Machine Learning in Robotics
Lecture 6: Gaussian Mixture Model and EM algorithm

Prof. Donghei Lee

Institute of Automatic Control Engineering
Technische Universität München

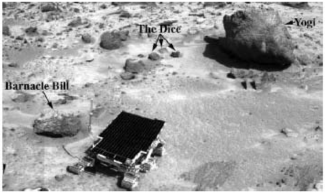
dhlee@tum.de

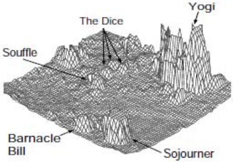
Today Lecture Outline

- Gaussian Mixture Model
- GMM Learning
- General Expectation Maximization Algorithm

Maximum Likelihood technique for a rover localization

Task: To perform rover localization by matching range maps.
Motivation: For greater autonomy in Mars rovers.





(Left) Annotated image, (Right) Terrain map generated from stereo image

- Global Map: panoramic imagery generated at the center of the area from lander.
- Local Map: Occupancy map of local terrain is generated using stereo vision on Sojourner.

Olson, Clark F., and Larry H. Matthies. *Maximum likelihood rover localization by matching range maps*. IEEE International Conference on Robotics and Automation. 1998.

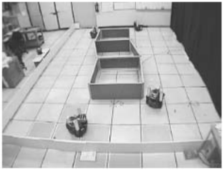
Maximum Likelihood technique for a rover localization

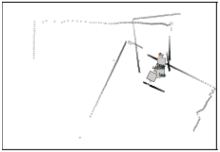
Task: To perform rover localization by matching range maps

- Rover position (x and y coordinates) denoted as X.
- Distances of nearby voxels (n occupied voxels) denoted as $D_1, D_2, \ldots, D_n$.
- The position X yielding the maximum likelihood value i.e. $\ln L(X) = l(X) = \sum_{i=1}^{n} \ln p(D_i|X)$ is chosen to be the position of the rover.
- $p(D_i|X)$ is a normal distribution with a constant additive term. Normal distribution models difference occupied voxels in btw global map and local map
- Results: Comparison of rover position determined by a human operator and by the proposed localization method. Similar results.

Localization for Mobile Robot Teams

Task: to infer the relative pose of every robot in the team without the use of GPS, external landmarks or instrumentation of the environment



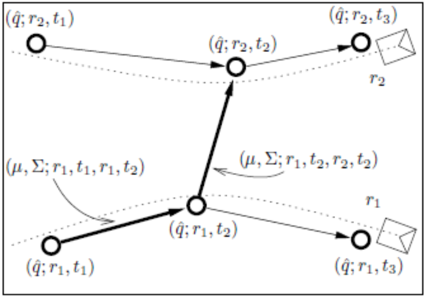


A. Howard, M. Mataric, G. Sukhatme. *Localization for Mobile Robot Teams Using Maximum Likelihood Estimation*. IROS 2002.

Localization for Mobile Robot Teams

Task: to infer the relative pose of every robot in the team

- r_1, r_2 : robots 1 and 2.
- Nodes: robot pose estimates.
- Arcs: observations (motion observation and robot observation).

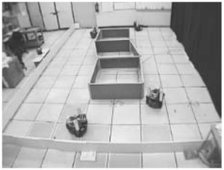


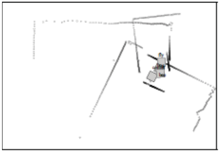
Aim: Maximize $P(O|H)$ using Maximum Likelihood Estimation , O: observation set (motion sensor and robot sensor), H: robot pose estimation set.

Localization for Mobile Robot Teams

Task: to infer the relative pose of every robot in the team

- experimental snapshots
- At t=1, the relative robot pose is completely unknown
- By time t=12 sec, both robots following the outer wall have observed both robots following inner wall. The two robots on the outer wall can correctly determine each other pose even though they have never seen each other.
- errors: At t=0, the localization error is high. By the time t=20, the robot performs stable localization, The average range error is about 5.5cm.

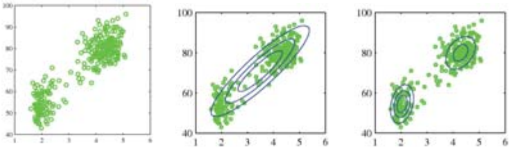




Mixture models

Consider the problem of modeling a pdf given a dataset of examples $X = \{\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)}, \ldots, \boldsymbol{x}^{(n)}\}$

- If the form of the underlying pdf is known (e.g. Single Gaussian distribution), the problem could be solved using the Maximum Likelihood Estimation method



Old Faithful data from Bishop2006

- Now we will consider an alternative density estimation method which is modeling the pdf with a mixture of parametric densities. In particular, we will focus on mixture models of Gaussian densities

$$\boxed{p(\boldsymbol{x}|\boldsymbol{\theta}) = \sum_{j=1}^{K} p(\boldsymbol{x}|\boldsymbol{\theta}_j)p(\omega_j) = \sum_{j=1}^{K} \pi_j \mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)}$$

Gaussian Mixture Model (GMM)

- Mixture of Gaussians
  - A superposition of K Gaussian densities $p(\boldsymbol{x}) = \sum_{j=1}^{K} \pi_j \mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)$
  - Parameters $\pi_j, \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j$
  - Properties: Asymmetry, multi-modality $0 \leq \pi_j \leq 1, \quad \sum_{j=1}^{K} \pi_j = 1$
- Previously, we estimated parameters for a single Gaussian distribution by MLE
- Log-likelihood function

$$l(\boldsymbol{\theta}) = \ln p(\boldsymbol{X}|\boldsymbol{\pi}, \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \sum_{i=1}^{n} \left[ln \left[\sum_{k=1}^{K} \pi_k \mathcal{N}(\boldsymbol{x}^{(i)}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \right] \right]$$

Gaussian Mixture Model (GMM)

Log-likelihood function

$$l(\boldsymbol{\theta}) = \ln p(\boldsymbol{X}|\boldsymbol{\pi}, \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \sum_{i=1}^{n} \left[ln \left[\sum_{k=1}^{K} \pi_k \mathcal{N}(\boldsymbol{x}^{(i)}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \right] \right]$$

Find the maximum of this function by differentiation
for $\boldsymbol{\Sigma}_k = \sigma_k^2 \boldsymbol{I}$

$$\frac{\partial l}{\partial \boldsymbol{\mu}_j} = 0 \rightarrow \hat{\boldsymbol{\mu}}_j = \frac{\sum_{i=1}^{n} p(\omega_j|\boldsymbol{x}^{(i)}, \boldsymbol{\theta})\boldsymbol{x}^{(i)}}{\sum_{i=1}^{n} p(\omega_j|\boldsymbol{x}^{(i)}, \boldsymbol{\theta})}$$

$$\frac{\partial l}{\partial \sigma_j} = 0 \rightarrow \hat{\sigma}_j^2 = \frac{\sum_{i=1}^{n} p(\omega_j|\boldsymbol{x}^{(i)}, \boldsymbol{\theta})(\boldsymbol{x}^{(i)} - \boldsymbol{\mu}_j)^2}{\sum_{i=1}^{n} p(\omega_j|\boldsymbol{x}^{(i)}, \boldsymbol{\theta})}$$

$$\frac{\partial l}{\partial \pi_j} = 0 \rightarrow \hat{\pi}_j = \frac{\sum_{i=1}^{n} p(\omega_j|\boldsymbol{x}^{(i)}, \boldsymbol{\theta})}{\sum_{k=1}^{K} \sum_{i=1}^{n} p(\omega_k|\boldsymbol{x}^{(i)}, \boldsymbol{\theta})}$$

Gaussian Mixture Model (GMM)

- NOT a closed form analytical solution for GMM parameters
- Due to responsibility depends on the GMM parameters
- Highly non-linear coupled system of equations

$\Rightarrow$ Iterative Numerical Optimization Technique is necessary. EM algorithm

EM for GMM

Given a Gaussian Mixture Model, the goal is to maximize the likelihood function w.r.t the parameters

1. Initialize $\pi_j, \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j$
2. E-step: Evaluate the responsibilities using the current parameters

$$p(\omega_k|\boldsymbol{x}^{(i)}, \theta) = \frac{\pi_k \mathcal{N}(\boldsymbol{x}^{(i)}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{j=1}^{K} \pi_j \mathcal{N}(\boldsymbol{x}^{(i)}|\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)}$$

3. M-step: Re-estimate the parameters using the current responsibilities

$$\hat{\boldsymbol{\mu}}_k = \frac{1}{n_k} \sum_{i=1}^{n} p(\omega_k|\boldsymbol{x}^{(i)}, \theta)\boldsymbol{x}^{(i)}$$

$$\hat{\boldsymbol{\Sigma}}_k = \frac{1}{n_k} \sum_{i=1}^{n} p(\omega_k|\boldsymbol{x}^{(i)}, \theta)(\boldsymbol{x}^{(i)} - \hat{\boldsymbol{\mu}}_k)(\boldsymbol{x}^{(i)} - \hat{\boldsymbol{\mu}}_k)^T$$

$$\hat{\pi}_k = \frac{n_k}{n} \quad \text{where} \quad n_k = \sum_{i=1}^{n} p(\omega_k|\boldsymbol{x}^{(i)}, \theta)$$

4. Evaluate the log-likelihood and check for convergence of either the parameters or the log-likelihood. If not converged, go to step 2.

$$l(\boldsymbol{\theta}) = \ln p(\boldsymbol{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{\pi}) = \sum_{i=1}^{n} \ln \sum_{k=1}^{K} \pi_k \mathcal{N}(\boldsymbol{x}^{(i)}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

Example

Probability to get a credit (A,B,C,D) from a lecture depends on the mean. Assume that the number of students for each credit (A,B,C,D) is a, b, c, d. Estimate the mean.

$$\begin{aligned}
\omega_1 &= A & P(A) &= 1/2 \\
\omega_2 &= B & P(B) &= \mu \\
\omega_3 &= C & P(C) &= 2\mu \\
\omega_4 &= D & P(D) &= 1/2 - 3\mu \\
&\text{where} & 0 \leq \mu &\leq 1/6
\end{aligned}$$

$$\begin{aligned}
& P(A) + P(B) + P(C) + P(D) = 1 \\
& P(a, b, c, d|\mu) = K(1/2)^a (\mu)^b (2\mu)^c (1/2 - 3\mu)^d \\
& \frac{\partial lnP(a, b, c, d|\mu)}{\partial \mu} = \frac{b}{\mu} + \frac{2c}{2\mu} - \frac{3d}{1/2 - 3\mu} = 0 \\
& \Rightarrow \mu = \frac{b + c}{6(b + c + d)}
\end{aligned}$$

Example

If it is known that $a = 14, b = 6, c = 9, d = 10$, then $\mu = 1/10$. However, now assume that c and d are known, but a and b are unknown.

$$a = \frac{0.5}{0.5+\mu}h \ , \ b = \frac{\mu}{0.5+\mu}h \ \Leftrightarrow \ \mu = \frac{b+c}{6(b+c+d)}$$

EM algorithm : Start with an initial parameter. Iterate E-step and M-step.

- Initialization : $\mu(0)$
- E-step : $b = \frac{\mu(t)}{1/2+\mu(t)}h = \mathbb{E}[b|\mu(t)]$
- M-step : $\mu(t+1) = \frac{b+c}{6(b+c+d)}$

Example

Given $h = 20, c = 9, d = 10$, estimate μ with an initial guess $\mu(0) = 0$ by the EM algorithm

t	$\mu(t)$	$b(t)$
0	0	0
1	0.0789	1. 3636
2	0.0848	1.4504
3	0.0852	1.4555
4	0.0852	1.4557
5	0.0852	1.4558
6	0.0852	1.4558

The Expectation-Maximization algorithm

- The EM is a general method for finding the ML estimate of the parameters of a pdf when the data has missing values
- Assume a dataset containing two types of features
  - A set of features X whose value is known. We call these the *incomplete* data
  - A set of features Z whose value is unknown. We call these the *missing* data
  - θ : model parameters
- We now define a joint pdf $p(X, Z|\theta)$ called the complete-data likelihood
- As suggested by its name, the EM algorithm operates by performing two basic operations over and over:
  - An Expectation step
  - A Maximization step

The Expectation-Maximization algorithm

- EXPECTATION : Find the expected value of the log-likelihood $\ln p(\boldsymbol{X}, \boldsymbol{Z}|\boldsymbol{\theta})$ with respect to the unknown data $\boldsymbol{Z}$, given the data $\boldsymbol{X}$ and the current parameter estimates $\boldsymbol{\theta}$

$$Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{old}) = \sum_{\boldsymbol{Z}} p(\boldsymbol{Z}|\boldsymbol{X}, \boldsymbol{\theta}^{old}) \ln p(\boldsymbol{X}, \boldsymbol{Z}|\boldsymbol{\theta})$$

- MAXIMIZATION : Find the argument $\boldsymbol{\theta}$ that maximizes the expected value defined by $Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{old})$

$$\boldsymbol{\theta}^{new} = \arg\max_{\boldsymbol{\theta}} Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{old})$$

- Convergence properties
  - Each iteration (E+M) is guaranteed to increase the log-likelihood.
  - EM algorithm is guaranteed to converge to a local maximum of the likelihood function.

每一步EM都会增加likelihood

EM可以保证likelihood收敛到局部最大值

EM algorithm to find MAP solution

- EM can be used to find MAP (maximum a posterior) solutions for models in which a prior $p(\boldsymbol{\theta})$ is defined over parameters.
- E-step : same as EM for ML
- M-step : Find the argument $\boldsymbol{\theta}$ that maximizes the expected value defined by $Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{old}) + lnp(\boldsymbol{\theta})$ instead of $Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{old})$

GMM revisited

- Consider the problem of maximizing the likelihood for the complete data set $\{\boldsymbol{X}, \boldsymbol{Z}\}$
- If complete data set $\{\boldsymbol{X}, \boldsymbol{Z}\}$ is given, the complete-data log likelihood function can be maximized trivially in closed form.
- In practice, the 隐藏的 latent variables are not given. Therefore, we consider the expectation of the complete-data log-likelihood, wrt the posterior distribution of the latent variables.

$$\boxed{\mathbb{E}[z_k^{(i)}] = \frac{\pi_k \mathcal{N}(\boldsymbol{x}^{(i)}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{j=1}^{K} \pi_j \mathcal{N}(\boldsymbol{x}^{(i)}|\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)} = \gamma(z_k^{(i)})}$$

未知数据后验概率

$$\boxed{\mathbb{E}_z\left[\ln P(\boldsymbol{X}, \boldsymbol{Z}|\boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{\pi})\right] = \sum_{i=1}^{n} \sum_{k=1}^{K} \gamma(z_k^{(i)}) \left\{\ln \pi_k + \ln \mathcal{N}(\boldsymbol{x}^{(i)}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)\right\}}$$

完整数据集的likelihood的期望

存在未知数据集Z时，根据未知数据的后验概率估计整个数据集的likeliood。

Relation to k-means

- EM for GMM
  - Soft assignment of data points to clusters
- K-means
  - Hard assignment of data points to clusters
  - A special case of ”EM for GMM”

$$\mathbb{E}[z_k^{(i)}] = \gamma(z_k^{(i)}) = \begin{cases} 1 & \text{if } \left|\boldsymbol{x}^{(i)} - \boldsymbol{\mu}_k\right| < \left|\boldsymbol{x}^{(i)} - \boldsymbol{\mu}_j\right| \quad , \forall j \\ 0 & \text{else} \end{cases}$$

EM algorithm

- For learning from partly unobserved data
- Maximum Likelihood estimate (MLE) vs. EM estimate
  - ML estimate

$$\boldsymbol{\theta} = argmax_{\boldsymbol{\theta}} p(\boldsymbol{X}|\boldsymbol{\theta})$$

  - EM estimate

$$\boldsymbol{\theta} = argmax_{\boldsymbol{\theta}} \mathbb{E}_Z[p(\boldsymbol{X}, \boldsymbol{Z}|\boldsymbol{\theta})]$$

Detection of target and arrow using GMM

Task: To perform archery by a humanoid robot iCub.
Proposed approach: reinforcement learning algorithms for learning the skill of archery

- EM based Reinforcement Learning (PoWER)
- chained vector regression (ARCHER)

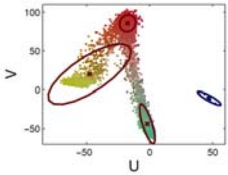
Subproblem: Image processing

- To detect where the target is
- To get the relative position of arrow from the target

Kormushev, Petar, et al. *Learning the skill of archery by a humanoid robot iCub*. IEEE-RAS International Conference on Humanoid Robots, 2010.

Detection of target and arrow using GMM

- The color detection is done in YUV color space.
- Only U and V components are used to ensure robustness against luminosity.
- A three component GMM for target, a single component GMM for arrow tip
- Bayesian Information Criterion (BIC) is used for optimizing the number of components in each GMM



Detection of target and arrow using GMM

After learning the likelihood value of each pixel in a new image can be used for classification of pixels. The classified image can be used to detect the center of target (red cross) and arrow (blue circle) in below figure.

Ground Plane Detection

Task : In mobile navigation, detection of ground and non-ground is useful in various application such as: object recognition, obstacle avoidance during autonomous navigation. This paper uses it for object tracking and following.

Conrad D. and DeSouza G. N. *Homography-based Ground Plane Detection for Mobile Robot Navigation Using a Modified EM Algorithm*. IEEE International Conference on Robotics and Automation. 2010.

Ground Plane Detection

- From two images, a large number of pixel correspondences are found by SIFT algorithm.
- EM used to classify pixel correspondences (x) from two images into 2 classes: *Ground plane* and *Non-Ground plane* in order to segment out the ground.
- Robot control: The robot uses pixels on the target object to follow. obstacle avoidance during autonomous navigation. It keeps the target object in the center of image view.

Ground Plane Detection

单应性

- Homography : a transformation matrix that relates the pixel coordinates of planar points as seen from two different viewing angles.

$$s\hat{p}_i = Hp_i$$

- Homography H is defined as $H = \hat{A}(R + \frac{t}{d}n^T)A^{-1}$ with $\hat{A}$ and A containing the intrinsic parameters of the cameras. The parameter Homography H is $\theta = \{R, t, n, d\}$, which is rotation matrix, translation vector, normal vector of the plane, distance between the camera and plane.
  本质的
- These parameters will be updated via EM.
- The pair of corresponding pixels $\hat{p}_i, p_i$ is referred to as pixel correspondence x_i.

Ground Plane Detection

- Expectation Maximization Algorithm

$$P(\boldsymbol{X}|\boldsymbol{Z},\boldsymbol{\theta}) = \frac{exp(-\frac{err_i^2}{2\sigma^2})}{\sum_i exp(-\frac{err_i^2}{2\sigma^2})} \quad with \quad err_i = \| \hat{p}_i - \frac{H_{ground}p_i}{s} \|$$

- where $\sigma = 3$, $H = A_1(R + \frac{\boldsymbol{t}}{d}\boldsymbol{n}^T)A_2^{-1}$, $\boldsymbol{\theta} = (R, \boldsymbol{t}, d, \boldsymbol{n})$, $\boldsymbol{X} = \boldsymbol{x}$ and $\boldsymbol{Z} = ground\ plane$.
- $P(\boldsymbol{X}|C_{non-ground},\boldsymbol{\theta}) = \frac{1-exp(-\frac{err_i^2}{2\sigma^2})}{\sum_i 1-exp(-\frac{err_i^2}{2\sigma^2})}$.
- After computing all posterior probabilities (E-step), the new model parameters are updated (M-step).
- In the M-step, an optimization algorithm (Simplex method) is used because it does not require an explicit gradient and shows faster convergence.
- Ground detection rate : 99.6%.

Announcements

- Further Reading
  - MLE: Duda Chap 3.1, 3.2, Bishop Chap. 1.2.4
  - GMM: Duda Chap. 3.4, Bishop Chap. 9.2
  - EM: Mitchell Chap. 6.12, Bishop Chap. 9.2-9.3
- Next Lecture
  - Nonparametric density estimation
  - Kernel Density Estimation, k-NNR, Parzen window