# 606 Assignment 3

*Brett Burk*

*Wednesday, February 25, 2015*

**4.24.i** Using the definition of variance:

$V[X] = E[(X-\mu)^2] = ((1-3.5)^2+(2-3.5)^2+(3-3.5)^2+(4-3.5)^2+(5-3.5)^2+(6-3.5)^2)/6 = (6.25+2.25+.25)/3 = 2.91\bar{6}$

**4.24.ii** Using $V[X] = E[X^2] - E[X]^2$:

$$((1+4+9+16+25+36)/6) - 3.5^2 = 2.91\bar{6}$$

I prefer the second because it is computationally simpler.
**4.36**

$$Var[X+Y] = Var[X] + Var[Y] + 2Cov(X,Y)$$

$$Cov(X,Y) = -0.5\sigma_x\sigma_y$$

$$Var[X+Y] = var[X] + Var[Y] - 1\sigma_x\sigma_y$$

Assuming by 'the same distribution' it is meant that mean(x) = mean(y) and var(x) = var(y):

$$Var[X+y] = 2var[X] - var[x] = Var[X] = Var[Y]$$

**4.40.a**

$$V[3X^2 - Y] = 9V[X^2] + V[Y]$$

$$9(E[X^4] - E[X^2]^2) + (E[Y^2] - E[Y]^2)$$

$$9(15-4) + (6-4) = 101$$

**4.40.b**

$$Cov(X,Y) = E[XY] - E[X]E[Y]$$

As they are independent

$$0 = E[XY] - E[X]E[Y]$$

$$E[X]E[Y] = E[XY]$$

$$15 * 14 = 1410$$

**4.40.c**

$$Cov(X,X^2) = E[X^3] - E[X]E[X^2]$$

$$5 - 1 * 2 = 3$$

**4.40.d**

$$V[X^2Y^2] = E[X^4Y^4] - (E[X^2]E[Y^2])^2 = E[X^4]E[Y^4] - (E[X^2]E[Y^2])^2 = 15 \times 94 - (2 \times 6)^2 = 1266$$

$$15 \times 94 - 4 \times 6 = 1386$$

**4.56**

```r
fourthree <- function(){
  roll <- sample(1:6,1)
  if(roll <= 3){
    return(-10)
  }
  if(roll %in% c(4,5)){
    return(0)
  }
  else{
    return(14)
  }
}
ft <- vector(length = 10000)
ft <- sapply(1:10000, function(x) fourthree())
expectation <- sum(ft)/length(ft)
expectation
```

```
## [1] -2.488
```

```r
variance <- sum(ft^2)/length(ft) - expectation^2
variance
```

```
## [1] 77.48986
```

(I shifted the answer on this one, as my original assumed 'winning \$24' meant getting back your money in addition to the \$24) **4.58**

```r
x.dist <- rpois(100000,1)
y.dist <- rpois(100000,2)
```

4.40.a:

```r
var(3*(x.dist^2) - y.dist)
```

```
## [1] 97.45811
```

4.40.b:

```r
mean(x.dist^4 * y.dist^4)
```

```
## [1] 1397.323
```

4.40.c:

```r
cov(x.dist, x.dist^2)
```

```
## [1] 2.933126
```

4.40.d:

```r
var(x.dist^2 * y.dist^2)
```

```
## [1] 1255.161
```

**5.4.a**

```r
 0.99^109 * .01^1
```

```
## [1] 0.003343769
```

**5.4.b**

```r
.99^109
```

```
## [1] 0.3343769
```

**5.4.c**
100

**5.12**

$$E[K] = \frac{k}{4}(\binom{k}{4}p^4(1-p)^{k-4} + \binom{k}{4}(1-p)^4 p^{k-4}) \text{ for } k = 4, 5, 6, 7$$

**5.20.a**

$$\mu_s = \frac{\text{tagged} \times \text{caught}}{\text{total}} = 2$$

$$\sigma_s = \frac{n(N-n)r(N-r)}{N^2(N-1)} \approx 1.73146$$

**5.20.b**

```r
little.n <- 20

big.n <- 500
big.k <- 50

temp.p <- 1
for(k in 0:2){
  temp.p <- temp.p - ((choose(big.k, k) * choose(big.n - big.k, little.n - k))/choose(big.n, little.n))

}
temp.p
```

```
## [1] 0.3224539
```

**5.28.a**

$$Cov(X,Y)_n = n(P_{Both} - P_X P_Y) = 10(0 - \frac{1}{12}) = \frac{-5}{5}$$

**5.28.b**

$$10(\frac{1}{6} - \frac{1}{12}) = \frac{5}{6}$$

**5.32** Assuming that homework could be given on the last class, we have that homework could be given on classes 2,3,5,7,11,13,17,19,23,29, thus we have $\frac{14}{42}$.
Let X be the number of days with no homework, with $n = 42$ and $p = \frac{1}{3}$ then it has a $\mu$ of 14.

**5.34** This is a hypergeometric distribution with $N = 30, n = 10, K = 6, and k = 3$ thus we have

$$\frac{\binom{6}{3}\binom{24}{7}}{\binom{30}{10}} = 0.23039$$

**5.38**

```
set.seed(01191984)
coupon <- function(n,p){
  have <- array(FALSE, dim=n)
  count <- 0
  while (sum(have) < n)
    {
      picked <- sample(1:n, 1, prob=p, replace=TRUE)
      have[picked] <- TRUE
      count <- count + 1
    }
  return(count)
}

p <- dbinom(0:6, 6, 0.5)
# p <- dbinom(0:52, 52, 0.5)
coupon(length(p), p)
```

```
## [1] 82
```

```
curr.counts <- array(0, dim=10000)
curr.counts <- sapply(1:length(curr.counts), function(x) coupon(7,p))
mean(curr.counts)
```

```
## [1] 95.6506
```

```
sd(curr.counts)
```

```
## [1] 68.64706
```

**5.39**

```
# Returns TRUE if they won the series
series.calc <- function(prob) {
  series.samp <- rbinom(4, 1, prob)
  while(length(series.samp) < 7 && max(table(series.samp)) < 4){
    series.samp <- c(series.samp, rbinom(1,1,prob))
  }
  return(sum(series.samp)/length(series.samp) > 0.5)
}

prob <- 0.25
series.low.counts <- array(0, dim=10000)
series.low.counts <- sapply(1:length(series.low.counts), function(x) series.calc(prob))
mean(series.low.counts)
```

```
## [1] 0.0731
```

```r
sd(series.low.counts)
```

```
## [1] 0.2603136
```

```r
prob <- 0.6
series.high.counts <- array(0, dim=10000)
series.high.counts <- sapply(1:length(series.high.counts), function(x) series.calc(prob))
mean(series.high.counts)
```

```
## [1] 0.7064
```

```r
sd(series.high.counts)
```

```
## [1] 0.4554336
```