Stochastic Super Fun Exercises 1

Rippy

February 3rd, 2020

Problem 1

This is essentially asking for the probability of A given B. This can be written as:

$$P(A|B) = \frac{P(AB)}{P(B)}$$

Which, subbing in numbers

$$P(A|B) = \frac{0.45}{0.6} = 0.75 = 75\%$$

Problem 2

a

Given the premise that

$$P(A) + P(B) + P(C) = 1$$

We know that either A, B, or C *will* happen. So, D will depend on whether A, B, or C happens. Thus, to find the probability of D, we split it into the probability of D for A, B, and C, which is D intersect A, B, or C. Thus we have:

$$P(D) = P(DA) + P(DB) + P(DC)$$

Using the definition that

$$P(D|A) = \frac{P(DA)}{P(A)}$$

Rewriting, we have:

$$P(D|A)P(A) = P(DA)$$

Now, subbing this into our probability of D, we have:

$$P(D) = P(D|A)P(A) + P(D|B)P(B) + P(D|C)P(C)$$

Which basically means, "The probability of D is equal to the probability of D, given A,B, or C occur, multiplied by the respective probabilities of A, B, or C occurring".

b

Where Chrome = C, Purchase = P, Firefox = F, and Other = O

$$P(P) = P(C)P(P|C) + P(F)P(P|F) + P(O)P(P|O)$$

Putting in the numbers, we get:

$$P(P) = (0.68)(0.065) + (0.19)(0.08) + (0.13)(0.045) = 0.06525 = 6.525\%$$

Problem 3

In order to find the total probability that the factory will go at least 15 weeks without an accident, we treat it like a geometric distribution. Knowing the expected value is 10, we use:

$$E[X] = \frac{1}{p}$$

To solve for p, yielding that p is 0.1.

Now, we must find the probability the factory has an accident on week 1, 2, 3, ..., 15 and add them together, to find the probability that the factory makes it through week 15 without an accident. Using the equation for the probability it makes it to week i we have:

$$P[x = i] = p(1-p)^{i-1}$$

Thus, we have:

$$P(15) = \sum_{i=1}^{15} p(1-p)^{i-1} = .794 = 79.4\% \textbf{ Chance to FAIL}$$

Thus, there is a 20.5% chance that they make it at least 15 weeks without incident.

Problem 4

We know that the integral of the density function over the full sample space must equal 1. Thus, we have:

$$\int_0^1 cx = 1$$
$$\frac{cx^2}{2}]_0^1 = 1$$
$$\frac{c}{2} = 1$$
$$c = 2$$

Using c, we integrate to find the probability it is greater than 1/2:

$$\int_{\frac{1}{2}}^1 2x = x^2]_{\frac{1}{2}}^1 = 1 - \frac{1}{4} = 75\%$$

We now find E[X] using:

$$E[X] = \int_0^1 x * 2x = \int_0^1 2x^2 = \frac{2x^3}{3}]_0^1 = \frac{2}{3}$$

Finally, we find the variance using:

$$\begin{aligned} Var(X) &= E[X^2] - (E[X])^2 \\ Var(X) &= \int_0^1 (2x^2)^2 dx - (\frac{2}{3})^2 \\ &= \frac{4}{5}x^5]_0^1 - \frac{4}{9} \\ &= \frac{4}{5} - \frac{4}{9} \\ &= \frac{16}{45} \end{aligned}$$

Problem 5

0.1 a

We know that the whole probabilities must add up to 1. Thus, we know:

$$c + 2c + 3c + 4c = 10c = 1$$

Thus, $c = \frac{1}{10}$ Now, solving for the expected value, we have:

$$\begin{aligned} E[X] &= \sum_i x_i p(x_i) \\ = (1)c + 2(2c) + 3(3c) + 4(4c) &= 30c \\ &= 3 \end{aligned}$$

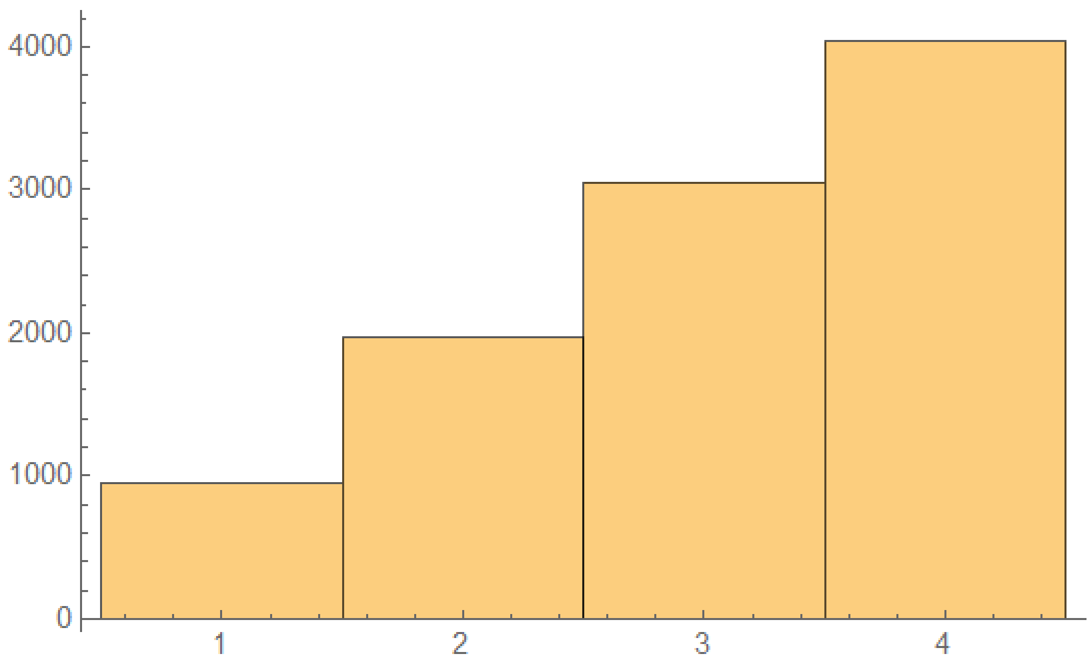
0.2 b

```
X := Module[{},
   u = U;
   value = Which[u < 1/10, 1, u < 3/10, 2, u < 6/10, 3, u < 1, 4];
   Return[value];
  ]
```

In[17]:= `data = Table[X, {10000}];`

In[22]:= `Histogram[data]`

Out[22]=



In[25]:= `Mean[data]`

Out[25]= $\frac{1207}{400}$

In[26]:= `N[1207/400]`

Out[26]= 3.0175

Problem 6

a

Assuming all 9 games are played, (the game does not end after 5 wins), then the win condition for A would be if A won 5 *or more* games. Thus, to find the probability that A wins, we must find the probability A wins exactly 5,6,7,8, and 9 games, then add those together. This is a binomial distribution. Thus, the probability of winning exactly i games is:

$$P[X = i] = \binom{n}{i} p^i (1-p)^{n-i}$$

Now, we add these up for 5,6,7,8, and 9 games, giving us the equation:

$$\sum_{i=5}^{9} \binom{9}{i} p^i (1-p)^{9-i}$$

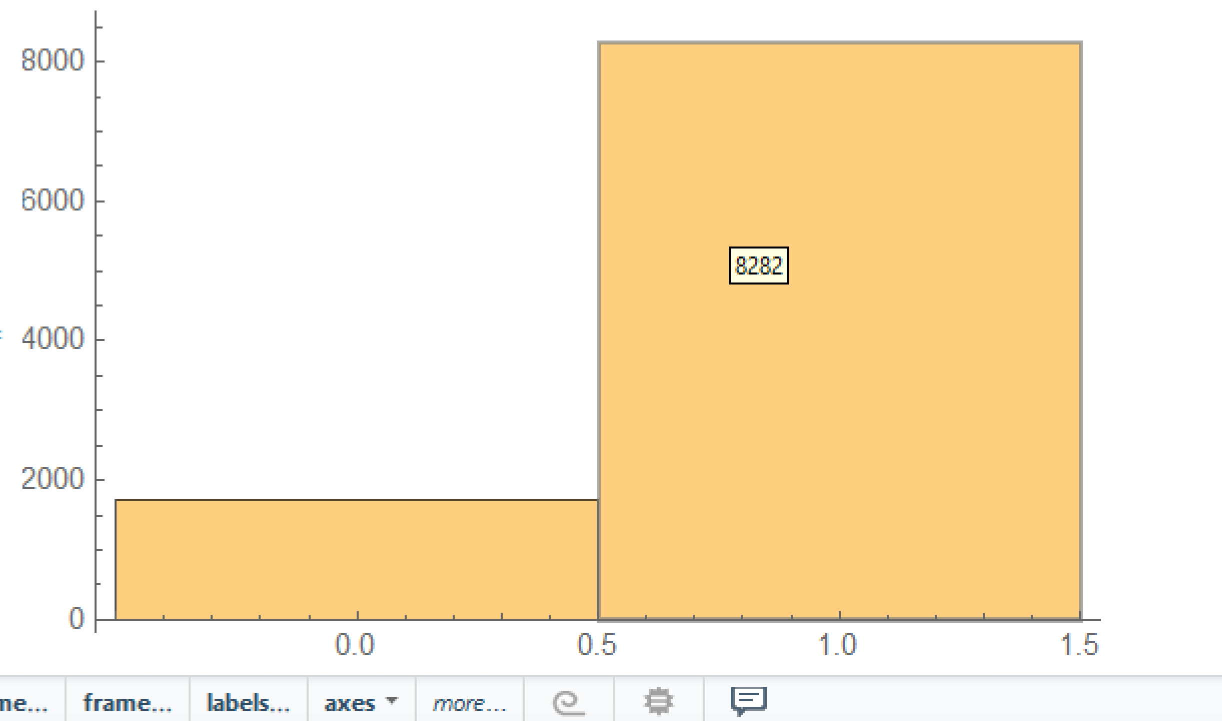
Which, comes out to be a 82.8% probability that A wins.

b

```
X := Module[{},
   u = U;
   value = Which[u < 0.65, 1, u < 1, 0];
   Return[value];
  ]
```

```
In[48]:= data = Table[game2, {10000}];

In[49]:= Histogram[data]
```

Out[49]=



```
In[25]:= Mean[data]

In[43]:= game2 := Module[{},
   u = X + X + X + X + X + X + X + X + X;
   value = Which[u < 5, 0, u < 10, 1];
   Return[value];
  ]
```