Exercise D: Rejection / Acceptance Sampling

Benedikt Ehinger

16. Dezember 2016

Videos J: Inverse Cumulative Function Integration & Accept-Reject, MCMC
Recommended Reading Material:

- `https://en.wikipedia.org/wiki/Rejection_sampling#Description`

0.1 Exercise 0

Use the inverse Cumulative Distribution Function (CDF) method to generate random numbers from a exponential with rate 1(use *runif* in R or *rand* in matlab). As seen in the lecture, the inverse function of the exponential is $-log(U)$. Plot the inverse function. Plot a histogram of your random numbers drawn from an exponential with rate 1.

1 Accept-Reject Method

1.1 Exercise 1

We want to sample from a standard normal function, but as it happens we forgot how to calculate the inverse cumulative of it!

Luckily we can use the accept-reject method instead. In order for it to work we need to have access to the probability density function *dnorm*. We also need a density g from which we can get random samples.

We need to specify $f(x)$,$g(x)$, $g^{-1}(x^*)$ and M

x^*: Use an uniform distribution between 0 and 1 (*runif*)

$f(x)$: Use a normal density function *dnorm*

$g(x)$: Use a naive constant function, in R function(x)return(1) in matlab @(x)1

$g^{-1}(x)$:We want to use a constant function, thus in this case we have a simple linear scaling to get our resulting estimates between -2.5 and 2.5: $g^{-1}(x^*) = 5x^* - 2.5$

m: We have to put m so that $m * g(x) >= f(x)$ for all x. For now let's use $m = 1$ and use $g(x) = constant$

u: Sampled from $unif(0,1)$

x: are random samples from $g(x)$ (here we need g^{-1} from above and the random sampling exercise from last time)

Now we are ready to use:

```
accept_reject = u < (f(x)/(M*g(x)))
```

This simplifies in our example to:

```
accept_reject = u < f(x)
```

If accept_reject is true, we take x as a random sample from $f(x)$, if we reject, we throw it away.

Use 10^4 samples. Plot a histogram and the mean acceptance rate.

We can improve the efficiency of this algorithm by using a smarter m and a better g. What is the optimal number to improve the accept_reject ratio? Hint: Remember that the maximum of a normal density function is at 0

Try out a m value of 0.2. How does the histogram look now?

We can also change $g(x)$. Use the function g(x) = −abs(0.15∗x)+0.5 Which is shaped like a pyramid/triangle/wedge. Changing the function g changes which parts of f we sample most often, this is also why we later have to normalize f by g in the $u <= \frac{f(x)}{M*g(x)}$ part to remove the effect of this biased sampling again.

$$g^{-1}(x) = -10/3 + 20/3 \cdot \begin{cases} 1 - \sqrt{0.5x}, & \text{if } x < 0.5 \\ \sqrt{0.5 - 0.5x}, & \text{if } x \geq 0.5 \end{cases} \tag{1}$$

You can see the functions in the following plot:

```
library(ggplot2)
x = seq(-2.5,2.5,0.01);
d = data.frame(x=x,y1=dnorm(x),y2=-abs(0.15*x)+0.5)
ggplot(d,aes(x=x))+
  geom_point(aes(y=y1),color='red')+
  geom_point(aes(y=y2),color='green')+
  geom_line(aes(y=1),color='purple')+
  theme_minimal()
```

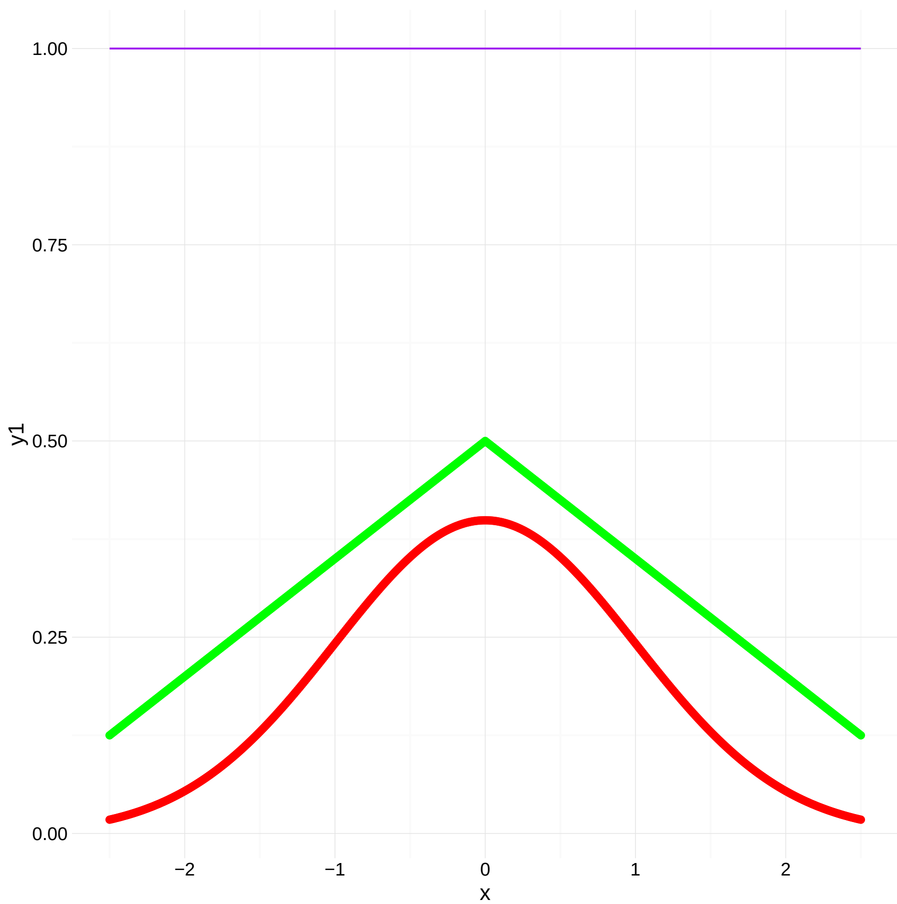

Abbildung 1: Red is the unknown density we want to estimate, purple was g(x) in the first exercise, green is the $g(x)$ we want now.