知能制御システム学

画像追跡 (3)
— ベイズ推定とパーティクルフィルタ —

東北大学 大学院情報科学研究科
鏡 慎吾
swk(at)ic.is.tohoku.ac.jp

2010.08.03

今日の内容

追跡対象が一時的に遮蔽される場合や，見えの似た物体が交錯するような場合は，過去の観測情報に基づいた予測を導入することが必要となる．そのような例としてパーティクルフィルタを紹介する．

できるだけ予備知識を必要とせず，かつ天下りのないように基礎の基礎から説明する．ただし厳密性には目を瞑る．

- 確率の基礎
- 観測モデル
- ベイズ推定
- 状態遷移モデル
- 逐次ベイズ推定
- パーティクルフィルタ

確率的画像追跡

問題定義: 画像の中で対象物体の存在する位置 (x, y) を求めたい.
（特に時刻間での物体の運動を考慮する場合に追跡と呼ぶことが多い）

(x, y) が一発で求まればよいが, なかなかそうはいかない.
確率分布を考える.

- 1次元の場合: x という確率変数が [0, 640) の範囲を動く. x がある値を取るような確率を求める, という作業をあらゆる x の値について行う.
- 2次元の場合: x, y という確率変数がそれぞれ [0, 640), [0, 480) の範囲を動く. (x, y) という組がある値を取るような確率を求める, という作業をあらゆる (x, y) の組について行う.

確率分布が求まれば, (例えば)その期待値として物体位置を推定できる.

なぜ期待値を取るのかは後述

確率変数, 確率, 確率密度

確率変数 x: 何か決められた定義域の値をランダムに取る

x が離散的な場合: x がある値 a を取る**確率** Pr(x = a) を考えることができる.

- 確率 Pr(x) は 0 〜 1 の値を取る.
- 確率 Pr(x) の定義域全範囲にわたる総和は 1 になる

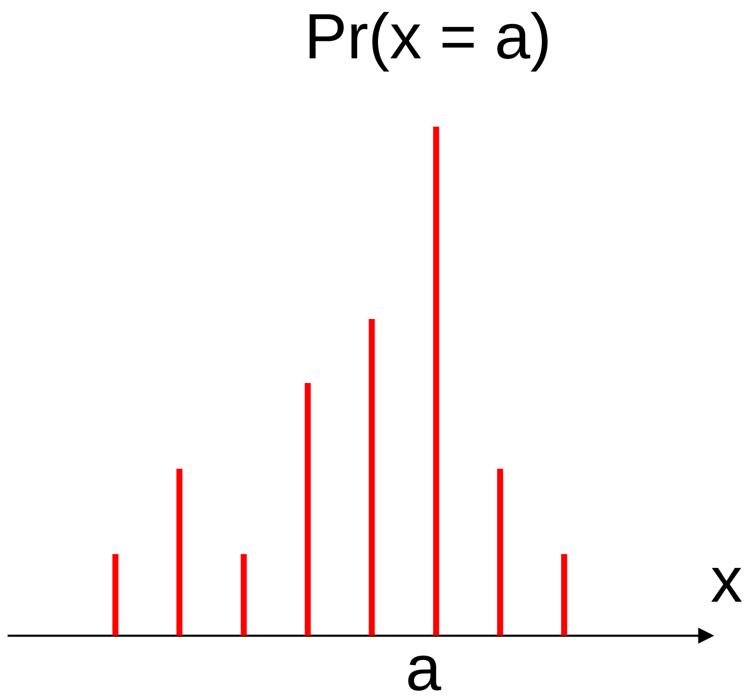
xが連続的な場合: x の値がある範囲に収まる確率 Pr(b <= x <= a) 等なら考えることができる. 常に範囲を考えるのは不便なので, xがある値を取るときの**確率密度** p(x = a) を考える. Pr(x <= a) の a による微分が p(x = a) である.

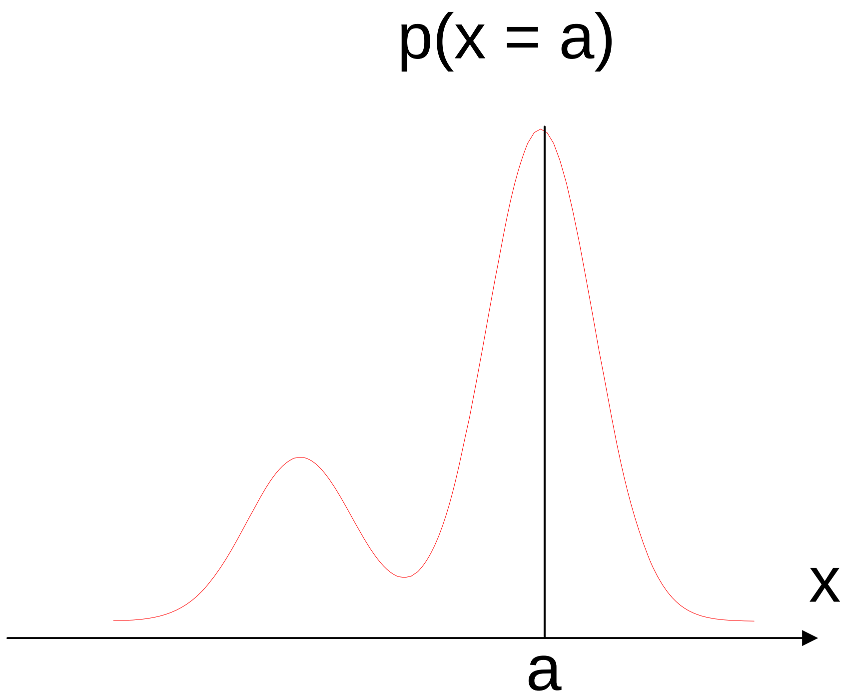
- 確率密度 p(x) は 0 以上の値を取る.
- 確率密度 p(x) の定義域全範囲にわたる積分は 1 である.

p(x) と書く場合, ある確率変数の確率密度分布 (とり得る x の値全体についての分布) を指す場合と, ある特定の x の値を考えている場合 (敢えて書くなら p(x = x) のこと)の両方があるので注意する. 前者の場合「確率密度分布」と呼ぶことが多い.

確率: 全部足すと1

確率密度: 面積が1





結合確率密度

$p(x_1 = a, x_2 = b)$:

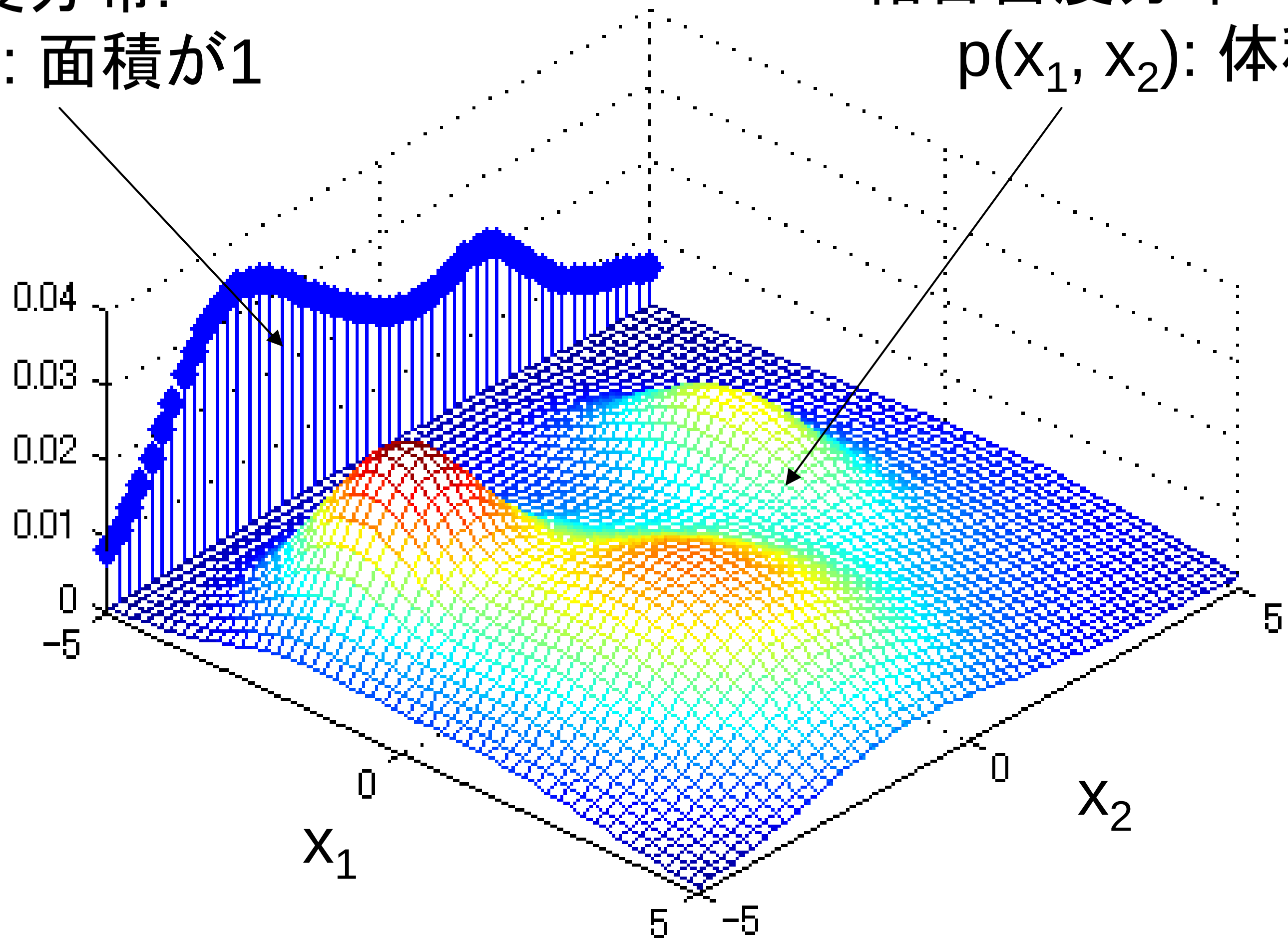
$x_1 = a, x_2 = b$ が同時に起きる確率密度．これを**同時確率密度**，あるいは**結合確率密度**と呼ぶ．$p(x_1, x_2)$ と書いたときの曖昧さは前ページと同じ.

$p(x_1, x_2)$ を x_1 に関して全範囲で積分すると，x_2 だけに関する確率密度 $p(x_2)$ が計算できる．この操作を周辺化と呼び，その結果得られるものを**周辺確率密度**と呼ぶ．これをさらに x_2 に関して全範囲で積分すると，もちろん 1 になる.

変数をたくさん書くと面倒なので，ベクトル $\mathbf{x} = (x_1, x_2)^T$ としてまとめて書くことがある．誤解の恐れがないときは (あるいは誤解の恐れの有無に関わらず，書き手が面倒なときは) ベクトルだかスカラだかあまり気にせずに書く場合もある．この講義では，書き手が面倒なので区別せずに書く.

周辺密度分布:
$p(x_2)$: 面積が1

結合密度分布:
$p(x_1, x_2)$: 体積が1



状態と観測

「追跡対象がどこにいるのか」を**状態** x と考える. これを求めたい.

画像などのセンサ情報から得られるのは**観測値** z である. 状態そのものが観測できるとは限らないし, 観測値から状態が一意に定まるとも限らない. 観測値も一般に確率変数である.

例1) x: 2次元位置, z: 2次元位置の観測値, v: 2次元ノイズ

$$z = x + v$$

例2) x: 3次元位置, z: 2次元位置の観測値, v: 2次元ノイズ, h(): ピンホールカメラモデル

$$z = h(x) + v$$

例3) x: 3次元位置, z: 画像, a: 物体の外見, b: 背景, v: 画像ノイズ, h(): 画像生成モデル

$$z = h(x, a, b) + v$$

観測モデルと逆問題

いろいろなケースを含む一般の場合として,

z = h(状態x, その他いろいろのランダムな要因)

と書けることが多い (何が「状態」で何が「その他いろいろ」とすべきかは一概には言えないが):

例1) 対象の位置を状態として, 外見や背景, センサノイズはその他とする

例2) 対象の位置と外見を状態として, 背景やセンサノイズはその他とする

例3) 対象の位置・外見も背景も状態として推定する. センサノイズはその他とする

「状態」と「その他いろいろ」がわかれば, 観測モデルから観測値は計算できる. しかしいま我々は, 観測値がわかったときに状態を求めたい. これは一般に容易ではない(逆問題)

条件付確率密度

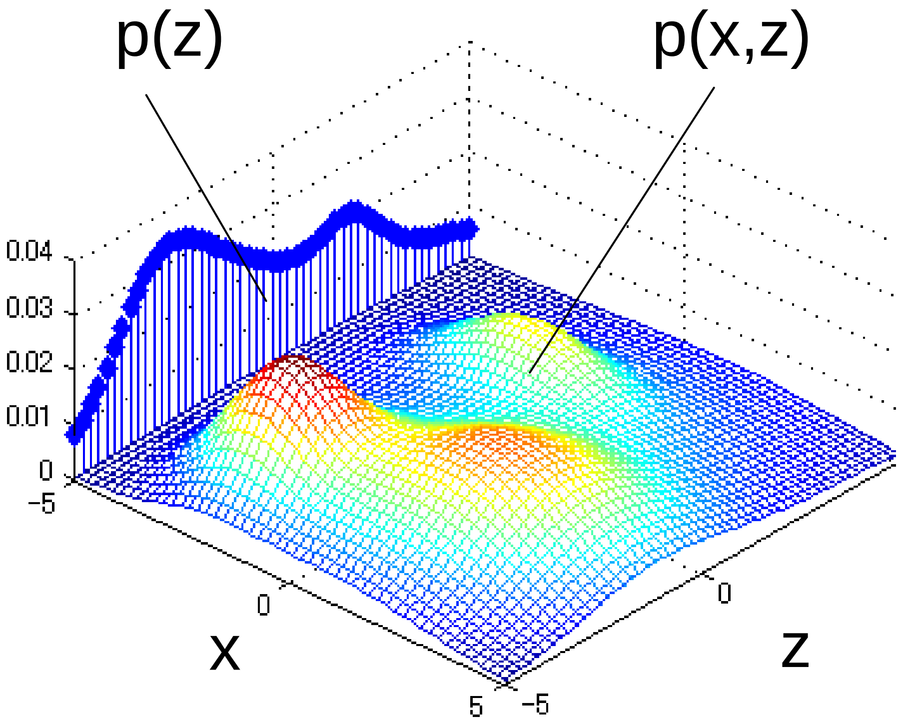
- 状態を知っていれば観測値について（少なくともその他いろいろ要因による影響の分を除いたある程度は）知ることができる.
- 観測値を知っていれば，状態について（完全ではないものの何らかの）情報を得ることができる.

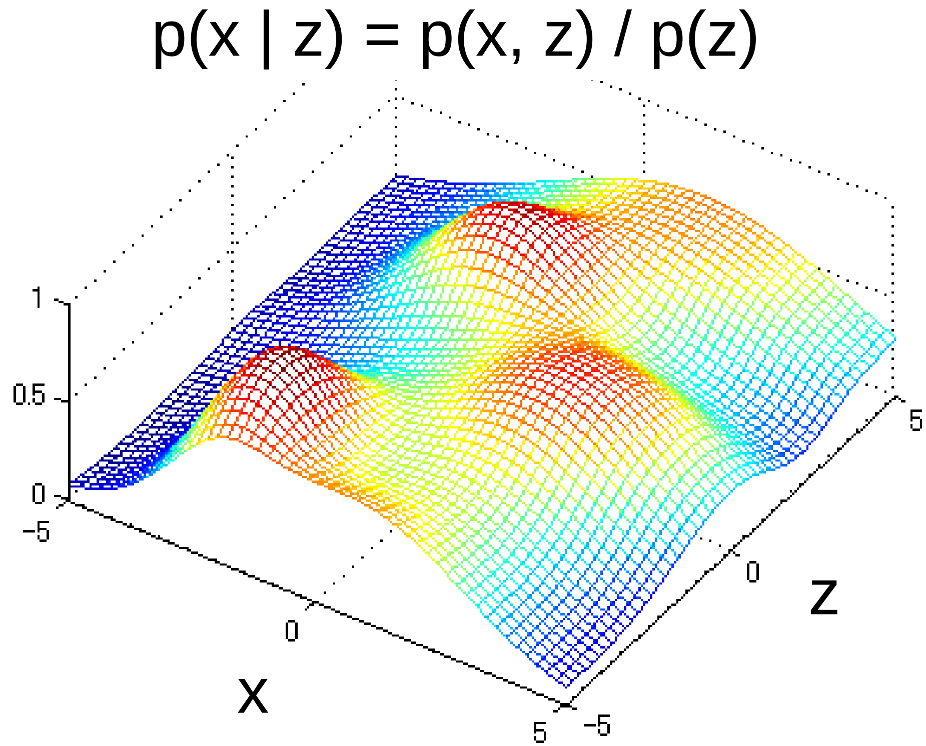
→ このような状況は**条件付確率密度**によって表現できる.

- 定義: $$\frac{p(x,z)}{p(z)} = p(x|z)$$

これも，p(x = a | z = b) のような具体的な値を入れた形で考えるとわかりやすいかも知れない.「z が値を b を取るとわかっているときに，x が値 a を取る確率密度」と解釈する.

p(x = a | z = b) p(z = b) = p(x = a, z = b) :「z が b になる確率密度」に,「z が b だったときに x が a になる確率密度」をかけると「x = a かつ z = b になる確率密度」が得られると解釈できる.





- p(x | z) は p(x, z) を p(z) = ∫ p(x, z) dx で正規化したものなので，z を固定して x の関数として見ると，積分して1になる．つまり確率密度分布の要件を満たしている．
- x を固定して z の関数として見ても，確率密度分布にならない（積分しても 1 にならない）

独立性

条件付確率から理解できるもう一つの重要な概念.

いま仮に p(x) p(z) = p(x, z) と表せたとする． このとき p(x | z) = p(x, z) / p(z) = p(x) p(z) / p(z) = p(x) である．つまり，z がある値を取ると知っているときの条件付確率 p(x | z) が，z について何も知らないときの確率 p(x) と一致する．すなわち，z に関する情報は，x に関して何らの情報をもたらさない.

このようなとき，つまり， p(x) p(z) = p(x, z) と表せるとき，x と z は **独立** であると呼ぶ.

注意) p(x | z) p(z) = p(x, z) は常に成り立つ（というかこれが p(x | z) の定義である）．p(x)p(z) = p(x, z) は常には成り立たない（独立なときのみ成り立つ）.

ベイズの定理

z = h(状態x, その他いろいろな要因)

関数 h() の形や，その他いろいろな要因の発生確率などを考えることで，条件付確率密度分布 p(z | x) を与えることができる場合が多い（順問題だから）.

しかし，p(x | z) を与えるのは難しいことが多い（逆問題だから）.

ところで定義より p(x | z) p(z) = p(x, z) = p(z | x) p(x) すなわち，

$$p(x \mid z) = p(z \mid x)\, p(x) / p(z)$$

が成り立つ．**ベイズの定理**と呼ぶ．p(x | z) が欲しいときに便利.

ベイズ推定

例えば，観測値 z = b が得られたとする．その際の x の確率密度「分布」p(x | z = b) を求めたい．p(x | z = b) = p(z = b | x) p(x) / p(z = b) として計算できる．このように状態推定を行うことをベイズ推定と呼ぶ．

p(z = b) の具体的な値を知らなくても，p(z = b | x) p(x) だけ計算して，その x による積分が 1 になるように正規化すればよい．

$$p(x \mid z = b) \propto p(z = b \mid x)\, p(x)$$

p(x): **事前確率密度分布**（観測する前に分かっている密度分布）

p(z = b | x): **観測尤度** (z = b を得たときの，x の尤もらしさ)

注: 尤度は x の関数として見なくてはならない．x について積分しても 1 にならないので確率ではない．

p(x | z = b): **事後確率密度分布**（観測後に分かった密度分布）

条件付期待値

$E[x] = \int x\, p(x)\, dx$: 確率変数 x の**期待値**（「x円もらえる確率密度が p(x) です．平均的にはいくらもらえると期待できますか?」の値）

$E[x \mid z] = \int x\, p(x \mid z)\, dx$: z を既知とした場合の x の**条件付期待値**．つまり事後確率密度分布を使って計算された期待値.

得られた観測値 z を使って状態 x を推定する方法は無数に考えられる．いまそのような推定法の一つを $\xi(z)$ と書くことにする.
$\xi(z)$ と真の状態 x の二乗誤差の条件付期待値 $E[(\xi - x)^2 \mid z]$ は,
$\xi(z) = E[x \mid z]$ という方法で推定すると最小になることが示せる.

この意味で，ベイズ推定によって得られた事後密度分布に対して期待値計算を行うこと(= 条件付期待値を計算すること)により，最適な状態推定が実現できるといえる.

条件付期待値が最小分散推定になることの証明

$\xi(z)$ が x に依存しないことに注意しながら，$E[(\xi(z) - x)^2 | z]$ を展開して ξ に関して平方完成する．

$$
\begin{aligned}
&E[\|\xi - x\|^2 \mid z] \\
&= \int (\xi - x)^{\mathrm{T}} (\xi - x) p(x|z) dx \\
&= \xi^{\mathrm{T}}\xi \int p(x|z) dx - 2\xi^{\mathrm{T}} \int x p(x|z) dx + \int x^{\mathrm{T}} x p(x|z) dx \\
&= \xi^{\mathrm{T}}\xi - 2\xi^{\mathrm{T}} \int x p(x|z) dx + \int x^{\mathrm{T}} x p(x|z) dx \\
&= \left\| \xi - \int x p(x|z) dx \right\|^2 - \left\| \int x p(x|z) dx \right\|^2 + \int x^{\mathrm{T}} x p(x|z) dx
\end{aligned}
$$

$\xi(z) = \int x\, p(x|z)\, dx$ で最小値を取ることがわかる．

事前確率密度の求め方

事前確率密度はどういう風に決めるのか

例1) 全くわからない → 一様分布
例2) 問題の構造や常識から決めることができる: たとえば
「人間が天井にはりついていることは滅多にない」とか.
例3) 前の時刻の状態推定結果を利用する.

時々刻々と変化していく状態 x の推定分布を, 毎時刻得られる観測 z によって例3の要領で逐次的に更新していくことができそうである.

もちろん, 現時刻の状態が, 前の時刻で推定された状態のままとは限らない. でも多くの「追跡問題」では, 前時刻と現時刻の状態は独立ではないと考えられ, 両者間の条件付確率を利用できる.

状態遷移モデル

時刻 k の状態を x_k と書く.

$x_k = f(x_{k-1},$ その他いろいろな要因$)$

1時刻分の予測を表現している．観測モデルのときと同じように，この関係から $p(x_k \mid x_{k-1})$ を与えることができる場合が多い.

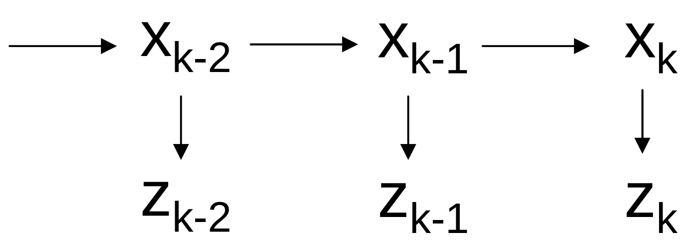
前時刻と現時刻の間のみの関係を考えて，2時刻前の影響は受けていないと考える（1次マルコフ性）．状態として，例えば位置だけでなく速度・加速度なども考えるようにすれば，これでもかなり現実的なモデル化ができる.

例1) 位置がランダムウォーク
例2) 速度がランダムウォーク，位置は速度の積分
例3) 加速度がランダムウォーク，以下略
例4) 運動方程式 + ランダムノイズ

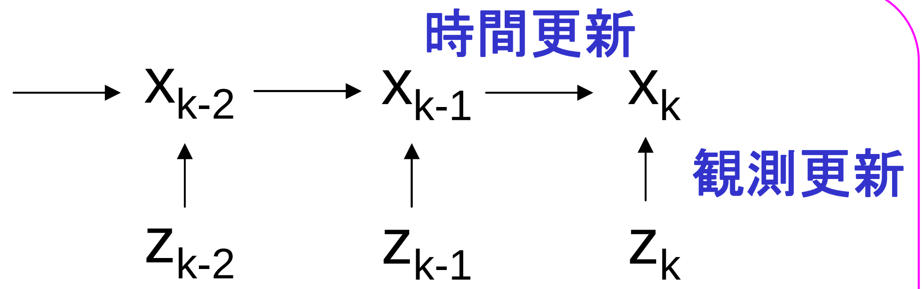
逐次的ベイズ推定

欲しいもの: $p(x_k \mid z_1, z_2, \cdots, z_k) = p(x_k \mid z_{1:k})$
つまり，時々刻々と得られる観測値 z_k を初期時刻から現時刻まで既知としたときの現時刻の状態の確率密度分布を得たい．z をたくさん書くのが面倒なので，$\{ z_1, z_2, \cdots, z_k \}$ を $z_{1:k}$ と書くことにする．

以下を仮定する:
[1] $p(z_k \mid x_k, z_{1:k-1}) = p(z_k \mid x_k)$ (観測は，現在の状態だけから決まる)
[2] $p(x_k \mid x_{k-1}, z_{1:k-1}) = p(x_k \mid x_{k-1})$ (状態は，前時刻の状態だけから決まる)



因果関係の流れ



計算手順の流れ

観測更新

$$p(x_k|z_{1:k}) = \frac{p(x_k, z_{1:k})}{p(z_{1:k})} = \frac{p(x_k, z_k, z_{1:k-1})}{p(z_k, z_{1:k-1})}$$

$$= \frac{p(z_k|x_k, z_{1:k-1})p(x_k, z_{1:k-1})}{p(z_k, z_{1:k-1})}$$

$$= \frac{p(z_k|x_k, z_{1:k-1})p(x_k|z_{1:k-1})p(z_{1:k-1})}{p(z_k|z_{1:k-1})p(z_{1:k-1})}$$

$$= \frac{p(z_k|x_k)p(x_k|z_{1:k-1})}{p(z_k|z_{1:k-1})}$$ （仮定[1]を使用）

$$\propto p(z_k|x_k)p(x_k|z_{1:k-1})$$

- 密度「分布」に興味があるので，x_k について積分したときに１になるように正規化すればよいから分母は不要
- 結局，現時刻の観測値を得る前の密度分布に観測尤度を乗算することになる

時間更新

$$p(x_k|z_{1:k-1}) = \int p(x_k, x_{k-1}|z_{1:k-1})dx_{k-1}$$

(周辺密度分布の計算)

$$\begin{aligned}
p(x_k, x_{k-1}|z_{1:k-1}) &= \frac{p(x_k, x_{k-1}, z_{1:k-1})}{p(z_{1:k-1})} \\
&= \frac{p(x_k|x_{k-1}, z_{1:k-1})p(x_{k-1}, z_{1:k-1})}{p(z_{1:k-1})} \\
&= \frac{p(x_k|x_{k-1}, z_{1:k-1})p(x_{k-1}|z_{1:k-1})p(z_{1:k-1})}{p(z_{1:k-1})} \\
&= p(x_k|x_{k-1}, z_{1:k-1})p(x_{k-1}|z_{1:k-1}) \\
&= p(x_k|x_{k-1})p(x_{k-1}|z_{1:k-1})
\end{aligned}$$

（仮定[2]を使用）

まとめて $p(x_k|z_{1:k-1}) = \int p(x_k|x_{k-1})p(x_{k-1}|z_{1:k-1})dx_{k-1}$

以上の式を利用して，適当な初期密度分布 $p(x_0)$ から始めて時間更新，観測更新を繰り返していくことで，現時刻の状態の確率密度分布が推定できる．これを逐次的ベイズ推定と呼ぶ．

時間更新と観測更新を順次繰り返していけばよい，というのがミソである．実は，それぞれの更新処理は，必ずしも条件付確率としてモデル化しなくてもよい．

例1) カルマンフィルタは，時間更新も観測更新も，関数 f() や h() の形をそのまま利用する．

例2) SIR型パーティクルフィルタは，時間更新は関数 f() の形をそのまま利用し，観測更新は条件付確率としてモデル化する．

実際の計算

以上のような確率密度のを直接計算するのは容易でない．まずそもそも連続の分布の計算は，一般には数値計算に頼るしかない．しかし，状態や観測の次元が高くなると数値計算も困難になる．さてどうするか．

例1) 問題のクラスを限定して，厳密計算する

→ 線形システム・正規分布に限定したのがカルマンフィルタ

例2) 数値的な近似解法を用いる

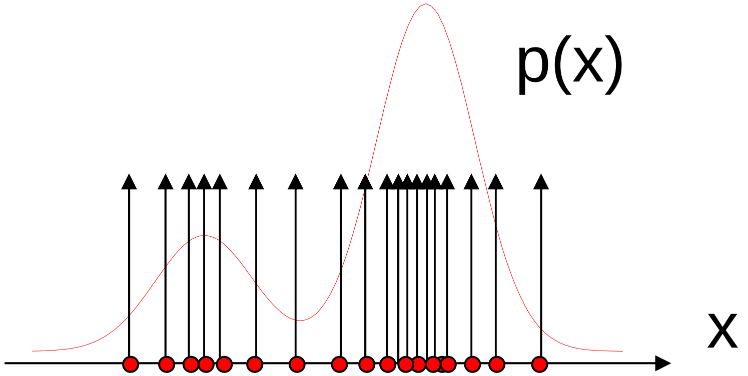
→ モンテカルロ近似を用いるのがパーティクルフィルタ

パーティクルフィルタ

確率密度分布を，状態空間に散らばるパーティクルの（重みつき）平均として表現する（モンテカルロ積分）．各パーティクルが，状態の候補であると考えるとわかりやすい．

$$p(x) \approx \frac{1}{N}\sum_{i=1}^{N}\delta(x - x^i)$$

$$\Pr(x \leq a) = \int_{-\infty}^{a} p(x)dx$$



パーティクルは疑似乱数によって発生させる．

- x ~ U([a, b]): 区間 [a, b] の一様分布に従って乱数発生
- x ~ N(m, σ^2): 平均 m，分散 σ^2 の正規分布に従って乱数発生
  「サンプル x を分布 N(m, σ^2) から抽出する」あるいは「N(m, σ^2)をサンプリングしてサンプル x を得る」と読む

確率密度分布 $p(x_k \mid z_{1:k})$ に従う乱数を発生させることで，状態推定を近似表現したい．さて，そのような乱数はどうやって発生させればよいのか?（分布を数式で表したりできているわけではない点に注意）

Importance Sampling:

容易にサンプルできない密度分布 $p(x)$ の代わりに，別の密度分布 $q(x)$（importance density または proposal density と呼ぶ）から x^i を抽出し，重み $\pi^i = p(x^i) / q(x^i)$ を用いた重みづけ平均によって表現する.

$$p(x) \approx \frac{1}{N} \sum_{i=1}^{N} \frac{p(x^i)}{q(x^i)} \delta(x - x^i)$$

サンプリングのしかたによって，パーティクルフィルタにはさまざまなバリエーションがある.

SIR型パーティクルフィルタ

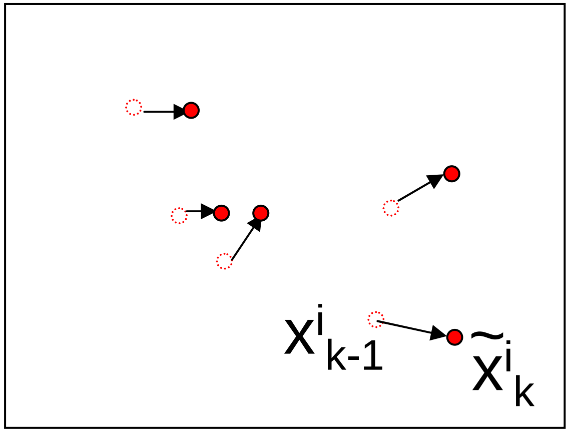
- Sampling Importance Resampling 型パーティクルフィルタ，または，bootstrap filter，Condensation などの名で知られる最も基本的なパーティクルフィルタ.
- 以下のように，時間更新，観測更新，リサンプリングを繰り返す.

時間更新は，状態遷移方程式 f() を各パーティクルに適用することで行う:

$$\tilde{x}_k^i = f(x_{k-1}^i, v_{k-1}^i)$$

v^i_{k-1}: 疑似乱数

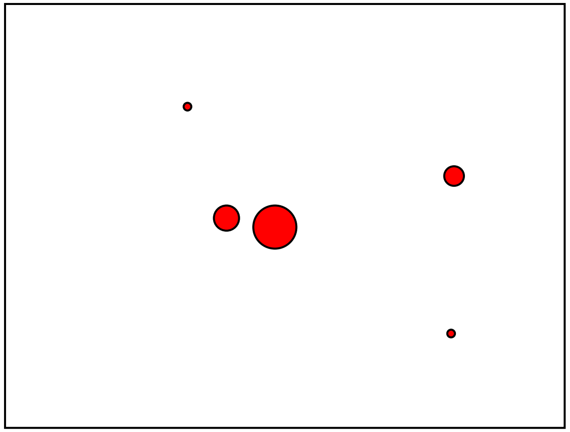
$$p(x_k | z_{1:k-1}) \approx \frac{1}{N} \sum_i \delta(x - \tilde{x}_k^i)$$



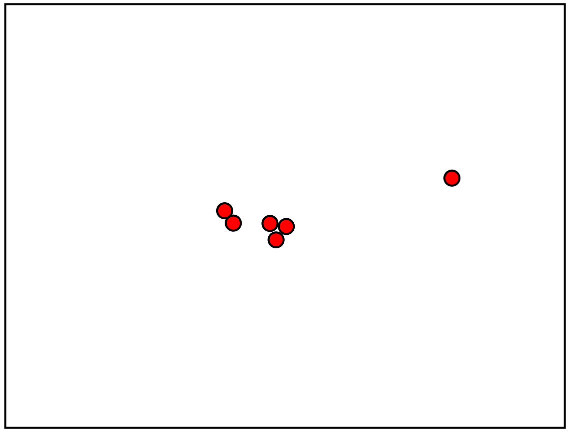
観測更新は，尤度関数を重みとすることで行う:

$$p(x_k|z_{1:k}) \propto p(z_k|x_k)p(x_k|z_{1:k-1})$$
$$\approx \frac{1}{N}\sum_i p(z_k|\tilde{x}_k^i)\delta(x-\tilde{x}_k^i)$$

よって，各サンプル点で尤度関数の値が評価できればよい（尤度関数が具体的な関数の形で与えられる必要はない）



このまま単純に繰り返していくと，重みがほぼ0の無意味なパーティクルだらけになってしまうので，リサンプリングを行う（各パーティクルの重みを出現確率として，N個のパーティクルを復元抽出する）

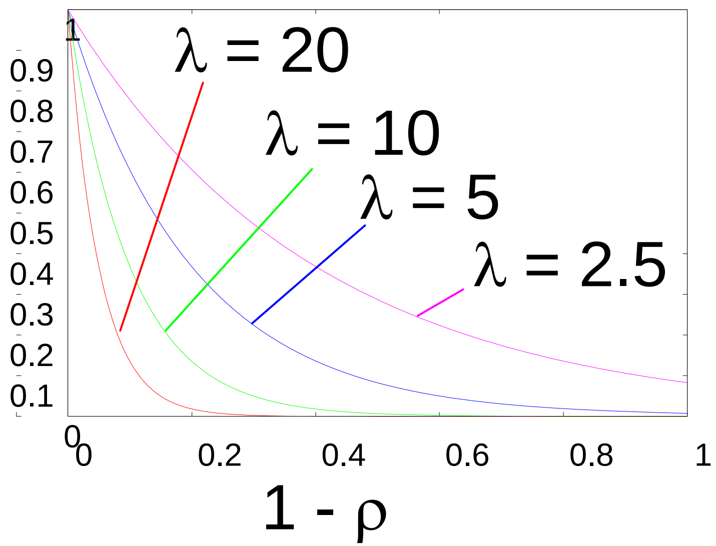


尤度関数の定義例

色ヒストグラムに基づく定義 (Perez, 2002)

追跡モデルの正規化色ヒストグラム q
候補領域 (パーティクル位置の周囲)の正規化色ヒストグラム: $p(x^i)$
$p(x^i)$ と q の Bhattacharya 係数: $\rho(p(x^i), q)$（前回資料参照）

尤度関数: $p(z|x^i) = \exp\left\{-\lambda\left[1 - \rho(p(x^i), q)\right]\right\}$ λ: 定数



サンプルプログラム

sample program:
- pf.cpp

- OpenCV の CvConDensation (SIR型パーティクルフィルタ) に全面的に依存している. なお, CvConDensation では, 線形な状態遷移モデルしか扱えない.
- 状態は, 画像上の x, y 座標とその各微分の 4 次元とし, 等速運動モデルを採用.
- 観測モデルは Perez (2002) の定義を採用. ヒストグラムの計算は前回と同じ.

References

- B. D. O. Anderson and J. B. Moore: Optimal Filtering, Prentice-Hall, 1979.
- B. Ristic, S. Arulampalam and N. Gordon: Beyond the Kalman Filter, Artech House, 2004.
- 片山: 新版 応用カルマンフィルタ, 朝倉書店, 2000.
- 樋口: 粒子フィルタ, 電子情報通信学会誌, Vol.88, No.12, pp.989-994, 2005.
- P. Perez, C. Hue, J. Vermaak and M. Gangnet: Color-Based Probabilistic Tracking, Proc. European Conference on Computer Vision (ECCV2002), pp.661-675, 2002.