STAT 406 Lab 4, 10/06/2015

1. Generate Random Variable

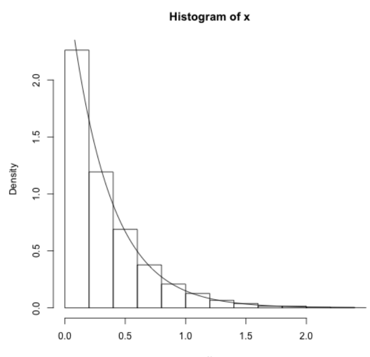
We learned from lecture that random variables from a particular distribution can be gen- erated from uniform random variables by inverting their cumulative distribution function (cdf). That is, if you create samples $U_1, \cdots, U_n$ from $Uniform(0,1)$ distribution and compute $F^{-1}(U_1), \cdots, F^{-1}(U_n)$, then what you get is samples from a distribution with cdf F . Using this property, we'll generate exponential random variables and gamma random variables.

(a) The Exponential(λ) distribution has cdf:

$$F(x) = 1 - e^{-\lambda x}, \lambda > 0$$

Using $runif$ function, generate 100 samples from $Exponential(3)$ distribution using the inversion method. Graph the density histogram for the sample with the true density superimposed for comparison.

```
n = 100
lambda = 3
x = -(1/lambda)*log(runif(n))
hist(x, prob = TRUE)
y = seq(0,10,length = 1000)
lines(y,dexp(y,3))
```



(b) $Gamma(k, \theta)$, $k > 0$, $\theta > 0$ distribution has density:

$$f(x) = \frac{\theta^k}{(k-1)!} x^{k-1} e^{-\theta x}$$

where $k\ and\ \theta$ are shape and rate parameters. What are $k\ and\ \theta$ for $Exponential(\lambda)$ distribution in terms of Gamma distribution?

Gamma distribution also has the property that if $X \sim Gamma(a, \theta)$ and $Y \sim Gamma(b, \theta)$, and X and Y are independent, then

$$X + Y \sim Gamma(a + b, \theta).$$

Using this property, create 10 samples from Gamma(5, 3) distribution.

```
n = 10
k = 5
lambda = 3
x = matrix(-(1/lambda)*log(runif(n*k)), ncol=k)
g = apply(x, 1, sum)
```

2. Rejection Sampling - (I)

Recall: three steps for generating a random variable $X \sim f$ via the Rejection Sampling:

- Generate $Y \sim g$.
- Generate $U \sim \mathsf{Uniform}(0,1)$.
- Check if $U \le \frac{f(Y)}{Mg(Y)}$. If true, set $X = Y$ (accept). Otherwise, return to step 1.

Several practical points noteworthy about choosing a good g distribution:

(1) Easy to simulate random variables from g, e.g., uniform, exponential, ect.;

(2) g should better have the same support $\mathcal{X} = \{x : g(x) > 0\}$ as f has;

(3) choose a g that give you as small M value as possible for an efficient algorithm(do not reject many Y): according to the rule

$$M \ge \frac{f(x)}{g(x)}, \text{for all } x \text{ in the support for } f \text{ and } g.$$

Revisit Lecture Example: Generating $Beta(\alpha, \beta)$ random variables
The Beta distribution has density

$$f(x) = \frac{1}{B(\alpha,\beta)} x^{\alpha-1}(1-x)^{\beta-1}, x \in (0,1);\; f(x) = 0, otherwise$$

For parameters $\alpha > 0, \beta > 0$ and $B(\alpha,\beta) = \int_0^1 x^{\alpha-1}(1-x)^{\beta-1}dx$ is called the normalizing constant of the density.

We consider the case with $\alpha \ge 1, \beta \ge 1$ and use rejection sampling to generate random variable from $Beta(\alpha, \beta)$ distribution.

From (1) and (2) above, we choose g as the uniform distribution with denstiy:

$$g(x) = 1, x \in (0,1)$$

From (3) above, notice that

$$\frac{f(x)}{g(x)} = \frac{1}{B(\alpha,\beta)} x^{\alpha-1}(1-x)^{\beta-1} \le \frac{1}{B(\alpha,\beta)} = M$$

.

So step 3 of the algorithm becomes Check if

$$U \le \frac{B(\alpha,\beta) f(Y)}{g(Y)} = Y^{\alpha-1}(1-Y)^{\beta-1}$$

So we have the following rejection algorithm for this case:

Algorithm (Rejection Sampling I) # loop the following 3 steps:

1. Generate $Y \sim \mathcal{U}(0,1)$ and $U \sim \mathcal{U}(0,1)$.
2. If $U \le Y^{\alpha-1}(1-Y)^{\beta-1}$, Stop the loop and return Y.
3. Else, reject Y and go back to Step 1.

Implement

```
RejBeta1 = function(n, alpha, beta){
Vy = numeric(n)
Vcpt = integer(n)
j=1;cpt=0;
while(j <=n ){
u = runif(1); y = runif(1); cpt =cpt + 1
if(u <= y^(alpha-1)*(1-y)^(beta - 1)){
Vy[j] = y; Vcpt[j] = cpt
j=j+1; cpt=0
}
}
return(list(Vy,Vcpt))
}

betarnd = RejBeta1(10, 2, 2)
```

3. More of LLN - Empirical Distribution Functions

Write a function which calculates the empirical distribution function of $X_1, \cdots, X_n \overset{\text{iid}}{\sim} N(0,1)$. The empirical distribution function is defined as

$$\Phi_m(x) = \frac{1}{n}\sum_{i=1}^{n} I(X_i \leq x)$$

where $I(X \leq x)$ is an indicator function, taking value 1 if $X \leq x$ and 0 otherwise. Plot this function for $x \in [-4, 4]$. Let $m = 5, 50, 500$ respectively and plot everything on the same figure, with the true Normal cdf superimposed for comparison.

```
Phi.m = function(m,xseq){
# xseq can be either a scalar or a vector
X = rnorm(m)
Phi.m.x = sapply(xseq,function(x){mean(X<x)})
return(Phi.m.x)
}

m = c(5,50,500)
xseq = seq(-4,4,by=0.01)
plot(0,xlim=c(-4,4),ylim=c(0,1),xlab="x",ylab="Phi.m(x)",main="Empirical CDF", type='n')
for (i in 1:length(m)){
par(new=T)
plot(xseq,Phi.n(m[i],xseq),col=i,axes=F,type="l",xlim=c(-4,4),ylim=c(0,1),xlab="",ylab="")
}
lines(xseq,pnorm(xseq,mean=0,sd=1),col="blue",lty=3,lwd=2)
legend("topleft",legend=paste(m,'points'),col=1:length(m),lty=1)
```