Introduction to Particle Filtering

Jose Franco UDRC Summer School, Jun. 2016

Motivation

We are interested in the estimation of the state of a signal which evolves through time. For illustrative purposes, we will focus on a simple **target tracking** example, where we will track an object which moves in one dimension with constant velocity. We are interested in solving the Bayesian filtering problem in such a way that we are **not constrained** by our model choice - We want to be able to use non-linear models, and to be able to represent arbitrary probability distributions. In this lecture we will present a filter that solves this problem in a probabilistic way using randomness to our advantage.

Estimation

Our goal is to represent these signals in a probabilistic way, such that we consider the associated uncertainty and can make informed decisions according to our confidence in the process of interest. Rather than a point estimate, we are interested in how likely is it for the signal to take different values in its state space.

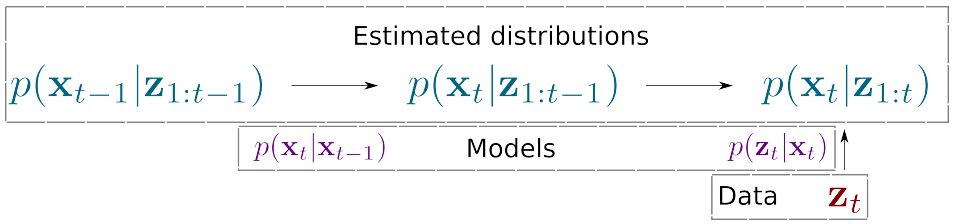
Our framework

We will focus on the estimation of signals that evolve sequentially through time, and which are observed through a sensor which does not necessarily fully describe the signal. These processes can be modelled under certain conditions as Partially Observed Markov Processes.

Modelling

We need to carefully model three things:

- What is our initial knowledge about the system state? (**Prior**)
- To the best of our knowledge, how does the state of our process evolve through time? (**Transition kernel**)
- How does the sensor work - i.e., how does it represent our process of interest, and how precise is it? (**Measurement model**)



Dynamical model

We consider as the state a vector $\mathbf{x}_t$ composed of the variables that interest us from the process we are studying at a given time step t. We assume that we have some knowledge of how the object evolves through time, based only on its previous state (Markov assumption), an amount of noise, and possibly an external, known perturbation $\mathbf{u}_t$, which we will not consider in this lecture.

Measurement model

Additionally, we receive measurements which represent observations of the system. Our measurement model indicates how the state $\mathbf{x_t}$ is transformed into a measurement $\mathbf{z_t}$ by the sensor, dependent on its particular characteristics and a noise variable.

The estimation problem

Armed with our models, we are left with the task of **recursively** estimating the **probability distributions** of the state random variable as time advances and we potentially receive measurements. We will formulate this process from a **Bayesian** point of view.

Bayesian Estimation

Our goal is to use our model:

- $p(\mathbf{x}_0)$, the *Prior* distribution of the process,
- $p(\mathbf{x}_t|\mathbf{x}_{t-1})$, the Markov transition kernel, and
- $p(\mathbf{z}_t|\mathbf{x}_t)$, the measurement likelihood,

to estimate the posterior probability distributions

$$p(\mathbf{x}_t|\mathbf{z}_{1:t})$$

Which summarise all the information that we know about the random variable, given the sequence of measurements we have received up to time t.

Prediction

After a time step passes, we can compute the predicted distribution using the Chapman-Kolmogorov equation,

$$p(\mathbf{x}_t|\mathbf{z}_{1:t-1}) = \int p(\mathbf{x}_{t-1}|\mathbf{z}_{1:t-1})p(\mathbf{x}_t|\mathbf{x}_{t-1})\,\mathrm{d}\mathbf{x}_{t-1}$$

Update

When a measurement $\mathbf{z}_t$ is received, we apply Bayes' rule to integrate it into the prior:

$$p(\mathbf{x}_t|\mathbf{z}_{1:t}) = \frac{p(\mathbf{x}_t|\mathbf{z}_{1:t-1})p(\mathbf{z}_t|\mathbf{x}_t)}{\int p(\mathbf{x}'_t|\mathbf{z}_{1:t-1})p(\mathbf{z}_t|\mathbf{x}'_t)\,\mathrm{d}\mathbf{x}'_t}$$

An example

We know previously that our target exists in the interval $[0, 10]$, our *surveillance region* (all units in metres). It moves at a constant rate, although external factors or a decision that it takes can possibly change its movement rate. Finally, we observe it with a low precision sensor which only tells us which metre the target is currently, and nothing about its velocity. Our model is the following:

- $\mathbf{x}_t = [x_t, \dot{x}_t]'$
- $p(\mathbf{x}_0) = \mathcal{U}(0, 10)$
- $p(\mathbf{x}_t|\mathbf{x}_{t-1}) = \mathcal{N}(\mathbf{x}_t; (x_{t-1} + \Delta_t\dot{x}_{t-1}, \dot{x}_{t-1}), \mathbf{Q}_t)$
- $p(\mathbf{z}_t|\mathbf{x}_t) = 0.1\mathcal{U}(0, 10) + 0.9\mathcal{U}(\lfloor x_t \rfloor, \lceil x_t \rceil)$

Remarks

Computing these integrals requires choosing a way to represent the probability distributions: - If the prior, Markov transition and likelihood functions are all linear and Gaussian, then the Kalman filter is the optimal solution - The integrals may be solved numerically over a grid, which works well for small state-spaces - Otherwise, the integrals may be approximated using Sequential Monte Carlo (SMC) methods

SMC Integration

Say we want to compute the expected value of a given function $f(\mathbf{x})$, and we can simulate N i.i.d. samples from $\mathbf{x}$, $x^{(i)} \sim p(\mathbf{x})$. We can approximate $p(\mathbf{x}) \approx P^N(\mathbf{x}) = \frac{1}{N}\sum_{i=1}^{N}\delta(\mathbf{x} - x^{(i)})$, and our expected value becomes

$$\mathrm{E}_{P^N}[f] = \int f(\mathbf{x})\frac{1}{N}\sum_{i=1}^{N}\delta(\mathbf{x} - x^{(i)})\,\mathrm{d}\mathbf{x} = \frac{1}{N}\sum_{i=1}^{N} f(x^{(i)})$$

This estimate is unbiased, and converges almost surely to the expected value of the original distribution.

Sampling

Sampling from these distributions is not trivial, however, as we do not have a closed form solution for them and they are only known up to a proportionality constant. An alternative would be to use MCMC or Gibbs samplers, but our sequential estimation problem lends itself better to use a Sequential Monte Carlo approach.

Importance sampling

A 'trick' may be used where we can sample from an alternative distribution $\pi(\mathbf{x})$ which is easier to sample from in a given sense, and then correct for the introduced bias to compute our expected values. By applying a small substitution, we get

$$\begin{aligned}\mathrm{E}[f] &= \int f(\mathbf{x})p(\mathbf{x})\,\mathrm{d}\mathbf{x} \\ &= \frac{\int f(\mathbf{x})\pi(\mathbf{x})w(\mathbf{x})\,\mathrm{d}\mathbf{x}}{\int \pi(\mathbf{x})w(\mathbf{x})\,\mathrm{d}\mathbf{x}}\end{aligned}$$

With $w(\mathbf{x}) = \frac{p(\mathbf{x})}{\pi(\mathbf{x})}$. The only requirement on $\pi(\mathbf{x})$ is that its support must coincide with that of $p(\mathbf{x})$. This way, we can compute an approximation of our expected value using samples from $\pi(\mathbf{x})$:

$$\begin{aligned}\mathrm{E}[f] &\approx \frac{\frac{1}{N}\sum_{i=1}^{N}\tilde{w}_i f(x^{(i)})}{\frac{1}{N}\sum_{i=1}^{N}\tilde{w}_i} \\ &= \sum w_i f(x^{(i)})\end{aligned}$$

$$\tilde{w}_i = \frac{p(x^{(i)})}{\pi(x^{(i)})} \qquad w_i = \frac{\tilde{w}_i}{\sum_{i=1}^{N}\tilde{w}_i}$$

Another interpretation of this is that the original probability distribution is being approximated as a weighted sum of samples, independently of f. It is important to select the importance function to be as close to our desired probability distribution as possible in order to obtain good quality estimates.

Sequential Importance Sampling

If the importance sampling functions are selected so that they are conditional on one another, we avoid the computational cost of recomputing the weights over the whole state sequence each time a new measurement is received:

$$\pi(\mathbf{x}_{0:t}|\mathbf{z}_{1:t}) = \pi(\mathbf{x}_0)\prod_{k=1}^{t}\pi(\mathbf{x}_k|\mathbf{x}_{0:k-1}, \mathbf{z}_{1:k})$$

Since the state trajectories are preserved, this allows us to update recursively the importance weights:

$$w_t^{(i)} \propto w_{t-1}^{(i)}\frac{p(\mathbf{z}_t|x_t^{(i)})p(x_t^{(i)}|x_{t-1}^{(i)})}{\pi(x_{0:t-1}^{(i)}|\mathbf{z}_{1:t})}$$

If at each time step we sample from the transition kernel, the importance weights take an interesting form:

$$\pi(\mathbf{x}_t) = p(\mathbf{x}_0)\prod_{k=1}^{t}p(\mathbf{x}_k|\mathbf{x}_{k-1})$$

$$w_t^{(i)} \propto w_{t-1}^{(i)} p(z_t|x_t)$$

Sampling from the prior distribution is usually feasible as it is required in order to predict the posterior distributions, and it guides our filter to regions that are likely according to the transition kernel. For each sample, a new sample is generated which represents the latest point in the simulated state path, and is weighted according to its observation likelihood.

Resampling

During the filtering process, some particles will usually accumulate most of the weight of the sample as they agree more with the received measurements. This is problematic since it means that some particles will have very low weights and will have negligible effects on the filtering distributions. A solution is to replace the set of particles by one where all the weights are equal, by **resampling** from the discrete distribution of particles in order to obtain an equally weighted distribution. If we start with the weighted approximation

$$\hat{P}^N(\mathbf{x}) = \sum_{i=1}^{N} w_i \delta(\mathbf{x} - \hat{x}^{(i)})$$

We can replace it by the unweighted approximation

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

Where now all the weights are equal, and the probability of selecting each sample in the new approximation is equal to its weight in the original sample:

$$p(x^{(i)} = \hat{x}^{(j)}) = w_j, \quad i, j = 1...N$$

In the resampled approximation, each sample contributes equally to the representation.

The bootstrap filter

The basic idea: Use random grids to approximate the distribution of interest, and Monte Carlo integration to compute statistics of interest Initialisation: Sample $\{x_0^{(i)}\}_{i=1}^N$
Importance sampling step: for $i = 1, ..., N$ sample $\tilde{x}_t^{(i)} \sim p(\mathbf{x}_t|x_{t-1}^{(i)})$ and compute the importance weights

$$w_t^{(i)} = \frac{\tilde{w}_t^{(i)}}{\sum_{i=1}^{N} \tilde{w}_t^{(i)}} \qquad \tilde{w}_t^{(i)} = w_{t-1}^{(i)} p(z_t|x_t^{(i)})$$

Selection step: Sample $x_t^{(i)}$ from $\tilde{x}_t^{(i)}$ with probability given by the importance weights
The selection step is important as it focuses the attention of the filter in the regions with higher probability

Summary

The sequential Monte Carlo estimation framework was discussed in the recursive Bayesian estimation context. The bootstrap filter was introduced, which is flexible and easy to implement.

References

[1] A. Doucet, N. de Freitas, and N. Gordon, editors. *Sequential Monte Carlo Methods in Practice*, volume 1. Springer-Verlag, 2001.

[2] Neil Gordon, David Salmond, and Adrian Smith. Novel approach to nonlinear non-Gaussian Bayesian state estimation. *IEE Proceedings F*, 1993.