Linear Algebra

LIN Qingfeng
February 26, 2021

1 Main Concept

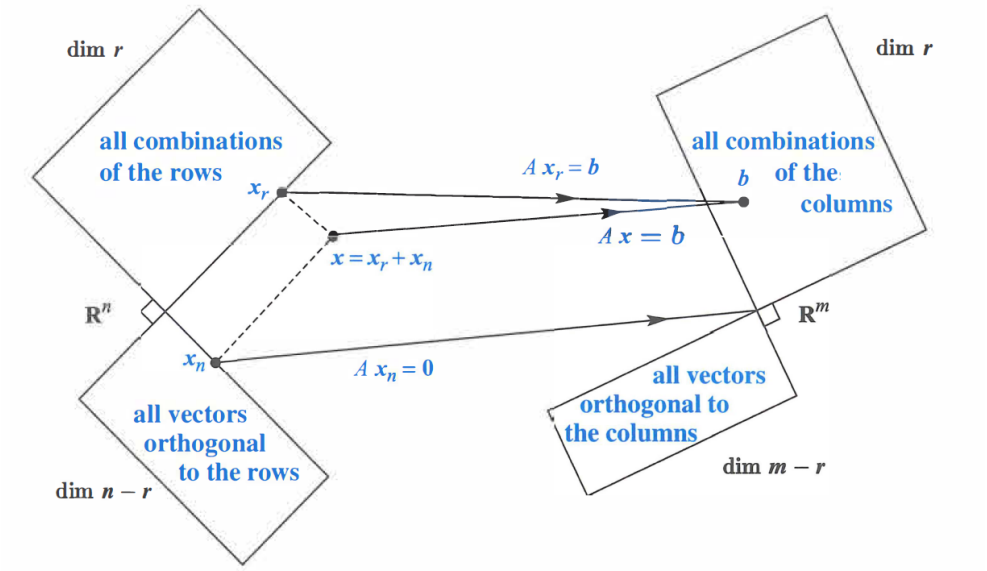

Figure 1: four subspace

- **Space** It is the most important concept in Linear Algebra. For any matrix, there are four fundamental vector spaces, there are as followed,

$$\begin{aligned} \operatorname{col}(A) &= \text{ The Column Space of } A = \{Ax : x \in R^m\} \\ \text{null }(A) &= \text{ The Nullspace of } A = \{x \in R^m : Ax = 0\} \\ \operatorname{row}(A) &= \text{ The Row Space of } A = \left\{A^\top x : x \in R^n\right\} \\ \text{null }\left(A^\top\right) &= \text{ The Left Nullspace of } A = \left\{x \in R^n : A^\top x = 0\right\} \end{aligned} \tag{1}$$

- **Linear transformation and its relation to matrices.**

  In fact, transformation is another word of "function". That is to say, we can regard 'matrix' as an operator. For example, giving a matrix A,

$$A = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \tag{2}$$

  When matrix A acts on a random vector x, $x \in \mathbb{R}^2$. It makes x rotate θ counterclockwise.

$$A = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} \tag{3}$$

  When matrix A acts on a random vector x, $x \in \mathbb{R}^2$. It makes x stretch 2 times.

- **Projection**
  Projection of b onto a is a vector p, the error $e = b - p$ is perpendicular to a , so we have,

$$\begin{aligned} & a^T(b - xa) = 0 \\ \implies & x = \frac{a^T b}{a^T a} \end{aligned} \tag{4}$$

  We can write the projection using a projection matrix,

$$\begin{aligned} & p = Pb \\ & P = \frac{aa^T}{a^T a} \end{aligned} \tag{5}$$

  Projection of vector b onto a set of independent vectors $\{a_i\}$ is a vector p, the error $e = b - p$ is perpendicular to each of a , so we have,

$$\begin{aligned} \implies & A^T(b - A\hat{x}) = 0 \\ \implies & A^T A\hat{x} = A^T b \end{aligned} \tag{6}$$

  We again have a projection matrix,

$$P = A\left(A^T A\right)^{-1} A^T \tag{7}$$

- **Whiten Transformation**

  Let $\mathbf{x}$ be a vector of zero-mean data. Form its covariance matrix,

$$\mathbf{\Sigma} = E\left(\mathbf{x}\mathbf{x}^T\right) \tag{8}$$

  If the data points in $\mathbf{x}$ are correlated, then their covariance, $\mathbf{\Sigma}$, will NOT be a diagonal matrix.

  In order to decorrelate the data, we need to transform it so that the transformed data will have a diagonal covariance matrix. This transform can be found by solving the eigenvalue problem. We find the eigenvectors and associated eigenvalues of $\mathbf{\Sigma}$ by solving

$$\mathbf{\Phi}^T \mathbf{\Sigma} \mathbf{\Phi} = \mathbf{\Lambda} \tag{9}$$

  If we wish to apply this diagonalizing transform to a single vector of data we just form:

$$\mathbf{y} = \mathbf{\Phi}^T \mathbf{x} \tag{10}$$

  Thus, the data $\mathbf{y}$ has been decorrelated: its covariance, $E\left[\mathbf{y}\mathbf{y}^T\right]$ is now a diagonal matrix, $\mathbf{\Lambda}$.

  The diagonal elements (eigenvalues) in $\mathbf{\Lambda}$ may be the same or different. If we make them all the same, then this is called whitening the data. Since each eigenvalue determines the length of its associated eigenvector, the covariance will correspond to an ellipse when the data is not whitened, and to a sphere (having all dimensions the same length, or uniform) when the data is whitened. Whitening is easy:

$$\mathbf{\Lambda}^{-1/2} \mathbf{\Phi}^T \mathbf{\Sigma} \mathbf{\Phi} \mathbf{\Lambda}^{-1/2} = \mathbf{I} \tag{11}$$

$$\mathbf{w} = A_w^T \mathbf{x}, \quad A_w = \mathbf{\Phi} \mathbf{\Lambda}^{-1/2} \tag{12}$$

- **Eigenvalues and Eigenvectors**
  For any square matrix A, almost all vectors change direction, when they are multiplied by A. Certain exceptional vectors x are in the same direction as Ax. Those are the "eigenvectors". And the eigenvalue λ tells whether the special vector $\boldsymbol{x}$ is stretched or shrunk or reversed or left unchanged - when it is multiplied by A.

  The basic equation is,

$$Ax = \lambda x \tag{13}$$

  There are some useful properties,
  (1) When A is squared. the eigenvectors stay the same. The eigenvalues are squared.
  (2) The projection matrix P has eigenvalues $\lambda = 1$ and $\lambda = 0$
  (3) The orthogonal matrix has absolute value of each λ is $|\lambda| = 1$
  (4) $Ax = 0x$ means that this eigenvector x is in the nullspace.

- **The Singular Value Decomposition**
  For any matrix A, it always has,

$$A = U\Sigma V^T \tag{14}$$

  where U is orthogonal, Σ is diagonal, and V is orthogonal.

  (how it works:) We can think of A as a linear transformation taking a vector $\mathbf{v}_1$ in its row space to a vector $\mathbf{u}_1 = A\mathbf{v}_1$ in its column space. The SVD arises from finding an orthogonal basis for the row space that gets transformed into an orthogonal basis for the column space: $A\mathbf{v}_i = \sigma_i \mathbf{u}_i$

  It's not hard to find an orthogonal basis for the row space - the GramSchmidt process gives us one right away. But in general, there's no reason to expect A to transform that basis to another orthogonal basis.

  The heart of the problem is to find an orthonormal basis $\mathbf{v}_1, \mathbf{v}_2, \ldots \mathbf{v}_r$ for the row space of A for which,

$$A \begin{bmatrix} \mathbf{v}_1 & \mathbf{v}_2 & \cdots & \mathbf{v}_r \end{bmatrix} = \begin{bmatrix} \sigma_1\mathbf{u}_1 & \sigma_2\mathbf{u}_2 & \cdots & \sigma_r\mathbf{u}_r \end{bmatrix} \tag{15}$$

  with $\mathbf{u}_1, \mathbf{u}_2, \ldots \mathbf{u}_r$ an orthonormal basis for the column space of A. Once we add in the nullspaces, this equation will become $AV = U\Sigma$. (We can complete the orthonormal bases $\mathbf{v}_1, \ldots \mathbf{v}_r$ and $\mathbf{u}_1, \ldots \mathbf{u}_r$ to orthonormal bases for the entire space any way we want. Since $\mathbf{v}_{r+1}, \ldots \mathbf{v}_n$ will be in the nullspace of A, the diagonal entries $\sigma_{r+1}, \ldots \sigma_n$ will be 0.

  Rather than solving for U, V and Σ simultaneously, we multiply both sides by $A^T = V\Sigma^T U^T$ to get:

$$\begin{aligned} A^TA &= V\Sigma U^{-1}U\Sigma V^T \\ &= V\Sigma^2 V^T \\ &= V \begin{bmatrix} \sigma_1^2 & & & \\ & \sigma_2^2 & & \\ & & \ddots & \\ & & & \sigma_n^2 \end{bmatrix} V^T. \end{aligned} \tag{16}$$

  This is in the form $Q\Lambda Q^T$; we can now find V by diagonalizing the symmetric positive definite (or semidefinite) matrix A^TA. The columns of V are eigenvectors of A^TA and the eigenvalues of A^TA are the values σ_i^2. (We choose σ_i to be the positive square root of λ_i.) To find U, we do the same thing with AA^T.

  Take a simple example,

$$\begin{bmatrix} 4 & 3 \\ 8 & 6 \end{bmatrix} = \frac{1}{\sqrt{5}} \begin{bmatrix} 1 & 2 \\ 2 & -1 \end{bmatrix} \quad \begin{bmatrix} \sqrt{125} & 0 \\ 0 & 0 \end{bmatrix} \quad \begin{bmatrix} .8 & .6 \\ .6 & -.8 \end{bmatrix} \tag{17}$$

  where $\mathbf{v}_1$ is an orthonormal basis for the row space. $\mathbf{u}_1$ is an orthonormal basis for the column space. $\mathbf{v}_2$ is an orthonormal basis for the nullspace. $\mathbf{u}_2$ is an orthonormal basis for the left nullspace.

2 Matrix

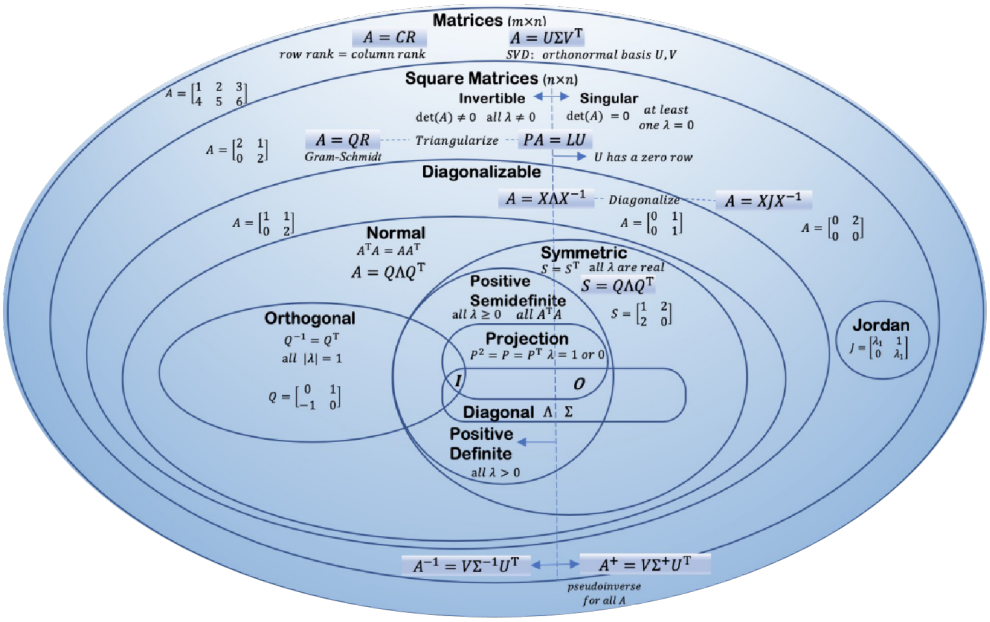

Figure 2: matrix world

- **Orthogonal matrix**:
  If the columns of $Q = \begin{bmatrix} \mathbf{q}_1 & \dots & \mathbf{q}_n \end{bmatrix}$ are orthonormal, then $Q^TQ = I$ is the identity

  A square orthonormal matrix Q is called an orthogonal matrix. If Q is square then $Q^\dagger Q = I$ tells us that $Q^T = Q^{-1}$.

- **Symmetric matrix**:
  If A is any matrix, the matrices A ^{T}A and AA ^{T} are both symmetric.
  A matrix is symmetric if and only if it is orthogonally diagonalizable.

$$\mathbf{A}^T = \left(\mathbf{EDE}^T\right)^T = \mathbf{E}^{TT}\mathbf{D}^T\mathbf{E}^T = \mathbf{EDE}^T = \mathbf{A} \tag{18}$$

  A symmetric matrix is diagonalized by a matrix of its orthonormal eigenvectors.

- **Projection matrix**:
  A projection matrix P, it has these properties,

$$\begin{aligned} P^T &= P \\ P^2 &= P \end{aligned} \tag{19}$$

  Suppose P has a QR decomposition, where R is invertible. Then P simplifies:

$$P = QQ^T \tag{20}$$

- **Vandermonde**:
  It can be used in curve fitting

$$\mathbf{H} = \begin{bmatrix} 1 & t_0 & t_0^2 \\ 1 & t_1 & t_1^2 \\ \vdots & \vdots & \vdots \\ 1 & t_{N-1} & t_{N-1}^2 \end{bmatrix} \tag{21}$$

3 Application

3.1 Estimation Theory

3.1.1 Best Linear Unbiased Estimator (BLUE)

- **Motivation**: It frequently occurs in practice that the MVU estimator, if it exists, cannot be found. For instance, we may not know the PDF of the data or even be willing to assume a model for it. Thus, it is reasonable to resort to a suboptimal estimator. A common approach is to restrict the estimator to be linear in the data and find the linear estimator that is unbiased and has minimum variance.
- **Data Model**:

$$\mathbf{x} = \mathbf{H}\boldsymbol{\theta} + \mathbf{w} \tag{22}$$

  where $E(\mathbf{w}) = \mathbf{0}$ and $\mathbf{C}_w = \mathbf{C}$.
- **Estimator**:

$$\hat{\boldsymbol{\theta}} = \left(\mathbf{H}^T\mathbf{C}^{-1}\mathbf{H}\right)^{-1}\mathbf{H}^T\mathbf{C}^{-1}\mathbf{x} \tag{23}$$

- **Optimality**: $\hat{\boldsymbol{\theta}}$ has the minimum variance of all unbiased estimators that are linear in $\mathbf{x}$.
- **Comments**: If $\mathbf{w}$ is a Gaussian random vector so that $\mathbf{w} \sim \mathcal{N}(\mathbf{0}, \mathbf{C})$, then $\hat{\boldsymbol{\theta}}$ is also the MVU estimator (for all linear or nonlinear functions of $\mathbf{x}$). Unfortunately, without knowledge of the PDF there is no way to determine the loss in performance by resorting to a BLUE.
- **Examples**: Same as above, if we observe,

$$x[n] = A + w[n] \quad n = 0, 1, \ldots, N-1 \tag{24}$$

  where $w[n]$ is white noise with variance σ^2, then the problem is to estimate A.

$$\begin{aligned} \hat{A} &= \frac{1^T \frac{1}{\sigma^2} 1\mathbf{x}}{1^T \frac{1}{\sigma^2} 1} \\ &= \frac{1}{N}\sum_{n=0}^{N-1} x[n] = \bar{x} \end{aligned} \tag{25}$$

$$\begin{aligned} \text{var}(\hat{A}) &= \frac{1}{1^T \frac{1}{\sigma^2} 1} \\ &= \frac{\sigma^2}{N} \end{aligned} \tag{26}$$

3.1.2 Least Squares Estimator (LSE)

- **Data Model**: The model is,

$$\mathbf{x} = \mathbf{s}(\boldsymbol{\theta}) + \mathbf{w} \tag{27}$$

  where $\mathbf{s}$ is a known function of $\boldsymbol{\theta}$ and the noise $\mathbf{w}$ has zero mean.

- **Estimator**: $\hat{\boldsymbol{\theta}}$ is the value of $\boldsymbol{\theta}$ that minimizes,

$$\begin{aligned} J(\boldsymbol{\theta}) &= (\mathbf{x} - \mathbf{s}(\boldsymbol{\theta}))^T (\mathbf{x} - \mathbf{s}(\boldsymbol{\theta})) \\ &= \sum_{n=0}^{N-1} (x[n] - s[n; \boldsymbol{\theta}])^2 \end{aligned} \tag{28}$$

- **Optimality**: Not optimal in general.
- **Comments**: The fact that we are minimizing a LS error criterion does not in general translate into minimizing the estimation error. Also, if $\mathbf{w}$ is a Gaussian random vector with $\mathbf{w} \sim \mathcal{N}\left(\mathbf{0}, \sigma^2 \mathbf{I}\right)$, then the LSE is equivalent to the MLE.
- **Example**: Assume that the signal model is $s[n] = A$ and we observe $x[n]$ for $n = 0, 1, \ldots, N-1$. Then, according to the LS approach, we can estimate A by minimizing,

$$J(A) = \sum_{n=0}^{N-1} (x[n] - A)^2. \tag{29}$$

Differentiating with respect to A and setting the result equal to zero produces,

$$\begin{aligned} \hat{A} &= \frac{1}{N} \sum_{n=0}^{N-1} x[n] \\ &= \bar{x} \end{aligned} \tag{30}$$

3.1.3 Recursive Least Square Estimation

- A linear recursive estimator can be written in the form

$$\begin{aligned} y_k &= H_k x + v_k \\ \hat{x}_k &= \hat{x}_{k-1} + K_k \left(y_k - H_k \hat{x}_{k-1}\right) \end{aligned} \tag{31}$$

That is, we compute $\hat{x}_k$ on the basis of the previous estimate $\hat{x}_{k-1}$ and the new measurement y_k.K_k is a matrix to be determined called the estimator gain matrix. The quantity $(y_k - H_k \hat{x}_{k-1})$ is called the correction term. Note that if the correction

The estimation error mean can be computed as

$$\begin{aligned} E\left(\epsilon_{x,k}\right) &= E\left(x - \hat{x}_k\right) \\ &= E\left[x - \hat{x}_{k-1} - K_k \left(y_k - H_k \hat{x}_{k-1}\right)\right] \\ &= E\left[\epsilon_{x,k-1} - K_k \left(H_k x + v_k - H_k \hat{x}_{k-1}\right)\right] \\ &= E\left[\epsilon_{x,k-1} - K_k H_k \left(x - \hat{x}_{k-1}\right) - K_k v_k\right] \\ &= \left(I - K_k H_k\right) E\left(\epsilon_{x,k-1}\right) - K_k E\left(v_k\right) \end{aligned} \tag{32}$$

In order to determine K_k, we choose to minimize the sum of the variances of the estimation errors at time k

$$\begin{aligned} J_k &= E\left[\left(x_1 - \hat{x}_1\right)^2\right] + \cdots + E\left[\left(x_n - \hat{x}_n\right)^2\right] \\ &= E\left(\epsilon_{x1,k}^2 + \cdots + \epsilon_{xn,k}^2\right) \\ &= E\left(\epsilon_{x,k}^T \epsilon_{x,k}\right) \\ &= E\left[\mathrm{Tr}\left(\epsilon_{x,k} \epsilon_{x,k}^T\right)\right] \\ &= \mathrm{Tr}\, P_k \end{aligned} \tag{33}$$

$$
\begin{aligned}
P_k =& E\left(\epsilon_{x,k}\epsilon_{x,k}^T\right) \\
=& E\left\{\left[\left(I-K_kH_k\right)\epsilon_{x,k-1}-K_kv_k\right]\left[\cdots\right]^T\right\} \\
=& \left(I-K_kH_k\right)E\left(\epsilon_{x,k-1}\epsilon_{x,k-1}^T\right)\left(I-K_kH_k\right)^T- \\
& K_kE\left(v_k\epsilon_{x,k-1}^T\right)\left(I-K_kH_k\right)^T-\left(I-K_kH_k\right)E\left(\epsilon_{x,k-1}v_k^T\right)K_k^T+ \\
& K_kE\left(v_kv_k^T\right)K_k^T
\end{aligned} \tag{34}
$$

Now note that $\epsilon_{x,k-1}$ [the estimation error at time $(k-1)$] is independent of v_k (the measurement noise at time k). Therefore,

$$
P_k = \left(I-K_kH_k\right)P_{k-1}\left(I-K_kH_k\right)^T+K_kR_kK_k^T \tag{35}
$$

We need to find the value of K_k that makes the cost function J_k as small as possible. The mean of the estimation error is zero for any value of K_k, So if we choose K_k to make the cost function (i.e., the trace of P_k) small then the estimation error will not only be zero-mean, but it will also be consistently close to zero.

$$
\frac{\partial J_k}{\partial K_k} = 2\left(I-K_kH_k\right)P_{k-1}\left(-H_k^T\right)+2K_kR_k \tag{36}
$$

$$
K_k = P_{k-1}H_k^T\left(H_kP_{k-1}H_k^T+R_k\right)^{-1} \tag{37}
$$

3.2 Optimization

- **KKT matrix**

  There are many useful optimization algorithms based on solving the KKT matrix, such as newton method, primal-dual method.

$$
\begin{bmatrix}
\nabla^2 f_0(x)+\sum_{i=1}^m \lambda_i\nabla^2 f_i(x) & Df(x)^T & A^T \\
-\operatorname{diag}(\lambda)Df(x) & -\operatorname{diag}(f(x)) & 0 \\
A & 0 & 0
\end{bmatrix}
\begin{bmatrix}
\Delta x \\ \Delta\lambda \\ \Delta\nu
\end{bmatrix}
=-
\begin{bmatrix}
r_{\text{dual}} \\ r_{\text{cent}} \\ r_{\text{pri}}
\end{bmatrix} \tag{38}
$$

  In each iteration, we need to solve this KKT matrix to get the direction of primal variables and dual variables.

- **Support Vector Machine(SVM)**

$$
\begin{aligned}
&\min_{\mathbf{w}\in H,b\in\mathbb{R}} \mathbf{w}^T\mathbf{w} \\
&\text{s.t. } \sum_{i=1}^m y_i\left(\langle\mathbf{w},\mathbf{x}_i\rangle+b\right)\geq c
\end{aligned} \tag{39}
$$

  The Lagrange dual problem can be formulation as

$$
\begin{array}{ll}
\max_\lambda & -\frac{1}{2}\lambda^T\mathbf{K}\lambda+\lambda^T\mathbf{1}_{n\times 1} \\
\text{s.t.} & \lambda\succeq 0 \\
& \lambda^T\mathbf{Y}\mathbf{1}_{n\times 1}=0
\end{array} \tag{40}
$$

3.3 Machine Learning

3.3.1 Kalman Filter

- The data model is,

$$\begin{aligned} x_k &= F_{k-1}x_{k-1} + G_{k-1}u_{k-1} + w_{k-1} \\ y_k &= H_k x_k + v_k \\ E\left(w_k w_j^T\right) &= Q_k \delta_{k-J} \\ E\left(v_k v_j^T\right) &= R_k \delta_{k-J} \\ E\left(w_k v_j^T\right) &= 0 \end{aligned} \tag{41}$$

- The error between the true state and the estimated state is denoted as,

$$\tilde{x}_k = x_k - \hat{x}_k \tag{42}$$

- Suppose we want to find the estimator that minimizes (at each time step) a weighted two-norm of the expected value of the estimation error $\tilde{x}_k$:

$$\min E\left[\tilde{x}_k^T S_k \tilde{x}_k\right] \tag{43}$$

where S_k is diagonal positive definite user-defined weighting matrix.

- If $\{w_k\}$ and $\{v_k\}$ are Gaussian, zero-mean, uncorrelated, and white, then the Kalman filter is the solution to the above problem.

3.3.2 Sequential Kalman Filter

- **Model**:

$$\begin{aligned} x_k &= F_{k-1}x_{k-1} + G_{k-1}u_{k-1} + w_{k-1} \\ y_k &= H_k x_k + v_k \\ w_k &\sim (0, Q_k) \\ v_k &\sim (0, R_k) \end{aligned} \tag{44}$$

where w_k and v_k are uncorrelated white noise sequences. The measurement covariance R_k is a diagonal matrix given as

$$R_k = \operatorname{diag}\left(R_{1k}, \cdots, R_{rk}\right)$$

At each time step k, the measurement-update equations are given as follows.
(a) Initialize the a posteriori estimate and covariance as

$$\begin{aligned} \hat{x}_{0k}^+ &= \hat{x}_k^- \\ P_{0k}^+ &= P_k^- \end{aligned} \tag{45}$$

(b) For $i = 1, \cdots, r$ (where r is the number of measurements), perform the following:

$$\begin{aligned} K_{ik} &= \frac{P_{i-1,k}^+ H_{ik}^T}{H_{ik} P_{i-1,k}^+ H_{ik}^T + R_{ik}} \\ &= \frac{P_{ik}^+ H_{ik}^T}{R_{ik}} \\ \hat{x}_{ik}^+ &= \hat{x}_{i-1,k}^+ + K_{ik}\left(y_{ik} - H_{ik}\hat{x}_{i-1,k}^+\right) \\ P_{ik}^+ &= \left(I - K_{ik}H_{ik}\right) P_{i-1,k}^+ \left(I - K_{ik}H_{ik}\right)^T + K_{ik}R_{ik}K_{ik}^T \end{aligned} \tag{46}$$

(c) Assign the a posteriori estimate and covariance as

$$\begin{aligned} \hat{x}_k^+ &= \hat{x}_{rk}^+ \\ P_k^+ &= P_{rk}^+ \end{aligned} \tag{47}$$

3.3.3 Principal Component Analysis

- Principal component analysis (PCA) has been called one of the most valuable results from applied linear algebra. PCA is used abundantly because it is a simple, non-parametric method of extracting relevant information from confusing data sets. With minimal additional effort PCA provides a roadmap for how to reduce a complex data set to a lower dimension to reveal the sometimes hidden, simplified structure that often underlie it.

  There are some important assumptions in PCA:

- **Linearity**: Linearity frames the problem as a change of basis. Several areas of research have explored how applying a nonlinearity prior to performing PCA could extend this algorithm - this has been termed kernel PCA.
- **Mean and variance are sufficient statistics**: The formalism of sufficient statistics captures the notion that the mean and the variance entirely describe a probability distribution. The only class of probability distributions that are fully described by the first two moments are exponential distributions (e.g. Gaussian, Exponential, etc). In order for this assumption to hold, the probability distribution of $\mathbf{x_i}$ must be exponentially distributed. Deviations from this could invalidate this assumption.
- **Large variances have important dynamics**: This assumption also encompasses the belief that the data has a high SNR. Hence, principal components with larger associated variances represent interesting dynamics, while those with lower variances represent noise.
- **The principal components are orthogonal**: This assumption provides an intuitive simplification that makes PCA soluble with linear algebra decomposition techniques.
- **SOLVING PCA: EIGENVECTORS OF COVARIANCE**

  The data set is $\mathbf{X}$, an $m \times n$ matrix, where m is the number of measurement types and n is the number of samples. The goal is, Find some orthonormal matrix $\mathbf{P}$ where $\mathbf{Y} = \mathbf{PX}$ such that $\mathbf{C_Y} \equiv \frac{1}{n-1}\mathbf{YY}^T$ is diagonalized. The rows of $\mathbf{P}$ are the principal components of $\mathbf{X}$

$$\begin{aligned}
\mathbf{C_Y} &= \frac{1}{n-1}\mathbf{YY}^T \\
&= \frac{1}{n-1}(\mathbf{PX})(\mathbf{PX})^T \\
&= \frac{1}{n-1}\mathbf{PXX}^T\mathbf{P}^T \\
&= \frac{1}{n-1}\mathbf{P}\left(\mathbf{XX}^T\right)\mathbf{P}^T \\
\mathbf{C_Y} &= \frac{1}{n-1}\mathbf{PAP}^T
\end{aligned} \tag{48}$$

  Note that we defined a new matrix $\mathbf{A} \equiv \mathbf{XX}^T$, where $\mathbf{A}$ is symmetric . A symmetric matrix can be diagonalized by orthogonal matrix, $\mathbf{A} = \mathbf{EDE}^T$, we selection, $\mathbf{P} \equiv \mathbf{E^T}$

$$\begin{aligned}
\mathbf{C_Y} &= \frac{1}{n-1}\mathbf{PAP}^T \\
&= \frac{1}{n-1}\mathbf{P}\left(\mathbf{P}^T\mathbf{DP}\right)\mathbf{P}^T \\
&= \frac{1}{n-1}\left(\mathbf{PP}^T\right)\mathbf{D}\left(\mathbf{PP}^T\right) \\
&= \frac{1}{n-1}\left(\mathbf{PP}^{-1}\right)\mathbf{D}\left(\mathbf{PP}^{-1}\right) \\
\mathbf{C_Y} &= \frac{1}{n-1}\mathbf{D}
\end{aligned} \tag{49}$$

- **A MORE GENERAL SOLUTION: SVD**

  Let $\mathbf{X}$ be an arbitrary $m \times n$ matrix and $\mathbf{X}^T\mathbf{X}$ be a rank r, square, symmetric $n \times n$ matrix.

- $\{\hat{\mathbf{v}}_1, \hat{\mathbf{v}}_2, \ldots, \hat{\mathbf{v}}_r\}$ is the set of orthonormal $n \times 1$ eigenvectors with associated eigenvalues $\{\lambda_1, \lambda_2, \ldots, \lambda_r\}$ for the symmetric matrix $\mathbf{X}^T\mathbf{X}$

$$\left(\mathbf{X}^T\mathbf{X}\right)\hat{\mathbf{v}}_i = \lambda_i\hat{\mathbf{v}}_i \tag{50}$$

- $\sigma_i \equiv \sqrt{\lambda_i}$ are positive real and termed the singular values.
- $\{\hat{\mathbf{u}}_1, \hat{\mathbf{u}}_2, \ldots, \hat{\mathbf{u}}_r\}$ is the set of orthonormal $m \times 1$ vectors defined by $\hat{\mathbf{u}}_\mathbf{i} \equiv \frac{\mathbf{1}}{\sigma_i}\mathbf{X}\hat{\mathbf{v}}_\mathbf{i}$
- We summarize the equations above, we have,

$$\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T \tag{51}$$

- Compared this method with eigenvector method, we define a new matrix Y,

$$\mathbf{Y} = \frac{1}{\sqrt{n-1}}\mathbf{X}^T \tag{52}$$

  where each column of $\mathbf{Y}$ has zero mean. The definition of $\mathbf{Y}$ becomes clear by analyzing $\mathbf{Y}^T\mathbf{Y}$.

$$\begin{aligned}
\mathbf{Y}^T\mathbf{Y} &= \left(\frac{1}{\sqrt{n-1}}\mathbf{X}^T\right)^T\left(\frac{1}{\sqrt{n-1}}\mathbf{X}^T\right) \\
&= \frac{1}{n-1}\mathbf{X}^{TT}\mathbf{X}^T \\
&= \frac{1}{n-1}\mathbf{X}\mathbf{X}^T \\
\mathbf{Y}^T\mathbf{Y} &= \mathbf{C}_\mathbf{X}
\end{aligned} \tag{53}$$

- By construction $\mathbf{Y}^T\mathbf{Y}$ equals the covariance matrix of $\mathbf{X}$. We know that the principal components of $\mathbf{X}$ are the eigenvectors of $\mathbf{C}_\mathbf{X}$. If we calculate the SVD of $\mathbf{Y}$, the columns of matrix $\mathbf{V}$ contain the eigenvectors of $\mathbf{Y}^T\mathbf{Y} = \mathbf{C}_\mathbf{X}$. Therefore, the columns of $\mathbf{V}$ are the principal components of $\mathbf{X}$. This second algorithm is