MAT-374: Probability: Random Numbers

Jack Utzerath

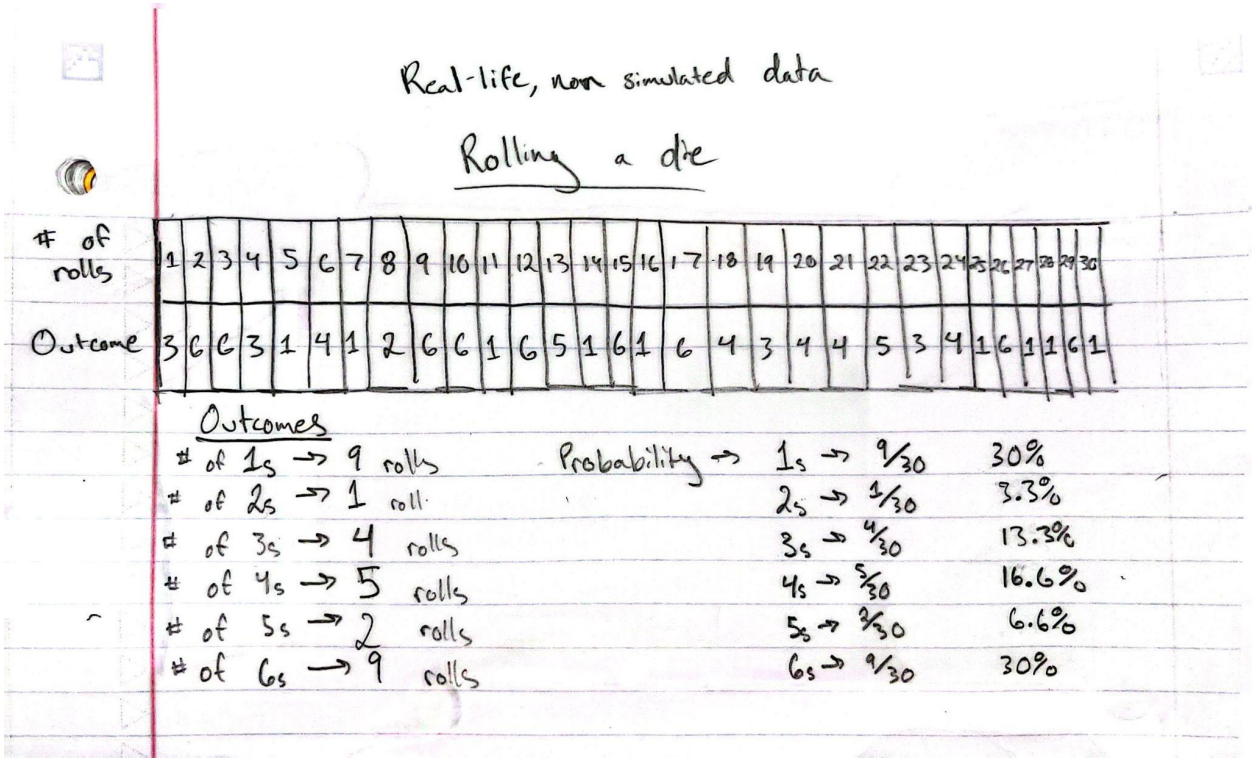
College of Technology, Grand Canyon University

MAT-374 Probability and Statistics

Molly Fleming

10/4/23

1) Goal- The Goal of this experiment is to compare the outcomes of rolling a standard die. The results will be a real-life and simulated data collection with a random number generator. This experiment will hopefully confirm the Law of Large Numbers. Setup- Get a die, and build a program with a random number generator. Design- Compares outcomes of rolling a die in real life and with an RNG to demonstrate the Law of Large Numbers. Procedure- Roll a six-sided die 30 times and record the outcome of each roll. Then code a computer-generated RNG to simulate 30 rolls once more. Then simulate 2000 rolls of six-sided die (mimicking real life) and compare the results.
2) Here is the detailed collection of real-life, non-simulated data. Sample Size = 30



Here I rolled a die 30 times. The most rolled side was 6 and 1. I rolled those 9 times each. The least rolled side is 2, which was only rolled once. Statistically, this is rare with only 3.3% chance of a 2 being rolled in the 30 rolls.

3) For this part I'll first describe the coding project in Matlab.

```
%num rolls
numRolls = 30

% Initialize an array
sideCounts = zeros(1, 6);

% Simulate rolling the die numRolls times
for roll = 1:numRolls
    % Generate a random number between 1 and 6
    outcome = randi(6);

    % Update the count for the corresponding s
    sideCounts(outcome) = sideCounts(outcome)
end

% Calculate the probabilities for each side
probabilities = sideCounts / numRolls;

% Display the results
for side = 1:6
    fprintf('Side %d: Count = %d, Probability
end
```

```
sideCounts = 1×6
     2     1     1     3     5     4

sideCounts = 1×6
     3     1     1     3     5     4

sideCounts = 1×6
     3     2     1     3     5     4

sideCounts = 1×6
     3     2     1     3     6     4

sideCounts = 1×6
     3     2     1     3     7     4

sideCounts = 1×6
     4     2     1     3     7     4

sideCounts = 1×6
     4     2     2     3     7     4

sideCounts = 1×6
     4     3     2     3     7     4

sideCounts = 1×6
     4     3     2     3     7     5

sideCounts = 1×6
     4     4     2     3     7     5

sideCounts = 1×6
     4     4     2     3     7     6

sideCounts = 1×6
     4     5     2     3     7     6

sideCounts = 1×6
     4     5     2     3     7     7

sideCounts = 1×6
     4     6     2     3     7     7

sideCounts = 1×6
     4     6     2     3     8     7

Side 1: Count = 4, Probability = 0.1333
Side 2: Count = 6, Probability = 0.2000
Side 3: Count = 2, Probability = 0.0667
Side 4: Count = 3, Probability = 0.1000
Side 5: Count = 8, Probability = 0.2667
Side 6: Count = 7, Probability = 0.2333
```

I made an array and simulated a dice roll by using randi in MatLab. I updated the side count every single time a roll happened as you can see in the output. I then outputted the side counts and probability of each side being rolled so that I could compare them with the other collections.

4) Detailed collection of simulated data from a correctly programmed or utilized RNG; large scale – sample size at least 1,000

```
%num rolls
numRolls = 1000

% Initialize an array
sideCounts = zeros(1, 6);

% Simulate rolling the die numRolls times
for roll = 1:numRolls
    % Generate a random number between 1 and 6
    outcome = randi(6);

    % Update the count for the corresponding s
    sideCounts(outcome) = sideCounts(outcome)
end

% Calculate the probabilities for each side
probabilities = sideCounts / numRolls;

% Display the results
for side = 1:6
    fprintf('Side %d: Count = %d, Probability
end
```

```
sideCounts = 1×6
     144   168   167   171   167   172

sideCounts = 1×6
     145   168   167   171   167   172

sideCounts = 1×6
     145   168   167   171   167   173

sideCounts = 1×6
     145   168   167   171   167   174

sideCounts = 1×6
     145   168   168   171   167   174

sideCounts = 1×6
     145   168   168   172   167   174

sideCounts = 1×6
     146   168   168   172   167   174

sideCounts = 1×6
     146   169   168   172   167   174

sideCounts = 1×6
     147   169   168   172   167   174

sideCounts = 1×6
     147   169   169   172   167   174

sideCounts = 1×6
     148   169   169   172   167   174

sideCounts = 1×6
     148   169   169   172   167   175

Side 1: Count = 148, Probability = 0.1480
Side 2: Count = 169, Probability = 0.1690
Side 3: Count = 169, Probability = 0.1690
Side 4: Count = 172, Probability = 0.1720
Side 5: Count = 167, Probability = 0.1670
Side 6: Count = 175, Probability = 0.1750
```

5) The probability of each outcome is the relative frequency divided by the total number of rolls in the respective data. For each data set (real-life, small-scale, and large-scale), the calculations are as follows.

Probability distribution

Real life

Outcomes (x)	X	f(x) expected p.	f(x) expected data	P(x) Probability from data
9	1	16.6%	5	30%
1	2			3.3%
4	3			13.3%
5	4			16.6%
2	5			6.6%
9	6			30%

$n = 30$

Small-generated data

Outcomes (x)	X	f(x) expected p.	f(x) expected data	P(x) probability from data
4	1	16.6%	5	13.3%
6	2	16.6%	5	20%
2	3	16.6%	5	6.7%
3	4	16.6%	5	10%
8	5	16.6%	5	26.7%
7	6	16.6%	5	23.3%

$n = 30$

Large-generated data → $n = 1000$

Outcomes (x)	X	f(x)	f(x)	P(x)
148	1	16.6%	~167	14.8%
169	2			16.9%
169	3			16.9%
172	4			17.2%
167	5			16.7%
175	6			17.5%

$n = 1000$

From a first glance at these tables, one can see that the probability from the large generated data was the closest to the theoretical probability. This concept can further be explained by looking at the standard deviation of the probability. Theoretically, there should be no deviation between the probability of each side of the die being rolled.

Standard Deviation of Experimental Probability-

Real Life- 11.4

Small-generated- 7.9

Large-generated- 0.96

6) As the sample size of random experiments increases, the relative frequency of the event gets closer to its theoretical probability. The expected probability of each one of the sides of the die being rolled is ⅙. By looking at the small scale experiment and the large scale experiment, one can see the difference in probability. The experimental probability of the small scale experiment was all over the place compared to the large scale experiment. This can be shown through the standard deviance of the experimental probability. The large scale standard deviance was 0.96 compared to the 7.9 of the small-scale. This is a drastic change in deviance. All in all, this experiment has affirmed the Law of Large Numbers.
7) As Christians, we believe that God created a universe with order and purpose. The concept of randomness points to the manifestation of God's design. This includes the events like rolling a die. The Law of Large Numbers aligns with the Christian worldview because it demonstrates the order and predictability that underlies random events. As we

increase the number of trials, the outcome becomes clearer. This reflects the divine design in the universe. On a larger scale, everything in this world points to a Creator. The DNA, planets, matter, and stars all point to a creator because of how seemingly impossible the things are. This is called general revelation.