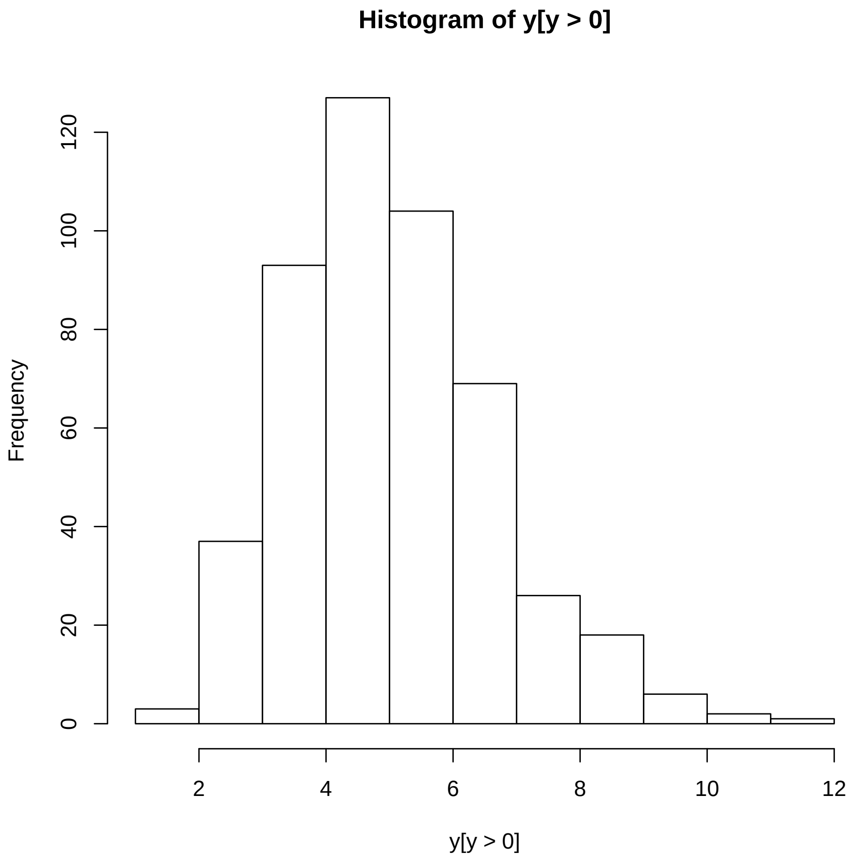
```
theta<-0.27      #probability of rain on a given day
alpha=10         #shape parameter for gamma
beta=2           #scale parameter for gamma
years<-5
N<-365*years     #sample size

rained<-rbinom(N,1,theta)        #generate binary rain/no rain vector
rainfall<-rgamma(N,alpha,beta)   #generate rainfall quantity vector

y<-rained*rainfall               #observed values are product of these

y[1:100]                         #print the first few y values

##   [1] 0.000000 3.564932 0.000000 0.000000 0.000000 5.749588 0.000000
##   [8] 0.000000 0.000000 5.958916 0.000000 0.000000 9.042803 0.000000
##  [15] 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
##  [22] 0.000000 0.000000 0.000000 3.469166 0.000000 5.825326 5.660475
##  [29] 0.000000 5.667014 0.000000 0.000000 0.000000 0.000000 0.000000
##  [36] 0.000000 0.000000 0.000000 3.204174 0.000000 0.000000 0.000000
##  [43] 0.000000 3.696990 5.035389 0.000000 0.000000 4.373850 0.000000
##  [50] 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 3.403126
##  [57] 0.000000 4.933507 0.000000 0.000000 5.963766 0.000000 0.000000
##  [64] 5.150737 0.000000 0.000000 0.000000 6.915682 7.748555 0.000000
##  [71] 4.359444 0.000000 0.000000 0.000000 6.681937 0.000000 0.000000
##  [78] 0.000000 2.355473 0.000000 0.000000 2.843608 0.000000 5.841786
##  [85] 0.000000 0.000000 0.000000 5.022743 0.000000 0.000000 0.000000
##  [92] 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
##  [99] 2.086538 0.000000

hist(y[y>0])                     #histogram of measurable rainfall amounts
```



Call STAN

```
library(rstan)

## Loading required package:  ggplot2
## rstan (Version 2.9.0-3, packaged:  2016-02-11 15:54:41 UTC, GitRev:
05c3d0058b6a)
## For execution on a local, multicore CPU with excess RAM we recommend
calling
## rstan_options(auto_write = TRUE)
## options(mc.cores = parallel::detectCores())

rstan_options(auto_write = TRUE)
options(mc.cores = parallel::detectCores())

stanfit<-stan("gamma_test.stan") #call STAN
print(stanfit)                   #print a summary of the results
```

```
## Inference for Stan model: gamma_test.
## 4 chains, each with iter=2000; warmup=1000; thin=1;
## post-warmup draws per chain=1000, total post-warmup draws=4000.
##
##           mean se_mean   sd     2.5%      25%      50%      75%    97.5%
## theta     0.27    0.00 0.01     0.25     0.26     0.27     0.27     0.29
## alpha     9.73    0.02 0.62     8.52     9.32     9.73    10.15    10.99
## beta      1.93    0.00 0.13     1.69     1.85     1.93     2.01     2.19
## lp__  -1963.53    0.04 1.29 -1966.92 -1964.13 -1963.21 -1962.58 -1962.08
##       n_eff Rhat
## theta  1408 1.00
## alpha   769 1.00
## beta    754 1.01
## lp__    965 1.01
##
## Samples were drawn using NUTS(diag_e) at Sat May 21 05:27:35 2016.
## For each parameter, n_eff is a crude measure of effective sample size,
## and Rhat is the potential scale reduction factor on split chains (at
## convergence, Rhat=1).
```

Launch shinystan

```
library(shinystan)              #downloaded from cran

## Loading required package:  shiny
##
## This is shinystan version 2.1.0

launch_shinystan(stanfit)

##
## Loading...
## Note:  for large models ShinyStan may take a few moments to launch.
##
## Listening on http://127.0.0.1:4692
```