Chapter 11

Further topics

11.1 Computing the distribution - CDF's

Until now we have focused on computing the mean $\mu = E[X]$ of the RV X. Often we want to know the distribution of the RV. Both direct Monte Carlo and MCMC generates samples of X, and so give us information on the distribution. Most of this section will consider the case where X is a real-valued random variable, not a random vector. We briefly consider random vectors at the end of this section.

In this section we are only concerned with a non-parametric estimation of the density $f(x)$. Another approach would be a parametric approach in which we assume that $f(x)$ belongs to some multi-parameter family of distributions (normal, gamma, ...) and then estimate the parameters.

We can get a quick look at the density function $f(x)$ by plotting a histogram of our sample of X. We normalize the histogram so that the area of a rectangle is equal to the fraction of samples that fall in that bin. So the total area in the histogram is 1. Then the histogram is an approximation to $f(x)$. The histogram is easy and great for some purposes. The obvious drawbacks: it does not give a smooth function as the estimate of the density, it depends on the bin width chosen and on where we start the first bin. The dependence on the bin width is very similar to the dependence of kernel density estimation on the bandwidth which we will look at in some detail later.

Another approach is to compute the cumulative distribution function (CDF):

$$F(t) = P(X \leq t) \tag{11.1}$$

Given a sample $X_1, X_2, \cdots, X_N$, the natural estimator for the CDF is the empirical CDF

defined by

$$\hat{F}_N(t) = \frac{1}{N}\sum_{i=1}^{N} 1_{X_i \le t} \tag{11.2}$$

Note that $F(t) = E[1_{X \le t}]$, so we can think of computing the CDF as computing the means of the one parameter family of RV's $1_{X \le t}$. The usual estimator for a mean is the sample mean. For the RV $1_{X \le t}$, the sample is $1_{X_1 \le t}, 1_{X_2 \le t}, \cdots, 1_{X_N \le t}$, so this sample mean is just $\hat{F}_N(t)$. In particular we can estimate the variance of $\hat{F}_N(t)$ and put error bars on it.

Note that for a fixed t, $1_{X \le t}$ is a Bernoulli trial. It is 1 with probability $F(t)$ and 0 with probability $1 - F(t)$. If we are doing direct Monte Carlo, then the samples are independent and so the variance of $\hat{F}(t)$ for N samples is

$$var(\hat{F}_N(t)) = \frac{\sigma^2}{N} \tag{11.3}$$

where σ^2 is the variance of $1_{X \le t}$. We can estimate this variance by the sample variance of $1_{X_1 \le t}, 1_{X_2 \le t}, \cdots, 1_{X_N \le t}$. Note that since $1_{X \le t}$ only takes on the values 0 and 1, a trivial calculation shows the variance is $F(t)[1 - F(t)]$. So we can also estimate the variance σ^2 by $\hat{F}_N(t)[1 - \hat{F}_N(t)]$. A little calculation shows this is the same as using the sample variance up to a factor of $N/(N-1)$.

Assume that $F(t)$ is continuous and strictly increasing (on the range of X.) Recall that if we let $U_i = F(X_i)$, then the U_i are i.i.d. with uniform distribution on $[0, 1]$. Let

$$\hat{G}_N(u) = \frac{1}{N}\sum_{i=1}^{N} 1_{U_i \le u} \tag{11.4}$$

This is empirical CDF for the U_i and is sometimes called the reduced empirical CDF for the X_i. Note that it does not depend on F. Note that the CDF of a uniform random variable U on $[0, 1]$ is just $G(u) = P(U \le u) = u$. The Kolmogorov-Smirnov statistic is

$$D_N = \sup_t |\hat{F}_N(t) - F(t)| = \sup_u |\hat{G}_N(u) - u| \tag{11.5}$$

We collect some facts in the following propostion

Proposition 1 *If the samples come from a direct Monte Carlo then*

1. *$N\hat{F}_N(t)$ has a binomial distribution with $p = F(t)$. The central limit theorem implies that for a fixed t, $\sqrt{N}(\hat{F}_N(t) - F(t))$ converges in distribution to a normal distribution with mean zero and variance $F(t)(1 - F(t))$.*

2. *The law of large numbers immediately implies that for each t, the random variables $\hat{F}_N(t)$ converge almost surely to $F(t)$. The Glivenko-Cantelli theorem gives a much stronger result. It says that with probability one, the convergence is uniform in t.*

3. *For $x > 0$*

$$\lim_{N\to\infty} P(\sqrt{N}D_N \leq x) = \sum_{k=-\infty}^{\infty} (-1)^k e^{-2(kx)^2} \tag{11.6}$$

If our samples come from an MCMC, then the samples are not independent. It is still true that $\hat{F}_N(t)$ is the sample mean of $1_{X\leq t}$. So we can use the techniques for error bars for MCMC (e.g., batched means) to put error bars on $\hat{F}_N(t)$.

11.2 Computing the distribution - Kernel density estimation

We follow chapter 8 of the Handbook.

Given a sample $X_1, X_2, \cdots, X_N$ we want an estimator of the density $f(x)$. The crude idea is to put mass $1/N$ at each X_i and then smear it out a little to get a smooth function. More precisely, we take a symmetric function $K(x)$ which is non-negative and has integral 1. This function is called the *kernel density*. It is helpful to think of the "spread" of this function being of order 1. We also have a parameter $h > 0$, called the *bandwidth*. The function $\frac{1}{h}K((x-c)/h)$ has integral 1, is centered at c and has width of order h. The kernel density estimator is then

$$\hat{f}(x,h) = \frac{1}{N}\sum_{i=1}^{N} \frac{1}{h} K(\frac{x - X_i}{h}) \tag{11.7}$$

We have to choose both the kernel density and the bandwidth. The choice of the kernel density is not so crucial and a natural choice for the kernel density is the standard normal density. With this choice

$$\hat{f}(x,h) = \frac{1}{N}\sum_{i=1}^{N} \phi(x, X_i, h) \tag{11.8}$$

where

$$\phi(x,\mu,h) = \frac{1}{h\sqrt{2\pi}} \exp(-\frac{(x-\mu)^2}{2h^2}) \tag{11.9}$$

The choice of the bandwidth is the crucial choice.

We need a criterion for the optimal choice of the bandwidth. A widely studied choice is the mean integrated square error (MISE):

$$MISE(h) = E\int [\hat{f}(x,h) - f(x)]^2\, dx \tag{11.10}$$

Another choice would be

$$E\int |\hat{f}(x,h) - f(x)|\, dx \tag{11.11}$$

The MISE has the advantage that we can compute things for it. A straightforward computation shows

$$MISE(h) = \int [E\hat{f}(x,h) - f(x)]^2\, dx + \int Var(\hat{f}(x,h))\, dx \tag{11.12}$$

In the first term, $E\hat{f}(x,h) - f(x)$ is the pointwise bias in the estimator. In the second term the integrand is a pointwise variance of the estimator.

We expect that the optimal h should go to zero as $N \to \infty$, and we also expect it goes to zero more slowly than $1/N$. One might expect that it goes like N^p for some p between 0 and 1, but it is not obvious what p should be. We will find an approximation to $MISE(h)$ for small h.

For the first term in (11.12) we first use the fact that $\hat{f}(x,h)$ is a sum of identically distributed terms. So

$$\begin{aligned} E[\hat{f}(x,h)] - f(x) &= E[\phi(x,X,h)] - f(x) && (11.13)\\ &= \frac{1}{h\sqrt{2\pi}}\int e^{-(x-u)^2/2h^2} f(u)\, du - f(x) && (11.14)\\ &= \frac{1}{h\sqrt{2\pi}}\int e^{-(x-u)^2/2h^2}[f(u) - f(x)]\, du && (11.15)\\ & && (11.16) \end{aligned}$$

In the integral u is close to x so we do a Taylor expansion:

$$\begin{aligned} & \frac{1}{h\sqrt{2\pi}}\int e^{-(x-u)^2/2t}[f(u) - f(x)]\, du && (11.17)\\ \approx\ & \frac{1}{h\sqrt{2\pi}}\int e^{-(x-u)^2/2h^2}[f'(x)(u-x) + \frac{1}{2}f''(x)(u-x)^2]\, du && (11.18)\\ =\ & \frac{1}{2}f''(x)h^2 && (11.19) \end{aligned}$$

Squaring this and integrating over x gives $\frac{1}{4}h^4||f''||_2^2$. where

$$||f''||_2^2 = \int (f''(x))^2\, dx \tag{11.20}$$

For the second term in (11.12) we use the fact that $\hat{f}(x,h)$ is a sum of i.i.d. terms. So

$$Var(\hat{f}(x,h)) = \frac{1}{N} Var(\phi(x,X,h)) \tag{11.21}$$

To compute $Var(\phi(x,X,h))$ we first compute the second moment:

$$\frac{1}{h^2 2\pi} \int \exp(-\frac{(x-u)^2}{h^2})\, f(u)\, du \tag{11.22}$$

We then need to integrate this over x. The result is $\frac{1}{h2\sqrt{\pi}}$. Next we compute the first moment:

$$\frac{1}{h\sqrt{2\pi}} \int \exp(-\frac{(x-u)^2}{2h^2})\, f(u)\, du \tag{11.23}$$

We must square this and then integrate the result over x:

$$\frac{1}{h^2 2\pi} \int dx \left[\int \exp(-\frac{(x-u)^2}{2h^2}) f(u) du \right]^2 \tag{11.24}$$

Do the x integral first and we see that it will give approximately $h1_{|u-v|\leq ch}$. This leads to the term being of order 1. Note that the second moment was proportional to $1/h$ which is large. So the term from the first moment squared is neglible compared to the second moment term. So the second term in (11.12) becomes $\frac{1}{h2\sqrt{\pi}}$.

Thus for small h we have

$$MISE(h) \approx \frac{1}{4} h^4 ||f''||_2^2 + \frac{1}{2Nh\sqrt{\pi}} \tag{11.25}$$

Minimizing this as a function of h we find the optimal choice of bandwidth is

$$h^* = \left(\frac{1}{2N\sqrt{\pi}||f''||_2^2} \right)^{1/5} \tag{11.26}$$

Of course the problem with the above is that we need $||f''||_2^2$ when f is precisely the function we are trying to estimate. A crude approach is the Gaussian rule of thumb. We pretend like f has a normal distribution with mean $\hat{\mu}$ and variance $\hat{\sigma}^2$. The computation of $||f''||_2^2$ for a normal density is straightforward but a bit tedious. Obviously it does not depend on the mean of the normal. We will argue that the dependence on the standard deviation σ must be of the form $c\sigma^{-5}$. Let $f_{\sigma,\mu}(x)$ be the density of the normal with mean μ and variance σ^2. Then

$$f_{\sigma,\mu}(x) = \frac{1}{\sigma} f_{1,0}(\frac{x-\mu}{\sigma}) \tag{11.27}$$

So

$$f''_{\sigma,\mu}(x) = \frac{1}{\sigma^3} f''_{1,0}(\frac{x-\mu}{\sigma}) \tag{11.28}$$

So

$$\begin{aligned} ||f''_{\mu,\sigma}||_2^2 &= \frac{1}{\sigma^6} \int [f''_{1,0}(\frac{x-\mu}{\sigma})]^2 \, dx & (11.29) \\ &= \frac{1}{\sigma^5} \int [f''_{1,0}(u)]^2 \, du & (11.30) \end{aligned}$$

With a bit of work (which we skip) one can compute the last integral. The result is that the Gaussian rule of thumb gives the following choice of bandwidth:

$$h_G = \left(\frac{4}{3N}\right)^{1/5} \hat{\sigma} \approx 1.06 \hat{\sigma} N^{-1/5} \tag{11.31}$$

In the figures we show some results of kernel density estimation. The distribution is a mixture of two normals. Each normal has variance 1. They are centered at $\pm c$ with $c = 2$ in the first four figures and $c = 10$ in the last figure. The mixture is given by taking the normal at $-c$ with probabiliity $1/3$ and the normal at $+c$ with probability $2/3$. In all the cases we used $10,000$ samples.

In each figure we estimate the variance of the distribution using the sample variance. We then use the Gaussian rule of thumb to compute the h_G. This is labelled "optimal" in the figures. We also do kernel density estimation with
$h = h_G/16, h_G/8, h_G/4, h_G/2, h_G * 2, h_G * 4, h_G * 8, h_G * 16$.

Figure 11.1 uses values of $h_G/16$ and $h_G/8$. The estimated $\hat{f}$ follows f pretty well but with significant fluctuation.

Figure 11.2 uses values of $h_G/4$ and $h_G/2$. There values do well. In fact, $h_G/2$ does better than h_G, shown in the next figure.

Figure 11.3 uses values of h_G and $2h_G$. The value $2h_G$ does rather poorly, and even the optimal value of h_G is significanty different from the true f.

Figure 11.4 uses values of $4h_G$ and $8h_G$. These values are way too large. The estimated f does not even resemble the true f.

In figure 11.5 the centers of the two normals are quite far apart, ± 10. The figure shows the kernel density estimation with the optimal choice of h from the Gaussian rule of thumb and the estimator with h equal to the optimal value divided by 8. Clearly the latter does much better. The Gaussian rule of thumb fails badly here. Note that as we move the centers of the two Gaussians apart the variance grows, so the Gaussian rule of thumb for the optimal h increases. But the optimal h should be independent of the separation of the two Gaussians.

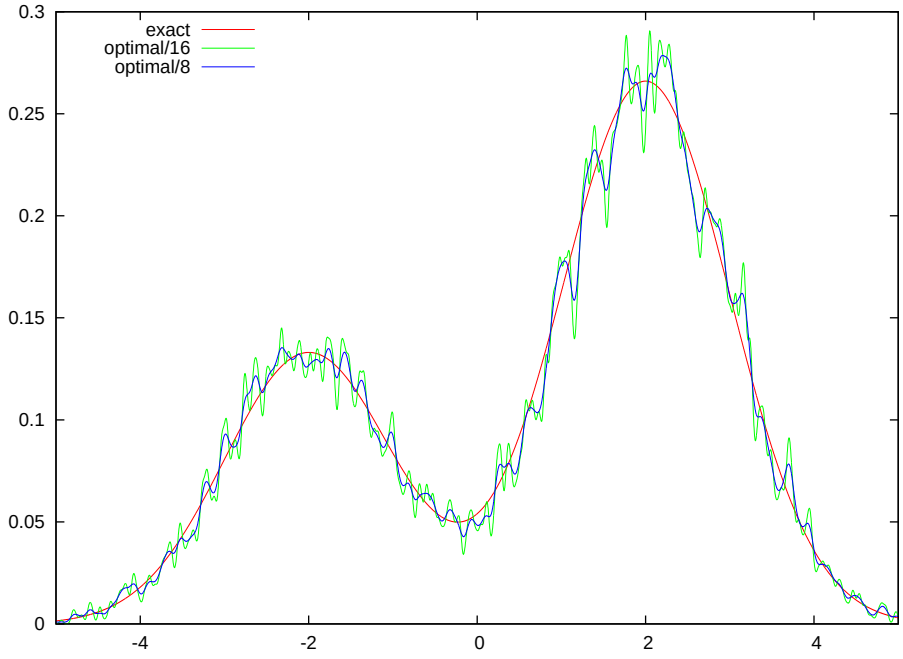

Figure 11.1:

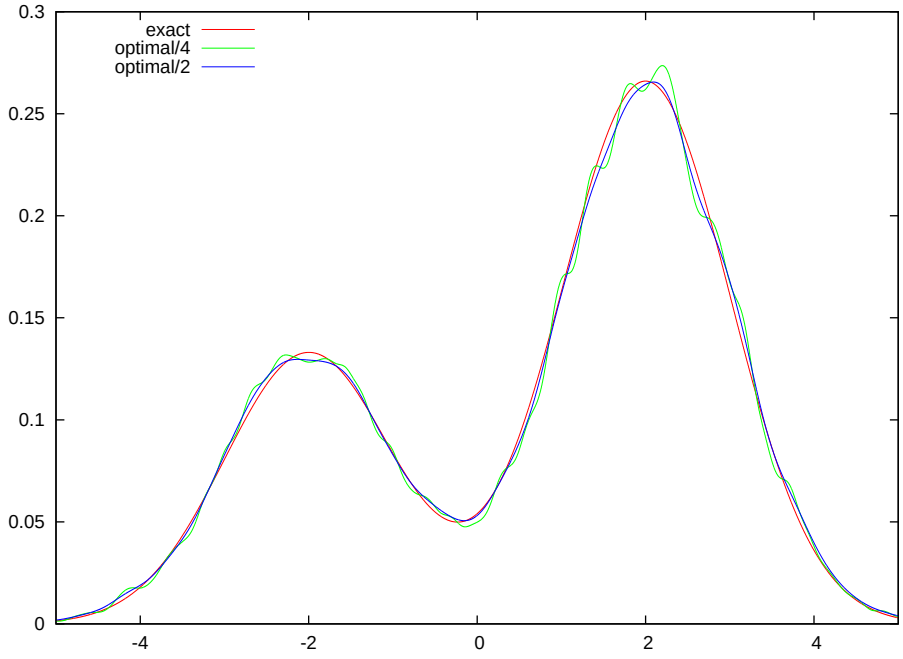

Figure 11.2:

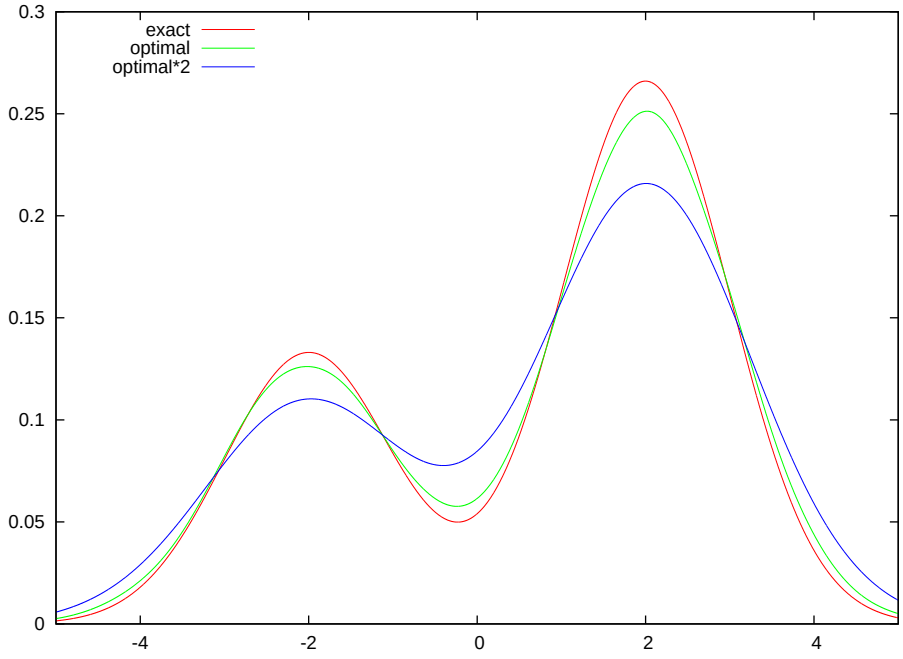

Figure 11.3:

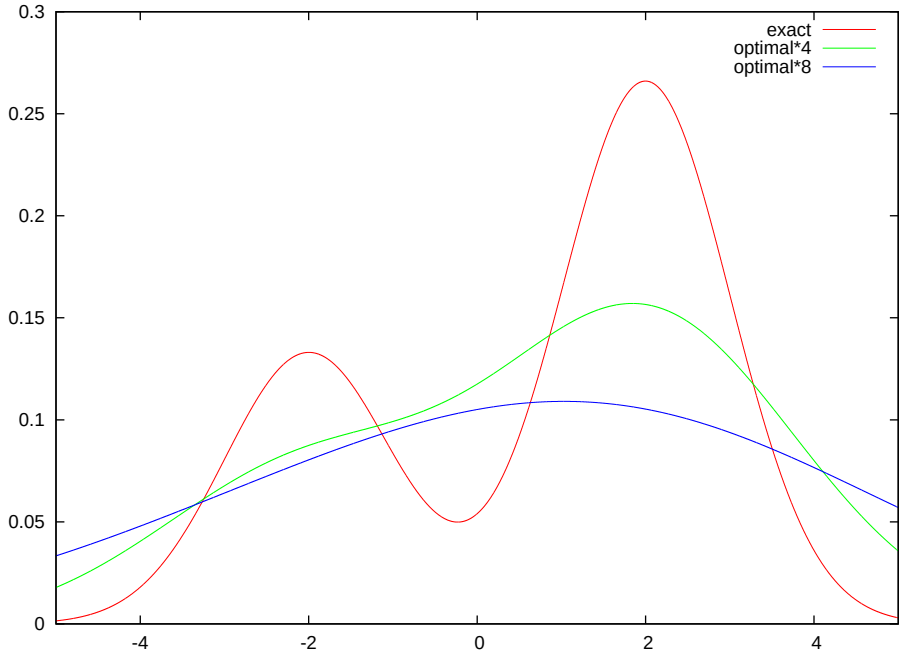

Figure 11.4:

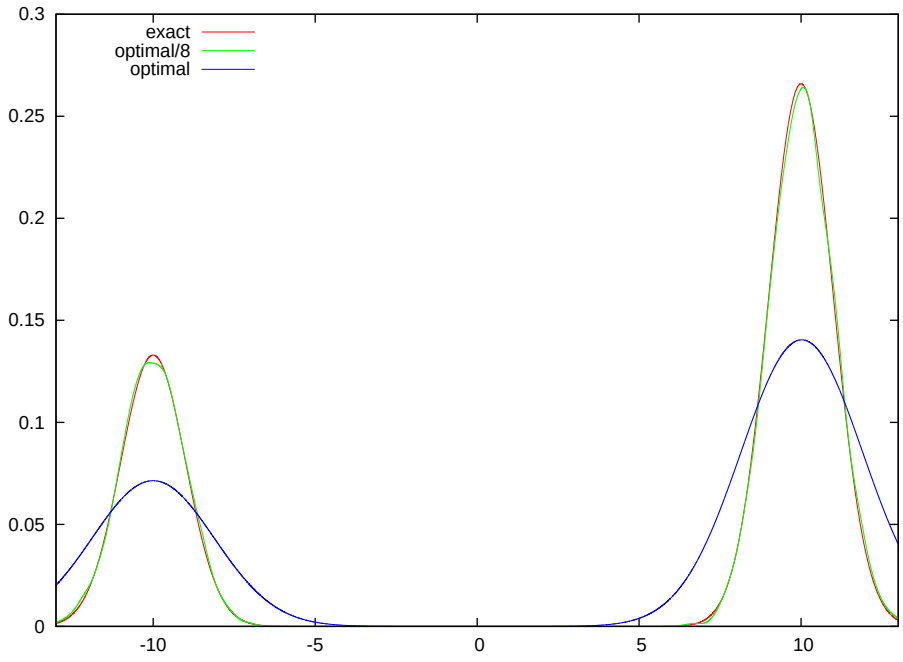

Figure 11.5:

We now consider multivariate kernel density estimation, i.e., estimating the density function for a random vector with dimension d. The obvious generalization of the estimator is

$$\hat{f}(x,h) = \frac{1}{N}\sum_{i=1}^{N} \frac{1}{h^d} K(\frac{x - X_i}{h}) \tag{11.32}$$

where $K(x)$ is now a non-negative function on R^d with integral 1.

There are some new issues in the multi-variate case. The different components of X may have vastly different scales. The kernel function needs to take this into account. This problem can occur in a more subtle way. Consider a two dimensional distribution in which the support of the density is a long narrow ellipsoid at an angle of 45 degrees with respect to the coordinates axes.

The problem of the components living on different scales can be dealt with by "pre-scaling." We rescale each component of the data so that they are on roughly the same scale. Then we do the kernel density estimation and then we rescale the resulting estimator.

The more subtle problem can be dealt with by "pre-whitening." Note that the covariance matrix of the data is far from the identity. We find a linear transformation of the data so that the covariance matrix of the new data is approximately the identity.

One possible multivariate kernel is to use a product form.

$$K(x,X,h) = \prod_{j=1}^{d} \frac{1}{h_j} k(\frac{x_j - X_j}{h_j}) \tag{11.33}$$

where $k(x)$ is a univariate kernel density. We can use a different h_j for each direction to account for different scale for the different directions. The h_j can be found as in the 1-d case.

11.3 Sequential monte carlo

We follow section 14.1 of the Handbook. For rigorous proof see the article "Particle Filters - A Theoretical Perspective" in *Sequential Monte Carlo Methods in Practice*, A. Doucet et al. (eds.).

11.3.1 Review of weighted importance sampling

We first review some things.

We want to compute $\mu = E[f(\vec{X})]$. Let $p(\vec{X})$ be the density of $\vec{X}$. We cannot sample from $p(\vec{X})$. But we can sample from $q(\vec{X})$ which is close to p in some sense. The importance sampling algorithm is as follows. Generate samples $\vec{X}_1, \cdots, \vec{X}_n$ according to the distribution $q(x)$. Then the estimator for μ is

$$\hat{\mu}_q = \frac{1}{n} \sum_{i=1}^{n} \frac{f(\vec{X}_i) p(\vec{X}_i)}{q(\vec{X}_i)} \tag{11.34}$$

We can think of this importance sampling Monte Carlo algorithm as just ordinary Monte Carlo applied to $E_q[f(\vec{X})p(\vec{X})/q(\vec{X})]$. $\hat{\mu}_q$ is an unbaised estimator of μ, i.e., $E_q\hat{\mu}_q = \mu$, and it converges to μ with probability one.

Suppose $p(x) = c_p p_0(x)$ where $p_0(x)$ is known, but c_p is unknown. And suppose we can sample from $q(x)$, but $q(x) = c_q q_0(x)$ where $q_0(x)$ is known and c_q is unknown. Then we can still do self-normalized or weighted importance sampling. The key observation is

$$\int f(x)p(x)dx = \frac{E_q[f(x)w(x)]}{E_q[w(x)]} \tag{11.35}$$

where $w(x) = p_0(x)/q_0(x)$ is a known function.

The self-normalized importance sampling algorithm is as follows. We generate samples $\vec{X}_1, \cdots, \vec{X}_n$ according to the distribution $q(x)$. Our estimator for $\mu = \int f(x)p(x)dx$ is

$$\hat{\mu}_{WI} = \frac{\sum_{i=1}^{n} f(\vec{X}_i) w(\vec{X}_i)}{\sum_{i=1}^{n} w(\vec{X}_i)} \tag{11.36}$$

where the WI subscript indicates this is the estimator coming from weighted importance sampling.

One way in which weighted importance sampling can do poorly is that the weighted are unbalanced, i.e., most of them are very small and only a few contribute to the overall weight. One measure of this is the effective sample size given by

$$\frac{(\sum_i w_i)^2}{\sum_i w_i^2} \tag{11.37}$$

where $w_i = w(\vec{X}_i)$.

11.3.2 Resampling

Before we discuss sequential monte carlo, we first consider the "resampling" that is part of the algorithm.

We want to compute $\mu = E[f(\vec{X})]$ using weighted importance sampling as above. We generate N samples $\vec{X}_1, \cdots, \vec{X}_N$ according to the density $q(x)$ and then approximate μ by

$$\hat{\mu}_{WI} = \sum_{i=1}^{N} p_i f(\vec{X}_i) \tag{11.38}$$

where p_i are the normalized importance sampling weights:

$$p_i = \frac{w_i}{\sum_{j=1}^{n} w_j} \tag{11.39}$$

Note that we can think of this as the integral of f with respect to the measure

$$\sum_{i=1}^{N} p_i \delta_{X_i}(x) \tag{11.40}$$

Resampling means that we replace this measure by

$$\frac{1}{N} \sum_{i=1}^{N} N_i \delta_{X_i}(x) \tag{11.41}$$

where the N_i are non-negative integers whose sum is N. So the new estimator for μ is

$$\hat{\mu}_R = \frac{1}{N} \sum_{i=1}^{N} N_i f(X_i) \tag{11.42}$$

There are different methods for choosing the N_i.

The simplest method is to draw N independent samples from the discrete density (11.40). This means the joint distribution of $N_1, N_2, \cdots, N_N$ is a multinomial distribution with N trials and probabilities $p_1, p_2, \cdots, p_N$. Note that many of the N_i will be zero. We would like to know that $\hat{\mu}_R$ converges to μ as $N \to \infty$. This is subtle. Note that the N_i are not independent.

We first show it is an unbiased estimator.

$$E[\hat{\mu}_R] = \frac{1}{N} \sum_{i=1}^{N} E[N_i f(X_i)] \tag{11.43}$$

We compute using the tower property:

$$\begin{aligned} E[N_i f(X_i)] &= E[E[N_i f(X_i) | X_1, \cdots, X_N]] & (11.44) \\ &= E[f(X_i) E[N_i | X_1, \cdots, X_N]] = E[f(X_i) N p_i] & (11.45) \end{aligned}$$

So

$$E[\hat{\mu}_R] = \sum_{i=1}^{N} E[f(X_i) p_i] = \mu \tag{11.46}$$

Next we show that for bounded functions f, the estimator converges to μ in the L^2 sense (and hence in probability). It converges a.s., but we do not prove this. See the article by Cristan. We already know that the estimator from weighted importance sampling converges to μ with probability one, i.e., $\hat{\mu}_{WI} \to \mu$ with probability one. Since f is bounded the bounded convergence theorem implies we also have convergence in L^2. So it suffices to show $\hat{\mu}_R - \hat{\mu}_{WI}$ converges to 0 in L^2, i.e., we need to show $E[(\hat{\mu}_R - \hat{\mu}_{WI})^2] \to 0$. Note that

$$\hat{\mu}_R - \hat{\mu}_{WI} = \frac{1}{N} \sum_{i=1}^{N} f(X_i)(N_i - p_i N) \tag{11.47}$$

We use conditioning again:

$$E[(\hat{\mu}_R - \hat{\mu}_{WI})^2] = E[E[(\hat{\mu}_R - \hat{\mu}_{WI})^2 | X_1, \cdots, X_N]] \tag{11.48}$$

We have

$$E[(\hat{\mu}_R - \hat{\mu}_{WI})^2 | X_1, \cdots, X_N]] \tag{11.49}$$

$$= \frac{1}{N^2} \sum_{i,j=1}^{N} f(X_i) f(X_j) E[(N_i - p_i N)(N_j - p_N) | X_1, \cdots, X_N] \tag{11.50}$$

Conditioned on $X_1, X_2, \cdots, X_N$, the p_i are constant and the N_i follow a multinomial distribution. So we can compute

$$E[(N_i - p_i N)(N_j - p_N) | X_1, \cdots, X_N] = -N p_i p_j \tag{11.51}$$

So

$$|E[(\hat{\mu}_R - \hat{\mu}_{WI})^2 | X_1, \cdots, X_N]]| \leq \frac{1}{N} \sum_{i,j=1}^{N} |f(X_i) f(X_j)| p_i p_j \leq \frac{||f||_\infty^2}{N} \tag{11.52}$$

Thus

$$E[(\hat{\mu}_R - \hat{\mu}_{WI})^2] = E[E[(\hat{\mu}_R - \hat{\mu}_{WI})^2 | X_1, \cdots, X_N]] \tag{11.53}$$

$$\leq E[|E[(\hat{\mu}_R - \hat{\mu}_{WI})^2 | X_1, \cdots, X_N]|] \leq \frac{||f||_\infty^2}{N} \to 0 \tag{11.54}$$

There are other ways to define the N_i. The definition above is not ideal because it introduces a fair amount of variance into the problem. Suppose we have a relatively small set of indices for which $p_i N$ is relatively large and the rest of the $p_i N$ are close to zero. The above procedure will replace the one copy for index i with N_i copies where the mean of N_i is $p_i N$ but the standard deviation is of order $\sqrt{N_i}$. It might be better to take the number of copies to be $p_i N$ rounded to the nearest integer. The following algorithm does something in this spirit.

Stratified resampling Let n_i be the largest integer less than or equal to $p_i N$. Note that the sum of the n_i cannot exceed N. Create c_i copies of X_i. We are still short $N_r = N - \sum_i n_i$ samples. Draw a sample of size N_r from $\{1, 2, \cdots, N\}$ uniformly and without replacement. Add these X_i to the previous ones. Finally, if it makes you feel better you can do a random permutation of our sample of size N. (What's the point?)

11.3.3 sequential MC

Now suppose that instead of a random vector we have a stochastic process $X_1, X_2, X_3, \cdots$. We will let X stand for $X_1, X_2, X_3, \cdots$. We want to estimate the mean of a function of the process $\mu = f(X)$. It doesn't make sense to try to give a probability density for the full infinite process. Instead we specify it through conditional densities:
$p_1(x_1), p_2(x_2|x_1), p_3(x_3|x_1, x_2), \cdots, p_n(x_n|x_1, x_2, \cdots, x_{n-1}), \cdots$. Note that it is immediate from the definition of conditional density that

$$\begin{aligned} p(x_1, x_2, \cdots, x_n) &= p_n(x_n|x_1, x_2, \cdots, x_{n-1}) p_{n-1}(x_{n-1}|x_1, x_2, \cdots, x_{n-2}) && (11.55) \\ &\cdots \quad p_3(x_3|x_1, x_2) p_2(x_2|x_1) p_1(x_1) && (11.56) \end{aligned}$$

We specify the proposal density in the same way:

$$\begin{aligned} q(x_1, x_2, \cdots, x_n) &= q_n(x_n|x_1, x_2, \cdots, x_{n-1}) q_{n-1}(x_{n-1}|x_1, x_2, \cdots, x_{n-2}) && (11.57) \\ &\cdots \quad q_3(x_3|x_1, x_2) q_2(x_2|x_1) q_1(x_1) && (11.58) \end{aligned}$$

So the likehood function is

$$w(x) = \prod_{n \geq 1} \frac{p_n(x_n|x_1, x_2, \cdots, x_{n-1})}{q_n(x_n|x_1, x_2, \cdots, x_{n-1})} \tag{11.59}$$

An infinite product raises convergence questions. But in applications f typically either depends on a fixed, finite number of the X_i or f depends on a finite but random number of the X_i. So suppose that f only depends on $X_1, \cdots, X_M$ where M may be random. To be more precise we assume that there is a random variable M taking values in the non-negative integers such that if we are given that $M = m$, then $f(X_1, X_2, \cdots)$ only depends on $X_1, \cdots, X_m$. So we can write

$$f(X_1, X_2, \cdots) = \sum_{m=1}^{\infty} 1_{M=m} \, f_m(X_1, \cdots, X_m) \tag{11.60}$$

We also assume that M is a stopping time. This means that the event $M = m$ only depends on $X_1, \cdots, X_m$. Now we define

$$w(x) = \sum_{m=1}^{\infty} 1_{M=m}(x_1, \cdots, x_m) \prod_{n=1}^{m} \frac{p_n(x_n|x_1, x_2, \cdots, x_{n-1})}{q_n(x_n|x_1, x_2, \cdots, x_{n-1})} \tag{11.61}$$

MORE Explain the potential problem that the weights can degenerate to the point that most are zero.

The final step is to modify the above by resampling at each time step. So the sequential Monte Carlo algorithm is as follows. Throughout N will be the number of samples. They are often called "particles." (There are variants where the number of particles changes with time, but we only consider an algorithm where the number stays constant.)

1. Initialize. Given N iid samples $X_1^1, \cdot, X_1^N$ from $q_1()$. The subscript 1 means $t = 1$.

2. Importance sampling. Given $X_{1:t-1}^1, \cdots, X_{1:t-1}^N$, generate (independently) Y_t^j from $q_t(\cdot|X_{1:t-1}^j)$. The Y_t^j are conditionally independent given $X_{1:t-1}^1, \cdots, X_{1:t-1}^N$, but not independent. Compute the weights

$$w_{t,j} = \frac{p_t(Y_t^j|X_{1:t-1}^j)}{q_t(Y_t^j|X_{1:t-1}^j)} \tag{11.62}$$

and then let $p_{t,j}$ be the normalized weights:

$$p_{t,j} = \frac{w_{t,j}}{\sum_{i=1}^N w_{t,j}} \tag{11.63}$$

3. Resample. Let $Z_{1:t}^j = (X_{1:t-1}^j, Y_t^j)$. Generate $X_{1:t}^j$ by independently sampling (N times) from the discrete distribution which has the values $Z_{1:t}^j$ with probability $p_{t,j}$. So we are drawing N independent samples from the mixture

$$\sum_{j=1}^N p_{t,j}\delta_{Z_{1:t}^j} \tag{11.64}$$

We do steps 2 and 3 for $t = 2, \cdots, T$ where T could be a random stopping time.

Example - random walk: We look at a one-dimensional random walk which only takes steps of ± 1. The probability of going right is p. For a proposal distribution we use a symmetric random walk which goes right or left with probability 1/2. We run the walk for 100 time steps. The random variable we study is the position of the walk at the end of the 100 steps. Note that we know μ exactly. It is $100(2p-1)$. We do two simulations, each with $2,000$ samples. One simulation is weighted importance sampling. The other is sequential MC using the multinomial resampling. Figure 11.6 shows the two estimators $\hat{\mu}_{WI}$ and $\hat{\mu}_R$ as a function of p, along with the exact result.

To see if the breakdown for the weighted importance sampling simulation somes from the weights becomes very unbalanced we plot the effective sample size as a function of time for

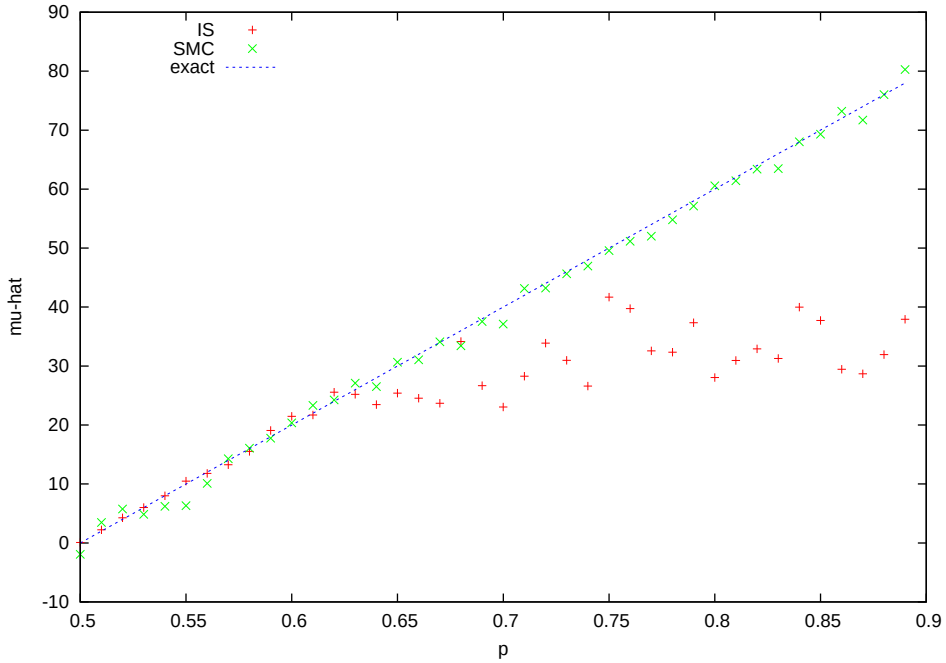

Figure 11.6:

several values of p in figure 11.7. The figure shows that for all values of p the effective sample size usually decreases with time. The rate of decrease gets larger as p moves away from 1/2.

In figure 11.8 we plot the effective sample size at time 100 as a function of p. Note that the value of p where the effective sample size becomes small corresponds with the value of p in figure 11.6 where the estimator $\hat{\mu}_R$ deviates significatly from μ.

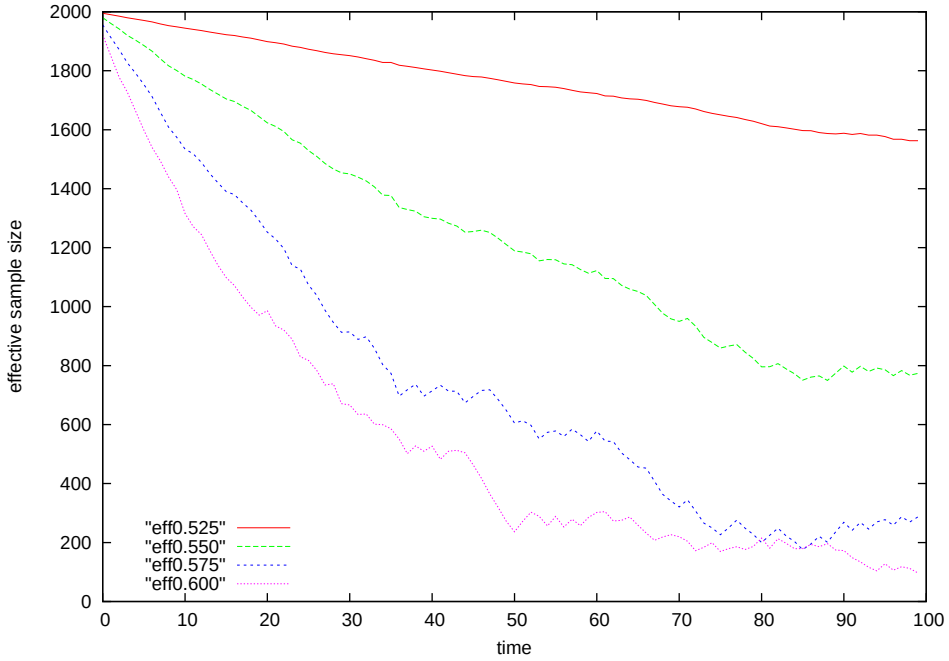

Figure 11.7:

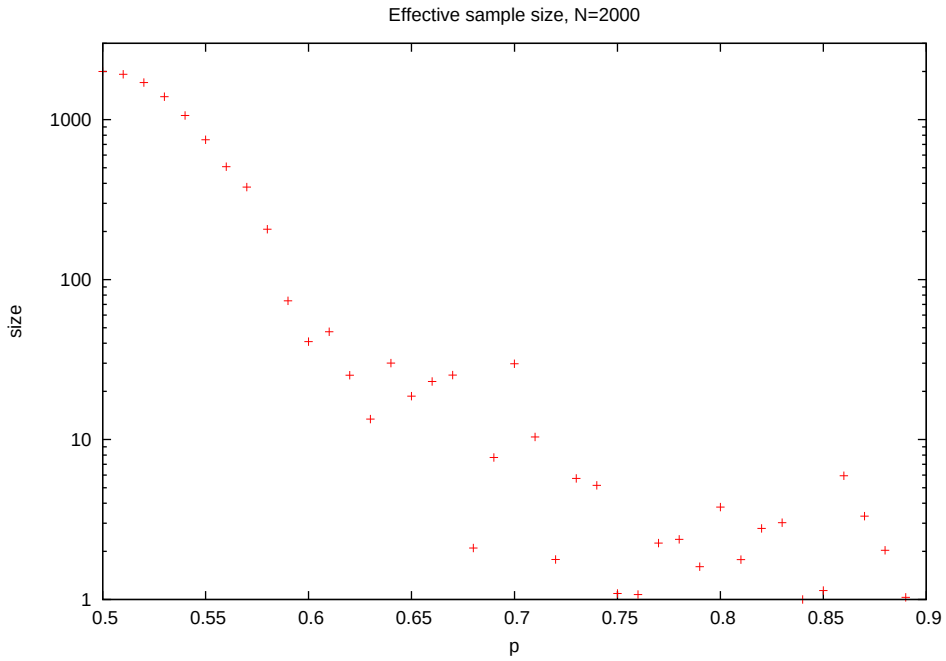

Figure 11.8: