Review of Linear Algebra.

1. Multiplication.

x is a column vector. $x^T y$ is scalar xy^T is a matrix
x^T is a row vector

$AB = A[b_1 \cdots b_n] = [Ab_1 \cdots Ab_n]$

$AB = \begin{bmatrix} -a_1^T- \\ \vdots \\ a_n^T \end{bmatrix} B = \begin{bmatrix} a_1^T B \\ \vdots \\ a_n^T B \end{bmatrix}$

$AB = (a_1 \cdots a_n) \begin{bmatrix} b_1^T \\ \vdots \\ b_n^T \end{bmatrix} = \sum_{i=1}^n a_i b_i^T$ sum of matrix

$AB = \begin{bmatrix} a_1^T \\ \vdots \\ a_n^T \end{bmatrix} [b_1 \cdots b_n] = \begin{bmatrix} a_1^T b_1 & a_1^T b_n \cdots \\ \vdots & \ddots \\ & \cdots a_n^T b_n \end{bmatrix}$ matrix of scalar

$U \Lambda U^T$ (when Λ is diagonal) $= [u_1 \cdots u_n] \begin{bmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_n \end{bmatrix} \begin{bmatrix} u_1^T \\ \vdots \\ u_n^T \end{bmatrix} = \cancel{\begin{bmatrix} \lambda_1 u_1 & & \\ & \ddots & \\ & & \lambda_n u_n \end{bmatrix} \begin{bmatrix} u_1^T \\ \vdots \\ u_n^T \end{bmatrix}}$

$= [\lambda_1 u_1 \cdots \lambda_n u_n] \begin{bmatrix} u_1^T \\ \vdots \\ u_n^T \end{bmatrix} = \sum_i \lambda_i u_i u_i^T$

2. Concepts to describe matrix.

① Hermitian / Symmetric: $A^H = A$. $A^T = A$

② Unitary / Orthogonal: $U^H U = U U^H = I$ $U^T U = U U^T = I$

Orthogonal matrix have some nice properties:

① $U^T U = \begin{bmatrix} u_1^T \\ \vdots \\ u_n^T \end{bmatrix} [u_1 \cdots u_n] = \begin{bmatrix} u_1^T u_1 & u_1^T u_2 \cdots \\ \cdots & \cdots \\ \cdots & \cdots \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & \cdots \\ 0 & 1 & 0 & \cdots \\ & \cdots & & \cdots \end{bmatrix}$

cols of U form a basis

② norm preserving: $\|Ux - Uy\|^2 = (Ux - Uy)^T (Ux - Uy)$
$= (x-y)^T U^T U (x-y) = \|x-y\|^2$

③ PSD / PD.

matrix A is PD if $\forall x \neq 0$, $x^T A x > 0$.

PSD if $\forall x \neq 0$, $x^T A x \geq 0$

3. Symmetric matrix have nice properties. They have special decompostion and their eigenvalues are indicative of other properties.

① Spectral Thm: If A is Hermitian. then $A = U \Lambda U^H$ where U is unitary and Λ is real diagonal.

If A is real and $A = A^T$ (Symmetric), then $A = U \Lambda U^T$ where U is ortho.

This is actually eigen-decomp of A:

$A = U \Lambda U^T \Rightarrow AU = U\Lambda \Rightarrow A[u_1 \cdots u_n] = [\lambda_1 u_1 \cdots \lambda_n u_n]$

$\Rightarrow A u_i = \lambda_i u_i$. λ_i are eigenvalues.

② For symmetric matrix. its eigenvalues are indicative of PD.

A is Hermitian/Symmetric $\Rightarrow \begin{cases} A \text{ is PD} \Leftrightarrow \lambda_i > 0 \\ A \text{ is PSD} \Leftrightarrow \lambda_i \geq 0 \end{cases}$

note $x^T A x = \sum_{i=1}^n \lambda_i x^T u_i u_i^T x = \sum_{i=1}^n \lambda_i \|u_i^T x\|^2$

4. PD is strongly related to symmetry. Sometimes, we just define PD in terms of symmetric matrix

① A matrix can be PD but not symmetric. $\begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix}$

but its eigenvalues are now complex: $\lambda_i = 1 \pm i$

we don't like this!

② most time, we care PD in terms of symmetric matrix

5. example.

For Gaussian. Σ is symmetric and PD.

① Symmetric $\Rightarrow \Sigma = U \Lambda U^T$.

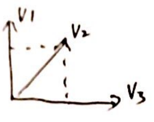
② PD $\Rightarrow \lambda_i > 0$.

$(x'=U^T x,\ \mu'=U^T\mu)$

so $(x-\mu)^T\Sigma^{-1}(x-\mu) = (x-\mu)^T U\Lambda^{-1}U^T(x-\mu) = (x'-\mu')^T\Lambda(x'-\mu')$

$=\sum\frac{(x_i'-\mu_i')^2}{\lambda_i}$ since $\lambda_i>0$. result is an ellipse!

6. Orthogonal and Independence. (note we talk about vectors here)



v_1 and v_3 are orthogonal. also independent.
v_2 and v_3 are independent but not orthogonal.
$\Rightarrow$ independence is a weaker condition.

v_2 and v_3 are independent, iff v_2 has an orthogonal component w.r.t. v_3. note v_2 could be decomposed. and one component is $\perp v_3$.

7. Intuition of Subspace.

① A subspace is a box. If ~~contains some in~~ can be spanned by some independent vectors

② A matrix can be seen as a collection of vectors $[a_1 \cdots a_p]$ thus has a correspondence to a subspace. Whether its col vectors are linearly independent is an interesting property.

③ if $a_1 \sim a_p$ are linearly independent. let $\langle A\rangle$ denote subspace spanned
$\dim(\langle A\rangle)=\text{rank}(A)=p$. denote $a_i\in R^N$. thus $p\le N$.
$\dim(\langle A\rangle^\perp)=N-p$.

Intuition: A n-length vector lives in n-dim space, or can be projected to space whose dim<n. But cannot be lifted to space whose dim>n. so if $a_1\cdots a_p$ are independent, then $p\le N$.

④ Space $\langle A\rangle$ and $\langle A\rangle^\perp$ form orthogonal decomposition of space R^N. Any vector in $\langle A\rangle$ is orthogonal to every vector in $\langle A\rangle^\perp$

8. signal as a vector

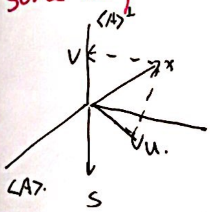
① A signal can be seen as a vector

e.g. $s(n)=\theta_2 n^2+\theta_1 n+\theta_0 = [\cancel{\phi}1, n, n^2]\begin{bmatrix}\theta_0\\ \theta_1\\ \theta_2\end{bmatrix}$ $\in R^3$.

② A bunch of signals live in a subspace:
$s(1) \cdots s(n)$ lives in R^3

③ Assumption.
we assume $x=s+w$. s is the signal. w is noise and x is what we observed.

We Assume s lives in space $\langle A\rangle$. w must have some component in $\langle A\rangle^\perp$.



$x=u+v. =s+w.$

w may not totally live in $\langle A\rangle^\perp$

idea: if we can project x into $\langle A\rangle$. we can have u, thus some sense of s. But How to project a vector on-to a subspace?

9. Projection is denoted by an operator π_A. where $\langle A\rangle$ is the subspace to project on.
$\pi_A x=u$.

① matrix A can express π_A: $\pi_A = A(A^HA)^{-1}A^H$.

note: let $u\in\langle A\rangle$, $v\in\langle A\rangle^\perp=\langle B\rangle$. then $u=A\theta$, $v=B\phi$. Basis.

$\pi_A x = A(A^HA)^{-1}A^HA\theta + A(A^HA)^{-1}A^HB\phi$
$= A\theta = u.$

Review of MVG.

1. Covariance.

① Covariance is a measure of joint variability of two random Variables

e.g. if X has large value, then Y is likely to have large value. then X and Y have positive covariance

② $X = [x_1 \cdots x_m]^T$ where each component x_i is a random variable, we call X a random vector.

③ Random vector has covariance matrix: pairwise covariance of components.

$$\begin{bmatrix} cov(x_1,x_1) & \cdots & \\ \vdots & \ddots & \\ \cdots & & cov(x_n,x_n) \end{bmatrix}$$

Calculation

$cov(x,x) = E((X-\mu)(X-\mu)^T] = \Sigma$

④ $cov(x,x) = Var(x)$. For one r.variable, its covariance with itself is called Variance, which is a characterization of uncertainty.

2. MVG. Definition.

$$\phi(x) = \phi(x;\mu,\Sigma) = (2\pi)^{-\frac{N}{2}} |\Sigma|^{-\frac{1}{2}} \exp\{-\frac{1}{2}(x-\mu)^T \Sigma^{-1}(x-\mu)\}$$

where Σ is symmetric and PD.

① MVG is fully characterized by μ, Σ. denote $X \sim N(\mu,\Sigma)$

② Properties of MVG can be proven by its characteristic function.

characteristic function:

$$\Phi_x(\omega) = E[e^{-j\omega^T x}] = \int e^{-j\omega^T x} f(x)\,dx = e^{-j\omega^T\mu - \frac{1}{2}\omega^T\Sigma\omega}$$

$$\Phi_x = \underbrace{e^{-j\omega^T\mu - \frac{1}{2}\omega^T\Sigma\omega}}$$

3. Properties of MVG.

① linear transformation.

$$\left.\begin{array}{l} X \sim N(\mu,\Sigma) \\ Y = AX \end{array}\right. \Rightarrow Y \sim N(A\mu, A\Sigma A^T).$$

note $\Phi_y = E[e^{-j\omega^T y}] = E[e^{-j\omega^T Ax}] = e^{-j\omega^T A\mu - \frac{1}{2}\omega^T A\Sigma A^T \omega}$

linear transformation of Gaussian is Gaussian. $\sim N(A\mu, A\Sigma A^T)$

e.g. $\frac{1}{n}\sum_{i=1}^{n} x_i$ is Gaussian. where x_i is Gaussian

e.g. $Y = AX + b \Rightarrow Y \sim (A\mu + b, A\Sigma A^T)$

many things can be expressed as linear transformation

② Marginals.

$$X = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \quad \mu = \begin{bmatrix} \mu_1 \\ \mu_2 \end{bmatrix} \quad \Sigma = \begin{bmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{bmatrix}$$

then $X_1 \sim N(\mu_1, \Sigma_{11})$ $X_2 = N(\mu_2, \Sigma_{22})$

note $X_1 = [I, 0]\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = Ax$

③ Conditioning.

$$X_2|_{X_1=x_1} \sim N(\tilde{\mu}, \tilde{\Sigma}) \qquad \begin{array}{l} \tilde{\mu} = \mu_2 + \Sigma_{21}\Sigma_{11}^{-1}(x_1-\mu_1) \\ \tilde{\Sigma} = \Sigma_{22} - \Sigma_{21}\Sigma_{11}^{-1}\Sigma_{12} \end{array}$$

To prove this, we present a different question.

$X = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$. $X_1|_{X_2=x_2} \sim N(E(X_1|x_2=x_2), C_{x_1|x_2})$.

where $C_{x_1|x_2} = C_{x_1} - A C_{x_2x_1}$, $E(X_1|X_2=x_2) = A(x_2-\mu_2)+\mu_1$.

and A solves $AC_{x_2} = C_{x_1x_2}$

(a). $(x_1-\mu_1) - A(x_2-\mu_2)$ and X_2 are uncorrelated.

$E[((x_1-\mu_1) - A(x_2-\mu_2)) \cdot x_2^T] = C_{x_1x_2} - AC_{x_2} = 0$

(b).

$$\begin{bmatrix} (x_1-\mu_1)-A(x_2-\mu_2) \\ x_2 \end{bmatrix} = \begin{bmatrix} I & -A \\ 0 & I \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} A\mu_2-\mu_1 \\ 0 \end{bmatrix}$$

$\Rightarrow$ is Gaussian..

So $(x_1-\mu_1)-A(x_2-\mu_2)$ and x_2 are independent.

(c) $\Phi[e^{v'x_1} | x_2=x_2] = e^{j v' \underbrace{(\mu_1+A(x_2-\mu_2))}_{\mu'}] - v' \underbrace{C(x_1 - A(x_2|x_1))}_{\Sigma'} v/2}$.

solving μ', Σ' gives you the result.

Given x is Gaussian $\Rightarrow$ $\left\{ \begin{array}{l} \text{Marginal} \\ \text{conditional} \end{array} \right.$ are Gaussian.

what about the other direction?

④ Affine Transformation,

$\Sigma_{b|a} = \Sigma_{bb} - \Sigma_{ba}\Sigma_{aa}^{-1}\Sigma_{ab}$

Just a representation.

Given $p(x_a) = N(x_a; \mu_a, \Sigma_a)$

$p(x_b|x_a) = N(x_b; \underbrace{Mx_a+b}_{\text{const}}, \Sigma_{b|a})$.

$\Rightarrow p(x_a, x_b) = N\left(\begin{pmatrix} x_a \\ x_b \end{pmatrix}; \begin{pmatrix} \mu_a \\ M\mu_a+b \end{pmatrix}, R \right)$.

$$R = \begin{pmatrix} \Sigma_a & \Sigma_a M^T \\ M\Sigma_a & \Sigma_{b|a} + M\Sigma_a M^T \end{pmatrix}$$

$\Rightarrow p(x_b) = N(x_b; \mu_b, \Sigma_b)$ where $\mu_b = M\mu_a + b$, $\Sigma_b = \Sigma_{b|a} + M\Sigma_a M^T$

$\Rightarrow p(x_a|x_b) = N(x_a; \mu_{a|b}, \Sigma_{a|b})$

very complex but calculable

Summary: $P(x_a, x_b) \Rightarrow P(x_a) \quad P(x_a|x_b)$

$\left. \begin{array}{l} p(x_a) \\ p(x_b|x_a) \end{array} \right\} \Rightarrow P(x_a, x_b) \Rightarrow P(x_b), P(x_a|x_b)$.

Everything is a Gaussian! Affine preserve Gaussian

4. Conditional Probability.

conditional probability is also a probability. We can view it as a new probability.

e.g. $p(x_1|x_2, x_3)$ if we let $\tilde{p}(x)$ to denote $p(x|x_3)$.

then $p(x_1|x_2, x_3) = \tilde{p}(x_1|x_2) = \dfrac{\tilde{p}(x_2|x_1)\tilde{p}(x_1)}{\tilde{p}(x_2)} = \dfrac{p(x_2|x_1, x_3)\, p(x_1|x_3)}{p(x_2|x_3)}$.

5. A linear System Example.

① Given $x_{t+1} = (Ax_t + b_t) + w_t$ where b_t, d_t are known vector.

$y_t = (Cx_t + d_t) + e_t$ $\quad w_t \sim N(0, Q)$ $e_t \sim N(0, R)$.

we also know $x_1 \sim N(\hat{x}_{1|0}, P_{1|0})$.

Here we use a notation: $p(x_t | y_{1:t-1}) = N(x_t; \hat{x}_{t|t-1}, P_{t|t-1})$

when $t=1$ $N(\hat{x}_{1|0}, P_{1|0})$.

② we know x_1, we can observe y_t, we want to know x_t.

So the problem is $p(x_t|y_{1:t}) \to p(x_{t+1}|y_{1:t})$.

③ First of all everything is linear transform of Gaussian. Everything is Gaussian.

④. denote $p(x_t|y_{1:t-1}) = N(\hat{x}_{t|t-1}, P_{t|t-1})$ $\tilde{p}(x_t) = p(\cdot|y_{1:t-1})$

⑤ $p(y_t|x_t, y_{1:t-1})$. since $y_t = (Cx_t + d_t) + e_t$. when x_t is given

$y_t|_{x_t, y_{1:t-1}} \sim N(Cx_t + d_t, R)$. note $Cx_t + d_t$ is given e_t is random variable: $e_t + \text{const}$.

since x_t is observed now, so observing $y_{1:t-1}$ does not matter.

$\Rightarrow p(y_t|x_t, y_{1:t-1}) = N(Cx_t + d_t, R) = \tilde{p}(y_t|x_t)$

⑥ $\left.\begin{array}{l}\tilde{p}(x_t)\\ \tilde{p}(y_t|x_t)\end{array}\right\} \Rightarrow \tilde{p}(x_t|y_t)$ is known. $\Rightarrow p(x_t|y_{1:t})$ is known.

denote $\tilde{p}(x_t|y_t) = N(\hat{x}_{t|t}, P_{t|t})$.

then $\hat{x}_{t|t}$, $P_{t|t}$ can be derived by $\hat{x}_{t|t-1}$, $P_{t|t-1}$, C, d_t, R

⑦ denote $\tilde{p}(x_t|y_t) = \tilde{\tilde{p}}(x_t)$. $\tilde{\tilde{p}}(x_t) = \tilde{p}(\cdot|y_t) = p(\cdot|y_{1:t})$.

⑧ $p(x_{t+1}|x_t, y_{1:t})$ since x_t is directly observed. $y_{1:t}$ does not matter.

$= p(x_{t+1}|x_t) = N(Ax_t + b_t, Q) = \tilde{\tilde{p}}(x_{t+1}|x_t)$.

⑨ $\left.\begin{array}{l}\tilde{\tilde{p}}(x_t)\\ \tilde{\tilde{p}}(x_{t+1}|x_t)\end{array}\right\} \Rightarrow \tilde{\tilde{p}}(x_{t+1})$ is known $\Rightarrow p(x_{t+1}|y_{1:t})$ is known

Recall our Affine transform: $\hat{x}_{t+1|t} = A\hat{x}_{t|t} + b_t$

$P_{t+1|t} = AP_{t|t}A^T + Q$.

By knowing x_t, observe $y_{1:t}$. we can say something about x_{t+1}.

6. Canonical Form

① Gaussian is fully characterized by μ, Σ. $\to$ its first and second moment. This is moment form

② Canonical form: introduce information vector ξ and information matrix Ω.

$\xi \leftrightarrow \mu$

$\Omega \leftrightarrow \Sigma$.

$\begin{cases}\Omega = \Sigma^{-1}\\ \Sigma = \Omega^{-1}\end{cases}$ $\begin{cases}\xi = \Sigma^{-1}\mu\\ \mu = \Omega^{-1}\xi\end{cases}$

③ derive:

$p(x) = \det(2\pi\Sigma)^{-\frac{1}{2}} \exp(-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu))$

$= \eta \exp(-\frac{1}{2}x^T\Sigma^{-1}x + x^T\Sigma^{-1}\mu - \frac{1}{2}\mu^T\Sigma^{-1}\mu)$

$= \eta' \exp(-\frac{1}{2}x^T\Omega x + x^T\xi)$.

$\xi = \begin{bmatrix}\xi_\alpha\\ \xi_\beta\end{bmatrix}$ $\Lambda = \begin{bmatrix}\Lambda_{\alpha\alpha} & \Lambda_{\alpha\beta}\\ \Lambda_{\beta\alpha} & \Lambda_{\beta\beta}\end{bmatrix}$

④ Marginalization

$\begin{cases}\eta = \eta_\alpha - \Lambda_{\alpha\beta}\Lambda_{\beta\beta}^{-1}\eta_\beta\\ \Lambda = \Lambda_{\alpha\alpha} - \Lambda_{\alpha\beta}\Lambda_{\beta\beta}^{-1}\Lambda_{\beta\alpha}.\end{cases}$

Conditioning: $\eta' = \eta_\alpha - \Lambda_{\alpha\beta}\beta$

$\Lambda' = \Lambda_{\alpha\alpha}$

Moment form: Margin is easy. Condition is hard.

Canonical Form: Margin is hard Condition is easy.

You can always transform your algorithm in information space and see what happens.

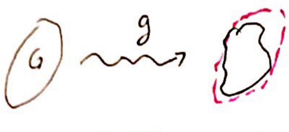
Now a linear transform of Gaussian is still Gaussian.

What about non linear transform? Of course not Gaussian.

So Question comes, How to "preserve" Gaussian for non-linear transform?

7. Uncent Transform.

① Idea: we may not get a Gaussian after the transform. But we can use Gaussian to approximate the transformed distribution.



② How to do this approximation?

<1> sample points on G.

<2> Compute transformed points under g.

<3>. Estimate Gaussian using the transformed points

③ Issues:

<1> What is a good sampling of G? Cannot be too much points but must capture important details

<2> How to sample? What's the algorithm?

④ Criteria: A good sampling.

suppose we have the sampling algorithm. It gives points $\chi^{[i]}$, $w^{[i]}$ as weight for point $\chi^{[i]}$. A good sampling would be:

$$\sum_i w^{[i]} = 1$$
$$\mu = \sum_i w^{[i]} \chi^{[i]}$$
$$\Sigma = \sum_i w^{[i]} (\chi^{[i]} - \mu)(\chi^{[i]} - \mu)^T$$

weights should sum to 1 and these samples should be able to reconstruct μ and Σ of G before they transform.

(They capture essence of Gaussian. μ. Σ)

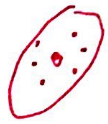
⑤ The Uncent transform is all about How to do this sampling.

<1> points: $\chi^{[0]} = \mu$. $\chi^{[i]} = \mu + (\sqrt{(n+\lambda)\Sigma})_i$ $i = 1 \cdots n$

$\chi^{[i]} = \mu - (\sqrt{(n+\lambda)\Sigma})_i$ $i = n+1 \cdots 2n$.

sample $2n+1$ points. First sample center μ. then sample symmetrically.

center is where most points are.



what is $(\sqrt{(n+\lambda)\Sigma})_i$? i means the i^{th} column.

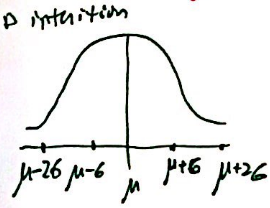
(i) $n+\lambda$ is a scaling factor.

(ii) $\sqrt{\Sigma}$ is square of a matrix

General square $\Sigma = UDU^{-1} = U\begin{pmatrix}\sqrt{d_{11}} & & \\ & \ddots & \\ & & \sqrt{d_{nn}}\end{pmatrix}\begin{pmatrix}\sqrt{d_{11}} & & \\ & \ddots & \\ & & \sqrt{d_{nn}}\end{pmatrix}U^{T}$

Cholesky decomposition $\Sigma = LL^T$. $\sqrt{\Sigma} = L$.

1D intuition



we center around μ. then $\mu \pm \sigma$. then scale factor. before σ.

<2> Weights.

$w_m^{[0]} = \frac{\lambda}{n+\lambda}$ $\quad w_c^{[0]} = w_m^{[0]} + (1 - \alpha^2 + \beta)$ weight for μ

$w_m^{[i]} = w_c^{[i]} = \frac{1}{2(n+\lambda)}$ $\quad i = 1 \cdots 2n$. weights for other points

α, β are params one can choose. they control certain things. plot to visualize!

one can plug back and see this satisfies our criteria.

<3> Reconstructed Gaussian (Statistical Estimation)

$$\mu' = \sum_{i=0}^{2n} w_m^{[i]} g(\chi^{[i]})$$
$$\Sigma' = \sum_{i=0}^{2n} w_c^{[i]} (g(\chi^{[i]}) - \mu')(g(\chi^{[i]}) - \mu')^T$$