UiO : Matematisk institutt
Det matematisk-naturvitenskapelige fakultet

STK-4051/9051 Computational Statistics Spring 2024 Ex7

Instructor: Odd Kolbjørnsen, oddkol@math.uio.no

#20

Exercise 20 (Testing rejection sampling)
The example in this exercise is (deceptively) simple. Nobody in their right mind would use rejection sampling to solve it. Yet it is sufficient to provide insight into a major issue with rejection algorithms. Besides it is always convenient to know answers theoretically when testing out a general method for use in problems that do not allow simple solutions.

Suppose you originally know that a random variable X of interest is normal $(0, 1)$. Further information is available through a measurement or an observation z, which we assume is the outcome of a random variable Z with conditional density $f(z|x)$ given $X = x$. We will further assume that $f(z|x) < A$ for all x, z and some constant $A < \infty$.

(a). Show that

$$p(x|z) = Cf(z|x)\exp(-\frac{1}{2}x^2)$$

is the posterior density of X given z (C is a normalization constant).

Suppose we want to sample from this posterior distribution.

$$p(x|z) = \frac{p(x)p(z|x)}{p(z)} \propto \frac{1}{\sqrt{2\pi}}\exp(-\tfrac{1}{2}x^2)f(z|x) \propto \exp(-\tfrac{1}{2}x^2)f(z|x)$$

(*b*). Write down an algorithm based on rejection from the normal $(0,1)$ (it is assumed that $f(z|x)$ is a computable function).

(*b*). We need that there exist an $\tilde{\alpha}$ such that

$$\frac{\exp(-\frac{1}{2}x^2)f(z|x)}{\frac{1}{\sqrt{2\pi}}\exp(-\frac{1}{2}x^2)} \leq M$$

or

$$\sqrt{2\pi}f(z|x) \leq M$$

which is ok due to the constraint on $f(z|x)$. In this case we can sample from $N(0,1)$ an accept with probability $\sqrt{2\pi}f(z|x)/M$

Let N be the number of attempts until acceptance.

(c). Calculate $E(N)$ when the conditional distribution of Z given $X = x$ is normal (ax, σ^2). Discuss the behavior of $E(N)$ as a function of z.

(c). For $Z|X = x \sim N(ax, \sigma^2)$, we have

$$\sqrt{2\pi} f(z|x) = \frac{1}{\sigma} \exp(-\tfrac{1}{2\sigma^2}(z - ax)^2) \leq \tfrac{1}{\sigma}$$

so given that we have simulated a value x, we will accept with probability $\exp(-\frac{1}{2\sigma^2}(z - ax)^2)$. Now the expected probability of accepting is:

$$E^X[\exp(-\tfrac{1}{2\sigma^2}(z - aX)^2)]$$

$$
\begin{aligned}
&E^X[\exp(-\tfrac{1}{2\sigma^2}(z-aX)^2)] \\
&= \int \exp(-\tfrac{1}{2\sigma^2}(z-ax)^2 \frac{1}{\sqrt{2\pi}} \exp(-\tfrac{1}{2}x^2)dx \\
&= \frac{1}{\sqrt{2\pi}} \int \exp(-\tfrac{1}{2}[\frac{z^2}{\sigma^2} - 2\frac{az}{\sigma^2}x + \tfrac{a^2}{\sigma^2}x^2 + x^2])dx \\
&= \frac{1}{\sqrt{2\pi}} \exp(-\tfrac{z^2}{2\sigma^2}) \int \exp(-\tfrac{1}{2}[\tfrac{a^2+\sigma^2}{\sigma^2}x^2 - 2\tfrac{az}{\sigma^2}x])dx \\
&= \frac{1}{\sqrt{2\pi}} \exp(-\tfrac{z^2}{2\sigma^2}) \int \exp(-\tfrac{a^2+\sigma^2}{2\sigma^2}[x^2 - 2\tfrac{az}{a^2+\sigma^2}x])dx \\
&= \exp(-\tfrac{z^2}{2\sigma^2}) \exp(\tfrac{a^2+\sigma^2}{2\sigma^2}(\tfrac{az}{a^2+\sigma^2})^2) \frac{1}{\sqrt{2\pi}} \int \exp(-\tfrac{a^2+\sigma^2}{2\sigma^2}[(x-\tfrac{az}{a^2+\sigma^2})^2])dx \\
&= \exp(-\tfrac{1}{2\sigma^2}z^2) \exp(\tfrac{1}{2\sigma^2}(\tfrac{a^2z^2}{a^2+\sigma^2})) \tfrac{\sigma}{\sqrt{a^2+\sigma^2}} \\
&= \exp(-\tfrac{z^2}{2(a^2+\sigma^2)}) \tfrac{\sigma}{\sqrt{a^2+\sigma^2}}
\end{aligned}
$$

so it will decrease with z deviating from zero, which is reasonable since we use a proposal distribution that is centered around zero. On average the number of samples needed to get 1 sample is given as is $1/P(accept)$. If we want n samples, the number of samples we need to porpose is :

$$
n \cdot \exp(\tfrac{z^2}{2(a^2+\sigma^2)}) \tfrac{\sqrt{a^2+\sigma^2}}{\sigma}
$$

(*d*). Write a program which carries out the rejection sampling. Keep track on the number of trials before acceptance. ($f(x|z)$ may be a general function, depending on x, or the normal one introduced above).

(*e*). Run the program a suitable number (m) times to generate m simulations of X from the distribution $p(x|z)$. Use $a = 0.5$ and $\sigma^2 = 0.1$. Let $z \geq 0$. Start at 0 and increase it in steps of 0.5. (Use at least 100 repetitions).

(*f*). Compute the mean of N for the simulations. Compare with the theoretical values.

```
#Extra 20
n = 100

a=0.5
sig2=0.1
sig=sqrt(sig2)
z = seq(0,2,by=0.5)
mu.sim = matrix(nrow=length(z),ncol=100)
sig2.sim = matrix(nrow=length(z),ncol=100)
N.sim = matrix(nrow=length(z),ncol=100)
```

```
x.sim=rep(NA,n)
for(i in 1:length(z))
{
  show(z[i])
for(j in 1:100)
{
  N = 0
  for(k in 1:n)
  {
    more = TRUE
    while(more)
    {
      N = N+1
      x = rnorm(1)
      p = exp(-0.5*(z[i]-a*x)^2/sig2)
      if(runif(1)<p)
        more = FALSE
    }
    x.sim[k] = x
  }
  mu.sim[i,j] = mean(x.sim)
  sig2.sim[i,j] = var(x.sim)
  N.sim[i,j] = N
}
}
```

(g). Compute mean and variance of the simulations of X. Again compare with the theoretical values. (You must calculate $E(X|z)$ and $\mathrm{var}(X|z)$ theoretically. Use theory for conditional Gaussian distributions.)

X of interest is normal $(0,1)$.
$Z|X=x \sim N(ax, \sigma^2)$,

$$\begin{bmatrix} Z \\ X \end{bmatrix} \sim N\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} a^2+\sigma^2 & a \\ a & 1 \end{bmatrix} \right)$$

$$E(Z) = E\big(E(Z|X)\big) = E(aX) = aE(X) = 0$$

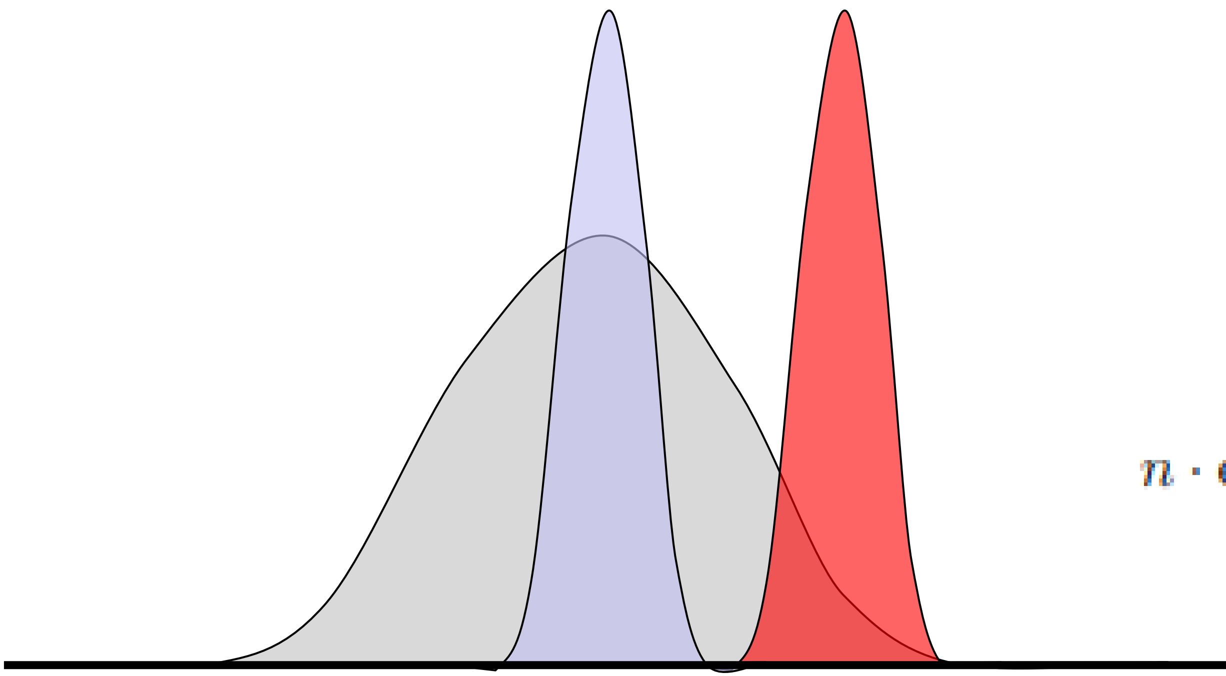
$$\begin{aligned} \mathrm{Cov}(Z,X) &= E(ZX) = E\big(E(ZX|X)\big) \\ &= E(XE(Z|X)) = E(XaX) = aE(X^2) = a \cdot 1 \end{aligned}$$

$$\begin{aligned} \mathrm{Var}(Z) &= \mathrm{Var}\{E(Z|X)\} + E(\mathrm{Var}(Z|X)) \\ &= \mathrm{Var}\{aX\} + E(\sigma^2) \\ &= a^2 + \sigma^2 \end{aligned}$$

$$E(X|Z) = \mu_X + \Sigma_{XZ}\Sigma_Z^{-1}(z-\mu_Z) = 0 + \frac{a}{a^2+\sigma^2}(z-0) = \frac{az}{a^2+\sigma^2}$$

$$\mathrm{Var}(X|Z) = \Sigma_X - \Sigma_{XZ}\Sigma_Z^{-1}\Sigma_{ZX} = 1 - \frac{a \cdot a}{a^2+\sigma^2} = \frac{\sigma^2}{a^2+\sigma^2}$$

(h). Try to make some general conclusions from this exercise. If your algorithm took a long time to converge in some of the examples, how could it have been improved?



$$n \cdot \exp\left(\frac{z^2}{2(a^2+\sigma^2)}\right)\frac{\sqrt{a^2+\sigma^2}}{\sigma}$$

#21

Exercise 21 (The impact of serial correlation in regression)
It is in economics and in many industrial applications of statistics quite common that error terms are serially correlated (also called auto-correlated). We shall in this exercise examine the effect of this on a simple regression method where such serial correlations are ignored. Consider the regression model

$$Y_t = \alpha + \beta x_t + \varepsilon_t, \quad t = 1, ..., n,$$

where α and β are the regression coefficients and ε_t the error terms. As basis for the study take the standard least-squares estimate of β, i.e.

$$\hat{\beta} = \sum_{i=1}^{n}(x_i - \overline{x})y_i / \sum_{i=1}^{n}(x_i - \overline{x})^2.$$

As model for ε_t take the AR(1) model

$$\varepsilon_t = a\varepsilon_{t-1} + \eta_t$$

where $\eta_1, ..., \eta_n$ is an i.i.d. sequence with mean 0 and variance σ_η^2. It can then be proved that

$$\text{var}(\varepsilon_t) = \frac{\sigma_\eta^2}{1 - a^2}$$

and that

$$\text{corr}(\varepsilon_t, \varepsilon_{t-s}) = a^{|t-s|},$$

assuming the model to be in stationary state. Although it is not hard to analyze the impact on the serial correlation analytically, we shall in the following run simulations. Let $n = 20$.

(a). Start by drawing $x_1, ..., x_{20}$ from the uniform distribution over $[-1, 1]$.

Let $x_1, ..., x_{20}$ be fixed throughout and take $\alpha = 0$ and $\beta = 2$ for the true model.

(b). Compute 100 copies of the error terms by sampling $\varepsilon_1, ..., \varepsilon_n$. Use $a = -0.9, -0.5, 0, 0.5, 0.8$ and 0.9 and employ the trick of common random numbers (which means that you use the same η-sequence). Arrange things so that $\sigma_\varepsilon = 1$ is the same value for all choices of a.

$$\text{var}(\varepsilon_t) = \frac{\sigma_\eta^2}{1 - a^2}$$

Hint: In order to generate the same sequence of random numbers several times, you can use the same seed number for each generation. The seed can be set to a given number N by the command

```
set.seed(N)
```

The command

```
eps <- arima.sim(20,model=list(ar=a),sd=sigma.eta)
```

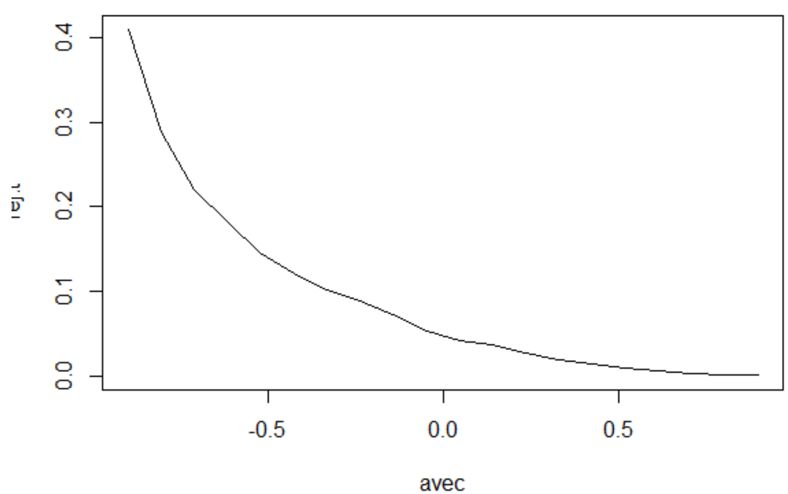
simulates an AR(1) time-series of length 20 with $a =$ `a` and $\sigma_\eta =$ `sigma.eta`.

```r
for(k in 1:length(avec))
{
  a = avec[k]
  sigma.eta = sigma*sqrt(1-a^2)
  set.seed(345)
  for(m in 1:M)
   {
    eps <- arima.sim(n,model=list(ar=a),sd=sigma.eta)
    y <- alpha+beta*x+eps
    fit <- lm(y~x)
    beta.est[m,] <- fit$coef
    sigma.est[m] <- summary(fit)$sigma
    tstat2[m] = (coef(summary(fit))[2,1]-beta)/coef(summary(fit))[2,2]
    rej[m] = abs(tstat2[m])>qt(0.975,n-2)
   }

  cat(k,": Estimated significance level",mean(rej),"\n")

  bias.beta[k,] = apply(beta.est,2,mean)-c(alpha,beta)
  var.beta[k,] =  apply(beta.est,2,var)
  bias.sig[k] = mean(sigma.est)-sigma
  var.sig[k] = var(sigma.est)
  rej.t[k] = mean(rej)
}
```

Learning:
you can investigate complex models by simulation



#22

Exercise 22 (Importance sampling)
Let $\phi(x) = (2\pi)^{-1/2}\exp(-x^2/2)$ be the standard normal density. Consider the integral

$$J = \int_{-\infty}^{\infty} (x+a)^2 \phi(x)\, dx = 1 + a^2.$$

(a). Compute J by Monte Carlo sampling from the standard normal. Use 100 and 1000 simulations and let $a = 0, 1, 2, 3, 4$.

(b). Calculate the theoretical standard deviation for the estimates and compare with the error of the actual values computed in (a). Note the dependency on a.

(c). Try importance sampling based om $g(x) = \phi(x - a)$. Redo (a) and (b).

(d). Formulate general conclusions. Note that g depends on a.

Solution to exercise 22. (a). We have

$$J = E[(X+a)^2] = E[X^2 + 2aX + a^2] = 1 + a^2$$

(b). We have with M simulations

$$\begin{aligned} M\text{Var}[\widehat{J}] =& E[(X+a)^4] - [E[(X+a)^2]]^2 \\ =& E[X^4 + 4aX^3 + 6a^2X^2 + 4a^3X + a^4] - [E[X^2 + 2aX + a^2]]^2 \\ =& 3 + 0 + 6a^2 + 0 + a^4 - [1 + 0 + a^2]^2 \\ =& 3 + 6a^2 + a^4 - 1 - 2a^2 - a^4 = 2 + 4a^2 \end{aligned}$$

Let $\phi(x) = (2\pi)^{-1/2}\exp(-x^2/2)$ be the standard normal density. Consider the integral

$$J = \int_{-\infty}^{\infty} (x+a)^2 \phi(x)\, dx = 1 + a^2.$$

(a). Compute J by Monte Carlo sampling from the standard normal. Use 100 and 1000 simulations and let $a = 0, 1, 2, 3, 4$.

(b). Calculate the theoretical standard deviation for the estimates and compare with the error of the actual values computed in (a). Note the dependency on a.

(c). Try importance sampling based om $g(x) = \phi(x-a)$. Redo (a) and (b).

(d). Formulate general conclusions. Note that g depends on a.

(c). We have

$$J = \int_{-\infty}^{\infty} (x+a)^2 \phi(x)\, dx = \qquad \int_{-\infty}^{\infty} (x+a)^2 \frac{\phi(x)}{\phi(x-a)} \phi(x-a)\, dx$$

An alternative would be to utilize that

$$\begin{aligned} J &= \int_{-\infty}^{\infty} (x+a)^2 \phi(x)\, dx \\ &= \int_{-\infty}^{\infty} y^2 \phi(y-a)\, dy \\ &\approx \frac{1}{M} \sum_{i=1}^{M} y_i^2 \end{aligned}$$

where now $y_i \sim N(a, 1)$.

#6.3

6.3. Consider finding $\sigma^2 = E\{X^2\}$ when X has the density that is proportional to $q(x) = \exp\{-|x|^3/3\}$.

a. Estimate σ^2 using importance sampling with standardized weights.

b. Repeat the estimation using rejection sampling.

c. Philippe and Robert describe an alternative to importance-weighted averaging that employs a Riemann sum strategy with random nodes [506, 507]. When draws $X_1, \ldots, X_n$ originate from f, an estimator of $E\{h(X)\}$ is

$$\sum_{i=1}^{n-1}(X_{[i+1]} - X_{[i]})h(X_{[i]})f(X_{[i]}), \tag{6.86}$$

where $X_{[1]} \leq \cdots \leq X_{[n]}$ is the ordered sample associated with $X_1, \ldots, X_n$. This estimator has faster convergence than the simple Monte Carlo estimator. When $f = cq$ and the normalization constant c is not known, then

$$\frac{\sum_{i=1}^{n-1}(X_{[i+1]} - X_{[i]})h(X_{[i]})q(X_{[i]})}{\sum_{i=1}^{n-1}(X_{[i+1]} - X_{[i]})q(X_{[i]})} \tag{6.87}$$

estimates $E\{h(X)\}$, noting that the denominator estimates $1/c$. Use this strategy to estimate σ^2, applying it post hoc to the output obtained in part (b).

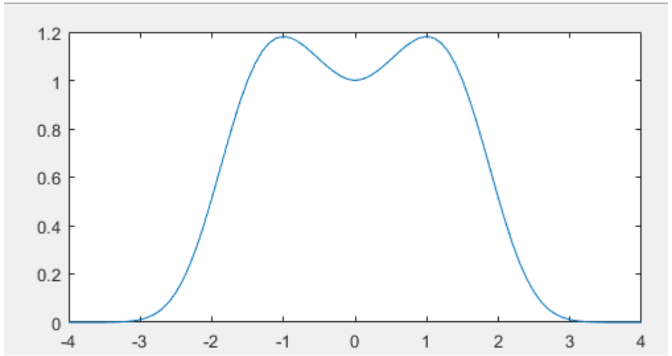
d. Carry out a replicated simulation experiment to compare the performance of the two estimators in parts (b) and (c). Discuss your results.

```
for(i in 1:M)
{

#a
x = rnorm(N)
w = q(x)/dnorm(x)
w = w/sum(w)
sig2.hat1[i] = sum(w*x^2)

#b
#The ratio q(x)/phi(x) has max point at x=-1,1
alpha = q(1)/dnorm(1)
w = q(x)/(alpha*dnorm(x))
x = x[runif(N)<w]
sig2.hat2[i] = mean(x^2)

#c
x.sort = sort(x)
x1 = x.sort[-length(x)]
xn = x.sort[-1]
sig2.hat3[i] = mean((xn-x1)*x1^2*q(x1))/mean((xn-x1)*q(x1))
}
```



	Estimate	Std.Error
Imp.samp	0.7771714	0.007834866
Rej.samp	0.7773261	0.010080669
P-R	0.7753223	0.001451753

Exercise 30 (Common random numbers)
Let $\Psi(\theta) = E_\theta(X)$, where X is a random variable with distribution depending on θ. In the situations we have in mind this distribution is complicated, but $\Psi(\theta)$ can be approximated by sampling, so that

$$\hat{\Psi}_n(\theta) = \frac{1}{N}\sum_{j=1}^{N} X_j^* \tag{1}$$

where $X_1^*, ..., X_n^*$ is an i.i.d. sample of X under θ.

(a). Write down the approximative distribution for the error $\hat{\Psi}_n(\theta) - \Psi(\theta)$ using $\sigma^2(\theta) = \text{var}_\theta(X)$.

Solution to exercise 30. (a). We have that

$$E[\hat{\Psi}_n(\theta)] = E_\theta[X_j^*] = \Psi(\theta)$$
$$\text{Var}[\hat{\Psi}_n(\theta)] = \frac{1}{N}\text{Var}[X_j^*] = \frac{1}{N}\sigma^2(\theta)$$

By the central limit theorem we then get

$$\hat{\Psi}_n(\theta) - \Psi(\theta) \approx N(0, \tfrac{1}{N}\sigma^2(\theta))$$

Frequently much interest is directed towards differences $\Psi(\theta_2) - \Psi(\theta_1)$ for two θ-values $\theta_1 < \theta_2$. This will be so if the derivative (or gradient) is to be estimated as part of some Monte Carlo numerical scheme. Another common situation is when we want to study differences in performance of some statistical procedure.

(b). Write down the approximative distribution of $\hat{\Psi}_n(\theta_1) - \hat{\Psi}_n(\theta_2)$ when $\hat{\Psi}_n(\theta_1)$ and $\hat{\Psi}_n(\theta_2)$ are obtained from different rounds of simulation.

(b). We have

$$\hat{\Psi}_n(\theta_j) - \Psi(\theta_j) \approx N(0, \frac{1}{N}\sigma^2(\theta_j)), \quad j = 1, 2$$

and combined with independence we get

$$\hat{\Psi}_n(\theta_1) - \hat{\Psi}_n(\theta_2) \approx N(\Psi(\theta_1) - \Psi(\theta_2), \tfrac{1}{N}(\sigma^2(\theta_1) + \sigma^2(\theta_2)))$$

When we sample X, we use a routine that can be written as a function

$$X = h(\theta, Z) \tag{2}$$

where Z is a random vector, perhaps a long one, having distribution not dependent on θ. The function h is possible very complicated, and may not necessarily be available in analytical form. We assume here that it possesses the necessary smoothness.

Inserting (2) into (1) yields

$$\hat{\Psi}_n(\theta) = \frac{1}{N}\sum_{i=1}^{N} h(\theta, Z_j^*), \tag{3}$$

where $Z_1^*, ..., Z_n^*$ is an i.i.d. sample from Z. Note that $\hat{\Psi}_n(\theta)$ in (3) for a given sequence $Z_1^*, ..., Z_n^*$ can be regarded as a deterministic function of θ. It is in practice possible to organize the computations so that the same sequence $Z_1, ..., Z_n^*$ goes into (3) for any value of θ by resetting the seed of the Monte Carlo generator.

(c). Determine the approximate distribution of $\hat{\Psi}_n(\theta_2) - \hat{\Psi}_n(\theta_1)$ when common random numbers are used.

Hint: Introduce $\rho(\theta_1, \theta_2) = \text{corr}\{h(\theta_2, Z^*), h(\theta_1, Z^*)\}$ and comment on how the magnitude of ρ influences the result.

(*c*). Determine the approximate distribution of $\widehat{\Psi}_n(\theta_2) - \widehat{\Psi}_n(\theta_1)$ when common random numbers are used.

Hint: Introduce $\rho(\theta_1, \theta_2) = \text{corr}\{h(\theta_2, Z^*), h(\theta_1, Z^*)\}$ and comment on how the magnitude of ρ influences the result.

$$\hat{\Psi}_n(\theta_1) - \hat{\Psi}_n(\theta_2) = \frac{1}{N}\sum_{j=1}^{N}[h(\theta_1, Z_j^*) - h(\theta_2, Z_j^*)]$$

which has the same expectation as before but variance

$$\begin{aligned}
&\text{Var}[\hat{\Psi}_n(\theta_1) - \hat{\Psi}_n(\theta_2)] \\
&\quad = \tfrac{1}{N}\text{Var}[h(\theta_1, Z_j^*) - h(\theta_2, Z_j^*)] \\
&\quad = \tfrac{1}{N}[\text{Var}[h(\theta_1, Z_j^*)] + \text{Var}[h(\theta_2, Z_j^*)] - 2\text{Cov}[h(\theta_1, Z_j^*), h(\theta_2, Z_j^*)]] \\
&\quad = \tfrac{1}{N}[\sigma^2(\theta_1) + \sigma^2(\theta_2) - 2\sigma(\theta_1)\sigma(\theta_2)\rho(\theta_1, \theta_2)] \\
&\quad = \tfrac{1}{N}(\sigma^2(\theta_1) + \sigma^2(\theta_2))[1 - \tfrac{2\sigma(\theta_1)\sigma(\theta_2)}{\sigma^2(\theta_1)+\sigma^2(\theta_2)}\rho(\theta_1, \theta_2)] \equiv \tfrac{1}{N}\tau^2(\theta_1, \theta_2)
\end{aligned}$$

Suppose that the difference $\theta_2 - \theta_1$ is fairly small so that

$$h(\theta_2, Z) \simeq h(\theta_1, Z) + \frac{\partial h(\theta_1, Z)}{\partial \theta}(\theta_2 - \theta_1).$$

(d). Use this approximation to find an alternative expression for the approximate distribution of $\widehat{\Psi}_n(\theta_2) - \hat{\Psi}_n(\theta_1)$.

(d). We then have

$$\hat{\Psi}_n(\theta_1) - \hat{\Psi}_n(\theta_2) \approx \frac{1}{N}\sum_{j=1}^{N} \frac{\partial h(\theta_1, Z_j^*)}{\partial \theta}(\theta_2 - \theta_1)$$

and we obtain the approximate distribution

$$\hat{\Psi}_n(\theta_1) - \hat{\Psi}_n(\theta_2) \approx N\left((\theta_2 - \theta_1)E\left[\frac{\partial h(\theta_1, Z_j^*)}{\partial \theta}\right], \tfrac{1}{N}\tau^2(\theta_1, \theta_2)\right)$$

Note that if $h(\cdot)$ is sufficiently smooth and both h and its derivative is integrable, we have

$$\begin{aligned} E\left[\frac{\partial h(\theta, Z_j^*)}{\partial \theta}\right] &= \int_z \frac{\partial h(\theta, Z_j^*)}{\partial \theta} f(z)dz \\ &= \frac{\partial}{\partial \theta}\int_z h(\theta, Z_j^*) f(z)dz = \frac{\partial}{\partial \theta}\Psi(\theta) \end{aligned}$$

(e). Compare the result on (d) with that in (b). Comment on the relevance for Monte Carlo based estimation of the derivative $\Psi'(\theta)$.

$$\frac{1}{N}(\sigma^2(\theta_1) + \sigma^2(\theta_2))$$

$$\frac{1}{N}(\sigma^2(\theta_1) + \sigma^2(\theta_2))[1 - \frac{2\sigma(\theta_1)\sigma(\theta_2)}{\sigma^2(\theta_1)+\sigma^2(\theta_2)}\rho(\theta_1, \theta_2)] \equiv \frac{1}{N}\tau^2(\theta_1, \theta_2)$$

(e). We have for $\theta_1 \approx \theta_2$ that

$$\frac{\partial\Psi(\theta_1)}{\partial\theta} \approx \frac{\Psi(\theta_1) - \Psi(\theta_2)}{\theta_1 - \theta_2}$$

which we can approximate by

$$\frac{\hat{\Psi}_n(\theta_1) - \hat{\Psi}_n(\theta_2)}{\theta_1 - \theta_2} \approx N\left(\Psi'(\theta), \frac{1}{N}\frac{\tau^2(\theta_1,\theta_2)}{(\theta_1-\theta_2)^2}\right)$$

Given that the denominator in the variance can be small, it is important to also make the nominator small.

Example #30

```
#Exercise 30
#Phi(theta)=E[(X-theta)^2], log(X)\sim N(theta,1)
theta1=1.1
theta2=1
```

```
#Independent draws
Z1 = rnorm(N,theta1,1);X1=exp(Z1)
Z2 = rnorm(N,theta2,1);X2=exp(Z2)
Phi1.hat = mean((X1-theta1)^2)
sig2.1.hat = sd((X1-theta1)^2)
Phi2.hat = mean((X2-theta2)^2)
sig2.2.hat = sd((X2-theta2)^2)
sig2.hat = sd(((X1-theta1)^2+(X2-theta2)^2))
```

```
#Common random numbers
Eps = rnorm(N)
Z1 = theta1+Eps;X1=exp(Z1)
Z2 = theta2+Eps;X2=exp(Z2)
Phi1.hat = mean((X1-theta1)^2)
Phi2.hat = mean((X2-theta2)^2)
sig2.hat = sd((X1-theta1)^2-(X2-theta2)^2)
```

```
[1] "Difference - independent draws"
[1] 9.256140 2.696670 2.696251
[1] "Derivative - independent draws"
[1] 92.56140 26.96670 26.96251
[1] "Difference - common random numbers"
[1] 10.604473  0.304442
[1] "Derivative - common random numbers"
[1] 106.04473   3.04442
Difference obtained by 100 000 000 simulations: 10.3686
Derivative obtained by 100 000 000 simulations: 103.686
```

Ex 30

- Run the test example.
- The use of correlated samples gives a 10 times improvement of the standard deviation
- This corresponds to the commonly known difference between a paired test and a two sample test.