FIT5197_ass3_wk10

Chuangfu Xie, 27771539

28/05/2018

As we have:

$$p(X = 0) = 0.5$$
$$p(X = 1) = 0.3$$
$$p(X = 0) = 0.2$$

We can define a function as the inverse transform sampler:

```
inv_sampler <- function(size, probs){
  U <- runif(size)
  sample<-rep(0,size)
  case1 <- which(U <= probs[1]) # x <= 0.5
  case2 <- which(U > probs[1] & U <= sum(probs[1:2])) # 0.5 < x <= 0.8
  case3 <- which(U > sum(probs[1:2])) # x > 0.8
  # reassign value
  sample[case1] <- 0
  sample[case2] <- 1
  sample[case3] <- 2
  return(sample)
}
```

```
props <-function(input){
  if (input==0){
    return(0.5)
  } else if (input==1){
    return(0.3)
  } else if (input==2){
    return(0.2)
  }
}
```

```
cnt=0
size = 1000
probs = c(0.5, 0.3, 0.2)
s = inv_sampler(size, probs)
C = props(0)/dnorm(0)
for (each in s){
  accept <- props(each)/(C*dnorm(each))
  u <- runif(1)
  if (accept > u){
    cnt = cnt + 1
  }
}
cat('Reject proportion is: ', 100*(size - cnt)/size,'%')
```

```
## Reject proportion is:  0.2 %
```