Principal Component Analysis

(PCA)

Kullback-Leibler divergence

The directed Kullback-Leibler divergence between Exp(λ_0) ('true' distribution) and Exp(λ) ('approximating' distribution) is given by:

$$\Delta(\lambda_0||\lambda) = \log(\lambda_0) - \log(\lambda) + \frac{\lambda}{\lambda_0} - 1.$$

$$D_{\mathrm{KL}}(P\|Q) = \sum_i P(i) \log \frac{P(i)}{Q(i)}.$$

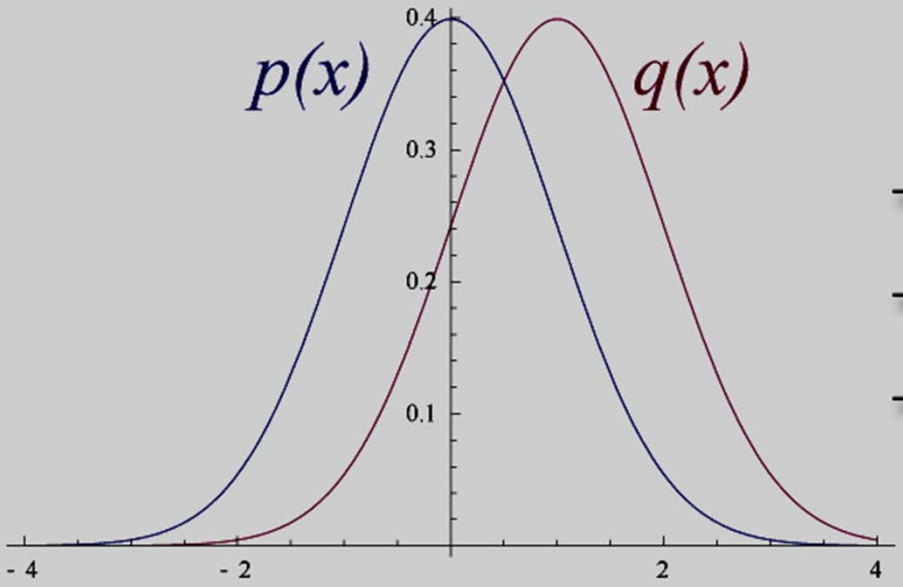
$$D_{KL}(p,q) = \sum_i p(i) \log \frac{p(i)}{q(i)} - \sum p(i) + \sum q(i)$$

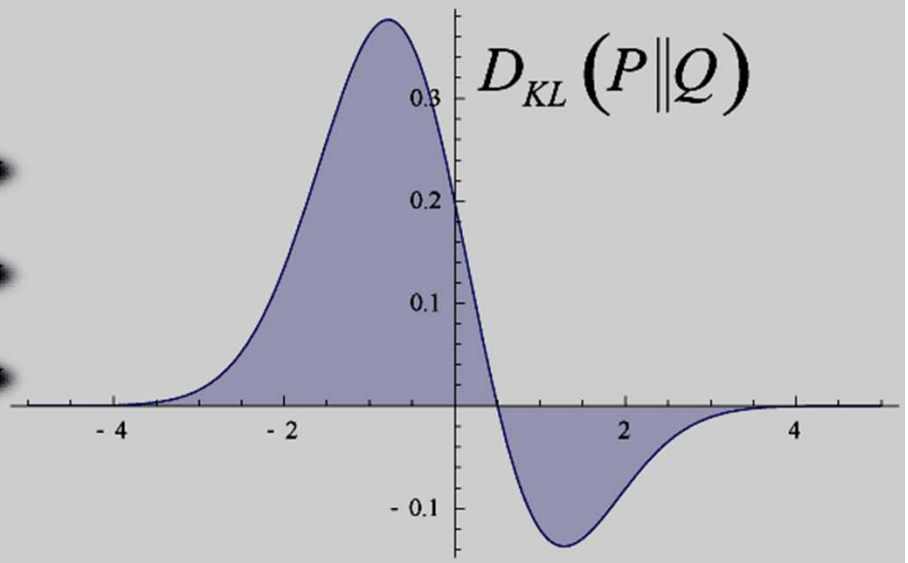
or

$$D_{KL}(p,q) = -\sum_i p(i) \log q(i) + \sum_i p(i) \log p(i)$$

$$= H(p,q) \quad - \quad \mathrm{H(p)}$$

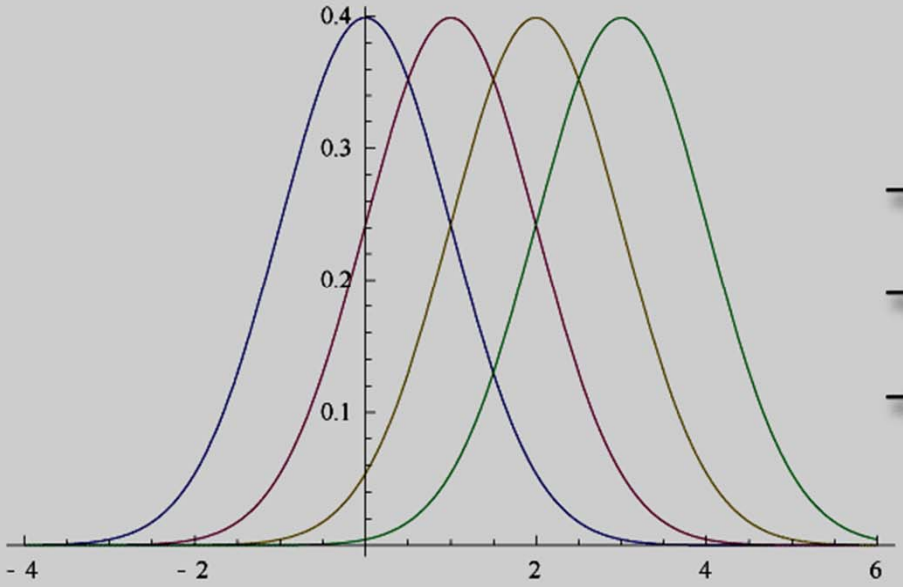
$$= \mathrm{cross\_entropy(P\,\&\,Q)} \quad - \quad \mathrm{entropy(p)}$$

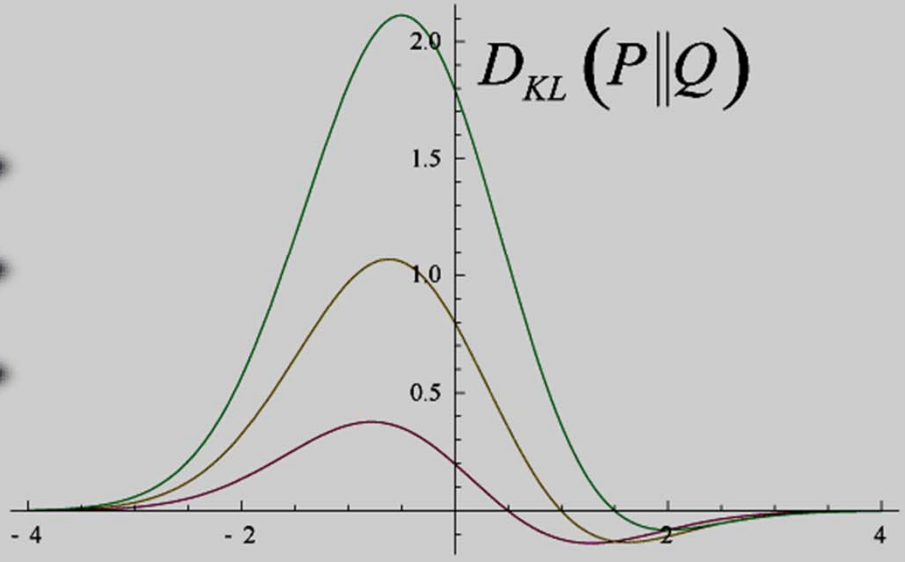




Original Gaussian PDF's

KL Area to be Integrated





- **Bregman divergence**

$$D_{BG}(p,q) = F(p) - F(q) - \langle \nabla F(q), p - q \rangle$$

- **Jensen–Shannon divergence:** **The Bregman distance associated with F for points (P, Q), is the difference between the value of F at point P and the value of the first-order Taylor expansion of F around point Q evaluated at point P. F is a continuously-differentiable real-valued and strictly convex function defined on a closed convex set.**

$$D_{JS}(p,q) = \frac{D_{KL}(P,M) + D(Q,M)}{2}; \text{ where } M = (P+Q)/2$$

- **Deviance information criterion**
- **Bayesian information criterion**
- **Quantum relative entropy**
- **Information gain in decision trees**
- **Solomon Kullback and Richard Leibler**
- **Information theory and measure theory**
- **Entropy power inequality**
- **Information gain ratio**
- **F-divergence**

Principal Component Analysis

- Eigen analysis, Karhunen-Loeve transform
- Eigenvectors: derived from Eigen decomposition of the scatter matrix
- A projection set that best explains the distribution of the representative features of an object of interest.
- PCA techniques choose a dimensionality-reducing linear projection that maximizes the scatter of all projected samples.

Principal Component Analysis Contd.

- Let us consider a set of N sample images $\{x_1, x_2, \ldots\ldots, x_N\}$ taking values in n-dimensional image space.
- Each image belongs to one of c classes $\{X_1, X_2, \ldots\ldots, X_c\}$.
- Let us consider a linear transformation, mapping the original n-dimensional *image space* to m-dimensional *feature space*, where $m < n$.
- The new feature vectors $y_k \in R^m$ are defined by the linear transformation –

$$y_k = W^T x_k \qquad k = 1, 2, \ldots\ldots, N$$

where, $W \in R^{n \times m}$ is a matrix with orthogonal columns representing the basis in feature space.

Principal Component Analysis Contd..

- **Total scatter matrix S_T is defined as**

$$S_T = \sum_{k=1}^{N} (x_k - \mu)(x_k - \mu)^T$$

where, *N* is the number of samples , and $\mu \in R^n$ is the mean image of all samples .

$$\sigma_{ij} = E[(x_i - \mu_i)(x_j - \mu_j)]$$

- **The scatter of transformed feature vectors $\{y_1, y_2, \ldots, y_N\}$ is $W^T S_T W$.**

- **In PCA, W_{opt} is chosen to maximize the determinant of the total scatter matrix of projected samples, *i.e.*,**

$$W_{opt} = \arg\max_{W} \left| W^T S_T W \right|$$

where $\{w_i \mid i = 1,2,\ldots,m\}$ is the set of *n* dimensional eigenvectors of S_T corresponding to *m* largest eigenvalues (check proof).

Principal Component Analysis Contd.

- Eigenvectors are called eigen images/pictures and also basis images/facial basis for faces.

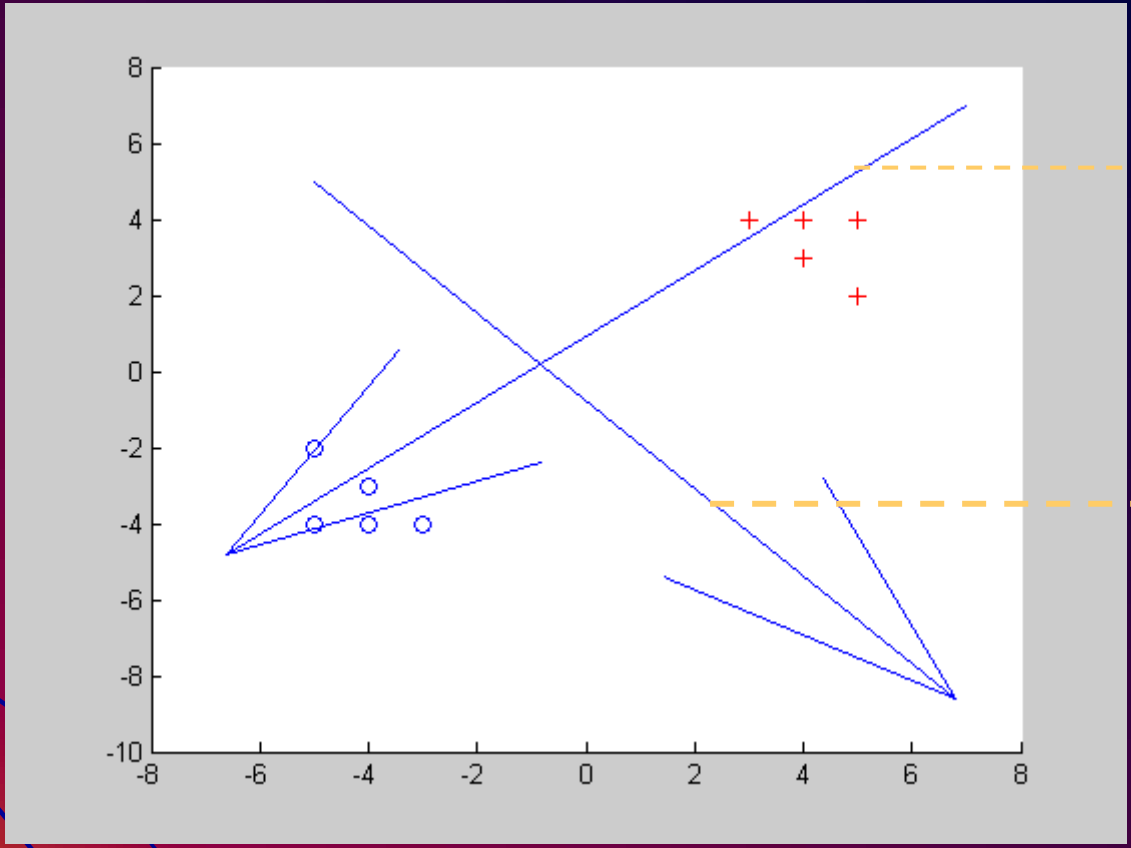
- Any data (say, face) can be reconstructed approximately as a weighted sum of a small collection of images that define a facial basis (eigen images) and a mean image of the face.

- Data form a scatter in the feature space through projection set (eigen vector set)

- Features (eigenvectors) are extracted from the training set without prior class information

➔Unsupervised learning

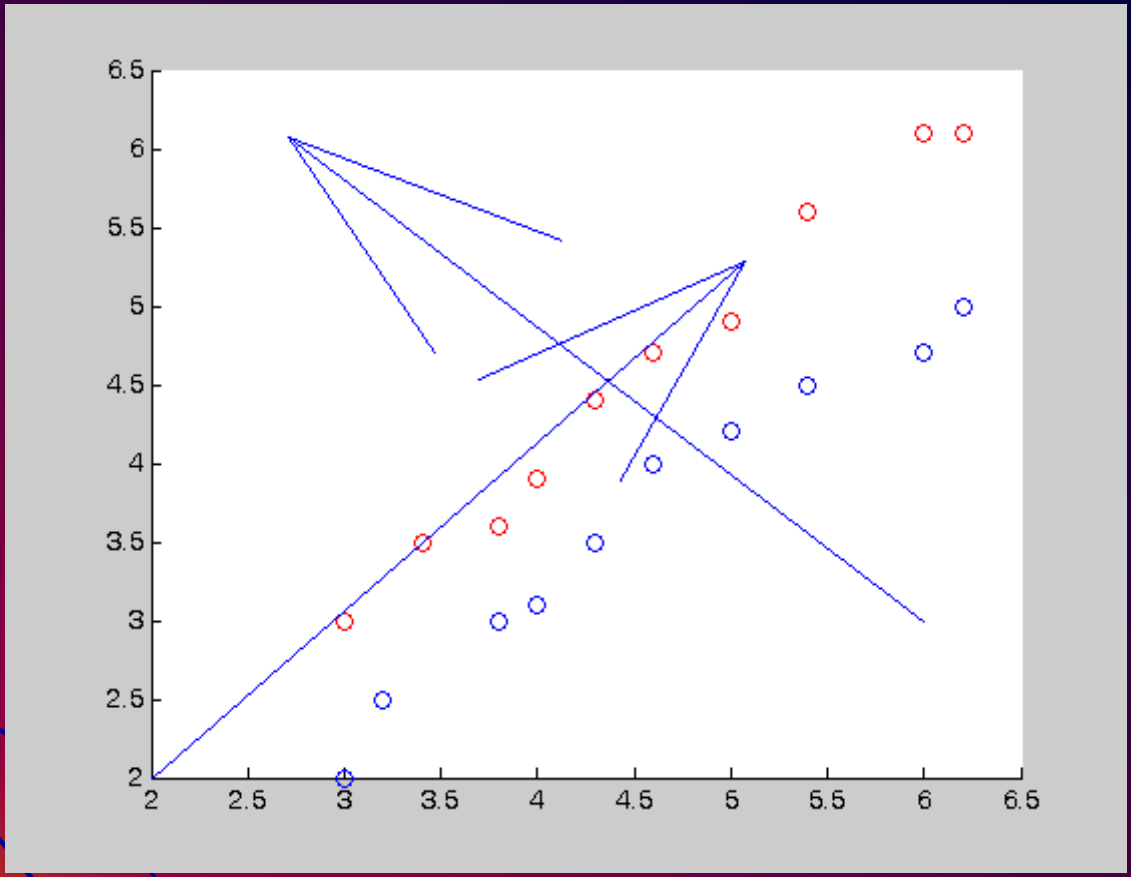
Demonstration of KL Transform



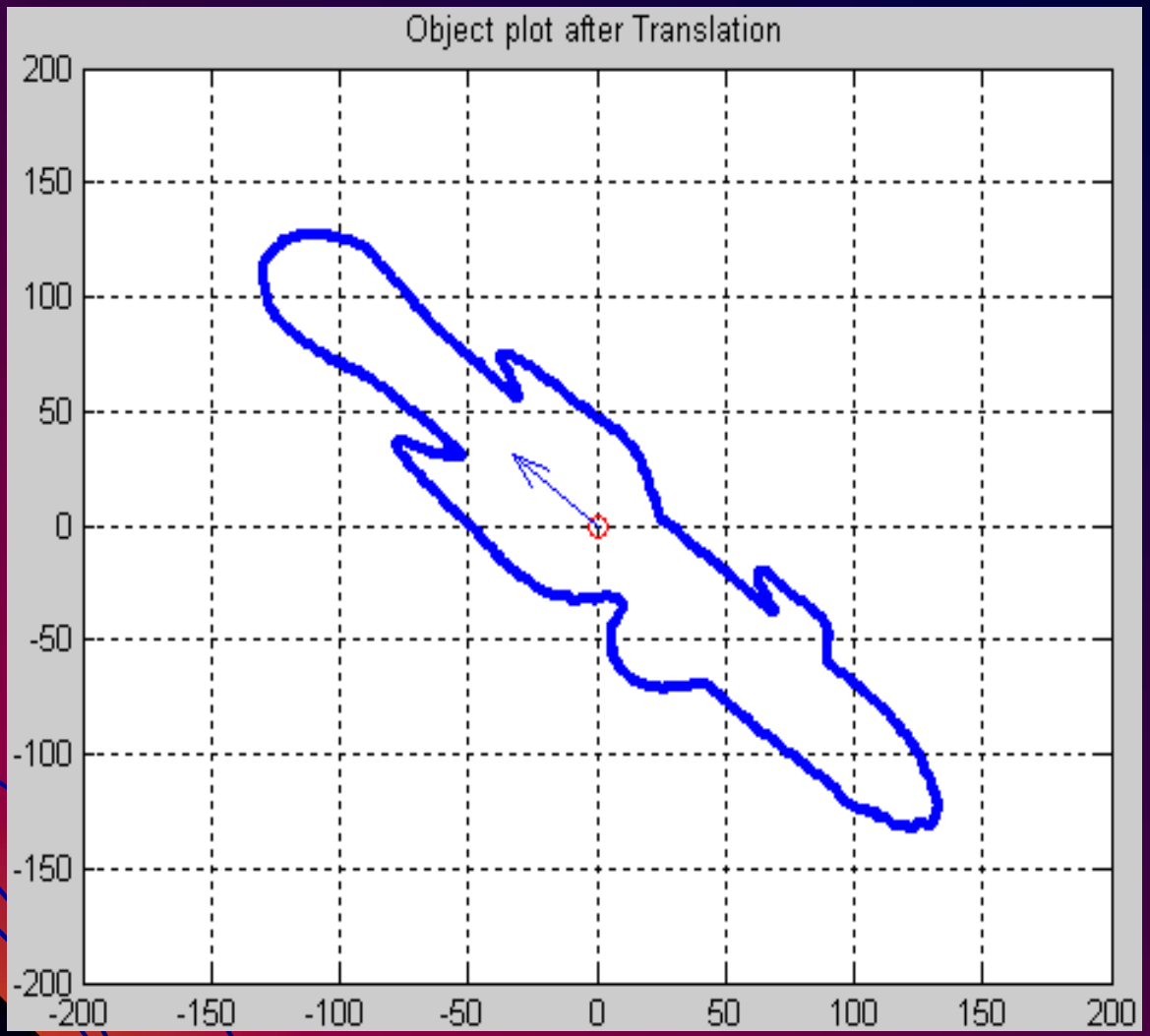
First eigen vector

Second eigen vector

Another One



Another Example



Source: SQUID Homepage

Principal components analysis (PCA) is a technique used to reduce multi-dimensional **data sets** to lower dimensions for analysis.

The applications include **exploratory data analysis** and generating predictive models. PCA involves the computation of the **eigenvalue decomposition** or **Singular value decomposition** of a data set, usually after mean centering the data for each attribute.

PCA is mathematically defined as an **orthogonal** **linear transformation**, that transforms the data to a new **coordinate system** such that the **greatest variance** by any projection of the data comes to lie on the **first coordinate** (called the first principal component), the second greatest variance on the second coordinate, and so on.

PCA can be used for **dimensionality reduction** in a data set by retaining those characteristics of the data set that contribute most to its **variance**, by keeping lower-order principal components and ignoring higher-order ones. Such low-order components often contain the "most important" aspects of the data. But this is not necessarily the case, depending on the application.

For a data matrix, X^T, with zero empirical mean (the empirical mean of the distribution has been subtracted from the data set), where each *column* is made up of results for a different subject, and each *row* the results from a different probe. This will mean that the PCA for our data matrix X will be given by:

$$Y = W^T X = \Sigma V^T,$$

where $W\Sigma V^T$ is the singular value decomposition (SVD) of X.

Goal of PCA:

Find some orthonormal matrix W^T, where $Y = W^T X$; such that

$COV(Y) \equiv (1/(n-1))YY^T$ is diagonalized.

The rows of W are the principal components of X, which are also the eigenvectors of COV(X).

Unlike other linear transforms (DCT, DFT, DWT etc.), PCA does not have a fixed set of basis vectors. Its basis vectors depend on the data set.

SVD – the theorem

Suppose M is an m-by-n matrix whose entries come from the field K, which is either the field of real numbers or the field of complex numbers. Then there exists a factorization of the form

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

where U is an m-by-m unitary matrix over K, the matrix Σ is m-by-n with nonnegative numbers on the diagonal and zeros off the diagonal, and V* denotes the conjugate transpose of V, an n-by-n unitary matrix over K. Such a factorization is called a *singular-value decomposition of M.*

The matrix V thus contains a set of orthonormal "input" or "analysing" basis vector directions for M.

The matrix U contains a set of orthonormal "output" basis vector directions for M. The matrix Σ contains the singular values, which can be thought of as scalar "gain controls" by which each corresponding input is multiplied to give a corresponding output.

A common convention is to order the values $\Sigma_{i,i}$ in non-increasing fashion. In this case, the diagonal matrix Σ is uniquely determined by M (though the matrices U and V are not).

For p = min(m,n) — U is m-by-p, Σ is p-by-p, and V is n-by-p.

The Karhunen-Loève transform is therefore equivalent to finding the singular value decomposition of the data matrix *X*, and then obtaining the reduced-space data matrix Y by projecting X down into the reduced space defined by only the first *L* singular vectors, W_L:

$$X = W\Sigma V^T; \quad Y = W_L^T X = \Sigma_L V_L^T$$

The matrix W of singular vectors of X is equivalently the matrix W of eigenvectors of the matrix of observed covariances $C = X X^T$ *(find out?) =:*

$$COV(X) = XX^T = W\Sigma\Sigma^T W^T = WDW^T$$

The eigenvectors with the largest eigenvalues correspond to the dimensions that have the strongest correlation in the data set. PCA is equivalent to empirical orthogonal functions (EOF).

PCA is a popular technique in pattern recognition. But it is not optimized for class separability. An alternative is the linear discriminant analysis, which does take this into account. PCA optimally minimizes reconstruction error under the L_2 norm.

PCA by COVARIANCE Method

We need to find a dxd orthonormal transformation matrix W^T, such that:

$$Y = W^T X$$

with the constraint that:
Cov(Y) is a diagonal matrix, and $W^{-1} = W^T$.

$$COV(Y) = E[YY^T] = E[(W^T X)(W^T X)^T]$$
$$= E[(W^T X)(X^T W)] = W^T E[XX^T]W$$
$$= W^T COV(X)W = W^T (WDW^T)W = D$$

$$WCOV(Y) = WW^T COV(X)W = COV(X)W$$

Can you derive from the above, that:

$$[\lambda_1 W_1, \lambda_2 W_2, ..., \lambda_d W_d] =$$
$$[COV(X)W_1, COV(X)W_2, ..., COV(X)W_d]$$

$$\mathbf{X} = \begin{bmatrix} X_1 \\ \vdots \\ X_n \end{bmatrix}$$

are random variables, each with finite variance, then the covariance matrix Σ is the matrix whose (*i*, *j*) entry is the covariance

$$\Sigma_{ij} = \operatorname{cov}(X_i, X_j) = \operatorname{E}\left[(X_i - \mu_i)(X_j - \mu_j)\right]$$

where

$$\mu_i = \operatorname{E}(X_i)$$

is the expected value of the *i*th entry in the vector **X**.[citation needed] In other words, we have

$$\Sigma = \begin{bmatrix} \operatorname{E}[(X_1 - \mu_1)(X_1 - \mu_1)] & \operatorname{E}[(X_1 - \mu_1)(X_2 - \mu_2)] & \cdots & \operatorname{E}[(X_1 - \mu_1)(X_n - \mu_n)] \\ \operatorname{E}[(X_2 - \mu_2)(X_1 - \mu_1)] & \operatorname{E}[(X_2 - \mu_2)(X_2 - \mu_2)] & \cdots & \operatorname{E}[(X_2 - \mu_2)(X_n - \mu_n)] \\ \vdots & \vdots & \ddots & \vdots \\ \operatorname{E}[(X_n - \mu_n)(X_1 - \mu_1)] & \operatorname{E}[(X_n - \mu_n)(X_2 - \mu_2)] & \cdots & \operatorname{E}[(X_n - \mu_n)(X_n - \mu_n)] \end{bmatrix}.$$

The inverse of this matrix, $\sum^{-1}$ is the **inverse covariance matrix**, also known as the **concentration matrix** or **precision matrix**;[1] see precision (statistics). The elements of the precision matrix have an interpretation in terms of partial correlations and partial variances.[citation needed]

Generalization of the variance

[e

The definition above is equivalent to the matrix equality

$$\Sigma = \operatorname{E}\left[(\mathbf{X} - \operatorname{E}[\mathbf{X}])\,(\mathbf{X} - \operatorname{E}[\mathbf{X}])^{\mathrm{T}}\right]$$

This form can be seen as a generalization of the scalar-valued variance to higher dimensions. Recall that for a scalar-valued random variable *X*

$$\sigma^2 = \operatorname{var}(X) = \operatorname{E}[(X - \operatorname{E}(X))^2] = \operatorname{E}[(X - \operatorname{E}(X)) \cdot (X - \operatorname{E}(X))].$$

Indeed, the entries on the diagonal of the covariance matrix $\sum$ are the variances of each element of the vector $\mathbf{X}$.

Conflicting nomenclatures and notations

[e

Nomenclatures differ. Some statisticians, following the probabilist William Feller, call this matrix the **variance** of the random vector X, because it is the natural generalization to higher dimensions of the 1-dimensional variance. Others call it the **covariance matrix**, because it is the matrix of covariances between the scalar components of the vector X. Thus

$$\operatorname{var}(\mathbf{X}) = \operatorname{cov}(\mathbf{X}) = \operatorname{E}\left[(\mathbf{X} - \operatorname{E}[\mathbf{X}])(\mathbf{X} - \operatorname{E}[\mathbf{X}])^{\mathrm{T}}\right].$$

However, the notation for the cross-covariance *between* two vectors is standard:

$$\operatorname{cov}(\mathbf{X}, \mathbf{Y}) = \operatorname{E}\left[(\mathbf{X} - \operatorname{E}[\mathbf{X}])(\mathbf{Y} - \operatorname{E}[\mathbf{Y}])^{\mathrm{T}}\right].$$

and *y* do not fully describe the distribution
A 2×2 covariance matrix is needed; the
directions of the arrows correspond to the
eigenvectors of this covariance matrix an
their lengths to the square roots of the
eigenvalues.

The var notation is found in William Feller's two-volume book *An Introduction to Probability Theory and Its Applications*,[2] but both forms are quite standard and there is no ambiguity between them.

The matrix Σ is also often called the variance-covariance matrix since the diagonal terms are in fact variances.

Properties [edit]

For $\Sigma = \mathrm{E}\left[(\mathbf{X} - \mathrm{E}[\mathbf{X}])\,(\mathbf{X} - \mathrm{E}[\mathbf{X}])^{\mathrm{T}}\right]$ and $\boldsymbol{\mu} = \mathrm{E}(\mathbf{X})$, where $\mathbf{X}$ is a random p-dimensional variable and $\mathbf{Y}$ a random q-dimensional variable, the following basic properties apply:[citation needed]

1. $\Sigma = \mathrm{E}(\mathbf{X}\mathbf{X}^{\mathrm{T}}) - \boldsymbol{\mu}\boldsymbol{\mu}^{\mathrm{T}}$
2. Σ is positive-semidefinite and symmetric.
3. $\operatorname{cov}(\mathbf{A}\mathbf{X} + \mathbf{a}) = \mathbf{A}\,\operatorname{cov}(\mathbf{X})\,\mathbf{A}^{\mathrm{T}}$
4. $\operatorname{cov}(\mathbf{X}, \mathbf{Y}) = \operatorname{cov}(\mathbf{Y}, \mathbf{X})^{\mathrm{T}}$
5. $\operatorname{cov}(\mathbf{X}_1 + \mathbf{X}_2, \mathbf{Y}) = \operatorname{cov}(\mathbf{X}_1, \mathbf{Y}) + \operatorname{cov}(\mathbf{X}_2, \mathbf{Y})$
6. If $p = q$, then $\operatorname{var}(\mathbf{X} + \mathbf{Y}) = \operatorname{var}(\mathbf{X}) + \operatorname{cov}(\mathbf{X}, \mathbf{Y}) + \operatorname{cov}(\mathbf{Y}, \mathbf{X}) + \operatorname{var}(\mathbf{Y})$
7. $\operatorname{cov}(\mathbf{A}\mathbf{X} + \mathbf{a}, \mathbf{B}^{\mathrm{T}}\mathbf{Y} + \mathbf{b}) = \mathbf{A}\,\operatorname{cov}(\mathbf{X}, \mathbf{Y})\,\mathbf{B}$
8. If $\mathbf{X}$ and $\mathbf{Y}$ are independent or uncorrelate, then $\operatorname{cov}(\mathbf{X}, \mathbf{Y}) = \mathbf{0}$

where $\mathbf{X}, \mathbf{X}_1$ and $\mathbf{X}_2$ are random p×1 vectors, $\mathbf{Y}$ is a random q×1 vector, $\mathbf{a}$ is a q×1 vector, $\mathbf{b}$ is a p×1 vector, and $\mathbf{A}$ and $\mathbf{B}$ are q×p matrices.

This covariance matrix is a useful tool in many different areas. From it a transformation matrix can be derived, called a whitening transformation, that allows one to completely decorrelate the data[citation needed] or, from a different point of view, to find an optimal basis for representing the data in a compact way[citation needed] (see Rayleigh quotient for a formal proof and additional properties of covariance matrices). This is called principal components analysis (PCA) and the Karhunen-Loève transform (KL-transform).

As a linear operator [edit]

Applied to one vector, the covariance matrix maps a linear combination, $\mathbf{c}$, of the random variables, $\mathbf{X}$, onto a vector of covariances with those variables: $\mathbf{c}^{\mathrm{T}}\Sigma = \operatorname{cov}(\mathbf{c}^{\mathrm{T}}\mathbf{X}, \mathbf{X})$. Treated as a bilinear form, it yields the covariance between the two linear combinations: $\mathbf{d}^{\mathrm{T}}\Sigma\mathbf{c} = \operatorname{cov}(\mathbf{d}^{\mathrm{T}}\mathbf{X}, \mathbf{c}^{\mathrm{T}}\mathbf{X})$. The variance of a linear combination is then $\mathbf{c}^{\mathrm{T}}\Sigma\mathbf{c}$, its covariance with itself.

Similarly, the (pseudo-)inverse covariance matrix provides an inner product, $\langle c - \mu | \Sigma^{+} | c - \mu \rangle$ which induces the Mahalanobis distance, a measure of the "unlikelihood" of c.[citation needed]

Which matrices are covariance matrices? [edit]

From the identity just above, let $\mathbf{b}$ be a $(p \times 1)$ real-valued vector, then

$$\operatorname{var}(\mathbf{b}^{\mathrm{T}}\mathbf{X}) = \mathbf{b}^{\mathrm{T}} \operatorname{var}(\mathbf{X})\mathbf{b},$$

which must always be nonnegative since it is the variance of a real-valued random variable. and the symmetry of the covariance matrix's definition it follows that only a positive-semidefinite matrix can be a covariance matrix.[citation needed] The answer to the converse question, whether *every* symmetric positive semi-definite matrix is a covariance matrix, is "yes." To see this, suppose $\mathbf{M}$ is a p×p positive-semidefinite matrix. From the finite-dimensional case of the spectral theorem, it follows that $\mathbf{M}$ has a nonnegative symmetric square root, that can be denoted by $\mathbf{M}^{1/2}$. Let $\mathbf{X}$ be any p×1 column vector-valued random variable whose covariance matrix is the p×p identity matrix. Then

$$\operatorname{var}(\mathbf{M}^{1/2}\mathbf{X}) = \mathbf{M}^{1/2}(\operatorname{var}(\mathbf{X}))\mathbf{M}^{1/2} = \mathbf{M}.$$

Example of PCA

Samples: $x_1 = \begin{bmatrix} -1 \\ 1 \\ 2 \end{bmatrix}; x_2 = \begin{bmatrix} -2 \\ 3 \\ 1 \end{bmatrix}; x_3 = \begin{bmatrix} 4 \\ 0 \\ 3 \end{bmatrix};$ $\quad X = \begin{bmatrix} -1 & -2 & 4 \\ 1 & 3 & 0 \\ 2 & 1 & 3 \end{bmatrix}$

3-D problem, with N = 3.

Each column is an observation (sample) and each row a variable (dimension),

Mean of the samples: $\mu_x = \begin{bmatrix} 1/3 \\ 4/3 \\ 2 \end{bmatrix};$ $\quad \tilde{x}_1 = \begin{bmatrix} -4/3 \\ -1/3 \\ 0 \end{bmatrix}; \tilde{x}_2 = \begin{bmatrix} -7/3 \\ 5/3 \\ -1 \end{bmatrix}; \tilde{x}_3 = \begin{bmatrix} 11/3 \\ -4/3 \\ 1 \end{bmatrix};$

<u>Method – 1</u> (easiest)

$\tilde{X} = \begin{bmatrix} -4/3 & -7/3 & 11/3 \\ -1/3 & 5/3 & -4/3 \\ 0 & -1 & 1 \end{bmatrix};$ $\quad$ COVAR = $(\tilde{X}\ \tilde{X}^T)/2 = (1/2)\begin{bmatrix} 62/3 & -25/3 & 6 \\ -25/3 & 14/3 & -3 \\ 6 & -3 & 2 \end{bmatrix}$

Method – 2 (PCA defn.)

$$S_T = \left(\frac{1}{N-1}\right)\sum_{k=1}^{N}(x_k - \mu)(x_k - \mu)^T$$

$$\tilde{x}_1 = \begin{bmatrix} -4/3 \\ -1/3 \\ 0 \end{bmatrix}; \tilde{x}_2 = \begin{bmatrix} -7/3 \\ 5/3 \\ -1 \end{bmatrix}; \tilde{x}_3 = \begin{bmatrix} 11/3 \\ -4/3 \\ 1 \end{bmatrix};$$

C1 =

1.7778	0.4444	0
0.4444	0.1111	0
0	0	0

C2 =

5.4444	-3.8889	2.3333
-3.8889	2.7778	-1.6667
2.3333	-1.6667	1.0000

C3 =

13.4444	-4.8889	3.6667
-4.8889	1.7778	-1.3333
3.6667	-1.3333	1.0000

SigmaC =

20.6667	-8.3333	6.0000
-8.3333	4.6667	-3.0000
6.0000	-3.0000	2.0000

COVAR =
SigmaC/2 =

10.3333	-4.1667	3.0000
-4.1667	2.3333	-1.5000
3.0000	-1.5000	1.0000

Next do SVD, to get vectors.

For a face image with N samples and dimension d (=w*h, very large), we have:

The array X or Xavg of size d*N (N vertical samples stacked horizontally)

Thus XX^T will be of d*d, which will be very large. To perform eigen-analysis on such large dimension is time consuming and may be erroneous.

Thus often X^TX of dimension N*N is considered for eigen-analysis. Will it result in the same, after SVD? Lets check:

$$S = \tilde{X}\,\tilde{X}^{T} = (1/2)\begin{bmatrix} 62/3 & -25/3 & 6 \\ -25/3 & 14/3 & -3 \\ 6 & -3 & 2 \end{bmatrix} = \begin{matrix} 10.3333 & -4.1667 & 3.0000 \\ -4.1667 & 2.3333 & -1.5000 \\ 3.0000 & -1.5000 & 1.0000 \end{matrix}$$

$$S^{m} = \tilde{X}^{T}\,\tilde{X} = \begin{matrix} 0.9444 & 1.2778 & -2.2222 \\ 1.2778 & 4.6111 & -5.8889 \\ -2.2222 & -5.8889 & 8.1111 \end{matrix}$$

Lets do SVD of both:

$$S = \tilde{X}\,\tilde{X}^{T} =$$

10.3333	-4.1667	3.0000
-4.1667	2.3333	-1.5000
3.0000	-1.5000	1.0000

U =

-0.8846	-0.4554	-0.1010
0.3818	-0.8313	0.4041
-0.2680	0.3189	0.9091

S =

13.0404	0	0
0	0.6263	0
0	0	0.0000

V =

-0.8846	-0.4554	0.1010
0.3818	-0.8313	-0.4041
-0.2680	0.3189	-0.9091

$$S^{m} = \tilde{X}^{T}\,\tilde{X} =$$

0.9444	1.2778	-2.2222
1.2778	4.6111	-5.8889
-2.2222	-5.8889	8.1111

U =

-0.2060	0.7901	0.5774
-0.5812	-0.5735	0.5774
0.7872	-0.2166	0.5774

S =

13.0404	0	0
0	0.6263	0
0	0	0.0000

V =

-0.2060	0.7901	0.5774
-0.5812	-0.5735	0.5774
0.7872	-0.2166	0.5774

Samples:

Example, where d <> N:

$$x_1 = \begin{bmatrix} -3 \\ -3 \end{bmatrix}; x_2 = \begin{bmatrix} -2 \\ -2 \end{bmatrix}; x_3 = \begin{bmatrix} -1 \\ -1 \end{bmatrix}; x_4 = \begin{bmatrix} 4 \\ 4 \end{bmatrix}; x_5 = \begin{bmatrix} 5 \\ 5 \end{bmatrix}; x_6 = \begin{bmatrix} 6 \\ 7 \end{bmatrix};$$

2-D problem (d=2), with N = 6.

Each column is an observation (sample) and each row a variable (dimension),

X =

-3	**-2**	**-1**	**4**	**5**	**6**
-3	**-2**	**-1**	**4**	**5**	**7**

Mean of the samples:

$$\mu_x = \begin{bmatrix} 3 \ / \ 2 \\ 5 \ / \ 3 \end{bmatrix};$$

XM=

-4.5000	-3.5000	-2.5000	2.5000	3.5000	4.5000
-4.6667	-3.6667	-2.6667	2.3333	3.3333	5.3333

$XM^T * XM =$

42.0278	32.8611	23.6944	-22.1389	-31.3056	-45.1389
32.8611	25.6944	18.5278	-17.3056	-24.4722	-35.3056
23.6944	18.5278	13.3611	-12.4722	-17.6389	-25.4722
-22.1389	-17.3056	-12.4722	11.6944	16.5278	23.6944
-31.3056	-24.4722	-17.6389	16.5278	23.3611	33.5278
-45.1389	-35.3056	-25.4722	23.6944	33.5278	48.6944

$COVAR(X) = XM * XM^T$

=

77.5000	82.0000
82.0000	87.3333

$COVAR(X) = XM * XM^T$

$$= \begin{matrix} 77.5000 & 82.0000 \\ 82.0000 & 87.3333 \end{matrix}$$

U =

$$\begin{matrix} -0.6856 & -0.7280 \\ -0.7280 & 0.6856 \end{matrix}$$

S =

$$\begin{matrix} 164.5639 & 0 \\ 0 & 0.2694 \end{matrix}$$

V =

$$\begin{matrix} -0.6856 & -0.7280 \\ -0.7280 & 0.6856 \end{matrix}$$

$XM^T * XM =$

$$\begin{matrix} 42.0278 & 32.8611 & 23.6944 & -22.1389 & -31.3056 & -45.1389 \\ 32.8611 & 25.6944 & 18.5278 & -17.3056 & -24.4722 & -35.3056 \\ 23.6944 & 18.5278 & 13.3611 & -12.4722 & -17.6389 & -25.4722 \\ -22.1389 & -17.3056 & -12.4722 & 11.6944 & 16.5278 & 23.6944 \\ -31.3056 & -24.4722 & -17.6389 & 16.5278 & 23.3611 & 33.5278 \\ -45.1389 & -35.3056 & -25.4722 & 23.6944 & 33.5278 & 48.6944 \end{matrix}$$

U =

$$\begin{matrix} -0.5053 & -0.1469 & -0.7547 & 0.3882 & 0.0214 & 0.0486 \\ -0.3951 & -0.0654 & 0.3632 & 0.0984 & -0.4091 & 0.7284 \\ -0.2849 & 0.0162 & -0.0433 & -0.3456 & -0.7396 & -0.5002 \\ 0.2660 & 0.4241 & -0.5083 & -0.5306 & -0.1150 & 0.4429 \\ 0.3762 & 0.5057 & -0.0258 & 0.6601 & -0.4043 & -0.0539 \\ 0.5432 & -0.7337 & -0.1938 & 0.0541 & -0.3293 & 0.1332 \end{matrix}$$

S =

$$\begin{matrix} 164.5639 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.2694 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.0 \end{matrix}$$

V = U ??

REFERENCES

- **Statistical pattern Recognition; S. Fukunaga; Academic Press, 2000.**

- **Bishop – PR**

- **Satish Kumar - ANN**