Module 1

This is a single, concatenated file, suitable for printing or saving as a PDF for offline viewing. Please note that some animations or images may not work.

Module 1 Study Guide and Deliverables

Topics:	**Lecture 1:** Introduction to Enterprise Risk Analytics **Lecture 2:** Analyzing Risk in the Enterprise
Readings:	Lectures 1 and 2 online content
Discussions:	**Module 1 Discussion:** Introduce Yourself (non-graded)
Assignments:	**Assignment 1**: Individual Assignment covering Lecture 1 and Lecture 2. **Tutorial:** Tutorial 1 for solving Assignment 1
Assessments:	**Quiz 1**

Lecture 2

Learning Objectives

After you complete this lecture, you will be familiar with the following:

- Incorporating uncertainty into a deterministic spreadsheet model
- Using R for Monte Carlo simulation
- Interpreting the results of the simulation

Outsourcing Decision—Deterministic Spreadsheet Model[1]

In this module, we will work with the following simple outsourcing decision model:

Suppose that a manufacturer can produce a part for $125/unit with a fixed cost of $50,000. The alternative is to outsource production to a supplier at a unit cost of $175. Assume that the production volume is fixed and is equal to 1500. All parts should either be manufactured in-house or should be outsourced.

Decision point: The manufacturer wants to know which one is the best decision? Outsourcing or manufacturing in-house?

Adapted from Evans, J. R. (2016). Business analytics: Methods, models, and decisions (2nd edition). Pearson Education, Inc., pp. 22, 344–345.

We will first build an R model for the outsourcing decision.

Individual Exercise:

Before you read any further, please try to build the R model on your own.

Figure 2.1 presents the resulting model:

Figure 2.1

```
# MANUFACTURING COST
ManFC<-50000 #Fixed cost to manufacture
ManVC<-125 #Unit variable cost to manufacture

# OUTSOURCING COST
OutVC<-175 #Unit variable cost to outsource

# PRODUCTION VOLUME
Volume<-1500

#Output
ManTC<-ManFC+Volume*ManVC #Total cost to manufacture
OutTC<-Volume*OutVC #Total cost to outsource

Diff<-ManTC-OutTC #difference between manufacturing and out:
BestToManufacture<-Diff<0 #is it best to manufacture?

Diff
BestToManufacture
```

```
> Diff
[1] -25000
> BestToManufacture
[1] TRUE
```

The inputs for ManVC, ManFC, OutVC and Volume, namely, the fixed cost, unit variable cost, unit cost of outsourcing and the production volume, are given in the description of the problem. We calculate the total manufacturing cost, ManTC as follows:

Total manufacturing cost = Fixed cost + unit variable cost × production volume

ManTC=ManFC + ManVC × Volume

Thus, ManTC=$50,000 + $125 × 1500=$237,500.

We calculate the total outsourcing cost in cell B17 as follows:

Total outsourcing cost = unit cost (outsourced from supplier) × production volume (outsourced from supplier)

Cell OutTC=OutVC × Volume

Thus, OutTC= $175 × 1500 = $262,500.

In making the decision, what is important for us is the cost difference. Therefore, "Diff" calculates the difference between the total manufacturing cost and the total outsourcing cost; i.e, ManTC – OutTC. If this difference is smaller than zero, then this means that manufacturing in-house is less costly and the manufacturer should choose to manufacture in-house. Otherwise, the manufacturer should choose to outsource. We found that this difference is $237,500 – $262,500 = –$25,000. Thus, the best decision in this case turns out to be to manufacture in-house. Our model communicates the best decision to us in "BestToManufacture", which returns TRUE when ManTC is lower than OutTC and FALSE otherwise:

BestToManufacture = Diff<0

We are done with developing the R model for our problem. One of the major assumptions that we made is that all inputs are known with certainty. However, as we discussed in Lecture 1, uncertainty is inevitable and it may impact our decisions to a great extent. Now, assume that the production volume is uncertain in our model. This uncertainty of the production volume usually stems from uncertainty in the demand process; therefore we will use the terms "production volume" and "demand" interchangeably in this module. In the next section, we will learn how to incorporate this uncertainty into our R model.

Incorporating Uncertainty into the Model

The assumption that the demand and hence the production volume is uncertain is quite reasonable and indeed is a representation of a real-life situation. No one has a way of knowing the actual value of the demand; we can only make forecasts.

We model our uncertain variable — the demand (production volume) — as a *random variable* having a *probability distribution*. In statistics, a "*random variable* is any variable whose value cannot be predicted or set with certainty."[2] A *probability distribution* characterizes all values that a random variable may take. We will learn more about probability distributions in Lecture 3.

Individual Exercise:

Topics related to the notion of distribution are covered in Unit 3 of the laboratory "Statistics with Applications in Management." Please review the corresponding module.

One of the most widely used probability distributions is the normal distribution. "Assume that the demand is normally distributed with a mean of 1000 and a standard deviation of 100."[3] How do we come up with the numbers 1000 and 100? It will be clear in Lecture 5. For now, just follow along with me.

Question: How do we generate random values of demand in R?

Answer: We will use the R function rnorm(n, mean, sd) to generate random values from the normal distribution.

The rnorm function generates "n" random variables, normally distributed with mean "mean" and standard deviation "sd". Right now, we're going to content ourselves with 15 trials, so we'll use rnorm(15, 1000, 100). Note it would also be good syntax to write rnorm(n=15, mean=1000, sd=100).

Figure 2.2

```
> rnorm(15,1000,100)
 [1]  960.1202 1060.7778 1145.9164  985.9110 1096.1880
 [6] 1009.3442  983.4215  977.3950  946.3878  962.4008
[11]  994.8039  895.8200 1141.7891 1030.6372 1006.3650
```

If we don't want our result to return fractional values, we can also apply the round() function, as follows. Here I use the %>% operator from the magrittr library. It would work just as well to type round(rnorm(15, 1000, 100)), but %>% makes for more readable code.

Figure 2.3

```
> rnorm(15,1000,100) %>% round()
 [1] 1030 1066  888 1128  957  934 1101  976  939
[10] 1176  962  744  802  869 1064
```

Since we are generating random numbers, you should expect to get a different result from what you observe here, and indeed a different result each time you run this line of code.

Since the demand is random in real life, every time the *rnorm* function runs, a different demand is returned.

Question: So, how do we make our final decision?

Answer: We randomly generate the production volume and the associated decision several times and look at the distribution of the output (i.e., best decision) that we get.

For instance, if we generate 200 different demands, and record the best decision (either manufacture or outsource) each time; and if out of 200 scenarios, we get 150 outsource and 50 manufacture as the best decision, then this suggests the best decision under demand uncertainty is to outsource. This is exactly what Monte Carlo simulation does!

As you recall from Lecture 1, **Monte Carlo (MC) simulation** is the process of generating random values for uncertain inputs in a model, computing the output variables of interest, and then repeating this process for many trials to understand the distribution of the output results.[4] We will see how to implement MC simulation below.

Scalars and Vectors in R

For our purposes, a *scalar* is simply a number and a *vector* is a list of numbers with a set length.

Fortunately for us, when you perform a function using a scalar (such as ManVC) and a vector of length *n* such as volume in our new model, R will create a new vector of length *n* where each element is the output from the function applied to each element.

When you perform a function using two vectors ofequal length *n*, R will create a new vector of length *n* where each element is the function called on those corresponding elements. This may sound confusing, but it's easy to see what's going on in practice. Observe the examples in Figure 2.4:

(Note we create vectors manually by using the c() function):

Figure 2.4

```
> c(1,2,3)+1
[1] 2 3 4
> c(1,2,3)+c(10,20,30)
[1] 11 22 33
```

Here we first added the number 1 to the vector (1, 2, 3) and it returned the result of adding one to each element (1+1, 2+1, 3+1) Below that, we added the vector (1, 2, 3) to (10,20,30) and it returned the result of adding each element to the corresponding element in the next vector (10+1, 20+2, 30+3). We will use this to combine certain elements of our model (scalars) with uncertain elements (vectors).

Creating a model

We've already created a deterministic model for our outsourcing problem. We need only replace our determined (scalar) volume with an uncertain (vector) equivalent. Continuing with the example above, let's assume our demand (and thus, production volume) is normally distributed with mean 1000 and standard deviation 100, and for the purposes of demonstration, let's assume we only want to run 15 trials. We first add lines to define the number of trials, and the parameters of our volume distribution:

Figure 2.5

```
# NUMBER OF TRIALS
trials<-15
```

Figure 2.6

```
# PRODUCTION VOLUME
meanVol<-1000 #mean of volume random variable
sdVol<-100 #standard deviation of volume random variable
```

We then replace the old scalar production volume with a new vector of production volumes, using the parameters we defined above:

Figure 2.7

```
Volume<-rnorm(trials,meanVol,sdVol) %>% round()
```

Because of R's vector/scalar handling, our old formulas will still work perfectly. We may, however, wish to display our results in a logical manner. Here the cbind function* is called to create a matrix displaying our randomly generated volume, the difference between manufacturing and outsourcing, and whether it's best to manufacture, in one table. We use the as.data.frame function so that the resulting output will be nicely formatted. The full code and results are displayed in Figure 2.9.

Figure 2.8

```
result<-cbind(Volume,Diff) %>% as.data.frame() %>%
  cbind(BestToManufacture)
result
```

*We could call cbind just once here, but that would coerce BestToManufacture into an integer. In order to keep it as a Boolean, we must first create the dataframe.

Figure 2.9

```r
# NUMBER OF TRIALS
trials<-15

# MANUFACTURING COST
ManFC<-50000 #Fixed cost to manufacture
ManVC<-125 #Unit variable cost to manufacture

# OUTSOURCING COST
OutVC<-175
# PRODUCTION VOLUME
meanVol<-1000 #mean of volume random variable
sdVol<-100 #standard deviation of volume random variable
Volume<-rnorm(trials,meanVol,sdVol) %>% round()

#Output
ManTC<-ManFC+Volume*ManVC #Total cost to manufacture
OutTC<-Volume*OutVC #Total cost to outsource

Diff<-ManTC-OutTC #difference between manufacturing and outsourcing
BestToManufacture<-Diff<0 #is it best to manufacture?

result<-cbind(Volume,Diff) %>% as.data.frame() %>%
  cbind(BestToManufacture)
result
```

```
> result<-cbind(Volume,Diff) %>% as.data.frame() %>%
+   cbind(BestToManufacture)
> result
   Volume  Diff BestToManufacture
1     901  4950             FALSE
2     904  4800             FALSE
3     832  8400             FALSE
4     959  2050             FALSE
5     772 11400             FALSE
6    1129 -6450              TRUE
7     981   950             FALSE
8     983   850             FALSE
9     894  5300             FALSE
10    967  1650             FALSE
11    883  5850             FALSE
12   1089 -4450              TRUE
13    970  1500             FALSE
14   1121 -6050              TRUE
15   1070 -3500              TRUE
```

Monte Carlo Simulation

Now let's further assume that the cost to purchase from a supplier is also uncertain. All we know is that the minimum it may cost is \$160, the maximum is \$200, and the most likely value is \$175. We'll use the triangular distribution* to model this behavior. Determining uncertain inputs and which distributions to use is the most difficult part of simulation modeling; for now, we'll accept the distributions as outlined by the problem.

Figure 2.10

```r
#GENERATE TRIANGULARLY DISTRIBUTED RV's
rtri<-function(n,min,ml,max){
  qtri<-function(U){
    F<-(ml-min)/(max-min)
    if (U<F) {min+(U*(max-min)*(ml-min))^.5}
    else {max-((1-U)*(max-min)*(max-ml))^.5}
  }
  y<-runif(n)
  sapply(y,qtri)
}
```

*You are familiar with the normal distribution but you are not familiar with the triangular distribution yet. This is completely fine. To comprehend this section, you do not need to know the details about the triangular distribution. It will be covered in Lecture 3. Base R does not contain a function to randomly generate variables following a triangular distribution, so we're compelled to define our own. The code is reproduced in Figure 2.10.

Instead of running 15 trials, we'll run 10,000. We won't be able to neatly look at them all as we did before, but we will be able to make stronger inferences and summarize the results with numbers and graphics. The tradeoff is that a larger number of trials requires more computational power. We will see in Lecture 4 how to determine how many trials to run.

As with rnorm, we tell our rtri function the number of trials we want and the parameters we're using (see figure 2.11).

We'll need the e1071 package and the tidyverse package to generate some statistics and images for our model. The finished model is reproduced in Figure 2.12.

Figure 2.11

```
# OUTSOURCING COST
OutVCmin<-160
OutVCml<-175
OutVCmax<-200
OutVC<-rtri(trials,OutVCmin,OutVCml,OutVCmax) #Unit variable cost to outsource
```

Figure 2.12

```
library(tidyverse)
library(stats)
library(e1071)

#GENERATE TRIANGULARLY DISTRIBUTED RV's
rtri<-function(n,min,ml,max){
  qtri<-function(U){
    F<-(ml-min)/(max-min)
    if (U<F) {min+(U*(max-min)*(ml-min))^.5}
    else {max-((1-U)*(max-min)*(max-ml))^.5}
  }
  y<-runif(n)
  sapply(y,qtri)
}

# NUMBER OF TRIALS
trials<-10000

# MANUFACTURING COST
ManFC<-50000 #Fixed cost to manufacture
ManVC<-125 #Unit variable cost to manufacture

# OUTSOURCING COST
OutVCmin<-160
OutVCml<-175
OutVCmax<-200
OutVC<-rtri(trials,OutVCmin,OutVCml,OutVCmax) #Unit variable cost to outsource

# PRODUCTION VOLUME
meanVol<-1000 #mean of volume random variable
sdVol<-100 #standard deviation of volume random variable
Volume<-round(rnorm(trials,meanVol,sdVol))

#Output
ManTC<-ManFC+Volume*ManVC #Total cost to manufacture
OutTC<-Volume*OutVC #Total cost to outsource

Diff<-ManTC-OutTC #difference between manufacturing and outsourcing
BestToManufacture<-Diff<0 #is it best to manufacture?
```

With a little more work, we can determine several key statistics about our data (see Figure 2.13). We find the following results, which are explained in further detail below:

Figure 2.13

```
> mean(Diff) #various summary statistics
[1] -3412.851
> sd(Diff)
[1] 9852.455
> var(Diff)
[1] 97070864
> skewness(Diff)
[1] -0.3528949
> kurtosis(Diff)
[1] -0.1738879
> min(Diff)
[1] -42332.86
> max(Diff)
[1] 22769.2
> max(Diff)-min(Diff) #range
[1] 65102.06
```

Interpreting the Model

When we run the simulation for 10,000 trials, for each trial R calculates a cost difference (Diff: the manufacturing in-house – outsourcing). At the end of the simulation, we have a data set of the cost differences for each trial, $x_1, x_2, \ldots, x_N$.

- x_1 is the cost difference obtained in the first trial
- x_2 is the cost difference obtained in the second trial.
- x_N is the cost difference obtained in the Nth trial.

Variance is a widely used measure of variability. It is the mean squared deviation of the observations from the mean. It is calculated as follows:

$$s^2 = \frac{\sum_{i=1}^{N}(x_i - \bar{x})^2}{N-1}$$

Standard deviation is obtained by taking the square root of the variance, and gives us some idea of the dispersion of the data set. Unlike variance, it is in the same units as the output ($ here). We found a standard deviation of $9852. It's common to think of one observation in terms of how many standard deviations from the mean it is, so for example if we observed a difference of $5000, we would say it was

$$\frac{\$5000 - (-\$3412)}{\$9852} = .85$$

.85 standard deviations from the mean.

Skewness is a measure of asymmetry of a probability distribution. A negative skewness implies that the distribution is left-skewed and a positive skewness implies that the distribution is right-skewed.

Our Skewness is equal to –0.35, which suggest that the frequency distribution of the cost difference is left-skewed. This can be easily observed from the histogram in Figure 2.17.

Kurtosis measures whether a distribution has

i. A narrow peak and long tails (positive), or
ii. ii. A wider peak and short tails (negative)

relative to a distribution with similar variance.

Kurtosis is equal to –0.17, which suggest that the frequency distribution of the cost difference has short tails.

Minimum is the smallest value in the data set.

For the outsourcing decision model, the smallest cost-difference value obtained out of 10,000 trials is –$42,332.

Maximum is the largest value in the data set.

For the outsourcing decision model, the largest cost-difference value obtained out of 10,000 trials is $22,769.

A minimum cost difference of –$42,332 and a maximum cost difference of $22,769 suggest to manufacture in-house.

> **Note**: Do not interpret the minimum and the maximum as the worst case; as the number of iterations increases, they will typically continue to spread. However, these are the estimates of the best- and worst-case results that can be expected.

Range is the difference between the maximum value and the minimum value of a data set.

For the outsourcing decision model, the range is $22,769 – (–$42,332) = $65,012.

We can quickly generate a 5-number summary using the summary function on Diff. This will show us the minimum, 25th percentile, median, 75th percentile, and maximum of our data set.

Figure 2.14

```
> summary(Diff) #fast five number summary
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
 -42333   -9878   -2701   -3413    3771   22769
```

If we wanted to know how often it was better to manufacture, we can see how often the Diff variable, that is, the cost to manufacture minus the cost to outsource, is less than 0 by seeing how often the BestToManufacture variable is TRUE. Booleans, when converted to integers, are coerced to 1 for TRUE and 0 for FALSE, so we need only sum up the number of TRUES and divide by the number of trials. We observe about 60% of the time, manufacturing is the correct decision.

Figure 2.15

```
> #how likely is it that manufacturing is the best decision?
> sum(BestToManufacture)/trials
[1] 0.6036
```

Individual exercise:

How often would the company save at least $1000 by manufacturing as opposed to outsourcing?

Graphing Our Model

Let's use a histogram to graph our results. The code I'm using is in Figure 2.16, and requires ggplot from the tidyverse library. The resulting graph is in Figure 2.17.

We use the as.data.frame() function to coerce Diff into a form ggplot can read.

..count.. is a special element of ggplot which counts the number of observations within a bin, allowing us to get the probability density along the y-axis.

binwidth controls the width of a horizontal bar on the x-axis (in this case 2000).

If we wanted to control the number of bins instead, we could use bins instead.

ggtitle, xlab, and ylab all allow us to control the labels and the title.

Finally, the geom_vline call inserts a vertical line at x=0, where outsourcing and manufacturing break even.

Figure 2.16

```
#create a histogram of the trials
ggplot(as.data.frame(Diff))+
  geom_histogram(aes(x=Diff,y=..count../sum(..count..)),
                 binwidth=2000,color="dark blue",fill="blue")+
  ggtitle("Manufacturing - Outsourcing Costs")+
  xlab("difference") + ylab("probability")+
  geom_vline(xintercept=0,color="red")
```

Figure 2.17

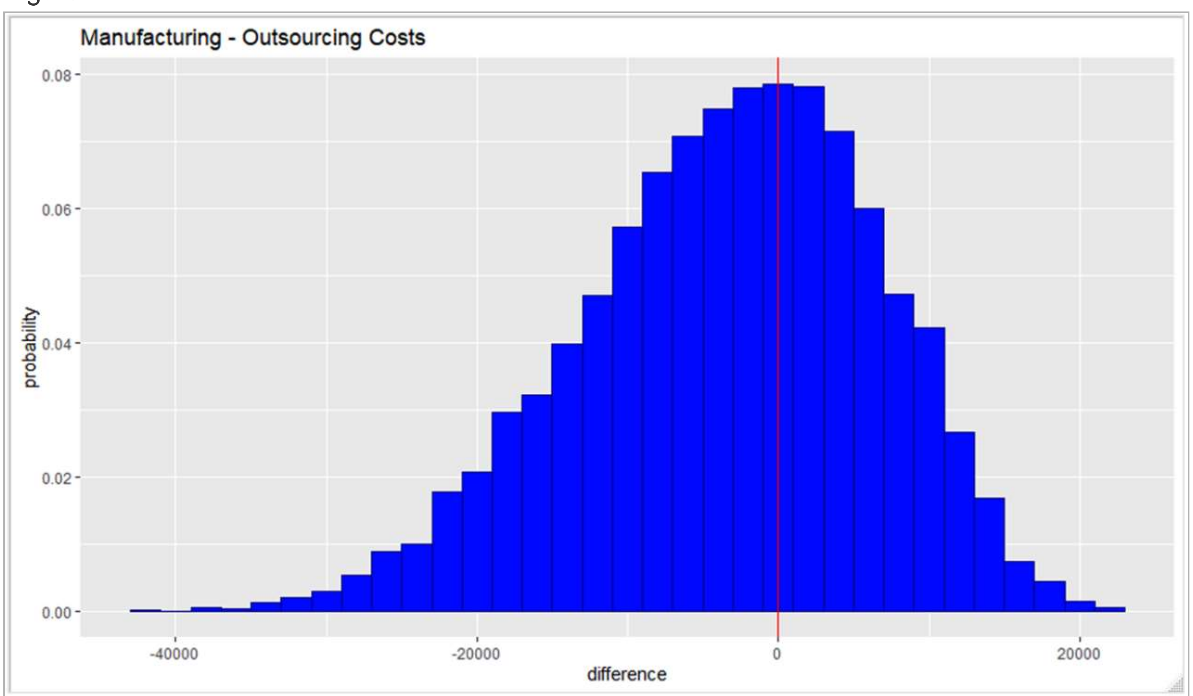

Figure 2.18

```r
library(tidyverse)
library(stats)
library(e1071)

#GENERATE TRIANGULARLY DISTRIBUTED RV's
rtri<-function(n,min,ml,max){
  qtri<-function(U){
    F<-(ml-min)/(max-min)
    if (U<F) {min+(U*(max-min)*(ml-min))^.5}
    else {max-((1-U)*(max-min)*(max-ml))^.5}
  }
  y<-runif(n)
  sapply(y,qtri)
}

# NUMBER OF TRIALS
trials<-10000

# MANUFACTURING COST
ManFC<-50000 #Fixed cost to manufacture
ManVC<-125 #Unit variable cost to manufacture

# OUTSOURCING COST
OutVCmin<-160
OutVCml<-175
OutVCmax<-200
OutVC<-rtri(trials,OutVCmin,OutVCml,OutVCmax) #Unit variable cost to outsource

# PRODUCTION VOLUME
meanVol<-1000 #mean of volume random variable
sdVol<-100 #standard deviation of volume random variable
Volume<-round(rnorm(trials,meanVol,sdVol))

#Output
ManTC<-ManFC+Volume*ManVC #Total cost to manufacture
OutTC<-Volume*OutVC #Total cost to outsource

Diff<-ManTC-OutTC #difference between manufacturing and outsourcing
BestToManufacture<-Diff<0 #is it best to manufacture?

mean(Diff) #various summary statistics
sd(Diff)
var(Diff)
skewness(Diff)
kurtosis(Diff)
min(Diff)
max(Diff)
max(Diff)-min(Diff) #range

summary(Diff) #fast five number summary

#how likely is it that manufacturing is the best decision?
sum(BestToManufacture)/trials

#create a histogram of the trials
ggplot(as.data.frame(Diff))+
  geom_histogram(aes(x=Diff,y=..count../sum(..count..)),
                 binwidth=2000,color="dark blue",fill="blue")+
  ggtitle("Manufacturing - Outsourcing Costs")+
  xlab("difference") + ylab("probability")+
  geom_vline(xintercept=0,color="red")
```

Lecture 1 Footnotes

[1] Evans, J. R. (2016). *Business analytics: Methods, models, and decisions* (2nd edition). Pearson Education, Inc., pp. 344-345.

[2] Ragsdale, C. T., (2015). *Spreadsheet modeling and decision analysis: A practical introduction to business analytics* (7th edition). Cengage Learning, p. 619.

[3] Evans, J. R. (2016). *Business analytics: Methods, models, and decisions* (2nd edition). Pearson Education, Inc., p. 379.

[4] Evans, J. R. (2016). *Business analytics: Methods, models, and decisions* (2nd edition). Pearson Education, Inc., p. 379.

Lecture 1 References

Evans, J. R. (2016). *Business analytics: Methods, models, and decisions* (2nd edition). Pearson Education, Inc.

Ragsdale, C. T., (2015). *Spreadsheet modeling and decision analysis: A practical introduction to business analytics* (7th edition). Cengage Learning.

Lecture 1 Summary Questions

1. How do we model an uncertain variable in R?
2. What is a “random variable” and what is a “probability distribution”?
3. What is the R function used to generate random values from the normal distribution?

Module 1 Individual Exercise

The problems in the individual exercise is designed for you to practice what you have learnt in Lecture 1 and Lecture 2. Do not submit anything to your facilitator and/or instructor.

Question: A mortgage lender wants to see how many applications they can process in 20 days. A mortgage application should be processed through six departments (department A to department F) before it is finalized. The number of applications received for each day (demand), and the number of applications processed per day for each department are defined as random variables that can take any integer value between 1 and 6 (like rolling a die). Initially, all departments have 4 applications in their inbox at the beginning of day 1.

1. Create a mathematical model of this problem in R.
    - You can use runif and round formulas to generate values for the demand and the production. Be careful here!
2. What is the total number of completed applications at the end of day 20?
3. Does that number change as you update/recalculate the random variables?
4. What is the expected number of completed applications per day?

Boston University Metropolitan College