

STAT 600 - HW 4

Kevin Korsurat

All Rcpp/RcppArmadillo can be found in my [GitHub](#).

Question 1

For this question, I will let t-distribution with 3 df as a density that we can sample from. The plots below shows the distribution of the target distribution $q(x)$, the distribution that we can sample from $g(x)$, and the envelope function based on $g(x)$ with the different value of α .

The density of the distributions

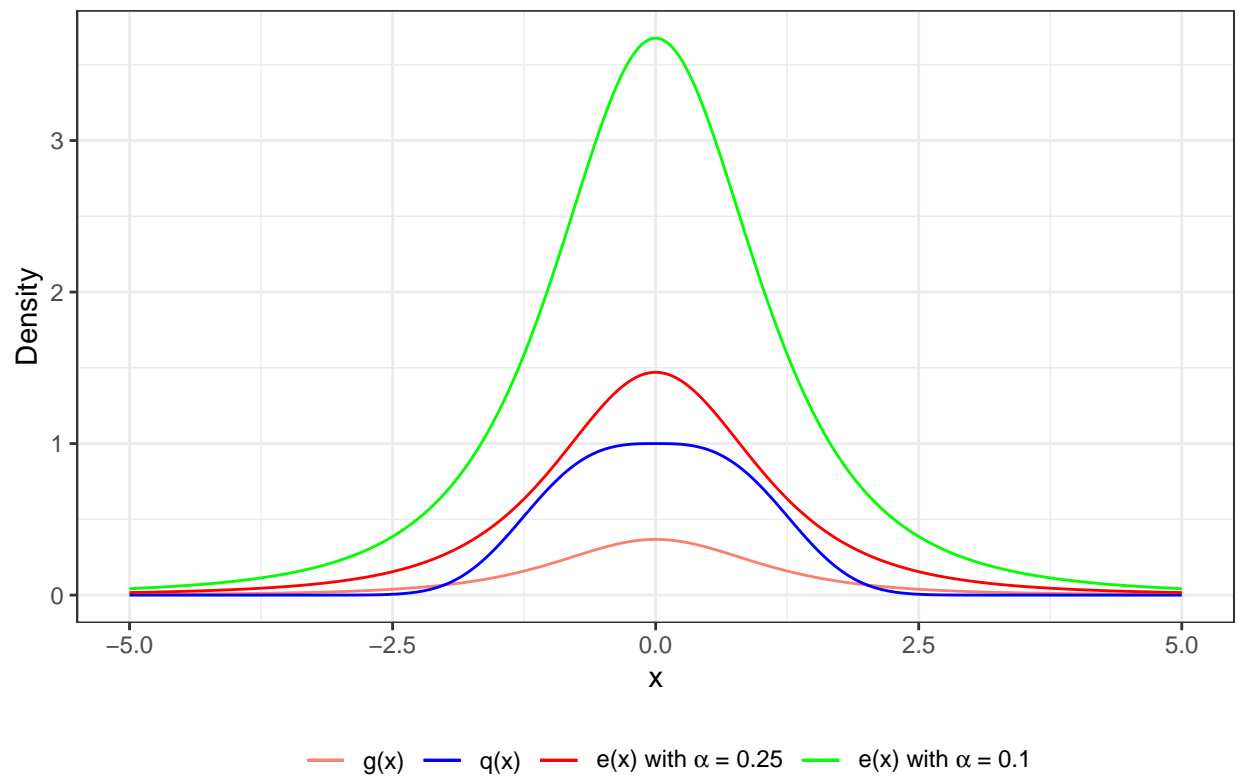


Figure 1: The density for all function used in the analysis.

Figure 2 shows that all of the proposed distribution have a heavier tail compared to the target distribution. Besides, as shown in the Figure 1, the envelope function for both α are greater than the target distribution for the all range of x .

First, we will measure the performance for estimatinf $\sigma^2 = E[X^2]$

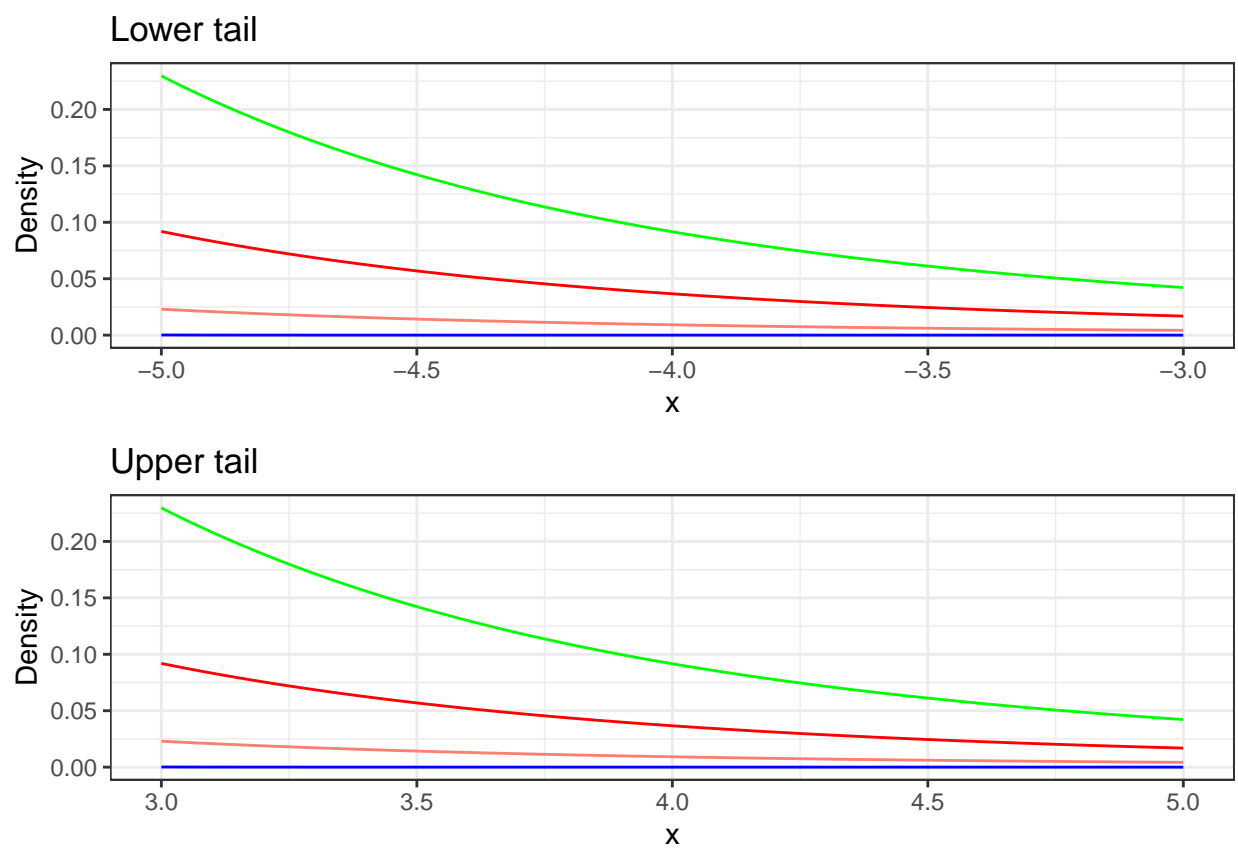


Figure 2: The tail density for all function used in the analysis.

Table 1: The result for estimating $E[X^2]$

n	Method	Estimate	Time
n = 100	Importance Sampling	0.77876 (SD = 0.0812)	5e-05 (SD = 5e-05)
	Rejection Sampling	0.77358 (SD = 0.09397)	7e-05 (SD = 8e-05)
	Sampling Importance Resampling	0.77545 (SD = 0.08866)	1e-04 (SD = 8e-05)
	Philippe and Robert	0.70836 (SD = 0.04281)	9e-05 (SD = 7e-05)
n = 1000	Importance Sampling	0.77693 (SD = 0.02489)	0.00024 (SD = 6e-05)
	Rejection Sampling	0.77649 (SD = 0.02947)	0.00034 (SD = 7e-05)
	Sampling Importance Resampling	0.77977 (SD = 0.04711)	0.00029 (SD = 1e-04)
	Philippe and Robert	0.76732 (SD = 0.00933)	0.00056 (SD = 0.00117)
n = 10000	Importance Sampling	0.77645 (SD = 0.00827)	0.0023 (SD = 0.00117)
	Rejection Sampling	0.77687 (SD = 0.00955)	0.00317 (SD = 0.00018)
	Sampling Importance Resampling	0.77775 (SD = 0.04272)	0.00246 (SD = 0.00149)
	Philippe and Robert	0.77543 (SD = 0.0014)	0.0048 (SD = 0.01091)

For each methods, I will run for 100, 1000, and 10000 times. Note that for the the rejection sampling, I will set α to be 0.25. Besides, for the SIR, I will let the important sampling step sampling for 500 observations.

Question 2

Appendix

```
knitr::opts_chunk$set(echo = FALSE)

library(tidyverse)
library(knitr)
library(kableExtra)
library(Rcpp)
library(RcppArmadillo)
library(devtools)
library(foreach)
library(doParallel)
library(ggplot2)
library(latex2exp)
library(gridExtra)

path <- "/Users/kevin-imac/Desktop/Github - Repo/"
if(! file.exists(path)){
  path <- "/Users/kevinkvp/Desktop/Github Repo/"
}

compileAttributes()
build()
install()
library(HW4MC)

### User-defined functions -----
meanSD <- function(x, dplace = 5){
  mm <- round(mean(x), digits = dplace)
  ss <- round(sd(x), digits = dplace)
  paste0(mm, " (SD = ", ss, ")")
}

### Plots
x <- seq(-5, 5, 0.01)
targetD <- sapply(x, tg_dist, a = 3, b = 3)
sampD <- dt(x, 3)

data.frame(x, g = sampD, q = targetD) %>%
  ggplot(aes(x = x)) +
  geom_line(aes(y = sampD, color = "A1")) +
  geom_line(aes(y = targetD, color = "A2")) +
  geom_line(aes(y = sampD/0.25, color = "A3")) +
  geom_line(aes(y = sampD/0.1, color = "A4")) +
  theme_bw() +
  theme(legend.position = "bottom") +
  labs(x = "x", y = "Density", color = " ", title = "The density of the distributions") +
  scale_color_manual(values = c("salmon", "blue", "red", "green"),
    labels = c("g(x)", "q(x)",
      TeX("e(x) with  $\\alpha = 0.25$ "),
      TeX("e(x) with  $\\alpha = 0.1$ ")))

x <- seq(-5, -3, 0.01)
```

```

targetD <- sapply(x, tg_dist, a = 3, b = 3)
sampD <- dt(x, 3)

p2 <- data.frame(x, g = sampD, q = targetD) %>%
  ggplot(aes(x = x)) +
  geom_line(aes(y = sampD, color = "A1")) +
  geom_line(aes(y = targetD, color = "A2")) +
  geom_line(aes(y = sampD/0.25, color = "A3")) +
  geom_line(aes(y = sampD/0.1, color = "A4")) +
  theme_bw() +
  theme(legend.position = "none") +
  labs(x = "x", y = "Density", color = " ", title = "Lower tail") +
  scale_color_manual(values = c("salmon", "blue", "red", "green"),
    labels = c("Distribution that we can sample from, g(x)",
      "Target Distribution, q(x)",
      TeX("Envelope Function with  $\alpha = 0.25$ "),
      TeX("Envelope Function with  $\alpha = 0.1$ ")))

x <- seq(3, 5, 0.01)
targetD <- sapply(x, tg_dist, a = 3, b = 3)
sampD <- dt(x, 3)

p3 <- data.frame(x, g = sampD, q = targetD) %>%
  ggplot(aes(x = x)) +
  geom_line(aes(y = sampD, color = "A1")) +
  geom_line(aes(y = targetD, color = "A2")) +
  geom_line(aes(y = sampD/0.25, color = "A3")) +
  geom_line(aes(y = sampD/0.1, color = "A4")) +
  theme_bw() +
  theme(legend.position = "none") +
  labs(x = "x", y = "Density", color = " ", title = "Upper tail") +
  scale_color_manual(values = c("salmon", "blue", "red", "green"),
    labels = c("Distribution that we can sample from, g(x)",
      "Target Distribution, q(x)",
      TeX("Envelope Function with  $\alpha = 0.25$ "),
      TeX("Envelope Function with  $\alpha = 0.1$ ")))

grid.arrange(grobs = list(p2, p3))

### Function: Run the model for each m and function of interests
runAll <- function(fx, n, seed = 1){

  ### (a): Importance Sampling
  set.seed(seed, kind = "L'Ecuyer-CMRG")
  registerDoParallel(5)
  impResult <- foreach(t = 1:1000, .combine = "rbind") %dopar% {
    startT <- Sys.time()
    estimate <- impSamp(h = fx, n = n, a = 3, b = 3, df = 3, showCalc = FALSE)
    totalT <- difftime(Sys.time(), startT)
    c(estimate, totalT)
  }
  stopImplicitCluster()
}

```

```

### (b): Rejection Sampling (alpha = 0.25)
set.seed(seed, kind = "L'Ecuyer-CMRG")
registerDoParallel(5)
rejResult <- foreach(t = 1:1000, .combine = "rbind") %dopar% {
  startT <- Sys.time()
  estimate <- rejSamp(h = fx, n = n, alp = 0.25, a = 3, b = 3, df = 3, showSamp = FALSE)
  totalT <- difftime(Sys.time(), startT)
  c(estimate, totalT)
}
stopImplicitCluster()

### (c): Sampling Importance Resampling
set.seed(seed, kind = "L'Ecuyer-CMRG")
registerDoParallel(5)
sirResult <- foreach(t = 1:1000, .combine = "rbind") %dopar% {
  startT <- Sys.time()
  estimate <- SIRSamp(h = fx, n = n, M = 500, a = 3, b = 3, df = 3)
  totalT <- difftime(Sys.time(), startT)
  c(estimate, totalT)
}
stopImplicitCluster()

### (d): Philippe and Robert
set.seed(seed, kind = "L'Ecuyer-CMRG")
registerDoParallel(5)
phroResult <- foreach(t = 1:1000, .combine = "rbind") %dopar% {
  startT <- Sys.time()
  estimate <- PhRoMethod(h = fx, n = n, alp = 0.25, a = 3, b = 3, df = 3)
  totalT <- difftime(Sys.time(), startT)
  c(estimate, totalT)
}
stopImplicitCluster()

list(impResult = impResult, rejResult = rejResult,
     sirResult = sirResult, phroResult = phroResult)
}

### Run for E[X^2]
x2n100 <- runAll(fx = function(x){x^2}, n = 100, seed = 1)
x2n1000 <- runAll(fx = function(x){x^2}, n = 1000, seed = 1)
x2n10000 <- runAll(fx = function(x){x^2}, n = 10000, seed = 1)

### Function: ggplot for all result

### Result: E[X^2]
data.frame(c(rep("n = 100", 4), rep("n = 1000", 4), rep("n = 10000", 4)),
           rep(c("Importance Sampling", "Rejection Sampling",
                "Sampling Importance Resampling", "Philippe and Robert"), 3),
           rbind(t(sapply(1:4, function(x){apply(x2n100[[x]], 2, meanSD)})),
                t(sapply(1:4, function(x){apply(x2n1000[[x]], 2, meanSD)})),
                t(sapply(1:4, function(x){apply(x2n10000[[x]], 2, meanSD)})))) %>%
kable(align = c("c", "l", "c", "c"), col.names = c("n", "Method", "Estimate", "Time"),

```

```
booktabs = TRUE, caption = "The result for estimating  $E[X^2]$ ") %>%  
kable_paper(full_width = FALSE) %>%  
column_spec(1, bold = TRUE) %>%  
collapse_rows(columns = 1, valign = "top")
```