Principal component analysis

- **Principal component analysis (PCA)**
  - also referred to as **Karhunen–Loève transform (KLT)** in signal processing.
- PCA provides a solution to the problem

$$\underset{B}{\operatorname{argmin}} \sum_{i=1}^{N} \left\| x_i - proj(x_i, \Pi_{B;m}) \right\|^2,$$

- That is, find a lower-dimensional (here m-dimensional) subspace (hyperplanes) such that the sum of data-to-hyperplane distances is minimized.

PCA

- Given the data matrix $X \in R^{n \times N}$, the scatter matrix is defined as $S = XX^T \in R^{n \times n}$.
- Scatter matrix is a positive semi-definite matrix. It can be eigen-decomposed as

$$XX^T = PDP^T$$

where $D \in R^{n \times n}$ is a diagonal matrix consisting of the eigenvalues $\lambda_1, \lambda_2, \dots, \lambda_n$ (w.l.o.g., from large to small) of the scatter matrix S.

PCA and eigen-decomposition

$$XX^T = PDP^T$$

- P's columns (denoted as $p_1, p_2, \ldots, p_n$) are the eigenvectors of S, and $P \in R^{n \times n}$ is an orthonormal matrix (i.e., $PP^T = I$). That is, the column vectors $p_1, p_2, \ldots, p_n$ ($p_k \in R^n$) are perpendicular to each other.
- All eigenvalues $\lambda_1, \lambda_2, \ldots, \lambda_n$ are nonnegative because XX^T is positive semi-definite.

- PCA employs the above eigen-decomposition structure to find the distance-minimized low-dimensional hyperplane (subspace) of the data.

PCA for dimension reduction

- Our problem is to find the m-dimensional subspace $\Pi_{B;m}$ such that the sum of data-to-hyperplane distances $\sum_{i=1}^{N}\left\|x_i - proj(x_i, \Pi_{B;m})\right\|^2$ is minimized. This is referred to as a **dimension-reduction problem**.

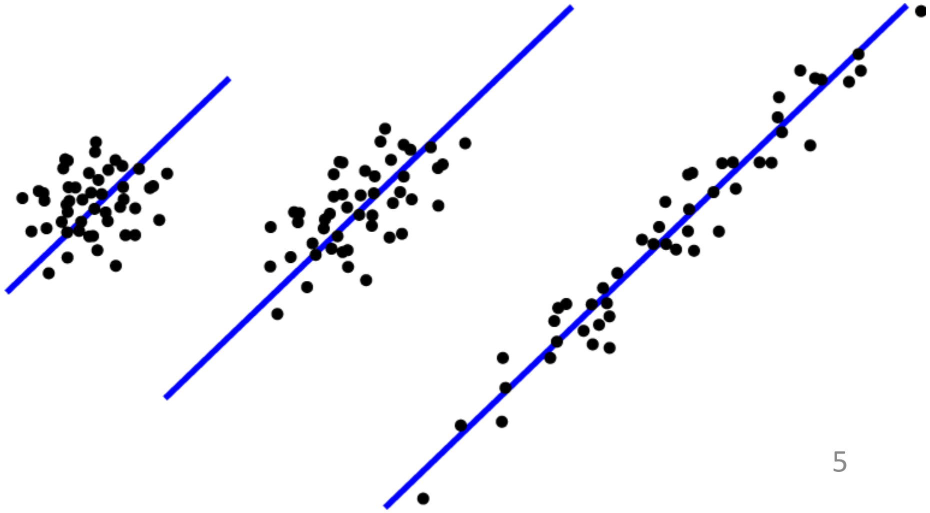
- **Property**: The solution of the above problem is the subspace spanned by the eigenvectors corresponding to the largest m eigenvalues of XX^T.

- That is, the solution is the space spanned by the eigenvectors $\{p_1, p_2, \ldots, p_m\}$, where p_k is the k-th column of P.

Example, $m = 1$

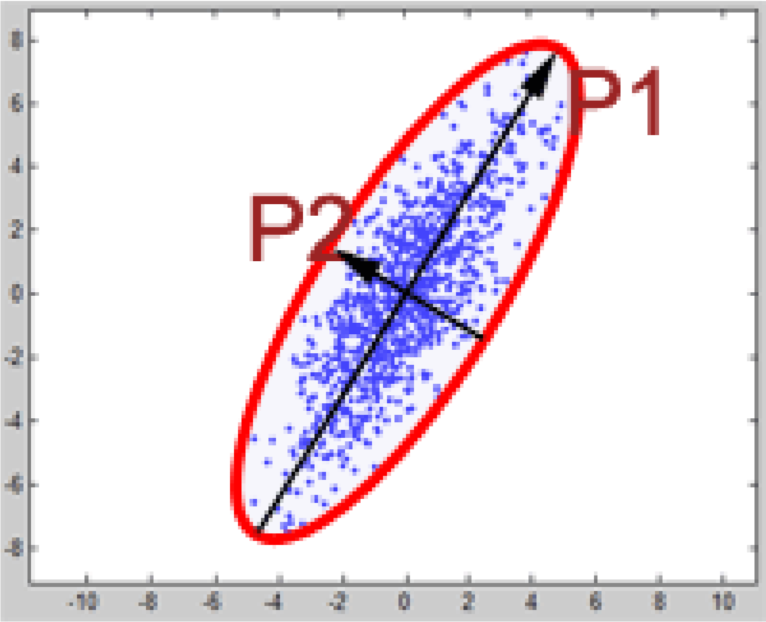
- When $m = 1$, the problem is equivalent to finding a 1D line (passing through the origin) such that the sum of distances from the data points to the line is minimized.
- The solution is the eigenvector corresponding to the largest eigenvalue of the scatter matrix. This axis is usually called the **first principal axis** or **main axis.**

Illustration of $n = 2$ and $m = 1$.



Example, $m = 2$

- When $m = 2$, we then find the **first** and the **second principal axes** of the dataset.



Partial sum of eigenvalues

- Note that in PCA, because $XX^T = PDP^T$, we have $$tr(XX^T) = tr(PDP^T) = tr(DP^TP) = tr(D) = \sum_{i=1}^{n} \lambda_i.$$
  - In the above, we have used a property of the trace operation, $tr(AB) = tr(BA)$.
- Besides, $tr(XX^T) = tr(X^TX) = \sum_{i=1}^{N} \|x_i\|^2$
- Hence, $\sum_{i=1}^{N} \|x_i\|^2 = \sum_{i=1}^{n} \lambda_i$. This is interpreted as the property that the total energy of the training data ($\sum_{i=1}^{N} \|x_i\|^2$) is equivalent to the sum of eigenvalues ($\sum_{i=1}^{n} \lambda_i$).

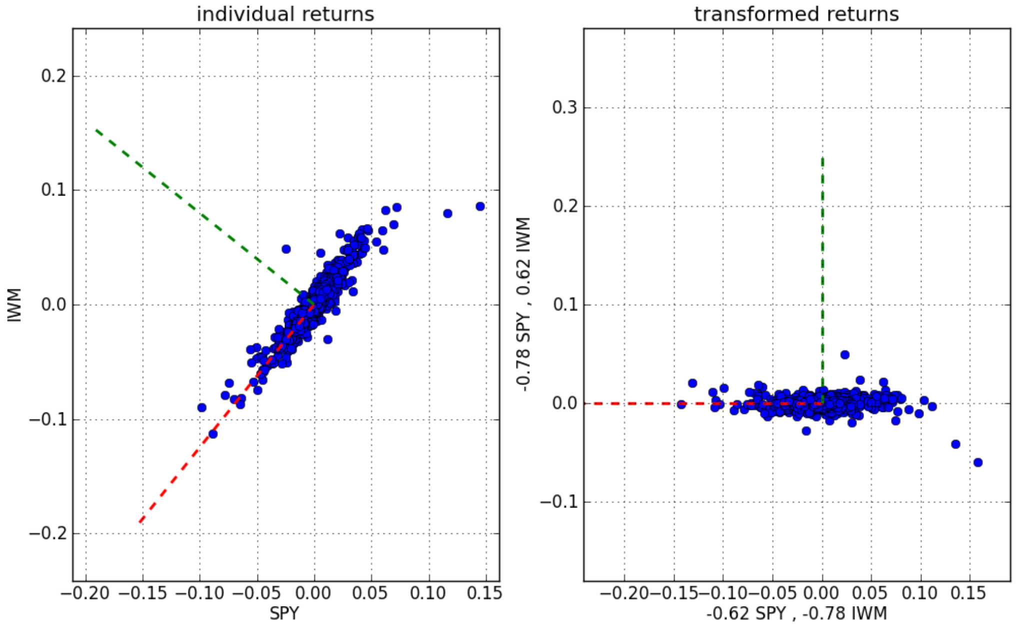
Partial sum of eigenvalues (cont.)

$$\sum_{i=1}^{N} \|x_i\|^2 = \sum_{i=1}^{n} \lambda_i$$

- Also note that the eigenvalues have been arranged in an descending order, $\lambda_1 \geq \lambda_2 \geq \ \dots \ \geq \lambda_n$.
- Hence, in dimension reduction from PCA, we usually compute the partial sum of the eigenvalues $\sum_{j=1}^{m} \lambda_j$ to determine the reduced dimension m.
- Eg., if we want to keep 95% energy of the data, we can find a minimal value of m such that $\frac{\sum_{j=1}^{m} \lambda_j}{\sum_{i=1}^{n} \lambda_i} >$ 95% for the determination of m.

Applications of PCA

- **Data compression**: because PCA reduces the data (signal) dimension from n to m, it can be used for data compression.
- PCA can reduce the redundancy of the data and results in a more concise and efficient representation.



Applications of PCA (cont.)

- **Feature extraction**: In pattern recognition, PCA is widely used as a feature extractor because it can reduce the redundancy of the data (signal) source and extract the key part of the data.
  - Either the coefficients or the reconstructed signals can be employed as the features.

Singular value decomposition

- In the above, we compute the eigenvalues/eigenvectors of the scatter matrix XX^T for PCA.
- PCA can also be computed via the singular value decomposition (SVD) of the data matrix X directly:

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

- Since $XX^T = U\Sigma V^T V\Sigma^T U^T = U\Sigma\Sigma^T U^T$, it yields that $P = U$ and $D = \Sigma\Sigma^T$ via the SVD computation.

Incremental computation of PCA

- The scatter matrix is an $n \times n$ matrix, where n is the dimension of data samples.
- A useful property is that the scatter matrix can be computed incrementally or online to **save memory**. Because

$$S_N = XX^T = \sum_{i=1}^{N} x_i x_i^T,$$

when S_{N-1} is available, we can obtain S_N simply by

$$S_N = S_{N-1} + x_N x_N^T.$$

Incremental computation of PCA (Cont.)

- In the incremental computation, we do not have to remember all the dataset X.
- Assume S_{N-1}, the scatter matrix of $N-1$ data points is computed and stored; S_N can be obtained incrementally when x_N, the N-th data sample, is available.
- Only S_N is stored, x_N can be dropped out without affecting the PCA computation.

Centered vs. non-centered PCA

- In the above, the PCA is referred to as the **non-centered PCA**, where the axes learned all pass the origin.
- Sometimes we will perform center-normalization in advance:
  - we compute the center of the dataset, $\bar{x} = \frac{1}{N}\sum_{i=1}^{N} x_i$, and shift the origin to the data center by $x_i \leftarrow x_i - \bar{x}$. Then, we perform PCA for the center-normalized data. This is called **centered PCA**. Centered PCA can often approximate the data more accurately.

PCA and matrix decomposition

- In the above, we have found the bases (that spans the hyperplane $\Pi_{B;m}$) for the problem $\underset{B}{\operatorname{argmin}} \sum_{i=1}^{N} \left\| x_i - proj(x_i, \Pi_{B;m}) \right\|^2$ via PCA.
- This is equivalent to finding the bases for the matrix factorization problem,

$$\underset{B,C}{\operatorname{argmin}} \|X - BC\|_F^2 = \sum_{i=1}^{N} \|x_i - Bc_i\|^2, B \in R^{n \times m}$$

- Hence, $\hat{B} = P_{1:m} = [p_1, p_2, \dots, p_m]$.

PCA and matrix decomposition (cont.)

- Besides the bases, how to find the optimal coefficients C for matrix decomposition?
- As the PCA bases (i.e., eigenvectors of the scatter matrix) are orthonormal, the coefficients can be found by inner products,

$$\hat{c}_i = \hat{B}^T x_i = p_{1:m}^T x_i, i = 1 \dots N.$$

Infinite solutions of the matrix factorization

- As we know, the bases found by PCA spans an m-dimensional subspace $\Pi_{B;m}$ that minimizes $\sum_{i=1}^{N}\left\|x_i - proj(x_i, \Pi_{B;m})\right\|^2$.
- However, there are infinite sets of bases that can span the same subspace, $\Pi_{B;m}$.
- This can be reflected in the matrix decomposition formulation $\underset{B,C}{\operatorname{argmin}}\|X - BC\|_F^2$.
- If $\hat{B}$ and $\hat{C}$ are the solutions of this formulation, then $\hat{B}Q$ and $Q^T\hat{C}$ are also the solutions for any orthonormal matrix $Q \in R^{m\times m}$ satisfying that $QQ^T = I$.
- This shows again that matrix factorization problem has infinite solutions.

Infinite solutions of the matrix decomposition (cont.)

$$\underset{B,C}{\operatorname{argmin}} \|X - BC\|_F^2 \text{ with } m < n$$

- Hence, the bases of the above **matrix factorization problem** can be obtained by PCA as $\hat{B} = P_{1:m}Q$, where $Q \in R^{m \times m}$ is an arbitrary matrix satisfying $QQ^T = I$. (infinite solutions)
- The coefficients can then be found as $\hat{C} = \hat{B}^T X$.

PCA from another point of view

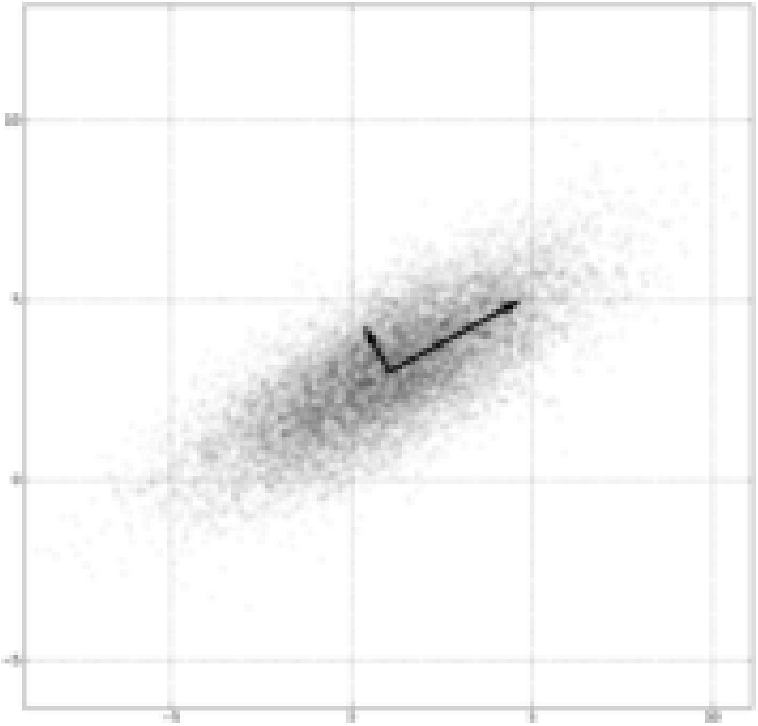
- In the above, we have stated that PCA finds the subspace $\Pi_{B;m}$ such that the sum of data-to-subspace distances is minimized as $\min \sum_{i=1}^{N} \left\| x_i - proj(x_i, \Pi_{B;m}) \right\|^2$.
- Another explanation of PCA is that it maximizes the variance of the projected data (here, we assume the dataset is zero mean) as follows,

$$\underset{\Pi_{B;m}}{\operatorname{argmax}} \sum_{i=1}^{N} \left\| proj(x_i, \Pi_{B;m}) \right\|^2$$

- The two formulations obtain the same solution, i.e., **PCA**.

Example

- PCA finds the axis of the largest variance (the first axis), and then the axis of the second largest variance (the second axis), and so on.



Derivation of PCA

on the viewpoint of maximizing the variance

Case of $\boldsymbol{m = 1}$ (i.e., find only a single axis)

- To project the data sample x_i onto the unit-length $\hat{x}$, we compute the inner product $\hat{x}^T x_i$. The variance (centered at zero) of the total data is

$$\sum_{i=1}^{N} (\hat{x}^T x_i)^2 = \sum\nolimits_{i=1}^{N} \hat{x}^T x_i x_i^T \hat{x} = \hat{x}^T \left(\sum\nolimits_{i=1}^{N} x_i x_i^T \right) \hat{x} = \hat{x}^T X X^T \hat{x}$$

- So, the optimization problem of maximizing the variance becomes

$$\underset{\hat{\boldsymbol{x}}}{\mathrm{arg}max}\; \hat{x}^T X X^T \hat{x}$$

subject to the constraint $\|\hat{x}\| = 1$.

- According to the **Rayleigh quotient principle** (you should have learned it in linear algebra), the optimal solution is the first eigenvector of the scatter matrix XX^T.

Derivation of PCA (cont.)

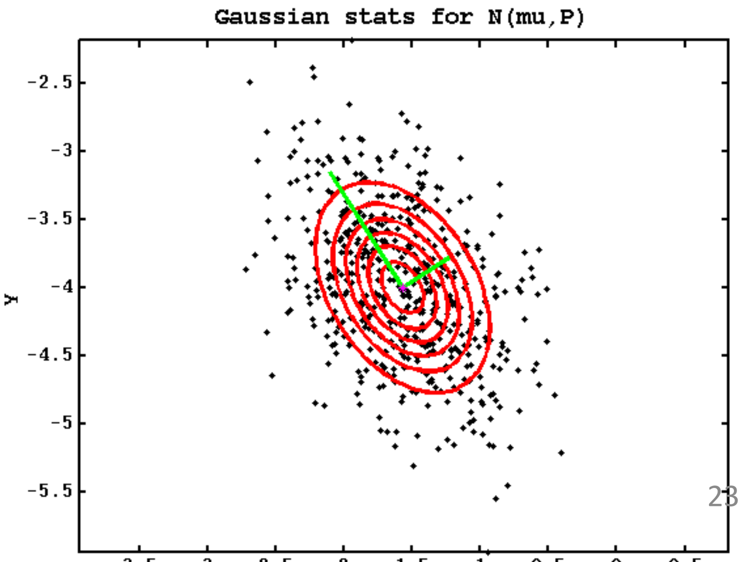
on the viewpoint of maximizing the variance

Case of $\boldsymbol{m > 1}$

- When $m > 1$, the above proof (via Rayleigh principle) can be easily extended to the case when we restrict the previous $m - 1$ axes to be the $m - 1$ largest eigenvectors, and increasingly find only the m-th axis.
- For the general case where no previous axes have been set, the proof is given in the Appendix.

Multivariate Gaussian distribution vs. PCA

- The isosurface (or level set) of a multivariate Gaussian distribution is hyper-ellipse shaped.
- The PCA axes of the hyper-ellipse-shaped point clouds (eg., sampled from multi-variate Gaussian distribution) are just the principal axes of the hyper ellipse.



Appendix

Proof of PCA

Principle Component Analysis (PCA)

- Given un-labeled training data: $x_i \in R^n, i = 1 \ldots N$
- Projecting onto the orthonormal bases $\{e_1, \ldots, e_m | e_j \in R^n\}$

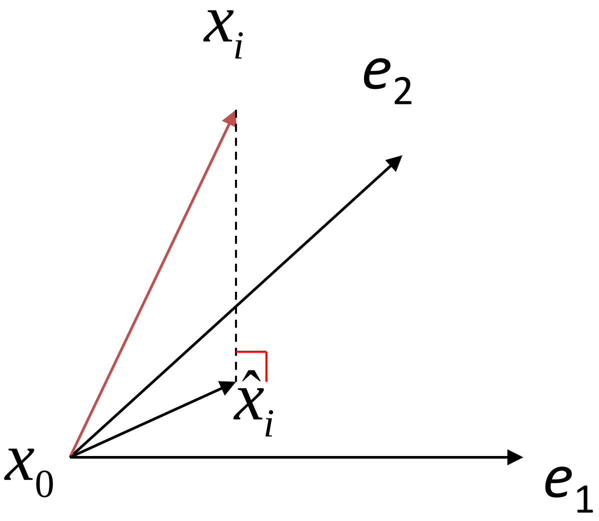
$$\hat{x}_i = x_0 + c_{i1}e_1 + c_{i2}e_2 + \ldots + c_{im}e_m,$$

$$e_j{}^T e_j = 1, e_j{}^T e_k = 0 \text{ for } j \neq k \text{ (orthonormal bases),}$$

where x_0 is the sample mean,

$$x_0 = \frac{1}{N}\sum_{i=1}^{N} x_i$$

$\hat{x}_i - x_i$ is minimized when orthogonal projection



Proof of PCA

- From orthogonal projection, the optimal coefficients should be

$$c_{ij} = (x_i - x_0)^T e_j$$

$$({e_j}^T e_j = 1, {e_j}^T e_k = 0 \text{ for } j \neq k)$$

Hence, the projected point is

$$\hat{x}_i = x_0 + e_1 (x_i - x_0)^T e_1 + \ldots + e_m (x_i - x_0)^T e_m$$

- The error is:

$$x_i - \hat{x}_i = (x_i - x_0) - [e_1 (x_i - x_0)^T e_1 + \ldots + e_m (x_i - x_0)^T e_m]$$

Proof of PCA

- Error:

$$x_i - \hat{x}_i$$
$$= (x_i - x_0) - [e_1(x_i - x_0)^T e_1 + \ldots + e_m(x_i - x_0)^T e_m]$$

- Goal: Finding the orthonormal bases $e_1, \ldots, e_m$ of the hyperplane to minimize the sum of squared error

$$E(e_1, \ldots, e_{ms}) = \sum_{i=1}^{N} ||x_i - \hat{x}_i||^2$$

$$\underset{e_1,\ldots,e_m}{\operatorname{argmin}} E(e_1, \ldots, e_m) = \sum_{i=1}^{N} (x_i - \hat{x}_i)^T (x_i - \hat{x}_i)$$

subject to $e_j{}^T e_j = 1$ & $e_j{}^T e_k = 0$ for $j \neq k$)

Proof of PCA

- As $x_i - \hat{x}_i = (x - x_0) - [e_1(x_i - x_0)^T e_1 + \ldots + e_m(x_i - x_0)^T e_m]$

we have $(x_i - \hat{x}_i)^T(x_i - \hat{x}_i) = Term1 + Term2 + Term3$,
where

$Term1 = ||x_i - x_0||^2$

$$\begin{aligned} Term2 = &-(x_i - x_0)^T[e_1(x_i - x_0)^T e_1 + \ldots + e_m(x_i - x_0)^T e_m] \\ &-[e_1^T(x_i - x_0)e_1^T + \ldots + e_m^T(x_i - x_0)e_m^T](x_i - x_0) \end{aligned}$$

$Term3 =$
$[e_1(x_i - x_0)^T e_1 + \ldots + e_1(x_i - x_0)^T e_1]^T[e_1(x_i - x_0)^T e_1 + \ldots + e_m(x_i - x_0)^T e_m]$

Proof of PCA

In the above, $Term$ 3 is

$$[e_1(x_i - x_0)^T e_1 + \ldots + e_1(x_i - x_0)^T e_1]^T [e_1(x_i - x_0)^T e_1 + \ldots + e_m(x_i - x_0)^T e_m]$$

$$= e_1^T(x_i - x_0)e_1^T e_1(x_i - x_0)^T e_1 + \ldots + e_m^T(x_i - x_0)e_m^T e_m(x_i - x_0)^T e_m + \sum_{j \neq k} e_j^T(x_i - x_0)e_j^T e_k(x_i - x_0)^T e_k$$

$$= e_1^T(x_i - x_0)\,(x_i - x_0)^T e_1 + \ldots + e_m^T(x_i - x_0)(x_i - x_0)^T e_m$$

(since $e_j{}^T e_j = 1,\ \ e_j{}^T e_k = 0$ for $j \neq k$)

$Term$ 2 is

$$-(x_i - x_0)^T[e_1(x_i - x_0)^T e_1 + \ldots + e_m(x_i - x_0)^T e_m] - [e_1^T(x_i - x_0)e_1^T + \ldots + e_m^T(x_i - x_0)e_m^T](x_i - x_0)$$

$$= -2[e_1^T(x_i - x_0)(x_i - x_0)^T e_1 + \cdots + e_m^T(x_i - x_0)(x_i - x_0)^T e_m]$$

$$= -2 \times Term3$$

(This is because both $(x_i - x_0)^T\ e_j(x_i - x_0)^T e_j$ and $e_j^T(x_i - x_0)e_j^T(x_i - x_0)$ are equal to $e_j^T(x_i - x_0)(x_i - x_0)^T e_j$.)

Proof of PCA

So, $Term1 \; + \; Term2 \; + \; Term3$

$= \; Term\,1 - 2Term\,3 + Term3$

$= Term1 - Term3$

$= ||x_i - x_0||^2$
$- [e_1^T(x_i - x_0)(x_i - x_0)^T e_1 + \cdots + e_m^T(x_i - x_0)(x_i - x_0)^T e_m]$

Proof of PCA

- Thus
$$E = \sum_{i=1}^{N} ||x_i - x_0||^2$$
$$- \sum_{i=1}^{N} [e_1^T (x_i - x_0)(x_i - x_0)^T e_1 + \cdots + e_m^T (x_i - x_0)(x_i - x_0)^T e_m]$$

- Since $\sum_{i=1}^{N} ||x_i - x_0||^2$ is a constant, minimizing E is equivalent to maximizing

$$V = \sum_{i=1}^{N} e_1^T (x_i - x_0)(x_i - x_0)^T e_1 + \cdots + e_m^T (x_i - x_0)(x_i - x_0)^T e_m \,,$$

subject to the constraint ${e_j}^T e_j = 1$ and ${e_j}^T e_k = 0$ for $j \neq k$.

Proof of PCA

- V can be written as

$$V = e_1^T [\sum_{i=1}^{N} (x_i - x_0)(x_i - x_0)^T]\, e_1 + \ldots + e_m^T [\sum_{i=1}^{N} (x_i - x_0)(x_i - x_0)^T]\, e_m$$

$$= e_1^T S e_1 + \ldots + e_m^T S e_m,$$

where S is the scatter matrix, $S = \sum_{i=1}^{N} (x_i - x_0)(x_i - x_0)^T$

- So, maximizing V is equivalent to maximizing the projected scatter,

$$e_1^T S e_1 + \ldots + e_m^T S e_m$$

subject to the constraint ${e_j}^T e_j = 1$ and ${e_j}^T e_k = 0$ for $j \neq k$.

Hence, minimizing the sum of squared distances to the hyperplane is equivalent to maximizing the variance within the hyperplane.

Proof of PCA

- For constrained optimization, we consider the Lagrange multipliers to derive its sufficient condition:

$$V_\lambda = V - \lambda_1(e_1^T e_1 - 1) - \lambda_2(e_2^T e_2 - 1) - \ldots - \lambda_m(e_m^T e_m - 1)$$

- Thus

$$\frac{\partial V_\lambda}{\partial e_j} = \frac{\partial(e_1^T S e_1 + \ldots + e_m^T S e_m)}{\partial e_j} - \lambda_j \frac{\partial(e_j^T e_j - 1)}{\partial e_j}$$

$$= \frac{\partial(e_j^T S e_j)}{\partial e_j} - 2\lambda_j e_j$$

$$= 2Se_j - 2\lambda_j e_j$$

$$= 0$$

Please refer to the note of MatrixCalculus, eq. (D.16)

- Hence the condition $Se_j = \lambda e_j$ need to be satisfied. This means that e_j should be the eigenvectors of the scatter matrix S.

Proof of PCA

- We thus have the property that the optimal e_j's should be the eigenvectors of S.
- Since V shall be maximized:

$$\begin{aligned}V &= e_1^T S e_1 + \ldots + e_m^T S e_m \\ &= \lambda_1 e_1^T e_1 + \lambda_2 e_2^T e_2 + \ldots + \lambda_m e_m^T e_m \\ &= \lambda_1 \| e_1 \|^2 + \lambda_2 \| e_2 \|^2 + \ldots + \lambda_m \| e_m \|^2 \\ &= \lambda_1 + \lambda_2 + \ldots + \lambda_m\end{aligned}$$

Choosing the largest eigenvalues for λ_1, .., λ_m can maximize V. The corresponding eigenvectors are thus the solutions for e_1, .., e_m.