Gamma Random Variable simulation

```
setwd(getwd())
library(MASS)
data(quine)

z = sum(quine$Days)
(MLE = z/(219+z))
```

```
## [1] 0.916476
```

```
(Postest = (z+0.5)/(z+220))
```

```
## [1] 0.9163172
```

```
# Based off a recent mathematical paper that I've forgotten the name of.
# I did not come up with this on my own - just implemented the pseudocode.
# Can probably find if required.
gamma.sim <- function() {

a <- 1.2;
d <- a - (1/3);
c <- 1/(sqrt(9*d))

h <- function(x) d*((1+c*x)^3)
g <- function(x) d*log((1+c*x)^3) - d*((1+c*x)^3) + d
h_star <- function(x) exp(-(x^2)/2)
f_star <- function(x) exp(g(x))


# if Y < f_star(X)/h_star(X) it means we have successfully sampled an X from the
# h(x)^a-1 * e^-h(x) * h'(x) distribution
# we want Y = h(X)/3 however, so return that.

while (TRUE) {
  X <- rnorm(1)
  Y <- runif(1)

  if ((X > (-1/c)) && (h_star(X)*Y) < f_star(X)) {
    return (h(X)/3)
  }
}
}

set.seed(1999)
n <- 1000
p <- rep(0, n)

for (i in 1:n) {
   p[i] <- gamma.sim()
}
```

```r
plot(qgamma(1:1000/1001, 1.2, 3), sort(p))
abline(0, 1, col="red")
```