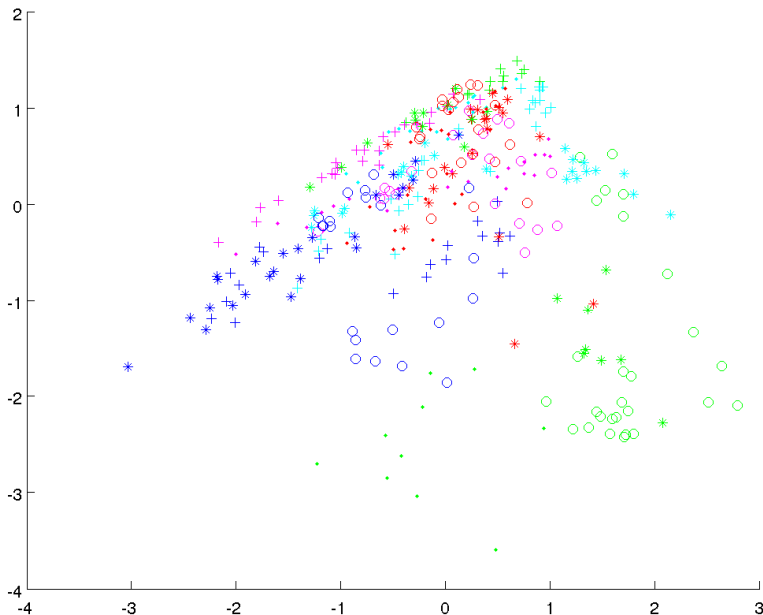
Step 3: Compute reduced representation (**principle components** of a data point)

$$z^i = \begin{pmatrix} w^{1\top} (x^i - \mu) / \sqrt{\lambda_1} \\ w^{2\top} (x^i - \mu) / \sqrt{\lambda_2} \\ \vdots \end{pmatrix}$$

# Look more into PCA\_leaf.m



# Interpreting the reduced representation



Principal direction:  
 $W =$

0.0938 0.1924

0.1902 0.0253

0.2266 -0.1800

-0.1850 0.4084

-0.1600 0.3825

-0.2063 0.3488

0.1940 -0.4037

0.2150 -0.3566

Shape  
features

-0.3723 -0.2001

-0.3657 -0.1974

-0.3602 -0.2037

-0.3175 -0.1886

-0.3056 -0.1243

-0.3482 -0.1829

Texture  
features

# Singular Value Decomposition (SVD)

- Singular value decomposition, known as SVD, is a factorization of a real matrix
- For a matrix  $M \in \mathbb{R}^{n \times m}$  ( $n \leq m$ )

$$M = U \Sigma V^T$$

$$M = [u_1 \ u_2 \ \dots \ u_n] \begin{bmatrix} \sigma_1 & 0 & 0 & \\ 0 & \ddots & 0 & 0 \\ 0 & 0 & \sigma_n & \end{bmatrix} [v_1 \ v_2 \ \dots \ v_m]^T$$

$U \in \mathbb{R}^{n \times n}$   
Left singular vectors  
(orthonormal)

$\Sigma \in \mathbb{R}^{n \times m}$   
Singular values

$V \in \mathbb{R}^{m \times m}$   
Right singular vectors  
(orthonormal)

where  $U \in \mathbb{R}^{n \times n}$ ,  $V^T \in \mathbb{R}^{m \times m}$ ,  $\Sigma \in \mathbb{R}^{n \times m}$

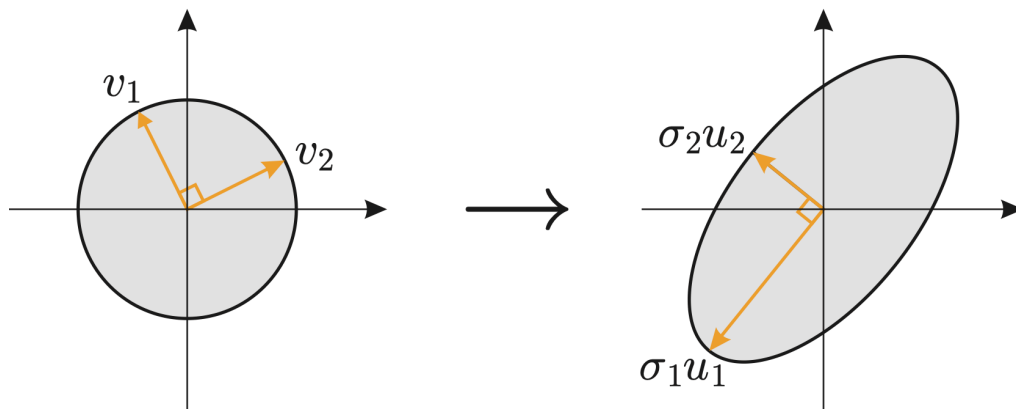
Typically  $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_n$

# Interpretations of SVD

- A pair of singular vectors  $(u, v)$  satisfies

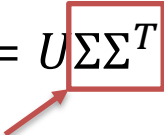
$$Mv = \sigma u \quad \text{and} \quad M^T u = \sigma v$$

- Geometry



# Relationship between SVD and eigendecomposition

$$M = U\Sigma V^{\top}$$

$$C := MM^T = U\Sigma V^T V \Sigma^T U^T = U \Sigma \Sigma^T U^T$$


$$\begin{bmatrix} \sigma_1^2 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \sigma_n^2 \end{bmatrix}$$

- The eigenvectors of  $C := MM^T$  is  $U$  (the left singular vectors of  $M$ )
- The eigenvalues of  $C$  is  $\sigma_i^2$  (squared singular values of  $M$ )
- Similar results can be derived for  $M^T M$

# Another way to perform PCA (using SVD)

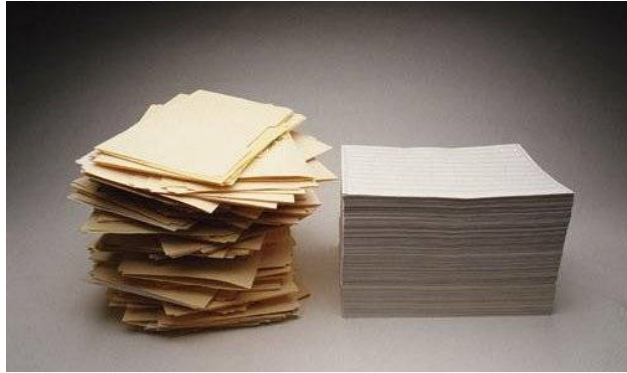
- Note that data covariance matrix can be written as

$$C = \frac{1}{m} \sum_{i=1}^m (x^i - \mu)(x^i - \mu)^\top = \frac{1}{m} X X^\top$$

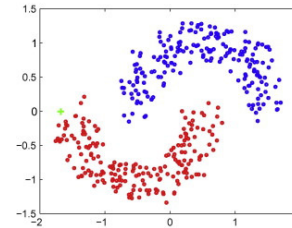
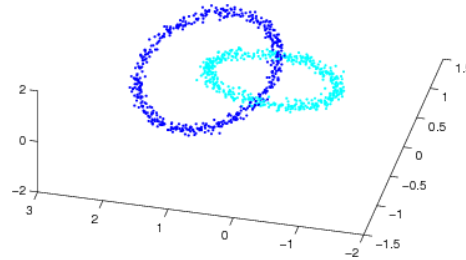
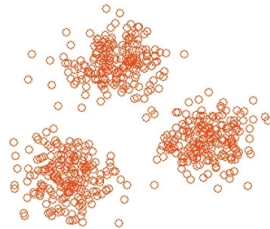
$$X = [x^1 - \mu, \dots, x^m - \mu] \in R^{n \times m}$$

- Eigenvectors of  $C$  corresponds to left singular vectors of  $X$
- Find the weight vectors  $\{w^1, w^2, \dots, w^r\}$  as the  $r$  left singular vectors of the data matrix  $X$  ( $r$  is the number of principle components)

# Documents collections



What are the relations  
between data points?



<http://submitpapers.com/misc/docCollection.html>



# Bag of words representation

document 1

Machine learning concerns the construction and study of systems that can learn from data.

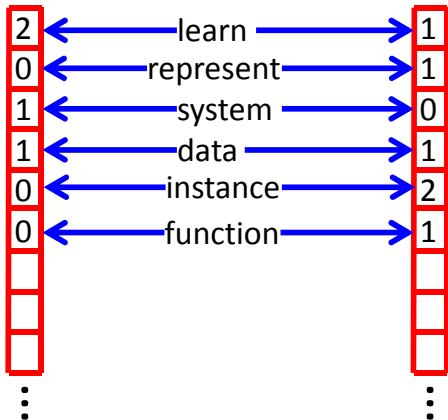
document 2

Representation of data instances and functions evaluated on these instances are part of all machine learning systems

...

Documents

Feature vector



Report Type: Original  
 Contention:  
 Atlanta Police Department  
 INCIDENT REPORT  
 Incident #: CDA # 00-000-0007  
 Report Status: Accused

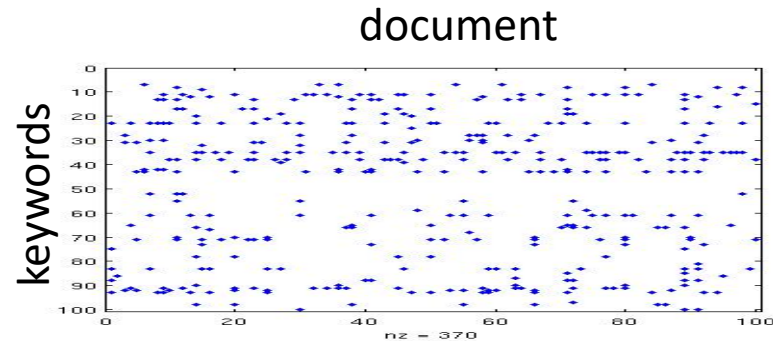
1	Incident #	00000000000000000000	Date	10/10/2000	Time	10:00:00	Reported By	10000000000000000000
2	Incident Type	00000000000000000000	Location	00000000000000000000	Address	00000000000000000000	City	00000000000000000000
3	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
4	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
5	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
6	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
7	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
8	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
9	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
10	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
11	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
12	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
13	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
14	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
15	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
16	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
17	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
18	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
19	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
20	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
21	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
22	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
23	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
24	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
25	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
26	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
27	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
28	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
29	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
30	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
31	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
32	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
33	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
34	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
35	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
36	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
37	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
38	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
39	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
40	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
41	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
42	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
43	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
44	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
45	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
46	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
47	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
48	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
49	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
50	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
51	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
52	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
53	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
54	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
55	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
56	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
57	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
58	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
59	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
60	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
61	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
62	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
63	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
64	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
65	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
66	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
67	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
68	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
69	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
70	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
71	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
72	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
73	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
74	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
75	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
76	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
77	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
78	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
79	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
80	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
81	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
82	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
83	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
84	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
85	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
86	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
87	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
88	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
89	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
90	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
91	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
92	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
93	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
94	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
95	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
96	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
97	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
98	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
99	Offense Code	00000000000000000000	Offense Description	00000000000000000000				
100	Offense Code	00000000000000000000	Offense Description	00000000000000000000				



# Latent semantic analysis (LSA)

- Bag-of-words model or term-document matrix  $M$  (more natural language processing techniques: WF/IDF, removing stop words, N-gram)
- Perform PCA of  $M$ : (Latent Semantic Analysis, LSA)
- principle components  $z^i$  for each document can be interpreted as “feature” of a document

$X =$



# Example: Atlanta Police reports

- 20000 police reports, extract 7000 keywords

*A free text of  
a police report*

*A bag of words for  
a police report*

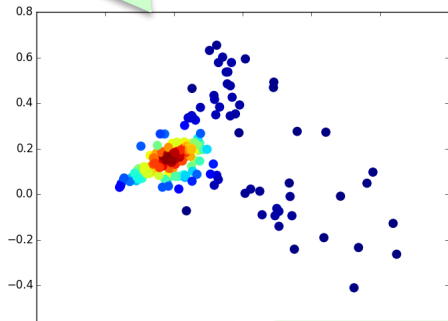
*I Investigator Pickering was  
advised of a traffic accident  
near the intersection of Ted  
Turner Drive and W. Peachtree  
Pl in which the suspects ran  
from the vehicle. Two  
individuals were observed by  
Atlanta Police Officers running  
from vehicle. One suspect was  
arrested a short distance from  
the crash scene. ... ..*

Terms	Counts
investigator	1
gun	0
kill	0
traffic	1
suspect	2
arrested	1
...	...

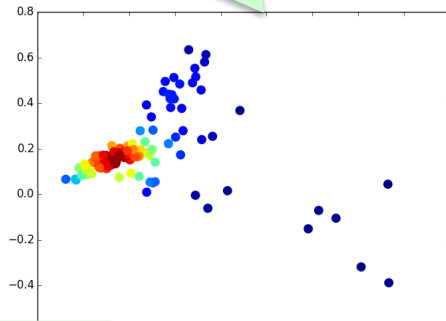
# Example: using PCA for data visualization

Atlanta police data, 20000 police reports, 7200 keywords (bi-gram), map into 2 principle components; shown 2d density estimation.

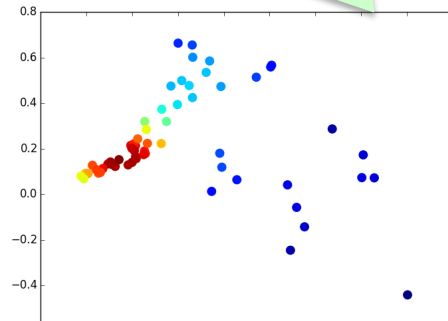
[FRAUD-IMPERS.<\$10,000]



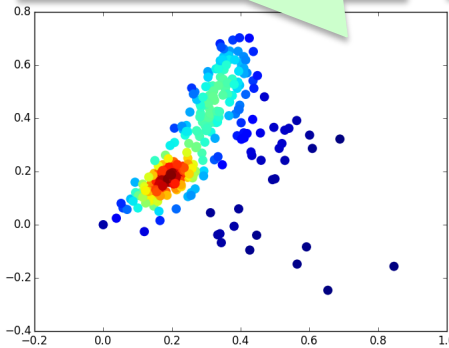
[FRAUD-SWINDLE<\$10,000]



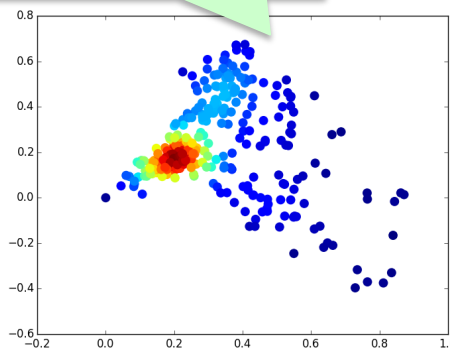
[FRAUD-USE\_OF\_CRCARD<\$10,000]



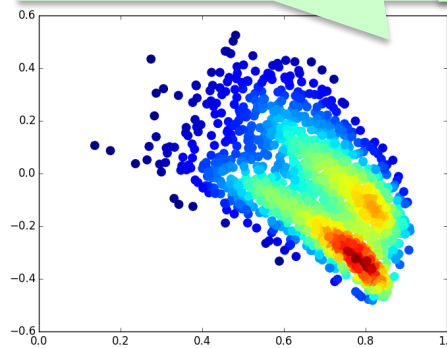
[LARCENY-FROM\_BUILDING]



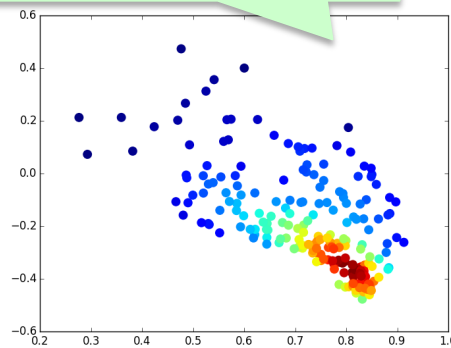
[LARCENY-OTHER\_OFFENSE]



[LAR-FROM\_VEHICLE\_NON-ACC]

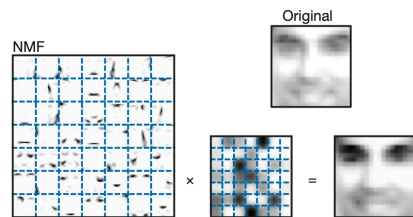


[LAR-FROM\_VEHICLE\_ACCESSORY]



# Extensions/variants of PCA

- Robust PCA: PCA is robust to outliers; it is a common practice to remove outliers to perform PCA; can be cast as a convex optimization problem (Candes, Li, Ma, Wright, 2009)
- Sparse PCA: traditional PCA combines all variables (using the weight vector), not ideal for high-dimensional data; sparse PCA combines just a few important features (solved by optimization) (Johnstone, Yu, 2009)
- Nonlinear PCA: kernelized PCA
- Nonnegative matrix factorization (NMF)
- ICA (independent component analysis):  
decompose the signal into additive components



Lee, Seung, 1999, Nature.