統計模擬 作業五

109354003 統碩一 吳書恆

109354027 統碩一 蔡海蓮

6/9/2021

1. Evaluate the CDF of Cauchy distribution F(x) = P(C < x) using the method of Important Sampling and other Variance Reduction Methods (at least two different methods). Consider x = –6, –5, –4, –3.5, –3, –2.5, –2.

這部分主要參考老師的講義舉一反三，這邊選用三個變異數縮減的方式，分別是 Antithetic Variate、Importance Sampling 和 Control Variate，其中 Control Variate 用 X^2 和 $X^4, X \sim Unif$ 來做爲控制參數。執行結果如下

要估計的值	Antithetic	Importance	Control
$P(C < -6)$	2.058371e-05	2.703322e-04	0.0744206248
$P(C < -5)$	2.628068e-05	2.099240e-04	0.0426754969
$P(C < -4)$	3.196501e-05	1.413918e-04	0.0195702518
$P(C < -3.5)$	3.466924e-05	1.120546e-04	0.0115296528
$P(C < -3)$	3.878265e-05	7.443965e-05	0.0058353711
$P(C < 2.5)$	4.863723e-05	5.175909e-05	0.0023561720
$P(C < -2)$	5.210675e-05	3.117060e-05	0.0006524311

2. Let X_i, $i = 1, \dots, 5$ be random variables, following the exponential distribution with mean 1. Consider the quantity θ defined by $\theta = P\{\sum_{i=1}^{5} iX_i \geq 21.6\}$. Propose at least three simulation methods to estimate θ and compare their variances.

這邊使用三個方式來估計 θ，分用 R 中的函數 `rexp()`、$F(x) \sim Unif$ 與 Antithetic Variate。

結果如下。

要估計的值	`rexp()`	$F(x) \sim Unif$	Antithetic
$\theta = P(\sum iX_i \geq 21.6)$	0.162	0.1715	0.16425
$Var(\theta)$	0.1358239	0.1421588	2.1125e-05

3. Let (X, Y) be a bivariate random variable following normal distribution, with mean (μ_x, μ_y) and variance matrix $\begin{pmatrix} \sigma_x^2 & \rho\sigma_x\sigma_y \\ \rho\sigma_x\sigma_y & \sigma_y^2 \end{pmatrix}$. Using Monte Carlo simulation to estimate the probability of $P(X+Y<k)$, where $k = 0, 1, 2, 3, 4$, $\mu_x = \mu_y = 0 = \rho$, and $\sigma_x = \sigma_y = 1$. In addition, propose at least three variance methods and compare their results with those using Monte Carlo simulation. Also, redo the preceding simulation if the correlation coefficient is –0.9, –0.5, 0.3, or 0.7.

由於 $X+Y \sim N(0, 2+\rho)$，可直接跳過從二維常態生成 (X, Y)。這邊使用三個變異數縮減的方式，分別是 Antithetic Variate、Stratified Sampling 和 Control Variate，其中 Stratified Sampling 是透過 `truncnorm` 套件完成分段常態的抽樣，Control Variate 只用 $X^2, X \sim Unif$ 來做爲控制參數，相關指令可參考附錄的講義，這邊列出幾個結果出來。

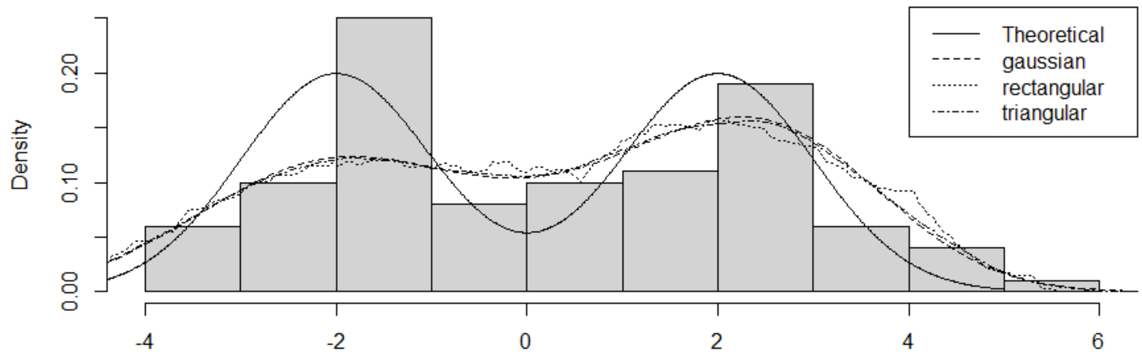
要估計的值	Antithetic	Stratified	Control
$P(X+Y<0), rho=-0.9$	0.002555378	0.000486196	0.000000000
$P(X+Y<1), rho=-0.5$	0.001666016	7.002643e-05	1.014878e-05
$P(X+Y<2), rho=0$	0.000746000	0.000460788	0.014491800
$P(X+Y<3), rho=0.3$	0.000238306	0.000211595	0.227917300
$P(X+Y<4), rho=0.7$	7.402811e-05	7.541101e-05	1.000227100

4. First, simulate 100 observations from a mixed distribution of N(–2,1) and N(2,1), each with probability 0.5. Then, use at least 3 density estimating methods to smooth the observations. You need to specify the parameters in the smoothing methods, and compare the results.

先透過套件 `LaplacesDemon` 套件生成混合常態，指令如下：

```
library(LaplacesDemon)
rn <- rnormm(100, p = c(0.5, 0.5), mu = c(-2, 2), sigma = c(1, 1))
```

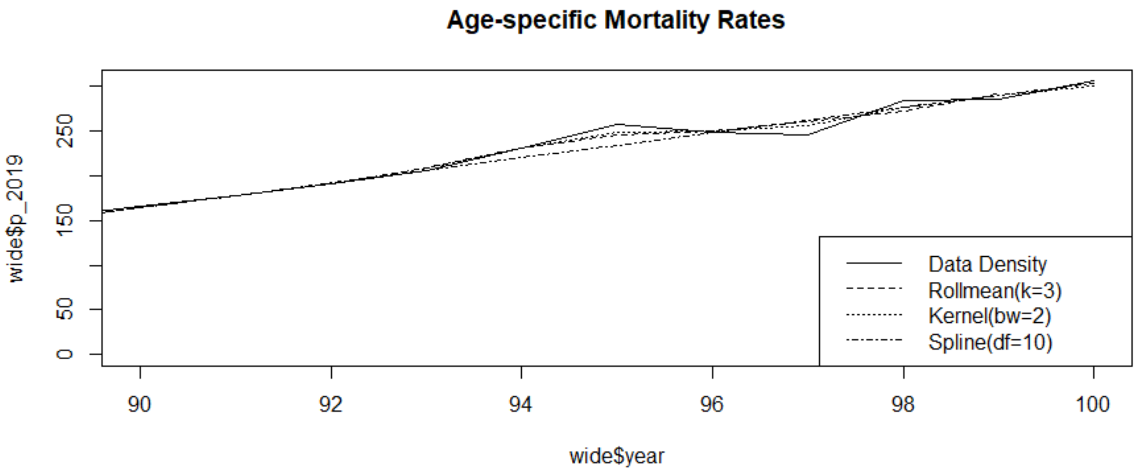
這邊用三個方式來平滑這個亂數的分布，分別是 gaussian、rectangular 和 triangular，結果如下。



發現 rectangular 較其他兩個方式震盪許多，這邊都設 bandwidth 的長度爲 1 以方便比較。

5. Visit the webpage of Department of Statistics, Ministry of Interior of the Taiwan Government and download the age-specific death records of year 2019. Use the smoothing techniques introduced in class to revise the age-specific mortality rates and compare with the values from 2019 Taiwan abridged life table. (Note: You only need to do the smoothing for the male/female, depending on your gender.)

此體由內政部網站找到資料並擷取下來，由於在 90 歲以下的平滑結果與時機資料幾乎重疊，以下只討論 90 歲以上的平滑狀況，這邊使用了三個方式 Running Mean、Kernel 和 Spline 來平滑，結果如下。



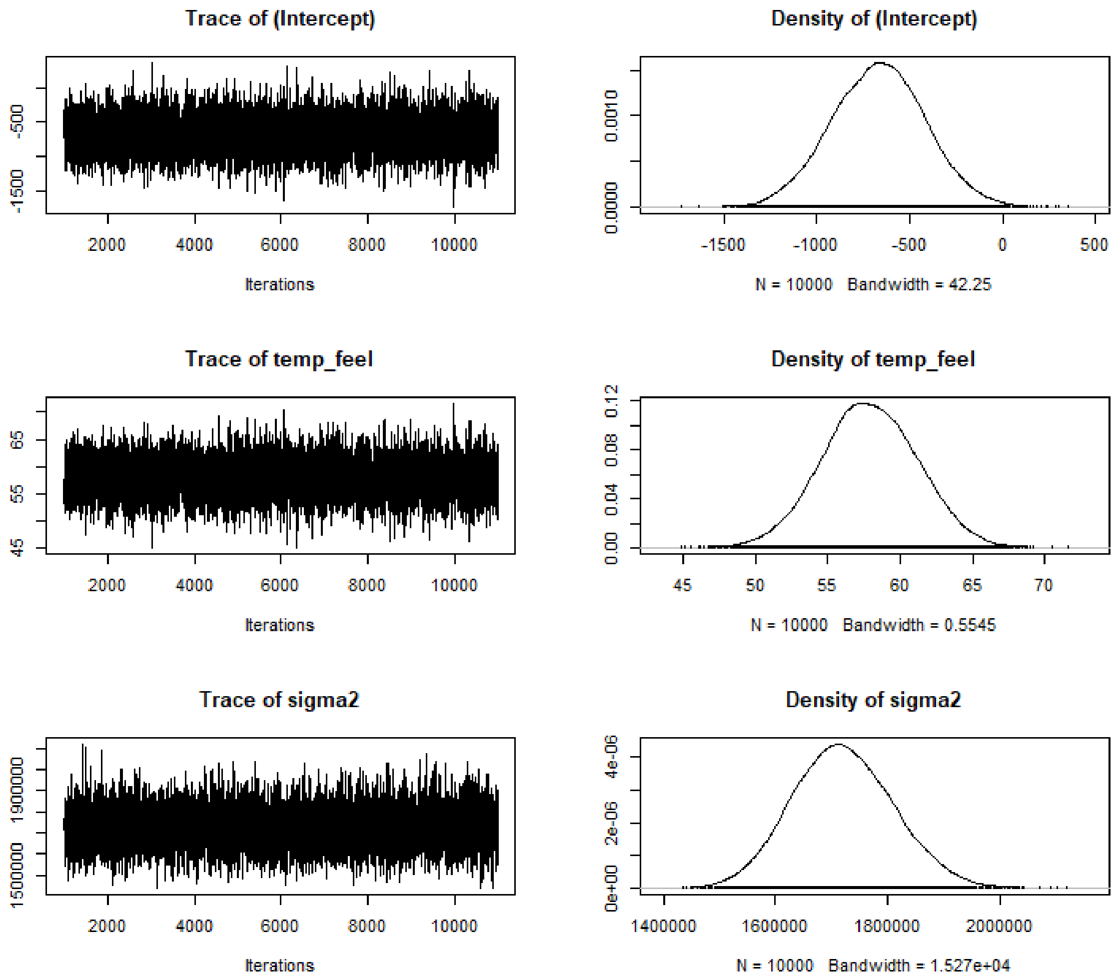
發現這幾個方式在調整特定參數後都表現的差不多。

6. Use “MCMCregress” in the module MCMCpack to obtain MCMC estimation of regression analysis. Duplicate the analysis in the lecture notes and apply the MCMC on the “bikes.csv” data. Compare your results with the regular simple linear regression.

這邊變數的選擇就如同老師上課的範例，Y 爲登記比散的人數（riders_registered），X 爲體感溫度（temp_feel），以下是一般迴歸分析與 MCMC 迴歸分析的結果比較。

方式	截距項（標準誤）	斜率項（標準誤）	殘差（標準誤）
`lm()`	-667.916 (251.608)	57.892 (3.306)	(1310)
`MCMCregress()`	-667.55 (251.464)	57.89 (3.316)	1720954 (90917)

發現在估計截距項（標準誤）和斜率項（標準誤）都差不多，殘差的部分還有待商榷。以下是 MCMC 執行迭代運算的結果。


Trace of (Intercept)
Iterations
Density of (Intercept)
N = 10000 Bandwidth = 42.25
Trace of temp_feel
Iterations
Density of temp_feel
N = 10000 Bandwidth = 0.5545
Trace of sigma2
Iterations
Density of sigma2
N = 10000 Bandwidth = 1.527e+04

Code:

```
#HW5
#1################################
#Impartance_sampling##########
importance_sampling <- function(k){
  t1 <- NULL
  for (i in 1:1000) {
    x <- runif(1000,0,k)
    y <- k/(pi*(1+x^2))
    z <- 0.5-y
    a <- mean(z)
    t1 <- c(t1,a)
  }
  return(cbind(mean(t1),var(t1)))
}

#conrtol variate###########
control <- function(k){
  t1 <- NULL
  for(i in 1:1000){
    x <- runif(1000,0,k)
    y <- x^2
    z <- x^4
    fx <- k/(pi*(1+x^2))
    g <- lm(fx~y+z)
    a1 <- g$coefficients[2]*(y-mean(y))
    a2 <- g$coefficients[3]*(z-mean(z))
    a <- 0.5 - fx + a1 + a2
    t1 <- c(t1, a)
  }
  return(cbind(mean(t1), var(t1)))
}

#antithetic############
antithetic <- function(k){
  t1 <- NULL
  for (i in 1:1000) {
    a <- runif(1000)
    b <- 1-a
    x <- qcauchy(a)
    x2 <- qcauchy(b)
    y <- sum(x>k)/1000
    y2 <- sum(x2>k)/1000
    t1 <- c(t1,(y+y2)/2)
  }
  return(cbind(mean(t1),var(t1)))
}

#result#####
t2 <- NULL
t3 <- NULL
t4 <- NULL
for(i in 1:7){
```

```
  b <- c(6, 5, 4, 3.5, 3, 2.5, 2)
  k2 <- importance_sampling(b[i])
  k3 <- control(b[i])
  k4 <- antithetic(b[i])
  t2 <- rbind(t2,k2)
  t3 <- rbind(t3,k3)
  t4 <- rbind(t4,k4)
}
t2 #importance_sampling result
t3 #control result
t4 #antithetic result


#2#################################
x <- matrix(rexp(10000), byrow = T, ncol = 5)
y <- x[,1]+2*x[,2]+3*x[,3]+4*x[,4]+5*x[,5]
sum(y >= 21.6)/2000
a <- y >= 21.6
var(a)

x1 <- matrix(runif(10000), byrow = T, ncol = 5)
y1 <- -log(x1)/1
z1 <- y1[,1]+2*y1[,2]+3*y1[,3]+4*y1[,4]+5*y1[,5]
sum(z1 >= 21.6)/2000
a1 <- z1 >= 21.6
var(a1)

a <- runif(2000*5)
b <- 1-a
x <- qexp(a)
x2 <- qexp(b)
y1 <- matrix(x, byrow = T, ncol = 5)
y2 <- matrix(x2, byrow = T, ncol = 5)
z1 <- sum(y1[,1]+2*y1[,2]+3*y1[,3]+4*y1[,4]+5*y1[,5]>=21.6)/2000
z2 <- sum(y2[,1]+2*y2[,2]+3*y2[,3]+4*y2[,4]+5*y2[,5]>=21.6)/2000
mean(c(z1, z2))
var(c(z1, z2))


#3#################################
library('truncnorm')

var.reduct <- function(k = 0, rho = 0, n1 = 1000, n2 = 100){
  z1 <- NULL
  z2 <- NULL
  z3 <- NULL
  z4 <- NULL
  for (i in 1:n1){
    # Opposite
    rn1 <- rnorm(n2, 0, sqrt(2 + rho))
    z1 <- c(z1, sum(rn1 < k)/n2)
    z2 <- c(z2, sum(-rn1 < k)/n2)
    # Stratified
```

```
    rn2 <- NULL
    for (j in 1:5){
      rn2 <- c(rn2, rtruncnorm(0.2*n2, a = qnorm((j-1)/5, sd = sqrt(2 + 
rho)),
                               b = qnorm(j/5, sd = sqrt(2 + rho)),
                               mean = 0,  sd = sqrt(2 + rho)))
    }
    z3 <- c(z3, sum(rn2 < k)/n2)
    # Control
    index <- runif(n2, 0, k)
    index2 <- index^2
    rn3 <- k*dnorm(index, 0, sqrt(2 + rho))
    b <- cov(rn3, index)/var(index)
    b[is.na(b)] <- 0
    z4 <- c(z4, 0.5 + rn3 - b*(index2-mean(index2)))
  }
  return(list("Antithetic mean and variance" = c(mean(c(z1, z2)), 
var(c(z1, z2))),
          "Stratified mean and variance" = c(mean(z3), var(z3)),
          "Control mean and variance" = c(mean(z4), var(z4)),
          "Theoretical mean" = pnorm(k, 0, sqrt(2 + rho))))
}

var.reduct(k = 0, rho = -0.9)
var.reduct(k = 1, rho = -0.5)
var.reduct(k = 2, rho = 0)
var.reduct(k = 3, rho = 0.3)
var.reduct(k = 4, rho = 0.7)


#4#################################
library(LaplacesDemon)
rn <- rnormm(100, p = c(0.5, 0.5), mu = c(-2, 2), sigma = c(1, 1))
hist(rn, freq = FALSE, main = "Hist of Mixture Normal")
x <- seq(-5, 5, 0.01)
lines(x, dnormm(x, p = c(0.5, 0.5), mu = c(-2, 2), sigma = c(1, 1)))
lines(density(rn2, kernel = "gaussian", bw = 1), lty = 2)
lines(density(rn2, kernel = "rectangular", bw = 1), lty = 3)
lines(density(rn2, kernel = "triangular", bw = 1), lty = 4)
legend('topright', legend = c("Theoretical", "gaussian", 
"rectangular",
                      "triangular"), lty = 1:4)


#5#################################
library(zoo)
wide <- read.csv("StatisticalSimulation/maledeathrates.csv", header = 
T)
plot(wide$year, wide$p_2019, type = 'l', xlim = c(90, 100),
    main = "Age-specific Mortality Rates")
lines(wide$year[2:100], rollmean(wide$p_2019, k=3), lty = 2)
lines(ksmooth(wide$year, wide$p_2019, "normal", bandwidth = 2), lty = 
3)
```

```
lines(smooth.spline(wide$year, wide$p_2019, df = 10), lty = 4)
legend('bottomright', legend = c("Data Density", "Rollmean(k=3)",
"Kernel(bw=2)",
                      "Spline(df=10)"), lty = 1:4)


#6#################################
library(MCMCpack)
bikes <- read.csv("StatisticalSimulation/bikes.csv", h = T)
summary(lm(riders_registered~temp_feel, data = bikes))
posterior <- MCMCregress(riders_registered~temp_feel, data = bikes)
summary(posterior)
windows()
plot(posterior)
```