COURSERA

ROBOTICS SPECIALIZATION

ROBOTICS: ESTIMATION AND LEARNING

Notes are taken by Jelena Kocić

April 2019

Contents

1. Gaussian Model Learning

1.1. Introduction

Sources of uncertainty in robotics:

- Sensor noise,
- Lack of knowledge about the world,
- Dynamic changes in motion and environment.

Methods for dealing with uncertainty:

- Probabilistic modeling,
- Machine learning.

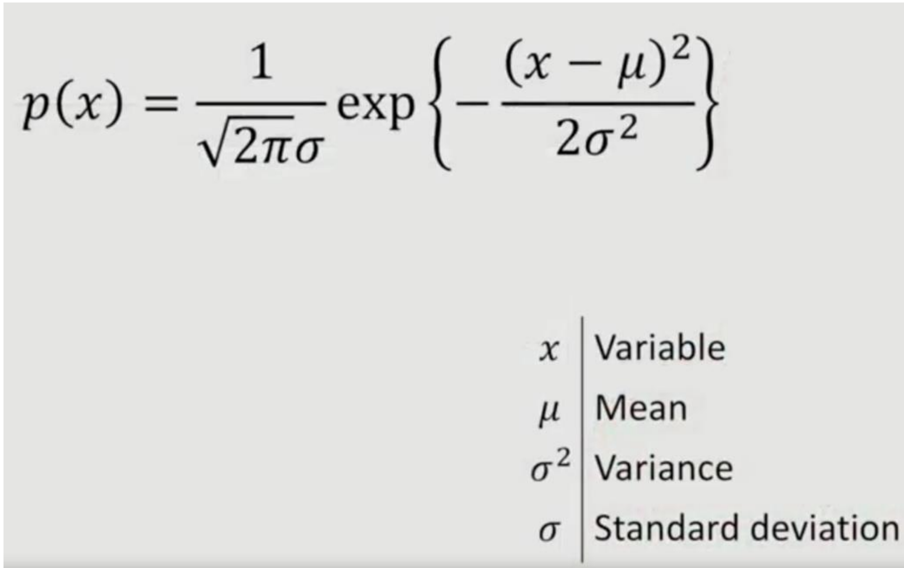
1.2. Single Dimensional Gaussian

1.2.1. 1.2.1 Gaussian Distribution

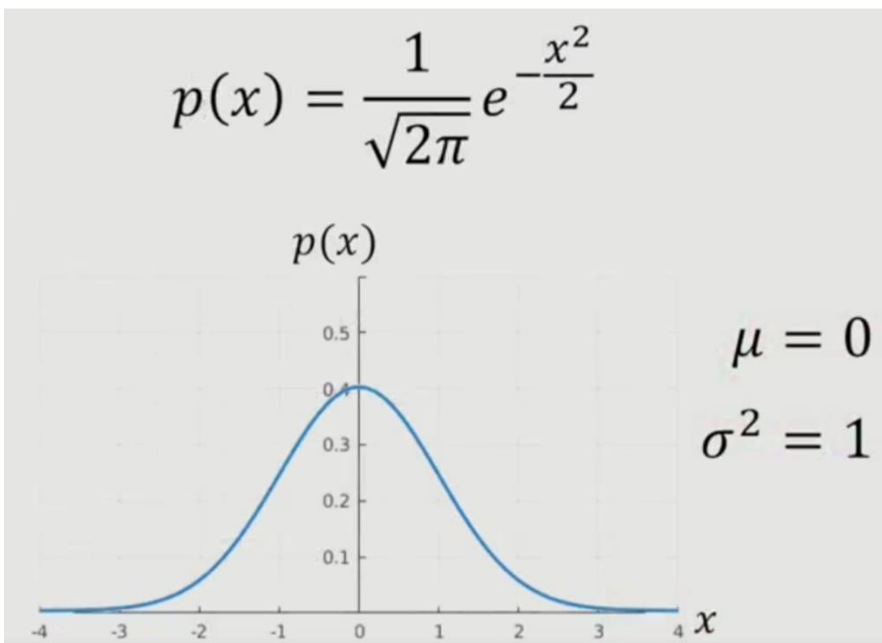
Gaussian distribution is a widely used continuous probabilistic representation, and it provides a useful way to estimate uncertainty in the world.

Why Gaussian distribution?

- only two parameters are needed to specify the Gaussian (mean and variance),
- mathematically the distribution has good properties (e.g. product of Gaussian distributions forms another Gaussian),
- the central limit theorem tells us that the expectation of the mean of any random variable converges to the Gaussian distribution.

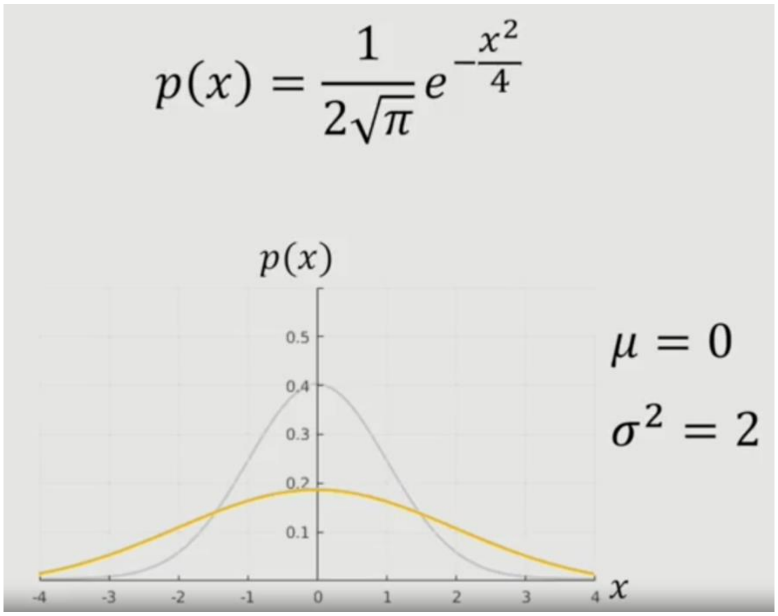


Standard Normal Distribution is when μ = 0 and σ^2 = 1.



The mean value determines the center of the distribution. We can also say that the peak location of the distribution changes.

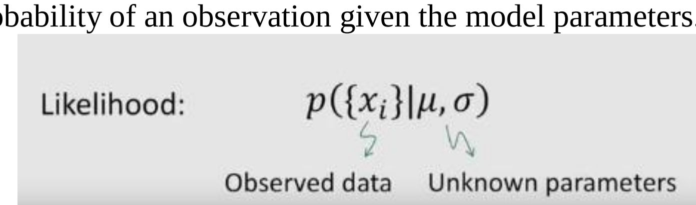
The variance changes the spread of the distribution. If the variance increases, the curve spreads out as compared to the standard Gaussian curve. Also, the peak value decreases so that the integral is still 1, which fulfills the properties of a probability density function. Conversely, a smaller variance tightens the curve and the peak value becomes bigger as well. So that the integral remains 1.



1.2.2. 1.2.2. Maximum Likelihood Estimate of Gaussian Model Parameters

Goal is to estimate the mean and the variance given observed data.

Likelihood is the probability of an observation given the model parameters.



Quiz:

Complete the paragraph using a correct match of terms.

Likelihood is defined as the [A] of the [B] given [C] values. It can be viewed as a function of [C] given [B]. By maximum likelihood estimation of [C], we mean a process to compute the [C] that are most likely to give the [B].

- ○ A. model parameter(s), B. probability, C. observed data
- ○ A. observed data, B. probability, C. model parameter(s)
- ◉ A. probability, B. observed data, C. model parameter(s)

Correct

Objective is to estimate the mean and the variance given observed data:

$$\hat{\mu}, \hat{\sigma} = \arg\max_{\mu,\sigma} p(\{x_i\}|\mu,\sigma)$$

Assuming independence of observations,

$$p(\{x_i\}|\mu,\sigma) = \prod_{i=1}^{N} p(x_i|\mu,\sigma)$$

$$\hat{\mu}, \hat{\sigma} = \arg\max_{\mu,\sigma} \prod_{i=1}^{N} p(x_i|\mu,\sigma)$$

It will turn out that maximizing log likelihood is simpler in many cases.

=>

$$(1)\quad \arg\max_{\mu,\sigma} \prod_{i=1}^{N} p(x_i|\mu,\sigma) = \arg\max_{\mu,\sigma} \ln\left\{\prod_{i=1}^{N} p(x_i|\mu,\sigma)\right\}$$

$$= \arg\max_{\mu,\sigma} \boxed{\sum_{i=1}^{N} \ln p(x_i|\mu,\sigma)}$$

NOTE 2:

$$\log(x_1 \times x_2 \times \cdots \times x_k) = \log(x_1) + \log(x_2) + \cdots + \log(x_k)$$

$$\hat{\mu}, \hat{\sigma} = \arg\max_{\mu,\sigma} \sum_{i=1}^{N} \ln p(x_i|\mu,\sigma)$$

(2) Gaussian! $\frac{1}{\sqrt{2\pi}\sigma} \exp\left\{-\frac{(x_i-\mu)^2}{2\sigma^2}\right\}$

$$\hat{\mu}, \hat{\sigma} = \arg\max_{\mu,\sigma} \sum_{i=1}^{N} \left\{-\frac{(x_i-\mu)^2}{2\sigma^2} - \ln\sigma\right\}$$

, where the constant is ignored.

Changing into minimization problem:

$$\hat{\mu}, \hat{\sigma} = \arg\min_{\mu,\sigma} \sum_{i=1}^{N} \left\{\frac{(x_i-\mu)^2}{2\sigma^2} + \ln\sigma\right\}$$

$$\hat{\mu}, \hat{\sigma} = \arg\min_{\mu,\sigma} \underbrace{\sum_{i=1}^{N} \left\{ \frac{(x_i - \mu)^2}{2\sigma^2} + \ln \sigma \right\}}_{J(\mu,\sigma)}$$

- At optimum,

$$\boxed{\frac{\partial J}{\partial \mu} = 0 \longrightarrow \hat{\mu}} \quad \boxed{\frac{\partial J(\hat{\mu}, \sigma)}{\partial \sigma} = 0 \rightarrow \hat{\sigma}}$$

If we apply the optimality condition for convex optimization, the first order derivative of J with respect to mu should be zero. From this, we can compute the maximum likelihood estimate of mu and we are going to write it as mu hat.

We apply the same optimality condition to compute the estimate of sigma. For this, we can use the value of mu hat in place of mu as a parameter.

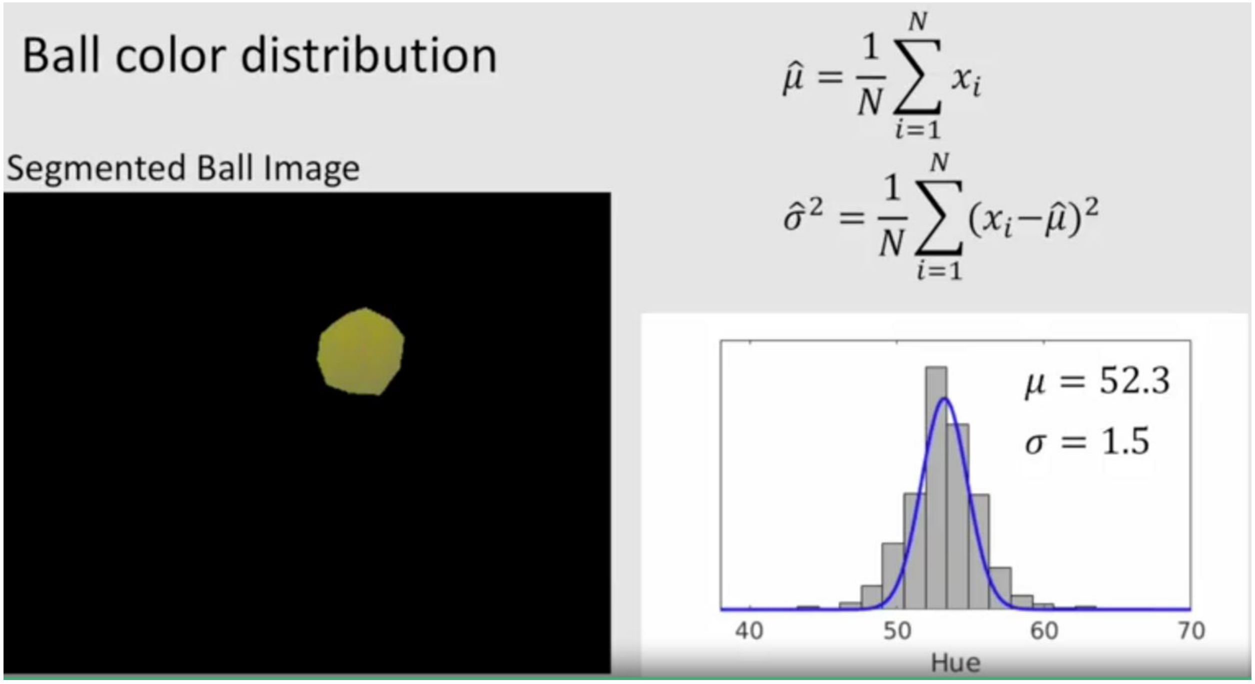
The MLE solution:

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

$$\hat{\sigma}^2 = \frac{1}{N} \sum_{i=1}^{N} (x_i - \hat{\mu})^2$$

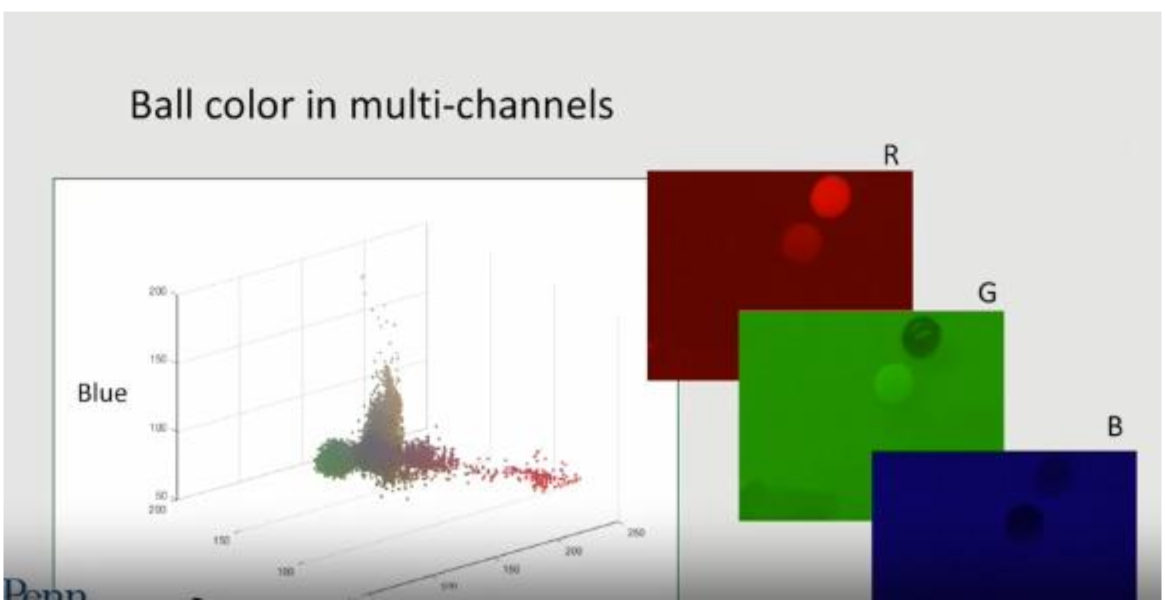
Mu hat is exactly the sample mean the average of the data, which is a natural estimate of data. Also, sigma hat square is just a sample variance.

An example:



1.3. Multivariate Gaussian

1.3.1. Multivariate Gaussian Distribution



Mathematically, the multivariate Gaussian is expressed as an exponential coupled with a scalar vector.

$$p(\mathbf{x}) = \frac{1}{(2\pi)^{D/2}|\Sigma|^{1/2}} \exp\left\{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T \Sigma^{-1} (\mathbf{x}-\boldsymbol{\mu})\right\}$$

D Number of Dimensions

$\mathbf{x}$ Variable

$\boldsymbol{\mu}$ Mean *vector*

Σ Covariance *matrix*

(Dimension = 1)

$$p(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left\{-\frac{(x-\mu)^2}{2\sigma^2}\right\}$$

D is the number of dimensions we are going to use. X is the vector of variables whose probability we are attempting to quantify. To signify that x is now a vector, we make x bold.

In contrast to the 1D case mu our mean is now a vector and sigma, our covariance matrix is a square matrix.

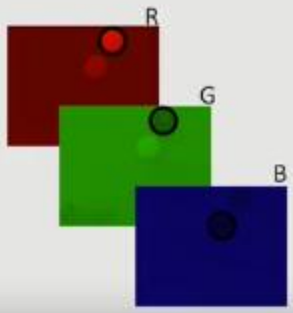
In our covariance matrix, there are two key components: diagonal terms: variance and off-diagonal terms: correlation.

(Dimension = 2)

$$\Sigma = \begin{bmatrix} \sigma_{x_1}^2 & \sigma_{x_1}\sigma_{x_2} \\ \sigma_{x_2}\sigma_{x_1} & \sigma_{x_2}^2 \end{bmatrix} \quad (\sigma_{x_1}\sigma_{x_2} = \sigma_{x_2}\sigma_{x_1})$$

$$p(\mathbf{x}) = \frac{1}{(2\pi)^{D/2}|\Sigma|^{1/2}} \exp\left\{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T\Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu})\right\}$$

Ball color in multi-channels



$$D = 3$$
$$\mathbf{x} = [x_R \quad x_G \quad x_B]$$

$$D = 3$$
$$\mathbf{x} = [x_R \quad x_G \quad x_B]$$

$$\boldsymbol{\mu} = [\mu_R \quad \mu_G \quad \mu_B]$$

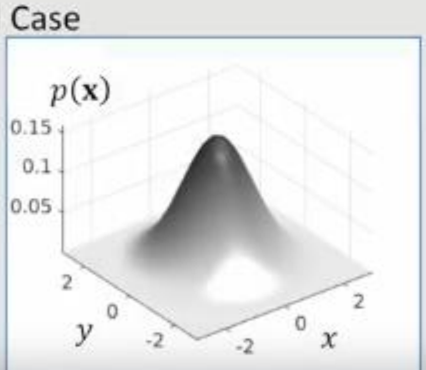
$$\Sigma = \begin{bmatrix} \sigma_{x_R}^2 & \sigma_{x_R}\sigma_{x_G} & \sigma_{x_R}\sigma_{x_B} \\ \sigma_{x_R}\sigma_{x_G} & \sigma_{x_G}^2 & \sigma_{x_G}\sigma_{x_B} \\ \sigma_{x_R}\sigma_{x_B} & \sigma_{x_G}\sigma_{x_B} & \sigma_{x_B}^2 \end{bmatrix}$$

Special case:

$$p(\mathbf{x}) = \frac{1}{2\pi} \exp\left\{-\frac{x^2+y^2}{2}\right\}$$

- 2D Zero-mean Spherical Case

$$D = 2$$
$$\mathbf{x} = [x \quad y]^T$$
$$\boldsymbol{\mu} = [0 \quad 0]^T$$
$$\Sigma = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

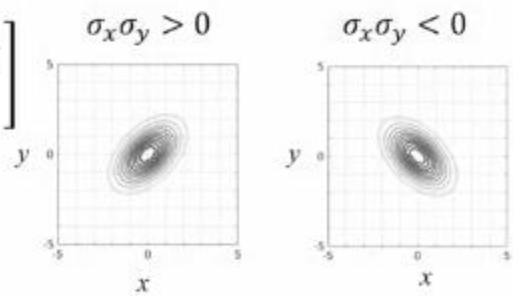


General case:

$$p(\mathbf{x}) = \frac{1}{(2\pi)^{D/2}|\Sigma|^{1/2}} \exp\left\{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T\Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu})\right\}$$

- 2D General Case

$$\Sigma = \begin{bmatrix} \sigma_x^2 & \sigma_x\sigma_y \\ \sigma_x\sigma_y & \sigma_y^2 \end{bmatrix}$$



There are two other properties I'm going to mention about sigma.

First, the covariance metrics must remain symmetric and positive definite. Meaning, the elements of sigma are symmetric about its diagonal line. The Eigenvalues of sigma must be positive.

Second, even when the covariance matrix has none zero correlation terms we can always find the coordinate transformation which makes the shape appear symmetric.

1.3.2. MLE of Multivariate Gaussian

Objective: Estimate the mean and the covariance matrix given observed data.

Likelihood: $p(\{\mathbf{x}_i\}|\boldsymbol{\mu},\Sigma)$

Observed data Unknown parameters

- Objective

$$\hat{\boldsymbol{\mu}}, \hat{\Sigma} = \arg\max_{\boldsymbol{\mu},\Sigma} p(\{\mathbf{x}_i\}|\boldsymbol{\mu},\Sigma)$$

$$\hat{\boldsymbol{\mu}}, \hat{\Sigma} = \arg\max_{\boldsymbol{\mu},\Sigma} \prod_{i=1}^{N} p(\mathbf{x}_i|\boldsymbol{\mu}, \Sigma)$$

(1) Take the log!

$$\arg\max \textit{likelihood} \leftrightarrow \arg\max \ln(\textit{likelihood})$$

$$\log(x_1 \times x_2 \times \cdots \times x_k) = \log(x_1) + \log(x_1) + \cdots + \log(x_k)$$

$$\boxed{\arg\max_{\boldsymbol{\mu},\Sigma} \prod_{i=1}^{N} p(\mathbf{x}_i|\boldsymbol{\mu}, \Sigma) \Rightarrow \arg\max_{\boldsymbol{\mu},\Sigma} \sum_{i=1}^{N} \ln p(\mathbf{x}_i|\boldsymbol{\mu}, \Sigma)}$$

The statement
$log(x_1 * x_2 * \ldots * x_k) = log(x_1) + log(x_1) + \ldots + log(x_k)$
should be
$log(x_1 * x_2 * \ldots * x_k) = log(x_1) + log(x_2) + \ldots + log(x_k)$

(2) Gaussian!

$$p(\mathbf{x}) = \frac{1}{(2\pi)^{D/2}|\Sigma|^{1/2}} \exp\left\{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T \Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu})\right\}$$

$$\hat{\boldsymbol{\mu}}, \hat{\Sigma} = \arg\max_{\boldsymbol{\mu},\Sigma} \sum_{i=1}^{N} \left\{-\frac{1}{2}(\mathbf{x}_i-\boldsymbol{\mu})^T \Sigma^{-1}(\mathbf{x}_i-\boldsymbol{\mu}) - \frac{1}{2}\ln|\Sigma| + c\right\}$$

$$\Updownarrow$$

$$\boxed{\hat{\boldsymbol{\mu}}, \hat{\Sigma} = \arg\min_{\boldsymbol{\mu},\Sigma} \sum_{i=1}^{N} \left\{\frac{1}{2}(\mathbf{x}_i-\boldsymbol{\mu})^T \Sigma^{-1}(\mathbf{x}_i-\boldsymbol{\mu}) + \frac{1}{2}\ln|\Sigma|\right\}}$$

$$\hat{\boldsymbol{\mu}}, \hat{\Sigma} = \arg\min_{\boldsymbol{\mu},\Sigma} \underbrace{\sum_{i=1}^{N} \left\{ \frac{1}{2} (\mathbf{x}_i - \boldsymbol{\mu})^T \Sigma^{-1} (\mathbf{x}_i - \boldsymbol{\mu}) + \frac{1}{2} \ln|\Sigma| \right\}}_{J(\boldsymbol{\mu}, \Sigma)}$$

- At optimum,

① $\dfrac{\partial J}{\partial \boldsymbol{\mu}} = \mathbf{0} \longrightarrow \hat{\boldsymbol{\mu}}$ ② $\dfrac{\partial J(\hat{\boldsymbol{\mu}}, \Sigma)}{\partial \Sigma} = 0 \longrightarrow \hat{\Sigma}$

- In summary, we have

$$\hat{\boldsymbol{\mu}} = \frac{1}{N} \sum_{i=1}^{N} \mathbf{x}_i$$

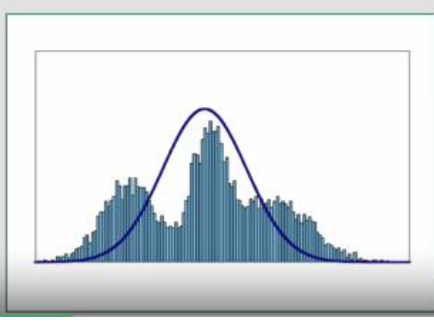
$$\hat{\Sigma} = \frac{1}{N} \sum_{i=1}^{N} (\mathbf{x}_i - \hat{\boldsymbol{\mu}})(\mathbf{x}_i - \hat{\boldsymbol{\mu}})^{\top}$$

1.4. Mixture of Gaussian

1.4.1. Gaussian Mixture Model (GMM)

Limitations of Single Gaussian

- Single Mode
- Symmetric



A single Gaussian cannot properly model a distribution if it has multiple modes or there is a lack of symmetry.

GMM is the sum of Gaussians.

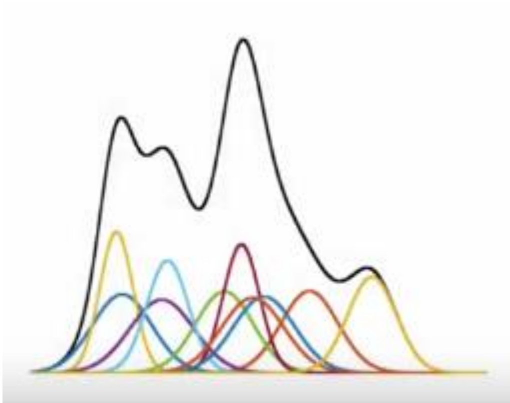
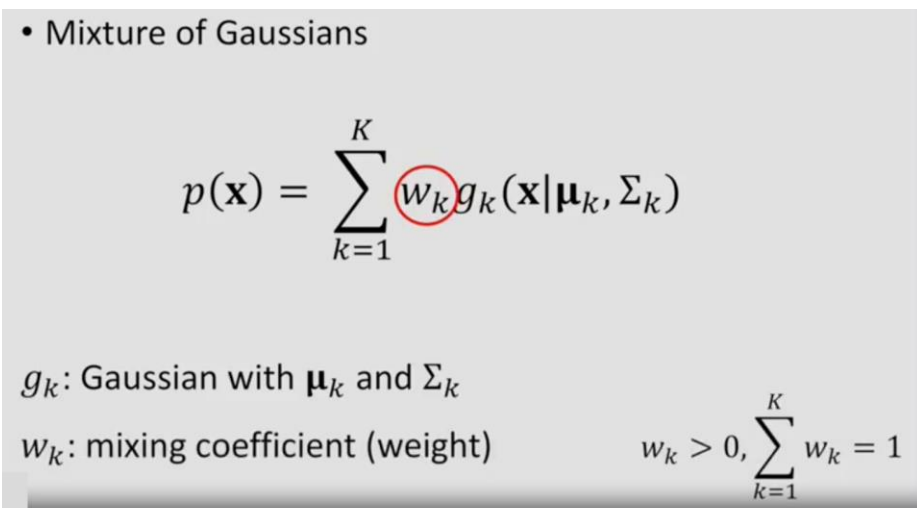
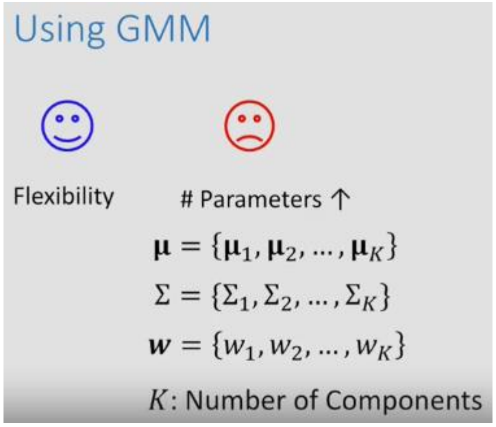


Figure. The colorful lines are ten random Gaussian curves. The black line is the sum of all the Gaussians.

If we choose the right Gaussian elements, than we can express any unusual distribution.

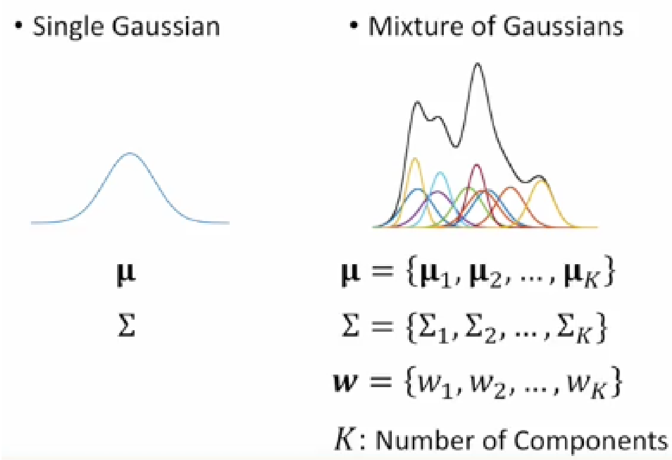


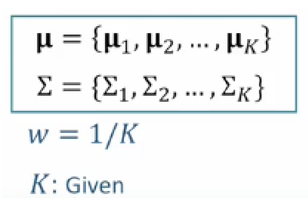
The weights, Ws should be all positive and they must sum to 1. This make sure that the distribution of GMM is a probability density that integrals to 1.



Bigger number of parameters. –> No analytic solution. Overfitting.

1.4.2. GMM Parameter Estimation via EM





Learning GMM Parameters

Likelihood: $p(\{\mathbf{x}_i\}|\boldsymbol{\mu}, \Sigma)$

Observed data   Unknown parameters

$$\hat{\boldsymbol{\mu}}, \hat{\Sigma} = \arg\max_{\boldsymbol{\mu},\Sigma} p(\{\mathbf{x}_i\}|\boldsymbol{\mu}, \Sigma)$$

$$\boldsymbol{\mu} = \{\boldsymbol{\mu}_k\}$$

$$\Sigma = \{\Sigma_k\} \quad k = 1,2, \dots, K$$

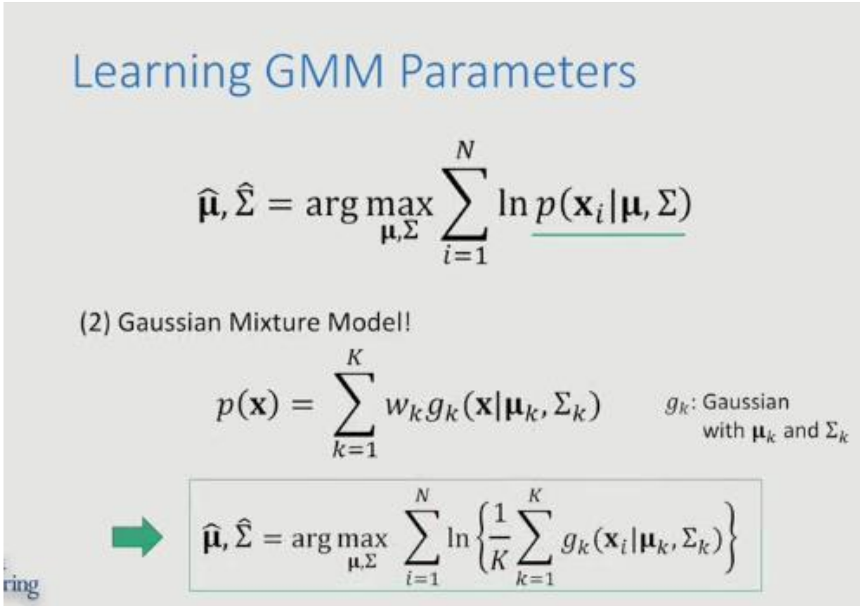
$$\hat{\boldsymbol{\mu}}, \hat{\Sigma} = \arg\max_{\boldsymbol{\mu},\Sigma} \prod_{i=1}^{N} p(\mathbf{x}_i|\boldsymbol{\mu}, \Sigma)$$

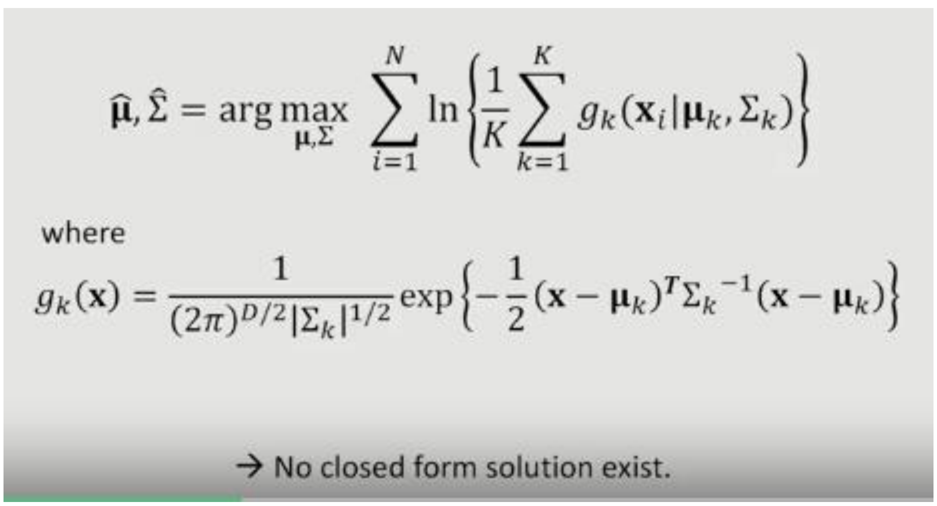
(1) Take the log!

$$\arg\max likelihood \leftrightarrow \arg\max \ln(likelihood)$$

$$\log(x_1 \times x_2 \times \cdots \times x_k) = \log(x_1) + \log(x_1) + \cdots + \log(x_k)$$

$$\arg\max_{\boldsymbol{\mu},\Sigma} \prod_{i=1}^{N} p(\mathbf{x}_i|\boldsymbol{\mu}, \Sigma) \Rightarrow \arg\max_{\boldsymbol{\mu},\Sigma} \sum_{i=1}^{N} \ln p(\mathbf{x}_i|\boldsymbol{\mu}, \Sigma)$$





But when we apply the specific probability model of the GMM into the equation, which is a sum of Gaussians, we have this. It turns out that we cannot further simplify this formula analytically, because there appears a summation of Gaussians inside the log function. This implies we can estimate the parameters only via iterative computations.

EM for GMM:

1. Special case: EM for GMM Parameter Estimation
2. General EM Algorithm

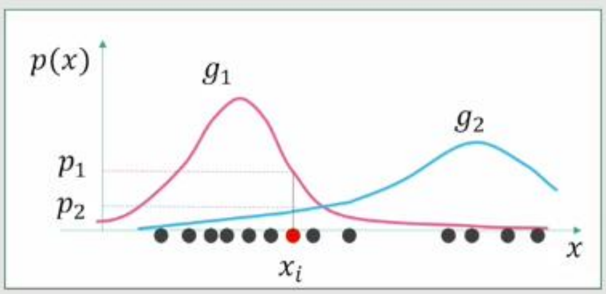
Initial $\boldsymbol{\mu}$ and Σ

Latent variable z

- Latent Variable

$$z_k^i = \frac{g_k(\mathbf{x}_i|\boldsymbol{\mu}_k, \Sigma_k)}{\sum_{k=1}^{K} g_k(\mathbf{x}_i|\boldsymbol{\mu}_k, \Sigma_k)}$$

$$z_k^i = \frac{g_k(\mathbf{x}_i|\boldsymbol{\mu}_k, \Sigma_k)}{g_1(\mathbf{x}_i|\boldsymbol{\mu}_1, \Sigma_1) + g_2(\mathbf{x}_i|\boldsymbol{\mu}_2, \Sigma_2)}$$



$$z_1^i = \frac{p_1}{p_1 + p_2}$$

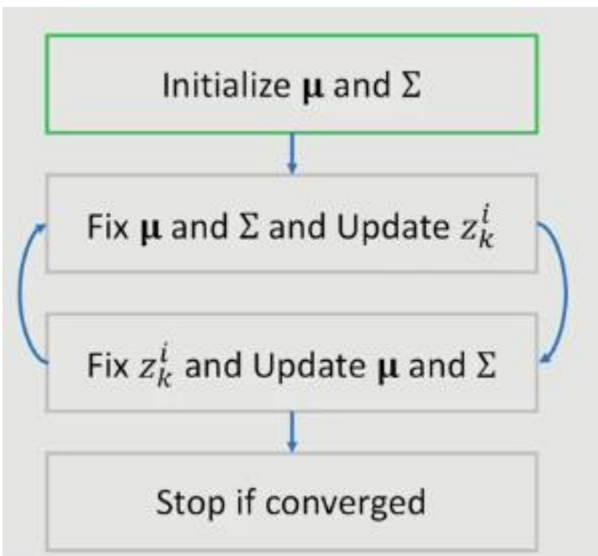
$$z_2^i = \frac{p_2}{p_1 + p_2}$$

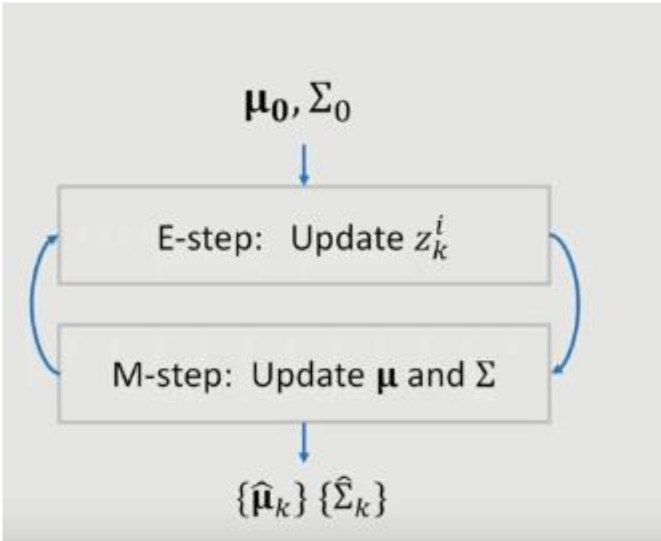
- Mean and Covariance Matrix

$$\hat{\boldsymbol{\mu}}_k = \frac{1}{z_k}\sum_{i=1}^{N} z_k^i \mathbf{x}_i$$

$$\hat{\Sigma}_k = \frac{1}{z_k}\sum_{i=1}^{N} z_k^i (\mathbf{x}_i - \hat{\boldsymbol{\mu}}_k)(\mathbf{x}_i - \hat{\boldsymbol{\mu}}_k)^\top$$

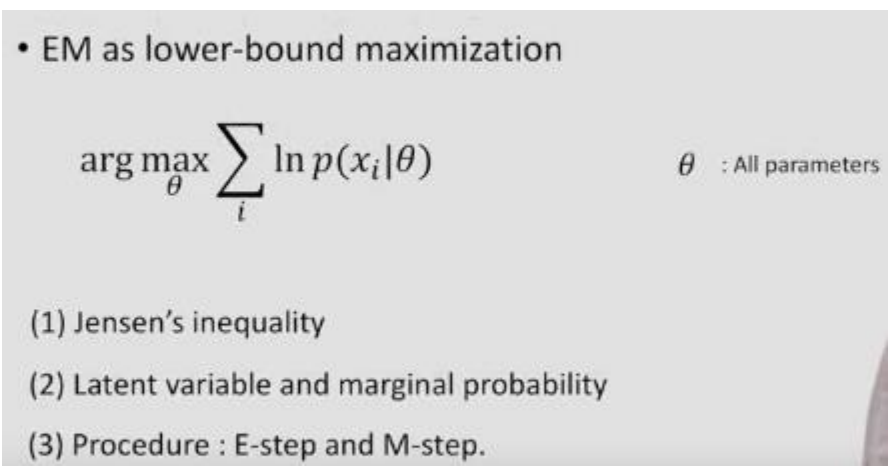
$$z_k = \sum\nolimits_{i=1}^{N} z_k^i$$

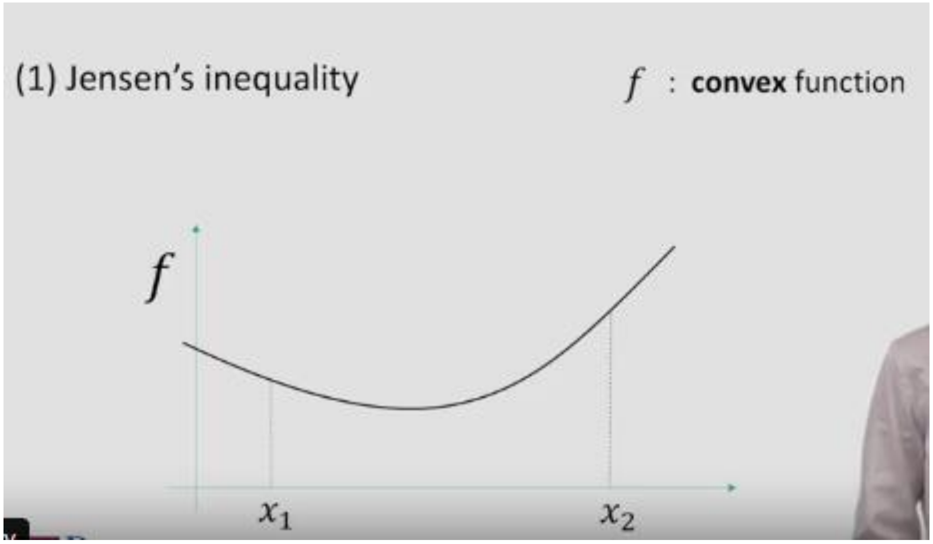


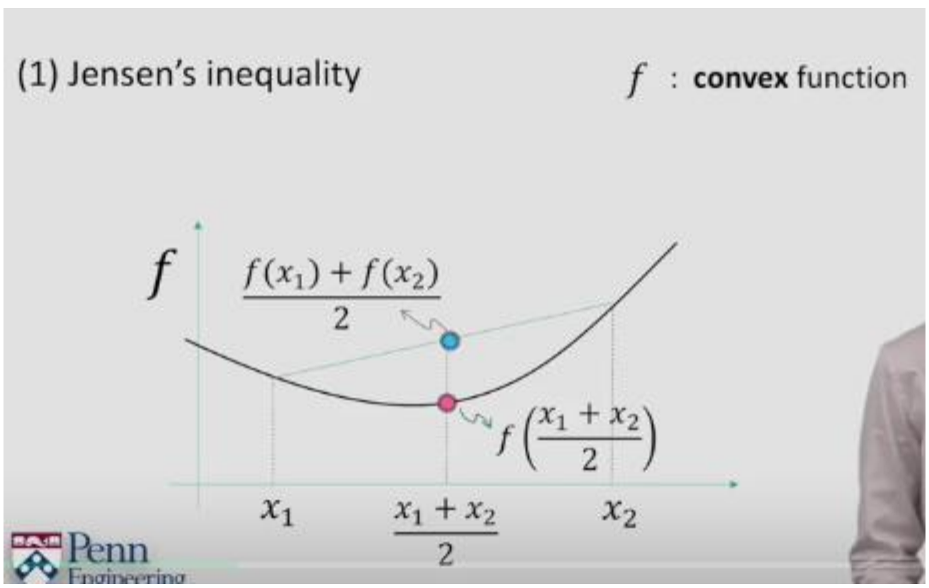


1.4.3. Expectation-Maximization (EM)

Let consider the EM Algorithm as a maximization of a lower bound of an object function.




(1) Jensen's inequality
f : **convex** function
f
x_1
x_2


(1) Jensen's inequality
f : **convex** function
f
$\frac{f(x_1)+f(x_2)}{2}$
$f\left(\frac{x_1+x_2}{2}\right)$
x_1
$\frac{x_1+x_2}{2}$
x_2
Penn

(1) Jensen’s inequality f : **convex** function

$$f(a_1 x_1 + a_2 x_2) \le a_1 f(x_1) + a_2 f(x_2)$$

$$a_1 + a_2 = 1$$
$$a_1 \ge 0$$
$$a_2 \ge 0$$

(1) Jensen’s inequality f : **convex** function

$$f\left(\sum a_i x_i\right) \le \sum a_i f(x_i)$$

$$\sum a_i = 1$$
$$a_i \ge 0$$

(1) Jensen's inequality

$\ln$ is *concave*

$$\ln\left(\sum a_i p_i\right) \geq \sum a_i \ln p_i$$

$$\sum a_i = 1$$

$$a_i \geq 0$$

(2) latent variable z

$$p(X|\theta) = \sum_Z p(X, Z|\theta)$$

(From definition of marginal probability)

(2) latent variable

$$\ln p(X|\theta) = \ln \sum_Z p(X, Z|\theta) \quad \boxed{\text{Log-likelihood}}$$

$$= \ln \sum_Z q(Z) \frac{p(X, Z|\theta)}{q(Z)} \geq \sum_Z q(Z) \ln \frac{p(X, Z|\theta)}{q(Z)}$$

$\boxed{\text{Lower bound}}$

q(Z) is a valid probability distribution over Z

(2) latent variable

$$\ln p(X|\theta) = \ln \sum_Z p(X,Z|\theta) \geq \sum_Z q(Z) \ln \frac{p(X,Z|\theta)}{q(Z)}$$

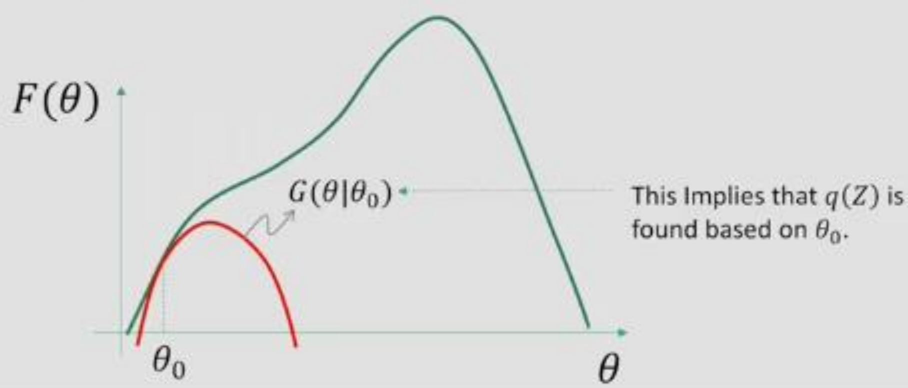
Log-likelihood

Lower bound

Find q!

(3a) Find a lower bound G with an initial guess

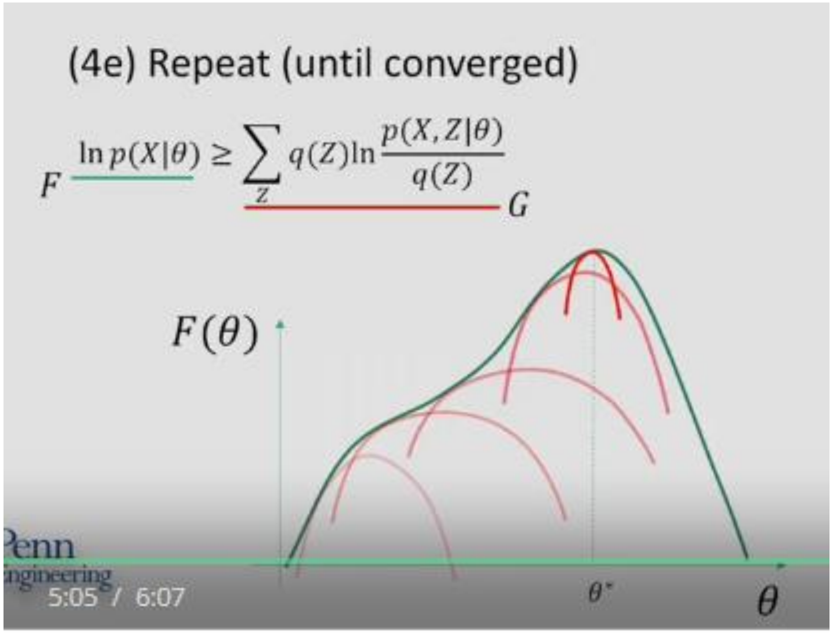
$$\underbrace{\ln p(X|\theta)}_{F} \geq \underbrace{\sum_Z q(Z) \ln \frac{p(X,Z|\theta)}{q(Z)}}_{G}$$

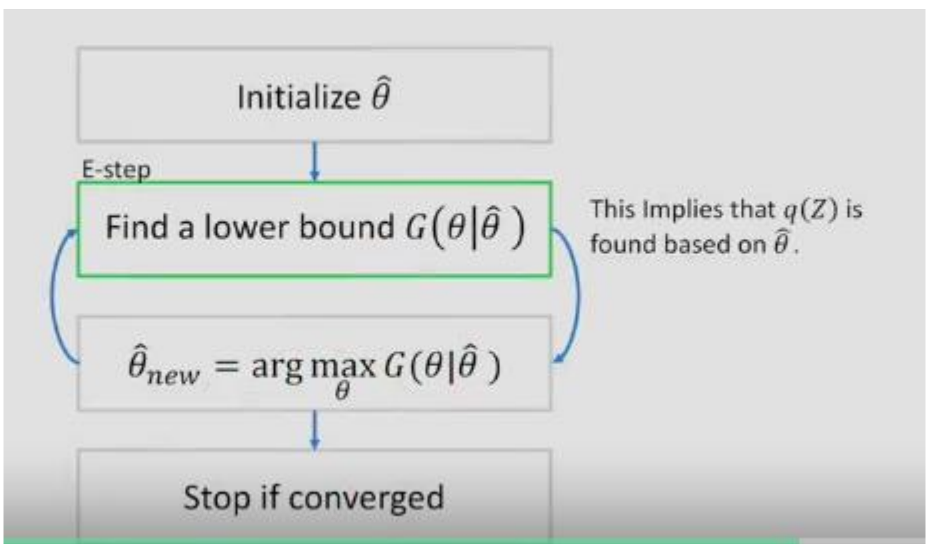


(3b) Find $\theta^* = \arg\max_\theta G(\theta|\theta_0)$

(3c) Find a new lower bound G with $\theta_1 \leftarrow \theta^*$

(3d) Find $\theta^* = \arg\max_\theta G(\theta|\theta_1)$



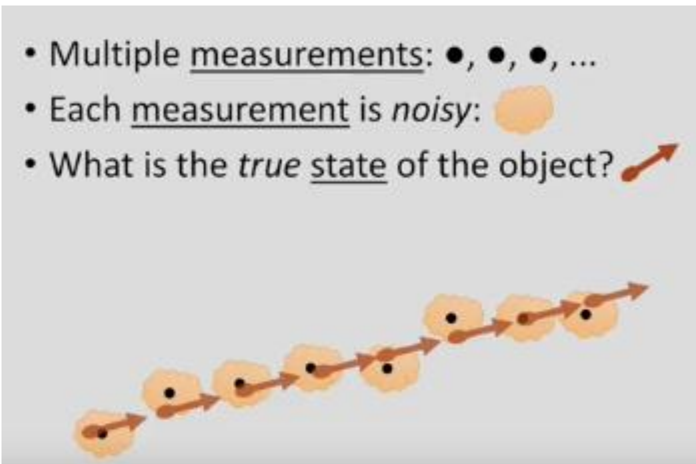


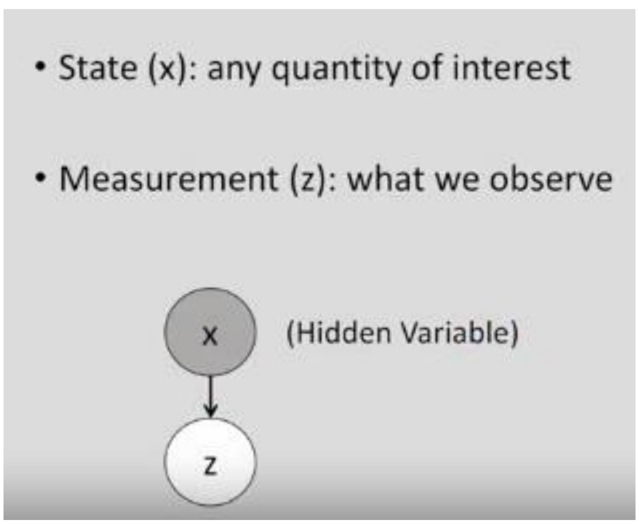
PROGRAMMING ASSIGNMENT:
Color Learning and Target Detection

2. Bayesian Estimation – Target Tacking

2.1. Kalman Filter Motivation

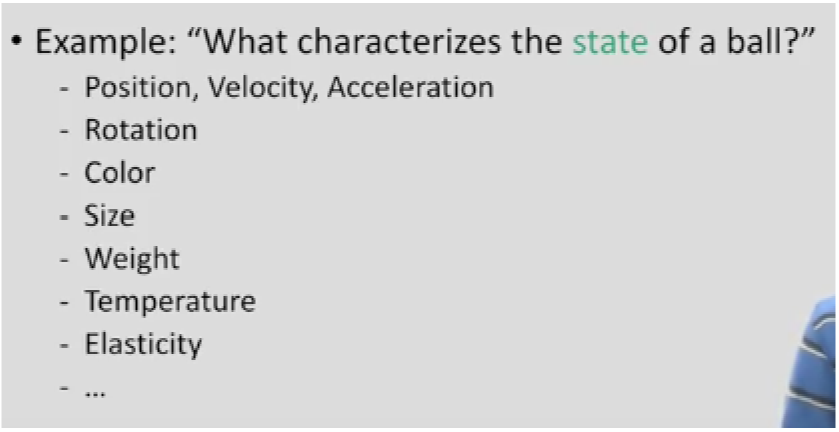
The Kalman Filter is an optimal tracking algorithm for linear systems that is widely used in many applications. Examples of tracking includes pedestrian and vehicle tracking for self-driving cars or items traveling along a conveyor belt on an assembly line.

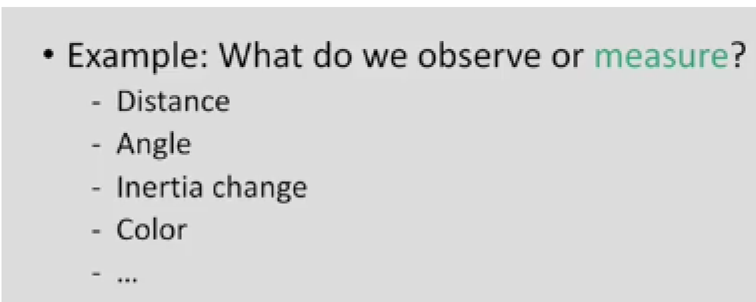




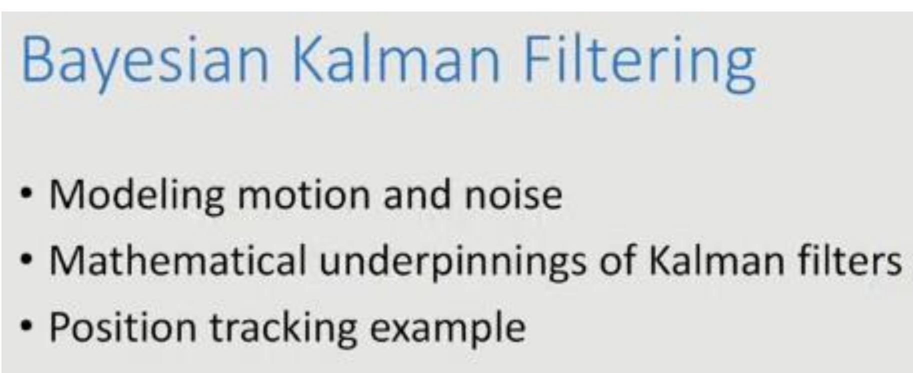
The concepts of measurements should be disambiguated from the concepts of state.

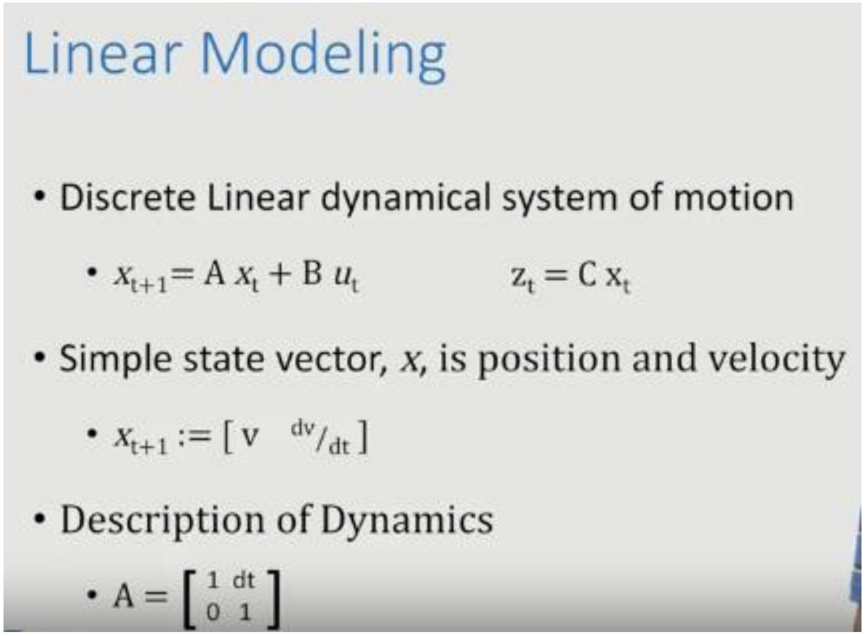
There's <u>a true state of the world</u>, but the robots can only observe a shadow of that world. For instance, the true position of the soccer ball may be 11 meters away from the robot, but the robot thinks that it is 11.32567 meters away. The robot observes this position of the ball through its camera, and this measurement through the camera gives a noisy estimate of the state. One source of the noise is the collection of pixels that can be misclassified between the ball and the surrounding area. We saw this kind of noise in module one.





2.2. Kalman Filter Model – System and Measurement Models





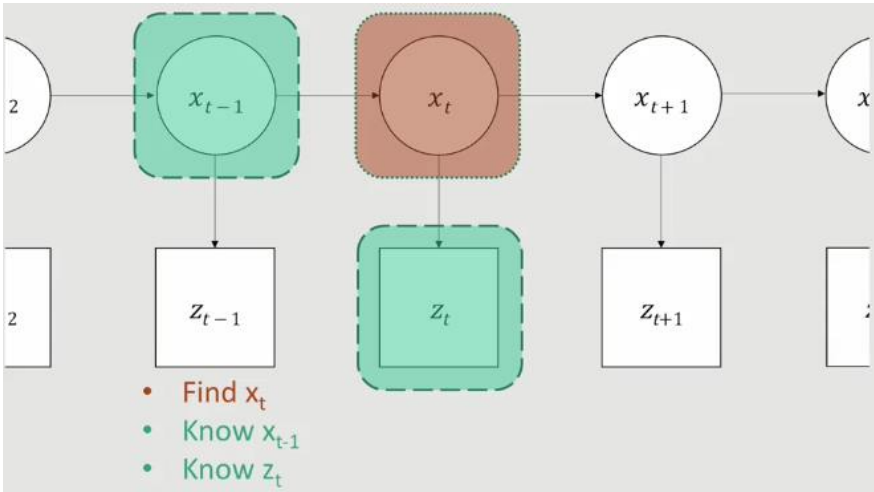
In a simple example, the state, x, will be indexed by time steps, t. The state will be comprised of position v, in meters, and velocity dvdt measured in meters per second. Due to dynamics, the state changes in each time step. Going from the current time step t to the next time step t+1. This change is captured by A, the state transition matrix. Sometimes notated as pi. The state transition matrix combines state information to describe the state at the next step in time.

The transition simplifies the current state to depend only on the previous state making our mathematical lives easier.

With the state being in position in velocity, we know that the position must change in time based on the velocity. The state transition matrix captures this with the given formulation.

The robot will not directly measure X unfortunately, but the robot may observe portions of x through it's sensors. This portion is labeled z, where the relationship between the state and measurement is given by the mixing matrix, c.

For completeness, the term u is included. Which represents any external input not dependent on the state, x. We will not explore this extra term in this module, instead, it is set to zero.



Bayesian modeling

- Prediction using state dynamics model

  $p(x_{t+1}|x_t)$

- Inference from noisy measurements

  $p(z_t|x_t)$

- Model x_t with a Gaussian (mean and covariance)

  $p(x_t) = \mathcal{N}(x_t, P_t)$

- Apply linear dynamics

$$p(x_{t+1}|x_t) = Ap(x_t)$$
$$p(z_t|x_t) = Cp(x_t)$$

- Add noise for motion and observations

$$p(x_{t+1}|x_t) = Ap(x_t) + v_m$$
$$p(z_t|x_t) = Cp(x_t) + v_o$$

- Introduce Gaussian model of x_t

$$p(x_{t+1}|x_t) = A\mathcal{N}(x_t, P_t) + \mathcal{N}(0, \Sigma_m)$$
$$p(z_t|x_t) = C\mathcal{N}(x_t, P_t) + \mathcal{N}(0, \Sigma_o)$$

- Consolidate expression using special properties

$$p(x_{t+1}|x_t) = A\mathcal{N}(x_t, P_t) + \mathcal{N}(0, \Sigma_m)$$
$$p(z_t|x_t) = C\mathcal{N}(x_t, P_t) + \mathcal{N}(0, \Sigma_o)$$

- Apply *linear* transform to Gaussian distributions

$$p(x_{t+1}|x_t) = \mathcal{N}(Ax_t, AP_tA^T) + \mathcal{N}(0, \Sigma_m)$$
$$p(z_t|x_t) = \mathcal{N}(Cx_t, CP_tC^T) + \mathcal{N}(0, \Sigma_o)$$

- Apply summation

$$p(x_{t+1}|x_t) = \mathcal{N}(Ax_t, AP_tA^T + \Sigma_m)$$
$$p(z_t|x_t) = \mathcal{N}(Cx_t, CP_tC^T + \Sigma_o)$$

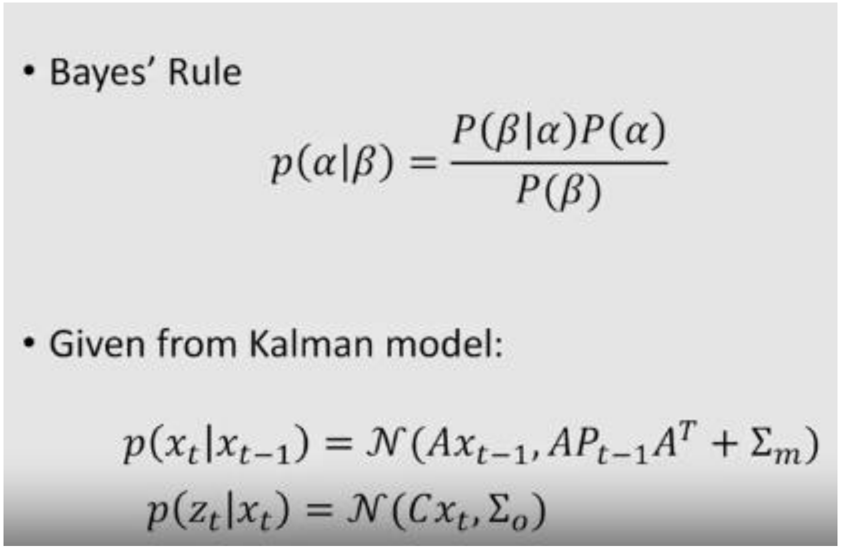
2.3. MAP Estimate of KF – Maximum-A-Posterior Estimation

- Bayes' Rule

$$p(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

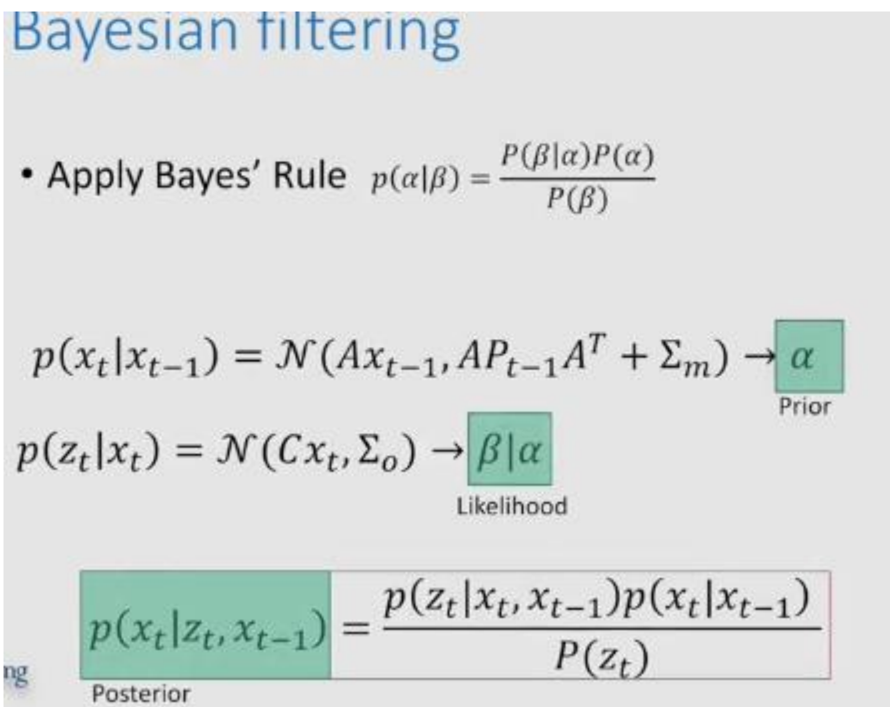
- Given from Kalman model:

$$p(x_t|x_{t-1}) = \mathcal{N}(Ax_{t-1}, AP_{t-1}A^T + \Sigma_m)$$
$$p(z_t|x_t) = \mathcal{N}(Cx_t, CP_tC^T + \Sigma_o)$$



From the dynamical system, the probability of the state given only the previous state can be represented with the prior information alpha.

Representing the information from our measurement model, beta provides observational evidence. Conditioned on a state, this evidence presents a constrained probability distribution known as the likelihood. Altogether, Bayes' Rule helps us to formulate an expression for the posterior probability.



Note :

The equation:

$$p(z_t|x_t) = N(Cx_t, \textstyle\sum_0)$$

should actually be :

$$p(z_t|x_t) = N(Cx_t, CP_tC^T + \textstyle\sum_0)$$

as you will observe in the following slides.

The posterior probability represents our best estimate of the state x of t, given information from both the previous state, x of t- 1, and observation zt. This estimate will provide a basis for the new mean over Gaussian distribution representing our state x of t.

The Maximum A-Posterior estimation technique can provide optimal best estimate of the distribution.

The map estimate is formed as an optimization problem over all values in the posterior distribution.

We drop the probabilities that are independent of the state such as the distribution of all measurements z of t unconditioned on the state x of t. Fully expanded, we see a maximization over the product of Gaussians.

A trick to calculate the MAP estimate is to take the logarithm of the product. The logarithm represents a monotonic function. So the optimal value of x sub t in a logarithmic function remains the optimal value of x sub t in the original function.

- Posterior distribution is another Gaussian
- MAP Estimates "optimal" x_t value
- Use MAP estimates to form a new mean and variance for the state

- Calculate the Maximum A Posteriori Estimate

$$\hat{x}_t = \underset{x_t}{\operatorname{argmax}}\, p(x_t|z_t, x_{t-1})$$

$$\hat{x}_t = \underset{x_t}{\operatorname{argmax}} \frac{p(z_t|x_t)p(x_t|x_{t-1})}{P(z_t)}$$

$$\hat{x}_t = \underset{x_t}{\operatorname{argmax}}\, p(z_t|x_t)p(x_t|x_{t-1})$$

$$\hat{x}_t = \underset{x_t}{\operatorname{argmax}}\, \mathcal{N}(Cx_t, CP_tC^T + \Sigma_o)\mathcal{N}(Ax_{t-1}, AP_{t-1}A^T + \Sigma_m)$$

- Calculate the Maximum A Posteriori Estimate

$$\hat{x}_t = \underset{x_t}{\operatorname{argmax}}\, \mathcal{N}(Cx_t, CP_tC^T + \Sigma_o)\mathcal{N}(Ax_t, AP_{t-1}A^T + \Sigma_m)$$

- Simplify with these substitutions

$$P = P_t = AP_{t-1}A^T + \Sigma_m$$
$$R = CP_tC^T + \Sigma_o$$

- Simplify the exponential form of $\mathcal{N}$ via logarithms

$$\hat{x}_t = \underset{x_t}{\operatorname{argmin}} \begin{array}{c} (z_t - Cx_t)R^{-1}(z_t - Cx_t) \\ +(x_t - Ax_{t-1})P^{-1}(x_t - Ax_{t-1}) \end{array}$$

- Solve optimization by setting the derivative to zero

$$\hat{x}_t = \underset{x_t}{\operatorname{argmin}} \begin{matrix} (z_t - Cx_t)R^{-1}(z_t - Cx_t) \\ +(x_t - Ax_{t-1})P^{-1}(x_t - Ax_{t-1}) \end{matrix}$$

$$0 = \frac{d}{dx_t}\begin{pmatrix} (z_t - Cx_t)R^{-1}(z_t - Cx_t) \\ +(x_t - Ax_{t-1})P^{-1}(x_t - Ax_{t-1}) \end{pmatrix}$$

All values inside the argmin function of the form,

$AR^{-1}A$ are actually of the form $AR^{-1}A^T$

where A = $(z_t - Cx_t), (x_t - Ax_{t-1})$

- Collect terms in the derivative

$$(C^T R^{-1} C + P^{-1})x_t = z_t^T R^{-1} C + P^{-1} A x_{t-1}$$

$$x_t = (C^T R^{-1} C + P^{-1})^{-1}(z_t^T R^{-1} C + P^{-1} A x_{t-1})$$

- Apply the *Matrix Inversion Lemma*

$$(C^T R^{-1} C + P^{-1})^{-1} = P - PC^T(R + CPC^T)^{-1}CP$$

- Define *Kalman Gain*: $K = PC^T(R + CPC^T)^{-1}$

- Expand the terms

$$x_t = (C^T R^{-1} C + P^{-1})^{-1} (C^T R^{-1} y_t + P^{-1} A x_{t-1})$$

$$x_t = (P - KCP)(C^T R^{-1} z_t + P^{-1} A x_{t-1})$$

$$x_t = A x_{t-1} + P C^T R^{-1} z_t - KCA x_{t-1} - KCPC^T R^{-1} y_t$$

$$x_t = A x_{t-1} - KCA x_{t-1} + (PC^T R^{-1} - KCPC^T R^{-1}) y_t$$

$$x_t = A x_{t-1} - KCA x_{t-1} + K y_t$$

$$\boxed{\hat{x}_t = A x_{t-1} + K(z_t - CA x_{t-1})}$$

- Convince yourself that $K = PC^T R^{-1} - KCPC^T R^{-1}$

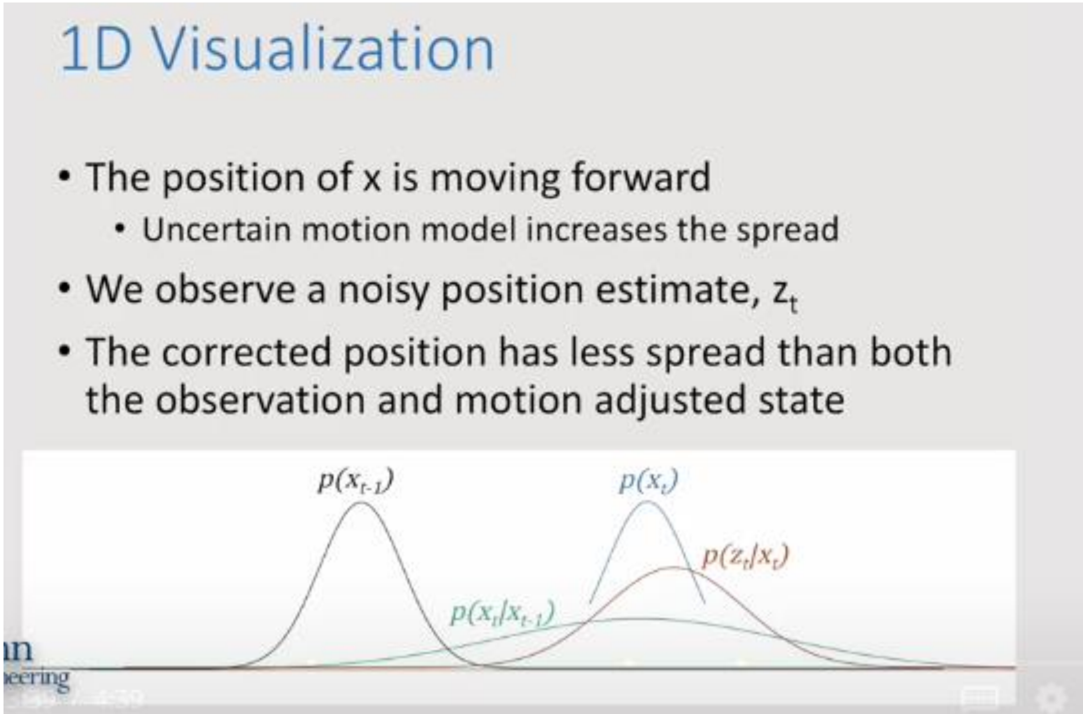
The first equation mentioned on the slide should be:

$$x_t = (C^T R^{-1} C + P^{-1})^{-1} (C^T R^{-1} z_t + P^{-1} A x_{t-1})$$

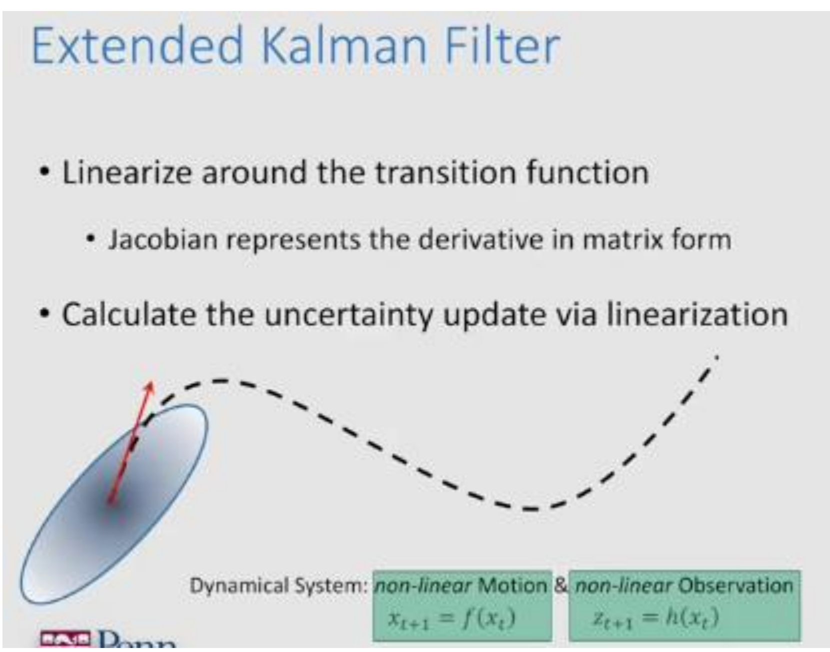
The z_t should replace the y_t in this and all the subsequent equations.

- Must update the covariance of the state

$$\boxed{\hat{P}_t = P - KCP}$$



2.4. Non-Linear Variations – Extended Kalman and Unscented Kalman Filter



- Covariance prediction $p(x_{t+1}|x_t) = \mathcal{N}(Ax_t, AP_tA^T + \Sigma_m)$

$$p(x_{t+1}|x_t) = \mathcal{N}\left(f(x_t), \frac{\partial f}{\partial x} P_t \frac{\partial f^T}{\partial x} + \Sigma_m\right)$$

- Kalman Gain

$$K = PC^T(\Sigma_o + CPC^T)^{-1}$$

$$K = P\frac{\partial h^T}{\partial x}\left(\Sigma_o + \frac{\partial h}{\partial x} P_t \frac{\partial h^T}{\partial x}\right)^{-1}$$

- Overall update

$$\hat{x}_t = f(x_{t-1}) + K(z_t - h(f(x_t)))$$

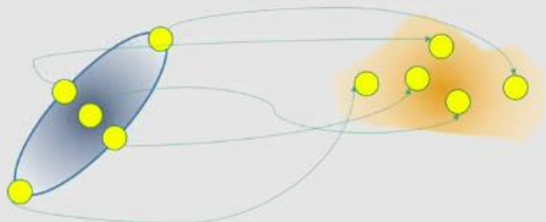
$$\hat{P}_t = P - K\frac{\partial h}{\partial x}P$$

Unscented Kalman Filter

- Noise distribution characterized by a set of points
- Transform these points in non-linear fashion
- Re-estimate mean and covariance for recalculating Gaussian distribution to characterize uncertainty

Unscented Kalman Filter

- Model the distribution based on a set of *sigma points*
  - Statistics of these points captures the distribution



- Approximate new distribution by computing the statistics of the *sigma points* $\mathcal{X}_i$
- Approximate a Gaussian with first two moments
  - Recalculate sigma points

- State prediction
  - Mean of Sigma points run through dynamic system

$$x_{t+1|x_t} = \frac{1}{N_{\mathcal{X}}} \sum_i f(\mathcal{X}_i)$$

- Uncertainty prediction
  - Covariance of sigma points run through dynamic system

$$P_{t+1|x_t} = \frac{1}{N_{\mathcal{X}}} \sum_i \left(f(\mathcal{X}_i) - x_{t+1|x_t}\right)\left(f(\mathcal{X}_i) - x_{t+1|x_t}\right)^T$$

- Expected Observation
    - Mean of Sigma points' expected obsrvation

$$z_{t+1|x_t} = \frac{1}{N_{\mathcal{X}}} \sum_i h(f(\mathcal{X}_i))$$

Recall Linear System
$K = PC^T(\Sigma_o + CPC^T)^{-1}$

- Kalman Gain
    - Utilize sigma points, not observation model

$$K = \frac{1}{N_{\mathcal{X}}} \sum_i \left(f(\mathcal{X}_i) - x_{t+1|x_t}\right)\left(h\big(f(\mathcal{X}_i)\big) - z_{t+1|x_t}\right)^T \cdot \left(\frac{1}{N_{\mathcal{X}}} \sum_i \left(h\big(f(\mathcal{X}_i)\big) - z_{t+1|x_t}\right)\left(h\big(f(\mathcal{X}_i)\big) - z_{t+1|x_t}\right)^T\right)^{-1}$$

- Update just as the linear filter
    - The covariance update will be slightly different
- See notes for good resources on further details

PROGRAMMING ASSIGNMENT:
Kalman Filter Tracking

3. Mapping

PROGRAMMING ASSIGNMENT:
2D Occupancy Grid Mapping

4. Bayesian Estimation – Localization

PROGRAMMING ASSIGNMENT:
Particle Filter Based Localization