

Estimating the expected value with non-*iid* data

For *iid* random variables X_1, \dots, X_n , the sample mean satisfies

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

and

$$\text{var}\bar{X} = \text{var}X_i/n.$$

These results must be generalized if the data are dependent, non-identically distributed, or both.

Unequal variances: First let's continue to assume that the X_i are independent, and that they all have the same expected value μ . However the distributions of the X_i may differ. For example, they may have different variances, in which case we can write $\text{var}X_i = \sigma_i^2$. The variances may differ in practice, for example, if individuals are assessed for the same trait using instruments of differing precision.

When the X_i have expected value μ and variance σ_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

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

The following simulation shows illustrates this fact. The σ_i^2 values are simulated as exponential random variables.

```

v <- NULL
vt <- NULL

## Consider sample means for different sample sizes.
for (p in c(5,10,15,20,25,30))
{
  ## Generate p variances and copy them across the columns of a px1000
  ## array.
  V <- rexp(p)
  W <- array(V, c(p,1000))

  ## Generate a px1000 array of normal draws all with expected value 0,
  ## with row i having variance V[i].
  X <- array(rnorm(p*1000), c(p,1000))
  X <- sqrt(W) * X

  ## Get the sample means.
  M <- apply(X, 2, mean)

  ## This is approximately the population variance of the sample means.
  v <- c(v, var(M))

  ## This is the theoretical value.
  vt <- c(vt, sum(V)/p^2)
}

```

Dependence: Now we turn to non-independent data. First we need some definitions that help to describe the dependence between two random variables. The *population covariance* between random variables X and Y , where pairs X, Y are observed jointly, is

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

The covariance can be any number. A positive value means that when X is

observed to be greater than its mean, Y tends to be greater than its mean as well. A negative covariance means that when X is observed to be greater than its mean, Y tends to be less than its mean. When the covariance is zero, knowing that X is greater than its mean provides no information about whether Y is greater or less than its mean. Note that these statements are symmetric – swap the labels “ X ” and “ Y ” and the statements are equally true.

The *correlation coefficient* between X and Y is

$$\text{corr}(X, Y) = \frac{\text{cov}(X, Y)}{\text{SD}(X) \cdot \text{SD}(Y)}.$$

The correlation coefficient is closely related to the covariance, but it always falls between -1 and 1 . This provides a more universal scale for measuring association between two random variables.

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

$$\widehat{\text{cov}}(X, Y) = \frac{1}{n-1} \sum_i (X_i - \bar{X})(Y_i - \bar{Y}),$$

and the sample correlation coefficient is

$$\widehat{\text{corr}}(X, Y) = \frac{\widehat{\text{cov}}(X, Y)}{\hat{\sigma}_X \hat{\sigma}_Y}.$$

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

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

To generate the data, let X_1 be normal with expected value 0 and variance σ_X^2 . Then generate the subsequent X_i values according to the rule

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

where the ϵ_i are independent and normal with expected value zero and variance τ^2 . The ϵ_i are called the *errors* and τ^2 is the *error variance*.

It is a fact that each term of the resulting X_i sequence has expected value 0 and variance σ_X^2 . The X_i are an identically distributed sequence, but are not independent.

The following R code generates AR(1) data.

```
## The sample size.
n <- 100

## The error variance.
t2 <- 1

## The AR(1) coefficient.
alpha <- 0.5

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

## Simulate the first value separately.
X <- sqrt(s2)*rnorm(1)

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

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

```

## The error variance.
t2 <- 1

## Consider these AR(1) coefficients.
for (alpha in c(0,0.2,0.5,0.8))
{
  ## The data variance.
  s2 <- t2 / (1 - alpha^2)

  ## Storage for the variances of the sample means.
  V <- NULL

  for (n in c(5,10,20,40))
  {
    ## Storage for 1000 dependent sequences of length n, each stored
    ## in a column of X.
    X <- array(0, c(n, 1000))

    ## Simulate the first value separately.
    X[1,] <- sqrt(s2)*rnorm(1000)

    ## Simulate the rest of the sequence.
    for (i in (2:n))
    {
      X[i,] <- alpha*X[i-1,] + sqrt(t2)*rnorm(1000)
    }

    v <- var(colMeans(X))
    V <- c(V, v)
  }

  ## 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 α 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 defined to be

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

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

$$\begin{aligned} X_t &= \alpha X_{t-1} + \epsilon_t \\ &= \alpha(\alpha X_{t-2} + \epsilon_{t-1}) + \epsilon_t \\ &= \alpha^2 X_{t-2} + \alpha \epsilon_{t-1} + \epsilon_t \end{aligned}$$

and carrying on as above q times, we get

$$X_t = \alpha^q X_{t-q} + \alpha^{q-1} \epsilon_{t-q+1} + \alpha^{q-2} \epsilon_{t-q+2} + \dots + \epsilon_t.$$

We will also need two facts. First, that

$$\text{cov}(A_1 + A_2 + \dots + A_r, B) = \text{cov}(A_1, B) + \text{cov}(A_2, B) + \dots + \text{cov}(A_r, B),$$

and second, that since X_t is independent of ϵ_s when $s > t$, that $\text{cov}(X_t, \epsilon_s) = 0$ when $s > t$. Thus we get

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

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 σ^2 along the diagonal, so this reduces to the familiar “ σ^2/n ” formula in the *iid* case.

The following program calculates the value of $\sum_{ij} C_{ij}/n^2$ for various values of α and n . Compare the results to the simulation given above.

```
## The error variance.
t2 <- 1

## Consider these AR(1) coefficients.
for (alpha in c(0,0.2,0.5,0.8))
{
  F <- NULL

  ## Consider these sample sizes.
  for (n in c(5,10,20,40))
  {
    C <- array(0, c(n,n))
    for (i in (1:n))
    {
      for (j in (1:n))
      {
        C[i,j] = alpha^abs(i-j)*t2/(1-alpha^2)
      }
    }

    F <- c(F, sum(C)/n^2)
  }

  print(F)
}
```

What we see here is that $\text{var}\bar{X}$ decreases with sample size, but increases as α increases. However it does not necessarily decrease in an inverse linear pattern with respect to sample size (doubling the sample size does not necessarily cut the variance in half). The reason for this is that dependent data are less informative than independent data. The greater the value of α , the less information is present in the sample, and hence \bar{X} is a less precise (but still unbiased) estimate of the expected value.