# Principal Component Analysis

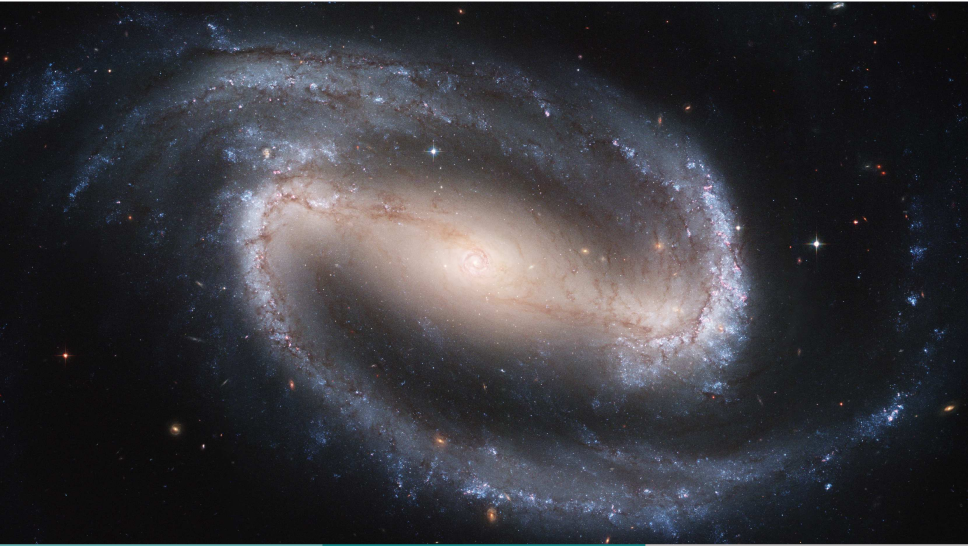
CS5240 Theoretical Foundations in Multimedia

Leow Wee Kheng

Department of Computer Science
School of Computing
National University of Singapore

# Motivation

How wide is the widest part of NGC 1300?

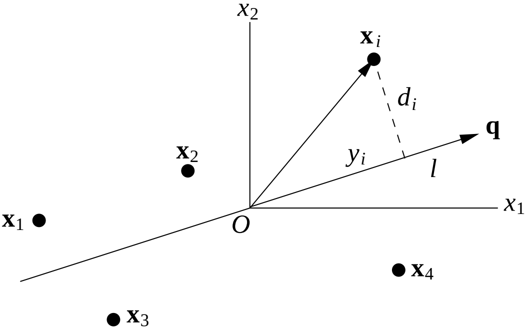


## How thick is the thickest part of NGC 4594?

Use **principal component analysis**.

# Maximum Variance Estimate

Consider a set of points $\mathbf{x}_i$, $i = 1, \ldots, n$, in an $m$-dimensional space such that their mean $\boldsymbol{\mu} = \mathbf{0}$, i.e., centroid is at the origin.




Want to find a line $l$ through the origin that maximizes the projections $y_i$ of the points $\mathbf{x}_i$ on $l$.

Let $\mathbf{q}$ denote the unit vector along line $l$.
Then, the projection $y_i$ of $\mathbf{x}_i$ on $l$ is

$$y_i = \mathbf{x}_i^\top \mathbf{q}. \tag{1}$$

The mean squared projection, which is the variance, $V$ over all points is

$$V = \frac{1}{n} \sum_{i=1}^{n} y_i^2 = \frac{1}{n} \sum_{i=1}^{n} (\mathbf{x}_i^\top \mathbf{q})^2 \geq 0. \tag{2}$$

Because $\mathbf{x}_i^\top \mathbf{q} = \mathbf{q}^\top \mathbf{x}_i$, expanding Eq. 2 gives

$$V = \frac{1}{n} \sum_{i=1}^{n} (\mathbf{q}^\top \mathbf{x}_i)(\mathbf{x}_i^\top \mathbf{q}) = \mathbf{q}^\top \left[ \frac{1}{n} \sum_{i=1}^{n} \mathbf{x}_i \mathbf{x}_i^\top \right] \mathbf{q}. \tag{3}$$

The middle factor is the covariance matrix $\mathbf{C}$ of the data points

$$\mathbf{C} = \frac{1}{n} \sum_{i=1}^{n} \mathbf{x}_i \mathbf{x}_i^\top. \tag{4}$$

We want to find a unit vector $\mathbf{q}$ that maximizes the variance $V$, i.e.,

$$\text{maximize } V = \mathbf{q}^\top \mathbf{C}\,\mathbf{q} \quad \text{subject to } \|\mathbf{q}\| = 1. \tag{5}$$

This is a constrained optimization problem.

Use Lagrange multiplier method:
combine $V$ and the constraint using Lagrange multiplier $\lambda$

$$\text{maximize } \; V' = \mathbf{q}^\top \mathbf{C}\,\mathbf{q} - \lambda(\mathbf{q}^\top \mathbf{q} - 1). \tag{6}$$

# Lagrange Multiplier Method

Lagrange multiplier is a method for solving constrained optimization.

Consider this problem:

$$\text{maximize } f(\mathbf{x}) \quad \text{subject to } g(\mathbf{x}) = c. \tag{7}$$

Lagrange multiplier method introduces a Lagrange multiplier $\lambda$ to combine $f(\mathbf{x})$ and $g(\mathbf{x})$:

$$L(\mathbf{x}, \lambda) = f(\mathbf{x}) + \lambda\,(g(\mathbf{x}) - c). \tag{8}$$

The sign of $\lambda$ can be positive or negative.

Then, solve for the stationary point of $L(\mathbf{x}, \lambda)$:

$$\frac{\partial L(\mathbf{x}, \lambda)}{\partial \mathbf{x}} = 0. \tag{9}$$

- If $\mathbf{x}_0$ is a solution of the original problem, then there is a $\lambda_0$ such that $(\mathbf{x}_0, \lambda_0)$ is a stationary point of $L$.
- Not all stationary points yield solutions of the original problem.
- The same method applies to minimization problem.
- Multiple constraints can be combined by adding multiple terms.

$$\text{minimize } f(\mathbf{x}) \quad \text{subject to } g_1(\mathbf{x}) = c_1,\ g_2(\mathbf{x}) = c_2 \tag{10}$$

is solved with

$$L(\mathbf{x}) = f(\mathbf{x}) + \lambda_1(g_1(\mathbf{x}) - c_1) + \lambda_2(g_2(\mathbf{x}) - c_2). \tag{11}$$

Now, we differentiate $V'$ with respect to $\mathbf{q}$ and set to 0:

$$\frac{\partial V'}{\partial \mathbf{q}} = 2\mathbf{q}^\top \mathbf{C} - 2\lambda\, \mathbf{q}^\top = 0. \tag{12}$$

Rearranging the terms gives

$$\mathbf{q}^\top \mathbf{C} = \lambda\, \mathbf{q}^\top. \tag{13}$$

Since the covariance matrix $\mathbf{C}$ is symmetric (homework), $\mathbf{C}^\top = \mathbf{C}$. So, transposing both sides of Eq. 13 gives

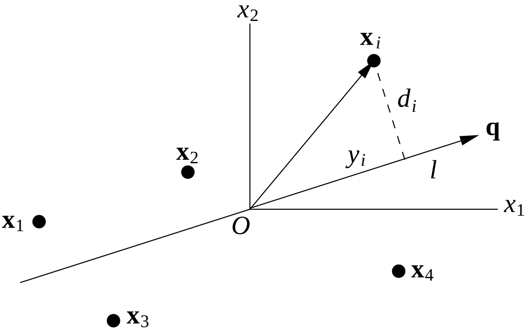
$$\mathbf{C}\,\mathbf{q} = \lambda\, \mathbf{q}. \tag{14}$$

- Eq. 14 is called an eigenvector equation.
- $\mathbf{q}$ is the eigenvector.
- $\lambda$ is the eigenvalue.

Thus, the eigenvector $\mathbf{q}$ of $\mathbf{C}$ gives the line that maximizes variance $V$.

The perpendicular distance $d_i$ of $\mathbf{x}_i$ from the line $l$ is

$$d_i = \|\mathbf{x}_i - y_i \mathbf{q}\| = \|\mathbf{x}_i - (\mathbf{x}_i^\top \mathbf{q})\, \mathbf{q}\|. \tag{15}$$



The squared distance is

$$\begin{aligned} d_i^2 &= \|\mathbf{x}_i - (\mathbf{x}_i^\top \mathbf{q})\,\mathbf{q}\|^2 = (\mathbf{x}_i - (\mathbf{x}_i^\top \mathbf{q})\,\mathbf{q})^\top (\mathbf{x}_i - (\mathbf{x}_i^\top \mathbf{q})\,\mathbf{q}) \\ &= \mathbf{x}_i^\top \mathbf{x}_i - \mathbf{x}_i^\top (\mathbf{x}_i^\top \mathbf{q})\mathbf{q} - (\mathbf{x}_i^\top \mathbf{q})\mathbf{q}^\top \mathbf{x}_i + (\mathbf{x}_i^\top \mathbf{q})\mathbf{q}^\top (\mathbf{x}_i^\top \mathbf{q})\mathbf{q} \end{aligned}$$

With $\mathbf{x}_i^\top \mathbf{q}$ being a scalar and $\mathbf{x}_i^\top \mathbf{q} = \mathbf{q}^\top \mathbf{x}_i$, we obtain

$$d_i^2 = \mathbf{x}_i^\top \mathbf{x}_i - (\mathbf{x}_i^\top \mathbf{q})^2.$$

Averaging over all $i$ gives

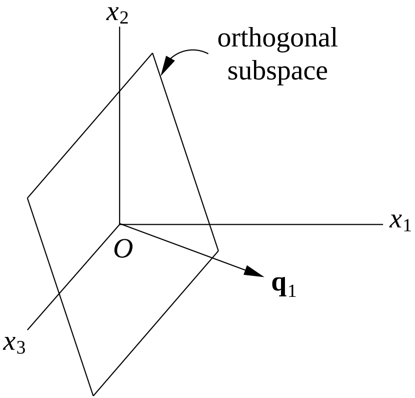
$$D = \frac{1}{n} \sum_{i=1}^{n} d_i^2 = \frac{1}{n} \sum_{i=1}^{n} \mathbf{x}_i^\top \mathbf{x}_i - V. \tag{16}$$

So, maximizing variance $V$ means
minimizing mean squared distance $D$ to data points $\mathbf{x}_i$.

The eigenvalue $\lambda = V$, the variance (homework); so $\lambda \geq 0$.

Name $\mathbf{q}$ as the first eigenvector $\mathbf{q}_1$.
The component of $\mathbf{x}_i$ orthogonal to $\mathbf{q}_1$ is $\mathbf{x}'_i = \mathbf{x}_i - \mathbf{x}_i^\top \mathbf{q}_1 \mathbf{q}_1$.

Repeat previous method on $\mathbf{x}'_i$ in an $(m-1)$-D subspace orthogonal to $\mathbf{q}_1$ to get $\mathbf{q}_2$. Then, repeat to get $\mathbf{q}_3, \ldots, \mathbf{q}_m$.



# Eigendecomposition

In general, a $m \times m$ covariance matrix $\mathbf{C}$ has $m$ eigenvectors:

$$\mathbf{C}\,\mathbf{q}_j = \lambda_j\,\mathbf{q}_j, \quad j = 1, \ldots, m. \tag{17}$$

Transpose both sides of the equations to get

$$\mathbf{q}_j^\top \mathbf{C} = \lambda_j\,\mathbf{q}_j^\top, \quad j = 1, \ldots, m, \tag{18}$$

which are row matrices. Stack the row matrices into a column to get

$$\begin{bmatrix} \mathbf{q}_1^\top \\ \vdots \\ \mathbf{q}_m^\top \end{bmatrix} \mathbf{C} = \begin{bmatrix} \lambda_1 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \lambda_m \end{bmatrix} \begin{bmatrix} \mathbf{q}_1^\top \\ \vdots \\ \mathbf{q}_m^\top \end{bmatrix} \tag{19}$$

Denote matrix $\mathbf{Q}$ and $\mathbf{\Lambda}$ as

$$\mathbf{Q} = [\mathbf{q}_1 \ \cdots \ \mathbf{q}_m], \quad \mathbf{\Lambda} = \mathrm{diag}(\lambda_1, \ldots, \lambda_m). \tag{20}$$

Then, Eq. 19 becomes

$$\mathbf{Q}^\top \mathbf{C} = \mathbf{\Lambda} \mathbf{Q}^\top \tag{21}$$

or

$$\mathbf{C} = \mathbf{Q} \mathbf{\Lambda} \mathbf{Q}^\top. \tag{22}$$

This is the matrix equation for eigendecomposition.

**Properties:**

- The eigenvectors are orthonormal:

$$\begin{aligned} \mathbf{q}_j^\top \mathbf{q}_j &= 1, \\ \mathbf{q}_j^\top \mathbf{q}_k &= 0, \quad \text{for } k \neq j. \end{aligned} \tag{23}$$

  So, the eigenmatrix $\mathbf{Q}$ is orthogonal:

$$\mathbf{Q}^\top \mathbf{Q} = \mathbf{Q}\,\mathbf{Q}^\top = \mathbf{I}. \tag{24}$$

- The eigenvalues are arranged to be sorted $\lambda_j \geq \lambda_{j+1}$.

# General PCA

In general, the mean $\boldsymbol{\mu}$ of the data points $\mathbf{x}_i$ is **not** at the origin.
In this case, we subtract $\boldsymbol{\mu}$ from each $\mathbf{x}_i$,
obtaining shifted or zero-mean data points $\mathbf{x}_i - \boldsymbol{\mu}$.

PCA transforms $\mathbf{x}_i$ into a new vector $\mathbf{y}_i$ through $\mathbf{Q}$ as follows:

$$\mathbf{y}_i = \mathbf{Q}^\top(\mathbf{x}_i - \boldsymbol{\mu}) = \begin{bmatrix} \mathbf{q}_1^\top(\mathbf{x}_i - \boldsymbol{\mu}) \\ \vdots \\ \mathbf{q}_m^\top(\mathbf{x}_i - \boldsymbol{\mu}) \end{bmatrix} = \sum_{j=1}^{m} (\mathbf{x}_i - \boldsymbol{\mu})^\top \mathbf{q}_j \, \mathbf{q}_j, \tag{25}$$

Each component of $\mathbf{y}_i$ is

$$y_{ij} = (\mathbf{x}_i - \boldsymbol{\mu})^\top \mathbf{q}_j. \tag{26}$$

This is the projection of $\mathbf{x}_i - \boldsymbol{\mu}$ on $\mathbf{q}_j$.

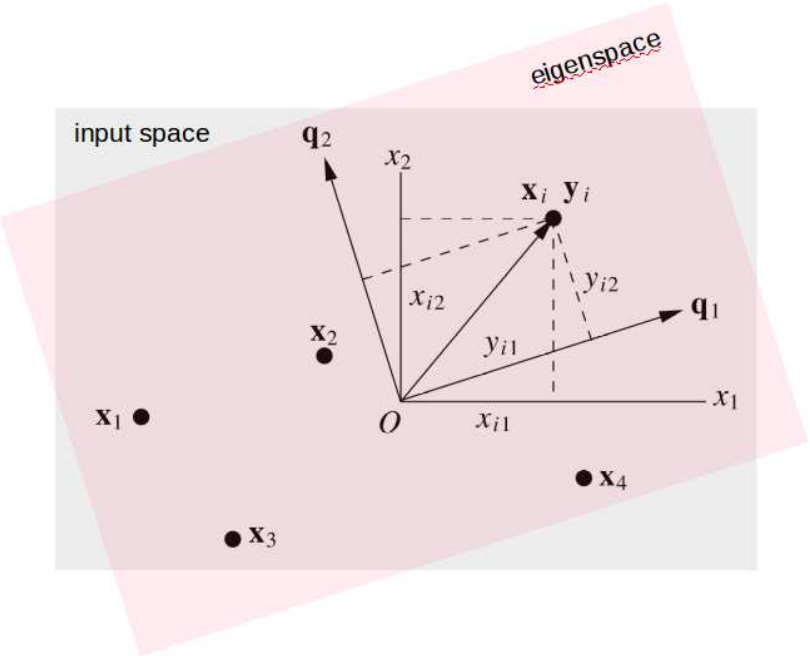
The original $\mathbf{x}_i$ can be recovered from $\mathbf{y}_i$:

$$\mathbf{x}_i = \mathbf{Q}\,\mathbf{y}_i + \boldsymbol{\mu}. \tag{27}$$

**Caution!**

- $\mathbf{x}_i \neq \mathbf{y}_i + \boldsymbol{\mu}$.
- $\mathbf{x}_i$ is in the original input space but $\mathbf{y}_i$ is in the eigenspace. Let $\hat{\mathbf{x}}_j$ denote the unit vectors that form the input space. $\mathbf{q}_j$ are the eigenvectors that form the eigenspace. Then,

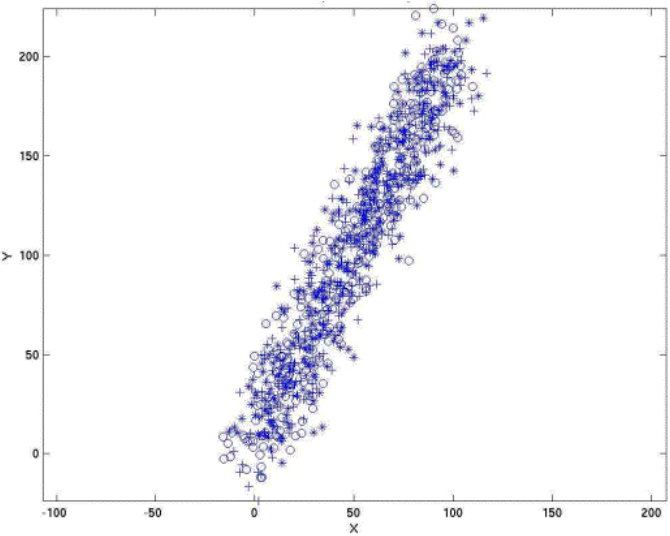
$$\begin{aligned}\mathbf{x}_i &= (x_{i1}, x_{i2}, \ldots, x_{im}) &= \sum_{j=1}^{m} x_{ij}\hat{\mathbf{x}}_j, \\ \mathbf{y}_i &= (y_{i1}, y_{i2}, \ldots, y_{im}) &= \sum_{j=1}^{m} y_{ij}\mathbf{q}_j.\end{aligned}$$


eigenspace
input space
$\mathbf{q}_2$
$x_2$
$\mathbf{x}_i$ $\mathbf{y}_i$
$y_{i2}$
$x_{i2}$
$\mathbf{q}_1$
$\mathbf{x}_2$
$y_{i1}$
$x_1$
$\mathbf{x}_1$
$O$
$x_{i1}$
$\mathbf{x}_4$
$\mathbf{x}_3$

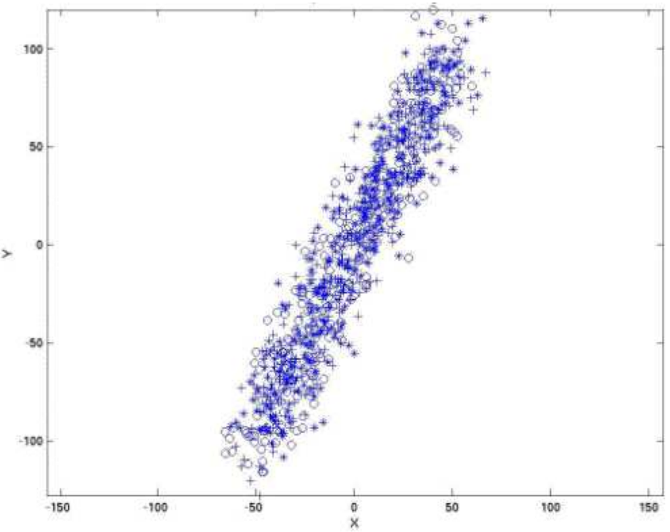
# Properties of PCA

- Mean $\boldsymbol{\mu}_y$ over all $\mathbf{y}_i$ is $\mathbf{0}$ (homework).
- Variance $\sigma_j^2$ along $\mathbf{q}_j$ is $\lambda_j$ (homework).
- Since $\lambda_1 \geq \cdots \geq \lambda_m \geq 0$, so $\sigma_1 \geq \cdots \geq \sigma_m \geq 0$.
- $\mathbf{q}_1$ gives orientation of the largest variance.
- $\mathbf{q}_2$ gives orientation of largest variance orthogonal to $\mathbf{q}_1$ (2nd largest variance).
- $\mathbf{q}_j$ gives orientation of largest variance orthogonal to $\mathbf{q}_1, \ldots, \mathbf{q}_{j-1}$ ($j$-th largest variance).
- $\mathbf{q}_m$ is orthogonal to all other eigenvectors (least variance).

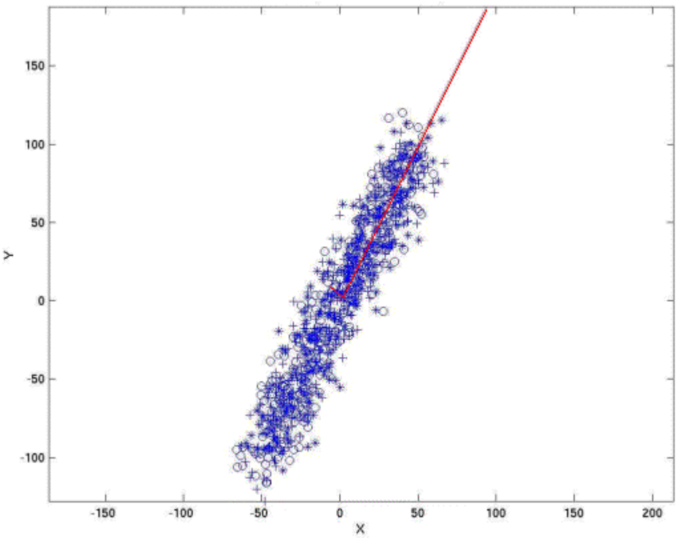
# Data Points



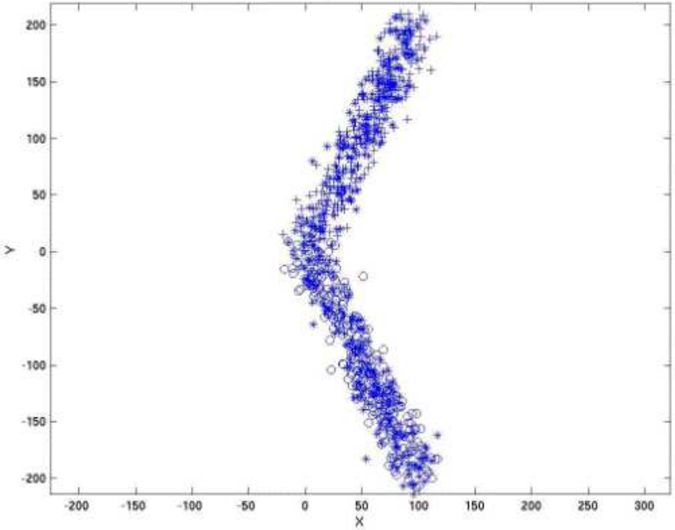
# Centroid at Origin



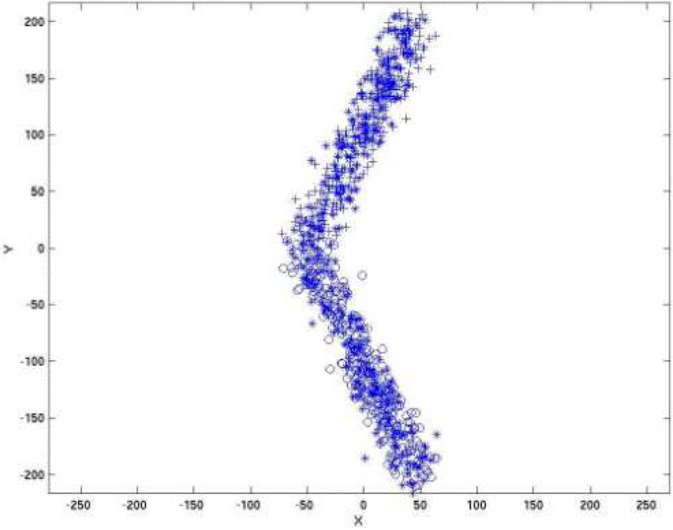
# Principal Components



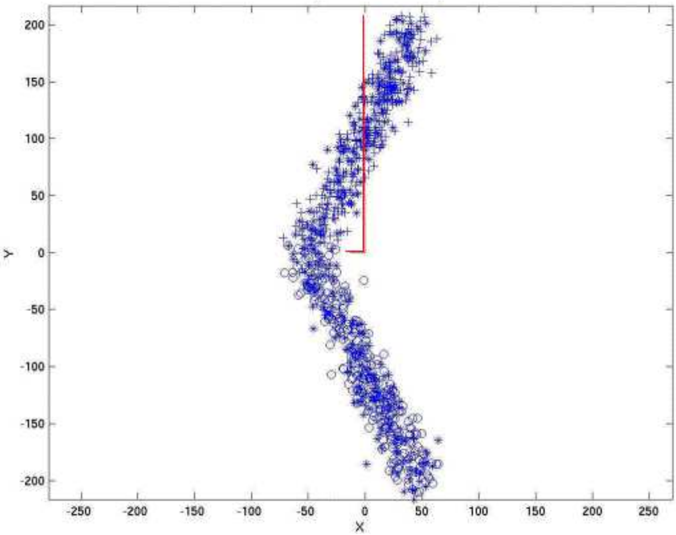
# Another Example



# Centroid at Origin



# Principal Components



# PCA Algorithm 1

Let $\mathbf{x}_i$ denote $m$-dimensional vectors (data points), $i = 1, ..., n$.

1. Compute the mean vector of the data points

$$\boldsymbol{\mu} = \frac{1}{n}\sum_{i=1}^{n} \mathbf{x}_i. \tag{28}$$

2. Compute the covariance matrix

$$\mathbf{C} = \frac{1}{n}\sum_{i=1}^{n} (\mathbf{x}_i - \boldsymbol{\mu})(\mathbf{x}_i - \boldsymbol{\mu})^\top. \tag{29}$$

3. Perform eigendecomposition of $\mathbf{C}$

$$\mathbf{C} = \mathbf{Q}\boldsymbol{\Lambda}\mathbf{Q}^\top. \tag{30}$$

- Some books and papers use sample covariance

$$\mathbf{C} = \frac{1}{n-1}\sum_{i=1}^{n}(\mathbf{x}_i - \boldsymbol{\mu})(\mathbf{x}_i - \boldsymbol{\mu})^\top. \tag{31}$$

Eq. 31 differs from Eq. 29 by only a constant.

Notes:

- Covariance matrix **C** is a $m \times m$ matrix.
- Eigendecomposition of very large matrix is inefficient.
- Example:
  - A $256 \times 256$ colour image has $256 \times 256 \times 3$ values.
  - Number of dimensions $m = 196680$.
  - **C** has a size of $196608 \times 196608$!!
  - Number of images $n$ is usually $\ll m$, e.g., 1000.

# PCA Algorithm 2

1. Compute mean $\boldsymbol{\mu}$ of data points $\mathbf{x}_i$.

$$\boldsymbol{\mu} = \frac{1}{n} \sum_{i=1}^{n} \mathbf{x}_i.$$

2. Form a $m \times n$ matrix $\mathbf{A}$, $n \ll m$:

$$\mathbf{A} = [(\mathbf{x}_1 - \boldsymbol{\mu}) \ \cdots \ (\mathbf{x}_n - \boldsymbol{\mu})]. \tag{32}$$

3. Compute $\mathbf{A}^\top \mathbf{A}$, which is just a $n \times n$ matrix.
4. Apply eigendecomposition on $\mathbf{A}^\top \mathbf{A}$:

$$(\mathbf{A}^\top \mathbf{A})\, \mathbf{q}_j = \lambda_j \mathbf{q}_j \,. \tag{33}$$

5. Pre-multiply $\mathbf{A}$ to Eq. 33 giving

$$\mathbf{A}\mathbf{A}^\top \mathbf{A}\mathbf{q}_j = \mathbf{A}\lambda_j \mathbf{q}_j. \tag{34}$$

6. Recover eigenvectors and eigenvalues.

   Since $\mathbf{A}\mathbf{A}^\top = n\mathbf{C}$ (homework), Eq. 34 is

$$
\begin{aligned}
n\mathbf{C}\,(\mathbf{A}\mathbf{q}_j) &= \lambda_j(\mathbf{A}\mathbf{q}_j) && (35)\\
\mathbf{C}\,(\mathbf{A}\mathbf{q}_j) &= \frac{\lambda_j}{n}(\mathbf{A}\mathbf{q}_j). && (36)
\end{aligned}
$$

   Therefore, eigenvectors of $\mathbf{C}$ are $\mathbf{A}\mathbf{q}_j$, and eigenvalues of $\mathbf{C}$ are $\lambda_j/n$.

# PCA Algorithm 3

1. Compute mean $\boldsymbol{\mu}$ of data points $\mathbf{x}_i$.

$$\boldsymbol{\mu} = \frac{1}{n} \sum_{i=1}^{n} \mathbf{x}_i.$$

2. Form a $m \times n$ matrix $\mathbf{A}$:

$$\mathbf{A} = [(\mathbf{x}_1 - \boldsymbol{\mu}) \ \cdots \ (\mathbf{x}_n - \boldsymbol{\mu})].$$

3. Apply singular value decomposition (SVD) on $\mathbf{A}$:

$$\mathbf{A} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^\top.$$

4. Recover eigenvectors and eigenvalues (page 31).

# Singular Value Decomposition

Singular value decomposition (SVD) decomposes a matrix $\mathbf{A}$ into

$$\mathbf{A} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^\top. \tag{37}$$

- Column vectors of $\mathbf{U}$ are left singular vectors $\mathbf{u}_j$, and $\mathbf{u}_j$ are orthonormal.
- Column vectors of $\mathbf{V}$ are right singular vectors $\mathbf{v}_j$, and $\mathbf{v}_j$ are orthonormal.
- $\boldsymbol{\Sigma}$ is diagonal and contains singular values $s_j$.
- Rank of $\mathbf{A}$ = number of non-zero singular values.

Notice that

$$\mathbf{A}\mathbf{A}^\top = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^\top \left(\mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^\top\right)^\top = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^\top\mathbf{V}\boldsymbol{\Sigma}\mathbf{U}^\top = \mathbf{U}\boldsymbol{\Sigma}^2\mathbf{U}^\top$$

and eigendecomposition of $\mathbf{A}\mathbf{A}^\top$ is

$$\mathbf{A}\mathbf{A}^\top = \mathbf{Q}\boldsymbol{\Lambda}\mathbf{Q}^\top.$$

So, eigenvector of $\mathbf{A}\mathbf{A}^\top = \mathbf{u}_j$, eigenvalue of $\mathbf{A}\mathbf{A}^\top = s_j^2$.

On the other hand,

$$\mathbf{A}^\top\mathbf{A} = \left(\mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^\top\right)^\top \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^\top = \mathbf{V}\boldsymbol{\Sigma}\mathbf{U}^\top\mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^\top = \mathbf{V}\boldsymbol{\Sigma}^2\mathbf{V}^\top.$$

Compare with the eigendecomposition of $\mathbf{A}^\top\mathbf{A}$:

$$\mathbf{A}^\top\mathbf{A} = \mathbf{Q}\boldsymbol{\Lambda}\mathbf{Q}^\top.$$

So, eigenvector of $\mathbf{A}^\top\mathbf{A} = \mathbf{v}_j$, eigenvalue of $\mathbf{A}^\top\mathbf{A} = s_j^2$.

4. Recover eigenvectors and eigenvalues.

   Compare $\mathbf{A}\mathbf{A}^\top = n\mathbf{C}$ with eigendecomposition of $\mathbf{C}$:

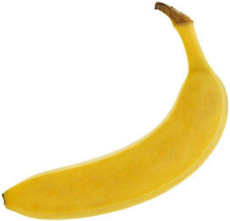
$$\mathbf{A}\mathbf{A}^\top = \mathbf{U}\mathbf{\Sigma}^2\mathbf{U}^\top,$$

$$\mathbf{C} = \mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^\top.$$

   Therefore, eigenvectors $\mathbf{q}_j$ of $\mathbf{C} = \mathbf{u}_j$ in $\mathbf{U}$, and eigenvalues $\lambda_j$ of $\mathbf{C} = s_j^2/n$, for $s_j$ in $\mathbf{\Sigma}$.

# Application Examples

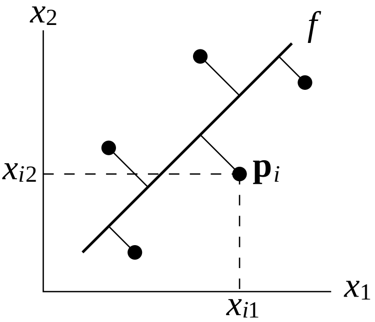
Identify the 1st and 2nd major axes of these objects.

Apply PCA for line fitting in 2-D.

Compute PCA of the points. Then,

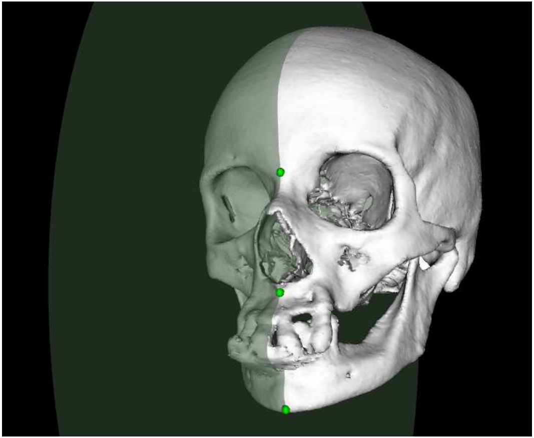
- mean of points = a point on line
- 1st eigenvector = unit vector along line



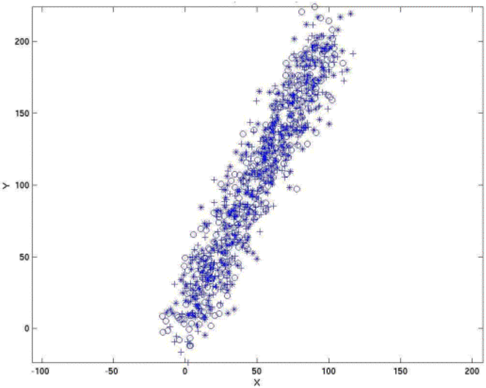
Apply PCA for plane fitting in 3-D.

Compute PCA of the points. Then,

- mean of points = a point on plane
- 3rd eigenvector = unit normal vector of plane



# Dimensionality Reduction




If a point represents a $256 \times 256$ colour image,
then the dimensionality is $256 \times 256 \times 3 = 196680$!!

But, not all 196680 eigenvalues are non-zero.

**Case 1:** Number of data vectors $n \leq m$ number of dimensions.

- Data vectors are all independent.
  Then, number of non-zero eigenvalues (eigenvectors) $= n - 1$,
  i.e., rank of covariance matrix $= n - 1$.
  Why not $= n$?

- Data vectors are not independent.
  Then, rank of covariance matrix $< n - 1$.

**Case 2:** Number of data vectors $n > m$.

- $m$ or more independent data vectors.
  Then, rank of covariance matrix $= m$.

- Fewer than $m$ data vectors are independent.
  In this case, what is the rank of covariance matrix?

In practice, it is often possible to reduce the dimensionality.

Eigenmatrix $\mathbf{Q}$ is

$$\mathbf{Q} = [\mathbf{q}_1 \ \cdots \ \mathbf{q}_m]\,. \tag{38}$$

PCA maps a data point $\mathbf{x}$ to a vector $\mathbf{y}$ in the eigenspace as

$$\mathbf{y} = \mathbf{Q}^\top(\mathbf{x} - \boldsymbol{\mu}) = \sum_{j=1}^{m} (\mathbf{x} - \boldsymbol{\mu})^\top \mathbf{q}_j \mathbf{q}_j, \tag{39}$$

which has $m$ dimensions.

Pick $l$ eigenvectors with the largest eigenvalues to form truncated $\widehat{\mathbf{Q}}$:

$$\widehat{\mathbf{Q}} = [\mathbf{q}_1 \ \cdots \ \mathbf{q}_l], \tag{40}$$

which spans a subspace of the eigenspace.

Then, $\widehat{\mathbf{Q}}$ maps $\mathbf{x}$ to $\widehat{\mathbf{y}}$, an estimate of $\mathbf{y}$:

$$\widehat{\mathbf{y}} = \widehat{\mathbf{Q}}^\top(\mathbf{x} - \boldsymbol{\mu}) = \sum_{j=1}^{l} (\mathbf{x} - \boldsymbol{\mu})^\top \mathbf{q}_j \mathbf{q}_j, \tag{41}$$

which has $l < m$ dimensions.

$$\begin{array}{ccccccc}
\mathbf{x} & & \mathbf{y} & & \widehat{\mathbf{y}} & & \mathbf{x} \\
\begin{bmatrix} x_1 \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ x_m \end{bmatrix} & \begin{matrix} \xrightarrow{\mathbf{Q}^\top} \\ \xleftarrow{\mathbf{Q}} \end{matrix} & \begin{bmatrix} y_1 \\ \vdots \\ y_l \\ y_{l+1} \\ \vdots \\ y_m \end{bmatrix} & & \begin{bmatrix} y_1 \\ \vdots \\ y_l \end{bmatrix} & \begin{matrix} \xleftarrow{\widehat{\mathbf{Q}}^\top} \\ \not\xrightarrow{\widehat{\mathbf{Q}}} \end{matrix} & \begin{bmatrix} x_1 \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ x_m \end{bmatrix}
\end{array}$$

dimensionality reduction

Difference between $\mathbf{y}$ and $\widehat{\mathbf{y}}$ is

$$\mathbf{y} - \widehat{\mathbf{y}} = \sum_{j=l+1}^{m} (\mathbf{x} - \boldsymbol{\mu})^\top \mathbf{q}_j \mathbf{q}_j. \tag{42}$$

Beware:

$$\mathbf{y} = \mathbf{Q}^\top (\mathbf{x} - \boldsymbol{\mu})$$

and, therefore,

$$\mathbf{x} = \mathbf{Q}\,\mathbf{y} + \boldsymbol{\mu}.$$

But,

$$\widehat{\mathbf{y}} = \widehat{\mathbf{Q}}^\top (\mathbf{x} - \boldsymbol{\mu}),$$

whereas

$$\widehat{\mathbf{x}} = \widehat{\mathbf{Q}}\,\widehat{\mathbf{y}} + \boldsymbol{\mu} \neq \mathbf{x}. \tag{43}$$

Why?

With $n$ data points $\mathbf{x}_i$, sum-squared error $E$ between $\mathbf{x}_i$ and its estimate $\widehat{\mathbf{x}}_i$ is

$$E = \sum_{i=1}^{n} \|\widehat{\mathbf{x}}_i - \mathbf{x}_i\|^2. \tag{44}$$

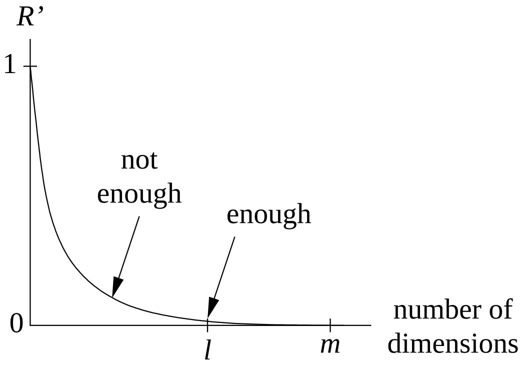
When all $m$ dimensions are kept, the total variance is

$$\sum_{j=1}^{m} \sigma_j^2 = \sum_{j=1}^{m} \lambda_j. \tag{45}$$

When $l$ dimensions are used, the ratio $R'$ of unaccounted variance is:

$$R'(l) = \frac{\displaystyle\sum_{j=l+1}^{m} \sigma_j^2}{\displaystyle\sum_{j=1}^{m} \sigma_j^2} = \frac{\displaystyle\sum_{j=l+1}^{m} \lambda_j}{\displaystyle\sum_{j=1}^{m} \lambda_j}. \tag{46}$$

A sample plot of $R'$ vs. number of dimensions used:

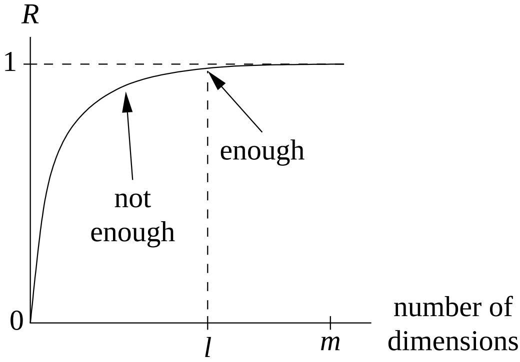



How to choose appropriate $l$?
Choose $l$ such that larger than $l$ doesn't reduce $R'$ significantly:

$$R'(l) - R'(l+1) < \epsilon. \tag{47}$$

Alternatively, compute ratio $R$ of accounted variance:

$$R(l) = \frac{\displaystyle\sum_{j=1}^{l} \sigma_j^2}{\displaystyle\sum_{j=1}^{m} \sigma_j^2} = \frac{\displaystyle\sum_{j=1}^{l} \lambda_j}{\displaystyle\sum_{j=1}^{m} \lambda_j}. \tag{48}$$




$l$ is large enough if
$R(l+1) - R(l) < \epsilon$.

# Summary

- Eigendecomposition of covariance matrix = PCA.
- In practice, PCA is computed using SVD.
- PCA maximizes variances along the eigenvectors.
- Eigenvalues = variances along eigenvectors.
- Best fitting plane computed by PCA minimizes distance to points.
- PCA can be used for dimensionality reduction.

# Probing Questions

- When applying PCA to plane fitting, how to check whether the points really lie close to the plane?
- If you apply PCA to a set of points on a curve surface, where do you expect the eigenvectors to point at?
- In dimensionality reduction, the $m$-D vector $\mathbf{y}$ is reduced to a $l$-D vector $\widehat{\mathbf{y}}$. Since $l \neq m$, vector subtraction is undefined. But, subtraction of Eq. 39 and 41 gives

$$\mathbf{y} - \widehat{\mathbf{y}} = \sum_{j=l+1}^{m} (\mathbf{x} - \boldsymbol{\mu})^\top \mathbf{q}_j \mathbf{q}_j.$$

  How is this possible? Is there a contradiction?
- Show that when $n \leq m$, there are at most $n - 1$ eigenvectors.

# Homework I

1. Describe the essence of PCA in one sentence.
2. Show that the covariance matrix $\mathbf{C}$ of a set of data points $\mathbf{x}_i$, $i = 1, \ldots, n$, is symmetric about its diagonal.
3. Show that the eigenvalue $\lambda$ of covariance matrix $\mathbf{C}$ equals the variance $V$ as defined in Eq. 2.
4. Show that the mean $\boldsymbol{\mu}_y$ over all $\mathbf{y}_i$ in the eigenspace is $\mathbf{0}$.
5. Show that the variance $\sigma_j^2$ along eigenvector $\mathbf{q}_j$ is $\lambda_j$.
6. Show that $\mathbf{A}\mathbf{A}^\top = n\mathbf{C}$.
7. Derive the difference $\mathbf{Q}\,\mathbf{y} - \widehat{\mathbf{Q}}\,\widehat{\mathbf{y}}$ in dimensionality reduction.

# Homework II

8. Suppose $n \leq m$, and the truncated eigenmatrix $\widehat{\mathbf{Q}}$ contains all the eigenvectors with non-zero eigenvalues, $\widehat{\mathbf{Q}} = [\mathbf{q}_1 \cdots \mathbf{q}_{n-1}]$.

   Now, map an input vector $\mathbf{x}$ to $\mathbf{y}$ and $\widehat{\mathbf{y}}$ respectively by the complete $\mathbf{Q}$ (Eq. 39) and the truncated $\widehat{\mathbf{Q}}$ (Eq. 41). What is the error $\mathbf{y} - \widehat{\mathbf{y}}$?

   What is the result of mapping $\widehat{\mathbf{y}}$ back to the input space by $\widehat{\mathbf{Q}}$ (Eq. 43)?
9. Q3 of AY2015/16 Final Evaluation.

# References


1. G. Strang, *Introduction to Linear Algebra*, 4th ed., Wellesley-Cambridge, 2009.
   `www-math.mit.edu/~gs`
2. C. Shalizi, *Advanced Data Analysis from an Elementary Point of View*, 2015.
   `www.stat.cmu.edu/~cshalizi/ADAfaEPoV`