Particle Filters

A. Giavaras

Contents

1 Particle Filters

Gaussian filtering is a useful technique in order to perform nonlinear filtering and we have seen the extended and unscented Kalman filtering. However, these methods do not perform well when

- The models are highly nonlinear
- When the posterior distribution is significantly non-Gaussian e.g. a multimodal density

For these kinds of problems we need a different type of approximation to the posterior density. One such approach is the particle filter method that we will discuss in this section. The idea behind this filtering technique is to aquire point-wise estimates, called samples, of the posterior $p(\mathbf{x}_k|\mathbf{y}_{1:k})$. Then, as the number of samples increases, the accuracy of the approximation increases.

Remark 1.1. Particle filters are also known as sequential importance resampling or sequential Monte Carlo. The basis of these methods is an algorithm called sequential importance sampling.

2 Introduction to Particle Filtering

The basic idea behind particle filtering is to use a non-parametric representation of the posterior

$$p(\mathbf{x}_k|\mathbf{y}_{1:k}) \approx \sum_{i=1}^{N} w_k^{(i)} \delta(\mathbf{x}_k - \mathbf{x}_k^{(i)}) \tag{1}$$

where $\mathbf{x}_k^{(i)}$ are particles and $w_k^{(i)}$ are associated weights.

We can then perform filtering by propagating $\mathbf{x}_k^{(i)}$ over time and updating the weights.

2.1 Monte Carlo sampling

Given independent samples $\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \dots, \mathbf{x}^{(N)} \sim p(\mathbf{x})$ we can approximate

$$E[\mathbf{g}(\mathbf{x})] \approx \frac{1}{N} \sum_{i=1}^{N} \mathbf{g}(\mathbf{x}^{(i)}) \tag{2}$$

$$p(\mathbf{x}) \approx \frac{1}{N} \sum_{i=1}^{N} \delta(\mathbf{x} - \mathbf{x}^{(i)}) \tag{3}$$

Monte Carlo approximations have the following characteristics

- Non-parametric approximation to $p(\mathbf{x})$
- Approximates all kinds of densities $p(\mathbf{x})$
- Does not suffer from the curse of dimensionality

Remark 2.1. **Curse of Dimensionality**

However, we should note that it is often difficult to generate samples from $p(\mathbf{x})$

2.2 Importance sampling

Importance sampling can be used when it is difficult to sample from $p(\mathbf{x})$. Importance sampling generates samples $\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \ldots, \mathbf{x}^{(N)}$ from a a proposal density $q(\mathbf{x})$. We can then set $p(\mathbf{x})$ as

$$p(\mathbf{x}) \approx \sum_{i=1}^{N} w^{(i)} \delta(\mathbf{x} - \mathbf{x}^{(i)}) \tag{4}$$

where the weights are given by

$$w^{(i)} = \frac{\tilde{w}^{(i)}}{\sum_{n=1}^{N} \tilde{w}^{(i)}}, \quad \tilde{w}^{(i)} = \frac{p(\mathbf{x}^{(i)})}{q(\mathbf{x}^{(i)})} \tag{5}$$

Importance sampling is a flexible and powerful tool. It performs well as long as

- It is easy to sample from $q(\mathbf{x})$
- The support of $q(\mathbf{x})$ contains the support of $p(\mathbf{x})$
- $q(\mathbf{x})$ is similar to $p(\mathbf{x})$

2.3 Sequential Importance Sampling (SIS)

Recall that our goal is to recursively and accurately approximate the filtering density $p(\mathbf{x}_k|\mathbf{y}_{1:k})$. In order to do so we assumed that both the motion and measurement models, i.e. $p(\mathbf{x}_k|\mathbf{x}_{k-1})$ and $p(\mathbf{y}_k|\mathbf{x}_k)$ respectively, can be evaluated point-wise. For example, for the model

$$\mathbf{x}_k = f(\mathbf{x}_{k-1}) + \mathbf{q}_{k-1}, \quad \mathbf{q}_{k-1} \sim N(\mathbf{0}, \mathbf{Q}_{k-1}) \tag{6}$$

$$\mathbf{y}_k = h(\mathbf{x}_{k-1}) + \mathbf{r}_{k-1}, \quad \mathbf{r}_{k-1} \sim N(\mathbf{0}, \mathbf{R}_{k-1}) \tag{7}$$

then the density

$$p(\mathbf{x}_k|\mathbf{x}_{k-1}) = N(\mathbf{x}_k; f(\mathbf{x}_{k-1}), \mathbf{Q}_{k-1}) \tag{8}$$

is generally easy to be evaluated for any values of $\mathbf{x}_k$ and $\mathbf{x}_{k-1}$.

Particle filters are also known as sequential importance resampling or sequential Monte Carlo. The basis of these methods is an algorithm called sequential importance sampling. The standard SIS algorithm is outlined below

Remark 2.2. **Standard SIS Algorithm**

1. For $i = 1, \ldots, N$ at each time k do:
   - Draw $\mathbf{x}_k^{(i)} \sim q(\mathbf{x}_k|\mathbf{x}_{k-1}^{(i)}, \mathbf{y}_k)$
   - Compute weights

$$w_k^{(i)} \propto w_{k-1}^{(i)} \frac{p(\mathbf{y}_k|\mathbf{x}_k^{(i)})q(\mathbf{x}_k|\mathbf{x}_{k-1}^{(i)})}{q(\mathbf{x}_k|\mathbf{x}_{k-1}^{(i)}, \mathbf{y}_k)} \tag{9}$$

   - Normalize the weights
2. Approximate

$$p(\mathbf{x}_k|\mathbf{y}_{1:k}) \approx \sum_{i=1}^{N} w_k^{(i)} \delta(\mathbf{x}_k - \mathbf{x}_k^{(i)}) \tag{10}$$

Now that we have a method to approximate $p(\mathbf{x}_k|\mathbf{y}_{1:k})$ we ask ourselves what is the MMSE estimate of $\mathbf{x}_k$. We can calculate the MMSE estimate from this approximation via

$$\hat{\mathbf{x}}_k = \sum_{i}^{N} w_k^{(i)} \mathbf{x}_k^{(i)} \tag{11}$$

We can view the particle filter representation as approximating the posterior PDF with a posterior PMF where the weight, $w_k^{(i)}$, gives us the discrete probability that the state is $\mathbf{x}_k^{(i)}$, Using the definition of expected value on this PMF gives us the solution above.

Example 2.1. Nonlinear Filter Benchmark

We will demonstrate particle filtering using the following benchmark for nonlinear filters

$$x_k = \frac{x_{k-1}}{2} + \frac{25x_{k-1}}{1+x_{k-1}^2} + 8\cos(1.2k) + q_{k-1} \tag{12}$$

$$y_k = \frac{x_k^2}{20} + r_k \tag{13}$$

where $q_{k-1} \sim N(0, 10)$ and $r_k \sim N(0, 1)$.

2.4 Questions

1. Which of the following statements, regarding the usefulness of particle filters, are true?
   (a) Particle filters are useful as they can handle almost any models
   (b) Particle filters are useful as within reasonable computational complexity, the particle filter always gives us the best performance.
   (c) Particle filters are useful as they give us a compact description of the posterior density.
   (d) Particle filters are useful as they give a non-parametric description of the posterior.
2. Assuming that we describe our posterior using the following approximation

$$p(\mathbf{x}_k|\mathbf{y}_{1:k}) \approx \sum_{i=1}^{N} w_k^{(i)} \delta(\mathbf{x}_k - \mathbf{x}_k^{(i)}) \tag{14}$$

   what is the MMSE estimate of $\mathbf{x}_k$?
   (a) $\hat{\mathbf{x}}_k = \sum_i^N w_k^{(i)} \mathbf{x}_k^{(i)}$
   (b) $\hat{\mathbf{x}}_k = \mathbf{x}_k^{(j)}$ where $j = argmax_i w_k^{(i)}$
   (c) $\hat{\mathbf{x}}_k = \frac{1}{N} \sum_i^N \mathbf{x}_k^{(i)}$

(d) It is not possible to calculate a MMSE estimate from this approximation.

3. Consider Figure 2.

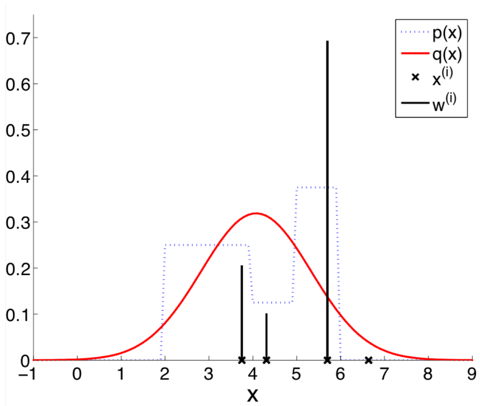
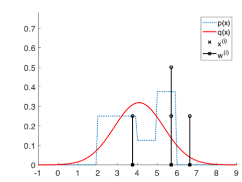
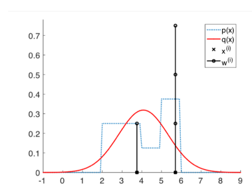
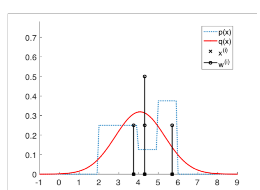


Fig. 1: Question figure.

Perform resampling on the density and illustrate the result. Assume that the numbers 0.65, 0.03, 0.84 and 0.93 are drawn uniformly from $[0, 1]$. Which one of the following figures illustrates the resampled particles?

(a) 

(b) 

(c) 

3 Answers

1. Which of the following statements, regarding the usefulness of particle filters, are true?
   (a) Particle filters are useful as they can handle almost any models
   (b) Particle filters are useful as within reasonable computational complexity, the particle filter always gives us the best performance.
   (c) Particle filters are useful as they give us a compact description of the posterior density.
   (d) Particle filters are useful as they give a non-parametric description of the posterior.

   Answer:

   The main advantage with particle filters is that they offer a non-parametric description of our posterior. As a consequence of this, it can be used to handle almost any model as long as we can evaluate them point-wise. However, it is hard to argue that we get a compact description of our posterior as we typically use a couple of thousand parameters or even more. Further, it is not a solution that should be used to solve all problems, for example, if we have linear and Gaussian models we should use the ordinary Kalman filter we know that this gives us the optimal solution. So correct options are A and D.

2. Assuming that we describe our posterior using the following approximation

$$p(\mathbf{x}_k|\mathbf{y}_{1:k}) \approx \sum_{i=1}^{N} w_k^{(i)} \delta(\mathbf{x}_k - \mathbf{x}_k^{(i)}) \qquad (15)$$

   what is the MMSE estimate of $\mathbf{x}_k$?
   (a) $\hat{\mathbf{x}}_k = \sum_i^N w_k^{(i)} \mathbf{x}_k^{(i)}$
   (b) $\hat{\mathbf{x}}_k = \mathbf{x}_k^{(j)}$ where $j = argmax_i w_k^{(i)}$
   (c) $\hat{\mathbf{x}}_k = \frac{1}{N} \sum_i^N \mathbf{x}_k^{(i)}$
   (d) It is not possible to calculate a MMSE estimate from this approximation.

   Answer:

   We can view the particle filter representation as approximating the posterior PDF with a posterior PMF where the weight, $w_k^{(i)}$, gives us the discrete probability that the state is $\mathbf{x}_k^{(i)}$, Using the definition of expected value on this PMF gives us the solution above. So the correct answer is option A.

3. Consider Figure 2.

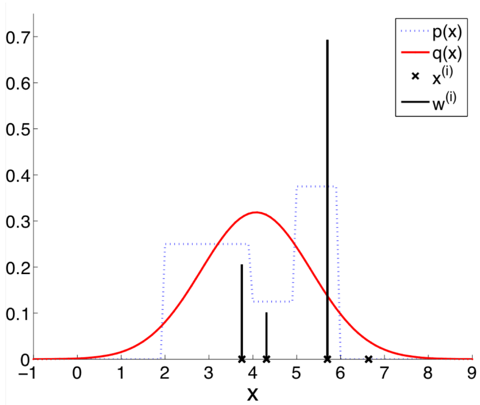
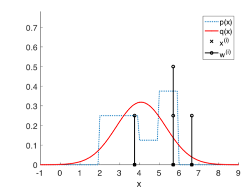
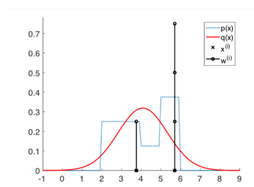
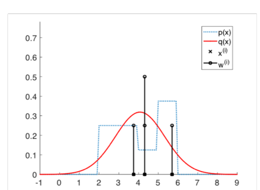


Fig. 2: Question figure.

Perform resampling on the density and illustrate the result. Assume that the numbers 0.65, 0.03, 0.84 and 0.93 are drawn uniformly from $[0, 1]$. Which one of the following figures illustrates the resampled particles?

(a) 

(b) 

(c) 

Answer:

If particles are ordered in ascending order, $x^{(1)} < \ldots < x^{(4)}$ resampling gives the following $x^{(1)} = 3.8$ and $x^{(2)} = x^{(3)} = x^{(4)} = 5.7$. Thus, option B is the correct answer.

References

[1] Åström K. J., Murray R. M. *Feedback Systems. An Introduction for Scientists and Engineers*

[2] Philip , Florent Altche1, Brigitte dAndrea-Novel, and Arnaud de La Fortelle *The Kinematic Bicycle Model: a Consistent Model for Planning Feasible Trajectories for Autonomous Vehicles?* HAL Id: hal-01520869, https://hal-polytechnique.archives-ouvertes.fr/hal-01520869

[3] Marcos R. O., A. Maximo *Model Predictive Controller for Trajectory Tracking by Differential Drive Robot with Actuation constraints*