

# 粒子フィルタとLDAへの応用

五十嵐未来

## 1 粒子フィルタ

### 1.1 状態空間モデルと隠れマルコフモデル

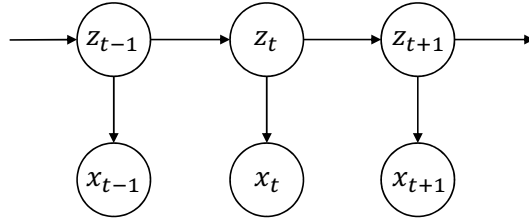


図 1: 動的モデルのグラフィカルモデル例

図 1 のようなモデルを考える。 $z$  は潜在変数であり、 $x$  が観測データである。このとき、一般に潜在変数が

離散：隠れマルコフモデル

連続：状態空間モデル

と呼ばれる。例えば、状態空間モデルであれば、

$$x_t = \alpha z_t + \epsilon_t, \quad \epsilon_t \sim N(0, \sigma_\epsilon) \quad (1)$$

$$z_t = \beta z_{t-1} + \tau_t, \quad \tau_t \sim N(0, \sigma_\tau) \quad (2)$$

のようなモデルを想定している。

このような線形正規状態空間モデルの推定法としては、カルマンフィルタと呼ばれる手法が一般的に用いられる。しかし、 $h, g$  を任意の非線形関数、 $\epsilon_t, \tau_t$  が任意の非正規分布に従う誤差項とした

$$x_t = h(z_t, \epsilon_t) \quad (3)$$

$$z_t = g(z_{t-1}, \tau_t) \quad (4)$$

のような非線形非正規状態空間モデルも考えられる。このようなモデルにも用いることが出来る推定法の一つが粒子フィルタ（Particle filter）である<sup>1</sup>。以下では、 $\alpha, \beta, \epsilon, \tau$  といったパラメータは既知であり、観測データ  $x_{1:T}$  から、観測できない潜在変数  $z_{1:T}$  を推定したいという状況を想定する<sup>2</sup>。ここでのモデルは、式 (3)(4) のような一般の非線形非正規状態空間モデルであり、図 1 のように、 $x_t$

<sup>1</sup>逐次モンテカルロ法（Sequential Monte Carlo）とも呼ばれる

<sup>2</sup>これらのモデルパラメータも未知として推定する状況は、ここでは考えない。キーワードとしては「自己組織型状態空間モデリング」（Kitagawa, 1998）などで調べるとよい。

は  $z_t$  にのみ、 $z_t$  は  $z_{t-1}$  にのみ依存するものとする。したがって、 $x_{1:T}, z_{1:T}$  の結合分布は

$$p(x_{1:T}, z_{1:T}) = p(z_0) \prod_{t=1}^T p(x_t|z_t)p(z_t|z_{t-1}) \quad (5)$$

である。

## 1.2 粒子フィルタ

時刻  $t+1$  において、観測データ  $x_{1:t}$  が与えられたもとで、新たな観測データ  $x_{t+1}$  を生成する潜在変数  $z_{t+1}$  の条件付確率分布は

$$\begin{aligned} p(z_{t+1}|x_{1:t}) &= \int p(z_{t+1}|z_{1:t})p(z_{1:t}|x_{1:t})dz_{1:t} \\ &= \int p(z_{t+1}|z_{1:t})\frac{p(z_{1:t}|x_{1:t})}{q(z_{1:t}|x_{1:t})}q(z_{1:t}|x_{1:t})dz_{1:t} \\ &= \int p(z_{t+1}|z_{1:t})\omega(z_{1:t})q(z_{1:t}|x_{1:t})dz_{1:t} \end{aligned} \quad (6)$$

で与えられる。ただし、 $\omega(z_{1:t}) = \frac{p(z_{1:t}|x_{1:t})}{q(z_{1:t}|x_{1:t})}$  であり、 $q(z_{1:t}|x_{1:t})$  は、分析者が自由に設定できる提案分布である。

提案分布からのサンプル  $z_{1:t}^{(s)} \sim q(z_{1:t}|x_{1:t})$  ( $s = 1, \dots, S$ ) を用いた近似  $\tilde{p}_s$  は  $S \rightarrow \infty$  で  $p(z_{1:t}|x_{1:t})$  に弱収束することが知られている。

$$\begin{aligned} \tilde{p}_s(z_{1:t}|x_{1:t}) &= \sum_{s=1}^S \frac{\omega(z_{1:t}^{(s)})z_{1:t}^{(s)}}{\sum_{s'=1}^S \omega(z_{1:t}^{(s')})} \\ &\xrightarrow{d} \omega(z_{1:t})q(z_{1:t}|x_{1:t}) \\ &= p(z_{1:t}|x_{1:t}) \end{aligned} \quad (7)$$

よって、式 (6) の積分を  $\tilde{p}_s$  を用いて近似すると、

$$\begin{aligned} p(z_{t+1}|x_{1:t}) &= \int p(z_{t+1}|z_{1:t})p(z_{1:t}|x_{1:t})dz_{1:t} \\ &\approx \int p(z_{t+1}|z_{1:t})\tilde{p}_s(z_{1:t}|x_{1:t})dz_{1:t} \\ &= \sum_{s=1}^S \bar{\omega}(z_{1:t}^{(s)})p(z_{t+1}|z_{1:t}^{(s)}), \quad \bar{\omega}(z_{1:t}^{(s)}) = \frac{\omega(z_{1:t}^{(s)})}{\sum_{s'=1}^S \omega(z_{1:t}^{(s')})} \end{aligned} \quad (8)$$

提案分布は自由に決めることができるため、

$$\begin{aligned} q(z_{1:t}|x_{1:t}) &= q(z_t|x_{1:t}, z_{1:t-1})q(z_{t-1}|x_{1:t-1}, z_{1:t-2}) \cdots q(z_1|x_1, z_0)q(z_0) \\ &= q(z_t|x_{1:t}, z_{1:t-1})q(z_{1:t-1}|x_{1:t-1}) \end{aligned} \quad (9)$$

と仮定してよい。ただし、数学的には  $q(z_{1:t}|x_{1:t}) = q(z_t|x_{1:t}, z_{1:t-1})q(z_{1:t-1}|x_{1:t})$  のところに、上の

定式化を仮定している点に注意が必要である。したがって、重み  $\omega(z_{1:t}^{(s)})$  の更新式は

$$\begin{aligned}
\frac{\omega(z_{1:t}^{(s)})}{\omega(z_{1:t-1}^{(s)})} &= \frac{p(z_{1:t}^{(s)}|x_{1:t})}{q(z_{1:t}^{(s)}|x_{1:t})} \bigg/ \frac{p(z_{1:t-1}^{(s)}|x_{1:t-1})}{q(z_{1:t-1}^{(s)}|x_{1:t-1})} \\
&= \frac{p(z_{1:t}^{(s)}, x_{1:t})/p(x_{1:t})}{p(z_{1:t-1}^{(s)}, x_{1:t-1})/p(x_{1:t-1})} \times \frac{q(z_{1:t-1}^{(s)}|x_{1:t-1})}{q(z_t^{(s)}|x_{1:t}, z_{1:t-1}^{(s)})q(z_{1:t-1}^{(s)}|x_{1:t-1})} \\
&\propto \frac{p(z_{1:t}^{(s)}, x_{1:t})}{p(z_{1:t-1}^{(s)}, x_{1:t-1})} \times \frac{1}{q(z_t^{(s)}|x_{1:t}, z_{1:t-1}^{(s)})} \\
&= \frac{p(z_{1:t}^{(s)}, x_{1:t}|x_{1:t-1}, z_{1:t-1}^{(s)})}{q(z_t^{(s)}|x_{1:t}, z_{1:t-1}^{(s)})} \\
&\propto \frac{p(x_t|z_t^{(s)}, z_{1:t-1}^{(s)}, x_{1:t-1})p(z_t^{(s)}|z_{1:t-1}^{(s)}, x_{1:t-1})}{p(z_t^{(s)}|x_{1:t-1}, z_{1:t-1}^{(s)})} \\
&= p(x_t|z_t^{(s)}, z_{1:t-1}^{(s)}, x_{1:t-1})
\end{aligned} \tag{10}$$

よって、次の更新式を得る。

$$\omega(z_{1:t}^{(s)}) \propto \omega(z_{1:t-1}^{(s)})p(x_t|z_t^{(s)}, z_{1:t-1}^{(s)}, x_{1:t-1}) \tag{11}$$

このようにサンプルの重み  $\omega(z_{1:t}^{(s)})$  を逐次的に更新していくことができる<sup>3</sup>。

### 1.3 リサンプリング

ここまでで粒子フィルタの半分を説明したことになる。しかし、上述したサンプリングでは、「ランダムに  $z_{t+1}^{(s)}$  を発生させて、尤度  $p(x_{t+1}|z_{t+1}^{(s)})$  に応じて重みを計算する」という操作を繰り返しているが、たまたまそれらしいサンプル  $z_{t+1}^{(s)}$  が得られれば良いが、ほとんどの施行は無駄になっていることが分かる。さらに、データ数が増えると、一部の重みにだけ尤度が偏ってしまい、さらにサンプリング効率が悪くなる。そこで、大きくなった重みを分裂させ、小さくなった重みを消滅させることで、サンプリング効率悪化の原因である重みの偏りを解消したい。これがリサンプリングという操作である<sup>4</sup>。

リサンプリングは毎回行われるわけではなく、Effective Sample Size (ESS) という値を監視し、閾値  $r$  よりも小さくなった場合にのみリサンプリングが行われる。この、リサンプリングを不必要に行ってはいけない理由については、Appendix に図解している。ESS は、

$$ESS = \frac{1}{\sum_{s=1}^S \bar{\omega}(z_{1:t}^{(s)})^2} \tag{12}$$

で与えられるが、式からも分かるように重みの偏りが大きくなると ESS が小さくなる。つまり、ESS によって近似効率を監視し、あまりにも効率が悪化した場合、リサンプリングによって重みの偏りを正常値に戻すという手法がリサンプリングである。

リサンプリングにもいくつか方法はあるが、代表的なものとしては、「 $\omega(z_{1:t}^{(s)})$  の確率で  $\{z_{1:t}^{(s)}\}_{s=1}^S$  から  $S$  個を復元抽出し新しい  $\{z_{1:t}^{(s)}\}$  とする」復元抽出法がある。復元抽出法を用いた粒子フィルタのアルゴリズムイメージを図 2 に示す。

<sup>3</sup>実際には log を取って更新していく方が（アンダーフローを避けるためにも）無難である。

<sup>4</sup>『トピックモデルによる統計的潜在意味解析』（佐藤 2015）には、重み付きサンプルによる事後分布はサンプル数に依存するが、現実的にはそれほど数を大きくすることができないためリサンプリングを行う、と書いてあるが、これはやや誤解を招く表現のように思われる。サンプル数に依存することや、粒子フィルタが常に計算コストとの兼ね合いを考えなければならない手法であることは間違いないが、リサンプリングを行う理由は、上述したようにサンプリング効率の改善である。伊庭ら (2005) などを参照するとよい。

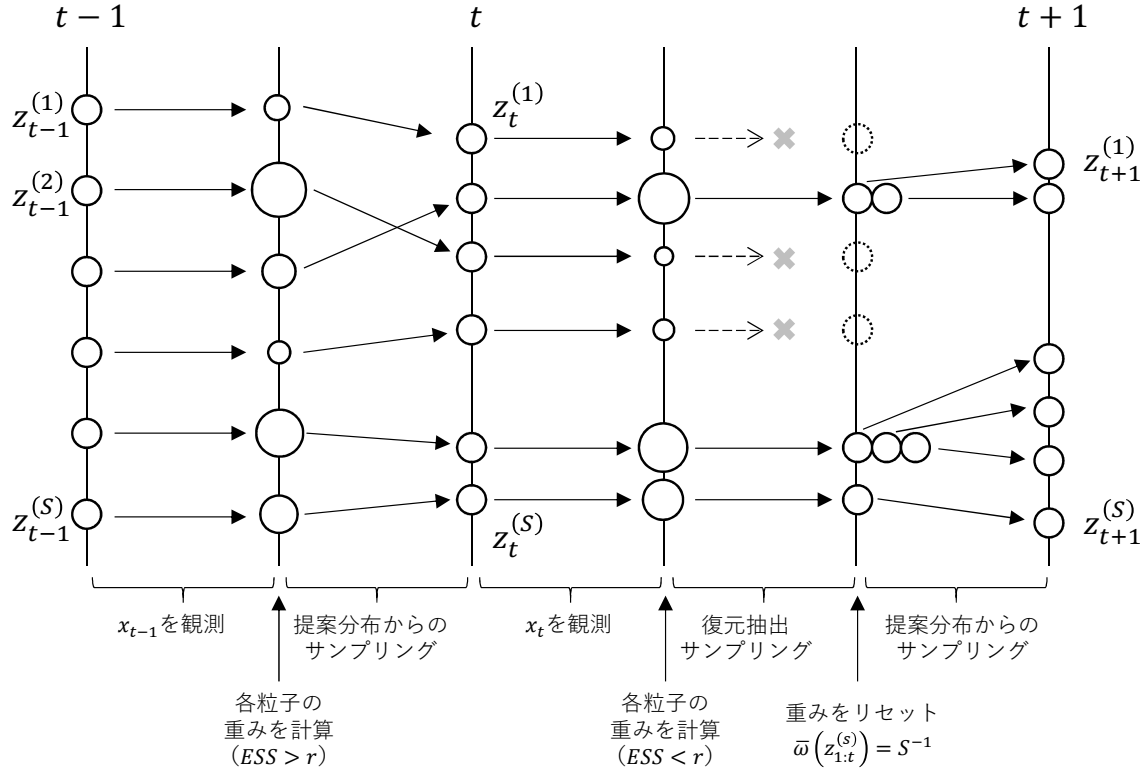


図 2: 粒子フィルタのアルゴリズムイメージ

粒子フィルタについて、伊庭ら (2005) に面白い例えがあったため紹介する（変数のアルファベットと添え字のみ本稿と合わせている）。

粒子フィルタのアルゴリズムは、データ  $x_t$  が「餌」、粒子  $z_t^{(s)}$  が「動物」だとすると理解しやすいかもしれない。 $t$  が増すごとに「動物」は遷移密度  $p(z_t^{(s)} | z_{t-1}^{(s)})$  に従ってランダムに移動し、餌  $x_t$  の近くにたまたま行けたものが、尤度  $p(x_t | z_t^{(s)})$  に比例して太ったり痩せたりする（動物たち全部の体重の和は一定とする）。太ったものは分裂して増え、痩せると死んでしまうというのが、リサンプリングである。

## 2 LDA の粒子フィルタ

### 2.1 重みの更新

LDA に粒子フィルタを適用するために、まず文書データを単語単位で時系列順に並べる。この時、時系列情報がない場合は、ランダムにシャッフルした単語に番号を付けるなどする。時系列のインデックスは  $i$  ( $i = 1, \dots, W$ ) ( $W$  は総単語数) とし、 $w_i, z_i, d_i$  はそれぞれ、 $i$  番目の単語、 $i$  番目の単語のトピック、 $i$  番目の単語に対応する文書である。ここでは、データが逐次的に手に入り、オンライン的に推定をする状況を想定しており、全く新しい文書が入ってくることもあれば、過去に処理した文書に新しい単語が追加されることもある。

$w^{(i)}, z^{(i)}$  を  $i$  番目までの全単語のベクトルとすると、 $w^{(i-1)}, z^{(i-1)}$  及び新しく観測した  $w_i$  が与え

られた下での、潜在変数の粒子  $z_i^{(s)}$  の分布は

$$p(z_i^{(s)} = k | w_i = v, w^{(i-1)}, z^{(i-1)(s)}, \alpha, \beta) \propto \left( N_{d_i k}^{(i-1)(s)} + \alpha_k \right) \frac{N_{kv}^{(i-1)(s)} + \beta}{\sum_v (N_{kv}^{(i-1)(s)} + \beta_v)} \quad (13)$$

である。ただし、 $N_{d_i k}^{(i-1)(s)}$ ,  $N_{kv}^{(i-1)(s)}$  は  $w^{(i-1)}$ ,  $z^{(i-1)}$  から計算される統計量である<sup>5</sup>。各粒子の重みの比は

$$\begin{aligned} \frac{\omega(z^{(i)(s)})}{\omega(z^{(i-1)(s)})} &\propto \frac{p(w_i | z_i^{(s)}, w^{(i-1)}, z^{(i-1)(s)}) p(z_i^{(s)} | w^{(i-1)}, z^{(i-1)(s)})}{q(z_i^{(s)} | w^{(i)}, z^{(i-1)(s)})} \\ &= \frac{p(w_i | z_i^{(s)}, w^{(i-1)}, z^{(i-1)(s)}) p(z_i^{(s)} | w^{(i-1)}, z^{(i-1)(s)})}{p(z_i^{(s)} | w^{(i)}, z^{(i-1)(s)})} \\ &= \frac{\frac{p(w_i, z_i^{(s)}, w^{(i-1)}, z^{(i-1)(s)})}{p(z_i^{(s)}, w^{(i-1)}, z^{(i-1)(s)})} \times p(z_i^{(s)} | w^{(i-1)}, z^{(i-1)(s)})}{\frac{p(z_i^{(s)}, w^{(i)}, z^{(i-1)(s)})}{p(w^{(i)}, z^{(i-1)(s)})}} \\ &= \frac{p(w^{(i)}, z^{(i-1)(s)})}{p(z_i^{(s)}, w^{(i-1)}, z^{(i-1)(s)})} \times \frac{p(w^{(i)}, z^{(i-1)(s)})}{p(w^{(i)}, z^{(i)(s)})} \times p(z_i^{(s)} | w^{(i-1)}, z^{(i-1)(s)}) \\ &= \frac{p(w^{(i)}, z^{(i-1)(s)}) p(z_i^{(s)} | w^{(i-1)}, z^{(i-1)(s)})}{p(z_i^{(s)} | w^{(i-1)}, z^{(i-1)(s)}) p(w^{(i-1)}, z^{(i-1)(s)})} \\ &= \frac{p(w_i, w^{(i-1)}, z^{(i-1)(s)})}{p(w^{(i-1)}, z^{(i-1)(s)})} \\ &= p(w_i | w^{(i-1)}, z^{(i-1)(s)}) \\ &= \sum_{k=1}^K p(w_i, z_i^{(s)} = k | w^{(i-1)}, z^{(i-1)(s)}) \end{aligned} \quad (14)$$

よって次の重み更新式を得る。

$$\omega(z^{(i)(s)}) \propto \omega(z^{(i-1)(s)}) \sum_{k=1}^K p(w_i, z_i^{(s)} = k | w^{(i-1)}, z^{(i-1)(s)}) \quad (15)$$

この重みを用いた重み付きサンプルによる事後分布の近似は、 $\bar{\omega}(z^{(i)(s)})$  を正規化した重みとして

$$p(z_i = k | w^{(i)}, z^{(i-1)}) \approx \sum_{s=1}^S \bar{\omega}(z^{(i)(s)}) \mathbb{I}(z_i^{(s)} = k) \quad (16)$$

で得られる。これは、前節でも説明した通り、粒子数  $S$  を限りなく大きくすれば、事後分布に分布収束することが保証されている。

## 2.2 活性化リサンプリング

リサンプリングについて、Canini et al. (2009) は、過去サンプルの活性化によるリサンプリングを提案している（手法の名前としては “Residual resampling” (Liu and Chen, 1998) である）。 $ESS < r$  になると、過去の観測データ  $w^{(i)}$  のなかから  $R$  個の活性化用サンプル  $w_j$  をランダムに選

<sup>5</sup>このように、パラメータを積分消去して周辺化事後分布  $p(z_i | w^{(i)}, z^{(i-1)})$  を近似するための粒子フィルタを “Rao-Blackwellized Particle filter” と呼ぶ

び、以下の条件付確率に従い  $z_j^{(s)}$  をリサンプリングする。

$$p(z_j^{(s)} = k | w_j = v, w_{\setminus j}^{(i)}, z_{\setminus j}^{(i)(s)}, \alpha, \beta) \propto \left( N_{d_j k \setminus j}^{(i)(s)} + \alpha_k \right) \frac{N_{kv \setminus j}^{(i)(s)} + \beta}{\sum_v (N_{kv \setminus j}^{(i)(s)} + \beta_v)} \quad (17)$$

Canini et al. (2009) の粒子フィルタによる LDA 推定アルゴリズムを Algorithm 1 に示す。

---

**Algorithm 1** 粒子フィルタによる LDA 推定

---

```

1: initialize weights  $\omega(z^{(0)(s)}) = S^{-1}$ ,  $s = 1, \dots, S$ 
2: for  $i = 1, \dots, W$  do
3:   for  $s = 1, \dots, S$  do
4:     update  $\omega(z^{(i)(s)})$  using (15)
5:     sample  $z_i^{(s)}$  from (13)
6:     increment  $N_{d_i k}^{(s)}, N_{kv}^{(s)}$  if  $z_i^{(s)} = k, w_i = v$ 
7:   end for
8:   normalize weights
9:   if  $ESS < r$  then
10:    resample particles  $z_j^{(s)}, \{z_j^{(s)}\} \rightarrow R(i)$ 
11:    for  $j$  in  $R(i)$  do
12:      for  $s = 1, \dots, S$  do
13:        decrement  $N_{d_j k}^{(s)}, N_{kv}^{(s)}$  if  $z_j^{(s)} = k, w_j = v$ 
14:        sample  $z_j^{(s)}$  from (17)
15:        increment  $N_{d_j k}^{(s)}, N_{kv}^{(s)}$  if  $z_j^{(s)} = k, w_j = v$ 
16:      end for
17:    end for
18:    set  $\omega(z^{(0)(s)}) = S^{-1}$ ,  $s = 1, \dots, S$ 
19:  end if
20: end for

```

---

リサンプリングに復元抽出法を用いない理由は、サンプルの多様性を確保するためである<sup>6</sup>。この詳細については、原論文や諸文献には特記されていないが、筆者の解釈としては次のようなものである。

時系列モデルのように、 $z_{t-1}$  が  $z_t$  へと遷移していくようなモデルであれば、現在の粒子のリサンプリングは次の世代に大きく影響する。しかし、LDA における  $z_i^{(s)}$  は  $z^{(i-1)(s)}$  で定まる事後分布からのサンプルであり、現在の世代の粒子  $z_i^{(s)}$  をリサンプリングしたとしても、数千、数万単位の単語に関する統計量  $N_{d_i k}^{(s)}, N_{kv}^{(s)}$  に与える影響はごくわずかである。それであれば、過去世代の粒子  $z_j^{(s)}$  を直接リサンプリングした方が、粒子の多様性が生まれ、非効率なサンプリングの改善につながる。

しかし、May et al. (2014) では、Canini et al. (2009) の活性化リサンプリングは、オンライン学習において重要な操作ではないとする検証結果を示しており、議論が分かれるところである<sup>7</sup>。

---

<sup>6</sup>佐藤 (2015) では、計算コストがかさむため復元抽出ではなく活性化リサンプリングを行うと述べているが、原論文を見てもこの解釈は間違っている。

<sup>7</sup>そもそも、LDA のオンライン学習は、Python にも実装されている Hoffman et al. (2010) のオンライン変分ペイズが主流であり、粒子フィルタに関する研究はそこまで進んでいない印象を受ける。一方で、粒子数を増やしていけば事後分布に分布収束するという性質は一つの長所であり、社会科学においては選択肢にも入りうる手法なのではないかと筆者は考える。

## Appendix

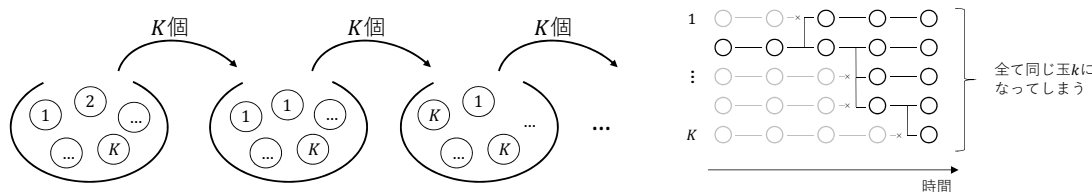


図 3: 復元抽出法の模式図

ここでは 1.3 節で述べたリサンプリングを不必要に行ってはいけない理由を簡単に説明する。数式的な詳細は Doucet et al. (2001) の 4.3.3 節などを参照するとよい。今、ある袋に  $K$  種類の玉が一個ずつ計  $K$  個入っているとす (図 3 の一番左)。これを初期状態として、「 $K$  個の玉を等確率で復元抽出する」操作を繰り返す (図 3 の左側)。すると、一度選ばれなかった玉は消滅するので、何回も繰り返すうちに (具体的には  $K$  の数倍程度の回数)、ある一つの玉に収束してしまう (図 3 の右側)。ここから得られる教訓は、「復元抽出を繰り返すと思ってもよらぬ結果につながる」である。

さらに、粒子フィルタにおけるリサンプリングは、重み付き確率での復元抽出であり、数回程度でこのような状況になっていしまう。そして最終的には、残ってほしい粒子までもが消滅してしまうのである。したがって、ESS の値を監視して、必要な時にだけリサンプリングを行うのが望ましい。

## 引用文献

- Canini, K., Shi, L., and Griffiths, T. (2009, April). Online inference of topics with latent Dirichlet allocation. *In Artificial Intelligence and Statistics*, 65-72.
- Doucet, A., de Freitas, N., and Gordon, N. (Eds.)(2000). *Sequential Monte Carlo methods in practice*, Springer.
- Hoffman, M., Bach, F. R., and Blei, D. M. (2010). Online learning for latent dirichlet allocation. *In advances in neural information processing systems*, 856-864.
- Kitagawa, G. (1998). A self-organizing state-space model. *Journal of the American Statistical Association*, 1203-1215.
- Liu, J. S. and Chen, R. (1998). Sequential Monte Carlo methods for dynamic systems. *Journal of the American statistical association*, 93(443), 1032-1044.
- 佐藤一誠. (2015). トピックモデルによる統計的潜在意味解析. コロナ社.
- 伊庭幸人, 種村正美, 大森裕浩, 和合肇, 佐藤整尚, 高橋明彦. (2005). 計算統計 II マルコフ連鎖モンテカルロ法とその周辺. 岩波書店.