STA3127 Statistical Computing

Jeong Geonwoo

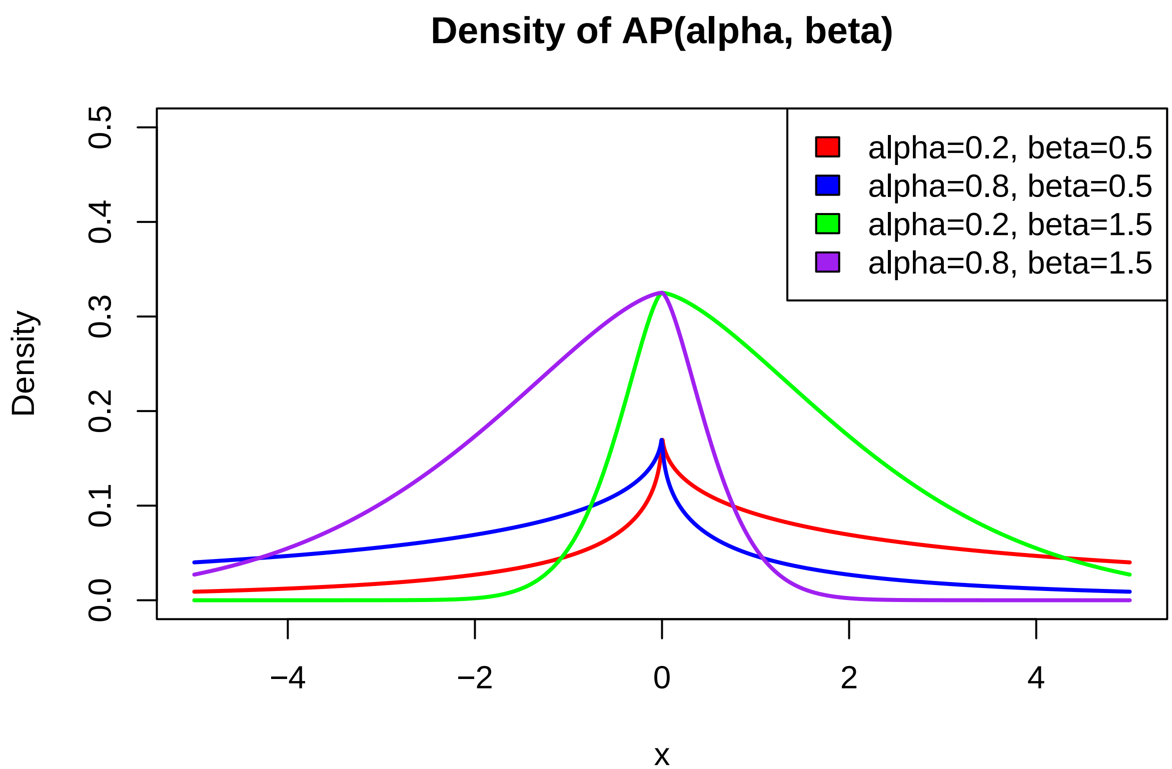
DUE Monday, October 16

```r
rm(list=ls())
set.seed(2018122062)
```

Q1

i) To visualize the shape of the density function, draw the density curves with (0.2, 0.5), (0.8, 0.5), (0.2, 1.5), and (0.8, 1.5) for (α, β) in a single plot.

```r
AP <- function(x, alpha, beta) {
  delta <- 2 * alpha^beta * (1 - alpha)^beta / (alpha^beta + (1 - alpha)^beta)

  ifelse(x <= 0,
         (delta^(1/beta)/gamma(1 + 1/beta)) * exp(-delta/alpha^beta * abs(x)^beta),
         (delta^(1/beta)/gamma(1 + 1/beta)) * exp(-delta/(1 - alpha)^beta * abs(x)^beta)
  )
}

x_range <- seq(-5, 5, length.out = 1000)

params <- list(c(0.2, 0.5), c(0.8, 0.5), c(0.2, 1.5), c(0.8, 1.5))
colors <- c("red", "blue", "green", "purple")

plot(NULL, xlim=c(-5,5), ylim=c(0, 0.5), xlab="x", ylab="Density",
     main="Density of AP(alpha, beta)")

for (i in 1:length(params)) {
  lines(x_range, sapply(x_range, AP, alpha=params[[i]][1], beta=params[[i]][2]),
        col=colors[i], lwd=2)
}

legend_labels <- sprintf("alpha=%.1f, beta=%.1f",
                         sapply(params, `[[`, 1), sapply(params, `[[`, 2))
legend("topright", legend=legend_labels, fill=colors)
```



ii) Show that if $\beta = 1$, the density is reduced to that of an asymmetric Laplace distribution discussed in the previous assignment. Specify the values of m, λ, and κ that match the expressions $\mathrm{AP}(\alpha, 1)$ and $\mathrm{AL}(m, \lambda, \kappa)$.

[Note: This means that you can generate random variables from $\mathrm{AP}(\alpha, 1)$ using your algorithm in the previous assignment.]

$$
\begin{aligned}
& AL\left(x; m = 0, \lambda = 2\sqrt{\alpha(1-\alpha)}, k = \sqrt{\frac{\alpha}{1-\alpha}}\right) \\
&= \left(\frac{2\sqrt{\alpha(1-\alpha)}}{\sqrt{\frac{\alpha}{1-\alpha}} + \sqrt{\frac{1-\alpha}{\alpha}}}\right) \exp\left(-2\sqrt{\alpha(1-\alpha)}x \cdot \mathrm{sign}(x) \sqrt{\frac{\alpha}{1-\alpha}}^{\mathrm{sign}(x)}\right) \\
&= \begin{cases} 2\alpha(1-\alpha)\exp\left(-2\sqrt{\alpha(1-\alpha)}|x|\sqrt{\frac{1-\alpha}{\alpha}}\right), & x \leqslant 0 \\ 2\alpha(1-\alpha)\exp\left(-2\sqrt{\alpha(1-\alpha)}|x|\sqrt{\frac{\alpha}{1-\alpha}}\right), & x > 0 \end{cases} \\
&= \begin{cases} 2\alpha(1-\alpha)\exp(2(1-\alpha)x), & x \leqslant 0 \\ 2\alpha(1-\alpha)\exp(-2\alpha x), & x > 0 \end{cases}
\end{aligned}
$$

$$
\begin{aligned}
&AP\left(x;\alpha,\beta=1\right)\\
&=\begin{cases}\frac{\delta_{\alpha,\beta}^{1/\beta}}{\Gamma(1+1/\beta)}\exp\left(-\frac{\delta_{\alpha,\beta}}{\alpha^\beta}|x|^\beta\right), & x\le 0\\ \frac{\delta_{\alpha,\beta}^{1/\beta}}{\Gamma(1+1/\beta)}\exp\left(-\frac{\delta_{\alpha,\beta}}{(1-\alpha)^\beta}|x|^\beta\right), & x>0\end{cases}\\
&where\quad \alpha\in(0,1),\beta=1,\ \delta_{\alpha,\beta}=2\alpha^\beta(1-\alpha)^\beta/\left[\alpha^\beta+(1-\alpha)^\beta\right]\\
&=\begin{cases}2\alpha(1-\alpha)\exp(-2(1-\alpha)|x|), & x\leqslant 0\\ 2\alpha(1-\alpha)\exp(-2\alpha|x|), & x>0\end{cases}\\
&=\begin{cases}2\alpha(1-\alpha)\exp(2(1-\alpha)x), & x\leqslant 0\\ 2\alpha(1-\alpha)\exp(-2\alpha x), & x>0\end{cases}
\end{aligned}
$$

From above result, we can say that the density of $AP\left(x;\alpha,\beta=1\right)$ is reduced to the density of $AL\left(x;m=0,\lambda=2\sqrt{\alpha(1-\alpha)},k=\sqrt{\frac{\alpha}{1-\alpha}}\right)$

iii) Now, suppose we wish to generate random variables from $\mathrm{AP}(\alpha,\beta),\beta\neq 1$**, using the rejection method with a proposal distribution** $\mathrm{AP}(\gamma,1),\gamma\in(0,1)$**. Denote the proposal density by** g**. Assuming** $\beta>1$**, derive** $c\equiv c(\alpha,\beta,\gamma)=\sup_x[f(x)/g(x)]$ **as a function of** α,β**, and** γ**. Observe what happens if** $\beta<1$**.**

$$
AP(x;\alpha,\beta)=f(x)=\begin{cases}\left(\frac{2\alpha^\beta(1-\alpha)^\beta}{(\alpha^\beta+(1-\alpha)^\beta)}\right)^{1/\beta}\cdot\frac{1}{\Gamma\left(1+\frac{1}{\beta}\right)}\cdot\exp\left(-\frac{2(1-\alpha)^\beta}{(\alpha^\beta+(1-\alpha)^\beta)}(-x)^\beta\right), & x\leqslant 0\\ \left(\frac{2\alpha^\beta(1-\alpha)^\beta}{(\alpha^\beta+(1-\alpha)^\beta)}\right)^{1/\beta}\cdot\frac{1}{\Gamma\left(1+\frac{1}{\beta}\right)}\cdot\exp\left(-\frac{2\alpha^\beta}{(\alpha^\beta+(1-\alpha)^\beta)}x^\beta\right), & x>0\end{cases}
$$

$$
AP(\alpha=\gamma,beta=1)=g(x)=\begin{cases}2\gamma(1-\gamma)\cdot\exp(2(1-\gamma)x), & x\le 0\\ 2\gamma(1-\gamma)\cdot\exp(-2\gamma x), & x>0\end{cases}
$$

$$
\frac{f(x)}{g(x)}=\begin{cases}\left(\frac{2\alpha^\beta(1-\alpha)^\beta}{(\alpha^\beta+(1-\alpha)^\beta)}\right)^{1/\beta}\frac{1}{\Gamma\left(1+\frac{1}{\beta}\right)}\frac{1}{2\gamma(1-\gamma)}\exp\left(-\frac{2(1-\alpha)^\beta}{\alpha^\beta+(1-\alpha)^\beta}(-x)^\beta-2(1-\gamma)x\right) & ,x\le 0\\ \left(\frac{2\alpha^\beta(1-\alpha)^\beta}{(\alpha^\beta+(1-\alpha)^\beta)}\right)^{1/\beta}\frac{1}{\Gamma\left(1+\frac{1}{\beta}\right)}\frac{1}{2\gamma(1-\gamma)}\exp\left(-\frac{2\alpha^\beta}{\alpha^\beta+(1-\alpha)^\beta}x^\beta+2\gamma x\right) & ,x>0\end{cases}
$$

$$
\sup_x\frac{f(x)}{g(x)}=\begin{cases}\left(\frac{2\alpha^\beta(1-\alpha)^\beta}{(\alpha^\beta+(1-\alpha)^\beta)}\right)^{1/\beta}\frac{1}{\Gamma\left(1+\frac{1}{\beta}\right)}\frac{1}{2\gamma(1-\gamma)}\exp\left(-\inf_x\left[\frac{2(1-\alpha)^\beta}{\alpha^\beta+(1-\alpha)^\beta}(-x)^\beta+2(1-\gamma)x\right]\right) & ,x\le 0\\ \left(\frac{2\alpha^\beta(1-\alpha)^\beta}{(\alpha^\beta+(1-\alpha)^\beta)}\right)^{1/\beta}\frac{1}{\Gamma\left(1+\frac{1}{\beta}\right)}\frac{1}{2\gamma(1-\gamma)}\exp\left(-\inf_x\left[\frac{2\alpha^\beta}{\alpha^\beta+(1-\alpha)^\beta}x^\beta-2\gamma x\right]\right) & ,x>0\end{cases}
$$

The above statement makes sense because $\left(\frac{2\alpha^\beta(1-\alpha)^\beta}{(\alpha^\beta+(1-\alpha)^\beta)}\right)^{1/\beta}\frac{1}{\Gamma\left(1+\frac{1}{\beta}\right)}\frac{1}{2\gamma(1-\gamma)}>0$ is independent of x and $\sup_x exp(-h(x))=exp(-\inf_x h(x))$ where x is a feasible value.

$$
\begin{aligned}
&\text{Let}\quad t=-x\quad\text{when}\quad x\le 0\\
&\inf_x\left[\frac{2(1-\alpha)^\beta}{\alpha^\beta+(1-\alpha)^\beta}(-x)^\beta+2(1-\gamma)x\right]=\inf_t\left[\frac{2(1-\alpha)^\beta}{\alpha^\beta+(1-\alpha)^\beta}t^\beta-2(1-\gamma)t\right],\quad t\ge 0\\
&\frac{\partial}{\partial t}\left[\frac{2(1-\alpha)^\beta}{\alpha^\beta+(1-\alpha)^\beta}t^\beta-2(1-\gamma)t\right]=\frac{2(1-\alpha)^\beta\cdot\beta}{\alpha^\beta+(1-\alpha)^\beta}t^{\beta-1}+2\gamma-2\overset{\text{set}}{=}0\\
&\therefore t=\left(\frac{\alpha^\beta+(1-\alpha)^\beta}{(1-\alpha)^\beta\cdot\beta}(1-\gamma)\right)^{\frac{1}{\beta-1}},\quad t\ge 0
\end{aligned}
$$

$$
\begin{aligned}
&\text{When} \quad x > 0, \\
&\frac{\partial}{\partial x}\left[\frac{2\alpha^\beta}{\alpha^\beta + (1-\alpha)^\beta} x^\beta - 2\gamma x\right] = \frac{2\alpha^\beta \beta}{\alpha^\beta + (1-\alpha)^\beta} x^{\beta-1} - 2\gamma \overset{\text{set}}{=} 0 \\
&\therefore x = \left(\frac{\alpha^\beta + (1-\alpha)^\beta}{\alpha^\beta \beta}\gamma\right)^{\frac{1}{\beta-1}}, \quad x > 0
\end{aligned}
$$

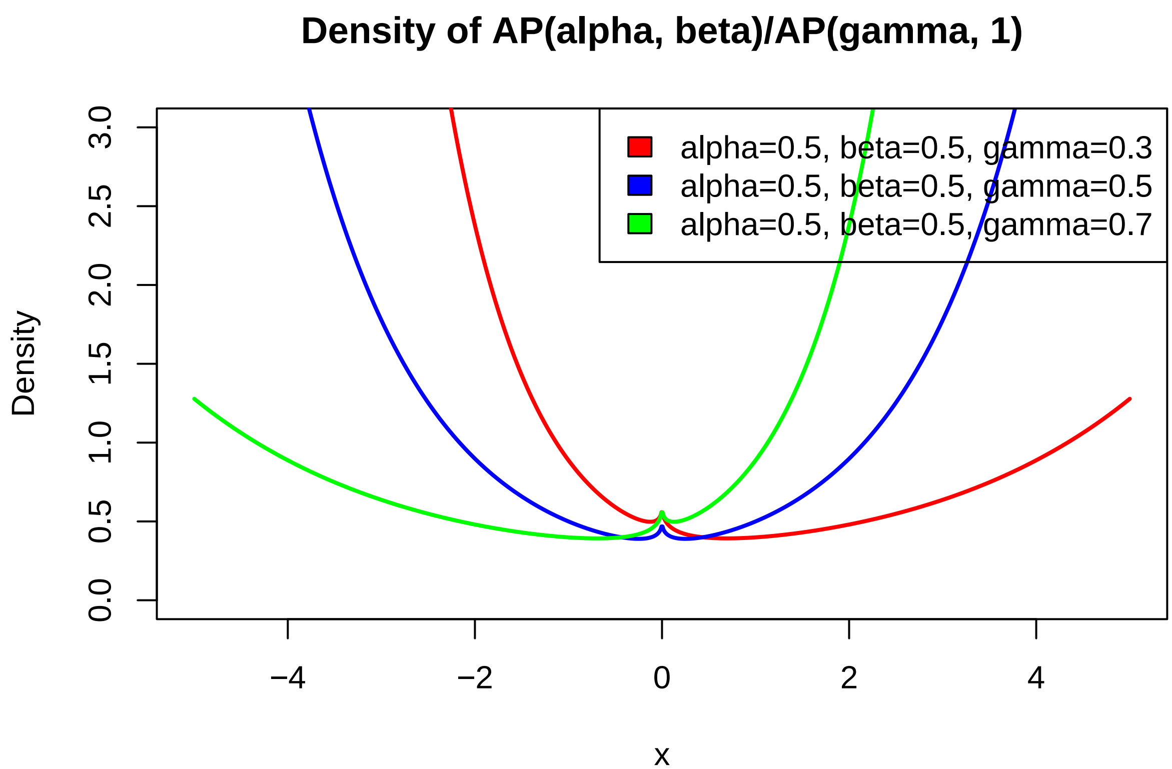
We can get $\sup_x \frac{f(x)}{g(x)}$ by plugging in x obtained from above result.

$$
\sup_x \frac{f(x)}{g(x)} = \begin{cases} \left(\frac{2\alpha^\beta(1-\alpha)^\beta}{(\alpha^\beta+(1-\alpha)^\beta)}\right)^{1/\beta} \frac{1}{\Gamma\left(1+\frac{1}{\beta}\right)} \frac{1}{2\gamma(1-\gamma)} \exp\left(-\inf_x\left[\frac{2(1-\alpha)^\beta}{\alpha^\beta+(1-\alpha)^\beta}(-x)^\beta + 2(1-\gamma)x\right]\right) & , x \le 0 \\ \left(\frac{2\alpha^\beta(1-\alpha)^\beta}{(\alpha^\beta+(1-\alpha)^\beta)}\right)^{1/\beta} \frac{1}{\Gamma\left(1+\frac{1}{\beta}\right)} \frac{1}{2\gamma(1-\gamma)} \exp\left(-\inf_x\left[\frac{2\alpha^\beta}{\alpha^\beta+(1-\alpha)^\beta}x^\beta - 2\gamma x\right]\right) & , x > 0 \end{cases}
$$

$$
\begin{aligned}
&c = c(\alpha,\beta,\gamma) \\
&= \sup_x \frac{f(x)}{g(x)} = \max\left(\delta_{\alpha,\beta}^{1/\beta} \frac{1}{\Gamma\left(1+\frac{1}{\beta}\right)} \frac{1}{2\gamma(1-\gamma)} \exp\left(-\left(\frac{2}{\beta}(1-\gamma) - 2(1-\gamma)\right)\left(\frac{\alpha^\beta + (1-\alpha)^\beta}{(1-\alpha)^\beta \beta}(1-\gamma)\right)^{\frac{1}{\beta-1}}\right),\right. \\
&\left.\delta_{\alpha,\beta}^{1/\beta} \frac{1}{\Gamma\left(1+\frac{1}{\beta}\right)} \frac{1}{2\gamma(1-\gamma)} \exp\left(-\left(\frac{2}{\beta}\gamma - 2\gamma\right)\left(\frac{\alpha^\beta + (1-\alpha)^\beta}{\alpha^\beta \beta}\gamma\right)^{\frac{1}{\beta-1}}\right)\right) \\
&\text{where } \delta_{\alpha,\beta} = 2\alpha^\beta(1-\alpha)^\beta / \left[\alpha^\beta + (1-\alpha)^\beta\right]
\end{aligned}
$$

If $\beta < 1$, $\frac{f(x)}{g(x)}$ has no superior value. Because the gradient $\frac{\partial}{\partial t}\left[\frac{2(1-\alpha)^\beta}{\alpha^\beta+(1-\alpha)^\beta}t^\beta - 2(1-\gamma)t\right] = \frac{2(1-\alpha)^\beta\beta}{\alpha^\beta+(1-\alpha)^\beta}t^{\beta-1} + 2\gamma - 2$ and $\frac{\partial}{\partial x}\left[\frac{2\alpha^\beta}{\alpha^\beta+(1-\alpha)^\beta}x^\beta - 2\gamma x\right] = \frac{2\alpha^\beta\beta}{\alpha^\beta+(1-\alpha)^\beta}x^{\beta-1} - 2\gamma$ obtained from above become smaller, eventually becoming less than zero as x moves away from zero. This means that as x moves away from 0, the proposal distribution $AP(\gamma, 1)$ becomes smaller faster than the target distribution $AP(\alpha,\beta)$. We can see this visually in the results below.

```r
APAP <- function(x, alpha, beta, gamma){
  return (AP(x,alpha,beta) / AP(x,gamma,1))
}

params <- list(c(0.5, 0.5, 0.3), c(0.5, 0.5, 0.5), c(0.5, 0.5, 0.7))
colors <- c("red", "blue", "green")
plot(NULL, xlim=c(-5,5), ylim=c(0, 3), xlab="x", ylab="Density",
     main="Density of AP(alpha, beta)/AP(gamma, 1)")
for (i in 1:length(params)) {
  lines(x_range, sapply(x_range, APAP, alpha=params[[i]][1], beta=params[[i]][2],
                        gamma=params[[i]][3]), col=colors[i], lwd=2)
}
legend_labels <- sprintf("alpha=%.1f, beta=%.1f, gamma=%.1f",
                         sapply(params, `[[`, 1), sapply(params, `[[`, 2), sapply(params, `[[`, 3))
legend("topright", legend=legend_labels, fill=colors)
```



iv) For any given $\beta > 1$, show that γ chosen as $\gamma = \alpha$ optimizes the constant c to achieve the highest efficiency in the rejection method, that is, $\mathrm{AP}(\alpha, 1)$ is the best proposal distribution for $\mathrm{AP}(\alpha, \beta), \beta > 1$, among the class of $\mathrm{AP}(\gamma, 1), \gamma \in (0, 1)$.

The result from iii) which is

$$\sup_x \frac{f(x)}{g(x)} = \max\left(\delta_{\alpha,\beta}^{1/\beta} \frac{1}{\Gamma\left(1+\frac{1}{\beta}\right)} \frac{1}{2\gamma(1-\gamma)} \exp\left(-\left(\frac{2}{\beta}(1-\gamma) - 2(1-\gamma)\right)\left(\frac{\alpha^\beta + (1-\alpha)^\beta}{(1-\alpha)^\beta \beta}(1-\gamma)\right)^{\frac{1}{\beta-1}}\right),\right.$$

$$\left.\delta_{\alpha,\beta}^{1/\beta} \frac{1}{\Gamma\left(1+\frac{1}{\beta}\right)} \frac{1}{2\gamma(1-\gamma)} \exp\left(-\left(\frac{2}{\beta}\gamma - 2\gamma\right)\left(\frac{\alpha^\beta + (1-\alpha)^\beta}{\alpha^\beta \beta}\gamma\right)^{\frac{1}{\beta-1}}\right)\right)$$

$$\text{where } \delta_{\alpha,\beta} = 2\alpha^\beta(1-\alpha)^\beta / \left[\alpha^\beta + (1-\alpha)^\beta\right]$$

means that :

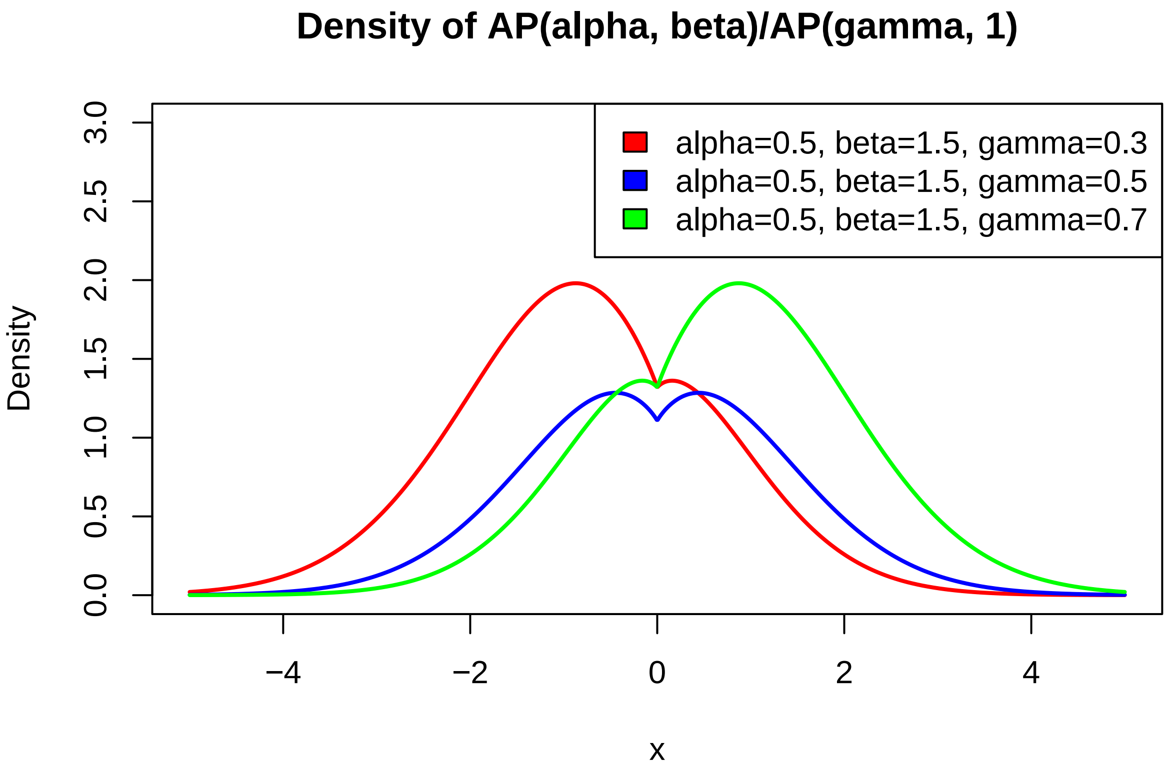
if $\alpha > \gamma$, $c = \sup_x \frac{f(x)}{g(x)} = \left(\frac{2\alpha^\beta(1-\alpha)^\beta}{(\alpha^\beta+(1-\alpha)^\beta)}\right)^{1/\beta} \frac{1}{\Gamma(1+\frac{1}{\beta})} \frac{1}{2\gamma(1-\gamma)} \exp\left(-\left(\frac{2}{\beta}(1-\gamma) - 2(1-\gamma)\right)\left(\frac{\alpha^\beta+(1-\alpha)^\beta}{(1-\alpha)^\beta\beta}(1-\gamma)\right)^{\frac{1}{\beta-1}}\right)$

if $\alpha = \gamma$, $c = \sup_x \frac{f(x)}{g(x)} = \left(\frac{2\alpha^\beta(1-\alpha)^\beta}{(\alpha^\beta+(1-\alpha)^\beta)}\right)^{1/\beta} \frac{1}{\Gamma(1+\frac{1}{\beta})} \frac{1}{2\gamma(1-\gamma)} \exp\left(-\left(\frac{2}{\beta}(1-\gamma) - 2(1-\gamma)\right)\left(\frac{\alpha^\beta+(1-\alpha)^\beta}{(1-\alpha)^\beta\beta}(1-\gamma)\right)^{\frac{1}{\beta-1}}\right)$
is equal to $c = \sup_x \frac{f(x)}{g(x)} = \left(\frac{2\alpha^\beta(1-\alpha)^\beta}{(\alpha^\beta+(1-\alpha)^\beta)}\right)^{1/\beta} \frac{1}{\Gamma(1+\frac{1}{\beta})} \frac{1}{2\gamma(1-\gamma)} \exp\left(-\left(\frac{2}{\beta}\gamma - 2\gamma\right)\left(\frac{\alpha^\beta+(1-\alpha)^\beta}{\alpha^\beta\beta}\gamma\right)^{\frac{1}{\beta-1}}\right)$

if $\alpha < \gamma$, $c = \sup_x \frac{f(x)}{g(x)} = \left(\frac{2\alpha^\beta(1-\alpha)^\beta}{(\alpha^\beta+(1-\alpha)^\beta)}\right)^{1/\beta} \frac{1}{\Gamma(1+\frac{1}{\beta})} \frac{1}{2\gamma(1-\gamma)} \exp\left(-\left(\frac{2}{\beta}\gamma - 2\gamma\right)\left(\frac{\alpha^\beta+(1-\alpha)^\beta}{\alpha^\beta\beta}\gamma\right)^{\frac{1}{\beta-1}}\right)$

Above expression means that when $\alpha > \gamma$, $f(x)/g(x)$ has a superior value when $x \leq 0$. When $\alpha < \gamma$, in contrast, $f(x)/g(x)$ has a superior value when $x > 0$, with fixed α, β in both cases. We can see this visually in the results below.

```r
params <- list(c(0.5, 1.5, 0.3), c(0.5, 1.5, 0.5), c(0.5, 1.5, 0.7))
colors <- c("red", "blue", "green")
plot(NULL, xlim=c(-5,5), ylim=c(0, 3), xlab="x", ylab="Density",
     main="Density of AP(alpha, beta)/AP(gamma, 1)")
for (i in 1:length(params)) {
  lines(x_range, sapply(x_range, APAP, alpha=params[[i]][1], beta=params[[i]][2],
                        gamma=params[[i]][3]), col=colors[i], lwd=2)
}
legend_labels <- sprintf("alpha=%.1f, beta=%.1f, gamma=%.1f",
                         sapply(params, `[[`, 1), sapply(params, `[[`, 2), sapply(params, `[[`, 3))
legend("topright", legend=legend_labels, fill=colors)
```

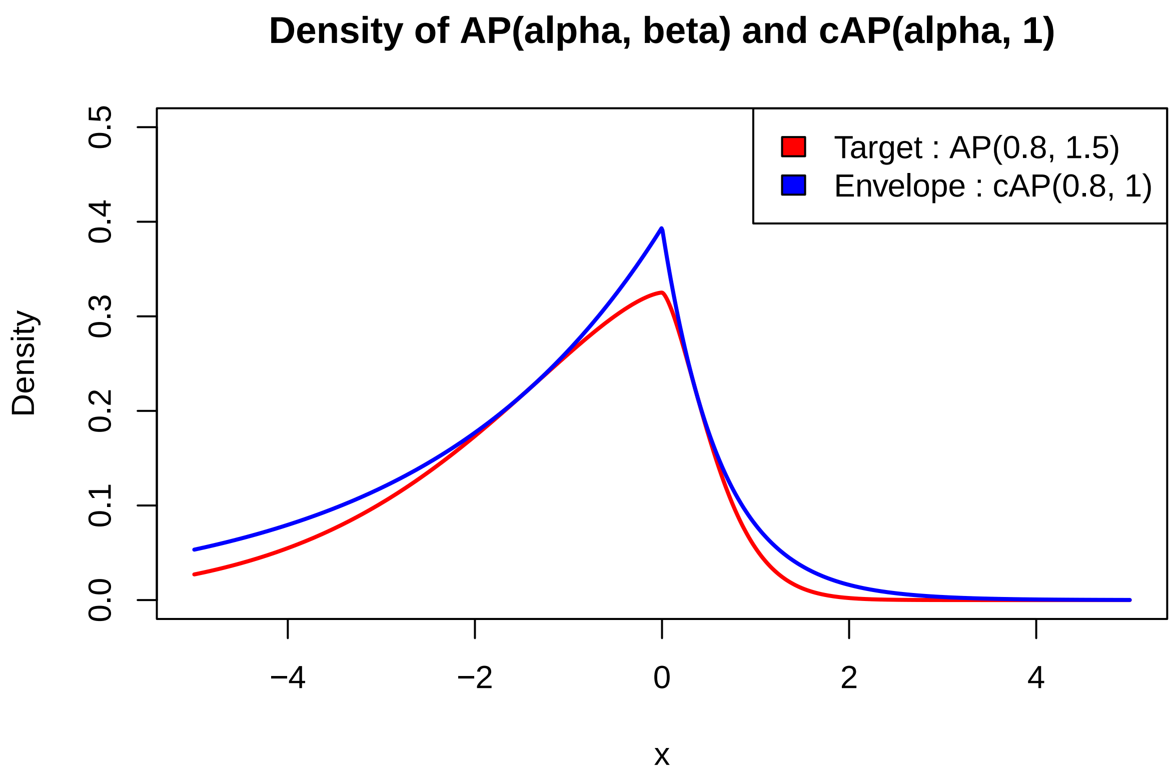


In conclusion, AP(α,1) is the best proposal distribution for AP(α,β), $\beta > 1$, among the class of AP(γ,1), $\gamma \in (0, 1)$.

v) For (α,β) = (0.8,1.5), overlay the target density $f(x)$ and the envelope $cg(x)$ with the best c to compare the two functions.

```r
alpha = 0.8
beta = 1.5

best_c <- function(alpha, beta){
  delta <- 2 * alpha^beta * (1 - alpha)^beta / (alpha^beta + (1 - alpha)^beta)
  g_constant <- gamma(1 + 1/beta) * 2*alpha*(1-alpha)
  exp_part1 <- (2/beta * alpha - 2*alpha)
  exp_part2 <- alpha * (alpha^beta + (1-alpha)^beta) / alpha^beta / beta
  return (delta^(1/beta) / g_constant * exp(-exp_part1*exp_part2^(1/(beta-1))))
}
```

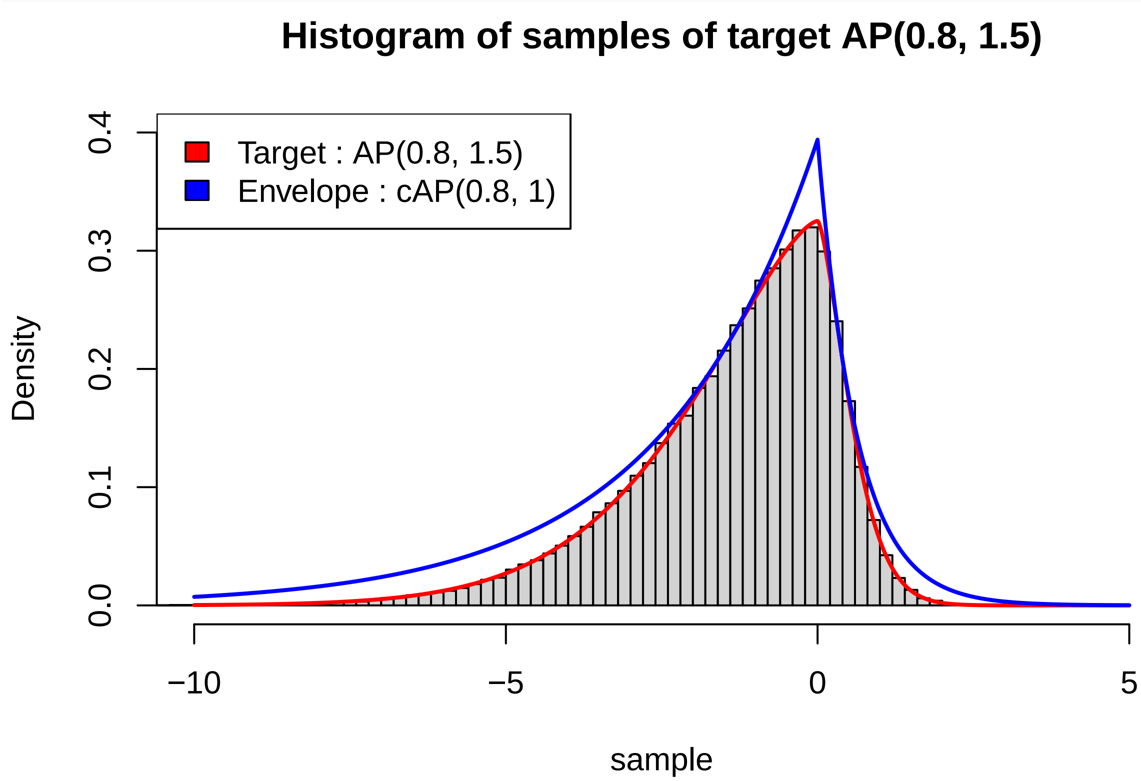
```
envelope <- function(x, alpha, beta){
  return (best_c(alpha, beta) * AP(x, alpha, beta=1))
}

colors <- c("red", "blue")
plot(NULL, xlim=c(-5,5), ylim=c(0, 0.5), xlab="x", ylab="Density",
     main="Density of AP(alpha, beta) and cAP(alpha, 1)")
lines(x_range, sapply(x_range, AP, alpha=alpha, beta=beta), col=colors[1], lwd=2)
lines(x_range, sapply(x_range, envelope, alpha=alpha, beta=beta), col=colors[2], lwd=2)
legend_labels <- c("Target : AP(0.8, 1.5)", "Envelope : cAP(0.8, 1)")
legend("topright", legend=legend_labels, fill=colors)
```



vi) Using the constructed rejection sampling algorithm above, generate 10^5 draws from $AP(0.8, 1.5)$, and overlay the true density function and a histogram with a suitable bin size. Also, estimate the acceptance probability of the algorithm based on your simulation, and compare it with the true value.

```
AL_cdf_inv <- function(U, m, lambda, kappa) {
  sapply(U, function(Ui) {
    if (Ui < kappa^2 / (1 + kappa^2)) {
      cdf_inv <- kappa / lambda * log((1 + kappa^2) / kappa^2 * Ui) + m
    } else {
      cdf_inv <- - 1 / (kappa * lambda) * log((1 + kappa^2)*(1-Ui)) + m
    }
    return(cdf_inv)
  })
}

n_samples = 10^5
```

```r
samples_target = rep(NA, n_samples)
now_index = 1
reject = 0
bestc = best_c(alpha, beta)

while (is.na(samples_target[n_samples])){
  sample_proposal <- AL_cdf_inv(runif(1), m=0, lambda=2*sqrt(alpha*(1-alpha)),
                                kappa=sqrt(alpha/(1-alpha)))
  if (runif(1) <= AP(sample_proposal, alpha=alpha, beta=beta)
      / (bestc * AP(sample_proposal, alpha=alpha, beta=1))){
    samples_target[now_index] = sample_proposal
    now_index = now_index + 1
  } else {
    reject = reject + 1
  }
}

hist(samples_target, xlim=c(-10,5), ylim=c(0,0.4), breaks = 100, freq=FALSE,
     main="Histogram of samples of target AP(0.8, 1.5)", xlab="sample")
lines(seq(-10, 5, length.out = 10000), sapply(seq(-10, 5, length.out = 10000),
                                              AP, alpha=alpha, beta=beta), col=colors[1] ,lwd=2)
lines(seq(-10, 5, length.out = 10000), sapply(seq(-10, 5, length.out = 10000),
                                              envelope, alpha=alpha, beta=beta), col=colors[2] ,lwd=2)
legend_labels <- c("Target : AP(0.8, 1.5)", "Envelope : cAP(0.8, 1)")
legend("topleft", legend=legend_labels, fill=colors)
```




```r
cat("Acceptance probability of the algorithm:", n_samples / (n_samples+reject),"\n",
"True acceptance probability:", 1/best_c(0.8, 1.5))
```

```
## Acceptance probability of the algorithm: 0.811234
```

```
##  True acceptance probability: 0.8122986
```

Q2

```r
rm(list=ls())
```

i) Consider sampling from $\mathrm{TN}_A\left(\mu, \sigma^2\right)$ using the rejection method with a proposal distribution $\mathrm{N}\left(\mu, \sigma^2\right)$. Let f be the target density and g be the proposal density. Derive the expression of $f(x)/[cg(x)]$, where $c = \sup_x[f(x)/g(x)]$.

$$
\begin{aligned}
g(x) &= \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) \\
f(x) &= \frac{g(x)}{P(x \in A)} \mathbb{I}(x \in A)
\end{aligned}
$$

As $f(x) \leqslant \frac{g(x)}{P(x \in A)}$, we can say $\frac{f(x)}{g(x)} \leqslant \frac{1}{P(x \in A)}$

From that, $c = \sup_x\left[\frac{f(x)}{g(x)}\right] = \frac{1}{p(x \in A)}$.

$$
\begin{aligned}
\text{therefore,} \quad \frac{f(x)}{cg(x)} &= \frac{\frac{g(x)}{p(x \in A)}\mathbb{I}(x \in A)}{\frac{1}{p(x \in A)}g(x)} \\
&= \mathbb{I}(x \in A)
\end{aligned}
$$

ii) Find the acceptance probability of the algorithm in i) in terms of a normal probability.

Since the acceptance probability $= \frac{1}{c}$,

$$
\begin{aligned}
P(acceptance) = \frac{1}{c} &= P(x \in A) \\
&= \int_{x \in A} \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) dx
\end{aligned}
$$

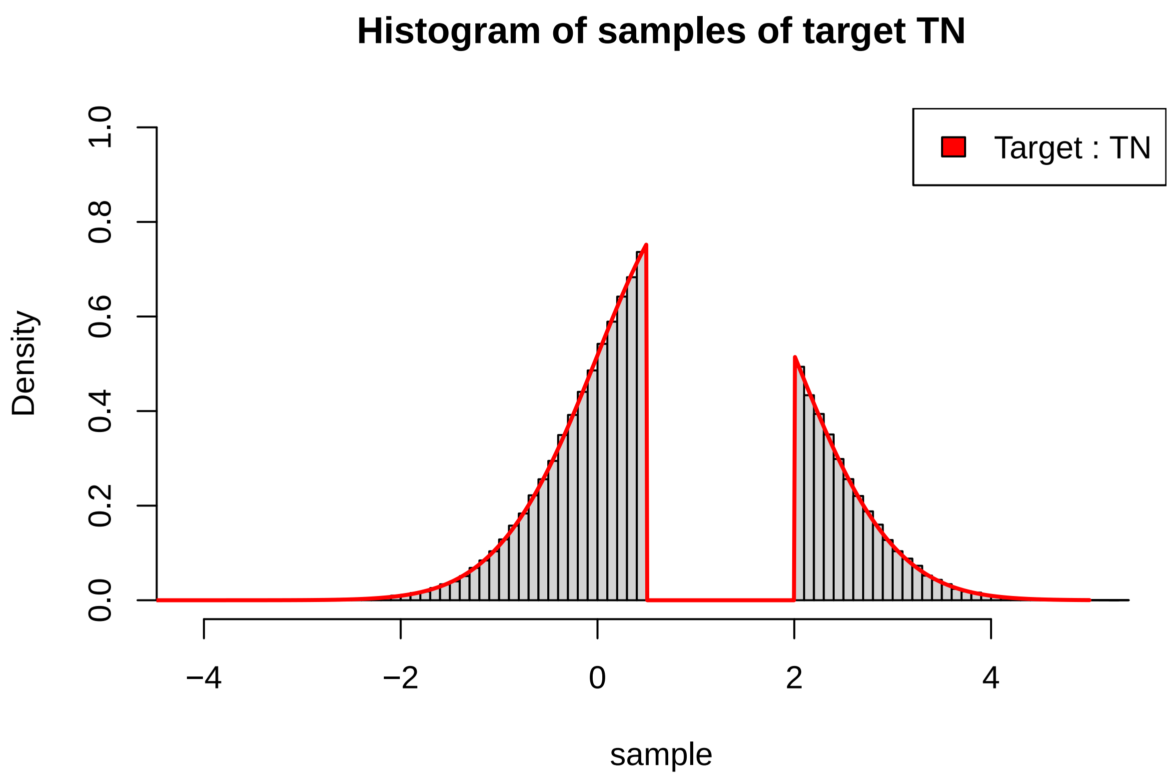
iii) Generate 10^5 observations from $\mathrm{TN}_{(-\infty,0.5)\cup(2,\infty)}(1,1)$ using the rejection sampling with a proposal distribution $\mathrm{N}(1,1)$. Use the qnorm function to sample from the proposal distribution. Compare the estimated acceptance probability obtained from your simulation with the true acceptance probability. Draw a histogram of the obtained values with a suitable bin size.

```r
x_range <- seq(-5, 5, length.out = 1000)

g <- function(x, mu, sigma){
  # dnorm
  return (1/sqrt(2*pi)/sigma * exp(-(x-mu)^2 / 2*sigma^2))
}

f <- function(x, a, b, mu, sigma){
  # truncated dnorm
  dnorm_part <- (1/sqrt(2*pi)/sigma * exp(-(x-mu)^2 / 2*sigma^2))
  normalizing <- 1 - pnorm(b,mu,sigma) + pnorm(a,mu,sigma)
  indicator <- 1*(x < a) + 1*(x > b)
  return (dnorm_part / normalizing * indicator)
}
```

```r
envelop <- function(x, mu, sigma){
  return (c_ * g(x,mu,sigma))
}

n_samples = 10^5
samples_target = rep(NA, n_samples)
now_index = 1
reject = 0
c_ = 1 / (1 - pnorm(2, 1, 1) + pnorm(0.5, 1, 1))

while (is.na(samples_target[n_samples])){
  sample_proposal <- qnorm(runif(1), 1, 1)
  if (runif(1) <= f(sample_proposal, 0.5, 2, 1, 1)
      / g(sample_proposal, 1, 1) / c_){
    samples_target[now_index] = sample_proposal
    now_index = now_index + 1
  } else {
    reject = reject + 1
  }
}

cat("Estimated acceptance probability:",
    length(samples_target)/(length(samples_target)+reject),"\n",
"True acceptance probability:",1/c_)
```

```
## Estimated acceptance probability: 0.4673203
##  True acceptance probability: 0.4671928
```

```r
colors <-("red")
hist(samples_target, ylim=c(0,1), breaks = 100, freq = FALSE,
     main="Histogram of samples of target TN", xlab="sample")
lines(x_range, sapply(x_range, f, a=0.5, b=2, mu=1, sigma=1),
      col=colors[1], lwd=2)
legend_labels <- c("Target : TN")
legend("topright", legend=legend_labels, fill=colors)
```



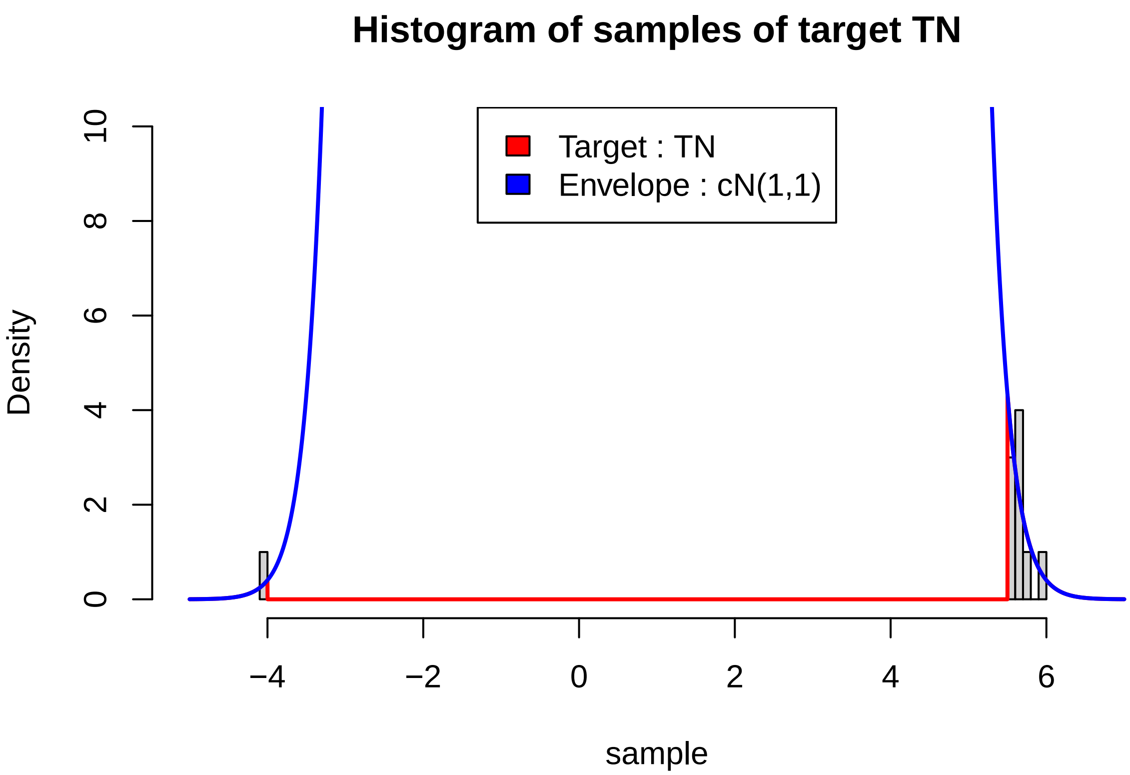
iv) Consider attempting the strategy outlined in i) to sample from $\mathrm{TN}_{(-\infty,-4)\cup(5.5,\infty)}(1,1)$**. What is the issue of this approach?**

If we find an envelope $= cg(x)$ that covers the target distribution, too many samples are rejected because c becomes too large. This happens because the target distribution is a truncated normal distribution, where the nonzero region is so far away from μ that the density has large values and then suddenly becomes zero at the border of the nonzero region.

We can understand this intuitively with the visual result below. Even sampling 10 samples takes about 10 seconds.

```
x_range <- seq(-5, 7, length.out = 10000)

envelop <- function(x, mu, sigma){
  return (c_ * g(x,mu,sigma))
}

n_samples = 10 # Even sampling 10 takes about 10 seconds.
samples_target = rep(NA, n_samples)
now_index = 1
reject = 0
c_ = 1 / (1 - pnorm(5.5, 1, 1) + pnorm(-4, 1, 1))

while (is.na(samples_target[n_samples])){
  sample_proposal <- qnorm(runif(1), 1, 1)
  if (runif(1) <= f(sample_proposal, -4, 5.5, 1, 1)
      / g(sample_proposal, 1, 1) / c_){
    samples_target[now_index] = sample_proposal
    now_index = now_index + 1
  } else {
```

```
    reject = reject + 1
  }
}

colors <- c("red", "blue")
hist(samples_target, xlim = c(-5,7), ylim=c(0,10), breaks = 100,
     freq = FALSE, main="Histogram of samples of target TN", xlab="sample")
lines(x_range, sapply(x_range, f, a=-4, b=5.5, mu=1, sigma=1),
      col=colors[1], lwd=2)
lines(x_range, sapply(x_range, envelop, mu=1, sigma=1),
      col=colors[2], lwd=2)
legend_labels <- c("Target : TN", "Envelope : cN(1,1)")
legend("top", legend=legend_labels, fill=colors)
```



v) Instead, find a map g such that $g(U) \sim \text{TN}_{(-\infty,-4)\cup(5.5,\infty)}(1,1)$, where $U \sim$ Unif $(0,1)$, using the inverse transform method. Your g must take single U for its input, that is, $g : (0,1) \to \mathbb{R}$.

Assuming that $X = g(U) \sim \text{TN}_{(-\infty,-4)\cup(5.5,\infty)}(1,1)$,

$$\text{Let } \Phi_{1,1}(x) = pnorm(x,1,1).$$

$$f(x) = \frac{\frac{1}{\sqrt{2\pi}}exp(-\frac{(x-1)^2}{2})}{\Phi_{1,1}(-4) + (1-\Phi_{1,1}(5.5))},\ x \in (-\infty,-4)\cup(5.5,\infty)$$

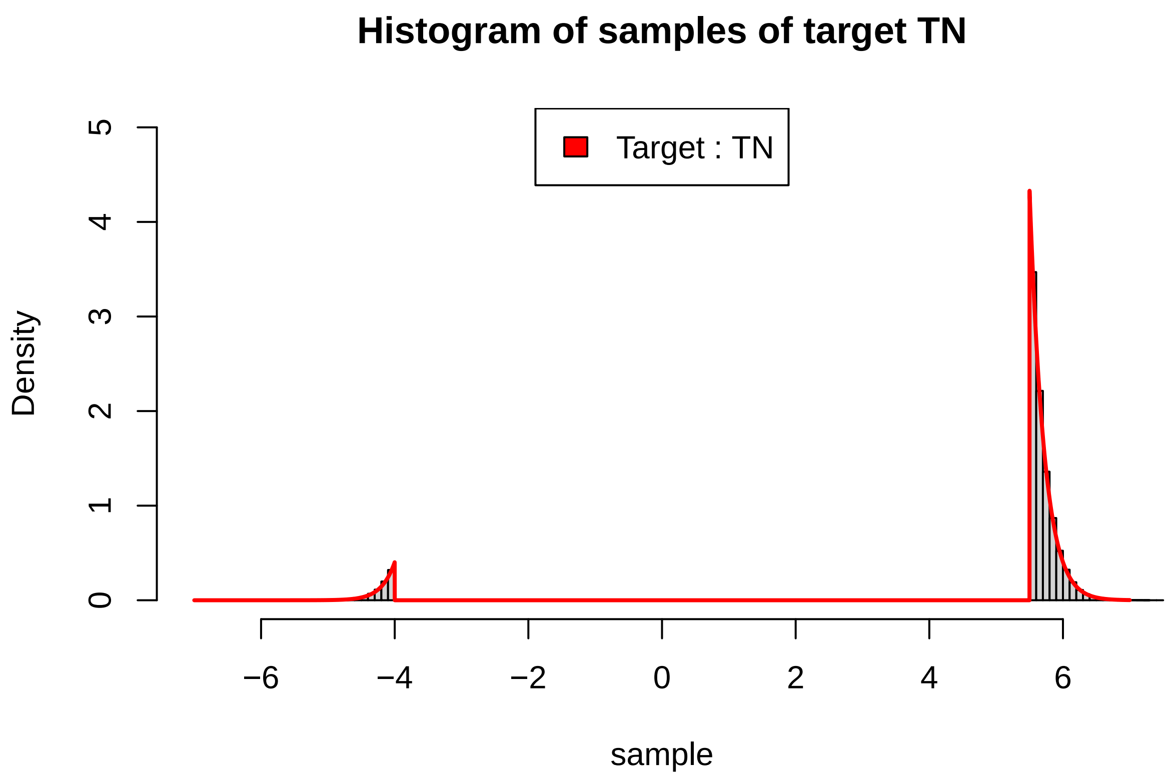
$$F(x) = \int_{-\infty}^{x} f(x)dx$$

$$= \begin{cases} \frac{\Phi_{1,1}(x)}{\Phi_{1,1}(-4)+(1-\Phi_{1,1}(5.5))}, & ,x < -4 \\ \frac{\Phi_{1,1}(-4)}{\Phi_{1,1}(-4)+(1-\Phi_{1,1}(5.5))}, & ,-4 \le x \le 5.5 \\ \frac{\Phi_{1,1}(-4)-\Phi_{1,1}(5.5)+\Phi_{1,1}(x)}{\Phi_{1,1}(-4)+(1-\Phi_{1,1}(5.5))}, & ,x > 5.5 \end{cases}$$

$$
\begin{aligned}
g(U) &= F^{-1}(U) \\
&= \begin{cases} \Phi_{1,1}^{-1}(U\left(\Phi_{1,1}(-4) + (1 - \Phi_{1,1}(5.5))\right)) & , U < \frac{\Phi_{1,1}(-4)}{\Phi_{1,1}(-4)+(1-\Phi_{1,1}(5.5))} \\ \Phi_{1,1}^{-1}(U\left(\Phi_{1,1}(-4) + (1 - \Phi_{1,1}(5.5))\right) - \Phi_{1,1}(-4) + \Phi_{1,1}(5.5)) & , U \geq \frac{\Phi_{1,1}(-4)}{\Phi_{1,1}(-4)+(1-\Phi_{1,1}(5.5))} \end{cases}
\end{aligned}
$$

vi) (C) Using v), generate 10^5 observations from $\mathrm{TN}_{(-\infty,-4)\cup(5.5,\infty)}(1,1)$, and draw a histogram of the obtained values with a suitable bin size. What is the advantage of this approach over using the above rejection method?

```
x_range <- seq(-7, 7, length.out = 10000)

g <- function(U, a, b, mu, sigma) {
  pnorm1 <- pnorm(a, mu, sigma)
  pnorm2 <- pnorm(b, mu, sigma)
  pnorm12 <- 1-pnorm2+pnorm1
  if (U <= pnorm1 / pnorm12) {
    return(qnorm(U * pnorm12, 1, 1))
  } else {
    return(qnorm(pnorm12*U +pnorm2 - pnorm1, 1, 1))
  }
}
n_samples = 10^5
samples_target = rep(NA, n_samples)
U <- runif(n_samples)
for (i in 1:n_samples){
  samples_target[i] <- g(U[i], a=-4, b=5.5, mu=1, sigma=1)
}

colors <- c("red")
hist(samples_target, xlim = c(-7,7), ylim=c(0,5), breaks = 100,
     freq = FALSE, main="Histogram of samples of target TN", xlab="sample")
lines(x_range, sapply(x_range, f, a=-4, b=5.5, mu=1, sigma=1),
      col=colors[1], lwd=2)
legend_labels <- c("Target : TN")
legend("top", legend=legend_labels, fill=colors)
```



The inverse CDF method has the advantage that when the target distribution's densities are separated and far apart, they can be easily sampled as long as the inverse of the CDF is available. Especially when sampling from a truncated normal distribution, for example, the inverse CDF method is much faster than the rejection method.