Basic Probability and Distributions
Sampling, Tracking
Tracking via the Particle Filter

CMSC 828D

Fall 2000

Probability notation and definitions

- D set of all events, Null event $\emptyset$
- Probability of an event A occurring *P(A)*
- $P(D) = 1$
- $P(\emptyset) = 0$
- for any A, $0 \le P(A) \le 1$
- if $A \subset B$, then $P(A) \le P(B)$
- $P(A \cup B) = P(A) + P(B) - P(A \cap B)$

  Probability of either of two events occurring

- Probability of both events occurring *P(A,B)*

  $P(A \cap B) = P(A \mid B)P(B) = P(B \mid A)P(A)$
- Leads directly to Bayes Rule $P(A \mid B) = \dfrac{P(B \mid A)P(A)}{P(B)}$
  - Way to transfer conditional probabilities
- Bayesian Inference
- Independence of two events A and B

  P(A|B)=P(A)P(B)
- Conditional independence

  P(A,B|C)=P(A|C)P(B|C)

Probability Distributions

- Instead of single events we look at now a large collection of events.
- Assume that these events can be characterized by a number
- "take to the limit" and look at values of probability for values of x along the real line
- probabilities associated with x taking on a range of values.

  [a,b] (a,b] (-∞, ∞) etc.
- Convenient to look at two distribution functions

  probability density function $P(a < x < b) = \int_a^b p(x)dx$

  cumulative density function $F(a) = \int_{-\infty}^{a} p(x)dx = P(-\infty < x \le a)$

- For continuous density functions *P(x=a)* =0
- Example density function: gaussian $N(\mu,\sigma) = \dfrac{1}{\sqrt{2\pi}\sigma}\exp\left(-\dfrac{1}{2}\left(\dfrac{x-\mu}{\sigma}\right)^2\right)$

Working with distributions

- *E(x)* is the expected value of a random variable

  $\mathrm{E}[x] = \sum_{i \in \text{values}} x_i p(x_i)$ $\quad \mathrm{E}[x] = \int_D x p(x) dx$ $\quad \mathrm{E}(g(x)) = \int_{-\infty}^{\infty} g(x)p(x)dx$
- E(x) is nothing but the mean or average of x
- Variance $\mathrm{var}(x) = \mathrm{E}[x^2 - (\mathrm{E}(x))^2]$
- Variance is the difference between the expected value of the square and E(x)^2.
- Estimates departures from the mean $\mathrm{var}(x) = \int_{-\infty}^{\infty} x^2 p(x)dx - \left[\int_{-\infty}^{\infty} xp(x)dx\right]^2$
- Knowing the distribution and how to integrate functions of x with respect to it we can compute probabilities
- Sampling techniques -- attempt to compute probabilities by approximating the integral.
- Use known values at a few sample points.

Computing expectations with samples

- Distribution is a device to compute expectations
- Given a distribution of points $\boldsymbol{u}^i$ and a distribution on these points $f(\boldsymbol{u}^i)$

Represent a probability distribution

$$p_f(\boldsymbol{X}) = \frac{f(\boldsymbol{X})}{\int f(\boldsymbol{U})d\boldsymbol{U}}$$

by a set of N weighted samples

$$\{(\boldsymbol{u}^i, w^i)\}$$

where $\boldsymbol{u}^i \sim s(\boldsymbol{u})$ and $w^i = f(\boldsymbol{u}^i)/s(\boldsymbol{u}^i)$.

- Compute expectations using the sample points and weights $\displaystyle\int g(\boldsymbol{U})p_f(\boldsymbol{U})d\boldsymbol{U} \approx \frac{\sum_{i=1}^{N} g(\boldsymbol{u}^i)w^i}{\sum_{i=1}^{N} w^i}$

Sampling

- Basic problem for Monte-Carlo Methods
  - Integrate a function *f* over a region of volume *V*
  - Integral may be hard to calculate because
    - the function is not known explicitly,
    - region over which the integral is to be taken cannot be characterized
    - Integral is over many dimensions (e.g. 100s)
  - Approximate integral somehow
- Von Neumann while working on the Manhattan project, approximated integral as

$$\int f\, dV \approx V\langle f\rangle \pm V\sqrt{\frac{\langle f^2\rangle - \langle f\rangle^2}{N}}$$

$$\langle f\rangle \equiv \frac{1}{N}\sum_{i=1}^{N} f(x_i) \qquad \langle f^2\rangle \equiv \frac{1}{N}\sum_{i=1}^{N} f^2(x_i)$$

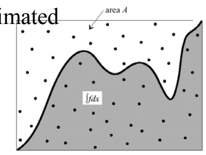


Figure 7.6.1. Monte Carlo integration. Random points are chosen within the area A. The integral of the function f is estimated as the area of A multiplied by the fraction of random points that fall below the curve f. Refinements on this procedure can improve the accuracy of the method; see text.

Bayesian Inference

- Convert the simple Bayes formula into a powerful way to look at any new piece of information.
- Probabilistic model with some parameters
- Fixing parameters allows predicting the probabilities of events. Can calculate *P*(measurements|parameters)
- **Prior:** We have an estimate of *P*(parameters)
- **Posterior:** Given measurements, we want to update our estimate of the parameters. *P*(parameters|measurements)
- Bayesian inference formula is

$$P(\text{parameters}|\text{measurements}) = \frac{P(\text{measurements}|\text{parameters})P(\text{parameters})}{P(\text{measurements})}$$

Tracking

- Components:
  - a motion model that predicts the new state of the system.
    - Allows one to predict $\mathbf{y}_i$
  - Measurement $\mathbf{y}_i = f(\mathbf{y}_{i-1}) + \mathbf{w}_{i-1}$
    - Measure things that can also be predicted by your model
    - E.g. position of a point, or some other quantity
    - Measurement satisfies equation
  - Use Bayesian framework $\mathbf{x}_i = g(\mathbf{y}_i) + \mathbf{v}_i$
  - Estimate posterior distribution of $\mathbf{y}_i$
- When equations were linear and noise models were Gaussian, the Kalman filter applies
- When equations are nonlinear and noise is Gaussian we can use the Extended Kalman filter
- Another approach is to use sampling

Tracking as inference

- Given an estimate of parameters at the new step as **y**
- Use measurement and Bayes rule to improve the estimate

$$P(\mathbf{y}\,|\,\mathbf{x}) = \frac{P(\mathbf{x}\,|\,\mathbf{y})P(\mathbf{y})}{P(\mathbf{x})} = \frac{P(\mathbf{x}\,|\,\mathbf{y})P(\mathbf{y})}{\int P(\mathbf{x}\,|\,\mathbf{y})d\mathbf{y}}$$

- Denominator only depends upon the data, and not our estimates of **y**
- Thus it is constant w. r. to **y** and we can write

$$P(\mathbf{y}\,|\,\mathbf{x}) \propto P(\mathbf{x}\,|\,\mathbf{y})P(\mathbf{y})$$

- Often evaluating the denominator is hard and this proportionality equation is used

Representing the posterior using samples

- Bayes rule (again)

$$p(\boldsymbol{U}|\boldsymbol{V}=v_0) = \frac{p(\boldsymbol{V}=\boldsymbol{v}_0|\boldsymbol{U})p(\boldsymbol{U})}{\int p(\boldsymbol{V}=\boldsymbol{v}_0|\boldsymbol{U})p(\boldsymbol{U})d\boldsymbol{U}} = \frac{1}{K}p(\boldsymbol{V}=\boldsymbol{v}_0|\boldsymbol{U})p(\boldsymbol{U})$$

- Evaluating *K*

$$K = \int p(\boldsymbol{V}=\boldsymbol{v}_0|\boldsymbol{U})p(\boldsymbol{U})d\boldsymbol{U} = \mathrm{E}\left[\frac{\sum_{i=1}^{N} p(\boldsymbol{V}=\boldsymbol{v}_0|\boldsymbol{u}^i)w^i}{\sum_{i=1}^{N} w^i}\right] \approx \frac{\sum_{i=1}^{N} p(\boldsymbol{V}=\boldsymbol{v}_0|\boldsymbol{u}^i)w^i}{\sum_{i=1}^{N} w^i}$$

$$\int g(\boldsymbol{U})p(\boldsymbol{U}|\boldsymbol{V}=v_0)d\boldsymbol{U} = \frac{1}{K}\int g(\boldsymbol{U})p(\boldsymbol{V}=\boldsymbol{v}_0|\boldsymbol{U})p(\boldsymbol{U})d\boldsymbol{U} \approx \frac{1}{K}\frac{\sum_{i=1}^{N} g(\boldsymbol{u}^i)p(\boldsymbol{V}=\boldsymbol{v}_0|\boldsymbol{u}^i)w^i}{\sum_{i=1}^{N} w^i} \approx \frac{\sum_{i=1}^{N} g(\boldsymbol{u}^i)p(\boldsymbol{V}=\boldsymbol{v}_0|\boldsymbol{u}^i)w^i}{\sum_{i=1}^{N} p(\boldsymbol{V}=\boldsymbol{v}_0|\boldsymbol{u}^i)w^i}$$

- Evaluate the posterior
- Equiv. to computing E with weight $w'^i = p(\boldsymbol{V}=\boldsymbol{v}_0|\boldsymbol{u}^i)w^i$

Resampling

- Original points may not sample the posterior well
- Resample … distribute points according to the pdf of the posterior and compute new points $\mathbf{u}_j$ and weights w_j

Algorithm

- Initialize
- Predict using the motion model
- Measure
- Use measurements to obtain new weights
- Resample to generate new points and new weights
- Loop

Algorithm

Initialization: Represent $P(\boldsymbol{X}_0)$ by a set of N samples

$$\left\{(\boldsymbol{s}_0^{k,-}, w_0^{k,-})\right\}$$

where

$$\boldsymbol{s}_0^{k,-} \sim P_s(\boldsymbol{S})$$

and

$$w_0^{k,-} = P(\boldsymbol{s}_0^{k,-})/P_s(\boldsymbol{S} = \boldsymbol{s}_0^{k,-})$$

Ideally, $P(\boldsymbol{X}_0)$ has a simple form and $\boldsymbol{s}_0^{k,-} \sim P(\boldsymbol{X}_0)$ and $w_0^{k,-} = 1$.

Prediction: Represent $P(\boldsymbol{X}_i|\boldsymbol{y}_0, \boldsymbol{y}_{i-1})$ by

$$\left\{(\boldsymbol{s}_i^{k,-}, w_i^{k,-})\right\}$$

where

$$\boldsymbol{s}_i^{k,-} = f(\boldsymbol{s}_{i-1}^{k,+}) + \xi_i^k$$

Algorithm - 2

Correction: Represent $P(\boldsymbol{X}_i|\boldsymbol{y}_0, \boldsymbol{y}_i)$ by

$$\left\{(\boldsymbol{s}_i^{k,+}, w_i^{k,+})\right\}$$

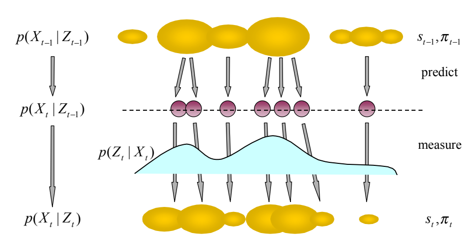
where

$$\begin{aligned} \boldsymbol{s}_i^{k,+} &= \boldsymbol{s}_i^{k,-} \\ w_i^{k,+} &= P(\boldsymbol{Y}_i = \boldsymbol{y}_i|\boldsymbol{X}_i = \boldsymbol{s}_i^{k,-})w_i^{k,-} \end{aligned}$$

Resampling: Normalise the weights so that $\sum_i w_i^{k,+} = 1$ and compute the variance of the normalised weights. If this variance exceeds some threshold, then construct a new set of samples by drawing, with replacement, N samples from the old set, using the weights as the probability that a sample will be drawn. The weight of each sample is now $1/N$.

Algorithm 19.8: *A practical particle filter resamples the posterior.*

The Condensation algorithm



Improving the algorithm

- Make the distribution of sample points "better"
- Recall error estimate of MC method

$$\int f\, dV \approx V\langle f\rangle \pm V\sqrt{\frac{\langle f^2\rangle - \langle f\rangle^2}{N}}$$

$$\langle f\rangle \equiv \frac{1}{N}\sum_{i=1}^{N} f(x_i) \qquad \langle f^2\rangle \equiv \frac{1}{N}\sum_{i=1}^{N} f^2(x_i)$$

- Error can be reduced by
  - Increasing N
  - Reducing variance of f computed on the sampled points
  - Using deterministic sets of points called quasi-random points to do the sampling.

Conventional tracking algorithms

- Assume image motion model (e.g., affine)
- Compute flow for patches
- Obtain parameters of the transformation for patches
- Track …
- Not very robust … but could be important for applications.
- J. Shi and C. Tomasi. Good Features to Track. IEEE Conference on Computer Vision and Pattern Recognition, June 1994, pp. 593-600