STA 601 - Lab 4

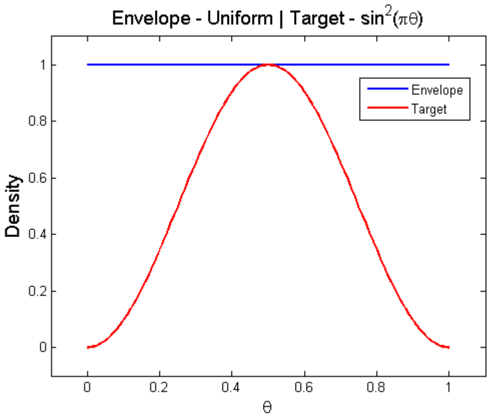
Kedar Prabhudesai

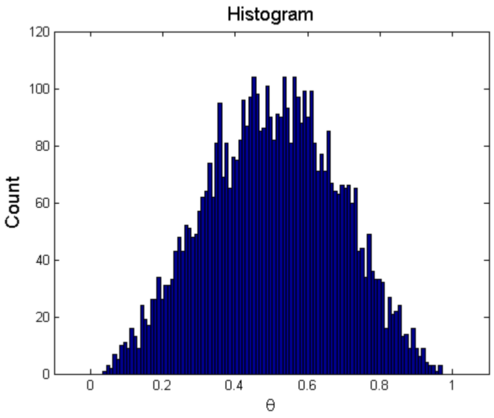
September 27, 2013

1. I implemented the Rejection Sampling algorithm in MATLAB (code attached below). However, I used a slightly different approach. Here are the steps:

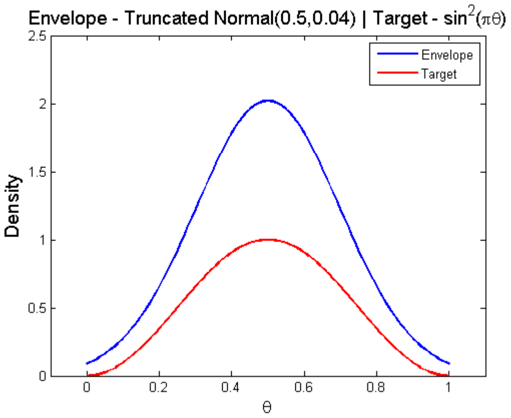
   - Draw a Sample X from the Envelope Distribution
   - Now, draw a sample $Y \sim U[0, f(X)]$, where $f(.)$ is the pdf of Envelope Density.
   - Check if $Y \leq g(X)$, where $g(.)$ is the target density from which we desire to sample.
   - If the above condition is satisfied, accept X, otherwise reject X.

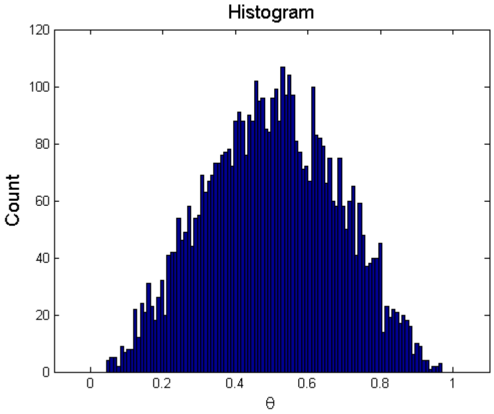
   Here are results using a $Uniform[0, 1]$ envelope density. Histogram of 10000 samples.





2. The envelope density should be greater than or equal to the target density over the support. I chose Normal density, $\mathcal{N}(\mu, \sigma^2)$, $\mu = 0.5$; $\sigma^2 = 0.04$. Then I truncated the density between $(0, 1)$. I get the following results:





3. Efficiency:
In the case of Uniform density, we end up rejecting a lot of samples because they have equal probability of being drawn in the $[0, 1]$ interval. Hence an efficient envelope density would be one that conforms to general shape of the target density. In this case, we do not end up rejecting a lot of samples drawn from $\mathcal{N}(0.5, 0.04)$, because samples in $[0, 1]$ have more or less similar probabilities of being drawn. Hence it is more efficient than Uniform.

Appendix:

```
%% STA 601: Lab 4
% Author: Kedar S Prabhudesai
% Created on: 09/25/2013

close all;
clear all;

% Support
theta = 0:0.01:1;
% Target Function
tarFn = (sin(pi*theta)).^2;
% Envelope Distribution
% envDist = makedist('Uniform','lower',0,'upper',1);
envDist = makedist('Normal','mu',0.5,'sigma',0.2);
envDist = envDist.truncate(0,1);
% Total Number of Samples
nSamples = 10000;
% Make sure that the envelope distribution is greater than or equal to the
% target function
figure;
plot(theta,envDist.pdf(theta),'b','LineWidth',2);hold on;
plot(theta,(sin(pi*theta)).^2,'r','LineWidth',2);hold off;
% title('Envelope − Uniform | Target − sin^2(\pi\theta)','FontSize',14);
title('Envelope − Truncated Normal(0.5,0.04) | Target − sin^2(\pi\theta)','FontSize',14);
legend('Envelope','Target');
xlabel('\theta','FontSize',14);ylabel('Density','FontSize',14);
xlim([−0.1 1.1]);%ylim([−0.1 1.1]);

% Samples from Envelope Distribution
envSamples = envDist.random(1,nSamples);
% Get Uniform samples to evaluate condition
tarSamples = envDist.pdf(envSamples).*rand(1,nSamples);

% Evaluate Condition whether to accept or reject the samples from the
% envelope
condAccept = (tarSamples ≤ ((sin(pi*envSamples)).^2));
% Keep only Accepted Points
result = envSamples(condAccept);

% Make Histogram
figure;
hist(result,100);
title('Histogram','FontSize',14);
xlabel('\theta','FontSize',14);ylabel('Count','FontSize',14);
xlim([−0.1 1.1]);
```