Alexander Beckwith

Math 855 - Prob w/ Applications

HW5

In [ ]:
```
import numpy as np
```

1. The joint frequency function of two discrete random variables, X and Y,is given in the following table:

a. Find the marginal frequency functions of X and Y .

b. Find the conditional frequency function of X given Y = 1 and of Y given X = 1.

In [ ]:
```
axes = {"x": 0, "y": 1}
marg_freq = np.array(
    [
        [0.10, 0.05, 0.02, 0.02],
        [0.05, 0.20, 0.05, 0.02],
        [0.02, 0.05, 0.20, 0.04],
        [0.02, 0.02, 0.04, 0.10],
    ]
)
print("We can represent the table as an array:")
print(marg_freq,"\n")

y_marg = np.sum(marg_freq, axis=axes["x"])
x_marg = np.sum(marg_freq, axis=axes["y"])
print("1a | The marginal frequency function of x is", x_marg)
print("1a | The marginal frequency function of y is", y_marg,"\n")
print("We can see that the matrix is symmetrical. In this case, the conditional

y_sum_x1 = y_marg[0]
x_sum_y1 = x_marg[0]

x_when_y1 = [round(n / x_sum_y1, 4) for n in marg_freq[0]]
y_when_x1 = [round(row[0] / y_sum_x1, 4) for row in marg_freq]

print("1b | The conditional freq function of y given x = 1 is", y_when_x1)
print("1b | The conditional freq function of x given y = 1 is", x_when_y1)
```

```
We can represent the table as an array:
[[0.1  0.05 0.02 0.02]
 [0.05 0.2  0.05 0.02]
 [0.02 0.05 0.2  0.04]
 [0.02 0.02 0.04 0.1 ]]

1a | The marginal frequency function of x is [0.19 0.32 0.31 0.18]
1a | The marginal frequency function of y is [0.19 0.32 0.31 0.18]

We can see that the matrix is symmetrical. In this case, the conditional freq
func of X when Y = 1 is the same as the freq func of Y when X = 1

1b | The conditional freq function of y given x = 1 is [0.5263, 0.2632, 0.105
3, 0.1053]
1b | The conditional freq function of x given y = 1 is [0.5263, 0.2632, 0.105
3, 0.1053]
```

1. Three players play 10 independent rounds of a game, and each player has probability 13 of winning each round. Find the joint distribution of the numbers of games won by each of the three players.

In [ ]:
```python
n_players = 3
n_rounds = 10
player_prob = [round(1 / 3, 4) for i in range(n_players)]
print("3 | The prob of each player winning a particular round is", player_prob)
print("The joint distribution expands these probs to include 10 games,")
print("where each value is the prob of choosing a particular winner from a part
joint_prob = np.array([list(player_prob) for j in range(n_rounds)]) / n_rounds
print(joint_prob)
```

```
3 | The prob of each player winning a particular round is [0.3333, 0.3333, 0.3
333]
The joint distribution expands these probs to include 10 games,
where each value is the prob of choosing a particular winner from a particular
game.
[[0.03333 0.03333 0.03333]
 [0.03333 0.03333 0.03333]
 [0.03333 0.03333 0.03333]
 [0.03333 0.03333 0.03333]
 [0.03333 0.03333 0.03333]
 [0.03333 0.03333 0.03333]
 [0.03333 0.03333 0.03333]
 [0.03333 0.03333 0.03333]
 [0.03333 0.03333 0.03333]
 [0.03333 0.03333 0.03333]]
```