Estimating the expected value with non-iid data

For iid random variables $X_1, \ldots, X_n$, with

$$EX_i = \mu \quad \text{var}(X_i) = \sigma^2$$

the sample mean satisfies

$$E\bar{X} = \mu$$

and

$$\text{var}\bar{X} = \sigma^2/n.$$
Unequal variances

Suppose the $X_i$ have a common mean $\mu$, but have different variances

$$\text{var} X_i = \sigma_i^2.$$  

It is a fact that

$$E \bar{X} = \mu$$

and

$$\text{var} \bar{X} = \sum_i \sigma_i^2/n^2.$$
If we write

$$\bar{\sigma}^2 = \sum_i \sigma_i^2 / n$$

to denote the average variance, then

$$\text{var} \bar{X} = \frac{\bar{\sigma}^2}{n}.$$
Generating data with unequal variances

Since \( \text{var}(c \cdot X) = c^2 \text{var}(X) \), we can generate a dataset with unequal variances by starting with an iid set

\[ X_1, \ldots, X_n \]

specifying a set of constants

\[ c_1, \ldots, c_n, \]

(not all equal), and then taking as our data

\[ c_1 X_1, \ldots, c_n X_n. \]

The variances we get are

\[ c_1^2 \sigma^2, \ldots, c_n^2 \sigma^2. \]

Note that as we do replications, the same \( c_i \) values should be used throughout.
This simulation shows illustrates that \( \text{var}(\bar{X}) = \sigma^2/n \).

```r
## The number of simulation replications.
nrep = 1e4

## Sample sizes.
SS = seq(5,30,5)

## Storage for the results.
R = array(0, c(length(SS),2))

## Consider sample means for different sample sizes.
for (k in 1:length(SS))
{
  p = SS[k] ## The sample size for this iteration.

  ## Generate p different variances, store as an array.
  V = rexp(p)
  W = matrix(V, nrow=nrep, ncol=p, byrow=TRUE)

  ## Generate a 1000 x p array of normal values, with row i having
```

5
## variance $V[i]$.  
X = array(rnorm(nrep*p), c(nrep,p))
X = sqrt(W) * X

## Get the sample mean of each row.  
M = apply(X, 1, mean)

R[k,1] = var(M)   ## The simulation estimate
R[k,2] = mean(V)/p   ## The theoretical value
}
Dependence

First we need a way to describe the dependence between two random variables. The population covariance between random variables $X$ and $Y$ is

$$\text{cov}(X,Y) = E(X - EX)(Y - EY) = EXY - EX \cdot EY.$$ 

Note that pairs $X,Y$ must be observed jointly for this to make sense.

A positive covariance means that when $X$ is greater than its mean, $Y$ tends to be greater than its mean as well. A negative covariance means that when $X$ is greater than its mean, $Y$ tends to be less than its mean.
Properties of the covariance

\[ \text{cov}(X, Y) = \text{cov}(Y, X) \]
\[ \text{cov}(X, Y) = 0 \text{ when } X \text{ and } Y \text{ are independent} \]
\[ \text{cov}(c \cdot X, Y) = c \cdot \text{cov}(X, Y) \]
\[ \text{cov}(c \cdot X, d \cdot Y) = c \cdot d \cdot \text{cov}(X, Y) \]
\[ \text{cov}(X + c, Y + d) = \text{cov}(X, Y) \]
\[ \text{cov}(X, Y + Z) = \text{cov}(X, Y) + \text{cov}(X, Z) \]
Estimating the covariance

Given a bivariate (paired) dataset \((X_1, Y_1), \ldots, (X_n, Y_n)\), the sample covariance is

\[
\hat{\text{cov}}(X, Y) = \frac{1}{n - 1} \sum_i (X_i - \bar{X})(Y_i - \bar{Y}).
\]
Generating dependent data

One way to generate non-independent data is using an autoregressive (AR) process. We will only consider a special case called AR(1). Choose a number $0 \leq \alpha < 1$ and a variance parameter $\tau^2 > 0$. Define

$$\sigma^2_X = \frac{\tau^2}{1 - \alpha^2}.$$

To generate the data, let $X_1$ be normal with expected value 0 and variance $\sigma^2_X$. Then generate the other $X_i$ values according to the rule

$$X_i = \alpha X_{i-1} + \epsilon_i,$$

where the $\epsilon_i$ are independent and normal with expected value zero and variance $\tau^2$.

It is a fact that each term of the resulting $X_i$ sequence has expected value 0 and variance $\sigma^2_X$. The $X_i$ are an identically distributed sequence, but are not independent.
The following R code generates one AR(1) sequence.

```r
n = 100     ## The sample size.
t2 = 1      ## The error variance.
alpha = 0.5 ## The AR(1) coefficient.

## The data variance.
s2 = t2 / (1 - alpha^2)

## Simulate the first value.
X = rnorm(1, sd=sqrt(s2))

## Simulate the rest of the sequence.
for (i in 2:n)
{
    X[i] = alpha*X[i-1] + rnorm(1, sd=sqrt(t2))
}
```
The following R code defines a function that returns a matrix whose rows are AR(1) sequences.

```r
## Generate nrep datasets from an AR(1) process.
arsim = function(alpha, t2, nrep, n) {
  ## The data variance.
  s2 = t2 / (1 - alpha^2)

  ## Storage for nrep dependent sequences of length n, each stored
  ## in a row of X.
  X = array(0, c(nrep,n))

  ## Simulate the first value.
  X[,1] = rnorm(nrep, sd=sqrt(s2))

  ## Simulate the rest of the sequence.
  for (i in (2:n))
    { X[,i] = alpha*X[,i-1] + rnorm(nrep, sd=sqrt(t2)) }
}
```
return(X)
}

Now we can see what happens to the variance of the sample mean $\bar{X}$ when the $X_i$ are dependent.

```r
t2 = 1          ## The error variance.
SS = c(5,10,20,40)  ## Sample sizes

## Consider these AR(1) coefficients.
for (alpha in c(0,0.3,0.6,0.9))
{
  V = NULL
  for (k in 1:length(SS))
  {
    ## The sample size.
    ss = SS[k]
    
    ## The AR(1) data.
    X = arsim(alpha, t2, nrep, ss)
    
    ## The mean of each row.
    Xbar = apply(X, 1, mean)
  }
}
```
## The sampling variance of Xbar.

\[ V[k] = \text{var}(Xbar) \]

} 

## Print out the results for a single value of alpha.

print(V)

}
For *iid* data, if the sample size doubles, the variance of \( \bar{X} \) is cut in half. How does the variance of \( \bar{X} \) for AR(1) data depend on the value of \( \alpha \) and on the sample size?

To understand what is happening here, we need to consider the covariance matrix of the \( X_i \) sequence. This is a \( n \times n \) matrix \( C \) whose \( i, j \) position is

\[
C_{i,j} = \text{cov}(X_i, X_j).
\]

Another fact, which we will not prove, is that the variance of \( \bar{X} \) is given by

\[
\text{var} \bar{X} = \sum_{ij} C_{ij}/n^2.
\]

Note that for *iid* data, \( C \) is a diagonal matrix with \( \sigma^2 \) along the diagonal, so this reduces to the familiar “\( \sigma^2/n \)” formula in the *iid* case.
It is a fact that for an AR(1) sequence,

$$\text{cov}(X_i, X_j) = \frac{\alpha^{|i-j|} \tau^2}{1 - \alpha^2}.$$

To derive this fact, note that we can write the AR(1) series as follows

$$X_i = \alpha X_{i-1} + \epsilon_i$$
$$\quad = \alpha(\alpha X_{i-2} + \epsilon_{i-1}) + \epsilon_i$$
$$\quad = \alpha^2 X_{i-2} + \alpha \epsilon_{i-1} + \epsilon_i$$

and carrying on as above $q$ times, we get

$$X_i = \alpha^q X_{i-q} + \alpha^{q-1}\epsilon_{i-q+1} + \alpha^{q-2}\epsilon_{i-q+2} + \cdots + \epsilon_i.$$
Since $X_i$ is independent of $\epsilon_j$ when $j > i$, we get

$$
\text{cov}(X_i, X_{i-q}) = \alpha^q \text{cov}(X_{i-q}, X_{i-q}) = \alpha^q \text{var}(X_{i-q}) = \alpha^q \tau^2 / (1 - \alpha^2).
$$

Now if we write $i - q = j$, we get

$$
\text{cov}(X_i, X_j) = \alpha^{i-j} \tau^2 / (1 - \alpha^2).
$$

when $j < i$. By symmetry,

$$
\text{cov}(X_i, X_j) = \alpha^{j-i} \tau^2 / (1 - \alpha^2).
$$

when $j > i$, so for any $i, j,$

$$
\text{cov}(X_i, X_j) = \alpha^{|i-j|} \tau^2 / (1 - \alpha^2).
$$
The following program calculates the value of $\sum_{ij} C_{ij}/n^2$ for various values of $\alpha$ and $n$. Compare the results to the previous simulation.

```r
t2 = 1 ## The error variance.
SS = c(5,10,20,40) ## Sample sizes

## Consider these AR(1) coefficients.
for (alpha in c(0,0.3,0.6,0.9))
{
  F = NULL
  for (k in 1:length(SS))
  {
    ## The sample size.
    n = SS[k]

    ## Construct the covariance matrix.
    C = array(0, c(n,n))
    for (i in (1:n))
    {
      for (j in (1:n))
      {
        for (j in (1:n))
        {
      ```
\[ C[i,j] = \alpha^{\text{abs}(i-j)} \cdot \frac{t2}{1-\alpha^2} \]

\[
F[k] = \frac{\text{sum}(C)}{n^2}
\]

\[
\text{print}(F)
\]
**Observations**

\( \text{var} \bar{X} \) decreases as the sample size increases.

\( \text{var} \bar{X} \) increases as \( \alpha \) increases.

When \( \alpha > 0 \), \( \text{var} \bar{X} \) drops by less than a factor of 1/2 when the sample size doubles.