Alexander Beckwith

Math 855 - Prob w/ Applications

Exam 1

In [ ]:

```
from math import comb

# local imports
from prob_dist import binom, pois_dist, hgeom, geom
from samplespace import CoinSampleSpace
```

1. (5pts) In a game of poker, what is the probability that a five-card hand from a 52-card deck (i.e., 13 different values and 4 different suits)

(a) (3pts) will contain exact one ace?

(b) (2pts) will contain exact one pair?

5a | This situation can be modelled using a hypergeometric frequency function with parameters:

In [ ]:

```
n = 52      # the number of cards in a deck
r = 4       # the number of aces in a deck
m = 5       # the number of cards to be drawn
k = 1       # the number of aces to be drawn
result = hgeom(k, n, r, m)
print(f"5a | The probability of drawing exactly one ace when drawing 5 cards fr
```

```
5a | The probability of drawing exactly one ace when drawing 5 cards from a st
andard deck is 0.2995
```

5b | The probability of drawing exactly one pair when drawing 5 cards from a standard deck is as follows:

In [ ]:

```
ways_pick5_from52 = comb(52, 5)
# there are 13 different card categories that make pairs
ways_pair_from5 = 0
```

1. (5pts) A drawer of socks contains seven black socks, eight blue socks, and nine green socks. Two socks are chosen in the dark.

(a) (3pts) What is the probability that they match?

(b) (2pts) What is the probability that a black pair is chosen?

In [ ]:

```
# setting values
blk = 7
blu = 8
grn = 9
sox = blk + blu + grn
```

```
# prob of each during first pick
p_blu = blu / sox
p_blk = blk / sox
p_grn = grn / sox

# changes to counts after first pick
blk -= 1
blu -= 1
grn -= 1
sox -= 1

# prob of picking match for each on 2nd pick
p_blu_blu = p_blu * (blu / sox)
p_blk_blk = p_blk * (blk / sox)
p_grn_grn = p_blu * (grn / sox)

# prob of any of these situations happening
p_pair = p_blk_blk + p_blu_blu + p_grn_grn

print(f"2a | The prob that a match is chosen is {round(p_pair, 4)}")
print(f"2b | The prob that a black pair is chosen is {round(p_blk_blk, 4)}")
```

```
2a | The prob that a match is chosen is 0.2935
2b | The prob that a black pair is chosen is 0.0761
```

1. (6pts) Suppose that 5 cards are dealt from a 52-card deck.

(a) (3pts) What is the probability that they contain at least one face cards?

(b) (3pts) Given the first one is a face card, what is the probability of at least one more face cards?

In [ ]:

```
# establish deck and pick counts
n_face_cards = 4 * 3
n_deck_cards = 52
n_not_face = n_deck_cards - n_face_cards

# we want complement, so seek prob of no face cards
p_no_face = 1
p_no_face *= (n_not_face / n_deck_cards)
p_no_face *= (n_not_face / (n_deck_cards - 1))
p_no_face *= (n_not_face / (n_deck_cards - 2))
p_no_face *= (n_not_face / (n_deck_cards - 3))
p_no_face *= (n_not_face / (n_deck_cards - 4))

# the prob of at least one face card is the complement of drawing no face cards
print(f"3a | The prob of drawing at least one face card is {round(1 - p_no_face

# establish deck and pick counts
n_face_cards = 4 * 3 - 1
n_deck_cards = 52 - 1
n_not_face = n_deck_cards - n_face_cards

# we want complement, so seek prob of no face cards
p_no_face = 1
p_no_face *= (n_not_face / n_deck_cards)
p_no_face *= (n_not_face / (n_deck_cards - 1))
p_no_face *= (n_not_face / (n_deck_cards - 2))
p_no_face *= (n_not_face / (n_deck_cards - 3))

# the prob of at least one more face card is the complement of drawing no face
print(f"3b | The prob of drawing at least one more face card is {round(1 - p_nc
```

```
3a | The prob of drawing at least one face card is 0.6717
3b | The prob of drawing at least one more face card is 0.5732
```

1. (6pts) A bin contains 3 different types of disposable flashlights. The probability that a type 1 flashlight will give over 100 hours of use is 0.7, with the corresponding probabilities for type 2 and type 3 flashlights being 0.4 and 0.3, respectively. Suppose that 20 percent of the flashlights in the bin are type 1, 30 percent are type 2 and 50 percent are type 3.

(a) (3pts) What is the probability that a randomly chosen flashlight will give more than 100 hours of use?

(b) (3pts) Given the flashlight lasted over 100 hours, what is the conditional probability that it was a type 2 flashlight.

In [ ]:

```
# prob of picking each flashlight
p_pick_type1 = 0.2
p_pick_type2 = 0.3
p_pick_type3 = 0.5

# prob of each flashlight lasting over 100h
p_100h_type1 = 0.7
p_100h_type2 = 0.4
p_100h_type3 = 0.3

# prob of each flashlight being picked and lasting over 100h
p_type1_pick100h = p_pick_type1 * p_100h_type1
p_type2_pick100h = p_pick_type2 * p_100h_type2
p_type3_pick100h = p_pick_type3 * p_100h_type3

# total prob of picking flashlight and it lasting over 100h
p_total_pick100h = sum([p_type1_pick100h, p_type2_pick100h, p_type3_pick100h])

print(f"4a | The prob of a randomly chosen flashlight lasting over 100h is {rou

p_cond_type2 = p_type2_pick100h / p_total_pick100h
print(f"4b | Given that the flashlight laster over 100h, the prob that it was t
```

```
4a | The prob of a randomly chosen flashlight lasting over 100h is 0.41
4b | Given that the flashlight laster over 100h, the prob that it was type 2 i
s 0.2927
```

1. (6pts) On a multiple-choice exam with 3 possible answers for each of the 5 questions.

(a) (3pts) What is the probability that a student will get 3 correct answers just by guessing?

(b) (3pts) What is the probability that a student will get 4 or more correct answers just by guessing?

In [ ]:

```python
n_questions = 5
p_correct_guess = 1 / 3

# can be modelled using binom distribution
p_3_correct = binom(n_questions, 3, p_correct_guess)
print(f"5a | The prob of getting 3 correct answers by guessing is {round(p_3_cc

# only 5 questions total, so options for 4 or more are:
p_4_correct = binom(n_questions, 4, p_correct_guess)
p_5_correct = binom(n_questions, 5, p_correct_guess)

p_4plus_correct = p_4_correct + p_5_correct
print(f"5b | The prob of getting at least 4 correct answers by guessing is {rou
```

```
5a | The prob of getting 3 correct answers by guessing is 0.1646
5b | The prob of getting at least 4 correct answers by guessing is 0.0453
```

1. (6pts) Suppose that the number of the accidents occurring on a highway each day is a Poisson random variable with parameter λ = 3.

(a) (3pts) Find the probability that 2 or more accidents occur today.

(b) (3pts) Repeat part (a) under the assumption that at least 1 accident occurs today.

In [ ]:

```python
# we must assume that the units for lambda are in accidents / day
# creating distribution
acc_dist = pois_dist(3)

# formatting dist as dictionary
acc_list = acc_dist["x"]
prob_list = acc_dist["prob"]
dict_range = range(len(acc_list))

acc_dict = {acc_list[i]:prob_list[i] for i in dict_range}
prob_0or1 = acc_dict[0] + acc_dict[1]
prob_2plus = 1 - prob_0or1
print(f"6a | The prob of 2 or more accidents is the complement of there being c

prob_not_0 = sum(prob_list[1:])
prob_2plus_given1 = prob_2plus / prob_not_0
print(f"6b | The prob of 2 or more accidents given that one has occurred is the
```

```
6a | The prob of 2 or more accidents is the complement of there being one or t
wo accidents, or 0.8009
6b | The prob of 2 or more accidents given that one has occurred is the prob o
f 2+ accidents / prob of 1+ accidents, 0.8428
```

1. (6pts) When three friends go for coffee, they decide who will pay the check by each flipping a fair coin and then letting the “odd person” pay. If all three flips are the same (so there is no odd person), then they make a second round of flips, and continue to do so until there is an odd person.

(a) (3pts) What is the probability that there is an odd person?

(b) (3pts) What is the probability that exactly 5 rounds of flips are made?

In [ ]:

```
css_3flips = CoinSampleSpace(3).ss
print("Ways to flip 3 coins:", css_3flips)
ways_total = 8
ways_odd = 6
p_odd = ways_odd / ways_total

print(f"7a | The prob of there being an odd person each round of flips is {p_oc

# geometric distribution for prob of rounds til first success
prob_5rounds = geom(5, p_odd)
print(f"7b | The prob of there being 5 rounds to get an odd is {round(prob_5rou
```

```
Ways to flip 3 coins: ['ttt', 'tth', 'tht', 'thh', 'htt', 'hth', 'hht', 'hhh']
7a | The prob of there being an odd person each round of flips is 0.75
7b | The prob of there being 5 rounds to get an odd is 0.0029
```