Unit 2 Slide 121 Example Solution Using R

Hanning Su
Student ID: 855767

August 21, 2020

R code

We first calculate μ_r by the formula:

$$\mu_r = \mathbb{E}\left[(1+i_k)^{-r}\right] = \frac{0.15}{1.05^r} + \frac{0.60}{1.06^r} + \frac{0.25}{1.07^r}$$

```
mu_1 <- (1/1.05)*0.15 + (1/1.06)*0.6 + (1/1.07)*0.25
mu_2 <- (1/1.05^2)*0.15 + (1/1.06^2)*0.6 + (1/1.07^2)*0.25
mu_3 <- (1/1.05^3)*0.15 + (1/1.06^3)*0.6 + (1/1.07^3)*0.25
```

Implement recursive function for the first three moments:

$$E\left[a_n\right] = \mu_1\left(1 + E\left[a_{n-1}\right]\right)$$

$$E\left[a_n^2\right] = \mu_2\left(1 + 2E\left[a_{n-1}\right] + E\left[a_{n-1}^2\right]\right)$$

$$E\left[a_n^3\right] = \mu_3\left(1 + 3E\left[a_{n-1}\right] + 3E\left[a_{n-1}^2\right] + E\left[a_{n-1}^3\right]\right)$$

```
moment_calculator <- function (n, r) {
  if (r == 1)
  {
    if (n == 1) {
      return( mu_1 ) #Base case
    } else {
      return (mu_1 * (1 + moment_calculator(n - 1, r))) #Recursive case
    }
  }
  else if (r == 2)
  {
    if (n == 1) {
      return( mu_2 ) #Base case
    } else {
      return (mu_2 * (1 + 2 * moment_calculator(n - 1, 1)
                      + moment_calculator(n - 1, r))) #Recursive case
    }
  }
  else if (r == 3)
  {
    if (n == 1) {
      return( mu_3 ) #Base case
    } else {
      return (mu_3 * (1 + 3 * moment_calculator(n - 1, 1) + 3 * moment_calculator(n - 1, 2)
                      + moment_calculator(n - 1, r))) #Recursive case
    }
  }
  else
```

```
  {
    print("higher moment functionality not yet available.")
  }
}
```

```
#Prepare dataframe for the resultes
df <- data.frame(n = c(1:20),
                 First_Moment = c(rep(NA, 20)),
                 Second_Moment = c(rep(NA, 20)),
                 Third_Moment = c(rep(NA, 20)))

#Populate the dataframe
for (r in 2:4) {
  for (n in 1:20){
    df[[r]][n] = moment_calculator(n, r - 1)
  }
}

#Table created using Stargazer
```

Table 1: Moments

n	First_Moment	Second_Moment	Third_Moment
1	0.943	0.888	0.837
2	1.831	3.352	6.139
3	2.668	7.120	19.000
4	3.457	11.955	41.339
5	4.201	17.653	74.175
6	4.902	24.036	117.857
7	5.563	30.953	172.235
8	6.186	38.273	236.810
9	6.773	45.882	310.840
10	7.327	53.686	393.426
11	7.848	61.601	483.580
12	8.340	69.561	580.273
13	8.803	77.505	682.477
14	9.240	85.387	789.188
15	9.651	93.164	899.447
16	10.039	100.806	1,012.354
17	10.405	108.283	1,127.077
18	10.750	115.577	1,242.853
19	11.075	122.668	1,358.995
20	11.381	129.546	1,474.887