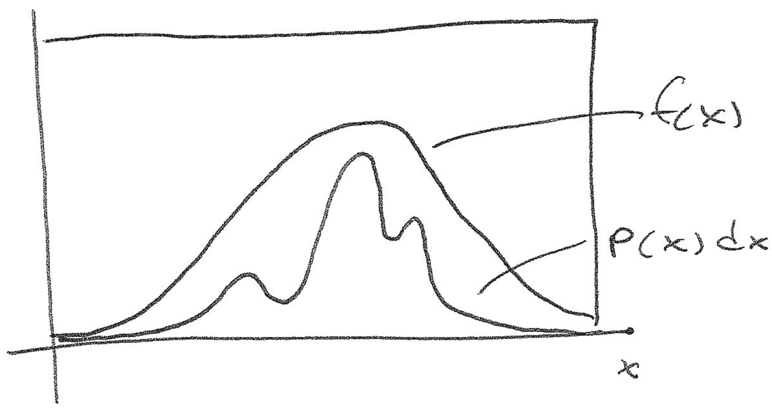
Rejection Method

Consider sampling from a complex $p(x)\,dx$



Imagine sampling from within the rectangular area uniformly ... EASY!

If the sample falls within the area under $p(x)$ then keep it, else try again.

Since the area under the density curve is proportional to the probability, the samples will be dist. according to $p(x)$.

This method suffers from a lot of rejects.
WASTEFUL COMPUTATIONS!

Instead choose a COMPARISON FUNCTION $f(x)$
st. $f(x) > p(x)\,dx$

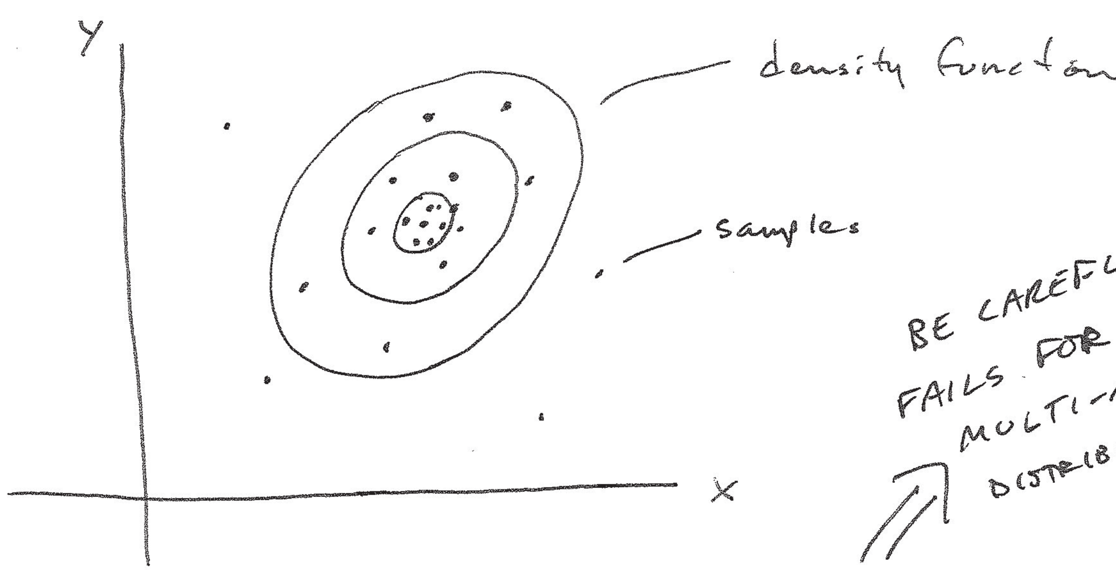
To sample from $f(x)$ uniformly, choose a uniform sample from the range x and choose a uniform sample between $[0, f(x)]$ in y. You could also choose y to be $[0,1]$ and compare to the ratio $p(x)/f(x)$

Sampling

One way to explore a probability density function is by drawing samples.

In a hypothesis space, this is equivalent to selecting a set of points in the space according to their probabilities. In this sense, the set of samples is considered to be representative of the density function.

The fact that these samples are representative is reflected in the fact that they can be used to obtain the mean, variance and higher order moments of the density function numerically



BE CAREFUL
FAILS FOR MULTI-MODAL DISTRIBUTIONS!

Mean estimate of x is the mean of the x-values
Mean estimate of y is the mean of the y-values of the samples
Variances of the estimates are the variances of the samples etc.

Metropolis - Hastings MCMC

Nicholas Metropolis 1953

W. Keith Hastings 1970

Metropolis - Hastings algorithm is a rejection sampling method.

It is often used in conjunction with a Markov chain Monte Carlo algorithm to generate a list of samples.

Start with a sample x, compute $p(x)$.

Now propose a new sample y, compute $p(y)$.

If $\frac{P(y)}{P(x)} > 1$ (or $P(y) > P(x)$)

$x_{new} = y$;

Else

$$x_{new} = \begin{cases} y & \text{with prob } \frac{P(y)}{P(x)} \\ x & \text{with prob } 1 - P(y)/P(x) \end{cases}$$

END

Metropolis - Hastings cont.

A more compact algorithm

```
p = min([1, P(y)/P(x)])

r = rand();
If r < p
    x_new = y;
else
    x_new = x;
end
```

How do we choose y?

we take steps in the space away from x.

If we go too far, we risk falling into low prob areas.

If we stay too close, we are not exploring!

Choose step size by monitoring acceptance rate.

Must wait for chain to come to equilibrium with the probability density...

BURN IN   How to know????

Useful Acceptance Rate monitoring

from Larry Bretthorst

```
if acceptanceRatio < 0.1                     steps are way too big
    dx = dx/10 ;
else
    if acceptanceRatio < 0.34
        if dx > 0.1 * sigma                  steps > 0.1σ still too large
            dx = dx/2
        end
    else
        if acceptanceRatio > 0.67            few steps wasted
            if dx > abs(avg)                 we are in burn in... let it go
                % do nothing
            else
                dx = dx * 1.1                crank it up a bit
            end
        end
    end
end
```

Note that this requires monitoring:

acceptance Ratio
sigma of samples
avg of samples

considerable programming overhead

FOR EACH PARAMETER